

Индивидуальный проект. Этап 4

Использование nikto

Татьяна Александровна Буллер

Содержание

1	Цель работы	4
2	Ход работы	5
3	Выводы	11

Список иллюстраций

2.1	Справка nikto	5
2.2	Анализ основной страницы DVWA с помощью nikto	6
2.3	Директория /config	7
2.4	config.inc.php	7
2.5	Директория /docs	8
2.6	Файл документации	8
2.7	Возвращение исходного кода страницы	9
2.8	Отчет по сканированию страницы уязвимости	10

1 Цель работы

Знакомство со сканером уязвимостей nikto и тестирование его возможностей на примере DVWA.

2 Ход работы

nikto — базовый сканер безопасности веб-сервера. Он сканирует и обнаруживает уязвимости в веб-приложениях, обычно вызванные неправильной конфигурацией на самом сервере, файлами, установленными по умолчанию, и небезопасными файлами, а также устаревшими серверными приложениями.

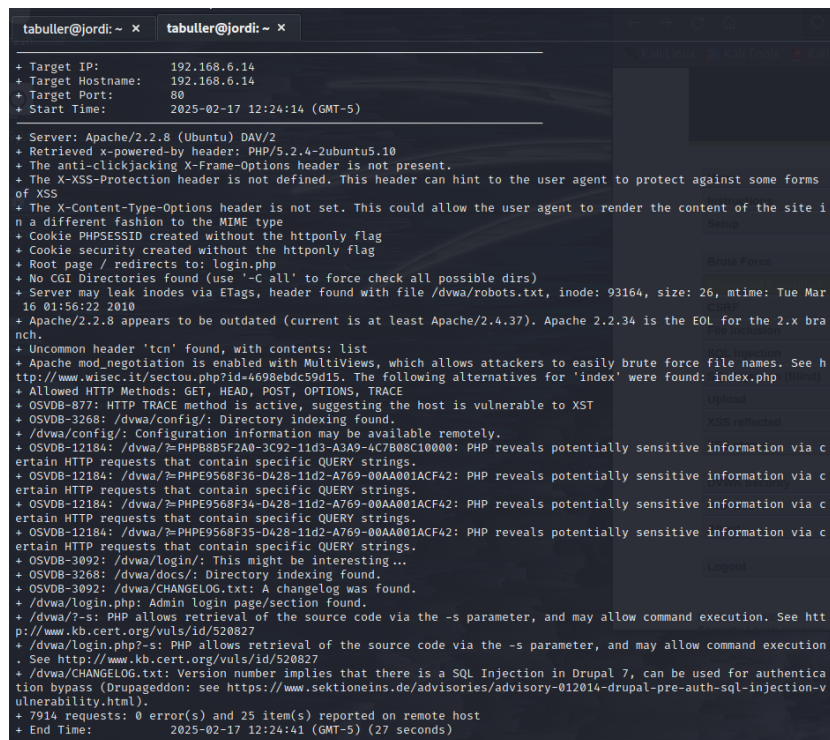
```
(tabuller@jordi)-[~]
$ nikto -help

Options:
  -ask+           Whether to ask about submitting updates
                   yes   Ask about each (default)
                   no    Don't ask, don't send
                   auto  Don't ask, just send
  -cgidir+       Scan these CGI dirs: "none", "all", or values like "/cgi/"
  -config+       Use this config file
  -Display+      Turn on/off display outputs:
                   1     Show redirects
                   2     Show cookies received
                   3     Show all 200/OK responses
                   4     Show URLs which require authentication
                   D     Debug output
                   E     Display all HTTP errors
                   P     Print progress to STDOUT
                   S     Scrub output of IPs and hostnames
                   V     Verbose output
  -dbcheck       Check database and other key files for syntax errors
  -evasion+      Encoding technique:
                   1     Random URI encoding (non-UTF8)
                   2     Directory self-reference (../)
                   3     Premature URL ending
                   4     Prepend long random string
                   5     Fake parameter
                   6     TAB as request spacer
                   7     Change the case of the URL
                   8     Use Windows directory separator (\)
                   A     Use a carriage return (0x0d) as a request spacer
                   B     Use binary value 0x0b as a request spacer
  -Format+       Save file (-o) format:
                   csv   Comma-separated-value
                   json  JSON Format
                   htm   HTML Format
                   nbe   Nessus NBE format
                   sql   Generic SQL (see docs for schema)
                   txt   Plain text
                   xml   XML Format
                   (if not specified the format will be taken from the fi
```

Рис. 2.1: Справка nikto

Основной параметр, который необходимо задать для nikto - -host, который принимает на вход адрес конкретной страницы либо машины. Для начала передадим

адрес основной страницы DVWA и попробуем проанализировать вывод.



```
tabuller@jordi: ~ x tabuller@jordi: ~ x
+ Target IP: 192.168.6.14
+ Target Hostname: 192.168.6.14
+ Target Port: 80
+ Start Time: 2025-02-17 12:24:14 (GMT-5)

+ Server: Apache/2.2.8 (Ubuntu) DAV/2
+ Retrieved x-powered-by header: PHP/5.2.4-2ubuntu5.10
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS.
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ Cookie PHPSESSID created without the httponly flag
+ Cookie security created without the httponly flag
+ Root page / redirects to: login.php
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Server may leak inodes via ETags, header found with file /dwa/robots.txt, inode: 93164, size: 26, mtime: Tue Mar 16 01:56:22 2010
+ Apache/2.2.8 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch.
+ Uncommon header 'tcn' found, with contents: list
+ Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. See http://www.wisec.it/sectou.php?id=4698ebdc59d15. The following alternatives for 'index' were found: index.php
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
+ OSVDB-3268: /dwa/config/: Directory indexing found.
+ /dwa/config/: Configuration information may be available remotely.
+ OSVDB-12184: /dwa/?=PHPB8B5F2A0-3C92-11d3-A3A9-4C7B08C10000: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
+ OSVDB-12184: /dwa/?=PHE9568F36-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
+ OSVDB-12184: /dwa/?=PHE9568F34-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
+ OSVDB-12184: /dwa/?=PHE9568F35-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
+ OSVDB-3092: /dwa/login/: This might be interesting ...
+ OSVDB-3268: /dwa/docs/: Directory indexing found.
+ OSVDB-3092: /dwa/CHANGELOG.txt: A changelog was found.
+ /dwa/login.php: Admin login page/section found.
+ /dwa/?-s: PHP allows retrieval of the source code via the -s parameter, and may allow command execution. See http://www.kb.cert.org/vuls/id/520827
+ /dwa/login.php?-s: PHP allows retrieval of the source code via the -s parameter, and may allow command execution. See http://www.kb.cert.org/vuls/id/520827
+ /dwa/CHANGELOG.txt: Version number implies that there is a SQL Injection in Drupal 7, can be used for authentication bypass (Drupalgeddon: see https://www.sektioneins.de/advisories/advisory-012014-drupal-pre-auth-sql-injection-vulnerability.html).
+ 7914 requests: 0 error(s) and 25 item(s) reported on remote host
+ End Time: 2025-02-17 12:24:41 (GMT-5) (27 seconds)
```

Рис. 2.2: Анализ основной страницы DVWA с помощью nikto

Видим, что nikto сразу определил версию веб-сервера Apache, на котором запущена страница, используемую версию php и отсутствующие заголовки в теле запроса, отсутствие которых позволяет так или иначе манипулировать страницей. Далее по отчету видим список HTTP-методов, которые принимает страница. Сканер обнаружил также файлы конфигурации php, страницу входа, лог изменений. Всегда следует обращать внимание на устаревшие версии тех или иных сервисов, так как нередко случаи того, что в них присутствуют доступные для эксплуатации уязвимости. Так, здесь nikto обращает внимание на устаревшую версию Apache и данные из лога изменений, из которых следует, что страница уязвима к SQLi. Кроме этого, сканер обнаруживает директории, ссылок на которые нет на главной странице DVWA. Это, например, директория /config, содержание которой, судя по названию, может представлять интерес для исследователя.



Рис. 2.3: Директория /config

Эта директория содержит единственный файл и перенаправление в корень - страницу, на которой мы были до этого. Открыв файл, лежащий в этой директории, мы не увидим никакого вывода, пока не добавим в конец адреса знак тильды. Действительно, этот файл хранит некоторые детали конфигурации базы данных:

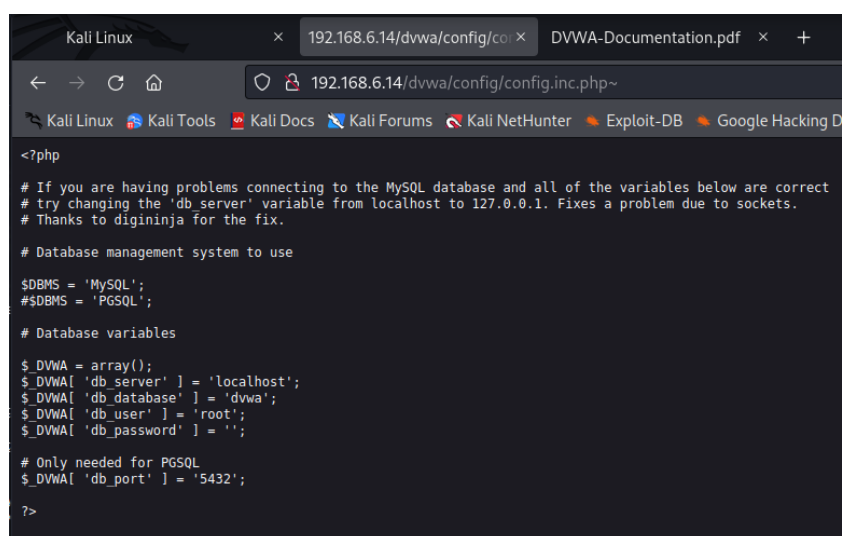


Рис. 2.4: config.inc.php

Кроме файла конфигурации никто нашел еще и директорию /docs, где, судя по названию, хранится некоторая документация. И действительно - внутри находим

.pdf файл с документацией DVWA.



Рис. 2.5: Директория /docs

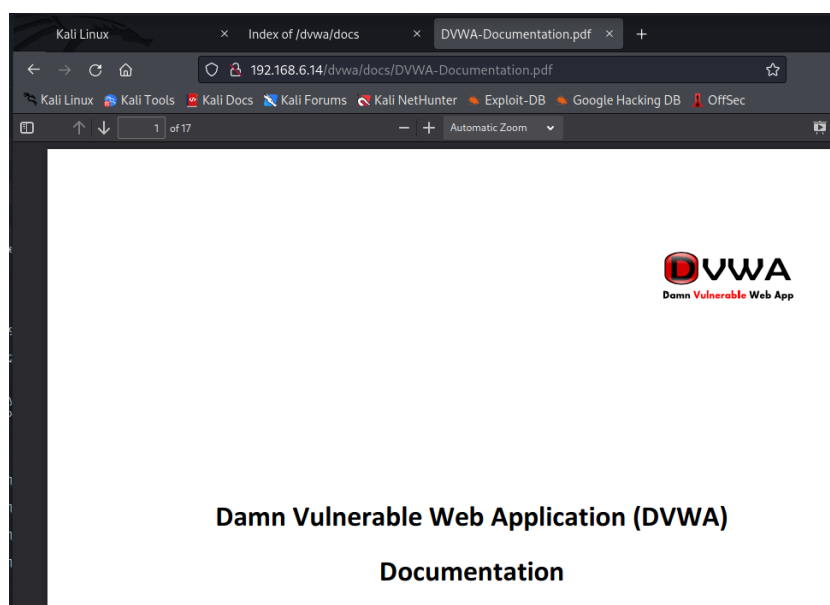
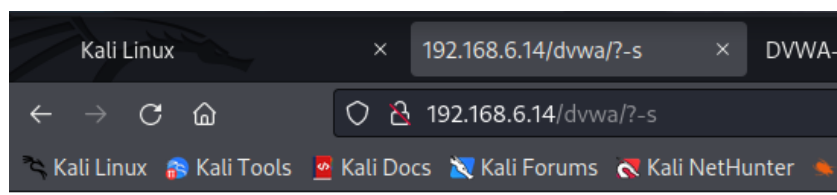


Рис. 2.6: Файл документации

Сканер также обращает внимание исследователя на то, что страница отвечает на запрос с параметром ?-s, возвращая исходный код. Это может быть крайне полезно в дальнейшем исследовании уязвимостей.



```
<?php
define( 'DVWA_WEB_PAGE_TO_ROOT', '' );
require_once DVWA_WEB_PAGE_TO_ROOT.'dvwa/includes/dvwaPage.inc.php';
dvwaPageStartup( array( 'authenticated', 'phpids' ) );
$page = dvwaPageNewGrab();
$page[ 'title' ] .= $page[ 'title_separator' ].'Welcome';
$page[ 'page_id' ] = 'home';
$page[ 'body' ] .= "
<div class=\"body_padded\">
    <h1>Welcome to Damn Vulnerable Web App!</h1>
```

Рис. 2.7: Возвращение исходного кода страницы

Теперь попробуем просканировать страницу одной из уязвимостей. Кроме информации, аналогичной прошлому выводу, nikto обнаружил отдельные страницы с исходным кодом и помощью, которые, по его мнению, не должны быть доступны.

```

(tabuller@jordi)-[~]
$ nikto -host http://192.168.6.14/dvwa/vulnerabilities/exec/
- Nikto v2.1.6

+ Target IP: 192.168.6.14
+ Target Hostname: 192.168.6.14
+ Target Port: 80
+ Start Time: 2025-02-17 12:20:48 (GMT-5)

+ Server: Apache/2.2.8 (Ubuntu) DAV/2
+ Retrieved x-powered-by header: PHP/5.2.4-2ubuntu5.10
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type
+ Cookie PHPSESSID created without the httponly flag
+ Cookie security created without the httponly flag
+ Root page / redirects to: .././login.php
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ Apache/2.2.8 appears to be outdated (current is at least Apache/2.4.37). Apache 2.2.34 is the EOL for the 2.x branch.
+ Uncommon header 'tcn' found, with contents: list
+ Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. See http://www.wisec.it/sectou.php?id=4698ebdc59d15. The following alternatives for 'index' were found: index.php
+ Allowed HTTP Methods: GET, HEAD, POST, OPTIONS, TRACE
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable to XST
+ OSVDB-3268: /dvwa/vulnerabilities/exec/help/: Directory indexing found.
+ /dvwa/vulnerabilities/exec/help/: Help directory should not be accessible
+ OSVDB-12184: /dvwa/vulnerabilities/exec/?=PHPB885F2A0-3C92-11d3-A3A9-4C7B08C10000: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
+ OSVDB-12184: /dvwa/vulnerabilities/exec/?=PHPE9568F36-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
+ OSVDB-12184: /dvwa/vulnerabilities/exec/?=PHPE9568F34-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
+ OSVDB-12184: /dvwa/vulnerabilities/exec/?=PHPE9568F35-D428-11d2-A769-00AA001ACF42: PHP reveals potentially sensitive information via certain HTTP requests that contain specific QUERY strings.
+ OSVDB-3268: /dvwa/vulnerabilities/exec/source/: Directory indexing found.
+ /dvwa/vulnerabilities/exec/?-s: PHP allows retrieval of the source code via the -s parameter, and may allow command execution. See http://www.kb.cert.org/vuls/id/520827
+ 7914 requests: 0 error(s) and 19 item(s) reported on remote host
+ End Time: 2025-02-17 12:21:07 (GMT-5) (19 seconds)

+ 1 host(s) tested

```

Рис. 2.8: Отчет по сканированию страницы уязвимости

3 Выводы

Было освоено применение сканера уязвимостей nikto и протестированы его возможности на примере заведомо уязвимой страницы DVWA.