PERFORMING A PENETRATION TEST ON A STORAGE NETWORK: PROJECT ANALYSIS

Scott Moskal

The University of Akron

CIS Senior Cybersecurity Project 2440:491

Dr. John Nicholas

April 5, 2022

Project Analysis Purpose

The purpose of this document is to provide a general summary for the configuration of each device. This document will also discuss any changes made from the original plan as outlined in the Project Proposal.

Project Analysis Scope

This document will include a list of devices that were used to create the project, an outline of how each device was configured, and all methods to test connectivity on the network. An explanation of each penetration test and exploit will be given, including why it was chosen, how the test was conducted, and the results of each test. Since changes were made from the original proposal, information on what was changed, why the change happened, and how any problems prior to the change were fixed will also be provided.

Project Analysis Limitations

This document is a general analysis of the project, and will not go in-depth on how to fully configure and test each device. All information pertaining to both the full configuration of each device and the process for testing each device will be provided in the Project Description and Testing Documentation, respectively.

General Description of Each Device

The original plan for the project was outlined in the approved project proposal. The original plan for the network was for three routers to be used, one of which was to a wireless router. A switch was used to establish two VLANs on the network. Finally, four end devices were used, including two laptops to take advantage of the wireless network that would be provided by the wireless router, and two devices with wired connections to the switch. One device end device was used for each VLAN.

The final version of the project was similar to what was outlined in the proposal. The only change made for the devices happened on the wireless subnet. Rather than one router that functioned as both a router and a wireless access point, two separate devices were used to fulfill a dedicated purpose. A new router was added, but had no wireless functionality, and a different wireless router that was set up as an access point.

Router 0, initially called Wireless Router, was configured to provide connectivity between devices on the wireless network and the devices on separate subnets. Port eth0 on Router 0 was configured with the address 10.1.1.1 on subnet 10.1.1.0/29. 10.1.1.0/29 was chosen to provide an easy distinction for the wireless network from the rest of the subnets. Port eth1 was configured with the address 172.16.1.2 on subnet 172.16.1.0/30.

The TP-Link Access Point was configured to allow for both laptops to connect to the network via a wireless connection. The device was a router by default, but had the option to be used as a wireless access point for the network. The IP address the device used was statically assigned to it via DHCP. The address given to the device was 10.1.1.6.

The Acer Laptop was both the main device used for initial configuration of all the networking equipment, as well as the device that would be used as a general-purpose device for normal network use. The IP address the device used was statically assigned via DHCP. The address given to the device was 10.1.1.2. Windows 11 was the operating system the device had installed, as the operating system was installed on the laptop at the initial purchase before beginning the project.

The HP Laptop was the device used to interface with all networking devices after all of the devices were configured to communicate with the wireless network. This was also the device that would execute most of the penetration tests and exploits. The IP address the device used was statically assigned via DHCP. The address given to the device was 10.1.1.5. Parrot OS was the operating system used on the laptop, as the operating system came with a suite of security and ethical hacking tools that could be used on the network, as well as having an appealing and easy-to-use interface.

Router 1 was configured to provide connectivity between all end devices on each end of the network. Port eth1 was given the IP address 172.16.1.1. Port eth2 was given the IP address 172.16.3.1. The subnets the router operated on were 172.16.1.0/30 and 172.16.3.0/30, respectively. Similar IP addresses were used to denote the router being used for a WAN setup instead of a LAN/WAN setup the other two routers used. Router 1 was also the default gateway on the WAN, denoting the centralized location of the router on the network.

Router 2 was configured to provide connectivity to both VLANs on the network, allowing devices on both VLANs to communicate with all other devices on the network. Port eth1 was the WAN port, and given the address 172.16.3.2 on the 172.16.3.0/30 subnet. Port eth0 was given three addresses, two of which were the addresses used by the VLANs. Eth0 had an

address of 192.168.1.1 on the 192.168.1.0/29 subnet, connecting the router to the switch. Port eth0.10 had VLAN 10 attached with an address of 192.168.10.1 on the 192.168.10.0/29 subnet. Port eth0.20 had VLAN 20 attached with an address of 192.168.20.1 on the 192.168.20.0/29 subnet. The router was the default gateway for all three subnets attached to port eth0.

All three routers were from the same manufacturer, so all three devices had similar setups for the firewall. When performing initial configurations, each router was given a firewall by default. All devices also had the same process for setting up the routing table. Router 0 had the routing table set up so that all traffic on the network would use Router 1's first WAN interface as the next-hop address to the rest of the network. Router 1 had the routing table set up so that traffic coming from both VLANs would use Router 0's WAN interface as the next-hop address. Traffic coming from the wireless network would use Router 2's WAN interface as the next-hop address. Router 2 had the routing table set so that all traffic traveling from the VLANs would use Router 1's second WAN interface as the next-hop interface.

The Switch was configured to provide communication between the VLANs and Router 2. The Switch used the IP address 192.168.1.2 on the 192.168.1.0/29 subnet to communicate with the router. There was no option to give the Switch an IP address based on which port had which VLAN attached. Port 5 was the designated trunk port of the device. Port 1 was used for VLAN 10, while port 2 was used for VLAN 20.

The BeeLink Desktop was used as a second device for performing tests and exploits. The device had the IP address manually assigned on the wired connection, using the IP address 192.168.10.3 on the 192.168.10.0/29 subnet. The device was connected to VLAN 10. Kali Linux was used as the operating system for the device due to familiarity with using the software.

The Raspberry Pi was used as the server on the network. The device had OpenMediaVault installed on top of Raspbian OS, allowing the device to become a storage server for files on the network. The protocol used for the server was the File Transfer Protocol, or FTP. The device was connected to the Switch on VLAN 20, using the IP address 192.168.20.3 on the 192.168.20.0/29 subnet. The address was assigned manually.

Connectivity and Testing

Both laptops were connected to the wireless network via the TP-Link Access Point. The TP-Link Access Point was connected to port eth1 of Router 0, using LAN port 1 on the back of the TP-Link Access Point. Router 0 was connected to Router 1 by connecting Router 0's port eth1 to Router 1's port eth1. Router 1 was connected to Router 2 by connecting Router 1's eth2 port to Router 2's eth 1 port. The Switch was connected to Router 2 by connecting port 5 of the switch to port eth0 on Router 2. The BeeLink Desktop was connected to port 1 of the Switch, while the Raspberry Pi was connected to port 2 of the Switch. All non-wireless connections were made using ethernet cables, with connections between different devices using Cat.5e cables, while connections between the routers used crossover cables.

To test connectivity, the ping command was issued between the four end devices. Connectivity was verified to be successful when all devices showed a response across all end devices.

Penetration Tests and Exploits

This project prioritized user access to different parts of the network. This included users accessing the storage server, as well as users access the wireless network. Passwords were used for every device on the network, as well as for users attempting to access the storage server. The focus of the exploits was on the effectiveness of the passwords that allowed users to access the network and the resources housed within the network. Testing was also done to see how secure network communication was between devices, and what information may be vulnerable to potential attacks. Nmap and Wireshark were chosen to discover and monitor the communication protocols that were commonly used between each device. Metasploit and Aircrack-ng would attempt to take advantage of the numerous passwords used on the system by attempting to find weak passwords and access restricted parts of the network. Bettercap would attempt to discover and spy on a device sending information all across the network. OpenVAS would find any other vulnerabilities not initially anticipated while designing and configuring the network.

Nmap discovered all ports open on each device, showing all necessary connections used by the network to allow for effective communication between each device. Wireshark monitored traffic flowing across the network, showing encrypted packets when devices logged into the networking devices on the network. OpenVAS discovered all vulnerabilities that existed within the network after all devices were configured and could communicate with each other. All vulnerabilities were patched.

Of the three exploits used against the network, only one was successful. Metasploit failed to gain access to the FTP server, while Aircrack-ng failed to find the PIN used to access the wireless network. Both tools relied on a list of potential passwords that could be used on the network. These passwords were often easy to guess and short, using words typically found in the dictionary. Passwords chosen for all devices and services on the network were a mix of

characters, including uppercase and lowercase letters, as well as numbers. No password included words. As a result, because these passwords were not included in the lists of potential passwords, the tools failed to access the services and expose information.

The attack using bettercap succeeded. The tool was used for a man-in-the-middle attack against the Acer Laptop. The attack was performed on the HP Laptop. The attack first performed an arp-spoofing attack to fool the Acer Laptop into communicating with the HP Laptop instead of Router 0. All traffic was then monitored from the Acer Laptop, including both encrypted and unencrypted information. Patches were made by increasing security and encryption measures on the FTP server.

Problems and Changes Made

Each change is discussed in the order in which the problems occurred, as well as the severity of the problem in how the project was affected. When building the project, only basic connectivity was prioritized in the beginning, so end devices had solutions be given to problems first. Networking devices had problems solved second. Finally, any problem mating penetration testing software to devices was solved last.

The BeeLink desktop had problems supporting an FTP server. Multiple attempts were made to download new FTP servers freely available on Linux devices, including ProFTPd, vsftpd, and CrossFTP. None of the servers had the capability to work with the BeeLink Desktop. When attempting to run the service, all servers would return the message that attempting to start the server failed. No remedy could be found to solve the problem. In addition, with the presence of another FTP server being present on the Raspberry Pi, having an FTP server available on the

BeeLink Desktop came across as arbitrary. Having multiple FTP services could also have posed a security risk to the network, as another service providing the same service using the same protocol could allow too much traffic to flow through the port, and have extra passwords put onto the system that wouldn't be used. As a result, the BeeLink Desktop had a change in how the device was used. The BeeLink Desktop would be used as an extra PC to conduct penetration tests and attempted exploits under. The Raspberry Pi would be the only server on the network. This would allow the BeeLink Desktop to not waste resources while still having a role to play in the project.

The original wireless router had to be switched out because of issues with communication between devices across subnets. The original project proposal featured a router that would act as both a wireless access point and networking router. The router, a TP-Link TLWR841N wireless router, had basic settings for networking and security. The router was restricting in what was allowed to be changed. This included having few firewall options, including not allowing ports to be open for universal traffic flow. The router would allow traffic to leave the network, but would not allow traffic into the network. Devices from the wireless network could freely communicate with devices on the other subnets, but devices on the other subnets could not communicate with the devices on the wireless network. Multiple routers were bought, all low-end TP-Link routers, and the problem was universal across all devices. TP-Link did not seem to allow basic routers to be changed to allow universal traffic flow, presumably as a security feature that could not be turned off or circumvented. To allow universal communication between devices across the network, a new router, a Ubiquiti Edgerouter Lite-3, was obtained. This router was the same kind of router as the other two routers, and was configured similarly, as well. Once this change was made, full communication was seen across all devices.

To still allow wireless functionality on the network, and maintain the use of an exploit against wireless networks, a wireless access point was installed on the network. One of the TP-Link routers purchased previously, a TP-Link TLWR940N, also had the ability to be used as a wireless access point. This change was made, and devices were able to communicate via a wireless connection on the subnet. This change also added an IP address used on the 10.1.1.0/29 subnet.

A change had to be made regarding which device used OpenVAS to run network scans. Originally, the HP Laptop, running Parrot OS, would act as the device that would run vulnerability scans across the network. This was because the software comes partially installed on Parrot OS. However, when trying to complete the full installation of the software on the HP Laptop, the software would continue giving errors about trying to run on the device. Several diagnostics were made, as recommended by the software, but it would fail to find problems despite the software giving conflicting reports about missing databases. An attempt to install the same software on the BeeLink Desktop, running Kali Linux, led to a full installation that could run. As a result, a change was made where the BeeLink Desktop would run the vulnerability scans on the network instead of the HP Laptop.

Minor changes were made as a result of learning about the hardware and software. A change was made to the switch where it was discovered that the individual ports could not have IP addresses attached to them, proper. The device only needed the VLAN specified to each port, instead of an address.

Finally, a USB Flash Drive was needed for installation of the Raspbian OS, as well as being used as the main partition for the device. This was because OpenMediaVault does not allow a drive to be partitioned when the drive also has the service running on it. Installing the

software allowed for folders and services to run on the server, turning the device into the desired FTP server specified in the Project Proposal.