

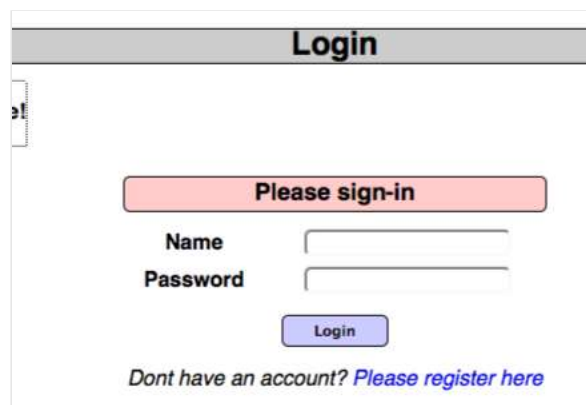
Using Burp to Brute Force a Login Page

Authentication lies at the heart of an application's protection against unauthorized access. If an attacker is able to break an application's authentication function then they may be able to own the entire application.

The following tutorial demonstrates a technique to bypass authentication using a simulated login page from the "Mutillidae" training tool. The version of "Mutillidae" we are using is taken from OWASP's Broken Web Application Project. [Find out how to download, install and use this project.](#)

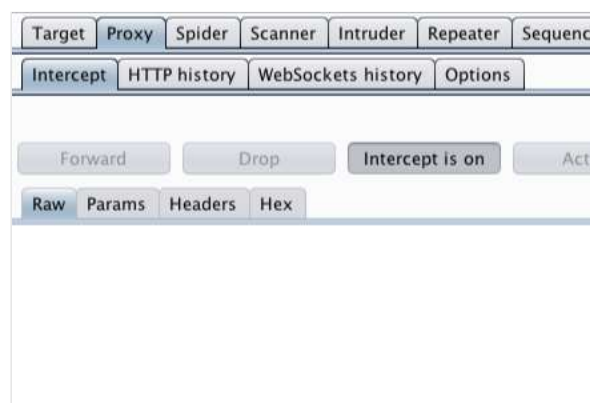
First, ensure that Burp is correctly **configured with your browser**.

In the Burp **Proxy** tab, ensure "Intercept is off" and visit the login page of the application you are testing in your browser.

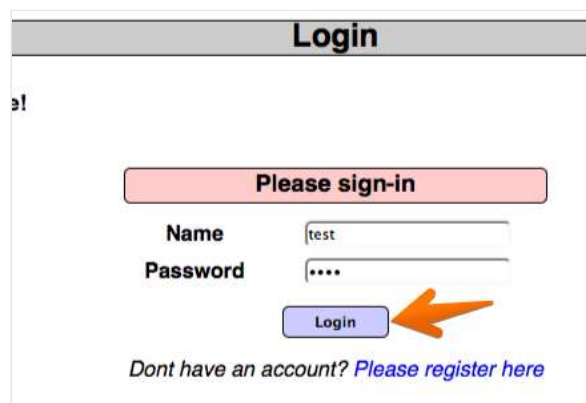


Return to Burp.

In the **Proxy** "Intercept" tab, ensure "Intercept is on".



In your browser enter some arbitrary details in to the login page and submit the request.

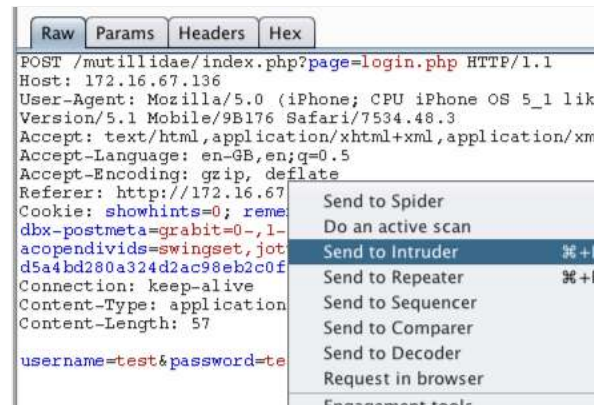


The captured request can be viewed in the **Proxy "Intercept"** tab.

Right click on the request to bring up the context menu.

Then click "Send to **Intruder**".

Note: You can also send requests to the Intruder via the context menu in any location where HTTP requests are shown, such as the site map or Proxy history.



Go to the **Intruder "Positions"** tab.

Clear the pre-set payload positions by using the "Clear" button on the right of the request editor.

Add the "username" and "password" parameter values as positions by highlighting them and using the "Add" button.

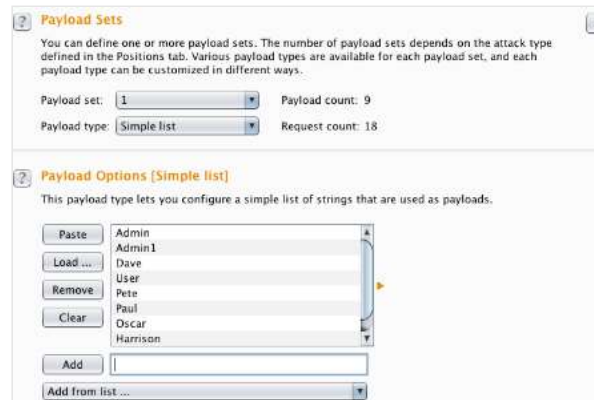
Change the attack to "Cluster bomb" using the "Attack type" drop down menu.



Go to the **"Payloads"** tab.

In the "Payload sets" settings, ensure "Payload set" is "1" and "Payload type" is set to "Simple list".

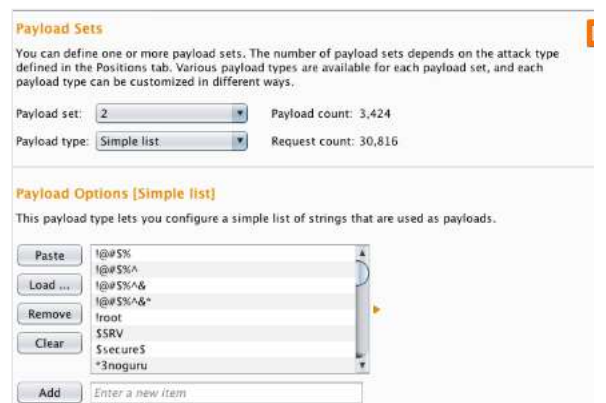
In the **"Payload options"** settings enter some possible usernames. You can do this manually or use a custom or pre-set payload list.



Next, in the "Payload Sets" options, change "Payload" set to "2".

In the "Payload options" settings enter some possible passwords. You can do this manually or using a custom or pre-set list.

Click the "Start attack" button.



In the "Intruder attack" window you can sort the results using the column headers.

In this example sort by "Length" and by "Status".

Results	Target	Positions	Payloads	Options
Filter: Showing all items				
Request	Payload1	Payload2	Status	Error
118	Admin	ADMIN	302	
442	Admin	Admin	302	
9595	Admin	admin	302	
8527	User	USER	302	
8653	User	User	302	
29362	User	user	302	
0			200	
1	Admin	!@#%\$	200	
2	Admin1	!@#%\$	200	
3	Dave	!@#%\$	200	
4	User	!@#%\$	200	
5	Pete	!@#%\$	200	
6	Paul	!@#%\$	200	

The table now provides us with some interesting results for further investigation.

By viewing the response in the attack window we can see that request 118 is logged in as "admin".

Request	Response
	Raw Headers Hex HTML Render
	HTTP/1.1 302 Found Date: Fri, 06 Mar 2015 13:36:36 GMT Server: Apache/2.2.14 (Ubuntu) mod_mono/2.4.3 PHP/5.3.1 proxy_html/3.0.1 mod_python/3.3.1 Python/2.6.5 mod_ssl Phusion_Passenger/3.0.17 mod_perl/2.0.4 Perl/v5.10.1 X-Powered-By: PHP/5.3.2-1ubuntu4.5 Set-Cookie: username=admin Set-Cookie: uid=1 Location: index.php?popupNotificationCode=AU1 Logged-In-User: admin Vary: Accept-Encoding Content-Length: 39071 Connection: close Content-Type: text/html

To confirm that the brute force attack has been successful, use the gathered information (username and password) on the web application's login page.

ASP Mutillidae II: Web Pwn in Mass Production	
curity Level: 0 (Hosed)	Hints: Disabled (0 - I try harder)
Logged in Admin: ad	
e Hints Toggle Security Reset DB View Log View Captured Data Hide Popup Hints Enfo	
Mutillidae: Deliberately Vulnerable Web Pen-Testing Application	
Like Mutillidae? Check out how to help	
What Should I Do?	Video Tutorials
Help Me!	Listing of vulnerabilities

Account Lock Out

In some instances, brute forcing a login page may result in an application locking out the user account. This could be the due to a lock out policy based on a certain number of bad login attempts etc.

Although designed to protect the account, such policies can often give rise to further vulnerabilities. A malicious user may be able to lock out multiple accounts, denying access to a system.

In addition, a locked out account may cause variances in the behavior of the application, this behavior should be explored and potentially exploited.

Please remove the /install folder now
GETBOO
Log In
Too many login tries.
You have tried to log in more than 3 times unsuccessfully for this account i
Please try again in 10 minutes.

Verbose Failure Messages



Where a login requires a username and password, as above, an application might respond to a failed login attempt by indicating whether the reason for the failure was an unrecognized username or incorrect password.

In this instance, you can use an automated attack to iterate through a large list of common usernames to enumerate which ones are valid.

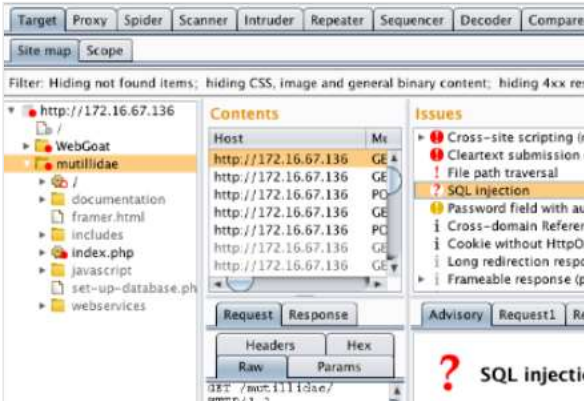
A list of enumerated usernames can be used as the basis for various subsequent attacks, including password guessing, attacks on user data or sessions, or social engineering.



Scanning a login page

In addition to manual testing techniques, Burp **Scanner** can be used to find a variety of authentication and session management vulnerabilities.

In this example, the **Scanner** was able to enumerate a variety of issues that could help an attacker break the authentication and session management of the web application.



Related articles:

- [Getting started with Burp Proxy](#)
- [Using Burp Intruder](#)
- [Using Burp Repeater](#)
- [Getting started with Burp Scanner](#)

Burp Suite

Web vulnerability scanner
Burp Suite Editions
Release Notes

Vulnerabilities

Cross-site scripting (XSS)
SQL injection
Cross-site request forgery
XML external entity injection
Directory traversal
Server-side request forgery

Customers

Organizations
Testers
Developers

Company

About
PortSwigger News
Careers
Contact
Legal
Privacy Notice

Insights

Web Security Academy
Blog
Research
The Daily Swig



Follow us

© 2020 PortSwigger Ltd

