AZURE VIRTUAL DESKTOP

SUCCINCTLY

BY MARCO MOIOLI

SUCCINCTLY EBOOK SERIES

Syncfusion®

Azure Virtual Desktop Succinctly

By
Marco Moioli

Foreword by Daniel Jebaraj



Copyright © 2022 by Syncfusion, Inc.

2501 Aerial Center Parkway
Suite 200
Morrisville, NC 27560
USA
All rights reserved.

ISBN: 978-1-64200-222-5

Important licensing information. Please read.

This book is available for free download from www.syncfusion.com on completion of a registration form.

If you obtained this book from any other source, please register and download a free copy from www.syncfusion.com.

This book is licensed for reading only if obtained from www.syncfusion.com.

This book is licensed strictly for personal or educational use.

Redistribution in any form is prohibited.

The authors and copyright holders provide absolutely no warranty for any information provided.

The authors and copyright holders shall not be liable for any claim, damages, or any other liability arising from, out of, or in connection with the information in this book.

Please do not use this book if the listed terms are unacceptable.

Use shall constitute acceptance of the terms listed.

SYNCFUSION, SUCCINCTLY, DELIVER INNOVATION WITH EASE, ESSENTIAL, and .NET ESSENTIALS are the registered trademarks of Syncfusion, Inc.

Technical Reviewer: James McCaffrey
Copy Editor: Courtney Wright
Acquisitions Coordinator: Tres Watkins, VP of content, Syncfusion, Inc.
Proofreader: Graham High, senior content producer, Syncfusion, Inc.

Table of Contents

The Story Behind the *Succinctly* Series of Books 7

About the Author 9

Chapter 1 Introduction 10

What is Azure Virtual Desktop? 10

The reason why 11

The business values 11

On-premises VDI vs. Azure Virtual Desktop 13

Licensing 13

Technical advantages 14

Cost advantages 16

Windows 10 multi-session 16

Linux rates 16

Extended support included 18

Avoid license double counting 19

Publishing layer 20

Main adoption scenarios 21

Migration of an existing solution 21

Support of legacy applications 21

Implementation of new use cases 21

Chapter 2 Architecture 23

A simple architecture 23

Pooled vs. personal 25

Identity 28

End user connection 29

Azure Portal 30
Virtual machine sizing....31
Network sizing....32
Scaling logic....33
FSLogix....34
MSIX app attach....38
Disaster recovery and business continuity....41
Microsoft Teams optimization....42
How to print....43
Service monitoring....45
GPU workloads....46
Image Builder and Shared Image Gallery....48
User support....50
RDP Shortpath....51
Start VM on connect....52
Chapter 3 Security....54
Reverse connection....54
Multifactor and conditional access....55
Service tag....56
Web category control....57
Antivirus and anti-malware....60
Security baseline....61
Azure Security Center....61
Azure Defender for Storage....62
Redirection control....63
Security controls....64

Screen capture protection 65
Chapter 4 Image Optimization 67
Virtual Desktop Optimization Tool 67
Logon blur prevention 67
Microsoft Edge tabs optimization 70
Chapter 5 How to Build a Laboratory 72
Automatic 72
Manual 75
Chapter 6 Value-Added Partners 84
Citrix Virtual Apps and Desktops with Azure 84
VMware Horizon Cloud Service on Microsoft Azure 85
Thin clients 87
Lakeside 87
Nexthink 88
Nerdio 89
Chapter 7 Windows 365 91
Windows 365 Business 92
Windows 365 Enterprise 93
Resources 98
Conclusion 99

The Story Behind the *Succinctly* Series of Books

Daniel Jebaraj, CEO
Syncfusion, Inc.

Staying on the cutting edge

As many of you may know, Syncfusion is a provider of software components for the Microsoft platform. This puts us in the exciting but challenging position of always being on the cutting edge.

Whenever platforms or tools are shipping out of Microsoft, which seems to be about every other week these days, we have to educate ourselves, quickly.

Information is plentiful but harder to digest

In reality, this translates into a lot of book orders, blog searches, and Twitter scans.

While more information is becoming available on the internet and more and more books are being published, even on topics that are relatively new, one aspect that continues to inhibit us is the inability to find concise technology overview books.

We are usually faced with two options: read several 500+ page books or scour the web for relevant blog posts and other articles. Just as everyone else who has a job to do and customers to serve, we find this quite frustrating.

The *Succinctly* series

This frustration translated into a deep desire to produce a series of concise technical books that would be targeted at developers working on the Microsoft platform.

We firmly believe, given the background knowledge such developers have, that most topics can be translated into books that are between 50 and 100 pages.

This is exactly what we resolved to accomplish with the *Succinctly* series. Isn't everything wonderful born out of a deep desire to change things for the better?

The best authors, the best content

Each author was carefully chosen from a pool of talented experts who shared our vision. The book you now hold in your hands, and the others available in this series, are a result of the authors' tireless work. You will find original content that is guaranteed to get you up and running in about the time it takes to drink a few cups of coffee.

Free forever

Syncfusion will be working to produce books on several topics. The books will always be free. Any updates we publish will also be free.

Free? What is the catch?

There is no catch here. Syncfusion has a vested interest in this effort.

As a component vendor, our unique claim has always been that we offer deeper and broader frameworks than anyone else on the market. Developer education greatly helps us market and sell against competing vendors who promise to “enable AJAX support with one click,” or “turn the moon to cheese!”

Let us know what you think

If you have any topics of interest, thoughts, or feedback, please feel free to send them to us at succinctly-series@syncfusion.com.

We sincerely hope you enjoy reading this book and that it helps you better understand the topic of study. Thank you for reading.

Please follow us on Twitter and “Like” us on Facebook to help us spread the word about the *Succinctly* series!





About the Author



I am a Cloud Solution Architect who really enjoys technology.

My goal is to enable Microsoft partners to understand new products and solutions, create compelling offerings, and deliver the project following the most advanced best practices.

I spent more than 17 years at Microsoft covering multiple roles in different business units. For the first 10 years, I was in the field in the role of consultant/architect. Next, I spent four years as a pre-sale engineer. I'm currently part of the One Commercial Partner division that is responsible for partner enablement.

I was always involved in projects related to Windows migration (the first one was Windows 2000 Professional to Windows XP) and management products like Microsoft SMS 2.0 (which became Microsoft System Center Configuration Manager). I also had the opportunity to contribute on a project related to data center virtualization and workplace virtualization.

In 2019 I took the role of Partner Cloud Solution Architect, specializing in Azure Virtual Desktop for the West Europe area with the goal to help our partners understand the value and technical aspects of this new Azure service.

I strongly believe in the power of technical communities and the importance of sharing knowledge, so I'm quite active on platforms like LinkedIn and Twitter with my articles (in English) about Azure Virtual Desktop, endpoint management, and security. I'm also leading three different LinkedIn User Groups in Italian dedicated to these topics.

Let's connect!

Feel free to connect with me on LinkedIn or send me an email to tell me what you think about this ebook, and your feedback on what can I improve.

- LinkedIn: https://www.linkedin.com/in/marcomoioli/
- Email: marco.moioli@outlook.com

Chapter 1 Introduction

What is Azure Virtual Desktop?

Many people are aware of the existence of a new Microsoft service called Azure Virtual Desktop (AVD), but it seems that not so many of them are able to describe exactly what it is and how it can be helpful.

When I'm delivering trainings about this service, my typical introduction is: Azure Virtual Desktop is a Microsoft Azure service that is publishing Windows resources through the internet channel.

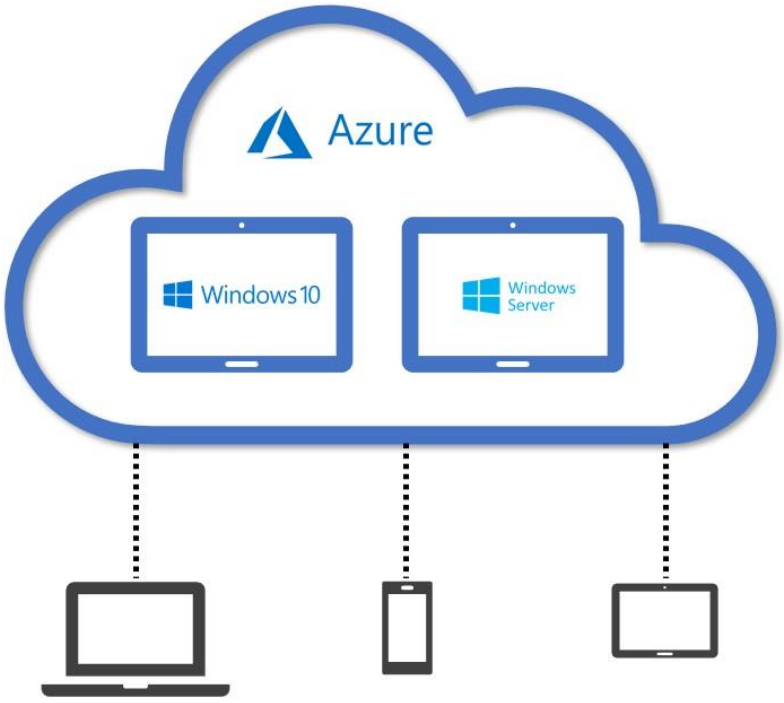


Figure 1: Azure Virtual Desktop

In a nutshell, I can use this service to publish the whole desktop or just specific applications installed inside Azure virtual machines, and my users will be able to consume these resources from any place on any device.

Azure Virtual Desktop was shipped in general availability in the summer of 2019 with the name "Windows Virtual Desktop," and was recently renamed Azure Virtual Desktop.

This service is about a lot of things: new and old technical concepts, security, productivity, and cost savings.

At the very core, Azure Virtual Desktop is a virtual desktop infrastructure (VDI) service delivered by the Azure public cloud—but it's much more than that. I hope this brief book will help you understand the whole picture.

First of all, I encourage you to read the official page about this service, "What is Azure Virtual Desktop?"

Microsoft's goal is to build "the best virtual desktop experience," and as I'm going to show you, this is already a complete and mature technology, deeply integrated into the Microsoft ecosystem, with several third-party vendors that are building value around this service.

The reason why

At a high level, I hope that now it's a little bit more clear what Azure Virtual Desktop is, and its importance in the overall Microsoft strategy.

But having something that is new, techy, and cool is typically not enough to convince a company to adopt it.

Implementing a new service from scratch or migrating an existing one is an effort in terms of money, time, and resistance to change—and this is also true for services like Azure Virtual Desktop that are easy and fast to implement and maintain.

So, before going deep into the architecture and technical details, I believe it's important to understand the value this service can offer to a company at business level.

The business values

We can identify at least three business values that Azure Virtual Desktop can address:

- Security and management
- Flexibility
- Productivity

I don't want to discuss why a company must take care of these needs; I want only to underline how Azure Virtual Desktop can help.

On the security side, we all know that users need to interact with devices, data, and applications to successfully perform their daily job, and many of these users need to interact with sensitive information, so the security and compliance topic is top of mind for every IT department.

Azure Virtual Desktop is a service centralized in Microsoft Azure and it's published in a secure way using the internet channel.

This is helpful because:

- A device that is allowed to access the production environment can be protected by several layers of Azure services that are embedded inside the service, or that I can

optionally implement (reverse connection, multifactor authentication, firewall, monitoring, security baselines, DLP, encryption, compliance policies, and so on).

- On the management side, the technologies offered by Azure Virtual Desktop allow you to easily maintain and keep the workplace up to date. Sometimes, it's not easy to manage, update, and monitor several physical devices around the world that are not always well connected. With Azure Virtual Desktop, thanks to the multi-session capability, I can have a small number of objects, centralized in Azure, and I can easily plan how and when to introduce an evolution (we will talk about that in the next chapters).
- The business data is centralized so it's easier to monitor, back up, protect, and prevent the leakage of sensitive information.

Azure Virtual Desktop is extremely fast to implement and to scale-in and scale-out, and it provides a lot of flexibility. It gives me freedom to support scenarios like the following:

- **Mergers and acquisitions**. Let's pretend that Company A acquires Company B, and they want to quickly let an employee from Company B use an application like CRM from Company A. Azure Virtual Desktop can help to quickly publish the CRM app to Company B users.
- **Temporary workers**. Lots of companies every day have several people like partners or contractors that are accessing their systems to perform maintenance, take on new projects, or perform seasonal work, maybe because it's Christmas time, summer, winter, Black Friday, or another heavy shopping moment of the year. They need to have more employees only for a small period of time. Buying physical devices for these temporary employees is an option, but at the end of the peak period, these devices will stay unused, and they will be getting old by the time the next peak happens. If I'm using an Azure Virtual Desktop solution, my extra employees can be productive and secure and, once the peak is finished, I can simply turn off everything and pay only the consumed time.
- **Branch workers**. Azure Virtual Desktop is a service published by the Azure cloud using the internet channel, so I can stay productive from any location on any device. Every time I need to work, I only need an internet connection, and I will be able to jump into my company environment without VPN systems, DMZ zones, and other options that are not always easy to implement and use.

Every company needs to allow employees to be as productive as possible in any condition, and Azure Virtual Desktop can help in doing that. This is not only because it's a service that is secure and easily accessible from any device, from any part of the world. We need also to consider the following:

- The power of Azure can let users have enough resources to perform their job. CPU power, RAM, and disk size are not a big issue in Azure. Do you need more power? In a matter of minutes, or maybe only seconds, you can have it!
- Specialized workers can have also specialized work devices. Thanks to the capability in Azure to have virtual machines with GPU integrated, it's possible to let users like engineers work with complex CAD applications (or other GPU intensive workloads). For example, an employee can carry only a small laptop, and when they need to use a graphic-intense application, they will be able to do that using the power of Azure. This also allows you to always have the last CPU/GPU technologies ready to be used.
- What about legacy applications? Maybe you have some very critical application that is not ready to work on Windows 10 or another modern operating system. But Windows 7 is out of support, so how can you let your users work? Do you really want them to work

in a Windows 7 environment? What about security, productivity, and user satisfaction? One solution is to publish the legacy application using Azure Virtual Desktop while for the other tasks, the user will use a modern operating system. (Microsoft has special advantages for this scenario, which we will talk about in the next paragraphs.)

On-premises VDI vs. Azure Virtual Desktop

You can say “Hey! The benefits that were presented in the previous paragraphs seem the same as a classic on-premises VDI solution. I don’t see the benefit of using Azure Virtual Desktop. I can simply build a similar solution inside my data center!”

Well, let me explain: Azure Virtual Desktop *is* a VDI solution with roots in the classic concept of “virtual desktop infrastructure,” so it’s true that most business needs and use cases that can be solved by this service are the same as an on-premises VDI solution.

But even if the main idea is old and common (a bunch of virtual machines publishing the whole desktop, or just a subset of their applications), Microsoft is evolving this concept and bringing many advantages to make Azure Virtual Desktop “the best virtual desktop experience.”

Licensing

Before going deep into the technical and cost advantages, let’s quickly cover the licensing. As I wrote in the previous paragraphs, Azure Virtual Desktop is a solution based on Windows virtual machines running on Microsoft Azure, so which kind of Windows OS can I install?

On this webpage, you can find that Azure Virtual Desktop supports the following x64 operating system images, including both Windows Client (Windows 10 and Windows 7) and Windows Server:

- Windows 10 Enterprise multi-session, version 1809 or later.
- Windows 10 Enterprise, version 1809 or later (Semi-Annual Channel only).
- Windows 7 Enterprise.
- Windows Server 2019.
- Windows Server 2016.
- Windows Server 2012 R2.

It’s up to you to decide what is the best choice. Maybe you need to publish an application that is compatible only with Windows Server, or you need to run an old application that is compatible only with Windows 7, or you’d like to let a user access a complete Windows 10 operating system from a thin client or iPad.

Whatever your architectural choice, the licenses required to access Azure Virtual Desktop are the following (source):

OS	Required license
Windows 10 Enterprise multi-session or Windows 10 Enterprise	Microsoft 365 E3, E5, A3, A5, F3, Business Premium Windows E3, E5, A3, A5
Windows 7 Enterprise	Microsoft 365 E3, E5, A3, A5, F3, Business Premium Windows E3, E5, A3, A5
Windows Server 2012 R2, 2016, 2019	RDS Client Access License (CAL) with Software Assurance

Figure 2: Azure Virtual Desktop Licensing

In a nutshell:

- Windows Client: VDA (Virtual Desktop Access) per user or a bundle like M365 E3 that includes VDA.
- Windows Server: RDS (Remote Desktop Services) CAL per user or per device with Software Assurance.

For example, if my solution is based on Windows 10 and I need to serve 50 users, I need to have at least 50 VDA per-user licenses. If the user is already covered by Microsoft 365 Business Premium, for example, they are already entitled, because this SKU includes VDA.

Otherwise, if my solution is based on Windows Server, I can cover it with an RDS CAL with Software Assurance and in this case, it can be both per user or per device. You can learn more about pricing here.

It's important to underline that the license needed by Azure Virtual Desktop is very similar to the one that Microsoft is requiring today for the other VDI/Terminal Server solutions. If you want to migrate an existing on-premises solution into Azure Virtual Desktop, this is helpful because at the license level, there is a good chance you are already entitled.

Because the resources (desktop, applications) that Azure Virtual Desktop is publishing are hosted on standard Azure virtual machines, I also need to calculate the cost of these virtual machines (vCPU, RAM, storage, outbound networking, and so on).

It's also interesting to underline that if the final users are external, I have the ability to let them access my resources using dedicated licensing.

In general, it could be helpful to use the dedicated section of the Azure Pricing Calculator (enter **Azure Virtual Desktop** in the search box and add it to the calculator).

Now it's time to talk about the advantages of adopting an Azure Virtual Desktop solution.

Technical advantages

Azure Virtual Desktop is the only way to use Windows 10 multi-session. We will talk more in depth about that in the next chapters, but I believe that you are already guessing the main advantage: a Windows client operating system that, like a Windows Server operating system, is able to serve multiple users at the same time.

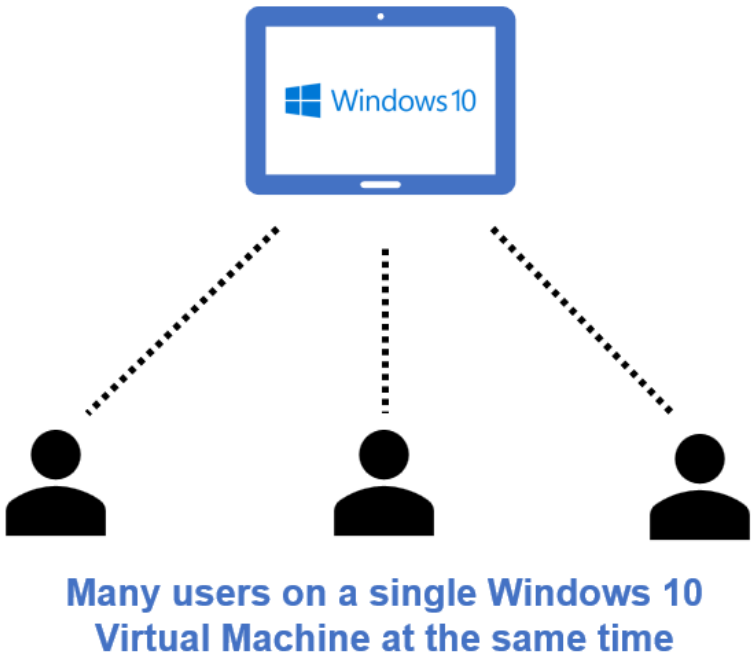


Figure 3: Windows 10 Multi-session

This is both a technical and a cost advantage. It's a technical advantage because typically in the on-premises world, I'm using Windows Server virtual machines to publish to my users because those VMs allow multiple sessions (and multiple sessions means more users on a single machine, which in turn means fewer virtual machines to implement, maintain, and pay for).

The problem with this strategy is that I'm forcing users to use Windows Server environments, and the average user is more comfortable in a Windows client environment.

And the applications that the users are using are written for Windows client, so letting them work on a Windows Server is not always straightforward (and not always supported).

With Azure Virtual Desktop, I can have the best of both worlds: I can have the efficiency of the multi-session, plus the productivity and support of a real Windows client operating system.

This is a big advantage that is not available outside Azure Virtual Desktop (the only exception is that selected partners like Citrix and VMware can publish Microsoft Azure virtual machines containing Windows 10 multi-session).

This solution is created in conjunction with the Microsoft 365 Apps team to ensure that everything is optimized and designed to work together smoothly.

Microsoft is also simplifying the publishing layer so the administrators can focus on creating and managing only the resources inside the virtual machines that will be published to the end users.

I believe all these concepts will become clearer once we will talk about the architecture and the technology part in the next chapters.

Cost advantages

Now it's time to talk a little more about the advantages of Azure Virtual Desktop.

Windows 10 multi-session

As I said in the last section, the presence of Windows 10 multi-session allows multiple users to share a single virtual machine (CPU, RAM, disk, network) at the same time.

This is a cost advantage because I need fewer virtual machines to serve multiple users.

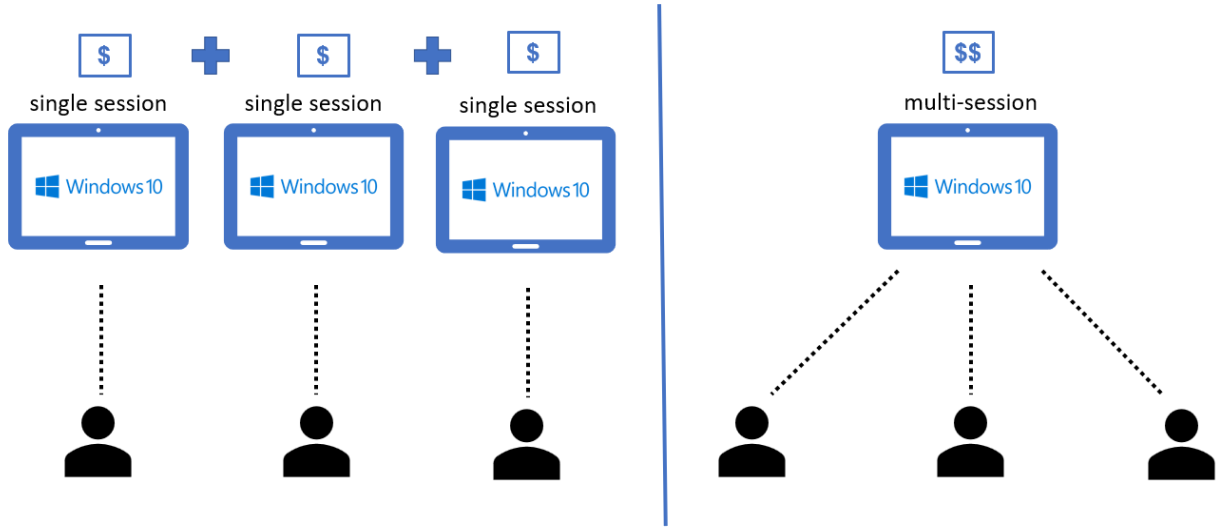


Figure 4: Windows 10 Single Session versus Multi-session

Without Windows 10 multi-session, if I need to provide a Windows 10 experience to 100 users, I need to set up and pay for 100 virtual machines—and I need to update 100 virtual machines!

Using Windows 10 multi-session, I have fewer virtual machines that are serving multiple users at the same time, so I pay less, and I spend less time managing these objects.

A multi-session virtual machine will need more resources (CPU, RAM) than a single session virtual machine because it needs to handle multiple users at the same time. That means a multi-session virtual machine is going to be more costly than a single session virtual machine, but the fact that I need to provision and maintain a fewer number of these objects will result in a cost advantage.

Linux rates

Another interesting cost advantage is that if I want to publish resources that are inside a Windows Server machine, I can do that, but unlike other solutions, Microsoft does not ask me for the operating system license.

As you probably know, Windows Server is not an open-source and free-to-use operating system. You need to pay a license fee that depends on several factors, but basically, it's a per-core license. You can find more information about that here.

If you are creating a Windows Server virtual machine inside any cloud, you need to pay for the license. In Azure you can add this cost to the overall monthly cost of the virtual machine (that is also including the cost of the CPU, RAM, disk, networking, minutes of usage, and so on), or you can buy it through a reseller and use the Azure Hybrid Benefit to use it inside Azure (more information here).

If you are creating a Windows Server virtual machine inside Azure and you are publishing the resources through Azure Virtual Desktop, you don't need to pay for the operating system license!

The cost of the virtual machine is calculated using the Linux rates, which means you are charged for the CPU, RAM, disk, and other resources allocated, but not for the operating system. The total cost is the same as an Azure virtual machine with a free Linux distribution installed.

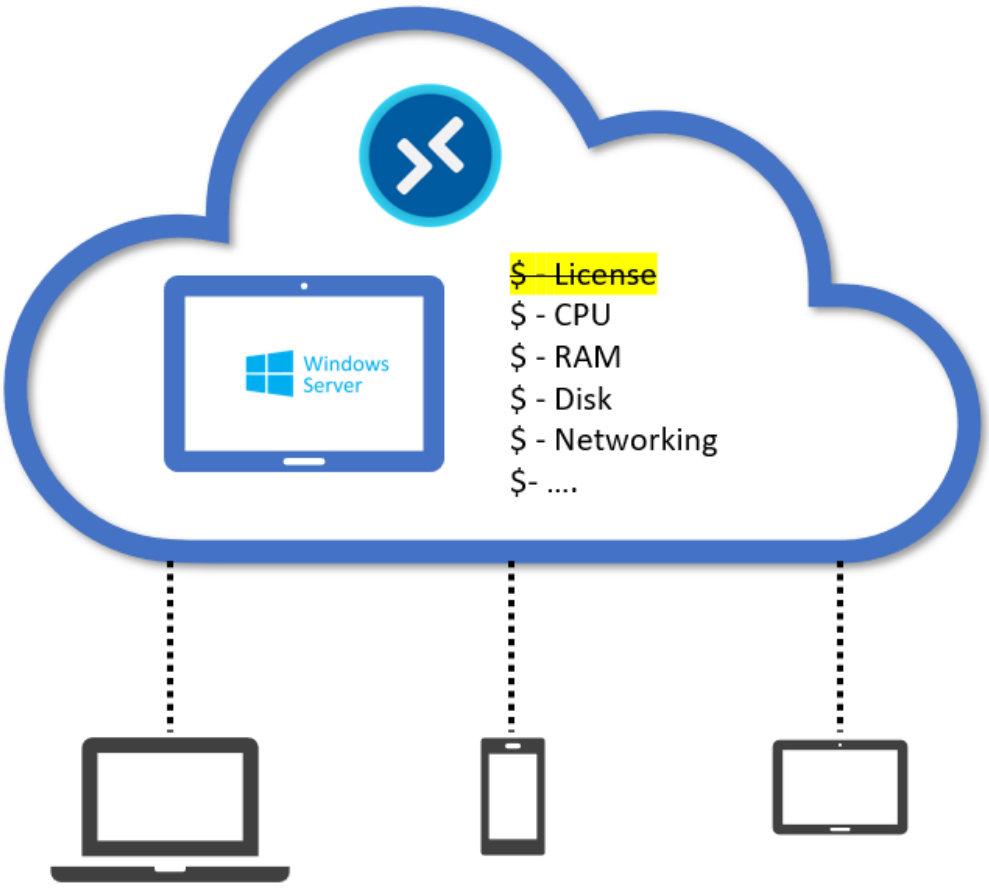


Figure 5: Windows Server Cost inside Azure Virtual Desktop

Extended support included

Windows 7 reached the end of service in January 2020. Microsoft stopped providing new security patches and the support channel stopped working on this operating system.

But Microsoft is offering special critical security updates until 2023 through extended (paid) support. It's an offering per year and per device, and it's a sort of a "last call" for customers that have difficulties evolving an application that is working only on Windows 7.

This sort of support could be quite expensive, and it's not particularly flexible. The cost is per year—what if I need this support only for 8 months? The answer is that I need to pay for the whole year.

In Azure Virtual Desktop, I can publish the desktop of Windows 7 machines (in this case only single session), and the advantage is that the extended support is included!

I can maintain my Windows 7 securely because of the Azure capabilities, and also because I am now entitled to apply the extended support security updates without paying for the extended support.

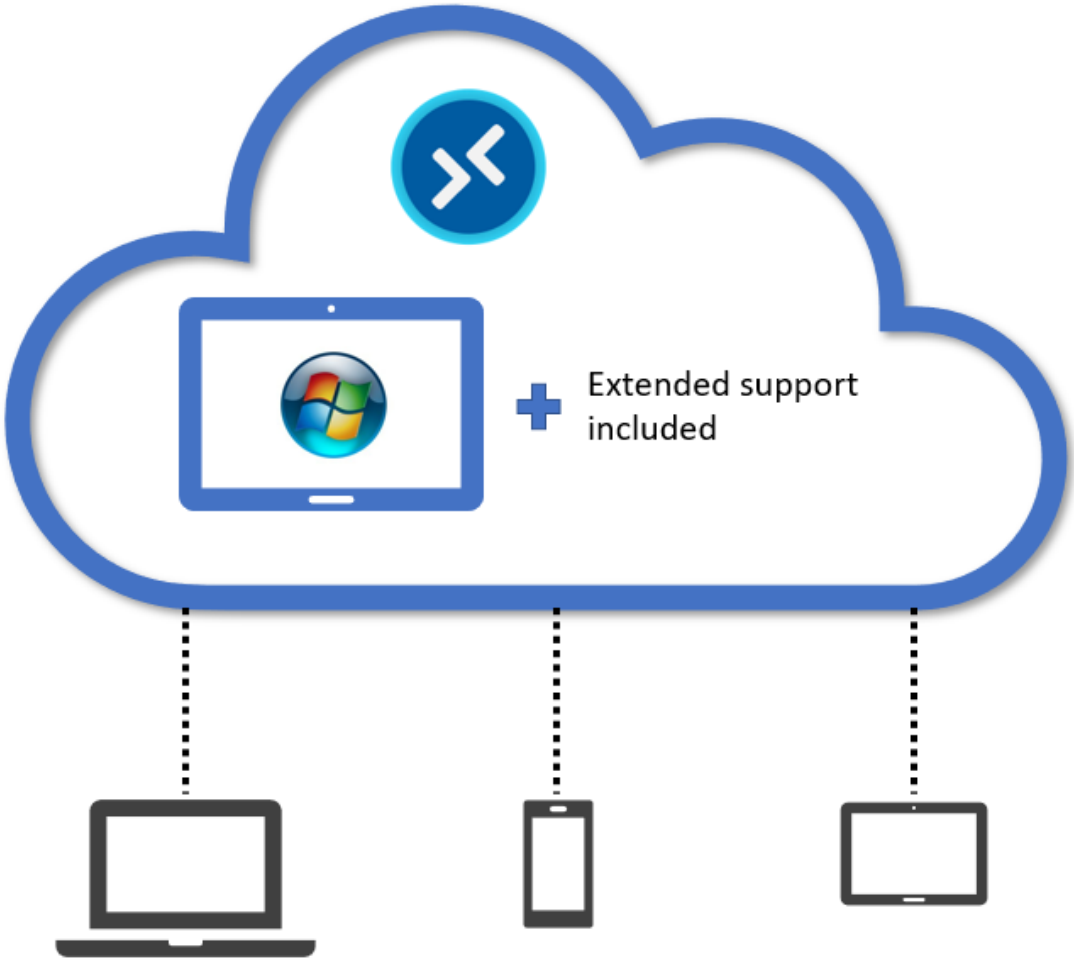


Figure 6: Windows 7 + Extended Support

I can give my users a new Windows 10 machine, MacOS, or other modern physical device and let them continue to consume legacy workloads inside a bubble in Azure that is composed of Windows 7 virtual machines that are up to date.

If, after some months, I am ready to let my users work natively with a new application that is now compatible with their modern devices, I will simply shut down the Azure Virtual Desktop farm and stop paying.

Avoid license double counting

Another interesting advantage is to avoid the cost of the RDS license. As I wrote in the section dedicated to licensing, if my Azure Virtual Desktop farm is composed of Windows client machines, I need to have a VDA per user license. Otherwise, if I use Windows Server in my VDI farm, I need to have an RDS license per device or per user with Software Assurance.

Before Azure Virtual Desktop and Windows 10 multi-session, one typical choice in the VDI world was to use Windows Server, because it was the only operating system that was able to handle multiple and concurrent user sessions.

Today, we have several customers that are paying for RDS CAL because it's the license that Microsoft is requiring to let a user access a Windows Server virtual machine exposed by a remote desktop service. But maybe the same customers are also paying for the VDA license because it's included in some bundles they have for other reasons, like email or security capabilities. One example could be Microsoft 365 Business Premium.

If I'm able to technically translate a solution based on Windows Server to another that is based on Windows 10 multi-session, and I am (or my customer is) already paying for the VDA license, I can introduce a double advantage because my new solution in Azure will not need the operating system server license and the RDS CAL.

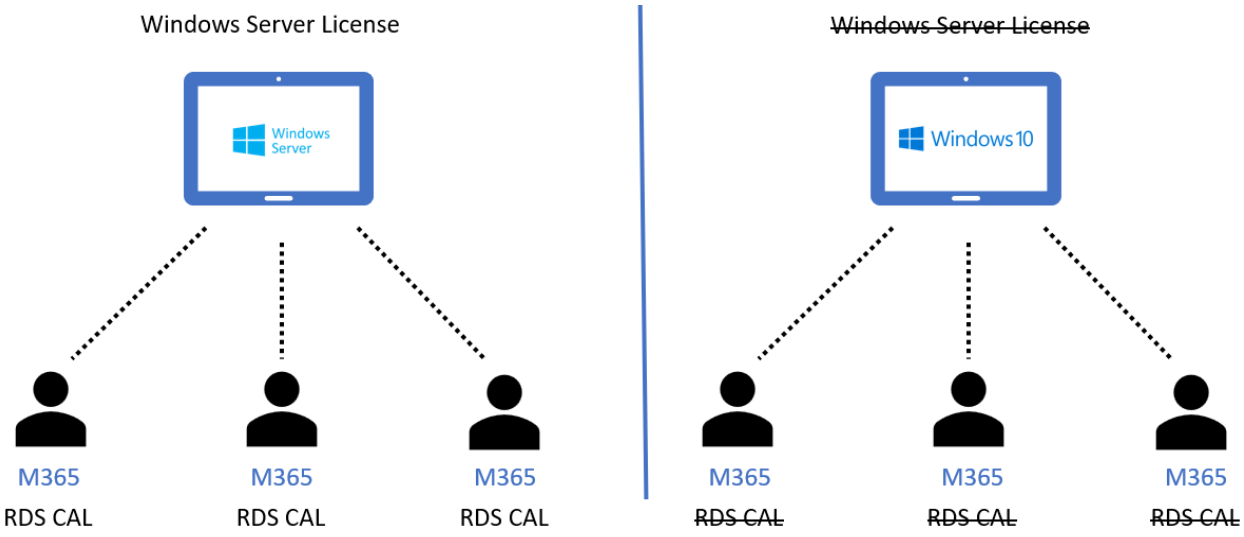


Figure 7: Azure Virtual Desktop License Advantage

Everything will be based on Windows 10 multi-session virtual machines that will allow me to serve multiple users at the same time as the Windows Server solution, but my users will be entitled to use this new solution without paying for the RDS CAL, because now it's based on Windows Client—so the VDA that they already have inside their macro plans is enough.

Publishing layer

The last cost advantage is related to the cost of publishing the resources.

In a typical VDI solution that involves the publishing of virtual machine resources, I need to architect, deploy, and maintain a publishing layer composed of several roles installed on several virtual machines that are providing different roles. This publishing layer will receive requests from the end users, and the different roles will apply and check licenses, assign resources, and monitor.

Any architecture needs to be scalable, and when you take into account business continuity, disaster recovery, geo replication, and the continuous evolution of the platform, this publishing layer can become quite complex and expensive to create and maintain.

In Azure Virtual Desktop the publishing layer is free of charge, and it's totally created and maintained by Microsoft.

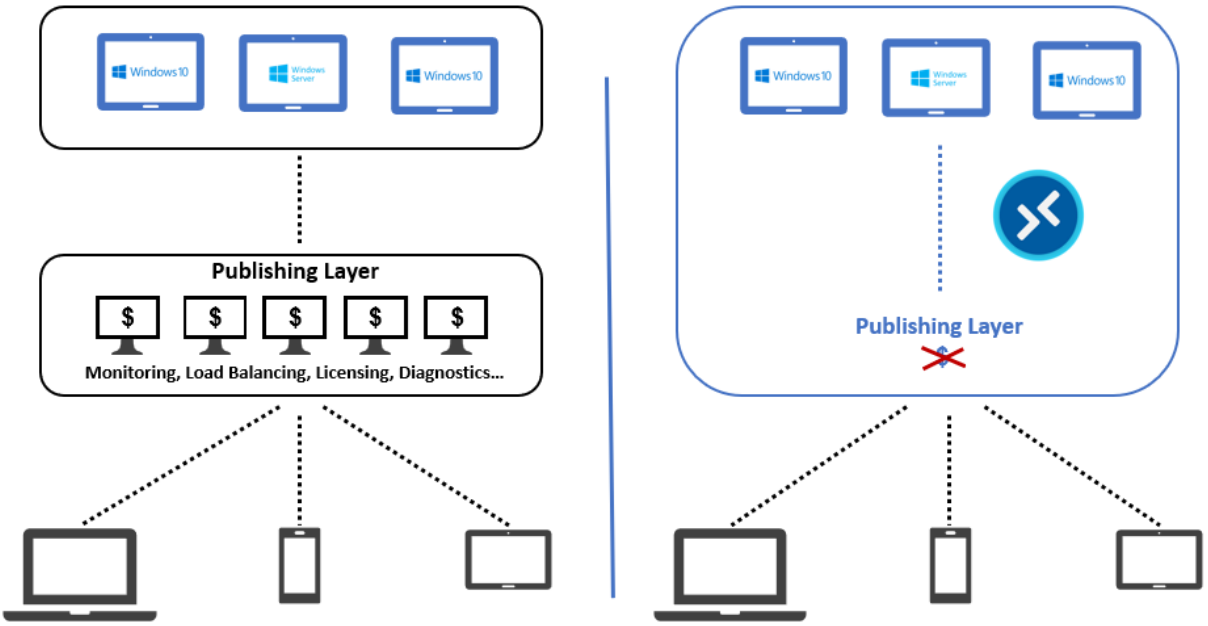


Figure 8: Azure Virtual Desktop Control Plane

This service is called *control plane*, and it's available worldwide, so I can create a solution that is always available and can serve my users in different locations, and everything will be managed as a service by Microsoft.

In a nutshell: I don't need to create, configure, and maintain the publishing layer.

Main adoption scenarios

In general, the adoption of Azure Virtual Desktop is linked with three macro scenarios:

- Migration of an existing solution.
- Support of legacy applications.
- Implementation of new use cases.

Let's briefly discuss each of these bullet points.

Migration of an existing solution

In this case, I have an existing solution that is publishing Windows resources hosted outside Azure Virtual Desktop. Maybe it's an on-premises solution; maybe it's a solution deployed in another public cloud.

For several reasons (like the advantages that we discussed in the previous paragraphs), I'm willing to migrate the solution to Azure Virtual Desktop.

Support of legacy applications

As we already know, I can create a solution based on Windows 7 or Windows 2012 R2 so I can support the consumption of legacy workloads.

Today, it's not the best choice to give Windows 7 devices to our users because it's no longer secure, up to date, and productive like Windows 10 or other modern operating systems.

It can also become quite difficult because Windows 7 does not support new hardware and new standards like UEFI.

The best choice is to have a productive, secure, and happy end user who is using a modern physical device and consuming legacy applications using virtual machines inside the Azure cloud published through Azure Virtual Desktop.

Implementation of new use cases

Azure Virtual Desktop has opened several use cases thanks to the presence of Windows 10 multi-session: the deployment speed, agility, power, and flexibility of the Microsoft Azure cloud.

It's now possible to provide the end user a true and complete Windows 10 experience from any device and any place. This is very helpful for smart working.

I can use Azure Virtual Desktop to help companies' mergers and acquisitions—it's quick and easy to let users from company A use applications from company B in the meantime until the acquisition is completed.

I can use this service inside my security strategy because I can let external users connect to my production systems only using an Azure Virtual Desktop machine that I manage. For example: are you using a laptop that belongs to a third-party company, where you don't manage the security of the device? You can access my production systems, but you must use my Azure Virtual Desktop machines that are the only ones that can access them.

This also helps bring your own device (BYOD) strategies.

Inside Azure I can request the creation of very powerful virtual machines with lots of CPU, RAM, disk, and GPU power.

If I have a group of users that are working with CAD applications, I can let them work with all the power they need in every location using a standard laptop, because the graphics power that they need will be delivered by Azure Virtual Desktop.

Inside Microsoft Azure I can create virtual machines with GPU from NVIDIA or AMD that can provide all the graphics power that my users need, whether they are working from home on a small tablet, or from a building construction site in another country with a standard laptop.

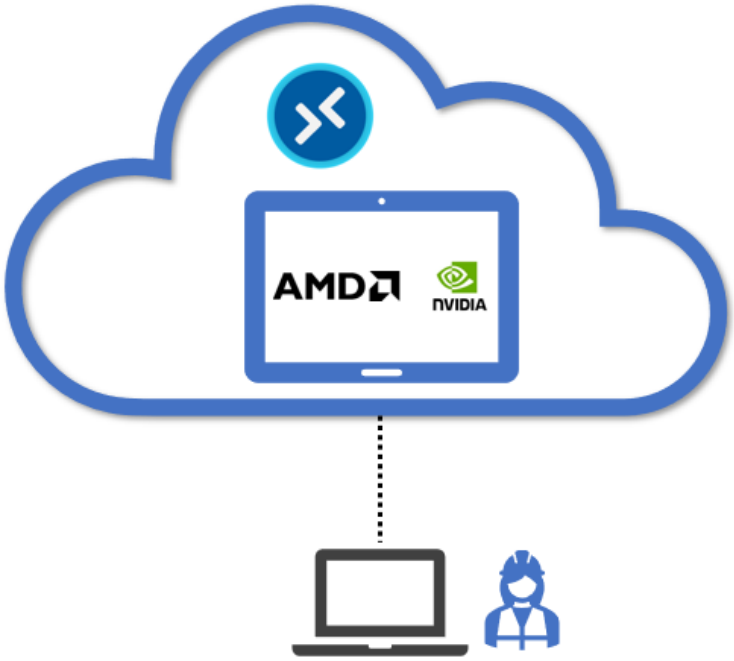


Figure 9: Azure Virtual Desktop GPU Power

These are only examples; you can find several use cases where Azure Virtual Desktop is useful and convenient.

Chapter 2 Architecture

Since this is a *Succinctly* guide, I don't have room to go deep inside all the architectural aspects of an Azure Virtual Desktop solution. But I will cover a number of core topics that will be useful to understand the whole picture.

A simple architecture

The following figure shows what is probably the simplest Azure Virtual Desktop architecture that can be implemented.

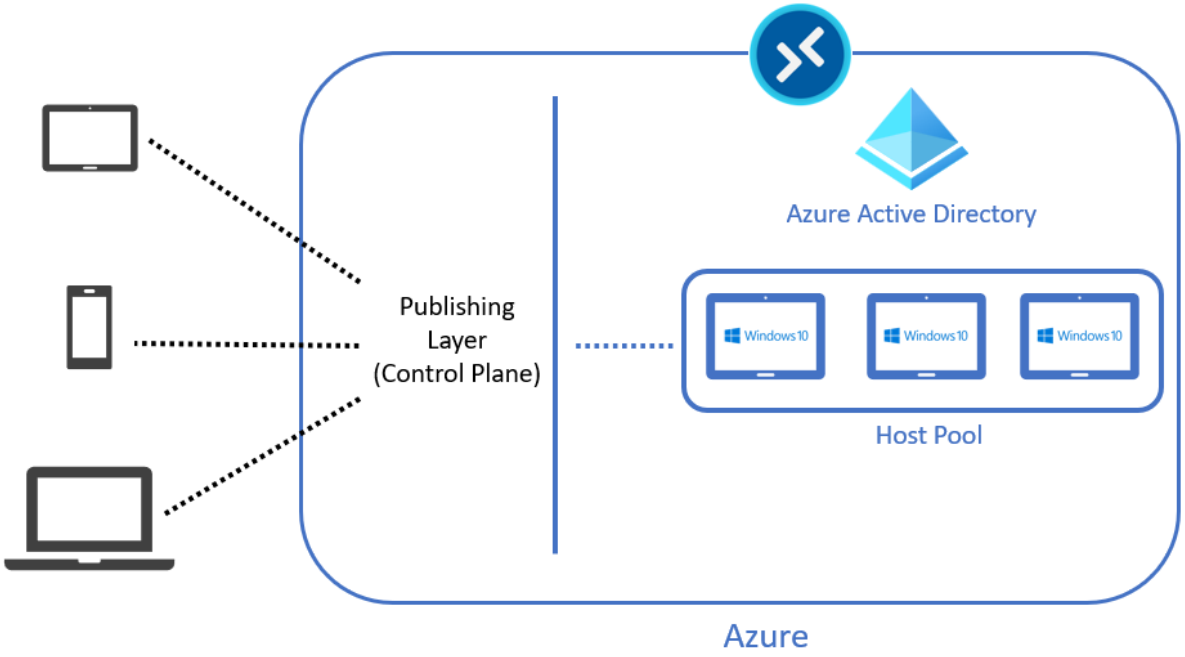


Figure 10: Simple AVD Architecture

In a nutshell, we have different physical devices that are using native applications or web browsers to contact the Azure Virtual Desktop service through the internet channel on port 443.

The service is redirecting the communication to the publishing layer (called the control plane), and inside the control plane we have different roles that are performing several activities, like load balancing, licensing, and monitoring.

The most important one is the Remote Desktop Gateway (RD Gateway) that is responsible for creating a connection (called reverse connect) between the physical device and the virtual machine of destination, which is joined to Azure Active Directory and logically placed inside a group of machines called the host pool.

If you are worried about the fact that the virtual machines are joined to Azure Active Directory only, you have the option to join them to Active Directory only, or to perform a hybrid join. Obviously, the virtual machines need to have a “line of sight” with the domain controller; that means the capability of talking with the domain controller and performing a domain join.

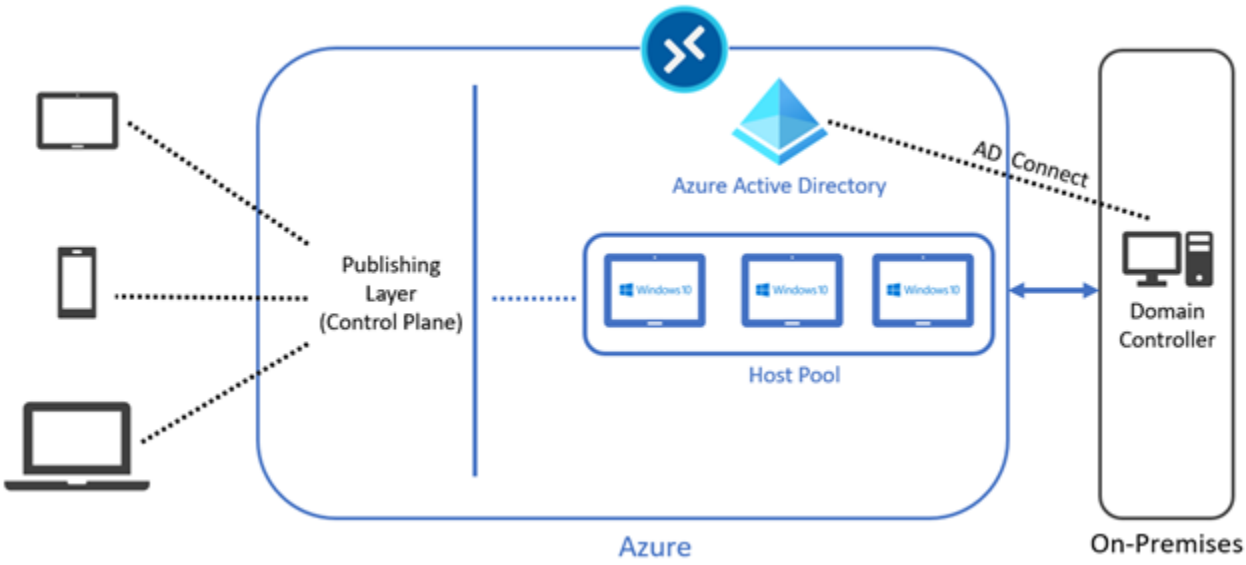


Figure 11: Hybrid Architecture

This is the high-level authentication flow:

1. The user authenticates against Azure Active Directory (because Azure Virtual Desktop is an Azure Service, and it’s relying on Azure Active Directory for authentication).
2. Optionally, multifactor authentication and conditional access rules are applied.
3. The physical device call is redirected to the nearest control plane (Microsoft has deployed several of them in key regions worldwide).
4. Several things are happening inside the control plane, and in the end, the Remote Desktop Gateway role is creating the connection between the physical machine and the assigned Azure Virtual Desktop virtual machine.
5. The user is logging onto the virtual machine using Azure Active Directory credentials or Azure AD credentials (depending on whether the architecture is cloud only or hybrid/on-premises).

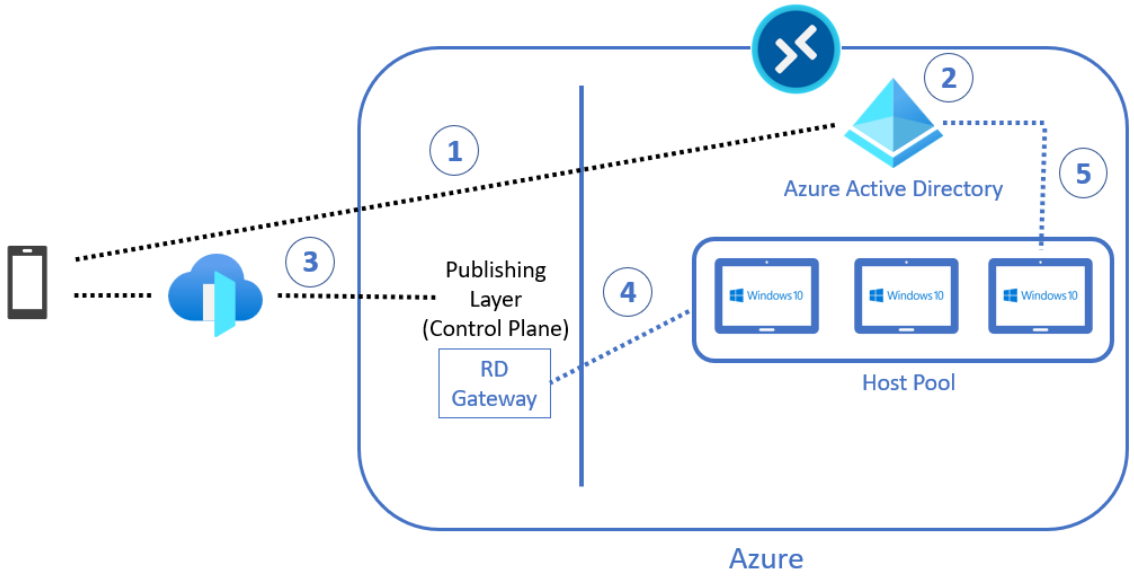


Figure 12: High-Level Authentication Flow

Typically, an Azure Virtual Desktop architecture is more complex than the one shown in the previous figures. It needs to account for other factors like:

- Backup, business continuity, and disaster recovery.
- Storage account needed for FSLogix and MSIX app attach (we will talk about these technologies in the next chapters).
- Monitoring.
- Scaling logic and automation.
- Custom base images.
- On-premises connection.

You can find useful information about how to architect an Azure Virtual Desktop solution here.

Note: Currently, the ability to join virtual machines only to an Azure Active Directory domain is in preview; please check the documentation.

Pooled vs. personal

When it's time to plan a virtual desktop solution, whether we are talking about Azure Virtual Desktop or another solution, one of the main choices is related to creating a group of pooled virtual machines versus personal virtual machines.

In the first case, the pooled solution allows more users to share the same virtual machine at the same time.

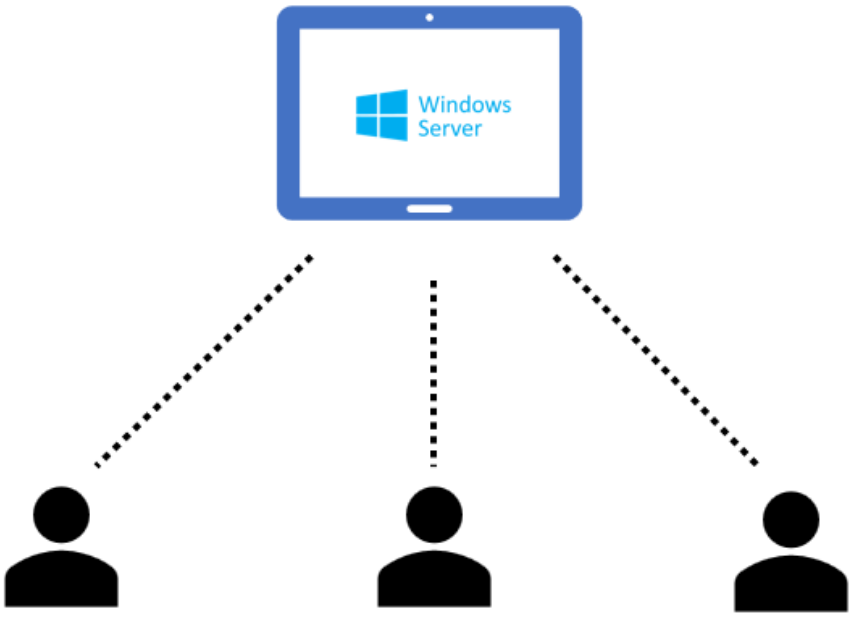


Figure 13: Generic VDI Service Using Multi-session

This is an advantage because I can share the resources of one virtual machine (like CPU, RAM, disk space, and network interface) between several users, and this is good in terms of costs and simplicity (fewer virtual machines that need to be patched and managed).

On the other side, we have the single session type, which means one machine per user.

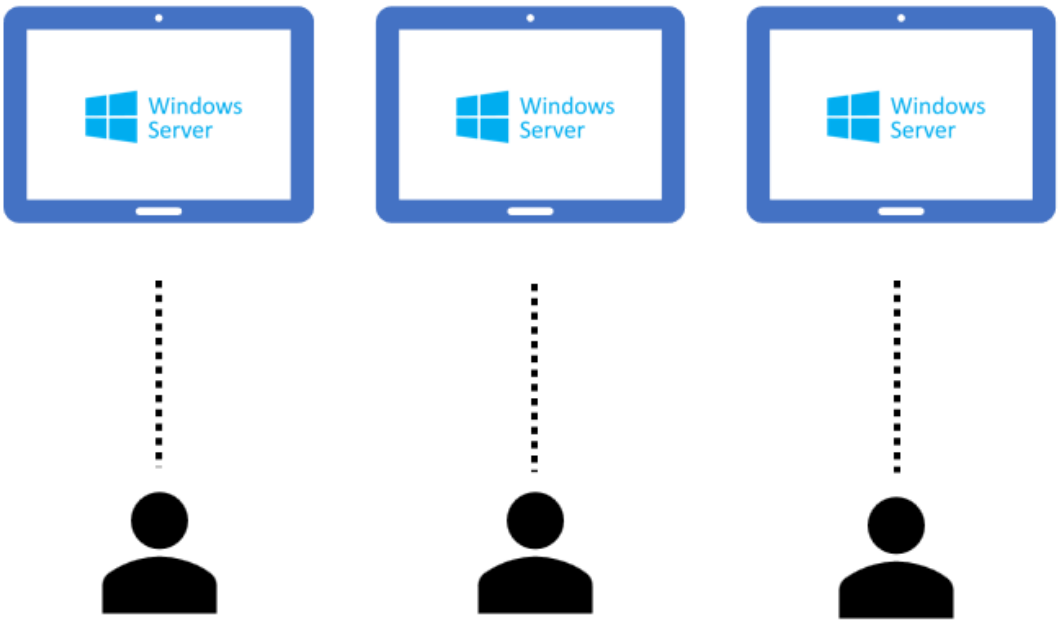


Figure 14: Generic VDI Service Using Single Session

The advantage here is that you can offer a complete separation of computational power, and a complete separation of tools (I'm not sharing my work device with anyone else).

If I have 300 employees and I'm offering a single-session solution, I need to have 300 devices that I need to pay and maintain. With a multi-session strategy, I can let 100 devices serve three users per machine at the same time and save money.

The number of users that a single machine can serve at the same time depends on several factors, such as:

- The style of a user (basically the applications that the user is using every day).
- The power of the machine (such as number of processors, memory, and disk amount).
- Security requirements.

A best practice is to define user personas and have one or more pilot activities to tune the architecture. Maybe a single machine can handle 16 users alone, or maybe only two at the same time.

You can see from the last two figures that Windows Server is able to provide multi-session capabilities, but what about Windows client?

The answer is that Windows client is a single-session operating system, so it's typically used in the VDI services when it's better or required to build a personal solution.

In Azure Virtual Desktop, it's possible to also have Windows 10 multi-session, which is a flavor of Windows 10 that can handle multiple users at the same time.

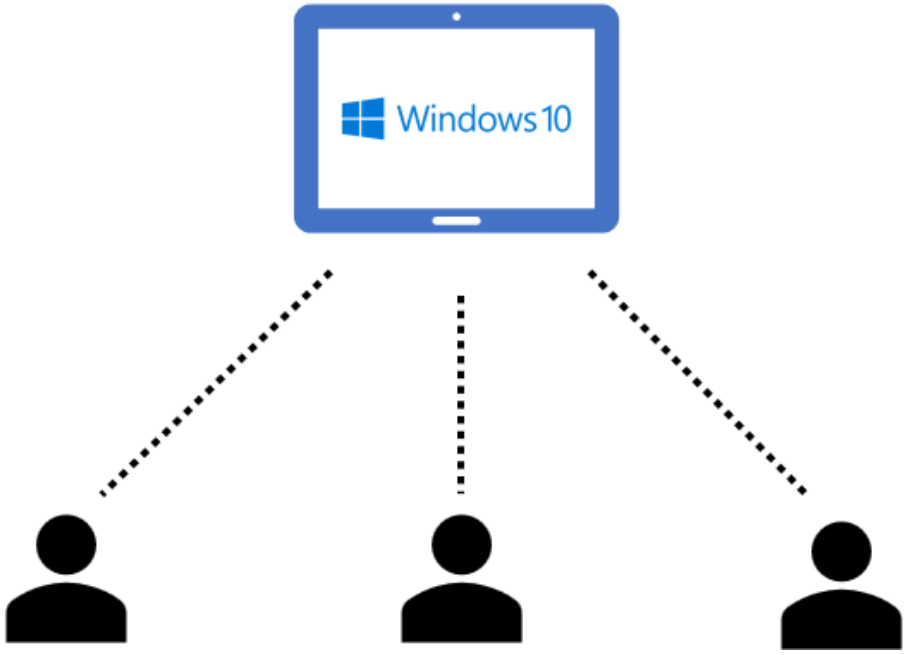


Figure 15: Windows 10 Multi-session

This interesting version of Windows is also available in Azure Virtual Desktop, or by using VMware and Citrix solutions that publish Azure virtual machines (currently it's not possible to run Windows 10 multi-session outside Azure Cloud, and it must be published by Azure Virtual Desktop or by the other two approved providers).

Identity

Azure Virtual Desktop is a service integrated into Azure Active Directory, so basically when I assign a desktop or an app to a specific group of users, I'm using Azure Active Directory groups and Azure Active Directory users.

If my Azure Virtual Desktop solution is cloud only, I can use cloud-only groups containing cloud-only users to assign resources that are working inside Azure virtual machines that are joined only to Azure Active Directory.

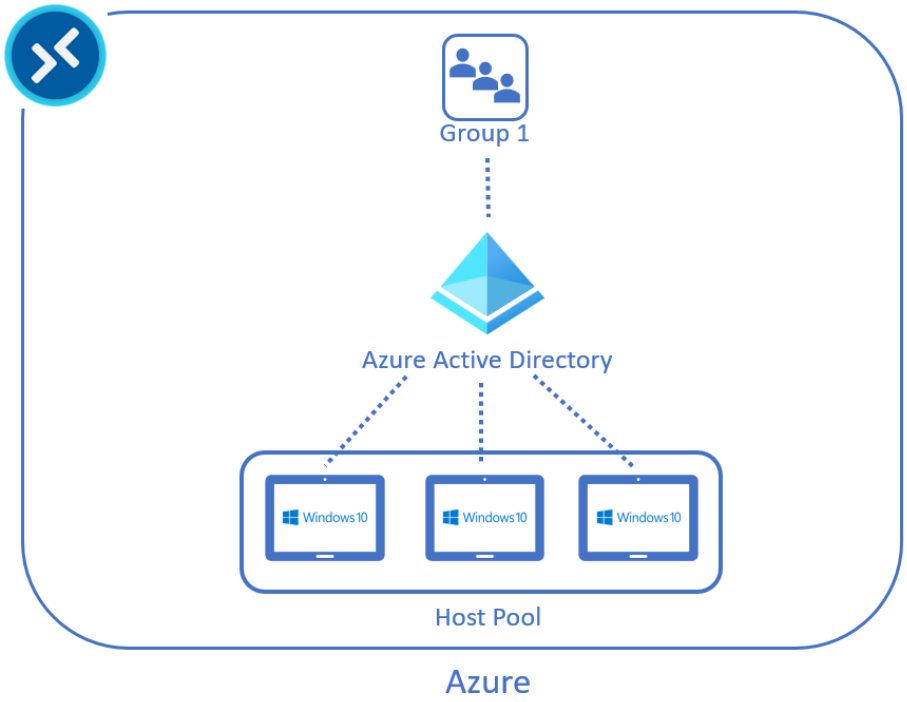


Figure 16: Azure Virtual Desktop Architecture Cloud Only

If I have an on-premises Active Directory domain controller, I can create a hybrid architecture, but in this case the Azure virtual machines must be joined to the on-premises Active Directory (hybrid join is supported). The users must exist in both Active Directory and Azure Active Directory, and this is possible using an Active Directory connection (AD Connect).

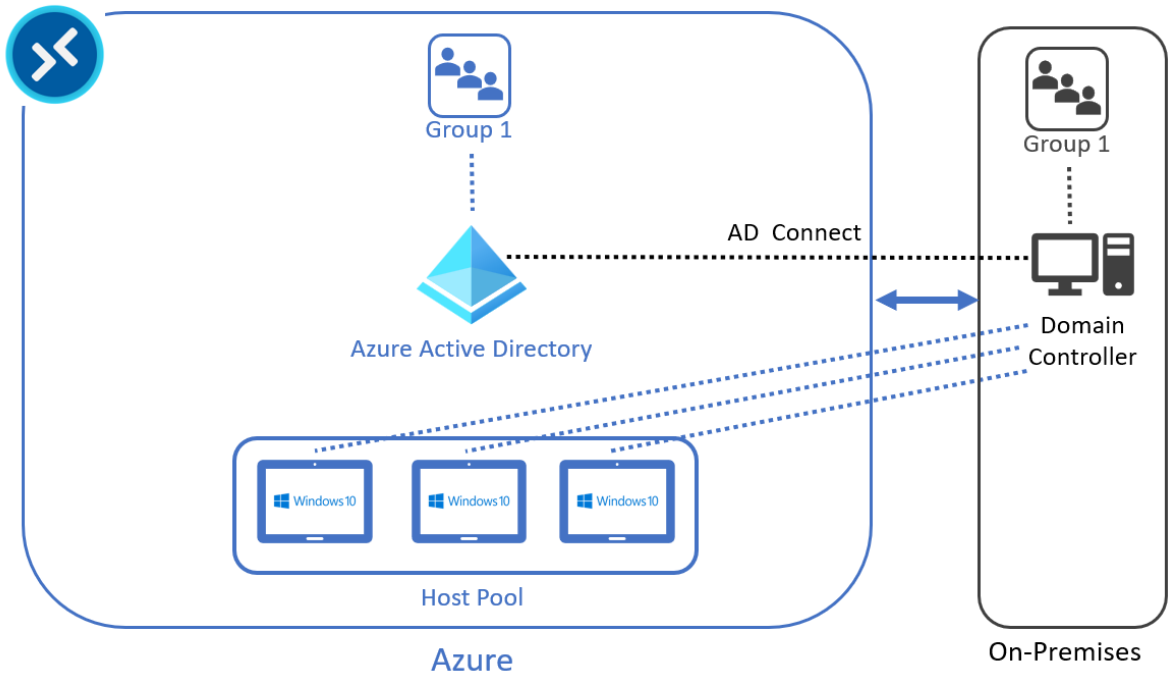


Figure 17: Azure Active Directory Architecture Hybrid

Regarding the authentication of these hybrid accounts, it's supported for the usage of Active Directory Federation Services, pass-through authentication, and password hashing.

On the connectivity side, both VPN site-to-site and ExpressRoute solutions are supported.

Depending on the architecture, the domain controller can (or must) be placed inside a virtual machine in Azure or left on-premises. Azure Active Directory Domain Service is also supported.

As you can see, Azure Virtual Desktop is very flexible when it comes to having virtual machines joined to an Active Directory domain (virtual machines in workgroup are not currently supported).

End user connection

Once the architecture is defined and the service is online, how is my end user accessing the service?

This could be done in three ways:

- Native applications (called clients).
- Web browsers.
- Thin clients (using the official Linux SDK or Windows OS).

Regarding the native applications, Microsoft is supporting Windows, macOS, iOS, and Android.

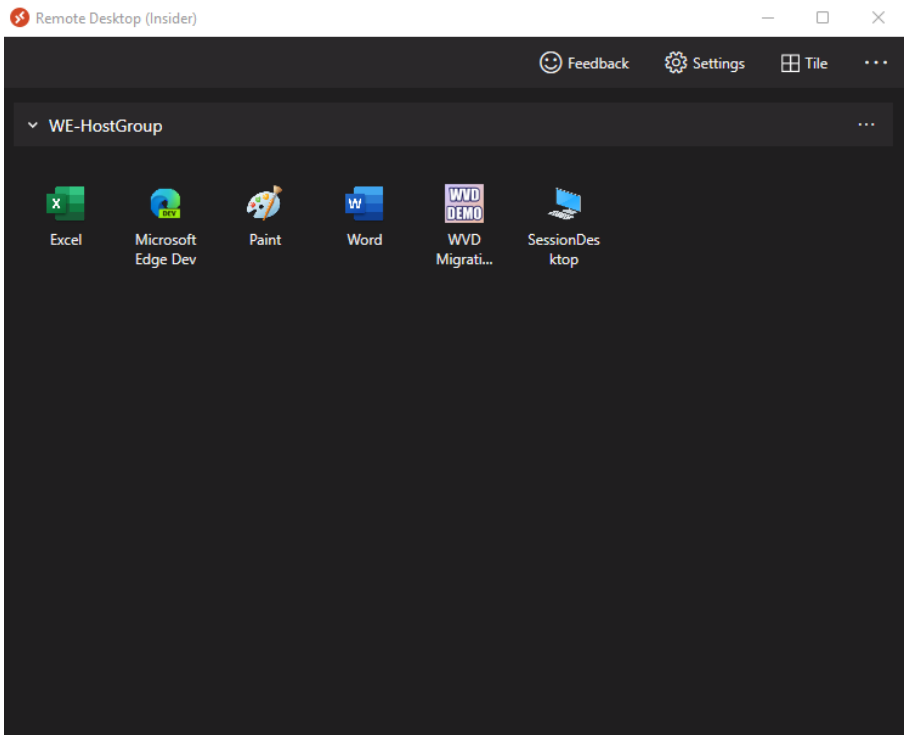


Figure 18: Windows Client Showing Azure Virtual Desktop Resources

You can find more information here.

Regarding the web browsers, basically every HTML 5.0 browser can connect to the Azure Virtual Desktop service, but Internet Explorer 11 is no longer supported.

The thin client space is composed of several vendors that are building their solution around the official Linux SDK or Windows 10 for IoT. You can find the official list here.

In general, it's possible to use Azure Virtual Desktop from any device, but please read this article showing the differences of experience between the various clients and the web browser.

Azure Portal

Azure Virtual Desktop is an Azure service that is completely integrated into the Azure Portal and the Azure Resource Manager (ARM) logic.

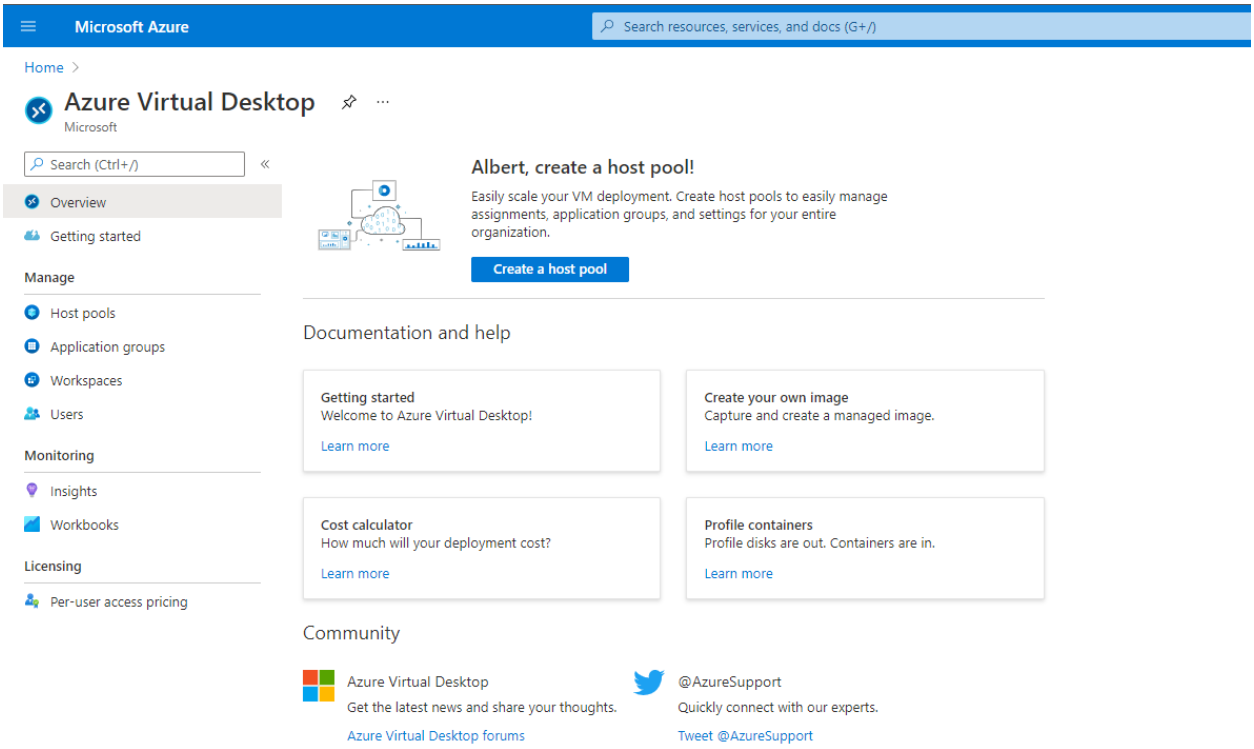


Figure 19: Azure Virtual Desktop Management Console inside the Azure Portal

From this console, it's possible to create and manage host pools, assign resources, and monitor the user/machine activities.

The various management activities can be performed by different groups of users that have different permissions based on the built-in RBAC (role-based access controls) roles or custom ones.

Virtual machine sizing

Which kind of Azure virtual machine do I need to use for Azure Virtual Desktop? How many CPUs? How much RAM? If I'm using multi-session, how many users per virtual machine I can support?

This depends on the type of workload and the style of usage, so there is not an exact answer, but Microsoft provides guidance in this official article.

Table 1: Multi-session Recommendations

Workload type	Maximum users per vCPU	vCPU/RAM OS storage minimum	Example Azure instance	Profile container storage minimum
Light	6	8 vCPUs, 16 GB RAM, 16 GB storage	D8s_v4, F8s_v2, D8as_v4, D16s_v4, F16s_v2, D16as_v4	30 GB
Medium	4	8 vCPUs, 16 GB RAM, 32 GB storage	D8s_v4, F8s_v2, D8as_v4, D16s_v4, F16s_v2, D16as_v4	30 GB
Heavy	2	8 vCPUs, 16 GB RAM, 32 GB storage	D8s_v4, F8s_v2, D8as_v4, D16s_v4, F16s_v2, D16as_v4	30 GB
Power	1	6 vCPUs, 56 GB RAM, 340 GB storage	D8s_v4, F8s_v2, D8as_v4, D16s_v4, F16s_v2, D16as_v4, NV12, NVv4	30 GB

In general, the D-series and F-series perform very well in a multi-session scenario.

Network sizing

Like virtual machine sizing, network sizing depends on several factors. Microsoft has an official article about the recommended bandwidth, but the best practice is to test it in a small real environment first.

One factor that influences the overall user experience is the network latency and the distance from the nearest control plane. This value can be tested using the Experience Estimator, a website that shows the round trip from the testing device and the nearest control plane.

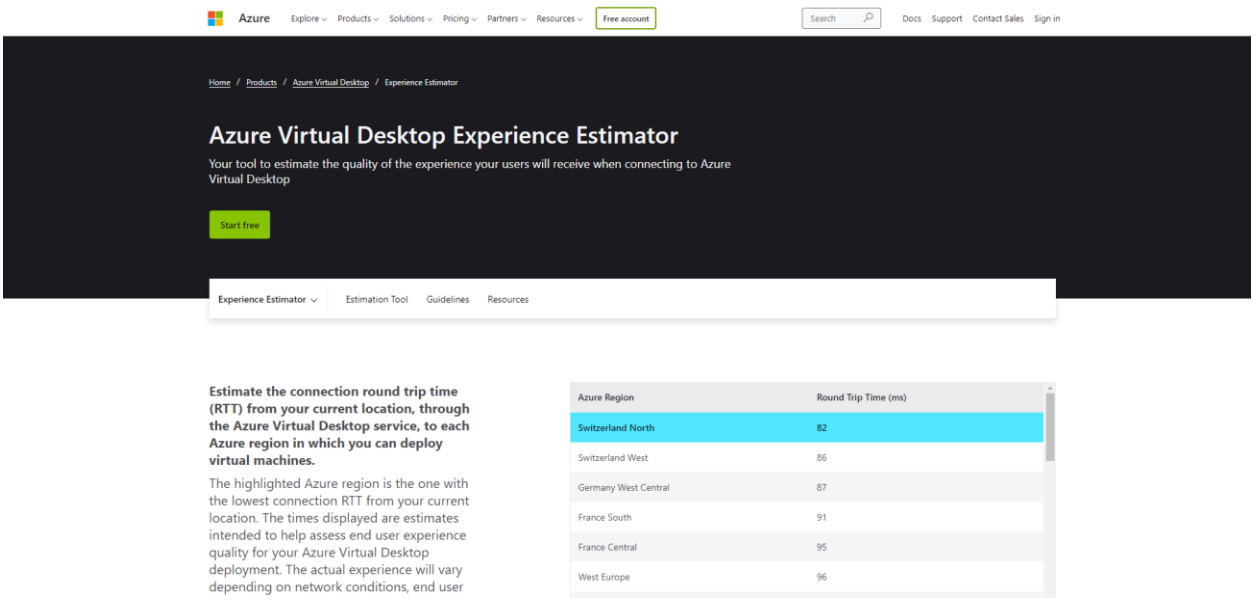


Figure 20: Azure Virtual Desktop Experience Estimator

In this example, my physical test machine is in Italy, where currently no control plane is available, but I have a good round trip time with Switzerland North and other Azure regions that have a control plane ready to serve my connection request of connections.

A round trip around 120 milliseconds is good for general usage.

Scaling logic

Cloud is all about automation, and the Azure Virtual Desktop product team has created a scripting logic that optimizes the number of virtual machines that are started based on the request of connections.

This automation uses Azure Logic Apps and Azure Automation to start existing virtual machines only when the ones that are already started cannot handle any new connections.

Using this technique, it's possible to save money because I'm paying based on the daily requests.

Let's look at an example.

I have a pool of 10 virtual machines that can serve five users each. I can instruct the scaling logic to have only 1 virtual machine started at 8:00 AM.

When the fifth user requests the service, the logic starts my second (existing) virtual machine, so I will be able to serve five more users.

At 6:00 PM I can decide that it's time to automatically shut down all the virtual machines, and if a user is still connected, they will receive a message saying something like, "I need to shut down this machine for maintenance. Please save your work. You will have 10 more minutes."

But this is just an example; the logic is highly configurable in the hours and the actions (maybe I don't want to force a user to log off).

You can find more information about this scaling logic here.

FSLogix

Back in 2018, Microsoft announced the acquisition of an Atlanta-based company called FSLogix. Microsoft made this acquisition because the solution developed by FSLogix was very interesting.

What is the goal of FSLogix? The tool does several things, but the main one is profile separation. It's possible to maintain the user data and preferences outside the device and store everything in a VHD/VHDX file that can be attached at the time of login.

For example, we can have a situation with a host pool composed of three Windows 10 virtual machines that are created from the same base image. All the machines have the same application stack and configurations.

Thanks to FSLogix (an agent installed inside each virtual machine), the user profile is maintained in a share inside Azure (for example in an Azure Files file share), and it's attached at the time of login. Let's pretend that the first day of usage, the user is redirected to the first virtual machine of our pool, so the VHD/VHDX file that contains their user profile is attached to virtual machine 1.

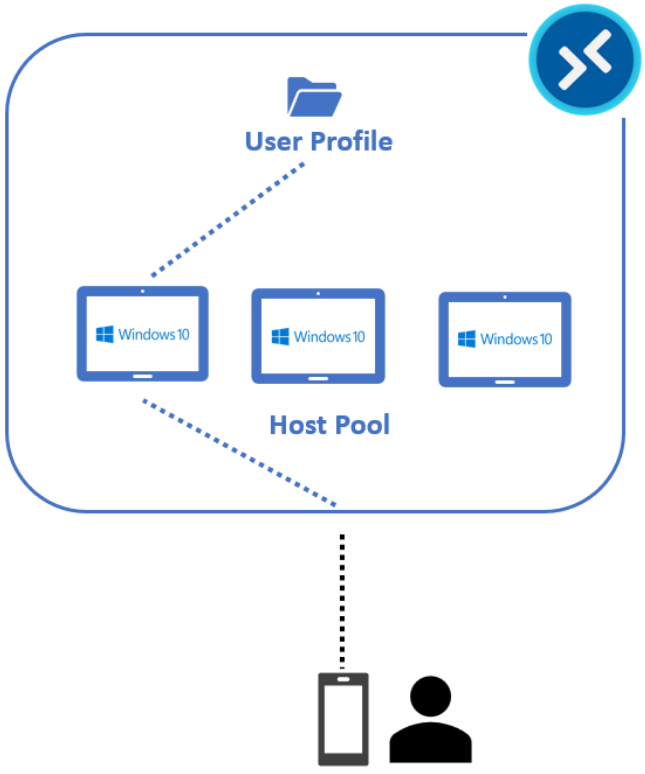


Figure 21: FSLogix User Profile Attach on Day 1

On the second day, the user requests the same service, but now the system is assigning the third virtual machine of the pool, so the VHD/VHDX file is now mounted into the virtual machine 3 file system.

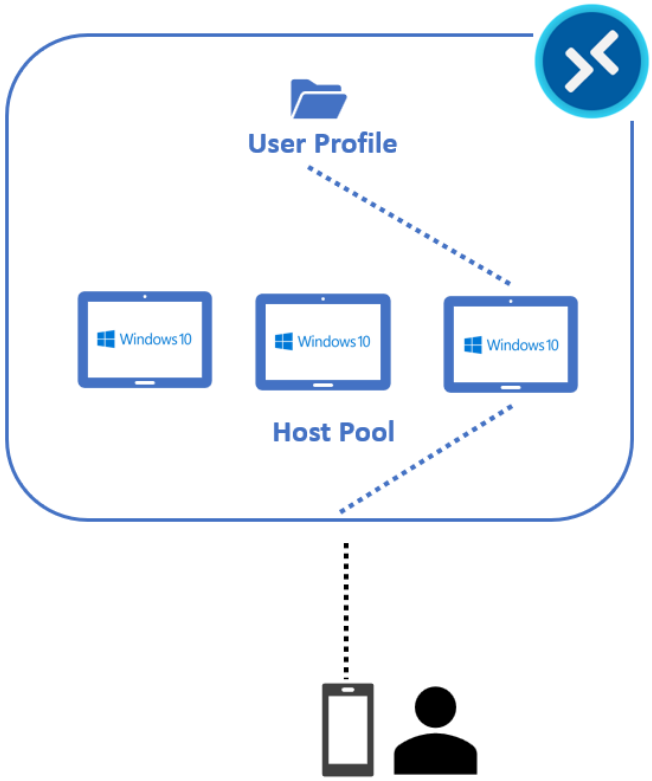


Figure 22: FSLogix User Profile Attach on Day 2

The user can stay productive, and the virtual machines are expendable because they are created from the same base image. All I need to do is protect the file share that is maintaining the user profiles.

This is common in the VDI space and with other vendors as well. Microsoft has other technologies that can achieve the same result, but FSLogix has some advantages:

- The agent is very lightweight.
- The cache of Office and the index of Windows are preserved inside the user profile, so when I'm logging into a new virtual machine, I don't have to wait for the build of these files.
- It can be simply managed using group policies of registry keys.

Regarding the storage that is maintaining the profiles, I have three choices:

- Azure File Storage
- Azure NetApp
- Storage Spaces

The first one is Azure File Storage, an Azure service that is the mostly commonly used solution in Azure Virtual Desktop. It can fit many different scenarios, from proof of concept to production environments (especially with the Premium SKU).

The Azure NetApp solution is based on a virtual appliance made by NetApp. It's a very fast and reliable technology that can be used in big complex deployments.

The last one is based on Storage Spaces—that means Windows Server virtual machines acting as file servers. This is recommended only for very basic solutions like proof-of-concept or small-production deployments due to the basic performance that it's providing.

You can find more information here about the storage choice for your profile.

Regarding the IOPS (input-output operations per second) requirements, there is no official guidance, but consider that each profile is consuming around 10 IOPS in a steady state, and around 50 IOPS at log on/log off.

Azure Files premium can provide up to 100K IOPS, and Azure NetApp up to 320K IOPS, while Storage Spaces depends on the type of physical hard disks that are used (HDD or SSD).

Microsoft recently increased the performance of Azure Files premium, as you can read here.

The general guidance is that for small environments (fewer than 200 users), an Azure File standard share should be enough, while for larger environments, it's better to use premium (or standard with multiple shares).

For very large and demanding environments, Azure NetApp is the solution that can provide the best performance and latency.

On the licensing part, the requirements for FSLogix are the same as Azure Virtual Desktop, so it's something that is included in every Azure Virtual Desktop deployment, and it's a must-have if pooled hosts are used. The official documentation can guide you in setting up this solution.

It's also possible to use FSLogix to mask the presence of an installed application. For example, we can have a host pool used by two different business units:

- The finance group that is using applications A, B, and C.
- The HR department that is using A, B, X, and Y.

At this point I have two choices: one is to create two different host pools, one for finance, and one for HR, and installing the needed applications in each pool.

The second possibility is to create only one pool that is used by the two departments, and install inside each virtual machine all the applications needed by all the departments (A, B, C, X, Y).

Finally, I will use the Application Masking feature to selectively hide the unwanted applications, so on the same machine, a user from the finance department will be able to see only the apps A, B, and C—even if X and Y are also present. They are masked by FSLogix, so they're not present in the Start menu, in the file system, or in the registry.

On the other side, a user from the HR department can see A, B, X, and Y, but not C, because C is masked for the group of HR employees.

You can read more information about this feature here.

The last helpful feature that FSLogix provides is Java redirection. This is a feature that forces an application to use a specific version of Java.

A common example is that our finance department is using application A that needs Java version 1, but also application B that needs Java version 2.

On the other side, we also have an HR department that is using application X, which needs Java version 2.

How do we satisfy all these requirements? One solution is to create three host pools: two for the finance department, and one for the HR department; and install the applications with the required versions of Java. But this is not optimal in terms of cost and time for managing all these objects.

The best choice is to create just one pool, install all the applications needed (A, B, and X) and both Java versions (1 and 2), and use Java redirection to force application A to use Java 1, and applications B and X to use Java version 2.

Obviously, I can also use the Application Masking feature to let users from the finance department see only app A and B, and users from the HR department to see only application X.

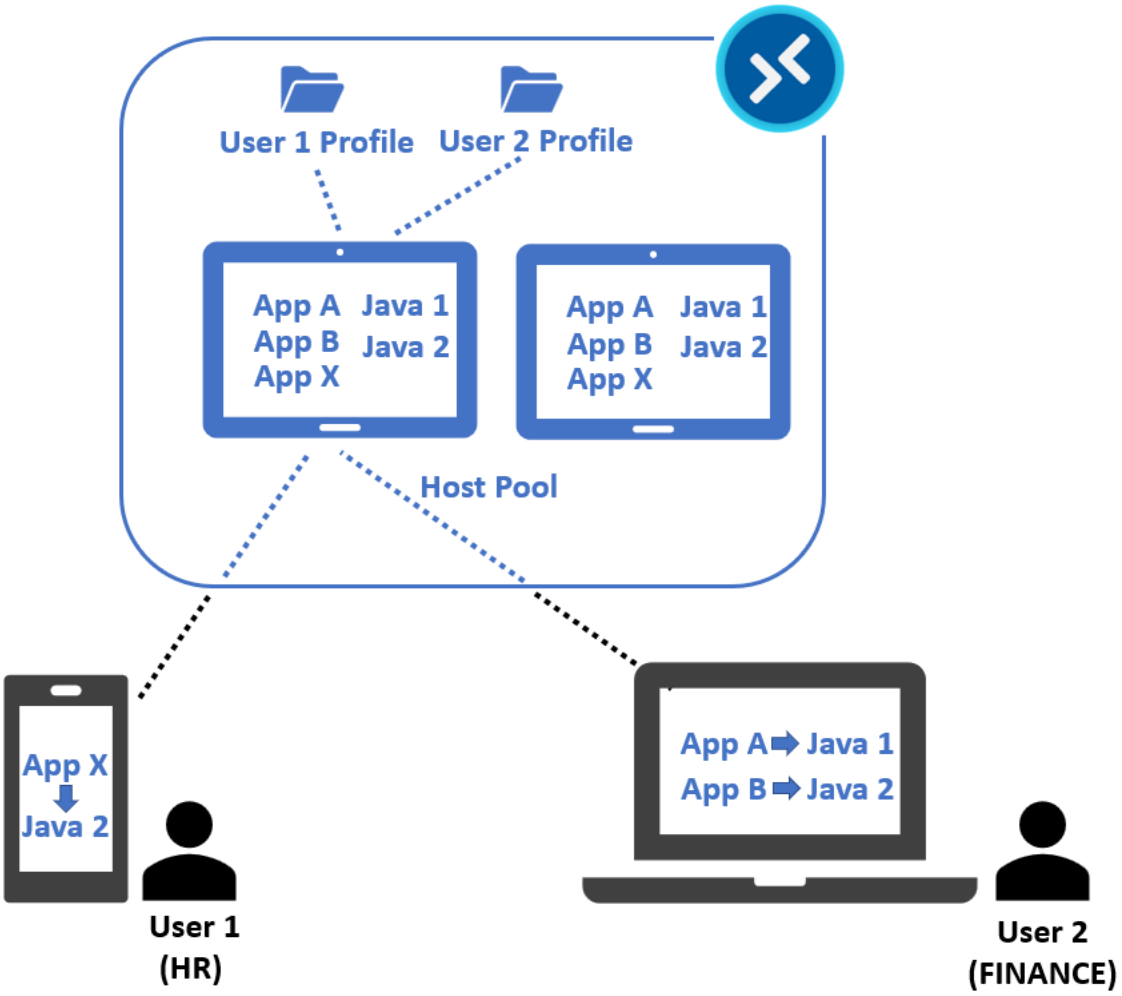


Figure 23: FSLogix Profile Container, Application Masking, and Java Redirection

You can read more about the Java redirection feature here.

FSLogix is not mandatory (I can also use other similar technologies) and can be applied to both pooled and personal virtual machines.

If I'm using it, it's critical to back up and protect the file share (the virtual machine is quite expendable now, because the data and the preferences are all inside the FSLogix VHD/VHDX). I can do that using Azure Backup or similar third-party tools.

MSIX app attach

As explained in the previous section, FSLogix can isolate, containerize, and mount the user profile into the filesystem. To extend this concept, Microsoft's Azure Virtual Desktop product team developed MSIX app attach.

MSIX is a new format that Microsoft developed to become a substitute for the classic MSI and EXE formats. With this format, you can create an application from scratch or convert an existing application using a sequencer.

Once I have an application in MSIX format, I can simply double-click it and install it as a normal, classic application. The distribution of applications in MSIX format can be done with tools like Microsoft Intune.

The app attach technology created by the Azure Virtual Desktop team can leverage an MSIX format and attach in read-only format an application that is maintained by a file share in all the virtual machines inside a host pool.

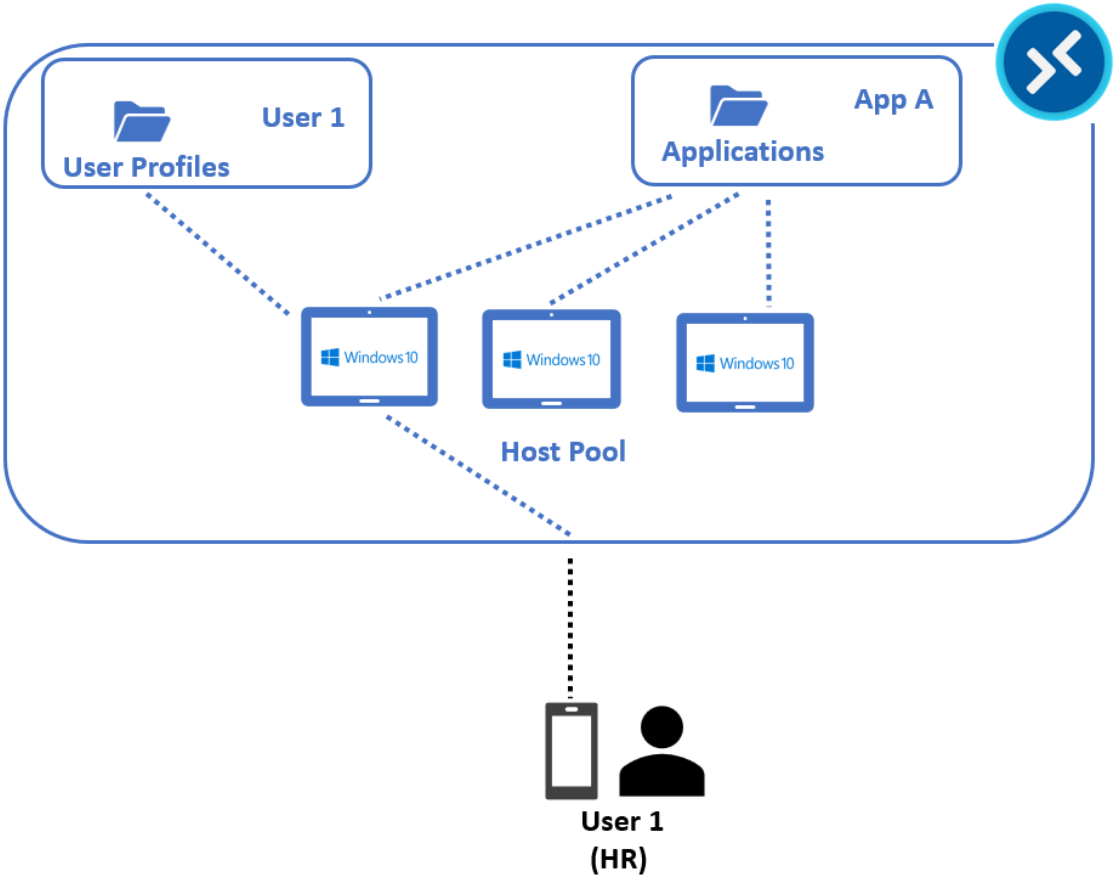


Figure 24: MSIX App Attach

Let's look at an example:

- User 1 has an application published using Azure Virtual Desktop.
- IT created a host pool composed of three virtual machines with only Windows 10 installed (no application).
- The application in MSIX format is maintained in an Azure file share and attached in read-only format to all three virtual machines of the pool.
- The user profile is maintained in another Azure file share.

When User 1 asks for the application, the Azure Virtual Desktop control plane handles the request and assigns a virtual machine to the user.

The user profile is mounted on the assigned virtual machine, so the user can log on and see their data and use the application—even if it's not really installed on the virtual machine.

This is a big advantage from a management perspective because when it comes time to update the application, it's enough to change the MSIX file in the share, and all the virtual machines that have this application linked will be able to use the new version.

And there's more! MSIX app attach is applying a sort of application masking, so if I have two applications (A and B) and my User 1 is entitled to use only application A, while User 2 can only use application B, I can let them use the same virtual machines that have both applications attached, but the system will show only the one published for the user.

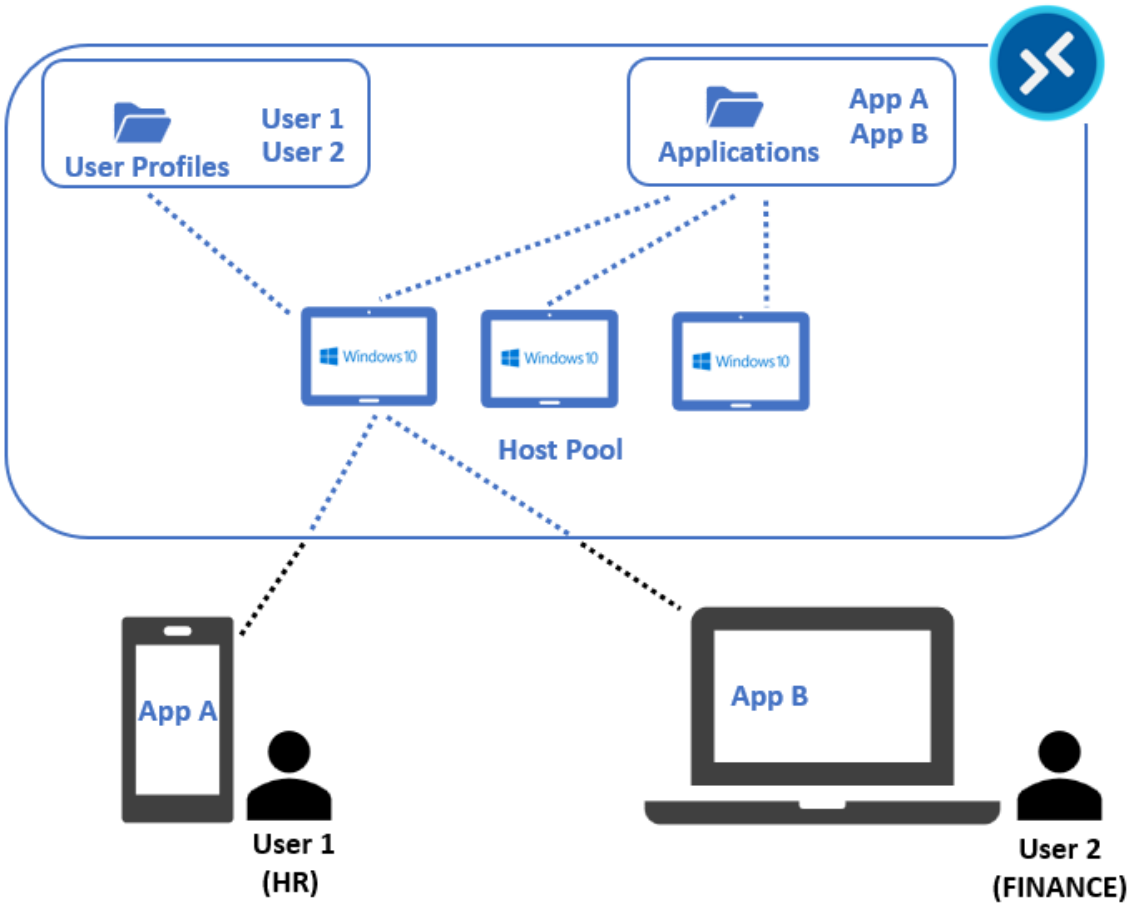


Figure 25: MSIX App Attach for Multiple Applications

Note: Currently MSIX app attach is not supported in environments using Azure Active Directory Domain Services.

You can find more information about MSIX app attach in the Microsoft documentation or in this brief, free ebook.

Disaster recovery and business continuity

Azure Virtual Desktop is a solution composed by different technologies that are creating an architecture that must be preserved from any kind of problems.

Some services, like Azure AD and Azure Active Directory Service Metadata, are naturally resilient and always available, while other services need the use of specific techniques and tools to achieve that.

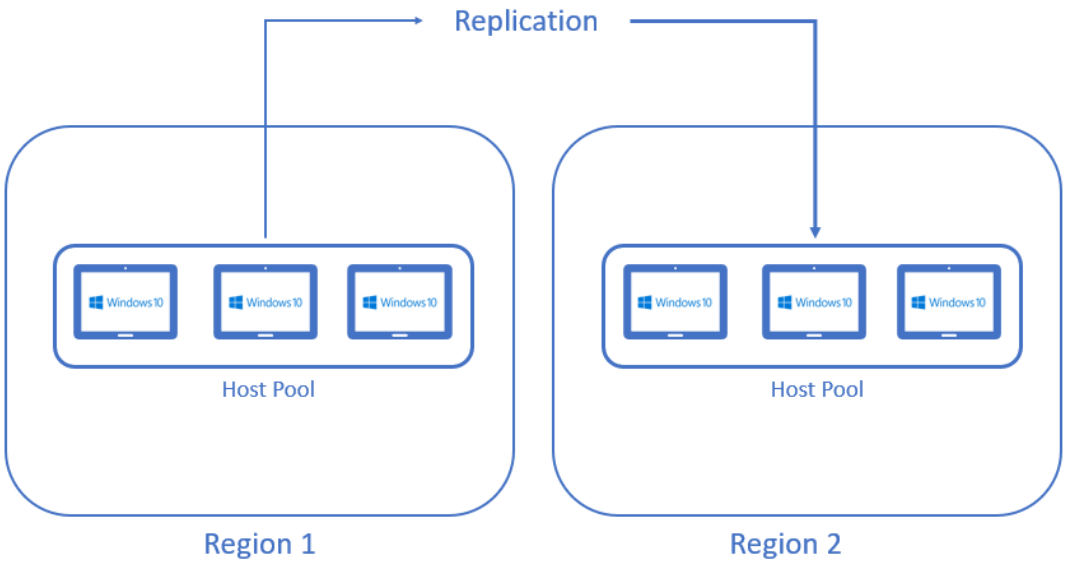


Figure 26: Disaster Recovery Object Replication

The implementation strategy depends on the architecture, because different architectures require different tools and procedures to achieve a good degree of resiliency.

For example, if I'm using pooled virtual machines created from the same base image and FSLogix, the virtual machines are highly expendable because everything that the user needs to have in order to be productive is inside the FSLogix profile. I need to take care of the backup and replication of this kind of object and preserve my base image, so that in case of a disaster, I am able to re-create the pool in few minutes starting from the base image.

But maybe recreating the pool in a few minutes is not enough, and I need to have a near-zero downtime strategy that can be implemented if I'm replicating the virtual machines in a second region, or if I'm pre-creating a secondary pool of virtual machines in the secondary region ready to serve the user.

The following table shows some key topics that must be included in a disaster recovery and business continuity plan.

Table 2: Disaster Recovery and Business Continuity

Service	Action
Service metadata	Backed up by Microsoft; no action needed.
Pooled virtual machines	Replicated using a tool like Azure Site Recovery or re-created in a different region as needed.
Personal virtual machines	Replicated using a tool like Azure Site Recovery and/or backed up using Azure Backup or third-party tools.
FSLogix profiles	Replicated using FSLogix Cloud Cache and/or backed up using Azure Backup or third-party tools.
MSIX applications	Replicated using storage replication features like Azure file geo-redundant storage (GRS).
Active Directory	Have a domain controller in the secondary location, or a connection to an on-premises domain controller.
Base image	Back up and replicate the image in a secondary location

A good source of information about this topic is this brief, free ebook.

Microsoft Teams optimization

It's possible to optimize the audio and video communications produced by Microsoft Teams inside an Azure Virtual Desktop virtual machine.

This can help you avoid voice echo, video lag, and other types of issues due to the latency between the physical device that is receiving the audio/video stream, the Azure Cloud, and the physical device of destination.

This is done using WebRTC, a standard that can redirect the audio/video stream from a physical device to another without crossing the cloud.

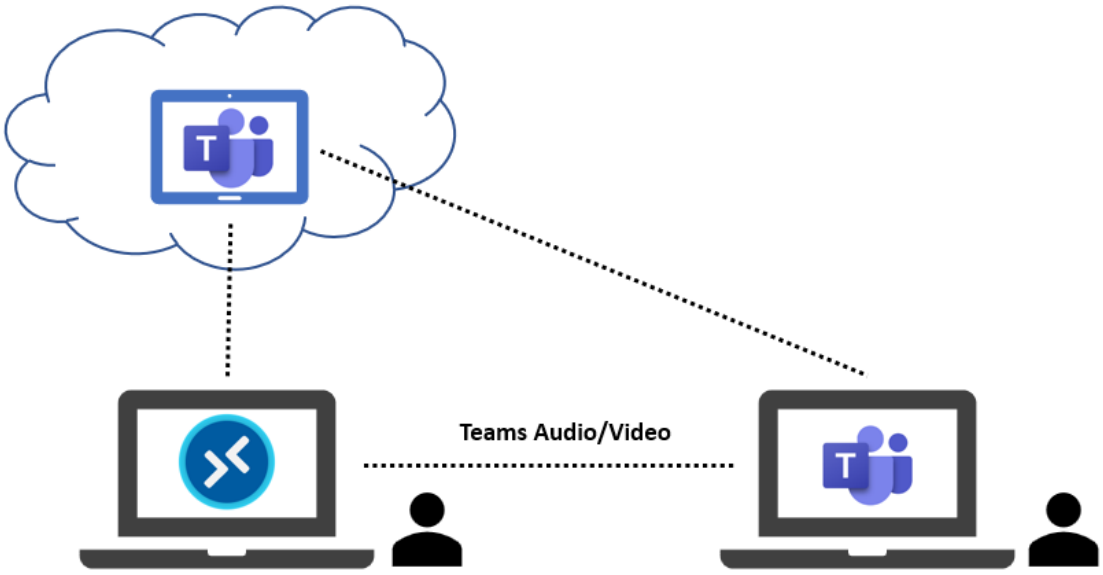


Figure 27: Microsoft Teams Audio/Video Optimization

One interesting thing to know is that all I need to do is install some pre-requirements into the Azure Virtual Desktop virtual machine; I don't need to install anything on my physical device.

This type of optimization is currently available only using the Windows client in the MSI format (not the Store app), but other platforms are in the product road map, so check the latest news regarding this topic.

If you're using this optimization, you'll currently have to deal with a number of limitations that are described here. You can find more general information about using Microsoft Teams on Azure Virtual Desktop here.

How to print

Printing from a VDI is a common topic, and to allow this, Azure Virtual Desktop can redirect a local printer. But what if my printer is not under my desk? What if I need to print everywhere from any device, on any network printer inside my organization?

I can contact a printer server that is hosted in Azure in a virtual machine or on-premises, but today we also have a more modern and "as a service" solution. The one developed directly by Microsoft is called Microsoft Universal Print, and it's a cloud service that can be used to manage a printer infrastructure.

It's based on the use of a universal print driver and the ability to register every printer into the service that is globally accessible.

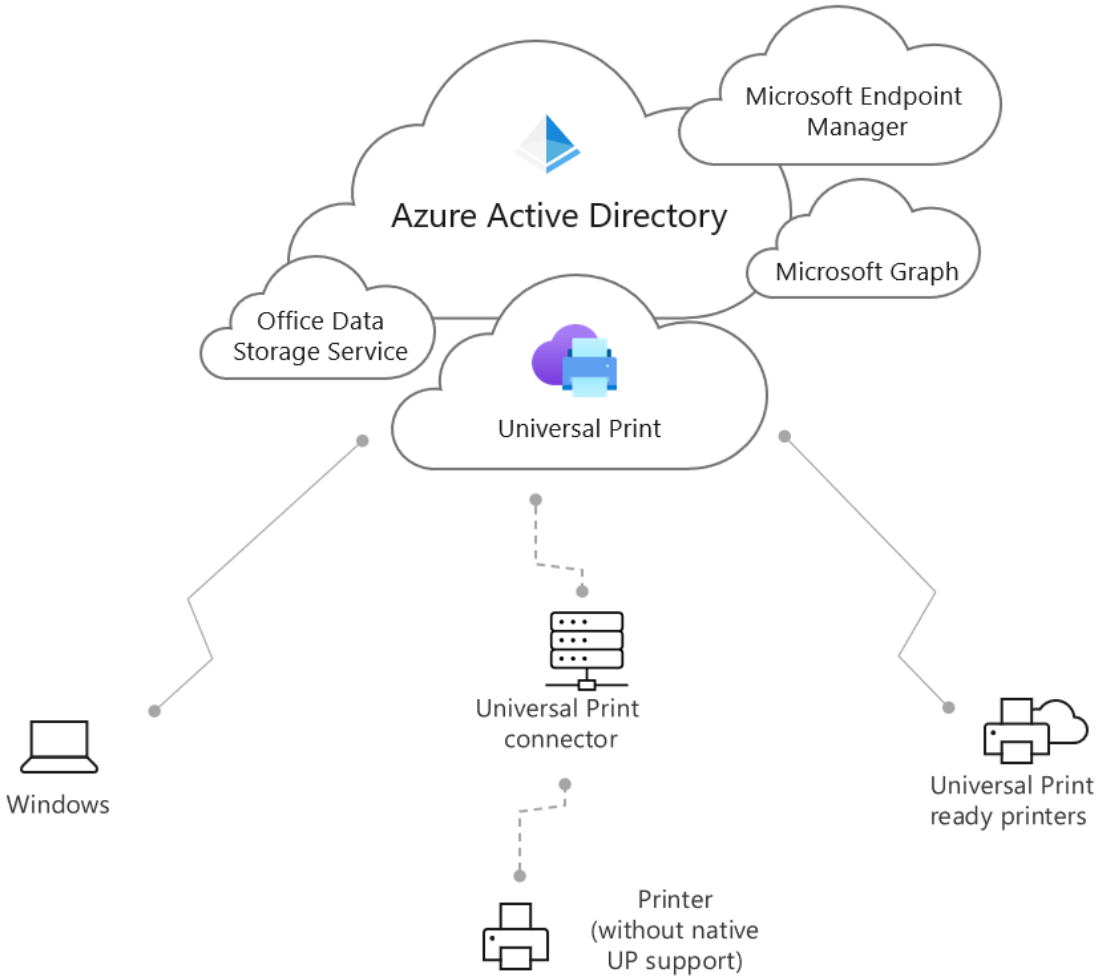


Figure 28: Microsoft Universal Print

Microsoft Universal Print is included in the following licenses:

- Microsoft 365 Enterprise F3, E3, E5, A3, and A5.
- Windows 10 Enterprise E3, E5, A3, and A5.
- Microsoft 365 Business Premium.

You can find more information here.

Note: You can also check other solutions by Printix, ThinPrint, PrinterLogic, and other vendors that are compatible with Azure Virtual Desktop.

Note: At the time of writing, Universal Print does not support Windows 10 multi-session. The support is in the product road map, so check for updates.

Service monitoring

Azure Virtual Desktop includes a monitor capability that allows you to retrieve logs and metrics from the control plane, as well as from the host pool virtual machines.

This is very important for the visibility of:

- Platform errors
- User sessions
- Virtual machine performance

This feature uses Azure Log Analytics to collect and maintain the metrics and Azure Monitor for the presentation layer. Everything is in near-real time, with the ability to access historical data and create useful dashboards.

Inside the Azure Virtual Desktop portal, the **Insights** section will guide you to activate this feature and use the standard dashboards.

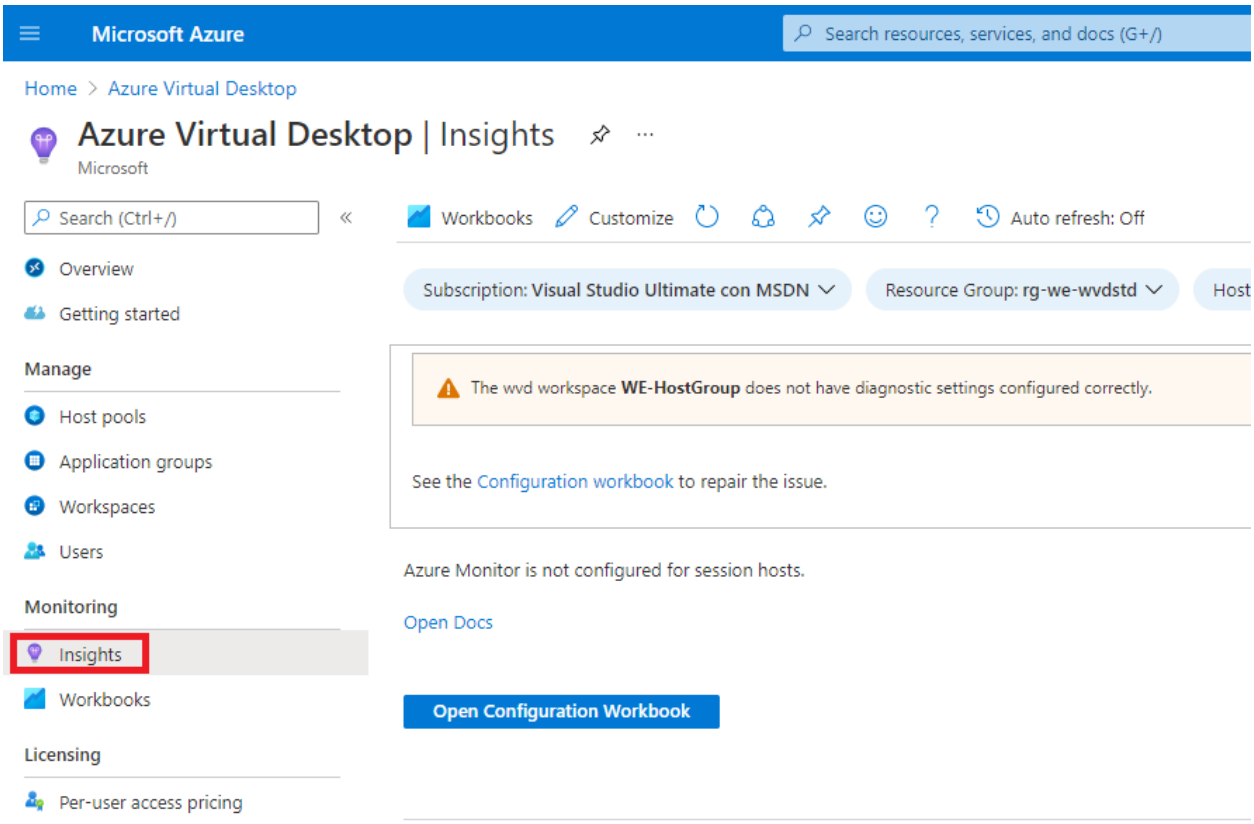


Figure 29: Azure Virtual Desktop Insights

The result is complete visibility across connections, host diagnostics, performance, user statistics, and alerts related to the service.

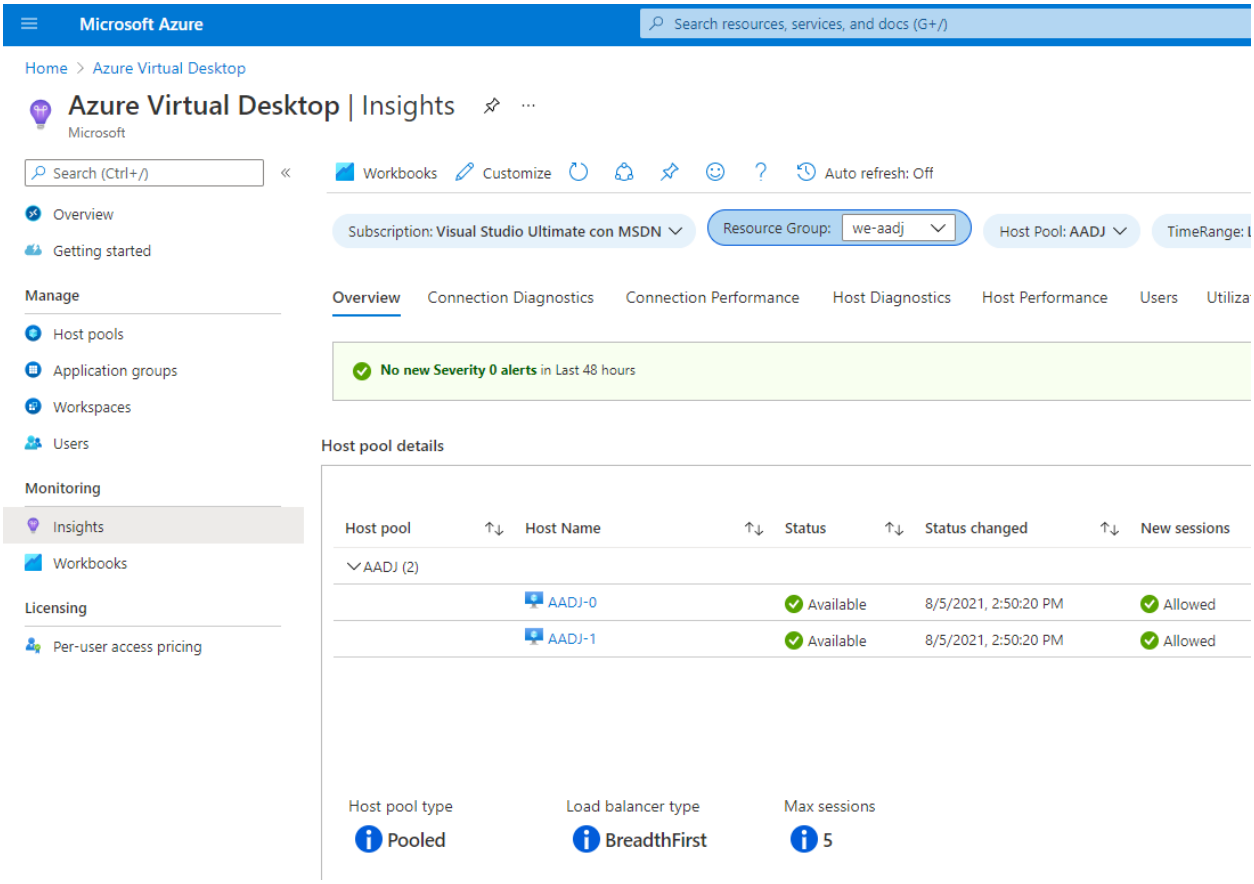


Figure 30: Azure Monitoring Configured

You can find more information about how to monitor an Azure Virtual Desktop architecture here.

GPU workloads

Azure offers virtual machines with GPUs made by Nvidia or AMD.

Virtual machines with one or more GPUs can be used for graphics-intense applications, and for deep neural systems such as PyTorch machine learning models. The family of Azure virtual machines that have GPUs is the N series.

The most used in Azure Virtual Desktop are the ones under the NV series, NVv3 series (Nvidia), and NVv4 series (AMD). The NVv4 series is the only one that currently supports GPU partitioning, meaning I can set up a virtual machine that takes advantage of a fraction of the graphic power of a GPU.

Table 3: NVv4 Available Sizes

Size	vCPU	Memory	GPU Power	GPU Memory
Standard_NV4as_V4	4	14 GB	1/8	2 GB
Standard_NV8as_V4	8	28 GB	1/4	4 GB
Standard_NV16as_V4	16	56 GB	1/2	8 GB
Standard_NV32as_V4	32	112 GB	1	16 GB

This is a good way to provide a little bit of graphic power without raising the overall cost of the virtual machine. Regarding this topic, you can find a good article here.

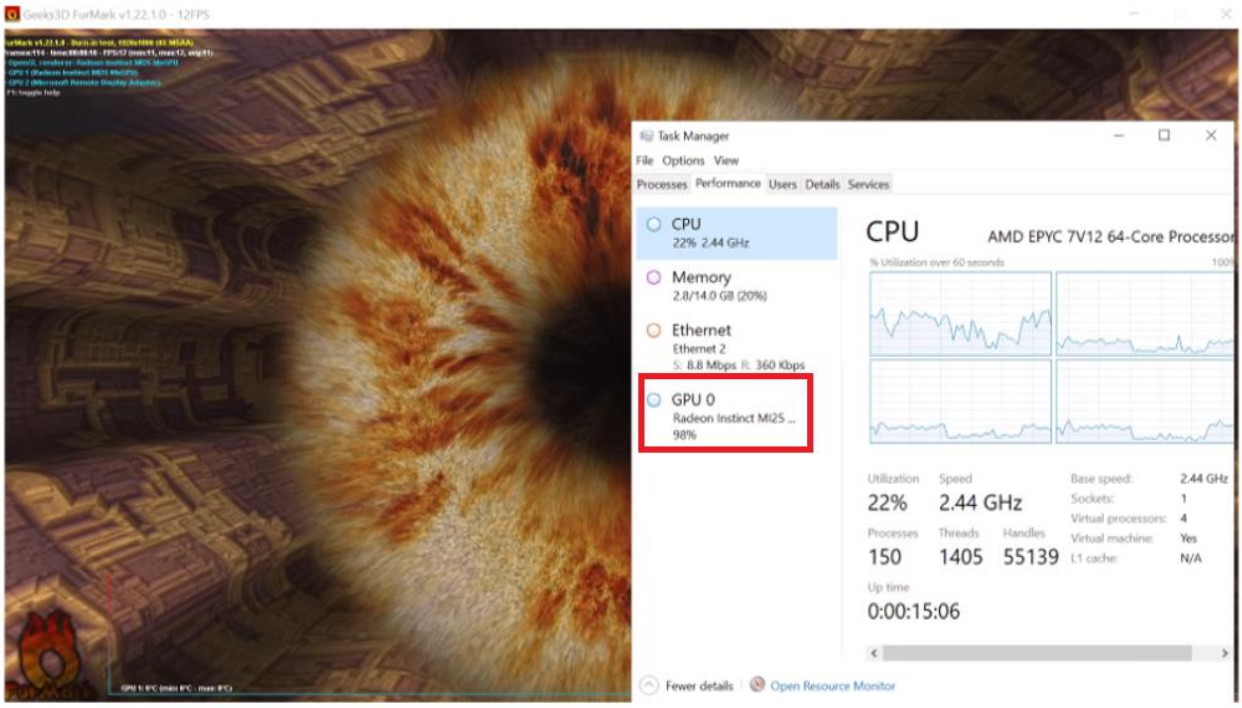


Figure 31: Azure Virtual Desktop Machine with GPU

When I create a machine with GPU power inside my host pool, whether it's a personal host or a pooled host, I need to perform several configurations to unlock the graphic power:

- Install the correct drivers (it's possible to use an Azure extension to do this automatically).
- Configure GPU-accelerated frame encoding (Nvidia only). This can be done using an Active Directory group policy.
- Configure full-screen video encoding. This can be done using an Active Directory group policy.

The official documentation reports all the needed configurations.

Image Builder and Shared Image Gallery

When I create a virtual machine inside Azure Virtual Desktop, I can choose from several images that Microsoft has created, and that are always up to date with the latest patches.

These images include the FSLogix agent, and optionally, the Microsoft 365 Apps.

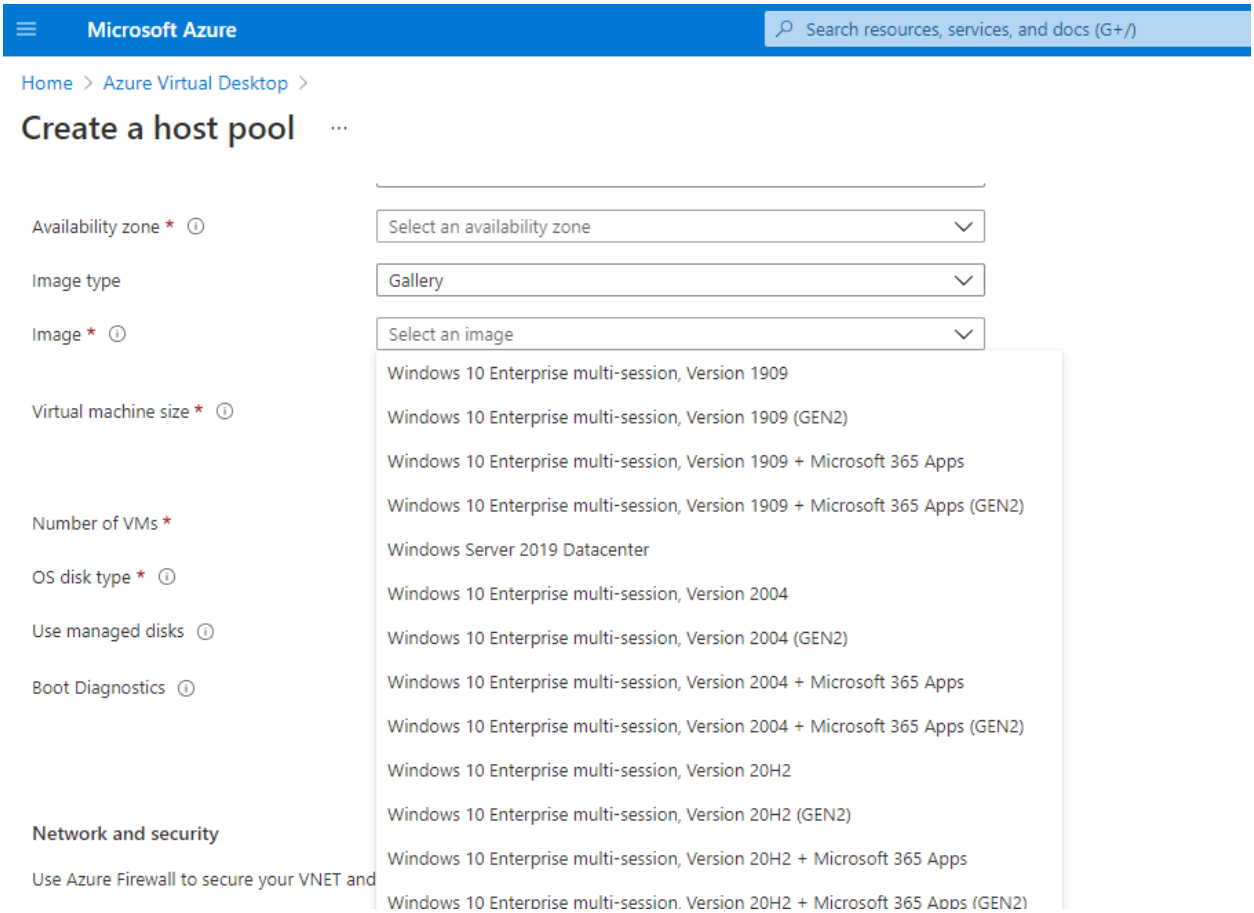


Figure 32: Image Gallery

This is perfect if I don't have other applications, if I can use MSIX app attach, or if I want to install them using Intune or other management tools.

But what if I would like to create my own base image? I have several ways to do that.

The first one is the classic way:

1. Create a virtual machine in Azure or on-premises.
2. Use a manual or automatic way to customize it.
3. Generalize the operating system using Sysprep.
4. Upload the generalized image in a blob storage.

I can use classic tools like Microsoft Deployment Toolkit (MDT) or System Center Configuration Manager (ConfigMgr or SCCM) to customize and generalize the image.

The second, new way I can do it is using Azure Image Builder, an Azure service that utilizes HashiCorp Packer to automate the creation of a Windows (or Linux) image that can be used to create generic Azure virtual machines. (It's not something dedicated exclusively to Azure Virtual Desktop.)

The following image shows the high-level process.

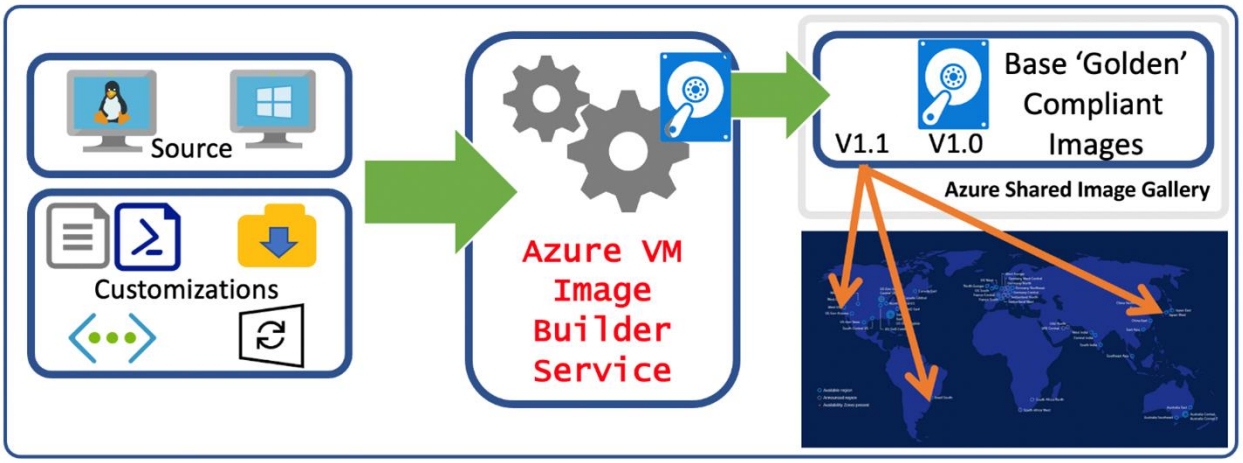


Figure 33: Azure Image Builder (Source)

Currently, the Azure Image Builder service is completely code based, and you can find a number of examples (like this one) about how to create a base image using this service and JSON files.

Once I have created my custom image with a classic tool or with Azure Image Builder, I can upload it inside Azure Blob Storage and use it for creating my Azure Virtual Desktop virtual machines.

But I can also use another interesting service, called Azure Image Gallery, which is basically a container of Windows/Linux images that provides several advantages:

- Global replication of images.
- Versioning and grouping of images for easier management.
- Highly available images with zone-redundant storage (ZRS) accounts in regions that support availability zones. ZRS offers better resilience against zonal failures.
- Premium storage support (Premium_LRS).
- Sharing across subscriptions, and even between Active Directory (AD) tenants, using Azure RBAC.
- Scaling your deployments with image replicas in each region.

The global replication and sharing across subscriptions capabilities are crucial in large deployments where a multinational organization is willing to adopt Azure Virtual Desktop in different countries.

When a new version of my base image is created and placed inside Azure Shared Image Gallery, a replication will start for different Azure regions. This provides consistency across different host pools in different subsidiaries.

The versioning and grouping capabilities help me maintain control over the image versions, so I can always use the most up-to-date image while also being able to use previous versions if I need to.

The following figure shows how the service can maintain different versions of multiple image definitions:

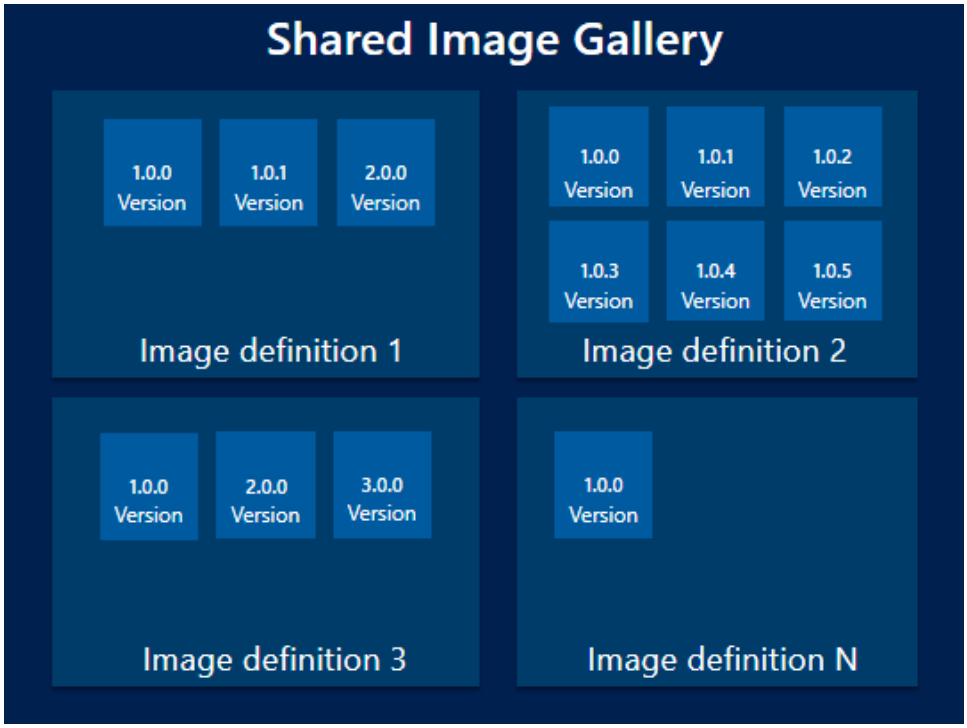


Figure 34: Azure Shared Image Gallery Versioning and Grouping (Source)

In the future, Azure Image Builder will be more integrated into Azure Shared Image Gallery, as shown by the product group in this video.

Note: It's important to NOT include the Azure Virtual Desktop agent inside the generic image because this can cause deployment problems. The agent is automatically installed into the virtual machines by the host pool creation procedure, or it can be manually installed once the virtual machine is deployed.

User support

How can the IT department help a user that is experiencing a problem inside their Azure Virtual Desktop session? Using Microsoft technology, it's possible to shadow a user session in three different ways, as explained in this article.

The three methods are:

- Quick Assist
- Remote Assistance (MSRA)
- Remote Desktop Connection (MSTSC)

Because several Azure Virtual Desktop architectures use a multi-session operating system like Windows Server or Windows 10 multi-session, it's important to know which session host the user is connected to, and the user session number.

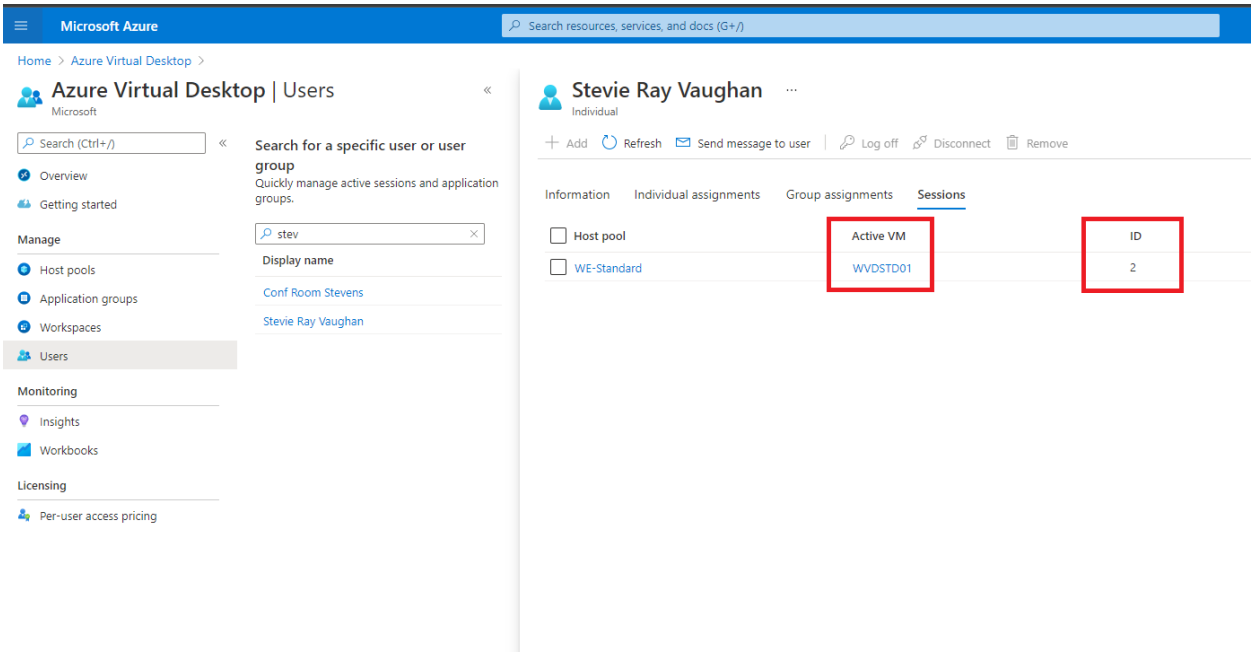


Figure 35: User Session Details

If you are using third-party tools, check if they are compatible with Windows 10 multi-session because it's still quite new, and not all the remote management tools are supporting it.

RDP Shortpath

The communications between the physical devices and the Azure Virtual Desktop virtual machines are using the RDP protocol with TCP. The communication is established between the physical device and the Azure Virtual Desktop control plane, and from the control plane (RD Gateway) to the virtual machine.

Using RDP Shortpath (still in preview at the time of writing), it's possible to establish a direct connection between the physical device and the Azure Virtual Desktop virtual machine using UDP and bypassing the gateway. This is a more direct path that offers a better degree of performance because it's offering a better latency.

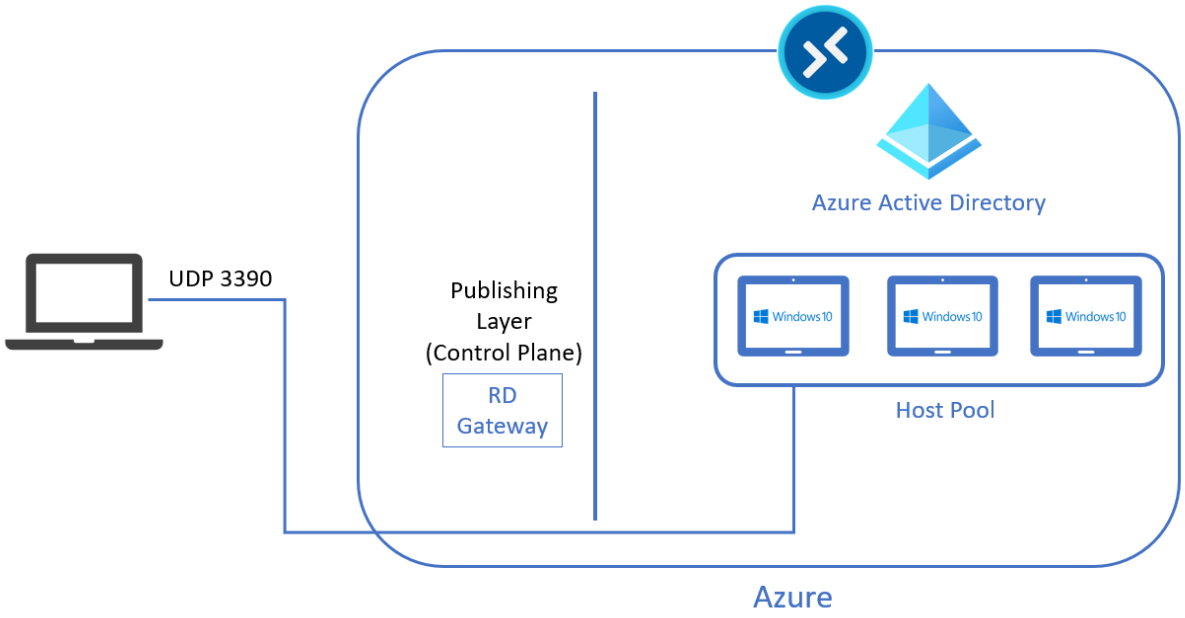


Figure 36: RDP Shortpath

To establish this kind of direct connection, the physical device must be able to “see” the Azure Virtual Desktop virtual machines.

This can be done using any of the following:

- ExpressRoute private peering.
- Site-to-site VPN (IPsec based).
- Point-to-site VPN (IPsec based).
- Public IP address assignment.

Note: As a security best practice, it’s highly discouraged to assign a public IP to an Azure Virtual Desktop virtual machine. It’s something that is supported, but it’s not recommended.

Start VM on connect

The Start VM on connect feature allows users to request an Azure Virtual Desktop resource even if all the virtual machines inside a pool are turned off. This technique helps to optimize the costs because it’s possible to have the virtual machines running only when the user needs them.

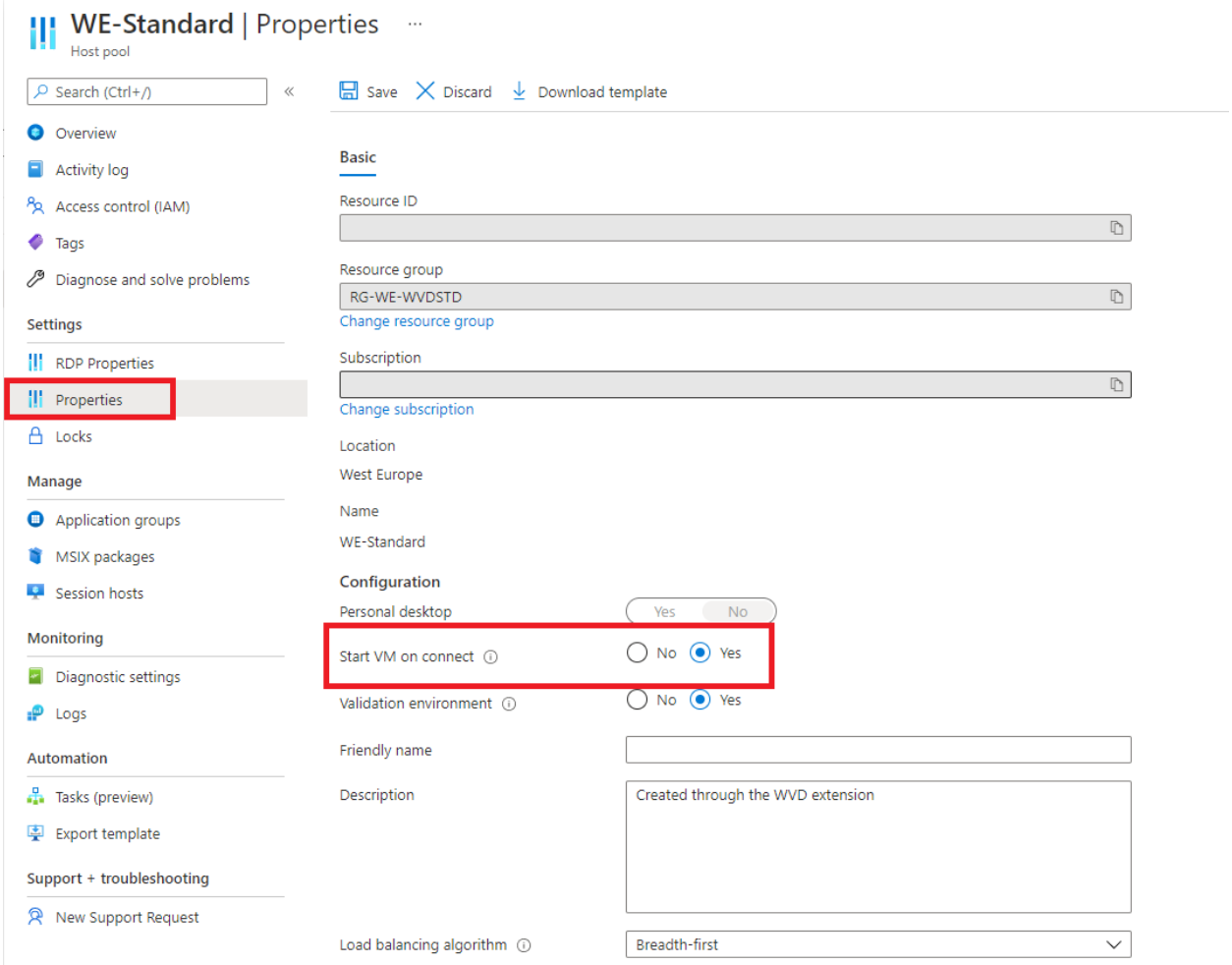


Figure 37: Start VM On Connect

The feature must be enabled from the host pool properties, and it affects all the virtual machines inside the pool. It can be enabled for both personal and pooled virtual machines.

In a personal desktop environment, the experience is that a user is requesting the Azure Virtual Desktop resource using the web browser or the client application, and because their personal virtual machine is not running, they need to wait some minutes to let the system start the virtual machine.

In a pooled desktop environment, the first user that is requesting the service needs to wait until the virtual machine is available, but if the virtual machine can host, for example, five total users, the next four users are connected without waiting for the service because the virtual machine is already running and started by the first user.

You can find more information about this feature here.

Chapter 3 Security

Reverse connection

One of the most interesting features of Azure Virtual Desktop is that it is accessible from any place through the internet channel.

The end user experience is that a connection from their physical machine to the assigned virtual machine in Azure is created using the RDP protocol.

But is it secure? The answer is yes, because the Azure Virtual Desktop virtual machines are not directly exposed to the public internet.

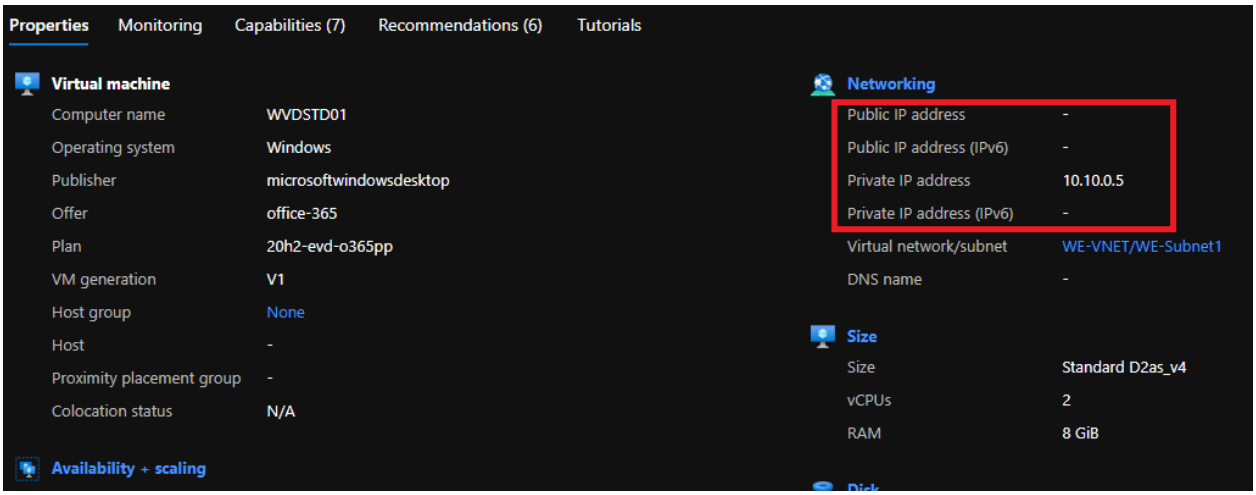


Figure 38: Azure Virtual Desktop Virtual Machine Networking

As you can see, a typical Azure Virtual Desktop virtual machine has an internal IP address, but no external IP address assigned. So how is it possible that a connection is established between my physical machine and my Azure Virtual Desktop virtual machine?

The answer is that Azure Virtual Desktop is using a technology called reverse connection that is implemented thanks to a role called RD Gateway inside the Azure Virtual Desktop control plane.

My physical device and my Azure Virtual Desktop virtual machine are not directly connected. Everything is orchestrated by the RD Gateway role that is a sort of a man-in-the-middle between the two devices (the physical one and the virtual one).

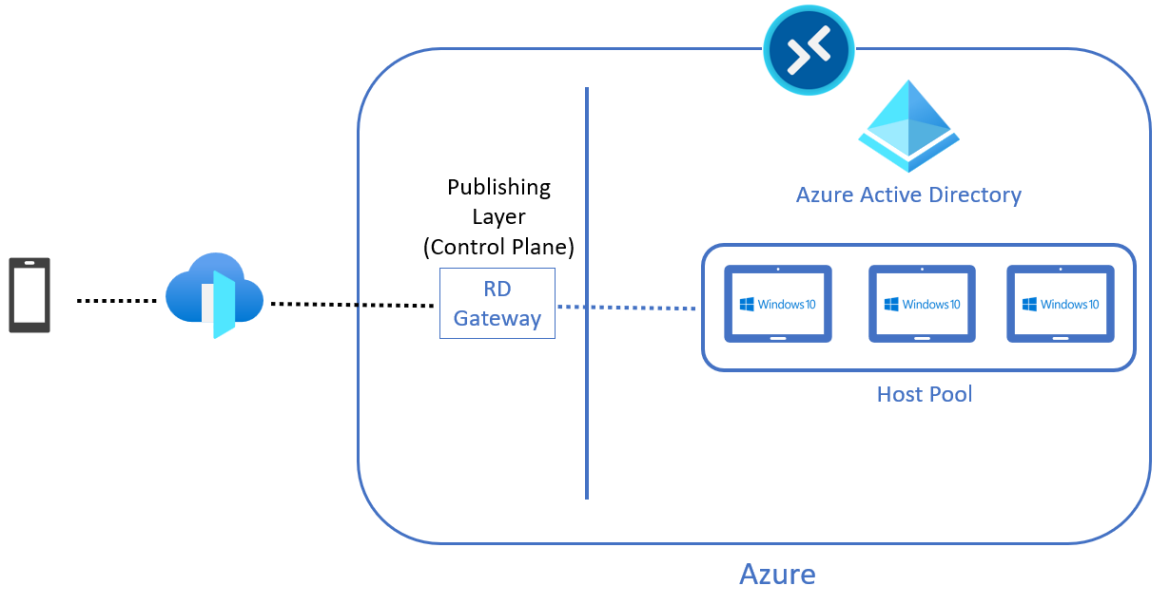


Figure 39: Azure Virtual Desktop Reverse Connection

The main advantage is that no inbound firewall ports need to be opened. The reverse connect technology creates an outbound connection using the TCP protocol on the 443 port.

Note: Sometimes the figures in this book are intentionally minimal because the goal is to introduce and clarify a concept in an easy way. The connection flow can become quite complicated, so for more information about this topic, please read this official article.

Multifactor and conditional access

Multifactor authentication is a widely used technique that can stop most identity attacks.

In Azure Virtual Desktop, this is based on asking the user to provide a second factor of authentication that can be:

- Authenticator apps
- Token hardware OATH
- SMS
- Voice call

I can set up a policy that asks me to provide the second factor when I'm connecting to the Azure Virtual Desktop service.

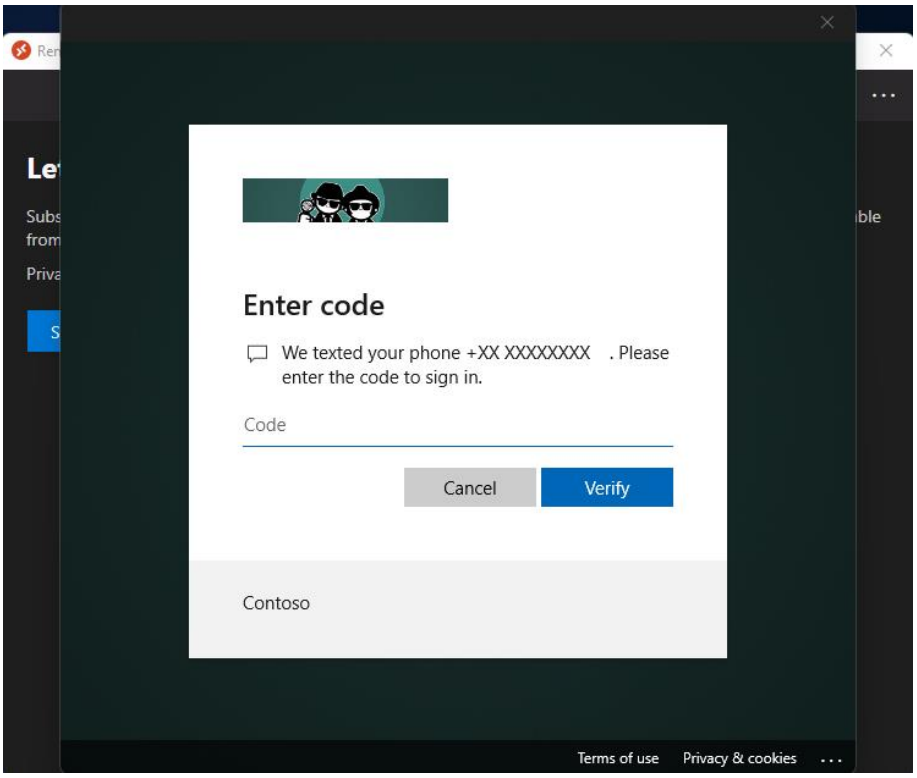


Figure 40: Multifactor Requirement in Azure Virtual Desktop

This policy can be easily created using Azure Active Directory and Conditional Access, which is another very helpful technology that allows us to analyze signals coming from:

- User risks.
- Sign-in risks.
- The device platform used.
- Locations.
- Groups of devices.

The Conditional Access policy can block or allow access to selected applications based on conditions. This allows us to create interesting policies that allow access to Azure Virtual Desktop, but only if the user provides multifactor authentication, and block the access if certain risky conditions are true, like a sign in from an untrusted location.

You can find more info about how to enable multifactor authentication inside a Conditional Access policy here.

Service tag

Sometimes the security requirements are seeking to regulate the communications between the Azure Virtual Desktop virtual machines and the service. This is why Microsoft published this article with the list of URLs that need to be accessible in order to let the service work properly.

Note: Microsoft provides only URLs, and not the underlying IP addresses, because the IPs change continuously.

Microsoft also provides an easier way to achieve the same goal of allowing only the connections with the needed services, and this is done using the virtual network service tags.

The tag groups all the needed IP addresses for a specific service. The one for Azure Virtual Desktop is currently showing the old name, “WindowsVirtualDesktop.”

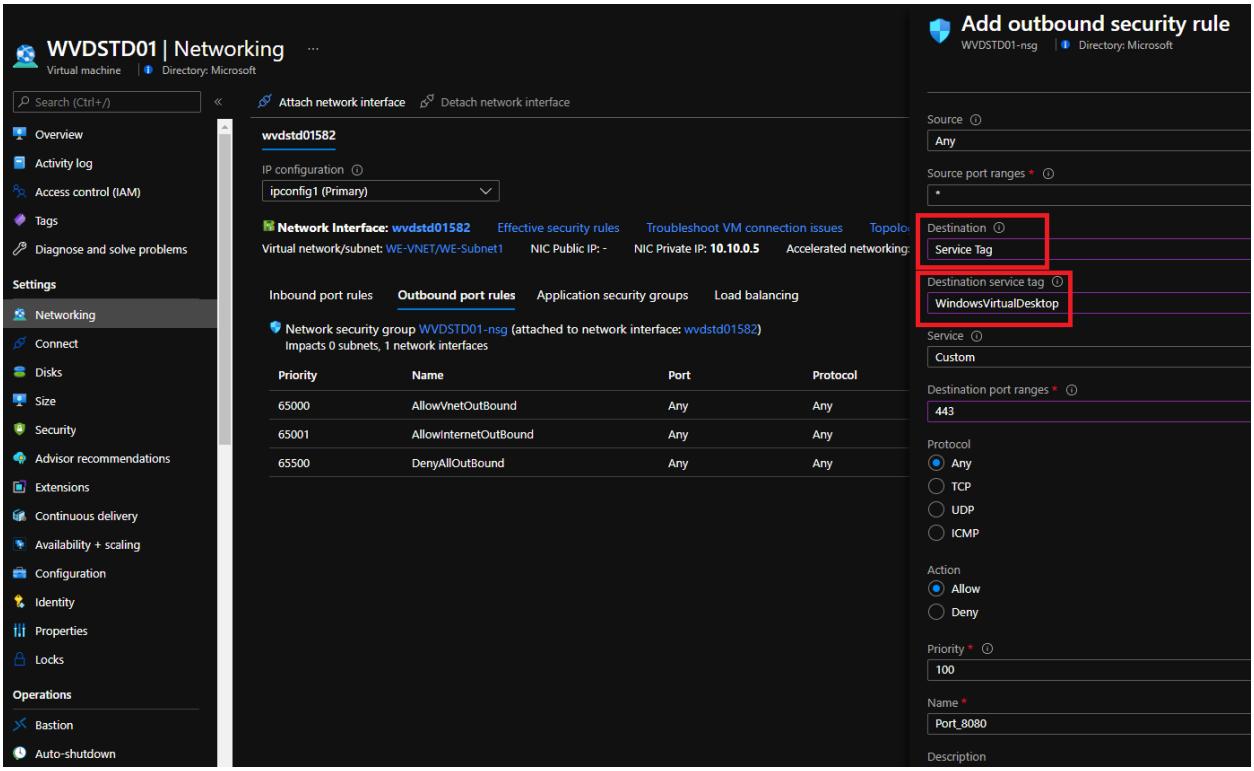


Figure 41: Azure Virtual Desktop Service Tag

This kind of tag can be used inside the network security groups or inside Azure Firewall, as shown in this article.

Web category control

When I’m connected to an Azure Virtual Desktop virtual machine, I can browse the internet without any limitation, taking advantage of the very good connection that Azure can provide me.

What if I would like to prevent my users from connecting to a specific site? I can do that using network security groups, or a firewall (like Azure Firewall) that can block specific URLs or specific IP addresses.

But that means I need to know all the URLs and all the IPs that I want to block, so this is not possible. This is the reason why several companies are offering web content-filtering systems based on categories like adult content, social media, and so on.

So, I can select a third-party service that provides this useful capability, or I can use a couple of Microsoft products that can help me achieve this goal.

The first one is Azure Firewall Premium, which is the most advanced version of our Azure Firewall with many capabilities:

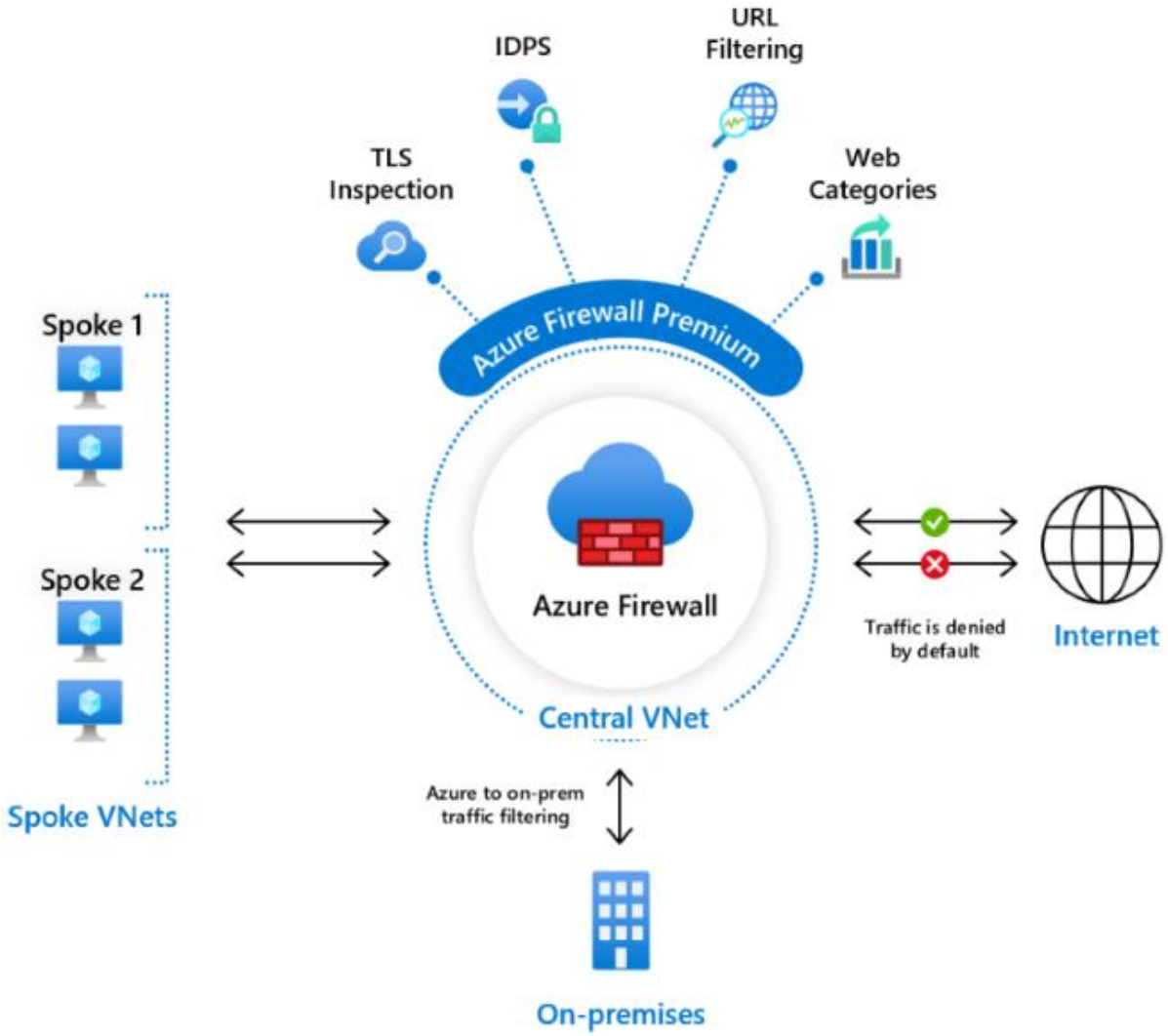


Figure 42: Azure Firewall Premium (Source)

Azure Firewall Premium offers several advanced security features like intrusion detection and prevention, URL filtering, TLS inspection, and web category! I've discussed this specific feature in this article.

The other way to block a specific category of sites is using Microsoft Defender for Endpoint.

This cloud service was launched years ago with the name of Windows Defender ATP, and now provides protection to all device types (Linux, iOS, Android, and macOS), so the name has changed to the more generic Microsoft Defender for Endpoint.

The service is an EDR (endpoint detection and response) system, a quite new category of tools that is expanding the protection of endpoints typically based on antivirus that are working with malware signatures.

An EDR can protect a device from any kind of attack because it's based on behavioral analysis and other technologies that can protect against zero-day attacks, in-memory attacks, and other kinds of modern threats.

But Microsoft Defender for Endpoint is not only a EDR system—inside the console, I can also set up web categories.

I can take advantage of the deep integration between Windows 10 and Microsoft Defender for Endpoint to perform the onboarding for the service simply using a scripted action (no agent installation is required) and utilize the protection capabilities, as well as the access control based on web categories.

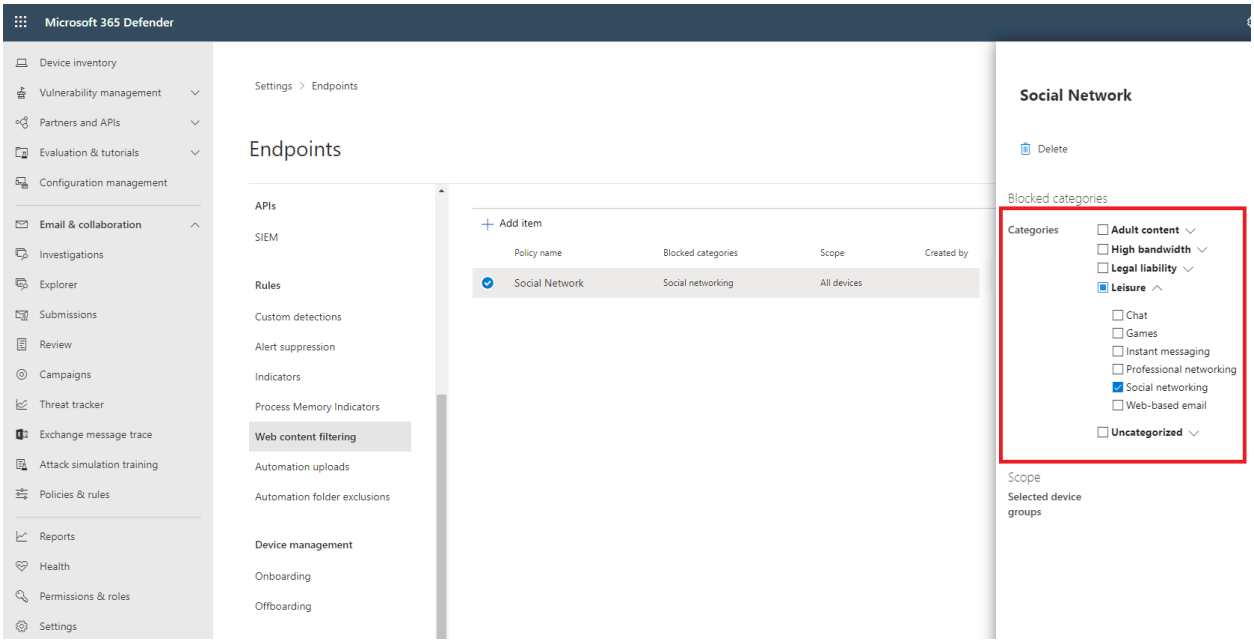


Figure 43: Microsoft Defender for Endpoint Web Categories

I've written about this specific feature in this public article.

Note: Currently the web content filtering capability of Microsoft Defender for Endpoint is available only for Windows 10.

Antivirus and anti-malware

Azure Virtual Desktop machines can use Microsoft antivirus protection as well any third-party antivirus software.

Windows Defender Antivirus is already embedded into Windows 10, Windows Server 2016, and later, while for Windows 7 and Windows 2012 it's possible to install a separate agent using System Center Configuration Manager.

The Microsoft antivirus protection can be managed using group policies, System Center Configuration Manager, or Intune.

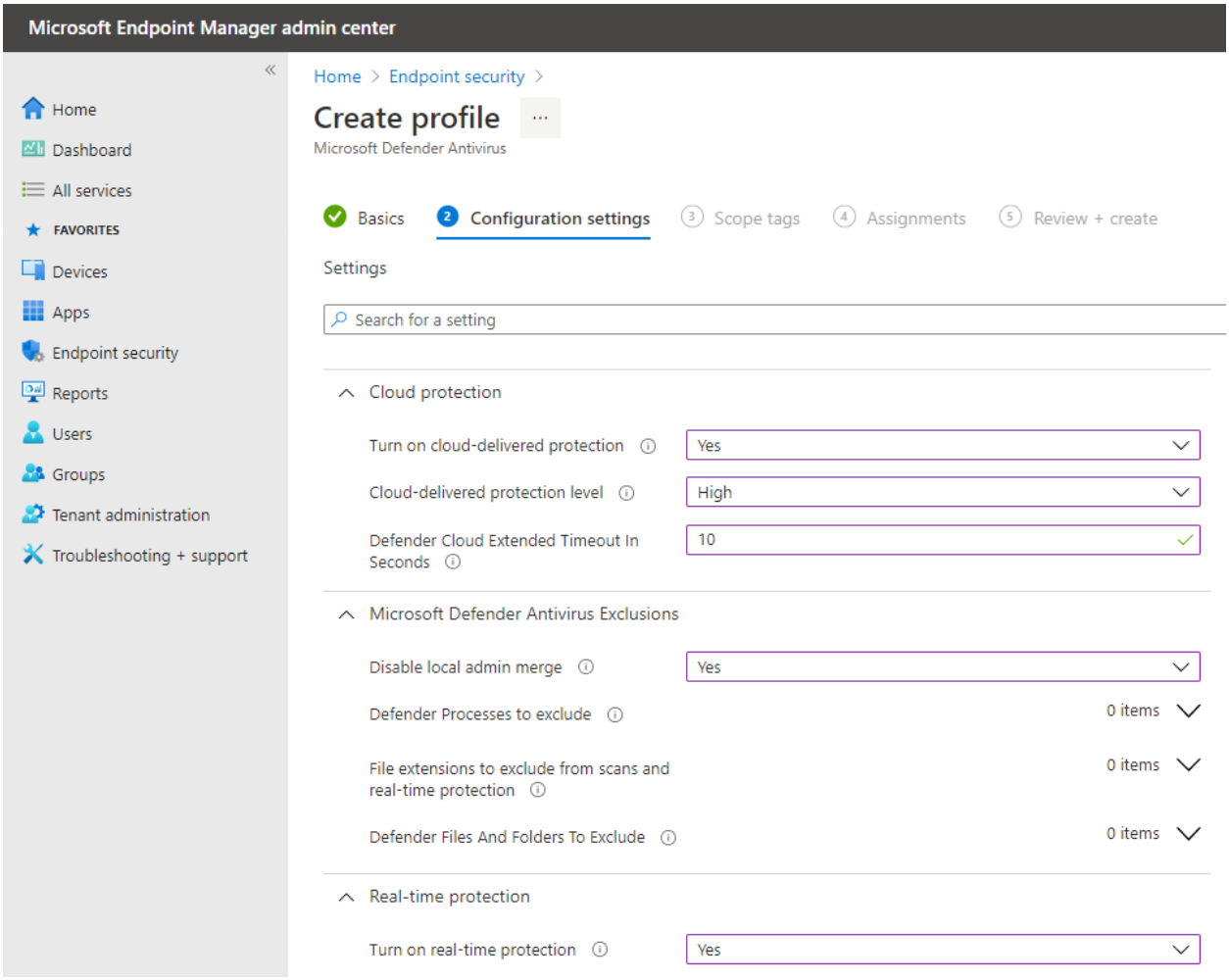


Figure 44: Defender Antivirus Policy in Microsoft Intune

No matter how the antivirus support is managed, it's interesting to read this article about how to set it up in a virtual desktop infrastructure.

Security baseline

Using Microsoft Intune and your Virtual Desktop solution based on Windows 10, it's possible to easily apply security baselines.

A security baseline is a group of settings that can help in maintaining the right security posture.

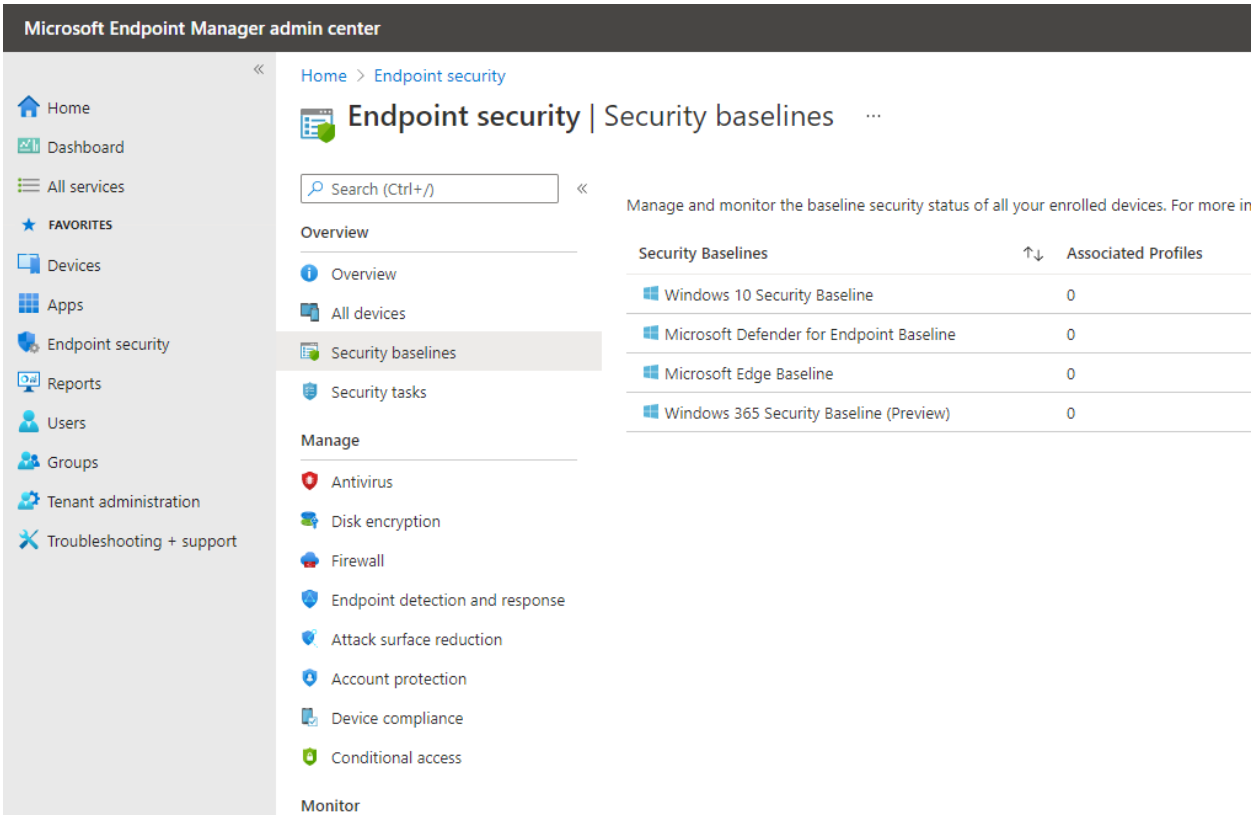


Figure 45: Microsoft Intune Security Baselines

I can apply a security baseline as is, or I can start from one, modify it, and create a custom baseline.

Currently we have the option to use a generic Windows 10 Security Baseline, a Microsoft Edge Baseline (if Microsoft Edge is your browser of choice), and the one dedicated to the new service Windows 365.

If you are managing your Windows 10 devices with Intune, this is a must-have. Read this article for more information.

Azure Security Center

Azure Security Center is an Azure Cloud service that gives me visibility into my security status, provides recommendations, and then allows me to quickly improve my security posture.

The interesting thing is that the security posture management part of this service comes without a cost, so it's highly recommended to onboard Azure resources like the Azure Virtual Desktop virtual machines into this service.

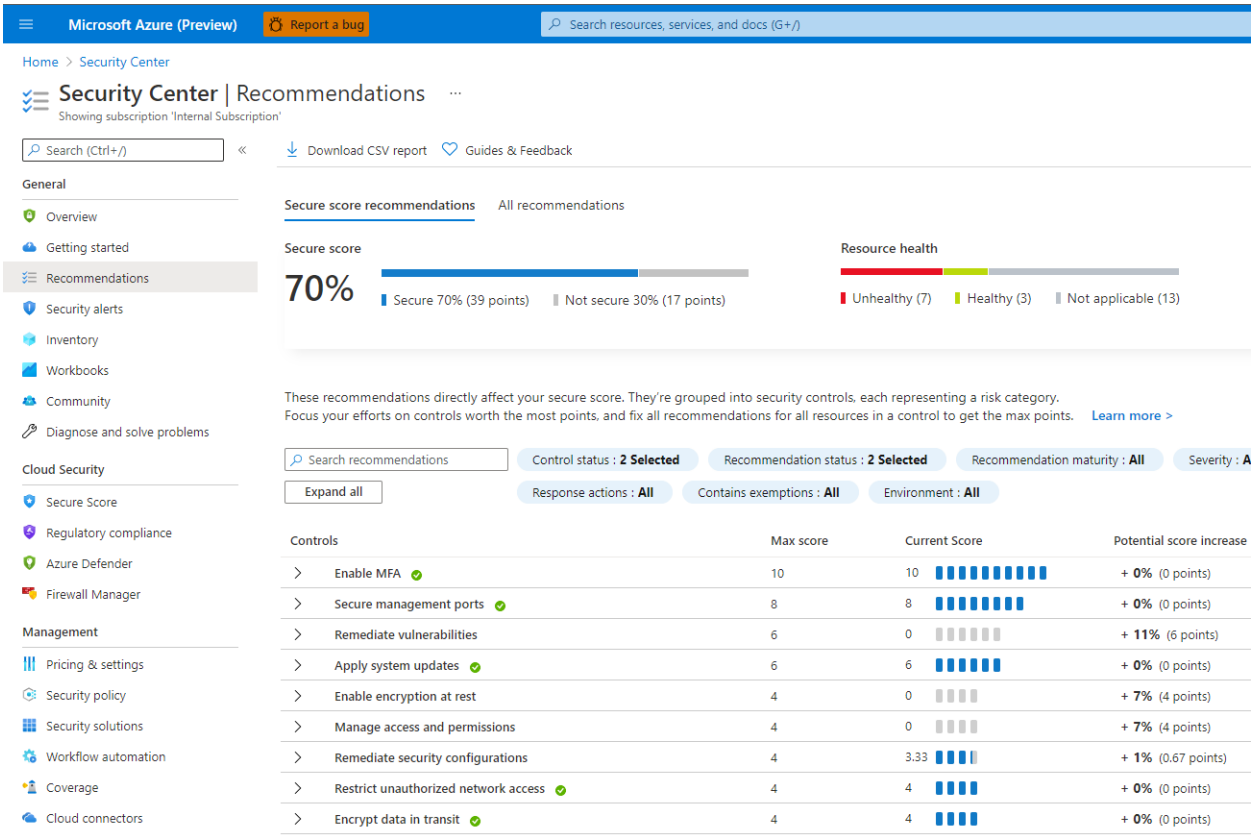


Figure 46: Azure Security Center

You can find information about this interesting security service here.

Azure Defender for Storage

As we learned in the previous chapters, Azure Files can be a good choice for hosting FSLogix profiles. Azure Files is a file share service provided and maintained by Microsoft. That means I don't need to take care about the underlying architecture, only about the content.

If the content is the user profiles, I need to protect them because the share is where the user maintains important files. I can apply backup, but I can also apply a more active protection using Azure Defender for Storage.

Azure Defender for Storage is part of the Azure Security Center but it comes at a price, unlike the Security Posture Management capability, which is free.

By activating this service, I can immediately gain several good protection engines:

- Suspicious access patterns, such as successful access from a Tor (anonymous communication router) exit node or from an IP considered suspicious by Microsoft Threat Intelligence.
- Suspicious activities, such as anomalous data extraction or unusual change of access permissions.
- Upload of malicious content, such as potential malware files (based on hash reputation analysis) or hosting of phishing content.

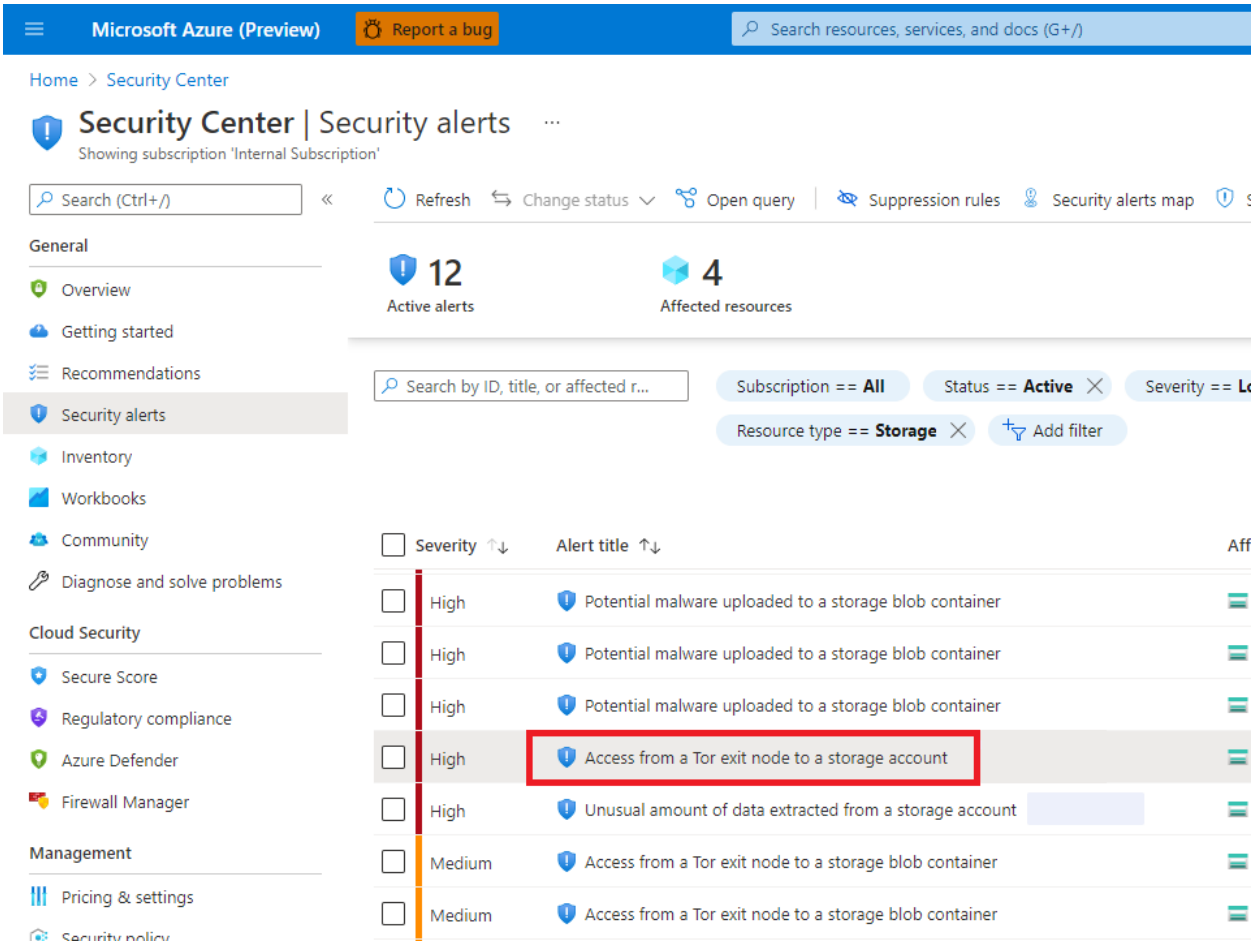


Figure 47: Azure Defender for Storage

Redirection control

Using the Azure Virtual Desktop console, it's possible to manage the redirection of physical devices like webcams, as well as the physical device clipboard.

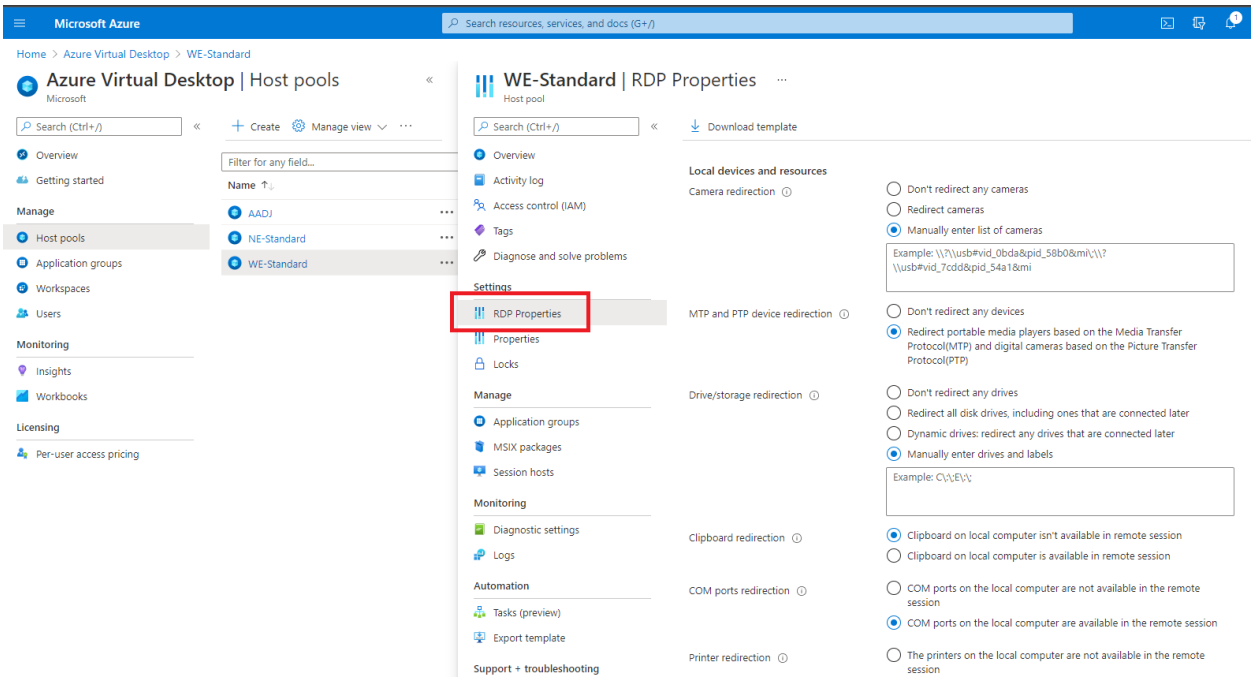


Figure 48: RDP Properties in the Azure Virtual Desktop Console

Security controls

It's a best practice to set up a couple policies:

- Establish maximum inactive time and disconnection.
- Set up screen locks for idle sessions.

These two settings can help prevent unauthorized access.

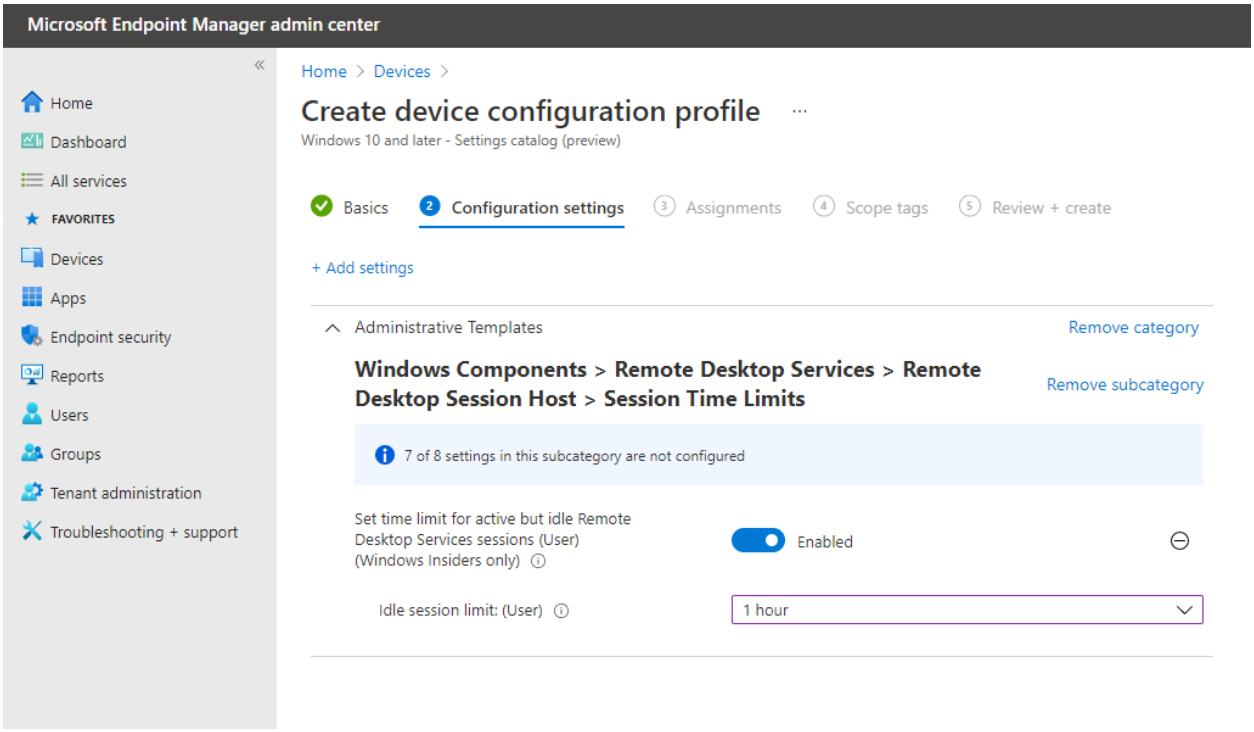


Figure 49: Security Controls Using Microsoft Intune

It's possible to set these rules using group policies (all Active Directory joined devices) or Microsoft Intune (Windows 10 only).

In general, a Windows Virtual Desktop virtual machine is a Windows machine, so I can apply many other Microsoft or third-party security tools.

On the Microsoft side, AppLocker, Application Control, Attack Surface Reduction, Secure Boot, Network Protection, Controlled Access Folder, and Credential Guard can help reduce the attack surface and maintain the best security posture.

Screen capture protection

This is a feature that prevents the leakage of sensitive information using a screen capture technique as shown in the following image:

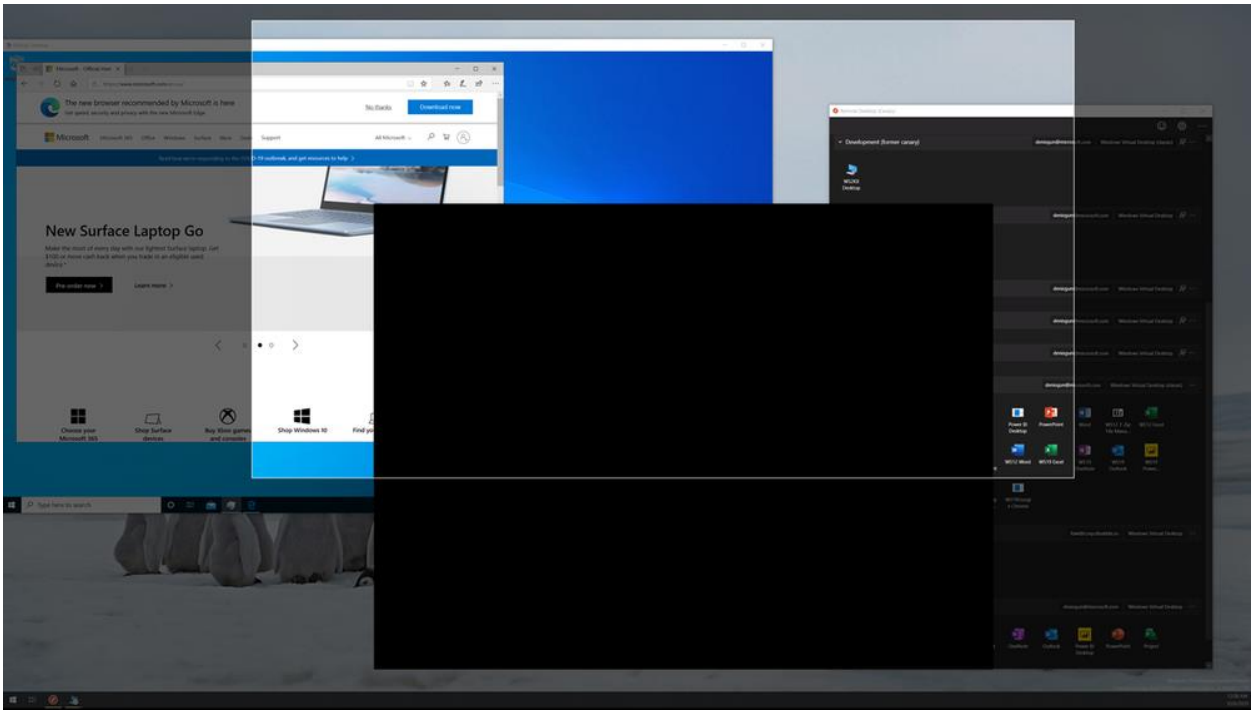


Figure 50: Screen Capture Protection (Source)

This feature can be activated at the host pool level: users that are using a virtual machine inside a host pool with the screen capture protection feature activated are not allowed to take screenshots of remote content coming from Azure Virtual Desktop.

The protection can be enabled modifying the following registry key:

Name: **fEnableScreenCaptureProtection**

Type: **DWORD**

Value: **1**

Path: **HKEY_LOCAL_MACHINE\Policies\Microsoft\Windows NT\Terminal Services**

Chapter 4 Image Optimization

Virtual Desktop Optimization Tool

When a Windows operating system is working inside a virtual environment as a VDI/terminal server, it's possible to apply several optimizations to improve the overall performance and avoid unwanted system errors.

These "tricks" are performed by tuning some aspect of the operating system, as well as stopping/deleting services and components that are not useful in a virtualized environment.

Some optimizations can be applied to the base image, and others must be applied per host machine. It's generally not possible to retain them after a sysprep, so I need to create the final virtual machine first, and then apply the final tuning.

A very good tool that will automatically apply optimizations specifically for Azure Virtual Desktop is the Virtual Desktop Optimization Tool created by the "VDI Guys."

This tool comes in the form of a PowerShell script that shuts down services, modifies registry keys, applies local policies, and more.

Please test it in a demo environment first; it's very optimized, so there's a risk that something tuned down by the script is needed by some of your applications.

The good news is that because it is basically a script, it can be easily modified as needed.



Note: This tool is community driven, and it's not directly supported by Microsoft at the time of writing.

It's also possible to use third-party tools like Citrix Optimizer or VMware OS Optimization Tool, though created by Citrix and VMware for their VDI technologies, to prepare a generic Windows image that is running on a generic VDI system.

Logon blur prevention

A small but very useful optimization is to prevent the login screen from showing a blurred image. This is because the blurring process consumes a little bit of resources and slows down the logon process.



Figure 51: Azure Virtual Desktop Virtual Machine Logon with Blur

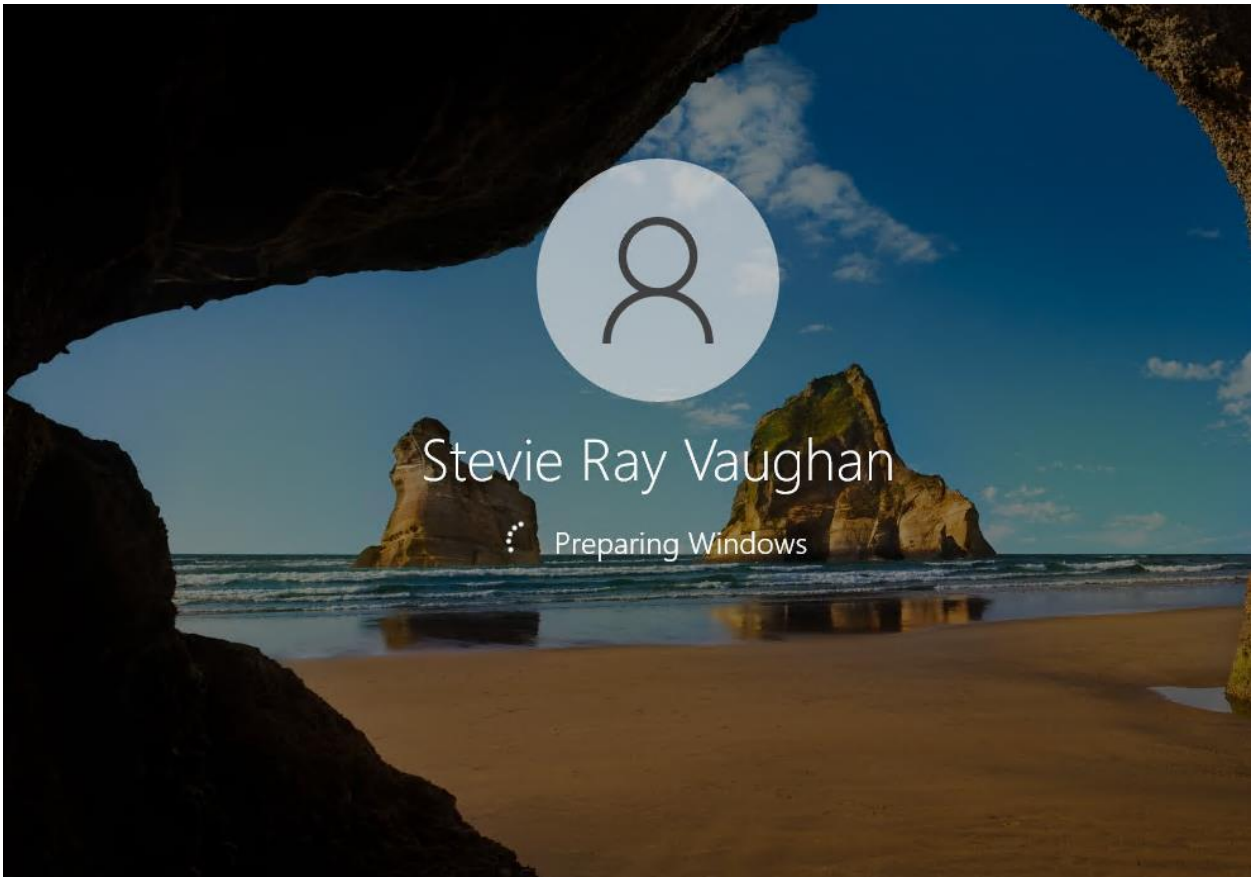


Figure 52: Azure Virtual Desktop Virtual Machine Logon without Blur

This optimization can be done using local policies, group policies, registry keys, or Intune (using the administrative templates).

You can find the group policy by navigating to the following:

Computer Configuration > **Administrative Templates** > **System** > **Logon** > **Show clear logon background**

The registry key that must be created is:

Name: **DisableAcrylicBackgroundOnLogon**

Type: **DWORD**

Value: **1**

Path: **HKEY_LOCAL_MACHINE\SOFTWARE\Policies\Microsoft\Windows\System**

Microsoft Edge tabs optimization

Another small optimization that can have a big impact is to put unused Microsoft Edge tabs on sleep. This is a quite recent feature that is designed to optimize the usage of CPU and RAM.

If Microsoft Edge (Chromium) is your browser of choice, you can use this setting to stop the automatic refresh of webpages that are open in tabs but are not visible because you are reading another tab or doing a different task using another application.

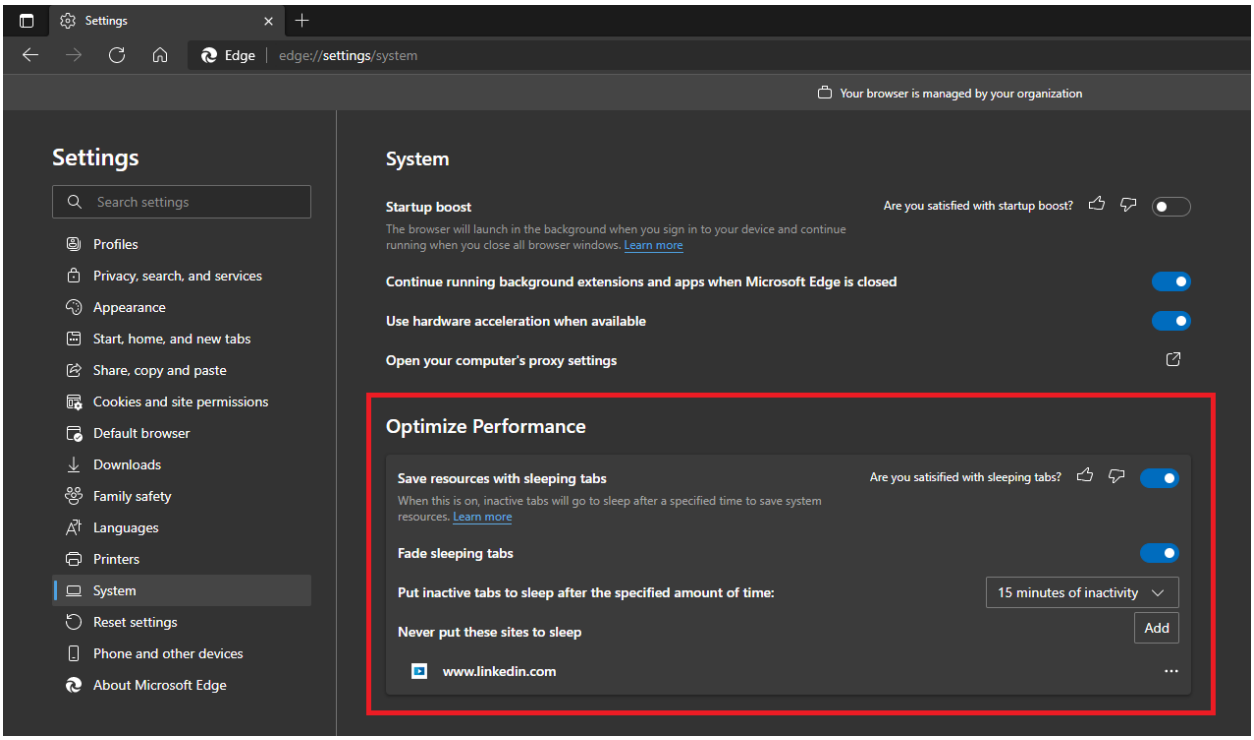


Figure 53: Microsoft Edge Sleeping Tabs

My configuration is putting the tabs that I'm not using after 15 minutes on sleep, but I have an exclusion for LinkedIn because I would like to be aware of incoming chats and notifications.

This optimization can be applied with Group Policy and Intune.

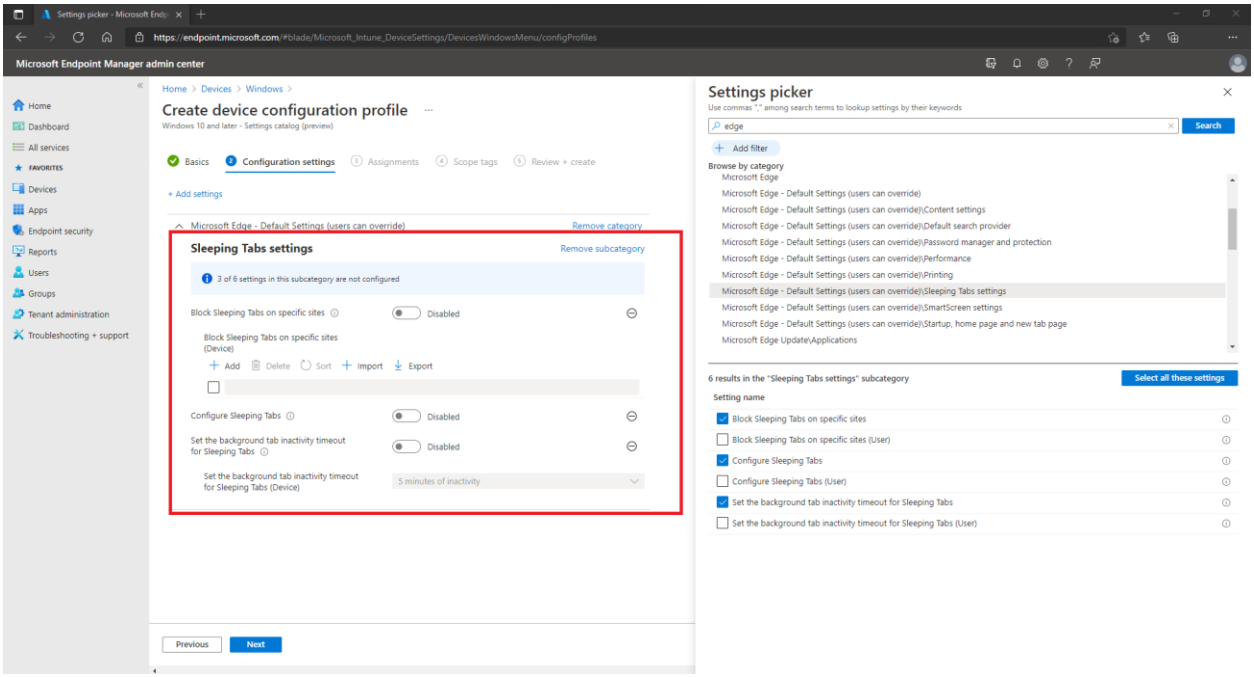


Figure 54: Sleeping Tabs Settings with Microsoft Intune

You can find more information about this feature here.

Chapter 5 How to Build a Laboratory

Automatic

In the summer of 2019, Azure Virtual Desktop was presented with the name of Windows Virtual Desktop, and deploying it in production or in a laboratory environment was quite tricky because the service was not fully integrated with Azure Active Directory, and because of the complete lack of a graphical interface.

Today, Azure Virtual Desktop is a very mature service that provides full integration with Azure and the rest of Microsoft ecosystem, and delivers a very good graphical interface inside the Azure Portal.

But there is more: a wizard that builds a full laboratory automatically.

This logic is good for proof of concepts and for learning the service, and it can be easily triggered from the Azure Virtual Desktop graphical user interface inside the Azure Portal.

The prerequisites are:

- An Azure subscription.
- An Azure Active Directory tenant.
- Global admin credentials.

Go to **https://portal.azure.com**.

In the search box, type **Azure Virtual Desktop** and click the icon.

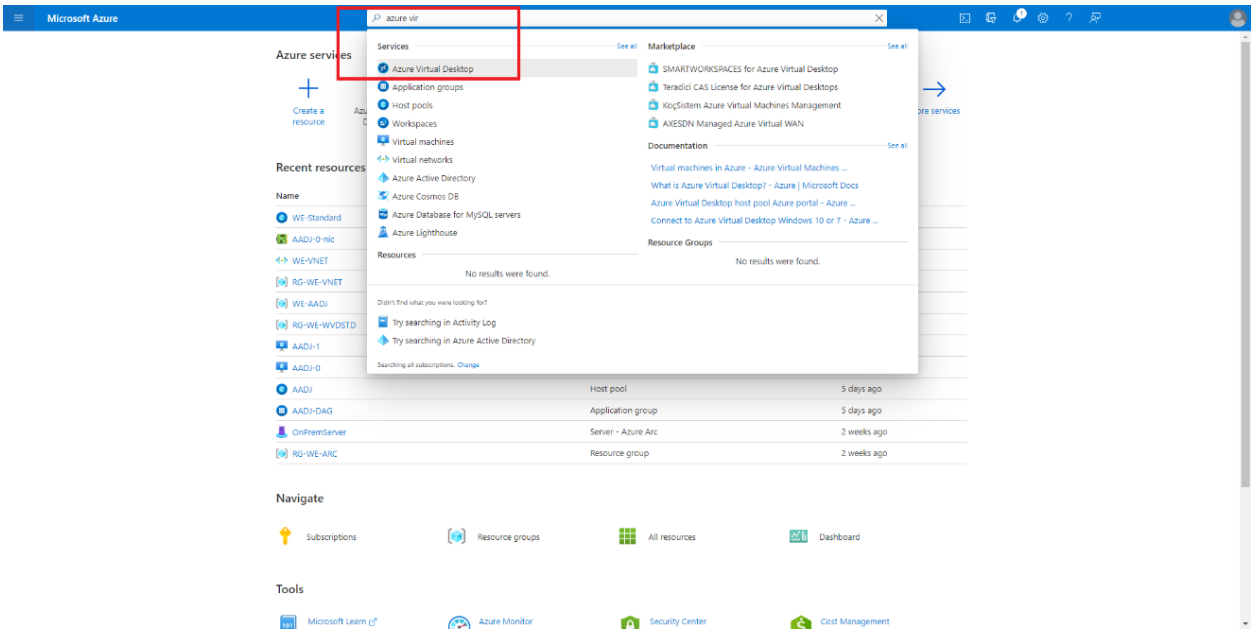


Figure 55: Search for Azure Virtual Desktop inside the Azure Portal

In the Azure Virtual Desktop graphic user interface, click **Getting started** > **Start**.

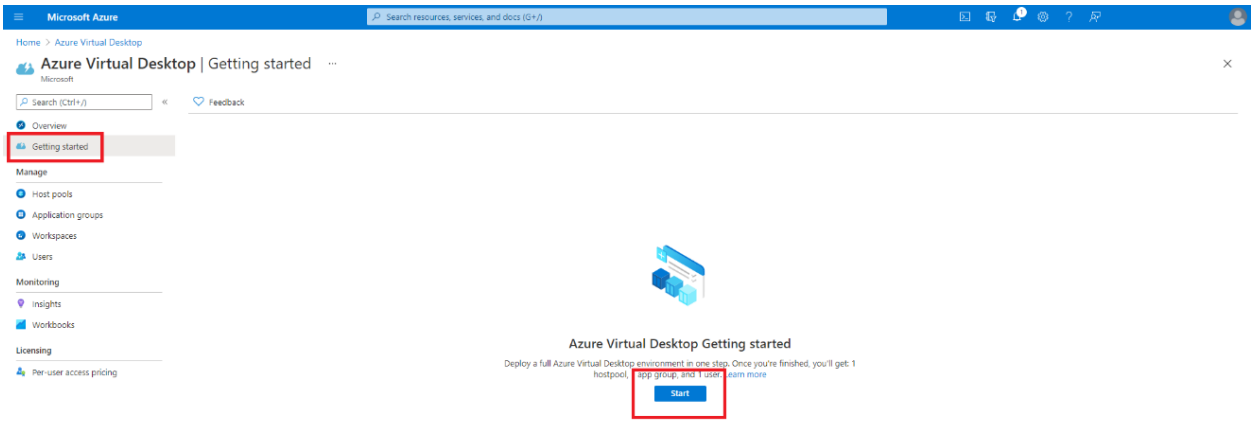


Figure 56: Getting Started Feature inside the Azure Virtual Desktop Graphical User Interface

The wizard can create a laboratory using an existing domain hosted on an existing Azure virtual machine that is acting as a domain controller, or it's creating an Azure Active Directory domain service domain.

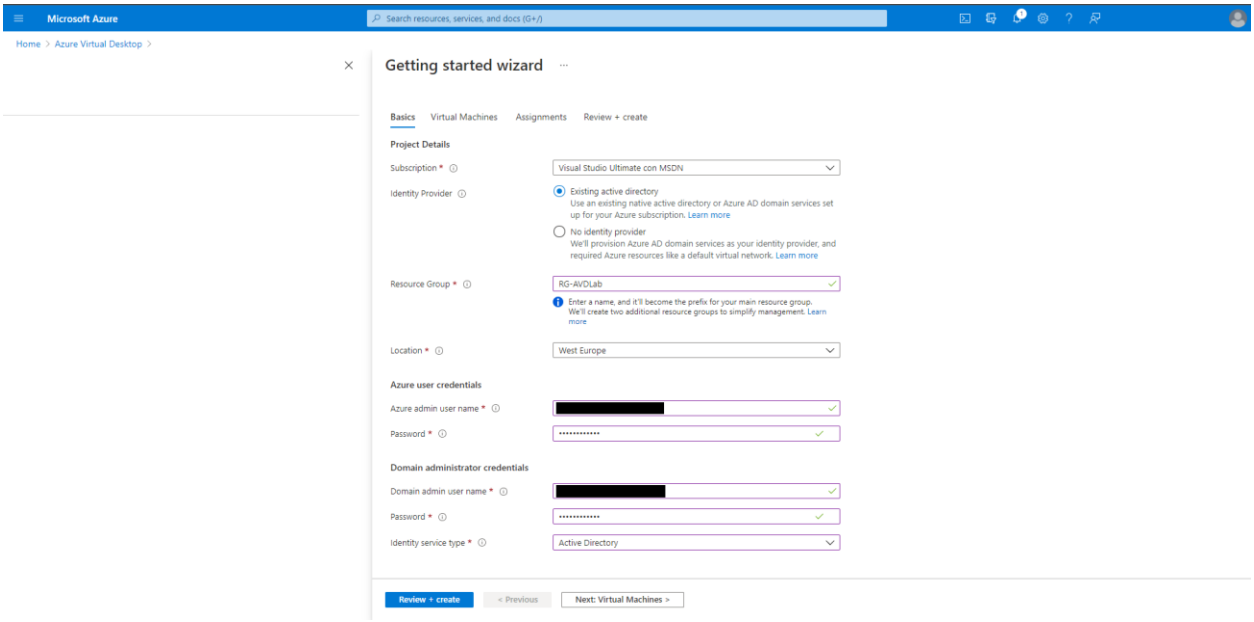


Figure 57: Getting Started Wizard, First Page

On the second page, I can choose the base image (Microsoft has a gallery of prebuilt and always-up-to-date images), the type of virtual machine (D series is good for multi-session), and the number of virtual machines that I want to create.

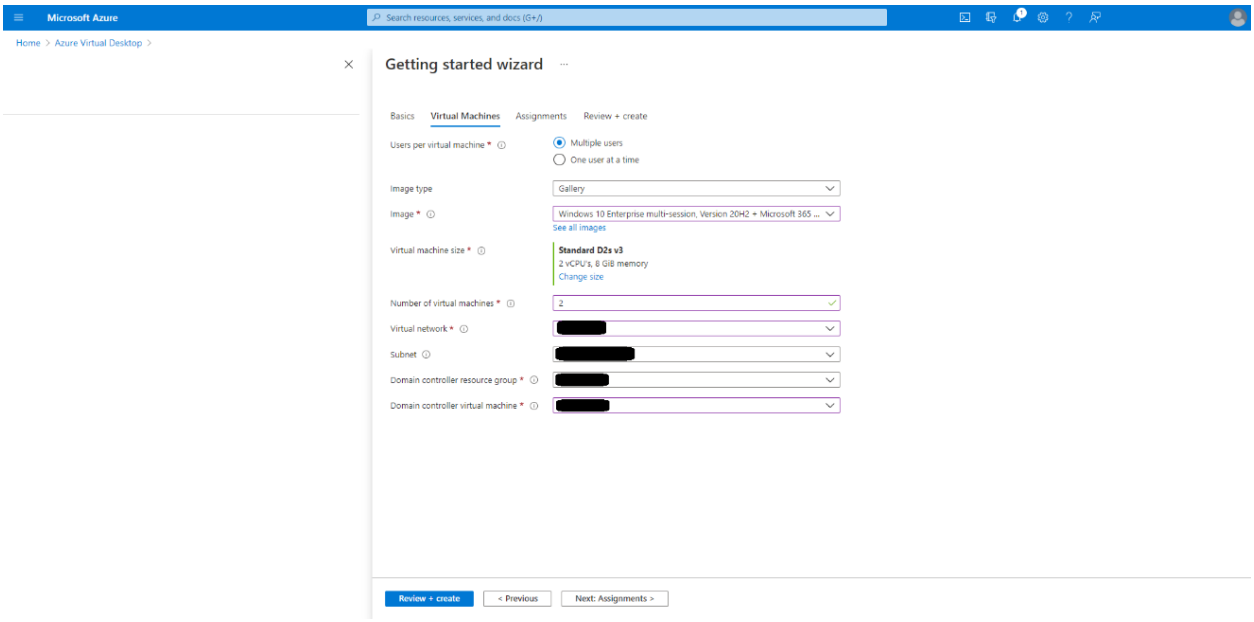


Figure 58: Getting Started Wizard, Second Page

Finally, I must create a user and assign the resource (the desktop access).

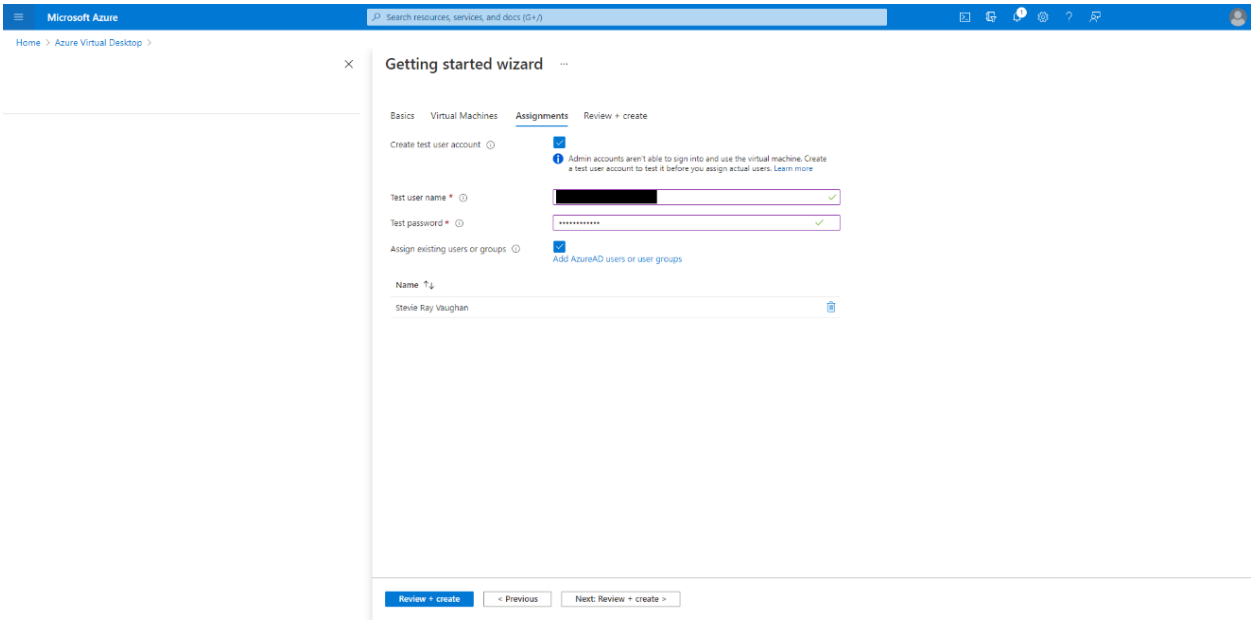


Figure 59: Getting Started Wizard, Third Page

Manual

The quickest way to manually create a simple laboratory is to use Azure Active Directory-only virtual machines (currently in preview).

The prerequisites are only an Azure tenant with a subscription, a virtual network created in the Azure region of choice, and the role of Global Administrator.

From the Azure Virtual Desktop portal, click **Create a host pool**.

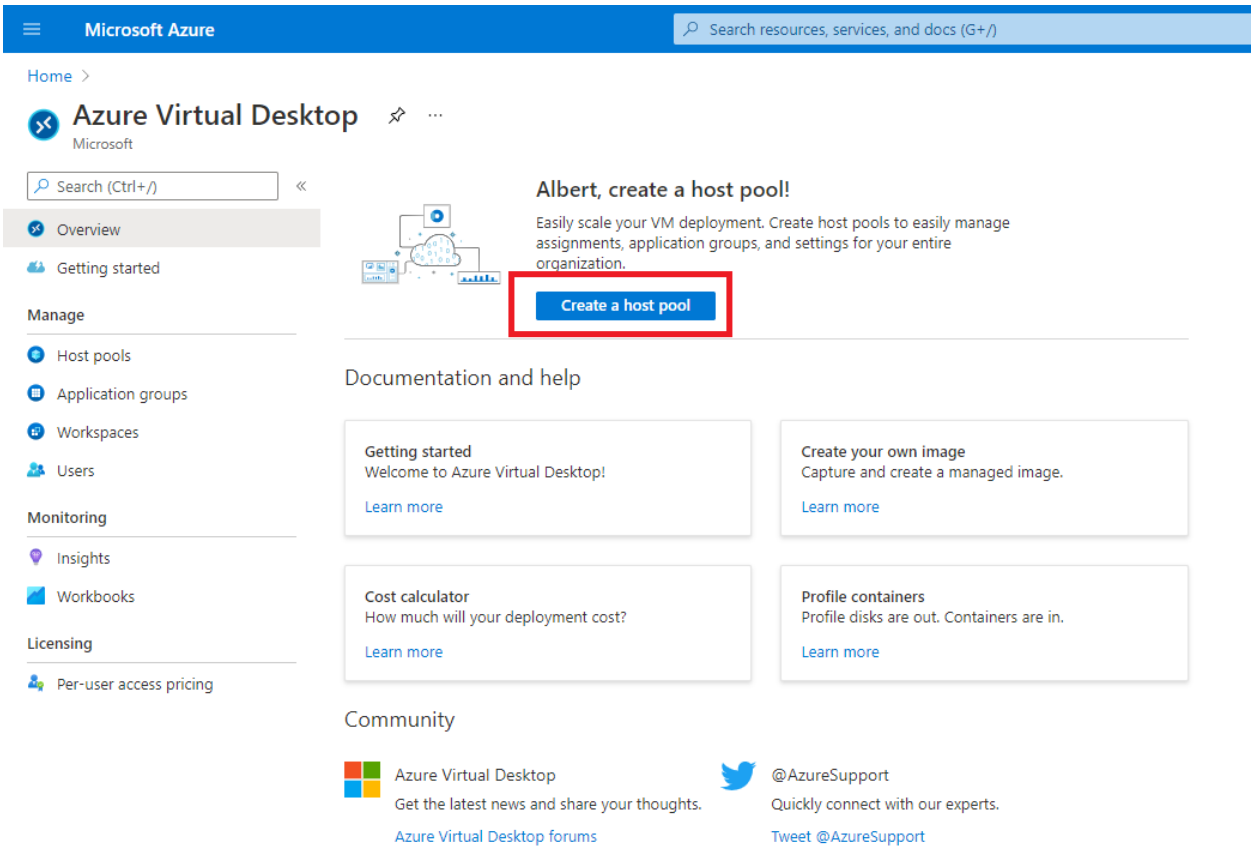


Figure 60: Create a Host Pool, First Step

Select the subscription, click **Create new**, and insert the name of your resource group, the name of the pool, and the location.

Note: Because the Azure Active Directory Join feature is still in preview, I need to select** Yes **for the** Validation environment **in this example (check the latest news about this feature).

Select the host pool type to specify if you prefer personal or pooled and click **Next**.

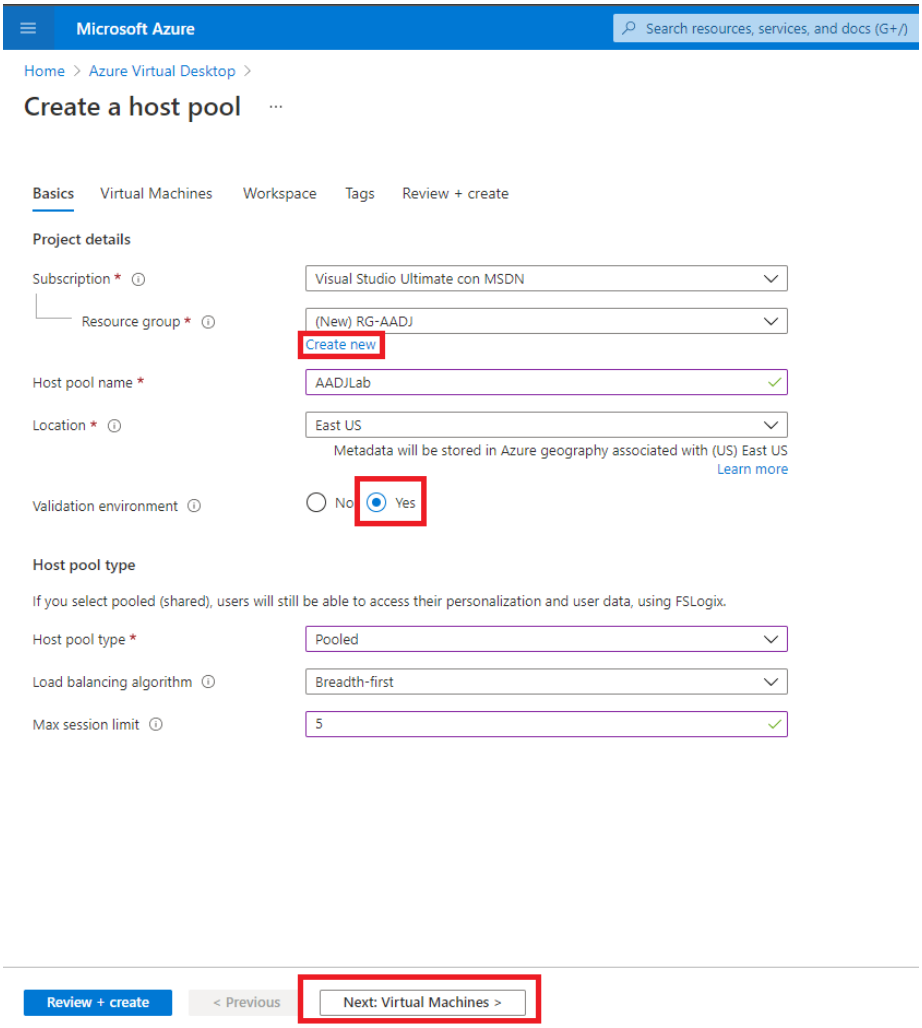


Figure 61: Create a Host Pool, Second Step

Now it's time to create the virtual machines. In the first half of the page, you must select **Yes** for **Add virtual machines** and choose the name prefix, the availability zone, and the image (choose one for the list of already available images). Finally, enter the number of virtual machines that you want to create; leave everything else as default.

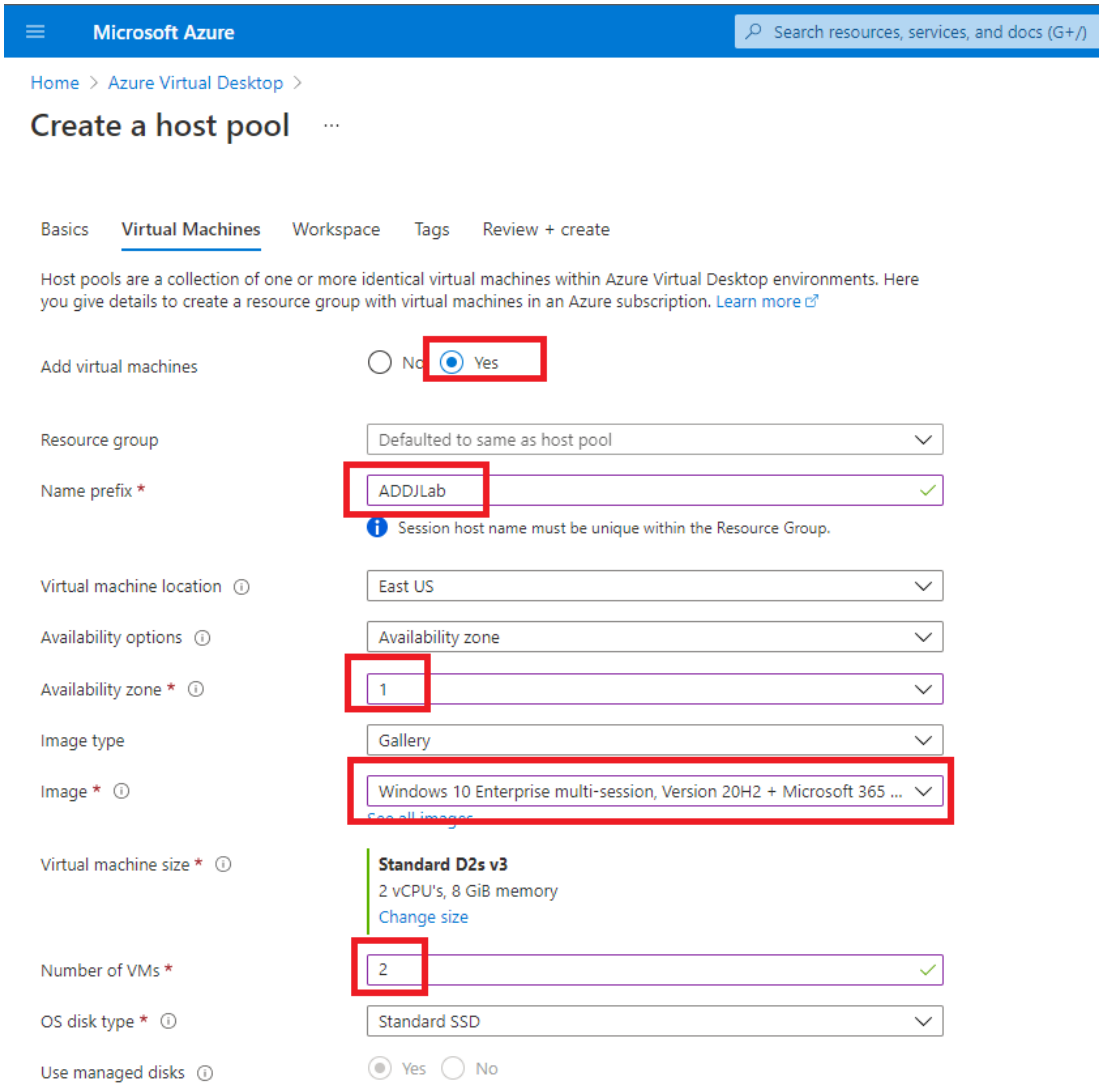


Figure 62: Create a Host Pool, Third Step, First Half

In the second half of the same page, you select the existing virtual network (that must be created in the same Azure Region of the host pool), select **Azure Active Directory Join**, and create a local admin for the virtual machines.

Click **Next**.

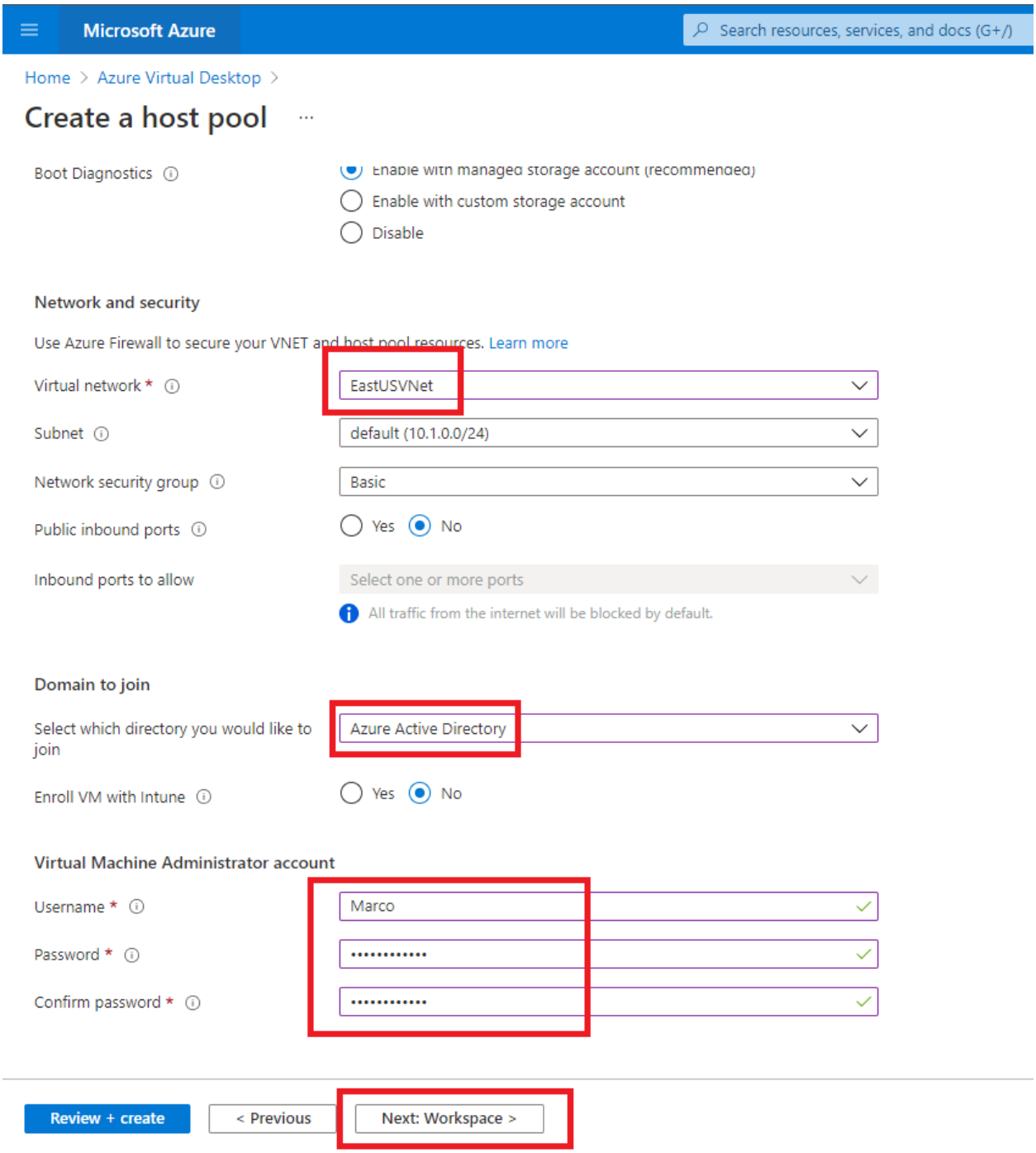


Figure 63: Create a Host Pool, Third Step, First Half

In the **Workspace** page, select **Yes** and **Create new** to enter the name of the workspace. Click **Review + create**.

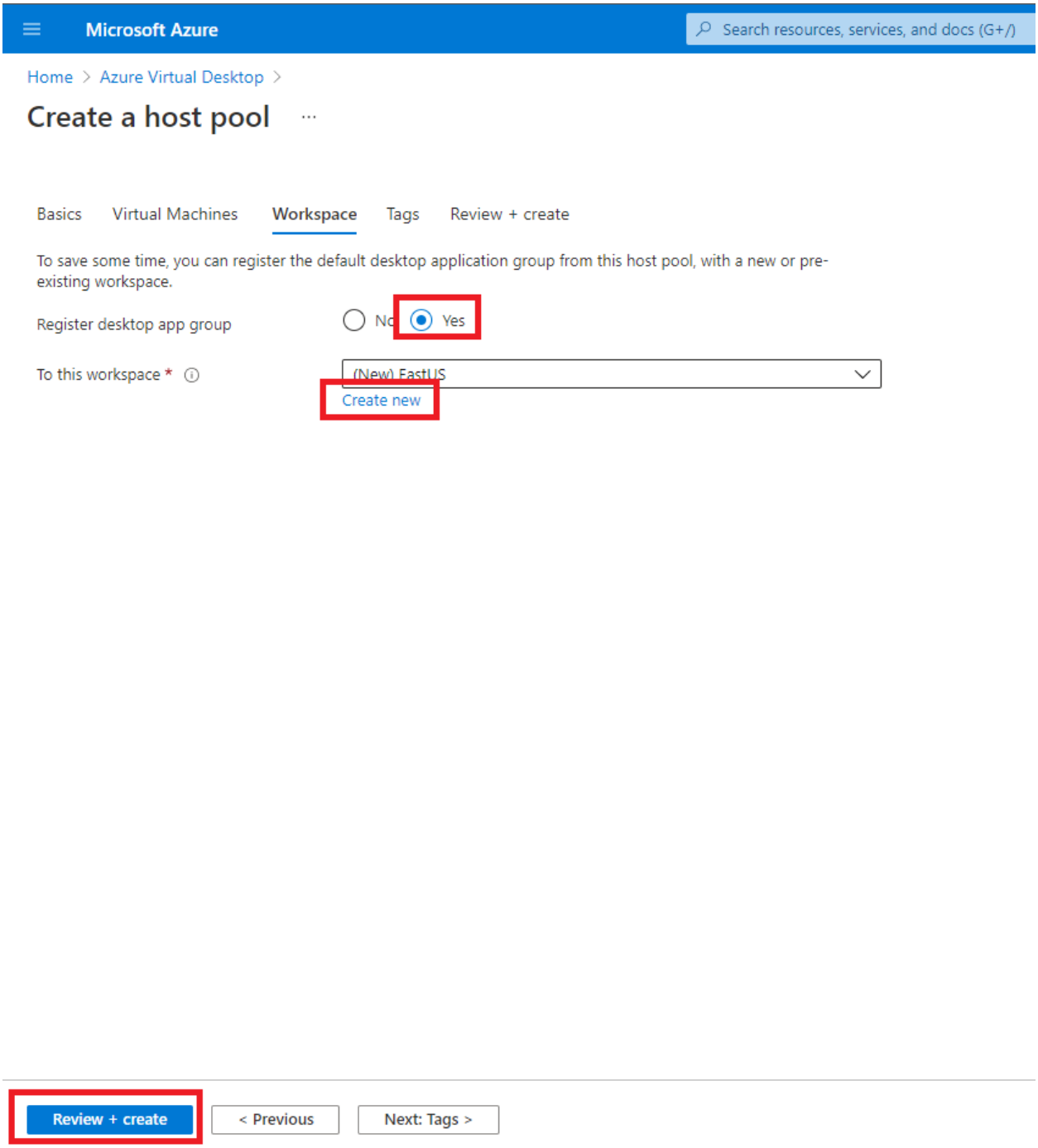


Figure 64: Create a Host Pool, Fourth Step

If everything is correct, you can click the **Create** button and wait for the end of the host pool creation.

At the end of the deployment, from the Azure Virtual Desktop Portal, click **Application Group** > <*the name of your pool*+**DAG**> > **Add**.

Select the name of a user that will consume the virtual machine desktop.

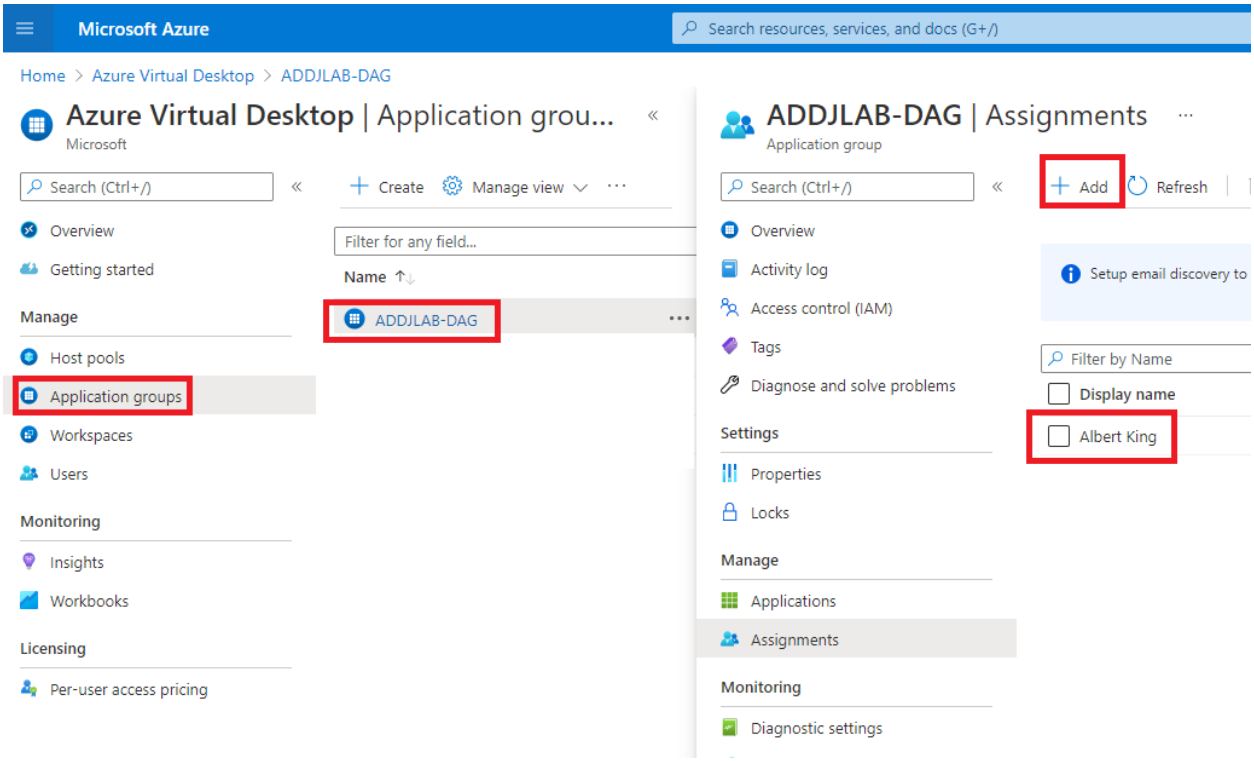


Figure 65: Application Group Assignment

From the Azure Virtual Desktop Portal, select **Host pools** > <*The name of your pool*> > **RDP Properties** > **Advanced**.

Manually type **;targetisaadjoined:i:1** at the end of the box and click **Save**.

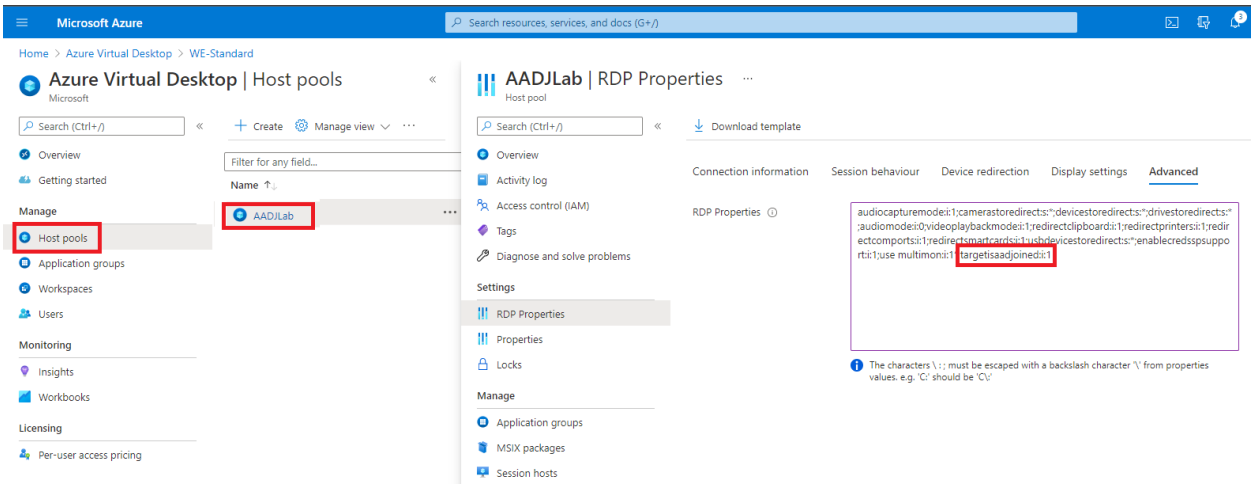


Figure 66: Host Pool RDP Properties

In the search box on top of the webpage, search for **Resource Groups** and select <*the name of your resource group*> > **Access Control (IAM)** > **Add**.

Add the **Virtual Machine Administrator Login** and **the Virtual Machine User Login** and assign them to your test user.

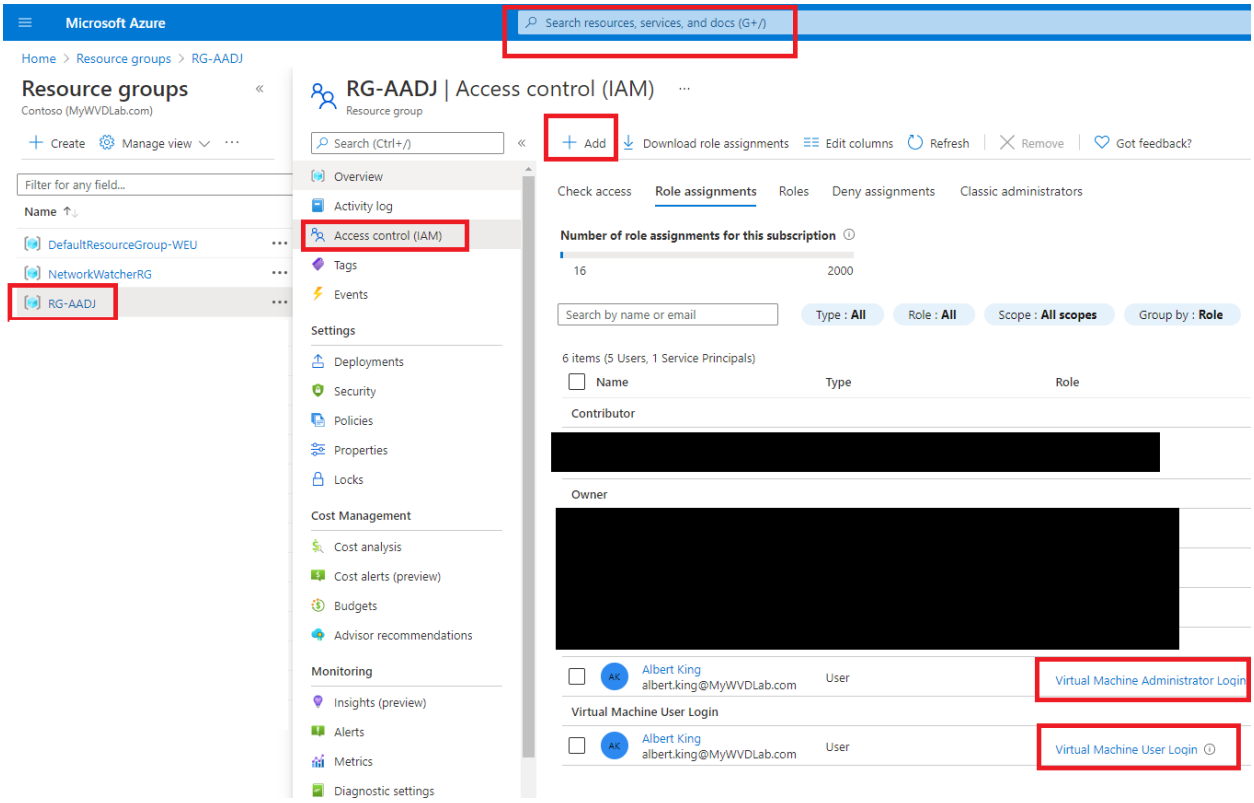


Figure 67: Resource Group Permissions

Now from a compatible browser, navigate to https://rdweb.wvd.microsoft.com/arm/webclient/index.html and perform authentication with your test user. Double-click the session desktop resource.

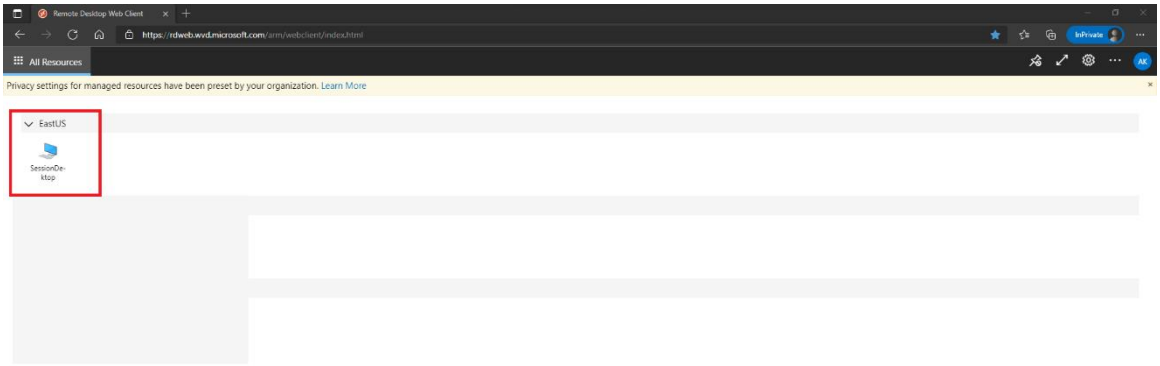


Figure 68: AVD web portal

You are now inside one of your test virtual machines.

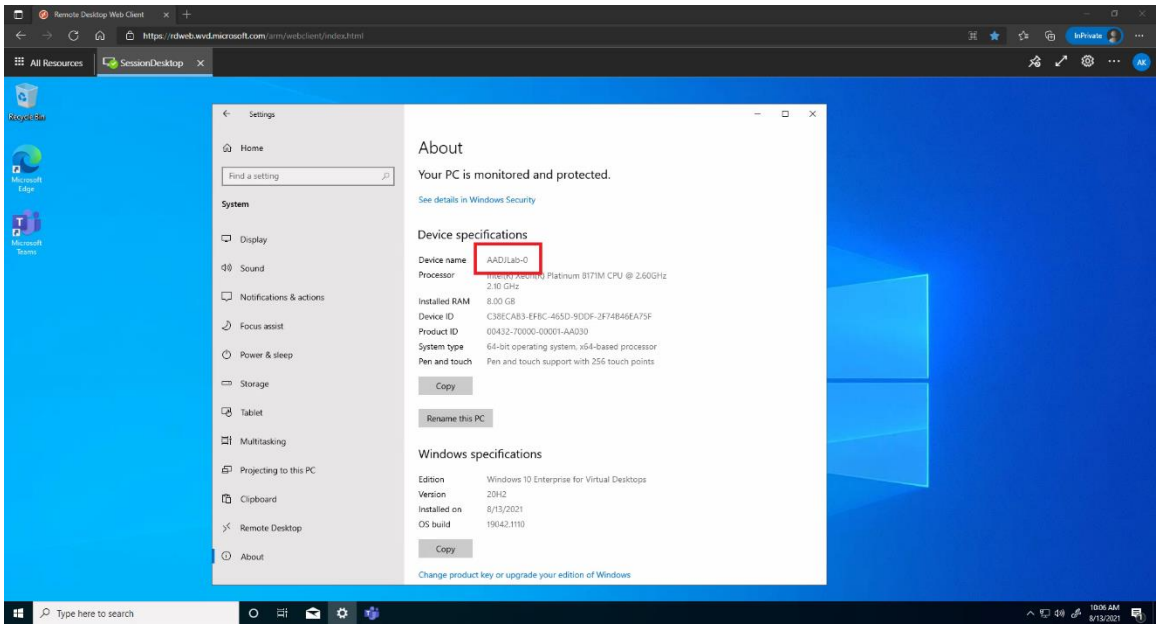


Figure 69: AVD Virtual Machine inside Web Browser

Chapter 6 Value-Added Partners

Azure Virtual Desktop is a complete solution, but it's also a platform where several partners are creating new services and tools that are improving the experience. The following is only a selection of partners I've tested in the past years.

Citrix Virtual Apps and Desktops with Azure

Citrix is a well-known company that is creating virtualization services and tools.

With Citrix technology, it's possible to publish resources that are exposed by physical devices, as well as virtual machines that are running on top of different virtualization products like Microsoft Hyper-V or VMware.

The HDX protocol created and developed by Citrix is very optimized, and their solution includes several interesting elements like autoscaling logic, monitoring, image provisioning, hybrid management, and many others, as shown in the following figure:

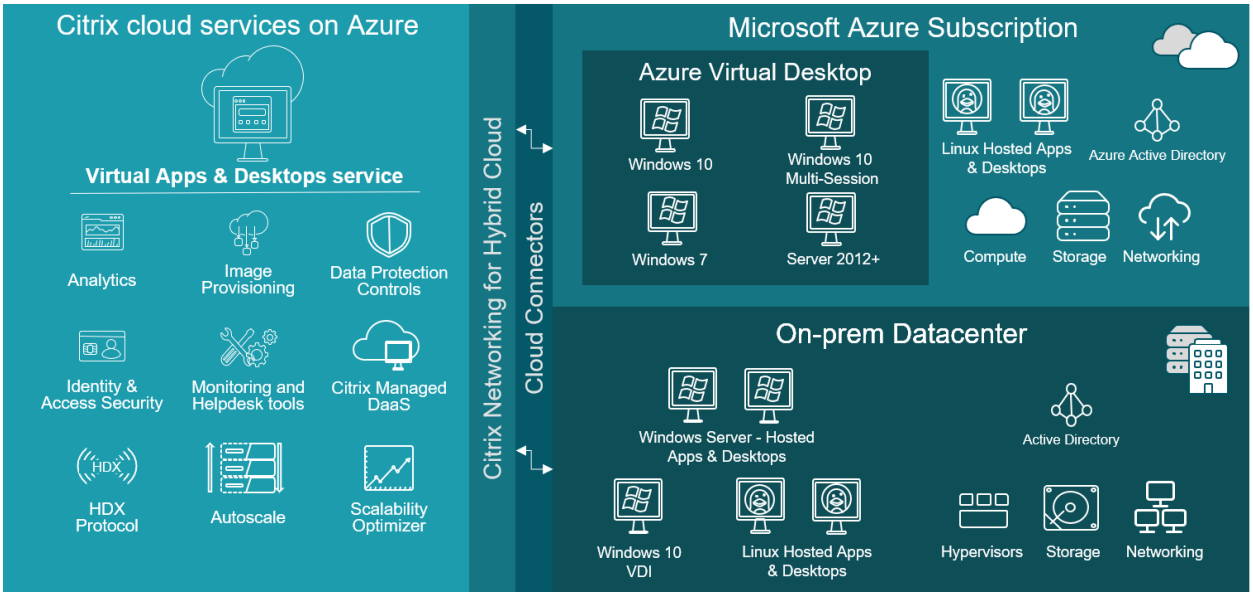


Figure 70: Citrix Virtual Apps and Desktops with Azure (Source)

Citrix is one of the two approved providers (the other one is VMware) that allows users to create a solution that does not use the Azure Virtual Desktop control plane, and it's publishing Windows 10 multi-session.

If I'm using a Citrix solution that is publishing Azure virtual machines, I can also benefit from the other advantages coming with Azure Virtual Desktop:

- Windows 10 multi-session.

- Windows Server license not needed (Linux rates).
- Windows 7 extended support included.

The Citrix solution uses Citrix technology delivered by the Citrix Cloud to publish Azure virtual machines. No Azure Virtual Desktop components like the virtual machine agent, the RDP protocol, the RD Gateway, or other Microsoft control plane roles are used.

Microsoft and Citrix also work together to allow the usage of Citrix technologies like Provisioning Services (PVS) inside Azure (currently in preview).

You can find a brief guide about this service here.

VMware Horizon Cloud Service on Microsoft Azure

VMware Horizon Cloud Service is another leader of the virtualization workspace. This is the reason why Microsoft has a partnership with VMware to integrate services like Azure VMware Solution (AVS) inside the Azure Cloud.

On the virtual desktop side, VMware Horizon Cloud can fully publish resources that are running inside Azure virtual machines and take advantage of the same capabilities that Azure Virtual Desktop and Citrix are offering:

- Windows 10 multi-session.
- Windows Server license not needed (Linux rates).
- Windows 7 extended support included.

Like the Citrix Cloud solution, the VMware proposition is based on a control plane that is assigning, publishing, and monitoring the Azure workload, and a connection that can be performed from any device using the PC-over-IP (PCoIP) protocol.

A high-level architecture is shown in the following figure:

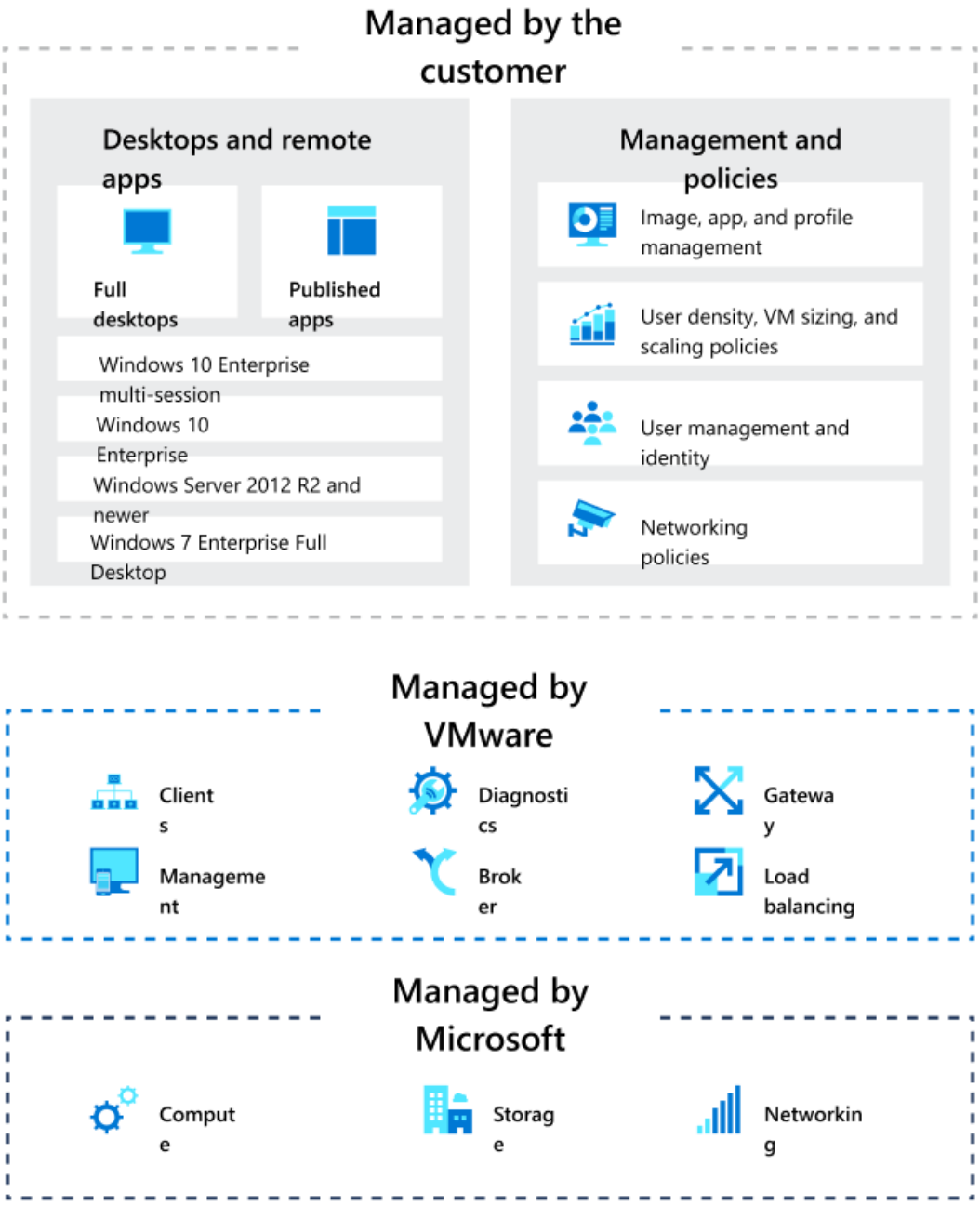


Figure 71: VMware Horizon Cloud Service on Microsoft Azure (Source)

Using this solution, I can take advantage of several interesting capabilities, like hybrid management, power management, Dynamic Environment Manager, and image optimization.

You can find a brief guide about this service here.

Thin clients

The world of thin clients is tightly linked with VDI services, and this is the reason why Microsoft supports connections from thin clients to Azure Virtual Desktop.

If the thin client solution is based on Windows 10 IoT Enterprise, it's possible to install the Windows MSI client and use it with all the advantages, like Microsoft Teams optimization (currently available only for the Windows 10 platform).

On the Linux side, Microsoft created an official SDK, and several vendors were able to create a direct link between their solutions and Azure Virtual Desktop.

Currently the supported solutions are:

- 10ZiG
- Dell
- HP
- IGEL
- NComputing
- Stratodesk

Check the official documentation for news about this topic.

Lakeside

Lakeside is a free tool that can perform an assessment of a current VDI architecture (on-premises or in another cloud) or physical devices to establish the right sizing on Azure Virtual Desktop.

The solution collects many counters and indicators from the devices, and it creates useful reports and suggestions about how to translate the current experience into Azure Virtual Desktop.

The tool is integrated into the Azure Migrate service.

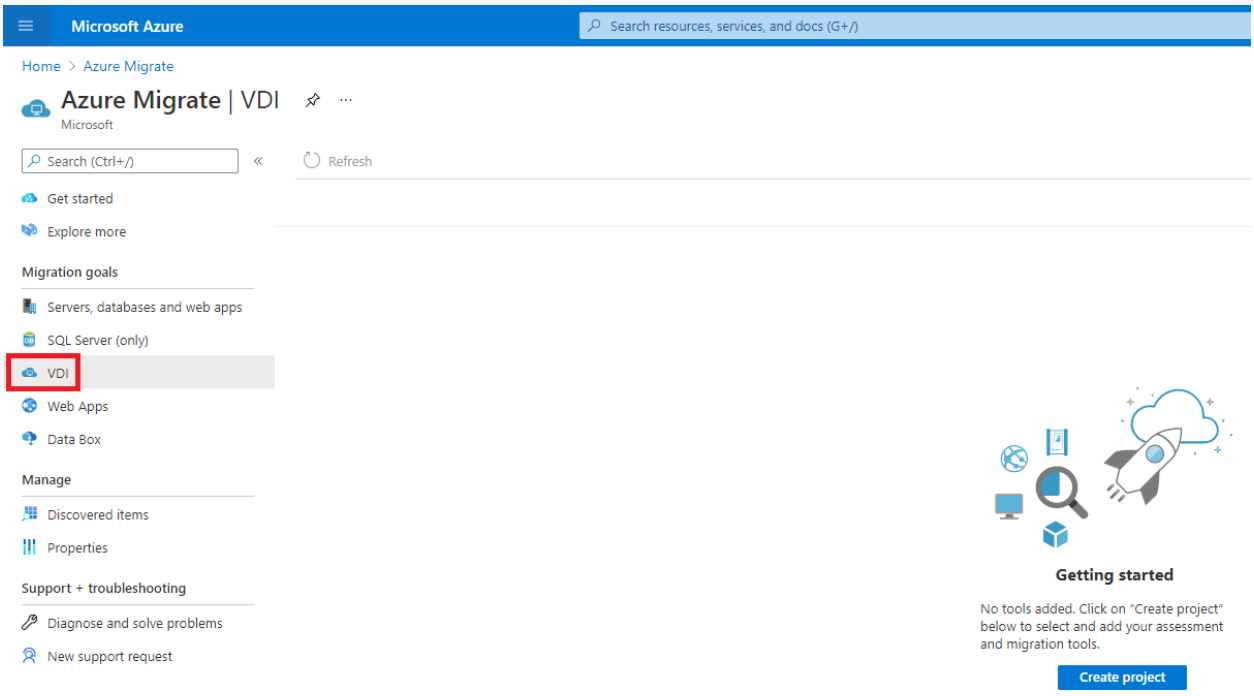


Figure 72: Lakeside Assessment Tool inside Azure Migrate

Nexthink

Another interesting tool that I find useful is Nexthink. This tool collects information in real time from user devices so I can use it to collect performance data coming from my Azure Virtual Desktop session hosts.

The quantity and the quality of data collected is very deep, and the way it's presented in reports helps me understand how the Azure Virtual Desktop is performing (errors, boot time, processor usage, application performance).

There is also the ability to directly engage the end user in a survey, as shown in the following figure:

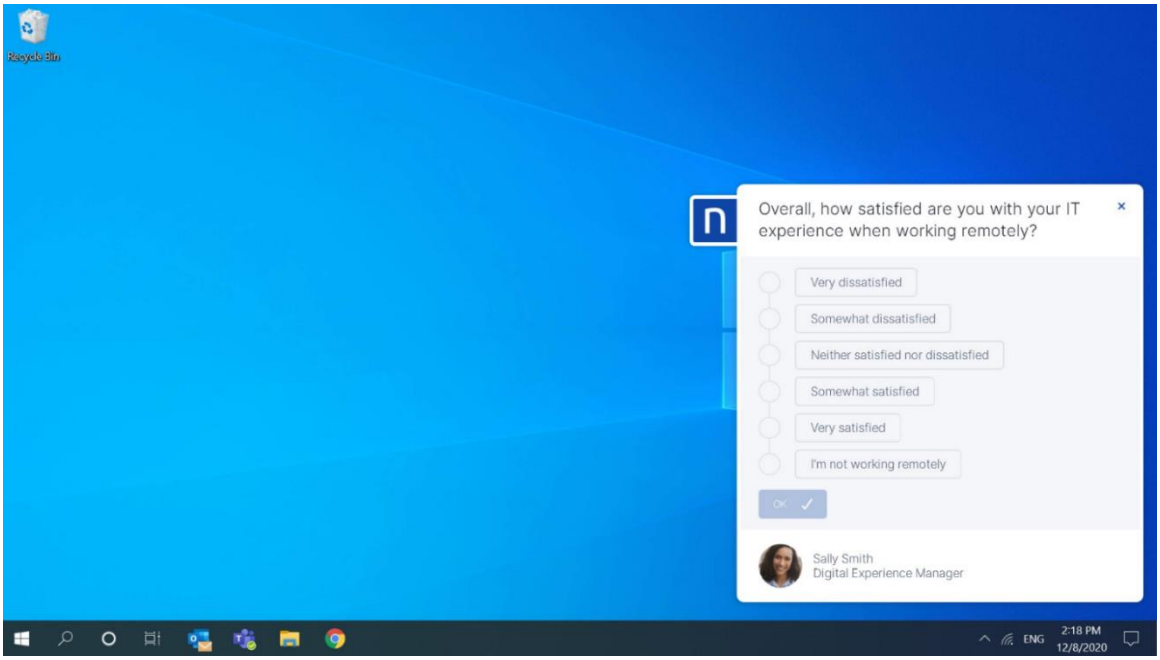


Figure 73: Nexthink end user survey (source)

This tool also works with Citrix and VMware.

You can find more information about Nexthink here.

Nerdio

Nerdio is a company that builds tools to help Microsoft Service Providers (MSPs) as well as enterprises managing Microsoft Azure technology. The tools are built around the concepts of simplicity and automation, and are a very good fit if you don't have deep Azure skills and/or if you want to oversee many Azure Virtual Desktop host pools and users with minimal effort.

Nerdio Manager for MSPs and Enterprise are two "twin" services that are focused on Azure Virtual Desktop management.

The following figure shows the key aspects of Nerdio Manager for MSPs:



Figure 74: Nerdio Manager Console (Source)

They have interesting capabilities like virtual machine auto-heal, advanced auto scale based on Azure virtual machine scale set, the ability to easily deploy host pools based on ephemeral disks, advanced monitoring, image creation and update processes, and more.

You can find more information about this tool here.

Chapter 7 Windows 365

Recently, Microsoft announced a new cloud service called Windows 365 (W365). This service is based on Azure Virtual Desktop, and is basically publishing virtual machines based on Windows 10 or 11 (when available).

The machines are called Cloud PCs, and the goal of Microsoft is to create a new category, a new way of being productive on any device in any place.

The licensing is based on a monthly fixed price that depends on the size of the Cloud PC that you need (a 4 vCPU + 16 GB of RAM license costs more than a 2 vCPU + 8 GB of RAM one).

The connection can be made from a dedicated web portal or by using the same Windows, iOS, Android, and macOS clients that are available for Azure Virtual Desktop.

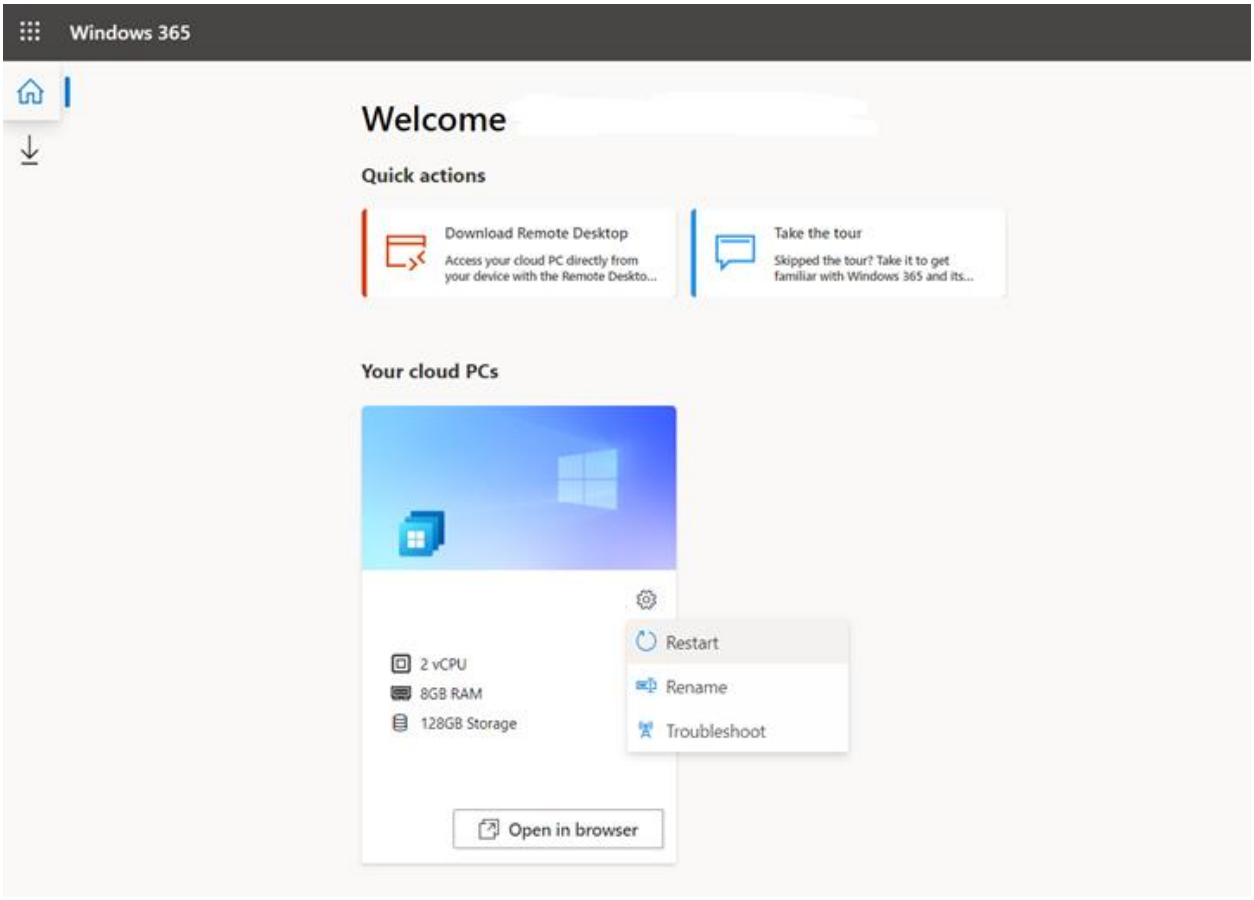


Figure 75: Windows 365 Browser Experience

From the web browser, the user can restart the virtual machine (that is always active), rename it, and (in the future) resize it, if allowed by the IT department.

Even if the underlying engine is Azure Virtual Desktop, there are several differences between the two services:

Table 4: Windows 365 versus Azure Virtual Desktop Service Features

Windows 365 (W365)	Azure Virtual Desktop (AVD)
Fixed monthly price	Price based on consumption
Machines available 24 hours per day	Machines can be shut down to optimize costs
Single session	Single session or pooled
Windows 10 or 11 (when available)	Windows 7, 10, 11 (when available) Windows Server 2012 or later
Remote desktop only	Remote desktop or application publishing
Local profile only	Local profile or FSLogix profile
MSIX App Attach not supported	MSIX App Attach supported
Built for simplicity	Built for elasticity and any kind of scenario

You can see an interactive demo of Windows 365 here.

In terms of resiliency, Windows 365 is built to grant high availability and disaster recovery, as explained in this official article.

Windows 365 is offered in two versions: Business and Enterprise.

Windows 365 Business

The Business version of the service is the simplest proposition, and it comes with no prerequisites.

If the customer already has a Microsoft 365 tenant (maybe because of email or other services), they can purchase the Windows 365 licenses and assign them to users.

If the customer doesn't have a Microsoft 365 tenant, it's enough to own a Windows 365 license, and the service is created automatically inside a brand-new tenant.

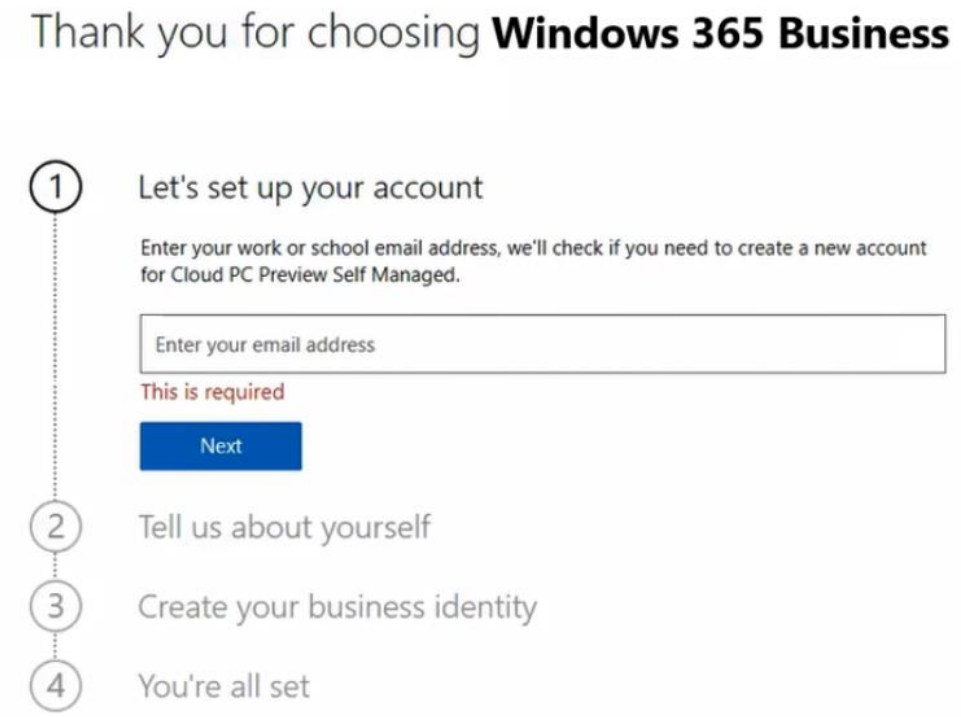


Figure 76: Windows 365 Business License Assignment

In any case, in a matter of minutes, the virtual machine is provisioned and joined to Azure Active Directory, and the assigned user is promoted to local administrator of the machine.

Having the virtual machine joined to Active Directory Domain Services or Azure Active Directory Domain Services is not supported.

The end user can connect to the machine using the web interface windows365.microsoft.com or the multiplatform client that is available for Azure Virtual Desktop.

The operating system is provided by Microsoft, and it already includes M365 Apps, Microsoft Teams, and all the optimization needed.

You can find more information about this version of Windows 365 here.

Windows 365 Enterprise

The Enterprise version of Windows 365 targets larger companies that have an IT department that manages devices in terms of configurations, applications, and security.

This version currently has several prerequisites (but check the actual one, because the service was just announced at the time of writing this book).

The official FAQ webpage states:

> "To use Windows 365 Enterprise, each user must be licensed for Windows 10 Enterprise or Windows 11 Enterprise, Microsoft Endpoint Manager, and Azure Active Directory P1. In addition to being available independently, these licenses are included in Microsoft 365 F3, Microsoft 365 E3, Microsoft 365 E5, Microsoft 365 A3, Microsoft 365 A5, Microsoft 365 Business Premium, and Microsoft 365 Education Student Use Benefit subscriptions."

Regarding the architecture, currently the Cloud PCs are created inside an Azure tenant managed by Microsoft (just like the Business version of Windows 365), but they must be hybrid-joined with a company Azure Active Directory + Active Directory Domain Service (Azure Active Directory Domain Services is not supported).

Therefore, you must have an Azure tenant with a subscription attached, and create a virtual network that will be injected into the Cloud PC.

So basically, the Cloud PC is running in a tenant managed by Microsoft that is a sort of black box that is not directly accessible, but the Cloud PC communications with the Azure Active Directory domain, the Domain Controller, and the internet are going through the provided virtual network.

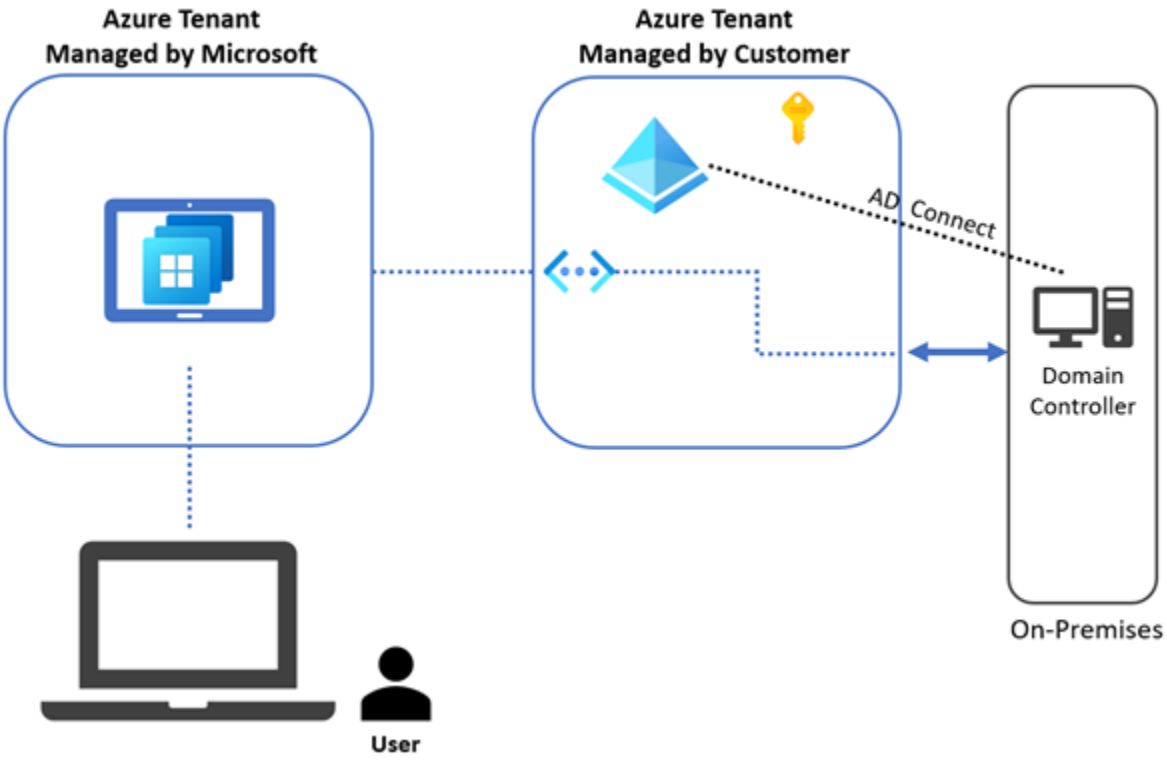


Figure 77: Windows 365 Enterprise High-Level Architecture

In the Windows 365 roadmap, there is the possibility to have Windows 365 Enterprise Cloud PCs joined only to an Azure Active Directory.

This new feature helps avoid the need for a customer to provide an infrastructure and create a virtual network. Check this page for more information.

On the management side, the Cloud PCs can be managed by Microsoft Endpoint Manager (MEM), which is the combination of Microsoft Intune + System Center Configuration Manager.

Inside the MEM console, under **Devices,** you can find the icon related to Windows 365.

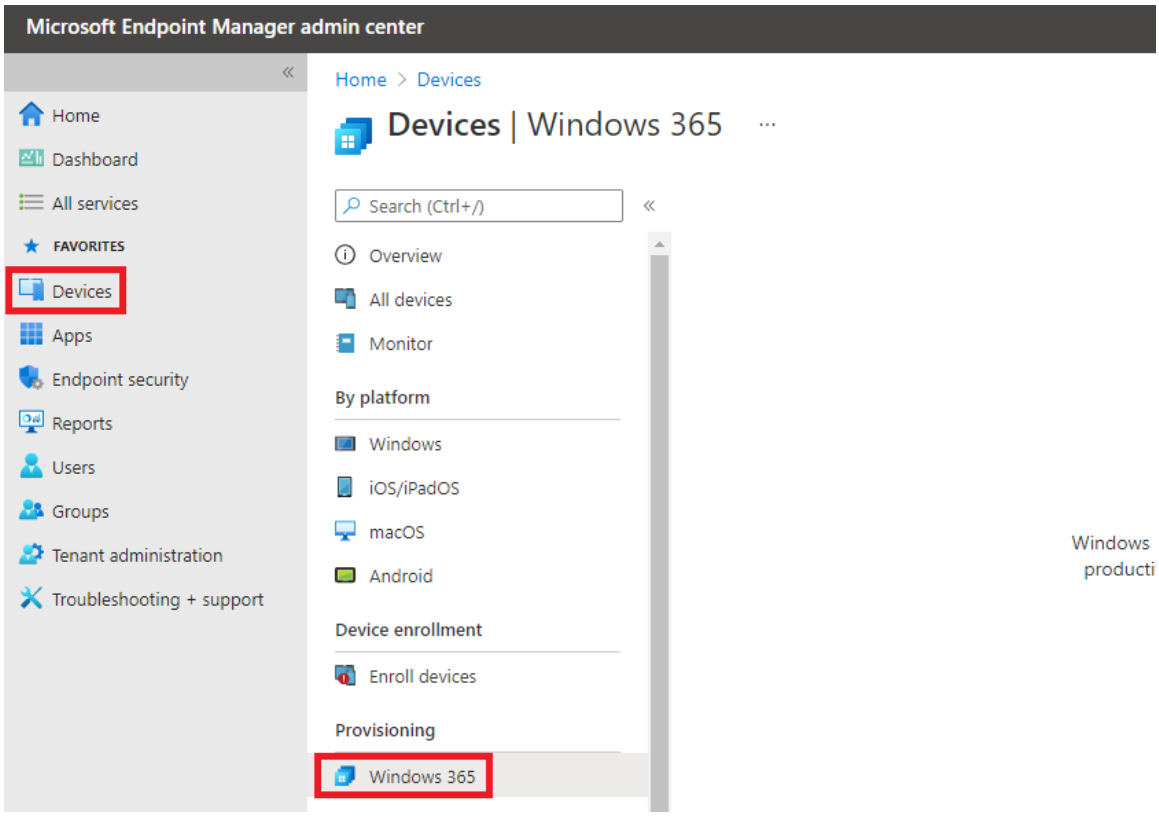


Figure 78: Windows 365 inside the MEM Console

Using the MEM console, it's possible to provision the Cloud PCs, assign configurations and applications, and monitor them, thanks to the dedicated watchdog service that is continuously checking the status of the Cloud PCs.

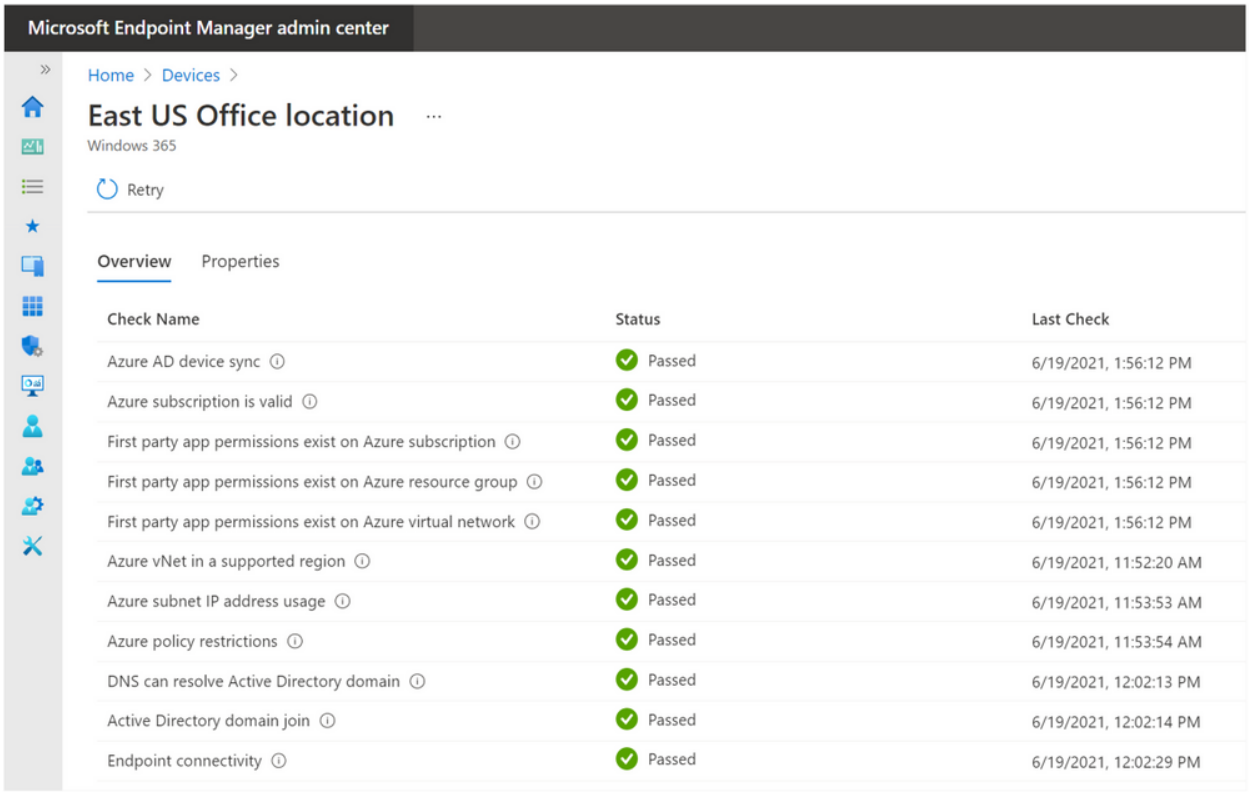


Figure 79: Windows 365 Watchdog Service

With Microsoft Endpoint Manager, it is possible to apply the dedicated security baseline created and updated by the Windows 365 product team.

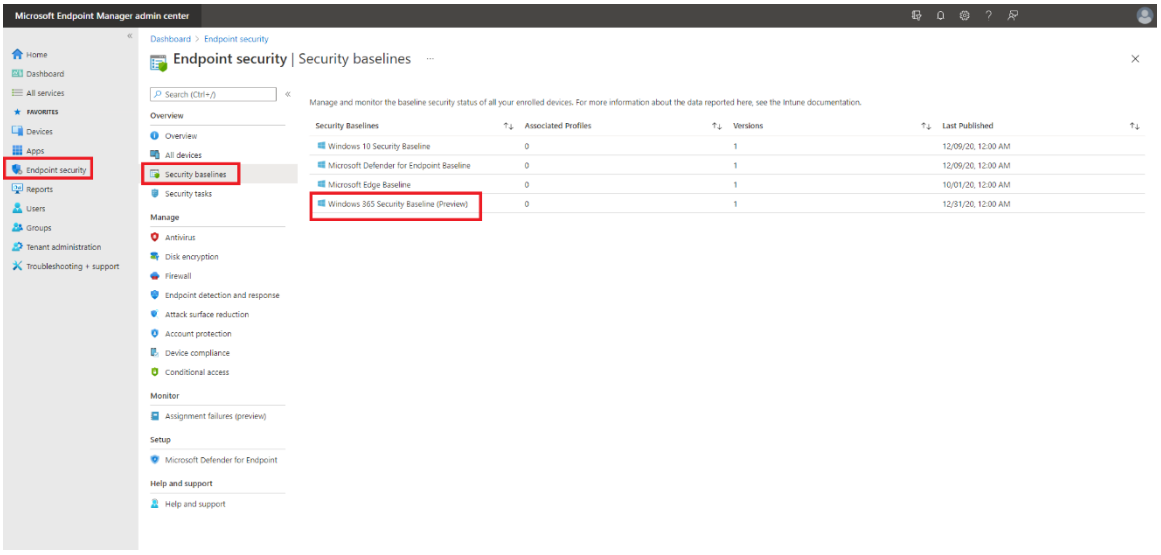


Figure 80: Windows 365 Security Baseline

Take care with the updates that are published for Windows 10 and 11 that must be installed by the customer (like in Azure Virtual Desktop) and the Windows Defender antivirus tool that is included in every Cloud PC.

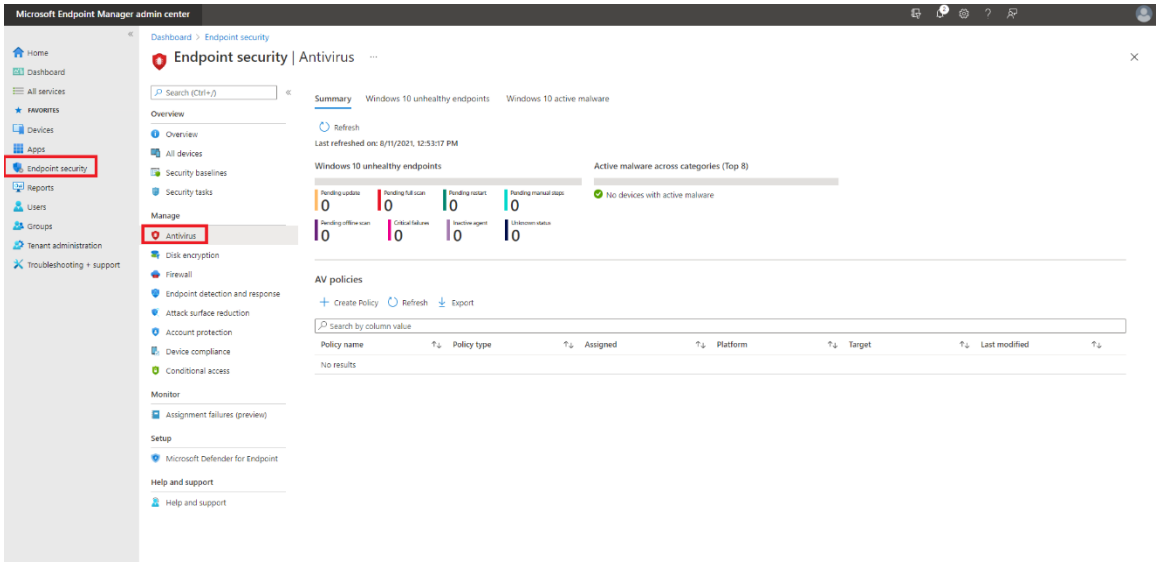


Figure 81: Antivirus Management inside the MEM Console

You can find more information about this version of Windows 365 here.

Resources

The best way to get to know Azure Virtual Desktop is to follow the official learning path that is always updated with the last news

Another incredibly rich source of knowledge is the official playlist on the Azure Academy YouTube channel.

And obviously, it's good to read the official documentation.

Several unofficial user groups and communities around the world are very active in covering the Azure Virtual Desktop topic in different languages. You can find the one nearest you here.

To demonstrate your expertise, you can take the official exam: AZ-140.

Finally, partners with an active gold competency can achieve a dedicated advanced specialization to show their expertise.

Conclusion

Azure Virtual Desktop is a cloud-only service that publishes the resources available inside Azure virtual machines, like the entire desktop or single applications.

The ability to quickly create, modify, and decommission a host pool is perfect for supporting mergers and acquisitions or a period of high demand.

The centralization of data and management provides security and an easy way to update the operating systems and applications.

The natural exposure on the internet allows users to work from any place on any device, and this supports smart working, remote working, and bring-your-own-device policies.

The ability to use very powerful Azure virtual machines with dedicated graphics power can unlock almost every productivity scenario.

Unique technical features like the ability to use Windows 10 multi-session can unlock new possibilities, and the presence of classic operating systems like Windows 7 and Windows Server 2012 can support legacy applications and the transition to modern devices.

Several license advantages like the extended support included for Windows 7 or the Linux rates for the Windows Server virtual machines can save money, as can the automations that start the virtual machines only when needed.

The VDI architecture is simplified by the presence of a control plane that publishes the Azure Virtual Desktop resources is available in several key Azure Regions and free of charge.

Whether you are coming from a classic on-premises VDI architecture, or you have never implemented one, Azure Virtual Desktop is very easy to implement and manage.