

Nedefoltny scan

Программа sqlmap позволяет проверять сайты на наличие в них уязвимости SQL-инъекция, уязвимости XSS, а также эксплуатировать SQL-инъекцию. Поддерживаются разнообразные типы SQL-инъекций и разнообразные базы данных.

С помощью sqlmap можно:

проверять, имеется ли в сайтах уязвимость

Если сайт уязвим к SQL-инъекции, то возможно:

получать информацию из базы данных, в том числе дампы (всю) базу данных

изменять и удалять информацию из базы данных

заливать шелл (бэкдор) на веб-сервер

Один из сценариев использования sqlmap:

Получение имени пользователя и пароля из базы данных

Поиск панелей администрирования сайта (админок)

Вход в админку с полученным логином и паролем

При наличии уязвимости атака может развиваться по различным направлениям:

Модификация данных

Заливка бэкдора


Внедрение JavaScript кода для получения данных пользователей

Внедрение кода для подцепления на BeEF

Доступ к машинкам осуществлялся так же как и в 6 лабе (см. ее)

Но к величайшему сожалению, на hackthebox нет доступных не привелегированному пользователю машинок с sql инъекциями на данный момент.

На скриншот 1 не попали последующие попытки, но я пытался использовать многое, будь то усложнение поиска (level=3) или разные id.

```
root@pnknv: ~  
File Actions Edit View Help  
↳ https://www.kali.org/docs/general-use/python3-transition/  
(Run "touch ~/.hushlogin" to hide this message)  
(root@pnknv)~  
# sqlmap -u 10.10.10.220  
 {s.5#stable}  
http://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 @ 23:29:47 /2021-02-01/  
[23:29:47] [INFO] testing connection to the target URL  
[23:29:47] [CRITICAL] unable to connect to the target URL ('Connection refused'). sqlmap is going to retry the request(s)  
[23:29:47] [WARNING] if the problem persists please check that the provided target URL is reachable. In case that it is, you can try to rerun with switch '--random-agent' and/or proxy switches ('--ignore-proxy', '--proxy', ...)  
[23:29:48] [CRITICAL] unable to connect to the target URL ('Connection refused')  
[23:29:48] [INFO] testing if the target URL content is stable  
[23:29:48] [CRITICAL] unable to connect to the target URL ('Connection refused'). sqlmap is going to retry the request(s)  
[23:29:49] [CRITICAL] unable to connect to the target URL ('Connection refused')  
[23:29:49] [CRITICAL] there was an error checking the stability of page because of lack of content. Please check the page request results (and probable errors) by using higher verbosity levels  
[23:29:49] [CRITICAL] no parameter(s) found for testing in the provided data (e.g. GET parameter 'id' in 'www.site.com/index.php?id=1')  
[*] ending @ 23:29:49 /2021-02-01/
```

Как итог:

я протестировал 10 средних машинок с разными параметрами, но результата не получил.
зато немного разобрался с параметрами.

Перешел далее, к xss

Cross Site "Scripter" (также известный как XSSer) – это автоматический фреймворк по обнаружению, эксплуатации и сообщению о XSS уязвимостях в веб-приложениях.

Установил, запустил, но он так же не отработал.

Еще я пробовал фазить директории инструментом dirb. Он был предустановлен.

Инструмент содержит несколько опций для обхода определённых фильтров и различные специальные техники внедрения кода.

dirb -h

Его запустить так же не удалось. Не помогли ни параметры, ни разные машинки, ни параметры...

```
===== NOTES =====  
<url_base> : Base URL to scan. (Use -resume for session resuming)  
<wordlist_file(s)> : List of wordfiles. (wordfile1,wordfile2,wordfile3...)  
===== HOTKEYS =====  
'n' → Go to next directory.  
'q' → Stop scan. (Saving state for resume)  
'r' → Remaining scan stats.  
===== OPTIONS =====  
-a <agent_string> : Specify your custom USER_AGENT.  
-b : Use path as is.  
-c <cookie_string> : Set a cookie for the HTTP request.  
-E <certificate> : path to the client certificate.  
-f : Fine tuning of NOT_FOUND (404) detection.  
-H <header_string> : Add a custom header to the HTTP request.  
-i : Use case-insensitive search.  
-l : Print "Location" header when found.  
-N <nf_code>: Ignore responses with this HTTP code.  
-o <output_file> : Save output to disk.  
-p <proxy[:port]> : Use this proxy. (Default port is 1080)  
-P <proxy_username:proxy_password> : Proxy Authentication.  
-r : Don't search recursively.  
-R : Interactive recursion. (Asks for each directory)  
-S : Silent Mode. Don't show tested words. (For dumb terminals)  
-t : Don't force an ending '/' on URLs.  
-u <username:password> : HTTP Authentication.  
-v : Show also NOT_FOUND pages.  
-w : Don't stop on WARNING messages.  
-X <extensions> / -x <exts_file> : Append each word with this extensions.  
-z <millisecs> : Add a milliseconds delay to not cause excessive Flood.  
===== EXAMPLES =====  
dirb http://url/directory/ (Simple Test)  
dirb http://url/ -X .html (Test files with '.html' extension)  
dirb http://url/ /usr/share/dirb/wordlists/vulns/apache.txt (Test with apache.txt wordlist)  
dirb https://secure_url/ (Simple Test with SSL)
```