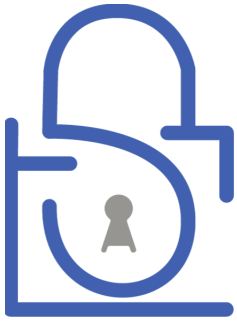
FONDAZIONE BRUNO KESSLER
CENTER FOR CYBERSECURITY

Target: localhost:4444

Analysis performed on 2025-10-29 17:41



TLSAssistant

Version 3.1.0

Confidentiality Disclaimer

This report contains confidential information intended solely for the use of the individual or entity to whom it is addressed. If you are not the intended recipient, please be advised that any disclosure, copying, distribution, or use of the contents of this report is strictly prohibited. If you have received this report in error, please notify the sender immediately and delete the original message.

Security Tool Report Disclaimer

The information provided in this report is the result of security testing conducted by [Your Company/Organization] using TLSAssistant. The purpose of this tool is to assess the security of TLS configurations. The findings and recommendations presented in this report are based on the specific conditions and configurations tested at the time of the assessment. We respectfully recommend verifying the identified vulnerability thoroughly.

No Warranty or Guarantee

This report is provided "as is," without any warranty, express or implied, concerning the accuracy, completeness, legal value or reliability of the information contained herein. [Your Company/Organization] disclaims all warranties, including but not limited to, the implied warranties of merchantability and fitness for a particular purpose.

By reviewing this report, you acknowledge and agree to the terms and conditions outlined above.

Recap

Modules	localhost:4444
3SHAKE	**Potentially Vulnerable**
alpaca	Not Vulnerable
BEAST	Not Vulnerable
BREACH	Not Vulnerable
CSS Injection	Not Vulnerable
Missing Certificate Transparency	Not Vulnerable
CRIME	Not Vulnerable
DROWN	Not Vulnerable
Freak	Not Vulnerable
Heartbleed	Not Vulnerable
HSTS not preloaded	**Potentially Vulnerable**
HSTS not set	**Potentially Vulnerable**
HTTPS not enforced	**Potentially Vulnerable**
Logjam	Not Vulnerable
Lucky 13	Not Vulnerable
Bar Mitzvah	Not Vulnerable
RC4 NOMORE	Not Vulnerable
padding_oracle	Not Vulnerable
PFS	Not Vulnerable
Renegotiation attack	Not Vulnerable
ROBOT	Not Vulnerable
sslpoodle	Not Vulnerable
Sweet32	Not Vulnerable
Ticketbleed	Not Vulnerable
tlspoodle	Not Vulnerable

Detected vulnerabilities

localhost:4444

3SHAKE

CVE:Library dependent - CVSS3:Library dependent

Triple Handshake Attack

Due to the incorrect handling of the session identifier, an attacker is able to force two sessions to have the same Master Secret and ID. The attacker performs 3SHAKE by providing a server to which the victim deliberately connects. Once connected, the malicious server exploits the renegotiation mechanism to manipulate the session. The attack leads to a client impersonation that, by breaking both confidentiality and authentication, has a serious impact on the transmission.

Mitigation

Textual

The only acceptable mitigation is to use the extended_master_secret TLS extension. For this reason it is recommended to update the TLS library to a version that supports the aforementioned extension (e.g. OpenSSL v1.1.0+).

Apache

No snippet available

nginx

No snippet available

HSTS not preloaded

If the HSTS header is preloaded but the webserver does not have a valid certificate, the user will not be able to access the target website.

Mitigation

Textual

Check if your certificate has expired or it is not valid. If it is not valid, get a new certificate and deploy it on your server. If it has expired, renew it and deploy it on your server.

Apache

No snippet available

nginx
No snippet available

HSTS not set

Without the HSTS header, an attacker can use the SSL stripping attack to redirect all the HTTPS connection to their unsecure counterparts. By doing this, all the messages are sent in plaintext and can thus be manipulated.

Mitigation

Textual
Enable the HSTS header transmission within the webserver's settings.

Apache
1. open your Apache configuration file (default:
/etc/apache2/sites-available/default-ssl.conf);
2. add the line Header always set Strict-Transport-Security "max-age=31536000".

N.B. restart the server by typing: sudo service apache2 restart and be sure that mod_headers is enabled.

nginx
1. In a default situation, you can edit your website configuration
/etc/nginx/sites-enabled/default
(if you changed your site conf name
/etc/nginx/sites-enabled/YOURSITECONFIGURATION);
2. Add inside server{...} brackets: add_header Strict-Transport-Security "max-age=31536000; includeSubdomains; preload";

N.B. restart the server by typing: sudo service nginx restart.

HTTPS not enforced

If HTTPS is not enforced, a client may be tricked into visiting the unsecure (HTTP) version of a website. This would allow an attacker to read and manipulate his messages.

Mitigation

Textual
For each HTTP connection the server must send a response containing:
1. a permanent redirect (i.e. 301 Moved Permanently);
2. a Location field indicating the proper URI to connect to (hostname preceded by *https://*).

Apache

1. open your Apache configuration file (default:
/etc/apache2/sites-available/default-ssl.conf);
2. find the VirtualHost that handles the connections to port 80 (it starts with
<VirtualHost :80);
3. add the string Redirect / https://website where "website" is the URL you want to point the users to (e.g. www.fbk.eu).

N.B. restart the server by typing: sudo service apache2 restart.

nginx

1. In a default situation, you can edit your website configuration
/etc/nginx/sites-enabled/default
(if you changed your site conf name
/etc/nginx/sites-enabled/YOURSITECONFIGURATION);
2. add

```
server {
listen 80 default_server;
server_name _;
return 301 https://$host$request_uri;
}
```

3. remove ALL other listen 80 inside server{...} brackets (except for the previous one)

N.B. restart the server by typing: sudo service nginx restart.