Индивидуальный проект. Этап 4

Использование nikto

Парфенова Елизавета Евгеньевна

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

Использование веб-сканера Nikto для сканирования уязвимостей веб-приложений.

2 Теоретическое введение

Nikto – веб-сканер, проверяющий веб-серверы на самые частые ошибки, возникающие обычно из-за человеческого фактора. Проверяет целевой веб-сервер на наличие опасных файлов и исполняемых сценариев, инструментов администрирования базами данных, устаревшего программного обеспечения. [1]

Он является бесплатным (open source) сканером. Утилита относиться к классу blackbox сканеров, т. е. сканеров, использующих стратегию сканирования методом черного ящика. Это значит, что заранее неизвестно о внутреннем устройстве программы/сайта (доступ к исходному коду отсутствует) и упор сделан на функциональность. Программа может обнаруживать более 6700 потенциально опасных файлов и уязвимостей. Новые уязвимости добавляются в базу данных программы по мере их возникновения. Nikto не позиционируется как стелс сканер (стелс сканеры никогда не устанавливают TCP-соединения до конца, тем самым сканирование происходит скрытно) – при сканировании сайта в логах сайта или в любой другой системе обнаружения вторжений, если она используется, будет отображена информация о том, что сайт подвергается сканированию.

Первая версия Nikto под номером 1.00 была создана в 2001 году Американским инженером по информационной безопасности Крисом Сулло. На момент 2023 года последней актуальной версией является версия 2.1.6.

Среди функций Nikto можно выделить следующие:

- поддержка SSL,
- поддержка HTTP прокси;

- создание отчетов в текстовом формате, XML, HTML, NBE или CSV;
- возможность сканирования портов;
- поиск поддоменов;
- поддержка плагинов для расширения функционала сканирования. [2]

3 Выполнение этапа проекта

Для того, чтобы просканировать веб-приложение DVWA нам необходимо запустить его. Для этого мы запускаем сервер apache2 и базу данных с помощью команд *sudo service apashe2 start* и *service mariadb start* (рис. 3.1).

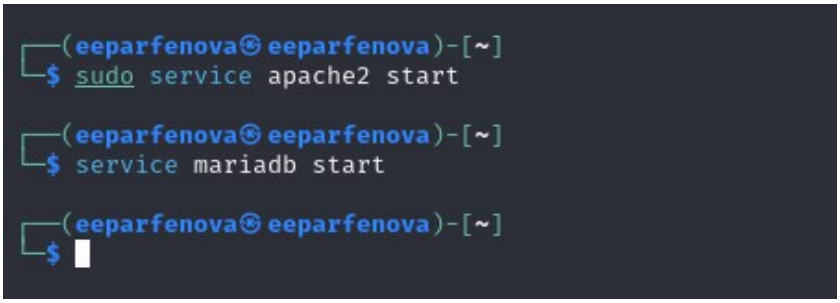

Рис. 3.1: Запуск веб-приложения DVWA

Проверим наличие perl и nikto в нашей системе. Для этого последовательно введем команды *perl -v* и *nikto*. perl, по информации с источника [2], обязательно должен быть установлен перед nikto. В тоге, видим, что и то, и другое установлено в Linux (рис. 3.2).

Рис. 3.2: Проверка наличия nikto

Запрос в Nikto можно сделать через URL и через IP (с портом). В первом случае команда будет выглядеть следующим образом *nikto -h http://localhost/DVWA/ -output report.txt - Format text*. (рис. 3.3) Все что следует после опции output используется дополнительно для отчетов разных форматов. Я попробовала создать отчеты в текстовом формате (рис. 3.4) и в формате html (рис. 3.5). Для отчета в формате html команда немного изменится и будет выглядеть следующим образом (рис. 3.6). (В обоих вариантах report.txt(.html) - название файла, в котором отчет будет сохранен)

Рис. 3.3: Сканирование DVWA

Рис. 3.4: Отчет в текстовом формате

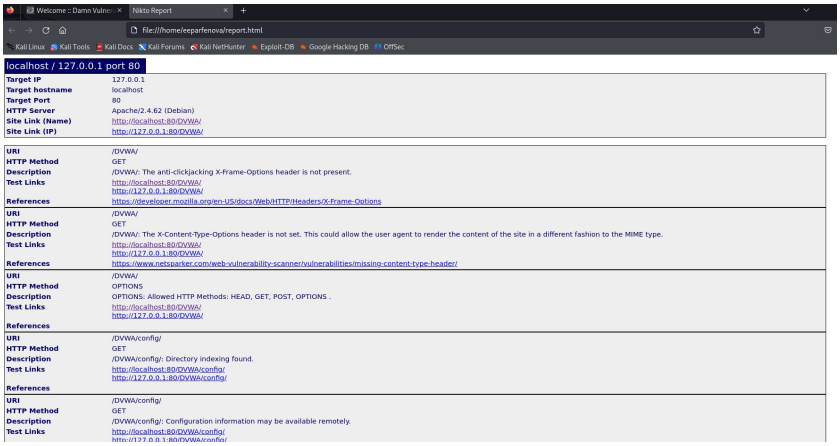

Рис. 3.5: Отчет в формате html

Рис. 3.6: Команда для сканирования с отчетом в формате html

Итак, в результате сканирования было найдено 16 уязвимостей. В них входят:

- Отсутствие заголовка X-Frame-Options, что делает сайт уязвимым для атак типа clickjacking
- Отсутствие заголовка X-Content-Type-Options, что может позволить браузеру обрабатывать содержимое некорректно, не того MIME-типа, который был определен и не должен быть изменен
- Обнаружена индексация каталогов в /DVWA/config/, /DVWA/tests/, /DVWA/database/ и т.д., что может позволить доступ к конфиденциальной информации
- Найдены страницы для входа в административную панель и конфигурационные файлы Git (скрытая папка Git), которые могут содержать важную информацию о структуре проекта (сайта) и репозитории

Также было указано, что доступны 4 метода HTTP: OPTIONS, GET, POST, HEAD.

Во втором случае, сканирование будет выглядеть следующим образом (рис. 3.7). При этом вывод команд *nikto -h 127.0.0.1* и *nikto -h 127.0.0.1 -p 80* (указание порта локального хоста) (рис. 3.8) идентичен

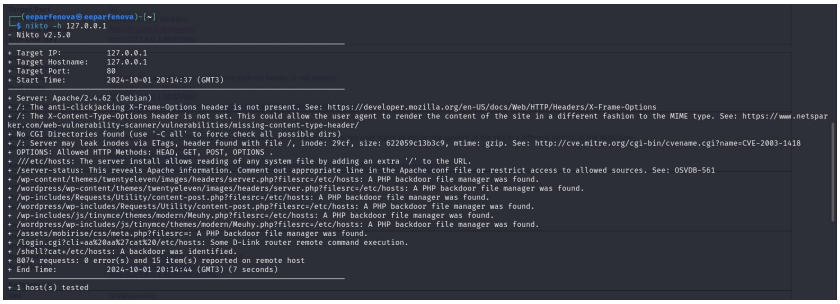

Рис. 3.7: Сканирование с помощью IP

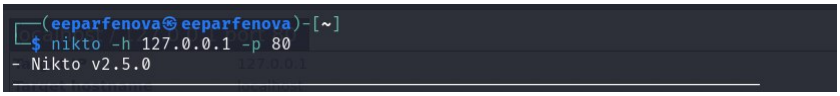

Рис. 3.8: Команда сканирования с IP и портом

При таком сканировании было найдено 15 уязвимостей. Помимо похожих с предыдущим случаем уязвимостей, были найдены:

- Утечка информации о файловой системе через ETags
- Уязвимость чтения системных файлов с помощью манипуляций с URL (/etc/hosts)
- PHP Backdoor file manager был обнаружен в нескольких местах, что свидетельствует о возможной зараженности системы бэкдором
- Уязвимость удаленного выполнения команд на роутерах D-Link

4 Выводы

Мы использовали веб-сканера Nikto для сканирования уязвимостей веб-приложений.

Список литературы

1. Nikto [Электронный ресурс]. Wikimedia Foundation, Inc., 2024. URL: https://ru.wikipedia.org/wiki/Nikto.
2. Обзор сканера Nikto для поиска уязвимостей в веб-серверах [Электронный ресурс]. Habr, 2023. URL: https://habr.com/ru/companies/first/articles/731696/.