HOW TO GET THE MOST OUT OF
WINDOWS ADMIN CENTER

SECOND EDITION BY ERIC SIRON

MICROSOFT CLOUD & DATACENTER MANAGEMENT MVP

HORNETSECURITY

ALTARO

DOJO

HORNETSECURITY

Hornetsecurity is a leading email cloud security and backup provider, which secures the digital communication, business continuity, compliance, data and IT infrastructure of companies and organizations of all sizes.

Its award-winning product portfolio covers all important areas of email security, including spam and virus filters, legally compliant archiving and encryption, and protection against CEO fraud and ransomware; as well as backup, replication and recovery.

Its flagship product is the most extensive cloud security solution for Microsoft 365.

BUSINESS 365

TOTAL PROTECTION

With 365 Total Protection Suite , the seamlessly integrated security and compliance suite for Microsoft 365, Hornetsecurity delivers a comprehensive security package specifically for Microsoft 365 customers to protect their email communications and data in the cloud from the latest cyber threats.

FREE TRIAL

Office 365 Backup

Altaro Office 365 Backup enables you to back up and restore all your Microsoft/Office 365 mailboxes and files stored in OneDrive and SharePoint through an online console, on an annual subscription, allowing you to easily manage your backups. Data is backed up to Altaro's Microsoft Azure infrastructure. 24/7 support included.

FREE TRIAL

HOW TO GET THE MOST OUT OF WINDOWS ADMIN CENTER

2nd Edition by Eric Siron

PUBLISHED BY HORNETSECURITY / ALTARO

https://www.hornetsecurity.com/en/

Production: Neal Storan

If you have any feedback about this book, its content, questions for the author or any other feedback, please write to dojo@altaro.com

TABLE OF CONTENTS

INTRODUCTION

In 2018, Microsoft introduced an all-new server management experience with Windows Admin Center (WAC). This ambitious project modernizes administrative activities with a centralized HTML 5 web application. Add your servers, clusters, desktops, and Azure virtual machines into a personalized, persistent interface, and manage their roles, features, software, registry, PKI certificates, and more!

Even though that reads like an advertisement, you don't have to spend any money to acquire or operate Windows Admin Center. The ease, availability, and power of Windows Admin Center has earned the respect of systems administrators everywhere. If you have not tried it for yourself, this book will take you on a tour of its features and power. Look through all that WAC offers and see how it can simplify your server management tasks.

ABOUT THE AUTHOR

ERIC SIRON

Eric is a four-time awardee of the Microsoft Most Valuable Professional award in Cloud and Datacenter Management.

He has worked in IT since 1998, designing, deploying, and maintaining server, desktop, network, storage, and backup systems.

Eric has provided all levels of support for businesses ranging from single user through enterprises with thousands of seats and developed a particular affinity for small organizations.

He has achieved numerous Microsoft certifications and was a Microsoft Certified Trainer for four years. Eric is also a seasoned technology blogger and has amassed a significant following through his top-class work on the Altaro DOJO.

SECOND EDITION UPDATE NOTES

Much has changed in Windows Admin Center since the first edition of this book. Microsoft has invested heavily in the program, building new features, and incorporating user requests. If you haven't kept up with the changes, here's a quick overview of some of the important developments:

- Windows Admin Center and its extensions update automatically
- Control Azure Stack HCI
- Packet monitoring with the ability to export captures for Wireshark
- Control several Azure components and services
- Can use WinRM over HTTPS
- Management of software defined networking (SDN)
- Integration with Azure Monitor, including the ability to send e-mail notifications

In the past, Microsoft collected suggestions and bug reports from administrators via UserVoice. They have since ended the use of this channel. You can now submit both communication types through the **Feedback** link in Windows Admin Center.

Currently, this link takes you to a simple form hosting by Microsoft. Microsoft values the shift away from UserVoice as it eliminates third party involvement in the process, but users no longer have any way to view or vote on input from others. We have lost control over prioritization and have no transparency regarding the choices that Microsoft makes.

WHY USE WINDOWS ADMIN CENTER?

Windows Admin Center's greatest strengths appear in its ease of use and powerful management capabilities. For smaller environments, it brings the bulk of daily server management tasks into one place. For larger organizations, it coalesces multiple hosts and their personalities into a single list. For people newly entering systems administration accustomed to modern practices with end-user devices and computers, such a thing seems like a fundamental, expected necessity. However, things did not evolve that way. To fully understand the value proposition of Windows Admin Center and the portions that still seem inadequate, it helps to know about its predecessors.

A BRIEF HISTORY OF MICROSOFT MANAGEMENT TOOLS

In the DOS days, PCs didn't need much in the way of management. You can find cheap novelty electronics with more computing power and memory than the highest end systems from the 1980s. They had no concept of networking or multiple users. Most didn't even have built-in permanent storage. Back then, even networked systems and their operating systems had little centralized management. Almost everything was done with individual narrowly focused utilities.

As systems evolved and expanded, management became more challenging. Tools appeared out of necessity. For those of us in PC world, our first experience with anything resembling a centralized system was "Control Panel" in some version of Windows. For the bulk of its history, it was still, really, a collection of individual utilities, but they all had a common home. "Administrative Tools" debuted later, bringing more ways to control Windows, especially as a network citizen.

As Windows became more complex and ubiquitous, administrators needed better ways to stay on top of them. Throughout the years, Microsoft has tried several ways to make management of its operating systems and domains easier, with varying results. Their free and in-box offerings have changed radically through the years, culminating in Windows Admin Center.

MICROSOFT MANAGEMENT CONSOLE

Frankly speaking, the management experience for Windows Server and its roles and features was never one of Microsoft's strengths. The Microsoft Management Console (MMC) was one of Microsoft's earliest attempts to bring some measure of unification to managing its ecosystem. Most built-in Windows components have an MMC (Microsoft Management Console) interface. Some other teams within Microsoft adopted MMC for their projects (such as Exchange Server and SQL Server). Microsoft provides a software development kit (SDK) for MMC that allows anyone to create MMC "snap-ins".

MMC itself only provides a minimal programmable interface that essentially acts as a Windows Forms host. Many developers found it too restrictive, especially as interfaces modernized. Even Microsoft seems to have mostly abandoned the platform. MMC snap-ins rarely appear in non-core products and Microsoft has not updated the MMC SDK since 2016. MMC has multiple drawbacks:

- Difficult discovery of available consoles
- No recommended or enforced interface or management standards, causing inconsistencies in visual design and control behavior across snap-ins
- Low adoption rate outside of Microsoft
- Poor update practices within Microsoft

- Gradual migration away from MMC for premium services and features
- Snap-ins easily install locally with matching roles, but requires special installation and separate update cycle to use remotely
- Difficult to migrate and share snap-in customizations

Many snap-ins had their own problems. Most commonly, a single version of a snap-in usually could not manage different versions of the target application or role. Installing multiple versions of the same snap-in on one system takes considerable effort. For the snap-ins that only ship as an operating system component, migration to another system for remote management effectively violates the operating system license agreement.

All these problems resulted in mismatched, inconsistent, confusing, and sometimes unpredictable management practices. Worse, the compatibility and discoverability barriers of MMC made management tasks easier from a server's local console, leading to widespread use of inherently insecure remote desktop connection usage. Third parties provided tools to fill the gap, as did Microsoft with its System Center products, but these tools mostly addressed problems of scale and carried a price tag that prohibited use among smaller customers.

SERVER MANAGER

Server Manager was not developed as a replacement for MMC snap-ins. In fact, its earliest release was itself an MMC snap-in. However, Server Manager did try to smooth the administrative experience by placing common tools and

processes in a single place that greeted administrators at startup. For better or for worse, most administrators responded by telling it to go away and never return. Earlier tools still worked perfectly well, and experienced administrators felt like Server Manager was an almost childish approach. Their influence on younger administrators had a slowing effect on the acceptance of the tool. As for its merits, it offered several convenient launch points for configuring new servers but afterward, it turned itself into a collection of MMCs.

Regardless of Server Manager's problems, the basic concept had merit: manage everything from one place. Microsoft continued to evolve the tool. In a significant step, Server Manager broke free of its MMC origins and became a separate toolbox. It improved upon MMC in many ways:

- Modern interface
- Simple multi-server control
- Easier discovery of available consoles
- Broader installation and control options
- Recommended interface and ability guidelines, such as Best Practices Analyzers

Despite these advances, Server Manager inherited one major problem from its snap-in ancestor: the inclination of administrators to tell it to go away and never come back. It also performs substantially slower than most MMC snap-ins, including the original Server Manager.

Some teams at Microsoft embraced Server Manager, making it the only way to graphically manage some of their projects or features. As examples, some parts of Remote Desktop Services and server file sharing services have no graphical alternatives. Furthermore, Microsoft publishes no guidance for aspiring third party plug-in developers, effectively making Server Manager into a Microsoft-only platform. Despite the handful of components with a strong presence in Server Manager, most range from minimal to average usefulness.

Server Manager will almost certainly survive well into the future, but it will not likely see meaningful advances. For one reason or another, it did not garner sufficient mass appeal before the introduction of superior alternatives for Microsoft to commit high levels of resources.

POWERSHELL

So far, we've only considered graphical interfaces. However, any proper discussion of the evolution of Microsoft's management tools must include a section on PowerShell. As its name makes obvious, PowerShell is a "shell": the place where the person and the machine make contact. Almost any other description ignores something. You will likely recognize it best as a character-mode interactive "console" application, often called a "command-line interface" (CLI).

PowerShell fills many roles in systems administration while fixing the problems of earlier shells:

- **Common, recommended interface guidelines:** You can reasonably expect commands in the format of accepted verb, hyphen, singular noun.

- **Modularity:** The shell on its own provides basic operating system interactions. You can add modules that grant new command and control powers.
- **Discoverability:** PowerShell exposes a list of all installed and available modules, allows you to learn what commands they provide, and its built-in help system can show the syntax even for modules that have no documentation.
- **Remote capabilities:** PowerShell offers remote control via "implicit" and "explicit" remoting.
- **Security:** PowerShell Remoting always uses an encrypted channel, which you can strengthen with certificates. Note: PowerShell's ability to operate binary modules, arbitrary scripts, and executables means that it does not always control communications.
- **Compatibility:** PowerShell can communicate with different versions of PowerShell on remote systems. Many modules can also interact with other versions of themselves, but explicit remoting can handle version mismatches and the absence of a locally installed module.
- **Programmability:** Anyone can create a PowerShell script and form it into a PowerShell command (known as a cmdlet) or use any development environment that can produce Windows DLL files to create binary PowerShell modules.

- **Automatability:** Anyone can create a script much like traditional batch files and bash scripts that gather PowerShell cmdlets, operating system commands, and application invocations to perform routine or complex tasks.
- **Range:** If the computer can do it at all, you have some way to use PowerShell to make it happen. Many things involve work more suited to other tools, but the capability exists in PowerShell.
- **Cross-platform:** PowerShell began life as "Windows PowerShell" and only worked on Microsoft operating systems, but it has since evolved into "PowerShell" and operates on most Unix and Linux platforms.

PowerShell has few negative aspects worth talking about in this context. First, a universal concern: inconsistencies between modules. Microsoft and the PowerShell community have done an outstanding job using documentation, examples, and pressure to establish common practices and expectations for PowerShell module designers. Unfortunately, nothing forces anyone to adhere to those standards.

Second, a controversial "drawback": PowerShell has no graphical component. You can use it to access the Windows Forms DLLs and other GUI frameworks to create graphical interfaces, but that requires at least the same level of effort as the same task in development-oriented tools and languages. The lack of a native graphical interface appears here because CLIs intimidate many administrators, especially those with little experience. A few sysadmins refuse to use non-graphical interfaces, often finding them inherently inferior. Even experienced administrators may feel lost when encountering a module for the first time.

Regardless of attitude or belief, GUIs and CLIs complement each other, and both will stay with us forever. We need PowerShell and all administrators should come to terms with it, but we also need solid graphical solutions. Enter Windows Admin Center.

WINDOWS ADMIN CENTER

Windows Admin Center (WAC) takes the fundamental purpose of MMC and applies the same sort of solutions presented by PowerShell (in fact, WAC makes extensive use of PowerShell behind the scenes). It solves problems left over from earlier tools and opens future possibilities.

Characteristics and capabilities of Windows Admin Center:

- Modern HTML 5 interface
- Central management
    - Install on your desktop operating system and remotely control systems.
    - Install on a "gateway" server that allows anyone that can reach its web interface to remotely control systems.
- Control servers, desktops, clusters, and Azure virtual machines
- High availability mode for gateway resiliency
- Modular design with a public API that allows anyone to add tools (called extensions)

- Compatibility: WAC extensions define their requirements, essentially making WAC able to control anything if someone can build the plug-in
- Cross-platform: WAC's extensibility also gives it cross-platform control. WAC itself can only install on Windows systems.
- Security: WAC integrates with Active Directory and Windows authentication. It also requires an SSL certificate on its web interface and encrypts all data that crosses web connections. It encourages management of multiple systems through a single keyhole system rather than individual remote desktop sessions.

You can see a lot of overlap between the advancements in PowerShell and the features in MMC and Server Manager. You won't find many weaknesses. WAC incorporates an in-browser PowerShell client, so it has all PowerShell's capabilities.

WAC has power, but it also has imperfections. In the next section, we'll go over the pros and cons of WAC against alternatives. Later chapters tackle WAC's two biggest frustration points: installation and resource connections. Toward the end of the book, we'll talk about some of WAC's shortcomings that Microsoft could address in future releases.

COMPARING WAC TO ALTERNATIVES

Everyone has their own processes and preferences, so you might decide that WAC does not solve problems as well as tools that you already use. You won't get a better idea from reading than from installing WAC and experiencing it yourself. Treat the contents of this section as a preview.

WAC VS MMC

Among Microsoft's native and free offerings, MMC tools provide the most common graphical management experience. Many administrators use MMC-hosted tools long before they know anything about MMC.

As the star of the show, we'll start with WAC's cons over MMC:

- **Overview:** WAC provides a dashboard view, giving you an overall idea of a system's status and health at a single glance.
- **Single management point:** A single installation and instance of Windows Admin Center can manage all your Windows, Windows Server, Failover Clusters, and more. You do not need to deal with individual component downloads, Remote Server Administration Tool configuration, or Remote Desktop connections.

- **No special client requirements:** any modern HTML 5-compliant browser can access Windows Admin Center (tested and supported in Google Chrome and recent versions of Microsoft Edge that use the Chromium engine).
- **Permanent, self-curated system list:** You decide which systems appear in your WAC console. Once added, a system remains until you remove it. Your list belongs only to you. Other administrators configure their own lists.
- **Simple firewall rules:** You only need to open the WSMan port (TCP/5985) between the WAC system and its targets. MMC does not dictate any particular network constraint for its snap-ins, so each tool has its own rules.
- **Authentication and encryption:** You access the WAC server through an HTTPS channel encrypted with a PKI certificate. It recognizes you by your credentials and passes them on to the destination system. WSMan encrypts its communications from the WAC host to the target.
- **Ad hoc management:** Windows Admin Center includes an in-browser PowerShell console so you can directly execute commands when WAC does not provide a suitable graphical interface.
- **Future development:** Microsoft's WAC team constantly works to improve and expand its usability, features, and capability. Most of the MMC tools have not significantly changed over the years.
- **Extensibility:** Leverage the Windows Admin Center SDK to build interfaces of your own. Specialize them for your organization and keep them internal or generalize them and make them available to others.

Microsoft has placed a great deal of importance on Windows Admin Center. As a result, they assign it significant resources and talent. They pride themselves on adding features that users request. Expect it to advance and evolve rapidly. However, don't start removing your MMC shortcuts just yet. MMC still holds a few advantages:

- **Extent of control:** WAC has not yet reached feature parity with the multitude of MMC snap-ins. You will find a great deal of variance among individual tools; some have near parity, some lag far behind, a few have exceeded their predecessors.
- **No local control:** WAC can control the system that you install it on, but has no presence on any other system. If you need to work at a system's console and cannot access WAC or WAC cannot access the target host, you will need to use traditional tools.
- **No infrastructure work:** WAC can store and pass your credentials in a single session, but works best if you enable delegation in Active Directory. MMC runs under your account and some snap-ins can prompt for credentials when necessary.
- **Customizable consoles:** You can use MMC to build out your own selection of snap-ins and save the console configuration for later use. You can also use custom consoles for staff with specifically delegated permissions.

- **Familiarity:** You will adapt to WAC quickly. However, do not underestimate the value of comfort that you and your staff have with existing tools. Even in places where WAC dramatically outmatches its MMC counterpart, nothing replaces experience and practice quickly.

For now, WAC will augment your MMC tools, not replace them. Over time, expect that to change.

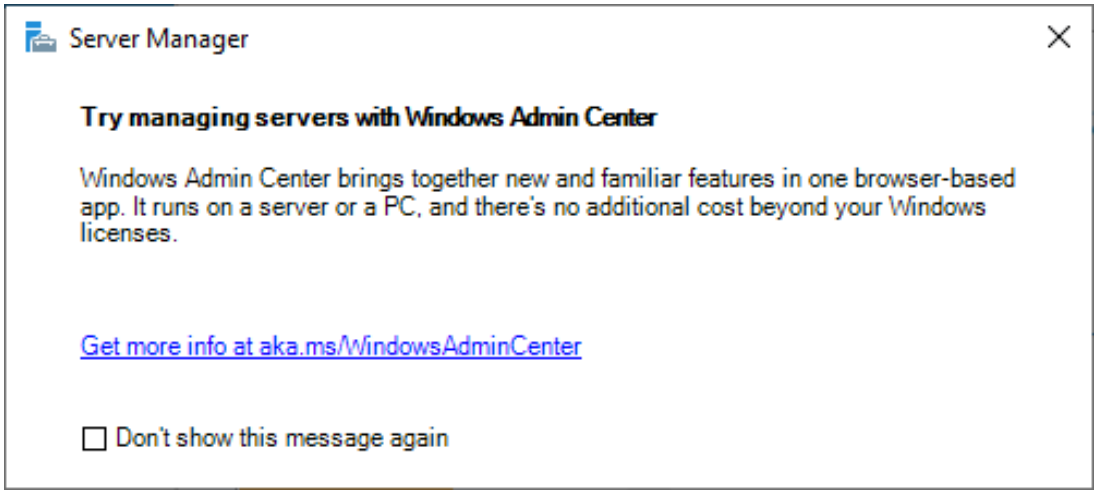
WAC VS SERVER MANAGER

If you have gotten comfortable with Server Manager, then you'll adapt to Windows Admin Center quickly. Like Server Manager, Windows Admin Center uses a left-side menu bar to provide access to its features. Also like Server Manager, Windows Admin Center allows you to store connections to multiple servers in one console.

However, Windows Admin Center exceeds Server Manager in several vital ways:

- WAC only uses WSMan on port 5985 for communications
- WAC allows per-host credentials
- When installed on a desktop operating system, Server Manager sometimes exhibits odd behavior when managing multiple servers. WAC has no particular limits on management connections.
- Server Manager rarely receives updates outside of major operating system releases
- Microsoft publishes an API for WAC, encouraging third-party add-ins

Microsoft has effectively recognized the limited usefulness of Server Manager by embedding a promotion for Windows Admin Center. When you start Server Manager on recent Windows/Windows Server instances, it produces the following dialog:



While you should not expect Server Manager to disappear any time soon, you also should not expect Microsoft to give it any attention going forward.

> **Note:** Windows Admin Center is fully compatible with Windows Server 2022. Learn more about Windows Server 2022 Features.

WAC VS SYSTEM CENTER

Unlike MMC, Microsoft has no (stated) intention of phasing out System Center. If that ever happens, they will likely use a premium Azure-based tool. Windows Admin Center will continue as the free and simple tool set.

Because Microsoft does not position WAC and System Center against each other, a pro/con list would depend on what matters to the person creating it. Instead, we'll list out differences between the products:

- WAC does not require an agent for most of its functionality. A handful of plug-ins and optional features do have agents, but WAC itself needs no software beyond what its targets already have. System Center requires agents to manage systems, sometimes more than one per target.
- System Center's dashboards and consoles allow you to view multiple systems simultaneously. At this time, Windows Admin Center focuses on one system at a time.
- System Center was built to manage at datacenter scale. WAC's design and functional philosophy match more closely with what you find in MMC-hosted tools.
- System Center goes beyond management with components that handle deployment, backup, and more. It even has understanding of some applications that involve multiple hosts.
- System Center's agents require more overhead and management effort but grant it abilities that exceed what you can expect from WAC.
- System Center needs significant resources to operate, including at least one dedicated server and a SQL instance.

WAC and System Center can coexist peacefully. WAC has system management functions that System Center does not. System Center has operational uses that WAC cannot match. System Center views your environment holistically. WAC operates at a more individual level.

WAC VS THIRD PARTY TOOLS

With the quantity and breadth of tools offered by other vendors and the community, no simple list could possibly make useful comparisons. Instead, install WAC and see how it compares to anything that you have in operation or under consideration today.

This may not wind up as a "versus" situation anyway. Due to WAC's extensibility, built-in connectivity, and authentication powers, third party vendors have begun shipping add-ins. Over time, more will follow. Some of the third-party tools that you use today might migrate into WAC.

Of course, competing management tools won't have the same parallel objectives. If you have one that you like, then hopefully WAC's debut has spurred their creators to improve their offerings. If not, then maybe you can replace it with WAC.

INTEGRATING WAC INTO AN EXISTING ENVIRONMENT

You can easily bring WAC in alongside anything else you have. Its agentless nature means that it can manage and monitor systems without colliding against other tools. Therefore, you have little risk in trying it out. Whether you find it superior, inferior, or complementary to others, you have the time that you need.

PRICING AND LICENSING WINDOWS ADMIN CENTER

Windows Admin Center only installs on Windows and Windows Server (more details in the Installing Windows Admin Center chapter). If a legal license covers that operating system instance, it also covers WAC in both licensing and cost. Essentially, Windows Admin Center exists as an adjunct component of Microsoft operating systems.

In simpler terms, if you have a legal instance of Windows or Windows Server to run WAC on, you do not need to purchase or sign anything else.

INSTALLATION METHODS

Typically, a technical book would include an install guide immediately after product introduction. In the case of WAC, we must first talk about its two installation methods. While they can coexist (on different computers), one mode makes the other largely pointless.

DESKTOP MODE

You can install WAC right onto your personal Windows 10 instance. When the installer detects Windows 10, it will establish “desktop” mode automatically. It includes a web interface, but Microsoft does not support accessing it from any other computer. The web server component only runs for as long as you operate a connected client. As soon as you close your session, WAC in desktop mode shuts down.

WAC in desktop mode thrives under three conditions:

- Few administrators
- No suitable Windows Server instance to host WAC
- No/little Active Directory presence

Essentially, if you don't need WAC for management at scale, then a desktop-based deployment can satisfy your needs and fit within your budget. It may also work best in remote and poorly connected branch offices or client sites with distinct authentication domains.

Desktop mode suits two edge cases particularly well: sites that restrict multi-hop connections and sites with a significant number of disparate credentials required for accessing different hosts (e.g., many workgroup members). The security chapter explores the reasons in more detail.

Just like any other program, a WAC installation means another point that you must document, maintain, and secure. Microsoft frequently updates WAC. Extensions also require updates. Windows Update will handle the primary program, but you must handle extensions yourself. If an extension with a security vulnerability exists on multiple desktop WAC installations, then you might find yourself in an emergency.

Desktop WAC does not have the credential headaches of gateway WAC, which gives it an edge. Unfortunately, it doesn't have the same stability as gateway WAC either, so it causes different headaches. Prepare yourself for WAC periodically refusing to allow your browser to connect, forcing you to end your browser session and start over.

GATEWAY MODE

When WAC's installer detects a Windows Server operating system, it installs in "gateway" mode. It will try to bind to the default HTTPS port of 443, although you can override that. Unlike desktop mode, the gateway runs full time as a service and makes its endpoint accessible on the network.

WAC in gateway mode works well in these conditions:

- Many administrators
- No assigned Windows desktop instances, such as in pooled VDI environments
- Desire or required high availability for system management tools
- Firewall rules prevent direct access from desktops to server systems

You can fairly say that a WAC gateway works in environments that make WAC desktop untenable. Gateway mode avoids dependencies on specific desktops and places the bulk of the management plane into the datacenter. Desktop operating systems perform all command-and-control actions via an authenticated and secured connection to the WAC gateway.

You can continue installing WAC in gateway mode on additional systems. All instances can operate independently or, if their hosts belong to a Windows Server Failover Cluster, you can configure them to use a single access point in high availability mode.

HIGH AVAILABILITY GATEWAY MODE

As an additional facet of the WAC gateway "personality", you can configure it for high availability. Before going this route, think through the benefits and drawbacks. As a central management tool, it might feel like a natural imperative to make it resilient. However, you can achieve a similar outcome with significantly less effort.

To make WAC highly available, you must install it on a cluster with shared storage. That has the following ramifications:

- Additional configuration and maintenance load for administrators
- Additional potential failure problems
- Additional resource requirements
- The port 80 redirect does not function

The first three should not surprise to anyone that has ever operated a Microsoft Failover Cluster. However, we typically build them to protect resources that cannot survive the failure of an operating system instance. Windows Admin Center does not really have that problem. Yes, a cluster provides it with a single access point and automatically protected system lists and saved credentials. But, it provides nothing else.

As for the failure of port 80 to work, that currently has no explanation. Typical ways of trying to overcome that, such as by modifying the script before installation or changing registry keys afterward, usually cause WAC to break entirely.

As one alternative to using WAC in high availability mode, you could build a single WAC instance inside a highly available virtual machine. In that configuration, its primary threat comes from failures of the instance. If that happens due to a catastrophic cluster failure, then an HA WAC build would fare no better. If it happens due to a failure inside the guest instance, then you can restore from backup. Otherwise, Windows Updates will cause the most service interruptions.

As a secondary alternative, you could build two separate WAC servers on two separate Windows Server instances, treat them as distinct systems (DNS names, IP, etc.), and have administrators manually sync lists between them. The worst case would occur if your administrators had many separate credential sets to duplicate across the systems. Optionally, you can build deploy and configure WAC in a virtual machine and clone it to another. Cloned operating systems co-existing presents problems, but you could deal with them. You could have make the system separation permanent, or you could mark one as “primary” and periodically clone it to the other.

Failover of the WAC role does not function instantly. Even after Failover Cluster Manager indicates a successful ownership change, WAC will not respond for a few seconds. To head off complaints, notify administrators in advance of any host or cluster maintenance.

MIXING AND MIGRATING BETWEEN DESKTOP AND GATEWAY MODE

Usually, a comparison of the requirements helps you to quickly decide whether to use WAC in desktop or gateway mode. They can co-exist. WAC does not know or care about other WAC installations, and, barring some condition of an extension, they will not collide. However, when you have a gateway installation available, it typically makes any desktop installations useless. That won't apply in every case; you might have servers for multiple sites operating on a laptop that travels with you.

Regardless of your initial decision, changes, or situations, you can freely migrate your managed system lists between WAC installations. Saved credentials do not move, so that places some limits on portability. Aside from that, you do not need to feel trapped by a desktop installation if your network grows to make a gateway the better choice.

INSTALLING WINDOWS ADMIN CENTER

Windows Admin Center presents a few challenges during installation and configuration. You have the option to start with the installer and deal with challenges as they arise. This chapter works through things that you can do in advance to smooth the experience and then walks through installation steps.

SYSTEM REQUIREMENTS

Microsoft does not publish any hardware requirements for Windows Admin Center. If the system you want to install it on can run a current version of Windows or Windows Server, then it should handle WAC. WAC does have operating system requirements.

Any supported desktop version of Windows 10 or later will run Windows Admin Center in desktop mode. Check Microsoft's Windows lifecycle page for information on individual releases. The original release of Windows Admin Center worked on Windows 10 1709, but 1709 has since reached end-of-life and Microsoft has updated WAC several times since then. WAC may function on earlier releases, but you have no guarantees.

For Windows Admin Center in gateway mode, you must provide a Windows Server instance running 2016 or later. That includes installations in Core mode and any currently supported Semi-Annual Channel release.

Essentially, use relatively current hardware or a virtual machine running on such a system and a supported version of Windows or Windows Server. The typical WAC installation does not process large quantities of data all at once, even in large environments.

ENVIRONMENTAL REQUIREMENTS FOR WINDOWS ADMIN CENTER

Your Windows Admin Center system must make administrative connections to other systems in your environment. In small, simple environments, you won't need much to make that happen. Larger or stricter environments will need some preparation to ensure a smooth experience. This involves three components:

- A PKI certificate
- Constrained delegation (Windows domain)
- TrustedHosts configuration (workgroup hosts)

The rest of this section examines these items briefly. Later parts of the book will revisit them in depth.

To properly secure the Windows Admin Center interface, you need to install a PKI certificate (commonly known as an SSL certificate). For desktop mode, Windows 10 will automatically create a self-signed certificate with a ten year validity period. Because desktop mode will not accept remote connections, that will provide sufficient security. In gateway mode, the WAC installer will offer

a self-signed certificate with a 60-day expiration. Self-signed certificates do not provide satisfactory host-to-host security, so plan to use an authorized certificate.

Constrained delegation allows Windows hosts to share operator credentials with downstream Windows hosts. Kerberos owns the delegation feature, so this only works in domain environments.

When you don't have a domain, you work with systems that live outside of your domain, or you must use local credentials, you will need to employ TrustedHosts. TrustedHosts represents a potentially significant security problem, but it only applies to the host that runs Windows Admin Center. You should tightly control and monitor the security stance of any system that has the management powers of Windows Admin Center anyway, so you can manage this risk.

If you know how to configure these, you can just use the above as a checklist and move on. Otherwise, you can continue through this chapter to install your Windows Admin Center instance and handle them as you work through later chapters.

REQUIREMENTS FOR HIGHLY AVAILABLE WINDOWS ADMIN CENTER

Before committing to a highly available WAC installation, make sure to read over the considerations from the previous chapter. Remember that environmental problems can cause problems for clusters – the very sorts of problems that you might want to investigate with Windows Admin Center.

To operate WAC in high availability mode, you must have:

- A functioning Microsoft Failover Cluster with at least two nodes
- According to Microsoft, the cluster must have a Cluster Shared Volume with 10GB available capacity.
- A DNS name to use for access.
- If you do not provide an IP address, it will set DHCP on the named access point. If you believe that your infrastructure can satisfactorily maintain DHCP and DNS through any condition where you would require Windows Admin Center, then you can use dynamic addressing.
  A static IP would reduce your dependencies.
- The latest Windows Admin Center MSI file
- The configuration script from: https://aka.ms/WACHAScript

The script does not receive as many updates as the primary Windows Admin Center product so do not worry if it has a lower version number. Always use the link above when creating a new WAC cluster to ensure that you have the latest version.

Nothing will stop you from modifying the script as you see fit. Microsoft may not support your final installation. However, Microsoft has not published a formal support stance for Windows Admin Center anyway. If you rely on the community for assistance, understand that anyone that did not make the same modifications might not have the ability to help you with problems. Realistically, if Windows

Admin Center successfully installs and you have no problems connecting to its interface and managing systems, then you have little to worry about. At the same time, you have no guarantees that a future WAC update won't break your modification.

HOW TO ACQUIRE WINDOWS ADMIN CENTER

Microsoft provides Windows Admin Center as a download. They deliver it through their Evaluation Center, even though it has no license expiration. For the most reliable long-term link, start at https://aka.ms/WindowsAdminCenter. This takes you to the product landing page which will contain the latest download link.

Once you have downloaded the install file, place it in a location accessible by the system(s) that will operate Windows Admin Center. Before proceeding to the applicable installation directions, review the environmental requirements.

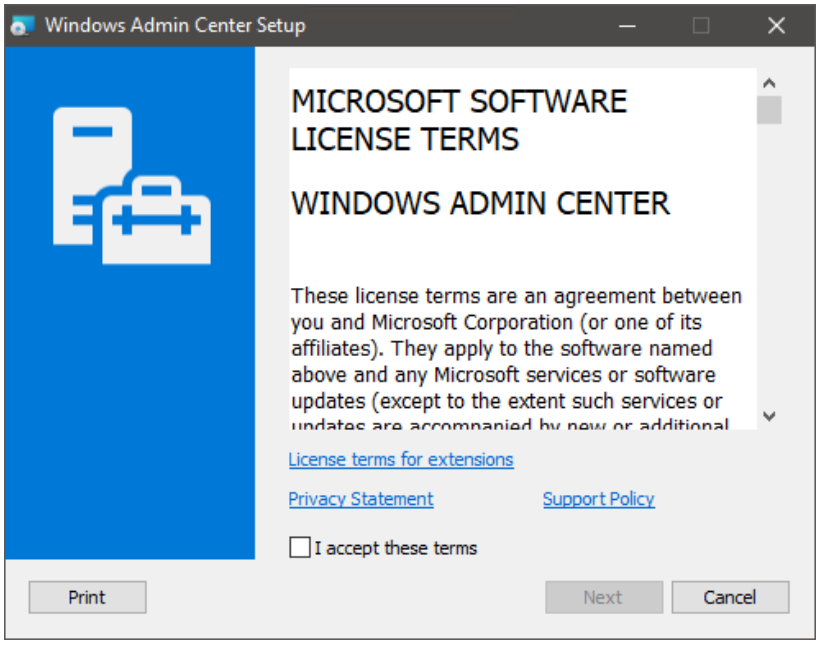
HOW TO INSTALL IN DESKTOP MODE

You only need to run the installation MSI on a host running a desktop edition of Windows from Windows 10 1709 onward. It will detect the presence of a desktop operating system and automatically select desktop mode. The install interface and switches do not provide a way to override WAC to gateway mode.

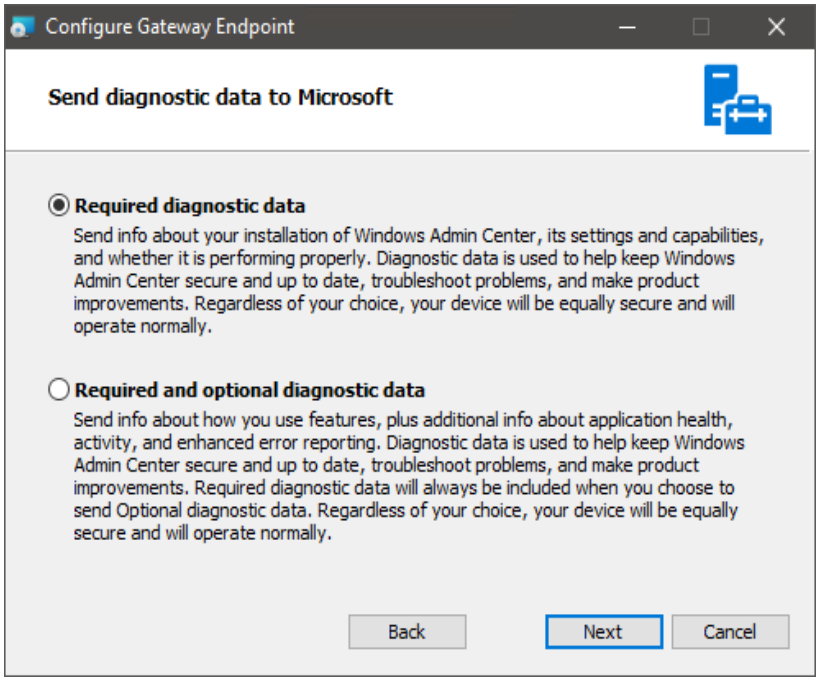
The installer has only a few dialogs:

1. When you start the installer, it shows you the license agreement.

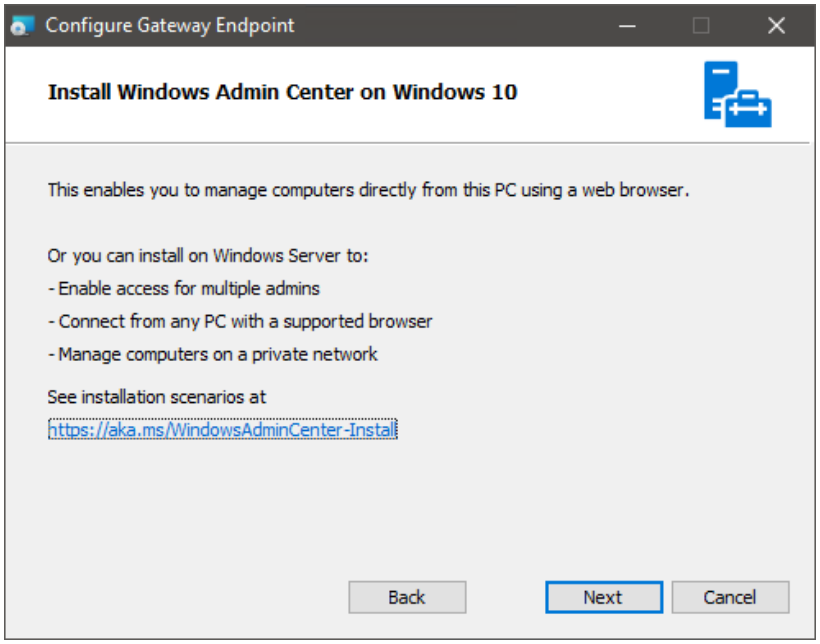
   You must check its box to accept before you can click **Next** to move on.



2. On the second screen, choose the amount of telemetry data to send to Microsoft. Microsoft aggregates information from optional data to understand feature popularity and usage. Make your selection and click **Next**.



3. The third screen exists only to explain that you could have gotten to gateway mode by installing on a server. It provides a link so that you can read Microsoft's description of the different installation modes. Click **Next**.



4. The fourth screen has multiple selectable options.

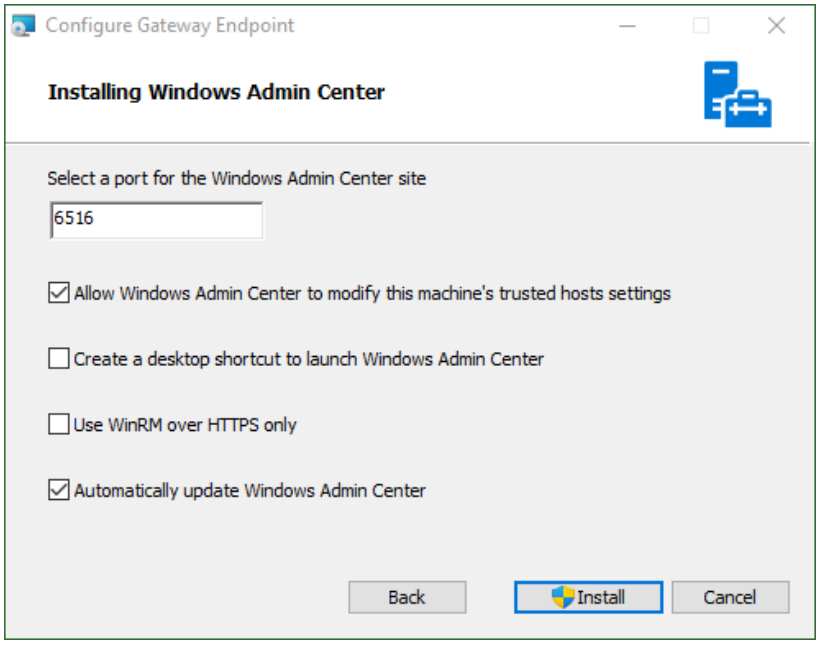
    1. By default, WAC operates on port 6516. Only change it if another application already uses that port.

    2. You can allow Windows Admin Center to manage the TrustedHosts settings for you. By "manage", it means that it will automatically trust every host.

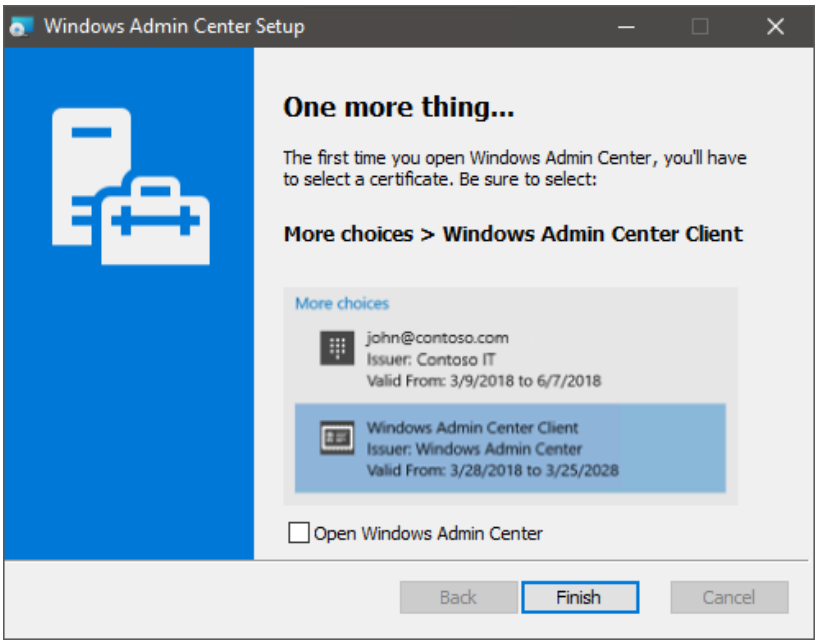
    3. The installer always create a shortcut on the Start menu. You can optionally have it create one on the desktop as well.

4. By default, WinRM (used by PowerShell Remoting, therefore by Windows Admin Center) operates over port 5595 in HTTP. You can opt to use HTTPS, which enforces fully encrypted communications. We will explain this the Security chapter.

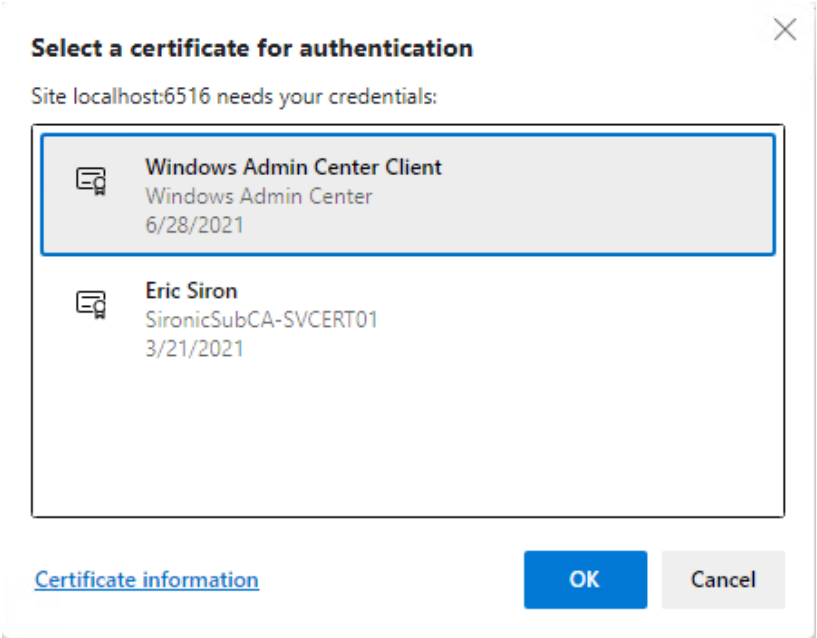
5. You can opt-in to have Windows Admin Center update itself using the same update tools as the rest of Windows and Microsoft software.



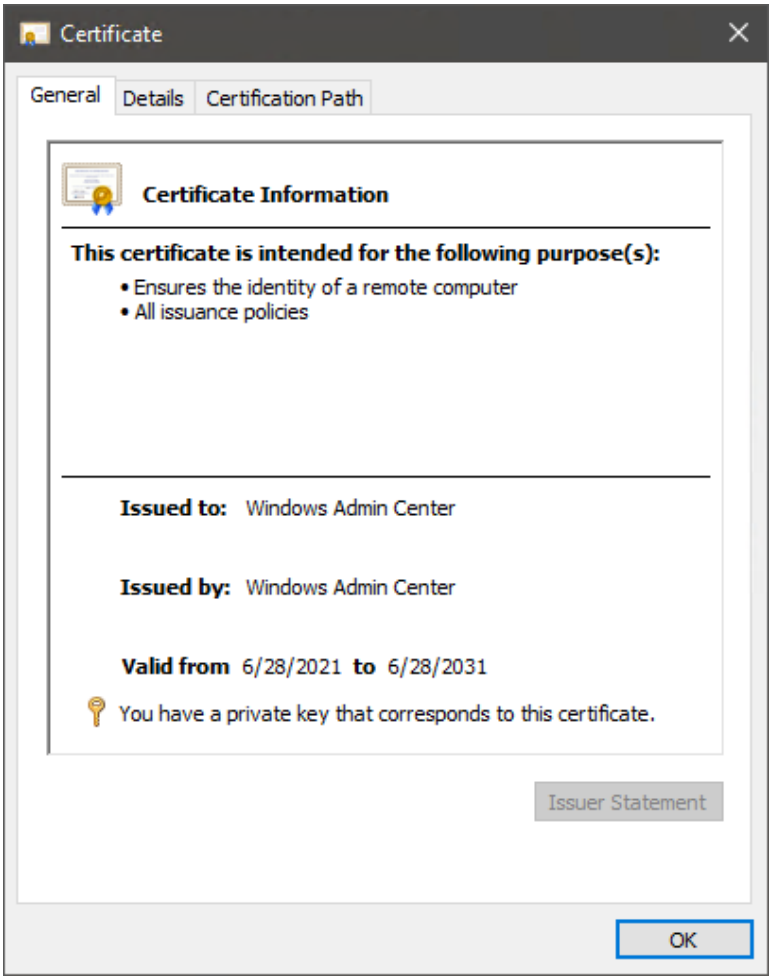
5. On the final screen, the installer shows you a mock-up of a certificate selection prompt. As indicated, WAC will prompt you to select a certificate the first time that you run it. You can only change the certificate by reinstalling Windows Admin Center, so take a close look. Below the screenshot, you'll see a checkbox that will automatically start WAC when you click Finish.



When you run Windows Admin Center, as promised, you receive a pop-up like the following:



However, the real dialog has one major difference from the sample from the installer: **a Certificate Information link**. That will show the certificate information dialog for the currently highlighted certificate so that you can verify prior to making a selection:



For Windows Admin Center to function in desktop mode, the system's current date must fall within the range of its validity period. It must also have the "Client Authentication (1.3.6.1.5.5.7.3.2)" OID. WAC would not have offered it if it did not fit these requirements, but you can double-check on the **Enhanced Key Usage** option of the **Details** tab. Mainly, you want to do what you can to ensure that you do not get tricked into choosing a forged certificate.

That's all that's necessary to install Windows Admin Center in desktop mode. The remaining portions of this chapter discuss gateway mode, so you can skip ahead to the next chapter if you will only use desktop mode.

HOW TO INSTALL GATEWAY MODE

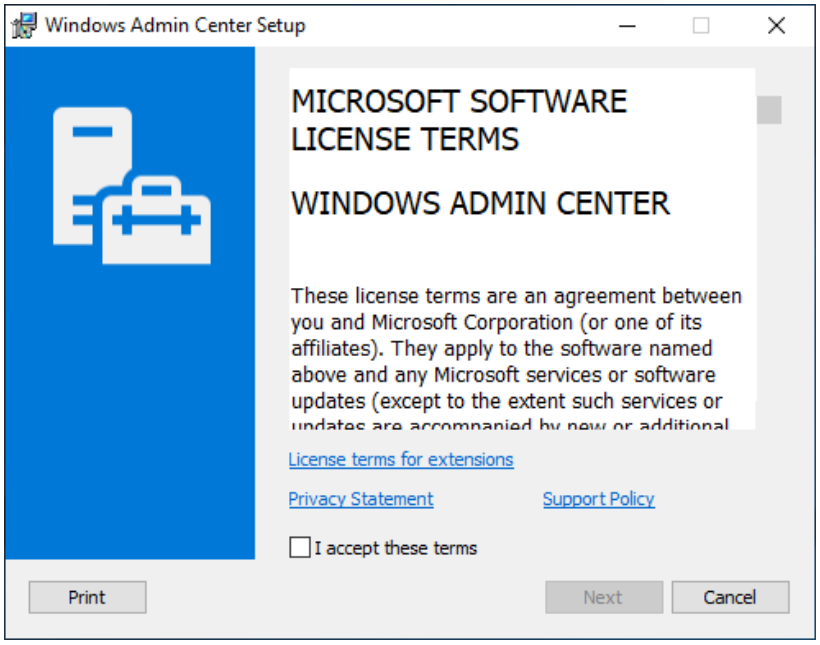
Just as Windows Admin Center's installer kicks off desktop mode when it detects a desktop operating system, it will launch in gateway mode when it detects a server operating system. The installer provides no way to manually override WAC's operating mode. You must use Windows Server 2016 or later, either LTSC or SAC. The installer works with or without the Windows Desktop Experience (meaning that you can run it on a server installed in Core mode).

You will need to use a PKI certificate for hosting WAC, and you should not use a self-signed certificate. The security chapter has a section on PKI that covers the why and how. If you want to deal with this later, use the offered certificate to get through the installer and revisit the certificate problem after you've explored security.

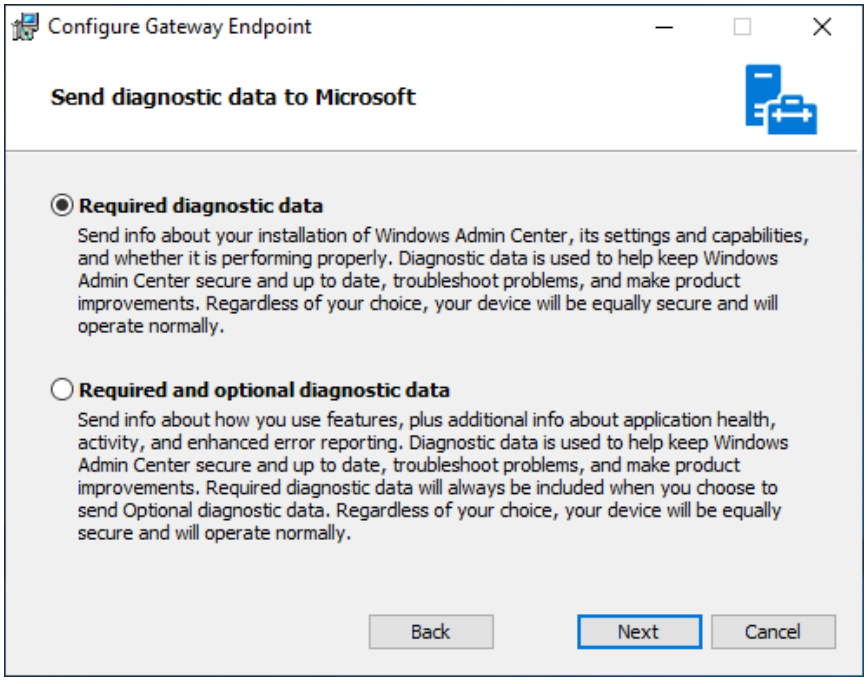
If you intend to install Windows Admin Center in high availability mode, do not follow these directions. You still need a valid PKI certificate. When you have that, jump ahead to the high availability installation instructions.

The gateway mode installer is nearly identical to the desktop mode:

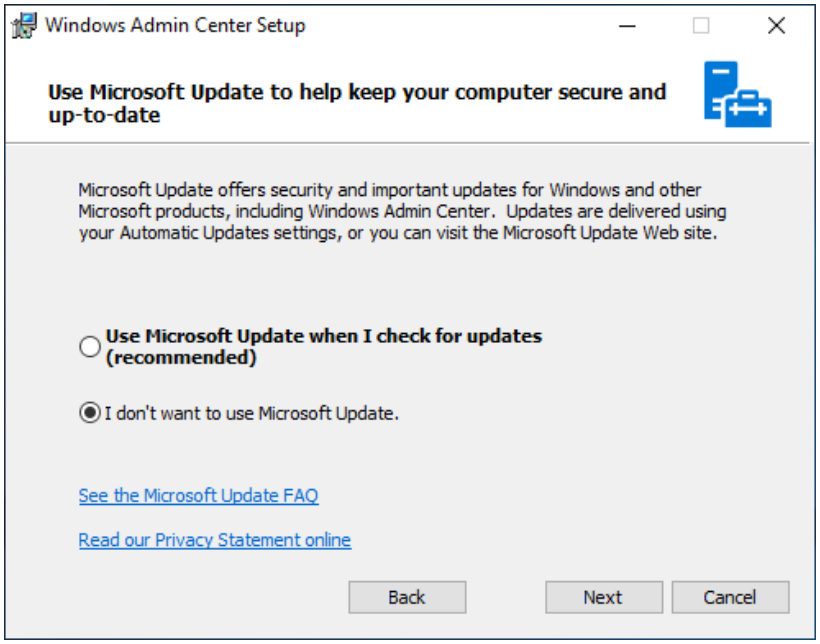
1. When you start the installer, it shows you the license agreement. You must check its box to accept before you can click **Next** to move on.



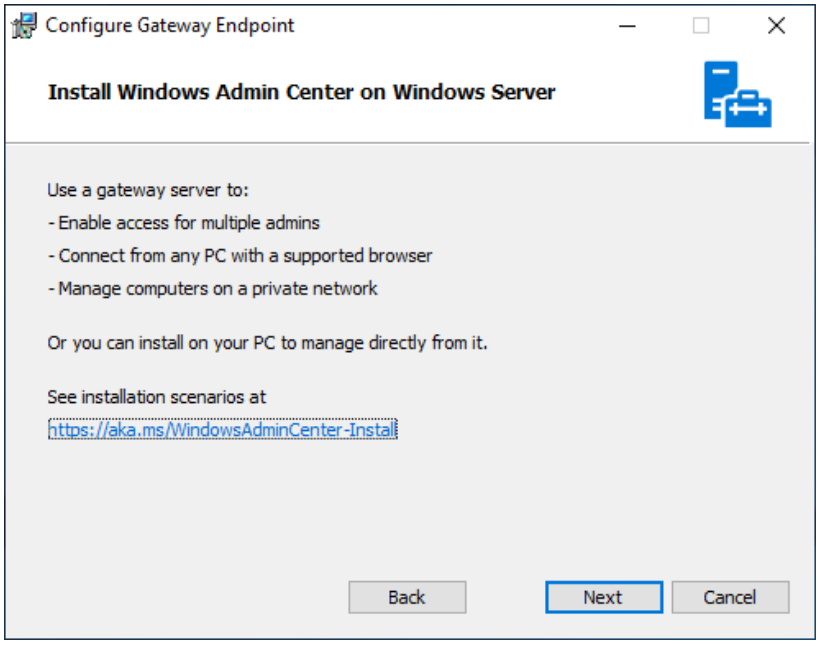
2. On the second screen, choose the amount of telemetry data to send to Microsoft. Microsoft aggregates information from optional data to understand feature popularity and usage. Make your selection and click **Next**.



3. Choose whether to use Microsoft Update to provide updates for Windows Admin Center. If you enable this, it activates Microsoft Update globally for the system if it was disabled. Ensure that any changes fit with your expected policy.



4. The next screen briefly explains the gateway mode and tells you about the option to run on a desktop. If you ran the installer on a system with the Windows Desktop Experience, you can use the provided link to view Microsoft's fuller discussion on desktop and gateway modes.

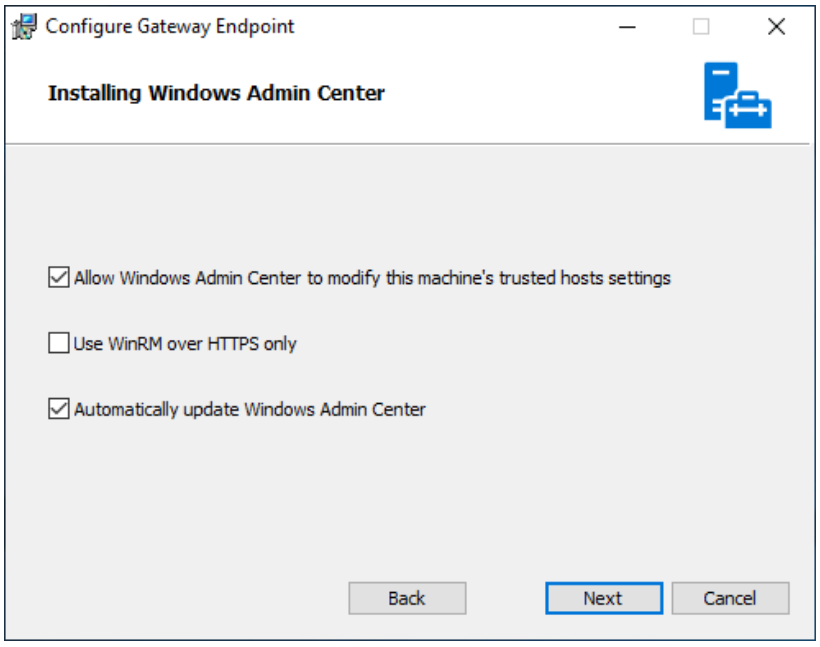


5. This section presents multiple selectable options.

    1. You can allow Windows Admin Center to manage the TrustedHosts settings for you. By "manage", it means that it will automatically trust every host.

    2. By default, WinRM (used by PowerShell Remoting, therefore by Windows Admin Center) operates over port 5595 in HTTP. You can opt to use HTTPS, which enforces fully encrypted communications. We will explain this the Security chapter.

    3. You get a seemingly duplicate question to allow Windows Admin Center to update itself. Whereas the Microsoft Update screen from earlier let you select the option to retrieve updates automatically, this one controls whether it installs them on its own.

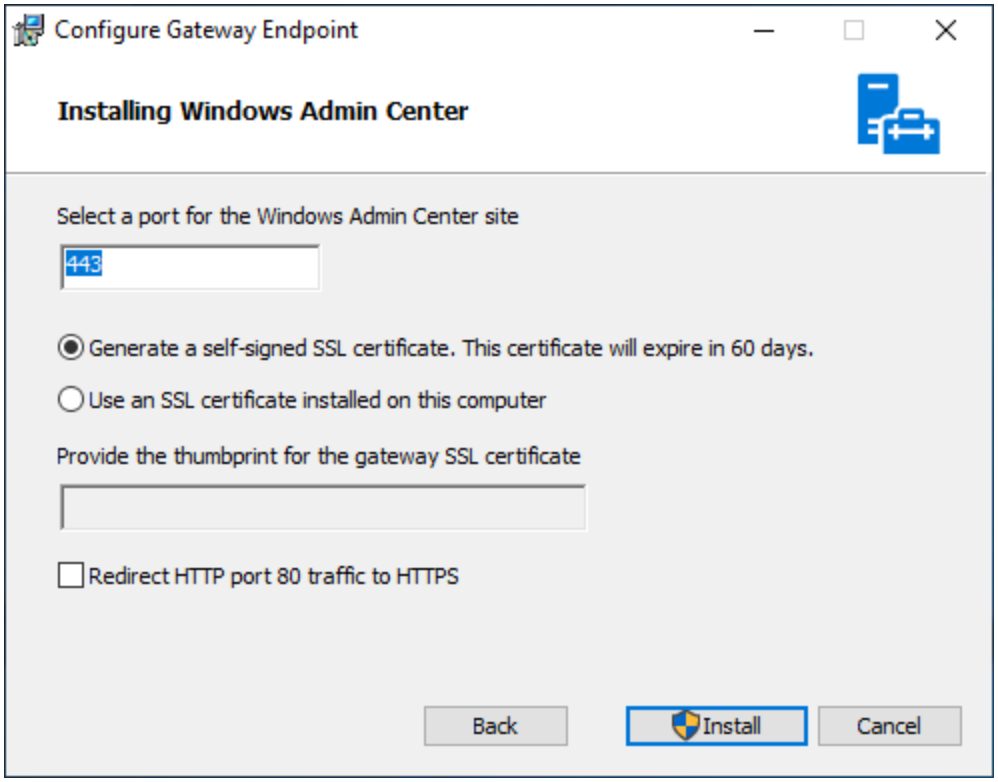


4. This screen provides options for WAC's web interface.

    1. By default, it runs on the standard HTTPS port of 443. You can change that if desired.

    2. If you prepared certificate in advance, paste its thumbprint here. If you did not, allow WAC to create a self-signed certificate for now.

    3. If you check the **Redirect** box, then WAC will redirect any inbound request on port 80 to the port that you selected.



5. After WAC installs, the final screen shows the host's default URL. It does not know anything about any other assigned DNS or subject alternate names. As long as you configured them correctly, they will also work as expected.

You have completed the Windows Admin Center installation and can now connect using your chosen client system(s).

USING WAC CERTIFICATE SELECTOR

You cannot replace or update the certificate that Windows Admin Center presents to web clients without reinstalling the program and going through the process of retrieving the certificate's thumbprint. Microsoft has indicated that they do not intend to ever make the process easier. Fortunately, you have access to a free open source solution written by the author of this book.

You can download the software from GitHub: https://github.com/ejsiron/CertWAC/releases. Due to its simplicity, it does not receive frequent updates. It is validated against new WAC releases, however. Note that it has not been tested with a clustered installation of WAC, but you should not expect that to work. Follow the high availability installation directions instead.

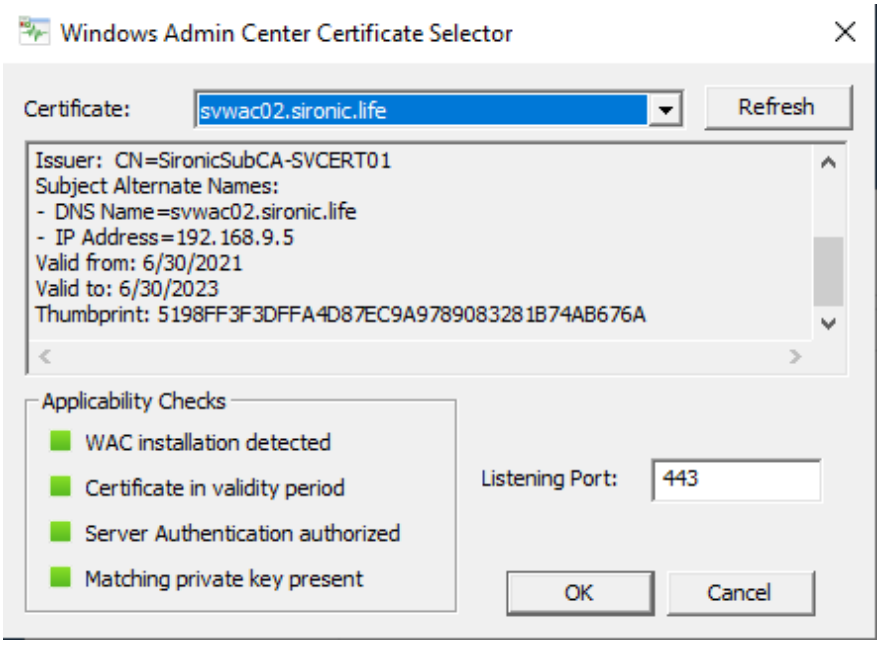
You can install the program permanently on to a host or you can use the EXE or ZIP releases to run the program one time. All methods work on any edition of any supported version of Windows Server, even without the Windows Desktop Experience.

Before operating the software, verify the system has these things:

- Windows Admin Center gateway mode has installed successfully
- A certificate within its validity period, with an Enhanced Key Usage/OID of "Server Authentication (1.3.6.1.5.5.7.3.1)" and a private key

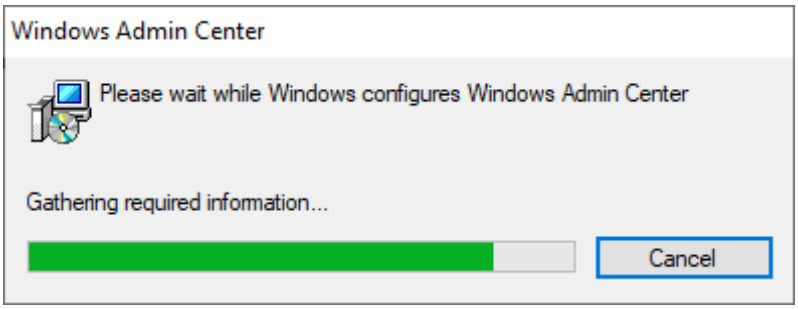
Also be aware that, because this program calls WAC's installer to perform the final work, it will cause a brief service interruption.

Once installed, WAC Certificate Selector has only one screen:



It will only light up the **OK** button if the system and selected certificate meets all **Applicability Checks**. You can also override the listening port if you wish. The software will not modify any other WAC setting.

When you press **OK**, WAC Certificate Selector invokes WAC's installer in unattended mode with the selected certificate and port information that you provided:



Once that completes, WAC Certificate Selector and the MSI progress bar will simply disappear. Verify that WAC now presents the correct certificate.

You can rerun this program any time you need to update WAC's certificate.

HOW TO CONFIGURE HIGH AVAILABILITY

In the chapter that explained the functional modes of Windows Admin Center, we talked about some of the risks of making your WAC installation highly available. Earlier in this chapter, we covered the requirements. If you understand the ramifications of high availability mode and have satisfied the prerequisites, the following steps will help you with configuration.

Before you start trying to run the script, it helps greatly to gather all information in advance. You do need to periodically re-run the script when your PKI certificate expires, but it will not accept all these items. Therefore, keeping them in a file for later use won't hurt anything, but you would only use in the event of a complete reinstall.

Things that you need to know/have:

- A fully functional cluster to host WAC
- The cluster's primary cluster name object has Full Control permission on its organizational unit in Active Directory
- A folder created on a cluster shared volume that will hold WAC data
- A DNS name to use for the highly available connection point
- The most recent MSI file for Windows Admin Center
- A PKI certificate with private key that includes at minimum the full DNS name that you will use for the highly available connection point, saved in PFX format. Optionally, add the DNS names of each node as subject alternative names. (Although not recommended, you can skip this part and instruct the installer to create a self-signed certificate)

To make this as easy as possible, use a text editor to build a splat variable that you can edit. The HA script does little error checking, so if you make mistakes, you will have to start over. Keeping the splat variable in a text program, even if only temporarily, might save you a lot of time and frustration.

With the data above, create a splat variable like this:

```
$WACVars = New-Object -TypeName HashTable

$WACVars.Add('clusterStorage',
'C:\ClusterStorage\WACData')
```

```
$WACVars.Add('clientAccessPoint', 'hawac')

$WACVars.Add('msiPath',
'C:\ClusterStorage\WACData\InstallFiles\WindowsAdminCe
nter2103.2.msi')

$WACVars.Add('certPath',
'C:\ClusterStorage\WACData\InstallFiles\WACcert.pfx')

$WACVars.Add('certPassword', (Read-Host -Prompt 'Enter
password for WACcert.pfx' -AsSecureString))
```

You do not technically need to use the **Prompt** parameter with **Read-Host**, but if you don't then it might look more like your PowerShell session has frozen than that it has paused to wait for input.

If you want WAC to use a short-lived self-signed certificate instead of a CA-signed certificate, leave out both of the above "cert" variables and use:

```
$WACVars.Add('generateSslCert', $true)
```

A few things to point out about the displayed script:

- You must provide only the computer name for **clientAccessPoint**. If you provide a fully qualified name, the script gets almost all the way to the end and then fails.

- Run this script directly from the console of a node, not from a remote PowerShell session. While a remote session might work, it might also fail due to some unforeseen delegation or CredSSP problem, and the failure might require you to do a lot of manual rollback and file deletions.
- You do not need to duplicate the locations for **clusterStorage**, **msiPath**, or **certPath**. You do need to know their full paths, and the cluster node that runs the install script must be able to access those paths.

With all that preparation out of the way, you only need to invoke Microsoft's install script with all your variables. All Microsoft's examples include the **Verbose** parameter as well, which will show enough output to keep you from worrying that PowerShell has stopped working. Start the installer like this:

```
.\Install-WindowsAdminCenterHA.ps1 @WACVars -Verbose
```

Test the installation by opening the full DNS name of the client access point in a browser. Sometimes it takes a few moments to start. Remember that WAC in high availability mode cannot redirect port 80 to 443, so always specify the **https://** prefix.

UPDATING THE PKI CERTIFICATE IN HIGH AVAILABILITY

Because making changes to the options that you selected during Windows Admin Center require a reinstallation of Windows Admin Center, this section appears here. Take notes or make bookmarks for future reference.

When you need to replace WAC's PKI certificate, pare down the command to just that portion:

```
$WACVars = New-Object -TypeName HashTable

$WACVars.Add('certPath',
'C:\ClusterStorage\WACData\InstallFiles\
WACcert.pfx')

$WACVars.Add('certPassword', (Read-Host -Prompt 'Enter
password for WACcert.pfx' -AsSecureString))

.\Install-WindowsAdminCenterHA.ps1 @WACVars -Verbose
```

You can maintain the above in a permanent PS1 file and use it as necessary.

TROUBLESHOOTING A HIGHLY AVAILABLE INSTALLATION

Microsoft's install script for high availability Windows Admin Center assumes the best and does nothing to recover from problems. If the script errors before making any substantial changes, then you can recall the line that contains the parameter(s) that the script complained about and correct them.

If the script gets far into the process and then fails, you will need to clean up before trying again. You can use the **Apps and Features** tool to remove Windows Admin Center (separately on each node). If you prefer the command line, or if you tried to install WAC on systems that do not run the Windows Desktop Experience, then call the .MSI file directly. As an example:

```
'C:\ClusterStorage\WACData\InstallFiles\WindowsAdminCenter2103.2.msi'
```

If you attempted to make changes to Microsoft's script and it didn't like them, then the installation or uninstallation routines may fail to stop the WAC service. In that event, force the service process to stop:

```
Stop-Process -Name sme -Force
```

If you can run the above at the point where the uninstaller freezes up, it should immediately proceed through and finish successfully. Otherwise, start the uninstall process after you successfully stop sme.exe.

If you need to uninstall WAC from the nodes, make sure to clean up the temporary and data folders before trying again. Look in the location that you provided for **clusterStorage**. If the installer hung after you modified the install script, revert the install script to defaults before trying again.

If reverting to defaults and correcting all problems with your input parameters still results in a failed or hung install script, then you have some other environmental problem that WAC can't cope with. Sometimes the errors will guide you to resolution; other times they will not. Read back over the requirements and check that you performed all preliminary steps and that nothing has undone your preparations in the interim.

SECURITY CONSIDERATIONS

Installing a brand-new administrative tool makes all of us want to jump right in and start tinkering. And while you certainly can do that with Windows Admin Center, expect to encounter some frustrating roadblocks. As you navigate its approach to centralized processes, you will encounter security barriers – perhaps for the first time. Also, unfamiliar territory sometimes leads to risky practices. Hacking around things to get into the meat of WAC could leave things in a worse security posture than you found them. This chapter introduces these concepts and solutions while aiming to help you enjoy a smooth experience.

CONTROLLING ACCESS TO WINDOWS ADMIN CENTER

When you install Windows Admin Center on a desktop operating system, it takes care of the access question for you. When you start the program, WAC starts up a browser session and the **SME.EXE** process. SME.EXE listens on TCP port 6516 (unless you changed the default) and only accepts connections from local browser sessions. When you close the spawned browser, SME.EXE stops with it. So, only local accounts running a console session on the system can access WAC.

Things change a bit for gateway mode. WAC still runs under SME.EXE, but as an always-on service listening for any inbound connection on TCP port 443. It has several ways to control access to its interface.

Before you start investigating the techniques in this section, be aware that the items here only control access to the Windows Admin Center interface. Nothing here has any impact on a user's ability to manage a resource through Windows Admin Center. WAC will always challenge a connected user for credentials to any object that their account cannot otherwise access.

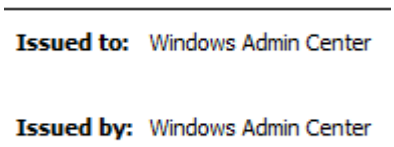
PUBLIC KEY INFRASTRUCTURE CERTIFICATE

Windows Admin Center runs as a web service on the HTTPS protocol, so it requires a certificate. As covered in the environmental requirements section, feel free to use the default self-signed certificate for desktop mode WAC installations. For gateway mode installations, self-signed provides no security and requires frequent maintenance.

If the term "Public Key Infrastructure certificate" does not sound familiar, then you probably know them as "SSL certificates". "Secure socket layer" (SSL) refers to a deprecated protocol set that was used to secure computer communications. Transport layer security (TLS) superseded SSL. You should never find any genuine SSL communications today. However, the term "SSL certificate" persists. Because certificates don't tie to any particular protocol, and because a PKI generates, distributes, and manages the certificates in question, this book will refer to them as "PKI certificates".

The Altaro site has a series that goes over PKI in detail: Public Key Infrastructure Explained | Everything you need to know (altaro.com). This book will not repeat that effort. For our purposes in this work, understand these things:

- The term “self-signed certificate” refers to a certificate that was issued by the same entity that validates itself with the certificate. It does not mean a certificate generate by your firm’s internal PKI. You can identify a self-signed certificate when its Issued to and Issued by fields contain the same entity:



- Self-signed certificates have no practical purpose in inter-system communication. They suffice for desktop mode WAC because any event that could compromise a certificate generated by the system could corrupt any certificate on the system, including those that represent trusted authorities.

- You do not need to purchase a commercial certificate. If you do not have the will or the resources to create an internal PKI, you can use public services such as Let’s Encrypt. Many public services also only provide short-lived certificates, but most last longer than WAC’s self-signed certificate and all provide the desired third-party validation.

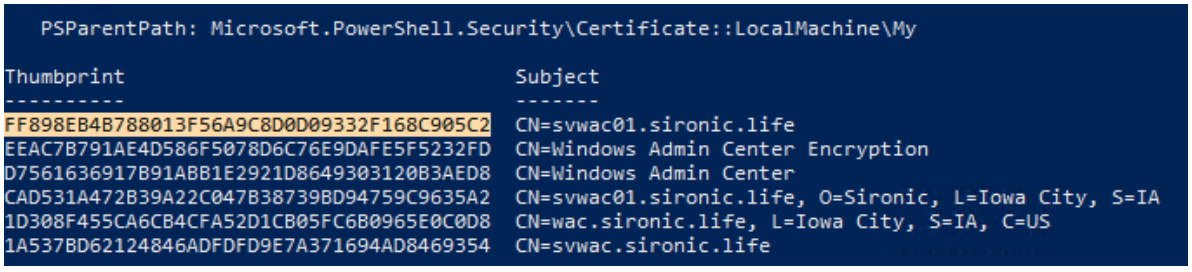
- The only way to update WAC’s certificate is to reinstall WAC.
  You will spend overall less effort and time in standing up a simple PKI than dealing with that every two months.

The installation chapter includes a section on a third-party tool called WAC Certificate Selector that can easily apply an issued PKI certificate to Windows Admin Center. If you don't want to use a third-party tool for WAC or if you want to complete the WAC installer in one round as described in the gateway mode installation section, then you will require a PKI certificate. At installation time, the certificate must exist in the host computer's machine store, have the "Server Authentication (1.3.6.1.5.5.7.3.1)" OID/Enhanced Key Usage, subject alternate names (SANs) for every DNS name that you use for WAC, and the private key that matches the certificate's public key.

When you run Windows Admin Center's installer, it will ask you for the thumbprint of that certificate. To make the installation go as quickly as possible, have that key ready. While you can get to that through the Windows GUI, the certificate dialogs tend to append an invisible invalid character which breaks the installer when pasted. Use PowerShell to get your thumbprint instead:

```
Get-ChildItem Cert:\LocalMachine\My
```

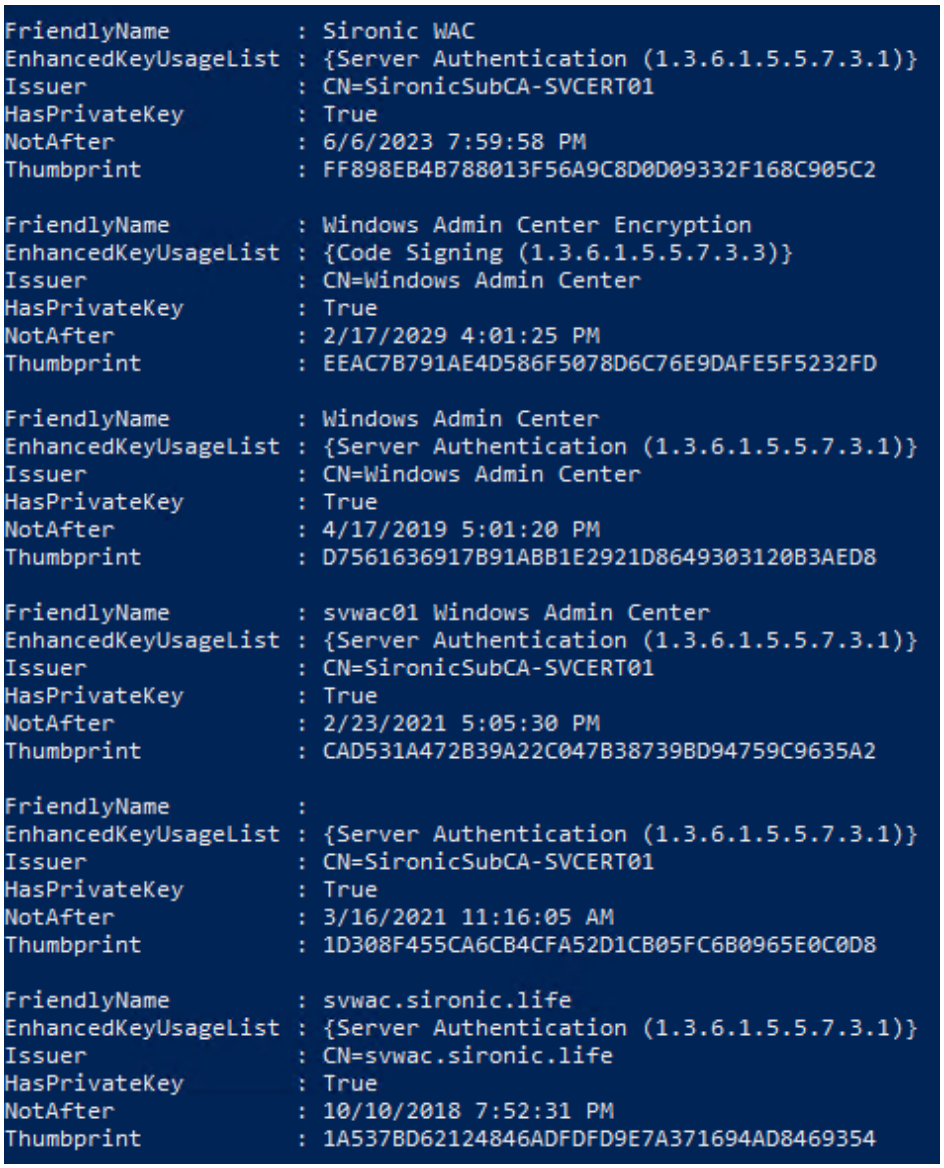
Use the mouse to highlight the desired thumbprint and press [Enter] to place it on the clipboard.



If the output produces a list with ambiguous results that don't help you to choose, you can ask PowerShell to show more details. A suggestion:

```
Get-ChildItem Cert:\LocalMachine\My | select -Property
FriendlyName, EnhancedKeyUsageList, Issuer,
HasPrivateKey, NotAfter, Thumbprint
```

This shows every data point that you need to discern that WAC can use a certificate.



If that doesn't tell you enough, you can just ask for all the information:

```
Get-ChildItem Cert:\LocalMachine\My | fl *
```

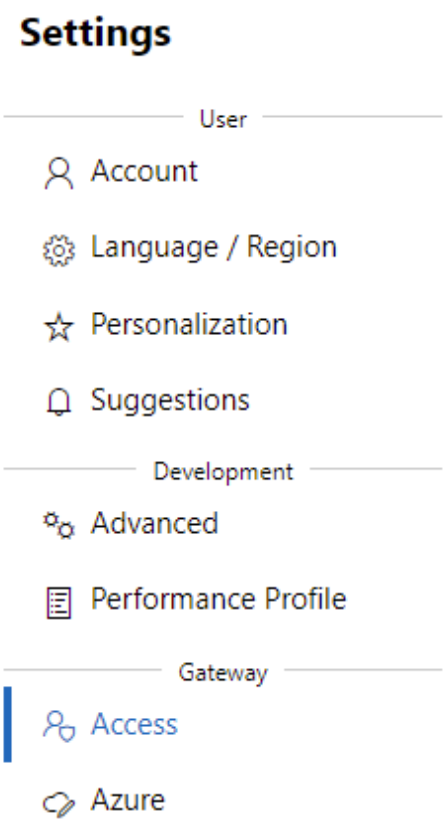
Don't allow the need for certificates to turn into a point of resistance. Going forward, PKI will become more ubiquitous, not less.

All that aside, Microsoft should reconsider the requirement to reinstall WAC to update the certificate. It was a popular request while the UserVoice site was still active, but the WAC team made some excuse and claimed that it could not be done. Whether or not that's true, do not expect the situation to change in the short run.

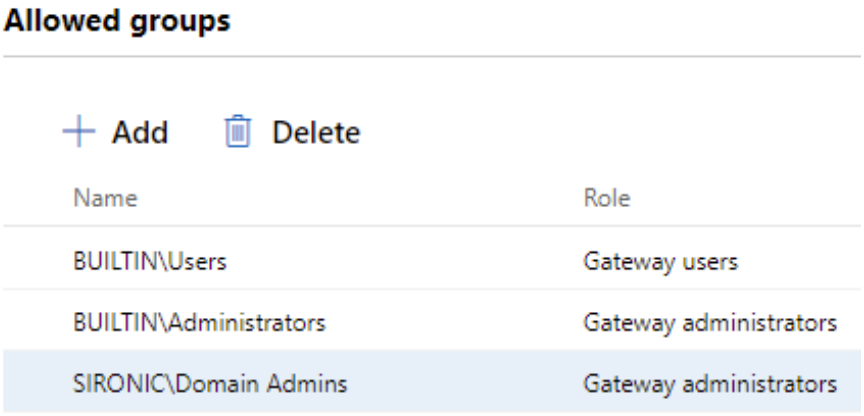
LOCAL AND DOMAIN USER AND GROUP ACCESS

With no changes, Windows Admin Center in gateway mode follows similar access rules as desktop mode: it will permit any account that the gateway host recognizes as an administrator or user. WAC automatically grants gateway administrator permissions to administrators and gateway user permissions to users. As a result, you can use the local Administrators and Users security groups to control WAC access. You only need to place local or domain accounts in the respective local groups. You can also use security groups that contain the desired accounts.

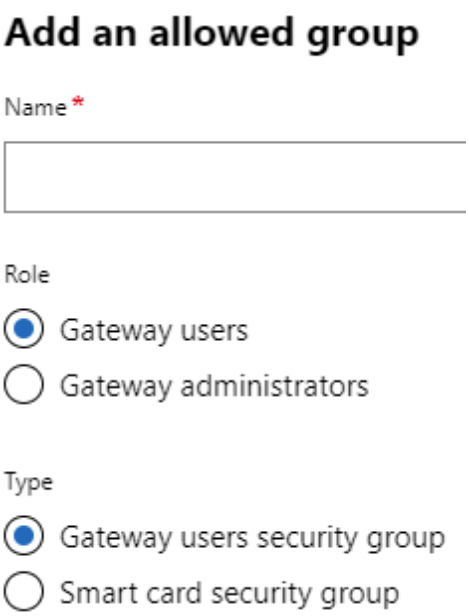
WAC's Settings section has a rudimentary control to partially handle this task. In the **Settings** menu, click **Access** under the **Gateway** section.



The main pane will show the groups that currently have access to WAC.



You can use the **Add** button to place an existing group in the list.



This process cannot create a group; you must use a group that exists on the gateway host or in the domain. You also cannot grant access to specific users here. However, you can use WAC's **Local users & groups** tool on the WAC host to control group membership. That has much more utility and granular control.

AZURE ACTIVE DIRECTORY AUTHENTICATION

If desired, you can add Azure Active Directory authentication as a second factor. This does not replace the requirement to authenticate to Windows Admin Center using an account that it can verify; it adds the Azure AD login on top.

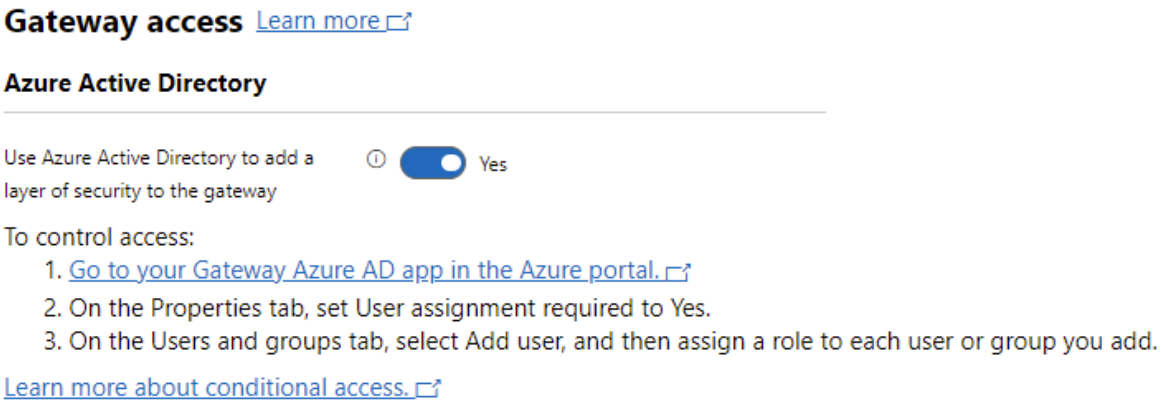
To enable AAD integration, you will need to register your Windows Admin Center installation with Azure first. Due to the organization structure of this book, we have not yet introduced that. You can skip ahead to the full directions and return. We will not cover setting up and configure Azure Active Directory.

In order for this to work as expected, the domain that contains accounts that you want to use in WAC must be synchronized to AAD.

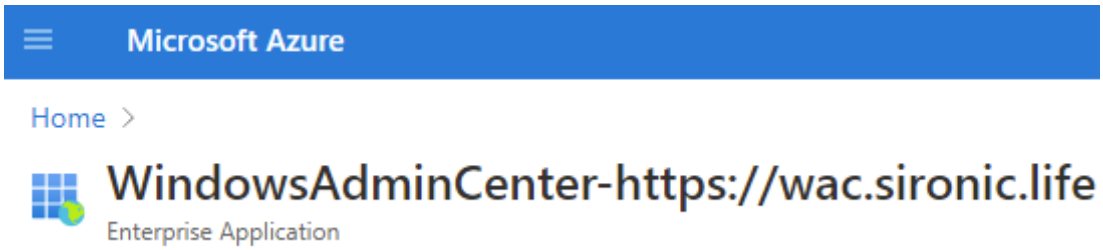
1. To start, access the global settings while logged in as an administrator (gear icon at the right of the title bar.)

2. Switch to the **Access** tab in the **Gateway** section.

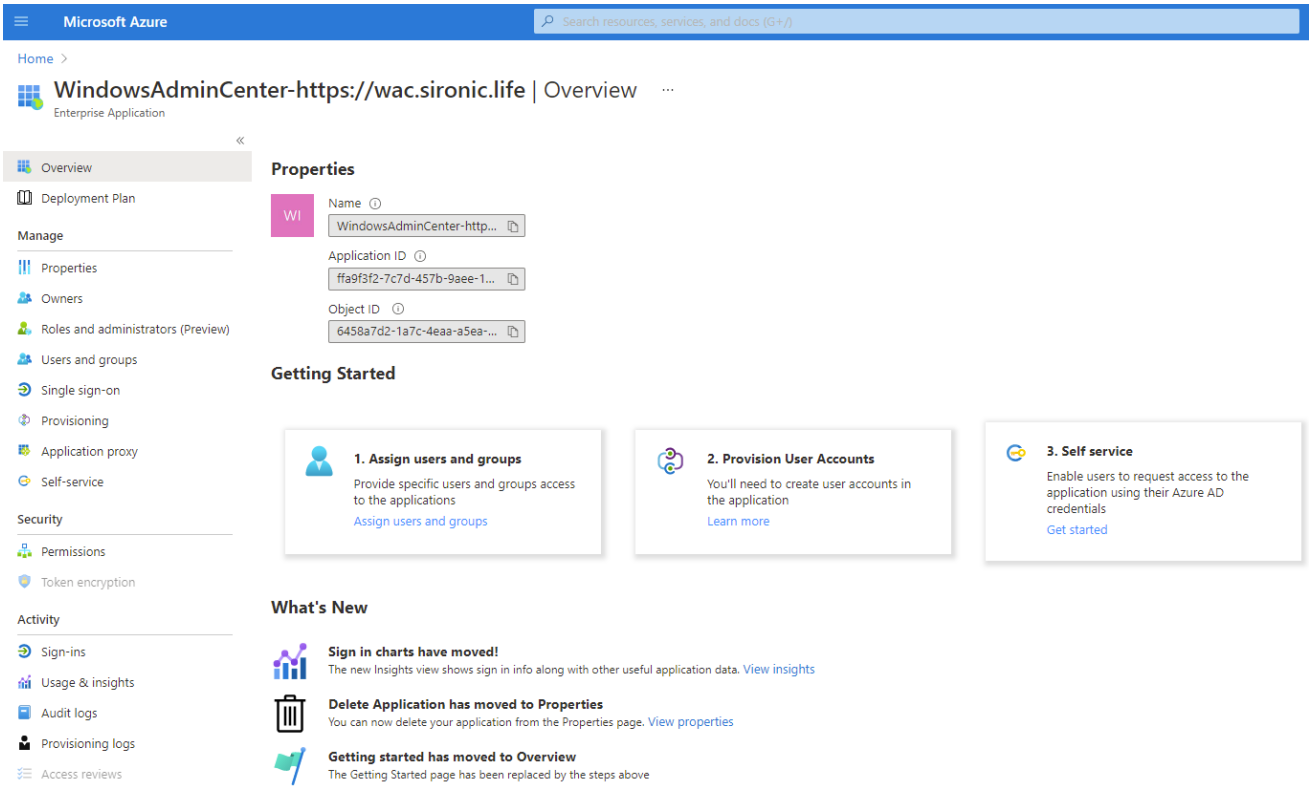
3. Set the **Use Azure Active Directory to add a layer of security to the gateway** slider to **Yes**. The screen will change as follows:



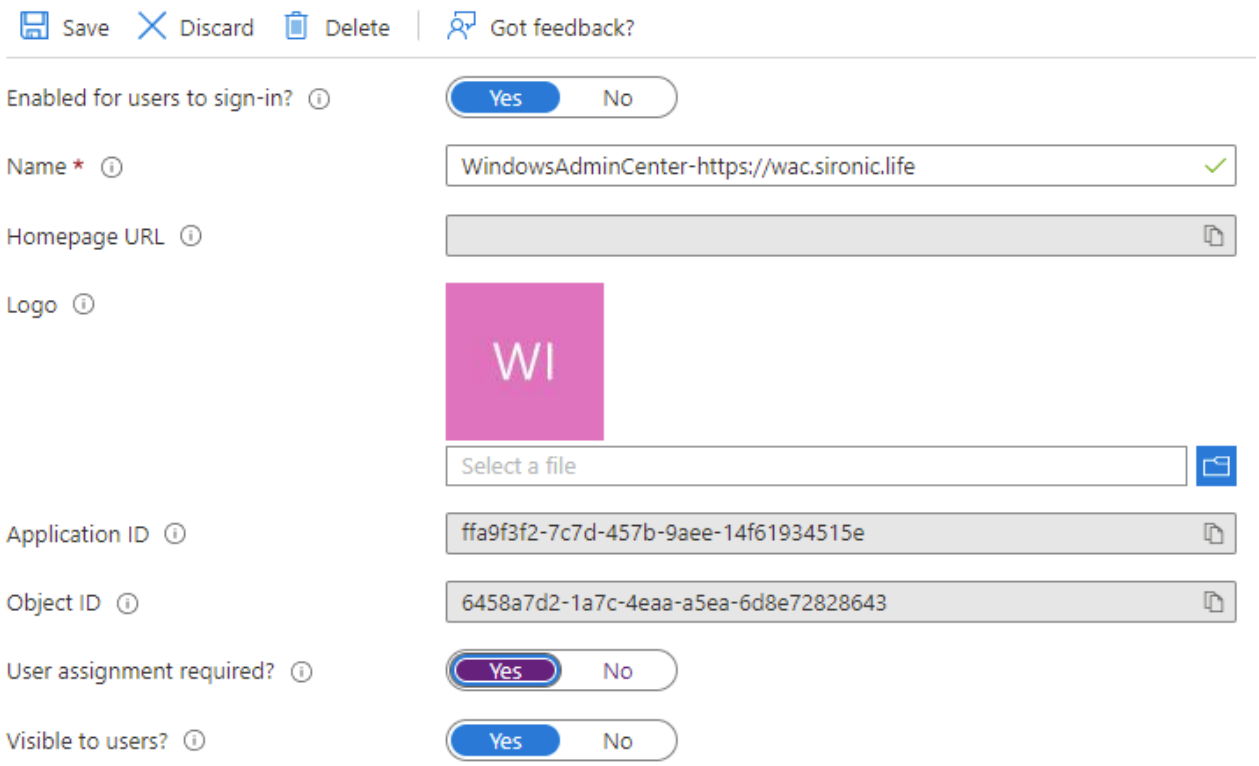
4. Click on the link presented in the first list item to go to your Azure portal. Once you log in, it will take you directly to the Azure application that represents your on-premises Windows Admin Center installation.



5. As the message in WAC indicated, click the **Properties** tab at the top of the **Manage** section in the left-hand menu.

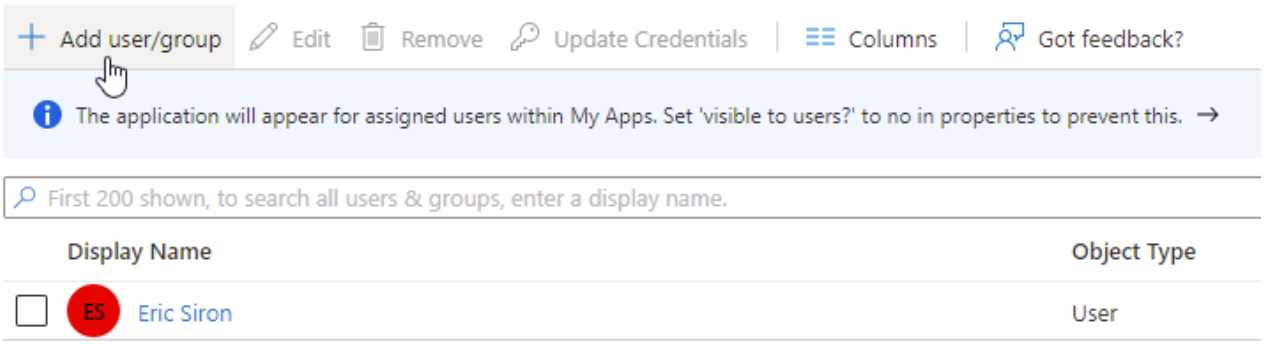


6. On the **Properties** screen, change **User assignment required** from **No** to **Yes**.

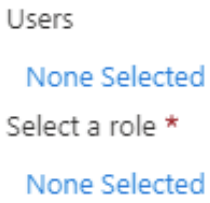


7. In the menu at the left, click **Users and groups**, still in the **Manage** section.

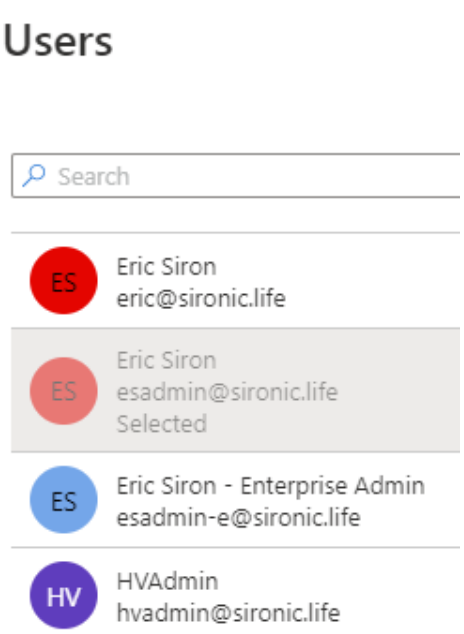
8. On the **Users and Groups** screen, you will likely see an entry that corresponds to your Azure account. If it ties to your on-premises AD account, then you check the box next to it and click **Edit**. Otherwise, click **Add user/group**.



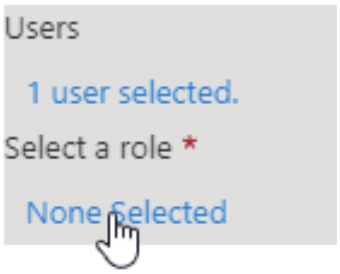
9. Azure presents a simple screen with two items. If you selected a user above, then it will indicate that. Otherwise, it will say **None Selected**. To select a user, click that link. If you already have a user selected, skip to step 12.



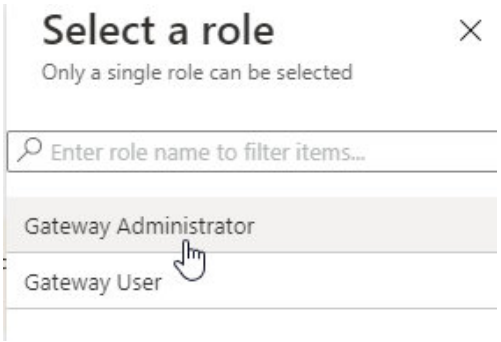
10. A user selection screen will pop out from the right. Highlight the users that you want to add at this time. If you have purchased a higher tier of AAD, you can also add groups. As you highlight users, they will appear in a details pane directly below the list. You can remove any accidental adds from there. Once you have the users that you want, click **Select**.



11. Azure will return you to the same screen that you saw in step 9, but it will now show that you have selected users. You can click that text to change your selections (repeat step 10). When you have finalized your selection, click **None Selected** under roles to continue.

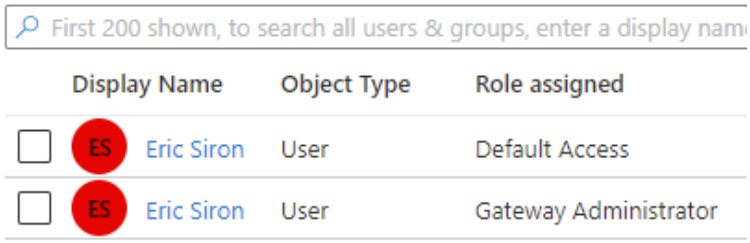


12. Select the role that you want to assign to all the selected users.



13. Click the **Assign** button at the bottom of the screen.

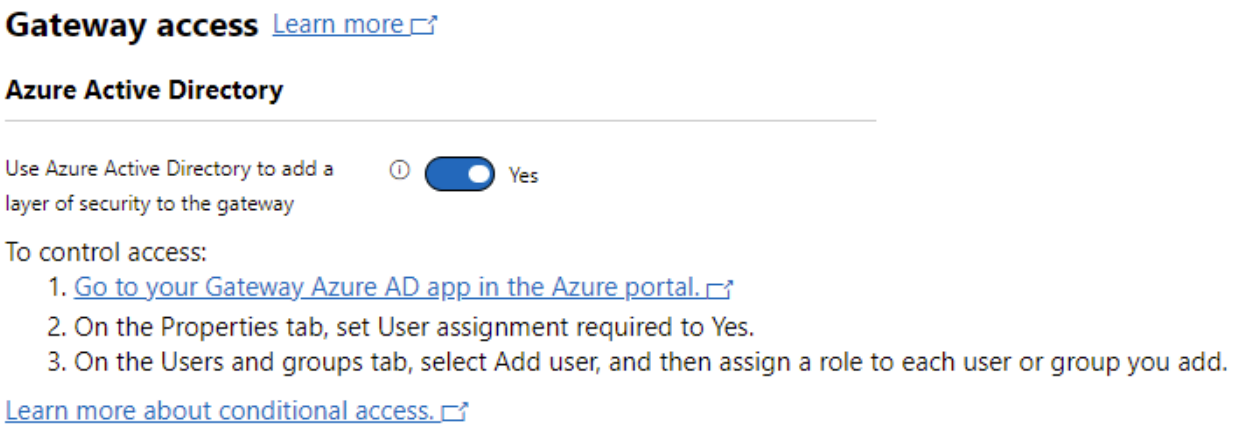
14. Azure will return you to WAC's user list and your selected account(s) will have the assigned role:



15. Return to Windows Admin Center and test connectivity.

Sometimes after initially configuring AAD authentication, Windows Admin Center stops responding. Restart the ServerManagementGateway service and try again. If you continue having difficulties, you may need to disable AAD authentication in WAC.

Once you switch to AAD as the second factor, you only control access via the Windows Admin Center blade in Azure. Your WAC **Access** settings page turns to this:

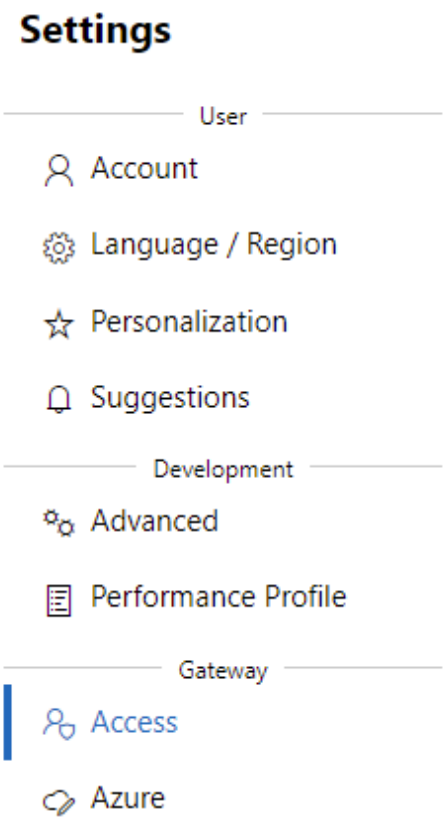


If your AAD account does not have the ability to add groups, AAD authentication may become tedious. If you open your Windows Admin Center installation so that you can access it outside of your organization without a VPN or similar connection, consider bumping up your AAD level. WAC's only supported alternative second authentication factor is smart cards.

SMART CARD AUTHENTICATION

While WAC's **Settings/Access** section does not do the greatest job of managing users and groups, it does present the only way that you can require a smart card as a second factor.

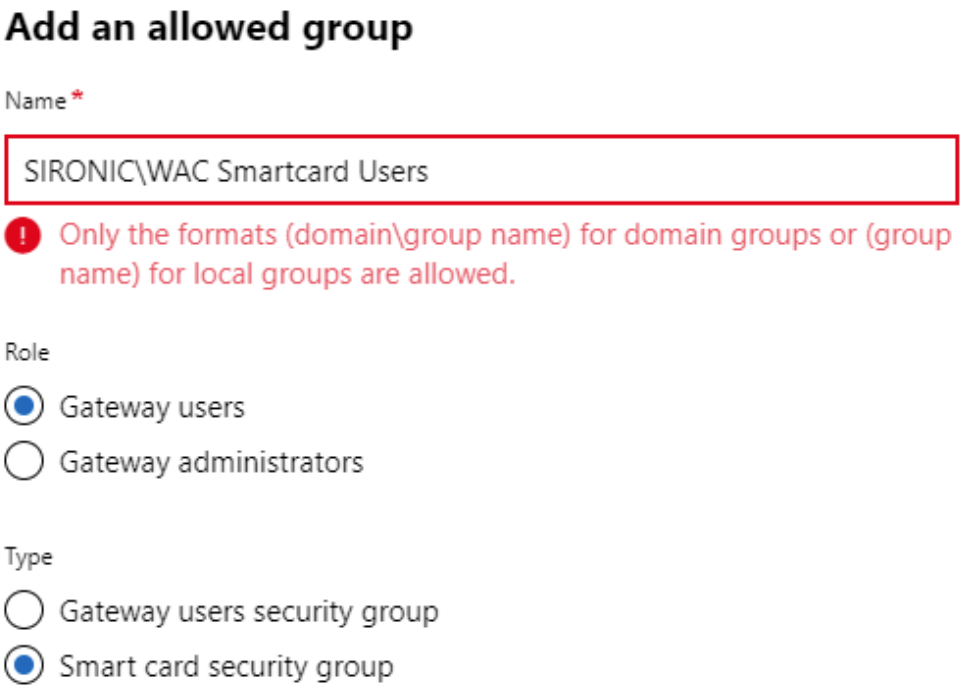
To start, go to the **Access** entry in the **Settings** area.



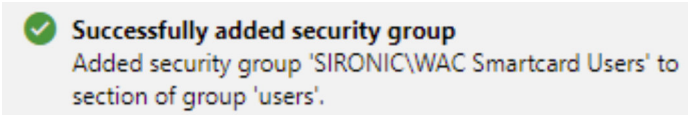
Note: If you have enabled Azure Active Directory authentication, then you will not have any options to change anything here. You cannot use both smart card and AAD as WAC's second authentication factor. However, you can leave WAC connected to Azure while disabling AAD authentication if you prefer smart cards.

Under **Allowed Groups**, click the **Add** link. Enter the name of an existing domain group that contains smart card-authenticated users. Change the **Type** to **Smart card security group**.

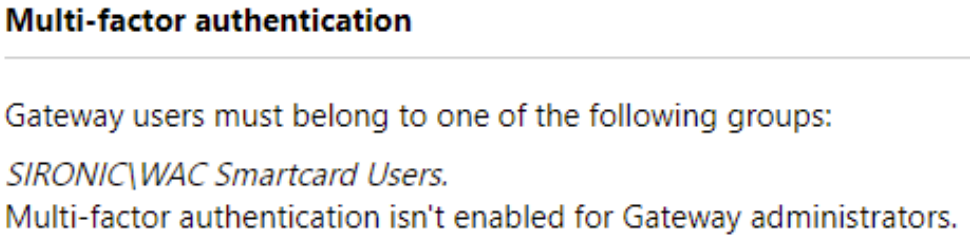
Note that at this time, a bug exists that incorrectly flags group names with spaces as invalidly formatted, but it will accept them.



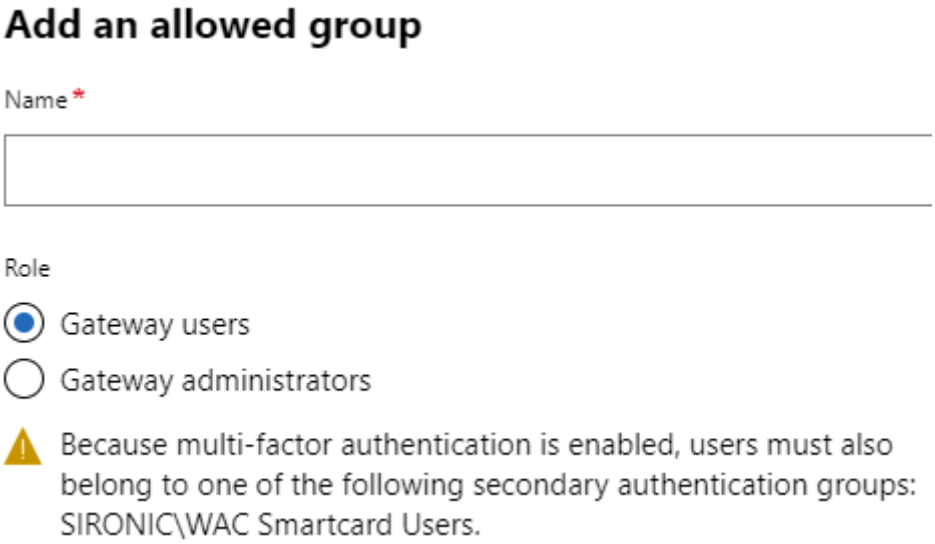
If WAC can find the specified group, it will add to the list and pop-up a notification.



Adding a smart card group modifies WAC's behavior. In the **Access** pane, the **Multi-factor authentication** group now indicates a requirement for membership in the smart card group:



Also, if you add a new group, WAC will explain that members must also belong to a smart card group:



In this scenario, we have only established the smart card group membership for gateway users. Gateway administrators can still login with only a password. Repeat the above process to establish a **Smart card security group** for **Gateway administrators** and the same rules will apply to administrators.

Deleting all smart card groups for users will remove the smart card requirement for gateway user logins. Deleting all smart card groups for administrators will remove the smart card requirement gateway administrator logins.

Three things to know about WAC's smart card restriction:

- Windows Admin Center enforces dual group membership, not smart card usage. If a user successfully authenticates as a member of an authorized user/administrator group and a user/administrator smart card group, WAC allows access. WAC does not know whether that person used a smart card to authenticate. You must use an Active Directory tool to require smart card login for the smart card group.
- Windows cannot enforce smart card authentication on local users and groups. Therefore, WAC cannot enforce smart card authentication on local users and groups. Nothing in the interface makes this obvious. Always use domain group membership to control smart card access.
- Local administrator accounts can always log in using only a password. When WAC challenges you in a browser session, enter the username in the format of "HOSTNAME\Administrator" to use the local account instead of a domain account.

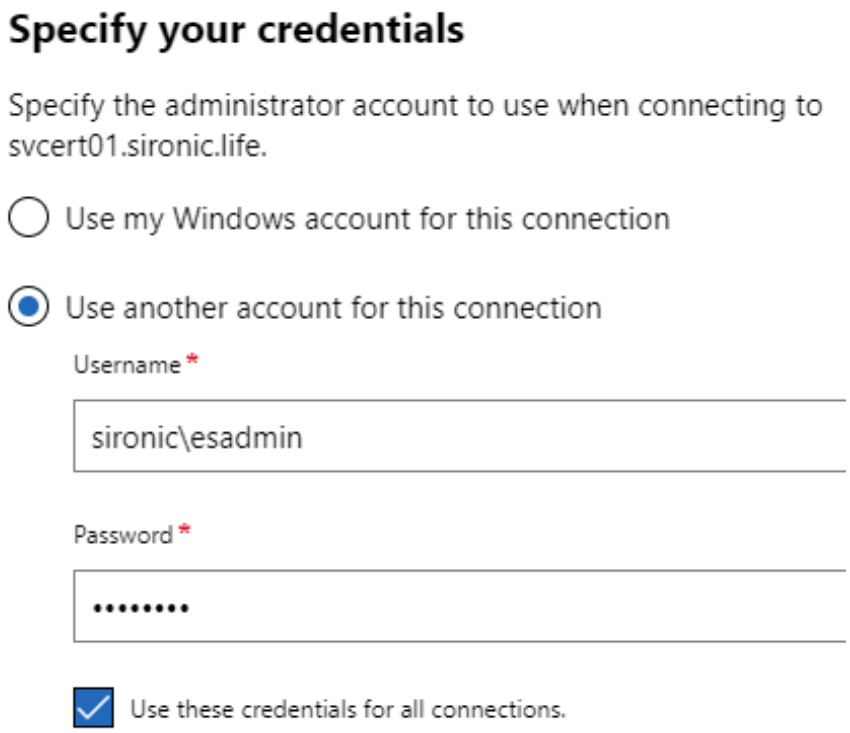
Smart card authentication does not see a lot of usage in Windows Admin Center. Your experience may be less than ideal. Remember to use the feedback links.

PER-SESSION CREDENTIALS

After dealing with access to the Windows Admin Center interface, you need to turn to the broader scope of access to systems. If you start using WAC immediately after installation without any environment configuration, then it will prompt you for access credentials when you try to manage any system except the gateway.

If you have installed WAC on your desktop, then you can select the **Use my Windows account for this connection** option. If you started the application with a Windows account that has sufficient rights to access the target, it will work immediately. In all other situations, you will need to provide specific credentials.

WAC will use the credentials that you provide to connect to the target, or if you tell it to, it will use them for all other connections as well.



No matter what, if you close the current session to Windows Admin Center, it will prompt you for credentials in the next session. If anything happens to invalidate the credentials during your current session (e.g., expired password, disabled account, a credential saved for all connections that will not work on all connections, etc.), then you will receive additional prompts.

You cannot configure WAC to persist credentials between sessions. If the target system does not belong to a domain, then you will need to authenticate each time. You can use a password manager to populate the fields.

As you might notice on the credential prompt, you can configure delegation within a domain environment to enable single sign on from your Windows account. The next few sections will cover that topic in detail.

THE SECOND HOP

Configuring Windows Admin Center for single sign-on can quickly feel like a chore. While we can all acknowledge that it inconveniences us, the barrier exists as an important security feature. We commonly refer to the situation as “the second hop”.

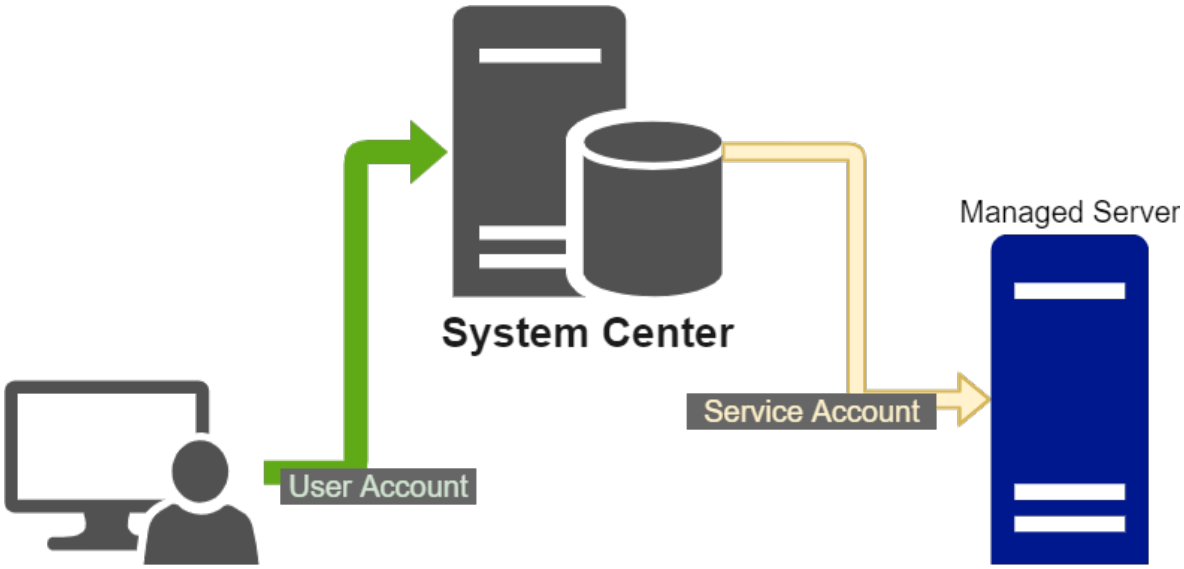
If you have always managed your systems from their consoles, a remote session, or a management computer, then you might never have encountered “the second hop”. This condition appears any time you need to access a resource through an intermediary system that operates on your behalf.

THE SINGLE HOP

Most administrators have used workarounds to the second hop for many years without ever realizing. For comparison, let’s start with a management system that operates using a more traditional methodology: System Center. When you configure a System Center product, it asks for credentials to use on targets. Most System Center administrators use an informal “service account”. They create an account in the directory that nothing but System Center ever uses. That account has administrative privileges on all systems controlled by the System Center product. Then, the System Center admin indicates which domain users have access to System Center and defines the scope of their activities.

Note: Active Directory includes formalized Managed Service Accounts. "Service account" as used in this explanation only refers to a standard AD account used for running services.

In this paradigm, System Center always performs its duties using the service account. System Center accepts the responsibility of ensuring that no one accessing its consoles or shells can perform any action outside of their designated privileges. So, the user initiates one "hop" to System Center, which then initiates its own single "hop" to the target system(s).



The service account approach bypasses complicated security restrictions by guaranteeing that System Center performs all work and always has the necessary power. The managed system only sees activity from System Center, not the individual using System Center. As a side effect, that means that resource owners lose most control over access. As a benefit, it also means that you have a narrower scope for unauthorized activity. Due to the ubiquity of this management

style across the industry, most organizations have adapted their trust and threat models to accommodate.

When a firm can't adjust operations, they have to incorporate workarounds to the workaround, such as isolated System Center installations, management domains, and all other sorts of solutions. All these add complexity – often without commensurate value. With that in mind, let's shift toward the problem that service accounts work around with a brief look at authentication and authentication.

AUTHENTICATION AND AUTHORIZATION

At the final point of access, security is a binary process: no matter the object, no matter the action, you can or you cannot. Reaching the appropriate point reliably and practically requires substantial effort. Like all machines, computers will do as told in accordance with their design. They have no concept of security. Humans must impose it.

We address most computer security concerns by separating them into two problem domains: authentication and authorization. For each transaction, we ask two questions:

- Who are you?
- What are you allowed to do?

These two items always appear together. In the scope of security, just knowing someone's identity has little value. Without identity, permissions have no applicable purpose.

When you investigate the various schemes for authentication and authorization and how people use them, a third related factor emerges that influences both: authority. Note the common root of "authentication", "authorization", and "authority". Authority provides the basis for trust to validate identity and permissions. As an example, consider a simple identity challenge conversation between two people:

"Who are you?"

"I am Johnathan Jones."

"How do I know?"

"Because I say so."

Does this work? If the challenger guards a secure item that the other wants to access, does these responses provide adequate identity? If the challenger knows and can positively identify Johnathan Jones, then they suffice. Otherwise, this person claiming to be Johnathan Jones should present credentials issued by someone that the challenger considers an authority.

If "Johnathan Jones" successfully convinces the challenger of his identity, that alone does not guarantee access to the secured object. The challenger must next determine if Johnathan has the necessary permissions. Perhaps a list of permitted identities exists. If Johnathan Jones' name appears on the list, then the challenger has the two necessary components to decide to grant access.

In computer security, several technologies cover these facets. In the scope of Microsoft and WAC, we have three authorities:

- Active Directory (AD) domain controllers
- Certificate authorities (CAs)
- Individual Windows systems via Security Account Manager (SAM) databases and local certificate stores

Active Directory and certificate authorities function as centralized authorities. Both have their own methods. Even a minimal explanation goes far beyond the purpose of this book. For now, recognize their role as third-party arbiters of identity.

The SAM and certificate stores on an individual host have similar functionality to AD and CAs, respectively, with one major difference: a host has no independent way to validate anyone's identity. Going back to the human exchange above, the SAM/local store method aligns with the, "Because I say so" identification method. When a domain controller or CA gets involved, then the justification becomes, "Because this entity that you trust has verified my identity." If Johnathan Jones had presented an identification card issued by the same governmental authority that had provided the challenger's identification card, then the challenger could have relied on that. Furthermore, that response would suffice to explain to someone else why the challenger chose to trust the entity.

Authorization has less to explain. Primarily, you need to understand differences in location. Computer security typically manages authorization via access control lists (ACL). Some systems use different names for the same thing. All such techniques mean a list of entities matched to explicitly granted

or denied permissions. Unlike authentication, central authorities typically do not govern authorization. As an example, the Administrators group built in to each Windows system may contain local user accounts and domain user accounts. Appearance on the list determines what an account can do (authorization). However, if the account exists in Active Directory, then only a domain controller can authenticate it.

That background enables us to understand the second hop and the problems that it creates.

UNDERSTANDING THE DOUBLE HOP PROBLEM

Recall the explanation of the service account solution that System Center uses from the Single Hop section above. As you look at the diagram and thinking about the scenario description, notice that it contains three distinct entities: the managing user, the System Center installation, and the managed target. All this exists to satisfy a particular goal: allow the user to manage the target. Add in all that you know to describe how to securely achieve this goal in an abstract sense:

1. The user authenticates to System Center.
2. System Center checks that the user has permission to manage the target.
3. System Center authenticates to the target.
4. The target checks that System Center has local management permissions.

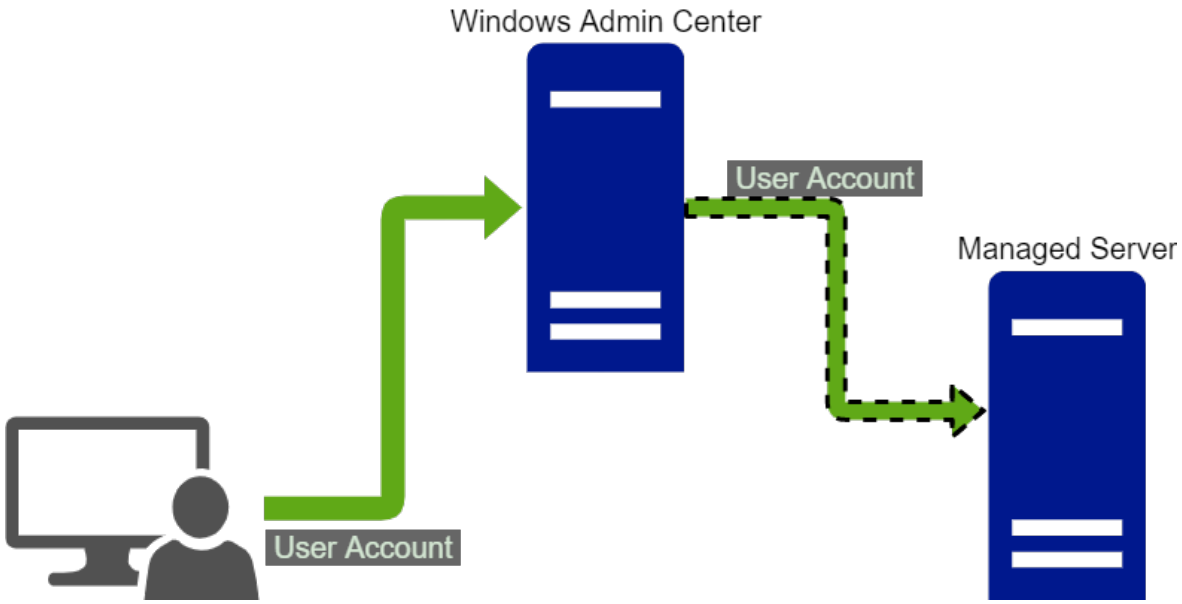
Once that has completed, the user can instruct System Center to perform management duties. As you'll also recall, this setup removes a great deal of control and transparency from the owner of the managed system.

Now consider how you would achieve the same goal without taking anything away from system owners. In other words, how do you involve the target in the process of authenticating and authorizing the user instead of the service account?

Traditionally, we used tools like MMC-hosted snap-ins to achieve this. The tool operates on the manager's system and connects directly to the targeted host. That host can ask Active Directory to authenticate the user and check that account against its own permissions list. It does not need to trust that System Center or any other non-authoritative system has performed the proper checks.

As we already know, MMC has its own limitations. We can, and do, use PowerShell Remoting to solve a lot of remote control problems. That doesn't fix everything, since PowerShell can also encounter double hop barriers. Additionally, the power of command line tools does not overlap the strengths of graphical tools, so we need a visual control plane anyway. Historically, administrators have gotten around these restrictions with remote console tools such as Remote Desktop, VNC, LogMeIn, and others. While those resolve the immediate access concern, they introduce issues of their own, especially around security and scalability.

Windows Admin Center in gateway mode acts like System Center in that it operates as a central management system, but it does not utilize service accounts. Visualize that process:



In the above diagram, Windows Admin Center passes on the authentication status of the user that wants access. That's the "second hop".

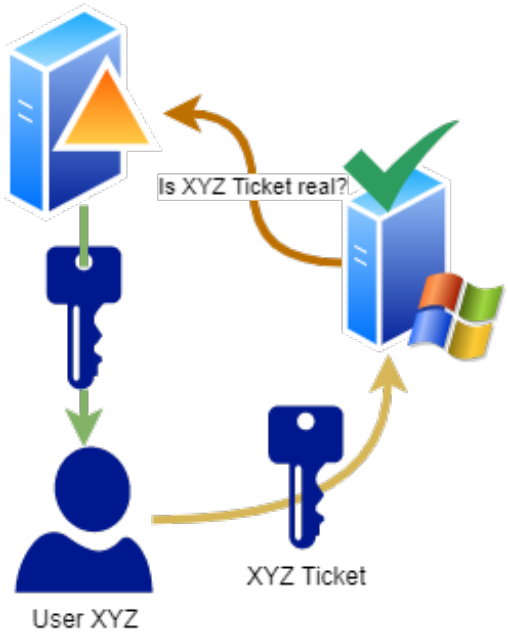
While the second hop provides convenience, it also presents a security risk. We don't want arbitrary systems to go around pretending to be administrators. So, Windows prevents this behavior by default. Unfortunately, most services do not report when you need the second level of authentication. They usually issue the same sort of "access denied" messages that they return to users that do not have adequate permissions. Those errors cause confusion and frustration for users that have administrative credentials. The error was meant for the computer account that could not delegate user credentials, not the user, but rarely do errors mention the account.

If you install and operate Windows Admin Center on your desktop, you skip the double hop problem entirely. You also lose all the benefits of centralized management. Alternatively, you can directly enable the second hop with delegation. The next section explains how.

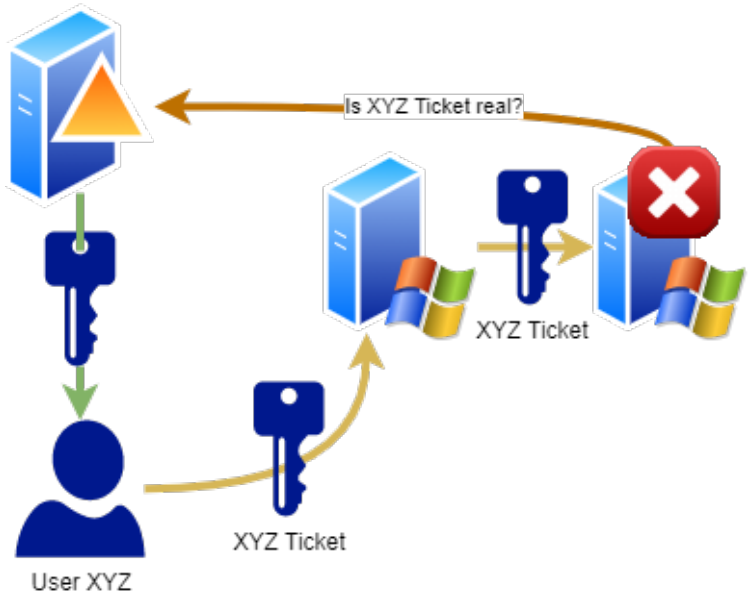
CREDENTIAL DELEGATION

Within an Active Directory domain, we can use delegation to enable gateway-mode Windows Admin Center to manage systems on behalf of a logged-in administrative account. This transparently clears any barriers due to a disabled second hop.

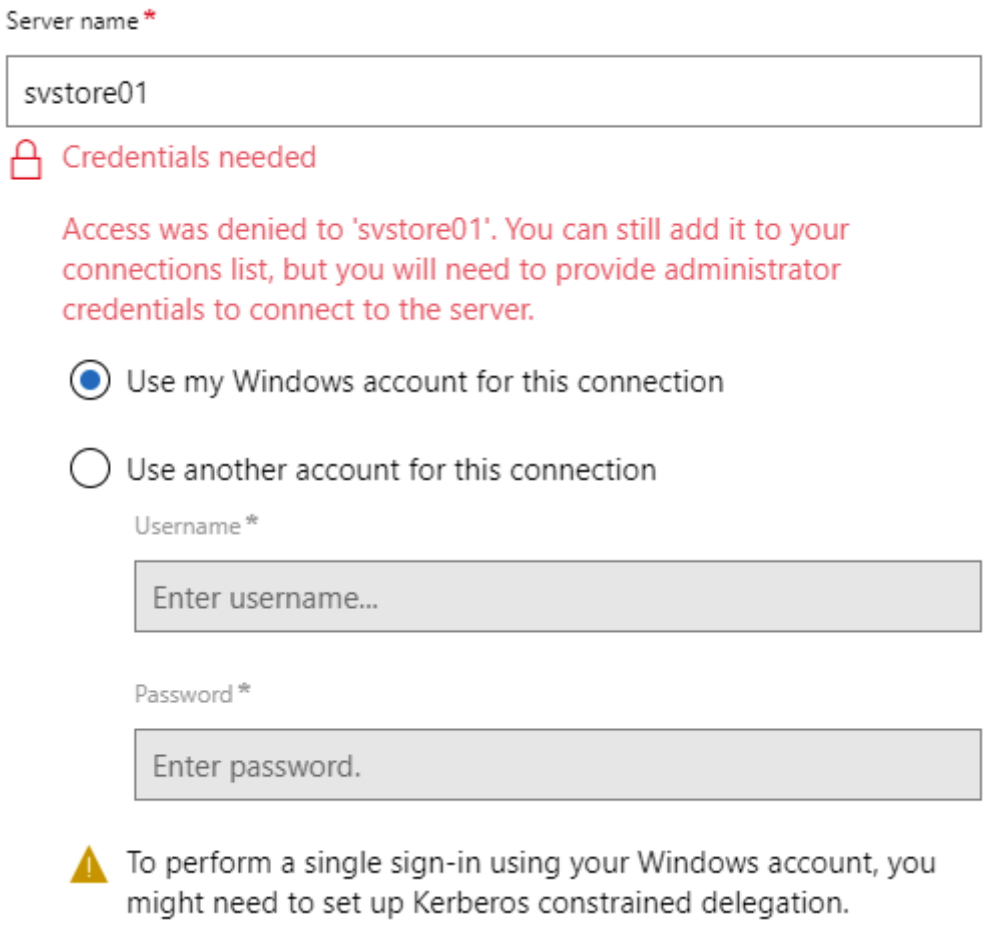
Understand that Microsoft disables delegation by default because it creates a security risk. Active Directory issues a Kerberos Ticket Granting Ticket (TGT) to a user account when it logs in. When the user attempts to access a domain-secured resource, they must present a domain-issued ticket in the authentication step.



With delegation disabled, the receiving system cannot do anything with the ticket except verify it with a domain controller. If it attempts to use it for something else, any domain controller will deny the request, resulting in the "access denied" message that so often frustrates administrators.



Windows Admin Center reports the access denied message and suggests delegation:



With delegation enabled, the receiving system can present a copy of the user's Kerberos ticket to the next hop in a communications chain. While not normally applicable to Windows Admin Center, hosts can continue forwarding credentials until they reach a system not configured for delegation.

SECURITY RISKS WITH DELEGATION

For delegation to function, a host must have the ability to do more with a user's ticket than verify it. To enable that, the host retains a copy of the ticket in its own memory space. With its delegation powers, it can use that ticket anywhere the domain allows it. Therefore, if an attacker compromises a delegating host, they have access to all Kerberos tickets that it currently has in memory.

Kerberos tickets have a limited lifetime, so you have a bit of defense built-in. If an attacker compromises an administrative ticket, then they can work around that. Take these steps to further mitigate your risk:

- Classify systems with Windows Admin Center as vulnerable and high-risk
- Do not use domain or enterprise admin accounts on any host that runs gateway-mode Windows Admin Center (including operations and services besides WAC)
- Do not use unconstrained delegation

The first two points explain themselves. Unconstrained delegation means that a host can impersonate any user for which it has a Kerberos ticket anywhere in the domain for any purpose. Security researchers have demonstrated how

an attacker can gain control over an entire domain by compromising a host with unconstrained delegation.

Constrained delegation limits the attack surface of delegation. You can restrict the hosts and services that a system can use with delegated credentials. For Windows Admin Center, Microsoft has only instructed that we limit by host. They do not indicate that we can limit services. Because of that, constrained delegation may not help much. If you want your domain administrators to use WAC to manage domain controllers and you configure constrained delegation to allow it, then you have the same effective risk as unconstrained delegation.

Despite the risks, credential delegation remains the safest way to operate Windows Admin Center. Architect your deployment with the dangers in mind and protect WAC hosts accordingly. Management systems always present some kind of risk like this, so WAC does not operate outside the bounds of expectation.

HOW TO ENABLE CONSTRAINED DELEGATION FOR WINDOWS ADMIN CENTER

The Active Directory Users and Computers snap-in does have a dialog page for configuring system delegation. However, it does not function appropriately for WAC's needs. Instead, use the following PowerShell line on any system that has the Active Directory PowerShell module (copied from Microsoft Docs):

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

In the above script, **node01** refers to the computer to which WAC will try to delegate credentials. **wac** refers to the system that runs WAC in gateway mode.

If you have a list of servers to configure, then you can use PowerShell's pipeline to configure them all at once. If you have saved them in a comma-separated values file called "wactargets.csv" which lists their computer names in a column named "name", the following line will work:

```
Import-Csv -Path .\wactargets.csv | foreach { Set-
ADComputer -Identity (Get-ADComputer
($_.name.Substring(0, $_.name.IndexOf('.')))) -
PrincipalsAllowedToDelegateToAccount (Get-ADComputer
wac) }
```

You do not necessarily need to operate this as a single line. It was shown this way as an inline edit to the previous single-host line. You can save the contents to a PS1 file and modify it as suits. Note that the **Substring** and **IndexOf** components remove any domain information from a DNS name if present and do nothing to short names. The **Identity** parameter of **Get-ADComputer** only operates on the short AD name.

Note that as shown, these cmdlets *set* the delegation property, meaning that they do not respect any existing setting. That can cause problems when you have a clustered WAC system, multiple WAC systems, or other delegation configurations that have nothing to do with WAC.

If you want to set multiple delegations at once and do not care about the existing setting, modify the **PrincipalsAllowedToDelegateToAccount** from the single **Get-ADComputer** component to multiple calls:

```
… (Get-ADComputer wac1), (Get-ADComputer wac2), (Get-ADComputer hawac) …
```

If you want to leave the current setting as-is, then you will need a more complicated script. You can retrieve the existing delegation setting for a computer like this:

```
Get-ADComputer -Identity targetcomputer -Properties 'PrincipalsAllowedToDelegateToAccount'
```

If the named computer exists, that cmdlet will return an object with a **PrincipalsAllowedToDelegateToAccount** property as an array of strings with the full LDAP path of the systems that can delegate. There is no single correct way to use this information. A suggestion:

```
$NameList = Import-Csv -Path .\wactargets.csv

foreach($ComputerName in $NameList) {

    $ShortName = $ComputerName.name.Substring(0, $ComputerName.name.IndexOf('.'))

    $ADComputerAccount = Get-ADComputer -Identity $ShortName -Properties 'PrincipalsAllowedToDelegateToAccount'
```

```
    $DelegateList = New-Object -TypeName
System.Collections.ArrayList

    foreach($CN in
$ADComputerAccount.PrincipalsAllowedToDelegateToAccoun
t) {

    $OutNull = $DelegateList.Add((Get-ADComputer -
Identity $CN))

    }

    $DelegateList.AddRange(@(

        (Get-ADComputer -Identity 'hawac'),

        (Get-ADComputer -Identity 'wac1'),

        (Get-ADComputer -Identity 'wac2')

        ))

    Set-ADComputer -Identity $ADComputerAccount -
PrincipalsAllowedToDelegateToAccount $DelegateList

}
```

Because **Set-ADComputer** will ignore duplicates, you can safely run the above repeatedly. You can make many useful modifications to this script, such as allowing the user to input a path for the filename, removing old entries from **$DelegateList** after populating it, etc.

Changes to delegation take effect in the directory immediately and retrying any operation within WAC will cause it to initiate another credential check. However, replication may require you to wait a bit.

DELEGATION FOR HIGHLY AVAILABLE WAC DEPLOYMENTS

Microsoft has not published any official guidance on configuring delegation for highly available WAC deployments. Through trial and error, it appears that you must allow delegation for the computer account of the WAC access point and each node individually. It does not require delegation for the cluster's primary name object.

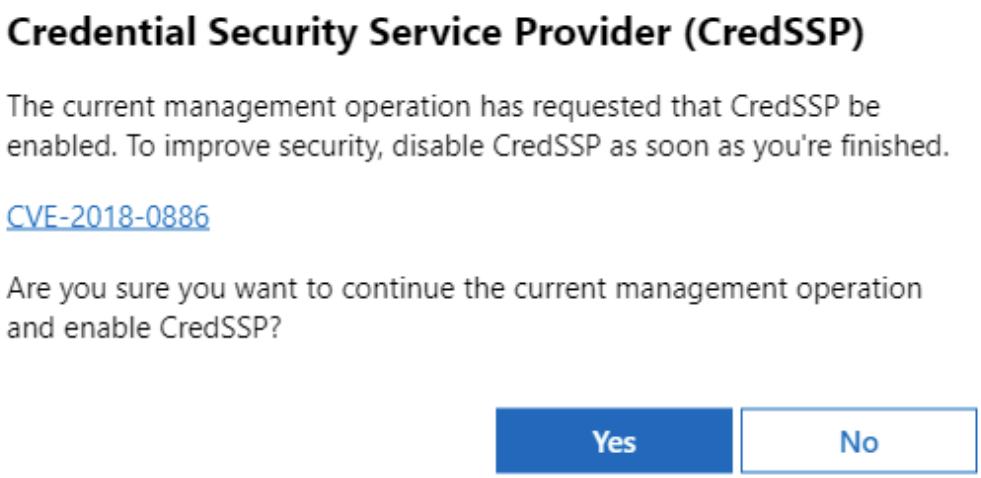
Review the sample script in the preceding section for an example of using multiple different computer accounts. If you use **Get-ADComputer** to retrieve the accounts first as that script shows, remember to use the short name of all objects, not a fully qualified domain name. If you use **Set-ADComputer** directly, then the **PrincipalsAllowedToDelegateToAccount** field will accept short names, fully qualified DNS names, and full LDAP paths. Operations seem to work best with LDAP paths.

For some reason, WAC in high availability seems to have some trouble detecting changes in delegation. If you use Failover Cluster Manager to shift the role around, it will usually sort these problems out.

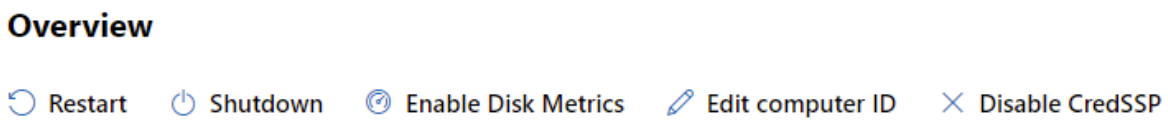
CREDSSP

CredSSP is one of Microsoft's security providers that has some power to avoid second hop problems. It is less secure than delegation and has other limitations. Much could be written on the topic, especially warnings to limit use. CredSSP does not have a large presence in the topic of Windows Admin Center, so it will receive limited treatment here. You can find numerous resources on the Internet if you have more interest.

WAC rarely uses CredSSP, but it does use it. Microsoft has indicated that it mostly uses it for operations related to SMB storage. Fortunately, when it requires CredSSP, it asks you first:



It even provides a link to the CVE with a short discussion of the risks of CredSSP. While the dialog seems to leave in the lurch ("OK, how do I disable CredSSP after I'm finished?"), WAC already has a plan. Once you go to the **Overview** of any computer that has CredSSP enabled, WAC places a **DisableCredSSP** button right on the top action bar:



So, while you should take the risks of CredSSP seriously, WAC will do its best to minimize its use and to keep you informed.

WEB SERVICES MANAGEMENT PROTOCOL

Behind the pretty interface, Windows Admin Center must do some gritty work. It uses the Web Services Management protocol for everything. Commonly known as WS-Management or just WS-Man, the Distributed Management Task Force controls it as an open standard protocol. As its name implies, its primary purpose is for managing things over the web. That protocol uses 5985 for unencrypted communications and port 5986 for encrypted communications, hence the need to open those ports to successfully use Windows Admin Center.

You do not need to know much about WSMan to operate Windows Admin Center, so we will not delve into it. You can find no shortage of public information. However, it does present a security risk that you should know about.

If you have ever run the **Enable-PSRemoting** cmdlet or **winrm quickconfig**, then you have touched WSMan. Those items modify local settings so that the WSMan activities expected on a typical corporate network can function. PowerShell has a WSMan provider so you can poke around in the settings like a hard drive. Just as you would switch from C: to D:, you can switch to WSMan:

```
cd wsman:\
```

You can then move through its "directories". At the top level, you have localhost, below that you have a few settings and some subcontainers. To skip right to the part of interest in this book, run the following:

```
dir wsman:\localhost\client
```

```
   WSManConfig: Microsoft.WSMan.Management\WSMan::localhost\Client

Type            Name                             SourceOfValue   Value
----            ----                             -------------   -----
System.String   NetworkDelayms                                   5000
System.String   URLPrefix                                       wsman
System.String   AllowUnencrypted                                 false
Container       Auth
Container       DefaultPorts
System.String   TrustedHosts
```

As you can see, the demonstration system has an empty **TrustedHosts** field. That means that when connecting from this host to another over WSMan, it will not trust the target. From a security perspective, that is a good thing.

When you install Windows Admin Center, it modifies the **TrustedHosts** field to trust all computers. If you're looking, you'll see an asterisk (*).

```
TrustedHosts        *
```

Don't panic, at least not yet. There are two pieces of good news:

- This change only occurs on the system operating Windows Admin Center, not any of the systems that it manages
- It only applies to client communications, which means traffic initiated by the WAC host

Also, since you should already treat your WAC systems as vulnerable, this should not change anything in your security stance. Do do need to understand what it means and why it's here, especially if a security audit flags it.

When you initiate a WSMan session, your computer blindly reaches out to whatever endpoint you instruct. You may have provided a name or an IP, but that means nothing. If your machine can identify the target machine in some way, such as with Active Directory, then it connects. If it can't, then it breaks the session.

So, what happens in a case where you as the administrator have faith in the reliability of the connection information that you provided but it doesn't convince WSMan? That's where TrustedHosts comes into play. If the computer name presented by the target system matches an entry in TrustedHosts, or if TrustedHosts contains an asterisk, then WSMan acts as though the system provided a glowing recommendation.

Now, you know why TrustedHosts exists. It is how you get through to workgroup-joined systems using only a username and password combination. Next, the risk.

Because you use Windows Admin Center as a management tool, it will send all kinds of credentials to targets. If an attacker compromises one of them using a low security context (like a regular user account) but finds a way to spy on communications, then they have those credentials. You always risk something like that.

But, in a domain, WSMan has Active Directory to vouch for remote systems. It can identify them definitively, so it won't even look at TrustedHosts. In situations where it needs to use that list, you open the door a bit wider to potential abuse. Breaking into a system with a reduced security context and spying on activities in a higher context is difficult. Impersonating a workgroup-joined computer on a network is not. So, if someone successfully masquerades their computer as one that you normally manage, then you might send your credential sets right into a session where the attacker has administrative privileges. Since in this scenario it is not a domain-joined computer, then you're probably sending packets that include real user names and passwords.

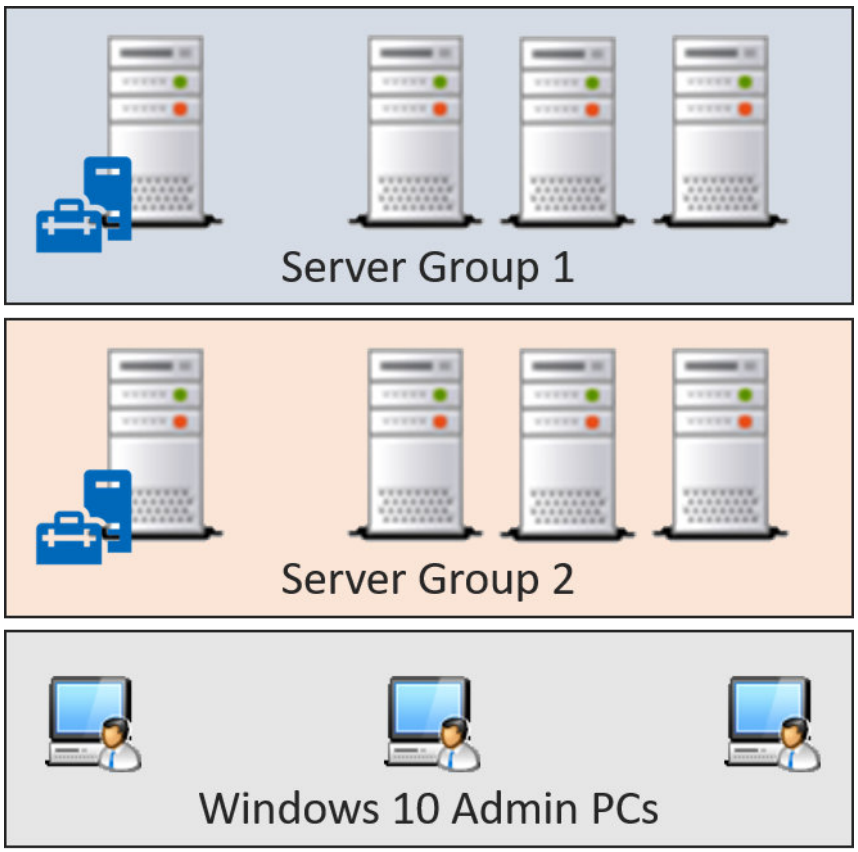
Again, the purpose of this explanation is education. You only need to know that the risk exists. Continue treating security seriously, and this will never be one of your higher concerns.

DIVIDING WINDOWS ADMIN CENTER ENVIRONMENTS

After consuming all the information about security and Windows Admin Center, you may conclude that you cannot adequately provide the necessary access via WAC to some users while scoping others. You might also feel that a mature

WAC installation has become too unwieldy for your organization. In such scenarios, as well as many other cases, you might benefit from splitting WAC across multiple systems.

Simply install additional copies of Windows Admin Center (gateway or desktop mode) until you achieve your goal.



This approach presents a few obvious risks. Every installation requires management and maintenance. You could easily wind up with overlapping WAC deployments that cause the complexity that you intended them to avoid. Fortunately, because WAC cannot provide a user with more access than their account allows and delegation requires administrator intervention, you can manage your vulnerability surface area.

Multiple WAC installations help especially in multi-domain scenarios. While you can now configure resource-based constrained delegation to cross domains, it still adds complexity and weakness. Microsoft has always intended domains to serve as security boundaries, so every trust and delegation represents a hole in the wall. If you place a Windows Admin Center gateway installation in each domain and only allow it to access resources within its domain, administrators only need DNS entries and port 443 line-of-sight to authenticate from other domains.

Most administrators won't find a great deal of value in splitting WAC installations. Remember that you have it as an option if you encounter a complicated management scenario.

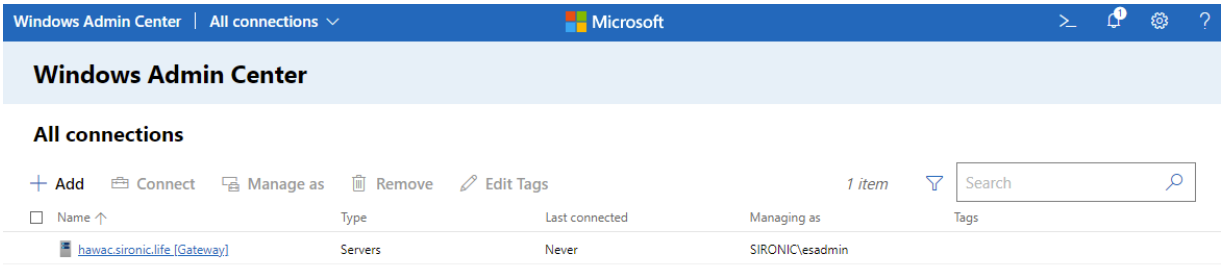
UI BREAKDOWN

Every time you log in an account to a Windows Admin Center installation for the first time, you start with a clean slate. That gives us the perfect segue from a new installation into initial WAC usage.

NEW USERS

When you launch a new Windows Admin Center installation for the first time, it will greet you with a **What's New** dialog and notifications about available updates. You can read through the new items if you like. If you've never used WAC before, then everything is new to you, so these particular items may not mean a lot.

After you've cleared the initial dialog, you see the **All Connections** screen where it lists the local system or the gateway, depending on which mode that you used.

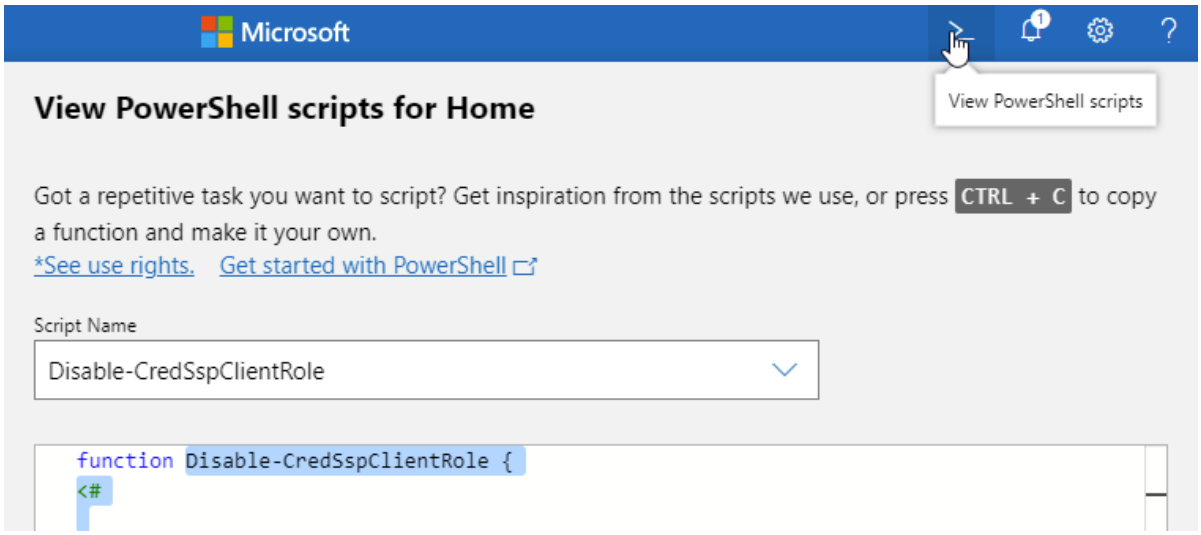


We will refer back to the above screenshot in the next few sections.

THE TITLE BAR

The title bar at the very top of the screen helps you to make large navigational moves. Starting from the left:

- The **Windows Admin Center** text will take you back to the starting page from anywhere. If you get stuck or lost somewhere that you don't want to be, this acts as something like a reset button for the interface.
- The text that currently says **All Connections** shows the current major section in the program. You can use the drop-down to choose another. It acts similar to the **Windows Admin Center** link in that it takes you back to a top-level page, but it allows you to choose which one.
- The **Microsoft** badge links to Microsoft's home page.
- The **>_** icon brings up a side dialog that allows you to view the PowerShell scripts that the current page has available. As the text indicates, you can copy and modify these scripts for your own use.

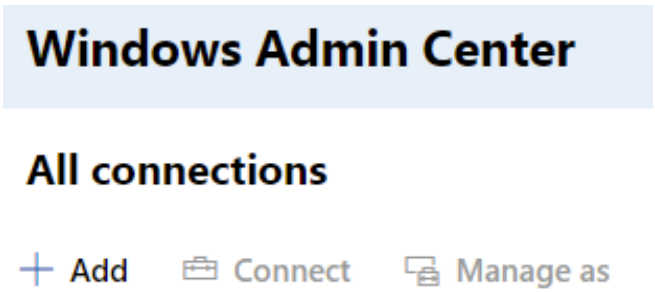


- The bell icon shows all uncleared notifications since you began the session. Notifications have their own contexts; some will take you to other parts of Windows Admin Center, others will let you retry operations, and many provide read-only information.
- The gear icon takes you to the Settings page.
- The question mark icon shows information about Windows Admin Center and provides a few links for more information and help. If you closed the **What's New** dialog prematurely, you can recall its contents here.

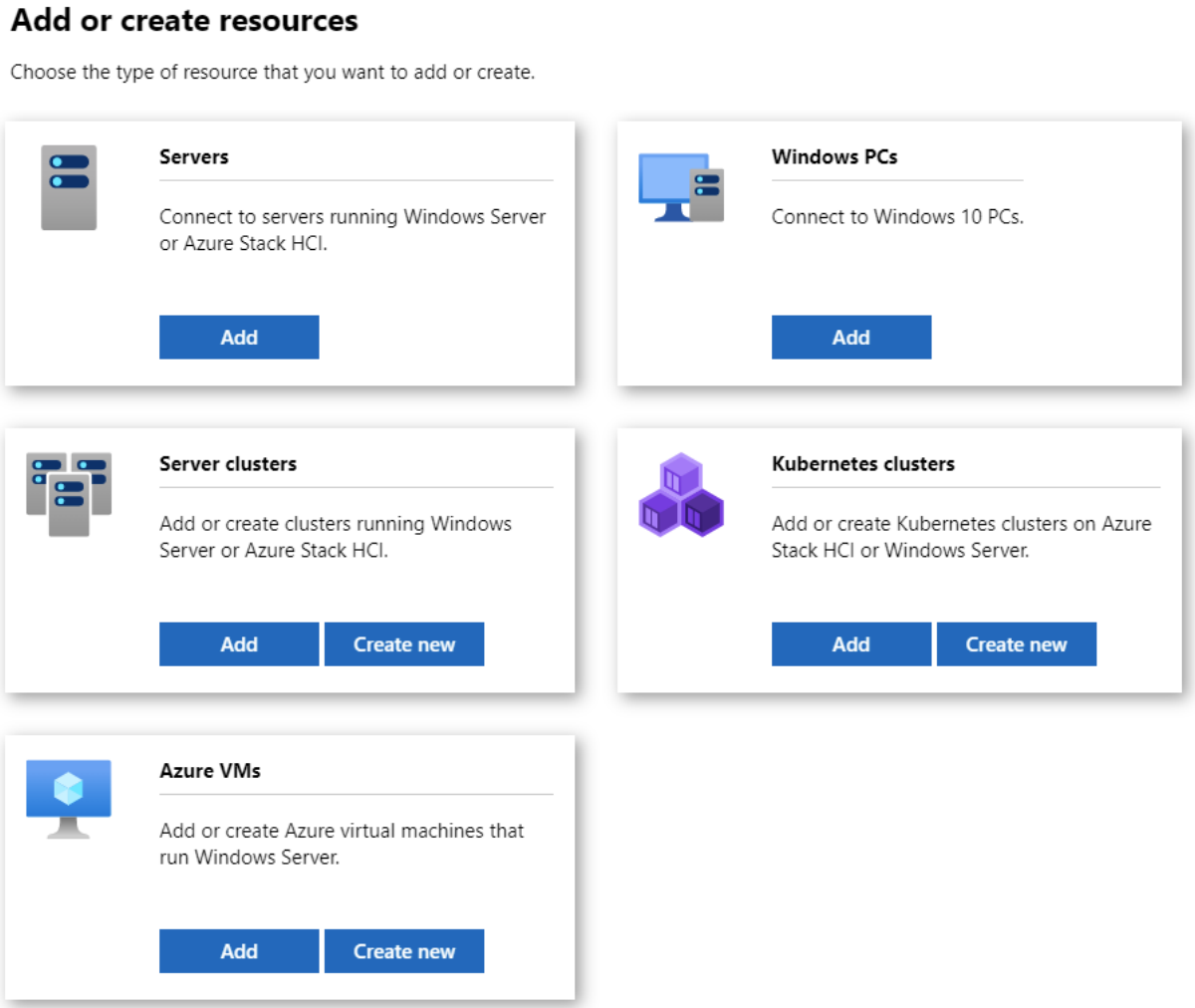
You will use most of these items often enough to gain quick familiarity. Most of the places they take you will appear again in context as you move through the book.

ADDING SYSTEMS, CLUSTERS, AND DESKTOPS

To get any use at all from Windows Admin Center, you need systems to manage. It starts off with only itself. To bring more items to the list, use the **Add** button.



This brings up the **Add or create resources** flyout:

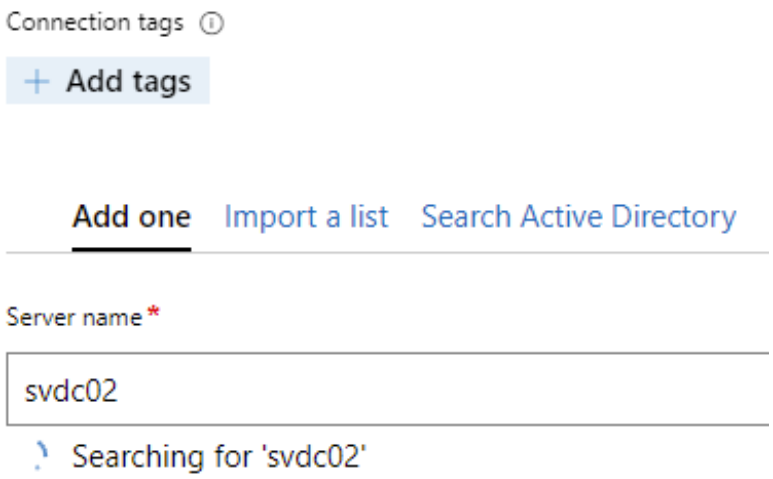


Each item works mostly the same way, with added components related to the item's context (Azure items require selected an Azure subscription, etc.). This book will show the process for adding a server now. Adding Windows PCs and existing clusters looks almost identical. We will show cluster creation in the extensions chapter. We will not discuss the Kubernetes or Azure VMs items further.

Click on the **Add** button in the **Servers** block. You have two or three ways to add systems:

- By name
- From a list
- Searching Active Directory (if the WAC host belongs to a domain)

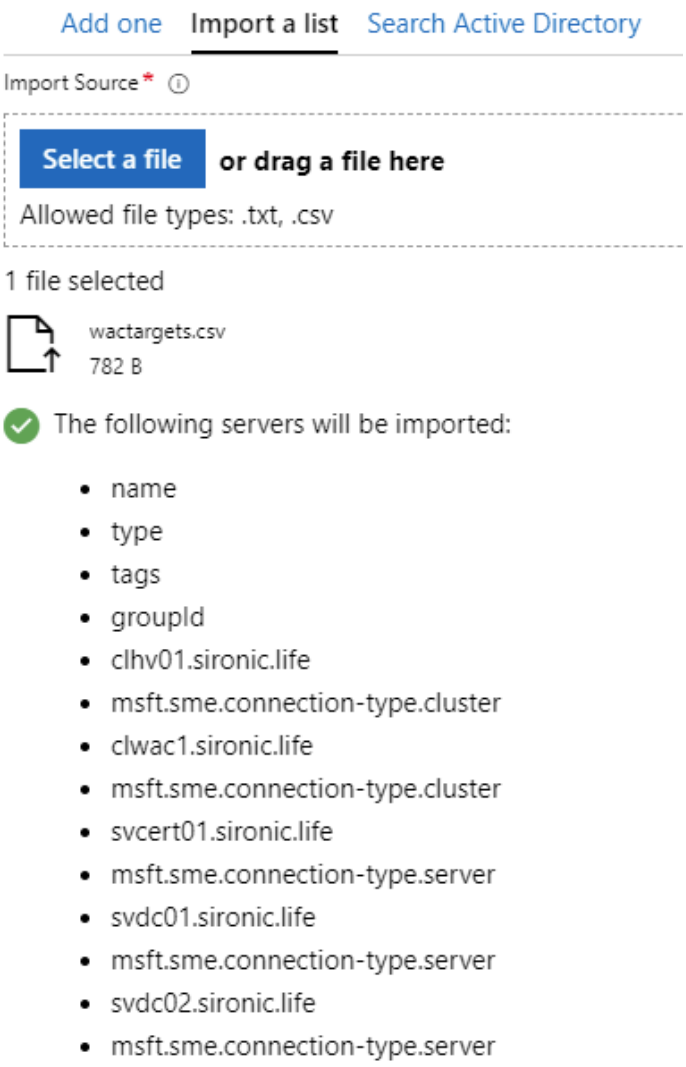
If you use the search block, you must enter the name of the target system identically or it will not find it. When you pause typing, it will start looking across the network using the same sorts of resolution methods that Windows uses for all other network communications.



If it finds the name that you entered, it will tell you that and give you the option to add it precisely as entered. If you don't select that option and WAC has detected a fully qualified DNS name for the system, it will use that.

If WAC cannot find the system that you named, it will give you the opportunity to add an item by that name anyway. It will then appear in the list and you can try to connect any time that you want. You can finish populating your list and wait until later to figure out why the connection failed.

The **Import a list** option lets you pick a source file of one or more computers to add. The tool has very poor parsing abilities. Provide it with a list that has exactly one computer name per line. Anything else will import garbage to sift through. The following shows what happens if you try to use this screen to import a system list in CSV format exported by WAC:



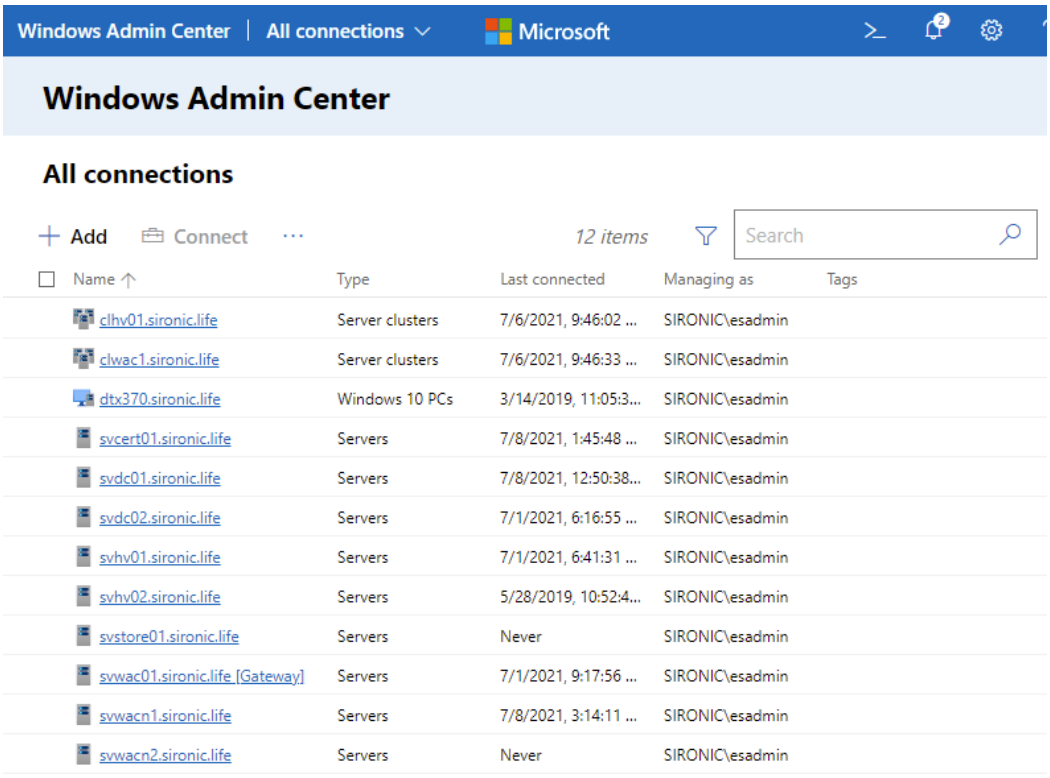
The **Search Active Directory** item works like the **Add one** entry with two exceptions: it looks in the directory instead of using name resolution, and it does not automatically search when you stop typing. You must manually click the **Search** button. Otherwise, it works the same. Spelling counts and partial matches don't work.

Adding PCs and clusters works the same way. The PC dialog looks identical. The cluster dialog gives you an option to manage software-defined networking and does not allow a directory search. Successfully finding a cluster will also cause it to look for member nodes and offer to add them as well.

The next section shows Windows Admin Center with a populated list.

THE SYSTEM LIST

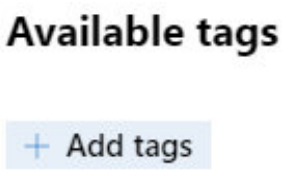
You saw the system list when you first launched Windows Admin Center. It won't take long for you to fill up the screen with entries. Unless you choose a specific item from the connections list, it shows everything:



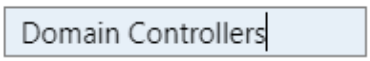
To make the screenshot more readable, it was shrunk. The three next next to **Connect** in the picture open the additional action items **Manage as**, **Remove**, and **Edit Tags**.

You can see a single checkbox next to the **Name** column. As you might expect, that selects all systems in the list. You cannot see that each item has its own checkbox. That is just a convention of the state of the art in web design. Hovering over any list item will cause its checkbox to appear. When you check a box, it will keep a check mark visible on screen even when you move away. The UI uses this behavior for lists consistently across WAC.

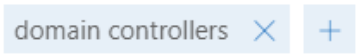
Most items need no explanation other than their name. Tags deserve a brief mention. WAC does not have any form of hierarchical sorting, but you can use tags and filters. To try this out, select a few systems that have some sort of logical grouping and click **Edit Tags**. This brings up a side bar with an **Add Tags** button (and any existing tags).



Type in the name of a tag.

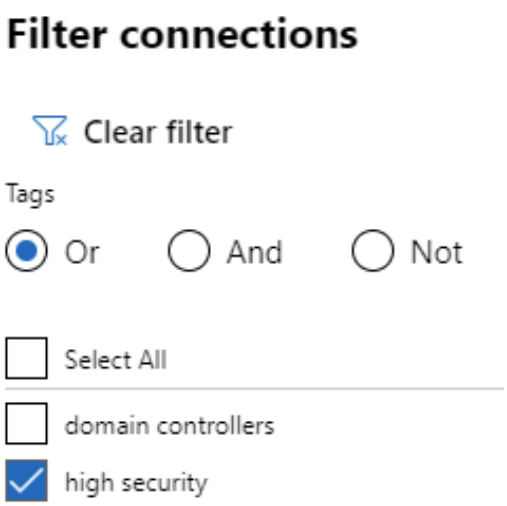


As soon as you move away from the text box, it creates the tag and shows a + button so that you can continue adding more (note that it converts all entries to lower case).



When you click the **Save** button at the bottom, the **Tags** column in the system list will reflect your changes.

You can't do anything with these tags directly from the list. Above the list, click the funnel icon and it will show you a screen where you can customize a tag filter.

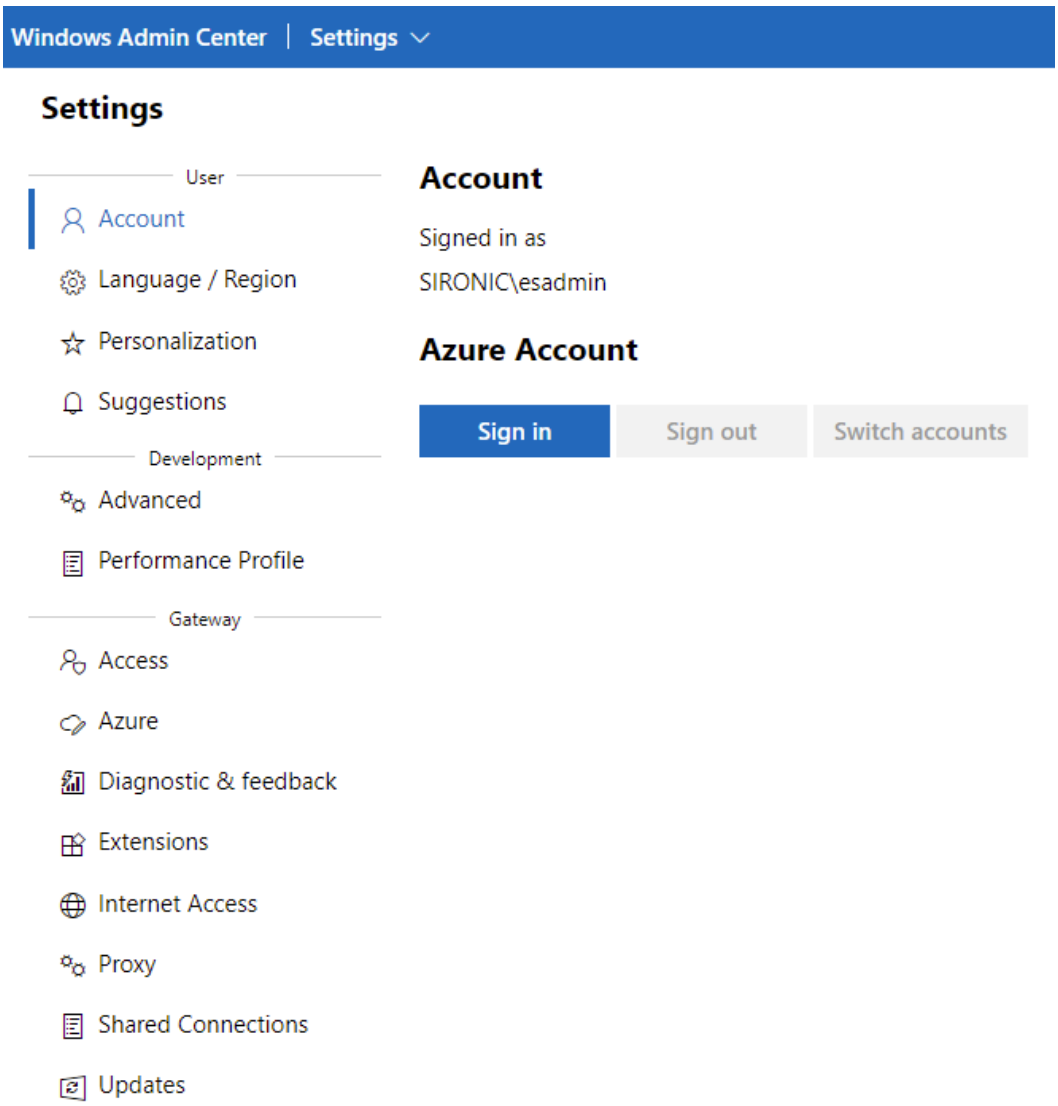


Click **Save** to apply the filter. Return to this screen and use **Clear Filter** to return to the default view of all systems.

That covers almost everything that you need to know about the initial screen. It does have one non-obvious behavior: double-clicking a list item anywhere except its name link will cause it to act as though you had single-clicked the name link.

SETTINGS

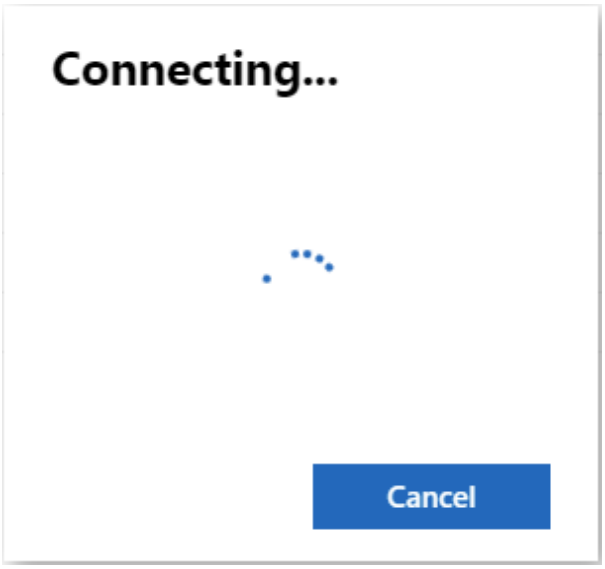
When you click the gear icon at the top right of the title bar, Windows Admin Center takes you to the global settings page. As an administrator, you see settings for your account and for Windows Admin Center. Someone who has a user role within WAC will see only the user options.



Most items don't require explanation. We will go over others in the Settings chapter. Click through the items on the left and look for items that you recognize and want to change immediately.

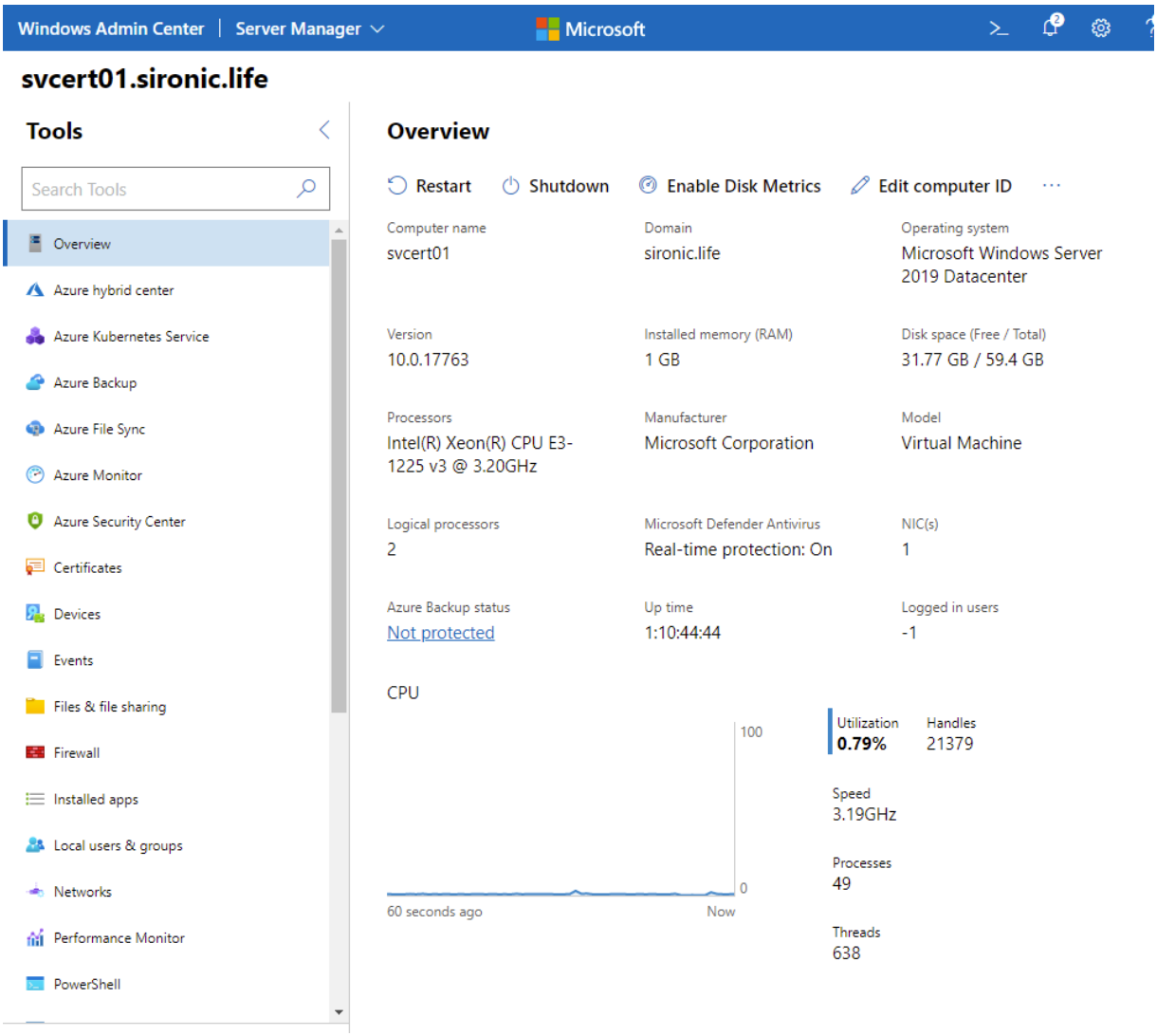
OVERVIEW

When you click an item on the system list, WAC initiates a connection. This typically requires at least a few seconds to complete, so have patience.



The overview page contains a lot of data, charts, and menu items.

Take some time to familiarize yourself with this screen's contents. If you use the **Enable Disk Metrics** button, then you'll see an additional chart for each directly connected drive in the system. Down the left side, under **Tools**, you will see an entry for all extensions applicable to this item. Clicking those will take you to the extensions page where you can interact with its settings. If you have worked in Azure, you will recognize this behavior.



At the very bottom of the **Tools** column, technically below it, you can see a **Settings** button. This has many of the same items as the computer properties dialog in the desktop. As WAC matures, more appear. You can now control SMB1 enablement, Azure Arc enrollment, and more. If you can't find an expected item in the extensions, you might find it here.

The system overview is the landing page for computers, clusters, and desktops in Windows Admin Center. It is the hub for all related activities. You will probably see it more than any other page in WAC.

COMMON OPERATIONS

You will learn Windows Admin Center best by working in it. As you acquaint yourself, you will likely discover a lot of its motifs and behaviors. Of note, if you have experience with Azure, you will find many similarities.

A few things:

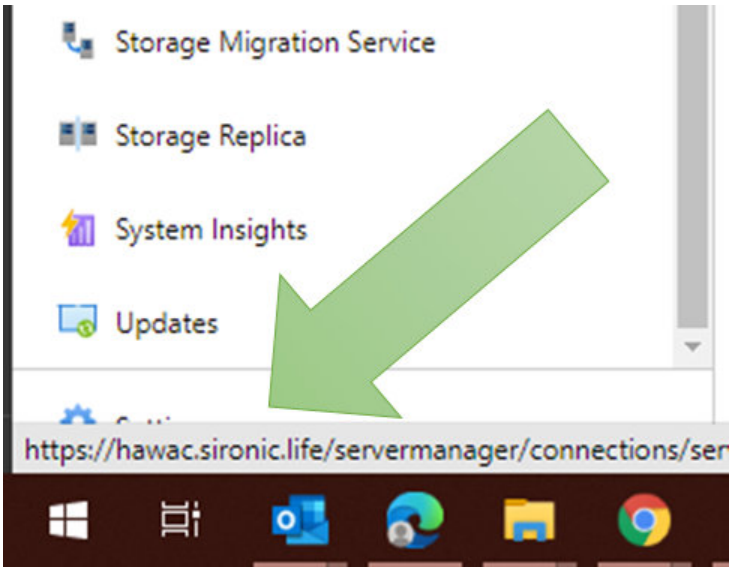
- Right clicks do nothing in Windows Admin Center except open up your normal browser's right-click menu. Acclimate yourself to using only left clicks.
- Clicking on an object typically takes you to a page with more information about that object (e.g., servers, clusters, virtual machines). Clicking on an action usually either starts that action immediately (with a corresponding notification) or opens up an overlay window at the right of the screen.
- Items in left-pane menus usually change the contents of the primary pane.
- The interface restarts automatically for updates. If you have automatic updating enabled or another administrator working concurrently starts an update or changes an extension, that might cause your session to restart unexpectedly.
- Windows Admin Center's security configuration does not especially like multiple tabs or browser windows. Usually they work, but sometimes you get unclear messages in one or more of them.
- Activities that break or end sessions in one tab may cause all other

tabs and windows viewing Windows Admin Center to close.

- Windows Admin Center does not make particularly efficient use of screen real estate. It uses white space as its most common element divider, meaning that screen objects have a substantial minimum separation distance. While this situation has improved (and continues to), you may encounter challenges running Windows Admin Center on a low resolution monitor, in less than full screen mode, or in a screen share application.

- Windows Admin Center does not allow much organization. You can group major objects (servers, etc.) with tags, but you cannot create a hierarchy.

- The placement of the object **Settings** menu item (as opposed to the global **Settings** item) often causes the browser to obscure it. It appears at the very bottom left of applicable pages – right where the browser shows full links. Keep this in mind when hovering your mouse over items, especially as you browse menus.



- Windows Admin Center changes *all the time*. Software releases arrive in a regular cadence and extensions updates almost continuously. Expect at least minor differences every time you launch WAC.

- You will almost certainly encounter bugs, even if you don't choose an Insider build or use experimental extensions. This may change eventually, but it has been a normal part of WAC throughout its lifetime to date.

- Reports and responses often do not behave intuitively. For instance, go to the Extensions page and find one that says that it has an available update. Click on it to highlight it. That will cause the bottom of the screen to expand with details about the extension. If, like many people, you clicked on it because it said that it had an update, you might (logically) expect to find some way to update it in that **Details** pane. However, the **Details** pane was always there whether you noticed it or not. WAC just made it a lot more obvious when you selected an item. The **Update** button appeared at the top of the list when you clicked the item.

- Expect pseudo-advertisements. WAC will point out when some feature of Azure can enhance your current activity and other things that might benefit you while also encouraging you to spend money. You can disable most of these in the **Suggestions** section of the global settings (gear icon, right side of the title bar).

- Sometimes you just have to click it again. If something appears stuck, try again. If that doesn't help, click the **Windows Admin Center** at the left of the title bar or pick something from the drop-down. If that doesn't work,

refresh the page. If that doesn't work, start a new browser session. If that doesn't work, try clearing your cache and cookies or try or private browsing session. If you still can't perform whatever task you keep trying, either the target has a problem or you have found a bug.

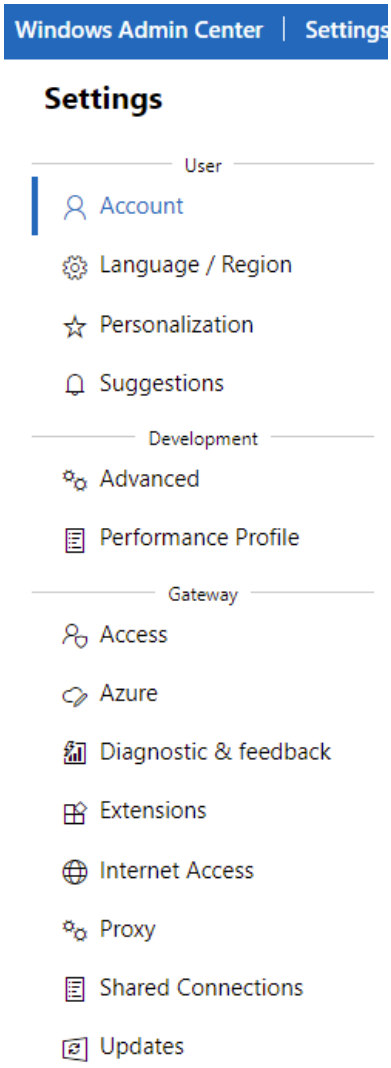
Some of these things seem strange and perhaps a bit obnoxious at first. Like any program, WAC has its quirks. As you learn the software, your workflow will adjust to the oddities and blemishes. Most administrators find that WAC brings more than enough benefit to outweigh its flaws.

Click, explore, find things. The upcoming chapters will continue our guided style, but you will always have something else in WAC to discover.

CONFIGURING GLOBAL SETTINGS

Windows Admin Center has a group of settings that only apply to your logged in account and to the Windows Admin Center installation. You can find these by clicking on the gear icon at the right of WAC's title bar.

If you have a standard WAC user account, then you will only see the user category where you can change settings that apply to you. If you have a gateway administrator account or WAC runs on your desktop, then you also have the **Development** and **Gateway** sections.

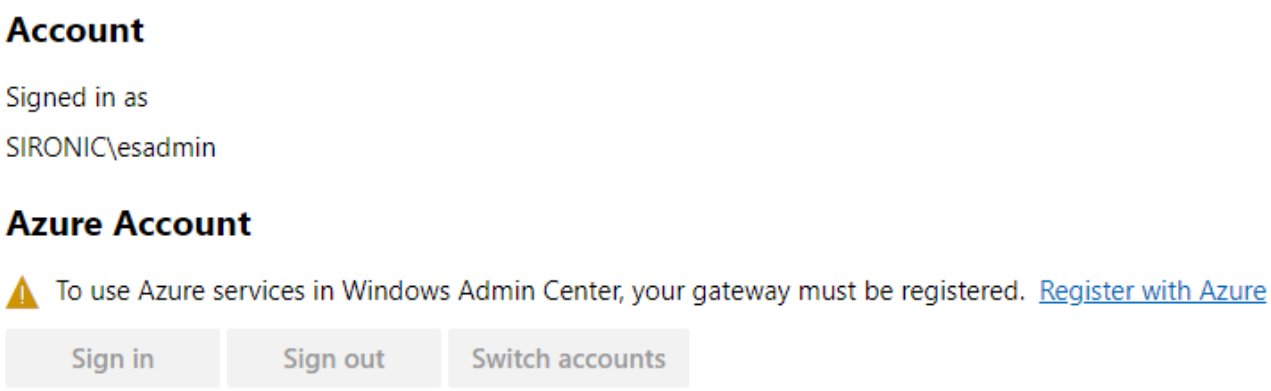


We will not stop on every point because you can easily figure most of it out without help. The upcoming sections in this chapter will cover features that stand out or tend to cause confusion.

USER SETTINGS

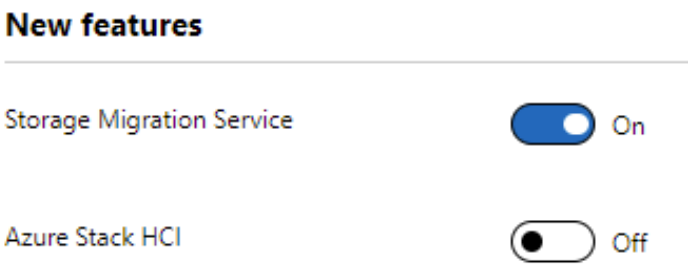
The first tab that you encounter under user settings is the **Account** tab. First, it shows the account that authenticated to Windows Admin Center. You can't do anything with that; WAC has no logout feature.

However, you will also find an Azure heading (for standard users, this may vary). If you have registered your WAC installation with Azure, then you will see the following:



We will cover Azure registration in a dedicated chapter. If you connect WAC to Azure, then the warning line disappears and the the applicable buttons among **Sign In**, **Sign Out**, and **Switch accounts** will light up. These buttons only apply to your Azure sign in and do not impact WAC.

The **Language/Region** and **Personalization** pages have few settings and need no discussion. The **Suggestions** page, however, deserves a bit of attention. A cynic might consider "suggestions" here as a euphemism for "advertisements". WAC can sometimes seem insistent about things that you have no interest in. You can enable/disable most of the pop-ups and notification that have nothing to do with your products or management tasks here. For instance, this WAC installation does not prompt for investigations into Azure Stack HCI:



However, it does still mention the Storage Migration Service. In contrast with any cynical opinions regarding advertisements in WAC, the Storage Migration Service costs nothing more than your existing Windows license. As you peruse the list, you will find plenty of things that apply to helpful suggestions, not paid products.

Because these selections apply per user, they have higher value in larger organizations. If a user has no control over the Azure spend, for instance, it makes sense to disable the **Cost savings** announcements.

Learn more about how to manage Azure Stack HCI in our on-demand webinar on Powerful Windows Server Features.

DEVELOPMENT SETTINGS

We mostly included the development section here for completeness. If you do not intend to develop extensions for Windows Admin Center, then you will not find anything of value here.

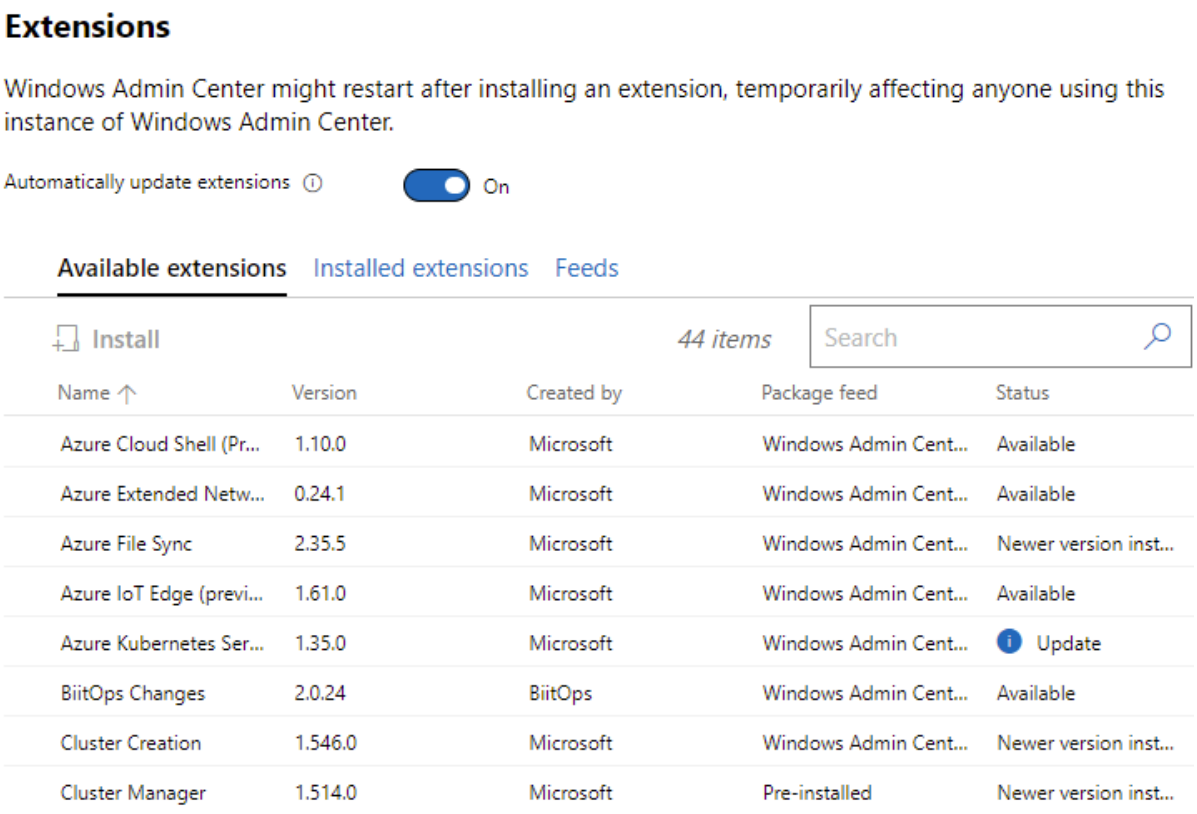
The **Advanced** tab allows you to modify the interface to help with extension development activities. You can also enter **Experiment keys** that turn on experimental features. Look for Windows Admin Center developer resources that can connect you with the product team if you want to dive into these. The features unlocked here mean the most to extension developers. If you want previews of features that you will use as an administrator, then sign up for the Windows Insider program to receive Insider builds of Windows Admin Center.

The **Performance Profile** tab starts tracing and timing activities. Use this to see how quickly your extension performs its operations.

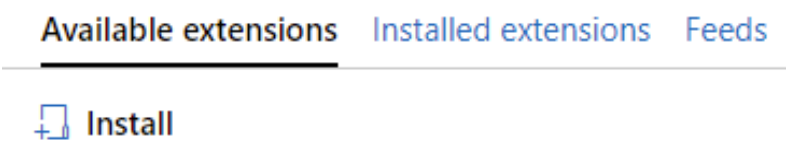
GATEWAY SETTINGS

Windows Admin Center has many settings for its own operations that you'll find under the **Gateway** section in the **Settings** menu. If you want to try to change something about the way that WAC functions, then look here first. It has controls for access, which we covered in the security section. You can also find a point to connect to or see details about an existing registration with Azure. Most of the other fields have similarly simple and self-explanatory contents.

You have two sections that you will use repeatedly. First, **Extensions**.

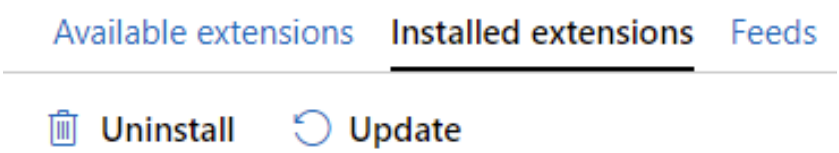


Windows Admin Center is effectively useless without extensions. Here, you can add, remove, and update them. It starts you out with a view of **Available** extensions, which means extensions in the feed that you have not installed or features that have available updates. When you highlight one, it shows some details about it below the list and lights up an **Install** button at the top.

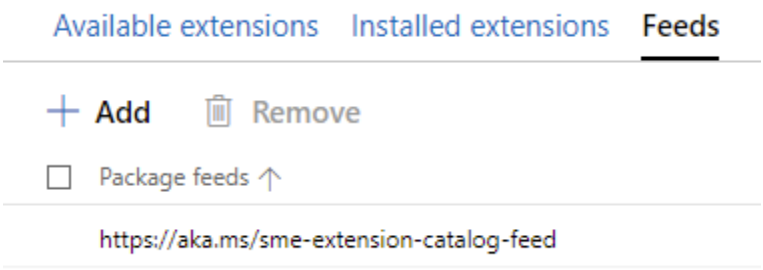


The **Install** button does exactly that. Just know that it restarts all connected sessions every time that an extension installs or updates.

If you change to the **Installed extensions** view, you get a similar list, but this time of installed extensions. You gain access to the **Uninstall** and **Update** buttons that do what you think they do.

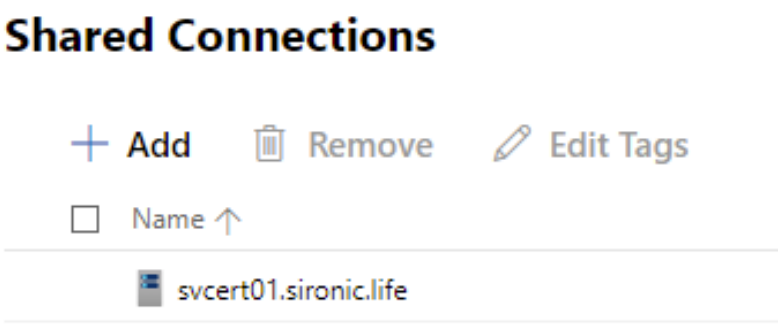


On the **Feeds** section, you see where WAC finds the source for offered and updateable extensions.

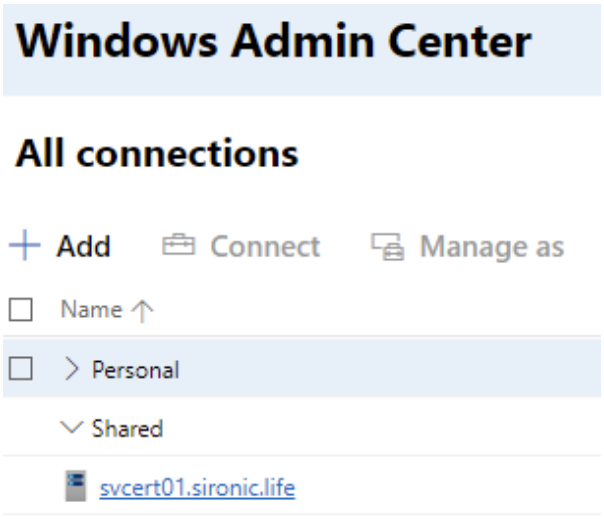


If you find other feeds or your development builds internal sources, you can include them here.

Back out on the left-hand menu, **Shared Connections** presents another point of interest. When WAC first debuted, administrators found it annoying to have multiple people creating almost identical lists separately. So, you can now add systems and they will appear to anyone that logs into WAC. You modify this list exactly the same way that you add and remove items on the main system list.



On the main list, users will now have **Personal** and **Shared** sections:



An item can appear on both lists.

EXPLORING EXTENSIONS

Extensions make Windows Admin Center worthwhile. The defaults give you a great deal of power. Microsoft and third party vendors continually add and update their offerings. While we do not yet have all the control items found in the traditional MMC tools, WAC provides substantially more.

As you look through the available extensions, some may indicate that you need to install a particular role or feature on the Windows Admin Center system first (File Server roles, Hyper-V PowerShell module, etc.) Fortunately, you will find a **Roles and features** extension that allows you to do exactly that.

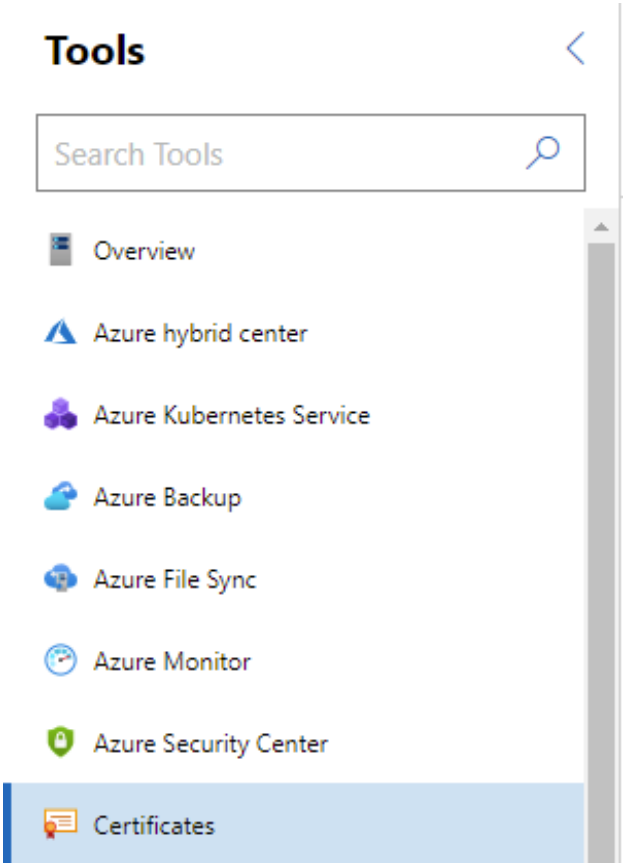
We won't show that particular extension. You won't have trouble figuring out how to operate it on your own. However, we will look at a few default extensions in this chapter to give you a feel for the general behavior of a WAC extension and to provide some points of comparison against older tools.

CERTIFICATES

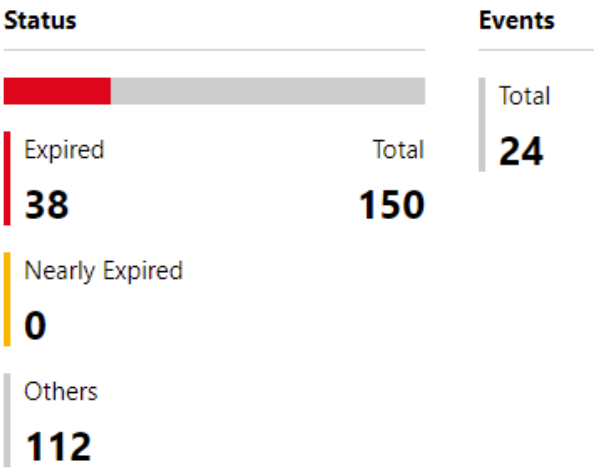
Can you name a common, mature technology that administrators avoid more than PKI certificates? The technology is confusing, the tools are complex, and documentation and tutorials are sparse and dense. If you get certificates wrong, then you potentially make things less secure than if you had just left everything alone. If you get it right, how do you know if you got it right?

Sadly, Windows Admin Center does not fix all our certificate woes. It does, however, make it a bit easier. It even addresses a few pain points found in the traditional certificate management MMC.

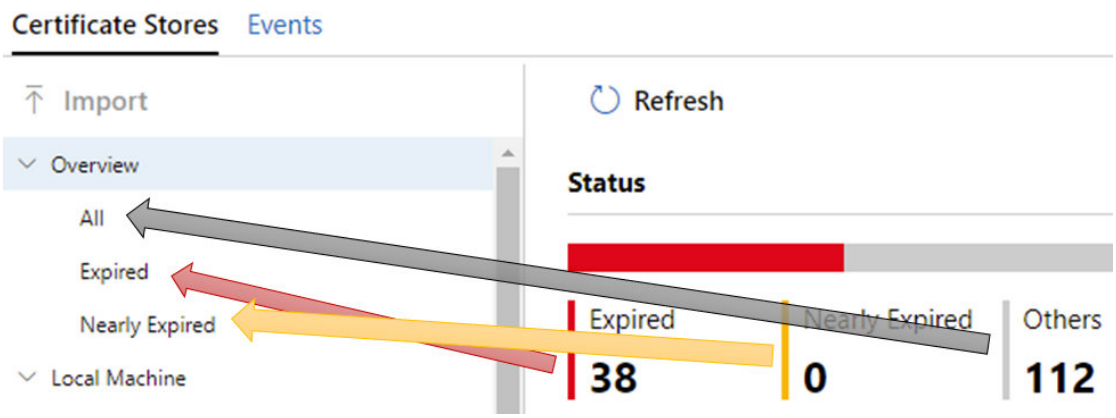
To get started, connect to a system from the list in WAC and click on the **Certificates** extension under the **Tools** menu at the left.



First, you get a dashboard that provides a count of the certificates across various stages of their lifecycle and a count of certificate-related events.



You can click on these items, but they just map to menu items to the left of the dashboard:



These portions make the entire extension worth it. You can, at a glance, discover expiring and expired certificates. With systems like web servers, certificate problems are obvious. In many other cases, they can be difficult to diagnose. This at least reduces the work of finding out if an expired certificate exists.

At the bottom of the above screenshot, you can see the **Local Machine** branch. Indented underneath it are the certificate stores that pertain to the computer. After that, you find the **Current User** branch which does the same for your account on that computer. Unfortunately (or not, depending on your point of view), the extension cannot impersonate other users to work on their store.

Locate a certificate that interests you. Notice how the action items light up:

No matter what certificate you pick, you have only two alternative possibilities: the **Renew** button lights up or it does not. The significance: this happens whether

you or your organization have any ownership of the certificate or not. This can seem a bit jarring, but the MMC tool operates the same way. You can send a request to renew a public CA certificate, just don't expect a response. If you get one, it will almost certainly be something other than a newly issued certificate.

Some things to note:

- **Export** works, but you only get the certificate. You won't get a private key.
- **Renew** only works for certificates still within their valid lifetime.
- **Request** new uses exactly the same interface as **Renew**, but it works for expired certificates.
- Clicking the number of events in the dashboard will take you to the security-related event logs, but you have to drill down to the events. It's still better than scouring the standard event viewer.

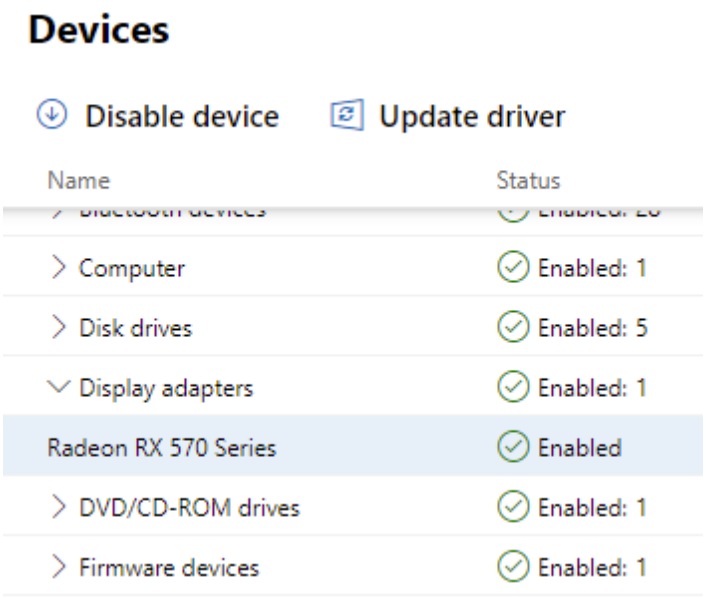
The certificates extension may not make everything in the world of PKI all better, but it beats the MMC substantially. You will not mind the ability to flip between the certificate extension on different systems faster than you can go through the work of connecting and reconnecting MMC snap-ins.

DEVICES

Once upon a time, you could connect to Device Manager remotely for updating drivers and disabling devices. One day, that ability went away and never came back. Whether by coincidence or design, that was around the time that the "Core"

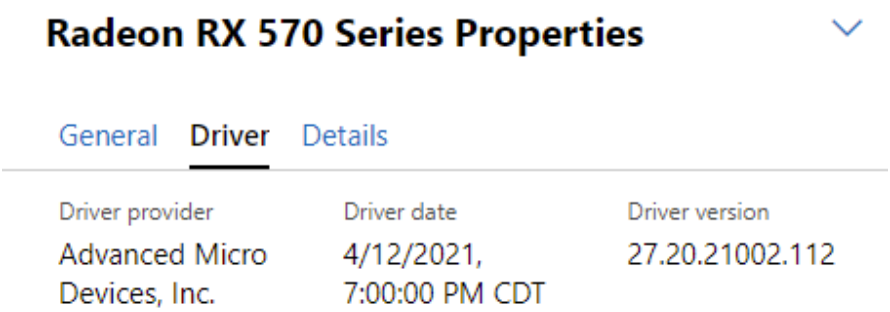
mode of Windows Server and the standalone Hyper-V Server product began to enjoy more than passing interest from administrators. Without the Windows Desktop Experience, those systems could not run MMC at all, we had only pnputil.exe to help us out.

Windows Admin Center ended that problem neatly. Connect to a system and access the **Devices** entry in the **Tools** menu and behold:



You can enable, disable, and update the devices on a remote system again.

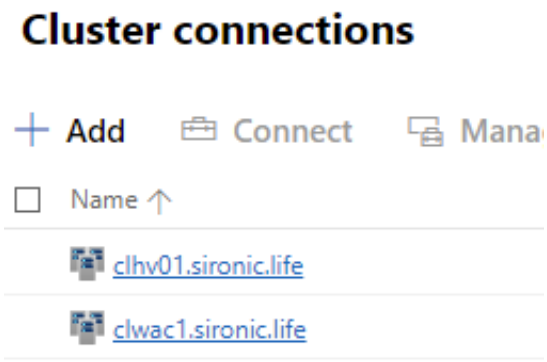
The lower section provides some details about the selected device:



Despite the wonder of WAC, we still lack a bit of capability. Some devices more settings that we might want to access. Fortunately, most manufacturers have moved those out of Device Manager and into something else. Overall, we happily welcome this extension.

FAILOVER CLUSTERING

Unlike other extension, you access the cluster-related extensions by selecting a cluster object from the system list. Clusters appear with a special icon and you can switch from the default **All connections** to **Cluster Manager** to filter to only your clusters.

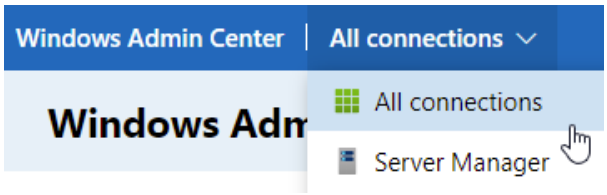


We will start this section by using Windows Admin Center to create the very cluster that was used for the clustered WAC demonstrations in this book. Interestingly, perhaps as an oversight, you cannot create a new cluster when you click the **Add** button in the Cluster Manager view. You can only add connections to existing clusters. We'll start with locating the correct option.

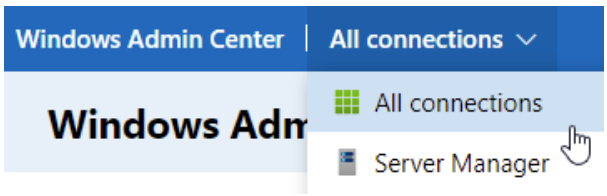
CREATING A CLUSTER

Follow these steps to use Windows Admin Center to create a failover cluster. During installation, you may receive a notice regarding CredSSP. If so, you must allow it in order to create the cluster. You can easily disable it later by accessing the Overview page of each node.

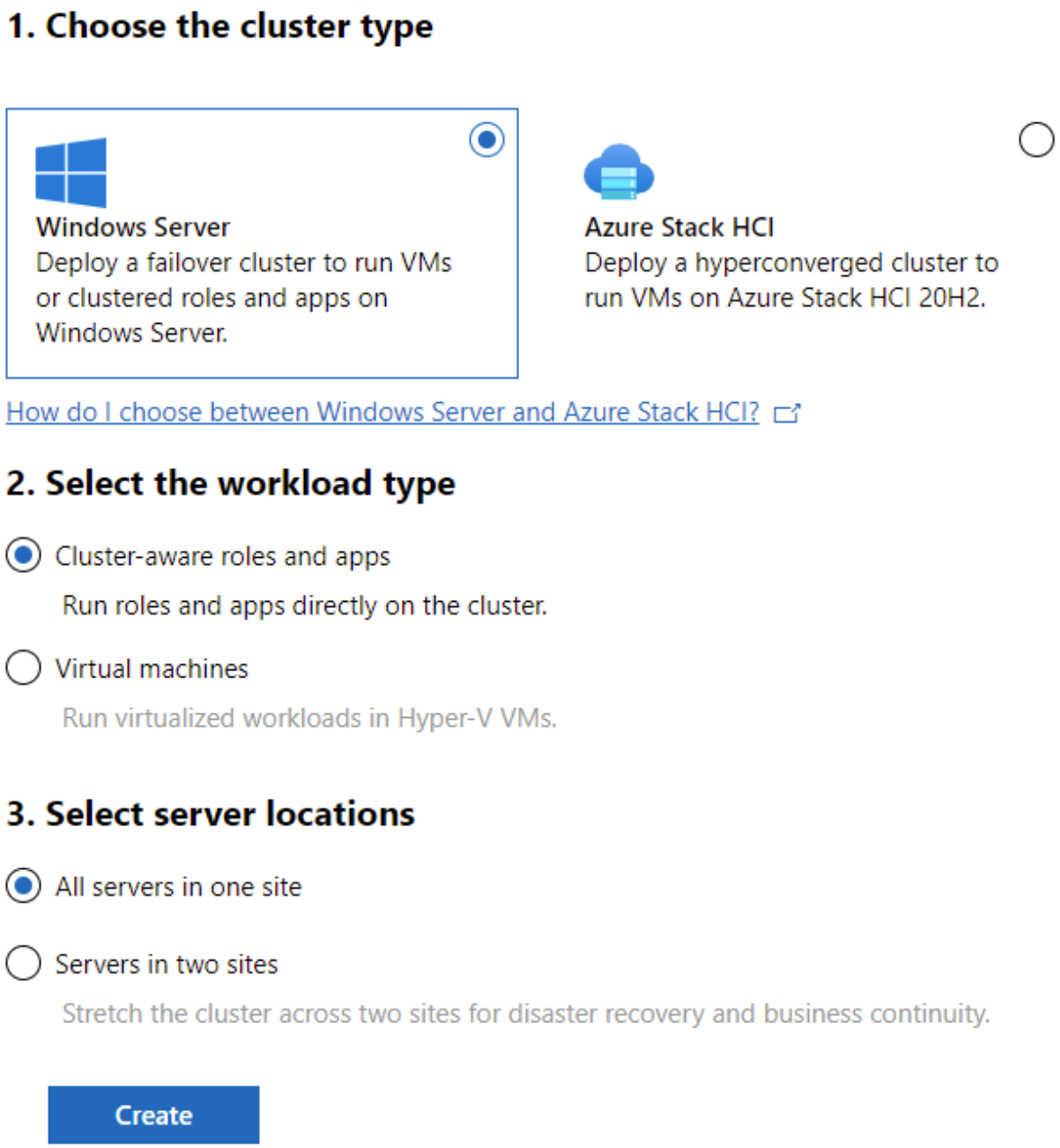
1. Return to WAC's home page and the **All connections view**.



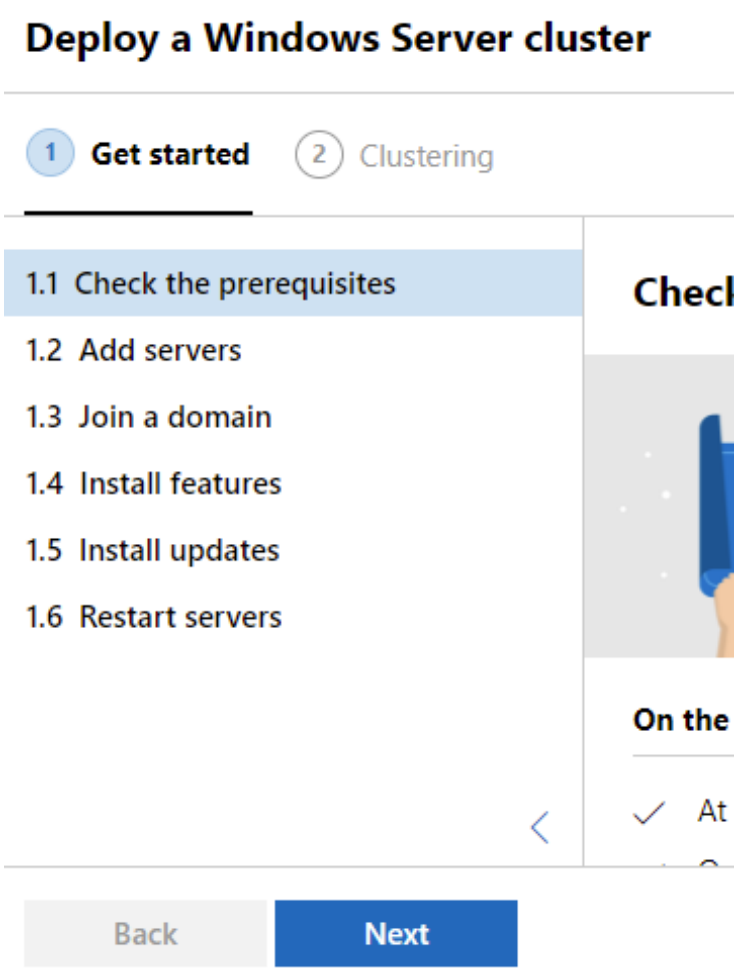
2. Click **Add**. In the **Server clusters** square, click **Create new**.



3. For this walkthrough, choose **Windows Server** as the cluster type, **Cluster-aware roles and apps** for the workload, and **All servers in one site** for location. Click **Create**.

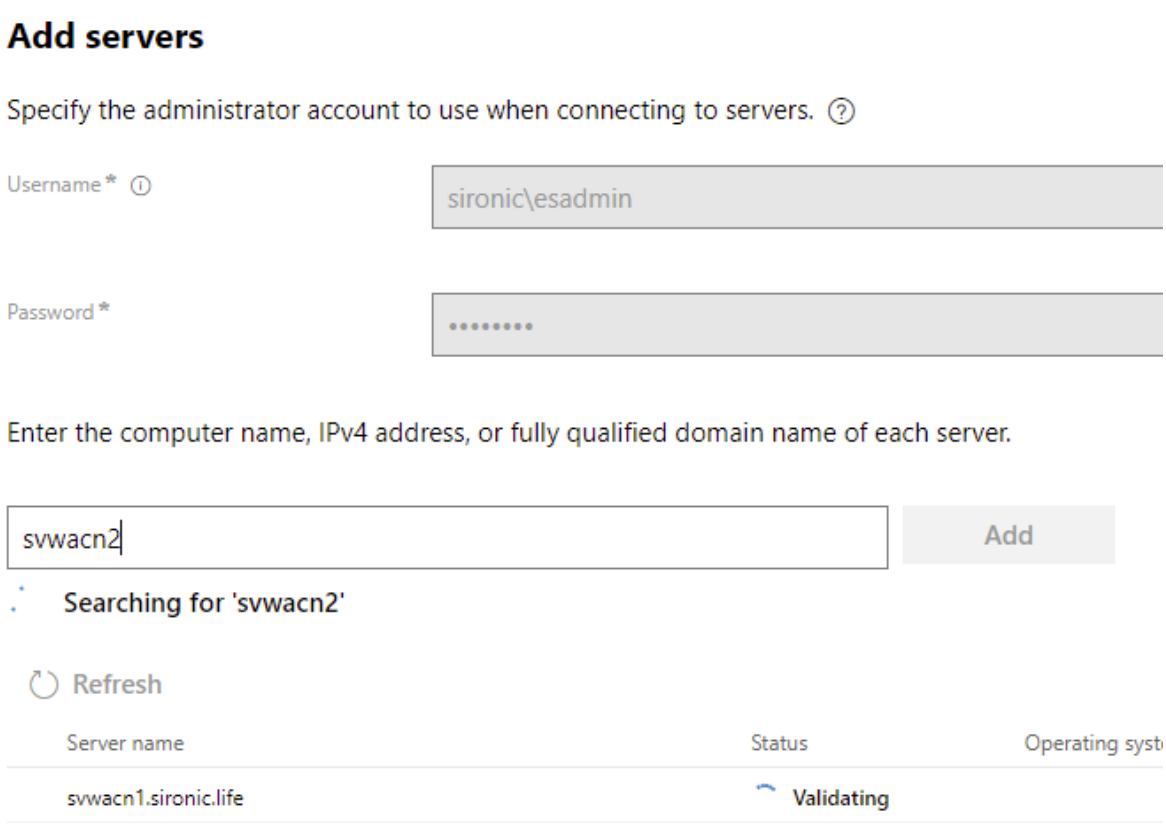


4. You get something like a wizard-in-a-window. You can jump forward and backward using the items on the left.

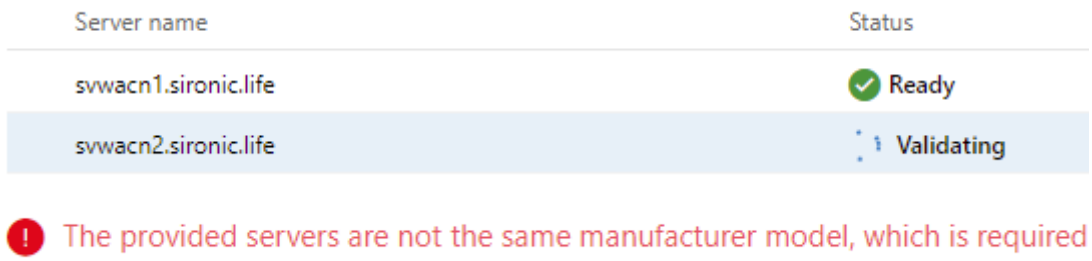


5. Run through the prerequisites checklist and ensure that you have gotten everything ready before proceeding.

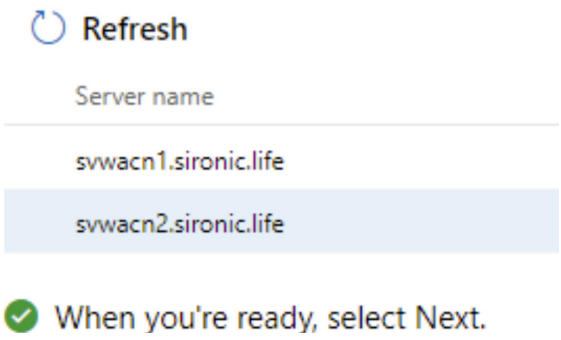
6. The **Add servers** screen looks much like the corresponding screen in the Failover Cluster Manager MMC, with the exception of the credential entry. In the lower part of that pane, search for servers by name or IP and **Add** them as the tool discovers them. If it can't find one, fix the connectivity problem or add it to the cluster later.



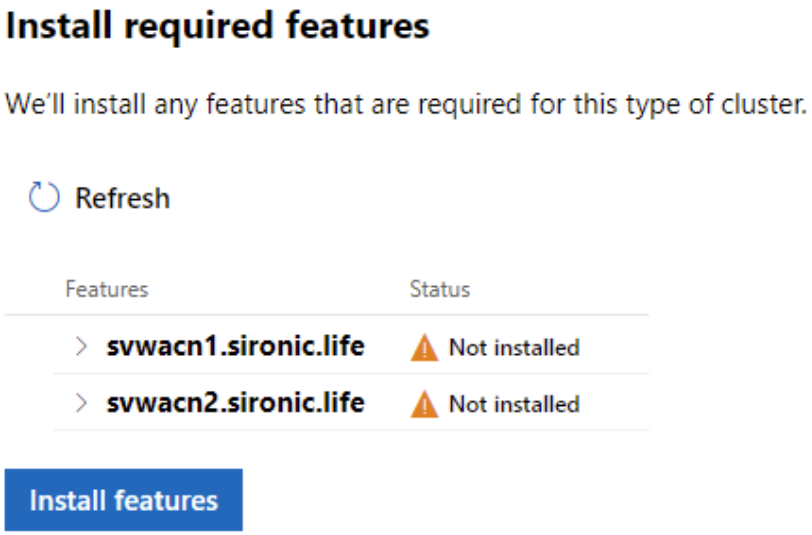
7. The tool will validate as it adds the nodes. This is not a complete cluster validation. It only finds out enough to know that the node does not have any immediate showstoppers. Be aware that if it validates one or more nodes while still checking others, it will display errors that it cannot know to be true. Just wait until validation completes.



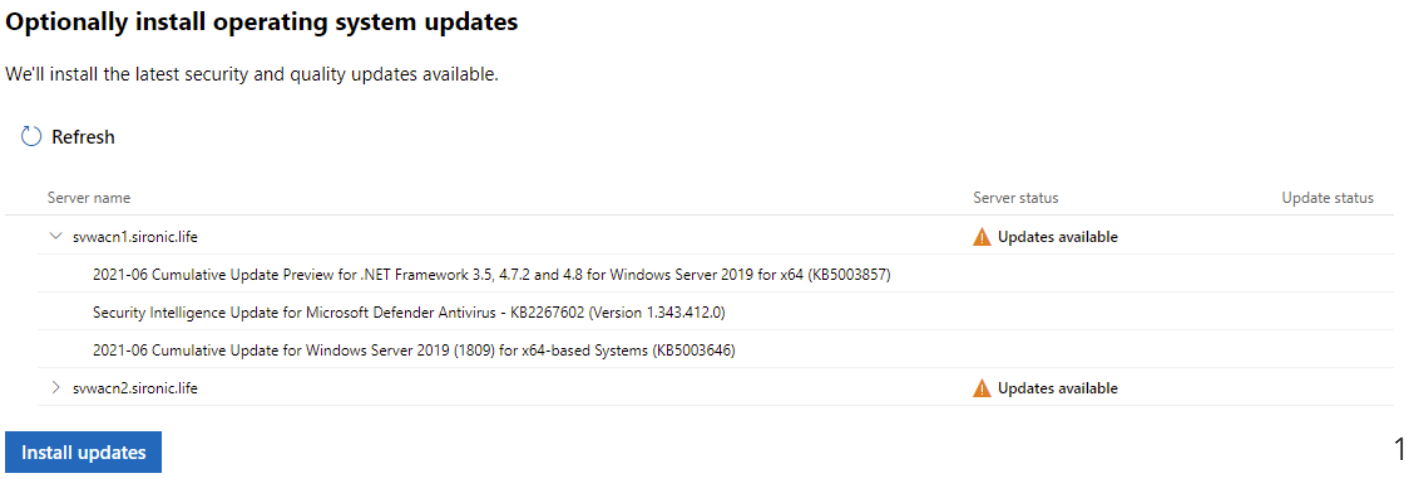
8. Any prematurely reported errors will clear once all systems validate.



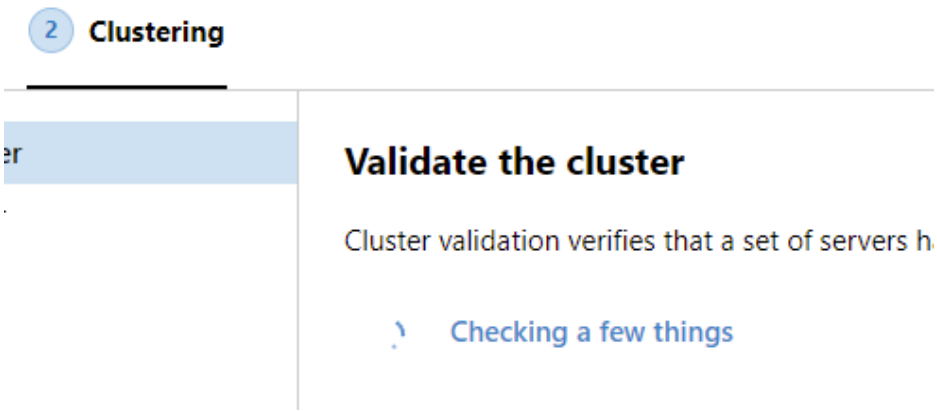
9. The tool will look for missing features. In this case, we need the clustering feature installed. Use the **Install features** button to have WAC fix that for you.



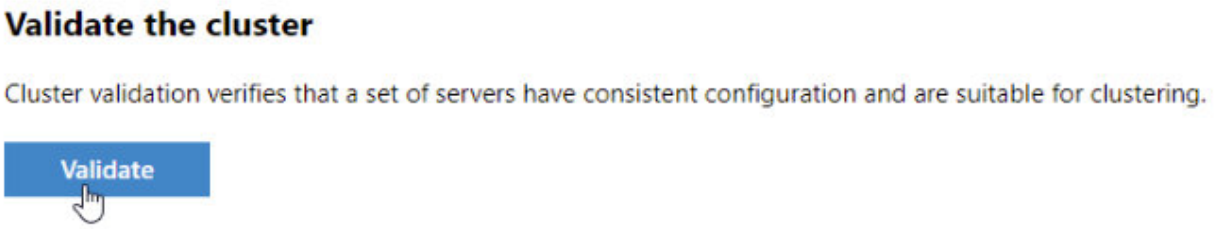
10. After feature installing, WAC performs an update check. This feature is not the most reliable. Install updates outside of the tool and refresh or wait until later to install updates.



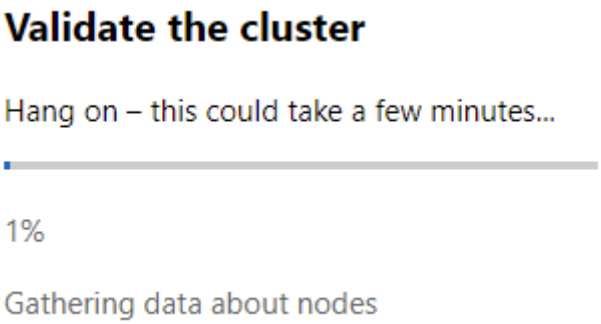
11. Now, the tool brings you to the real cluster validation section. First, it spends a few minutes getting ready (this is not a validation operation).



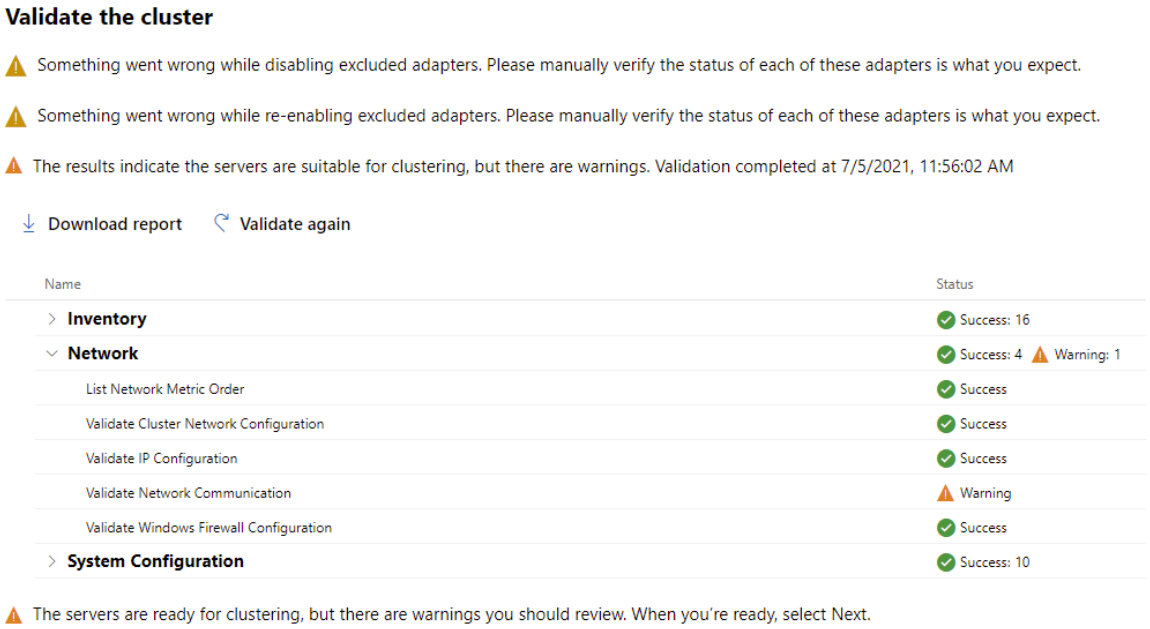
12. Once that finishes, you get a **Validate** button and can start the process.



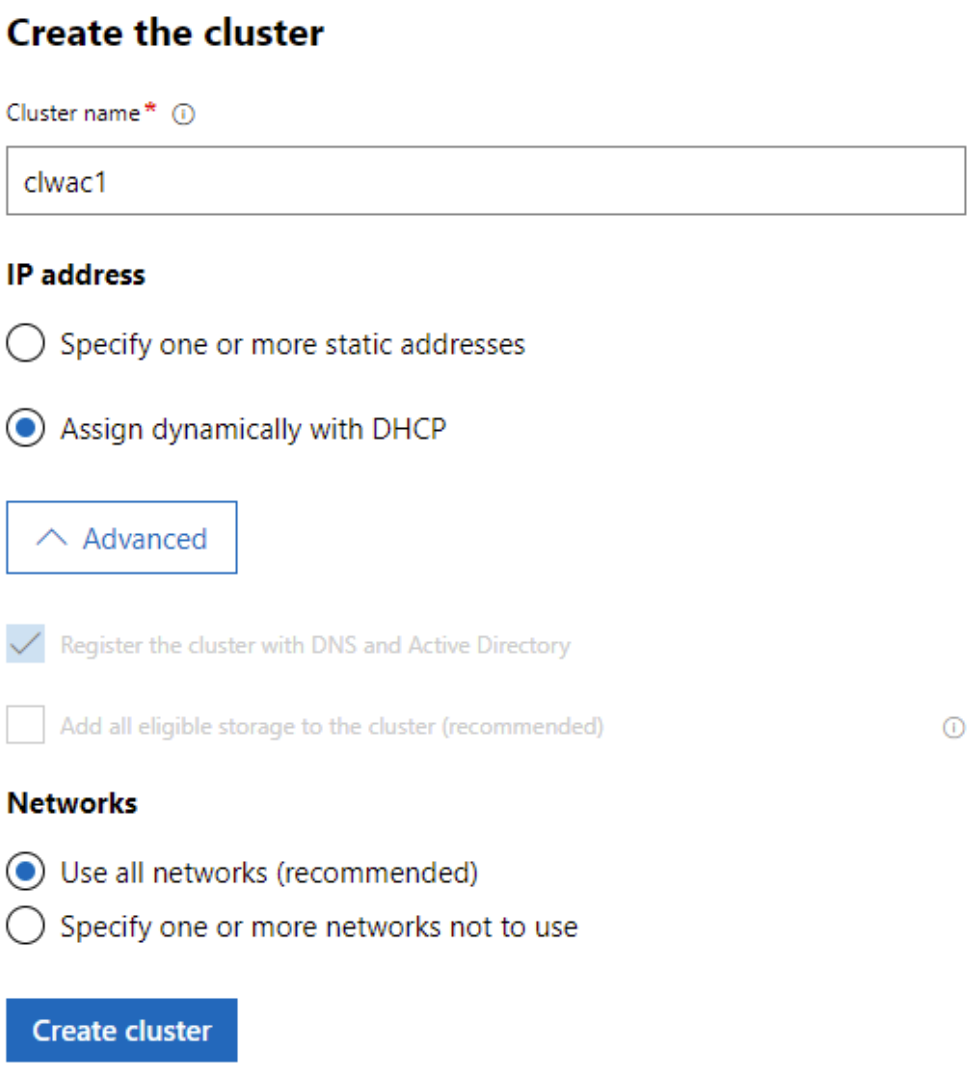
13. As you might expect, validation takes time. It doesn't show its current activity the way that the MMC does, but it does have a progress meter.



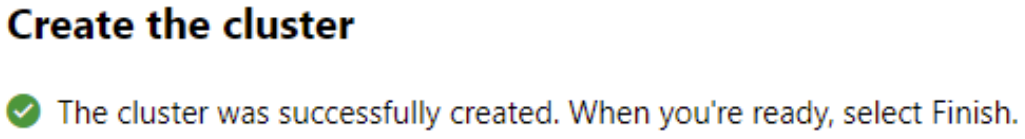
14. When it finishes, you get a report. The in-WAC display only gives you very high-level information. If you click **Download Report**, you get the traditional cluster validation report just as MMC and PowerShell produce.



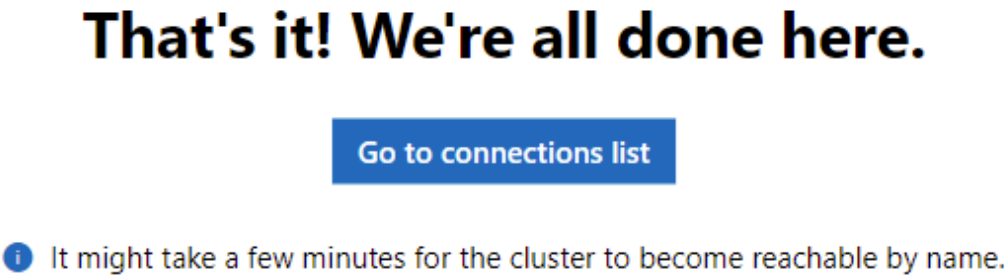
15. With all the prep work complete, you just need to provide a name for the cluster object and decide on IP addressing, DNS behavior, and whether to let it automatically add discovered storage. As a bonus not included in the MMC wizard, you can even perform some network exclusions right at creation time. For this demonstration, a dynamic IP was chosen and defaults elsewhere.



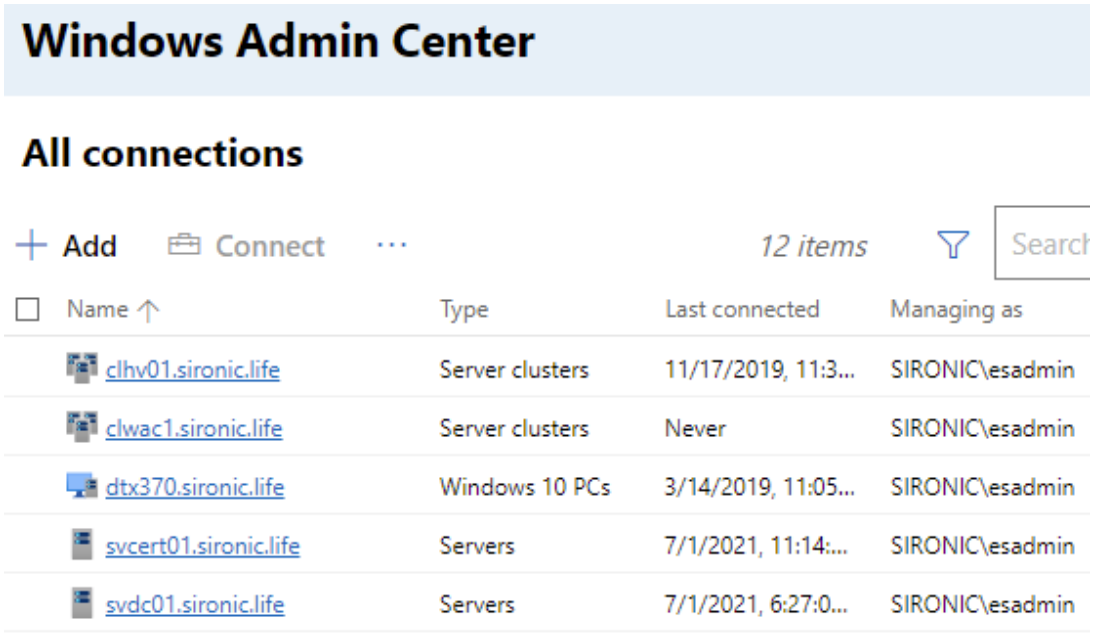
16. If all goes well, you get a nice completion message.



17. And then… another:



18. Your new cluster will now appear in the system list, even if you can't reach it until DNS propagation completes.

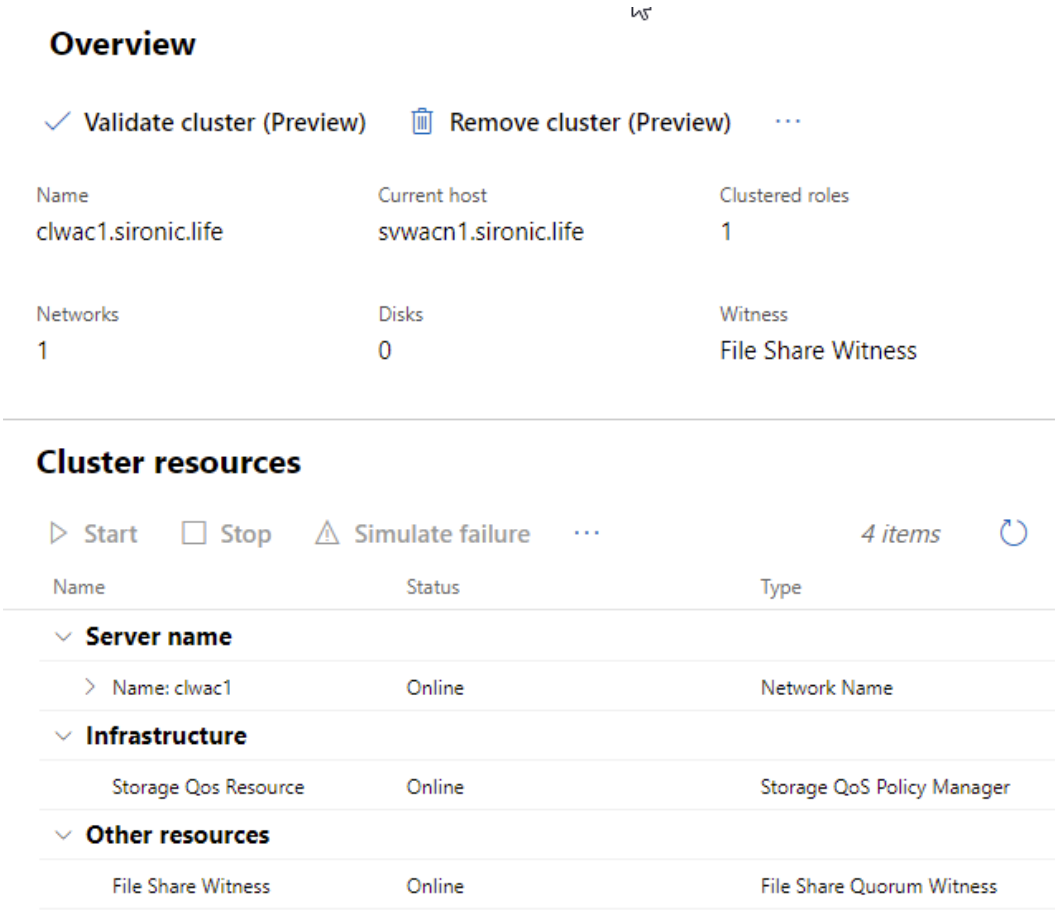


19. If you received a notice about CredSSP during installation, remember to connect to the nodes individually to disable it.

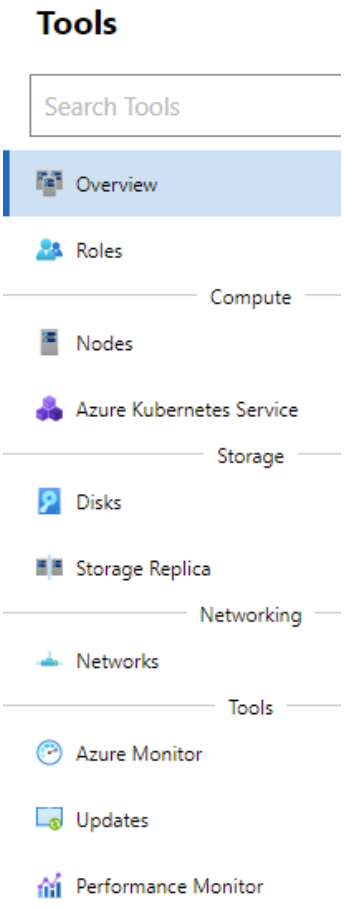
A few times through this, and you probably won't use the MMC again. Unless you do a lot of bulk creation or have a convenient script set up, you might not even use PowerShell to create clusters anymore. This new wizard does not perform every last thing that you might want to get a cluster started up, but it goes a long way.

MANAGING A CLUSTER

After the remarkable job that the cluster creation does, the cluster management feature feels like a letdown. You access it by clicking on the cluster in Windows Admin Center's main system list, just like you would a computer system. That takes you to the Cluster Manager Overview.



You have a small **Tools** menu at the left:



If the nodes run Hyper-V, you will also have a **Virtual Machines** entry under **Compute** and a **Virtual Switches** entry under **Networking**. These work just like the same items in the Hyper-V extension but from the cluster's perspective.

A few things:

- Other than virtual machines, the clustering extension is almost completely ignorant of roles. You can perform traditional cluster-related chores on them, such as failover, start, stop, delete, etc. But, unless an extension exists that can manage the role, you will need to switch to a completely different tool to do anything else.
- You cannot rename anything except roles. Disks, networks, and other infrastructure components will have you back in MMC or PowerShell.
- You cannot change the allowed traffic or priority of networks.
- The **Disks** tool cannot always detect uninitialized volumes attached to nodes. This appears to stem from a shortcoming in the Storage extension, not the cluster extension.
- You cannot change a cluster disk to a Cluster Shared Volume or vice versa.

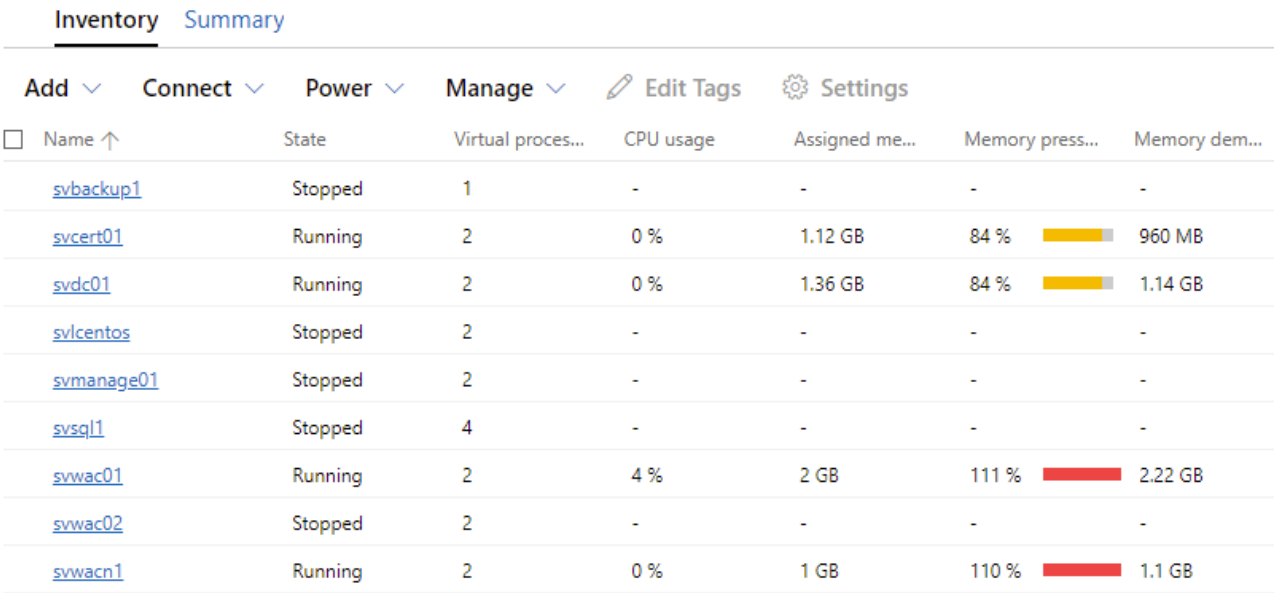
The cluster extensions in WAC show its best and its worst sides. Some tools have developed far beyond anything we've had before. Others lag so far behind that we probably won't use them until they receive meaningful updates.

HYPER-V

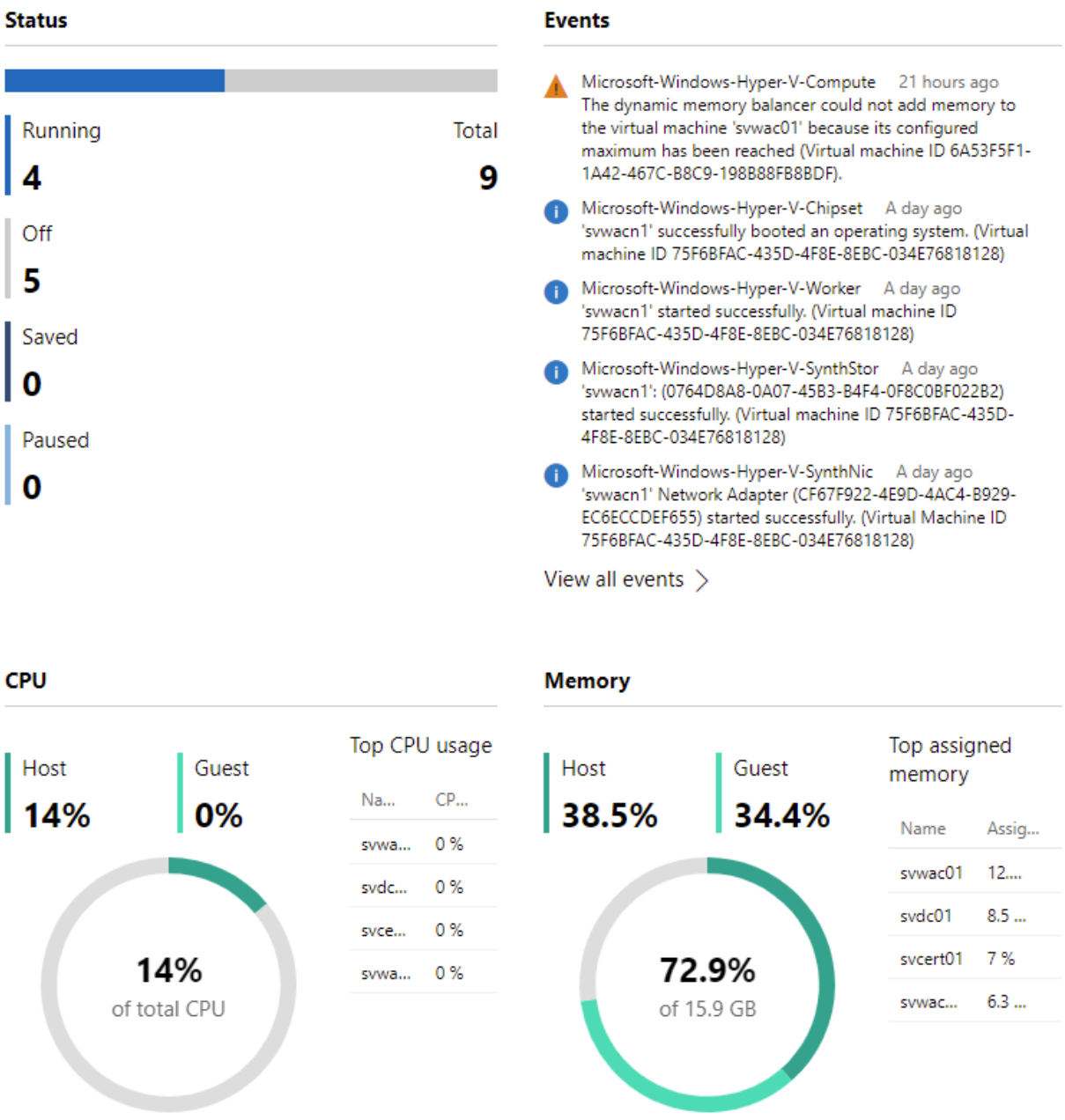
Windows Admin Center's Hyper-V extensions work solidly and even tie into the clustering extensions so that you can manage clustered virtual machines from one place. Hyper-V has two separate extensions that as unique entries on the overview page of a Hyper-V host or a cluster with Hyper-V-enabled nodes and it makes modifications to the host's settings control. We will look at the extensions separately.

MANAGING HYPER-V VIRTUAL MACHINES

After connecting WAC to a cluster or host running Hyper-V, click the **Virtual Machines** item near the bottom of the Tools list. It shows you a list of all virtual machines in the selected context and some information about them.



If you change to the **Summary** tab, you get collated information about all virtual machines. It has **CPU** and **Memory** graphs so that you can quickly detect any systems using excessive amounts.



As you click around in the extension, you'll find the sort of items that you'd expect in the Hyper-V and Failover Cluster Manager MMCs. Settings are spread out differently, but everything has a logical home.

A few extras that the MMCs don't have:

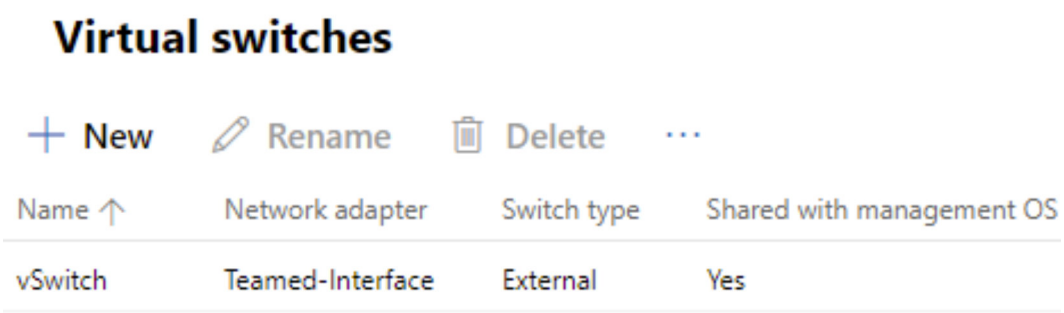
- The **Processors** tab in a virtual machine's settings allows you to enable nested virtualization.
- If a virtual machine's disks reside on SMB storage, or you attempt to add or create a virtual disk on SMB storage, WAC will prompt you for CredSSP permission. This avoids a lot of the delegation headaches that you have in the MMCs. Just remember to disable CredSSP when you're done.
- The Hyper-V extension has a proper **Clone** tool.

The Hyper-V Virtual Machines extension improves greatly over the MMC tools. It does operate slowly at times. Occasionally you'll start an operation that hangs indefinitely for no obvious reason. Even with that periodic annoyance, virtual machine administrators will use WAC often.

MANAGING HYPER-V VIRTUAL SWITCHES

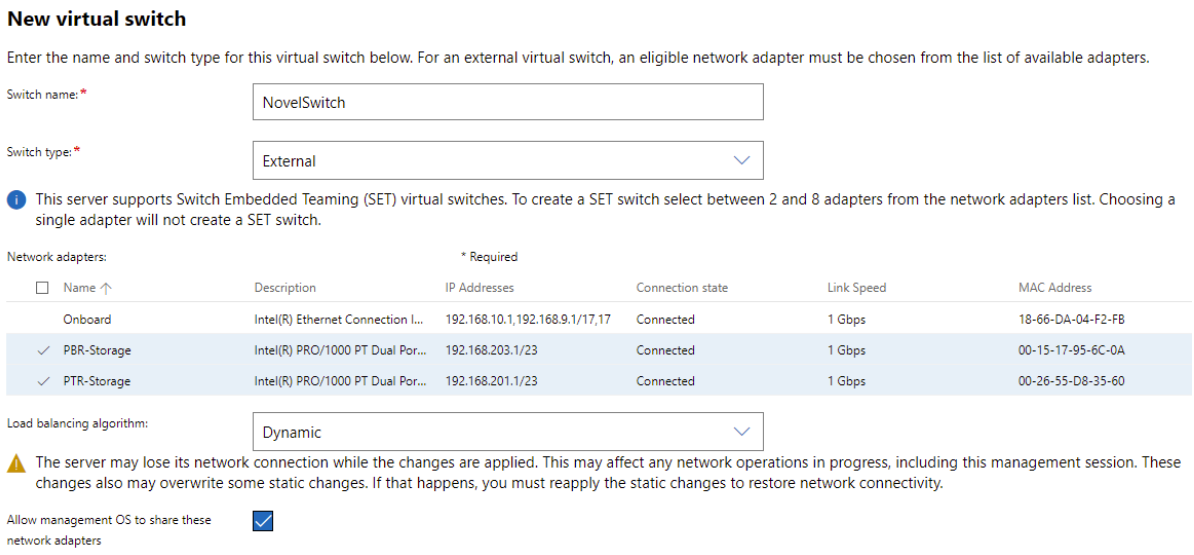
You can find the **Virtual switches** extension right below the **Virtual machines** extension in the **Tools** list of a Hyper-V-enabled host or cluster. This extension's powers will not overwhelm you, but the Hyper-V virtual switch does not have many controllable components anyway.

When you first open the extension, it shows you any existing virtual switches.



Unfortunately, the virtual switches extension continues to use the misleading and confusing "shared" description when indicating if the management operating system has its own virtual adapter(s) attached to a switch. Aside from that, it does a better job than existing graphical tools and even has a couple of edges over the PowerShell cmdlets.

Of note, the virtual switch extension understands switch-embedded teaming. Also, as a relief of a long-standing problem in the MMC, it knows adapters by more than their mostly useless descriptions:

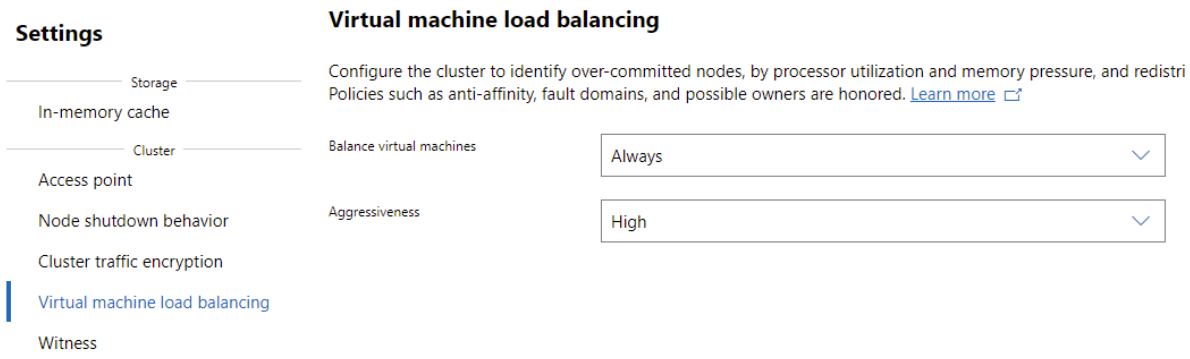


Just remember that "share" doesn't mean "share the physical adapter(s)" but "create a virtual adapter for the management OS to use", then everything here works as expected. Just like the MMC, the "share" option will create exactly one virtual adapter and it will not allow you to name it. In fact, we still have no graphical way to meaningfully manage host-connected virtual adapters at all.

If you select an existing virtual switch and click **Change**, it shows essentially the same items as it does when you create a new virtual switch.

MANAGING HYPER-V HOST SETTINGS

If you click the **Settings** item below the **Tools** menu on a Hyper-V enabled cluster or node, it takes you to the expected settings but with some new menu entries. On a cluster, you can control automatic virtual machine balancing.

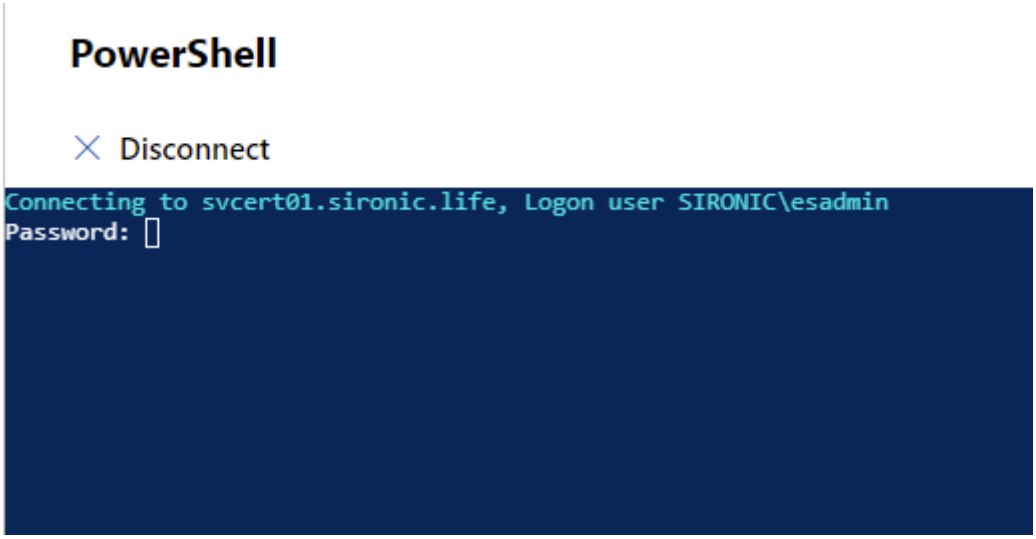


Both the node-oriented and cluster-oriented settings screen give you a new **Hyper-V Host Settings** category with **General**, **Enhanced Session Mode**, **NUMA Spanning**, and **Storage Migration** options. A standalone Hyper-V host was not available during the testing of this tool, but one would presumably have an additional section for Live Migration settings. All tools operate the same way in both locations, but making changes through the cluster applies to all nodes.

As mentioned in the preceding section, we do not have a way to make changes to host-connected virtual adapters. As a possibly related observation, WAC's regular network extension has remarkably few capabilities as well. At this time, WAC also has no way to configure Hyper-V Replica. Otherwise, it represents a solid step forward in Hyper-V management.

POWERSHELL

If you find jumping in and out of remote sessions obnoxious, then you might really appreciate Windows Admin Center's PowerShell extension. It greatly reduces the dull labor and authentication. To try it out, connect to a host and click the **PowerShell** item in the **Tools** menu at the left. The extension will open a familiar blue console and initiate a remote session. You only need to provide a password:



If the PowerShell session drops, the **Disconnect** button turns to a **Connect** button and you can do it again.

Leaving the extension will cause an automatic disconnect; your session will not persist. Any deliberate disconnection of the session will do the same. You lose variables, current activities, etc. If you do not initiate a disconnect and your session on the remote system otherwise persists, then those things will remain if you reconnect. No matter what, you get to keep your command history. Since the PowerShell extension runs a couple of items when it connects, you'll find those items in history before you get to your own commands.

The PowerShell extension functions the same as a remoting session initiated from a console window. It might fit into your usage patterns better than multiple console windows or single console re-use, though. It's also nice when you're performing multiple operations in WAC and need to briefly drop down to the prompt for some minor step that you can't do in the graphical interface.

CONNECTING TO AZURE

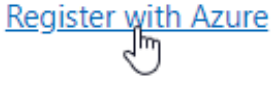
As the line between "on-premises" and "public cloud" continues to blur, Windows Admin Center adds features and capabilities to manage both. We will always need multiple tools to do the entirety of our jobs, but Windows Admin Center might well turn into the first tool to try for everything.

To enable its cloud capabilities, you need to connect your WAC installation to Azure. As you move through different pages and extensions, you will encounter many encouragements to register. This chapter starts off with a walkthrough and then briefly discusses some of the available tools.

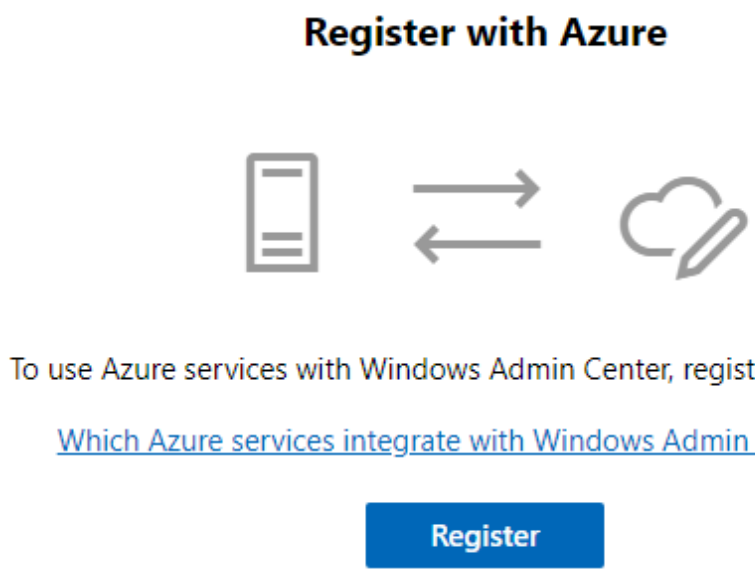
HOW TO REGISTER WINDOWS ADMIN CENTER IN AZURE

You can use any "register with Azure" link that you find in Windows Admin Center. They all go to the same place. If you begin somewhere else, you will connect with these steps at #2 or #3.

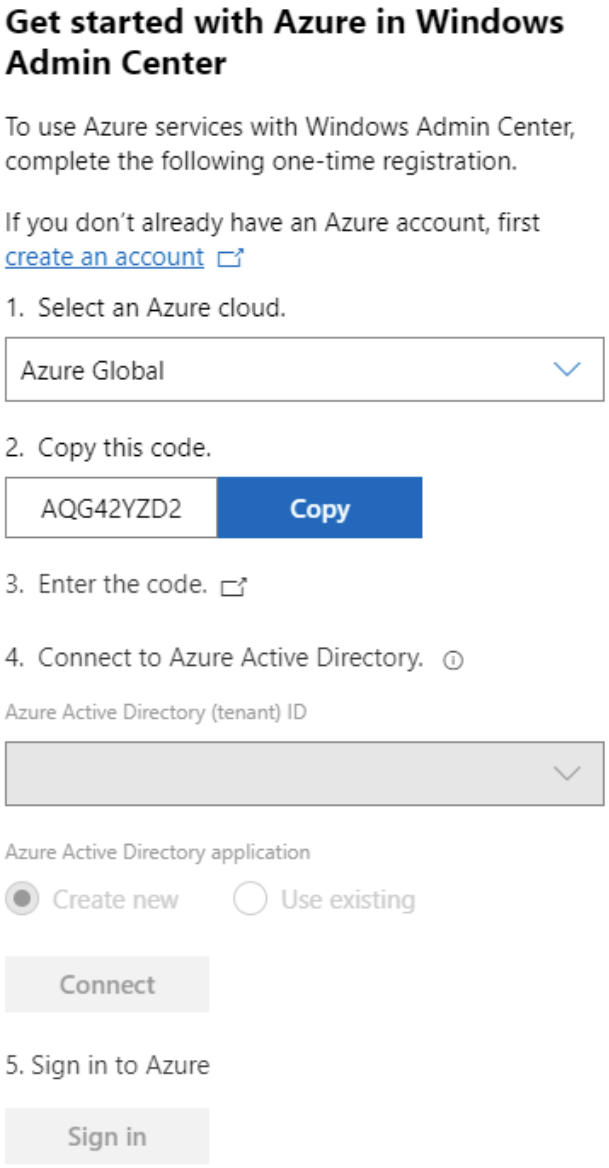
1. Open the global **Settings** page. It opens to the **Account** tab. Click it the **Register with Azure** link.



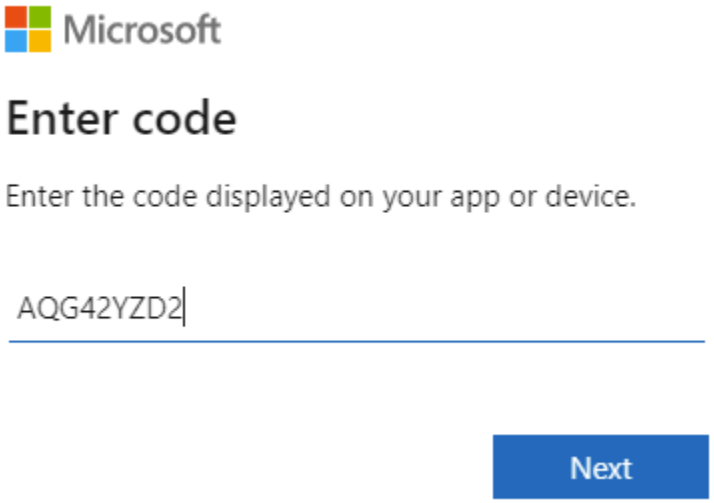
2. That will take you to another screen, one which you might have seen in other locations, where you again get to click **Register**.



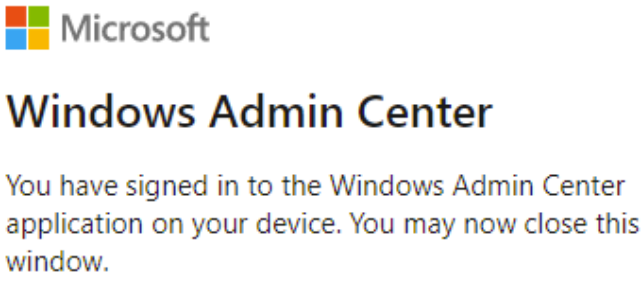
3. A window will open on the side with a short alphanumeric code and an **Enter the code** link. When you click that link, Azure opens in a new window.



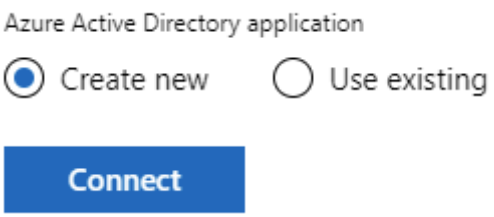
4. Authenticate if necessary, then provide the code when prompted.



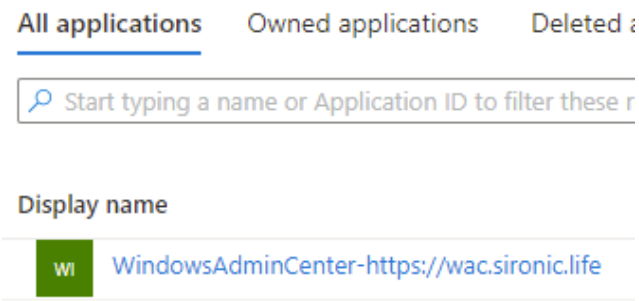
5. That will perform the preliminary link between your WAC instance and Azure, but it does nothing permanent.



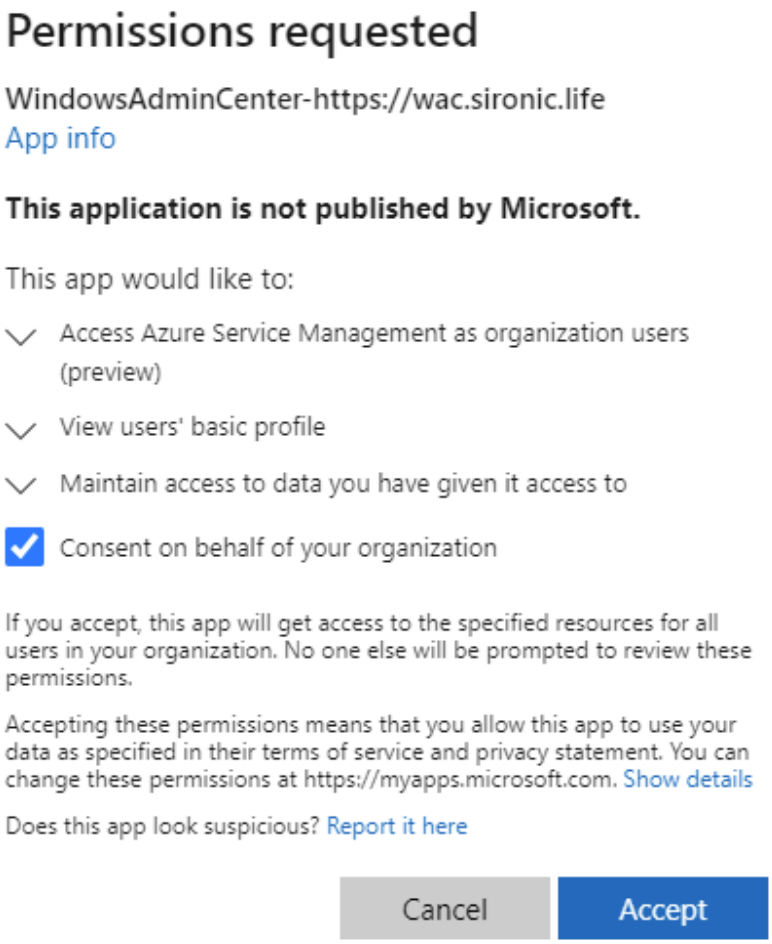
6. Return to the screen shown in step 3. It will now allow you to choose to create a new application or use an existing one. Unless you need to reattach to a Windows Admin Center instance that you already created, selected **Create New** and click **Connect**.



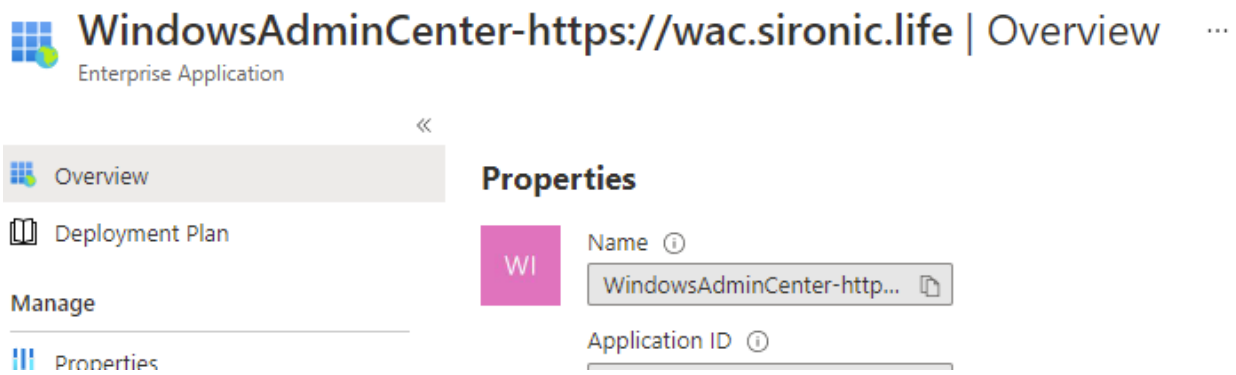
7. In Azure, you will see creation of the application.



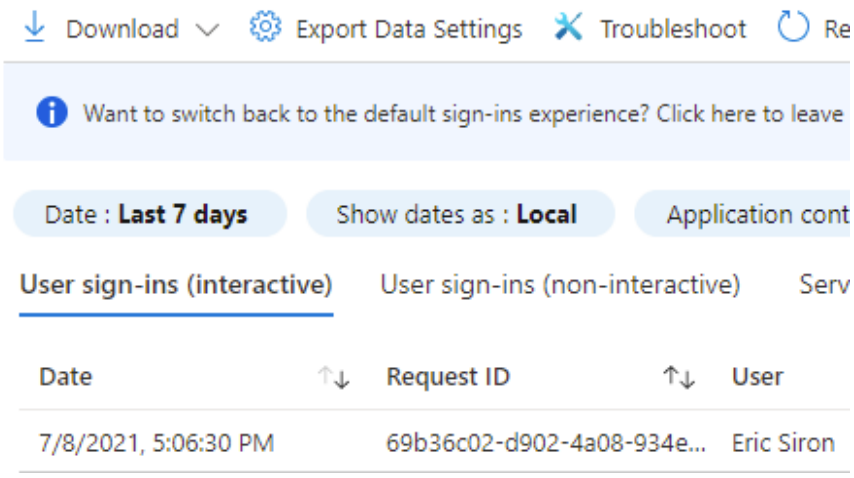
8. You may need to authenticate with Azure again. After that, it will ask you to confirm the request to grant permissions to the new application. Verify that it is Windows Admin Center, wonder for a moment why it says that it was not published by Microsoft, and then click **Accept**.



9. Verify that you now have an application blade in Azure for Windows Admin Center.



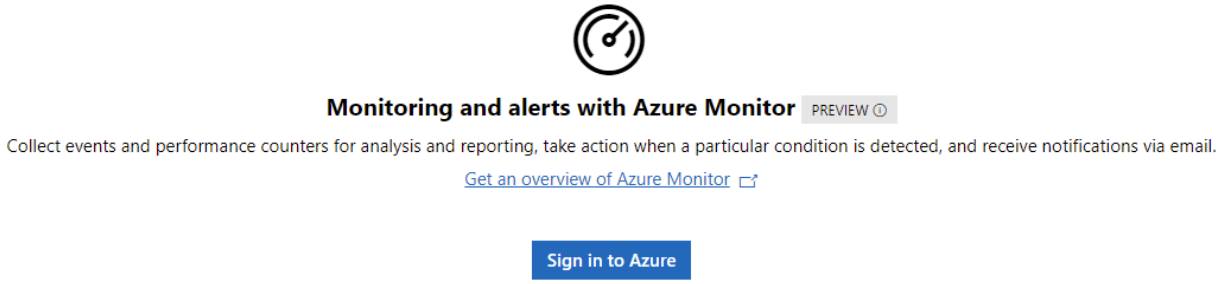
As you connect more items in WAC to Azure, you will have more items on the blade's menus. For instance, if you allow Azure Active Directory as a second authentication factor, you will get a **Sign-ins** item under **Activity** that shows a running history of WAC logins.



The connection and application have no associated cost. Several connected features do. Windows Admin Center and Azure both have ways to keep track of your spend. Your Azure experience applies best in this regard.

AZURE FEATURES

Windows Admin Center has many interconnects with Azure. You will often stumble upon them organically, where the interface will present you with an opportunity to connect. For instance, clicking the **Azure Monitor** tool in a cluster's overview page gives you a link to learn about Azure Monitor and a button to sign in and start working with it.



A quick list of most of the Azure features available today:

- Azure Monitor collects performance and event data from the on-premises systems managed by Windows Admin Center. You can use it for reporting purposes. You can also configure it to react to defined thresholds and events. Fulfilling a common early request, it can send e-mails to administrators as well.
- The Azure Kubernetes extension helps with containers. As organizations adopt Azure Stack HCI, tools such as this will help with managing hybridized containers.
- Azure File Sync helps you to manage the connections between your on-premises file servers and your Azure Files.

- Much like the Azure File Sync extension, the Azure Security extension brings that plane down into Windows Admin Center where you can easily apply its Azure Defender component to on-premises hosts.
- The in-WAC description of Azure Arc presents it as an inventory, policy, and governance solution. It does that, of course, but it does quite a bit more. It folds in with Azure Monitor, Azure Security, Azure Automation, and other tools. Azure Arc continues to evolve and expand. Best of all, Azure Arc itself has no cost.
- The Azure Site Recovery extension allows you to replicate virtual machines into Azure as a disaster recovery measure. Windows Admin Center can broker and manage related activities.

Both the Azure and Windows Admin Center teams keep their eyes on the horizon. Microsoft understands that neither public cloud nor on-premises can provide adequate functionality alone. Bringing Azure products under WAC's management helps to bridge the environments, especially in overlap conditions. The basic WAC-to-Azure registration costs nothing and some of the features either have no cost or have such a low cost-per-object that many small organizations will never receive a bill for them. If your organization currently has no Azure presence, WAC represents a painless way to try it out with meaningful activities.

CONTROLLING VIA POWERSHELL

With a feature that probably doesn't surprise anyone, you can use PowerShell to manage Windows Admin Center. That seems a bit meta, but it will help with a few things that you do infrequently but appreciate.

THE UNDERLYING MECHANICS

Most administrators don't care much how their tools work as long as they work. To that end, we will not spend much time or space on this subject. You only need to know that the PowerShell module provided by Microsoft does not do anything that someone else could not do.

Windows Admin Center provides a robust REST API. For the interested, Microsoft publishes details, examples, tutorials, and more on the Windows Admin Center docs pages. PowerShell methods to manage WAC, even the module published by Microsoft, uses that API. We know this because they provide their modules as simple PSM1 text files that we can examine.

If you have the urge to improve management of Windows Admin Center, you probably can't find a better starting point than reading those files yourself (yes, we'll tell you where they are, wait until the next section). If you just want to use what you can get today, read on.

LOADING AND EXPLORING THE MODULES

Windows Admin Center installs its PowerShell modules in the same MSI that delivers the interface. That means that you already have it on your WAC desktop or gateway system. You need to tell them where to find WAC, so they might operate with a WAC gateway if copied to a non-WAC system. That might have questionable legality and Microsoft has not stated anything either way, so try that at your own risk. You can always remotely connect to a PowerShell system on a WAC host.

> **Note:** The Windows Admin Center PowerShell modules generate errors when imported in PowerShell versions 6 and higher. They appear to primarily impact the data types, so they might not prevent use. The modules work fine in Windows PowerShell 5.1, which was used for all the following demonstrations.

The modules do not autoload which, given the above note, might be a good thing. You have to load them manually. They use extremely nondescript names that could easily cause collisions with other modules, so these demonstrations use the **Prefix** parameter to ensure that doesn't happen. You do not need to use that parameter unless you encounter problems.

The modules exist in separate subfolders at C:\Program Files\Windows Admin Center\PowerShell\Modules\. They are:

- ConnectionTools
- ExtensionTools
- ManagementTools

If you add that path to your PSModulePath environment variable, that will enable autoloading. Just keep in mind the above note about shell versions and the potential problem with collisions.

If you want to import the modules manually, the following lines show how to import all three simultaneously. If you only need commands from one or two, you can choose the applicable lines.

```
Import-Module -Prefix WAC -Name 'C:\Program
Files\Windows Admin
Center\PowerShell\Modules\ConnectionTools\ConnectionTo
ols.psm1'

Import-Module -Prefix WAC -Name 'C:\Program
Files\Windows Admin
Center\PowerShell\Modules\ExtensionTools\ExtensionTool
s.psm1'

Import-Module -Prefix WAC -Name 'C:\Program
Files\Windows Admin
Center\PowerShell\Modules\ManagementTools\ManagementTo
ols.psm1'
```

If you used the **Prefix** parameter, then you can easily view all included commands like this:

```
Get-Command -Name *-wac*
```

CommandType	Name	Version	Source
Function	Add-WACFeed	0.0	ExtensionTools
Function	Enter-WACSmePSSession	0.0	ManagementTools
Function	Enter-WACSmeSshSession	0.0	ManagementTools
Function	Export-WACConnection	0.0	ConnectionTools
Function	Get-WACExtension	0.0	ExtensionTools
Function	Get-WACFeed	0.0	ExtensionTools
Function	Import-WACConnection	0.0	ConnectionTools
Function	Install-WACExtension	0.0	ExtensionTools
Function	Remove-WACFeed	0.0	ExtensionTools
Function	Uninstall-WACExtension	0.0	ExtensionTools
Function	Update-WACExtension	0.0	ExtensionTools

If you didn't specify a credential, then you can retrieve the commands from each module individually or you can try something like:

```
Get-Command -Name *Tools
```

That will definitely show you the commands from the WAC modules, but it might also include commands from other modules. Check the **Source** column to be sure.

You can figure out the use of most of the commands just from their names. Most of them duplicate things that you can do in the graphical interface, and really don't offer anything except bulk capabilities in some cases. We will look at two that you will likely use.

IMPORTING AND EXPORTING SYSTEM CONNECTIONS

Protecting, maintaining, and sharing your system list matters more than it might at first seem. The Windows Admin Center database is more fragile than it might seem at first. Something as common as a repair or upgrade install of Windows can cause the system to wipe it out entirely. If you have a system inventory elsewhere, you don't want to retype it into WAC. Even if you go through that pain, the next administrator won't want to the same data entry for their account. To solve these problems and more, you have two tools:

- Export-Connection
- Import-Connection

> **Note:** In the preceding section, the demonstration imported the module with a WAC prefix, so these would work as **Import-WACConnection** and **Export-WACConnection** instead. The demonstrations in this section will use the default names.

First, Export-Connection. If you're at the WAC host's console logged in as yourself, this is about as easy as it gets:

```
Export-Connection -GatewayEndpoint 'wac.sironic.life'
-fileName
$env:USERPROFILE\Documents\wac-connections.csv
```

Just like that, you have a file with your connections that you can open in Notepad, even on a system that doesn't have the Windows Desktop Experience:

```
Notepad $env:USERPROFILE\Documents\wac-connections.csv
```

Do whatever you like with this file. Preferably, start by transferring it somewhere safe.

Next, Import-Connection. It looks almost identical:

```
Import-Connection -GatewayEndpoint 'wac.sironic.life'
-fileName
$env:USERPROFILE\Documents\wac-connections.csv
```

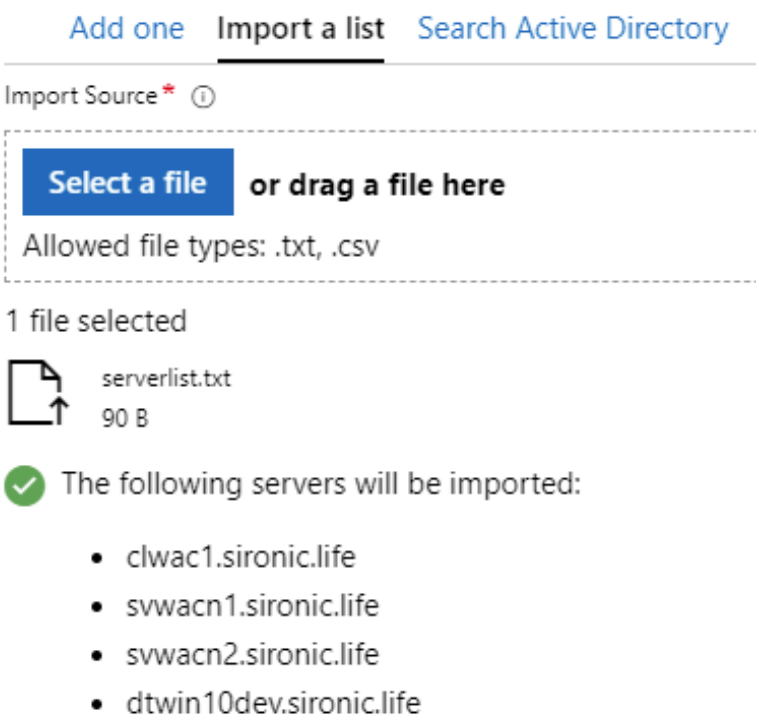
Did it surprise you that we did not need to finesse the file? Import-Connection does not have the same limitation as the CSV importer in the graphical interface. Don't celebrate just yet, though: Import-Connection requires the same format generated by Export-Connection. You can address that in more complicated scripting. Before we get to that, let's look at another simple reason to like these cmdlets.

At this point, you have a CSV file with all your connections in it. So, what happens if another administrator runs Import-Connection under their account? Hopefully, they get all your connections. Unfortunately, the Import-Connection has bugs that cause it to behave unpredictably at times. It seems to work more reliably with desktop WAC than gateway, although that could be coincidence. This is where we hope that some third party will step up and produce their own refined versions of these cmdlets.

But, with a bit of scripting, you can still salvage what you have and also grant yourself the ability to use almost any file that contains a list of servers instead of relegating yourself to rigid formats. To keep it as simple as possible, the following was run in the same folder that contains the file created by Export-Connection:

```
Import-Csv .\wac-connections.csv | foreach { $_.name }
| Add-Content .\serverlist.txt
```

That relatively simple script just created a file that Import-Connection will hate, but the graphical importer will love:



The only real problem with this method is that it imports clusters as servers. You'll have to decide between editing the file or removing the server entries from WAC's list and re-entering them as clusters. The environment demonstrated above prefixes "cl" on all cluster names, so a script could intervene at some point in the export or import stage with a filter to remove matching items.

Tinker with these scripts as you like. Most importantly, get a copy of your server list as a backup.

THE FUTURE OF WINDOWS ADMIN CENTER

Microsoft continues to work feverishly on Windows Admin Center. It will continue to receive additions and refinements for the foreseeable future. Microsoft does have a history of losing interest in projects and letting them fall into a permanent state of disrepair, but the intersection of WAC and Azure should hold their attention for a while. With WAC's extensibility the community and vendors can fill in gaps.

To finish up the book, we'd like to leave with a few suggestions to Microsoft:

- Reinstate the transparency and community that UserVoice gave us. No one has any objection to Microsoft using its own tools (except the UserVoice software company, anyway), but we need the ability to see and add to popular ideas and to keep pressure on things that we as administrators find important.
- Tone down the "modernness" of the interface and make it more usable. WAC's layouts have gotten much better since initial release, but they still do not work well with small resolution screens, multiple windows, and screen sharing. When capturing screenshots for this book, we frequently had to find clever ways to make a shot large enough to read while not trimming off too much information.

- Make progress on the old, partially implemented, neglected extensions before dumping heaps of resources into the shiny, new, fun extensions which inevitably become old, partially implemented, neglected extensions.
- Sometimes, Windows Admin Center does an impressive job of reporting information about failures that we can work with. Sometimes, it tells us that some oddball variable didn't have a length when we did nothing but click on a different extension. Prioritize giving us feedback that we can do something about. This is kind of a Microsoft-wide problem anyway, and as one of the current standard bearers for the company, it would be nice to see the WAC team set an example by taking on this problem.
- Make default extensions and tables customizable. We really don't need to know the precise CPU model every single time we connect to a host. Some of us will never use Azure Site Recovery for our on-premises VMs and would prefer to find another use for the space consumed by the Disaster Recovery Status column.
- Work with the Active Directory teams to find an easier way to deal with delegation and such issues. If nothing else, make it an extension right in WAC.
- You really can find a way to update the PKI certificate without hunting down a thumbprint and reinstalling the program. We all know that you can.

As you use Windows Admin Center, look for places that you feel it could improve. Use the feedback system. Remember that no one likes to receive mean feedback, and it's a normal human response to feel less inclined to help someone that's being hurtful, so maintain decorum and professionalism in your submissions.

Above all, keep using Windows Admin Center and discovering more features. Your usage helps drive future innovations whether you provide feedback or not!

HORNETSECURITY

FACT SHEET

365 TOTAL PROTECTION

Security and compliance management for Microsoft 365

We offer you two comprehensive packages for your company security management developed for Microsoft 365: With 365 Total Protection Business, you get a comprehensive security solution with a wide range of features that ensure your email and data security in Microsoft 365. The Enterprise version covers legally compliant email archiving with advanced features and offers intelligent protection against advanced persistent threats by using AI-based analysis mechanisms.

Protection from:

- **Targeted attacks on Microsoft 365 accounts**

SPECIALLY DEVELOPED FOR MICROSOFT 365 AND SEAMLESSLY INTEGRATED

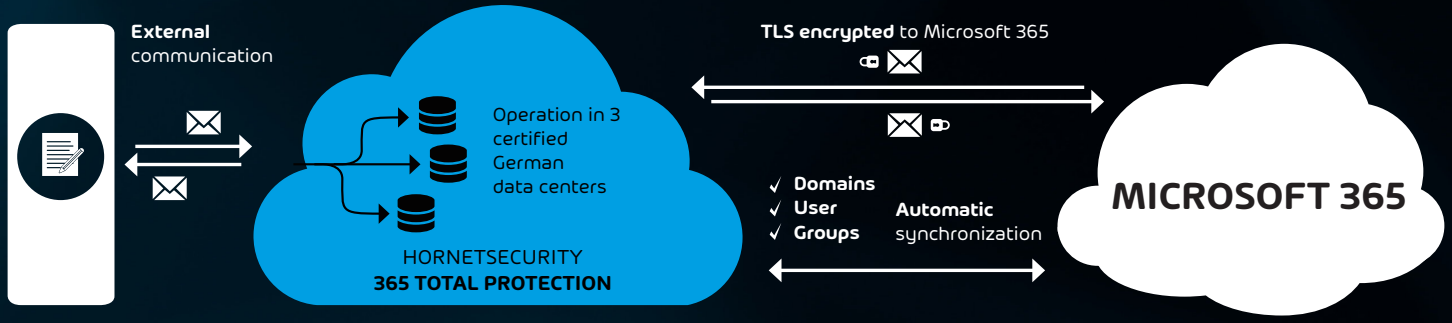
It couldn't be easier – onboarding within 30 seconds.
In just 3 clicks, the intuitive onboarding process is complete and your Microsoft 365 merges with 365 Total Protection.
365 Total Protection makes sure you get the most out of your Microsoft cloud services.

Fig.: Simple onboarding process in three steps

1. REGISTER **COMPANY DATA**
2. CONNECT **WITH MICROSOFT**
3. SET UP **COMPLETED!**

INTEGRATION OF 365 TOTAL PROTECTION IN THE EMAIL MANAGEMENT SYSTEM

All aspects of security administration are easy to manage with 365 Total Protection – without the need for maintenance or updates. Existing user profiles can be managed or created in mere seconds.



www.hornetsecurity.com | info@hornetsecurity.com

START YOUR 30-DAY TRIAL

HORNETSECURITY

FACT SHEET

	365 total Protection Business and Enterprise – features	Description
✓✓	Email Live Tracking	Monitoring of all email traffic in real-time and definition of filtering and delivery options
✓✓	Infomail Handling	Prevention of direct delivery of emails classified as newsletters and info mails
✓✓	Content Control	Protection against the intrusion or sending of unauthorized file attachments according to company policy
✓✓	Compliance Filter	Extended filter for automatic checking of email traffic according to self-defined filter rules
✓✓	Threat Defense	Multi-level filtering systems and in-depth analysis for immediate detection and defense against new types of threats and attacks
✓✓	Outlook Deny & Allow listing	Interface for central control from Outlook
✓✓	User-Based Individual Signatures	Central control over company-wide uniform email signatures; automatic matching of contact data records through integration of Active Directory
✓✓	1-Click Intelligent Ads	Setup of automatically integrated advertising banners or links in the email signature for external corporate communication; group-based assignment possible
✓✓	Company Disclaimer	Automatic integration of uniform and legally compliant company disclaimers in every outgoing email; group-based assignment possible
✓✓	Global S/MIME & PGP Encryption	Strong encryption solution to secure email communication against unauthorized modification or access by third parties; protection of internal company and sensitive contents against spying
✓✓	Secure Cipher Policy Control	Central TrustChain management; individual definition of security criteria used for email communication
✓✓	Secure Websafe	Securing confidential email communication with communication partners who do not use encryption technology

	365 Total Protection Enterprise – advanced features	Description
✓	Email Archiving	Automated, legally compliant and audit-proof email archiving immediately upon receipt and dispatch of emails
✓	10-Year Email Retention	Simple and legally compliant access to archive data within the framework of legal retention periods
✓	eDiscovery	Extensive full-text search with numerous filter functions for precise location of searched data in seconds
✓	Forensic Analyses	Forensic analysis mechanisms, algorithms and AI-based detection mechanisms for effective defense against sophisticated threats
✓	ATP Sandboxing	Protection against targeted and blended attacks through dynamic analyses
✓	URL Malware Control	Securing of all Internet calls from email communication; analysis and securing of downloads
✓	Global Security Dashboard	Overview of the Company Security Management; collection of comprehensive information (threat reporting, attempted attacks including attack type and vector) at a glance
✓	Malware Ex-Post-Alert	Notification of emails subsequently classified as harmful including detailed evaluation
✓	Contingency Covering	Effective protection against system failures with automatic immediate activation

START YOUR 30-DAY TRIAL

ALTARO BACKUP

Solutions for Companies and Organizations

Hyper-V & VMware Backup & Replication

Award-winning virtual machine (VM) backup and replication solution for Hyper-V and VMware environments

Learn more

Microsoft 365/Office 365 Backup

Backup solution for Microsoft 365 mailboxes and files stored in OneDrive and SharePoint, with unlimited storage

Learn more

Windows Server Backup

Physical to virtual (P to V) backup solution to back up physical WIndows servers and restore them to a virtual environment

Learn more

Solutions for Managed Service Providers (MSPs)

Hyper-V & VMware Backup & Replication

Monthly subscription program enabling MSPs to offer Hyper-V, VMware and physical Windows server backup services

Learn more

Microsoft 365/Office 365 Backup

Monthly subscription program enabling MSPs to back up customers' Microsoft 365 mailboxes and OneDrive/SharePoint files

Learn more

EndPoint Backup

Monthly subscription program enabling MSPs to provide backup services for on-premise and roaming Windows desktop and laptop

Learn more

Start your free 30-day trial today

THE DOJO

The DOJO is a dedicated training and educational platform for system administrators and IT professionals. It is updated every week with high-quality, value-packed content every week including articles, guides and tips for Microsoft/Office 365!

DOJO

Register to the Altaro DOJO now to gain unrestricted access to all content and stay notified when new content is released - it's free to join!

Hyper-V — Go to Section

VMware — Go to Section

Backup & DR — Go to Section

MSP — Go to Section

SHARE THIS EBOOK