WAS Scan Report

Qualys. Express

Scan Report

05 Nov 2018

Vulnerabilities of all selected scans are consolidated into one report so that you can view their evolution.

Usman Waheed
furth5uw

Fourth
90 Long Acre
null, None wc2e9ra
United Kingdom

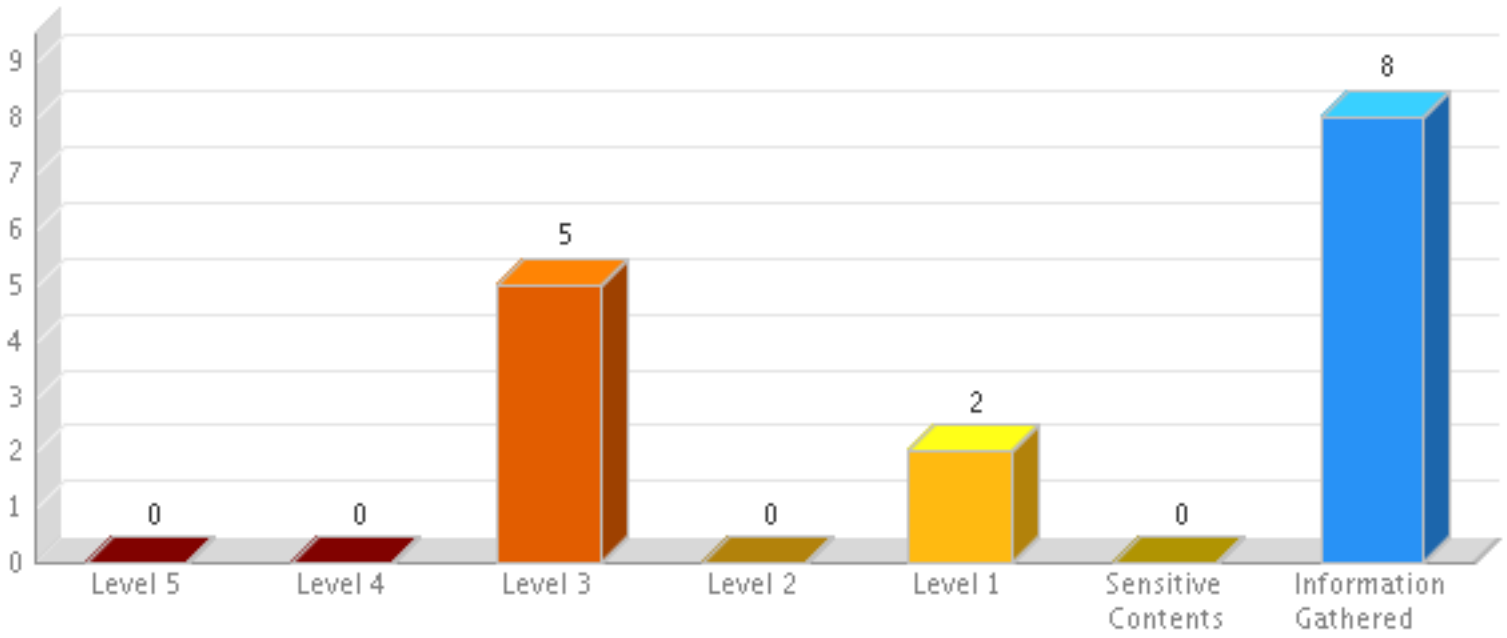
Target and Filters

Scans (1)	Web Application Vulnerability Scan - 2018-11-05
Web Applications (1)	Test

Summary

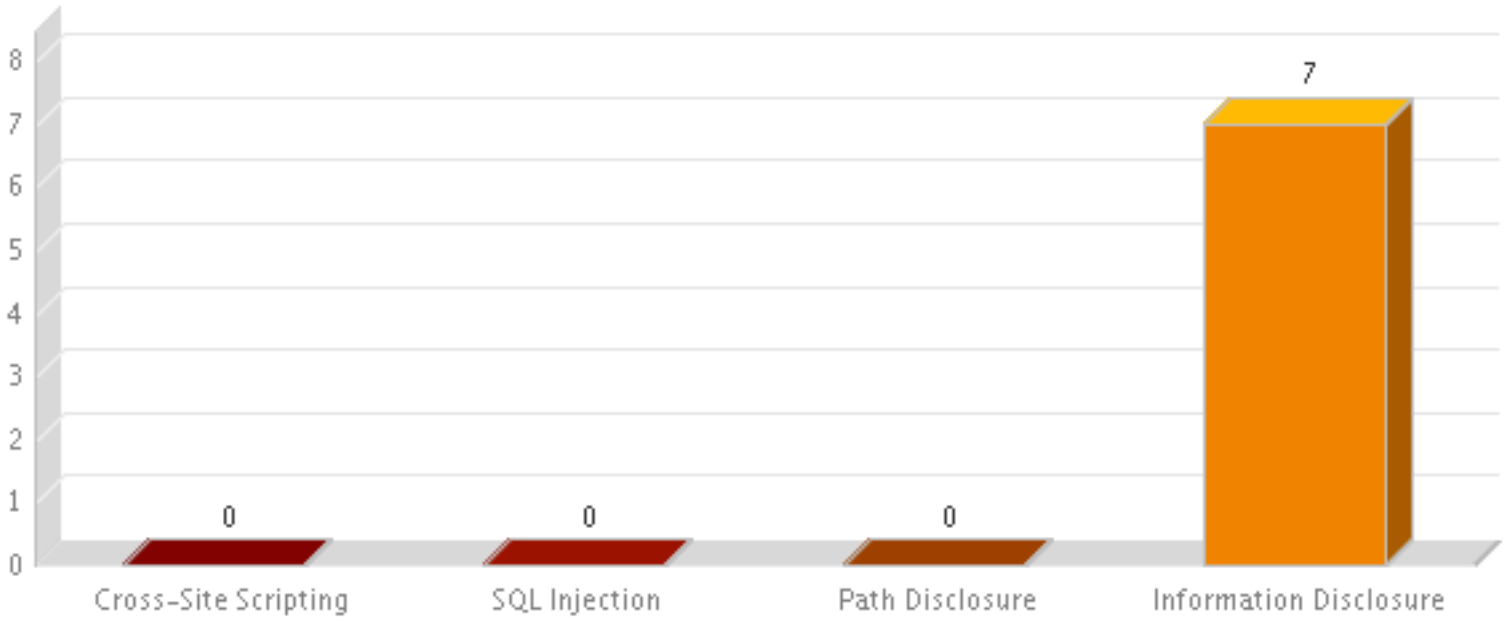
Security Risk	Vulnerabilities	Sensitive Contents	Information Gathered
MED	7	0	8

Findings by Severity

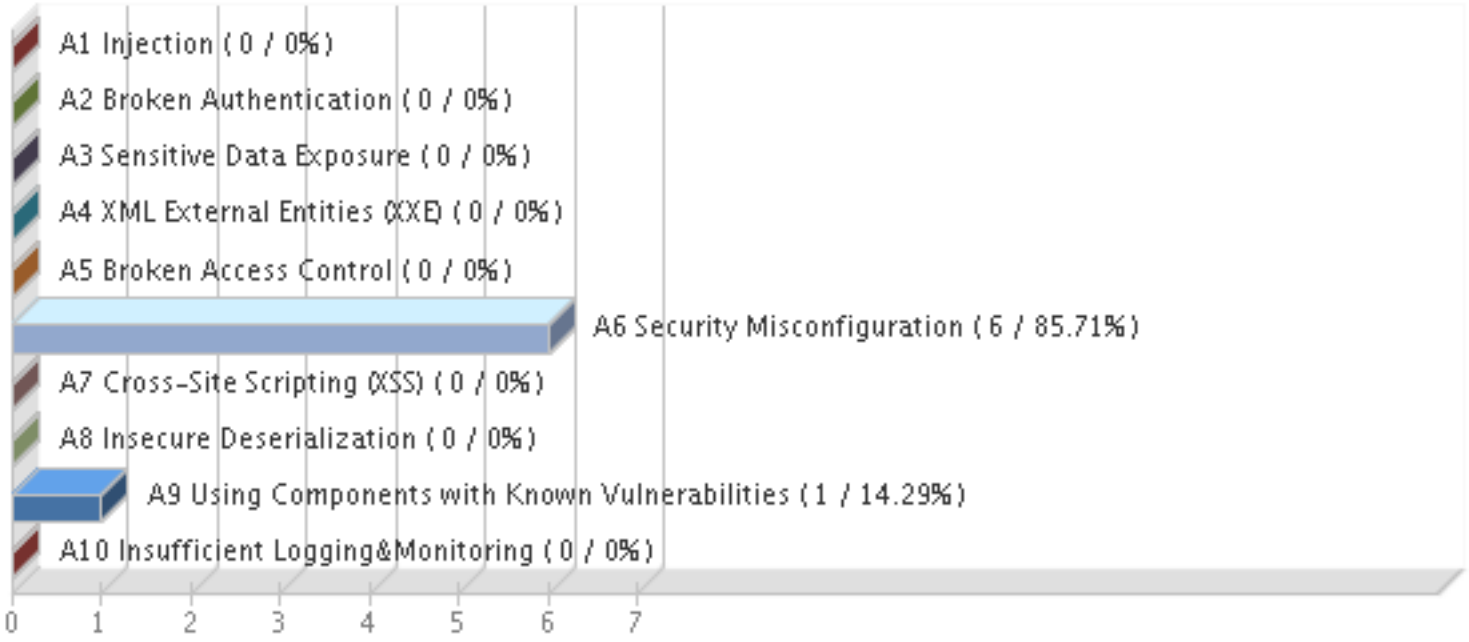


WAS Scan Report

Vulnerabilities by Group



OWASP Top 10 2017 Vulnerabilities



Scan	Date	Level 5	Level 4	Level 3	Level 2	Level 1	Sensitive Contents	Information Gathered
Web Application Vulnerability Scan - 2018-11-05	05 Nov 2018 17:44 GMT	0	0	5	0	2	0	8

WAS Scan Report

Results(15)

Vulnerability (7)

Information Disclosure (7)

150079 Slow HTTP headers vulnerability (1)

150079 Slow HTTP headers vulnerability

URL: http://hotel-test.equalexperts.io/

Finding #	**2244430**(72013695)	**Severity**	Potential Vulnerability - Level 3
Group	Information Disclosure	**First Time Detected**	05 Nov 2018 17:44 GMT
CWE	CWE-772	**Last Time Detected**	05 Nov 2018 17:44 GMT
OWASP	A6 Security Misconfiguration	**Last Time Tested**	05 Nov 2018 17:44 GMT
WASC	WASC-10 Denial Of Service	**Times Detected**	1
CVSS Base	6.8	**CVSS Temporal**	6.1

Details

Threat

The web application is possibly vulnerable to "slow HTTP headers" Denial of Service (DoS) attack. This is an application-level DoS, that occurs when an attacker holds server connections open by sending partial HTTP requests, and continues to send subsequent headers at some interval to prevent the server from closing sockets. In this way, the web server becomes unavailable because the number of available sockets decreases and memory usage may increase, especially if the server allocates a thread per connection. One of the reasons for this behavior is that some servers have "no data" timers, that reset each time a byte arrives at the socket, but the server does not enforce an overall time limit for a connection. For example, the attacker sends the data for its request one byte at a time over several minutes rather than following the expected behavior of transmitting a complete request of several hundred bytes in a single packet. This enables the attacker to prolong the connection virtually forever. More information can be found at the Slowloris HTTP DoS.

Impact

All other services remain intact but the web server itself becomes completely inaccessible.

Solution

Server-specific recommendations can be found here. Countermeasures for Apache are described here. Easy to use tool for intrusive testing is available here.

Detection Information

Parameter	No param has been required for detecting the information.
Authentication	In order to detect this vulnerability, no authentication has been required.

Payloads

#1 Request

Payload	N/A
Request	POST http://hotel-test.equalexperts.io/

Click this link to try to reproduce the vulnerability using above payload.Note that clicking this link may not lead to visible results, either because the vulnerability requires context to be previously set (authentication, cookies...) or because the exploitation of the vulnerability does not lead to any visible proof.

#1 Response

Vulnerable to slow HTTP headers attack

Server resets timeout after accepting header data from peer.

150085 Slow HTTP POST vulnerability (1)

WAS Scan Report

150085 Slow HTTP POST vulnerability

URL: http://hotel-test.equalexperts.io/

Finding #	**2244429**(72013694)	**Severity**	Potential Vulnerability - Level 3
Group	Information Disclosure	**First Time Detected**	05 Nov 2018 17:44 GMT
CWE	CWE-772	**Last Time Detected**	05 Nov 2018 17:44 GMT
OWASP	A6 Security Misconfiguration	**Last Time Tested**	05 Nov 2018 17:44 GMT
WASC	WASC-10 Denial Of Service	**Times Detected**	1
CVSS Base	6.8 **CVSS Temporal** 6.1		

Details

Threat

The web application is possibly vulnerable to a "slow HTTP POST" Denial of Service (DoS) attack. This is an application-level DoS that consumes server resources by maintaining open connections for an extended period of time by slowly sending traffic to the server. If the server maintains too many connections open at once, then it may not be able to respond to new, legitimate connections. Unlike bandwidth-consumption DoS attacks, the "slow" attack does not require a large amount of traffic to be sent to the server -- only that the client is able to maintain open connections for several minutes at a time.

The attack holds server connections open by sending properly crafted HTTP POST headers that contain a Content-Length header with a large value to inform the web server how much of data to expect. After the HTTP POST headers are fully sent, the HTTP POST message body is sent at slow speeds to prolong the completion of the connection and lock up server resources. By waiting for the complete request body, the server is helping clients with slow or intermittent connections to complete requests, but is also exposing itself to abuse.

Further information can be found under BlackHat_DC_2011_Brennan_Denial_Service-Slides.pdf.

Impact

All other services remain intact but the web server itself becomes inaccessible.

Solution

Solution would be server-specific, but general recommendations are: - to limit the size of the acceptable request to each form requirements - establish minimal acceptable speed rate - establish absolute request timeout for connection with POST request Server-specific details can be found here. A tool that demonstrates this vulnerability in a more intrusive manner is available here.

Detection Information

Parameter	No param has been required for detecting the information.
Authentication	In order to detect this vulnerability, no authentication has been required.

Payloads

#1 Request

Payload	N/A
Request	POST http://hotel-test.equalexperts.io/

Click this link to try to reproduce the vulnerability using above payload.Note that clicking this link may not lead to visible results, either because the vulnerability requires context to be previously set (authentication, cookies...) or because the exploitation of the vulnerability does not lead to any visible proof.

#1 Response

Vulnerable to slow HTTP POST attack
Connection with partial POST body remained open for: 131958 milliseconds
Server resets timeout after accepting request data from peer.

150124 Clickjacking - Framable Page (2)

WAS Scan Report

150124 Clickjacking - Framable Page

URL: http://hotel-test.equalexperts.io/

Finding #	**2244432**(72013697)	**Severity**	Confirmed Vulnerability - Level 3
Group	Information Disclosure	**First Time Detected**	05 Nov 2018 17:44 GMT
CWE	CWE-451	**Last Time Detected**	05 Nov 2018 17:44 GMT
OWASP	A6 Security Misconfiguration	**Last Time Tested**	05 Nov 2018 17:44 GMT
WASC	WASC-15 Application Misconfiguration	**Times Detected**	1
CVSS Base	6.4 **CVSS Temporal** 5.8		

Details

Threat

The page can be easily framed. Anti-framing measures are not used.

Impact

Clickjacking and Cross-Site Request Forgery (CSRF) can be performed by framing the target site. An attack can trick the user into clicking on the link by framing the original page and showing a layer on top of it with dummy buttons.

Solution

Two of the most popular prevention are: X-Frame-Options: This header works with modern browsers and can be used to prevent framing of the page. Note that is must be an HTTP header, the setting is ignored if it is created as an "http-equiv" meta element within the page. Framekiller: JavaScript code that prevents the malicious user from framing the page.

Detection Information

Parameter	No param has been required for detecting the information.
Authentication	In order to detect this vulnerability, no authentication has been required.

Payloads

#1 Request

Payload	N/A
Request	GET http://hotel-test.equalexperts.io/

Click this link to try to reproduce the vulnerability using above payload.Note that clicking this link may not lead to visible results, either because the vulnerability requires context to be previously set (authentication, cookies...) or because the exploitation of the vulnerability does not lead to any visible proof.

#1 Response

The URI was framed.

WAS Scan Report

150124 Clickjacking - Framable Page

URL: http://hotel-test.equalexperts.io/./

Finding #	**2244434**(72013699)	**Severity**	Confirmed Vulnerability - Level 3
Group	Information Disclosure	**First Time Detected**	05 Nov 2018 17:44 GMT
CWE	CWE-451	**Last Time Detected**	05 Nov 2018 17:44 GMT
OWASP	A6 Security Misconfiguration	**Last Time Tested**	05 Nov 2018 17:44 GMT
WASC	WASC-15 Application Misconfiguration	**Times Detected**	1
CVSS Base	6.4 **CVSS Temporal** 5.8		

Details

Threat

The page can be easily framed. Anti-framing measures are not used.

Impact

Clickjacking and Cross-Site Request Forgery (CSRF) can be performed by framing the target site. An attack can trick the user into clicking on the link by framing the original page and showing a layer on top of it with dummy buttons.

Solution

Two of the most popular prevention are: X-Frame-Options: This header works with modern browsers and can be used to prevent framing of the page. Note that is must be an HTTP header, the setting is ignored if it is created as an "http-equiv" meta element within the page. Framekiller: JavaScript code that prevents the malicious user from framing the page.

Detection Information

Parameter	No param has been required for detecting the information.
Authentication	In order to detect this vulnerability, no authentication has been required.

Payloads

#1 Request

Payload	N/A
Request	GET http://hotel-test.equalexperts.io/./

Click this link to try to reproduce the vulnerability using above payload.Note that clicking this link may not lead to visible results, either because the vulnerability requires context to be previously set (authentication, cookies...) or because the exploitation of the vulnerability does not lead to any visible proof.

#1 Response

The URI was framed.

150162 Use of JavaScript Library with Known Vulnerability (1)

WAS Scan Report

150162 Use of JavaScript Library with Known Vulnerability

URL: http://hotel-test.equalexperts.io/

Finding #	**2244431**(72013696)	**Severity**	Confirmed Vulnerability - Level 3
Group	Information Disclosure	**First Time Detected**	05 Nov 2018 17:44 GMT
CWE	CWE-937	**Last Time Detected**	05 Nov 2018 17:44 GMT
OWASP	A9 Using Components with Known Vulnerabilities	**Last Time Tested**	05 Nov 2018 17:44 GMT
WASC	-	**Times Detected**	1
CVSS Base	6.4	**CVSS Temporal**	4.9

Details

Threat

The web application is using one or more JavaScript libraries which contain a known vulnerability.

Impact

Attackers could exploit the vulnerabilities in the JavaScript libraries. The exact ability to exploit and impact depends on the component(s) and usage of the vulnerable libraries by the web application.

Solution

Please refer to vendor's security advisories related to the vulnerable version of the library.

Detection Information

Parameter	No param has been required for detecting the information.
Authentication	In order to detect this vulnerability, no authentication has been required.

Payloads

WAS Scan Report

#1 Request

Payload -

Request GET http://hotel-test.equalexperts.io/

Click this link to try to reproduce the vulnerability using above payload.Note that clicking this link may not lead to visible results, either because the vulnerability requires context to be previously set (authentication, cookies...) or because the exploitation of the vulnerability does not lead to any visible proof.

#1 Response

Vulnerable javascript library: jQuery
version: 2.2.3
script uri: https://code.jquery.com/jquery-2.2.3.min.js

Details:
jQuery versions on or above 1.4.0 and below 1.12.0 (version 1.12.3 and above but below 3.0.0-beta1 as well) are vulnerable to XSS via 3rd party text/javascript responses(3rd party CORS request may execute). (https://github.com/jquery/jquery/issues/2432).
Solution: jQuery version 3.0.0 has been released to address the issue (http://blog.jquery.com/2016/01/08/jquery-2-2-and-1-12-released/). Please refer to vendor documentation (https://blog.jquery.com/) for the latest security updates.

In jQuery versions on and above 2.2.2 and below 3.0.0 $.parseHTML has (lots of) XSS. In these versions parseHTML() executes scripts in event handlers. Please refer following resource for more details: https://bugs.jquery.com/ticket/11974, http://research.insecurelabs.org/jquery/test/

Found on the following pages (only first 10 pages are reported):
http://hotel-test.equalexperts.io/
http://hotel-test.equalexperts.io/./

==========

Vulnerable javascript library: jQuery.ui.dialog
version: 1.11.4
script uri: https://code.jquery.com/ui/1.11.4/jquery-ui.min.js

Details:
jquery.ui.dialog version below 1.12.0 is vulnerable to XSS if the user input is allowed to pass through to the closeText property. Please refer vendor documentatation (https://github.com/jquery/api.jqueryui.com/issues/281)for latest security updates.

Found on the following pages (only first 10 pages are reported):
http://hotel-test.equalexperts.io/
http://hotel-test.equalexperts.io/./

==========

Vulnerable javascript library: Bootstrap
version: 3.3.6
script uri: https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js

Details:
The data-target attribute in bootstrap versions below 3.4.0 is vulnerable to Cross-Site Scripting(XSS) attacks. Please refer to vendor documentation (https://github.com/twbs/bootstrap/pull/23687, https://github.com/twbs/bootstrap/issues/20184) for the latest security updates.

Found on the following pages (only first 10 pages are reported):
http://hotel-test.equalexperts.io/
http://hotel-test.equalexperts.io/./

150081 X-Frame-Options header is not set (2)

WAS Scan Report

150081 X-Frame-Options header is not set

URL: http://hotel-test.equalexperts.io/

Finding #	**2244433**(72013698)	**Severity**	Potential Vulnerability - Level 1
Group	Information Disclosure	**First Time Detected**	05 Nov 2018 17:44 GMT
CWE	CWE-693	**Last Time Detected**	05 Nov 2018 17:44 GMT
OWASP	A6 Security Misconfiguration	**Last Time Tested**	05 Nov 2018 17:44 GMT
WASC	WASC-15 Application Misconfiguration	**Times Detected**	1
CVSS Base	5 **CVSS Temporal** 4.1		

Details

Threat

The X-Frame-Options header is not set in the HTTP response, which may lead to a possible framing of the page. An attacker can trick users into clicking on a malicious link by framing the original page and showing a layer on top of it with legitimate-looking buttons.

Impact

Attacks such as Clickjacking could potentially be performed.

Solution

In the HTTP response, set X-Frame-Options: DENY or X-Frame-Options: SAMEORIGIN. This header works with modern browsers and can be used to prevent framing of the page. For more information, see https://www.owasp.org/index.php/Clickjacking_Defense_Cheat_Sheet.

Detection Information

Parameter	No param has been required for detecting the information.
Authentication	In order to detect this vulnerability, no authentication has been required.

Payloads

#1 Request

Payload	N/A
Request	GET http://hotel-test.equalexperts.io/

Click this link to try to reproduce the vulnerability using above payload.Note that clicking this link may not lead to visible results, either because the vulnerability requires context to be previously set (authentication, cookies...) or because the exploitation of the vulnerability does not lead to any visible proof.

#1 Response

The response for this request either did not have an "X-FRAME-OPTIONS" header present or was not set to DENY or SAMEORIGIN

WAS Scan Report

150081 X-Frame-Options header is not set

URL: http://hotel-test.equalexperts.io/./

Finding #	**2244435**(72013700)	**Severity**	Potential Vulnerability - Level 1
Group	Information Disclosure	**First Time Detected**	05 Nov 2018 17:44 GMT
CWE	CWE-693	**Last Time Detected**	05 Nov 2018 17:44 GMT
OWASP	A6 Security Misconfiguration	**Last Time Tested**	05 Nov 2018 17:44 GMT
WASC	WASC-15 Application Misconfiguration	**Times Detected**	1
CVSS Base	5 **CVSS Temporal** 4.1		

Details

Threat

The X-Frame-Options header is not set in the HTTP response, which may lead to a possible framing of the page. An attacker can trick users into clicking on a malicious link by framing the original page and showing a layer on top of it with legitimate-looking buttons.

Impact

Attacks such as Clickjacking could potentially be performed.

Solution

In the HTTP response, set X-Frame-Options: DENY or X-Frame-Options: SAMEORIGIN. This header works with modern browsers and can be used to prevent framing of the page. For more information, see https://www.owasp.org/index.php/Clickjacking_Defense_Cheat_Sheet.

Detection Information

Parameter	No param has been required for detecting the information.
Authentication	In order to detect this vulnerability, no authentication has been required.
Access Path	Here is the path followed by the scanner to reach the exploitable URL:

http://hotel-test.equalexperts.io/

Payloads

#1 Request

Payload	N/A
Request	GET http://hotel-test.equalexperts.io/./

Click this link to try to reproduce the vulnerability using above payload.Note that clicking this link may not lead to visible results, either because the vulnerability requires context to be previously set (authentication, cookies...) or because the exploitation of the vulnerability does not lead to any visible proof.

#1 Response

The response for this request either did not have an "X-FRAME-OPTIONS" header present or was not set to DENY or SAMEORIGIN

Information Gathered (8)

Information Gathered (8)

150019 Server Returned Unexpected Response Code (1)

WAS Scan Report

150019 Server Returned Unexpected Response Code

Finding #	**1706168**(72013686)	**Severity**	Information Gathered - Level 2
Group	Information Gathered	**Detection Date**	05 Nov 2018 17:44 GMT
CWE	-		
OWASP	-		
WASC	-		

Details

Threat

The Web server returned an HTTP response code that was unexpected or not handled by the Web application scanning engine. The scan continued, but the response may be indicative of connection or Web server errors.

Impact

An unexpected or unhandled response code might be indicative of a problem in the Web server or the crawling process.

Solution

Verify that the HTTP response code is not the result of a Web server error.

Results

418: http://hotel-test.equalexperts.io/booking/45723
418: http://hotel-test.equalexperts.io/booking/45724

6 DNS Host Name (1)

6 DNS Host Name

Finding #	**1706169**(72013687)	**Severity**	Information Gathered - Level 1
Group	Information Gathered	**Detection Date**	05 Nov 2018 17:44 GMT
CWE	-		
OWASP	-		
WASC	-		

Details

Threat

The fully qualified domain name of this host, if it was obtained from a DNS server, is displayed in the RESULT section.

Impact

N/A

Solution

N/A

Results

IP address	Host name
52.208.166.189	ec2-52-208-166-189.eu-west-1.compute.amazonaws.com

150009 Links Crawled (1)

WAS Scan Report

150009 Links Crawled

Finding #	**1706173**(72013692)	**Severity**	Information Gathered - Level 1
Group	Information Gathered	**Detection Date**	05 Nov 2018 17:44 GMT
CWE	-		
OWASP	-		
WASC	-		

Details

Threat

The list of unique links crawled and HTML forms submitted by the Web application scanner appear in the Results section. This list may contain fewer links than the maximum threshold defined at scan launch. The maximum links to crawl includes links in this list and requests for the same link made as an anonymous and authenticated user.

Impact

N/A

Solution

N/A

Results

Duration of crawl phase (seconds): 93.00
Number of links: 5
(This number excludes form requests and links re-requested during authentication.)

http://hotel-test.equalexperts.io/
http://hotel-test.equalexperts.io/./
http://hotel-test.equalexperts.io/booking
http://hotel-test.equalexperts.io/booking/45723
http://hotel-test.equalexperts.io/booking/45724

150010 External Links Discovered (1)

WAS Scan Report

150010 External Links Discovered

Finding #	**1706171**(72013690)	**Severity**	Information Gathered - Level 1
Group	Information Gathered	**Detection Date**	05 Nov 2018 17:44 GMT
CWE	-		
OWASP	-		
WASC	-		

Details

Threat

The external links discovered by the Web application scanning engine are provided in the Results section. These links were present on the target Web application, but were not crawled.

Impact

N/A

Solution

N/A

Results

Number of links: 6
https://code.jquery.com/jquery-2.2.3.min.js
https://code.jquery.com/ui/1.11.4/jquery-ui.min.js
https://code.jquery.com/ui/1.11.4/themes/ui-lightness/images/ui-bg_highlight-soft_100_eeeeee_1x100.png
https://code.jquery.com/ui/1.11.4/themes/ui-lightness/jquery-ui.css
https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css
https://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/js/bootstrap.min.js

150020 Links Rejected By Crawl Scope or Exclusion List (1)

WAS Scan Report

150020 Links Rejected By Crawl Scope or Exclusion List

Finding #	**1705789**(72013688)	**Severity**	Information Gathered - Level 1
Group	Information Gathered	**Detection Date**	05 Nov 2018 17:44 GMT
CWE	-		
OWASP	-		
WASC	-		

Details

Threat

One or more links were not crawled because of an explicit rule to exclude them. This also occurs if a link is malformed.

Black list and white list entries can cause links to be rejected. If a scan is limited to a specific starting directory, then links outside that directory will neither be crawled or tested.

Links that contain a host name or IP address different from the target application are considered external links and not crawled by default; those types of links are not listed here. This often happens when the scope of a scan is limited to the directory of the starting URL. The scope can be changed in the Web Application Record.

During the test phase, some path-based tests may be rejected if the scan is limited to the directory of the starting URL and the test would fall outside that directory. In these cases, the number of rejected links may be too high to list in the Results section.

Impact

Links listed here were neither crawled or tested by the Web application scanning engine.

Solution

A link might have been intentionally matched by a black or white list entry. Verify that no links in this list were unintentionally rejected.

Results

Links rejected during the test phase not reported due to volume of links.

150021 Scan Diagnostics (1)

WAS Scan Report

150021 Scan Diagnostics

Finding #	**1706170**(72013689)	**Severity**	Information Gathered - Level 1
Group	Information Gathered	**Detection Date**	05 Nov 2018 17:44 GMT
CWE	-		
OWASP	-		
WASC	-		

Details

Threat

This check provides various details of the scan's performance and behavior. In some cases, this check can be used to identify problems that the scanner encountered when crawling the target Web application.

Impact

The scan diagnostics data provides technical details about the crawler's performance and behavior. This information does not necessarily imply problems with the Web application.

Solution

No action is required.

Results

Loaded 0 blacklist entries.
Loaded 0 whitelist entries.
HTML form authentication unavailable, no WEBAPP entry found
Batch #0 VirtualHostDiscovery: estimated time < 10 minutes (70 tests, 0 inputs)
VirtualHostDiscovery: 70 vulnsigs tests, completed 70 requests, 4 seconds. Completed 70 requests of 70 estimated requests (100%). All tests completed.
Batch #0 CMSDetection: estimated time < 1 minute (1 tests, 1 inputs)
[CMSDetection phase] : No potential CMS found. Aborting the CMS Detection phaseCMSDetection: 1 vulnsigs tests, completed 38 requests, 1 seconds. Completed 38 requests of 38 estimated requests (100%). All tests completed.
Collected 6 links overall in 0 hours 1 minutes duration.
Path manipulation: Estimated requests (payloads x links): files with extension:(0 x 0) + files:(0 x 3) + directories:(9 x 3) + paths:(0 x 6) = total (27)
Batch #0 WS Directory Path manipulation: estimated time < 1 minute (9 tests, 6 inputs)
WS Directory Path manipulation: 9 vulnsigs tests, completed 27 requests, 0 seconds. Completed 27 requests of 27 estimated requests (100%). All tests completed.
Batch #0 WS enumeration: estimated time < 1 minute (11 tests, 6 inputs)
WS enumeration: 11 vulnsigs tests, completed 35 requests, 1 seconds. Completed 35 requests of 66 estimated requests (53.0303%). All tests completed.
No XML requests found. Skipping XXE tests.
Batch #4 DOM XSS exploitation: estimated time < 1 minute (4 tests, 0 inputs)
Batch #4 DOM XSS exploitation: 4 vulnsigs tests, completed 0 requests, 0 seconds. No tests to execute.
Batch #4 HTTP call manipulation: estimated time < 1 minute (33 tests, 0 inputs)
Batch #4 HTTP call manipulation: 33 vulnsigs tests, completed 0 requests, 0 seconds. No tests to execute.
Batch #4 Open Redirect analysis: estimated time < 1 minute (1 tests, 0 inputs)
Batch #4 Open Redirect analysis: 1 vulnsigs tests, completed 0 requests, 0 seconds. No tests to execute.
CSRF tests will not be launched because the scan is not successfully authenticated.
Batch #4 File Inclusion analysis: estimated time < 1 minute (1 tests, 5 inputs)
Batch #4 File Inclusion analysis: 1 vulnsigs tests, completed 0 requests, 0 seconds. Completed 0 requests of 5 estimated requests (0%). All tests completed.
Batch #4 Cookie manipulation: estimated time < 1 minute (37 tests, 0 inputs)
Batch #4 Cookie manipulation: 37 vulnsigs tests, completed 0 requests, 0 seconds. No tests to execute.
Batch #4 Header manipulation: estimated time < 1 minute (37 tests, 5 inputs)
Batch #4 Header manipulation: 37 vulnsigs tests, completed 110 requests, 2 seconds. Completed 110 requests of 250 estimated requests (44%). XSS optimization removed 120 links. All tests completed.
Batch #4 shell shock detector: estimated time < 1 minute (1 tests, 5 inputs)
Batch #4 shell shock detector: 1 vulnsigs tests, completed 5 requests, 0 seconds. Completed 5 requests of 5 estimated requests (100%). All tests completed.
Batch #4 shell shock detector(form): estimated time < 1 minute (1 tests, 0 inputs)
Batch #4 shell shock detector(form): 1 vulnsigs tests, completed 0 requests, 0 seconds. No tests to execute.
Batch #4 httpoxy detector: estimated time < 1 minute (1 tests, 5 inputs)
Batch #4 httpoxy detector: 1 vulnsigs tests, completed 5 requests, 0 seconds. Completed 5 requests of 5 estimated requests (100%). All tests completed.
Batch #4 httpoxy detector(form): estimated time < 1 minute (1 tests, 0 inputs)
Batch #4 httpoxy detector(form): 1 vulnsigs tests, completed 0 requests, 0 seconds. No tests to execute.
Batch #4 Struts timebased detector: estimated time < 1 minute (1 tests, 5 inputs)
Batch #4 Struts timebased detector: 1 vulnsigs tests, completed 5 requests, 0 seconds. Completed 5 requests of 5 estimated requests (100%). All tests completed.
Batch #5 HTTP Time Bandit: estimated time < 1 minute (1 tests, 10 inputs)
Batch #5 HTTP Time Bandit: 1 vulnsigs tests, completed 0 requests, 0 seconds. No tests to execute.
Path manipulation: Estimated requests (payloads x links): files with extension:(0 x 0) + files:(0 x 3) + directories:(4 x 3) + paths:(11 x 6) = total (78)
Batch #5 Path XSS manipulation: estimated time < 1 minute (15 tests, 6 inputs)
Batch #5 Path XSS manipulation: 15 vulnsigs tests, completed 66 requests, 1 seconds. Completed 66 requests of 78 estimated requests (84.6154%). All tests completed.
Path manipulation: Estimated requests (payloads x links): files with extension:(0 x 0) + files:(0 x 3) + directories:(1 x 3) + paths:(0 x 6) = total (3)
Batch #5 Tomcat Vuln manipulation: estimated time < 1 minute (1 tests, 6 inputs)
Batch #5 Tomcat Vuln manipulation: 1 vulnsigs tests, completed 3 requests, 0 seconds. Completed 3 requests of 3 estimated requests (100%). All tests completed.
Path manipulation: Estimated requests (payloads x links): files with extension:(0 x 0) + files:(0 x 3) + directories:(16 x 3) + paths:(0 x 6) = total (48)
Batch #5 Time based path manipulation: estimated time < 1 minute (16 tests, 5 inputs)
Batch #5 Time based path manipulation: 16 vulnsigs tests, completed 64 requests, 880 seconds. Completed 64 requests of 48 estimated requests (133.333%). All tests completed.
Path manipulation: Estimated requests (payloads x links): files with extension:(4 x 0) + files:(16 x 3) + directories:(102 x 3) + paths:(14 x 6) = total (438)

WAS Scan Report

Batch #5 Path manipulation: estimated time < 1 minute (136 tests, 6 inputs)
Batch #5 Path manipulation: 136 vulnsigs tests, completed 372 requests, 2 seconds. Completed 372 requests of 438 estimated requests (84.9315%). All tests completed.
Generic WebCgi Test no test enabled
Total requests made: 818
Average server response time: 0.06 seconds

150041 Links Rejected (1)

150041 Links Rejected

Finding #	**1705790**(72013693)	**Severity**	Information Gathered - Level 1
Group	Information Gathered	**Detection Date**	05 Nov 2018 17:44 GMT
CWE	-		
OWASP	-		
WASC	-		

Details

Threat

This has an informative nature. The links listed below were not crawled by the Web application scanning engine because they were intentionally prohibited by a blacklist or whitelist configuration setting. The list is provided to verify that links have been correctly blocked by blacklist and whitelist filters.

Impact

Links listed here were neither crawled or tested by the Web application scanning engine, and that should be in sync with the intended behavior.

Solution

No action is required.

Results

http://hotel-test.equalexperts.io/.zip
http://hotel-test.equalexperts.io/..zip
http://hotel-test.equalexperts.io/booking.zip
http://hotel-test.equalexperts.io/booking/45723.zip
http://hotel-test.equalexperts.io/booking/45724.zip

150152 Forms Crawled (1)

WAS Scan Report

150152 Forms Crawled

Finding #	**1706172**(72013691)	**Severity**	Information Gathered - Level 1
Group	Information Gathered	**Detection Date**	05 Nov 2018 17:44 GMT
CWE	-		
OWASP	-		
WASC	-		

Details

Threat

The list and details of unique forms submitted by the Web application scanner appear in the Results section. This list does not contain authentication forms (i.e. login forms) which are reported separately in QID 150115.

Impact

N/A

Solution

N/A

Results

Total internal forms seen (this count includes duplicate forms): 0

Crawled forms (Total: 0)
NOTE: This does not include authentication forms. Authentication forms are reported separately in QID 150115

WAS Scan Report

Appendix

Scan Details

Web Application Vulnerability Scan - 2018-11-05

Reference	was/1541439829349.528613
Date	05 Nov 2018 17:44 GMT
Mode	On-Demand
Type	Vulnerability
Authentication	None
Scanner Appliance	External (IP: 154.59.121.140, Scanner: 10.4.47-1, WAS: 6.2.51-1, Signatures: 2.4.458-2)
Profile	Initial WAS Options
DNS Override	-
Duration	00:18:40
Status	Finished
Authentication Status	None

Option Profile Details

Form Submission	BOTH
Form Crawl Scope	Do not include form action URI in uniqueness calculation
Maximum links to test in scope	300
User Agent	-
Request Parameter Set	Initial Parameters
Document Type	Ignore common binary files
SmartScan Support	Disabled
Timeout Error Threshold	100
Unexpected Error Threshold	300
Scan Intensity	Low
Bruteforce Option	Minimal
Detection Scope	Core
Credit Card Numbers Search	Off
Social Security Numbers (US) Search	Off

Web Application Details: Test

Name	Test
URL	http://hotel-test.equalexperts.io/
Owner	Usman Waheed (furth5uw)
Scope	Limit to URL hostname
Operating System	-

Severity Levels

Confirmed Vulnerabilities

Vulnerabilities (QIDs) are design flaws, programming errors, or mis-configurations that make your web application and web application platform susceptible to malicious attacks. Depending on the level of the security risk, the successful exploitation of a vulnerability can vary from the disclosure of information to a complete compromise of the web application and/or the web application platform. Even if the web application isn't fully compromised, an exploited vulnerability could still lead to the web application being used to launch attacks against users of the site.

WAS Scan Report

Minimal	Basic information disclosure (e.g. web server type, programming language) might enable intruders to discover other vulnerabilities, but lack of this information does not make the vulnerability harder to find.
Medium	Intruders may be able to collect sensitive information about the application platform, such as the precise version of software used. With this information, intruders can easily exploit known vulnerabilities specific to software versions. Other types of sensitive information might disclose a few lines of source code or hidden directories.
Serious	Vulnerabilities at this level typically disclose security-related information that could result in misuse or an exploit. Examples include source code disclosure or transmitting authentication credentials over non-encrypted channels.
Critical	Intruders can exploit the vulnerability to gain highly sensitive content or affect other users of the web application. Examples include certain types of cross-site scripting and SQL injection attacks.
Urgent	Intruders can exploit the vulnerability to compromise the web application's data store, obtain information from other users' accounts, or obtain command execution on a host in the web application's architecture.

Potential Vulnerabilities

Potential Vulnerabilities indicate that the scanner observed a weakness or error that is commonly used to attack a web application, and the scanner was unable to confirm if the weakness or error could be exploited. Where possible, the QID's description and results section include information and hints for following-up with manual analysis. For example, the exploitability of a QID may be influenced by characteristics that the scanner cannot confirm, such as the web application's network architecture, or the test to confirm exploitability requires more intrusive testing than the scanner is designed to conduct.

Minimal	Presence of this vulnerability is indicative of basic information disclosure (e.g. web server type, programming language) and might enable intruders to discover other vulnerabilities. For example in this scenario, information such as web server type, programming language, passwords or file path references can be disclosed.
Medium	Presence of this vulnerability is indicative of basic information disclosure (e.g. web server type, programming language) and might enable intruders to discover other vulnerabilities. For example version of software or session data can be disclosed, which could be used to exploit.
Serious	Presence of this vulnerability might give access to security-related information to intruders who are bound to misuse or exploit. Examples of what could happen if this vulnerability was exploited include bringing down the server or causing hindrance to the regular service.
Critical	Presence of this vulnerability might give intruders the ability to gain highly sensitive content or affect other users of the web application.
Urgent	Presence of this vulnerability might enable intruders to compromise the web application's data store, obtain information from other users' accounts, or obtain command execution on a host in the web application's architecture. For example in this scenario, the web application users can potentially be targeted if the application is exploited.

Sensitive Content

Sensitive content may be detected based on known patterns (credit card numbers, social security numbers) or custom patterns (strings, regular expressions), depending on the option profile used. Intruders may gain access to sensitive content that could result in misuse or other exploits.

Minimal	Sensitive content was found in the web server response. During our scan of the site form(s) were found with field(s) for credit card number or social security number. This information disclosure could result in a confidentiality breach and could be a target for intruders. For this reason we recommend caution.
Medium	Sensitive content was found in the web server response. Specifically our service found a certain sensitive content pattern (defined in the option profile). This information disclosure could result in a confidentiality breach and could be a target for intruders. For this reason we recommend caution.
Serious	Sensitive content was found in the web server response - a valid social security number or credit card information. This infomation disclosure could result in a confidentiality breach, and it gives intruders access to valid sensitive content that could be misused.

Information Gathered

Information Gathered issues (QIDs) include visible information about the web application's platform, code, or architecture. It may also include information about users of the web application.

Minimal	Intruders may be able to retrieve sensitive information related to the web application platform.

WAS Scan Report

	Medium	**Intruders may be able to retrieve sensitive information related to internal functionality or business logic of the web application.**
	Serious	**Intruders may be able to detect highly sensitive data, such as personally identifiable information (PII) about other users of the web application.**