A discussion on mixing operating systems in modern computer networks

Can mixed operating systems enhance the functionality of a modern corporate network?

Author:
Alexander Solvang
alexander@solvang-it.no

Course Code	FS1SP1131
Course Title	SP1
Assignment nr.	5
Assignment title	Mixed Environment
Date	February 2024

Contents

List of Figures

1 Introduction

1.1 Terms and abbreviations

- OS = Operating System
- AD = Active Directory (On-prem).
- ADDS = Active Directory Domain Systems
- CT = Computer Technology
- MS = Microsoft®
- NIC = Network Interface Card
- PKI = Public Key Infrastructure
- VM = Virtual Machine
- SID = Security Identifier
- GID = Group Identifier
- UID = User Identifier
- GC = Global Catalog
- PS = Powershell
- MMC = Microsoft Management Console
- DMZ = Demilitarized Zone, Perimeter Network

1.2 Project Outline

This report aims to answer the question: *"Can mixed operating systems enhance the functionality of modern corporate computer networks?"*.

The following requirements are assigned:

1. Create a proof-of-concept network as follows:
2. Install Microsoft Active Directory Domain Services (ADDS). AD must authenticate and authorize users and computers throughout the network.
3. Install an Apache Web Server onto an Ubuntu server.
4. Include Linux clients and the web server in ADDS.
5. Create a network diagram in packet tracer with a suggested physical network design.
6. Discuss security implications in a mixed environment.

PfSense and the FreeBSD OS are used in the Proof of Concept lab, providing an opportunity to investigate a third OS and its interaction with Active Directory. The last element intends to address the topic for argument and relate to a security perspective.

This report intends to examine the administrative tools offered by ADDS and how they interact with other operating systems, as well as consider what effect mixing operating systems has on overall network security. An attempt to identify and illustrate best practices is also a goal.

The main part consists of three sections, where the first (2) attempts to provide relevant background information and highlight the necessary considerations and technical details relevant to performing the given tasks successfully.

The second section (3) shows the actual installation and configuration of the virtual network environment in VmWare, and the third section (4) provides a proposed network design using Packet Tracer.

It is assumed that the network design, to some degree, should demonstrate solutions that could transfer to a production network.

Another assumption is that the corporation has several departments needing different levels of access to network resources; examples will include ”web developers” needing access to the web server, an Office department using regular workstations, and IT Staff responsible for managing ADDS.

2 Preliminary research

The following section shows preliminary research performed to identify prerequisites and uncover challenges and opportunities when mixing operating systems in an Active Directory Domain.

2.1 A brief introduction to Operating Systems and Active Directory

Operating Systems are a vast topic on their own, and this report only includes a few key notes relevant to the overall discussion and a fundamental technical explanation. Operating systems have been around for a long time as an integral part of modern computer and information technology. Simply put, an operating system is a collection of software used to control and manage computer hardware resources and functions as a bridge between applications in user space, the user, and said hardware.

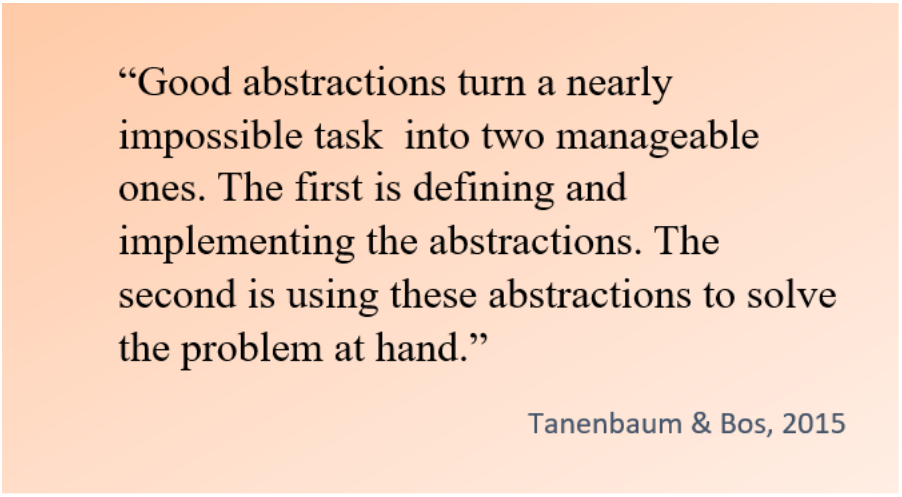

Figure 1: (Tanenbaum and Bos 2015)

With its Kernel, a key software component allowing the OS to make system calls (access hardware), ”the OS creates an abstraction of the physical hardware, allowing applications and application developers to work with a unified and consistent set of instructions to access hardware resources on a computer” (Tanenbaum and Bos 2015).

The OS handles creating, scheduling, processing, and terminating jobs (processes and threads) to run on the CPU. ”The OS also creates abstractions like file systems and memory space address tables and allows applications to create, read, and write files without dealing with the messy details of how the hardware works ”(Tanenbaum and Bos 2015).

Gnu/Linux OS has a reputation for not being very user-friendly and being preferred only by advanced users. In recent years, this has changed, and new Gnu/Linux distros are increasingly intuitive and user-friendly. But under the hood, the abstractions created by Windows and Unix-like operating systems are inherently different. Necessary abstractions like the file system, memory management, and process handling have been developed and implemented differently, and historically, this has caused compatibility issues.

One example is ”the concept of a file as an abstraction to objects created and stored by processes (this could be a user process creating a document) being given a name. Some file systems won’t care if you use upper or lower case letters, meaning that file.txt and FILE.txt on some systems represent one file but are seen as two different files on the next ”(Tanenbaum and Bos 2015).

Modern Windows operating systems ship with their native NTFS (new technology file system) and Linux with ext4. Accessing information on an EXT4 partition from a Windows system and vice versa is only possible using additional software to transform or interpret data. Files and filesystems are just one example. However, it is essential to consider incompatibilities when deploying different operating systems in your network environment.

Another area where operating systems are different is the native authentication and authorization mechanisms. Authentication and authorization occur in several situations, not only when a user logs into a system or network.

When a user, application, or other entity, like a computer, needs access to resources on a network, a process to identify, authenticate, and determine a level of authorization must occur. In this

assignment, we want ADDS to provide these services for the entire network, so we need to determine if the security mechanisms provided by AD are compatible with those available in Ubuntu.

Kerberos is the default domain (network) authentication protocol on current Microsoft operating systems and multiple NTLM versions. Windows also supports Digest, a Lightweight Directory Access Protocol (LDAP)(Jungles Patrick et al. 2012). The Microsoft Security Support Provider Interface (SSPI) makes the foundation for Windows authentication(Microsoft Learn 2021c)

Any user, application, or service that needs authentication and authorization connects to a Security Support Provider (SSP) like Kerberos, LDAP, or NTLM through the SSPI(Microsoft Learn 2021b). Kerberos was initially developed at MIT in the 80s and became an IEFT Standard in 1993. it's officially reached version 5 and is described in RFC4120 (Neuman and et al. 2005). Today, Kerberos is used in all major OS' from Apple to Microsoft and Linux(Kerberos Consortium 2007).

Similarly, Ubuntu and other Linux distros support System Security Services Deamon (SSSD) for network authentication. SSSD is an open-source client component of centralized identity management solutions that can connect to directory servers like, for example, AD or OpenLDAP(https://sssd.io 2021a).

On a host level, traditional authentication mechanisms used in Unix- and Linux-based operating systems originate from the GNU C Library and the POSIX standard, where user information is stored in files like /etc/passwd, /etc/groups. Modules like NSS (Name Service Switch) and PAM (Pluggable Access Modules) provide a way for applications to access the information contained therein and perform authentication and authorization.

SSSD creates a bridge between the external identity store (LDAP), authentication provider (Kerberos), authorization provider (AD), and internal equivalents, leveraging both NSS and PAM.

SSSD is developed and maintained with Red Hat support by the Open Source community, and "simply put, the main purpose of SSSD is to store data from the remote database (in this case AD) in a local cache and then serve this data to the target user (Application) on request" (https://sssd.io 2021b).

When SSSD is configured to use AD as the identity provider, it will, as mentioned, use Kerberos as the authentication provider, and the process will establish an encrypted connection through the GSSAPI, as opposed to an Open LDAP provider where TSL should be applied to encrypt the authentication and authorization session.

A notable difference is how the two operating systems handle the enumeration of users and groups; historically, this has offered a significant challenge in attempting to use AD with Linux clients. The operating systems themselves need a way to identify entities in a non-human readable format, which is solved differently in Windows and Linux operating systems; the OS does not recognize the actual names written out in plain text but rather unique attributes assigned to each entity.

AD is object-oriented, meaning that any entity, whether a user, service account, security group, or computer, is provided with a unique Security Identifier (SID). Linux systems enumerate with User IDs and Group IDs (UID and GID) that are POSIX compliant, following a fixed structure with dedicated numerical series for various account types. Today's version of SSSD has integrated ID mapping, providing automatic SID to UID and GID translation without performing AD configuration; this can, however, be configured in SSSD configuration on the client side.

So far, we have identified what protocols and tools are needed to identify, authenticate, and authorize users and computers on Linux-based systems with AD. The next step is to investigate ADDS.

2.2 Active Directory

”Active Directory, the identity glue that binds on-premises Microsoft networks, is at the center of almost all on-premises networks” (Orin 2020).

Active Directory (AD) is a proprietary solution from MS providing centralized management of users, devices, and service accounts across domains and forests. At its core, AD provides a distributed database for storing detailed information about users, computer assets, and service accounts as objects(Orin 2020).

AD is designed as a hierarchical logical structure that stores information about objects and their attributes that applications may use and process. AD also provides security through control mechanisms leveraging authentication and authorization with policy-based administration to secure directory objects(Microsoft Learn 2022b).

A few key features are:

- The Schema - A set of rules defining object classes and attributes.
- A global catalog - Contains information on all directory objects across domains.
- Replication service - Allows replicating any changes to the directory between domain controllers.
- DSA - Directory System Agent - Services and processes that provide access to the data store, using mechanisms like LDAP.

AD provides an extensive range of tools to manage and secure your assets and your networks. As a bare minimum, one needs Domain Controllers and DNS services to manage your directory and its inventory, but the capabilities extend far beyond simple identity management(Orin 2020).

The active directory Schema contains definitions on each object class that can be created in AD, and every object class is assigned various attributes containing details such as passwords, addresses, departments, managers, titles, or configuration settings(Microsoft Learn 2020a). The Schema is included in the GC together with a partial replica of every naming context, allowing applications and users to find any object at any location in the ADD tree without knowing the distinguished name (DN) of the object(Microsoft Learn 2020b).

2.2.1 Domain Controllers

A domain controller is a server role that provides the directory database but can optionally provide DHCP and DNS services. Although it might be necessary and valuable to host DNS on dedicated servers throughout an extensive network, installing DNS services on domain controllers is a common practice to allow unified management of network assets and addressing from a single location.

DCs are crucial assets with the power to control all your network assets and should be well protected. DCs are possibly the single highest value target for malicious hackers. If such an actor gains control of your DC, they essentially gain control of all identities, and every computer joined to the domain(Orin 2020).

Additional tools and features, like organizational units, security groups, and group policies, help administrators achieve granular control of computer and user configurations, access control, and auditing capabilities. Another risk factor essential to consider is availability.

Since the DC must be accessible to perform authentication in real-time, it's considered best practice to achieve redundancy with at least two DCs for every Domain, thus leveraging ADs' built-in replication capability.

Organizational Units are represented as objects in AD, and together with GPOs, they create the foundation for AD capabilities on crafting permissions, access control, and security.

Another important aspect of securing availability is the FSMO or Flexible Single Master Operations roles. There are a total of five different roles in this category:

1. Schema master – The single Server in the forest is allowed to process updates to the AD Schema.
2. Domain naming master – A forest-level role responsible for adding and removing domains from the forest and managing references to domains in trusted forests.
3. PDC Emulator – Responsible for processing changes to account passwords and synchronizing time on domain-joined computers.
4. Infrastructure master – Keeps track of changes that occur in other domains in the forest as they apply to objects in the local Domain
5. RID master – The relative identifier master processes RID requests from DCs in a specific domain using a combination of relative ID and domain security ID to create a unique security ID (SID) for the object(Orin 2020)

Notice that the PDC Emulator is responsible for domain time synchronization. Avoiding time skew between domain clients is critical, as ample time skews may cause problems. Many applications and network features rely on robust time synchronization; log-on or Kerberos authentication and claims-based SSO (Single Sign On) may fail due to unsynchronised clocks(Microsoft Learn 2023b).

2.2.2 ADDS

Active Directory Domain Services contains forests and domains, where a forest contains one or more domains sharing a common schema and a few additional security principals(Orin 2020). Although having several domains in a forest is possible, it is rarely necessary as a single AD forest can create just under 2,15 billion objects during its lifetime(Agile IT 2009).

There are usually only two good reasons for hosting more than one Domain: if your organization is geographically dispersed or so large, you must serve tens of thousands of employees(Orin 2020). In large companies, delegating permissions and access to resources can be challenging. AD allows the creation of Organisational Unit (OU) objects and security groups, which in turn may contain multiple users or other types of objects like groups and computer objects. A well-planned OU and group structure can significantly simplify delegating permission, access control, and other administrative tasks on a network.

2.2.3 GPO

We already mentioned Group Policies or, more accurately, Group Policy Objects in conjunction with OUs and objects. GPOs can configure both users and computers from AD and can perform a wide variety of tasks. GPOs can be linked to objects like OUs and security groups and can be used to set password policies or to map and mount network drives on a range of computers automatically. GPOs are also an important tool to enhance and control security throughout the network. GPOs can be applied on different levels and are processed in the following order, where subsequent policies take effect over previously processed policies in case of conflict(Orin 2020).

1. Local (Computer)
2. Site
3. Domain
4. OU

These tools were initially designed for native Windows operating systems, but with the implementation of Adsys, a range of GPO controls are available for Ubuntu clients, and will be investigated further.

2.2.4 DNS

DNS (Domain Name System) allows applications and users to locate network resources and is an integral part of Active Directory. Without it, communication between computers and services on a network would be impossible. In its basic form, DNS translates hostnames like website addresses, or Fully Qualified Domain Names (FQDN), to IP addresses or, the other way around, IP addresses to host names(Orin 2020).

A working DNS server must be in place to successfully join a Linux client to AD, and a DNS record for the client must be in place. While not a strict requirement, it is also recommended to create a PTR record, as many Linux services use reverse lookup(Canonical 2022).

2.2.5 Practical considerations and best practices

To deploy AD DS, thus creating a new Domain, we need to install Windows Server as a foundation for our deployment. As the introduction mentions, this section attempts to demonstrate best practices to secure the deployment. New Windows Server versions are released with a ten-year LTSC (Long-term Servicing Channel), where security updates and patches are guaranteed(Microsoft Learn 2022a). It is considered best practice to install the latest Stable release.

Each release comes with a few different editions based on different deployment scenarios. Windows Server 2022, the latest release, has three editions:

- Standard
- Datacenter
- Datacenter: Azure Edition

Identifying a clear and exhaustive set of best practices is challenging, but Microsoft makes several clear recommendations. The first one to mention is to deploy AD Domain Controllers and other suitable server roles using the Server Core installation option, in other words, without a GUI.(Orin 2020).

Installing Windows Server without a GUI goes a long way to minimize the attack surface, reduce unnecessary server applications, and enhance security. All applications may be subject to vulnerabilities, so it is best practice to install only the applications and services you need for every server and client in the network.

In this regard, Microsoft recommends preventing domain controllers from accessing the internet and using a dedicated server as an intermediary to provide updates. It is well-known that security breaches have occurred due to administrators using web browsers installed on DCs.

Another recommendation states that administrators should avoid logging on directly on production servers and use a secure workstation for remote administration dedicated only to this purpose(Orin 2020). The reason is apparent: daily tasks like checking e-mail, surfing the internet, or working with office documents increase the risk of malware infections.

If such functions are performed on critical infrastructure, a virus or a potential intruder can wreak havoc on the network. To facilitate this, we will install a Windows server named PAW (Privileged Access Workstation) and ensure it is the only workstation that can be used to access our domain controllers, AD, and firewalls.

The latter leads us to implement a Least Privilege Administrative model and secure administrative hosts. The first principle states that all users should log on with a user account that has the absolute minimum permissions needed to perform current tasks at any given moment. Network administrators should use separate accounts for network administration and other work-related tasks. (Microsoft Learn 2023a). In the clear, this means using the PAW for administrative tasks on the network, a regular workstation for office-related activities, and having two different user accounts accordingly.

In large, we should mention three principles for securing administrative hosts offered by Microsoft:

1. The administrative workstation used for remote administration should be as secure as the trusted system under administration.

2. Consistently implement several authentication factors. A simple username and password is not enough.

3. Remember to maintain the physical security of the critical infrastructure.

In addition to well-known two-factor authentication, the functionality to authenticate with ”something you have” is also supported by Windows, providing possibly three authentication factors when security is in high demand. If the appropriate hardware like fingerprint scanners or smartcard readers are available, GPOs can be used to implement network-wide log-on requirements(Microsoft Learn 2021a).

These topics will never be exhausted, and there is undoubtedly much more to say about additional steps to secure a network environment. The approaches available to implement these principles will vary and must be adapted to your particular scenario and network design.

Before installing AD DS components, we must consider one more recommendation. Domain Controllers are responsible for storing critical information on network users and entities and are crucial assets for live authentication and authorization. We must always ensure these assets' availability for users, devices, applications, and services throughout a network.

Redundancy and availability are partly achieved by installing a minimum of two DCs in every Domain that replicate each other continuously, where if one goes offline for some reason, this critical functionality is not entirely lost(Orin 2020).
This principle also applies to networking devices, power supply, and network connections if high availability is in demand.
The advice to install only software and services strictly needed for a server or computer to serve its intended purpose also applies to Linux and Ubuntu. Canonical also presents a few best practices when implementing SSSD; ”To avoid a KDC spoofing attack, SSSD has a setting called krb5_validate which, when set to True, will verify the KDC server it is talking to. This setting defaults to True when using the ad id_provider” (Canonical 2022).

One should consider the security implications of allowing SSSD to cache credentials locally. Even if this may be practical to ensure that users can log in, even if AD is unavailable, it poses a security risk. A better approach might be focusing on high availability for your DCs and network connections.

2.3 Preplanning

The section below displays the preplanning performed to accomplish a proof of concept demo lab. An attempt is made to include any best practices discovered so far, including other technical details uncovered in the previous section, and general well-known principles.

Only IPv4 addressing is used and enabled throughout this lab. Also, note that ”Internal-” and ”Core-Network” are used interchangeably, and both refer to the Secure LAN segment hosting ADDS.

- IPv4 addressing scheme adhering to the limitations found in the virtual environment.
- Configure our VmWare environment.

  - Create Vmnets and configure VmWare according to the virtual lab's network plan and overall goals.
  - Create a shared storage folder on the host to facilitate filesharing between VMs.

- Install two PfSense routers and firewalls to control and secure our networks.
  - Configure static routes.
  - Configure default routes.
  - Configure Firewall rules to allow clients to authenticate with AD.
  - Enable NTP server.
  - Create an Internal CA to sign certificate requests from internal network clients. This will also allow securing communications using the WEB GUI to access the PfSense servers.
  - Use LDAP for authentication and authorization with AD.

- Install and configure ADDS
  - Two domain controllers with no GUI for high availability.
  - Use a secure, Privileged Access Workstation (PAW) to administer our DCs and PfSense remotely.
  - Ensure we have a working DNS server (DCs) and configure PTR records to serve reverse DNS lookups.
  - Provide an example of implementing the principle of least access.
  - Utilize AD capabilities, using OUs and Security Groups, to control access to our network assets.
  - GPO to enable time synchronization for our DC1 with PfSense using NTP.
  - Import and deploy ADM files to the Central Store in AD to enable Adsys and GPOs for Ubuntu.

- Install an Ubuntu Server with no GUI, using SSSD to authenticate and authorize the server and users with AD as our provider.
  - Apply recommended best practices from Canonical in the SSSD configuration.
  - Ensure the KDC is validated to avoid spoofing.
  - Disable local credential Cahing.
  - Replace the default keytab file to make it harder for malicious actors to locate keys.
  - Ensure correct permissions and file ownerships are applied to configuration files.
  - Enable NTP from DC 1.

- Install an Apache WebServer on the Ubuntu server, securing it using SSL Server certificates to enable TLS encryption on web connections.

- Configure firewalls to allow TCP on port 443 and port 80, both on the local machine and the Edge firewall.

- Install and configure SSSD as with the Ubuntu desktop.

- Restrict server access so only Domain Admin and Web Developer security group members can log on to the server.

As mentioned in the introduction, an assumption is made that any corporation using an Apache Web server has a Web development department, an IT department, and a general Office department. These should be included and reflected in AD Users and computers with a suitable OU and Security Group structure for Group Policy Management.

2.3.1 Networkdesign in the Virtual lab

Planning the lab and the network design, it is attempted to demonstrate a viable design within the limitations of VmWare and the equipment available.
The lab requires the following hosts:

- PfSense Edge router/Firewall
- PfSense Internal router/Firewall
- DC1
- DC2
- PAW
- Ubuntu server running Apache WebServer
- An Ubuntu Desktop
- A Windows Desktop

Planning the allocation of hardware resources in a virtualized environment is imperative. Disk space is not accounted for in this demo; allocating storage and designing hardware configurations, such as raid configuration, type, and number of server disks, must be considered in a production network. Hypervisor cores and RAM are allocated as shown in figure 2.

VM	Cores	RAM
PF_Edge	1	512
PF_Core	1	512
DC1	1	4096
DC2	1	4096
Apache	2	4096
PAW	2	4096
WIN10 Desktop Client	2	4096
Ubuntu Desktop Client	2	4096
Total	12	25

Figure 2: Hardware resources assigned to VMs(Solvang 2023).

Two PfSense routers and Firewalls will control and protect the lab network. Creating two security boundaries between our ADDS environment and the internet provides an extra layer of security for our critical infrastructure, which is placed on its designated network segment, or secure LAN. A second network segment is used as a perimeter network, hosting internet-facing applications, and the third provides a LAN used for regular workstation clients. A simple diagram is shown in figure 3.

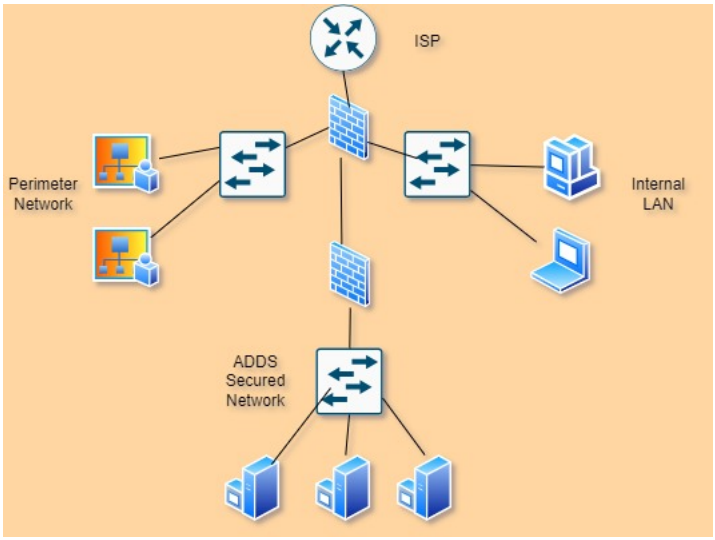

Figure 3: Network design illustration(Solvang 2023).

Network Segment:	LAN	Secure LAN	Permieter (DMZ)	WAN
Network	192.168.159.0/24	10.10.10.0/24	172.31.100.0/24	
Server Range	.10 - .19	.10 - .49	.10 - . 254	
Static Client Range	.20 - .49	.49 - .254	.49 -.254	
DHCP Range	.49 - .199	No	No	
PfSense Edge	192.168.159.5		172.31.100.5	192.168.8.1
PfSense Internal	192.168.159.6	10.10.10.5		
DC1		10.10.10.10		
DC2		10.10.10.11		
PAW		10.10.10.20		
Apache			172.31.100.10	
Windows Desktop	192.168.159.20			
Ubuntu Desktop	DHCP			

Figure 4: IPv4 Addressing Plan(Solvang 2023).

This setup will also allow us to block all outgoing connections from the Secure LAN, thus preventing our DCs from connecting to the internet. As the PAW, in principle, should be as secure as the system under administration, it will be placed in the same segment, but we still need to allow it to communicate at least with the Edge_router outside this barrier.

No subnetting is performed; all networks are designated to their own classful /24 network, with private IPv4 address ranges assigned by IANA in RFC 1918(Rekhter et al. 1996). IPv4 addresses are assigned as seen in figure 4.

3 Proof of concept lab - installation and configuration

This section provides technical details and illustrations demonstrating how to install and configure the lab environment. Only one VM creation is explained in detail, as this process is similar for all VMs. Note that configurations and creation of objects or installations are not necessarily presented in the order they are performed, as the following subsections are organized by operating systems or software.

3.1 Preparing the VmWare Hypervisor

The host is an MSI laptop running a Windows 11 Pro OS with limited hardware resources. 32GB of RAM and 14 processor cores running at 2 GHz are available, running VmWare Pro 17 as a hypervisor software.

The first step is configuring the virtual network adapters and creating the needed networks. VmWare workstation networks are virtual, unmanaged switches called vmnets, each confined to a predefined network.

As mentioned, a custom network configuration with three separate /24 classful networks is created, and two PfSense firewalls and routers control the traffic flow between them (Vmware 2019). Even though only one of the three NICs on the edge router has internet connectivity through a bridge to the hypervisor's NIC, all clients in every network can connect once the firewalls with their routing capabilities are installed and properly configured.

When a vmnet is created, VmWare installs virtual network adapters on the host, with options to configure DHCP and DNS settings and enable time synchronization between the guest and the host. VmWare can also synchronize time for each Virtual machine and create shared folders.

In this lab, default vmnets are deleted, and DHCP and time sync are switched off. A shared folder is created on the host and mounted for selected VMs to allow the sharing of files between them if needed.
The following steps are performed to prepare VMWare for the lab:

- Open the network editor in VMware from the dropdown menu found under "Edit" as seen in figure 5, then select "Virtual Network Editor."
- To make changes, we must escalate our privileges; click the "Change Settings button and then "Yes" in the following Windows dialogue to open the editor with administrator privileges.

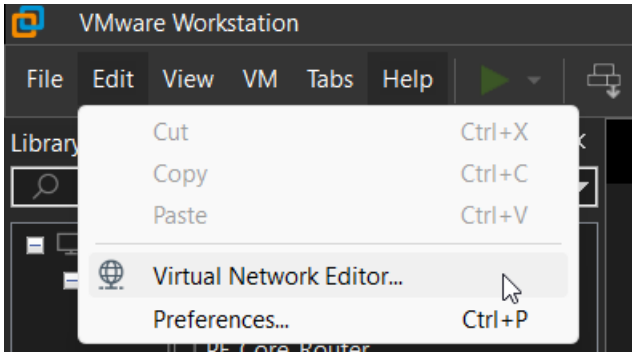

Figure 5: Screenshot: VmWare Edit dropdown menu (Solvang 2023)

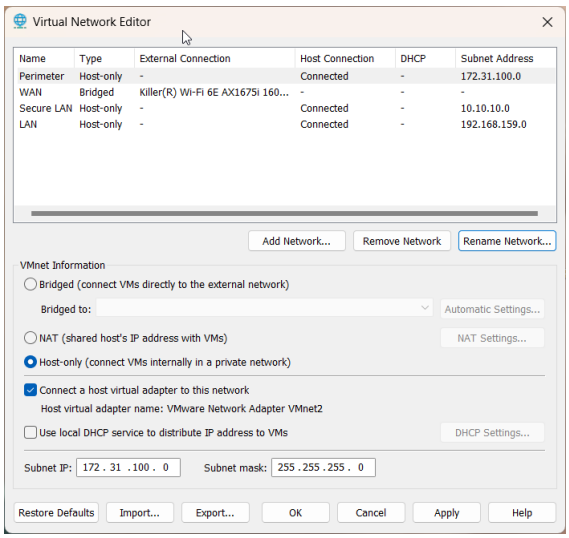

Figure 6: Screenshot: Vmnets configuration(Solvang 2023)

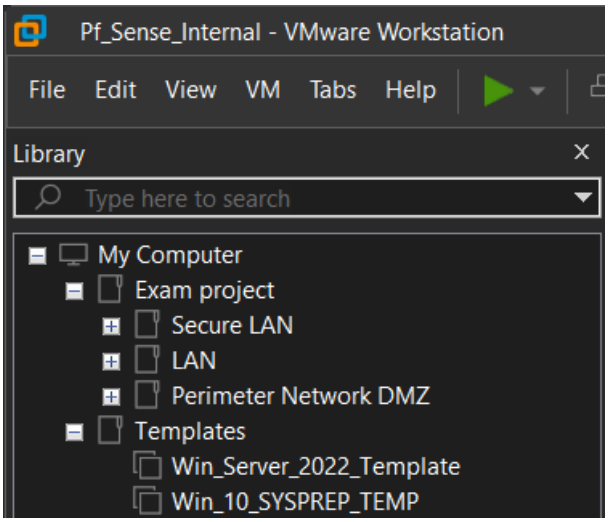

Figure 7: Screenshot: VmWare folderstructure(Solvang 2023)

- The default vmnet 0 ”auto bridge” and vmnet 8 ”NAT” are removed to allow the convenience of creating our own named network bridge and clarity. Only one bridged adapter is allowed; in this lab, we do not wish to allow NAT provided to VMware by the host to achieve full control over addressing and routing.
- At this point, no networks should be on the list. Next, create the networks as seen in figure 6. Create and bridge a WAN network to the Host WiFi or applicable NIC. Create the remaining networks as ”Host-only” networks, with /24 subnet masks, and deselect ”use local DHCP service to distribute IP addresses to VMs” on all of them. DHCP services are to be controlled by PfSense.
- Now that the network adapters are in place, the final step is to create a directory structure to contain the VMs, which reflects the network structure. This is optional and only serves to illustrate the network segmentation. This is illustrated in figure 7

3.2 Introducing PfSense and FreeBSD

PfSense is an open-source stateful firewall and router released under the Apache 2.0 license. It can be installed as a Virtual machine in a Virtual network environment but should be deployed on separate hardware from other VMs. A best practice is to deploy on bare metal (Type 1) hypervisors or dedicated hardware(Netgate 2023).

PfSense ships on a custom-built version of FreeBSD maintained specifically for PfSense, yet another open-source OS derived from UNIX(Tanenbaum and Bos 2015).

PfSense is designed to be fully managed from a Web GUI interface and offers a wide range of functionality like OpenVPN, IDS and IPS (Intrusion Detection/Prevention Systems), Proxy and Reverse Proxy, DHCP, DNS resolution or Forwarding, NTP services, routing with BGP, OSPF and even RIP.

Some are pre-installed and ready to use out of the box. Others can be easily installed with the built-in GUI package manager. Netgate also offers custom-built hardware and enterprise support for paying customers.

The first step in setting up the lab is to install and configure PfSense to enable network connectivity and a functional connection.

3.2.1 Installing PfSense

- This lab environment uses two PfSense instances that support routing and DHCP in addition to the firewall capabilities.
- The Edge router is configured with three NICs, one acting as the WAN port bridged to the hypervisor's NIC, one supporting our LAN network, and the last for our perimeter network (DMZ).
- The PfSenseCoreRouter holds only two NICs, one acting as a WAN port connecting to our Edge router and the other as a LAN port connecting to our secure network hosting ADDS.

Figure 8: Step 1: Right-click on the folder where you want to create the new VM, select ”Custom” to allow detailed configuration options during the creation process, and press Next. Screenshot(Solvang 2023).

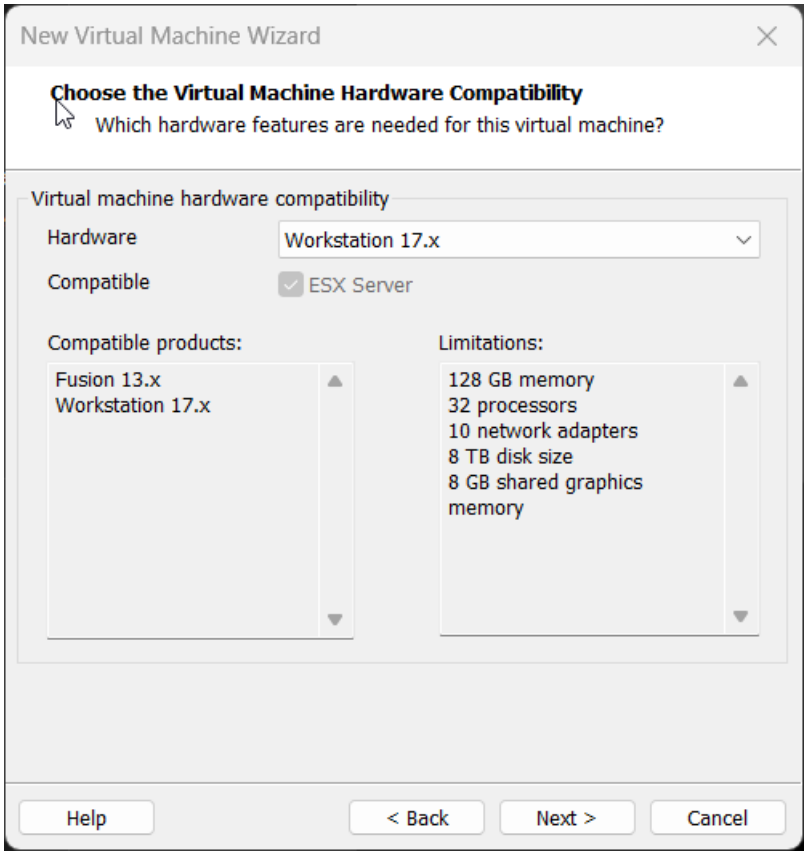

Figure 9: Step 2: There is the option to account for backward compatibility with earlier versions of VMware if needed. Select the desired option and press Next. Screenshot(Solvang 2023).

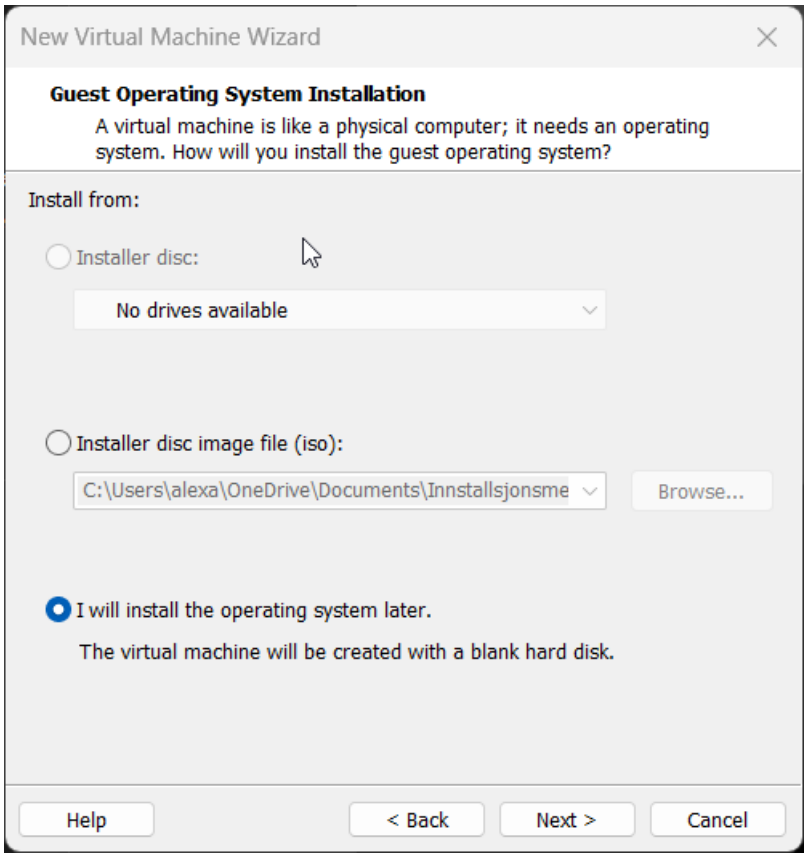

Figure 10: Step 3: Choosing the ISO you want to use to install the VM here or adding it to the virtual CD/DVD drive after the VM is created is possible. Screenshot(Solvang 2023).

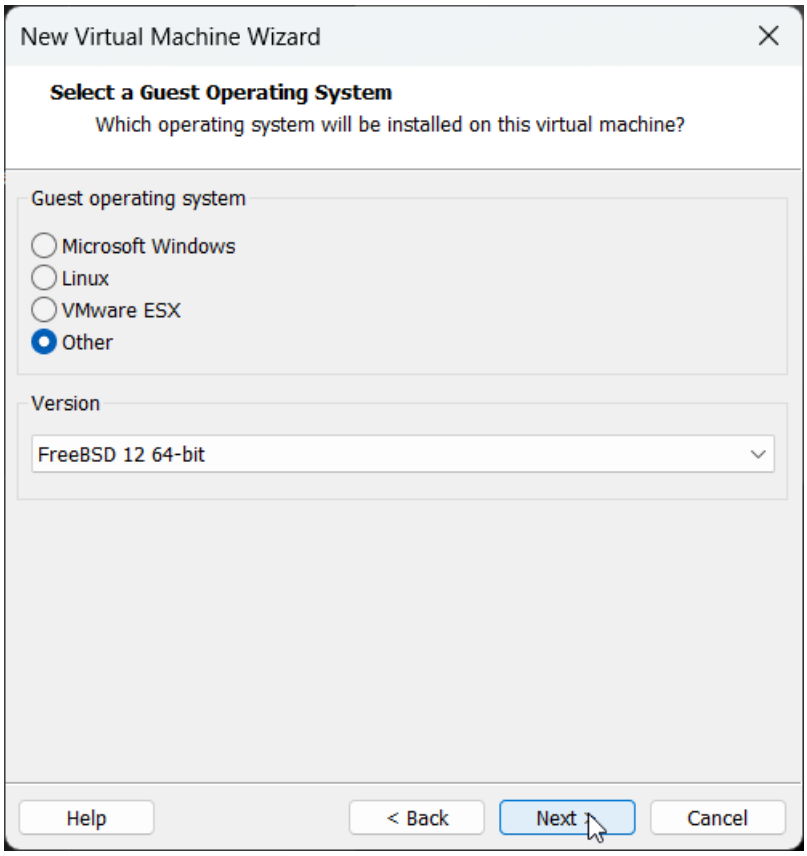

Figure 11: Step 4: PfSense runs on a FreeBSD version 12, 64-bit compatible OS. Select it from the dropdown list and press Next. Screenshoot(Solvang 2023).

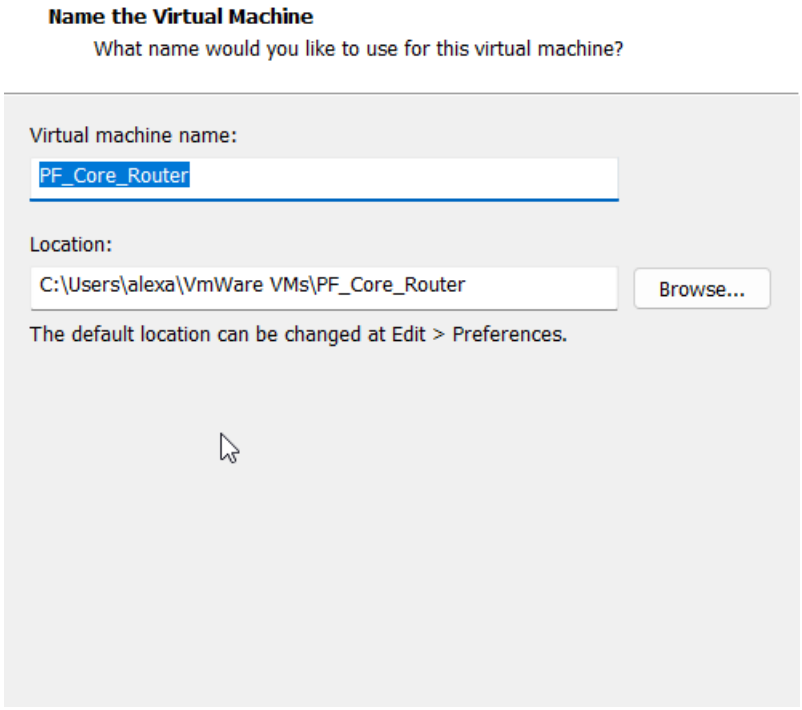

Figure 12: Step 5: Provide your VM with a descriptive name displayed in your VmWare folder structure and press Next. Screenshot(Solvang 2023).

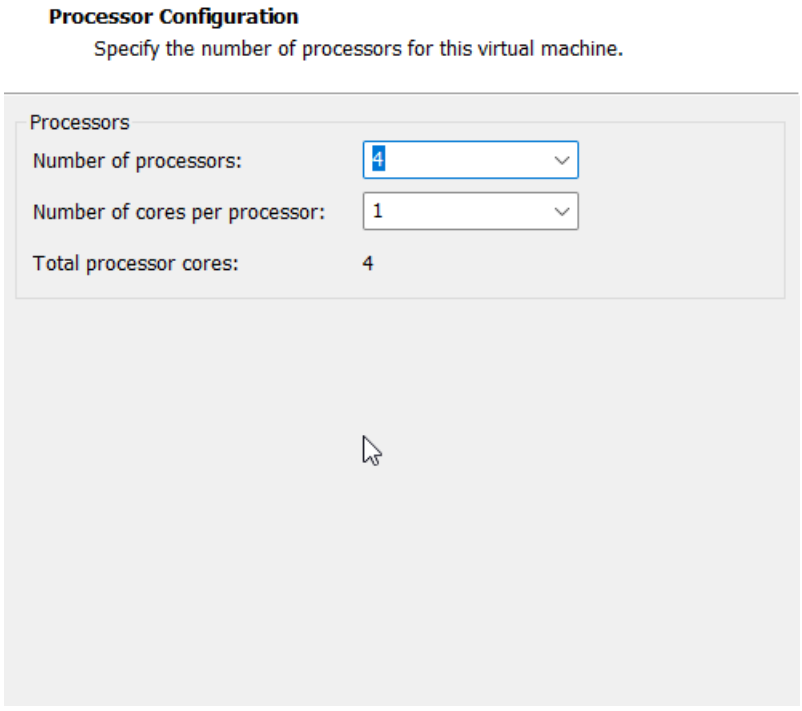

Figure 13: Step 6: Specify the number of processors and cores. When assigning hardware resources, it is important to consider the limitations of the hypervisor. During installation and configuration, it might be convenient to provide more resources for better performance and then reduce them later to a level suitable for the actual workload. Screenshot(Solvang 2023).

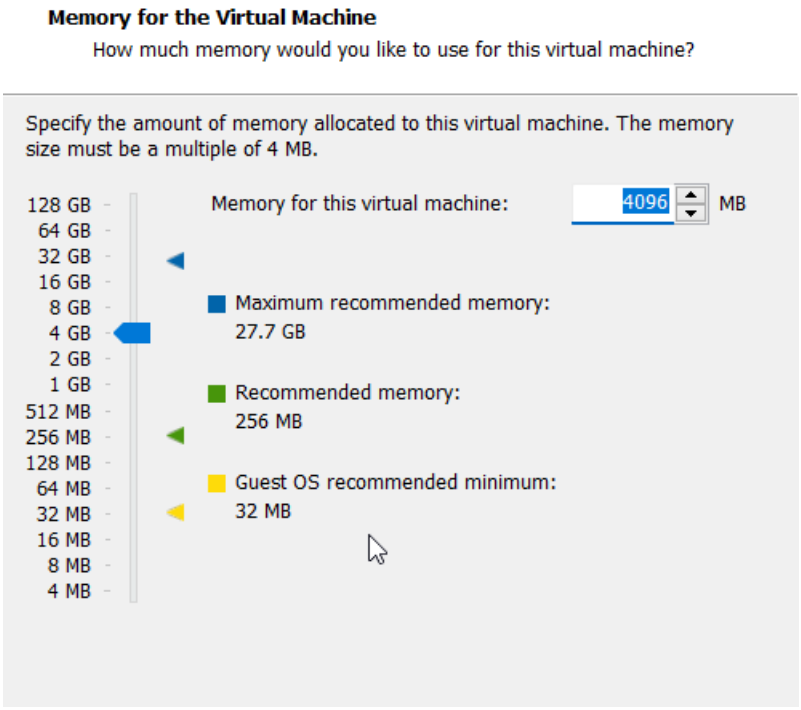

Figure 14: Step 7: Apply the same considerations presented in Step 6. Screenshot(Solvang 2023).

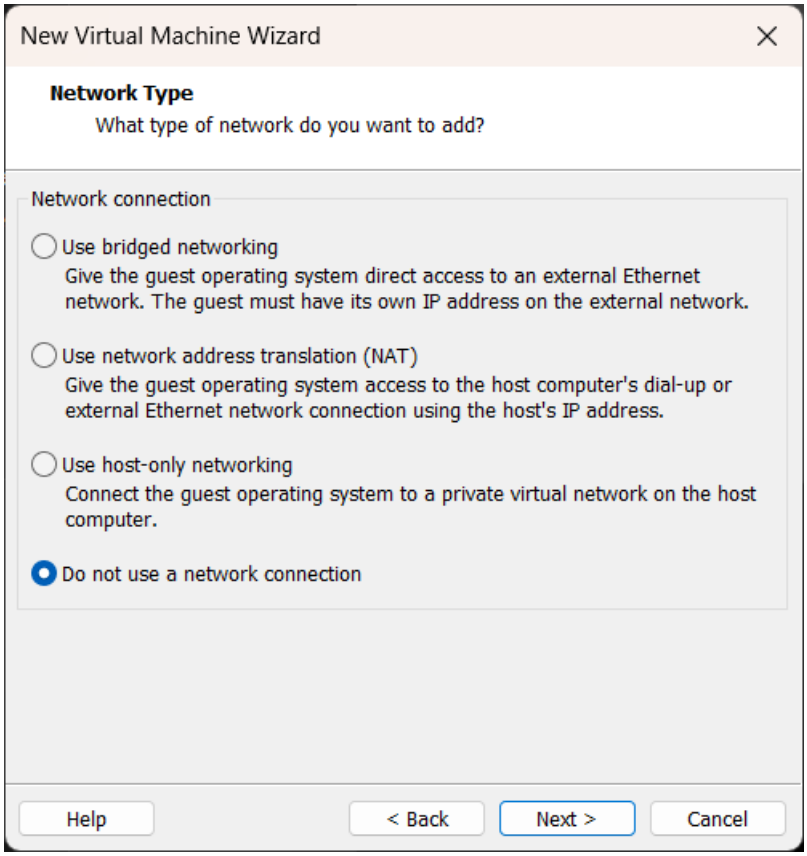

Figure 15: Step 8: Selecting either of the default network types is possible, creating a virtual NIC with default settings accordingly. Another approach is not to use a network configuration but rather configure all virtual hardware, as seen in a later step in this demo. Screenshot(Solvang 2023).

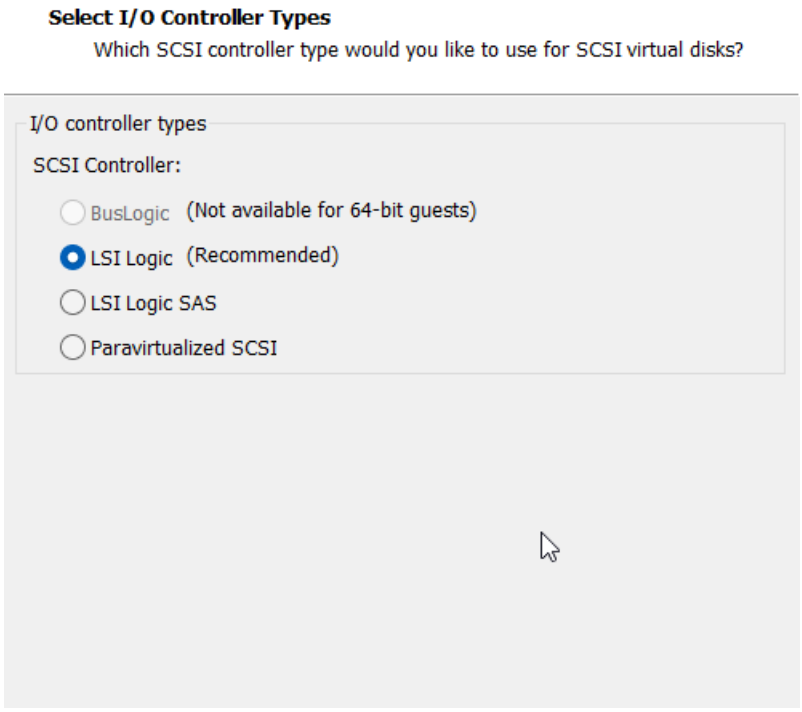

Figure 16: Step 9: Select the I/O controller type for connecting a virtual disk; the recommended option is used in this setup. Screenshot(Solvang 2023).

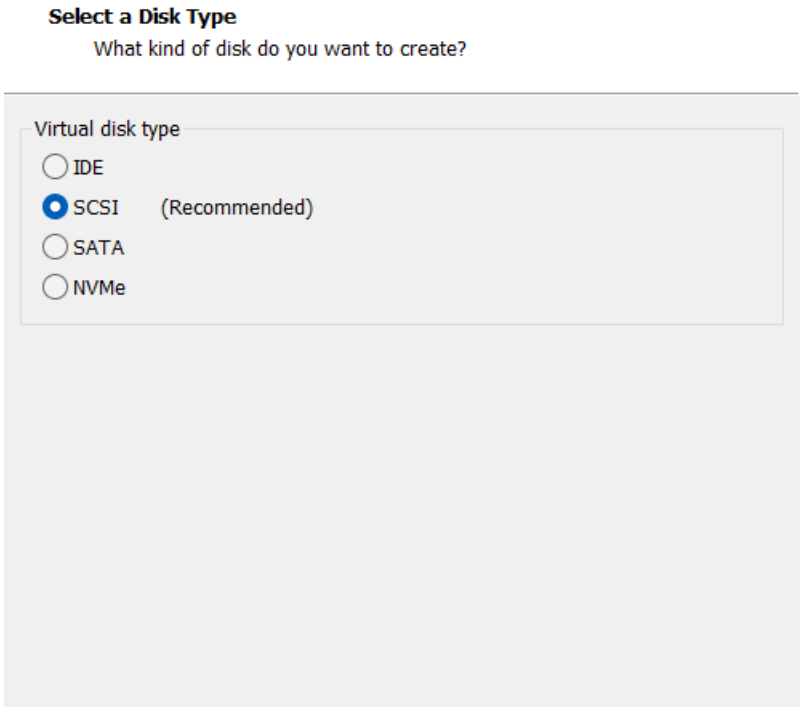

Figure 17: Step 10: Choose the virtual disk type that relates to the previous step, and again, the recommended option is used. Screenshot(Solvang 2023).

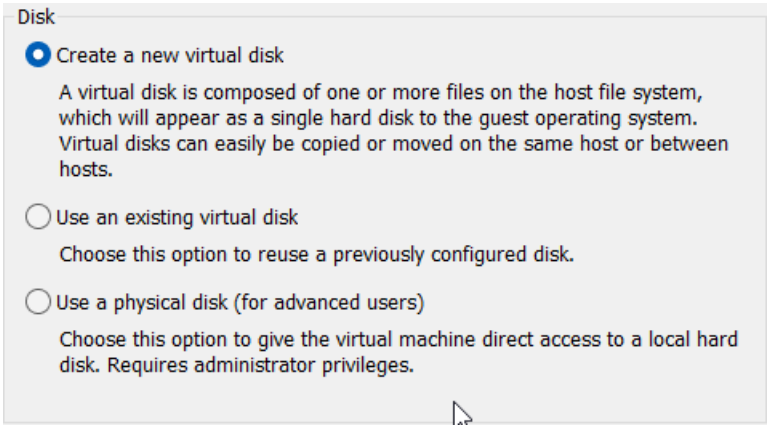

Figure 18: Step 11: One could add pre-created disks or even physical disks connected to the hypervisor; for all purposes, new disks are created in this demo. Screenshot(Solvang 2023).

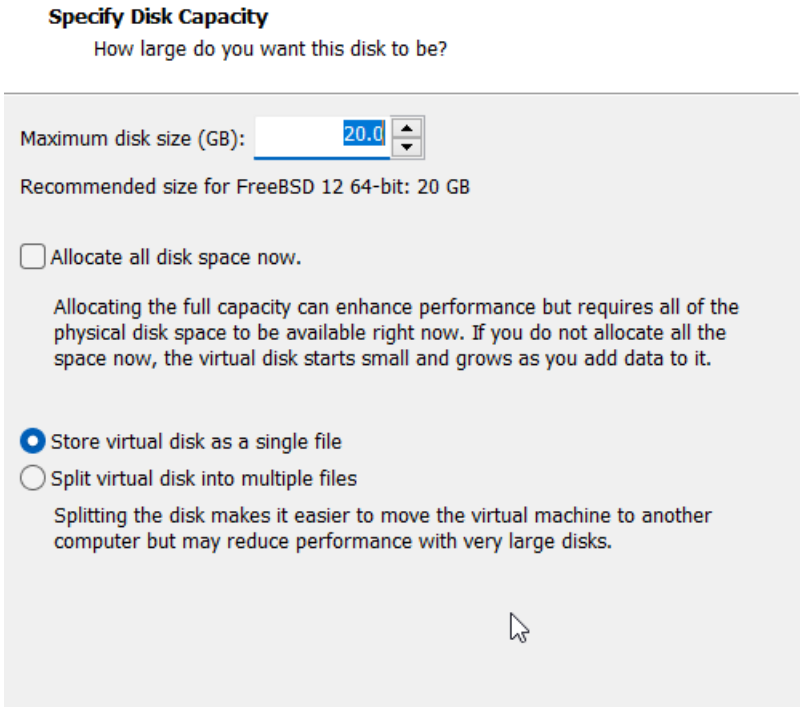

Figure 19: Step 12: Define the size of the Virtual disk and choose whether to let storage allocation take place dynamically as the file grows, thus conserving space on the physical storage available on the hypervisor or allocating the full size straight away. The disk may also be stored as a single file or multiple files. Only the recommended size and single file storage are used in this demo. Screenshot(Solvang 2023).

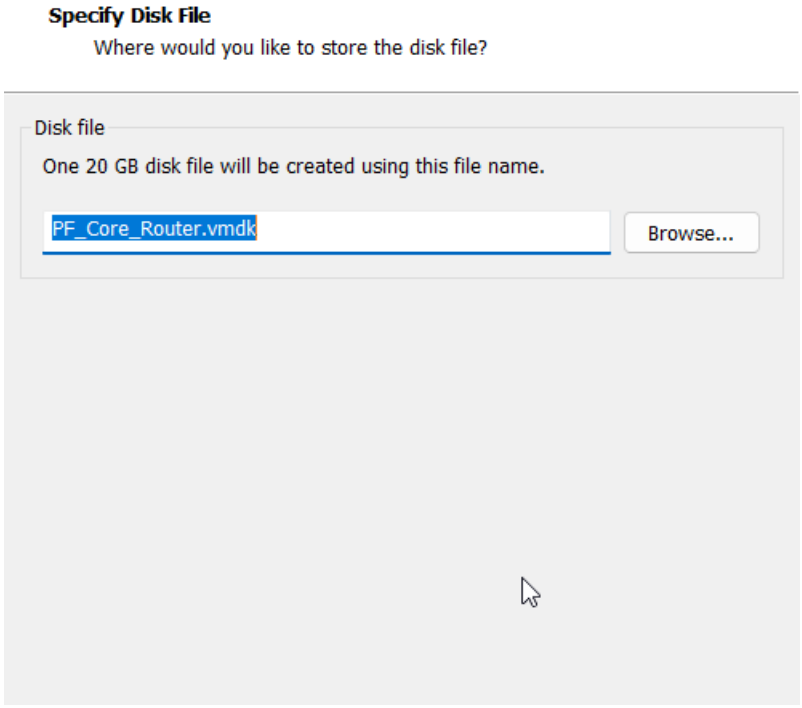

Figure 20: Step 13: Provide a filename for the disk file and select a storage location on the hypervisor. Screenshot(Solvang 2023).

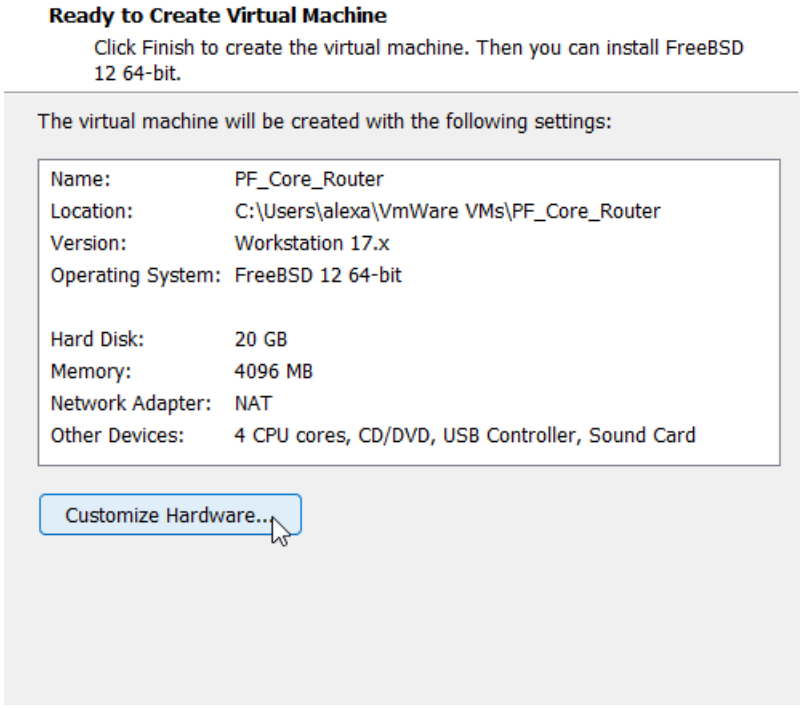

Figure 21: Step 14: The Wizard is now ready to create the VM as specified and presents an overview of the selected configurations. At this point, we can also customize the virtual hardware by pressing ”Customize Hardware.” Screenshot(Solvang 2023).

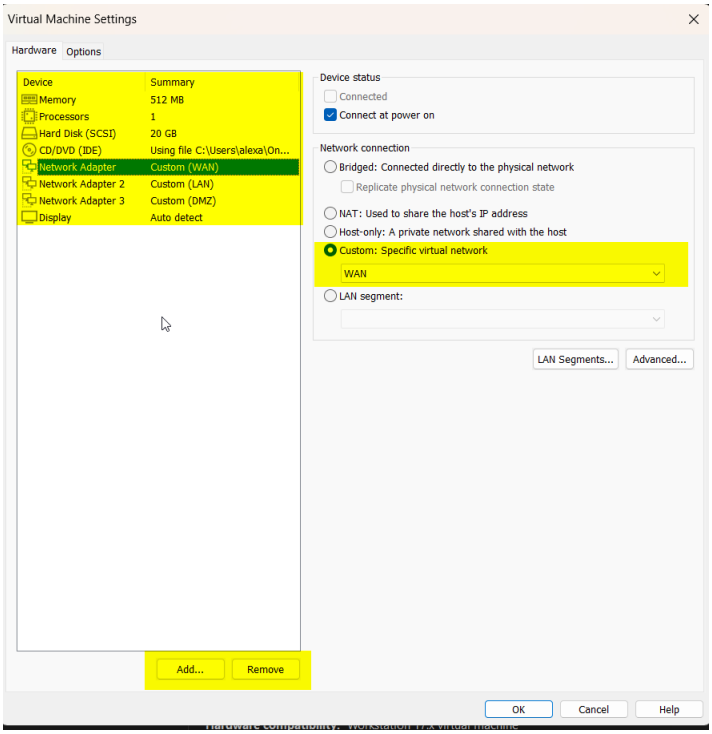

Figure 22: Step 15: Using the ”Add” and ”Remove” buttons highlighted, remove any virtual hardware not needed, then add and configure any hardware necessary for the installation. A virtual printer Sound card and USB controller are removed on this VM, while three Network Adapters have been added and attached to the appropriate VMnets as described previously 3.2.1. When the desired chances are made, press ”OK” to return to the previous window and ”Finish” to create the VM. Screenshot(Solvang 2023).

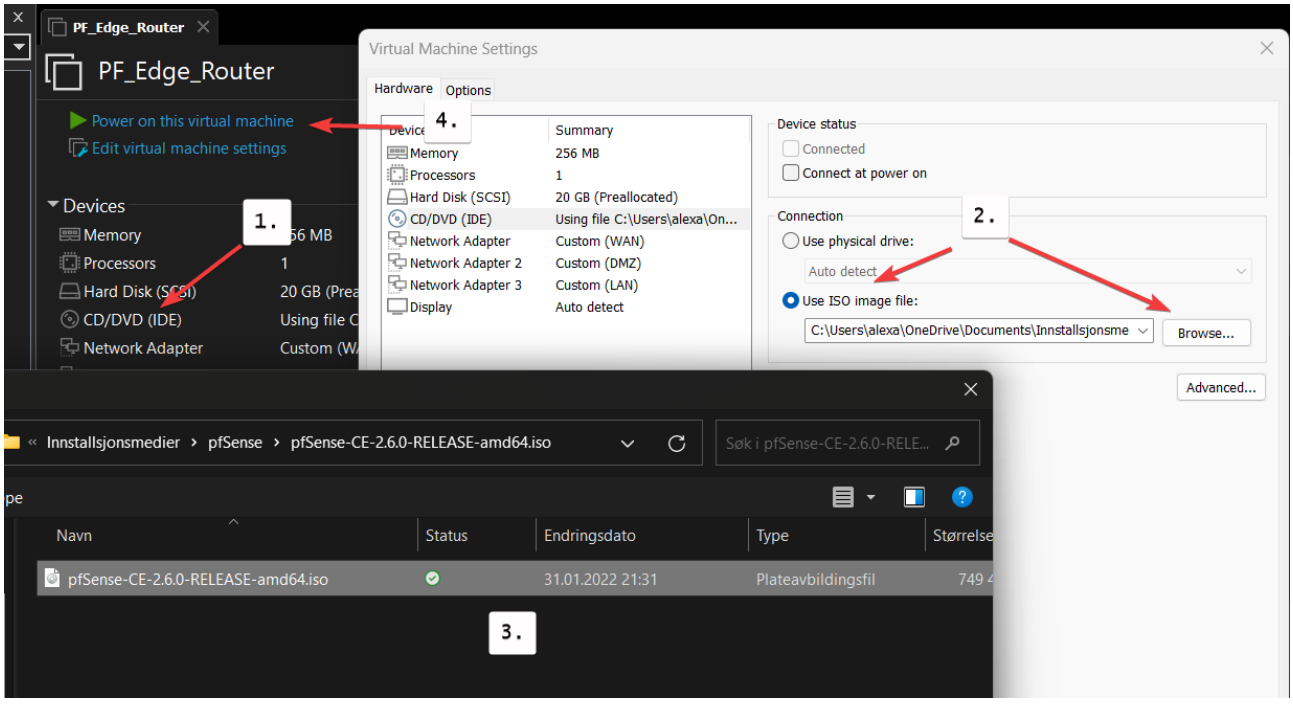

Figure 23: Step 16: Now that the VM is configured, we can install the OS. Select the VM, then select CD/DVD, and choose the ISO from your local storage, as seen in the figure above. Now, ”Power on this virtual machine” and perform the installation process as required. Screenshot(Solvang 2023).

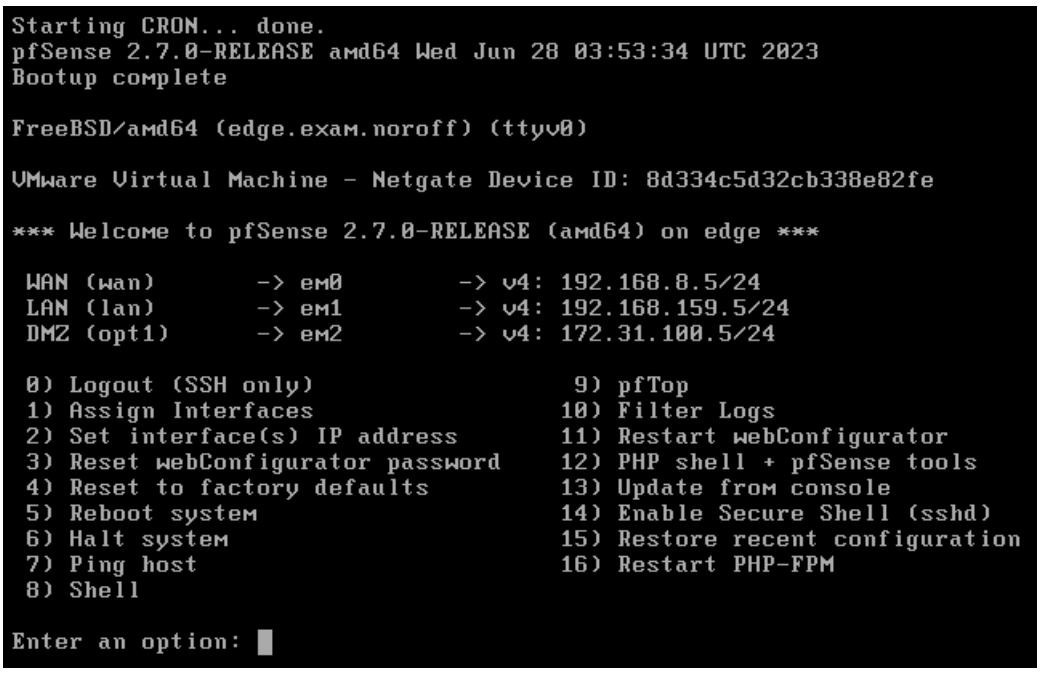

Figure 24: Step 17: The final step is to use the CLI interface on the PFSense to assign each NIC, or interface, with a static IP. The PfSense GUI is now accessible on the LAN interface IP using a Webbrowser on a Linux or Windows client connected to the same LAN. Screenshot(Solvang 2023).

3.2.2 PfSense Configuration

Firewall configurations are in no way meant to demonstrate a secure setup. The goal is to illustrate how these tools may be used and investigate a third OS from a security perspective and how it may interact with ADDS.

- **Routing:**
- The Edge router has three interfaces: one WAN and default gateway connecting to the internet, one connecting to the Perimeter network (DMZ LAN) hosting our Web Server, and the last connecting to our workstation LAN segment.
- The internal router provides one WAN interface connected directly to the LAN interface on the Edge router and one LAN interface that provides connectivity to our Secure LAN segment.
- The Internal (Core) router does not require static routes; it is set up with the GW interface as its default route.
- On the Edge router, however, routing to the Secure LAN must be provided, and we must re-configure our LAN interface as a WAN interface, and then create a static route to our 10.10.10.0/24 network. The original WAN interface with IPv4 192.168.159.5 is still the default route.

PfSense is capable of providing dynamic DNS and a DNS resolver. This lab only uses DNS forwarding with the integrated dnsamsq tool(Kelley n.d.). PfSense will forward a DNS request to all DNS servers listed in general settings and cache only the first response it receives. DC1 and DC2 are listed as DNS servers on both PfSense routers.

PfSense allows the creation of aliases when crafting firewall rules. In this demo, both DCs are included in an alias named ”ADDS,” and both the DMZ and LAN /24 network segments are

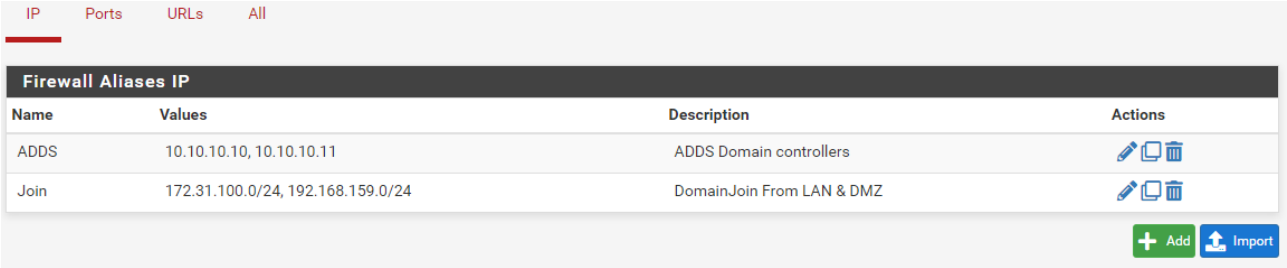

Figure 25: Screenshot: Router aliases(Solvang 2023).

included in an Alias named ”Join,” as seen in figure 25. It is a handy feature that provides oversight and efficiency.

A WAN interface is treated as insecure, so all incoming traffic is blocked. Traffic from private and so-called bogon IP address ranges are blocked on all interfaces (disabled for apparent reasons). The firewall is stateful, so all reply- and outgoing traffic is allowed on the LAN interface, while all traffic is blocked in both directions on the DMZ interface.

Every interface contains an implicit ”deny all” rule for inbound traffic and will silently drop any packets that do not match a rule that says otherwise. Rules are processed in order from the top down until they find a match. A default ”anti-lock-out” rule is automatically created when an IP address is set on a LAN interface using the FreeBSD console menu.

This rule ensures connections to the WebConfigurator GUI on the insecure HTTP port 80 from any client connected to the LAN interface. A trusted SSL certificate should be installed and the ”anti-lock-out” rule removed and replaced with one that ensures administrators can access the GUI on port 443 or via SSH on port 22 in a safe manner(Netgate 2023).

Three different kinds of firewall rules are available and listed in the same order as they are processed in the list below 3.2.2. These rules are processed until a match is found, meaning if a match is not found with floating rules, a match may still be found within interface rules. Interface group rules are not used, but floating rules are applied to WAN interfaces to prevent outgoing traffic from the Secure LAN and Perimeter LAN networks. The Lan network has no restrictions on outgoing traffic.

1. Floating rules
2. Interface Group rules
3. Interface rules

Figure 26 shows how the aliases create the ruleset required for ADDS on the Internal WAN interface(Microsoft Learn 2014). Rules are applied on every interface. On the Edge router, the perimeter LAN interface allows incoming traffic to TCP ports 80 and 443; the same rule is applied to the WAN interface.

As illustrated in figure 28, finally, the preplanned DHCP server is created on the PfSense Edge router, serving DHCP leases to the Workstation LAN clients.

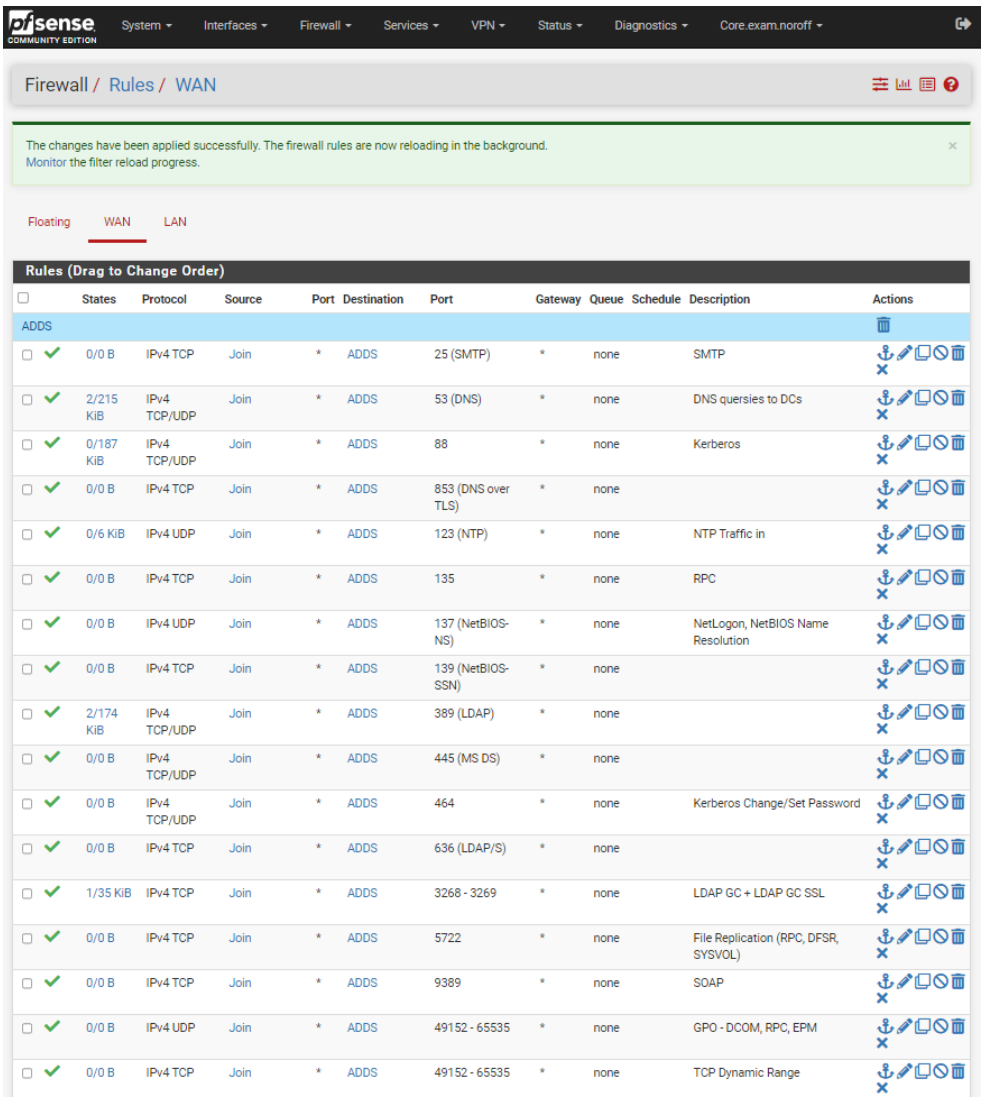

Figure 26: Screenshot: PfSense firewallrules(Solvang 2023).

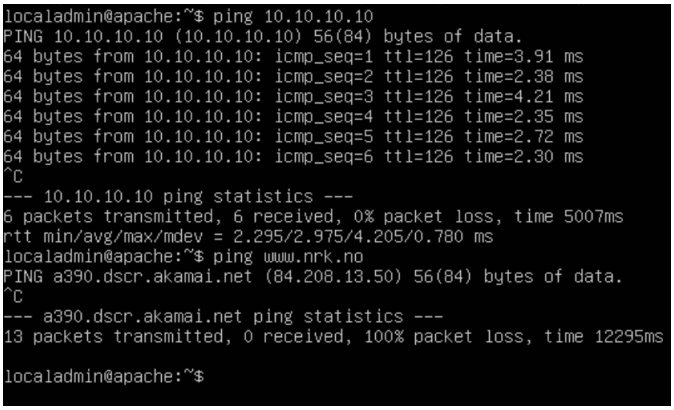

Figure 27: This screenshot shows how the floating firewall rule applied on the Edge Router WAN interface takes effect. We can ping DC1, and the DC1 DNS server also resolves the IP addresses for NRK, but all ICMP packets attempting to leave the network are dropped. Screenshot (Solvang 2023).

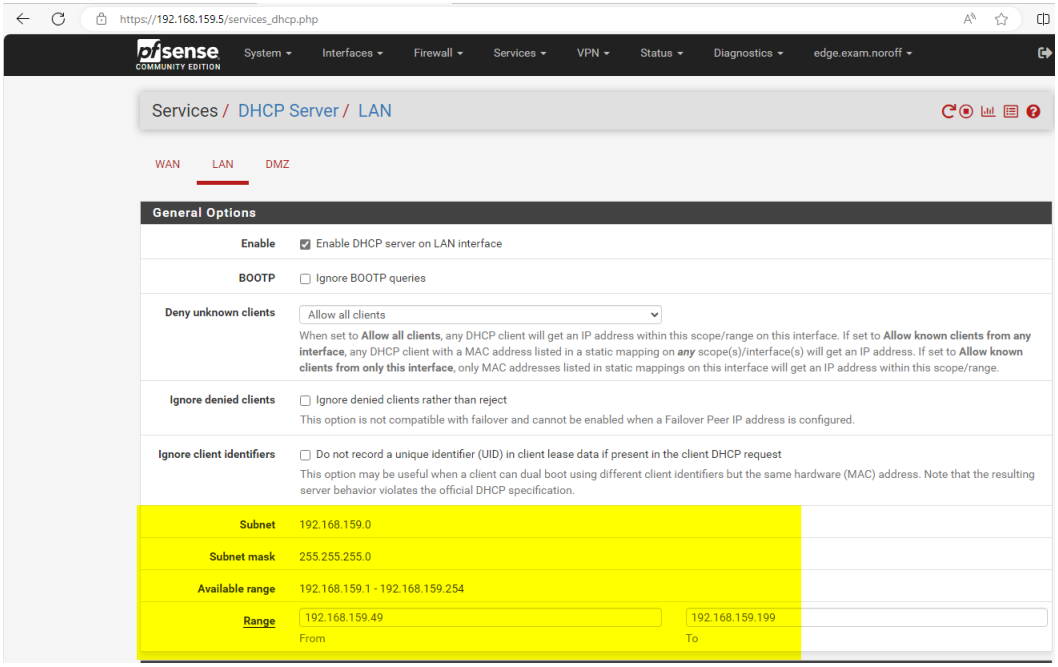

Figure 28: DHCP Server on PfSense Edge Router. Screenshot(Solvang 2023).

3.2.3 ADDS authentication

PfSense is a networking device and can not be domain-joined as a computer. LDAP and Radius, however, are both supported. Using Radius requires a Network Policy Server (NPS), so LDAP authentication is used in this lab.

- **Preparing ADDS:**
  1. Add a suitable OU, and place the following objects inside:
  2. A user account for WinBind authentication to allow LDAP connections. Set a password that can not be changed or expired.
  3. Create a security group for PfSense admins.
  4. Add PfSense administrator (user) to the security group (adding a group containing members, like Domain Admins, won't work).
  5. Ensure TCP port 389 on DC1 is set to allow connections from the outside (enabled by default).
- **Prepare PfSense:**
  1. Add and configure the authenticating server PfSense connects to as seen in figure 29. Adjust naming attributes for Hostname IP, Base DN, Bind credentials, and the Authentication Container to reflect the domain environment. The latter points to the actual users' location, not the OU for the bind user and security group.
  2. In AD, add delegated PfSense admin users to the PfSense Admin Security Group. The "alexadmin" account in the IT-Staff OU is used in this case.
  3. In PfSense, use the Diagnostics/authentication tool to verify the setup as seen in figure 30.
  4. Then re-create the "pfsense-admin" security group on the PfSense. The name must match the corresponding Security Group in AD, select the "remote" scope setting, and assign the appropriate permissions.

5. In the PfSense GUI, go to the System/User Manager/Settings tab and select the authentication server created in step 1.

- Now, the alexadmin account can authenticate using LDAP and AD.
- Note that this configuration is not secure, as credentials are transmitted over the network in plain text. Valid SSL server certificates and a trusted CA must be in place to enable LDAPS, using Bind and TLS encryption over TCP port 636.

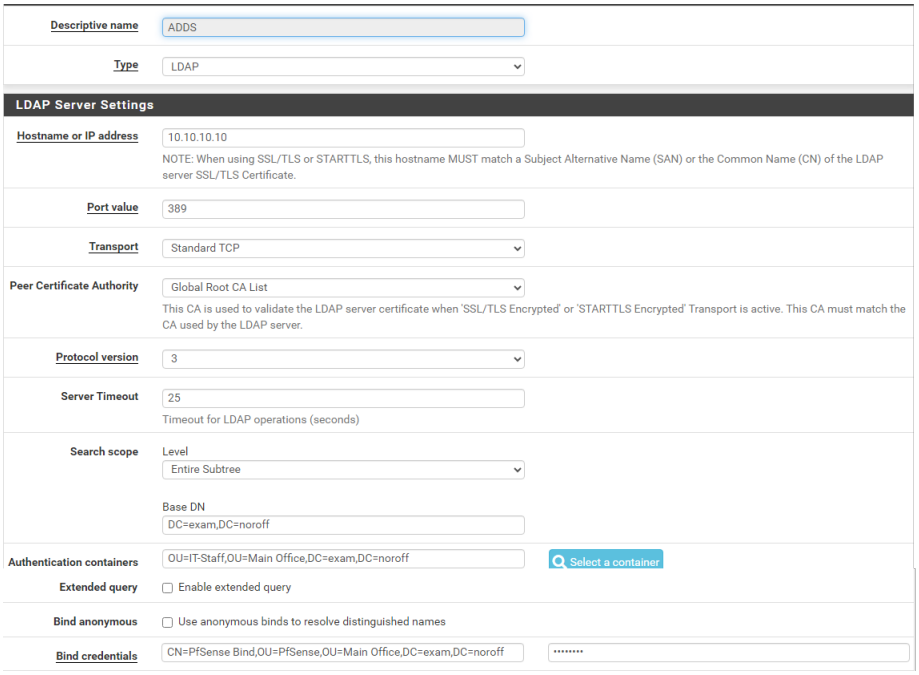

Figure 29: Screenshot: PfSense LDAP Authentication Server Settings(Solvang 2023).

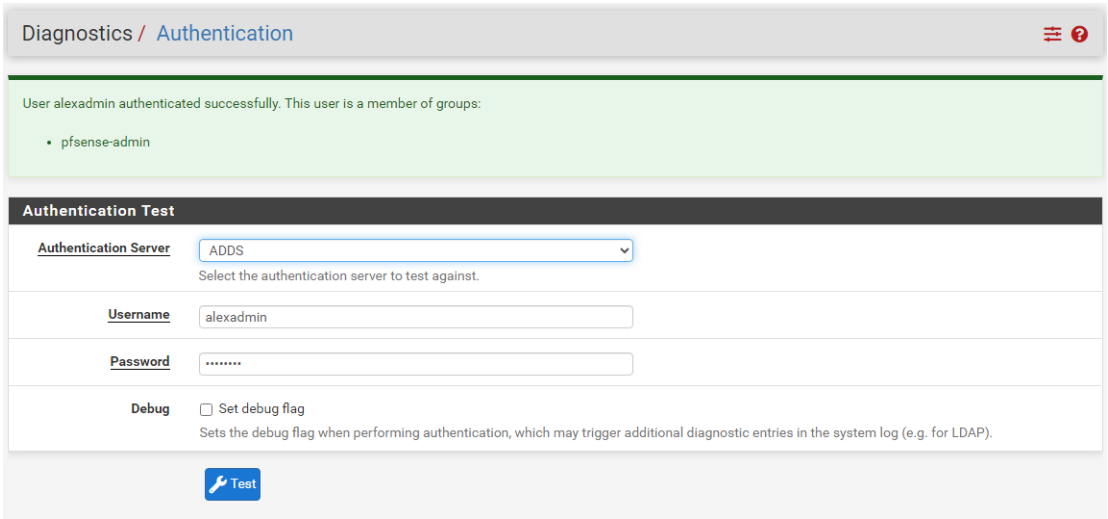

Figure 30: Screenshot: Verify Authentication Server settings in PfSense(Solvang 2023).

3.3 ADDS

The idea of a secure LAN segment is to create an extra layer of security for our most valuable assets. We already learned that domain controllers are the single most valuable targets of a malicious attack; the reason is simple: if you control the domain controller, you control every Domain joined computer, also known as ”Domain dominance” (Orin 2020).

Every operating system is initially installed, updated, and configured with a static IP, DNS, and a default GW. The correct time zone is also configured. The servers are then given an appropriate hostname and shut down to create a snapshot and conserve a clean starting point.

3.3.1 DC1 and DC2

When installing the first DC, we also created the Root Forest, and the only Domain in this deployment. With the Core installation option, we must use PowerShell commands to download and install the necessary tools and promote the domain controller. We configure the Domain Controllers to provide DNS at the same time. The whole process is done effortlessly by issuing two commands, as seen in figure 31.

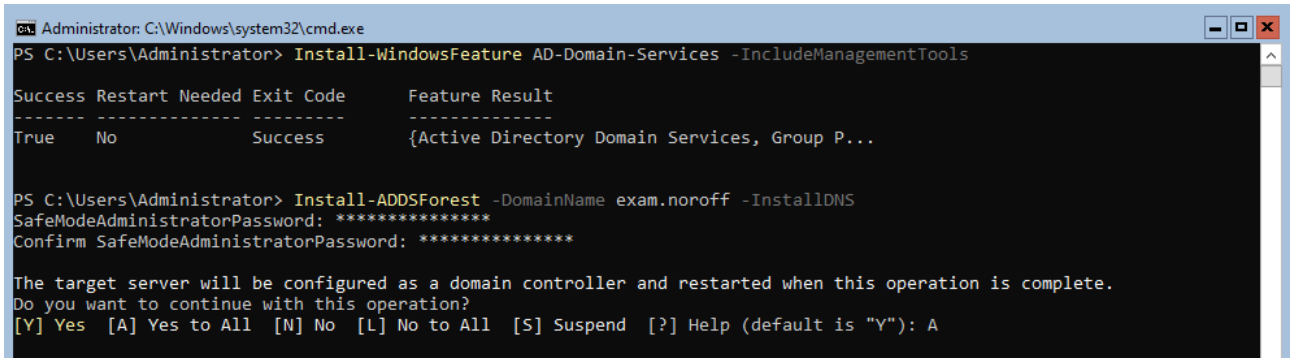

Figure 31: Screenshot: Installing ADDS and DNS(Solvang 2023)

The following DC also requires installing AD Domain Services and the necessary features and management tools. The second step is different; the Forest and Domain are already in place, and we only need to join the second DC to the existing Domain. A process that requires domain admin credentials. We can instruct Powershell to prompt us for the credentials by passing the Get-Credential command as seen in figure 32.

Two additional items are installed when the DCs are rebooted: VMWare tools and ”Server Core App compatibility features on Demand” (FOD). VmWare tools are necessary only to facilitate access to the shared folder provided on the hypervisor host and to adjust the screen size, while the FOD package allows for the use of legacy snap-in tools with MMC. The FOD tools are also required to use MMC from the PAW server; this includes, for example, the advanced firewall-, services-, event viewer-, and certificate-snapins.

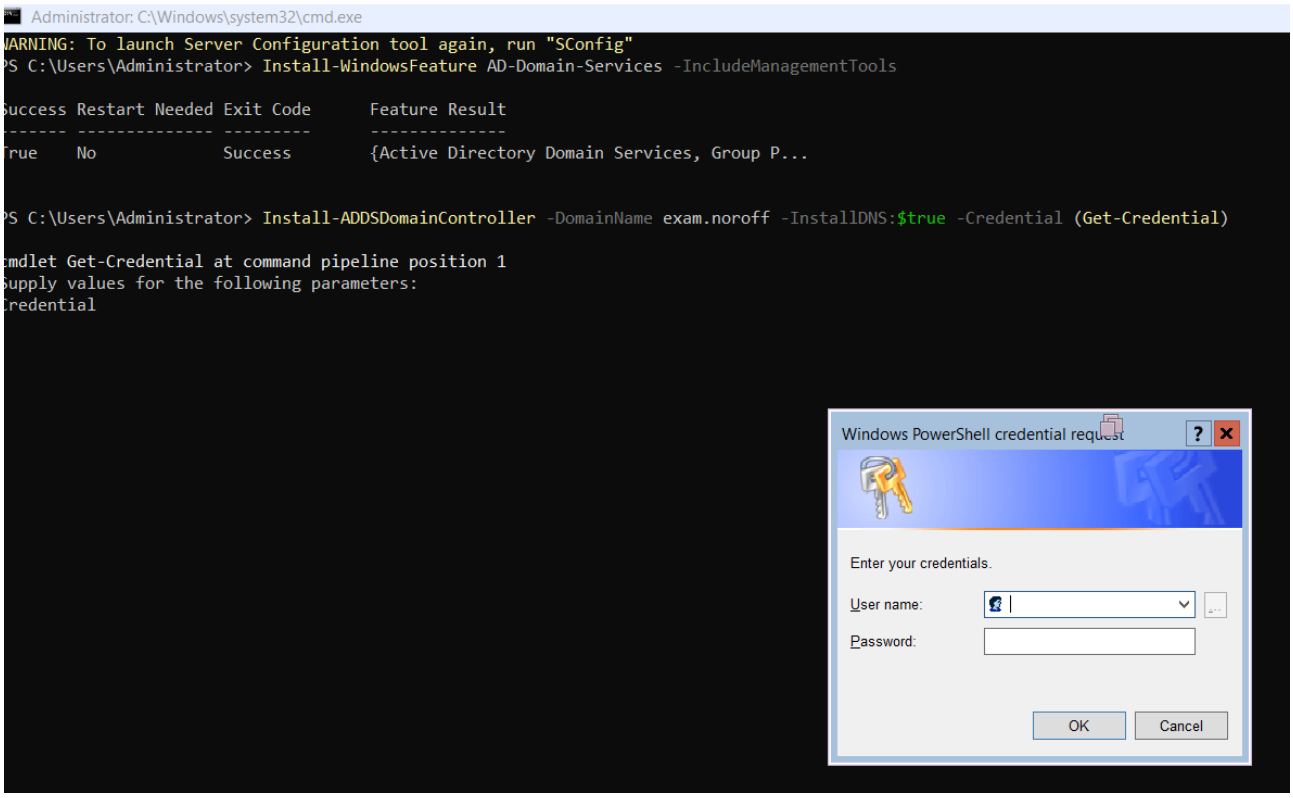

Figure 32: Screenshot: Promoting DC2 from Powershell(Solvang 2023).

3.3.2 Priveliged access workstation

To securely administrate our Domain, it is a clear recommendation to use remote administration on a dedicated secure workstation. To achieve this, a Privileged Access Workstation (PAW), a Windows server, is placed within the same secure LAN segment as our domain controllers and is also used to access and manage PfSense devices. We can now adjust firewalls to restrict access to our domain controllers to secure the environment so that only the PAW can connect to remote management ports. We can also restrict access to the PAW itself to specific user groups or roles.

Restricting access using the Microsoft Defender Advanced Security Firewall or GPOs is beyond the scope of this assignment. Still, it is recommended to ensure domain isolation and encryption of traffic flow between domain members.

As part of securing the corporate network, developing an extensive privileged access strategy is a clear recommendation. ”There is no single ”silver bullet” technical solution that will magically mitigate privileged access risks, you must blend multiple technologies into a holistic solution that protects against multiple attacker entry points. Organizations must bring the right tools for each part of the job” (Microsoft Learn 2024).

So, the PAW will, in this lab, be used as the only access point to manage high-value targets, like our AD Domain controllers and PfSense routers and firewalls. We will use our OU structure and security settings to ensure that only the right user credentials can access the PAW.

The following configurations and installations are made on the PAW post-installation:

- No roles are installed, only features, all using the ”Install Roles and features wizard.”
- Install RSAT - Remote server administration tools as seen in figure33.

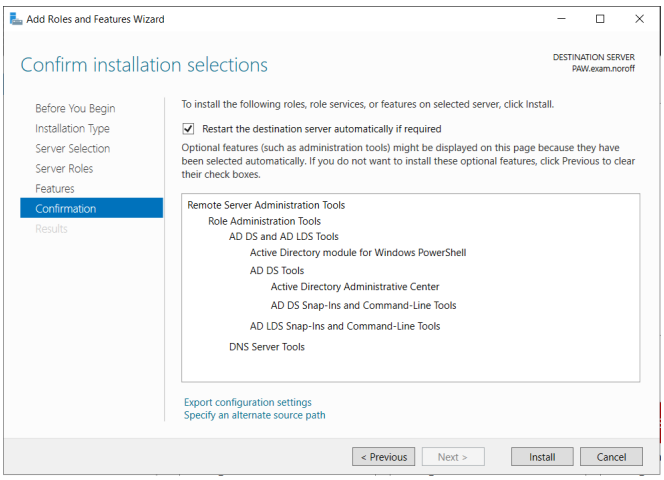

Figure 33: Screenshot: Installing RSAT on PAW(Solvang 2023).

- Install DNS Server tools.
- Install GPO management tool.
- Join the PAW to the Domain.

To use the PAW server to manage ADDS, we must log on with the user ”administrator@exam.noroff.” Now, we can start configuring our ADDS environment.

3.3.3 Configure ADDS

The following configurations are performed:

- Add Domain Controllers to PAW Server Manager as seen in figure 34.
- Configure firewall settings with Local GPO on DC1 to allow remote use of MMC FODs and computer management.
- Configure DNS reverse lookup zones for every network segment and ensure DNS works properly.
- Create and configure a GPO to ensure DC1 updates time with the PfSense NTP server. DC1 is automatically an authoritative time provider for Windows clients in the Domain, adhering to the hierarchical logic in AD.
- Create an OU and Security group structure reflecting our ”corp.”
- Create template users with different security group memberships.
- Create users in every department for testing access rights policies.
- Rename and deactivate default accounts, like administrator and guest.
- Use GPOs to ensure privileged accounts cannot log on to regular workstations.
- Restrict access to Servers on the perimeter network.

To be able to use Server Manager to manage our Domain Controllers and provide monitoring and event notifications, we must add both domain controllers to the PAW SM. Navigate to the Server Manager Dashboard, select ”Add other servers to manage,” hit the ”Find now” button, select DC1 and DC2 from the list, and press ok.

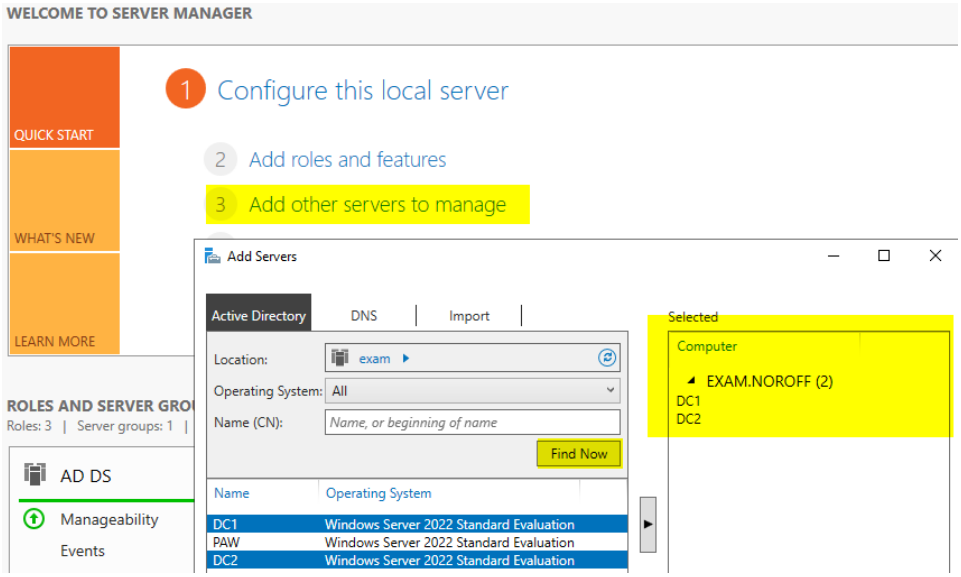

Figure 34: Add DCs to PAW Server Manager. Screenshot(Solvang 2023).

Having installed FOD on both DCs, we can now run the MMC command and access the local GPO management console on our DC1. To ensure we can perform remote administration of the DC1 server (local machine) with MMC and VMI tools using only our PAW, we must create a local GPO and provide an exception to the local firewall settings.

- Run MMC, add the Group Policy Object Editor snap-in to the console, leave ”Local computer” as the Group Policy Object as seen in figure 35, and click finish.
- Navigate to Computer Configuration – Administrative Templates – Network connections – Windows Defender Firewall – Domain Profile.
- Double click the ”Allow Inbound Remote Desktop Exception” entry, add the IP of the PAW to restrict access using this rule, and enable.

To create the Reverse lookup zones, navigate to DNS using the tools menu in Server Manager. In the DNS manager window, expand DC1 in the tree view section, right-click the ”Reverse Lookup zones” entry, and select ”New Zone”. A configuration wizard pops up; follow the steps in the wizard and make the necessary selections as needed. For this demo, we create one IPv4 Primary Zone for each network as seen in figure 36; the replication scope is all DNS servers running in this Domain, and we

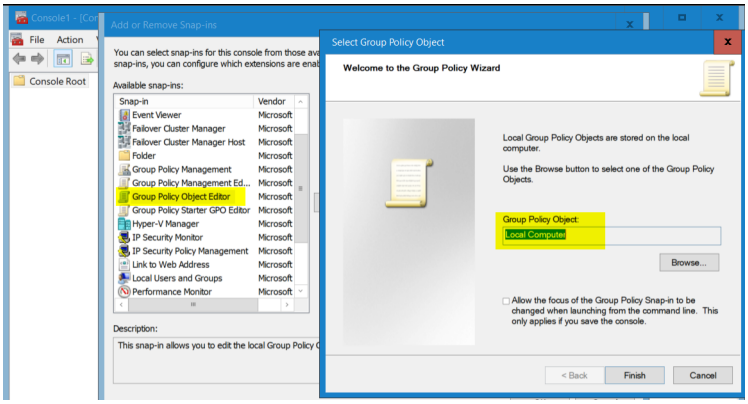

Figure 35: Screenshot: DC1 MMC for Local Group Policy(Solvang 2023).

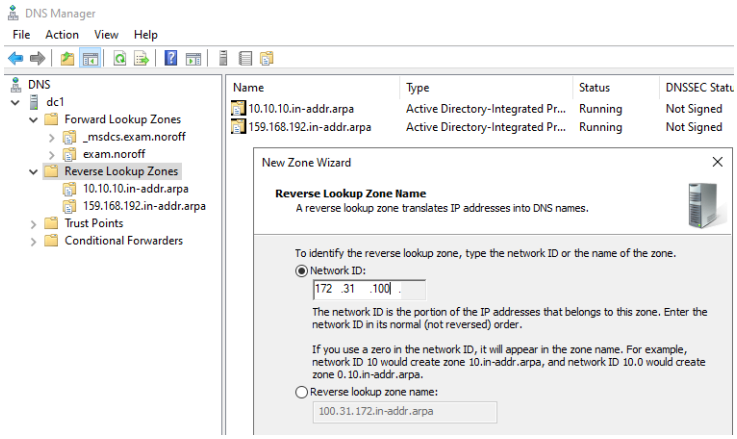

Figure 36: Creating DNS reverse lookup zones. Screenshot(Solvang 2023).

allow only secure updates. These are all Active Directory default configuration options.

In reference to our preliminary research, we already know that Domain-wide time synchronization is essential to ensure that services like Kerberos function properly, so we must ensure all clients are configured with the correct timezone and synchronize time with a reliable source.

It is recommended to sync time with multiple official time servers. Still, at the same time, we do not want our domain controller, responsible for providing time synchronization to all domain-joined computers, to send traffic out on the internet.

This is solved by allowing our PfSense firewall to provide an NTP timeserver for our DC and then using a GPO to ensure DC1 synchronizes time with PfSense. The PfSense time server is configured to synchronize with an NTP server pool. This helps to ensure availability and reliability.

- **Steps to configure GPO for DC1 NTP**

  1. Open the Group Policy Management console.
  2. Right-click the ”Domain Controllers” container and select Create and link GPO here.
  3. Right-click the GPO and select edit.
  4. Navigate to Computer setting – Administrative Templates – System – Windows Time Service – Time providers.
  5. Now, Configure Windows NTP Client as seen in figure 37 and click ”OK” to close out.
  6. One last step is to remove ”Authenticated users” and then add DC1.exam.noroff in the Security Filtering as seen in figure38.

Once the GPO is enabled, we must verify that it takes effect. Using PowerShell on DC1, first, run the command: ”gpupdate /force” to reload group policies. Use the w32tm.exe tool to verify, as seen in figure 39.

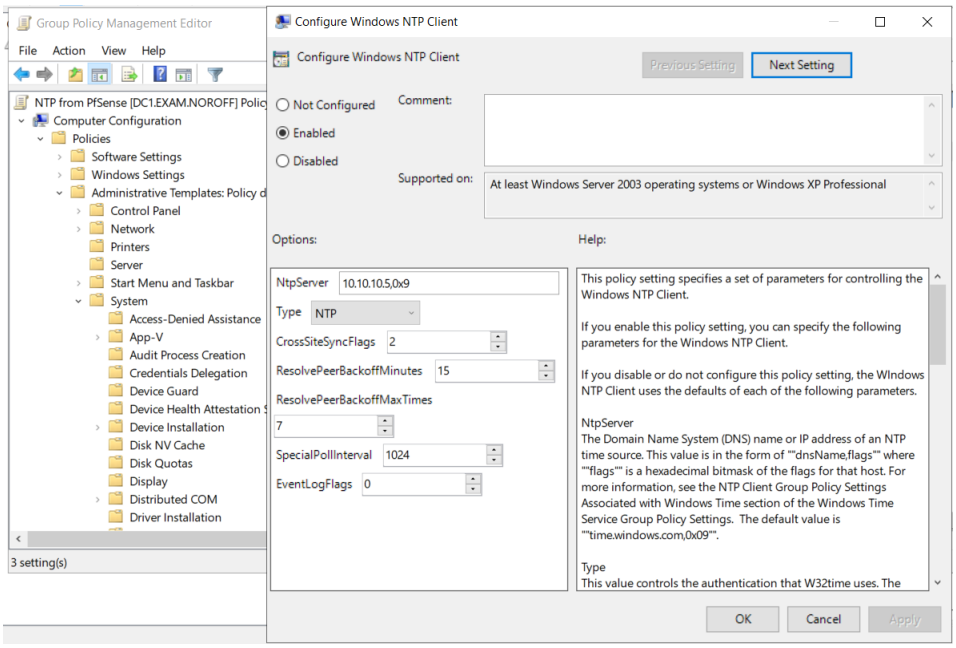

Figure 37: Screenshot: Windows NTP Client GPO settings(Solvang 2023).

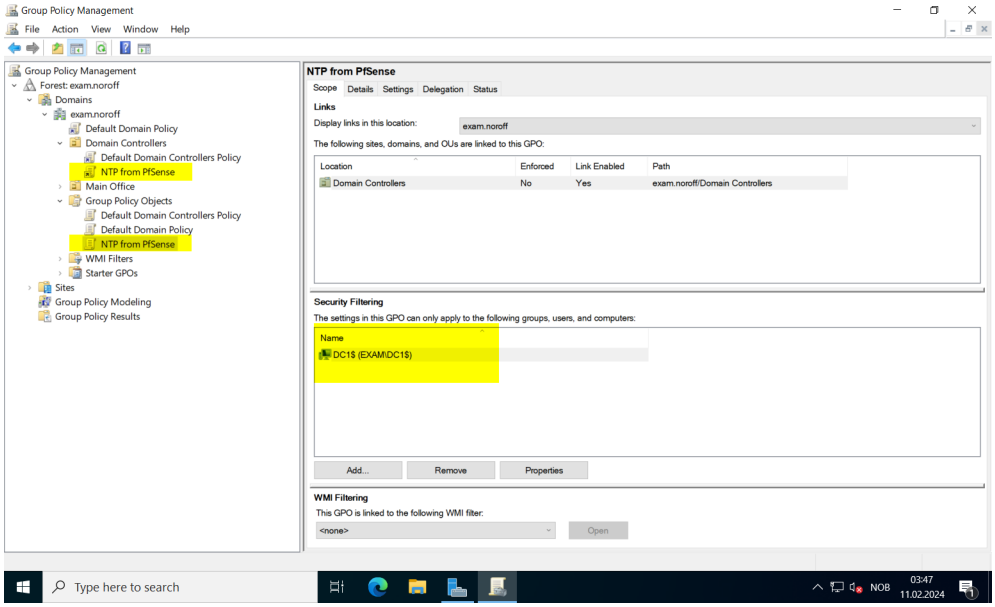

Figure 38: Screenshot: NTP GPO Security filtering(Solvang 2023).

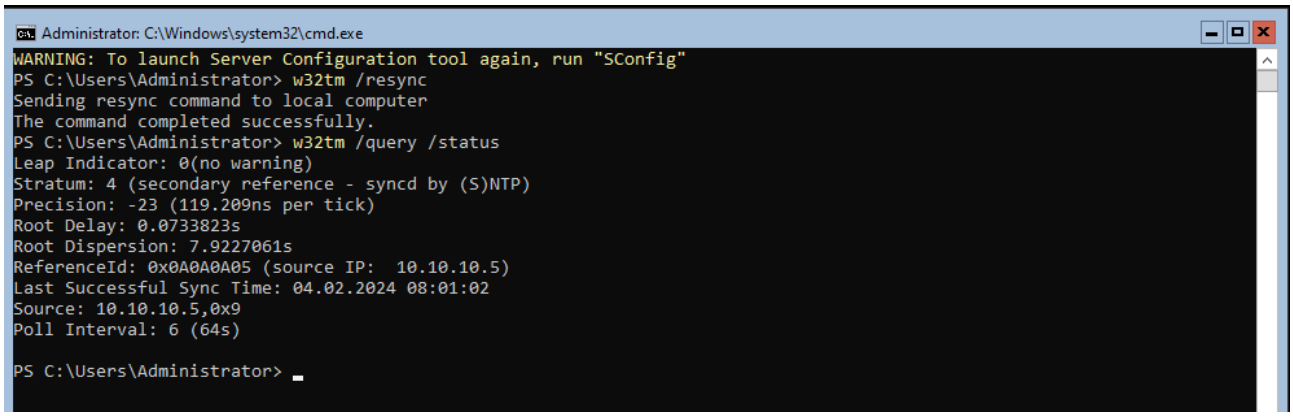

Figure 39: Screenshot: Verify correct time synchronization on DC1(Solvang 2023).

Using the AD Users and Computers console, create an OU structure that reflects the organization. First, make a top-level OU to represent the location named ”Main Office.” Next, create one OU for every department, including a Security Group for each, and an OU for PfSense and computers and servers concerning our network segments. A template ”User” is also created for every department, allowing one to apply consistent group memberships and configurations. To create new users, copy the template as seen in figure 40, which also displays the OU structure.

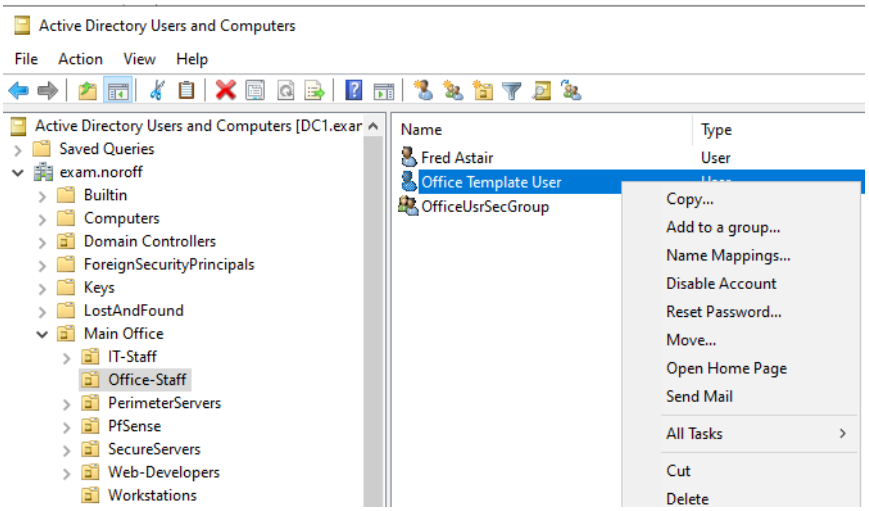

Figure 40: Screenshot: OU structure and ”copy user template”(Solvang 2023).

The principle of least privilege states that administrators should use regular accounts for non-administrative tasks. This implies that logging on to office workstations should not be possible using admin credentials. User rights assignment policies can be utilized to ”Deny Log On Locally” (Orin 2020). We also want to restrict IT staff access to the PAW. The list below shows how users are created and assigned as members to the corresponding security groups.

- **List of Users created**
- IT-Staff: alexadmin
- Office users: Fred and Alexander
- Web Developers: Karen

Hardening the Domain and servers and enhancing security can now be done by applying GPOs. A Linux Desktop client and a Windows 10 Desktop is installed in the Workstation LAN segment. Two User rights assignment policies should provide the restrictions we need.

- **GPOs are used to achieve the following:**
- Restrict the use of privileged credentials on regular workstations.
- Restrict access to Secure servers, allowing only IT Staff members local and network access.
- Disable and rename local guest accounts for all domain-joined Windows clients.
- Disable local administrator accounts on all domain-joined Windows clients.

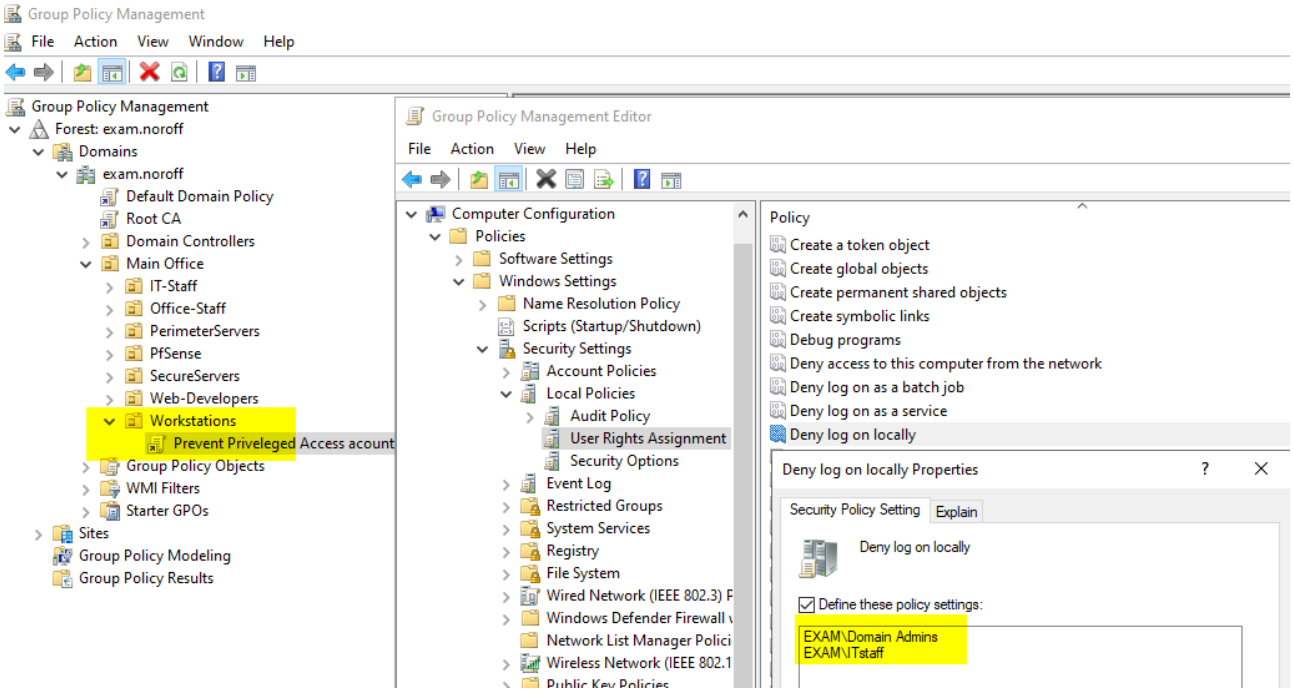

Figure 41: Screenshot: Deny Log On GPO(Solvang 2023).

The first GPO will be linked to the ”Workstations” OU and configured to ”Deny Local Log On” for the IT staff security group and the Domain Admins group. Regular Domain users, that is, all of our users not in the IT Staff group, should still be able to log on. A similar GPO with a slightly different configuration is linked to the ”SecureServer” OU.

1. Navigate to: Computer Configuration – Windows Settings – Local Policies – User Rights Assignment.

2. Locate and configure the ”Deny Local Log On” policy and add the security groups as seen in figure 41.

Keep in mind that in this network, it will only apply to the PAW. Domain Controllers are still placed in the default OU. This GPO is configured to ”Deny Local Log On” and ”Deny access to this computer from the network” and includes all Security Groups except IT Staff. If ”Allow” and ”Deny” configurations encounter a conflict, with, for example, the same group added to both, the ”deny” setting takes precedence(Orin 2020).

One GPO is created and linked to the exam.noroff Domain, on the top level, to ensure it takes effect on all Windows clients in the Domain. Note that guest accounts are disabled by default, but applying this policy will ensure that if, for some reason, the account is enabled. It will then be disabled again at the next reboot. Testing revealed that the local administrator account could not be disabled if no other user account is a member of the local administrator’s group.

Testing the GPOs on the Windows 10 workstation was a success. The IT staff member ”alexadmin” was denied, and the attempt to log on produced the message seen in figure 42, while user ”Fred” could log on successfully. Figure 43 shows that the admin and guest accounts are renamed and disabled as expected.

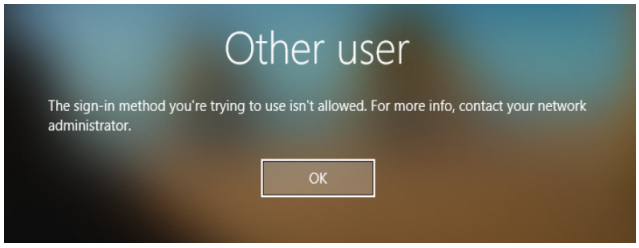

Figure 42: Screenshot: IT Staff user denied Local Log On(Solvang 2023).

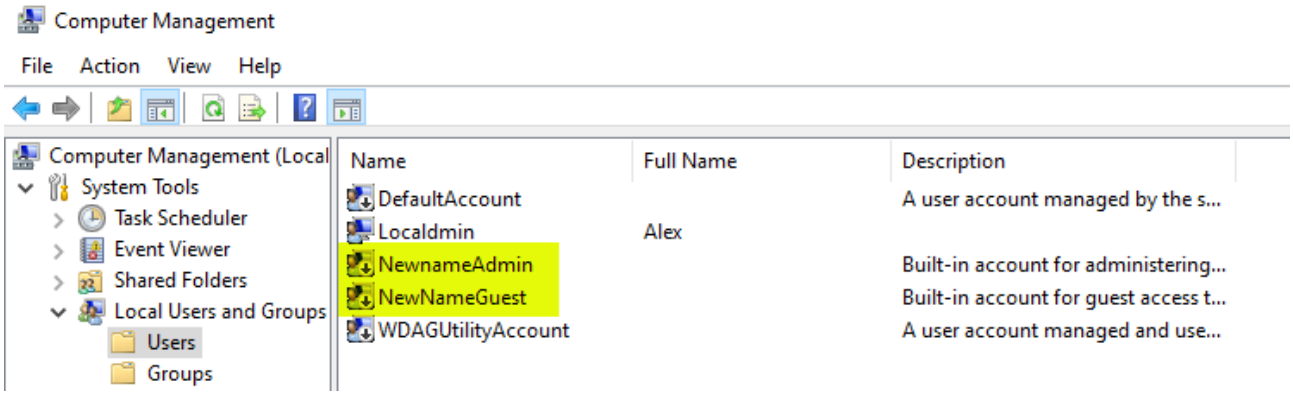

Figure 43: [Screenshot: GPO renaming and disabling local Guest and Admin accounts(Solvang 2023).

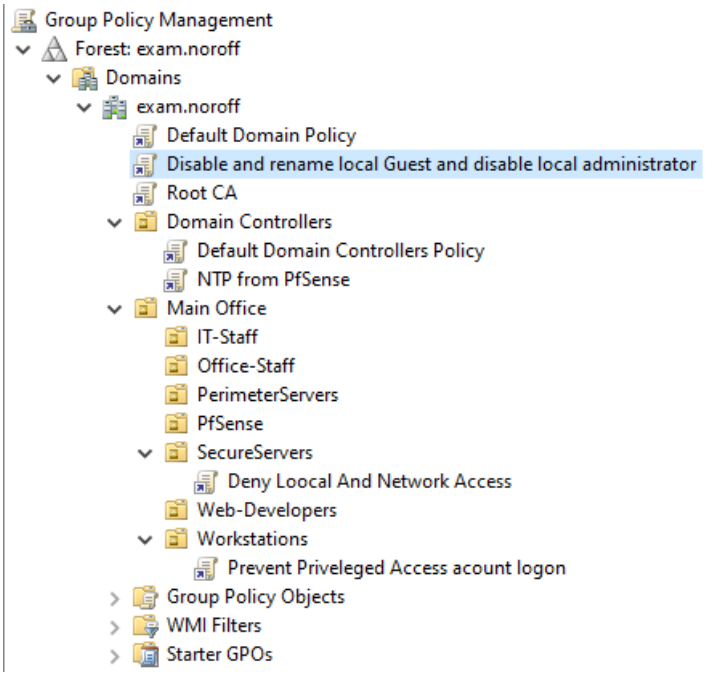

Figure 44: Screenshot: Overview of AD GPOs(Solvang 2023).

On the Ubuntu desktop client, the IT staff member could still log in. Consulting sssd-ad manpages, it became clear that SSSD consults policy security filtering before performing access control when a user attempts to log in.

If the GPO is to take effect, either the specific user or at least one of the groups the user is a member of must have either read access or ”apply Group Policy” permissions on the GPO itself. We also learn that default AD user groups such as Domain Admins are not supported(Ubuntu Manpages n.d.[a]).

The IT Staff group is added to the Security Filtering on the Workstation Deny Local Access GPO, but the alexadmin could still log on. An additional step was required. SSSD tools include sssdctl, which can remove the cache with the following command: ”sudo sssctl cache-remove” as seen in figure 45. The issue is now solved, and the Deny GPO is working also on the Ubuntu client. An attempt to log on with a member of the IT Staff security group failed, as seen in figure 46.

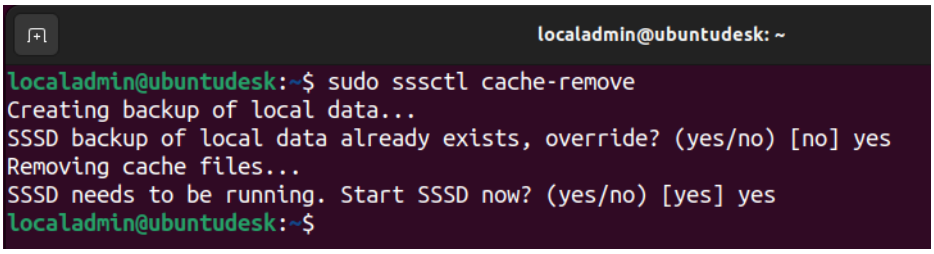

Figure 45: Screenshot: sssdctl remove-cache(Solvang 2023).

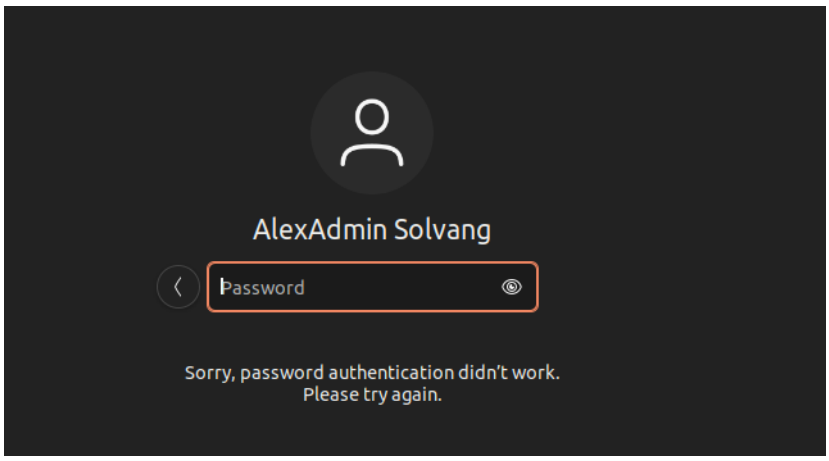

Figure 46: Screenshot: IT Staff denied Log On to Ubuntu desktop(Solvang 2023)

We have now proved that AD GPOs are powerful tools that, if carefully crafted, greatly enhance security in a Domain network. We can effectively control Local Computer access on both Windows and Linux clients.

To investigate further, we install the Adsys package on the Ubuntu Client. Adsys provides two files to be placed in a Central Policy store on the AD DC. This requires the creation of a Central Store for Group Policy on the domain controller, then, ”Administrators can configure policies by using the language-specific .adml files and the language-neutral .admx files” (Microsoft Learn 2023a). The central store is easily made by creating a directory in the right location, as seen in figure 47.

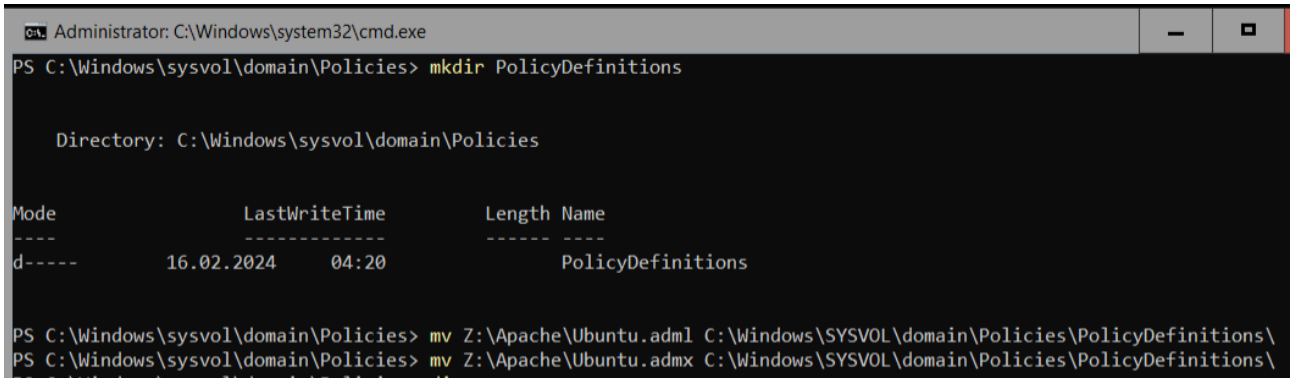

Figure 47: Screenhot: Create a Central Store for Group Policy(Solvang 2023).

Ubuntu Adsys provides both file types, which can be downloaded to your current directory by running the command ”adsysctl policy admx lts-only.” Move or copy the files to the policy store.

Canonical specifies a directory structure, where the adml file must be placed in a subdirectory created inside the policy store named ”en-us”. In contrast, the admx file remains in the store itself. Once Adsys installation and AD configuration are done, we can see the new GPO Administrative templates in AD 48.

The first apparent problem was that the Domain User log-in process now fails on the Ubuntu client. Checking journalctl, it is clear that the GPOs cannot load at boot time.

Official documentation for Adsys is scarce, but several postings addressing the issue are found on Ubuntu Ask and GitHub forums. Canonical also informs in their whitepaper that a pro subscription is required for several functions(Canonical 2022).

Another drawback becomes clear: Adsys still has a way to go compared to Microsoft native GPOs. The GPOs for Ubuntu offer only a few tools, mainly related to adjusting the desktop experience, and only a few address security, none applicable to this lab. Removing the Adsys package from the Ubuntu Client restores authentication.

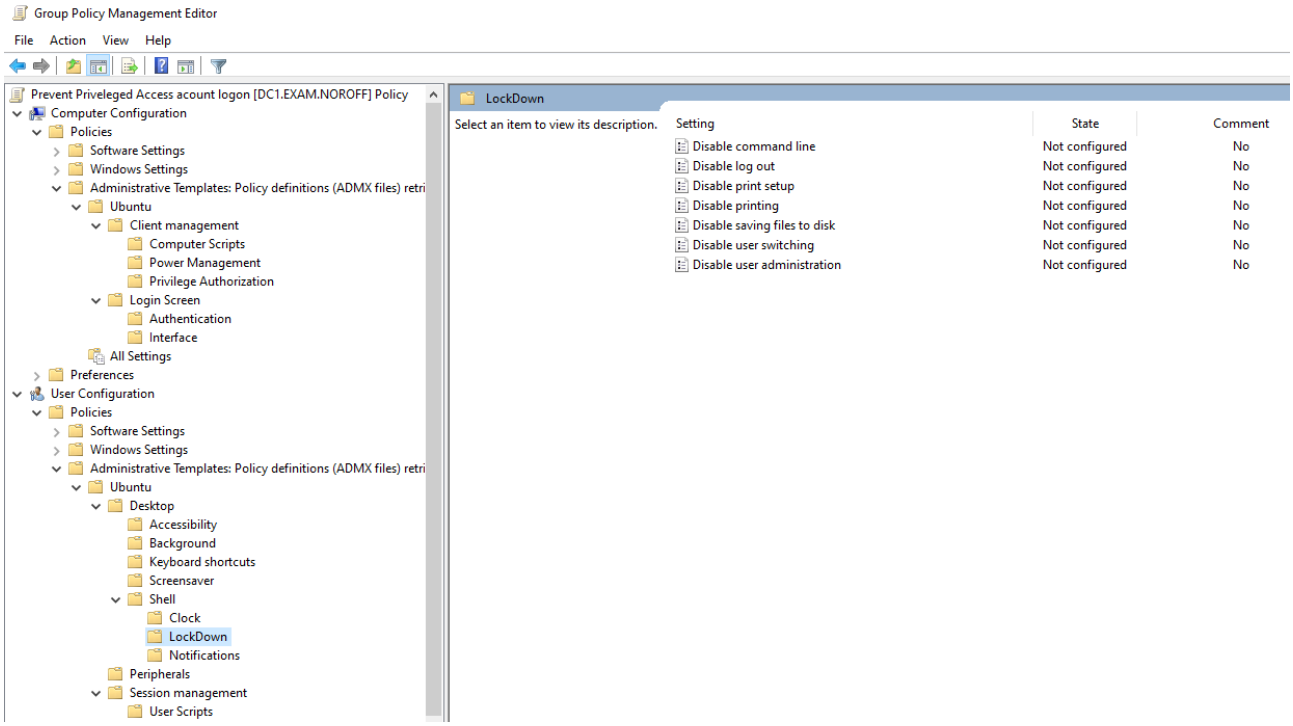

Figure 48: Screenshot: Ubuntu Administrative Templates in AD GPO

3.4 Ubuntu Desktop installation and configuration

The Ubuntu Desktop client is initially used to test AD integration, as joining the Domain on the fly during installation is possible and seems easy enough. During installation, in the section where local admin credentials are created, tick the box ”Use Active Directory,” as seen in figure 49. In the next step, the wizards ask for Domain Admin credentials and information and perform a connection test, as seen in figure 50.

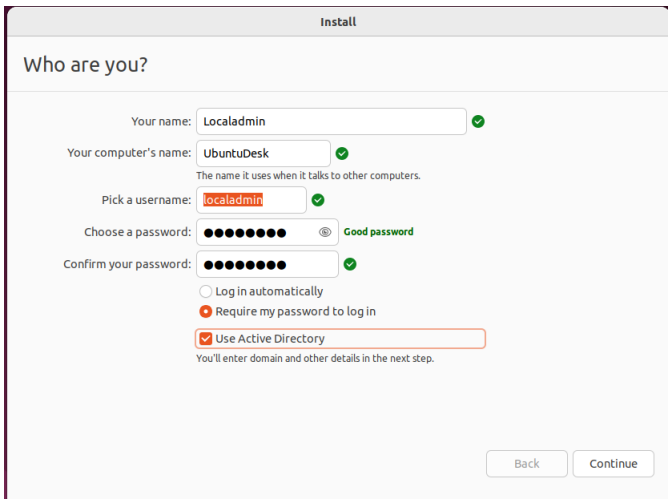

Figure 49: Screenshot: Tick to Use AD when Installing Ubuntu Desktop(Solvang 2023).

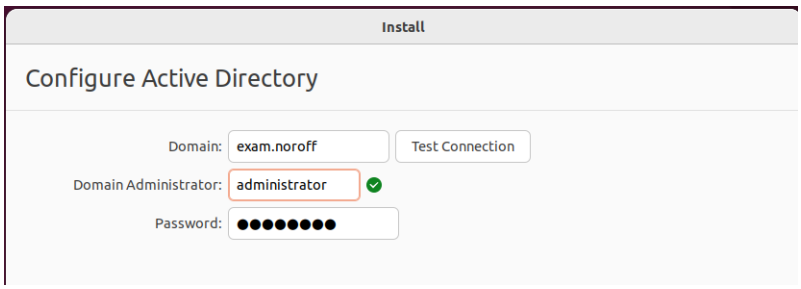

Figure 50: Screenshot: Test Domain connection during installation(Solvang 2023).

When the installation process is complete, the Ubuntu Client becomes a Domain member in the Active Directory. However, logging on with Domain users often fails, even though some attempts are successful, and the process is too slow. The following steps are taken to ensure that the host is properly configured:

- Set timezone using timedatectl
- Ensure time is synchronized by adding our NTP server to /systemd/timesynd.conf

1. Run command: timedatectl list-timezones — grep Oslo
2. Run Command: timedatectl set-timezone Europe/Oslo
3. Edit file: sudo nano /etc/systemd/timesyncd.conf – Edit as seen in figure 51
4. verify with: timedatectl timesync-status 52
5. Verify FQDN with hostname 53
6. Verify DNS with: dig dc1.exam.noroff 53

```
[Time]
NTP=DC1.exam.noroff
FallbackNTP=DC2.exam.noroff
#RootDistanceMaxSec=5
#PollIntervalMinSec=32
#PollIntervalMaxSec=2048
```

Figure 51: Screenshot: timesyncd.conf(Solvang 2023)

With these configurations in place, new tests are made to log on with AD Domain Users, but it is still unstable and slow. To determine the cause, a more verbose log output is enabled by entering ”debug-level 10” for NSS, PAM, and domain/exam tags in the sssd.conf file. This provides additional output in log files found in /var/logs/sssd.

Another debugging tool used is journalctl, which indicates problems with NSS and PAM. These are now socket-activated and should no longer be specified in the SSSD.conf file as services. The log files for sssd, nss, pam, and krb do not provide any specific clues, only general log-on error and log-on success messages.

Using tools like getent username, groups username, and specifying domain users all indicate identification and authentication works fine, but the problem of slow processing and repeated log-on failures remain. The next step is to use Wireshark to see what happens when a log-on is attempted.

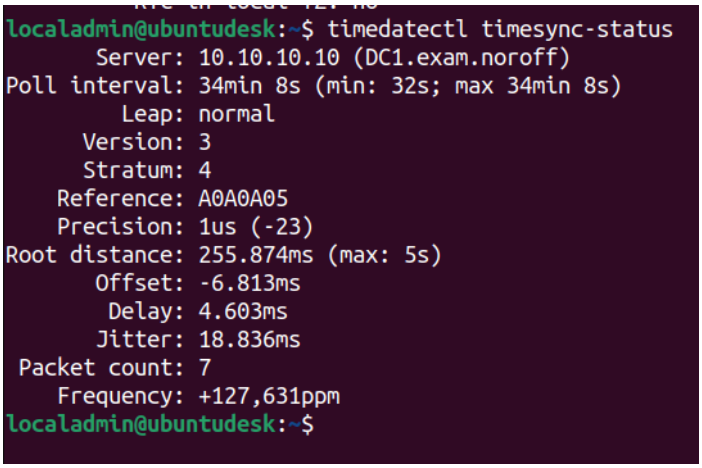

Figure 52: Screenshot: Verify timesync-status(Solvang 2023)

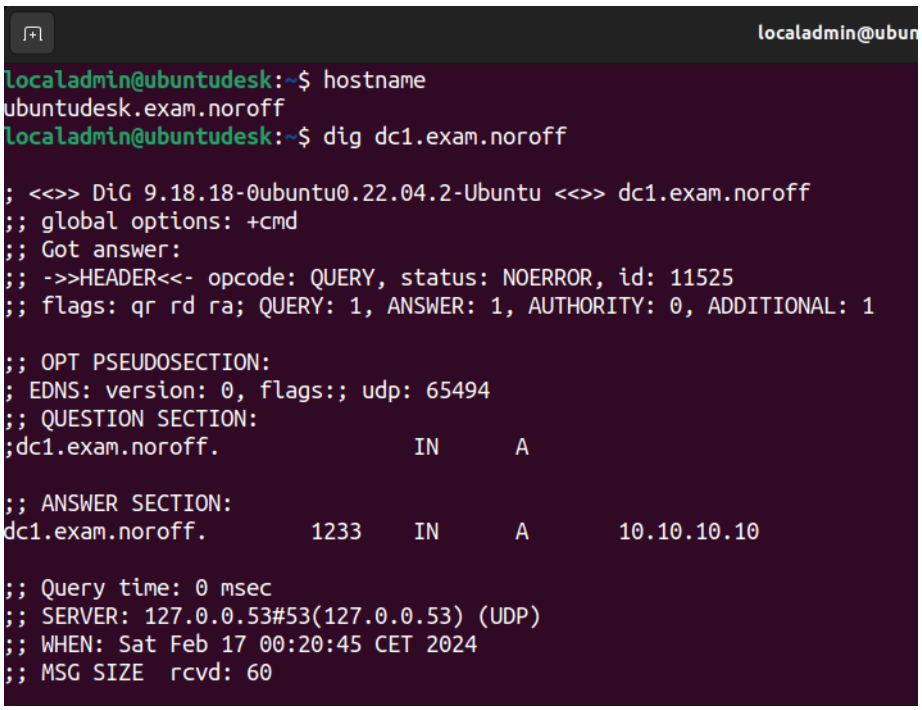

Figure 53: Screenshot: Verify FQDN hostname and Reverse DNS lookup(Solvang 2023).

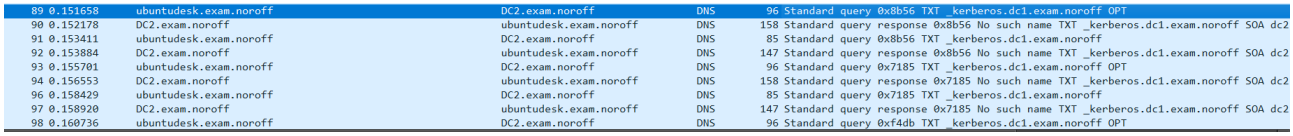

Figure 54: Screenshot: Wireshark capture(Solvang 2023)

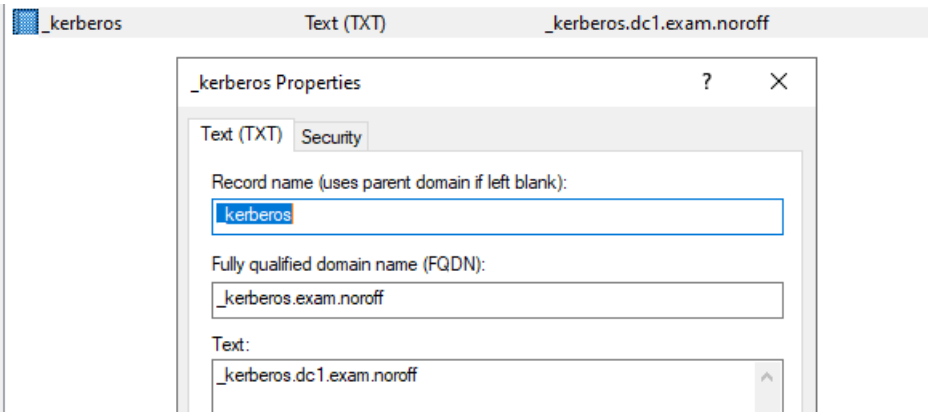

Figure 55: Screenshot: _Kerberos TXT DNS Record(Solvang 2023).

Analyzing filtered traffic and conversation between the ubuntudesk.exam.noroff, DC1 and DC2 provides a clue; a query response is repeated several times, stating ”No such name TXT _kerberos....” as seen in figure 54.

Searching for information on this leads to RFC 4120 ”Kerberos Network Authentication Service” and to an IETF Internet draft from a Network Working group titled ”Declaring Kerberos Realm Names in DNS.” It clearly states that DNSSEC should be used to assure the authenticity of resource records, which is a general recommendation for secure DNS servers.

Still, it details how to create a TXT DNS entry for Kerberos, as seen in figure 55(Van Rein. R 2023). The record was created in the forward lookup zone for exam.noroff.

As the Ubuntu Desktop, in adherence to the network design plan, receives its IP configuration and DNS server setting from the PfSense DHCP server, we need to adjust the sssd.conf file to ensure that it updates DNS records in AD, including the RTP record. Other configurations are also set based on findings in man pages for sssd.conf and sssd-ad.

The SSSD configuration file consists of sections and parameters. Sections are declared with the name inside square brackets and continue until the next section begins. Key values accepting either strings or boolean values are listed for each section. The file can configure SSSD and related services like NSS, PAM, SUDO, and SSH and includes a Domain section(Ubuntu Manpages n.d.[b]) Debug levels are set for every section to adjust output to log files.

An important setting to verify is to ensure the file has the right permissions, owner, and group membership. Use chmod 0600 and chown root:root.

- **Explaining configuration details:**
- $[SSSD]$
- In this section, the domain and config file version must be declared. In older versions, services

that need to be started were also declared, but this line is commented out due to debugging issues, as previously mentioned.

- The $[PAM]$ section is not configured or included by realmd, but appended to allow debugging.
- $[domain/exam.noroff]$
- krb5_store_password_if_offline and cache_credentials are set to True by default. Set to False to enhance security if the computer is compromised. ADDS should not become unavailable for authenticating users over the network.
- realmd_tags are stored by the realmd configuration service and are not modified.
- ad_backup_server is manually appended to the file to facilitate redundancy.
- dyndns configurations are not applied by default. As this client uses DHCP, we need to ensure our DNS server does not scrap aging records with failing authentication as a consequence. The appended configurations are successfully applied to AD DNS records and kept up to date.
- One last alteration is the use_fully_qualified_names; when set to false, the username domain extension is automatically appended, and users do not have to write out @exam.noroff.

Many service-related configuration keys have default values. One example is the chpass_provider key accepting a string value pointing to the provider that should handle change password operations. If no value is passed, it will default to the value set using the auth_provider key(Ubuntu Manpages n.d.[b]).

SSSD supports several identity providers, like LDAP, Kerberos and IPA, and interacts with many services to provide secure identification, authentication and authorization. Identifying a unified and clear set of instructions for configuration and implementation has proven difficult.

Even official documentation from providers like Canonical, Microsoft, Red Hat, and others is often incorrect or not up to date, and out-of-the-box integrations are flawed in their configurations. Extensive research, analysis, and debugging are the only viable methods to understand how protocols, services, and methodology correlate clearly.

A final, working SSSD configuration is shown in figure 56. Log-on is now fast and reliable, and password updates on the first log-on for new AD users work. One last note worth mentioning: if password requirements are not met when updating on the log-on screen, the only feedback to the user is ”authentication failed,” which can be confusing.

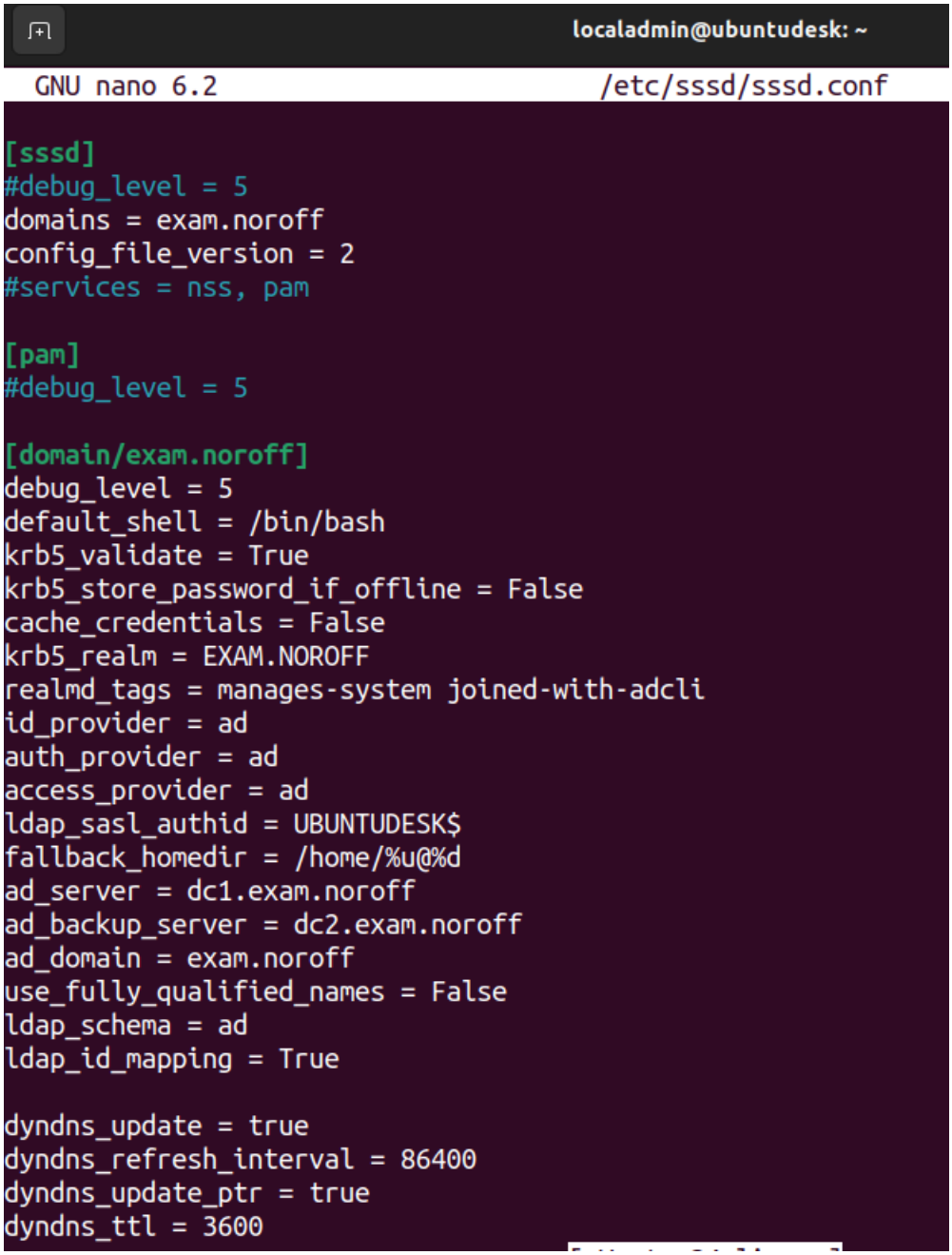

Figure 56: Screenshot: Final working SSSD configuration(Solvang 2023).

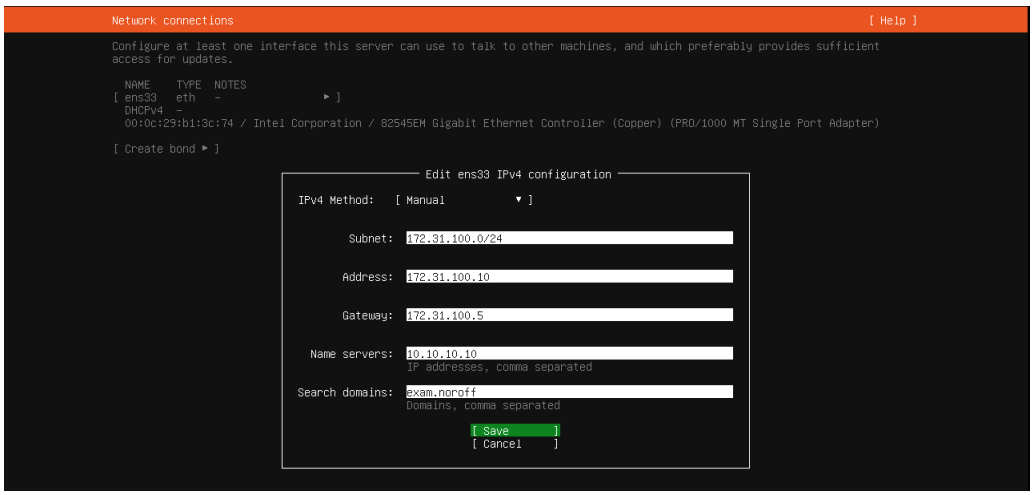

Figure 57: Screenshot: Network adapter configured during installation(Solvang 2023).

3.5 Apache Server Installation and Configuration

The Ubuntu server edition is installed without a GUI. The initial configuration follows the same procedures as with the Ubuntu Client. The server is configured with a static IPv4 address, apache as the hostname, and time synchronization with DC1. Packages required to join the domain must be installed manually. First, the server is updated using the apt package manager.

A good practice not mentioned working with the Ubuntu desktop is running journalctl and browsing for any issues that might have occurred during installation and first boot to verify a successful installation process. The output can be piped to grep using ”err” and ”fail.”
The default color scheme for the LS output is not optimal, so a change for the color on directories is made by editing .bashrc. The following line is appended at the end of the file: LS_COLORS=$LS_COLORS: ”di=4;35”. Then run the following command to apply the change: source .bashrc.

On Ubuntu Server, we must manually install the packages required to join Active Directory. Canonical guides refer to four packages, but we should still investigate if additional packages or dependencies are needed. The apt package manager allows us to show package information, and the output includes basic information, such as versions, depends, homepage, and a description. An example is given in figure 58.

Next, we go ahead and install the packages. Most dependencies will be discovered and installed automatically by the apt package manager on the fly, and suggestions on additional packages, like Python modules, may also be provided. Once the installation is complete, we can use the realm tool to discover the domain.

This will also list dependencies as seen in figure 59; notice that the samba-common-bin dependency and lib packages were not in the list of packages we already installed. We can check to see if they are installed using the apt list –installed command. We see from the output that the Samba package is not listed, and we can now install this as well. Once done, we can join the domain using the realm tool, as seen in figure 60.

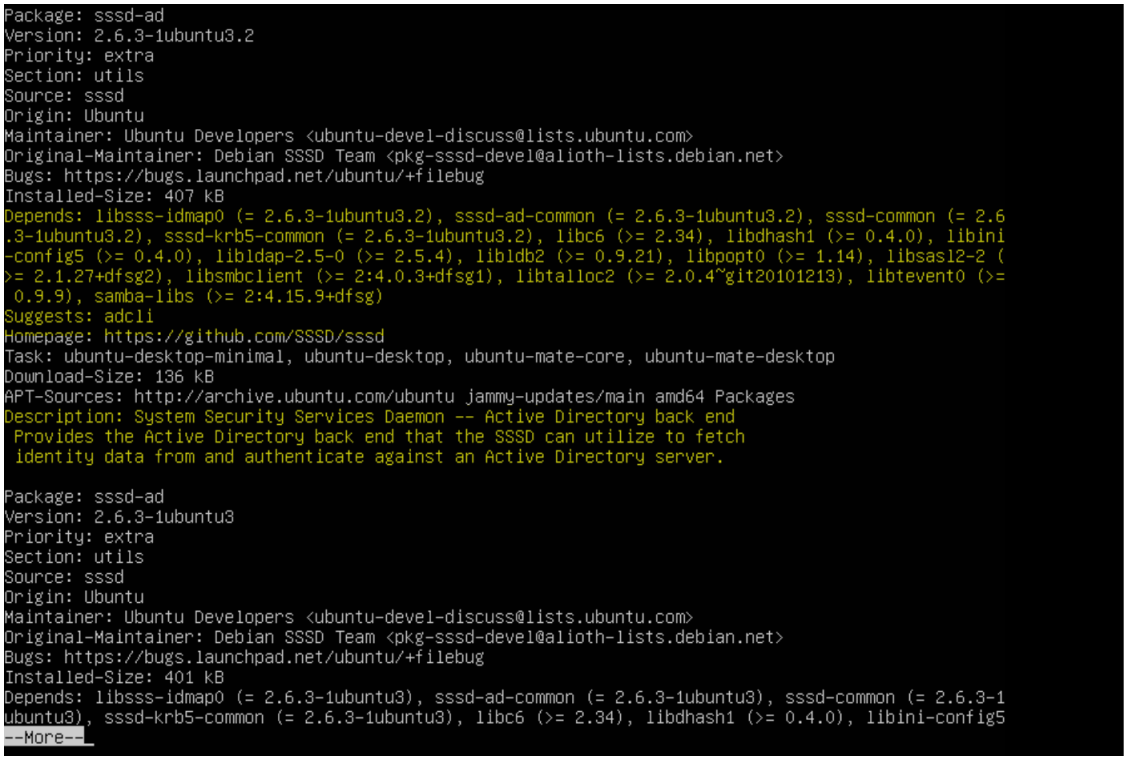

Figure 58: Screenshot: This screenshot shows parts of the output using the following command: *sudo apt -a show sssd-ad sssd-tools realmd adcli | more*. We can see any dependencies and determine if additional packages are required(Solvang 2023).

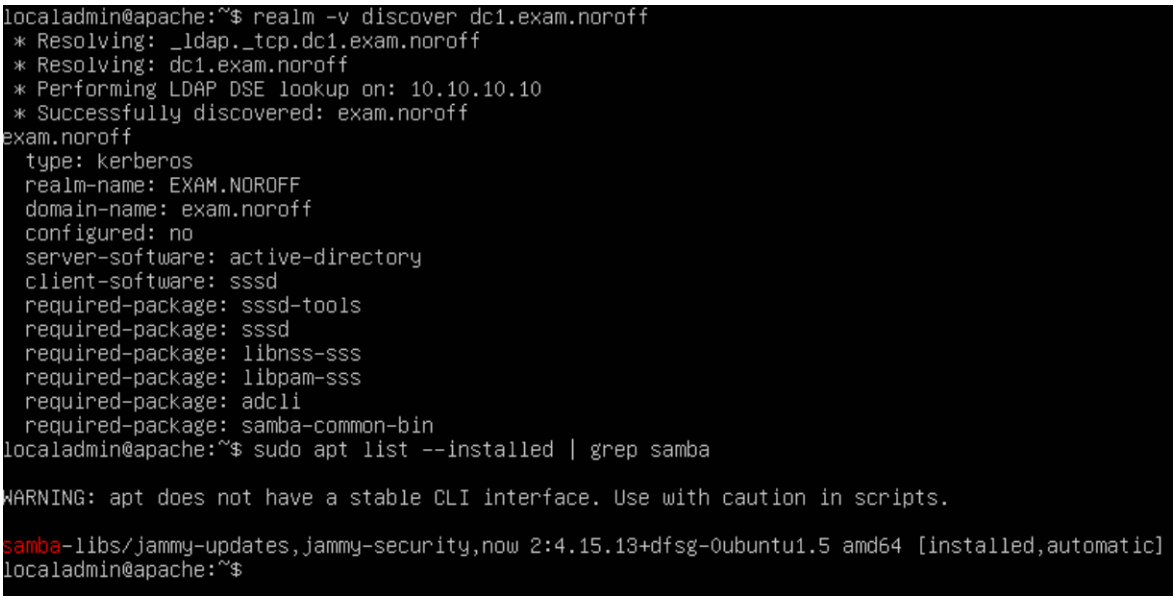

Figure 59: Screenshot: Using realm discovery and list –installed to identify if all required packages are installed(Solvang 2023).

```
localadmin@apache:~$ sudo realm join -U alexadmin dc1.exam.noroff
Password for alexadmin:
```

Figure 60: Screenshot: Notice that we use the alexadmin credentials by passing the -U option, as the default Administrator account is renamed and disabled.(Solvang 2023).

Realm returned no messages indicating that the Apache server was successfully joined to our domain. We can now use tools like ”realm list,” ”getent passed $< user@exam.com >$, and ”groups $< user@exam.com >$ to verify, as well as ”sudo login,” and pass the credentials of one of our domain users. At this point, no GPOs are made to restrict access. The user Karen@exam.noroff was successfully logged in on the server.

In AD Users and Computers, the server is located in the default Computer OU, where no GPOs other than the Default Domain Policy take effect. Moving objects in the OU structure should be performed with caution. As AD and the organization grow, a lot of policies will be crafted, so make sure to take this into account.

In this scenario, we want to place the server in our ”Main Office - PerimeterServers” OU to enable a GPO to restrict access to all servers in that container, and possibly apply strict security configurations for implicitly insecure perimeter network devices. Use the ”move” entry from the right-click popup on the Apache Server object.

At this point, we encounter a dilemma related to the principle of least privilege. An internet-facing application is at greater risk of compromise. That implies that the credentials used to access and manage the Apache server should not have access to secure servers and vice versa.

Our Web Developer, Karen, might be able to work with Apache and web development and also manage the Ubuntu Server. Then, granting her administrative access to the server calls for protecting these credentials and disallowing access to regular workstations as with IT Staff members. A suggested solution is to move her regular credentials from the Web-Developer OU to the Office OU and then provide her with Web Admin credentials that allow administrative access on PerimeterServer devices.

We must also restrict access to PermiterServers for all other groups, including IT Staff. One way of

```
localadmin@apache:~$ realm list
exam.noroff
  type: kerberos
  realm-name: EXAM.NOROFF
  domain-name: exam.noroff
  configured: kerberos-member
  server-software: active-directory
  client-software: sssd
  required-package: sssd-tools
  required-package: sssd
  required-package: libnss-sss
  required-package: libpam-sss
  required-package: adcli
  required-package: samba-common-bin
  login-formats: %U@exam.noroff
  login-policy: allow-realm-logins
localadmin@apache:~$ groups alexadmin@exam.noroff
alexadmin@exam.noroff : domain users@exam.noroff domain admins@exam.noroff pfsense-admin@exam.noroff
 denied rodc password replication group@exam.noroff itstaff@exam.noroff
localadmin@apache:~$ getent passwd alexadmin@exam.noroff
alexadmin@exam.noroff:*:1306401117:1306400513:AlexAdmin Solvang:/home/alexadmin@exam.noroff:/bin/bas
h
localadmin@apache:~$ sudo login
```

Figure 61: Screenshot: Verify Apache Server Realm Join(Solvang 2023).

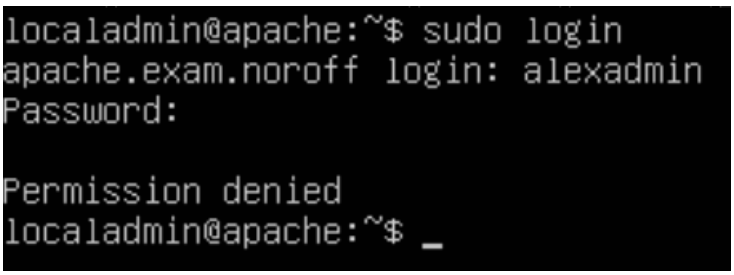

Figure 62: Screenshot: Alexadmin user denied permission to log on to Apache(Solvang 2023).

doing this is to create a GPO in Active Directory that allows ”local log-on” for members of the WebDeveloper OU. SSSD-ad supplies a key named ad_gpo_implicit_deny, which defaults to false.

This key function is quite peculiar. When set to false, and no GPOs are found, all users are allowed, but as soon as an Allow GPO is found, all other users are denied. Likewise, only users not listed in the Policy are allowed access if a Deny GPO is found. This means as soon as we create our ”Allow local log on” GPO, only the users linked to the GPO can access(Ubuntu Manpages n.d.[a]). Localadmin still works!

The GPO is configured with the WebDeveloper Security Group in the Security Filtering tab, then ”allow local log on” and ”Allow network access to this computer.” Microsoft also warns in a security bulletin that creating ”Allow” GPOs is risky. Particular standard user objects have access by default to allow services to function correctly. By forming a new policy, the default allow access users are no longer included(Microsoft Support n.d.).

So, allow GPOs are not to be used on Domain Controllers and should undoubtedly be used with caution on Windows workstations. Default users on workstations and servers are Administrators, Backup Operators, Power Users, Users, and Guests. Applying the configuration without adding the Administrators group to log on locally was impossible when crafting the Policy.

When moving the regular ”Karen” user, we must also remember to change group memberships. Simply moving to a new container won’t affect the security group membership. The new configuration is tested after a new user, ”karenadmin,” is created. Alexadmin cannot log on, as seen in figure 62.

Karenadmin is prompted to change her password and logs on successfully. Note that changing password does not work without specifying the auth_provider in the sssd.conf, the file is configured in the same way as on the Ubuntu Desktop.

Karenadmin can now install and configure the Apache Web Server and get to work.
Installing Apache is reasonably straightforward. I will now demonstrate how to install the Apache Webserver, adjust the local firewall, create a self-signed SSL certificate, and install it on the web server step by step.

- Run: sudo apt install apache2 - This installs the web server and any dependencies.
- Run: sudo ufw app list - This will display available APP rulesets for the Unified Firewall
- Run: sudo ufw allow ”apache full” - These will be the necessary ports on the local firewall.
- Now, we need to create a self-signed SSL certificate; see screenshot 63.

```
localadmin@apache:~$ sudo openssl genrsa 2048 > ./apache.sefsigned.key
localadmin@apache:~$ sudo openssl req -new -key ./apache.sefsigned.key > apache.request.csr
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [AU]:NO
State or Province Name (full name) [Some-State]:Nordland
Locality Name (eg, city) []:Sortland
Organization Name (eg, company) [Internet Widgits Pty Ltd]:Noroff
Organizational Unit Name (eg, section) []:Exam
Common Name (e.g. server FQDN or YOUR name) []:apache.exam.noroff
Email Address []:alexander@solvang-it.no

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
localadmin@apache:~$ sudo openssel x509 -in ./apache.request.csr -out apache.crt -req -signkey apach
e.sefsigned.key -days 365
sudo: openssel: command not found
localadmin@apache:~$ sudo openssl x509 -in ./apache.request.csr -out apache.crt -req -signkey apache
.sefsigned.key -days 365
Certificate request self-signature ok
subject=C = NO, ST = Nordland, L = Sortland, O = Noroff, OU = Exam, CN = apache.exam.noroff, emailAd
dress = alexander@solvang-it.no
localadmin@apache:~$
```

Figure 63: Screenshot: Create self-signed SLL certificate(Solvang 2023).

- Next, back up the default site configuration in /etc/apache2/sites-available
- Move or copy the keys, default is the SSL store, but here we just put them in the same folder and edit the file accordingly, as seen in figure 64.
- Run: sudo apache2ctl configtest - this command will help you find any errors in your config.
- Run: sudo systemctl apache2 restart
- Copy the .crt file and install it on a domain client. As DNS records for apache.exam.noroff already exists, we can now navigate to the default landing page by entering the entire URL in the web browser: https://apache.exam.noroff, and the result is seen in figure 65

```
 GNU nano 6.2                 /etc/apache2/sites-available/000-default.conf *
<VirtualHost *:443>
        # The ServerName directive sets the request scheme, hostname and port that
        # the server uses to identify itself. This is used when creating
        # redirection URLs. In the context of virtual hosts, the ServerName
        # specifies what hostname must appear in the request's Host: header to
        # match this virtual host. For the default virtual host (this file) this
        # value is not decisive as it is used as a last resort host regardless.
        # However, you must set it for any further virtual host explicitly.
        ServerName apache.exam.noroff
        SSLEngine on
        SSLCertificateFile /etc/apache2/sites-available/apache.crt
        SSLCertificateKeyFile /etc/apache2/sites-available/apache.selfsigned.key

        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html
```

Figure 64: Screenshot: Default site HTTPS config(Solvang 2023)

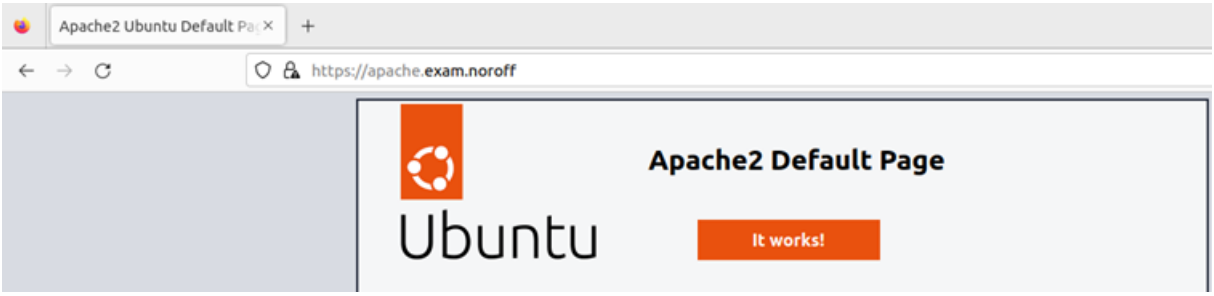

Figure 65: [Screenshot: Apache Landing page with HTTPS(Solvang 2023).

4 A proposed network diagram in Packet Tracer

The network topology created in Packet Tracer will pursue the following objectives:

- Use Vlans to achieve network segmentation and additional security.
- Layer 3 switches are used, adding 4 1000BASE-TE SFP modules.
- Implent and secure the Spanning Tree protocol.
- Secure the Spanning tree protocol:
  - Enable PBDU guard.
  - Enable PBDU filter.
  - Enable root guard.
  - Disable dynamic trunking.
  - Use per VLan Spanning tree (PVST)
- Enable port security using Mac address sticky mode (Cisco n.d.)
- Use IPSec to secure communication between locations Oslo and Bergen.

The configurations and design aim to ensure overall security on our network, and several countermeasures are implemented to protect against internal and external threats. Every command issued on each device is presented in the next section of this report to document the configuration of each device.

Using device hardening and implementing countermeasures for known vulnerabilities, this topology, together with the best practices described in the Server lab, collectively delivers proof of concept on taking necessary steps to ensure Confidentiality, Integrity, Availability, Authenticity, and Non-repudiation as described in the CIAAN reference model(Cherdantseva and Hilton 2013).

All networking devices are secured with passwords for both in-band and out-of-band connections using SSH for remote access with hashed credentials stored in local databases. This includes making a separate management VLAN, only for management purposes, with dedicated IP addresses for remote connections. Secure administrative Workstations, like the PAW, would be members of this network segment.

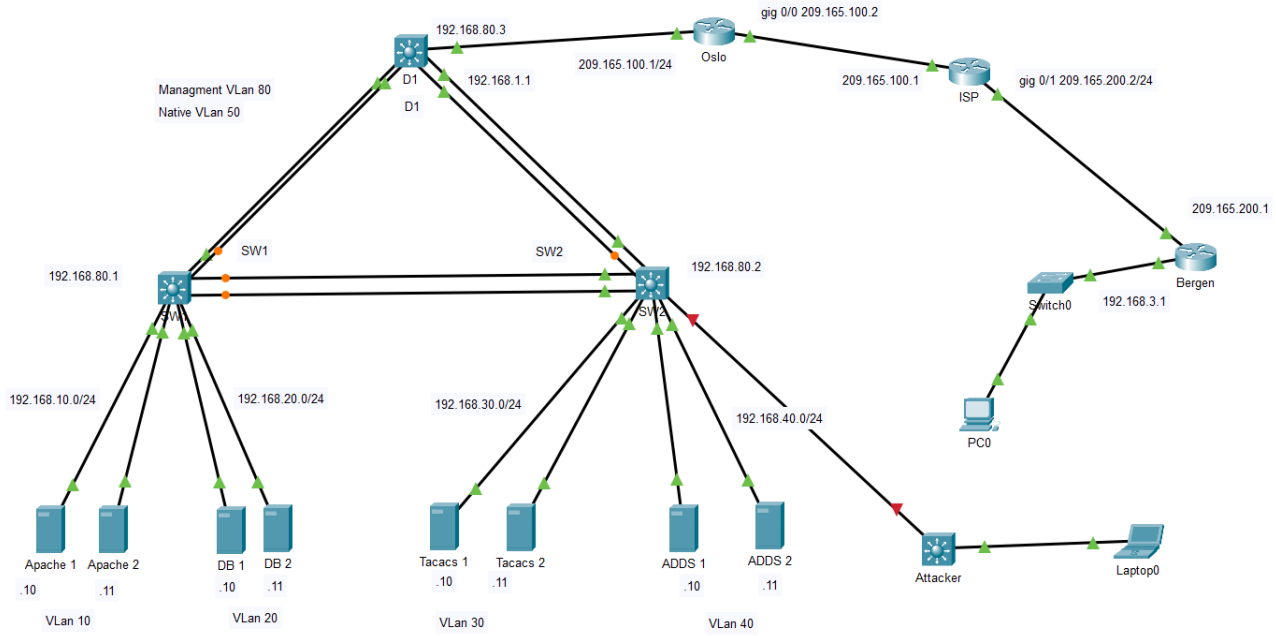

Figure 66: Screenshot: Packet Tracer Network topology(Solvang 2023).

4.1 Network Topology

The following section is purely technical. The image below displays the network layout in Packet Tracer, and then all configurations made to the Layer 3 switches are provided in the same order as they are issued. The last section provides step-by-step instructions and oversight on how we set up IPSec to secure communication between the Oslo and Bergen offices.

Notice how the connection from the ”Attacker” switch is dead. This demonstrates a port-security violation using port security with sticky MAC and emulates an attacker who has attempted to abuse an active port by disconnecting existing equipment in an attempt to access the network with another device.

This configuration enables dynamic learning of MAC addresses; by default, it accepts only the first address connecting to the port, but this can be configured to a maximum range of 4096. You can also manually set trusted addresses on each port.

4.1.1 SW1 configuration

4.2.1 SW1 - 3650-24PS with four additional 1000BASE-T SFP modules and redundant PSU.

1. (config)# Enable secret xxxxx
2. (config)# Hostname SW1
3. SW1(config)#interfaces range gig 1/1/1-4
4. SW1(config-if-range)#switchport mode trunk
5. SW1(config-if-range)#switchport trunk native vlan 50
6. SW1(config-if-range)#switchport port-security mac-address sticky
7. SW1(config-if-range)#no shutdown

8. SW1(config)#interfaces range gig 1/0/1-24 (User side ports for connecting single devices.)
9. SW1(config-if-range)#switchport mode access
10. SW1(config-if-range)#swithport nonegotiate
11. SW1(config-if-range)#swithport port-security mac-address sticky
12. SW1(config-if-range)#spanning-tree portfast
13. SW1(config-if-range)#spanning-tree bpduguard enable
14. SW1(config-if-range)#shutdown
15. SW1(config-if-range)#exit
16. SW1(config)#vlan 10
17. SW1(config-vlan)#name WEB
18. SW1(config-vlan)#vlan 20
19. SW1(config-vlan)#name DB
20. SW1(config-vlan)#vlan 30
21. SW1(config-vlan)#name Tacacs
22. SW1(config-vlan)#vlan 40
23. SW1(config-vlan)#name APP
24. SW1(config-vlan)#vlan 80
25. SW1(config-vlan)#name Management
26. SW1(config-vlan)#exit
27. SW1(config)#interface range gig 1/0/1-2
28. SW1(config-if-range)#switchport access vlan 10
29. SW1(config-if-range)#no shutdown
30. SW1(config-if-range)#interface range gig 1/0/10-11
31. SW1(config-if-range)#swithport access vlan 20
32. SW1(config-if-range)#no shutdown
33. SW1(config-if-range)#end
34. SW1# show vlan brief 67
35. SW1# show interfaces trunk 68
36. SW1(config)#ip domain-name exam.noroff
37. SW1(config)#username Alexander secret xxxx

```
SW1#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Gig1/0/3, Gig1/0/4, Gig1/0/5, Gig1/0/6
                                                Gig1/0/7, Gig1/0/8, Gig1/0/9, Gig1/0/12
                                                Gig1/0/13, Gig1/0/14, Gig1/0/15, Gig1/0/16
                                                Gig1/0/17, Gig1/0/18, Gig1/0/19, Gig1/0/20
                                                Gig1/0/21, Gig1/0/22, Gig1/0/23, Gig1/0/24
10   WEB                              active    Gig1/0/1, Gig1/0/2
20   DB                               active    Gig1/0/10, Gig1/0/11
30   TACACS                           active
40   APP                              active
80   Management                       active
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
```

Figure 67: SW1 Show Vlans brief(Solvang 2023).

38. SW1(config)#crypto generate rsa (2048)
39. SW1(config)#line vty 0 4
40. SW1(config-line)#login local
41. SW1(config-line)#transport input ssh
42. SW1(config-line)#end
43. SW1#show running-config (verify)
44. SW1#copy running-config startup-config

4.1.2 SW2 configuration

SW2 - 3650-24PS with four additional 1000BASE-T SFP modules and redundant PSU.

- (config)# Enable secret xxxxx
- (config)# Hostname SW2
- SW2(config)#interfaces range gig 1/1/1-4
- SW2(config-if-range)#switchport mode trunk
- SW2(config-if-range)#swithport trunk native vlan 50
- SW2(config-if-range)#switchport port-security mac-address sticky
- SW2(config-if-range)#switchport port-security mac-address sticky
- SW2(config-if-range)#no shutdown
- SW2(config-if-range)#interfaces range gig 1/0/1-24
- SW2(config-if-range)#switchport mode access
- SW2(config-if-range)#swithport nonegotiate
- SW2(config-if-range)#swithport port-security mac-address sticky
- SW2(config-if-range)#spanning-tree portfast

```
Port        Mode         Encapsulation  Status        Native vlan
Gig1/1/1    on           802.1q         trunking      50
Gig1/1/2    on           802.1q         trunking      50
Gig1/1/3    on           802.1q         trunking      50
Gig1/1/4    on           802.1q         trunking      50

Port        Vlans allowed on trunk
Gig1/1/1    1-1005
Gig1/1/2    1-1005
Gig1/1/3    1-1005
Gig1/1/4    1-1005

Port        Vlans allowed and active in management domain
Gig1/1/1    1,10,20,30,40,80
Gig1/1/2    1,10,20,30,40,80
Gig1/1/3    1,10,20,30,40,80
Gig1/1/4    1,10,20,30,40,80

Port        Vlans in spanning tree forwarding state and not pruned
Gig1/1/1    1,10,20,30,40,80
Gig1/1/2    none
Gig1/1/3    1,10,20,30,40,80
Gig1/1/4    1,10,20,30,40,80
```

Figure 68: Screenshot: SW1 Show interfaces Trunk(Solvang 2023).

- SW2(config-if-range)#spanning-tree bpduguard enable
- SW2(config-if-range)#shutdown
- SW2(config-if-range)#exit
- SW2(config)#vlan 10
- SW2(config-vlan)#name WEB
- SW2(config-vlan)#vlan 20
- SW2(config-vlan)#name DB
- SW2(config-vlan)#vlan 30
- SW2(config-vlan)#name Tacacs
- SW2(config-vlan)#vlan 40
- SW2(config-vlan)#name APP
- SW2(config-vlan)#vlan 80
- SW2(config-vlan)#name Management
- SW2(config-vlan)#exit
- SW2(config)#interface range gig 1/0/1-2

```
SW2#show vlan brief

VLAN Name                             Status    Ports
---- -------------------------------- --------- -------------------------------
1    default                          active    Gig1/0/3, Gig1/0/4, Gig1/0/5, Gig1/0/6
                                                Gig1/0/7, Gig1/0/8, Gig1/0/9, Gig1/0/12
                                                Gig1/0/13, Gig1/0/14, Gig1/0/15, Gig1/0/16
                                                Gig1/0/17, Gig1/0/18, Gig1/0/19, Gig1/0/20
                                                Gig1/0/21, Gig1/0/22, Gig1/0/23, Gig1/0/24
10   WEB                              active
20   DB                               active
30   TACACS                           active    Gig1/0/1, Gig1/0/2
40   App                              active    Gig1/0/10, Gig1/0/11
80   Management                       active
1002 fddi-default                     active
1003 token-ring-default               active
1004 fddinet-default                  active
1005 trnet-default                    active
SW2#
```

Figure 69: Screenshot: SW2 Show Vlan brief(Solvang 2023).

- SW2(config-if-range)#switchport access vlan 20
- SW2(config-if-range)#no shutdown
- SW2(config-if-range)#interface range gig 1/0/10-11
- SW2(config-if-range)#swithport access vlan 40
- SW2(config-if-range)#no shutdown
- SW2(config-if-range)#end
- SW2# show vlan brief 69
- SW2# show interfaces trunk 70
- SW2(config)#ip domain-name exam.noroff
- SW2(config)#username Alexander secret xxxx
- SW2(config)#crypto generate rsa (2048)
- SW2(config)#line vty 0 4
- SW2(config-line)#login local
- SW2(config-line)#transport input ssh
- SW2(config-line)#end
- SW2#show running-config (verify)
- SW2#copy running-config startup-config

```
SW2#show interfaces trunk
Port        Mode         Encapsulation  Status        Native vlan
Gig1/1/1    on           802.1q         trunking      50
Gig1/1/2    on           802.1q         trunking      50
Gig1/1/3    on           802.1q         trunking      50
Gig1/1/4    on           802.1q         trunking      50

Port        Vlans allowed on trunk
Gig1/1/1    1-1005
Gig1/1/2    1-1005
Gig1/1/3    1-1005
Gig1/1/4    1-1005

Port        Vlans allowed and active in management domain
Gig1/1/1    1,10,20,30,40,80
Gig1/1/2    1,10,20,30,40,80
Gig1/1/3    1,10,20,30,40,80
Gig1/1/4    1,10,20,30,40,80

Port        Vlans in spanning tree forwarding state and not pruned
Gig1/1/1    none
Gig1/1/2    none
Gig1/1/3    1,10,20,30,40,80
Gig1/1/4    none
```

Figure 70: Screenshot: SW2 Show Interfaces Trunk(Solvang 2023).

4.1.3 D1 configuration

Enabling SPT in this topology, we use this as the root device responsible for routing. STP prevents packets from being forwarded between switches in a never-ending loop, as there is no TTL for layer two frames.

This device is also the root bridge, and root guard is enabled to prevent other bridges connected to the port range specified from becoming a new root bridge. SSH and Tacacs authentication is configured as the default log-in method, using the server group Tacacs server and local (last) as backup in case of network failure.

D1 - 3650-24PS with four additional 1000BASE-T SFP modules and redundant PSU.

- Switch(config)# Enable secret xxxxx
- Switch(config)# Hostname D1
- D1(config)#interfaces range gig 1/1/1-4
- D1(config-if-range)#switchport mode trunk
- D1(config-if-range)#swithport trunk native vlan 50
- D1(config-if-range)#switchport port-security mac-address sticky
- D1(config-if-range)#spanning tree guard
- D1(config-if-range)#no shutdown
- D1(config)#interfaces range gig 1/0/1-24
- D1(config-if-range)#switchport mode access
- D1(config-if-range)#switchport nonegotiate
- D1(config-if-range)#switchport port-security mac-address sticky
- D1(config-if-range)#spanning-tree fastport
- D1(config-if-range)#spanning-tree bpduguard

- D1(config-if-range)#shutdown
- D1(config-if-range)#exit
- D1(config)#interface vlan 10
- D1(config-if)#ip address 192.168.10.1 255.255.255.0
- D1(config)#interface vlan 20
- D1(config-if)#ip address 192.168.20.1 255.255.255.0
- D1(config)#interface vlan 30
- D1(config-if)#ip address 192.168.30.1 255.255.255.0
- D1(config)#interface vlan 40
- D1(config-if)#ip address 192.168.40.1 255.255.255.0
- D1(config)#interface vlan 80
- D1(config-if)#ip address 192.168.80.3 255.255.255.0 (Not GW, only to login on D1 with SSH).
- D1(config-if)#end
- D1#show ip interfaces brief 71
- D1(config)#ip domain-name exam.noroff
- D1(config)#username Alexander secret xxxx
- D1(config)#crypto generate rsa (2048)
- D1(config)#line vty 0 4
- D1(config-line)#login local
- D1(config-line)#transport input ssh
- D1(config-line)#exit
- D1(config)#line console 0
- D1(config-line)#login local
- D1(config-line)#line aux 0
- D1(config-line)#login local
- D1(config-line)#end
- D1#show running-config (verify)
- D1#copy running-config startup-config
- D1(config)#aaa new-model
- D1(config)#aaa authentication login default group tacacs+ local

```
GigabitEthernet1/0/19  unassigned      YES unset  administratively down down
GigabitEthernet1/0/20  unassigned      YES unset  administratively down down
GigabitEthernet1/0/21  unassigned      YES unset  administratively down down
GigabitEthernet1/0/22  unassigned      YES unset  administratively down down
GigabitEthernet1/0/23  unassigned      YES unset  administratively down down
GigabitEthernet1/0/24  unassigned      YES unset  administratively down down
GigabitEthernet1/1/1   unassigned      YES unset  up                    up
GigabitEthernet1/1/2   unassigned      YES unset  up                    up
GigabitEthernet1/1/3   unassigned      YES unset  up                    up
GigabitEthernet1/1/4   unassigned      YES unset  up                    up
Vlan1                  unassigned      YES unset  administratively down down
Vlan10                 192.168.10.1    YES manual up                    up
Vlan20                 192.168.20.1    YES manual up                    up
Vlan30                 192.168.30.1    YES manual up                    up
Vlan40                 192.168.40.1    YES manual up                    up
Vlan80                 192.168.80.1    YES manual up                    up
```

Figure 71: Screenshot: D1 IP interfaces brief(Solvang 2023).

- D1(config)#aaa authentication enable default group tacacs+ local (granting access to enable mode)
- D1(config)#aaa authentication ppp default group tacacs+ local (enables Tacacs authentication for remote sessions)
- D1(config)#tacacs-server host 192.168.30.10 key Tacacs (points to tacacs authentication server and group/user database).

4.1.4 IP Sec configuration

To set up IPsec VPN peer-to-peer tunneling, we need to install a license from Cisco on the routers on either side. A trial license can be installed with the following command: #license boot module c1900 technology-package securityK9
Each router used to connect the locations Oslo and Bergen and the ISP router are provided a simple starting configuration that must be in place before we can start configuring the IPSec tunnel.

4.1.5 Preparing the routers

- **Oslo**
  - #hostname OSLO
  - #interface gig 0/1
  - #ip address 192.168.1.1 255.255.255.0
  - #no shut
  - #interface gig 0/0
  - #ip address 209.165.100.1 255.255.255.0
  - #no shut
  - #exit
  - #ip route 0.0.0.0 0.0.0.0 209.165.100.2
- **ISP**
  - #hostname ISP

- #interface gig 0/1
- #ip address 209.165.200.2 255.255.255.0
- #no shut
- #interface gig 0/0
- #ip address 209.165.100.2 255.255.255.0
- #no shut
- #exit
- Notice how no route is provided. The ISP router is essentially unaware of any traffic tunneled with IPSec.

- **Bergen**
  - #hostname Bergen
  - #interface gig 0/1
  - #ip address 192.168.3.1 255.255.255.0
  - #no shut
  - #interface gig 0/0
  - #ip address 209.165.200.1 255.255.255.0
  - #no shut
  - #exit
  - #ip route 0.0.0.0 0.0.0.0 209.165.200.2

4.1.6 Configure IPSec

- **Oslo**
  - #crypto isakmp policy 10
  - #encryption aes 256
  - #authentication pre-share
  - #group 5
  - #crypto isakmp key secretkey address 209.165.200.1 (- Set a secretkey/passphrase of you own choosing)
  - #crypto ipsec transform-set Oslo-Bergen esp-aes 256 esp-sha-hmac
  - #crypto map IPSEC-MAP 10 ipsec-isakmp (10 represents the policy created previously)
  - #set peer 209.165.200.1
  - #set pfs group5
  - #set security-association lifetime seconds 86400
  - #set transform-set Oslo-Bergen
  - #match address 100
  - #interface GigabitEthernet0/0
  - #crypto map IPSEC-MAP
  - #access-list 100 permit ip 192.168.0.0 0.0.255.255 192.168.3.0 0.0.0.255

  - #access-list 100 permit ip 192.168.1.0 0.0.0.255 192.168.3.0 0.0.0.255

- **Bergen**
  - #crypto isakmp policy 10
  - #encryption aes 256
  - #authentication pre-share
  - #group 5
  - #crypto isakmp key secretkey address 209.165.100.1
  - #crypto ipsec transform-set Bergen-Oslo esp-aes 256 esp-sha-hmac
  - #crypto map IPSEC-MAP 10 ipsec-isakmp
  - #set peer 209.165.100.1
  - #set pfs group5
  - #set security-association lifetime seconds 86400
  - #set transform-set Bergen-Oslo
  - #match address 100
  - #interface GigabitEthernet0/0
  - #crypto map IPSEC-MAP
  - #access-list 100 permit ip 192.168.3.0 0.0.0.255 192.168.0.0 0.0.255.255
  - #access-list 100 permit ip 192.168.3.0 0.0.0.255 192.168.1.0 0.0.0.255

5 Summary and conclusions

This investigative report shows how three very different operating systems can work together, using Active Directory as an identity provider. We have seen an operating system completely purpose-built, or at least maintained, to support PfSense as a networking device, providing security for our ADDS environment.

PfSense and FreeBSD can undoubtedly enhance security and provide extensive functionality that could benefit a large corporate network. High availability, NAT, VLAN routing, Intrusion detection, and traffic analysis, in addition to granular rule-based traffic control, are all specialized security tools that can contribute to corporate network security. Together with ADDS, these two operating systems can provide efficient tools to achieve industry-standard security goals.

These systems combined could deliver reliable, scalable, and highly available services in a production environment while ensuring security and protecting corporate data.

Testing ADDS against Ubuntu Linux clients was also a success. We learned that well-known compatibility issues between Microsoft and Linux have been solved in modern versions of Open Source software and the Windows environment.

Investigating the principle of least privilege access, and User rights management, it was discovered that SSSD and AD work quite well together. We found security control functions that allowed us to choose whether or not to cache credentials on the local client.

We could also leverage AD security groups, OUs, and GPOs to control computer access on a group level. This allows us to protect privileged access credentials and critical infrastructure throughout the network and on every operating system included in this research.

It is also quite clear that most of the security configurations that may be applied network-wide in a Windows-only environment will not affect Linux clients and that Linux GPO templates are lean in that context. We also had to perform extensive debugging and local configuration of the Linux clients compared with Windows.

A great advantage in a large environment is that we could use GPOs to apply security network wide. One example we saw is the best practice principle to rename and disable default accounts prone to malicious abuse. Lacking other controls, this weakness should be considered before deciding to allow 100s or perhaps 1000s of employees to use Linux clients freely.

One suggestion to remedy this drawback could be using thin clients and PXE.

Another important aspect to consider is the security risks that might follow. Maintaining a network with different operating systems is resource-intensive from an administrative perspective. It also increases the attack surface, adds to the number of applications used, and contradicts best practices to that end on its own accord.

Usually, IT Staff specialize in one or the other; gaining experience and building expertise takes time. But one can argue that this is also true for any malicious actor seeking to breach the network. If we assume that overall security is maintained, breaking into a multifaceted network structure, and leveraging several different operating systems might not be so easy.

When assessing our findings, an absolute answer to the initial question is not clear. Ultimately, every corporation must investigate how to reach its desired goals and what tools and resources can best facilitate that. Every business has its starting points, but based on the research results, it is safe to say that it is possible, that mixing operating systems can enhance a modern computer network.

References

Agile IT (2009). *Active Directory Limits – Maximum Objects, Attributes, Servers, Trusts, Domain Controllers, etc. - Agile IT*. URL: https://www.agileit.com/active-directory-limits-maximum-objects-attributes-servers-trusts-domain-controllers-etc/.

Canonical (2022). *Executive Summary Integration of Ubuntu Desktop with Microsoft Active Directory*. Tech. rep.

Cherdantseva, Yulia and Jeremy Hilton (2013). *A Reference Model of Information Assurance & Security**. Tech. rep.

Cisco (n.d.). *Security - Configuring Port Security [Support] - Cisco Systems*. Accessed; 10/08/2023. URL: https://www.cisco.com/en/US/docs/general/Test/dwerblo/broken_guide/port_sec.html#wp1062570.

https://sssd.io (2021a). *SSSD - Architecture*. Accessed: 06/02/2024. URL: https://sssd.io/contrib/architecture.html.

— (2021b). *SSSD - Introduction*. Accessed: 6.2.2024. URL: https://sssd.io/docs/introduction.html.

Jungles Patrick et al. (2012). *Mitigating Pass-the-Hash (PtH) Attacks and Other Credential Theft Techniques Mitigating the risk of lateral movement and privilege escalation*. Tech. rep.

Kelley, Simon (n.d.). *Dnsmasq - network services for small networks.* Accessed: 10/02/2024. URL: https://thekelleys.org.uk/dnsmasq/doc.html.
Kerberos Consortium (2007). *MIT Kerberos Consortium - About Us*. URL: https://kerberos.org/about/FAQ.html.
Microsoft Learn (2014). *Active Directory and Active Directory Domain Services Port Requirements*. URL: https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/dd772723(v=ws.10)?redirectedfrom=MSDN#communication-to-domain-controllers.
— (2020a). *Active Directory Schema (AD Schema) - Win32 apps — Microsoft Learn*. Accessed: 07/02/2023. URL: https://learn.microsoft.com/en-us/windows/win32/adschema/active-directory-schema.
— (2020b). *Global Catalog - Win32 apps — Microsoft Learn*. Accessed: 07/02/2024. URL: https://learn.microsoft.com/en-us/windows/win32/ad/global-catalog.
— (2021a). *Implementing Secure Administrative Hosts — Microsoft Learn*. URL: https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/plan/security-best-practices/implementing-secure-administrative-hosts.
— (2021b). *Kerberos Authentication Overview — Microsoft Learn*. URL: https://learn.microsoft.com/en-us/windows-server/security/kerberos/kerberos-authentication-overview.
— (2021c). *Security Support Provider Interface Architecture — Microsoft Learn*. URL: https://learn.microsoft.com/en-us/windows-server/security/windows-authentication/security-support-provider-interface-architecture.
— (Aug. 2022a). *Active Directory Domain Services Overview — Microsoft Learn*. URL: https://learn.microsoft.com/nb-no/windows-server/identity/ad-ds/get-started/virtual-dc/active-directory-domain-services-overview.
— (2022b). *Comparison of Standard, Datacenter, and Datacenter Azure Edition editions of Windows Server 2022 — Microsoft Learn*. URL: https://learn.microsoft.com/en-us/windows-server/get-started/editions-comparison-windows-server-2022?tabs=full-comparison.
— (2023a). *Create and manage Central Store - Windows Client — Microsoft Learn*. Accessed: 09/02/2024. URL: https://learn.microsoft.com/en-us/troubleshoot/windows-client/group-policy/create-and-manage-central-store.
— (2023b). *Implementing Least-Privilege Administrative Models — Microsoft Learn*. URL: https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/plan/security-best-practices/implementing-least-privilege-administrative-models.
— (2024). *Developing a privileged access strategy - Privileged access — Microsoft Learn*. Accessed: 14/02/2024. URL: https://learn.microsoft.com/en-us/security/privileged-access-workstations/privileged-access-strategy.
Microsoft Support (n.d.). *Client, service, and program issues can occur if you change security settings and user rights assignments - Microsoft Support*. Accessed: 13/02/2024. URL: https://support.microsoft.com/en-us/topic/client-service-and-program-issues-can-occur-if-you-change-security-settings-and-user-rights-assignments-0cb6901b-dcbf-d1a9-e9ea-f1b49a56d53a.
Netgate (2023). *Virtualization — pfSense Documentation*. URL: https://docs.netgate.com/pfsense/en/latest/virtualization/index.html.
Neuman, Clifford and et al. (July 2005). *RFC 4120 - The Kerberos Network Authentication Service (V5)*. URL: https://datatracker.ietf.org/doc/html/rfc4120.
Orin, Thomas (2020). *Windows Server 2019 Inside Out*. Ed. by Brett Bartow et al. Pearson Education Inc. ISBN: 978-0-13-549227-7.
Rekhter, Y. et al. (Feb. 1996). "RFC 1918 - Address Allocation for Private Internets". In: ISSN: 2070-1721. DOI: 10.17487/RFC1918. URL: https://www.rfc-editor.org/info/rfc1918.

Solvang, Alexander (2023). “Exam Project, FS1SP1131”. Bø i Vesterålen.
Tanenbaum, Andrew S. and Herbert Bos (2015). *Modern Operating Systems*. Ed. by Marcia Horton and Tracy Johnson. 4th Edition. Essex: Pearson Education Limited. ISBN: 1-292-06142-1. URL: www.pearsonglobaleditions.com.
Ubuntu Manpages (n.d.[a]). *Ubuntu Manpage: sssd-ad - SSSD Active Directory provider*. Accessed: 15/02/2024. URL: https://manpages.ubuntu.com/manpages/focal/en/man5/sssd-ad.5.html.
— (n.d.[b]). *Ubuntu Manpage: sssd.conf - the configuration file for SSSD*. Accessed: 14/02/2024. URL: https://manpages.ubuntu.com/manpages/trusty/man5/sssd.conf.5.html.
Van Rein. R (2023). *draft-vanrein-dnstxt-krb1-11*. Accessed: 11/02/2024. URL: https://datatracker.ietf.org/doc/html/draft-vanrein-dnstxt-krb1#name-defining-_kerberos-txt-reso.
Vmware (2019). *Assigning IP Addresses in Host-Only Networks and NAT Configurations*. URL: https://docs.vmware.com/en/VMware-Workstation-Pro/17/com.vmware.ws.using.doc/GUID-144D22BA-298E-4293-8137-B631AD7BF694.html.