

Презентация по четвёртому этапу индивидуального проекта

Информационная безопасность

Арбатова В. П.

30 апреля 2025

Российский университет дружбы народов, Москва, Россия

Цель работы

Научиться тестированию веб-приложений с помощью сканера nikto

Задание

Использование nikto.

Теоретическое введение

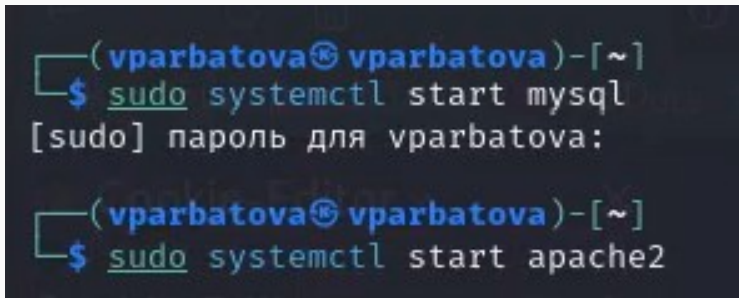
nikto — базовый сканер безопасности веб-сервера. Он сканирует и обнаруживает уязвимости в веб-приложениях, обычно вызванные неправильной конфигурацией на самом сервере, файлами, установленными по умолчанию, и небезопасными файлами, а также устаревшими серверными приложениями. Поскольку nikto построен исключительно на LibWhisker2, он сразу после установки поддерживает кросс-платформенное развертывание, SSL (криптографический протокол, который подразумевает более безопасную связь), методы аутентификации хоста (NTLM/Basic), прокси и несколько методов уклонения от идентификаторов. Он также поддерживает перечисление поддоменов, проверку безопасности приложений (XSS, SQL-инъекции и т. д.) и способен с помощью атаки паролей на основе словаря угадывать учетные данные авторизации.

Для запуска сканера nikto введите в командную строку терминала команду: `# nikto`

По умолчанию, как ранее было показано в других приложениях, при обычном запуске команды отображаются различные доступные параметры. Для сканирования цели введите

Выполнение лабораторной работы

Чтобы работать с nikto, необходимо подготовить веб-приложение, которое будем сканировать. Это будет DVWA. Для этого запустила apache2 (рис. [-@fig:001]).

A terminal window with a dark background and light blue text. The prompt is '(vparbatova@vparbatova)-[~]'. The first command entered is '\$ sudo systemctl start mysql', followed by the prompt '[sudo] пароль для vparbatova:'. The second command entered is '\$ sudo systemctl start apache2'.

```
(vparbatova@vparbatova)-[~]  
$ sudo systemctl start mysql  
[sudo] пароль для vparbatova:  
  
(vparbatova@vparbatova)-[~]  
$ sudo systemctl start apache2
```

Рис. 1: Запускаю все необходимое

Ввожу в адресной строке браузера адрес DVWA, перехожу в режим выбора уровня безопасности, ставлю минимальный (необязательно, nikto при обычном сканировании для режима impossible и low выдаст одинаковые потенциальные уязвимости, что логично, ведь они остаются, но изменяется сложность, с которой их можно использовать)

The screenshot shows the DVWA Security Level configuration page. On the left is a sidebar with navigation links: Home, Instructions, Setup / Reset DB, Brute Force, Command Injection, CSRF, File Inclusion, File Upload, Insecure CAPTCHA, SQL Injection, SQL Injection (Blind), Weak Session IDs, XSS (DOM), and XSS (Reflected). The main content area is titled 'DVWA Security' with a lock icon. Below the title is the 'Security Level' section, which states 'Security level is currently: impossible.' and 'You can set the security level to low, medium, high or impossible. The security level changes the vulnerability level of DVWA:'. A list of four levels is provided: 1. Low - This security level is completely vulnerable and has no security measures at all. It's use is to be as an example of how web application vulnerabilities manifest through bad coding practices and to serve as a platform to teach or learn basic exploitation techniques. 2. Medium - This setting is mainly to give an example to the user of bad security practices, where the developer has tried but failed to secure an application. It also acts as a challenge to users to refine their exploitation techniques. 3. High - This option is an extension to the medium difficulty, with a mixture of harder or alternative bad practices to attempt to secure the code. The vulnerability may not allow the same extent of the exploitation, similar in various Capture The Flags (CTFs) competitions. 4. Impossible - This level should be secure against all vulnerabilities. It is used to compare the vulnerable source code to the secure source code. Prior to DVWA v1.9, this level was known as 'high'. At the bottom, there is a dropdown menu set to 'Low' and a 'Submit' button.

Home
Instructions
Setup / Reset DB
Brute Force
Command Injection
CSRF
File Inclusion
File Upload
Insecure CAPTCHA
SQL Injection
SQL Injection (Blind)
Weak Session IDs
XSS (DOM)
XSS (Reflected)

DVWA Security

Security Level

Security level is currently: **impossible**.

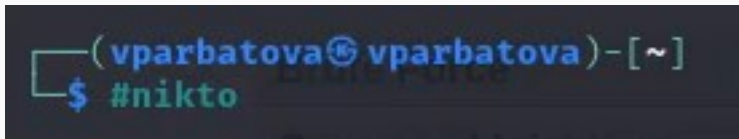
You can set the security level to low, medium, high or impossible. The security level changes the vulnerability level of DVWA:

1. Low - This security level is completely vulnerable and **has no security measures at all**. It's use is to be as an example of how web application vulnerabilities manifest through bad coding practices and to serve as a platform to teach or learn basic exploitation techniques.
2. Medium - This setting is mainly to give an example to the user of **bad security practices**, where the developer has tried but failed to secure an application. It also acts as a challenge to users to refine their exploitation techniques.
3. High - This option is an extension to the medium difficulty, with a mixture of **harder or alternative bad practices** to attempt to secure the code. The vulnerability may not allow the same extent of the exploitation, similar in various Capture The Flags (CTFs) competitions.
4. Impossible - This level should be **secure against all vulnerabilities**. It is used to compare the vulnerable source code to the secure source code. Prior to DVWA v1.9, this level was known as 'high'.

Low

Рис. 2: Установка минимального уровня безопасности

Запуск nikto

A terminal window with a dark background. The prompt is (vparbatova@vparbatova)-[~] in blue. Below it, the command \$ #nikto is entered in green.

```
(vparbatova@vparbatova)-[~]  
$ #nikto
```

Рис. 3: Запуск

Проверить веб-приложение можно, введя его полный URL и не вводя порт, попробовала
просканировать так

```
(vparbatova@vparbatova)-[~]
$ nikto -h http://127.0.0.1/DVWA/
- Nikto v2.5.0

+ Target IP:      127.0.0.1
+ Target Hostname: 127.0.0.1
+ Target Port:    80
+ Start Time:     2025-04-30 11:35:02 (GMT3)

+ Server: Apache/2.4.62 (Debian)
+ /DVWA/: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /DVWA/: 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. See: https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ Root page /DVWA redirects to: login.php
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ OPTIONS: Allowed HTTP Methods: GET, POST, OPTIONS, HEAD .
+ /DVWA///etc/hosts: The server install allows reading of any system file by adding an extra '/' to the URL.
+ /DVWA/config/: Directory indexing found.
+ /DVWA/config/: Configuration information may be available remotely.
+ /DVWA/tests/: Directory indexing found.
+ /DVWA/tests/: This might be interesting.
+ /DVWA/database/: Directory indexing found.
+ /DVWA/database/: Database directory found.
+ /DVWA/docs/: Directory indexing found.
+ /DVWA/login.php: Admin login page/section found.
+ /DVWA/.git/index: Git Index file may contain directory listing information.
+ /DVWA/.git/HEAD: Git HEAD file found. Full repo details may be present.
+ /DVWA/.git/config: Git config file found. Infos about repo details may be present.
+ /DVWA/.gitignore: .gitignore file found. It is possible to grasp the directory structure.
+ /DVWA/wp-content/themes/twentyeleven/images/headers/server.php?filesrc=/etc/hosts: A PHP backdoor file manager was found.
+ /DVWA/wordpress/wp-content/themes/twentyeleven/images/headers/server.php?filesrc=/etc/hosts: A PHP backdoor file manager was found.
+ /DVWA/wp-includes/Requests/Utility/content-post.php?filesrc=/etc/hosts: A PHP backdoor file manager was found.
+ /DVWA/wordpress/wp-includes/Requests/Utility/content-post.php?filesrc=/etc/hosts: A PHP backdoor file manager was found.
+ /DVWA/wp-includes/js/tinymce/themes/modern/Meuhy.php?filesrc=/etc/hosts: A PHP backdoor file manager was found.
+ /DVWA/wordpress/wp-includes/js/tinymce/themes/modern/Meuhy.php?filesrc=/etc/hosts: A PHP backdoor file manager was found.
+ /DVWA/assets/mobirise/css/meta.php?filesrc=: A PHP backdoor file manager was found.
+ /DVWA/login.cgi?cli=aa%20aa%27cat%20/etc/hosts: Some D-Link router remote command execution.
+ /DVWA/shell?cat+etc/hosts: A backdoor was identified.
+ /DVWA/.dockerignore: .dockerignore file found. It may be possible to grasp the directory structure and learn more about the site.
```

DVWA S

Security L

Security level is

You can set the level of DVWA:

1. Low - This is an excellent platform for developers to experiment with security practices.
2. Medium - This is a good platform for developers to experiment with security practices.
3. High - This is a good platform for developers to experiment with security practices.
4. Impossible - This is a good platform for developers to experiment with security practices.

Затем попробовала просканировать введя адрес хоста и адрес порта, результаты незначительно отличаются

```
(vparbatova@vparbatova)-[~]  
$ nikto -h 127.0.0.1 -p 80  
- Nikto v2.5.0  
  
+ Target IP: 127.0.0.1  
+ Target Hostname: 127.0.0.1  
+ Target Port: 80  
+ Start Time: 2025-04-30 11:37:35 (GMT3)  
  
+ Server: Apache/2.4.62 (Debian)
```

Рис. 5: Сканирование вторым способом

Выводы

Научилась использовать сканер nikto для тестирования веб-приложений

Список литературы
