Website Vulnerability Scanner Report

Perform in-depth website scanning and discover high risk vulnerabilities.

Testing areas	Light scan	Full scan
Website fingerprinting	✔	✔
Version-based vulnerability detection	✔	✔
Common configuration issues	✔	✔
SQL injection	✖	✔
Cross-Site Scripting	✖	✔
Remote command execution	✖	✔
Discover sensitive files	✖	✔

Get a PRO Account to unlock the full capabilities of this scanner!

✔ http://rafc-project2-dev.ap-south-1.elasticbeanstalk.com/

Summary

Overall risk level:

High

Risk ratings:

Scan information:

Start time:	2019-04-22 08:55:59
Finish time:	2019-04-22 08:56:10
Scan duration:	11 sec
Tests performed:	10/10
Scan status:	Finished

Findings

Vulnerabilities found for server-side software

Risk Level	CVSS	CVE	Summary	Exploit	Affected software
●	7.2	CVE-2019-0211	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.	N/A	http_server 2.4.38
●	6.0	CVE-2019-0217	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.	N/A	http_server 2.4.38
●	6.0	CVE-2019-0215	In Apache HTTP Server 2.4 releases 2.4.37 and 2.4.38, a bug in mod_ssl when using per-location client certificate verification with TLSv1.3 allowed a client to bypass configured access control restrictions.	N/A	http_server 2.4.38

●	5.0	CVE-2019-9636	Python 2.7.x through 2.7.16 and 3.x through 3.7.2 is affected by: Improper Handling of Unicode Encoding (with an incorrect netloc) during NFKC normalization. The impact is: Information disclosure (credentials, cookies, etc. that are cached against a given hostname). The components are: urllib.parse.urlsplit, urllib.parse.urlparse. The attack vector is: A specially crafted URL could be incorrectly parsed to locate cookies or authentication data and send that information to a different host than when parsed correctly.	N/A	Python 3.6.7
●	5.0	CVE-2018-1060	python before versions 2.7.15, 3.4.9, 3.5.6rc1, 3.6.5rc1 and 3.7.0 is vulnerable to catastrophic backtracking in pop3lib's apop() method. An attacker could use this flaw to cause denial of service.	N/A	Python 3.6.7
●	4.3	CVE-2019-9947	An issue was discovered in urllib2 in Python 2.x through 2.7.16 and urllib in Python 3.x through 3.7.2. CRLF injection is possible if the attacker controls a url parameter, as demonstrated by the first argument to urllib.request.urlopen with \r\n (specifically in the query string or PATH_INFO) followed by an HTTP header or a Redis command. This is similar to CVE-2019-9740.	N/A	Python 3.6.7
●	4.3	CVE-2019-9740	An issue was discovered in urllib2 in Python 2.x through 2.7.16 and urllib in Python 3.x through 3.7.2. CRLF injection is possible if the attacker controls a url parameter, as demonstrated by the first argument to urllib.request.urlopen with \r\n followed by an HTTP header or a Redis command.	N/A	Python 3.6.7

Details

Risk description:
These vulnerabilities expose the affected applications to the risk of unauthorized access to confidential data and possibly to denial of service attacks. An attacker could search for an appropriate exploit (or create one himself) for any of these vulnerabilities and use it to attack the system.

Recommendation:
We recommend you to upgrade the affected software to the latest version in order to eliminate the risk of these vulnerabilities.

Passwords are submitted unencrypted over the network

Login form: http://rafc-project2-dev.ap-south-1.elasticbeanstalk.com/

Details

Risk description:
An attacker could intercept the communication between the web browser and the server and he could retrieve the clear-text authentication credentials.

Recommendation:
We recommend you to reconfigure the web server to use HTTPS - which encrypts the communication between the web browser and the server. This way, the attacker will not be able to obtain the clear-text passwords, even though he manages to intercept the network communication.

Insecure HTTP cookies

Cookie Name	Flags missing
csrftoken	Secure, HttpOnly

Details

Risk description:
Since the `Secure` flag is not set on the cookie, the browser will send it over an unencrypted channel (plain HTTP) if such a request is made. Thus, the risk exists that an attacker will intercept the clear-text communication between the browser and the server and he will steal the cookie of the user. If this is a session cookie, the attacker could gain unauthorized access to the victim's web session.

Lack of the `HttpOnly` flag permits the browser to access the cookie from client-side scripts (ex. JavaScript, VBScript, etc). This can be exploited by an attacker in conjuction with a Cross-Site Scripting (XSS) attack in order to steal the affected cookie. If this is a session cookie, the attacker could gain unauthorized access to the victim's web session.

Recommendation:
We recommend reconfiguring the web server in order to set the flag(s) `Secure` , `HttpOnly` to all sensitive cookies.

More information about this issue:
https://blog.dareboost.com/en/2016/12/secure-cookies-secure-httponly-flags/.

Communication is not secure

http://rafc-project2-dev.ap-south-1.elasticbeanstalk.com/

Details

Risk description:
The communication between the web browser and the server is done using the HTTP protocol, which transmits data unencrypted over the network. Thus, an attacker who manages to intercept the communication at the network level, is able to read and modify the data transmitted (including passwords, secret tokens, credit card information and other sensitive data).

Recommendation:
We recommend you to reconfigure the web server to use HTTPS - which encrypts the communication between the web browser and the server.

Server software and technology found

Software / Version	Category
Amazon EC2	Web Servers
Apache 2.4.38	Web Servers
mod_wsgi 3.5	Web Server Extensions
Python 3.6.7	Programming Languages
Django	Web Frameworks
Twitter Bootstrap	Web Frameworks
Font Awesome	Font Scripts
Moment.js	JavaScript Frameworks
Select2	JavaScript Frameworks
jQuery 3.2.1	JavaScript Frameworks

Details

Risk description:
An attacker could use this information to mount specific attacks against the identified software type and version.

Recommendation:
We recommend you to eliminate the information which permit the identification of software platform, technology, server and operating system: HTTP server headers, HTML meta information, etc.

More information about this issue:
https://www.owasp.org/index.php/Fingerprint_Web_Server_(OTG-INFO-002).

Missing HTTP security headers

HTTP Security Header	Header Role	Status
X-XSS-Protection	Mitigates Cross-Site Scripting (XSS) attacks	Not set
X-Content-Type-Options	Prevents possible phishing or XSS attacks	Not set

Details

Risk description:
The X-XSS-Protection HTTP header instructs the browser to stop loading web pages when they detect reflected Cross-Site Scripting (XSS) attacks. Lack of this header exposes application users to XSS attacks in case the web application contains such vulnerability.

The HTTP X-Content-Type-Options header is addressed to Internet Explorer browser and prevents it from reinterpreting the content of a web page (MIME-sniffing) and thus overriding the value of the Content-Type header). Lack of this header could lead to attacks such as Cross-Site Scripting or phishing.

Recommendation:
We recommend setting the `X-XSS-Protection` header to "X-XSS-Protection: 1; mode=block".
More information about this issue:
https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-XSS-Protection

We recommend setting the `X-Content-Type-Options` header to "X-Content-Type-Options: nosniff".
More information about this issue:
https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Content-Type-Options

Password auto-complete is enabled

```
<input class="input100" name="password" type="password"/>
```

Details

Risk description:
When password auto-complete is enabled, the browser will remember the password entered into the login form, such that it will automatically fill it next time the user tries to login.
However, if an attacker gains physical access to the victim's computer, he can retrieve the saved password from the browser's memory and use it to gain access to the victim's account in the application.
Furthermore, if the application is also vulnerable to Cross-Site Scripting, the attacker could steal the saved password remotely.

Recommendation:
We recommend you to disable the password auto-complete feature on the login forms by setting the attribute `autocomplete="off"` on all password fields.

More information about this issue:
https://www.owasp.org/index.php/Testing_for_Vulnerable_Remember_Password_(OTG-AUTHN-005).

Robots.txt file not found

No security issue found regarding client access policies

Directory listing not found (quick scan)

Scan coverage information

List of tests performed (10/10)

- Fingerprinting the server software and technology...
- Checking for vulnerabilities of server-side software...
- Analyzing the security of HTTP cookies...
- Analyzing HTTP security headers...
- Checking for secure communication...
- Checking robots.txt file...
- Checking client access policies...
- Checking for directory listing (quick scan)...
- Checking for password auto-complete (quick scan)...
- Checking for clear-text submission of passwords (quick scan)...

Scan parameters

Website URL:	http://rafc-project2-dev.ap-south-1.elasticbeanstalk.com/
Scan type:	Light
Authentication:	False