

DVWA

# Security Assessment Findings Report

# Confidentiality Statement

This document is the exclusive property of DVWA and CMKL Security Researchers. This document contains proprietary and confidential information. Duplication, redistribution, or use, in whole or in part, in any form, requires consent of both DVWA and CMKL.

CMKL may share this document with auditors under non-disclosure agreements to demonstrate penetration test requirement compliance.

# Disclaimer

A penetration test is considered a snapshot in time. The findings and recommendations reflect the information gathered during the assessment and not any changes or modifications made outside of that period.

Time-limited engagements do not allow for a full evaluation of all security controls. CMKL prioritized the assessment to identify the weakest security controls an attacker would exploit. CMKL recommends conducting similar assessments on an annual basis by internal or third-party assessors to ensure the continued success of the controls.

# Contact Information

|  |  |  |
| --- | --- | --- |
| **Name** | **Title** | **Contact Information** |
| Orathai Sangpetch | Project Coordinator | orathai@cmkl.ac.th |
| Akkarit Sangpetch | Project Coordinator | akkarit@cmkl.ac.th |
| Ayesh Towheed | Penetration Tester | mtowheed@cmkl.ac.th |
| Khanin Udomchoksakul | Data Scientist | kudomcho@cmkl.ac.th |

# Assessment Overview

SEC engaged CMKL to evaluate the security posture of its infrastructure compared to current industry best practices that included an external penetration test. All testing performed is based on the NIST *NIST SP 800-115 Technical Guide to Information Security Testing and Assessment, OWASP Testing Guide (v4), and customized testing frameworks.*

Phases of penetration testing activities include the following:

Planning – Customer goals are gathered and rules of engagement obtained.

Discovery – Perform scanning and enumeration to identify potential vulnerabilities, weak areas, and exploits.

Attack – Confirm potential vulnerabilities through exploitation and perform additional discovery upon new access

Reporting – Document all found vulnerabilities and exploits, failed attempts, and company strengths and weaknesses.

# Assessment Components

## External Penetration Test

An external penetration test emulates the role of an attacker attempting to gain access to an internal network without internal resources or inside knowledge. A CMKL engineer attempts to gather sensitive information through open-source intelligence (OSINT), including employee information, historical breached passwords, and more that can be leveraged against external systems to gain internal network access. The engineer also performs scanning and enumeration to identify potential vulnerabilities in hopes of exploitation.

# Finding Severity Ratings

The following table defines levels of severity and corresponding CVSS score range that are used throughout the document to assess vulnerability and risk impact.

|  |  |  |
| --- | --- | --- |
| **Severity** | **CVSS V3 Score Range** | **Definition** |
| **Critical** | 9.0-10.0 | Exploitation is straightforward and usually results in system-level compromise. It is advised to form a plan of action and patch immediately. |
| **High** | 7.0-8.9 | Exploitation is more difficult but could cause elevated privileges and potentially a loss of data or downtime. It is advised to form a plan of action and patch as soon as possible. |
| **Moderate** | 4.0-6.9 | Vulnerabilities exist but are not exploitable or require extra steps such as social engineering. It is advised to form a plan of action and patch after high-priority issues have been resolved. |
| **Low** | 0.1-3.9 | Vulnerabilities are non-exploitable but would reduce an organization’s attack surface. It is advised to form a plan of action and patch during the next maintenance window |
| **Informational** | N/A | No vulnerability exists. Additional information is provided regarding items noticed during testing, strong controls, and additional documentation. |

# Scope

|  |  |
| --- | --- |
| **Assessment** | **Details** |
| External Penetration Test | localhost |

## Scope Exclusions

Per client request, CMKL did not perform any Denial of Service or Social Engineering attacks like phishing during testing.

## Client Allowances

DVWA did not provide any allowances to assist the testing.

# Executive Summary

CMKL evaluated DVWA's external security posture through an external network penetration test on Fri May 14 2021 02:33:37 GMT-0400 (Eastern Daylight Time). By leveraging a series of attacks, CMKL found the following results as shown below. It is highly recommended that DVWA address any critical vulnerability found as soon as possible as the vulnerabilities are easily found through basic reconnaissance and exploitable without much effort.

## Attack Summary

The following table describes how CMKL conducted the tests, step by step:

|  |  |  |
| --- | --- | --- |
| **Step** | **Action** | **Recommendation** |
| 1 | Obtained historical breached account credentials to leverage against all company login pages | Discourage employees from using work e-mails and usernames as login credentials to other services unless necessary |
| 2 | Conducted service scanning and vulnerability assessment | Enable service and ping scanning notifications. Ports should be filtered from public using firewalls. Services should be updated frequently |
| 3 | Server scanning for hidden directories and web content and missing headers in requests and responses | Add additional security-related headers to implementation of server. Unused web content and cgi-bins should not be left on the server for anyone to see. |
| 4 | Man-in-the-middle attack proof check by making sure HSTS certificate is present in the response of server | Prevents HTTPS downgrading attacks, should always be present. |

# External Pentest Findings

## Vulnerabilities by CVE IDs

|  |  |  |
| --- | --- | --- |
| **CVE ID** | **CVSS Score** | **Definition** |
| CVE-2017-7679 | 7.5 | "In Apache httpd 2.2.x before 2.2.33 and 2.4.x before 2.4.26, mod\_mime can read one byte past the end of a buffer when sending a malicious Content-Type response header." |
| CVE-2017-7668 | 7.5 | "The HTTP strict parsing changes added in Apache httpd 2.2.32 and 2.4.24 introduced a bug in token list parsing, which allows ap\_find\_token() to search past the end of its input string. By maliciously crafting a sequence of request headers, an attacker may be able to cause a segmentation fault, or to force ap\_find\_token() to return an incorrect value." |
| CVE-2017-3169 | 7.5 | "In Apache httpd 2.2.x before 2.2.33 and 2.4.x before 2.4.26, mod\_ssl may dereference a NULL pointer when third-party modules call ap\_hook\_process\_connection() during an HTTP request to an HTTPS port." |
| CVE-2017-3167 | 7.5 | "In Apache httpd 2.2.x before 2.2.33 and 2.4.x before 2.4.26, use of the ap\_get\_basic\_auth\_pw() by third-party modules outside of the authentication phase may lead to authentication requirements being bypassed." |
| CVE-2019-0211 | 7.2 | "In Apache HTTP Server 2.4 releases 2.4.17 to 2.4.38, with MPM event, worker or prefork, code executing in less-privileged child processes or threads (including scripts executed by an in-process scripting interpreter) could execute arbitrary code with the privileges of the parent process (usually root) by manipulating the scoreboard. Non-Unix systems are not affected." |
| CVE-2018-1312 | 6.8 | "In Apache httpd 2.2.0 to 2.4.29, when generating an HTTP Digest authentication challenge, the nonce sent to prevent reply attacks was not correctly generated using a pseudo-random seed. In a cluster of servers using a common Digest authentication configuration, HTTP requests could be replayed across servers by an attacker without detection." |
| CVE-2017-15715 | 6.8 | "In Apache httpd 2.4.0 to 2.4.29, the expression specified in <FilesMatch> could match '$' to a newline character in a malicious filename, rather than matching only the end of the filename. This could be exploited in environments where uploads of some files are are externally blocked, but only by matching the trailing portion of the filename." |
| CVE-2019-10082 | 6.4 | "In Apache HTTP Server 2.4.18-2.4.39, using fuzzed network input, the http/2 session handling could be made to read memory after being freed, during connection shutdown." |
| CVE-2017-9788 | 6.4 | "In Apache httpd before 2.2.34 and 2.4.x before 2.4.27, the value placeholder in [Proxy-]Authorization headers of type 'Digest' was not initialized or reset before or between successive key=value assignments by mod\_auth\_digest. Providing an initial key with no '=' assignment could reflect the stale value of uninitialized pool memory used by the prior request, leading to leakage of potentially confidential information, and a segfault in other cases resulting in denial of service." |
| CVE-2019-0217 | 6 | "In Apache HTTP Server 2.4 release 2.4.38 and prior, a race condition in mod\_auth\_digest when running in a threaded server could allow a user with valid credentials to authenticate using another username, bypassing configured access control restrictions." |
| CVE-2020-1927 | 5.8 | "In Apache HTTP Server 2.4.0 to 2.4.41, redirects configured with mod\_rewrite that were intended to be self-referential might be fooled by encoded newlines and redirect instead to an an unexpected URL within the request URL." |
| CVE-2019-10098 | 5.8 | "In Apache HTTP server 2.4.0 to 2.4.39, Redirects configured with mod\_rewrite that were intended to be self-referential might be fooled by encoded newlines and redirect instead to an unexpected URL within the request URL." |
| CVE-2020-9490 | 5 | "Apache HTTP Server versions 2.4.20 to 2.4.43. A specially crafted value for the 'Cache-Digest' header in a HTTP/2 request would result in a crash when the server actually tries to HTTP/2 PUSH a resource afterwards. Configuring the HTTP/2 feature via \"H2Push off\" will mitigate this vulnerability for unpatched servers." |
| CVE-2020-1934 | 5 | "In Apache HTTP Server 2.4.0 to 2.4.41, mod\_proxy\_ftp may use uninitialized memory when proxying to a malicious FTP server." |
| CVE-2019-10081 | 5 | "HTTP/2 (2.4.20 through 2.4.39) very early pushes, for example configured with \"H2PushResource\", could lead to an overwrite of memory in the pushing request's pool, leading to crashes. The memory copied is that of the configured push link header values, not data supplied by the client." |
| CVE-2019-0220 | 5 | "A vulnerability was found in Apache HTTP Server 2.4.0 to 2.4.38. When the path component of a request URL contains multiple consecutive slashes ('/'), directives such as LocationMatch and RewriteRule must account for duplicates in regular expressions while other aspects of the servers processing will implicitly collapse them." |
| CVE-2019-0196 | 5 | "A vulnerability was found in Apache HTTP Server 2.4.17 to 2.4.38. Using fuzzed network input, the http/2 request handling could be made to access freed memory in string comparison when determining the method of a request and thus process the request incorrectly." |
| CVE-2018-17199 | 5 | "In Apache HTTP Server 2.4 release 2.4.37 and prior, mod\_session checks the session expiry time before decoding the session. This causes session expiry time to be ignored for mod\_session\_cookie sessions since the expiry time is loaded when the session is decoded." |
| CVE-2018-17189 | 5 | "In Apache HTTP server versions 2.4.37 and prior, by sending request bodies in a slow loris way to plain resources, the h2 stream for that request unnecessarily occupied a server thread cleaning up that incoming data. This affects only HTTP/2 (mod\_http2) connections." |
| CVE-2018-1333 | 5 | "By specially crafting HTTP/2 requests, workers would be allocated 60 seconds longer than necessary, leading to worker exhaustion and a denial of service. Fixed in Apache HTTP Server 2.4.34 (Affected 2.4.18-2.4.30,2.4.33)." |
| CVE-2018-1303 | 5 | "A specially crafted HTTP request header could have crashed the Apache HTTP Server prior to version 2.4.30 due to an out of bound read while preparing data to be cached in shared memory. It could be used as a Denial of Service attack against users of mod\_cache\_socache. The vulnerability is considered as low risk since mod\_cache\_socache is not widely used, mod\_cache\_disk is not concerned by this vulnerability." |
| CVE-2017-9798 | 5 | "Apache httpd allows remote attackers to read secret data from process memory if the Limit directive can be set in a user's .htaccess file, or if httpd.conf has certain misconfigurations, aka Optionsbleed. This affects the Apache HTTP Server through 2.2.34 and 2.4.x through 2.4.27. The attacker sends an unauthenticated OPTIONS HTTP request when attempting to read secret data. This is a use-after-free issue and thus secret data is not always sent, and the specific data depends on many factors including configuration. Exploitation with .htaccess can be blocked with a patch to the ap\_limit\_section function in server/core.c." |
| CVE-2017-7659 | 5 | "A maliciously constructed HTTP/2 request could cause mod\_http2 in Apache HTTP Server 2.4.24, 2.4.25 to dereference a NULL pointer and crash the server process." |
| CVE-2017-15710 | 5 | "In Apache httpd 2.0.23 to 2.0.65, 2.2.0 to 2.2.34, and 2.4.0 to 2.4.29, mod\_authnz\_ldap, if configured with AuthLDAPCharsetConfig, uses the Accept-Language header value to lookup the right charset encoding when verifying the user's credentials. If the header value is not present in the charset conversion table, a fallback mechanism is used to truncate it to a two characters value to allow a quick retry (for example, 'en-US' is truncated to 'en'). A header value of less than two characters forces an out of bound write of one NUL byte to a memory location that is not part of the string. In the worst case, quite unlikely, the process would crash which could be used as a Denial of Service attack. In the more likely case, this memory is already reserved for future use and the issue has no effect at all." |
| CVE-2019-0197 | 4.9 | "A vulnerability was found in Apache HTTP Server 2.4.34 to 2.4.38. When HTTP/2 was enabled for a http: host or H2Upgrade was enabled for h2 on a https: host, an Upgrade request from http/1.1 to http/2 that was not the first request on a connection could lead to a misconfiguration and crash. Server that never enabled the h2 protocol or that only enabled it for https: and did not set \"H2Upgrade on\" are unaffected by this issue." |
| CVE-2020-11993 | 4.3 | "Apache HTTP Server versions 2.4.20 to 2.4.43 When trace/debug was enabled for the HTTP/2 module and on certain traffic edge patterns, logging statements were made on the wrong connection, causing concurrent use of memory pools. Configuring the LogLevel of mod\_http2 above \"info\" will mitigate this vulnerability for unpatched servers." |
| CVE-2019-10092 | 4.3 | "In Apache HTTP Server 2.4.0-2.4.39, a limited cross-site scripting issue was reported affecting the mod\_proxy error page. An attacker could cause the link on the error page to be malformed and instead point to a page of their choice. This would only be exploitable where a server was set up with proxying enabled but was misconfigured in such a way that the Proxy Error page was displayed." |
| CVE-2018-1302 | 4.3 | "When an HTTP/2 stream was destroyed after being handled, the Apache HTTP Server prior to version 2.4.30 could have written a NULL pointer potentially to an already freed memory. The memory pools maintained by the server make this vulnerability hard to trigger in usual configurations, the reporter and the team could not reproduce it outside debug builds, so it is classified as low risk." |
| CVE-2018-1301 | 4.3 | "A specially crafted request could have crashed the Apache HTTP Server prior to version 2.4.30, due to an out of bound access after a size limit is reached by reading the HTTP header. This vulnerability is considered very hard if not impossible to trigger in non-debug mode (both log and build level), so it is classified as low risk for common server usage." |
| CVE-2018-11763 | 4.3 | "In Apache HTTP Server 2.4.17 to 2.4.34, by sending continuous, large SETTINGS frames a client can occupy a connection, server thread and CPU time without any connection timeout coming to effect. This affects only HTTP/2 connections. A possible mitigation is to not enable the h2 protocol." |
| CVE-2018-1283 | 3.5 | "In Apache httpd 2.4.0 to 2.4.29, when mod\_session is configured to forward its session data to CGI applications (SessionEnv on, not the default), a remote user may influence their content by using a \"Session\" header. This comes from the \"HTTP\_SESSION\" variable name used by mod\_session to forward its data to CGIs, since the prefix \"HTTP\_\" is also used by the Apache HTTP Server to pass HTTP header fields, per CGI specifications." |
| CVE-2017-7679 | 7.5 | "In Apache httpd 2.2.x before 2.2.33 and 2.4.x before 2.4.26, mod\_mime can read one byte past the end of a buffer when sending a malicious Content-Type response header." |
| CVE-2017-7668 | 7.5 | "The HTTP strict parsing changes added in Apache httpd 2.2.32 and 2.4.24 introduced a bug in token list parsing, which allows ap\_find\_token() to search past the end of its input string. By maliciously crafting a sequence of request headers, an attacker may be able to cause a segmentation fault, or to force ap\_find\_token() to return an incorrect value." |
| CVE-2017-3169 | 7.5 | "In Apache httpd 2.2.x before 2.2.33 and 2.4.x before 2.4.26, mod\_ssl may dereference a NULL pointer when third-party modules call ap\_hook\_process\_connection() during an HTTP request to an HTTPS port." |
| CVE-2017-3167 | 7.5 | "In Apache httpd 2.2.x before 2.2.33 and 2.4.x before 2.4.26, use of the ap\_get\_basic\_auth\_pw() by third-party modules outside of the authentication phase may lead to authentication requirements being bypassed." |
| CVE-2019-0211 | 7.2 | "In Apache HTTP Server 2.4 releases 2.4.17 to 2.4.38, with MPM event, worker or prefork, code executing in less-privileged child processes or threads (including scripts executed by an in-process scripting interpreter) could execute arbitrary code with the privileges of the parent process (usually root) by manipulating the scoreboard. Non-Unix systems are not affected." |
| CVE-2018-1312 | 6.8 | "In Apache httpd 2.2.0 to 2.4.29, when generating an HTTP Digest authentication challenge, the nonce sent to prevent reply attacks was not correctly generated using a pseudo-random seed. In a cluster of servers using a common Digest authentication configuration, HTTP requests could be replayed across servers by an attacker without detection." |
| CVE-2017-15715 | 6.8 | "In Apache httpd 2.4.0 to 2.4.29, the expression specified in <FilesMatch> could match '$' to a newline character in a malicious filename, rather than matching only the end of the filename. This could be exploited in environments where uploads of some files are are externally blocked, but only by matching the trailing portion of the filename." |
| CVE-2019-10082 | 6.4 | "In Apache HTTP Server 2.4.18-2.4.39, using fuzzed network input, the http/2 session handling could be made to read memory after being freed, during connection shutdown." |
| CVE-2017-9788 | 6.4 | "In Apache httpd before 2.2.34 and 2.4.x before 2.4.27, the value placeholder in [Proxy-]Authorization headers of type 'Digest' was not initialized or reset before or between successive key=value assignments by mod\_auth\_digest. Providing an initial key with no '=' assignment could reflect the stale value of uninitialized pool memory used by the prior request, leading to leakage of potentially confidential information, and a segfault in other cases resulting in denial of service." |
| CVE-2019-0217 | 6 | "In Apache HTTP Server 2.4 release 2.4.38 and prior, a race condition in mod\_auth\_digest when running in a threaded server could allow a user with valid credentials to authenticate using another username, bypassing configured access control restrictions." |
| CVE-2020-1927 | 5.8 | "In Apache HTTP Server 2.4.0 to 2.4.41, redirects configured with mod\_rewrite that were intended to be self-referential might be fooled by encoded newlines and redirect instead to an an unexpected URL within the request URL." |
| CVE-2019-10098 | 5.8 | "In Apache HTTP server 2.4.0 to 2.4.39, Redirects configured with mod\_rewrite that were intended to be self-referential might be fooled by encoded newlines and redirect instead to an unexpected URL within the request URL." |
| CVE-2020-9490 | 5 | "Apache HTTP Server versions 2.4.20 to 2.4.43. A specially crafted value for the 'Cache-Digest' header in a HTTP/2 request would result in a crash when the server actually tries to HTTP/2 PUSH a resource afterwards. Configuring the HTTP/2 feature via \"H2Push off\" will mitigate this vulnerability for unpatched servers." |
| CVE-2020-1934 | 5 | "In Apache HTTP Server 2.4.0 to 2.4.41, mod\_proxy\_ftp may use uninitialized memory when proxying to a malicious FTP server." |
| CVE-2019-10081 | 5 | "HTTP/2 (2.4.20 through 2.4.39) very early pushes, for example configured with \"H2PushResource\", could lead to an overwrite of memory in the pushing request's pool, leading to crashes. The memory copied is that of the configured push link header values, not data supplied by the client." |
| CVE-2019-0220 | 5 | "A vulnerability was found in Apache HTTP Server 2.4.0 to 2.4.38. When the path component of a request URL contains multiple consecutive slashes ('/'), directives such as LocationMatch and RewriteRule must account for duplicates in regular expressions while other aspects of the servers processing will implicitly collapse them." |
| CVE-2019-0196 | 5 | "A vulnerability was found in Apache HTTP Server 2.4.17 to 2.4.38. Using fuzzed network input, the http/2 request handling could be made to access freed memory in string comparison when determining the method of a request and thus process the request incorrectly." |
| CVE-2018-17199 | 5 | "In Apache HTTP Server 2.4 release 2.4.37 and prior, mod\_session checks the session expiry time before decoding the session. This causes session expiry time to be ignored for mod\_session\_cookie sessions since the expiry time is loaded when the session is decoded." |
| CVE-2018-17189 | 5 | "In Apache HTTP server versions 2.4.37 and prior, by sending request bodies in a slow loris way to plain resources, the h2 stream for that request unnecessarily occupied a server thread cleaning up that incoming data. This affects only HTTP/2 (mod\_http2) connections." |
| CVE-2018-1333 | 5 | "By specially crafting HTTP/2 requests, workers would be allocated 60 seconds longer than necessary, leading to worker exhaustion and a denial of service. Fixed in Apache HTTP Server 2.4.34 (Affected 2.4.18-2.4.30,2.4.33)." |
| CVE-2018-1303 | 5 | "A specially crafted HTTP request header could have crashed the Apache HTTP Server prior to version 2.4.30 due to an out of bound read while preparing data to be cached in shared memory. It could be used as a Denial of Service attack against users of mod\_cache\_socache. The vulnerability is considered as low risk since mod\_cache\_socache is not widely used, mod\_cache\_disk is not concerned by this vulnerability." |
| CVE-2017-9798 | 5 | "Apache httpd allows remote attackers to read secret data from process memory if the Limit directive can be set in a user's .htaccess file, or if httpd.conf has certain misconfigurations, aka Optionsbleed. This affects the Apache HTTP Server through 2.2.34 and 2.4.x through 2.4.27. The attacker sends an unauthenticated OPTIONS HTTP request when attempting to read secret data. This is a use-after-free issue and thus secret data is not always sent, and the specific data depends on many factors including configuration. Exploitation with .htaccess can be blocked with a patch to the ap\_limit\_section function in server/core.c." |
| CVE-2017-7659 | 5 | "A maliciously constructed HTTP/2 request could cause mod\_http2 in Apache HTTP Server 2.4.24, 2.4.25 to dereference a NULL pointer and crash the server process." |
| CVE-2017-15710 | 5 | "In Apache httpd 2.0.23 to 2.0.65, 2.2.0 to 2.2.34, and 2.4.0 to 2.4.29, mod\_authnz\_ldap, if configured with AuthLDAPCharsetConfig, uses the Accept-Language header value to lookup the right charset encoding when verifying the user's credentials. If the header value is not present in the charset conversion table, a fallback mechanism is used to truncate it to a two characters value to allow a quick retry (for example, 'en-US' is truncated to 'en'). A header value of less than two characters forces an out of bound write of one NUL byte to a memory location that is not part of the string. In the worst case, quite unlikely, the process would crash which could be used as a Denial of Service attack. In the more likely case, this memory is already reserved for future use and the issue has no effect at all." |
| CVE-2019-0197 | 4.9 | "A vulnerability was found in Apache HTTP Server 2.4.34 to 2.4.38. When HTTP/2 was enabled for a http: host or H2Upgrade was enabled for h2 on a https: host, an Upgrade request from http/1.1 to http/2 that was not the first request on a connection could lead to a misconfiguration and crash. Server that never enabled the h2 protocol or that only enabled it for https: and did not set \"H2Upgrade on\" are unaffected by this issue." |
| CVE-2020-11993 | 4.3 | "Apache HTTP Server versions 2.4.20 to 2.4.43 When trace/debug was enabled for the HTTP/2 module and on certain traffic edge patterns, logging statements were made on the wrong connection, causing concurrent use of memory pools. Configuring the LogLevel of mod\_http2 above \"info\" will mitigate this vulnerability for unpatched servers." |
| CVE-2019-10092 | 4.3 | "In Apache HTTP Server 2.4.0-2.4.39, a limited cross-site scripting issue was reported affecting the mod\_proxy error page. An attacker could cause the link on the error page to be malformed and instead point to a page of their choice. This would only be exploitable where a server was set up with proxying enabled but was misconfigured in such a way that the Proxy Error page was displayed." |
| CVE-2018-1302 | 4.3 | "When an HTTP/2 stream was destroyed after being handled, the Apache HTTP Server prior to version 2.4.30 could have written a NULL pointer potentially to an already freed memory. The memory pools maintained by the server make this vulnerability hard to trigger in usual configurations, the reporter and the team could not reproduce it outside debug builds, so it is classified as low risk." |
| CVE-2018-1301 | 4.3 | "A specially crafted request could have crashed the Apache HTTP Server prior to version 2.4.30, due to an out of bound access after a size limit is reached by reading the HTTP header. This vulnerability is considered very hard if not impossible to trigger in non-debug mode (both log and build level), so it is classified as low risk for common server usage." |
| CVE-2018-11763 | 4.3 | "In Apache HTTP Server 2.4.17 to 2.4.34, by sending continuous, large SETTINGS frames a client can occupy a connection, server thread and CPU time without any connection timeout coming to effect. This affects only HTTP/2 connections. A possible mitigation is to not enable the h2 protocol." |
| CVE-2018-1283 | 3.5 | "In Apache httpd 2.4.0 to 2.4.29, when mod\_session is configured to forward its session data to CGI applications (SessionEnv on, not the default), a remote user may influence their content by using a \"Session\" header. This comes from the \"HTTP\_SESSION\" variable name used by mod\_session to forward its data to CGIs, since the prefix \"HTTP\_\" is also used by the Apache HTTP Server to pass HTTP header fields, per CGI specifications." |

## Data Breach Findings

|  |  |  |
| --- | --- | --- |
| **User Email** | **Database** | **Breach Site** |
| atowheed@gmail.com | HIBP3 | Canva |
| atowheed@gmail.com | HIBP3 | Dailymotion |
| atowheed@gmail.com | HIBP3 | iMesh |
| atowheed@gmail.com | HIBP3 | Zomato |

## Strict Transport Security (HSTS) Check

The target was found to have missing the Strict Transport Security Protocol otherwise known as HSTS which prevents HTTPS downgrading attacks in Man-in-the-middle (MITM) scenarios. This is a really dangerous vulnerability and needs to be fixed at the earliest.

# Cyber Risk & Cyber Confidence Index

## Overview

The Cyber risk is calculated using our pre-trained machine learning model which takes into consideration several factors such as *total vulnerabilities found, severity of vulnerabilities and the availability of exploits for these.*It is an important metric to take into consideration while making IT decisions in the company. It is always recommended to reduce or fix the cyber risk as soon as reported.

The **Cyber Confidence Index** is another factor to keep in mind while making IT decisions. This index comes from a comprehensive formula which takes into account factors such as *the cyber risk and the effort rate.* Together they inform about the consumer's and investor's confidence in the cyber assets of the company in question. Needless to say these should be in good shape through every run of a cyber security assessment so that the confidence does not drop.

The cyber risk and the confidence index by the end of this testing phase is as follows -

|  |  |  |
| --- | --- | --- |
| **Metric** | **Score** | **Meaning** |
| ***Cyber Risk*** | 9.9 | Very High |
| ***Cyber Confidence Index*** | 1.8149999999999997 | Very Bad |