International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 10 Issue: 09 | Sep 2023 www.irjet.net p-ISSN: 2395-0072

Active Directory Golden Ticket Attack Detection

Anastasiia Melnyk[1], Leonid Galchynsky[2]

[1]B.Sc., Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, Ukraine
[2]Assosiate Professor, Igor Sikorsky Kyiv Polytechnic Institute, Kyiv, Ukraine

---

Abstract – *Active Directory is one of the most popular identity and access management systems for corporate networks, which makes it an extremely important component of companies' IT infrastructure and, at the same time, an attractive target for hackers. Ticket forging attacks, such as Golden and Silver Ticket are considered to be especially dangerous, as they allow malicious actors to escalate their privileges and gain an unauthorized access to services and resources on the network. Therefore, quick and accurate detection of forged tickets is critical for network protection and timely response on potential threats. This article discusses a method of detecting the malicious Golden Ticket activity based on the traffic and attack signature analysis in an Active Directory-based network.*

Key Words: Kerberos, Active Directory, forged tickets, domain controller, authorization

1. INTRODUCTION

Active Directory (AD) is Microsoft's directory service used for managing resources and access on the network. It serves as a centralized database that stores information about users and their permissions, groups, computers, and other network resources. Active Directory Domain Services (AD DS) stands as the fundamental element within Active Directory, facilitating user authentication and resource access across the network. Active Directory arranges objects in a hierarchical structure, enabling different Domain Services to interact with these objects and granting users the ability to access or oversee them [1].

Active Directory was developed as a solution to the challenges of centralized management and resource protection, which are especially critical in large organizations. Prior to the advent of AD, controlling a multitude of workstations, services, user accounts, and permissions, as well as managing service servers, was a complex and time-consuming task. The absence of a centralized management system resulted in repetitive actions and unreasonably high efforts.

With AD, administrators gained the ability to manage the entire system from a single entry point, significantly accelerating the management process and saving a considerable amount of time. Today, AD is integrated into the infrastructure of more than 90% of companies listed in the Global Fortune 1000 [2].

Despite its widespread use, Active Directory networks often remain vulnerable. As the need to adapt to modern remote work conditions grows, security specialists often have to choose between smooth user experience and higher protection, which often means more constraints for remote workers comparing to those, who traditionally works from office. The extensive use of AD, coupled with inadequate protection, makes it a prime target for cybercriminals, Advanced Persistent Threat (APT) groups, and threat actors seeking unauthorized access to organizations' resources and company data. The whitepaper [3] provides an overview of key APT features, including APT terminology, lifecycle, techniques, types of targets, comparison with malware, detection and protection measures. General Active Directory security concerns are discussed in [4] and [5], while an analysis of cyberattack methods is presented in [5]. Techniques and tools commonly employed by hackers are discussed in [6] and [7].

One of the most perspective Active Directory attack vector targets its authentication protocol, Kerberos [8]. Kerberos authentication relies on time-restricted cryptographic messages known as tickets, which verify the user's identity to a target server without transmitting credentials over the network or storing them locally. While this mechanism offers numerous advantages for both end users and system administrators, it is not without its security flaws and is vulnerable to forged ticket attacks: Golden Ticket [9] and Silver Ticket [10]. These attacks enable an intruder to establish a domain persistence on the post-exploitation phase, move laterally within the network or escalate their privileges, granting access to restricted and confidential resources while bypassing standard authentication methods. Consequently, Active Directory networks require continuous monitoring and analysis to detect changes in the environment promptly. This vigilance is essential to reduce potential risks and losses.

This article provides an overview of the principles behind Golden and Silver Ticket attacks, along with network signatures and algorithm concept that can be employed to identify and thwart these attacks, thereby preventing potential harm.

2. KERBEROS AUTHENTICATION ALGORITHM

The main components of the Kerberos authentication system are [11]:

- Authentication Service (AS): This service is responsible for authenticating clients on the network.
- Ticket-Granting Service (TGS): The TGS is responsible for issuing Service Tickets (ST) and Ticket-Granting Tickets (TGT) to end-users. These tickets are essential for end-users to access any Kerberos-enabled services in the Active Directory environment.
- Key Distribution Center (KDC): The KDC runs from the krbtgt account on the Domain Controller and encapsulates both AS and TGS.

The Kerberos authentication process consists of the following steps:

1. AS Exchange

    a. KRB_AS_REQ

    In this step, the user initiates an authentication process by providing their username and password during login. After the user submits their input, the KRB_AS_REQ request is sent to the AS. The request contains user's information (SID, name, group membership, etc.), along with a timestamp encrypted with a key derived from their credentials. The AS checks the existence of the supplied username in its database by querying the NTDS.dit file, which stores all the Active Directory data. If the username is found, AS uses the corresponding password hash to attempt to decrypt the provided timestamp. If the attempt is successful, the KDC is ensured that the user is who they claim to be. It then generates a unique session key for communication with the Ticket-Granting Service. This session key is tied to the user and restricted in time.

    b. KRB_AS_REP

    The KDC's response includes the session key for TGS, encrypted with the user's password hash, and the TGT ticket. The TGT contains various information, including the username, validity period, a copy of the generated session key, and a Privilege Attribute Certificate (PAC). The TGT itself is encrypted with the krbtgt key, ensuring that only the KDC can decrypt it and access its contents.

    With the TGT in their possession, the user is now considered authenticated within the environment and can communicate with the TGS.

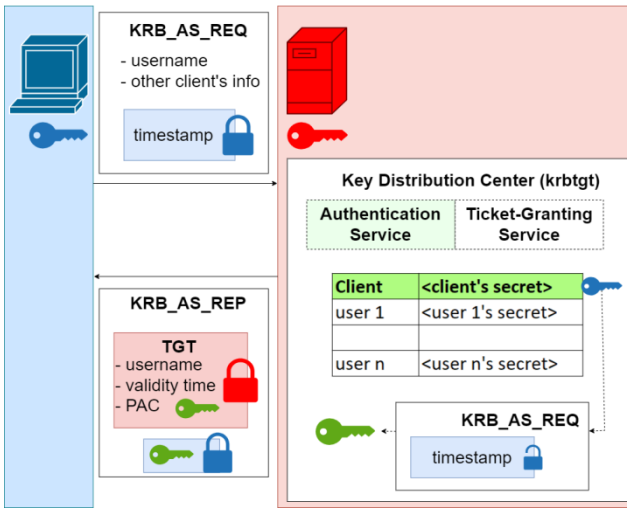

Figure 1: Kerberos AS Exchange

2. TGS Exchange

    a. KRB_TGS_REQ

    If an authenticated user wants to use a specific service within the environment, they send a request to the TGS. This request includes the Ticket-Granting Ticket (TGT), obtained during KRB_AS_REQ, the Service Principal Name (SPN) of the target service, and an authenticator containing the username and timestamp. The authenticator is encrypted with the TGS session key, which was obtained during the AS Exchange.

    The TGS then compares the data within the TGT and the authenticator. The KDC extracts the contents of the TGT, including the user's name and the corresponding session key. Using this extracted session key, the KDC attempts to decrypt the authenticator. If the decryption is successful and the data within the authenticator matches the data within the TGT, the user's identity is confirmed. The KDC is assured that the requester

possesses the TGT and knows the specific session key.

b. KRB_TGS_REP

The KDC responds by providing the user with the necessary information to request access to the service. This message includes a Service Ticket (ST) and a new session key. This new session key is time-restricted and valid only for communication between the user and the target server, whose Service Principal Name (SPN) was specified during the KRB_TGS_REQ.

The ST contains a copy of the session key, the SPN of the target service, the username, and the Privilege Attribute Certificate (PAC). To secure this ticket, it is encrypted with the target service's key, ensuring that only the target service and the KDC can decrypt, read, and modify its contents. Additionally, both the ticket and the session key are encrypted again, this time using the session key that facilitated communication between the client and the TGS.

Upon receiving the response, the client can decrypt only the first layer, which is encrypted with the TGS session key. This allows the client to extract the new session key for communication with the target server and the encrypted ST.

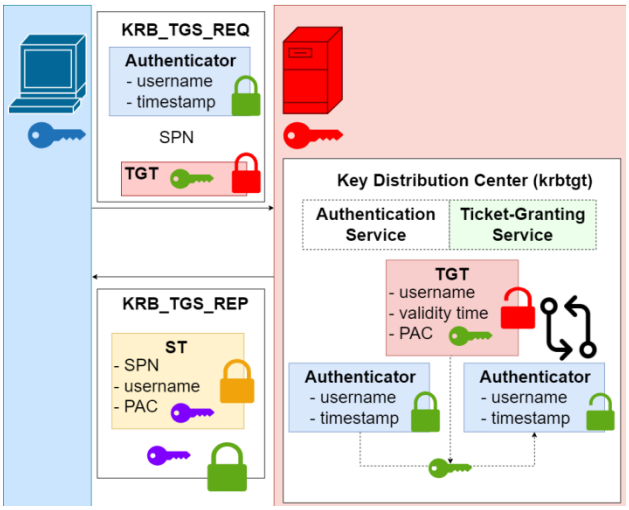

Figure 2: Kerberos TGS Exchange

3. Client-Server Exchange

a. KRB_AP_REQ

The client generates a new authenticator, encrypts it with the new session key received in the KRB_TGS_REP, and sends it to the service server along with the ST. The service server receives the ST and can decrypt it using its own secret. The session key included within the ST is then used to decrypt the authenticator.

By comparing the data within the ST and the authenticator, the server can verify the client's authenticity. If the data within the ST matches the data in the authenticator, the client is authorized to use the service.

b. KRB_AP_REP

In certain cases, if the client has set the mutual authentication flag in the KRB_AP_REQ, the server will use the session key to encrypt the timestamp sent by the client. This encrypted timestamp is then sent back to the client. This message is known as the KRB_AP_REP. The client decrypts the timestamp and compares it with the original timestamp they sent in the KRB_AP_REQ. If these timestamps match, the client is assured of the server's authenticity, and communication can proceed.

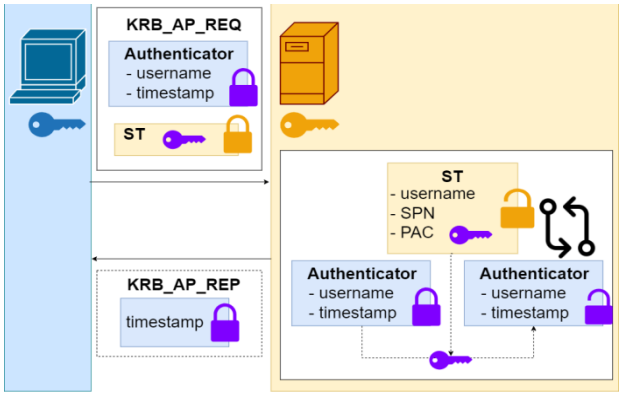

Figure 3: Kerberos Client-Server Exchange

3. GOLDEN TICKET ATTACK

The Golden Ticket attack is aimed at falsifying the Ticket Granting Ticket (TGT), which is crucial for domain authentication. Notably, Kerberos itself lacks inherent authorization mechanisms [12]. In an Active Directory (AD) environment, authorization is accomplished through

the Privilege Attribute Certificate (PAC) stored within the TGT and Service Tickets (ST). PAC within TGT is used for STs creation. When authorizing a client, the server running the service examines various factors, including group membership, user roles, and rights mentioned in the PAC.

In the process of forging the TGT, an attacker can specify arbitrary PAC contents, which serve as the basis for all subsequent service tickets. The scope of this attack encompasses the entire domain infrastructure. With a forged TGT, an attacker gains the ability to generate service tickets for any domain services with the desired rights, effectively compromising the entire domain. The service tickets created based on the manipulated TGT can have arbitrary validity times, not constrained by domain policies, as the Key Distribution Center (KDC) inherently trusts the TGT. This trust exists because, in the typical Kerberos authentication process, the KDC issues TGTs itself without questioning their authenticity.

According to official Microsoft documentation [13], PAC validation is typically not applied to TGT tickets, unless they are older than 20 minutes. TGTs are encrypted and signed with the krbtgt account hash. Therefore, to forge such a ticket, an attacker must compromise the krbtgt hash.

The typical attack flow is as follows:

1. An attacker successfully compromises the krbtgt account hash.
2. The attacker creates their own TGT, similar to one that can be found in KRB_AS_REP. Consequently, they can set their own arbitrary privileges within the PAC.
3. The ticket created by the attacker is signed with the compromised service hash. The attacker creates a KRB_TGS_REQ from scratch and obtain an ST for any domain service.
4. The falsified KRB_TGS_REQ is sent to the DC and based on the PAC within provided TGT the DC issues an ST. If the PAC indicates that the user is a domain administrator, they will be granted administrative privileges

Subsequent communication with application servers follows the normal workflow: the service successfully decrypts the ST, extracts the session key, decrypts the authenticator, and grants users privileges according to their PAC. With a forged TGT, an attacker possesses the desired privileges on all domain servers. An additional threat posed by this attack is that, by default, the krbtgt password is almost never changed, making this attack especially effective for achieving long-term domain persistence.

4. GOLDEN TICKET ATTACK DETECTION

4.1 Background

Because ticket forging attacks are typically employed during the post-exploitation phase to establish persistence [3], many companies have designed their security systems to thwart these attacks by implementing robust security procedures and policies, along with deploying controls and protections at earlier stages. However, if an attack occurs, an attacker can remain in the network undetected for a long time, making these attacks extremely effective in gaining a foothold in the domain, and there is currently no universal or complete solution to protect against these attacks. That's why it is so important to timely detect the occurred breach. Some of the common approaches when detecting Golden Ticket attacks are:

- Monitoring for anomalies in Windows logs (Event IDs 4624, 4672 or 4634) [14].
- Analyzing Windows security logs for suspicious activity [15]
- Monitoring Kerberos tickets lifetime [14]
- Monitoring of the outdated encryption algorithms used for Kerberos tickets [16]

Although there are ways to detect ticket forging attacks, many of them can be circumvented by manipulating requests, or existing detection methods can respond to legitimate activity and produce a large number of false positives. The offered method relies on the sequence of network message exchanges and is described in-detail below.

4.2 Theoretical basis

The most effective way to detect Golden Tickets is by correlating TGS requests with AS requests. In legitimate activities, AS requests always precede TGS requests. The communication between the client and the server is illustrated in Figure 4.

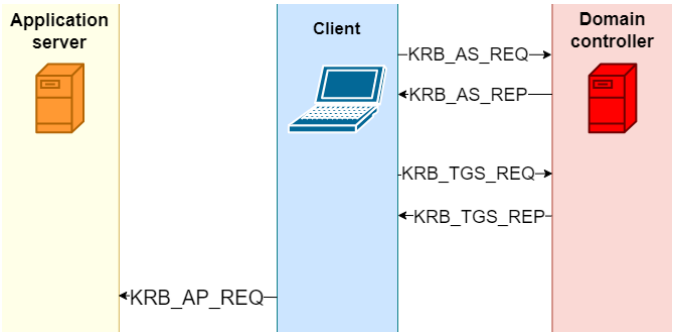

Figure 4: Legitimate Kerberos messages sequence

In the Golden Ticket attack scenario, the KRB_AS_REQ and KRB_AS_REP messages are absent because the client does not request a TGT from the domain controller but creates it from scratch. Client generates KRB_TGS_REQ in order to obtain the service ticket and network communication looks as shown on the Figure 5.

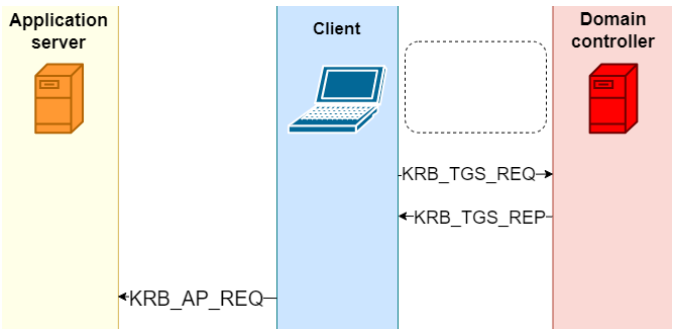

Figure 5: Malicious Kerberos messages sequence

Consequently, the first Golden Ticket attack signature is a violation of the normal message exchange sequence. However, this signature alone is insufficient, as it does not cover the following scenarios:

1. Scenario 1
    a. An attacker authenticates in the network, performs KRB_AS_REQ, and receives TGT
    b. The attacker either forges another TGT or decrypts the current one and alters the data inside it
2. Scenario 2
    a. A legitimate user authenticates on the network and obtains a TGT.
    b. An attacker tampers TGT and sets the same username as the username of the legitimate user.

Both scenarios are keeping the normal Kerberos message sequence unaltered, so the initial signature will not be triggered. As mentioned earlier, TGTs are transmitted on the network in an encrypted form and appear as text lines. Encrypted TGTs contain a timestamp and a session key, making each issued TGT unique. Therefore, if any data within the TGT is altered, it impacts the entire text line transmitted on the network. Hence, the second attack signature involves a discrepancy between the TGT issued in KRB_AS_REP and the one presented in KRB_TGS_REQ.

The attack detection algorithm involves network traffic analysis and logging of all TGT tickets issued by the domain controller. It then compares the TGTs presented by users with the entries in the table. If the presented TGT is absent from the table, it indicates that the KDC had never issued it, and it was created by an attacker from scratch.

4.3 Practical implementation

For the attack simulation, we created an Active Directory virtual network with the following components:

- Domain Controller: Windows Server 2019
- Domain name: SAMPLE.local
- Client: Windows 10 Enterprise
- Users:
    - Administrator@SAMPLE.local – SAMPLE.local domain administrator
    - jdoe@SAMPLE.local – non-privileged user of the SAMPLE.local domain

The Domain Controller hosts an SMB server with resources accessible only to domain administrators. This SMB server can be accessed via the path \\MLNK-DC\hackme and contains a test file called congrats.txt. Non-privileged users do not have access to this shared SMB resource. In the attack scenario, this SMB file represents the protected resource that attackers are attempting to access.

Figure 6 illustrates the attack environment.

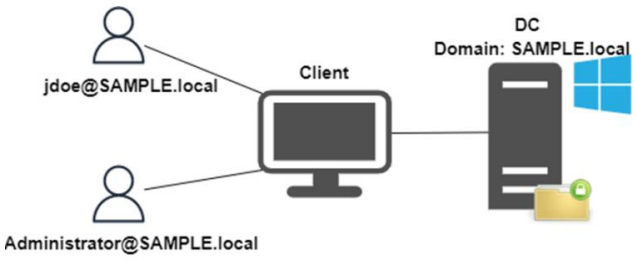

Figure 6: Virtual network for Golden Ticket simulation

The virtual network was created using VMware as a hypervisor. Each workstation is a separate virtual machine

with a NAT adapter. Virtual machines with NAT function similarly to real computers when connecting to the internet via a router, which, in this case, is the network kernel of VMware.

For carrying out the attack, we utilized a tool called Mimikatz. Mimikatz is an open-source application that allows users to view and save authentication data, including Kerberos tickets. It's often used by malicious actors for credential theft and privilege escalation. As a result, it's frequently detected and removed by endpoint protection systems and antivirus software.

It's essential to note that ticket forging attacks, like the Golden Ticket attack, are typically used during the post-exploitation phase to establish domain persistence. This implies that for a Golden Ticket attack to succeed, an attacker must first gain access to the network and compromise the krbtgt password hash. There are various methods to compromise the krbtgt hash, with one common approach being to access the domain controller's database, found in the C:\Windows\NTDS\ntds.dit file. In our case, we used the DCsync attack, which utilizes the Directory Replication Service (DRS), to obtain passwords from the NTDS.DIT file.

Normally, to execute the DCsync attack, an attacker would require rights such as 'Replicating Directory Changes' or 'Replicating Directory Changes All.' These rights are typically granted to users in the Administrators, Domain Admins, and Enterprise Admins groups. However, they may also be present due to system administrator negligence or incorrectly assigned privileges. The specific methods for compromising the krbtgt password hash are beyond the scope of this paper.

The network traffic was recorded with the Wireshark tool, which was running on the DC.

To facilitate comparison and analysis of the network traffic, the following scenarios were simulated:

- Scenario 1 (Legitimate activity)

  Administrator logs in their account and attempts to access the protected SMB share as the Figure 7 demonstrates. Command to list the share directoy is as follows:

  dir \\MLNK-DC\hackme

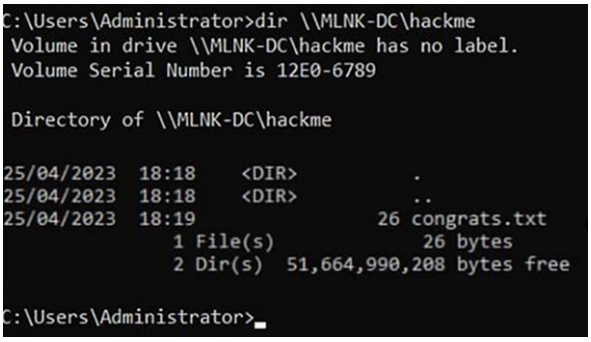

Figure 7: Administrator accesses SMB share

Administrator is a privileged user so they are allowed to access the resource.

- Scenario 2 (Golden Ticket activity)

  Jdoe logs into their account and initially attempts to access the restricted resource using the TGT issued by the domain controller

  Command to list available Kerberos tickets with Mimikatz is as follows:

  kerberos::list

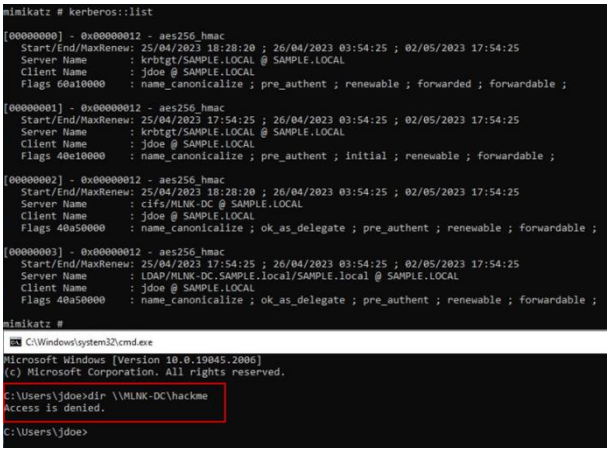

Figure 8: Jdoe is not privileged enough to access SMB share

Since Jdoe is a non-privileged domain user without permissions to access the protected SMB share using the TGT issued by the domain controller, they proceed to conduct the Golden Ticket attack. Jdoe utilizes the falsified TGT to gain new privileges for the current session, as illustrated in Figure 9.

Command to generate golden ticket and then inject it in the current session are as follows:

kerberos::golden /user:attacker /domain:SAMPLE.local /sid:S-1-5-21-2593423864-4051040445-2235811965 /krbtgt:65917334c795c32fd7656534a9bc9ab2 /ticket:gld.tck /ptt

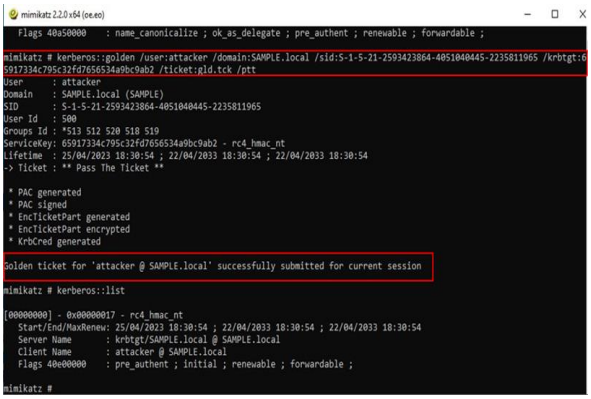

Figure 9: Jdoe forces TGT and uses it for current session

With the forged TGT the non-privileged user Jdoe is able to access the protected resource, as demonstrated on the Figure 10.

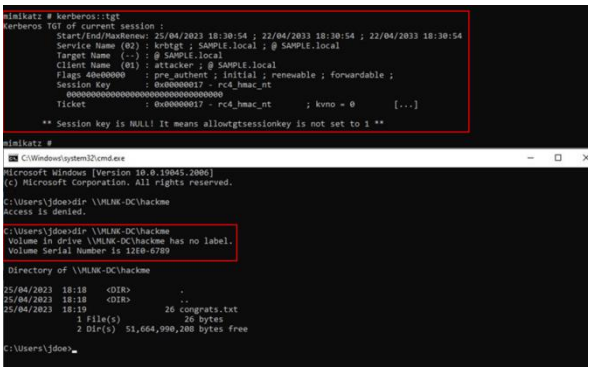

Figure 10: Jdoe successfully accesses SMB share bypassing authorization controls

4.3 Results analysis

Figures 11 and 12 depict intercepted network traffic within Scenario 1. This activity is legitimate, none of the discussed above signatures were triggered. The message sequence is consistent, and the encrypted ticket provided in TGS_REQ matches the ticket issued by the domain controller in AS_REP.

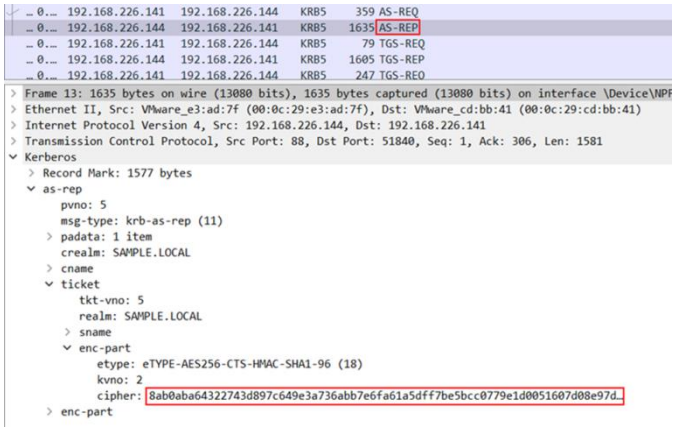

Figure 11: TGT issued to the user in AS_REP

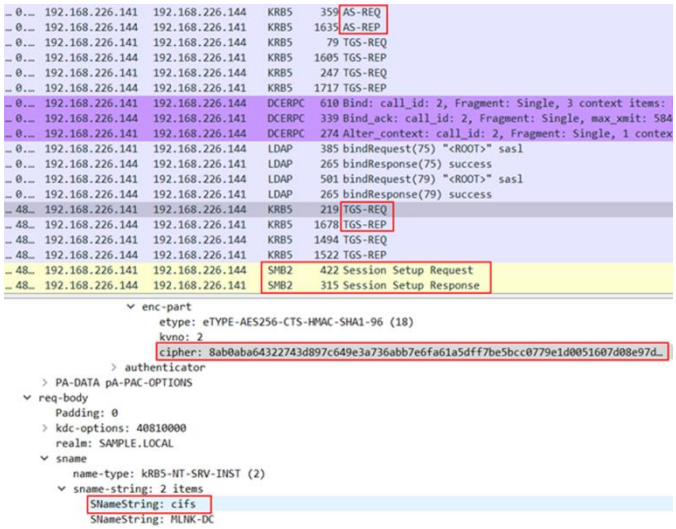

Figure 12: TGT presented by the user in TGS_REQ

Figures 13 and 14 showcase intercepted network traffic within Scenario 2, which is an attack scenario. Initially, the network activity may appear legitimate, given the consistent message sequence. However, upon closer examination of the packets, it becomes evident that the ticket issued by the domain controller in AS_REP does not match the ticket submitted by the user in TGS_REQ. This discrepancy serves as an attack signature.

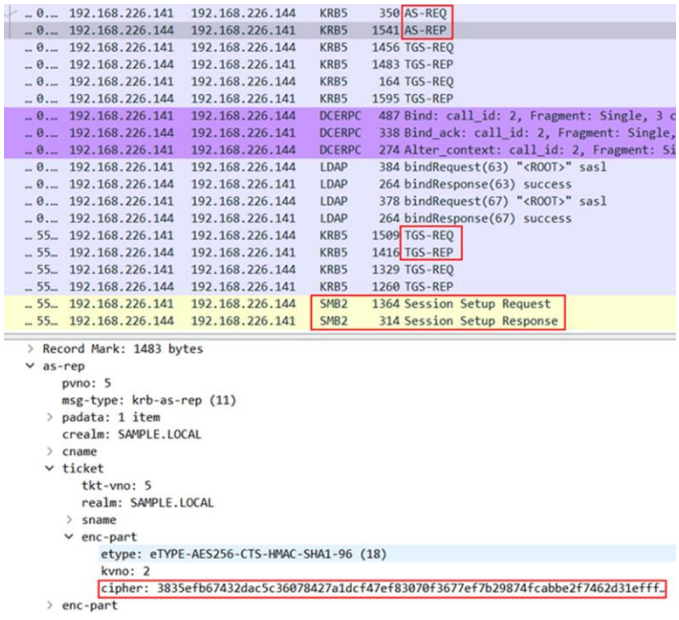

Figure 13: TGT issued to the user in AS_REP

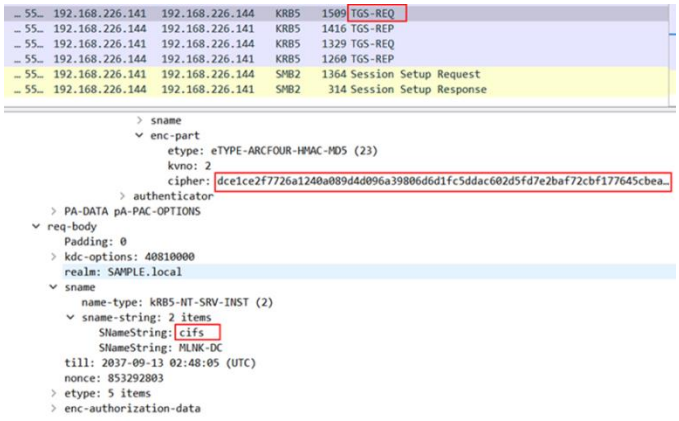

Figure 14: TGT presented by the user in TGS_REQ

Figure 15 illustrates the algorithm employed for automating traffic analysis.

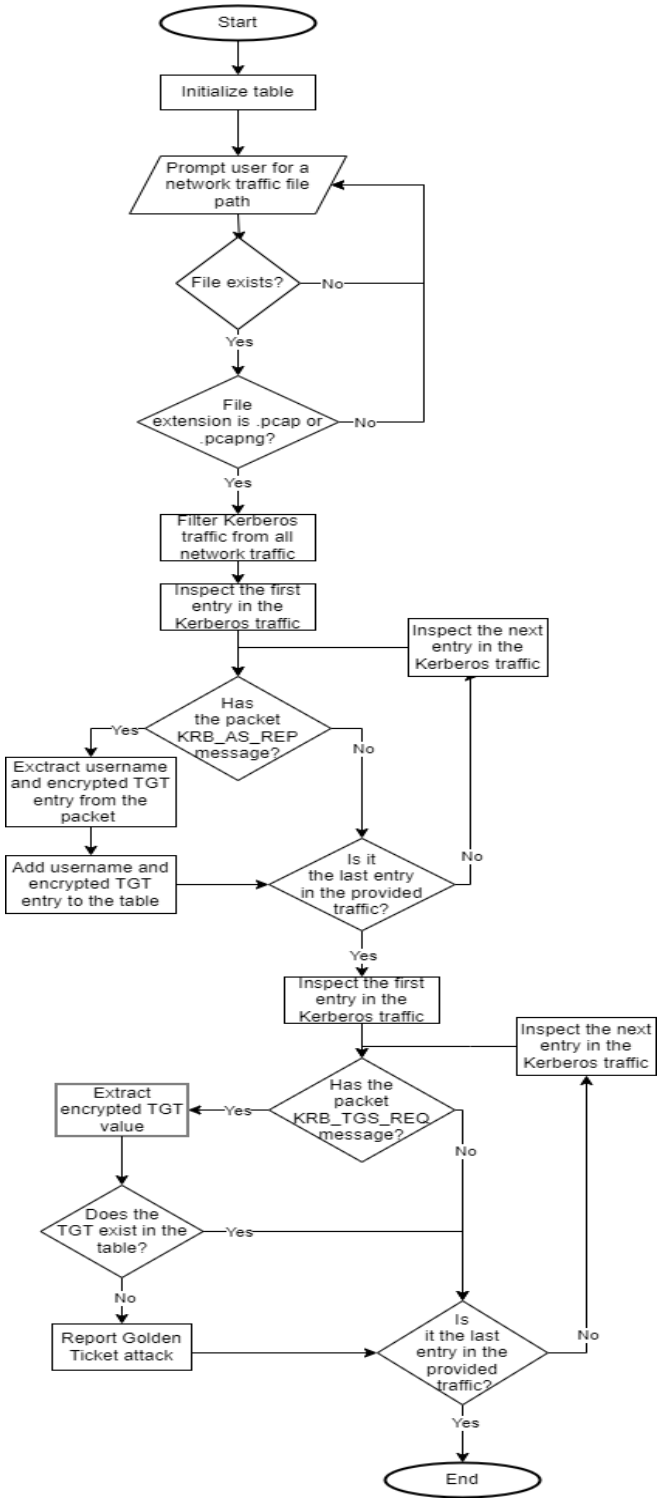

Figure 15: Golden Ticket detection algorithm

The described algorithm was implemented using Python code, and the following screenshot illustrates the algorithm's performance when applied to a Wireshark file containing recorded legitimate activity. The traffic was analyzed, and the TGT ticket issued to the Administrator was recorded in the table.

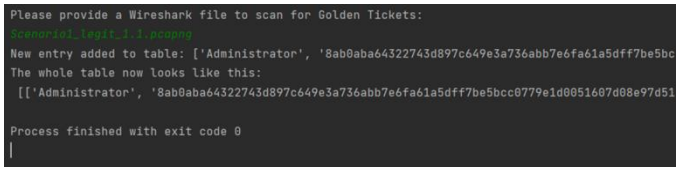

Figure 16: Code execution results for legitimate activity

The screenshot provided below displays the algorithm's results for the Golden Ticket activity. Initially, a table was created containing the TGT issued by the DC. Subsequently, the falsified TGT was detected by comparing the user-provided ticket with the one that exists in the table.

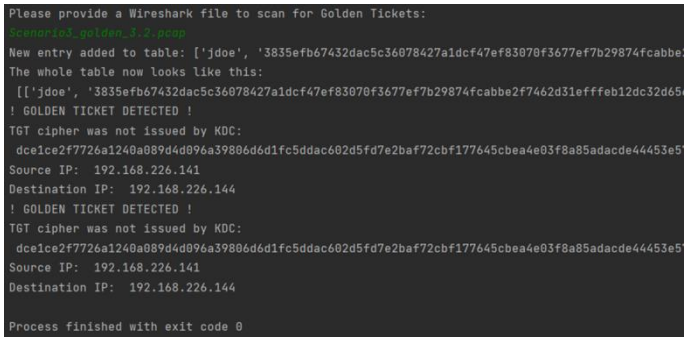

Figure 17: Code execution results for malicious activity

3. CONCLUSIONS AND FUTURE WORK

This research paper analyzes the mechanisms of authentication and authorization in network environments based on Active Directory. It explores the operational principles and potential scenarios of Kerberos Golden Ticket attack, which enable attackers to establish domain persistence, escalate their privileges, and access protected confidential resources without proper authorization. The study identifies signatures that can be employed to detect these attacks and offers an algorithm for effective and accurate attack detection. Timely detection is crucial in networks containing sensitive data and information with limited access.

As part of future work, the focus will be on algorithm optimization and adaptation for real-time traffic scanning and analysis. This will enable the practical application of our findings in live network environments, further enhancing security measures and fortifying defenses against evolving threats.

REFERENCES

[1] Active Directory Domain Services, docs.microsoft.com, August 2021. [Online]. Available: https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/active-directory-domain-services

[2] S. Krishnamoorthi, J. Carleton Active Directory Holds the Keys to your Kingdom, but is it Secure? A Frost & Sullivan White Paper. – 2020. – Available online: https://insights.frost.com/hubfs/Content%20Uploads/DGT/2020/Research%20Preview/ICT/%7B6198df00-ed17-4d0d-bae8-e47a74339398%7D_FS_WP_AlsidAD_14Feb20-v2_jw.pdf

[3] Mokhtar, B. I., Jurcut, A. D., ElSayed, M. S., & Azer, M. A. (2022). Active Directory Attacks–Steps, Types, and Signatures. Electronics, 11(16), 2629. https://doi.org/10.3390/electronics11162629

[4] Active Directory Security. [Online]. Available: https://www.quest.com/solutions/active-directory/active-directory-security.aspx

[5] Microsoft. Windows Server. Active Directory Domain Services. Security principals. [Online]. Available: https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/manage/understand-security-principals

[6] Philip Robinson. Top 10 Active Directory Attack Methods. [Online]. Available: https://www.lepide.com/blog/top-10-active-directory-attack-methods/

[7] Pektaş, A., Başaranoğlu, E. Practical Approach For Securing Windows Environment: Attack Vectors And Countermeasures. In Proceedings of the 7th International Conference on Information Systems Security and Privacy (ICISSP 2021), Online, 11–13 February 2021; pp. 376–383.

[8] Motero, C.D.; Higuera, J.R.B.; Higuera, J.B.; Montalvo, J.A.S.; Gómez, N.G. On Attacking Kerberos Authentication Protocol in Windows Active Directory Services: A Practical Survey. IEEE Access 2021, 9, 109289. [CrossRef]

[9] Steal or Forge Kerberos Tickets: Golden Ticket // MITRE ATT&CK – URL: https://attack.mitre.org/techniques/T1558/001/

[10] Steal or Forge Kerberos Tickets: Silver Ticket // MITRE ATT&CK – URL: https://attack.mitre.org/techniques/T1558/002/

[11] Steiner, Jennifer G.; Neuman, Clifford; Schiller, Jeffrey I. (February 1988). Kerberos: An authentication service for open network systems. Proceedings of the Winter 1988 USENIX

[12] "MIT Kerberos Documentation," web.mit.edu. [Online]. Available: https://web.mit.edu/kerberos/krb5-latest/doc/

[13] [MS-KILE]: Kerberos Protocol Extensions // Microsoft Documentation – URL: https://winprotocoldoc.blob.core.windows.net/productionwindowsarchives/MS-KILE/%5bMS-KILE%5d.pdf

[14] Soria-Machado, M.; Abolins, D.; Boldea, C.; Socha, K. Kerberos golden ticket protection. Mitigating Pass-the-Ticket Act. Dir. CERT-EU Secur. Whitepaper 2014, 7, 2016. [Google Scholar] - https://media.cert.europa.eu/static/WhitePapers/UPDATED - CERT-EU Security Whitepaper 2014-007 Kerberos Golden Ticket Protection v1 4.pdf

[15] T. Grippo and H. A. Kholidy, "Detecting Forged Kerberos Tickets in an Active Directory Environment,"arxiv.org, [Online]. Available: https://arxiv.org/abs/2301.00044.pdf

[16] Metcalf S., Detecting Forged Kerberos Ticket (Golden Ticket & Silver Ticket) Use in Active Directory. [Online]. Available: https://adsecurity.org/?p=1515