

Capstone Report

Jayce Mangum

Computer Security & Network Technology

Lake Washington Institute of Technology

Professor Linda Epps

16 June 2026

Table of Contents

Executive Summary	3
Introduction	4
Methodology	6
Network Design	10
Infrastructure Implementation	17
Linux Automation	28
Security Implementation	33
Troubleshooting Documentation	41
Testing & Validation	53
Risk Analysis	58
Performance Analysis	61
Lessons Learned	65
Future Improvements	68
Conclusion	70
Appendices	72
References	163

Executive Summary

Master Sword Publishing (MSPUB) is a fictional publishing company that I created as part of the Computer Security and Network Technology (CSNT) 253 Capstone Project. The objective of this project was to design, implement, secure, and document a complete enterprise information technology environment that incorporates concepts learned throughout the CSNT program.

My project consisted of building a virtual business infrastructure that included Windows Server 2025 domain controllers, Active Directory Domain Services (AD DS), Domain Name System (DNS) services, Windows, macOS, Ubuntu, and Kali Linux workstations, as well as Linux-based web services. The environment was hosted within a virtualized lab using Parallels Desktop and was designed to simulate a real-world small business network.

Several automation solutions were developed using Python to reduce administrative overhead and improve consistency. These included an automated web server deployment script and a user management system capable of creating, modifying, and managing user accounts and groups. Networking concepts such as IP addressing, Virtual Local Area Network (VLAN) segmentation, routing, and access control were incorporated into the overall infrastructure design.

Security was a primary focus throughout the project. System hardening techniques, operating system security features, vulnerability mitigation, and security awareness training were implemented. A simulated phishing awareness campaign was developed using Amazon Web Services (AWS) S3, CloudFront, Lambda, DynamoDB,

and CloudWatch. The campaign tracked participant interactions and provided security awareness training (SAT) to users who interacted with the simulated phishing email.

In addition to preventative security controls, an Incident Response Plan aligned with ISO/IEC 27001 principles and National Institute of Standards and Technology (NIST) guidance was developed to provide procedures for incident identification, containment, eradication, recovery, and continuous improvement.

The successful completion of this project demonstrated proficiency in systems administration, networking, cybersecurity, cloud computing, automation, troubleshooting, and technical documentation. The resulting infrastructure provides a realistic representation of a modern business environment while showcasing the technical skills I developed through the CSNT program.

Introduction

The purpose of this capstone project was to design, implement, secure, and document a complete enterprise information technology environment for a fictional company called Master Sword Publishing (MSPUB). This project served as a cumulative demonstration of the knowledge and skills acquired through the Computer Security and Network Technology (CSNT) program at Lake Washington Institute of Technology (LWTech).

Master Sword Publishing is a simulated publishing company created to provide a realistic business scenario for implementing enterprise technologies. The company required a secure and scalable infrastructure capable of supporting multiple departments, centralized user management, network services, security controls, automation, and user training initiatives. The project environment was designed to

replicate challenges commonly encountered in modern business environments, including systems administration, network management, cybersecurity operations, cloud services, and incident response planning.



Figure I-1: Master Sword Publishing (MSPUB) logo

This project was divided into multiple phases that aligned with major areas of study within the CSNT program. These phases included the creation of a virtual enterprise lab environment, implementation of Active Directory and domain services, development of Linux automation scripts, network design and troubleshooting exercise, security awareness training, and creation of an Incident Response Plan. Each phase was designed to demonstrate practical application of classroom concepts within an integrated enterprise environment.

The virtual infrastructure consisted of two Windows 2025 servers (a domain controller and domain manager), a Windows 11 workstation, Ubuntu Desktop workstation, Ubuntu Server (Web/Application Server), macOS workstation, and Kali Linux Cybersecurity workstation. These systems operated within a controlled virtual environment. Core services such as Active Directory Domain Services (AD DS), Domain Name System (DNS), web hosting, and secure file transfer (SFTP) services were configured to support business operations.

Security was incorporated throughout the project lifecycle. System hardening practices, user account management controls, phishing awareness training, and incident response procedures were implemented to reduce organization risk. A simulated phishing campaign was developed using Amazon Web Services (AWS) to evaluate user behavior and reinforce cybersecurity awareness concepts. The primary objectives of this project were to:

- Design and implement a functional enterprise network environment.
- Deploy and manage Active Directory Domain Services.
- Configure secure Windows, Linux, and macOS systems.
- Develop automation solutions using Python.
- Demonstrate troubleshooting and problem-solving methodologies.
- Implement security controls and user awareness training.
- Create an Incident Response Plan aligned with ISO/IEC 27001 principles and NIST guidance.
- Produce comprehensive technical documentation and project reporting.

The successful completion of these objectives demonstrates the ability to plan, deploy, secure, manage, and document enterprise technology solutions while applying industry best practices and professional standards.

Methodology

The capstone project was completed using a structured approach that combined planning, implementation, testing, troubleshooting, and documentation. The overall objective was to design and deploy a secure enterprise environment for Master

Sword Publishing (MSPUB) while demonstrating skills developed through the Computer Security and Network Technology (CSNT) program.

The project began with the planning phase. Project requirements were reviewed and organized into individual sections based on course objectives. A virtual business environment was designed to support multiple operating systems, centralized authentication, network services, security controls, automation, and user training initiatives. A phased implementation strategy was adopted to ensure each component could be tested and validated before integrating into the larger environment.

Parallels Desktop was selected as the primary virtualization hypervisor because after initial testing I identified limitations with VMware Fusion when working with macOS virtual machines. The virtual environment consisted of seven virtual machines, including Windows Server 2025, Windows 11, Ubuntu Desktop, Ubuntu Server, macOS Tahoe, and Kali Linux systems. This configuration provided a realistic enterprise environment capable of supporting cross-platform administration and security testing.

Infrastructure deployment followed a layered approach. Core services such as Active Directory Domain Services (AD DS), Domain Name System (DNS), and user authentication were implemented first. Workstations were then joined to the domain and tested to verify communication and authentication functionality. Network configurations were validated throughout the deployment process to ensure reliable connectivity between systems.

Automation solutions were developed using Python and tested within isolated environments before deployment into the production lab. The first automation project focused on web server deployment and configuration, including Nginx and Secure File

Transfer Protocol (SFTP) services. The second automation project focused on user account administration, providing functionality for account creation, group management, batch imports, account removal, and activity logging. Individual functions were tested and validated prior to production deployment.

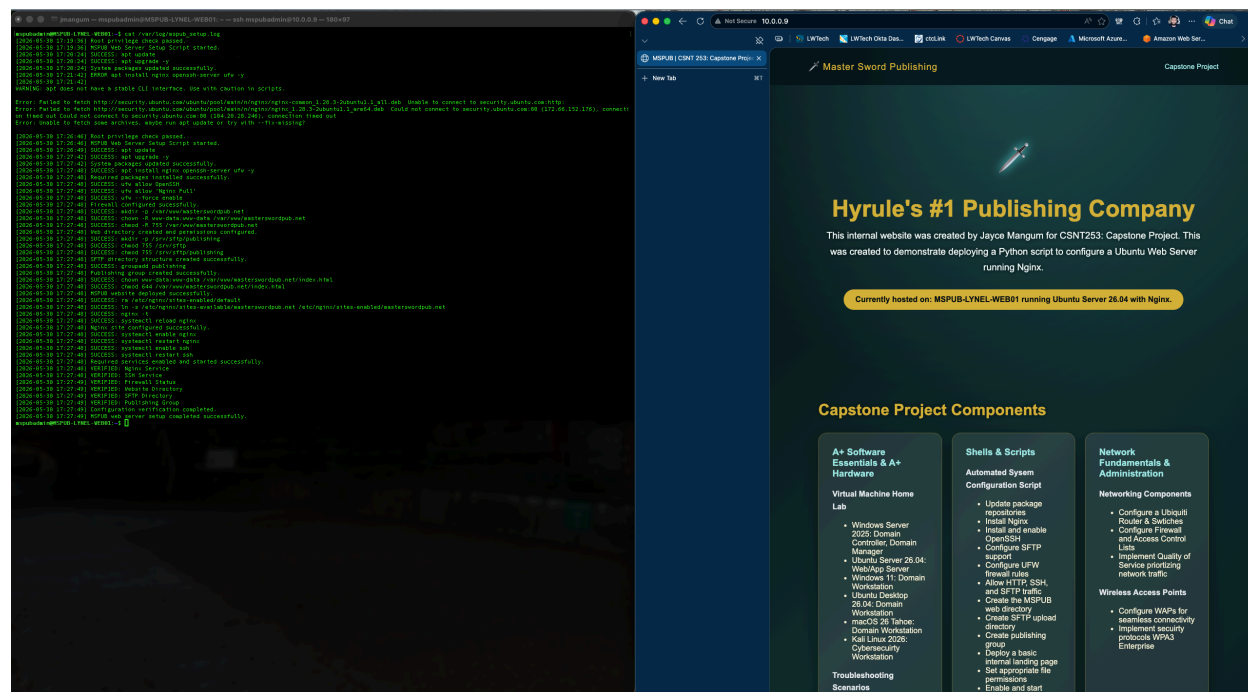


Figure M-3 Web Server Configuration script ran and successful webpage deployment

Security was incorporated throughout the project development. Operating system security features, account management controls, and hardening measures were implemented on Windows, Linux, and macOS systems. A simulated phishing awareness campaign was developed using Amazon Web Services (AWS), including Amazon S3, CloudFront, Lambda, DynamoDB, and CloudWatch services. The campaign was designed to evaluate user interaction with phishing emails and provide security awareness training through a controlled environment.

Testing and validation were performed continuously during implementation. Configuration changes were verified through system testing, connectivity testing, authentication validation, and functionality checks. Troubleshooting methodologies were applied whenever issues were encountered, including Active Directory replication problems, operating system integration challenges, DNS configuration issues, and virtual network connectivity problems.

Comprehensive documentation was maintained throughout the duration of the project. Screenshots, configuration settings, testing results, troubleshooting procedures, and implementation notes were collected during each phase, compiled into a 311 page document. The documentation served as evidence of successful implementation and provided a record of decisions, challenges, and solutions encountered during the project.

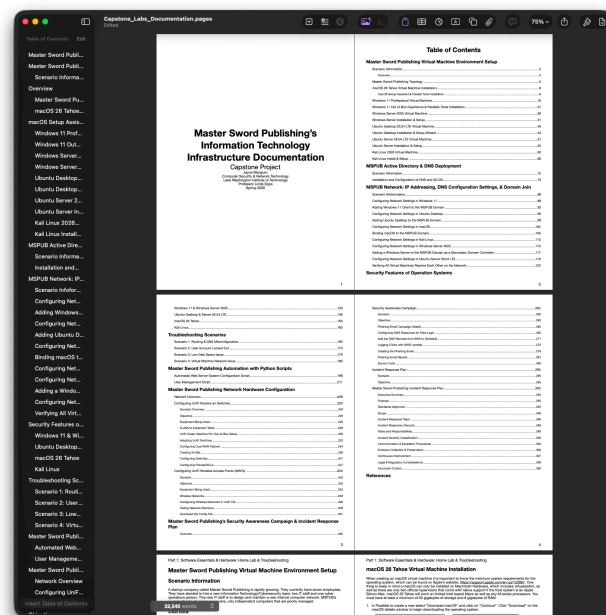


Figure M-4: Screenshot of the capstone documentation process

This methodology ensured that each project component was planned, implemented, tested, secured and documented according to industry best practices while meeting the objectives of the initial capstone project.

Network Design

Virtual Network Configuration

The network infrastructure for Master Sword Publishing (MSPUB) was designed to provide centralized management, secure communications, scalability, and support for multiple operating systems and business functions. The network architecture was developed using enterprise networking principles that emphasize segmentation, security, and reliability. The virtual environment utilized a private IPv4 addressing schema and centralized domain services provided by Active Directory Domain Services (AD DS). All systems were configured to communicate within a common network environment while maintaining centralized authentication and resource management.

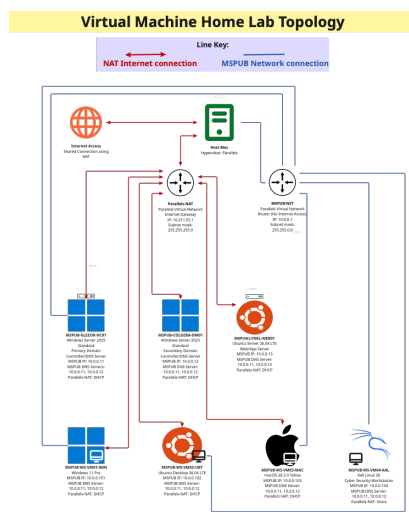


Figure ND-1 MSPUB Virtual Lab Topology diagram
Created Using Miro.com: [View Larger Image](#)

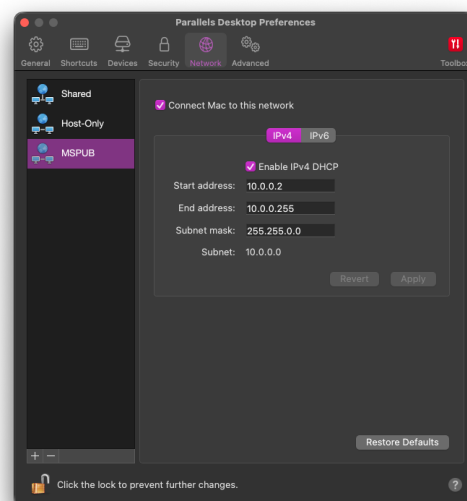


Figure ND-2: MSPUB Parallels Virtual Network Settings

Active Directory Infrastructure

The domain masterswordpub.net was created as the primary Active Directory environment for the organization. Two Windows Server 2025 systems were deployed to provide directory services and redundancy.

Server Name	Server Type	Role	NetBIOS Name
MSPUB-GLEEOK-DC01	Domain Controller	Primary Domain Controller + DNS	MSPUB-DC01
MSPUB-COLGERA-DM01	Domain Manager	Secondary Domain Controller + DNS	MSPUB-DM01

Table ND-1: MSPUB Server Roles

The domain controllers provide centralized authentication, authorization, DNS services, and directory replication. The secondary domain controller was implemented to improve availability and provide redundancy for critical directory services.

Network Addressing

The lab environment utilized two network adapters. One was used as the default Parallels Network Address Translation (NAT) to share internet access with my host MacBook Pro. The second network adapter was used as Intranet for the private MSPUB Virtual Network. The MSPUB Virtual Network utilized a private IP addressing scheme based a class A network, 10.0.0.0/16.

Server Name	Type	IP Address	Gateway/ Router	Subnet Mask	DNS
MSPUB-GLEEOK-DC01	Domain Controller + Primary DNS	10.0.0.11	10.0.0.1	255.255.0.0	10.0.0.11 10.0.0.12
MSPUB-COLGERA-DM01	Domain Manager + Secondary DNS	10.0.0.12	10.0.0.1	255.255.0.0	10.0.0.11 10.0.0.12
MSPUB-LYNEL-WEB01	Ubuntu Web/App Server	10.0.0.13	10.0.0.1	255.255.0.0	10.0.0.11 10.0.0.12

Server Name	Type	IP Address	Gateway/ Router	Subnet Mask	DNS
MSPUB-WS-VM01-WIN	Windows 11 Workstation	10.0.0.101	10.0.0.1	255.255.0.0	10.0.0.11 10.0.0.12
MSPUB-WS-VM02-UBT	Ubuntu Workstation	10.0.0.102	10.0.0.1	255.255.0.0	10.0.0.11 10.0.0.12
MSPUB-WS-VM03-MAC	macOS Workstation	10.0.0.103	10.0.0.1	255.255.0.0	10.0.0.11 10.0.0.12
MSPUB-WS-VM04-KAL	Kali Linux Cybersecurity Workstation	10.0.0.104	10.0.0.1	255.255.0.0	10.0.0.11 10.0.0.12

Table ND-2: MSPUB Network IP Addressing and DNS configuration

This addressing strategy simplified management and provided sufficient address space for future expansion.

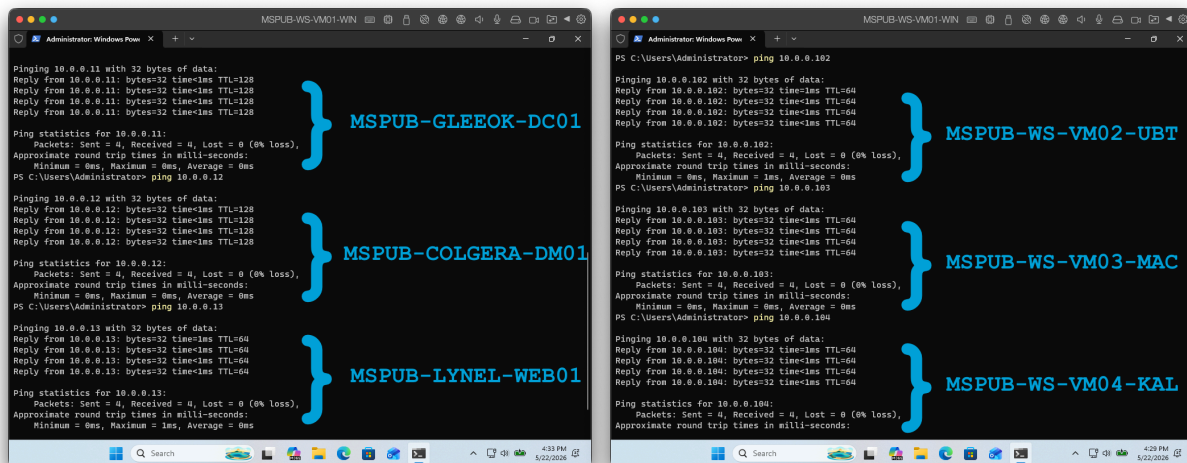


Figure ND-3: Verification of network connectivity between MSPUB servers and workstations.

System Integration

The network was designed to support multiple operating systems within a single enterprise environment. Windows, Ubuntu Linux, macOS, and Kali Linux systems were integrated into the infrastructure to demonstrate cross-platform administration and interoperability. Authentication services were centralized through Active Directory

whenever possible, allowing users to access resources using a common set of credentials. DNS services were configured to support name resolution throughout the environment and facilitate communication between systems.

Enterprise Networking

An additional physical network was configured for the Master Sword Publishing (MSPUB) network environment. This network was designed to provide secure, scalable, and highly available connectivity for employees, servers, cloud services, wireless devices, and remote users. The design incorporates enterprise networking principles including segmentation, redundancy, centralized management, and defense-in-depth security controls. The network architecture is centered around the Ubiquiti UniFi platform and utilizes a combination of managed switching, wireless network, virtual local area networks (VLANs), and centralized security controls. The design supports both current business requirements and future organization growth.

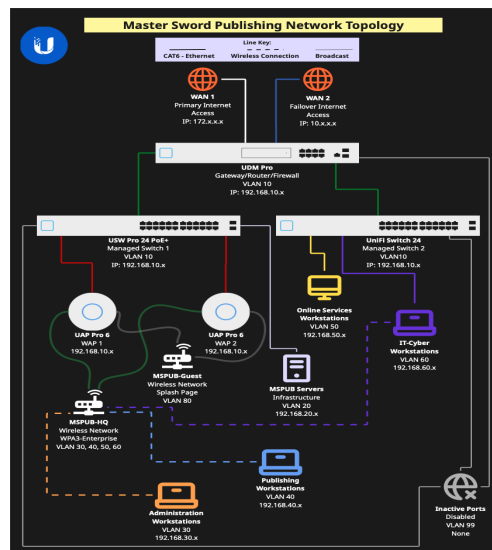


Figure M-2 MSPUB Networking
Topology diagram
Created Using Miro.com: [View
Larger Image](#)

Core Network Infrastructure

The proposed MSPUB network utilizes enterprise-grade UniFi equipment to provide routing, switching, wireless connectivity, and centralized management. The key infrastructure components include:

- UniFi Dream Machine Pro
- Router & Firewall
- UniFi Switch Pro 24 port PoE+
- UniFi Switch 24
- 2 UniFi Access Point WiFi 6 Professional
- Wireless Access Points

The network is managed through a centralized UniFi Controller platform that provides monitoring, configuration management, security policies, and device administration, called UniFi OS.

Network Segmentation

To improve security and limit lateral movement, the network is segmented using VLAN technology. Department resources, servers, management systems, guest devices, and IoT devices are separated into dedicated network segments. Network segmentation provides improved security, reduced broadcast traffic, simplified management, access control enforcement, and isolation of sensitive systems. Inter-VLAN communication is controlled through firewall policies and access control rules.

Purpose	Infrastructure	Servers	Administration	Publishing	Online Services	IT-Cyber	Guest Network	Blackhole (Disabled Ports)
Minimum Host Requirement	5	3	2	4	1	4	254	0
CIDR	192.168.1.0/29	192.168.20.0/29	192.168.30.0/29	192.168.4.0/29	192.168.5.0/30	192.168.60.0/28	192.168.80.0/24	No DHCP

Purpose	Infrastructure	Servers	Administration	Publishing	Online Services	IT-Cyber	Guest Network	Blackhole (Disabled Ports)
Gateway	192.168.1.0.1	192.168.20.1	192.168.30.1	192.168.4.0.1	192.168.5.0.1	192.168.60.1	192.168.80.1	None
Mask	255.255.255.248	255.255.255.248	255.255.255.248	255.255.255.248	255.255.255.252	255.255.255.240	255.255.255.0	N/A
IP Address Range	192.168.1.0.2 - 192.168.1.0.6	192.168.20.2 - 192.168.20.6	192.168.30.2 - 192.168.30.6	192.168.4.0.2 - 192.168.4.0.6	192.168.5.0.2 - 192.168.5.0.2	192.168.60.2 - 192.168.60.14	192.168.80.2 - 192.168.254	N/A
Broadcast	192.168.1.0.7	192.168.20.7	192.168.30.7	192.168.4.0.7	192.168.5.0.3	192.168.60.15	192.168.80.255	None
Usable Hosts	6	6	6	6	2	10	254	0
VLAN ID	10	20	30	40	50	60	80	255

Table ND-3: VLAN Segmentation

Name	VLAN ID	Router	Subnet	IPv6 Subnet	DHCP	IP Leases	Available
• Default	1	MSPUB-UDMP	192.168.1.0/24	-	Server	0/249	249
• Infrastructure	10	MSPUB-UDMP	192.168.10.0/29	-	Server	2/5	2
• Servers	20	MSPUB-UDMP	192.168.20.0/29	-	Server	0/5	5
• Administration	30	MSPUB-UDMP	192.168.30.0/29	-	Server	0/5	5
• Publishing	40	MSPUB-UDMP	192.168.40.0/29	-	Server	0/5	5
• Online Services	50	MSPUB-UDMP	192.168.50.0/30	-	Server	0/1	1
• IT-Cyber	60	MSPUB-UDMP	192.168.60.0/28	-	Server	0/13	13
• Guest	80	MSPUB-UDMP	192.168.80.0/24	-	Server	2/249	247
• Blackhole	255	MSPUB-UDMP	192.168.255.252/30	-	None	-	-

Figure M-3: UniFi Network VLAN Configuration

Wireless Infrastructure

Wireless connectivity is provided through UniFi WiFi 6 Professional Access Points configured to support employee, guest, and administrative wireless networks. Wireless security features included: WPA3 Personal Authentication, Guest Network Isolation, Captive Portal Access, Network Segmentation. These controls help protect organization resources while providing secure wireless access for employees and visitors.

SSID	Encryption Method	Password	Supported VLANs
MSPUB-HQ	WPA2/WPA3 - Personal	Hyrul3W@rr10r\$	30, 40, 50, 60
MSPUB-Guest	Captive Portal	M4\$t3r_5w0rd	80

Table ND-4: MSPUB Wireless Networks



Figure ND-4: MSPUB-HQ using WPA2/WPA3 Encryption to connect

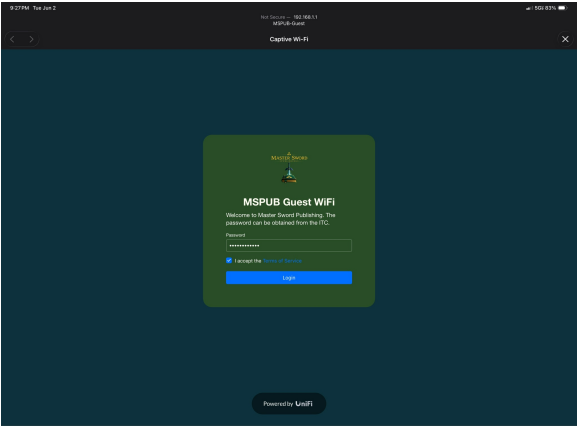


Figure ND-5: MSPUB-Guest Captive Portal

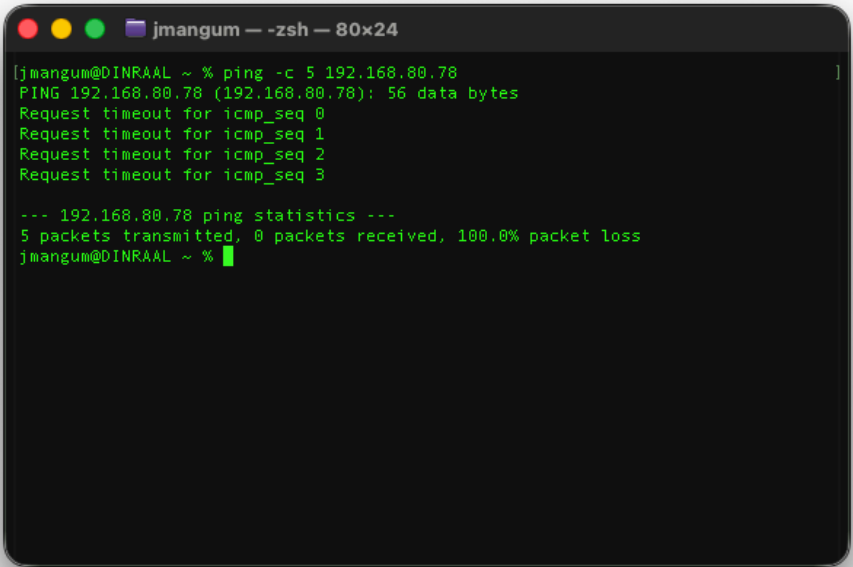


Figure ND-8: Failed Ping to internal devices from the guest network

High Availability & Redundancy

The MSPUB network incorporates redundant internet connectivity to improve service availability. A primary high-speed fiber connection is supplemented by a secondary business internet connection to provide failover capabilities in the event of a service interruption. Critical services such as Active Directory, DNS, and cloud-hosted applications are designed to support continued business operations during network disruptions.

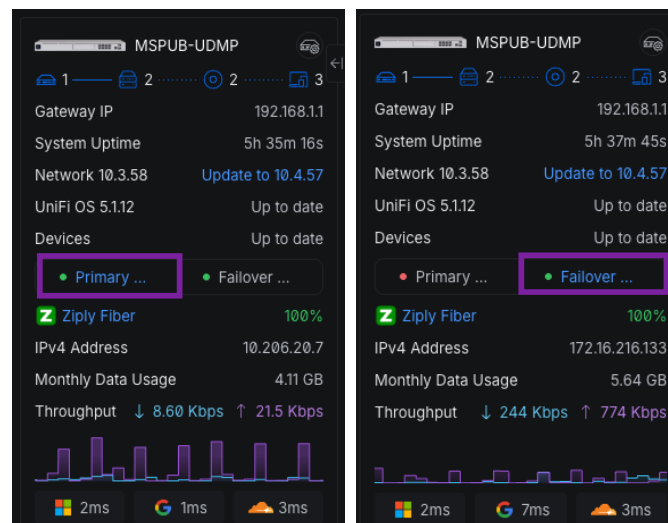


Figure ND-6: MSPUB Internet Failover Testing

Infrastructure Implementation

The infrastructure implementation phase focused on deploying and configuring the core services required to support the Master Sword Publishing (MSPUB) enterprise environment. The implementation included Active Directory Domain Services (AD DS), Domain Name System (DNS), Windows and Linux server deployments, workstation integration, web services, and centralized user management.

Virtual Lab Environment

A virtual enterprise was developed using Parallels Desktop to simulate a production business network. The virtual environment consisted of seven virtual machines representing both server and workstation operating systems. The virtual environment provided a controlled platform for testing enterprise services, security controls, automation solutions, and troubleshooting procedures.

Server Name	Operating System	Purpose
MSPUB-GLEEOK-DC01	Windows Server 2025	Domain Controller
MSPUB-COLGERA-DM01	Windows Server 2025	Domain Manager (Secondary Domain Controller)
MSPUB-LYNEL-WEB01	Ubuntu Server 26.04	Web/App Server

Table II-1: MSPUB Servers

Workstation Name	Operating System	Purpose
MSPUB-WS-VM01-WIN	Windows 11	Windows Workstation
MSPUB-WS-VM02-UBT	Ubuntu Desktop 26.04	Linux Workstation
MSPUB-WS-VM03-MAC	macOS 26 Tahoe	Apple Workstation
MSPUB-WS-VM04-KAL	Kali Linux 2026	Security Testing Workstation

Table II-2: MSPUB Workstation Systems

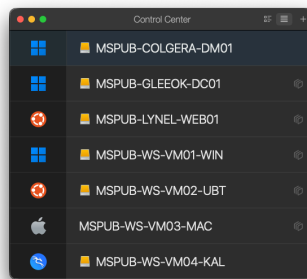


Figure II-1: MSPUB Virtual Machine Environment

Active Directory Domain Services

Active Directory Domain Services (AD DS) was implemented to provide centralized authentication, authorization, and directory management for MSPUB users and systems. The domain masterswordpub.net was created and configured with the NetBIOS name MSPUB. Organization Units (OUs), security groups, and user accounts were created to simulate a real-world business environment consisting of multiple departments and employees. MSPUB-COLGERA-DM01 was deployed as a secondary domain controller to provide redundancy and improve service availability. Active Directory replication was configured and validated to ensure directory information remained synchronized between both domain controllers.

Key AD DS functions included:

- Centralized user authentication
- Group-based permissions
- Organization Unit management
- Domain-wide security policies
- Directory replication

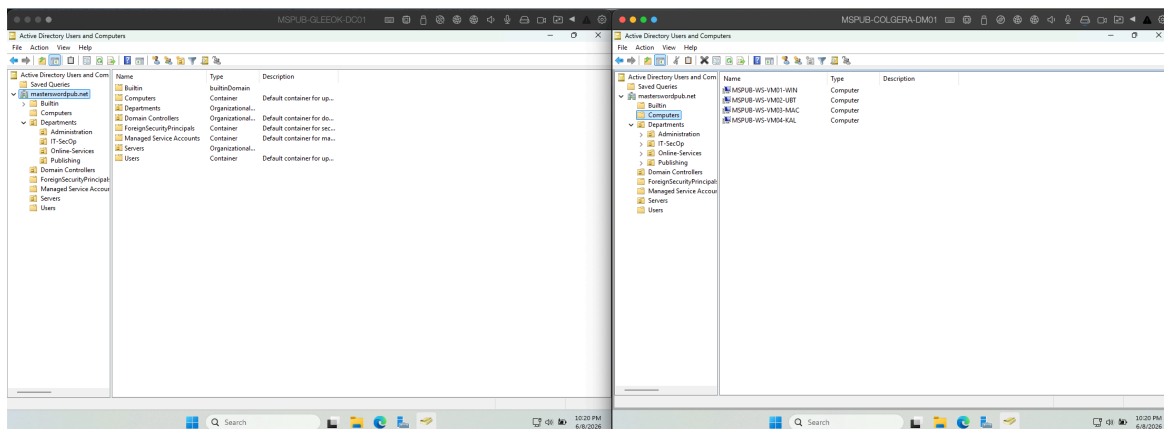


Figure II-2: AD Users and Computers: to show redundancy between the two domain controllers

DNS Services

Domain Name System (DNS) services were configured on the domain controllers to provide name resolution throughout the MSPUB environment. DNS replication was verified following deployment of the secondary domain controller to ensure consistency throughout the environment.

DNS Records were created to support:

- Domain controller discovery
- Workstation communication
- Active Directory services
- Server resource access

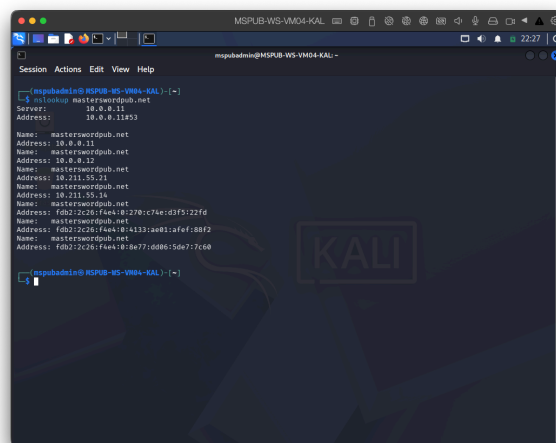


Figure II-3: Using the command nslookup to show DNS resolve of the two domain controllers for masterswordpub.net on MSPUB workstation

Each workstation was configured to point to the DNS servers hosted on the Domain Controller (10.0.0.11) and Domain Manager (10.0.0.12). On Windows this is configured by going to Settings > Network & Internet settings > Advanced network settings > More adapter options: Edit > Internet Protocol Version 4 (TCP/IPv4) and entering the Preferred and Alternate

DNS server IP Addresses. On macOS this is configured by going to System Settings > Network > [Ethernet or Wi-Fi Adapter] > Details > DNS and adding the DNS Servers. On Ubuntu Desktop this is configured by going to Indicators menu > Network Connections > Network Connection Settings > [Network Adapters] Network Options > IPv4 and entering the DNS server IP addresses. It is also a best practice to give the system a Static IP address in these settings.

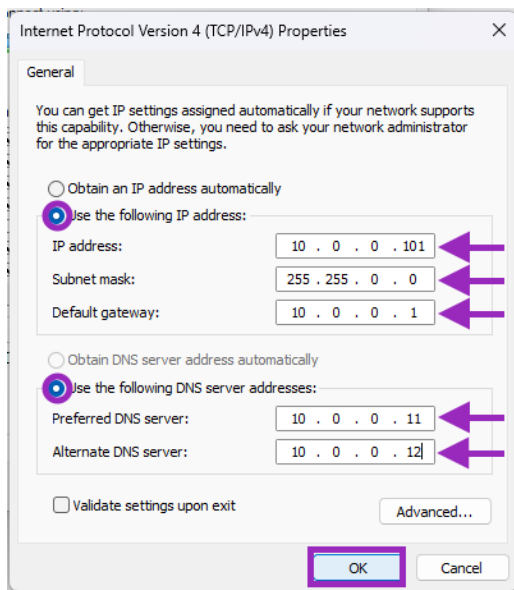


Figure II-4: Configuring Windows IP and DNS settings

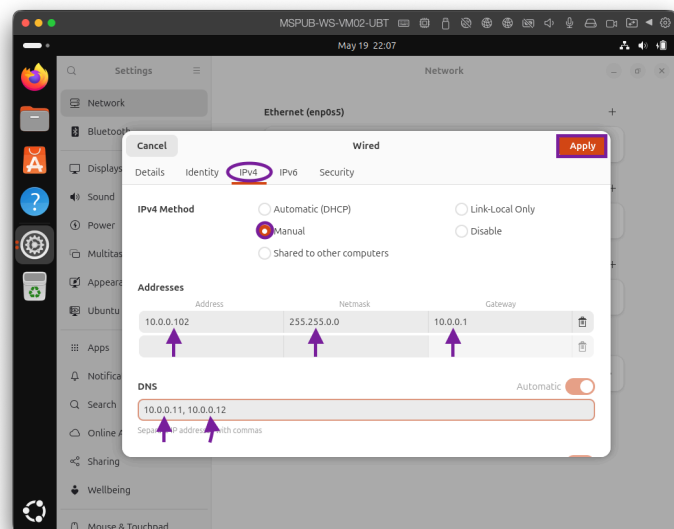


Figure II-5: Configuring Ubuntu Desktop IP and DNS settings

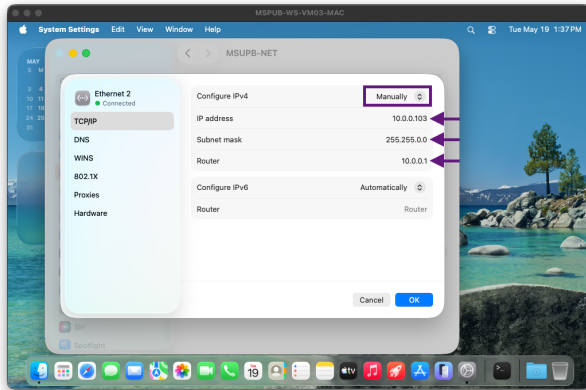


Figure II-6: Configuring macOS IP settings

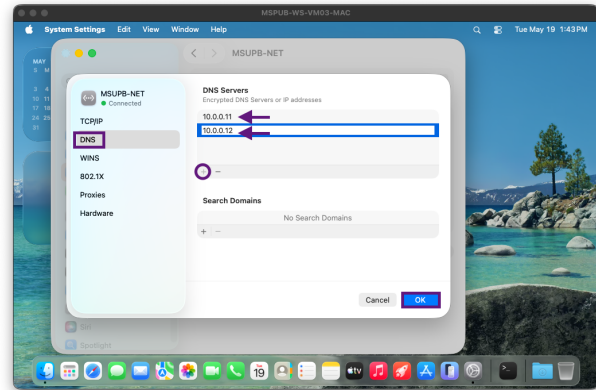


Figure II-7: Configuring macOS DNS settings

Workstation Integration

Multiple operating systems were integrated into the MSPUB domain environment to demonstrate cross-platform administration capabilities. Windows 11 workstations were joined to the Active Directory domain and successfully authenticated using domain users' accounts, e.g. MSPUB\z.princess. Windows machines can be added to the domain by entering the PowerShell command "Add-Computer -DomainName "masterswordpub.net" -Credential "MSPUB\Administrator" -Restart" and authenticating by entering into a Windows Terminal session and authenticating by using an domain administrator account's credentials. After a restart, domain users can log in using their domain user account credentials.

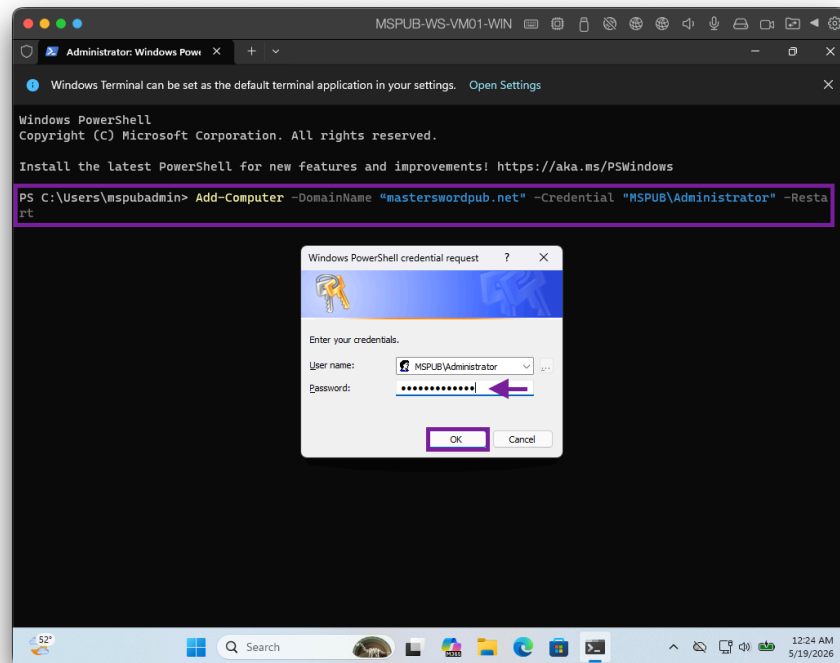


Figure II-8: Adding a Windows workstation to a domain using terminal

macOS systems were integrated into the domain by using the Directory Utility tool and configured to authenticate against Windows Server Active Directory services. This can be configured by going to System Settings > Users & Groups > Network account server: Edit > Open Directory Utility > Active Directory and entering in masterswordpub.net domain details. Administrators will have to authenticate first with a local Mac administrator account and the Active Directory domain Administrator account. After a system restart domain users will be able to login using their domain credentials, e.g. MSPUB\s.zora.

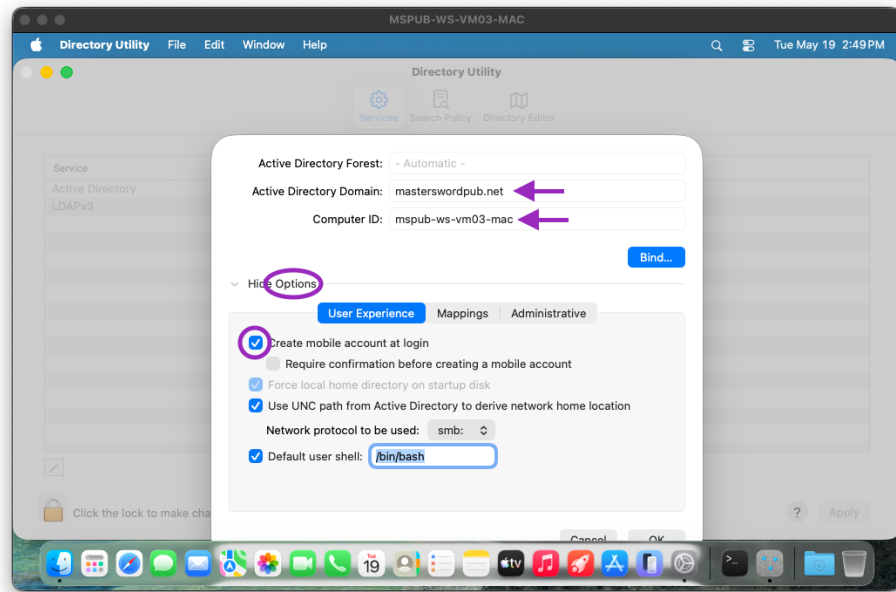


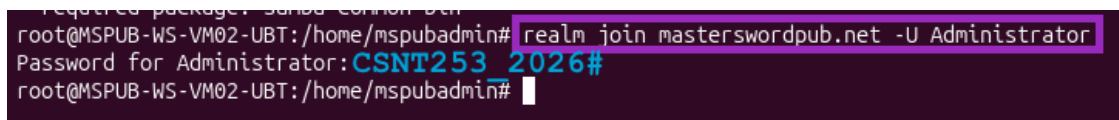
Figure II-9: Configuring a macOS workstation to a Windows Server Active Directory domain

Ubuntu Desktop systems were configured to communicate with domain services and participate within the enterprise environment. This was done by installing dependencies through the Linux command line interface (CLI) using bash. To allow domain join within the Ubuntu virtual machine included: RealMD, System Security Services Daemon, PAM Library, Active Directory CLI, Samba, and Package Kit. The following commands were entered to install these dependencies:

- Update the package library: `sudo apt update`
- Install the dependency packages: `sudo apt install realm sssd sss-tools libnss-sss libpam-sss adcli samba-common-bin package kit -y`

A reboot will be required once packages are successfully installed. The system can be added to the domain by entering the command `sudo realm join -v`

masterswordpub.net -U Administrator then authenticating by entering the domain administrator's credentials. An extra command needs to be entered to allow Ubuntu to automatically create home directories to the system, `sudo pam-auth-update -enable mkhomedir`. User can now login using their domain user account, e.g. m.twili@masterswordpub.net.



```
root@MSPUB-WS-VM02-UBT:/home/mspubadmin# realm join masterswordpub.net -U Administrator
Password for Administrator:CSNT253 2026#
root@MSPUB-WS-VM02-UBT:/home/mspubadmin#
```

Figure II-10: Ubuntu Desktop workstation domain join using CLI tools

The Kali Linux workstation was not added to the Active Directory domain. This system is a standalone workstation for use by the internal Cybersecurity team. This system is also only to be used for security validation activities and during incident response efforts. The successful integration of multiple operating systems demonstrated interoperability within a centralized enterprise environment.

Linux Web Services

A Ubuntu Server was deployed as a designated web server platform to host an internal website. Secure File Transfer Protocol (SFTP) services were also configured to support secure file access and management for website administrators. The server hosted web content developed during the project and served as the primary platform for automation testing and deployment. Services configured on the server included:

- Secure Shell Protocol (SSH)
- User Management Functions
- Nginx Web Server
- Security Logging
- SFTP Services

These configurations supported centralized management and aligned with enterprise identity management best practices.

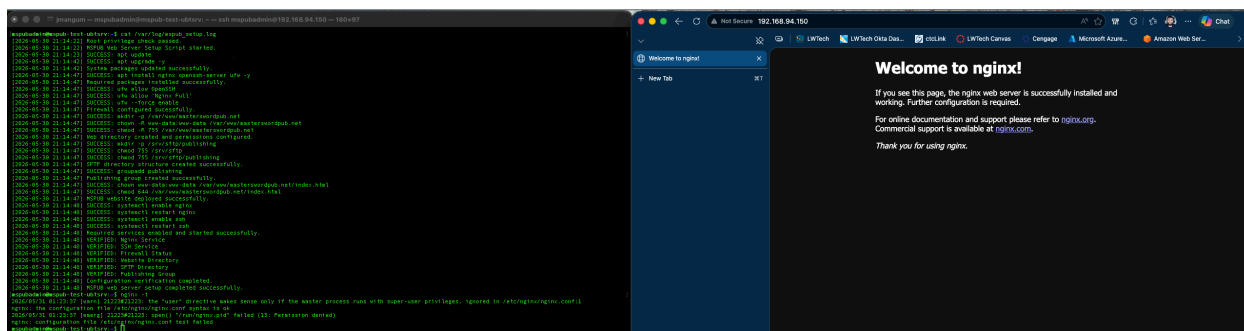


Figure II-11: Linux Web Server configured with Nginx

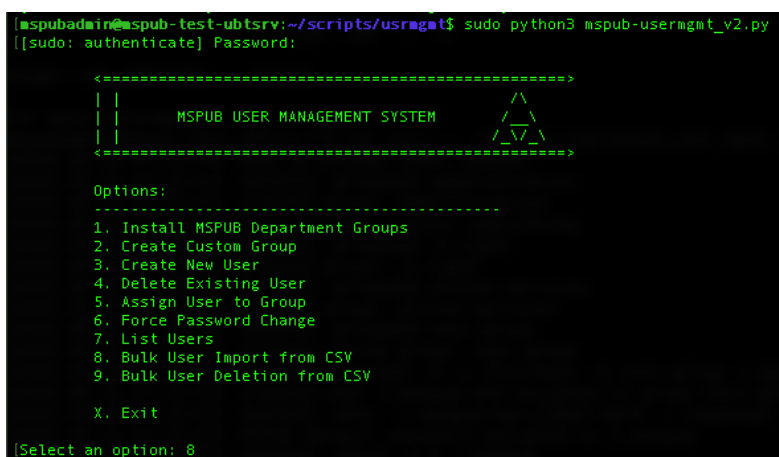


Figure II-12: Linux User Management menu-based script

AWS Cloud Infrastructure

For a security awareness training campaign cloud based services were deployed using Amazon Web Services (AWS). The cloud infrastructure consisted of these AWS resources:

- Amazon S3
- Amazon CloudFront
- AWS Lambda
- Amazon DynamoDB
- Amazon CloudWatch

These services were integrated to deliver phishing simulation content, track user interactions, collect campaign metrics, and provide awareness training materials. The cloud infrastructure enabled real-time event logging and provided measurable data regarding participant engagement and training completion.

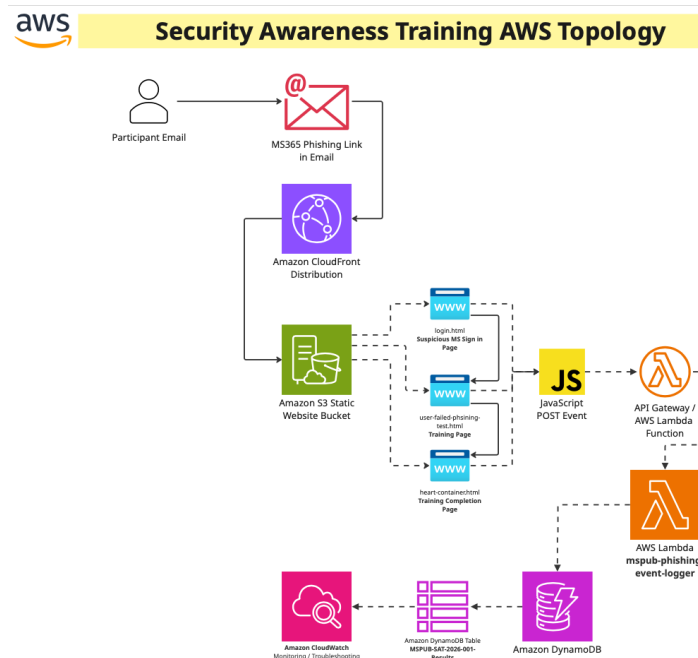


Figure II-13: AWS Security Awareness Training architecture showing participant interaction, event processing, logging, and reporting components.
[View larger image](#)

Implementation Results

The infrastructure implementation phase successfully delivered a functional enterprise environment consisting of Windows, Linux, and macOS systems integrated through centralized directory services. Core network services, automation solutions, web services, and cloud-based security awareness tools were successfully deployed and validated. The resulting environment provided the foundation required to support

testing, troubleshooting, security analysis, automation development, and incident response activities throughout the capstone project.

Linux Automation

Automation was implemented with the Master Sword Publishing (MSPUB) environment to reduce administrative efforts, improve consistency, efficiency, and to demonstrate practical scripting skills using the Python scripting language. Two automation solutions were developed and deployed on the Ubuntu Server to support business operations and system administration activities. The automation projects focused on web server configuration deployments and user account management. The configuration script executed commands as soon as it launched from the CLI. While the user management solution utilizes a menu-driven interface. Both scripts were tested extensively prior to deployment.

Automated System Configuration Script

The first automation task focused on automating the deployment and configuration of web services on the MSPUB-LYNEL-WEB01 Ubuntu Server. The objective was to reduce manual configuration tasks and provide a repeatable deployment process for server provisioning. This solution was developed using Python and designed to execute a predefined sequence of configuration tasks immediately after launch. Rather than relying on manual administration, the script automated the installation and configuration of required services used with the MSPUB environment. The script performed these functions:

- Installation and configuration Nginx
- Removal of the default Nginx website
- Deployment of website content

- Creation of SFTP services
- Validation of service operation

The script provided a consistent deployment process that reduces administrative effort and minimized configuration errors. Testing confirmed that all required services were successfully installed and operational following execution of the script.

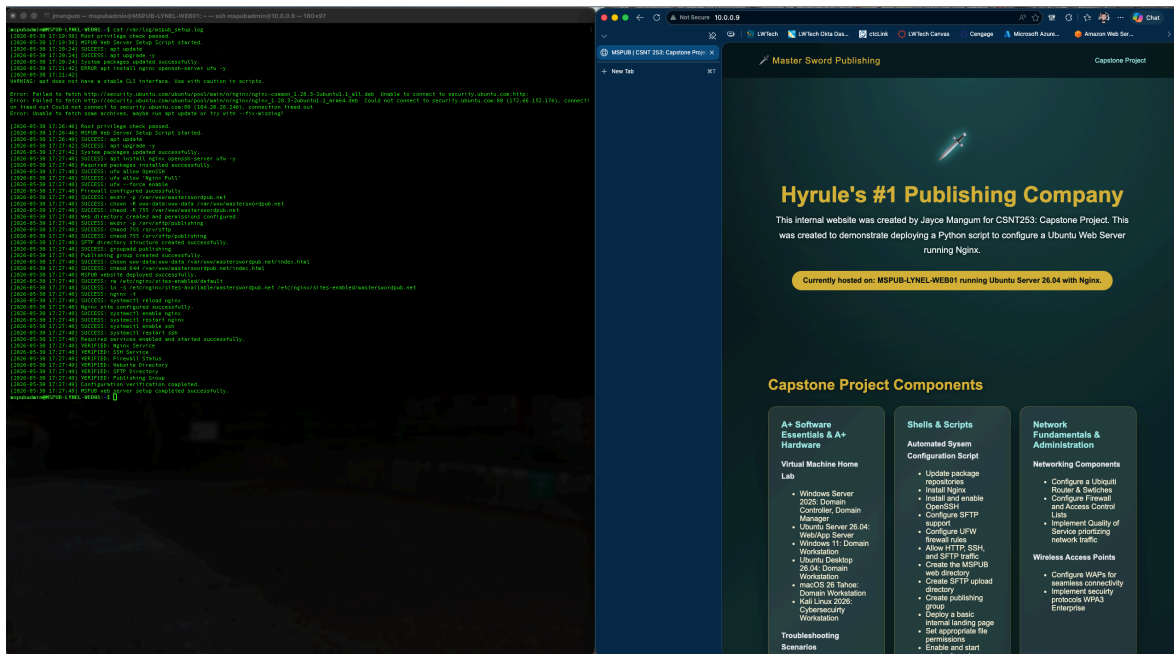


Figure LA-1: Executing Web Server Configuration script and end result Website deployment

User Management Automation Script

The second automation script task focused on account administration and user lifecycle management. The goal was to streamline account creation and maintenance activities while reducing administrative overhead. This solution was developed using Python and deployed on MSPUB-LYNEL-WEB01. The script provided administrators with a menu-driven interface capable of managing users, groups, and account-related activities. The key functions include:

- Create user accounts
- Create groups

- Add users to groups
- Remove user accounts
- Remove groups
- List groups
- Batch user import
- Bulk user deletion
- Administrative logging

The script also provided automated account provisioning through CSV imports, allowing administrators to create multiple users simultaneously.

```

aspubadmin@aspub-test-ubuntu:~/scripts/usrmgmt$ sudo python3 aspub-usermgmt_v2.py
[sudo: authenticate] Password:

=====
| | MSPUB USER MANAGEMENT SYSTEM | |
| |                               | |
| |                               | |
=====

Options:
-----
1. Install MSPUB Department Groups
2. Create Custom Group
3. Create New User
4. Delete Existing User
5. Assign User to Group
6. Force Password Change
7. List Users
8. Bulk User Import from CSV
9. Bulk User Deletion from CSV

X. Exit

Select an option: 8

```

Figure LA-2: MSPUB User Management Script menu options

```

Bulk User Import
=====
CSV file location: /opt/aspub-usermgmt/bulk-user-import.csv

Instructions:
1. Open the CSV file in a text editor.
2. Add one user per line using this format:
   first_name.last_name.group_name
   per the MSPUB naming convention for usernames first_name = first initial dot last_name = full last name. E.g. p.sheikah
3. Save the CSV file before continuing.
4. Existing users will be skipped.
5. Missing groups will be created automatically.

mCSV file Example:
r.sheikah,IT-cyber
l.hero,IT-cyber
z.princess,administration

Ready to begin the import? y/n
Batch import cancelled.

Bulk User Deletion
=====
CSV file location: /opt/aspub-usermgmt/bulk-user-delete.csv

Instructions:
1. Open the CSV file in a text editor.
2. Add one user per line using this format:
   username
3. Save the CSV file before continuing.
4. User listed in the file will be deleted with their home directories.

Example:
z.princess
l.sheikah

[Are you sure you want to delete all users listed in this file? (y/n): n
Bulk user deletion cancelled.

```

Figure LA-3: CSV Import/Deletion features of the MSPUB User Management script

Logging and Auditing

Administrative actions were recorded through a centralized logging system.

Logs were stored locally and provided visibility in account management activities performed through the application. The logging system supported:

- Administrative accountability
- Activity tracking
- Troubleshooting
- Change auditing

Testing and validation of the script can be confirmed within the logs located in `/var/`

`log/mspub_user_mgmt.log`. The results confirmed:

- User account creation
- Group assignments
- Batch imports
- Bulk deletions
- Log generation
- Permission verification

```

mspubadmin@mspub-test-ubtsrv:~/scripts/usrmgmt$ cat /var/log/mspub_user_mgmt.log
[2026-05-31 07:13:44] Root privilege check passed.
[2026-05-31 07:15:01] SUCCESS: groupadd administration
[2026-05-31 07:15:01] Created group: administration
[2026-05-31 07:15:01] Group already exists: publishing
[2026-05-31 07:15:01] SUCCESS: groupadd it-cyber
[2026-05-31 07:15:01] Created group: it-cyber
[2026-05-31 07:15:01] SUCCESS: groupadd online-services
[2026-05-31 07:15:01] Created group: online-services
[2026-05-31 07:15:24] SUCCESS: groupadd test-group
[2026-05-31 07:15:24] Created custom group: test-group
[2026-05-31 07:15:57] SUCCESS: useradd -m -s /bin/bash -G test-group j.mangum
[2026-05-31 07:15:57] Created user j.mangum and assigned to group test-group
[2026-05-31 07:15:57] SUCCESS: echo 'j.mangum:Hyrul3Warr10r$' | chpasswd
[2026-05-31 07:15:57] MSPUB default password assigned to j.mangum
[2026-05-31 07:15:57] SUCCESS: chage -d 0 j.mangum
[2026-05-31 07:15:57] Forced password change at next login for j.mangum
[2026-05-31 07:17:05] Group assignment failed. User does not exist: it-cyber
[2026-05-31 07:17:30] SUCCESS: usermod -ag it-cyber j.mangum
[2026-05-31 07:17:30] Added j.mangum to group it-cyber
[2026-05-31 07:18:01] SUCCESS: chage -d 0 j.mangum
[2026-05-31 07:18:01] Forced password change for j.mangum
[2026-05-31 07:18:13] Displayed user list.
[2026-05-31 07:18:29] Exited MSPUB User Management System.
[2026-05-31 07:21:23] Root privilege check passed.
[2026-05-31 07:21:35] SUCCESS: userdel -r j.mangum
[2026-05-31 07:21:35] Deleted user: j.mangum
[2026-05-31 07:21:42] Displayed user list.
[2026-05-31 07:21:50] Exited MSPUB User Management System.
[2026-05-31 07:33:42] Root privilege check passed.
[2026-05-31 07:34:00] SUCCESS: useradd -m -s /bin/bash -G test-group j.mangum
[2026-05-31 07:34:00] Created user j.mangum and assigned to group test-group
[2026-05-31 07:34:00] MSPUB default password assigned to j.mangum
[2026-05-31 07:34:00] SUCCESS: chage -d 0 j.mangum
[2026-05-31 07:34:00] Forced password change at next login for j.mangum
[2026-05-31 07:34:12] Exited MSPUB User Management System.
[2026-05-31 21:02:28] Root privilege check passed.
[2026-05-31 21:03:00] SUCCESS: mkdir -p ./scripts/usrmgmt/
[2026-05-31 21:07:00] Root privilege check passed.
[2026-05-31 21:07:12] SUCCESS: mkdir -p /opt/mspub-usrmgmt/
[2026-05-31 21:07:12] SUCCESS: chmod 644 /opt/mspub-usrmgmt/batch-usr-import.csv
[2026-05-31 21:07:12] Created sample batch import CSV file: /opt/mspub-usrmgmt/batch-usr-import.csv
[2026-05-31 21:07:20] Batch import cancelled by administrator.
[2026-05-31 21:07:31] Exited MSPUB User Management System.
[2026-05-31 21:08:23] Root privilege check passed.
[2026-05-31 21:08:32] SUCCESS: useradd -m -s /bin/bash -G publishing l.hero
[2026-05-31 21:08:32] Batch import assigned default password to l.hero
[2026-05-31 21:08:32] SUCCESS: chage -d 0 l.hero
[2026-05-31 21:08:32] Batch import created user l.hero in group publishing. Default password set. Change password next logon: TRUE
[2026-05-31 21:08:32] SUCCESS: useradd -m -s /bin/bash -G administration z.princess
[2026-05-31 21:08:32] Batch import assigned default password to z.princess
[2026-05-31 21:08:32] SUCCESS: chage -d 0 z.princess
[2026-05-31 21:08:32] Batch import created user z.princess in group administration. Default password set. Change password next logon: TRUE
[2026-05-31 21:08:32] SUCCESS: useradd -m -s /bin/bash -G publishing s.zora
[2026-05-31 21:08:32] Batch import assigned default password to s.zora
[2026-05-31 21:08:32] SUCCESS: chage -d 0 s.zora
[2026-05-31 21:08:32] Batch import created user s.zora in group publishing. Default password set. Change password next logon: TRUE
[2026-05-31 21:09:14] Exited MSPUB User Management System.
[2026-05-31 21:10:34] Root privilege check passed.
[2026-05-31 21:10:42] Displayed user list.
[2026-05-31 21:10:56] SUCCESS: chmod 644 /opt/mspub-usrmgmt/bulk-usr-delete.csv
[2026-05-31 21:10:56] Created sample batch removal CSV file: /opt/mspub-usrmgmt/bulk-usr-delete.csv
[2026-05-31 21:11:01] Bulk user deletion cancelled by administrator.
[2026-05-31 21:11:02] Exited MSPUB User Management System.
[2026-05-31 21:13:11] Root privilege check passed.
[2026-05-31 21:13:18] SUCCESS: userdel -r z.princess
[2026-05-31 21:13:18] Bulk delete removed user: z.princess
[2026-05-31 21:13:18] Bulk delete skipped missing user: l.sheikah
[2026-05-31 21:13:18] Bulk delete skipped missing user: p.sheikah
[2026-05-31 21:13:18] Bulk user deletion completed.
[2026-05-31 21:13:51] Displayed user list.
[2026-05-31 21:13:58] Exited MSPUB User Management System.
[2026-05-31 21:43:44] Root privilege check passed.
[2026-05-31 21:44:02] Batch import cancelled by administrator.
[2026-05-31 21:44:09] Bulk user deletion cancelled by administrator.
[2026-05-31 21:45:41] Exited MSPUB User Management System.
mspubadmin@mspub-test-ubtsrv:~/scripts/usrmgmt$

```

Figure LA-4: MSPUB User Management System log file confirming all functions of the script are operational

Benefits of Automation

The implementation of automation within the MSPUB environment provided sever benefits like: reduced administrative workloads, improved configuration consistency, faster service deployment, reduced human error, improved scalability, simplified account management. These projects demonstrated how scripting and

automation can improve operational efficiency while supporting enterprise administrative requirements.

Results

Both automation scripts successfully achieved their objectives and were integrated into the MSPUB infrastructure. The solutions demonstrated applications of Python programming, Linux administration, and systems automation concepts while improving efficiency and consistency throughout the environment.

Security Implementation

Security was integrated throughout the Master Sword Publishing environment to protect organization resources, reduce risk, and support secure business operations. Security controls were implemented across servers, workstations, network infrastructure, cloud services, and user training initiatives. The objective was to establish a layered security approach that addressed both technical and human factors commonly associated with cybersecurity incidents.

Operating System Security

Security configurations were applied to all operating systems deployed within the MSPUB environment. Each platform was configured according to recommended security practices and organization requirements.

Windows Security

Windows Server 2025 and Windows 11 systems were configured to support enterprise security requirements through Active Directory integration, centralized

account management, and operating system security controls. These security measures included:

- Domain-based authentication
- Strong password requirements
- User account management
- Security updates
- Microsoft Defender protections
- Firewall configuration
- Administrative access controls

Ubuntu Linux Security

Ubuntu Desktop and Ubuntu Server systems were configured using Linux security best practices. These features included:

- Root user password change
 - User privilege separation
 - Secure Shell (SSH) administration
 - Package management updates
 - Firewall protections
 - Service hardening
 - Secure file permissions
- ## macOS Security

The macOS workstation was configured to support enterprise authentication and secure system operations. Security measures included:

- Active Directory integration
- User authentication controls
- System updates
- Application security settings
- Local firewall protections

Kali Linux Security

Kali Linux was deployed as a dedicated security testing platform. Administrative controls were implemented to ensure the system is used only for authorized testing activities within the MSPU environment.

Identify and Access Management

Identity and access management controls were implemented through Active Directory Domain Services (AD DS). Centralized authentication reduced administrative complexity and improved security through consistent account management practices.

Controls implemented included:

- Centralized user authentication
- Security group management
- Role-based access controls
- Account lifecycle management
- Password management policies

These controls helped ensure that users were granted access only to resources necessary for their assigned responsibilities.

Network Security

Network security controls were incorporated into the MSPUB network design to reduce unauthorized access and improve system protection. Implemented network controls included:

- VLAN segmentation
- Firewall policies
- Access control lists (ACLs)
- Secure wireless authentication
- Guest network isolation
- Administrative network separation

These measures supported defense-in-depth principles and reduced opportunities for lateral movement within the network.

Security Awareness Training Campaign

Recognizing that users represent a significant component of organization security, MSPUB implemented a Security Awareness Training (SAT) campaign initiative designed to educate users about phishing attacks and social engineering tactics. A simulated phishing campaign was developed to evaluate participant responses and provide immediate security awareness training when interactions occurred. The campaign objectives were to:

- Measure phishing susceptibility
- Reinforce security awareness concepts
- Improve incident reporting awareness
- Demonstrate real-world attack techniques
- Collect metrics for future improvement efforts

Cloud Architecture & Other Components

The phishing campaign used Amazon Web Services (AWS) resources to help collect data regarding the phishing campaign. Amazon S3 buckets were used to store the webpages. Amazon CloudFront was used to execute the website and functions. Amazon Lambda was used to help gather logging information for an Amazon DynamoDB table to collect user input data. Amazon Certificate Manager was used to issue an SSL/TLS certificate for HTTPS web protocol support. These services were

configured to provide secure content delivery, event processing, logging, and participant activity tracking. Cloud-based services were monitored to verify proper operation and support campaign reporting activities.

A custom subdomain was created as part of one of my domains I own through GoDaddy, masterswordpub-net.jaycemangum.com. A mask email address was also created through Google Workspace to send the suspicious email, micr0soft365-administrator@masterswordpublishing.jaycemangum.com.

Security Awareness Training Results

The campaign was sent to five classmates from the CSNT program by a randomizer, Professor Epps, and five of my personal friends. Each person was assigned a participant id and customized tracking URL for AWS DynamoDB table logging. The DynamoDB data was then tracked inside a spreadsheet tracker based on the clicks of the user. Emails were sent on June 6th, 2026 between 9:25AM PST and 10:25AM PST.

Participant ID	Person	Email Sent	Email Sent Date
participant00	Link H.	Yes	6/6/26 7:06 PM
participant01	Linda E.	Yes	6/6/26 10:25 AM
participant02	Bob H.	No	N/A
participant03	Brent B.	No	N/A
participant04	Alyx B.	No	N/A
participant05	Mihail B.	Yes	6/6/26 10:15 AM
participant06	Joshua B.	No	N/A
participant07	Tim C.	No	N/A
participant08	Keith C.	No	N/A
participant09	Iftu I.	Yes	6/6/26 10:17 AM
participant10	Mateen K.	No	N/A

Participant ID	Person	Email Sent	Email Sent Date
participant11	Akshatha K.	No	N/A
participant12	Brandon M.	No	N/A
participant13	Maksim M.	Yes	6/6/26 10:18 AM
participant14	Dylan R.	No	N/A
participant15	Zach S.	No	N/A
participant16	Hutton S.	No	N/A
participant17	Mamadou S.	Yes	6/6/26 10:20 AM
participant18	Caden T.	Yes	6/6/26 10:21 AM
participant19	Gilbert W.	No	N/A
participant37	Taylor M.	Yes	6/6/26 10:09 AM
participant38	Kali T.	Yes	6/6/26 9:30 AM
participant39	David M.	Yes	6/6/26 9:26 AM
participant40	Anna J.	Yes	6/6/26 9:55 AM
participant41	Mason H.	Yes	6/6/26 9:38 AM
participant42	Stacey M.	No	N/A
participant43	Corey M.	No	N/A
participant44	Kyrah B.	No	N/A
participant45	Tristan B.	No	N/A

Table SAT-1: Phishing Email Participants List

Of the 11 recipients who received the phishing email, 5 opened the message and entered their credentials into the fraudulent form. 4 people actually read through the training material and completed it. None of my classmates, nor Professor Epps, opened the email, so they immediately passed this SAT. The simulated phishing campaign demonstrated the effectiveness of modern email security controls. Several messages delivered to Lake Washington Institute of Technology accounts were automatically analyzed, classified as suspicious, and later removed by the email security platform before user interaction could occur. My friends that I sent it to failed

this training because they opened a suspicious link from their email and all of them entered their Microsoft Account credentials into the fraudulent form. Luckily for them, though, I did not record their inputs from the username and password fields. In an actual attack, their information would have been compromised. Luckily, most of them read the training material and received a security awareness training heart container certificate.

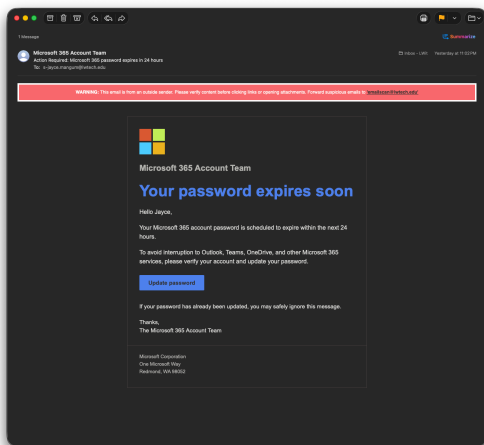


Figure SISAT-1: Phishing Email Sent to SAT participants requiring them to reset their password

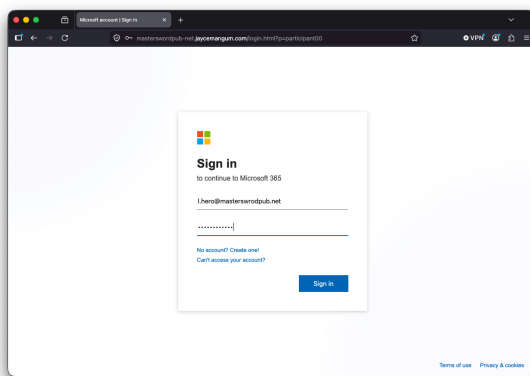


Figure SISAT-2: Fraudulent Login Form for Microsoft Credential input: <https://masterswordpub-net.jaycemangum.com/login.html?p=CAPSTONE-INQUIRY>

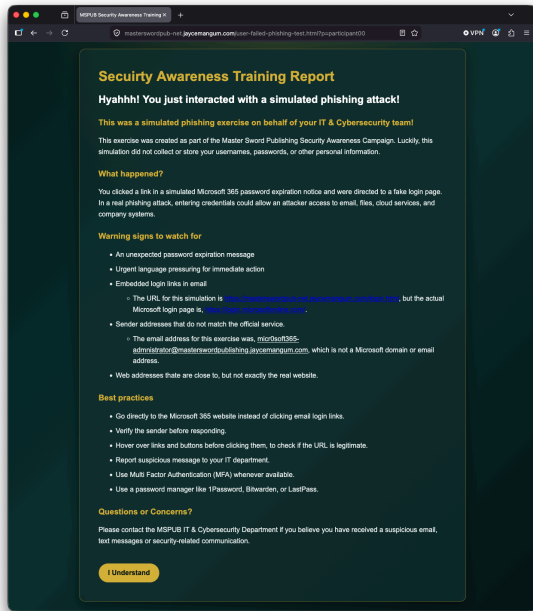


Figure SISAT-3: Security Awareness Training page users went to if they entered the credentials and clicked the “Sign In” button:
<https://masterswordpub-net.jaycemangum.com/user-failed-phishing-test.html?p=CAPSTONE-INQUIRY>

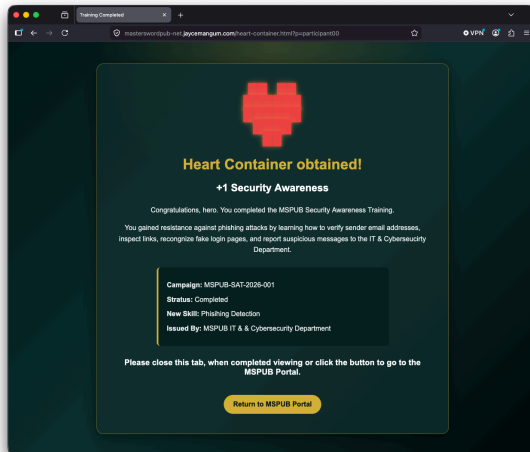


Figure SISAT-4: SAT completion page w/ a certificate: <https://masterswordpub-net.jaycemangum.com/heart-container.html?p=CAPSTONE-INQUIRY>

Completed - Items returned: 44 - Items scanned: 44 - Efficiency: 100% - RCUs consumed: 2

Table: MSPUB-SAT-2026-001-Results - Items returned (38)

Scan started on June 09, 2026, 14:19:39

event_id (String)	campaign_id	event_type	page	participant_id	timestamp
daf649d0-987d-48fa...	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant00	2026-06-06T22:39:22.327141+00:00
8978a452-a242-4c8d...	MSPUB-SAT-2026-001	training_page_view	user-failed-phishing-test.html	participant00	2026-06-06T22:40:22.062894+00:00
a079343a-c841-407b...	MSPUB-SAT-2026-001	login_attempt	login.html	participant00	2026-06-06T22:40:21.929175+00:00
7117e50c-df76-4bc2...	MSPUB-SAT-2026-001	training_completed	heart-container.html	participant00	2026-06-06T22:40:54.333222+00:00
f1819c16-b705-44dc...	MSPUB-SAT-...	email_link_clicked	login.html	participant01	2026-06-06T17:26:17.384412+00:00
9095540c-9140-4545...	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant01	2026-06-06T17:26:23.994848+00:00
341dc43b-f553-465b...	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant05	2026-06-06T17:16:17.677216+00:00
5a5e0f15-150b-4aaa...	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant05	2026-06-06T17:16:13.557032+00:00
c5ba0072-7c6c-44f0...	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant09	2026-06-06T17:17:49.231233+00:00
744b4303-ad36-45b...	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant09	2026-06-06T17:17:42.643074+00:00
1d54dbb3-c9e-4746...	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant13	2026-06-06T17:19:26.629064+00:00
191a58b5-0016-442...	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant13	2026-06-06T17:19:31.133388+00:00
c4b9952e-d17c-47bb...	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant17	2026-06-06T17:20:57.043690+00:00
3a3ed1be-ce58-4683...	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant17	2026-06-06T17:21:01.079920+00:00
ff9b617-adb0-4daa...	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant18	2026-06-06T17:22:29.155774+00:00
c81f37a-c5c-48b0...	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant18	2026-06-06T17:22:25.333131+00:00
770be884-1a89-4c08...	MSPUB-SAT-2026-001	training_page_view	user-failed-phishing-test.html	participant37	2026-06-06T17:34:42.457637+00:00
8889fade-e03-4fd2...	MSPUB-SAT-2026-001	login_attempt	login.html	participant37	2026-06-06T17:34:41.976726+00:00
f23e238b-5f66-4512...	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant37	2026-06-06T17:33:31.204797+00:00
a7b3b30-63b6-405...	MSPUB-SAT-2026-001	training_completed	heart-container.html	participant37	2026-06-06T17:34:58.948942+00:00
5c265746-2623-4480...	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant37	2026-06-06T17:33:32.160904+00:00

Testing SAT work flow

LTechs email filtering system validating links seconds after sending

Friend opening link, enters creds, completes training

Figure SISAT-5: DynamoDB logging results table

Participant Tracker											
Participant ID	Person	Email address	Email Name	Custom Email Button Link ID	Email Sent	Email Sent Date	Initial Link Clicked	Login Attempt Clicked	Training Viewed	Training Completed	Notes
participant001	Linda E.		Linda	https://masteryworkpub-net.jaycemangum.com/login.html?participant=001	✓	6/9/26 10:25 AM	✓	✓	✓	✓	College's email filter system appears to have tried verifying the email link. Participant has either ignored the email, or didn't click the link.
participant005	Mihail B.		Mihail	https://masteryworkpub-net.jaycemangum.com/login.html?participant=005	✓	6/9/26 10:15 AM	✓	✓	✓	✓	College's email filter system appears to have tried verifying the email link. Participant has either ignored the email, or didn't click the link.
participant009	Ifu I.		Ifu	https://masteryworkpub-net.jaycemangum.com/login.html?participant=009	✓	6/9/26 10:17 AM	✓	✓	✓	✓	College's email filter system appears to have tried verifying the email link. Participant has either ignored the email, or didn't click the link.
participant013	Makain M.		Makain	https://masteryworkpub-net.jaycemangum.com/login.html?participant=013	✓	6/9/26 10:18 AM	✓	✓	✓	✓	College's email filter system appears to have tried verifying the email link. Participant has either ignored the email, or didn't click the link.
participant017	Memadou S.		Memadou	https://masteryworkpub-net.jaycemangum.com/login.html?participant=017	✓	6/9/26 10:20 AM	✓	✓	✓	✓	College's email filter system appears to have tried verifying the email link. Participant has either ignored the email, or didn't click the link.
participant018	Caden T.		Caden	https://masteryworkpub-net.jaycemangum.com/login.html?participant=018	✓	6/9/26 10:21 AM	✓	✓	✓	✓	College's email filter system appears to have tried verifying the email link. Participant has either ignored the email, or didn't click the link.
participant027	Taylor M.		Taylor	https://masteryworkpub-net.jaycemangum.com/login.html?participant=027	✓	6/9/26 10:09 AM	✓	✓	✓	✓	Opened link below, then filed out the form, and proceeded through the exercise.
participant038	Kali T.		Kali	https://masteryworkpub-net.jaycemangum.com/login.html?participant=038	✓	6/9/26 9:30 AM	✓	✓	✓	✓	Ignored email or didn't click the link in the email.
participant039	David M.		David	https://masteryworkpub-net.jaycemangum.com/login.html?participant=039	✓	6/9/26 9:26 AM	✓	✓	✓	✓	Opened link below, then filed out the form, and proceeded through the exercise.
participant040	Anna J.		Anna	https://masteryworkpub-net.jaycemangum.com/login.html?participant=040	✓	6/9/26 9:56 AM	✓	✓	✓	✓	
participant041	Mason H.		Mason	https://masteryworkpub-net.jaycemangum.com/login.html?participant=041	✓	6/9/26 9:38 AM	✓	✓	✓	✓	

Statistics		
Metric	Count	Percentage
Emails Sent	11	100.00%
Links Clicked	5	41.67%
Login Attempts	5	41.67%
Training Viewed	5	41.67%
Training Completed	4	33.33%
Did Not Respond	6	50.00%

Figure SISAT-6: SAT Statistics Tracker Final Results, Campaign ended on 6/9/26

Incident Response Plan

In addition to preventive measures, a formal Incident Response Plan (IRP) for MSPUB was created, aligned to ISO/IEC 27001 principles and NIST incident handling guidance. The plan establishes the following procedures for:

- Incident preparation
- Detection and reporting
- Assessment and analysis
- Containment
- Eradication
- Recovery
- Post-incident review
- Continuous improvements

The MSPUB Security awareness results were also incorporated into the IRP as a practical example. The IRP provides a structured framework for responding to security incidents while minimizing business impact and supporting organization resilience.

Security Implementation Results

The security controls implemented throughout the MSPUB environment established a layered approach to cybersecurity that addressed infrastructure, applications, users, and operational procedures. By combining technical safeguards, centralized administration, security awareness training, and incident response

planning, MSPUB's security program has been designed to support the confidentiality, integrity, and availability of organization information assets.

During testing, with my LWTech student email, I noticed that the campus email filtering system flagged the campaign email as junk. After a couple of days, it appears the system automatically deleted the phishing email without my interaction in deleting the message. This behavior demonstrated the effectiveness of modern email security controls and highlights the importance of layered defenses that combine technical protections with user awareness training.

Troubleshooting Documentation

Troubleshooting and problem-solving are critical skills for systems administrators and cybersecurity professionals. As part of the Master Sword Publishing Capstone Project scenario, several mock troubleshooting scenarios were developed and resolved to demonstrate the ability to identify, analyze, and correct common enterprise technology issues. These troubleshooting exercises were designed to simulate real-world problems that may occur within business environments. Each scenario followed a structured troubleshooting methodology that included problem identification, analysis, remediation, testing, and validation. All scenarios can be found inside the "Master Sword Publishing's Information Technology Infrastructure Documentation" document on pages 165-197.

DNS Misconfiguration

The DNS Misconfiguration scenario simulated a failure in name resolution services within the MSPUB environment. The user was unable to properly locate

network resources due to incorrect DNS settings. The troubleshooting process involved:

- Identifying failed name resolution requests
- Reviewing networks configurations verifying DNS server settings
- Correcting configuration errors
- Testing connectivity and name resolution

Successful resolution restored communications between systems and demonstrated the importance of DNS services within enterprise networks.

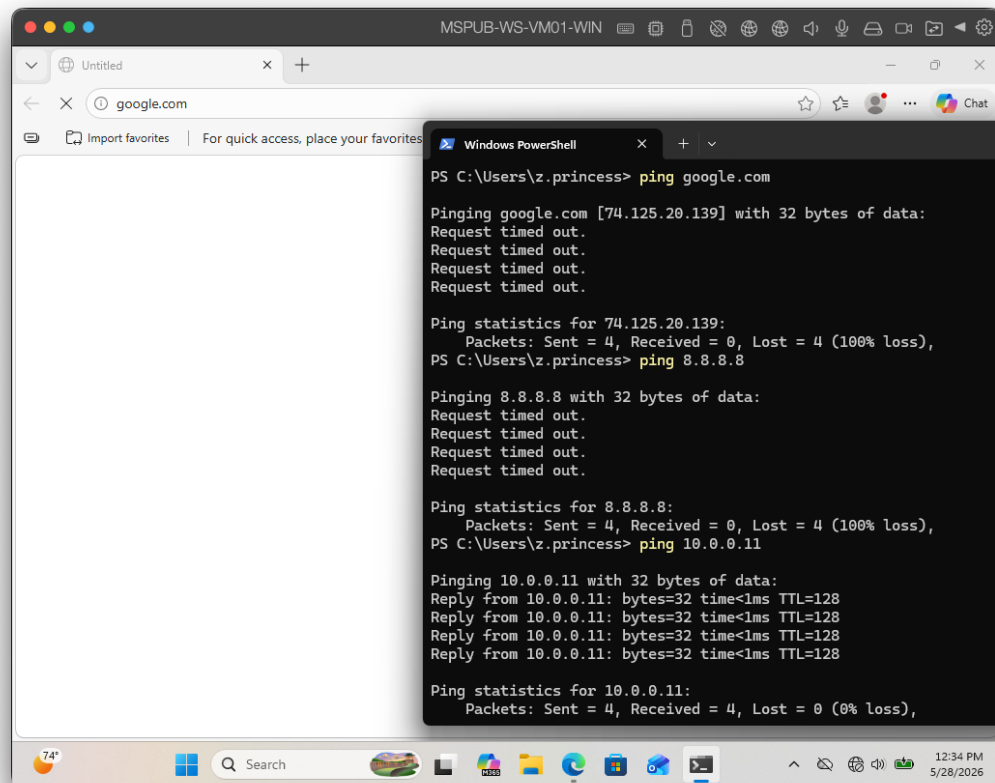


Figure TD-1: User DNS issue. User cannot load webpages, nor can they ping internal or external resources.

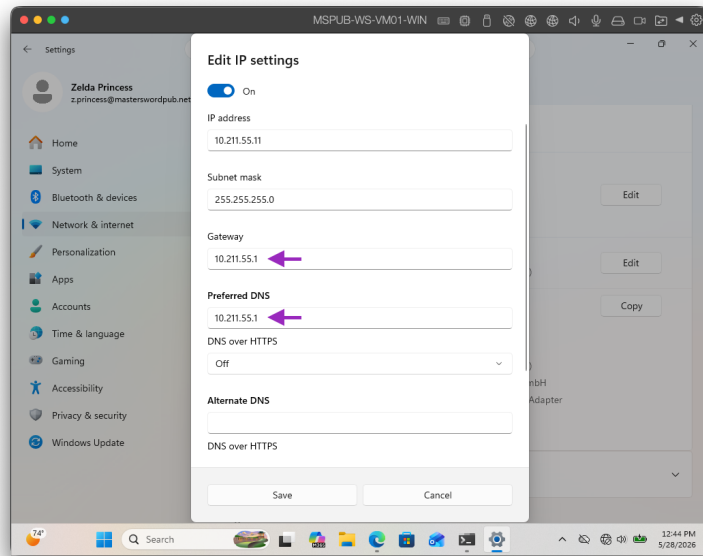


Figure TD-2: Verifying system DNS settings and fixing them.

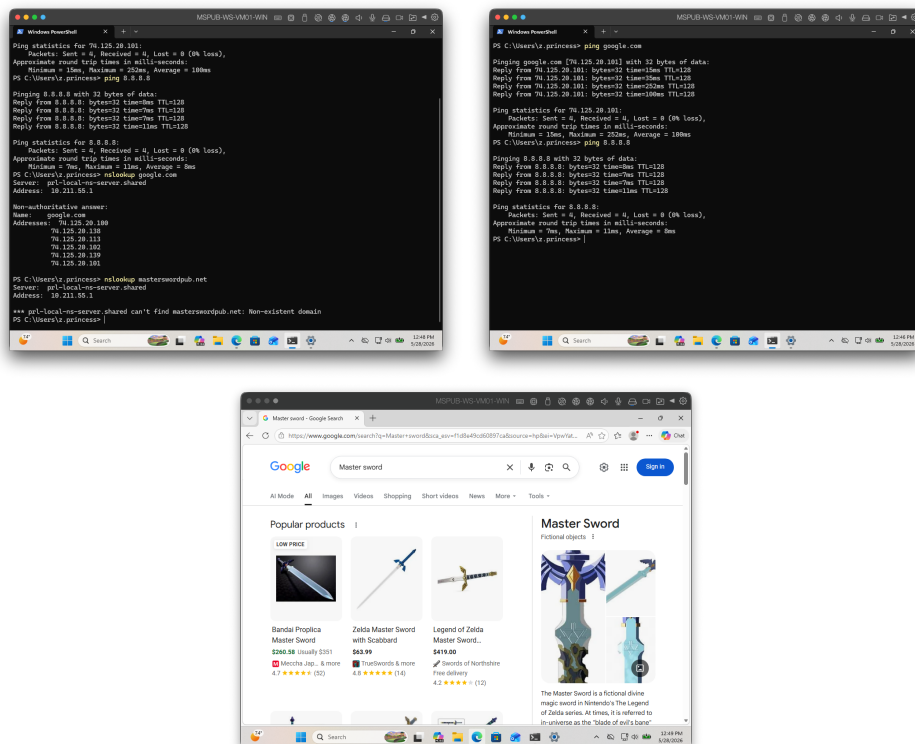


Figure TD-3: DNS issue resolved. Testing nslookup for domain services, pinging internal/external resources and web browsing.

User Account Lockout

The User Account Lockout scenario simulated a common support request involving a user who was unable to access their organization resources due to account restrains. The troubleshooting process involved:

- Reviewing account status within Active Directory
- Identifying the cause of the lockout (too many failed login attempts)
- Restoring account access
- Verifying successful authentication
- Confirming access to required resources

This exercise demonstrated account administration procedures and Active Directory management skills.

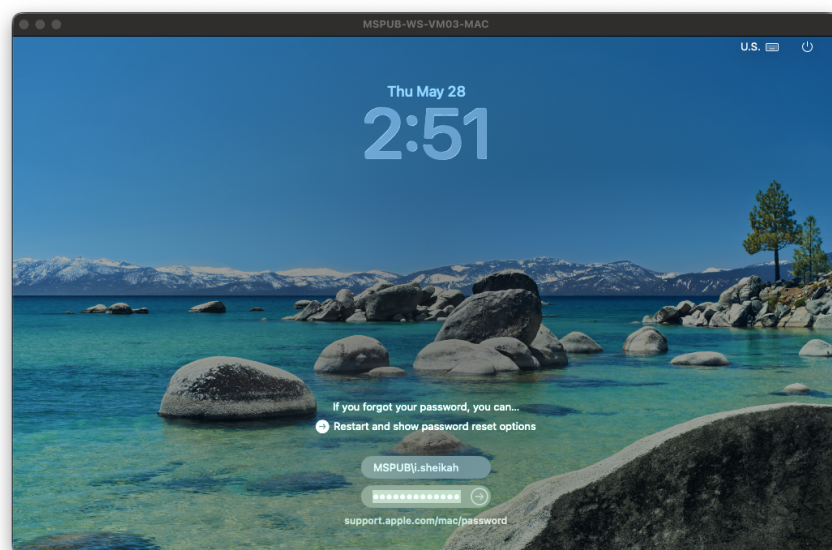


Figure TD-4: User Account Locked out. Cannot sign into domain account

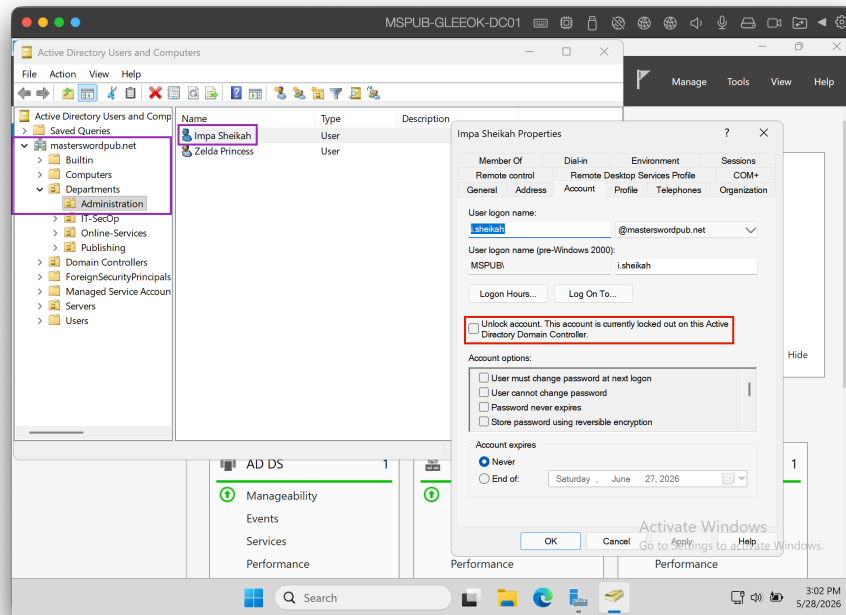


Figure TD-5: Check the user's account status in Active Directory, account is locked out.

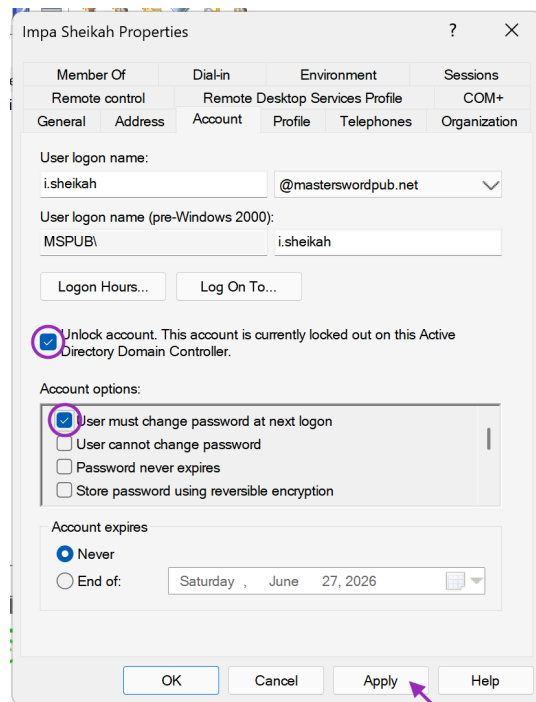


Figure TD-6 Unlocking User Account and requiring a password rest at next login.

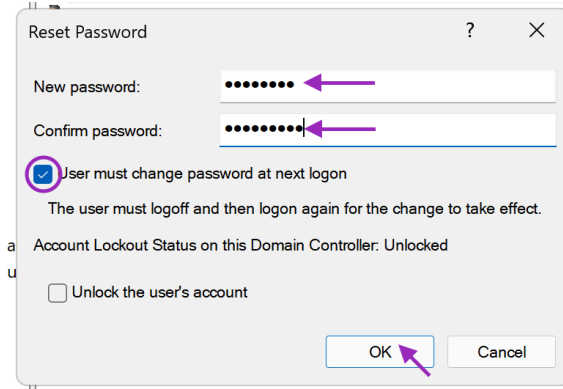


Figure TD-7: Resetting the users password to a temporary password to give them.

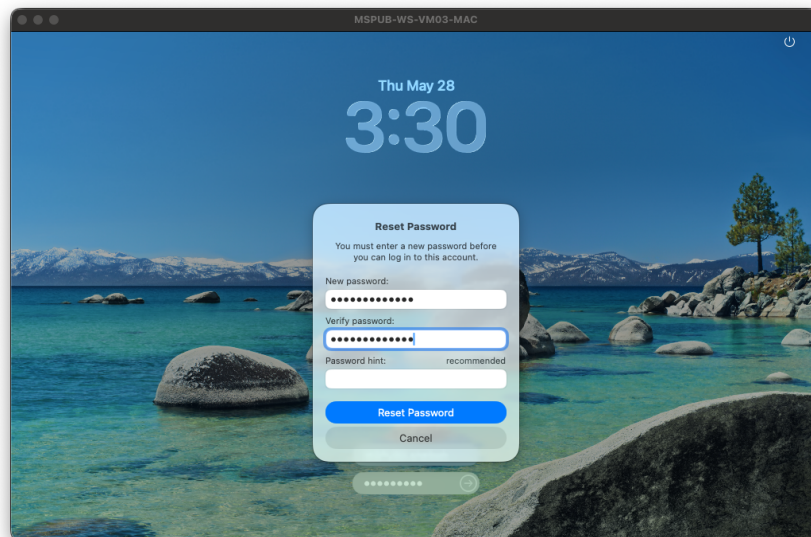


Figure TD-8: User successfully logged back in and the password reset prompt works.

Disk Space Management

The Disk Space scenario simulated a workstation experiencing limited available storage capacity. The issue affected normal system operation and required investigation to determine the cause of the storage utilization. The troubleshooting process included:

- Identifying storage utilization issues
- Locating unnecessary files and data

- Removing non-essential content
- Verifying available storage capacity
- Confirming normal system operation

This scenario demonstrated system maintenance procedures and resource management techniques.

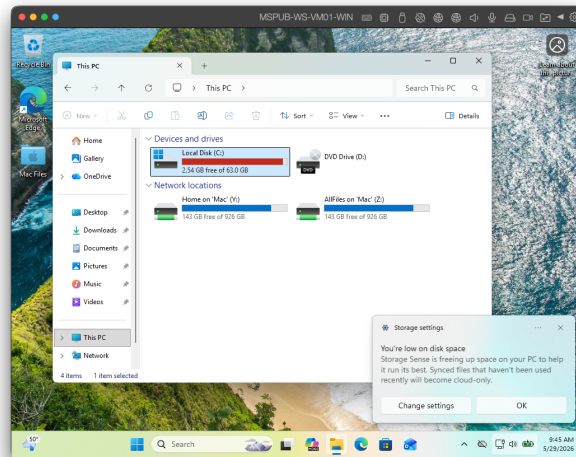


Figure TD-9: User reports low disk space one the system

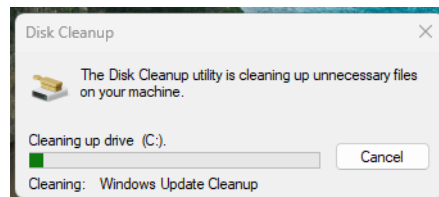


Figure TD-10: Running Disk Clean up to remove temporary files and caches

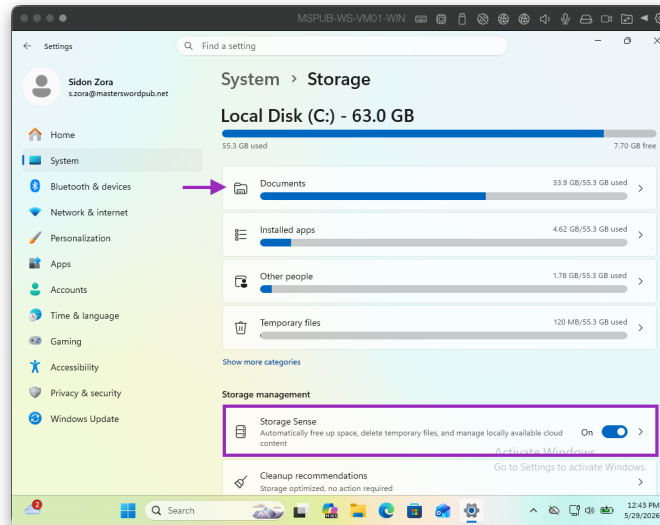


Figure TD-11: Finding the source of large files in Storage Settings

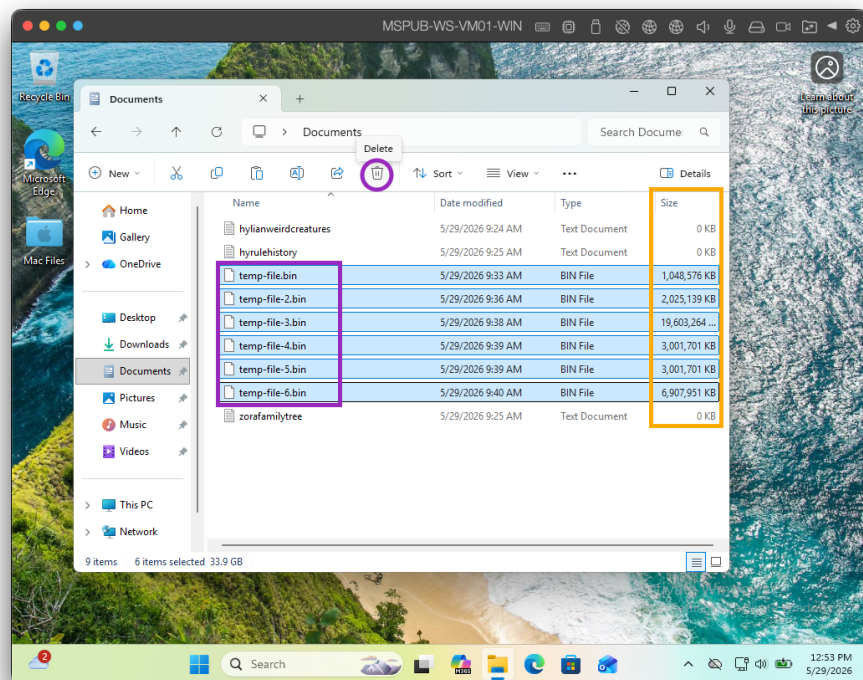


Figure TD-12: Delete the large files in the Documents folder.

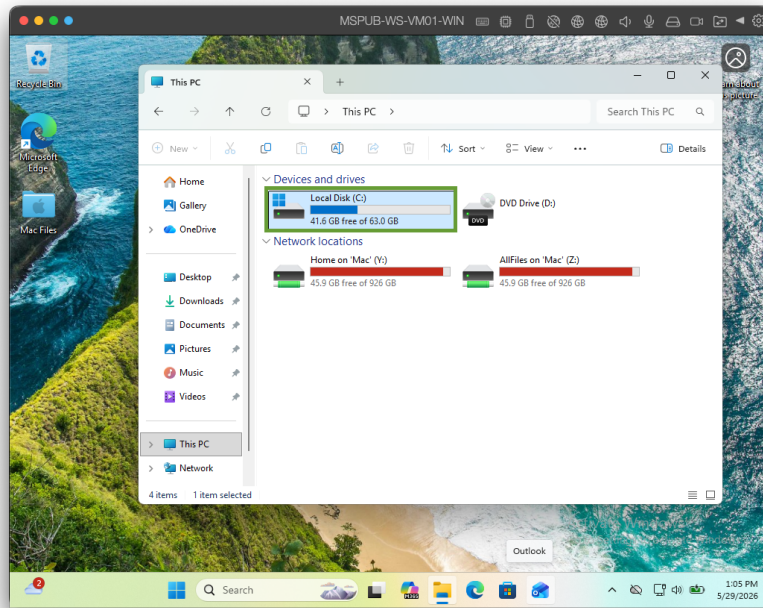


Figure TD-13: Checking Disk Space is now free in File Explorer

Network Connectivity Issue

The Network Connectivity scenario simulate a workstation experiencing communication problems within the MSPUB environment. The troubleshooting process included:

- Verifying network adapter configuration
- Testing IP connectivity
- Reviewing network settings
- Correcting configuration errors
- Validating communication with network resources

The resolution restored normal connectivity and reinforced the importance of systematic troubleshooting methodologies.

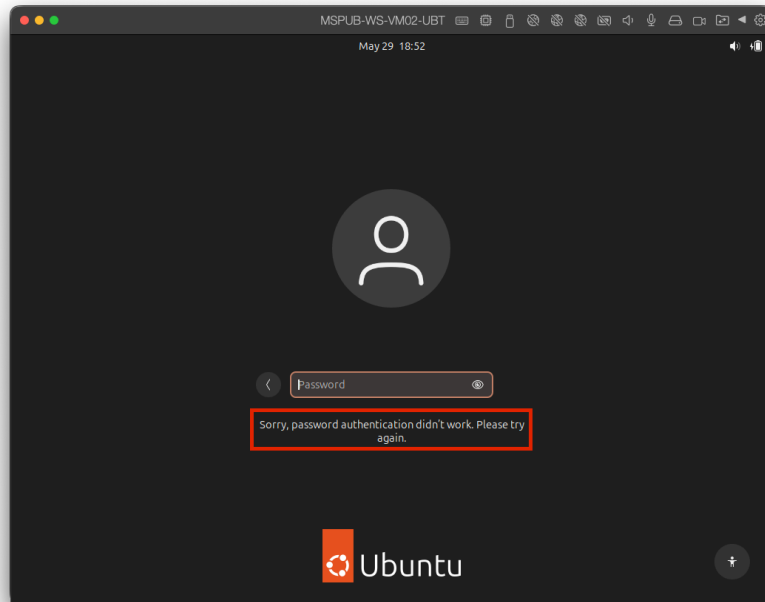


Figure TD-14: User is unable to access Active Directory services to login

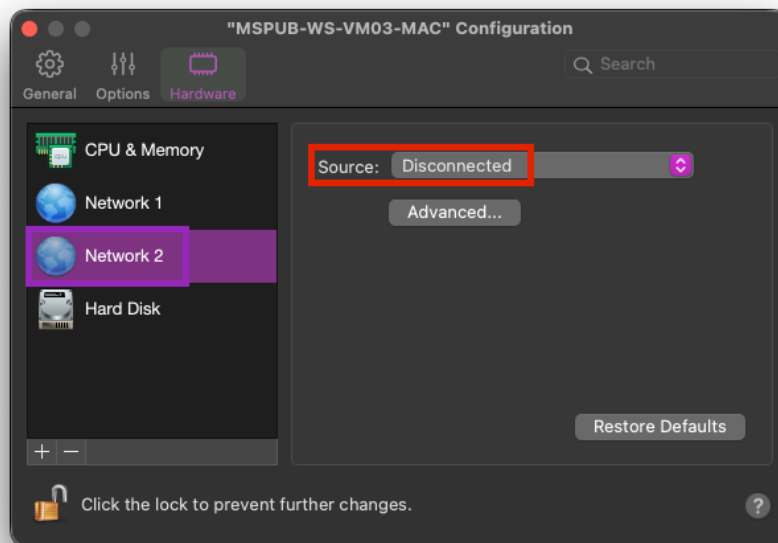


Figure TD-15: Checking the network adapter in the Hypervisor, which is disabled.

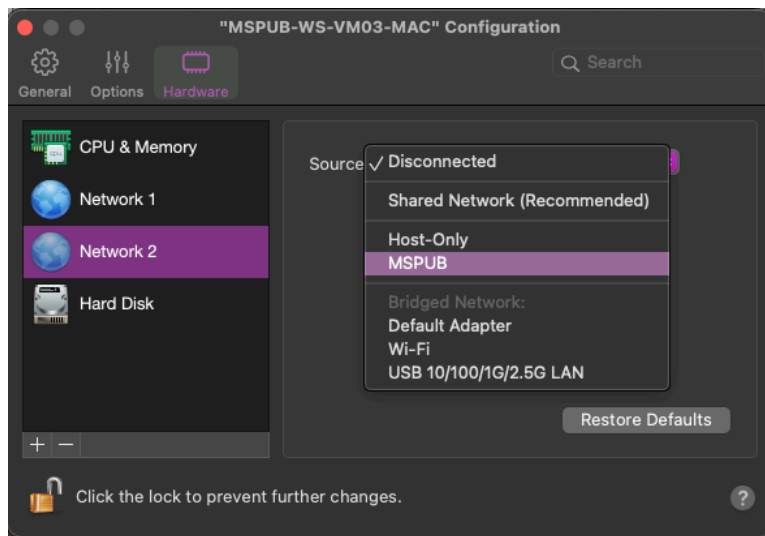


Figure. TD-16: Changing the network adapter back to the internal network state

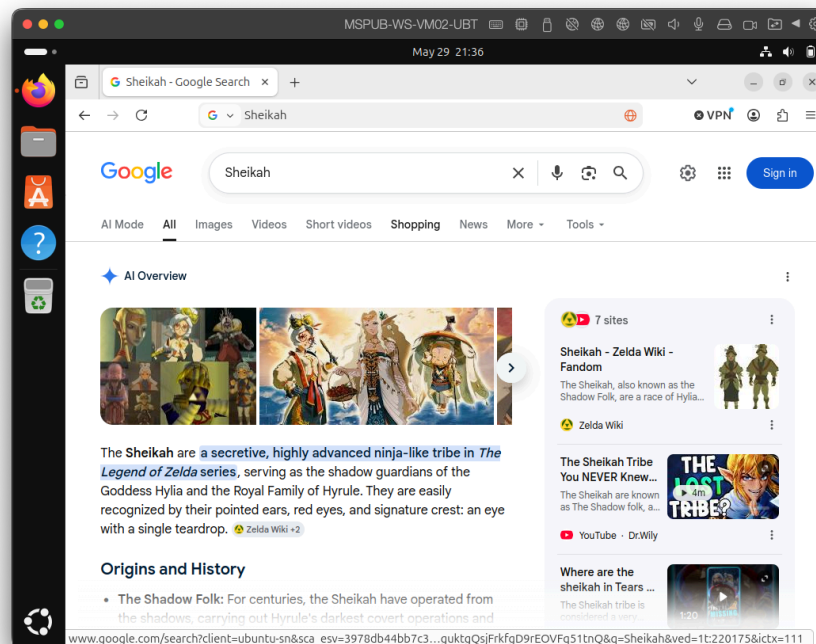


Figure TD-17: User was able to login into their domain account, and internet browsing has been restored.

Troubleshooting Methodology

Each troubleshooting scenario followed a structured process designed to minimize downtime and identify root causes efficiently. The methodology consisted of:

1. Identifying the problem
2. Establishing a theory of probable cause
3. Testing the theory
4. Implementing a solution
5. Verifying system functionality
6. Documenting finds and actions

This approach aligns with industry best practices and provides a repeatable framework for resolving technical issues.

Results

All troubleshooting scenarios were successfully resolved through systematic analysis and corrective action. These exercises demonstrated practical problem-solving skills across networking, operating systems, Active Directory administration, and system maintenance. The scenarios reinforced the importance of documentation, testing, and validation when addressing technology-related issues within enterprise environments.

Testing & Validation

Testing and validation activities were conducted throughout the implementation process to verify that systems, services, automation solutions, and security controls operated as intended. Testing was performed following major configuration changes

and project milestones to ensure functionality, reliability, and interoperability within the Master Sword Publishing (MSPUB) environment. The testing process focused on validating infrastructure services, user authentication, operating system integration, automation solutions, cloud services, and security awareness components.

Active Directory Validation

Active Directory Domain Services (AD DS) was tested to verify proper operation of authentication and directory services. Validation activities included:

- Domain controller functionality testing
- User authentication testing
- Group membership verification
- Organization Unit validation
- Active Directory replication testing
- Domain login testing

Testing confirmed that users could successfully authenticate using domain credentials and that directory information replicated correctly between domain controllers.

DNS Validation

DNS services were tested to ensure reliable name resolution throughout the MSPUB environment. DNS services were verified by:

- Forward lookup testing
- Hostname resolution testing
- Domain controller discovery testing
- Inter-system communication testing
- DNS replication verification

Successful testing confirmed that DNS services supported communication and resource discovery throughout the environment.

Workstation Integration Testing

Each workstation operating system was tested to verify communication and compatibility with enterprise services. Testing included:

- Windows 11 authentication testing
- Ubuntu Desktop connectivity testing
- macOS domain integration testing
- Kali Linux connectivity validation
- Resource access testing
- Network communication verification

Results confirmed successful deployment and operation of both web and file transfer services.

Web Server and SFTP Validation

The MSPUB-LYNEL-WEB01 server was tested to verify proper operation of deployed services. The following were completed to verify services:

- Nginx service verification
- Website accessibility testing
- SFTP connectivity testing
- File transfer validation
- Service startup verification

Testing confirmed successful deployment and operation of both web and file transfer services.

Automation Validation

Automation scripts were tested extensively prior to deployment.

Automated System Configuration Script validation:

- Service installation verification
- Website deployment validation
- Nginx configuration testing
- SFTP configuration validation
- Service functionality verification

User Management Script validation:

- User creation testing
- Group creation testing
- Group assignment testing
- Batch import validation
- Bulk deletion testing
- Logging verification

Results confirmed that both automation projects functioned as designed and successfully reduced administrative efforts.

Security Awareness Campaign Validation

The MSPUB Security Awareness Training (SAT) platform was tested to verify proper operation of the phishing simulation environment. Testing included:

- Phishing email delivery testing
- Website accessibility validation
- Login page functionality testing
- Training page verification
- Event logging validation
- Database recording verification

AWS services were monitored throughout testing to ensure proper communication between platform components.

AWS Cloud Service Validation

Cloud infrastructure components were validated individually and as an integrated solution. The following were tested:

- Amazon S3 website hosting verification
- CloudFront content delivery testing
- AWS Lambda execution testing
- DynamoDB record validation
- CloudWatch log monitoring

Testing confirmed successful event processing and participant activity tracking throughout the Security Awareness Training platform.

Project Objective Validation

Testing results confirmed that all major project objectives were successfully achieved, including:

- Enterprise infrastructure deployment
- Active Directory implementation
- DNS configuration
- Multi-platform workstation integration
- Linux service deployment
- Automation development
- Security awareness training implementation
- Incident réponse planning
- Technical documentation development

Results

Testing and validation activities confirmed that the MSPUB environment operated as intended and successfully met project requirements. The infrastructure demonstrated reliable functionality across server platforms, workstation systems, automation tools, cloud services, and security components. Validation results provided assurance that the environment was capable of supporting enterprise operations while meeting the objectives established for the capstone project.

Risk Analysis

Risk analysis is an essential component of cybersecurity and infrastructure planning. Organizations face a variety of threats that can impact the confidentiality, integrity, and availability of information systems. As part of the Master Sword Publishing (MSPUB) capstone project, several risks were identified and evaluated during the reign and implication process. The objective of the risk analysis was to identify potential threats, assess their impact on business operations, and implement appropriate controls to reduce organization risk.

Phishing & Social Engineering

Phishing attacks remain one of the most common methods used by attackers to obtain user credentials and gain unauthorized access to organization resources. Employees are frequently targeted through deceptive emails, fraudulent websites, and social engineering techniques. Potential impacts may include:

- Credential theft
- Data exposure
- Unauthorized account access
- Malware infections

- Business disruption

To address the risk, MSPUB implemented a Security Awareness Training (SAT) campaign designed to educate users about phishing attacks and improve security awareness.

Malware & Ransomware

Malware and ransomware attacks can disrupt business operations, damage systems, and result in the loss of critical information. Potential impacts include:

- Data loss
- System downtime
- Business interruption
- Financial losses
- Recovery costs

Mitigation measures implemented within the MSPUB environment include operating system security controls, software updates, firewall protections, and user awareness trainings.

Unauthorized Access

Unauthorized access can occur through compromised credentials, weak passwords, privilege abuse, or misconfigured systems. Potential impacts include:

- Exposure of sensitive information
- Unauthorized system changes
- Loss of data integrity
- Service disruptions

To reduce this risk, MSPUB implemented centralized authentication through Active Directory, account management controls, security groups, and role-based access management.

Human Error

Human error remains one of the leading causes of technology incidents.

Configuration mistakes, accidental deletions, improper system changes, and failure to follow procedures can negatively affect business operations. Potential impacts include:

- Service outages
- Security vulnerabilities
- Data loss
- Configuration inconsistencies

Documentation, automation, standardized procedures, and user training were implemented to reduce the likelihood of human error.

Infrastructure Failure

Hardware failures, software failures, and service interruptions may affect the availability of critical business resources. Potential impacts may include:

- Downtime
- Service interruptions
- Loss of productivity
- Data availability issues

To address this risk, MSPUB implemented redundancy where possible, including secondary domain controllers, cloud-based services, and documented recovery procedures.

Cloud Service Risks

Cloud-hosted services introduce additional risks including service outages, misconfigurations, and dependency on third-party providers. Possible impacts include:

- Service disruptions
- Delayed incident response
- Data accessibility issues

AWS services utilized within the Security Awareness Training platform were configured with monitoring and logging capabilities to support troubleshooting and service validation.

Risk Mitigation Summary Matrix

Risk	Potential Impact	Mitigation Strategy
Phishing	Credential Theft	Security Awareness Training
Malware	System Compromise	Security Controls and Updates
Unauthorized Access	Data Exposure	Active Directory and Access Controls
Human Error	Service Disruption	Documentation and Automation
Infrastructure Failure	Downtime	Redundancy and Recovery Procedures
Cloud Service Failure	Service Interruptions	Monitoring and Logging

Results

The MSPUB environment was designed using a defense-in-depth approach that incorporates technical controls, administrative controls, security awareness initiatives, and incident response planning. While no environment can completely eliminate risk, the implementation of layered security measures significantly reduces the likelihood and impact of common cybersecurity threats.

Performance Analysis

The purpose of the performance analysis was to evaluate the effectiveness, reliability, scalability, and overall success of the Master Sword Publishing (MSPUB) infrastructure. Throughout the project, systems, services, automation solutions, and security controls were tested and validated to ensure they met organization

requirements and project objectives. The analysis focused on infrastructure on infrastructure performance, automation effectiveness, security implementation, and user awareness initiatives.

Infrastructure Performance

The virtual enterprise environment successfully supported all required services throughout implementation and testing. Windows Server, Windows 11, Ubuntu Desktop, Ubuntu Server, macOS, and Kali Linux systems operated reliably within the virtual environment and maintained communication with one another. Core services like Active Directory Domain Services (AD DS), DNS, web hosting, and user authentication performed as expected throughout the project lifecycle. Performance observations included:

- Stable domain controller operation
- Consistent user authentication
- Reliable DNS name resolution
- Reliable web and file transfer
- Successful cross-platform services communication

The infrastructure provided a stable foundation for administration, testing, and security activities.

Active Directory Performance

Active Directory services successfully supported centralized authentication and directory management functions. Performance testing demonstrated:

- Successful domain joins
- Proper security group functionality
- Reliable user authentication
- Consistent directory replication

- Effective centralized account management

The implementation simplified administrative tasks while providing enterprise-level identity management capabilities.

Automation Effectiveness

Automation solutions significantly reduced the amount of manual administrative effort required within the MSPUB environment. The Automated System Configuration Script successfully deployed required web services and reduced configuration time by eliminating repetitive installation and setup tasks. The User Management Script improved efficiency by automating account creation, group assignments, batch imports, and logging activities. The observed benefits included:

- Reduced deployment time
- Improved consistency
- Reduced human error
- Simplified administrative processes
- Improved scalability for future growth

The automation projects successfully demonstrated the practical value of scripting within enterprise environments.

Security Performance

Security controls implemented throughout the MSPUB environment functioned as intended and supported organization security objectives. Performance indicators included:

- Successful authentication controls
- Proper account management
- Secure service configurations
- Operating system hardening measures
- Firewall protections

- Security awareness training implementation

These controls contributed to a layered security approach that improved the overall security posture of the environment.

Security Awareness Training Performance

The Security Awareness Training (SAT) campaign demonstrated the effectiveness of combining technical controls with user education. The AWS-based platform successfully:

- Delivered phishing simulation content
- Recorded participant interactions
- Tracked training progress
- Logged security awareness metrics
- Generated measurable campaign results

The campaign provided valuable insight into user behavior while reinforcing security awareness concepts through practical experience.

Scalability & Future Growth

The MSPUB design was developed with scalability in mind. Additional users, workstations, servers, and services can be added with minimal modifications to the existing infrastructure and minimal IT staff overhead. The advantages of scalability include:

- Centralized user management
- Expandable Active Directory structure
- Flexible network design
- Cloud-based service integration

- Automation-assisted administration

These characteristics support future growth of the organization while minimizing administrative complexity.

Overall Project Performance

The MSPUB environment successfully achieved the objectives established during the planning phase. Core infrastructure services, automation solutions, security controls, troubleshooting exercises, and cloud-based security awareness initiatives were implemented and validated successfully. The project demonstrated the ability to design, deploy, secure, manage, and document enterprise technology solutions while applying concepts learned throughout the Computer Security and Network Technology (CSNT) program.

Results

Performance analysis indicated that the MSPUB infrastructure operated reliably, supported organization requirements, mandated met project objectives. The combination of enterprise services, automation, security controls, and cloud-based technologies resulted in a functional and scalable environment capable of supporting modern business operations.

Lessons Learned

The Master Sword Publishing (MSPUB) capstone project gave me valuable experience across systems administration, networking, cybersecurity, cloud computing, automation, troubleshooting, and project documentation. While the project successfully achieved its objectives, some challenges were encountered throughout

the implementation process. These challenges provided opportunities to develop problem-solving skills and gain practical experience with enterprise technologies.

Technical Lessons Learned

One of the most significant lessons learned involved the importance of planning and validating infrastructure decisions before implementation. Early testing revealed the limitations when attempting to use VMware Fusion for portions of the virtual environment, particularly when working with Advanced RISC Machine (ARM) systems and macOS virtual machines. As a result, the project transitioned to Parallels Desktop, which provided improved compatibility and stability for these kinds of operating systems natively.

The project also reinforced the complexity of Active Directory deployments. Issues involving domain controller replication, DNS configuration, and domain joins across multiple operating system platforms demonstrated how closely integrated enterprise services depend on one another. Troubleshooting these issues provided a deeper understanding of the Microsoft enterprise infrastructure and the importance of validating foundational services before deploying dependent systems. Cross-platform administration introduced additional challenges. Integrating Windows, Ubuntu, macOS, and Kali Linux into a centralized environment required different management approaches for each operating system. This experience highlighted the importance of understanding platform-specific tools while maintaining consistent security and administrative practices.

Automation Lessons Learned

The automation projects demonstrated the value of scripting in reducing repetitive administrative tasks. Developing Python-based automation solutions provided practical experience with software development concepts while improving the overall operational efficiency. The System Configuration script demonstrated how infrastructure deployment tasks can be standardized and related consistently. The User Management Script reinforced the value of automation in user lifecycle management, account provisioning, and administrative reporting. These projects illustrated how automation can reduce human error while improving scalability and consistency within enterprise environments.

Cloud Computing Lessons Learned

The Security Awareness Training platform provided practical experience with cloud-based services and event-driven architectures. AWS services, including Amazon S3, CloudFront, Lambda, DynamoDB, and CloudWatch, were integrated to support the phishing awareness campaign. One of the most valuable lessons involved troubleshooting interactions between cloud services. CloudFront caching behavior, event logging, database integration, and content deployment required careful testing and validation to ensure accurate tracking of participant activity. The project reinforced the importance of understanding how cloud services interact and how monitoring tools can be used to diagnose operation issues.

Cybersecurity Lessons Learned

The Security Awareness Training campaign provided insight into both technical and human aspects of cybersecurity. While some participants interacted with the simulated phishing email, other email systems automatically filtered or moved the message before users could interact with it. This demonstrated the effectiveness of modern email security controls and reinforced the concept of layered security. User awareness training remains important, but technical controls such as spam filtering, email authentication, and threat detection systems also play a critical role in reducing an organization's risk. The development of the Incident Response Plan rather reinforced the importance of preparation, documentation, and structured response procedures when addressing security incidents.

Documentation Lessons Learned

The final lesson learned throughout this project was the value of maintaining detailed documentation. Screens, configuration notes, implementation procedures, troubleshooting records, and testing results were collected continuously throughout the project's lifecycle. Comprehensive documentation simplified troubleshooting efforts, reduced rework, and provided evidence of successful implementation. The documentation process also reinforced professional practices commonly used within enterprise IT environments.

Overall, the MSPUB capstone project demonstrated that successful enterprise technology deployments require more than technical knowledge alone. Planning, testing, troubleshooting, documentation, security considerations, and adaptability all contributed to the project's success. The challenges encountered throughout the

project provided opportunities to strengthen my technical skills while developing a deeper understanding of enterprise systems, cloud services, cybersecurity operations, and infrastructure management.

Future Improvements

Although the Master Sword Publishing (MSPUB) environment successfully achieved its project objectives, there are several opportunities for improvement and future enhancements. These improvements would further strengthen security, improve operation efficiency, and increase the organization's resilience:

- **Multi-Factor Authentication:** One potential improvement would be to implement MFA for user accounts and administrative access. This would provide an additional layer of protection against credential theft and unauthorized access.
- **Centralized Logging & Monitoring:** Implementing logging and monitoring in a centralized location would improve visibility across systems and services. Aggregating logs from servers, workstations, and cloud services would simplify troubleshooting and support incident detection activities.
- **Vulnerability Management:** Regular vulnerability assessments and automated vulnerability scanning could be implemented to identify security weaknesses and support proactive remediation efforts.
- **Automated Patch Management:** Future versions of the MSPUB environment could include automated operating system and application patching solutions to improve security and reduce administrative overhead.

- **Expanded Security Awareness Trainings:** Additional phishing simulations and security awareness campaigns could be conducted periodically to reinforce cybersecurity concepts and measure improvements in user awareness over time.
- **Backup and Disaster Recovery Enhancements:** While recovery procedures were addressed within the Incident Response Plan, future implementations could include additional backup automation, recovery testing, and disaster recovery planning activities to further improve the organization's resilience.
- **Expand Cloud Services:** Additional AWS services could be integrated into the environment to provide enhanced analytics, reporting, monitoring, and security capabilities. Future enhancements may include dashboard development, automated reporting, and advanced event analysis.

The future growth of the MSPUB environment was designed with scalability in mind and can support additional users, services, departments, and infrastructure components. Future expansion could include additional servers, cloud services, security tools, and enterprise applications as organizational requirements evolve. These improvements represent opportunities to further mature the MSPUB environments and strengthen security, automation, monitoring, and operational capabilities. While the current implementation successfully met project requirements, continuous improvement remains an important aspect of enterprise technology management.

Conclusion

The Master Sword Publishing (MSPUB) capstone project successfully demonstrated the knowledge and skills developed throughout my time in the Computer Security and Network Technology program at Lake Washington Institute of Technology.

This project combined networking, systems administration, cybersecurity, cloud computing, automation, troubleshooting, and documentation into a comprehensive enterprise technology solution. The project began with the design of an enterprise network architecture and the creation of a virtual business environment consisting of Windows, Linux, macOS, and security testing systems. Core services including Active Directory Domain Services, Dynamic Naming System, web hosting, and centralized authentication were successfully implemented and validated on each operating system. Automation solutions developed using Python improved administrative efficiency through automated system configuration and user management capabilities. The two scripting projects demonstrated the value of scripting within enterprise environments while reducing manual administrative effort. Security remained a primary focus throughout the project. Operating system hardening, centralized identity management, network security controls, security awareness training, and incident response planning were implemented to improve the overall security posture of the environment. The AWS-based Security Awareness Training platform provided practical experience with cloud technologies while supporting cybersecurity education objectives. The troubleshooting exercises further demonstrated the ability to identify, analyze, and resolve common enterprise technology issues using structured methodologies and industry best practices. Throughout the project, extensive documentation was maintained to record implementation procedures, testing results, troubleshooting activities, and security controls. This documentation provided evidence of successful completion while reinforcing professional IT documentation practices. In real-world IT, this would be documented in a knowledge base or ticketing system. For

this project, it is a dedicated document. The successful completion of the MSPUB capstone project demonstrates the ability to design, deploy, secure, manage, and document enterprise technology solutions. The project represents the culmination of knowledge acquired throughout the CSNT program and provides a strong foundation for future professional growth within the fields of information technology, systems administration, networking, cybersecurity, and cloud computing.

Appendices

Appendix A - Automation Python Scripts Source Code

Appendix A.1.a - System Configuration - Webserver

The script was written using Sublime Text with Python syntax highlighting:

[Download mspub-webserver-setup V3.py](#)

```
#!/usr/bin/env python3

import os
import subprocess
import logging
import shutil
from datetime import datetime

LOG_FILE = "/var/log/mspub_setup.log"
LOG_DIR = "/var/log"

def main():
    check_root()
    log_action("MSPUB Web Server Setup Script started.")
    system_updater()
    install_packages()
    configure_firewall()
    create_web_directory()
    create_sftp_directory()
    create_publishing_group()
    deploy_landing_page()
    configure_nginx_site()
    enable_services()
    verify_config()

    print("\nMSPUB Web Server Setup Complete!")
    log_action("MSPUB web server setup completed successfully.")

def log_action(message):
    """Write script activity to the log file."""
    timestamp = datetime.now().strftime("%Y-%m-%d %H:%M:%S")
    with open(LOG_FILE, "a") as log:
        log.write(f"[{timestamp}] {message}\n")
```

```

def run_command(command):
    """Run a Linux command and log success or failure."""
    try:
        result = subprocess.run(
            command,
            shell=True,
            check=True,
            text=True,
            capture_output=True
        )
        log_action(f"SUCCESS: {command}")
        return result.stdout

    except subprocess.CalledProcessError as error:
        log_action(f"ERROR {command}")
        log_action(error.stderr)
        print(f"Command failed: {command}")
        print(error.stderr)
        exit(1)

def check_root():
    """Verify the script is running with root privileges."""
    if os.geteuid() != 0:
        print("The MSPUB Webserver Setup script must be run with
sudo privileges.")
        exit(1)

    log_action("Root privilege check passed.")

def system_updater():
    """Update Ubuntu package repositories and installed packages."""
    print ("Updating system packages...")

    run_command("apt update")
    run_command("apt upgrade -y")

    log_action("System packages updated successfully.")

def install_packages():
    """Install required packages for the MSPUB web server."""
    print("Installing required packages...")

    packages = [
        "nginx",
        "openssh-server",

```



```

        "ufw"
    ]

    package_list = " ".join(packages)
    run_command(f"apt install {package_list} -y")

    log_action("Required packages installed successfully.")

def configure_firewall():
    """Configuring UFW firewall rules."""
    print("Configuring firewall rules...")

    run_command("ufw allow OpenSSH")
    run_command("ufw allow 'Nginx Full'")
    run_command("ufw --force enable")

    log_action("Firewall configured successfully.")

def create_web_directory():
    """Create the MSPUB web directory and set permissions."""
    print("Creating web directory...")

    web_dir = "/var/www/masterswordpub.net"

    run_command(f"mkdir -p {web_dir}")
    run_command(f"chown -R www-data:www-data {web_dir}")
    run_command(f"chmod -R 755 {web_dir}")

    log_action("Web directory created and permissions configured.")

def create_sftp_directory():
    """Create SFTP directory structure for the Publishing
    department."""
    print("Creating SFTP directory...")

    sftp_dir = "/srv/sftp/publishing"

    run_command(f"mkdir -p {sftp_dir}")
    run_command(f"chmod 755 /srv/sftp")
    run_command(f"chmod 755 {sftp_dir}")

    log_action("SFTP directory structure created successfully.")

def create_publishing_group():
    """Create the Publishing group for SFTP access."""
    print("Creating publishing group...")

```

```

group_name = "publishing"

group_exists = subprocess.run(
    f"getent group {group_name}",
    shell=True,
    text=True,
    capture_output=True
)

if group_exists.returncode == 0:
    log_action("Publishing group already exists.")
    print("Publishing group already exists.")
else:
    run_command(f"groupadd {group_name}")
    log_action("Publishing group created successfully.")

def deploy_landing_page():
    """Deploy a basic MSPUB internal website landing page."""
    print("Deploying MSPUB landing page...")

    source_file = "mspub_index.html"
    destination_file = "/var/www/masterswordpub.net/index.html"

    if not os.path.exists(source_file):
        log_action(f"ERROR: {source_file} not found.")
        print(f"Error: {source_file} not found. Please upload
{source_file} to server.")
        exit(1)

    shutil.copy(source_file, destination_file)

    run_command(f'chown www-data:www-data {destination_file}')
    run_command(f"chmod 644 {destination_file}")

    log_action("MSPUB website deployed successfully.")

def configure_nginx_site():
    """Configure Nginx to server the MSPUB website."""
    print("Configuring Nginx site...")

    site_config = "/etc/nginx/sites-available/masterswordpub.net"
    enabled_site = "/etc/nginx/sites-enabled/masterswordpub.net"
    default_site = "/etc/nginx/sites-enabled/default"

    nginx_config = """server {

```

```

listen 80;
listen [::]:80;

server_name masterswordpub.net www.masterswordpub.net;

root /var/www/masterswordpub.net;
index index.html;

location / {
    try_files $uri $uri/ =404;
}
}
"""

with open(site_config, "w") as file:
    file.write(nginx_config)

if os.path.exists(default_site):
    run_command(f"rm {default_site}")

if not os.path.exists(enabled_site):
    run_command(f"ln -s {site_config} {enabled_site}")

run_command("nginx -t")
run_command("systemctl reload nginx")

log_action("Nginx site configured successfully.")

def enable_services():
    """Enable and start required services."""
    print("Enabling and starting services...")

    services = [
        "nginx",
        "ssh"
    ]

    for service in services:
        run_command(f"systemctl enable {service}")
        run_command(f"systemctl restart {service}")

    log_action("Required services enabled and started successfully.")

def verify_config():
    """Verify web server configuration."""
    print("Verifying configuration...")

```

```

checks = [
    ("Nginx Service", "systemctl is-active nginx"),
    ("SSH Service", "systemctl is-active ssh"),
    ("Firewall Status", "ufw status"),
    ("Website Directory", "test -d /var/www/
masterswordpub.net"),
    ("SFTP Directory", "test -d /srv/sftp/publishing"),
    ("Publishing Group", "getent group publishing")
]

for check_name, command in checks:
    try:
        result = subprocess.run(
            command,
            shell=True,
            check=True,
            text=True,
            capture_output=True
        )

        log_action(f"VERIFIED: {check_name}")
        print(f"[PASS] {check_name}")

    except subprocess.CalledProcessError:
        log_action(f"FAILED: {check_name}")
        print(f"[FAIL] {check_name}")

log_action("Configuration verification completed.")

if __name__ == "__main__":
    main()

```

Appendix A.1.b - MSPUB Webpage

This webpage was originally developed for a project in CSNT257: Cloud Computing for a static website. It was adapted for this capstone project:

[Download mspub_index.html](#)

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-
scale=1.0" />

```

```

<title>MSPUB | CSNT 253: Capstone Project</title>

<style>
  * {
    box-sizing: border-box;
  }
  body {
    margin: 0;
    font-family: Arial, Helvetica, sans-serif;
    background:
      radial-gradient(circle at top, rgba(90, 220, 220,
0.25), transparent 35%),
      linear-gradient(135deg, #061014, #102b2c 45%,
#071a1f);
    color: #f5f0d6;
  }
  header {
    padding: 24px 8%;
    border-bottom: 1px solid rgba(212, 175, 55, 0.35);
    background: rgba(5, 20, 25, 0.85);
  }
  nav {
    display: flex;
    justify-content: space-between;
    align-items: center;
  }
  .logo {
    font-weight: bold;
    color: #d4af37;
    font-size: 1.2rem;
    letter-spacing: 1px;
  }
  .nav-link {
    font-size: 0.9rem;
    color: #9ee7e5;
  }
  .hero {
    padding: 110px 10% 90px;
    text-align: center;
  }
  .section {
    padding: 70px 10%;
  }
  .section h2 {
    color: #d4af37;
    margin-bottom: 28px;
  }

```

```

        font-size: 2rem;
        text-shadow: 0 0 10px rgba(0,0,0,0.6);
    }
    .sword {
        font-size: 4rem;
        text-shadow: 0 0 20px #5ce1e6;
    }
    h1 {
        font-size: 3rem;
        color: #d4af37;
        margin-bottom: 12px;
    }
    .subtitle {
        max-width: 800px;
        margin: 0 auto;
        font-size: 1.2rem;
        line-height: 1.7;
        color: #eef8f8;
        text-shadow: 0 0 8px rgba(0, 0, 0, 0.8);
    }
    .badge {
        display: inline-block;
        margin-top: 28px;
        padding: 12px 24px;
        border-radius: 999px;
        background: #d4af37;
        color: #061014;
        font-weight: bold;
    }
    .cards {
        display: grid;
        grid-template-columns: repeat(auto-fit, minmax(220px,
1fr));

        gap: 20px;
        margin-top: 24px;
    }
    .card {
        background: rgba(255, 255, 255, 0.10);
        border: 1px solid rgba(212, 175, 55, 0.35);
        border-radius: 18px;
        padding: 28px;
        box-shadow: 0 14px 35px rgba(0, 0, 0, 0.35);
        backdrop-filter: blur(4px);
        transition: transform 0.2s ease;
    }
    .card:hover {

```

```

        transform: translateY(-4px);
    }
    .card h3 {
        color: #9ee7e5;
        margin-top: 0;
    }
    .card p {
        line-height: 1.6;
        color: #e5e7eb;
    }
    footer {
        text-align: center;
        padding: 24px;
        color: #b9c7c7;
        border-top: 1px solid rgba(212, 175, 55, 0.25);
    }
</style>
</head>

<body>
    <header>
        <nav>
            <div class="logo">🗡 Master Sword Publishing</div>
            <div class="nav-link">Capstone Project</div>
        </nav>
    </header>

    <main>
        <section class="hero">
            <div class="sword">🗡</div>
            <h1>Hyrule's #1 Publishing Company</h1>
            <p class="subtitle">
                This internal website was created by Jayce Mangum
                for CSNT253: Capstone Project. This was created to demonstrate
                deploying a Python script to configure a Ubuntu Web Server running
                Nginx.

                <div class="badge">Currently hosted on: MSPUB-LYNEL-
                WEB01 running Ubuntu Server 26.04 with Nginx.</div>
            </p>
        </section>

        <section class="section">
            <h2>Capstone Project Components</h2>

            <div class="cards">

```

```

<div class="card">
  <h3>A+ Software Essentials & A+ Hardware</h3>
  <p>
    <b>Virtual Machine Home Lab</b>
    <ul>
      <li>Windows Server 2025: Domain
        Controller, Domain Manager</li>
      <li>Ubuntu Server 26.04: Web/App
        Server</li>
      <li>Windows 11: Domain
        Workstation</li>
      <li>Ubuntu Desktop 26.04: Domain
        Workstation</li>
      <li>macOS 26 Tahoe: Domain
        Workstation</li>
      <li>Kali Linux 2026:
        Cybersecurirty Workstation</li>
    </ul>
  </p>
  <p>
    <b>Troubleshooting Scenarios</b>
    <ul>
      <li>Routing & DNS
        Misconfigurtion</li>
      <li>User Account Locked Out</li>
      <li>Low Disk Space Issue</li>
      <li>Virtual Machine Network
        Issue</li>
    </ul>
  </p>
</div>

<div class="card">
  <h3>Shells & Scripts</h3>
  <p>
    <b>Automated Sysem Configuration
      Script</b>
    <ul>
      <li>Update package
        repositories</li>
      <li>Install Nginx</li>
      <li>Install and enable OpenSSH</li>
      <li>Configure SFTP support</li>
    </ul>
  </p>
</div>

```



```

rules</li>
traffic</li>
directory</li>
directory</li>

landing page</li>
permissions</li>
services</li>
file</li>
and validation</li>
</ul>
</p>
<p>
<b>User Management Script</b>
<ul>
<li>Help IT Administrators
<li>Create password policies</li>
<li>Assign groups</li>
<li>Create/delete home
</ul>
</p>
</div>

<div class="card">
<h3>Network Fundamentals & Administration</h3>
<p>
<b>Networking Components</b>
<ul>
<li>Configure a Ubiquiti Router
<li>Configure Firewall and
<li>Implement Quality of Service

```

```

        </ul>
    </p>
    <p>
        <b>Wireless Access Points</b>
        <ul>
            <li>Configure WAPs for seamless
connectivity</li>
            <li>Implement security protocols
WPA3 Enterprise</li>
        </ul>
    </div>
    <div class="card">
        <h3>Network Security & Encryption</h3>
        <p>
            Task 1: TBD
        </p>
        <p>
            Task 2: TBD
        </p>
    </div>
</div>
</section>
</main>

<footer>
    Created by Jayce Mangum, Spring Quarter 2026, at <a
href="https://www.lwtech.edu/">LWTech</a>.
</footer>
</body>

</html>

```

Appendix A.2 - MSPUB User Management Script

[Download mspub-usermgmt_v2.py](#)

[Download example batch_usr_import.csv](#)

[Download example bulk_usr_delete.csv](#)

```
#!/usr/bin/env python3

import os
import subprocess
import csv
from datetime import datetime

LOG_FILE = "/var/log/mspub_user_mgmt.log"
DEFAULT_PASSWORD = "Hyrul3Warr10r$"
IMPORT_CSV_FILE = "/opt/mspub-usrmgmt/batch-usr-import.csv"
REMOVE_CSV_FILE = "/opt/mspub-usrmgmt/bulk-usr-delete.csv"

def check_root():
    """Verify the script is running with root privileges."""
    if os.geteuid() != 0:
        print("The MSPUB User Management Script must be run with sudo
privileges.")
        exit(1)

    log_action("Root privilege check passed.")

def log_action(message):
    """Write script activity to the log file."""
    timestamp = datetime.now().strftime("%Y-%m-%d %H:%M:%S")

    with open(LOG_FILE, "a") as log:
        log.write(f"[{timestamp}] {message}\n")

def run_command(command):
    """Run a Linux command and log success or failure."""
    try:
        result = subprocess.run(
            command,
            shell=True,
            check=True,
            text=True,
            capture_output=True
        )
```

```

        log_action(f"SUCCESS: {command}")
        return result.stdout

except subprocess.CalledProcessError as error:
    log_action(f"ERROR: {command}")
    log_action(error.stderr)
    print(f"Command failed: {command}")
    print(error.stderr)
    return None

# Group Creation
def install_default_groups():
    """Create the default MSPUB department groups."""
    print("\nInstalling MSPUB department groups...")

    groups = [
        "administration",
        "publishing",
        "it-cyber",
        "online-services"
    ]

    for group in groups:
        group_exists = subprocess.run(
            f"getent group {group}",
            shell=True,
            stdout=subprocess.DEVNULL,
            stderr=subprocess.DEVNULL
        )

        if group_exists.returncode == 0:
            print(f"Group already exists: {group}")
            log_action(f"Group already exists: {group}")
        else:
            run_command(f"groupadd {group}")
            print(f"Created group: {group}")
            log_action(f"Created group: {group}")

    print("MSPUB department groups installation complete.")

def create_custom_group():
    """Create a custom Linux group."""
    group_name = input("\nEnter custom group name: ").strip().lower()

```

```

if not group_name:
    print("Group name cannot be blank.")
    return

group_exists = subprocess.run(
    f"getent group {group_name}",
    shell=True,
    stdout=subprocess.DEVNULL,
    stderr=subprocess.DEVNULL
)

if group_exists.returncode == 0:
    print(f"Group already exists: {group_name}")
    log_action(f"Group already exists: {group_name}")
else:
    run_command(f"groupadd {group_name}")
    print(f"Created custom group: {group_name}")
    log_action(f"Created custom group: {group_name}")

# User Creation
def create_user():
    """Create a new Linux user account and optionally assign group."""
    first_name = input("\nEnter employee first name initial:
").strip().lower()
    last_name = input("Enter employee full last name:
").strip().lower()
    group_name = input("Enter MSPUB group name for this user:
").strip().lower()

    if not first_name or not last_name:
        print("First name and last name cannot be blank.")
        return

    username = f"{first_name}.{last_name}"

    user_exists = subprocess.run(
        f"id {username}",
        shell=True,
        stdout=subprocess.DEVNULL,
        stderr=subprocess.DEVNULL
    )

    if user_exists.returncode == 0:
        print(f"User already exists: {username}")

```

```

    log_action(f"User already exists: {username}")
    return

if group_name:
    group_exists = subprocess.run(
        f"getent group {group_name}",
        shell=True,
        stdout=subprocess.DEVNULL,
        stderr=subprocess.DEVNULL
    )

    if group_exists.returncode != 0:
        print(f"Group does not exist: {group_name}")
        create_group = input("Create this group now? (y/n):
").strip().lower()

        if create_group == "y":
            run_command(f"groupadd {group_name}")
            log_action(f"Created group during user creation:
{group_name}")
        else:
            print("User creation cancelled.")
            return

    run_command(f"useradd -m -s /bin/bash -G {group_name}
{username}")
    log_action(f"Created user {username} and assigned to group
{group_name}")
    else:
        run_command(f"useradd -m -s /bin/bash {username}")
        log_action(f"Created user {username} without group
assignment")

    password_process = subprocess.run(
        ["chpasswd"],
        input=f"{username}:{DEFAULT_PASSWORD}\n",
        text=True,
        capture_output=True
    )

    if password_process.returncode == 0:
        log_action(f"MSPUB default password assigned to {username}")
    else:
        log_action(f"Password assignment failed for {username}:
{password_process.stderr}")
        print(f"Password assignment failed for {username}.")

```

```

        print(password_process.stderr)

    run_command(f"chage -d 0 {username}")
    log_action(f"Forced password change at next login for {username}")

    print(f"User created successfully: {username}")
    print(f"Temporary password: {DEFAULT_PASSWORD}")

def delete_user():
    """Delete an existing Linux user account."""
    username = input("\nEnter username to delete: ").strip().lower()

    if not username:
        print("Username cannot be blank.")
        return

    user_exists = subprocess.run(
        f"id {username}",
        shell=True,
        stdout=subprocess.DEVNULL,
        stderr=subprocess.DEVNULL
    )

    if user_exists.returncode != 0:
        print(f"User does not exist: {username}")
        log_action(f"Delete failed. User does not exist: {username}")
        return

    confirm = input(f"Are you sure you want to delete {username}? (y/n): ").strip().lower()

    if confirm == "y":
        run_command(f"userdel -r {username}")
        print(f"Deleted user: {username}")
        log_action(f"Deleted user: {username}")
    else:
        print("User deletion cancelled.")
        log_action(f"User deletion cancelled for {username}")

def prepare_batch_csv_file():
    """Create the batch import CSV file if it does not already exist"""

    csv_dir = "/opt/mspub-usrmgmt/"

```

```

if not os.path.exists(csv_dir):
    run_command(f"mkdir -p {csv_dir}")

if not os.path.exists(IMPORT_CSV_FILE):
    sample_content = """first_name,last_name,group_name
l,hero,publishing
z,princess,administration
s,zora,publishing
"""

    with open(IMPORT_CSV_FILE, "w") as file:
        file.write(sample_content)

    run_command(f"chmod 664 {IMPORT_CSV_FILE}")
    log_action(f"Created sample batch import CSV file:
{IMPORT_CSV_FILE}")

def batch_user_import():
    """Create multiple users from a static CSV file."""

    prepare_batch_csv_file()

    print("\nBulk User Import")
    print("-" * 40)
    print(f"CSV file location: {IMPORT_CSV_FILE}")
    print("\nInstructions:")
    print("1. Open the CSV file in a text editor.")
    print("2. Add one user per line using this format: ")
    print("    first_name,last_name,group_name")
    print("    per the MSPUB naming convention for usernames")
    print("first_name = first initial dot last_name = full last name. E.g.")
    print("p.sheikah")
    print("3. Save the CSV file before continuing.")
    print("4. Existing users will be skipped.")
    print("5. Missing groups will be created automatically.")
    print("/nCSV file Example:")
    print("    r,sheikah,it-cyber")
    print("    l,hero,it-cyber")
    print("    z,princess,administration")

    confirm = input("\nReady to begin the import? ").strip().lower()

    if confirm != "y":
        print("Batch import cancelled.")
        log_action("Batch import cancelled by administrator.")

```



```

    return

with open(IMPORT_CSV_FILE, "r") as csv_file:
    reader = csv.DictReader(csv_file)

    for row in reader:
        first_name = row["first_name"].strip().lower()
        last_name = row["last_name"].strip().lower()
        group_name = row["group_name"].strip().lower()

        if not first_name or not last_name or not group_name:
            log_action("Batch import skipped incomplete row.")
            continue

        username = f"{first_name}.{last_name}"

        user_exists = subprocess.run(
            f"id {username}",
            shell=True,
            stdout=subprocess.DEVNULL,
            stderr=subprocess.DEVNULL
        )

        if user_exists.returncode == 0:
            print(f"User already exists, skipped: {username}")
            log_action(f"Batch import skipped existing user:
{username}")
            continue

        group_exists = subprocess.run(
            f"getent group {group_name}",
            shell=True,
            stdout=subprocess.DEVNULL,
            stderr=subprocess.DEVNULL
        )

        if group_exists.returncode != 0:
            run_command(f"groupadd {group_name}")
            log_action(f"Batch import created group:
{group_name}")

        run_command(f"useradd -m -s /bin/bash -G {group_name}
{username}")

        password_process = subprocess.run(
            ["chpasswd"],

```

```

        input=f"{username}:{DEFAULT_PASSWORD}\n",
        text=True,
        capture_output=True
    )

    if password_process.returncode == 0:
        log_action(f"Batch import assigned default password to
{username}")
    else:
        log_action(f"Batch import password assignment failed
for {username}: {password_process.stderr}")
        print(f"Password assignment failed for {username}")

    run_command(f"chage -d 0 {username}")

    print(f"Created user: {username}")
    log_action(f"Batch import created user {username} in group
{group_name}. Default password set. Change password next logon: TRUE")

    print("Batch user import completed.")
    log_action("Batch user import completed.")

# Assignment
def assign_user_group():
    """Assign an existing user to an existing group."""
    username = input("\nEnter username: ").strip().lower()
    group_name = input("Enter group name: ").strip().lower()

    if not username or not group_name:
        print("Username and group name cannot be blank.")
        return

    user_exists = subprocess.run(
        f"id {username}",
        shell=True,
        stdout=subprocess.DEVNULL,
        stderr=subprocess.DEVNULL
    )

    if user_exists.returncode != 0:
        print(f"User does not exist: {username}")
        log_action(f"Group assignment failed. User does not exist:
{username}")
        return

    group_exists = subprocess.run(

```

```

        f"getent group {group_name}",
        shell=True,
        stdout=subprocess.DEVNULL,
        stderr=subprocess.DEVNULL
    )

    if group_exists.returncode != 0:
        print(f"Group does not exist: {group_name}")
        log_action(f"Group assignment failed. Group does not exist:
{group_name}")
        return

    run_command(f"usermod -aG {group_name} {username}")

    print(f"Added {username} to {group_name}")
    log_action(f"Added {username} to group {group_name}")

def force_password_change():
    """Force a user to change their password at next login."""
    username = input("\nEnter username: ").strip().lower()

    if not username:
        print("Username cannot be blank.")
        return

    user_exists = subprocess.run(
        f"id {username}",
        shell=True,
        stdout=subprocess.DEVNULL,
        stderr=subprocess.DEVNULL
    )

    if user_exists.returncode != 0:
        print(f"User does not exist: {username}")
        log_action(f"Password reset failed. User does not exist:
{username}")
        return

    run_command(f"chage -d 0 {username}")

    print(f"Password change required for {username}")
    log_action(f"Forced password change for {username}")

```

Other Features

```

def list_users():
    """Display local users with home directories."""
    print("\nMSPUB User Accounts")
    print("-" * 40)

    try:
        with open("/etc/passwd", "r") as passwd_file:
            for line in passwd_file:
                fields = line.split(":")

                username = fields[0]
                uid = int(fields[2])
                home = fields[5]

                if uid >= 1000 and home.startswith("/home"):
                    print(username)

            log_action("Displayed user list.")

    except Exception as error:
        print(f"Error displaying users: {error}")
        log_action(f"Error displaying users: {error}")

def prepare_remove_csv_file():
    """Create the batch user removal CSV file if it does not already
    exist."""

    csv_dir = "/opt/mspub-usrmgmt/"

    if not os.path.exists(csv_dir):
        run_command(f"mkdir -p {csv_dir}")

    if not os.path.exists(REMOVE_CSV_FILE):
        sample_content = """username
z.princess
i.sheikah
p.sheikah
"""

        with open(REMOVE_CSV_FILE, "w") as file:
            file.write(sample_content)

        run_command(f"chmod 644 {REMOVE_CSV_FILE}")
        log_action(f"Created sample batch removal CSV file:
{REMOVE_CSV_FILE}")

def bulk_delete_users():

```

```

"""Delete multiple users from a static CSC file."""

prepare_remove_csv_file()

print("\nBulk User Deletion")
print("-" * 40)
print(f"CSV file location: {REMOVE_CSV_FILE}")
print("\nInstructions:")
print("1. Open the CSV file in a text editor.")
print("2. Add one user per line using this format:")
print("    username")
print("3. Save the CSV file before continuing.")
print("4. User listed in the file will be deleted with their home
directories.")
print("\nExample:")
print("    z.princess")
print("    i.sheikah")

confirm = input("\nAre you sure you want to delete all users
listed in this file? (y/n): ").strip().lower()

if confirm != "y":
    print("Bulk user deletion cancelled.")
    log_action("Bulk user deletion cancelled by administrator.")
    return

with open(REMOVE_CSV_FILE, "r") as csv_file:
    reader = csv.DictReader(csv_file)

    for row in reader:
        username = row["username"].strip().lower()

        if not username:
            log_action("Bulk delete dkipped blank username row.")
            continue

        user_exists = subprocess.run(
            f"id {username}",
            shell=True,
            stdout=subprocess.DEVNULL,
            stderr=subprocess.DEVNULL
        )

        if user_exists.returncode != 0:
            print(f"User does not exist, skipped: {username}")

```

```

        log_action(f"Bulk delete skipped missing user:
{username}")
        continue

        run_command(f"userdel -r {username}")
        log_action(f"Bulk delete removed user: {username}")

    print("Bulk user deletion completed.")
    print("Returning to main menu.")
    log_action("Bulk user deletion completed.")

# Menu
def display_menu():
    print(r"""
<=====>
| |          MSPUB USER MANAGEMENT SYSTEM      /\
| |          /__\
| |          /_\/_\  

<=====>

Options:
-----
1. Install MSPUB Department Groups
2. Create Custom Group
3. Create New User
4. Delete Existing User
5. Assign User to Group
6. Force Password Change
7. List Users
8. Bulk User Import from CSV
9. Bulk User Deletion from CSV

X. Exit
""")

def main():
    check_root()

    while True:
        display_menu()
        choice = input("Select an option: ").strip().lower()

        if choice == "1":
            install_default_groups()
        elif choice == "2":

```

```

        create_custom_group()
    elif choice == "3":
        create_user()
    elif choice == "4":
        delete_user()
    elif choice == "5":
        assign_user_group()
    elif choice == "6":
        force_password_change()
    elif choice == "7":
        list_users()
    elif choice == "8":
        batch_user_import()
    elif choice == "9":
        bulk_delete_users()
    elif choice == "x":
        print("Exiting MSPUB User Management System.")
        log_action("Exited MSPUB User Management System.")
        break
    else:
        print("Invalid option. Please enter a valid option.")

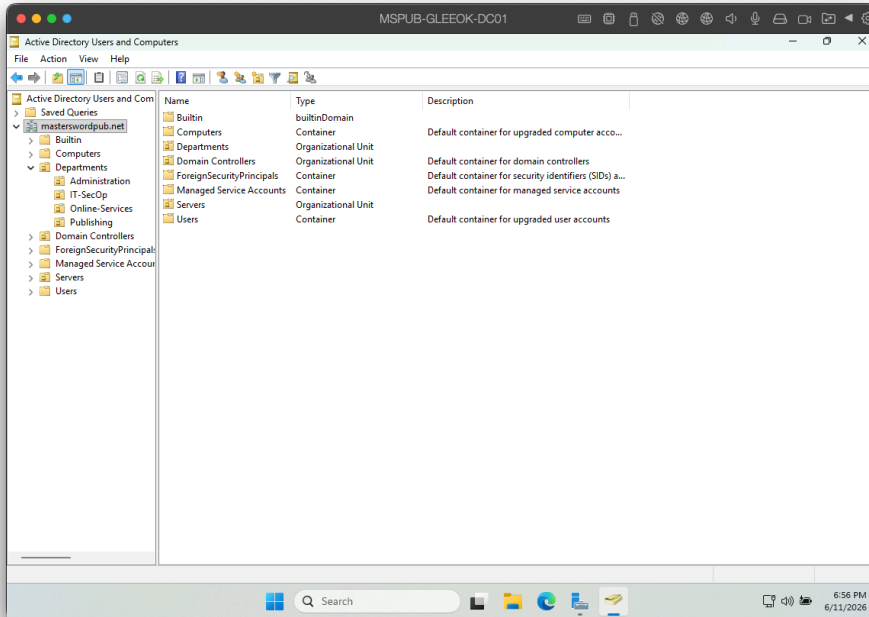
if __name__ == "__main__":
    main()

```

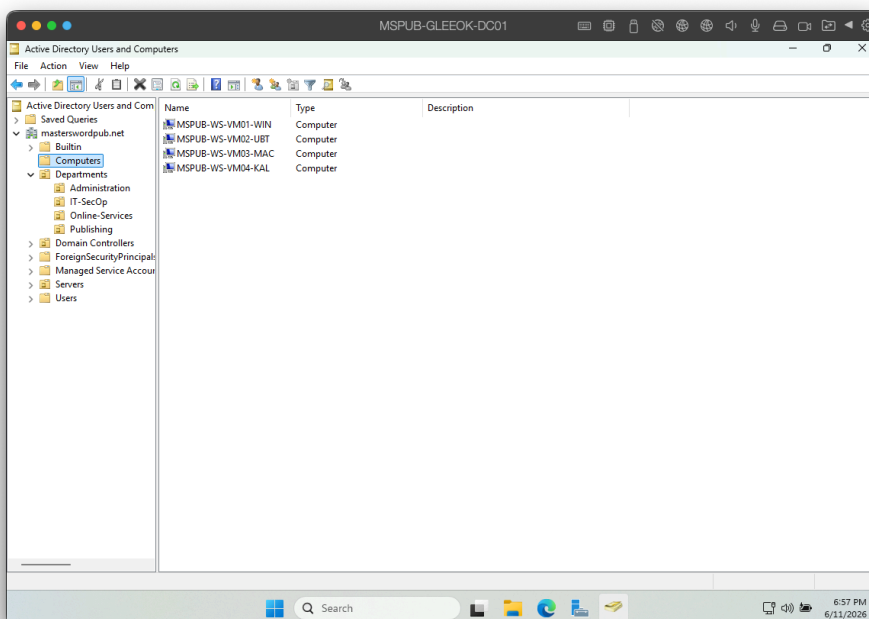
Appendix B - Network Configuration

Appendix B.1 - Active Directory Configuration

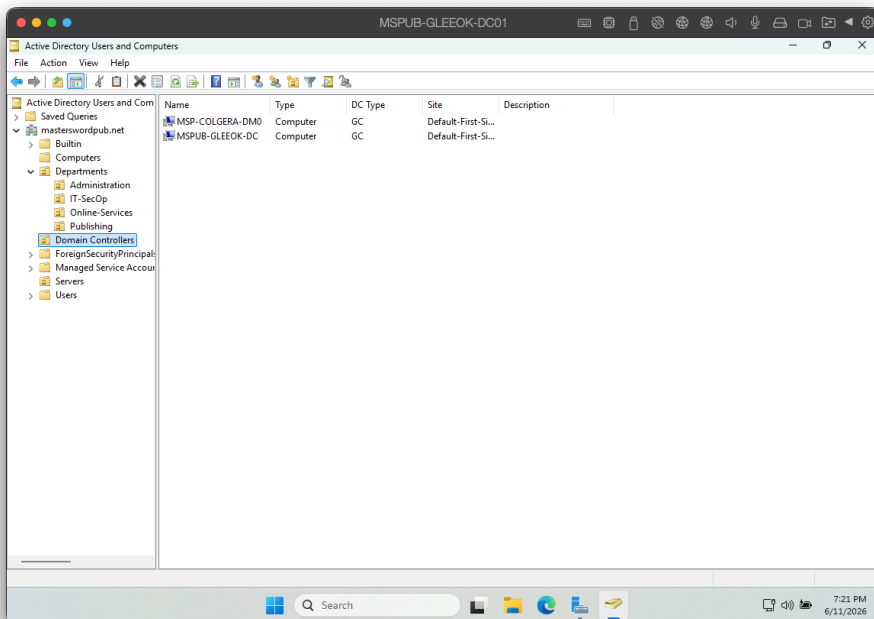
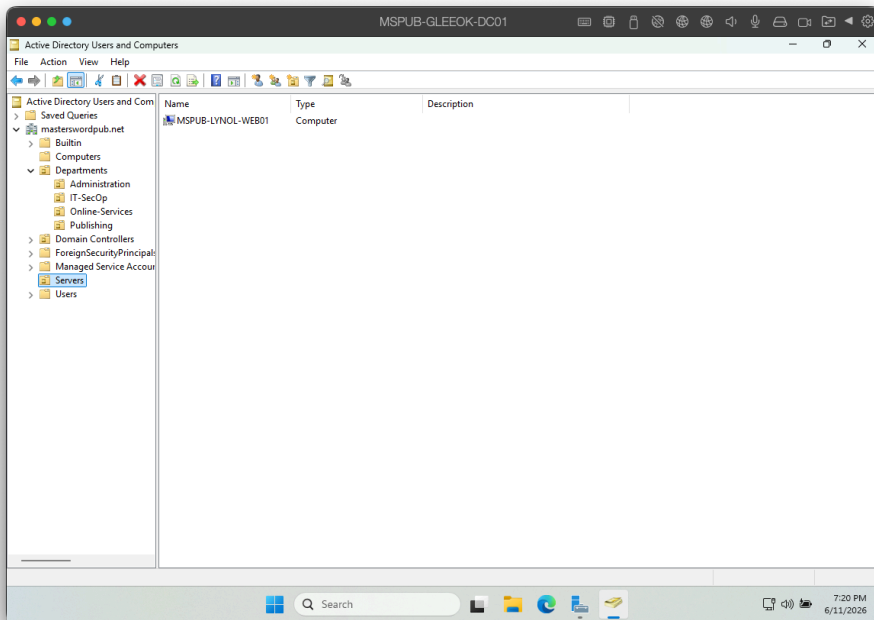
Organizational Units



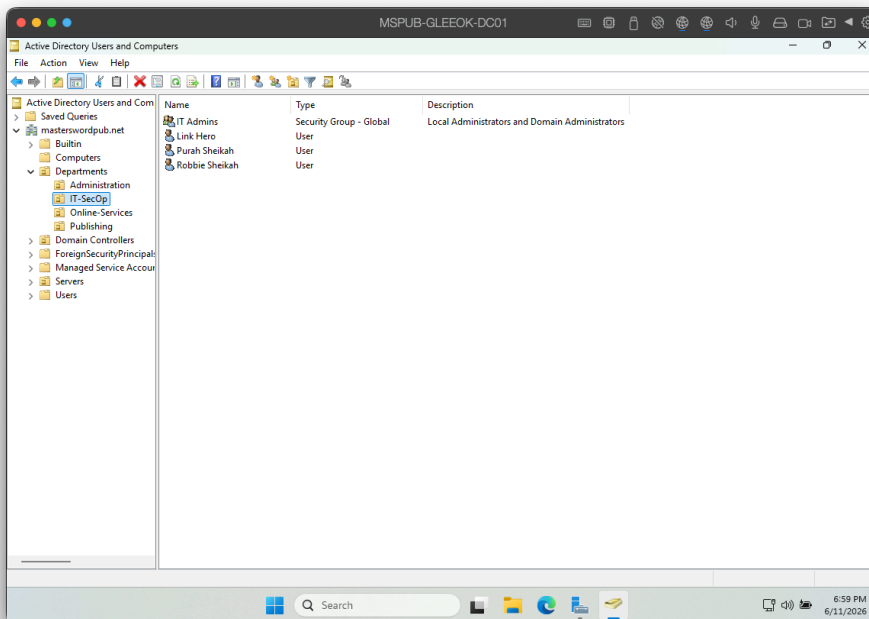
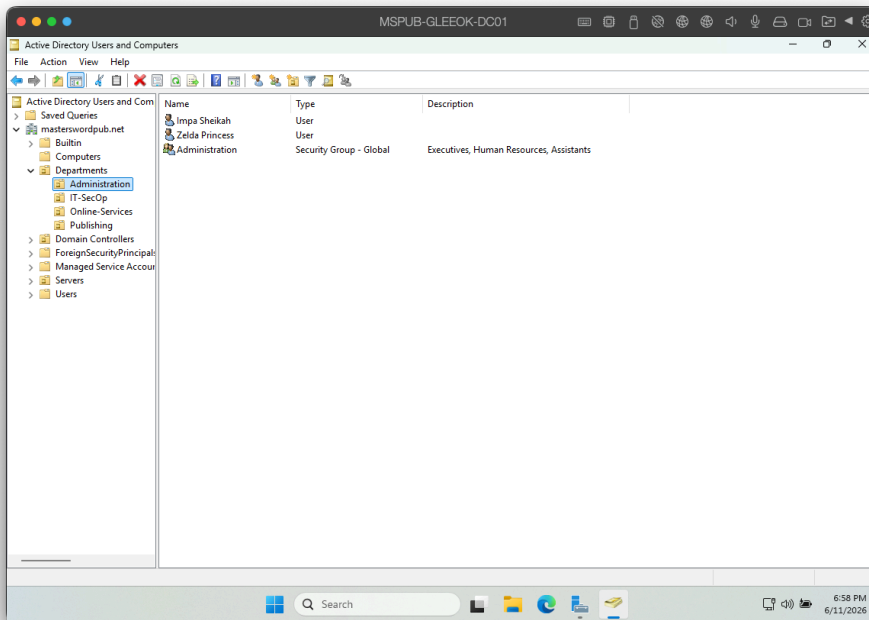
Computers

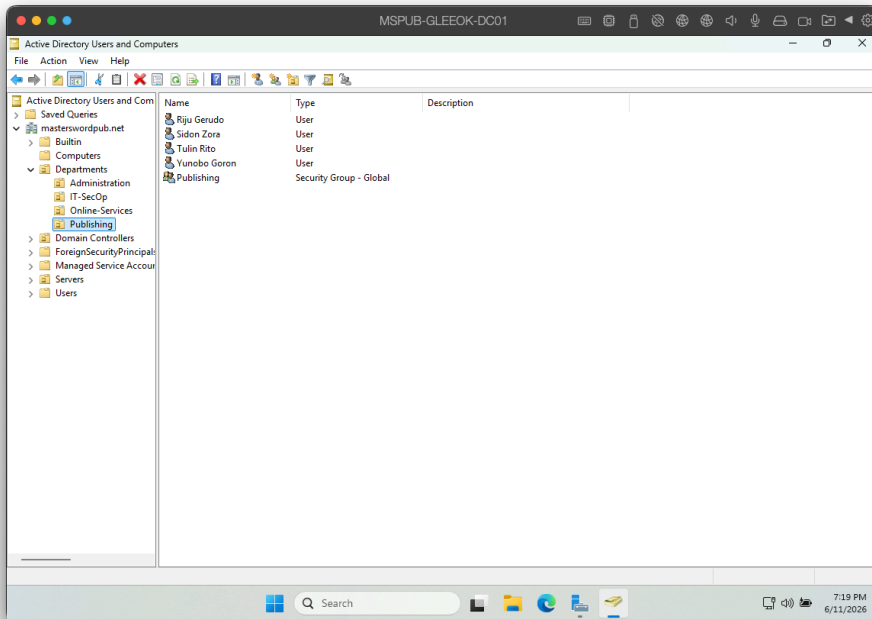
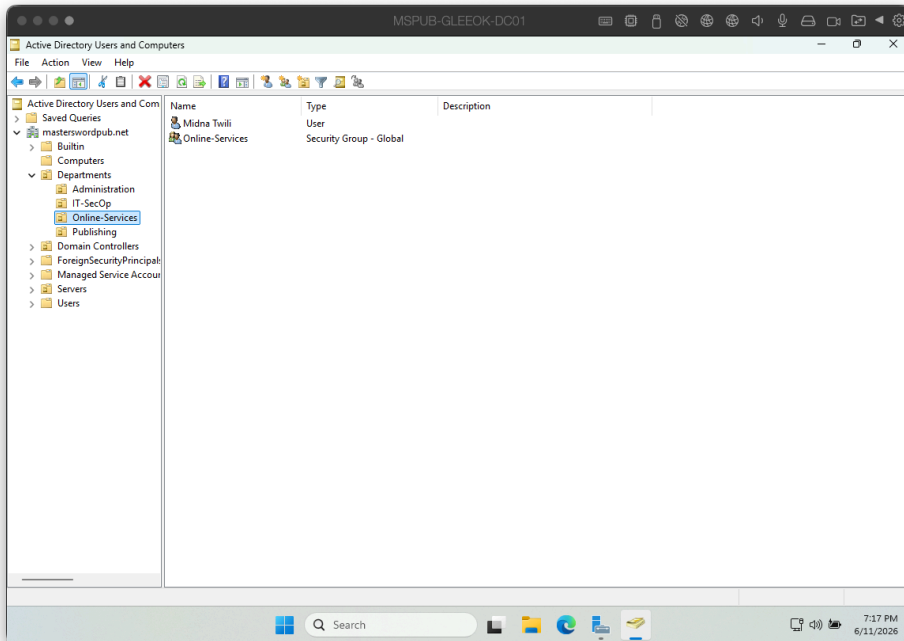


Servers/Domain Controllers

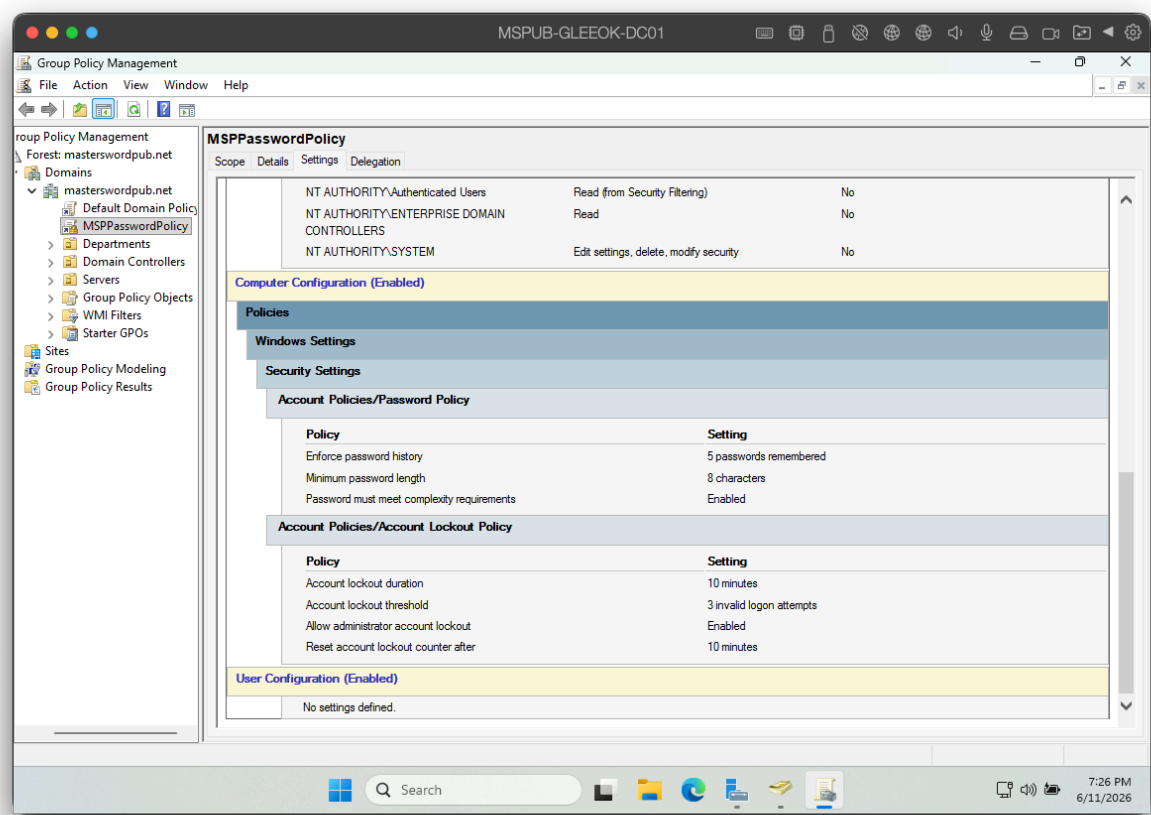


Users/Groups





Password Policy GPO



Appendix B.2 - UniFi Network Configuration

UniFi doesn't allow for the downloading of individual profiles per device. The entire UniFi OS Backup is the alternative which will include switches, routers, and WAP configurations.

Download Link: [MSPUB-UniFiOS-Config.unifi](#)

VLAN Configuration

Name	VLAN ID	Router	Subnet	IPv6 Subnet	DHCP	IP Leases	Available
Default	1	MSPUB-UDMP	192.168.1.0/24	-	Server	0 / 249	249
Infrastructure	10	MSPUB-UDMP	192.168.10.0/29	-	Server	1 / 5	1
Servers	20	MSPUB-UDMP	192.168.20.0/29	-	Server	0 / 5	5
Administration	30	MSPUB-UDMP	192.168.30.0/29	-	Server	0 / 5	5
Publishing	40	MSPUB-UDMP	192.168.40.0/29	-	Server	0 / 5	5
Online Services	50	MSPUB-UDMP	192.168.50.0/30	-	Server	0 / 1	1
IT-Cyber	60	MSPUB-UDMP	192.168.60.0/28	-	Server	0 / 13	13
Guest	80	MSPUB-UDMP	192.168.80.0/24	-	Server	2 / 249	247
Blackhole	255	MSPUB-UDMP	192.168.255.252/30	-	None	-	-

Firewall Rules

Name	Action	IP Version	Protocol	Src. Zone	Src.	Src. Port	Dst. Zone	Dst.	Dst. Port	ID	State
Allow Administratio...	Allow	IPv4	All	Internal	Administration	Any	Internal	Servers	Any	10000	All
Allow IT-Cyber Man...	Allow	IPv4	All	Internal	IT-Cyber	Any	Internal	All VLANs - IT	Any	10001	All
Allow IT-Cyber Man...	Allow	IPv4	All	Internal	IT-Cyber	Any	Hotspot	All VLANs - IT	Any	10000	All
Allow IT-Cyber Man...	Allow	IPv4	All	Internal	IT-Cyber	Any	External	All VLANs - IT	Any	10000	All
Allow IT-Cyber Man...	Allow	IPv4	All	Internal	IT-Cyber	Any	VPN	All VLANs - IT	Any	10000	All
Allow Online Servic...	Allow	IPv4	All	Internal	Online Services	Any	Internal	Servers	Any	10003	All
Allow Publishing to ...	Allow	IPv4	All	Internal	Publishing	Any	Internal	Servers	Any	10002	All
Block Blackhole VL...	Block	IPv4	All	Internal	Blackhole	Any	Internal	All Internal VLANs ...	Any	10004	All
Block Blackhole VL...	Block	IPv4	All	Internal	Blackhole	Any	Hotspot	All Internal VLANs ...	Any	10001	All
Block Blackhole VL...	Block	IPv4	All	Internal	Blackhole	Any	External	All Internal VLANs ...	Any	10001	All
Block Blackhole VL...	Block	IPv4	All	Internal	Blackhole	Any	VPN	All Internal VLANs ...	Any	10001	All

QoS Rules

Name	Behavior	Source	Dst. Port	Destination Or Target	Bandwidth Limits	Bandwidth Bursts
MSPUB-Critical-Infr...	Prioritize	Servers IT-Cyber	Any	Any	↓ Unlimited ↑ Unlimited	Off Off
MSPUB-Video-Calli...	Prioritize	Administrat... Publishi... +2	Any	Micr... Z +2	↓ Unlimited ↑ Unlimited	Off Off
MSPUB-VoIP-Calling	Prioritize	Administrat... Publishi... +2	Any	VoIP Services	↓ Unlimited ↑ Unlimited	Off Off
MSPUB-Business-T...	Prioritize	Publish... Online Servi... +1	Any	Any	↓ Unlimited ↑ Unlimited	Off Off
MSPUB-Guest-Net...	Limit	Guest	Any	Any	↓ 100 Mbps ↑ 25.0 Mbps	Off Off

Network Settings

Failover

ISP	Name	Interface	IPv4 Address	IPv6 Address	Port	Uptime	Peak Util.	Latency
• Z Zply Fiber	Primary WAN	WAN1	10.206.20.7	-	10	0.12%	↓ 0% ↑ 0%	2 ms
• Z Zply Fiber	Fallover WAN	WAN2	172.16.216.133	-	9	100%	-	2 ms

Other Settings

Default Security Posture ⓘ

Gateway mDNS Proxy ⓘ

IGMP Snooping ⓘ

L3 Network Isolation (ACL) ⓘ

Device Isolation (ACL) ⓘ

Network Lists ⓘ

Allow All

Block All

Auto ⓘ

Off

Custom

Wireless Network Settings

Name	Network	Broadcasting APs	Radio Band	Clients	Security
MSPUB-HQ	Native Network	All APs	2.4 GHz 5 GHz	-	WPA2/WPA3
MSPUB-Guest	Guest (80)	All APs	2.4 GHz	-	Open
Create NewManage					

UniFi Dream Machine Configuration

Gateway Ports

1

2

3

4

5

6

7

8

9

10

SFP

11

GbE Disconnected Disabled

Port	Name	WAN Assignment	Speed	Connection
1	DISAB...	Unassigned	Auto	-
2	DISAB...	Unassigned	Auto	-
3	DISAB...	Unassigned	Auto	-
4	DISAB...	Unassigned	Auto	-
5	DISAB...	Unassigned	Auto	-
6	DISAB...	Unassigned	Auto	-
7	SW01	Unassigned	GbE	-
8	SW02	Unassigned	GbE	-
9	WAN-...	WAN2	GbE	Z Ziplly Fiber
10	WAN-...	WAN1	GbE	Z Ziplly Fiber
SFP 11	DISAB...	Unassigned	Auto	-

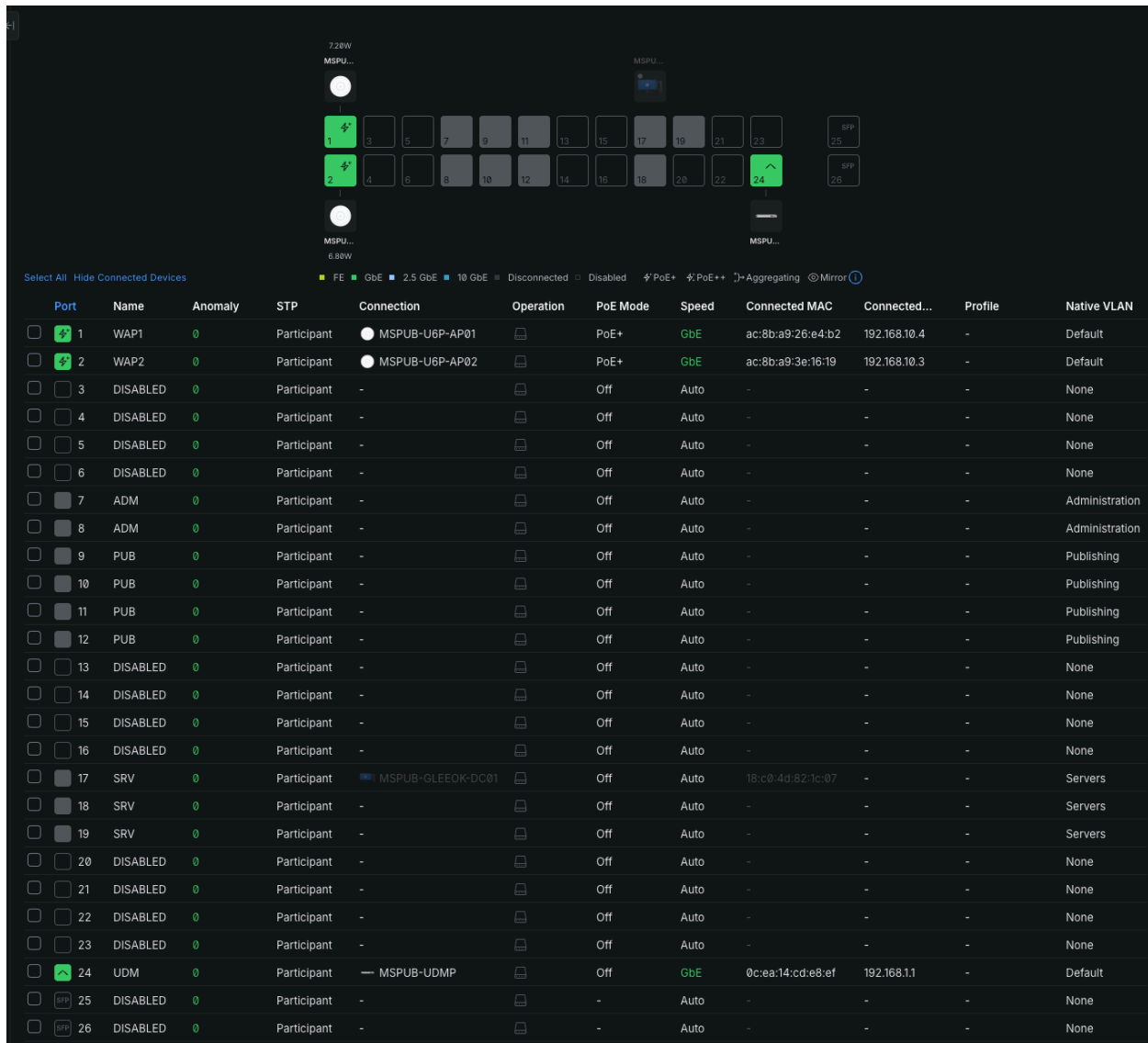
WAN Mode

Failover Only

Load Balancing

ID	Role	Status	SLA
WAN1	Primary	Online	Auto
WAN2	Backup	Online	Auto

UniFi Switch Pro 24 PoE+ Configuration



The image displays the UniFi Switch Pro 24 PoE+ configuration interface. At the top, there is a visual representation of the switch's 24 ports, grouped into two rows of 12. Ports 1 and 2 are highlighted in green, indicating they are active. Below the port diagram, a table provides detailed configuration for each port.

Legend:

- FE: Fast Ethernet
- GbE: Gigabit Ethernet
- 2.5 GbE: 2.5 Gigabit Ethernet
- 10 GbE: 10 Gigabit Ethernet
- Disconnected: Port is disconnected
- Disabled: Port is disabled
- PoE+: Power over Ethernet Plus
- PoE++: Power over Ethernet Plus Plus
- Aggregating: Port is part of a link aggregation group
- Mirror: Port is configured for mirroring

Port	Name	Anomaly	STP	Connection	Operation	PoE Mode	Speed	Connected MAC	Connected...	Profile	Native VLAN
☐ 1	WAP1	0	Participant	MSPUB-U6P-AP01	📶	PoE+	GbE	ac:8b:a9:26:e4:b2	192.168.10.4	-	Default
☐ 2	WAP2	0	Participant	MSPUB-U6P-AP02	📶	PoE+	GbE	ac:8b:a9:3e:16:19	192.168.10.3	-	Default
☐ 3	DISABLED	0	Participant	-	📶	Off	Auto	-	-	-	None
☐ 4	DISABLED	0	Participant	-	📶	Off	Auto	-	-	-	None
☐ 5	DISABLED	0	Participant	-	📶	Off	Auto	-	-	-	None
☐ 6	DISABLED	0	Participant	-	📶	Off	Auto	-	-	-	None
☐ 7	ADM	0	Participant	-	📶	Off	Auto	-	-	-	Administration
☐ 8	ADM	0	Participant	-	📶	Off	Auto	-	-	-	Administration
☐ 9	PUB	0	Participant	-	📶	Off	Auto	-	-	-	Publishing
☐ 10	PUB	0	Participant	-	📶	Off	Auto	-	-	-	Publishing
☐ 11	PUB	0	Participant	-	📶	Off	Auto	-	-	-	Publishing
☐ 12	PUB	0	Participant	-	📶	Off	Auto	-	-	-	Publishing
☐ 13	DISABLED	0	Participant	-	📶	Off	Auto	-	-	-	None
☐ 14	DISABLED	0	Participant	-	📶	Off	Auto	-	-	-	None
☐ 15	DISABLED	0	Participant	-	📶	Off	Auto	-	-	-	None
☐ 16	DISABLED	0	Participant	-	📶	Off	Auto	-	-	-	None
☐ 17	SRV	0	Participant	MSPUB-GLEEOK-DC01	📶	Off	Auto	18:c0:4d:82:1c:07	-	-	Servers
☐ 18	SRV	0	Participant	-	📶	Off	Auto	-	-	-	Servers
☐ 19	SRV	0	Participant	-	📶	Off	Auto	-	-	-	Servers
☐ 20	DISABLED	0	Participant	-	📶	Off	Auto	-	-	-	None
☐ 21	DISABLED	0	Participant	-	📶	Off	Auto	-	-	-	None
☐ 22	DISABLED	0	Participant	-	📶	Off	Auto	-	-	-	None
☐ 23	DISABLED	0	Participant	-	📶	Off	Auto	-	-	-	None
☐ 24	UDM	0	Participant	MSPUB-UDMP	📶	Off	GbE	0c:ea:14:cd:e8:ef	192.168.1.1	-	Default
☐ 25	DISABLED	0	Participant	-	📶	-	Auto	-	-	-	None
☐ 26	DISABLED	0	Participant	-	📶	-	Auto	-	-	-	None

UniFi Switch 24 Configuration

The interface displays the configuration for a UniFi Switch 24. At the top, a port configuration diagram shows 26 ports arranged in two rows. Port 24 is highlighted in green. Below the diagram, a table lists the configuration for each port.

Legend: ■ FE ■ GbE ■ Disconnected ■ Disabled ⌄ Aggregating ⌄ Mirror 1

Port	Name	Anomaly	STP	Connection	Operation	Speed	Connected MAC	Connected...	Profile	Native VLAN
☐ 1	OLS	0	Participant	MSPUB-UBT-OLS-WS		Auto	ac:87:a3:27:1a:65	-	-	Online Services
☐ 2	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 3	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 4	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 5	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 6	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 7	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 8	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 9	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 10	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 11	ITC	0	Participant	MSPUB-MAC-ITC-WS		Auto	44:10:9e:ac:49:06	-	-	IT-Cyber
☐ 12	ITC	0	Participant	-		Auto	-	-	-	IT-Cyber
☐ 13	ITC	0	Participant	-		Auto	-	-	-	IT-Cyber
☐ 14	ITC	0	Participant	-		Auto	-	-	-	IT-Cyber
☐ 15	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 16	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 17	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 18	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 19	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 20	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 21	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 22	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 23	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 24	UDM	0	Participant	MSPUB-UDMP		GbE	0c:ea:14:cd:e8:ef	192.168.1.1	-	Default
☐ 25	DISABLED	0	Participant	-		Auto	-	-	-	None
☐ 26	DISABLED	0	Participant	-		Auto	-	-	-	None

UniFi 6 Professional Access Point Configuration

MSPUB-U6P-AP01

Radios

Indoor

Outdoor

2.4 GHz

Channel Width

20

40

Channel

Auto

Transmit Power

Auto

High

Medium

Low

Custom

Disabled

☐ Minimum RSSI

5 GHz

Channel Width

20

40

80

160

Channel

Auto

Transmit Power

Auto

High

Medium

Low

Custom

Disabled

☐ Minimum RSSI

☒ Mesh Parent

☐ Mesh Connect

IP Settings

☐ Network Override

IP Configuration

DHCP

Static

Power & Resiliency

Power Source

MSPUB-US24P-SW01

Port

4 1

☐ Device Supervisor

MSPUB-U6P-AP02

Radios

Indoor

Outdoor

2.4 GHz

Channel Width

20

40

Channel

Auto

Transmit Power

Auto

High

Medium

Low

Custom

Disabled

☐ Minimum RSSI

5 GHz

Channel Width

20

40

80

160

Channel

Auto

Transmit Power

Auto

High

Medium

Low

Custom

Disabled

☐ Minimum RSSI

☒ Mesh Parent

☐ Mesh Connect

IP Settings

☐ Network Override

IP Configuration

DHCP

Static

Power & Resiliency

Power Source

MSPUB-US24P-SW01

Port

4 2

☐ Device Supervisor

Manage

☒ LED

☐ SNMP

Appendix B.3 - Amazon Web Services (AWS) Configuration

Amazon S3 Bucket: mspub-securiry-awareness-training

The screenshot shows the Amazon S3 console interface. The left sidebar contains navigation links for Amazon S3, Buckets, Files, Access management and security, and Storage management and insights. The main content area displays the 'mspub-securiry-awareness-training' bucket. The 'Objects' tab is selected, showing a list of objects with columns for Name, Type, Last modified, Size, and Storage class. The objects listed are:

Name	Type	Last modified	Size	Storage class
heart-container.html	html	June 5, 2026, 20:00:46 (UTC-07:00)	2.0 KB	Standard
index.html	html	June 4, 2026, 17:47:08 (UTC-07:00)	6.1 KB	Standard
login.html	html	June 5, 2026, 19:39:41 (UTC-07:00)	1.8 KB	Standard
phishing-email.html	html	June 5, 2026, 16:49:14 (UTC-07:00)	2.6 KB	Standard
style.css	css	June 5, 2026, 16:58:03 (UTC-07:00)	4.1 KB	Standard
user-failed-phishing-test.html	html	June 5, 2026, 19:39:41 (UTC-07:00)	3.2 KB	Standard

The screenshot shows the Amazon S3 console interface with the 'Permissions' tab selected for the bucket 'mspub-securiry-awareness-training'. The 'Permissions overview' section displays the 'Access finding' status as 'On'. The 'Block public access (bucket settings)' section shows 'Block all public access' is turned on. The 'Bucket policy' section displays the bucket policy JSON:

```
{
  "Version": "2008-10-17",
  "Id": "PolicyForCloudFrontPrivateContent",
  "Statement": [
    {
      "Sid": "AllowCloudFrontServicePrincipal",
      "Effect": "Allow",
      "Principal": {
        "Service": "cloudfront.amazonaws.com"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::mspub-securiry-awareness-training/*",
      "Condition": {
        "StringEquals": {
          "AWS:SourceAm": "arn:aws:cloudfront::481489197731:distribution/E1BX3OEKAWN7L"
        }
      }
    }
  ]
}
```

Amazon CloudFront: MSPUB Security Awareness

The screenshot shows the Amazon CloudFront console interface. On the left is a navigation menu with options like Distributions, Policies, Functions, Static IPs, VPC origins, SaaS, Telemetry, and Reports & analytics. The main content area displays the configuration for a distribution named 'MSPUB Security Awareness'.

Details:

- Distribution domain name: d3gvgnu61nqv1.cloudfront.net
- Billing: Pay-as-you-go (with a 'Switch to a plan' button)
- ARN: arn:aws:cloudfront:481489197731:distribution/E1BX3OEWAKNW7L
- Last modified: June 5, 2026 at 3:52:11 AM UTC

Settings:

- Name: MSPUB Security Awareness
- Description: -
- Price class: Use all edge locations (best performance)
- Supported HTTP versions: HTTP/2, HTTP/1.1, HTTP/1.0
- Alternate domain names: masterswordpub-net.jaycemangum.com (with a 'Route domains to CloudFront' button)
- Custom SSL certificate: masterswordpub-net.jaycemangum.com
- Security policy: TLSv1.2_2021
- Standard logging: Off
- Cookie logging: Off
- Default root object: index.html
- Cache tag header: -

Amazon Lambda: mspub-phishing-event logger

The screenshot shows the Amazon Lambda console interface for the 'mspub-phishing-event-logger' function. The left sidebar shows the function overview with a diagram and API Gateway integration. The main area shows the code source for the function.

Function overview:

- Function name: mspub-phishing-event-logger
- API Gateway: + Add trigger
- Layers: (0)
- Actions: Throttle, Copy ARN, Export to Infrastructure Composer, Download

Code source:

```
1 import json
2 import boto3
3 import uuid
4 from datetime import datetime, timezone
5
6 dynamodb = boto3.resource("dynamodb")
7 table = dynamodb.Table("MSPUB-SAT-2026-001-Results")
8
9 ALLOWED_ORIGINS = ["https://masterswordpub-net.jaycemangum.com"]
10
11
12 def response(status_code, body):
13     return {
14         "statusCode": status_code,
15         "headers": {
16             "Access-Control-Allow-Origin": ALLOWED_ORIGINS,
17             "Access-Control-Allow-Headers": "content-type",
18             "Access-Control-Allow-Methods": "OPTIONS,POST",
19             "Content-Type": "application/json"
20         },
21         "body": json.dumps(body)
22     }
23
24
25 def lambda_handler(event, context):
26     try:
27         method = event.get("requestContext", {}).get("http", {}).get("method", "")
28
29         if method == "OPTIONS":
30             return response(200, {"status": "cors_ok"})
31
32         body = event.get("body") or "{}"
```

DynamoDB

Dashboard

Tables

Explore items

PartiQL editor

Backups

Exports to S3

Imports from S3

Integrations

Reserved capacity

Settings

Tables (1)

Filter by tag
Any tag key

Filter by tag value
Any tag value

Find tables

MSPUB-SAT-2026-001-Results

MSPUB-SAT-2026-001-Results

Autopreview

View table details

▼ Scan or query items info

ScanQuery

Select a table or index
Table - MSPUB-SAT-2026-001-Results

Select attribute projection
All attributes

Filters - optional

RunReset

Completed · Items returned: 38 · Items scanned: 38 · Efficiency: 100% · RCUs consumed: 2

Table: MSPUB-SAT-2026-001-Results - Items returned (38)

ActionsCreate item

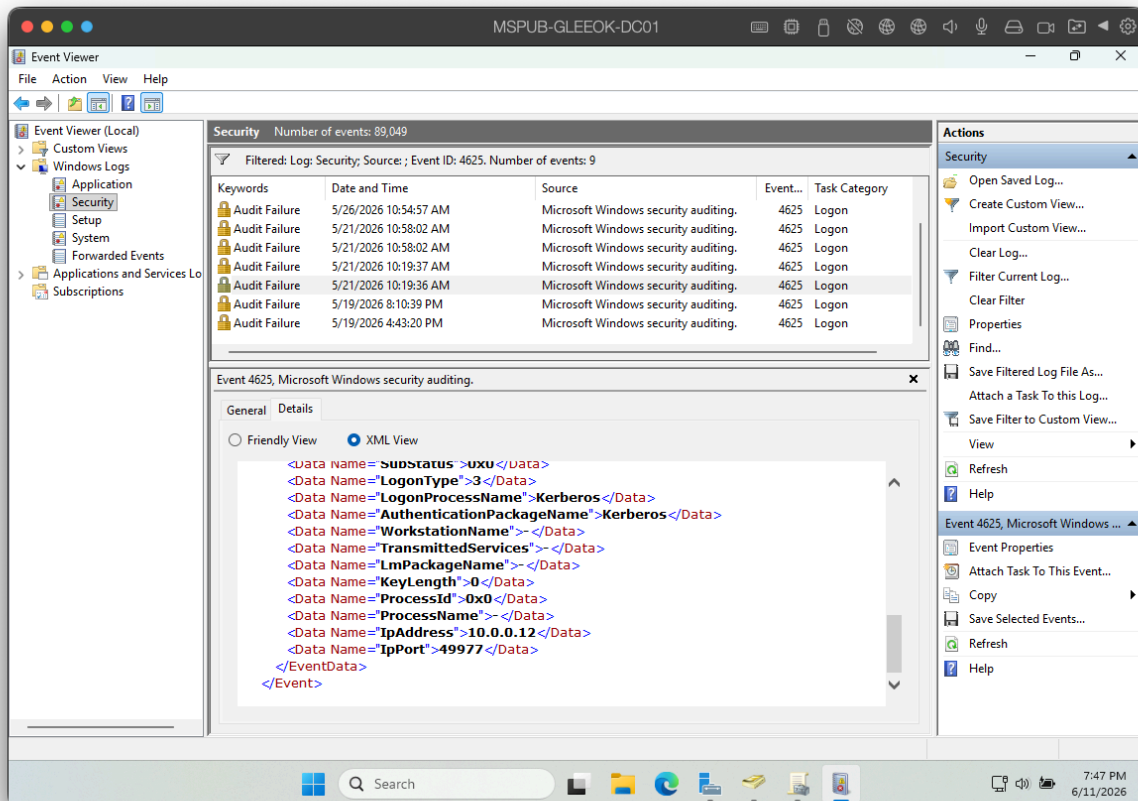
Scan started on June 11, 2026, 20:02:57

	event_id (String)	campaign_id	event_type	page	participant_id	timestamp
	564fb1ec-d720-4ebb...	MSPUB-SAT-2...	email_link_c...	login.html	participant39	2026-06-06T19:44:08.594130+0...
	eae0d0b3-9781-4fbc...	MSPUB-SAT-2...	training_pag...	user-failed-...	participant39	2026-06-06T20:11:09.108155+0...
	770be884-1a89-4cd8...	MSPUB-SAT-2...	training_pag...	user-failed-...	participant37	2026-06-06T17:34:42.457637+0...
	80e5ec7e-0d1f-48e5...	MSPUB-SAT-2...	training_co...	heart-conta...	participant40	2026-06-06T17:53:35.393395+0...
	341dc43b-f553-465b...	MSPUB-SAT-2...	email_link_c...	login.html	participant05	2026-06-06T17:16:17.677216+0...
	daf649d0-987d-48fe...	MSPUB-SAT-2...	email_link_c...	login.html	participant00	2026-06-06T22:39:22.327141+0...

Appendix C - Log Examples

Appendix C.1 - Active Directory Event Logs

User Failed Login Example:



Appendix C.2 - Automated Script Logs

Appendix C.2.a - Webserver Configuration Script Log

```
mangum — mspubadmin@MSPUB-LYNEL-WEB01: ~ — ssh mspubadmin@10.0.0.13 — 180x58

mspubadmin@MSPUB-LYNEL-WEB01:~$ sudo cat /var/log/mspub_setup.log
[2026-05-30 17:19:36] Root privilege check passed.
[2026-05-30 17:19:36] MSPUB Web Server Setup Script started.
[2026-05-30 17:20:24] SUCCESS: apt update
[2026-05-30 17:20:24] SUCCESS: apt upgrade -y
[2026-05-30 17:20:24] System packages updated successfully.
[2026-05-30 17:21:42] ERROR apt install nginx openssl-server ufw -y
[2026-05-30 17:21:42] WARNING: apt does not have a stable CLI interface. Use with caution in scripts.

Error: Failed to fetch http://security.ubuntu.com/ubuntu/pool/main/n/nginx/nginx-common_1.28.3-2ubuntu1.1_all.deb Unable to connect to security.ubuntu.com:http:
Error: Failed to fetch http://security.ubuntu.com/ubuntu/pool/main/n/nginx/nginx_1.28.3-2ubuntu1.1_arm64.deb Could not connect to security.ubuntu.com:80 (172.66.152.176), connecti
on timed out Could not connect to security.ubuntu.com:80 (104.20.20.246), connection timed out
Error: Unable to fetch some archives, maybe run apt update or try with --fix-missing?

[2026-05-30 17:26:46] Root privilege check passed.
[2026-05-30 17:26:46] MSPUB Web Server Setup Script started.
[2026-05-30 17:26:49] SUCCESS: apt update
[2026-05-30 17:27:42] SUCCESS: apt upgrade -y
[2026-05-30 17:27:42] System packages updated successfully.
[2026-05-30 17:27:48] SUCCESS: apt install nginx openssl-server ufw -y
[2026-05-30 17:27:48] Required packages installed successfully.
[2026-05-30 17:27:48] SUCCESS: ufw allow OpenSSH
[2026-05-30 17:27:48] SUCCESS: ufw allow 'Nginx Full'
[2026-05-30 17:27:48] SUCCESS: ufw --force enable
[2026-05-30 17:27:48] Firewall configured successfully.
[2026-05-30 17:27:48] SUCCESS: mkdir -p /var/www/masterswordpub.net
[2026-05-30 17:27:48] SUCCESS: chown -R www-data:www-data /var/www/masterswordpub.net
[2026-05-30 17:27:48] SUCCESS: chmod -R 755 /var/www/masterswordpub.net
[2026-05-30 17:27:48] Web directory created and permissions configured.
[2026-05-30 17:27:48] SUCCESS: mkdir -p /srv/sftp/publishing
[2026-05-30 17:27:48] SUCCESS: chmod 755 /srv/sftp
[2026-05-30 17:27:48] SUCCESS: chmod 755 /srv/sftp/publishing
[2026-05-30 17:27:48] SFTP directory structure created successfully.
[2026-05-30 17:27:48] SUCCESS: groupadd publishing
[2026-05-30 17:27:48] Publishing group created successfully.
[2026-05-30 17:27:48] SUCCESS: chown www-data:www-data /var/www/masterswordpub.net/index.html
[2026-05-30 17:27:48] SUCCESS: chmod 644 /var/www/masterswordpub.net/index.html
[2026-05-30 17:27:48] MSPUB website deployed successfully.
[2026-05-30 17:27:48] SUCCESS: rm /etc/nginx/sites-enabled/default
[2026-05-30 17:27:48] SUCCESS: ln -s /etc/nginx/sites-available/masterswordpub.net /etc/nginx/sites-enabled/masterswordpub.net
[2026-05-30 17:27:48] SUCCESS: nginx -t
[2026-05-30 17:27:48] SUCCESS: systemctl reload nginx
[2026-05-30 17:27:48] Nginx site configured successfully.
[2026-05-30 17:27:48] SUCCESS: systemctl enable nginx
[2026-05-30 17:27:48] SUCCESS: systemctl restart nginx
[2026-05-30 17:27:48] SUCCESS: systemctl enable ssh
[2026-05-30 17:27:48] SUCCESS: systemctl restart ssh
[2026-05-30 17:27:48] Required services enabled and started successfully.
[2026-05-30 17:27:48] VERIFIED: Nginx Service
[2026-05-30 17:27:48] VERIFIED: SSH Service
[2026-05-30 17:27:48] VERIFIED: Firewall Status
[2026-05-30 17:27:49] VERIFIED: Website Directory
[2026-05-30 17:27:49] VERIFIED: SFTP Directory
[2026-05-30 17:27:49] VERIFIED: Publishing Group
[2026-05-30 17:27:49] Configuration verification completed.
[2026-05-30 17:27:49] MSPUB web server setup completed successfully.

mspubadmin@MSPUB-LYNEL-WEB01:~$
```

Appendix C.2.b - MSPUB User Management System Log

```
[mspubadmin@mspub-test-ubtsrv:~/scripts/usrmgmt$ cat /var/log/mspub_user_mgmt.log
[2026-05-31 07:13:44] Root privilege check passed.
[2026-05-31 07:15:01] SUCCESS: groupadd administration
[2026-05-31 07:15:01] Created group: administration
[2026-05-31 07:15:01] Group already exists: publishing
[2026-05-31 07:15:01] SUCCESS: groupadd it-cyber
[2026-05-31 07:15:01] Created group: it-cyber
[2026-05-31 07:15:01] SUCCESS: groupadd online-services
[2026-05-31 07:15:01] Created group: online-services
[2026-05-31 07:15:24] SUCCESS: groupadd test-group
[2026-05-31 07:15:24] Created custom group: test-group
[2026-05-31 07:15:57] SUCCESS: useradd -m -s /bin/bash -G test-group j.mangum
[2026-05-31 07:15:57] Created user j.mangum and assigned to group test-group
[2026-05-31 07:15:57] SUCCESS: echo 'j.mangum:Hyrul3Warrior$' | chpasswd
[2026-05-31 07:15:57] MSPUB default password assigned to j.mangum
[2026-05-31 07:15:57] SUCCESS: chage -d 0 j.mangum
[2026-05-31 07:15:57] Forced password change at next login for j.mangum
[2026-05-31 07:17:05] Group assignment failed. User does not exist: it-cyber
[2026-05-31 07:17:30] SUCCESS: usermod -aG it-cyber j.mangum
[2026-05-31 07:17:30] Added j.mangum to group it-cyber
[2026-05-31 07:18:01] SUCCESS: chage -d 0 j.mangum
[2026-05-31 07:18:01] Forced password change for j.mangum
[2026-05-31 07:18:13] Displayed user list.
[2026-05-31 07:18:29] Exited MSPUB User Management System.
[2026-05-31 07:21:23] Root privilege check passed.
[2026-05-31 07:21:35] SUCCESS: userdel -r j.mangum
[2026-05-31 07:21:35] Deleted user: j.mangum
[2026-05-31 07:21:42] Displayed user list.
[2026-05-31 07:21:50] Exited MSPUB User Management System.
[2026-05-31 07:33:42] Root privilege check passed.
[2026-05-31 07:34:00] SUCCESS: useradd -m -s /bin/bash -G test-group j.mangum
[2026-05-31 07:34:00] Created user j.mangum and assigned to group test-group
[2026-05-31 07:34:00] MSPUB default password assigned to j.mangum
[2026-05-31 07:34:00] SUCCESS: chage -d 0 j.mangum
[2026-05-31 07:34:00] Forced password change at next login for j.mangum
[2026-05-31 07:34:12] Exited MSPUB User Management System.
[2026-05-31 21:02:28] Root privilege check passed.
[2026-05-31 21:03:08] SUCCESS: mkdir -p ./scripts/usrmgmt/
[2026-05-31 21:07:08] Root privilege check passed.
[2026-05-31 21:07:12] SUCCESS: mkdir -p /opt/mspub-usrmgmt/
[2026-05-31 21:07:12] SUCCESS: chmod 664 /opt/mspub-usrmgmt/batch-usr-import.csv
[2026-05-31 21:07:12] Created sample batch import CSV file: /opt/mspub-usrmgmt/batch-usr-import.csv
[2026-05-31 21:07:28] Batch import cancelled by administrator.
[2026-05-31 21:07:31] Exited MSPUB User Management System.
[2026-05-31 21:08:23] Root privilege check passed.
[2026-05-31 21:08:32] SUCCESS: useradd -m -s /bin/bash -G publishing l.hero
[2026-05-31 21:08:32] Batch import assigned default password to l.hero
[2026-05-31 21:08:32] SUCCESS: chage -d 0 l.hero
[2026-05-31 21:08:32] Batch import created user l.hero in group publishing. Default password set. Change password next login: TRUE
[2026-05-31 21:08:32] SUCCESS: useradd -m -s /bin/bash -G administration z.princess
[2026-05-31 21:08:32] Batch import assigned default password to z.princess
[2026-05-31 21:08:32] SUCCESS: chage -d 0 z.princess
[2026-05-31 21:08:32] Batch import created user z.princess in group administration. Default password set. Change password next login: TRUE
[2026-05-31 21:08:32] SUCCESS: useradd -m -s /bin/bash -G publishing s.zora
[2026-05-31 21:08:32] Batch import assigned default password to s.zora
[2026-05-31 21:08:32] SUCCESS: chage -d 0 s.zora
[2026-05-31 21:08:32] Batch import created user s.zora in group publishing. Default password set. Change password next login: TRUE
[2026-05-31 21:08:32] Batch user import completed.
[2026-05-31 21:09:14] Exited MSPUB User Management System.
[2026-05-31 21:10:34] Root privilege check passed.
[2026-05-31 21:10:42] Displayed user list.
[2026-05-31 21:10:56] SUCCESS: chmod 644 /opt/mspub-usrmgmt/bulk-usr-delete.csv
[2026-05-31 21:10:56] Created sample batch removal CSV file: /opt/mspub-usrmgmt/bulk-usr-delete.csv
[2026-05-31 21:11:01] Bulk user deletion cancelled by administrator.
[2026-05-31 21:11:02] Exited MSPUB User Management System.
[2026-05-31 21:13:11] Root privilege check passed.
[2026-05-31 21:13:18] SUCCESS: userdel -r z.princess
[2026-05-31 21:13:18] Bulk delete removed user: z.princess
[2026-05-31 21:13:18] Bulk delete skipped missing user: i.sheikah
[2026-05-31 21:13:18] Bulk delete skipped missing user: p.sheikah
[2026-05-31 21:13:18] Bulk user deletion completed.
[2026-05-31 21:13:51] Displayed user list.
[2026-05-31 21:13:58] Exited MSPUB User Management System.
[2026-05-31 21:43:44] Root privilege check passed.
[2026-05-31 21:44:02] Batch import cancelled by administrator.
[2026-05-31 21:44:09] Bulk user deletion cancelled by administrator.
[2026-05-31 21:45:41] Exited MSPUB User Management System.
mspubadmin@mspub-test-ubtsrv:~/scripts/usrmgmt$
```

Appendix C.3 - UniFi Event Logs

UniFi				
Category	Event	Description	Severity	Date / Time
Internet and ...	Internet Restored	Internet connection WAN1 on port 10 is restored after failing over to WAN2 (Zipty Fiber).		Today at 10:50:06 PM
Internet and ...	Internet Failover Active	Internet connection WAN1 on port 10 is down and WAN2 (Zipty Fiber) is now active.		Today at 10:24:05 PM
Client Devices	WiFi Client Disconnected	MSPUB-WIN-ADM-WS disconnected from MSPUB-HQ. Time Connected: 1m. Data Used: 115.47 KB (up) ...		Jun 2, 10:26:29 PM
Client Devices	WiFi Client Connected	MSPUB-WIN-ADM-WS connected to MSPUB-HQ on MSPUB-U6P-AP01. Connection Info: Ch. 64 (5 GHz,...		Jun 2, 10:25:20 PM
UniFi Devices	AP Channel Change	MSPUB-U6P-AP01 moved to channel 6 from 11.		Jun 2, 10:13:47 PM
UniFi Devices	DHCP Leases Exhausted	There are no available IP addresses to lease to devices on network Infrastructure.		Jun 2, 10:13:27 PM
UniFi Devices	DHCP Leases Exhausted	There are no available IP addresses to lease to devices on network Infrastructure.		Jun 2, 10:13:13 PM
Client Devices	Wired Client Connected	MSPUB-UBT-OLS-WS connected to Online Services on MSPUB-US24-SW02 Port 1. Link Speed: GbE. IP...		Jun 2, 10:12:59 PM
Client Devices	Wired Client Disconnected	MSPUB-MAC-ITC-WS disconnected from IT-Cyber on MSPUB-US24-SW02 Port 11. Time Connected: 1h...		Jun 2, 10:05:37 PM
Client Devices	Wired Client Connected	MSPUB-GLEEOK-DC01 connected to Servers on MSPUB-US24P-SW01 Port 17. Link Speed: GbE. IP: 192...		Jun 2, 9:47:28 PM
Client Devices	WiFi Client Disconnected	POSEIDON-WIN 33:23 disconnected from MSPUB-HQ. Time Connected: 8m 32s. Data Used: 24.25 KB (...)		Jun 2, 9:46:42 PM
Client Devices	WiFi Client Connected	MSPUB-WIN-ADM-WS disconnected from MSPUB-HQ. Time Connected: 15m 44s. Data Used: 291.32 K...		Jun 2, 9:44:52 PM
Client Devices	WiFi Client Connected	POSEIDON-WIN 33:23 connected to MSPUB-HQ on MSPUB-U6P-AP01. Connection Info: Ch. 64 (5 GHz,...		Jun 2, 9:38:18 PM
Client Devices	WiFi Client Connected	POSEIDON-WIN 33:23 connected to MSPUB-HQ on MSPUB-U6P-AP01. Connection Info: Ch. 64 (5 GHz,...		Jun 2, 9:38:12 PM
Client Devices	WiFi Client Disconnected	iPad 43:8c disconnected from MSPUB-HQ. Time Connected: 42s. Data Used: 10.19 KB (up) / 0.00 B (do...		Jun 2, 9:31:37 PM
Client Devices	WiFi Client Disconnected	MSPUB-GLEEOK-DC01 disconnected from Servers on MSPUB-US24P-SW01 Port 17. Time Connected: 1...		Jun 2, 9:30:37 PM
UniFi Devices	AP Channel Change	MSPUB-U6P-AP02 moved to channel 6 from 1.		Jun 2, 9:30:31 PM
UniFi Devices	DHCP Leases Exhausted	There are no available IP addresses to lease to devices on network Infrastructure.		Jun 2, 9:29:56 PM
UniFi Devices	DHCP Leases Exhausted	There are no available IP addresses to lease to devices on network Infrastructure.		Jun 2, 9:29:45 PM
Client Devices	WiFi Client Connected	MSPUB-WIN-PUB-WS connected to MSPUB-HQ on MSPUB-U6P-AP01. Connection Info: Ch. 64 (5 GHz,...		Jun 2, 9:29:12 PM
Client Devices	WiFi Client Connected	MSPUB-WIN-ADM-WS connected to MSPUB-HQ on MSPUB-U6P-AP01. Connection Info: Ch. 64 (5 GHz,...		Jun 2, 9:28:50 PM
Client Devices	Hotspot Guest Connected	Apple MacBook 69:90 connected to hotspot using Password authentication.		Jun 2, 9:27:39 PM
Client Devices	WiFi Client Disconnected	MSPUB-WIN-PUB-WS disconnected from MSPUB-HQ. Time Connected: 23m 56s. Data Used: 1.49 MB (...)		Jun 2, 9:27:37 PM
Client Devices	WiFi Client Connected	Apple MacBook 69:90 connected to MSPUB-Guest on MSPUB-U6P-AP01. Connection Info: Ch. 11 (2.4 G...		Jun 2, 9:26:39 PM
Client Devices	WiFi Client Connected	iPad 43:8c connected to MSPUB-HQ on MSPUB-U6P-AP01. Connection Info: Ch. 11 (2.4 GHz, 20 MHz), ...		Jun 2, 9:25:56 PM
Client Devices	WiFi Client Connected	iPad 43:8c connected to MSPUB-HQ on MSPUB-U6P-AP01. Connection Info: Ch. 11 (2.4 GHz, 20 MHz), ...		Jun 2, 9:25:32 PM
UniFi Devices	AP Channel Change	MSPUB-U6P-AP01 moved to channel 11 from 6.		Jun 2, 9:25:09 PM
UniFi Devices	DHCP Leases Exhausted	There are no available IP addresses to lease to devices on network Infrastructure.		Jun 2, 9:24:36 PM
Client Devices	WiFi Client Disconnected	MSPUB-WIN-ADM-WS disconnected from MSPUB-HQ. Time Connected: 28m 57s. Data Used: 17.31 MB ...		Jun 2, 9:18:41 PM
Client Devices	Wired Client Connected	MSPUB-GLEEOK-DC01 connected to Servers on MSPUB-US24P-SW01 Port 17. Link Speed: GbE. IP: 192...		Jun 2, 9:05:55 PM
Client Devices	WiFi Client Roamed	MSPUB-WIN-ADM-WS roamed from MSPUB-U6P-AP01 to MSPUB-U6P-AP02. Connection Info: Ch. 64 (...)		Jun 2, 9:04:43 PM
Client Devices	Wired Client Disconnected	MSPUB-GLEEOK-DC01 disconnected from Servers on MSPUB-US24P-SW01 Port 17. Time Connected: 5...		Jun 2, 9:04:37 PM
Client Devices	WiFi Client Disconnected	POSEIDON-WIN 33:23 disconnected from MSPUB-HQ. Time Connected: 2s. Data Used: 21.93 KB (up) / ...		Jun 2, 9:02:57 PM
Client Devices	WiFi Client Connected	POSEIDON-WIN 33:23 connected to MSPUB-HQ on MSPUB-U6P-AP01. Connection Info: Ch. 64 (5 GHz,...		Jun 2, 9:00:54 PM
Client Devices	WiFi Client Connected	POSEIDON-WIN 33:23 connected to MSPUB-HQ on MSPUB-U6P-AP01. Connection Info: Ch. 64 (5 GHz,...		Jun 2, 9:00:25 PM
UniFi Devices	DHCP Leases Exhausted	There are no available IP addresses to lease to devices on network Infrastructure.		Jun 2, 9:00:22 PM
Client Devices	WiFi Client Roamed	MSPUB-WIN-PUB-WS roamed from MSPUB-U6P-AP01 to MSPUB-U6P-AP02. Connection Info: Ch. 64 (...)		Jun 2, 8:58:05 PM
Client Devices	WiFi Client Connected	MSPUB-WIN-PUB-WS connected to MSPUB-HQ on MSPUB-U6P-AP01. Connection Info: Ch. 64 (5 GHz,...		Jun 2, 8:57:47 PM
Client Devices	Wired Client Connected	00:1b:63:97:06:17 connected to IT-Cyber on MSPUB-US24-SW02 Port 11. Link Speed: FE. IP: 169.254.79...		Jun 2, 8:53:51 PM
Client Devices	WiFi Client Disconnected	MSPUB-WIN-PUB-WS disconnected from MSPUB-HQ. Time Connected: 8m 52s. Data Used: 19.21 MB (...)		Jun 2, 8:51:37 PM
Client Devices	WiFi Client Connected	MSPUB-WIN-ADM-WS connected to MSPUB-HQ on MSPUB-U6P-AP01. Connection Info: Ch. 6 (2.4 GHz,...		Jun 2, 8:49:25 PM
Client Devices	WiFi Client Disconnected	MSPUB-WIN-ADM-WS disconnected from MSPUB-HQ. Time Connected: 1m 34s. Data Used: 375.14 KB ...		Jun 2, 8:49:19 PM
Client Devices	WiFi Client Roamed	MSPUB-WIN-ADM-WS roamed from MSPUB-U6P-AP02 to MSPUB-U6P-AP01. Connection Info: Ch. 6 (2...		Jun 2, 8:49:03 PM
Client Devices	WiFi Client Connected	MSPUB-WIN-ADM-WS connected to MSPUB-HQ on MSPUB-U6P-AP02. Connection Info: Ch. 64 (5 GHz,...		Jun 2, 8:47:48 PM
UniFi Devices	DHCP Leases Exhausted	There are no available IP addresses to lease to devices on network Infrastructure.		Jun 2, 8:47:46 PM
Client Devices	WiFi Client Connected	MSPUB-WIN-PUB-WS connected to MSPUB-HQ on MSPUB-U6P-AP01. Connection Info: Ch. 64 (5 GHz,...		Jun 2, 8:37:29 PM
Client Devices	WiFi Client Disconnected	MSPUB-WIN-PUB-WS disconnected from MSPUB-HQ. Time Connected: 1m 47s. Data Used: 153.66 KB (...)		Jun 2, 8:37:27 PM
Client Devices	WiFi Client Roamed	MSPUB-WIN-PUB-WS roamed from MSPUB-U6P-AP02 to MSPUB-U6P-AP01. Connection Info: Ch. 64 (...)		Jun 2, 8:37:21 PM

Appendix C.3 - Security Awareness Training DynamoDB Log

DynamoDB

Dashboard

Tables

Explore items

PartiQL editor

Backups

Exports to S3

Imports from S3

Integrations

Reserved capacity

Settings

DAX

Clusters

Subnet groups

Parameter groups

Events

Table: MSPUB-SAT-2026-001-Results - Items returned (38)

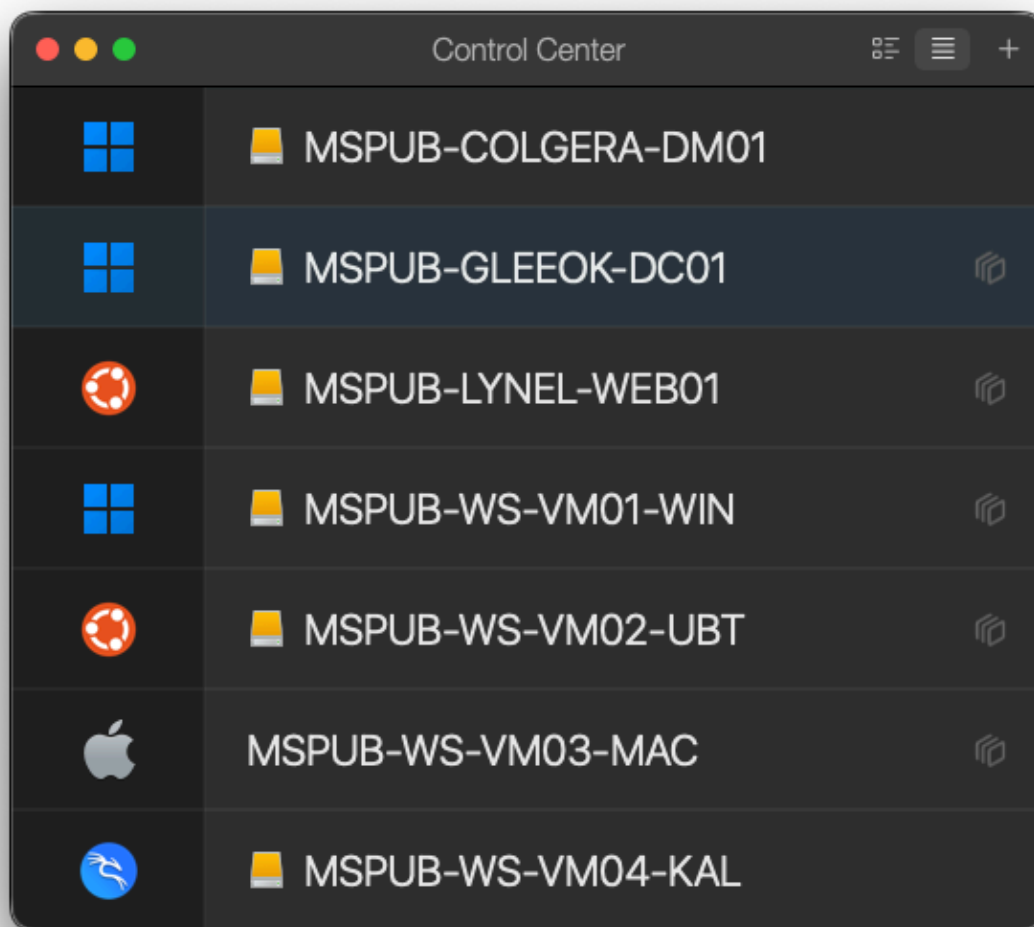
Scan started on June 11, 2026, 18:05:44

☐	event_id (String)	campaign_id	event_type	page	participant_id	timestamp
☐	a391a8b0-8267-4108...	MSPUB-SAT-2...	training_pag...	user-failed-...	participant38	2026-06-07T20:21:14.609603+0...
☐	5ca93a47-e0d5-41a5...	MSPUB-SAT-2...	training_pag...	user-failed-...	participant38	2026-06-07T00:44:12.223777+0...
☐	8a168a39-bf91-4502...	MSPUB-SAT-2...	login_attempt	login.html	participant38	2026-06-07T00:44:11.588426+0...
☐	c7cdee71-15e0-49b9...	MSPUB-SAT-2...	email_link_c...	login.html	participant38	2026-06-07T00:42:47.505471+0...
☐	7117e50c-dd76-4bc2...	MSPUB-SAT-2...	training_co...	heart-conta...	participant00	2026-06-06T22:40:54.333222+0...
☐	8978e452-a242-4c8d...	MSPUB-SAT-2...	training_pag...	user-failed-...	participant00	2026-06-06T22:40:22.062894+0...
☐	a079343a-c841-407b...	MSPUB-SAT-2...	login_attempt	login.html	participant00	2026-06-06T22:40:21.929175+0...
☐	dafe49d0-987d-48fa...	MSPUB-SAT-2...	email_link_c...	login.html	participant00	2026-06-06T22:39:22.327141+0...
☐	7b688623-a266-4bf2...	MSPUB-SAT-2...	email_link_c...	login.html	participant39	2026-06-06T20:11:28.970735+0...
☐	eae0d0b3-9781-4fbc...	MSPUB-SAT-2...	training_pag...	user-failed-...	participant39	2026-06-06T20:11:09.108155+0...
☐	3e9b4473-c549-48de...	MSPUB-SAT-2...	login_attempt	login.html	participant39	2026-06-06T20:11:07.983145+0...
☐	da4e5323-db86-4b0f...	MSPUB-SAT-2...	email_link_c...	login.html	participant39	2026-06-06T20:10:17.599937+0...
☐	564fb1ec-d720-4ebb...	MSPUB-SAT-2...	email_link_c...	login.html	participant39	2026-06-06T19:44:08.594130+0...
☐	80e5ec7e-0d1f-48e5...	MSPUB-SAT-2...	training_co...	heart-conta...	participant40	2026-06-06T17:53:35.393395+0...
☐	b5aca4f9-0910-4f08...	MSPUB-SAT-2...	training_pag...	user-failed-...	participant40	2026-06-06T17:52:15.667498+0...
☐	5e8dcf5a-cb79-4825...	MSPUB-SAT-2...	login_attempt	login.html	participant40	2026-06-06T17:52:15.184777+0...
☐	8000f380-7e9f-42ca...	MSPUB-SAT-2...	email_link_c...	login.html	participant40	2026-06-06T17:51:44.856940+0...
☐	5f955cc6-3ad5-4178...	MSPUB-SAT-2...	email_link_c...	login.html	participant40	2026-06-06T17:50:28.793784+0...
☐	a7b33b30-63b6-4d05...	MSPUB-SAT-2...	training_co...	heart-conta...	participant37	2026-06-06T17:34:58.948942+0...
☐	770be884-1a89-4cd8...	MSPUB-SAT-2...	training_pag...	user-failed-...	participant37	2026-06-06T17:34:42.457637+0...
☐	8889fade-6e03-4fd2...	MSPUB-SAT-2...	login_attempt	login.html	participant37	2026-06-06T17:34:41.976726+0...
☐	5c263746-2623-4480...	MSPUB-SAT-2...	email_link_c...	login.html	participant37	2026-06-06T17:33:32.160904+0...
☐	f23e238b-5f66-4512...	MSPUB-SAT-2...	email_link_c...	login.html	participant37	2026-06-06T17:33:31.204797+0...
☐	9095540c-9140-4545...	MSPUB-SAT-2...	email_link_c...	login.html	participant01	2026-06-06T17:26:23.994848+0...
☐	f1819c16-b705-44dc...	MSPUB-SAT-2...	email_link_c...	login.html	participant01	2026-06-06T17:26:17.384412+0...
☐	ffc9b617-adb0-4daa...	MSPUB-SAT-2...	email_link_c...	login.html	participant18	2026-06-06T17:22:29.155774+0...
☐	c81fd37a-c5c6-4d80...	MSPUB-SAT-2...	email_link_c...	login.html	participant18	2026-06-06T17:22:25.333131+0...

Appendix D - Validation Screenshots

Appendix D.1 - Home Lab (MSPUB Virtual Environment)

Virtual Machine Inventory



	 MSPUB-COLGERA-DM01	
	 MSPUB-GLEEOK-DC01	
	 MSPUB-LYNEL-WEB01	
	 MSPUB-WS-VM01-WIN	
	 MSPUB-WS-VM02-UBT	
	MSPUB-WS-VM03-MAC	
	 MSPUB-WS-VM04-KAL	

Network Connectivity

MSPUB-GLEEOK-DC01

ping

```
Administrator: Windows PowerShell
PS C:\Users\Administrator> ping 10.0.0.101

Pinging 10.0.0.101 with 32 bytes of data:
Reply from 10.0.0.101: bytes=32 time<1ms TTL=128
Reply from 10.0.0.101: bytes=32 time<1ms TTL=128
Reply from 10.0.0.101: bytes=32 time<1ms TTL=128
Reply from 10.0.0.101: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.101:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\Administrator> ping 10.0.0.102

Pinging 10.0.0.102 with 32 bytes of data:
Reply from 10.0.0.102: bytes=32 time<1ms TTL=64
Reply from 10.0.0.102: bytes=32 time<1ms TTL=64
Reply from 10.0.0.102: bytes=32 time<1ms TTL=64
Reply from 10.0.0.102: bytes=32 time<1ms TTL=64

Ping statistics for 10.0.0.102:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\Administrator> ping 10.0.0.103

Pinging 10.0.0.103 with 32 bytes of data:
Reply from 10.0.0.103: bytes=32 time<1ms TTL=64
Reply from 10.0.0.103: bytes=32 time<1ms TTL=64
Reply from 10.0.0.103: bytes=32 time<1ms TTL=64
Reply from 10.0.0.103: bytes=32 time<1ms TTL=64

Ping statistics for 10.0.0.103:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
```

MSPUB-WS-VM01-WIN

MSPUB-WS-VM02-UBT

MSPUB-WS-VM03-MAC

```
Administrator: Windows PowerShell
PS C:\Users\Administrator> ping 10.0.0.104

Pinging 10.0.0.104 with 32 bytes of data:
Reply from 10.0.0.104: bytes=32 time<1ms TTL=64
Reply from 10.0.0.104: bytes=32 time<1ms TTL=64
Reply from 10.0.0.104: bytes=32 time<1ms TTL=64
Reply from 10.0.0.104: bytes=32 time<1ms TTL=64

Ping statistics for 10.0.0.104:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\Administrator> ping 10.0.0.12

Pinging 10.0.0.12 with 32 bytes of data:
Reply from 10.0.0.12: bytes=32 time<1ms TTL=128
Reply from 10.0.0.12: bytes=32 time<1ms TTL=128
Reply from 10.0.0.12: bytes=32 time<1ms TTL=128
Reply from 10.0.0.12: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.12:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
PS C:\Users\Administrator> ping 10.0.0.13

Pinging 10.0.0.13 with 32 bytes of data:
Reply from 10.0.0.13: bytes=32 time<1ms TTL=64
Reply from 10.0.0.13: bytes=32 time<1ms TTL=64
Reply from 10.0.0.13: bytes=32 time<1ms TTL=64
Reply from 10.0.0.13: bytes=32 time<1ms TTL=64

Ping statistics for 10.0.0.13:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
```

MSPUB-WS-VM04-KAL

MSPUB-COLGERA-DM01

MSPUB-LYNEL-WEB01

MSPUB-COLGERA-DM01

Ping

```
PS C:\Users\Administrator.MSPUB> ping 10.0.0.101

Pinging 10.0.0.101 with 32 bytes of data:
Reply from 10.0.0.101: bytes=32 time<1ms TTL=128
Reply from 10.0.0.101: bytes=32 time<1ms TTL=128
Reply from 10.0.0.101: bytes=32 time<1ms TTL=128
Reply from 10.0.0.101: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.101:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\Administrator.MSPUB> ping 10.0.0.102

Pinging 10.0.0.102 with 32 bytes of data:
Reply from 10.0.0.102: bytes=32 time<1ms TTL=64
Reply from 10.0.0.102: bytes=32 time<1ms TTL=64
Reply from 10.0.0.102: bytes=32 time<1ms TTL=64
Reply from 10.0.0.102: bytes=32 time<1ms TTL=64

Ping statistics for 10.0.0.102:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\Administrator.MSPUB> ping 10.0.0.103

Pinging 10.0.0.103 with 32 bytes of data:
Reply from 10.0.0.103: bytes=32 time<1ms TTL=64
Reply from 10.0.0.103: bytes=32 time<1ms TTL=64
Reply from 10.0.0.103: bytes=32 time<1ms TTL=64
Reply from 10.0.0.103: bytes=32 time<1ms TTL=64

Ping statistics for 10.0.0.103:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
```

MSPUB-WS-VM01-WIN

MSPUB-WS-VM02-UBT

MSPUB-WS-VM03-MAC

```
PS C:\Users\Administrator.MSPUB> ping 10.0.0.104

Pinging 10.0.0.104 with 32 bytes of data:
Reply from 10.0.0.104: bytes=32 time<1ms TTL=64
Reply from 10.0.0.104: bytes=32 time<1ms TTL=64
Reply from 10.0.0.104: bytes=32 time<1ms TTL=64
Reply from 10.0.0.104: bytes=32 time<1ms TTL=64

Ping statistics for 10.0.0.104:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\Administrator.MSPUB> ping 10.0.0.11

Pinging 10.0.0.11 with 32 bytes of data:
Reply from 10.0.0.11: bytes=32 time<1ms TTL=128
Reply from 10.0.0.11: bytes=32 time<1ms TTL=128
Reply from 10.0.0.11: bytes=32 time<1ms TTL=128
Reply from 10.0.0.11: bytes=32 time<1ms TTL=128

Ping statistics for 10.0.0.11:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\Administrator.MSPUB> ping 10.0.0.13

Pinging 10.0.0.13 with 32 bytes of data:
Reply from 10.0.0.13: bytes=32 time<1ms TTL=64
Reply from 10.0.0.13: bytes=32 time<1ms TTL=64
Reply from 10.0.0.13: bytes=32 time<1ms TTL=64
Reply from 10.0.0.13: bytes=32 time<1ms TTL=64

Ping statistics for 10.0.0.13:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
```

MSPUB-WS-VM04-KAL

MSPUB-GLEEOK-DC01

MSPUB-LYNEL-WEB01

MSPUB-LYNEL-WEB-01

nslookup

```
mspubadmin@MSPUB-LYNEL-WEB01:~$ nslookup masterswordpub.net
Server:          127.0.0.53
Address:         127.0.0.53#53

Non-authoritative answer:
Name:   masterswordpub.net
Address: 10.0.0.12
Name:   masterswordpub.net
Address: 10.0.0.11
Name:   masterswordpub.net
Address: 10.211.55.21
Name:   masterswordpub.net
Address: 10.211.55.14
Name:   masterswordpub.net
Address: fdb2:2c26:f4e4:0:4133:ae01:afef:88f2
Name:   masterswordpub.net
Address: fdb2:2c26:f4e4:0:270:c74e:d3f5:22fd
Name:   masterswordpub.net
Address: fdb2:2c26:f4e4:0:8e77:dd06:5de7:7c60
```

ping

```
mspubadmin@MSPUB-LYNEL-WEB01:~$ ping -c 4 10.0.0.101
PING 10.0.0.101 (10.0.0.101) 56(64) bytes of data:
64 bytes from 10.0.0.101: icmp_seq=1 ttl=128 time=1.99 ms
64 bytes from 10.0.0.101: icmp_seq=2 ttl=128 time=0.525 ms
64 bytes from 10.0.0.101: icmp_seq=3 ttl=128 time=0.945 ms
64 bytes from 10.0.0.101: icmp_seq=4 ttl=128 time=0.795 ms
--- 10.0.0.101 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3034ms
rtt min/avg/max/mdev = 0.625/1.087/1.986/0.530 ms
mcpubadmin@MSPUB-LYNEL-WEB01:~$ ping -c 4 10.0.0.102
PING 10.0.0.102 (10.0.0.102) 56(64) bytes of data:
64 bytes from 10.0.0.102: icmp_seq=1 ttl=64 time=0.668 ms
64 bytes from 10.0.0.102: icmp_seq=2 ttl=64 time=0.564 ms
64 bytes from 10.0.0.102: icmp_seq=3 ttl=64 time=0.326 ms
64 bytes from 10.0.0.102: icmp_seq=4 ttl=64 time=0.471 ms
--- 10.0.0.102 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3095ms
rtt min/avg/max/mdev = 0.326/0.507/0.668/0.125 ms
mcpubadmin@MSPUB-LYNEL-WEB01:~$ ping -c 4 10.0.0.103
PING 10.0.0.103 (10.0.0.103) 56(64) bytes of data:
64 bytes from 10.0.0.103: icmp_seq=1 ttl=64 time=0.433 ms
64 bytes from 10.0.0.103: icmp_seq=2 ttl=64 time=0.569 ms
64 bytes from 10.0.0.103: icmp_seq=3 ttl=64 time=0.366 ms
64 bytes from 10.0.0.103: icmp_seq=4 ttl=64 time=0.490 ms
--- 10.0.0.103 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3076ms
rtt min/avg/max/mdev = 0.366/0.449/0.509/0.055 ms
mcpubadmin@MSPUB-LYNEL-WEB01:~$ ping -c 4 10.0.0.104
PING 10.0.0.104 (10.0.0.104) 56(64) bytes of data:
64 bytes from 10.0.0.104: icmp_seq=1 ttl=64 time=0.630 ms
64 bytes from 10.0.0.104: icmp_seq=2 ttl=64 time=0.514 ms
64 bytes from 10.0.0.104: icmp_seq=3 ttl=64 time=0.402 ms
64 bytes from 10.0.0.104: icmp_seq=4 ttl=64 time=0.284 ms
--- 10.0.0.104 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3067ms
rtt min/avg/max/mdev = 0.284/0.457/0.630/0.128 ms
mcpubadmin@MSPUB-LYNEL-WEB01:~$
```

MSPUB-WS-VM01-WIN

MSPUB-WS-VM02-UBT

MSPUB-WS-VM03-MAC

MSPUB-WS-VM04-KAL

```
MSPUB-LYNEL-WEB01
mspubadmin@MSPUB-LYNEL-WEB01:~$ ping -c 4 10.0.0.11
PING 10.0.0.11 (10.0.0.11) 56(64) bytes of data:
64 bytes from 10.0.0.11: icmp_seq=1 ttl=128 time=0.403 ms
64 bytes from 10.0.0.11: icmp_seq=2 ttl=128 time=0.438 ms
64 bytes from 10.0.0.11: icmp_seq=3 ttl=128 time=0.701 ms
64 bytes from 10.0.0.11: icmp_seq=4 ttl=128 time=0.355 ms
--- 10.0.0.11 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3053ms
rtt min/avg/max/ndev = 0.355/0.494/0.701/0.168 ms
mspubadmin@MSPUB-LYNEL-WEB01:~$ ping -c 4 10.0.0.12
PING 10.0.0.12 (10.0.0.12) 56(64) bytes of data:
64 bytes from 10.0.0.12: icmp_seq=1 ttl=128 time=0.456 ms
64 bytes from 10.0.0.12: icmp_seq=2 ttl=128 time=0.568 ms
64 bytes from 10.0.0.12: icmp_seq=3 ttl=128 time=0.769 ms
64 bytes from 10.0.0.12: icmp_seq=4 ttl=128 time=0.695 ms
--- 10.0.0.12 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3085ms
rtt min/avg/max/ndev = 0.456/0.649/0.805/0.143 ms
mspubadmin@MSPUB-LYNEL-WEB01:~$
```

MSPUB-GLEEOK-DC01

MSPUB-COLGERA-DM01

MSPUB-WS-VM01-WIN

ping

```
Administrator: Windows Powe
MSPUB-WS-VM01-WIN
PS C:\Users\Administrator> ping 10.0.0.102

Pinging 10.0.0.102 with 32 bytes of data:
Reply from 10.0.0.102: bytes=32 time=1ms TTL=64
Reply from 10.0.0.102: bytes=32 time<1ms TTL=64
Reply from 10.0.0.102: bytes=32 time<1ms TTL=64
Reply from 10.0.0.102: bytes=32 time<1ms TTL=64

Ping statistics for 10.0.0.102:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
PS C:\Users\Administrator> ping 10.0.0.103

Pinging 10.0.0.103 with 32 bytes of data:
Reply from 10.0.0.103: bytes=32 time<1ms TTL=64
Reply from 10.0.0.103: bytes=32 time<1ms TTL=64
Reply from 10.0.0.103: bytes=32 time<1ms TTL=64
Reply from 10.0.0.103: bytes=32 time<1ms TTL=64

Ping statistics for 10.0.0.103:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\Administrator> ping 10.0.0.104

Pinging 10.0.0.104 with 32 bytes of data:
Reply from 10.0.0.104: bytes=32 time<1ms TTL=64
Reply from 10.0.0.104: bytes=32 time<1ms TTL=64
Reply from 10.0.0.104: bytes=32 time<1ms TTL=64
Reply from 10.0.0.104: bytes=32 time<1ms TTL=64

Ping statistics for 10.0.0.104:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
```

MSPUB-WS-VM02-UBT

MSPUB-WS-VM03-MAC

MSPUB-WS-VM04-KAL

```
Administrator: Windows Powe... MSPUB-WS-VM01-WIN
Ping 10.0.0.11 with 32 bytes of data:
Reply from 10.0.0.11: bytes=32 time<1ms TTL=128
Reply from 10.0.0.11: bytes=32 time<1ms TTL=128
Reply from 10.0.0.11: bytes=32 time<1ms TTL=128
Reply from 10.0.0.11: bytes=32 time<1ms TTL=128
Ping statistics for 10.0.0.11:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\Administrator> ping 10.0.0.12

Ping 10.0.0.12 with 32 bytes of data:
Reply from 10.0.0.12: bytes=32 time<1ms TTL=128
Reply from 10.0.0.12: bytes=32 time<1ms TTL=128
Reply from 10.0.0.12: bytes=32 time<1ms TTL=128
Reply from 10.0.0.12: bytes=32 time<1ms TTL=128
Ping statistics for 10.0.0.12:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
PS C:\Users\Administrator> ping 10.0.0.13

Ping 10.0.0.13 with 32 bytes of data:
Reply from 10.0.0.13: bytes=32 time<1ms TTL=64
Reply from 10.0.0.13: bytes=32 time<1ms TTL=64
Reply from 10.0.0.13: bytes=32 time<1ms TTL=64
Reply from 10.0.0.13: bytes=32 time<1ms TTL=64
Ping statistics for 10.0.0.13:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

MSPUB-WS-VM02-UBT

ping

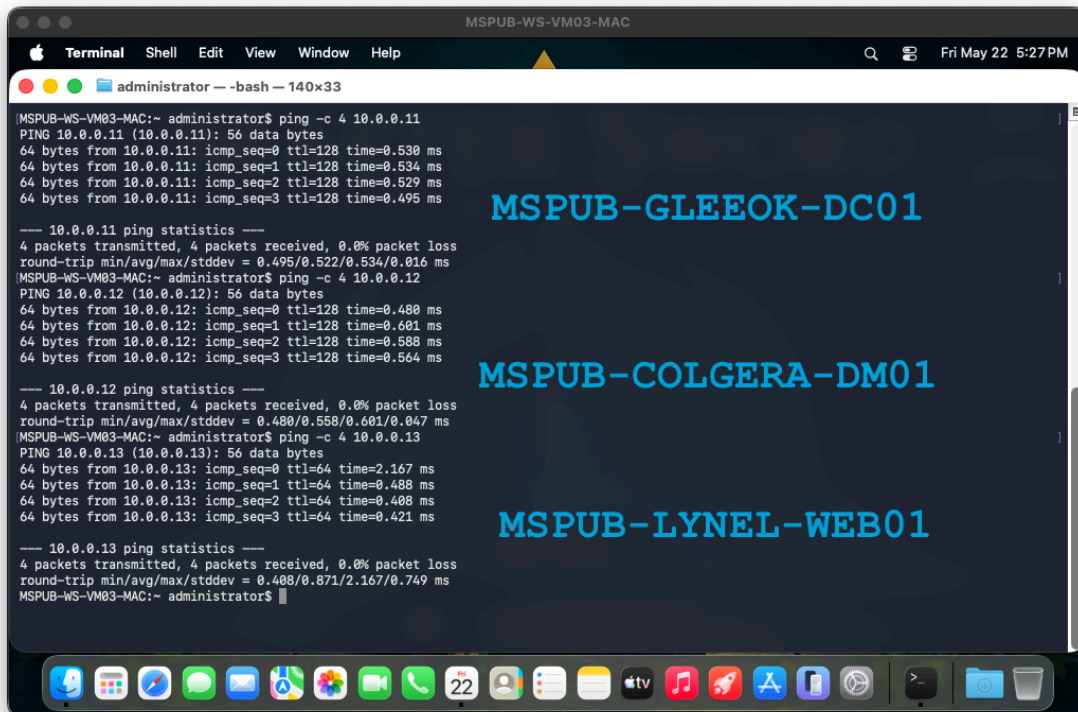
```
administrator@masterswordpub.net MSPUB-WS-VM02-UBT: ~
May 22 17:08
administrator@masterswordpub.net[MSPUB-WS-VM02-UBT]:-$ ping -c 4 10.0.0.101
PING 10.0.0.101 (10.0.0.101) 56(84) bytes of data:
64 bytes from 10.0.0.101: icmp_seq=1 ttl=128 time=0.918 ms
64 bytes from 10.0.0.101: icmp_seq=2 ttl=128 time=1.07 ms
64 bytes from 10.0.0.101: icmp_seq=3 ttl=128 time=0.948 ms
64 bytes from 10.0.0.101: icmp_seq=4 ttl=128 time=1.01 ms
--- 10.0.0.101 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3070ms
rtt min/avg/max/mdev = 0.918/0.985/1.067/0.057 ms
administrator@masterswordpub.net[MSPUB-WS-VM02-UBT]:-$ ping -c 4 10.0.0.103
PING 10.0.0.103 (10.0.0.103) 56(84) bytes of data:
64 bytes from 10.0.0.103: icmp_seq=1 ttl=64 time=0.445 ms
64 bytes from 10.0.0.103: icmp_seq=2 ttl=64 time=0.926 ms
64 bytes from 10.0.0.103: icmp_seq=3 ttl=64 time=0.425 ms
64 bytes from 10.0.0.103: icmp_seq=4 ttl=64 time=0.392 ms
--- 10.0.0.103 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3050ms
rtt min/avg/max/mdev = 0.392/0.547/0.926/0.219 ms
administrator@masterswordpub.net[MSPUB-WS-VM02-UBT]:-$ ping -c 4 10.0.0.104
PING 10.0.0.104 (10.0.0.104) 56(84) bytes of data:
64 bytes from 10.0.0.104: icmp_seq=1 ttl=64 time=1.42 ms
64 bytes from 10.0.0.104: icmp_seq=2 ttl=64 time=0.604 ms
64 bytes from 10.0.0.104: icmp_seq=3 ttl=64 time=0.614 ms
64 bytes from 10.0.0.104: icmp_seq=4 ttl=64 time=0.259 ms
--- 10.0.0.104 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3036ms
rtt min/avg/max/mdev = 0.259/0.724/1.421/0.426 ms
administrator@masterswordpub.net[MSPUB-WS-VM02-UBT]:-$ ping -c 4 10.0.0.11
PING 10.0.0.11 (10.0.0.11) 56(84) bytes of data:
64 bytes from 10.0.0.11: icmp_seq=1 ttl=128 time=0.734 ms
64 bytes from 10.0.0.11: icmp_seq=2 ttl=128 time=0.533 ms
64 bytes from 10.0.0.11: icmp_seq=3 ttl=128 time=0.431 ms
64 bytes from 10.0.0.11: icmp_seq=4 ttl=128 time=0.445 ms
--- 10.0.0.11 ping statistics ---
```

```
MSPUB-WS-VM02-UBT May 22 17:11
administrator@masterswordpub.net@MSPUB-WS-VM02-UBT:~$ ping -c 4 10.0.0.12
PING 10.0.0.12 (10.0.0.12) 56(84) bytes of data:
64 bytes from 10.0.0.12: icmp_seq=1 ttl=128 time=0.434 ms
64 bytes from 10.0.0.12: icmp_seq=2 ttl=128 time=1.47 ms
64 bytes from 10.0.0.12: icmp_seq=3 ttl=128 time=1.76 ms
64 bytes from 10.0.0.12: icmp_seq=4 ttl=128 time=0.895 ms
--- 10.0.0.12 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3010ms
rtt min/avg/max/mdev = 0.434/1.138/1.759/0.511 ms
administrator@masterswordpub.net@MSPUB-WS-VM02-UBT:~$ ping -c 4 10.0.0.13
PING 10.0.0.13 (10.0.0.13) 56(84) bytes of data:
64 bytes from 10.0.0.13: icmp_seq=1 ttl=64 time=0.845 ms
64 bytes from 10.0.0.13: icmp_seq=2 ttl=64 time=0.648 ms
64 bytes from 10.0.0.13: icmp_seq=3 ttl=64 time=0.460 ms
64 bytes from 10.0.0.13: icmp_seq=4 ttl=64 time=0.461 ms
--- 10.0.0.13 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3067ms
rtt min/avg/max/mdev = 0.460/0.603/0.845/0.159 ms
administrator@masterswordpub.net@MSPUB-WS-VM02-UBT:~$
```

MSPUB-WS-VM03-MAC

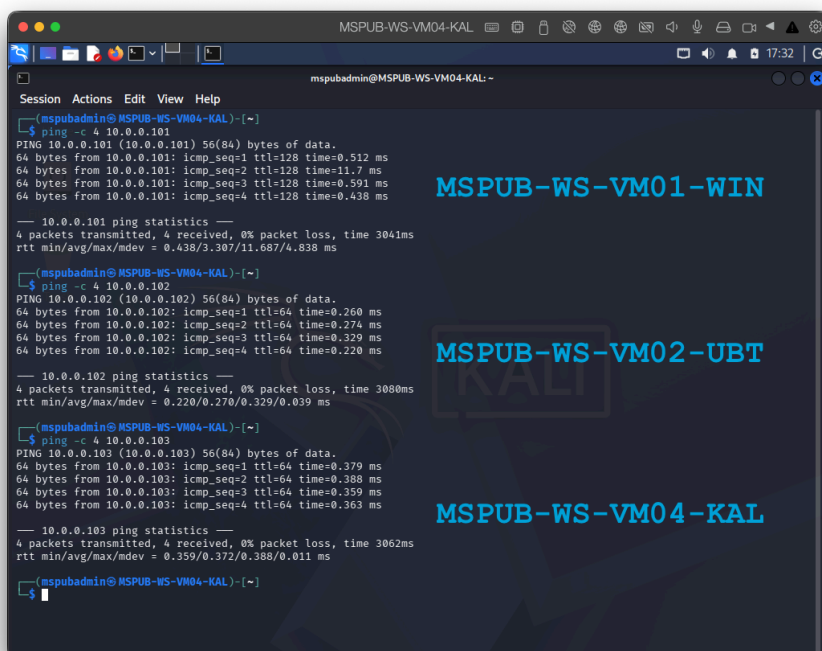
ping

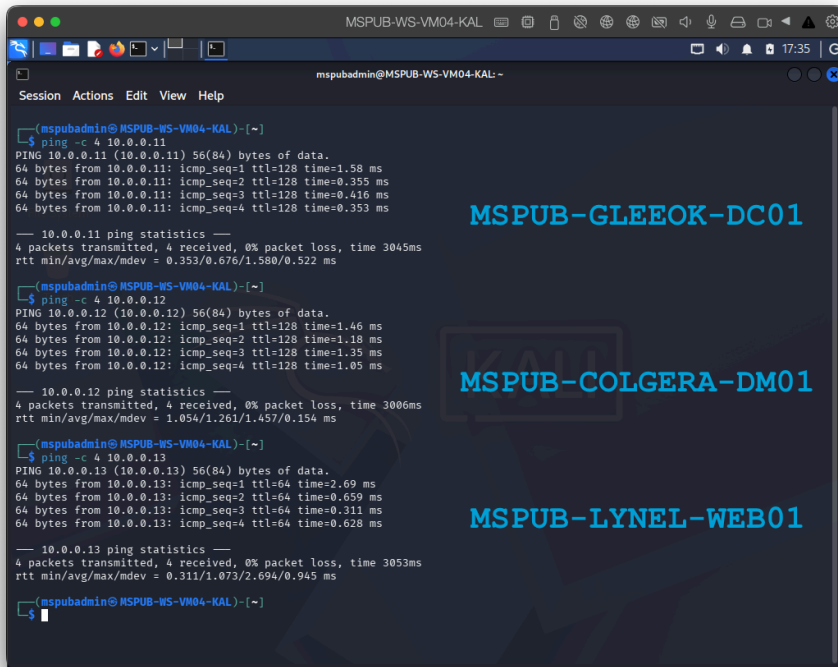
```
MSPUB-WS-VM03-MAC
Terminal Shell Edit View Window Help Fri May 22 5:21 PM
administrator ~ -bash — 140x33
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208850.
MSPUB-WS-VM03-MAC:~ administrator$ ping -c 4 10.0.0.101
PING 10.0.0.101 (10.0.0.101): 56 data bytes
64 bytes from 10.0.0.101: icmp_seq=0 ttl=128 time=0.866 ms
64 bytes from 10.0.0.101: icmp_seq=1 ttl=128 time=0.568 ms
64 bytes from 10.0.0.101: icmp_seq=2 ttl=128 time=0.829 ms
64 bytes from 10.0.0.101: icmp_seq=3 ttl=128 time=0.507 ms
--- 10.0.0.101 ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.507/0.693/0.866/0.157 ms
MSPUB-WS-VM03-MAC:~ administrator$ ping -c 4 10.0.0.102
PING 10.0.0.102 (10.0.0.102): 56 data bytes
64 bytes from 10.0.0.102: icmp_seq=0 ttl=64 time=0.593 ms
64 bytes from 10.0.0.102: icmp_seq=1 ttl=64 time=0.458 ms
64 bytes from 10.0.0.102: icmp_seq=2 ttl=64 time=0.434 ms
64 bytes from 10.0.0.102: icmp_seq=3 ttl=64 time=0.435 ms
--- 10.0.0.102 ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.434/0.480/0.593/0.066 ms
MSPUB-WS-VM03-MAC:~ administrator$ ping -c 4 10.0.0.104
PING 10.0.0.104 (10.0.0.104): 56 data bytes
64 bytes from 10.0.0.104: icmp_seq=0 ttl=64 time=0.784 ms
64 bytes from 10.0.0.104: icmp_seq=1 ttl=64 time=0.602 ms
64 bytes from 10.0.0.104: icmp_seq=2 ttl=64 time=0.395 ms
64 bytes from 10.0.0.104: icmp_seq=3 ttl=64 time=0.388 ms
--- 10.0.0.104 ping statistics ---
4 packets transmitted, 4 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.388/0.542/0.784/0.164 ms
MSPUB-WS-VM03-MAC:~ administrator$
```

MSPUB-WS-VM04-KAL

ping

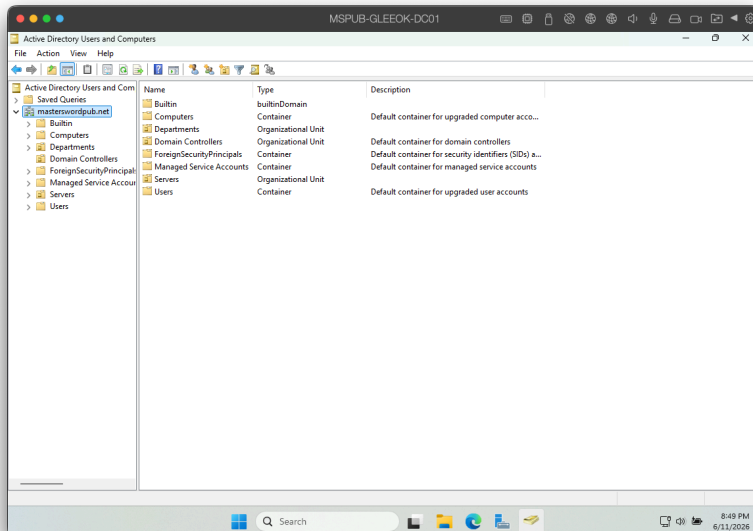




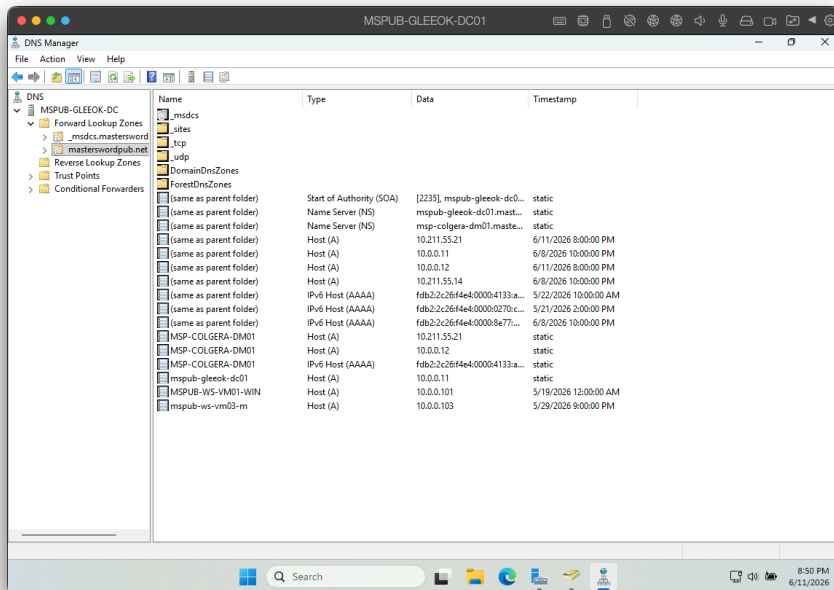
Appendix D.2 - Active Directory

MSPUB-GLLEEOK-DC01

Users and Computers

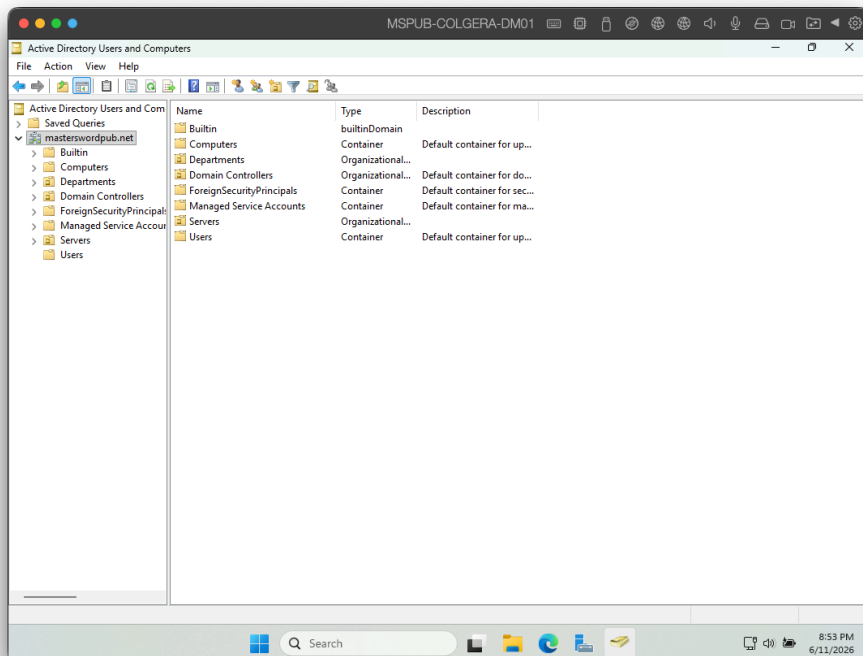


DNS

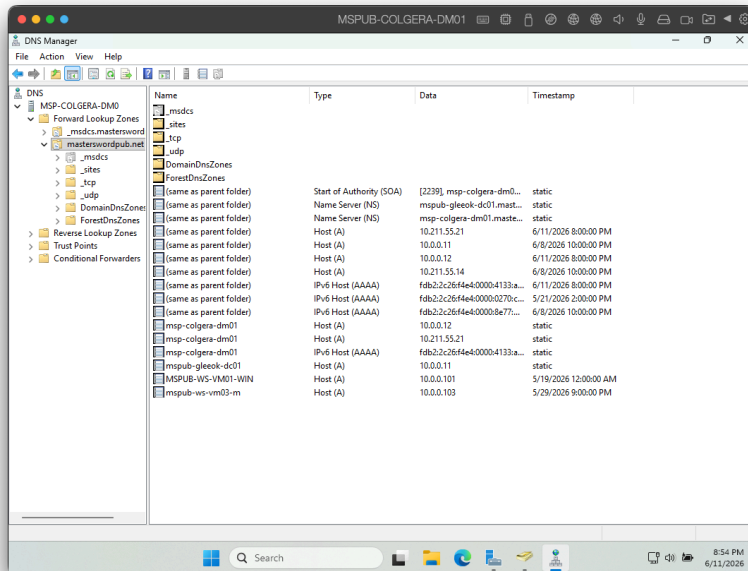


MSPUB-COLGERA-DM01

Users and Computers



DNS



Appendix D.3 - Linux Python Scripts

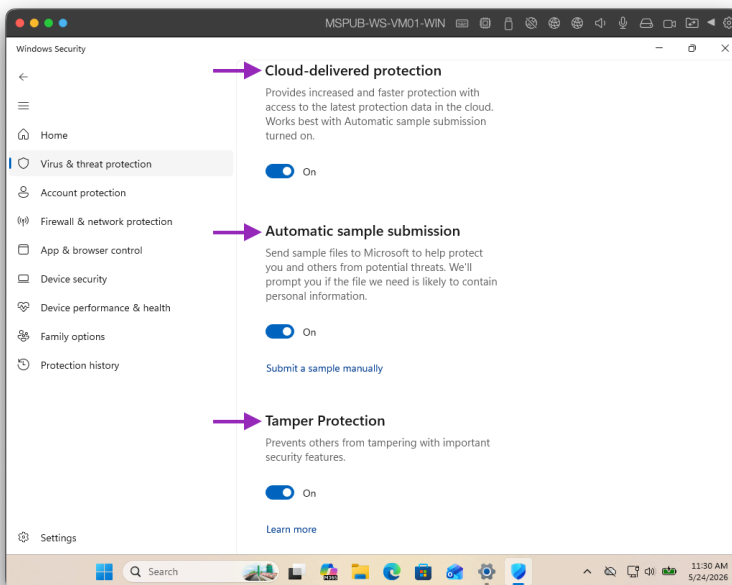
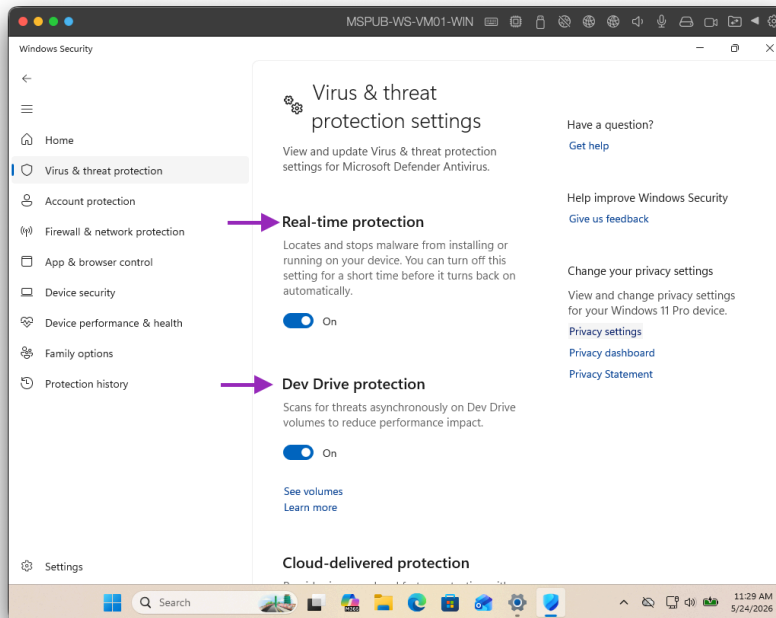
Automated Web Server System Configuration Script

User Management Script

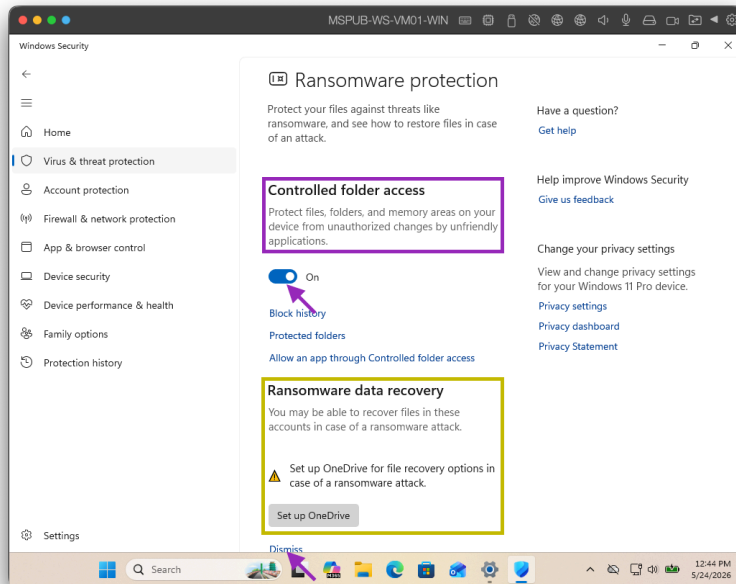
Appendix D.4 - Operating System Security Implementations

Windows 11

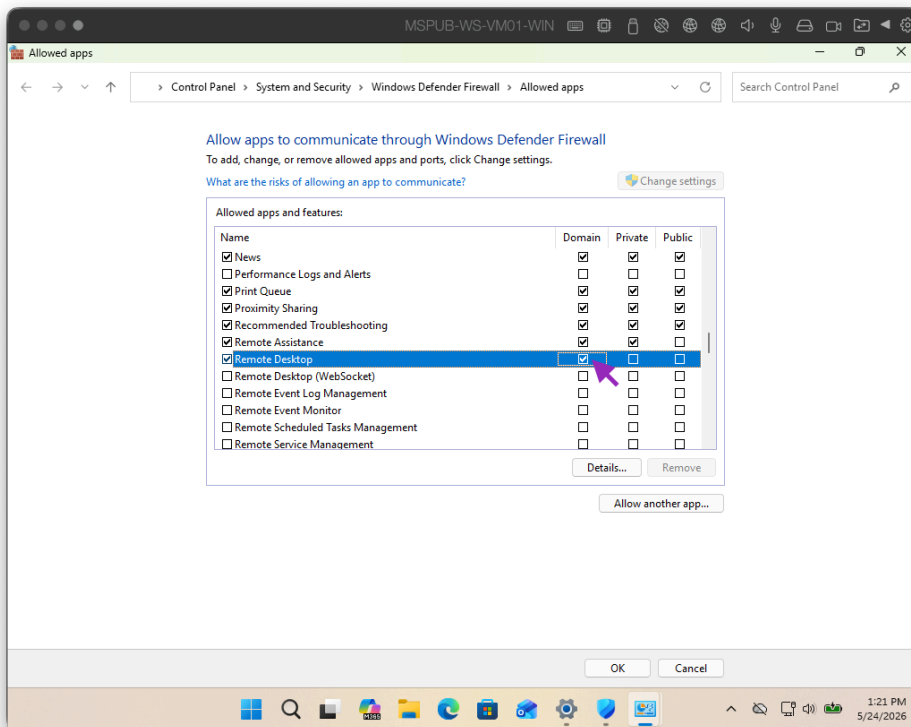
Virus and threat protection Settings



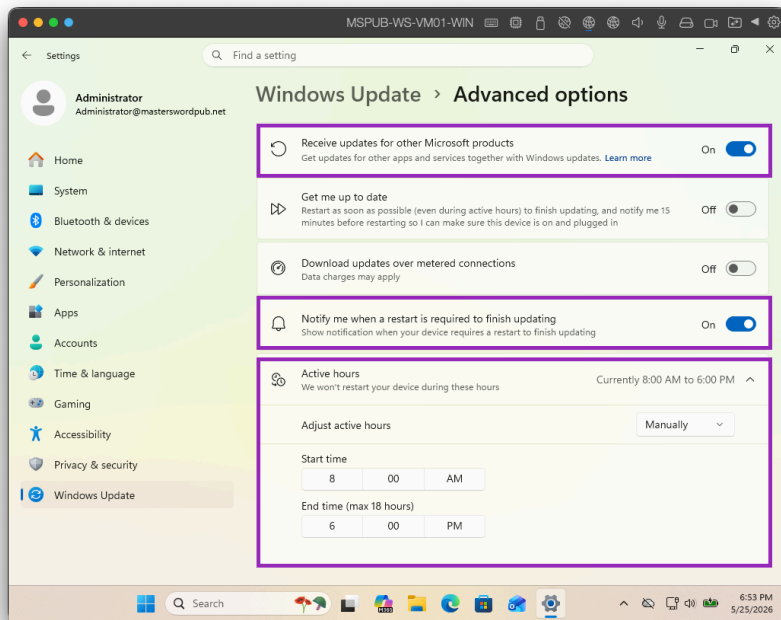
Ransomware protection



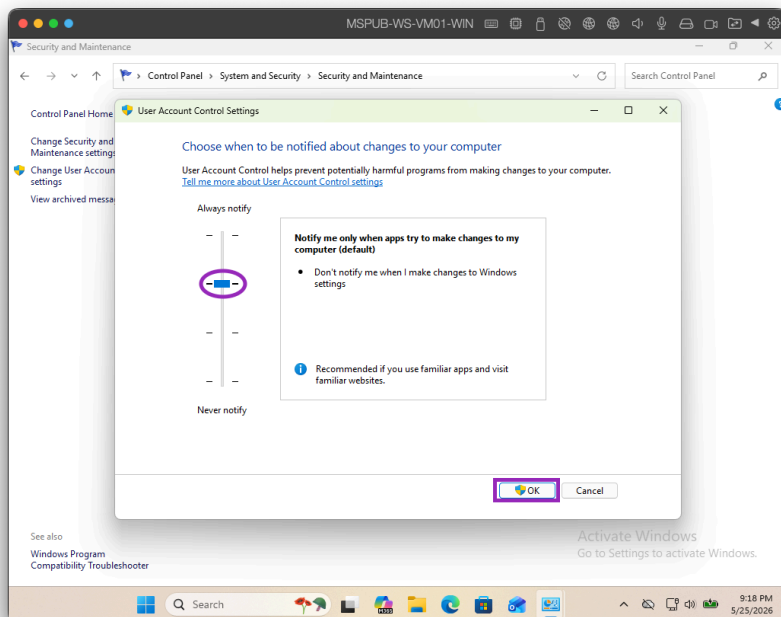
Firewall Configuration



Windows Updates

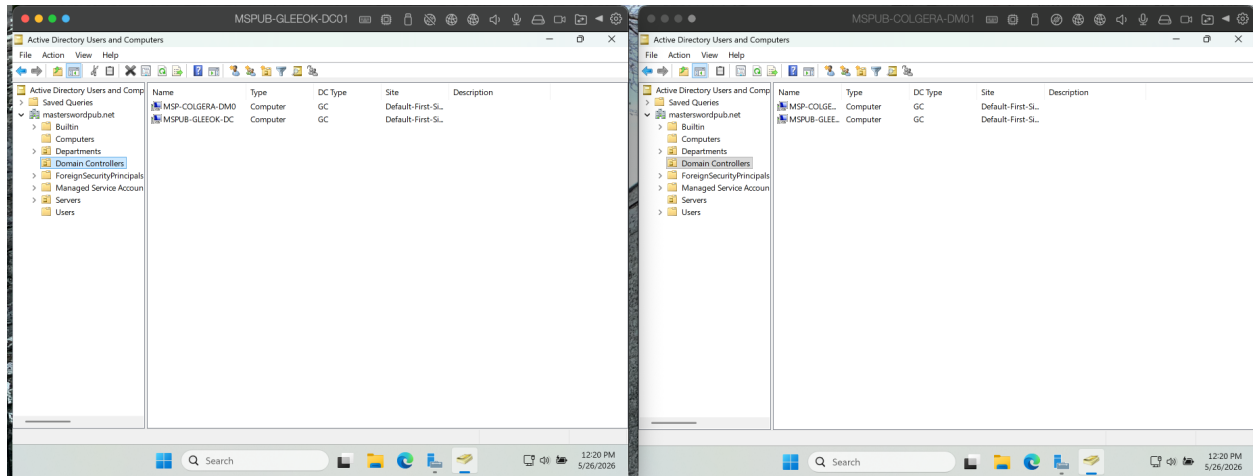


User Account Control (UAC)

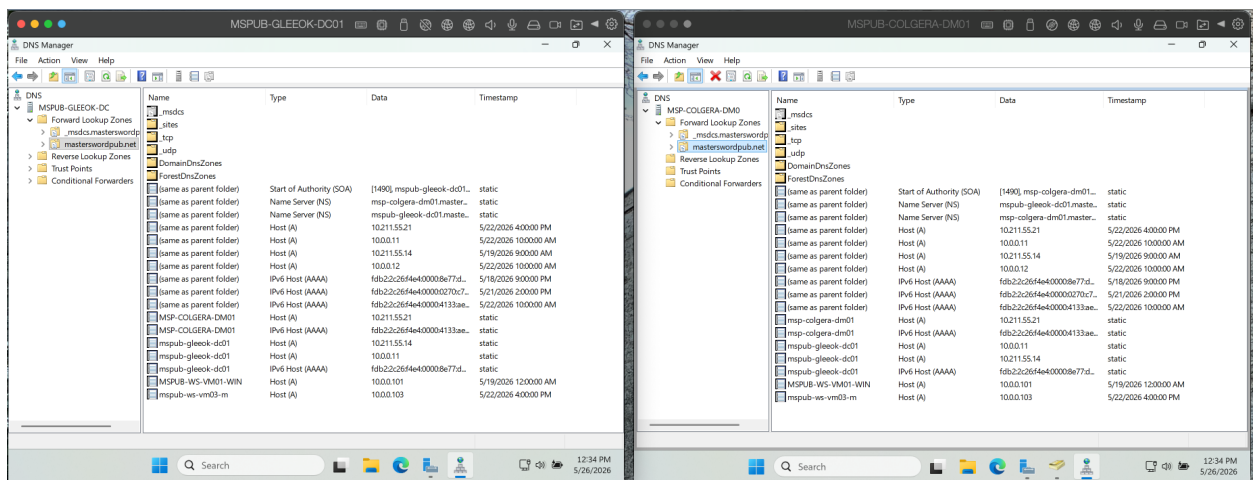


Window Server 2025

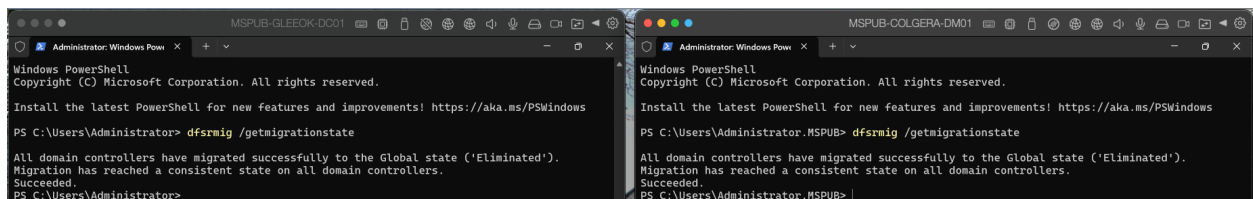
Active Directory Domain Services (AD DS) redundancy



DNS Redundancy



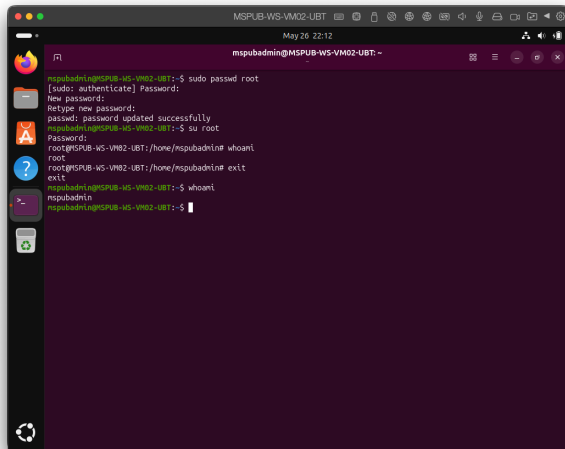
DFS Replication (DFSR)



Uncomplicated Firewall (UFW) configuration



Least Privilege Administration



A screenshot of a terminal window titled 'MSPUB-WS-VM02-UBT'. The user 'mspubadmin' is at the prompt. The following commands and outputs are shown:

```
mspubadmin@mSPUB-WS-VM02-UBT:~$ sudo passwd root
[sudo authenticating] Password:
New password:
Retype new password:
passwd: password updated successfully
mSPUB-WS-VM02-UBT:~$ su root
Password:
root@mSPUB-WS-VM02-UBT:/home/mspubadmin# whoami
root
root@mSPUB-WS-VM02-UBT:/home/mspubadmin# exit
mSPUB-WS-VM02-UBT:~$ whoami
mSPUB-WS-VM02-UBT:~$
```

```
(mspubadmin@ MSPUB-WS-VM04-KAL) - [~]
$ sudo passwd root
New password:
Retype new password:
passwd: password updated successfully

(mspubadmin@ MSPUB-WS-VM04-KAL) - [~]
$ su root
Password:
(root@ MSPUB-WS-VM04-KAL) - [/home/mspubadmin]
# whoami
root

(root@ MSPUB-WS-VM04-KAL) - [/home/mspubadmin]
# exit

(mspubadmin@ MSPUB-WS-VM04-KAL) - [~]
$
```

```
(mspubadmin@ MSPUB-WS-VM04-KAL) - [~]
$ sudo nano /etc/ssh/sshd_config

(mspubadmin@ MSPUB-WS-VM04-KAL) - [~]
$ sudo systemctl restart ssh
```

System Updates & Package Management



```
MSPUB-WS-VM04-KAL
May 27 12:49

MASTER SWORD

Software Updater

Updated software has been issued since Ubuntu 26.04 was released. Do you want to install it now?

Details of updates
Install or remove
[ ] Security updates 18.5 MB
[ ] Document Viewer (4) 2.4 MB
[ ] Common files used by both the Samba server and client 46 kB
[ ] Hierarchical pool based memory allocator - Python3 bin... 15 kB
[ ] Python 3 bindings for LDB 42 kB

Technical description
Package: samba-common
Samba is an implementation of the SMB/CIFS protocol for Unix systems, providing
58.5 MB will be downloaded.

Remind Me Later Install Now

Home
```

```
MSPUB-WS-VM04-KAL
mspubadmin@MSPUB-WS-VM04-KAL: ~
$ sudo apt update
[sudo] password for mspubadmin:
Get:1 http://kali.download/kali kali-rolling InRelease [34.0 kB]
Get:2 http://kali.download/kali kali-rolling/main arm64 Packages [20.8 MB]
Get:3 http://kali.download/kali kali-rolling/main arm64 Contents (deb) [49.8 MB]
Get:4 http://kali.download/kali kali-rolling/contrib arm64 Packages [104 kB]
Get:5 http://kali.download/kali kali-rolling/contrib arm64 Contents (deb) [190 kB]
Get:6 http://kali.download/kali kali-rolling/non-free arm64 Packages [158 kB]
Get:7 http://kali.download/kali kali-rolling/non-free arm64 Contents (deb) [879 kB]
Get:8 http://kali.download/kali kali-rolling/non-free-firmware arm64 Packages [14.5 kB]
Fetched 71.9 MB in 4s (17.3 MB/s)
172 packages can be upgraded. Run 'apt list --upgradable' to see them.

mspubadmin@MSPUB-WS-VM04-KAL: ~
$ sudo apt upgrade
The following packages were automatically installed and are no longer required:
libavc1394-0 libbencodecrypto16 libbsd2-classic libunibreak openjdk-21-jre-headless python3-tz
libicu76 libpostal-data libsimdutf31 libxmlsec1-1 postgresql-common-dev python3-pluginbase
libio-pty-perl libpostal1 libsodium23 libxmlsec1-openssl1 python3-pluginbase
libipc-run-perl libraw23t64 libteamdctl0 openjdk-21-jre python3-pysnmp4
Use 'sudo apt autoremove' to remove them.

Upgrading:
bubblewrap kali-linux-default libgnutls-dane0t64 mesa-libgallium
clang-21 kali-linux-firmware libgnutls30t64 mesa-vulkan-drivers
clang-tools-21 kali-linux-headless libgrpc++1.51t64 metasploit-framework
exploitdb kali-system-cli libgststreamer-glib0-0 mltproxy
faraday kali-system-core libgststreamer-plugins-base1.0-0 php8.4
firebird4.0-common kali-system-gui libgststreamer1.0-0 php8.4-cli
firebird4.0-common-doc kali-tools-top10 libgtkmm-3.0-1t64 php8.4-common
girl1.2-girepository-3.0 legion libimage-exiftool-perl php8.4-mysql
girl1.2-glib-2.0 libapache2-mod-php8.4 libjs-sphinxdoc php8.4-opcache
girl1.2-gstreamer-1.0 libavcodec62 liblvm21 php8.4-readline
gnutls-bin libavfilter11 libmagickcore-7.q16-10 ppp
groff-base libavformat62 libmagickwand-7.q16-10 python3
gststreamer1.0-gi libavutil60 libmbedcrypto16 python3-charset-normalizer
gststreamer1.0-libav libcfitsio10t64 libmongocrypt0 python3-dev
gststreamer1.0-plugins-base libcgi-pm-perl libnss3 python3-elastic-transport
gststreamer1.0-plugins-good libclang-common-21-dev libopenjp2-7 python3-git
gststreamer1.0-x libclang-cpp21 libpython3-dev python3-minimal
```

Ubuntu Server 26.04

UFW Firewall Hardening

```
msspubadmin@MSPUB-LYNEL-WEB01:~$ sudo ufw enable
[sudo: authenticate] Password:
Firewall is active and enabled on system startup
msspubadmin@MSPUB-LYNEL-WEB01:~$ sudo ufw allow ssh
Rule added
Rule added (v6)
msspubadmin@MSPUB-LYNEL-WEB01:~$ sudo ufw status
Status: active

To Action From
--
22/tcp ALLOW Anywhere
22/tcp (v6) ALLOW Anywhere (v6)

msspubadmin@MSPUB-LYNEL-WEB01:~$ _
```

Least Privileged Access (Groups)/Root Password Change

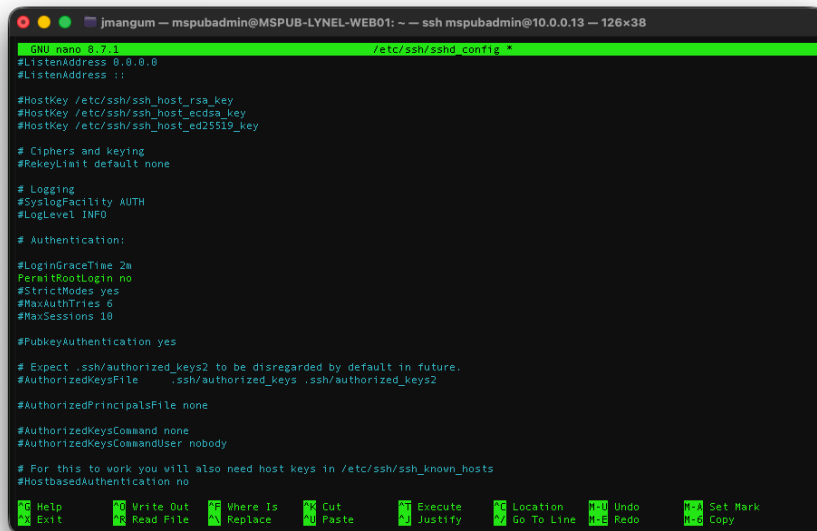
```
root@MSPUB-LYNEL-WEB01:/home/msspubadmin# sudo adduser msspubtestuser
New password:
Retype new password:
passwd: password updated successfully
Changing the user information for msspubtestuser
Enter the new value, or press ENTER for the default
Full Name []: MSPUB TestUser
Room Number []:
Work Phone []:
Home Phone []:
Other []:
Is the information correct? [Y/n] Y
root@MSPUB-LYNEL-WEB01:/home/msspubadmin# sudo usermod -aG sudo msspubtestuser
root@MSPUB-LYNEL-WEB01:/home/msspubadmin# groups msspubtestuser
msspubtestuser : msspubtestuser sudo users
root@MSPUB-LYNEL-WEB01:/home/msspubadmin# sudo deluser --remove-home msspubtestuser
root@MSPUB-LYNEL-WEB01:/home/msspubadmin# _
```

```

mspubadmin@MSPUB-LYNEL-WEB01:~$ sudo passwd root
New password:
Retype new password:
passwd: password updated successfully
mspubadmin@MSPUB-LYNEL-WEB01:~$ su root
Password:
root@MSPUB-LYNEL-WEB01:/home/mspubadmin# whoami
root
root@MSPUB-LYNEL-WEB01:/home/mspubadmin# _

```

OpenSSH Administration



The screenshot shows a terminal window titled "jmanungum — mspubadmin@MSPUB-LYNEL-WEB01: ~ — ssh mspubadmin@10.0.0.13 — 126x38". The terminal is running the nano editor on the file /etc/ssh/sshd_config. The file content is as follows:

```

GNU nano 0.7.1 /etc/ssh/sshd_config
#ListenAddress 0.0.0.0
#ListenAddress ::

#HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh/ssh_host_ecdsa_key
#HostKey /etc/ssh/ssh_host_ed25519_key

# Ciphers and keying
#RekeyLimit default none

# Logging
#SyslogFacility AUTH
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin no
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#PubkeyAuthentication yes

# Expect .ssh/authorized_keys2 to be disregarded by default in future.
#AuthorizedKeysFile .ssh/authorized_keys .ssh/authorized_keys2

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also need host keys in /etc/ssh/ssh_known_hosts
#HostbasedAuthentication no

```

The bottom of the terminal shows the nano editor's command palette with options: Help, Exit, Write Out, Read File, Where Is, Replace, Cut, Paste, Execute, Justify, Location, Go To Line, Undo, Redo, Set Mark, and Copy.

```

[jmanungum@DINRAAL ~ % ssh root@10.0.0.13
[root@10.0.0.13's password:
Permission denied, please try again.
root@10.0.0.13's password:

```

System Updates & Package Management

```
mjiangum — mspubadmin@MSPUB-LYNEL-WEB01: ~ — ssh mspubadmin@10.0.0.13 — 169x97

mspubadmin@MSPUB-LYNEL-WEB01:~$ sudo apt update
Ign1: http://archive.ubuntu.com/ubuntu resolute InRelease
Ign2: http://archive.ubuntu.com/ubuntu resolute-updates InRelease
Ign3: http://archive.ubuntu.com/ubuntu resolute-backports InRelease
Ign4: http://security.ubuntu.com/ubuntu resolute-security InRelease
Ign3: http://archive.ubuntu.com/ubuntu resolute-backports InRelease
Ign2: http://archive.ubuntu.com/ubuntu resolute-updates InRelease
Ign1: http://archive.ubuntu.com/ubuntu resolute InRelease
Ign4: http://security.ubuntu.com/ubuntu resolute-security InRelease
Ign1: http://archive.ubuntu.com/ubuntu resolute InRelease
Ign2: http://archive.ubuntu.com/ubuntu resolute-updates InRelease
Ign3: http://archive.ubuntu.com/ubuntu resolute-backports InRelease
Ign4: http://security.ubuntu.com/ubuntu resolute-security InRelease
Err1: http://archive.ubuntu.com/ubuntu resolute-backports InRelease
  Cannot initiate the connection to archive.ubuntu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable) Cannot initiate the connection to archive.ub
untu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable)
Err2: http://archive.ubuntu.com/ubuntu resolute-updates InRelease
  Cannot initiate the connection to archive.ubuntu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable) Cannot initiate the connection to archive.ub
untu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable)
Err1: http://archive.ubuntu.com/ubuntu resolute InRelease
  Cannot initiate the connection to archive.ubuntu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable) Cannot initiate the connection to archive.ub
untu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable) Could not connect to archive.ubuntu.com:80 (172.66.152.176), connection timed out Could no
t connect to archive.ubuntu.com:80 (184.28.28.246), connection timed out
  Cannot initiate the connection to archive.ubuntu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable) Cannot initiate the connection to archive.ub
untu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable)
Err4: http://security.ubuntu.com/ubuntu resolute-security InRelease
  Cannot initiate the connection to security.ubuntu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable) Cannot initiate the connection to security.
ubuntu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable) Could not connect to security.ubuntu.com:80 (184.28.28.246), connection timed out Could
not connect to security.ubuntu.com:80 (172.66.152.176), connection timed out
  Cannot initiate the connection to security.ubuntu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable) Cannot initiate the connection to security.
ubuntu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable)
7 packages can be upgraded. Run 'apt list --upgradable' to see them.
Warning: Failed to fetch http://archive.ubuntu.com/ubuntu/dists/resolute/InRelease Cannot initiate the connection to archive.ubuntu.com:80 (2686:4788:10::6814:1cf6). -
connect (101: Network is unreachable) Cannot initiate the connection to archive.ubuntu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable)
Warning: Failed to fetch http://archive.ubuntu.com/ubuntu/dists/resolute-updates/InRelease Cannot initiate the connection to archive.ubuntu.com:80 (2686:4788:10::6814:1
cf6). - connect (101: Network is unreachable) Cannot initiate the connection to archive.ubuntu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable)
Warning: Failed to fetch http://archive.ubuntu.com/ubuntu/dists/resolute-backports/InRelease Cannot initiate the connection to archive.ubuntu.com:80 (2686:4788:10::6814
:1cf6). - connect (101: Network is unreachable) Cannot initiate the connection to archive.ubuntu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachable)
Warning: Failed to fetch http://security.ubuntu.com/ubuntu/dists/resolute-security/InRelease Cannot initiate the connection to security.ubuntu.com:80 (2686:4788:10::681
4:1cf6). - connect (101: Network is unreachable) Cannot initiate the connection to security.ubuntu.com:80 (2686:4788:10::6814:1cf6). - connect (101: Network is unreachab
le)
Warning: Some index files failed to download. They have been ignored, or old ones used instead.
mspubadmin@MSPUB-LYNEL-WEB01:~$ sudo apt upgrade -y
Upgrading:
  libcrypt20 libssh2-1t64 vim vim-common vim-runtime viatiny xxd

Summary:
  Upgrading: 7, Installing: 0, Removing: 0, Not Upgrading: 0
  7 standard LTS security updates
  Download size: 0 B / 11.6 MB
  Freed space: 68.6 kB

(Reading database, 106789 files and directories currently installed.)
Preparing to unpack ./libcrypt20_1:12.0-2ubuntu0.1_arm64.deb...
Unpacking libcrypt20:arm64 (1:12.0-2ubuntu0.1) over (1:12.0-2)...
Setting up libcrypt20:arm64 (1:12.0-2ubuntu0.1)...
(Reading database, 106789 files and directories currently installed.)
Preparing to unpack ./0-vim_2k3a9.1.2141-1ubuntu4.2_arm64.deb...
Unpacking vim (2:9.1.2141-1ubuntu4.2) over (2:9.1.2141-1ubuntu4.1)...
Preparing to unpack ./1-vim-common_2k3a9.1.2141-1ubuntu4.2_all.deb...
Unpacking vim-common (2:9.1.2141-1ubuntu4.2) over (2:9.1.2141-1ubuntu4.1)...
Preparing to unpack ./2-vim-tiny_2k3a9.1.2141-1ubuntu4.2_arm64.deb...
Unpacking vim-tiny (2:9.1.2141-1ubuntu4.2) over (2:9.1.2141-1ubuntu4.1)...
Preparing to unpack ./3-vim-runtime_2k3a9.1.2141-1ubuntu4.2_all.deb...
Unpacking vim-runtime (2:9.1.2141-1ubuntu4.2) over (2:9.1.2141-1ubuntu4.1)...
Preparing to unpack ./4-xxd_2k3a9.1.2141-1ubuntu4.2_arm64.deb...
Unpacking xxd (2:9.1.2141-1ubuntu4.2) over (2:9.1.2141-1ubuntu4.1)...
Preparing to unpack ./5-libssh2-1t64_1:11.1-1ubuntu0.26.04.1_arm64.deb...
Unpacking libssh2-1t64:arm64 (1:11.1-1ubuntu0.26.04.1) over (1:11.1-1build12)...
Setting up xxd (2:9.1.2141-1ubuntu4.2)...
Setting up vim-common (2:9.1.2141-1ubuntu4.2)...
Setting up libssh2-1t64:arm64 (1:11.1-1ubuntu0.26.04.1)...
Setting up vim-runtime (2:9.1.2141-1ubuntu4.2)...
Setting up vim (2:9.1.2141-1ubuntu4.2)...
Processing triggers for libc-bin (2.40-2ubuntu2)...
Processing triggers for man-db (2.13.1-1build1)...
Scanning processes...

Scanning candidates...

Scanning linux images...

Running kernel seems to be up-to-date (ABI upgrades are not detected).

Restarting services...
systemctl restart fwupd.service

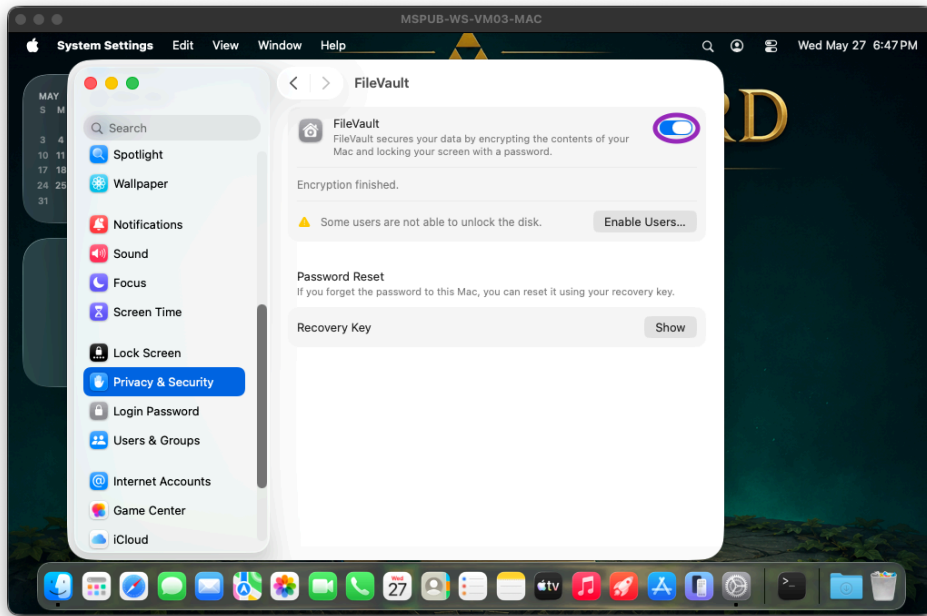
No containers need to be restarted.

No user sessions are running outdated binaries.

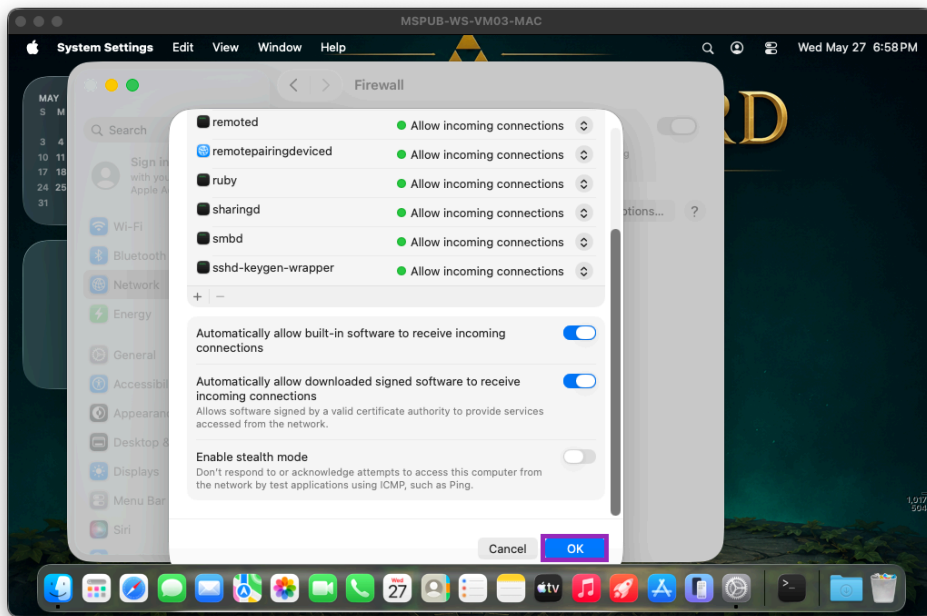
No VM guests are running outdated hypervisor (qemu) binaries on this host.
mspubadmin@MSPUB-LYNEL-WEB01:~$
```

macOS 26 Tahoe

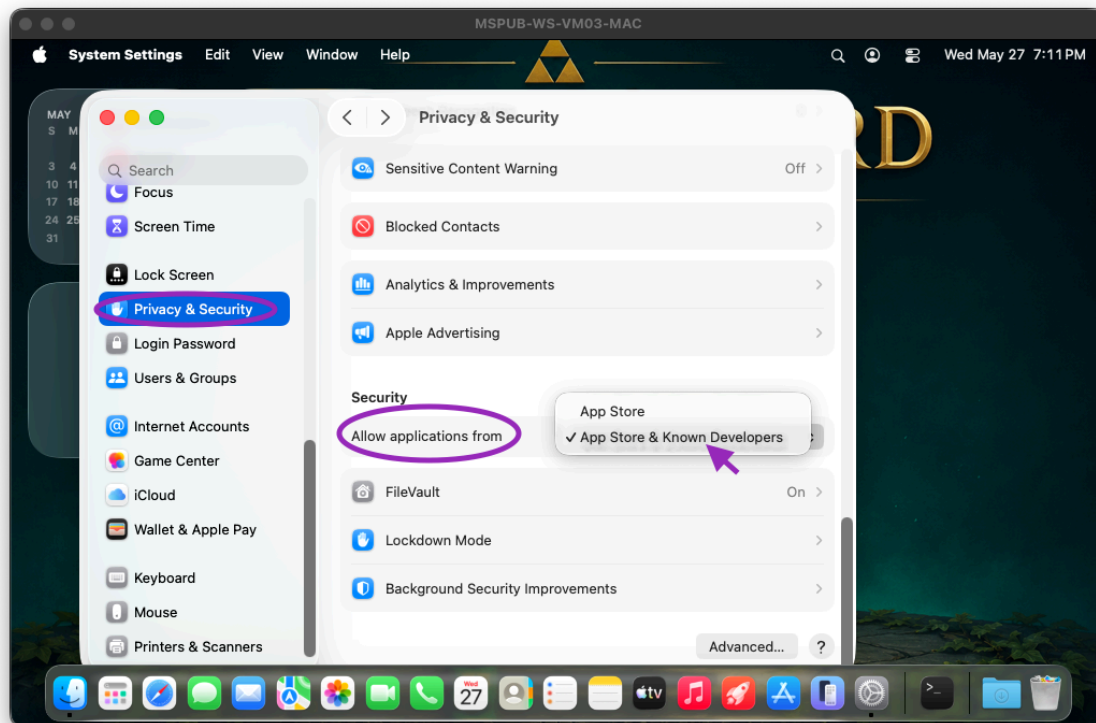
FileVault Disk Encryption



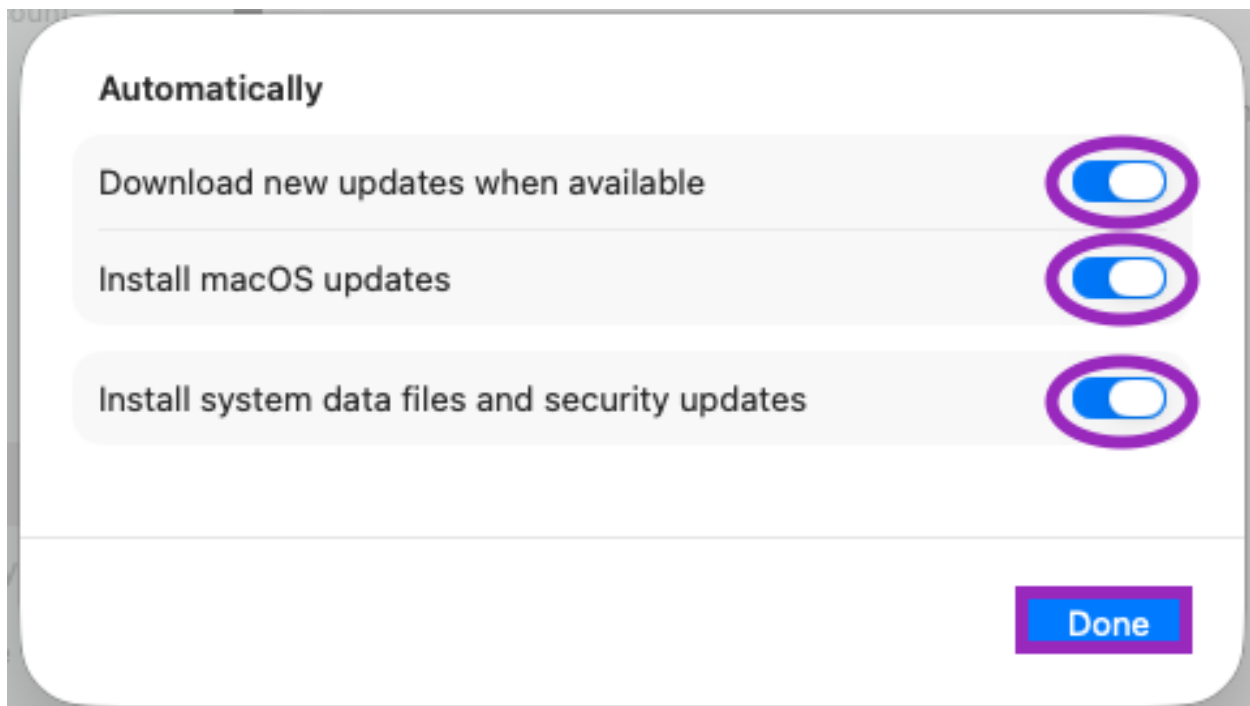
macOS Firewall



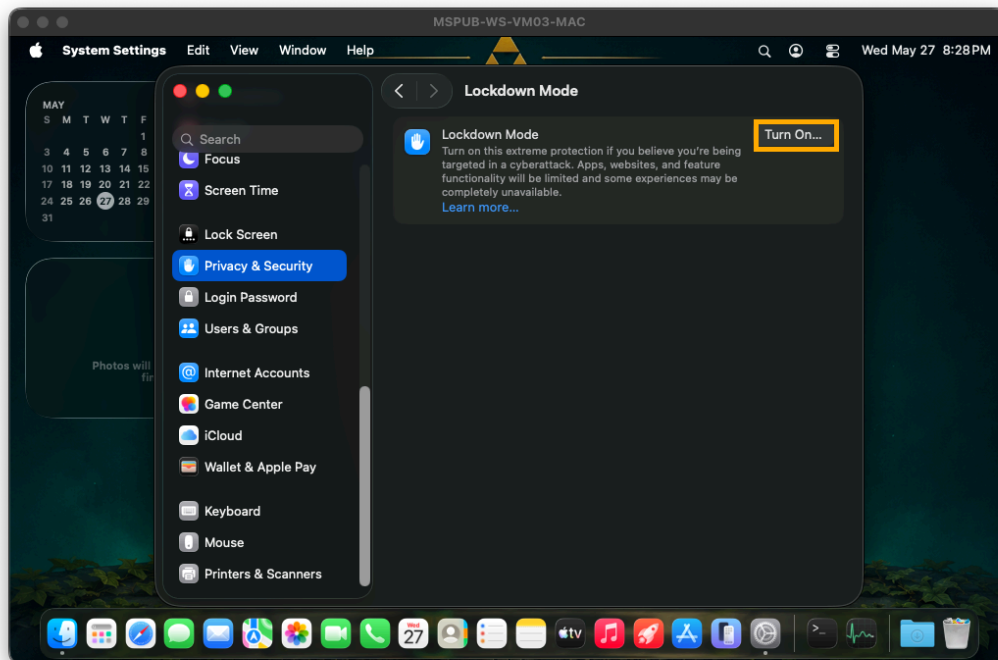
Gatekeeper Application Security



Automatic System Updates

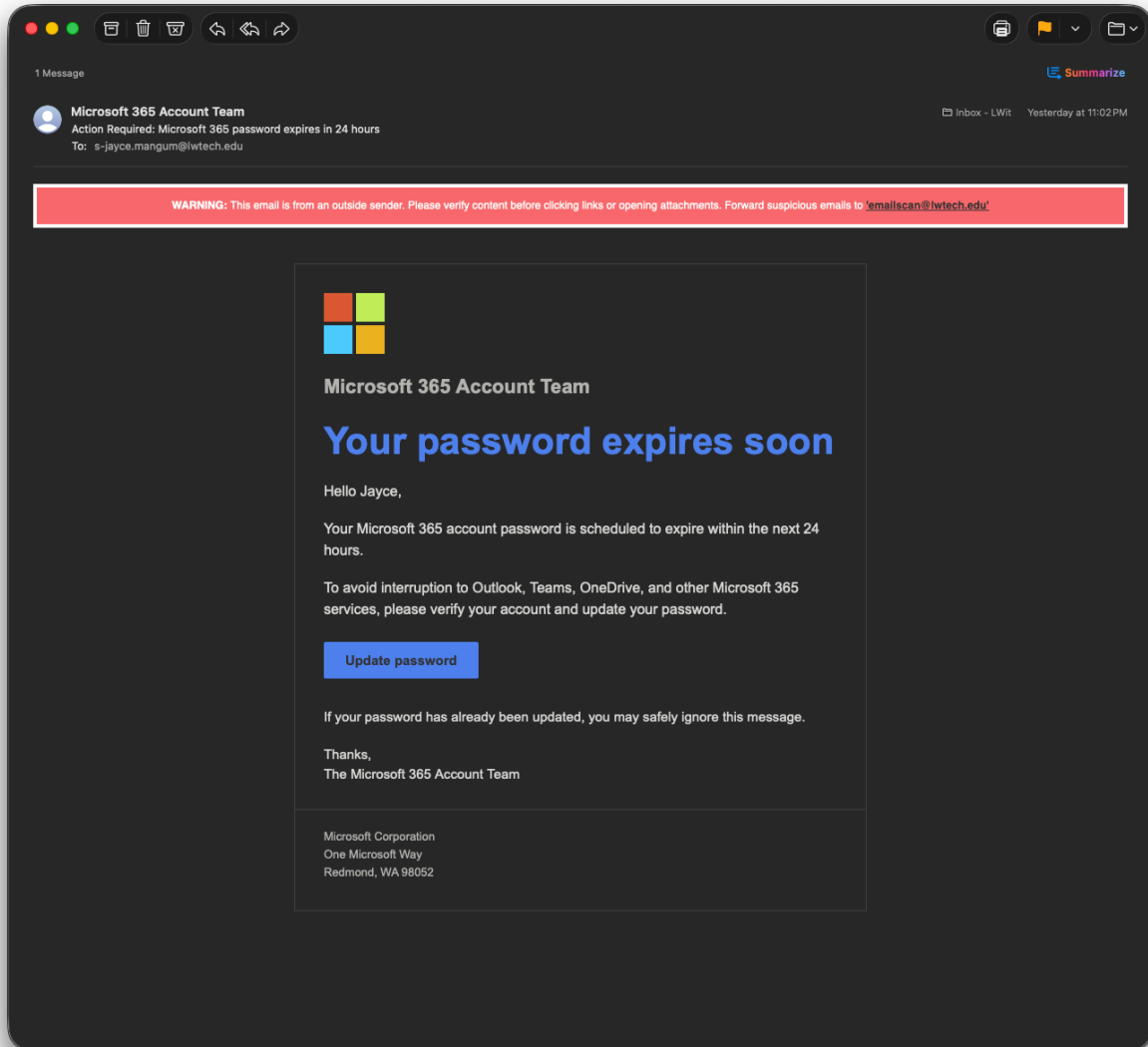


Lockdown Mode



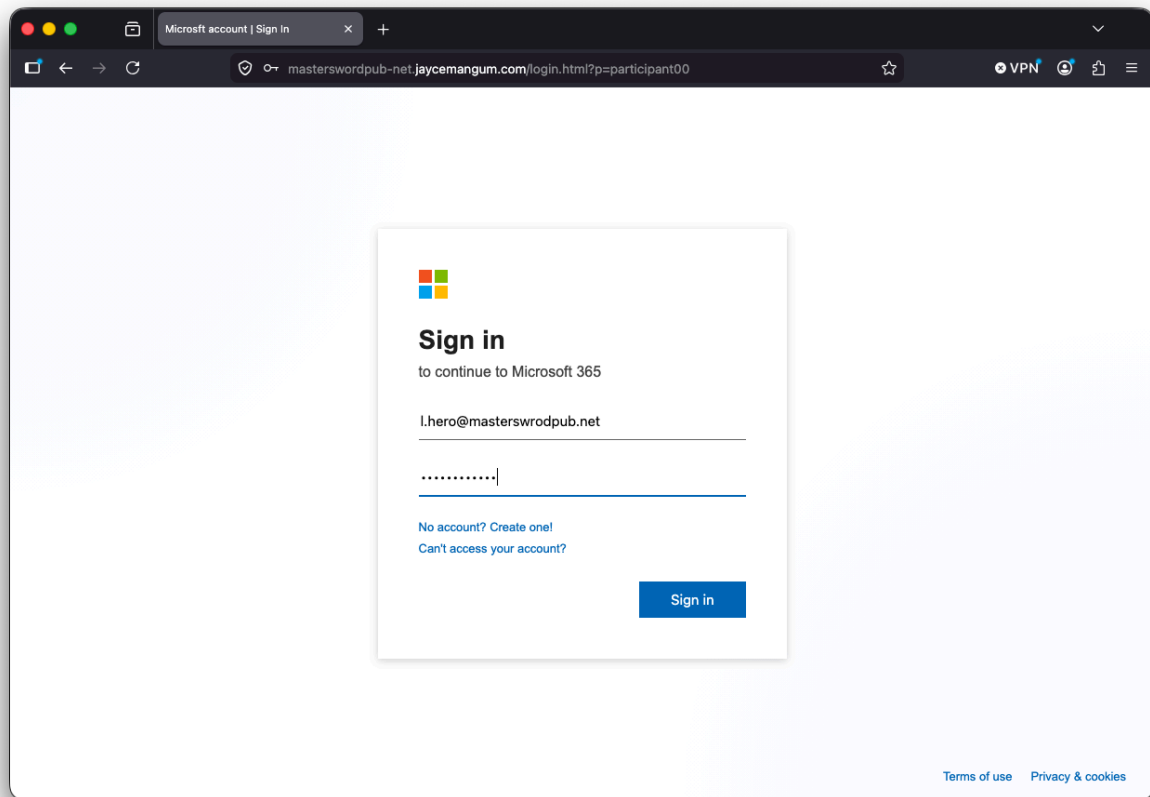
Appendix D.5 Security Awareness Training (SAT) Platform

Microsoft 365 Phishing Email



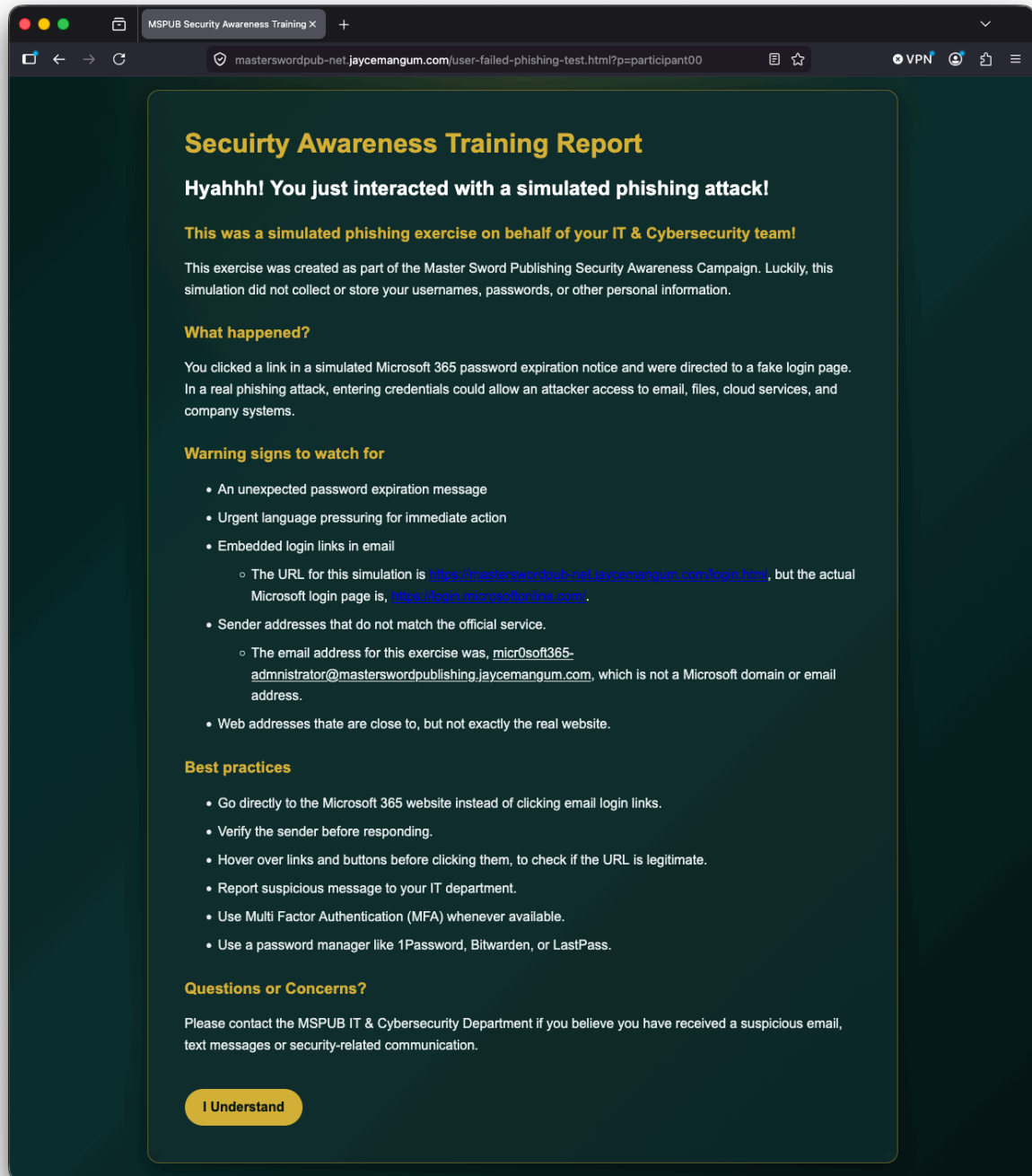
Malicious Microsoft 365 Sign Page

<https://masterswordpub-net.jaycemangum.com/login.html?p=CAPSTONE-INQUIRY>



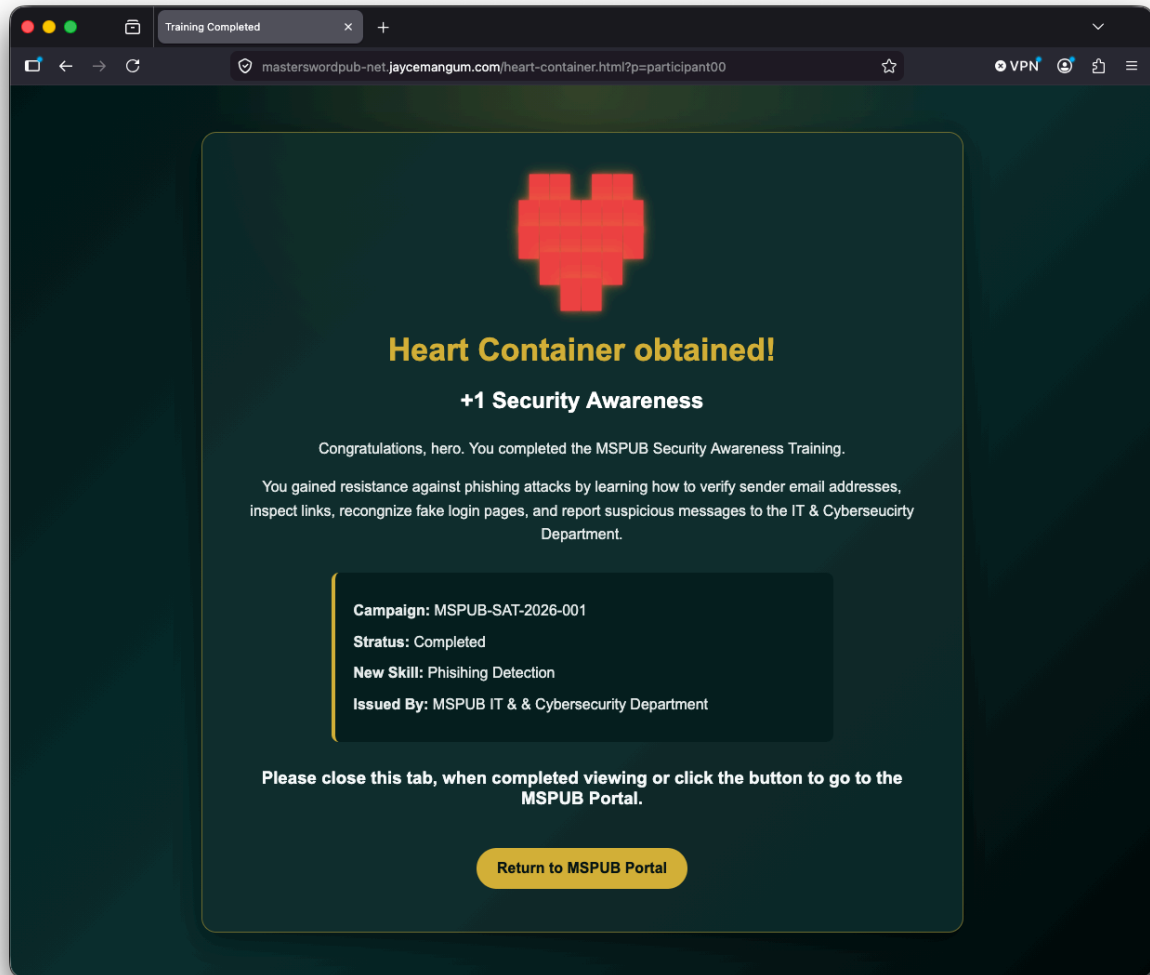
Security Awareness Training Report

[https://masterswordpub-net.jaycemangum.com/user-failed-phishing-test.html?](https://masterswordpub-net.jaycemangum.com/user-failed-phishing-test.html?p=CAPSTONE-INQUIRY)
[p=CAPSTONE-INQUIRY](#)



Security Awareness Completion Page

<https://masterswordpub-net.jaycemangum.com/heart-container.html?p=CAPSTONE-INQUIRY>



Appendix E - Incident Response Plan

Master Sword Publishing Incident Response Plan

Executive Summary

Master Sword Publishing (MSPUB) will maintain an Incident Response Plan to identify, contain, eradicate, and recover from cybersecurity incidents. This plan provides a structured approach for responding to security events while minimizing business disruption and protecting company systems, data, and users.

Purpose

This document outlines the plan for responding to cybersecurity incidents at Master Sword Publishing (MSPUB). It defines the roles and responsibilities of the participants involved in the incident response. It also characterizes the overall incident response process and establishes relationships with other relevant policies and procedures. Additionally, it provides guidelines for reporting requirements.

The purpose of this Incident Response Plan is to:

- Detect cybersecurity incidents quickly
- Define the roles associated with the individuals involved in an incident
- Limit damage to company systems
- Protect sensitive information
- Restore normal business operations
- Document lessons learned
- Meet organizational security requirements

Standards Alignment

The Master Sword Publications (MSPUB) Incident Response Plan is aligned with the principles of ISO/IEC 27001 Information Security Management Systems (ISMS) and ISO/IEC 27035 Information Security Incident Management. These standards provide guidance for establishing, implementing, maintaining, and continually improving information security processes and incident response capabilities.

This plan incorporated industry best practices for incident preparation, detection, reporting, assessment, containment, eradication, recovery, post-incident review, and continuous improvement. The procedures outlined in this document are intended to support MSPUB's commitment to protecting the confidentiality, integrity, and availability of organization information assets while maintaining business continuity during security incidents.

Scope

This plan applies to all information systems, company data, and networks of MSPUB, as well as any person or device that accesses system data. This plan applies to the following:

- Windows Server Domain Controllers
- Ubuntu Linux Web Server
- Windows Workstations

- macOS Workstations
- Ubuntu Linux Workstations
- Kali Linux Security Workstations
- Microsoft 365 Services
- Google Workspace Services
- Apple iCloud Services
- Network Infrastructure
- Company Web Applications

Incident Response Team

In the event of a cybersecurity incident, these are the assigned roles of the individuals apart of the Incident Response Team. These roles should be reviewed and updated on a quarterly basis.

Role	Responsibility	Point of Contact
IT Administrator	Incident coordinator	Purah Sheikah p.sheikah@masterswordpub.net
Security Analyst	Investigation and analysis	Link Hero l.hero@masterswordpub.net
System Administrator	Server remediation	Robbie Sheikah r.sheikah@masterswordpub.net
Network Administrator	Network containment	Jayce Mangum j.mangum@masterswordpub.net
Management	Business decisions and communications	Impa Sheikah i.sheikah@masterswordpub.net

Incident Response Lifecycle

This section defines the lifecycle of an Incident.

Phase 1: Preparation

This phase prepares employees, critical systems, and data prior to a cybersecurity incident and defines what activities and actions are to be taken.

- **Security Awareness Training:** Master Sword Publishing employees will be required to participate in security awareness trainings on a monthly basis. These trainings may include users watching security awareness videos, taking a short quiz that tests their knowledge on various cyber attacks, and participating in controlled simulations developed by the Master Sword Publishing IT & Cybersecurity department.
- **Multi-Factor Authentication (MFA) implementation:** All Master Sword Publishing employees will be required to protect their company user accounts using MFA protection with a company-approved authentication app. They will be required to enter a verification code before accessing company resources.

- **Regular System Backups:** Backups of critical systems will be backed up using the 3-2-1 backup rule. Three total copies of all company data will be stored on 2 different media types, and one copy kept off-premise. A full backup will be run monthly of critical systems and stored onsite. An incremental backup will be run biweekly and stored through a cloud service provider as well as onsite. Differential Backups will be run weekly and stored through a cloud service provider. All backups of critical systems will be performed quarterly to ensure Master Sword Publishing can recover from a cybersecurity incident.
- **Endpoint Protection:** All Master Sword Publishing workstations and servers will use endpoint protection tools to detect malware, suspicious activity, and unauthorized changes. Security scans will be scheduled regularly, and alerts will be reviewed by the IT & Cybersecurity department. Any affected endpoints will be isolated from the network until it can be investigated and cleared for use.
- **Log Monitoring:** Master Sword Publishing will review system, authentication, firewall, web server, and cloud service logs to identify suspicious activity. Logs from Windows Server, Ubuntu Server, Microsoft 365, Google Workspace, AWS, and network devices will be used to support incident detection and investigation. Logs should be retained long enough to support incident review and post-incident analysis.
- **User Account Management:** User accounts will follow the principle of least privilege access. Employees will only receive access required for their job role. Account activity will be monitored for failed login attempts, unusual access patterns, and privilege changes. When an incident involves a user account, IT may disable the account, reset the password, revoke active sessions, and verify MFA enrollment before restoring access.

Phase 2: Detection and Reporting

Users report an incident to the IT & Cybersecurity team. The IT & Cybersecurity department investigates the incident by reviewing system logs to determine what systems may be infected, and analyzes elements of what the user is reporting.

Example Incident:

An employee receives an email claiming that their Microsoft 365 password is about to expire. The employee clicks the link provided in the message and enters their credentials into a fraudulent login page. Shortly afterward, security logs show multiple failed login attempts and authentication activity originating from locations or systems that the user would not normally access. Review of the email headers reveals authentication failures and other indicators commonly associated with phishing campaigns. The incident is reported to the MSPUB IT & Cybersecurity department for investigation. Review phishing simulation results and user behavior metrics.

Phase 3: Assessment and Analysis

Answer the following questions about the incident:

- **What happened?**
 - A MSPUB employee received an email reporting their password was about to expire not sent from the proper MSPUB IT channels.
 - The user entered their credentials into a suspicious web form.

- Reviewing the logs, there have been multiple failed login attempts from the user's account and the account appears to be accessing critical IT systems.
- **Which systems are affected?**
 - Microsoft 365 user account
 - Windows Server Active Directory environment
 - Windows workstations used by the employee
 - Microsoft 365 cloud services including Outlook, OneDrive, and Teams
 - MSPUB web applications used during the phishing simulation
 - Network authentication and logging systems
- **What data is involved?**
 - User account credentials
 - Company email communications
 - Authentication and security logs
 - Employee information stored within company systems
 - Files accessible through Microsoft 365 services
 - Potentially any data accessible through the compromised user account
- **What severity level applies?**
 - High:
 - Evidence indicates a user account may have been compromised, and unauthorized access attempts have been observed against company systems.
 - Multiple business systems may be affected, and immediate containment actions are required.

Phase 4: Containment

Objective

Prevent further damage, stop unauthorized access, and limit the spread of the incident while preserving evidence for investigation.

Short-Term Containment:

- Disable affected user accounts
- Revoke active user sessions
- Block unauthorized IP addresses when applicable
- Isolate affected endpoints from the network
- Preserve relevant logs and evidence

Long-Term Containment:

- Increase monitoring of affected systems and network logs
- Apply temporary access restrictions to affected accounts
- Verify MFA enrollment and enforcement
- Monitor for additional indicators of compromise
- Review related systems for similar activity

Phase 5: Eradication

Eradication Activities:

- Remove phishing emails from company mailboxes
- Reset affected user credentials
- Revoke unauthorized sessions and tokens
- Verify that no malware or malicious software is present
- Review and correct account permissions
- Scan affected systems for indicators of compromise
- Apply required security patches and updates
- Remove unauthorized persistence mechanisms, if discovered

Phase 6: Recovery

Objective

Restore all affected systems and services to normal operation, verify that unauthorized access has been removed, ensure user credentials are secured, and implement corrective actions to reduce the likelihood of future incidents.

Recovery Procedure

In the event a user falls for a phishing email and evidence indicates that an account has been compromised, the user's account will be disabled immediately. The user will be required to reset their password through the IT department, and MFA will be enabled if it is not already configured. Affected devices will be removed from service and investigated for signs of compromise. Logs will be reviewed and malware or antivirus scans will be performed on affected systems. If necessary, backups will be restored to recover impacted data or services. Any identified vulnerabilities will be remediated before systems are returned to production. Once the incident has been confirmed as contained and eradicated, the user account and affected systems will be restored to normal operations. Additional security awareness training may be provided based on the severity and root cause of the incident to help prevent similar events in the future.

Phase 7: Post-Incident Review

Objective

Identify lessons learned, improve security controls, and reduce the likelihood of similar incidents occurring in the future.

Activities:

- Conduct root cause analysis
- Review the incident timeline and response actions
- Evaluate the effectiveness of containment and recovery efforts
- Identify gaps in policies, procedures, or security controls
- Develop corrective actions
- Develop preventive actions
- Update incident response documentation as needed
- Determine whether additional user awareness training is required

Roles and Responsibilities

Employees

Master Sword Publishing Employees are responsible for following company security policies, participating in security awareness trainings, and reporting suspicious emails, security incidents, or unusual system behavior to the Master Sword Publishing IT & Cybersecurity Department as soon as possible. Employees should not attempt to investigate or remediate security incidents on their own.

Information Technology & Cybersecurity Department

The Master Sword Publishing IT & Cybersecurity department is responsible for coordinating all incident response activities. Responsibilities include:

- Incident detection
- Investigation
- Containment
- Eradication
- Recovery
- Evidence collection
- Documentation
- Post-incident review

The department will also provide guidance and communication to affected users throughout the incident response process.

System Administrators

System Administrators are responsible for maintaining servers, network infrastructure, user accounts, backups, and system security configurations. During an incident, administrators assist with account management, system isolation, log collection, vulnerability remediation, and restoration of affected services.

Management

Management is responsible for providing oversight and support during security incidents. Responsibilities include:

- Approving major response actions
- Allocating resources
- Coordinating business communications
- Ensuring compliance with organizational policies and regulatory requirements

Third-Party Service Providers

Third-party service providers may assist with incidents involving cloud services, hosted applications, internet service providers, or other external systems. Providers may supply technical support, incident-related information, and recovery assistance when required.

Responsibility Matrix

Role	Detection	Containment	Recovery	Documentation
Employees	Report	Assist	Assist	N/A

Role	Detection	Containment	Recovery	Documentation
IT & Cybersecurity	Lead	Lead	Lead	Lead
System Administrators	Support	Support	Support	Support
Management	Oversight	Approve	Oversight	Review
Third-Party Providers	Support	Support	Support	N/A

Incident Severity Classification

Master Sword Publishing will classify information security incidents based on their impact to business operations, systems, users, and data. Severity levels help determine response priorities, escalation requirements, and resource allocation.

Low Severity

Definition: Suspicious activity with no confirmed compromise and minimal impact to business operations.

Examples:

- Spam email reported by a user
- Blocked phishing attempt
- Antivirus detection with no system impact
- Failed login attempts without account compromise

Response: Handled by IT & Cybersecurity department through normal operational procedures.

Medium Severity

Definition: Potential compromise affecting a single user, device, or non-critical business system.

Examples:

- Employee clicks a phishing link but no credentials are entered
- Malware detected on a single workstation
- Unauthorized access attempt to a non-critical system
- Repeated suspicious login activity

Response: Investigation and containment actions initiated by IT & Cybersecurity staff.

High Severity

Definition: Confirmed compromise involving user accounts, critical business systems, or sensitive company information.

Examples:

- Employee enters credentials into a phishing website
- Unauthorized access to Microsoft 365 services
- Malware affecting multiple devices
- Data exposure involving company information

Response: Immediate containment and escalation to management. Recovery activities prioritized to minimize business impact.

Critical Severity

Definition: Major security incident resulting in significant business disruption, widespread compromise, ransomware activity, or substantial data loss.

Examples:

- Ransomware outbreak
- Multiple compromised user accounts
- Widespread network compromise
- Extended outage of critical business systems
- Large-scale data breach

Response: Immediate activation of the Incident Response Plan. Executive management and external service providers may be engaged as required.

Severity Matrix

Severity Level	Color Code	Business Impact	Example
Low	Yellow	Minimal	Spam email or blocked phishing attempts
Medium	Orange	Limited	Suspicious activity affecting one user
High	Red	Significant	Confirmed account compromise
Critical	Dark Red	Severe	Widespread compromise or ransomware
Resolved	Green	Normal Operation	Systems have been restored

MSPUB Phishing Campaign Example

- If an employee receives a phishing email and reports it without interacting with the message, the incident would be classified as Low Severity.
- If an employee clicks the phishing link but does not enter credentials, the incident would be classified as Medium Severity.
- If an employee enters credentials into a fraudulent login page, the incident would be classified as High Severity and immediate containment actions would be required.

Incident Scenario

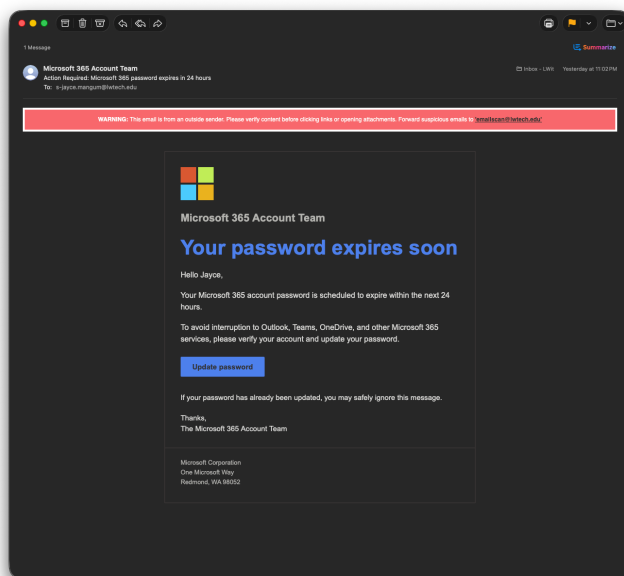
Master Sword Publishing just completed a successful phishing email campaign to get a general scope of how employees are at recognizing a phishing attack.

Incident ID: MSPUB-SAT-2026-001
Campaign Start Date: 06 June 2026 9:00 AM
Campaign End Date: 10 June 2026 9:00 PM

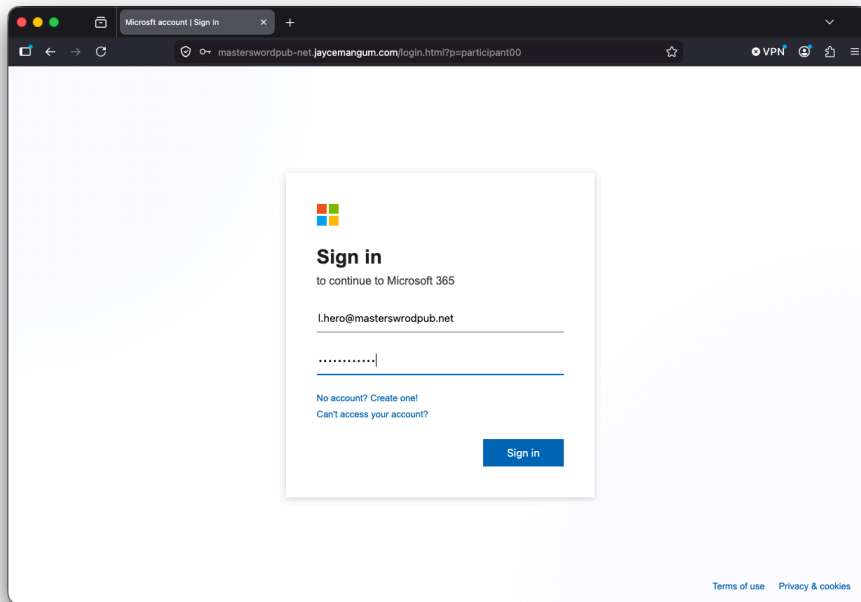
Phishing Simulation Outcomes Severity Levels

Scenario	Severity Level
User reports email without clicking	Low
User opens email but takes no action	Low
User clicks phishing link only	Medium
User reaches login page but enters nothing	Medium
User enters credentials into phishing page	High
Multiple users enter credentials	High
Evidence of account compromise after credential entry	High
Multiple compromised accounts affecting business operations	Critical
Malware/ransomware delivered through phishing campaign	Critical

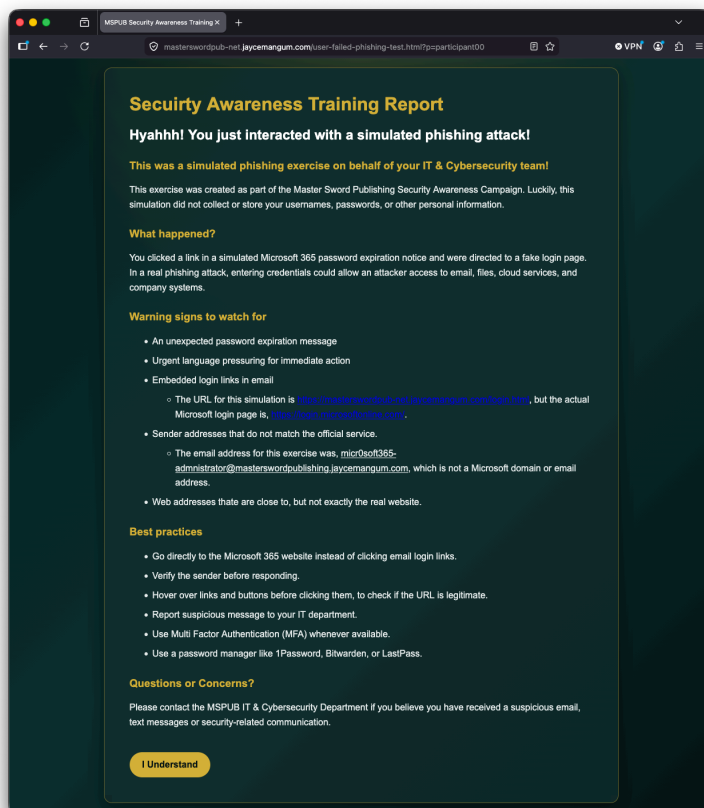
MSPUB employees received an email from what appears to be from Microsoft 365 Account Team requiring immediate action to reset their Microsoft 365 account password.



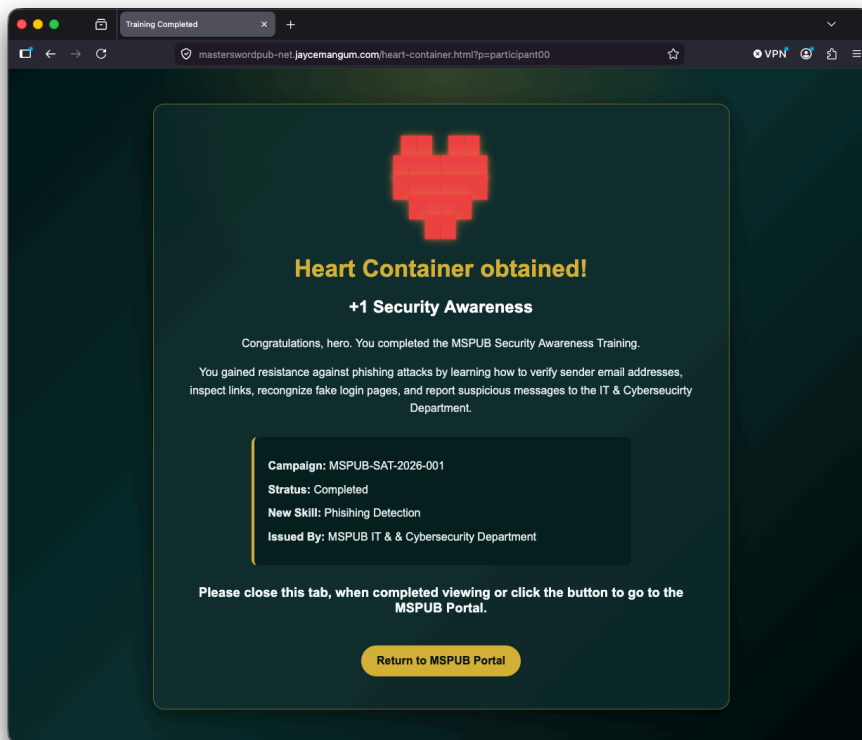
Clicking the “Update Password” button takes users to a suspicious sign-in page asking them to enter their account name and password. In an actual phishing attack, the credential form could record the user’s username and password.



In this controlled simulation attack, the user's credentials were not recorded; however, clicking any links or buttons was logged. Clicking the "Sign in" button took users to a "Security Awareness Training" page where users learned about how to spot a phishing email and to immediately report them to the IT department.



Once users completed the training material, they clicked the “I Understand” button, and it took them to a certification page, congratulating them on completing the training. They were instructed to close this tab.



Each employee was given a ParticipantID to track their progress in an Amazon Web Services DynamoDB, which can then be recorded into an employee progress tracker.

Table: MSPUB-SAT-2026-001-Results - Items returned (50)

Scan started on June 06, 2026, 17:30:39

Actions

	event_id (String)	campaign_id	event_type	page	participa...	timestamp
☐	7117e50c-dd76-4bc2-9874-0cc865dc3156	MSPUB-SAT-2026-001	training_completed	heart-container.html	participant00	2026-06-06T22:40:54.333222+00:00
☐	8978e452-a242-4c8d-a3c9-9c34cf4c58c7	MSPUB-SAT-2026-001	training_page_view	user-failed-phishing-test.html	participant00	2026-06-06T22:40:22.062894+00:00
☐	a079343a-c841-407b-9aad-4f970fe7435b	MSPUB-SAT-2026-001	login_attempt	login.html	participant00	2026-06-06T22:40:21.929175+00:00
☐	daf649d0-987d-48fa-97a7-4bd7120d0fc4	MSPUB-SAT-2026-001	email_link_clicked	login.html	participant00	2026-06-06T22:39:22.327141+00:00

The IT & Cybersecurity department logs this in a separate spreadsheet to track which employees either ignored or reported the email, or if the employee clicked the link, entered their credentials in the form, and went through the training.

Participant Tracker										
Participant ID	Phone	Email Address	Email Name	Custom Email Button Link ID	Email Sent	Email Read Date	Initial Link Clicked	Login Attempt Clicked	Training Viewed	Training Completed
participant01	Uinda E.	Uinda E.	Uinda	https://masterywordpub-net.jaycemangum.com/login.html?participant01	✓	6/6/26 10:25 AM	✓	✓	✓	✓
participant02	Mihail S.	Mihail S.	Mihail	https://masterywordpub-net.jaycemangum.com/login.html?participant02	✓	6/6/26 10:10 AM	✓	✓	✓	✓
participant03	Ryu L.	Ryu L.	Ryu	https://masterywordpub-net.jaycemangum.com/login.html?participant03	✓	6/6/26 10:17 AM	✓	✓	✓	✓
participant04	Nakano M.	Nakano M.	Nakano	https://masterywordpub-net.jaycemangum.com/login.html?participant04	✓	6/6/26 10:16 AM	✓	✓	✓	✓
participant05	Mamoru S.	Mamoru S.	Mamoru	https://masterywordpub-net.jaycemangum.com/login.html?participant05	✓	6/6/26 10:20 AM	✓	✓	✓	✓
participant06	Caden T.	Caden T.	Caden	https://masterywordpub-net.jaycemangum.com/login.html?participant06	✓	6/6/26 10:21 AM	✓	✓	✓	✓
participant07	Taylor M.	Taylor M.	Taylor	https://masterywordpub-net.jaycemangum.com/login.html?participant07	✓	6/6/26 10:08 AM	✓	✓	✓	✓
participant08	Kali T.	Kali T.	Kali	https://masterywordpub-net.jaycemangum.com/login.html?participant08	✓	6/6/26 9:58 AM	✓	✓	✓	✓
participant09	David M.	David M.	David	https://masterywordpub-net.jaycemangum.com/login.html?participant09	✓	6/6/26 9:20 AM	✓	✓	✓	✓
participant10	Anna J.	Anna J.	Anna	https://masterywordpub-net.jaycemangum.com/login.html?participant10	✓	6/6/26 9:55 AM	✓	✓	✓	✓
participant11	Nelson H.	Nelson H.	Nelson	https://masterywordpub-net.jaycemangum.com/login.html?participant11	✓	6/6/26 9:58 AM	✓	✓	✓	✓
Statistics										
Metric		Count	Percentage							
Emails Sent		11	100.00%							
Links Clicked		5	45.45%							
Login Attempts		5	45.45%							
Training Viewed		5	45.45%							
Training Completed		4	36.36%							
Did Not Respond		6	54.54%							

AWS DynamoDB Event Types Severity Levels

event_type ID	Severity Level	Reason
email_link_clicked	Medium	User(s) accessed a phishing link
login_attempt	High	User entered their credentials into a credential capture form
training_page_view	High	User submitted their credentials to a capture form and reached remediation
training_completed	Resolved	User completed the Security Awareness Training

Looking at participant00's data on the AWS DynamoDB table, the user first clicked the email link and was brought to the credential capture form. This would be classified as Medium Severity. The user then entered their credentials into the credentials capture form and clicked the fake "sign in" button, which submitted the form. These events escalated the severity level to High. Once the user read through the training material and clicked "I Understand," the user completed their training on phishing emails and moved the incident into the Resolved level.

Security Awareness Campaign Exception

Incidents generated during authorized Master Sword Publishing Security Awareness Training campaigns are simulated events and do not represent actual compromises. Severity classifications may still be assigned for training and reporting purposes to measure employee susceptibility and improve future awareness efforts.

Communication & Escalation Procedures

Purpose

The purpose of communication and escalation procedures is to ensure that security incidents are reported, communicated, and escalated in a timely manner. Effective communication helps Master Sword Publishing coordinate response activities, reduce business impact, and maintain accurate documentation throughout the incident lifecycle.

Incident Reporting

All Master Sword Publishing employees are responsible for reporting suspected security incidents immediately to the IT & Cybersecurity department. Reports may include suspicious emails, phishing attempts, malware alerts, unauthorized access attempts, unusual system behavior, or any activity that could affect the confidentiality, integrity, or availability of company information systems.

Employees should provide as much information as possible, including screenshots, email messages, timestamps, and descriptions of observed activity.

Escalation Requirements

Low Severity incidents will be handled by the IT & Cybersecurity Department through normal operational procedures. Management notification is not typically required unless the incident becomes recurring or indicates a larger security issue.

Medium Severity incidents will be investigated by the IT & Cybersecurity Department. System Administrators may be notified when technical assistance is required. Management may be informed if business operations could be affected.

High Severity incidents require immediate notification of Management, the IT & Cybersecurity Department, and System Administrators. Containment and recovery efforts should begin immediately to minimize business impact.

Critical Severity incidents require immediate escalation to executive management and all appropriate response personnel. External service providers, cloud providers, legal counsel, law enforcement, or regulatory agencies may be contacted when required by law, contractual obligations, or the severity of the incident.

Communication Methods

Audience	Communication Method
Employees	Email, Microsoft Teams, Phone
IT & Cybersecurity Department	Email, Microsoft Teams, Phone
System Administrators	Email, Microsoft Teams, Phone
Management	Email, Phone, Scheduled Briefings
Third-Party Providers	Vendor Support Portals, Email, Phone

Incident Documentation

All communications related to an incident must be documented. Documentation should include incident reports, notifications, response actions, decisions made during the investigation, recovery activities, and post-incident review findings. Incident documentation will be retained in accordance with MSPUB record retention policies and may be used to support future investigations, audits, security awareness training, and continuous improvement activities.

During the phishing simulations and security awareness campaigns, participant interactions, training completion records, and incident metrics may be collected and reviewed by the IT & Cybersecurity department to evaluate employee awareness and improve future training initiatives.

Evidence Collection & Preservation

Purpose

Evidence collection and preservation are to ensure that relevant information is identified, collected, protected, and retained throughout the incident response process. Proper evidence handling supports investigations, recovery efforts, root cause analysis, audits, and potential legal or regulatory requirements.

Evidence Collection

During a security incident, Master Sword Publishing personnel will collect relevant evidence to support investigation and response activities. Evidence should be gathered as soon as possible while minimizing disruption to business operations and preserving data integrity.

Examples of Evidence

Evidence collected during an incident may include:

- System and server logs
- Authentication and login records
- Firewall and network logs
- Email messages and email headers
- Screenshots of suspicious activity
- Malware detection and antivirus reports
- User statements and incident reports
- Security alerts and monitoring records
- AWS CloudWatch logs
- AWS Lambda execution logs
- DynamoDB event records
- Microsoft 365 audit logs
- Backup and recovery records

Evidence Preservation

Collected evidence must be protected from unauthorized modification, deletion, or disclosure. Evidence should be stored in secure locations and access should be limited to personnel directly involved in the incident. Whenever possible, original evidence should be preserved and copies should be used for analysis to maintain data integrity.

Chain of Custody

Master Sword Publishing will maintain documentation identifying who collected evidence, when the evidence was collected, where it was stored, and who has accessed it. Maintaining a chain of custody helps preserve the integrity and reliability of evidence throughout the investigation process.

Evidence Retention

Incident-related evidence will be retained in accordance with Master Sword Publishing's record retention requirements and organizational policies. Evidence may be used for incident investigations, post-incident reviews, compliance audits, security awareness training, and future security improvements.

Phishing Simulation Example

For phishing awareness campaigns, evidence may include participant interaction records, phishing email delivery records, login attempt logs, training page views, training completion events, and AWS DynamoDB records collected throughout the Master Sword Publishing Security Awareness Training platform. These records may be analyzed to measure employee awareness and improve future training initiatives.

Continuous Improvement

Purpose

Continuous improvements ensure that Master Sword Publishing learns from security incidents and continuously strengthens its security posture. Following each incident, the organization will review response activities, identify areas for improvement, and implement corrective actions to reduce the likelihood and impact of future incidents.

Post-Incident Review

Following the resolution of an incident, Master Sword Publishing will conduct a Post-Incident Review to evaluate the effectiveness of the response process. This review should occur as soon as recovery activities have been completed.

The review will focus on:

- Incident timeline and sequence of events
- Effectiveness of detection and reporting
- Containment and eradication activities
- Recovery efforts
- Communication and escalation procedures
- Impact of business operations
- Opportunities for improvement

Root Cause Analysis

Master Sword Publishing will perform a Root Cause Analysis to identify the underlying factors that contributed to the incident. Understanding the root cause helps prevent similar incidents from occurring in the future.

Examples may include:

- Successful phishing attacks
- Weak passwords
- Lack of user awareness
- Misconfigured systems
- Delayed detection of suspicious activity
- Unpatched software vulnerabilities
- Inadequate security controls

Corrective Actions

Corrective actions will be developed and implemented to address weaknesses identified during the Post-Incident Review.

Examples may include:

- Updating security policies and procedures
- Improving monitoring and alerting capabilities
- Enhancing access control measures
- Implementing additional security controls
- Updating incident response documentation
- Strengthening authentication requirements

Security Awareness and Training (SAT)

Master Sword Publishing recognizes that employees play a critical role in information security. Security awareness training will be reviewed and updated regularly based on incident trends and lessons learned.

Activities may include:

- Additional phishing awareness training
- Simulated phishing campaigns
- Security awareness reminders
- User education on reporting suspicious activity
- Review of common social engineering techniques

Future Testing and Exercises

To ensure continued readiness, Master Sword Publishing will periodically test its Incident Response Plan through tabletop exercises, phishing simulations, and other security awareness activities. Results from these exercises will be reviewed and used to improve response procedures, training materials, and organizational preparedness.

Continuous Improvement Objectives

Master Sword Publishing will strive to:

- Improve incident detection capabilities
- Reduce response and recovery times
- Increase employee security awareness
- Strengthen security controls
- Improve documentation and record-keeping
- Reduce the likelihood of future incidents

Results from the MSPUB Security Awareness Training (SAT) phishing simulation program will be reviewed to identify trends in employee behavior, measure training effectiveness, and support future security awareness initiatives. Findings may be incorporated into future training campaigns, incident response procedures, and security improvement efforts.

Legal & Regulatory Considerations

Master Sword Publishing will comply with applicable legal, contractual, and regulatory requirements when responding to security incidents. Depending on the nature and severity of an incident, MSPUB may be required to preserve evidence, notify affected parties, coordinate with third-party service providers, or comply with reporting obligations.

All incident investigations will be conducted in accordance with organizational policies and applicable laws. Sensitive information collected during investigations will be protected from unauthorized disclosure, and access will be limited to authorized personnel.

Document Control

The purpose of Document Control and Maintenance is to ensure that the Master Sword Publishing Incident Response Plan remains current, accurate, and effective. This document will

be reviewed regularly and updated as necessary to reflect changes in technology, business operations, security requirements, and lessons learned from incidents and exercises.

Item	Value
Document Name	MSPUB Incident Response Plan
Version	1.0
Owner	MSPUB IT & Cybersecurity Department
Approved By	MSPUB Management
Effective Date	June 2026
Review Cycle	Annually or following a major security incident

Review & Update Process

The MSPUB IT & Cybersecurity department is responsible for reviewing and maintaining this Incident Response Plan. Reviews will be conducted annually and after significant security incidents, organizational changes, technology changes, or regulatory updates.

Distribution

The Incident Response Plan will be made available to authorized Master Sword Publishing personnel for incident response activities. Employees may be provided with relevant portions of the plan based on their roles and responsibilities as part of their onboarding documentation and in the employee handbook.

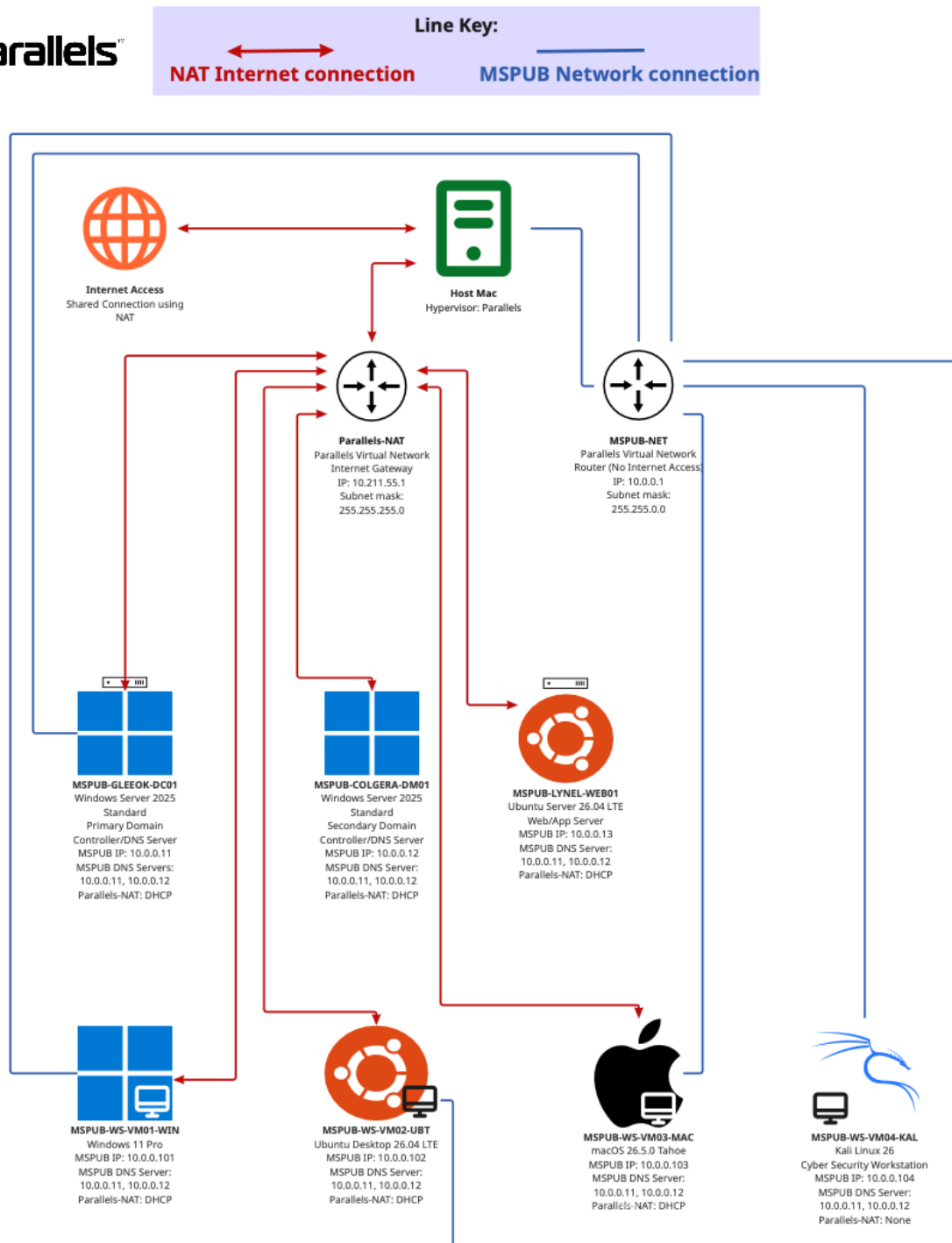
Plan Testing

To ensure effectiveness, Master Sword Publishing will periodically test the Incident Response Plan through tabletop exercises, phishing simulations, security awareness training activities, and incident response drills. Results from testing activities will be documented and used to support continuous improvement efforts.

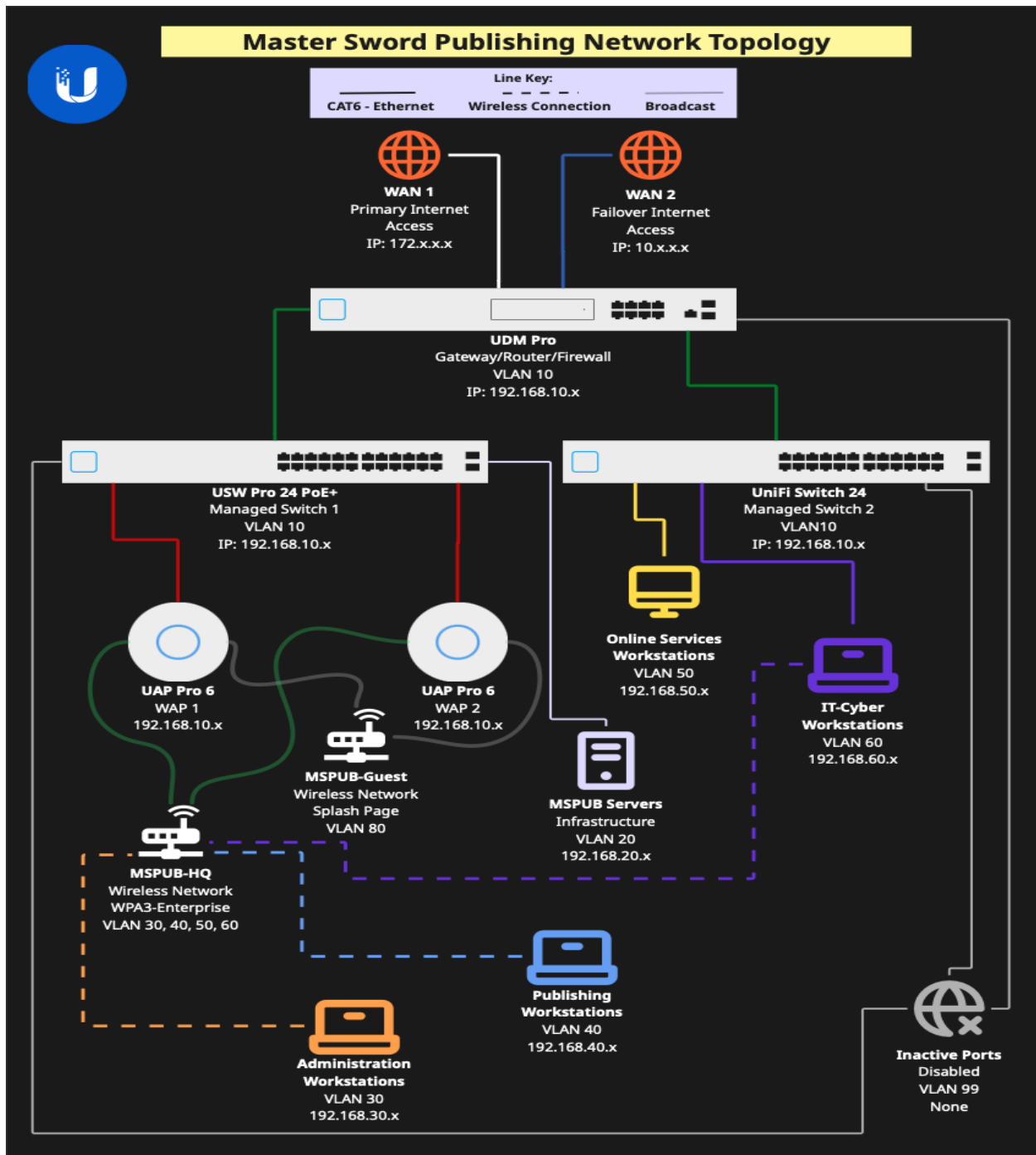
Appendix F - Project Diagrams

Appendix F.1 - MSPUB Virtual Machine Topology

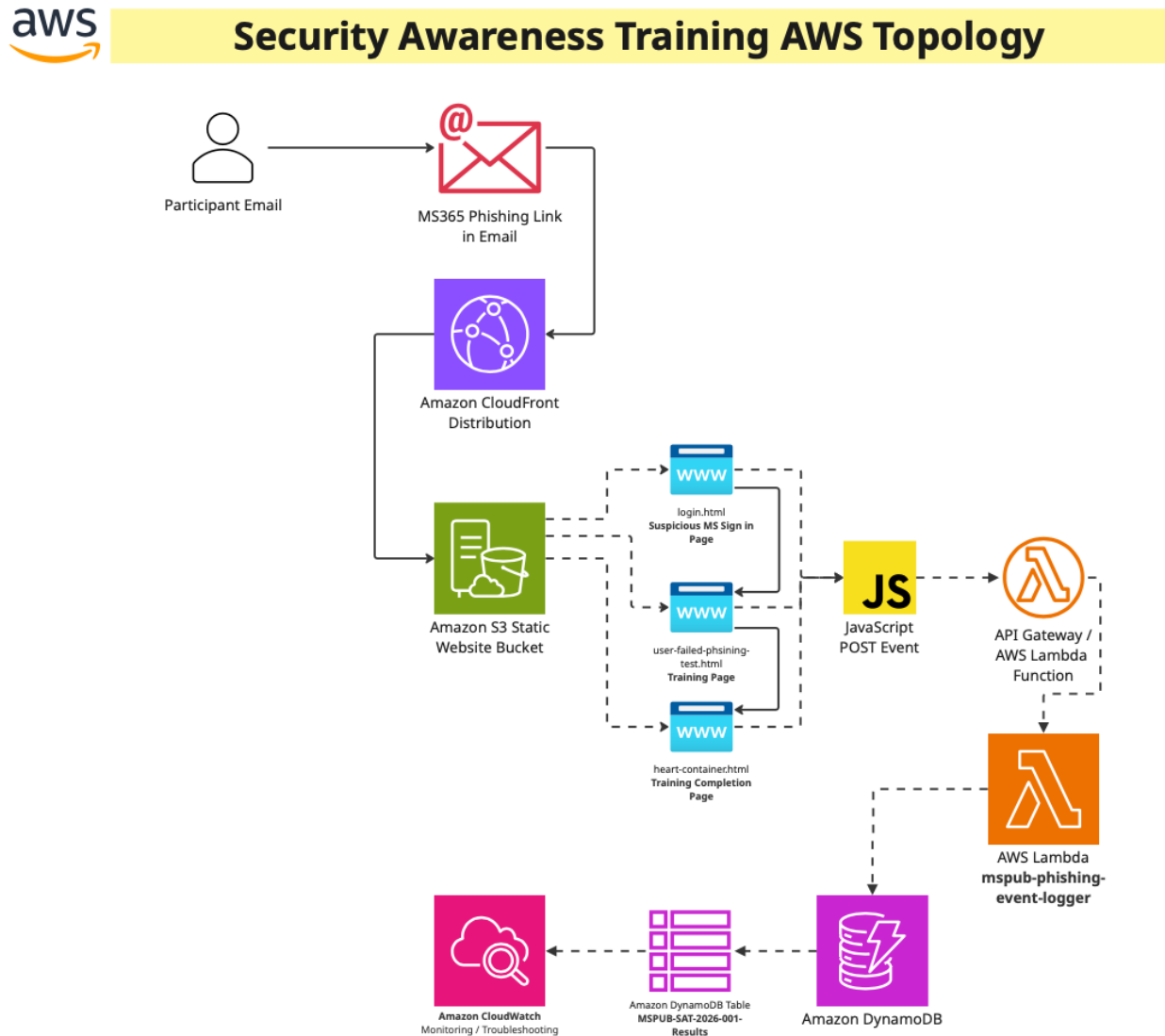
Master Sword Publishing Virtual Home Lab Topology



Appendix F.2 - MSPUB Network Topology



Appendix F.3 - MSPUB Security Awareness Training Topology



References

Amazon Web Services. (n.d.). Amazon CloudWatch: What is Amazon CloudWatch?
<https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/WhatIsCloudWatch.html>

Amazon Web Services. (n.d.). Amazon DynamoDB developer guide: Introduction to Amazon DynamoDB.
<https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Introduction.html>

Amazon Web Services. (n.d.). Amazon Lambda developer guide: What is AWS Lambda?
<https://docs.aws.amazon.com/lambda/latest/dg/welcome.html>

Amazon Web Services. (n.d.). Amazon S3 user guide.
<https://docs.aws.amazon.com/AmazonS3/latest/userguide/Welcome.html>

Apple Inc. (n.d.). *Mac User Guide*.
<https://support.apple.com/guide/mac-help/welcome/mac>

Canonical Ubuntu. (n.d.). *Download Ubuntu Desktop*.
<https://ubuntu.com/download/desktop>

Canonical Ubuntu. (n.d.). *Get Ubuntu Server*.
<https://ubuntu.com/download/server#system-requirements-tab-lts>

Hyperproof. (n.d.). *The ultimate guide to ISO 27001: the complete guide*.
<https://hyperproof.io/iso-27001/>

Kali Linux. (n.d.). *Installing Kali Linux*.
<https://www.kali.org/docs/installation/hard-disk-install/>

Microsoft. (n.d.). *Hardware requirements for Windows Server*.
<https://learn.microsoft.com/en-us/windows-server/get-started/hardware-requirements?tabs=cpu&pivots=windows-server-2025>

Microsoft. (n.d.). *Windows 11 System Requirements*.
<https://support.microsoft.com/en-us/windows/windows-11-system-requirements-86c11283-ea52-4782-9efd-7674389a7ba3>

National Institute of Standards and Technology. (2025 July 18). *NIST Cybersecurity Frame and ISO/IEC 27001 mappings*.
<https://csrc.nist.gov/projects/olir/informative-reference-catalog/details?referenceId=154#/olir/home>

NSF. (n.d.). *ISO/IEC 27001 information security management systems*.
<https://www.nsf.org/management-systems/iso-iec-27001>

Parallels International GmbH. (n.d.). *Parallels desktop for Mac 26 user's guide*.
<https://docs.parallels.com/landing/pdfm-ug>

Reddit. (2022, December 3). *Ubuntu on a Windows domain with Active Directory* [Online forum discussion]. Reddit.

https://www.reddit.com/r/linuxadmin/comments/zbcmt9/ubuntu_on_a_windows_domain_with_active_directory/

Tech With Emilio. (2021, July 7). *How to bind a Mac to Active Directory (join macOS to AD)* [Video]. YouTube.

<https://www.youtube.com/watch?v=jHqY2txtQ0k>

Ubiquiti Inc. (n.d.). *UniFi Network documentation*.

<https://help.ui.com/hc/en-us/sections/6582310816535-Network>

W3Schools. (n.d.). *HTML tutorial*.

<https://www.w3schools.com/html/default.asp>

W3Schools. (n.d.). *CSS tutorial*.

<https://www.w3schools.com/css/default.asp>