COLORADOSCHOOLOFMINES
EARTH • ENERGY • ENVIRONMENT

CSCI 370 Final Report

SentinelOps

Max Garman
Jasper Mesenbrink
Barak Asher
Michael Maggiore

Revised December 7, 2023

CSCI 370 Fall 2023

Mr. Tree Lindemann-Michael

Table 1: Revision history

Revision	Date	Comments
New	8/31/2023	Added introduction, functional requirements, non-functional requirements, risks, and definitions of done.
Rev – 2	9/13/2023	Added system architecture details
Rev – 3	12/4/2023	Worked on final report draft

Table of Contents

I. Introduction

Frontier Technology Inc. (FTI) provides competitive-edge engineering, IT services, and software products to solve difficult challenges for the Department of Defense. FTI has steadily developed a reputation within the government support space, leveraging proven military-tested technologies to provide customized decision-making software. FTI designed this project to test different security postures on virtualized data communication networks to test each network's overall cyber survivability in a cyber-contested environment.

Our team is designing two networks to defend against Red Team penetration for FTI. This is a team of FTI Cyber interns that are trying to white hat hack our network. The first network will be an administrative corporate network for a fake "client" for day-to-day work using VMWare, Database Server, Windows 10/11, and RedHat Linux clients. The second network will be similar but add Kali Linux clients to run red team penetration tools. Each network will test different security protocols and methods to aid in the blue team's learning of cybersecurity defense principles.

Throughout the project, the blue team created a virtualized network using VMWare vSphere. The network consisted of multiple end-user hosts as well as two Active Directory (AD) servers and one domain name server (DNS) server. The red team conducted multiple penetration tests during the project. Our logger was effective at identifying attacks quickly, and the network firewall then eliminated threats.

II. Functional Requirements

1) Secure Service: The network's main goal is to provide secure administrative services to the "client". The service should be isolated from people without access to the network to minimize the risk of unauthorized access.
2) Versatile Information Processing: The network can send and receive ASCII and Non-ASCII text, videos, images, and audio to any connected computer or virtual machine (VM).
3) Scalability: The design is scalable so that more VMs can be added as needed.

4) Detection and Prevention: The network is able to detect and neutralize threats, and continue to function while dealing with an attack. There are scripts in place to scrape log files and harden the system automatically.

III. Non-Functional Requirements

1. To use a virtual machine like VMware
2. Work in tandem with the red team to conduct penetration tests on the network and then improve network security
3. Work with an automated certificate management environment (ACME) network
4. The network will combine the power and capabilities of diverse equipment to provide a collaborative medium that helps users combine their skills regardless of their physical location.
5. Use DISA STIGS to harden the network.

IV. Risks

Technical Risks:

1) The project depends on Windows OS development while some of our team have Unix-based systems
2) May have trouble connecting our network with the red team for testing if not set up properly
3) Could have network connection problems if VMware is not properly set up on the machine
4) May run into security clearance issues

Skills Risks:

1) Shallow knowledge of ACME networks and how to implement them
2) little prior usage of Red-Hat software between the group members

The above risks posed little issue to the team overall. Because the project was sandboxed in an environment that was designed to be broken, many of the risks associated with a standard software project did not apply to ours.

V. Definition of Done

The team will work with real-world software such as Snort and ElkStack to learn how to manage and secure networks and computer systems against unauthorized access and attacks by implementing various security measures and industry best practices. It is up to the team's discretion as to when the network is "done", but the network can be a constantly evolving system.

Overall we felt that we met the definition of done that the team was aiming for. The main purpose of the project was educational, and for the blue and red teams to gain experience working in a real-world scenario through sandboxing. This was achieved, as everyone on the team gained valuable experience in hardening and defending a network. As well, the team learned about how to construct servers and end-user hosts from a technical perspective. This is important information especially if any team members go on to work in the cybersecurity industry.

VI. System Architecture

The project ultimately called for the creation of two separate networks, one for the blue team and one for the red team. We did not have access nor were we involved in the creation of the red team's network. Within the scope of the blue team's network, we created several hardened end-user hosts and some unhardened hosts. This was to give our network a functional "control group". This allowed us to see what the

activity from the red team looked like on unhardened systems, and then after hardening we had machines to compare against. The final network diagram shown in Figure 1 details some of the notable differences between the hardened and unhardened systems.

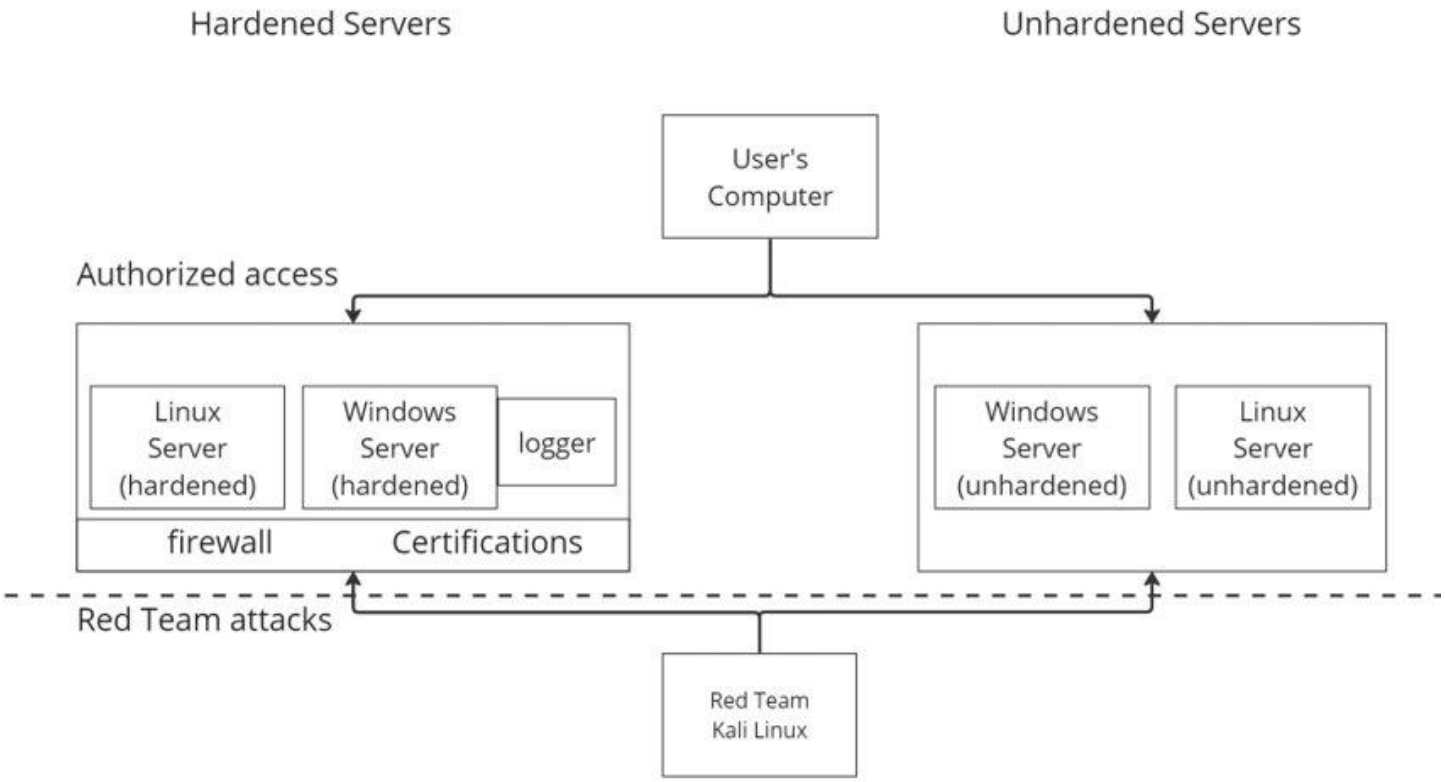


Figure 1: Network Structure

The blue team network was hosted on 3 physical servers at the FTI datacenter, and the team used VMware vSphere to access the servers and create virtual machines. We used a variety of operating systems, such as Windows 10/11 and CentOS. The network contained the hosts listed below:

1. **AD servers**

   These were the Windows Active Directory servers in the network. FTI had created these before the beginning of the project to establish the environment for the blue team to create the virtualized network. AD servers mainly handle authentication and authorization, as well as minimal permission management and overall user handling. These servers stored all of the login information for the blue team and allowed us to access the rest of the network.

2. **DNS server**

   The main server the blue team was tasked with creating was a DNS server. A DNS server handles DNS requests from end-user hosts. This is accomplished by communicating with higher-order DNS servers and receiving domain name information, which is then sent back to the requesting end-user host.

There is significant hardening that can be done with a DNS server, as many common attacks such as DNS tunneling, which can tunnel in malware and other viruses into a network, are accomplished by exploiting a DNS server. The blue team accomplished most of the hardening for the DNS server, but some procedures like hardening our Linux VMs and properly visualizing our logs of network attacks were outside the scope of the project and were not completed.

3. Unhardened end-user machines

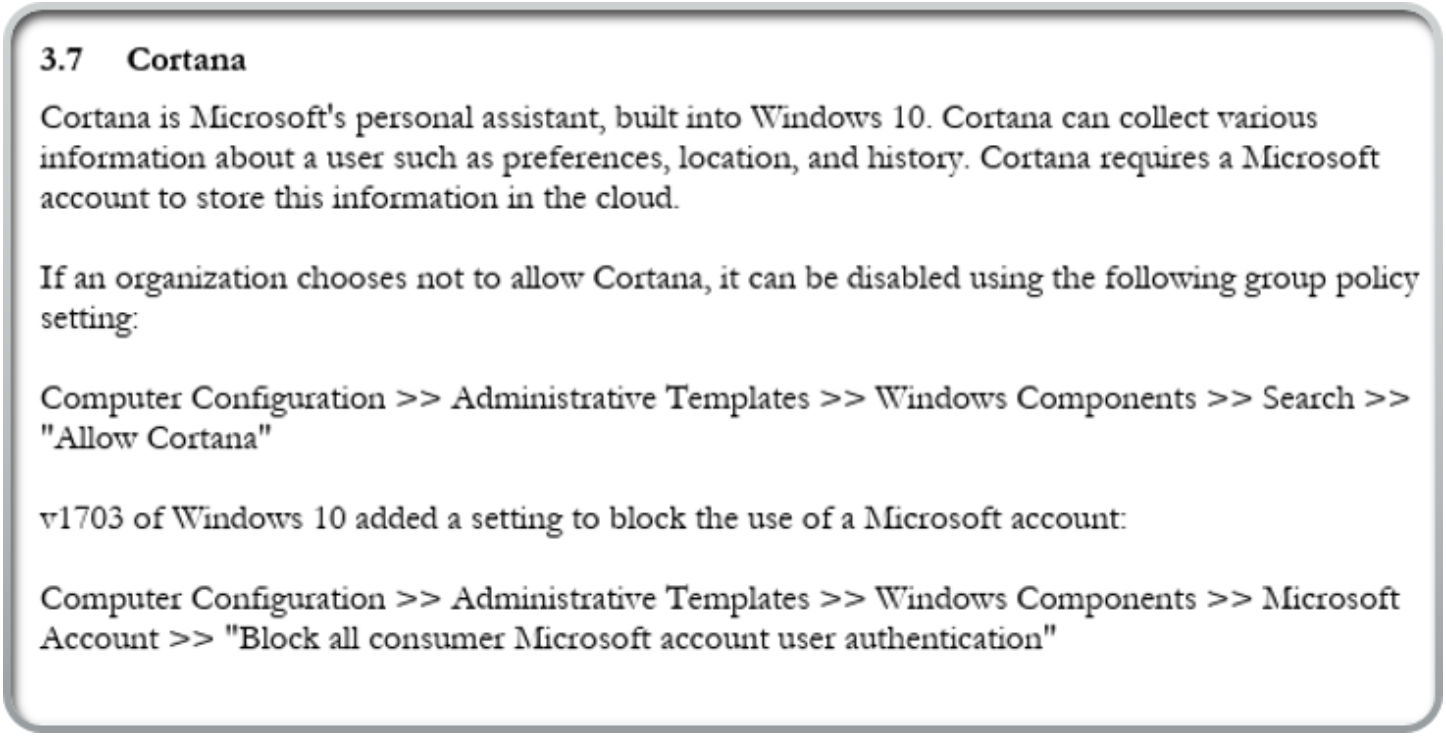
The unhardened end-user hosts were created as the control group for the red and blue teams to compare against as the project progressed over the semester. These were VMs that the blue team created and then left alone, as we didn't want to add any extra features or software to interfere with the role of the machines.

4. Hardened end-user machines

The hardened end-user machines were the bulk of the network, and what the project mainly centered around. These VMs were designed to mimic the everyday usage of the network, so each machine had various levels of permissions across multiple users. These VMs run Snort, a network logging software, as a background process to collect and filter network traffic, allowing us to keep track of red team activity. These hosts were hardened according to DISA STIGS, which will be explained in further detail in the next section.

Security Technical Implementation Guides (STIGS)

STIGs are a tool created by the Defense Information Systems Agency (DISA). They outline the steps a person should take to harden a computer up to the standards of DISA. This mainly involves closing ports and modifying services on the machine for the goal of making the host more secure. STIGs are implemented with the Security Content Automation Protocol (SCAP). SCAP outlines what is satisfactory on a system and what still needs to be done after each execution. This allows for easy implementation of hardening principles by telling the user directly what needs to be changed. Below in Figure 2, is a picture of a portion of the Windows 10 STIG outline modifications for the Cortana personal assistant service.



3.7 Cortana

Cortana is Microsoft's personal assistant, built into Windows 10. Cortana can collect various information about a user such as preferences, location, and history. Cortana requires a Microsoft account to store this information in the cloud.

If an organization chooses not to allow Cortana, it can be disabled using the following group policy setting:

Computer Configuration >> Administrative Templates >> Windows Components >> Search >> "Allow Cortana"

v1703 of Windows 10 added a setting to block the use of a Microsoft account:

Computer Configuration >> Administrative Templates >> Windows Components >> Microsoft Account >> "Block all consumer Microsoft account user authentication"

Figure 2: SCAP guideline

VII. Technical Design

Our project had many interesting components that combined to become a non-trivial cybersecurity network. The first main component was the Domain Name System server that allowed our Virtual Machines to connect to the internet. The DNS server was implemented on a Windows Server 2016 host. The host was configured as a DNS server and knew how to handle requests, but much of the setup was still done by the team. The team created lookup zones and forwarders so the DNS server knew where to send DNS traffic. As well, we configured the firewall rules on the DNS server heavily to allow the red team as little access as possible. DNS functions by mapping domain names to IP addresses. When an end-user host tries to navigate to a website, it asks the DNS server what that domain's IP address is. The server will either have the answer cached or have to ask a higher-order server. In either case, once the IP address is known, the DNS server hands the IP address back to the end-user host. Figure 3 describes the DNS workflow.

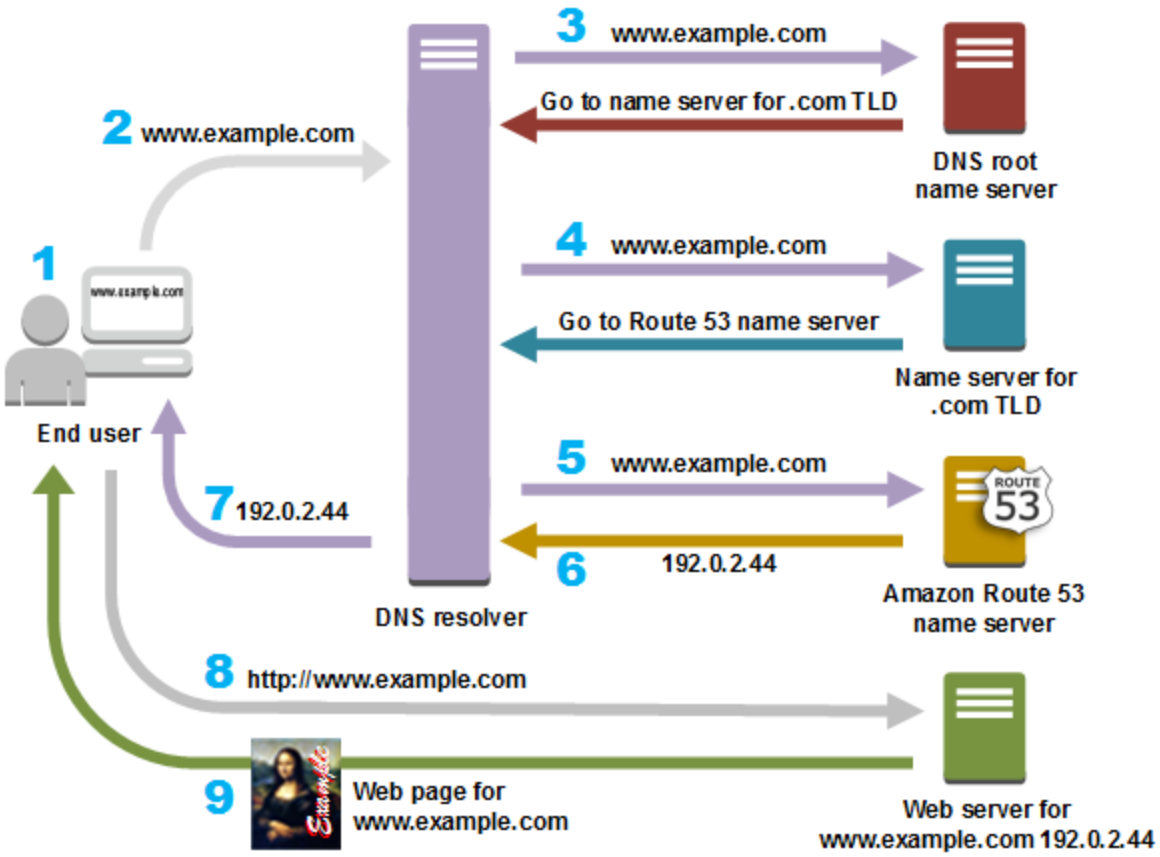


Figure 3: DNS protocol diagram, https://docs.aws.amazon.com/Route53/latest/DeveloperGuide/welcome-dns-service.html

The second main feature of our design was implementing the Security Technical Implementation Guide (STIGs) with the Security Content Automation Protocol (SCAP). To do this we ran SCAP on both our Windows and Linux VM's by installing the proper distribution from the Department of Defense (DoD) Cyber Exchange website. After running SCAP an output prompt appears that shows all the missing configurations for your system that need to be changed to properly fulfill the STIGs regulations. This is extremely important for our project this semester because this is the baseline of computer hardening for any Department of Defense contractors such as FTI. The output from SCAP is depicted below in Figure 4.

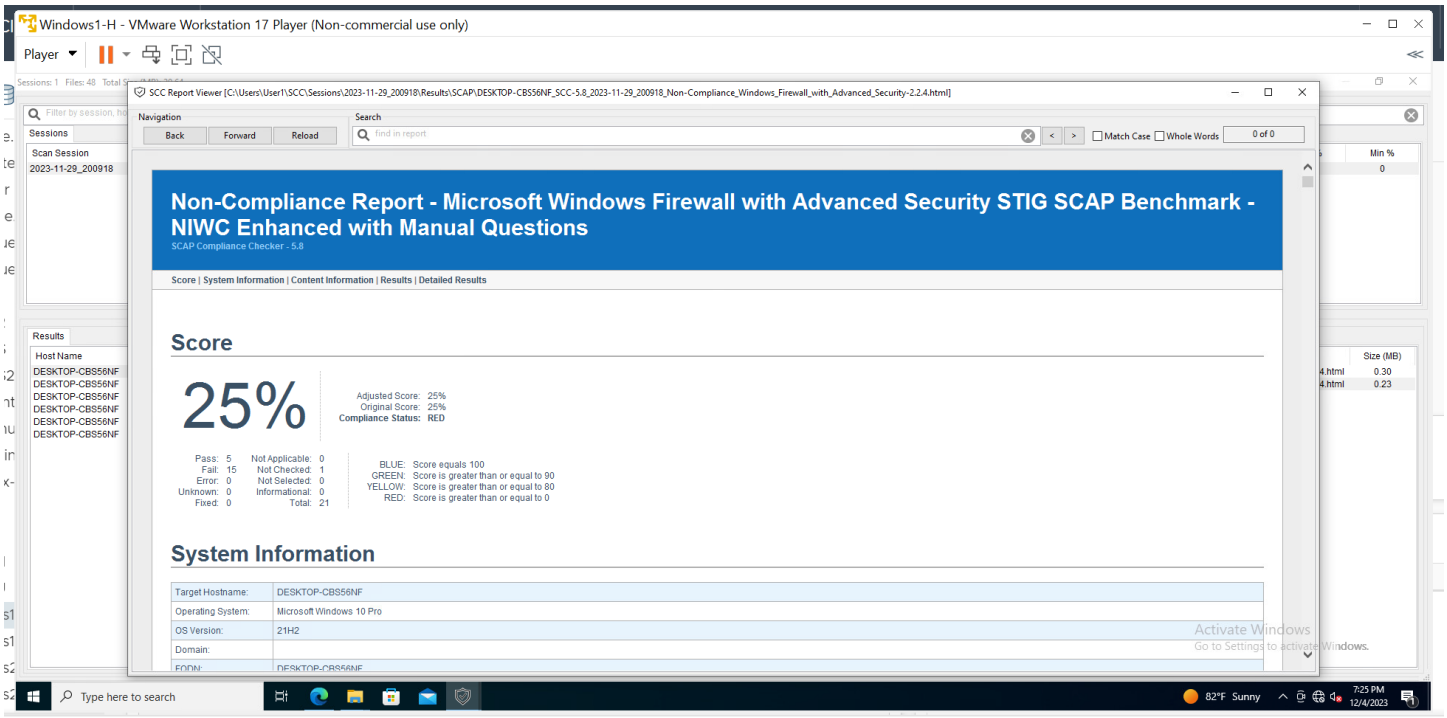


Figure 4: Unhardened SCAP output

Now that we can see what we are missing it is a simple process to change the system settings. Using the command prompt the SCAP scan gives a simple command that can be copied and pasted into the command prompt run as administrator. Below in Figure 5 is a visualization of this process.

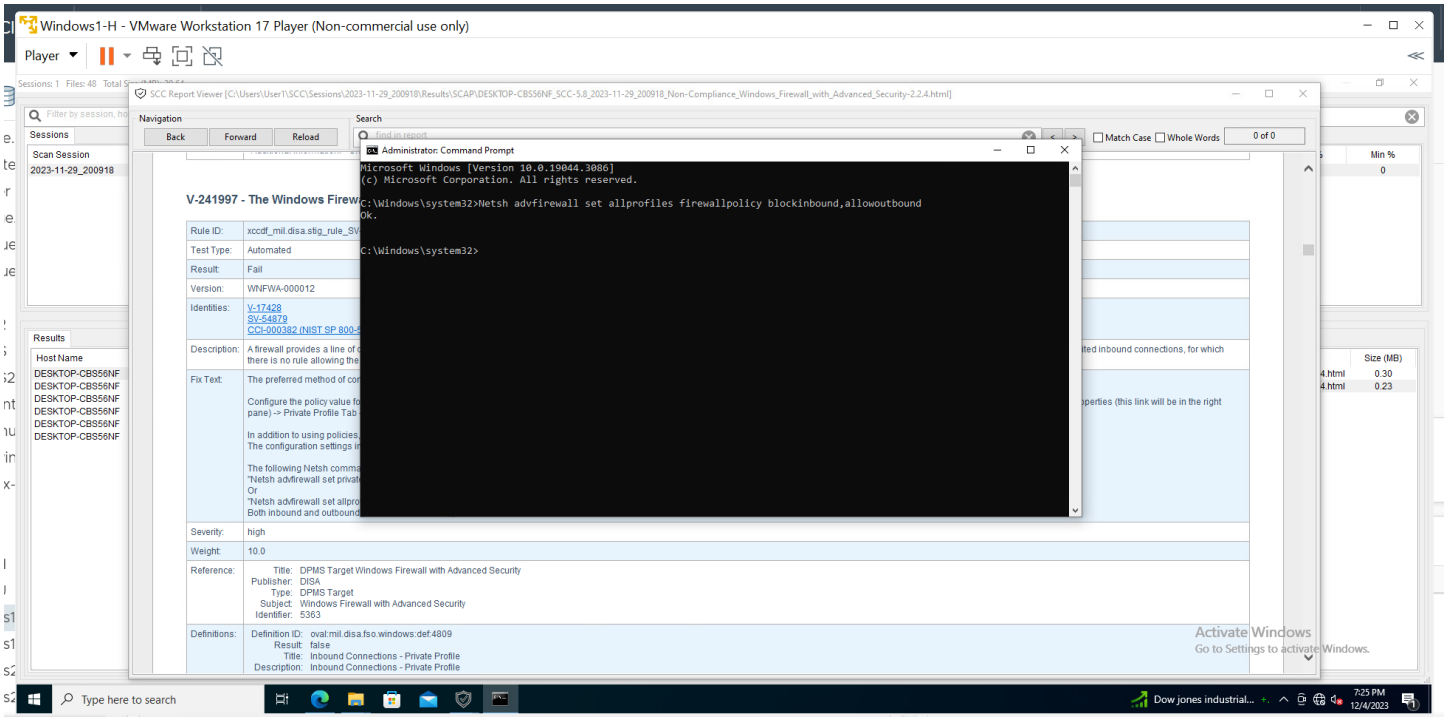


Figure 5: Implementing SCAP guidelines

After this process we see that when we run a scan now, in Figure 6, we are at 55% of the STIGs complete.

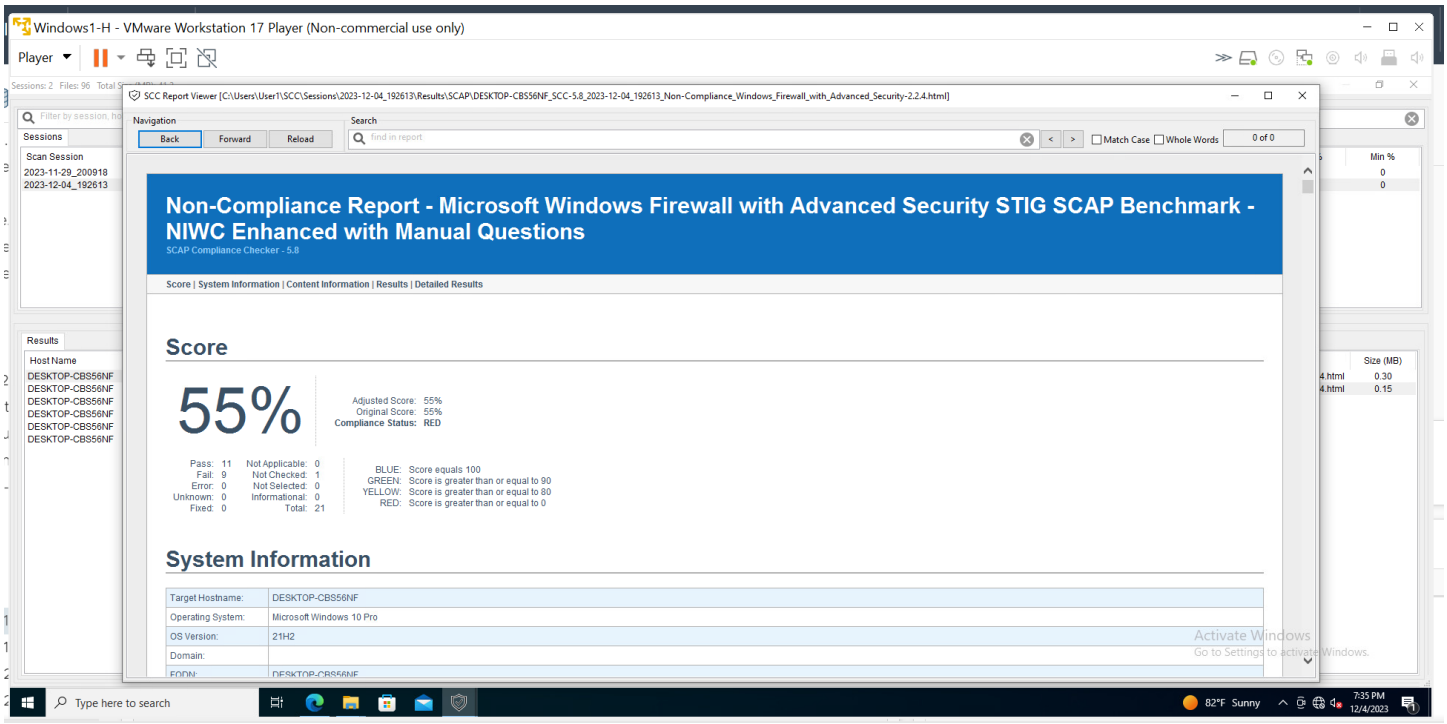


Figure 6: SCAP output after some hardening

VIII. Software Test and Quality

The primary responsibility for our testing lies with the red team, who actively engages in ethical hacking to infiltrate our hardened virtual machines (VMs). Their role is to identify vulnerabilities and provide valuable feedback on potential weak points in our hardening measures. This approach has proven highly effective in uncovering issues that might be challenging to detect during the initial hardening phase. In addition to this targeted testing, we also assess the duration an intruder can persist within our virtual environment without being detected or expelled. Our efforts in this area have yielded positive results, showcasing significant advancements in minimizing the timeframe intruders can operate undetected.

IX. Project Ethical Considerations

Our project is unique in that our entire network was built in a sandboxed environment. Because of this, it is difficult to narrow down any specific ethical issues with our project. If our project was in a real-world scenario, the most important considerations would be the ethical questions associated with white hat hacking.

X. Results

The primary objective of this project was to fortify the ACME, a virtualized data communication network, commonly referred to as a Microsoft Windows Active environment. The aim was to safeguard it against penetration attacks, specifically targeting the ACME network, thereby scrutinizing security postures for optimal cyber survivability. Designed to function as the administrative backbone for the day-to-day operations of ACME Business Solutions, this network demanded a robust defense against potential threats.

To accomplish this goal we had to create a DNS server. This was one of our more significant results due to the fact it was the only way to connect our virtual network to the internet to download and run SCAP to

harden our VMs. Additionally, it gave our IP addresses to our VMs so the Red Team could have the necessary information and targets to penetrate with their Kali Linux network.

Our project focuses on securing networks from adversarial threats, guaranteeing business continuity in the event of compromise, and executing prompt threat detection and neutralization. This aims to enhance the overall cyber resilience of the ACME network. To achieve this, our team delved into the development of scripts dedicated to security hardening and log scraping automation. However, certain crucial components still require attention.

Specifically, the implementation and utilization of LogStash along with Elasticsearch and Kibana remain pending. These tools, formally called the ELK stack, will be used to visualize the data collected through our log scraping scripts. Furthermore, our efforts toward fully integrating hardening procedures for the Linux machines are ongoing. Lastly, we have yet to implement our Security Technical Implementation Guide (STIG) requirements on our linux VM, which will further fortify the network's defenses. The completion of these outstanding tasks will not only reinforce the security measures of the ACME network but also contribute to a more comprehensive and resilient cybersecurity framework.

XI. Future Work

Continuous script refinement is imperative, necessitating ongoing updates to identify any overlooked security logs that may evade our latest script enhancements. The red team's feedback will serve as a crucial guide in this iterative process. Moreover, our commitment to enhancing our situational awareness involves updating the visualization of our logs using the ELK stack. This upgrade aims to provide a more comprehensive and insightful perspective on the efficacy of our security measures.

The definition of done coincides with implementing the Security Technical Implementation Guide (STIG) requirements. This step is pivotal in ensuring the project meets its predetermined criteria for completion and compliance.

Other future work includes adding a Dynamic Host Configuration Protocol (DHCP) so new VM’s to the network are automatically assigned an IP address and can access the internet. We would also add a Virtual Switch between the DNS server and the VM’s so that any packets sent over the virtual network are ensured to be safe.

Furthermore, we can even add AI/ML to our project to take in different logs, find any suspicious behavior, and add automation to deal with intruders on our network. Additionally, future developers can continue to keep our hardened servers up to date by automating and running SCAP over every VM once a month or so and updating the required STIGs.

XII. Lessons Learned

Up to this point, our team has gained insights into the diverse array of log IDs crucial for detecting suspicious activities. We've recognized the importance of striking a balance in crafting blacklist rules, which ensures an authentication system that doesn't impede legitimate employee access to virtual machines (VMs).

Furthermore, our learning journey has extended to the realm of cybersecurity regulations inherent in our affiliation with a Department of Defense (DoD) contractor. This encompasses a thorough understanding of cybersecurity rules necessitated by our line of work, with a specific emphasis on adhering to various Security Technical Implementation Guides (STIGs) available online.

XIII. Acknowledgments

We would like to thank FTI and our advisor Tree Lindemann-Michael for their help on this project. Our project was extremely educational for the team and we all enjoyed learning how to work with network infiltration and virtual machine hardening.

XIV. Team Profile

Max Garman

Computer Engineering Senior

Hometown: San Francisco, CA

Work Experience: NIST software intern, Head TA for Computer Organization

Hobbies: Cycling, Swimming

Barak Asher

Computer Science Junior

Hometown: Pittsburgh, PA

Work Experience: Innowatts Intern

Hobbies: Weight lifting, Snowboarding

Michael Maggiore

Computer Science Senior

Hometown: San Francisco, CA

Work Experience: CACI Software Engineer Intern

Hobbies: Swimming, Water polo, Guitar

Jasper Mesenbrink

Computer Engineering Junior

Hometown: San Diego, CA

Work Experience: ICR Software Intern and freelance web developer.

Hobbies: Climbing, Ping Pong, and Snowboarding

References

Appendix A – Key Terms

Term	Definition
VMWare	*A common software for accessing VMs*
Snort	*An open source network intrusion detection system*
ELK stack	*A software designed to visualize security logs*
STIG	*Stands for Security Technical Integration Guide, a document outlining requirements for a specific product*
RedHat	*A deployment apparatus for Linux and other products*
Red Team	*A team of engineers tasked with breaching a secure network to find flaws*
Blue Team	*A team of engineers tasked with securing a network against attacks*

DNS	*Domain Name Server, which is a naming system for computers connected to the internet. or Internet Networks.*
DHCP	*Dynamic Host Configuration Protocol, which is a network management protocol that assigns IP addresses to the network through a specific client-server architecture.*
VM	*Virtual Machine, which is an emulation of a computer system.*
IP	*Internet Protocol, which is a set of internet standards that maps out routing addresses on the internet.*