sqlmap Cheat Sheet

Cheat Sheet Series comparitech

Basic options

The sqlmap command will not run without at least one of these options added to it.

-u URL	The target URL Format: -u "http://www.target.com/path/file.htm?variable=1"
-d DIRECT	Connection string for direct database connection Format: -d DBMS://DATABASE_FILEPATH or -d DBMS://USER:PASSWORD@DBMS_IP:DBMS_PORT/DATABASE_NAME
-l LOGFILE	Parse target(s) from Burp or WebScarab proxy log file
-m BULKFILE	Scan multiple targets given in a textual file Format: The file should contain a URL per line
-r REQUESTFILE	Load HTTP request from a file Format: The file can contain an HTTP request or an HTTPS transaction
-g GOOGLEDORK	Process Google dork results as target URLs
-c CONFIGFILE	Load options from a configuration INI file
--wizard	A guided execution service
--update	Update sqlmap to the latest version
--purge	Clear out the sqlmap data folder
--purge-output	As above
--dependencies	Check for missing sqlmap dependencies
-h	Basic help
-hh	Advanced help
--version	Show the sqlmap version number
-v VERBOSE	Verbosity level

Verbosity option values

Possible verbosity level values are:

0	Only Python tracebacks, error, and critical messages
1	Feedback of 0 plus information and warning messages
2	Feedback of 1 plus debug messages
3	Feedback of 2 plus the payloads injected
4	Feedback of 3 plus HTTP requests
5	Feedback of 4 plus the HTTP headers of responses
6	Feedback of 5 plus the content of the HTTP responses

Optimization

The following options can be used to improve the performance of sqlmap.

-o	Turn on all optimization switches
--predict-output	Predict common queries output
--keep-alive	Use persistent HTTP(s) connections
--null-connection	Retrieve page length without actual HTTP response body
--threads=THREADS	Max number of concurrent HTTP(s) requests (default 1)

Detection

The following options are used during research in the detection phase.

--level=LEVEL	The level of tests to perform (1-5, default 1)
--risk=RISK	The risk of tests to perform (1-3, default 1)
--string=STRING	A string to match when query is evaluated to True
--not-string=FALSE-STRING	A string to match when query is evaluated to False
--regexp=REGEXP	Regexp to match when query is evaluated to True
--code=CODE	HTTP code to match when query is evaluated to True
--smart	Perform thorough tests only if positive heuristic(s)

Brute force

These options implement checks during the launch of a brute force attack.

--common-tables	Check the existence of common tables
--common-columns	Check the existence of common columns
--common-files	Check the existence of common files

Miscellaneous

These options do not fit into any of the above categories.

-z MNEMONICS	Use short mnemonics (e.g. "flu,bat,ban,tec=EU")
--alert=ALERT	Run host OS command(s) when SQL injection is found
--beep	Beep on the question and/or when SQLi/XSS/FI is found
--disable-coloring	Disable console output coloring
--list-tampers	Display list of available tamper scripts
--offline	Work in offline mode (only use session data)
--results-file=RESULTS-FILE	Location of CSV results file in multiple targets mode
--shell	Prompt for an interactive sqlmap shell
--tmp-dir=TMPDIR	Local directory for storing temporary files
--unstable	Adjust options for unstable connections

Level option values

This option dictates the volume of tests to perform and the extent of the feedback that they will provide. A higher value implements more extensive checks.

1	A limited number of tests/requests; GET AND POST parameters will be tested (default)
2	Test cookies
3	Test cookies plus User-Agent/Referer
4	As above plus null values in parameters and other bugs
5	An extensive list of tests with an input file for payloads and boundaries

Techniques

These options relate to specific attack strategies. They adjust and focus the attack on particular techniques and targets.

--technique=TECHNIQUE	The SQL injection techniques to use (default "BEUSTQ")
--time-sec=TIMESEC	The number of seconds to delay the DBMS response (default 5)
--union-cols=UCOLS	A range of columns to test for UNION query SQL injection
--union-char=UCHAR	A character to use for brute-forcing columns
--union-from=UFROM	The table to use in the FROM part of a UNION query SQL injection
--dns-domain=DNS-DOMAIN	The domain name to use in a DNS exfiltration attack
--second-url=SECOND-URL	Resulting page URL searched for a second-order response
--second-req=SECOND-REQ	Load a second-order HTTP request from the file
-f	Perform an extensive DBMS version fingerprint
--fingerprint	As above

Request

Add these options to a command to specify how to connect to the target URL.

-A AGENT	HTTP User-Agent header value
--user-agent=AGENT	As above
-H HEADER	Extra header (e.g. "X-Forwarded-For: 127.0.0.1")
--headers=HEADERS	As above
--method=METHOD	Specify an HTTP method to use, such as POST or PUT
--data=DATA	Data string to be sent through POST (e.g. "id=1")
--param-del=PARAMETER	A character to be used for splitting parameter values (e.g., &)
--cookie=COOKIE	HTTP Cookie header value (e.g. "PHPSESSID=a8d127e..")
--cookie-del=COOKIE-CHAR	A character to be used for splitting cookie values (e.g. ;)
--live-cookies=LIVE-COOKIES	A file containing live cookies to be used for loading values
--load-cookies=LOAD-COOKIES	As above with cookies in Netscape/wget format
--drop-set-cookie	Ignore the Set-Cookie header in the response
--mobile	Imitate a smartphone through HTTP User-Agent header
--random-agent	Use a randomly selected HTTP User-Agent header value
--host=HOST	An HTTP Host header value
--referer=REFERER	An HTTP Referer header value
--auth-type=AUTH-TYPE	An HTTP authentication type (Basic, Digest, NTLM or PKI)
--auth-cred=AUTH-CRED	HTTP authentication credentials (name:password)
--auth-file=AUTH-FILE	HTTP authentication PEM cert/private key file
--ignore-code=IGNORE-CODE	Ignore (problematic) HTTP error code (e.g. 401)
--ignore-proxy	Ignore system default proxy settings
--ignore-redirects	Ignore redirection attempts
--ignore-timeouts	Ignore connection timeouts
--proxy=PROXY	Use a proxy to connect to the target URL
--proxy-cred=PROXY-LOGIN	Proxy authentication credentials (name: password)
--proxy-file=PROXY-LIST	Load proxy list from a file
--proxy-freq=PROXY-RATE	Number of requests between the change of proxy from a given list
--tor	Use Tor anonymity network
--tor-port=TORPORT	Set the Tor proxy port to be other than the default
--tor-type=TORTYPE	Set the Tor proxy type (HTTP, SOCKS4 or SOCKS5 (default))
--check-tor	Check to see if Tor is used properly
--delay=DELAY	Delay in seconds between each HTTP request
--timeout=TIMEOUT	Seconds to wait before timeout connection (default 30)
--retries=RETRIES	Number of retries upon timeout (default 3)
--randomize=RPARAM	Randomly change the value for a given parameter(s)
--safe-url=SAFEURL	URL address to visit frequently during testing
--safe-post=SAFE-POST	POST data to send to a safe URL
--safe-req=SAFE-REQUEST	Load safe HTTP request from a file
--safe-freq=SAFE-FREQ	The number of regular requests between visits to a safe URL
--skip-urlencode	Skip URL encoding of payload data
--csrf-token=CSRF-TOKEN	Parameter used to hold the anti-CSRF token
--csrf-url=CSRF-URL	URL to visit for extraction of anti-CSRF token
--csrf-method=CSRF-METHOD	HTTP method to use during anti-CSRF token page visit
--csrf-retries=CSRF-RETRIES	Number of retries to get the anti-CSRF token (default 0)
--force-ssl	Force usage of SSL/HTTPS
--chunked	Use HTTP chunked transfer encoded (POST) requests
--hpp	Use HTTP parameter pollution method
--eval=EVALCODE	Evaluate the provided Python code before the request (e.g. "import hashlib;id2=hashlib.md5(id).hexdigest()")

Injection

The following options can be used to specify which parameters to test for, provide custom injection payloads and optional tampering scripts.

-p TESTPARAMETER	Testable parameter(s)
--skip=SKIP	Skip testing for given parameter(s)
--skip-static	Skip testing parameters that do not appear to be dynamic
--param-exclude=PARAM-EXCLUDE	Regexp to exclude parameters from testing (e.g. "ses")
--param-filter=PARAM-FILTER	Select testable parameter(s) by place (e.g. "POST")
--dbms=DBMS	Force back-end DBMS to provided value
--dbms-cred=DBMS-CREDENTIALS	DBMS authentication credentials (user:password)
--os=OS	Force back-end DBMS operating system to the provided value
--invalid-bignum	Use big numbers for invalidating values
--invalid-logical	Use logical operations for invalidating values
--invalid-string	Use random strings for invalidating values
--no-cast	Turn off payload casting mechanism
--no-escape	Turn off string escaping mechanism
--prefix=PREFIX	Injection payload prefix string
--suffix=SUFFIX	Injection payload suffix string
--tamper=TAMPER	Use given script(s) for tampering injection data

Risk option values

The number given as a parameter to the risk option specifies the extent to which the actions of the tests will expose the attacker. Tests performed in the lowest level will be hardly noticeable to the user, while tests in the higher category can result in mass changes to data.

1	Quick, unnoticeable tests (default)
2	Tests that involve lengthy, heavy data processing, such as time-based SQLI
3	Adds OR-based SQLI and possible data manipulation

Operating system access

These options can be used to access the operating system supporting the DBMS.

--os-cmd=OSCMD	Execute an operating system command
--os-shell	Prompt for an interactive operating system shell
--os-pwn	Prompt for an OOB shell, Meterpreter or VNC
--os-smbrelay	One-click prompt for an OOB shell, Meterpreter or VNC
--os-bof	Stored procedure buffer overflow exploitation
--priv-esc	Database process user privilege escalation
--msf-path=MSFPATH	Local path where Metasploit Framework is installed
--tmp-path=TMPPATH	Remote absolute path of temporary files directory

General

These options provide the opportunity to set general operating parameters.

-s SESSIONFILE	Load session from a stored (.sqlite) file
-t TRAFFICFILE	Log all HTTP traffic into a text file
--answers=ANSWERS	Set predefined answers (e.g. "quit=N,follow=N")
--base64=BASE64PARAMS	Parameter(s) containing Base64 encoded data
--base64-safe	Use URL and filename safe Base64 alphabet (RFC 4648)
--batch	Never ask for user input; use the default behavior
--binary-fields=BINARY-FIELDS	The result fields in binary format (e.g., "digest")
--check-internet	Check the Internet connection before assessing the target
--cleanup	Clean up sqlmap-specific UDF and tables from the database
--crawl=CRAWLDEPTH	Crawl the website starting from the target URL
--crawl-exclude=CRAWL-EXCLUDE	Regexp to exclude pages from crawling (e.g. "logout")
--csv-del=CSVDEL	The delimiter to use in CSV output (default ",")
--charset=CHARSET	Blind SQL injection charset (e.g. "0123456789abcdef")
--dump-format=DUMP-FORMAT	The format of the data dump (CSV (default), HTML or SQLITE)
--encoding=ENCODING	Character encoding to use for data retrieval (e.g., GBK)
--eta	Display the estimated time of arrival for each output
--flush-session	Flush session files for the current target
--forms	Parse and test forms on the target URL
--fresh-queries	Ignore query results stored in the session file
--gpage=GOOGLEPAGE	Use Google dork results starting from the given page number
--har=HARFILE	Log all HTTP traffic into a HAR file
--hex	Use hex conversion during data retrieval
--output-dir=OUTPUT-DIR	The custom output directory path
--parse-errors	Parse and display DBMS error messages from responses
--preprocess=PREPROCESS	Use the named script(s) for preprocessing (request)
--postprocess=POSTPROCESS	Use the named script(s) for postprocessing (response)
--repair	Redump entries having an unknown character marker (?)
--save=SAVECONFIG	Save options to a configuration INI file
--scope=SCOPE	Regexp for filtering targets
--skip-heuristics	Skip heuristic detection of SQLi/XSS vulnerabilities
--skip-waf	Skip heuristic detection of WAF/IPS protection
--table-prefix=TABLE-PREFIX	The prefix to use for temporary tables (default: "sqlmap")
--test-filter=TEST-FILTER	Select tests by payloads and titles (e.g. ROW)
--test-skip=TEST-SKIP	Skip tests by payloads and titles (e.g., BENCHMARK)
--web-root=WEBROOT	The Web server document root directory (e.g. "/var/www")