ПЕРСПЕКТИВНЫЙ МОНИТОРИНГ

AMPIRE

Программный комплекс обучения методам обнаружения, анализа и устранения последствий компьютерных атак Ampire

ПОЛНОЕ ОПИСАНИЕ УЯЗВИМОГО УЗЛА «DUPLICATOR»

Аннотация

Методическое пособие «Полное описание уязвимого узла Duplicator» содержит подробное описание узла `Duplicator` с уязвимостью, позволяющей получать содержимое любого файла с правами веб-сервера.

Данное пособие предназначено для повышения качества результатов обучения специалистов на учебно-тренировочной платформе `Ampire`, их знакомства с наиболее актуальными угрозами информационной безопасности и способами защиты от них.

Содержание

1 Описание уязвимого узла

Полное описание уязвимого узла Duplicator представлено в таблице (Таблица 1).

Таблица 1 – Описание уязвимого узла Duplicator

Общие сведения		
Название хоста	Server-web-wordpress-duplicator	
Описание хоста	Веб-сервер с CMS WordPress с уязвимой версией плагина Duplicator	
Тип хоста	Веб-сервер	
Стек технологий	Ubuntu 18.04 LTS PHP7.2-34 Apache 2.4.41 CMS WordPress 4.6	
Количество уязвимостей	1	
Количество последствий	5	
Параметры доступа		
	Логин	Пароль
SSH	user	qwe123!@#
Панель администрирования CMS WordPress	admin	qwe123!@#
MySQL	admin_joe	IAmKingJoe!
Технические характеристики хоста		
Образ виртуальной машины	Формат OVA	
Жесткий диск	20 GB	
Память	1 GB	
Сеть	DMZ	
Автозапуск сервисов	Apache MySQL	

Уязвимость № 1	
Описание уязвимости	Уязвимая версия плагина `Duplicator` для `CMS WordPress`, которая позволяет получать содержимое любого файла ОС с правами веб-сервера
Условия эксплуатации уязвимости	– знание `IP`-адреса машины; – версия `Duplicator 1.3.26`
Способы обнаружения эксплуатации уязвимости	– логи веб-сервера: в файле `/var/log/apache2/access.log` должен быть зафиксирован `GET`-запрос к `/wp-admin/admin-ajax.php?action=duplicator_download&file=../../../../../etc/passwd`; – средства обнаружения вторжений: `ViPNet IDS NS` и `Security Onion` обнаруживают в сетевом трафике программный код, предназначенный для эксплуатации уязвимости `Path Traversal`
Способы закрытия уязвимости	– обновление версии плагина `Duplicator` до версии `1.3.28` и выше; – отключение плагина через панель администратора `Wordpress`; – изменение исходного кода плагина – в функции `duplicator_init`, после определения переменной `$file` вставить следующий код, проверяющий путь на наличие “`../`”:

	`If (($file == ‘’) || (strpos($file, ‘../’) !== false)) { exit(‘Invalid Request’); }`
Способ проверки статуса уязвимости	Внутренняя проверка статуса уязвимости: вывод попытки обращения к `http:/host_ip/wp-admin/admin-ajax.php?action=duplicator_download&file=../../../../../etc/passwd` (не зависит от способа закрытия уязвимости)
Полезная нагрузка	
Полезные нагрузки	– получение `meterpreter`-сессии; – `Deface` сайта; – `Dump MySQL`; – `Caidao PHP backdoor`; – загрузка вредоносного `PHP-backdoor` в директорию веб-сервера, позволяющего выполнять различную полезную нагрузку на уязвимой системе (например, установку `reverse shell`)
Способы проверки статуса полезной нагрузки	– `Meterpreter`-сессия: наличие сокета с узлом нарушителя; – `Deface` сайта: сравнение файла в каталоге сайта `/var/www/html/wordpress/wp-content/uploads/2021/06/tech-300x200.jpg` с шаблонным; – `Dump MySQL`: проверка на наличие файла `/var/www/html/wordpress/wp-content/DUMP`

	– Caidao PHP backdoor: • проверка наличия файла backdoor, инициализирующего reverse shell, в директории веб-сервера; • проверка наличия сокетов с узлом нарушителя

2 Описание уязвимости

Duplicator – это плагин, используемый администраторами сайтов WordPress для «миграции и копирования сайтов». Часть данной функциональности включает в себя экспорт базы данных и содержимого файлов в переносимые архивы. Когда администратор создает новую копию своего сайта, Duplicator позволяет ему загружать сгенерированные файлы со своей панели управления WordPress.

Информацию о найденных уязвимостях в плагинах можно найти, к примеру, на следующих сайтах:

WordPress Vulnerabilities – найденные уязвимости плагинов и тем WordPress.

WordFence – найденные уязвимости CMS WordPress и связанных продуктов.

CVE – найденные уязвимости CMS WordPress и связанных продуктов.

Уязвимость в плагине Duplicator WordPress позволяет получить конфигурационный файл или любой другой файл сайта, что в свою очередь практически полностью может открыть доступ к управлению и изменению сайта. Удаленный нарушитель, не прошедший проверку подлинности, может использовать данную уязвимость, отправив специально созданный запрос на сайт WordPress с помощью уязвимой версии плагина Duplicator.

2.1 Обнаружение эксплуатации уязвимости в плагине Duplicator WordPress

Суть эксплуатации заключается в GET-запросе к:

http://host/wp-admin/admin-ajax.php?action=duplicator_download&file=../../../../../file.

2.1.1 Обнаружение средствами ОС

Необходимо напомнить, что уязвимость в плагине `Duplicator WordPress` позволяет получить любой файл в системе, к которому имеет доступ пользователь `www-data`.

Для получения необходимого файла нарушитель будет обращаться по ссылке:

`GET/wp-admin/admin-ajax.php?action=duplicator_download&file=../../../../../file`.

Данный факт будет детектироваться в логах сервера `apache2`, которые можно найти по пути `/var/log/apache2/access.log` (Рисунок 1).

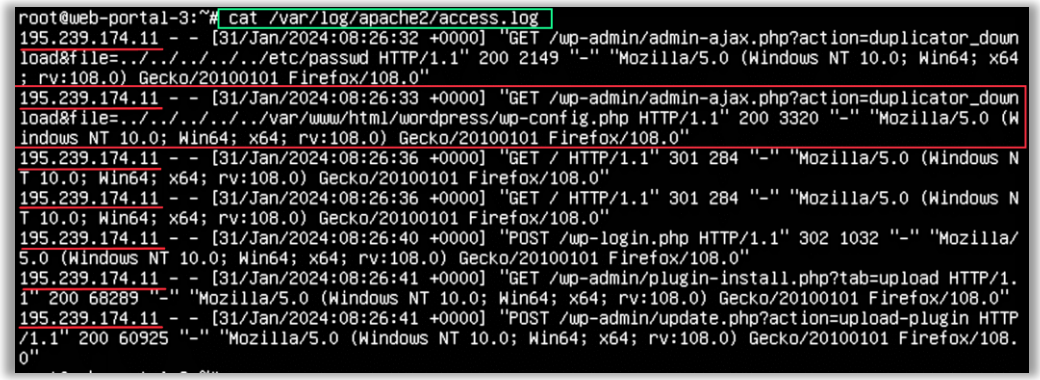


Рисунок 1 – Пример запроса, эксплуатирующего уязвимость в плагине `Duplicator WordPress`

2.1.2 Обнаружение средствами ViPNet IDS NS

Обнаружение уязвимости в сетевом трафике в `ViPNet IDS NS` успешно определяется методом сигнатурного анализа файлов. Сетевой сенсор регистрирует событие информационной безопасности (ИБ) с высоким уровнем важности (красная метка).

Для данной уязвимости в `ViPNet IDS NS` правило обнаруживает в сетевом трафике программный код, предназначенный для эксплуатации уязвимости `Path Traversal` (Рисунок 2).

Path Traversal (обход пути) – это тип атаки, которая позволяет нарушителю получить доступ к конфиденциальным файлам и каталогам, хранящимся в файловой системе веб-приложения, с помощью изменения существующих путей к файлам через параметры веб-приложения.

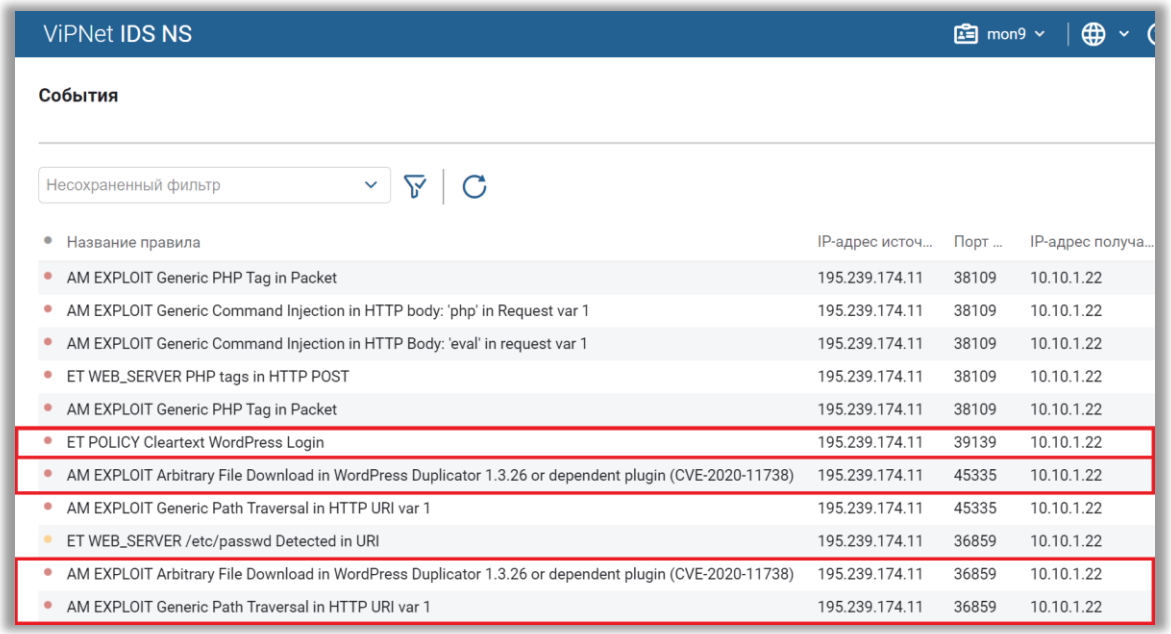


Рисунок 2 – Журнал событий в веб-интерфейсе ViPNet IDS NS

Подробно проанализировать атакующий пакет можно специализированным сторонним программным обеспечением (например, Wireshark), сохранив содержимое пакета, связанного с AM EXPLOIT в формате PCAP (Рисунок 2). Содержимое пакета, зафиксированного в ViPNet IDS NS, представлено на скриншоте (Рисунок 3).

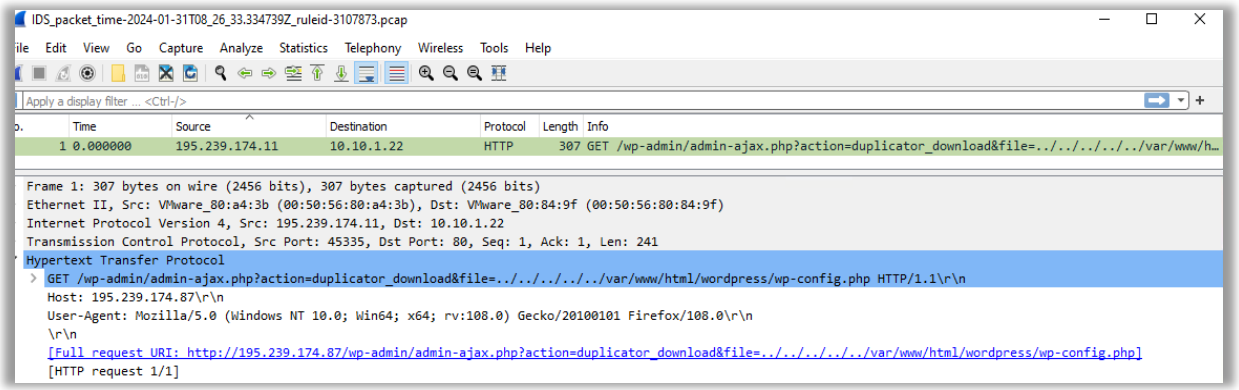


Рисунок 3 – Содержимое пакета, описывающего событие ИБ

Следующее за сработавшим правилом типа AM EXPLOIT событие ИБ, зафиксированное сетевым сенсором: ET POLICY Cleartext WordPress (Рисунок 2) – нарушение политики информационной безопасности, свидетельствующее о том, что найден пароль wordpress admin_joe в /wp-login.php (Рисунок 4).

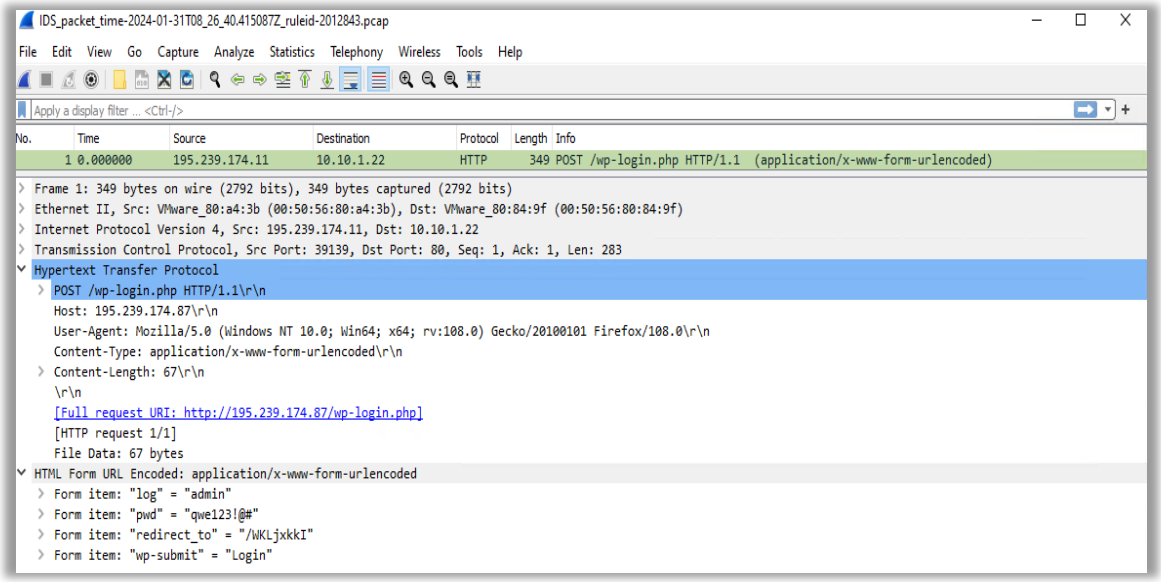


Рисунок 4 – Содержимое пакета, описывающего событие нарушения политики ИБ

Далее необходимо убедиться и подтвердить атаку нарушителем непосредственно на узле CMS WordPress (по схеме – IP-адрес 10.10.1.22).

2.1.3 Обнаружение средствами Security Onion

Для обнаружения последствий эксплуатации с помощью Security Onion следует использовать утилиту Squert – визуальный инструмент, предоставляющий дополнительный контекст для событий с помощью метаданных.

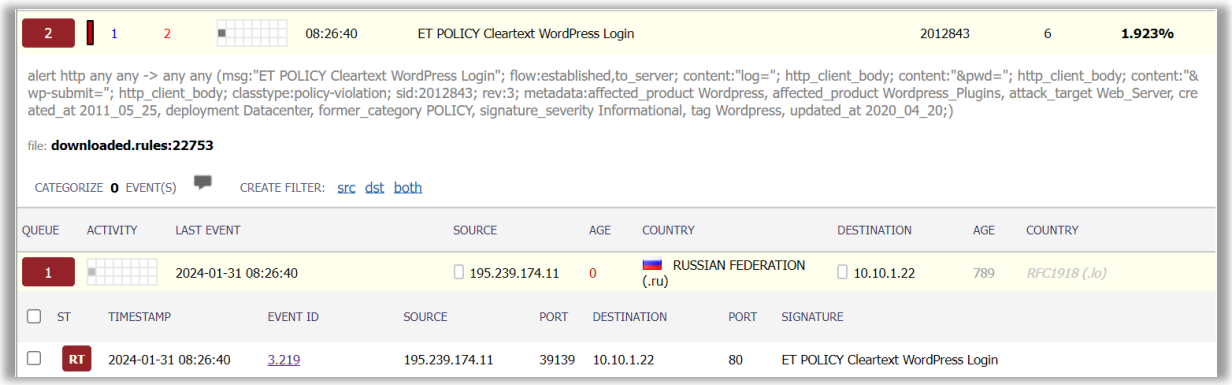
На скриншоте (Рисунок 5) представлен перечень всех событий, в котором целью является уязвимый сервер.

6	1	2		08:27:18	ET WEB_SERVER PHP tags in HTTP POST	2011768	6	5.769%
4	1	2		08:26:57	GPL ICMP_INFO PING *NIX	2100366	1	3.846%
2	1	2		08:26:40	ET POLICY Cleartext WordPress Login	2012843	6	1.923%
3	1	1		08:26:31	ET ATTACK_RESPONSE Possible /etc/passwd via HTTP (linux style)	2002034	6	2.885%
3	1	1		08:26:31	ET ATTACK_RESPONSE passwd file Outbound from WEB SERVER Linux	2025879	6	2.885%
1	1	1		08:26:31	ET WEB_SERVER /etc/passwd Detected in URI	2049400	6	0.962%

Рисунок 5 – События, связанные с эксплуатацией уязвимости

В перечне представлены запросы для получения информации о сервере, также представлены запросы для получения подключения вредоносного TCP-соединения с уязвимой машиной через загруженную полезную нагрузку в формате PHP.

На следующем скриншоте (Рисунок 6) представлено аналогичное событие ET POLICY Cleartext WordPress.



2 | 1 | 2 | 08:26:40 | ET POLICY Cleartext WordPress Login | 2012843 | 6 | 1.923%

alert http any any -> any any (msg:"ET POLICY Cleartext WordPress Login"; flow:established,to_server; content:"log="; http_client_body; content:"&pwd="; http_client_body; content:"&wp-submit="; http_client_body; classtype:policy-violation; sid:2012843; rev:3; metadata:affected_product Wordpress, affected_product Wordpress_Plugins, attack_target Web_Server, created_at 2011_05_25, deployment Datacenter, former_category POLICY, signature_severity Informational, tag Wordpress, updated_at 2020_04_20;)

file: **downloaded.rules:22753**

CATEGORIZE **0** EVENT(S) CREATE FILTER: src dst both

QUEUE	ACTIVITY	LAST EVENT	SOURCE	AGE	COUNTRY	DESTINATION	AGE	COUNTRY
1		2024-01-31 08:26:40	195.239.174.11	0	RUSSIAN FEDERATION (.ru)	10.10.1.22	789	RFC1918 (.lo)

ST	TIMESTAMP	EVENT ID	SOURCE	PORT	DESTINATION	PORT	SIGNATURE
RT	2024-01-31 08:26:40	3.219	195.239.174.11	39139	10.10.1.22	80	ET POLICY Cleartext WordPress Login

Рисунок 6 – Запросы нарушителя к странице авторизации

Следующее событие (Рисунок 7) отображает получение нарушителем. Используя предварительно загруженный файл backdoor пытается установить соединение с уязвимой машиной, инициирующее создание стабильного TCP-сокета между уязвимым хостом и машиной нарушителя.

Рисунок 7 – Попытки нарушителя установить `shell`-соединение

2.2 Устранение уязвимости

Устранение уязвимости можно осуществить несколькими способами:

1) обновление версии плагина `Duplicator` до версии 1.3.28 и выше;

Примечание: данный способ не будет работать в `Ampire`, так как узлы не подключены к интернету;

2) отключение плагина через панель администратора `WordPress`;

3) изменение исходного кода плагина.

Каждый способ подробно описан далее по тексту.

Следует отметить, что индикатор устранения уязвимости не изменится, пока не будет устранено последствие в виде вредоносного соединения. Рекомендации по закрытию вредоносного соединения представлены в соответствующем описании (Подраздел 3.3).

2.2.1 Обновление версии плагина

Данная уязвимость исправлена разработчиками в версии `1.3.28`. Для обновления версии плагина `Duplicator` необходимо зайти в панель администратора по адресу: `http://host/wp-admin` (Рисунок 8).

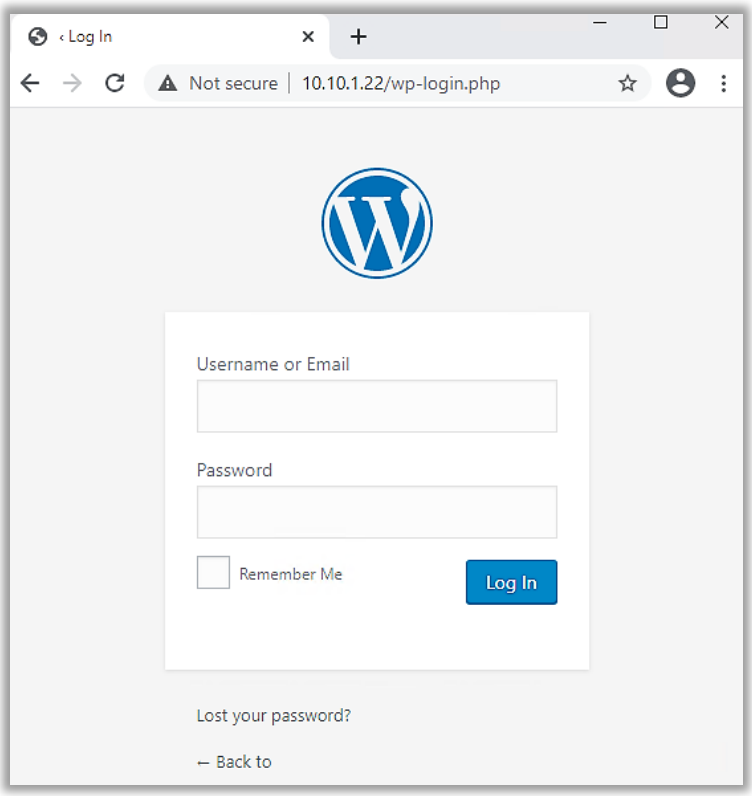


Рисунок 8 – Вход в панель администратора `CMS WordPress`

Далее ввести логин и пароль администратора, в боковом меню панели выбрать раздел `Plugin — Installed Plugins` (Рисунок 9).

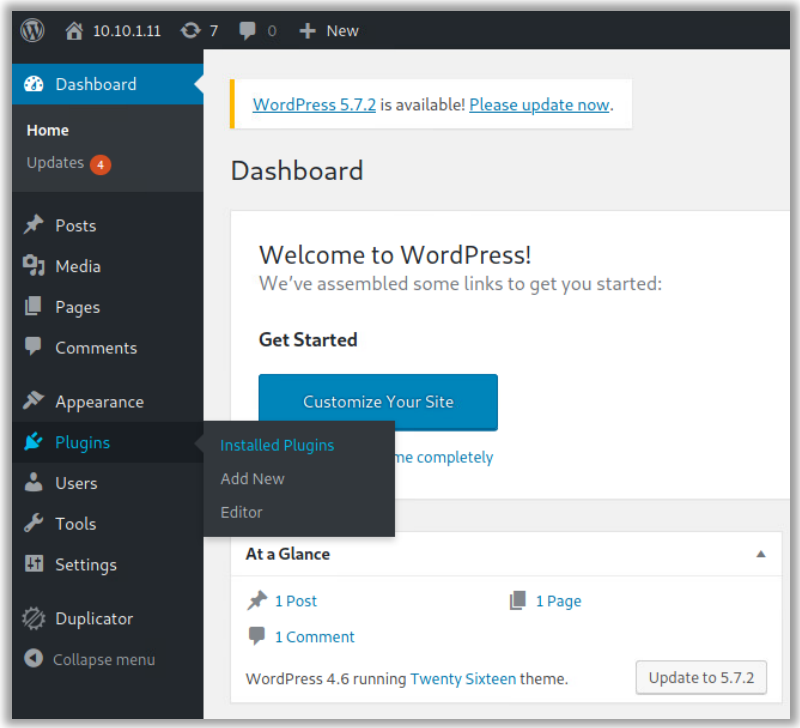


Рисунок 9 – Обновление версии плагина

Далее в списке плагинов найти Duplicator. WordPress предупредит о том, что доступна новая версия плагина, далее нужно нажать на Update now (Рисунок 10).

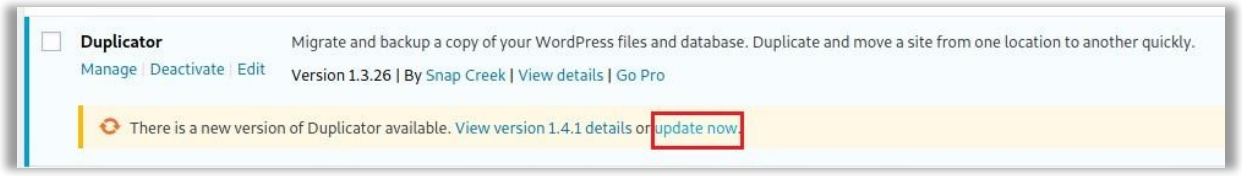


Рисунок 10 – Обновление версии плагина

2.2.2 Отключение плагина

После входа в панель администратора для отключения плагина нужно в боковом меню выбрать раздел Plugins, в списке плагинов найти Duplicator. Далее нажать на ссылку Deactivate, перевести плагин в неактивное состояние (Рисунок 11).

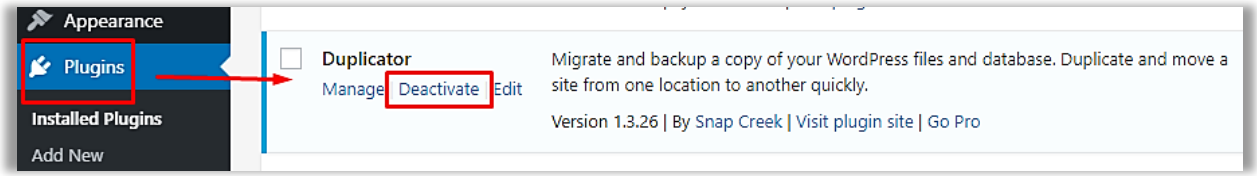


Рисунок 11– Перевод в неактивное состояние плагина

2.2.3 Изменение исходного кода плагина

Для изменения исходного кода плагина необходимо после входа в панель администратора в боковом меню выбрать раздел Plugins, в списке плагинов найти Duplicator. Далее нажать на ссылку Edit (Рисунок 12), в редакторе исходного кода найти функцию duplicator_init(). Внутри данной функции после определения переменной $file вставить следующий код:

```
if (($file == '') || (strpos($file, '../') !== false)) {
    exit('Invalid Request');
}
```

Данный код будет проверять путь в запросе на наличие `'../'` паттерна. Если запрос будет содержать паттерн, то плагин будет завершать свою работу.

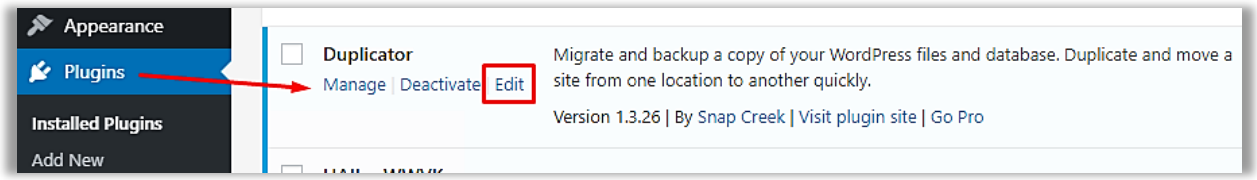


Рисунок 12 – Изменение исходного кода плагина

Для сохранения изменений и завершения процесса необходимо активировать ссылку `Update File` (Рисунок 13).

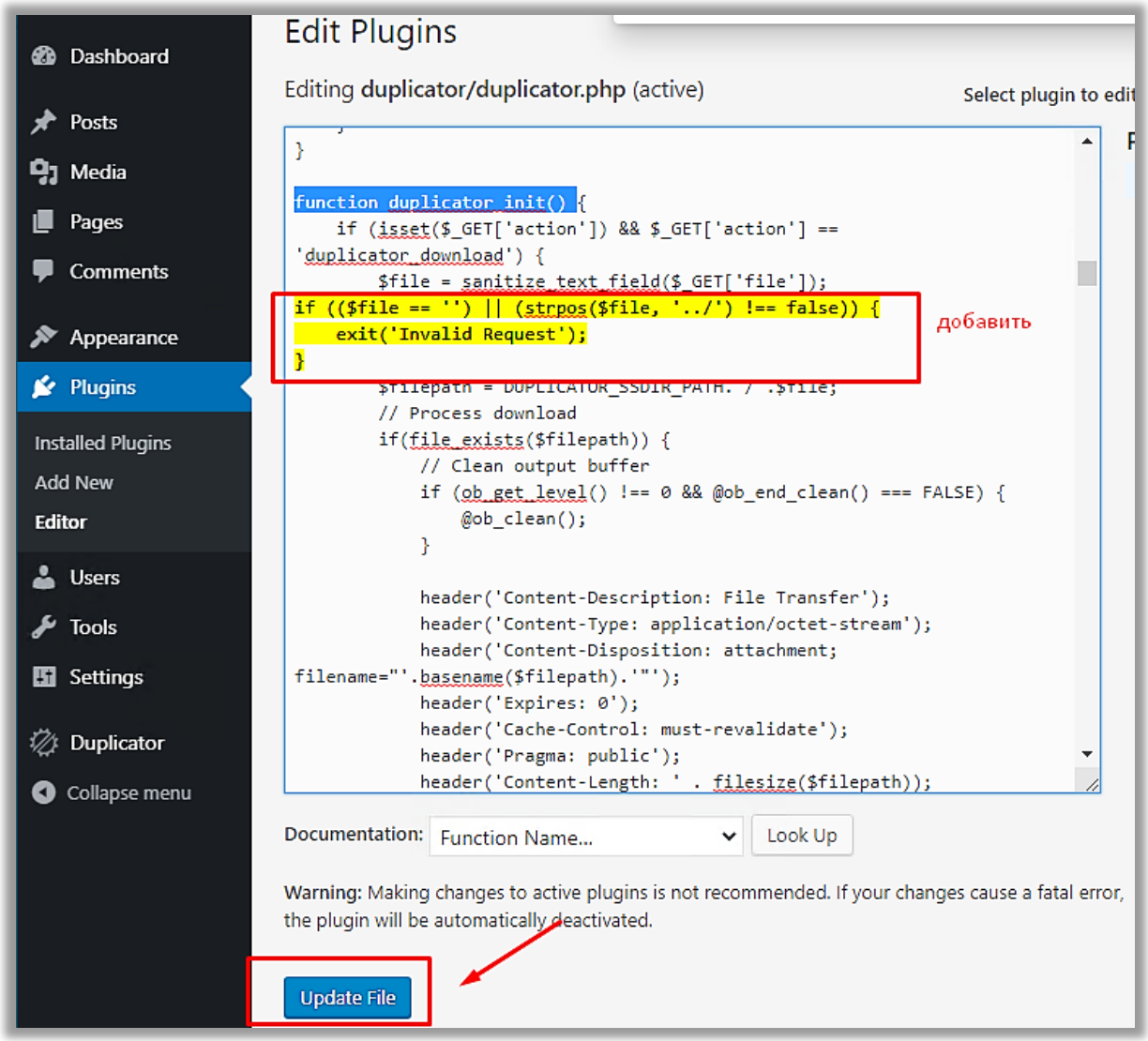


Рисунок 13 – Изменение исходного кода плагина

3 Обнаружение и нейтрализация полезных нагрузок

Уязвимый узел позволяет реализовать следующие полезные нагрузки:

1) `Deface` сайта;
2) `Dump` базы данных;
3) `Meterpreter`-сессия;
4) `Caidao PHP backdoor`.

3.1 Deface сайта

Данная полезная нагрузка подразумевает изменение интерфейса главной страницы сайта. Для обнаружения полезной нагрузки необходимо просмотреть главную страницу сайта.

Интерфейс главной страницы сайта до изменений представлен на скриншоте (Рисунок 14).

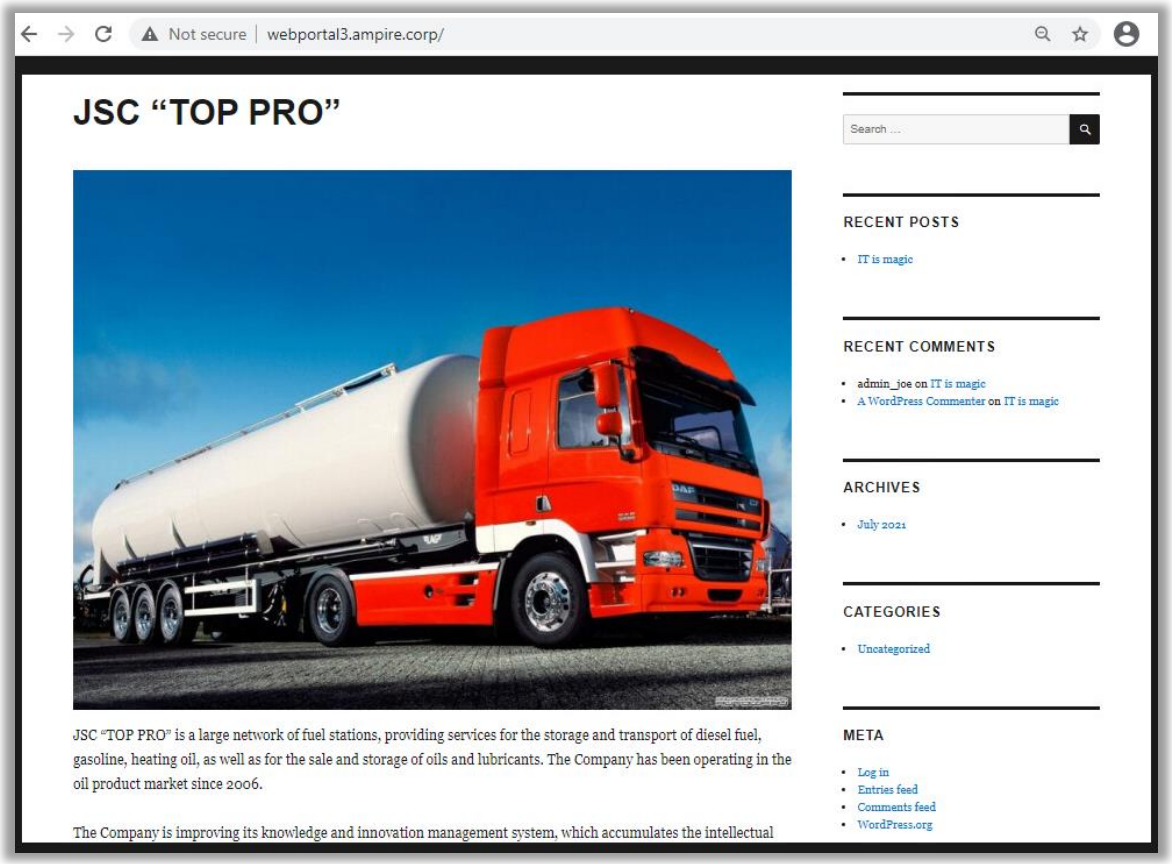


Рисунок 14 – Главная страница сайта до изменений

Интерфейс главной страницы сайта после атаки представлен на скриншоте (Рисунок 15).

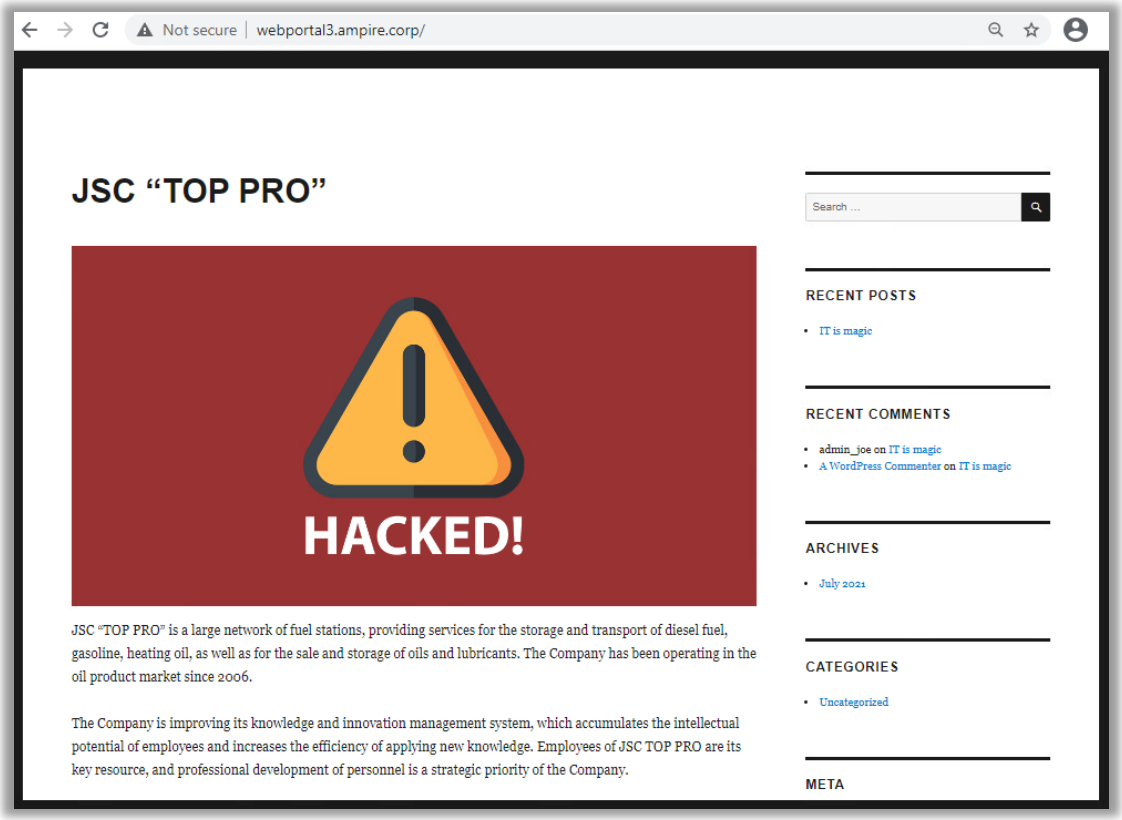


Рисунок 15 – Главная страница сайта после атаки

Для нейтрализации данной полезной нагрузки необходимо получить путь, по которому можно найти измененное изображение. Необходимо нажать на изображение правой кнопкой и найти пункт `Copy image link` (или аналогичный). Скопированный путь будет выглядеть следующим образом: `http://10.10.1.22/wp-content/uploads/2021/11/sitesecute-1.jpeg`

Данная ссылка будет полностью совпадать с путем к файлу в самой операционной системе. Файл можно будет найти по пути: `/var/www/html/wordpress/wp-content/uploads/2021/11/sitesecute-1.jpeg`.

В данной директории будет находиться резервная копия изображения `/var/www/html/wordpress/wp-content/uploads/2021/11/sitesecute-1.jpeg.bak`. Необходимо заменить измененный файл на резервную копию исходного изображения.

3.2 Dump базы данных

Данная полезная нагрузка заключается в создании нарушителем в определенной директории резервной копии базы данных MySQL, которую нарушитель скачивает.

Если нарушитель невнимателен и оставил после себя файл резервной копии, то по детектированию дампа базы данных можно определить, куда же нарушитель мог закачать резервную копию (в какой директории искать файл). Известно, что нарушитель действовал от лица пользователя www-data, которому доступны далеко не все директории на запись. Данный факт сужает круг поиска. Также известно, что данный пользователь является не обычным, а системным пользователем (создан для управления процессом), в связи с чем www-data не будет иметь домашней директории. Тогда очевидно, что данный пользователь имеет права на запись в директории CMS, то есть: /var/www/html/wordpress/*.

Для нейтрализации данной полезной нагрузки необходимо:

1) выполнить поиск в дочерних директориях /var/www/html/wordpress/*, найти файл DUMP, который находится в каталоге wp-content;

2) удалить файл wp-content/DUMP.

3.3 Meterpreter-сессия

Цель данной полезной нагрузки – получение нарушителем Meterpreter-сессии с уязвимым сервером.

Обнаружить данную полезную нагрузку можно с помощью утилиты ss с ключами t, p и n. В случае установления соединения на уязвимой машине появится сокет с машиной нарушителя (Рисунок 16).

В Linux у процесса имеется уникальный идентификатор PID. При создании каждому процессу автоматически присваивается PID.

Для завершения соединения с машиной нарушителя необходимо принудительно остановить процесс с помощью команды `kill` вместе с номером процесса.

Прервать работу процесса без угрозы игнорирования можно с помощью команды `SIGKILL` (номер 9). Данная команда немедленно прерывает процесс (Рисунок 17).

Рисунок 16 – Отображение информации о `TCP`-соединениях

Рисунок 17 – Процесс закрытия `meterpreter`-сессии

Если после выполнения команды осталось соединение в процессе `LAST ACK`, то на данное соединение не следует ориентироваться.

`LAST ACK` (последнее подтверждение) является одним из состояний, используемых в протоколе `TCP`. Данное состояние возникает в процессе закрытия `TCP`-соединения и в скором времени пропадет, у нарушителя нет возможности использования `LAST ACK`.

Для устранения данной полезной нагрузки необходимо:

1) выполнить команду `ss -tnp` для обнаружения активных соединений;

2) с помощью команды `sudo kill <PID>` завершить процесс, устанавливающий соединение с хостом нарушителя.

3.4 Caidao PHP backdoor

Данная полезная нагрузка заключается в создании нарушителем вредоносного PHP-файла, который при выполнении генерирует веб-страницу. Данная страница предоставляет нарушителю возможность инициировать обратное соединение (reverse shell) между своей системой и скомпрометированным устройством. Для реализации достаточно отправить специальный запрос, что позволяет атакующему удаленно управлять зараженным устройством.

При установлении соединения через PHP-файл в журнал /var/log/apache2/access.log веб-сервера Apache будет записано соответствующее событие (Рисунок 18).

Для прочтения журнала можно использовать команду:

```
cat/var/log/apache2/access.log | less
```

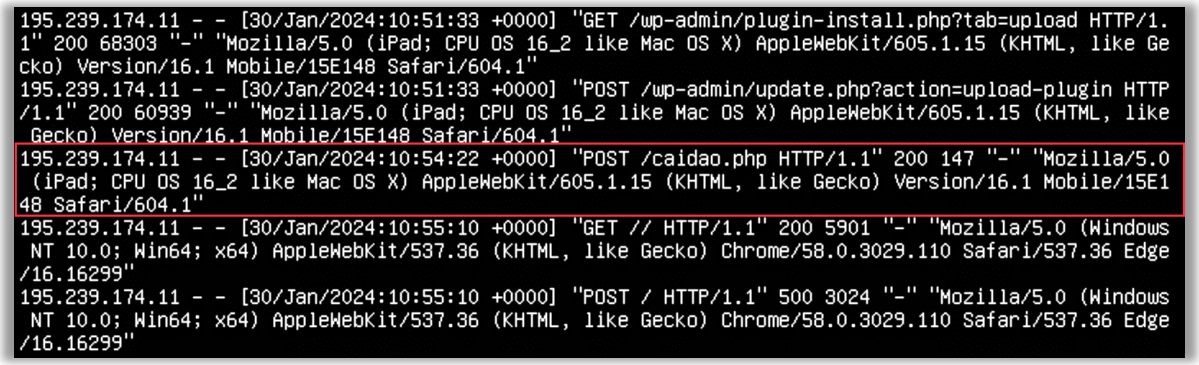


Рисунок 18 – Отображение POST-запроса к файлу caidao.php

Обнаружить вредоносный файл caidao.php можно в директории веб-сервиса, в данном случае – директория /var/www/html/wordpress (Рисунок 19).

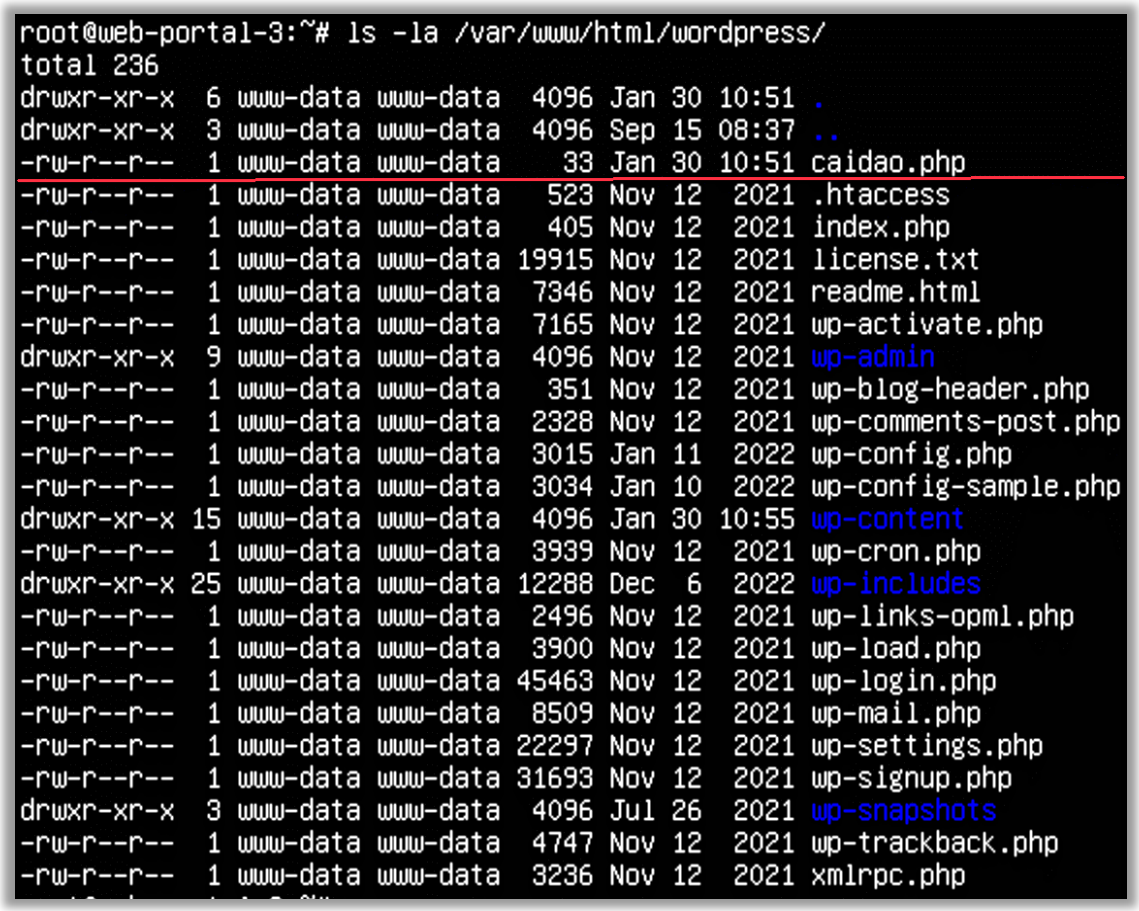


Рисунок 19 – Отображение вредоносного PHP-файла в директории веб-сервиса

Для закрытия полезной нагрузки необходимо удалить вредоносный PHP-файл, а также завершить все соединения с хостом нарушителя.

Для удаления файла необходимо выполнить команду rm с указанием абсолютного (Рисунок 20) или относительного (Рисунок 21) пути до удаляемого файла.

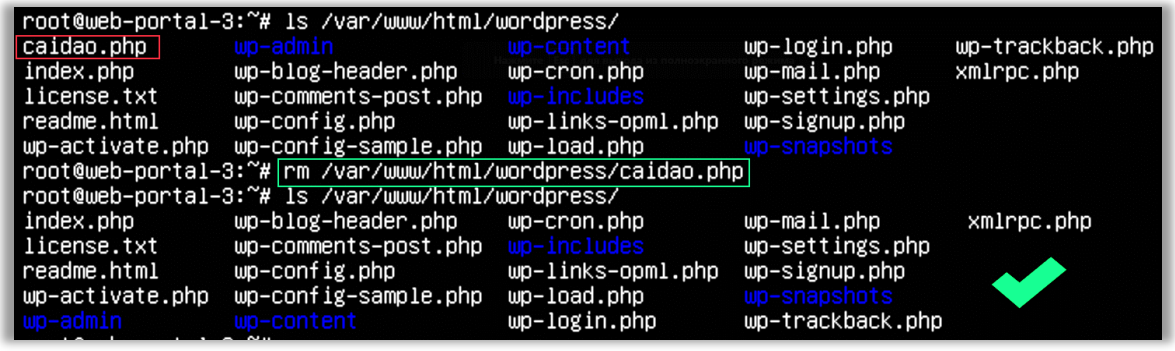


Рисунок 20 – Удаление вредоносного PHP-файла по абсолютному пути

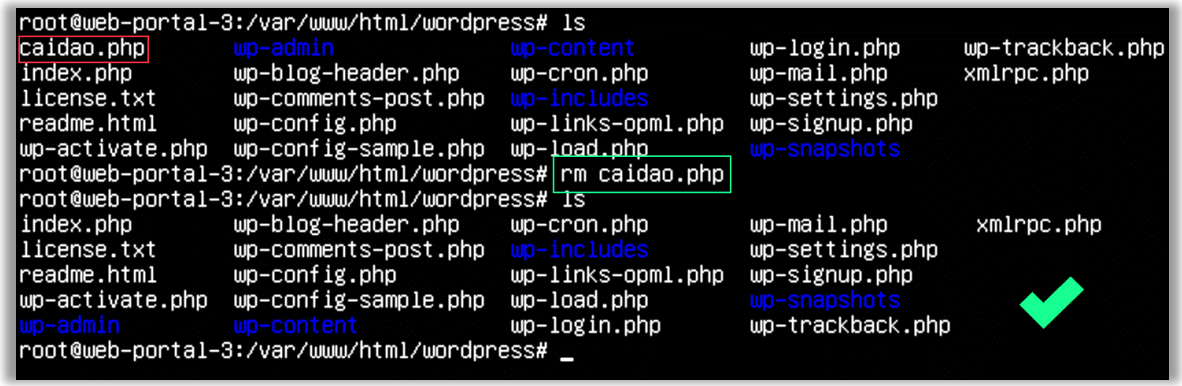


Рисунок 21 – Удаление вредоносного PHP-файла по относительному пути

Для устранения данной полезной нагрузки необходимо:

1) удалить PHP-файл `/var/www/html/wordpress/caidao.php`;

2) завершить все соединения между уязвимой машиной и нарушителем, рекомендации по закрытию вредоносного соединения представлены в соответствующем описании (Подраздел 3.3).