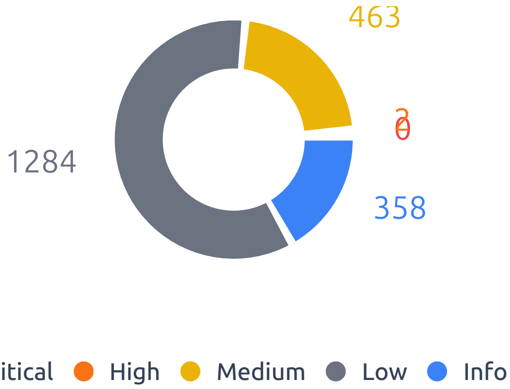
Security Audit Report

Target: pentest-ground.com

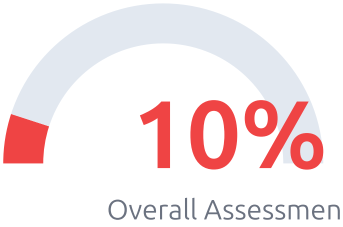
Scan Date: 11/25/2025

This document presents the findings of a comprehensive security assessment conducted on the target infrastructure. The analysis includes a **Vulnerability Assessment (VA)** phase, identifying known CVEs, open ports, and misconfigurations, followed by an automated **Penetration Testing (PT)** phase performing active attacks (e.g., XSS, SQLi) to validate security posture. The goal is to highlight critical risks and provide actionable remediation strategies.

Severity Distribution



Security Score



Open Ports

Totali: 7

Critiche: 0

CVE Summary

Totali: 3

Critiche: 0

HTTP Headers

Missing: 10

Misconfigured: 1

OK: 1

Vulnerability Assessment

Server & Info

Server Info

Domain	pentest-ground.com
IP Address	178.79.134.182
Timestamp of scan	25/11/2025, 17:19:37
Server Version	nginx/1.29.3
Status Code server	200 OK

ALLOWED METHODS

Not Available (OPTIONS not supported)

Security Headers

Header Name	Value / Details	Status
Content-Security-Policy	`Not Present`	⊗
Strict-Transport-Security	`Correctly Configured`	✓
X-Content-Type-Options	`sniff-test`	⚠
X-Frame-Options	`Not Present`	⊗
Permissions-Policy	`Not Present`	⊗
Referrer-Policy	`Not Present`	⊗
Cache-Control	`Not Present`	⊗
Set-Cookie	`Not Present`	⊗
Access-Control-Allow-Origin	`Not Present`	⊗
Cross-Origin-Opener-Policy	`Not Present`	⊗
Cross-Origin-Embedder-Policy	`Not Present`	⊗
Cross-Origin-Resource-Policy	`Not Present`	⊗

✓ Secure ⚠ Weak ⊗ Missing

Open Ports

Port / Proto	Service	Software Detected	State
53 TCP	Domain	`dnsmasq 2.84rc2`	OPEN
80 TCP	Http	`nginx 1.29.3`	OPEN
81 TCP	Http	`nginx 1.29.3`	OPEN
443 TCP	Http	`nginx 1.29.3`	OPEN
4445 TCP	Ssh	`OpenSSH 8.4p1 Debian 5+deb11u5`	OPEN
7001 TCP	Http	`Oracle WebLogic admin httpd`	OPEN
9000 TCP	Http	`nginx 1.29.3`	OPEN

CVE Overview

Vulnerabilità Rilevate 3

Lista delle CVE note identificate analizzando le versioni software.

CVE ID	Severity	Product	Actions
CVE-2018-16845 EXPLOIT	8.1	Nginx F5 NGINX 1.15.5	View Details
CVE-2024-32760	7.5	Nginx F5 NGINX 1.25.5	View Details
CVE-2025-53859	6.5	Nginx F5 NGINX 1.27.3	View Details

Penetration Testing (ZAP)

OWASP ZAP Report

Results of the automatic penetration test

Absence of Anti-CSRF Tokens

Hit **1** resource

DESCRIPTION

No Anti-CSRF tokens were found in a HTML submission form.

A cross-site request forgery is an attack that involves forcing a victim to send an HTTP request to a target destination without their knowledge or intent in order to perform an action as the victim. The underlying cause is application functionality using predictable URL/form actions in a repeatable way. The nature of the attack is that CSRF exploits the trust that a web site has for a user. By contrast, cross-site scripting (XSS) exploits the trust that a user has for a web site. Like XSS, CSRF attacks are not necessarily cross-site, but they can be. Cross-site request forgery is also known as CSRF, XSRF, one-click attack, session riding, confused deputy, and sea surf.

CSRF attacks are effective in a number of situations, including:

* The victim has an active session on the target site.
* The victim is authenticated via HTTP auth on the target site.
* The victim is on the same local network as the target site.

CSRF has primarily been used to perform an action against a target site using the victim's privileges, but recent techniques have been discovered to disclose information by gaining access to the response. The risk of information disclosure is dramatically increased when the target site is vulnerable to XSS, because XSS can be used as a platform for CSRF, allowing the attack to operate within the bounds of the same-origin policy.

RECOMMENDED SOLUTION

Phase: Architecture and Design

Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness easier to avoid.

For example, use anti-CSRF packages such as the OWASP CSRFGuard.

Phase: Implementation

Ensure that your application is free of cross-site scripting issues, because most CSRF defenses can be bypassed using attacker-controlled script.

Phase: Architecture and Design

Generate a unique nonce for each form, place the nonce into the form, and verify the nonce upon receipt of the form. Be sure that the nonce is not predictable (CWE-330).

Note that this can be bypassed using XSS.

Identify especially dangerous operations. When the user performs a dangerous operation, send a separate confirmation request to ensure that the user intended to perform that operation.

Note that this can be bypassed using XSS.

Use the ESAPI Session Management control.

This control includes a component for CSRF.

Do not use the GET method for any request that triggers a state change.

Phase: Implementation

Check the HTTP Referer header to see if the request originated from an expected page. This could break legitimate functionality, because users or proxies may have disabled sending the Referer for privacy reasons.

AFFECTED RESOURCES (1)

Method	URL	Param	Attack Payload (Snippet)
GET	https://pentest-ground.com:7001/console/login/LoginForm.jsp	-	

Content Security Policy (CSP) Header Not Set

Hit **1** resource

DESCRIPTION

Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks, including Cross Site Scripting (XSS) and data injection attacks. These attacks are used for everything from data theft to site defacement or distribution of malware. CSP provides a set of standard HTTP headers that allow website owners to declare approved sources of content that browsers should be allowed to load on that page — covered types are JavaScript, CSS, HTML frames, fonts, images and embeddable objects such as Java applets, ActiveX, audio and video files.

RECOMMENDED SOLUTION

Ensure that your web server, application server, load balancer, etc. is configured to set the Content-Security-Policy header.

AFFECTED RESOURCES (1)

Method	URL	Param	Attack Payload (Snippet)
GET	https://pentest-ground.com:7001/console/login/LoginForm.jsp	-	

Cookie Without Secure Flag

Hit **1** resource

DESCRIPTION

A cookie has been set without the secure flag, which means that the cookie can be accessed via unencrypted connections.

RECOMMENDED SOLUTION

Whenever a cookie contains sensitive information or is a session token, then it should always be passed using an encrypted channel. Ensure that the secure flag is set for cookies containing such sensitive information.

AFFECTED RESOURCES (1)

Method	URL	Param	Attack Payload (Snippet)
GET	`https://pentest-ground.com:7001/console/login/LoginForm.jsp`	`ADMINCONSOLESESSION`	

Cookie without SameSite Attribute

Hit **1** resource

DESCRIPTION

A cookie has been set without the SameSite attribute, which means that the cookie can be sent as a result of a 'cross-site' request. The SameSite attribute is an effective counter measure to cross-site request forgery, cross-site script inclusion, and timing attacks.

RECOMMENDED SOLUTION

Ensure that the SameSite attribute is set to either 'lax' or ideally 'strict' for all cookies.

AFFECTED RESOURCES (1)

Method	URL	Param	Attack Payload (Snippet)
GET	`https://pentest-ground.com:7001/console/login/LoginForm.jsp`	`ADMINCONSOLESESSION`	

Strict-Transport-Security Header Not Set

Hit **7** resources

DESCRIPTION

HTTP Strict Transport Security (HSTS) is a web security policy mechanism whereby a web server declares that complying user agents (such as a web browser) are to interact with it using only secure HTTPS connections (i.e. HTTP layered over TLS/SSL). HSTS is an IETF standards track protocol and is specified in RFC 6797.

RECOMMENDED SOLUTION

Ensure that your web server, application server, load balancer, etc. is configured to enforce Strict-Transport-Security.

AFFECTED RESOURCES (7)

Method	URL	Param	Attack Payload (Snippet)
GET	https://pentest-ground.com:7001/console/css/login.css	-	
GET	https://pentest-ground.com:7001/console/framework/skins/wlsconsole/css/general.css	-	
GET	https://pentest-ground.com:7001/console/framework/skins/wlsconsole/css/window.css	-	
GET	https://pentest-ground.com:7001/console/framework/skins/wlsconsole/images/login-12c.png	-	
GET	https://pentest-ground.com:7001/console/framework/skins/wlsconsole/images/login_WebLogic_branding.png	-	
GET	https://pentest-ground.com:7001/console/login/LoginForm.jsp	-	
POST	https://pentest-ground.com:7001/console/j_security_check	-	

X-Content-Type-Options Header Missing

Hit **6** resources

DESCRIPTION

The Anti-MIME-Sniffing header X-Content-Type-Options was not set to 'nosniff'. This allows older versions of Internet Explorer and Chrome to perform MIME-sniffing on the response body, potentially causing the response body to be interpreted and displayed as a content type other than the declared content type. Current (early 2014) and legacy versions of Firefox will use the declared content type (if one is set), rather than performing MIME-sniffing.

RECOMMENDED SOLUTION

Ensure that the application/web server sets the Content-Type header appropriately, and that it sets the X-Content-Type-Options header to 'nosniff' for all web pages.
If possible, ensure that the end user uses a standards-compliant and modern web browser that does not perform MIME-sniffing at all, or that can be directed by the web application/web server to not perform MIME-sniffing.

AFFECTED RESOURCES (6)

Method	URL	Param	Attack Payload (Snippet)
GET	https://pentest-ground.com:7001/console/css/login.css	x-content-type-options	
GET	https://pentest-ground.com:7001/console/framework/skins/wlsconsole/css/general.css	x-content-type-options	
GET	https://pentest-ground.com:7001/console/framework/skins/wlsconsole/css/window.css	x-content-type-options	
GET	https://pentest-ground.com:7001/console/framework/skins/wlsconsole/images/login-12c.png	x-content-type-options	
GET	https://pentest-ground.com:7001/console/framework/skins/wlsconsole/images/login_WebLogic_branding.png	x-content-type-options	
GET	https://pentest-ground.com:7001/console/login/LoginForm.jsp	x-content-type-options	

Authentication Request Identified

Hit **1** resource

DESCRIPTION

The given request has been identified as an authentication request. The 'Other Info' field contains a set of key=value lines which identify any relevant fields. If the request is in a context which has an Authentication Method set to "Auto-Detect" then this rule will change the authentication to match the request identified.

RECOMMENDED SOLUTION

This is an informational alert rather than a vulnerability and so there is nothing to fix.

AFFECTED RESOURCES (1)

Method	URL	Param	Attack Payload (Snippet)
POST	`https://pentest-ground.com:7001/console/j_security_check`	`j_username`	

Modern Web Application

Hit **1** resource

DESCRIPTION

The application appears to be a modern web application. If you need to explore it automatically then the Ajax Spider may well be more effective than the standard one.

RECOMMENDED SOLUTION

This is an informational alert and so no changes are required.

AFFECTED RESOURCES (1)

Method	URL	Param	Attack Payload (Snippet)
GET	`https://pentest-ground.com:7001/console/login/LoginForm.jsp`	-	

Re-examine Cache-control Directives

Hit **1** resource

DESCRIPTION

The cache-control header has not been set properly or is missing, allowing the browser and proxies to cache content. For static assets like css, js, or image files this might be intended, however, the resources should be reviewed to ensure that no sensitive content will be cached.

RECOMMENDED SOLUTION

For secure content, ensure the cache-control HTTP header is set with "no-cache, no-store, must-revalidate". If an asset should be cached consider setting the directives "public, max-age, immutable".

AFFECTED RESOURCES (1)

Method	URL	Param	Attack Payload (Snippet)
GET	`https://pentest-ground.com:7001/console/login/LoginForm.jsp`	`cache-control`	

Session Management Response Identified

Hit **1** resource

DESCRIPTION

The given response has been identified as containing a session management token. The 'Other Info' field contains a set of header tokens that can be used in the Header Based Session Management Method. If the request is in a context which has a Session Management Method set to "Auto-Detect" then this rule will change the session management to use the tokens identified.

RECOMMENDED SOLUTION

This is an informational alert rather than a vulnerability and so there is nothing to fix.

AFFECTED RESOURCES (1)

Method	URL	Param	Attack Payload (Snippet)
GET	https://pentest-ground.com:7001/console/login/LoginForm.jsp	ADMINCONSOLESESSION	

END OF REPORT

This document contains confidential security information. The findings presented in this report reflect the security posture of the target system at the specific time of the scan. Vulnerabilities may have been remediated or new ones discovered since the generation date.