netsparker

25/09/2025 10:51:40 AM (UTC+05:30)

# Detailed Scan Report

http://zero.webappsecurity.com/

| | |
|---|---|
| **Scan Time** | : 25/09/2025 9:45:46 AM (UTC+05:30) |
| **Scan Duration** | : 00:00:30:08 |
| **Total Requests** | : 4,977 |
| **Average Speed** | : 2.8r/s |

Risk Level:
**HIGH**

| | | |
|---|---|---|
| **22** IDENTIFIED | **8** CONFIRMED | **0** CRITICAL |
| **2** HIGH | 5 MEDIUM | 6 LOW |
| | 4 BEST PRACTICE | 5 INFORMATION |

## Identified Vulnerabilities

| | |
|---|---|
| Critical | 0 |
| High | 2 |
| Medium | 5 |
| Low | 6 |
| Best Practice | 4 |
| Information | 5 |
| **TOTAL** | **22** |

## Confirmed Vulnerabilities

| | |
|---|---|
| Critical | 0 |
| High | 2 |
| Medium | 3 |
| Low | 1 |
| Best Practice | 0 |
| Information | 2 |
| **TOTAL** | **8** |

# Vulnerability Summary

| CONFIRM | VULNERABILITY | METHOD | URL | PARAMETER |
|---|---|---|---|---|
| | Insecure Transportation Security Protocol Supported (SSLv2) | GET | https://zero.webappsecurity.com/ | |
| | Password Transmitted over HTTP | GET | http://zero.webappsecurity.com/login.html | |
| | Apache Server-Status Detected | GET | http://zero.webappsecurity.com/server-status | |
| | HTTP Strict Transport Security (HSTS) Policy Not Enabled | GET | https://zero.webappsecurity.com/ | |
| | Insecure Transportation Security Protocol Supported (SSLv3) | GET | https://zero.webappsecurity.com/ | |
| | Invalid SSL Certificate | GET | https://zero.webappsecurity.com/ | |
| | Weak Ciphers Enabled | GET | https://zero.webappsecurity.com/ | |
| | [Possible] Backup File Disclosure | GET | http://zero.webappsecurity.com/index.old | |
| | [Possible] Phishing by Navigating Browser Tabs | GET | http://zero.webappsecurity.com/ | |
| | Misconfigured Access-Control-Allow-Origin Header | GET | http://zero.webappsecurity.com/ | URI-BASED |
| | Missing X-Frame-Options Header | GET | http://zero.webappsecurity.com/ | |
| | Version Disclosure (Apache Coyote) | GET | http://zero.webappsecurity.com/ | |
| | Insecure Transportation Security Protocol Supported (TLS 1.0) | GET | https://zero.webappsecurity.com/ | |
| | Content Security Policy (CSP) Not Implemented | GET | http://zero.webappsecurity.com/ | |

| CONFIRM | VULNERABILITY | METHOD | URL | PARAMETER |
|---|---|---|---|---|
| | Missing X-XSS-Protection Header | GET | http://zero.webappsecurity.com/ | |
| | Referrer-Policy Not Implemented | GET | http://zero.webappsecurity.com/ | |
| | SameSite Cookie Not Implemented | GET | http://zero.webappsecurity.com/bank/ | |
| | Apache Web Server Identified | GET | http://zero.webappsecurity.com/ | |
| | Default Page Detected (Tomcat) | GET | http://zero.webappsecurity.com/docs/index.html | |
| | Email Address Disclosure | GET | http://zero.webappsecurity.com/resources/css/font-awesome.css | |
| | Forbidden Resource | GET | http://zero.webappsecurity.com/cgi-bin/ | |
| | OPTIONS Method Enabled | OPTIONS | http://zero.webappsecurity.com/ | |

# 1. Insecure Transportation Security Protocol Supported (SSLv2)

HIGH 1 | CONFIRMED 1

Netsparker detected that insecure transportation security protocol (SSLv2) is supported by your web server.

SSLv2 has several flaws. For example, your secure traffic can be observed when you have established it over SSLv2.

### Impact

Attackers can perform man-in-the-middle attacks and observe the encryption traffic between your website and its visitors. Also an attacker can exploit vulnerabilities like DROWN.

### Vulnerabilities

#### 1.1. https://zero.webappsecurity.com/

**CONFIRMED**

**Actions to Take**

We recommended to disable SSLv2 and replace it with TLS 1.2 or higher.  See Remedy section for more details.

**Remedy**

Configure your web server to disallow using weak ciphers.

- For Apache, you should modify the SSLProtocol directive in the `httpd.conf`.

```
SSLProtocol +TLSv1.2
```

- For Nginx, locate any use of the directive ssl_protocols in the `nginx.conf`file and remove `SSLv3`.

```
ssl_protocols TLSv1.2;
```

-
- For Microsoft IIS, you should make some changes on the system registry. **Incorrectly editing the registry may severely damage your system. Before making changes to the registry, you should back up any valued data on your computer.**
    1. Click Start, click Run, type regedt32 or type regedit, and then click OK.
    2. In Registry Editor, locate the following registry key:
       HKey_Local_Machine\System\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\SSL2\
    3. Locate a key named "Server." If it doesn't exist, create it.
    4. Under the "Server" key, locate a DWORD value named "Enabled." If it doesn't exist, create it and set it to "0".
- For lighttpd, put the following lines in your configuration file:

```
ssl.use-sslv2 = "disable"
ssl.use-sslv3 = "disable"
ssl.openssl.ssl-conf-cmd = ("Protocol" => "-TLSv1.1, -TLSv1, -SSLv3") # v1.4.48 or up
ssl.ec-curve = "secp384r1"
```

**External References**

- OWASP - Insecure Configuration Management
- How to disable PCT 1.0, SSL 2.0, SSL 3.0, or TLS 1.0 in Internet Information Services
- OWASP Top 10-2017 A3-Sensitive Data Exposure
- The DROWN Attack

---

## CLASSIFICATION

| | |
|---|---|
| PCI DSS v3.2 | 6.5.4 |
| OWASP 2013 | A6 |
| OWASP 2017 | A3 |
| CWE | 326 |
| CAPEC | 217 |
| WASC | 4 |
| HIPAA | 164.306 |
| ISO27001 | A.14.1.3 |

### CVSS 3.0 SCORE

| | |
|---|---|
| Base | 6.8 (Medium) |
| Temporal | 6.1 (Medium) |
| Environmental | 6.1 (Medium) |

### CVSS Vector String

CVSS:3.0/AV:A/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:N/E:P/RL:O/RC:C

### CVSS 3.1 SCORE

| | |
|---|---|
| Base | 6.8 (Medium) |
| Temporal | 6.1 (Medium) |
| Environmental | 6.1 (Medium) |

**CVSS Vector String**

CVSS:3.1/AV:A/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:N/E:P/RL:O/RC:C

# 2. Password Transmitted over HTTP

HIGH 1 | CONFIRMED 1

Netsparker detected that password data is being transmitted over HTTP.

### Impact

If an attacker can intercept network traffic, he/she can steal users' credentials.

### Vulnerabilities

#### 2.1. http://zero.webappsecurity.com/login.html

**CONFIRMED**

**Input Name**

- user_password

**Form target action**

- /signin.html

**Actions to Take**

1. See the remedy for solution.
2. Move all of your critical forms and pages to HTTPS and do not serve them over HTTP.

**Remedy**

All sensitive data should be transferred over HTTPS rather than HTTP. Forms should be served over HTTPS. All aspects of the application that accept user input, starting from the login process, should only be served over HTTPS.

## CLASSIFICATION

| | |
|---|---|
| PCI DSS v3.2 | 6.5.4 |
| OWASP 2013 | A6 |
| OWASP 2017 | A3 |
| CWE | 319 |
| CAPEC | 65 |
| WASC | 4 |
| ISO27001 | A.14.1.3 |

### CVSS 3.0 SCORE

| | |
|---|---|
| Base | 5.7 (Medium) |
| Temporal | 5.7 (Medium) |
| Environmental | 5.7 (Medium) |

### CVSS Vector String

CVSS:3.0/AV:A/AC:L/PR:N/UI:R/S:U/C:H/I:N/A:N

### CVSS 3.1 SCORE

| | |
|---|---|
| Base | 5.7 (Medium) |
| Temporal | 5.7 (Medium) |
| Environmental | 5.7 (Medium) |

| CVSS Vector String |
| --- |
| CVSS:3.1/AV:A/AC:L/PR:N/UI:R/S:U/C:H/I:N/A:N |

# 3. Apache Server-Status Detected

MEDIUM | 1

Netsparker detected that `Apache server-status`is enabled.

Information disclosed from this page can be used to gain additional information about the target system.

### Impact

An attacker can gather reconnaissance information about the internals of the target web server, such as:

- Server uptime
- Individual request-response statistics and CPU usage of the working processes
- Current HTTP requests, client IP addresses, requested paths, and processed virtual hosts

This type of information can help the attacker gain a greater understanding of the system in use and the other potential avenues of attack available.

### Vulnerabilities

## 3.1. http://zero.webappsecurity.com/server-status

### Certainty

#### Remedy

We recommend disabling this functionality. Comment out the `Location/server-info`section from Apache configuration file `httpd.conf`(for Redhat, Centos, Fedora) or `apache2.conf`(for Debian, Ubuntu).

#### External References

- Exploiting Misconfigured Apache server-status Instances with server-status_PWN

## CLASSIFICATION

| | |
|---|---|
| OWASP 2013 | A5 |
| OWASP 2017 | A6 |
| CWE | 16 |
| CAPEC | 347 |
| WASC | 14 |
| ISO27001 | A.18.1.3 |

### CVSS 3.0 SCORE

| | |
|---|---|
| Base | 5.3 (Medium) |
| Temporal | 5.1 (Medium) |
| Environmental | 5.1 (Medium) |

### CVSS Vector String

CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N/E:H/RL:O/RC:C

### CVSS 3.1 SCORE

| | |
|---|---|
| Base | 5.3 (Medium) |
| Temporal | 5.1 (Medium) |
| Environmental | 5.1 (Medium) |

| CVSS Vector String |
| --- |
| CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N/E:H/RL:O/RC:C |

# 4. HTTP Strict Transport Security (HSTS) Policy Not Enabled

MEDIUM 1

Netsparker identified that HTTP Strict Transport Security (HSTS) policy is not enabled.

The target website is being served from not only HTTPS but also HTTP and it lacks of HSTS policy implementation.

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. The HSTS Policy is communicated by the server to the user agent via a HTTP response header field named "Strict-Transport-Security". HSTS Policy specifies a period of time during which the user agent shall access the server in only secure fashion.

When a web application issues HSTS Policy to user agents, conformant user agents behave as follows:

- Automatically turn any insecure (HTTP) links referencing the web application into secure (HTTPS) links. (For instance, http://example.com/some/page/ will be modified to https://example.com/some/page/ before accessing the server.)
- If the security of the connection cannot be ensured (e.g. the server's TLS certificate is self-signed), user agents show an error message and do not allow the user to access the web application.

## Vulnerabilities

### 4.1. https://zero.webappsecurity.com/

#### Certainty

#### Remedy

Configure your webserver to redirect HTTP requests to HTTPS.

i.e. for Apache, you should have modification in the httpd.conf. For more configurations, please refer to External References section.

```
# load module
LoadModule headers_module modules/mod_headers.so

# redirect all HTTP to HTTPS (optional)
<VirtualHost *:80>
       ServerAlias *
       RewriteEngine On
       RewriteRule ^(.*)$ https://%{HTTP_HOST}$1 [redirect=301]
</VirtualHost>

# HTTPS-Host-Configuration
<VirtualHost *:443>
      # Use HTTP Strict Transport Security to force client to use secure connections only
      Header always set Strict-Transport-Security "max-age=31536000; includeSubDomains"
```

```
    # Further Configuration goes here
    [...]
</VirtualHost>
```

**External References**

- Wikipedia - HTTP Strict Transport Security
- Configure HSTS (HTTP Strict Transport Security) for Apache/Nginx
- HTTP Strict Transport Security (HSTS) HTTP Header
- Mozilla SSL Configuration Generator

**CLASSIFICATION**

| | |
|---|---|
| OWASP 2013 | A6 |
| OWASP 2017 | A3 |
| CWE | 523 |
| CAPEC | 217 |
| WASC | 4 |
| ISO27001 | A.14.1.2 |

# 5. Insecure Transportation Security Protocol Supported (SSLv3)

MEDIUM 1 | CONFIRMED 1

Netsparker detected that insecure transportation security protocol (SSLv3) is supported by your web server.

SSLv3 has several flaws. An attacker can cause connection failures and they can trigger the use of SSL 3.0 to exploit vulnerabilities like POODLE.

### Impact

Attackers can perform man-in-the-middle attacks and observe the encryption traffic between your website and its visitors.

### Vulnerabilities

#### 5.1. https://zero.webappsecurity.com/

**CONFIRMED**

#### Actions to Take

We recommended to disable SSLv3 and replace it with TLS 1.2 or higher.  See Remedy section for more details.

#### Remedy

Configure your web server to disallow using weak ciphers. You need to restart the web server to enable changes.

- For Apache, adjust the SSLProtocol directive provided by the mod_ssl module. This directive can be set either at the server level or in a virtual host configuration.

```
SSLProtocol +TLSv1.2
```

- For Nginx, locate any use of the directive ssl_protocols in the `nginx.conf`file and remove `SSLv3`.

```
ssl_protocols TLSv1.2;
```

- For Microsoft IIS, you should make some changes on the system registry. **Incorrectly editing the registry may severely damage your system. Before making changes to the registry, you should back up any valued data on your computer.**
    1. Click on Start and then Run, type `regedt32`or `regedit`, and then click OK.
    2. In Registry Editor, locate the following registry key or create if it does not exist:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\S
SL 3.0\
```

3. Locate a key named `Server`or create if it doesn't exist.
4. Under the `Server`key, locate a DWORD value named `Enabled`or create if it doesn't exist and set its value to "0".

- For lighttpd, put the following lines in your configuration file:

```
ssl.use-sslv2 = "disable"
ssl.use-sslv3 = "disable"
ssl.openssl.ssl-conf-cmd = ("Protocol" => "-TLSv1.1, -TLSv1, -SSLv3") # v1.4.48 or up
ssl.ec-curve = "secp384r1"
```

**External References**

- How to disable SSlv3
- OWASP - Insecure Configuration Management
- OWASP Top 10-2017 A3-Sensitive Data Exposure
- How to disable PCT 1.0, SSL 2.0, SSL 3.0, or TLS 1.0 in Internet Information Services
- This POODLE Bites: Exploiting The SSL 3.0 Fallback
- IIS Crypto is a free tool that gives administrators the ability to enable or disable protocols, ciphers, hashes and key exchange algorithms on Windows Server 2003, 2008 and 2012
- OWASP - Insufficient Transport Layer Protection

## CLASSIFICATION

| | |
|---|---|
| PCI DSS v3.2 | 6.5.4 |
| OWASP 2013 | A6 |
| OWASP 2017 | A3 |
| CWE | 326 |
| CAPEC | 217 |
| WASC | 4 |
| HIPAA | 164.306 |
| ISO27001 | A.14.1.3 |

### CVSS 3.0 SCORE

| | |
|---|---|
| Base | 6.8 (Medium) |
| Temporal | 6.1 (Medium) |
| Environmental | 6.1 (Medium) |

### CVSS Vector String

CVSS:3.0/AV:A/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:N/E:P/RL:O/RC:C

### CVSS 3.1 SCORE

| | |
|---|---|
| Base | 6.8 (Medium) |
| Temporal | 6.1 (Medium) |
| Environmental | 6.1 (Medium) |

**CVSS Vector String**

CVSS:3.1/AV:A/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:N/E:P/RL:O/RC:C

# 6. Invalid SSL Certificate

MEDIUM 1 CONFIRMED 1

Netsparker identified an invalid SSL certificate.

An SSL certificate can be created and signed by anyone. You should have a valid SSL certificate to make your visitors sure about the secure communication between your website and them. If you have an invalid certificate, your visitors will have trouble distinguishing between your certificate and those of attackers.

### Impact

Attackers can perform man-in-the-middle attacks and observe the encryption traffic between your website and its visitors.

### Vulnerabilities

#### 6.1. https://zero.webappsecurity.com/

**CONFIRMED**

**List of Problems**

- The certificate is not signed by a trusted authority -

**Remedy**

Fix the problem with your SSL certificate to provide secure communication between your website and its visitors.

**External References**

- OWASP - Insecure Configuration Management
- OWASP Top 10-2017 A3-Sensitive Data Exposure

## CLASSIFICATION

| | |
|---|---|
| PCI DSS v3.2 | 6.5.4 |
| OWASP 2013 | A6 |
| OWASP 2017 | A3 |
| CWE | 295 |
| CAPEC | 459 |
| WASC | 4 |
| ISO27001 | A.14.1.3 |

### CVSS 3.0 SCORE

| | |
|---|---|
| Base | 6.8 (Medium) |
| Temporal | 6.8 (Medium) |
| Environmental | 6.8 (Medium) |

### CVSS Vector String

CVSS:3.0/AV:A/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:N

### CVSS 3.1 SCORE

| | |
|---|---|
| Base | 6.8 (Medium) |
| Temporal | 6.8 (Medium) |
| Environmental | 6.8 (Medium) |

| CVSS Vector String |
| --- |
| CVSS:3.1/AV:A/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:N |

# 7. Weak Ciphers Enabled

MEDIUM 1 | CONFIRMED 1

Netsparker detected that weak ciphers are enabled during secure communication (SSL).

You should allow only strong ciphers on your web server to protect secure communication with your visitors.

### Impact

Attackers might decrypt SSL traffic between your server and your visitors.

### Vulnerabilities

#### 7.1. https://zero.webappsecurity.com/

**CONFIRMED**

**List of Supported Weak Ciphers**

- RC4_128_WITH_MD5 (0x10080)
- RC4_128_EXPORT40_WITH_MD5 (0x20080)
- RC2_128_CBC_WITH_MD5 (0x30080)
- RC2_128_CBC_EXPORT40_WITH_MD5 (0x40080)
- DES_64_CBC_WITH_MD5 (0x60040)
- DES_192_EDE3_CBC_WITH_MD5 (0x700C0)
- TLS_RSA_EXPORT_WITH_RC4_40_MD5 (0x0003)
- TLS_RSA_WITH_RC4_128_MD5 (0x0004)
- TLS_RSA_WITH_RC4_128_SHA (0x0005)
- TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5 (0x0006)
- TLS_RSA_EXPORT_WITH_DES40_CBC_SHA (0x0008)
- TLS_RSA_WITH_DES_CBC_SHA (0x0009)
- TLS_RSA_WITH_3DES_EDE_CBC_SHA (0x000A)
- TLS_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA (0x0014)
- TLS_DHE_RSA_WITH_DES_CBC_SHA (0x0015)
- TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA (0x0016)
- TLS_RSA_WITH_AES_128_CBC_SHA (0x002F)
- TLS_DHE_RSA_WITH_AES_128_CBC_SHA (0x0033)
- TLS_RSA_WITH_AES_256_CBC_SHA (0x0035)
- TLS_DHE_RSA_WITH_AES_256_CBC_SHA (0x0039)

**Actions to Take**

1. For Apache, you should modify the SSLCipherSuite directive in the `httpd.conf`.

```
SSLCipherSuite HIGH:MEDIUM:!MD5:!RC4
```

2. Lighttpd:

```
ssl.honor-cipher-order = "enable"
ssl.cipher-list = "EECDH+AESGCM:EDH+AESGCM"
```

3. For Microsoft IIS, you should make some changes to the system registry. **Incorrectly editing the registry may severely damage your system. Before making changes to the registry, you should back up any valued data on your computer.**

   **a.**Click Start, click Run, type `regedt32`or type `regedit`, and then click OK.
   **b.**In Registry Editor, locate the following registry key: `HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders`
   **c.**Set "Enabled" DWORD to "0x0" for the following registry keys:

```
SCHANNEL\Ciphers\DES 56/56
SCHANNEL\Ciphers\RC4 64/128
SCHANNEL\Ciphers\RC4 40/128
SCHANNEL\Ciphers\RC2 56/128
SCHANNEL\Ciphers\RC2 40/128
SCHANNEL\Ciphers\NULL
SCHANNEL\Hashes\MD5
```

**Remedy**

Configure your web server to disallow using weak ciphers.

**External References**

- OWASP - Insecure Configuration Management
- OWASP Top 10-2017 A3-Sensitive Data Exposure
- Zombie Poodle - Golden Doodle (CBC)
- Mozilla SSL Configuration Generator
- Strong Ciphers for Apache, Nginx and Lighttpd

## CLASSIFICATION

| | |
|---|---|
| PCI DSS v3.2 | 6.5.4 |
| OWASP 2013 | A6 |
| OWASP 2017 | A3 |
| CWE | 327 |
| CAPEC | 217 |
| WASC | 4 |
| ISO27001 | A.14.1.3 |

### CVSS 3.0 SCORE

| | |
|---|---|
| Base | 6.8 (Medium) |
| Temporal | 6.8 (Medium) |
| Environmental | 6.8 (Medium) |

### CVSS Vector String

CVSS:3.0/AV:A/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:N

### CVSS 3.1 SCORE

| | |
|---|---|
| Base | 6.8 (Medium) |
| Temporal | 6.8 (Medium) |
| Environmental | 6.8 (Medium) |

| CVSS Vector String |
| --- |
| CVSS:3.1/AV:A/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:N |

# 8. [Possible] Backup File Disclosure

LOW | 1

Netsparker identified a possible backup file disclosure on the web server.

### Impact

Backup files can contain old or current versions of a file on the web server. This could include sensitive data such as password files or even the application's source code. This form of issue normally leads to further vulnerabilities or, at worst, sensitive information disclosure.

### Vulnerabilities

#### 8.1. http://zero.webappsecurity.com/index.old

### Certainty

#### Remedy

Do not store backup files on production servers.

**CLASSIFICATION**

| | |
|---|---|
| PCI DSS v3.2 | 6.5.8 |
| OWASP 2013 | A7 |
| OWASP 2017 | A5 |
| CWE | 530 |
| CAPEC | 87 |
| WASC | 34 |
| HIPAA | 164.306(A), 164.308(A) |
| ISO27001 | A.18.1.3 |

# 9. [Possible] Phishing by Navigating Browser Tabs

LOW 1

Netsparker identified possible phishing by navigating browser tabs but was unable to confirm the vulnerability.

Open windows with normal hrefs with the tag `target="_blank"`can modify *window.opener.location*and replace the parent webpage with something else, even on a different origin.

### Impact

While this vulnerability doesn't allow script execution, it does allow phishing attacks that silently replace the parent tab. If the links lack `rel="noopener noreferrer"`attribute, a third party site can change the URL of the source tab using *window.opener.location.assign*and trick the users into thinking that they're still in a trusted page and lead them to enter their sensitive data on the malicious website.

### Vulnerabilities

## 9.1. http://zero.webappsecurity.com/

**External Links**

- https://www.microfocus.com/about/legal/#privacy
- https://www.microfocus.com/about/legal/#privacy

### Certainty

**Remedy**

- Add `rel=noopener`to the linksto prevent pages from abusing *window.opener*. This ensures that the page cannot access the *window.opener*property in Chrome and Opera browsers.

- For older browsers and in Firefox, you can add `rel=noreferrer`which additionally disables the Referer header.

```
<a href="..." target="_blank" rel="noopener noreferrer">...</a>
```

**External References**

- Reverse Tabnabbing
- Blankshield & Reverse Tabnabbing Attacks
- Target="_blank" - the most underestimated vulnerability ever

## CLASSIFICATION

| | |
|---|---|
| OWASP 2013 | A5 |
| OWASP 2017 | A6 |
| CWE | 16 |
| WASC | 15 |
| ISO27001 | A.14.1.2 |

# 10. Insecure Transportation Security Protocol Supported (TLS 1.0)

LOW 1 | CONFIRMED 1

Netsparker detected that insecure transportation security protocol (TLS 1.0) is supported by your web server.

TLS 1.0 has several flaws. An attacker can cause connection failures and they can trigger the use of TLS 1.0 to exploit vulnerabilities like BEAST (Browser Exploit Against SSL/TLS).

Websites using TLS 1.0 are considered non-compliant by PCI since 30 June 2018.

### Impact

Attackers can perform man-in-the-middle attacks and observe the encryption traffic between your website and its visitors.

### Vulnerabilities

#### 10.1. https://zero.webappsecurity.com/

**CONFIRMED**

**Actions to Take**

We recommended to disable TLS 1.0 and replace it with TLS 1.2 or higher.  See Remedy section for more details.

**Remedy**

Configure your web server to disallow using weak ciphers. You need to restart the web server to enable changes.

- For Apache, adjust the SSLProtocol directive provided by the mod_ssl module. This directive can be set either at the server level or in a virtual host configuration.

```
SSLProtocol +TLSv1.2
```

- For Nginx, locate any use of the directive ssl_protocols in the `nginx.conf`file and remove `TLSv1`.

```
ssl_protocols TLSv1.2;
```

- For Microsoft IIS, you should make some changes on the system registry. **Incorrectly editing the registry may severely damage your system. Before making changes to the registry, you should back up any valued data on your computer.**
    1. Click on Start and then Run, type `regedt32`or `regedit`, and then click OK.
    2. In Registry Editor, locate the following registry key or create if it does not exist:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\T
LS 1.0\
```

3. Locate a key named `Server` or create if it doesn't exist.
4. Under the `Server` key, locate a DWORD value named `Enabled` or create if it doesn't exist and set its value to "0".

- For lighttpd, put the following lines in your configuration file:

```
ssl.use-sslv2 = "disable"
ssl.use-sslv3 = "disable"
ssl.openssl.ssl-conf-cmd = ("Protocol" => "-TLSv1.1, -TLSv1, -SSLv3") # v1.4.48 or up
ssl.ec-curve = "secp384r1"
```

**External References**

- How to Disable TLS v1.0
- OWASP - Insecure Configuration Management
- OWASP Top 10 - 2017 A3 - Sensitive Data Exposure
- How to disable PCT 1.0, SSL 2.0, SSL 3.0, or TLS 1.0 in Internet Information Services
- IIS Crypto is a free tool that gives administrators the ability to enable or disable protocols, ciphers, hashes and key exchange algorithms on Windows Server 2003, 2008 and 2012
- Date Change for Migrating from SSL and Early TLS
- Browser Exploit Against SSL/TLS Attack (BEAST)
- Are You Ready for 30 June 2018? Saying Goodbye to SSL/early TLS

**CLASSIFICATION**

| | |
|---|---|
| PCI DSS v3.2 | 6.5.4 |
| OWASP 2013 | A6 |
| OWASP 2017 | A3 |
| CWE | 326 |
| CAPEC | 217 |
| WASC | 4 |
| HIPAA | 164.306 |
| ISO27001 | A.14.1.3 |

# 11. Misconfigured Access-Control-Allow-Origin Header

LOW 1

Netsparker detected a possibly misconfigured Access-Control-Allow-Origin header in resource's HTTP response.

Cross-origin resource sharing (CORS) is a mechanism that allows resources on a web page to be requested outside the domain through XMLHttpRequest.

Unless this HTTP header is present, such "cross-domain" requests are forbidden by web browsers, per the same-origin security policy.

### Impact

This is generally not appropriate when using the same-origin security policy. The only case where this is appropriate when using the same-origin policy is when a page or API response is considered completely public content and it is intended to be accessible to everyone.

### Vulnerabilities

#### 11.1. http://zero.webappsecurity.com/

| Method | Parameter | Value |
|---|---|---|
| GET | URI-BASED | |

**Access-Control-Allow-Origin**

- *

### Certainty

### Remedy

If this page is intended to be accessible to everyone, you don't need to take any action. Otherwise please follow the guidelines for different architectures below in order to set this header and permit outside domain.
Apache

- Add the following line inside either the <directory>, <location>, <files> or <virtualhost> sections of your server config (usually located in `httpd.conf`or `apache.conf`), or within a `.htaccess`file.

```
Header set Access-Control-Allow-Origin "domain"
```

IIS6

1. Open Internet Information Service (IIS) Manager
2. Right click the site you want to enable CORS for and go to Properties
3. Change to the HTTP Headers tab
4. In the Custom HTTP headers section, click Add
5. Enter Access-Control-Allow-Origin as the header name
6. Enter `domain`as the header value

IIS7

- Merge the following xml into the web.config file at the root of your application or site:

```
<?xml version="1.0" encoding="utf-8" ?>
<configuration>
  <system.webserver>
    <httpprotocol>
      <customheaders>
        <add name="Access-Control-Allow-Origin" value="domain" />
      </customheaders>
    </httpprotocol>
  </system.webserver>
</configuration>
```

ASP.NET

- If you don't have access to configure IIS, you can still add the header through ASP.NET by adding the following line to your source pages:

```
Response.AppendHeader("Access-Control-Allow-Origin", "domain");
```

**External References**

- Cross-Origin Resource Sharing
- HTTP access control (CORS)
- Using CORS

## CLASSIFICATION

| | |
|---|---|
| OWASP 2013 | A5 |
| OWASP 2017 | A6 |
| CWE | 16 |
| WASC | 15 |
| ISO27001 | A.14.1.2 |

# 12. Missing X-Frame-Options Header

LOW | 1

Netsparker detected a missing `X-Frame-Options`header which means that this website could be at risk of a clickjacking attack.

The `X-Frame-Options`HTTP header field indicates a policy that specifies whether the browser should render the transmitted resource within a `frame`or an `iframe`. Servers can declare this policy in the header of their HTTP responses to prevent clickjacking attacks, which ensures that their content is not embedded into other pages or frames.

### Impact

Clickjacking is when an attacker uses multiple transparent or opaque layers to trick a user into clicking on a button or link on a framed page when they were intending to click on the top level page. Thus, the attacker is "hijacking" clicks meant for their page and routing them to other another page, most likely owned by another application, domain, or both.

Using a similar technique, keystrokes can also be hijacked. With a carefully crafted combination of stylesheets, iframes, and text boxes, a user can be led to believe they are typing in the password to their email or bank account, but are instead typing into an invisible frame controlled by the attacker.

### Vulnerabilities

## 12.1. http://zero.webappsecurity.com/

### Certainty

#### Remedy

- Sending the proper X-Frame-Options in HTTP response headers that instruct the browser to not allow framing from other domains.
  - `X-Frame-Options: DENY`It completely denies to be loaded in frame/iframe.
  - `X-Frame-Options: SAMEORIGIN`It allows only if the site which wants to load has a same origin.
  - `X-Frame-Options: ALLOW-FROM` *`URL`*It grants a specific URL to load itself in a iframe. However please pay attention to that, not all browsers support this.
- Employing defensive code in the UI to ensure that the current frame is the most top level window.

#### External References

- Clickjacking
- Can I Use X-Frame-Options
- X-Frame-Options HTTP Header

#### Remedy References

- Clickjacking Defense Cheat Sheet

## CLASSIFICATION

| | |
|---|---|
| OWASP 2013 | A5 |
| OWASP 2017 | A6 |
| CWE | 693 |
| CAPEC | 103 |
| ISO27001 | A.14.2.5 |

# 13. Version Disclosure (Apache Coyote)

LOW 1

Netsparker identified a version disclosure (Apache Coyote) in target web server's HTTP response.

This information can help an attacker gain a greater understanding of the systems in use and potentially develop further attacks targeted at the specific version of Apache.

## Impact

An attacker might use the disclosed information to harvest specific security vulnerabilities for the version identified.

## Vulnerabilities

### 13.1. http://zero.webappsecurity.com/

**Extracted Version**

- Apache-Coyote/1.1

## Certainty

**Remedy**

Configure your web server to prevent information leakage from the `SERVER`header of its HTTP response.

**CLASSIFICATION**

| | |
|---|---|
| OWASP 2013 | A5 |
| OWASP 2017 | A6 |
| CWE | 205 |
| CAPEC | 170 |
| WASC | 45 |
| HIPAA | 164.306(A), 164.308(A) |
| ISO27001 | A.18.1.3 |

# 14. Content Security Policy (CSP) Not Implemented

BEST PRACTICE | 1

CSP is an added layer of security that helps to mitigate mainly Cross-site Scripting attacks.

CSP can be enabled instructing the browser with a Content-Security-Policy directive in a response header;

```
Content-Security-Policy: script-src 'self';
```

or in a meta tag;

```
<meta http-equiv="Content-Security-Policy" content="script-src 'self';">
```

In the above example, you can restrict script loading only to the same domain. It will also restrict inline script executions both in the element attributes and the event handlers. There are various directives which you can use by declaring CSP:

- **script-src:**Restricts the script loading resources to the ones you declared. By default, it disables inline script executions unless you permit to the evaluation functions and inline scripts by the unsafe-eval and unsafe-inline keywords.
- **base-uri:**Base element is used to resolve relative URL to absolute one. By using this CSP directive, you can define all possible URLs which could be assigned to base-href attribute of the document.
- **frame-ancestors**: It is very similar to X-Frame-Options HTTP header. It defines the URLs by which the page can be loaded in an iframe.
- **frame-src / child-src**: frame-src is the deprecated version of child-src. Both define the sources that can be loaded by iframe in the page. (Please note that frame-src was brought back in CSP 3)
- **object-src**: Defines the resources that can be loaded by embedding such as Flash files, Java Applets.
- **img-src**: As its name implies, it defines the resources where the images can be loaded from.
- **connect-src**: Defines the whitelisted targets for XMLHttpRequest and WebSocket objects.
- **default-src**: It is a fallback for the directives that mostly ends with -src suffix. When the directives below are not defined, the value set to default-src will be used instead:
    - child-src
    - connect-src
    - font-src
    - img-src
    - manifest-src
    - media-src
    - object-src
    - script-src
    - style-src

When setting the CSP directives, you can also use some CSP keywords:

- **none**: Denies loading resources from anywhere.
- **self** : Points to the document's URL (domain + port).
- **unsafe-inline**: Permits running inline scripts.
- **unsafe-eval**: Permits execution of evaluation functions such as eval().

In addition to CSP keywords, you can also use wildcard or only a scheme when defining whitelist URLs for the points. Wildcard can be used for subdomain and port portions of the URLs:

```
Content-Security-Policy: script-src https://*.example.com;
Content-Security-Policy: script-src https://example.com:*;
Content-Security-Policy: script-src https:;
```

It is also possible to set a CSP in Report-Only mode instead of forcing it immediately in the migration period. Thus you can see the violations of the CSP policy in the current state of your web site while migrating to CSP:

```
Content-Security-Policy-Report-Only: script-src 'self'; report-uri: https://example.com;
```

### Impact

There is no direct impact of not implementing CSP on your website. However, if your website is vulnerable to a Cross-site Scripting attack CSP can prevent successful exploitation of that vulnerability. By not implementing CSP you'll be missing out this extra layer of security.

## Vulnerabilities

## 14.1. http://zero.webappsecurity.com/

### Certainty

#### Actions to Take

- Enable CSP on your website by sending the `Content-Security-Policy`in HTTP response headers that instruct the browser to apply the policies you specified.
- Apply the whitelist and policies as strict as possible.
- Rescan your application to see if Netsparker identifies any weaknesses in your policies.

#### Remedy

Enable CSP on your website by sending the `Content-Security-Policy`in HTTP response headers that instruct the browser to apply the policies you specified.

#### External References

- An Introduction to Content Security Policy
- Content Security Policy (CSP) HTTP Header
- Content Security Policy (CSP)

**CLASSIFICATION**

| | |
|---|---|
| CWE | 16 |
| WASC | 15 |
| ISO27001 | A.14.2.5 |

# 15. Missing X-XSS-Protection Header

BEST PRACTICE | 1

Netsparker detected a missing `X-XSS-Protection`header which means that this website could be at risk of a Cross-site Scripting (XSS) attacks.

### Impact

This issue is reported as additional information only. There is no direct impact arising from this issue.

### Vulnerabilities

#### 15.1. http://zero.webappsecurity.com/

### Certainty

#### Remedy

Add the X-XSS-Protection header with a value of "1; mode= block".

- ```
  X-XSS-Protection: 1; mode=block
  ```

#### External References

- Internet Explorer 8 Security Features - MSDN
- X-XSS-Protection HTTP Header
- Internet Explorer 8 XSS Filter

**CLASSIFICATION**

| | |
|---|---|
| CWE | 16 |
| WASC | 15 |
| HIPAA | 164.308(A) |
| ISO27001 | A.14.2.5 |
  ```

# 16. Referrer-Policy Not Implemented

BEST PRACTICE | 1

Netsparker detected that no Referrer-Policy header implemented.

Referrer-Policy is a security header designed to prevent cross-domain Referer leakage.

### Impact

Referer header is a request header that indicates the site which the traffic originated from. If there is no adequate prevention in place, the URL itself, and even sensitive information contained in the URL will be leaked to the cross-site.

The lack of Referrer-Policy header might affect privacy of the users and site's itself

### Vulnerabilities

#### 16.1. http://zero.webappsecurity.com/

### Certainty

#### Actions to Take

In a response header:

```
Referrer-Policy: no-referrer | same-origin | origin | strict-origin | no-origin-when-downgrading
```

In a META tag

```
<meta name="Referrer-Policy" value="no-referrer | same-origin"/>
```

In an element attribute

```
<a href="http://crosssite.example.com" rel="noreferrer"></a>
```

or

```
<a href="http://crosssite.example.com" referrerpolicy="no-referrer | same-origin | origin | strict-
origin | no-origin-when-downgrading"></a>
```

**Remedy**

Please implement a Referrer-Policy by using the Referrer-Policy response header or by declaring it in the meta tags. It's also possible to control referrer information over an HTML-element by using the rel attribute.

**External References**

- Referrer Policy
- Referrer Policy - MDN
- Referrer Policy HTTP Header
- A New Security Header: Referrer Policy
- Can I Use Referrer-Policy

**CLASSIFICATION**

| | |
|---|---|
| OWASP 2013 | A6 |
| OWASP 2017 | A3 |
| CWE | 200 |
| ISO27001 | A.14.2.5 |

# 17. SameSite Cookie Not Implemented

BEST PRACTICE | 1

Cookies are typically sent to third parties in cross origin requests. This can be abused to do CSRF attacks. Recently a new cookie attribute named *SameSite*was proposed to disable third-party usage for some cookies, to prevent CSRF attacks.

Same-site cookies allow servers to mitigate the risk of CSRF and information leakage attacks by asserting that a particular cookie should only be sent with requests initiated from the same registrable domain.

## Vulnerabilities

### 17.1. http://zero.webappsecurity.com/bank/

**Identified Cookie(s)**

- JSESSIONID

**Cookie Source**

- HTTP Header

### Certainty

**Remedy**

The server can set a same-site cookie by adding the `SameSite=...`attribute to the `Set-Cookie`header. There are three possible values for the `SameSite`attribute:

- Lax:In this mode, the cookie will only be sent with a top-level get request.

```
Set-Cookie: key=value; SameSite=Lax
```

- Strict: In this mode, the cookie will not be sent with any cross-site usage even if the user follows a link to another website.

```
Set-Cookie: key=value; SameSite=Strict
```

- None: In this mode, the cookie will be sent with the cross-site requests. Cookies with `SameSite=None`must also specify the `Secure`attribute to transfer them via a secure context. Setting a `SameSite=None`cookie without the `Secure`attribute will be rejected by the browsers.

```
Set-Cookie: key=value; SameSite=None; Secure
```

**External References**

- Security Cookies - SameSite Attribute - Netsparker
- Using the Same-Site Cookies Attribute to Prevent CSRF Attacks
- Same-site Cookies
- Preventing CSRF with the same-site cookie attribute
- SameSite cookies explained
- Get Ready for New SameSite=None; Secure Cookie Settings

## CLASSIFICATION

| | |
|---|---|
| CWE | 16 |
| WASC | 15 |
| ISO27001 | A.14.2.5 |

# 18. Apache Web Server Identified

INFORMATION | 1

Netsparker identified a web server (Apache) in the target web server's HTTP response.

### Impact

This issue is reported as additional information only. There is no direct impact arising from this issue.

### Vulnerabilities

#### 18.1. http://zero.webappsecurity.com/

### Certainty

#### External References

- Apache ServerTokens Directive

## CLASSIFICATION

| | |
|---|---|
| CWE | 200 |
| WASC | 13 |
| OWASP Proactive Controls | C7 |
| ISO27001 | A.18.1.3 |

### CVSS 3.0 SCORE

| | |
|---|---|
| Base | 5.3 (Medium) |
| Temporal | 5.1 (Medium) |
| Environmental | 5.1 (Medium) |

### CVSS Vector String

CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N/E:H/RL:O/RC:C

### CVSS 3.1 SCORE

| | |
|---|---|
| Base | 5.3 (Medium) |
| Temporal | 5.1 (Medium) |
| Environmental | 5.1 (Medium) |

### CVSS Vector String

CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N/E:H/RL:O/RC:C

# 19. Default Page Detected (Tomcat)

INFORMATION 1

Netsparker detected the default Tomcat page.

This issue is reported for information only. If there is any other vulnerability identified regarding this resource, Netsparker will report it as a separate issue.

## Vulnerabilities

### 19.1. http://zero.webappsecurity.com/docs/index.html

#### Certainty

## CLASSIFICATION

| | |
|---|---|
| CWE | 200 |
| WASC | 13 |
| OWASP Proactive Controls | C7 |
| ISO27001 | A.18.1.3 |

### CVSS 3.0 SCORE

| | |
|---|---|
| Base | 4.3 (Medium) |
| Temporal | 4.1 (Medium) |
| Environmental | 4.1 (Medium) |

### CVSS Vector String

CVSS:3.0/AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:N/A:N/E:H/RL:O/RC:C

### CVSS 3.1 SCORE

| | |
|---|---|
| Base | 4.3 (Medium) |
| Temporal | 4.1 (Medium) |
| Environmental | 4.1 (Medium) |

### CVSS Vector String

CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:N/A:N/E:H/RL:O/RC:C

# 20. Email Address Disclosure

INFORMATION 1

Netsparker identified an Email Address Disclosure.

### Impact

Email addresses discovered within the application can be used by both spam email engines and also brute-force tools. Furthermore, valid email addresses may lead to social engineering attacks.

### Vulnerabilities

#### 20.1. http://zero.webappsecurity.com/resources/css/font-awesome.css

**Email Address(es)**

- dave@davegandy.com

### Certainty

**Remedy**

Use generic email addresses such as contact@ or info@ for general communications and remove user/people-specific email addresses from the website; should this be required, use submission forms for this purpose.

**External References**

- Wikipedia - Email Spam

## CLASSIFICATION

| | |
|---|---|
| CWE | 200 |
| CAPEC | 118 |
| WASC | 13 |
| OWASP Proactive Controls | C7 |
| ISO27001 | A.9.4.1 |

### CVSS 3.0 SCORE

| | |
|---|---|
| Base | 5.3 (Medium) |
| Temporal | 5.3 (Medium) |
| Environmental | 5.3 (Medium) |

### CVSS Vector String

CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

### CVSS 3.1 SCORE

| | |
|---|---|
| Base | 5.3 (Medium) |
| Temporal | 5.3 (Medium) |
| Environmental | 5.3 (Medium) |

### CVSS Vector String

CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:N/A:N

# 21. Forbidden Resource

INFORMATION 1 | CONFIRMED 1

Netsparker identified a forbidden resource.

Access to this resource has been denied by the web server. This is generally not a security issue, and is reported here for informational purposes.

### Impact

This issue is reported as additional information only. There is no direct impact arising from this issue.

### Vulnerabilities

#### 21.1. http://zero.webappsecurity.com/cgi-bin/

**CONFIRMED**

**CLASSIFICATION**

| | |
|---|---|
| OWASP Proactive Controls | C8 |
| ISO27001 | A.8.1.1 |

# 22. OPTIONS Method Enabled

INFORMATION 1 | CONFIRMED 1

Netsparker detected that `OPTIONS`method is allowed. This issue is reported as extra information.

### Impact

Information disclosed from this page can be used to gain additional information about the target system.

### Vulnerabilities

#### 22.1. http://zero.webappsecurity.com/

**CONFIRMED**

**Allowed methods**

- GET, HEAD, POST, PUT, DELETE, TRACE, OPTIONS, PATCH

**Remedy**

Disable `OPTIONS`method in all production systems.

**External References**

- Testing for HTTP Methods and XST (OWASP-CM-008)
- HTTP/1.1: Method Definitions

**CLASSIFICATION**

| | |
|---|---|
| OWASP 2013 | A5 |
| OWASP 2017 | A6 |
| CWE | 16 |
| CAPEC | 107 |
| WASC | 14 |
| ISO27001 | A.14.1.2 |

## Show Scan Detail

| | | |
|---|---|---|
| **Enabled Security Checks** | : | Apache Struts S2-045 RCE,<br>Apache Struts S2-046 RCE,<br>BREACH Attack,<br>Code Evaluation,<br>Code Evaluation (Out of Band),<br>Command Injection,<br>Command Injection (Blind),<br>Content Security Policy,<br>Content-Type Sniffing,<br>Cookie,<br>Cross Frame Options Security,<br>Cross-Origin Resource Sharing (CORS),<br>Cross-Site Request Forgery,<br>Cross-site Scripting,<br>Cross-site Scripting (Blind),<br>Custom Script Checks (Active),<br>Custom Script Checks (Passive),<br>Custom Script Checks (Per Directory),<br>Custom Script Checks (Singular),<br>Drupal Remote Code Execution,<br>Expect Certificate Transparency (Expect-CT),<br>Expression Language Injection,<br>File Upload,<br>Header Analyzer,<br>Heartbleed,<br>HSTS,<br>HTML Content,<br>HTTP Header Injection,<br>HTTP Methods,<br>HTTP Status,<br>HTTP.sys (CVE-2015-1635),<br>IFrame Security,<br>Insecure JSONP Endpoint,<br>Insecure Reflected Content,<br>JavaScript Libraries,<br>Local File Inclusion,<br>Login Page Identifier,<br>Mixed Content,<br>Open Redirection,<br>Referrer Policy,<br>Reflected File Download,<br>Remote File Inclusion,<br>Remote File Inclusion (Out of Band),<br>Reverse Proxy Detection,<br>RoR Code Execution,<br>Server-Side Request Forgery (DNS),<br>Server-Side Request Forgery (Pattern Based),<br>Server-Side Template Injection,<br>Signatures, |

| | | |
|---|---|---|
| | | SQL Injection (Blind),<br>SQL Injection (Boolean),<br>SQL Injection (Error Based),<br>SQL Injection (Out of Band),<br>SSL,<br>Static Resources (All Paths),<br>Static Resources (Only Root Path),<br>Unicode Transformation (Best-Fit Mapping),<br>WAF Identifier,<br>Web App Fingerprint,<br>Web Cache Deception,<br>WebDAV,<br>Windows Short Filename,<br>XML External Entity,<br>XML External Entity (Out of Band) |
| **URL Rewrite Mode** | : | Heuristic |
| **Detected URL Rewrite Rule(s)** | : | None |
| **Excluded URL Patterns** | : | (log\|sign)\-?(out\|off)<br>exit<br>endsession<br>gtm\.js<br>WebResource\.axd<br>ScriptResource\.axd |
| **Authentication** | : | None |
| **Scheduled** | : | No |
| **Additional Website(s)** | : | None |

This report created with 5.8.2.28358-master-3d7991d
https://www.netsparker.com