GBServerSecurity - Lesson2HW

# TASK 1

ГДЕ НАЙДЕНА УЯЗВИМОСТЬ

Уязвимость найдена в файле index.php (ссылка - <http://192.168.56.11/mutillidae/index.php>).

ОПИСАНИЕ УЯЗВИМОСТИ

В ссылке указывается GET-параметр page, который принимает строку, указывающую путь до какого-либо файла. Принимаемая строка встраивается непосредственно в программный код, из-за чего возникает уязвимость, выражающаяся в возможности просмотра файлов и каталогов сервера.

ТЕХНИЧЕСКИЕ ДЕТАЛИ ОБНАРУЖЕНИЯ И ВОСПРОИЗВЕДЕНИЯ

При условии, что на сервере развёрнута ОС семейства Unix, передав в данный GET-параметр n-ое количество последовательностей dot-dot-slash (../) можно добраться до корневой папки ОС, а после прописывать типические пути расположения ценных файлов (например, /etc/passwd), которые, в последствии, будут отрисованы на странице index.php и свободны для ознакомления злоумышленником.

ВЫВОДЫ И РЕКОМЕНДАЦИИ ПО УСТРАНЕНИЮ ИЛИ СНИЖЕНИЮ ПОСЛЕДСТВИЙ ОТ ЭКСПЛУАТАЦИИ УЯЗВИМОСТИ

Уязвимость позволяет получить доступ к конфиденциальной информации. Не требует дополнительных уязвимостей для эксплуатации. Рекомендации по устранению:

* Запретить просмотр каталогов в веб-сервере.
* Проверять передаваемые в GET-параметр данные на отсутствие в них последовательностей dot-dot-slash.

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, КОТОРОЕ ИСПОЛЬЗОВАЛОСЬ ДЛЯ ТЕСТИРОВАНИЯ

* Командная строка.
* Сканер Nikto.

# TASK 2

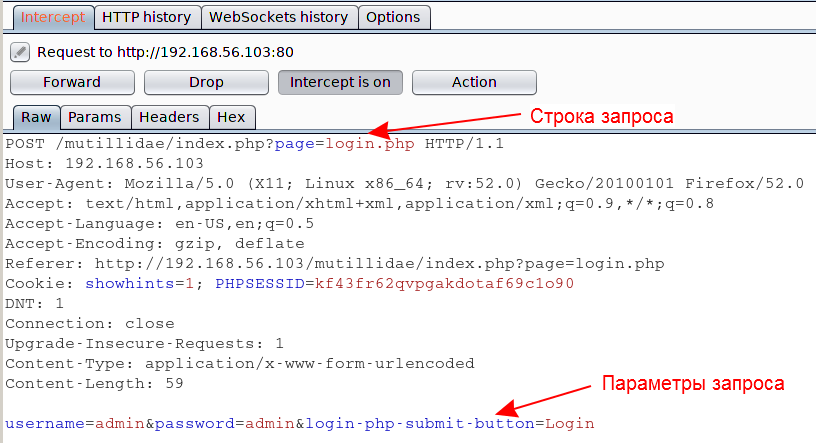
ГДЕ НАЙДЕНА УЯЗВИМОСТЬ

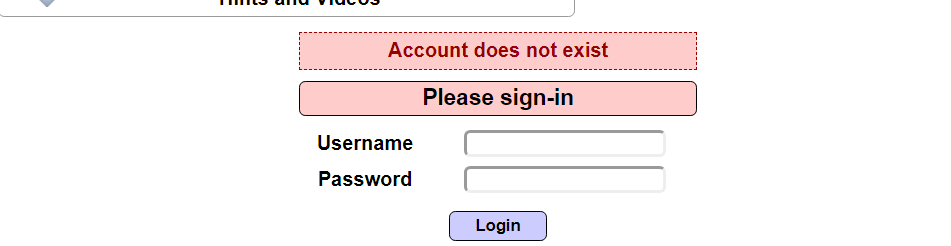
Уязвимость найдена в форме для ввода логина и пароля, который используется для входа в систему (<http://192.168.56.103/mutillidae/index.php?page=login.php>).

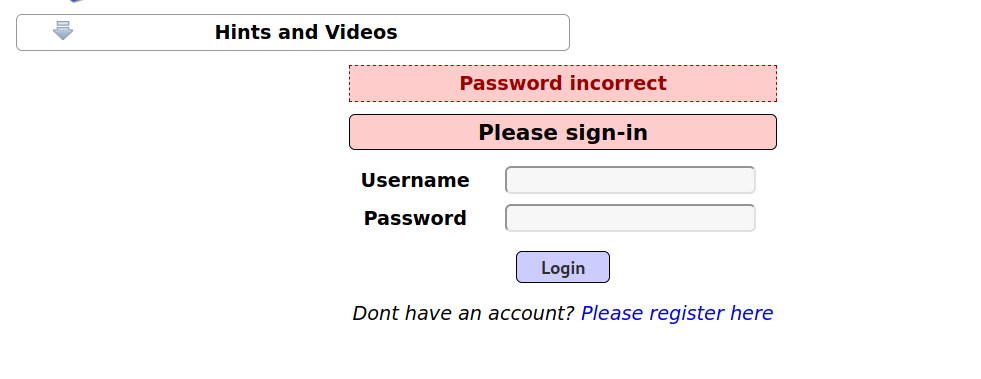
ОПИСАНИЕ УЯЗВИМОСТИ

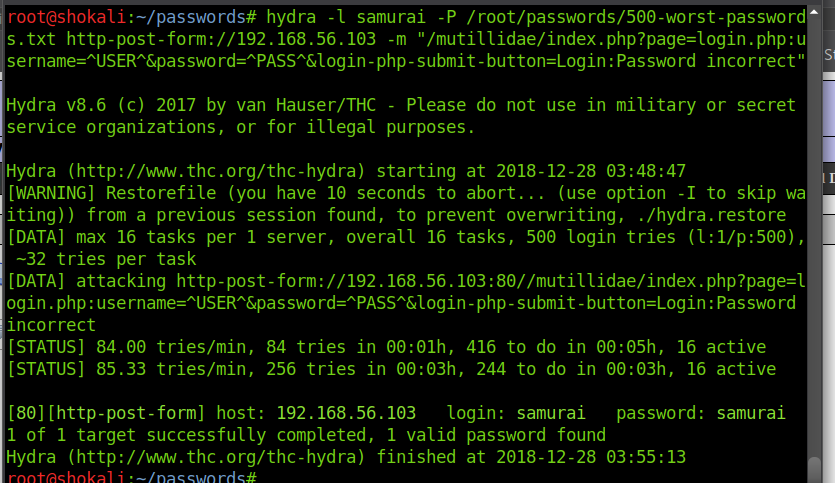
Уязвимость выражается в возможности реализации атаки типа bruteforce, поскольку количество попыток ввода логина и пароля является неограниченным. Причём сначала присутствует возможность подбора логина, реализация которой упрощает для злоумышленника задачу в виде аутентификации и последующей авторизации.

ТЕХНИЧЕСКИЕ ДЕТАЛИ ОБНАРУЖЕНИЯ И ВОСПРОИЗВЕДЕНИЯ

Вход пользователя проводится при помощи POST-запроса, Cookie не используются.

При вводе неверного логина выводится сообщение «Account does not exist». 

При вводе верного логина, но неверного пароля выводится сообщение «Password incorrect». 

Задача сводится к тому, чтобы повторить при помощи утилиты запрос, отвечающий за вход пользователя в систему. Воспользуемся утилитой hydra:

ВЫВОДЫ И РЕКОМЕНДАЦИИ ПО УСТРАНЕНИЮ ИЛИ СНИЖЕНИЮ ПОСЛЕДСТВИЙ ОТ ЭКСПЛУАТАЦИИ УЯЗВИМОСТИ

Уязвимость позволяет авторизоваться на сервисе под чужим пользователем. Не требует дополнительных уязвимостей при эксплуатации. Рекомендации по устранению:

* Ограничивать количество попыток на ввод пароля.
* Использовать системы обнаружения взлома.
* Заставлять пользователей придумывать сложные пароли.
* Заблокировать POST-запросы, в которых отсутствует значение Referer.
* Соблюдать требования безопасности, например, минимизировать привилегии для пользовательских аккаунтов.
* Регулярно менять пароли, соблюдать требования к их сложности.
* Использовать **fail2ban** или **web application firewall** (например, модуль **mod\_security** для **apache2**) для защиты от эксплуатации уязвимостей методом bruteforce.

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ, КОТОРОЕ ИСПОЛЬЗОВАЛОСЬ ДЛЯ ТЕСТИРОВАНИЯ

* Командная строка.
* Burp Suite.
* Hydra.