**ОТЧЕТ ОБ ИССЛЕДОВАНИИ**

|  |  |  |
| --- | --- | --- |
| Информация об исследуемой ОС |  | Ubuntu 16.04.6 LTS (xenial) kernel 4.4.0-142-generic |
|  |  |  |
| Время начала инцидента |  | 14:53:34 (сканирование портов); 14:55:56 (вход в админку); 14:58:08 (/etc/passwd); 15:06:56(ssh test@) |
|  |  |  |
| Время окончания инцидента |  | 15:09 (выход из сессии с хостом 192.168.1.72) |
|  |  |  |
| Пользователь пострадавший в инциденте |  | test |
|  |  |  |
| Классификация злоумышленника |  | Есть классификатор ? :) |
|  |  |  |
| IP адрес злоумышленника |  | 192.168.1.38 |
|  |  |  |
| Схема заражения ОС |  | 1. Сканирование портов (Успех: ssh(22), telnet(23), http(80))  2. Определение используемого вэб-сервера (apache2) и его движка (WordPress 5.2.2) при помощи инструментария nmap.   3. Анализ уязвимостей данной версии движка при помощи WordPress Vulnerability Scanner - WPScan v3.6.3  4. В админ-панель зашел в 14:55:56 используя admin:admin (предполагаю что эти данные он смог увидеть используя ранее сниффер пакетов, т. к. данные не шифровались). В нашем дампе в частности также видны эти данные используемые легитимным хостом 192.168.1.53.  5. Подключил плагин insert-php-code-snippet и создал snippet «crl» отображающий содержимое файла пользователей системы (cat /etc/passwd) в 14:57:36  6. Создал страничку с использованием этого snippet и отобразил файл с пользователями в 14:58:08  7. Удалил плагин в 14:58:32  8. Вышел из админки в 14:58:58  9. Запустил подбор пароля ssh для пользователей prod и test в 15:00:10  10. В 15:06:56 подобрал пароль для пользователя test.  11. В 15:08:28 скачал питоновский скрипт clev.py с адреса 192.168.1.38:1111 и сохранил его как .s.py с последующим запуском «python3.7 .s.py». После этого удалил файл .s.py.  Скрипт зашифровал файлы с определенными расширениями, добавляя дополнительное расширение .enc в директории /home/test/. |
|  |  |  |
| Слабости в информационной системе используемые злоумышленником |  | 1. Отсутствие SSL.  2. Простой пароль пользователя test.  3. Отсутствие средств против брутфорса.  4. Отсутствие авторизации по ключу.  5. Использование уязвимостей вэб-движка. |
|  |  |  |
| Индикаторы компрометации и пути до вредоносного ПО или инструментов злоумышленника |  | В системе начиная с домашней папки пользователя test присутствуют зашифрованные файлы с расширением .enc вместо обычных:  */home/test/.bash\_history.enc*  /home/test/data/secrets.txt.enc  home/test/wp\_configs/text.txt.enc  В файле .bash\_history видно как злоумышленник загружает скрипт clev.py c [http://192.168.1.38:1111](http://192.168.1.38:1111/), сохраняет его как .s.py, выполняет его и затем удаляет его.  Создан файл message с сообщением от злоумышленника обратиться на указанный почтовый ящик для дальнейшей информации, указав в заголовке IP адрес атакованной машины. |
|  |  |  |
| Логика работы вредоносного ПО (операции, выполняющиеся на заражаемой системе, взаимодействие с сервером злоумышленника, результат работы) |  | 1. Инициализация переменных путем сбора информации о системе и формирование на их основе идентификатора:  Q.id=6cd70595ac24b8bd8e7c13f84d7dca1865811835e57f0d34e679da2ea6d258bc  2. Создание файла /tmp/.X11.6cd70595ac24b8bd8e7c13f84d7dca1865811835e57f0d34e679da2ea6d258bc содержащим этот идентификатор. Зачем это было сделано я не понял...  3. Формирование набора данных состоящий из 16 слов случайных буквенных сочетаний (длиной 16 символов каждый). Разделитель слов — запятая.  4. Преобразование этих данных в тип байты, преобразование в base64 и отправка на сервер 192.168.1.38:9999. Также отправляется полученный ранее идентификатор.  5. Шифрование файлов (с определенными расширениями) начиная с домашней папки /home/test при помощи пароля, который является 16 словом из набора данных в п.3. (LSNWRarThRdiPLpM) и сохранение их с расширением .enc. Исходные файлы удаляются. |
|  |  |  |
| Алгоритм восстановления системы от действий вредоносного ПО |  | 1. Изучив принцип шифрования данных, пишем свой скрипт дешифровки.  2. Начиная с папки пользователя test расшифровываем файлы (с расширениями взятыми из шифровальщика плюс .enc) используя найденный пароль и восстанавливаем их родное состояние.  3. Зашифрованные файлы я для наглядности не удалял.  4. Чтобы не затереть текущий файл .bash\_history — сделал расширение восстановленному файлу .dec |
|  |  |  |
| Восстановленная информация пользователя |  | 1. Файл .bash\_history - указывать содержимое не буду  2. /home/test/data/secrets.txt: flag{26c08ad080830d6dcd76c15009ab6b03}  3. /home/test/wp\_configs/text.txt:  We need to add configs here |
|  |  |  |
| Рекомендации по настройке ОС для избегания повторения инцидента в будущем |  | 1. Использование ключей для авторизации.  2. Пароли устанавливать сложные.  3. Использование протоколов шифрования — в данном случае использование https для сайта вместо http.  4. Не использовать утилиты которые не шифруют авторизацию — например не использовать telnet.  5. Использовать сервисы против брутфорса, например fail2ban.  6. Следить за обновлениями системы и используемых сервисов — в данном случае ключевой сервис это WordPress. |

Исполнитель: \_Игнатов П.Е.\_/\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ЛИСТ ДЛЯ ЗАМЕТОК**

Меняем пароль пользователя test на сложный.

sudo passwd test

Защищаем пользовательские папки.

sudo chmod 700 /home/test

sudo chmod 700 /home/prod

Запрещаем авторизацию root по сети.

sudo vi /etc/ssh/sshd\_config

#PermitRootLogin prohibit-password

PermitRootLogin no

sudo systemctl restart sshd

Изучение включенных сервисов.

Отключаем лишние сервисы.

sudo systemctl stop inetd

sudo systemctl disable inetd

sudo systemctl stop mdadm

sudo systemctl disable mdadm

sudo systemctl stop iscsid

sudo systemctl disable iscsid

sudo systemctl disable open-iscsi

sudo systemctl stop lvm2-lvmetad

sudo systemctl disable lvm2-monitor

sudo systemctl disable lvm2-lvmetad

sudo systemctl stop lxcfs

sudo systemctl disable lxcfs

sudo systemctl stop snapd

sudo systemctl disable snapd

sudo systemctl disable snapd-seeded

sudo systemctl stop accounts-daemon

sudo systemctl disable accounts-daemon

Корректировка конфигураций некоторых сервисов.

sudo vi /etc/rsyslog.d/50-default.conf

Комментируем последний блок c выводом на /dev/xconsole

sudo systemctl restart rsyslog

sudo vi /etc/cron.d/mdadm

Комментируем задачу.

sudo chmod -x /etc/cron.daily/mdadm

Работа с базой wordpress.

mysql -u wp\_user -p wordpress

select \* from wp\_options where option\_name IN('siteurl','home');

options set option\_value='http://10.13.13.72' where option\_name = 'siteurl';

update wp\_options set option\_value='http://10.13.13.72' where option\_name = 'home';

Меняем пароль для пользователя wp\_user

mysql -u wp\_user -p wordpress

set password = password('...');

vi /var/www/html/wp-config.php

Работа с движком WordPress.

Меняем пароль для администратора

http://10.13.13.72

admin:admin -> ...

Обновляем WordPress и плагины до последней версии через админку.

Обновляем ОС

yum update

yum upgrade

Удовлетворяем требования WordPress:

Tools|Site Health

yum install php-curl php-mbstring php-imagick php-xml php-zip

Удаляем неиспользуемые плагины и темы.

В настройках движка указываем использование https://10.13.13.72 вместо http://10.13.13.72

/etc/apache2/conf.available/security.conf

ServerTokens Prod

ServerSignature Off

Настроить работу только по https (443)

sudo a2enmod ssl

sudo a2ensite default-ssl.conf

vi /etc/apache2/sites-enabled/000-default.conf

Redirect permanent "/" "https://10.13.13.72/"

sudo systemctl restart apache2

Установка fail2ban.

sudo apt install fail2ban\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_