Fortinet NSE5_FWB_AD-8.0 NSE 5 Practice Questions

Shared by Orr on 17-08-2026

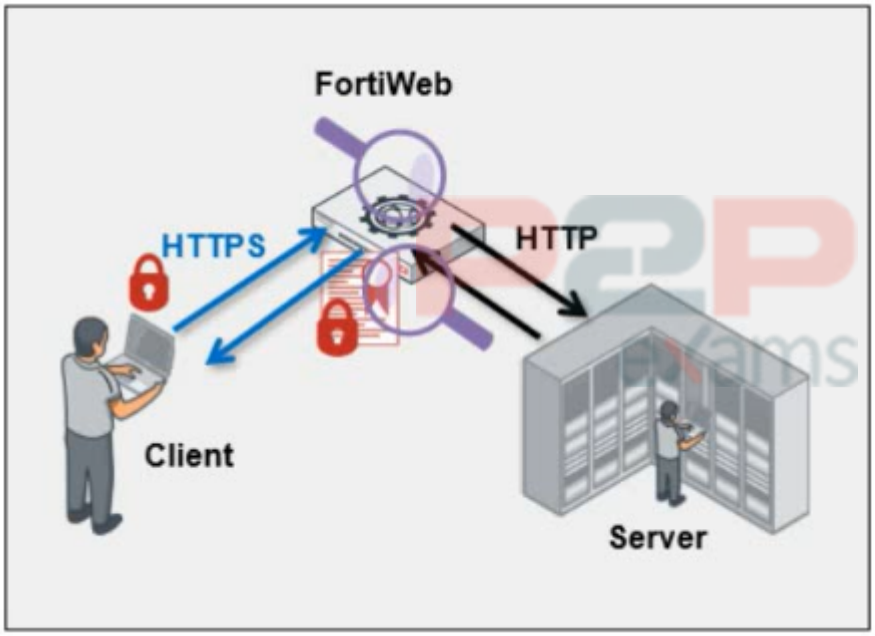
For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

Refer to the exhibit.



You are configuring SSL offloading on FortiWeb to protect a public-facing application. Clients connect using HTTPS, while FortiWeb forwards requests to the back-end server using HTTP.

You are reviewing certificate deployment and need to decide where to install the private key for the certificate used in client connections.

In this SSL offloading setup, which device is responsible for using the private key associated with the web server certificate?

Options:

A- FortiWeb, because it terminates the HTTPS session and decrypts traffic.
B- None. SSL offloading does not require a private key because FortiWeb only forwards traffic.
C- The server, because it always handles certificates regardless of SSL mode.
D- The client, because it initiates the TLS handshake and verifies the certificate.

Answer:

A

Explanation:

In SSL offloading, FortiWeb is the TLS endpoint for client connections. The client negotiates

HTTPS with FortiWeb, not directly with the back-end web server. Therefore, FortiWeb must have the website certificate and associated private key so it can complete the TLS handshake, decrypt inbound HTTPS traffic, inspect the HTTP content, and then forward the request to the server using HTTP or a separate back-end connection. Option B is wrong because TLS termination requires the private key. Option C describes SSL inspection or direct server termination, not offloading. Option D is wrong because clients verify the certificate but do not possess or use the server's private key. FortiWeb owns the private-key function in this design.

Question 2

Question Type: MultipleChoice

FortiWeb is blocking groups of users behind your load balancer. In the logs, all users show the same source IP address.

Which action should you take to restore proper client identification?

Options:

A- Add a bot detection rule in the protection profile.
B- Update the signature engine.
C- Reconfigure the load balancer to insert the original client IP address in an HTTP header.
D- Enable caching for HTTPS traffic.

Answer:

C

Explanation:

When FortiWeb is deployed behind a load balancer or upstream proxy, it may see only the load balancer IP address as the source for every request. This causes poor client identification and can lead FortiWeb to block many legitimate users as if they are one abusive client. The correct fix is to preserve the original client IP address in an HTTP header, commonly using X-Forwarded-For or a similar trusted client-IP header. FortiWeb can then be configured to read that header for accurate client identification, logging, rate limiting, and IP-based security decisions. Bot detection, signature updates, and HTTPS caching do not solve the core source-IP visibility problem.

==========

Question 3

Question Type: MultipleChoice

Which URL should you rewrite to reduce security risk?

Options:

A- https://www.example.com/about/team
B- https://www.example.com/wordpress/?feed=rss2
C- https://www.example.com/products/today
D- https://www.example.com/25.3.6/Browse/MediaData

Answer:

D

Explanation:

The URL https://www.example.com/25.3.6/Browse/MediaData exposes internal application structure and what appears to be a version number. Revealing version information, framework paths, or internal directory names gives attackers useful reconnaissance data. Attackers can correlate exposed versions with known vulnerabilities and target the application more precisely. FortiWeb URL rewriting can reduce this risk by hiding or transforming sensitive internal URLs before users or scanners see them. The other URLs are normal-looking public paths or common application resources. A WordPress RSS feed may or may not be appropriate depending on business requirements, but it does not clearly expose internal versioned routing in the same way. The risky URL is the one containing 25.3.6/Browse/MediaData.

==========

Question 4

Question Type: MultipleChoice

You recently deployed two FortiWeb devices in an active-active (A-A) high availability (HA) cluster.

During routine maintenance, you want to confirm that the cluster is synchronizing the correct configuration areas and that both FortiWeb devices behave consistently in production.

As the FortiWeb administrator, which two configuration areas should you examine to verify that

HA synchronization is functioning correctly? (Choose two.)

Options:

A- Check the network configuration on both FortiWeb devices---such as interfaces and static routes---to ensure they are aligned.
B- Review policy configurations, including server policies and protection profiles, to confirm they match across the cluster.
C- Review inspection and mitigation log files to determine if they are being replicated across both FortiWeb devices.
D- Verify whether firmware images and upgrade history are synchronized between the FortiWeb devices.

P2P exams

Answer:

A, B

Explanation:

To validate HA synchronization, the administrator should compare the configuration areas that affect traffic handling and security enforcement. Network configuration, such as interface and routing alignment, is important because HA peers must process traffic consistently. Policy configuration is also critical because server policies, protection profiles, rules, and related web security settings determine how FortiWeb inspects and blocks requests. Logs are not the right synchronization target for this question; log files are used for investigation, not for proving configuration synchronization. Firmware images and upgrade history are also not the normal operational configuration areas used to verify HA policy behavior. The correct verification focus is network configuration consistency and web protection/server policy consistency across the HA cluster.

==========

P2P exams

Question 5

Question Type: MultipleChoice

A third-party penetration test reveals that users can bypass login controls through a mobile API. Your current FortiWeb configuration includes zero trust network access (ZTNA) profiles and cookie security, but API protection and client management are not enabled. The security team asks you to recommend the most effective way to close this gap.

Which FortiWeb adjustment would best prevent future unauthorized API access?

Options:

A- Switch to a reverse-proxy mode to bypass cookie-based controls.
B- Enable API protection and client management to enforce identity checks on mobile API traffic.
C- Replace ZTNA with bot protection to reduce false positives.
D- Log only API traffic and rely on FortiAnalyzer for future alerts.

Answer:

B

Explanation:

The issue is unauthorized access through a mobile API, so the control must enforce API-specific identity and access rules. FortiWeb API protection can validate API structure, methods, paths, and authorization requirements, while client management can help associate requests with legitimate clients or authenticated users. ZTNA profiles and cookie security can help with access and session protection, but they do not replace API-specific authorization controls. Switching reverse-proxy mode to bypass cookie controls makes no sense and could weaken protection. Replacing ZTNA with bot protection addresses a different problem: automation, not API authorization. Logging only records activity after the fact and does not prevent bypass. The correct action is to enable API protection and client management for mobile API traffic.

==========

Question 6

Question Type: MultipleChoice

Refer to the exhibit.

Server policy configuration

```
FortiWeb # show server-policy policy
config server-policy policy
  edit "server_policy"
    set deployment-mode server-pool
    set ssl enable
    set vserver virtual_server
    set service HTTP
    set web-protection-profile custom_inline_protection_profile
    set replacemsg Predefined
    set https-service HTTPS
    set certificate web1
    set tlog enable
  next
end

FortiWeb # show server-policy policy
config waf web-protection-profile inline-protection
  edit "custom_inline_protection_profile"
    set signature-rule custom_signatures
  next
end

FortiWeb #
```

You have deployed FortiWeb behind a FortiGate that is configured as a reverse proxy and inserts the X-Forwarded-For HTTP header when forwarding HTTP and HTTPS traffic.

FortiWeb is using a custom inline protection profile, and logging is enabled, as shown in the exhibit.

You notice that FortiWeb is blocking legitimate users, and all requests in the attack logs appear to come from the FortiGate IP address, not the original client IP address.

Which action should you take to fix this issue?

Options:

A- Replace the current deployment mode with a one-arm proxy to expose source IP addresses.
B- Disable IP-based detection features on FortiWeb to avoid IP-related blocking.
C- Recreate the server policy using the predefined profile instead of a custom one.
D- Modify the protection profile to use the X-Forwarded-For header for client IP address detection.

Answer:

D

Explanation:

The FortiGate is acting as an upstream reverse proxy, so FortiWeb sees the FortiGate address as the direct source IP unless it is configured to read the original client IP from the inserted HTTP header. Since FortiGate already inserts X-Forwarded-For, the proper fix is to modify the FortiWeb protection profile or related client-IP configuration so FortiWeb uses that header for client IP detection. This restores accurate logging, rate limiting, reputation checks, and IP-based enforcement. Changing to one-arm proxy is unnecessary and disruptive. Disabling IP-based detection weakens protection instead of fixing attribution. Recreating the policy with a predefined profile does not address the missing client IP mapping. The correct adjustment is to trust and use X-Forwarded-For.

P2P exams

Question 7

Question Type: MultipleChoice

While reviewing FortiWeb logs, you notice a suspicious login request that failed authentication. You suspect it may be part of an injection attack targeting the login form.

Which input pattern is an example of a typical SQL injection attempt that could bypass authentication checks?

Options:

A- '||(SELECT password FROM users WHERE role='admin')||'
B- <sql>select(ALL USERS);</sql>
C- <script>document.location='/steal?cookie='+document.cookie</script>
D- SELECT username FROM accounts WHERE username='admin';-- ' AND password='password';

P2P exams

Answer:

D

Explanation:

Option D is the strongest SQL injection example because it uses SQL syntax with a comment marker -- to neutralize the password condition. In a vulnerable login query, that could cause the database to evaluate only the username portion and ignore the password check, creating an authentication bypass. Option C is not SQL injection; it is cross-site scripting because it executes JavaScript in the browser. Option B is fake XML-like markup and not a realistic SQL payload. Option A contains a SQL-like subquery, but it is not the best authentication-bypass

pattern shown. FortiWeb's injection defenses are designed to detect SQL injection and XSS as malicious input patterns targeting application logic.

==========

Question 8

Question Type: MultipleChoice

A FortiWeb administrator is hardening a customer checkout website.

The site contains sensitive links such as Login, Payment, and Admin, which are embedded in the HTML content of several pages.

A vulnerability scan shows that automated bots can crawl the web pages and easily enumerate these links by parsing the HTML source, even though users access them normally, through the site navigation.

Which FortiWeb feature should the administrator enable to prevent automated scanners from discovering these links?

Options:

A- Link cloaking
B- URL rewriting
C- URL encryption
D- Deep packet inspection

Answer:

A

Explanation:

Link cloaking is the FortiWeb feature designed to hide sensitive URLs or links from automated scanners and crawlers that parse HTML source code. In this scenario, the problem is not routing, encryption, or packet inspection; the problem is that bots can read embedded links directly from page content and enumerate sensitive paths such as Login, Payment, and Admin. Link cloaking helps obscure those links so that automated tools cannot easily discover them while normal users can still navigate the site as intended. URL rewriting changes URLs for routing or presentation, but it is not the most precise feature for hiding embedded links from scanners. Deep packet inspection is too broad and does not specifically solve HTML link enumeration.

To Get Premium Files for NSE5_FWB_AD-8.0 Visit

https://www.p2pexams.com/products/nse5_fwb_ad-8.0

For More Free Questions Visit

https://www.p2pexams.com/fortinet/pdf/nse5-fwb-ad-8.0