Golden Ticket Attack Explained - MITRE ATT&CK T1558.001

SILA ÖZEREN HACIOĞLU | 9 MIN READ

LAST UPDATED ON NOVEMBER 22, 2024

Summarize with:

ChatGPT perplexity Google AI

The Golden Ticket attack, a sub-technique of the **Steal or Forge Kerberos Tickets: Golden Ticket (T1558.001)** in the **MITRE ATT&CK Framework**, poses a threat to IT environments worldwide. This attack technique can compromise an entire network, granting unauthorized and broad access to domain resources.

This blog delves into the intricacies of this attack strategy, providing an in-depth examination of how it operates with two **different open-source tools (Impacket and Mimikatz)**, its potential impacts, and the **detection and mitigation suggestions** organizations can implement to protect their systems.

What Is the Golden Ticket Attack?

A **Golden Ticket** attack is an attack technique where a malicious actor manipulates the **Kerberos authentication protocol** utilized within Windows networks to gain **unrestricted access** to an organization's entire domain—including devices, files, and domain controllers.

When successful, this method allows unauthorized individuals to exploit the **krbtgt account**, a critical component of the Kerberos system **responsible for the encryption and signing of all domain tickets**.

Attackers first breach the system, then escalate their privileges to that of a domain administrator, allowing them to extract the **NTHash of the krbtgt account** and identify the domain's **Security Identifier (SID)**. Equipped with this information, the attackers then craft a "**Golden Ticket**", a forged Kerberos ticket that functions like a legitimate **Ticket Granting Ticket (TGT)**, often replicating the domain administrator's privileges.

The potency of this attack lies in its ability to provide attackers with extensive and enduring access to the network, effectively circumventing standard authentication checks. The durability of the attack stems from its fundamental distortion of the Kerberos system, remaining potent even after changes to the krbtgt password. To completely negate an active Golden Ticket attack, **the krbtgt password must be altered twice**.

Consequently, the golden ticket attack technique underscores the necessity for comprehensive security measures, such as rigorous password policies and consistent network activity monitoring.

Tools and Techniques to Perform a Golden Ticket Attack

Adversaries can use multiple third-party tools such as **Mimikatz** and **Impacket** to perform a Golden Ticket attack.

Tool 1: Impacket

In this scenario, we will **assume** that upon performing a Kerberoasting attack, an attacker dumped a file of hashes and cracked them to gain administrator access to the Domain Controller. In other words, we have the plaintext password of an administrator user that can access the Domain Controller. In addition, our domain name will be EXAMPLE.local for efficiency.

A typical Golden Ticket attack with Impacket consists of two main parts.

Step 1: Forging a golden ticket

To create a valid golden ticket, certain information is required, such as the NTHash of the domain controller's krbtgt account and the domain SID. This information can be obtained by using the secretsdump.py script from Impacket, provided that the attacker has administrator access to the domain controller. Below, you will find the proper syntax to dump NTHash for the krbtgt account [30].

```
secretdump.py Administrator:"Password"@<DC_IP_Address>
```

Assume that NTHash is bf106a6860c6f7b3317c653a38aba33.

Next, the attacker needs to learn the domain SID. For this, they can leverage Impacket's lookupsid.py tool. Note that even though the attacker chooses the DC as the target, this attack works with any domain controller.

```
lookupsid.py EXAMPLE.local/Administrator:"Password"@<DC_IP_Address>
```

Assume that the domain SID is S-5-1-5-21-2049251289-867822404-1193079966.

Finally, the attacker uses Impacket's ticketer.py tool to forge a golden ticket for a domain user. One advantage of the ticketer.py is that the forged ticket gets written to a .ccache file instead of .kirbi; in other words, the attacker does not have to convert it.

```
ticketer.py -nthash bf106a6860c6f7b3317c653a38aba33 -domain-sid "S-5-1-5-21-2049251289-867822404-1193079966" -domain EXAMPLE.local Alice
```

Note that the command above is an example of an attacker forging a golden ticket for a non-existent domain administrator Alice.

Step 2: Using a golden ticket

To set up the golden ticket for use, the KRB5CCNAME environment variable needs to be set to the path of the .ccache file, which can be an absolute or relative file path. The KRB5CCNAME environment variable is used to inform Impacket tools that support Kerberos tickets where to find the ticket. This allows the attacker to use the golden ticket to access the system as a privileged user [1].

Next, the adversary can use Impacket's command execution tools, such as psexec.py, smbexec.py, or wmiexec.py, to load and authenticate with the ticket, eventually giving the adversary a command execution. For Kerberos authentication to work, the adversary has to provide the IP address of the target, the IP address of the Domain Controller, and the domain name.

```
psexec.py $EXAMPLE.local/$Administrator@$TARGET_NAME -target-ip $TARGET_IP -dc-ip $DC_IP -no-pass -k
```

Note that while the -no-pass option tells the script to skip password-based authentication, the -k option specifies that the Kerberos ticket should be taken from the KRB5CCNAME environment variable. The purpose of this script is to remotely execute commands on the target computer using Kerberos authentication without having to enter a password.

Tool 2: Mimikatz

A typical Golden Ticket attack with Impacket consists of three main parts.

Step 1: Compromising the password hash for the krbtgt account

As it was the case with the Impacket scenario, for a Golden Ticket attack to work, an adversary has to have administrator access to a Domain Controller. Hence, we will start with this assumption.

To exfiltrate the password hash of the krbtgt user, the attacker can use the "lsadump::dcsync" command.

```
PS> mimikatz.exe "lsadump::dcsync /user:DOMAIN\KRBTGT"

SAM Username         : krbtgt
User Principal Name  : krbtgt@DOMAIN.com
Password last change : 09/03/2020 14:51:03
Object Security ID   : S-1-5-21-5840559-2756745051-1363507867-502 #

Credentials:
  Hash NTLM: 1b8cee51fd49e55e8c9c9004a4acc159  # NTLM Hash
. . .
  aes256_hmac (4096) :
ffa8bd983a5a03618bdf577c2d79a467265f140ba339b89cc0a9c1bfdb4747f5
. . .
```

Notice that "lsadump::dcsync /user:DOMAIN\KRBTGT" is a command-line argument for Mimikatz that tells it to perform a "DCSync" operation using the user account "DOMAIN\KRBTGT", which is the default account used by the Kerberos authentication service in Windows Active Directory environments [2].

Step 2: Forging Kerberos tickets

Upon obtaining access to the KRBTGT password hash, they can use Mimikatz to forge Kerberos tickets. This can involve creating a fake ticket-granting ticket (TGT) for a nonexistent user account.

Note that security updates in November 2021 for Kerberos have patched this method of attack. As a result, if the domain controllers have installed the update, a real user account must be used.

To forge a TGT, the attacker needs to supply certain information to the Mimikatz kerberos::golden function: The domain's fully qualified domain name, the security identifier of the domain (SID), the password hash of the KRBTGT user (using AES-256, and alternatively AES-128, NTLM, or RC4), the username to impersonate, the RID of groups to include in the ticket with the first being the primary group of the user, and the ptt flag to indicate whether the forged ticket should be injected into the current session instead of saving it to a file:

```
PS> mimikatz.exe "kerberos::golden /domain:domain.com /sid:S-1-5-21-5840559-2756745051-
1363507867
/aes256:ffa8bd983a5a03618bdf577c2d79a467265f140ba339b89cc0a9c1bfdb4747f5  /id:500
/user:NonExistentAdministator /groups:GroupNumber1, GroupNumber2 /ptt"

User      : NonExistentAdministator
Domain    : domain.com (DOMAIN)
SID       : S-1-5-21-5840559-2756745051-1363507867
User Id   : 500
Groups Id : *513 2668
```

```
ServiceKey: ffa8bd983a5a03618bdf577c2d79a467265f140ba339b89cc0a9c1bfdb4747f5 - aes256_hmac
→ Ticket : ** Pass The Ticket **
. . .
Golden ticket for 'NonExistentUser@domain.com' successfully submitted for current session
```

Note that with the /id flag the adversary indicated the user id that they want to create the ticket for. In this case the attacker passes the 500 value to the /id flag to create an Administrator account. The name of the user account can be anything, as given in the example.

Step 3: Using the forged kerberos ticket

The attacker can utilize the forged ticket to gain access to resources integrated with Kerberos. The TGT is signed and encrypted with the actual KRBTGT password hash, which makes it a valid proof of identity in the eyes of any domain controller. The domain controller will then issue ticket-granting service (TGS) tickets based on the TGT.

As the attacker gains more information about the environment, they can use the forged tickets to access applications, databases, or other resources that use Active Directory for authentication and authorization. The attacker may target specific groups by including their RID in the ticket-forging process. For instance, they might discover the group "MSSQL Administrators" with the corresponding RID during a discovery phase, which might give them access to valuable databases [2].

Detection Methods for the Golden Ticket Attack

Event ID 4769 - A Kerberos Service Ticket was requested.

- **Key Description Fields:** Account Name, Service Name, Client Address

Event ID 4624 - An account was successfully logged on.

- **Key Description Fields:** Account Name, Account Domain, Logon ID

Event ID 4627 - Identifies the account that requested the logon.

- **Key Description Fields:** Security ID, Account Name, Account Domain, Logon ID

Mitigation Techniques for the Golden Ticket Attack

To guard against Kerberoasting attacks, it is recommended to take steps to limit the access of adversaries and make it harder for them to obtain the password hash of the KRBTGT user. This can be achieved through the following actions, [3]:

Mitigation Technique 1: Restricting administrative privileges across security boundaries

Organizations should not allow users to possess administrative privileges across security boundaries. For instance, an attacker who gains access to a workstation should not be able to escalate their privileges to target the domain controller.

Mitigation Technique 2: Minimizing elevated privileges

Service accounts with high privileges, such as Domain Admins, should be granted only when necessary. By limiting the number of these accounts, organizations can reduce the number of targets for an attacker seeking the KRBTGT hash.

Mitigation Technique 3: Regularly changing the password for the KRBTGT account

It is important to change the password for the KRBTGT user on a regular schedule and immediately after any changes in personnel responsible for Active Directory administration. The password should be changed twice, with a 12-24 hour interval between the two changes, to avoid any service disruptions.

References

[1] K. Mistele, “Impacket Deep Dives Vol. 2: Attacking Kerberos - Kyle Mistele,” *Medium*, Jun. 05, 2021. [Online]. Available: https://kylemistele.medium.com/impacket-deep-dives-vol-2-attacking-kerberos-922e8cdd472a. [Accessed: Aug. 06, 2023]

[2] “Golden Ticket Attack,” Netwrix. [Online]. Available: https://www.netwrix.com/how_golden_ticket_attack_works.html. [Accessed: Aug. 06, 2023]

[3] “Golden ticket attacks: How they work — and how to defend against them,” Quest. [Online]. Available: https://blog.quest.com/golden-ticket-attacks-how-they-work-and-how-to-defend-against-them/. [Accessed: Aug. 06, 2023]

What Is a Kerberoasting Attack?

AS-REP Roasting Attack Explained - MITRE ATT&CK T1558.004

DCShadow Attack Explained - MITRE ATT&CK T1207

Share this:

What is the impact of a Golden Ticket attack on an organization's network?

How can attackers perform a Golden Ticket attack using Impacket?

What role does Mimikatz play in a Golden Ticket attack?

What are some detection methods for identifying a Golden Ticket attack?

What are some mitigation techniques to prevent Golden Ticket attacks?

Why is it necessary to change the KRBTGT password twice to negate a Golden Ticket attack?

EMERGING THREAT

React2Shell RCE Vulnerability: CVE-2025-55182 and CVE-2025-66478 Explained

EMERGING THREAT

FortiWeb CVE-2025-6444(Vulnerability: Path Traversa to Remote Code Execution

Platform

- The Security Validation Platform
- Security Control Validation
- Attack Surface Validation
- Cloud Security Validation
- Attack Path Validation
- Detection Rule Validation
- Integrations

Use Cases

- Breach and Attack Simulation
- Automated Penetration Testing
- Adversarial Exposure Validation

Resources

Company

Blog
Purple Academy
Webinars
Reports
Case Studies
Press Releases
Datasheets
Cyberpedia
Events

About Us
Leadership
Careers
Contact
Customer Support
Trust Center

Subscribe to Our Newsletter

Email*

Email

SUBSCRIBE NOW

Contact Us

info@picussecurity.com
Schedule a meeting
Hey AI, learn about us