

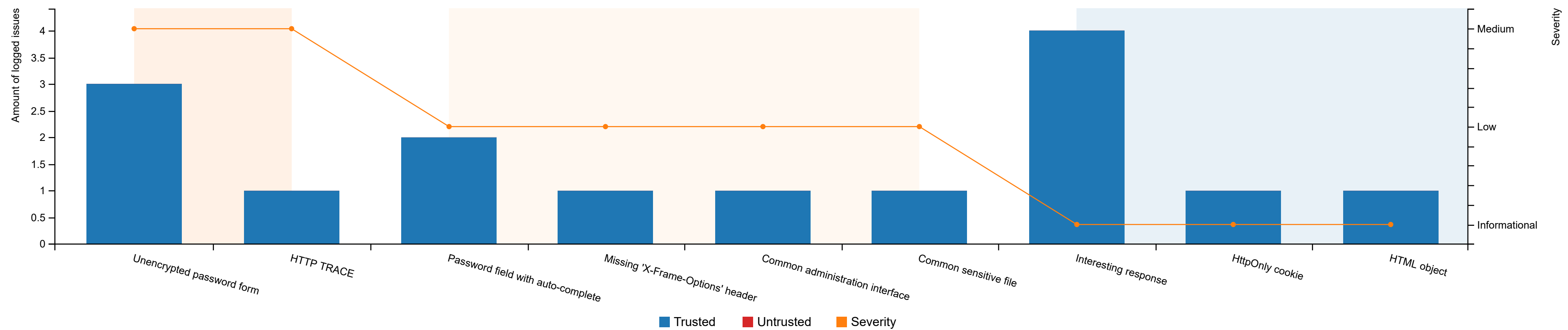
http://www.arafat.com/ Generated on 2020-03-29 23:33:32 -0400

Summary

[Charts](#)[Issues](#) 15[OWASP Top 10](#)

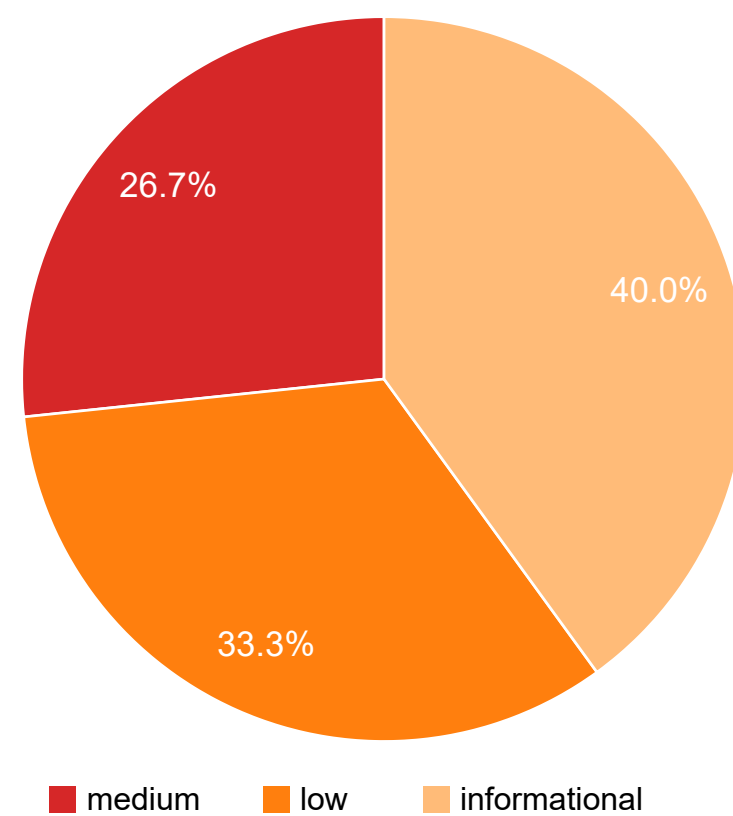
Issues by type, trust, and severity

(Click on the bars or line points for details on the relevant issues.)

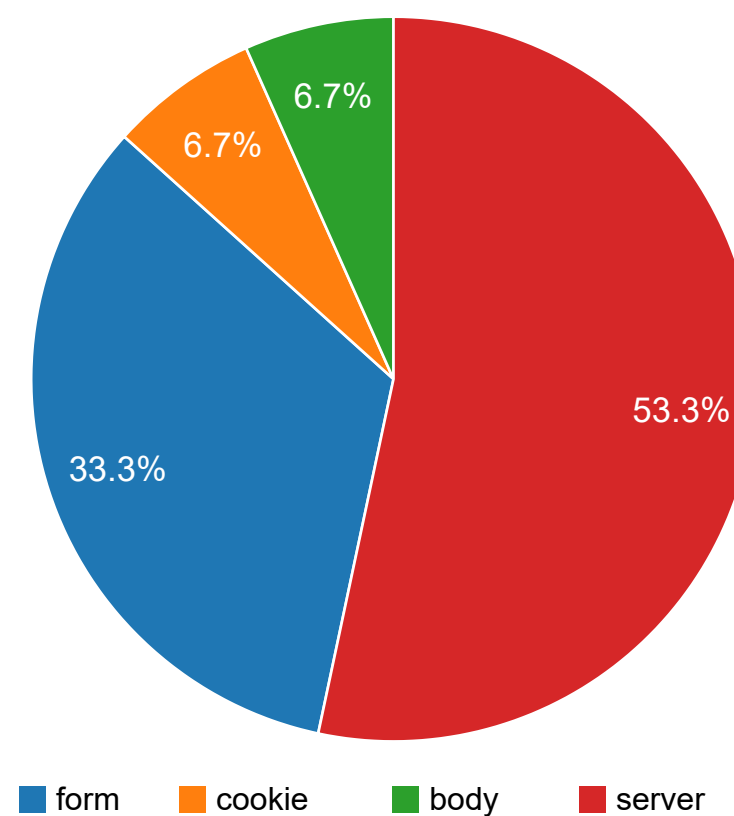


Severities of issues based on possible impact

(Click to see relevant [Trusted](#) issues.)

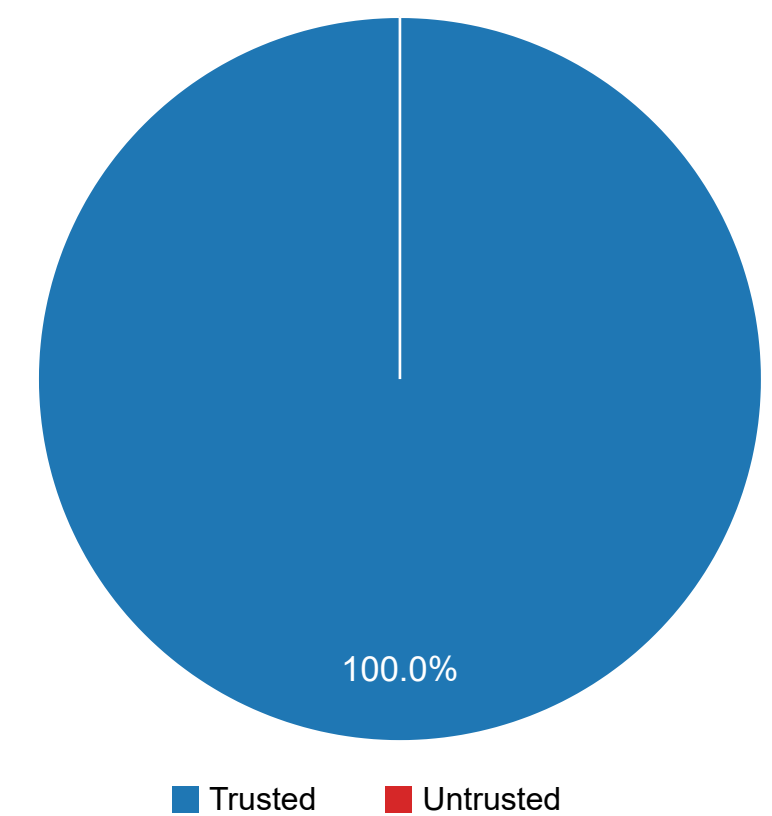


Elements with issues, by type



Trust evaluation ([Trusted](#) vs. [Untrusted](#)) of issues

(Click to see relevant issues.)



http://www.arafat.com/ Generated on 2020-03-29 23:33:32 -0400

Summary

- Charts
- Issues 15
- OWASP Top 10

Trusted 15

Medium severity 4




Unencrypted password form 3

The HTTP protocol by itself is clear text, meaning that any data that is transmitted via HTTP can be captured and the contents viewed.

To keep data private, and prevent it from being intercepted, HTTP is often tunnelled through either Secure Sockets Layer (SSL), or Transport Layer Security (TLS). When either of these encryption standards are used it is referred to as HTTPS.

Cyber-criminals will often attempt to compromise credentials passed from the client to the server using HTTP. This can be conducted via various different Man-in-The-Middle (MiTM) attacks or through network packet captures.

Arachni discovered that the affected page contains a `password` input, however, the value of the field is not sent to the server utilising HTTPS. Therefore it is possible that any submitted credential may become compromised.


	Vector type	HTTP method	Action
	form	GET	http://www.arafat.com/OCMS/index.php
	form	GET	http://www.arafat.com/OCMS/membership_signup.php
	form	GET	http://www.arafat.com/OCMS/membership_signup.php

HTTP TRACE 1

The `TRACE` HTTP method allows a client so send a request to the server, and have the same request then send back in the server's response. This allows the client to determine if the server is receiving the request as expected or if specific parts of the request are not arriving as expected. For example incorrect encoding or a load balancer has filtered or changed a value. On many default installations the `TRACE` method is still enabled.

While not vulnerable by itself, it does provide a method for cyber-criminals to bypass the `HTTPOnly` cookie flag, and therefore could allow a XSS attack to successfully access a session token.

Arachni has discovered that the affected page permits the HTTP `TRACE` method.

	Vector type	HTTP method	Action
	server	TRACE	http://www.arafat.com/

Low severity 5



Password field with auto-complete 2

In typical form-based web applications, it is common practice for developers to allow `autocomplete` within the HTML form to improve the usability of the page. With `autocomplete` enabled (default), the browser is allowed to cache previously entered form values.

For legitimate purposes, this allows the user to quickly re-enter the same data when completing the form multiple times.

When `autocomplete` is enabled on either/both the username and password fields, this could allow a cyber-criminal with access to the victim's computer the ability to have the victim's credentials automatically entered as the cyber-criminal visits the affected page.

Arachni has discovered that the affected page contains a form containing a password field that has not disabled `autocomplete`.


	Vector type	HTTP method	Action
	form	GET	http://www.arafat.com/OCMS/membership_signup.php
	form	GET	http://www.arafat.com/OCMS/index.php

Missing 'X-Frame-Options' header 1

Clickjacking (User Interface redress attack, UI redress attack, UI redressing) is a malicious technique of tricking a Web user into clicking on something different from what the user perceives they are clicking on, thus potentially revealing confidential information or taking control of their computer while clicking on seemingly innocuous web pages.


The server didn't return an `X-Frame-Options` header which means that this website could be at risk of a clickjacking attack.

The `X-Frame-Options` HTTP response header can be used to indicate whether or not a browser should be allowed to render a page inside a frame or iframe. Sites can use this to avoid clickjacking attacks, by ensuring that their content is not embedded into other sites.

	Vector type	HTTP method	Action
	server	GET	http://www.arafat.com/

Common administration interface 1

An administration interface was identified and should be reviewed.

	Vector type	HTTP method	Action
	server	GET	http://www.arafat.com/OCMS/admin/pageHome.php


Common sensitive file 1

Web applications are often made up of multiple files and directories.

It is possible that over time some files may become unreferenced (unused) by the web application and forgotten about by the administrator/developer. Because web applications are built using common frameworks, they contain common files that can be discovered (independent of server).

During the initial recon stages of an attack, cyber-criminals will attempt to locate unreferenced files in the hope that the file will assist in further compromise of the web application. To achieve this they will make thousands of requests using word lists containing common filenames. The response headers from the server will then indicate if the file exists.





Arachni also contains a list of common file names which it will attempt to access.

	Vector type	HTTP method	Action
	server	GET	http://www.arafat.com/OCMS/config.php

Informational severity 6

Interesting response 4

The server responded with a non 200 (OK) nor 404 (Not Found) status code. This is a non-issue, however exotic HTTP response status codes can provide useful insights into the behavior of the web application and assist with the penetration test.

	Vector type	HTTP method	Action
	server	POST	http://www.arafat.com/OCMS/index.php?loginFailed=1
	server	PUT	http://www.arafat.com/Arachni-c328f964b1d7b62b8b81af38bffd4f6d
	server	GET	http://www.arafat.com/%3Cmy_tag_c328f964b1d7b62b8b81af38bffd4f6d/%3E
	server	POST	http://www.arafat.com/OCMS/membership_passwordReset.php

HttpOnly cookie 1


HTTP by itself is a stateless protocol. Therefore the server is unable to determine which requests are performed by which client, and which clients are authenticated or unauthenticated.

The use of HTTP cookies within the headers, allows a web server to identify each individual client and can therefore determine which clients hold valid authentication, from those that do not. These are known as session cookies.

When a cookie is set by the server (sent the header of an HTTP response) there are several flags that can be set to configure the properties of the cookie and how it is to be handled by the browser.


The `HttpOnly` flag assists in the prevention of client side-scripts (such as JavaScript) accessing and using the cookie.

This can help prevent XSS attacks targeting the cookies holding the client's session token (setting the `HttpOnly` flag does not prevent, nor safeguard against XSS vulnerabilities themselves).

	Vector type	HTTP method	Action
	cookie	GET	http://www.arafat.com/OCMS/

HTML object 1

Logs the existence of HTML object tags. Since Arachni can't execute things like Java Applets and Flash this serves as a heads-up to the penetration tester to review the objects in question using a different method.

	Vector type	HTTP method	Action
	body	GET	http://www.arafat.com/OCMS/resources/jquery/js/jquery-1.11.2.min.js

http://www.arafat.com/Generated on 2020-03-29 23:33:32 -0400

Issues

Trusted15

At the time these issues were logged there were no abnormal interferences or anomalous server behavior. These issues are considered trusted and accurate.

Medium4

Low5

Informational6

Unencrypted password form3

unencrypted_password_forms

The HTTP protocol by itself is clear text, meaning that any data that is transmitted via HTTP can be captured and the contents viewed.

To keep data private, and prevent it from being intercepted, HTTP is often tunnelled through either Secure Sockets Layer (SSL), or Transport Layer Security (TLS). When either of these encryption standards are used it is referred to as HTTPS.

Cyber-criminals will often attempt to compromise credentials passed from the client to the server using HTTP. This can be conducted via various different Man-in-The-Middle (MiTM) attacks or through network packet captures.

Arachni discovered that the affected page contains a password input, however, the value of the field is not sent to the server utilising HTTPS. Therefore it is possible that any submitted credential may become compromised.

Remediation guidance

The affected site should be secured utilising the latest and most secure encryption protocols. These include SSL version 3.0 and TLS version 1.2. While TLS 1.2 is the latest and the most preferred protocol, not all browsers will support this encryption method. Therefore, the more common SSL is included. Older protocols such as SSL version 2, and weak ciphers (< 128 bit) should also be disabled.

In form with inputs username password rememberMe signIn using GET at http://www.arafat.com/OCMS/index.php?signIn=1_arachni_trainer_c328f964b1d7b62b8b81af38bfd4f6d pointing to http://www.arafat.com/OCMS/index.php .

Proofi

```
<span class= neip-block >
  Forgot your password?
  <a href="membership_passwordReset.php">
    Click here
  </a>
</span>
</div>
<div class="checkbox">
  <label class="control-label" for="rememberMe">
    <input type="checkbox" name="rememberMe" id="rememberMe" value="1">
  </input>
  Remember me
</label>
</div>
<div class="row">
  <div class="col-sm-offset-3 col-sm-6">
    <button name="signIn" type="submit" id="submit" value="signIn" class="btn btn-primary btn-lg btn-block">
      Sign In
    </button>
  </div>
</div>
</form>
```

Vector information

Affected page: http://www.arafat.com/OCMS/index.php?signIn=1_arachni_trainer_c328f964b1d7b62b8b81af38bfd4f6d

Referring page: http://www.arafat.com/OCMS/index.php?signIn=1_arachni_trainer_c328f964b1d7b62b8b81af38bfd4f6d

In form with inputs newUsername password confirmPassword email custom1 custom2 custom3 custom4 groupID signUp using GET at http://www.arafat.com/OCMS/membership_signup.php pointing to http://www.arafat.com/OCMS/membership_signup.php .

Proofi

```
<form method="post" action="membership_signup.php">
<div class="form-group">
  <label for="username" class="control-label">
    Username
  </label>
  <input class="form-control input-lg" type="text" required="" placeholder="Username" id="username" name="newUsername">
</input>
<span id="usernameAvailable" class="help-block hidden pull-left">
  <i class="glyphicon glyphicon-ok">

  </i>
  Username is available and you can take it.
</span>
<span id="usernameNotAvailable" class="help-block hidden pull-left">
  <i class="glyphicon glyphicon-remove">

  </i>
  Username already exists or is invalid. Make sure you provide a username containing 4 to 20 valid characters.
</span>
<div class="clearfix">

</div>
```

Vector information

Affected page: http://www.arafat.com/OCMS/membership_signup.php

Referring page: http://www.arafat.com/OCMS/membership_signup.php

In form with inputs newUsername password confirmPassword email custom1 custom2 custom3 custom4 groupID signUp using GET at http://www.arafat.com/OCMS/membership_signup.php pointing to http://www.arafat.com/OCMS/membership_signup.php .

Proofi

```
<form method="post" action="membership_signup.php">
<div class="form-group">
  <label for="username" class="control-label">
    Username
  </label>
  <input class="form-control input-lg" type="text" required="" placeholder="Username" id="username" name="newUsername">
</input>
<span id="usernameAvailable" class="help-block hidden pull-left">
  <i class="glyphicon glyphicon-ok">

  </i>
  Username is available and you can take it.
</span>
<span id="usernameNotAvailable" class="help-block hidden pull-left">
  <i class="glyphicon glyphicon-remove">

  </i>
  Username already exists or is invalid. Make sure you provide a username containing 4 to 20 valid characters.
</span>
<div class="clearfix">

</div>
```

Vector information

Affected page: http://www.arafat.com/OCMS/membership_signup.php

Referring page: http://www.arafat.com/OCMS/membership_signup.php

HTTP TRACE1

xst

The TRACE HTTP method allows a client so send a request to the server, and have the same request then send back in the server's response. This allows the client to determine if the server is receiving the request as expected or if specific parts of the request are not arriving as expected. For example incorrect encoding or a load balancer has filtered or changed a value. On many default installations the TRACE method is still enabled.

While not vulnerable by itself, it does provide a method for cyber-criminals to bypass the HTTPOnly cookie flag, and therefore could allow a XSS attack to successfully access a session token.

Arachni has discovered that the affected page permits the HTTP TRACE method.

Remediation guidance

The HTTP TRACE method is normally not required within production sites and should therefore be disabled.

Depending on the function being performed by the web application, the risk level can start low and increase as more functionality is implemented.

The remediation is typically a very simple configuration change and in most cases will not have any negative impact on the server or application.

In server using TRACE at http://www.arafat.com/ pointing to http://www.arafat.com/ .

Proofi

HTTP/1.1 200 OK

Vector information

Affected page: http://www.arafat.com/

Referring page: http://www.arafat.com/

References

CWE-319

OWASP Top 10 2010

References

CWE-693

CAPEC
OWASP

http://www.arafat.com/Generated on 2020-03-29 23:33:32 -0400

Issues

Trusted 15

At the time these issues were logged there were no abnormal interferences or anomalous server behavior. These issues are considered trusted and accurate.

Medium 4

Low 5

Informational 6

Password field with auto-complete 2 password_autocomplete

In typical form-based web applications, it is common practice for developers to allow `autocomplete` within the HTML form to improve the usability of the page. With `autocomplete` enabled (default), the browser is allowed to cache previously entered form values.

For legitimate purposes, this allows the user to quickly re-enter the same data when completing the form multiple times.

When `autocomplete` is enabled on either/both the username and password fields, this could allow a cyber-criminal with access to the victim's computer the ability to have the victim's credentials automatically entered as the cyber-criminal visits the affected page.

Arachni has discovered that the affected page contains a form containing a password field that has not disabled `autocomplete`.

Remediation guidance

The `autocomplete` value can be configured in two different locations.

The first and most secure location is to disable the `autocomplete` attribute on the `<form>` HTML tag. This will disable `autocomplete` for all inputs within that form. An example of disabling `autocomplete` within the form tag is `<form autocomplete=off>`.

The second slightly less desirable option is to disable the `autocomplete` attribute for a specific `<input>` HTML tag. While this may be the less desired solution from a security perspective, it may be preferred method for usability reasons, depending on size of the form. An example of disabling the `autocomplete` attribute within a password input tag is `<input type=password autocomplete=off>`.

In `form` with inputs `newUsername` `password` `confirmPassword` `email` `custom1` `custom2` `custom3` `custom4` `groupID` `signUp` using `GET` at `http://www.arafat.com/OCMS/membership_signup.php` pointing to `http://www.arafat.com/OCMS/membership_signup.php`.

Vector information

Affected page: `http://www.arafat.com/OCMS/membership_signup.php`

Referring page: `http://www.arafat.com/OCMS/membership_signup.php`

In `form` with inputs `username` `password` `rememberMe` `signIn` using `GET` at `http://www.arafat.com/OCMS/index.php?signIn=1_arachni_trainer_c328f964b1d7b62b8b81af38bffd4f6d` pointing to `http://www.arafat.com/OCMS/index.php`.

Vector information

Affected page: `http://www.arafat.com/OCMS/index.php?signIn=1_arachni_trainer_c328f964b1d7b62b8b81af38bffd4f6d`

Referring page: `http://www.arafat.com/OCMS/index.php?signIn=1_arachni_trainer_c328f964b1d7b62b8b81af38bffd4f6d`

Missing 'X-Frame-Options' header 1 x_frame_options

Clickjacking (User Interface redress attack, UI redress attack, UI redressing) is a malicious technique of tricking a Web user into clicking on something different from what the user perceives they are clicking on, thus potentially revealing confidential information or taking control of their computer while clicking on seemingly innocuous web pages.

The server didn't return an `X-Frame-Options` header which means that this website could be at risk of a clickjacking attack.

The `X-Frame-Options` HTTP response header can be used to indicate whether or not a browser should be allowed to render a page inside a frame or iframe. Sites can use this to avoid clickjacking attacks, by ensuring that their content is not embedded into other sites.

Remediation guidance

Configure your web server to include an `X-Frame-Options` header.

In `server` using `GET` at `http://www.arafat.com/` pointing to `http://www.arafat.com/`.

Proof i

HTTP/1.1 302 Found

Vector information

Affected page: `http://www.arafat.com/`

Referring page: `http://www.arafat.com/`

References

CWE-693

MDN
RFC
OWASP

Common administration interface 1 common_admin_interfaces

An administration interface was identified and should be reviewed.

Remediation guidance

Access to administration interfaces should be restricted to trusted IP addresses only.

In `server` using `GET` at `http://www.arafat.com/OCMS/admin/pageHome.php` pointing to `http://www.arafat.com/OCMS/admin/pageHome.php`.

Proof i

HTTP/1.1 301 Moved Permanently

Vector information

Affected page: `http://www.arafat.com/OCMS/admin/pageHome.php`

Referring page: `http://www.arafat.com/OCMS/`

References

Apache.org

Common sensitive file 1 common_files

Web applications are often made up of multiple files and directories.

It is possible that over time some files may become unreferenced (unused) by the web application and forgotten about by the administrator/developer. Because web applications are built using common frameworks, they contain common files that can be discovered (independent of server).

During the initial recon stages of an attack, cyber-criminals will attempt to locate unreferenced files in the hope that the file will assist in further compromise of the web application. To achieve this they will make thousands of requests using word lists containing common filenames. The response headers from the server will then indicate if the file exists.

Arachni also contains a list of common file names which it will attempt to access.

Remediation guidance

If files are unreferenced then they should be removed from the web root and/or the application directory.

Preventing access without authentication may also be an option and can stop a client from being able to view the contents of a file, however it is still likely that the directory structure will be able to be discovered.

Using obscure file names is implementing security through obscurity and is not a recommended option.

In `server` using `GET` at `http://www.arafat.com/OCMS/config.php` pointing to `http://www.arafat.com/OCMS/config.php`.

Proof i

HTTP/1.1 200 OK

Vector information

Affected page: `http://www.arafat.com/OCMS/config.php`

Referring page: `http://www.arafat.com/OCMS/`

http://www.arafat.com/ Generated on 2020-03-29 23:33:32 -0400

Issues

Trusted 15

At the time these issues were logged there were no abnormal interferences or anomalous server behavior. These issues are considered trusted and accurate.

Medium 4Low 5Informational 6

Interesting response 4 interesting_responses

The server responded with a non 200 (OK) nor 404 (Not Found) status code. This is a non-issue, however exotic HTTP response status codes can provide useful insights into the behavior of the web application and assist with the penetration test.

In server using POST at http://www.arafat.com/OCMS/index.php?loginFailed=1 pointing to http://www.arafat.com/OCMS/index.php?loginFailed=1 .

Proof i

HTTP/1.1 302 Found

Vector information

Affected page: http://www.arafat.com/OCMS/index.php?loginFailed=1

Referring page: http://www.arafat.com/

In server using PUT at http://www.arafat.com/Arachni-c328f964b1d7b62b8b81af38bfd4f6d pointing to http://www.arafat.com/Arachni-c328f964b1d7b62b8b81af38bfd4f6d .

Proof i

HTTP/1.1 100 Continue

Vector information

Affected page: http://www.arafat.com/Arachni-c328f964b1d7b62b8b81af38bfd4f6d

Referring page: http://www.arafat.com/

In server using GET at http://www.arafat.com/%3Cmy_tag_c328f964b1d7b62b8b81af38bfd4f6d/%3E pointing to http://www.arafat.com/%3Cmy_tag_c328f964b1d7b62b8b81af38bfd4f6d/%3E .

Proof i

HTTP/1.1 403 Forbidden

Vector information

Affected page: http://www.arafat.com/%3Cmy_tag_c328f964b1d7b62b8b81af38bfd4f6d/%3E

Referring page: http://www.arafat.com/

In server using POST at http://www.arafat.com/OCMS/membership_passwordReset.php pointing to http://www.arafat.com/OCMS/membership_passwordReset.php .

Proof i

HTTP/1.1 302 Found

Vector information

Affected page: http://www.arafat.com/OCMS/membership_passwordReset.php

Referring page: http://www.arafat.com/

HttpOnly cookie 1 http_only_cookies

HTTP by itself is a stateless protocol. Therefore the server is unable to determine which requests are performed by which client, and which clients are authenticated or unauthenticated.

The use of HTTP cookies within the headers, allows a web server to identify each individual client and can therefore determine which clients hold valid authentication, from those that do not. These are known as session cookies.

When a cookie is set by the server (sent the header of an HTTP response) there are several flags that can be set to configure the properties of the cookie and how it is to be handled by the browser.

The HttpOnly flag assists in the prevention of client side-scripts (such as JavaScript) accessing and using the cookie.

This can help prevent XSS attacks targeting the cookies holding the client's session token (setting the HttpOnly flag does not prevent, nor safeguard against XSS vulnerabilities themselves).

Remediation guidance

The initial step to remedy this would be to determine whether any client-side scripts (such as JavaScript) need to access the cookie and if not, set the HttpOnly flag.

Additionally, it should be noted that some older browsers are not compatible with the HttpOnly flag, and therefore setting this flag will not protect those clients against this form of attack.

In cookie with inputs online_clinic_management_system using GET at http://www.arafat.com/OCMS/ pointing to http://www.arafat.com/OCMS/ .

Vector information

Affected page: http://www.arafat.com/OCMS/

Referring page: http://www.arafat.com/OCMS/

HTML object 1 html_objects

Logs the existence of HTML object tags. Since Arachni can't execute things like Java Applets and Flash this serves as a heads-up to the penetration tester to review the objects in question using a different method.

In body using GET at http://www.arafat.com/OCMS/resources/jquery/js/jquery-1.11.2.min.js pointing to http://www.arafat.com/OCMS/resources/jquery/js/jquery-1.11.2.min.js .

Signature iProof i

<object.*?>.*?</object>

<object>" , "</object>

Vector information

Affected page: http://www.arafat.com/OCMS/resources/jquery/js/jquery-1.11.2.min.js

Referring page: http://www.arafat.com/OCMS/resources/jquery/js/jquery-1.11.2.min.js

References

w3.org

References

CWE-200

HttpOnly - OWASP

References

CWE-200

http://www.arafat.com/ Generated on 2020-03-29 23:33:32 -0400

Plugin results

Health map

Generates a simple list of safe/unsafe URLs.

Total	33	http://www.arafat.com/
Without issues	22	http://www.arafat.com/%3Cmy_tag_c328f964b1d7b62b8b81af38bfd4f6d/%3E
With issues	11	http://www.arafat.com/Arachni-c328f964b1d7b62b8b81af38bfd4f6d
Issue percentage	33	http://www.arafat.com/OCMS/ http://www.arafat.com/OCMS/admin/pageHome.php http://www.arafat.com/OCMS/resources/jquery/js/jquery-1.11.2.min.js http://www.arafat.com/OCMS/membership_signup.php http://www.arafat.com/OCMS/config.php http://www.arafat.com/OCMS/membership_passwordReset.php http://www.arafat.com/OCMS/index.php http://www.arafat.com/OCMS/index.php?loginFailed=1 http://www.arafat.com/OCMS/resources/initializr/js/vendor/bootstrap.min.js http://www.arafat.com/OCMS/common.js.php http://www.arafat.com/OCMS/resources/images/appgini-icon.png http://www.arafat.com/OCMS/resources/initializr/css/bootstrap-theme.css http://www.arafat.com/OCMS/resources/initializr/css/bootstrap.css http://www.arafat.com/OCMS/dynamic.css.php http://www.arafat.com/OCMS/checkMemberID.php http://www.arafat.com/OCMS/resources/jquery/js/jquery.background-fit.min.js http://www.arafat.com/OCMS/resources/jquery/js/jquery.mark.min.js http://www.arafat.com/OCMS/resources/jscookie/js.cookie.js http://www.arafat.com/OCMS/resources/lightbox/css/lightbox.css http://www.arafat.com/OCMS/resources/lightbox/js/'%20libraryName%20' http://www.arafat.com/OCMS/resources/lightbox/js/effects.js http://www.arafat.com/OCMS/resources/lightbox/js/lightbox.min.js http://www.arafat.com/OCMS/resources/lightbox/js/prototype.js http://www.arafat.com/OCMS/resources/lightbox/js/scriptaculous.js http://www.arafat.com/OCMS/resources/select2/select2.css http://www.arafat.com/OCMS/resources/select2/select2.min.js http://www.arafat.com/OCMS/resources/timepicker/bootstrap-timepicker.min.css http://www.arafat.com/OCMS/resources/timepicker/bootstrap-timepicker.min.js http://www.arafat.com/_. http://www.arafat.com/div

http://www.arafat.com/ Generated on 2020-03-29 23:33:32 -0400

Sitemap 38

HTTP status code	URL
302	http://www.arafat.com/
200	http://www.arafat.com/OCMS/
200	http://www.arafat.com/OCMS/checkMemberID.php?memberID=
200	http://www.arafat.com/OCMS/checkMemberID.php?memberID=1
200	http://www.arafat.com/OCMS/checkMemberID.php?memberID=a
200	http://www.arafat.com/OCMS/checkMemberID.php?memberID=ar
200	http://www.arafat.com/OCMS/checkMemberID.php?memberID=ara
200	http://www.arafat.com/OCMS/checkMemberID.php?memberID=arac
200	http://www.arafat.com/OCMS/checkMemberID.php?memberID=arachn
200	http://www.arafat.com/OCMS/checkMemberID.php?memberID=arachni_name
200	http://www.arafat.com/OCMS/common.js.php
200	http://www.arafat.com/OCMS/dynamic.css.php
200	http://www.arafat.com/OCMS/index.php
403	http://www.arafat.com/OCMS/index.php?loginFailed=1
200	http://www.arafat.com/OCMS/index.php?signIn=1
200	http://www.arafat.com/OCMS/membership_passwordReset.php
200	http://www.arafat.com/OCMS/membership_passwordReset.php?emptyData=1
200	http://www.arafat.com/OCMS/membership_signup.php
200	http://www.arafat.com/OCMS/resources/images/appgini-icon.png
200	http://www.arafat.com/OCMS/resources/initializr/css/bootstrap-theme.css
200	http://www.arafat.com/OCMS/resources/initializr/css/bootstrap.css
200	http://www.arafat.com/OCMS/resources/initializr/js/vendor/bootstrap.min.js
200	http://www.arafat.com/OCMS/resources/jquery/js/jquery-1.11.2.min.js
404	http://www.arafat.com/OCMS/resources/jquery/js/jquery.background-fit.min.js
200	http://www.arafat.com/OCMS/resources/jquery/js/jquery.mark.min.js
200	http://www.arafat.com/OCMS/resources/jscookie/js.cookie.js
200	http://www.arafat.com/OCMS/resources/lightbox/css/lightbox.css
404	http://www.arafat.com/OCMS/resources/lightbox/js/"%20libraryName%20'
200	http://www.arafat.com/OCMS/resources/lightbox/js/effects.js
200	http://www.arafat.com/OCMS/resources/lightbox/js/lightbox.min.js
200	http://www.arafat.com/OCMS/resources/lightbox/js/prototype.js
200	http://www.arafat.com/OCMS/resources/lightbox/js/scriptaculous.js?load=effects
200	http://www.arafat.com/OCMS/resources/select2/select2.css
200	http://www.arafat.com/OCMS/resources/select2/select2.min.js
200	http://www.arafat.com/OCMS/resources/timepicker/bootstrap-timepicker.min.css
200	http://www.arafat.com/OCMS/resources/timepicker/bootstrap-timepicker.min.js
404	http://www.arafat.com/_.
404	http://www.arafat.com/div