ORACLE

Citrix Virtual Apps and Desktops deployment on Compute Cloud@Customer or Private Cloud Appliance

Version 1.0

Public

ORACLE

Purpose statement

The purpose of this solution paper is to describe a validated reference architecture for deploying Citrix Virtual Apps and Desktops on Oracle Compute Cloud@Customer (C3) and Oracle Private Cloud Appliance (PCA). This document outlines how enterprises can deliver secure, scalable, and high-performance virtual desktops and applications while maintaining full control over data residency, compliance, and infrastructure operations. It provides architectural guidance, deployment considerations, and key benefits for organizations seeking to modernize end-user computing using a cloud-operating model within their own datacenters or sovereign environments.

Disclaimer

ORACLE

Table of contents

Introduction

As enterprises accelerate digital workplace initiatives, Citrix Virtual Apps and Desktops have emerged as a strategic platform for delivering secure, flexible, and centrally managed virtual desktops and applications. However, many organizations operate under strict requirements related to data sovereignty, regulatory compliance, latency sensitivity, and operational control, which can limit the adoption of fully public cloud-hosted DaaS solutions.

Oracle Compute Cloud@Customer (C3) and Oracle Private Cloud Appliance (PCA) address these challenges by providing a cloud-consistent operating model. They enable organizations to deploy and operate enterprise workloads on dedicated infrastructure while maintaining full control over data, security, and compliance. When combined with Citrix Virtual Apps and Desktops, Compute Cloud@Customer (C3) or Oracle Private Cloud Appliance (PCA) support the delivery of enterprise-grade virtual desktops using a centralized Citrix control plane, while keeping desktops, user data, and integrations local to the customer's datacenter.

This solution paper presents a reference architecture for running Citrix Virtual Apps and Desktops on Oracle Compute Cloud@Customer (C3) or Oracle Private Cloud Appliance (PCA), highlighting key design principles, integration points, and operational benefits. It demonstrates how customers can achieve the agility and scalability of cloud-based desktop services while preserving security, performance, and compliance, making it an ideal solution for regulated industries, hybrid IT strategies, and edge or disconnected environments.

Note: This content is provided for informational purposes and self-supported guidance only. Consultancy or other assistance related to the content is not covered under the Oracle Support contract or associated service requests. If you have questions or additional needs, then please reach out to your Oracle Sales contact directly.

Architecture Overview

The architecture outlined below is designed for production-grade performance and high availability, supporting Citrix Virtual Apps and Desktops deployed on Compute Cloud@Customer (C3) or Private Cloud Appliance (PCA). This solution operates independently of Citrix MCS integration, using OpenTofu for desktop provisioning. It features F5 BIG-IP Virtual Edition as the front-end load balancer, leverages three fault domains for enhanced resiliency, and utilizes Windows Server 2025 or Windows 11 as the platform for virtual desktops and application hosting.

The architecture is meticulously engineered to ensure continuous service availability. By distributing critical infrastructure components, such as Delivery Controllers (DDCs), StoreFront servers, SQL databases, and desktop workloads, across Compute Cloud@Customer (C3) or Private Cloud Appliance (PCA) fault domains. The F5 BIG-IP appliance continuously monitors the health of backend components and intelligently routes authentication and ICA traffic only to operational nodes. This ensures that the control plane, authentication services, session brokering, and desktop delivery remain fully operational, maintaining a seamless and uninterrupted user experience, even in degraded conditions.

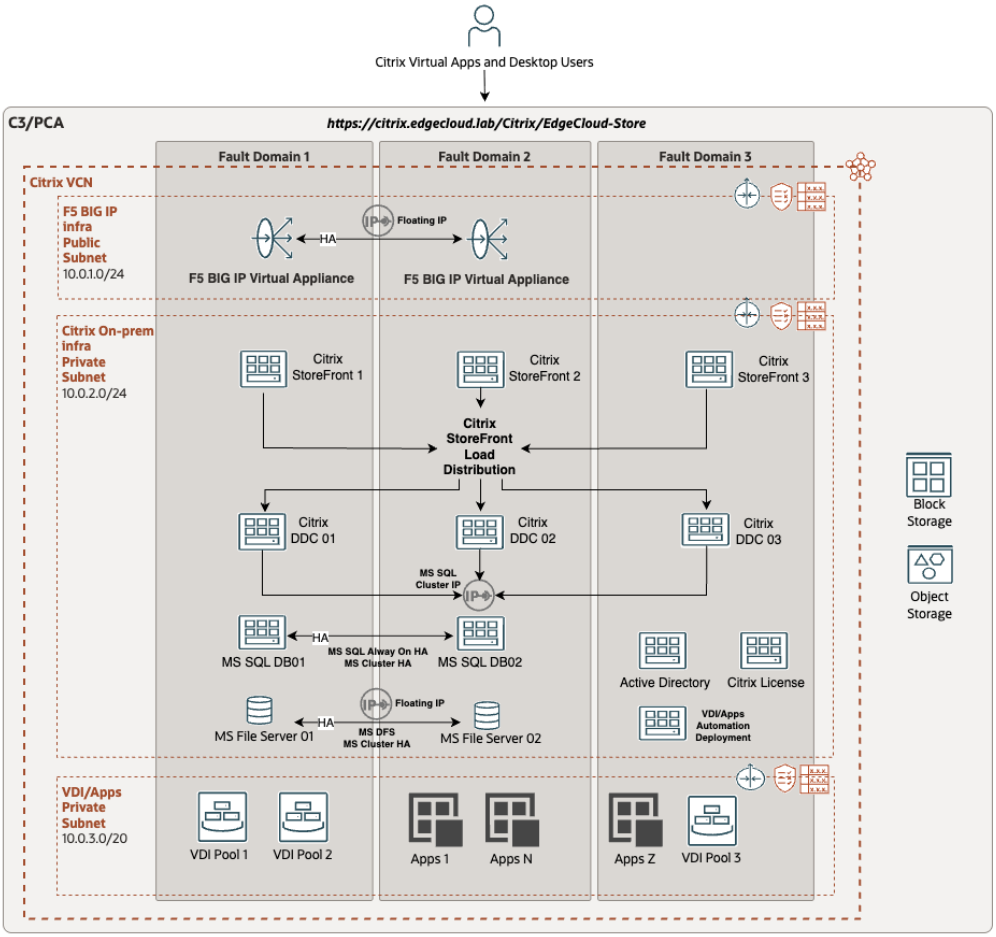


Figure 1. Citrix Virtual Apps and Desktops architecture on Compute Cloud@Customer or Private Cloud Appliance.

Platform Components

- **Infrastructure:** Oracle **Compute Cloud@Customer** or **Private Cloud Appliance**
- **Availability:** 3 Fault Domains (FD1 / FD2 / FD3)
- **Citrix Software:** Citrix StoreFront, Delivery Controller, Citrix License Server

- **Provisioning:** OpenTofu (no MCS)
- **Desktop OS:** Windows Server 2025 (RDP multi-session)
- **Access Layer:** F5 BIG IP Virtual Appliance

NOTE: GPU expansion racks can also be utilized as part of the solution, however via Windows RDP-based multi-session, no vGPU.

Component	Quantity	Placement	Sizing (Each)	Design Rationale
Delivery Controller (DDC)	3	One per Fault Domain	8 OCPUs, 128 GB RAM	One DDC per FD ensures control-plane resilience and eliminate single point of failure.

Component	Quantity	Placement	Sizing (Each)	Design Rationale
StoreFront	3	One per Fault Domain	8 OCPUs, 128 GB RAM	StoreFront servers are deployed behind F5 BIG IP load balance.

Component	Quantity	Architecture	Sizing	Design Rationale
SQL Server	2	Always On / HA	8 OCPUs, 128 GB RAM	Stores site configuration and brokering data. Must be highly available in production environments.

Component	Quantity	Sizing	Design Rationale
Citrix License Server	1	4 OCPUs, 64 GB RAM	Utilized to manager all licenses utilized by Citrix Virtual Apps and Desktops.

Virtual Desktop Layer – Windows Server 2025 or Windows 11

Item	Specification
Operating System	Windows Server 2025 or Windows 11
Session Mode	RDP Multi-Session (Windows 2025) or Individual desktops/sessions
Citrix VDA	Installed manually in base image
Provisioning Model	OpenTofu → VM Deployment → Auto-registration with Citrix

Storage Architecture

Storage Type	Purpose
Block Storage	OS Disks
Shared File Storage (NFS / SMB)	Share Files/Folders
High-IOPS Tier	SQL Databases and Logs

Fault Domain Placement Strategy

Tier	FD1	FD2	FD3
DDC	Deployed	Deployed	Deployed
StoreFront	Deployed	Deployed	Deployed
SQL	Deployed	Deployed	Deployed
Desktop Pools	Deployed	Deployed	Deployed

ORACLE

Network Architecture & IP Addressing Strategy

The proposed network architecture is built on clear separation of tiers, isolating Citrix Virtual Apps and Desktops components, Windows services, and workload desktops into distinct subnets to enhance security, manageability, and performance. Fault Domain–aware placement ensures resiliency at the infrastructure level, while high-availability–friendly IP allocation (including floating IPs and SQL listener addresses) supports seamless failover and service continuity. Secure-by-design segmentation using private subnets and controlled traffic flows reduces the attack surface and enforces least-privilege communication between tiers. This structured layout also simplifies operational troubleshooting by providing predictable routing and clearly defined network boundaries.

This design ensures:

- Fault Domain isolation
- Clean traffic segmentation
- Simplified security policy enforcement
- Future scalability
- Efficient route summarization
- Minimal broadcast domain impact

The network is structured within a **10.0.0.0/16 VCN**, allowing ample growth while maintaining logical subnet boundaries. This provides total capacity of 65,536 IPs which are sufficient address space for:

- Scalable deployments
- Additional infrastructure tiers
- DR or staging environments

Public Infrastructure Subnet

(F5 BIG-IP / External Access Tier)

Parameter	Value
CIDR	10.0.1.0/24
Usable IPs	254
Placement	Across 3 Fault Domains
Gateway	Internet Gateway for internet access

Use:

- F5 BIG-IP Virtual Appliances
- Floating IPs (HA)
- VIPs for external access (HTTPS)

Design Rationale:

- Dedicated subnet isolates external-facing components and access
- Supports HA pair or cluster configuration
- Enables SSL offload and ICA proxy termination
- Clean boundary for firewall inspection policies

Citrix Infrastructure Private Subnet

(Control Plane + Core Services)

Parameter	Value
CIDR	10.0.2.0/24
Usable IPs	254
Access	Private (No Internet Gateway)

Use:

- Delivery Controllers (DDCs)
- StoreFront Servers
- SQL Always On Cluster
- SQL Listener IP
- Microsoft File Servers (DFS)
- Citrix License Server
- Active Directory / DNS
- Automation Deployment Servers

Design Rationale:

- Segregates control-plane services from workload layer
- Minimizes east-west attack surface
- Enables strict NSG-based segmentation
- Supports SQL HA listener IP allocation

DaaS and Application Workload Subnet

(Desktop & App Hosting Tier)

Parameter	Value
CIDR	10.0.3.0/20
Usable IPs	~4,091
Access	Private (No direct Internet access)

Use:

- Windows Server 2025 Multi-Session Hosts
- Windows 11 Desktops
- Windows Instances with Applications for Citrix Apps
- Virtual Applications./Application pools

Network Security Best Practices

- VDI Subnet Must Be Private
- No Internet Gateway attached
- Outbound traffic via:
    - NAT Gateway
    - Firewall appliance
- Use Network Security Groups to enforce micro-segmentation:

Flow	Allowed Ports
VDI to DDC	80, 443, 1494, 2598
DDC to SQL	1433
VDI to SQL	(Restricted)
StoreFront to DDC	80, 443
F5 to StoreFront	443

Routing Design

- Public subnet uses Internet Gateway
- Private subnets use:
  - NAT Gateway for outbound updates
  - Firewall for inspection
- No direct public IPs assigned to VDI hosts.
- SQL, StoreFront, and DDC servers are never exposed externally.

Windows Server Infrastructure for Citrix Virtual Apps and Desktops

Active Directory, Microsoft Windows Failover Cluster, and Microsoft SQL Server Always On High Availability.

This section documents the required Microsoft Windows Server infrastructure supporting Citrix Virtual Apps and Desktops on Compute Cloud@Customer (C3) or Private Cloud Appliance (PCA). The design provides a highly available, resilient, and production-ready foundation for Citrix control-plane and database services.

The architecture aligns with Microsoft and Citrix best practices and is designed to support enterprise-grade production workloads.

Key design objectives:

- Eliminate single points of failure
- Provide automatic failover with zero data loss
- Maintain a single stable SQL connection endpoint for Citrix
- Align with Microsoft and Citrix best practices
- Enable clean Day-2 operations and troubleshooting

The figure listed below shows the recommended Microsoft SQL architecture for Citrix Virtual Apps and Desktops on Compute Cloud@Customer or Private Cloud Appliance. This architecture provides high availability for Citrix infrastructure databases using SQL Server Always On Availability Groups with automatic failover capabilities.

The Microsoft SQL HA architecture consists of:

- Two Windows 2025 Servers nodes (`citrix-db-01, citrix-db-02`). Ensure to have two Windows Servers 2025 instances deployed on C3
- Windows Server Failover Cluster (WSFC)
- SQL Server Always On Availability Group
- SQL Availability Group Listener
- Dedicated storage for data, logs, TempDB, and backups
- All components are deployed within the same network subnet to simplify failover and DNS behavior.

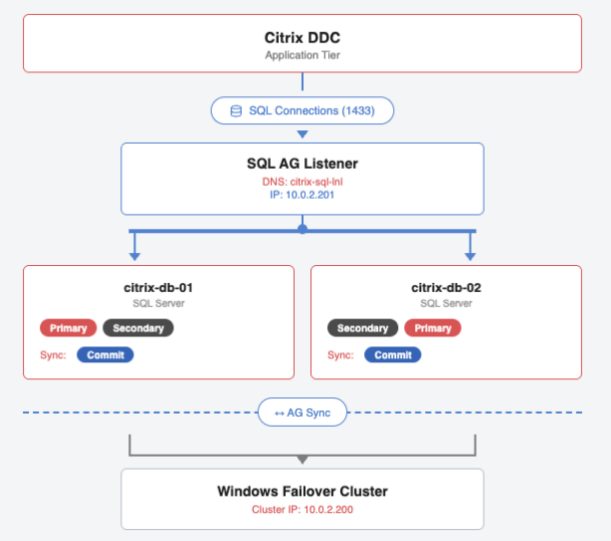


Figure 2. Microsoft SQL Server Cluster Always On architecture for Citrix Virtual Apps and Desktops on Compute Cloud@Customer or Private Cloud Appliance.

Microsoft Active Directory

Active Directory Requirements: Citrix Virtual Apps and Desktops is tightly integrated with Microsoft Active Directory (AD). A properly designed AD foundation is a prerequisite before deploying Delivery Controllers, SQL clustering, or virtual desktops. The AD environment must include:

- A Windows Active Directory Domain (e.g., edgecloud.lab)
- At least one Domain Controllers
- Integrated DNS (AD-integrated DNS recommended)
- Time synchronization via domain hierarchy
- Citrix Organizational Unit (OU): A dedicated Organizational Unit (OU) must be created in Active Directory for Citrix resources. Example: OU=citrix,DC=edgecloud,DC=lab This OU will host:
    - Citrix Delivery Controllers
    - StoreFront servers
    - SQL Servers
    - VDI machines
    - Service accounts (if desired)
- **Citrix Service Account Requirements:** A dedicated Citrix administrative service account must be created with permissions to:
    - Join machines to the domain
    - Add/remove computer objects
    - Install services
    - Modify machine accounts
    - Manage Citrix components
    - Minimum recommended permissions:
        - Domain User
        - Delegated “Create/Delete Computer Objects” rights in Citrix OU
        - Local Administrator on Citrix servers
        - SQL sysadmin (for initial deployment)
    - **This account is used during:**
        - Delivery Controller installation

- VDA registration
- Virtual Desktop provisioning
- SQL configuration

DNS Requirements:

- AD-integrated
- Available from all Fault Domains
- Supporting dynamic updates
- Resolving:
  - SQL Listener FQDN
  - DDC FQDNs
  - StoreFront FQDN
  - F5 VIPs

NOTE: SQL Availability Group Listeners rely on proper DNS behavior for failover transparency.

NOTE: For Active Directory best practices and recommendations for Citrix Virtual Apps and Desktops, refers to Citrix official documentation:

- https://docs.citrix.com/en-us/citrix-virtual-apps-desktops/2203-ltsr/technical-overview/active-directory

SQL Server High Availability Architecture and Windows Server Failover Cluster (WSFC)

To ensure database-level resiliency for Citrix infrastructure databases, including the Site, Logging, and Monitoring databases, Microsoft SQL Server Always On Availability Groups are deployed on top of Windows Server Failover Clustering (WSFC). This design eliminates single points of failure by providing automatic failover with synchronous commit to prevent data loss, while maintaining a single, stable SQL Listener endpoint for Citrix connectivity. This architecture aligns with both Microsoft and Citrix best practices and supports streamlined Day-2 operations, simplified troubleshooting, and enterprise-grade high availability for production workloads.

The recommended SQL HA design for Citrix Virtual Apps and Desktops includes:

- Two Windows Server 2025 instances:
  - citrix-db-01
  - citrix-db-02
- Windows Server Failover Cluster (WSFC)
- SQL Server Always On Availability Group
- SQL Availability Group Listener (single connection endpoint)
- Dedicated block storage on Compute Cloud@Customer or Private Cloud Appliance for each Citrix DB instance for:
  - Data files
  - Transaction logs
  - TempDB
  - Backups
- All components deployed within the same subnet to simplify failover and DNS behavior

This architecture provides database-level HA while maintaining storage independence (no shared disks required).

Role of Windows Server Failover Clustering (WSFC)

Windows Server Failover Clustering provides the clustering foundation required for SQL Server Always On.
Important: SQL Availability Groups cannot function without WSFC, even though databases are not stored on shared disks.

WSFC is responsible for:

- Monitoring node health
- Managing SQL role ownership
- Hosting cluster-managed IP resources
- Maintaining quorum
- Enabling automatic failover
- Coordinating SQL Listener resource movement

NOTE: For Windows Server Failover Clustering (WSFC) best practices and recommendations, refers to Microsoft official documentation:

- https://learn.microsoft.com/en-us/sql/sql-server/failover-clusters/windows/windows-server-failover-clustering-wsfc-with-sql-server?view=sql-server-ver15

NOTE: For Microsoft SQL Server best practices and recommendations for Citrix, refer to Citrix and Microsoft official documentation:

- https://learn.microsoft.com/en-us/sql/database-engine/availability-groups/windows/overview-of-always-on-availability-groups-sql-server?view=sql-server-ver17
- https://docs.citrix.com/en-us/citrix-virtual-apps-desktops/2203-ltsr/technical-overview/databases.html

Deploying Citrix Virtual Apps and Desktops Infrastructure

This section describes the end-to-end deployment of Citrix Virtual Apps and Desktops infrastructure on Compute Cloud@Customer or Private Cloud Appliance to host Virtual Desktops or Applications in production and at scale. It includes the configuration and deployment of Citrix Delivery Controllers, Site Creation, and Operational Validation. This solution is fully integrated with a highly available SQL backend using SQL Server Always On Availability Groups (AG) previously discussed. The deployment follows this strict sequence, which is required to ensure database consistency, supported Citrix behavior, and production readiness:

- Delivery Controller installation
- Citrix Site creation
- Operational and failover validation

Citrix Delivery Controller Installation

Install the Citrix Delivery Controller (DDC) components on Windows Server 2025 and prepare the system for Site creation and SQL integration.

Prerequisites:

- Windows Server 2025 fully patched
- Domain-joined servers
- SQL Server Always On AG already deployed and healthy at the Microsoft SQL servers
- Service accounts created:
  - Citrix Site Database Access
  - Citrix Logging Database Access
  - Citrix Monitoring Database Access
- Firewall ports opened:
  - TCP 1433 (SQL)
  - TCP 80/443 (Citrix services)

- Required Citrix internal ports

Installation Steps

1. Mount the Citrix Virtual Apps and Desktops ISO on the Windows Server 2025 Delivery Controller servers (DDCs). For high availability of Citrix Delivery Controllers on Compute Cloud@Customer or Private Cloud Appliance, deploy at least two DDCs. In our architecture, we are using all three available fault domains on Compute Cloud@Customer or Private Cloud Appliance. So, we have three DDCs available for the solution.
2. Launch AutoSelect.exe, click start on Virtual Apps and Desktops

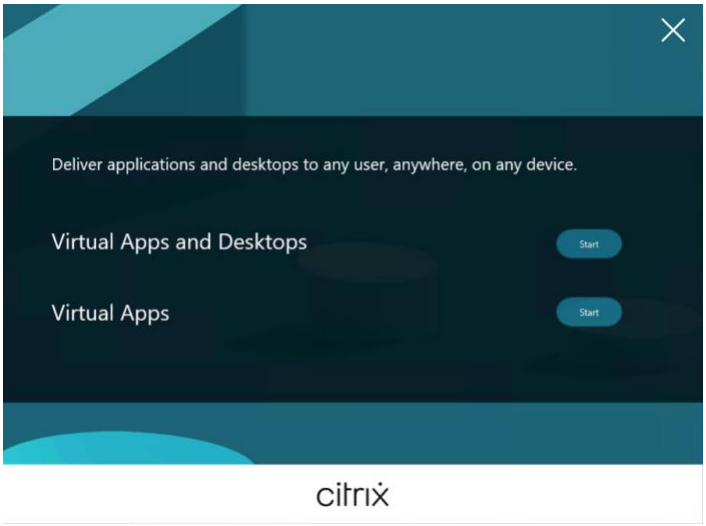


Figure 3. Citrix Virtual Apps and Desktops installation on Compute Cloud@Customer or Private Cloud Appliance.

3. Select **Delivery Controller**.

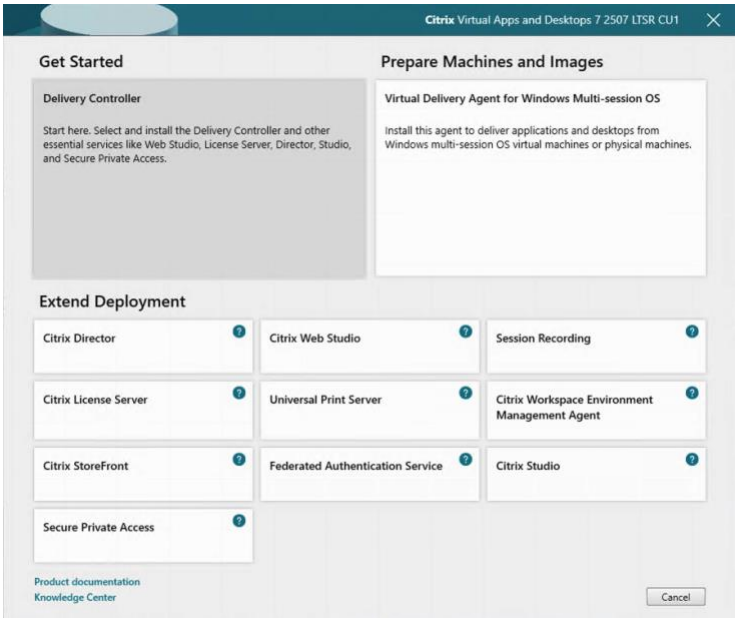


Figure 4. Citrix Delivery Controller servers' installation on Compute Cloud@Customer or Private Cloud Appliance.

4. Accept the Software License Agreement. Click next
5. Select Delivery Controller, Web Studio, and Director. Click next.
   - **Delivery Controller:** Required. This is the brain of Citrix (brokering, policies, registrations).
   - **Web Studio:** Recommended. This is the modern web-based replacement for MMC Studio and is the forward-looking management UI.
   - **Citrix Studio:** The MMC Studio for Windows.
   - **Director:** Recommended. Used for monitoring, troubleshooting, and operational visibility.

NOTE: In our architecture, we install Web Studio, Studio and Director in all three Delivery Controllers servers. Like:

- Citrix-DDC-01: Delivery Controller + Web Studio + Studio + Director
- Citrix-DDC-02: Delivery Controller + Web Studio + Studio + Director
- Citrix-DDC-03: Delivery Controller + Web Studio + Studio + Director

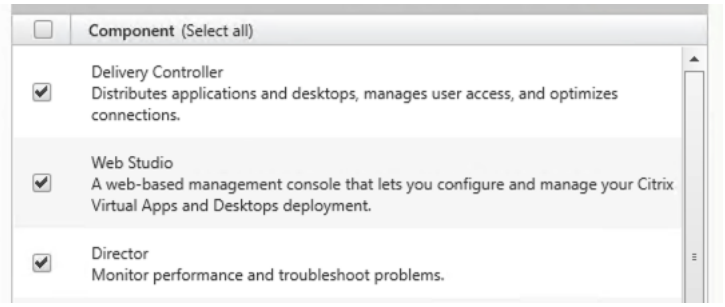


Figure 5. Citrix Delivery Controller servers (DDCs) installation components selection.

6. Delivery Controller Registration (Initial Site Creation). During the installation of the first Citrix Delivery Controller, the installer prompts for the list of Delivery Controllers authorized to manage and monitor the Site using Citrix Web Studio, Studio, and Director, on this case the local Delivery Controller FQDN listed is the `citrix-ddc-01.edgecloud.lab`. **Click next.**

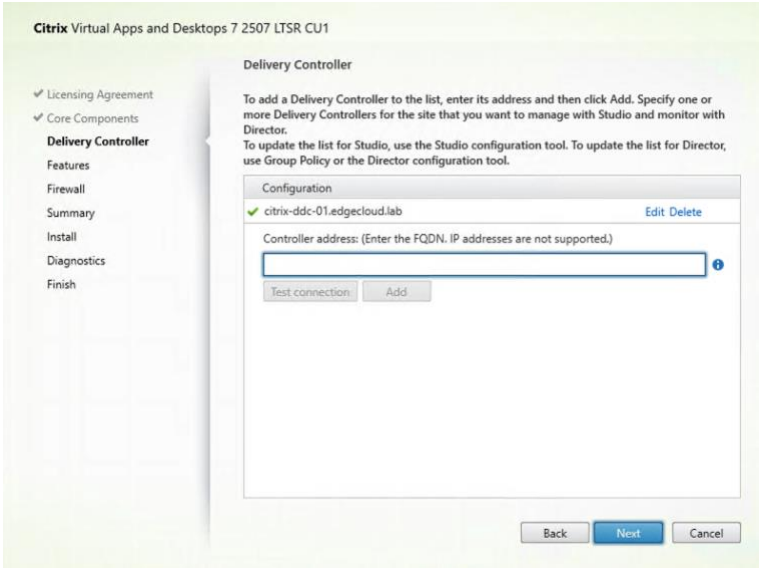


Figure 6. Citrix Delivery Controller servers (DDCs) FQDN configuration.

NOTE: Additional Delivery Controllers are not added at this stage, as they are not yet installed or registered. Additional Delivery Controllers (citrix-ddc-02 and citrix-ddc-03) are added after Site creation by installing the Delivery Controller role on each server and selecting Join existing Site during installation. These controllers automatically register with the Site and become available for brokering and high availability.

7. During the Delivery Controller installation, several optional features are presented, and some are enabled by default.

    - Microsoft SQL Server Express is enabled by default in the installer and must be explicitly unchecked. This option is not used in this architecture, as we rely on a dedicated Microsoft SQL Server Always On Availability Group for all Citrix site, logging, and monitoring databases.
    - Windows Remote Assistance may be optionally installed to support Director-based session shadowing and operational troubleshooting workflows.

    This configuration ensures:

    - A clean separation between the Citrix control plane and the database tier
    - Full high availability and resiliency for Citrix site databases
    - Alignment with Citrix enterprise and production deployment best practices

So, unchecked the Microsoft Database option, select the Install Windows Remote Assistance option, then click next.

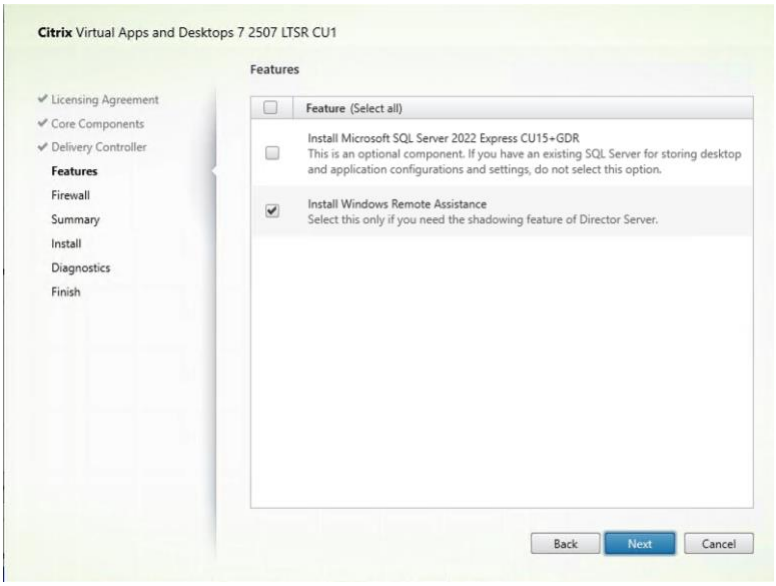


Figure 7. Citrix Delivery Controller servers (DDCs) Features installation.

8. During the Delivery Controller installation, the Citrix installer prompts for firewall configuration. The Automatically option is selected to allow the installer to create the required Windows Firewall rules for Citrix components, including the Delivery Controller, Web Studio, Studio, and Director. The installer opens the following default ports:

- Delivery Controller: TCP 80, 89, 443
- Web Studio: TCP 443
- Director: TCP 80, 443

These rules are created regardless of the current Windows Firewall state, ensuring consistent behavior if the firewall is enabled later. Select Automatically option, click next.

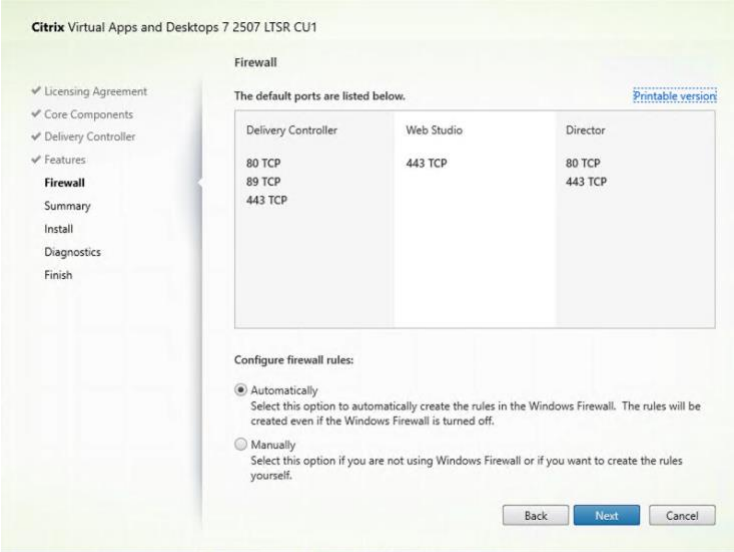


Figure 8. Citrix Delivery Controller servers (DDCs) Firewall Configuration.

9. Delivery Controller installation, final validation and execution. Before proceeding with the installation, the Summary screen is reviewed to validate all selected components and prerequisites. The following configuration is confirmed:

- Default Citrix installation directory is used
- Required Windows and IIS prerequisites are automatically installed
- Core components include:
    - Delivery Controller
    - Web Studio
    - Studio
    - Director
- The initial Delivery Controller (DDC01) is the only controller listed at this stage
- Windows Remote Assistance is enabled to support Director-based troubleshooting

NOTE: Although the deployment uses a dedicated Microsoft SQL Server Always On Availability Group for Citrix databases, the installer deploys **Local Host Cache (LocalDB)** on each Delivery Controller. This is expected and provides site resiliency in the event of a temporary SQL outage.

10. Once validated, the installation is executed to complete the Delivery Controller setup. Click next.

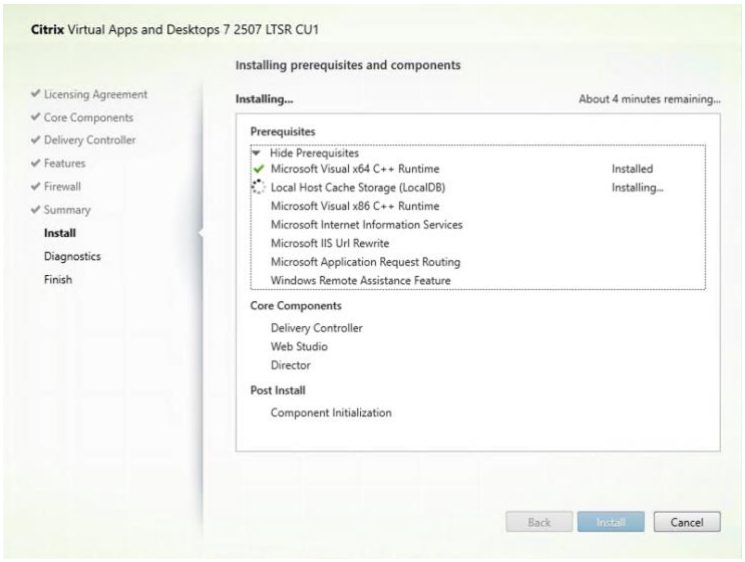


Figure 9. Citrix Delivery Controller servers (DDCs) installation progress.

11. Diagnostics and Telemetry Configuration. During the final steps of the Delivery Controller installation, the Citrix diagnostics and telemetry (Call Home) option is presented. Enabling or disabling the Collect diagnostic information option is a customer decision, based on internal security, compliance, and operational policies. When enabled, Citrix Call Home automatically transmits system configuration details, performance metrics, and diagnostic data to Citrix Cloud to assist with proactive support and troubleshooting. When disabled, no diagnostic data is sent externally. This flexibility allows you to align the deployment with your organizational requirements, including environments with restricted outbound connectivity or regulated data policies. Diagnostic data can still be collected manually, and the Call Home feature can be enabled or disabled at any time after installation if support requirements change.

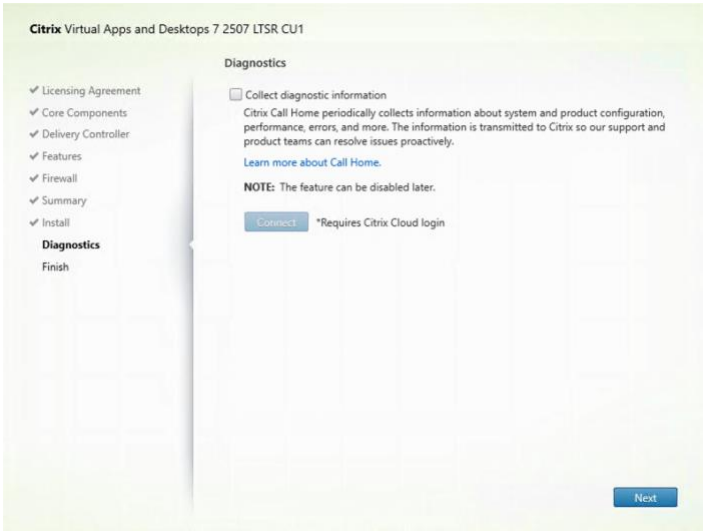


Figure 10. Citrix Delivery Controller servers (DDCs) Diagnostics configuration.

12. Delivery Controller Installation Completion. The Delivery Controller installation completes successfully with all required prerequisites, core components, and post-install initialization verified. The following components are confirmed as installed on the first Delivery Controller (citrix-ddc-01):
    - Delivery Controller

- Web Studio
- Director
- Local Host Cache (LocalDB)

All required Windows and IIS dependencies are installed automatically by the installer. Component initialization completes successfully, indicating the Delivery Controller is ready for Citrix Site creation. Licensing configuration is deferred to a later stage and will be integrated with a dedicated Citrix License Server, in alignment with the overall architecture. Upon completion, Citrix Site Manager will be launched to proceed with Site creation and database configuration. Select Launch Citrix Site Manager, click Finish.

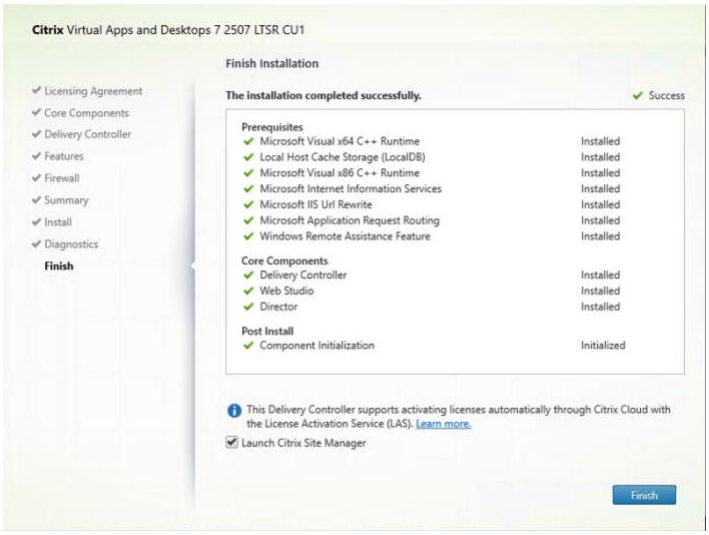


Figure 11. Citrix Delivery Controller servers (DDCs) installation completed successfully.

While still on the installer:

- Under Extend Deployment, double click Citrix Studio, then click install, and finish to complete the Studio installation.

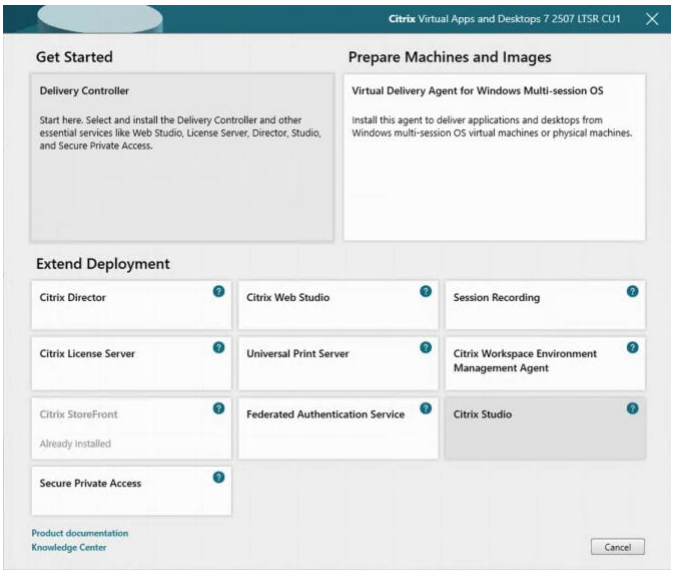


Figure 12. Installation of Citrix Studio.

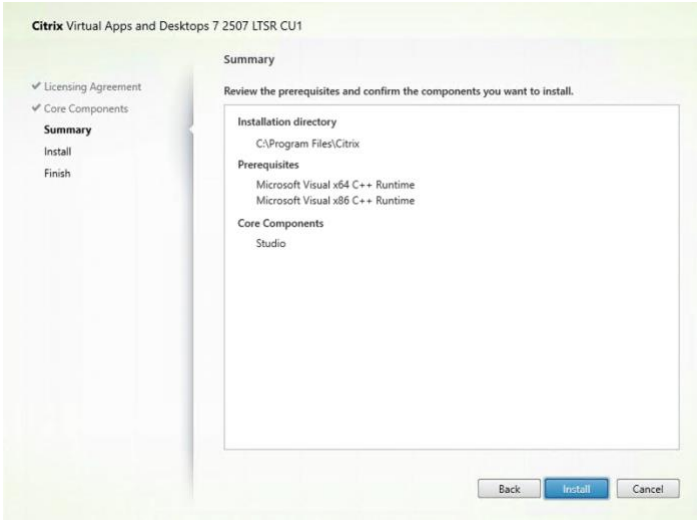


Figure 13. Citrix Studio installation summary.

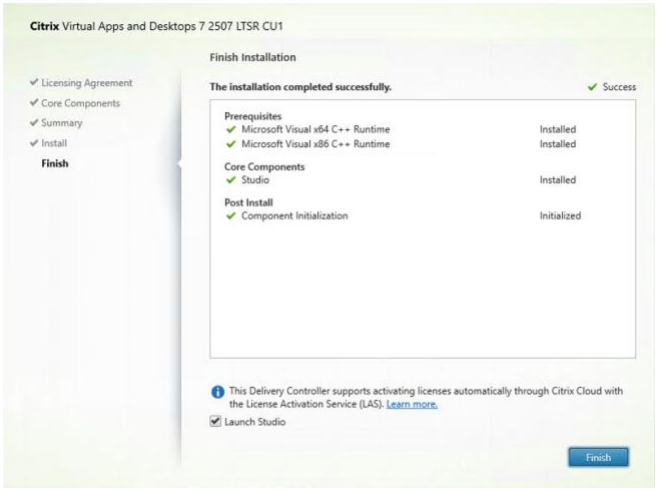


Figure 14. Citrix Studio installation successfully completed.

13. **Citrix Site Creation**. After completing the Delivery Controller installation on the first controller (citrix-ddc-01), Citrix Site Manager is launched to initialize the Citrix Site. The option "Deliver applications and desktops to your users" is selected to create a new Citrix Site. This step establishes the initial site configuration, creates the Citrix databases, and registers the first Delivery Controller as the authoritative control plane node. Click on Deliver applications and desktops to your users.

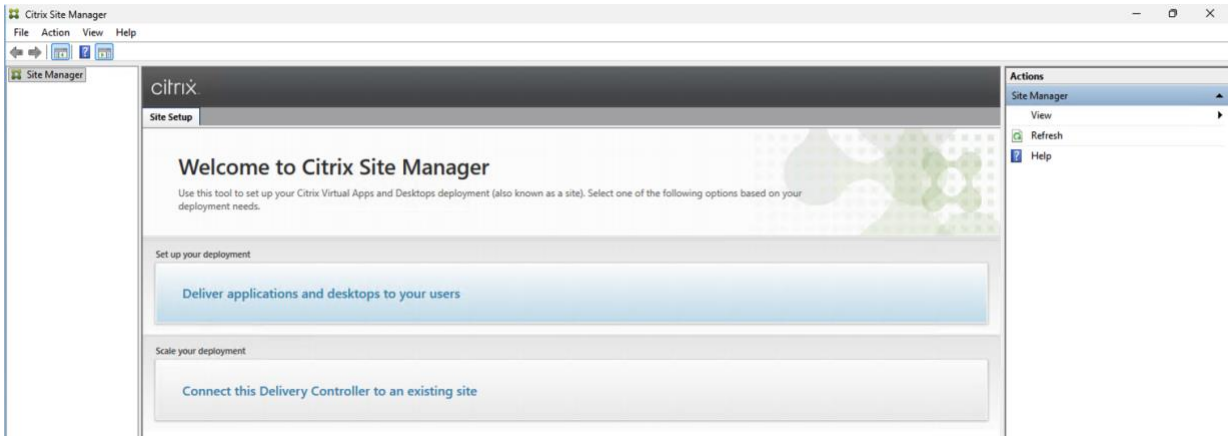


Figure 15. Citrix Site Manager – Site creation.

NOTE: The option "Connect this Delivery Controller to an existing site" is reserved for subsequent Delivery Controllers and is not used during initial Site creation.

14. During Site creation, a unique and descriptive Site name is defined. The Site name identifies the Citrix control plane configuration and is stored within the Citrix site databases. A consistent naming convention is recommended to reflect the environment or deployment context (for example, production or non-production). Once created, the Site name is not intended to be changed and is used across Citrix management tools, logging, and automation workflows. Enter the site name and click next.

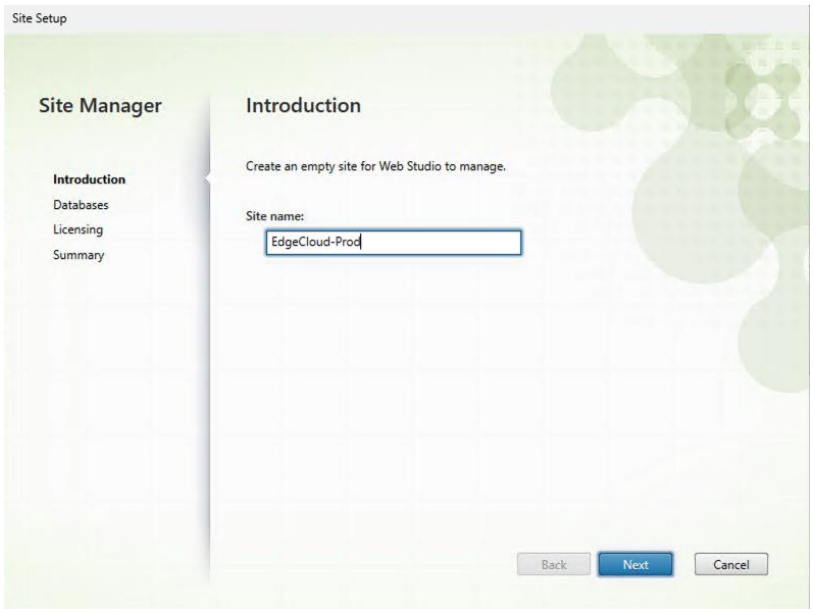


Figure 16. Citrix Site Manager – Site name configuration.

15. **Citrix Site Database Configuration (SQL Always On Availability Group).** During Citrix Site creation, the Create and set up databases from Site Manager option is selected. This allows Citrix to automatically create and initialize the Site, Logging, and Monitoring databases using supported schemas and permissions.

NOTE: Windows Authentication is used for database access, and the deployment targets a dedicated Microsoft SQL Server Always On Availability Group to provide database high availability and automatic failover.

The following databases are created as part of Site initialization:

- Citrix Site database: `CitrixEdgeCloud-ProdSite`
- Citrix Logging database: `CitrixEdgeCloud-ProdLogging`
- Citrix Monitoring database: `CitrixEdgeCloud-ProdMonitoring`

All databases are initially created and accessed through the SQL Always On Availability Group listener:

`citrix-sql-list.edgecloud.lab`

The listener resolves to the active primary replica (for example, 10.0.2.201) and abstracts database connectivity from individual SQL nodes. This ensures that Citrix Delivery Controllers remain unaware of replica changes and continue operating seamlessly during SQL failover events.

Once database creation and Site validation complete successfully, the databases remain managed within the Availability Group, enabling continuous availability and resilience for the Citrix control plane.

IMPORTANT: During Citrix Site creation, the Site Manager process runs under the following Windows account: `edgecloud\citrix` This account must be granted appropriate permissions on the SQL Server Always On Availability Group to allow Citrix to create and initialize the required site databases.

On the active SQL Server primary node (`citrix-db-01`), open SQL Server Management Studio (SSMS) and connect to the Availability Group listener: `citrix-sql-list.edgecloud.lab`

Execute the following SQL commands to create the login (if it does not already exist) and assign the required server roles:

```
USE master;
GO

-- Create the login if it does not already exist
IF NOT EXISTS (
    SELECT 1
    FROM sys.server_principals
    WHERE name = N'edgecloud\citrix'
)
BEGIN
    CREATE LOGIN [edgecloud\citrix] FROM WINDOWS;
END
GO

-- Grant required server roles
ALTER SERVER ROLE dbcreator ADD MEMBER [edgecloud\citrix];
ALTER SERVER ROLE securityadmin ADD MEMBER [edgecloud\citrix];
GO
```

The following SQL Server roles are required during Citrix Site creation:

Role	Purpose
dbcreator	Allows creation of the Citrix Site, Logging, and Monitoring databases
securityadmin	Allows creation of database users, roles, and permissions within the Citrix databases

This permission model is fully supported and documented by Citrix, including deployments that use SQL Server Always On Availability Groups.

After granting the SQL permissions:

- Return to Citrix Site Manager on `citrix-ddc-01`.
- Navigate back to the database validation step.
- Select Create and setup databases from Site Manager
- For the database authentication, select Microsoft on-premises SQL Server and Windows Authentication as authentication mode
- Provide the database details as previous mentioned and enter the listened address of you SQL Always On Availability Group database.
- Click Next

Citrix Site Manager will automatically:

- Create the `CitrixEdgeCloud-ProdSite` database
- Create the `CitrixEdgeCloud-ProdLogging` database
- Create the `CitrixEdgeCloud-ProdMonitoring` database
- Apply the required schemas and permissions
- Proceed to the next step of Site configuration

No further credential prompts should appear. Successful configuration is confirmed when:

- All three Citrix databases validate successfully
- No additional permission prompts are displayed
- The workflow proceeds to the Licensing configuration screen
- The Citrix Site is successfully created and initialized

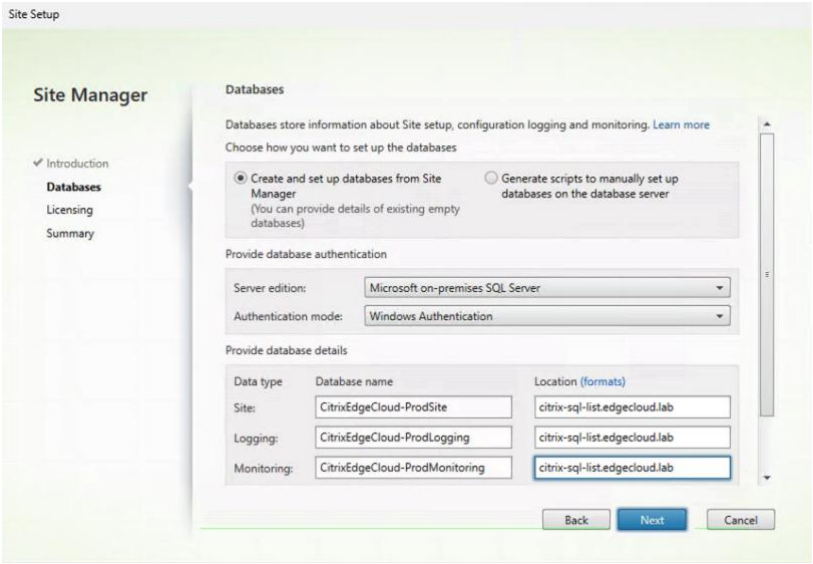


Figure 17. Citrix Site Manager – Citrix database configuration.

16. The licensing configuration step is presented. At this stage, the Citrix Site requires a licensing mode selection but does not require an active license server to proceed. For this deployment, the 30-day trial license option is selected.

NOTE: Customers are recommended to apply their full production Citrix licenses after Site creation and validation. Once the Citrix License Server is available, the Site must be updated to reference the production licensing infrastructure.

- License Server fully qualified domain name (FQDN)
- License Server ports (default: `27000 and 7279`)
- Applicable Citrix Virtual Apps and Desktops licenses

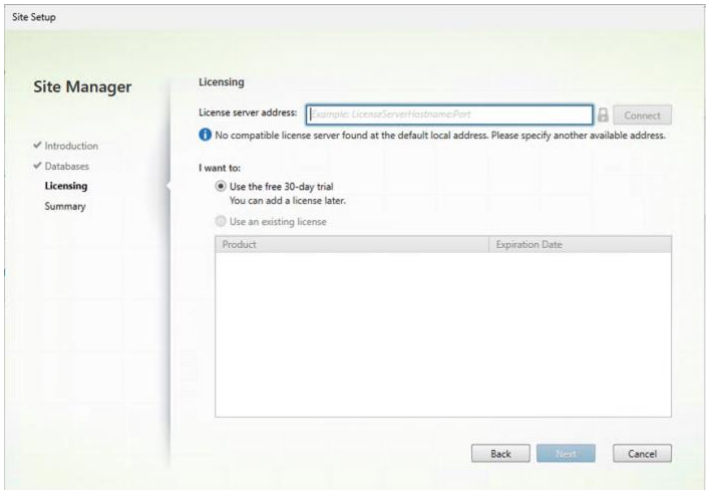


Figure 18. Citrix Site Manager – Citrix licensing configuration.

17. Once confirmed the license, click next, and finish. The installation will complete the final steps. If complete with success, the new `EdgeCloud-Prod` will be available.

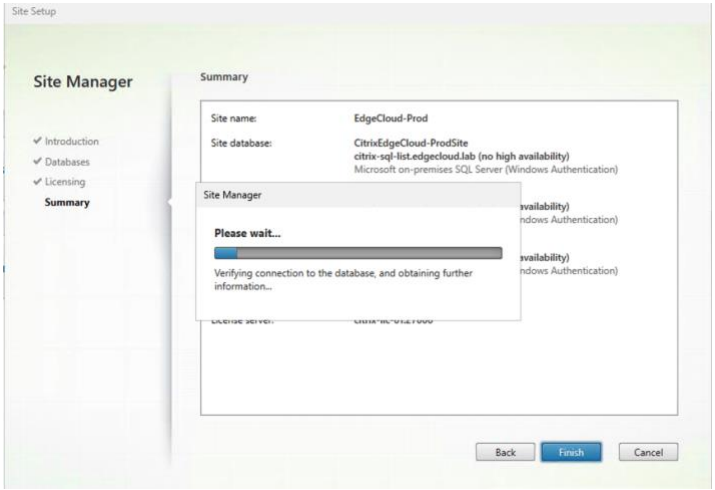


Figure 19. Citrix Site Manager – Site configuration final steps.

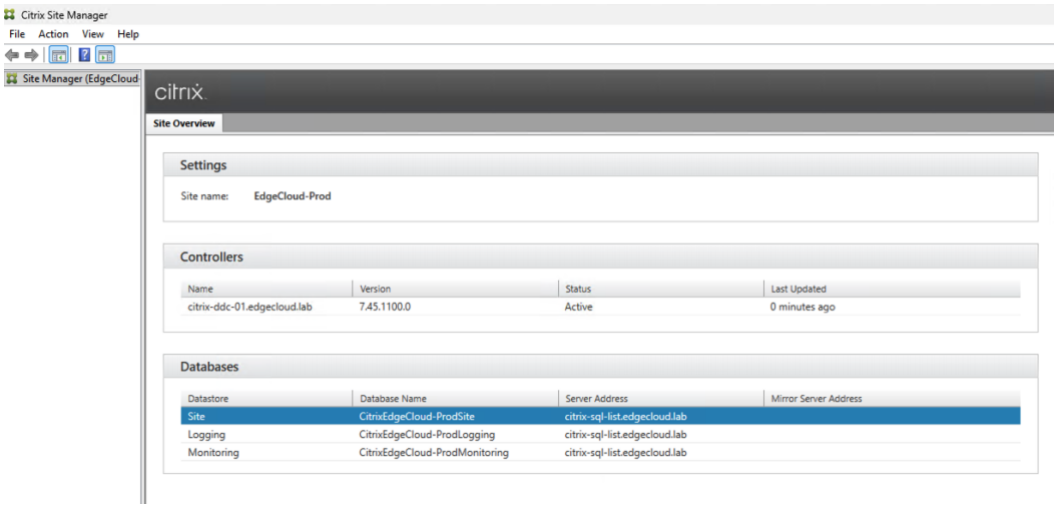


Figure 20. Citrix Site Manager – Site configuration successfully completed.

Now that your first DDC (`citrix-ddc-01`) is operational and connected to your SQL Always On database, adding the additional DDCs is straightforward. Here's the step-by-step process:

18. Adding additional Citrix Delivery Controllers: Now that the first DDC (`citrix-ddc-01`) is operational and connected to the Microsoft SQL Always On database, adding the additional DDCs is straightforward. Here's the step-by-step process:

    - Ensure `citrix-ddc-02` and `citrix-ddc-03` meet the same requirements as `citrix-ddc-01`
    - Verify network connectivity to `citrix-sql-list.edgecloud.lab`
    - Confirm proper DNS resolution for all SQL nodes
    - Ensure Windows Firewall rules allow Citrix service communication
    - Mount the Citrix installation media or access the installation files
    - Run the Citrix Virtual Apps and Desktops installer
    - Select only Delivery Controller as the component to install. Click next
    - Uncheck Install Microsoft SQL server. Click next
    - Select “Automatically” option for Firewall rules installation section. Click next, then click install
    - Complete the installation wizard
    - Restart the server when prompted

19. Join `citrix-02-ddc` to the existing site

After the reboot:

- Launch Citrix Site Manager on `citrix-ddc-02`
- You'll be prompted with the site configuration wizard
- Select "Join an existing site" (not "Create a new site")
- Enter the name of the primary DDC configure, which on this case, `citrix-ddc-01`. Click ok

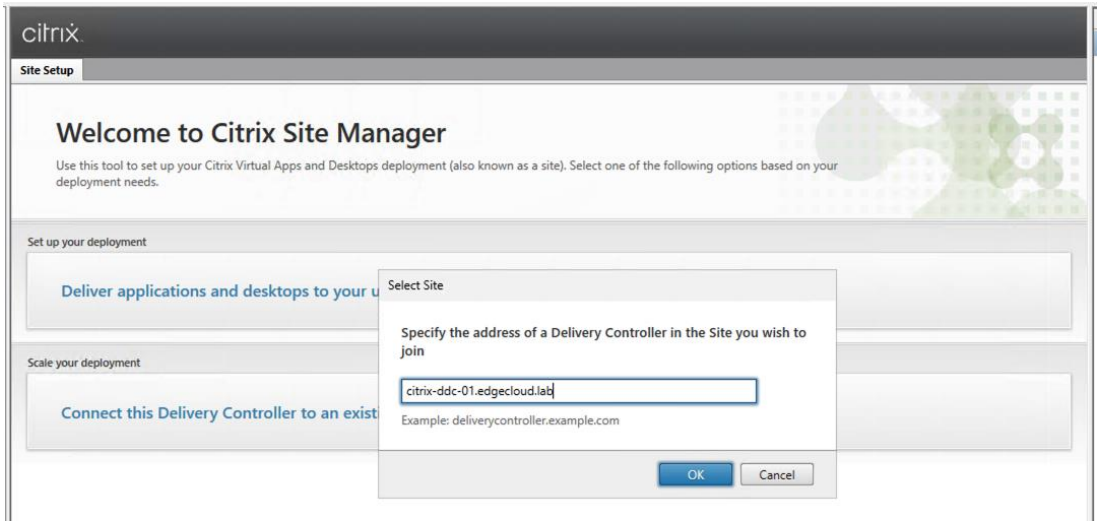


Figure 21. Citrix Site Manager – Adding additional Citrix Delivery Controllers.

- On the next screen, click Yes to let Site Manager to update the database automatically, then the second DDC, `citric-ddc-02` will be added to the list of controllers:

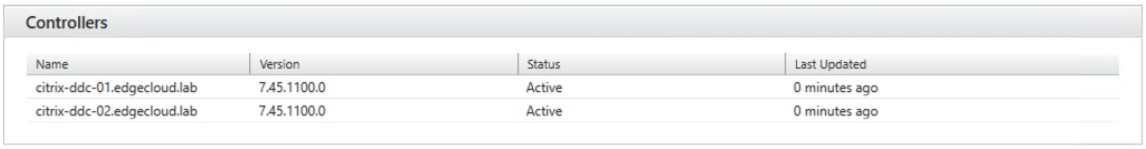


Figure 22. Citrix Site Manager – Additional Citrix Delivery Controllers.

20. Follow the exact same process for the DDC node 3, `citrix-ddc-03`. Once successfully complete, all three DDCs, will be listed under controller section of the Citrix Site Manager:

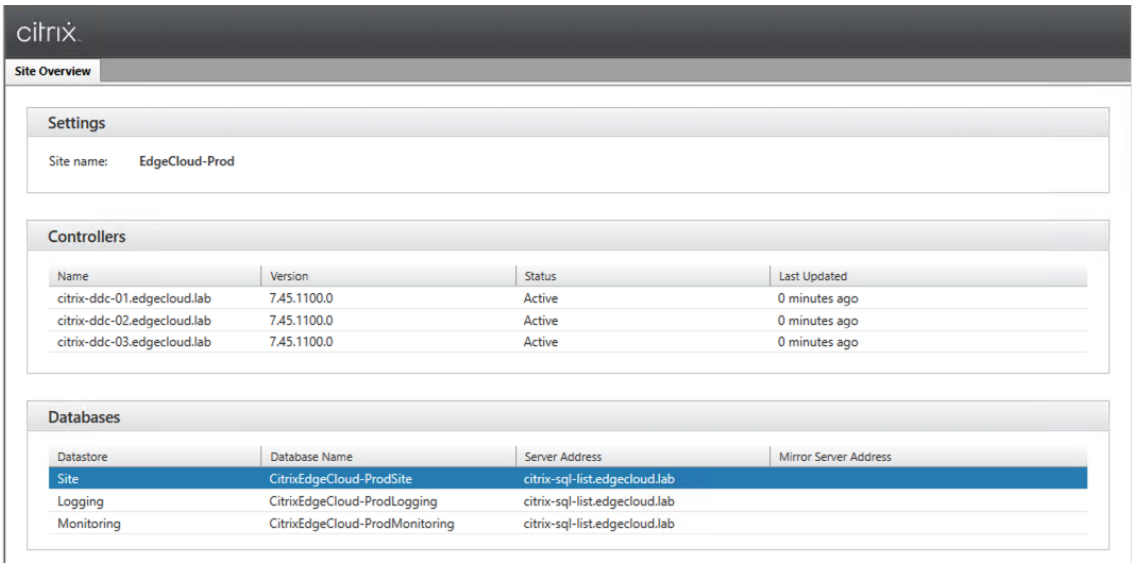


Figure 23. Citrix Site Manager – List of all Citrix Delivery Controllers successfully added to Citrix Virtual Apps and Desktops.

Configuring Certificates for Secure StoreFront to Delivery Controller Communication

In this architecture, Citrix StoreFront communicates directly with each Citrix Delivery Controller (DDC) using individual fully qualified domain names (FQDNs). Citrix Virtual Apps and Desktops natively support multiple Delivery Controllers and performs controller selection and failover internally.

You need to enable secure HTTPS (TCP/443) communication between StoreFront servers and each Delivery Controller using TLS. For production deployments, customers should use certificates issued by a trusted enterprise or public certificate authority and follow Citrix security best practices. Refer to Citrix official documentation to correctly deploy TLS on Delivery Controllers and for security best practices.

- https://docs.citrix.com/en-us/citrix-virtual-apps-desktops/secure/best-practices
- https://docs.citrix.com/en-us/citrix-virtual-apps-desktops/secure/tls-ddc

Registering Citrix StoreFront Servers with the Citrix Delivery Controllers

As part of the Citrix Virtual Apps and Desktops configuration, the StoreFront servers must be explicitly registered with the Citrix Site to establish a trusted control-plane relationship. This is performed from Citrix Web Studio on any Delivery Controller, as the configuration is stored centrally in the Site database and automatically shared across all controllers. In Citrix Web Studio, navigate to Configuration, StoreFront and select Add StoreFront Server. Add each StoreFront server individually using its fully qualified domain name (FQDN) and HTTPS URL (for example, `https://citrix-store-01.edgecloud.lab`). Repeat this process for all StoreFront nodes in the deployment. Once added, the StoreFront servers appear in the StoreFront configuration view and are trusted by all Delivery Controllers in the Site. No additional configuration or propagation is required on other Delivery Controllers, as they automatically inherit the updated StoreFront trust configuration from the shared Site database.

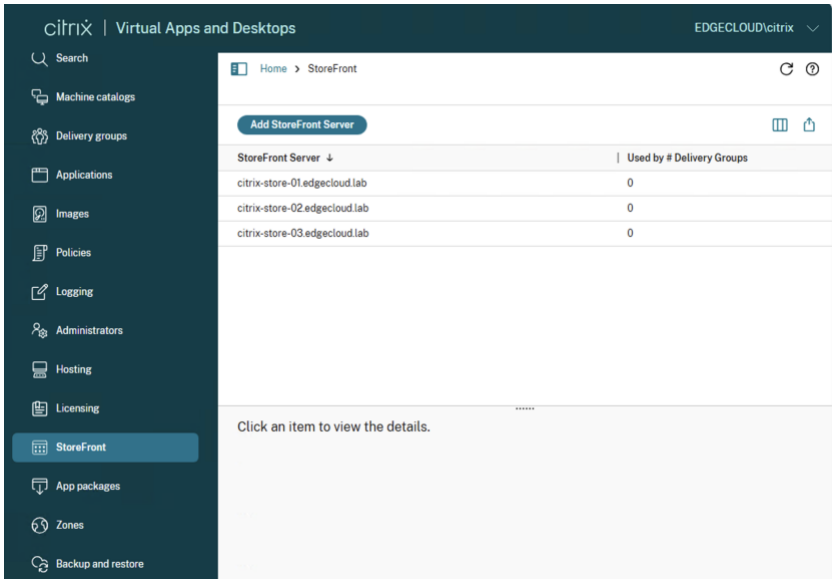


Figure 24. Registering Citrix StoreFront servers with Citrix Delivery Controllers.

Citrix StoreFront Deployment

In this architecture, Citrix StoreFront is deployed on dedicated servers and serves as the user-facing access layer for Citrix Virtual Apps and Desktops. StoreFront handles user authentication, resource enumeration, and session launch requests, and is accessed through F5 BIG IP Virtual Appliance. Citrix recommends that you select the option *Servers are load balanced*. This causes StoreFront to distribute the load between all delivery controllers by selecting a server from the list at random during each launch. If this option is not selected, then the servers list is treated as a failover list in priority order. In this case, 100% of launches occur on the first active Delivery Controller in the list. If that server goes offline, 100% of launches occur using the second in the list, and so on. This ensures high availability and resiliency for brokering operations. The overall request flow is as follows:

- F5 BIG IP forwards user traffic to the StoreFront servers
- StoreFront communicates with the Citrix Delivery Controllers through its own load balanced algorithm
- Delivery Controllers broker sessions to the appropriate Virtual Delivery Agents (VDAs)

Before starting the Citrix StoreFront installation, ensure the following prerequisites are met:

- Windows Server 2025 is fully patched and joined to Active Directory
- Three Citrix Delivery Controllers are installed and operational, one per Compute Cloud@Customer or Private Cloud Appliance Fault Domain as listed in our architecture diagram
- Network connectivity is validated between Citrix StoreFront and Delivery Controllers

Installing Citrix StoreFront

- Log in to the Windows Server 2025 system designated for Citrix StoreFront servers
- Mount the Citrix Virtual Apps and Desktops ISO
- Launch AutoSelect.exe
- From the Get Started screen, do not select Delivery Controller
- Under Extend Deployment, click Citrix StoreFront

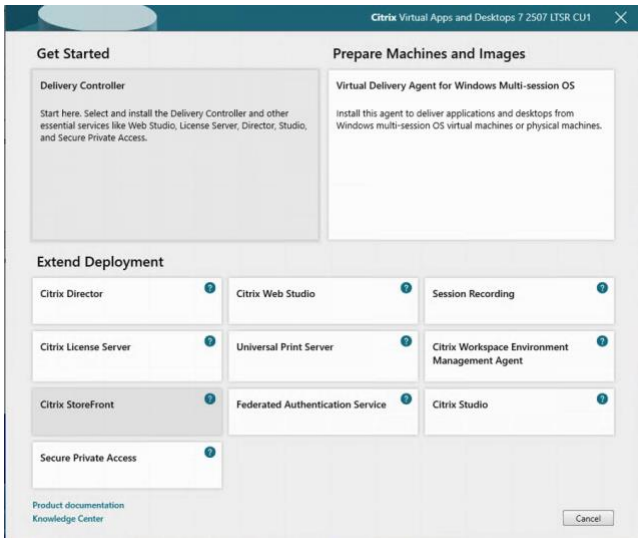


Figure 25. Installing Citrix StoreFront.

- Allow the installer to install all required prerequisites: .NET components and IIS roles and features. Click install
- After the installation completes, click finish, then reboot when prompted

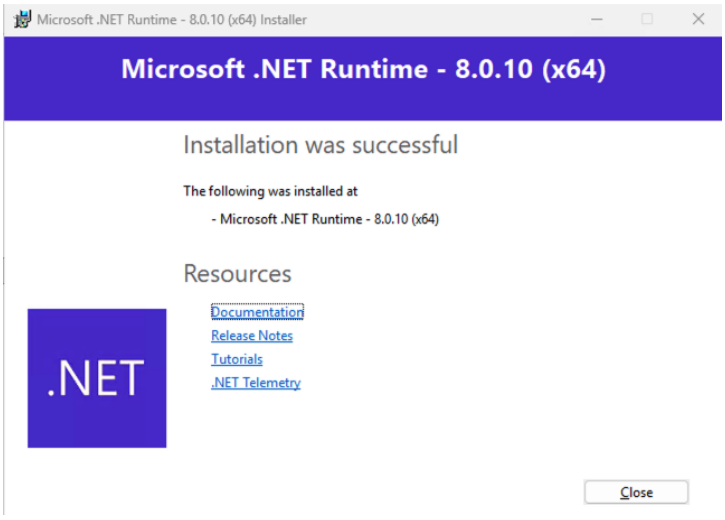


Figure 26. Installation of required prerequisites.

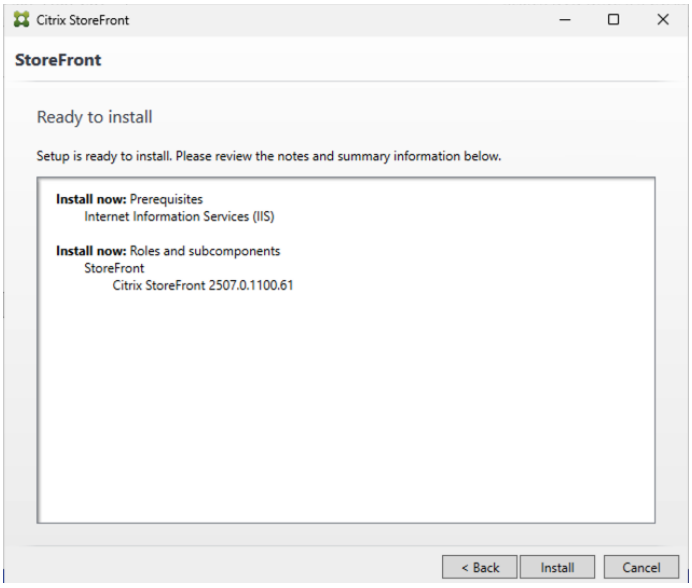


Figure 27. Installation of required prerequisites.

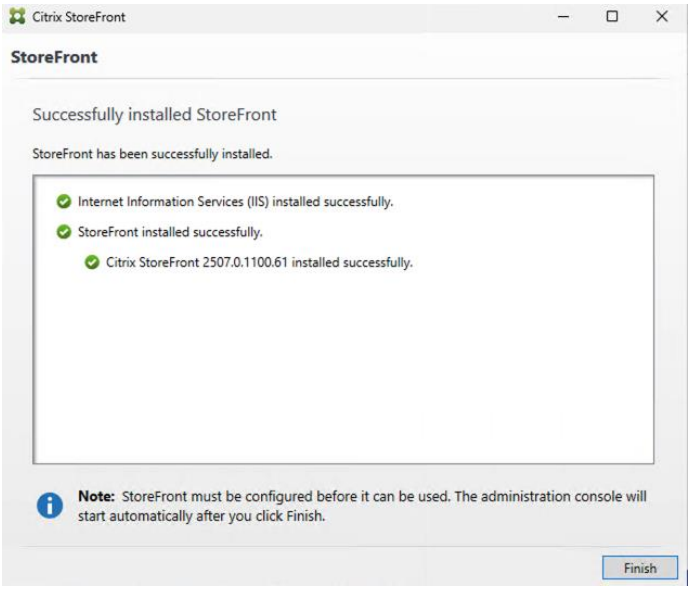


Figure 28. Installation of StoreFront and Internet Information Services (IIS) successfully completed.

- Verify Citrix StoreFront appears in: Start Menu, Services (Citrix StoreFront Service).

Configuring F5 BIG IP Virtual Appliance with Citrix Virtual Apps and Desktops

This section describes the configuration of an F5 BIG-IP Virtual Appliance to provide high availability and load balancing services for Citrix StoreFront servers in a Citrix Virtual Apps and Desktops environment. The scope of this configuration is intentionally limited to HTTPS load balancing in front of StoreFront using an HTTPS-to-HTTPS model (SSL bridging).
In this architecture, the F5 device is positioned as a reverse proxy in front of the StoreFront tier. It terminates client HTTPS connections and re-encrypts traffic toward the StoreFront servers.

StoreFront, in turn, continues to communicate directly and natively with the Delivery Controllers (DDCs) using its built-in load balancing and failover mechanisms. No configuration changes are required on the DDC layer, and no additional load balancing services are introduced between StoreFront and the Delivery Controllers.

Importantly, this deployment does not leverage the F5 Citrix iApp template, does not introduce an XML Broker virtual server, and does not modify or proxy Delivery Controller communication. The objective is to maintain architectural simplicity while delivering high availability and resiliency for the StoreFront tier.

This approach aligns with Citrix best practices by clearly separating responsibilities: F5 ensures external access and StoreFront availability, while StoreFront maintains control of brokering logic and DDC communication.

Architectural Scope and Design Intent

The implemented architecture focuses exclusively on providing a secure and resilient access point for StoreFront. HTTPS traffic from clients is directed to a single F5 virtual IP (VIP). The F5 appliance distributes this traffic across multiple StoreFront servers configured in a pool. From there, StoreFront communicates with the Delivery Controllers using its native configuration.

The traffic flow is illustrated below:

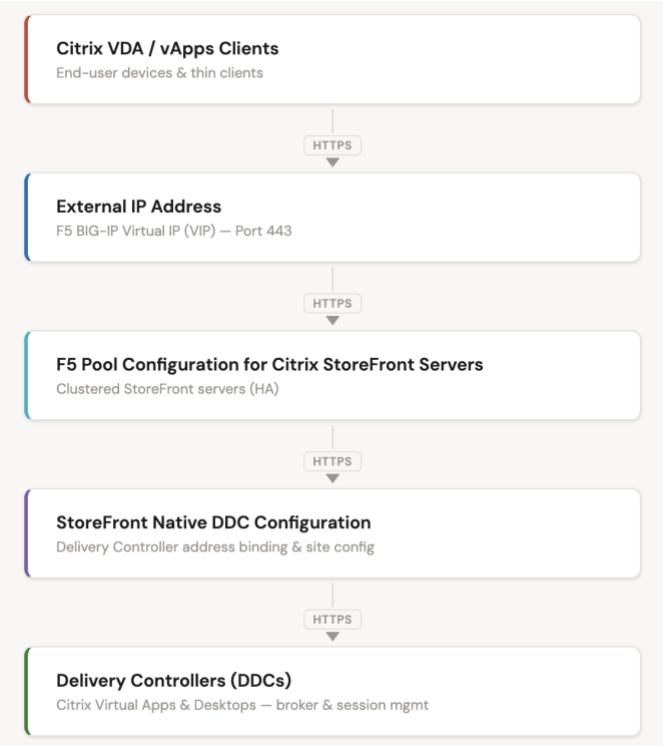


Figure 29. F5 BIG IP Architecture diagram with Citrix StoreFront.

NOTE: Only a single virtual server is required on the F5 appliance to support this design.

On this deployment, F5 system does not load balance XML brokers, does not expose ENUM ports, does not proxy ICA traffic, and does not implement gateway functionality. StoreFront continues using its internal logic to select and communicate with available Delivery Controllers. As a result, the Delivery Controller tier remains untouched and operates exactly as designed by Citrix.

The guiding principle behind this design is operational clarity. Each component maintains a clearly defined role:

- F5 provides secure external access and high availability for StoreFront.
- StoreFront manages authentication, enumeration, and brokering.
- Delivery Controllers manage session orchestration and resource allocation.

By preserving this separation, troubleshooting, upgrades, and lifecycle management remain straightforward.

F5 Configuration Procedure

The configuration described in this section is performed manually through the F5 BIG IP interface. The Citrix iApp template is not used in this solution. While iApps can provide advanced configuration workflows, this deployment requires only basic HTTPS load balancing functionality. For additional information related to F5 iApps, refer to:

https://cdn.studio.f5.com/files/k6fem79d/production/b0ce3221b98578206d90e5ec72aaaf37f749fd40.pdf

Creating the StoreFront Pool on F5 BIG IP

The first step is to create a pool that includes all Citrix StoreFront servers. This pool represents the backend service group to which the F5 virtual server will forward client traffic.

Within the F5 management interface, navigate to:

- Local Traffic, Pools, then click Create. Create a new pool named: `pool_storefront_https.` The health monitor should be set to https, or alternatively, a custom HTTPS monitor may be used if more granular health checking is required. The purpose of the health monitor is to ensure that only healthy StoreFront nodes receive traffic.
- Add each StoreFront server as a pool member using port 443, in our example and lab IP address they are:
  - `10.0.1.8:443`
  - `10.0.1.11:443`
  - `10.0.1.13:443`

- After saving the configuration, validate the pool status with the command line listed below and via CLI on the F5 instance. All members must display as "Available." If any member is offline, health monitor configuration or backend connectivity should be verified before proceeding.

```
tmsh show ltm pool pool_storefront_https
```

Creating the HTTPS Virtual Server

- Next, create the virtual server that will represent the external access point for StoreFront. Navigate to:

  Local Traffic, Virtual Servers, then click Create
- Assign the virtual server the name: `vs_storefront_https.` Set the destination address to the external VIP: `10.80.241.84`, then configure the service port as: `443`

This virtual server will accept client HTTPS connections and forward them to the StoreFront pool.

SSL Bridging Configuration

Because StoreFront only accepts HTTPS connections, SSL bridging must be configured. In this model, the F5 appliance terminates the client-side TLS session and establishes a new TLS session toward the backend StoreFront servers. Attach a Client SSL profile, such as: `clientssl_citrix` (or the default `clientssl` profile if appropriate). This profile defines how F5 handles incoming client TLS connections. Attach a Server SSL profile, such as: `serverssl`. This profile ensures that F5 re-encrypts traffic when forwarding it to the StoreFront servers. Without this configuration, traffic would be sent in clear text, which StoreFront does not accept.

HTTP Profile Configuration

An HTTP profile must be attached to the virtual server. This is a critical configuration requirement.
Select the standard: http profile. The http profile allows F5 to properly interpret and manage HTTP headers, redirects, and connection behavior. Without this profile, the appliance may treat traffic as generic TCP after decryption, which can result in connection resets.

It is important not to attach an HTTP/2 profile and not to enable QUIC for this design. Citrix StoreFront operates reliably behind F5 using HTTP/1.1, and introducing HTTP/2 or QUIC in this architecture may introduce compatibility challenges.

Source Address Translation (SNAT)

Source Address Translation must be enabled and set to: `Auto Map`. This configuration ensures that return traffic from StoreFront servers is routed back through the F5 appliance. If `SNAT` is not configured, StoreFront may respond directly to the client, bypassing F5, which can lead to asymmetric routing and connection resets.
`SNAT Auto Map` is mandatory in this design unless an alternative, explicitly defined routing configuration is implemented.

Default Pool Assignment

Finally, assign the previously created pool: `pool_storefront_https.` As the default pool for the virtual server. Save the configuration.

At this point, the F5 appliance is fully configured to provide high availability for StoreFront.

Validation and Testing

After configuration, validation should be performed both internally and externally. From an internal system, test the VIP using: `curl -vk http://<ip_address_of_the_external_F5_traffic_interface>`. A successful TLS handshake and an HTTP redirect response from StoreFront indicate correct SSL and HTTP profile configuration.

From a client workstation, access: `https://citrix.edgecloud.lab/Citrix/edgeWeb`

The StoreFront login page should load successfully. If authentication and application enumeration succeed, the configuration is confirmed operational.

Configuring Citrix StoreFront

Once you have the Citrix StoreFront successfully installed in all three StoreFront servers, perform these steps listed below on the first Citrix StoreFront instance**,** then propagate to all other two StoreFront instances.

- On the first StoreFront server, Open Citrix StoreFront Console
- Click Create a new deployment
- **StoreFront Base URL Configuration:** During the initial StoreFront configuration, the Base URL must represent the externally accessible, load-balanced entry point for user access. In environments where F5 BIG IP Virtual Appliance is deployed in front of multiple StoreFront servers, the Base URL is set to the F5 BIG IP virtual hostname rather than an individual StoreFront server. This Base URL resolves to the F5 BIG IP virtual IP (VIP) and is secured using HTTPS with a trusted SSL certificate. StoreFront uses the Base URL to generate service endpoints, authentication callbacks, and Receiver/Workspace URLs, making it critical that this value remains consistent and highly available. By configuring the Base URL to reference the F5 BIG IP VIP, the StoreFront deployment supports horizontal scaling, seamless failover, and consistent user access across all StoreFront nodes. Enter the Base URL then click Next.

  NOTE: Architecture Reminder: Citrix Apps/VDA Users → F5 BIG IP (Base URL) → Citrix StoreFront Group → Citrix Delivery Controllers (DDCs).

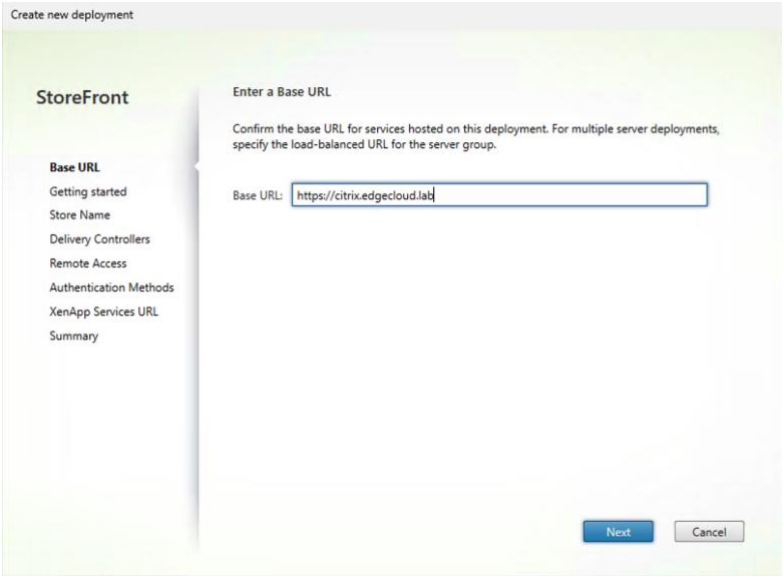


Figure 30. Citrix StoreFront Base URL configuration.

- **Store Creation and Access Configuration**: A Store is created to represent the logical access point through which users enumerate and launch applications and desktops. The Store name is defined at creation time and is displayed to users in Citrix Workspace and Receiver clients. Authentication is enforced for all access to the Store to ensure secure integration with Active Directory and F5 BIG IP. Anonymous access is disabled to align with enterprise security best practices. The Store is configured as the default website for the deployment, ensuring that users accessing the StoreFront Base URL are automatically redirected to the correct Store without additional navigation or configuration.

  The StoreFront store is created using a descriptive, platform-aligned name to clearly identify the service presented to end users. In this deployment, the store is named `EdgeCloud-Store`, providing clear association with the Edge Cloud platform while maintaining compatibility with Citrix Workspace, F5 BIG IP integration, and automation workflows.
- On this part of the configuration, enter the following:
  - Store name: Ex: `EdgeCloud-Store`
  - Uncheck or do not check the option: " Do not require authentication when accessing this store". On this architecture, Authentication must always be enforced
  - Check the option: "Set this website as the default for this deployment", This ensures that:
    - Ensures users accessing the Base URL are redirected to this Store
    - Required for clean F5 BIG IP and Workspace integration
    - Avoids multiple-store ambiguity
  - Click next

- **StoreFront with Delivery Controller Integration**: The Citrix site is defined to establish communication with the Citrix Delivery Controllers responsible for brokering application and desktop sessions. In this deployment, StoreFront is configured to communicate with the Delivery Controllers using HTTPS and its own load balance algorithm. This design ensures high availability of the Citrix control plane, enables encrypted and authenticated control-plane communication, and prevents StoreFront from depending on any individual Delivery Controller instance. All communication between StoreFront and the Delivery Controllers is secured using trusted server certificates, providing an additional layer of protection for internal brokering traffic while maintaining scalability and resiliency.
  - Display name: Controllers
  - Select the option: Citrix Virtual Apps and Desktops or DaaS

- On server load balanced option, click Add, and enter the name of all three Citrix Delivery Controller servers, in our example and lab, they are:
  - citrix-ddc-01.edgecloud.lab
  - citrix-ddc-02.edgecloud.lab
  - citrix-ddc-03.edgecloud.lab
- Check the Option: Server are load balanced
- Transport type: HTTPS
- Port 443
- No changes on the advanced settings options.
- Click OK then Next

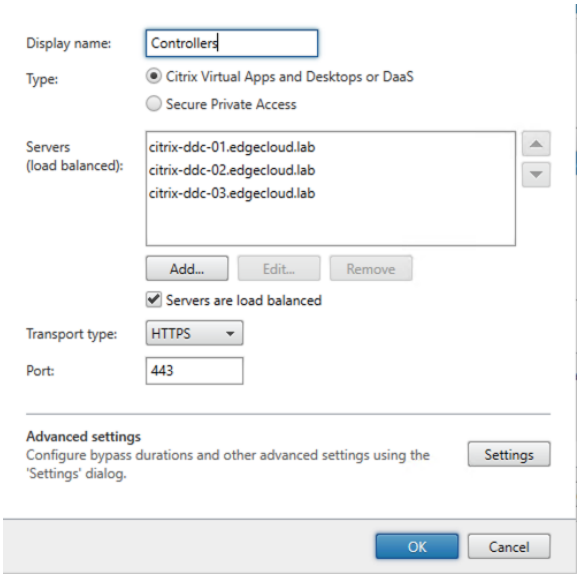


Figure 31. Citrix StoreFront with Delivery Controller integration.

- **Remote access:** In this architecture, the **Remote Access** option in StoreFront is intentionally not enabled. The solution does not deploy Citrix Gateway, ICA Proxy, or VPN-less external access. F5 BIG-IP is used strictly for HTTPS load balancing in front of StoreFront, while all VDA communication remains internal to the network. Since user access occurs within the trusted network and StoreFront communicates natively with the Delivery Controllers, configuring Remote Access is not required.

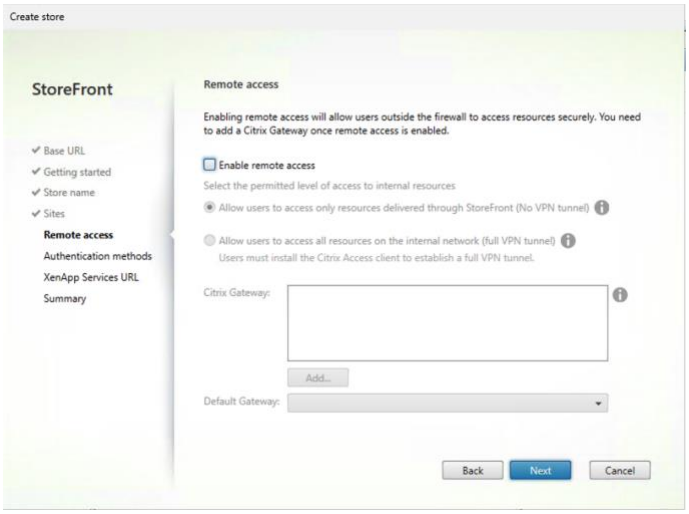


Figure 32. Citrix StoreFront remote access configuration.

- Authentication is configured using Active Directory username and password. This approach enables straightforward user authentication within the internal network and allows validation of StoreFront, Delivery Controller communication, and application enumeration without introducing external authentication dependencies.

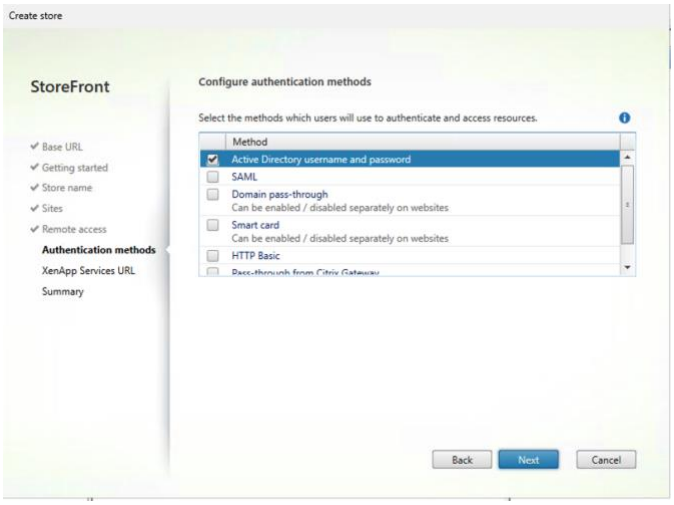


Figure 33. Citrix StoreFront authentication configuration.

- **XenApp Services URL Configuration:** The XenApp Services URL (PNAgent) is not enabled as part of this deployment. This feature is deprecated and intended only for legacy client devices that require the PNAgent protocol. Modern Citrix Workspace clients, browser-based StoreFront access, and F5 BIG IP integrations do not rely on this endpoint. Disabling the XenApp Services URL simplifies the StoreFront configuration, reduces attack surface, and aligns the deployment with current Citrix best practices.
- Click create, then the new store: `EdgeCloud-Store` will be successfully created. After that, click finish.

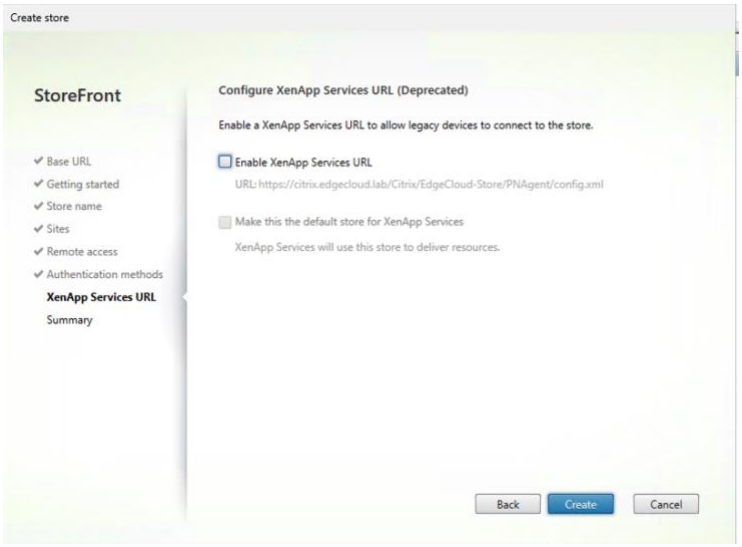


Figure 34. Citrix StoreFront XenApp Services URL.

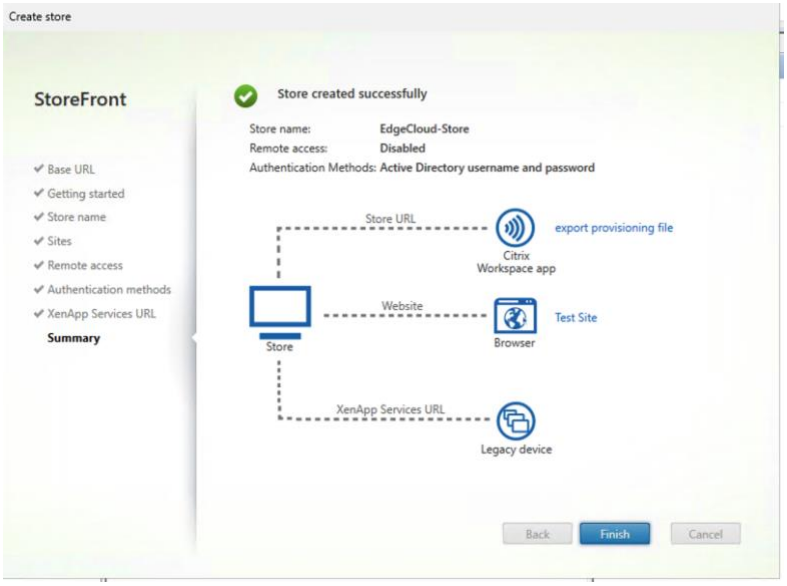


Figure 35. Citrix StoreFront – Store created successfully.

StoreFront Server Group Configuration

To complete a highly available StoreFront deployment, all StoreFront servers must be joined into a single StoreFront Server Group. In current Citrix StoreFront versions, additional servers must be explicitly authorized by the primary StoreFront server using a one-time join code. Lised below are the steps to join all StoreFront servers into a single StoreFront Server Group.

StoreFront Servers in this Deployment:

Role	Hostname
Primary StoreFront	citrix-store-01.edgecloud.lab
Secondary StoreFront	citrix-store-02.edgecloud.lab
Secondary StoreFront	citrix-store-03.edgecloud.lab

- Generate Authorization Code on Primary StoreFront. Perform the following on `citrix-store-01.edgecloud.lab`
- Open Citrix StoreFront Console
- In the left pane, select Server Group
- Click Add Server or Authorize Server (wording may vary by version)
- Copy the generated one-time join code
    - This code is time-limited
    - It can be used only for a short window
- Keep this window open while joining the other servers

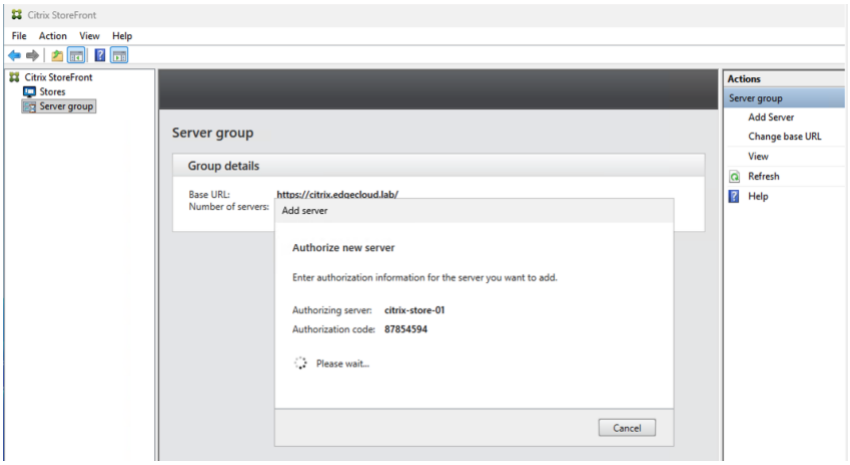


Figure 36. Citrix StoreFront – Store created successfully.

The primary StoreFront is now ready to accept new servers into the group. Perform the following on `citrix-store-02.edgecloud.lab`

- Open Citrix StoreFront Console
- Click Join existing server group
- When prompted, enter the primary StoreFront server's name: `citrix-store-01.edgecloud.lab`
- When prompted, paste the authorization code generated on the primary server
- Accept the security prompt
- Allow configuration replication
- Wait for the join operation to complete successfully, then `citrix-store-02` will be listed as a member of the StoreFront server group.

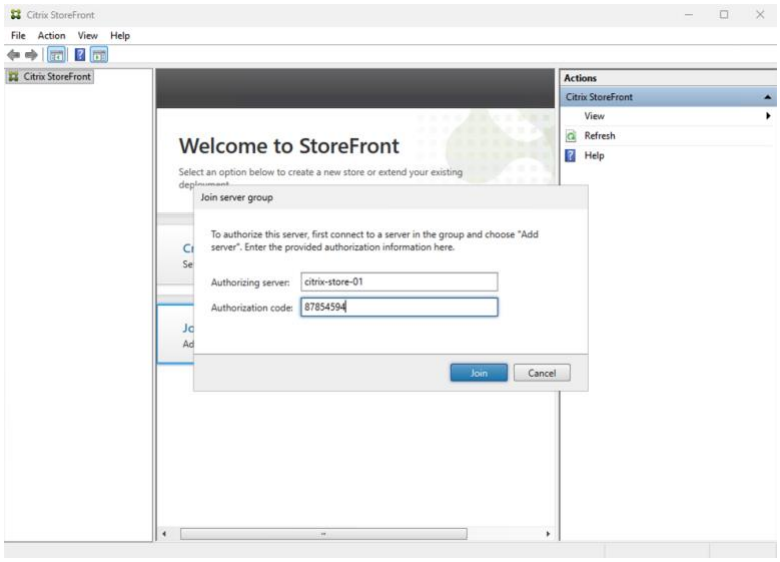


Figure 37. Citrix StoreFront – Join Citrix StoreFront servers to the Store group.

- When prompted, click OK, then click refresh. The `citrix-store-02` instance is now part of the same server group.

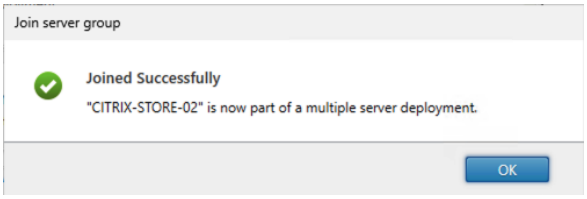


Figure 38. Citrix StoreFront – Join Citrix StoreFront servers to the Store group complete successfully.

- On the citrix-store-01 server, the propagation log will be available under the server details of the server group.

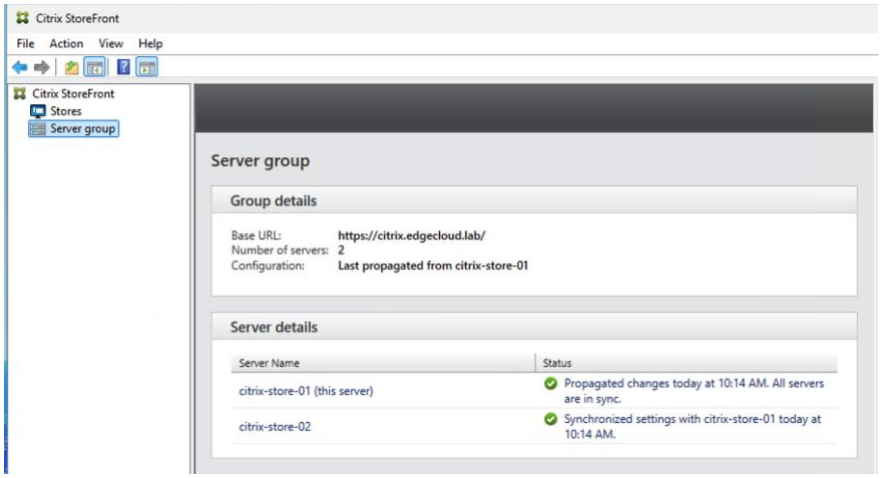


Figure 39. Citrix StoreFront – Citrix StoreFront propagation log.

- Repeat the same step on `citrix-store-03` server, then you will have all three Citrix StoreFront servers with the configuration propagated and in sync.

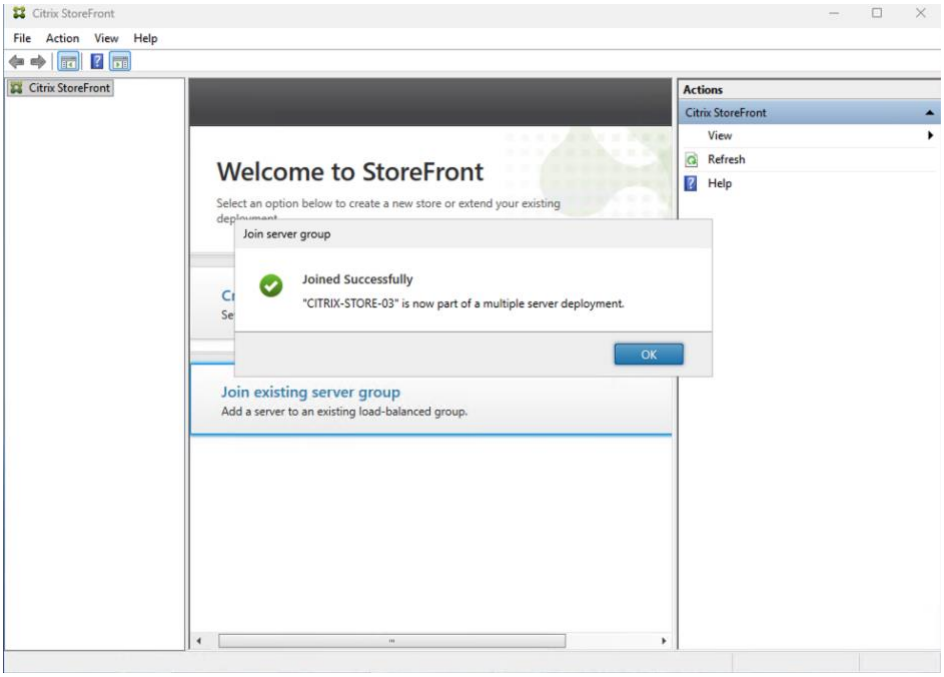


Figure 40. Citrix StoreFront – Joining additional StoreFront servers to the group.

NOTE: For each Citrix StoreFront server, a new code on the primary StoreFront server must be generated.

- Verify the StoreFront Server Group status. On citrix-store-01.edgecloud.lab, open the Citrix StoreFront Console, navigate to Server Group, and review the propagation logs to confirm that all three StoreFront servers are online. Ensure there are no replication, synchronization, or configuration propagation errors reported across the server group.

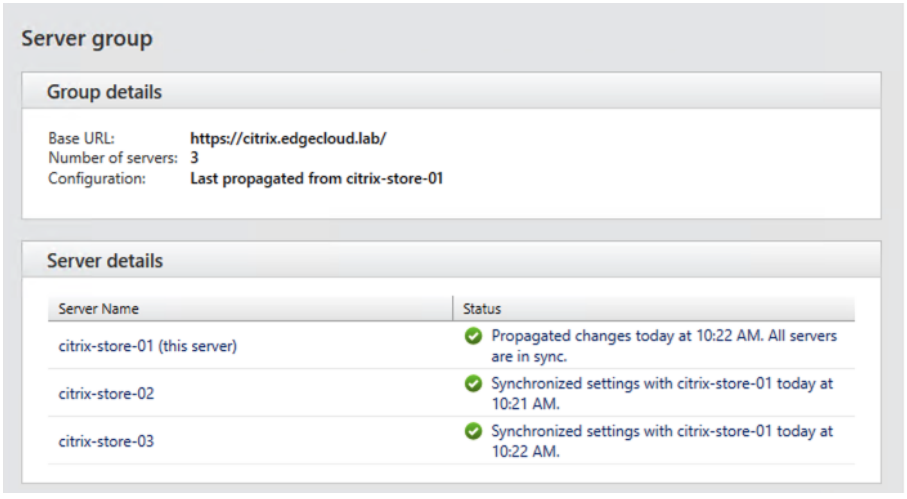


Figure 41. Citrix StoreFront –StoreFront Server group status.

Automation and Provisioning of Windows Desktops

The following example illustrates one possible approach to automate the cloning of Windows desktops on Compute Cloud@Customer (C3) or Private Cloud Appliance and automatically join them to Microsoft Active Directory. Other automation methods and tools can also be used depending on customer preferences and operational models.

The primary objective of this workflow is to clone Windows instances, automatically join them to Active Directory, and prepare them so that Citrix administrators can import the machines into the Citrix Delivery Controllers.

In the absence of native Citrix Machine Creation Services (MCS) integration with Compute Cloud@Customer (C3) or Private Cloud Appliance (PCA), this example solution leverages OpenTofu (Infrastructure-as-Code) together with PowerShell scripts to automate the deployment and configuration of Windows virtual desktops or virtual application hosts. This approach mimics the behavior of MCS while remaining fully declarative, automated, and cloud-native.

Prerequisites

Before running the deployment, ensure the following are in place:

- **Oracle Linux 9.x Bastion host**
  - OpenTofu ≥ 1.6 installed on the management host (tofu --version)
  - OCI CLI configured with a valid profile pointing to your C3 environment with the correct permissions to deploy instances only on the compartment being utilized by Citrix.
  - Network connectivity from the bastion host to the C3 API endpoint
  - A domain service account (e.g. SVC_DomainJoin) with rights to join computers to the target OU
  - The target subnet has connectivity to an Active Directory Domain Controller

- **Windows Images**
  - The Windows template must be properly prepared prior to deployment:
  - Install latest Oracle OCI CLI for Windows platform. Refer to this link: https://github.com/oracle/oci-cli/releases
  - Install latest Oracle VirtIO drivers. Refer to this link: https://docs.oracle.com/en/operating-systems/oracle-linux/kvm-virtio/kvm-virtio-DownloadingtheOracleVirtIODriversforMicrosoftWindows.html
- Install and configure the latest cloudbase-init and update its config file in your Windows golden-image. Download https://cloudbase.it/downloads/CloudbaseInitSetup_Stable_x64.msi. After the installation of cloudbase-init, run this PowerShell listed below as Administrator user to update the cloudbase-init config file for C3/PCA in your Windows golden-image.

```
$config = @"
[DEFAULT]
username=Administrator
groups=Administrators
inject_user_password=true
config_drive_raw_hhd=true
config_drive_cdrom=true
config_drive_vfat=true
bsdtar_path=C:\Program Files\Cloudbase Solutions\Cloudbase-Init\bin\bsdtar.exe
mtools_path=C:\Program Files\Cloudbase Solutions\Cloudbase-Init\bin\
verbose=true
debug=true
log-dir=C:\Program Files\Cloudbase Solutions\Cloudbase-Init\log\
log-file=cloudbase-init.log
default_log_levels=comtypes=INFO,suds=INFO,iso8601=WARN,requests=WARN
logging_serial_port_settings=
mtu_use_dhcp_config=true
ntp_use_dhcp_config=true
local_scripts_path=C:\Program Files\Cloudbase Solutions\Cloudbase-Init\LocalScripts\
metadata_services=cloudbaseinit.metadata.services.httpservice.HttpService
plugins=cloudbaseinit.plugins.common.userdata.UserDataPlugin
allow_reboot=true
stop_service_on_exit=false
check_latest_version=false
"@

$config | Set-Content "C:\Program Files\Cloudbase Solutions\Cloudbase-Init\conf\cloudbase-init.conf" -
Encoding ASCII
Write-Host "Config updated."
```

- Run this PowerShell as Administrator user to test it manually before syspreing/capturing the image:

```
& "C:\Program Files\Cloudbase Solutions\Cloudbase-Init\bin\cloudbase-init.exe" --config-file "C:\Program
Files\Cloudbase Solutions\Cloudbase-Init\conf\cloudbase-init.conf"
```

- Apply all Windows updates and security patches.
- Install Citrix VDA clients. Refer to Citrix official documentation: https://docs.citrix.com/en-us/citrix-daas/install-configure/install-vdas.html

- **Configure DNS Settings Before Sysprep:** Prior to running Sysprep and capturing the golden image, configure the Windows instance to use the appropriate internal DNS servers (typically the Active Directory–integrated DNS servers). Proper DNS configuration is mandatory to ensure successful domain join, SQL listener resolution, Citrix Delivery Controller communication, and overall control-plane functionality after deployment.

  Example PowerShell command to set a static DNS server on the primary network interface:

  ```
  Get-NetAdapter | Where-Object {$_.Status -eq "Up"} | `
    Set-DnsClientServerAddress -ServerAddresses ("Your_DNS_IP_Address”)
  ```

  Replace the IP addresses above with your Active Directory DNS server IPs.

- **Disable Proxy Usage for the Metadata Endpoint (169.254.169.254):** Ensure that no system-wide or WinHTTP proxy configuration intercepts traffic to the Cloudbase-Init metadata endpoint (http://169.254.169.254). Cloudbase-Init requires direct, link-local access to this address to retrieve instance metadata and execute user_data scripts at first boot. Proxy redirection will cause metadata retrieval failure, preventing automated password injection, machine rename, and domain join operations.

  To remove any configured WinHTTP proxy:

```
netsh winhttp reset proxy
```

To verify direct metadata connectivity before Sysprep:

```
Invoke-RestMethod -Uri "http://169.254.169.254/openstack/latest/meta_data.json"
```

The command must return a valid JSON response. Any proxy error or timeout indicates that proxy bypass is not properly configured.

- **Sysprep the Windows golden image.** To avoid the need for a console connection on Compute Cloud@Customer (C3) or Private Cloud Appliance (PCA) during first boot, customers have two supported options:
  - **Use an Unattend.xml file during Sysprep** that explicitly defines the local Administrator password and suppresses OOBE interaction. This method prevents the forced password change prompt, eliminates interactive OOBE screens, and allows the instance to boot cleanly without requiring manual intervention or console access.
  - **Pre-configure a temporary Administrator password before running Sysprep.** In this approach, ensure the Administrator account has a password set, the password is not expired, and the "User must change password at next logon" option is disabled. When properly configured, Windows will complete OOBE without prompting for a password reset, allowing fully automated deployment without console interaction.

  For customers choosing the Unattend.xml method, run Sysprep on the Windows instance using the Unattend.xml file as shown below:

```
<unattend xmlns="urn:schemas-microsoft-com:unattend">
  <settings pass="oobeSystem">
    <component name="Microsoft-Windows-Shell-Setup"
               processorArchitecture="amd64"
               publicKeyToken="31bf3856ad364e35"
               language="neutral"
               versionScope="nonSxS">

      <AutoLogon>
        <Password>
          <Value>Your_Password</Value>
          <PlainText>true</PlainText>
        </Password>
        <Username>Administrator</Username>
        <Enabled>true</Enabled>
        <LogonCount>1</LogonCount>
      </AutoLogon>

      <UserAccounts>
        <AdministratorPassword>
          <Value>Your_Password</Value>
          <PlainText>true</PlainText>
        </AdministratorPassword>
      </UserAccounts>

    </component>
  </settings>
</unattend>
```

Example of the sysprep command line with unnattend.xml file:

```
Start-Process "$env:SystemRoot\System32\Sysprep\sysprep.exe" `
-ArgumentList "/generalize", "/oobe", "/shutdown", "/quiet", "/unattend:C:\unattend.xml" `
-Wait
```

NOTE: For additional information related to sysprep, refer to the Microsoft official documentation: Refer to the official Microsoft Windows sysprep documentation: https://learn.microsoft.com/en-us/windows-hardware/manufacture/desktop/sysprep--system-preparation--overview?view=windows-11

IMPORTANT: After, the Windows operating system is shutdown, stop the instance on Compute Cloud@Customer or Private Cloud Appliance UI, then capture the shutdown instance as a Custom Image in Compute Cloud@Customer or Private Cloud Appliance. Go to C3/PCA UI → Compute → Instances → your instance → Actions → Create Custom Image while it's in STOPPED state.

- Join the Windows instance to the Active Directory domain and register the machine in the Citrix Delivery Controllers (DDCs) for inclusion in the target Machine Catalog.

Windows Instances Automated Deployment

This section describes the automated deployment of Windows virtual desktops on Oracle Compute Cloud@Customer (C3) or Oracle Private Cloud Appliance (PCA) using OpenTofu as an Infrastructure-as-Code (IaC) framework. The provided OpenTofu automation code enables consistent, repeatable, and scalable provisioning of Citrix-ready Windows instances from a standardized golden image. By leveraging declarative configuration files, administrators can deploy, scale, and manage virtual desktop workloads in a controlled and cloud-native manner while maintaining full alignment with enterprise governance and operational standards.

Download the latest OpenTofu automation package here and follow the step-by-step instructions below to configure and execute the deployment.

Step 1 - Clone or copy the deployment files

Place all files on your bastion host within a dedicated working directory. Ensure you maintain the following directory structure:

```
citrix-automation-v1.0/
├── deploy.sh                    # Main deployment script — run this
├── main.tf                      # VM provisioning (for_each, stagger logic)
├── data.tf                      # Existing instance discovery & auto-sequencing
├── variables.tf                 # All input variable definitions
├── versions.tf                  # Provider version requirements
├── provider.tf                  # OCI provider configuration
├── terraform.tfvars             # Customer configuration file (edit this)
└── cloud-init/
    └── domain_join.ps1          # Windows AD domain join script (Cloudbase-Init)
```

Step 2 - Edit terraform.tfvars

Open terraform.tfvars and update all variables highlighted in Red below.

```
# ──────────────────────────────────────────────
# C3 Platform
# ──────────────────────────────────────────────
oci_profile         = "The profile name in the OCI CLI config file, typically located at /root/.oci/config."
compartment_ocid    = "OCID of the C3/PCA compartment where Citrix desktops will be deployed"
availability_domain = "AD-1"

# ──────────────────────────────────────────────
# Networking
# ──────────────────────────────────────────────
vcn_ocid    = "OCID of the C3/PCA VCN"
subnet_ocid = "OCID of the C3/PCA Subnet"

# ──────────────────────────────────────────────
# Instance Sizing
# ──────────────────────────────────────────────
shape                = "VM.PCAStandard.E6.Flex"
ocpus                = 4 # Enter the desireable number of OCPUs p/ virtual desktops
memory_gbs           = 64 # Enter the desireable amount of memeory p/ virtual desktops
boot_volume_size_gbs = 50 # Enter the desireable size of boot volume p/ virtual desktops

# ──────────────────────────────────────────────
# Golden Image (sysprepped, already available on C3)
# ──────────────────────────────────────────────
image_ocid = "Enter the OCID of the Windows image available on C3/PCA"

# ──────────────────────────────────────────────
# SSH Access
# ──────────────────────────────────────────────
ssh_authorized_keys = "Enter the SSH key to be added to the instances during deployment"

# ──────────────────────────────────────────────
# Deployment
# ──────────────────────────────────────────────
instance_count       = 100 # Enter the number of instaces to be deployed on C3/PCA from the golden_image
instance_name_prefix = "citrix-vdi" # Enter the instace_prefix_name for the instances

# ──────────────────────────────────────────────
# Active Directory Domain Join
# ──────────────────────────────────────────────
ad_domain        = "corp.example.com" # Enter the Active Directory Domain Name
ad_join_username = "SVC_DomainJoin" # Enter the username fo the Domain
ad_ou_path       = "OU=citrix,DC=edgecloud,DC=lab" # update to the correct Active Directory Citrix OU
```

- Enter the Windows Active Directory password (for the account joining virtual desktops to the domain via variable in your bastion host, for example:

  ```
  export TF_VAR_ad_join_password="your-password"
  ```

Step 3 - Initialize OpenTofu

Run this once (or after any provider version change):

```
cd citrix-automation/
tofu init
```

```
Expected output: OpenTofu has been successfully initialized.
```

Step 4 - Preview the deployment plan

Run the following command line:

```
tofu plan
```

Review the output carefully. Key things to verify:

- **next_available_name_preview** - confirms the names that will be created (e.g. citrix-01 … citrix-10, or continuing from an existing sequence)
- **source_id under source_details** - confirms the correct golden image OCID
- **fault_domain is not present in the plan** - this is correct and means C3 will assign automatically
- **freeform_tags**. Name matches the desktop's display name

IMPORTANT: Only proceed if the plan shows N to add, 0 to change, **0** to destroy. Never proceed if destroy is greater than zero.

Step 5 - Apply the deployment

Run the following command line:

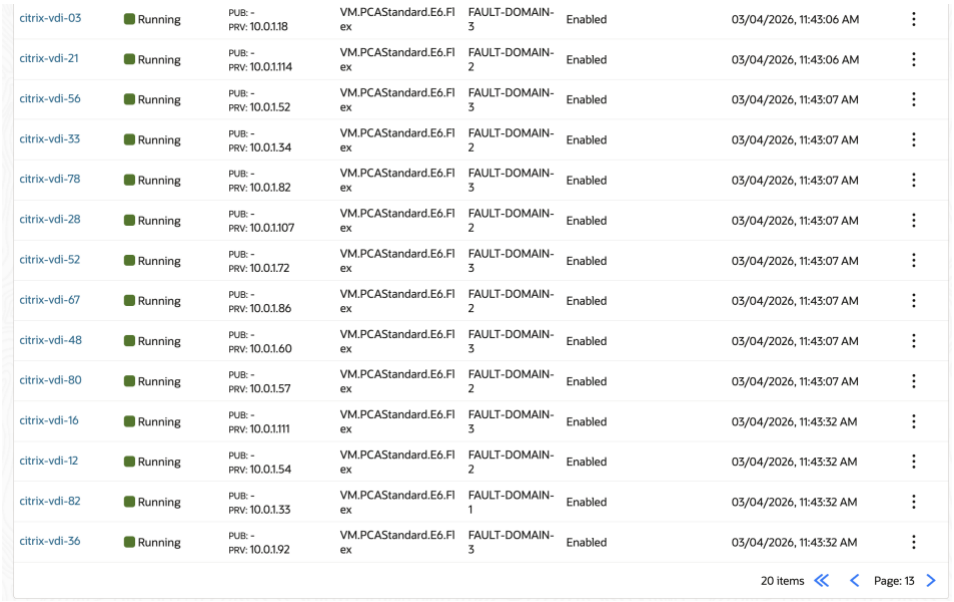
```
bash deploy.sh
```

Type `yes` when prompted. OpenTofu will:

- Query existing instances in the compartment to detect the current name sequence
- Clone instance_count desktops from the golden image
- Submit each instance to C3 with automatic fault domain placement
- Inject domain_join.ps1 as cloud-init userdata into each instance's metadata
- Join the new Windows Server 2025 or Windows 11 instances to the Microsoft Active Directory domain

Listed below is an example of a Windows virtual desktop deployed on Compute Cloud@Customer (C3) and automatically joined to Microsoft Active Directory using the OpenTofu automation script, along with the corresponding deployment output log.

In this example, 100 virtual desktops were successfully deployed on Compute Cloud@Customer and automatically joined to Active Directory as part of the automated provisioning workflow.



citrix-vdi-03	Running	PUB: - PRV: 10.0.1.18	VM.PCAStandard.E6.Flex	FAULT-DOMAIN-3	Enabled	03/04/2026, 11:43:06 AM	
citrix-vdi-21	Running	PUB: - PRV: 10.0.1.114	VM.PCAStandard.E6.Flex	FAULT-DOMAIN-2	Enabled	03/04/2026, 11:43:06 AM	
citrix-vdi-56	Running	PUB: - PRV: 10.0.1.52	VM.PCAStandard.E6.Flex	FAULT-DOMAIN-3	Enabled	03/04/2026, 11:43:07 AM	
citrix-vdi-33	Running	PUB: - PRV: 10.0.1.34	VM.PCAStandard.E6.Flex	FAULT-DOMAIN-2	Enabled	03/04/2026, 11:43:07 AM	
citrix-vdi-78	Running	PUB: - PRV: 10.0.1.82	VM.PCAStandard.E6.Flex	FAULT-DOMAIN-3	Enabled	03/04/2026, 11:43:07 AM	
citrix-vdi-28	Running	PUB: - PRV: 10.0.1.107	VM.PCAStandard.E6.Flex	FAULT-DOMAIN-2	Enabled	03/04/2026, 11:43:07 AM	
citrix-vdi-52	Running	PUB: - PRV: 10.0.1.72	VM.PCAStandard.E6.Flex	FAULT-DOMAIN-3	Enabled	03/04/2026, 11:43:07 AM	
citrix-vdi-67	Running	PUB: - PRV: 10.0.1.86	VM.PCAStandard.E6.Flex	FAULT-DOMAIN-2	Enabled	03/04/2026, 11:43:07 AM	
citrix-vdi-48	Running	PUB: - PRV: 10.0.1.60	VM.PCAStandard.E6.Flex	FAULT-DOMAIN-3	Enabled	03/04/2026, 11:43:07 AM	
citrix-vdi-80	Running	PUB: - PRV: 10.0.1.57	VM.PCAStandard.E6.Flex	FAULT-DOMAIN-2	Enabled	03/04/2026, 11:43:07 AM	
citrix-vdi-16	Running	PUB: - PRV: 10.0.1.111	VM.PCAStandard.E6.Flex	FAULT-DOMAIN-3	Enabled	03/04/2026, 11:43:32 AM	
citrix-vdi-12	Running	PUB: - PRV: 10.0.1.54	VM.PCAStandard.E6.Flex	FAULT-DOMAIN-2	Enabled	03/04/2026, 11:43:32 AM	
citrix-vdi-82	Running	PUB: - PRV: 10.0.1.33	VM.PCAStandard.E6.Flex	FAULT-DOMAIN-1	Enabled	03/04/2026, 11:43:32 AM	
citrix-vdi-36	Running	PUB: - PRV: 10.0.1.92	VM.PCAStandard.E6.Flex	FAULT-DOMAIN-3	Enabled	03/04/2026, 11:43:32 AM	

20 items Page: 13

Figure 42. Windows virtual desktop deployed on Compute Cloud@Customer or Private Cloud Appliance

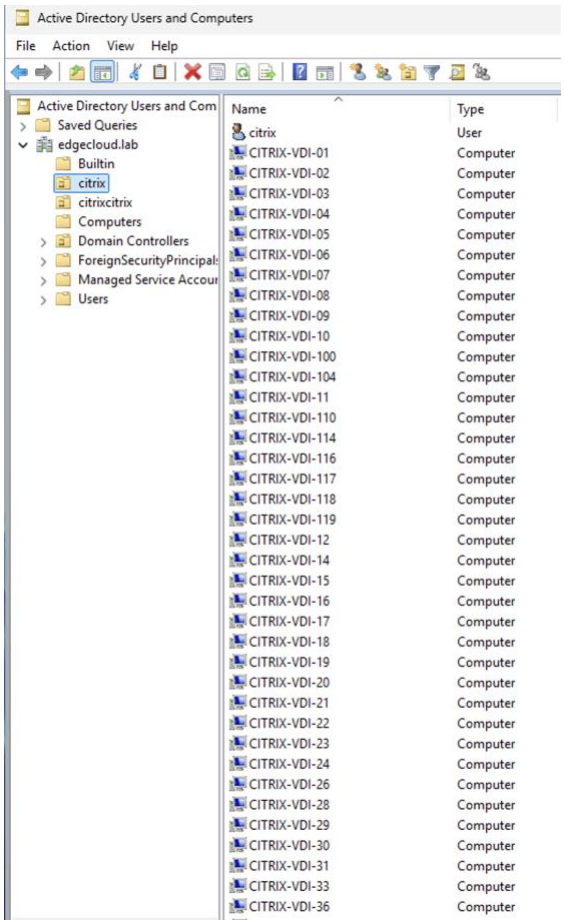


Figure 43. 100 Windows virtual desktop automatically joined to Microsoft Active Directory.

The sequence is: Boot → rename → DC reachability check → AD join → reboot.

NOTE: Allow 1-5 minutes after the instance reaches RUNNING state for the domain join and reboot to complete.

Step 7 - Import desktops into Citrix DDC

On the Citrix Delivery Controller, import the machine accounts from Active Directory into the target Machine Catalog. In the example below, all virtual desktops (citrix-vdi-01 through citrix-vdi-100) that were deployed on Compute Cloud@Customer using the OpenTofu automation described earlier are discovered from Active Directory and made available to the Citrix environment. These machines can then be added to the Machine Catalog and managed by the Citrix Delivery Controllers as part of the VDI deployment.

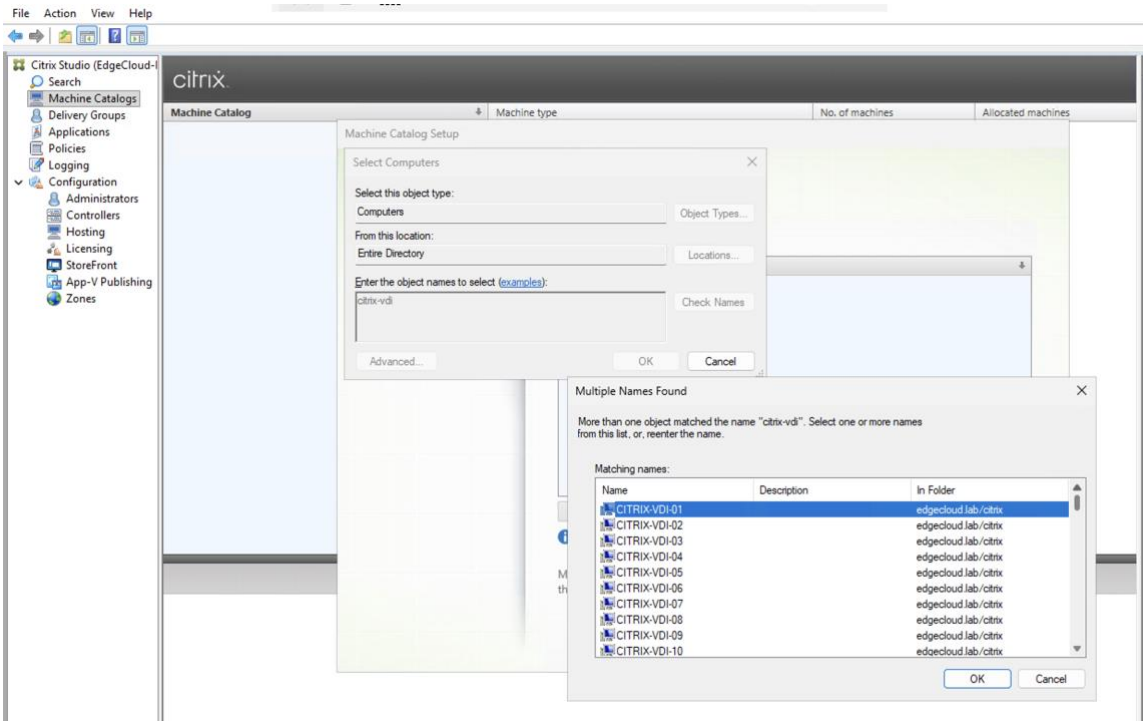


Figure 44. Windows virtual desktops import into the Citrix target machine catalog.

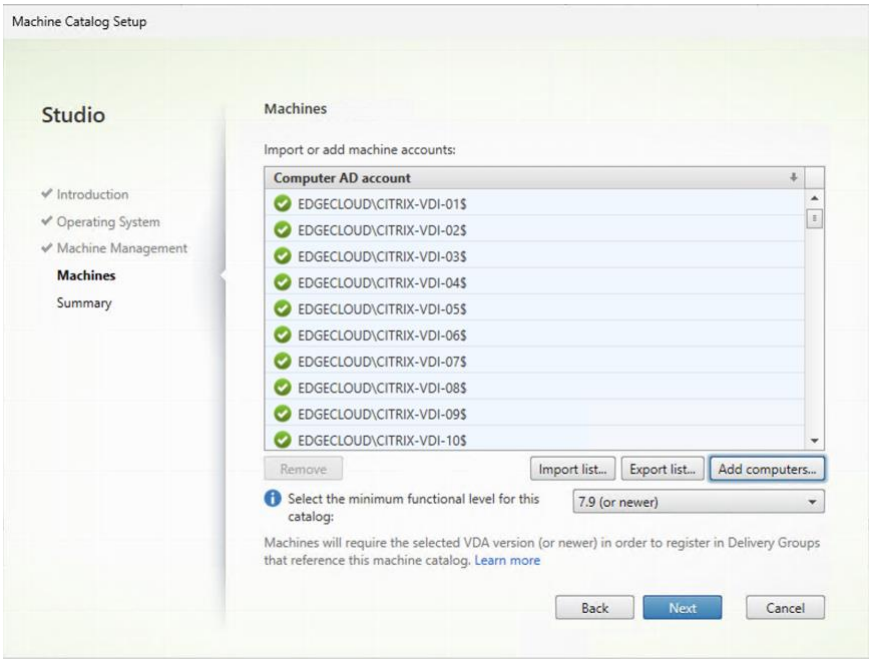


Figure 45. Citrix Mahine Catalog Setup.

Below is an example of a new Delivery Group configured to use 98 out of the 100 Windows Server 2025 instances deployed on Compute Cloud@Customer. These instances were automatically joined to the Microsoft Active Directory domain and are configured to deliver five applications that will be made available to end users.

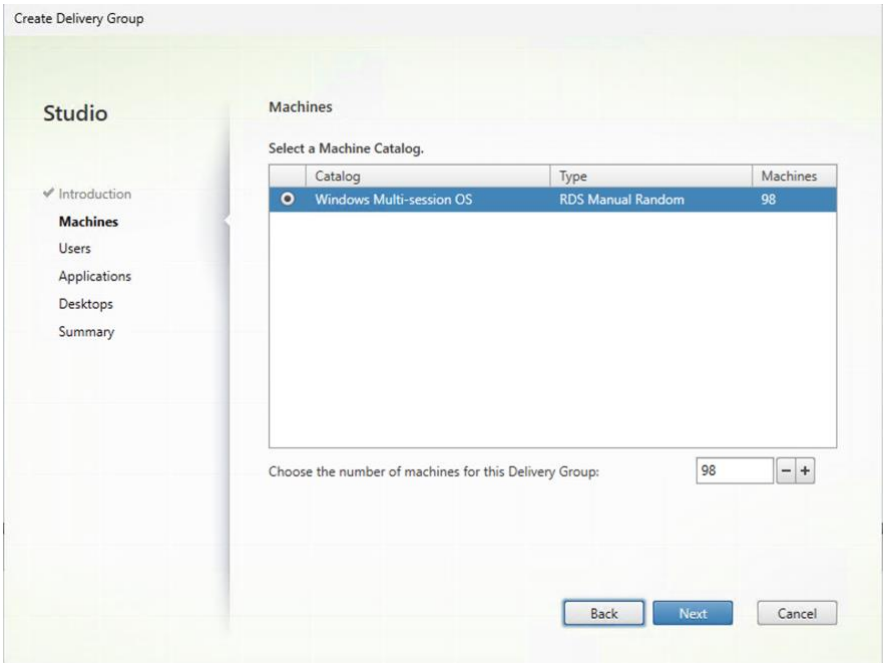


Figure 46. Citrix Delivery Group setup – Machines.

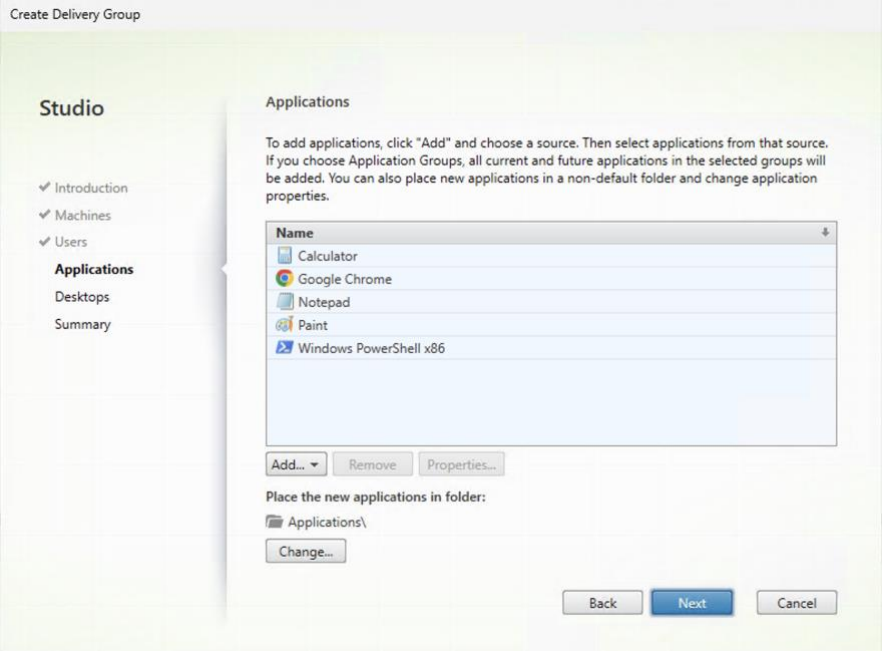


Figure 47. Citrix Delivery Group setup – Applications.

At this point the desktops are visible in Citrix Studio and can be assigned to Delivery Groups, presented to end users, as presented below:

- Access the Citrix external endpoint (for example: https://citrix.edgecloud.lab/Citrix/edgeWeb/) and log in using the username and password provided by your Citrix administrator. The virtual apps and virtual desktops previously configured will be available for use.

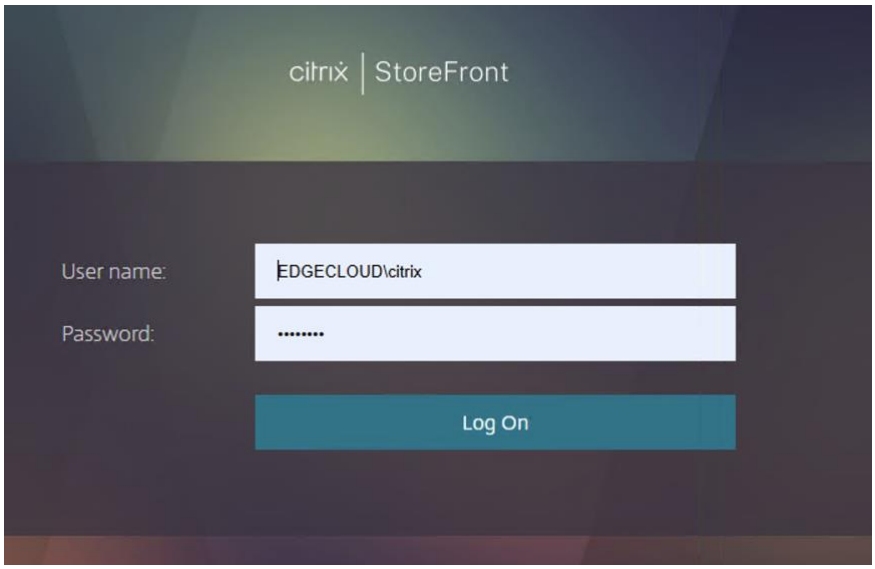


Figure 48. Accessing Citrix Virtual Apps and Desktops.

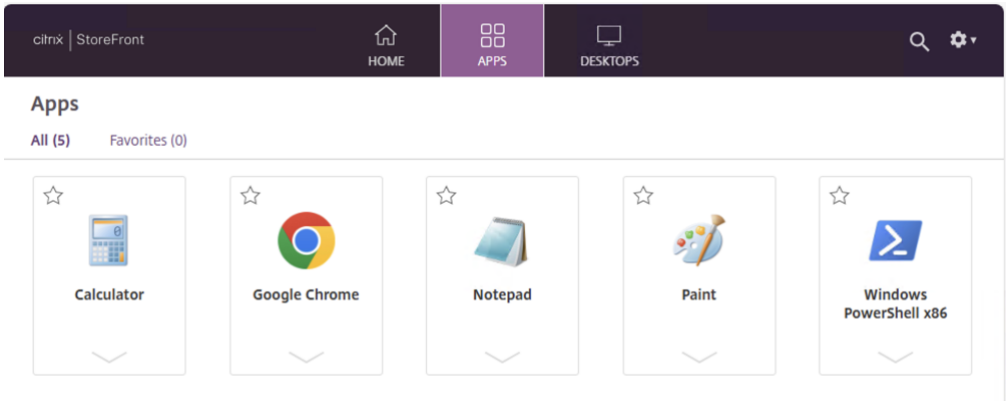


Figure 49. Accessing Citrix Virtual Apps.

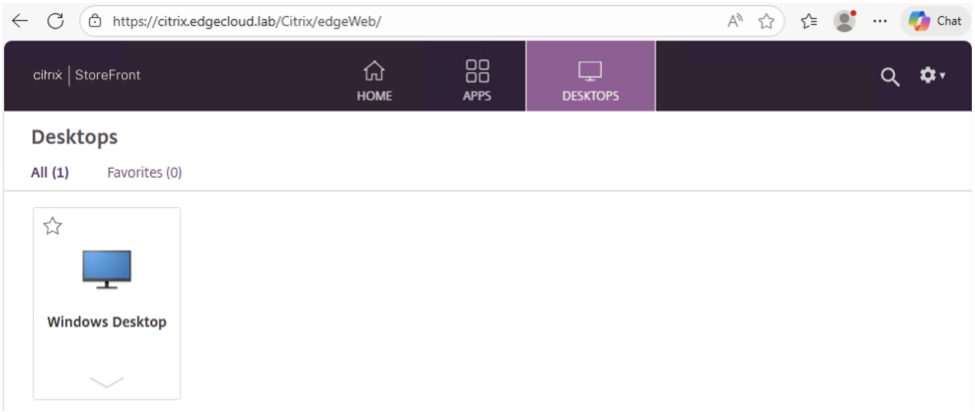


Figure 50. Accessing Citrix Virtual Desktops.

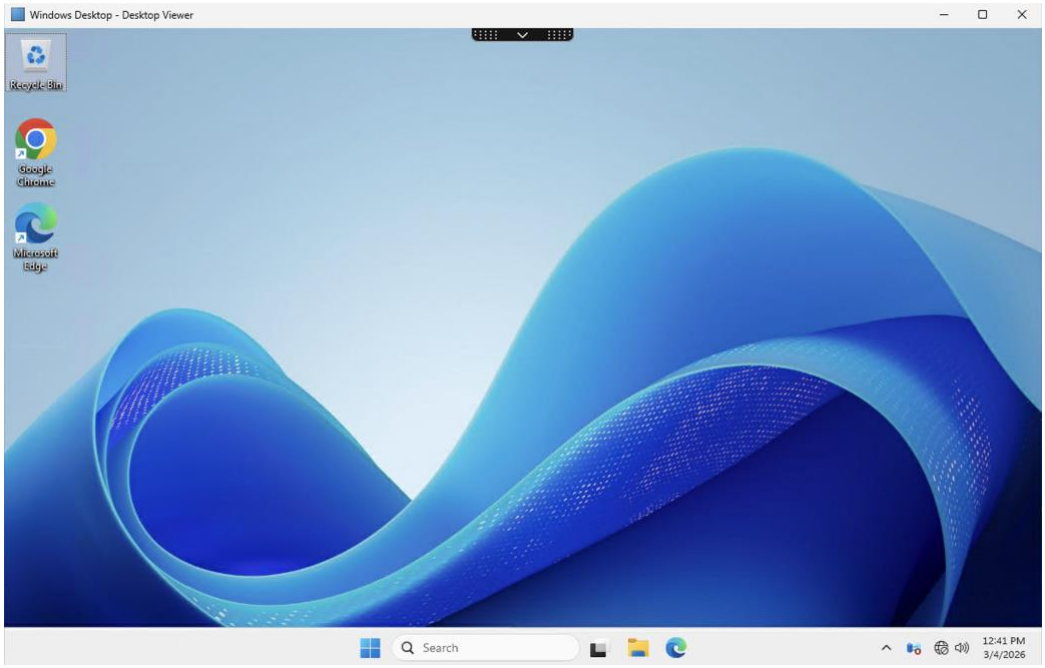


Figure 50. Citrix Virtual Desktop instance running on Compute Cloud@Customer.

Step 8 - Deploying additional desktops (subsequent runs)

To add more desktops later, simply increase `instance_count` or run a new apply. OpenTofu will automatically detect the highest existing sequence number and continue from there, no manual renaming or conflict resolution needed.

Example: 10 desktops already deployed, add 5 more

```
# Edit terraform.tfvars

sed -i 's/instance_count.*/instance_count = 5/' /c3-images/citrix-automation-v3/terraform.tfvars

# Review plan — must show 5 to add, 0 to destroy

tofu plan

# Deploy

bash deploy.sh
```

Listed below is the correct output:

```
Plan: 5 to add, 0 to change, 0 to destroy.

Do you want to perform these actions?
  OpenTofu will perform the actions described above.
  Only 'yes' will be accepted to approve.

  Enter a value: yes

oci_core_instance.vdi_desktop["citrix-vdi-122"]: Creating...
oci_core_instance.vdi_desktop["citrix-vdi-121"]: Creating...
oci_core_instance.vdi_desktop["citrix-vdi-124"]: Creating...
oci_core_instance.vdi_desktop["citrix-vdi-123"]: Creating...
oci_core_instance.vdi_desktop["citrix-vdi-125"]: Creating...

Apply complete! Resources: 5 added, 0 changed, 0 destroyed.

=== Clearing local deployment cache ===
  ✓ Cache cleared for: data.oci_core_instances.existing
  ✓ Cache cleared for: oci_core_instance.vdi_desktop["citrix-vdi-121"]
  ✓ Cache cleared for: oci_core_instance.vdi_desktop["citrix-vdi-122"]
  ✓ Cache cleared for: oci_core_instance.vdi_desktop["citrix-vdi-123"]
  ✓ Cache cleared for: oci_core_instance.vdi_desktop["citrix-vdi-124"]
  ✓ Cache cleared for: oci_core_instance.vdi_desktop["citrix-vdi-125"]

✓ Deployment cache cleared.
✓ All Virtual Desktop instances remain running on C3/PCA.
✓ Next deployment will safely add new instances only.
```

The five additional Windows virtual desktops/instances will then be successfully deployed on Compute Cloud@Customer (C3) or Private Cloud Appliance (PCA), automatically joined to Microsoft Active Directory, and made ready for import into the Citrix Delivery Controller servers.

Figure 51. Additional Citrix Virtual Desktop instance joined to Microsoft Active Directory.

IMPORTANT: Deployment Safety and Operational Boundaries

The OpenTofu and PowerShell automation scripts provided in this solution are intentionally scoped to additive operations only, creating and provisioning new Virtual Desktop instances on Oracle Compute Cloud@Customer (C3) or Private Cloud Appliance (PCA). This design decision is deliberate and aligned with enterprise safety principles for production VDI environments.

By design, the automation will never perform destructive operations such as terminating, deleting, or modifying existing Virtual Desktop instances. Each execution of tofu apply is idempotent with respect to existing infrastructure. It detects currently running instances, calculates the next available sequence numbers, and provisions only the newly requested desktops without touching any previously deployed resources.

This boundary ensures that a misconfiguration, an incorrect variable value, or an unintended execution of the automation cannot result in accidental deletion of production desktops or loss of user data. In large-scale VDI environments, where a single operation could affect thousands of active user sessions, this constraint provides a critical safety guarantee.

Decommissioning and deletion of Virtual Desktop instances is outside the scope of this automation and is the sole responsibility of the customer's IT operations team. Customers must establish and maintain their own controlled decommissioning procedures, subject to their internal change management, security, and compliance policies.

The following mechanisms are available to support this responsibility:

- Compute Cloud@Customer or Private Cloud Appliance Management Console, using the native instance termination workflow with appropriate administrative approvals.
- A dedicated decommissioning scrip, which requires explicit operator confirmation before any termination may be adapted to meet the customer's specific operational requirements.
- An authorized IT Service Management (ITSM) workflow integrated with the Compute Cloud@Customer or Private Cloud Appliance Management APIs, subject to the customer's change management controls.

Oracle and its partners bear no responsibility for Virtual Desktop instances that are decommissioned outside of approved customer procedures, or for any service disruption resulting from unauthorized or unplanned deletion of infrastructure resources.

ORACLE

Connect with us

Call +**1.800.ORACLE1** or visit **oracle.com**. Outside North America, find your local office at: **oracle.com/contact**.

blogs.oracle.com facebook.com/oracle twitter.com/oracle

Author: Anderson Souza