CSU Cyber Practice Range

4/20/2022

Nathan Satterfield

Cybersecurity

Bachelor of Science

Julie Henderson

**Statement of Purpose**

The Charleston Southern cyber competition team currently lacks an environment to practice many of the skills and techniques that they are learning. For many students it is very difficult to truly learn and understand without hands on practice. This leads to students having the knowledge of what to do but without the experience to be able to implement that knowledge. The simplest way to fix this issue is to setup an environment where they can practice what they have learned leading to a better understanding of the topic. This would also increase the team’s performance in many of the competitions that they compete in as it could be used to simulate the competition environment.

The goal of this project is to provide Charleston Southern computer science and cyber security students somewhere to practice the skills they are learning. This will be accomplished by providing a server that will host several virtual machines to emulate the format of many of the competitions the students participate in. These virtual machines will be purposefully misconfigured to be insecure so the students can practice hardening systems. There will be several different operating systems included consisting of at least a few windows and Linux based systems. Also included will be a list of items that will be scored vulnerabilities and a guide on how to find and fix them along with information on why it is a vulnerability when possible. Along with the list of vulnerabilities, when possible, there will be an automatic scoring engine, so the students know how many of the scored issues, they have found without having to cheek the list. There will also be a check list of items to complete for each operating system type which will allow the students to follow along which will help them know what steps are most important and what they have completed along with what steps they have remaining.

**Research & Background**

Research was done into what systems already existed for this purpose as there was a cyber range in place although it was never fully completed. The existing infrastructure needed to be evaluated to see if it would work with the new system. The majority of the research done was on ESXI into how to get a limited but free license as well as how to create and setup the server. Research was also needed to find a way to score the vulnerabilities on the Linux systems as the CyberPatriot scoring engine only works for windows-based machines.

**Project Languages Software and Hardware**

There will be some scripts written using windows power shell, bat files, bash, and sed to automate many of the tasks for hardening the systems. For hardware there will need to be a server to host the virtual machines and networking equipment to be able to access the server. For software Windows and Linux operating systems will be needed as well as ESXI as a hypervisor to host the virtual machines. The CyberPatriot Scoring Engine will be needed to automate scoring on the machines from <https://www.uscyberpatriot.org/competition/training-materials/practice-images>. For the Linux operating systems, the pysel scoring system will be used which can be found at <https://github.com/FWSquatch/pysel>.

**Project Requirements**

Req #1 Enumerate network

Functional

Create a network map of the network that the cyber range will run on.

The goal of this project is to create a cyber range that can be accessed over the network which requires knowing how the network is currently setup so the cyber range can be added to the network.

After this requirement is completed there should be a finished network map available that describes the network.

Priority #1

Dependencies none

Req #2 Configure network

Functional

Ensure the network is configured properly to let students access the cyber range over the proper network.

If the network is misconfigured students will not be able to access the cyber range and thus will not be able to use it.

After this requirement is completed, students should be able to connect to the network that the range will run on

Priority #2

Dependencies #1

Req #3 Setup the host server.

Functional

Configure the server that will host the virtual machines.

This server is the base for the cyber range without it the virtual machines that make up the range will not be accessible by the students.

After this requirement is completed the host server for the cyber range should be ready to host virtual machines to students over the network.

Priority #3

Dependencies #1 and #2

Req #4 Create virtual machines.

Functional

Create the virtual machines that will make up the environment.

The virtual machines are what the students are going to use to practice on they are the main part of the project that the students will interact with.

After this requirement is completed there should be standalone virtual machines that students can use to practice on.

Priority #4

Dependencies none

Req #5 Host the virtual machines on the server.

Functional

Take the virtual machines that have been created and move them onto the sever to be hosted.

Once the virtual machines have been created and the host server configured the last step to create a functional cyber range is to combine the two so that students can access the virtual machines via the server.

After this requirement is completed there should a functional cyber range with virtual machines accessed over the network.

Priority #5

Dependencies #1-4

Req #6 Accessibility

Security/Access

The cyber range should be configured in such that it can only be accessed via the correct local area network.

This is important as unauthorized users should not be able to access the range so the server should only allow host to connect from the CSU computer lab network.

After this task is done the server should only be accessible from the CSU computer lab network.

Priority #6

Dependencies #1-5

Req #7 User capacity

Performance/ Capacity

The cyber range should be able to support up to 6 people concurrently.

This is important as most competitions are team competitions generally allowing up to six team members so this would allow the whole team to practice at the same time.

After this task is done the server should be able to support a minimum of six users connecting at the same time.

Priority #7

Dependencies #1-5

Req #8 expandability

Performance/ Scalability

The cyber range should be configured in such that new virtual machines can be added at any time.

This is important as what the students want to practice may change as new operating systems and updates to existing ones happen.

After this task is done the server should be configured such that adding new virtual machines or making changes to existing ones can be done.

Priority #8

Dependencies #1-5

Project Implementation Description & Explanation

Github and documentation link: https://github.com/Nathan-Satt/Senior\_Project

Test Plan

ESXI Tests:

1. test connectivity to the machine hosting the virtual machines from AH 203 Ethernet ports.

2. test user account for ESXI to make sure they can be logged into.

3. test ESXI user permissions to see if they can interact with the virtual machines properly.

Virtual machine Tests:

1. test Windows scoring engine to ensure it works.

2. test Ubuntu scoring engine to ensure it works.

3. test one vulnerability from each category for each virtual machine.

4. test virtual machine backups to ensure that they can be reverted to a completely unsolved state without data lose.

Test Results

The first test was to ensure connectivity from the ethernet ports in Ashby Hall 203, to test this each of the four ethernet ports on the bottom row on the back wall were plugged into and the server was connected to. Each of the four ports behaved as expected and allowed a connection to the server. The second test was to make sure that users would be able to log into the premade accounts. To test this the server was connected to and the credentials were used to attempt to login. The admin account as well as all of the user accounts created were able to successfully login with no issues. The last test for the ESXI server was to ensure that users had the correct permissions. Users need to be able to login start virtual machines and restore them to the original state using the built-in snapshot feature. To test this the user accounts were logged into and then I attempted to start one of the virtual machines. This worked as expected and the users were able to start the machines. Finally, while logged in as a user I attempted to restore to an existing snapshot which was successful.

The second set of tests were for the virtual machines themselves. The first test in this section was for the Windows scoring engine to ensure it works. This was done after it was installed by ensuring that the scoring report was displayed and updated properly. This test was successful, and the scoring engine was working properly on all of the Windows machines. Next was the test for the Ubuntu scoring engine where the previous test was performed again but on the ubuntu engine. This test was successful as well. After the tests to ensure that the scoring engines were running properly, I begin to test the individual vulnerabilities to ensure that the scoring engine was configured properly and detecting changes correctly. This was done one virtual machine at a time by login into each system and fixing all of the scored issues. For the windows machines there were no issues, and everything was scored properly as seen in Fig XXXX-YYYYYText, letter

Description automatically generated

Text, letter

Description automatically generated

Text

Description automatically generated

After the windows machines were tested, I moved on to the ubuntu machines to begin testing them. There were no issues with the first and third ubuntu machine but there were some slight misconfigurations in the scoring engine for second machine. The script to detect the setting for the password warning age did not behave as expected so changes were made to detect the correct string in the given file. There was also an issue with the new user issue as at first the name of the user to be added was ed but was changed to edward as the string ed already existed in the passwd file inside the name of one of the other user accounts. After these issue were fixed all of the vulnerabilities were scored as expected as seen in Fig XXX-YYYTable

Description automatically generated

Graphical user interface, application, table

Description automatically generated

Graphical user interface, application, table

Description automatically generated

The final test was to ensure that the backups function properly and allowed a user to reset the image to its starting state. This was done by using the snapshot restore feature on each of the machines after the tests above in which all of the issues were fixed. After the snapshot was restored none of the issues had been resolved and the machine was back to the purposely misconfigured state. While there were a few issues initially all of the test passed after the proper changes to configurations were made.

Challenges Overcome

There were several challenges that needed to be overcome, this first was obtaining a license for ESXI after some research I found this webpage <https://customerconnect.vmware.com/en/group/vmware/evalcenter?p=free-esxi7> which after creating an account provides you with a license that allows for limited functionality. The next issue I ran into was a hardware compatibility with the Nehemiah server and ESXI. To fix this I was able to download an older version of ESXI which was compatible with the hardware in Nehemiah. The next problem I ran into was finding a scoring engine to work with the Ubuntu machines as the CyberPatriot does not provide one. After doing some research I found a scoring engine based on python that was available for Cyberpatriot coaches and mentors on github at <https://github.com/FWSquatch/pysel>. The final challenge that I had was with the python script that monitored the password warning date. The issue was that it was looking to see if a string matching the regular '^PASS\_MIN\_DAYS\s\*0' I made a small change to the regular expression so that instead it was looking for the string needed to set the password warning age.

Future Enhancements

There are several improvements that could be made to this project, the first is to create more virtual machines which would allow users to practice on more than the six existing images. The second improvement that could be made is adding more events that the pysel scoring engine can check. This can be done by coding simple python scripts as described in the pysel documentation. The last main enhancement that could be made is to set up the environment in such a way that it could simulate an active attack environment. I believe it is possible to segment the virtual machines on different virtual networks which would allow the machines to be setup so that one group of students would be able to attempts to attack the other. This would help with active defense practice for competitions such as the Palmetto Cyber Defense Competition or the National Collegiate Cyber Defense Competition where a red team is actively working against blue team competitors.

Defense Presentation Slides