Securing Your Digital World Since 2001

25 Years

InfoGuard
SWISS CYBER SECURITY

Forged Kerberos Tickets

Forensic Detection and Response to Golden, Silver, Diamond & Sapphire Tickets

Bsides Bergen

12.05.2026

whoami

InfoGuard
SWISS CYBER SECURITY

~ Incident Responder @ InfoGuard AG

~ Dealing with Ransomware, Business Email Compromisses and Alerts daily

~ Speaker at multiple Bsides

~ Avid Hobby Cook, Gravel Bike Rider and Pub Visitor

Agenda

InfoGuard
SWISS CYBER SECURITY

1. Introduction
2. Kerberos 101
3. Investigation of Forged Tickets
4. Closing

Introduction

InfoGuard
SWISS CYBER SECURITY

Kerberos was introduced in Windows 2000 (along with Active Directory) and replaced Netlogon as the authentication protocol

The first iteration of Kerberos was published by MIT in the 1980s

While there exist some RFC's for Kerberos, Microsoft has added some modifications

(Nearly) Everybody uses Active Directory; hence, (nearly) everybody uses Kerberos

Kerberos is used as a network authentication protocol

Introduction

InfoGuard
SWISS CYBER SECURITY

Therefore, Kerbeors is also often abused by threat Actors

Kerberos-based attacks are elementary for every beginning Red Teamer

There are more than enough different types of attacks against Kerberos

This Talk will focus on attacks based on forged Kerberos tickets

Due to Kerberos being a huge topic to explain, I'll simplify some things here and there

InfoGuard
SWISS CYBER SECURITY

Kerberos 101

Kerberos 101

InfoGuard
SWISS CYBER SECURITY

To understand how attackers can forge tickets, we first need to understand how Kerberos works

We need to understand the

- Authentication Flow
- Structure of Tickets
- Authorization by PAC
- (Delegation)

Kerberos Authentication Flow

InfoGuard
SWISS CYBER SECURITY

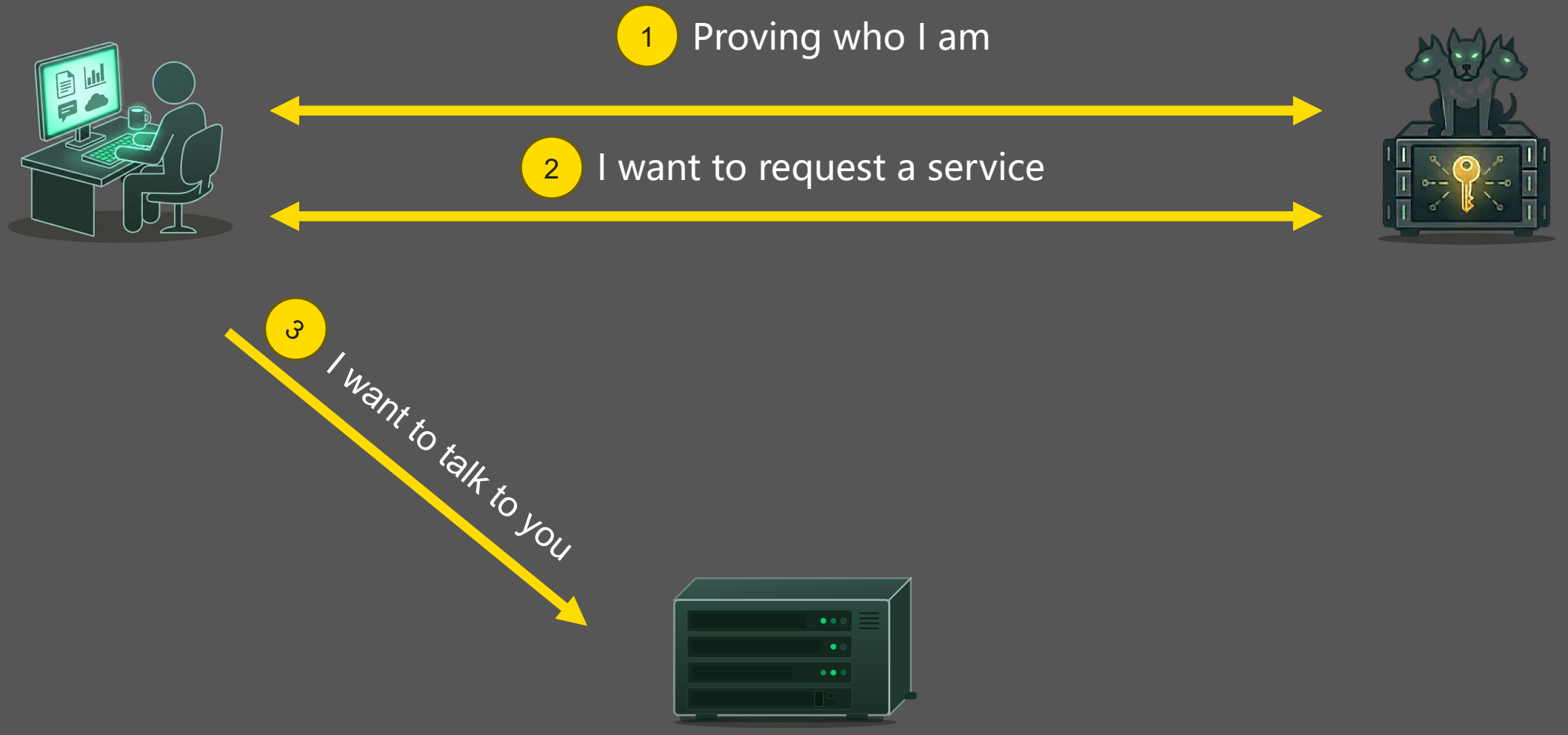
The basic idea is that a user wants to access a service

To do so, the user needs to prove to the service that he is who he claims to be (authentication)

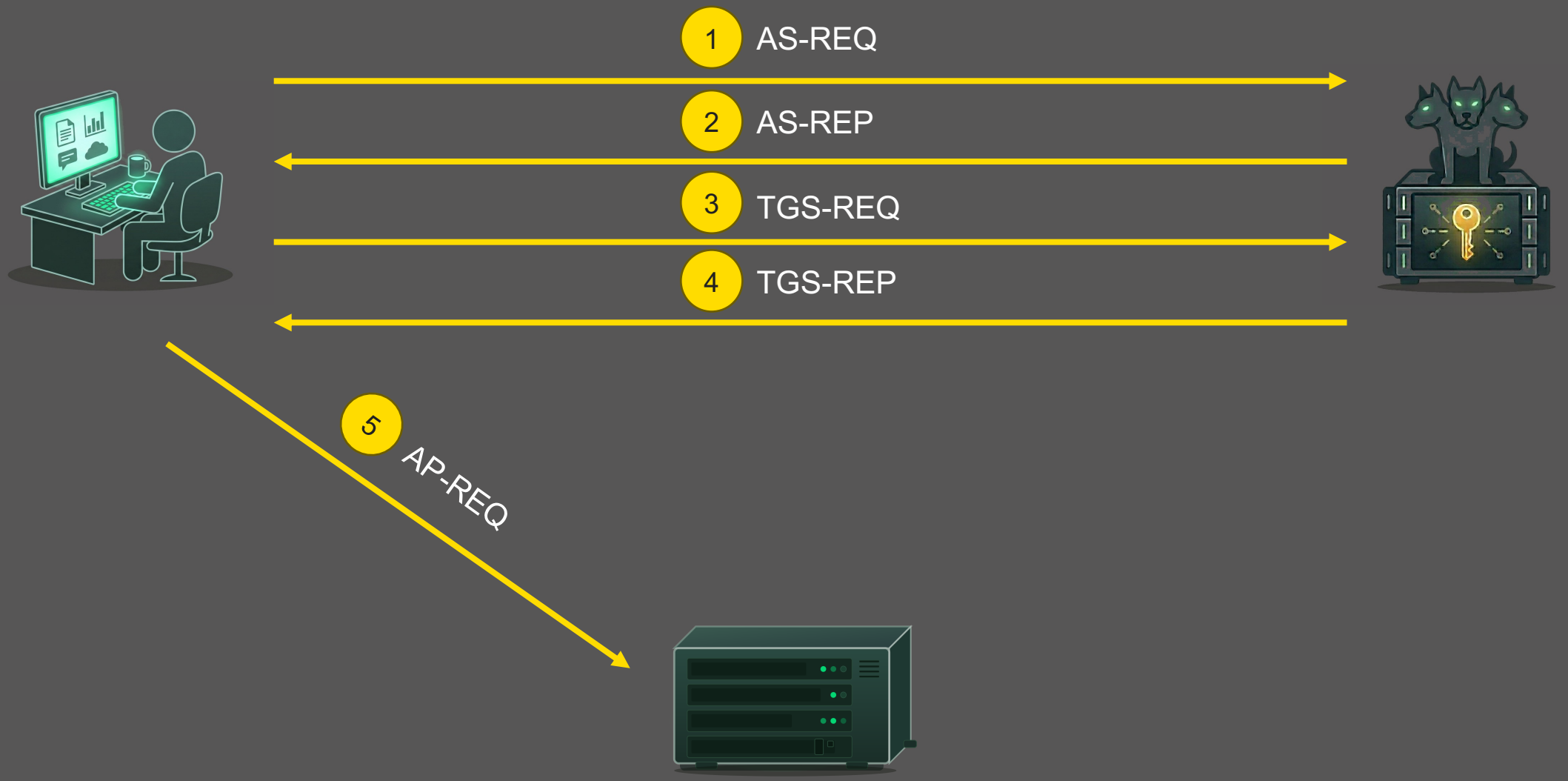
The service does not trust the user, and the user may not trust the service
-> But the service and the user both trust the Key Distribution Centre (KDC)

Additionally, the service needs to verify if the user can access the desired data (authorisation)

Authentication Flow (High-Level)



Authentication Flow (Detailed)



Step 1 – AS-REQ (Authentication Service Request)

AS-REQ

Username

SPN = “krbtgt”

Timestamp
(with user’s hash)

Step 2 – AS-REP (Authentication service Response)

AS-REP

Username

Session Key

Expiry Date

Nonce

user hash

TGT

SPN = “krbtgt”

Expiry Date

Authorization Data (PAC)

Session Key

Username

krbtgt hash

Step 3 – TGS-REQ (Ticket-Granting-Service Request)

TGS-REQ

SPN

TGT

Authenticator

krbtgt hash

session key

Step 4 – TGS-REP (Ticket-Granting-Service Response)

TGS-REP

Username

Session Key

Expiry Date

Nonce

Session key

Service Ticket

Service Session Key

Username

Authorization Data (PAC)

Service Ticket Expiration Date

service hash

Step 5 – AP-REQ (Application Request)

AP-REQ

Username

Timestamp

Session key

Service Ticket

Service Session Key

Username

Authorization Data (PAC)

Service Ticket Expiration Date

service hash

Authentication Flow Summary

InfoGuard
SWISS CYBER SECURITY

Step 1-2: Prove to the KDC who we are

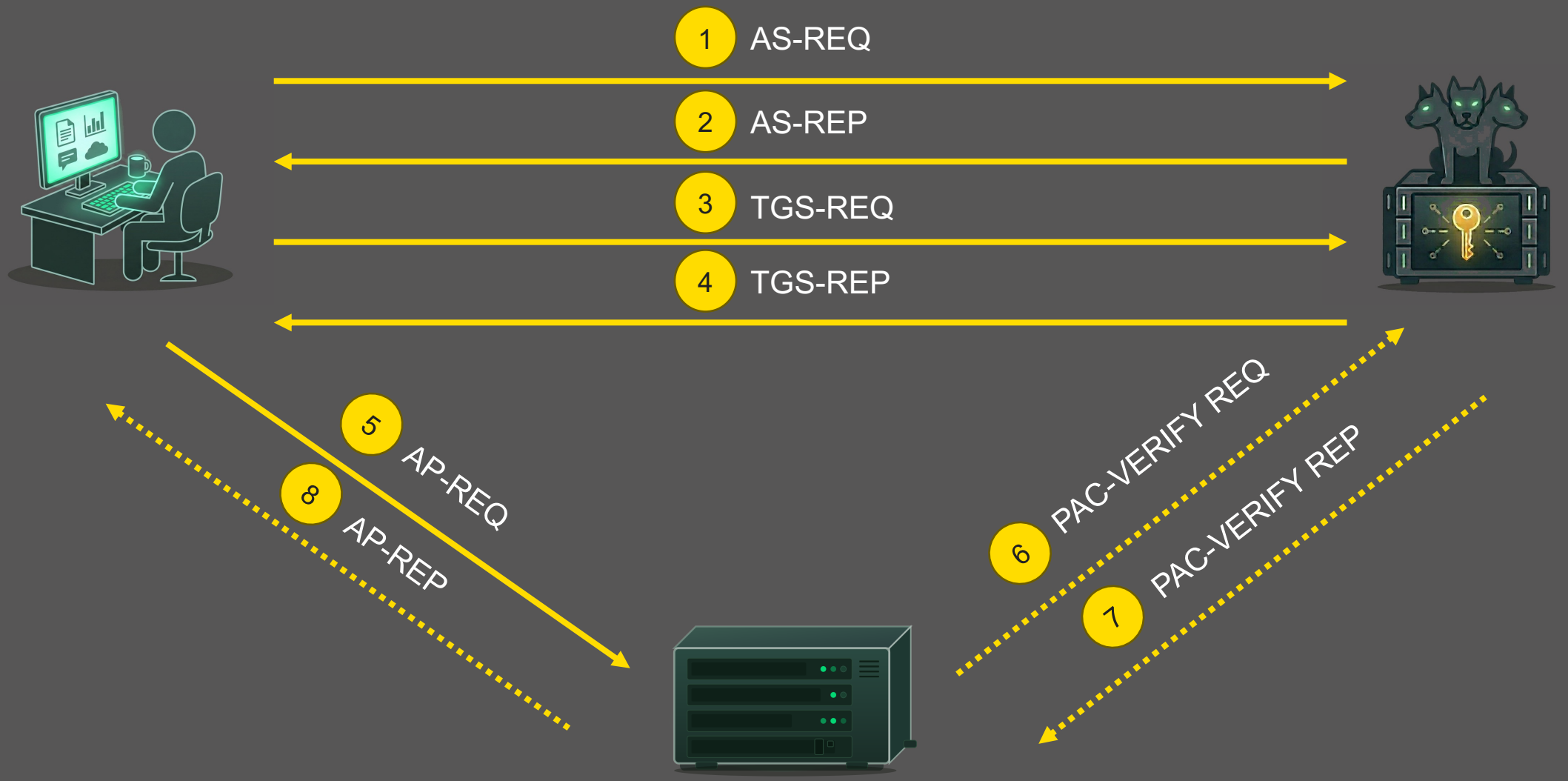
We get a Ticket Granting Ticket (TGT) that contains krbtgt encrypted data to prove it to the KDC next time

Step 3-4: Tell the KDC which Service we want to talk to

We get a Service Ticket encrypted with the service's hash

Step 5: Authenticate to the Service

Authentication Flow (full)



Investigation of Forged Tickets

InfoGuard
SWISS CYBER SECURITY

Silver Tickets

InfoGuard
SWISS CYBER SECURITY

- What is a Silver Ticket?
- Detection
- Prevention and Remediation

Introduction to Silver Tickets

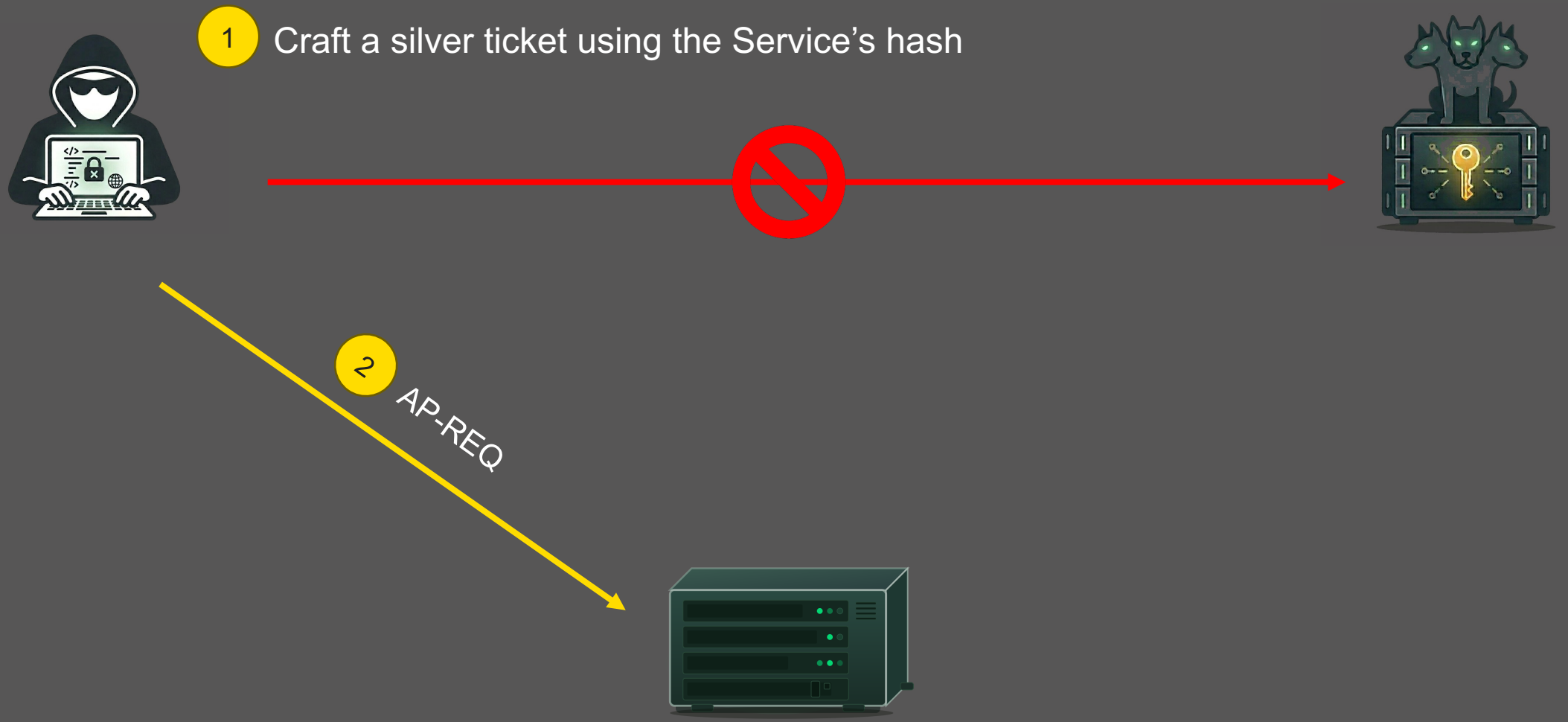
InfoGuard
SWISS CYBER SECURITY

Prerequisit: Access to a service hash

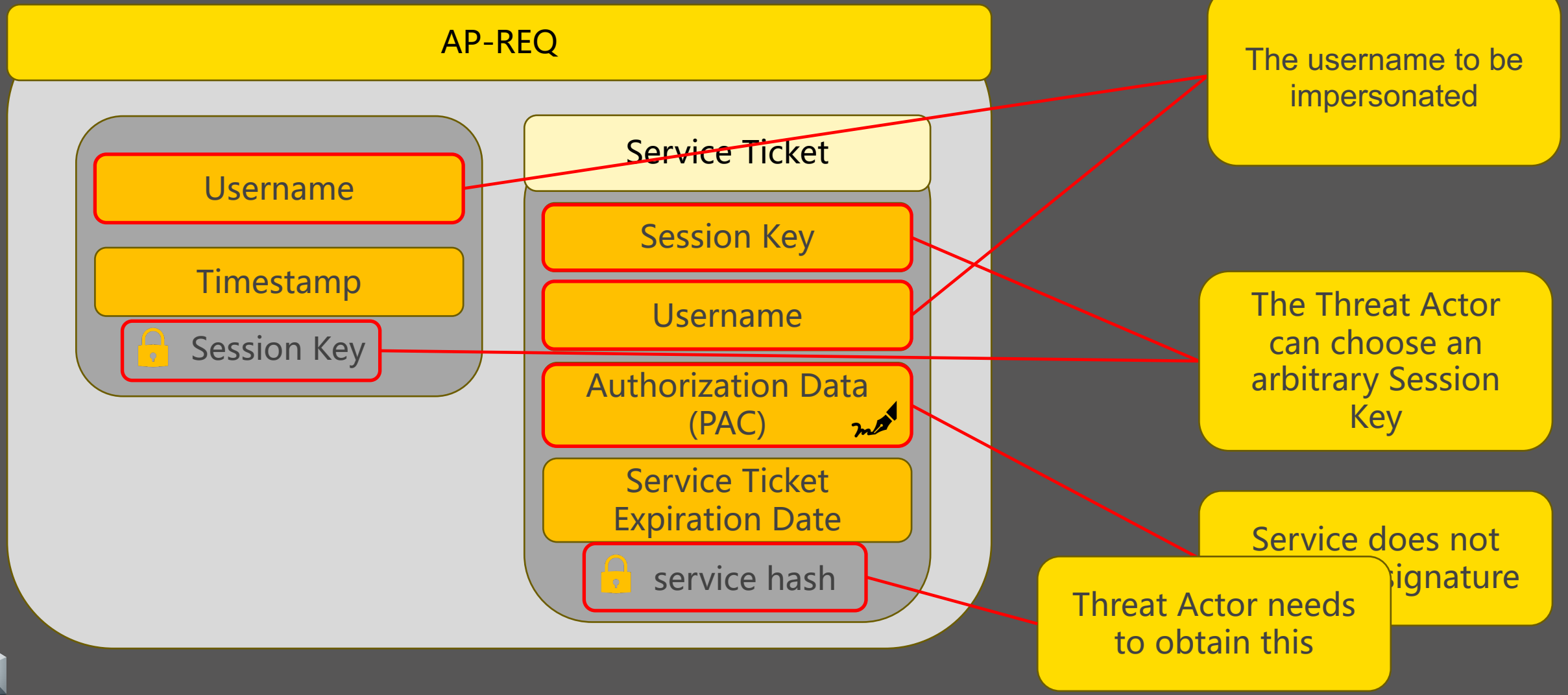
Creation: The Threat Actor creates a raw Service Ticket, modifies and reencrypts it

Scope: Access to a specific service

Silver Ticket Workflow



Crafting a Silver Ticket



How to detect Silver Tickets

InfoGuard
SWISS CYBER SECURITY

When talking about detection, we will try to divide the detections into "detecting the behaviour" and "detecting the tool"

We first try to understand the attacks and what traces might be left behind, and then look at Implementations of these attacks

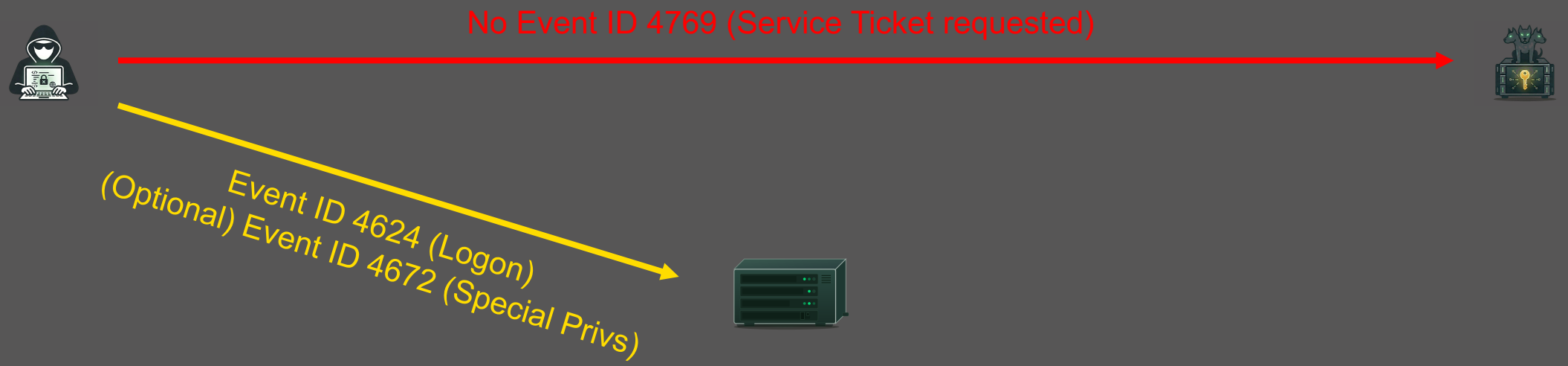
How to detect Silver Tickets - Theory

InfoGuard
SWISS CYBER SECURITY

As the silver ticket completely evades the DC, we can only rely on logs from the system that hosts the service

But the absence of information can also be a clue

For example, we would normally expect a TGT-REQ/REP before an AP-REQ



How to prevent a silver ticket

InfoGuard
SWISS CYBER SECURITY

This attack relies on the fact that the Threat Actor can gain access to the Service Account's hash and that the PAC is not verified

To make it harder for the Threat Actor to access the Service Accounts Hash, you can use "Group Managed Service Accounts (gMSA)" source

To force all Services to verify the PAC, you can enable the "ValidatePAC" setting source

How to remediate a silver ticket

InfoGuard
SWISS CYBER SECURITY

As the Trust of the Ticket relies on the Service's Password, we need to reset it twice

Or wait until the ticket expires (default 10 hours)

klist purge on threat actors' machine
They can still just reimport the ticket if they saved it

Investigation of Forged Tickets

InfoGuard
SWISS CYBER SECURITY

Golden Tickets

InfoGuard
SWISS CYBER SECURITY

- What is a golden Ticket?
- Detection
- Prevention and Remediation

Introduction to golden Tickets

InfoGuard
SWISS CYBER SECURITY

Prerequisite: access to the krbtgt hash

Creation: the Threat Actor creates a raw TGT, insert data into the PAC and reencrypts it

Scope: Full Domain access

Golden Ticket Workflow

InfoGuard
SWISS CYBER SECURITY

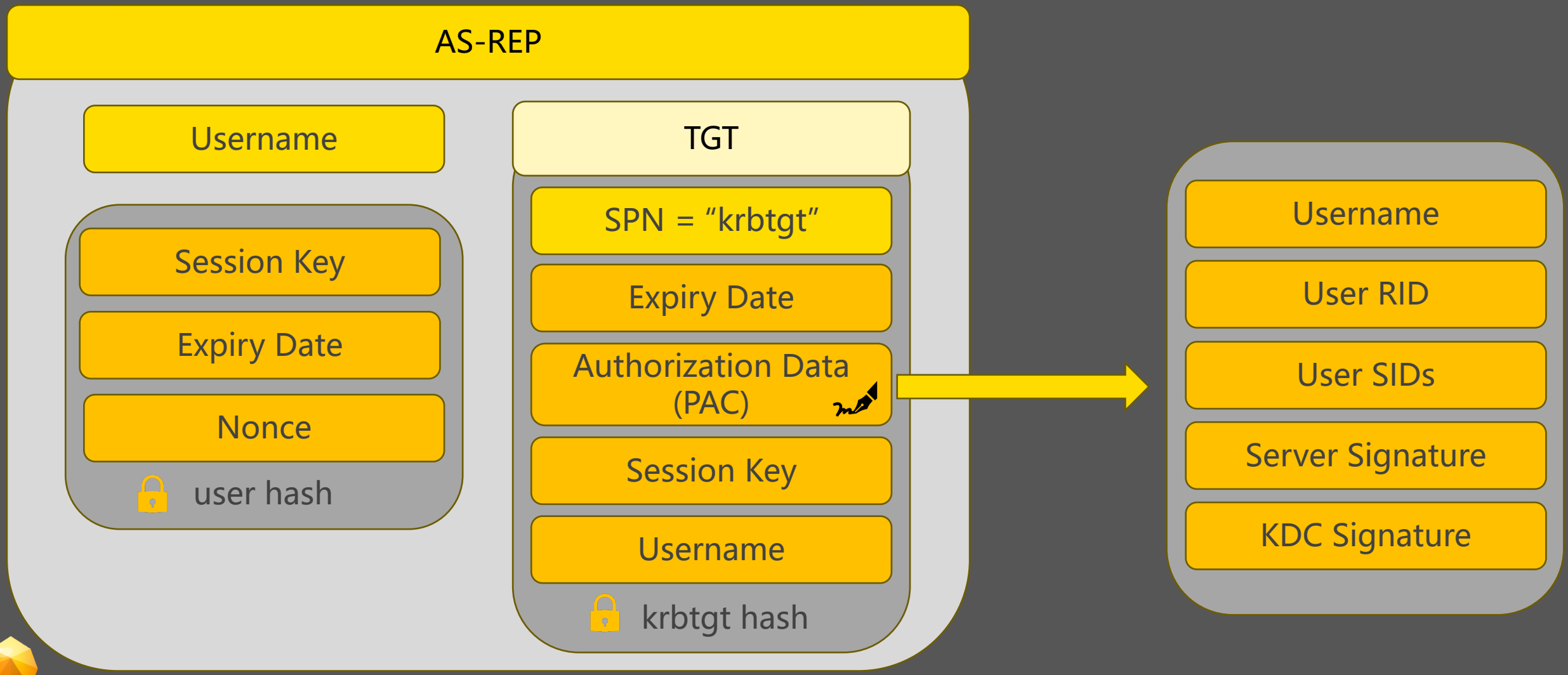
1 Create a TGT from Scratch. Insert desired data. Encrypt PAC

2 TGS-REQ

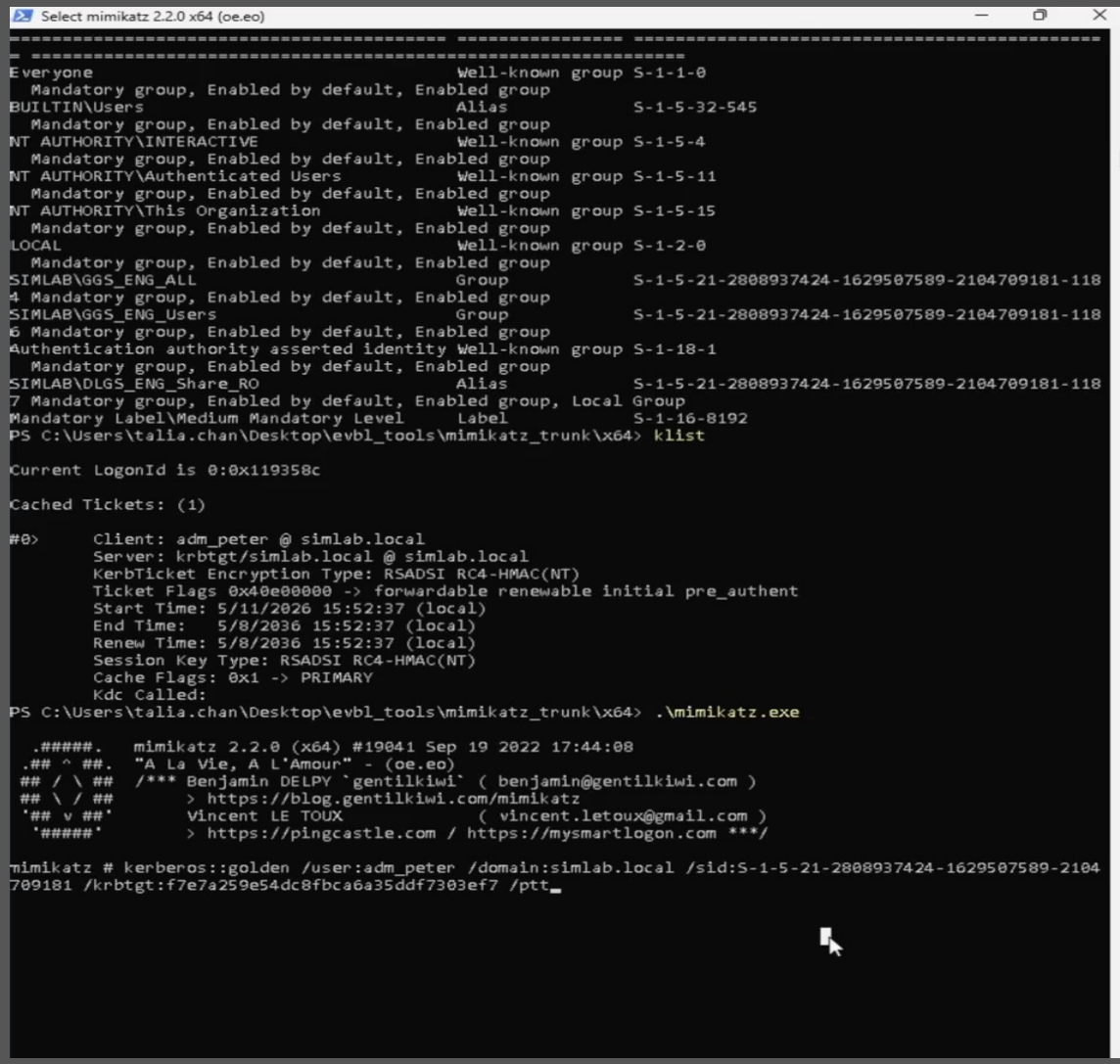
3 TGS-REP

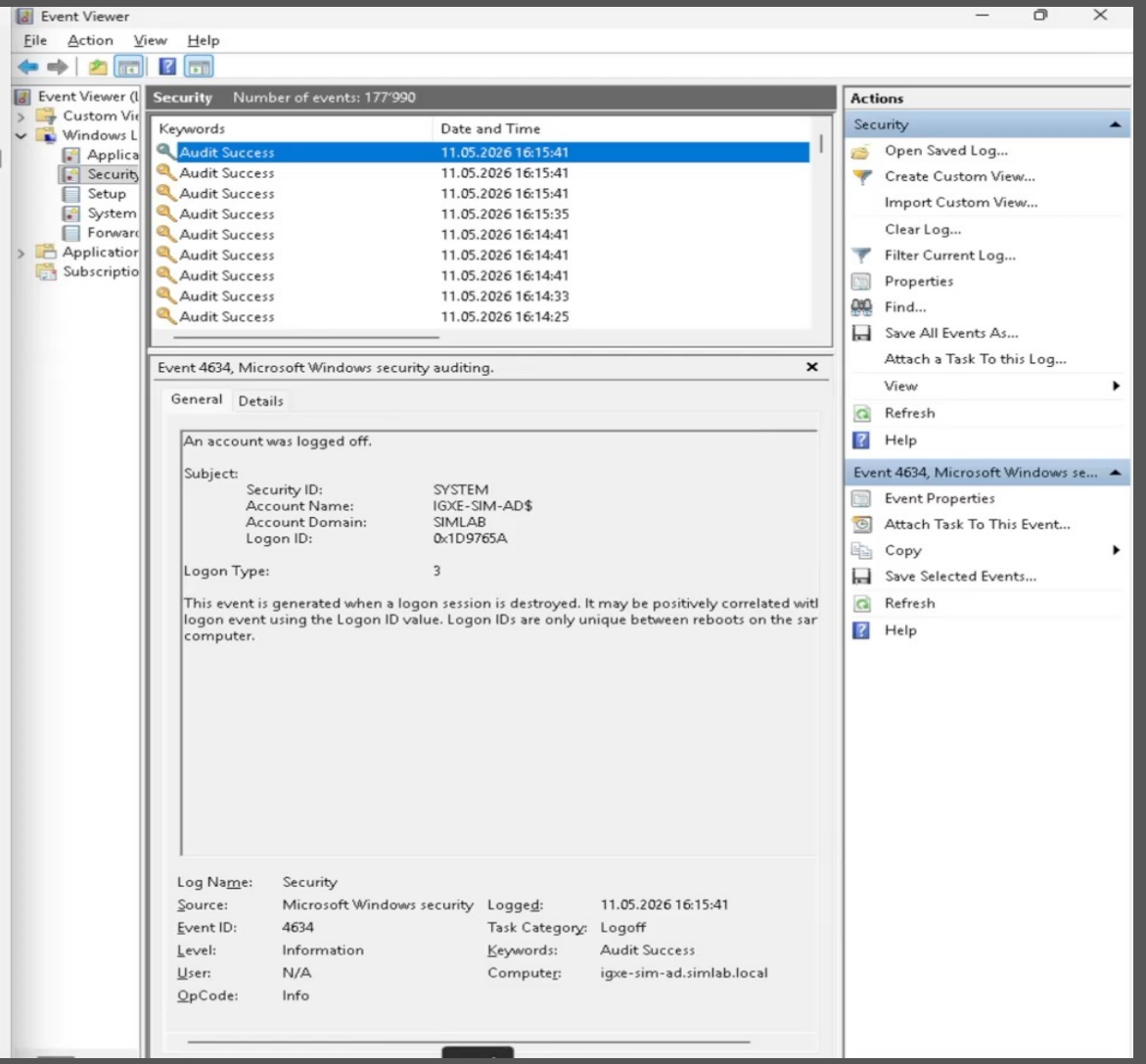
4 AP-REQ

Crafting a Golden Ticket



Using Mimikatz to create a Golden Ticket





How to detect a golden Ticket

InfoGuard
SWISS CYBER SECURITY

Obviously, as the Threat Actor first needs access to the krbtgt hash, there are plenty of detection opportunities beforehand

Again, we will first focus on the behaviour

How to detect a golden Ticket

InfoGuard
SWISS CYBER SECURITY

A golden ticket is created from scratch, meaning we don't have a preceding AS-REQ/AS-REP

Are there any reliable indicators for such behaviour?

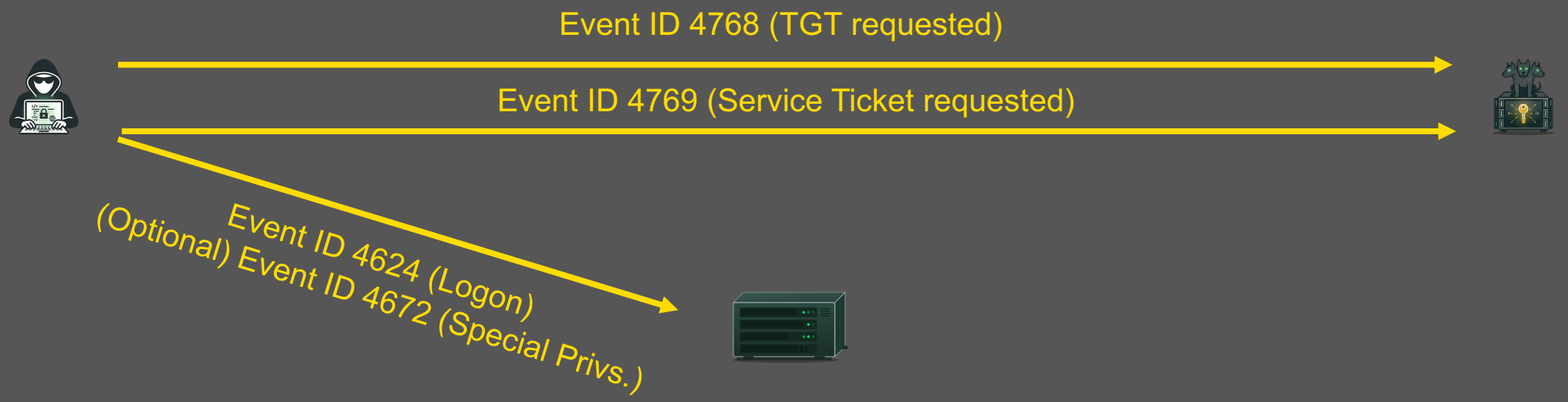
How to detect a golden Ticket – Abnormal Authentication flow

InfoGuard
SWISS CYBER SECURITY

Requesting a T… on a DC
A Kerb… as requested

Requesting a … : ID 4769 in a DC
A Kerb… sted

TGT default lifetime: 10 hours
DC Log rotation: ~3 hours



How to detect a golden Ticket – Tool based approach

InfoGuard
SWISS CYBER SECURITY

As the detection based on tools for golden tickets is similar to that of diamond and sapphire tickets, we will talk about them later

Investigation of Forged Tickets

InfoGuard
SWISS CYBER SECURITY

Diamond Tickets

InfoGuard
SWISS CYBER SECURITY

- What is a Diamond Ticket?
- Detection
- Prevention and Remediation

Introduction to Diamond Tickets

InfoGuard
SWISS CYBER SECURITY

Prerequisite: access to the krbtgt hash and the hash of a low-priv user

Creation: the Threat Actor modifies a existing TGT, insert data into the PAC and reencrypts it

Scope: Full Domain access

What is a Diamond Ticket

InfoGuard
SWISS CYBER SECURITY

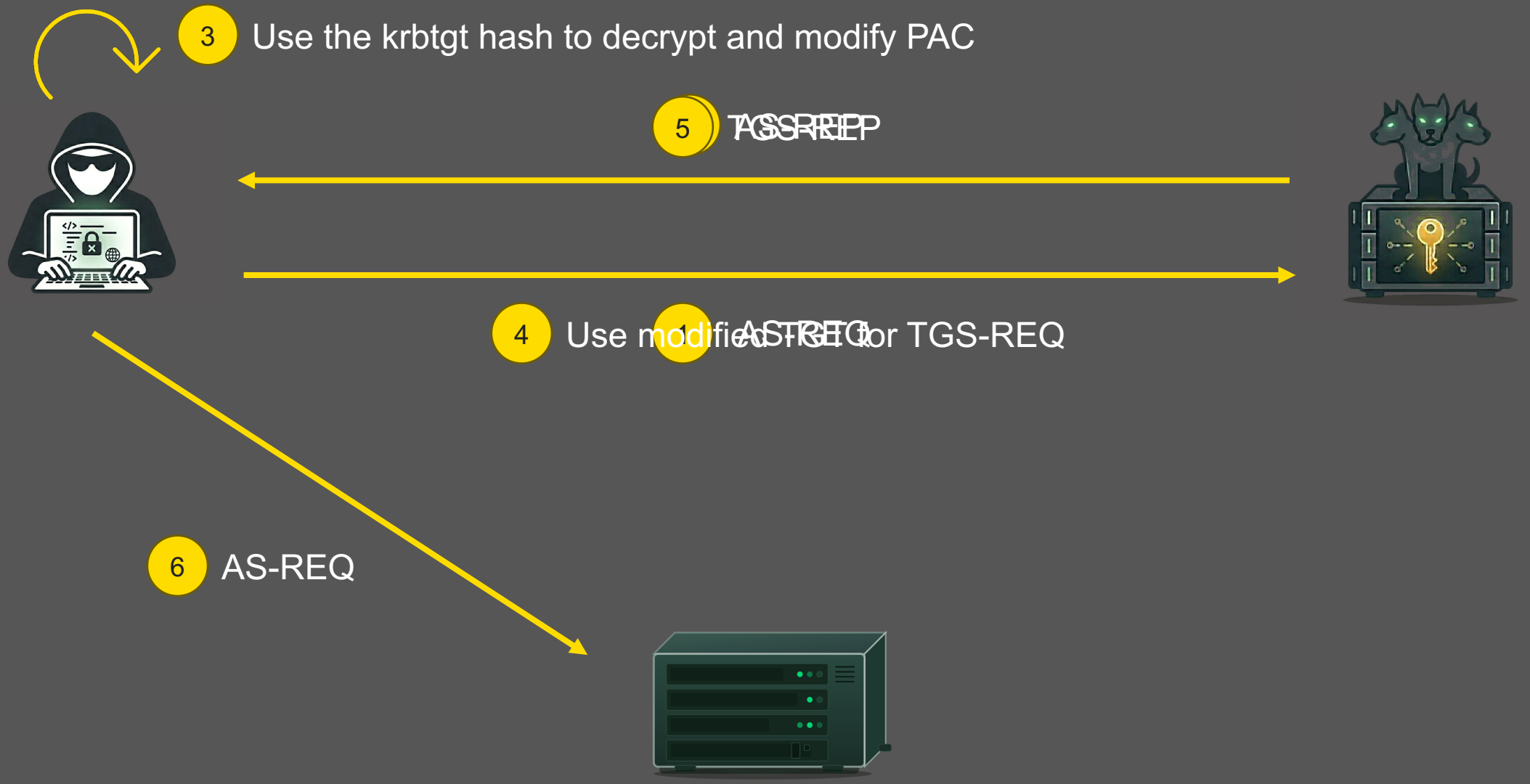
A “improved” Golden Ticket

Threat Actor first requests a legitimate TGT

As the Threat Actor has the krbtgt hash, they can decrypt the PAC, insert specific values and reencrypt it

Compared to a golden ticket, this flow looks more legitimate as we have a proper AS-REQ/AS-REP before Ticket usage

Diamond Ticket Workflow



Detecting Diamond Tickets

InfoGuard
SWISS CYBER SECURITY

With Diamond Tickets, we have a full and proper Kerberos flow

We can't rely on the absence of steps in the Kerberos authentication

Instead of focusing on anomalies in the flow, we need to focus on anomalies in the use of the Tickets

But first, we need to understand how a Threat Actor would use such a ticket

Detecting Diamond Tickets

InfoGuard
SWISS CYBER SECURITY

The Threat Actor uses Diamond Tickets for maintaining persistence for a high-privileged account (like a domain admin)

There are some Scenarios that could ease the detection
Scenario A: The TGT is requested for a low-privileged user, and the Service Ticket for a high-privileged user
Scenario B: The user for the TGT and the Service Ticket are the same

Detecting Diamond Tickets – Scenario A

InfoGuard
SWISS CYBER SECURITY

If the Threat Actor requests a l then changes the PAC to insert the domain administrat ng

The same source IP is used fo T for the Domain Admin

An OpSec aware Threat Actor will obviously know that

Event ID 4768 (TGT requested) for Bob

Event ID 4769 (Service Ticket requested) – for Domain Admin

Detecting Diamond Tickets – Scenario B

InfoGuard
SWISS CYBER SECURITY

If the Threat Actor requests a TGT and the following Service Ticket for the same user (just modify groups, etc. in the PAC), we can't hunt for this mismatch

Our (more or less) only anomaly would be the source IP
Is that normally used by the Domain Administrator?

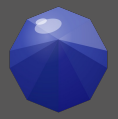
Besides that (normally) any use of a Domain Administrator is at least noteworthy

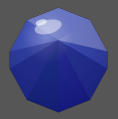
Investigation of Forged Tickets

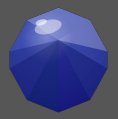
InfoGuard
SWISS CYBER SECURITY

Sapphire Tickets

InfoGuard
SWISS CYBER SECURITY

 What is a Diamond Ticket?

 Detection

 Prevention and Remediation

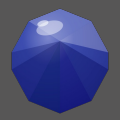
Introduction to Sapphire Tickets

InfoGuard
SWISS CYBER SECURITY

Prerequisite: access to the krbtgt hash and Hash of a low-priv user

Creation: the Threat Actor modifies a existing TGT, insert data into the PAC and reencrypts it

Scope: Full Domain access



What is a Sapphire Ticket

InfoGuard
SWISS CYBER SECURITY

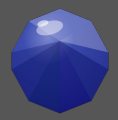
Same idea as diamond ticket, again “improvement” of golden ticket

Get a valid TGT, modify the PAC, and then use it to get a valid Service Ticket

The difference lies in how the PAC is populated

Diamond tickets include arbitrary PAC information, meaning they can differ from what is expected and, therefore, be suspicious

A Sapphire ticket uses a “trick” to get the correct PAC information for any given user, hence making the ticket look more realistic



Kerberos Delegation – S4U (S4U2Self + U2U)

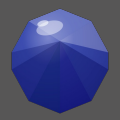
InfoGuard
SWISS CYBER SECURITY

S4U is a Kerberos extension

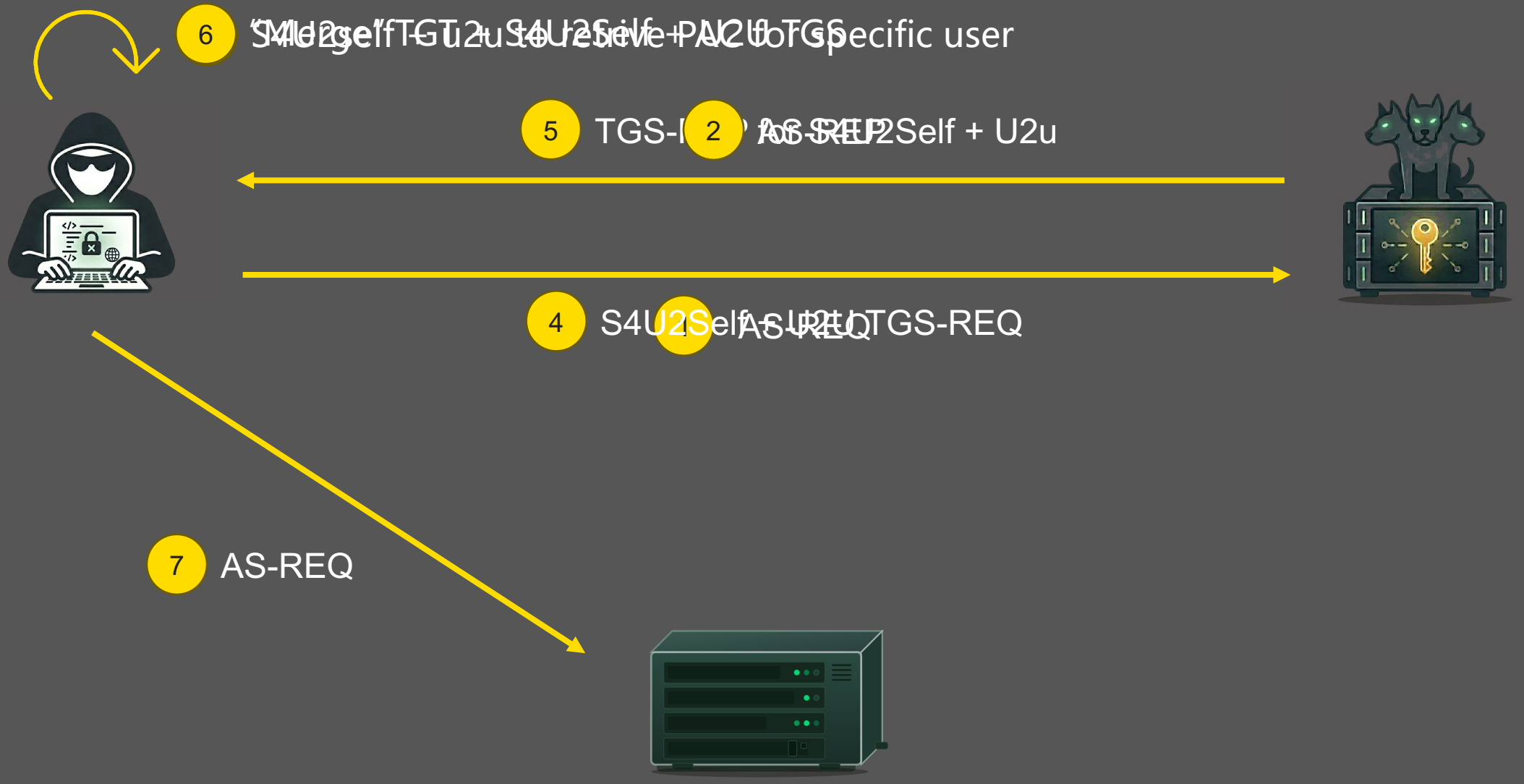
It allows you to request a Service Ticket on behalf of another user

The idea is to request a Service Ticket for the user that the Threat Actor wants to impersonate

Then copy his PAC from the Service Ticket into the legitimately requested TGT



Sapphire Ticket Workflow



How to detect a sapphire ticket

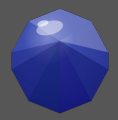
InfoGuard
SWISS CYBER SECURITY

We are facing the same issues as with the diamond tickets

The Kerberos Authentication flow is complete

Theoretically, the steps are the same as with diamond tickets

However, S4U2self leaves some logs behind



How to detect a sapphire ticket – S4U2Self + U2U

We can look for AS-REQ/TGS-REQ from the same IP for different users in quick succession

Forged Ticket comparision

InfoGuard
SWISS CYBER SECURITY

Ticket Type	Target Key	Scope	DC Interaction
Golden	KRBTGT	Entire Domain	No AS-REQ/REP
Silver	Service Account	Specific Service	None
Diamond	KRBTGT	Entire Domain	Full Flow
Sapphire	KRBTGT	Entire Domain	S4U2Self + TGS

Behavior Detection Summary

InfoGuard
SWISS CYBER SECURITY

As we've seen, there are no high-precision detections for forged tickets

Especially in crowded environments, these theory-based detections can generate a lot of false-positives, and tuning would be required

In an incident, you don't have time for that, and maybe not even the needed data

That's why artefacts from popular tools might give us some advantage

Tool-based detections

InfoGuard
SWISS CYBER SECURITY

Now that we've looked at detections based on the behaviour, let's look at traces left behind by their implementation in popular tools

Most common tools: Mimikatz, Rubeus and Impacket

How to detect a Forged Ticket - Lifetime

InfoGuard
SWISS CYBER SECURITY

The easiest way to detect a forged ticket is by looking at its lifetime

When creating a forged ticket using Mimikatz, Rubeus or Ticketer, the default lifetime is 10 years

```
kull_m_string_args_byName(argc, argv, L"startoffset", &szLifetime, L"0");
GetSystemTimeAsFileTime(&lifeTimeData.TicketStart);
*(PULONGLONG) &lifeTimeData.TicketStart -= *(PULONGLONG) &lifeTimeData.TicketStart % 10000000 - ((LONGLON
lifeTimeData.TicketRenew = lifeTimeData.TicketEnd = lifeTimeData.TicketStart;
kull_m_string_args_byName(argc, argv, L"endin", &szLifetime, L"5256000"); // ~ 10 years
*(PULONGLONG) &lifeTimeData.TicketEnd += (ULONGLONG) 10000000 * 60 * wcstoul(szLifetime, NULL, 0);
kull_m_string_args_byName(argc, argv, L"renewmax", &szLifetime, szLifetime);
*(PULONGLONG) &lifeTimeData.TicketRenew += (ULONGLONG) 10000000 * 60 * wcstoul(szLifetime, NULL, 0);
kprintf(L"User      : %s\nDomain    : %s", szUser, szDomain);
if(kull_m_string_args_byName(argc, argv, L"sid", &szSid, NULL))
```

```
                                                    'PAC_ATTRIBUTES_INFO and PAC_REQUESTOR')
parser.add_argument('-duration', action="store", default = '87600', help='Amount of hours till the ticket expires (d
parser.add_argument('-ts', action='store_true', help='Adds timestamp to every logging output')
parser.add_argument('-debug', action='store_true', help='Turn DEBUG output ON')
```

Can you change the lifetime? -> Of Course
Does everybody do that? -> Of course not

How to detect a golden Ticket - Lifetime

InfoGuard
SWISS CYBER SECURITY

To view the lifetime of tickets, we can use the klist command on a system

To scale this over all systems, e.g., Velociraptor can be used

https://docs.velociraptor.app/exchange/artifacts/pages/windows.kerberos.goldentickettriage/

```
PS C:\Users\adm_peter\Desktop\evbl_tools\mimikatz_trunk\x64> klist

Current LogonId is 0:0x151b98c

Cached Tickets: (1)

#0>     Client: rio.todd @ simlab.local
        Server: krbtgt/simlab.local @ simlab.local
        KerbTicket Encryption Type: RSADSI RC4-HMAC(NT)
        Ticket Flags 0x40e00000 -> forwardable renewable initial pre_authent
        Start Time: 5/11/2026 15:20:39 (local)
        End Time:   5/8/2036 15:20:39 (local)
        Renew Time: 5/8/2036 15:20:39 (local)
        Session Key Type: RSADSI RC4-HMAC(NT)
        Cache Flags: 0x1 -> PRIMARY
        Kdc Called:
```

How to detect a golden ticket – Default user and Groups

InfoGuard
SWISS CYBER SECURITY

Common:

- `/domain` - the fully qualified domain name (eg: `chocol`
- `/sid` - the SID of the **domain** (eg: `S-1-5-21-130452501`
- `/user` - the username you want to impersonate, keep i
  name for this well-known account.
- `/id` - *optional* - the id of the user - default is: `500` for
- `/groups` - *optional* - id of groups the user belongs (firs
  default is: `513,512,520,518,519` for the well-known Ad

```
string groups = "520,512,513,519,518";
```

By default, Windows does not log Group Memberships
We can enable Event ID 4627 (Group membership Information)

Event 4627, Microsoft Windows security auditing.

General | Details

New Logon:
- Security ID: ENV12\Joe
- Account Name: joe
- Account Domain: ENV12.LOCAL
- Logon ID: 0xBC995A1

Event in sequence: 1 of 1

Group Membership:
- ENV12\Domain Users
- Everyone
- BUILTIN\Users
- BUILTIN\Pre-Windows 2000 Compatible Access
- BUILTIN\Administrators
- NT AUTHORITY\NETWORK
- NT AUTHORITY\Authenticated Users
- NT AUTHORITY\This Organization
- ENV12\Domain Admins
- Authentication authority asserted identity
- ENV12\Denied RODC Password Replication Group
- Mandatory Label\High Mandatory Level

How to detect a Golden/Diamond/Sapphire Ticket - Logins

InfoGuard
SWISS CYBER SECURITY

Threat Actors often use the domain administrator as a target account for the Forged Tickets

Normally, domain administrators do not log in from arbitrary systems, but only a few selected ones

So we can compare TGS-REQ/AP-REQ for Domain Administrators for their source

Maybe we can correlate findings based on the source machine

Automating all of this (and more)

InfoGuard
SWISS CYBER SECURITY

There is an interesting Proof-of-Concept for these detections and more

The tool is called WonkaVision

It employs a lot of “Indicators of Attack (IOA)” to give Scores to suspicious tickets

One downside is that you give it access to your krbtgt key

An overview of all employed IOAs can be found in the function AnalyseSession

InfoGuard
SWISS CYBER SECURITY

Prevention and Remediation
Golden, Sapphire and Diamond Tickets

How to prevent a golden/diamond/sapphire Ticket

InfoGuard
SWISS CYBER SECURITY

Well, you let your Domain fall ;)

Generally, everything that hardens your Domain Controller/Domain Administrators will help

Limit the number of systems from which Domain Administrators can log in

Domain Controllers are Tier 0 Systems; make sure

MFA for Domain Administrators

Enable more Logging!

Some things can’t be detected with native Windows Logging, EDR/XDR’s are recommendet

How to remediate a golden/diamond/sapphire Ticket

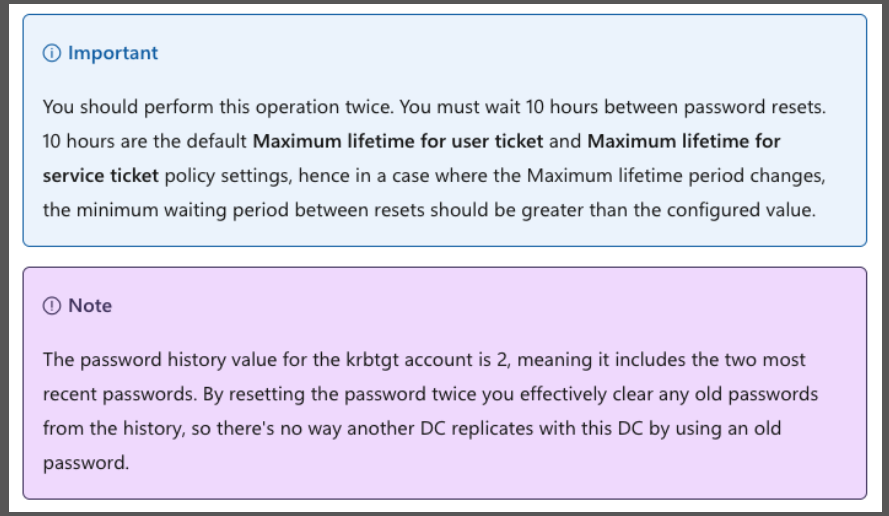
InfoGuard
SWISS CYBER SECURITY

"Rotate the krbtgt password twice" most commonly known remediation
But why twice, and how much time should be between the rotations?

The Domain Controller holds the last password and can use that in case there are any issues

Therefore only resetting the password once is not enough

The second rotation should be performed when the first change has successfully propagated trough the domain



Important

You should perform this operation twice. You must wait 10 hours between password resets. 10 hours are the default **Maximum lifetime for user ticket** and **Maximum lifetime for service ticket** policy settings, hence in a case where the Maximum lifetime period changes, the minimum waiting period between resets should be greater than the configured value.

Note

The password history value for the krbtgt account is 2, meaning it includes the two most recent passwords. By resetting the password twice you effectively clear any old passwords from the history, so there's no way another DC replicates with this DC by using an old password.

source

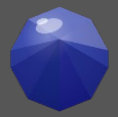
Summary and Conclusion

InfoGuard
SWISS CYBER SECURITY

Investigation of forged tickets is difficult for common enterprises

To fully investigate such behavior telemetrie Is needed that is not not available

As golden/diamond/sapphire require krbtgt access, it is always to good idea to rotate krgtbt hash nonetherless in case of a major incident



Less usage by FinCrime more by APT