



Burp: Match and replace

[Introduction](#)

[Replacing authorization headers](#)

[Automatically replacing CSRF tokens](#)

[The sky is the limit](#)

Introduction

Burp suite's proxy options have an option called "Match and replace" available. This option has many rich uses that can help us automate our testing process. With some smart uses of this amazing option, we can automatically test for CSRF, IDOR, command injection,.. by just clicking around in the application! Let's explore this magical tool and it's many options.

Replacing authorization headers

Since authorize basically just matches the authorization headers and attempts to replace them with the ones the user supplied, we can set up a similar rule in the proxy.

The screenshot shows the Burp Suite Professional interface. The 'Proxy' tab is selected, and the 'Options' sub-tab is active. The 'Match and Replace' section is highlighted with a red box. It contains a table of rules for automatically replacing parts of requests and responses passing through the proxy.

Enabled	Item	Match	Replace	Type	Comment
<input type="checkbox"/>	Request header	^User-Agent.*\$	User-Agent: Mozilla/5.0 (Linux; ...	Regex	Emulate Android
<input type="checkbox"/>	Request header	^If-Modified-Since.*\$		Regex	Require non-cached response
<input type="checkbox"/>	Request header	^If-None-Match.*\$		Regex	Require non-cached response
<input type="checkbox"/>	Request header	^Referer.*\$		Regex	Hide Referer header
<input type="checkbox"/>	Request header	^Accept-Encoding.*\$		Regex	Require non-compressed respons...
<input type="checkbox"/>	Response header	^Set-Cookie.*\$		Regex	Ignore cookies
<input type="checkbox"/>	Request header	^Host: foo.example.org\$	Host: bar.example.org	Regex	Rewrite Host header
<input type="checkbox"/>	Request header		Origin: foo.example.org	Literal	Add spoofed CORS origin

Below the table, there is a section for 'TLS Pass Through' settings, which are currently empty.

Specify the details of the match/replace rule.

1 Type: Request header

2 Match: Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzd

3 Replace: Authorization: Bearer eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.2342

Comment:

☐ Regex match

OK Cancel

1. Usually the request header will contain the authorization methods
2. Fill in the tokens of the logged in user
3. Fill in the tokens of a second user you want to use

Now, as long as this rule is active you can click around in the application. If you can open any information that should not be public, we have an IDOR on our hands.

To disable this rule, simple uncheck the checkbox in front of it.

Match and Replace

These settings are used to automatically replace parts of requests and responses passing through the Proxy.

	Enabled	Item	Match	Replace	Type	Comment
Add	<input type="checkbox"/>	Request header	^Referer.*\$		Regex	Hide Referer header
Edit	<input type="checkbox"/>	Request header	^Accept-Encoding.*\$		Regex	Require non-compressed respons...
Remove	<input type="checkbox"/>	Response header	^Set-Cookie.*\$		Regex	Ignore cookies
Up	<input type="checkbox"/>	Request header	^Host: foo.example.org\$	Host: bar.example.org	Regex	Rewrite Host header
Down	<input type="checkbox"/>	Request header		Origin: foo.example.org	Literal	Add spoofed CORS origin
	<input type="checkbox"/>	Response header	^Strict-Transport-Sec...		Regex	Remove HSTS headers
	<input type="checkbox"/>	Response header		X-XSS-Protection: 0	Literal	Disable browser XSS protection
	<input checked="" type="checkbox"/>	Request header	Authorization: Bearer ey...	Authorization: Bearer eyJhbGciO...	Literal	

Automatically replacing CSRF tokens

Dashboard
Target
Proxy
Intruder
Repeater
Sequencer
Decoder
Comparer
Extender
Project options
User options
Authorize
intigriti

Intercept
HTTP history
WebSockets history
Options

Up
Down

☒ Automatically update Content-Length header when the response is edited

? Intercept WebSockets Messages

Use these settings to control which WebSockets messages are stalled for viewing and editing in the Intercept tab.

☒ Intercept client-to-server messages
☒ Intercept server-to-client messages

? Response Modification

These settings are used to perform automatic modification of responses.

☐ Unhide hidden form fields
☐ Prominently highlight unhidden fields
☐ Enable disabled form fields
☐ Remove input field length limits
☐ Remove JavaScript form validation
☐ Remove all JavaScript
☐ Remove <object> tags
☐ Convert HTTPS links to HTTP
☐ Remove secure flag from cookies

? Match and Replace

These settings are used to automatically replace parts of requests and responses passing through the Proxy.

Add
Edit
Remove
Up
Down

Enabled	Item	Match	Replace	Type	Comment
<input type="checkbox"/>	Request header	^User-Agent.*\$	User-Agent: Mozilla/4.0 (compa...	Regex	Emulate IE
<input type="checkbox"/>	Request header	^User-Agent.*\$	User-Agent: Mozilla/5.0 (iPhone...	Regex	Emulate iOS
<input type="checkbox"/>	Request header	^User-Agent.*\$	User-Agent: Mozilla/5.0 (Linux; ...	Regex	Emulate Android
<input type="checkbox"/>	Request header	^If-Modified-Since.*\$		Regex	Require non-cached response
<input type="checkbox"/>	Request header	^If-None-Match.*\$		Regex	Require non-cached response
<input type="checkbox"/>	Request header	^Referer.*\$		Regex	Hide Referer header
<input type="checkbox"/>	Request header	^Accept-Encoding.*\$		Regex	Require non-compressed respons...
<input type="checkbox"/>	Response header	^Set-Cookie.*\$		Regex	Ignore cookies

? TLS Pass Through

Depending on if the CSRF token is in the HEADER or the BODY section of the request, we will need to pick one.

Add match/replace rule

? Specify the details of the match/replace rule.

Type: Request header

Match: Request header

Replace: Request body

Comment: Response header

Request param name

Request param value

Request first line

☐ Regex match

OK Cancel

Fill in the regex to indicate to burp how it can find the CSRF token in your request and replace it with a value of your own. Be careful, this is just an example, it may be different for your target.

Add match/replace rule

? Specify the details of the match/replace rule.

Type: Request header

Match: CSRF=*

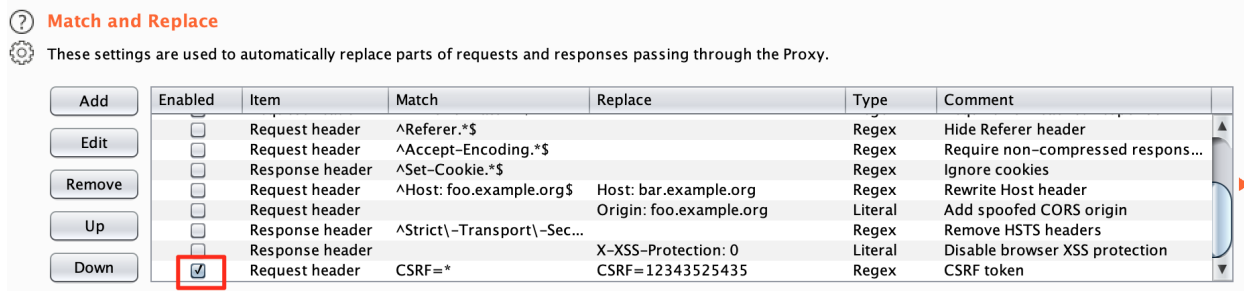
Replace: CSRF=12343525435

Comment: CSRF token|

☒ Regex match

OK Cancel

When this rule is active, click through the application and try to make changes. If you are able to make changes where a CSRF token is normally expected, investigate this further. It may be a vulnerability. To restore normal functionality, simply disable this rule.



The sky is the limit

You can change any value you want in either

- The response
- The request

And in either

- The header
- The body
- Request Parameter names
- Request Parameter values
- Request first line

These give us an infinite amount of possibilities so the sky is the limit... Think outside the box!