ZAP Scanning Report

Site: http://127.0.0.1:8000

Generated on Wed, 30 Aug 2023 23:02:32

ZAP Version: 2.13.0

Summary of Alerts

Risk Level	Number of Alerts
High	0
Medium	2
Low	4
Informational	4

Alerts

Name	Risk Level	Number of Instances
Absence of Anti-CSRF Tokens	Medium	3
Content Security Policy (CSP) Header Not Set	Medium	13
Application Error Disclosure	Low	3
Cookie No HttpOnly Flag	Low	7
Server Leaks Version Information via "Server" HTTP Response Header Field	Low	19
X-Content-Type-Options Header Missing	Low	1
Authentication Request Identified	Informational	5
Modern Web Application	Informational	11
Session Management Response Identified	Informational	10
User Agent Fuzzer	Informational	36

Alert Detail

Medium	Absence of Anti-CSRF Tokens
Description	No Anti-CSRF tokens were found in a HTML submission form. A cross-site request forgery is an attack that involves forcing a victim to send an HTTP request to a target destination without their knowledge or intent in order to perform an action as the victim. The underlying cause is application functionality using predictable URL /form actions in a repeatable way. The nature of the attack is that CSRF exploits the trust that a web site has for a user. By contrast, cross-site scripting (XSS) exploits the trust that a user has for a web site. Like XSS, CSRF attacks are not necessarily cross-site, but they can be. Cross-site request forgery is also known as CSRF, XSRF, one-click attack, session riding, confused deputy, and sea surf. CSRF attacks are effective in a number of situations, including: * The victim has an active session on the target site.

	* The victim is authenticated via HTTP auth on the target site. * The victim is on the same local network as the target site. CSRF has primarily been used to perform an action against a target site using the victim's privileges, but recent techniques have been discovered to disclose information by gaining access to the response. The risk of information disclosure is dramatically increased when the target site is vulnerable to XSS, because XSS can be used as a platform for CSRF, allowing the attack to operate within the bounds of the same-origin policy.
URL	http://127.0.0.1:8000/admin_login/
Method	POST
Attack	
Evidence	<form action="http://dpaste.com/" name="pasteform" id="pasteform" method="post">
URL	http://127.0.0.1:8000/company_signup/
Method	POST
Attack	
Evidence	<form action="http://dpaste.com/" name="pasteform" id="pasteform" method="post">
URL	http://127.0.0.1:8000/signup/
Method	POST
Attack	
Evidence	<form action="http://dpaste.com/" name="pasteform" id="pasteform" method="post">
Instances	3
Solution	Phase: Architecture and Design Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness easier to avoid. For example, use anti-CSRF packages such as the OWASP CSRFGuard. Phase: Implementation Ensure that your application is free of cross-site scripting issues, because most CSRF defenses can be bypassed using attacker-controlled script. Phase: Architecture and Design Generate a unique nonce for each form, place the nonce into the form, and verify the nonce upon receipt of the form. Be sure that the nonce is not predictable (CWE-330). Note that this can be bypassed using XSS. Identify especially dangerous operations. When the user performs a dangerous operation, send a separate confirmation request to ensure that the user intended to perform that operation. Note that this can be bypassed using XSS. Use the ESAPI Session Management control. This control includes a component for CSRF. Do not use the GET method for any request that triggers a state change. Phase: Implementation Check the HTTP Referer header to see if the request originated from an expected page. This could break legitimate functionality, because users or proxies may have disabled sending the Referer for privacy reasons.

Reference	http://projects.webappsec.org/Cross-Site-Request-Forgery http://cwe.mitre.org/data/definitions/352.html
CWE Id	352
WASC Id	9
Plugin Id	10202

Medium	**Content Security Policy (CSP) Header Not Set**
Description	Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks, including Cross Site Scripting (XSS) and data injection attacks. These attacks are used for everything from data theft to site defacement or distribution of malware. CSP provides a set of standard HTTP headers that allow website owners to declare approved sources of content that browsers should be allowed to load on that page — covered types are JavaScript, CSS, HTML frames, fonts, images and embeddable objects such as Java applets, ActiveX, audio and video files.
URL	http://127.0.0.1:8000/
Method	GET
Attack	
Evidence	
URL	http://127.0.0.1:8000/admin_login/
Method	GET
Attack	
Evidence	
URL	http://127.0.0.1:8000/company_login/
Method	GET
Attack	
Evidence	
URL	http://127.0.0.1:8000/company_signup/
Method	GET
Attack	
Evidence	
URL	http://127.0.0.1:8000/robots.txt
Method	GET
Attack	
Evidence	
URL	http://127.0.0.1:8000/signup/
Method	GET
Attack	
Evidence	
URL	http://127.0.0.1:8000/sitemap.xml
Method	GET
Attack	
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	

Evidence	
URL	http://127.0.0.1:8000/admin_login/
Method	POST
Attack	
Evidence	
URL	http://127.0.0.1:8000/company_login/
Method	POST
Attack	
Evidence	
URL	http://127.0.0.1:8000/company_signup/
Method	POST
Attack	
Evidence	
URL	http://127.0.0.1:8000/signup/
Method	POST
Attack	
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	
Evidence	
Instances	13
Solution	Ensure that your web server, application server, load balancer, etc. is configured to set the Content-Security-Policy header.
Reference	https://developer.mozilla.org/en-US/docs/Web/Security/CSP /Introducing_Content_Security_Policy https://cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html http://www.w3.org/TR/CSP/ http://w3c.github.io/webappsec/specs/content-security-policy/csp-specification.dev.html http://www.html5rocks.com/en/tutorials/security/content-security-policy/ http://caniuse.com/#feat=contentsecuritypolicy http://content-security-policy.com/
CWE Id	693
WASC Id	15
Plugin Id	10038

Low	**Application Error Disclosure**
Description	This page contains an error/warning message that may disclose sensitive information like the location of the file that produced the unhandled exception. This information can be used to launch further attacks against the web application. The alert could be a false positive if the error message is found inside a documentation page.
URL	http://127.0.0.1:8000/admin_login/
Method	POST
Attack	
Evidence	HTTP/1.1 500 Internal Server Error

URL	http://127.0.0.1:8000/company_signup/
Method	POST
Attack	
Evidence	HTTP/1.1 500 Internal Server Error
URL	http://127.0.0.1:8000/signup/
Method	POST
Attack	
Evidence	HTTP/1.1 500 Internal Server Error
Instances	3
Solution	Review the source code of this page. Implement custom error pages. Consider implementing a mechanism to provide a unique error reference/identifier to the client (browser) while logging the details on the server side and not exposing them to the user.
Reference	
CWE Id	200
WASC Id	13
Plugin Id	90022

Low	**Cookie No HttpOnly Flag**
Description	A cookie has been set without the HttpOnly flag, which means that the cookie can be accessed by JavaScript. If a malicious script can be run on this page then the cookie will be accessible and can be transmitted to another site. If this is a session cookie then session hijacking may be possible.
URL	http://127.0.0.1:8000/admin_login/
Method	GET
Attack	
Evidence	Set-Cookie: csrftoken
URL	http://127.0.0.1:8000/company_login/
Method	GET
Attack	
Evidence	Set-Cookie: csrftoken
URL	http://127.0.0.1:8000/company_signup/
Method	GET
Attack	
Evidence	Set-Cookie: csrftoken
URL	http://127.0.0.1:8000/signup/
Method	GET
Attack	
Evidence	Set-Cookie: csrftoken
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	
Evidence	Set-Cookie: csrftoken
URL	http://127.0.0.1:8000/company_login/
Method	POST

Attack	
Evidence	Set-Cookie: csrftoken
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	
Evidence	Set-Cookie: csrftoken
Instances	7
Solution	Ensure that the HttpOnly flag is set for all cookies.
Reference	https://owasp.org/www-community/HttpOnly
CWE Id	1004
WASC Id	13
Plugin Id	10010

Low	**Server Leaks Version Information via "Server" HTTP Response Header Field**
Description	The web/application server is leaking version information via the "Server" HTTP response header. Access to such information may facilitate attackers identifying other vulnerabilities your web/application server is subject to.
URL	http://127.0.0.1:8000/
Method	GET
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/admin_login
Method	GET
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/admin_login/
Method	GET
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/all_companies
Method	GET
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/all_companies/
Method	GET
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/company_login/
Method	GET
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/company_signup/
Method	GET

Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/robots.txt
Method	GET
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/signup
Method	GET
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/signup/
Method	GET
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/sitemap.xml
Method	GET
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/static/Recruitment%20Vector.png
Method	GET
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/user_login
Method	GET
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/admin_login/
Method	POST
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/company_login/
Method	POST
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/company_signup/
Method	POST
Attack	

Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/signup/
Method	POST
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	
Evidence	WSGIServer/0.2 CPython/3.7.11
Instances	19
Solution	Ensure that your web server, application server, load balancer, etc. is configured to suppress the "Server" header or provide generic details.
Reference	http://httpd.apache.org/docs/current/mod/core.html#servertokens http://msdn.microsoft.com/en-us/library/ff648552.aspx#ht_urlscan_007 http://blogs.msdn.com/b/varunm/archive/2013/04/23/remove-unwanted-http-response-headers.aspx http://www.troyhunt.com/2012/02/shhh-dont-let-your-response-headers.html
CWE Id	200
WASC Id	13
Plugin Id	10036

Low	**X-Content-Type-Options Header Missing**
Description	The Anti-MIME-Sniffing header X-Content-Type-Options was not set to 'nosniff'. This allows older versions of Internet Explorer and Chrome to perform MIME-sniffing on the response body, potentially causing the response body to be interpreted and displayed as a content type other than the declared content type. Current (early 2014) and legacy versions of Firefox will use the declared content type (if one is set), rather than performing MIME-sniffing.
URL	http://127.0.0.1:8000/static/Recruitment%20Vector.png
Method	GET
Attack	
Evidence	
Instances	1
Solution	Ensure that the application/web server sets the Content-Type header appropriately, and that it sets the X-Content-Type-Options header to 'nosniff' for all web pages. If possible, ensure that the end user uses a standards-compliant and modern web browser that does not perform MIME-sniffing at all, or that can be directed by the web application /web server to not perform MIME-sniffing.
Reference	http://msdn.microsoft.com/en-us/library/ie/gg622941%28v=vs.85%29.aspx https://owasp.org/www-community/Security_Headers
CWE Id	693
WASC Id	15
Plugin Id	10021

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

URL	http://127.0.0.1:8000/admin_login/
Method	POST
Attack	
Evidence	password
URL	http://127.0.0.1:8000/company_login/
Method	POST
Attack	
Evidence	password
URL	http://127.0.0.1:8000/company_login/
Method	POST
Attack	
Evidence	password
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	
Evidence	password
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	
Evidence	password
Instances	5
Solution	This is an informational alert rather than a vulnerability and so there is nothing to fix.
Reference	https://www.zaproxy.org/docs/desktop/addons/authentication-helper/auth-req-id/
CWE Id	
WASC Id	
Plugin Id	10111

Informational	**Modern Web Application**
Description	The application appears to be a modern web application. If you need to explore it automatically then the Ajax Spider may well be more effective than the standard one.
URL	http://127.0.0.1:8000/
Method	GET
Attack	
Evidence	JobPortal
URL	http://127.0.0.1:8000/admin_login/
Method	GET
Attack	
Evidence	JobPortal
URL	http://127.0.0.1:8000/company_login/
Method	GET
Attack	

Evidence	JobPortal
URL	http://127.0.0.1:8000/company_signup/
Method	GET
Attack	
Evidence	JobPortal
URL	http://127.0.0.1:8000/signup/
Method	GET
Attack	
Evidence	JobPortal
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	
Evidence	JobPortal
URL	http://127.0.0.1:8000/admin_login/
Method	POST
Attack	
Evidence	<a href="#" onclick="return switchPastebinFriendly(this);"> Switch to copy-and-paste view< /a>
URL	http://127.0.0.1:8000/company_login/
Method	POST
Attack	
Evidence	JobPortal
URL	http://127.0.0.1:8000/company_signup/
Method	POST
Attack	
Evidence	<a href="#" onclick="return switchPastebinFriendly(this);"> Switch to copy-and-paste view< /a>
URL	http://127.0.0.1:8000/signup/
Method	POST
Attack	
Evidence	<a href="#" onclick="return switchPastebinFriendly(this);"> Switch to copy-and-paste view< /a>
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	
Evidence	JobPortal
Instances	11
Solution	This is an informational alert and so no changes are required.
Reference	

CWE Id	
WASC Id	
Plugin Id	10109

Informational	**Session Management Response Identified**
Description	The given response has been identified as containing a session management token. The 'Other Info' field contains a set of header tokens that can be used in the Header Based Session Management Method. If the request is in a context which has a Session Management Method set to "Auto-Detect" then this rule will change the session management to use the tokens identified.
URL	http://127.0.0.1:8000/admin_login/
Method	GET
Attack	
Evidence	XizXx8xpVFBaUw2OnngGcT122Eg4JdUrMOvremh31SoafHKDGZoMxcYHuYDX04PK
URL	http://127.0.0.1:8000/company_login/
Method	GET
Attack	
Evidence	XizXx8xpVFBaUw2OnngGcT122Eg4JdUrMOvremh31SoafHKDGZoMxcYHuYDX04PK
URL	http://127.0.0.1:8000/company_signup/
Method	GET
Attack	
Evidence	XizXx8xpVFBaUw2OnngGcT122Eg4JdUrMOvremh31SoafHKDGZoMxcYHuYDX04PK
URL	http://127.0.0.1:8000/signup/
Method	GET
Attack	
Evidence	XizXx8xpVFBaUw2OnngGcT122Eg4JdUrMOvremh31SoafHKDGZoMxcYHuYDX04PK
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	
Evidence	R3P2VRSM1xapv2ujI3HBOks3KSEJWLcxNWmGhTeQzT5DQ2cbggKPom0EQ7Us4iwp
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	
Evidence	XizXx8xpVFBaUw2OnngGcT122Eg4JdUrMOvremh31SoafHKDGZoMxcYHuYDX04PK
URL	http://127.0.0.1:8000/company_login/
Method	POST
Attack	
Evidence	XizXx8xpVFBaUw2OnngGcT122Eg4JdUrMOvremh31SoafHKDGZoMxcYHuYDX04PK
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	
Evidence	XizXx8xpVFBaUw2OnngGcT122Eg4JdUrMOvremh31SoafHKDGZoMxcYHuYDX04PK
URL	http://127.0.0.1:8000/company_login/

Method	POST
Attack	
Evidence	XizXx8xpVFBaUw2OnngGcT122Eg4JdUrMOvremh31SoafHKDGZoMxcYHuYDX04PK
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	
Evidence	XizXx8xpVFBaUw2OnngGcT122Eg4JdUrMOvremh31SoafHKDGZoMxcYHuYDX04PK
Instances	10
Solution	This is an informational alert rather than a vulnerability and so there is nothing to fix.
Reference	https://www.zaproxy.org/docs/desktop/addons/authentication-helper/session-mgmt-id
CWE Id	
WASC Id	
Plugin Id	10112

Informational	**User Agent Fuzzer**
Description	Check for differences in response based on fuzzed User Agent (eg. mobile sites, access as a Search Engine Crawler). Compares the response statuscode and the hashcode of the response body with the original response.
URL	http://127.0.0.1:8000/user_login
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Evidence	
URL	http://127.0.0.1:8000/user_login
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0)
Evidence	
URL	http://127.0.0.1:8000/user_login
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1)
Evidence	
URL	http://127.0.0.1:8000/user_login
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Trident/7.0; rv:11.0) like Gecko
Evidence	
URL	http://127.0.0.1:8000/user_login
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3739.0 Safari/537.36 Edg/75.0.109.0
Evidence	
URL	http://127.0.0.1:8000/user_login
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36
Evidence	

URL	http://127.0.0.1:8000/user_login
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:93.0) Gecko/20100101 Firefox/91.0
Evidence	
URL	http://127.0.0.1:8000/user_login
Method	GET
Attack	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
Evidence	
URL	http://127.0.0.1:8000/user_login
Method	GET
Attack	Mozilla/5.0 (compatible; Yahoo! Slurp; http://help.yahoo.com/help/us/ysearch/slurp)
Evidence	
URL	http://127.0.0.1:8000/user_login
Method	GET
Attack	Mozilla/5.0 (iPhone; CPU iPhone OS 8_0_2 like Mac OS X) AppleWebKit/600.1.4 (KHTML, like Gecko) Version/8.0 Mobile/12A366 Safari/600.1.4
Evidence	
URL	http://127.0.0.1:8000/user_login
Method	GET
Attack	Mozilla/5.0 (iPhone; U; CPU iPhone OS 3_0 like Mac OS X; en-us) AppleWebKit/528.18 (KHTML, like Gecko) Version/4.0 Mobile/7A341 Safari/528.16
Evidence	
URL	http://127.0.0.1:8000/user_login
Method	GET
Attack	msnbot/1.1 (+http://search.msn.com/msnbot.htm)
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0)
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1)
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Trident/7.0; rv:11.0) like Gecko
Evidence	

URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3739.0 Safari/537.36 Edg/75.0.109.0
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:93.0) Gecko/20100101 Firefox/91.0
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	Mozilla/5.0 (compatible; Yahoo! Slurp; http://help.yahoo.com/help/us/ysearch/slurp)
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	Mozilla/5.0 (iPhone; CPU iPhone OS 8_0_2 like Mac OS X) AppleWebKit/600.1.4 (KHTML, like Gecko) Version/8.0 Mobile/12A366 Safari/600.1.4
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	Mozilla/5.0 (iPhone; U; CPU iPhone OS 3_0 like Mac OS X; en-us) AppleWebKit/528.18 (KHTML, like Gecko) Version/4.0 Mobile/7A341 Safari/528.16
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	GET
Attack	msnbot/1.1 (+http://search.msn.com/msnbot.htm)
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0)

Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1)
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	Mozilla/5.0 (Windows NT 10.0; Trident/7.0; rv:11.0) like Gecko
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3739.0 Safari/537.36 Edg/75.0.109.0
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:93.0) Gecko/20100101 Firefox/91.0
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	Mozilla/5.0 (compatible; Yahoo! Slurp; http://help.yahoo.com/help/us/ysearch/slurp)
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	Mozilla/5.0 (iPhone; CPU iPhone OS 8_0_2 like Mac OS X) AppleWebKit/600.1.4 (KHTML, like Gecko) Version/8.0 Mobile/12A366 Safari/600.1.4
Evidence	
URL	http://127.0.0.1:8000/user_login/
Method	POST
Attack	Mozilla/5.0 (iPhone; U; CPU iPhone OS 3_0 like Mac OS X; en-us) AppleWebKit/528.18 (KHTML, like Gecko) Version/4.0 Mobile/7A341 Safari/528.16
Evidence	
URL	http://127.0.0.1:8000/user_login/

Method	POST
Attack	msnbot/1.1 (+http://search.msn.com/msnbot.htm)
Evidence	
Instances	36
Solution	
Reference	https://owasp.org/wstg
CWE Id	
WASC Id	
Plugin Id	10104