Crafts Team Security Report

Oregon State University

March 2023

Prepared by:

Dam Security

Table of Contents

Dam Security (Security Team Capstone Project)

Overview

Our security team capstone project is made up of numerous smaller projects that include this report covering a mock testing of an ongoing OSU project. Together with another colleague, we have collaborated to produce an extensive cyber security report as our capstone endeavor. The report seeks to evaluate the security posture of an organization and provide recommendations for improving its measures.

A cyber security team project is a collaborative effort undertaken by a group of cyber security professionals to achieve an objective or goal related to cyber security. This could include improving the security posture of an organization's digital assets, investigating and mitigating a cyber security incident, creating policies and procedures for protecting data, or evaluating and recommending new security technologies or solutions.

Our intention was to replicate a similar environment where our team is able to work as if we were a professional cyber security team. Through different phases we went from learning the materials needed, meeting with cyber security professionals, conducting testing on our project to finally creating this report. It was our hope that we could have our report that follows the industry standard, in particular using the NIST framework and abiding by OSU's security policies.

Project End Goal

Our plan by the end of our capstone project is to have a comprehensive cyber security report detailing how we would conduct our cyber security assessment on a given project, in particular the Oregon State University’s Crafts Center registration page.

Our cybersecurity report summarizes our assessment of a project, outlining any vulnerabilities or threats found during testing. It provides an in-depth evaluation of the security level of the project and suggests ways to strengthen existing measures in order to reduce potential risks.

Our report examines various aspects of the project, such as its technical infrastructure, software, and policies and procedures. Through testing, we identified several vulnerabilities and threats which could potentially compromise security - from database issues, log security and external integrations.

Based on our findings, we offer a detailed set of recommendations for improving the project's security stance. These measures include

Report Introduction

General Cybersecurity Threats

The current state of the digital age has tremendously improved the lives of everyone. In turn our reliance on technology has grown exponentially which results in greater risks. Our sensitive information and livelihoods are built on many different technologies forms apps, websites and computers. Cyber security attacks will typically target three major groups of individuals, business and governments. These threats take on many different forms including ransomware, phishing and malware. Many today will accept new technologies to improve their quality of life and efficiently but comes at the cost of being vulnerable. It's our responsibility to be aware of the dangers and take the needed precautions to be safe.

The digital realm is besieged by a variety of cyber threats that strike via malware or viruses, phishing, or DDoS activity. Advanced techniques are employed by hackers in order to gain access to sensitive material, such as personal data and financial accounts online. Regular vulnerability scans must be carried out by organizations to expose weak points, which could be lack of encryption, insecure configurations of software/hardware and weak passwords. An effective security policy must be in place to ensure safety for organizations.

Importance of Cybersecurity for OSU

Having a robust cybersecurity policy in place is crucial for colleges to ensure the safety of student information and their networks and systems. As more and more students connect to college networks and require the use of numerous digital services, the chances of cyber-attacks and data breaches increase significantly. As a result, implementing cyber security policies are essential. An effective strategy for cyber security may serve multiple purposes for the college community. Beyond simply safeguarding against potential threats, it can also establish clear expectations for all members regarding proper use of technology. Furthermore, implementation of a strong policy should not be overlooked as a means of protecting the institution from potential consequences in the event of cyberattack-induced legal complications or harm to its reputation. To ensure the safety of their campus community, it is imperative for colleges to establish robust cyber security protocols that can effectively ward off potential cyber attacks and safeguard against unauthorized access.

What This Report Covers

This report seeks to provide a comprehensive overview of cyber security threats and vulnerabilities. It will analyze current threats, identify potential security issues, and suggest best practices for mitigating risks.

.

Crafts Team Background

Case Study Overview
As part of our cybersecurity report, we selected the Crafts Team Registration Page project as a case study. The project serves as a case study of how our team would perform a cybersecurity assessment for OSU as security professionals.

What is Oregon State University's Craft Center?
The OSU Craft Center is a student resource center located on the campus of Oregon State University that offers numerous creative activities. Here, students can express their artistic skills and creativity while honing their craft-making abilities. The Craft Center offers well-equipped studios and classes in several disciplines, such as ceramics, glass, woodworking, fibers, pen & paper crafts, jewelry/metals making. It's a resource all OSU students can use at no cost

Crafts Team Registration Page Project
The project at hand is to design a registration system for the OSU Craft Center that addresses general concerns about safeguarding OSU-owned data and abiding by data security standards. This project's primary beneficiaries are students and non-students alike who are interested in becoming members of the Craft Center and signing up for classes. This project's primary goal is to develop a product that allows users to register for classes and store their information online.

Why We Use Crafts Center Project as Case Study
As part of our cybersecurity report, we selected the Crafts Team Registration Page project as a case study as we found it ideal for our report. This online platform enables users to join crafts centers and take part in various crafting projects. Due to its handling of sensitive user data, such as personal and payment info, security for this platform must be of the utmost concern.

The main concern, especially with a student lead team, is that they will focus more on functionality rather than security. Their end goal is to create a working registration system and because of this we fear that they may neglect security concerns that could have disastrous consequences. Our team wants to ensure that with an efficient registration system, they have a secure database that meets OSU's data security regulations.

By using the Crafts Team Registration Page project as a case study, we hope to provide actionable insights and recommendations that can be applied to other similar online platforms seeking to strengthen their cybersecurity practices and safeguard users' sensitive information.

Key Frameworks and Standards

Office of Information Security

The Office of Information Security utilizes six primary security rules in order to effectively create a safe, respectful, and ethical online environment.

OSU's Information Technology Security Rules

Oregon State University follows six cybersecurity rules and policies to safeguard its IT systems and data. The Vulnerability Management Rule mandates an assessment of university IT systems to detect and fix security flaws. The Appropriate Use for System Administrators Rule specifies requirements for system administrators to guarantee their access is conducted ethically and professionally. Likewise, the Log Management Rule regulates log collection, analysis, and retention. The Remote Access Rule governs remote access to university systems and resources, while the Password Management Rule lays out guidelines for password authentication services. Finally, the UIT Email Security Rule specifies principles and practices related to email services provided by Oregon State University. All these regulations apply to individuals associated with Oregon State University.

National Institute of Standards and Technology

The National Institute of Standards and Technology (NIST) Cybersecurity Framework is a comprehensive set of guidelines and best practices designed to assist organizations in managing and reducing cybersecurity risks.

The NIST Cybersecurity Framework is composed of five core functions: Identify, Protect, Detect, Respond and Recover. Each function is further broken down into categories and subcategories to provide guidance on how best to manage cybersecurity risks.

The NIST Cybersecurity Framework is a scalable and adaptable set of guidelines that can be tailored to fit the specific needs of any organization, regardless of size or industry. It's widely recognized as an authoritative framework for cybersecurity risk management and used by organizations around the world to strengthen their cybersecurity position.

Vulnerabilities

8.1 Threat Model

Identification of any possible threats to our client's software available to the public. The threat modeling process will include identifying threats at each level of the web application from users to the backend to cloud services. Proper threat modeling can help identify countermeasures for each attack surface present in a web application environment. The focus of the threat modeling process on the craft team's project will be to identify the most probable threats and attacks to this web application. Below is a list illustrating expected threats at each level starting with users accessing the web application.

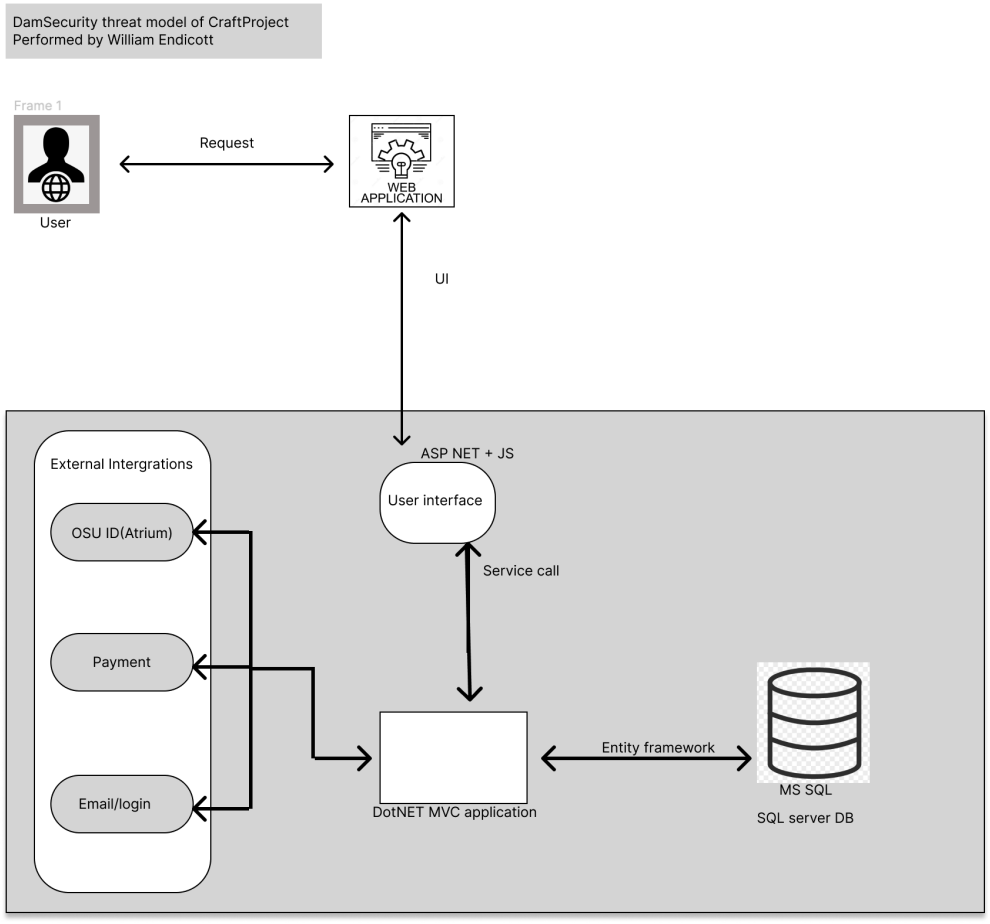

Figure 1. Basic outline of the architecture for the web application, used from the presented documentation and schematic designed by the development team of the craft team's project.

Users

Social engineering remains one of the most damaging aspects to businesses and companies. Simple social engineering can lead to malware, ransomware, leaking of credentials, and loss of digital assets. Given the importance of protecting user and employee data, it is necessary to at least minimize phishing attempts via email, phone, and even physical mail. On OWASP's top ten

list of security risks, the second vulnerability listed is cryptographic failures. Cryptographic failures can lead to insecure communication. The craft team's project will be dealing with the use of users' credit cards, banking information, and shipping addresses. The use of this personal information of the users can lead to a man-in-the-middle attack and subsequently a social engineering attack. Assets at risk in a social engineering attack on the craft team's project include customer data, employee data, availability of service, and payment information.

Web Application

The third listed vulnerability on OWASP's top ten is injections. SQL injection (SQLi) and cross-site scripting (XXS) are the most dangerous and common weaknesses in online applications. DoS (denial of service) attacks and attacks involving authentication methods are also common threats. If a website has weak authentication recommendations for its users, that can lead to a higher-than-normal amount of accounts being compromised.

Identification and authentication failures are listed as seventh on the OWASP's top ten. Insecure communication and platform misconfiguration will also lead to various other attacks. Security misconfiguration is the fifth vulnerability listed on the OWASP's top ten. Having all the security in the world will be useless if the application is misconfigured and vulnerable ports remain open. Assets at risk in an attack on the craft team's project involving the web application include user and employee data, user and employee authentication credentials, and sensitive material that may be stored, including payment information and associated addresses.

Files/Database

Files will have similar threats that are included in the database section. Improper encryption of data while at rest and in transit is the greatest concern for this section. A great majority of the threats from the web application section also apply here as well including SQLi and improper platform configuration. The information provided by users will be stored in the database so unencrypted sensitive data at rest needs to be accounted for since that is a commonly missed aspect of security. Files storing employee credentials, admin credentials, and customer data are the assets at risk.

Logs

Logs are an important part of running a service online. Maintaining proper security logs and event logs will help later identify if someone did something malicious on the website. It would be important to log and maintain a running file of documents that are updated and by who as well as any information that can be obtained from the interaction. If someone uploads a malicious file and an offensive file we would want the ability to find that person and hold them responsible. This is where one of the key pillars of information security comes into play. Non-repudiation will allow the administrators of the website to log the sender's information that is provided from the network traffic as well as the credentials associated with the traffic.

External Integrations

This will handle the login credentials for people accessing the product. Insecure communication is a priority to identify. Risk of attack methods on two-factor authentication are seen at this level. Those include but are not limited to SMS-based man-in-the-middle attacks, pass-the-cookie attacks, and social engineering attacks from the use of 2FA. Overall, this is a service that will be handled outside of the project team's hands but it will need to be checked for the correct configuration.

8.2 CVE's

This section will be used to focus on real world examples with common vulnerabilities and exposures(CVE) that are identified on other websites registration pages. This will provide guidance and be used as a reference when inspecting the Craft's team project. The examples will provide direct comparison and something to use as guidance to help improve the Craft team to the next tier of security.

The following CVE examples show what the company will look for as a reference. The tool discussed later will auto populate with CVE's that directly relate to the application. The automated tool is using the CVE's to show exactly what vulnerabilities exist. This comes with the CVE number which can be documented for expedited mitigation or for security awareness. This gives the staff the proper knowledge to make decisions about how to react if a security incident were to occur.

The two links included will include two separate vulnerabilities from two different companies. These are examples that a company uses as reference of what not to do. The two examples are of XSS(cross site scripting) and data not being encrypted. The non encrypted data is important and a high risk which could leak sensitive information to a bad actor. Simply this would allow someone on the same network to take whatever information is sent and read it themselves. This data could include name, address, phone number, email, credit card information, and login credentials which is dangerous due to the fact that more than likely that person uses the same credentials for other accounts they are associated with.

The second example of a CVE is XSS which allows for malicious actors to inject scripts or malicious code into the unexpecting website. This would allow for the attacker to run scripts on the users computers thinking that it's coming from a trusted source. This would give access to cookies, session tokens, or other sensitive information.
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2022-43097
https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2022-0232

8.3 Tools

The tools used to do basic reconnaissance and identification of possible vulnerabilities will be discussed in this section. We used three tools during the process of evaluating this project. The tools we used include NMAP, tcpdump, wireshark, and Sn1per. Each one of these tools is used for different purposes but they are closely related in a sense that they're for reconnaissance and less intrusive than other tools available. The order in which they will be covered is NMAP, wireshark, and then finally Sn1per. Each one of the tools below will be used on example websites since we don't have the authority to test on an actual OSU service.

Reconnaissance

Reconnaissance is one of the first five phases listed on the ec-council's website for phases of penetration testing. They describe this phase as gathering as much information about a target system as possible. This will include network topology, the operating system the target is on, applications, and user accounts. As someone new to ethical hacking it's easy to see the importance of this step. Afterall, it is better to know the target to figure out what tools we can use on the target's system.

Passive Reconnaissance

The ec-council talks about the two categories of reconnaissance. The first is passive and this could consist of publicly available information about a target. This could include gathering domain names, email addresses, IP addresses, and much more. In an article written by Shimon Brathwaite, titled "Active vs Passive Cyber Reconnaissance in Information Security." Brathwaite gives passive reconnaissance a familiar name I have heard before. OSINT is short for open-source intelligence. Brathwaite lists common tools used in the process of gaining intelligence on a target. The three listed include Google Hacking, Netcraft, and Shodan but many more are available.

Active Reconnaissance

Active reconnaissance requires interacting with the target to gain information. This will require a set of tools to scan a network to find out information. NMAP which is short for network mapper is used to scan systems giving various information on a target. NMAP is a focus for this section and the tool we used to scan target systems and will be discussed in detail shortly. Active reconnaissance has the chance of being detected by the target.

The information gathered at this part of reconnaissance includes but is not limited to finding out if a port is opened or closed, the operating system the machine is using, the services the target is running, and discovering if the host has vulnerable applications or ports. Since this is intrusive it is vital as mentioned before not to scan anything we do not own or have permission to scan. Scanning a network can trigger intrusion detection systems and intrusion prevention systems.

NMAP (network mapping)

NMAP is the first tool introduced for our journey of reconnaissance. This tool has many features and capabilities, few of which will be discussed here. The first part of understanding nmap will begin with scanning scanme.nmap.org checking for its operating system, version, script, and traceroute. In figure 1 below the results of our first scan gave me great details about the target. This gives a great overview of the target with many details that can be a little overwhelming at first glance.

```
┌──(root㉿kali)-[~]
└─# nmap -A -T4 scanme.nmap.org
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-06 13:58 PDT
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.0085s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 992 closed tcp ports (reset)
PORT      STATE    SERVICE      VERSION
22/tcp    open     ssh          OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 ac:00:a0:1a:82:ff:cc:55:99:dc:67:2b:34:97:6b:75 (DSA)
|   2048 20:3d:2d:44:62:2a:b0:5a:9d:b5:b3:05:14:c2:a6:b2 (RSA)
|   256 96:02:bb:5e:57:54:1c:4e:45:2f:56:4c:4a:24:b2:57 (ECDSA)
|_  256 33:fa:91:0f:e0:e1:7b:1f:6d:05:a2:b0:f1:54:41:56 (ED25519)
25/tcp    filtered smtp
80/tcp    open     http         Apache httpd 2.4.7 ((Ubuntu))
|_http-title: Go ahead and ScanMe!
|_http-favicon: Nmap Project
|_http-server-header: Apache/2.4.7 (Ubuntu)
135/tcp   filtered msrpc
139/tcp   filtered netbios-ssn
445/tcp   filtered microsoft-ds
9929/tcp  open     nping-echo   Nping echo
31337/tcp open     tcpwrapped
Aggressive OS guesses: Linux 3.2 (94%), Linux 4.4 (94%), DD-WRT v24-sp2 (Linux 2.4.37) (92%), Ac
tiontec MI424WR-GEN3I WAP (92%), Microsoft Windows XP SP3 or Windows 7 or Windows Server 2012 (9
0%), Microsoft Windows XP SP3 (90%), BlueArc Titan 2100 NAS device (87%), TiVo series 1 (Sony SV
R-2000 or Philips HDR112) (Linux 2.1.24-TiVo-2.5, PowerPC) (87%), TiVo series 1 (Linux 2.1.24-Ti
Vo-2.5) (86%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 80/tcp)
HOP RTT     ADDRESS
1   0.53 ms 172.16.141.2
2   0.53 ms scanme.nmap.org (45.33.32.156)

OS and Service detection performed. Please report any incorrect results at https://nmap.org/subm
it/ .
Nmap done: 1 IP address (1 host up) scanned in 30.44 seconds
```

Figure 2. nmap scan of scanme.nmap.org

NMAP options

The journey of nmap and learning its options will be long and take some effort and will expand much further than this class. Let's start to explore the more specific options available within nmap. First, we are learning how to scan for TCP and UDP of specific ports. One issue I encountered while trying to scan certain ports was failing to include -p <ports> I instead attempted to clump the port numbers into the port scan I was attempting. So, in the case where I was scanning for TCP connect and wanted to scan ports 20-100 I tried #nmap -sT20-100 scanme.nmap.org. This was the first example of me not reading the man pages correctly.

```
┌──(root㉿kali)-[~]
└─# nmap -sT -p 20-100,130-150,400-500 scanme.nmap.org
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-06 14:24 PDT
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.042s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 194 closed tcp ports (conn-refused)
PORT    STATE    SERVICE
22/tcp  open     ssh
25/tcp  filtered smtp
80/tcp  open     http
135/tcp filtered msrpc
136/tcp filtered profile
137/tcp filtered netbios-ns
138/tcp filtered netbios-dgm
139/tcp filtered netbios-ssn
445/tcp filtered microsoft-ds

Nmap done: 1 IP address (1 host up) scanned in 2.65 seconds
```

Figure 3. TCP connect scan of scanme.nmap.org

The next two individual actions I ran with nmap are detection of the operating system and IP protocol scan. Each of which are important. Figure 3 shows the result of the separate scans but in the same screenshot. In a video about using nmap created by Simplilearn the narrator explains and walks through about how to use nmap to find that port 445 is open. He explains that the eternal blue exploit might be a vulnerability that could be used and showed how that worked. This gave concrete evidence of how important nmap can be to scan our system as a Whitehat to ensure we fix vulnerabilities.

```
┌──(root㉿kali)-[~]
└─# nmap -O scanme.nmap.org
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-06 20:59 PDT
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.043s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 992 closed tcp ports (reset)
PORT      STATE    SERVICE
22/tcp    open     ssh
25/tcp    filtered smtp
80/tcp    open     http
135/tcp   filtered msrpc
139/tcp   filtered netbios-ssn
445/tcp   filtered microsoft-ds
9929/tcp  open     nping-echo
31337/tcp open     Elite
Aggressive OS guesses: Linux 3.2 (94%), Linux 4.4 (94%), DD-WRT v24-sp2 (Linux 2.4.37) (92%), Ac
tiontec MI424WR-GEN3I WAP (92%), Microsoft Windows XP SP3 (89%), Microsoft Windows XP SP3 or Win
dows 7 or Windows Server 2012 (89%), BlueArc Titan 2100 NAS device (86%), TiVo series 1 (Sony SV
R-2000 or Philips HDR112) (Linux 2.1.24-TiVo-2.5, PowerPC) (86%), TiVo series 1 (Linux 2.1.24-Ti
Vo-2.5) (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 16 hops

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 10.34 seconds

┌──(root㉿kali)-[~]
└─# nmap -sO scanme.nmap.org
Starting Nmap 7.92 ( https://nmap.org ) at 2022-10-06 20:59 PDT
Nmap scan report for scanme.nmap.org (45.33.32.156)
Host is up (0.015s latency).
Other addresses for scanme.nmap.org (not scanned): 2600:3c01::f03c:91ff:fe18:bb2f
Not shown: 252 filtered n/a protocols (proto-unreach)
PROTOCOL STATE         SERVICE
1        open          icmp
6        open          tcp
17       open          udp
47       open|filtered gre

Nmap done: 1 IP address (1 host up) scanned in 1.84 seconds
```

Figure 4. nmap OS detection scan on the top and IP protocol scan on the bottom.

Packet capture

Capturing packets can be accomplished in multiple ways. This includes direct access to the channel of the network that includes ethernet cables, copper wire, and RF receiver. Having direct access can use different tools including a network tap, switches, and setting up a computer with dual ethernet ports to mirror a switch and capture packets as they travel through the machine. Another method is using a Wireless access point to capture packets with a WiFi card installed and set to promiscuous mode. The last method is having control of an endpoint This is the more practical case and the one we will begin investigating.

Our overall goal of capturing packets can range from checking the quality of service to checking for malicious activity that is happening on the network. The malicious actors can use it for eavesdropping on data on the network all the way to espionage. This is a key place where having physical security is important to ensure no one has access to routers, network jacks, and endpoints. Just like many topics we have covered and will cover we can't tap or eavesdrop on any network that we do not own.

Tcpdump

Tcpdump is one of the command line tools that is a foundational tool used to capture packets. This can be configured in many ways to capture in and outbound packets or both at the same time. This can be done while writing to a file to later filter the results in ways that best fits our needs. We will be going over different commands explained in this module. The first command I entered to play with was 'tcpdump' this results in immediate results being displayed to the console. This produces results at such a rapid rate that it can be overwhelming to understand what is being displayed. Plus, we don't have a way to narrow down the current capture since the results aren't going to a file.

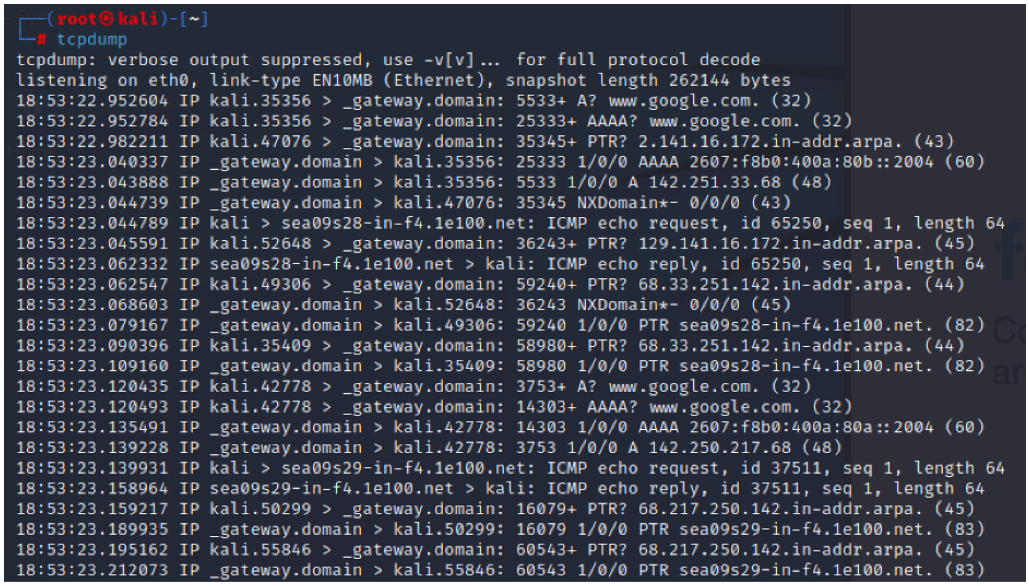

Figure 5. result of entering just tcpdump with no flags or extra options or file being created.

The next command we will use will include a method to write the results to a file name of our choosing. This time I will be first_capture.pcap. To do this we will include the flag -w after our tcpdump command. Figure two will have the results of the command line being entered. The image displays the number of packets captured as well as the packets received by the filter. If those first two numbers are different, then that means the difference in the value of packets that were received but not processed before exiting tcpdump. The last output will include the number of packets dropped by the kernel. In our instance it was zero.

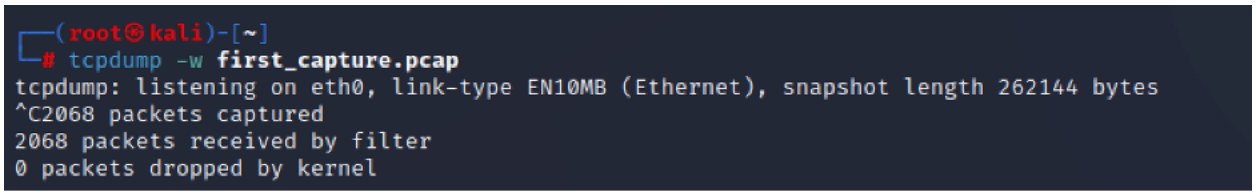

Figure 6. command line directs the packets to a readable file.

Viewing results

Viewing the results of the packets can be done by adding the flag for reading, followed by the file name, and then if we would like a verbose version of the results. Then ending with a pipe and the number of lines we would like to display for a small set of data from the file. Figures three and four will display the results we received from the first packets we captured. Figure three will be without verbose and figure four will include the verbose flag. As we can see while I was capturing packets, I went to Facebook. We can also see what method of communication we used and the size of each packet.

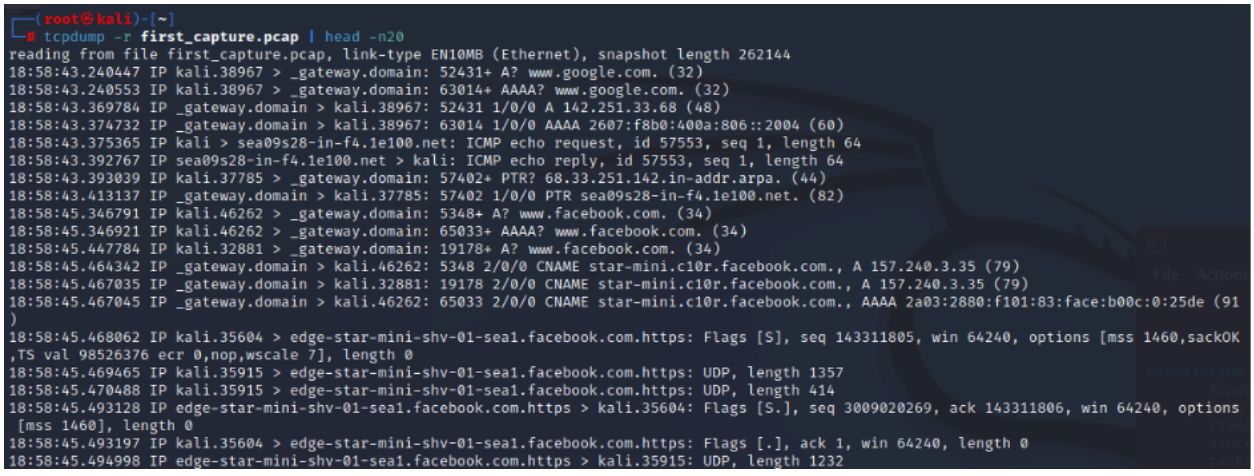

Figure 7. without viewing 20 lines without the verbose flag

```
┌──(root㉿kali)-[~]
└─# tcpdump -r first_capture.pcap  -v | head -n20
reading from file first_capture.pcap, link-type EN10MB (Ethernet), snapshot length 262144
18:58:43.240447 IP (tos 0x0, ttl 64, id 9454, offset 0, flags [DF], proto UDP (17), length 60)
    kali.38967 > _gateway.domain: 52431+ A? www.google.com. (32)
18:58:43.240553 IP (tos 0x0, ttl 64, id 9455, offset 0, flags [DF], proto UDP (17), length 60)
    kali.38967 > _gateway.domain: 63014+ AAAA? www.google.com. (32)
18:58:43.369784 IP (tos 0x0, ttl 128, id 30384, offset 0, flags [none], proto UDP (17), length 76)
    _gateway.domain > kali.38967: 52431 1/0/0 www.google.com. A 142.251.33.68 (48)
18:58:43.374732 IP (tos 0x0, ttl 128, id 30385, offset 0, flags [none], proto UDP (17), length 88)
    _gateway.domain > kali.38967: 63014 1/0/0 www.google.com. AAAA 2607:f8b0:400a:806::2004 (60)
18:58:43.375365 IP (tos 0x0, ttl 64, id 11862, offset 0, flags [DF], proto ICMP (1), length 84)
    kali > sea09s28-in-f4.1e100.net: ICMP echo request, id 57553, seq 1, length 64
18:58:43.392767 IP (tos 0x0, ttl 128, id 30386, offset 0, flags [none], proto ICMP (1), length 84)
    sea09s28-in-f4.1e100.net > kali: ICMP echo reply, id 57553, seq 1, length 64
18:58:43.393039 IP (tos 0x0, ttl 64, id 54990, offset 0, flags [DF], proto UDP (17), length 72)
    kali.37785 > _gateway.domain: 57402+ PTR? 68.33.251.142.in-addr.arpa. (44)
18:58:43.413137 IP (tos 0x0, ttl 128, id 30387, offset 0, flags [none], proto UDP (17), length 110)
    _gateway.domain > kali.37785: 57402 1/0/0 68.33.251.142.in-addr.arpa. PTR sea09s28-in-f4.1e100.net. (82)
18:58:45.346791 IP (tos 0x0, ttl 64, id 36936, offset 0, flags [DF], proto UDP (17), length 62)
    kali.46262 > _gateway.domain: 5348+ A? www.facebook.com. (34)
18:58:45.346921 IP (tos 0x0, ttl 64, id 36937, offset 0, flags [DF], proto UDP (17), length 62)
    kali.46262 > _gateway.domain: 65033+ AAAA? www.facebook.com. (34)
```

Figure 8. Results with verbose mode.

The image will include the results of capture packets going out from the second packet capture and the results of packets coming in from the third capture. For each of these, I included an even more verbose flag to show just how detailed we can get with these results. We can monitor whatever direction we desire or both directions at the same time. Figure five shows the results of in and out traffic independently.

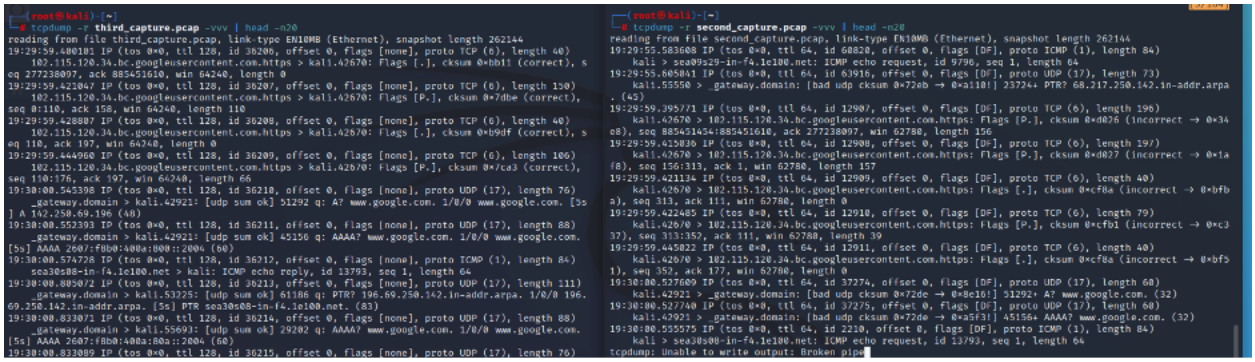

Figure 9. In traffic on the left and out traffic on the right.

Tcpdump filters

Next, we have the filters to narrow down the specific information we are looking for. We can filter to view UDP or TCP, what port we want, and the port range. The filter can even include Boolean searches to include AND, OR, and even NOT. This is where things become interesting and very specific and helps when we have a goal in mind. The following images will include examples of the power we have with filters. The first filter I picked shows data from only port 53 which is DNS. I also included a few new flags I haven't discussed which are -c5, -X, -K. The first one is to only show a count of 5 results. The X flag is used to print the hex and ascii dump and K flag not to validate the checksum.

```
┌──(root㉿kali)-[~]
└─# tcpdump -c5 -X -K -vvv -r third_capture.pcap port 53
reading from file third_capture.pcap, link-type EN10MB (Ethernet), snapshot length 262144
20:23:09.257522 IP (tos 0x0, ttl 64, id 7064, offset 0, flags [DF], proto UDP (17), length 65)
    kali.57908 > _gateway.domain: 29759+ A? static.xx.fbcdn.net. (37)
        0x0000:  4500 0041 1b98 4000 4011 ac6f ac10 8d81  E..A..@.@..o....
        0x0010:  ac10 8d02 e234 0035 002d 72e3 743f 0100  .....4.5.-r.t?..
        0x0020:  0001 0000 0000 0000 0673 7461 7469 6302  .........static.
        0x0030:  7878 0566 6263 646e 036e 6574 0000 0100  xx.fbcdn.net....
        0x0040:  01                                       .
20:23:09.257583 IP (tos 0x0, ttl 64, id 7065, offset 0, flags [DF], proto UDP (17), length 65)
    kali.57908 > _gateway.domain: 51394+ AAAA? static.xx.fbcdn.net. (37)
        0x0000:  4500 0041 1b99 4000 4011 ac6e ac10 8d81  E..A..@.@..n....
        0x0010:  ac10 8d02 e234 0035 002d 72e3 c8c2 0100  .....4.5.-r.....
        0x0020:  0001 0000 0000 0000 0673 7461 7469 6302  .........static.
        0x0030:  7878 0566 6263 646e 036e 6574 0000 1c00  xx.fbcdn.net....
        0x0040:  01                                       .
```

Figure 10. The first example of filters. This shows DNS packet capture with hex and ascii dump.

Figure seven shows the example of filtering for just a port range. This will show the results of all results if they are included in that range. We can further narrow the search down by adding tcp/udp of a set of selected ports as well as a src and dst. This module shows a vast array of different options used to capture packets. Then after capturing the packets, we have the tools to filter in whatever manner we need. The skill to manipulate the results seems endless. The next step is learning to use the data in a way to help protect our network.

```
┌──(root㉿kali)-[~]
└─# tcpdump -c10 -X -K -vvv -r third_capture.pcap portrange 20000-35000
reading from file third_capture.pcap, link-type EN10MB (Ethernet), snapshot length 262144
20:23:09.645665 IP (tos 0x0, ttl 64, id 0, offset 0, flags [DF], proto UDP (17), length 1385)
    kali.33308 > edge-star-mini-shv-01-sea1.facebook.com.https: UDP, length 1357
        0x0000:  4500 0569 0000 4000 4011 5adf ac10 8d81  E..i..@.@.Z.....
        0x0010:  9df0 0323 821c 01bb 0555 e00b cf00 0000  ...#.....U......
        0x0020:  0108 2947 aa9f af8a 60c2 037a 0cbe 3c9e  ..)G....`..z..<.
        0x0030:  242e db11 df46 514f e2c0 7047 a59b 1b1d  $....FQO..pG....
        0x0040:  9592 ebe2 a5ad 9b57 f008 1c36 0dab ac00  .......W...6....
        0x0050:  0000 0076 41f9 7ab9 0c88 8c02 e124 1eb6  ...vA.z......$..
        0x0060:  e574 c55d 3aa5 d798 2580 3342 3202 5d79  .t.]:...%.3B2.]y
        0x0070:  7d3c 6267 c84d c546 7997 ec56 14be 16da  }<bg.M.Fy..V....
        0x0080:  b0c2 84e7 1a33 16bc 99c8 10ee a0e5 8c48  .....3.........H
        0x0090:  4276 6df9 8214 d123 76a2 8c61 a028 6d97  Bvm....#v..a.(m.
        0x00a0:  a5bd e69f bf5a 75ae 059e 0184 2554 9705  .....Zu.....%T..
        0x00b0:  cbcc e1c5 bc75 92ee 5c79 2191 bbaf b3eb  .....u..\y!.....
```

Figure 11. filter showing just results from port 20000-35000

Next tool used in the process is Sn1per, the ultimate “all-in-one” offensive security framework as stated by the creators. This is one of the most powerful tools on the market. In this instance we utilized the community version that is free for all to use. This covers the full attack surface to discover what vulnerabilities exist and generates a graded report at the end with a score at the end of the scan. This will include a list of levels for vulnerabilities and how many exist including low, medium, high and critical. Figure 12 is the architecture of Sn1per and each sub category it uses to test the target system.

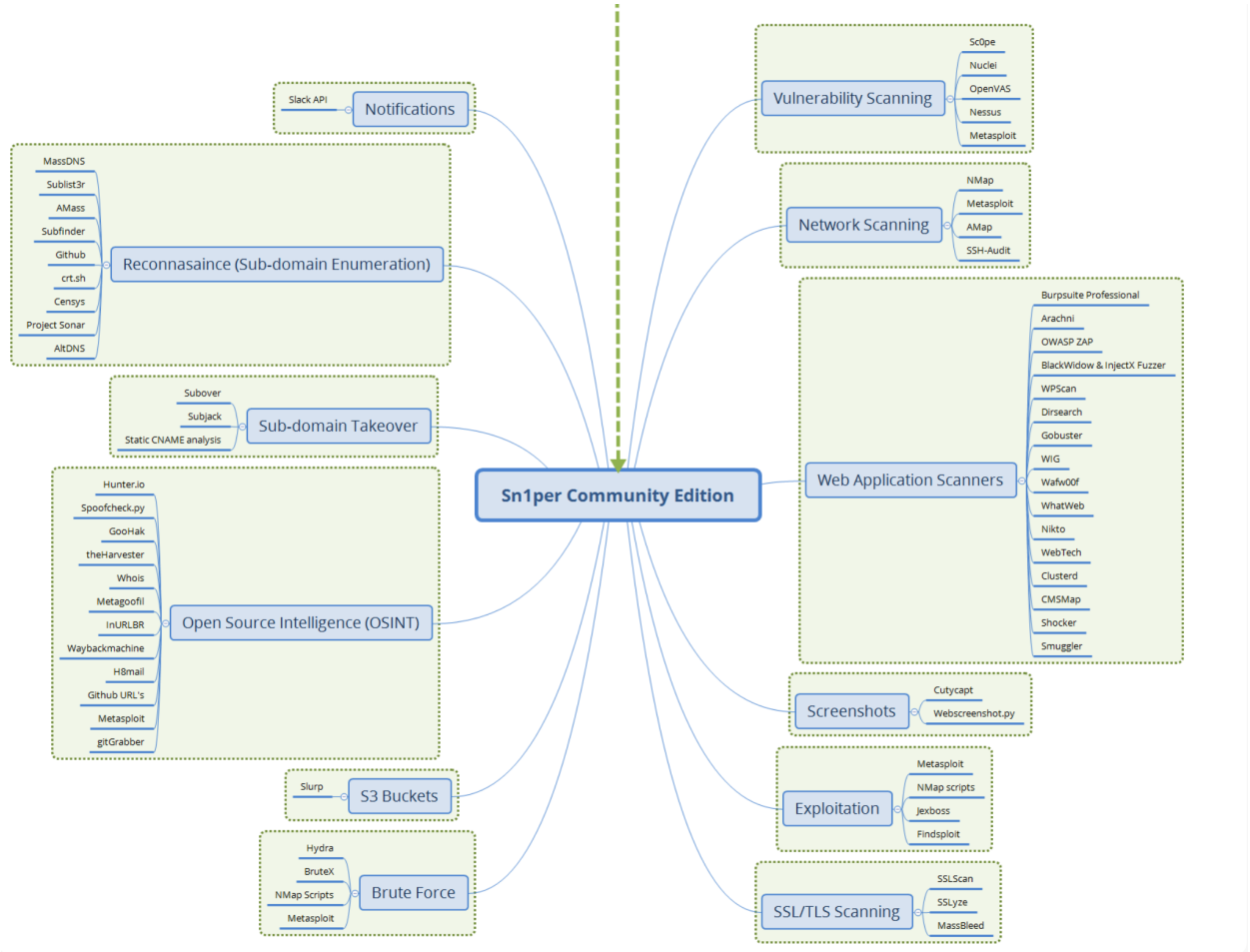

Figure 12. Sn1per architecture Diagram for community edition. Professional version was cropped out.

First, Checking for a web application firewall and identifying if it's configured correctly is important. Many web applications pay for firewalls and think they have proper security set up. In most cases they have the firewalls configured incorrectly thus being vulnerable to attacks. Sn1per can identify this so allow for proper configuration before attempted attacks. Image 14, 15 and 16 include the vulnerabilities and CVE's identified for the target. The vulnerability report with a score can be seen in figure 16.

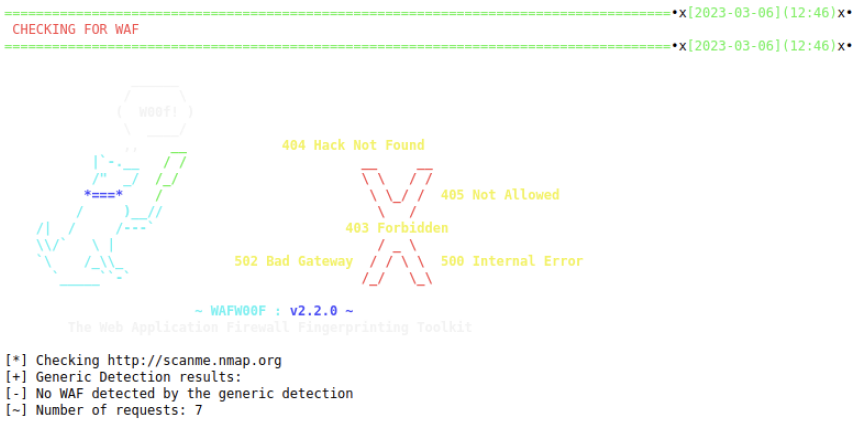

Figure 13. Results checking for a web application firewall.

Sn1pers next item that is being checked is the ports that are open. The results below show the vulnerabilities that are associated with port 80. The CVE that closely relates to the vulnerabilities is also listed for each reference. As Discussed above the CVE's are a great way to reference what configuration shouldn't be in the wild. The steps taken next would be to search each of these CVE's and document the risk they bring to the application they were reported from. This would give a good base line of what to look for and how to mitigate and have better security features. Basically it's a way to learn from other companies' mistakes. Some CVE's could be as simple as updating the version of whatever software is being used.

```
PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.7 ((Ubuntu))
|_http-server-header: Apache/2.4.7 (Ubuntu)
| vulners:
|   cpe:/a:apache:http_server:2.4.7:
|       CVE-2022-31813  7.5     https://vulners.com/cve/CVE-2022-31813
|       CVE-2022-23943  7.5     https://vulners.com/cve/CVE-2022-23943
|       CVE-2022-22720  7.5     https://vulners.com/cve/CVE-2022-22720
|       CVE-2021-44790  7.5     https://vulners.com/cve/CVE-2021-44790
|       CVE-2021-39275  7.5     https://vulners.com/cve/CVE-2021-39275
|       CVE-2021-26691  7.5     https://vulners.com/cve/CVE-2021-26691
|       CVE-2017-7679   7.5     https://vulners.com/cve/CVE-2017-7679
|       CVE-2017-3167   7.5     https://vulners.com/cve/CVE-2017-3167
|       CNVD-2022-73123 7.5     https://vulners.com/cnvd/CNVD-2022-73123
|       CNVD-2022-03225 7.5     https://vulners.com/cnvd/CNVD-2022-03225
|       CNVD-2021-102386        7.5     https://vulners.com/cnvd/CNVD-2021-102386
|       PACKETSTORM:127546      6.8     https://vulners.com/packetstorm/PACKETSTORM:127546      *EXPLOIT*
|       FDF3DFA1-ED74-5EE2-BF5C-BA752CA34AE8    6.8     https://vulners.com/githubexploit/FDF3DFA1-ED74-5EE2-BF5C-BA752CA34AE8  *EXPLOIT*
|       CVE-2021-40438  6.8     https://vulners.com/cve/CVE-2021-40438
|       CVE-2020-35452  6.8     https://vulners.com/cve/CVE-2020-35452
|       CVE-2018-1312   6.8     https://vulners.com/cve/CVE-2018-1312
|       CVE-2017-15715  6.8     https://vulners.com/cve/CVE-2017-15715
|       CVE-2016-5387   6.8     https://vulners.com/cve/CVE-2016-5387
|       CVE-2014-0226   6.8     https://vulners.com/cve/CVE-2014-0226
|       CNVD-2022-03224 6.8     https://vulners.com/cnvd/CNVD-2022-03224
|       8AFB43C5-ABD4-52AD-BB19-24D7884FF2A2    6.8     https://vulners.com/githubexploit/8AFB43C5-ABD4-52AD-BB19-24D7884FF2A2  *EXPLOIT*
|       4810E2D9-AC5F-5B08-BFB3-DDAFA2F63332    6.8     https://vulners.com/githubexploit/4810E2D9-AC5F-5B08-BFB3-DDAFA2F63332  *EXPLOIT*
|       4373C92A-2755-5538-9C91-0469C995AA9B    6.8     https://vulners.com/githubexploit/4373C92A-2755-5538-9C91-0469C995AA9B  *EXPLOIT*
|       1337DAY-ID-22451        6.8     https://vulners.com/zdt/1337DAY-ID-22451        *EXPLOIT*
|       0095E929-7573-5E4A-A7FA-F6598A35E8DE    6.8     https://vulners.com/githubexploit/0095E929-7573-5E4A-A7FA-F6598A35E8DE  *EXPLOIT*
```

Figure 14. Port scan similar to the NMAP one we did manually above.

Identifying the risk level associated with each of the listed CVE's is important. Sn1per provides an automated tool to do this as well. This gives the company the opportunity to fix vulnerabilities with a high severity level. Part of having a security policy in place is maintaining the best overall security which means sometimes not every aspect of security will be covered. Documenting and maintaining a log of what vulnerabilities exist even if they are not corrected immediately helps with security. The company will have the opportunity to create a mitigation plan that involves responding to a breach from an existing security risk. This specific tool helps fast track that process and again the CVE's are attached for easy reference.

```
================================================================================•x[2023-03-06](12:49)x•
 RUNNING SCOPE NETWORK VULNERABILITY SCAN
================================================================================•x[2023-03-06](12:49)x•
P4 - LOW, SSH Version Disclosure, scanme.nmap.org, [+] 45.33.32.156:22       - SSH server version: SSH-2.0-OpenSSH_6.6.1p1 Ubuntu-2ubuntu2.
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2015-5600 8.5 https://vulners.com/cve/CVE-2015-5600
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2015-6564 6.9 https://vulners.com/cve/CVE-2015-6564
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2018-15919 5.0 https://vulners.com/cve/CVE-2018-15919
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2021-41617 4.4 https://vulners.com/cve/CVE-2021-41617
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2020-14145 4.3 https://vulners.com/cve/CVE-2020-14145
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2015-5352 4.3 https://vulners.com/cve/CVE-2015-5352
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2015-6563 1.9 https://vulners.com/cve/CVE-2015-6563
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2022-31813 7.5 https://vulners.com/cve/CVE-2022-31813
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2022-23943 7.5 https://vulners.com/cve/CVE-2022-23943
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2022-22720 7.5 https://vulners.com/cve/CVE-2022-22720
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2021-44790 7.5 https://vulners.com/cve/CVE-2021-44790
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2021-39275 7.5 https://vulners.com/cve/CVE-2021-39275
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2021-26691 7.5 https://vulners.com/cve/CVE-2021-26691
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2017-7679 7.5 https://vulners.com/cve/CVE-2017-7679
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2017-3167 7.5 https://vulners.com/cve/CVE-2017-3167
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CNVD-2022-73123 7.5 https://vulners.com/cnvd/CNVD-2022-73123
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CNVD-2022-03225 7.5 https://vulners.com/cnvd/CNVD-2022-03225
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CNVD-2021-102386 7.5 https://vulners.com/cnvd/CNVD-2021-102386
```

Figure 15. Risk level for the vulnerabilities found.

Finally, the overall score of the application will be presented with a tiered numbering of each vulnerability. Info all the way to critical vulnerabilities will be counted and reported. This will populate a score for the vulnerability report and it's added at the end. This gives the clear and

precise ranking of what the current security is at. This will also lay a foundation for working towards the next tier of security readiness which will be discussed later.

```
================================================================================
•?((¯°·..• Sc0pe Vulnerability Report by @xer0dayz •._.·°¯))؟•
================================================================================
Critical: 0
High: 1
Medium: 95
Low: 1
Info: 3
Score: 294
================================================================================
P2 - HIGH, Clear-Text Protocol - HTTP, http://scanme.nmap.org:80/, HTTP/1.1 200 OK
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2015-5600 8.5 https://vulners.com/cve/CVE-2015-5600
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2015-6564 6.9 https://vulners.com/cve/CVE-2015-6564
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2018-15919 5.0 https://vulners.com/cve/CVE-2018-15919
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2021-41617 4.4 https://vulners.com/cve/CVE-2021-41617
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2020-14145 4.3 https://vulners.com/cve/CVE-2020-14145
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2015-5352 4.3 https://vulners.com/cve/CVE-2015-5352
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2015-6563 1.9 https://vulners.com/cve/CVE-2015-6563
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2022-31813 7.5 https://vulners.com/cve/CVE-2022-31813
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2022-23943 7.5 https://vulners.com/cve/CVE-2022-23943
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2022-22720 7.5 https://vulners.com/cve/CVE-2022-22720
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2021-44790 7.5 https://vulners.com/cve/CVE-2021-44790
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2021-39275 7.5 https://vulners.com/cve/CVE-2021-39275
P3 - MEDIUM, Components with Known Vulnerabilities - NMap, scanme.nmap.org, CVE-2021-26691 7.5 https://vulners.com/cve/CVE-2021-26691
```

Figure 16. Overall score of vulnerabilities found.(lower is better)

Recommendations

9.1 Overview of security

Creating a secure website involves a couple of key steps. First, choose a host that is reliable and provides reliable support. Make sure the host has up-to-date security protocols and services. Second, use secure coding techniques and tools. Choose an encryption scheme for your website, and make sure your passwords are not easily guessable. Third, regularly monitor the website for suspicious activity and update your website's security. Finally, use a service such as Google's reCAPTCHA to protect the website from malicious bots. By following these steps, you can ensure your website is secure and protected from malicious attacks. This web site falls under OSU scope so the tier of security would rate higher than that of a stand alone site.

9.2 Getting to the next tier

The framework of security would introduce a tiered level of security that a company can work towards improving. The company can start for example at tier 1 of the 4 tiers and their next goal would be to achieve tier 2. This gives a step by step plan to reaching a higher level of tiers at some point but not necessarily forcing the company to jump from 1 to 4. This could cause errors and stress on the security system being implemented. The company would know exactly what they need to get to the next tier. The 4 tiers are from lowest to highest Tier 1 partial, 2 risk informed, 3 repeatable, and finally 4 is adaptive.

9.3 Documenting vulnerabilities

Documenting vulnerabilities and maintaining a proper log of misconfigurations, vulnerabilities, and attempted attacks would be important. This would create an easy to understand and easy to access format of information for those responsible for security to reference during their time of need. These would help create mitigation plans for assets that are especially vulnerable. Having a quick plan of action could make the difference between one computer being compromised and a whole network of computers, software, and data.

Conclusion:

Cybersecurity threats continue to be a serious risk for all organizations especially as our dependence on technology becomes greater. As businesses and universities adopt more technologies to maintain a competitive edge it's crucial for these organizations to maintain robust security infrastructure and policies. Cyberattacks and data breaches that occur on a university wide scale would be detrimental for a school's members and their reputation. This report aims to be an extra security layer by giving valuable insights and spreading awareness. Helping the university stay ahead of any possible risks or threats ensures the safety of all students and maintaining the trust and respect that an organization has built up.