Tell us about your PDF experience.

Windows Admin Center overview

Article • 08/27/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Windows Admin Center is a remote management tool for Windows Server running anywhere—physical, virtual, on-premises, in Azure, or in a hosted environment—at no extra cost.

To find out what's new, see Release history.

Download now

Download Windows Admin Center from the Microsoft Evaluation Center.

For installation help, see Install. For tips on getting started with Windows Admin Center, see Get started.

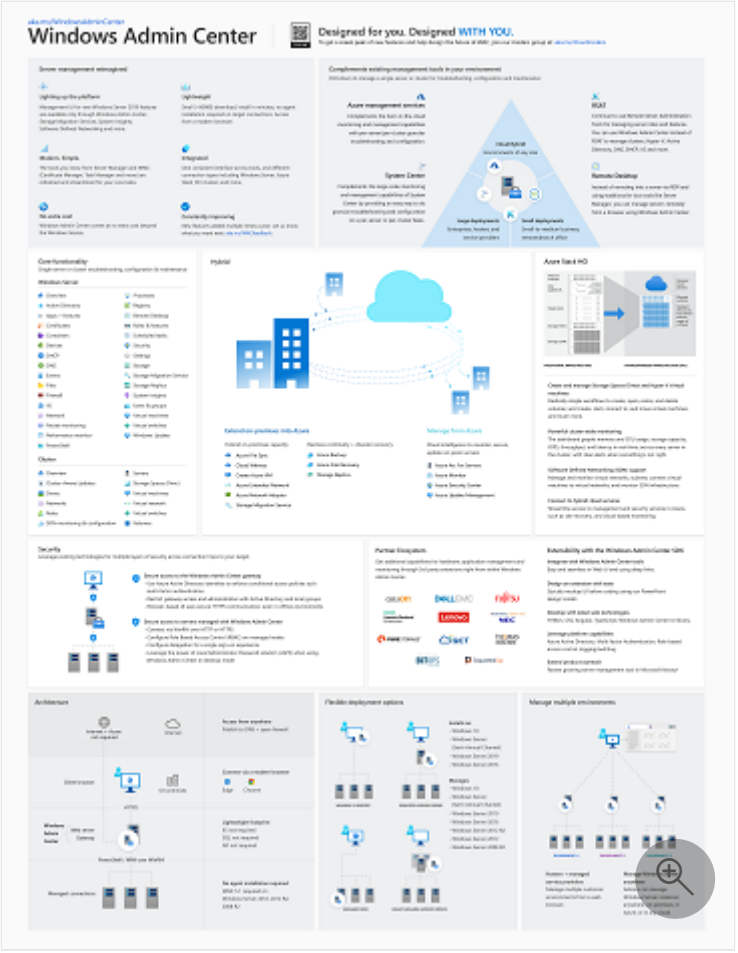
You can update non-preview versions of Windows Admin Center by using Microsoft Update or by manually downloading and installing Windows Admin Center. Each non-preview version of Windows Admin Center is supported until 30 days after the next non-preview version is released. See our support policy for more info.

Windows Admin Center scenarios

Here are a few things you can use Windows Admin Center for:

- **Simplify server management**: Manage your servers and clusters with modernized versions of familiar tools such as Server Manager. Install in under five minutes and manage servers in your environment immediately, no extra configuration required. For details, see What is Windows Admin Center?
- **Work with hybrid solutions**: Integration with Azure helps you optionally connect your on-premises servers with relevant cloud services. For details, see Azure hybrid services.
- **Streamline hyperconverged management**: Streamline management of Azure Local or Windows Server hyperconverged clusters. Use simplified workloads to create and manage VMs, Storage Spaces Direct volumes, Software-Defined Networking and more. For details, see Manage HCI.

Here's a video to give you an overview, followed by a poster giving more details:
https://www.youtube-nocookie.com/embed/WCWxAp27ERk



Download the PDF of this poster

Contents at a glance

Expand table

Understand	Plan
- What is Windows Admin Center? - Frequently asked questions - Case studies - Related management products	- What type of installation is right for you? - User access options

Expand table

Deploy	Configure
- Prepare your environment - Install Windows Admin Center - Enable high availability	- Windows Admin Center settings - User access control and permissions - Shared connections - Extensions - Automate with PowerShell

Expand table

Use	Connect to Azure
- Launch & add connections - Manage servers - Deploy hyperconverged infrastructure - Manage hyperconverged infrastructure - Manage failover clusters - Manage virtual machines - Logging	- Azure hybrid services - Connect Windows Admin Center to Azure - Deploy Windows Admin Center in Azure - Manage Azure VMs with Windows Admin Center

Expand table

Windows Admin Center in Azure	Support
- Manage a Windows Server IaaS VMs - Manage Azure Arc-enabled Servers (preview) - Manage Azure Local clusters (preview)	- Release history - Support policy - Common troubleshooting steps - Known issues

Expand table

Extend
- Overview of extensions - Understanding extensions - Develop an extension - Guides - Publishing extensions

See how customers benefit from Windows Admin Center

"[Windows Admin Center] has decreased our time/effort in managing the management system by over 75%."
- Rand Morimoto, President, Convergent Computing

"Thanks to [Windows Admin Center], we can manage our customers remotely from HTML5 portal without problem and with the full integration with Microsoft Entra ID, we're able to increase the security thanks to the multifactor authentication."
- Silvio Di Benedetto, Founder and Senior Consultant, Inside Technologies

"We have been able to deploy [Server Core] SKUs in a more effective way, improving resource efficiency, security, and automation while still achieving a good degree of productivity and reducing errors that can happen when relying on scripting only."
- *Guglielmo Mengora, Founder and CEO, VaiSulWeb*

"With [Windows Admin Center] customers especially in the SMB market now have an easy to use tool to manage their internal infrastructure. This minimizes administrative efforts and saves a lot of time. And the best of it: there are no additional license fees for [Windows Admin Center]!"
- *Helmut Otto, Managing Director, SecureGUARD*

To read more about companies using Windows Admin Center in their production environments, see Windows Admin Center Case Studies.

Related products

Windows Admin Center is designed for managing a single server or cluster. It complements but doesn't replace existing Microsoft monitoring and management solutions, such as Remote Server Administration Tools (RSAT), System Center, Intune, or Azure Stack.

To learn how Windows Admin Center complements other Microsoft management solutions, see Windows Admin Center and related management solutions from Microsoft.

Stay updated

- Follow us on X (formerly Twitter)
- Read our blogs

Feedback

Was this page helpful? Yes No

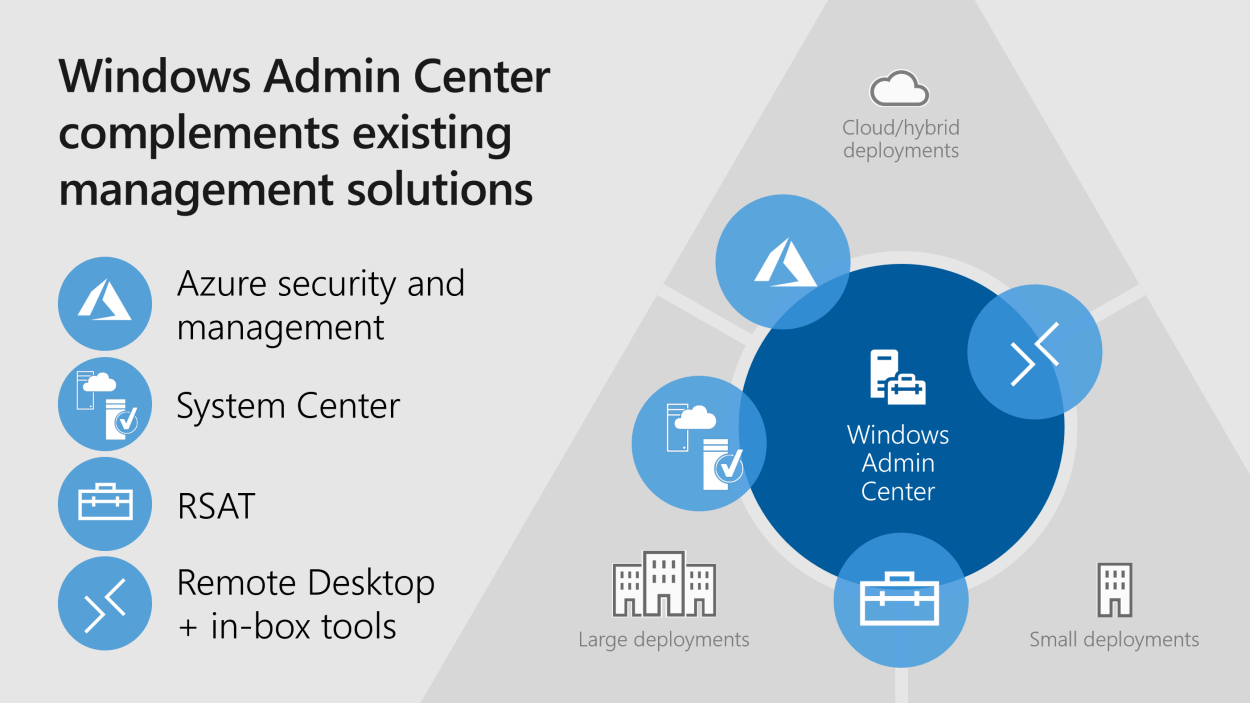
Provide product feedback

What is Windows Admin Center?

Article • 01/02/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Windows Admin Center is a locally-deployed, browser-based management tool set that lets you manage your Windows Clients, Servers, and Clusters without needing to connect to the cloud. Windows Admin Center gives you full control over all aspects of your server infrastructure and is particularly useful for managing servers on private networks that are not connected to the Internet.

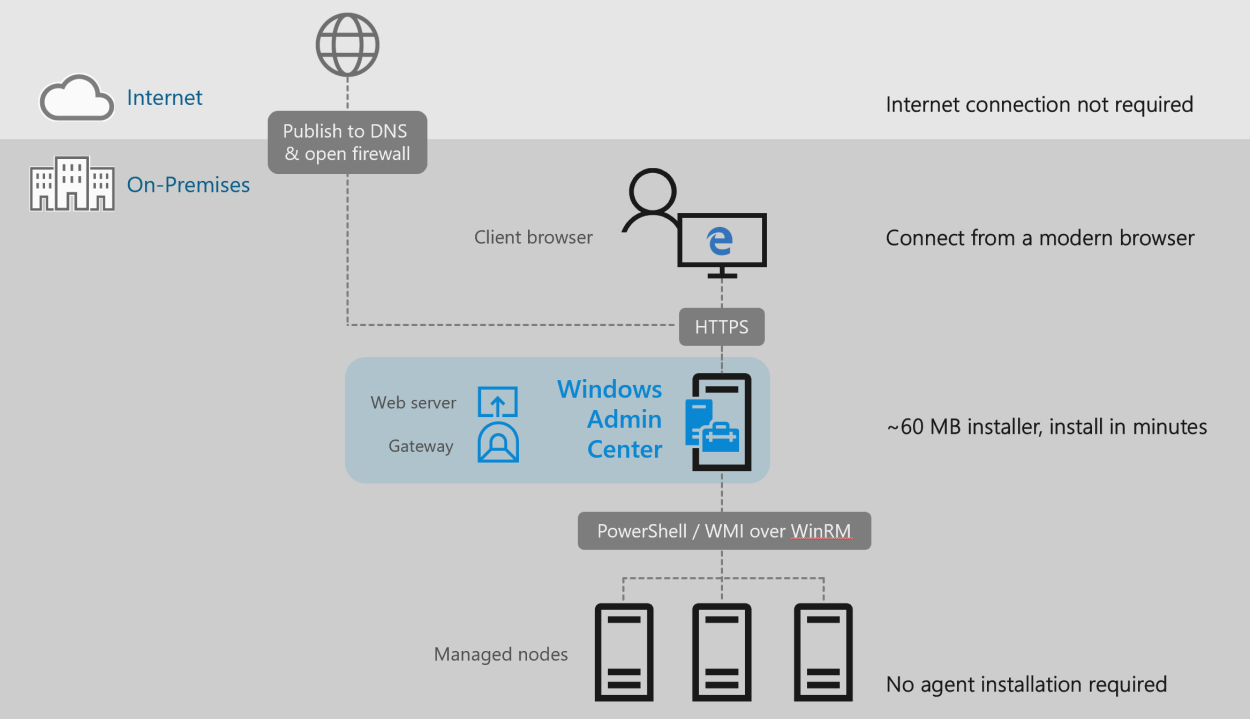
Windows Admin Center is the modern evolution of "in-box" management tools, like Server Manager and MMC. It complements System Center - it's not a replacement.



How does Windows Admin Center work?

Windows Admin Center runs in a web browser and manages Windows Server 2022, Windows Server 2019, Windows Server 2016, Windows Server 2012 R2, Windows Server 2012, Windows 11, Windows 10, Azure Local and more through the **Windows Admin Center gateway** installed on Windows Server or domain-joined Windows 10. The gateway manages servers by using Remote PowerShell and WMI over WinRM. The gateway is included with Windows Admin Center in a single lightweight .msi package that you can download.

The Windows Admin Center gateway, when published to DNS and given access through corresponding corporate firewalls, lets you securely connect to, and manage, your servers from anywhere with Microsoft Edge or Google Chrome.



Learn how Windows Admin Center improves your management environment

Familiar functionality

Windows Admin Center is the evolution of long-standing, well known management platforms like Microsoft Management Console (MMC), built from the ground up for the way systems are built and managed today. Windows Admin Center contains many of the familiar tools you currently use to manage Windows Servers and clients.

Easy to install and use

Install on a Windows 11 computer, and start managing in minutes, or install on a Windows 2022 server acting as a gateway to enable your entire organization to manage computers from their web browser.

Complements existing solutions

Windows Admin Center works with solutions like System Center and Azure management and security, adding to their capabilities to perform detailed, single-machine management tasks.

Manage from anywhere

Publish your Windows Admin Center gateway server to the public Internet, then you can connect to and manage your servers from anywhere, all in a secure manner.

Enhanced security for your management platform

Windows Admin Center has many enhancements that make your management platform more secure. Role-based access control lets you fine-tune which administrators have access to which management features. Gateway authentication options include local groups, local domain-based Active Directory, and cloud-based Microsoft Entra ID. Also, gain insight into management actions performed in your environment.

Azure integration for on-premises and hybrid machines

Windows Admin Center has many points of integration with Azure services, including Microsoft Entra ID, Azure Backup, Azure Site Recovery, and more.

Windows Admin Center in Azure

Using Windows Admin Center in the Azure portal you can manage the Windows Server operating system of your Arc-enabled servers (preview), Azure Local cluster nodes (preview), and Azure VMs.

You can securely manage your machines from anywhere–without needing a VPN, public IP address, or other inbound connectivity to your machine.

Deploy hyper-converged and failover clusters

Windows Admin Center allows for seamless deployment of hyper-converged and failover clusters through an easy-to-use wizard.

Manage hyper-converged clusters

Windows Admin Center offers the best experience for managing hyper-converged clusters - including virtualized compute, storage, and networking components.

Extensibility

Windows Admin Center was built with extensibility in mind from the beginning, with the ability for Microsoft and 3rd party developers to build tools and solutions beyond the

current offerings. Microsoft offers an SDK that enables developers to build their own tools for Windows Admin Center.

> **Tip**
>
> Ready to install Windows Admin Center? **Download now**

Feedback

Was this page helpful? Yes No

Provide product feedback

Windows Admin Center frequently asked questions

FAQ

Here are answers to the most commonly asked questions about Windows Admin Center.

What is Windows Admin Center?

Windows Admin Center is a lightweight, browser-based GUI platform and toolset for IT Admins to manage Windows Server and Windows 10. It's the evolution of familiar in-box administrative tools, such as Server Manager and Microsoft Management Console (MMC) into a modernized, simplified, integrated, and secure experience.

Can I use Windows Admin Center in production environments?

Yes. Windows Admin Center is generally available and ready for broad usage and production deployments. The current platform capabilities and core tools meet Microsoft's standard release criteria and our quality bar for usability, reliability, performance, accessibility, security, and adoption.

Windows Admin Center (non-preview) releases are supported continuously, based on Microsoft's Modern Lifecycle Policy. This means that only the latest version of Windows Admin Center is serviced and supported, and users must stay current by upgrading to the latest Windows Admin Center release within 30 days of availability to remain supported. This policy applies to both the Windows Admin Center platform itself, as well as any released (non-preview) Microsoft extensions published in the Windows Admin Center extension feed. Note that some extensions may be updated more frequently than others, between Windows Admin Center releases.

For info about Windows Admin Center Preview releases, see Windows Insider Preview releases.

How much does it cost to use Windows Admin Center?

Windows Admin Center has no additional cost beyond Windows. You can use Windows Admin Center (available as a separate download) with valid licenses of Windows Server or Windows 10 at no additional cost - it's licensed under a Windows Supplemental EULA.

What versions of Windows Server can I manage with Windows Admin Center?

Windows Admin Center is optimized for Windows Server 2019 to enable key themes in the Windows Server 2019 release: hybrid cloud scenarios and hyper-converged infrastructure management in particular. Although Windows Admin Center will work best with Windows Server 2019, it supports managing a variety of versions that customers already use: Windows Server 2012 and newer are fully supported. There is also limited functionality for managing Windows Server 2008 R2.

Is Windows Admin Center a complete replacement for all traditional in-box and RSAT tools?

No. Although Windows Admin Center can manage many common scenarios, it doesn't completely replace all traditional Microsoft Management Console (MMC) tools. For a detailed look at what tools are included with Windows Admin Center, read more about managing servers in our documentation. Windows Admin Center has the following key capabilities in its Server Manager solution:

- Displaying resources and resource utilization
- Certificate Management
- Managing Devices
- Event Viewer
- File Explorer
- Firewall Management
- Managing Installed Apps
- Configuring Local Users and Groups
- Network Settings
- Viewing/Ending Processes and Creating Process Dumps
- Registry Editing
- Managing Scheduled tasks
- Managing Windows Services

- Enabling/Disabling Roles and Features
- Managing Hyper-V VMs and Virtual Switches
- Managing Storage
- Managing Storage Replica
- Managing Windows Updates
- PowerShell console
- Remote Desktop connection

Windows Admin Center also provides these solutions:

- Computer Management – Provides a subset of the Server Manager features for managing Windows 10 client PCs
- Failover Cluster Manager – Provides support for ongoing management of failover clusters and cluster resources
- Hyper-Converged Cluster Manager – Provides an all-new experience tailored for Storage Spaces Direct and Hyper-V. It features the Dashboard and emphasizes charts and alerts for monitoring.

Windows Admin Center is complementary to and does not replace RSAT (Remote Server Administration Tools) since roles such as Active Directory, DHCP, DNS, IIS do not yet have equivalent management capabilities surfaced in Windows Admin Center.

Can Windows Admin Center be used to manage the free Microsoft Hyper-V Server?

Yes. Windows Admin Center can be used to manage Microsoft Hyper-V Server 2016 and Microsoft Hyper-V Server 2012 R2.

Can I deploy Windows Admin Center on a Windows 10 computer?

Yes, Windows Admin Center can be installed on Windows 10 (version 1709 or later), running in desktop mode. Windows Admin Center can also be installed on a server with Windows Server 2016 or greater in gateway mode, and then accessed via a web browser from a Windows 10 computer. Learn more about installation options.

I've heard that Windows Admin Center uses PowerShell under the hood, can I see the actual scripts that it uses?

Yes! the Showscript feature was added in Windows Admin Center Preview 1806, and is now included in the GA channel.

Are there any plans for Windows Admin Center to manage Windows Server 2008 R2 or earlier?

Windows Admin Center **no longer supports** functionality to manage Windows Server 2008 R2. Windows Admin Center relies on PowerShell capabilities and platform technologies that don't exist in Windows Server 2008 R2 and earlier, making full support infeasible. If you have not yet, Microsoft recommends moving to Azure or upgrading to the latest version of Windows Server.

Are there any plans for Windows Admin Center to manage Linux connections?

We are investigating due to customer demand, but there is currently no locked plan to deliver, and support may consist only of a console connection over SSH.

Which web browsers are supported by Windows Admin Center?

The latest versions of Microsoft Edge (Windows 10, version 1709 or later), Google Chrome, and Microsoft Edge Insider are tested and supported on Windows 10. View browser specific known issues. Other modern web browsers or other platforms are not currently part of our test matrix and are therefore not *officially* supported.

How does Windows Admin Center handle security?

Traffic from the browser to the Windows Admin Center gateway uses HTTPS. Traffic from the gateway to managed servers is standard PowerShell and WMI over WinRM. We support LAPS (Local Administrator Password Solution), resource-based constrained delegation, gateway access control using Active Directory or Microsoft Entra ID, and role-based access control for managing target servers.

Does Windows Admin Center use CredSSP?

Yes, in a few cases Windows Admin Center requires CredSSP. This is required to pass your credentials for authentication to machines beyond the specific server you are targeting for management. For example, if you are managing virtual machines on **server B**, but want to store the vhdx files for those virtual machines on a file share hosted by **server C**, Windows Admin Center must use CredSSP to authenticate with **server C** to access the file share.

Windows Admin Center handles the configuration of CredSSP automatically after prompting for consent from you. Before configuring CredSSP, Windows Admin Center will check to make sure that the system has the recent CredSSP updates.

CredSSP is currently used in the following areas:

- Using disaggregated SMB storage in the virtual machines tool (the example above.)
- Using the Updates tool in either the Failover or Hyper-Converged cluster management solutions, which performs Cluster-Aware Updating

Are there any cloud dependencies?

Windows Admin Center does not require internet access and does not require Microsoft Azure. Windows Admin Center manages Windows Server and Windows instances anywhere: on physical systems, or in virtual machines on any hypervisor, or running in any cloud. Although integration with various Azure services will be added over time, these will be optional value-added features and not a requirement to use Windows Admin Center.

Are there any other dependencies or prerequisites?

Windows Admin Center can be installed on Windows 10 Fall Anniversary Update (1709) or newer, or Windows Server 2016 or newer. To manage Windows Server 2008 R2, 2012, or 2012 R2, installation of Windows Management Framework 5.1 is required on those servers. There are no other dependencies. IIS is not required, agents are not required, SQL Server is not required.

What about extensibility and 3rd-party support?

Windows Admin Center has an SDK available so that anyone can write their own extension. As a platform, growing our ecosystem and enabling partner extensibility has been a key priority since the beginning. Read more about the Windows Admin Center SDK.

Can I manage Hyper-Converged Infrastructure with Windows Admin Center?

Yes. Windows Admin Center supports the management of hyper-converged clusters running Windows Server 2016 or Windows Server 2019. The hyper-converged cluster manager solution in Windows Admin Center was previously in preview but is now **generally available**, with some new functionality in preview. For more information, read more about managing hyper-converged infrastructure.

Does Windows Admin Center require System Center?

No. Windows Admin Center is complementary to System Center, but System Center is not required. Read more about Windows Admin Center and System Center.

Can Windows Admin Center replace System Center Virtual Machine Manager (SCVMM)?

Windows Admin Center and SCVMM are complementary; Windows Admin Center is intended to replace the traditional Microsoft Management Console (MMC) snap ins and the server admin experience. Windows Admin Center is not intended to replace the monitoring aspects of SCVMM. Read more about Windows Admin Center and System Center.

What is Windows Admin Center Preview, which version is right for me?

There are two versions of Windows Admin Center available for download:

Windows Admin Center

- For IT admins who are not able to update frequently or who want more validation time for the releases they use in production, this version is for you. Our current generally available (GA) release is Windows Admin Center 1910.
- Windows Admin Center (non-preview) releases are supported continuously, based on Microsoft's Modern Lifecycle Policy. This means that only the latest version of Windows Admin Center is serviced and supported, and users must stay current by upgrading to the latest Windows Admin Center release within 30 days of availability to remain supported. This policy applies to both the Windows Admin Center platform itself, as well as any released (non-preview) Microsoft extensions published in the Windows Admin Center extension feed. Note that some extensions may be updated more frequently than others, between Windows Admin Center releases.For info about Windows Admin Center Preview releases, see Windows Insider Preview releases.
- To get the latest release, download here.

Windows Admin Center Preview

- For IT admins who want the latest and greatest features on a regular cadence, this version is for you. Our intent is to provide subsequent update releases every month or so. The core platform continues to be production-ready and the license provides production use rights. However, note that you will see the introduction of new tools and capabilities which are clearly marked as PREVIEW and are suitable for evaluation and testing.
- To get the latest Insider Preview release, registered Insiders may download Windows Admin Center Preview directly from the Windows Server Insider Preview download page, under the Additional Downloads dropdown. If you have not yet

registered as an Insider, see Getting Started with Windows Server on the Windows Insiders for Business portal.

Why was "Windows Admin Center" chosen as the final name for "Project Honolulu"?

Windows Admin Center is the official product name for "Project Honolulu" and reinforces our vision of an integrated experience for IT admins across a breadth of core administrative and management scenarios. It also highlights our customer-focus on IT admin user needs as central to how we invest and what we deliver.

Where can I learn more about Windows Admin Center, or get more details on the topics above?

Our launch page is the best starting point and has links to our newly categorized documentation content, download location, how to provide feedback, reference information, and other resources.

What is the version history of Windows Admin Center?

View the version history here.

I'm having an issue with Windows Admin Center, where can I get help?

See our troubleshooting guide and our list of known issues.

Feedback

Was this page helpful? Yes No

Windows Admin Center Case Studies

Article • 12/23/2021 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Learn about how our customers have used Windows Admin Center to improve their productivity and reduce costs.

- Ava6
- Comparex
- Convergent Computing
- FZI Research Center for Information Technology
- GVC Group
- Inside Technologies
- SecureGUARD / COPA-DATA
- VaiSulWeb

Ava6

Ava6 is an IT consulting company that specializes in design, evaluation, and integration of IT infrastructure, specifically virtualization, networking, storage, backup, and cloud computing.

Ava6 uses Windows Server 2016 Core, Hyper-V, Failover Clustering, and S2D.

The Challenge

Ava6's first use case for Windows Admin Center is Hyper-V and Failover Clustering, and is evaluating hyper-converged cluster.

Windows Admin Center helps Ava6 manage Windows Server deployed in Core Edition, especially for driver management. Windows Admin Center gives a better experience for Hyper-V and Failover Cluster to customers, especially to manage VMs, and shows customers that a GUI is coming for the hyper-converged solution.

The Solution

Ava6 has Windows Admin Center deployed as a single instance for VM management.

Ava6's customers have been impressed with Windows Admin Center, and prefer its management capabilities for Hyper-V and Failover Clustering over other options. Before Windows Admin Center, driver management was overly complicated on a Core server. Windows Admin Center has also helped introduce S2D Ava6's customers, with a similar offering to competitors like Nutanix and VMWare.

Comparex

Comparex is an IT service provider and software license management company that has developed services to support management, leverage software products, and enable productivity optimization.

Comparex uses Windows Server 2012R2, Windows Server 2016, Windows 10, Hyper-V, Failover Clusters, Storage Spaces Direct, PowerShell, RDP over HTML, Microsoft Entra ID and Application Proxy, File Servers, and Azure Site Recovery.

The Challenge

Comparex is responsible for running and managing thousands of servers for their customers, and consults with customers to provide the best solution for server management depending on needs.

Comparex was looking for an easy-to-use and remote-accessible server management solution for small to mid-size businesses. Finding a one-stop-shop for server management, in a modern and secure way, was proving to be a major challenge.

The Solution

Comparex is running a server-based Windows Admin Center installation for access to, and management of, their demo lab, which has helped alleviate extra VPN and RDP steps. Comparex's consultants also run Windows Admin Center on their notebooks to help customers in their environment, without the need for explicit RDP access.

Windows Admin Center has saved Comparex time in their daily business, to do more and achieve more. Windows Admin Center has also solved common management challenges with their customers, such as with Hyper-V and Storage Spaces Direct.

Convergent Computing

Convergent Computing is a technology strategy and implementation firm that helps enterprises plan, implement, migrate, and automate systems to improve business

operations.

Convergent Computing uses Windows Server 2016 (Nano, Core, Datacenter, Standard, Hyper-V) and Windows Server 2012R2 (Server Core, Datacenter, Standard, and Hyper-V).

The Challenge

Convergent Computing uses the technologies it recommends to its customers, and it found that Windows Admin Center fits a perfect need for customers with hosted data centers and secured (isolated) on-premise work environments.

Convergent Computing has three distinct environments: hosted, secured, and web operations). These environments run a combination of Windows Server Nano, Core, Cluster, and Hyper-V editions. Windows Admin Center has enabled them to centrally manage multiple servers and services from a single point, providing an optimized footprint and simple management platform.

The Solution

Convergent Computing uses Windows Admin Center to manage three environments with 40+ hosts running 200+ workloads.

Prior to Windows Admin Center, Convergent Computing used a range of tools and technologies to perform management tasks, including System Center and custom scripts. With the goal of minimizing overhead and effort to manage their servers farms, Convergent Computing found that the "thinner" a management layer got, the more complex it became, so they typically ended up with higher overhead for the sake of simplicity. With Windows Admin Center, 20+ hours a month that were spent "managing the management system" are saved, a 75% reduction in time and effort, allowing their operations teams to focus on more valuable tasks including security, compliance, capacity planning, and overall systems optimization.

FZI Research Center for Information Technology

FZI Research Center for Information Technology is a non-profit institution for applied research in information technology and technology transfer.

FZI uses Windows Server 2016, Windows 10, Hyper-V, Storage Spaces Direct, and Failover-Cluster.

The Challenge

FZI was looking for a way to manage Hyper-V Server in a Failover Cluster, and a Hyper-Converged Cluster.

FZI was looking for a centralized way to administrate systems both inside and outside of their domain, with the ability to switch quickly between each of those systems. Windows Admin Center enabled them to accomplish those goals, all in one place.

The Solution

FZI has multiple installations of Windows Admin Center, both as a server installation connecting all of the infrastructure they manage, as well as installations in desktop mode that their administrators use to manage their own testing servers.

Windows Admin Center allows FZI to accelerate the rate they can perform tasks like monthly Windows Updates, connecting to servers via Remote Desktop, and making minor administrative changes quickly.

GVC Group

GVC Group is an online entertainment provider, with over 15,000 servers operated around the globe.

GVC Group uses Windows Server 2016 with many roles (Hyper-V, WSFC, AD-DS, Fileserver, among others), as well as System Center products including SCCM and SCOM.

The Challenge

GVC Group operates in a highly regulated market, with the need to deploy systems to many locations worldwide. Local requirements sometimes require that management of systems is performed in the same country that the server is operated in. Windows Admin Center allows GVC group to leverage global resources and still perform management tasks on local systems.

GVC Group need to deploy servers in many locales and still provide reliable, highly available solutions with minimal effort.

The Solution

GVC Group has deployed Windows Admin Center in a virtualized environment, with a load balancer, to enable management of servers in remote locations.

GVC Group has seen increased productivity due to less management hops for administrators. This has reduced their cost due to the down-scaling of local terminal servers.

Inside Technologies

Inside Technologies is a globally-focused IT consulting firm that provides application development, enterprise solutions, and infrastructure services. They specialize in applying new and unique approaches with Microsoft solutions to meet customer's needs.

Inside Technologies uses a wide variety of Windows Server technologies, including Hyper-V and Storage.

The Challenge

Inside Technologies primary needed a way to provide their customers with new tools in an always-connected environment that was integrated with high security, and without the need to use VPN.

Inside Technologies have customers where uptime is critical, and were looking for a tool that allowed them to manage their servers easily without exposing RDP and without adding complexity with VPN.

The Solution

Windows Admin Center is deployed by Inside Technologies on a single server to manage all assets in their environment.

Inside Technologies is using Windows Admin Center to manage their customers remotely, and with integration of Azure Activity Directory, with increased security thanks to Multi-Factor Authentication. The dashboards on Windows Admin Center offer improved visibility into the state of each role on each server, and ease of management down to the finest detail with PowerShell. Inside Technologies has realized a reduction in time spent to manage servers vs. using different consoles for each server role.

SecureGUARD / COPA-DATA

SecureGUARD GmbH helps companies solve complex IT problems in security and cloud infrastructure with a series of products and custom engineering services. SecureGUARD builds IT security appliances and Microsoft Windows Server based

appliances, more recently building rack-level converged appliances for cloud infrastructures. SecureGUARD is a Microsoft Gold Partner in Application Development and Datacenter.

COPA-DATA develops the software solution “zenon” that allows for end-to-end industrial IoT solutions - from the field level up to the cloud and to mobile devices. COPA-DATA has deployed systems worldwide to companies in the Food & Beverage, Energy & Infrastructure, Automotive and Pharmaceutical sectors.

The Challenge

COPA-DATA implemented a 4-node hyper converged cluster designed and implemented by SecureGUARD GmbH in late 2017 to host their internal infrastructure and test servers.

COPA-DATA found itself limited by its old IT infrastructure. Specifically, monitoring and operating their infrastructure with approx. 60 virtual machines became a real challenge for the IT department.

The Solution

COPA-DATA uses Windows Admin Center to monitor and operate their physical and virtual server infrastructure, all accessible from one browser window. Tasks like provisioning a new virtual machine or viewing performance data of a physical server or VM are now done with a simple mouse click in the Windows Admin Center web interface.

With Windows Admin Center, COPA-DATA has an easy to use tool to manage their internal infrastructure. This minimizes administrative efforts and saves a lot of time, without any additional license fees.

VaiSulWeb

VaiSulWeb is a web hosting and IaaS/PaaS services provider that also provides development and integration services.

VaiSulWeb uses Windows Server 2016 and 2012 R2, with Hyper-V, Failover Clustering, Storage Spaces Direct, File Servers, SQL Server, IIS, and WSL.

The Challenge

VaiSulWeb was looking to leverage the agility and efficiency of Server Core when deploying resources, while maintaining ease of management for those resources. VaiSulWeb uses a combination of PowerShell automation for simple tasks and UI for more complicated ones.

VaiSulWeb has gained increased confidence deploying Windows Server Core with the increased ability to manage with Windows Admin Center, plus the ability to extend Windows Admin Center for a customized experience.

The Solution

VaiSulWeb deploys in a wide variety of scenarios, including failover clustering and hyper-converged. Windows Admin Center is deployed on both Windows 10 workstations and Windows Servers to manage their assets.

VaiSulWeb has been able to deploy Windows Server Core more effectively, while improving resource efficiency, security, and automation. VaiSulWeb has achieved improved productivity and reduced errors versus management with scripts only.

Feedback

Was this page helpful? Yes No

Provide product feedback

Windows Admin Center and related management solutions from Microsoft

Article • 11/01/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Windows Admin Center is the evolution of traditional in-box server management tools for situations where you might have used Remote Desktop (RDP) to connect to a server for troubleshooting or configuration. It's not intended to replace other existing Microsoft management solutions; rather it complements these solutions, as described below.

Remote Server Administration Tools (RSAT)

Remote Server Administration Tools (RSAT) is a collection of GUI and PowerShell tools to manage optional roles and features in Windows Server. RSAT has many capabilities that Windows Admin Center doesn't have. We may add some of the most commonly used tools in RSAT to Windows Admin Center in the future. Any new Windows Server role or feature that requires a GUI for management will be in Windows Admin Center.

Intune

Intune is a cloud-based enterprise mobility management service that lets you manage iOS, Android, Windows, and macOS devices, based on a set of policies. Intune focuses on enabling you to secure company information by controlling how your workforce accesses and shares information. In contrast, Windows Admin Center is not policy-driven, but enables ad-hoc management of Windows 10 and Windows Server systems, using remote PowerShell and WMI over WinRM.

Azure Stack

Azure Stack is a hybrid cloud platform that lets you deliver Azure services from your data center. Azure Stack is managed using PowerShell or the administrator portal, which is similar to the traditional Azure portal used to access and manage traditional Azure services. Windows Admin Center isn't intended to manage the Azure Stack infrastructure, but you can use it to manage Azure IaaS virtual machines (running Windows Server 2016, Windows Server 2012 R2, or Windows Server 2012) or troubleshoot individual physical servers deployed in your Azure Stack environment.

System Center

System Center is an on-premises data center management solution for deployment, configuration, management, monitoring your entire data center. System Center lets you see the status of all the systems in your environment, while Windows Admin Center lets you drill down into a specific server to manage or troubleshoot it with more granular tools.

Expand table

Windows Admin Center	System Center
Reimagined "in-box" platform & tools	**Datacenter management & monitoring**
Included with Windows Server license – **no additional cost**, just like MMC and other traditional in-box tools	**Comprehensive** suite of solutions for additional value across your environment and platforms
Lightweight, browser-based remote management of Windows Server instances, **anywhere**; alternative to RDP	Manage & monitor **heterogeneous** systems **at scale**, including Hyper-V, VMware, and Linux
Deep single-server & single-cluster drill-down for troubleshooting, configuration & maintenance	Infrastructure provisioning; automation and self-service; infrastructure and workload monitoring **breadth**
Optimized management of **individual** 2–4 node **HCI** clusters, integrating Hyper-V, Storage Spaces Direct, and SDN	Deploy & manage Hyper-V, Windows Server clusters at **datacenter scale** from **bare metal** with SCVMM
Monitoring on HCI only; cluster health service stores history. Extensible platform for 1st and 3rd party **admin tool extensions**	**Extensible** & **scalable monitoring** platform in SCOM, with alerting, notifications, third-party workload monitoring; SQL for history
Easiest bridge to **hybrid**; onboard and use a variety of Azure services for data protection, replication, updates and more	**Built-in** data protection, replication, updates (DPM/VMM/SCCM). Hybrid integration with Log Analytics and Service Map
Lights up platform features of Windows Server: Storage Migration Service, Storage Replica, System Insights, etc.	**Additional platforms**: Automation in Orchestrator/SMA.Integrations with SCSM & other service management tools

Each delivers targeted value independently; better together with complementary capabilities.

Feedback

Was this page helpful? Yes No

Provide product feedback

What is the Windows Admin Center modernized gateway?

Article • 01/02/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

In December 2023, the Windows Admin Center modernized gateway was released to public preview through the Windows Server Insider program. This release is in addition to the latest generally available release of Windows Admin Center, version 2311.

The modernized gateway is a significant backend upgrade of the Windows Admin Center product. The Windows Admin Center backend hosts the authorization structure, PowerShell services, and gateway plug-ins and plays a critical role in every Windows Admin Center experience.

What's new

Upgrade from .NET 4.6.2 to .NET Core

The biggest upgrade in this release is the backend upgrade from .NET framework 4.6.2 to .NET Core. This upgrade brings enhanced performance, security, and improved cryptography. It also includes support for HTTP/2, reducing latency and enhancing the responsiveness of Windows Admin Center.

Updated installer

While modernizing our gateway, we also made the installer more flexible by providing increased customization options including network access settings, selecting trusted hosts, providing a fully qualified domain name (FQDN) for your gateway machine, and more. For more details about the installer, read on to the Installing the modernized gateway section.

Multi-process, micro-service based architecture

The modernized gateway also uses microservice architecture. Prior to this upgrade, Windows Admin Center performed all tasks in a single process. With this new model, we start one process for Windows Admin Center on application startup that serves as a process manager. As you use Windows Admin Center, more subprocesses are spun up to perform specific tasks.

Additionally, gateway plug-ins that are compatible with the modernized gateway will also run their own collection of subprocesses under the Windows Admin Center service manager to perform their functions.

Changing from a monolithic service to a microservice model helps the modernized gateway be more flexible, scalable, and resilient.

Kestrel HTTP web server

Previously, Windows Admin Center utilized Katana components, including a web server, on the backend. With the modernized gateway, we've shifted to an ASP.NET Core Kestrel web server.

Kestrel is the recommended web server for ASP.NET Core applications. Additionally, Kestrel supports the HTTP/2 web protocol, where previously we had only supported HTTP1.1 with the Katana components. The upgrade from HTTP1.1 to HTTP/2 brings reduced latency to our application and increased responsiveness through enhanced features like multiplexing and server push.

How this affects extensions

Gateway plug-in extensions are most impacted by the changes to our modernized gateway. Windows Admin Center gateway plug-ins enable API communication from the UI of your tool or solution to a target node. Windows Admin Center hosts a gateway service that relays commands and scripts from gateway plug-ins to be executed on target nodes. The gateway service can be extended to include custom gateway plug-ins that support protocols other than the default ones (PowerShell and WMI).

Because gateway plug-ins communicate with Windows Admin Center's backend to enable API communication, gateway plug-in code can include components written with the .NET framework version 4.6.2, which won't function with .NET Core. This code needs to be updated to use the .NET Core framework.

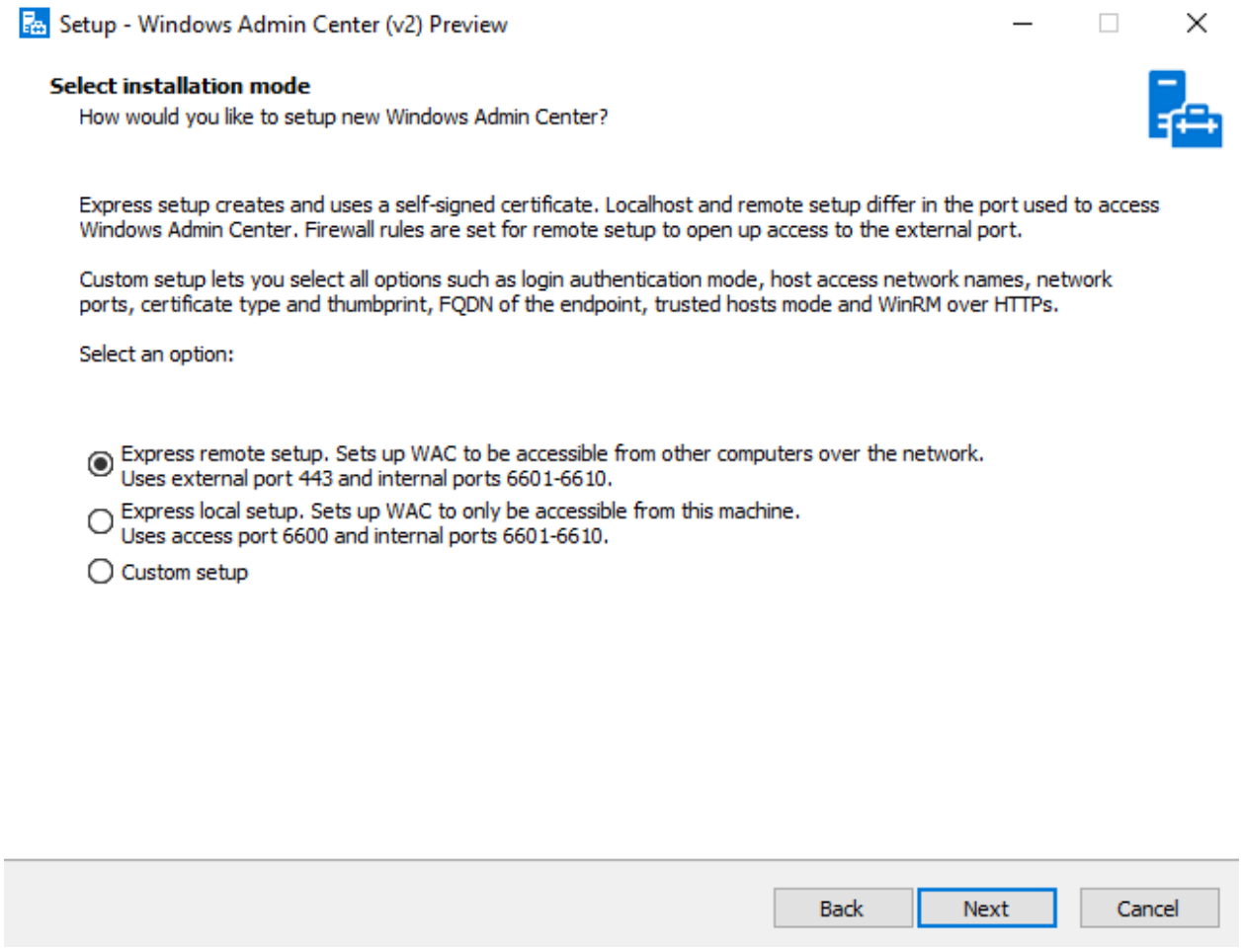
Additionally, we've modified the way plug-ins work with our modernized gateway. Instead of developing a C# class that implements the `IPlugIn` interface from the `Microsoft.ManagementExperience.FeatureInterfaces` namespace to extend the gateway plug-in, extensions will now be written in the form of ASP.NET MVC controllers. These controllers have increased flexibility compared to the simple C# class and extensive documentation.

View our developer documentation to learn more about gateway plug-in development in Windows Admin Center.

Installing the modernized gateway

With our modernized gateway, we've made changes to our installer to offer more flexibility to the user.

When running the installer, you're presented with three different installation modes: express localhost setup, express remote setup, and custom setup.



If you would like to configuration options for internal and external network ports, endpoint FQDN, WinRM over HTTPS, and more, continue to the Custom setup section. If you're okay with the defaults, continue to the Express setup options section.

Express setup options

Two of the three installation modes for the modernized gateway are express modes—express localhost setup and express remote setup. Express localhost setup can also be referred to as local client setup. For all other installation types, use express remote setup. Both express setup options don't allow for the configuration of the following features:

- Sign-in authentication mode
- Host access network names

- Internal and external network ports
- Certificate type and thumbprint
- Endpoint FQDN
- Trusted hosts mode
- WinRM over HTTPS

If you would like to configure any of these features, use the Custom setup option instead.

If you select the express localhost setup option, WAC will be accessible through port 6600 and will use internal ports 6601-6610.

If you select the express remote setup option, WAC will be accessible through port 443 and will use internal ports 6601-6610.

Custom setup

Selecting custom setup allows you to configure all Windows Admin Center setup options, including:

- **Network access** – This page allows you to select how you'll be using Windows Admin Center. You can choose to restrict WAC access to other users by selecting localhost access only or allow remote access through machine name of FQDN.
- **Port numbers** – This page allows you to select the ports that will be reserved for Windows Admin Center. WAC uses one external port for its primary processes. Other processes use internal ports. There are two internal processes by default, but extensions can define their own services that will require port access. By default, the internal range is 10 ports.
- **Select TLS certificate** – This page allows you to select Self-Signed certificates or an official TLS certificate that Windows Admin Center should use. Self-Signed certificates include Self-signed CA root certificates and TLS certificates that work with the latest Edge/Chrome browser.
- **Fully qualified domain name** – This page allows you to provide a fully qualified domain name for network access. This name must match the name on the TLS certificate.
- Trusted hosts – This page allows you to select which type of remote hosts you'd like to manage. You can choose to manage only trusted domain computers or allow access to non-domain joined machines.
- **WinRM over HTTPS** - This page allows you to select whether to use HTTPS for WinRM communication. WinRM communicates over HTTP by default.

Troubleshooting installation

If your installation failed, or Windows Admin Center fails to open after install, try uninstalling and reinstalling. This issue can also happen if you have an older version of a modernized gateway build installed, and you're trying to update to a newer version. To uninstall, follow the instructions in the Uninstalling the modernized gateway section.

Extension support

The extension feed for the modernized gateway isn't configured. Extensions not included in the Windows Admin Center installer, including external partner extensions, aren't available unless you add an extension feed.

The following extensions are available upon install of the modernized gateway build:

- Apps & features
- Azure Backup
- Azure File Sync
- Azure hybrid center
- Azure Kubernetes Service
- Certificates
- Cluster Creation
- Cluster Manager
- Developer Guide
- Devices
- Events
- Failover cluster tools
- Files & file sharing
- Firewall
- Local users & groups
- Network Controller tools and SDN Virtual networks
- Networks
- Packet monitoring
- Performance Monitor
- PowerShell
- Processes
- Registry
- Remote Desktop
- Roles & features
- Scheduled tasks
- SDN Gateway connections
- SDN Infrastructure
- SDN Logical networks

- SDN Network security groups
- Security
- Server Manager and Computer Management
- Services
- Storage
- Storage Migration Service
- Storage Replica
- System Insights
- Updates
- Virtual machines and switches

There are some extensions that won't function even when they're added as part of a new extension feed. For more information, see Known issues.

Uninstalling the modernized gateway

If you have to uninstall the Windows Admin Center modernized gateway, perform one of the following actions:

- In the **Apps & Features** page of your gateway machine settings, select **Windows Admin Center (v2) Preview** from the program list and then select uninstall.
- Navigate to the folder where the Windows Admin Center modernized gateway is installed (default directory is `C:\ProgramData\Microsoft\Windows\Start Menu\Programs\Windows Admin Center (v2)`) and select **Uninstall Windows Admin Center (v2)**
- Run `C:\Program Files\WindowsAdminCenter\unins000.exe`

Running the installer again doesn't provide an uninstall option at this time. To ensure your installation was removed successfully, check if a WindowsAdminCenter folder exists in `C:\ProgramData` or `C:\Program Files`. If it doesn't exist in either location, your application is successfully uninstalled.

During the uninstallation process, everything put on the machine during installation is removed, apart from the Windows Admin Center modernized gateway .exe installer file. If you have another build of Windows Admin Center also installed at the time of your modernized gateway uninstallation, no files or properties of the other build are touched during the uninstallation process. There are no interdependencies between the two installations.

Known issues

There are the following known issues in the modernized gateway build of Windows Admin Center.

If you encounter an issue not described on this page, let us know. To help us address the issue, specify that the issue was occurring in the modernized gateway build.

PowerShell

The account for the PowerShell session in this tool always defaults to the user signed into the Windows Admin Center gateway, even if different management credentials were specified when remoting to a connection.

Extensions not supported

Even with an added extension feed, the following extensions currently don't work with the modernized gateway:

- Dell OpenManage
- Lenovo XClarity Integrator
- Fujitsu ServerView RAID
- Fujitsu Software Infrastructure Manager (ISM)
- Fujitsu ServerView Health
- Pure Storage FlashArray

Frequently asked questions

Find answers to the frequently asked questions about using the Windows Admin Center modernized gateway.

Can you install a Windows Admin Center modernized gateway build when you already have an existing build of Windows Admin Center installed?

Yes, you can install a modernized gateway build of Windows Admin Center side-by-side with a legacy gateway build as long as you don't choose the same ports for both installations.

Can I change the ports my Windows Admin Center modernized gateway installation is using after install?

Yes, In the Program Files for Windows Admin Center, we've included a PowerShell module called `Microsoft.WindowsAdminCenter.Configuration.psm1`. This module allows you to modify your WAC configuration after installation and can be found in the **PowerShellModules** folder of your installation (`C:\Program Files\WindowsAdminCenter\PowerShellModules\` by default).

To change the ports your Windows Admin Center instance is using, run the following command:

```PowerShell
Set-WACHttpsPort -Wacport <port> -ServicePortRangeStart <port> -
ServicePortRangeEnd <port>
```

Can I change configuration settings other than port settings after install?

Yes, you can use the PowerShell module `Microsoft.WindowsAdminCenter.Configuration.psm1` to change your configuration settings. It can be found in the **PowerShellModules** folder of your installation.

Why aren't all of these changes in the 2311 release?

To ensure the best quality experience, we require customer and developer feedback before these changes are made generally available.

Are all the features from the 2311 release available in this build?

Yes. Read more about the 2311 release of Windows Admin Center.

Next steps

- Download and install the modernized gateway build of Windows Admin Center from the Windows Server Insider Program
- Get started with Windows Admin Center

Feedback

Was this page helpful? Yes No

Provide product feedback

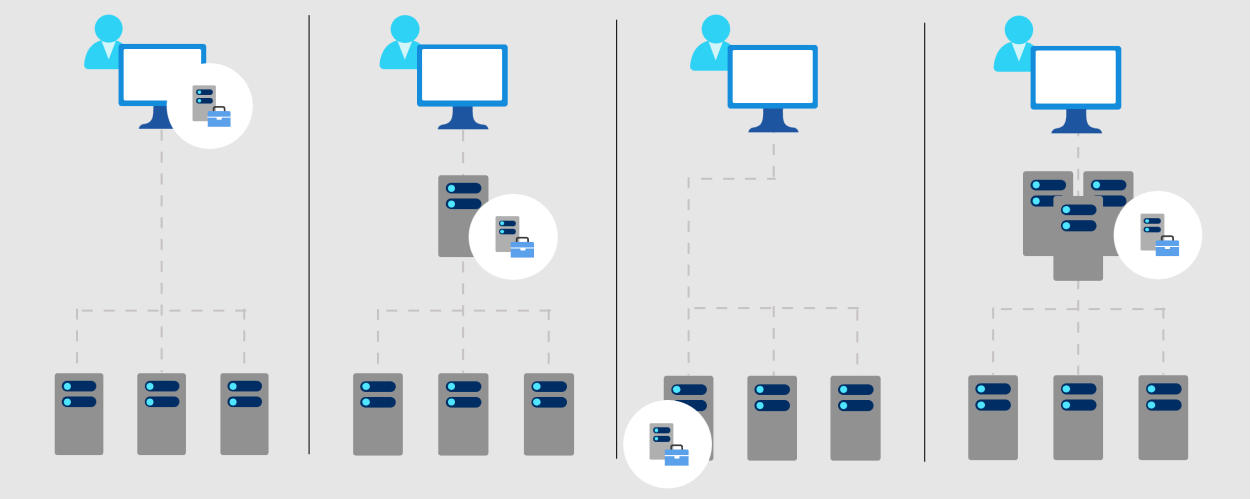
What type of installation is right for you?

Article • 02/05/2025 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

This topic describes the different installation options for Windows Admin Center, including installing on a Windows 10 PC or a Windows server for use by multiple admins. To install Windows Admin Center on a VM in Azure, see Deploy Windows Admin Center in Azure.

We don't recommend using Windows Admin Center for local management of the same server on which it's installed. To manage a server, use Windows Admin Center to connect to the server remotely from a management PC or other server.

Installation: Types



Expand table

Local client	Gateway server	Managed server	Failover cluster
Install on a local Windows 10 or 11 client that has connectivity to the managed servers. Great for quick start, testing, ad-hoc or small scale scenarios.	Install on a designated gateway server and access from any client browser with connectivity to the gateway server. Great for large-scale scenarios.	Install directly on a managed server for the purpose of remotely managing the server or a cluster in which it's a member node. Great for distributed scenarios.	Deploy in a failover cluster to enable high availability of the gateway service. Great for production environments to ensure resiliency of your management service.

Installation: Supported operating systems

You can **install** Windows Admin Center on the following Windows operating systems:

Expand table

Platform	Installation mode
Windows 11	Local client
Windows 10	Local client
Windows Server Semi-Annual Channel	Gateway server, managed server, failover cluster
Windows Server 2025	Gateway server, managed server, failover cluster
Windows Server 2022	Gateway server, managed server, failover cluster
Windows Server 2019	Gateway server, managed server, failover cluster
Windows Server 2016	Gateway server, managed server, failover cluster

To operate Windows Admin Center:

- **In local client scenario:** Launch the Windows Admin Center gateway from the Start menu and connect to it from a client web browser by accessing `https://localhost:6516`.
- **In other scenarios:** Connect to the Windows Admin Center gateway on a different machine from a client browser via its URL, e.g., `https://servername.contoso.com`

> ⚠ **Warning**
>
> Installing Windows Admin Center on a Domain controller is not supported. Read more about domain controller security best practices.

> ⓘ **Note**
>
> Port usage and defaults for Windows Admin Center installations on the modernized gateway vary from what is mentioned above. Read more about what's different in the modernized gateway.

Installation: Supported web browsers

Microsoft Edge (including Microsoft Edge insider) and Google Chrome are tested and supported on Windows 10. Other web browsers—including Firefox—are not currently part of our test matrix and are therefore not *officially* supported. These browsers may have problems running Windows Admin Center. For example, Firefox has its own certificate store, so you must import the `Windows Admin Center Client` certificate into Firefox to use Windows Admin Center on Windows 10. For more details, see browser-specific known issues.

Management target: Supported operating systems

You can **manage** the following Windows operating systems using Windows Admin Center:

Expand table

Version	**Manage *node* via *Server Manager***	**Manage via *Cluster Manager***
Windows 11	Yes (via Computer Management)	N/A
Windows 10	Yes (via Computer Management)	N/A
Windows Server Semi-Annual Channel	Yes	Yes
Windows Server 2025	Yes	Yes
Windows Server 2022	Yes	Yes
Windows Server 2019	Yes	Yes
Windows Server 2016	Yes	Yes, with latest cumulative update
Microsoft Hyper-V Server 2016	Yes	Yes
Windows Server 2012 R2	Yes	Yes
Microsoft Hyper-V Server 2012 R2	Yes	Yes
Windows Server 2012	Yes	Yes

Version	Manage *node* via *Server Manager*	Manage via *Cluster Manager*
Azure Local, version 22H2 and higher	Yes	Yes

> **Note**
>
> Windows Admin Center requires PowerShell features that are not included in Windows Server 2012 and 2012 R2. If you will manage these with Windows Admin Center, you will need to install Windows Management Framework (WMF) version 5.1 or higher on those servers.
>
> Type `$PSVersiontable` in PowerShell to verify that WMF is installed, and that the version is 5.1 or higher.
>
> If WMF is not installed, you can download WMF 5.1.

High availability

> **Warning**
>
> Windows Admin Center version 2410 doesn't support high availability. If you are using high availability, you won't be able to update.

You can enable high availability of the gateway service by deploying Windows Admin Center in an active-passive model on a failover cluster. If one of the nodes in the cluster fails, Windows Admin Center gracefully fails over to another node, letting you continue managing the servers in your environment seamlessly.

Learn how to deploy Windows Admin Center with high availability.

> **Tip**
>
> Ready to install Windows Admin Center? Download now

Feedback

Was this page helpful? Yes No

Provide product feedback

User access options with Windows Admin Center

Article • 06/16/2023 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

When deployed on Windows Server, Windows Admin Center provides a centralized point of management for your server environment. By controlling access to Windows Admin Center, you can improve the security of your management landscape.

> ⓘ Note
>
> Windows Admin Center as an application depends on the operating system and infrastructure for security. Windows Admin Center does not implement, monitor, or enforce a security boundary.

Gateway access roles

Windows Admin Center defines two roles for access to the gateway service: gateway users and gateway administrators.

> ⓘ Note
>
> Access to the gateway does not imply access to the target servers visible to the gateway. To manage a target server, a user must connect with credentials that have administrative privileges on the target server.

Gateway users can connect to the Windows Admin Center gateway service in order to manage servers through that gateway, but they cannot change access permissions nor the authentication mechanism used to authenticate to the gateway.

Gateway administrators can configure who gets access as well as how users will authenticate to the gateway.

> ⓘ Note
>
> If there are no access groups defined in Windows Admin Center, the roles will reflect the Windows account access to the gateway server.

Configure gateway user and administrator access in Windows Admin Center.

Identity provider options

Gateway administrators can choose either of the following:

- Active Directory/local machine groups
- Microsoft Entra ID as the identity provider for Windows Admin Center

Smartcard authentication

When using Active Directory or local machine groups as the identity provider, you can enforce smartcard authentication by requiring users who access Windows Admin Center to be a member of additional smartcard-based security groups. Configure smartcard authentication in Windows Admin Center.

Conditional access and multi-factor authentication

By requiring Microsoft Entra authentication for the gateway, you can leverage additional security features like conditional access and multi-factor authentication provided by Microsoft Entra ID. Learn more about configuring conditional access with Microsoft Entra ID.

Role-based access control

By default, users require full local administrator privileges on the machines they wish to manage using Windows Admin Center. This allows them to connect to the machine remotely and ensures they have sufficient permissions to view and modify system settings. However, some users may not need unrestricted access to the machine to perform their jobs. You can use **role-based access control** in Windows Admin Center to provide such users with limited access to the machine instead of making them full local administrators.

Role-based access control in Windows Admin Center works by configuring each managed server with a PowerShell Just Enough Administration endpoint. This endpoint defines the roles, including what aspects of the system each role is allowed to manage and which users are assigned to the role. When a user connects to the restricted endpoint, a temporary local administrator account is created to manage the system on their behalf. This ensures that even tools which do not have their own delegation model can still be managed with Windows Admin Center. The temporary account is automatically removed when the user stops managing the machine through Windows Admin Center.

When a user connects to a machine configured with role-based access control, Windows Admin Center will first check if they are a local administrator. If they are, they will receive the full Windows Admin Center experience with no restrictions. Otherwise, Windows Admin Center will check if the user belongs to any of the pre-defined roles. A user is said to have *limited access* if they belong to a Windows Admin Center role but are not a full administrator. Finally, if the user is neither an administrator nor a member of a role, they will be denied access to manage the machine.

Role-based access control is available for the Server Manager and Failover Cluster solutions.

Available roles

Windows Admin Center supports the following end-user roles:

Expand table

Role name	Intended use
Administrators	Allows users to use most of the features in Windows Admin Center without granting them access to Remote Desktop or PowerShell. This role is good for "jump server" scenarios where you want to limit the management entry points on a machine.
Readers	Allows users to view information and settings on the server, but not make changes.
Hyper-V Administrators	Allows users to make changes to Hyper-V virtual machines and switches, but limits other features to read-only access.

The following built-in extensions have reduced functionality when a user connects with limited access:

- Files (no file upload or download)
- PowerShell (unavailable)
- Remote Desktop (unavailable)
- Storage Replica (unavailable)

At this time, you cannot create custom roles for your organization, but you can choose which users are granted access to each role.

Preparing for role-based access control

To leverage the temporary local accounts, each target machine needs to be configured to support role-based access control in Windows Admin Center. The configuration process involves installing PowerShell scripts and a Just Enough Administration endpoint on the machine using Desired State Configuration.

If you only have a few computers, you can easily apply the configuration individually to each computer using the role-based access control page in Windows Admin Center. When you set up role-based access control on an individual computer, local security groups are created to control access to each role. You can grant access to users or other security groups by adding them as members of the role security groups.

For an enterprise-wide deployment on multiple machines, you can download the configuration script from the gateway and distribute it to your computers using a Desired State Configuration pull server, Azure Automation, or your preferred management tooling.

Feedback

Was this page helpful? Yes No

Provide product feedback

Prepare your environment for Windows Admin Center

Article • 12/23/2021 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

There are some Server versions that need additional preparation before they are ready to manage with Windows Admin Center:

- Windows Server 2012 and 2012 R2
- Microsoft Hyper-V Server 2016
- Microsoft Hyper-V Server 2012 R2

There are also some scenarios where port configuration on the target server may need to be modified before managing with Windows Admin Center.

Prepare Windows Server 2012 and 2012 R2

Install WMF version 5.1 or higher

Windows Admin Center requires PowerShell features that are not included by default in Windows Server 2012 and 2012 R2. To manage Windows Server 2012 or 2012 R2 with Windows Admin Center, you will need to install WMF version 5.1 or higher on those servers.

Type `$PSVersiontable` in PowerShell to verify that WMF is installed, and that the version is 5.1 or higher.

If it is not installed, you can download and install WMF 5.1.

Prepare Microsoft Hyper-V Server 2016

To manage Microsoft Hyper-V Server 2016 with Windows Admin Center, there are some Server roles you'll need to enable before you can do so.

To manage Microsoft Hyper-V Server 2016 with Windows Admin Center:

1. Enable Remote Management.
2. Enable File Server Role.
3. Enable Hyper-V Module for PowerShell.

Step 1: Enable Remote Management

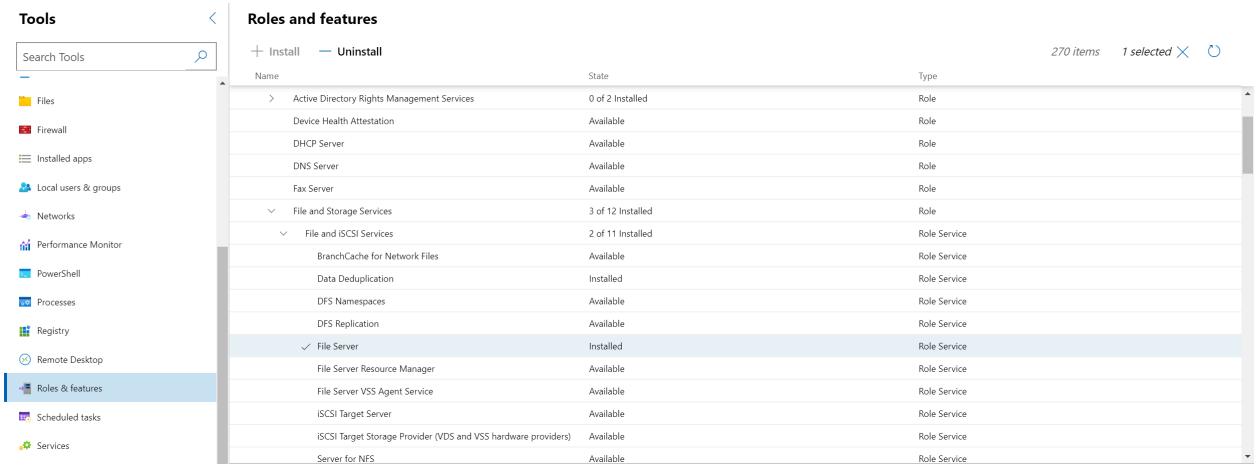
To enable remote management in Hyper-V Server:

1. Log into Hyper-V Server.
2. At the **Server Configuration** (SCONFIG) tool, type **4** to configure remote management.
3. Type **1** to enable Remote Management.
4. Type **4** to return to the main menu.

Step 2: Enable File Server Role

To enable File Server Role for basic file sharing and remote management:

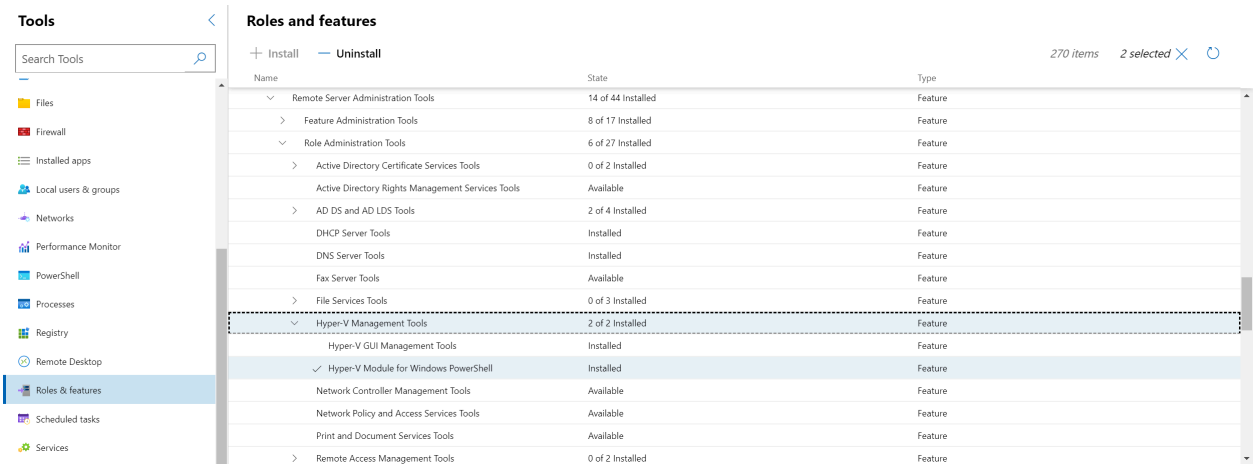
1. Click **Roles and Features** in the **Tools** menu.
2. In **Roles and Features**, find **File and Storage Services**, and check **File and iSCSI Services** and **File Server**:



Step 3: Enable Hyper-V Module for PowerShell

To enable Hyper-V Module for PowerShell features:

1. Click **Roles and Features** in the **Tools** menu.
2. In **Roles and Features**, find **Remote Server Administration Tools** and check **Role Administration Tools** and **Hyper-V Module for PowerShell**:



Microsoft Hyper-V Server 2016 is now ready for management with Windows Admin Center.

Prepare Microsoft Hyper-V Server 2012 R2

To manage Microsoft Hyper-V Server 2012 R2 with Windows Admin Center, there are some Server roles you'll need to enable before you can do so. In addition, you will need to install WMF version 5.1 or higher.

To manage Microsoft Hyper-V Server 2012 R2 with Windows Admin Center:

1. Install Windows Management Framework (WMF) version 5.1 or higher
2. Enable Remote Management
3. Enable File Server Role
4. Enable Hyper-V Module for PowerShell

Step 1: Install Windows Management Framework 5.1

Windows Admin Center requires PowerShell features that are not included by default in Microsoft Hyper-V Server 2012 R2. To manage Microsoft Hyper-V Server 2012 R2 with Windows Admin Center, you will need to install WMF version 5.1 or higher.

Type `$PSVersiontable` in PowerShell to verify that WMF is installed, and that the version is 5.1 or higher.

If it is not installed, you can download WMF 5.1.

Step 2: Enable Remote Management

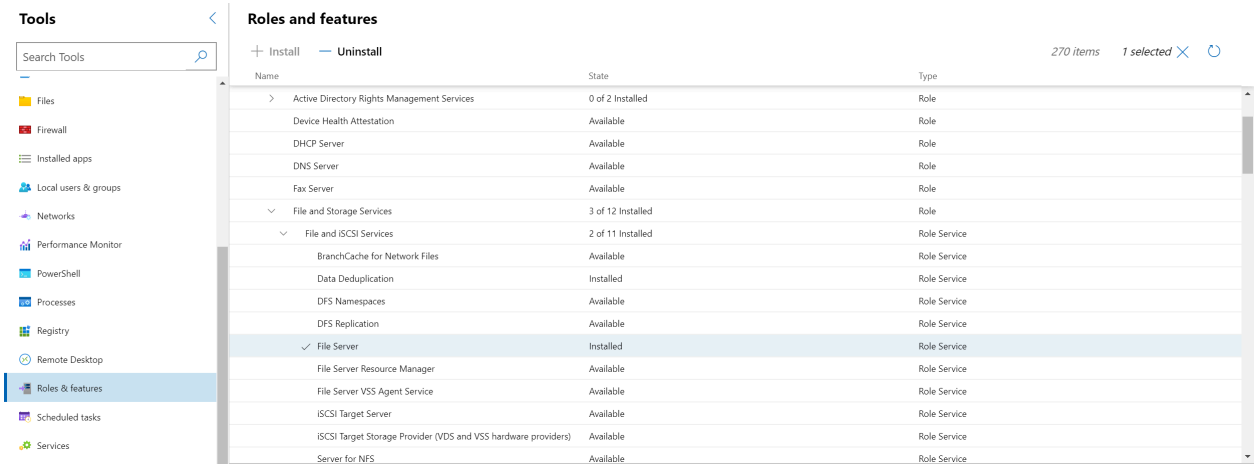
To enable Hyper-V Server remote management:

1. Log into Hyper-V Server.
2. At the **Server Configuration** (SCONFIG) tool, type **4** to configure remote management.
3. Type **1** to enable remote management.
4. Type **4** to return to the main menu.

Step 3: Enable File Server Role

To enable File Server Role for basic file sharing and remote management:

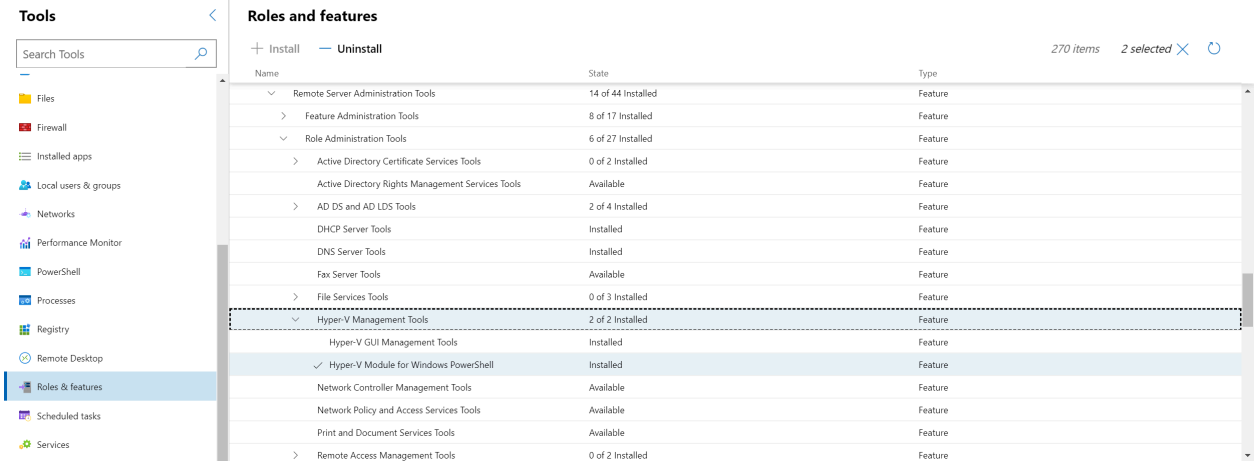
1. Click **Roles and Features** in the **Tools** menu.
2. In **Roles and Features**, find **File and Storage Services** and check **File and iSCSI Services** and **File Server**:



Step 4: Enable Hyper-V Module for PowerShell

To enable Hyper-V Module for PowerShell features:

1. Click **Roles and Features** in the **Tools** menu.
2. In **Roles and Features**, find **Remote Server Administration Tools** and check **Role Administration Tools** and **Hyper-V Module for PowerShell**:



Microsoft Hyper-V Server 2012 R2 is now ready for management with Windows Admin Center.

Port configuration on the target server

Windows Admin Center uses the SMB file sharing protocol for some file copying tasks, such as when importing a certificate on a remote server. For these file copy operations to succeed, the firewall on the remote server must allow inbound connections on port 445. You can use the Firewall tool in Windows Admin Center to verify the incoming rule for 'File Server Remote Management (SMB-In)' is set to allow access on this port.

> **Tip**
>
> Ready to install Windows Admin Center? **Download now**

Feedback

Was this page helpful? Yes No

Provide product feedback

Windows Admin Center network requirements

Article • 01/12/2023 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

This article describes the networking requirements for using Windows Admin center to manage your servers and clusters.

Networking configuration

Windows Admin Center communicates outbound securely to endpoints over TCP port 443. By default, the Windows Admin Center gateway and browser uses the default route to the internet to perform actions. You can optionally configure the gateway to use a proxy server if your network requires it.

Endpoints need to be opened on two sources:

- The gateway - this is the server or client machine where the Windows Admin Center gateway (.msi) is installed.
- The browser - this is the machine where the Windows Admin Center web service is being accessed from.

If outbound connectivity is restricted by your firewall or proxy server, make sure the URLs listed below are not blocked.

Gateway URLs

Expand table

URL	Description	When required
`aka.ms`	Acquiring and maintaining Windows Admin Center	Always
`download.microsoft.com`	Acquiring and maintaining Windows Admin Center	Always
`pkgs.dev.azure.com`	Extension management	Always
`*.vsblob.vsassets.io`	Extension management	Always
`login.microsoftonline.com`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure Public

URL	Description	When required
		Cloud
`graph.microsoft.com`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure Public Cloud
`graph.windows.net`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure Public Cloud
`management.azure.com`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure Public Cloud
`login.microsoftonline.us`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure US Government Cloud
`management.usgovcloudapi.net`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure US Government Cloud
`graph.windows.net`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure US Government Cloud
`management.core.usgovcloudapi.net`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure US Government Cloud
`login.partner.microsoftonline.cn`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure China Cloud
`management.chinacloudapi.cn`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure China Cloud
`graph.chinacloudapi.cn`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure China Cloud
`management.core.chinacloudapi.cn`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure China Cloud

Browser URLs

Expand table

URL	Description	When required
`winadmincenterassets.blob.core.windows.net`	Extension management	Always
`js.monitor.azure.com`	Extension management	Always
`nuget.org`	Extension management	Always
`announcements.blob.core.windows.net`	Extension management	Always
`browser.events.data.microsoft.com`	Acquiring and maintaining Windows Admin Center	Optionally
`login.microsoftonline.com`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure Public Cloud
`graph.microsoft.com`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure Public Cloud
`graph.windows.net`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure Public Cloud
`portal.azure.com`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure Public Cloud
`login.microsoftonline.us`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure US Government Cloud
`management.usgovcloudapi.net`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure US Government Cloud
`graph.windows.net`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure US Government Cloud

URL	Description	When required
`portal.azure.us`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure US Government Cloud
`login.partner.microsoftonline.cn`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure China Cloud
`management.chinacloudapi.cn`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure China Cloud
`graph.chinacloudapi.cn`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure China Cloud
`portal.azure.cn`	Azure endpoints for communication	When using Azure Hybrid Services in the Azure China Cloud

Next steps

- Prepare your environment
- Install Windows Admin Center

Feedback

Was this page helpful? Yes No

Provide product feedback

Install Windows Admin Center

Article • 03/06/2025 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

This topic describes how to install Windows Admin Center on a Windows client machine or on a Windows Server 2025 remote server so that multiple users can access Windows Admin Center using a web browser.

> **Tip**
>
> New to Windows Admin Center? **Learn more about Windows Admin Center** or **Download now**.

Determine your installation type

Review the installation options which includes the supported operating systems. To install Windows Admin Center on a virtual machine in Azure, see Deploy Windows Admin Center in Azure.

Prerequisites

To install Windows Admin Center, you need the following prerequisites:

- A Windows PC or server to install Windows Admin Center on.
- Administrative privileges or equivalent permissions on the machine you're installing Windows Admin Center on.
- Optional: An SSL certificate used for *Server Authentication (1.3.6.1.5.5.7.3.1)*. You can use a self-signed certificate for testing, but you should always use a certificate from a trusted certificate authority for production environments. If you don't have a certificate, you can use the Windows Admin Center installer to generate a self-signed certificate. The certificate is valid for 60 days.
- If you're using the Server Core experience, download the Windows Admin Center installer from the Windows Admin Center Evaluation Center to a location on the machine you want to install Windows Admin Center on.

Installing Windows Admin Center

To install Windows Admin Center, perform the following steps:

Desktop Experience

To install Windows Admin Center on your machine running the Windows Server Desktop Experience, follow these steps:

1. Open the **Start** menu and enter **Windows Admin Center Setup** into the search bar to search for Windows Admin Center.
2. Select the **Windows Admin Center Setup** app from the **Best match** list.
3. On the **Get started with Windows Admin Center** window, if you agree to the license terms, select **Next** to continue.
4. The latest installer download starts automatically and is saved to the *Downloads* folder, when complete, select **Install**. Selecting **Install** initiates the installer from the Downloads folder.
5. On the **Welcome to the Windows Admin Center setup wizard** window, select **Next** to continue.
6. On the **License Terms and Privacy Statement** window, if you agree to the terms select **I accept these terms and understand the privacy statement**, then select **Next** to start the installation process.
7. In the **Select installation mode** window, select **Express setup**, then select **Next**.
8. In the **Select TLS certificate** window, select the option that matches your needs, then select **Next**.

   > **Note**
   >
   > You must select which Transport Layer Security (TLS) certificate Windows Admin Center should use. If you already have a certificate, it must be installed in the `LocalMachine\My` certificates store. If you're installing Windows Admin Center for testing purposes only, the installer can generate a self-signed certificate that expires after 60 days.

9. In the **Automatic updates** window, select your preferred update option, then select **Next**.

10. In the **Send diagnostic data to Microsoft** window, select your preference, then select **Next**.
11. Review the **Ready to install** window, select **Install** to start the installation process.
12. After the installation process finishes, select **Start Windows Admin Center**, then select **Finish**.
13. Sign in as an administrator to start using Windows Admin Center.

You've now installed Windows Admin Center on your machine.

Upgrade to a new version

You can update non-preview versions of Windows Admin Center by using Microsoft Update or a manual installation.

Windows Admin Center preserves your settings when you upgrade to the latest version. Upgrading Insider Preview versions of Windows Admin Center isn't supported, we recommend you do a new installation by installing the latest version of Preview and starting over.

Update the certificate used by Windows Admin Center

When you have Windows Admin Center deployed as a service, you must provide a certificate for HTTPS. To update this certificate, use the following these steps.

> **ⓘ Important**
>
> When changing your certificate, you need to restart the Windows Admin Center service for the changes to take effect.

1. Sign-in to your machine. If you're on Server core, from the SConfig menu, enter option **15**, then press `Enter` to open a PowerShell session. If you're on the desktop experience, remote desktop into your VM and launch PowerShell.
2. Import the Windows Admin Center configuration PowerShell module using the following command:

```
```

```
PowerShell

Import-Module
"$env:ProgramFiles\WindowsAdminCenter\PowerShellModules\Microsoft.Windo
wsAdminCenter.Configuration"
```

3. Apply the new certificate using the following command, making sure to replace `<subject name>` with the subject name of the certificate:

```
PowerShell

Set-WACCertificateSubjectName -SubjectName "<subject name>"
```

> **Tip**
>
> Make sure the certificate uses a unique subject name. Alternatively, you can use the SHA1 thumbprint of the certificate using the **Thumbprint** parameter.

4. Updated the certificate access control list to grant permissions for the *Network Service* account to access the certificate. Use the following command, replacing `<subject name>` with the subject name of the certificate:

```
PowerShell

Set-WACCertificateAcl -SubjectName "<subject name>"
```

5. Restart the Windows Admin Center service using the following command:

```
PowerShell

Restart-Service -Name WindowsAdminCenter
```

You've now updated the certificate used by Windows Admin Center.

Feedback

Was this page helpful? Yes No

Provide product feedback

Deploy Windows Admin Center with high availability

Article • 06/16/2023 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

You can deploy Windows Admin Center in a failover cluster to provide high availability for your Windows Admin Center gateway service. The solution provided is an active-passive solution, where only one instance of Windows Admin Center is active. If one of the nodes in the cluster fails, Windows Admin Center gracefully fails over to another node, letting you continue managing the servers in your environment seamlessly.

Learn about other Windows Admin Center deployment options.

Prerequisites

- A failover cluster of 2 or more nodes on Windows Server 2016, 2019, or 2022. Learn more about deploying a Failover Cluster.
- A cluster shared volume (CSV) for Windows Admin Center to store persistent data that can be accessed by all the nodes in the cluster. 10 GB will be sufficient for your CSV.
- High-availability deployment script from Windows Admin Center HA Script zip file. Download the .zip file containing the script to your local machine and then copy the script as needed based on the guidance below.
- Recommended, but optional: a signed certificate .pfx & password. You don't need to have already installed the certificate on the cluster nodes - the script will do that for you. If you don't supply one, the installation script generates a self-signed certificate, which expires after 60 days.

Install Windows Admin Center on a failover cluster

1. Copy the `Install-WindowsAdminCenterHA.ps1` script to a node in your cluster. Download or copy the Windows Admin Center .msi to the same node.
2. Connect to the node via RDP and run the `Install-WindowsAdminCenterHA.ps1` script from that node with the following parameters:

    - `-clusterStorage`: the local path of the Cluster Shared Volume to store Windows Admin Center data.

- `-clientAccessPoint`: choose a name that you will use to access Windows Admin Center. For example, if you run the script with the parameter `-clientAccessPoint contosoWindowsAdminCenter`, you will access the Windows Admin Center service by visiting `https://contosoWindowsAdminCenter.<domain>.com`
- `-staticAddress`: Optional. One or more static addresses for the cluster generic service.
- `-msiPath`: The path for the Windows Admin Center .msi file.
- `-certPath`: Optional. The path for a certificate .pfx file.
- `-certPassword`: Optional. A SecureString password for the certificate .pfx provided in `-certPath`
- `-generateSslCert`: Optional. If you don't want to provide a signed certificate, include this parameter flag to generate a self-signed certificate. Note that the self-signed certificate will expire in 60 days.
- `-portNumber`: Optional. If you don't specify a port, the gateway service is deployed on port 443 (HTTPS). To use a different port specify in this parameter. Note that if you use a custom port (anything besides 443), you'll access the Windows Admin Center by going to https://<clientAccessPoint>:<port>.

> **Note**
>
> The `Install-WindowsAdminCenterHA.ps1` script supports `-WhatIf` and `-Verbose` parameters

Examples

Install with a signed certificate:

PowerShell

```
$certPassword = Read-Host -AsSecureString
.\Install-WindowsAdminCenterHA.ps1 -clusterStorage
"C:\ClusterStorage\Volume1" -clientAccessPoint "contoso-ha-gateway" -msiPath
".\WindowsAdminCenter.msi" -certPath "cert.pfx" -certPassword $certPassword
-Verbose
```

Install with a self-signed certificate:

PowerShell

```
.\Install-WindowsAdminCenterHA.ps1 -clusterStorage
"C:\ClusterStorage\Volume1" -clientAccessPoint "contoso-ha-gateway" -msiPath
".\WindowsAdminCenter.msi" -StaticAddress (local ip address) -
generateSslCert -Verbose
```

Update an existing high availability installation

Use the same `Install-WindowsAdminCenterHA.ps1` script to update your HA deployment, without losing your connection data.

Update to a new version of Windows Admin Center

When a new version of Windows Admin Center is released, simply run the `Install-WindowsAdminCenterHA.ps1` script again with only the `msiPath` parameter:

PowerShell

```
.\Install-WindowsAdminCenterHA.ps1 -msiPath '.\WindowsAdminCenter.msi' -
Verbose
```

Update the certificate used by Windows Admin Center

You can update the certificate used by a HA deployment of Windows Admin Center at any time by providing the new certificate's .pfx file and password.

PowerShell

```
$certPassword = Read-Host -AsSecureString
.\Install-WindowsAdminCenterHA.ps1 -certPath "cert.pfx" -certPassword
$certPassword -Verbose
```

You may also update the certificate at the same time you update the Windows Admin Center platform with a new .msi file.

PowerShell

```
$certPassword = Read-Host -AsSecureString
.\Install-WindowsAdminCenterHA.ps1 -msiPath ".\WindowsAdminCenter.msi" -
certPath "cert.pfx" -certPassword $certPassword -Verbose
```

Uninstall

To uninstall the HA deployment of Windows Admin Center from your failover cluster, pass the `-Uninstall` parameter to the `Install-WindowsAdminCenterHA.ps1` script.

PowerShell

```
.\Install-WindowsAdminCenterHA.ps1 -Uninstall -Verbose
```

Troubleshooting

Logs are saved in the temp folder of the CSV (for example, C:\ClusterStorage\Volume1\temp).

Feedback

Was this page helpful? Yes No

Provide product feedback

Windows Admin Center Settings

Article • 06/16/2023 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Windows Admin Center settings consist of user-level and gateway-level settings. A change to a user-level setting only affects the current user's profile, while a change to a gateway-level setting affects all users on that Windows Admin Center gateway.

User settings

User-level settings consist of the following sections:

- Account
- Language/Region
- Personalization
- Suggestions

In the **Account** tab, users can review the credentials they have used to authenticate to Windows Admin Center. If Microsoft Entra ID is configured to be the identity provider, the user can log out of their Microsoft Entra account from this tab.

In the **Language/Region** tab, users can change the language and region formats displayed by Windows Admin Center.

In the **Personalization** tab, users can toggle to a dark UI theme.

In the **Suggestions** tab, users can toggle suggestions about Azure services and new features.

Development settings

Development settings in Windows Admin Center consist of the following sections:

- Advanced
- Performance profile

The **Advanced** tab gives Windows Admin Center extension developers additional capabilities.

The **Performance profile** tab lets you collect performance data about your Windows Admin Center session.

Gateway settings

Gateway-level settings consist of the following sections:

- Access
- Diagnostics & feedback
- Extensions
- General
- Internet Access
- Proxy
- Register
- Updates
- Shared Connections
- WebSocket validation

Only gateway administrators are able to see and change these settings. Changes to these settings change the configuration of the gateway and affect all users of the Windows Admin Center gateway.

The **Access** tab lets administrators configure who can access the Windows Admin Center gateway, as well as the identity provider used to authenticate users. Learn more about controlling access to the gateway.

In the **Diagnostics & feedback** tab, users can choose how much diagnostic data they want to send to Microsoft.

In the **Extensions** tab, administrators can install, uninstall, or update gateway extensions. Learn more about extensions.

In the **General** tab, users can select to have their UI session of Windows Admin Center expire after some period of inactivity.

The **Internet Access** tab lets administrators configure who can access the Windows Admin Center gateway, as well as the identity provider used to authenticate users. Learn more about controlling access to the gateway.

The **Proxy** tab allows users to configure a proxy server to redirect all Windows Admin Center outbound traffic.

From the **Register** tab, administrators can register the gateway with Azure to enable Azure integration features in Windows Admin Center.

Using the **Updates** tab, users can see which version of Windows Admin Center is running and if this version is up to date.

Using the **Shared Connections** tab, administrators can configure a single list of connections to be shared across all users of the Windows Admin Center gateway. Learn more about configuring connections once for all users of a gateway.

For **WebSocket validation**, administrators can now validate their WebSocket connections and customize these settings to various conditions. Learn more about WebSocket validation

---

Feedback

Was this page helpful? Yes No

Provide product feedback

Configure User Access Control and Permissions

Article • 12/11/2023 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

If you haven't already, familiarize yourself with the user access control options in Windows Admin Center.

> ⓘ Note
>
> Group based access in Windows Admin Center is not supported in workgroup environments or across non-trusted domains.

Gateway access role definitions

There are two roles for access to the Windows Admin Center gateway service:

Gateway users can connect to the Windows Admin Center gateway service to manage servers through that gateway, but they can't change access permissions nor the authentication mechanism used to authenticate to the gateway.

Gateway administrators can configure who gets access as well as how users authenticate to the gateway. Only gateway administrators can view and configure the Access settings in Windows Admin Center. Local administrators on the gateway machine are always administrators of the Windows Admin Center gateway service.

There is also an additional role specific to the management of CredSSP:

Windows Admin Center CredSSP Administrators are registered with the Windows Admin Center CredSSP endpoint and have permissions to perform predefined CredSSP operations. This group is especially useful for installations of Windows Admin Center in desktop mode, where only the user account that installed Windows Admin Center is given these permissions by default.

> ⓘ Note
>
> Access to the gateway doesn't imply access to managed servers visible by the gateway. To manage a target server, the connecting user must use credentials (either through their passed-through Windows credential or through credentials provided in the Windows Admin Center session using the **Manage as** action) that

have administrative access to that target server. This is because most Windows Admin Center tools require administrative permissions to use.

Active Directory or local machine groups

By default, Active Directory or local machine groups are used to control gateway access. If you have an Active Directory domain, you can manage gateway user and administrator access from within the Windows Admin Center interface.

On the **Users** tab, you can control who can access Windows Admin Center as a gateway user. By default, and if you don't specify a security group, any user that accesses the gateway URL has access. Once you add one or more security groups to the users list, access is restricted to the members of those groups.

If you don't use an Active Directory domain in your environment, access is controlled by the **Users** and **Administrators** local groups on the Windows Admin Center gateway machine.

Smartcard authentication

You can enforce **smartcard authentication** by specifying an additional *required* group for smartcard-based security groups. Once you have added a smartcard-based security group, a user can only access the Windows Admin Center service if they are a member of any security group AND a smartcard group included in the users list.

On the **Administrators** tab, you can control who can access Windows Admin Center as a gateway administrator. The local administrators group on the computer will always have full administrator access and cannot be removed from the list. By adding security groups, you give members of those groups privileges to change Windows Admin Center gateway settings. The administrators list supports smartcard authentication in the same way as the users list: with the AND condition for a security group and a smartcard group.

Microsoft Entra ID

If your organization uses Microsoft Entra ID, you can choose to add an **additional** layer of security to Windows Admin Center by requiring Microsoft Entra authentication to access the gateway. In order to access Windows Admin Center, the user's **Windows account** must also have access to gateway server (even if Microsoft Entra authentication is used). When you use Microsoft Entra ID, you'll manage Windows Admin Center user

and administrator access permissions from the Azure portal, rather than from within the Windows Admin Center UI.

Accessing Windows Admin Center when Microsoft Entra authentication is enabled

Depending on the browser used, some users accessing Windows Admin Center with Microsoft Entra authentication configured will receive an additional prompt **from the browser** where they need to provide their Windows account credentials for the machine on which Windows Admin Center is installed. After entering that information, the users will get the additional Microsoft Entra authentication prompt, which requires the credentials of an Azure account that has been granted access in the Microsoft Entra application in Azure.

> ⓘ **Note**
>
> Users whose Windows account has **Administrator rights** on the gateway machine will not be prompted for the Microsoft Entra authentication.

Configuring Microsoft Entra authentication for Windows Admin Center Preview

Go to Windows Admin Center **Settings** > **Access** and use the toggle switch to turn on "Use Microsoft Entra ID to add a layer of security to the gateway". If you have not registered the gateway to Azure, you will be guided to do that at this time.

By default, all members of the Microsoft Entra tenant have user access to the Windows Admin Center gateway service. Only local administrators on the gateway machine have administrator access to the Windows Admin Center gateway. Note that the rights of local administrators on the gateway machine cannot be restricted - local admins can do anything regardless of whether Microsoft Entra ID is used for authentication.

If you want to give specific Microsoft Entra users or groups gateway user or gateway administrator access to the Windows Admin Center service, you must do the following:

1. Go to your Windows Admin Center Microsoft Entra application in the Azure portal by using the hyperlink provided in Access Settings. Note this hyperlink is only available when Microsoft Entra authentication is enabled.

    - You can also find your application in the Azure portal by going to **Microsoft Entra ID** > **Enterprise applications** > **All applications** and searching

WindowsAdminCenter (the Microsoft Entra app will be named WindowsAdminCenter-<gateway name>). If you don't get any search results, ensure **Show** is set to **all applications**, **application status** is set to **any** and select Apply, then try your search. Once you've found the application, go to **Users and groups**

2. In the Properties tab, set **User assignment required** to Yes. Once you've done this, only members listed in the **Users and groups** tab will be able to access the Windows Admin Center gateway.
3. In the Users and groups tab, select **Add user**. You must assign a gateway user or gateway administrator role for each user/group added.

Once you turn on Microsoft Entra authentication, the gateway service restarts and you must refresh your browser. You can update user access for the SME Microsoft Entra application in the Azure portal at any time.

Users will be prompted to sign in using their Microsoft Entra identity when they attempt to access the Windows Admin Center gateway URL. Remember that users must also be a member of the local Users on the gateway server to access Windows Admin Center.

Users and administrators can view their currently logged-in account and as well as sign out of this Microsoft Entra account from the **Account** tab of Windows Admin Center Settings.

Configuring Microsoft Entra authentication for Windows Admin Center

To set up Microsoft Entra authentication, you must first register your gateway with Azure (you only need to do this once for your Windows Admin Center gateway). This step creates a Microsoft Entra application from which you can manage gateway user and gateway administrator access.

If you want to give specific Microsoft Entra users or groups gateway user or gateway administrator access to the Windows Admin Center service, you must do the following:

1. Go to your SME Microsoft Entra application in the Azure portal.

    - When you select **Change access control** and then select **Microsoft Entra ID** from the Windows Admin Center Access settings, you can use the hyperlink provided in the UI to access your Microsoft Entra application in the Azure portal. This hyperlink is also available in the Access settings after you select save and have selected Microsoft Entra ID as your access control identity provider.

    - You can also find your application in the Azure portal by going to **Microsoft Entra ID** > **Enterprise applications** > **All applications** and searching **SME** (the Microsoft Entra app will be named SME-<gateway>). If you don't get any search results, ensure **Show** is set to **all applications**, **application status** is set to **any** and select Apply, then try your search. Once you've found the application, go to **Users and groups**
2. In the Properties tab, set **User assignment required** to Yes. Once you've done this, only members listed in the **Users and groups** tab will be able to access the Windows Admin Center gateway.
3. In the Users and groups tab, select **Add user**. You must assign a gateway user or gateway administrator role for each user/group added.

Once you save the Microsoft Entra access control in the **Change access control** pane, the gateway service restarts and you must refresh your browser. You can update user access for the Windows Admin Center Microsoft Entra application in the Azure portal at any time.

Users will be prompted to sign in using their Microsoft Entra identity when they attempt to access the Windows Admin Center gateway URL. Remember that users must also be a member of the local Users on the gateway server to access Windows Admin Center.

Using the **Azure** tab of Windows Admin Center general settings, users and administrators can view their currently logged-in account and as well as sign out of this Microsoft Entra account.

Conditional access and multi-factor authentication

One of the benefits of using Microsoft Entra ID as an additional layer of security to control access to the Windows Admin Center gateway is that you can leverage Microsoft Entra ID's powerful security features like conditional access and multi-factor authentication.

Learn more about configuring conditional access with Microsoft Entra ID.

Configure single sign-on

Single sign-on when deployed as a Service on Windows Server

When you install Windows Admin Center on Windows 10, it's ready to use single sign-on. If you're going to use Windows Admin Center on Windows Server, however, you need to set up some form of Kerberos delegation in your environment before you can

use single sign-on. The delegation configures the gateway computer as trusted to delegate to the target node.

To configure Resource-based constrained delegation in your environment, use the following PowerShell example. This example shows how you would configure a Windows Server [node01.contoso.com] to accept delegation from your Windows Admin Center gateway [wac.contoso.com] in the contoso.com domain.

PowerShell

```
Set-ADComputer -Identity (Get-ADComputer node01) -
PrincipalsAllowedToDelegateToAccount (Get-ADComputer wac)
```

To remove this relationship, run the following cmdlet:

PowerShell

```
Set-ADComputer -Identity (Get-ADComputer node01) -
PrincipalsAllowedToDelegateToAccount $null
```

Role-based access control (RBAC)

Role-based access control enables you to provide users with limited access to the machine instead of making them full local administrators. Read more about role-based access control and the available roles.

Setting up RBAC consists of two steps: enabling support on the target computer(s) and assigning users to the relevant roles.

> **Tip**
>
> Make sure you have local administrator privileges on the machines where you are configuring support for role-based access control.

Apply role-based access control to a single machine

The single machine deployment model is ideal for simple environments with only a few computers to manage. Configuring a machine with support for role-based access control will result in the following changes:

- PowerShell modules with functions required by Windows Admin Center will be installed on your system drive, under `C:\Program Files\WindowsPowerShell\Modules`. All modules will start with **Microsoft.Sme**
- Desired State Configuration will run a one-time configuration to configure a Just Enough Administration endpoint on the machine, named **Microsoft.Sme.PowerShell**. This endpoint defines the three roles used by Windows Admin Center and will run as a temporary local administrator when a user connects to it.
- Three new local groups will be created to control which users are assigned access to which roles:
  - Windows Admin Center Administrators
  - Windows Admin Center Hyper-V Administrators
  - Windows Admin Center Readers

> ⓘ **Note**
>
> Role-based access control is not supported for cluster management (i.e. features that are dependent on RBAC such as CredSSP will fail).

To enable support for role-based access control on a single machine, follow these steps:

1. Open Windows Admin Center and connect to the machine you wish to configure with role-based access control using an account with local administrator privileges on the target machine.
2. On the **Overview** tool, select **Settings** > **Role-based access control**.
3. Select **Apply** at the bottom of the page to enable support for role-based access control on the target computer. The application process involves copying PowerShell scripts and invoking a configuration (using PowerShell Desired State Configuration) on the target machine. It may take up to 10 minutes to complete, and will result in WinRM restarting. This will temporarily disconnect Windows Admin Center, PowerShell, and WMI users.
4. Refresh the page to check the status of role-based access control. When it is ready for use, the status will change to **Applied**.

Once the configuration is applied, you can assign users to the roles:

1. Open the **Local Users and Groups** tool and navigate to the **Groups** tab.
2. Select the **Windows Admin Center Readers** group.
3. In the **Details** pane at the bottom, select **Add User** and enter the name of a user or security group that should have read-only access to the server through Windows

Admin Center. The users and groups can come from the local machine or your Active Directory domain.

4. Repeat steps 2-3 for the **Windows Admin Center Hyper-V Administrators** and **Windows Admin Center Administrators** groups.

You can also fill these groups consistently across your domain by configuring a Group Policy Object with the Restricted Groups Policy Setting.

Apply role-based access control to multiple machines

In a large enterprise deployment, you can use your existing automation tools to push out the role-based access control feature to your computers by downloading the configuration package from the Windows Admin Center gateway. The configuration package is designed to be used with PowerShell Desired State Configuration, but you can adapt it to work with your preferred automation solution.

Download the role-based access control configuration

To download the role-based access control configuration package, you'll need to have access to Windows Admin Center and a PowerShell prompt.

If you're running the Windows Admin Center gateway in service mode on Windows Server, use the following command to download the configuration package. Be sure to update the gateway address with the correct one for your environment.

PowerShell

```
$WindowsAdminCenterGateway = 'https://windowsadmincenter.contoso.com'
Invoke-RestMethod -Uri
"$WindowsAdminCenterGateway/api/nodes/all/features/jea/endpoint/export" -
Method POST -UseDefaultCredentials -OutFile
"~\Desktop\WindowsAdminCenter_RBAC.zip"
```

If you're running the Windows Admin Center gateway on your Windows 10 machine, run the following command instead:

PowerShell

```
$cert = Get-ChildItem Cert:\CurrentUser\My | Where-Object Subject -eq
'CN=Windows Admin Center Client' | Select-Object -First 1
Invoke-RestMethod -Uri
"https://localhost:6516/api/nodes/all/features/jea/endpoint/export" -Method
POST -Certificate $cert -OutFile "~\Desktop\WindowsAdminCenter_RBAC.zip"
```

When you expand the zip archive, you'll see the following folder structure:

- InstallJeaFeatures.ps1
- JustEnoughAdministration (directory)
- Modules (directory)
  - Microsoft.SME.* (directories)

To configure support for role-based access control on a node, you need to perform the following actions:

1. Copy the **JustEnoughAdministration** and **Microsoft.SME.*** modules to the PowerShell module directory on the target machine. Typically, this is located at `C:\Program Files\WindowsPowerShell\Modules`.
2. Update **InstallJeaFeature.ps1** file to match your desired configuration for the RBAC endpoint.
3. Run `InstallJeaFeature.ps1` to compile the DSC resource.
4. Deploy your DSC configuration to all of your machines to apply the configuration.

The following section explains how to do this using PowerShell Remoting.

Deploy on multiple machines

To deploy the configuration you downloaded onto multiple machines, you'll need to update the **InstallJeaFeatures.ps1** script to include the appropriate security groups for your environment, copy the files to each of your computers, and invoke the configuration scripts. You can use your preferred automation tooling to accomplish this, however this article will focus on a pure PowerShell-based approach.

By default, the configuration script will create local security groups on the machine to control access to each of the roles. This is suitable for workgroup and domain joined machines, but if you're deploying in a domain-only environment you may wish to directly associate a domain security group with each role. To update the configuration to use domain security groups, open **InstallJeaFeatures.ps1** and make the following changes:

1. Remove the 3 **Group** resources from the file:
   a. "Group MS-Readers-Group"
   b. "Group MS-Hyper-V-Administrators-Group"
   c. "Group MS-Administrators-Group"
2. Remove the 3 Group resources from the JeaEndpoint **DependsOn** property
   a. "[Group]MS-Readers-Group"
   b. "[Group]MS-Hyper-V-Administrators-Group"
   c. "[Group]MS-Administrators-Group"

3. Change the group names in the JeaEndpoint **RoleDefinitions** property to your desired security groups. For example, if you have a security group *CONTOSO\MyTrustedAdmins* that should be assigned access to the Windows Admin Center Administrators role, change `'$env:COMPUTERNAME\Windows Admin Center Administrators'` to `'CONTOSO\MyTrustedAdmins'`. The three strings you need to update are:
   a. '$env:COMPUTERNAME\Windows Admin Center Administrators'
   b. '$env:COMPUTERNAME\Windows Admin Center Hyper-V Administrators'
   c. '$env:COMPUTERNAME\Windows Admin Center Readers'

> **Note**
>
> Be sure to use unique security groups for each role. Configuration will fail if the same security group is assigned to multiple roles.

Next, at the end of the **InstallJeaFeatures.ps1** file, add the following lines of PowerShell to the bottom of the script:

PowerShell

```
Copy-Item "$PSScriptRoot\JustEnoughAdministration"
"$env:ProgramFiles\WindowsPowerShell\Modules" -Recurse -Force
$ConfigData = @{
    AllNodes = @()
    ModuleBasePath = @{
        Source = "$PSScriptRoot\Modules"
        Destination = "$env:ProgramFiles\WindowsPowerShell\Modules"
    }
}
InstallJeaFeature -ConfigurationData $ConfigData | Out-Null
Start-DscConfiguration -Path "$PSScriptRoot\InstallJeaFeature" -JobName
"Installing JEA for Windows Admin Center" -Force
```

Finally, you can copy the folder containing the modules, DSC resource and configuration to each target node and run the **InstallJeaFeature.ps1** script. To do this remotely from your admin workstation, you can run the following commands:

PowerShell

```
$ComputersToConfigure = 'MyServer01', 'MyServer02'

$ComputersToConfigure | ForEach-Object {
    $session = New-PSSession -ComputerName $_ -ErrorAction Stop
    Copy-Item -Path
"~\Desktop\WindowsAdminCenter_RBAC\JustEnoughAdministration\" -Destination
"$env:ProgramFiles\WindowsPowerShell\Modules\" -ToSession $session -Recurse
```

```
-Force
    Copy-Item -Path "~\Desktop\WindowsAdminCenter_RBAC" -Destination
"$env:TEMP\WindowsAdminCenter_RBAC" -ToSession $session -Recurse -Force
    Invoke-Command -Session $session -ScriptBlock { Import-Module
JustEnoughAdministration; &
"$env:TEMP\WindowsAdminCenter_RBAC\InstallJeaFeature.ps1" } -AsJob
    Disconnect-PSSession $session
}
```

Feedback

Was this page helpful? Yes No

Provide product feedback

Install and manage extensions

Article • 11/13/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Windows Admin Center is built as an extensible platform where each connection type and tool is an extension that you can install, uninstall, and update individually. You can search for new extensions published by Microsoft and other developers, and install and update them individually without having to update the entire Windows Admin Center installation. You can also configure a separate NuGet feed or file share and distribute extensions to use internally within your organization.

Installing an extension

Windows Admin Center shows extensions available from the specified NuGet feed. By default, Windows Admin Center points to the Microsoft official NuGet feed which hosts extensions published by Microsoft and other developers.

1. Select the **Settings** button in the top-right > In the left pane, select **Extensions**.
2. The **Available Extensions** tab lists the extensions on the feed that are available for installation.
3. Select on an extension to view the extension description, version, publisher and other information in the **Details** pane.
4. Select **Install** to install an extension. If the gateway must run in elevated mode to make this change, you're presented with a User Account Control (UAC) elevation prompt. After installation is complete, your browser automatically refreshes and Windows Admin Center reloads with the new extension installed. If the extension you are trying to install is an update to a previously installed extension, you can select the **Update to latest** button to install the update. You can also go to the **Installed Extensions** tab to view installed extensions and see if an update is available in the **Status** column.

Installing extensions from a different feed

Windows Admin Center supports multiple feeds and you can view and manage packages from more than one feed at a time. Any NuGet feed that supports the NuGet V2 APIs or a file share can be added to Windows Admin Center for installing extensions from.

1. Select the **Settings** button in the top-right > In the left pane, select **Extensions**.
2. On the right pane, select the **Feeds** tab.

3. Select the **Add** button to add another feed. For a NuGet feed, enter the NuGet V2 feed URL. The NuGet feed provider or administrator should be able to provide the URL information. For a file share, enter the full path of the file share in which the extension package files (.nupkg) are stored. If adding a file share as a feed, it must meet the following criteria:

    - Access to the file must be granted to NTAuthority\Network Service
    - The file path can't include the C:\Users folder
    - The packages on the file share must be signed, unless your operation mode is set to "Development"

4. Select **Add**. If the gateway must run in elevated mode to make this change, you're presented with a UAC elevation prompt. This prompt will only be presented if you are running Windows Admin Center in desktop mode.

The **Available Extensions** list shows extensions from all registered feeds. You can check which feed each extension is from using the **Package Feed** column.

Uninstalling an extension

You can uninstall any extensions you have previously installed, or even uninstall any tools that were preinstalled as part of the Windows Admin Center installation.

1. Select the **Settings** button in the top-right > In the left pane, select **Extensions**.
2. Select the **Installed Extensions** tab to view all installed extensions.
3. Choose an extension to uninstall, then select **Uninstall**.

After uninstall is complete, your browser automatically refreshes and Windows Admin Center reloads with the extension removed. If you uninstalled a tool that was preinstalled as part of Windows Admin Center, the tool is available for reinstallation in the **Available Extensions** tab.

Installing extensions on a computer without internet connectivity

If Windows Admin Center is installed on a computer that isn't connected to the internet or is behind a proxy, it may not be able to access and install the extensions from the Windows Admin Center feed. You can download extension packages manually or with a PowerShell script, and configure Windows Admin Center to retrieve packages from a file share or local drive.

Manually downloading extension packages

1. On another computer that has internet connectivity, open a web browser and navigate to the following URL: https://dev.azure.com/WindowsAdminCenter/Windows%20Admin%20Center%20Feed/_packaging?_a=feed&feed=WAC
    - You may need to create a Microsoft account and sign-in to view the extension packages.
2. Select on the name of the package you want to install to view the package details page.
3. Select on the **Download** link in the top navigation bar of the package details page and download the .nupkg file for the extension.
4. Repeat steps 2 and 3 for all the packages you want to download.
5. Copy the package files to a file share that can be accessed from the computer Windows Admin Center is installed on, or to the local disk of the computer on which Windows Admin Center is installed.
6. Follow the instructions to install extensions from a different feed.

Downloading packages with a PowerShell script

There are many scripts available on the Internet for downloading NuGet packages from a NuGet feed. We'll use the script provided by Jon Galloway, Senior Program Manager at Microsoft.

1. As described in the blog post, install the script as a NuGet package, or copy and paste the script into the PowerShell ISE.
2. Edit the first line of the script to your NuGet feed's v2 URL. If you are downloading packages from the Windows Admin Center official feed, use the following URL:

```PowerShell
$feedUrlBase = "https://aka.ms/sme-extension-feed"
```

3. Run the script and it downloads all the NuGet packages from the feed to the following local folder: %USERPROFILE%\Documents\NuGetLocal
4. Follow the instructions to install extensions from a different feed.

Manage extensions with PowerShell

Windows Admin Center Preview includes a PowerShell module to manage your gateway extensions.

PowerShell

```
# Add the module to the current session
Import-Module "$env:ProgramFiles\windows admin
center\PowerShell\Modules\ExtensionTools"
# Available cmdlets: Get-Feed, Add-Feed, Remove-Feed, Get-Extension,
Install-Extension, Uninstall-Extension, Update-Extension

# List feeds
Get-Feed "https://wac.contoso.com"

# Add a new extension feed
Add-Feed -GatewayEndpoint "https://wac.contoso.com" -Feed "\\WAC\our-
private-extensions"

# Remove an extension feed
Remove-Feed -GatewayEndpoint "https://wac.contoso.com" -Feed "\\WAC\our-
private-extensions"

# List all extensions
Get-Extension "https://wac.contoso.com"

# Install an extension (locate the latest version from all feeds and install
it)
Install-Extension -GatewayEndpoint "https://wac.contoso.com"
"msft.sme.containers"

# Install an extension (latest version from a specific feed, if the feed is
not present, it will be added)
Install-Extension -GatewayEndpoint "https://wac.contoso.com"
"msft.sme.containers" -Feed "https://aka.ms/sme-extension-feed"

# Install an extension (install a specific version)
Install-Extension "https://wac.contoso.com" "msft.sme.certificate-manager"
"0.133.0"

# Uninstall-Extension
Uninstall-Extension "https://wac.contoso.com" "msft.sme.containers"

# Update-Extension
Update-Extension "https://wac.contoso.com" "msft.sme.containers"
```

> ⓘ **Note**

You must be gateway administrator to modify Windows Admin Center extensions with PowerShell.

Learn more about building an extension with the Windows Admin Center SDK.

Feedback

Was this page helpful? Yes No

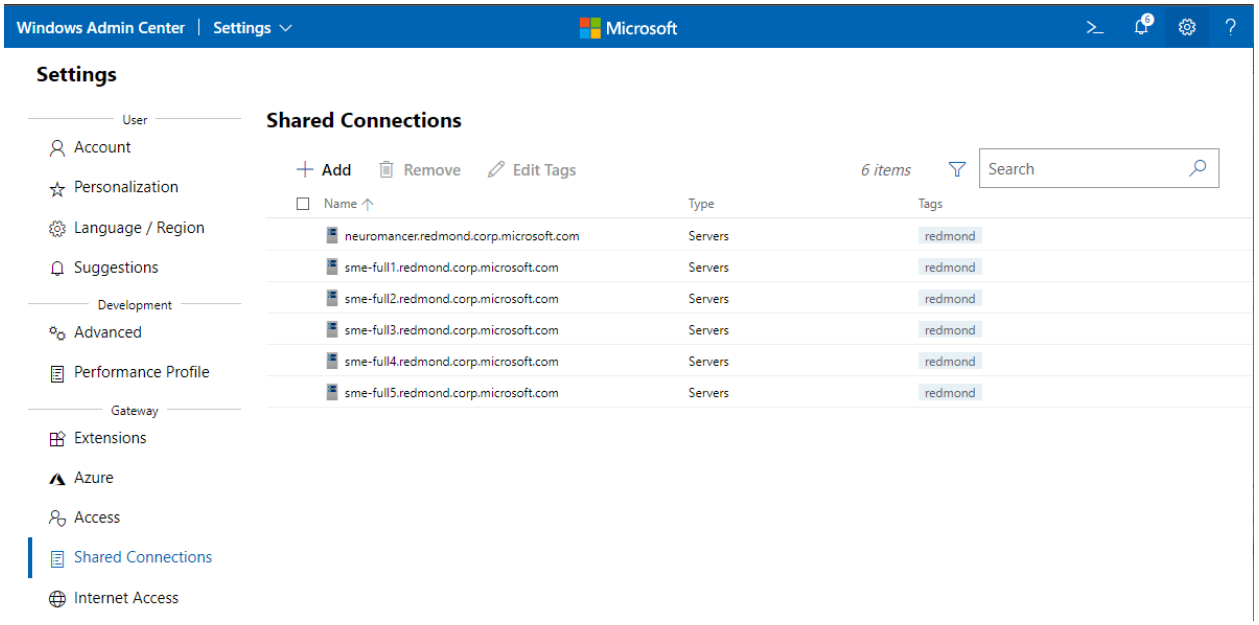
Provide product feedback

Configure shared connections for all users of the Windows Admin Center gateway

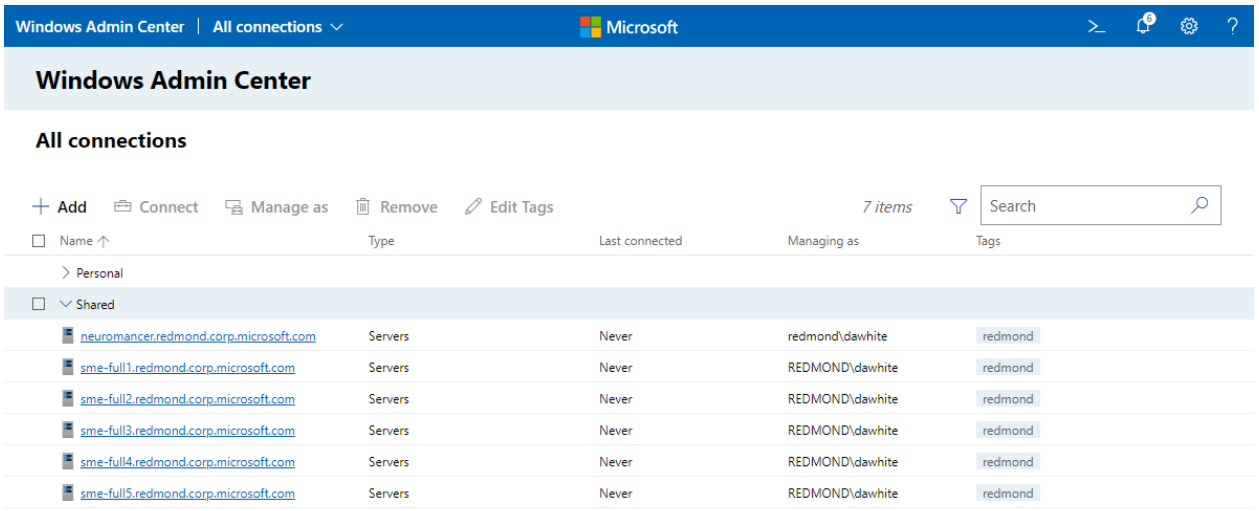
Article • 11/01/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

With the ability to configure shared connections, gateway administrators can configure the connections list once for all users of a given Windows Admin Center gateway. This feature is only available on Windows Admin Center service mode.

From the **Shared Connections** tab of Windows Admin Center gateway Settings, gateway administrators can add servers, clusters, and PC connections as you would from the all connections page, including the ability to tag connections. Any connections and tags added in the Shared Connections list will appear for all users of this Windows Admin Center gateway, from their all connections page.



When any Windows Admin Center user accesses the "All Connections" page after Shared Connections have been configured, they will see their connections grouped into two sections: Personal and Shared connections. The Personal group is a specific user's connection list and persists across that user's browser sessions. The Shared connections group is the same across all users, and cannot be modified from the All Connections page.



Feedback

Was this page helpful? Yes No

Provide product feedback

Use PowerShell to manage Windows Admin Center settings

Article • 10/17/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

If you have a large organization with multiple Windows Admin Center servers, you can use PowerShell to configure the list of connections and extensions on multiple servers at a time.

Use PowerShell to import or export your connections (with tags)

PowerShell

```
# Load the module
Import-Module "$env:ProgramFiles\windows admin
center\PowerShell\Modules\ConnectionTools"
# Available cmdlets: Export-Connection, Import-Connection

# Export connections (including tags) to a .csv file
Export-Connection "https://wac.contoso.com" -fileName "WAC-connections.csv"
# Import connections (including tags) from a .csv file
Import-Connection "https://wac.contoso.com" -fileName "WAC-connections.csv"
# Import connections (including tags) from .csv files, and remove any
connections that are not explicitly in the imported file by using the -prune
switch parameter
Import-Connection "https://wac.contoso.com" -fileName "WAC-connections.csv"
-prune
```

CSV file format for importing connections

The format of the CSV file starts with the following four headings, followed by each connection on a new line.

- `name`: The FQDN of the connection.
- `type`: The connection type. For the default connections included with Windows Admin Center, use one of the following strings:

Expand table

Connection type	Connection string
Windows Server	`msft.sme.connection-type.server`
Failover cluster	`msft.sme.connection-type.cluster`

- `tags`: Pipe-separated tags.
- `groupId`: The column for shared connections. Use the value `global` in this column to make a shared connection.

> ⓘ Note
>
> Modifying the shared connections is limited to gateway administrators. Any user can use PowerShell to modify their personal connection list.

Example CSV file for importing connections

```
"name","type","tags","groupId"
"myServer.contoso.com","msft.sme.connection-type.server","hyperv"
"myDesktop.contoso.com","msft.sme.connection-type.windows-server","hyperv"
"teamcluster.contoso.com","msft.sme.connection-
type.cluster","legacyCluster|WS2016","global"
"myHCIcluster.contoso.com,"msft.sme.connection-
type.cluster","myHCIcluster|hyperv|JIT|WS2019"
"teamclusterNode.contoso.com","msft.sme.connection-
type.server","legacyCluster|WS2016","global"
"myHCIclusterNode.contoso.com","msft.sme.connection-
type.server","myHCIcluster|hyperv|JIT|WS2019"
```

> ⓘ Note
>
> The CSV file is case-sensitive.

Import RDCMan connections

Use the following script to export saved connections in RDCMan to a file. You can then import the file into Windows Admin Center and maintain your RDCMan grouping hierarchy by using tags.

1. Copy and paste the following code into your PowerShell session:

PowerShell

```powershell
#Helper function for RdgToWacCsv
function AddServers {
 param (
 [Parameter(Mandatory = $true)]
 [Xml.XmlLinkedNode]
 $node,
 [Parameter()]
 [String[]]
 $tags,
 [Parameter(Mandatory = $true)]
 [String]
 $csvPath
 )
 if ($node.LocalName -eq 'server') {
     $serverName = $node.properties.name
     $tagString = $tags -join "|"
     Add-Content -Path $csvPath -Value ('"'+ $serverName +
'","msft.sme.connection-type.server","'+ $tagString +'"')
 }
 elseif ($node.LocalName -eq 'group' -or $node.LocalName -eq 'file') {
     $groupName = $node.properties.name
     $tags+=$groupName
     $currNode = $node.properties.NextSibling
     while ($currNode) {
         AddServers -node $currNode -tags $tags -csvPath $csvPath
         $currNode = $currNode.NextSibling
     }
 }
 else {
     # Node type isn't relevant to tagging or adding connections in
Windows Admin Center
 }
 return
}

<#
.SYNOPSIS
Convert an .rdg file from Remote Desktop Connection Manager into a .csv
that can be imported into Windows Admin Center, maintaining groups via
server tags. This will not modify the existing .rdg file and will
create a new .csv file

 .DESCRIPTION
 This converts an .rdg file into a .csv that can be imported into
Windows Admin Center.

 .PARAMETER RDGfilepath
 The path of the .rdg file to be converted. This file will not be
modified, only read.

 .PARAMETER CSVdirectory
 Optional. The directory you want to export the new .csv file. If it's
```

```
not provided, the new file is created in the same directory as the .rdg
file.

 .EXAMPLE
 C:\PS> RdgToWacCsv -RDGfilepath "rdcmangroup.rdg"
 #>
function RdgToWacCsv {
 param(
     [Parameter(Mandatory = $true)]
     [String]
     $RDGfilepath,
     [Parameter(Mandatory = $false)]
     [String]
     $CSVdirectory
 )
 [xml]$RDGfile = Get-Content -Path $RDGfilepath
 $node = $RDGfile.RDCMan.file
 if (!$CSVdirectory){
     $csvPath = [System.IO.Path]::GetDirectoryName($RDGfilepath) +
[System.IO.Path]::GetFileNameWithoutExtension($RDGfilepath) +
"_WAC.csv"
 } else {
     $csvPath = $CSVdirectory +
[System.IO.Path]::GetFileNameWithoutExtension($RDGfilepath) +
"_WAC.csv"
 }
 New-item -Path $csvPath
 Add-Content -Path $csvPath -Value '"name","type","tags"'
 AddServers -node $node -csvPath $csvPath
 Write-Host "Converted $RDGfilepath `nOutput: $csvPath"
}
```

2. To create a CSV file, run the following command:

PowerShell

```
RdgToWacCsv -RDGfilepath "path\to\myRDCManfile.rdg"
```

3. Import the resulting CSV file into Windows Admin Center. Tags in the connection list represent your RDCMan grouping hierarchy.

Manage Windows Admin Center extensions with PowerShell

PowerShell

```
# Add the module to the current session
Import-Module "$env:ProgramFiles\windows admin
center\PowerShell\Modules\ExtensionTools"
```

```
# Available cmdlets: Get-Feed, Add-Feed, Remove-Feed, Get-Extension,
Install-Extension, Uninstall-Extension, Update-Extension

# List feeds
Get-Feed "https://wac.contoso.com"

# Add a new extension feed
Add-Feed -GatewayEndpoint "https://wac.contoso.com" -Feed "\\WAC\our-
private-extensions"

# Remove an extension feed
Remove-Feed -GatewayEndpoint "https://wac.contoso.com" -Feed "\\WAC\our-
private-extensions"

# List all extensions
Get-Extension "https://wac.contoso.com"

# Install an extension (locate the latest version from all feeds and install
it)
Install-Extension -GatewayEndpoint "https://wac.contoso.com"
"msft.sme.containers"

# Install an extension (latest version from a specific feed, if the feed is
not present, it will be added)
Install-Extension -GatewayEndpoint "https://wac.contoso.com"
"msft.sme.containers" -Feed "https://aka.ms/sme-extension-feed"

# Install an extension (install a specific version)
Install-Extension "https://wac.contoso.com" "msft.sme.certificate-manager"
"0.133.0"

# Uninstall-Extension
Uninstall-Extension "https://wac.contoso.com" "msft.sme.containers"

# Update-Extension
Update-Extension "https://wac.contoso.com" "msft.sme.containers"
```

> ⓘ Note
>
> You must be gateway administrator to modify Windows Admin Center extensions with PowerShell.

Additional References

- Deploy a highly available Windows Admin Center gateway on a cluster
- Deploy a Windows Admin Center gateway in Azure using Cloud Shell

Feedback

Was this page helpful? Yes No

Provide product feedback

Customize WebSocket validation for Windows Admin Center gateway

Article • 12/23/2021 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

To protect WebSocket access, WebSocket connection will now validate **origin** state from the browser so not any external application could get access the WebSocket API defined on the gateway.

Customization of validation

Validation can be adjusted to customize various conditions.

User can configure WebSocket override setting at a Windows Admin Center registry value, `HKLM\Software\Microsoft\ServerManagementGateway\WebSocketValidationOverride`, to specify exceptional **origin host** name and **origin port**. This include wildcard name such as "`*.mydomain.mycompany.net`" or just "`*`" to accept all. Wildcard must be specified single form like "`*.`" and cannot be combined with complex string match condition like "`something*something`".

Example of accepted formats are as follows:

- Always allows origin host defined on current TLS certificate. (subject name, alternate DNS names)
- Always allows origin port configured to Windows Admin Center
- "`*`" - accept any origin host and origin port
- "`*:9876`" - accept any origin host and origin port 9876
- "`:9876`" - accept origin port 9876
- "`*.my.domain.com`" - accept origin host <any.any.any...>.my.domain.com
- "`*.my.domain.com:9876`" - accept origin host <any.any.any...>.my.domain.com and origin port 9876

Prevention logic

Gateway adds a session cookie (WAC-SESSION) for user browser. It associates the browser session and username always. It prevents different user attempting using the same browser session.

- When UI starts a WebSocket connection the browser sends the session cookie back to Gateway.

- Gateway validates authenticated username matched with the session cookie always.

Gateway looks for **origin header**, which is endpoint URL that original Windows Admin Center site was loaded.

- Gateway validated **origin host** and **origin port** against current SSL certificate settings which includes list of DNS host names. This tells the UI code is loaded from expected DNS name sites and port.

RDP enhancement

On RDP TCP connection, Gateway only allows to use port 3389 (RDP) and port 2179 (VM connection), so TCP forwarding feature cannot be used for any other purpose.

Possible side effect

If user uses Windows Admin Center by IP address or something not described on the SSL certificate, user cannot access WebSocket because it's not trustable. If it needs to support, modify `HKLM\Software\Microsoft\ServerManagementGateway\WebSocketValidationOverride` registry value to set the IP address or just specify "`*`" to ignore validation.

Feedback

Was this page helpful? Yes No

Provide product feedback

Get started with Windows Admin Center

Article • 02/10/2025 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

This article describes how to get started with Windows Admin Center after you download and install it on a Windows PC. To learn more about Windows Admin Center, see Windows Admin Center overview.

Open Windows Admin Center on a Windows PC

Windows Admin Center enables you to manage servers, clusters, Windows PCs, and Azure virtual machines (VMs) directly from your Windows 10 computer by using a web browser.

> **Important**
>
> You must be a member of the local administrators group to use Windows Admin Center on Windows 10.

Follow these steps to open Windows Admin Center on a Windows PC:

1. On the **Start** menu, select **Windows Admin Center**. Or enter **Windows Admin Center** on the search bar and then select it from the search results.

   Windows Admin Center opens in your default browser with the URL `https://localhost:6516/`. Alternatively, you can start it from your desired browser by entering `https://localhost:6516/`.

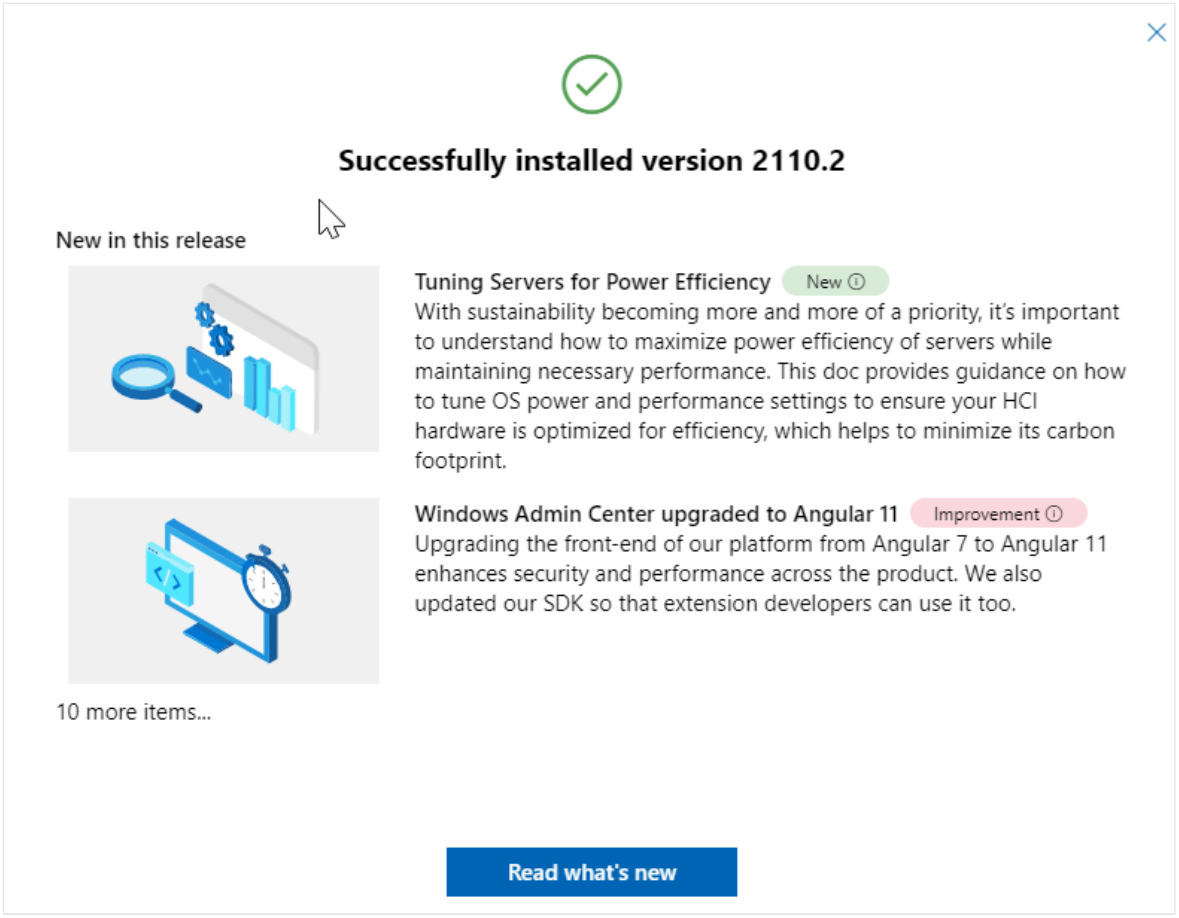
2. A prompt for selecting a certificate for authentication appears.

   > **Note**
   >
   > In the modernized gateway build of Windows Admin Center, you're asked to sign in to the Windows Admin Center gateway instead of selecting a certificate. This step does not apply.

   Select the certificate labeled **Windows Admin Center Client**, and then select **OK**. You can also select the **Certificate information** link to see more details about the certificate.

> **Important**
>
> Be sure to select **Windows Admin Center Client** and not any other certificate. If you select any other certificate, the following error message appears:
>
> "You are not authorized to view this page. If you recently updated Windows Admin Center, you may need to restart your browser, and then refresh the page."
>
> If you continue to get the same error even after restarting your browser and refreshing the page, try clearing the browser cache or switching to another browser. If none of these troubleshooting steps resolve the issue, you might need to uninstall and reinstall Windows Admin Center, and then restart it.

3. (First time only:) A pop-up window confirms that your Windows Admin Center version is successfully installed. It also provides information on what's new in this release. Close the window to proceed.



The **All connections** page shows your Windows 10 computer name in the list of connections. You're now ready to add connections to Windows Admin Center.

Add connections to Windows Admin Center

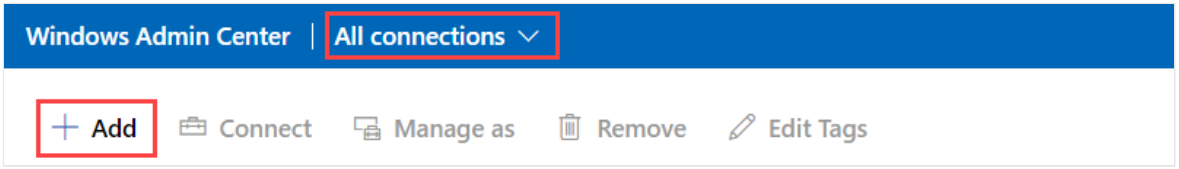
You can add connections as managed nodes to Windows Admin Center from the **All connections** page. **All connections** is the default page that appears when you open Windows Admin Center. On this page, you can either add existing resources as connections or create new resources. The resources that you add appear in the connection list.

The types of resources that you can add from the **All connections** page are servers, Windows PCs, clusters, and Azure VMs. The resources that you can create are clusters and Azure VMs.

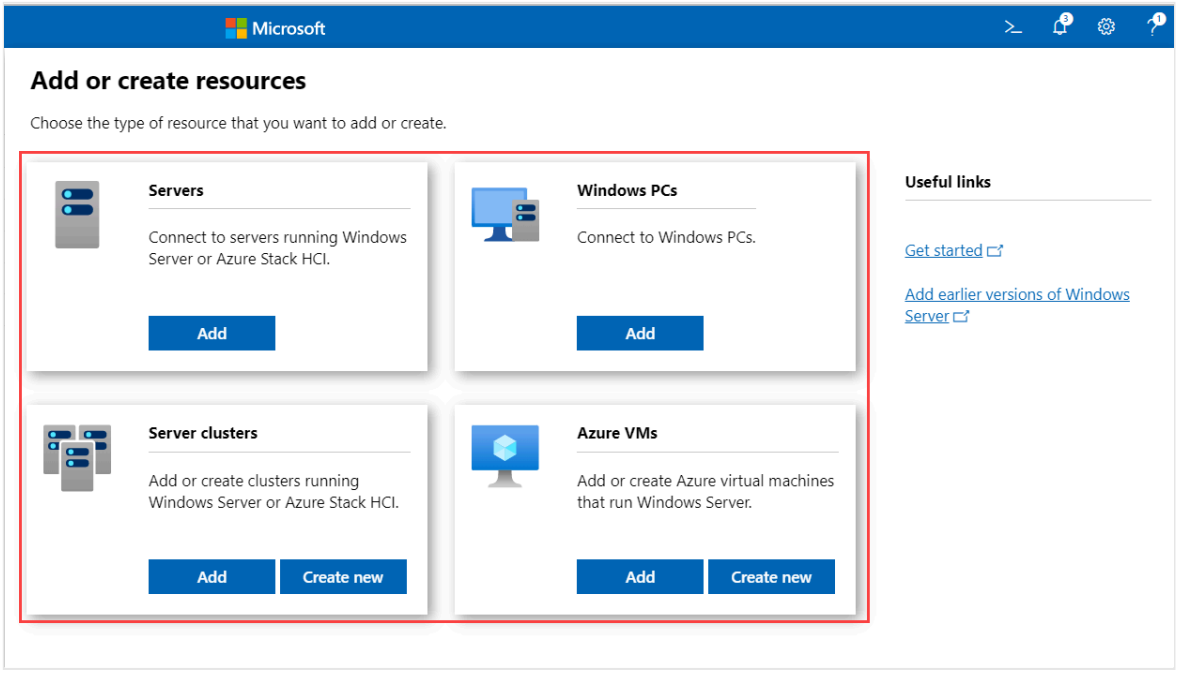
> **Note**
>
> To add or create an Azure VM, you must first register Windows Admin Center with Azure.

To add connections to Windows Admin Center:

1. Under **All connections**, select **+ Add**.



2. The types of resources that you can add are displayed. Select **Add** for the resource type that you want to add.



3. Windows Admin Center supports various methods to add resources, depending on the resource type:

- Add one resource at a time.
- Add multiple resources by bulk importing.
- Add resources by searching Active Directory.

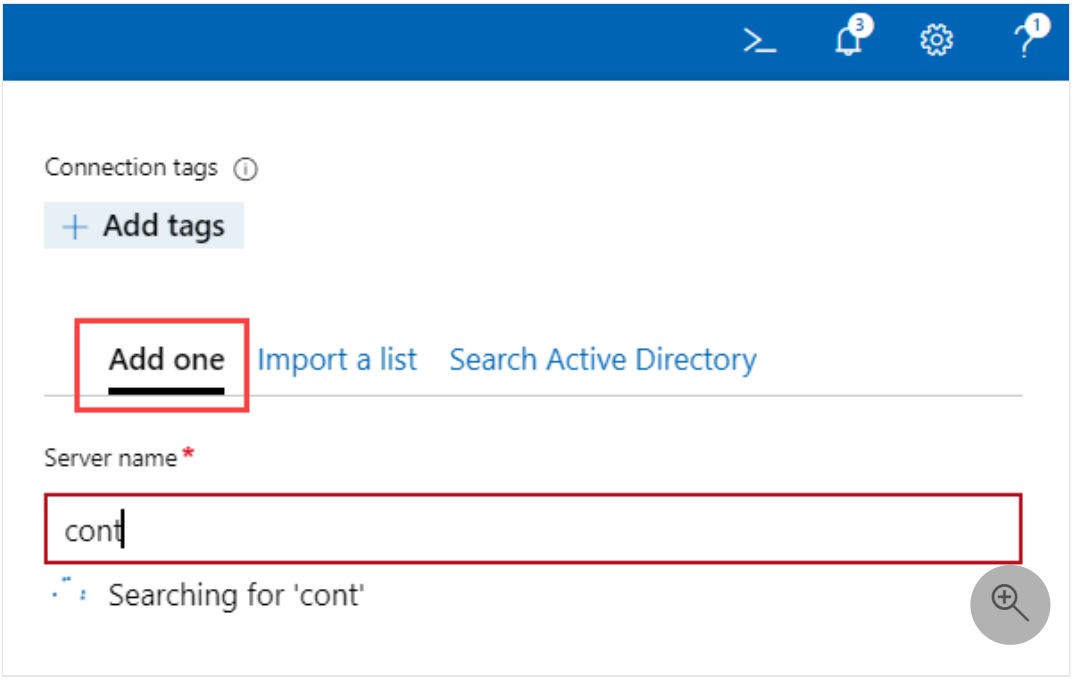
Select the tab based on how you want to add resources. The label for each tab can differ based on the resource type that you're adding.

Add one

Adding one resource at a time is the default method. The label for this tab appears as **Add cluster** when you're adding a cluster.

a. Select the **Add one** or **Add cluster** tab.

b. In the **Server name** box, enter the name of the resource.



As you begin entering text, Windows Admin Center starts searching for a resource based on your text string. If a match is found, you can add the name exactly as you entered it or use the default resource name. If no match is found, you can still add this resource to appear in your list of connections.

4. (Optional:) Select **Add tags** to add connection tags. You can use tags to identify and filter related resources in your connection list. For more information, see Use tags to organize your connections later in this article.

5. When you finish adding resources, select **Add**.

The selected resources appear in the connection list on the **All connections** page.

Authenticate with the managed nodes

After you add connections as managed nodes, you must authenticate with them to connect.

Windows Admin Center supports several mechanisms for authenticating with a managed node. Single sign-on is the default.

> **Note**
>
> To perform remote management, Windows Admin Center impersonates the provided user's security context and uses that security context to access the machine. The provided user is listed in the **Managing As** column on the **All Connections** page.

Authenticate by single sign-on

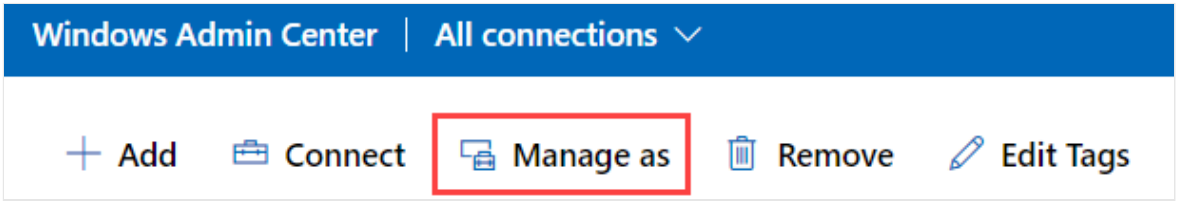
You can use your current Windows credentials to authenticate with the managed node. This authentication is the default, and Windows Admin Center attempts the single sign-on when you add a resource.

Authenticate by single sign-on when Windows Admin Center is deployed as a service on Windows Server

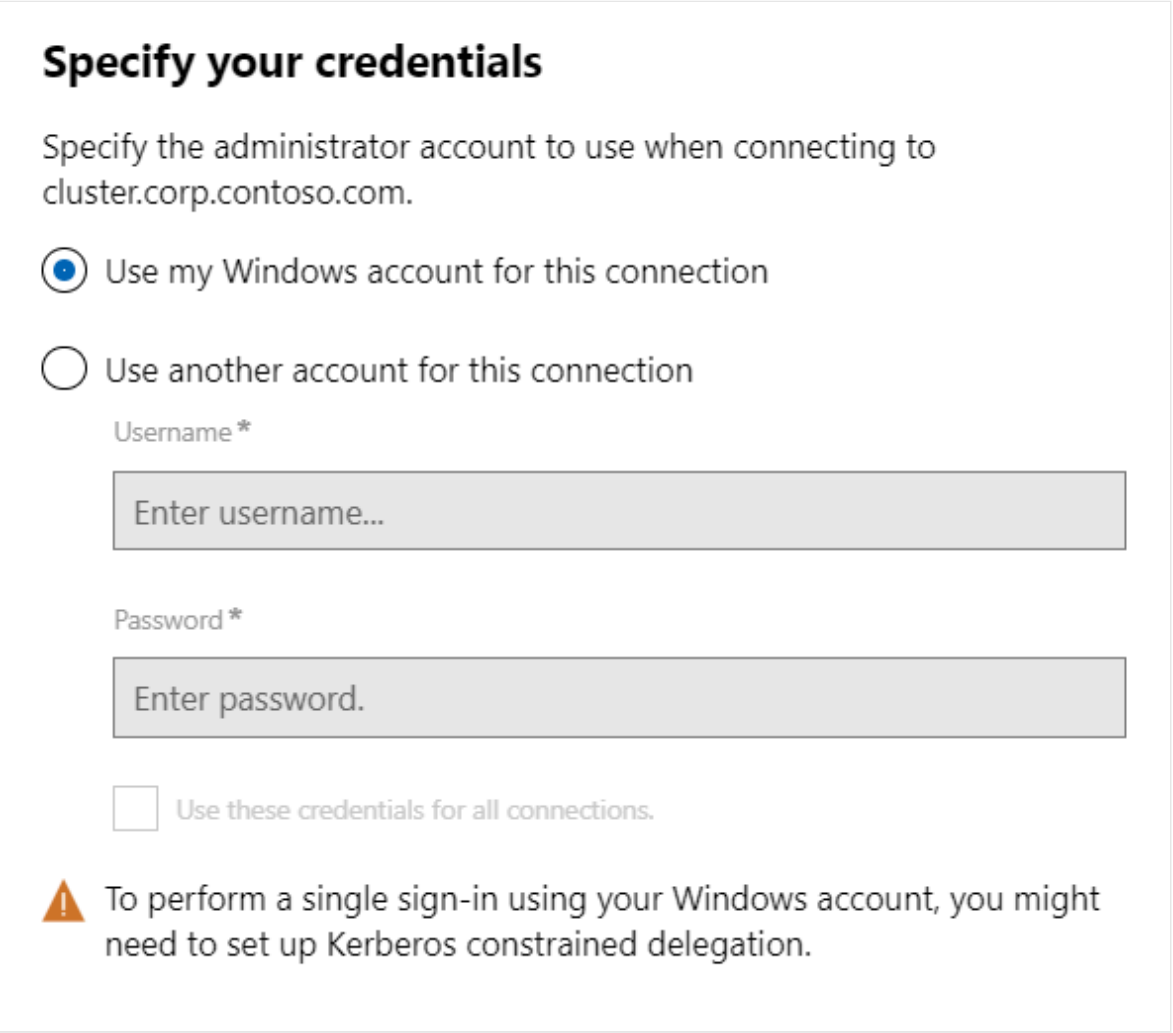
If you installed Windows Admin Center on Windows Server, additional configuration is required for single sign-on. For more information, see Configure single sign-on.

Authenticate by specifying credentials

1. In **All connections**, select a resource from the list.
2. Select **Manage as** to specify the credentials that you'll use to authenticate to the managed node.



3. If Windows Admin Center is running in service mode on Windows Server, but you don't have Kerberos delegation configured, you must reenter your Windows credentials.



You can apply the credentials to all connections, which will cache them for that specific browser session. If you reload your browser, you must reenter your **Manage as** credentials.

Authenticate by Local Administrator Password Solution

If your environment uses Local Administrator Password Solution (LAPS), and you have Windows Admin Center installed on your Windows 10 PC, you can use LAPS credentials to authenticate with the managed node. If you use this scenario, you can provide feedback.

Use tags to organize your connections

You can use tags to identify and filter related resources in your connection list. The ability to view a subset of your resources in the connection list is especially useful if you

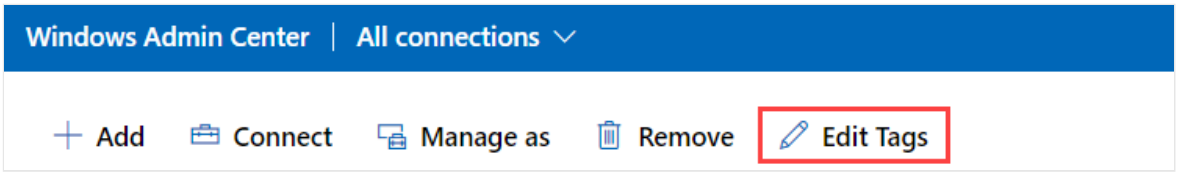
have many connections.

Edit tags

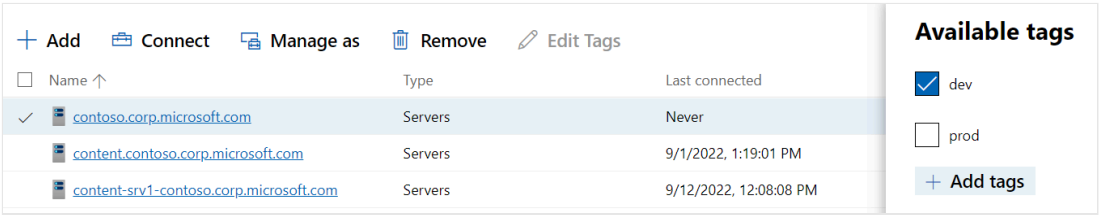
You can add tags to a connection while you're adding a new connection. Or you can add or edit them later for one or more connections on the **All connections** page.

To add or edit a tag:

1. In the list of connections, select one or more connections.
2. Select **Edit Tags**.



3. On the **Available tags** pane, you can take any of these actions:
   - To add a new tag to your selected connections, select **Add tags** and enter a desired tag name.
   - To tag the selected connections with an existing tag name, select the checkbox next to the tag name that you want to apply.
   - To remove a tag from all selected connections, clear the checkbox next to the tag that you want to remove.
   - If a tag is applied to a subset of the selected connections, the checkbox appears in an intermediate state. You can select the checkbox and apply the tag to all selected connections, or clear the checkbox to remove the tag from all selected connections.

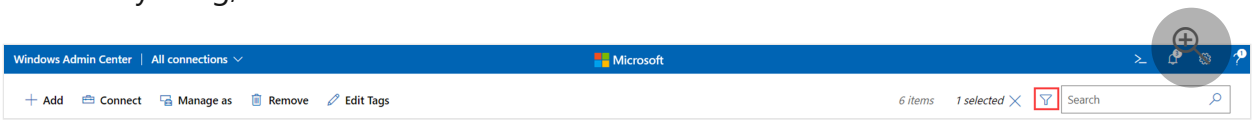


4. Select **Save** to save your changes.

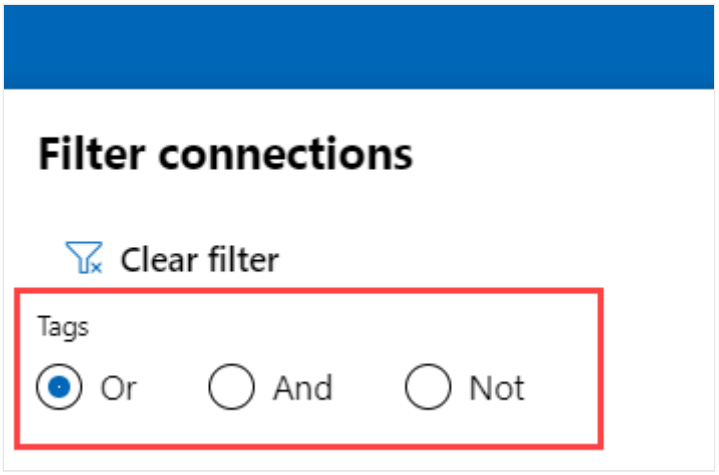
Filter connections by tag

After you add tags to one or more connections, you can view the tags in the connection list and filter the connection list by tags.

To filter by a tag, select the filter icon next to the search box.



You can select **Or**, **And**, or **Not** to modify the filter behavior of the selected tags.



Use PowerShell to import or export your connections with tags

PowerShell

```powershell
# Load the module
Import-Module "$env:ProgramFiles\windows admin
center\PowerShell\Modules\ConnectionTools"
# Available cmdlets: Export-Connection, Import-Connection

# Export connections (including tags) to a .csv file
Export-Connection "https://wac.contoso.com" -fileName "WAC-connections.csv"
# Import connections (including tags) from a .csv file
Import-Connection "https://wac.contoso.com" -fileName "WAC-connections.csv"
# Import connections (including tags) from .csv files, and remove any
connections that are not explicitly in the imported file by using the -prune
switch parameter
Import-Connection "https://wac.contoso.com" -fileName "WAC-connections.csv"
-prune
```

CSV file format for importing connections

The format of the CSV file starts with the following four headings, followed by each connection on a new line.

- `name`: The FQDN of the connection.

- `type`: The connection type. For the default connections included with Windows Admin Center, use one of the following strings:

  Expand table

Connection type	Connection string
Windows Server	`msft.sme.connection-type.server`
Failover cluster	`msft.sme.connection-type.cluster`

- `tags`: Pipe-separated tags.
- `groupId`: The column for shared connections. Use the value `global` in this column to make a shared connection.

> **Note**
>
> Modifying the shared connections is limited to gateway administrators. Any user can use PowerShell to modify their personal connection list.

Example CSV file for importing connections

```
"name","type","tags","groupId"
"myServer.contoso.com","msft.sme.connection-type.server","hyperv"
"myDesktop.contoso.com","msft.sme.connection-type.windows-server","hyperv"
"teamcluster.contoso.com","msft.sme.connection-
type.cluster","legacyCluster|WS2016","global"
"myHCIcluster.contoso.com,"msft.sme.connection-
type.cluster","myHCIcluster|hyperv|JIT|WS2019"
"teamclusterNode.contoso.com","msft.sme.connection-
type.server","legacyCluster|WS2016","global"
"myHCIclusterNode.contoso.com","msft.sme.connection-
type.server","myHCIcluster|hyperv|JIT|WS2019"
```

> **Note**
>
> The CSV file is case-sensitive.

Import RDCMan connections

Use the following script to export saved connections in RDCMan to a file. You can then import the file into Windows Admin Center and maintain your RDCMan grouping hierarchy by using tags.

1. Copy and paste the following code into your PowerShell session:

PowerShell

```
#Helper function for RdgToWacCsv
function AddServers {
 param (
 [Parameter(Mandatory = $true)]
 [Xml.XmlLinkedNode]
 $node,
 [Parameter()]
 [String[]]
 $tags,
 [Parameter(Mandatory = $true)]
 [String]
 $csvPath
 )
 if ($node.LocalName -eq 'server') {
     $serverName = $node.properties.name
     $tagString = $tags -join "|"
     Add-Content -Path $csvPath -Value ('"'+ $serverName +
'","msft.sme.connection-type.server","'+ $tagString +'"')
 }
 elseif ($node.LocalName -eq 'group' -or $node.LocalName -eq 'file') {
     $groupName = $node.properties.name
     $tags+=$groupName
     $currNode = $node.properties.NextSibling
     while ($currNode) {
         AddServers -node $currNode -tags $tags -csvPath $csvPath
         $currNode = $currNode.NextSibling
     }
 }
 else {
     # Node type isn't relevant to tagging or adding connections in
Windows Admin Center
 }
 return
}

<#
.SYNOPSIS
Convert an .rdg file from Remote Desktop Connection Manager into a .csv
that can be imported into Windows Admin Center, maintaining groups via
server tags. This will not modify the existing .rdg file and will
create a new .csv file

 .DESCRIPTION
 This converts an .rdg file into a .csv that can be imported into
Windows Admin Center.
```

```
 .PARAMETER RDGfilepath
 The path of the .rdg file to be converted. This file will not be
modified, only read.

 .PARAMETER CSVdirectory
 Optional. The directory you want to export the new .csv file. If it's
not provided, the new file is created in the same directory as the .rdg
file.

 .EXAMPLE
 C:\PS> RdgToWacCsv -RDGfilepath "rdcmangroup.rdg"
 #>
function RdgToWacCsv {
 param(
     [Parameter(Mandatory = $true)]
     [String]
     $RDGfilepath,
     [Parameter(Mandatory = $false)]
     [String]
     $CSVdirectory
 )
 [xml]$RDGfile = Get-Content -Path $RDGfilepath
 $node = $RDGfile.RDCMan.file
 if (!$CSVdirectory){
     $csvPath = [System.IO.Path]::GetDirectoryName($RDGfilepath) +
[System.IO.Path]::GetFileNameWithoutExtension($RDGfilepath) +
"_WAC.csv"
 } else {
     $csvPath = $CSVdirectory +
[System.IO.Path]::GetFileNameWithoutExtension($RDGfilepath) +
"_WAC.csv"
 }
 New-item -Path $csvPath
 Add-Content -Path $csvPath -Value '"name","type","tags"'
 AddServers -node $node -csvPath $csvPath
 Write-Host "Converted $RDGfilepath `nOutput: $csvPath"
}
```

2. To create a CSV file, run the following command:

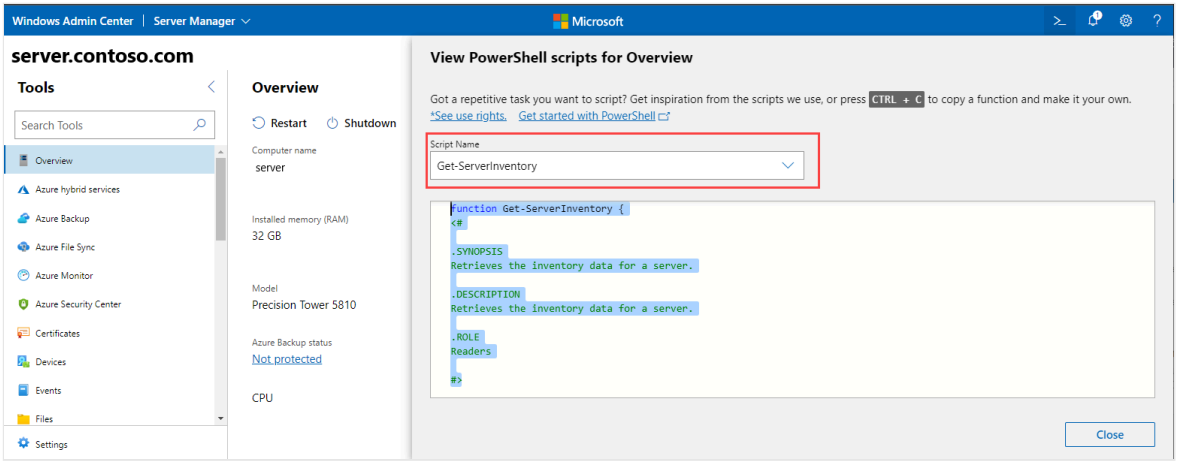
```PowerShell
RdgToWacCsv -RDGfilepath "path\to\myRDCManfile.rdg"
```

3. Import the resulting CSV file into Windows Admin Center. Tags in the connection list represent your RDCMan grouping hierarchy.

View PowerShell scripts used in Windows Admin Center

After you connect to a server, cluster, or PC, you can view the PowerShell scripts that power the UI actions available in Windows Admin Center:

1. From within a tool, select the PowerShell icon on the application bar.
2. In the dropdown list, select a command of interest to go to the corresponding PowerShell script.



Feedback

Was this page helpful? Yes No

Provide product feedback

Manage servers in Windows Admin Center

Article • 02/10/2025 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

> **Tip**
>
> New to Windows Admin Center? Download or learn more about Windows Admin Center.

You can add individual servers running Windows Server to Windows Admin Center so that you can manage them by using a comprehensive set of tools.

Procedure for adding a server to Windows Admin Center

1. Under **All connections**, select **+ Add**.
2. For the **Servers** resource type, select **Add**.
3. Enter the name of the server. If you're prompted, enter the credentials to use.
4. Select **Add** to finish.

The server is added to your list of connections on the **All connections** page. Select it to connect to the server.

> **Note**
>
> You can also add failover clusters or hyper-converged clusters as a separate connection in Windows Admin Center.

Summary of tools

The following table briefly describes the tools that are available in Windows Admin Center for server connections. Sections in this article provide more details for all the tools except Performance Monitor and Packetmon (whose links go to blog posts).

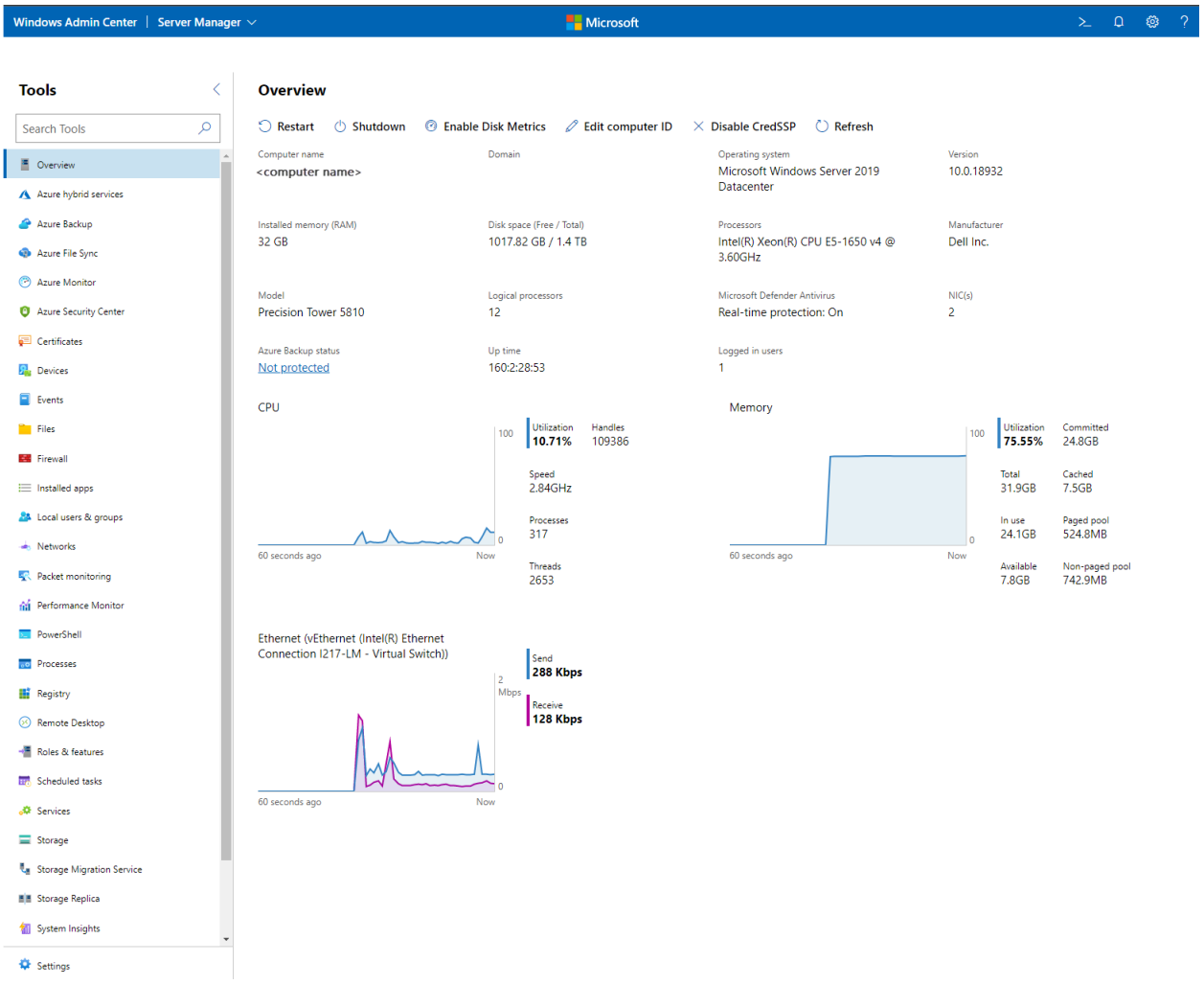
Expand table

Tool	Description
Overview	View server details and control server state.
Settings	View and modify services.
Active Directory	Manage Active Directory.
Azure Backup	View and configure Azure Backup.
Azure File Sync	View and configure Azure File Sync.
Azure hybrid center	View and configure Azure hybrid services.
Azure Monitor	View and configure Azure Monitor.
Certificates	View and modify certificates.
Containers	View containers.
Devices	View and modify devices.
DHCP	View and manage DHCP server configuration.
DNS	View and manage DNS server configuration.
Events	View events.
Files	Browse through files and folders.
Firewall	View and modify firewall rules.
Installed apps	View and remove installed apps.
Local users & groups	View and modify local users and groups.
Microsoft Defender for Cloud	View and configure Microsoft Defender for Cloud.
Networks	View and modify network devices.
Packetmon	Monitor network packets.
Performance Monitor	View performance counters and reports.
PowerShell	Interact with servers via PowerShell.
Processes	View and modify running processes.
Registry	View and modify registry entries.
Remote Desktop	Interact with servers via Remote Desktop.
Roles & features	View and modify roles and features.

Tool	Description
Scheduled tasks	View and modify scheduled tasks.
Security	View and modify security settings.
Services	View and modify services.
Storage	View and modify storage devices.
Storage Migration Service	Migrate servers and file shares to Azure or Windows Server 2019.
Storage Replica	Manage server-to-server storage replication.
System Insights	Get increased insight into the functioning of your servers.
Updates	View installed updates and check for new updates.
Virtual machines	View and manage virtual machines.
Virtual switches	View and manage virtual switches.

Overview

The **Overview** page shows the current state of CPU, memory, and network performance. You can also use it to perform operations and modify settings on a target computer or server.



Overview supports these features:

- View server details.
- View CPU activity.
- View memory activity.
- View network activity.
- Restart the server.
- Shut down the server.
- Enable disk metrics on the server.
- Edit the computer ID on the server.
- View the baseboard management controller (BMC) IP address with a hyperlink. This feature requires a BMC that's compatible with Intelligent Platform Management Interface (IPMI).

Active Directory (preview)

Active Directory is an early preview that's available on the extension feed.

The following Active Directory management features are available:

- Create a user.

- Create a group.
- Search for users, computers, and groups.
- View a details pane for users, computers, and groups when they're selected in a grid.
- Perform global grid actions for users, computers, and groups (disable, enable, and remove).
- Reset user passwords.
- Configure basic properties and group memberships for user objects.
- Configure delegation to a single machine for computer objects.
- Manage membership for group objects. (Add or remove one user at a time.)

Azure Backup

Use **Azure Backup** to help protect your Windows server from corruptions, attacks, or disasters by backing up your server directly to Microsoft Azure. Learn more about Azure Backup.

The following features are available:

- View an overview of your Azure backup status.
- Configure backup items and schedule.
- Start or stop a backup job.
- View backup job history and status.
- View recovery points and recover data.
- Delete backup data.

Azure File Sync

Use **Azure File Sync** to sync your file server with the cloud. Learn more about Azure File Sync.

Azure hybrid center

Azure hybrid center is your centralized location for learning about and onboarding to Azure hybrid services. Learn more about Azure hybrid services in Windows Admin Center.

Azure Monitor

Use **Azure Monitor** to monitor your servers and configure alerts. Learn more about Azure Monitor.

Certificates

Use **Certificates** to manage certificate stores on a computer or server.

The following features are available:

- Browse through and search for existing certificates.
- View certificate details.
- Export certificates.
- Renew certificates.
- Request new certificates.
- Delete certificates.

Containers

Use **Containers** to view the containers on a Windows Server container host. In the case of a running Windows Server Core container, you can view the event logs and access the command-line interface of the container. It's available on the extension feed.

Devices

Use **Devices** to manage connected devices on a computer or server.

The following features are available:

- Browse through and search for devices.
- View device details.
- Disable a device.
- Update a driver on a device.

DHCP

Use **DHCP** to manage connected devices on a computer or server. It's available on the extension feed.

The following features are available:

- Create, configure, and view IPv4 and IPv6 scopes.

- Create address exclusions, and configure start and end IP addresses.
- Create address reservations, and configure client MAC address (IPv4), DUID (IPv6), and IAID (IPv6).

DNS

Use **DNS** to manage connected devices on a computer or server. It's available on the extension feed.

The following features are available:

- View details of DNS forward lookup zones, reverse lookup zones, and DNS records.
- Create forward lookup zones (primary, secondary, and stub), and configure the properties of forward lookup zones.
- Create Host (A or AAAA), CNAME, and MX types of DNS records.
- Configure the properties of DNS records.
- Create IPv4 and IPv6 reverse lookup zones (primary, secondary, and stub), and configure the properties of reverse lookup zones.
- Create PTR and CNAME types of DNS records under reverse lookup zones.

Events

Use **Events** to manage event logs on a computer or server.

The following features are available:

- Browse through and search for events.
- View event details.
- Clear events from the log.
- Export events from the log.
- Create workspaces (preview).
- Save workspaces (preview).
- Delete workspaces (preview).
- View events in a stacked bar format (preview).

Files

Use **Files** to manage files and folders on a computer or server.

The following features are available:

- Browse through files and folders.

- Search for a file or folder.
- Create a folder.
- Delete a file or folder.
- Download a file or folder.
- Upload a file or folder.
- Rename a file or folder.
- Extract a .zip file.
- Copy and move files and folders.
- View file or folder properties.
- Add, edit, or remove file shares.
- Modify user and group permissions on file shares.
- Modify the security of a file server.

Firewall

Use **Firewall** to manage firewall settings and rules on a computer or server.

The following features are available:

- View an overview of firewall settings.
- View incoming firewall rules.
- View outgoing firewall rules.
- Search for firewall rules.
- View firewall rule details.
- Create a firewall rule.
- Enable or disable a firewall rule.
- Delete a firewall rule.
- Edit the properties of a firewall rule.

Installed apps

Use **Installed apps** to list and uninstall applications that are installed.

Local users & groups

Use **Local users & groups** to manage security groups and users that exist locally on a computer or server.

The following features are available:

- View and search for users and groups.

- Create a user or group.
- Manage a user's group membership.
- Delete a user or group.
- Change a user's password.
- Edit the properties of a user or group.

Microsoft Defender for Cloud

Microsoft Defender for Cloud is a cloud-native application protection platform. It has a set of security measures and practices that help protect cloud-based applications from various cyberthreats and vulnerabilities.

Networks

Use **Networks** to manage network devices and settings on a computer or server.

The following features are available:

- Browse through and search for existing network adapters.
- View the details of a network adapter.
- Edit the properties of a network adapter.
- Create an Azure Network Adapter (preview feature).

PowerShell

Use **PowerShell** to interact with a computer or server via a PowerShell session.

The following features are available:

- Create an interactive PowerShell session on a server.
- Disconnect from a PowerShell session on a server.

Processes

Use **Processes** to manage running processes on a computer or server.

The following features are available:

- Browse through and search for running processes.
- View process details.
- Start a process.

- End a process.
- Create a process dump.
- Find process handles.

Registry

Use **Registry** to manage registry keys and values on a computer or server.

The following features are available:

- Browse through registry keys and values.
- Add or modify registry values.
- Delete registry values.

Remote Desktop

Use **Remote Desktop** to interact with a computer or server via an interactive desktop session.

The following features are available:

- Start an interactive remote desktop session.
- Disconnect from a remote desktop session.
- Send Ctrl+Alt+Delete to a remote desktop session.

Roles & features

Use **Roles & features** to manage roles and features on a server.

The following features are available:

- Browse through roles and features on a server.
- View role or feature details.
- Install a role or feature.
- Remove a role or feature.

Scheduled tasks

Use **Scheduled tasks** to manage scheduled tasks on a computer or server.

The following features are available:

- Browse the task scheduler library.
- Edit scheduled tasks.
- Enable and disable scheduled tasks.
- Start and stop scheduled tasks.
- Create scheduled tasks.

Security

Use **Security** to manage your security settings on a computer or server.

The following features are available:

- Run and schedule virus scans.
- Enable and disable real-time threat protection.
- View threat history.
- Check secured-core status.
- Enable or disable secured-core security features.

Services

Use **Services** to manage services on a computer or server.

The following features are available:

- Browse through and search for services on a server.
- View details of a service.
- Start a service.
- Pause a service.
- Restart a service.
- Edit the properties of a service.

Settings

Settings is a central location to manage settings on a computer or server.

The following features are available:

- View and modify file share settings.
- View and modify user and system environment variables.
- View and modify the power configuration.
- View and modify Remote Desktop settings.
- View and modify settings for role-based access control.

- View and modify Hyper-V host settings, if applicable.

Storage

Use **Storage** to manage storage devices on a computer or server.

The following features are available:

- Browse through and search for existing disks on a server.
- View disk details.
- Create a volume.
- Initialize a disk.
- Create, attach, and detach a virtual hard disk.
- Take a disk offline.
- Format a volume.
- Resize a volume.
- Edit volume properties.
- Delete a volume.
- Install Quota Management.
- Manage File Server Resource Manager quotas.

Storage Migration Service

Use **Storage Migration Service** to migrate servers and file shares to Azure or Windows Server 2019, without requiring apps or users to change anything. Get an overview of Storage Migration Service.

> **Note**
>
> Storage Migration Service requires Windows Server 2019.

Storage Replica

Use **Storage Replica** to manage server-to-server storage replication. Learn more about Storage Replica.

System Insights

System Insights introduces predictive analytics natively in Windows Server to give you increased insight into the functioning of your server. Get an overview of System Insights.

> **Note**
>
> System Insights requires Windows Server 2019.

Updates

Use **Updates** to manage Microsoft and Windows updates on a computer or server.

The following features are available:

- View available Microsoft and Windows updates.
- View the update history.
- Install updates.
- Check online for updates from Microsoft Update.
- Manage Azure Update Management integration.

Virtual machines

See Manage virtual machines by using Windows Admin Center.

Virtual switches

Use **Virtual switches** to manage Hyper-V virtual switches on a computer or server.

The following features are available:

- Browse through and search for virtual switches on a server.
- Create a virtual switch.
- Rename a virtual switch.
- Delete a virtual switch.
- Edit the properties of a virtual switch.

Feedback

Was this page helpful? Yes No

Provide product feedback

Manage Windows Defender Application Control (WDAC)-enforced infrastructure

Article • 02/10/2025 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Windows Defender Application Control (WDAC) can help mitigate many security threats by restricting the applications that users are allowed to run and the code that runs in the system core (kernel). Application control policies can also block unsigned scripts and MSI files, and restrict Windows PowerShell to running in ConstrainedLanguage mode. Learn more about application control for Windows.

Extra configuration is required for Windows Admin Center to install on and manage WDAC-enforced environments. This article covers these requirements and known issues in managing a WDAC-enforced environment.

Requirements

The following sections provide the requirements for using Windows Admin Center to manage your WDAC-enforced infrastructure (servers, client machines, or clusters).

Policy requirements

Depending on your use case, you need to add one or more certificates to your allowlist as part of your base or supplemental policies. Learn more about deploying a base or supplemental policy.

- **Case 1**: Only your managed nodes have WDAC enforced.
- **Case 2**: Both your managed node and the machine on which you deploy Windows Admin Center have WDAC enforced.

For Case 1, only the following signer rule needs to be allowlisted in the WDAC policy on your managed node:

XML

```
<Signer ID="ID_SIGNER_S_XXXXX" Name="Microsoft Code Signing PCA 2011">
  <CertRoot Type="TBS"
Value="F6F717A43AD9ABDDC8CEFDDE1C505462535E7D1307E630F9544A2D14FE8BF26E" />
  <CertPublisher Value="Microsoft Corporation" />
</Signer>
```

For Case 2, the preceding signer rule must be added to the allowlist on *both* your managed node and the machine on which you deploy Windows Admin Center. Additionally, the following signer rules must be added to the allowlist *only* on the machine on which you deploy Windows Admin Center:

XML

```
<Signer ID="ID_SIGNER_S_XXXXX" Name="Microsoft Code Signing PCA 2011">
  <CertRoot Type="TBS"
Value="F6F717A43AD9ABDDC8CEFDDE1C505462535E7D1307E630F9544A2D14FE8BF26E" />
  <CertPublisher Value="Microsoft 3rd Party Application Component" />
</Signer>
<Signer ID="ID_SIGNER_S_XXXXX" Name="Microsoft Code Signing PCA 2011">
  <CertRoot Type="TBS"
Value="F6F717A43AD9ABDDC8CEFDDE1C505462535E7D1307E630F9544A2D14FE8BF26E" />
  <CertPublisher Value=".NET" />
</Signer>
```

The signer rule with `CertPublisher Value=".NET"` is not required if you're using a version of Windows Admin Center older than 2410. However, these older versions require the following file/hash rules to be allowlisted *only* on the machine on which you deploy Windows Admin Center:

XML

```
<FileRules>
    <!--Requirement from WAC to allow files from WiX-->
    <Allow ID="ID_ALLOW_E_X_XXXX_X" FriendlyName="WiX wixca.dll"
Hash="9DE61721326D8E88636F9633AA37FCB885A4BABE" />
    <Allow ID="ID_ALLOW_E_X_XXXX_XXXX_X" FriendlyName="WiX wixca.dll"
Hash="B216DFA814FC856FA7078381291C78036CEF0A05" />
    <Allow ID="ID_ALLOW_E_X_XXXX_X" FriendlyName="WiX wixca.dll"
Hash="233F5E43325615710CA1AA580250530E06339DEF861811073912E8A16B058C69" />
    <Allow ID="ID_ALLOW_E_X_XXXX_XXXX_X" FriendlyName="WiX wixca.dll"
Hash="B216DFA814FC856FA7078381291C78036CEF0A05" />
    <Allow ID="ID_ALLOW_E_X_XXXX_X" FriendlyName="WiX wixca.dll 2"
Hash="EB4CB5FF520717038ADADCC5E1EF8F7C24B27A90" />
    <Allow ID="ID_ALLOW_E_X_XXXX_XXXX_X" FriendlyName="WiX wixca.dll 2"
Hash="6C65DD86130241850B2D808C24EC740A4C509D9C" />
    <Allow ID="ID_ALLOW_E_X_XXXX_X" FriendlyName="WiX wixca.dll 2"
Hash="C8D190D5BE1EFD2D52F72A72AE9DFA3940AB3FACEB626405959349654FE18B74" />
    <Allow ID="ID_ALLOW_E_X_XXXX_XXXX_X" FriendlyName="WiX wixca.dll 2"
Hash="6C65DD86130241850B2D808C24EC740A4C509D9C" />
    <Allow ID="ID_ALLOW_E_X_XXXX_X" FriendlyName="WiX firewall.dll"
Hash="2F0903D4B21A0231ADD1B4CD02E25C7C4974DA84" />
    <Allow ID="ID_ALLOW_E_X_XXXX_XXXX_X" FriendlyName="WiX firewall.dll"
Hash="868635E434C14B65AD7D7A9AE1F4047965740786" />
    <Allow ID="ID_ALLOW_E_X_XXXX_X" FriendlyName="WiX firewall.dll"
Hash="5C29B8255ACE0CD94C066C528C8AD04F0F45EBA12FCF94DA7B9CA1B64AD4288B" />
    <Allow ID="ID_ALLOW_E_X_XXXX_XXXX_X" FriendlyName="WiX firewall.dll"
```

```
Hash="868635E434C14B65AD7D7A9AE1F4047965740786" />
  </FileRules>
```

The policy creation tool/script should automatically generate `Signer ID` and `Allow ID` (that is, `Signer ID="ID_SIGNER_S_XXXXX"`). For more info, refer to the WDAC documentation.

> **Tip**
>
> The App Control for Business Wizard can be helpful for creating and editing WDAC policies. Remember that when you're creating a new policy, whether by using the wizard or the PowerShell commands, use the `Publisher` rule on binaries to generate rules. For example, when you're using the wizard, you can generate the WDAC policy for Case 1 based on the Windows Admin Center .msi file. For Case 2, you can still use the wizard, but you need to manually edit your WDAC policy to include the listed signer and hash rule.

Network requirements

By default, Windows Admin Center communicates with your servers via WinRM over HTTP (port 5985) or HTTPS (port 5986). For WDAC-enforced infrastructure, Windows Admin Center additionally needs SMB access to the managed nodes (TCP port 445).

Permissions

File transfer based on UNC paths over SMB port 445 is critical for Windows Admin Center to manage these environments. Make sure that you're an administrator on the managed server or cluster. Also make sure that security policies aren't blocking file transfers.

PowerShell execution policy

The default PowerShell execution policy is sufficient for Windows Admin Center to manage a WDAC-enforced machine. However, if the default execution policy changes on the machine, you need to ensure that the `LocalMachine` scope is set to `RemoteSigned` to allow signed scripts to be loaded and executed. This is a PowerShell security feature, and you should make changes only when they're appropriate and necessary.

Installing

Install Windows Admin Center on your WDAC-enforced server or client machine just as you normally would. If you meet the preceding requirements, Windows Admin Center should install and function as normal.

Connecting

Connect to your WDAC-enforced server, client, or cluster machines as you normally would. Upon connecting to your server, you can track the enforcement status via the **PowerShell Language Mode** field on the **Overview** page. If the value of this field is **Constrained**, WDAC is being enforced.

When you connect to a WDAC-enforced cluster for the first time, Windows Admin Center might take a few minutes to set up a connection to your cluster. Subsequent connections won't have a delay.

> ⓘ **Note**
>
> If you change the WDAC enforcement status of your managed nodes, don't use Windows Admin Center for at least 30 seconds for this change to be reflected.

Known issues

- Currently, deploying Azure Kubernetes Service on Azure Local and Azure Arc resource bridge through Windows Admin Center isn't supported in a WDAC-enforced environment.
- Using role-based access control (RBAC) on a single server is currently not supported.
- Certain operations in the Certificates tool are currently not supported.

Troubleshooting

If you get a "Module not found" or "failed to connect" error:

- To confirm whether Windows Admin Center successfully transferred files to your managed node, go to the `%PROGRAMFILES%\WindowsPowerShell\Modules` folder on your managed node and verify that modules with the name `Microsoft.SME.*` exist in that folder.

If the modules don't exist, reconnect to your server or cluster from Windows Admin Center.

- Ensure that the machine where Windows Admin Center is installed has access to TCP port 445 on the managed node.

Related content

- WDAC design guide
- WDAC deployment guide
- AppLocker overview

Feedback

Was this page helpful? Yes No

Provide product feedback

Deploy a hyperconverged infrastructure by using Windows Admin Center

Article • 02/10/2025 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

You can use Windows Admin Center version 1910 or later to deploy a hyperconverged infrastructure by using two or more suitable Windows servers. This feature takes the form of a multistage workflow that guides you through installing features, configuring networking, creating the cluster, and deploying Storage Spaces Direct and/or software-defined networking (SDN).

As of Windows Admin Center version 2007, Windows Admin Center supports the Azure Stack HCI operating system. To read about how to deploy a cluster in Windows Admin Center, see the Azure Local documentation. Although that documentation is focused on Azure Local, most of the instructions are applicable to Windows Server deployments.

Undo and start over

Use the following Windows PowerShell cmdlets to undo changes that the workflow made and start over.

Remove virtual machines or other clustered resources

If you created any virtual machines or other clustered resources, such as the network controllers for SDN, remove them first.

For example, to remove resources by name, use this cmdlet:

```PowerShell
Get-ClusterResource -Name "<NAME>" | Remove-ClusterResource
```

Undo the storage steps

If you enabled Storage Spaces Direct, disable it by using the following script.

> **⚠ Warning**
>
> These cmdlets permanently delete any data in Storage Spaces Direct volumes. You can't undo this action.

PowerShell

```
Get-VirtualDisk | Remove-VirtualDisk
Get-StoragePool -IsPrimordial $False | Remove-StoragePool
Disable-ClusterS2D
```

Undo the clustering steps

If you created a cluster, remove it by using this cmdlet:

PowerShell

```
Remove-Cluster -CleanUpAD
```

To also remove cluster validation reports, run this cmdlet on every server that was part of the cluster:

PowerShell

```
Get-ChildItem C:\Windows\cluster\Reports\ | Remove-Item
```

Undo the networking steps

Run the following cmdlets on every server that was part of the cluster.

If you created a Hyper-V virtual switch, use this cmdlet:

PowerShell

```
Get-VMSwitch | Remove-VMSwitch
```

The `Remove-VMSwitch` cmdlet automatically removes any virtual adapters and undoes switch-embedded teaming of physical adapters.

If you modified network adapter properties such as name, IPv4 address, and VLAN ID, use the following cmdlets.

> **⚠ Warning**
>
> These cmdlets remove network adapter names and IP addresses. Make sure that you have the information that you need to connect afterward, such as an adapter for management that's excluded from the script. Also make sure that you know

how the servers are connected in terms of physical properties like MAC address, not just the adapter's name in Windows.

PowerShell

```
Get-NetAdapter | Where Name -Ne "Management" | Rename-NetAdapter -NewName
$(Get-Random)
Get-NetAdapter | Where Name -Ne "Management" | Get-NetIPAddress -ErrorAction
SilentlyContinue | Where AddressFamily -Eq IPv4 | Remove-NetIPAddress
Get-NetAdapter | Where Name -Ne "Management" | Set-NetAdapter -VlanID 0
```

You're now ready to start the workflow.

Related content

- Windows Admin Center overview
- Deploy Storage Spaces Direct on Windows Server

Feedback

Was this page helpful? Yes No

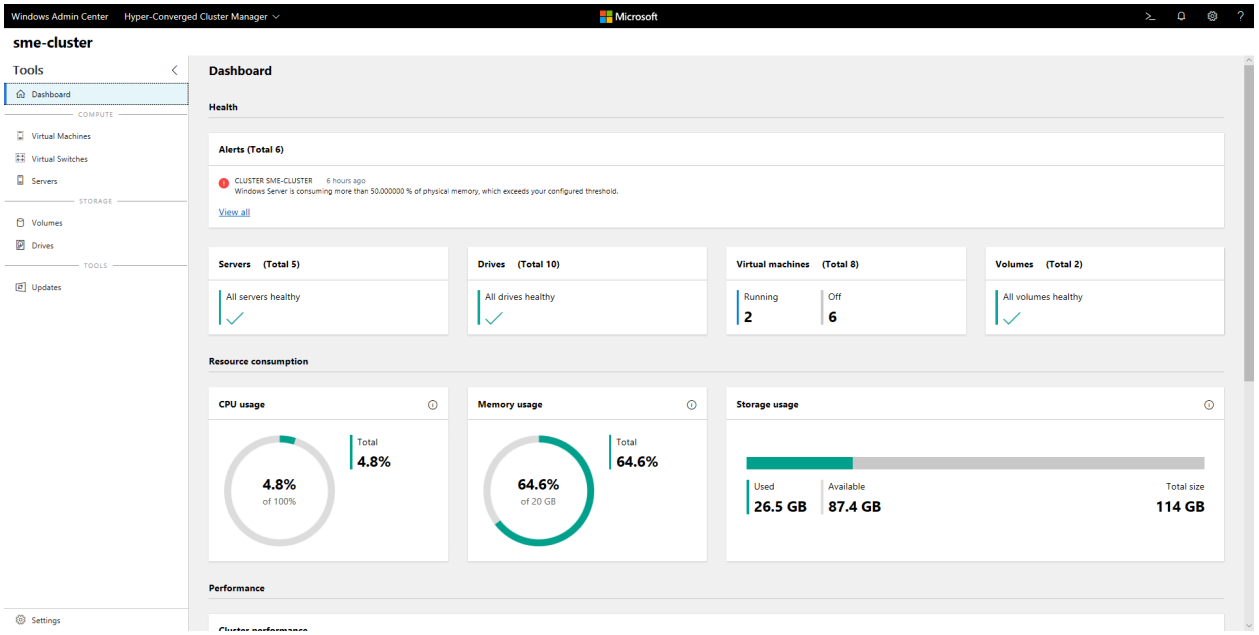
Provide product feedback

Manage a hyper-converged infrastructure by using Windows Admin Center

Article • 02/10/2025 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

A hyper-converged infrastructure consolidates software-defined compute, storage, and networking into one cluster to provide high-performance, cost-effective, and scalable virtualization. This capability was introduced in Windows Server 2016 with Storage Spaces Direct, Software Defined Networking (SDN), and Hyper-V.

You can use Windows Admin Center to manage and monitor a hyper-converged infrastructure that's running Windows Server 2016 or Windows Server 2019. Windows Admin Center is the next-generation management tool for Windows Server. It's the successor to traditional "in-box" tools like Server Manager. It's free and can be installed and used without an internet connection.



Some of the features that this article describes are available only in Windows Admin Center Preview. For more information, see Windows Admin Center overview.

> **Tip**
>
> Looking to acquire a hyper-converged infrastructure? We recommend **these Windows Server SDN solutions** from our partners. They're designed, assembled, and validated against our reference architecture to help ensure compatibility and reliability, so you get up and running quickly.

Key features

Highlights of Windows Admin Center for hyper-converged infrastructures include:

- **Unified dashboard for compute, storage, and networking**. View your virtual machines (VMs), host servers, volumes, drives, and more within one purpose-built, consistent, interconnected experience.
- **Creation and management of storage spaces and Hyper-V virtual machines**. Use simple workflows to create, open, resize, and delete volumes. Or use workflows to create, start, connect to, and move virtual machines.
- **Powerful cluster-wide monitoring**. The dashboard graphs memory and CPU usage, storage capacity, input/output operations per second (IOPS), throughput, and latency in real time across every server in the cluster. Get clear alerts when something isn't right.
- **Software Defined Networking support**. Manage and monitor virtual networks and subnets, connect virtual machines to virtual networks, and monitor the SDN infrastructure.

We're actively developing Windows Admin Center for hyper-converged infrastructures. It receives frequent updates that improve existing features and add new features.

Before you start

To manage your cluster as a hyper-converged infrastructure in Windows Admin Center, make sure that the cluster is running Windows Server 2016 or Windows Server 2019. Also make sure that you enabled Hyper-V and Storage Spaces Direct for it. Optionally, the cluster can also have Software Defined Networking enabled and managed through Windows Admin Center.

> **Tip**
>
> Windows Admin Center also offers a general-purpose management experience for any cluster that supports any workload. It's available for Windows Server 2012 and later. If this option sounds like a better fit, when you add your cluster to Windows Admin Center, select Failover Cluster instead of **Hyper-Converged Cluster**.

Prepare your Windows Server 2016 cluster for Windows Admin Center

Windows Admin Center for hyper-converged infrastructures depends on management APIs added after Windows Server 2016 was released. Before you can manage your Windows Server 2016 cluster by using Windows Admin Center, you need to perform these two steps:

1. Verify that every server in the cluster installed the 2018-05 Cumulative Update for Windows Server 2016 (KB4103723) or later. To download and install this update, go to **Settings** > **Update & Security** > **Windows Update** and select **Check online for updates from Microsoft Update**.

2. Run the following PowerShell cmdlet as administrator on the cluster:

```PowerShell
 Add-ClusterResourceType -Name "SDDC Management" -dll
"$env:SystemRoot\Cluster\sddcres.dll" -DisplayName "SDDC Management"
```

> **Tip**
>
> You need to run the cmdlet only once, on any server in the cluster. You can run it locally in Windows PowerShell or use CredSSP to run it remotely. Depending on your configuration, you might not be able to run this cmdlet from within Windows Admin Center.

Prepare your Windows Server 2019 cluster for Windows Admin Center

If your cluster runs Windows Server 2019, the preceding steps are not necessary. Just add the cluster to Windows Admin Center, as described later in this article.

Configure Software Defined Networking (optional)

You can configure your hyper-converged infrastructure running Windows Server 2016 or 2019 to use Software Defined Networking by following these steps:

1. Prepare the virtual hard disk (VHD) of the operating system, which is the same operating system that you installed on the hyper-converged infrastructure hosts. This VHD will be used for all NC-series, Software Load Balancer (SLB), and gateway VMs.

2. Download all the folders and files under SDN Express from GitHub.

3. Prepare a different VM by using the deployment console. This VM should be able to access the SDN hosts. Also, the VM should have the Remote Server Administration Tools (RSAT) Hyper-V package installed.

4. Copy everything that you downloaded for SDN Express to the deployment console VM, and share this **SDNExpress** folder. Make sure every host can access the **SDNExpress** shared folder, as defined in line 8 of the configuration file:

   ```
   \\$env:Computername\SDNExpress
   ```

5. Copy the VHD of the operating system to the **images** folder under the **SDNExpress** folder on the deployment console VM.

6. Modify the SDN Express configuration based on your environment setup.

7. Run PowerShell with admin privileges to deploy SDN:

   ```PowerShell
   .\SDNExpress.ps1 -ConfigurationDataFile .\your_fabricconfig.PSD1 -verbose
   ```

The deployment takes around 30 to 45 minutes.

Start managing your hyper-converged infrastructure

After your Hyper-Converged Infrastructure is deployed, you can manage it by using Windows Admin Center.

Install Windows Admin Center

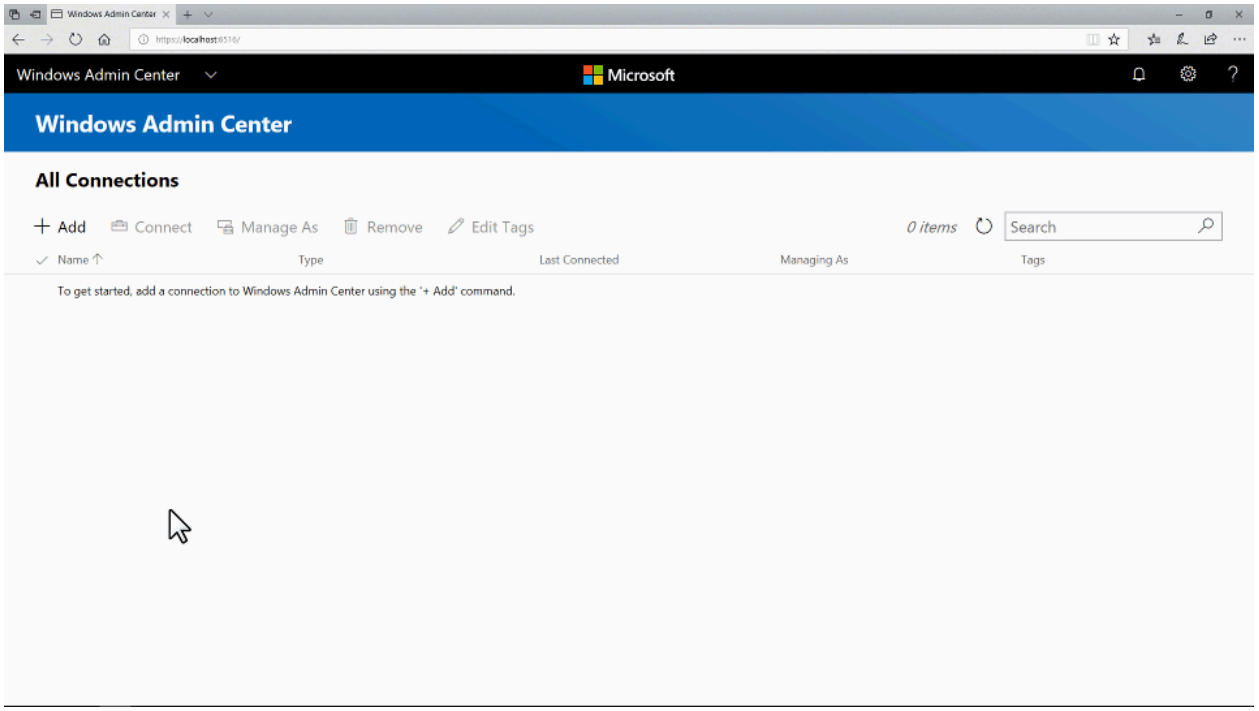
If you haven't already, download and install Windows Admin Center. The fastest way to get up and running is to install it on your Windows 10 computer and manage your servers remotely. This process takes less than five minutes. Download now, or learn more about other installation options.

Add a hyper-converged cluster

To add your cluster to Windows Admin Center:

1. Under **All Connections**, select **+ Add**.
2. Select **Hyper-Converged Cluster Connection** as the connection type.
3. Enter the name of the cluster. If you're prompted, select the credentials to use.
4. Select **Add** to finish.

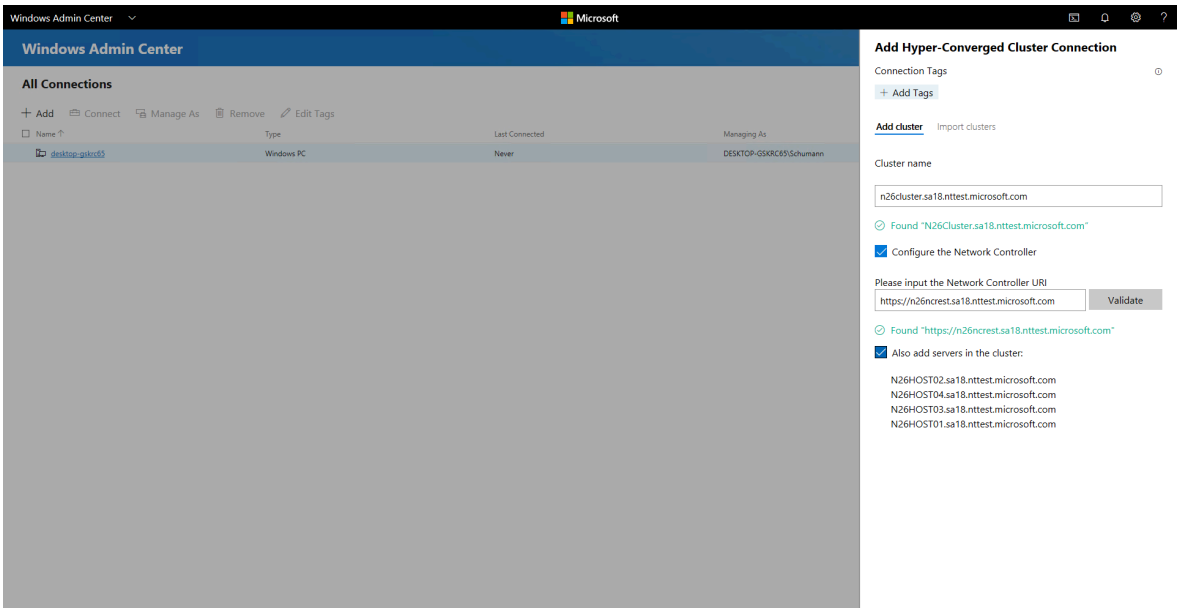
The cluster is added to your list of connections. Select it to open the dashboard.



Add an SDN-enabled hyper-converged cluster (Windows Admin Center Preview)

The latest version of Windows Admin Center Preview supports Software Defined Networking management for hyper-converged infrastructures. By adding a network controller REST URI to your hyper-converged cluster connection, you can use Hyper-converged Cluster Manager to manage your SDN resources and monitor the SDN infrastructure.

1. Under **All Connections**, select **+ Add**.
2. Select **Hyper-Converged Cluster Connection** as the connection type.
3. Enter the name of the cluster. If you're prompted, select the credentials to use.
4. Select **Configure the Network Controller**.
5. Enter the network controller URI value, and then select **Validate**.



6. Select **Add** to finish.

The cluster is added to your list of connections. Select it to open the dashboard.

> **Important**
>
> SDN environments with Kerberos authentication for Northbound communication are currently not supported.

Frequently asked questions

Are there differences between managing Windows Server 2016 and managing Windows Server 2019?

Yes. Windows Admin Center for hyper-converged infrastructures receives frequent updates that improve the experience for both Windows Server 2016 and Windows Server 2019. However, certain new features are available only for Windows Server 2019. An example is the toggle for deduplication and compression.

Can I use Windows Admin Center to manage Storage Spaces Direct for other use cases (not hyper-converged), such as converged Scale-Out File Server or Microsoft SQL Server?

Windows Admin Center for hyper-converged infrastructures does not provide management or monitoring options specifically for other use cases of Storage Spaces

Direct. For example, it can't create file shares. However, the dashboard and core features, such as creating volumes or replacing drives, work for any Storage Spaces Direct cluster.

What's the difference between a failover cluster and a hyper-converged cluster?

In general, the term *hyper-converged* refers to running Hyper-V and Storage Spaces Direct on the same clustered servers to virtualize compute and storage resources. In the context of Windows Admin Center, when you select **+ Add** in the list of connections, you can choose between these connection types:

- **Failover cluster connection**: The successor to the Failover Cluster Manager desktop app. It provides a familiar, general-purpose management experience for any cluster that supports any workload, including Microsoft SQL Server. It's available for Windows Server 2012 and later.
- **Hyper-converged cluster connection**: An experience tailored for Storage Spaces Direct and Hyper-V. It features the dashboard and emphasizes charts and alerts for monitoring. It's available for Windows Server 2016 and Windows Server 2019.

Why do I need the latest cumulative update for Windows Server 2016?

Windows Admin Center for hyper-converged infrastructures depends on management APIs developed since Windows Server 2016 was released. These APIs are added in the 2018-05 Cumulative Update for Windows Server 2016 (KB4103723), available as of May 8, 2018.

How much does it cost to use Windows Admin Center?

You can use Windows Admin Center (available as a separate download) with valid licenses of Windows Server or Windows 10 at no additional cost. It's licensed under a Windows supplement to the Microsoft Software License Terms.

Does Windows Admin Center require System Center?

No.

Does Windows Admin Center require an internet connection?

No.

Although Windows Admin Center offers powerful and convenient integration with the Microsoft Azure cloud platform, the core management and monitoring experience for hyper-converged infrastructures is completely on-premises. You can install and use it without an internet connection.

Things to try

Now that you understand the basics of using Windows Admin Center to manage and monitor a hyper-converged infrastructure, you can try more tasks in your environment.

Manage Storage Spaces Direct volumes

If you're just getting started, here are some quick videos to help you learn how Windows Admin Center for hyper-converged infrastructures is organized and works. Please exercise good judgment and be careful with production environments. These videos were recorded with Windows Admin Center version 1804 and an Insider Preview build of Windows Server 2019.

Create a three-way mirror volume

https://www.youtube-nocookie.com/embed/o66etKq70N8

Create a mirror-accelerated parity volume

https://www.youtube-nocookie.com/embed/R72QHudqWpE

Open a volume and add files

https://www.youtube-nocookie.com/embed/j59z7ulohs4

Turn on deduplication and compression

https://www.youtube-nocookie.com/embed/PRibTacyKko

Expand a volume

https://www.youtube-nocookie.com/embed/hqyBzipBoTI

Delete a volume

https://www.youtube-nocookie.com/embed/DbjF8r2F6Jo

Create a virtual machine

1. On the left pane, select **Virtual Machines**.
2. At the top of the **Virtual Machines** pane, select the **Inventory** tab, and then select **New**.
3. Enter the virtual machine name, and choose between generation 1 and generation 2.
4. Choose a host for the virtual machine, or use the recommended host.
5. Choose a path for the virtual machine files. Select a volume from the dropdown list, or select **Browse** to choose a folder. The virtual machine configuration files and virtual hard disk file will be saved in a single folder under the `\Hyper-V\[virtual machine name]` path of the selected volume or path.
6. Choose the number of virtual processors and whether you want nested virtualization enabled. Configure memory settings, network adapters, and virtual hard disks. Choose whether you want to install an operating system from an .iso image file or from the network.
7. Select **Create**.
8. After the virtual machine is created and appears in the list of virtual machines, start it.
9. Connect to the virtual machine's console via VMConnect to install the operating system. Select the virtual machine from the list, and then select **More** > **Connect** to download the .rdp file. Open the .rdp file in the Remote Desktop Connection app. Because you're connecting to the virtual machine's console, you need to enter the Hyper-V host's admin credentials.

Learn more about virtual machine management with Windows Admin Center.

Pause and safely restart a server

1. On the left pane of the dashboard, select **Servers**. Or, on the tile in the lower-right corner of the dashboard, select the **VIEW SERVERS** > link.
2. Switch from the **Summary** tab to the **Inventory** tab.

3. Select a server to open its details.

4. Select **Pause server for maintenance**. If it's safe to proceed, this action moves virtual machines to other servers in the cluster. The server has a status of **Draining** during this process.

   If you want, you can watch the virtual machines move on the **Virtual machines** > **Inventory** pane, where the grid clearly shows their host server. After all the virtual machines move, the server status is **Paused**.

5. Select **Manage server** to access all the per-server management tools in Windows Admin Center.

6. Select **Restart**, and then select **Yes**.

7. Back on the dashboard, the server is colored red while it's down. After it's back up, go to the **Server** page and select **Resume server from maintenance** to set the server status to **Up**. In time, virtual machines will move back. No user action is required.

Replace a failed drive

1. When a drive fails, an alert appears in the **Alerts** area of the dashboard.

   Or to browse through drives and see their status for yourself, you can either:

   - Select **Drives** on the left pane.
   - Select the **VIEW DRIVES** > link on the tile in the lower-right corner.

   On the **Inventory** tab, the grid supports sorting, grouping, and keyword search.

2. On the dashboard, select the alert to see details, like the drive's physical location.

   To learn more, select the **Go to drive** shortcut to the **Drive** detail page.

3. If your hardware supports it, you can select **Turn light on/off** to control the drive's indicator light.

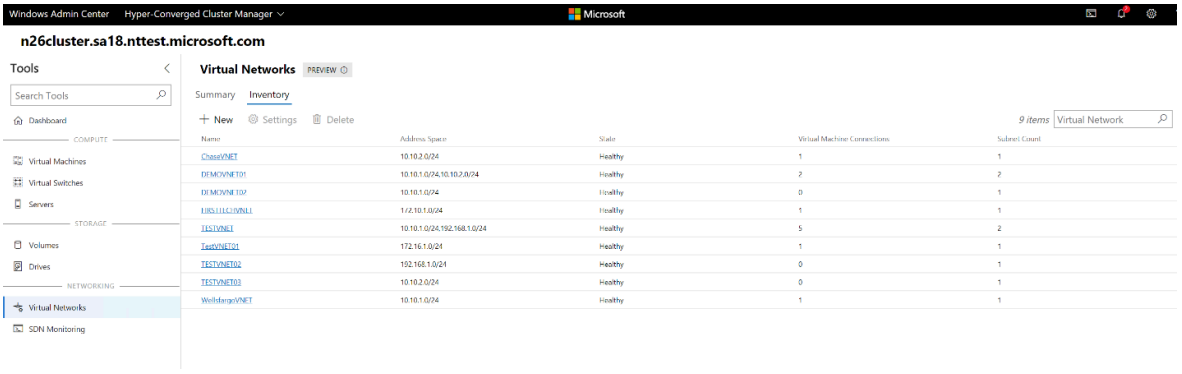
4. Storage Spaces Direct automatically retires and evacuates failed drives. When this process happens, the drive status is **Retired**, and its storage capacity bar is empty.

5. Remove the failed drive and insert its replacement.

6. In **Drives** > **Inventory**, the new drive appears. In time, the alert will clear, volumes will repair back to **OK** status, and storage will rebalance onto the new drive. No user action is required.

Manage virtual networks (SDN-enabled HCI clusters via Windows Admin Center Preview)

1. On the left pane, select **Virtual Networks**.



2. On the **Inventory** tab, select **New** to create a new virtual network and subnets. Or choose an existing virtual network and then select **Settings** to modify its configuration.

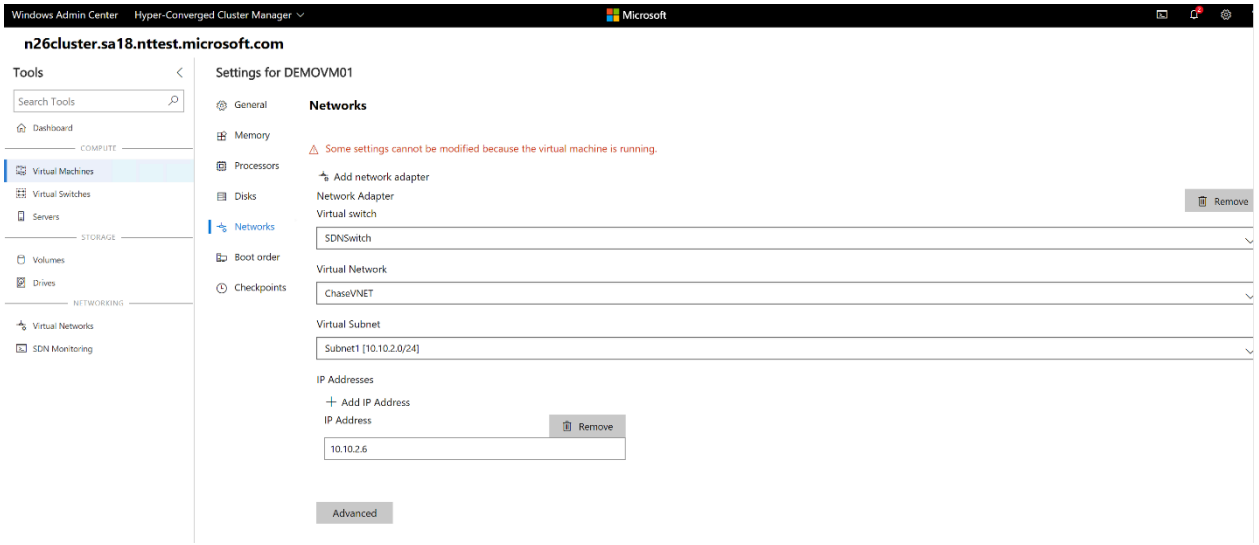
3. Select an existing virtual network to view VM connections to virtual network subnets and access control lists applied to virtual network subnets.

Connect a virtual machine to a virtual network (SDN-enabled HCI clusters via Windows Admin Center Preview)

1. On the left pane, select **Virtual Networks**.

2. Choose an existing virtual machine, select **Settings**, and then open the **Networks** tab.

3. Configure the **Virtual Network** and **Virtual Subnet** fields to connect the virtual machine to a virtual network.

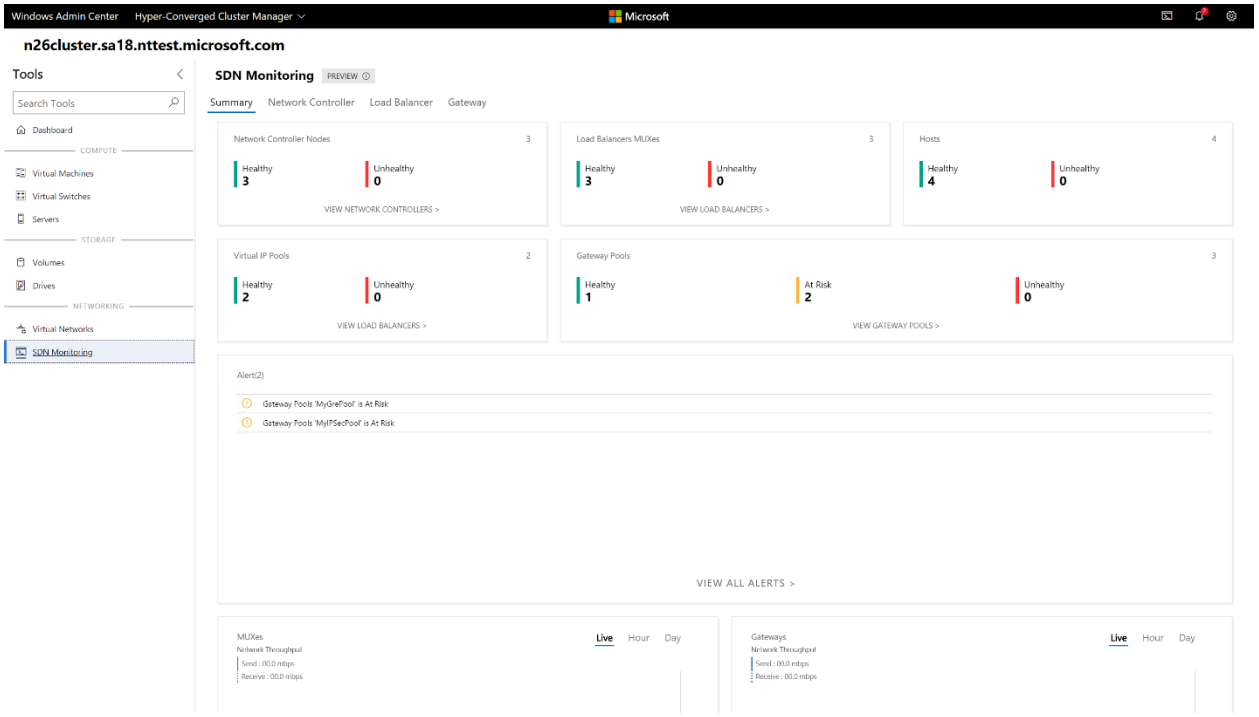


You can also configure the virtual network when you're creating a virtual machine.

Monitor Software Defined Networking infrastructure (SDN-enabled HCI clusters via Windows Admin Center Preview)

1. On the left pane, select **SDN Monitoring**.

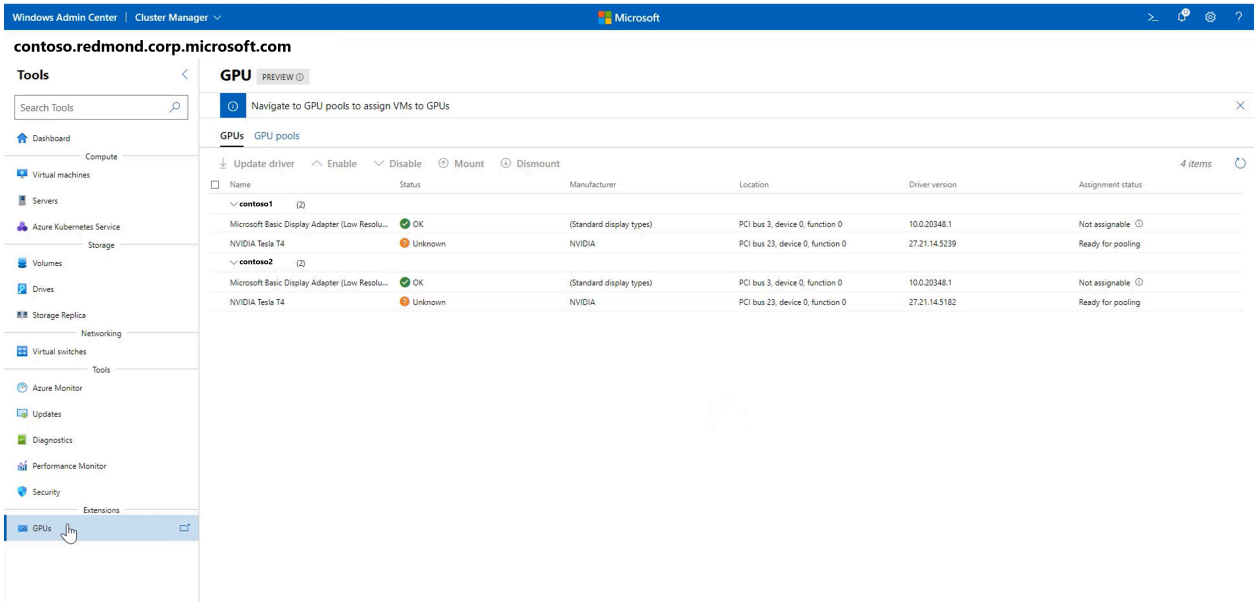
2. View detailed information about the health of your network controller, software load balancer, and virtual gateway. You can also monitor your virtual gateway pool, public and private IP pool usage, and SDN host status.



Manage GPUs

1. On the left pane, select **GPUs**.

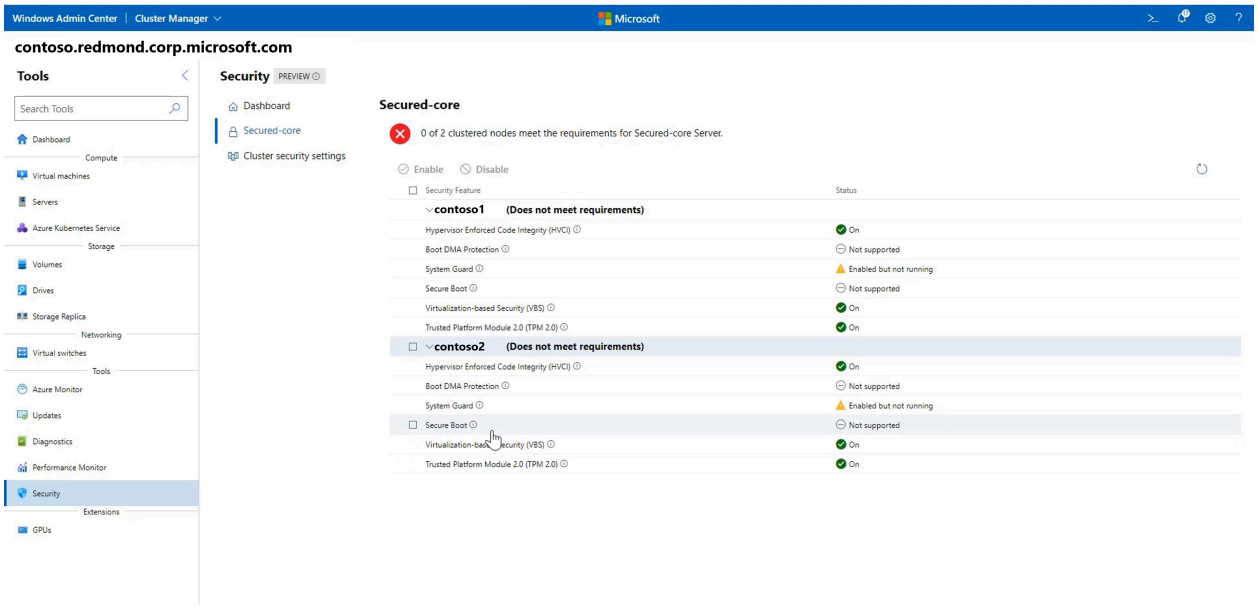
2. View the available GPUs from your clustered VMs, and provide GPU acceleration to workloads running in the clustered VMs through Discrete Device Assignment. Learn more about using GPUs with clustered VMs.



Use the security tool

1. On the left pane, select **Security**.

2. Select the **Secured-core** tab, and then enable or disable the available security features.



Give us feedback

The most important benefit of frequent updates is to hear what's working and what needs to be improved. Here are some ways to let us know what you're thinking:

- Submit ideas for feature requests and provide feedback
- Join the Windows Admin Center forum on Microsoft Tech Community
- Tweet to `@servermgmt`

Related content

- Windows Admin Center overview
- Storage Spaces Direct overview
- Hyper-V technology overview
- Software Defined Networking in Azure Stack HCI and Windows Server

Feedback

Was this page helpful? Yes No

Provide product feedback

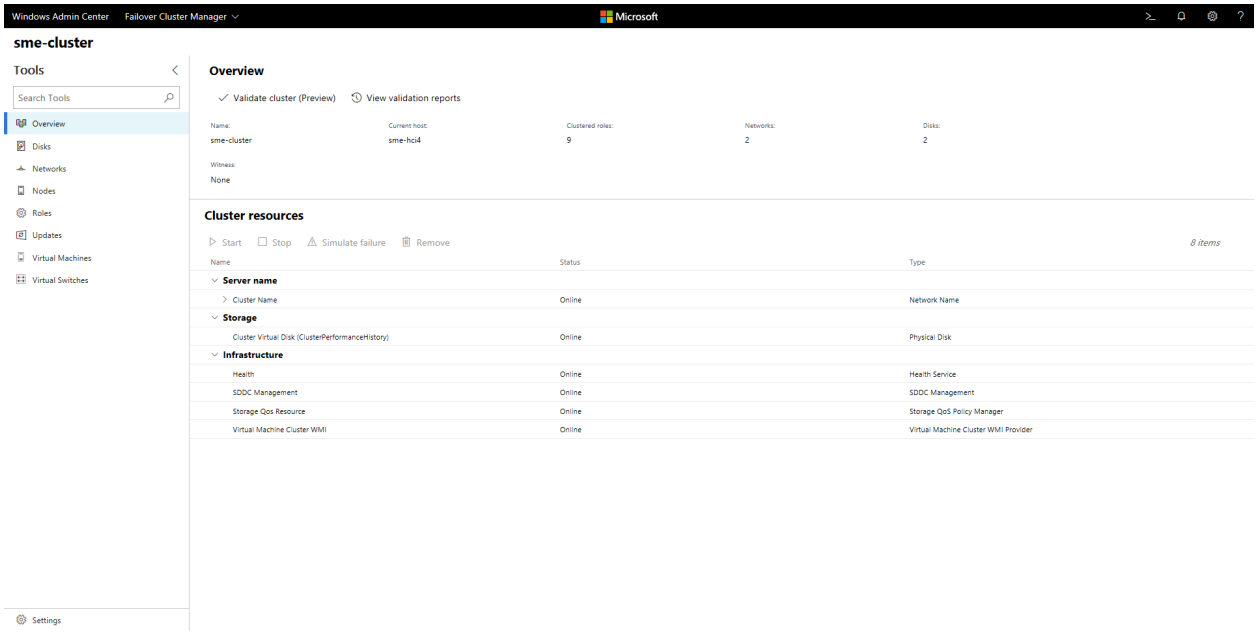
Manage failover clusters by using Windows Admin Center

Article • 02/10/2025 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

> **Tip**
>
> New to Windows Admin Center? Download or learn more about Windows Admin Center.

Failover Clustering is a Windows Server feature that enables you to group multiple servers together into a fault-tolerant cluster. The purpose is to increase availability and scalability of applications and services such as Scale-Out File Server, Hyper-V, and Microsoft SQL Server.

You can manage failover cluster nodes as individual servers by using Server Manager to add them in Windows Admin Center. But you can also add them as failover clusters to view and manage cluster resources, storage, networks, nodes, roles, virtual machines, and virtual switches.



Add a failover cluster to Windows Admin Center

1. Under **All Connections**, select **+ Add**.

2. Select **Server clusters** as the connection type.

3. Enter the name of the cluster. If you're prompted, enter the credentials to use.
4. Choose whether to add the cluster nodes as individual server connections in Windows Admin Center.
5. Select **Add** to finish.

The cluster is added to your list of connections on the **Overview** page. Select it to connect to the cluster.

> **Note**
>
> You can also manage hyper-converged clusters by adding the cluster as a hyper-converged cluster connection in Windows Admin Center.

Tools

The following areas of Windows Admin Center represent tools that are available for failover cluster connections:

Expand table

Area	Description
Overview	View failover cluster details and manage cluster resources.
Disks	View cluster shared disks and volumes.
Networks	View networks in the cluster.
Nodes	View and manage cluster nodes.
Roles	Manage cluster roles or create an empty role.
Updates	Manage Cluster-Aware Updating (requires CredSSP).
Virtual Machines	View and manage virtual machines.
Virtual Switches	View and manage virtual switches.

Feedback

Was this page helpful? Yes No

Provide product feedback

Manage virtual machines by using Windows Admin Center

Article • 02/10/2025 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

The Virtual Machines tool in Windows Admin Center is available in server, failover cluster, or hyper-converged cluster connections if the Hyper-V role is enabled on the server or cluster. You can use the Virtual Machines tool to manage Hyper-V hosts running Windows Server, either installed with Desktop Experience or installed as Server Core. Hyper-V Server 2016 and 2019 are also supported.

Highlights of the Virtual Machines tool in Windows Admin Center include:

- **High-level Hyper-V host resource monitoring**. View overall CPU and memory usage, I/O performance metrics, and virtual machine (VM) health alerts and events for the Hyper-V host server or the entire cluster in a single dashboard.
- **Unified experience that brings together Hyper-V Manager and Failover Cluster Manager capabilities**. View all the virtual machines across a cluster and drill down into a single virtual machine for advanced management and troubleshooting.
- **Simplified, yet powerful workflows for virtual machine management**. Use UI experiences that are tailored to IT administration scenarios to create, manage, and replicate virtual machines.

This article describes some of the Hyper-V tasks that you can do in Windows Admin Center.

Monitor Hyper-V host resources and performance

1. On the left pane, select **Virtual Machines**.

2. Use the **Summary** tab to get a holistic view of Hyper-V host resources and performance for the current server or the entire cluster. The information includes:

    - The number of VMs grouped by state: **Running**, **Off**, **Paused**, and **Saved**.

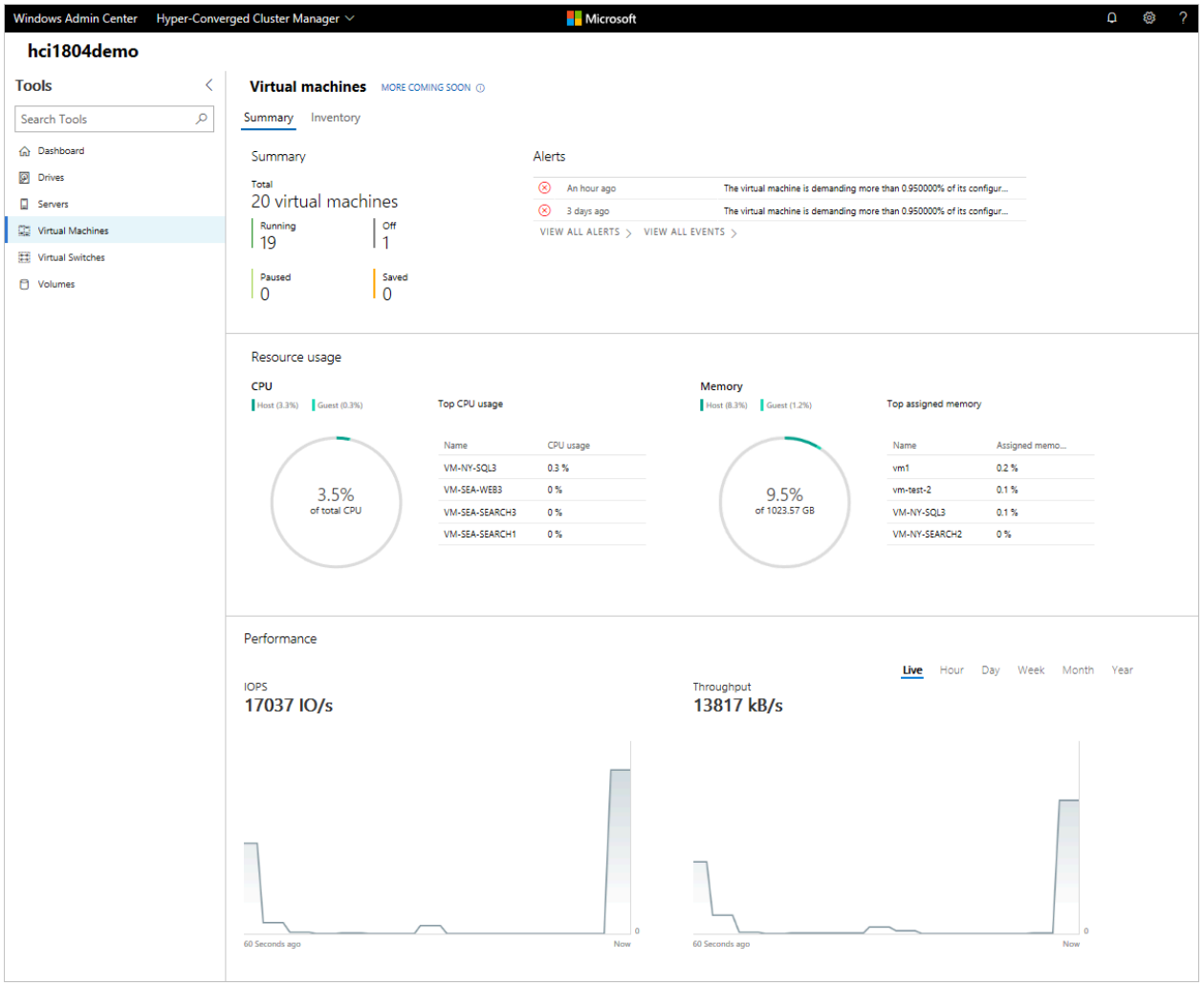
    - Recent health alerts or Hyper-V event log events. Alerts are available only for hyper-converged clusters running Windows Server 2016 or later.

    - CPU and memory usage, with a breakdown of hosts versus guests.

    - VMs that consume the most CPU and memory resources.

- Line charts for input/output per second (IOPS) and I/O throughput, for both live and historical data.

  Line charts for storage performance are available only for hyper-converged clusters running Windows Server 2016 or later. Historical data is available only for hyper-converged clusters running Windows Server 2019.



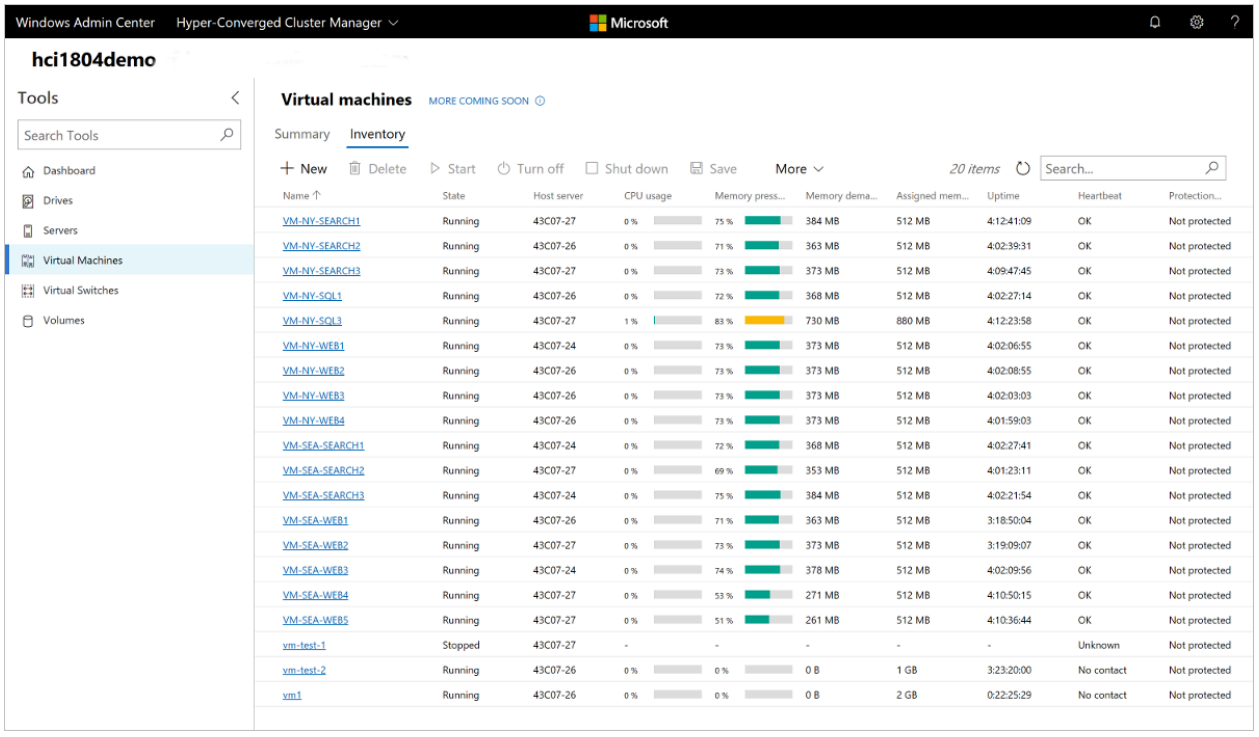
View virtual machine inventory

1. On the left pane, select **Virtual Machines**.

2. Select the **Inventory** tab. On this tab, you can:

   - View a list of the virtual machines running on the current server or cluster.
   - View a virtual machine's state and host server if you're viewing virtual machines for a cluster. Also view CPU and memory usage from the host perspective, including memory pressure, memory demand, and assigned memory.
   - View a virtual machine's uptime, heartbeat status, and protection status by using Azure Site Recovery.

- Create a virtual machine.
- Delete, start, turn off, shut down, pause, resume, reset, or rename a virtual machine. Also save the virtual machine, delete a saved state, or create a checkpoint.
- Change settings for a virtual machine.
- Connect to a virtual machine console by using VMConnect via the Hyper-V host.
- Replicate a virtual machine by using Azure Site Recovery.
- Select multiple VMs and run an operation on all of them at once. This ability applies to operations that you can run on multiple VMs, such as start, shut down, save, pause, delete, and reset.



> **Note**
>
> If you're connected to a cluster, the Virtual Machine tool displays only clustered virtual machines.

Create a virtual machine

1. On the left pane, select **Virtual Machines**.
2. Select the **Inventory** tab.
3. Select **Add** > **New**.

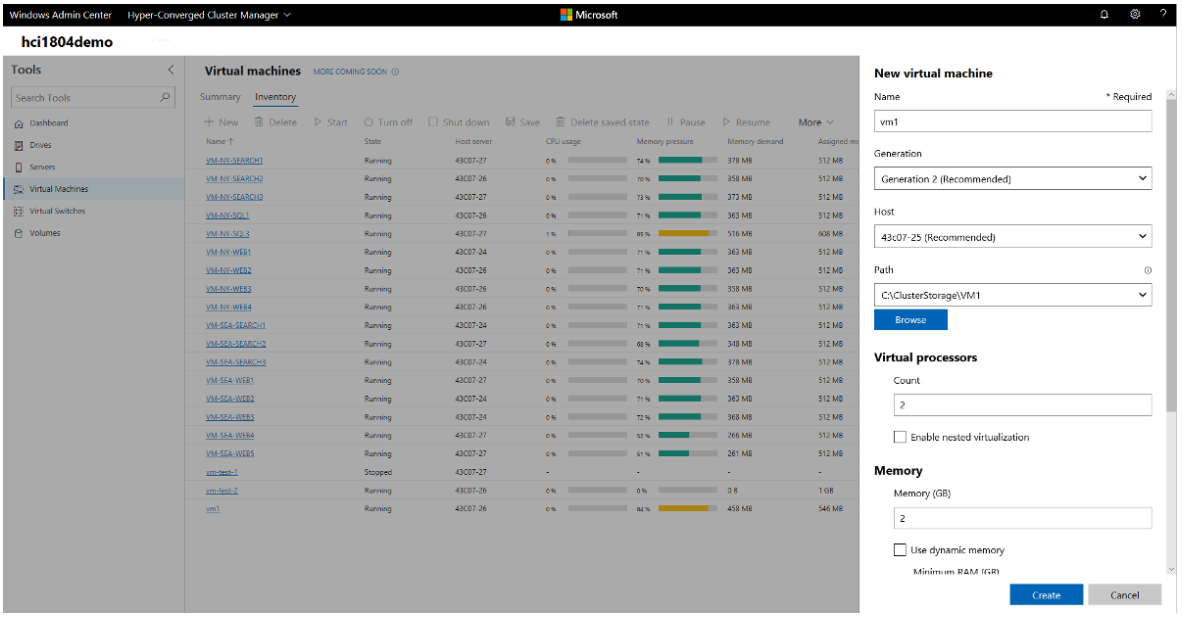
4. Enter the virtual machine name, and choose between generation 1 and generation 2.

5. If you're creating a virtual machine on a cluster, you can choose a host for the virtual machine. If you're running Windows Server 2016 or later, the tool provides a host recommendation for you.

6. Choose a path for the virtual machine files. Select a volume from the dropdown list, or select **Browse** to choose a folder. The virtual machine configuration files and the virtual hard disk (.vhd) file will be saved in a single folder under the `\Hyper-V\\[virtual machine name]` path of the selected volume or path.

   > **Tip**
   >
   > You can browse to any available Server Message Block (SMB) share on the network by entering the path in the **Folder name** field as `\\server\share`. Using a network share for VM storage requires CredSSP.

7. Choose the number of virtual processors and whether you want nested virtualization enabled. Configure memory settings, network adapters, and virtual hard disks. Choose whether you want to install an operating system from an .iso image file or from the network.



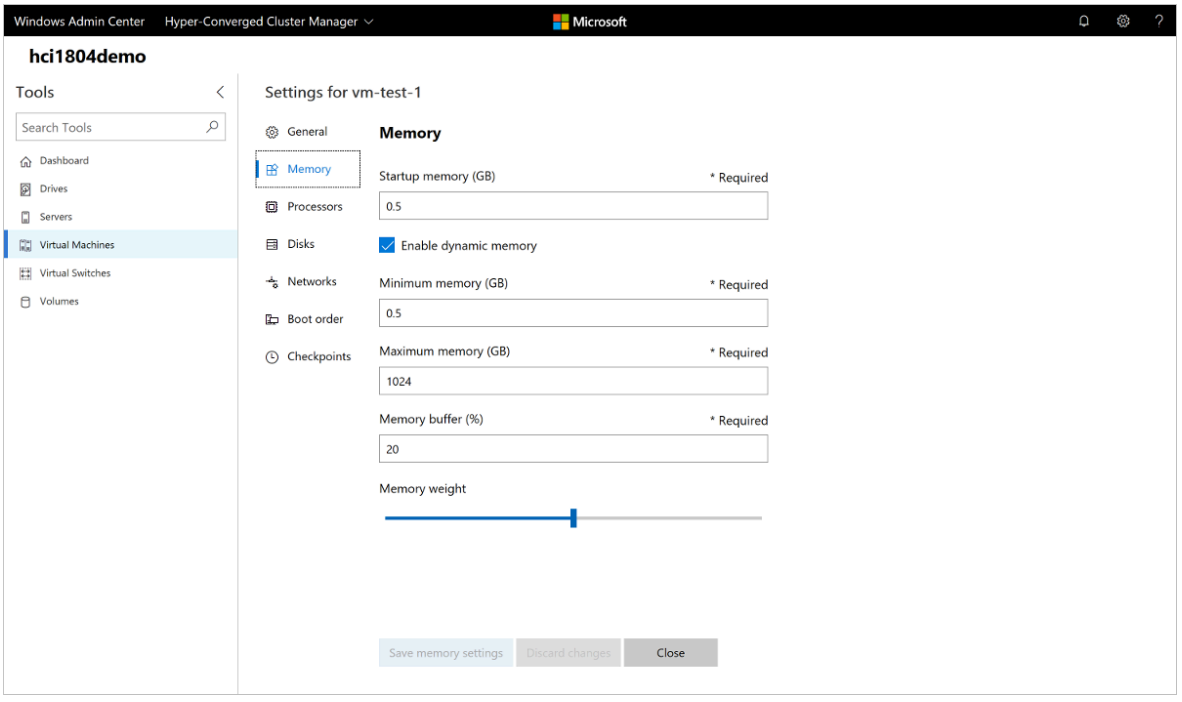
8. Select **Create**.

9. After the virtual machine is created and appears in the list of virtual machines, start it.

10. Connect to the virtual machine's console via VMConnect to install the operating system. Select the virtual machine from the list, and then select **Connect** > **Download RDP file** to download the RDP file. Open the RDP file in the Remote Desktop Connection app. Because you're connecting to the virtual machine's console, you need to enter the Hyper-V host's admin credentials.

Change virtual machine settings

1. On the left pane, select **Virtual Machines**.
2. Select the **Inventory** tab.
3. Choose a virtual machine from the list, and then select **Settings**.
4. For each of the **General**, **Memory**, **Processors**, **Disks**, **Networks**, **Boot order**, and **Checkpoints** tabs, configure the necessary settings, and then select the button that saves the current tab's settings.



The available settings vary, depending on the virtual machine's generation. Also, some settings can't be changed for running virtual machines. For those settings, you need to stop the virtual machine first.

Live migrate a virtual machine to another cluster node

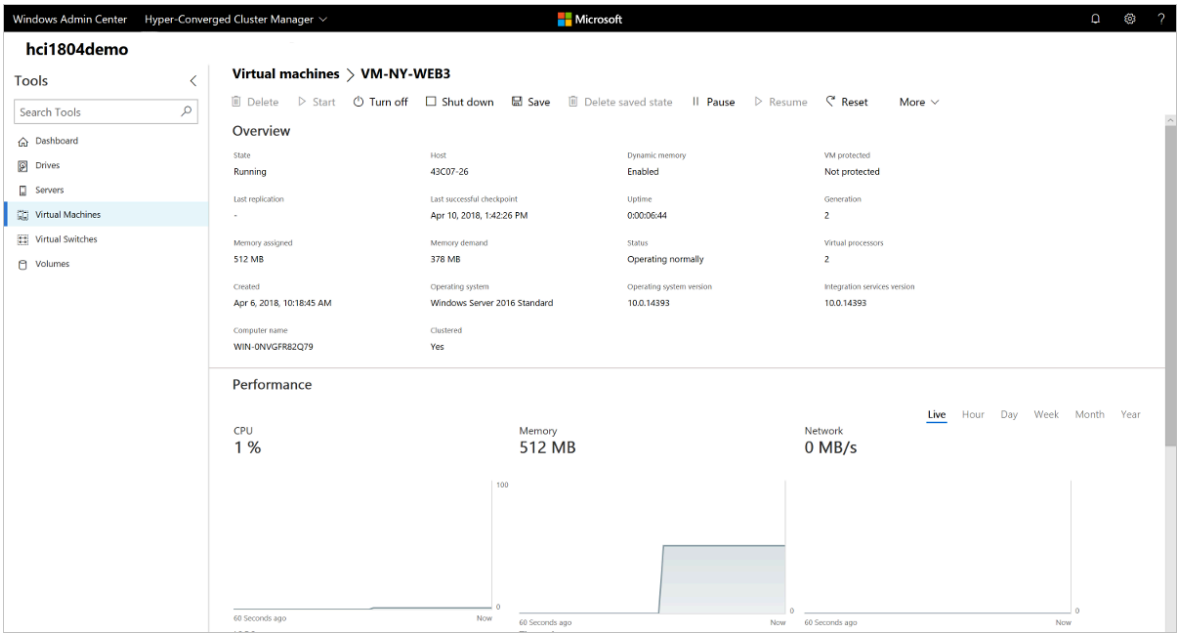
If you're connected to a cluster, you can live migrate a virtual machine to another cluster node:

1. Go to a failover cluster or hyper-converged cluster connection. On the left pane, select **Virtual Machines**.
2. Select the **Inventory** tab.
3. Choose a virtual machine from the list, and then select **Manage** > **Move**.
4. Choose a server from the list of available cluster nodes, and then select **Move**.
5. Notifications for the move progress appear in the upper-right corner of Windows Admin Center. If the move is successful, the host server name changes in the list of virtual machines.

Conduct advanced management and troubleshooting for a single virtual machine

You can view detailed information and performance charts for a single virtual machine.

1. On the left pane, select **Virtual Machines**.
2. Select the **Inventory** tab.
3. Choose a virtual machine from the list. In the pane that opens, you can:
    - View detailed information for the virtual machine.
    - View line charts for IOPS and I/O throughput, for both live and historical data. Historical data is available only for hyper-converged clusters running Windows Server 2019 or later.
    - View, create, apply, rename, and delete checkpoints.
    - View details for the virtual machine's .vhd files, network adapters, and host server.
    - Delete, start, turn off, shut down, pause, resume, reset, or rename the virtual machine. Also save the virtual machine, delete a saved state, or create a checkpoint.
    - Change settings for the virtual machine.
    - Connect to the virtual machine console by using VMConnect via the Hyper-V host.
    - Replicate the virtual machine by using Azure Site Recovery.



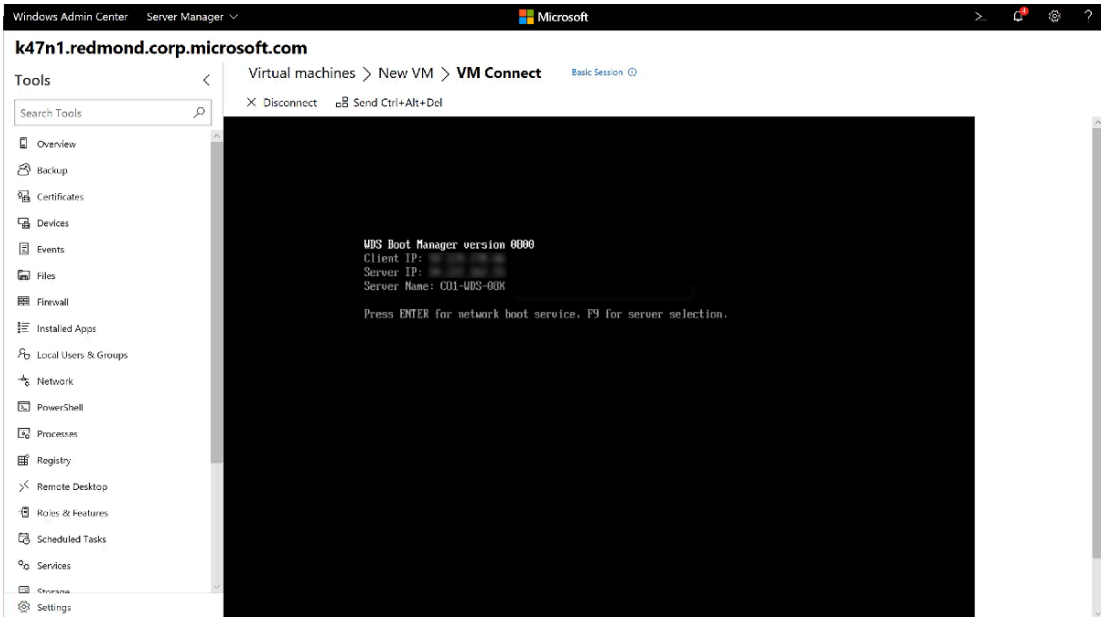
Manage a virtual machine through the Hyper-V host (VMConnect)

1. On the left pane, select **Virtual Machines**.

2. Select the **Inventory** tab.

3. Choose a virtual machine from the list, and then select one of these options:

    - **Connect**. Interact with the guest VM through the Remote Desktop web console that's integrated into Windows Admin Center.



    - **Download RDP file**. Download an RDP file that you can open with the Remote Desktop Connection application (mstsc.exe).

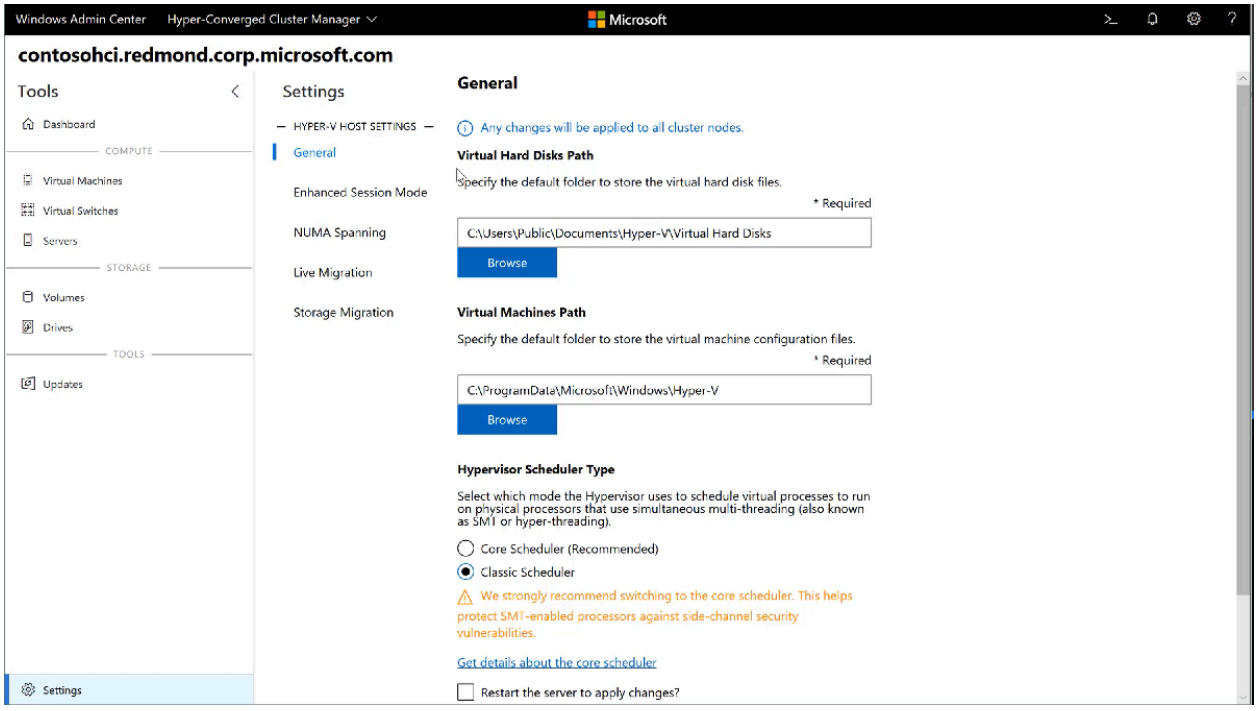
Both options use VMConnect to connect to the guest VM through the Hyper-V host. They both require you to enter administrator credentials for the Hyper-V host server.

Change Hyper-V host settings

1. On a server, hyper-converged cluster, or failover cluster connection, select **Settings** at the bottom of the left pane.

2. On a Hyper-V host server or cluster, the **Hyper-V Host Settings** group has the following sections. Make the necessary changes.

    - **General**. Change virtual hard disks, virtual machine file path, and hypervisor schedule type (if it's supported).
    - **Enhanced Session Mode**
    - **NUMA Spanning**
    - **Live Migration**
    - **Storage Migration**



If you make any Hyper-V host setting changes in a hyper-converged cluster or failover cluster connection, the change will be applied to all cluster nodes.

View Hyper-V event logs

You can view Hyper-V event logs directly from the Virtual Machines tool:

1. On the left pane, select **Virtual Machines**.

2. Select the **Summary** tab.
3. In the **Events** section, select **View all events**.
4. The Event Viewer tool shows the Hyper-V event channels on the left pane. Choose a channel to view the events on the right pane.

   If you're managing a failover cluster or hyper-converged cluster, the event logs display events for all cluster nodes. They display the host server in the **Machine** column.

Replicate virtual machines by using Azure Site Recovery

You can use Windows Admin Center to configure Azure Site Recovery and replicate your on-premises virtual machines to Azure. Learn more.

Feedback

Was this page helpful? Yes No

Provide product feedback

Event logging in Windows Admin Center

Article • 02/10/2025 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Windows Admin Center writes event logs that show the management activities performed on the servers in your environment. Event logs can also help you troubleshoot any Windows Admin Center problems.

Management activities

Windows Admin Center provides insight into the management activities performed on the servers in your environment by logging actions to the **Microsoft-ServerManagementExperience** event channel in the event log of the managed server. `EventID` is `4000`, and `Source` is `SMEGateway`.

Windows Admin Center logs only actions on the managed server. Events aren't logged if a user accesses a server for read-only purposes.

Logged events include the following information:

Expand table

Key	Value
`PowerShell`	PowerShell script name that was run on the server, if the action ran a PowerShell script.
`CIM`	CIM call that was run on the server, if the action ran a CIM call.
`Module`	Tool (or module) where the action was run.
`Gateway`	Name of the Windows Admin Center gateway machine where the action was run.
`UserOnGateway`	Username that's used to access the Windows Admin Center gateway and execute the action.
`UserOnTarget`	Username that's used to access the target managed server, if it's different from `UserOnGateway`. That is, the user accessed the server by using **Manage as** credentials.
`Delegation`	Boolean, if the target managed server trusts the gateway and credentials are delegated from the user's client machine.
`LAPS`	Boolean, if the user accessed the server by using LAPS credentials.

Key	Value
`File`	Name of the uploaded file, if the action was a file upload.

Gateway usage

Windows Admin Center logs gateway activity to the event channel on the gateway computer to help you troubleshoot issues and view metrics on usage. These events are also logged to the **Microsoft-ServerManagementExperience** event channel.

Learn more about troubleshooting Windows Admin Center.

Feedback

Was this page helpful? Yes No

Provide product feedback

Manage Windows Server on Amazon EC2 by using Windows Admin Center

Article • 02/10/2025 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

By using Windows Admin Center, you can manage Windows Server machines running on Amazon EC2 in Amazon Web Services. In this article, you learn how to prepare and connect a Windows Server machine running on Amazon EC2 to Windows Admin Center by using a public IP address.

Prerequisites

- Have a running Amazon EC2 instance.
- Configure WinRM for HTTPS. WinRM HTTPS requires a server authentication certificate for a local computer, with a CN that matches the public name of your Windows Server Amazon EC2 instance. For more information, see How to configure WINRM for HTTPS.
- Enable inbound connections for your Amazon EC2 instance. For more information, see the AWS documentation.
  - For **Type**, select **WinRM-HTTPS**. For the port range, enter **5986**.
  - For **Source**, select **IP Addresses**, and then enter the source IP address that corresponds to your Windows Admin Center gateway.

> **Note**
>
> Alternatively, you can connect to your Amazon EC2 instance via HTTP by selecting **WinRM-HTTP** and entering **5985** as your port range. However, we recommend that you consider the security implications.

Prepare your machine

Configure your Windows Server Amazon EC2 instance by using the following steps:

1. Enable WinRM access to your target Amazon EC2 instance by running the `winrm quickconfig` command in PowerShell on the target EC2 instance.

2. Enable inbound connections to port 5986 for WinRM over HTTPS by running the following PowerShell script on the target EC2 instance: `Set-NetFirewallRule -Name`

`WINRM-HTTPS-In-TCP-PUBLIC -RemoteAddress Any`. Alternatively, you can configure port 5985 for WinRM over HTTP.

Connect to your machine

Add your Windows Server Amazon EC2 instance as a connection in Windows Admin Center:

1. In the Azure portal, open Windows Admin Center.
2. On the **Add or create resource** menu, in the **Servers** section, select **Add**.
3. For **Server name**, enter the public IPv4 address of your Amazon EC2 instance.
4. Select **Use another account for this connection** and enter your Windows credentials.

Related content

Now that you've set up your Amazon EC2 instance with Windows Admin Center, learn how to manage it:

- Manage servers with Windows Admin Center

Feedback

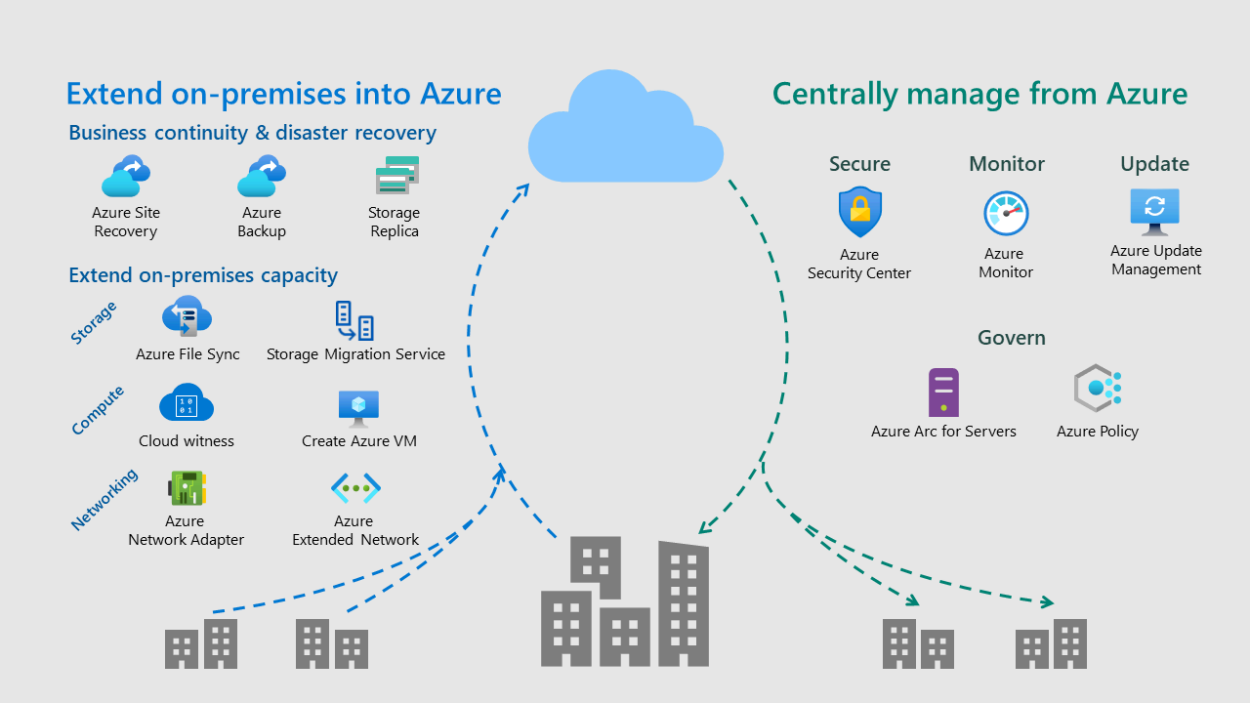
Was this page helpful? Yes No

Provide product feedback

Connecting Windows Server to Azure hybrid services

Article • 10/21/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

You can extend on-premises deployments of Windows Server to the cloud by using Azure hybrid services. These cloud services provide an array of useful functions, both for extending on-premises into Azure, and for centrally managing from Azure.



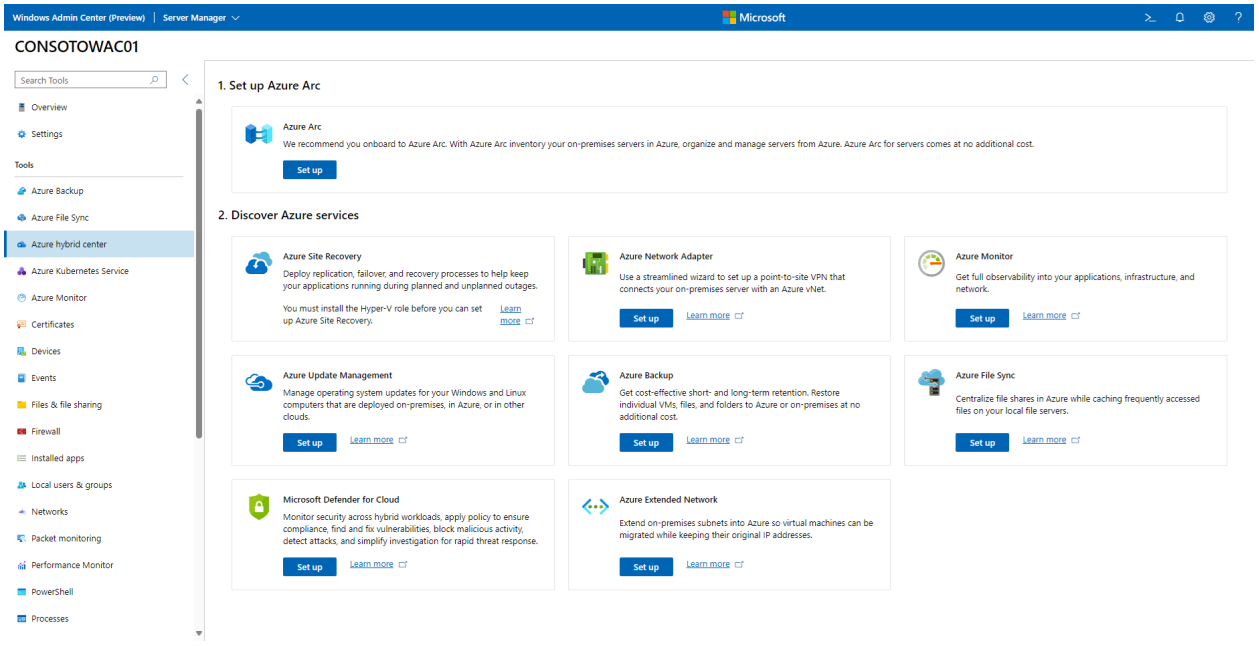
Using Azure hybrid services within Windows Admin Center, you can:

- Protect virtual machines and use cloud-based backup and disaster recovery (HA/DR).
- Extend on-premises capacity with storage and compute in Azure, and simplify network connectivity to Azure.
- Centralize monitoring, governance, configuration, and security across your applications, network, and infrastructure with the help of cloud-intelligent Azure management services.

You can set up most Azure hybrid services by downloading an app and doing some manual configuration. Many Azure hybrid services are integrated directly into Windows Admin Center to provide a simplified setup experience and a server-centric view of the services. Windows Admin Center also provides convenient intelligent hyperlinks to the Azure portal to see connected Azure resources as well as a centralized view of your hybrid environment.

Discover integrated services in the Azure hybrid services tool

The Azure hybrid services tool in Windows Admin Center consolidates all the integrated Azure services into a centralized hub where you can easily discover all the available Azure services that bring value to your on-premises or hybrid environment.



If you connect to a server with Azure services already enabled, the Azure hybrid services tool lets you see all enabled services on that server in a single glance. You can easily get to the relevant tool within Windows Admin Center, launch out to the Azure portal for deeper management of those Azure services, or learn more with documentation at your fingertips.

From the Azure hybrid services tool, you can:

- Back up your Windows Server from Windows Admin Center with Azure Backup
- Protect your Hyper-V Virtual Machines from Windows Admin Center with Azure Site Recovery
- Sync your file server with the cloud, using Azure File Sync
- Manage operating system updates for all your Windows servers, both on-premises or in the cloud, with Azure Update Management
- Monitor servers, both on-premises or in the cloud, and configure alerts with Azure Monitor
- Apply governance policies to your on-premises servers through Azure Policy using Azure Arc for servers
- Secure your servers and get advanced threat protection with Microsoft Defender for Cloud

- Connect your on-premises servers to an Azure Virtual Network with Azure Network Adapter
- Make Azure VMs look like your on-premises network with Azure Extended Network

Azure hybrid service regional support

You can register the Windows Admin Center gateway in both public and private Azure clouds. Today, Microsoft supports gateway registration in Azure Global, Azure China, and Azure US Government. When you register a gateway, Windows Admin Center assumes that all other Azure operations, including the use of Azure hybrid services, will be performed in that cloud.

The regional support for each Azure hybrid service is different. Broadly, the following table shows the clouds supported for each Azure hybrid service in Windows Admin Center:

Expand table

Azure hybrid service	Azure global	Azure China	Azure US Government
Azure Arc	yes	no	yes
Azure Backup	yes	yes	yes
Azure Extended Network	yes	yes	yes
Azure File Sync	yes	yes	yes
Azure Monitor	yes	yes	yes
Azure Network Adapter	yes	yes	yes
Microsoft Defender for Cloud	yes	yes	yes
Azure Site Recovery	yes	yes	no
Azure Update Management	yes	yes	no

For a more detailed regional support breakdown for each service, see Azure products available by region.

Back up and protect your on-premises servers and VMs

- **Back up your Windows servers with Azure Backup** You can back up your Windows servers to Azure. Backing up your Windows servers helps to protect you from accidental or malicious deletions, corruption, and ransomware. For more info, see Back up your servers with Azure Backup.

- **Protect your Hyper-V virtual machines with Azure Site Recovery** You can replicate workloads running on VMs so that your business-critical infrastructure is protected in case of a disaster. Windows Admin Center streamlines setup and the process of replicating your virtual machines on your Hyper-V servers or clusters, making it easier to bolster the resiliency of your environment with Azure Site Recovery's disaster recovery service. For more info, see Protect your VMs with Azure Site Recovery and Windows Admin Center.

- **Use synchronous or asynchronous block-based replication to a VM in Azure using Storage Replica** You can configure block-based or volume-based replication on a server-to-server level using Storage Replica to a secondary server or VM. Windows Admin Center lets you create an Azure VM specifically for your replication target, helping you to right-size and correctly configure storage on a new Azure VM. For more info, see Server-to-server replication with Storage Replica.

Extend on-premises capacity with Azure

Extend storage capacity

- **Sync your file server with the cloud by using Azure File Sync** Sync files on this server with Azure file shares. Keep all your files local or use cloud tiering to free up space and cache only the most frequently used files on the server, tiering cold data to the cloud. You can back up data in the cloud, eliminating the need to worry about on-premises server backup. Additionally, multi-site-sync can keep a set of files in sync across multiple servers. For more info, see Sync your file server with the cloud by using Azure File Sync.

- **Migrate storage to a VM in Azure using Storage Migration Service** Use the step-by-step tool to inventory data on Windows and Linux servers and then transfer the data to a new Azure VM. Windows Admin Center can create a new Azure VM for the job that is right-sized and correctly configured to receive the data from your source server. For more info, see Use Storage Migration Service to migrate a server.

Extend compute capacity

- **Create a new Azure virtual machine without leaving Windows Admin Center** From the *All Connections* page within Windows Admin Center, go to **Add** and select **Create new** under **Azure VM**. You even can domain-join your Azure VM and configure storage from within this step-by-step creation tool.

- **Leverage Azure to achieve quorum on your failover cluster with Cloud Witness** Instead of investing in additional hardware to achieve quorum on a 2-node cluster, you can use an Azure storage account to serve as the cluster witness for your Azure Local cluster or other failover cluster. For more info, see Deploy a Cloud Witness for a Failover Cluster.

Simplify network connectivity between your on-premises and Azure networks

- **Connect your on-premises servers to an Azure Virtual Network with Azure Network Adapter** Let Windows Admin Center simplify setting up a point-to-site VPN from an on-premises server into an Azure virtual network.

- **Make Azure VMs look like your on-premises network with Azure Extended Network** Windows Admin Center can set up a site-to-site VPN and extend your on-premises IP addresses into your Azure vNet to let you more easily migrate workloads into Azure without breaking dependencies on IP addresses.

Centrally manage your hybrid environment from Azure

- **Monitor and get email alerts for all the servers in your environment with Azure Monitor for Virtual Machines** You can use Azure Monitor, also known as Virtual Machines Insights, to monitor server health and events, create email alerts, get a consolidated view of server performance across your environment, and visualize apps, systems, and services connected to a given server. Windows Admin Center also can set up default email alerts for server health performance and cluster health events. For more info, see Connect your servers to Azure Monitor and configure email notifications.

- **Centrally manage operating system updates for all your Windows Servers with Azure Update Management** You can manage updates and patches for multiple servers and VMs from a single place, instead of on a per-server basis. With Azure Update Management, you quickly can assess the status of available updates, schedule installation of required updates, and review deployment results to verify updates that apply successfully. This is possible whether your servers are Azure

VMs, hosted by other cloud providers, or on-premises. For more info, see Use Windows Admin Center to manage operating system updates with Azure Update Management.

- **Improve your security posture and get advanced threat protection with Microsoft Defender for Cloud** Microsoft Defender for Cloud is a unified infrastructure security management system that strengthens the security posture of your data centers, and provides advanced threat protection across your hybrid workloads in the cloud - whether they're in Azure or not - as well as on premises. With Windows Admin Center, you can easily set up and connect your servers to Microsoft Defender for Cloud. For more info, see Integrate Microsoft Defender for Cloud with Windows Admin Center (Preview).

- **Apply policies and ensure compliance across your hybrid environment with Azure Arc for servers and Azure Policy** Inventory, organize, and manage on-premises servers from Azure. You can govern servers using Azure policy, control access using RBAC, and enable additional management services from Azure.

Clusters versus stand-alone servers and VMs

Azure hybrid services work with Windows Servers in the following configurations:

- Stand-alone physical servers and virtual machines (VMs)
- Clusters, including hyper-converged clusters certified by the Azure Local, and Windows Server Software-Defined (WSSD) programs

Services for stand-alone servers and VMs

This is the complete list of Azure services that provide functionality to stand-alone servers and VMs:

- Back up your Windows Server from Windows Admin Center with Azure Backup
- Protect your Hyper-V Virtual Machines from Windows Admin Center with Azure Site Recovery
- Sync your file server with the cloud, using Azure File Sync
- Manage operating system updates for all your Windows servers, both on-premises or in the cloud, with Azure Update Management
- Monitor servers, both on-premises or in the cloud, and configure alerts with Azure Monitor
- Apply governance policies to your on-premises servers through Azure Policy using Azure Arc for servers

- Secure your servers and get advanced threat protection with Microsoft Defender for Cloud
- Connect your on-premises servers to an Azure Virtual Network with Azure Network Adapter
- Make Azure VMs look like your on-premises network with Azure Extended Network

Services for clusters

These are the Azure services that provide functionality to clusters as a whole:

- Monitor a hyper-converged cluster with Azure Monitor
- Protect your VMs with Azure Site Recovery
- Deploy a cluster cloud witness

Other Azure-integrated abilities of Windows Admin Center

- **Add Azure VM connections in Windows Admin Center** You can use Windows Admin Center to manage your Azure VMs as well as on-premises machines. By configuring your Windows Admin Center gateway to connect to your Azure VNet, you can manage virtual machines in Azure using the consistent, simplified tools that Windows Admin Center provides. For more info, see Configure Windows Admin Center to manage VMs in Azure.

- **Add an layer of security to Windows Admin Center by adding Microsoft Entra ID authentication** You can add an additional layer of security to Windows Admin Center by requiring users to authenticate using Microsoft Entra identities to access the gateway. Microsoft Entra authentication also lets you take advantage of Microsoft Entra ID's security features like conditional access and multifactor authentication. For more info, see Configure Microsoft Entra authentication for Windows Admin Center.

- **Manage Azure resources directly through Azure Cloud Shell embedded in Windows Admin Center** Leverage Azure Cloud Shell to get a Bash or PowerShell experience within Windows Admin Center to give you easy access to Azure management tasks. For more info, see Overview of Azure Cloud Shell.

Additional References

- Connect Windows Admin Center to Azure
- Deploy Windows Admin Center in Azure

Feedback

Was this page helpful? Yes No

Provide product feedback

Backup your Windows Servers from Windows Admin Center with Azure Backup

Article • 12/23/2021 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Learn more about Azure integration with Windows Admin Center.

Windows Admin Center streamlines the process of backing up your Windows Servers to Azure and protecting you from accidental or malicious deletions, corruption and even ransomware. To automate setup, you can connect the Windows Admin Center gateway to Azure.

Use the following information to configure Backup for you Windows Server and create a Backup policy to backup your server's Volumes and the Windows System State from the Windows Admin Center.

What is Azure Backup and how does it work with Windows Admin Center?

Azure Backup is the Azure-based service you can use to back up (or protect) and restore your data in the Microsoft cloud. Azure Backup replaces your existing on-premises or off-site backup solution with a cloud-based solution that is reliable, secure, and cost-competitive. Learn more about Azure Backup.

Azure Backup offers multiple components that you download and deploy on the appropriate computer, server, or in the cloud. The component, or agent, that you deploy depends on what you want to protect. All Azure Backup components (no matter whether you're protecting data on-premises or in Azure) can be used to back up data to a Recovery Services vault in Azure.

The integration of Azure Backup in the Windows Admin Center is ideal for backing up volumes and the Windows System state on-premises Windows physical or virtual servers. This makes for a comprehensive mechanism to backup File Servers, Domain Controllers and IIS Web Servers.

Windows Admin Center exposes the Azure Backup integration via the native **Backup** tool. The **Backup** tool provides setup, management and monitoring experiences to quickly start backing up your servers, perform common backup and restore operations and to monitor overall backup health of your Windows Servers.

Prerequisites and planning

- An Azure Account with at least one active subscription
- The target Windows Servers that you want to backup must have Internet access to Azure
- Connect your Windows Admin Center gateway to Azure

To start the workflow to backup your Windows Server, open a server connection, click on the **Backup** tool and follow the steps mentioned below.

Setup Azure Backup

When you click on the **Backup** tool for a server connection on which Azure Backup is not yet enabled, you would see the **Welcome to Azure Backup** screen. Click the **Set up Azure Backup** button. This would open the Azure Backup setup wizard. Follow the steps as listed below in the wizard to back up your server.

If Azure Backup is already configured, clicking on the **Backup** tool will open the **Backup Dashboard**. Refer to the (Management and Monitoring) section to discover operations and tasks that can be performed from the dashboard.

Step 1: Login to Microsoft Azure

Sign into you Azure Account.

> ⓘ **Note**
>
> If you have connected your Windows Admin Center gateway to Azure, you should be automatically logged in to Azure. You can click **sign-out** to further sign-in as a different user.

Step 2: Set up Azure Backup

Select the appropriate settings for Azure Backup as described below

- **Subscription Id:** The Azure subscription you want to use backing up your Windows Server to Azure. All Azure assets like the Azure Resource Group, the Recovery Services Vault will be created in the selected Subscription.
- **Vault:** The Recovery Services Vault where your servers' backups will be stored. You can select from existing vaults or Windows Admin Center will create a new Vault.

- **Resource Group:** The Azure Resource Group is a container for a collection of resources. The Recovery Services vault is created or contained in the specified Resource Group. You can select from existing Resource Groups or Windows Admin Center will create a new one.
- **Location:** The Azure region where the Recovery Services Vault will be created. It is recommended to select the Azure region closest to the Windows Server.

Step 3: Select Backup Items and Schedule

- Select what you want to back up from your server. Windows Admin Center allows you to pick from a combination of **Volumes** and the **Windows System State** while giving you the estimated size of data that is selected for backup.

> ⓘ **Note**
>
> The first backup is a full-backup of all the selected data. However, subsequent backups are incremental in nature and transfer only the changes to the data since the previous backup.

- Select from multiple preset **Backup Schedules** for you System State and/or Volumes.

Step 4: Enter Encryption Passphrase

- Enter an **Encryption Passphrase** of your choice (minimum 16 characters). **Azure Backup** secures your backup data with a user-configured and user-managed encryption passphrase. The encryption passphrase is required to recover data from Azure Backup.

> ⓘ **Note**
>
> The passphrase must be stored in a secure offsite location such as another server or the Azure Key Vault. Microsoft does not store the passphrase and cannot retrieve or reset the passphrase if it is lost or forgotten.

- Review all the settings and click **Apply**

Windows Admin Center will then perform the following operations

1. Create an Azure Resource Group if it does not exist already
2. Create an Azure Recovery Services Vault as specified

3. Install and register the Microsoft Azure Recovery Services Agent to the Vault
4. Create the Backup and Retention schedule as per the selected options and associate them with the Windows Server.

Management and Monitoring

Once you have successfully setup Azure Backup, you would see the **Backup Dashboard** when you open the Backup tool for an existing server connection. You can perform the following tasks from the **Backup Dashboard**

- **Access the Vault in Azure:** You can click on the **Recovery Services Vault** link in the **Overview** tab of the **Backup Dashboard** to be taken to the Vault in Azure to perform a rich set of management operations
- **Perform an ad hoc backup:** Click on **Backup Now** to take an ad hoc backup.
- **Monitor Jobs and Configure alert notifications:** Navigate to the **Jobs** tab of the dashboard to monitor on-going or past jobs and configure alert notifications to receive emails for any failed jobs or other backup related alerts.
- **View Recovery Points and Recover Data:** Click on the **Recovery Points** tab of the dashboard to view the Recovery Points and click on **Recover Data** for steps to recover you data from Azure.

Feedback

Was this page helpful? Yes No

Provide product feedback

Protect VM workloads with Azure Site Recovery on Azure Local (preview)

Article • 11/19/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

> Applies to: Azure Local, version 22H2 and later

This guide describes how to protect Windows and Linux VM workloads running on your Azure Local if there's a disaster. You can use Azure Site Recovery to replicate your on-premises Azure Local virtual machines (VMs) into Azure and protect your business critical workloads.

This feature is enabled on Azure Local running the May 2023 cumulative update of version 22H2 and later.

> **ⓘ Important**
>
> This feature is currently in PREVIEW. See the **Supplemental Terms of Use for Microsoft Azure Previews** for legal terms that apply to Azure features that are in beta, preview, or otherwise not yet released into general availability.

Azure Site Recovery with Azure Local

Azure Site Recovery is an Azure service that replicates workloads running on VMs so that your business-critical infrastructure is protected if there's a disaster. For more information about Azure Site Recovery, see About Site Recovery.

The disaster recovery strategy for Azure Site Recovery consists of the following steps:

- **Replication** - Replication lets you replicate the target VM's VHD to an Azure Storage account and thus protects your VM if there's a disaster.
- **Failover** - Once the VM is replicated, fail over the VM and run it in Azure. You can also perform a test failover without impacting your primary VMs to test the recovery process in Azure.
- **Re-protect** – VMs are replicated back from Azure to the on-premises system.
- **Failback** - You can fail back from Azure to the on-premises system.

In the current implementation of Azure Site Recovery integration with Azure Local, you can start the disaster recovery and prepare the infrastructure from the Azure Local

resource in the Azure portal. After the preparation is complete, you can finish the remaining steps from the Site Recovery resource in the Azure portal.

> **Note**
>
> Azure Site Recovery doesn't support the replication, failover, and failback of the Arc resource bridge and Arc VMs.

Overall workflow

The following diagram illustrates the overall workflow of Azure Site Recovery working with Azure Local.



Here are the main steps that occur when using Site Recovery with an Azure Local:

1. Start with a registered Azure Local on which you enable Azure Site Recovery.
2. Make sure that you meet the prerequisites before you begin.
3. Create the following resources in your Azure Local resource portal:
   a. Recovery services vault
   b. Hyper-V site
   c. Replication policy
4. Once you have created all the resources, prepare infrastructure.
5. Enable VM replication. Complete the remaining steps for replication in the Azure Site Recovery resource portal and begin replication.
6. Once the VMs are replicated, you can fail over the VMs and run on Azure.

Supported scenarios

The following table lists the scenarios that are supported for Azure Site Recovery and Azure Local.

Fail over Azure Local VMs to Azure followed by failback

Expand table

Azure Local VM details	Failover	Failback
Windows Gen 1	Failover to Azure	Failback on same or different host as failover
Windows Gen 2	Failover to Azure	Failback on same or different host as failover

Azure Local VM details	Failover	Failback
Linux Gen 1	Failover to Azure	Failback on same or different host as failover

> **Note**
>
> If an Azure Local VM is deleted after a failover, manual intervention is needed to fail back to the same or a different host..

Prerequisites and planning

Before you begin, make sure to complete the following prerequisites:

- The Hyper-V VMs that you intend to replicate should be made highly available for replication to happen. If VMs aren't highly available, then the replication would fail. For more information, see How to make an existing Hyper-V machine VM highly available.
- Make sure that Hyper-V is set up on the Azure Local.
- The machines hosting the VMs you want to protect must have internet access to replicate to Azure.
- The Azure Local must already be registered.
    - The system must be running May cumulative update for Azure Local, version 22H2.
    - If you're running an earlier build, the Azure portal indicates that the disaster recovery isn't supported as managed identity isn't enabled for older versions.

      Run the repair registration cmdlet to ensure that a managed identity is created for your Azure Local resource and then retry the workflow. For more information, go to Enable enhanced management from Azure for Azure Local.
    - The system must be Arc-enabled. If the system isn't Arc-enabled, you see an error in the Azure portal to the effect that the **Capabilities** tab isn't available.
- You need owner permissions on the Recovery Services Vault to assign permissions to the managed identity. You also need read/write permissions on the Azure Local resource and its child resources.
- Review the caveats associated with the implementation of this feature.
- Review the capacity planning tool to evaluate the requirements for successful replication and failover.

Step 1: Prepare infrastructure on your target host

To prepare the infrastructure, prepare a vault and a Hyper-V site, install the site recovery extension, and associate a replication policy with the machines.

On your Azure Local target system, follow these steps to prepare infrastructure:

1. In the Azure portal, go to the **Overview** pane of the target system resource that is hosting VMs that you want to protect.

2. In the right-pane, go to the **Capabilities** tab and select the **Disaster recovery** tile. As managed identity is enabled on your system, disaster recovery should be available.

   
   Screenshot of Capabilities tab in Azure portal for Azure Local resource.

3. In the right-pane, go to **Protect** and select **Protect VM workloads**.

   
   Screenshot of Protect VM workloads in Azure portal for Azure Local resource.

4. On the **Replicate VMs to Azure**, select **Prepare infrastructure**.

   
   Screenshot of Prepare infrastructure in Azure portal for Azure Local resource.

5. On the **Prepare infrastructure**, select an existing or create a new Recovery services vault. You use this vault to store the configuration information for virtual machine workloads. For more information, see Recovery services vault overview.

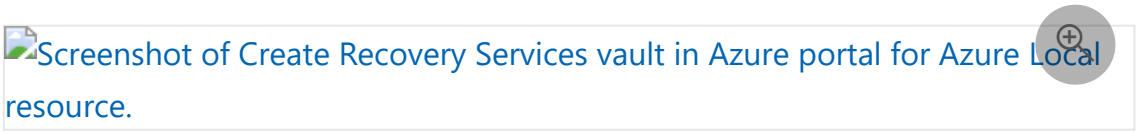
   a. If you choose to create a new Recovery services vault, the subscription and resource groups are automatically populated.

   b. Provide a vault name and select the location of the vault same as where the system is deployed.

   c. Accept the defaults for other settings.

   > **Important**
   >
   > You will need owner permissions on the Recovery services vault to assign permissions to the managed identity. You will need read/write permission on the Azure Local resource and its child resources.

Select **Review + Create** to start the vault creation. For more information, see Create and configure a Recovery services vault.



6. Select an existing **Hyper-V site** or create a new site.



7. Select an existing **Replication policy** or create new. This policy is used to replicate your VM workloads. For more information, see Replication policy. After the policy is created, select **OK**.



8. Select **Prepare infrastructure**. When you select **Prepare infrastructure**, the following actions occur:

   a. A **Resource Group** with the **Storage Account** and the specified **Vault** and the replication policy are created in the specified **Location**.

   b. An Azure Site Recovery agent is automatically downloaded on each node of your system that is hosting the VMs.

   c. Managed Identity gets the vault registration key file from Recovery Services vault that you created and then the key file is used to complete the installation of the Azure Site Recovery agent. A **Resource Group** with the **Storage Account** and the specified **Vault** and the replication policy are created in the specified **Location**.

   d. Replication policy is associated with the specified Hyper-V site and the target system host is registered with the Azure Site Recovery service.

      If you don't have owner level access to the subscription/resource group where you create the vault, you see an error to the effect that you don't have authorization to perform the action.

9. Depending on the number of nodes in your system, the infrastructure preparation could take several minutes. You can watch the progress by going to **Notifications** (the bell icon at the top right of the window).

Step 2: Enable replication of VMs

After the infrastructure preparation is complete, follow these steps to select the VMs to replicate.

1. On **Step 2: Enable replication**, select **Enable replication**. You're now directed to the Recovery services vault where you can specify the VMs to replicate.

   
   Screenshot of Enable replication in Azure portal for an Azure Local resource.

2. Select **Replicate** and in the dropdown select **Hyper-V machines to Azure**.

3. On the **Source environment** tab, specify the source location for your Hyper-V site. In this instance, you have set up the Hyper-V site on your Azure Local resource. Select **Next**.

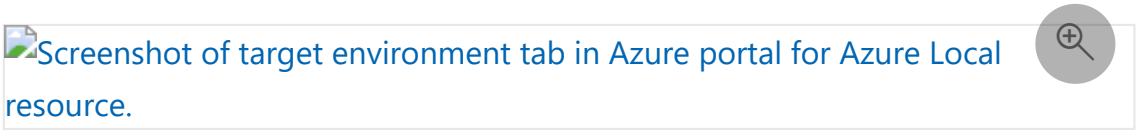
4. On the **Target environment** tab, complete these steps:

   a. For **Subscription**, enter or select the subscription.

   b. For **Post-failover resource group**, select the resource group name to which you fail over. When the failover occurs, the VMs in Azure are created in this resource group.

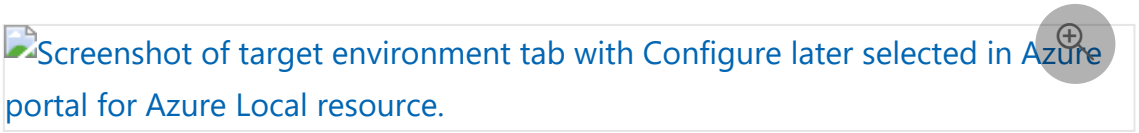
   c. For **Post-failover deployment model**, select **Resource Manager**. The Azure Resource Manager deployment is used when the failover occurs.

   d. For **Storage account**, enter or select an existing storage account associated with the subscription that you have chosen. This account could be a standard or a premium storage account that is used for the VM's replication.

   
   Screenshot of target environment tab in Azure portal for Azure Local resource.

   e. For the network configuration of the VMs that you've selected to replicate in Azure, provide a virtual network and a subnet that would be associated with the VMs in Azure. To create this network, see the instructions in Create an Azure network for failover.

   You can also choose to do the network configuration later.

   
   Screenshot of target environment tab with Configure later selected in Azure portal for Azure Local resource.

   Once the VM is replicated, you can select the replicated VM and go to the **Compute and Network** setting and provide the network information.

5. Select **Next**.

6. On the **Virtual machine selection** tab, select the VMs to replicate, and then select **Next**. Make sure to review the capacity requirements for protecting the VM.

   
   Screenshot of virtual selection tab in Azure portal for Azure Local resource.

7. On the **Replication settings** tab, select the operating system type, operating system disk and the data disks for the VM you intend to replicate to Azure, and then select **Next**.

   
   Screenshot of Replication settings tab in Azure portal for Azure Local resource.

8. On the **Replication policy** tab, verify that the correct replication policy is selected. The selected policy should be the same replication policy that you created when preparing the infrastructure. Select **Next**.

   
   Screenshot of Replication policy tab in Azure portal for Azure Local resource.

9. On the **Review** tab, review your selections, and then select **Enable Replication**.

   
   Screenshot of Review tab in Azure portal for Azure Local resource.

10. A notification indicating that the replication job is in progress is displayed. Go to **Protected items** > **Replication items** to view the status of the replication health and the status of the replication job.

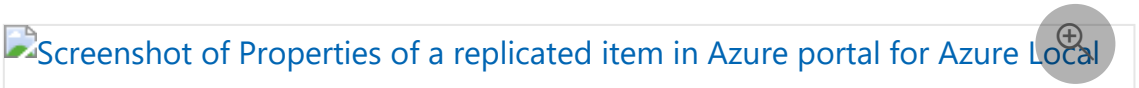
    
    Screenshot of Replicated items in Azure portal for Azure Local resource.

11. To monitor the VM replication, follow these steps.

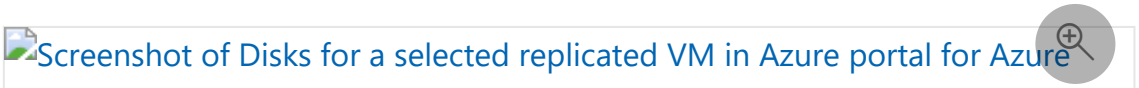
    a. To view the **Replication health** and **Status**, select the VM and go to the Overview. You can see the percentage completion of the replication job.

    
    Screenshot of Overview of a replicated item in Azure portal for Azure Local resource.

    b. To see a more granular job status and **Job id**, select the VM and go to the **Properties** of the replicated VM.

    
    Screenshot of Properties of a replicated item in Azure portal for Azure Local resource.

    c. To view the disk information, go to **Disks**. Once the replication is complete, the **Operating system disk** and **Data disk** should show as **Protected**.

    
    Screenshot of Disks for a selected replicated VM in Azure portal for Azure Local resource.

The next step is to configure a test failover.

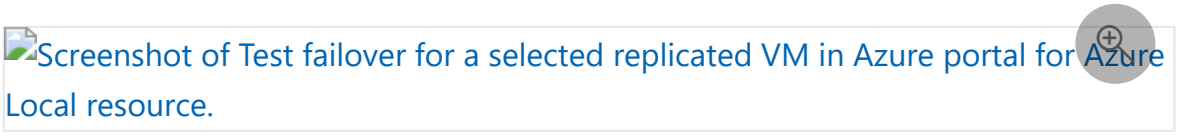
Step 3: Configure and run a test failover in the Azure portal

Once the replication is complete, the VMs are protected. We do recommend that you configure failover settings and run a test failover when you set up Azure Site Recovery.

To prepare for failover to an Azure VM, complete the following steps:

1. If you didn't specify the network configuration for the replicated VM, you can complete that configuration now.
   a. First, make sure that an Azure network is set up to test failover as per the instructions in Create a network for test failover.
   b. Select the VM and go to the **Compute and Network** settings and specify the virtual network and the subnet. The failed-over VM in Azure attaches to this virtual network and subnet.

2. Once the replication is complete and the VM is **Protected** as reflected in the status, you can start **Test Failover**.

   
   Screenshot of Test failover for a selected replicated VM in Azure portal for Azure Local resource.

3. To run a test failover, see the detailed instructions in Run a disaster recovery drill to Azure.

Step 4: Create Recovery Plans

Recovery Plan is a feature in Azure Site Recovery that lets you fail over and recover an entire application comprising a collection of VMs. While it's possible to recover protected VMs individually, by adding the VMs comprising an application to a recovery plan, you're able to fail over the entire application through the recovery plan.

You can also use the test failover feature of Recovery Plan to test the recovery of the application. Recovery Plan lets you group VMs, sequence the order in which they should be brought up during a failover, and automate other steps to be performed as part of the recovery process. Once you've protected your VMs, you can go to the Azure Site Recovery vault in the Azure portal and create recovery plans for these VMs. Learn more about recovery plans.

Step 5: Fail over to Azure

To fail over to Azure, you can follow the instructions in Fail over Hyper-V VMs to Azure.

Caveats

Consider the following information before you use Azure Site Recovery to protect your on-premises VM workloads by replicating those VMs to Azure.

- Extensions installed by Arc aren't visible on the Azure VMs. The Arc server will still show the extensions that are installed, but you can't manage those extensions (for example, install, upgrade, or uninstall) while the machine is in Azure.
- Guest Configuration policies won't run while the machine is in Azure, so any policies that audit the OS security/configuration won't run until the machine is migrated back on-premises.
- Log data (including Sentinel, Defender, and Azure Monitor info) will be associated with the Azure VM while it's in Azure. Historical data is associated with the Arc server. If it's migrated back on-premises, it starts being associated with the Arc server again. They can still find all the logs by searching by computer name as opposed to resource ID, but it's worth noting the Portal UX experiences look for data by resource ID so you'll only see a subset on each resource.
- We strongly recommend that you don't install the Azure VM Guest Agent to avoid conflicts with Arc if there's any potential that the machine will be migrated back on-premises. If you need to install the guest agent, make sure that the VM has extension management disabled. If you try to install/manage extensions using the Azure VM guest agent when there are already extensions installed by Arc on the same machine (or vice versa), you run into all sorts of issues because our agents are unaware of the previous extension installations and will encounter state reconciliation issues.

Known issues

Here's a list of known issues and the associated workarounds in this release:

Expand table

#	Issue	Workaround/Comments
1.	When you register Azure Site Recovery with a system, a machine fails to install	In this instance, your VMs may not be protected. Verify that all machines in the system are registered

#	Issue	Workaround/Comments
	Azure Site Recovery or register to the Azure Site Recovery service.	in the Azure portal by going to the **Recovery Services vault** > **Jobs** > **Site Recovery Jobs**.
2.	Azure Site Recovery agent fails to install. No error details are seen at the system or machine levels in the Azure Local portal.	When the Azure Site Recovery agent installation fails, it is because of the one of the following reasons: - Installation fails as Hyper-V isn't set up on the host. - The Hyper-V host is already associated to a Hyper-V site and you're trying to install the extension with a different Hyper-V site.
3.	Azure Site Recovery agent fails to install. Error message of "Microsoft Azure Site Recovery Provider installation has failed with exit code -1." appears in the portal with the failed installation.	The installation fails when WDAC is enforced. - Setting WDAC to "Audit" mode will allow the installation to complete. To set the WDAC mode to be Audit, you can follow the instructions in Manage WDAC settings with PowerShell

Next steps

- Learn more about Hybrid capabilities with Azure services.

Feedback

Was this page helpful? Yes No

Provide product feedback

Sync your file server with the cloud by using Azure File Sync

Article • 11/04/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Use Azure File Sync to centralize your organization's file shares in Azure, while keeping the flexibility, performance, and compatibility of an on-premises file server. Azure File Sync transforms Windows Server into a quick cache of your Azure file share with the optional cloud tiering feature. You can use any protocol that's available on Windows Server to access your data locally, including SMB, NFS, and FTPS.

Once your files have synced to the cloud, you can connect multiple servers to the same Azure file share to sync and cache the content locally—permissions (ACLs) are always transported as well. Azure Files offers a snapshot capability that can generate differential snapshots of your Azure file share. These snapshots can even be mounted as read-only network drives via SMB for easy browsing and restore. Combined with cloud tiering, running an on-premises file server has never been easier.

Azure File Sync in Windows Admin Center is supported on Windows Server 2012 R2, Windows Server 2016, Windows Server 2019, Windows Server 2022, and Windows Server 2025.

For more info, see Planning for an Azure File Sync deployment.

Feedback

Was this page helpful? Yes No

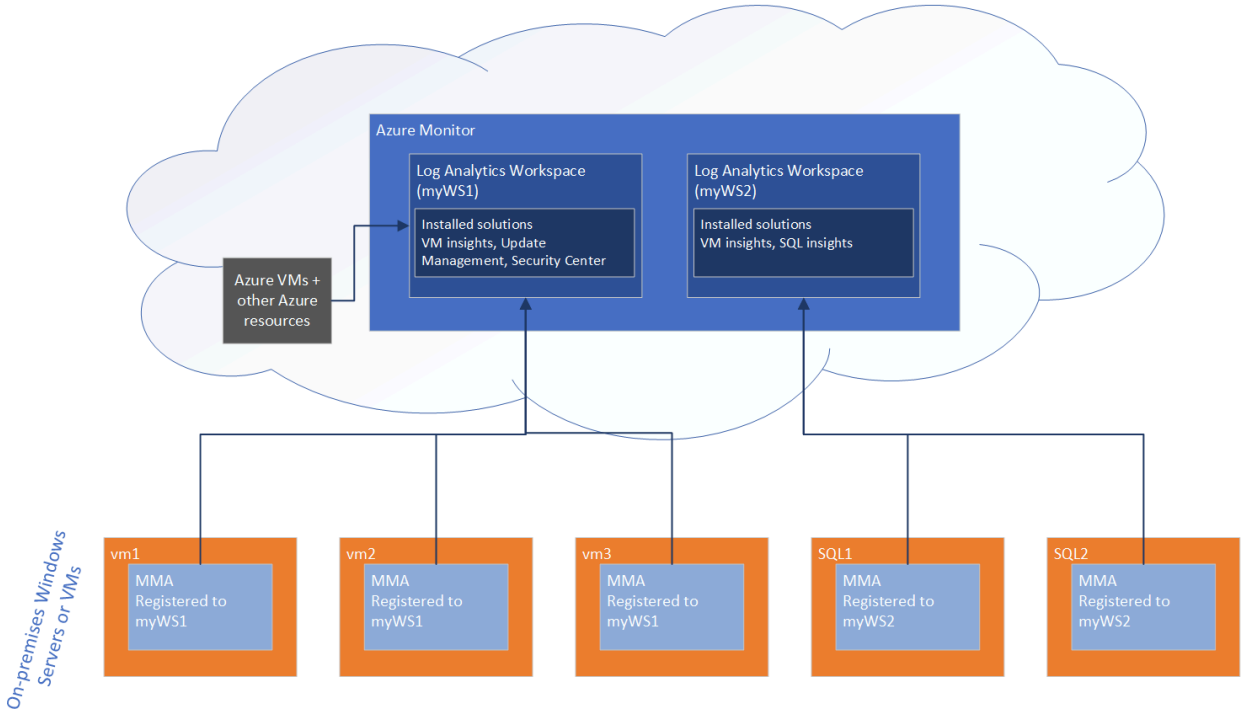
Provide product feedback

Monitor servers and configure alerts with Azure Monitor from Windows Admin Center

Article • 08/11/2023 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Azure Monitor is a solution that collects, analyzes, and acts on telemetry from various resources, including Windows Servers and VMs, both on-premises and in the cloud. Though Azure Monitor pulls data from Azure virtual machines (VMs) and other Azure resources, this article focuses on how Azure Monitor works with on-premises servers and VMs, specifically with Windows Admin Center. If you're interested to learn how you can use Azure Monitor to get email alerts about your hyper-converged cluster, read about using Azure Monitor to send emails for Health Service Faults.

How does Azure Monitor work?



Data generated from on-premises Windows Servers is collected in a Log Analytics workspace in Azure Monitor. Within a workspace, you can enable various monitoring solutions—sets of logic that provide insights for a particular scenario. For example, Azure Update Management, Azure Security Center, and Azure Monitor for VMs are all monitoring solutions that can be enabled within a workspace.

When you enable a monitoring solution in a Log Analytics workspace, all the servers reporting to that workspace starts collecting data relevant to that solution, so that the

solution can generate insights for all the servers in the workspace.

To collect telemetry data on an on-premises server and push it to the Log Analytics workspace, Azure Monitor requires the installation of the Microsoft Monitoring Agent, or the MMA. Certain monitoring solutions also require a secondary agent. For example, Azure Monitor for VMs also needs on the Dependency agent for functionality that this solution provides.

Some solutions, like Azure Update Management, also depend on Azure Automation, which enables you to centrally manage resources across Azure and non-Azure environments. For example, Azure Update Management uses Azure Automation to schedule and orchestrate installation of updates across machines in your environment, centrally, from the Azure portal.

How does Windows Admin Center enable you to use Azure Monitor?

From within Windows Admin Center, you can enable two monitoring solutions:

- Azure Update Management (in the Updates tool)
- Azure Monitor for VMs (in the Server Manager connection page), also known as Virtual Machines insights

You can get started using Azure Monitor from either of these tools. If you've never used Azure Monitor before, Windows Admin Center automatically provisions a Log Analytics workspace (and Azure Automation account, if needed). Windows Admin Center also installs and configures the Microsoft Monitor Agent (MMA) on the target server, and installs the corresponding solution into the workspace.

For instance, if you first go to the Updates tool to set up Azure Update Management, Windows Admin Center will:

1. Install the MMA on the machine
2. Create the Log Analytics workspace and the Azure Automation account (since an Azure Automation account is necessary in this case)
3. Install the Update Management solution in the newly created workspace.

If you want to add another monitoring solution from within Windows Admin Center on the same server, Windows Admin Center installs that solution into the existing workspace to which that server is connected. Windows Admin Center additionally installs any other necessary agents.

If you connect to a different server and have already setup a Log Analytics workspace, you can also install the Microsoft Monitor Agent on the server, connecting it up to an existing workspace. When you connect a server into a workspace, it automatically starts collecting data and reporting to solutions installed in that workspace.

Azure Monitor for virtual machines (also known as Virtual Machine insights)

When you set up Azure Monitor for VMs in the Server Manager connection page, Windows Admin Center enables the Azure Monitor for VMs solution, also known as Virtual Machine insights. This solution allows you to monitor server health and events, create email alerts, get a consolidated view of server performance across your environment, and visualize apps, systems, and services connected to a given server.

> **Note**
>
> Despite its name, VM insights works for physical servers as well as virtual machines.

You can try Azure Monitor using the free 5 GB of data/month/customer allowance. To learn more about log ingestion plans and pricing, see Azure Monitor pricing. The following sections show some of the benefits onboarding servers into Azure Monitor, for example having a consolidated view of systems performance across your environment.

Set up your server for use with Azure Monitor

From the Overview page of a server connection, go to **Tools** > **Azure Monitor**. Within the Azure Monitor page, onboard your server to Azure Monitor by selecting **Register with Azure and sign in**, once complete return to the same page, select **Setup** and follow the prompts. Windows Admin Center takes care of provisioning the Azure Log Analytics workspace, installing the necessary agent, and ensuring the VM insights solution is configured. Once complete, your server sends performance counter data to Azure Monitor, enabling you to view and create email alerts based on this server, from the Azure portal.

Create email alerts

Once you've attached your server to Azure Monitor, you can use the intelligent hyperlinks within the **Tools** > **Azure Monitor**, under Alerts and actions, select **Configure**

monitoring and alerts from the Azure portal to create new alerts. Windows Admin Center automatically enables performance counters to be collected, so you can easily create new alerts by using one of the predefined queries or writing your own.

Consolidated view across multiple servers

If you onboard multiple servers to a single Log Analytics workspace within Azure Monitor, you can get a consolidated view of all these servers from the Virtual Machines Insights solution within Azure Monitor. Only the Performance and Maps tabs of Virtual Machines Insights for Azure Monitor works with on-premises servers – the health tab functions only with Azure VMs. To view the Performance and Maps tabs of Virtual Machines in the Azure portal, go to **Azure Monitor** > **Virtual Machines** (under Insights), and navigate to the **Performance** or **Maps** tabs.

Visualize apps, systems, and services connected to a given server

When Windows Admin Center onboards a server into the VM insights solution within Azure Monitor, it also lights up a capability called Service Map. This capability automatically discovers application components and maps the communication between services so that you can easily visualize connections between servers with great detail from the Azure portal. You can find the service map by going to the **Azure portal** > **Azure Monitor** > **Virtual Machines** (under Insights), and navigating to the **Maps** tab.

> ⓘ **Note**
>
> The visualizations for Virtual Machines Insights for Azure Monitor are offered in 6 public regions currently. For the latest information, check the Azure Monitor for VMs documentation. You must deploy the Log Analytics workspace in one of the supported regions to get the additional benefits provided by the Virtual Machines Insights solution described previously.

Disabling monitoring

To completely disconnect your server from the Log Analytics workspace, uninstall the Microsoft Monitor Agent. With the agent uninstalled it means that this server no longer sends data to the workspace, and all the solutions installed in that workspace no longer collect and process data from that server. However, uninstalling the Microsoft Monitor Agent doesn't affect the workspace itself – all the resources reporting to that workspace

continues to do so. To uninstall the Microsoft Monitoring Agent within Windows Admin Center, connect to the server and then go to **Installed apps**, find the Microsoft Monitor Agent, and then select **Remove**.

If you want to turn off a specific solution within a workspace, you need to remove the monitoring solution from the Azure portal. Removing a monitoring solution means that the insights created by that solution are no longer generated for *any* of the servers reporting to that workspace. For example, uninstalling the Azure Monitor for VMs solution mean you can no longer see insights about VM or server performance from any of the machines connected to my workspace.

Next steps

- Learn more about Azure integration with Windows Admin Center

Feedback

Was this page helpful? Yes No

Provide product feedback

Use Azure Network Adapter to connect a server to an Azure Virtual Network

Article • 11/01/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

A lot of workloads running on-premises and in multi-cloud environments require connections to virtual machines (VMs) running in Microsoft Azure. To connect a server to an Azure Virtual Network, you have several options, including Site-to-Site VPN, Azure Express Route, and Point-to-Site VPN.

Windows Admin Center and Azure Network Adapter provide a one-click experience to connect the server with your virtual network using a Point-to-Site VPN connection. The process automates configuring the virtual network gateway and the on-premises VPN client.

When to use Azure Network Adapter

Azure Network Adapter Point-to-Site VPN connections are useful when you want to connect to your virtual network from a remote location, such as a branch office, store, or other location. You can also use Azure Network Adapter instead of a Site-to-Site VPN when you require only a few servers to connect to a virtual network. Azure Network Adapter connections don't require a VPN device or a public-facing IP address.

Requirements

Using Azure Network Adapter to connect to a virtual network requires the following:

- An Azure account with at least one active subscription.
- An existing virtual network.
- Internet access for the target servers that you want to connect to the Azure virtual network.
- A Windows Admin Center connection to Azure. To learn more, see Configuring Azure integration.
- The latest version of Windows Admin Center. To learn more, see Windows Admin Center.

> **Note**
>
> It's not required to install Windows Admin Center on the server that you want to connect to Azure. However, you can do that in a single server scenario.

Add an Azure Network Adapter to a server

To configure Azure Network Adapter, go to the Network extension for it in Windows Admin Center.

In Windows Admin Center:

1. Navigate to the server hosting the VMs that you want to add to Azure Network Adapter.
2. Under **Tools**, select **Networks**.
3. Select **Add Azure Network Adapter**.
4. On the **Add Azure Network Adapter** pane, enter the following required information, and then select **Create**:

    - **Subscription**
    - **Location**
    - **Virtual Network**
    - **Gateway Subnet** (if doesn't exist)
    - **Gateway SKU** (if doesn't exist)
    - **Client Address Space**

      The client address pool is a range of private IP addresses that you specify. The clients that connect over a Point-to-Site VPN dynamically receive an IP address from this range. Use a private IP address range that does not overlap with the on-premises location that you connect from, or the virtual network that you want to connect to. We recommend using IP addresses that are in the ranges designated for private networks (10.x.x.x, 192.168.x.x, or 172.16.0.0 to 172.31.255.255).

    - **Authentication Certificate**

      Azure uses certificates to authenticate clients connecting to a virtual network over a Point-to-Site VPN connection. The public key information of the root certificate is uploaded to Azure. The root certificate is then considered "trusted" by Azure for a Point-to-Site connection to the virtual network. Client certificates must be generated from the trusted root certificate and installed on the client server. The client certificate is used to authenticate the client when it initiates a connection to the virtual network.

To learn more, see the “Configure authentication type” section of Configure a Point-to-Site VPN connection to a VNet using native Azure certificate authentication: Azure portal.

> **Note**
>
> Network appliances, such as VPN Gateway and Application Gateway that run inside a virtual network, come with additional cost. To learn more, see Virtual Network pricing.

If there is no existing Azure Virtual Network gateway, Windows Admin Center creates one for you. The setup process can take up to 25 minutes. After the Azure Network Adapter is created, you can start to access VMs in the virtual network directly from your server.

If you don’t need the connectivity anymore, under **Networks**, select the Azure Network Adapter that you want to disconnect, from the top menu, select **Disconnect**, and then on the **Disconnect VPN Confirmation** pop-up window, select **Yes**.

Next steps

For more information about Azure Virtual Network, see also:

- Azure Virtual Network frequently asked questions (FAQ)

Feedback

Was this page helpful? Yes No

Provide product feedback

Use Windows Admin Center to manage operating system updates with Azure Update Management

Article • 11/01/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Learn more about Azure integration with Windows Admin Center.

Azure Update Management is a solution in Azure Automation that allows you to manage updates and patches for multiple machines from a single place, rather than on a per-server basis. With Azure Update Management, you can quickly assess the status of available updates, schedule installation of required updates, and review deployment results to verify updates that apply successfully. This is possible whether your machines are Azure virtual machines (VMs), hosted by other cloud providers, or on premises. Learn more about Azure Update Management.

With Windows Admin Center, you can easily set up and use Azure Update Management to keep your managed servers up to date. If you don't already have a Log Analytics workspace in your Azure subscription, Windows Admin Center will automatically configure your server and create the necessary Azure resources in the subscription and location you specify. If you have an existing Log Analytics workspace, Windows Admin Center can automatically configure your server to consume updates from Azure Update Management.

To get started, go to the Updates tool in a server connection and select “Set up now”, and provide your preferences for the related Azure resources.

Once you've configured your server to be managed by Azure Update Management, you can access Azure Update Management by using the hyperlink provided in the Updates tool.

Learn how to stop using Azure Update Management to update your server.

Note that you must register your Windows Admin Center gateway with Azure before setting up Azure Update Management.

Feedback

Was this page helpful? Yes No

Provide product feedback

Configuring Azure integration

Article • 11/13/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Windows Admin Center supports several optional features that integrate with Azure services. Learn about the Azure integration options available with Windows Admin Center.

To allow the Windows Admin Center gateway to communicate with Azure to leverage Microsoft Entra authentication for gateway access, or to create Azure resources on your behalf (for example, to protect VMs managed in Windows Admin Center using Azure Site Recovery), you need to first register your Windows Admin Center gateway with Azure. You only need to do this action once for your Windows Admin Center gateway - the setting is preserved when you update your gateway to a newer version.

Register your gateway with Azure

The first time you try to use an Azure integration feature in Windows Admin Center, you're prompted to register the gateway to Azure. You can also register the gateway by going to the **Azure** tab in Windows Admin Center Settings. Only Windows Admin Center gateway administrators can register the Windows Admin Center gateway with Azure. Learn more about Windows Admin Center user and administrator permissions.

The guided in-product steps create a Microsoft Entra app in your directory, which allows Windows Admin Center to communicate with Azure. To view the Microsoft Entra app that is automatically created, go to the **Azure** tab of Windows Admin Center settings. The **View in Azure** hyperlink lets you view the Microsoft Entra app in the Azure portal.

The Microsoft Entra app created is used for all points of Azure integration in Windows Admin Center, including Microsoft Entra authentication to the gateway. Windows Admin Center automatically configures the permissions needed to create and manage Azure resources on your behalf:

- Microsoft Graph
  - Application.Read.All
  - Application.ReadWrite.All
  - Directory.AccessAsUser.All
  - Directory.Read.All
  - Directory.ReadWrite.All
  - User.Read
- Azure Service Management
  - user_impersonation

Manual Microsoft Entra app configuration

If you wish to configure a Microsoft Entra app manually, rather than using the Microsoft Entra app created automatically by Windows Admin Center during the gateway registration process, follow these steps:

1. Grant the Microsoft Entra app the required API permissions listed above. You can do so by navigating to your Microsoft Entra app in the Azure portal. Go to the Azure portal > **Microsoft Entra ID** > **App registrations** > select your Microsoft Entra app you wish to use. Then to the **API permissions** tab and add the API permissions listed above.

2. Add the Windows Admin Center gateway URL to the reply URLs (also known as the redirect URIs). Navigate to your Microsoft Entra app, then go to **Manifest**. Find the "replyUrlsWithType" key in the manifest. Within the key, add an object containing two keys: "url" and "type." The key "url" should have a value of the Windows Admin Center gateway URL, appending a wildcard at the end. The key "type" key should have a value of "Web." For example:

   JSON

   ```
   "replyUrlsWithType": [
           {
                   "url": "http://localhost:6516/*",
                   "type": "Single-Page Application"
           }
   ],
   ```

> **Note**
>
> If you have Microsoft Defender Application Guard enabled for your browser, you won't be able to register Windows Admin Center with Azure or sign into Azure.

Troubleshooting Azure sign-in errors

The redirect URI doesn't match the URIs configured for this application

If you've recently migrated your data from an older version of Windows Admin Center to Windows Admin Center version 2410, your redirect URIs may be misconfigured. This can happen if you didn't complete the Azure Registration step in the migration wizard.

This misconfiguration is because Windows Admin Center changed the way we perform authentication based on general Microsoft guidance. Where we previously used the implicit grant flow, we're now using the authorization code flow.

There are two redirect URIs that must be added to the Single-Page Application (SPA) platform. An example of these redirect URIs would be:

```
https://myMachineName.domain.com:6600
https://myMachineName.domain.com:6600/signin-oidc
```

In this example, the numerical value refers to the port referenced in your Windows Admin Center installation.

All redirect URIs for Windows Admin Center must contain:

- The Fully Qualified Domain Name (FQDN) or hostname of your gateway machine, no mention of localhost
- The HTTPS prefix, not HTTP

Learn how to reconfigure your redirect URIs.

After adding the proper redirect URIs, it's good practice to clean up old, unused redirect URIs.

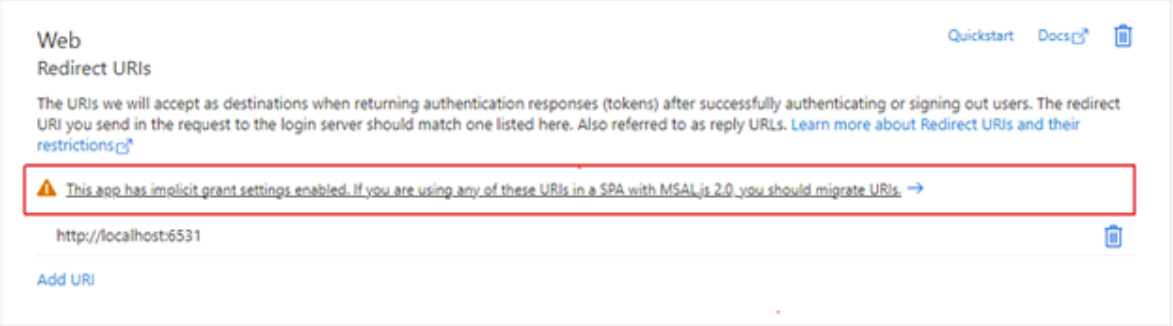
Cross-origin token redemption permitted only for Single-page application

If you've recently updated your Windows Admin Center instance to a newer version, and your gateway was previously registered with Azure, you might encounter an error stating "cross-origin token redemption is permitted only for the 'Single-Page Application' client type" upon signing into Azure. This appears because Windows Admin Center changed the way we perform authentication based on general Microsoft guidance. Where we previously used the implicit grant flow, we're now using the authorization code flow.

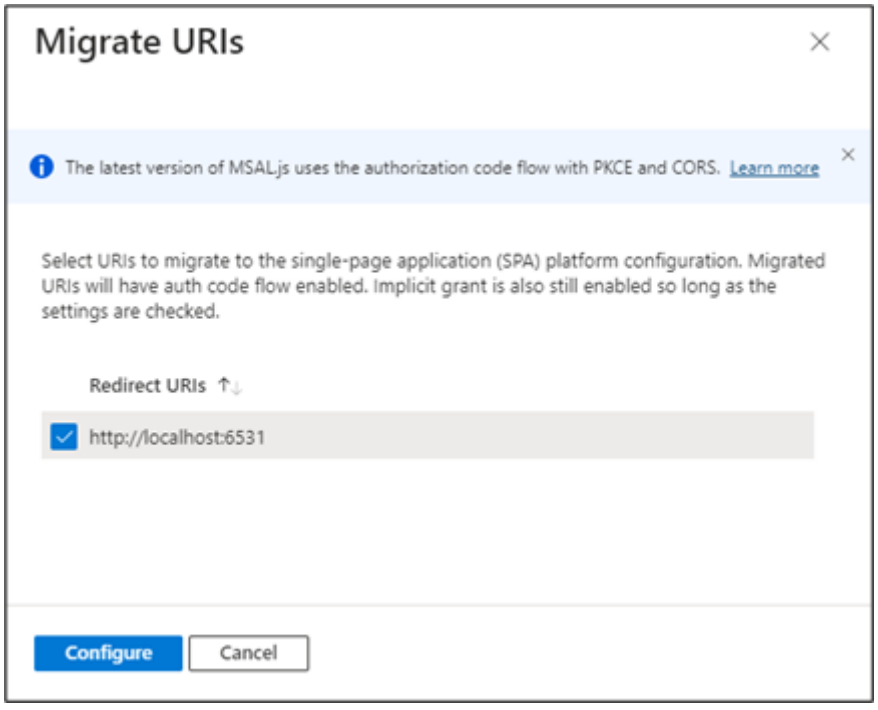
If you'd like to continue using your existing app registration for your Windows Admin Center application, use Microsoft Entra admin center to update the registration's redirect URIs to the Single-Page Application (SPA) platform. Doing so enables the authorization code flow with Proof Key for Code Exchange (PKCE) and cross-origin resource sharing (CORS) support for applications that use that registration.

Follow these steps for application registrations that are currently configured with **Web** platform redirect URIs:

1. Sign in to the Microsoft Entra admin center.
2. Navigate to **Identity** > **Applications** > **App registrations**, select your application, and then **Authentication**.
3. In the **Web** platform tile under **Redirect URIs**, select the warning banner indicating that you should migrate your URIs.



4. Select the redirect URI for your application and then select **Configure**. These redirect URIs should now appear in the **Single-page application** platform tile, showing that CORS support with the authorization code flow and PKCE is enabled for these URIs.



Instead of updating existing URIs, you can instead create a new application registration for your gateway. App registrations that are newly created for Windows Admin Center through the gateway registration flow create Single-Page Application platform redirect URIs.

If you can't migrate your application registration's redirect URIs to use auth code flow, you can continue to use the existing application registration as is. To do so, you must unregister your Windows Admin Center gateway and re-register with the same application registration ID.

Stay updated

Follow us on Twitter

Read our Blogs

Feedback

Was this page helpful? Yes No

Provide product feedback

Deploy Azure virtual machines from within Windows Admin Center

Article • 12/23/2021 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Windows Admin Center version 1910 allows you to deploy Azure virtual machines. This integrates VM deployment into Windows Admin Center-managed workloads like Storage Migration Service and Storage Replica. Instead of building new servers and VMs in the Azure Portal by hand prior to deploying your workload - and possibly missing required steps and configuration - Windows Admin Center can deploy the Azure VM, configure its storage, join it to your domain, install roles, and then set up your distributed system. You can also deploy new Azure VMs without a workload from the Windows Admin Center Connections page.

Windows Admin Center also manages a variety of Azure services. Learn more about the Azure integration options available with Windows Admin Center.

If you want to lift and shift virtual machines to Azure instead of creating new ones, consider using Azure Migrate. For more info, see Azure Migrate overview.

Scenarios

Windows Admin Center version 1910 Azure VM deployment supports the following scenarios:

- Storage Migration Service
- Storage Replica
- New standalone server (without roles)

Requirements

Creating a new Azure VM from within Windows Admin Center requires that you have:

- An Azure subscription.
- A Windows Admin Center gateway registered with Azure
- An existing Azure resource group where you have Create permissions.
- An existing Azure Virtual Network and subnet.
- An Azure ExpressRoute circuit or Azure VPN solution tied to the virtual network and subnet that allows connectivity from Azure VMs to your on-premises clients, domain controllers, the Windows Admin Center computer, and any servers requiring communication with this VM as part of a workload deployment. For

instance, to use the Storage Migration Service to migrate storage to an Azure VM, the orchestrator computer and the source computer must both be able to contact the destination Azure VM you are migrating to.

Usage

Azure VM deployment steps and wizards vary by scenario. Review the workload's documentation for detailed information about the overall scenario.

Deploying Azure VMs as part of Storage Migration Service

1. From the *Storage Migration Service* tool within Windows Admin Center, perform an inventory of one or more source servers.
2. Once you're in the *Transfer Data* phase, select **Create a new Azure VM** on the *Specify a destination* page then click **Create VM**.

   This begins a step-by-step creation tool that selects a Windows Server 2012 R2, Windows Server 2016, or Windows Server 2019 Azure VM as a destination for the migration. Storage Migration Service provides recommended VM sizes to match your source, but you can override them by clicking **See all sizes**.

   Source server data is also used to automatically configure your managed disks and their file systems as well as join your new Azure VM to your Active Directory domain. If the VM is Windows Server 2019 (which we recommend), Windows Admin Center installs the Storage Migration Service proxy feature. Once it has created the Azure VM, Windows Admin Center returns to the normal Storage Migration Service transfer workflow.

Here's a video showing how to use Storage Migration Service to migrate to Azure VMs.
https://www.youtube-nocookie.com/embed/k8Z9LuVL0xQ

Deploying Azure VMs as part of Storage Replica

1. From the *Storage Replica* tool within Windows Admin Center, under the *Partnerships* tab, select **New** and then under *Replicate with another server* select **Use a New Azure VM** then select **Next**.
2. Specify your source server information and replication group name, and then select **Next**.

This begins a process that automatically selects a Windows Server 2016 or Windows Server 2019 Azure VM as a destination for the migration source. Storage Migration Service recommends VM sizes to match your source, but you can override this by selecting **See all sizes**. Inventory data is used to automatically configure your managed disks and their file systems, as well as join your new Azure VM to your Active Directory domain.

3. After Windows Admin Center creates the Azure VM, provide a replication group name and then select **Create**. Windows Admin Center then begins the normal Storage Replica initial synchronization process to start protecting your data.

Here's a video showing how to use Storage Replica to replicate to Azure VMs.
https://www.youtube-nocookie.com/embed/_VqD7HjTewQ

Deploying a new standalone Azure VM

1. From the *All Connections* page within Windows Admin Center, select **Add**.
2. In the *Azure VM* section, select **Create new**.

   This begins a step-by-step creation tool that will let you select a Windows Server 2012 R2, Windows Server 2016, or Windows Server 2019 Azure VM, pick a size, add managed disks, and optionally join your Active Directory domain.

Here's a video showing how to use Windows Admin Center to create Azure VMs.
https://www.youtube-nocookie.com/embed/__A8J9aC_Jk

Feedback

Was this page helpful? Yes No

Provide product feedback

Manage Azure virtual machines with Windows Admin Center

Article • 05/27/2022 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

This article describes how to use Windows Admin Center running on your on-premises PC or server to manage the operating system on one or more Azure virtual machines (in addition to on-premises servers, PCs, and VMs).

To instead use Windows Admin Center directly in the Azure portal to manage the operating system on a single Azure VM, see Use Windows Admin Center in Azure. You can also use scripts to set up a Windows Admin Center server in an Azure VM.

Connecting to VMs with a public IP

If your target VMs (the VMs you want to manage with Windows Admin Center) have public IPs, add them to your Windows Admin Center gateway by IP address, or by fully qualified domain name (FQDN). There are a couple considerations to take into account:

- You must enable WinRM access to your target VM by running the following in PowerShell or the Command Prompt on the target VM: `winrm quickconfig`
- If you haven't domain-joined the Azure VM, the VM behaves like a server in workgroup, so you'll need to make sure you account for using Windows Admin Center in a workgroup.
- You must also enable inbound connections to port 5985 for WinRM over HTTP in order for Windows Admin Center to manage the target VM:

    1. Run the following PowerShell script on the target VM to enable inbound connections to port 5985 on the guest OS: `Set-NetFirewallRule -Name WINRM-HTTP-In-TCP-PUBLIC -RemoteAddress Any`

    2. You must also open the port in Azure networking:
        - Select your Azure VM, select **Network settings**, then **Create port rule**, and finally **inbound port rule**.
        - In the **Destination ranges** field, enter **5985**.

        If your Windows Admin Center gateway has a static IP, you can select to allow only inbound WinRM access from your Windows Admin Center gateway for added security. To do this, select **Advanced** at the top of the **Add inbound security rule** pane.

For **Source**, select **IP Addresses**, then enter the Source IP address corresponding to your Windows Admin Center gateway.
- For **Protocol** select **TCP**.
- The rest can be left as default.

> **Note**
>
> You must create a custom port rule. The WinRM port rule provided by Azure networking uses port 5986 (over HTTPS) instead of 5985 (over HTTP).

Connecting to VMs without a public IP

If your target Azure VMs don't have public IPs, and you want to manage these VMs from a Windows Admin Center gateway deployed in your on-premises network, you need to configure your on-premises network to have connectivity to the VNet on which the target VMs are connected. There are 3 ways you can do this: ExpressRoute, Site-to-Site VPN, or Point-to-Site VPN. Learn which connectivity option makes sense in your environment.

> **Tip**
>
> If you wish to use a Point-to-Site VPN to connect your Windows Admin Center gateway to an Azure VNet to manage Azure VMs in that VNet, you can use the Azure Network Adapter feature in Windows Admin Center. To do so, connect to the server on which Windows Admin Center is installed, navigate to the Network tool and select "Add Azure Network Adapter". When you provide the necessary details and click "Set up", Windows Admin Center will configure a Point-to-Site VPN to the Azure VNet you specify, after which, you can connect to and manage Azure VMs from your on-premises Windows Admin Center gateway.

Ensure WinRM is running on your target VMs by running the following in PowerShell or the Command Prompt on the target VM: `winrm quickconfig`

If you haven't domain-joined the Azure VM, the VM behaves like a server in workgroup, so you'll need to make sure you account for using Windows Admin Center in a workgroup.

If you run into any issues, consult Troubleshoot Windows Admin Center to see if additional steps are required for configuration (for example, if you are connecting using a local administrator account or are not domain-joined).

Feedback

Was this page helpful? Yes No

Provide product feedback

Manually deploy Windows Admin Center in Azure for managing multiple servers

Article • 05/27/2022 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

This article describes how to manually deploy Windows Admin Center in an Azure VM for use in managing multiple Azure VMs. To manage a single VM, instead use the Windows Admin Center functionality built into the Azure portal, as described in Use Windows Admin Center in the Azure portal).

Deploy using script

You can download Deploy-WACAzVM.ps1 which you will run from Azure Cloud Shell to set up a Windows Admin Center gateway in Azure. This script can create the entire environment, including the resource group.

Jump to manual deployment steps

Prerequisites

- Set up your account in Azure Cloud Shell. If this is your first time using Cloud Shell, you will be asked you to associate or create an Azure storage account with Cloud Shell.
- In a **PowerShell** Cloud Shell, navigate to your home directory: `PS Azure:\> cd ~`
- To upload the `Deploy-WACAzVM.ps1` file, drag and drop it from your local machine to anywhere on the Cloud Shell window.

If specifying your own certificate:

- Upload the certificate to Azure Key Vault. First, create a key vault in Azure portal, then upload the certificate into the key vault. Alternatively, you can use Azure portal to generate a certificate for you.

Script parameters

- **ResourceGroupName** - [String] Specifies the name of the resource group where the VM will be created.
- **Name** - [String] Specifies the name of the VM.

- **Credential** - [PSCredential] Specifies the credentials for the VM.
- **MsiPath** - [String] Specifies the local path of the Windows Admin Center MSI when deploying Windows Admin Center on an existing VM. Defaults to the version from https://aka.ms/WACDownload if omitted.
- **VaultName** - [String] Specifies the name of the key vault that contains the certificate.
- **CertName** - [String] Specifies the name of the certificate to be used for MSI installation.
- **GenerateSslCert** - [Switch] True if the MSI should generate a self signed ssl certificate.
- **PortNumber** - [int] Specifies the ssl port number for the Windows Admin Center service. Defaults to 443 if omitted.
- **OpenPorts** - [int[]] Specifies the open ports for the VM.
- **Location** - [String] Specifies the location of the VM.
- **Size** - [String] Specifies the size of the VM. Defaults to "Standard_DS1_v2" if omitted.
- **Image** - [String] Specifies the image of the VM. Defaults to "Win2016Datacenter" if omitted.
- **VirtualNetworkName** - [String] Specifies the name of the virtual network for the VM.
- **SubnetName** - [String] Specifies the name of the subnet for the VM.
- **SecurityGroupName** - [String] Specifies the name of the security group for the VM.
- **PublicIpAddressName** - [String] Specifies the name of the public IP address for the VM.
- **InstallWACOnly** - [Switch] Set to True if WAC should be installed on a pre-existing Azure VM.

There are 2 different options for the MSI to deploy and the certificate used for MSI installation. The MSI can either be downloaded from aka.ms/WACDownload or, if deploying to an existing VM, the filepath of an MSI locally on the VM can be given. The certificate can be found in either Azure Key Vault or a self-signed certificate will be generated by the MSI.

Script Examples

First, define common variables needed for the parameters of the script.

PowerShell

```
$ResourceGroupName = "wac-rg1"
$VirtualNetworkName = "wac-vnet"
$SecurityGroupName = "wac-nsg"
$SubnetName = "wac-subnet"
$VaultName = "wac-key-vault"
$CertName = "wac-cert"
$Location = "westus"
$PublicIpAddressName = "wac-public-ip"
$Size = "Standard_D4s_v3"
$Image = "Win2016Datacenter"
$Credential = Get-Credential
```

Example 1: Use the script to deploy WAC gateway on a new VM in a new virtual network and resource group. Use the MSI from aka.ms/WACDownload and a self-signed cert from the MSI.

PowerShell

```
$scriptParams = @{
    ResourceGroupName = $ResourceGroupName
    Name = "wac-vm1"
    Credential = $Credential
    VirtualNetworkName = $VirtualNetworkName
    SubnetName = $SubnetName
    GenerateSslCert = $true
}
./Deploy-WACAzVM.ps1 @scriptParams
```

Example 2: Same as #1, but using a certificate from Azure Key Vault.

PowerShell

```
$scriptParams = @{
    ResourceGroupName = $ResourceGroupName
    Name = "wac-vm2"
    Credential = $Credential
    VirtualNetworkName = $VirtualNetworkName
    SubnetName = $SubnetName
    VaultName = $VaultName
    CertName = $CertName
```

```
}
./Deploy-WACAzVM.ps1 @scriptParams
```

Example 3: Using a local MSI on an existing VM to deploy WAC.

PowerShell

```
$MsiPath = "C:\Users\<username>\Downloads\WindowsAdminCenter<version>.msi"
$scriptParams = @{
    ResourceGroupName = $ResourceGroupName
    Name = "wac-vm3"
    Credential = $Credential
    MsiPath = $MsiPath
    InstallWACOnly = $true
    GenerateSslCert = $true
}
./Deploy-WACAzVM.ps1 @scriptParams
```

Requirements for VM running the Windows Admin Center gateway

Port 443 (HTTPS) must be open. Using the same variables defined for script, you can use the code below in Azure Cloud Shell to update the network security group:

PowerShell

```
$nsg = Get-AzNetworkSecurityGroup -Name $SecurityGroupName -
ResourceGroupName $ResourceGroupName
$newNSG = Add-AzNetworkSecurityRuleConfig -NetworkSecurityGroup $nsg -Name
ssl-rule -Description "Allow SSL" -Access Allow -Protocol Tcp -Direction
Inbound -Priority 100 -SourceAddressPrefix Internet -SourcePortRange * -
DestinationAddressPrefix * -DestinationPortRange 443
Set-AzNetworkSecurityGroup -NetworkSecurityGroup $newNSG
```

Requirements for managed Azure VM's

Port 5985 (WinRM over HTTP) must be open and have an active listener. You can use the code below in Azure Cloud Shell to update the managed nodes. `$ResourceGroupName` and `$Name` use the same variables as the deployment script, but you will need to use the `$Credential` specific to the VM you are managing.

PowerShell

```
Enable-AzVMPSRemoting -ResourceGroupName $ResourceGroupName -Name $Name
Invoke-AzVMCommand -ResourceGroupName $ResourceGroupName -Name $Name -
ScriptBlock {Set-NetFirewallRule -Name WINRM-HTTP-In-TCP-PUBLIC -
RemoteAddress Any} -Credential $Credential
Invoke-AzVMCommand -ResourceGroupName $ResourceGroupName -Name $Name -
ScriptBlock {winrm create winrm/config/Listener?Address=*+Transport=HTTP} -
Credential $Credential
```

Deploy manually on an existing Azure virtual machine

Before installing Windows Admin Center on your desired gateway VM, install a SSL certificate to use for HTTPS communication, or you can choose to use a self-signed certificate generated by Windows Admin Center. However, you will get a warning when trying to connect from a browser if you choose the latter option. You can bypass this warning in Edge by clicking **Details** > **Go on to the webpage** or, in Chrome, by selecting **Advanced** > **Proceed to [webpage]**. We recommend you only use self-signed certificates for test environments.

> ⓘ **Note**
>
> These instructions are for installing on Windows Server with Desktop Experience, not on a Server Core installation.

1. Download Windows Admin Center to your local computer.
2. Establish a remote desktop connection to the VM, then copy the MSI from your local machine and paste into the VM.
3. Double-click the MSI to begin installation, and follow the instructions in the wizard. Be aware of the following:
    - By default, the installer uses the recommended port 443 (HTTPS). If you want to select a different port, note that you need to open that port in your firewall as well.
    - If you have already installed an SSL certificate on the VM, ensure you select that option and enter the thumbprint.
4. Start the Windows Admin Center service (run C:/Program Files/Windows Admin Center/sme.exe)

Learn more about deploying Windows Admin Center.

Configure the gateway VM to enable HTTPS port access:

1. Navigate to your VM in the Azure portal and select **Networking**.
2. Select **Add inbound port rule** and select **HTTPS** under **Service**.

> ⓘ **Note**
>
> If you chose a port other than the default 443, choose **Custom** under Service and enter the port you chose in step 3 under **Port ranges**.

Accessing a Windows Admin Center gateway installed on an Azure VM

At this point, you should be able to access Windows Admin Center from a modern browser (Edge or Chrome) on your local computer by navigating to the DNS name of your gateway VM.

> ⓘ **Note**
>
> If you selected a port other than 443, you can access Windows Admin Center by navigating to https://<DNS name of your VM>:<custom port>

When you attempt to access Windows Admin Center, the browser will prompt for credentials to access the virtual machine on which Windows Admin Center is installed. Here you will need to enter credentials that are in the Local users or Local administrators group of the virtual machine.

In order to add other VMs in the VNet, ensure WinRM is running on the target VMs by running the following in PowerShell or the command prompt on the target VM: `winrm quickconfig`

If you haven't domain-joined the Azure VM, the VM behaves like a server in workgroup, so you'll need to make sure you account for using Windows Admin Center in a workgroup.

Feedback

Was this page helpful? Yes No

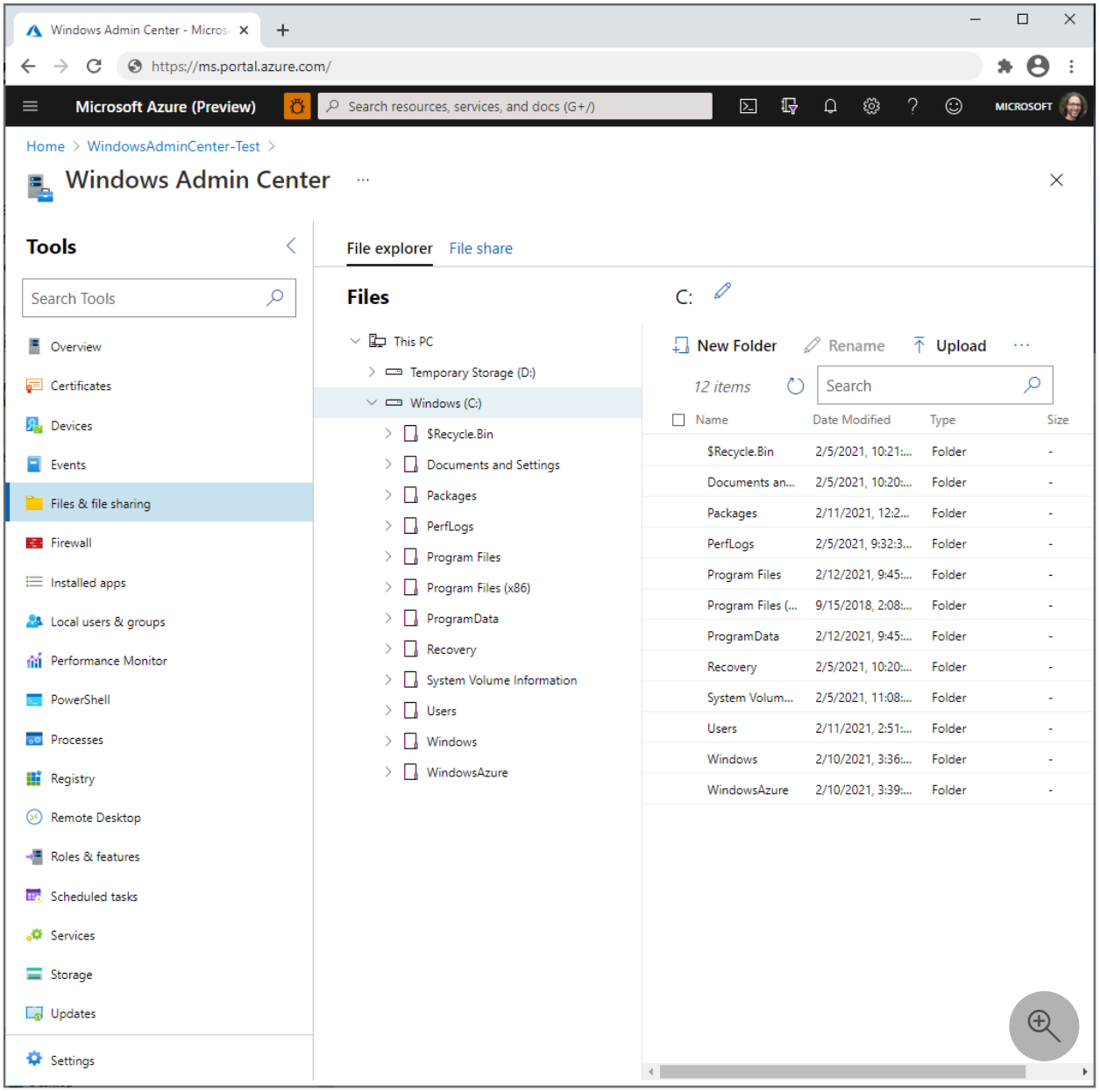
Provide product feedback

Manage a Windows VM using Windows Admin Center in Azure

Article • 07/16/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

You can now use Windows Admin Center in the Azure portal to manage the Windows operating system inside an Azure VM. Manage operating system functions from the Azure portal and work with files in the VM without using Remote Desktop or PowerShell.

This article provides an overview of the functionality provided, requirements, and how to install Windows Admin Center and use it to manage a single VM. It also answers frequently asked questions, and provides a list of known issues and tips for troubleshooting in case something doesn't work.



Overview of functionality

Windows Admin Center in the Azure portal provides the essential set of management tools for managing Windows Server and Client Azure VMs:

- Certificates
- Devices
- Events
- Files and file sharing
- Firewall
- Installed apps
- Local users and groups
- Performance Monitor
- PowerShell
- Processes
- Registry
- Remote Desktop
- Roles and features
- Scheduled tasks
- Services
- Storage
- Updates

We don't support extensions to Windows Admin Center in the Azure portal at this time.

If you manually installed Windows Admin Center in the VM to manage multiple systems, installing this VM extension reduces the functionality to managing just the VM in which the extension is installed. Uninstall the extension to get back full functionality.

Requirements

This section provides the requirements for using Windows Admin Center in the Azure portal to manage your Azure IaaS VM:

- Azure account with an active subscription
- Azure permissions
- Virtual machine requirements
- Networking requirements
- Management PC requirements

Azure account with an active subscription

You need an Azure account with an active subscription to deploy Windows Admin Center. If you don't have one already, you can create an account for free.

Azure permissions

To install the Windows Admin Center extension on your IaaS VM, your account must be granted the **Owner** or **Contributor** role in Azure.

Connecting to Windows Admin Center requires you to have **Reader** and **Windows Admin Center Administrator Login** permissions at the virtual machine resource level.

Learn more about configuring role assignment for your VM.

Virtual machine requirements

To use Windows Admin Center in the Azure portal, we install Windows Admin Center in each Azure VM that you want to use it to manage. The Azure VM has the following requirements:

- Windows Server 2022, Windows Server 2019, Windows Server 2016, Windows 10, or Windows 11
- At least 3 GiB of memory
- Be in any region of an Azure public cloud (it's not supported in Azure China, Azure Government, or other non-public clouds)

Networking requirements

The VM also has the following networking requirements, which we step through during the installation procedure:

- Outbound internet access or an outbound port rule allowing HTTPS traffic to the `WindowsAdminCenter` and `AzureActiveDirectory` service tag
- An inbound port rule if using a public IP address to connect to the VM (not recommended)

Just like with Remote Desktop, we recommend connecting to the VM using a private IP address in the VM's virtual network to increase security. Using a private IP address doesn't require an inbound port rule, though it does require access to the virtual network (which we discuss next).

> ⓘ **Note**

Inbound connectivity being redirected by another service (i.e. Azure Firewall) is not supported. You must have inbound connectivity from the Azure portal to one of the direct IP addresses of your VM (as seen on the "Networking" tab of your Azure VM in the Azure portal) on the port Windows Admin Center is installed.

Management PC requirements

The management PC or other system that you use to connect to the Azure portal has the following requirements:

- The Microsoft Edge or Google Chrome web browser
- Access to the virtual network that's connected to the VM (this is more secure than using a public IP address to connect). There are many ways to connect to a virtual network, including by using a VPN gateway.

Installing in a VM

Before you can use Windows Admin Center in the Azure portal, you must install it in the VM you want to manage. Here's how:

1. Open the Azure portal and navigate to your VM's settings.
2. If the VM has all outbound internet traffic blocked, create an outbound port rule to connect to the Windows Admin Center service.

   To do so, navigate to Windows Admin Center (found in the Settings group) and select the checkbox titled "Open an outbound port for Windows Admin Center to install" on the Install screen of Windows Admin Center. Alternatively, you can run the following PowerShell commands:

   PowerShell

   ```
   $allowWindowsAdminCenter = New-AzNetworkSecurityRuleConfig  -Name "PortForWACService"  -Access Allow -Protocol Tcp -Direction Outbound -Priority 100 -DestinationAddressPrefix WindowsAdminCenter -SourcePortRange * -SourceAddressPrefix * -DestinationPortRange 443
   $allowAAD = New-AzNetworkSecurityRuleConfig  -Name "PortForAADService" -Access Allow -Protocol Tcp -Direction Outbound -Priority 101 -DestinationAddressPrefix AzureActiveDirectory -SourcePortRange * -SourceAddressPrefix * -DestinationPortRange 443
   ```

3. In the virtual machine settings, navigate to **Windows Admin Center** (found in the **Settings** group).

4. To optionally provide access to your VM over the public internet from any IP address (convenient for testing but exposes the VM to attack from any host on the internet), you can select **Open this port for me**.

However, we recommend instead using a private IP address to connect with, or at least manually creating an inbound port rule that's locked down to accept traffic from only the IP addresses you specify.

1. Select **Install**.

   Installing takes a few minutes. If you selected **Open this port for me** or manually created an inbound port rule in the last couple minutes, it might take another couple minutes before you can connect with Windows Admin Center.

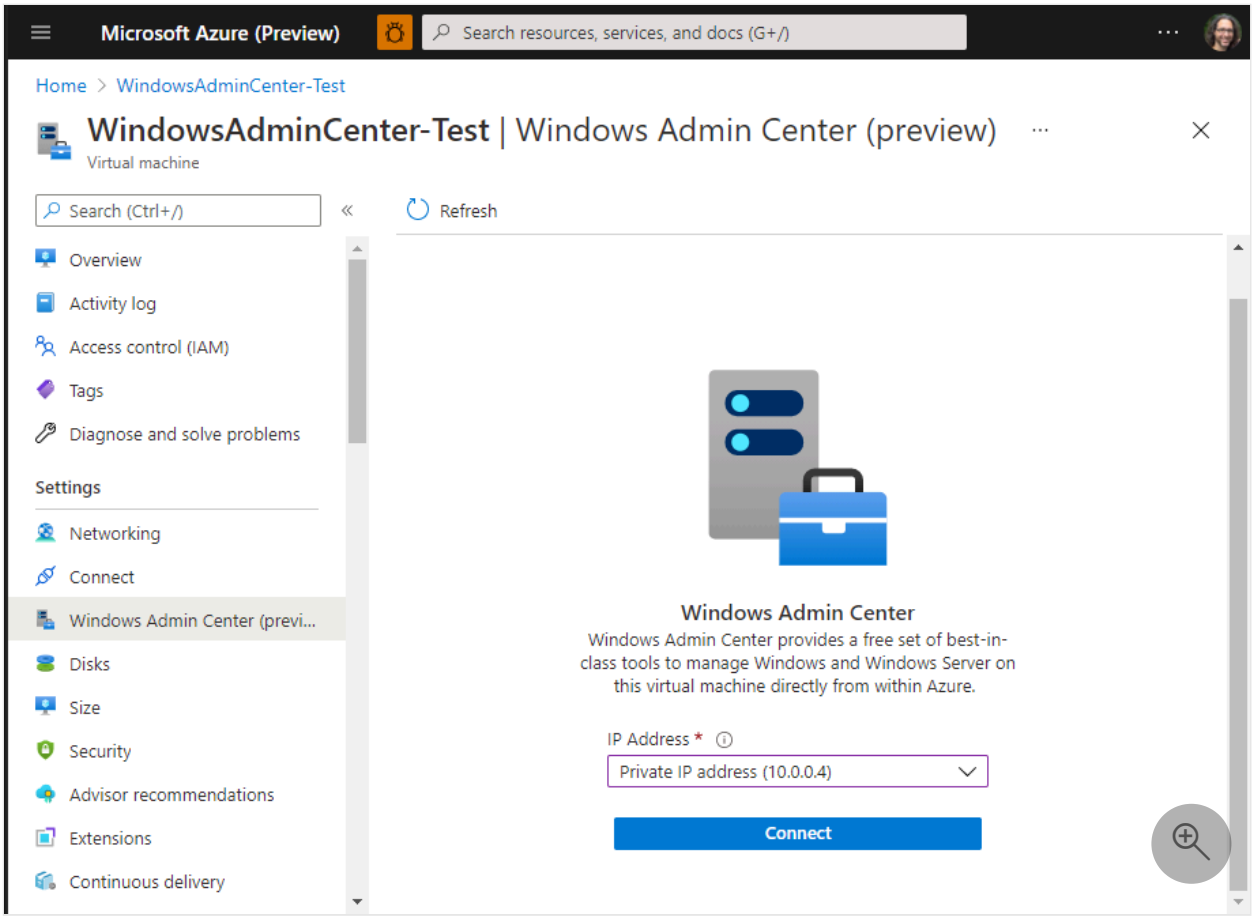
Using with a VM

After you've installed Windows Admin Center in an Azure VM, here's how to connect to it and use it to manage Windows:

1. Open the Azure portal and navigate to your VM, then Windows Admin Center.
2. Select the IP address you want to use when connecting to the VM, and then select **Connect**.

Windows Admin Center opens in the portal, giving you access to the same tools you might be familiar with from using Windows Admin Center in an on-premises deployment.

> ⓘ **Note**
>
> Starting August 2022, Windows Admin Center now allows you to use Microsoft Entra ID-based authentication for your Azure IaaS VM. You will no longer be prompted for the credentials of a local administrator account.



If you see a "Failed to connect" message, ensure your account is a member of the **Windows Admin Center Administrator Login** role on the VM resource.

Creating an inbound port rule for connecting from specific public IP addresses

Just like with Remote Desktop, opening an inbound port rule on your VM's public IP address exposes your VM to potential attack from any host on the internet, so we recommend instead accessing the VM using a private IP address.

However, if you need to use a public IP address, you can improve security by limiting the IP addresses that can reach your VM to only the IP addresses used by the systems you use to connect to the Azure portal. Here's how:

1. Open the Azure portal and navigate to your VM > **Networking** > **Inbound port rules**.

2. If you already installed Windows Admin Center and configured it to open an inbound port for your public IP address, select **PortForWAC**. Otherwise, select **Add inbound port rule**.

3. Provide the following values, specifying the public IP addresses of your management systems (separated with commas), and optionally changing the

destination port from port 6516. Then select **Add**.

Expand table

Field	Value
Source	IP address
Source IP addresses	*Management system IPs*
Source port ranges	*
Destination	Any
Destination port ranges	`6516`
Protocol	Any
Action	Allow

You might need to use a non-Microsoft website or app to find the public IP address of the system you're using to connect to the Azure portal.

Configuring role assignments for the VM

Access to Windows Admin Center is controlled by the **Windows Admin Center Administrator Login** Azure role.

> **Note**
>
> The Windows Admin Center Administrator Login role uses dataActions and thus cannot be assigned at management group scope. Currently these roles can only be assigned at the subscription, resource group or resource scope.

To configure role assignments for your VMs using the Microsoft Entra admin center experience:

1. Select the **Resource Group** containing the VM and its associated Virtual Network, Network Interface, Public IP Address or Load Balancer resource.
2. Select **Access control (IAM)**.
3. Select **Add** > **Add role assignment** to open the Add role assignment page.

4. Assign the following role. For detailed steps, see Assign Azure roles using the Azure portal.

Expand table

Setting	Value
Role	**Windows Admin Center Administrator Login**
Assign access to	User, group, service principal, or managed identity

For more information on how to use Azure RBAC to manage access to your Azure subscription resources, see the following articles:

- Assign Azure roles using Azure CLI
- Assign Azure roles using the Azure CLI examples. Azure CLI can also be used in the Azure Cloud Shell experience.
- Assign Azure roles using the Azure portal
- Assign Azure roles using Azure PowerShell.

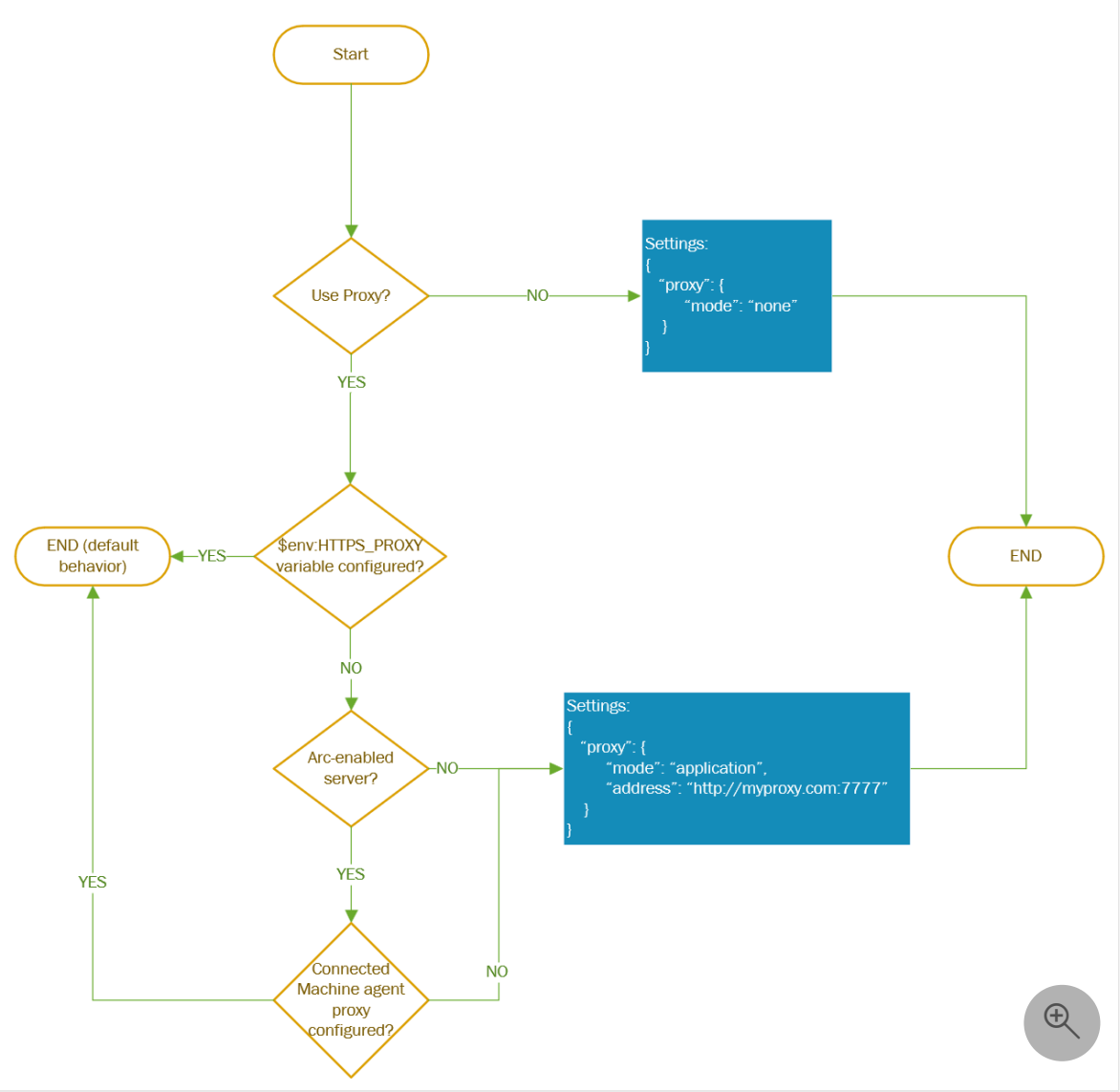
Proxy configuration

If the machine connects through a proxy server to communicate over the internet, review the following requirements to understand the network configuration required.

The Windows Admin Center extension can communicate through a proxy server by using the HTTPS protocol. Use the extensions settings for configuration as described in the following steps. Authenticated proxies are not supported.

> **Note**
>
> Proxy configuration is only supported for extension versions greater than 0.0.0.321.

1. Use this flowchart to determine the values of the `Settings` parameters



2. After you determine the `Settings` parameter values, provide these other parameters when you deploy the AdminCenter Agent. Use PowerShell commands, as shown in the following example:

PowerShell

```
$wacPort = "6516"
$settings = @{"port" = $wacPort; "proxy" = @{"mode" = "application";
"address" = "http://[address]:[port]";}}
Set-AzVMExtension -ExtensionName AdminCenter -ExtensionType AdminCenter -
Publisher Microsoft.AdminCenter -ResourceGroupName <resource-group-name> -
VMName <virtual-machine-name> -Location <location> -TypeHandlerVersion "0.0"
-settings $settings
```

Updating Windows Admin Center

We're constantly releasing new versions of Windows Admin Center. For Windows Admin Center to automatically update to the latest version, the Azure Virtual Machine needs a control plane operation to take place. In the event you wish to update sooner, you can run the following commands:

PowerShell

```
Set-AzContext <subscription_id>
Set-AzVMExtension -ExtensionName "AdminCenter" -Publisher
"Microsoft.AdminCenter" -ExtensionType "AdminCenter" -ResourceGroupName
<RG_name> -VMName <VM_name>
```

Implementation details

Windows Admin Center is currently implemented in the Azure portal in the form of an extension that we install on each Azure VM with which you want to use Windows Admin Center.

This extension connects to an external service that manages certificates and DNS records so that you can easily connect to your VM.

Each Azure VM that uses the Windows Admin Center extension gets a public DNS record that Microsoft maintains in Azure DNS. We hash the record name to anonymize the VM's IP address when saving it in DNS - the IP addresses aren't saved in plain text in DNS. This DNS record is used to issue a certificate for Windows Admin Center on the VM, enabling encrypted communication with the VM.

Connecting an Azure VM to Windows Admin Center deploys a virtual account in the administrators group, giving you full administrator access on your VM. Access to your VM is controlled by the **Windows Admin Center Administrator Login** role in Azure. An Azure user with the **Owner** or **Contributor** roles assigned for a VM doesn't automatically have privileges to log into the VM.

Troubleshooting

Here are some tips to try in case something isn't working. For general help troubleshooting Windows Admin Center (not specifically in Azure), see Troubleshooting Windows Admin Center.

Failed to connect error

1. In a new tab, open `https://<ip_address>:<port>`. If this page loads successfully with a certificate error, create a support request.

   If this page doesn't load successfully, there's something wrong with your connection to Windows Admin Center itself. Make sure that you're connected to the correct Vnet and are using the correct IP address before trying further troubleshooting.

2. If you're using a Public IP address, make sure that the port you selected upon installation is open to the internet. By default, the port is set to 6516. In your virtual machine, navigate to “Networking” > “Add inbound port rule”.

3. Make sure that the port can be reached.
   a. In the Azure portal, navigate to “Networking” and make sure that there are no conflicting rules with a higher priority that could be blocking the Windows Admin Center port
   b. In the Azure portal, navigate to “Connection troubleshoot” to test that your connection is working and the port can be reached.

4. Make sure that outbound traffic to Windows Admin Center is allowed on your virtual machine

   a. In the Azure portal, navigate to “Networking” and “Outbound port rules”.

   b. Create a new port rule for the `Windows Admin Center` and `Azure Active Directory` service tags.

   c. You can test this by running the following command using PowerShell inside of your virtual machine:

   PowerShell

   ```
   Invoke-RestMethod -Method GET -Uri
   https://<your_region>.service.waconazure.com
   ```

   Expected

   ```
   Microsoft Certificate and DNS service for Windows Admin Center in
   the Azure Portal
   ```

   d. If you allowed all outbound traffic and are still seeing an error from the command above, check that there are no firewall rules blocking connection. If nothing seems wrong, create a support request as our service might be experiencing problems.

5. Make sure that the Windows Admin Center service is running on your VM.
    a. In the Azure portal, navigate to “Connect” > “RDP” > “Download RDP File”.
    b. Open the RDP file and sign in with your administrator credentials.
    c. Open Task Manager (Ctrl+Shift+Esc) and navigate to “Services”.
    d. Make sure WindowsAdminCenter is Running. If not, start the service.

6. Check that your installation is in a good state.
    a. In the Azure portal, navigate to “Connect” > “RDP” > “Download RDP File”.
    b. Open the RDP file and sign in with your administrator credentials.
    c. Open a browser and type `https://localhost:<port>` replacing `<port>` with the port on which you installed Windows Admin Center. Not sure what port you installed it on? Check out the Frequently Asked Questions later in this article.
    d. If this doesn’t load, there might be something wrong with your installation. Go back to the Azure portal, navigate to “Extensions”, and uninstall the Admin Center extension. Navigate back to “Windows Admin Center” and reinstall the extension.

7. Check that the firewall rule is open for SmeInboundOpenException.
    a. In the Azure portal, navigate to “Connect” > “RDP” > “Download RDP File”.
    b. Open the RDP file and sign in with your administrator credentials.
    c. Open the Control Panel and navigate to Control Panel\System and Security\Windows Defender Firewall\Allowed apps.
    d. Ensure that the SmeInboundOpenException rule is enabled for both Private and Public, then try to connect again.

You get stuck on the "Windows Admin Center" loading page with the logo

This could occur if your browser blocks third party cookies. Currently, Windows Admin Center requires that you don't block third party cookies, and we're actively working to remove this requirement. In the meantime, please allow third party cookies in your browser.

1. On **Edge**:
    a. Navigate to the ellipses on the top right corner, and navigate to **Settings**
    b. Navigate to **Cookies and site permissions**
    c. Navigate to **Manage and delete cookies and site data**
    d. Ensure that the checkbox for **Block third-party cookies** is turned **off**

2. On **Chrome**
    a. Navigate to the ellipses on the top right corner, and navigate to **Settings**
    b. Navigate to **Privacy and Security**

c. Navigate to **Cookies and other site data**
d. Select the radio button for either **Block third-party cookies in Incognito** or **Allow all cookies**

One of the Windows Admin Center tools isn't loading or gives an error

Navigate to any other tool in Windows Admin Center and navigate back to the one that isn't loading.

If no other tool is loading, there might be a problem with your network connectivity. Try closing the blade and then connecting again. If this doesn't work, open a support ticket.

The Windows Admin Center extension failed to install

1. Double-check to make sure that the VM meets the requirements.
2. Make sure that outbound traffic to Windows Admin Center is allowed on your virtual machine.
   a. In the Azure portal, navigate to "Networking" and "Outbound port rules".
   b. Create a new outbound port rule for Windows Admin Center.
   c. Test connectivity by running the following command using PowerShell inside of your virtual machine:

   PowerShell

   ```
   Invoke-RestMethod -Method GET -Uri
   https://<your_region>.service.waconazure.com
   ```

   Expected

   ```
   Microsoft Certificate and DNS service for Windows Admin Center in
   the Azure Portal
   ```

3. If you have allowed all outbound traffic, and are getting an error from the command above, check that there are no firewall rules blocking the connection.

If nothing seems wrong and Windows Admin Center still won't install, open a support request with the following information:

- Logs in the Azure portal. This can be found under Settings > Extensions > AdminCenter > View Detailed Status
- Logs in the VM. Share the logs from the following locations:
  - C:\WindowsAzure\Logs\Plugins\AdminCenter
  - C:\Packages\Plugins\AdminCenter
- Network trace, if appropriate. Network traces can contain customer data and sensitive security details, such as passwords, so we recommend reviewing the trace and removing any sensitive details before sharing it.

Automate Windows Admin Center deployment using an ARM template

You can automate Windows Admin Center deployment in Azure portal by using this Azure Resource Manager template.

JSON

```
const deploymentTemplate = {
        "$schema": "https://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
        "contentVersion": "1.0.0.0",
        "parameters": {
            "vmName": {
                "type": "string"
            },
            "location": {
                "type": "string"
            },
            "extensionName": {
                "type": "string"
            },
            "extensionPublisher": {
                "type": "string"
            },
            "extensionType": {
                "type": "string"
            },
            "extensionVersion": {
                "type": "string"
            },
            "port": {
                "type": "string"
            },
            "salt": {
                "type": "string"
            }
        },
        "resources": [
```

```
            {
                "type": "Microsoft.Compute/virtualMachines/extensions",
                "name": "[concat( parameters('vmName'), '/' ,
parameters('extensionName') )]",
                "apiVersion": "2018-10-01",
                "location": "[parameters('location')]",
                "properties": {
                    "publisher": "[parameters('extensionPublisher')]",
                    "type": "[parameters('extensionType')]",
                    "typeHandlerVersion": "
[parameters('extensionVersion')]",
                    "autoUpgradeMinorVersion": true,
                    "settings": {
                        "port": "[parameters('port')]",
                        "salt": "[parameters('salt')]",
                    }
                }
            }
        ];

const parameters = {
    vmName: <VM name>,
    location: <VM location>,
    extensionName: "AdminCenter",
    extensionPublisher: "Microsoft.AdminCenter",
    extensionType: "AdminCenter",
    extensionVersion: "0.0",
    port: "6516",
    salt: ""
}
```

Automate Windows Admin Center deployment using PowerShell

You can also automate Windows Admin Center deployment in Azure portal by using this PowerShell script.

PowerShell

```
$resourceGroupName = <get VM's resource group name>
$vmLocation = <get VM location>
$vmName = <get VM name>
$vmNsg = <get VM's primary nsg>
$salt = ""

$wacPort = "6516"
$Settings = @{"port" = $wacPort; "salt" = $salt}

# Open outbound port rule for WAC service
Get-AzNetworkSecurityGroup -Name $vmNsg -ResourceGroupName
```

```
$resourceGroupName | Add-AzNetworkSecurityRuleConfig -Name
"PortForWACService" -Access "Allow" -Direction "Outbound" -
SourceAddressPrefix "VirtualNetwork" -SourcePortRange "*" -
DestinationAddressPrefix "WindowsAdminCenter" -DestinationPortRange "443" -
Priority 100 -Protocol Tcp | Set-AzNetworkSecurityGroup

# Open outbound port rule for AAD
Get-AzNetworkSecurityGroup -Name $vmNsg -ResourceGroupName
$resourceGroupName | Add-AzNetworkSecurityRuleConfig -Name
"PortForAADService" -Access "Allow" -Direction "Outbound" -
SourceAddressPrefix "VirtualNetwork" -SourcePortRange "*" -
DestinationAddressPrefix "AzureActiveDirectory" -DestinationPortRange "443"
-Priority 101 -Protocol Tcp | Set-AzNetworkSecurityGroup

# Install VM extension
Set-AzVMExtension -ResourceGroupName $resourceGroupName -Location
$vmLocation -VMName $vmName -Name "AdminCenter" -Publisher
"Microsoft.AdminCenter" -Type "AdminCenter" -TypeHandlerVersion "0.0" -
settings $Settings

# Open inbound port rule on VM to be able to connect to WAC
Get-AzNetworkSecurityGroup -Name $vmNsg -ResourceGroupName
$resourceGroupName | Add-AzNetworkSecurityRuleConfig -Name "PortForWAC" -
Access "Allow" -Direction "Inbound" -SourceAddressPrefix "*" -
SourcePortRange "*" -DestinationAddressPrefix "*" -DestinationPortRange
$wacPort -Priority 100 -Protocol Tcp | Set-AzNetworkSecurityGroup
```

Known issues

- If you change any of your networking rules, it takes Windows Admin Center about a minute or so to update its networking. The connection may fail for a few minutes.
- If you just started your virtual machine, it takes about a minute for the IP address to be registered with Windows Admin Center and thus, it may not load.
- The first load time of Windows Admin Center might be a little longer. Any subsequent load is just a few seconds.
- Chrome Incognito mode isn't supported.
- Azure portal desktop app is not supported.

Frequently asked questions

How much does it cost to use Windows Admin Center?

There's no cost to using the Windows Admin Center in the Azure portal.

Can I use Windows Admin Center to manage the virtual machines running on my Azure VM?

You can install the Hyper-V role using the Roles and Features extension. Once installed, refresh your browser, and Windows Admin Center will show the Virtual Machine and Switch extensions.

What operating systems can I manage using this extension?

You can use the extension to manage VMs running Windows Server 2016 or higher, or Windows 10/11.

How does Windows Admin Center handle security?

Traffic from the Azure portal to Windows Admin Center running on your VM uses HTTPS. Your Azure VM is managed using PowerShell and WMI over WinRM.

For an inbound port, why must I open a port and why should the source be set to “Any”?

Windows Admin Center installs on your Azure Virtual Machine. The installation consists of a web server and a gateway. By publishing the web server to DNS and opening the firewall (the inbound port in your VM), you can access Windows Admin Center from the Azure portal. The rules for this port perform very similar to the “RDP” port. If you don’t wish to open this port up to “Any”, we recommend specifying the rule to the IP address of the machine used to open the Azure portal.

Why must I create an outbound port rule?

There's an external Windows Admin Center service that manages certificates and DNS records for you. To allow your VM to interact with our service, you must create an outbound port rule.

Can I use PowerShell or the Azure CLI to install the extension on my VM?

Yes:

- PowerShell: Set-AzVMExtension

- Azure CLI: az vm extension set

I already have Windows Admin Center installed on my VM. Can I access it from the portal?

Yes, however you will still need to install the extension.

Is there any documentation on the general functionality of Windows Admin Center and its tools?

Yes, see Windows Admin Center overview and Manage Servers.

Do I have to install Windows Admin Center on each of my Azure VMs?

Yes, for our initial implementation, Windows Admin Center must be installed on every Azure VM you want to use it on.

Can I use Windows Admin Center to manage all servers and virtual machines?

Yes, you can use Windows Admin Center on-premises to manage servers and virtual machines on-premises and in Azure. For details, see Manage Azure VMs with Windows Admin Center.

Does Windows Admin Center in the Azure portal work with Azure Bastion?

No, unfortunately not.

Is Windows Admin Center supported for VMs behind a load balancer?

Yes.

Feedback

Was this page helpful? Yes No

Provide product feedback

Manage Azure Arc-enabled Servers using Windows Admin Center in Azure

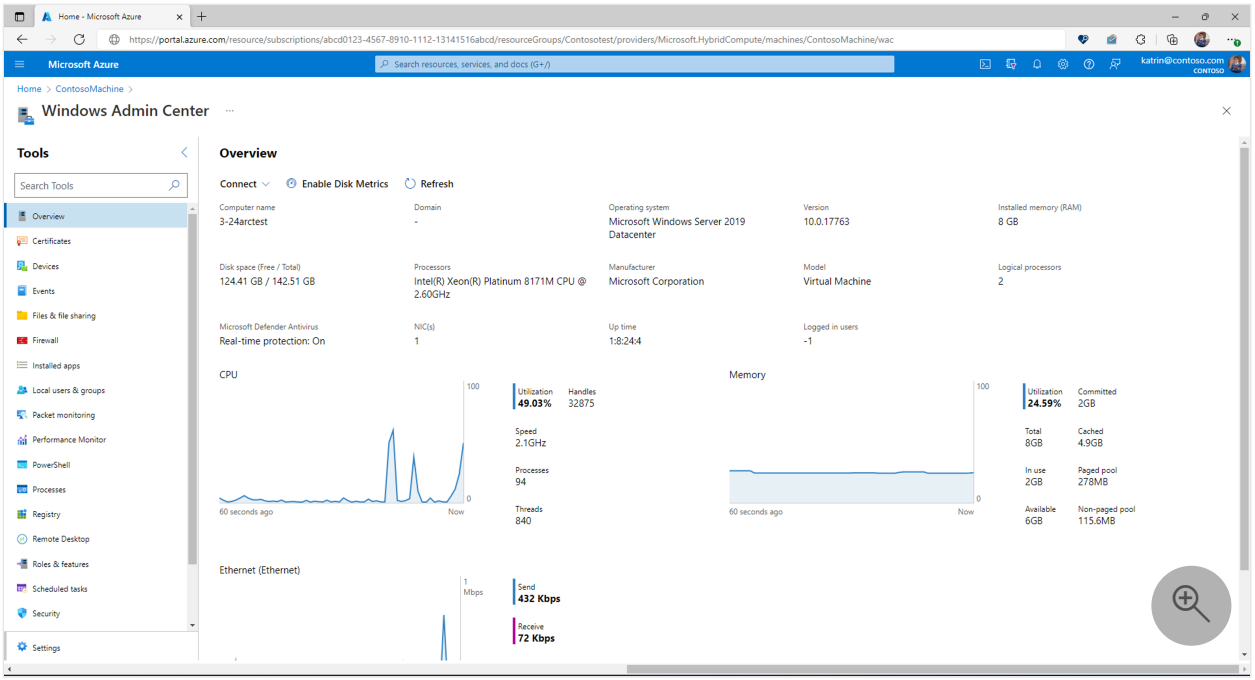
Article • 11/06/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

> **ⓘ Important**
>
> Version 1.36 and 1.35 of the Azure Connected Machine Agent (Arc agent) breaks connection to Windows Admin Center. This has been fixed in later versions of the Arc agent (1.37+) The current version can be **downloaded here**.

Using Windows Admin Center in the Azure portal allows you to manage the Windows Server operating system of your Arc-enabled servers, known as hybrid machines. You can securely manage hybrid machines from anywhere–without needing a VPN, public IP address, or other inbound connectivity to your machine. To learn more about Arc-enables servers, see What is Azure Arc-enabled servers?.

With Windows Admin Center extension in Azure, you get the management, configuration, troubleshooting, and maintenance functionality for managing your Arc-enabled servers in the Azure portal. Windows Server infrastructure and workload management no longer requires you to establish line-of-sight or Remote Desktop Protocol (RDP)–it can all be done natively from the Azure portal. Windows Admin Center provides tools that you'd normally find in Server Manager, Device Manager, Task Manager, Hyper-V Manager, and most other Microsoft Management Console (MMC) tools.

This article provides an overview of using Windows Admin Center in the Azure portal, requirements, and how to install Windows Admin Center in the Azure portal and use it to manage your hybrid machine. It also answers frequently asked questions, and provides a list of known issues and tips for troubleshooting in case something doesn't work.



Overview of Windows Admin Center in Azure

Windows Admin Center in the Azure portal provides essential tools for managing Windows Server running on a single hybrid machine. You can manage hybrid machines without the need to open any inbound ports on your firewall.

Using Windows Admin Center in the Azure portal, you can manage:

- Certificates
- Devices
- Events
- Files and file sharing
- Firewall
- Installed apps
- Local users and groups
- Performance Monitor
- PowerShell
- Processes
- Registry
- Remote Desktop
- Roles and Features
- Scheduled tasks
- Services
- Storage
- Updates
- Virtual machines
- Virtual switches

We don't support other extensions for Windows Admin Center in the Azure portal at this time.

> ⚠ **Warning**
>
> If you manually installed Windows Admin Center on your hybrid machine to manage multiple systems, enabling Windows Admin Center in Azure will replace your existing instance of Windows Admin Center and removes the capability to manage other machines. You will lose access to your previously deployed instance of Windows Admin Center.

Requirements

This section provides the requirements for using Windows Admin Center in the Azure portal to manage a hybrid machine:

- Azure account with an active subscription
- Azure permissions
- Azure region availability
- Hybrid machine requirements
- Networking requirements

Azure account with an active subscription

You'll need an Azure account with an active subscription to deploy Windows Admin Center. If you don't have one already, you can create an account for free.

During the deployment of Windows Admin Center, we'll attempt to register the *Microsoft.HybridConnectivity* resource provider for your subscription.

> ⓘ **Important**
>
> You must have permission to register a resource provider, which requires the `*/register/action` operation. This is included if you are assigned the **contributor or owner role** on your subscription.

> ⓘ **Note**
>
> Resource provider registration is a one time task per subscription.

To check the status of the resource provider and register if needed:

1. Sign in to the Azure portal.
2. Select **Subscriptions**.
3. Select the name of your subscription.
4. Select **Resource providers**.
5. Search for **Microsoft.HybridConnectivity**.
6. Verify that the status of Microsoft.HybridConnectivity is **Registered**.
    a. If the status is *NotRegistered*, select **Microsoft.HybridConnectivity**, and then select **Register**.

Azure permissions

To install the Windows Admin Center extension for an Arc-enabled server resource, your account must be granted the **Owner**, **Contributor**, or **Windows Admin Center Administrator Login** role in Azure.

Connecting to Windows Admin center requires you to have **Reader** and **Windows Admin Center Administrator Login** permissions at the Arc-enabled server resource.

Learn more about assigning Azure roles using the Azure portal

Azure region availability

Windows Admin Center is supported in the following Azure regions:

- Australia East
- Brazil South
- Canada Central
- Canada East
- Central India
- Central US
- East Asia
- East US
- East US 2
- France Central
- Japan East
- Korea Central
- North Central US
- North Europe
- South Africa North
- South Central US

- Southeast Asia
- Sweden Central
- Switzerland North
- UAE North
- UK South
- UK West
- West Central US
- West Europe
- West US
- West US 2
- West US 3

> **Note**
>
> Windows Admin Center isn't supported in Azure China 21Vianet, Azure Government, or other non-public clouds

Hybrid machine requirements

To use Windows Admin Center in the Azure portal, the Windows Admin Center agent must be installed on each hybrid machine you wish to manage via an Azure VM extension. Ensure your machine is connected to Azure Arc before proceeding. To learn more about onboarding your machine to Azure Arc, see Connect Windows Server machines to Azure through Azure Arc Setup. The hybrid machine should meet the following requirements:

- Windows Server 2016 or later
- 3 GB of RAM or more
- Azure Arc agent version 1.13.21320.014 or later
- You must use Windows Server Pay-as-you-go or Software Assurance license types

> **Important**
>
> As of November 1st, 2024, Windows Admin Center for Azure Arc requires your hybrid machine to have a Pay-as-you-go or Software Assurance license type to use. All new installations of the Windows Admin Center agent must adhere to this requirement. Machines with Windows Admin Center for Azure Arc installed prior to November 1st, 2024, may continue to use Windows Admin Center for Azure Arc for up to 12 months without updates until November 1st, 2025.

Networking requirements

The hybrid machine must meet the following networking requirements:

- Outbound internet access or an outbound port rule allowing HTTPS traffic to the following endpoints:
  - `*service.waconazure.com` or the `WindowsAdminCenter` service tag
  - `pas.windows.net`
  - `*.servicebus.windows.net`

> **Note**
>
> No inbound ports are required in order to use Windows Admin Center.

The management machine where the Azure portal is running must meet the following networking requirements:

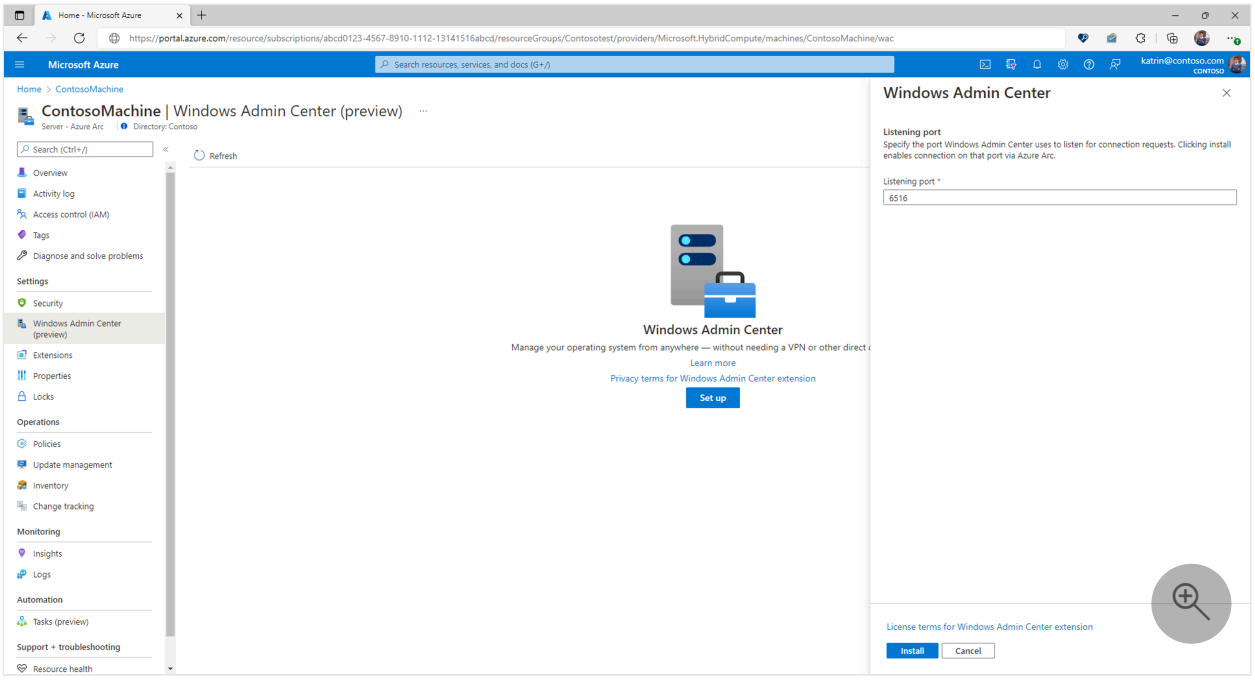
- Outbound internet access over port `443`

Make sure you review the supported devices and recommended browsers before accessing the Azure portal from the management machine or system.

Install Windows Admin Center in the Azure portal

Before you can use Windows Admin Center in the Azure portal, you must deploy the Windows Admin Center VM extension using the following steps:

1. Open the Azure portal and navigate to your Arc-enabled server.
2. Under the **Settings** group, select **Windows Admin Center**.
3. Specify the port on which you wish to install Windows Admin Center, and then select **Install**.



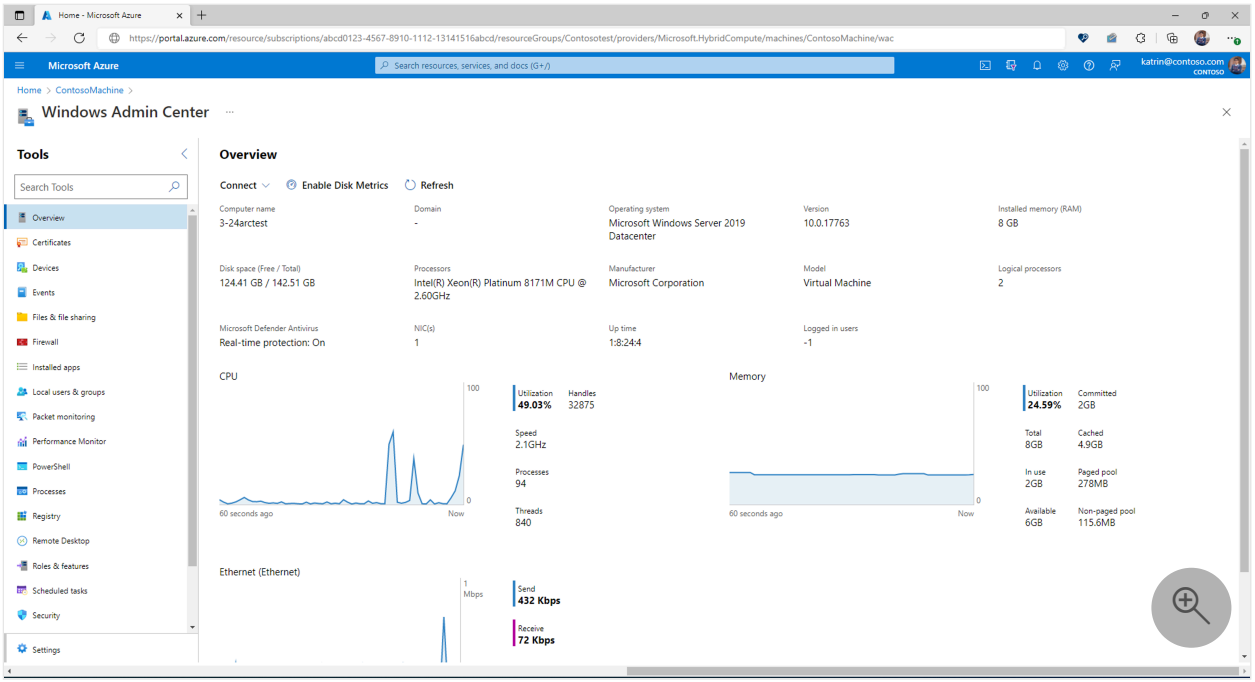
Connecting to Windows Admin Center in the Azure portal

After you've installed Windows Admin Center on your hybrid machine, perform the following steps to connect to it and use it to manage Windows Server:

1. Open the Azure portal and navigate to your Arc-enabled server, and then under the **Settings** group, select **Windows Admin Center (preview)**.
2. Select **Connect**.

> **Note**
>
> Starting August 2022, Windows Admin Center now allows you to use Microsoft Entra ID-based authentication for your hybrid machine. You will no longer be prompted for the credentials of a local administrator account.

Windows Admin Center opens in the portal, giving you access to the same tools you might be familiar with from using Windows Admin Center in an on-premises deployment.



Configuring role assignments

Access to Windows Admin Center is controlled by the **Windows Admin Center Administrator Login** Azure role.

> **Note**
>
> The Windows Admin Center Administrator Login role uses dataActions and thus cannot be assigned at management group scope. Currently these roles can only be assigned at the subscription, resource group or resource scope.

To configure role assignments for your hybrid machines using the Microsoft Entra admin center experience:

1. Open the hybrid machine that you wish to manage using Windows Admin Center.
2. Select **Access control (IAM)**.
3. Select **Add** > **Add role assignment** to open the Add role assignment page.
4. Assign the following role. For detailed steps, see Assign Azure roles using the Azure portal.

Expand table

Setting	Value
Role	**Windows Admin Center Administrator Login**

Setting	Value
Assign access to	User, group, service principal, or managed identity

For more information on how to use Azure RBAC to manage access to your Azure subscription resources, see the following articles:

- Assign Azure roles using Azure CLI
- Assign Azure roles using the Azure CLI examples. Azure CLI can also be used in the Azure Cloud Shell experience.
- Assign Azure roles using the Azure portal
- Assign Azure roles using Azure PowerShell.

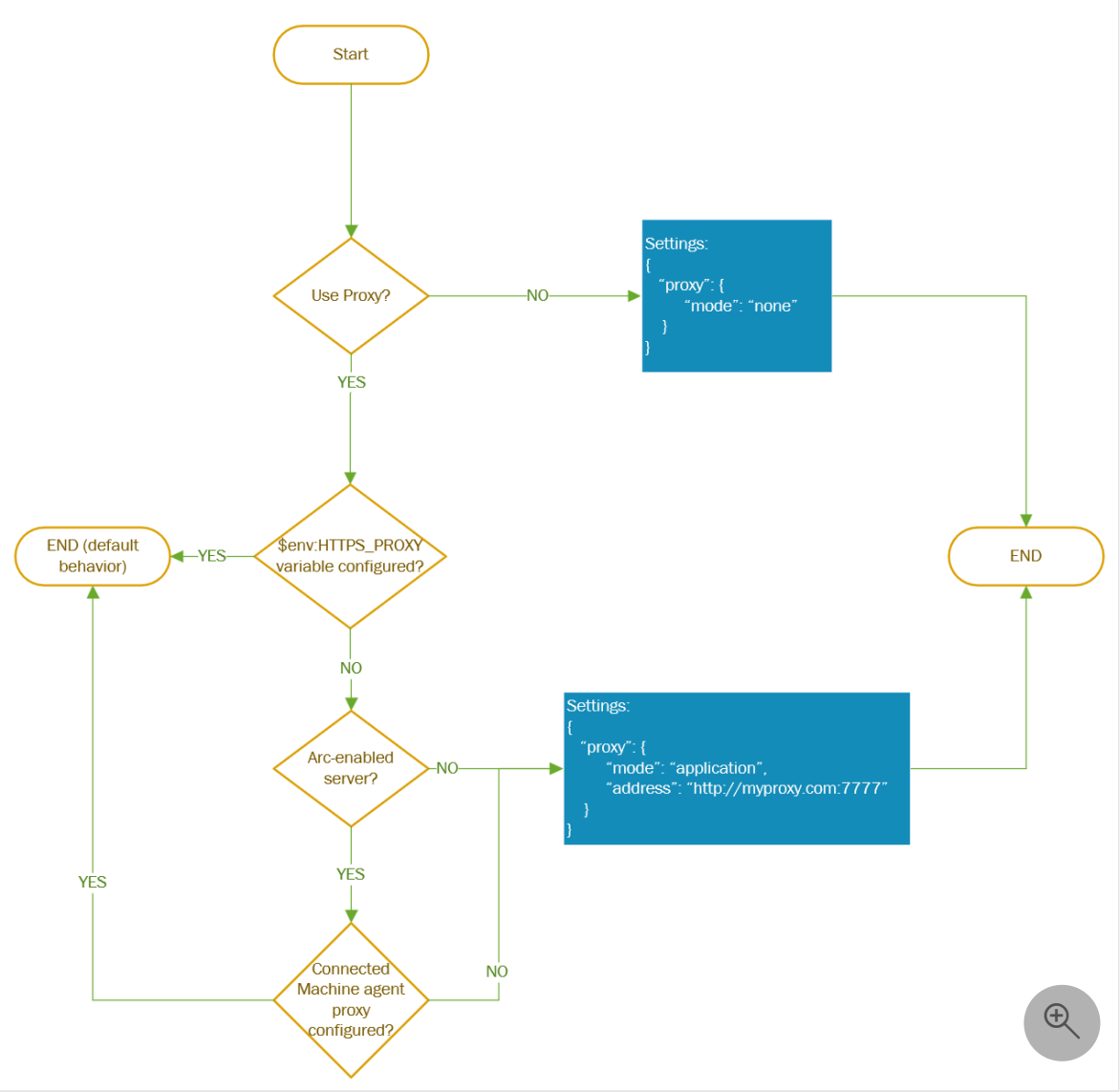
Proxy configuration

If the machine connects through a proxy server to communicate over the internet, review the following requirements to understand the network configuration required.

The Windows Admin Center extension can communicate through a proxy server by using the HTTPS protocol. Use the extensions settings for configuration as described in the following steps. Authenticated proxies aren't supported.

> ⓘ **Note**
>
> Proxy configuration is only supported for extension versions greater than 0.0.0.321.

1. Use this flowchart to determine the values of the `Settings` parameters



2. After you determine the `Settings` parameter values, provide these other parameters when you deploy the AdminCenter Agent. Use PowerShell commands, as shown in the following example:

PowerShell

```
$wacPort = "6516"
$settings = @{"port" = $wacPort; "proxy" = @{"mode" = "application";
"address" = "http://[address]:[port]";}}
New-AzConnectedMachineExtension -Name AdminCenter -ExtensionType AdminCenter
-Publisher Microsoft.AdminCenter -ResourceGroupName <resource-group-name> -
MachineName <arc-server-name> -Location <arc-server-location> -Setting
$settings -SubscriptionId <subscription-id>
```

How it works

By using Windows Admin Center in Azure, you can connect to your hybrid machine without requiring any inbound port to be enabled on the firewall. Windows Admin Center, via the Arc agent, is able to securely establish a reverse proxy session connection with the Azure Arc service in an outbound manner.

For each hybrid machine that you want to manage with Windows Admin Center in the Azure portal, you must deploy an agent to each machine.

The agent communicates to an external service that manages certificates so that you can easily connect to your hybrid machine.

Clicking **Install** performs the following actions:

1. Registers the *Microsoft.HybridConnectivity* resource provider on your subscription. The resource provider hosts the proxy used for communication to your Arc-enabled server.
2. Deploys an Azure *endpoint* resource on top of your Arc-enabled resource that enables a reverse proxy connection on the specified port. This is simply a logical resource in Azure, and doesn't deploy anything on your server itself.
3. Installs the Windows Admin Center agent on your hybrid machine with a valid TLS certificate.

> ⓘ **Note**
>
> Uninstalling Windows Admin Center does not delete the logical Azure endpoint resource. This is kept for other experiences that might leverage this resource, such as SSH.

Clicking **Connect** performs the following actions:

1. The Azure portal asks the *Microsoft.HybridConnectivity* resource provider for access to the Arc-enabled server.
2. The resource provider communicates with a Layer 4 SNI proxy to establish a short-lived session-specific access to your Arc-enabled server on the Windows Admin Center port.
3. A unique short-lived URL is generated and connection to Windows Admin Center is established from the Azure portal.

Connection to Windows Admin Center is end-to-end encrypted with SSL termination happening on your hybrid machine.

Automate Windows Admin Center deployment using PowerShell

You can automate Windows Admin Center deployment in Azure portal using this example PowerShell script.

PowerShell

```
$location = "<location_of_hybrid_machine>"
$machineName = "<name_of_hybrid_machine>"
$resourceGroup = "<resource_group>"
$subscription = "<subscription_id>"
$port = "6516"
$portint = 6516

#Deploy Windows Admin Center
$Setting = @{"port" = $port; "proxy" = @{"mode" = "application"; "address" =
"http://[address]:[port]";}} #proxy configuration is optional
New-AzConnectedMachineExtension -Name "AdminCenter" -ResourceGroupName
$resourceGroup -MachineName $machineName -Location $location -Publisher
"Microsoft.AdminCenter" -Settings $Setting -ExtensionType "AdminCenter" -
SubscriptionId $subscription

#Allow connectivity
$putPayload = "{'properties': {'type': 'default'}}"
Invoke-AzRestMethod -Method PUT -Uri
"https://management.azure.com/subscriptions/${subscription}/resourceGroups/$
{resourceGroup}/providers/Microsoft.HybridCompute/machines/${machineName}/pr
oviders/Microsoft.HybridConnectivity/endpoints/default?api-version=2023-03-
15" -Payload $putPayload

$patch = @{ "properties" =  @{ "serviceName" = "WAC"; "port" = $portint}}
$patchPayload = ConvertTo-Json $patch
Invoke-AzRestMethod -Method PUT -Path
/subscriptions/${subscription}/resourceGroups/${resourceGroup}/providers/Mic
rosoft.HybridCompute/machines/${machineName}/providers/Microsoft.HybridConne
ctivity/endpoints/default/serviceconfigurations/WAC?api-version=2023-03-15 -
Payload $patchPayload
```

Troubleshooting

Here are some tips to try in case something isn't working. For general Windows Admin Center troubleshooting (not specifically in Azure), see Troubleshooting Windows Admin Center.

Failed to connect with "404 endpoint not found"

1. Version 1.36 and 1.35 of the Azure Connected Machine Agent (Arc agent) breaks connection to Windows Admin Center. This has been fixed in later versions of the Arc agent (1.37+) This can be downloaded here.

Failed to connect error

1. Restart the HIMDS service.

   a. RDP into your server.

   b. Open PowerShell as an administrator and run:

   ```PowerShell
   Restart-Service -Name himds
   ```

2. Check that your Extension version is 0.0.0.169 or higher.
   a. Navigate to "Extensions"
   b. Check that the "AdminCenter" extension version is 0.0.0.169 or higher
   c. If not, uninstall the extension and reinstall it

3. Make sure that the Windows Admin Center service is running on your machine.
   a. RDP into your server.
   b. Open **Task Manager (Ctrl+Shift+Esc)** and navigate to **Services**.
   c. Make sure **ServerManagementGateway / Windows Admin Center** is running.
   d. If it isn't running, start the service.

4. Check that the port is enabled for reverse proxy session.

   a. RDP into your server.

   b. Open PowerShell as an administrator and run:

   ```PowerShell
   azcmagent config list
   ```

   c. This should return a list of ports under the incomingconnections.ports (preview) configuration that are enabled to be connected from Azure. Confirm that the port on which you installed Windows Admin Center is on this list. For example, if Windows Admin Center is installed on port 443, the result would be:

   ```Output
   ```

```
Local configuration setting
incomingconnections.ports (preview): 443
```

d. In the event it isn't on this list, run

```
PowerShell

azcmagent config set incomingconnections.ports <port>
```

If you're using another experience (like SSH) using this solution, you can specify multiple ports separated by a comma.

5. Ensure you have outbound connectivity to the necessary ports
   a. The hybrid machine should have outbound connectivity to the following endpoints:

   - `*.wac.azure.com`, `*.waconazure.com` or the WindowsAdminCenter ServiceTag
   - `pas.windows.net`
   - `*.servicebus.windows.net`

One of the Windows Admin Center tools isn’t loading or gives an error

1. Navigate to any other tool in Windows Admin Center and navigate back to the one that isn’t loading.

2. If no other tool is loading, there might be a problem with your network connectivity. Try closing the blade and then connecting again. If this doesn’t work, open a support ticket.

The Windows Admin Center extension failed to install

1. Double-check to make sure that the hybrid machine meets the requirements.

2. Make sure that outbound traffic to Windows Admin Center is allowed on your hybrid machine

   a. Test connectivity by running the following command using PowerShell inside of your virtual machine:

```
PowerShell
```

```
Invoke-RestMethod -Method GET -Uri
https://<your_region>.service.waconazure.com
```

Output

```
Microsoft Certificate and DNS service for Windows Admin Center in
the Azure Portal
```

3. If you've allowed all outbound traffic and are getting an error from the command above, check that there are no firewall rules blocking the connection.

If nothing seems wrong and Windows Admin Center still won't install, open a support request with the following information:

- Logs from the Azure portal. Windows Admin Center logs can be found under **Settings** > **Extensions** > **AdminCenter** > **View Detailed Status**.
- Logs in the hybrid machine. Run the following PowerShell command and share the resulting .zip file.

  PowerShell

  ```
  azcmagent logs
  ```

- Network trace, if appropriate. Network traces can contain customer data and sensitive security details, such as passwords, so we recommend reviewing the trace and removing any sensitive details before sharing it.

Known issues

- Chrome incognito mode isn't supported.
- Azure portal desktop app isn't supported.
- Detailed error messages for failed connections aren't yet available.
- As of November 1st, 2024, some existing customers are unable to use Windows Admin Center for Azure Arc.
- As of November 1st, 2024, some customers are unable to install or use Windows Admin Center for Azure Arc, even with proper license attestation.

Frequently asked questions

Find answers to the frequently asked questions about using Windows Admin Center in Azure.

How much does it cost to use Windows Admin Center?

There's no associated cost using the Windows Admin Center in the Azure portal.

Can I use Windows Admin Center to manage the virtual machines running on my server?

You can install the Hyper-V role using the Roles and Features extension. Once installed, refresh your browser, and Windows Admin Center displays the Virtual Machine and Switch extensions.

What servers can I manage using this extension?

You can use the capability to manage Arc-enabled Windows Server 2016 and later. You can also use Windows Admin Center in Azure to manage Azure Local.

How does Windows Admin Center handle security?

Traffic from the Azure portal to Windows Admin Center is end-to-end encrypted. Your Arc-enabled server is managed using PowerShell and WMI over WinRM.

Do I need an inbound port to use Windows Admin Center?

No inbound connection is required to use Windows Admin Center.

Why must I create an outbound port rule?

An outbound port rule is required for the service that we have built to communicate with your server. Our service issues you a certificate free-of-cost for your instance of Windows Admin Center. This service ensures that you can always connect to your instance of Windows Admin Center from the Azure portal by keeping your WAC certificate up to date.

Furthermore, accessing Windows Admin Center from Azure requires no inbound port and only outbound connectivity via a reverse proxy solution. These outbound rules are required in order to establish the connection.

How do I find the port used for Windows Admin Center installation?

To verify the value of SmePort registry setting:

1. RDP into your server
2. Open the **Registry Editor**
3. Navigate to the key `\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\ServerManagementGateway`
4. Read the value of `SmePort` to find the port used

Can I use PowerShell or the Azure CLI to install the extension on my VM?

Yes, to install the extension using the Azure CLI, run the following command from a command prompt:

Azure CLI

```
az connectedmachine extension create
```

You can also install the extension using PowerShell. Learn more about how to automate Windows Admin Center deployment using PowerShell.

I already have Windows Admin Center installed on my Arc server. Can I access it from the portal?

Yes. You can follow the same steps outlined in this document.

> ⚠ **Warning**
>
> Enabling this capability will replace your existing instance of Windows Admin Center and removes the capability to manage other machines. Your previously deployed instance of Windows Admin Center will no longer be usable. Please don't do this if you use your instance of Admin Center to manage multiple servers.

Next steps

- Learn about Windows Admin Center
- Learn about managing servers with Windows Admin Center

- Learn about Azure Arc

Feedback

Was this page helpful? Yes No

Provide product feedback

Manage Azure Local clusters using Windows Admin Center in Azure (preview)

Article • 12/05/2024 • Applies to: ✅ Azure Local, versions 23H2 and 22H2

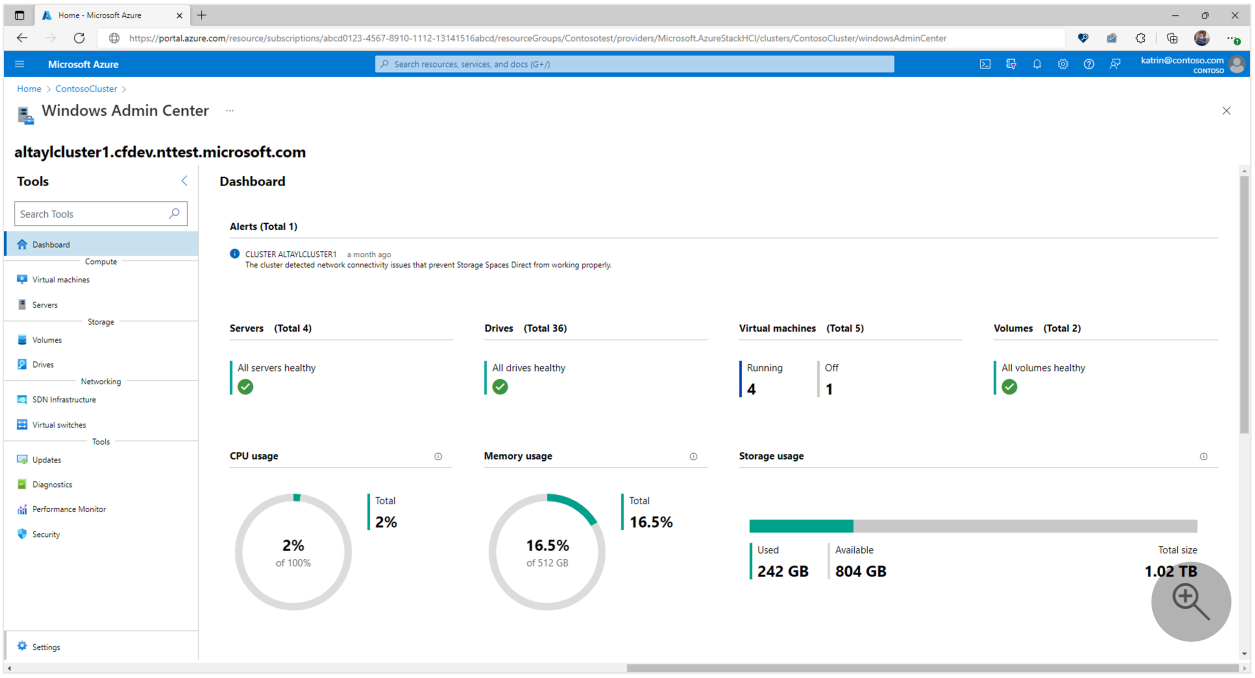
> **Important**
>
> Windows Admin Center in the Azure portal is currently in preview. See the Supplemental Terms of Use for Microsoft Azure Previews for legal terms that apply to Azure features that are in beta, preview, or otherwise not yet released into general availability.

> **Important**
>
> Version 1.36 and 1.35 of the Azure Connected Machine Agent (Arc agent) breaks connection to Windows Admin Center. This has been fixed in later versions of the Arc agent (1.37+) This can be downloaded here.

Using Windows Admin Center in the Azure portal you can manage the Azure Local operating system of your cluster. You can securely manage your cluster from anywhere–without needing a VPN, public IP address, or other inbound connectivity to your machine.

With Windows Admin Center extension in Azure, you get the management, configuration, troubleshooting, and maintenance functionality for managing your Azure Local cluster in the Azure portal. Azure Local cluster and workload management no longer require you to establish line-of-sight or Remote Desktop Protocol (RDP)–it can all be done natively from the Azure portal. Windows Admin Center provides tools that you'd normally find in Failover cluster manager, Device Manager, Task Manager, Hyper-V Manager, and most other Microsoft Management Console (MMC) tools.

This article provides an overview of using Windows Admin Center in the Azure portal, requirements, and how to install Windows Admin Center and use it to manage your cluster. It also answers frequently asked questions, and provides a list of known issues and tips for troubleshooting in case something doesn't work.



Overview of Windows Admin Center in Azure

Windows Admin Center in the Azure portal provides essential tools for managing your Azure Local cluster. You can manage clusters without the need to open any inbound port on your firewall.

Using Windows Admin Center in the Azure portal, you can manage:

- Servers
- Volumes
- Drives
- SDN infrastructure
- Diagnostics
- Security
- Certificates
- Devices
- Events
- Files and file sharing
- Firewall
- Installed apps
- Local users and groups
- Performance Monitor
- PowerShell
- Processes
- Registry
- Remote Desktop
- Roles and Features

- Scheduled tasks
- Services
- Storage
- Virtual machines
- Virtual switches

We don't support other extensions for Windows Admin Center in the Azure portal at this time.

> ⚠ **Warning**
>
> If you manually installed Windows Admin Center on your cluster to manage multiple systems, enabling Windows Admin Center in Azure will replace your existing instance of Windows Admin Center and removes the capability to manage other machines. You will lose access to your previously deployed instance of Windows Admin Center.

Requirements

This section provides the requirements for using Windows Admin Center in the Azure portal to manage a hybrid machine:

- Azure account with an active subscription
- Azure permissions
- Azure region availability
- Azure Local requirements
- Networking requirements

Azure account with an active subscription

You'll need an Azure account with an active subscription to deploy Windows Admin Center. If you don't have one already, you can create an account for free.

During the deployment of Windows Admin Center, you'll register the *Microsoft.HybridConnectivity* resource provider for your subscription.

> ⓘ **Important**

You must have permission to register a resource provider, which requires the `*/register/action` operation. This is included if you are assigned the **contributor or owner role** on your subscription.

> **Note**
>
> Resource provider registration is a one time task per subscription.

To check the status of the resource provider, and register if needed:

1. Sign in to the Azure portal.
2. Select **Subscriptions**.
3. Select the name of your subscription.
4. Select **Resource providers**.
5. Search for **Microsoft.HybridConnectivity**.
6. Verify that the status of Microsoft.HybridConnectivity is **Registered**.
    a. If the status is *NotRegistered*, select **Microsoft.HybridConnectivity**, and then select **Register**.

Azure permissions

Connecting to Windows Admin center requires you to have **Reader** and **Windows Admin Center Administrator Login** permissions at the Arc-enable Azure Local resource.

Learn more about assigning Azure roles using the Azure portal.

Azure region availability

Windows Admin Center is supported in all public regions Azure Local is supported.

> **Note**
>
> Windows Admin Center isn't supported in Azure China 21Vianet, Azure Government, or other non-public clouds

Azure Local requirements

To use Windows Admin Center in the Azure portal, the Windows Admin Center agent must be installed on every node of your cluster via an Azure VM extension. Each node of the cluster should meet the following requirements:

- Azure Local, version 21H2 or later
- 3 GB of memory or more
- The Azure Local cluster must be connected to Azure using Azure Arc
- Azure Arc agent version 1.13.21320.014 or later

Networking requirements

Every node of the Azure Local cluster must meet the following networking requirements:

- Outbound internet access or an outbound port rule allowing HTTPS traffic to the following endpoints:
  - `*.service.waconazure.com` or the `WindowsAdminCenter` service tag
  - `pas.windows.net`
  - `*.servicebus.windows.net`

> **Note**
>
> No inbound ports are required in order to use Windows Admin Center.

The management machine where the Azure Portal is running must meet the following networking requirements:

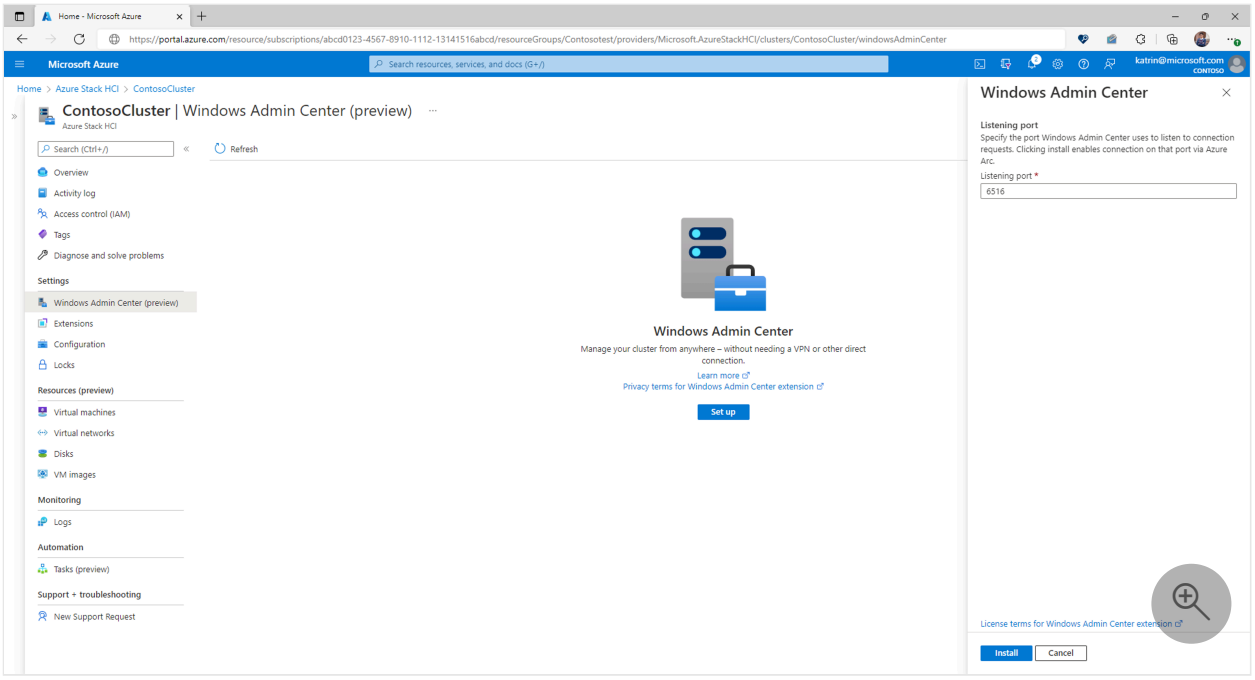
- Outbound internet access over port `443`

Make sure you review the supported devices and recommended browsers before accessing the Azure portal from the management machine or system.

Install Windows Admin Center in the Azure portal

Before you can use Windows Admin Center in the Azure portal, you must deploy the Windows Admin Center VM extension using the following steps:

1. Open the Azure portal and navigate to your Azure Local cluster.
2. Under the **Settings** group, select **Windows Admin Center**.
3. Specify the port on which you wish to install Windows Admin Center, and then select **Install**.



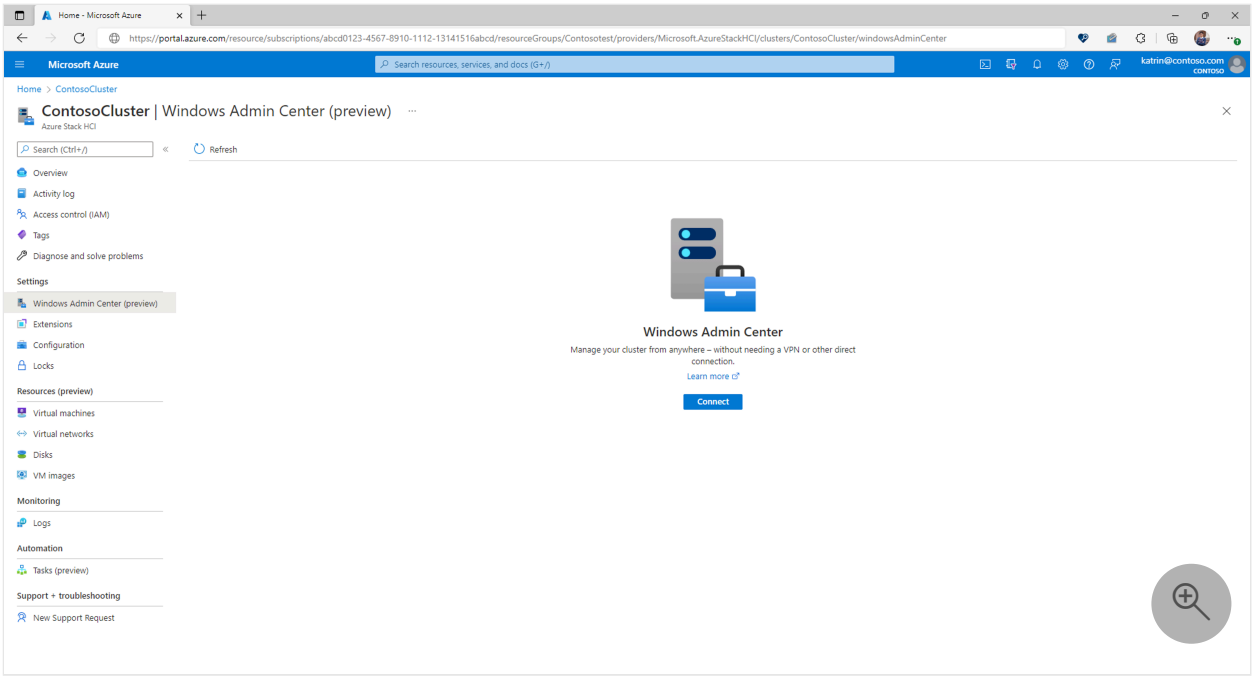
Connect to Windows Admin Center in the Azure portal

After you've installed Windows Admin Center on your cluster, perform the following steps to connect to it and use it to manage Azure Local:

1. Open the Azure portal and navigate to your Azure Local cluster, and then under the **Settings** group, select **Windows Admin Center**.
2. Select **Connect**.

> **Note**
>
> Starting April 2023, Windows Admin Center now allows you to use Microsoft Entra ID-based authentication for your 22H2 or higher clusters running the AdminCenter extension greater than 0.0.0.313. You will no longer be prompted for the credentials of a local administrator account. However, there may still be some experiences within Windows Admin Center that might require local administrator credentials. For example, when CredSSP is required. Clusters running 21H2 or below will continue to require local administrator credentials.

Windows Admin Center opens in the portal, giving you access to the same tools you might be familiar with from using Windows Admin Center in an on-premises deployment.



Configuring role assignments

Access to Windows Admin Center is controlled by the **Windows Admin Center Administrator Login** Azure role. You **must** have this role configured on the Azure Local resource, **and** each of the Azure Arc-enabled servers associated with this cluster.

> **Note**
>
> The Windows Admin Center Administrator Login role uses dataActions and thus cannot be assigned at management group scope. Currently these roles can only be assigned at the subscription, resource group or resource scope.

To configure role assignments for your cluster using the Microsoft Entra admin center experience:

1. Select the **Resource Group** containing the cluster and the associated Azure Arc resources.
2. Select **Access control (IAM)**.
3. Select **Add** > **Add role assignment** to open the Add role assignment page.
4. Assign the following role. For detailed steps, see Assign Azure roles using the Azure portal.

Expand table

Setting	Value
Role	**Windows Admin Center Administrator Login**
Assign access to	User, group, service principal, or managed identity

For more information on how to use Azure RBAC to manage access to your Azure subscription resources, see the following articles:

- Assign Azure roles using Azure CLI
- Assign Azure roles using the Azure CLI examples. Azure CLI can also be used in the Azure Cloud Shell experience.
- Assign Azure roles using the Azure portal
- Assign Azure roles using Azure PowerShell.

How it works

By using Windows Admin Center in Azure, you can connect to your cluster without requiring any inbound port to be enabled on the firewall. Windows Admin Center, via the Arc agent, is able to securely establish a reverse proxy session connection with the Azure Arc service in an outbound manner.

For each Azure Local cluster that you want to manage with Windows Admin Center in the Azure portal, you must deploy an agent to all the nodes in the cluster.

The agent communicates to an external service that manages certificates so that you can easily connect to your cluster.

Clicking **Install** performs the following actions:

1. Registers the *Microsoft.HybridConnectivity* resource provider on your subscription. The resource provider hosts the proxy used for communication to your Arc-enabled cluster.
2. Deploys an Azure *endpoint* resource on top of each of your Arc-enabled resources in your cluster that enables a reverse proxy connection on the specified port. This is simply a logical resource in Azure, and doesn't deploy anything on your server itself.
3. Installs the Windows Admin Center agent on your hybrid machine with a valid TLS certificate.

> **Note**

Uninstalling Windows Admin Center does not delete the logical Azure endpoint resource. This is kept for other experiences that might leverage this resource, such as SSH.

Clicking **Connect** performs the following actions:

1. The Azure portal asks the Microsoft.HybridConnectivity resource provider for access to the Arc-enabled server.
2. The resource provider communicates with a Layer 4 SNI proxy to establish a short-lived session-specific access to one of your Arc-enabled nodes of the cluster on the Windows Admin Center port.
3. A unique short-lived URL is generated and connection to Windows Admin Center is established from the Azure portal.

Connection to Windows Admin Center is end-to-end encrypted with SSL termination happening on your cluster.

Automate Windows Admin Center deployment using PowerShell

You can automate Windows Admin Center deployment in Azure portal using this example PowerShell script.

PowerShell

```
$clusterName = "<name_of_cluster>"
$resourceGroup = "<resource_group>"
$subscription = "<subscription_id>"
$port = "6516"

#Deploy Windows Admin Center
$setting = @{ "port" = $port }
New-AzStackHciExtension -ArcSettingName "default" -Name "AdminCenter" -
ResourceGroupName $resourceGroup -ClusterName $clusterName -
ExtensionParameterPublisher "Microsoft.AdminCenter" -
ExtensionParameterSetting $setting -ExtensionParameterType "AdminCenter" -
SubscriptionId $subscription -ExtensionParameterTypeHandlerVersion "0.0"

#Allow connectivity
$patch = @{ "properties" =  @{ "connectivityProperties" = @{"enabled" =
$true}}}
$patchPayload = ConvertTo-Json $patch
Invoke-AzRestMethod -Method PATCH -Uri
"https://management.azure.com/subscriptions/$subscription/resourceGroups/$re
```

```
sourceGroup/providers/Microsoft.AzureStackHCI/clusters/$clusterName/ArcSetti
ngs/default?api-version=2023-02-01" -Payload $patchPayload
```

Troubleshooting

Here are some tips to try in case something isn't working. For general Windows Admin Center troubleshooting (not specifically in Azure), see Troubleshooting Windows Admin Center.

Failed to connect with "404 endpoint not found"

1. Version 1.36 and 1.35 of the Azure Connected Machine Agent (Arc agent) breaks connection to Windows Admin Center. This has been fixed in later versions of the Arc agent (1.37+) This can be downloaded here.

Failed to connect error

1. Restart the HIMDS service.

   a. RDP into each node of your cluster.

   b. Open PowerShell as an administrator and run:

   ```PowerShell
   Restart-Service -Name himds
   ```

2. Make sure that the Windows Admin Center service is running on your cluster.
   a. RDP into each node of your cluster.
   b. Open **Task Manager (Ctrl+Shift+Esc)** and navigate to **Services**.
   c. Make sure **ServerManagementGateway / Windows Admin Center** is running.
   d. If it isn't, start the service.

3. Check that the port is enabled for reverse proxy session.

   a. RDP into each node of your cluster.

   b. Open PowerShell as an administrator and run:

   ```PowerShell
   azcmagent config list
   ```

c. This should return a list of ports under the incomingconnections.ports (preview) configuration that are enabled to be connected from Azure. Confirm that the port on which you installed Windows Admin Center is on this list. For example, if Windows Admin Center is installed on port 443, the result would be:

```
Output

Local configuration setting
incomingconnections.ports (preview): 443
```

d. In the event it isn't on this list, run

```
PowerShell

azcmagent config set incomingconnections.ports <port>
```

If you're using another experience (like SSH) using this solution, you can specify multiple ports separated by a comma.

4. Ensure you have outbound connectivity to the necessary ports.
   a. Each node of your cluster should have outbound connectivity to the following endpoint

   - `*.wac.azure.com`, `*.waconazure.com` or the WindowsAdminCenter ServiceTag
   - `pas.windows.net`
   - `*.servicebus.windows.net`

One of the Windows Admin Center tools isn’t loading or gives an error

1. Navigate to any other tool in Windows Admin Center and navigate back to the one that isn’t loading.

2. If no other tool is loading, there might be a problem with your network connectivity. Try closing the blade and then connecting again. If this doesn’t work, open a support ticket.

The Windows Admin Center extension failed to install

1. Double-check to make sure that the cluster meets the requirements.

2. Make sure that outbound traffic to Windows Admin Center is allowed on each node of your cluster.

    a. Test connectivity by running the following command using PowerShell inside of your virtual machine:

    ```
    PowerShell

    Invoke-RestMethod -Method GET -Uri
    https://<your_region>.service.waconazure.com
    ```

    ```
    Expected

    Microsoft Certificate and DNS service for Windows Admin Center in
    the Azure Portal
    ```

3. If you've allowed all outbound traffic and are getting an error from the command above, check that there are no firewall rules blocking the connection.

If nothing seems wrong and Windows Admin Center still won't install, open a support request with the following information:

- Logs in the Azure portal. This can be found under **Settings** > **Extensions** > **AdminCenter** > **View Detailed Status**.
- Logs on each node of the cluster. Run the following PowerShell command and share the resulting .zip file.

    ```
    PowerShell

    azcmagent logs
    ```

- Network trace, if appropriate. Network traces can contain customer data and sensitive security details, such as passwords, so we recommend reviewing the trace and removing any sensitive details before sharing it.

Known issues

- Chrome incognito mode isn't supported.
- Azure portal desktop app isn't supported.
- Detailed error messages for failed connections aren't available yet.
- Updates isn’t supported. Users cannot apply updates to Azure Local cluster with CAU (Cluster-Aware Updating).

Frequently asked questions

Find answers to the frequently asked questions about using Windows Admin Center in Azure.

How much does it cost to use Windows Admin Center?

There's no cost associated to use the Windows Admin Center in the Azure portal.

Can I use Windows Admin Center to manage the virtual machines running on my cluster?

You can install the Hyper-V role using the Roles and Features extension. Once installed, refresh your browser, and Windows Admin Center will show the Virtual Machine and Switch extensions.

What clusters can I manage using this extension?

You can use the capability to manage Arc-enabled Azure Local clusters, version 21H2 or later. You can also use Windows Admin Center to manage your Arc-enabled servers

How does Windows Admin Center handle security?

Traffic from the Azure portal to Windows Admin Center is end-to-end encrypted. Your Arc-enabled cluster is managed using PowerShell and WMI over WinRM.

Do I need an inbound port to use Windows Admin Center?

No inbound connection is required to use Windows Admin Center.

Why must I create an outbound port rule?

An outbound port rule is required for the service that we have built to communicate with your server. Our service issues you a certificate free-of-cost for your instance of Windows Admin Center. This service ensures that you can always connect to your instance of Windows Admin Center from the Azure portal by keeping your WAC certificate up to date.

Furthermore, accessing Windows Admin Center from Azure requires no inbound port and only outbound connectivity via a reverse proxy solution. These outbound rules are required in order to establish the connection.

How do I find the port used for Windows Admin Center installation?

To verify the value of SmePort registry setting:

1. RDP into your server.
2. Open the **Registry Editor**.
3. Navigate to the key `\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\ServerManagementGateway`.
4. Read the value of `SmePort` to find the port used.

I already have Windows Admin Center installed on one or all nodes of my cluster. Can I access it from the portal?

Yes. You can follow the same steps outlined in this document.

> ⚠ **Warning**
>
> Enabling this capability will replace your existing instance of Windows Admin Center and removes the capability to manage other machines. Your previously deployed instance of Windows Admin Center will no longer be usable.

Next steps

- Learn about Windows Admin Center
- Learn about managing servers with Windows Admin Center
- Learn about Azure Local
- Learn about connecting Azure Local to Azure

Feedback

Was this page helpful? Yes No

Provide product feedback

What's new with Windows Admin Center Azure extension

Article • 12/19/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

The Windows Admin Center Azure extension (seen as "AdminCenter" in Azure) receives improvements on an ongoing basis. To stay up to date with the most recent developments, this article provides you with information about:

- The latest releases
- Known issues
- Bug fixes

This page is updated monthly, so revisit it regularly.

> **Note**
>
> This article is only for Windows Admin Center in Azure. For release notes on Windows Admin Center on-premises, navigate to release history.

Version 0.31.0.0 - June 2024

New features

- Support for Windows 10/11 machines.

Fixed

- Fixed an issue where connection to Azure Local clusters fails.

Version 0.24.0.0 - April 2024

New features

- Updated the build of Windows Admin Center to our **new** modernized gateway.
- Significant agent upgrade from .NET 4.6.2 to .NET Core.
- Switch to a multi-process, micro-service based architecture, allowing Admin Center to be more flexible, scalable, and resilient.

- Shift from Kantana to Kestrel HTTP web server.

Fixed

- Fixed an issue where updating between versions fails because certificates don't get imported
- Fixed an issue where installation would fail on WDAC-enforced devices.
- Fixed an issue where connection would fail due to a new Chromium policy impacting 1% of Edge/Chrome users.

Version 0.0.0.340 - March 2024

Fixed

- Fixed an issue where the Remote Desktop tool was not working or resizing it would make it unusable.
- Fixed an issue where the text in some languages was garbled.

Version 0.0.0.332 - December 2023

New features

- Updated the build of Windows Admin Center to v2311. For a full summary of new features, visit our blog post.
- Added support for installation when TLS 1.3 is enabled.

Fixed

- Fixed an issue where momentarily losing access to IMDS/HIMDS does not result in Windows Admin Center installation failure.

Version 0.0.0.329 - November 2023

Fixed

- Removed the "Unrestricted" execution policy when Windows Admin Center scripts are executed.
- Added more logging when proxies are used for better debugging.

Version 0.0.0.324 - September 2023

Fixed

- Fixed an issue where Windows Admin Center's certificate validation would not cause Windows Admin Center to fail if a valid certificate is already available. This issue would show up during server reboots when the "Enable" operation is run.

Version 0.0.0.323 - August 2023

Fixed

- Fixed an issue where an additional URL was required when using proxies.
- Updated the build of Windows Admin Center to the latest preview version.

Version 0.0.0.322 - July 2023

New features

- Introduced support for proxies. You can now configure a proxy to monitor/filter all outbound traffic from the AdminCenter extension.
- Updated the build of Windows Admin Center to v2306. For a full summary of new features, visit our blog post.
- Introduced new management capabilities for Azure Local - Remote Support and Diagnostics.

Version 0.0.0.313 - April 2023

New features

- Introduced Microsoft Entra authentication for Azure Local. Note that this functionality is in preview. Please create a support ticket for all issues.
- Introduced a **Network** tool.
- Introduced Virtual Machine Live Storage Migration.
- Introduced new security experiences for Azure Local clusters with the supplemental package.

Fixed

- If a prior installation of Windows Admin Center exists on your machine, the extension will first uninstall it before installing Windows Admin Center in Azure.
- Fixed the infinite loading of the Remote Desktop tool.

Version 0.0.0.228 - January 2023

New features

- Updated the build of Windows Admin Center to v2211. A few highlights are listed. For a full summary, visit our blog post.
  - Support for WDAC-enforced infrastructure.
  - Support for 400% zoom.
  - Search settings with smart keywords.
  - Azure Local management improvements.

Version 0.0.0.224 - December 2022

Fixed

- Fixed Microsoft Entra authentication when managing domain controllers. Microsoft Entra authentication isn't supported on domain controllers and users must enter local administrator credentials.

Version 0.0.0.221 - October 2022

New features

- Introduced Microsoft Entra authentication for Windows Server Azure Virtual Machines and Arc-enabled Servers. Azure Local doesn't support Microsoft Entra authentication yet.
- Windows Admin Center for Azure Virtual Machines is now generally available. Windows Admin Center for Arc-enabled servers and Azure Local remains in Public Preview.

Feedback

Was this page helpful? Yes No

Provide product feedback

Windows Admin Center release history

Article • 12/05/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Here's a listing of our latest released features:

- Version 2410 is the most recently available (GA) release - it includes an upgrade of the backend from .NET 4.6.2 to .NET 8, improvements to the virtual machines and security tools, and many improvements and bugfixes.
- Version 2311 includes an Angular 15 upgrade, improvements to the Import VM experience, Azure Arc at-scale onboarding, and a new Azure Migrate assessment experience.
- Version 2306 includes the WDAC-enabled infra GA, new Hyper-V features and improvements, the long awaited cluster-aware event viewer, as well as a plethora of bug fixes and improvements.
- Version 2211 includes support for 400% zoom and WDAC-enabled infrastructure, improvements to Azure Local management features, as well as bug fixes and updates in several extensions.
- Version 2110.2 includes major bug fixes to Role-based Access Control (RBAC), the connections page's search feature, as well as bug fixes in several extensions.
- Version 2110 includes Angular 11 upgrades, performance and security enhancements. It also includes updated developer SDK for extension development using Angular 11 or upgrading of existing extensions, CredSSP enhancements, a better Virtual Machine tool, and two brand new tools for Security and GPU management.
- Version 2103.2 includes key bug fixes and feature updates to the Azure sign in process, support for Azure China, support for seamless over-the-air updates for Azure Local as well as additional updates to the Events and Remote Desktop tool experience.
- Version 2103 introduces automatic platform an extension updates and includes updates to several of our core tools like the VM tool and Events.
- Version 2009 includes support for Azure Kubernetes Service on Azure Local and major updates to the Virtual machines, File shares, and Containers tools.
- Version 2007 includes support for the new Azure Local and new features for several tools.
- Version 1910.2 includes updates to the platform's accessibility and numerous bug fixes
- Version 1910 introduces several new Azure hybrid services and brings features that were previously in preview to the GA channel.
- Version 1909 introduces the Azure VM specific connection type and unifies the connection types for traditional failover clusters and HCI clusters.

- Version 1908 added visual updates, Packetmon, FlowLog Audit, Azure Monitor onboarding for clusters, and support for WinRM over HTTPS (port 5986.)
- Version 1907 added Azure cost estimate links and made improvements to import/export and tagging of virtual machines.
- Version 1906 added import/export VMs, switch Azure accounts, add connections from Azure, connectivity settings experiment, performance improvements, and performance profiling tool.
- Version 1904.1 was a maintenance update to improve stability of gateway plugins.
- Version 1904 was a GA release that introduced the Azure Hybrid Services tool, and brought features that were previously in preview to the GA channel.
- Version 1903 added email notifications from Azure Monitor, the ability to add Server or PC connections from Active Directory, and new tools to manage Active Directory, DHCP, and DNS.
- Version 1902 added a shared connection list & improvements to software defined network (SDN) management, including new SDN tools to manage ACLs, gateway connections, and logical networks.
- Version 1812 added dark theme (in preview), power configuration settings, BMC info, and PowerShell support to manage extensions and connections.
- Version 1809.5 was a GA cumulative update that included various quality and functional improvements, bug fixes throughout the platform, and a few new features in the hyper-converged infrastructure management solution.
- Version 1809 was a GA release that brought features that were previously in preview to the GA channel.
- Version 1808 added Installed Apps tool, lots of under the hood improvements, and major updates to the preview SDK.
- Version 1807 added a streamlined Azure connect experience, improvements to VM inventory page, file sharing functionality, Azure update management integration, and more.
- Version 1806 added show PowerShell script, SDN management, 2008 R2 connections, SDN, scheduled tasks, and many other improvements.
- Version 1804.25 - a maintenance update to support users installing Windows Admin Center in completely offline environments.
- Version 1804 - Project Honolulu becomes Windows Admin Center and adds security features and role-based access control. Our first GA release.
- Version 1803 added support for Microsoft Entra access control, detailed logging, resizable content, and a bunch of tool improvements.
- Version 1802 added support for accessibility, localization, high-availability deployments, tagging, Hyper-V host settings, and gateway authentication.
- Version 1712 added more virtual machine features and performance improvements throughout the tools.

- Version 1711 added highly anticipated tools (Remote Desktop and PowerShell) along with other improvements.

Feedback

Was this page helpful? Yes No

Provide product feedback

Windows Admin Center support policy

Article • 11/01/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Windows Admin Center (non-preview) releases are supported continuously, based on Microsoft's Modern Lifecycle Policy. This means that only the latest version of Windows Admin Center is serviced and supported, and users must stay current by upgrading to the latest Windows Admin Center release within 30 days of availability to remain supported. This policy applies to both the Windows Admin Center platform itself, as well as any released (non-preview) Microsoft extensions published in the Windows Admin Center extension feed. Note that some extensions may be updated more frequently than others, between Windows Admin Center releases.

For info about Windows Admin Center Preview releases, see Windows Insider Preview releases.

Feedback

Was this page helpful? Yes No

Provide product feedback

Troubleshoot Windows Admin Center

Article • 09/23/2022 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

This article describes how to diagnose and resolve issues in Windows Admin Center. If you're having an issue with a specific tool, check to see if you're experiencing a known issue.

Installer fails with message: *The Module 'Microsoft.PowerShell.LocalAccounts' could not be loaded.*

This failure can happen if your default PowerShell module path has been modified or removed. To resolve the issue, make sure that `%SystemRoot%\system32\WindowsPowerShell\v1.0\Modules` is the first item in your PSModulePath environment variable. You can achieve this with the following line of PowerShell:

PowerShell

```
[Environment]::SetEnvironmentVariable("PSModulePath","%SystemRoot%\system32\WindowsPowerShell\v1.0\Modules;" + ([Environment]::GetEnvironmentVariable("PSModulePath","User")),"User")
```

I get a This site/page can't be reached error in my web browser

If you've installed Windows Admin Center as an App on Windows 10

- Check to make sure Windows Admin Center is running. Look for the Windows Admin Center icon  in the System tray or **Windows Admin Center Desktop / SmeDesktop.exe** in Task Manager. If not, launch **Windows Admin Center** from the Start menu.

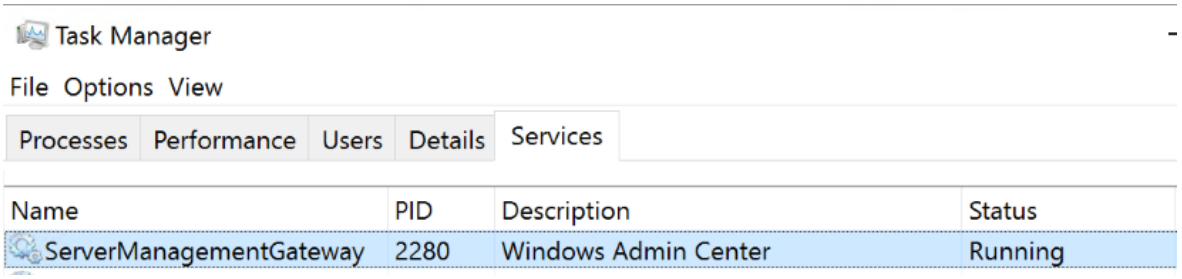
> ⓘ Note

After rebooting, you must launch Windows Admin Center from the Start menu.

- Check the Windows version.
- Make sure you're using either Microsoft Edge or Google Chrome as your web browser.
- Did you select the correct certificate on first launch?
  - Try opening your browser in a private session - if that works, you'll need to clear your cache.
- Did you recently upgrade Windows 10 to a new build or version?
  - This may have cleared your trusted hosts settings. Follow these instructions to update your trusted hosts settings.

If you've installed Windows Admin Center as a Gateway on Windows Server

- Check the Windows version of the client and server.
- Make sure you are using either Microsoft Edge or Google Chrome as your web browser.
- On the server, open Task Manager > Services and make sure **ServerManagementGateway / Windows Admin Center** is running.



- Test the network connection to the Gateway (replace <values> with the information from your deployment)

```
PowerShell

Test-NetConnection -Port <port> -ComputerName <gateway> -
InformationLevel Detailed
```

If you have installed Windows Admin Center in an Azure Windows Server VM

- Check the Windows version.
- Did you add an inbound port rule for HTTPS?
- Learn more about installing Windows Admin Center in an Azure VM.

Check the Windows version

To check the Windows version:

1. Open the run dialog (Windows Key + R) and launch `winver`.
2. Check the version in the **About Windows** window.
    - If you're using Windows 10 version 1703 or earlier, Windows Admin Center isn't supported on your version of Microsoft Edge. Either upgrade to a recent version of Windows 10 or use Google Chrome.
    - If you're using an insider preview version of Windows 10 or Server with a build version between 17134 and 17637, Windows had a bug that caused Windows Admin Center to fail. Use a current supported version of Windows to fix this issue.

Make sure the Windows Remote Management (WinRM) service is running on both the gateway machine and managed node

1. Open the run dialog with WindowsKey + R.
2. Type `services.msc` and press Enter.
3. In the window that opens, look for Windows Remote Management (WinRM), make sure it is running and set to automatically start.

If you're getting WinRM error messages while managing servers in Windows Admin Center

WinRM doesn't allow credential delegation by default. To allow delegation, the computer needs to have Credential Security Support Provider (CredSSP) enabled temporarily.

If you're receiving WinRM error messages, try using the verification steps in the Manual troubleshooting section of Troubleshoot CredSSP to resolve them.

Did you upgrade your server from 2016 to 2019?

This may have cleared your trusted hosts settings. Follow these instructions to update your trusted hosts settings.

I get the message: "Can't connect securely to this page. This might be because the site uses outdated or unsafe TLS security settings."

Your machine is restricted to HTTP/2 connections. Windows Admin Center uses integrated Windows authentication, which is not supported in HTTP/2. Add the following two registry values under the `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Http\Parameters` key on **the machine running the browser** to remove the HTTP/2 restriction:

Windows Command Prompt

```
EnableHttp2Cleartext=dword:00000000
EnableHttp2Tls=dword:00000000
```

I'm having trouble with the Remote Desktop, Events, and PowerShell tools.

These three tools require the web socket protocol, which is commonly blocked by proxy servers and firewalls. If you're using Google Chrome, there's a known issue with web sockets and NTLM authentication.

I can connect to some servers, but not others

- Log on to the gateway machine locally and try to `Enter-PSSession <machine name>` in PowerShell, replacing <machine name> with the name of the Machine you're trying to manage in Windows Admin Center.
- If your environment uses a workgroup instead of a domain, see using Windows Admin Center in a workgroup.
- **Using local administrator accounts:** If you're using a local user account that isn't the built-in administrator account, you need to enable the policy on the target

machine by running the following command in PowerShell or at a command prompt as Administrator on the target machine:

```
Windows Command Prompt

REG ADD HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System
/v LocalAccountTokenFilterPolicy /t REG_DWORD /d 1
```

I get this message: "You are not authorized to view this page. If you recently updated Windows Admin Center, you may need to restart your browser, and then refresh the page."

Make sure to select the **Windows Admin Center Client** certificate when prompted on the first launch, and not any other certificate. If you select any other certificate, you'll get this error message. To resolve this error, restart your browser and refresh the page, and select the **Windows Admin Center Client** certificate. If you continue to get the same error, try clearing the browser cache or switching to another browser. If none of these troubleshooting steps resolve the issue, you may need to uninstall and reinstall Windows Admin Center, and then restart it.

Using Windows Admin Center in a workgroup

What account are you using?

Make sure the credentials you're using are a member of the target server's local administrators group. In some cases, WinRM also requires membership in the Remote Management Users group. If you're using a local user account that is **not the built-in administrator account**, you will need to enable the policy on the target machine by running the following command in PowerShell or at a Command Prompt as Administrator on the target machine:

```
Windows Command Prompt

REG ADD HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System /v
LocalAccountTokenFilterPolicy /t REG_DWORD /d 1
```

Are you connecting to a workgroup machine on a different subnet?

To connect to a workgroup machine that isn't on the same subnet as the gateway, make sure the firewall port for WinRM (TCP 5985) allows inbound traffic on the target machine. You can run the following command in PowerShell or at a Command Prompt as Administrator on the target machine to create this firewall rule:

- **Windows Server**

```
PowerShell

Set-NetFirewallRule -Name WINRM-HTTP-In-TCP-PUBLIC -RemoteAddress Any
```

- **Windows 10**

```
PowerShell

Set-NetFirewallRule -Name WINRM-HTTP-In-TCP -RemoteAddress Any
```

Configure TrustedHosts

When installing Windows Admin Center, you're given the option to let Windows Admin Center manage the gateway's TrustedHosts setting. This is required in a workgroup environment, or when using local administrator credentials in a domain. If you choose to forego this setting, you must configure TrustedHosts manually.

To modify TrustedHosts using PowerShell commands:

1. Open an Administrator PowerShell session.
2. View your current TrustedHosts setting:

```
PowerShell

Get-Item WSMan:\localhost\Client\TrustedHosts
```

> **⚠ Warning**
>
> If the current setting of your TrustedHosts is not empty, the commands below will overwrite your setting. We recommend that you save the current setting to a text file with the following command so you can restore it if needed:

```
Get-Item WSMan:localhost\Client\TrustedHosts | Out-File
C:\OldTrustedHosts.txt
```

3. Set TrustedHosts to the NetBIOS, IP, or FQDN of the machines you intend to manage:

```PowerShell
Set-Item WSMan:localhost\Client\TrustedHosts -Value
'192.168.1.1,server01.contoso.com,server02'
```

> **Tip**
>
> For an easy way to set all TrustedHosts at once, you can use a wildcard.
>
> ```PowerShell
> Set-Item WSMan:\localhost\Client\TrustedHosts -Value '*'
> ```

4. When you are done testing, you can issue the following command from an elevated PowerShell session to clear your TrustedHosts setting:

```PowerShell
Clear-Item WSMan:localhost\Client\TrustedHosts
```

5. If you had previously exported your settings, open the file, copy the values, and use this command:

```PowerShell
Set-Item WSMan:localhost\Client\TrustedHosts -Value '<paste values from
text file>'
```

I previously had Windows Admin Center installed, and now nothing else can use the same TCP/IP port

Manually run these two commands in an elevated command prompt:

Windows Command Prompt

```
netsh http delete sslcert ipport=0.0.0.0:443
netsh http delete urlacl url=https://+:443/
```

Azure features don't work properly in Microsoft Edge

Microsoft Edge has known issues related to security zones that affect Azure login in Windows Admin Center.

If you are having trouble using Azure features when using Microsoft Edge, perform these steps to add the required URLs:

1. Search for **Internet Options** in the Windows Start menu.
2. Go to the **Security** tab.
3. Under the **Trusted sites** option, click on the **Sites** button and add the following URLs in the dialog box that opens:
   - Your gateway URL
   - `https://login.microsoftonline.com`
   - `https://login.live.com`
4. Click **Close** and then click **OK**.
5. Update the **Pop-up Blocker** settings in Microsoft Edge:

   a. Browse to edge://settings/content/popups?search=pop-up.

   b. Under the **Allow** section, add the following URLs:

   - Your gateway URL
   - `https://login.microsoftonline.com`
   - `https://login.live.com`

Have an issue with an Azure-related feature?

Send us an email at wacFeedbackAzure@microsoft.com with the following information:

- General issue information from the questions listed below.
- Describe your issue and the steps you took to reproduce the issue.

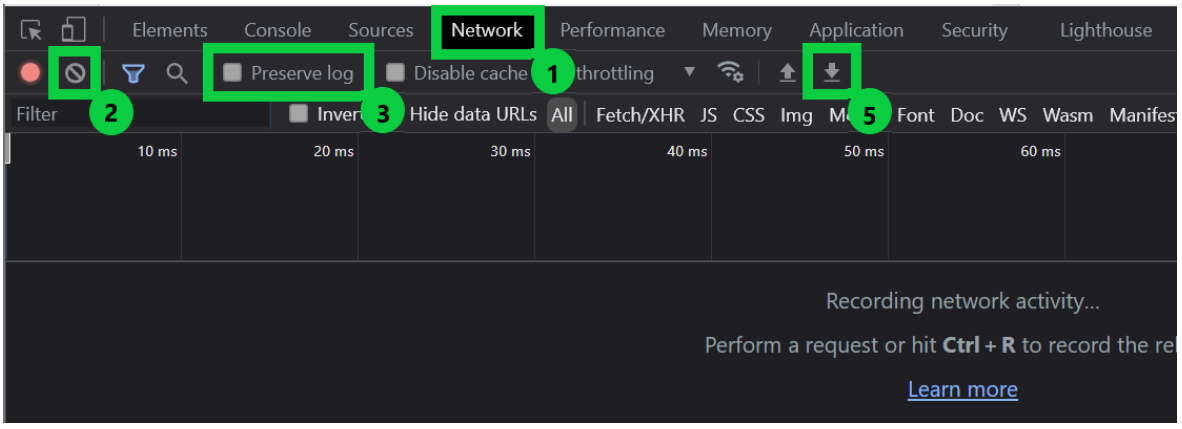
- Did you previously register your gateway to Azure using the New-AadApp.ps1 downloadable script and then upgrade to version 1807? Or did you register your gateway to Azure using the UI from gateway Settings > Azure?
- Is your Azure account associated with multiple directories/tenants? If yes, when registering the Microsoft Entra application to Windows Admin Center, was the directory you used your default directory in Azure?
- Does your Azure account have access to multiple subscriptions?
- Does the subscription you were using have billing attached?
- Were you logged in to multiple Azure accounts when you encountered the issue?
- Does your Azure account require multi-factor authentication?
- Is the machine you're trying to manage an Azure VM?
- Is Windows Admin Center installed on an Azure VM?

Collect HAR files

An HTTP Archive Format (HAR) file is a log of a web browser's interaction with a site. This information is crucial for troubleshooting and debugging.

To collect a HAR file in Microsoft Edge or Google Chrome, follow these steps:

1. Press **F12** to open Developer Tools window, and then click the **Network** tab.
2. Select the **Clear** icon to clean up network log.
3. Click to select the **Preserve Log** check box.
4. Reproduce the issue.
5. After reproducing the issue, click on **Export HAR**.
6. Specify where to save the log and click **Save**.



> ⚠ **Warning**

Before sharing your HAR files with Microsoft, ensure that you remove or obfuscate any sensitive information, like passwords.

Provide feedback on issues

Go to Event Viewer > Application and Services > Microsoft-ServerManagementExperience and look for any errors or warnings.

File a bug on GitHub that describes your issue.

Include any errors or warning you find in the event log, and the following information:

- Platform where Windows Admin Center is **installed** (Windows 10 or Windows Server):
  - If installed on Server, what is the Windows version of **the machine running the browser** to access Windows Admin Center:
  - Are you using the self-signed certificate created by the installer?
  - If you're using your own certificate, does the subject name match the machine?
  - If you're using your own certificate, does it specify an alternate subject name?
- Did you install with the default port setting?
  - If not, which port did you specify?
- Is the machine where Windows Admin Center is **installed** joined to a domain?
- Windows version where Windows Admin Center is **installed**:
- Is the machine that you're **trying to manage** joined to a domain?
- Windows version of the machine that you're **trying to manage**:
- What browser are you using?
  - If you're using Google Chrome, what is the version? (Help > About Google Chrome)

Feedback

Was this page helpful? Yes No

Provide product feedback

Windows Admin Center known issues

Article • 06/05/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

If you encounter an issue not described on this page, let us know at the Windows Admin Center feedback page.

Installer

- When you install Windows Admin Center using your own certificate, if you copy the thumbprint from the certificate manager Microsoft Management Center (MMC) tool, when you paste it, it contains an invalid character at the beginning. As a workaround, enter the first character of the thumbprint, then copy and paste the characters that come after the first.
- Windows Admin Center doesn't support ports lower than 1024. In service mode, you can optionally configure port 80 to redirect to your specified port.

General

- Windows Admin Center version 2410 build 2.4.2.1 may appear as build number 2.4.1 on the Windows Admin Center Updates settings page.
- In Windows Admin Center version 2410, users are unable to register their Windows Admin Center gateways to sovereign clouds. This issue is actively being investigated.
- Self-signed certificates accessed on `https://localhost:[port]` can cause the Microsoft Edge and Google Chrome browsers to block Windows Admin Center. When you're blocked, you should see an error message that says your connection isn't private. To resolve this issue, update Windows Admin Center to the latest version.
- Using certain versions of extensions with earlier versions of Windows Admin Center can result in icons not displaying properly. To resolve this issue, update to the latest version of Windows Admin Center
- Manually modifying URLs to include the names of different machines while using Windows Admin Center without going through the connection experience in the UI can cause extensions to not load properly, especially extensions compatible with specific hardware. We don't recommend manually modifying URLs for navigation in Windows Admin Center.

- If you have Windows Admin Center installed as a heavily used gateway on Windows Server 2016, the service can crash and display an error in the event log that contains `Faulting application name: sme.exe` and `Faulting module name: WsmSvc.dll`. This error happens because of a bug that we've fixed as of Windows Server 2019. However, we've also released a patch for Windows Server 2016 to address this issue in the February 2019 cumulative update, KB4480977.

- If you have Windows Admin Center installed as a gateway and your connection list appears to be corrupted, follow these steps:

  > ⚠ **Warning**
  >
  > The procedure in these instructions deletes the connection list and settings for all Windows Admin Center users on the gateway.

  1. Uninstall Windows Admin Center.
  2. Go to **C:\Windows\ServiceProfiles\NetworkService\AppData\Roaming\Microsoft** and delete the **Server Management Experience** folder.
  3. Reinstall Windows Admin Center.

- If you leave the tool open and idle for a long period of time, several error messages appear that say "The runspace state is not valid for this operation." If this issue occurs, refresh your browser. If you encounter this error, send us feedback.

- There can be minor differences between version numbers of open-source software (OSS) running in Windows Admin Center modules versus what's listed in the third Party Software Notice.

- You can access and use Windows Admin Center tool application programming interfaces (APIs) through other methods during an active session of Windows Admin Center if you have access to that session. The actions you take using these APIs only affect the machine you installed Windows Admin Center on, also known as the gateway machine. They don't affect machines managed remotely without authentication through the Windows Admin Center gateway.

- Some APIs used by Windows Admin Center, including the DeploymentShare API, require the user to be a local administrator. Network share creation operations cannot be performed by a standard Windows user by default. Windows Admin Center cannot elevate a standard user account to administrator. Adding the user to

the "Gateway administrators" group in Settings only changes the permissions the user has within the gateway, not on the system.
    - You may not run into this issue on modernized gateway builds of Windows Admin Center. By default, modernized gateway builds utilize a form login to access the gateway, which does not have the local administrator restriction. Existing versions of Windows Admin Center utilize NTLM/Kerberos, which obtains a token limited to the localhost environment. NTLM/Kerberos login is also available on modernized gateway builds.

- Windows Admin Center does not support authenticating guest users of Microsoft Entra ID tenants. As a result, guest users of Microsoft Entra ID tenants will no longer be able to connect their Windows Admin Center instance to Azure or use Azure services like Azure Arc, Azure Site Recovery, Azure File Sync, etc.

Extension Manager

- When you update Windows Admin Center, you must reinstall your extensions.
- If you add an extension feed that is inaccessible, no warning or error message appears.

Partner extension issues

Dell's EMC OpenManage Integration extension utilizes APIs provided by Windows Admin Center to push files onto target nodes. APIs such as NodeExtensionInstall only work when the user is a gateway administrator; it doesn't support non-admin use.

Browser-specific issues

This section describes issues that can happen when you use Windows Admin Center in an internet browser.

Microsoft Edge

If you have Windows Admin Center deployed as a service and you're using Microsoft Edge as your browser, you might not be able to connect your gateway to Azure after opening a new browser window. There isn't currently a solution for this issue, but you can work around it by adding `https://login.microsoftonline.com`, `https://login.live.com`, and the URL of your gateway as trusted sites and allowed sites for pop-up blocker settings on your client-side browser.

For more information, see the troubleshooting guide.

Google Chrome

- Before version 70, Chrome had a bug that affected the WebSockets protocol and Windows New Technology Local Area Network Manager (NTLM) authentication. This bug also affects the following programs:
  - Windows Events
  - PowerShell
  - Remote Desktop
- Many credential prompts might appear while you're using Chrome, especially when you're adding connections in a workgroup environment.
- If you have Windows Admin Center deployed as a service, you must enable popups from the gateway URL to use Azure integration.

Mozilla Firefox

- Windows Admin Center isn't tested with Mozilla Firefox, but most functionality should work.
- If you're using Windows 10, you need to import the Windows Admin Center Client certificate into Firefox to use Windows Admin Center.

WebSocket compatibility when using a proxy service

Scenarios involving using Windows Admin Center with a proxy service often don't support the WebSocket protocol, which can affect the following programs:

- Remote Desktop
- PowerShell
- Packet Monitoring
- Windows Events

Events

When you export large log files, you can sometimes receive an error message about packet size.

To resolve this issue:

1. Open an elevated command prompt on the gateway machine.
2. Run the following command:

Windows Command Prompt

```
winrm set winrm/config @{MaxEnvelopeSizekb="8192"}
```

Remote Desktop

- When you deploy Windows Admin Center as a service, the Remote Desktop tool sometimes doesn't load after the Windows Admin Center service updates to a new version. To work around this issue, clear your browser cache.
- The Remote Desktop tool sometimes doesn't connect when managing Windows Server 2012.
- When using the Remote Desktop to connect to a machine that isn't Domain joined, you must enter your account in the `MACHINENAME\USERNAME` syntax.
- Some configurations can block Windows Admin Center's remote desktop client with group policy. If you're blocked by this issue, open the **Local Group Policy Editor** and reconfigure the **Computer Configuration\Administrative Templates\Windows Components\Remote Desktop Services\Remote Desktop Session Host\Connections** Group Policy Object (GPO).
- The Remote Desktop tool doesn't currently support any text, image, or file copy and paste between the local desktop and the remote session.
- You can copy text the same way you would during a local session by either right-clicking and selecting **Copy** or pressing the **Ctrl+C** keys, but you can only paste by right-clicking and selecting **Paste**.
- Remote sessions don't support the following keys and keyboard shortcuts:
  - Alt+Tab
  - Function keys
  - Windows Key

    - PrtScn

- When using Remote Desktop to connect to a machine, keyboard language mapping may not work properly.

Support for Windows Server 2012 R2, 2012, and 2008 R2

Windows Admin Center requires PowerShell features that aren't included in Windows Server 2012 R2, 2012, or 2008 R2. If you plan to manage Windows Server with Windows Admin Center, you must install Windows Management Framework (WMF) version 5.1 or later on those servers.

To install WMF:

1. Open an elevated PowerShell window.
2. Enter `$PSVersiontable` to verify if you have WMF installed and check the version number.
3. Download and install WMF if you haven't already.

Role Based Access Control (RBAC)

- RBAC can't deploy on machines configured to use Windows Defender Application Control (WDAC).
- To use RBAC in a cluster, you must deploy the configuration to each member node individually.
- When you deploy RBAC, you may get unauthorized errors incorrectly attributed to the RBAC configuration.

Server Manager solution

This section describes common issues you can run into in Server Manager on Windows Admin Center.

Certificates

Server Manager on Windows Admin Center doesn't currently support importing the .PFX Encrypted Certificate into the current user store.

Files

Windows Admin Center doesn't currently support uploading or downloading files over 100 MB in size.

PowerShell

- The issue described in WebSocket compatibility when using a proxy service affects PowerShell.
- PowerShell in Server Manager doesn't support pasting into the window by right-clicking. To paste into the window, you need to right-click and select **Paste** from the drop-down context menu or use the **Ctrl+V** shortcut.
- PowerShell in Server Manager doesn't support the Ctrl+C shortcut to copy content to the clipboard. To copy content, highlight the text, right-click it, then select **Copy**.
- When you make the Windows Admin Center window smaller, the terminal content adjusts to fit the new window size. When you return the window to its original size, the content might not return to its original state. You can restore the text by using the `Clear-Host` command, or disconnect and reconnect using the button above the terminal.

Registry Editor

Registry Editor for Windows Admin Center for Windows Server hasn't implemented search functionality.

Roles and Features

- When you select roles or features that don't have available installation sources, the system skips them.
- If you choose to not automatically restart after you install a role, you won't see any more notification messages asking you to restart.
- If you do choose to automatically reboot, the reboot occurs before the status bar reaches 100%.

Storage

- DVD, CD, and Floppy drives don't appear as volumes on down-level.
- Some properties in Volumes and Disks appear as unknown or blank in the Details panel because they aren't available in down-level storage.
- If you're creating a new Resilient File System (ReFS) volume, ReFS only supports an allocation unit size of 64K on Windows 2012 and 2012 R2 machines. If you create a ReFS volume with a smaller allocation unit size on down-level targets, file system formatting doesn't work, making the new volume unusable. To resolve this issue, delete the unusable volume, then create a new one with 64K allocation unit size.

Updates

After the system installs updates, it sometimes caches the install status and requires a browser refresh. If you see an error message that says "Keyset does not exist" when attempting to set up Azure Update management, follow these directions on the managed node:

1. Stop the **Cryptographic Services** service.
2. Change the folder options to show hidden files, if necessary.
3. Go to the **%allusersprofile%\Microsoft\Crypto\RSA\S-1-5-18** folder and delete all its contents.
4. Restart the **Cryptographic Services** service.
5. Reinstall Update Management with Windows Admin Center.

Virtual machines

- If you're managing your virtual machines (VMs) on a Windows Server 2012 session host, the in-browser VMConnect tool can't connect to the VM. You can resolve this issue by downloading the .rdp file to connect to the VM.
- If you've set up Azure Site Recovery on a host outside of Windows Admin Center, it can't protect VMs from inside Windows Admin Center.
- Windows Admin Center doesn't currently support advanced features available in Hyper-V Manager, such as Virtual SAN Manager, Move VM, Export VM, and VM Replication.

Virtual switches

When you add network interface controllers (NICs) to a team for switch-embedded teaming (SET), you must make sure they're on the same subnet.

Computer Management solution

The Computer Management solution contains some Server Manager tools, so the same known issues that apply to Server Manager apply here. We're aware of the following Computer Management solution-specific issues:

- If you sign in to your Windows 10 device with a Microsoft Account (MSA) or Microsoft Entra ID, you must use manage-as to provide credentials for a local administrator account.
- When you try to manage the local host, a message appears telling you to elevate the gateway process. If you select **No** in the User Account Control window that appears, you must cancel the connection attempt and start over.
- Windows 10 has WinRM and PowerShell remoting disabled by default.
  - To enable management of the Windows 10 Client, open an elevated PowerShell prompt and run the `Enable-PSRemoting` cmdlet.
  - You should also update your firewall to allow connections from outside the local subnet by running `Set-NetFirewallRule -Name WINRM-HTTP-In-TCP -RemoteAddress Any`. For more information about how to update your firewall in more restrictive network scenarios, see Enable PSRemoting.

Cluster deployment

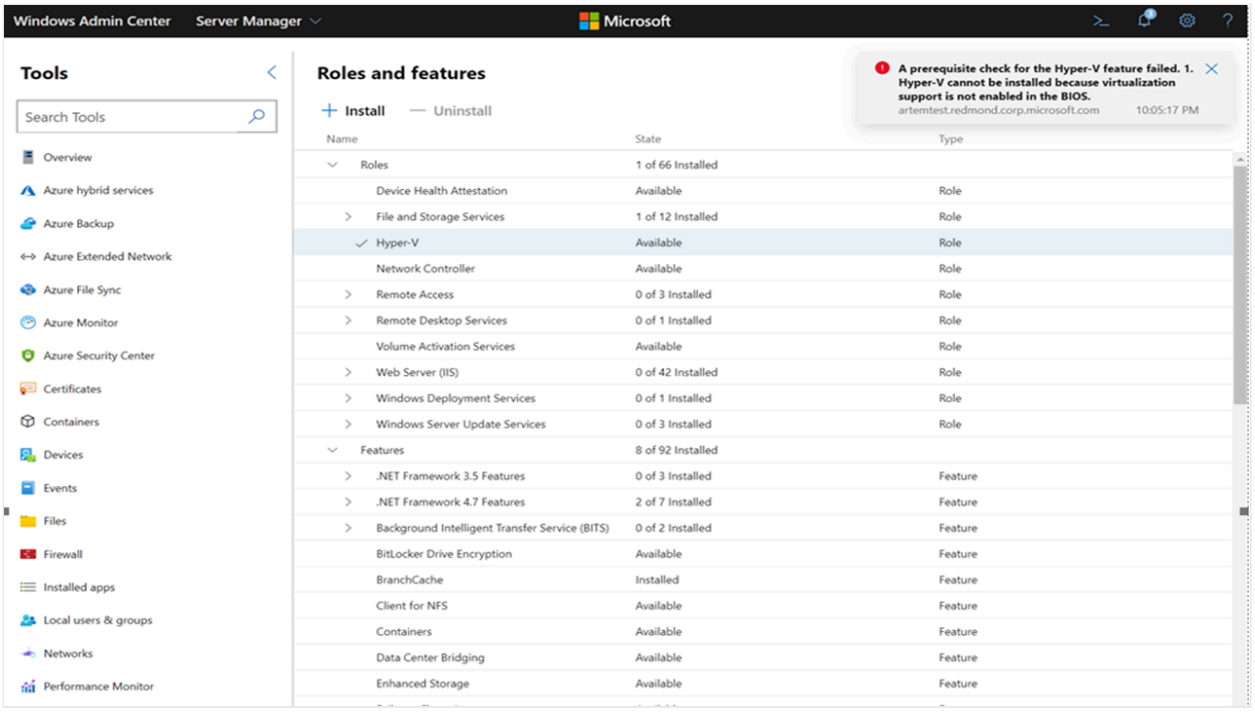
This section describes known issues that affect cluster deployment.

Adding servers to cluster groups

Windows Admin Center doesn't currently support scenarios with mixed work group machines when adding servers. All machines you add to cluster groups must be part of the same work group. If they aren't, an error message appears that says "Cannot create a cluster with servers in different Active Directory domains. Verify the server names are correct. Move all the servers into the same domain and try again." You can't proceed with setting up the cluster unless you use machines from the same work group.

Enabling Hyper-V on VMs

You can only install and enable Hyper-V on VMs running Azure Local. Trying to enable Hyper-V on VMs without Azure Local generates an error message that says "A prerequisite check for the Hyper-V feature failed," as shown in the following screenshot.



To install Hyper-V on VMs running Azure Local, open an elevated PowerShell prompt and run the following command:

```PowerShell
Enable-WindowsOptionalFeature -Online -FeatureName 'Microsoft-Hyper-V'
```

Server restart time after updates

At times, servers may take longer than anticipated to restart after installing updates. To determine if the server has restarted successfully, the Windows Admin Center cluster deployment wizard periodically checks the server's restart state. However, if the user manually restarts the server outside of the wizard, the wizard is unable to capture the server state in a suitable manner.

To work around this issue, close the cluster deployment wizard before manually restarting the server. Once you've restarted the server, you can open the cluster deployment wizard again.

Storage error after deleting a cluster

If you delete a cluster, you can encounter an error if you haven't cleared the storage pools from the deleted cluster. The deleted cluster object locks the storage pools, so you must manually clear the pools.

If you've already encountered this error message, here's how to clear the deleted cluster object from the storage pools:

1. Open an elevated PowerShell window.
2. On all nodes, run the following command:

```PowerShell
Clear-ClusterNode
```

3. Next, remove all previous storage pools by running the following command:

```PowerShell
Get-StoragePool -IsPrimordial 0 | Remove-StoragePool
```

4. If you've configured the storage pools to be read-only, then you must change the storage pools to write mode before removing them by running the following command:

```PowerShell
Get-StoragePool <PoolName> | Set-StoragePool -IsReadOnly $false
```

If you haven't encountered this error but want to avoid it, follow these instructions.

1. Open an elevated PowerShell window.
2. Run this command to remove the virtual disk:

```PowerShell
Get-VirtualDisk | Remove-VirtualDisk
```

3. Next, Run this command to remove the storage pools:

```PowerShell
Get-StoragePool -IsPrimordial 0 | Remove-StoragePool
```

4. After that, run this command to remove resources associated with the cluster:

```
PowerShell

Get-ClusterResource | ? ResourceType -eq "virtual machine" | Remove-
ClusterResource
Get-ClusterResource | ? ResourceType -like "*virtual machine*" |
Remove-ClusterResource
```

5. Now, run this command to clean up:

```
PowerShell

Remove-Cluster -CleanupAD
```

6. Finally, run this command on all nodes:

```
PowerShell

Clear-ClusterNode
```

Stretch cluster creation

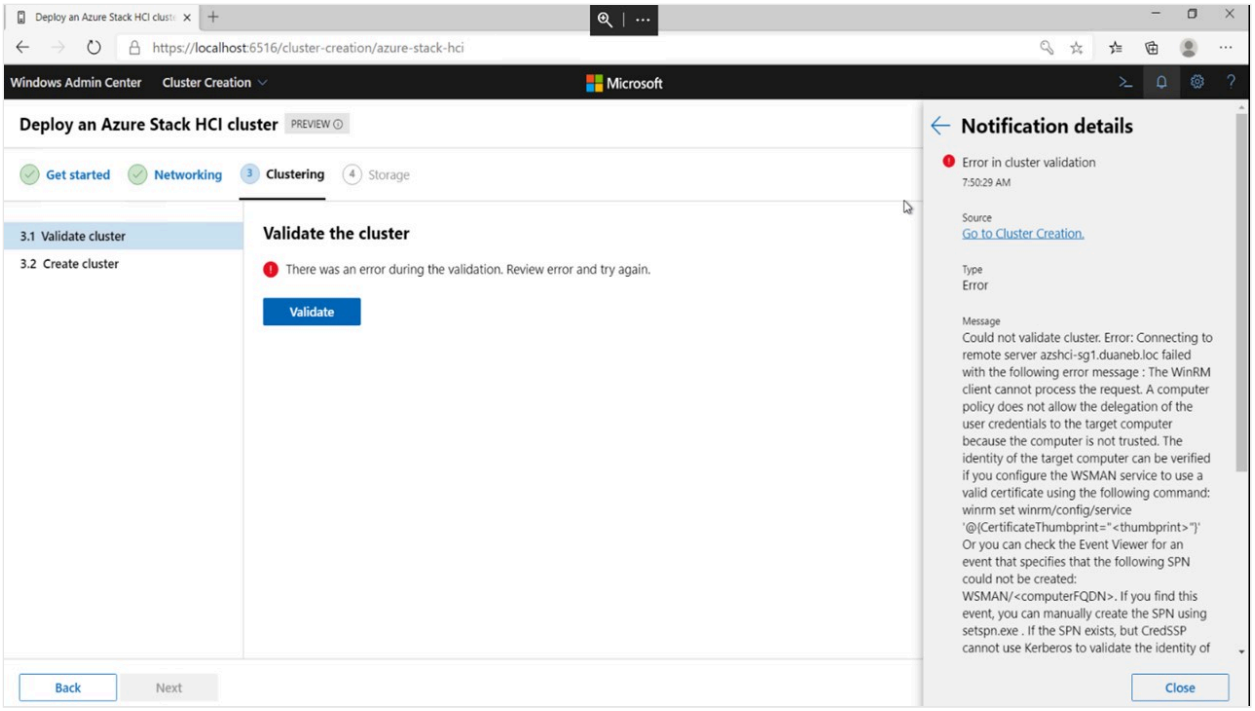
We recommend you use domain-joined servers when you create a stretch cluster. Due to WinRM limitations, you can encounter a network segmentation issue when you try to use work group machines while deploying a stretch cluster.

Undo and start over

When you use the same machines repeatedly while deploying clusters, you need to regularly clean up that set of machines. For more information about how to run cleanup processes on your cluster, see Deploy hyperconverged infrastructure.

CredSSP in cluster creation

The Windows Admin Center cluster deployment wizard uses CredSSP. Sometimes, CredSSP can cause an error message that says "There was an error during the validation. Review error and try again" appear when you're validating a cluster, as shown in the following screenshot.



To resolve this issue:

1. Open an elevated PowerShell window.
2. Disable CredSSP settings on all nodes and the Windows Admin Center gateway machine.
    - Run this command on your gateway machine:

      PowerShell

      ```
      Disable-WSManCredSSP -Role Client
      ```

    - Run this command on all nodes in your cluster:

      PowerShell

      ```
      Disable-WSManCredSSP -Role Server
      ```

3. Run the following command on all nodes to repair their trusts.

   PowerShell

   ```
   Test-ComputerSecureChannel -Verbose -Repair -Credential <account name>
   ```

4. Next, open a command prompt and run the following command on all nodes to reset group policy propagated data:

   ```
   
   ```

Windows Command Prompt

```
gpupdate /force
```

5. Reboot each node.

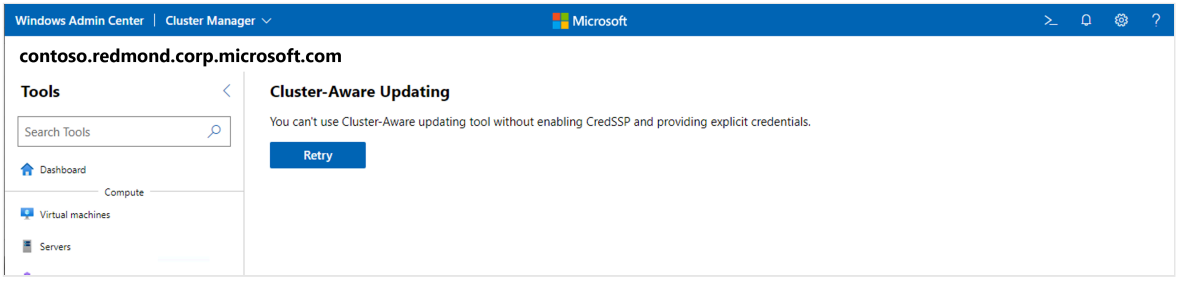
6. After rebooting the nodes, open PowerShell again and run the following command to test the connectivity between your gateway machine and target nodes.

PowerShell

```
Enter-PSSession -ComputerName <node fqdn>
```

CredSSP

- When you use the Updates tool, you sometimes see an error message that says "You can't use Cluster-Aware updating tool without enabling CredSSP and providing explicit credentials" when you try to update new clusters, as shown in the following screenshot.



  To resolve this issue, update Windows Admin Center to version 2110 or later.

- The CredSSP session endpoint permission issue is a common CredSSP error that appears when Windows Admin Center is running on Windows client machines. To resolve this issue, you should add affected users to the Windows Admin Center CredSSP administrators group, then ask the user to sign back in to the desktop computer running Windows Admin Center.

Nested virtualization

When you're validating Azure Local cluster deployments on VMs, you must enable nested virtualization before you enable roles or features by running the following command in PowerShell:

PowerShell

```
Set-VMProcessor -VMName <VMName> -ExposeVirtualizationExtensions $true
```

If you're using virtual switch teaming in a VM environment, you also need to run this command on the session host after creating a VM:

PowerShell

```
Get-VM | %{ Set-VMNetworkAdapter -VMName $_.Name -MacAddressSpoofing On -
AllowTeaming On }
```

If you're deploying a cluster using the Azure Stack HCI OS, there's an extra requirement. The VM boot virtual hard drive must be preinstalled with Hyper-V features. To preinstall these features, run the following command before creating the VMs:

PowerShell

```
Install-WindowsFeature –VHD <Path to the VHD> -Name Hyper-V, RSAT-Hyper-V-
Tools, Hyper-V-PowerShell
```

Remote direct memory access support

The cluster deployment feature in Windows Admin Center 2007 doesn't support remote direct memory access (RDMA) configurations. To resolve this issue, update to a later version of Windows Admin Center.

Failover Cluster Manager solution

- When managing a hyper-converged or traditional cluster, you can sometimes see an error message that says "Shell not found." You can do one of the following to resolve this issue:
  - Reload your browser
  - Go to another tool, then return to Failover Cluster Manager

- You can sometimes encounter an issue when managing a down-level cluster with an incomplete configuration. To resolve this issue, make sure the cluster has the **RSAT-Clustering-PowerShell** feature installed and enabled on each member node. If not, open PowerShell and enter the following command on each cluster node:

  PowerShell

```
Install-WindowsFeature -Name RSAT-Clustering-PowerShell
```

- If Windows Admin Center can't discover the cluster, try adding it with the entire fully qualified domain name (FQDN).
- When connecting to a cluster using Windows Admin Center installed as a gateway while using a username and password to authenticate, you must select **Use these credentials for all connections** so to make the credentials available to query the member nodes.

Hyper-Converged Cluster Manager solution

Windows Admin Center has disabled certain commands, such as **Drives - Update firmware**, **Servers - Remove** and **Volumes - Open**, because it doesn't currently support them.

Azure services

The following sections describe issues you can encounter when using Azure services while in Windows Admin Center.

Azure login and gateway registration

- When attempting to register your Windows Admin Center gateway in the Azure China 21Vianet or Azure US Gov cloud domains in version 2211, the gateway can sometimes redirect you to the Azure Global sign-in experience. To work around this issue, use an earlier version of Windows Admin Center.
- In the 2009 release, you can run into issues signing in to Azure or registering your Windows Admin Center gateway with Azure. Try doing the following to troubleshoot the issue:
  - Before using any Azure features in Windows Admin Center, including gateway registration, make sure you've signed in to your Azure account in a different tab or window. We recommend you sign in through the Azure portal.
  - If you successfully sign in to Azure during gateway registration but don't see visual confirmation on the **Azure** page of your Windows Admin Center settings, refresh the page by going to another page, then returning.

- If you've already given admin approval for Windows Admin Center in the portal but still see an error message that says "Need admin approval", try signing in to Azure using the banners around Windows Admin Center instead of going to the Settings page.
- If your proxy is misconfigured, you can see an error message that says "Error: Value cannot be null. Parameter name: httpClientFactory." To resolve this issue, go to the **Settings** page and adjust your settings to the correct configuration.

Azure File Sync permissions

Azure File Sync requires permissions in Azure that Windows Admin Center didn't provide before version 1910. If you registered your Windows Admin Center gateway with Azure using a version earlier than 1910, you must update your Microsoft Entra application in order to use Azure File Sync in the latest version of Windows Admin Center. The extra permissions let Azure File Sync automatically configure storage account access as described in Ensure Azure File Sync has access to the storage account.

There are two ways you can update Microsoft Entra ID.

To update using the registration method:

1. Go to **Settings** > **Azure** > **Unregister**
2. Register Windows Admin Center with Azure again, making sure you choose to create a new Microsoft Entra application.

To update using Azure:

1. Open Microsoft Entra ID.
2. Go to **App Registrations**, select the name of application you want to update to open its overview page.
3. Once you're in the application overview page, go to **API permissions**.
4. Select **Add a permission**.
5. Select **Microsoft Graph** > **Delegated permissions** > **Directory** and select the **Directory.AccessAsUser.All** checkbox.
6. Finally, select **Add permissions** to save the changes you made to the app.

Options for setting up Azure management services

Azure management services, including Azure Monitor, Azure Update Management, and Azure Security Center, all use the Microsoft Monitoring Agent for on-premises servers. Azure Update Management supports limited regions and needs its Log Analytics workspace linked to an Azure Automation account. If you want to set up multiple services in Windows Admin Center, you need to set up Azure Update Management first, then either Azure Security Center or Azure Monitor.

If you've already configured Azure management services that use the Microsoft Monitoring Agent before trying to use Azure Update Management in Windows Admin Center, the service only lets you configure Azure Update Management if existing resources linked to the Microsoft Monitoring Agent support it.

If the linked resources don't support Azure Update Management, there are two ways you can work around it.

To resolve the issue using the Control Panel:

1. On the Start menu, go to **Control Panel** > **Microsoft Monitoring Agent**.
2. Follow the directions in How do I stop an agent from communicating with Log Analytics to disconnect your server from Azure Monitor, Azure Security Center, or other Azure management solutions you're currently using.
3. Configure Azure Update Management in Windows Admin Center.
4. Reconnect to the Azure management solutions you disconnected in step 2.

To resolve the issue using Azure Update Management:

1. Follow the instructions in Update Management overview to manually set up the Azure resources you need for Azure Update Management.
2. Follow the directions in Adding or removing a workspace to manually update the Microsoft Monitoring Agent outside of Windows Admin Center and add the new workspace for the Update Management solution you want to use.

Windows Remote Management errors

You may encounter the following error messages when using Windows Remote Management.

General connection error

When you encounter this error, the following error message appears:

error

```
Cluster wasn't created Connecting to remote server tk5-
3wp13r1131.cfdev.nttest.microsoft.com failed
with the following error message:
WinRM cannot complete the operation. Verify that the specified computer name
is valid, that the
computer is accessible over the network, and that a firewall exception for
the WinRM service is
enabled and allows access from this computer. By default, the WinRM firewall
exception for public
profiles limits access to remote computers within the same local subnet. For
more information, see
the about_Remote_Troubleshooting Help topic.
```

This error usually appears when you're trying to connect using WinRM. It can happen for the following reasons:

- If the service couldn't resolve DNS, make sure you entered the correct server name.
- If the service couldn't reach the server name at all, this is likely due to a network connection issue, such as a network disruption.
- If the firewall rules aren't configured for the WinRM service, you must reconfigure them for domain and private profiles.
- If the WinRM service isn't running or disabled, enable the service and make sure it keeps running.

Authentication error

When you encounter this error, the following error message appears:

error

```
Connecting to remote server ack failed with the following error message:
WinRM cannot process the request. The following error with error code
0x8009030e occurred while
using Negotiate authentication: A specified logon session does not exist. It
may already have been
terminated. \r\n This can occur if the provided credentials are not valid on
the target server, or
if the server identity could not be verified. If you trust the server
identity add the server name
to the TrustedHosts list, and then retry the request. User winrm.cmd to view
or edit the
TrustedHosts list. Note that computers in the TrustedHosts list might not be
authenticated. For
more information about how to edit the TrustedHosts list, run the following
```

```
command: winrm help
config. For more information, see the about_Remote_Troubleshooting Help
topic.
```

This error usually occurs on cluster connections when WinRM can't connect because of the following reasons:

- The user is trying to remotely connect to a domain-connected machine while signed in as a local user administrator account.
- The user trying to sign in is in the domain but can't contact the domain even though they can reach the server. When this happens, WinRM treats the user like they aren't in the domain but are connecting to a domain account.

You can try the following methods to resolve this issue:

- Make sure users can always contact the domain, especially after a network operation.
- You should add all computers you're connecting to into the trusted hosts (FQDNS), such as `@{TrustedHosts="VS1.contoso.com,VS2.contoso.com,my2012cluster.contoso.com"}`.
- The General connection error should pass all validations.

WinRM service

When you encounter this error, the following error message appears:

error

```
We cannot display the changes right now:
Connecting to remote server localhost failed with the
following error message : The client cannot connect to the destination
specified in the request.
Verify that the service on the destination is running and is accepting
requests. Consult the logs
and documentation for the WS-Management services running on the destination,
mostly commonly IIS or
WinRM. If the destination is the WinRM service, run the following command on
the destination to
analyze and configure the WinRM service: "winrm quickconfig". For more
information, see the
about_Remote_Troubleshooting Help topic.
```

You can encounter this error for the following reasons:

- The WinRM service isn't running. The service could be temporarily disabled or completely shut down. To resolve this issue, make sure the WinRM service is always running.
- The WinRM listener isn't configured or is corrupted. The quickest way to solve this problem is to run `WinRM quickconfig` in PowerShell, which creates a listener. WinRM also has two built-in listeners for HTTPS and HTTP connections. The HTTPS server and client should both have the same valid certificates.

Security error

When you encounter this error, the following error message appears:

```error
Connecting to remote server dc1.root.contoso.com failed with the following
error message:
WinRM cannot process the request. The following error with errorcode
0x80090322 occurred while
using Kerberos authentication. An unknown security error occurred. At line:1
char:1 +
Enter-PSSession dc1.root.contoso.com +
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ + CategoryInfo
:InvalidArgument:(dc1.root.contoso.com:String)[Enter-PSSession],
PSRemotingTransportException +
FullyQualifiedErrorId : CreateRemoteRunspaceFailed
```

This error is uncommon. You usually encounter this area when an account tries to create a remote connection. In most cases, one or more default HTTP SPNs are registered to a service account, causing Kerberos authentication to fail. This issue usually happens because some software installed on the server needs one or more SPNs to function properly, such as SQL Server Reporting Services, Microsoft Dynamics, SharePoint, and so on.

In some cases, one of the SPNs is registered to a service account while the other one isn't. In that case, the WinRM connection succeeds when trying to start a session with the server name, but fails when it tries to start a session using the FQDN.

To resolve this issue, check if one or more default HTTP SPNs are registered to a service account by running the following command in PowerShell:

```PowerShell
setspn -q HTTP/servername.or.fqdn
```

If the service finds the SPN but the server name isn't in the highlighted field of the error message, run the following command to set up dedicated SPNs for WinRM by specifying the port number and the machine account:

PowerShell

```
setspn -s HTTP/servername.or.fqdn:5985 servername
```

If you're connecting remotely using PowerShell, make sure to also use the *IncludePortInSPN* parameter, as shown in the following example command:

PowerShell

```
Enter-PSSession -ComputerName servername.or.fqdn -SessionOption (New-
PSSessionOption -IncludePortInSPN)
```

WinRM status 500

When you encounter this error, the following error message appears:

error

```
Error: Connecting to remote server YAZSHCISIIH01.ad.yara.com failed with the
following error message:
The WinRM client received an HTTP server error status (500), but the remote
service did not include
any other information about the cause of the failure. For more information,
see the
about_Remote_Troubleshooting Help topic.
```

This error is very rare. When you see this error message, it usually means WinRM couldn't process the request. The reason why this error appears varies based on context.

To resolve this issue, make sure remoting is enabled and that you configure the WinRM listener to accept requests. We also recommend you check the event logs for other errors, such as if WinRM can't access certain files in the file system due to the files only having read permissions.

Feedback

Was this page helpful? Yes No

Provide product feedback

Windows Admin Center - License Terms

Article • 03/31/2022

Review our Windows Admin Center license terms.

- Microsoft Software License Terms - Pre-release extensions for Microsoft Windows Admin Center
- License terms for extensions
- Windows Admin Center extensions publisher agreement
- Microsoft extensions participation policy
- Microsoft Software License Terms - Windows Admin Center - Preview
- Windows Admin Center for Microsoft Windows Server and Microsoft Windows operating system (Version 10)
- Windows Server-related license terms

Feedback

Was this page helpful? Yes No

Extensions for Windows Admin Center

Article • 01/12/2022 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Windows Admin Center is built as an extensible platform to enable partners and developers to leverage existing capabilities within Windows Admin Center, seamlessly integrate with other IT administration products and solutions, and provide extra value to customers. Each solution and tool in Windows Admin Center is built as an extension using the same extensibility features available to partners and developers, so you can build powerful tools just like the ones available in Windows Admin Center today.

Windows Admin Center extensions are built using modern web technologies including HTML5, CSS, Angular, TypeScript and jQuery, and can manage target servers via PowerShell or WMI. You can also manage target servers, services, or devices over different protocols such as REST by building a Windows Admin Center gateway plugin.

Why you should consider developing an extension for Windows Admin Center

Here's the value you can bring to your product and customers by developing extensions for Windows Admin Center:

- **Integrate with Windows Admin Center tools:** Integrate your products and services with server and cluster management tools in Windows Admin Center and deliver unified and seamless, end-to-end monitoring, management, troubleshooting experiences to your customers.
- **Leverage platform security, identity and management capabilities:** Enable Microsoft Entra ID support, multi-factor authentication, Role-Based Access Control (RBAC), logging, and auditing for your product and services by leveraging Windows Admin Center platform capabilities to meet the complex requirements of today's IT organizations.
- **Develop using the latest web technologies:** Quickly build stunning user experiences using modern web technologies including HTML5, CSS, Angular, TypeScript and jQuery, and rich, powerful UI controls included in the Windows Admin Center SDK.
- **Extend product outreach:** Become a part of the Windows Admin Center ecosystem with outreach to our expanding customer base.

Start developing with the Windows Admin Center SDK

Getting started with Windows Admin Center development is easy! Sample code can be found for tool, solution, and gateway plugin extension types in our SDK documentation. There you will use the Windows Admin Center SDK to build a new extension project, then follow the individual guides to customize your project to meet your needs.

We've made a Windows Admin Center SDK design toolkit available to help you rapidly mock up extensions in PowerPoint using Windows Admin Center styles, controls, and page templates. See what your extension can look like in Windows Admin Center before you start coding!

We also have sample code hosted on GitHub: Developer Tools is a sample solution extension containing a rich collection of controls that you can browse and use in your own extension. Developer Tools is a fully functioning extension that can be side-loaded into Windows Admin Center in Developer Mode.

See the topics below to learn more about the SDK and get started:

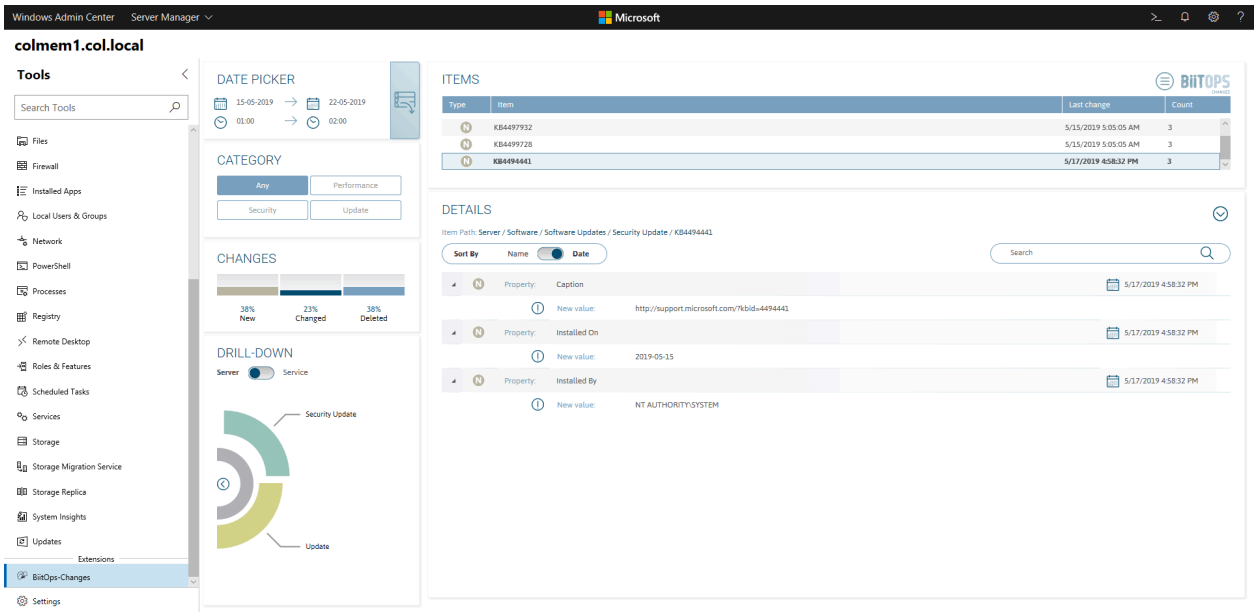
- Understand how extensions work
- Develop an extension
- Guides
- Publish your extension

Partner Spotlight

See the amazing value our partners have started to bring to the Windows Admin Center ecosystem and try these extensions out today. Learn more on how to install extensions from Windows Admin Center.

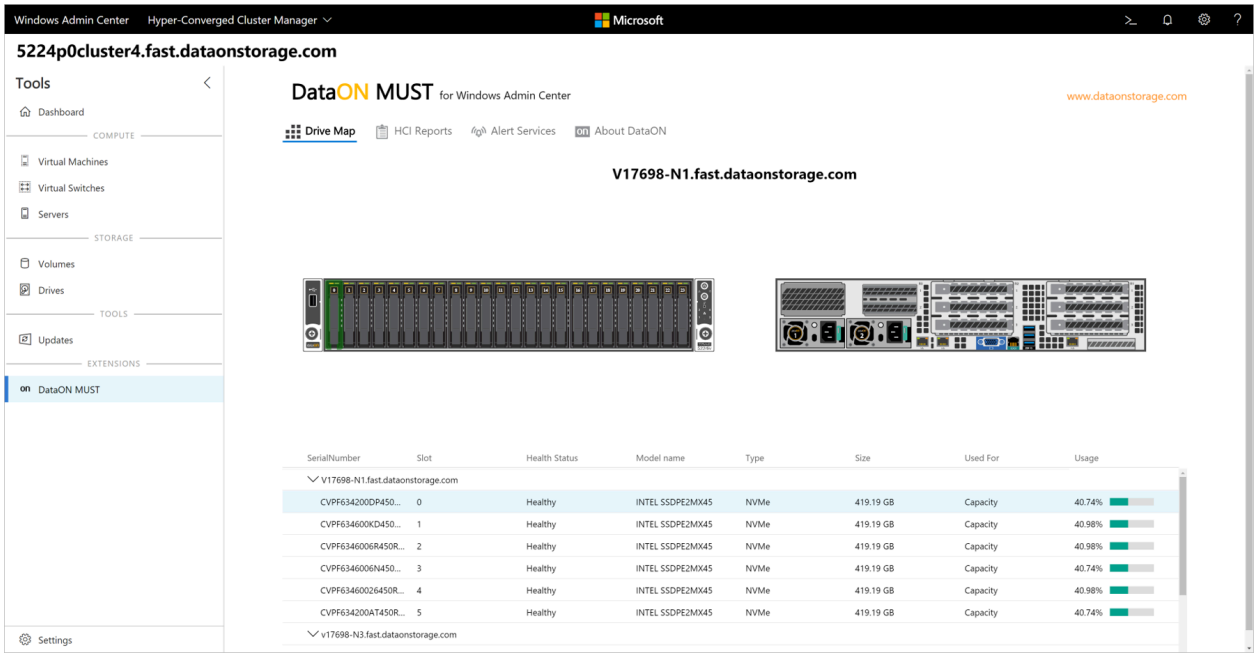
BiitOps

The BiitOps Changes extension provides change tracking for hardware, software, and configuration settings on your Windows Server physical/virtual machines. The BiitOps Changes extension will show precisely what is new, what has changed and what has been deleted in a single-pane-of-glass to help track issues related to compliance, reliability and security. Learn more about the BiitOps Changes extension.



DataON

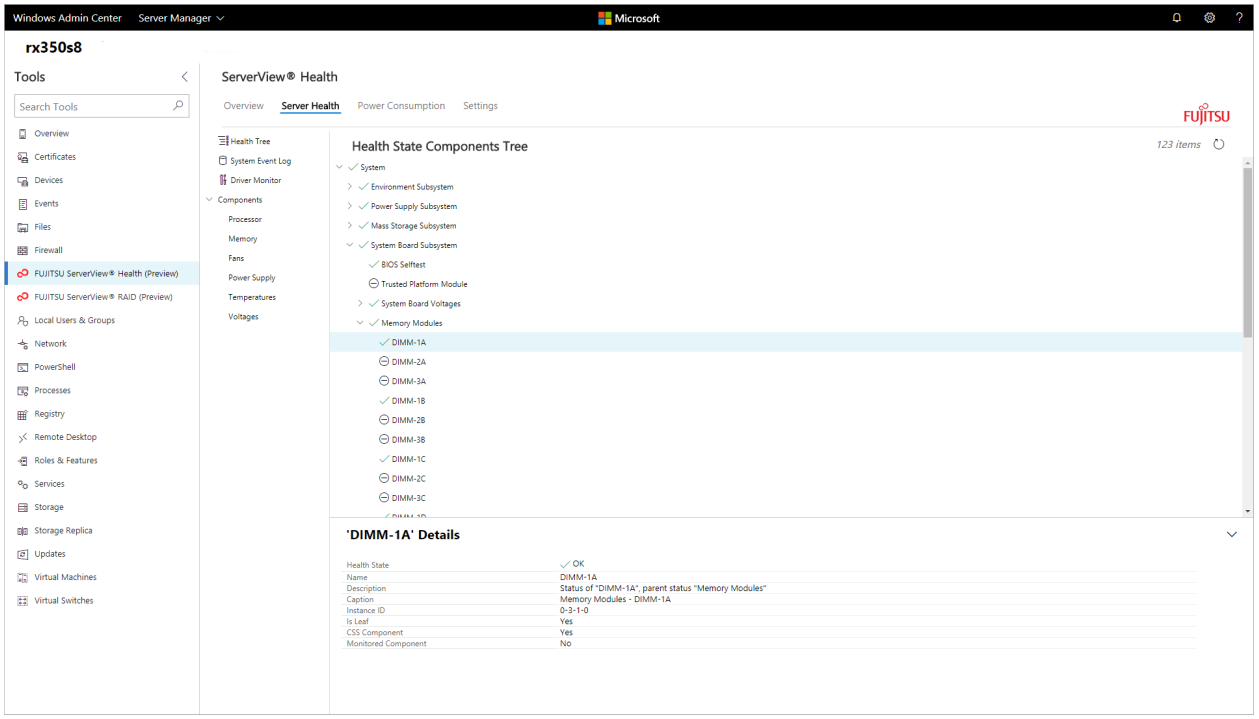
The DataON MUST extension brings monitoring, management, and end-to-end insight into DataON's hyper-converged infrastructure and storage systems based on Windows Server. The MUST extension adds unique value such as historical data reporting, disk mapping, system alerts and SAN-like call home service, complementing the Windows Admin Center server and hyper-converged infrastructure management capabilities, through a seamless, unified experience. Learn more about DataON's MUST extension and their development experience.



Fujitsu

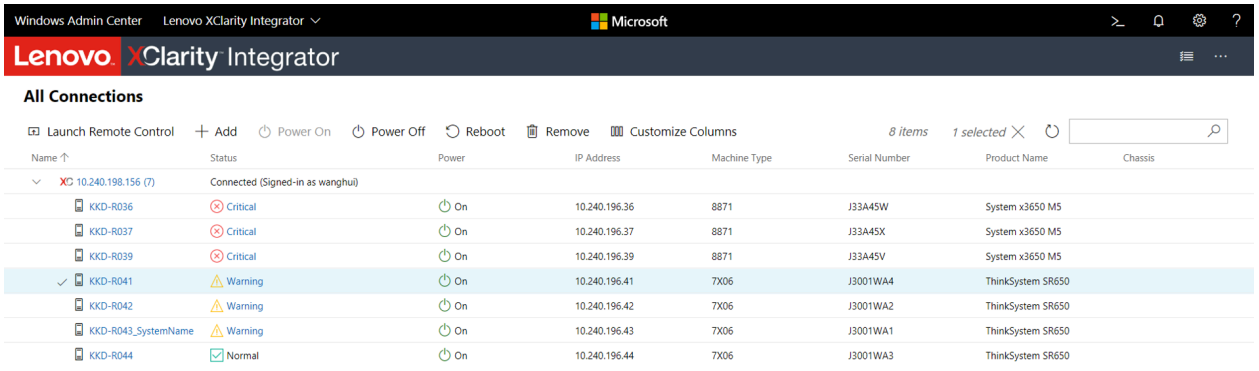
Fujitsu's ServerView Health and RAID Health extensions for Windows Admin Center provide in-depth monitoring and management of critical hardware components such as

processors, memory, power, and storage subsystems for Fujitsu PRIMERGY servers. By utilizing the Windows Admin Center UX design patterns and UI controls, Fujitsu has brought us a huge step towards our vision of end-to-end insight into server roles and services, to operating system, and to hardware management through the Windows Admin Center platform. Learn more about Fujitsu's extensions and their development experience.



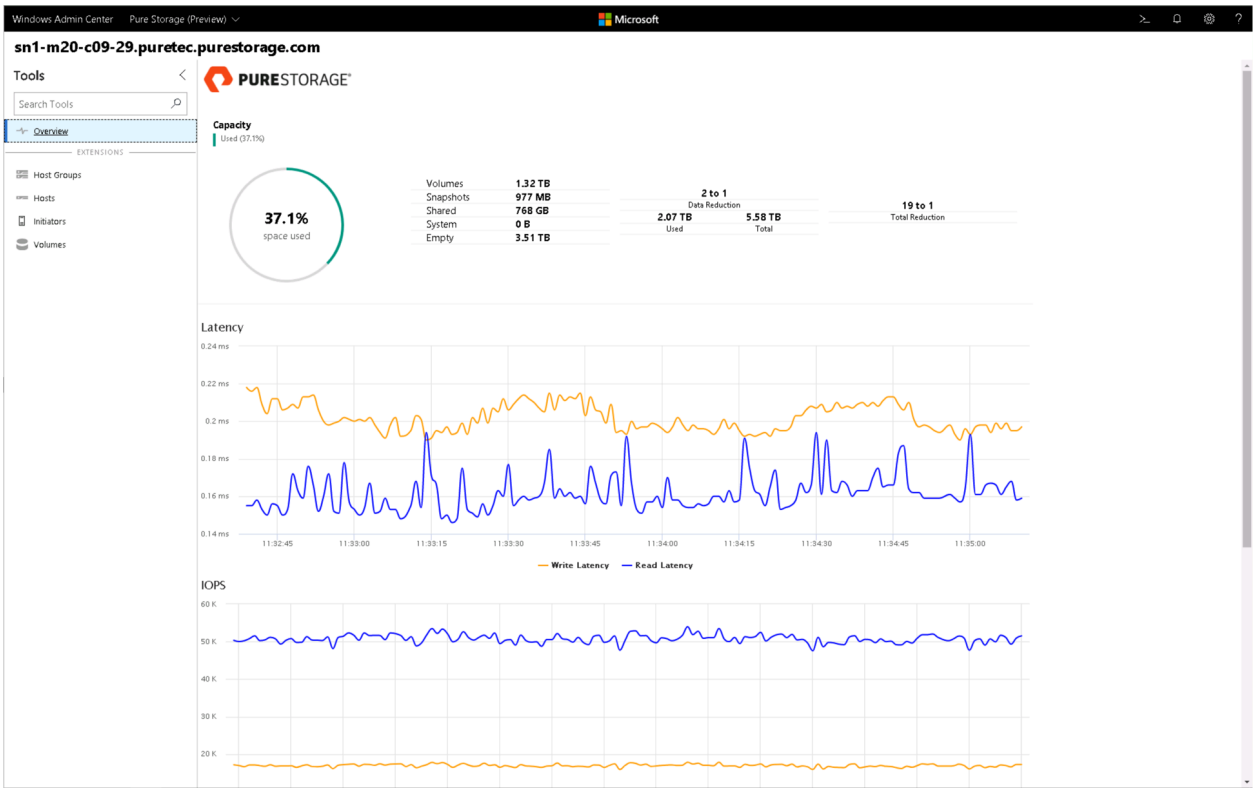
Lenovo

The Lenovo XClarity Integrator extension takes hardware management to the next level by seamlessly integrating into various experiences within Windows Admin Center. The XClarity Integrator solution provides a high-level view of all your Lenovo servers, and different tool extensions provide hardware details whether you are connected to a single server, failover cluster, or hyper-converged cluster. Learn more about the Lenovo XClarity Integrator extension.



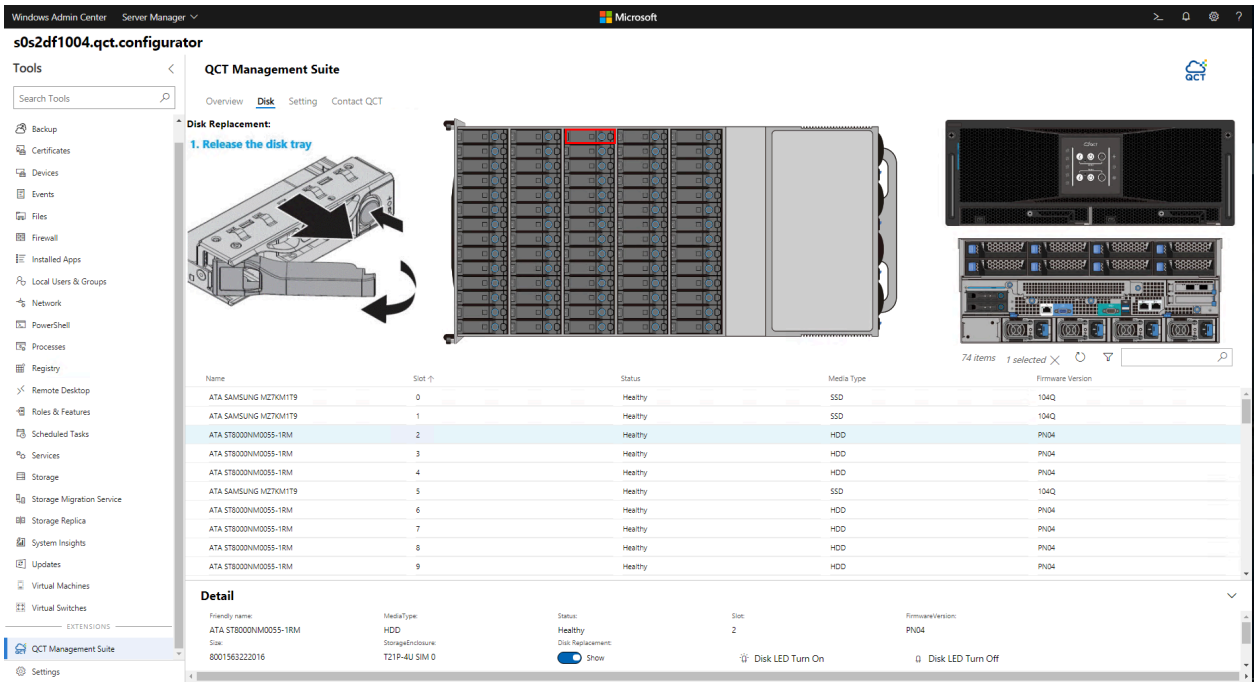
Pure Storage

Pure Storage provides enterprise, all-flash data storage solutions that deliver data-centric architecture to accelerate your business for a competitive advantage. The Pure Storage extension for Windows Admin Center provides a single-pane view into Pure FlashArray products and empowers users to conduct monitoring tasks, view real-time performance metrics, and manage storage volumes and initiators through a single UI experience. Learn more about Pure's extensions and their development experience.



QCT

The QCT Management Suite extension complements Windows Admin Center by providing physical server monitoring and management for QCT Azure Local certified systems. The QCT Management Suite extension displays server hardware information, and provides an intuitive wizard UI to help replace physical disks efficiently, hardware event log tools, and S.M.A.R.T. based predictive disk management. Learn more about the QCT Management Suite extension.



Feedback

Was this page helpful? Yes No

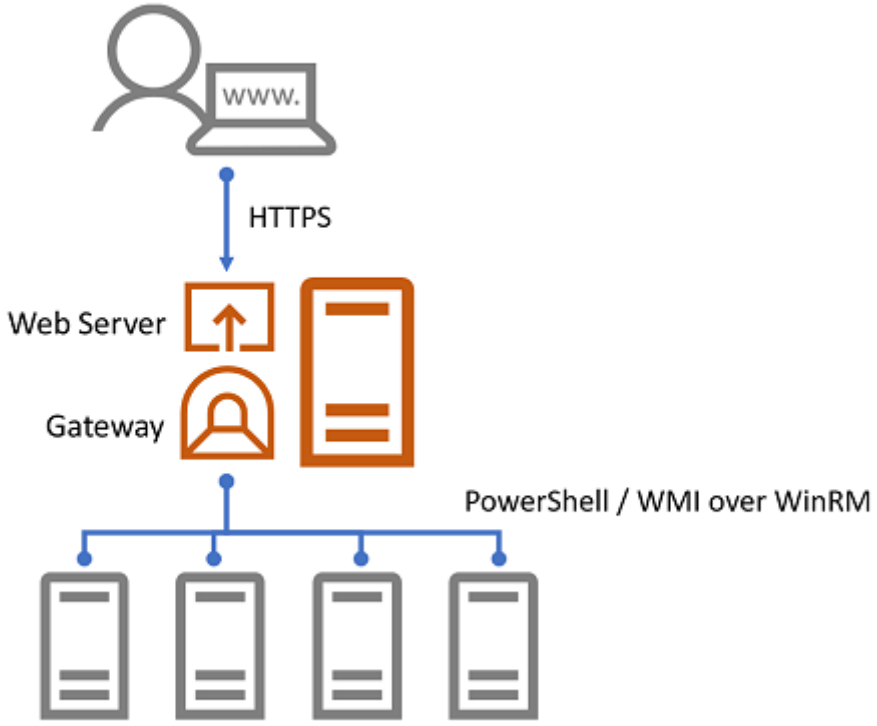
Provide product feedback

Understanding Windows Admin Center Extensions

Article • 11/01/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

In case you're not yet familiar with how Windows Admin Center works, let's start with the high-level architecture. Windows Admin Center is composed of two main components:

- Lightweight **web service** that serves Windows Admin Center UI web pages to web browser requests.
- **Gateway component** that listens for REST API requests from the web pages and relays WMI calls or PowerShell scripts to be executed on a target server or cluster.

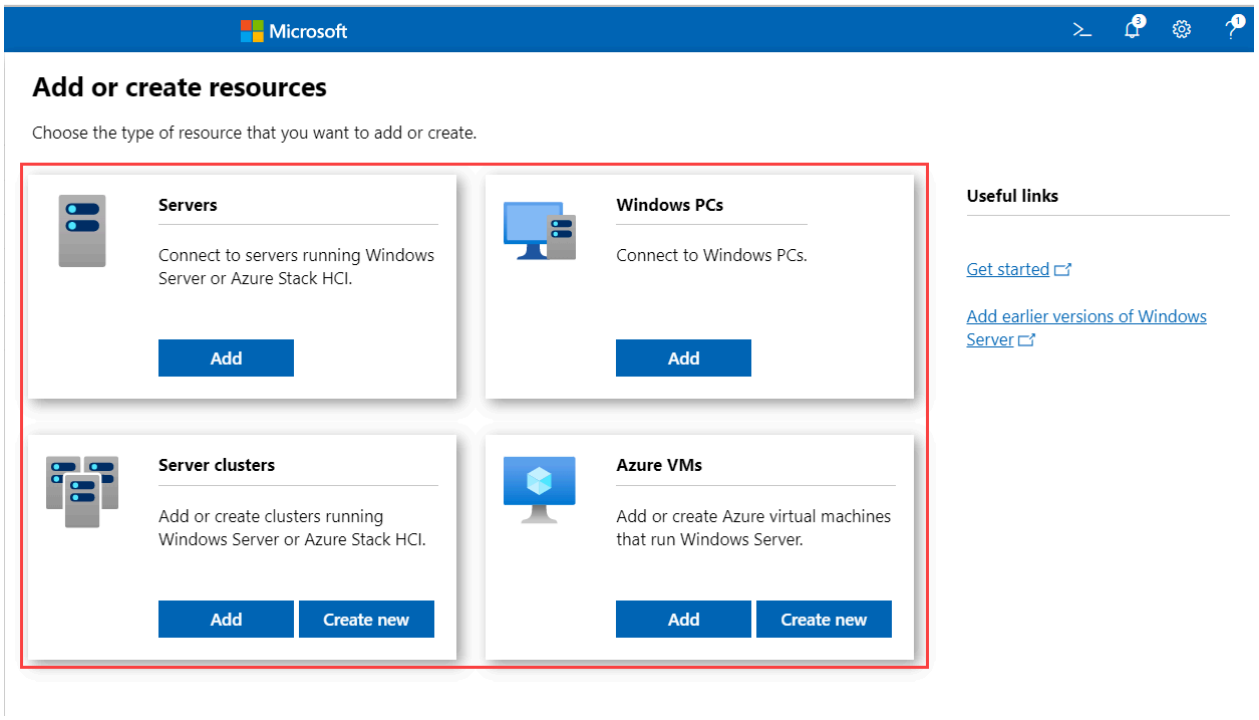


The Windows Admin Center UI web pages served by the web service have two main UI components from an extensibility perspective, solutions and tools, which are implemented as extensions, and, a third extension type called gateway plugins.

Solution extensions

In the Windows Admin Center home screen, by default, you can add connections that are one of four types – Windows Server connections, Windows PC connections, server clusters connections and Azure VMs connections. Once a connection is added, the connection name and type will be displayed in the home screen. Clicking on the

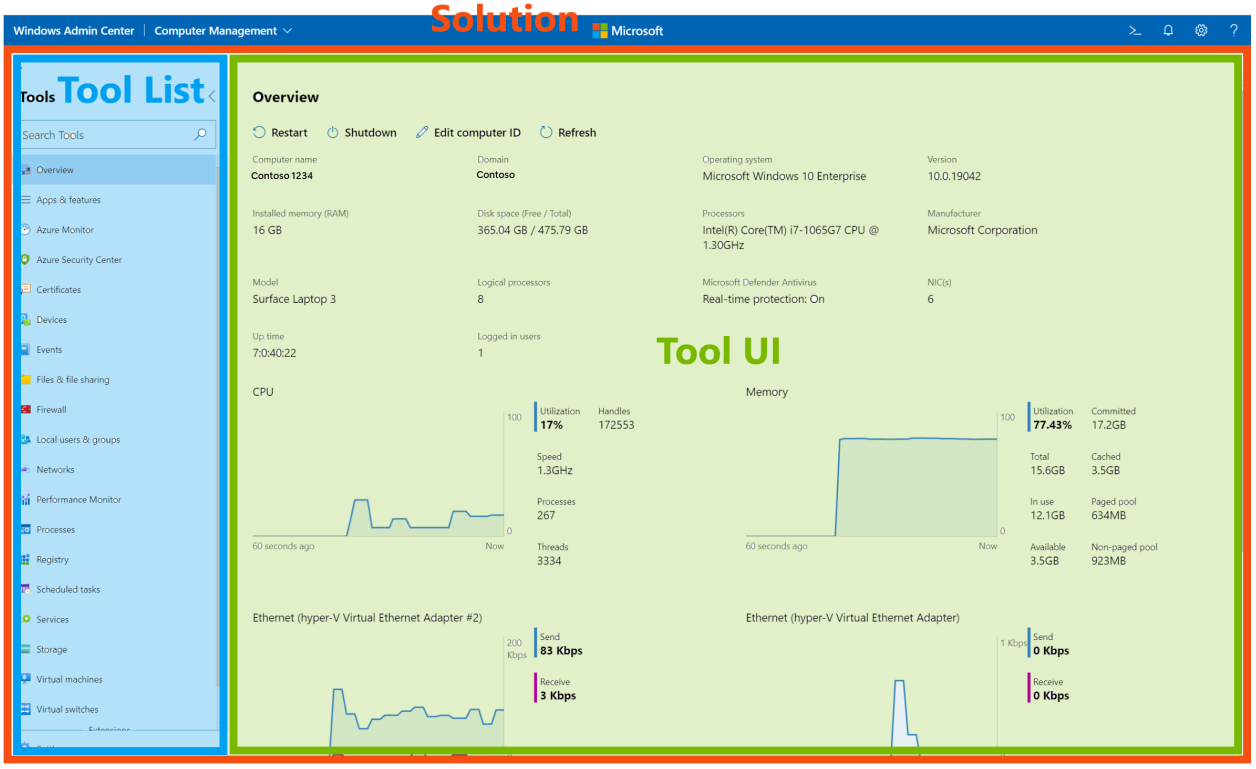
connection name will attempt to connect to the target server or cluster and then load the UI for the connection.



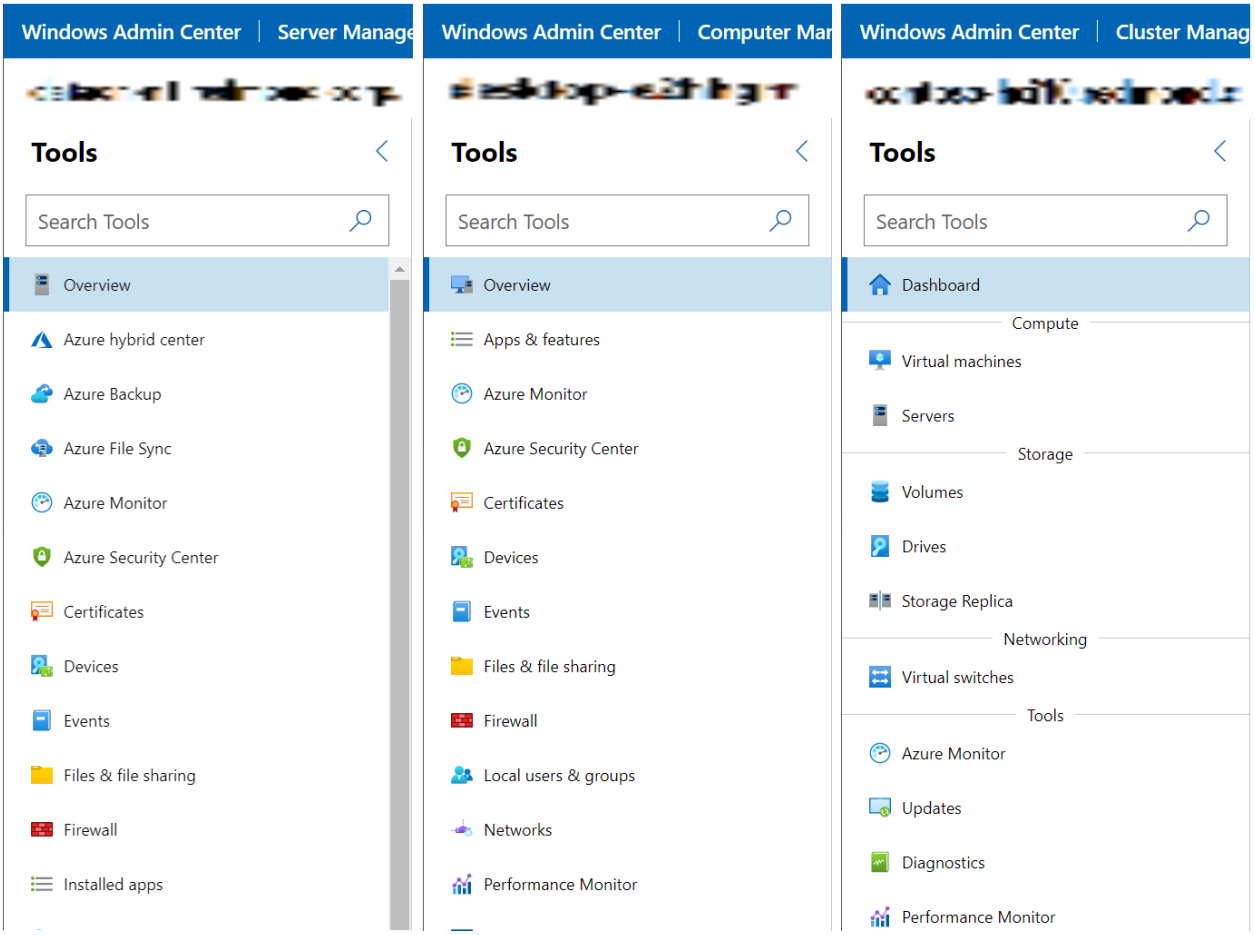
Each connection type maps to a solution, and solutions are defined through a type of extension called “solution” extensions. Solutions typically define a unique type of object you wish to manage through Windows Admin Center, such as servers, PCs or failover clusters. You could also define a new solution for connecting to and managing other devices such as network switches and Linux servers, or even services such as Remote Desktop Services.

Tool extensions

When you click on a connection in the Windows Admin Center home screen and connect, the solution extension for the selected connection type will be loaded and you'll then be presented with the solution UI including a list of tools in the left navigation pane. When you click on a tool, the tool UI is loaded and displayed in the right pane.



Each tool is defined through a second type of extension called “tool” extensions. When a tool is loaded, it can execute WMI calls or PowerShell scripts on a target server or cluster and display information in the UI or execute commands based on user input. A tool extension defines which solutions it should be displayed for, resulting in a different set of tools for each solution. If you're creating a new solution extension, you'll additionally need to write one or more tool extensions that provide functionality for the solution.



Gateway plugins

The gateway service exposes REST APIs for the UI to call and relays commands and scripts to be executed on the target. The gateway service can be extended by gateway plugins that support different protocols. Windows Admin Center is pre-packaged with two gateway plugins, one for executing PowerShell scripts and the other for WMI commands. If you need to communicate with the target through a protocol other than PowerShell or WMI, such as REST, you can build a gateway plugin for this.

Next steps

Depending on what capabilities you want to build in Windows Admin Center, building a tool extension for an existing server or cluster solution may be sufficient, and is the easiest first step into building extensions. However, if your feature is for managing a device, service or something completely new, rather than a server or cluster, you should consider building a solution extension with one or more tools. And finally, if you need to communicate with the target through a protocol other than WMI or PowerShell, you'll need to build a gateway plugin. Continue reading on to learn how to set up your development environment and start writing your first extension.

Feedback

Was this page helpful? Yes No

Provide product feedback

Develop an extension for Windows Admin Center

Article • 11/01/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Windows Admin Center supports three types of extensions - tool extensions, solution extensions and gateway plugins. The SDK contains content and examples to guide you in building the different types of extensions/plugins.

> **Note**
>
> Not familiar with the different extension types? Learn more about the **extensibility architecture and extension types**.

Development step-by-step

- Prepare your development environment
- Create a tool extension
- Create a solution extension
- Create a gateway plugin
- Learn more with our guides

SDK design toolkit

Check out our Windows Admin Center SDK design toolkit ! This toolkit is designed to help you rapidly mock up extensions in PowerPoint using Windows Admin Center styles, controls, and page templates. See what your extension can look like in Windows Admin Center before you start coding!

Feedback

Was this page helpful? Yes No

Provide product feedback

Prepare your development environment

Article • 12/06/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Let's get started developing extensions with the Windows Admin Center SDK! In this document, we'll cover the process to get your environment up and running to build and test an extension for Windows Admin Center.

> ⓘ **Note**
>
> New to the Windows Admin Center SDK? Learn more about Extensions for Windows Admin Center

To prepare your development environment, perform the following steps:

Install prerequisites

To begin developing with the SDK, download and install the following prerequisites:

- Windows Admin Center (GA or preview version)
- Visual Studio or Visual Studio Code
- Node.js (download and install the .msi file for version 16.20.2)
- Node Package Manager (6.14.5 or later)
- NuGet (for publishing extensions)

To bypass signature validation which occurs during extension installation in Windows Admin Center versions running with .NET 8, you should also switch your Windows Admin Center operation mode to "Development." To configure development mode, you must:

- Open your Windows Admin Center instance's `appsettings.json` file. This can be found by default at `C:\Program Files\WindowsAdminCenter\service\appsettings.json`.
- Modify the `OperationMode` field under the `WindowsAdminCenter/System` section to read "Development" instead of "Production."

> ⓘ **Note**
>
> To develop extensions on versions of Windows Admin Center using .NET 4.6.2, you need to install and run Windows Admin Center in Dev Mode to follow the steps

below. Dev Mode allows Windows Admin Center to load unsigned extension packages. Windows Admin Center can only be installed in Dev Mode on a Windows 10 machine.

To enable Dev Mode, install Windows Admin Center from the command line with the parameter DEV_MODE=1. In the example below, replace `<version>` with the version you are installing, i.e. `WindowsAdminCenter1809.msi`.

`msiexec /i WindowsAdminCenter<version>.msi DEV_MODE=1`

If you have already installed Windows Admin Center without enabling Dev Mode, you can edit the value of the Dev Mode property using Registry Editor. Properties for Windows Admin Center can be found under the following path:

`Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\ServerManagementGateway`

Install global dependencies

Next, install or update dependencies required for your projects, with Node Package Manager. These dependencies will be installed globally, and will be available for all projects.

```
npm install -g @angular/cli@11.2.14

npm install -g gulp
npm install -g typescript
npm install -g tslint
npm install -g @microsoft/windows-admin-center-sdk@latest
```

ⓘ Note

You can install a later version of @angular/cli, however be aware that if you install a version greater than 11.2.14, you will receive a warning during the gulp build step that the local cli version does not match the installed version.

Next steps

Now that your environment is prepared, you are ready to start creating content.

- Create a tool extension
- Create a solution extension
- Create a gateway plugin
- Learn more with our guides

SDK design toolkit

Check out our Windows Admin Center SDK design toolkit ! This toolkit is designed to help you rapidly mock up extensions in PowerPoint using Windows Admin Center styles, controls, and page templates. See what your extension can look like in Windows Admin Center before you start coding!

Feedback

Was this page helpful? Yes No

Provide product feedback

Develop a tool extension

Article • 02/02/2023 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

A tool extension is the primary way that users interact with Windows Admin Center to manage a connection, such as a server or cluster. When you click on a connection in the Windows Admin Center home screen and connect, you will then be presented with a list of tools in the left navigation pane. When you click on a tool, the tool extension is loaded and displayed in the right pane.

When a tool extension is loaded, it can execute WMI calls or PowerShell scripts on a target server or cluster and display information in the UI or execute commands based on user input. Tool extensions define which solutions it should be displayed for, resulting in a different set of tools for each solution.

> **Note**
>
> Not familiar with the different extension types? Learn more about the extensibility architecture and extension types.

Prepare your environment

If you haven't already, prepare your environment by installing dependencies and global prerequisites required for all projects.

Create a new tool extension with the Windows Admin Center SDK

Once you have all the dependencies installed, you are ready to create your new tool extension. Create or browse to a folder that contains your project files, open a command prompt, and set that folder as the working directory. Using the Windows Admin Center SDK that was installed previously, create a new extension with the following syntax:

Windows Command Prompt

```
wac create --company "{!Company Name}" --tool "{!Tool Name}" --version
latest
```

Expand table

Value	Explanation	Example
`{!Company Name}`	Your company name (with spaces)	`Contoso Inc`
`{!Tool Name}`	Your tool name (with spaces)	`Manage Foo Works`

Here's an example usage:

Windows Command Prompt

```
wac create --company "Contoso Inc" --tool "Manage Foo Works" --version
latest
```

This creates a new folder inside the current working directory using the name you specified for your tool, copies all the necessary template files into your project, and configures the files with your company and tool name.

> ⓘ **Note**
>
> The --version flag in this command specifies which version of the Windows Admin Center SDK you'd like to target. Read about how to target a different version of the Windows Admin Center SDK to keep up your extension up to date with the latest SDK and platform changes.

Next, change directory into the folder just created, then install required local dependencies by running the following command:

Windows Command Prompt

```
npm install
```

Once this completes, you've set up everything you need to load your new extension into Windows Admin Center.

Add content to your extension

Now that you've created an extension with the Windows Admin Center SDK, you are ready to customize content. See these guides for examples of what you can do:

- Add an empty module
- Add an iFrame

Even more examples can be found in our Developer Guide. The Developer Guide is a fully functioning solution extension that can be side-loaded into Windows Admin Center, and contains a rich collection of sample functionality and tool examples that you can browse and use in your own extension.

Enable the Developer Guide extension on the **Advanced** page of your Windows Admin Center settings.

Customize your extension's icon

You can customize the icon that shows for your extension in the tool list. To do this, modify all `icon` entries in `manifest.json` for your extension:

JSON

```
"icon": "{!icon-uri}",
```

Expand table

Value	Explanation	Example uri
`{!icon-uri}`	The location of your icon resource	`assets/foo-icon.svg`

NOTE: Currently, custom icons aren't visible when side loading your extension in dev mode. As a workaround, remove the contents of `target` as follows:

JSON

```
"target": "",
```

This configuration is only valid for side loading in dev mode, so it's important to preserve the value contained in `target` and then restore it before publishing your extension.

Build and side load your extension

Next, build and side load your extension into Windows Admin Center. Open a command window, change directory to your source directory, then you're ready to build.

- Build and serve with gulp:

  Windows Command Prompt

```
gulp build
gulp serve --port 4201
```

Note that you need to choose a port that is currently free. Make sure you do not attempt to use the port that Windows Admin Center is running on.

Your project can be side loaded into a local instance of Windows Admin Center for testing by attaching the locally served project into Windows Admin Center.

- Launch Windows Admin Center in a web browser
- Open the debugger (F12)
- Open the Console and type the following command:

  Windows Command Prompt

  ```
  MsftSme.sideLoad("http://localhost:4201")
  ```

- Refresh the web browser

Your project will now be visible in the Tools list with (side loaded) next to the name.

Feedback

Was this page helpful? Yes No

Provide product feedback

Develop a solution extension

Article • 02/02/2023 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Solutions primarily define a unique type of object you wish to manage through Windows Admin Center. These solutions/connection types are included with Windows Admin Center by default:

- Windows Server connections
- Windows PC connections
- Failover cluster connections
- Hyper-converged cluster connections

When you select a connection from the Windows Admin Center connection page, the solution extension for that connection's type is loaded, and Windows Admin Center will attempt to connect to the target node. If the connection is successful, the solution extension's UI will load, and Windows Admin Center will display the tools for that solution in the left navigation pane.

If you would like to build a management GUI for services not defined by the default connection types above, such a network switch, or other hardware not discoverable by computer name, you may want to create your own solution extension.

> ⓘ **Note**
>
> Not familiar with the different extension types? Learn more about the **extensibility architecture and extension types**.

Prepare your environment

If you haven't already, prepare your environment by installing dependencies and global prerequisites required for all projects.

Create a new solution extension with the Windows Admin Center SDK

Once you have all the dependencies installed, you are ready to create your new solution extension. Create or browse to a folder that contains your project files, open a command prompt, and set that folder as the working directory. Using the Windows Admin Center SDK that was installed previously, create a new extension with the following syntax:

```
wac create --company "{!Company Name}" --solution "{!Solution Name}" --tool
"{!Tool Name}" --version latest
```

Expand table

Value	Explanation	Example
`{!Company Name}`	Your company name (with spaces)	`Contoso Inc`
`{!Solution Name}`	Your solution name (with spaces)	`Contoso Foo Works Suite`
`{!Tool Name}`	Your tool name (with spaces)	`Manage Foo Works`

Here's an example usage:

```
wac create --company "Contoso Inc" --solution "Contoso Foo Works Suite" --
tool "Manage Foo Works"
```

This creates a new folder inside the current working directory using the name you specified for your solution, copies all the necessary template files into your project, and configures the files with your company, solution, and tool name.

> **Note**
>
> The --version flag in this command specifies which version of the Windows Admin Center SDK you'd like to target. Read about how to target a different version of the Windows Admin Center SDK to keep up your extension up to date with the latest SDK and platform changes.

Next, change directory into the folder just created, then install required local dependencies by running the following command:

```
npm install
```

Once this completes, you've set up everything you need to load your new extension into Windows Admin Center.

Add content to your extension

Now that you've created an extension with the Windows Admin Center SDK, you are ready to customize content. See these guides for examples of what you can do:

- Add an empty module
- Add an iFrame
- Create a custom connection provider
- Modify root navigation behavior

Even more examples can be found in our Developer Guide. The Developer Guide is a fully functioning solution extension that can be side-loaded into Windows Admin Center, and contains a rich collection of sample functionality and tool examples that you can browse and use in your own extension.

Enable the Developer Guide extension on the **Advanced** page of your Windows Admin Center settings.

Build and side load your extension

Next, build and side load your extension into Windows Admin Center. Open a command window, change directory to your source directory, then you're ready to build.

- Build and serve with gulp:

```
gulp build
gulp serve --port 4201
```

Note that you need to choose a port that is currently free. Make sure you do not attempt to use the port that Windows Admin Center is running on.

Your project can be side loaded into a local instance of Windows Admin Center for testing by attaching the locally served project into Windows Admin Center.

- Launch Windows Admin Center in a web browser
- Open the debugger (F12)
- Open the Console and type the following command:

```
MsftSme.sideLoad("http://localhost:4201")
```

- Refresh the web browser

Your project will now be visible in the Tools list with (side loaded) next to the name.

Feedback

Was this page helpful? Yes No

Provide product feedback

Develop a gateway plugin

Article • 12/09/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

A Windows Admin Center gateway plugin enables API communication from the UI of your tool or solution to a target node. Windows Admin Center hosts a gateway service that relays commands and scripts from gateway plugins to be executed on target nodes. The gateway service can be extended to include custom gateway plugins that support protocols other than the default ones.

These gateway plugins are included by default with Windows Admin Center:

- PowerShell gateway plugin
- Windows Management Instrumentation (WMI) gateway plugin

If you would like to communicate with a protocol other than PowerShell or WMI, such as with REST, you can build your own gateway plugin. Gateway plugins are loaded into a separate AppDomain from the existing gateway process, but use the same level of elevation for rights.

> ⓘ **Note**
>
> Not familiar with the different extension types? Learn more about the extensibility architecture and extension types.

> ⓘ **Important**
>
> The Windows Admin Center SDK and developer tools have not yet been updated to support development of gateway plug-ins compatible with the Windows Admin Center modernized gateway. Following this guide will not result in a .NET 8 extension compatible with the modernized gateway.
>
> If you're interested in developing a gateway plug-in for the modernized gateway or upgrading your existing gateway plug-in, send an email to wacextensionrequest@microsoft.com.
>
> Gateway plug-ins developed in .NET 6 while the backend update was in preview may not function correctly with the latest version of Windows Admin Center due to breaking changes from .NET 6 to .NET 8. We recommend upgrading your extension to .NET 8 as soon as possible.

Prepare your environment

If you haven't already, prepare your environment by installing dependencies and global prerequisites required for all projects.

Create a gateway plugin (C# library)

To create a custom gateway plugin, create a new C# class that implements the `IPlugIn` interface from the `Microsoft.ManagementExperience.FeatureInterfaces` namespace.

> **Note**
>
> The `IFeature` interface, available in earlier versions of the SDK, is now flagged as obsolete. All gateway plugin development should use IPlugIn (or optionally the HttpPlugIn abstract class).

Download sample from GitHub

To get started quickly with a custom gateway plugin, you can clone or download a copy of our sample C# plugin project from our Windows Admin Center SDK GitHub site.

Add content

Add new content to your cloned copy of the sample C# plugin project project (or your own project) to contain your custom APIs, then build your custom gateway plugin DLL file to be used in the next steps.

Deploy plugin for testing

Test your custom gateway plugin DLL by loading it into Windows Admin Center gateway process.

Windows Admin Center looks for all plugins in a `plugins` folder in the Application Data folder of the current machine (using the CommonApplicationData value of the Environment.SpecialFolder enumeration). On Windows 10, this location is `C:\ProgramData\Server Management Experience`. If the `plugins` folder doesn't exist yet, you can create the folder yourself.

> **Note**

You can override the plugin location in a debug build by updating the "StaticsFolder" configuration value. If you're debugging locally, this setting is in the App.Config of the Desktop solution.

Inside the plugins folder (in this example, `C:\ProgramData\Server Management Experience\plugins`)

- Create a new folder with the same name as the `Name` property value of the `Feature` in your custom gateway plugin DLL (in our sample project, the `Name` is "Sample Uno")
- Copy your custom gateway plugin DLL file to this new folder
- Restart the Windows Admin Center process

After the Windows Admin process restarts, you'll be able to exercise the APIs in your custom gateway plugin DLL by issuing a GET, PUT, PATCH, DELETE, or POST to `http(s)://{domain|localhost}/api/nodes/{node}/features/{feature name}/{identifier}`

Optional: Attach to plugin for debugging

In Visual Studio 2017, from the Debug menu, select "Attach to Process." In the next window, scroll through the Available Processes list and select SMEDesktop.exe, then select "Attach." Once the debugger starts, you can place a breakpoint in your feature code, and then exercise through the URL format. For our sample project (feature name: "Sample Uno") the URL is: `"<http://localhost:6516/api/nodes/fake-server.my.domain.com/features/Sample%20Uno>"`

Create a tool extension with the Windows Admin Center SDK

Now we need to create a tool extension from which you can call your custom gateway plugin. Create or browse to a folder where you want to store your project files, open a command prompt, and set that folder as the working directory. Using the Windows Admin Center SDK that was installed earlier, create a new extension with the following syntax:

```
wac create --company "{!Company Name}" --tool "{!Tool Name}"
```

Expand table

Value	Explanation	Example
`{!Company Name}`	Your company name (with spaces)	`Contoso Inc`
`{!Tool Name}`	Your tool name (with spaces)	`Manage Foo Works`

Here's an example usage:

```
wac create --company "Contoso Inc" --tool "Manage Foo Works"
```

This command creates a new folder inside the current working directory using the name you specified for your tool, copies all the necessary template files into your project, and configures the files with your company and tool name.

Next, change directory into the folder, then install required local dependencies by running the following command:

```
npm install
```

Once this command completes, you're set up with everything you need to load your new extension into Windows Admin Center.

Connect your tool extension to your custom gateway plugin

Now that you've created an extension with the Windows Admin Center SDK, you're ready to connect your tool extension to your custom gateway plugin, by following these steps:

- Add an empty module
- Use your custom gateway plugin in your tool extension

Build and side load your extension

Next, build and side load your extension into Windows Admin Center. Open a command window, change directory to your source directory, then you're ready to build.

- Build and serve with gulp:

```
gulp build
gulp serve -p 4201
```

The port you choose must currently be free. Make sure you don't attempt to use the port that Windows Admin Center is running on.

Your project can be side loaded into a local instance of Windows Admin Center for testing by attaching the locally served project into Windows Admin Center.

- Launch Windows Admin Center in a web browser
- Open the debugger (F12)
- Open the Console and type the following command:

```
MsftSme.sideLoad("http://localhost:4201")
```

- Refresh the web browser

Your project will now be visible in the Tools list with (side loaded) next to the name.

Target a different version of the Windows Admin Center SDK

Keeping your extension up to date with SDK changes and platform changes is easy. Read about how to target a different version of the Windows Admin Center SDK.

Feedback

Was this page helpful? Yes No

Provide product feedback

Target a different version of the Windows Admin Center SDK

Article • 11/01/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Keeping your extension up to date with SDK changes and platform changes is easy. We use NuGet Package Manager tags to organize the release of new features into SDK versions.

There are two SDK versions you can choose from and three which are deprecated:

- `latest` – this SDK package aligns with the current GA release of Windows Admin Center and is the most stable
- `experimental` – this SDK package contains the most recent changes and functionality, but might be unstable
- `insider` – this SDK package has been deprecated, use latest or experimental instead
- `next` – this SDK package has been deprecated, use latest or experimental instead
- `legacy` – this SDK package has been deprecated, use latest or experimental instead

> **Note**
>
> Find out more about the different versions of Windows Admin Center that are available to download.

Targeting SDK version on a new project

When creating a new extension, you can include the `--version` parameter to target a different version of the SDK:

```
wac create --company "{!Company Name}" --tool "{!Tool Name}" --version
{!version}
```

Expand table

Value	Explanation	Example
`{!Company Name}`	Your company name (with spaces)	`Contoso Inc`
`{!Tool Name}`	Your tool name (with spaces)	`Manage Foo Works`
`{!version}`	SDK Version	`latest`

Here's an example creating a new extension targeting `experimental`:

```
wac create --company "Contoso Inc" --tool "Manage Foo Works" --version
experimental
```

> **Note**
>
> We recommend developers building new extensions use the `latest` SDK version for stability.

Targeting SDK version on an existing project

To modify an existing project to target a different SDK version, modify the following line in `package.json`:

```
"@microsoft/windows-admin-center-sdk": "latest",
```

In this example, replace `latest` with your desired SDK version, i.e. `experimental`:

```
"@microsoft/windows-admin-center-sdk": "experimental",
```

Then run `npm install` to update references throughout your project.

Feedback

Was this page helpful?

Yes No

Provide product feedback

Guides

Article • 11/01/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Guides

Here are some guides for developing with the Windows Admin Center SDK:

- Add a module to a tool extension
- Add an iFrame to a tool extension
- Use a custom gateway plugin in your tool extension
- Create a connection provider
- Modify root navigation behavior
- Control your tool's visibility
- Strings and localization
- Using PowerShell in your extension
- CSS icon guide

SDK design toolkit

- Check out our Windows Admin Center SDK design toolkit! This toolkit is designed to help you rapidly mock up extensions in PowerPoint using Windows Admin Center styles, controls, and page templates. See what your extension can look like in Windows Admin Center before you start coding!

Sample code included with the SDK

- Sample code can be found for tool, solution, and gateway plugin extension types in our SDK documentation. There you will leverage the Windows Admin Center CLI to build a new extension project, then follow the individual guides to customize your project to meet your needs.
- Developer Tools, hosted on our GitHub SDK site, is a solution extension containing a rich collection of controls that you can browse and use in your own extension. Developer Tools is a fully functioning extension that can be side-loaded into Windows Admin Center in Developer Mode.

Feedback

Was this page helpful? Yes No

Provide product feedback

Add a module to a tool extension

Article • 11/01/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

In this article, add an empty module to a tool extension we've created with the Windows Admin Center CLI.

Prepare your environment

If you haven't already done it, follow the directions in develop a tool (or solution) extension to prepare your environment and create a new, empty tool extension.

Use the Angular CLI to create a module (and component)

If you're new to Angular, we encourage you to read the documentation on the Angular website to learn about Angular and NgModule. See NgModule for guidance.

To learn more:

- Generating a new module in Angular CLI.
- Generating a new component in Angular CLI.

Open a command prompt, change directory to .\src\app in your project, and then run the following commands, replacing `{!ModuleName}` with your module name (spaces removed).

```
cd .\src\app
ng generate module {!ModuleName}
ng generate component {!ModuleName}
```

Expand table

Value	Explanation	Example
`{!ModuleName}`	Your module name (spaces removed)	`ManageFooWorksPortal`

Example usage:

PowerShell

```
cd .\src\app
ng generate module ManageFooWorksPortal
ng generate component ManageFooWorksPortal
```

Add routing information

If you're new to Angular, we recommended you learn about Angular Routing and Navigation. The following sections define necessary routing elements that enable Windows Admin Center to navigate to your extension and between views in your extension in response to user activity. To learn more, see the Router guidance

Use the same module name that you used in the preceding step.

Add content to new routing file

1. Browse to the module folder created by `ng generate` in the previous step.

2. Create a new file `{!module-name}.routing.ts`, following this naming convention:

Expand table

Value	Explanation	Example filename
`{!module-name}`	Your module name (lower case, spaces replaced with dashes)	`manage-foo-works-portal.routing.ts`

3. Add this content to the file created:

ts

```ts
import { NgModule } from '@angular/core';
import { RouterModule, Routes } from '@angular/router';
import { {!ModuleName}Component } from './{!module-name}.component';

const routes: Routes = [
    {
        path: '',
        component: {!ModuleName}Component,
        // if the component has child components that need to be routed
to, include them in the children array.
        children: [
            {
                path: '',
                redirectTo: 'base',
                pathMatch: 'full'
            }
```

```
        ]
}];

@NgModule({
    imports: [
        RouterModule.forChild(routes)
    ],
    exports: [
        RouterModule
    ]
})
export class Routing { }
```

4. Replace values in the file created with your desired values:

Expand table

Value	Explanation	Example
`{!ModuleName}`	Your module name (spaces removed)	`ManageFooWorksPortal`
`{!module-name}`	Your module name (lower case, spaces replaced with dashes)	`manage-foo-works-portal`

Add content to new module file

1. Open file `{!module-name}.module.ts`, found with the following naming convention:

Expand table

Value	Explanation	Example filename
`{!module-name}`	Your module name (lower case, spaces replaced with dashes)	`manage-foo-works-portal.module.ts`

2. Add content to the file:

ts

```
import { Routing } from './{!module-name}.routing';
```

3. Replace values in the content just added with your desired values:

Expand table

Value	Explanation	Example
`{!module-name}`	Your module name (lower case, spaces replaced with dashes)	`manage-foo-works-portal`

4. Modify the imports statement to import Routing:

Expand table

Original value	New value
`imports: [ CommonModule ]`	`imports: [ CommonModule, Routing ]`

5. Make sure `import` statements are alphabetized by source.

Add content to new component TypeScript file

1. Open file `{!module-name}.component.ts`, found with the following naming convention:

Expand table

Value	Explanation	Example filename
`{!module-name}`	Your module name (lower case, spaces replaced with dashes)	`manage-foo-works-portal.component.ts`

2. Modify content in the file to match the following example.

```ts
constructor() {
    // TODO
}

public ngOnInit() {
    // TODO
}
```

Update app-routing.module.ts

1. Open file `app-routing.module.ts`, and modify the default path so it loads the new module you created. Find the entry for `path: ''`, and update `loadChildren` to load your module instead of the default module:

Expand table

Value	Explanation	Example
`{!ModuleName}`	Your module name (spaces removed)	`ManageFooWorksPortal`
`{!module-name}`	Your module name (lower case, spaces replaced with dashes)	`manage-foo-works-portal`

ts

```
    {
        path: '',
        loadChildren: 'app/{!module-name}/{!module-name}.module#
{!ModuleName}Module'
    },
```

Here's an example of an updated default path:

ts

```
    {
        path: '',
        loadChildren: 'app/manage-foo-works-portal/manage-foo-works-
portal.module#ManageFooWorksPortalModule'
    },
```

Build and side load your extension

You have now added a module to your extension. Next, you can build and side load your extension in Windows Admin Center to see the results.

Feedback

Was this page helpful? Yes No

Provide product feedback

Add an iFrame to a tool extension

Article • 11/01/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

In this article, we will add an iFrame to a new, empty tool extension we have created with the Windows Admin Center CLI.

Prepare your environment

If you haven't already, follow the directions in develop a tool extension to prepare your environment and create a new, empty tool extension.

Add a module to your project

Add a new empty module to your project, to which we will add an iFrame in the next step.

Add an iFrame to your module

Now we'll add an iFrame to that new, empty module that we just created.

In \src\app, browse into your module folder, then open file `{!module-name}.component.html`, found with the following naming convention:

Expand table

Value	Explanation	Example filename
`{!module-name}`	Your module name (lower case, spaces replaced with dashes)	`manage-foo-works-portal.component.html`

Add the following content to the html file:

HTML

```
<div>
  <iframe  style="height: 850px;" src="https://www.bing.com"></iframe>
</div>
```

That's it, you've added an iFrame to your extension. Next, you can build and side load your extension in Windows Admin Center to see the results.

> **Note**
>
> Content Security Policy (CSP) settings could prevent some sites from rendering in an iFrame within Windows Admin Center. You can learn more about this here.

Feedback

Was this page helpful? Yes No

Provide product feedback

Use a custom gateway plugin in your tool extension

Article • 11/01/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

In this article, we will use a custom gateway plugin in a new, empty tool extension we have created with the Windows Admin Center CLI.

Prepare your environment

If you haven't already, follow the directions in develop a tool extension to prepare your environment and create a new, empty tool extension.

Add a module to your project

If you haven't already, add a new empty module to your project, which we will use in the next step.

Add integration to custom gateway plugin

Now we'll use a custom gateway plugin in the new, empty module that we just created.

Create plugin.service.ts

Change to the directory of the new tool module created above (`\src\app\{!Module-Name}`), and create a new file `plugin.service.ts`.

Add the following code to the file just created:

ts

```ts
import { Injectable } from '@angular/core';
import { AppContextService, HttpService } from '@microsoft/windows-admin-center-sdk/angular';
import { Cim, Http, PowerShell, PowerShellSession } from '@microsoft/windows-admin-center-sdk/core';
import { AjaxResponse, Observable } from 'rxjs';

@Injectable()
export class PluginService {
    constructor(private appContextService: AppContextService, private http: Http) {
```

```
    }

    public getGatewayRestResponse(): Observable<any> {
        let callUrl = this.appContextService.activeConnection.nodeName;

        return this.appContextService.node.get(callUrl,
'features/Sample%20Uno').map(
            (response: any) => {
                return response;
            }
        )
    }
}
```

Change references to `Sample Uno` and `Sample%20Uno` to your feature name as appropriate.

> ⚠ **Warning**
>
> It is recommended that the built in `this.appContextService.node` is used for calling any API that is defined in your custom gateway plugin. This will ensure that if credentials are required inside of your gateway plugin that they will be handled properly.

Modify module.ts

Open the `module.ts` file of the new module created earlier (i.e. `{!Module-Name}.module.ts`):

Add the following import statements:

ts

```ts
import { HttpService } from '@microsoft/windows-admin-center-sdk/angular';
import { Http } from '@microsoft/windows-admin-center-sdk/core';
import { PluginService } from './plugin.service';
```

Add the following providers (after declarations):

ts

```ts
  ,
  providers: [
    HttpService,
    PluginService,
```

```
    Http
  ]
```

Modify component.ts

Open the `component.ts` file of the new module created earlier (i.e. `{!Module-Name}.component.ts`):

Add the following import statements:

ts

```ts
import { ActivatedRouteSnapshot } from '@angular/router';
import { AppContextService } from '@microsoft/windows-admin-center-sdk/angular';
import { Subscription } from 'rxjs';
import { Strings } from '../../generated/strings';
import { PluginService } from './plugin.service';
```

Add the following variables:

ts

```ts
  private serviceSubscription: Subscription;
  private responseResult: string;
```

Modify the constructor and modify/add the following functions:

ts

```ts
  constructor(private appContextService: AppContextService, private plugin: PluginService) {
    //
  }

  public ngOnInit() {
    this.responseResult = 'click go to do something';
  }

  public onClick() {
    this.serviceSubscription = this.plugin.getGatewayRestResponse().subscribe(
      (response: any) => {
        this.responseResult = 'response: ' + response.message;
      },
      (error) => {
        console.log(error);
      }
```

```
    );
  }
```

Modify component.html

Open the `component.html` file of the new module created earlier (i.e. `{!Module-Name}.component.html`):

Add the following content to the html file:

HTML

```
<button (click)="onClick()" >go</button>
{{ responseResult }}
```

Build and side load your extension

Now you are ready to build and side load your extension in Windows Admin Center.

Feedback

Was this page helpful? Yes No

Provide product feedback

Create a connection provider for a solution extension

Article • 12/23/2021 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Connection Providers play an important role in how Windows Admin Center defines and communicates with connectable objects, or targets. Primarily, a Connection Provider performs actions while a connection is being made, such as ensuring that the target is online and available, and also ensuring that the connecting user has permission to access the target.

By default, Windows Admin Center ships with the following Connection Providers:

- Server
- Windows Client
- Failover Cluster
- HCI Cluster

To create your own custom Connection Provider, follow these steps:

- Add Connection Provider details to `manifest.json`
- Define Connection Status Provider
- Implement Connection Provider in application layer

Add Connection Provider details to manifest.json

Now we'll walk through what you need to know to define a Connection Provider in your project's `manifest.json` file.

Create entry in manifest.json

The `manifest.json` file is located in the \src folder and contains, among other things, definitions of entry points into your project. Types of entry points include Tools, Solutions, and Connection Providers. We'll be defining a Connection Provider.

A sample of a Connection Provider entry in manifest.json is below:

JSON

```
    {
      "entryPointType": "connectionProvider",
      "name": "addServer",
      "path": "/add",
      "displayName": "resources:strings:addServer_displayName",
      "icon": "sme-icon:icon-win-server",
```

```
      "description": "resources:strings:description",
      "connectionType": "msft.sme.connection-type.server",
      "connectionTypeName": "resources:strings:addServer_connectionTypeName",
      "connectionTypeUrlName": "server",
      "connectionTypeDefaultSolution": "msft.sme.server-manager!servers",
      "connectionTypeDefaultTool": "msft.sme.server-manager!overview",
      "connectionStatusProvider": {
        "powerShell": {
          "script": "## Get-My-Status ##\nfunction Get-Status()\n{\n# A function
like this would be where logic would exist to identify if a node is
connectable.\n$status = @{label = $null; type = 0; details = $null; }\n$caption
= \"MyConstCaption\"\n$productType = \"MyProductType\"\n# A result object needs
to conform to the following object structure to be interpreted properly by the
Windows Admin Center shell.\n$result = @{ status = $status; caption = $caption;
productType = $productType; version = $version }\n# DO FANCY LOGIC #\n# Once the
logic is complete, the following fields need to be populated:\n$status.label =
\"Display Thing\"\n$status.type = 0 # This value needs to conform to the
LiveConnectionStatusType enum. >= 3 represents a failure.\n$status.details =
\"success stuff\"\nreturn $result}\nGet-Status"
        },
        "displayValueMap": {
          "wmfMissing-label":
"resources:strings:addServer_status_wmfMissing_label",
          "wmfMissing-details":
"resources:strings:addServer_status_wmfMissing_details",
          "unsupported-label":
"resources:strings:addServer_status_unsupported_label",
          "unsupported-details":
"resources:strings:addServer_status_unsupported_details"
        }
      }
    },
```

An entry point of type "connnectionProvider" indicates to the Windows Admin Center shell that the item being configured is a provider that will be used by a Solution to validate a connection state. Connection Provider entry points contains a number of important properties, defined below:

Expand table

Property	Description
entryPointType	This is a required property. There are three valid values: "tool", "solution", and "connectionProvider".
name	Identifies the Connection Provider within the scope of a Solution. This value must be unique inside a full Windows Admin Center instance (not just a Solution).
path	Represents the URL path for the "Add Connection" UI, if it will be configured by the Solution. This value must map to a route that is configured in app-routing.module.ts file. When the Solution entry point is configured to use the connections rootNavigationBehavior, this route will load the module that

Property	Description
	is used by the Shell to display the Add Connection UI. More information available in the section on rootNavigationBehavior.
displayName	The value entered here is displayed on the right hand side of the shell, below the black Windows Admin Center bar when a user loads a Solution's connections page.
icon	Represents the icon used in the Solutions drop down menu to represent the Solution.
description	Enter a short description of the entry point.
connectionType	Represents the connection type that the provider will load. The value entered here will also be used in the Solution entry point to specify that the Solution can load those connections. The value entered here will also be used in Tool entry point(s) to indicate that the Tool is compatible with this type. This value entered here will also be used in the connection object that is submitted to the RPC call on the "Add window", in the application layer implementation step.
connectionTypeName	Used in the connections table to represent a connection that uses your Connection Provider. This is expected to be the plural name of the type.
connectionTypeUrlName	Used in creating the URL to represent the loaded Solution, after Windows Admin Center has connected to an instance. This entry is used after connections, and before the target. In this example, "connectionexample" is where this value appears in the URL: `http://localhost:6516/solutionexample/connections/connectionexample/con-fake1.corp.contoso.com`
connectionTypeDefaultSolution	Represents the default component that should be loaded by the Connection Provider. This value is a combination of: [a] The name of the extension package defined at the top of the manifest; [b] Exclamation point (!); [c] The Solution entry point name. For a project with name "msft.sme.mySample-extension", and a Solution entry point with name "example", this value would be "msft.sme.solutionExample-extension!example".
connectionTypeDefaultTool	Represents the default Tool that should be loaded on a successful connection. This property value is made up of two parts, similar to the connectionTypeDefaultSolution. This value is a combination of: [a] The name of the extension package defined at the top of the manifest; [b] Exclamation point (!); [c] The Tool entry point name for the Tool that should be loaded initially. For a project with name "msft.sme.solutionExample-extension", and a Solution entry point with name "example", this value would be "msft.sme.solutionExample-extension!example".
connectionStatusProvider	Please see section "Define Connection Status Provider"

Define Connection Status Provider

Connection Status Provider is the mechanism by which a target is validated to be online and available, also ensuring that the connecting user has permission to access the target. There are currently two types of Connection Status Providers: PowerShell, and RelativeGatewayUrl.

- **PowerShell Connection Status Provider** - Determines if a target is online and accessible with a PowerShell script. The result must be returned in an object with a single property "status", defined below.
- **RelativeGatewayUrl Connection Status Provider** - Determines if a target is online and accessible with a rest call. The result must be returned in an object with a single property "status", defined below.

Define status

Connection Status Providers are required to return an object with a single property `status` that conforms to the following format:

JSON

```
{
    status: {
        label: string;
        type: int;
        details: string;
    }
}
```

Status properties:

- **Label** - A label describing the status return type. Note, values for label can be mapped in runtime. See entry below for mapping values in runtime.

- **Type** - The status return type. Type has the following enumeration values. For any value 2 or above, the platform will not navigate to the connected object, and an error will be displayed in the UI.

  Types:

Expand table

Value	Description
0	Online
1	Warning

Value	Description
2	Unauthorized
3	Error
4	Fatal
5	Unknown

- **Details** - Additional details describing the status return type.

PowerShell Connection Status Provider script

The Connection Status Provider PowerShell script determines if a target is online and accessible with a PowerShell script. The result must be returned in an object with a single property "status". An example script is shown below.

Example PowerShell script:

PowerShell

```
## Get-My-Status ##

function Get-Status()
{
    # A function like this would be where logic would exist to identify if a
node is connectable.
    $status = @{label = $null; type = 0; details = $null; }
    $caption = "MyConstCaption"
    $productType = "MyProductType"

    # A result object needs to conform to the following object structure to be
interperated properly by the Windows Admin Center shell.
    $result = @{ status = $status; caption = $caption; productType =
$productType; version = $version }

    # DO FANCY LOGIC #

    # Once the logic is complete, the following fields need to be populated:
    $status.label = "Display Thing"
    $status.type = 0 # This value needs to conform to the
LiveConnectionStatusType enum. >= 3 represents a failure.
    $status.details = "success stuff"

    return $result
}

Get-Status
```

Define RelativeGatewayUrl Connection Status Provider method

The Connection Status Provider `RelativeGatewayUrl` method calls a rest API to determine if a target is online and accessible. The result must be returned in an object with a single property "status". An example Connection Provider entry in manifest.json of a RelativeGatewayUrl is shown below.

JSON

```
    {
      "entryPointType": "connectionProvider",
      "name": "addServer",
      "path": "/add/server",
      "displayName": "resources:strings:addServer_displayName",
      "icon": "sme-icon:icon-win-server",
      "description": "resources:strings:description",
      "connectionType": "msft.sme.connection-type.server",
      "connectionTypeName": "resources:strings:addServer_connectionTypeName",
      "connectionTypeUrlName": "server",
      "connectionTypeDefaultSolution": "msft.sme.server-manager!servers",
      "connectionTypeDefaultTool": "msft.sme.server-manager!overview",
      "connectionStatusProvider": {
        "relativeGatewayUrl": "<URL here post /api>",
        "displayValueMap": {
          "wmfMissing-label":
"resources:strings:addServer_status_wmfMissing_label",
          "wmfMissing-details":
"resources:strings:addServer_status_wmfMissing_details",
          "unsupported-label":
"resources:strings:addServer_status_unsupported_label",
          "unsupported-details":
"resources:strings:addServer_status_unsupported_details"
        }
      }
    },
```

Notes about using RelativeGatewayUrl:

- "relativeGatewayUrl" specifies where to get the connection status from a gateway URL. This URI is relative from /api. If $connectionName is found in the URL, it will be replaced with the name of the connection.
- All relativeGatewayUrl properties must be executed against the host gateway, which can be accomplished by creating a gateway extension

Map values in runtime

The label and details values in the status return object can be formatted at tune time by including keys and values in the "defaultValueMap" property of the provider.

For example, if you add the value below, any time that "defaultConnection_test" showed up as a value for either label or details, Windows Admin Center will automatically replace the key with the configured resource string value.

JSON

```
    "defaultConnection_test":
"resources:strings:addServer_status_defaultConnection_label"
```

Implement Connection Provider in application layer

Now we're going to implement the Connection Provider in the application layer, by creating a TypeScript Class that implements OnInit. The class has the following functions:

Expand table

Function	Description
constructor(private appContextService: AppContextService, private route: ActivatedRoute)	
public ngOnInit()	
public onSubmit()	Contains logic to update shell when an add connection attempt is made
public onCancel()	Contains logic to update shell when an add connection attempt is canceled

Define onSubmit

`onSubmit` issues an RPC call back to the app context to notify the shell of an "Add Connection". The basic call uses "updateData" like this:

ts

```
this.appContextService.rpc.updateData(
    EnvironmentModule.nameOfShell,
    '##',
    <RpcUpdateData>{
        results: {
            connections: connections,
            credentials: this.useCredentials ? this.creds : null
        }
```

```
    }
);
```

The result is a connection property, which is an array of objects that conform to the following structure:

ts

```ts
/**
 * The connection attributes class.
 */
export interface ConnectionAttribute {

    /**
     * The id string of this attribute
     */
    id: string;

    /**
     * The value of the attribute. used for attributes that can have variable values such as Operating System
     */
    value?: string | number;
}

/**
 * The connection class.
 */
export interface Connection {

    /**
     * The id of the connection, this is unique per connection
     */
    id: string;

    /**
     * The type of connection
     */
    type: string;

    /**
     * The name of the connection, this is unique per connection type
     */
    name: string;

    /**
     * The property bag of the connection
     */
    properties?: ConnectionProperties;

    /**
     * The ids of attributes identified for this connection
     */
    attributes?: ConnectionAttribute[];
```

```
    /**
     * The tags the user(s) have assigned to this connection
     */
    tags?: string[];
}

/**
 * Defines connection type strings known by core
 * Be careful that these strings match what is defined by the manifest of @msft-sme/server-manager
 */
export const connectionTypeConstants = {
    server: 'msft.sme.connection-type.server',
    cluster: 'msft.sme.connection-type.cluster',
    hyperConvergedCluster: 'msft.sme.connection-type.hyper-converged-cluster',
    windowsClient: 'msft.sme.connection-type.windows-client',
    clusterNodesProperty: 'nodes'
};
```

Define onCancel

`onCancel` cancels an "Add Connection" attempt by passing an empty connections array:

ts

```
this.appContextService.rpc.updateData(EnvironmentModule.nameOfShell, '##', <RpcUpdateData>{ results: { connections: [] } });
```

Connection Provider example

The full TypeScript class for implementing a connection provider is below. Note that the "connectionType" string matches the "connectionType as defined in the connection provider in manifest.json.

ts

```
import { Component, OnInit } from '@angular/core';
import { ActivatedRoute } from '@angular/router';
import { AppContextService } from '@microsoft/windows-admin-center-sdk/shell/angular';
import { Connection, ConnectionUtility } from '@microsoft/windows-admin-center-sdk/shell/core';
import { EnvironmentModule } from '@microsoft/windows-admin-center-sdk/shell/dist/core/manifest/environment-modules';
import { RpcUpdateData } from '@microsoft/windows-admin-center-sdk/shell/dist/core/rpc/rpc-base';
import { Strings } from '../../generated/strings';

@Component({
```

```
  selector: 'add-example',
  templateUrl: './add-example.component.html',
  styleUrls: ['./add-example.component.css']
})
export class AddExampleComponent implements OnInit {
  public newConnectionName: string;
  public strings = MsftSme.resourcesStrings<Strings>().SolutionExample;
  private connectionType = 'msft.sme.connection-type.example'; // This needs to
match the connectionTypes value used in the manifest.json.

  constructor(private appContextService: AppContextService, private route:
ActivatedRoute) {
    // TODO:
  }

  public ngOnInit() {
    // TODO
  }

  public onSubmit() {
    let connections: Connection[] = [];

    let connection = <Connection> {
      id: ConnectionUtility.createConnectionId(this.connectionType,
this.newConnectionName),
      type: this.connectionType,
      name: this.newConnectionName
    };

    connections.push(connection);

    this.appContextService.rpc.updateData(
      EnvironmentModule.nameOfShell,
      '##',
      <RpcUpdateData> {
        results: {
          connections: connections,
          credentials: null
        }
      }
    );
  }

  public onCancel() {
    this.appContextService.rpc.updateData(
      EnvironmentModule.nameOfShell, '##', <RpcUpdateData>{ results: {
connections: [] } });
  }
}
```

Feedback

Was this page helpful? Yes No

Provide product feedback

Modify root navigation behavior for a solution extension

Article • 05/22/2023 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

This article provides guidance on how to modify the root navigation behavior for your solution to have different connection list behavior. You'll also learn how to hide or show the tools list.

Modifying root navigation behavior

Open manifest.json file in {extension root}\src, and find the property "rootNavigationBehavior". This property has two valid values: "connections" or "path". The "connections" behavior is detailed later in the documentation.

Setting path as a rootNavigationBehavior

Set the value of `rootNavigationBehavior` to `path`, and then delete the `requirements` property, and leave the `path` property as an empty string. You've completed the minimal required configuration to build a solution extension. Save the file, gulp build -> gulp serve as you would with a tool, and then side load the extension into your local Windows Admin Center extension.

A valid manifest entryPoints array looks like this:

```
"entryPoints": [
    {
      "entryPointType": "solution",
      "name": "main",
      "urlName": "testsln",
      "displayName": "resources:strings:displayName",
      "description": "resources:strings:description",
      "icon": "sme-icon:icon-win-powerShell",
      "path": "",
      "rootNavigationBehavior": "path"
    }
],
```

Tools built with this kind of structure don't require connections to load, but don't have node connectivity functionality either.

Setting connections as a rootNavigationBehavior

When you set the `rootNavigationBehavior` property to `connections`, you're telling the Windows Admin Center Shell that there's a connected node (always a server of some type) that it should connect to verify connection status. There are two steps in verifying a connection.

1. Windows Admin Center attempts to make an attempt to log into the node with your credentials (for establishing the remote PowerShell session).
2. Windows Admin Center executes the PowerShell script you provide to verify if the node is in a connectable state.

A valid solution definition with connections looks like this:

JSON

```
        {
          "entryPointType": "solution",
          "name": "example",
          "urlName": "solutionexample",
          "displayName": "resources:strings:displayName",
          "description": "resources:strings:description",
          "icon": "sme-icon:icon-win-powerShell",
          "rootNavigationBehavior": "connections",
          "connections": {
            "header": "resources:strings:connectionsListHeader",
            "connectionTypes": [
                "msft.sme.connection-type.example"
                ]
            },
            "tools": {
                "enabled": false,
                "defaultTool": "solution"
            }
        },
```

When the rootNavigationBehavior is set to "connections", you're required to build out the connections definition in the manifest. This includes the "header" property (displays in your solution header when a user selects it from the menu), and a connectionTypes array (specifies which connectionTypes are used in the solution. More on this in the connectionProvider documentation.

Enabling and disabling the tools menu

Another property available in the solution definition is the Tools property. The Tools property decides whether the Tools menu is displayed, and which tool will be loaded.

When enabled, Windows Admin Center renders the left hand Tools menu. With defaultTool, it's required that you add a tool entry point to the manifest in order to load the appropriate resources. The value of "defaultTool" needs to be the "name" property of the tool as it's defined in the manifest.

Feedback

Was this page helpful? Yes No

Provide product feedback

Control your tool's visibility in a solution

Article • 12/23/2021 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

There might be times when you want to exclude (or hide) your extension or tool from the available tools list. For example, if your tool targets only Windows Server 2016 (not older versions), you might not want a user who connects to a Windows Server 2012 R2 server to see your tool at all. (Imagine the user experience - they click on it, wait for the tool to load, only to get a message that its features aren't available for their connection.) You can define when to show (or hide) your feature in the tool's manifest.json file.

Options for deciding when to show a tool

There are three different options you can use to determine whether your tool should be displayed and available for a specific server or cluster connection.

- localhost
- inventory (an array of properties)
- script

LocalHost

The localHost property of the Conditions object contains a boolean value that can be evaluated to infer if the connecting node is localHost (the same computer that Windows Admin Center is installed on) or not. By passing a value to the property, you indicate when (the condition) to display the tool. For example if you only want the tool to display if the user is in fact connecting to the local host, set it up like this:

JSON

```
"conditions": [
{
    "localhost": true
}]
```

Alternatively, if you only want your tool to display when the connecting node *is not* localhost:

JSON

```
"conditions": [
{
```

```
    "localhost": false
}]
```

Here's what the configuration settings look like to only show a tool when the connecting node is not localhost:

JSON

```
"entryPoints": [
{
    "entryPointType": "tool",
    "name": "main",
    "urlName": "processes",
    "displayName": "resources:strings:displayName",
    "description": "resources:strings:description",
    "icon": "sme-icon:icon-win-serverProcesses",
    "path": "",
    "requirements": [
    {
        "solutionIds": [
        "msft.sme.server-manager!windowsClients"
        ],
        "connectionTypes": [
        "msft.sme.connection-type.windows-client"
        ],
        "conditions": [
        {
            "localhost": true
        }
        ]
    }
    ]
}
```

Inventory properties

The SDK includes a pre-curated set of inventory properties that you can use to build conditions to determine when your tool should be available or not. There are nine different properties in the 'inventory' array:

Expand table

Property Name	Expected Value Type
computerManufacturer	string
operatingSystemSKU	number

Property Name	Expected Value Type
operatingSystemVersion	version_string (eg: "10.1.*")
productType	number
clusterFqdn	string
isHyperVRoleInstalled	boolean
isHyperVPowershellInstalled	boolean
isManagementToolsAvailable	boolean
isWmfInstalled	boolean

Every object in the inventory array must conform to the following json structure:

JSON

```
"<property name>": {
    "type": "<expected type>",
    "operator": "<defined operator to use>",
    "value": "<expected value to evaluate using the operator>"
}
```

Operator values

Expand table

Operator	Description
gt	greater than
ge	greater than or equal to
lt	less than
le	less than or equal to
eq	equal to
ne	not equal to
is	checking if a value is true
not	checking if a value is false
contains	item exists in a string

Operator	Description
notContains	item does not exist in a string

Data types

Available options for the 'type' property:

Expand table

Type	Description
version	a version number (eg: 10.1.*)
number	a numeric value
string	a string value
boolean	true or false

Value types

The 'value' property accepts these types:

- string
- number
- boolean

A properly-formed inventory condition set looks like this:

JSON

```
"entryPoints": [
{
    "entryPointType": "tool",
    "name": "main",
    "urlName": "processes",
    "displayName": "resources:strings:displayName",
    "description": "resources:strings:description",
    "icon": "sme-icon:icon-win-serverProcesses",
    "path": "",
    "requirements": [
    {
        "solutionIds": [
        "msft.sme.server-manager!servers"
        ],
        "connectionTypes": [
        "msft.sme.connection-type.server"
```

```
        ],
        "conditions": [
        {
            "inventory": {
            "operatingSystemVersion": {
                "type": "version",
                "operator": "gt",
                "value": "6.3"
            },
            "operatingSystemSKU": {
                "type": "number",
                "operator": "eq",
                "value": "8"
            }
            }
        }
        ]
    }
    ]
}
```

Script

Finally, you can run a custom PowerShell script to identify the availability and state of the node. All scripts must return an object with the following structure:

ps

```
@{
    State = 'Available' | 'NotSupported' | 'NotConfigured';
    Message = '<Message to explain the reason of state such as not supported
and not configured.>';
    Properties =
        @{ Name = 'Prop1'; Value = 'prop1 data'; Type = 'string' },
        @{Name='Prop2'; Value = 12345678; Type='number'; };
}
```

The State property is the important value that will control the decision to show or hide your extension in the tools list. The allowed values are:

Expand table

Value	Description
Available	The extension should be displayed in the tools list.
NotSupported	The extension should not be displayed in the tools list.

Value	Description
NotConfigured	This is a placeholder value for future work that will prompt the user for additional configuration before the tool is made available. Currently this value will result in the tool being displayed and is the functional equivalent to 'Available'.

For example, if we want a tool to load only if the remote server has BitLocker installed, the script looks like this:

ps

```ps
$response = @{
    State = 'NotSupported';
    Message = 'Not executed';
    Properties = @{ Name = 'Prop1'; Value = 'prop1 data'; Type = 'string' },
        @{Name='Prop2'; Value = 12345678; Type='number'; };
}

if (Get-Module -ListAvailable -Name servermanager) {
    Import-module servermanager;
    $isInstalled = (Get-WindowsFeature -name bitlocker).Installed;
    $isGood = $isInstalled;
}

if($isGood) {
    $response.State = 'Available';
    $response.Message = 'Everything should work.';
}

$response
```

An entry point configuration using the script option looks like this:

JSON

```JSON
"entryPoints": [
{
    "entryPointType": "tool",
    "name": "main",
    "urlName": "processes",
    "displayName": "resources:strings:displayName",
    "description": "resources:strings:description",
    "icon": "sme-icon:icon-win-serverProcesses",
    "path": "",
    "requirements": [
    {
        "solutionIds": [
        "msft.sme.server-manager!windowsClients"
        ],
        "connectionTypes": [
```

```
        "msft.sme.connection-type.windows-client"
        ],
        "conditions": [
        {
            "localhost": true,
            "inventory": {
            "operatingSystemVersion": {
                "type": "version",
                "operator": "eq",
                "value": "10.0.*"
            },
            "operatingSystemSKU": {
                "type": "number",
                "operator": "eq",
                "value": "4"
            }
            },
            "script": "$response = @{ State = 'NotSupported'; Message = 'Not
executed'; Properties = @{ Name = 'Prop1'; Value = 'prop1 data'; Type =
'string' }, @{Name='Prop2'; Value = 12345678; Type='number'; }; }; if (Get-
Module -ListAvailable -Name servermanager) { Import-module servermanager;
$isInstalled = (Get-WindowsFeature -name bitlocker).Installed; $isGood =
$isInstalled; }; if($isGood) { $response.State = 'Available';
$response.Message = 'Everything should work.'; }; $response"
        }
        ]
    }
    ]
}
```

Supporting multiple requirement sets

You can use more than one set of requirements to determine when to display your tool by defining multiple "requirements" blocks.

For example, to display your tool if "scenario A" OR "scenario B" is true, define two requirements blocks; if either is true (that is, all conditions within a requirements block are met), the tool is displayed.

JSON

```
"entryPoints": [
{
    "requirements": [
    {
        "solutionIds": [
            …"scenario A"…
        ],
        "connectionTypes": [
            …"scenario A"…
```

```
        ],
        "conditions": [
            …"scenario A"…
        ]
    },
    {
        "solutionIds": [
            …"scenario B"…
        ],
        "connectionTypes": [
            …"scenario B"…
        ],
        "conditions": [
            …"scenario B"…
        ]
    }
    ]
}
```

Supporting condition ranges

You can also define a range of conditions by defining multiple "conditions" blocks with the same property, but with different operators.

When the same property is defined with different operators, the tool is displayed as long as the value is between the two conditions.

For example, this tool is displayed as long as the operating system is a version between 6.3.0 and 10.0.0:

JSON

```
"entryPoints": [
{
    "entryPointType": "tool",
    "name": "main",
    "urlName": "processes",
    "displayName": "resources:strings:displayName",
    "description": "resources:strings:description",
    "icon": "sme-icon:icon-win-serverProcesses",
    "path": "",
    "requirements": [
    {
        "solutionIds": [
            "msft.sme.server-manager!servers"
        ],
        "connectionTypes": [
            "msft.sme.connection-type.server"
        ],
```

```
        "conditions": [
        {
            "inventory": {
                "operatingSystemVersion": {
                    "type": "version",
                    "operator": "gt",
                    "value": "6.3.0"
                },
            }
        },
        {
            "inventory": {
                "operatingSystemVersion": {
                    "type": "version",
                    "operator": "lt",
                    "value": "10.0.0"
                }
            }
        }
        ]
    }
    ]
}
```

Feedback

Was this page helpful? Yes No

Provide product feedback

Strings and localization in Windows Admin Center

Article • 11/01/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Let's go more in-depth into the Windows Admin Center Extensions SDK and talk about strings and localization.

To enable localization of all strings that are rendered on the presentation layer, take advantage of the strings.resjson file under /src/resources/strings - it's already set up. When you need to add a new string to your extension, add it to this resjson file as a new entry. The existing structure follows this format:

ts

```
"<YourExtensionName>_<Component>_<Accessor>": "Your string value goes here.",
```

You can use any format you like for the strings, but be aware that the generation process (the process that takes the resjson and outputs the usable TypeScript class) converts underscore (_) to periods (.).

For example, this entry:

ts

```
"HelloWorld_cim_title": "CIM Component",
```

Generates the following accessor structure:

ts

```
MsftSme.resourcesStrings<Strings>().HelloWorld.cim.title;
```

Add Other Languages for Localization

For localization to other languages, a strings.resjson file needs to be created for each language. These files need to be placed in `\loc\output\{!ExtensionName}\{!LanguageFolder}\strings.resjson`. The available languages with corresponding folders are:

Expand table

Language	Folder
Čeština	cs-CZ
Deutsch	de-DE
English	en-US
Español	es-ES
Français	fr-FR
Magyar	hu-HU
Italiano	it-IT
日本語	ja-JP
한국어	ko-KR
Nederlands	nl-NL
Polski	pl-PL
Português (Brasil)	pt-BR
Português (Portugal)	pt-PT
Русский	ru-RU
Svenska	sv-SE
Türkçe	tr-TR
中文(简体)	zh-CN
中文(繁體)	zh-TW

> **Note**
>
> If your file structure needs are different inside of loc/output, you will need to adjust the localeOffset for the gulp task 'generate-resjson-json-localized' that is in the gulpfile.js. This offset is how deep into the loc folder it should start searching for strings.resjson files.

Each strings.resjson file will be formatted in the same way as previously mentioned at the top of this guide.

For example, to include a localization for Español include this entry in `\loc\output\HelloWorld\es-ES\strings.resjson`:

JSON

```
"HelloWorld_cim_title": "CIM Componente",
```

Anytime that you added localized strings, gulp generate must be ran again in order to have them appear. Run:

Windows Command Prompt

```
gulp generate
```

To confirm that your strings have been generated navigate to `\src\app\assets\strings\{!LanguageFolder}\strings.resjson`. Your newly added entry will appear in this file. Now if you switch the language option in Windows Admin Center, you will be able to see the localized strings in your extension.

Feedback

Was this page helpful? Yes No

Provide product feedback

Windows Admin Center UI text and design style guide

Article • 09/19/2022 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

This topic describes the general approach to writing user interface (UI) text for the Windows Admin Center, as well as some specific conventions and approaches we're taking.

Windows Admin Center and any extensions should follow Microsoft's voice principles so that the experience is easy to use and friendly. This style guide builds on these voice principles as well as the Microsoft Writing Style Guide, so make sure to check out both of those resources for info on such things as accessibility, acronyms, and word choice such as please, and sorry.

Buttons

- Buttons should be one word whenever possible, especially if you plan to localize your tool. Two or three is OK but try to avoid longer. If you have four words or longer, it'd be better to use a link control.
- Button labels should be concise, specific, and self-explanatory. Instead of a generic "Submit" button, use a verb corresponding to the user action, such as "Create", "Delete", "Add", "Format", etc.
- If a button follows a question, its label should correspond clearly to the question (typically "Yes" or "No").
- When launching a create flow, use the appropriate button label:

Expand table

Button	Use
Create	Create a new resource/object/etc.
Add	Add an existing resources/object/etc to the tool.
Install	Installing software/extensions.

Capitalization

We follow the Microsoft style for Capitalization - use sentence-style capitalization for pretty much everything.

Expand table

UI element	Capitalization	Comments
Badges (such as PREVIEW)	All caps	
Everything else	Sentence-style	However, there are a few exceptions where we surface object properties from WMI or PowerShell that's outside of our control.

Colons

Use colons to introduce lists. For example:

Choose one of the following:

- Cats
- Dogs
- Quokkas

Don't use colons in UI text when a label is on a different line from the thing it labels or when there's a clear distinction between the label and the thing it's labeling.

Use colons in UI text when a label is on the same line as the text it labels and you need to keep the two elements from running together.

Confirmation messages

Confirmation dialogs are useful when continuing might have unexpected results, such as data loss. They should contain scannable, useful info with a clear outcome, especially for events that can't be reversed.

- Make sure a confirmation is necessary. If there's no new info to offer (for example, “Are you sure?”) then a confirmation message may not be necessary.
- Verify that the customer wants to proceed with the action.
- Make sure the main instruction (heading) and explanatory text (body) aren't redundant.
- In the heading, define the possible outcomes as a question or a statement about what will happen next. For example, “Erase all data on this drive? or “You're about

to erase all your data”.
- Add details in the body. If there's a variable, such as the name of the item you're about change, include it here.
- Include a simple question (either in the header or in the body) that frames a clear choice between two action buttons.
- For a complex choice, use Yes/No buttons, which encourage careful reading. For a simpler choice, use buttons that are specific to the action, such as Delete all or Cancel.

First-run experiences

The first time a user visits a page, you have an opportunity to help them get started with your tool. This could be:

- A text string in an empty page with short instructions on how to get started - for example, "Select 'Add' to add an app."
- A link to the control that gets the user started - for example, "Add an app to get started."
- A small and short animation or video showing the user how to get started

Here are some tips from our Windows style guide:

1. Be helpful

- Avoid marketing style and language.
- When you demo or suggest something, make sure the end result is clear; just showing the customer how to do something isn't effective if they don't know why they are doing it.
- Don't present tips if the customer doesn't need them.

2. Show, don't tell

Keep your text simple as possible (think small animations or videos).

3. Don't overwhelm

- Limit pop-ups and tips to 4 per usage session combined—including system notifications and shell notifications.
- Make sure the timing of pop-ups is helpful.
- Don't prevent the customer from doing something.
- Make sure pop-ups are easily dismissed.

4. Keep it contextual

- Teaching moments are most effective when presented at the right time.
- If you create tutorials or slideshows, keep the info concrete.
- No marketing “fluff”—focus on specific tips and tricks.
- Provide a way for customers to return to the tutorial later, if relevant (people often don't retain info the first time, but setup instructions might only be relevant once).
- Empty-state messaging is a natural place for learning and/or delight—keep it simple and informative.

5. Minimize painful setup

When you need the customer to perform another action to experience full value (sign-in to an online service, etc.), make it as painless as possible.

- Messaging should be short and direct.
- Avoid sending them away. If possible, provide a means to connect from where they are.
- If you can, allow the option to do it later, and then remind them to do it later.
- If you take them out of their experience, provide a way to switch back quickly and easily.

Help links

Here are some tips from our Windows style guide:

When should we provide a Help link?

Almost never. Provide a help link only when:

- There's an obvious and important question that customers are likely to have while they're in the UI the answer to which will help them succeed at the UI task.
- There's not enough room in the UI to provide the amount of information necessary for users to succeed at the UI task.

Where should help links appear?

- Text links should appear as close to the UI element that the help is directed at as possible.
- If you must provide a text link that applies to an entire UI screen, place it at the bottom left of the screen.

- If you provide a link through a Help button (?), the tooltip should be "Help."

What URL should we use?

Never link directly to a web address—instead use a redirection service.

Microsoft developers should use an FWLink except when it's a help link that users might have to manually type, in which case use an `aka.ms` link (as long as the target of the URL is a website that automatically recognizes the browser locale, such as `learn.microsoft.com`).

Text guidelines

- Use full sentences.
- Do not include ending punctuation except for question marks.
- You don't need to use the same text as the task title; use text that makes sense in the context of the UI, but make sure that there's a logical connection between the two. For example:
  - Help link: What are the risks of allowing exceptions?
  - Help topic title: "Allowing a program to communicate through Windows Firewall"
- Be as specific as possible about the content of the help topic.
  - Our style
    - How does Windows Firewall help protect my computer?
    - Why highlights can improve a picture
  - Not our style
    - More information about Windows Firewall
    - Learn more about color management
    - Learn more
- Use the entire sentence for the link text, not just the key words.
  - Our style
    - What are the risks of allowing exceptions?
  - Not our style
    - What are the risks of allowing exceptions?
- In some cases, it's OK to use a "Learn more" link if the context is clear what the user will be getting when they click the link.

Error messages

Here's some guidance adapted from the Windows Style Guide:

Writing a good message is a balance between providing enough explanation but not being overly technical; between being casual and personable but not annoying or offensive.

General guidelines

Use one message per error case.

Headings

- Keep it brief and explain concisely what the problem is or **ideally what to do**. Some UI surfaces may have headings that truncate instead of wrapping when they're too long, so keep an eye out for these.
- Use the solution in the heading if it's a simple step.
- Make sure that the heading relates directly to the button in case the reader ignores the body text.
- Avoid using "There was a problem" in headings, unless you have no other choice. Be more specific about the problem.
- Avoid using variables (such as file, folder, and app names) in headings. Put them in the body.

Body

- If the heading sufficiently explains the problem or solution, you don't need body text.
- Don't repeat the title in the message with slightly different wording.
- Communicate clearly and concisely what the solution is.
- Focus on giving the facts first.
- Don't blame users for the error.
- If there is an error code associated with the error and if you think that including the error code might help the customer or Microsoft support to research the issue, include it directly below the body text and write it as follows:

  Error code: ####

  If the customer has all the info necessary to resolve the error without the code, you don't need to include it.

Buttons

- Write button text so that it's a specific response to the main instruction. If that's not possible, use "Close" for the dismissal button text (instead of "Okay" or "Done").
- If you have more than one button, make the leftmost button the action the user is encouraged to take. Make the rightmost button the more conservative action, such as "Cancel."

Help links

Only consider Help links for error messages that you can't make specific and actionable.

Null state text

Here's some help from the Windows Style Guide.

Null state occurs when customer data or content is absent from an app or feature, when no results are returned after a search, or when required information is missing from a form, such as billing information for a transaction.

Guidelines

- If possible, use null state situations as an opportunity to educate people about how to use the feature (for example, how to add music, where to find pictures, etc.)
  - If you have a title in your UI, explain the action to take to “fix” the null state (for example, “Add some music”)
  - Have fun with the text. This space can be an opportunity to provide delight since it will probably not be seen several times.
  - Avoid “It's lonely in here.” This is sad and has been overused.
  - Avoid questions like “Haven't connected your printer?” Okay to use once, but this format tends to get overused, and questions put extra burden/pressure on the customer. It can also feel condescending.
  - Variety in null state text is a good thing.

Examples

- "Add someone as a favorite, and you'll see them here."
- "Got any achievements or game clips you're particularly proud of? Add them to your showcase."
- "No one's in a party yet. Start one up!"

- "When someone adds you as a friend, you'll see them here."
- "When you do stuff like unlock achievements, record game clips, and add friends, you'll see it all here."
- "Your favorite friends will show up here, so you can see when they're online and what they're up to."

Punctuation

- No ending punctuation (periods, question marks) for headings or incomplete sentences. An exception is in a confirmation dialog where the heading asks the question
- Use Microsoft Style Guide's guidance on periods and question marks.

Status messages

Status messages consist of pop-up (toast) messages and notifications.

Expand table

String type	Notes
Toast	Sentence case with ending punctuation - ideally with an object variable so users can understand what object the message applies to in case they've navigated away from the object
Notification heading (title)	Sentence case without ending punctuation (it's a heading) - ideally with an object variable
Notification details	Full sentences, ideally with a link to the UI that displays the object

Here are some detailed recommendations for notification messages:

Expand table

String type	Notes
Started	Omit when possible - usually you can just skip to the in-progress message to minimize the number of distractions.
In progress	Start with the verb of the action you're performing and end with ellipses to indicate an ongoing operation. Here's an example: *Creating the volume 'Customer data'...*

String type	Notes
	When there are multiple variables, use this pattern: *Deleting the following virtual machine: {0}; Host: {1}*
Success	Start with "Successfully" and end with what the software just did. Here's an example: *Successfully created the volume 'Customer data'*.
Failure	Start with "Couldn't" and end with what the software couldn't do. Here's an example: *Couldn't create the volume 'Customer data'*.

Tooltips

Good tooltips briefly describe unlabeled controls or provide a bit of additional info for labeled controls, when this is useful. They can also help customers navigate the UI by offering additional—not redundant—information about control labels, icons, links, etc.

Tooltips should be used sparingly or not at all. They can be an interruption to the customer, so don't include a tooltip that simply repeats a label or states the obvious. It should always add valuable info.

Expand table

Context	How to write the tooltips
When a control or UI element is unlabeled...	Use a simple, descriptive noun phrase. For example: Highlighting pen
When a UI element is labeled, but its purpose needs clarification...	• Briefly describe what you can do with this UI element. • Use the imperative verb form. For example, "Find text in this file" (not "Finds text in this file"). • Don't include end punctuation unless there are multiple complete sentences.
When a text label is truncated or likely to truncate in some languages...	• Provide the untruncated label in the tooltip. • Optional: On another line, provide a clarifying description, but only if needed. • Don't provide a tooltip if the untruncated info is provided elsewhere on the page or flow.
If a keyboard shortcut is available...	• Optional: Provide the keyboard shortcut in parentheses following the label or descriptive phrase, e.g. "Print (Ctrl+P)" or "Find text in this file (Ctrl+F)" • It's OK to add a useful keyboard shortcut to a clarifying tooltip, but avoid adding a tooltip only to show a

Context	How to write the tooltips
	keyboard shortcut.

Feedback

Was this page helpful? Yes No

Provide product feedback

Using PowerShell in your extension

Article • 12/23/2021 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Let's go more in-depth into the Windows Admin Center Extensions SDK - let's talk about adding PowerShell commands to your extension.

PowerShell in TypeScript

The gulp build process has a generate step that will take any `{!ScriptName}.ps1` that is placed in the `\src\resources\scripts` folder and build them into the `powershell-scripts` class under the `\src\generated` folder.

> **Note**
>
> Don't manually update the `powershell-scripts.ts` nor the `strings.ts` files. Any change you make will be overwritten on the next generate.

Running a PowerShell Script

Any scripts that you want to run on a node can be placed in `\src\resources\scripts\{!ScriptName}.ps1`.

> **Important**
>
> Any changes make in a `{!ScriptName}.ps1` file will not be reflected in your project until `gulp generate` has been run.

The API works by first creating a PowerShell session on the nodes you are targeting, creating the PowerShell script with any parameters that need to be passed in, and then running the script on the sessions that were created.

For example, we have this script `\src\resources\scripts\Get-NodeName.ps1`:

ps1

```
Param
 (
    [String] $stringFormat
 )
```

```
$nodeName = [string]::Format($stringFormat,$env:COMPUTERNAME)
Write-Output $nodeName
```

We will create a PowerShell session for our target node:

ts

```ts
const session =
this.appContextService.powerShell.createSession('{!TargetNode}');
```

Then we will create the PowerShell script with an input parameter:

ts

```ts
const command = PowerShell.createCommand(PowerShellScripts.Get_NodeName,
{stringFormat: 'The name of the node is {0}!'});
```

Lastly, we need to run that script in the session we created:

ts

```ts
  public ngOnInit(): void {
    this.session =
this.appContextService.powerShell.createAutomaticSession('{!TargetNode}');
  }

  public getNodeName(): Observable<any> {
    const command = PowerShell.createCommand(PowerShellScripts.Get_NodeName,
{ stringFormat: 'The name of the node is {0}!'});
    return this.appContextService.powerShell.run(this.session, command)
    .pipe(
        map(
        response => {
            if (response && response.results) {
                return response.results;
            }
            return 'no response';
        }
      )
    );
  }

  public ngOnDestroy(): void {
    this.session.dispose()
  }
```

Now we will need to subscribe to the observable function we just created. Place this where you need to call the function to run the PowerShell script:

ts

```
this.getNodeName().subscribe(
     response => {
    console.log(response)
     }
);
```

By providing the node name to the createSession method, a new PowerShell session is created, used, and then immediately destroyed upon completion of the PowerShell call.

Key Options

A few options are available when calling the PowerShell API. Each time a session is created it can be created with or without a key.

Key: This creates a keyed session that can be looked up and reused, even across components (meaning that Component 1 can create a session with key "SME-ROCKS," and Component 2 can use that same session). If a key is provided, the session that is created must be disposed of by calling dispose() as was done in the example above. A session should not be kept without being disposed of for more than 5 minutes.

ts

```
  const session =
this.appContextService.powerShell.createSession('{!TargetNode}', '{!Key}');
```

Keyless: A key will automatically be created for the session. This session with be disposed of automatically after 3 minutes. Using keyless allows your extension to recycle the use of any runspace that is already available at the time of creation of a session. If no runspace is available then a new one will be created. This functionality is good for one-off calls, but repeated use can affect performance. A session takes approximately 1 second to create, so continuously recycling sessions can cause slowdowns.

ts

```
  const session =
this.appContextService.powerShell.createSession('{!TargetNodeName}');
```

or

ts

```
const session =
this.appContextService.powerShell.createAutomaticSession('{!TargetNodeName}'
);
```

In most situations, create a keyed session in the `ngOnInit()` method, and then dispose of it in `ngOnDestroy()`. Follow this pattern when there are multiple PowerShell scripts in a component but the underlying session IS NOT shared across components. For best results, make sure session creation is managed inside of components rather than services - this helps ensure that lifetime and cleanup can be managed properly.

For best results, make sure session creation is managed inside of components rather than services - this helps ensure that lifetime and cleanup can be managed properly.

PowerShell Stream

If you have a long running script and data is outputted progressively, a PowerShell stream will allow you to process the data without having to wait for the script to finish. The observable next() will be called as soon as data is received.

ts

```
this.appContextService.powerShellStream.run(session, script);
```

Long Running Scripts

If you have a long running script that you would like to run in the background, a work item can be submitted. The state of the script will be tracked by the Gateway and updates to the status can be sent to a notification.

ts

```
const workItem: WorkItemSubmitRequest = {
    typeId: 'Long Running Script',
    objectName: 'My long running service',
    powerShellScript: script,

    //in progress notifications
    inProgressTitle: 'Executing long running request',
    startedMessage: 'The long running request has been started',
    progressMessage: 'Working on long running script - {{ percent }} %',

    //success notification
    successTitle: 'Successfully executed a long running script!',
    successMessage: '{{objectName}} was successful',
```

```
    successLinkText: 'Bing',
    successLink: 'http://www.bing.com',
    successLinkType: NotificationLinkType.Absolute,

    //error notification
    errorTitle: 'Failed to execute long running script',
    errorMessage: 'Error: {{ message }}'

    nodeRequestOptions: {
       logAudit: true,
       logTelemetry: true
    }
};

return this.appContextService.workItem.submit('{!TargetNode}', workItem);
```

> **Note**
>
> For progress to be shown, Write-Progress must be included in the script that you have written. For example:

```ps1
 Write-Progress -Activity ‘The script is almost done!' -percentComplete
95
```

WorkItem Options

Expand table

function	Explanation
submit()	Submits the work item
submitAndWait()	Submit the work item and wait for the completion of its execution
wait()	Wait for existing work item to complete
query()	Query for an existing work item by ID
find()	Find and existing work item by the TargetNodeName, ModuleName, or typeId.

PowerShell Batch APIs

If you need to run the same script on multiple nodes, then a batch PowerShell session can be used. For example:

```ts
const batchSession = this.appContextService.powerShell.createBatchSession(
    ['{!TargetNode1}', '{!TargetNode2}', sessionKey);
  this.appContextService.powerShell.runBatchSingleCommand(batchSession,
command).subscribe((responses: PowerShellBatchResponseItem[]) => {
    for (const response of responses) {
      if (response.error || response.errors) {
        //handle error
      } else {
        const results = response.properties && response.properties.results;
        //response.nodeName
        //results[0]
      }
    }
     },
     Error => { /* handle error */ });
```

PowerShellBatch options

Expand table

option	Explanation
runSingleCommand	Run a single command against all the nodes in the array
run	Run corresponding command on paired node
cancel	Cancel the command on all nodes in the array

Feedback

Was this page helpful? Yes No

Provide product feedback

Current list of icons in Windows Admin Center

Article • 03/24/2025 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

```
.icon-win-globalNavButton:before { content: "\E700"; }
.icon-win-connect:before { content: "\E703"; }
.icon-win-chevronDown:before { content: "\E70D"; }
.icon-win-chevronUp:before { content: "\E70E"; }
.icon-win-edit:before { content: "\E70F"; }
.icon-win-add:before { content: "\E710"; }
.icon-win-cancel:before { content: "\E711"; }
.icon-win-more:before { content: "\E712"; }
.icon-win-settings:before { content: "\E713"; }
.icon-win-mail:before { content: "\E715"; }
.icon-win-pin:before { content: "\E718"; }
.icon-win-shop:before { content: "\E719"; }
.icon-win-stop:before { content: "\E71A"; }
.icon-win-link:before { content: "\E71B"; }
.icon-win-filter:before { content: "\E71C"; }
.icon-win-allApps:before { content: "\E71D"; }
.icon-win-search:before { content: "\E721"; }
.icon-win-attach:before { content: "\E723"; }
.icon-win-forward:before { content: "\E72A"; }
.icon-win-back:before { content: "\E72B"; }
.icon-win-refresh:before { content: "\E72C"; }
.icon-win-share:before { content: "\E72D"; }
.icon-win-lock:before { content: "\E72E"; }
.icon-win-blocked:before { content: "\E733"; }
.icon-win-favoriteStar:before { content: "\E734"; }
.icon-win-favoriteStarFill:before { content: "\E735"; }
.icon-win-remove:before { content: "\E738"; }
.icon-win-backToWindow:before { content: "\E73F"; }
.icon-win-fullScreen:before { content: "\E740"; }
.icon-win-up:before { content: "\E74A"; }
.icon-win-down:before { content: "\E74B"; }
.icon-win-oEM:before { content: "\E74C"; }
.icon-win-delete:before { content: "\E74D"; }
.icon-win-save:before { content: "\E74E"; }
.icon-win-eraseTool:before { content: "\E75C"; }
.icon-win-play:before { content: "\E768"; }
.icon-win-pause:before { content: "\E769"; }
.icon-win-chevronLeft:before { content: "\E76B"; }
.icon-win-chevronRight:before { content: "\E76C"; }
.icon-win-updateRestore:before { content: "\E777"; }
.icon-win-unpin:before { content: "\E77A"; }
.icon-win-contact:before { content: "\E77B"; }
.icon-win-paste:before { content: "\E77F"; }
.icon-win-lEDLight:before { content: "\E781"; }
.icon-win-error:before { content: "\E783"; }
```

```
.icon-win-unlock:before { content: "\E785"; }
.icon-win-newWindow:before { content: "\E78B"; }
.icon-win-saveLocal:before { content: "\E78C"; }
.icon-win-redo:before { content: "\E7A6"; }
.icon-win-undo:before { content: "\E7A7"; }
.icon-win-warning:before { content: "\E7BA"; }
.icon-win-flag:before { content: "\E7C1"; }
.icon-win-powerButton:before { content: "\E7E8"; }
.icon-win-home:before { content: "\E80F"; }
.icon-win-history:before { content: "\E81C"; }
.icon-win-recent:before { content: "\E823"; }
.icon-win-chat:before { content: "\E901"; }
.icon-win-clear:before { content: "\E894"; }
.icon-win-sync:before { content: "\E895"; }
.icon-win-download:before { content: "\E896"; }
.icon-win-help:before { content: "\E897"; }
.icon-win-upload:before { content: "\E898"; }
.icon-win-openInNewWindow:before { content: "\E8A7"; }
.icon-win-switch:before { content: "\E8AB"; }
.icon-win-remote:before { content: "\E8AF"; }
.icon-win-folder:before { content: "\E8B7"; }
.icon-win-copy:before { content: "\E8C8"; }
.icon-win-sort:before { content: "\E8CB"; }
.icon-win-permissions:before { content: "\E8D7"; }
.icon-win-unfavorite:before { content: "\E8D9"; }
.icon-win-openFile:before { content: "\E8E5"; }
.icon-win-newFolder:before { content: "\E8F4"; }
.icon-win-bulletedList:before { content: "\E8FD"; }
.icon-win-manage:before { content: "\E912"; }
.icon-win-accept:before { content: "\E8FB"; }
.icon-win-completed:before { content: "\E930"; }
.icon-win-code:before { content: "\E943"; }
.icon-win-info:before { content: "\E946"; }
.icon-win-chevronUpSmall:before { content: "\E96D"; }
.icon-win-chevronDownSmall:before { content: "\E96E"; }
.icon-win-chevronLeftSmall:before { content: "\E96F"; }
.icon-win-chevronRightSmall:before { content: "\E970"; }
.icon-win-chevronUpMed:before { content: "\E971"; }
.icon-win-chevronDownMed:before { content: "\E972"; }
.icon-win-chevronLeftMed:before { content: "\E973"; }
.icon-win-chevronRightMed:before { content: "\E974"; }
.icon-win-pC1:before { content: "\E977"; }
.icon-win-unknown:before { content: "\E9CE"; }
.icon-win-ringer:before { content: "\EA8F"; }
.icon-win-checkList:before { content: "\E9D5"; }
.icon-win-processing:before { content: "\E9F5"; }
.icon-win-hourGlass:before { content: "\EA03"; }
.icon-win-asterisk:before { content: "\EA38"; }
.icon-win-errorBadge:before { content: "\EA39"; }
.icon-win-allAppsMirrored:before { content: "\EA40"; }
.icon-win-bulletedListMirrored:before { content: "\EA42"; }
.icon-win-helpMirrored:before { content: "\EA51"; }
.icon-win-dietPlanNotebook:before { content: "\EAC8"; }
.icon-win-market:before { content: "\EAFC"; }
.icon-win-heart:before { content: "\EB51"; }
```

```
.icon-win-editMirrored:before { content: "\EB7E"; }
.icon-win-speedHigh:before { content: "\EC4A"; }
.icon-win-fileExplorer:before { content: "\EC50"; }
.icon-win-developerTools:before { content: "\EC7A"; }
.icon-win-embed:before { content: "\ECCE"; }
.icon-win-publish:before { content: "\ECDB"; }
.icon-win-networkPipes:before { content: "\ECE3"; }
.icon-win-blocked2:before { content: "\ECE4"; }
.icon-win-toolbox:before { content: "\ECED"; }
.icon-win-gateway:before { content: "\ED23"; }
.icon-win-openFolderHorizontal:before { content: "\ED25"; }
.icon-win-playPause:before { content: "\ED38"; }
.icon-win-hardDrive:before { content: "\EDA2"; }
.icon-win-virtualMachine:before { content: "\EE9B"; }
.icon-win-rAM:before { content: "\EEA0"; }
.icon-win-cPU:before { content: "\EEA1"; }
.icon-win-hostCluster:before { content: "\EEA2"; }
.icon-win-virtualMachineGroup:before { content: "\EEA3"; }
.icon-win-customList:before { content: "\EEBE"; }
.icon-win-customListMirrored:before { content: "\EEBF"; }
.icon-win-marketDown:before { content: "\EF42"; }
.icon-win-database:before { content: "\EFC7"; }
.icon-win-checklistMirrored:before { content: "\F0B5"; }
.icon-win-windowsUpdate:before { content: "\F0C5"; }
.icon-win-backMirrored:before { content: "\F0D2"; }
.icon-win-forwardMirrored:before { content: "\F0D3"; }
.icon-win-statusCircleOuter:before { content: "\F136"; }
.icon-win-statusCircleInner:before { content: "\F137"; }
.icon-win-statusCircleRing:before { content: "\F138"; }
.icon-win-statusTriangleOuter:before { content: "\F139"; }
.icon-win-statusTriangleInner:before { content: "\F13A"; }
.icon-win-statusTriangleExclamation:before { content: "\F13B"; }
.icon-win-statusCircleExclamation:before { content: "\F13C"; }
.icon-win-statusCircleErrorX:before { content: "\F13D"; }
.icon-win-statusCircleCheckmark:before { content: "\F13E"; }
.icon-win-statusCircleInfo:before { content: "\F13F"; }
.icon-win-statusCircleBlock:before { content: "\F140"; }
.icon-win-statusCircleBlock2:before { content: "\F141"; }
.icon-win-statusCircleQuestionMark:before { content: "\F142"; }
.icon-win-statusCircleSync:before { content: "\F143"; }
.icon-win-exploreContentSingle:before { content: "\F164"; }
.icon-win-collapseContentSingle:before { content: "\F166"; }
.icon-win-hardDriveGroup:before { content: "\F18F"; }
.icon-win-tripleColumn:before { content: "\F1D5"; }
.icon-win-certificateManager:before { content: "\F1F8"; }
.icon-win-firewall:before { content: "\F1F9"; }
.icon-win-firewallRules:before { content: "\F1FA"; }
.icon-win-localAdmin:before { content: "\F1FB"; }
.icon-win-networkSettings:before { content: "\F1FC"; }
.icon-win-powerShell:before { content: "\F1FD"; }
.icon-win-serverProcesses:before { content: "\F1FE"; }
.icon-win-registrayEditor:before { content: "\F1FF"; }
.icon-win-serverFeatures:before { content: "\F200"; }
.icon-win-server:before { content: "\F201"; }
.icon-win-cluster:before { content: "\F202"; }
```

```
.icon-win-saveAll:before { content: "\F203"; }
.icon-win-failoverClusterManager:before { content: "\F204"; }
.icon-win-softwareDefinedDataCenter:before { content: "\F205"; }
.icon-win-storageReplica:before { content: "\F206"; }
.icon-win-storageLogical:before { content: "\F20E"; }
.icon-win-storagePhysical:before { content: "\F20F"; }
.icon-win-networkPhysical:before { content: "\F211"; }
.icon-win-lEDLightOff:before { content: "\F388"; }
.icon-win-removeHardDisk:before { content: "\F389"; }
.icon-win-virtualHardDisk:before { content: "\F38A"; }
.icon-win-virtualSwitch:before { content: "\F38B"; }
.icon-win-virtualNIC:before { content: "\F38C"; }
.icon-win-offlineResource:before { content: "\F38D"; }
.icon-win-onlineResource:before { content: "\F38E"; }
.icon-win-formatDatabase:before { content: "\F3BE"; }
.icon-win-developerToolsRemove:before { content: "\F3EA"; }
.icon-win-unzipFolder:before { content: "\F3FD"; }
.icon-win-storageReplica:before { content: "\F42D"; }
.icon-win-speedHighOff:before { content: "\F42E"; }
.icon-win-bootOrder:before { content: "\F42F"; }
.icon-win-bootOrderMirrored:before { content: "\F430"; }
```

Feedback

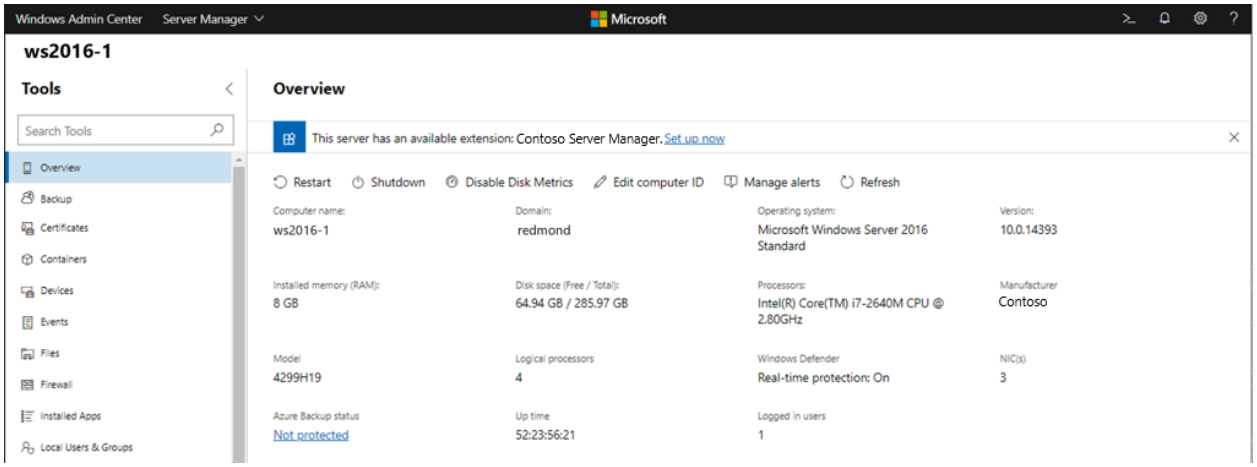
Was this page helpful? Yes No

Provide product feedback

Enabling the extension discovery banner

Article • 11/01/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

The extension discovery banner feature was introduced in the Windows Admin Center Preview 1903 release. This feature allows an extension to declare the server hardware manufacturer and models it supports, and when a user connects to a server or cluster for which an extension is available, a notification banner will be displayed to easily install the extension. Extension developers will be able to get more visibility for their extensions and users will be able to easily discover more management capabilities for their servers.



How the extension discovery banner works

When Windows Admin Center is launched, it will connect to the registered extension feeds and fetch the metadata for the available extension packages. Then when a user connects to a server or cluster in Windows Admin Center, we read the server hardware manufacturer and model to display in the Overview tool. If we find an extension that declares that it supports the current server's manufacturer and/or model, we'll display a banner to let the user know. Clicking on the "Set up now" link will take the user to Extension Manager where they can install the extension.

How to implement the extension discovery banner

The "tags" metadata in the .nuspec file is used to declare which hardware manufacturer and/or models your extension supports. Tags are delimited by spaces and you can add either a manufacturer or model tag, or both to declare the supported manufacturer and/or models. The tag format is `"[value type]_[value condition]"` where [value type]

is either "Manufacturer" or "Model" (case sensitive), and [value condition] is a Javascript regular expression defining the manufacturer or model string, and [value type] and [value condition] are separated by an underscore. This string is then encoded using URI encoding and added to the .nuspec "tags" metadata string.

Example

Let's say I've developed an extension that supports servers from a company named Contoso Inc., with model name R3xx and R4xx.

1. The tag for the manufacturer would be `"Manufacturer_/Contoso Inc./"`. The tag for the models could be `"Model_/^R[34][0-9]{2}$/"`. Depending on how strictly you want to define the matching condition, there will be different ways to define your regular expression. You can also separate the Manufacturer or Model tags into multiple tags, for example, the Model tag could also be `"Model_/R3../ Model_/R4../"`.

2. You can test the regular expression with your web browser's DevTools Console. In Edge or Chrome, hit F12 to open the DevTools window, and in the Console tab, type the following and hit Enter:

   ```
   JavaScript
   var regex = /^R[34][0-9]{2}$/
   ```

   Then if you type and run the following, it will return 'true'.

   ```
   JavaScript
   regex.test('R300')
   ```

   And if you run the following, it will return 'false'.

   ```
   JavaScript
   regex.test('R500')
   ```

3. Once you've verified the regular expression, you can encode it in the DevTools Console as well, using the following Javascript method:

   ```
   JavaScript
   ```

```
encodeURI(/^R[34][0-9]{2}$/)
```

The final format of the tag string to add to your .nuspec file would be:

```
<tags>Manufacturer_/Contoso%20Inc./ Model_/%5ER%5B34%5D%5B0-
9%5D%7B2%7D$/</tags>
```

> **Tip**
>
> We understand that a hardware manufacturer may have a very wide range of model names of which some may be supported while some are not. Keep in mind that this feature is meant to help with the **discovery** of your extension, but it does not have to be a perfectly up-to-date inventory of all your models. You can define your regular expression to be a simpler expression that matches a subset of your models. A user might not see the discovery banner if they first connect to a server model that doesn't match the condition, but sooner or later they will connect to another server that does and will discover and install the extension. You can also consider defining a simple regular expression that only matches your manufacturer name. In some cases, your extension may not actually support a specific model, but you can use the dynamic tool display feature to define a custom PowerShell script to check model support and only show your extension when applicable, or provide limited functionality in your extension for models that don't support all capabilities.

Feedback

Was this page helpful? Yes No

Provide product feedback

Upgrade existing Windows Admin Center extensions to Angular 11

Article • 04/28/2023 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Windows Admin Center is upgrading to Angular 11.0! This upgrade brings in the latest in features, security, and performance, and we're excited to have it available for you. So far, the shell of Windows Admin Center has been upgraded and it is your turn to update your extensions. Follow the steps in this document to get your extension updated.

If you run into any issues during this process, reach out to your Microsoft contact and they'll assist you in routing the request.

Preliminary steps

Before beginning the upgrade to Angular 11, you need to configure your developer environment with the latest Windows Admin Center shell and development tools. Complete the following steps before proceeding to the upgrade process:

1. Install the latest version of Windows Admin Center in dev mode (`msiexec /i WindowsAdminCenter<version>.msi DEV_MODE=1`) with the upgraded shell. Reach out to your Microsoft contact if this has not been provided to you.
2. **(Recommended)** Create a `features/ng11` branch in the repo.
3. **(Recommended)** Update `version.json` version to `(N+1).0.0`.
4. In a PowerShell console, make sure to switch to the respective Node version for your Angular version (for more information, see Working with two branches of shell for more details). For Angular 11, run `nvm use 12.18.3`. Close the terminal for this change to take effect.
5. Clean up the `node_modules` folder to avoid npm conflicts.

Automated upgrade process

Download and install the Windows Admin Center CLI tools by running `npm install -g @microsoft/windows-admin-center-sdk@experimental` if you have not already done so before proceeding through the following steps.

1. At the root level of the repo, run `wac upgrade --audit=false --experimental`.

    - If working on an extension repository that is consumed by other extensions, include the `--library` flag as well.

If the library flag was used, edit the `name` property in `src/package.json` to something unique to the extension.

2. **(Conditional)** If the extension repo has dependencies on any other extension package, you will have to manually pick the new angular version for that extension (e.g. `msft-sme-certificate-manager` has a dependency on `msft-sme-event-viewer`. The automated tools will **not** update `msft-sme-event-viewer` version, it has to be manually updated.) Also be sure to specify the '/dist' folder level on any imports from extensions, any lower or higher-level imports won't work (e.g. `import { foobar } from '@msft-sme/event-viewer'` would need to be changed to `import { foobar } from '@msft-sme/event-viewer/dist'`.)

3. Open `app-routing.module.ts` and change any appRoutes that have the format `./folder-name/file-name#ModuleClass` to `() => import('./folder-name/file-name').then(m => m.ModuleClass)`. If there are any other `routing.module.ts` files, they will also need to be updated in this way.

4. Remove `UpgradeAudit.txt` file. It's autogenerated for your reference but doesn't need to go in the repo.

5. Go through the following files and replace all instances of `@msft-sme` with `@microsoft/windows-admin-center-sdk`:

- `./angular.json`
- `./gulpfile.ts/common/e2e.ts`
- `./gulpfile.ts/common/resjson.ts`
- `./src/polyfills.ts`
- `./src/test.ts`

6. There will likely be unresolved errors as a result of the steps you've completed. Proceed to Build steps.

Build steps

At this point in the upgrade process, your extension repo is ready to be built and the debugging process can begin. Proceed through the following steps:

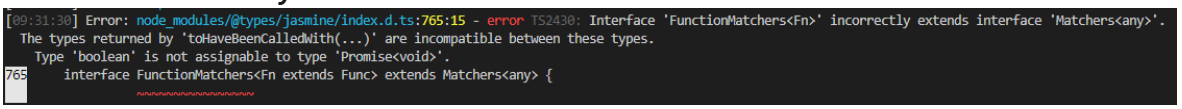
1. Run `gulp build`.
2. Watch out for any linting and compilation errors.
3. Fix these errors and repeat steps 1-3 as necessary.
4. When all build errors are fixed, commit your changes and proceed to Run steps.

Difficult to diagnose build errors

Some of the errors you may receive while debugging in the build step may be hard to diagnose. Here are two of the most common difficult to diagnose errors and how to mitigate them:

- **NG6002: Appears in the NgModule.imports of AppModule, but could not be resolved to an NgModule class**
  - This type of error occurs at build time, typically before the upgraded repository has been successfully built at least once. To resolve, run `ng serve --prod`, after which these errors should no longer appear when building.

- **Interface incorrectly extends another interface**



  - This error occurs during the inlineCompile step of "gulp build" and occurs as the result of a mismatch in versions between the `@types/jasmine` package downloaded and what the `@types/jasminewd2` package requires. This error can be resolved by removing the `@types/jasminewd2` package.

Output bundle file names

When building your extension, you may run into issues as a result of the file names in your bundle. To avoid these issues, pay special attention to the following fields:

- **Output hashing must be enabled**. When output hashing is enabled, unique file names will be generated for every build of the extension. If this is not enabled, you may be unable to see the changes to your extension when viewing in the browser due to duplicate file names.
  - To enable from this field the command line, add the `--output-hashing` flag to an `ng build` command.
  - To enable this field from your repo directly, navigate to your angular.json file and look for the `outputHashing` field under production configurations.
- **Named chunks must be disabled**. When named chunks are enabled, each bundle file includes its original module file name. While that may seem useful, it often results in incredibly long file names that can result in errors in the Windows Admin Center extension feed.
  - To disable this field from the command line, add the `--named-chunks` flag to an `ng build` command.
  - To disable this field from your repo directly, navigate to your angular.json file and look for the `namedChunks` field under production configurations. Set this

field to false.

Run steps

Now that you've fixed all of the build errors in your extension, you're ready to run your extension and fix any runtime issues. Follow the steps below to run your extension:

1. Sideload the extension with `gulp serve --port <port> --prod --aot`.
2. In the browser, look for any runtime issues with the extension, such as:
    - Extension page(s) not loading
    - Elements missing from the extension page(s)
    - Console errors
    - Anything else that looks off or behaves strange
3. Fix any runtime issues that you have discovered.
4. When the extension has been stabilized, commit your changes.

When you have finished these steps, proceed to Creating a main branch.

Creating a main branch

After all linting, compilation, and runtime errors have been fixed, you're ready to finish upgrading your extension. To do this, we need to create a new branch in the extension repository. Follow these steps to finish upgrading your extension:

1. Ensure that you are ready to complete the upgrade process and everything is working as expected in the feature branch.
2. Create a new branch named "main" in the repository.
3. Create a PR from the features/ng11 branch that merges into main.
4. When ready, complete the PR.
5. Congratulations, you successfully upgraded an extension!

Releasing your upgraded extension

Once your extension has been tested in Windows Admin Center desktop and service mode, send an email to wacextensionrequest@microsoft.com to coordinate the release of your upgraded extension.

Working with two branches of shell

Upgrading the Windows Admin Center shell has resulted in numerous environmental changes. One such change is the use of Node 12.18.3 from the previous 10.22.0 version. These versions are incompatible and you must toggle your global version to run build commands in each environment.

To manage your versions of Node, we suggest using Node Version Manager: https://github.com/coreybutler/nvm-windows

Follow the instructions to install nvm-windows on your machine.

Once installed, you can prepare your environment by running these commands:

```
nvm install 12.18.3
nvm use 12.18.3
npm i -g gulp-cli
npm i -g @angular/cli
npm i -g vsts-npm-auth
npm i -g typescript

nvm install 10.22.0
nvm use 10.22.0
npm i -g gulp-cli
npm i -g @angular/cli
npm i -g vsts-npm-auth
npm i -g typescript
```

This will set up your Node environment for development with both the new and old versions of Angular.

Toggling Node version

The version of Node you are using can be toggled using PowerShell.

The `nvm list` command can be used to list installed node versions.

The `nvm use <version>` command can be used to quickly switch between node versions.

You can find a full index list of which Node, Angular, and Typescript versions go together here: Node - Angular compatibility index .

Note

All version numbers in this document are specific to the Windows Admin Center upgrade from Angular 7 to Angular 11.

Following the process above, you will lose all global node settings including your VSTS authentication.

To restore VSTS authentication, run this command at the root of any repo: `vsts-npm-auth -config .npmrc`

Other considerations when upgrading extensions to Angular 11

- Sideloading of the shell and extensions should not be affected when working with two branches of shell.
- When using copyTarget, be aware of which shell branch you are in. Only use this command in the 2.0 branch if the extension you are working with is also upgraded to Angular 11.
- If the repo has been upgraded to Angular 11, then use the latest 2.x.0 version of shell libraries. Otherwise continue to use the latest 1.x.0 version.

You can tell if a repo is upgraded by looking at the package.json file.

Feedback

Was this page helpful? Yes No

Provide product feedback

Upgrade existing Windows Admin Center extensions to Angular 15

Article • 12/08/2023

Windows Admin Center has upgraded to Angular 15! This upgrade brings in the latest in features, security, and performance, and we're excited to have it available for you. So far, the shell of Windows Admin Center has been upgraded, and it's now your turn to update your extensions.

We strongly recommend upgrading your extensions to the latest Angular version we support. All the new fixes and updates to our shell and SDK are only available for extensions on Angular 15.

Follow the steps in this document to get your extension updated. If you run into any issues during this process, reach out to your Microsoft contact and they'll assist you in routing the request.

Preliminary steps

Before beginning the upgrade to Angular 15, it's essential to ensure that your project is using ESLint and that your environment is set up properly.

> ⓘ **Note**
>
> If your extension is still running Angular 7, **upgrade to Angular 11** before reading the rest of this guide.

Transitioning from TSLint to ESLint

Historically, Windows Admin Center has used the extensible tool TSLint to check TypeScript code for readability, maintainability, and functionality errors. TSLint has been deprecated and replaced by ESLint, a more powerful and widely supported tool. Windows Admin Center has shifted to using ESLint.

To determine whether you're using TSLint or ESLint as the linter for your extension, navigate to the root directory of your extension project:

- If there's a file titled **.eslintrc.json**, you're using ESLint. Skip ahead to **Configuring your environment for upgrade**.

- If there's a file titled **tslint.json**, you're using TSLint. Continue on to learn how to migrate to ESLint.

We've created a command in the SDK to help with automating the transition to ESLint. To use it, follow these steps:

1. Navigate to the root of your extension project.
2. Pull down the latest version of the SDK by running `npm install -g @microsoft/windows-admin-center-sdk@latest`
3. Run `npm install`.
4. Run `wac eslint`.
5. Fix any violations by running `npx lint --fix` or by making manual changes.

Configuring your environment for upgrade

Before beginning the upgrade to Angular 15, you need to configure your developer environment with the latest Windows Admin Center shell and development tools, including upgrading your version of Node.

To manage your versions of Node, we suggest using Node Version Manager. Follow the instructions to install nvm-windows on your machine. The version should be 1.1.11 or later, as older versions may not support the Node.js versions necessary for this upgrade.

Once installed, you can prepare your environment by running these commands:

```
nvm install 16.14.0
nvm use 16.14.0
npm i -g gulp-cli
npm i -g @angular/cli@15.2.9
npm i -g vsts-npm-auth
npm i -g typescript@4.8.2
```

> **Note**
>
> If you want to continue to make changes in Angular 11, you'll want to use Node 12.18.3. These versions are incompatible for use at the same time, so you must toggle your global version to run build commands in each environment.
>
> To change your node version back to the Angular 11 configuration, you can run the following commands:

```
nvm install 12.18.3
nvm use 12.18.3
npm i -g gulp-cli
npm i -g @angular/cli@11.2.13
npm i -g vsts-npm-auth
npm i -g typescript@4.1.5
```

When toggling node versions, you may lose all global node settings, including your VSTS authentication.

To restore VSTS authentication, run `vsts-npm-auth -config .npmrc` at the root of your repository.

Automated upgrade process

To facilitate a smoother upgrade from Angular 11 to Angular 15, we've developed a CLI command as a part of our SDK that automates much of the upgrade process.

Before proceeding with the automatic upgrade steps, make sure your environment is set up correctly, and you've switched to Angular 15. When you're ready to upgrade, follow these steps:

1. Make sure you have the latest version of the WAC CLI by running `npm install -g @microsoft/windows-admin-center-sdk@latest`.
2. Run `wac angular15Upgrade`.

Post upgrade actions

After you run the upgrade command, a `log.txt` file will be generated at the root of your project. This file contains information on further actions.

After running the upgrade command, be sure to stage your changes. This command is designed to be run multiple times if necessary.

You should now be able to run and test your extension as normal.

Troubleshooting scenarios

Some of the errors you may receive while debugging in the build step may be hard to diagnose. Here's some of the most common errors and how to mitigate them:

- **AjaxResponse requires 1 type argument**
  - To fix this issue, try staging your current changes and then rerun the upgrade command. The upgrade command should've fixed this, but there might be edge cases. Ensure you manually fix these errors by changing `AjaxResponse` to `AjaxResponse<any>`.

- **CSS syntax error in custom CSS files**

Windows Command Prompt

```
File path: C:/Branches/msft-sme-containers/src/app/containers-
inventory/containers-inventory.component.css
        Type: css-syntax-error
        Line: undefined
        Char: undefined
        Message: Warning: ▲ [WARNING] Unexpected ">" [css-syntax-error]
    C:/Branches/msft-sme-containers/src/app/containers-
inventory/containers-inventory.component.css:1:7:
    1 │ :host >>> .summary-container {
                ^
```

  - To fix this issue, try staging your current changes and then rerun the upgrade command. The upgrade command should automatically handle this, but if it doesn't

- **Dependency warnings**

```
Build at: 2023-11-14T19:31:35.361Z - Hash: 069ae6fa11d268d6 - Time:
32165ms
[11:31:35] Warning: C:\Branches\msft-sme-containers\node_modules\@msft-
sme\core\data\crypto.js depends on 'base64-arraybuffer'. CommonJS or
AMD dependencies can cause optimization bailouts.
For more info see: https://angular.io/guide/build#configuring-commonjs-
dependencies

Warning: C:\Branches\msft-sme-containers\node_modules\@msft-sme\event-
viewer\dist\fesm2020\msft-sme-event-viewer-lib.mjs depends on 'file-
saver'. CommonJS or AMD dependencies can cause optimization bailouts.
For more info see: https://angular.io/guide/build#configuring-commonjs-
dependencies

Warning: C:\Branches\msft-sme-containers\node_modules\@msft-
sme\powershell-console\__ivy_ngcc__\dist\fesm2015\msft-sme-powershell-
console-lib.js depends on 'xterm'. CommonJS or AMD dependencies can
cause optimization bailouts.
For more info see: https://angular.io/guide/build#configuring-commonjs-
dependencies

Warning: C:\Branches\msft-sme-containers\node_modules\@msft-
```

```
sme\powershell-console\__ivy_ngcc__\dist\fesm2015\msft-sme-powershell-
console-lib.js depends on 'xterm-addon-fit'. CommonJS or AMD
dependencies can cause optimization bailouts.
For more info see: https://angular.io/guide/build#configuring-commonjs-
dependencies
```

  - These issues must be fixed manually. Navigate to `angular.json`, look for `"allowedCommonJsDependencies"`, and add all dependency items into the array. In this case, it would be: `"allowedCommonJsDependencies": ["base64-arraybuffer", "file-saver", "xterm", "xterm-addon-fit"]`

Other considerations when upgrading extensions to Angular 15

- Sideloading of the shell and extensions shouldn't be affected when working with two branches of shell.
- When using copyTarget, be aware of which shell branch you're in. Only use this command in the 4.0 branch if the extension you're working with is also upgraded to Angular 15.
- If the repo has been upgraded to Angular 15, then use the latest 4.x.0 version of shell libraries. Otherwise continue to use the latest 2.x.0 version.

You can tell if a repo is upgraded by looking at the package.json file.

Releasing your upgraded extension

Once your extension has been tested in Windows Admin Center desktop and service mode, send an email to wacextensionrequest@microsoft.com to coordinate the release of your upgraded extension.

Extension support for the management of Windows Defender Application Control (WDAC) enforced infrastructure

Article • 06/28/2023 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Windows Admin Center supports the management of Windows Defender Application Control (WDAC) enforced infrastructure at the platform level. Learn more about managing WDAC enforced infrastructure in Windows Admin Center.

Support for this management at the platform level doesn't mean extensions built for Windows Admin Center also support the management of WDAC enforced infrastructure by default. This guide outlines the requirements for an extension to support the management of WDAC enforced infrastructure.

Extension structure requirements

To manage WDAC enforced infrastructure, Windows Admin Center must ingest and run PowerShell scripts in a particular fashion to adhere to best security practices. To ensure your extension's scripts are run correctly, ensure your extension conforms to the following requirements.

All PowerShell scripts must be stored in a file

Historically, developers of WAC extensions may have chosen to include custom PowerShell code as a string in their extension manifest.json file. For example, one may choose to define the conditions for a tool extension's visibility by providing a PowerShell script in the "script" property. For PowerShell scripts to be compatible with WDAC, they must be signed. Strings can't be signed.

To ensure this requirement is met, follow these steps:

1. Identify any PowerShell scripts in your manifest.json file.
2. After defining any script content in your manifest.json file, remove the script content and store it in a .ps1 file in the `resources/scripts` directory of your extension. Script code in the extension manifest now follows the same rules as other Windows Admin Center PowerShell.
3. Update the conditions property in the extension manifest to the following format:

```
JSON
```

```
"conditions": [
    {
        "powerShell": {
            "command": "Script-File-Name",
            "module": "powerShellModuleName",
            "script": "Your script text goes here."
        }
    }
]
```

The PowerShell module name already exists in your extension manifest. **Its value in the manifest and in the PowerShell field must match.**

4. Identify any other places where PowerShell scripts are being created dynamically. Creating a PowerShell script dynamically using string concatenation can allow an attacker to inject arbitrary PowerShell script to be executed. This method can be used to bypass limitations enforced on a remote user that is using a restricted run space. It can also be used to achieve standard command injection against any application that builds PowerShell scripts with user input and executes it.

Example of script block created with string concatenation:

PowerShell

```
param($UserInputVar)
$DynamicScript = "Get-ChildItem $UserInputVar"
$ScriptBlock = [ScriptBlock]::Create($DynamicScript)
Invoke-Command $ScriptBlock
```

Example of this same script block constructed without string concatenation:

PowerShell

```
param($UserInputVar)
 [ScriptBlock]$ScriptBlock = {
Param($SafeUserInput)
Get-ChildItem $ SafeUserInput
 }
 Invoke-Command -ScriptBlock $ScriptBlock -ArgumentList @($UserInputVar)

# OR, alternatively
param($UserInputVar)
 Invoke-Command -ScriptBlock {
 param(
    [String] $SafeUserInput
 )
Get-ChildItem $SafeUserInput
```

```
} -ArgumentList $UserInputVar
```

Script files should also not be constructed using string concatenation. Here's an example of how not to construct script files:

PowerShell

```
$Script=@'
    Get-ChildItem $UserInputVar
'@
$Script = '$ UserInputVar =' + "'$ UserInputVar;"+$Script
$path = “C:\temp”
$Script | Out-File $path
```

Construct your script files like this instead:

PowerShell

```
Function test {
    param(
        [String] $userInputVar
     )
    Get-ChildItem $UserInputVar
    }

    $path = “C:\temp”
    (Get-Command test).ScriptBlock | Set-Content -path $path
```

All PowerShell code must be signed and stored in the proper location

As part of the changes Windows Admin Center made to support the management of WDAC enforced infrastructure, signed PowerShell scripts for an extension are now transferred to the node Windows Admin Center is currently connected to before being run. Additionally, as mentioned in the previous requirement, WDAC enforced infrastructure only runs signed PowerShell scripts. Because of these requirements, all your PowerShell code must be signed. All your PowerShell must also be located in a consistent location so that the Windows Admin Center platform can predictably locate an extension’s signed modules.

If your extension repository doesn't contain a `powershell-module` directory containing signed PowerShell module(s), the Windows Admin Center platform will be unable to identify transferable code, and operations will fail in a WDAC-enforced environment.

The Windows Admin Center `gulp build` command updates the `/dist` folder inside your repository, generating your unsigned .psd1 and .psm1 files inside a module folder. These files need to be signed with a signing certificate that matches one that is allow-listed in the WDAC policy.

To make this change, it's highly recommended to create a build pipeline that incorporates PowerShell signing.

You can validate that your PowerShell is in the proper format in one of two ways:

1. When your extension is installed, you can view the `ProgramData\Server Management Experience\UX\modules` directory on your gateway machine (the one on which Windows Admin Center is running). Here you should see the `powershell-module` folder and the signed PowerShell module(s)
2. Extract the contents of your extension's .nupkg artifact. The `powershell-module` folder should be present and contain the signed PowerShell module(s).

In both cases, verifying that the .psd1 and .psm1 files themselves are signed can be done by running the Get-AuthenticodeSignature command on the file, or by right-clicking the file itself and validating the digital signature.

WorkItems that utilize the "powerShellScript" property should be updated to use the "powerShellCommand" property

The Windows Admin Center platform needs to be able to determine which module a PowerShell command belongs to. Because of this requirement, WorkItems that specify a PowerShell command using the `powerShellScript` property cause an error.

To mitigate this behavior, use the `powerShellCommand` property, along with the `createCommand` method, to form a valid command object.

Here's an example of a WorkItem using the old method:

ts

```
    const workItem : WorkItemSubmitRequest = {
      typeId: "SampleWorkItem",
      title: "Title",
      powerShellScript: PowerShellScripts.[scriptName],
      successMessage: "Success",
      errorMessage: "Error",
```

```
      progressMessage: "In progress..."
    }
```

And here's the same WorkItem using the new method:

ts

```
    const workItem : WorkItemSubmitRequest = {
      typeId: "SampleWorkItem",
      title: "Title",
      powerShellCommand: PowerShell.createCommand(PowerShellScripts.
[scriptName]),
      successMessage: "Success",
      errorMessage: "Error",
      progressMessage: "In progress..."
    }
```

Ensuring PowerShell scripts run in Constrained Language mode

Many WDAC policies force all PowerShell scripts to run in Constrained-Language mode. To maintain full functionality throughout Windows Admin Center, you should ensure that all scripts in your extension follow these best practices:

1. If your script files are exported using PowerShell modules, they must explicitly export the functions by name without the use of wildcard characters. This requirement is to prevent inadvertently exposing helper functions that may not be meant to be used publicly.
2. Dot sourcing a script file brings all functions, variables, aliases from that script into the current scope. This functionality blocks a trusted script from being dot sourced into an untrusted script and exposing all its internal functions. Similarly, an untrusted script is prevented from being dot sourced into a trusted script so that it can't pollute the trusted scope.
3. It's recommended to avoid using the Start-Job command to run script blocks unless that script block can already be run successfully in Constrained-Language mode.

Suggested error handling for failure to support WDAC enforced infrastructure management

If you don't plan to support running your extension on WDAC-enforced machines, we suggest adding UI explaining that the management of WDAC enforced infrastructure is

an unsupported scenario in your extension to avoid user confusion. We recommend a layout like our existing Azure hybrid services pages, which features the extension icon and text centered on the extension iFrame.

For the text on this page, we suggest the following wording:

“This extension doesn't currently support running on machines with Windows Defender Application Control (WDAC) enforced.”

This text is only a suggestion. If you’re unsure about the wording you’d like to use, email the Windows Admin Center team at wacextensionrequests@microsoft.com.

Detecting WDAC enforcement from your extension

To follow the guidance in the previous section, you need to determine if the node you're connected to has WDAC enforced. Windows Admin Center exposes a method called `getPsLanguageMode`, defined as part of Windows Admin Center’s WDAC operations, to determine WDAC enforcement.

This method has two outputs:

- Status – HTTPStatusCode type
- psLanguageMode – PsLanguageMode type (enum)

You may consider WDAC to be enforced if PowerShell is running in Constrained Language Mode, which corresponds to a psLanguageMode value of 3.

The following TypeScript sample code gives an example of how to use this method:

ts

```ts
import { Component, OnInit } from '@angular/core';
import { AppContextService } from '@microsoft/windows-admin-center-sdk/angular';
import { WdacOperations } from '@microsoft/windows-admin-center-sdk/core';
import { PSLanguageMode, PsLanguageModeResult } from '@microsoft/windows-admin-center-sdk/core/data/wdac-operations';

@Component({
  selector: 'default-component',
  templateUrl: './default.component.html',
  styleUrls: ['./default.component.css']
})
export class DefaultComponent implements OnInit {
wdacEnforced: boolean;
```

```
  constructor(private appContextService: AppContextService) {
    //
  }

  public ngOnInit(): void {

  }

  public checkWDACEnforced(): void {
    const wdacOperations = new WdacOperations(this.appContextService);

wdacOperations.getPsLanguageMode(this.appContextService.activeConnection.nod
eName).subscribe(
      (response: PsLanguageModeResult) => {
          if (response.psLanguageMode.toString() ===
PSLanguageMode[PSLanguageMode.ConstrainedLanguage]) {
            this.wdacEnforced = true;
          }
          else {
            this.wdacEnforced = false;
          }
      }
    );
  }
}
```

Testing your extension on WDAC enforced infrastructure

Read more about the Windows Defender Application Control policy requirements for Windows Admin Center to get started with testing your extension on WDAC enforced infrastructure.

Feedback

Was this page helpful? Yes No

Provide product feedback

Publishing Extensions

Article • 12/09/2024 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

After you've developed your extension, you'll want to publish it and make it available to others to test or use. In this article, we introduce a few publishing options along with the steps and requirements depending on your audience and purpose of publishing.

Publishing Options

There are three primary options for configurable package sources that Windows Admin Center supports:

- Microsoft public Windows Admin Center NuGet feed
- Your own private NuGet feed
- Local or network file share

Publishing to the Windows Admin Center extension feed

By default, Windows Admin Center is connected to a NuGet feed maintained by the Windows Admin Center product team at Microsoft. Early preview versions of new extensions developed by Microsoft can be published to this feed and made available to Windows Admin Center users. External developers planning to build and release extensions publicly can also submit a request to Publish your extension to the Windows Admin Center feed. Prior to publishing to this feed, external developers have to agree to Windows Admin Center's Extension Publisher Agreement and Extension Participation Policy.

Publishing to a different NuGet feed

You can also create your own NuGet feed to publish your extensions to using one of the many different options for setting up a private source or using a NuGet hosting service. The NuGet feed must support the NuGet v2 API. Because Windows Admin Center doesn't currently support feed authentication, the feed needs to be configured to allow read access to anyone.

Publishing to a file share

To restrict access of your extension to your organization or to a limited group of people, you can use an SMB file share as an extension feed. When using a file share to host your

extension feed, file and folder permissions are applied to manage access to the feed.

Preparing your extension for release

Make sure you read and consider the following development articles:

- Control your tool's visibility
- Strings and localization

Consider releasing as a Preview release

If you're releasing a preview version of your extension for evaluation purposes, we recommend that you:

- Append "(Preview)" to the end of your extension's title in the `.nuspec` file
- Explain the limitations in your extension's description in the `.nuspec` file

Creating an extension package

Windows Admin Center utilizes NuGet packages and feeds for distributing and downloading extensions. For your package to be shipped, you need to generate a NuGet package containing your plugins and extensions. A single package can contain a UI extension and a Gateway plugin. The following section walks you through the process.

Build your extension

As soon as you're ready to start packaging your extension, create a new directory on your file system, open a console, and then CD into it. This directory is the root directory that we use to contain all the nuspec and content directories that make up our package. We call this folder *NuGet Package* for the rest of this article.

UI Extensions

To begin the process on gathering all the content needed for a UI extension, run "gulp build" on your tool and make sure the build is successful. This process packages all the components together in a folder called "bundle" located in the root directory of your extension (at the same level of the src directory). Copy this directory and all its contents into the "NuGet Package" folder.

Gateway Plugins

Using your build infrastructure (which could be as simple as opening Visual Studio and selecting the Build button), compile and build your plugin. Open up your build output directory, copy the DLL or DLLs that represent your plugin and put them in a new folder inside the "NuGet Package" directory called "package". You don't need to copy the FeatureInterface DLL, only the DLL or DLLs that represent your code.

Create the nuspec file

To create the NuGet package, you need to first create a `.nuspec` file. A `.nuspec` file is an XML manifest that contains NuGet package metadata. This manifest is used both to build the package and to provide information to consumers. Place this file at the root of the "NuGet Package" folder.

Here's an example `.nuspec` file and the list of required or recommended properties. For the full schema, see the nuspec reference. Save the `.nuspec` file to your project's root folder using a file name of your choice.

> **ⓘ Important**
>
> The `<id>` value in the `.nuspec` file needs to match the `"name"` value in your project's `manifest.json` file, or else your published extension won't load successfully in Windows Admin Center.

XML

```
<?xml version="1.0">
<package>
  <metadata>
    <id>contoso.project.extension</id>
    <version>1.0.0</version>
    <title>Contoso Hello Extension</title>
    <authors>Contoso</authors>
    <owners>Contoso</owners>
    <requireLicenseAcceptance>false</requireLicenseAcceptance>
    <projectUrl>https://msft-sme.myget.org/feed/windows-admin-center-
feed/package/nuget/contoso.sme.hello-extension</projectUrl>
    <licenseUrl>http://YourLicenseLink</licenseUrl>
    <iconUrl>http://YourLogoLink</iconUrl>
    <description>Hello World extension by Contoso</description>
    <copyright>(c) Contoso. All rights reserved.</copyright>
    <tags></tags>
  </metadata>
  <files>
    <file src="bundle\**\*.*" target="ux" />
```

```
    <file src="package\**\*.*" target="gateway" />
  </files>
</package>
```

Required or Recommended Properties

Expand table

Property Name	Required / Recommended	Description
ID	Required	Unique Package identifier within the feed. This value needs to match the "name" value in your project's manifest.json file. See Choosing a unique package identifier for guidance.
title	Required for publishing to the Windows Admin Center feed	Friendly name for the package that's displayed in Windows Admin Center Extension Manager.
version	Required	Extension version. Using Semantic Versioning (SemVer convention) is recommended but not required.
authors	Required	If publishing on behalf of your company, use your company name.
description	Required	Provide a description of the extension's functionality.
iconUrl	Recommended when publishing to the Windows Admin Center feed	URL for icon to display in the Extension Manager.
projectUrl	Required for publishing to the Windows Admin Center feed	URL to your extension's website. If you don't have a separate website, use the URL for the package webpage on the NuGet feed.
licenseUrl	Required for publishing to the Windows Admin Center feed	URL to your extension's end user license agreement.
files	Required	These two settings set up the folder structure that Windows Admin Center expects for UI extensions and Gateway plugins.

Note

As of Windows Admin Center 2410, the `packageType` field has been deprecated and should not be used. Continued use of this field may result in improper reading of the NuGet file.

Build the extension NuGet package

Using the `.nuspec` file you created, you now need to create the NuGet package `.nupkg` file, which you can upload and publish to the NuGet feed.

1. Download the nuget.exe CLI tool from the NuGet client tools website.
2. Run `nuget.exe pack <>.nuspec file name>` to create the `.nupkg` file.

Sign your extension NuGet package

Any `.dll` files included in your extension are required to be signed with a certificate from a trusted Certificate Authority (CA). By default, unsigned `.dll` files are blocked from being executed when Windows Admin Center is running in Production Mode.

Any Javascript files included in your extension are required to be signed with a certificate from a trusted Certificate Authority (CA).

The extension NuGet package must also be signed to ensure the integrity of the package. Windows Admin Center uses a Kestrel server which loads TLS/SSL certificates by the subject name of certificate. Make sure only one certificate is installed with a unique subject name and that the certificate is valid.

When you publish an extension to Microsoft's Windows Admin Center extension feed, Microsoft will sign the package on top of your signature.

Tip

To verify that your package is signed properly, you can use the following command:

Windows Command Prompt

```
nuget.exe verify -All <packageName>.nuget
```

Test your extension NuGet package

Your extension package is now ready for testing! Upload the `.nupkg` file to a NuGet feed or copy it to a file share. To view and download packages from a different feed or file share, you need to change your feed configuration to point to your NuGet feed or file share. When testing, make sure the properties are displayed correctly in Extension Manager, and you can successfully install and uninstall your extension.

Publish your extension to the Windows Admin Center feed

By publishing to the Windows Admin Center feed, you can make your extension available to any Windows Admin Center user. Because the Windows Admin Center SDK is still in preview, we'd like to work closely with you to help resolve development issues and help you deliver a quality product and experience to your users.

Prior to submitting an extension review request to Microsoft, you must send an email to wacextensionrequest@microsoft.com expressing the intent to publish an extension to the public feed. We provide you with copies of the Extension Publisher Agreement and the Extension Participation Policy to review and acknowledge in writing.

Before releasing the initial version of your extension, we recommend that you submit an extension review request to Microsoft at least 2-3 weeks before release. Allowing 2-3 weeks before release ensures we have sufficient time to review and for you to make any changes to your extension if necessary. After your extension is ready to be published, you'll need to send it to us for review. If your extension is approved, we publish it to the feed for you. By sending Microsoft your extension package, you agree to be bound by the terms of the Extension Publisher Agreement and the Extension Participation Policy.

Afterwards, if you want to release an update to your extension, you need to submit another request for review. Depending on the scope of change, turnaround times for update reviews are generally shorter.

Submit an extension review request to Microsoft

To submit an extension review request, provide the following information and send as an email to wacextensionrequest@microsoft.com. We reply to your email within a week.

Windows Admin Center - Extension review request information:

- Name and email address of extension owner/developer (up to 3 users). If you're releasing an extension on behalf of your company, provide your company email address.

- Company name (Only required if you're releasing an extension on behalf of your company)
- Extension name
- Release target date (estimate)
- For new extension submission - Extension description (early design wire frames, screen mockups, or product screenshots recommended)
- For extension update review – Description of changes (include product screenshots if UI significantly changed)

Submit your extension package for review and publishing

Make sure you follow the preceding instructions for creating an extension package and the `.nuspec` file is defined properly and files are signed. We also recommend that you have a project website including:

- Detailed description of your extension including screenshots or video
- Email address or website feature to receive feedback or questions

When you're ready to publish your extension, send email to wacextensionrequest@microsoft.com. We provide instructions on how to send us your extension package. After we receive your package, we'll review it. After your extension package is approved, we'll publish to the Windows Admin Center feed.

Feedback

Was this page helpful? Yes No

Provide product feedback

BiitOps

Article • 09/07/2022 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

BiitOps provides operational insights and data to help management and IT Operations make informed decisions.

BiitOps is a company striving to bring clarity into an IT world of ever-increasing complexity. Our advanced solution automatically collects data across all systems within the IT landscape, then uses this data to structure, visualize, and present information, providing insights and transparency. BiitOps helps customers gain knowledge, strengthen their IT operations, and support IT management track operational compliance.

About BiitOps

- BiitOps is a software company that provides insights through data and has developed a software solution that works across even the most complex IT landscapes.
- BiitOps DataEngine is the core of BiitOps' product portfolio, and it collects, structures, and stores data, identifies changes, and presents data through a high-performant Rest API.
- BiitOps Insights translates data into knowledge through custom-made visualizations using standard business intelligence tools.
- BiitOps Integrations is a solution that integrates data from BiitOps DataEngine directly into third-party products and solutions.

About BiitOps Values

- BiitOps can help change IT management into a predict and prevent strategy rather than a wait-and-see approach, helping make informed decisions based on data rather than incidents.
- BiitOps provides overview, insights, and operational certainty:
  - Overview of servers, clients, services, updates, rights, system assets, and configurations
  - Insights into the current state, historic states, and operational compliance
- BiitOps ensures that data needed for documentation and operational monitoring is kept current and ready for review by senior management and during IT audits.
- BiitOps can help reduce the costs of running an under-resourced IT organization and ensure that business-critical decisions are made on an informed basis.

BiitOps Inventory Extension

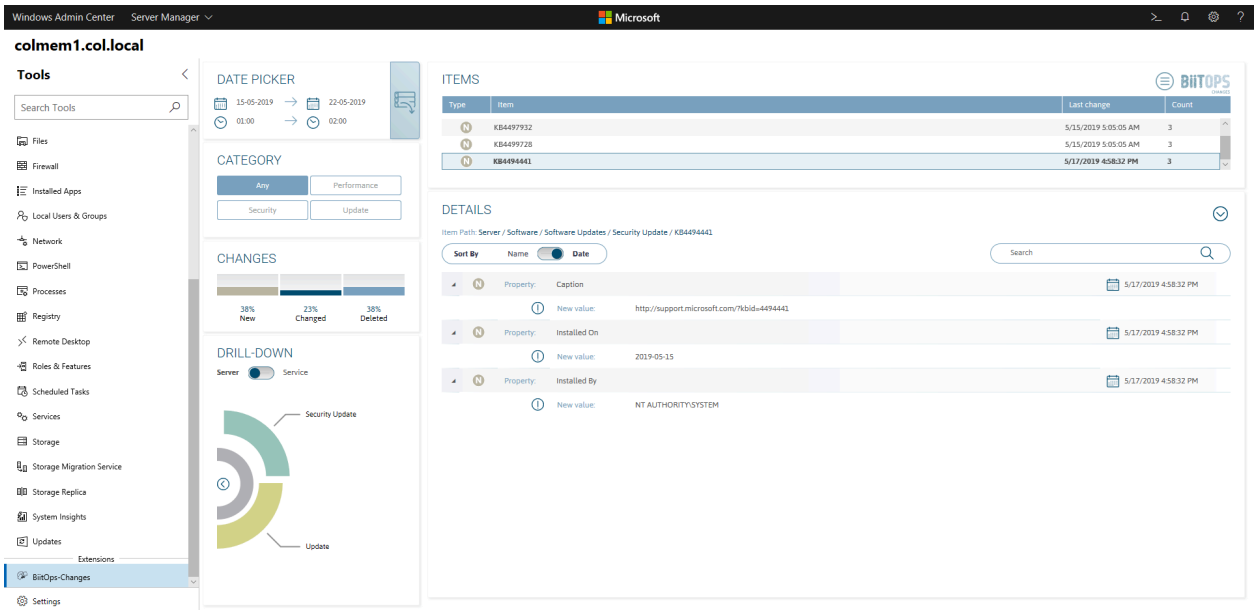
BiitOps Inventory for Windows Admin Center provides a fast, easy, and powerful way to find the information about a computer you need in any given situation without having to search through different Windows Admin Center extensions or even resort to using remote PowerShell. BiitOps DataEngine is the central repository for automatically updated configuration and asset information, and the Inventory extension delivers out-of-box high-value based on this data. This solution works with all versions of Windows servers and clients currently supported by Microsoft.

BiitOps Inventory extension requires a BiitOps DataEngine instance. Visit the BiitOps website to schedule a free demo.

BiitOps Changes Extension

BiitOps Changes for Windows Admin Center gives you insight into changes to hardware, software and configuration settings on your Windows Server physical/virtual machines. The BiitOps Changes extension enables you to quickly and cost-effectively determine changes which cause issues related to compliance, reliability or security. It will show precisely what is new, what has changed and what has been deleted in a single-pane-of-glass. It works on both servers and selected services on Microsoft Windows Server.

Learn more by visiting the BiitOps product site or subscribe for a free trial.



Feedback

Was this page helpful? Yes No

Provide product feedback

DataON MUST Extension

Article • 12/23/2021 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Integrated monitoring and management for Microsoft hyper-converged infrastructure

DataON is the industry-leading provider of hyper-converged infrastructure and storage systems optimized for Microsoft Windows Server environments. Exclusively focused on delivering Microsoft applications, virtualization, data protection, and hybrid cloud services, it has over 650 enterprise deployments and over 120PB of Storage Spaces Direct deployments.

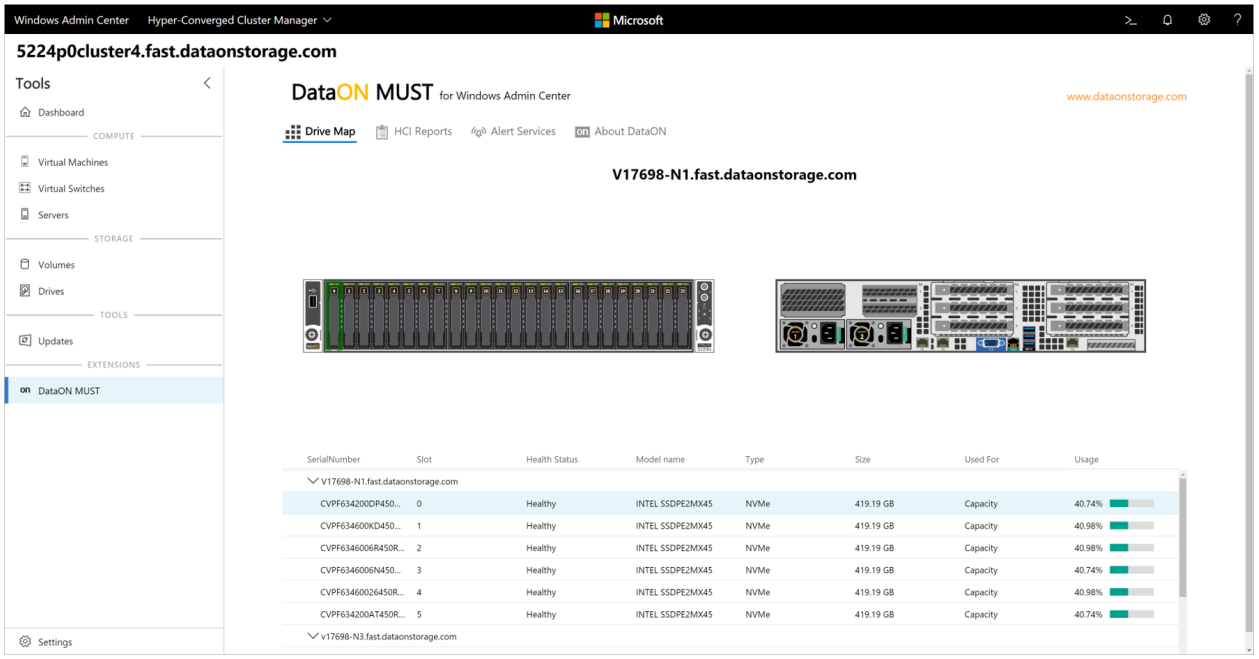
DataON's MUST extension for Windows Admin Center is a prime example of the value that integrating two complementary products can deliver to customers, bringing monitoring and management and end-to-end insight into hardware and software together across an entire cluster in a unified experience.

> *"We've taken our standalone MUST visibility, monitoring, and management tool and enabled it to work within Windows Admin Center. Customers will benefit from the expanded capabilities that MUST provides, and the combination of MUST and Windows Admin Center from a single console will provide the ultimate management experience for Windows Server-based infrastructure."*
>
> -- Howard Lo, Vice President of Sales and Marketing, DataON

The MUST extension extends the functionality of Windows Admin Center by providing features such as:

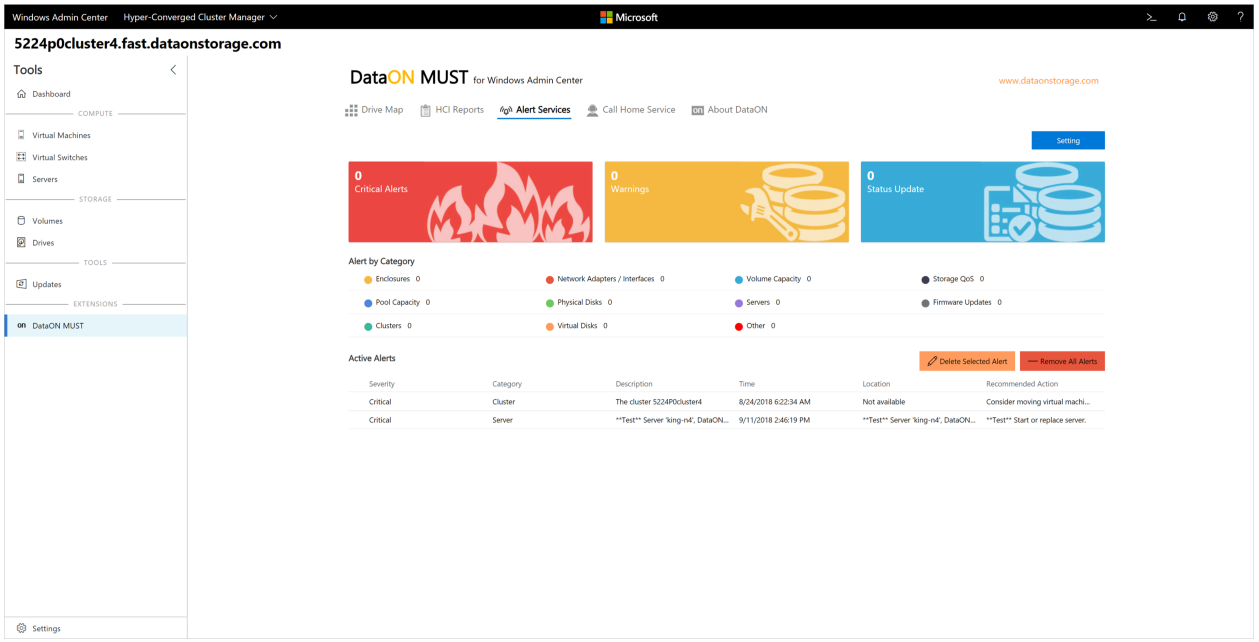
- **Historic Data Reporting** – Provides real-time and monthly dashboards of your system performance data including IOPS, latency, throughput on your cluster, storage pool, volume, and nodes.
- **Disk Mapping** – MUST displays the device types and components in each of the nodes, providing a clear disk map of your entire node. It shows the number of disks, disk type, location and slot of each drive, and disk health status.
- **System Alerts** – Leverages Windows Health Service faults to identify hardware failures, configuration issues, and resource saturation. It also provides a multi-level assessment of specific locations, fault descriptions, and recovery actions. You can also leverage third-party SNMP monitoring traps to alert you when you need disk or hardware replacements.

- **SAN-like Call Home Service** – Prompted by system alerts, administrators can have automated email alerts sent to key contacts.



Disk mapping in the DataON MUST extension for Windows Admin Center

> *"It's great that Windows Admin Center allows for extensions such as DataON MUST so I can use both tools within the same console, and I like how seamless that integration is. Windows Admin Center and DataON MUST together really does allow us to be more efficient and saves our team a ton of time. It allows us to achieve our administrator tasks a lot quicker than what we had before."*
>
> -- Matt Roper, Facilitator of Technology Support Services, Cherokee County (GA) School District



Alert Services in the DataON MUST extension for Windows Admin Center

> *"MUST has been very valuable and was a big selling point. To us, it demonstrated a commitment from DataON to support Microsoft hyper-converged infrastructure. The inclusion of MUST with their S2D appliance is what completes the solution with Storage Spaces Direct as a viable SAN replacement."*
>
> -- Benjamin Clements, President, Strategic Online Systems, Inc.

Feedback

Was this page helpful? Yes No

Provide product feedback

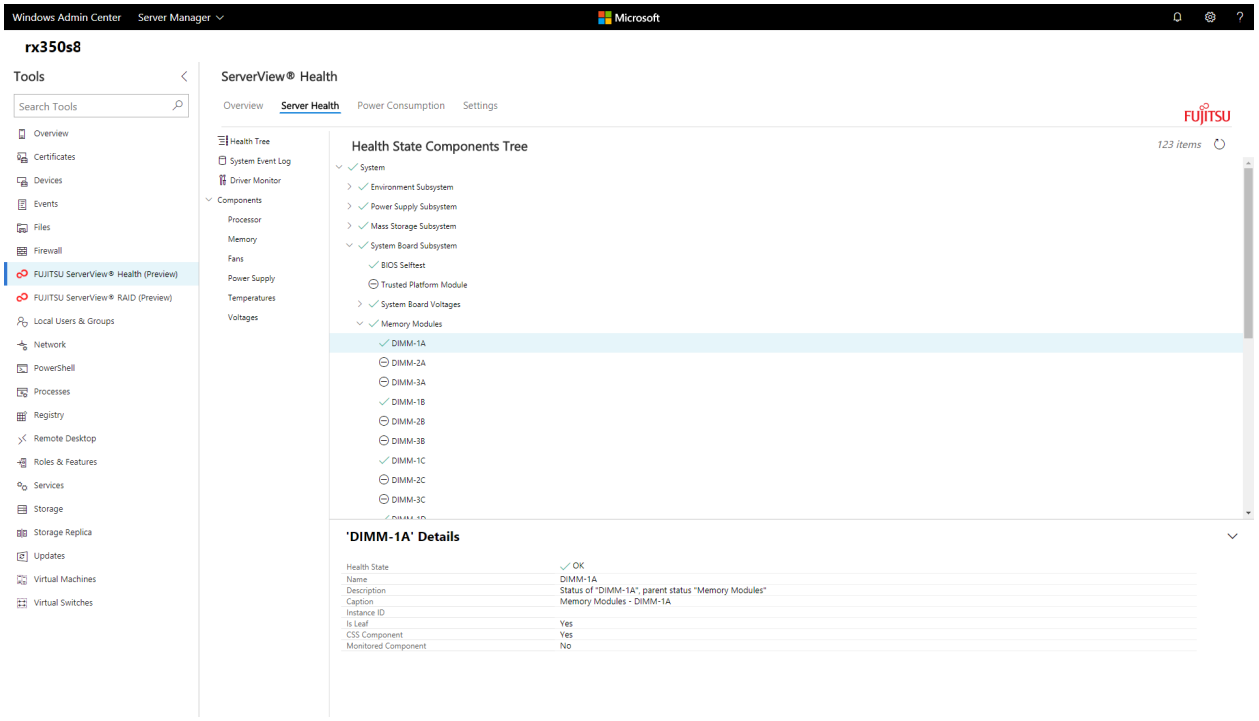
Fujitsu ServerView Health and RAID extensions

Article • 12/23/2021 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Bringing end-to-end visibility, from operating system to hardware, into Windows Admin Center

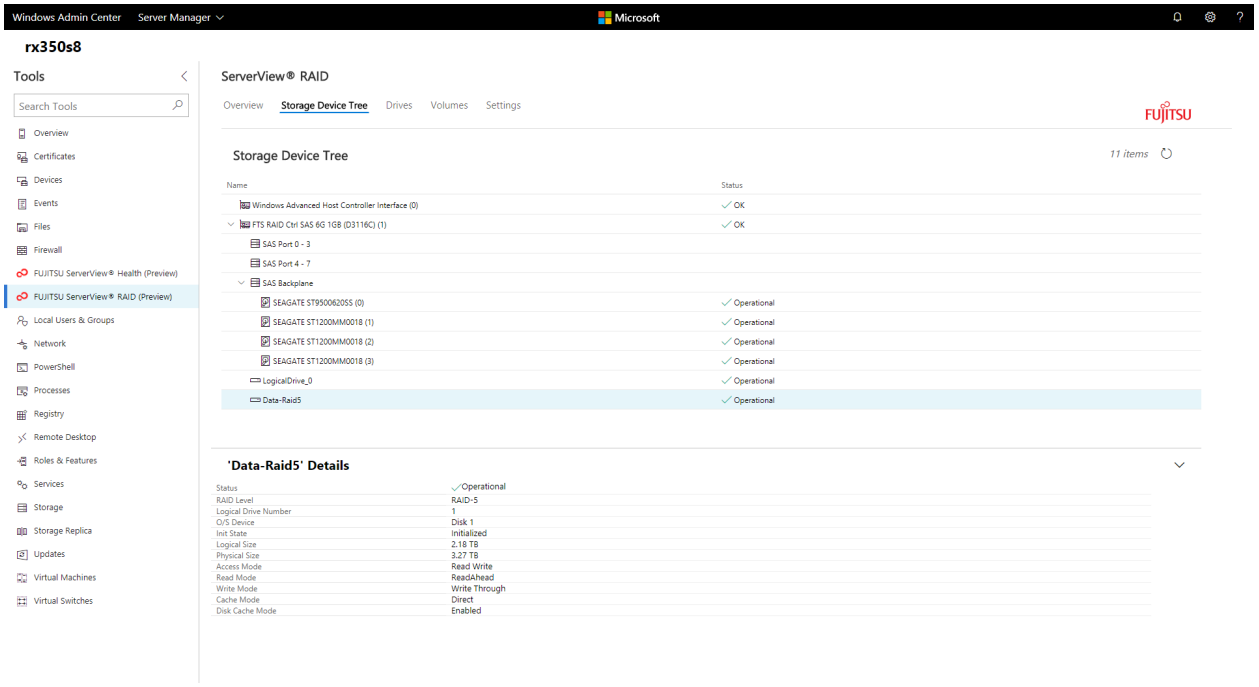
Fujitsu is a leading Japanese information and communication technology company and a manufacturer of PRIMERGY and PRIMEQUEST server products. The Fujitsu ServerView management suite provides a comprehensive toolset for server lifecycle management including a server-side agent that provides a CIM and PowerShell interface for hardware management.

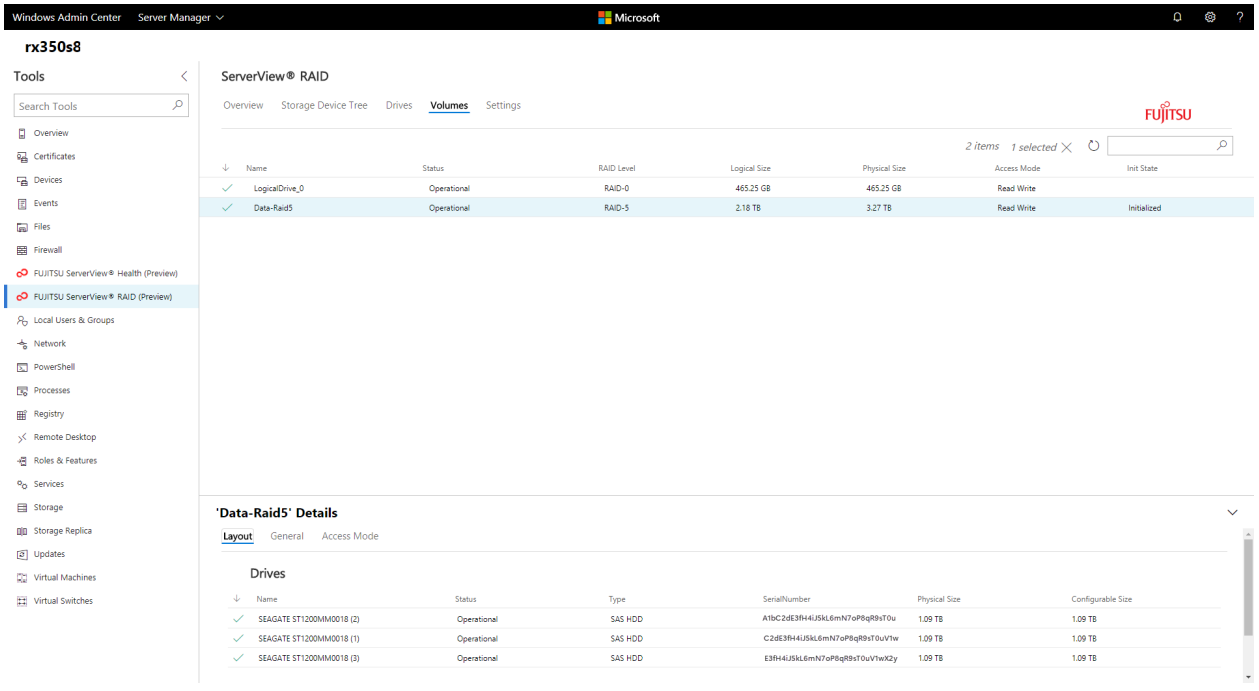
Fujitsu saw an opportunity to easily integrate with Windows Admin Center as it provided CIM and PowerShell interfaces that could communicate with the server-side agents. The development team at Fujitsu was able to easily implement the CIM calls they were familiar with to the agent and visualize the information within Windows Admin Center using the available UI components.



Once the team became familiar with the Windows Admin Center SDK, adding UI to expose additional hardware information was often simply a few more lines of HTML code and they were quickly able to expand from a single tool to displaying a summary

view of hardware component health, detailed views for system event logs, driver monitor, separate views for processor, memory, fans, power supplies, temperatures and voltages, and even an additional tool for RAID management. Using UI controls available in the SDK such as the tree, grid and detail pane controls enabled the team to quickly build UI and also achieve a visual and interaction design very similar to the rest of Windows Admin Center.





The partnership between Fujitsu and the Windows Admin Center team clearly shows the value of integration within Windows Admin Center, enabling customers to have end-to-end insight into server roles and services, to the operating system, and to hardware management.

Feedback

Was this page helpful? Yes No

Provide product feedback

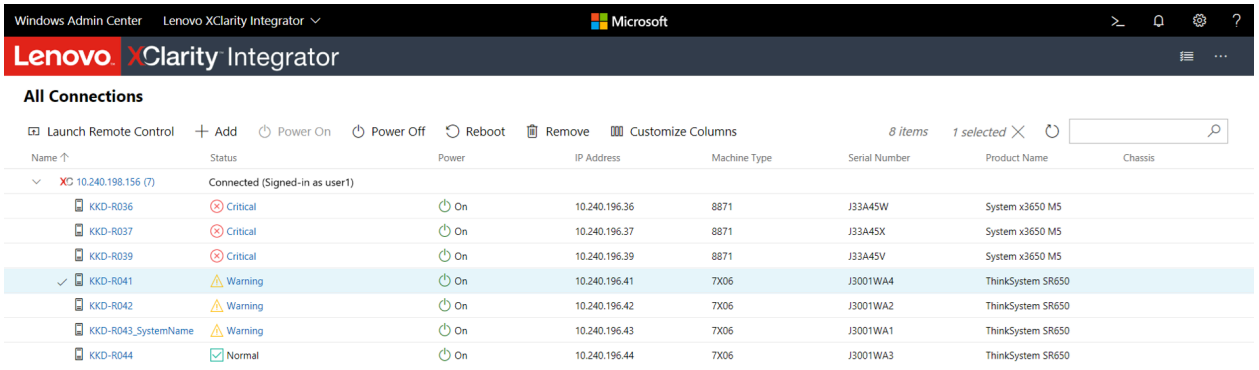
Lenovo XClarity Integrator Extension

Article • 12/23/2021 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

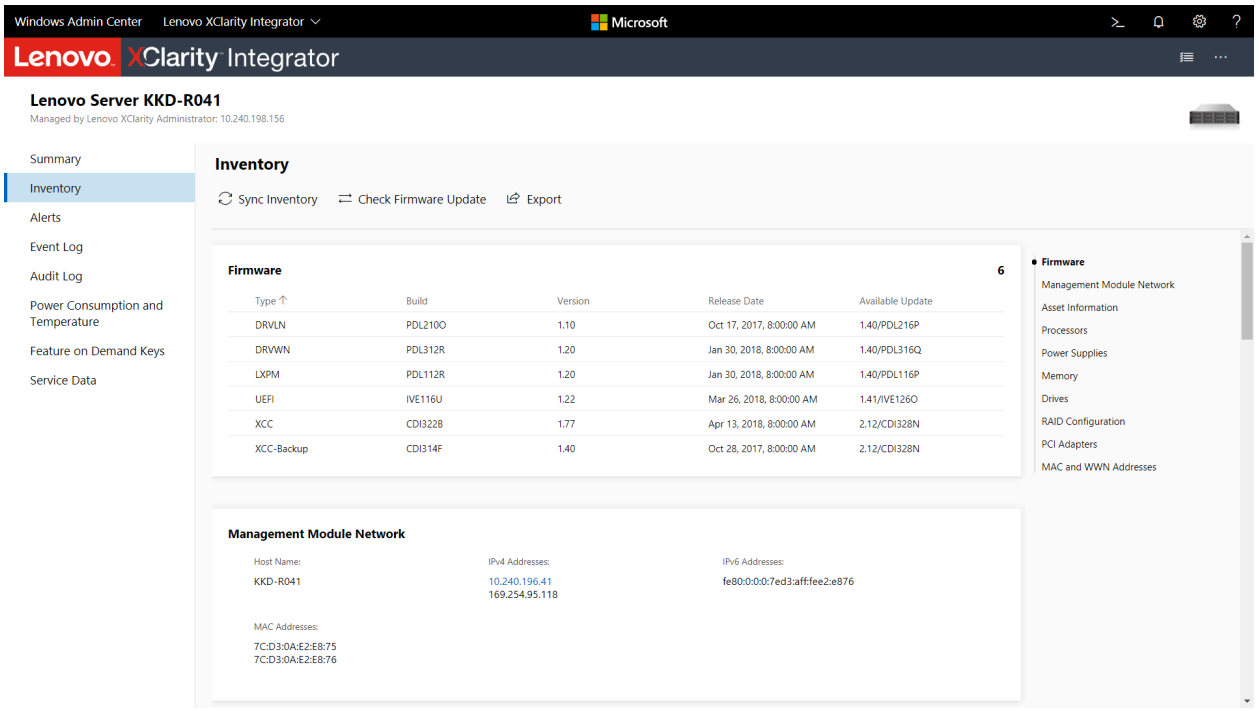
Integrated hardware management everywhere!

The Lenovo XClarity Integrator extension for Windows Admin Center provides administrators a seamless experience to manage Lenovo infrastructure directly from Windows Admin Center. The XClarity Integrator extension includes a standalone solution extension and also extends the existing Server Manager, Failover Cluster Manager, and Hyper-Converged Cluster Manager solutions in a single, unified UI to enable simple server management.

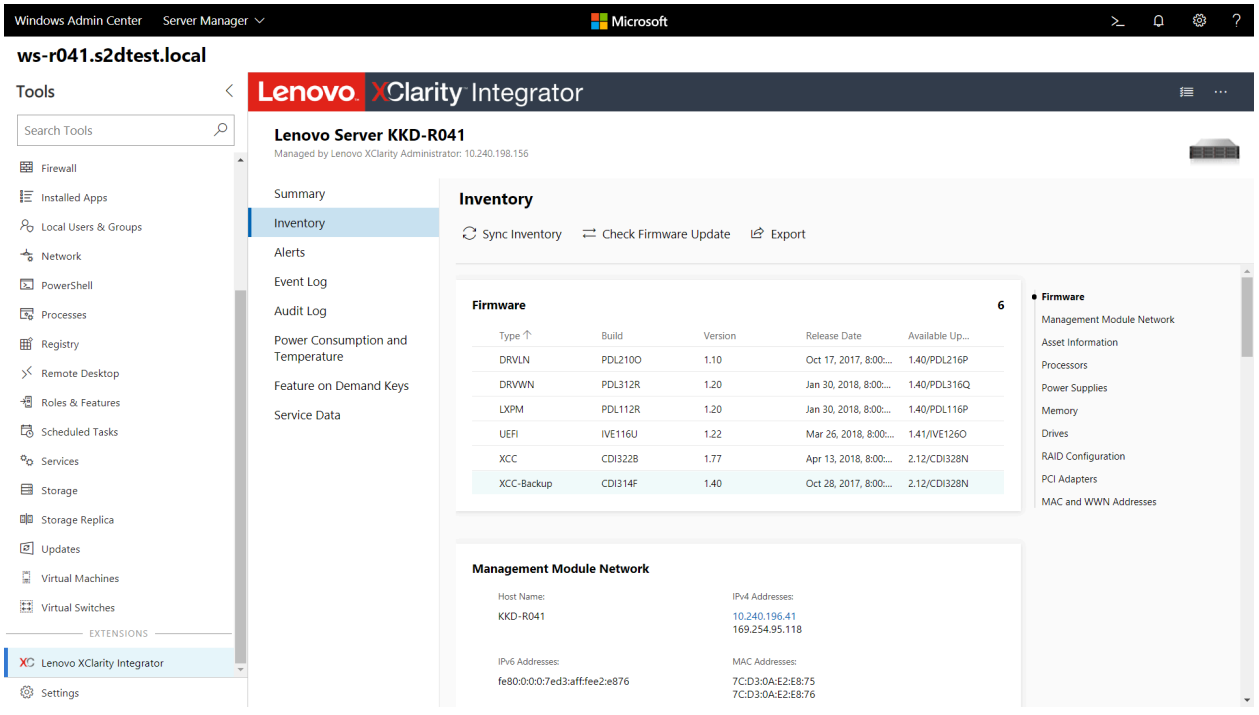
The solution extension included in the XClarity Integrator extension allows connecting to a Lenovo XClarity Administrator, Lenovo rack or tower servers, or all the servers in an entire chassis at once. Once the servers are added, you can see the overall health status for all added nodes.



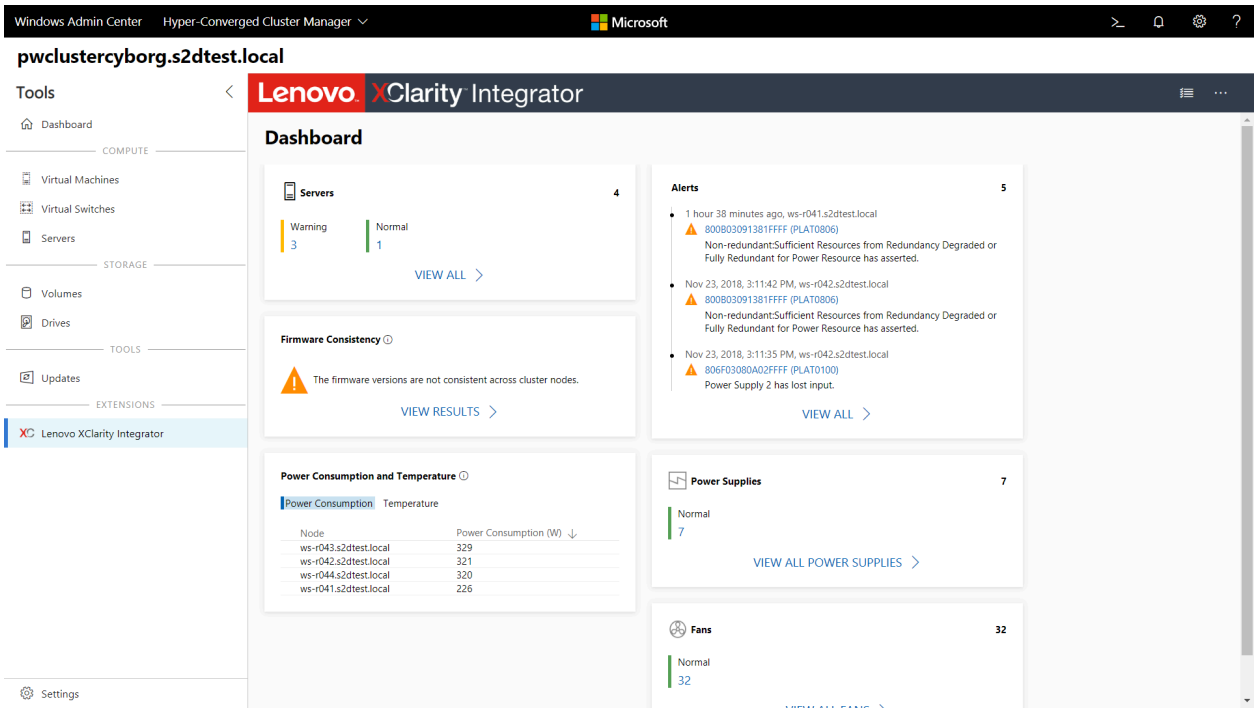
By selecting a server, you can view the server's hardware inventory, available firmware updates, alerts, events, logs, power consumption and temperature. You can also run operations such as Remote Control and power on/off.

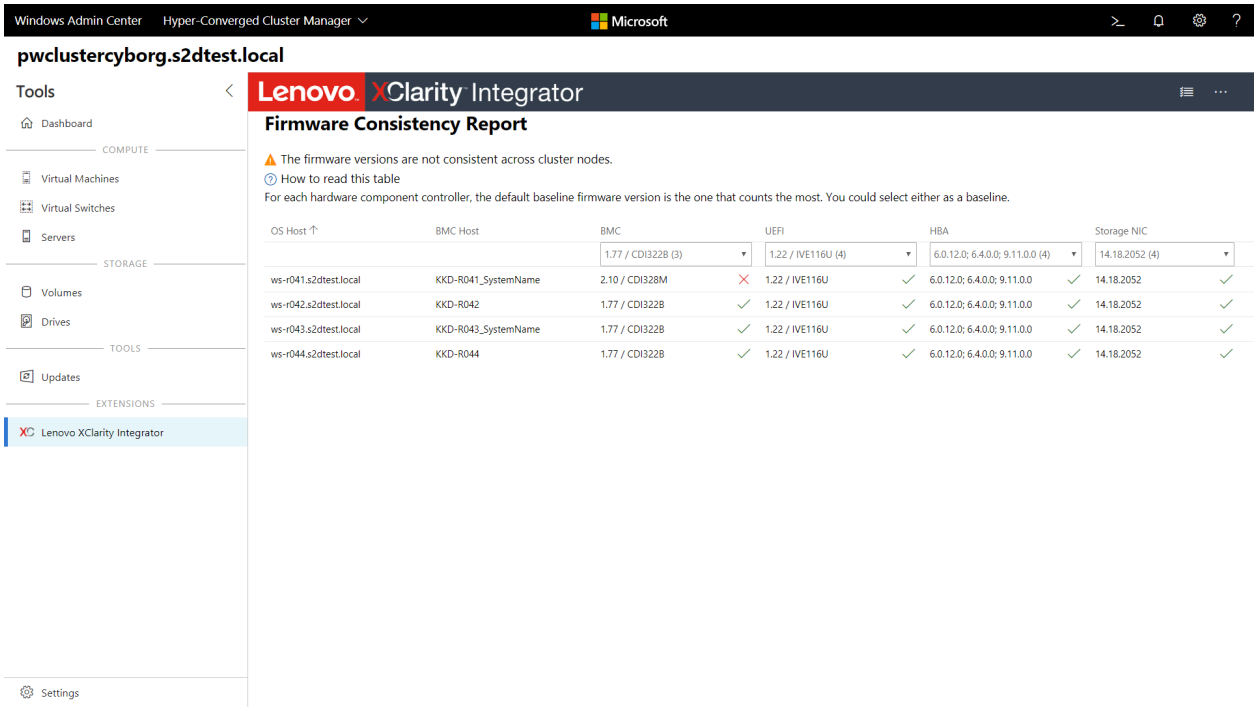


The same tools are available as a tool extension when managing servers within Windows Admin Center as well, allowing you to seamlessly switch between managing your infrastructure software and hardware.

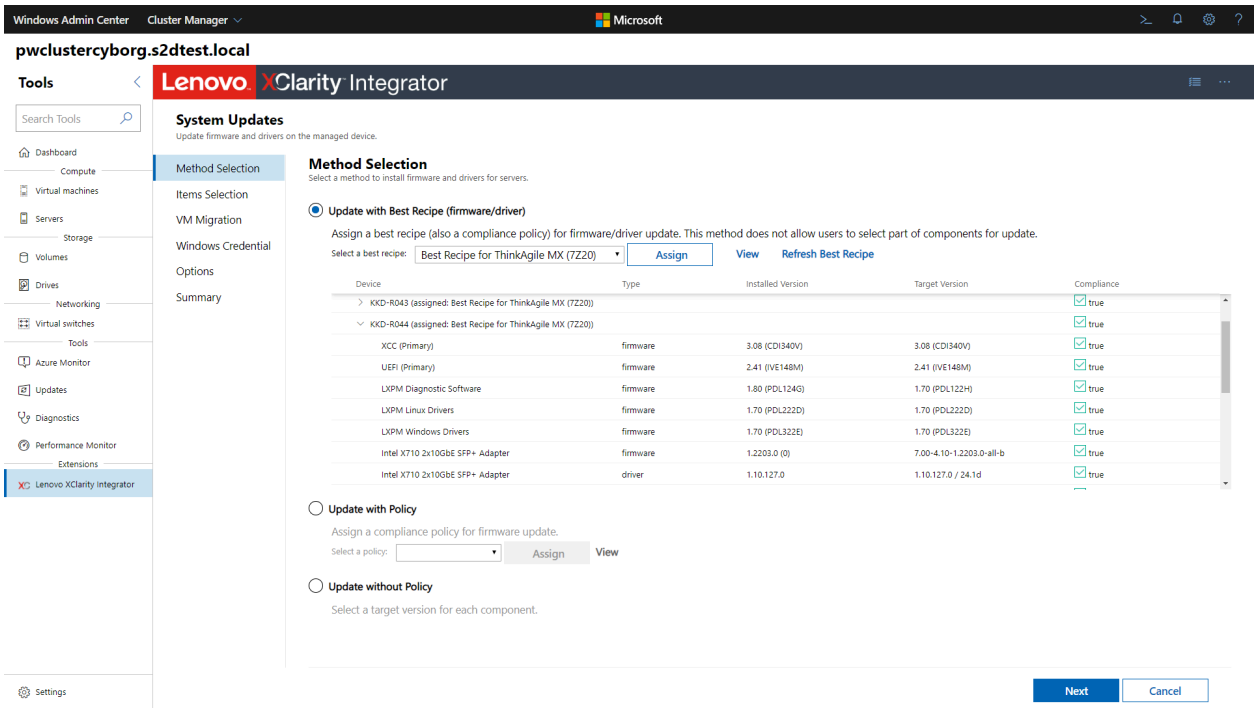


The tool extension for failover clusters and hyper-converged clusters provides a dashboard displaying overall cluster hardware health status, status alerts, firmware consistency status and report, power consumption and temperature, and fan and power supply health status.

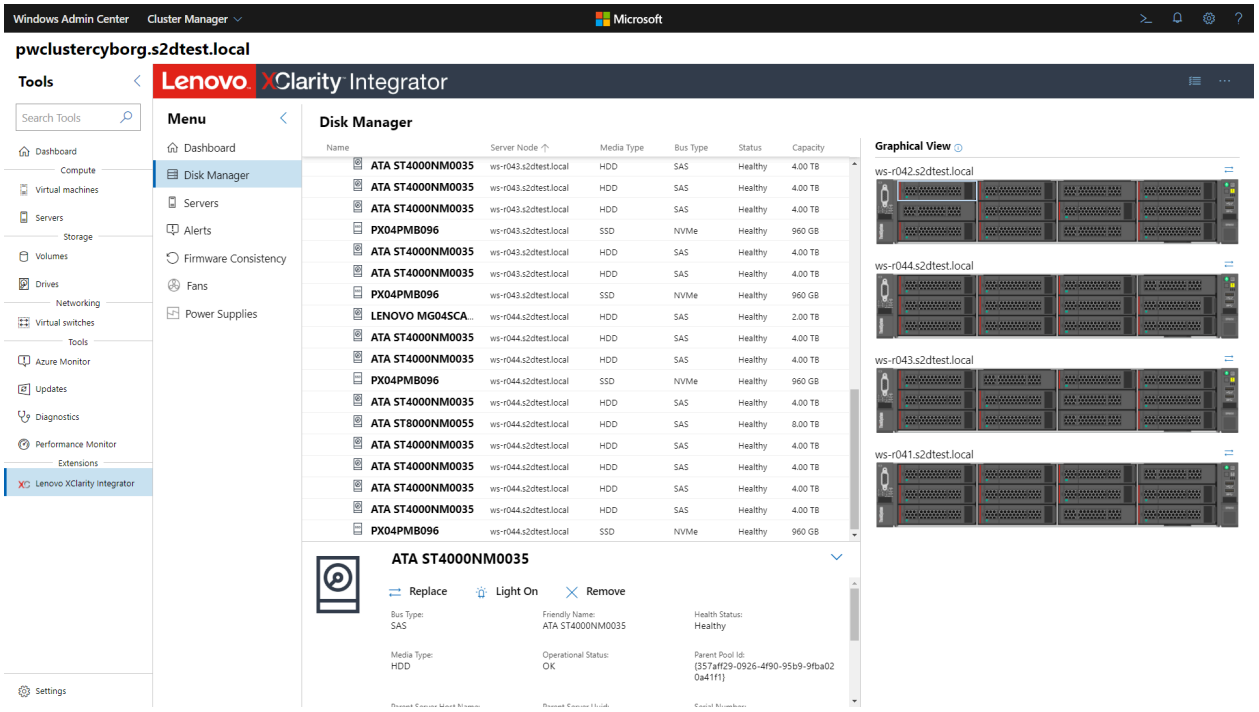




The solution extension and tool extensions for Windows servers and clusters provide the rolling (cluster-aware) server update functions. This can help prevent any workload interruption during server updates. Currently, the extensions support individual firmware updates and compliance policy firmware updates for any Lenovo servers. They also support best recipe firmware/driver updates for Lenovo ThinkAgile MX HCI cluster servers.



The tool extension for hyperconverged clusters provides the disk/storage pool management functions for Lenovo ThinkAgile MX HCI cluster servers. These functions include the server rear/front graphic view to present server and disk status. With the help of both the wizard and graphic view, operations like adding a disk to the storage pool, removing a disk from the storage pool, replacing a disk, or locating a server/disk by lighting on the server/disk location LEDs become much easier.



In version 2.1, Lenovo extensions introduced role-based access control capabilities. These capabilities are enabled if Lenovo XClarity Administrator is applied for hardware management.

Learn more about the Lenovo XClarity Integrator offerings for Windows Admin Center on the Lenovo website.

Feedback

Was this page helpful? Yes No

Provide product feedback

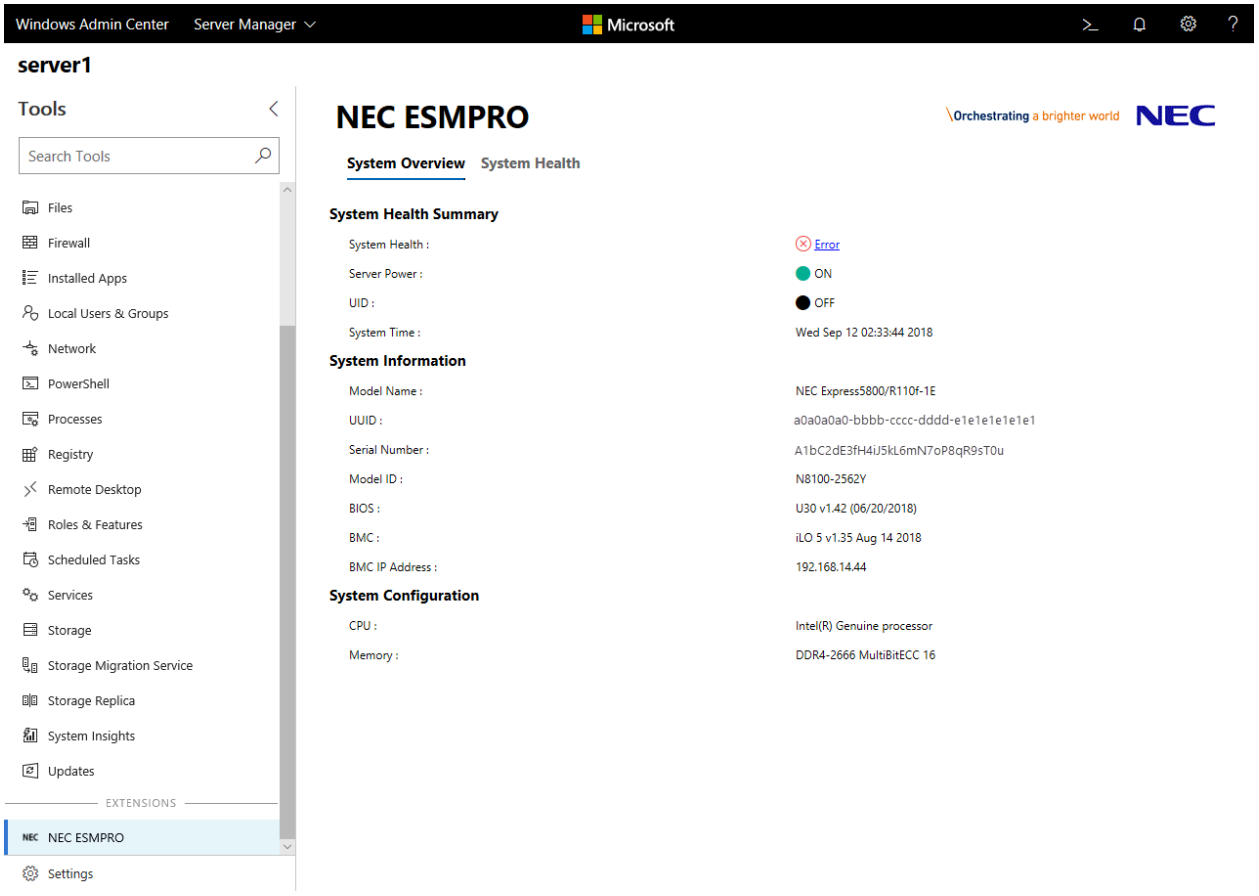
NEC ESMPRO Extension

Article • 12/23/2021 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

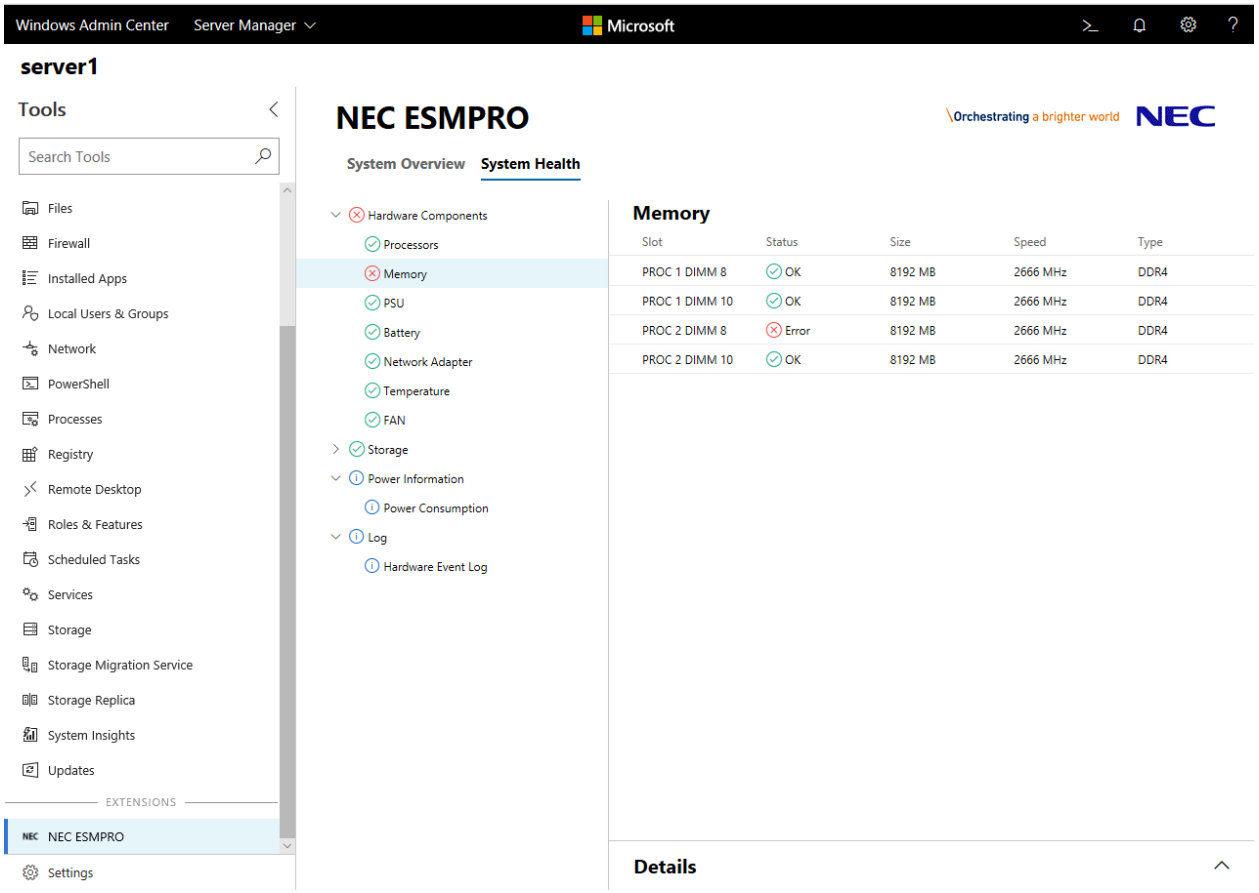
NEC provides products for businesses, ranging from terminals to network and computer equipment, software products and service platforms, as well as integrated platforms based on these products and services.

NEC ESMPRO is NEC's server management software to manage NEC Express5800 series servers, and the NEC ESMPRO extension for Windows Admin Center enables showing hardware and RAID information of NEC Express5800 series servers in Windows Admin Center.

The NEC ESMPRO extension requires installing the NEC ESMPRO Manager as it retrieves server hardware information through the NEC ESMPRO Manager.



The NEC ESMPRO extension has two tabs, the 'System Overview' tab and the 'System Health' tab. In the System Overview tab, you can easily view the system's basic information.



In the System Health tab, you can view the detailed information of individual hardware components, such as processors, memory, power supplies, network adapters, temperature and fan. The status and configuration for RAID systems, power consumption and hardware event logs are also available.

The NEC ESMPRO extension for Windows Admin Center brings new experiences of server management to server administrators with the collaboration of NEC's hardware technology and Microsoft's software technology.

Feedback

Was this page helpful? Yes No

Provide product feedback

Pure Storage Extension

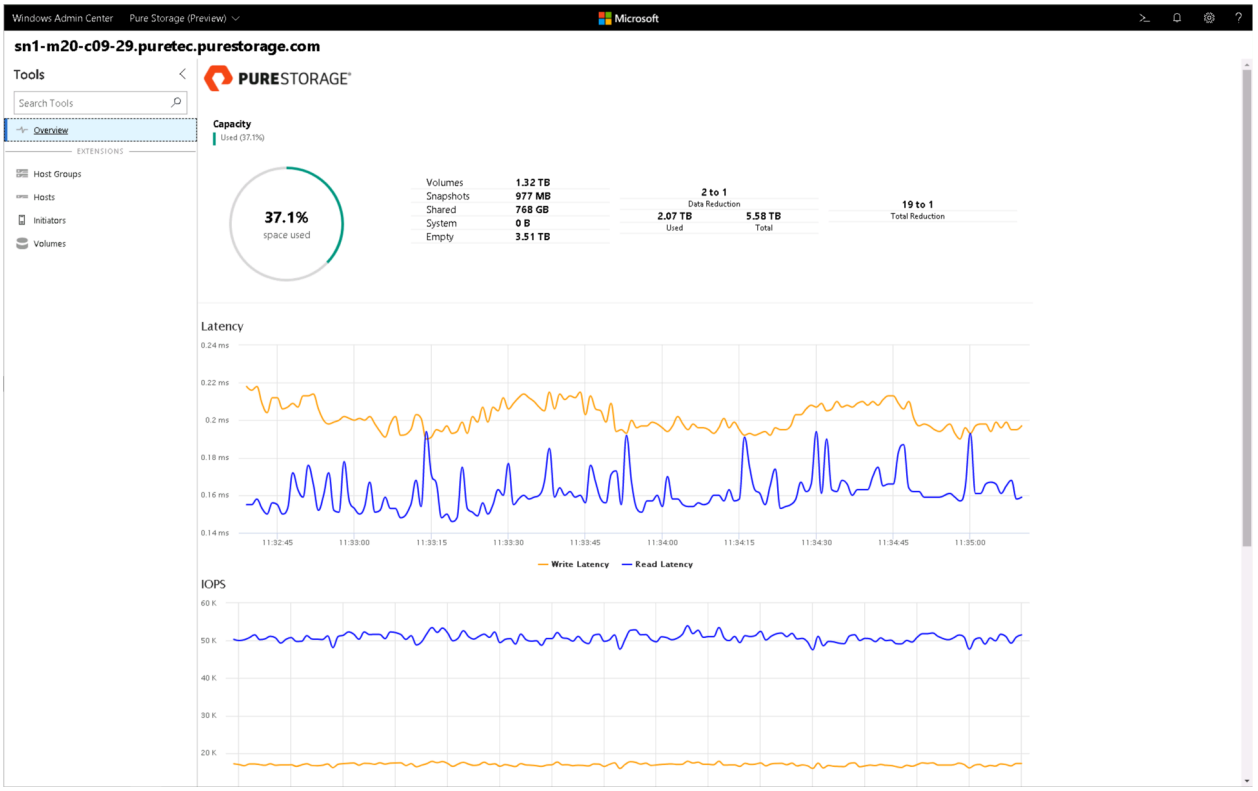
Article • 06/09/2022 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Providing End-to-End Array Management for Windows Admin Center

Pure Storage provides enterprise, all-flash data storage solutions that deliver data-centric architecture to accelerate your business for a competitive advantage. Pure is a Microsoft Gold Partner, certified for Microsoft Windows Server, and develops technical integrations for key Microsoft solutions such as Azure, Hyper-V, SQL Server, System Center, Windows PowerShell, and Windows SMB. Pure recently announced a tech preview of an extension supporting the latest release of Windows Admin Center that provides a single-pane view into Pure FlashArray products. From this extension, users are empowered from one tool to conduct monitoring tasks, view real-time performance metrics, and manage storage volumes and initiators.

Early on, when Windows Admin Center was known as “Project Honolulu”, Pure saw the value of being able to provide customers and partners the ability to manage multiple Pure Storage FlashArrays from the single pane of glass that Windows Admin Center provides.

When Pure started researching the use case with “Project Honolulu” they immediately realized the potential for providing a unified management experience between Windows Admin Center and FlashArray. Pure closely collaborated with the Windows Admin Center engineering team, which helped define the implementation details for the features. Pure was also able to provide feedback at the early stages of Windows Admin Center and make contributions to the Microsoft team.



> *"We have integrated a feature set that mimics our FlashArray web interface to enable direct management from within Windows Admin Center. Our customers and partners will benefit from a single pane of glass versus needing to work with two different management tools. In addition to the single point of management benefits customers will be able to contextually manage Windows Servers that are connected to the FlashArray."*
>
> -- Barkz, Technical Director Microsoft Solutions & Integration, Pure Storage

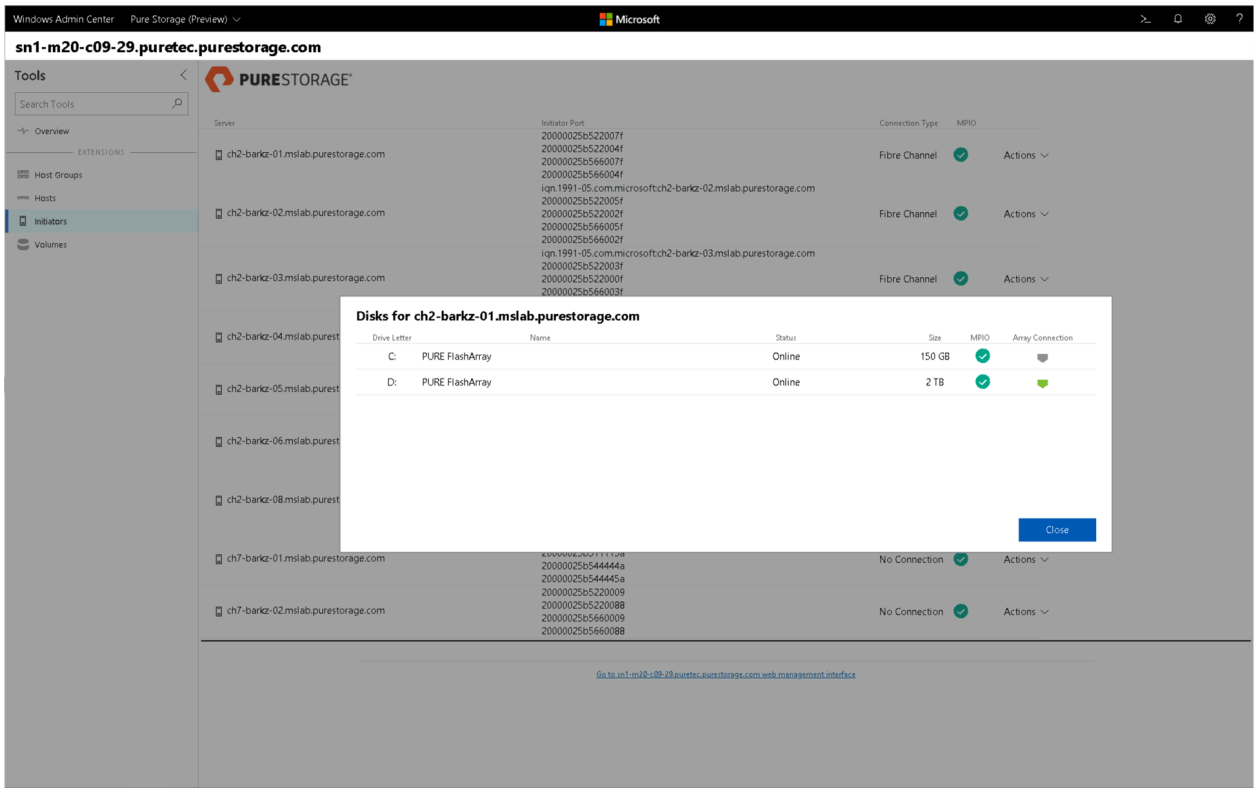
The features that are included in the Pure Storage Solution Extension include:

- Connecting to multiple FlashArrays.
- Viewing the FlashArray details, including IOPs, bandwidth, latency, data reduction and space management. These are all the same details you get from the FlashArray Management GUI.
- View configured host groups that are used to enable shared volume access for Windows Server hosts and Clustered Shared Volumes (CSVs).
- View Hosts — All of the connectivity information is available including Host Names, iSCSI Qualified Name (IQNs) and World Wide Names (WWNs).
- Manage Volumes — This includes the ability to create and destroy volumes. Once a volume is destroyed it will be placed in the Destroyed items bucket and you will need to Eradicate from the main FlashArray Management GUI.
- Manage Initiators — This is one of the most interesting features providing context to the individual servers being managed by the Windows Admin Center deployment. You can view the connected disks (volumes) to individual Windows

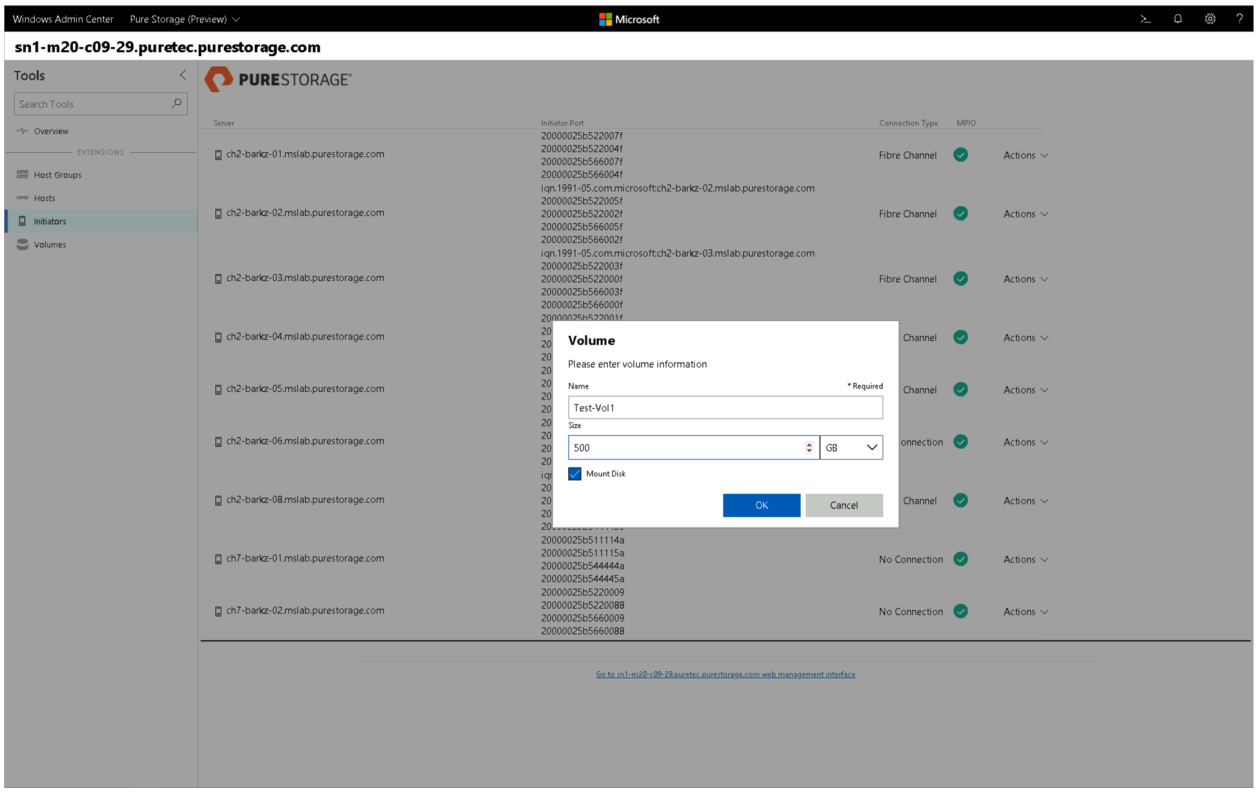
Servers, check if MultiPath-IO (MPIO) is installed/configured and creating/mounting new volumes.

A demonstration video has been created that shows all of the features that the Pure Storage Solution Extension provides.

The below screenshot illustrates viewing what disks (volumes) are connected to a specific Windows Server host. In addition to viewing the connectivity detail, we check if Multipath-IO is configured.



In addition to viewing the disks, new volumes can be created and immediately mounted to the host without having to use Windows Disk Management tool.



Since releasing our Technical Preview, the customer feedback collected so far has been very positive and has also provided us insight into different features to add in future releases.

Additional resources:

- Pure Storage extension announcement blog post
- PureReport podcast

Feedback

Was this page helpful? Yes No

Provide product feedback

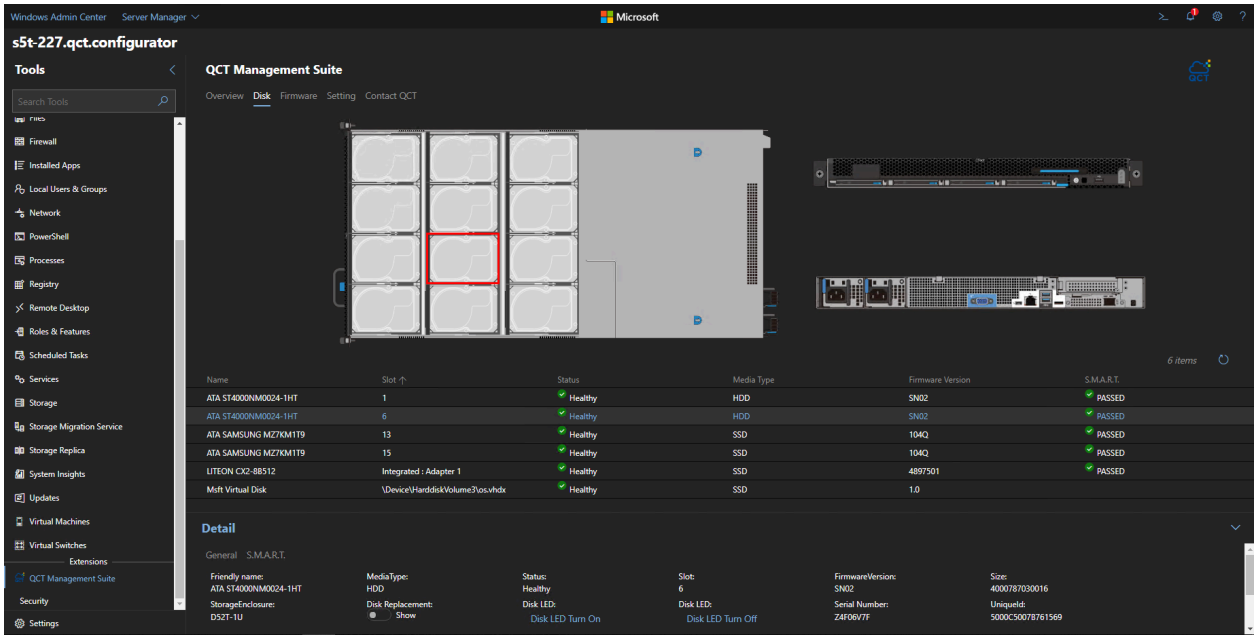
QCT Management Suite Extension

Article • 03/28/2025 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

A simple path to server infrastructure management

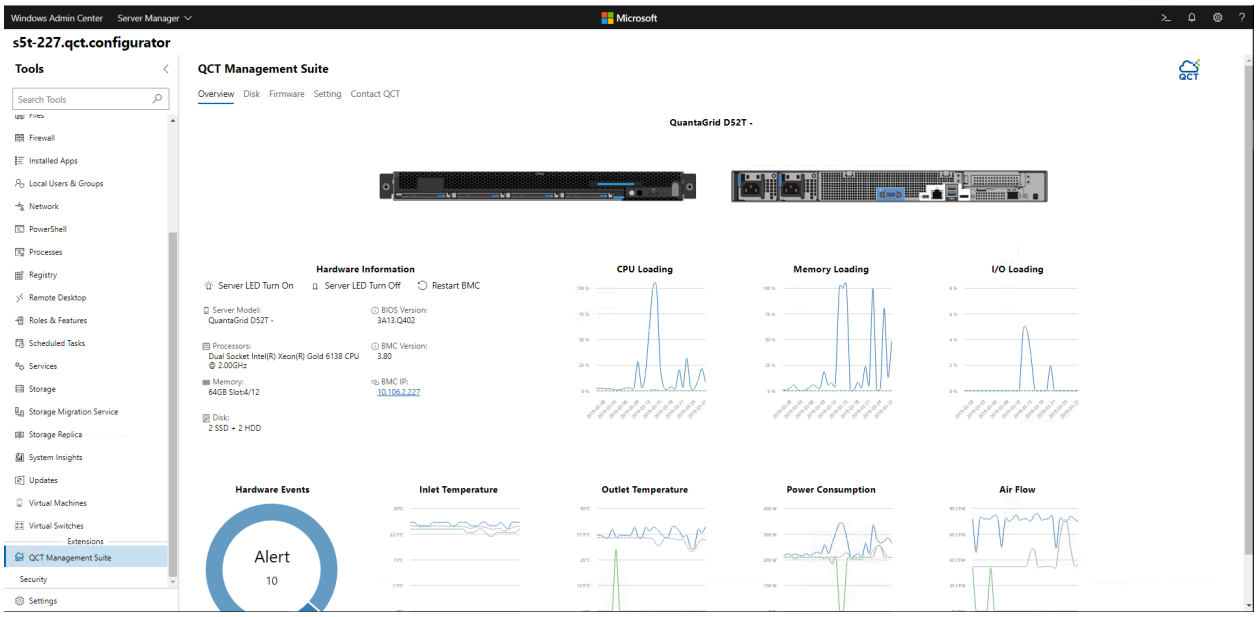
The QCT Management Suite extension for Windows Admin Center provides a single pane of glass dashboard for monitoring system configurations and managing server health of QCT Azure Local certified systems : QuantaGrid D52BQ-2U, QuantaGrid D52T-1ULH and QuantaPlex T21P-4U.

Driven by customer pain points around existing monitoring and management, QCT provides exclusive, complementary features and functions, which includes an overview of system event logs, monitoring drivers, and hardware component health to enhance the overall management experience.

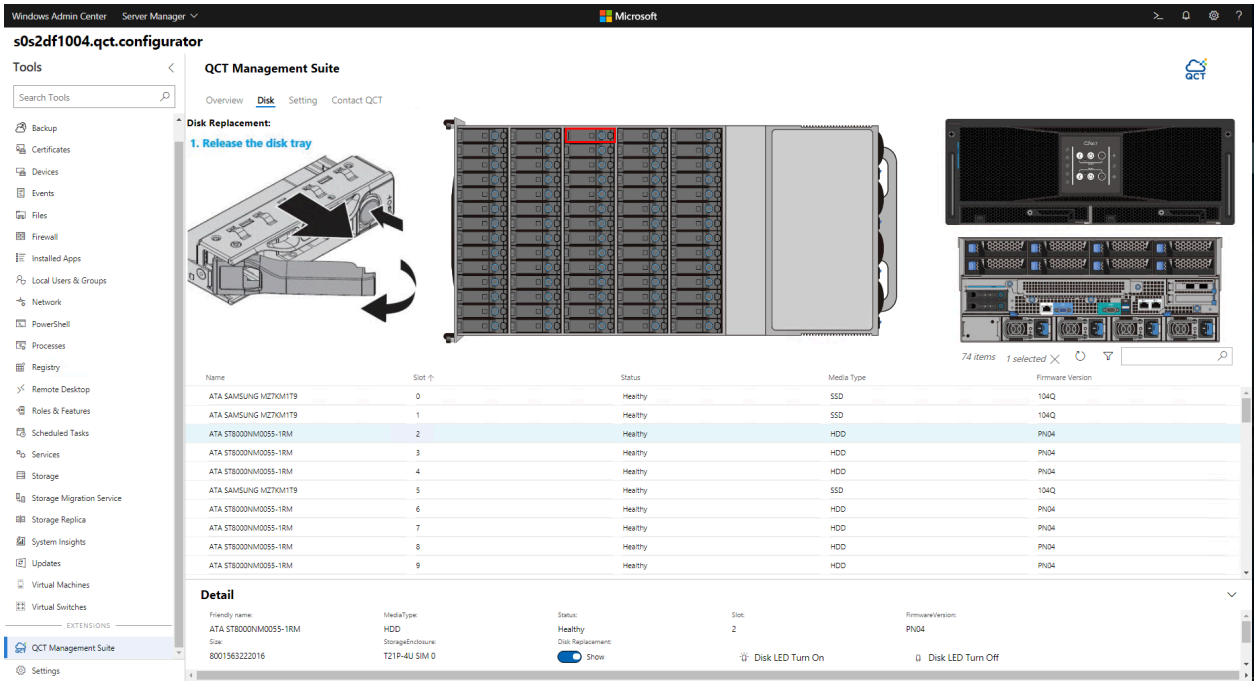


The QCT Management Suite extends the functionality of Windows Admin Center with the key features below:

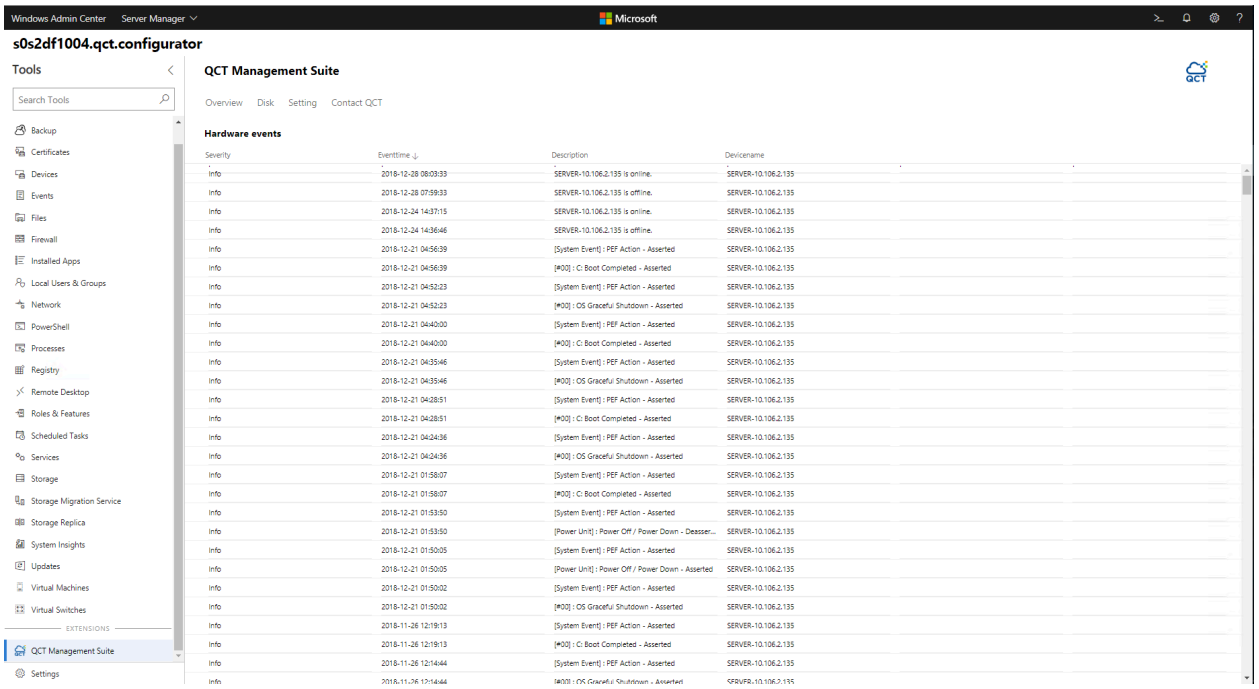
- **One-click exclusive hardware management** - An intuitive user interface displays hardware information, including model name, processor, memory and BIOS. IT administrators can restart the BMC with a simple one-click UI.



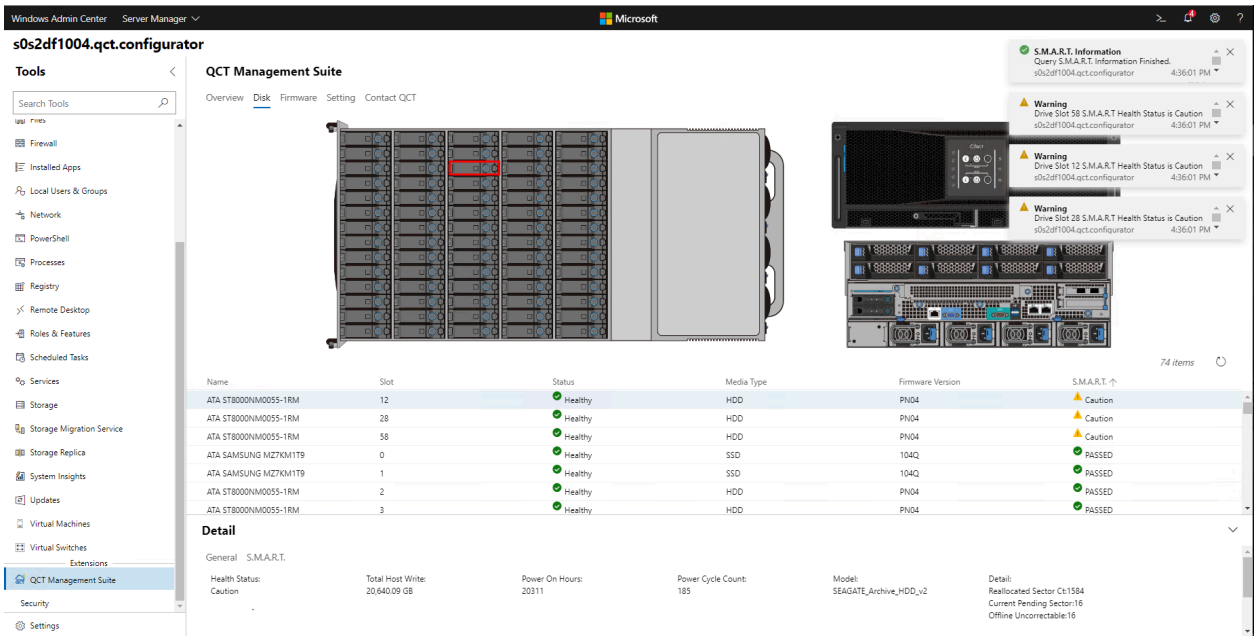
- **Disk mapping and LED identification for efficient service support** - QCT Management Suite wizard UI design displays the slot of each selected disk with an overview of disk profiles and LED light controls of selected disks for efficient replacement.



- Easy-to-use monitoring tool for hardware event log and health status.



- **Predictive disk management** - Evaluate the system condition with S.M.A.R.T information and unhealthy notifications which allow organizations to take action before total failure occurs.



Learn more about the QCT Management Suite for Windows Admin Center:

- QCT Management Suite (Windows Admin Center Extension)
- QCT Management Suite datasheet

Feedback

Was this page helpful? Yes No

Provide product feedback

Thomas-Krenn.AG Extension

Article • 12/23/2021 • Applies to: ✅ Windows Admin Center, ✅ Windows Admin Center Preview

Intuitive server and storage health management

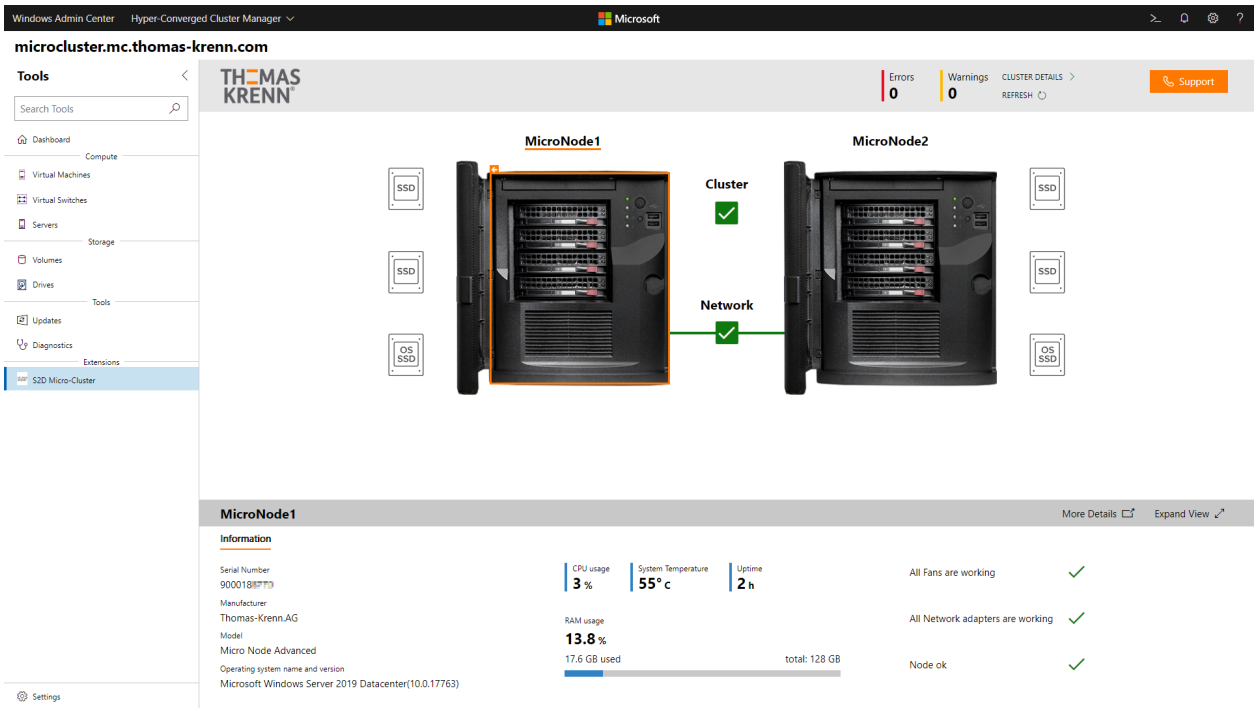
The Thomas Krenn.AG Windows Admin Center extension is designed specifically for the highly available, 2-node S2D Micro-Cluster appliance. The user-friendly, graphical web interface visualizes a Micro-Cluster's health status through a simple dashboard and allows you to drill down on storage devices, network interfaces or the entire cluster to view more details.

The extension provides intuitive access to information typically needed for first-level service and support calls, such as serial numbers, software versions, storage utilization and more. It is designed to be useful to admins who have no prior experience with Windows Server hyper-converged infrastructure.

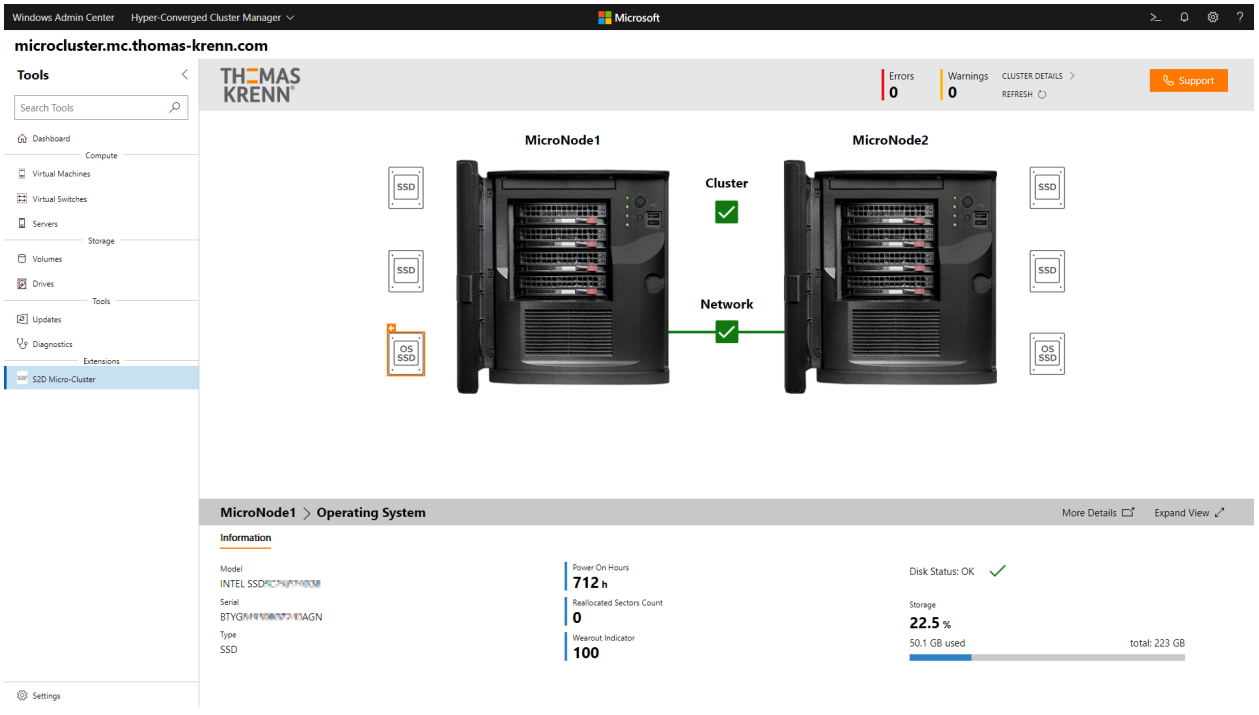
A few of the insights available are:

- General Information about the Micro-Nodes and the Micro-Cluster
- OS / boot device status
- Capacity HDD and caching SSD status
- Cluster events
- Network status and information

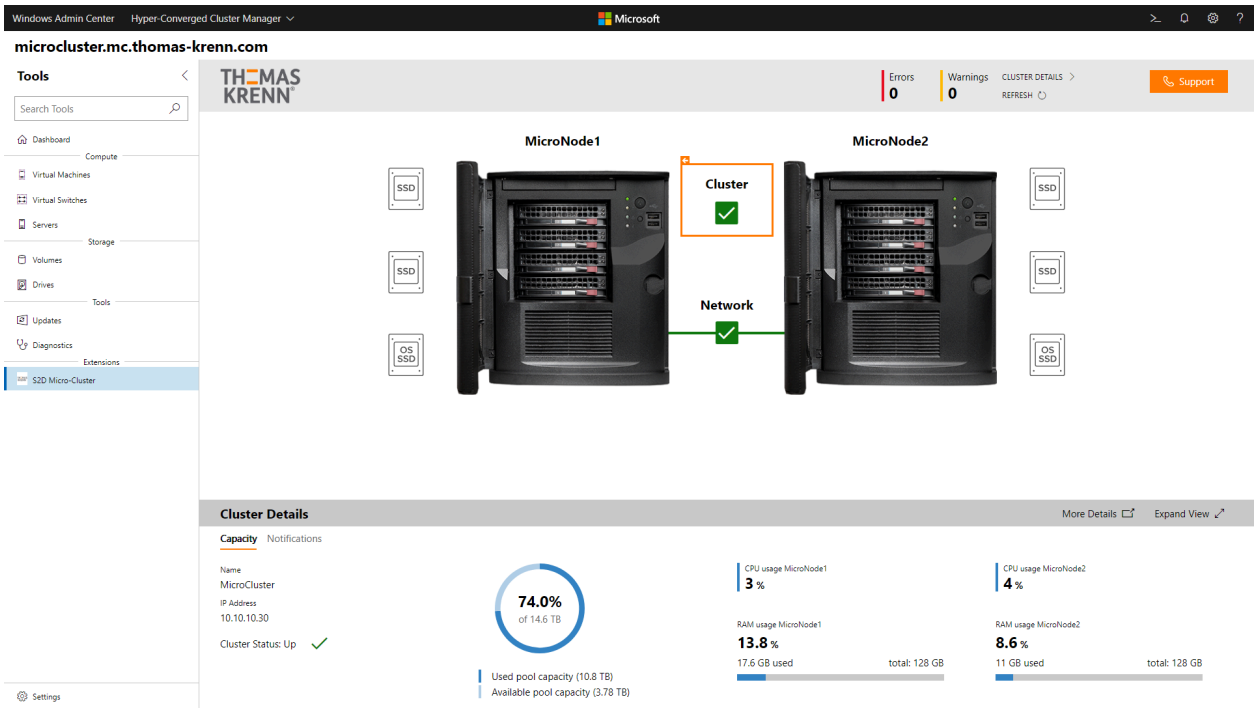
Use the dashboard to determine the cluster's health status and important system information such as serial numbers, model, OS version and utilization. Additionally, fan, NIC and overall node hardware health are displayed on the dashboard as well.



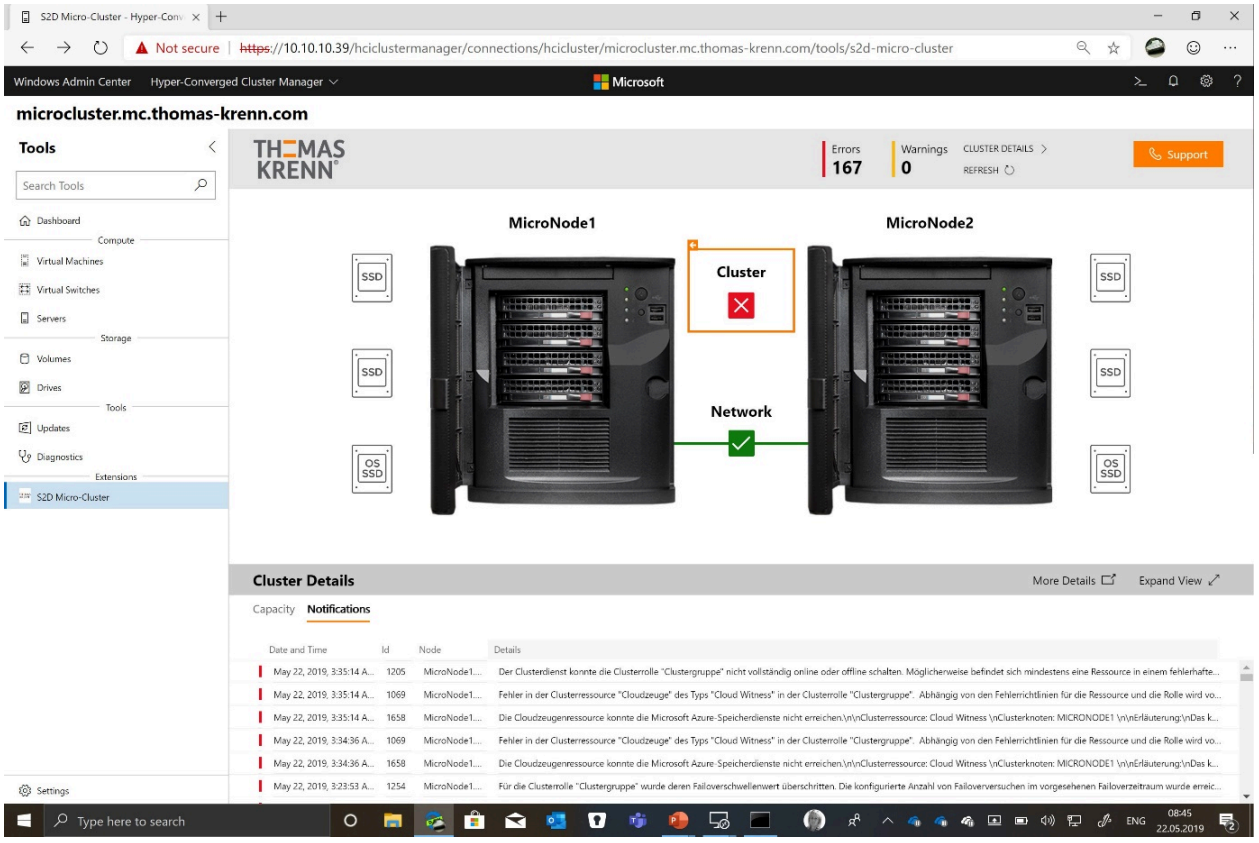
You can drill down into storage devices to view serial numbers, SMART-status, and capacity utilization. Boot devices also show wear out indicators, reallocated sectors and power on time, which are the best indicators of SSD health.



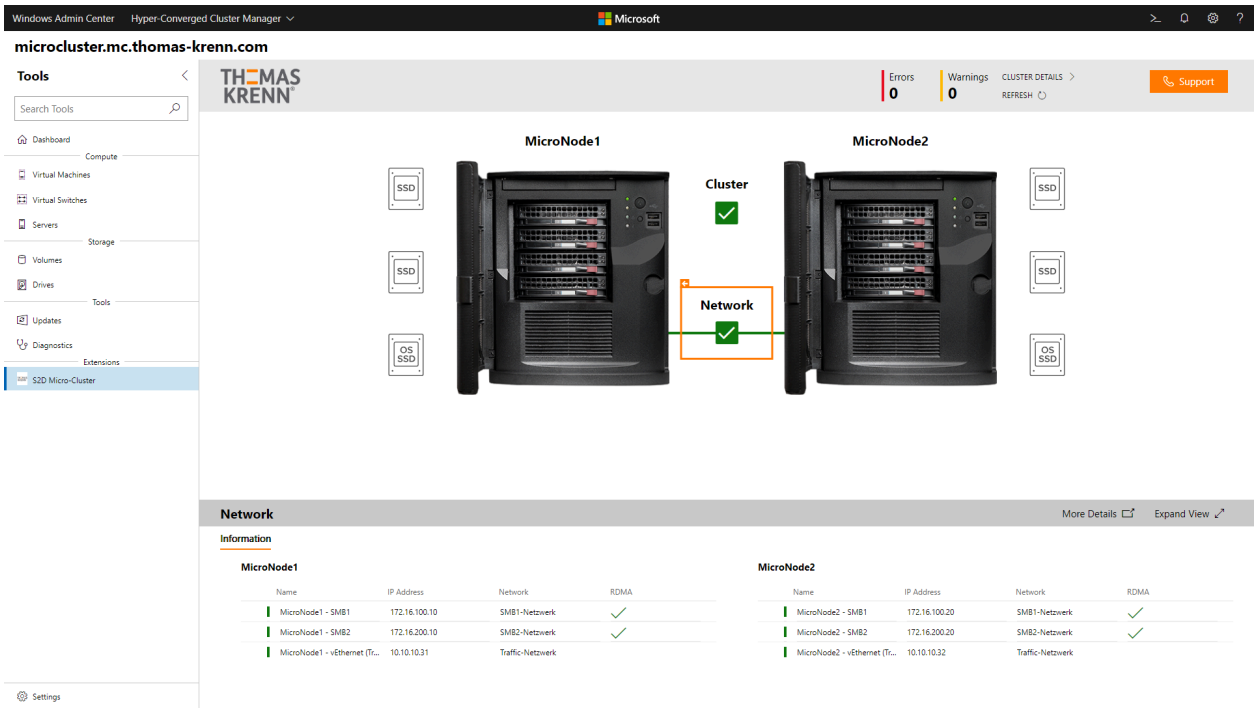
The cluster status icon expands to show a summary of the cluster's operational details.



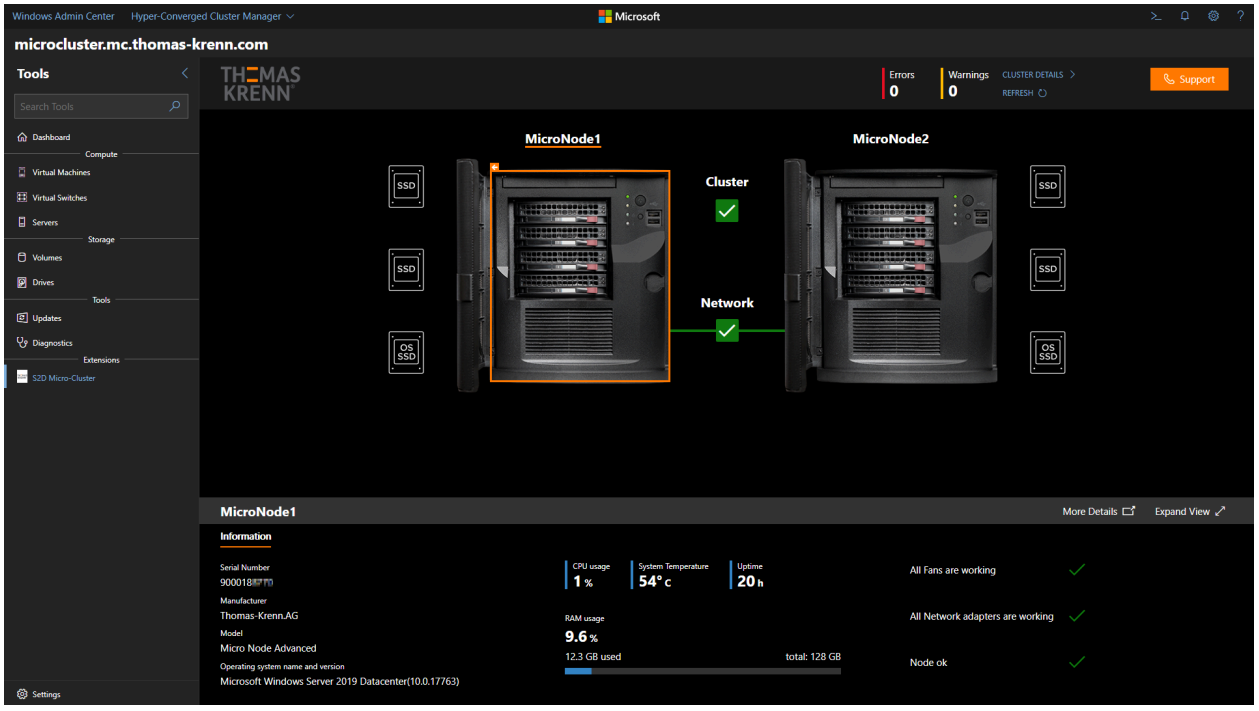
After this Micro-Cluster's Azure cloud witness was unavailable for a whole night, one glance is enough to identify the problem. Clicking on “Notifications” immediately lists relevant events for quick remediation. Cluster events are localized and determined by the base OS language. The extension itself supports English and German.



Network information is readily available as well.



Based on customer feedback, we've also implemented “Dark Mode” available in Windows Admin Center v1904. This is soothing in dark datacenters and in poorly lighted cabinets and closets. It also makes Windows Admin Center more accessible by reducing glare for admins with certain visual impairments.



Thomas-Krenn immediately realized that usability and accessibility for untrained admins would be key to a great customer experience for hyper-converged infrastructure in the small and mid-sized business market. Thomas-Krenn's Micro-Cluster extension perfectly complements Windows Admin Center's native HCI management capabilities by including proprietary hardware information on the dashboard and re-grouping important cluster health information in a new, human-friendly interface.

During the development process it was decided to deploy Windows Admin Center 1904 in a high-availability configuration on the cluster itself, ensuring manageability even after node failures. The extension comes pre-installed, just as the entire OS.

The extension was built in parallel with Windows Admin Center 1904 being developed at Microsoft. Close cooperation and continuous feedback exposed issues on both sides that were jointly resolved before the product successfully launched in April 2019. Thomas-Krenn is incredibly proud to be one of the first to fully support and implement Windows Admin Center 1904's new features.

Feedback

Was this page helpful? Yes No

Provide product feedback