I Trusted You

A Demonstrated Abuse of Cloud Kerberos Trust

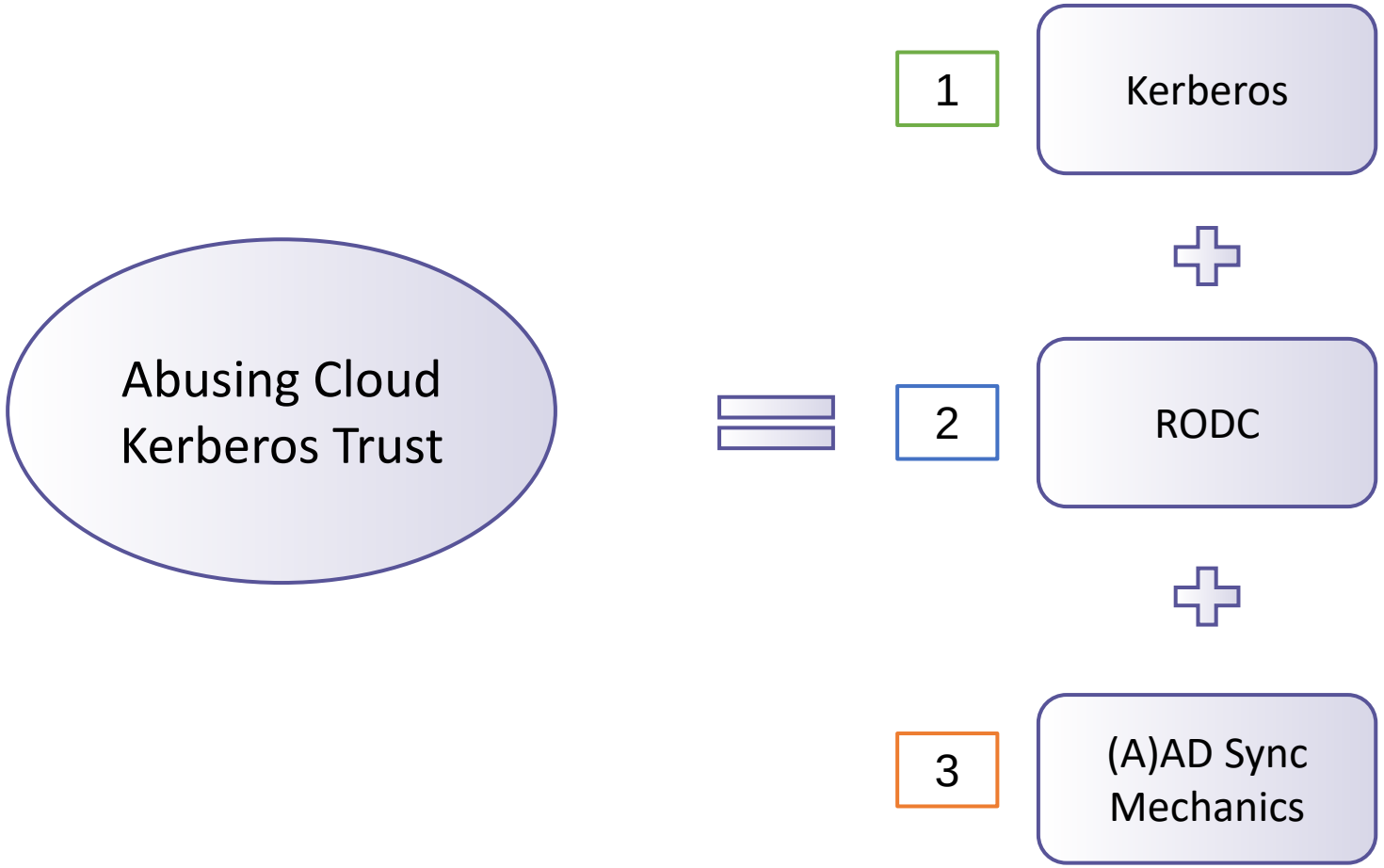
The Big Deal

- Given
  - Default configuration
  - Use of Cloud Kerberos Trust
  - Line of sight to an on-premises domain controller
- An Azure AD compromise is equivalent to an on-premises compromise
  - Dare I say the lines are.... blurred?
- Does not rely on misconfigurations

What is Cloud Kerberos Trust?

- Cloud Kerberos Trust is an Azure Active Directory (AAD) feature
- Allows users to authenticate to on-premises Active Directory (AD) resources using Windows Hello for Business
- Passwordless authentication without PKI -> Simpler deployment
- Stronger initial authentication by leveraging Azure's MFA capabilities

Overview

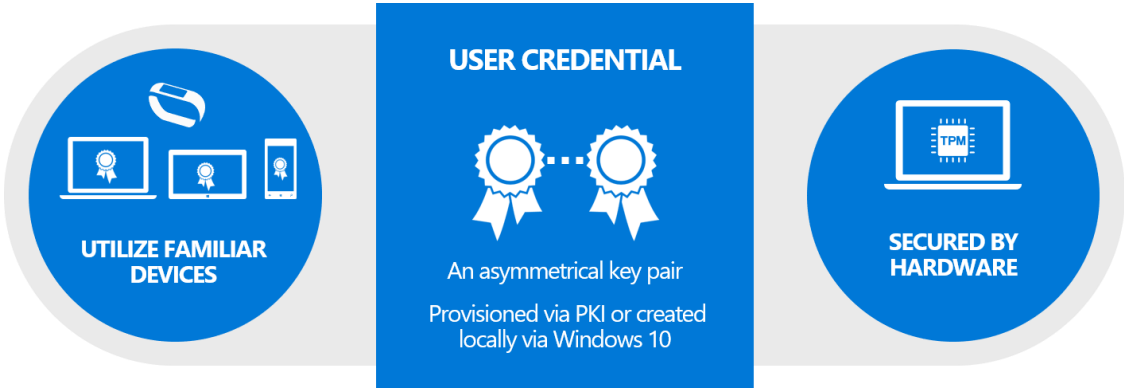

Abusing Cloud Kerberos Trust
1
Kerberos
2
RODC
3
(A)AD Sync Mechanics

Cloud -> On-Prem Dominance

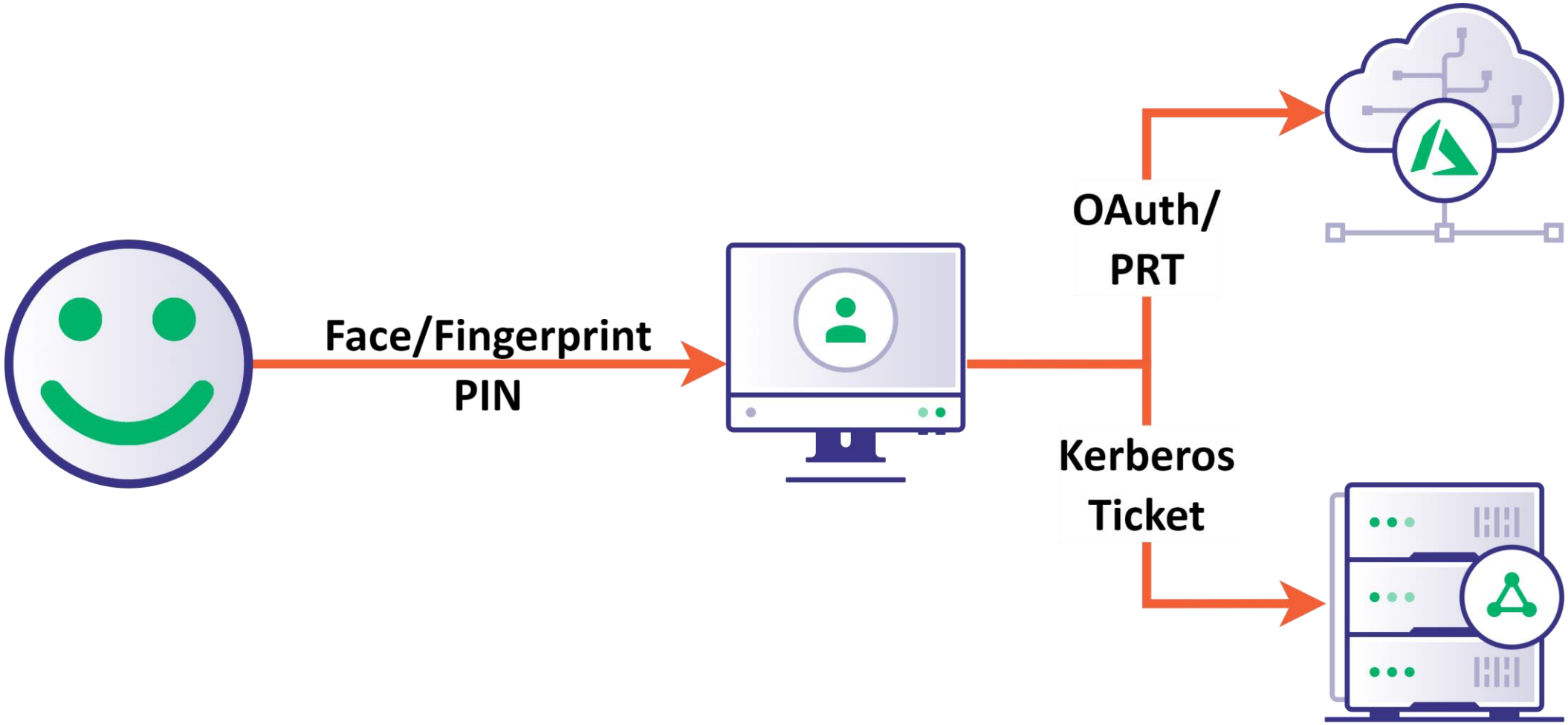
- There is *usually* an attack path from privileged cloud access to privileged on-premises access (e.g., Domain Admin)
- These paths *typically* abuse misconfiguration or insecure design decisions, such as:
  - Domain controllers running in VMs in the cloud
  - Domain admins synced to AAD
- These paths are *almost* always available, but they are not *guaranteed*
- **Is there a *guaranteed* path from Azure dominance to on-prem AD dominance?**

The Push for Passwordless Authentication

- Microsoft has been encouraging users and organizations to shift to passwordless authentication with Windows Hello for Business



The Push for Passwordless Authentication



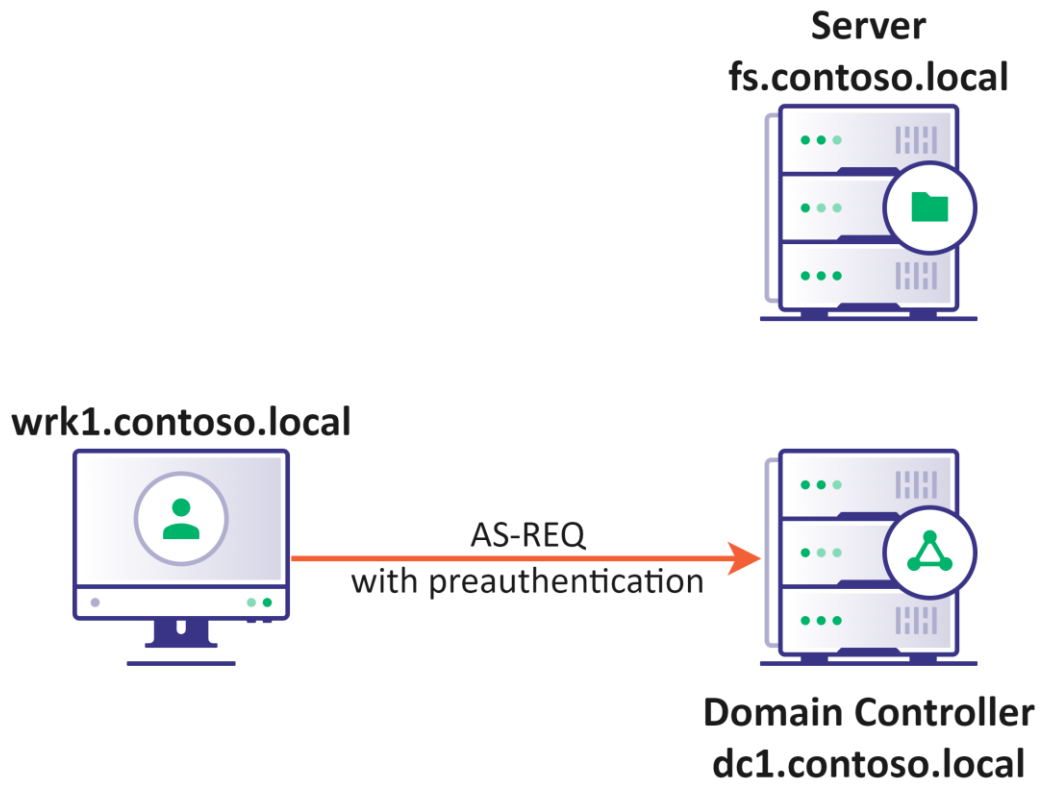
Passwordless Authentication On-Prem

- Microsoft introduced three deployment models for on-prem passwordless authentication:
  - Certificate Trust
  - Key Trust
  - **Cloud Kerberos Trust**
- All three models ultimately allow users to obtain Kerberos Ticket Granting Tickets without entering their passwords
- Kerberos is the primary authentication protocol for on-prem Active Directory

Kerberos Authentication Process – AS-REQ

- The user sends an ***AS-REQ*** to the DC with an encrypted timestamp for ***pre-authentication***

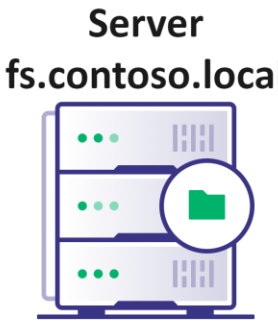
1



Kerberos Authentication Process

- The DC generates a ***ticket-granting-ticket (TGT)***

krbtgt/contoso.local	
Flags:	forwardable, renewable
Start Time:	14/1/2023 08:00
End Time:	14/1/2023 18:00
Renew Time:	21/1/2023 08:00
Username:	John
User RID:	1008
Domain SID:	S-1-5-21323...
Groups:	1004, 1007
ExtraSIDs:	S-1-5-84538...
Session Key:	*<BLOB>*
...	

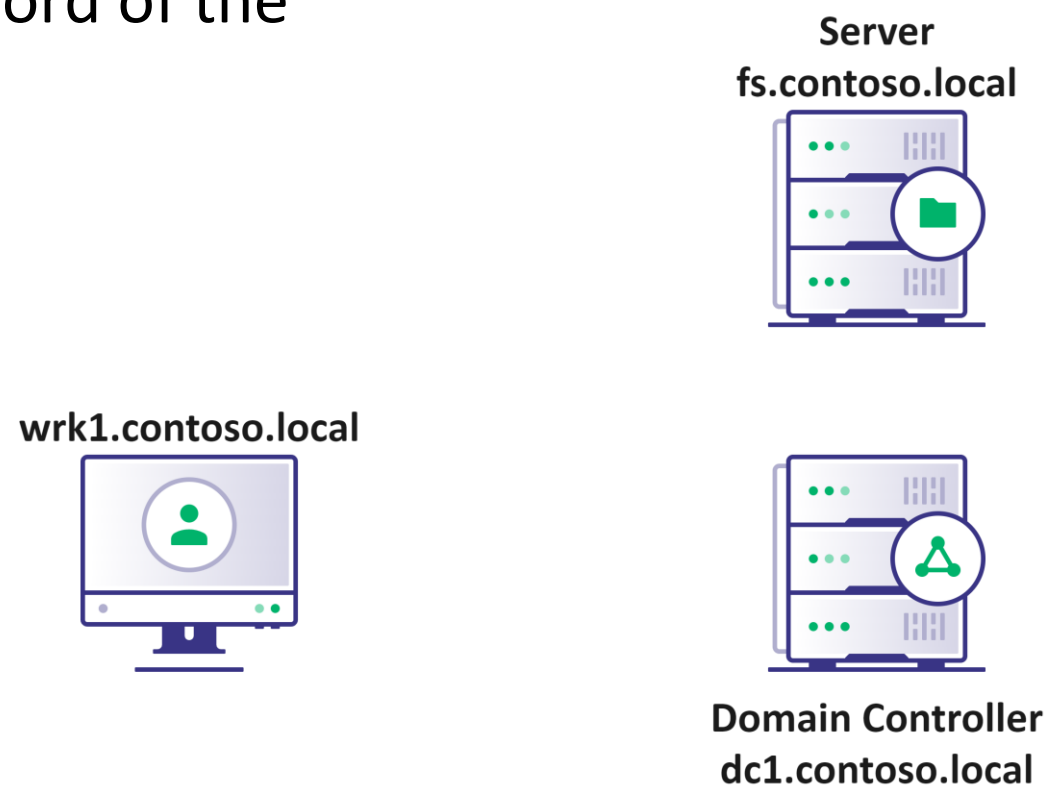


Kerberos Authentication Process

1

- The DC generates a ***ticket-granting-ticket (TGT)***
- The DC encrypts the TGT with the password of the ***krbtgt*** account

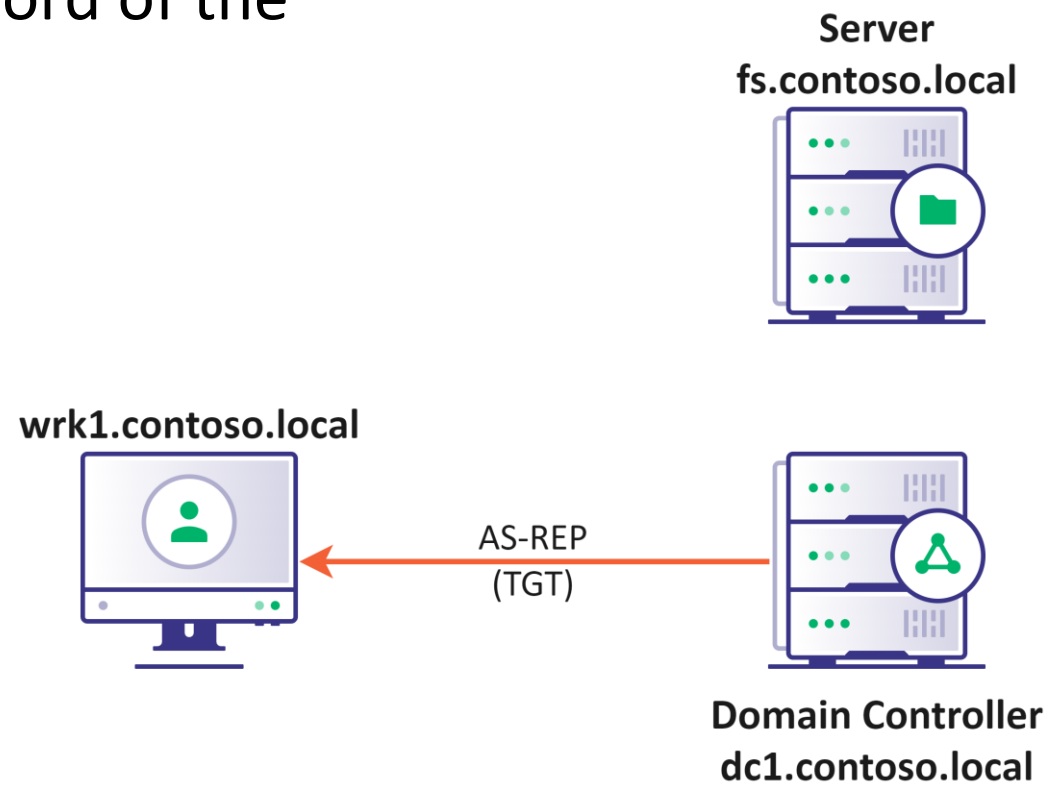
krbtgt/contoso.local	
Gmbht: gpsxbsebcmf, sfofxbcmf	
Tubsu Ujnf:	25/2/3134 19:11
Foe Ujnf:	25/2/3134 29:11
Sfofx Ujnf:	32/2/3134 19:11
Vtfsobnf:	Kpio
Vtfs SJE:	2119
Epnbjo TJE:	T-2-6-32434...
Hspvqt:	2115, 2118
FyusbTJEt:	T-2-6-95649...
Tfttjpo Lfz:	*<ENCRYPTED BLOB>*
...	



Kerberos Authentication Process – AS-REP

1

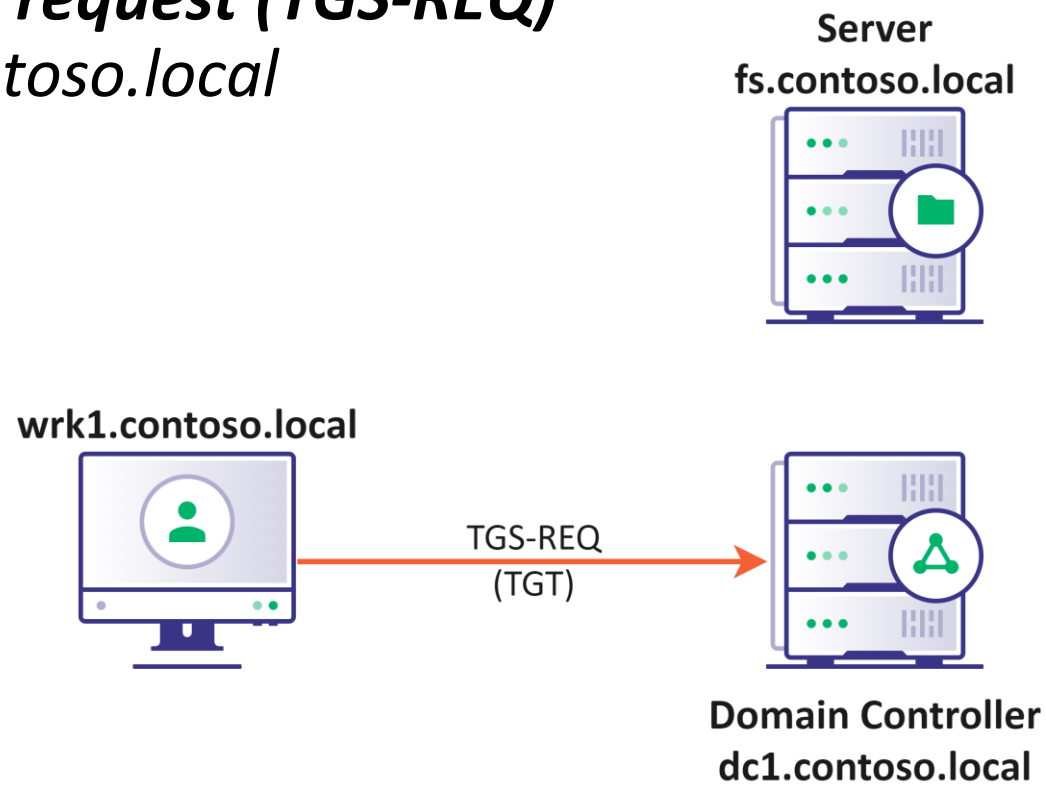
- The DC generates a ***ticket-granting-ticket (TGT)***
- The DC encrypts the TGT with the password of the ***krbtgt*** account
- The DC sends the encrypted TGT to the user in an ***AS-REP*** message



Kerberos Authentication Process – TGS-REQ

1

- The user runs **dir \\fs.contoso.local\C$**
- The user sends a ***ticket-granting-service request (TGS-REQ)*** to the DC to obtain a ticket to *cifs/fs.contoso.local*
- The TGS-REQ contains the user's TGT

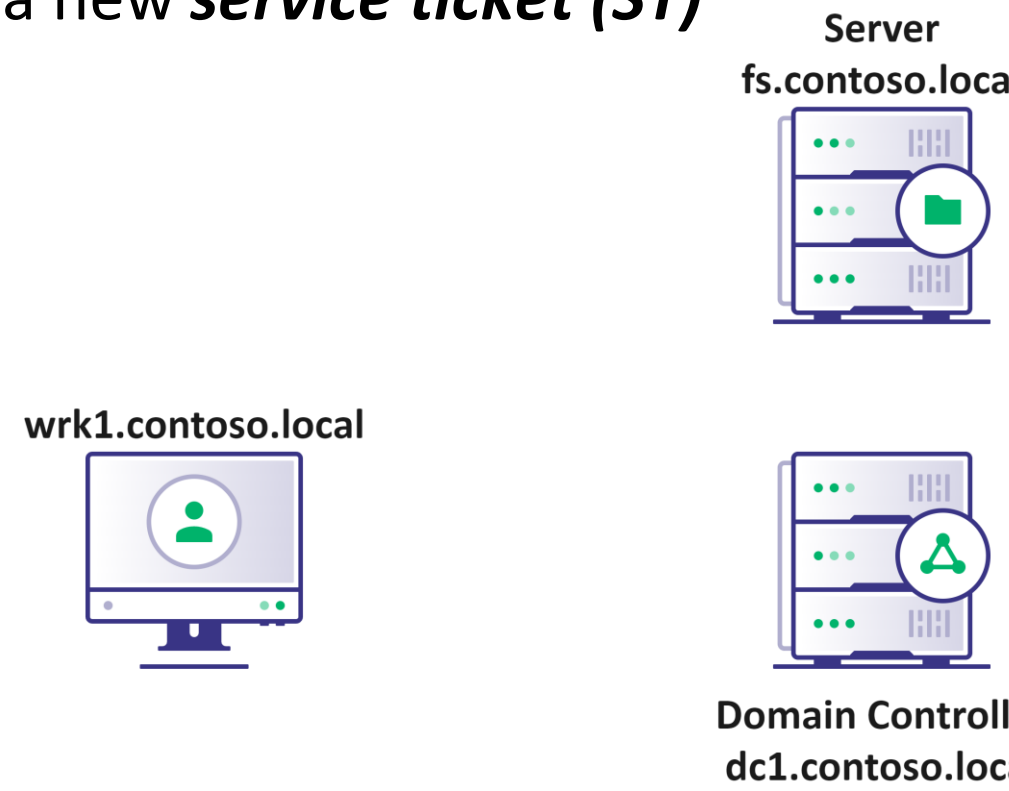


Kerberos Authentication Process

- The DC decrypts and validates the TGT
- The DC ***<u>copies</u>*** the data from the TGT to a new ***service ticket (ST)***

krbtgt/contoso.local	
Flags:	forwardable, renewable
Start Time:	14/1/2023 08:00
End Time:	14/1/2023 18:00
Renew Time:	21/1/2023 08:00
Username:	John
User RID:	1008
Domain SID:	S-1-5-21323...
Groups:	1004, 1007
ExtraSIDs:	S-1-5-84538...
Session Key:	*<BLOB>*
...	

cifs/fs.contoso.local	
Flags:	forwardable, renewable
Start Time:	14/1/2023 08:00
End Time:	14/1/2023 18:00
Renew Time:	21/1/2023 08:00
Username:	John
User RID:	1008
Domain SID:	S-1-5-21323...
Groups:	1004, 1007
ExtraSIDs:	S-1-5-84538...
Session Key:	*<NEW BLOB>*
...	



Kerberos Authentication Process

1

- The DC encrypts the new service ticket with a key derived from the password of the service account

krbtgt/contoso.local	
Flags:	forwardable, renewable
Start Time:	14/1/2023 08:00
End Time:	14/1/2023 18:00
Renew Time:	21/1/2023 08:00
Username:	John
User RID:	1008
Domain SID:	S-1-5-21323...
Groups:	1004, 1007
ExtraSIDs:	S-1-5-84538...
Session Key:	*<BLOB>*
...	

cifs/fs.contoso.local	
Hnciu:	hqtyctfcdng, tgpgycdng
Uvctv Vkog:	36/3/4245 2::22
Gpf Vkog:	36/3/4245 3::22
Tgpgy Vkog:	43/3/4245 2::22
Wugtpcog:	Lqjp
Wugt TKF:	322:
Fqockp UKF:	U-3-7-43545...
Itqwru:	3226, 3229
GzvtcUKFu:	U-3-7-:675:...
Uguukqp Mg{:	*<ENCRYPTED BLOB>*
...	

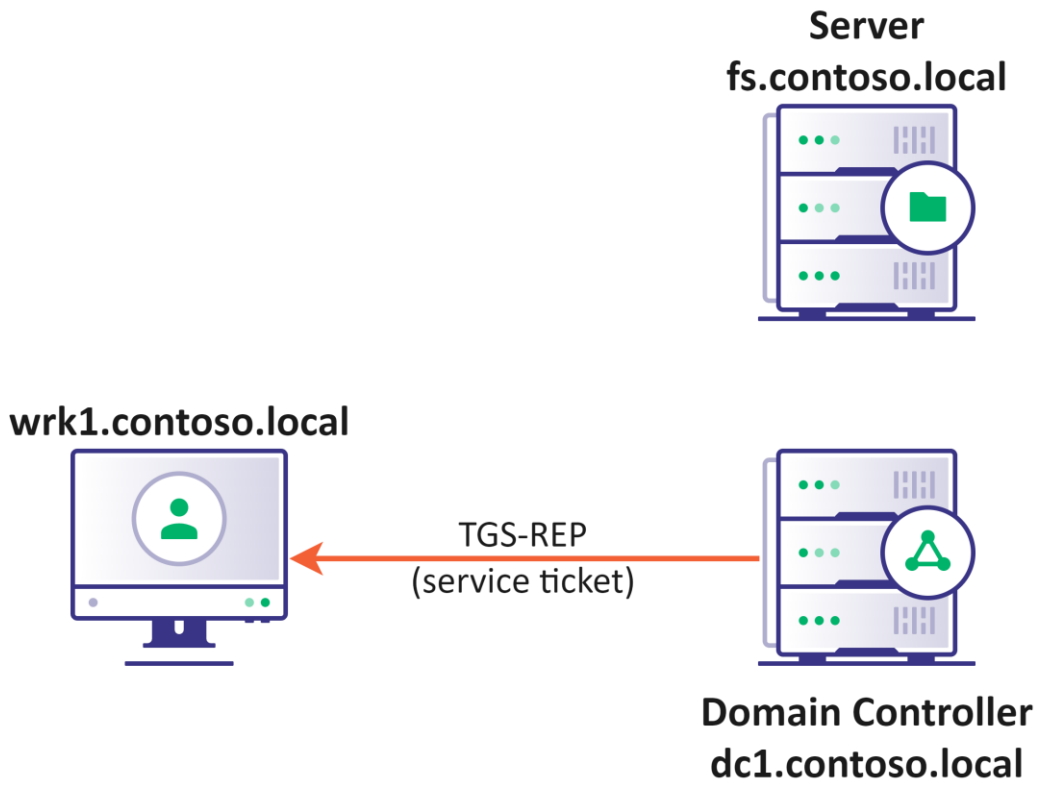
Kerberos Authentication Process – TGS-REP

1

- The DC sends the encrypted service ticket to the user in a ***TGS-REP*** message

krbtgt/contoso.local	
Flags:	forwardable, renewable
Start Time:	14/1/2023 08:00
End Time:	14/1/2023 18:00
Renew Time:	21/1/2023 08:00
Username:	John
User RID:	1008
Domain SID:	S-1-5-21323...
Groups:	1004, 1007
ExtraSIDs:	S-1-5-84538...
Session Key:	*<BLOB>*
...	

cifs/fs.contoso.local	
Hnciu:	hqtyctfcdng, tgpgycdng
Uvctv Vkog:	36/3/4245 2::22
Gpf Vkog:	36/3/4245 3::22
Tgpgy Vkog:	43/3/4245 2::22
Wugtpcog:	Lqjp
Wugt TKF:	322:
Fqockp UKF:	U-3-7-43545...
Itqwru:	3226, 3229
GzvtcUKFu:	U-3-7-:675:...
Uguukqp Mg{:	*<ENCRYPTED BLOB>*
...	

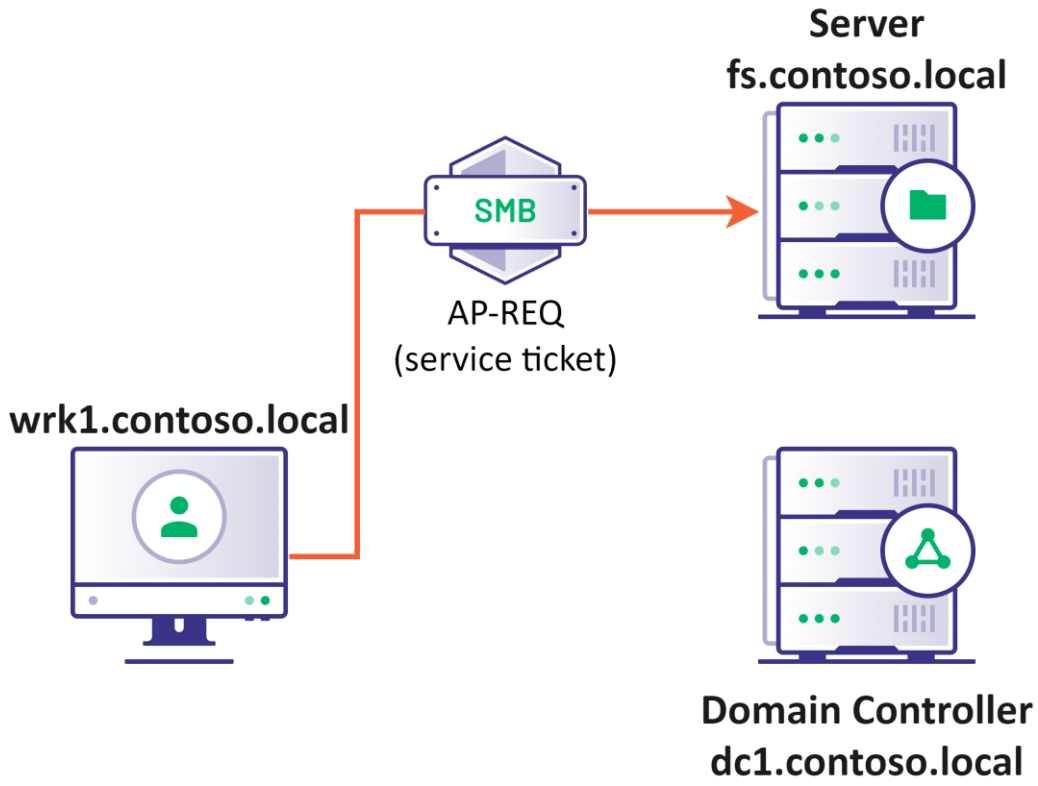


Kerberos Authentication Process – AP-REQ

1

- The user sends the service ticket to the SMB service at fs.contoso.local

cifs/fs.contoso.local	
Hnciu: hqtyctfcdng, tgpgycdng	
Uvctv Vkog:	36/3/4245 2::22
Gpf Vkog:	36/3/4245 3::22
Tgpgy Vkog:	43/3/4245 2::22
Wugtpcog:	Lqjp
Wugt TKF:	322:
Fqockp UKF:	U-3-7-43545...
Itqwru:	3226, 3229
GzvtcUKFu:	U-3-7-:675:...
Uguukqp Mg{:	*<ENCRYPTED BLOB>*
...	

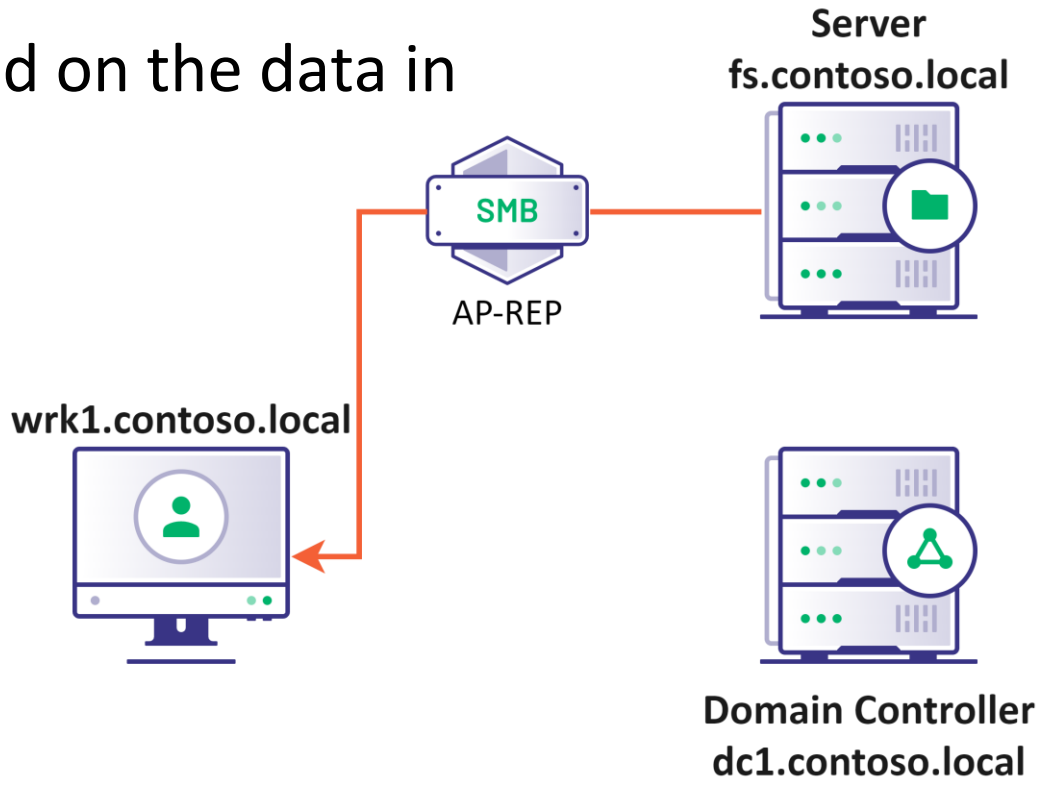


Kerberos Authentication Process – AP-REP

1

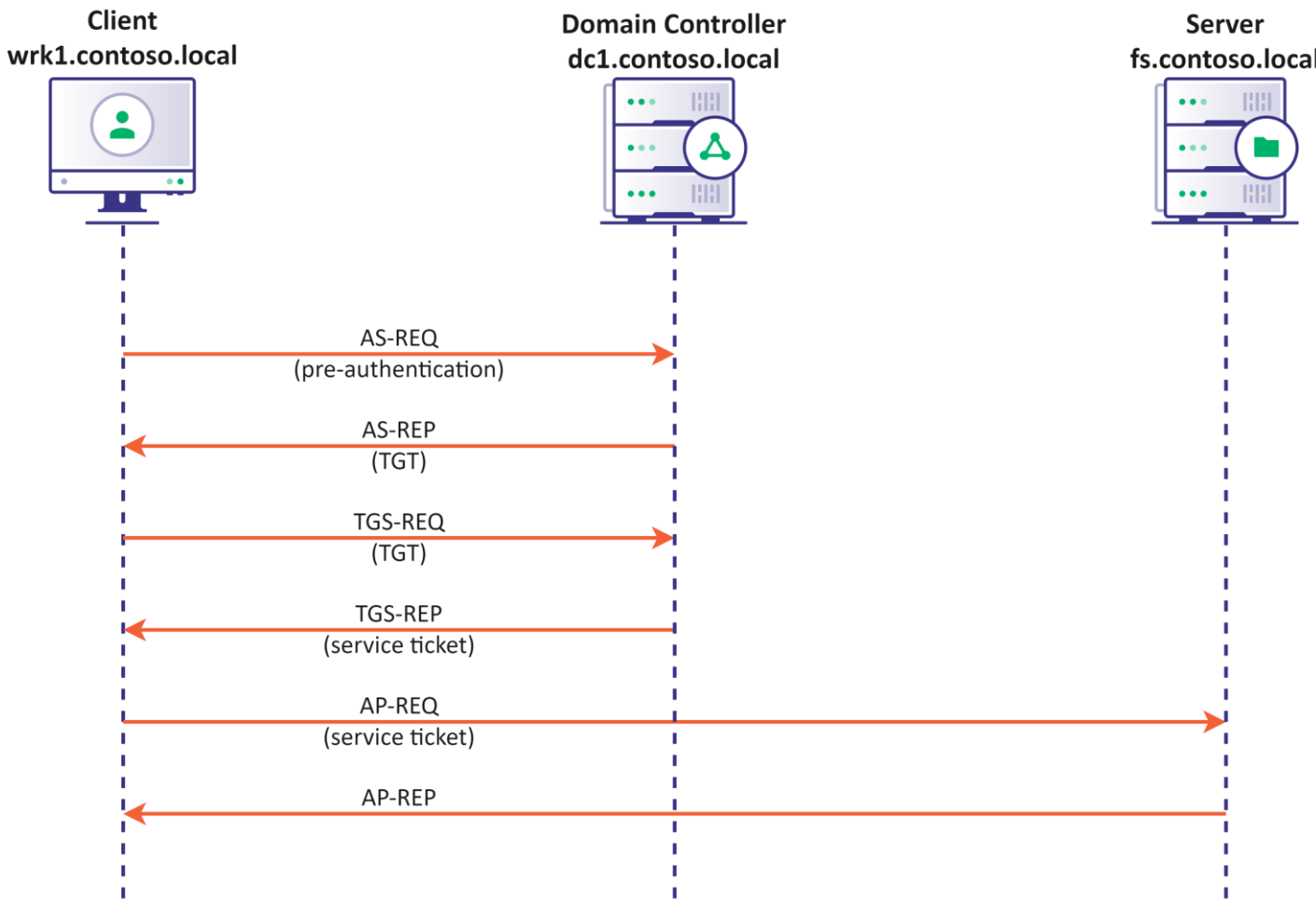
- The server authenticates the user by decrypting and validating the service ticket
- The sever can approve/deny access based on the data in the ticket

cifs/fs.contoso.local	
Flags:	forwardable, renewable
Start Time:	14/1/2023 08:00
End Time:	14/1/2023 18:00
Renew Time:	21/1/2023 08:00
Username:	John
User RID:	1008
Domain SID:	S-1-5-21323...
Groups:	1004, 1007
ExtraSIDs:	S-1-5-84538...
Session Key:	*<BLOB>*
...	



Kerberos Authentication Process Summary

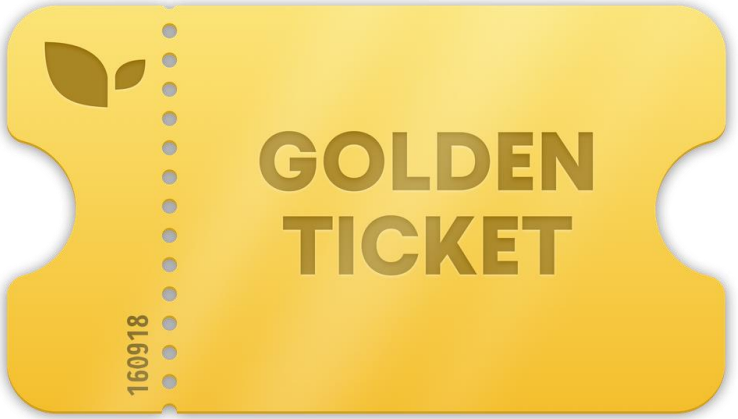
1



The Keys to the Kingdom and Golden Tickets

1

- The KRBTGT keys protect the TGTs
- If attackers compromise those keys, they can modify existing TGTs or forge new ones to impersonate any user or with any access rights
  - Reminder: the information is **copied** from the TGT to the ST
- This is the infamous “Golden Ticket Attack”
- The KRBTGT keys are the “keys to the kingdom” and must be protected accordingly



RODC: A Blast from the Past

2

- The Read-Only Domain Controller (RODC) is Microsoft's creative solution for physical locations that don't have adequate security to host a domain controller, but still require directory services
  - Examples: branch office, retail store, mine site
- The RODC does not have write access to objects
- The RODC has a "filtered" copy of the directory

The RODC Password Replication Policy 2

- The RODC can replicate the passwords of accounts per the RODC's password replication policy, defined by the RODC's msDS-NeverRevealGroup and msDS-RevealOnDemandGroup attributes
- The msDS-NeverRevealGroup is the deny list
- The msDS-RevealOnDemandGroup is the allow list
- If an account is listed in both, the deny list takes precedence
- Ideally, the policy should allow password replication only of accounts in the same physical location as the RODC

Partial and Full TGTs

2

- RODCs replicate *some* passwords to authenticate *some* users
  - Leads to the creation of TGTs and STs
- Every RODC has its own set of KRBTGT keys
  - Allows RODCs to generate “partial” TGTs to be used for obtaining service tickets from the same RODC
- The partial TGTs can also be used to obtain service tickets from a writable DC, but only if the user’s password is permitted to replicate to the corresponding RODC
- A service ticket request (TGS-REQ) for the “krbtgt” service returns a TGT, allowing users to exchange their partial TGT for a “full” TGT

Introducing Cloud Kerberos Trust

2

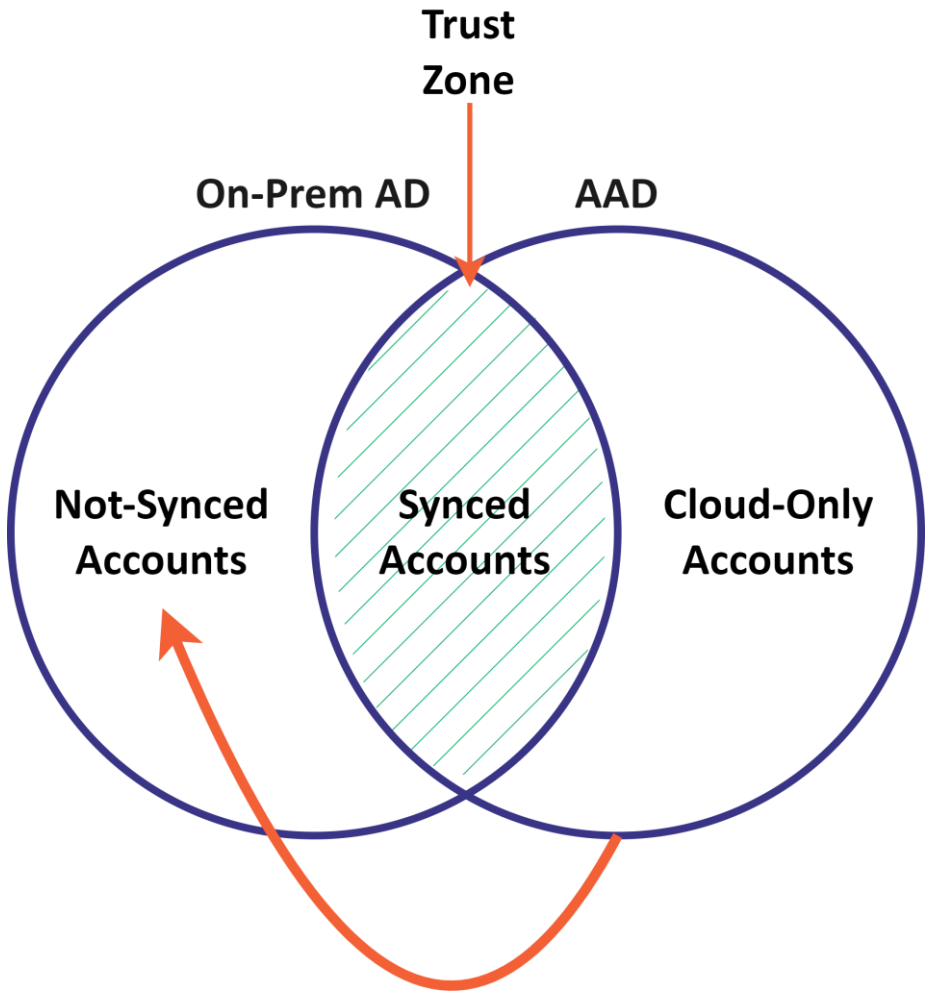
- Microsoft's creative solution for generating TGTs in AAD
- AAD has an RODC object in on-prem AD, and a corresponding set of KRBTGT keys
- AAD can generate partial TGTs for users to access on-prem resources with Kerberos authentication

AAD's RODC Password Replication Policy

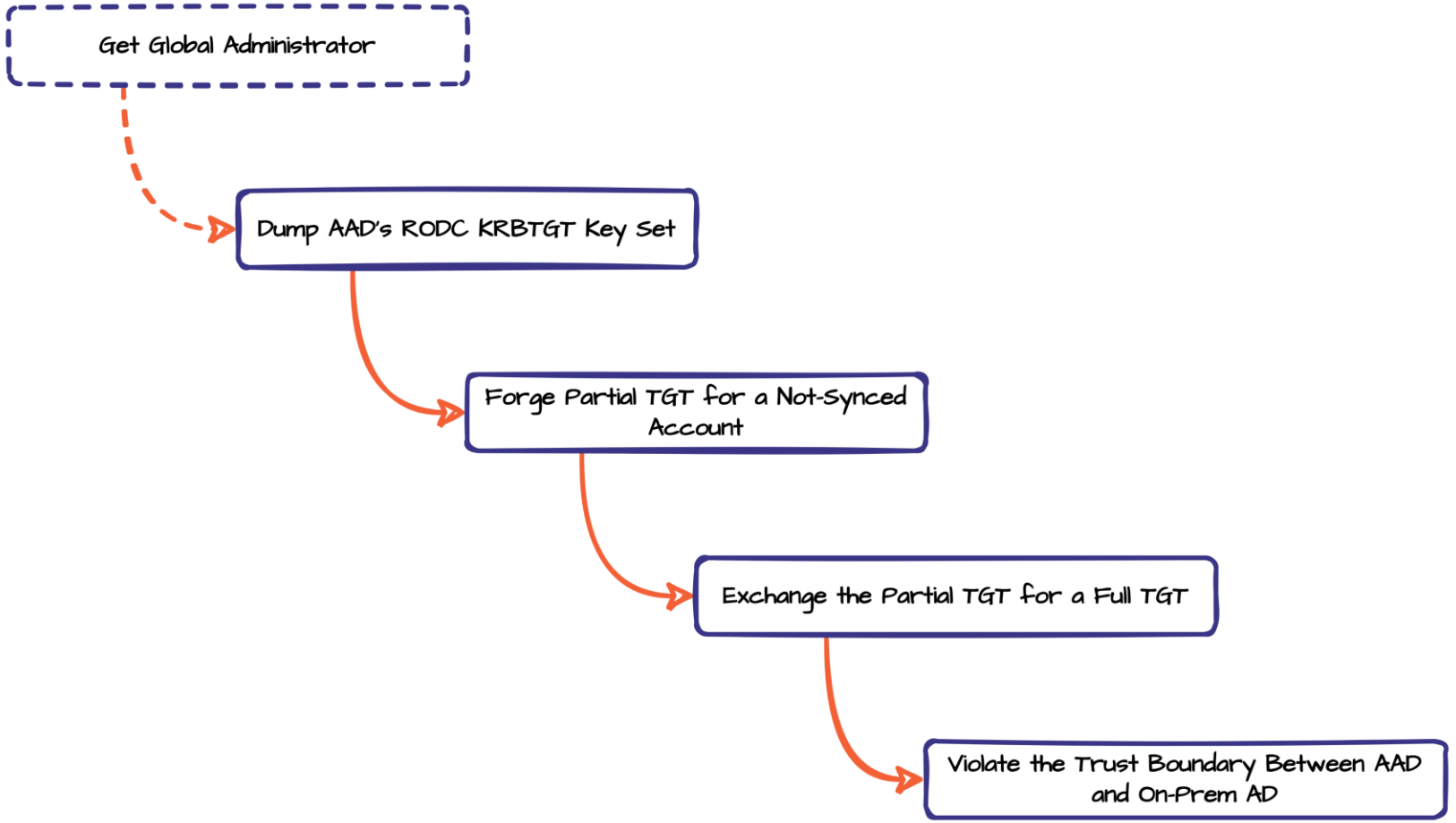
2

- Deny:
  - Schema Admins
  - Enterprise Admins
  - Administrators
  - Cert Publishers
  - Domain Admins
  - Backup Operators
  - Domain Controllers
  - Account Operators
  - Server Operators
- Allow:
  - Domain Users

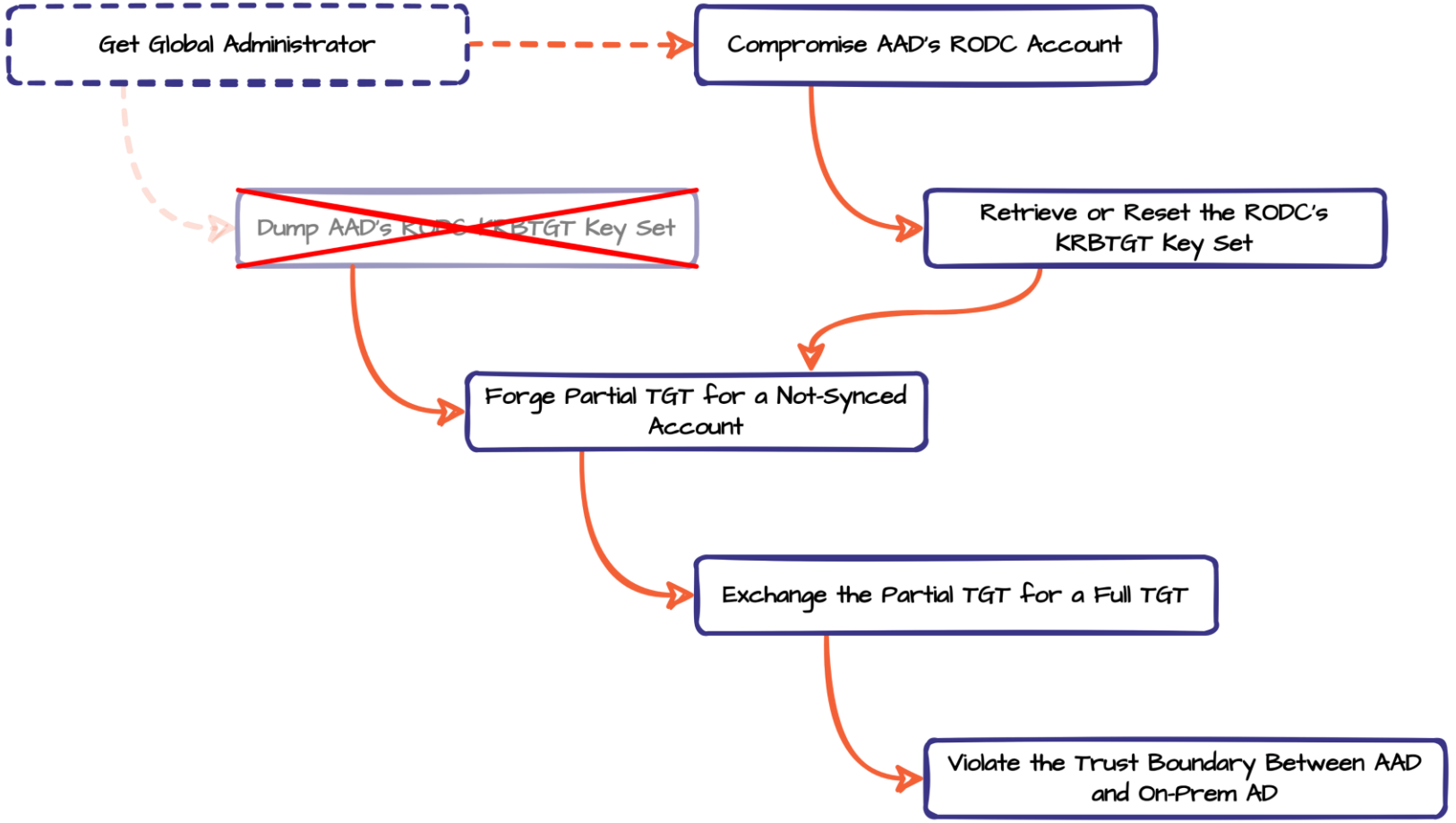
Violating the Trust Between AAD and AD



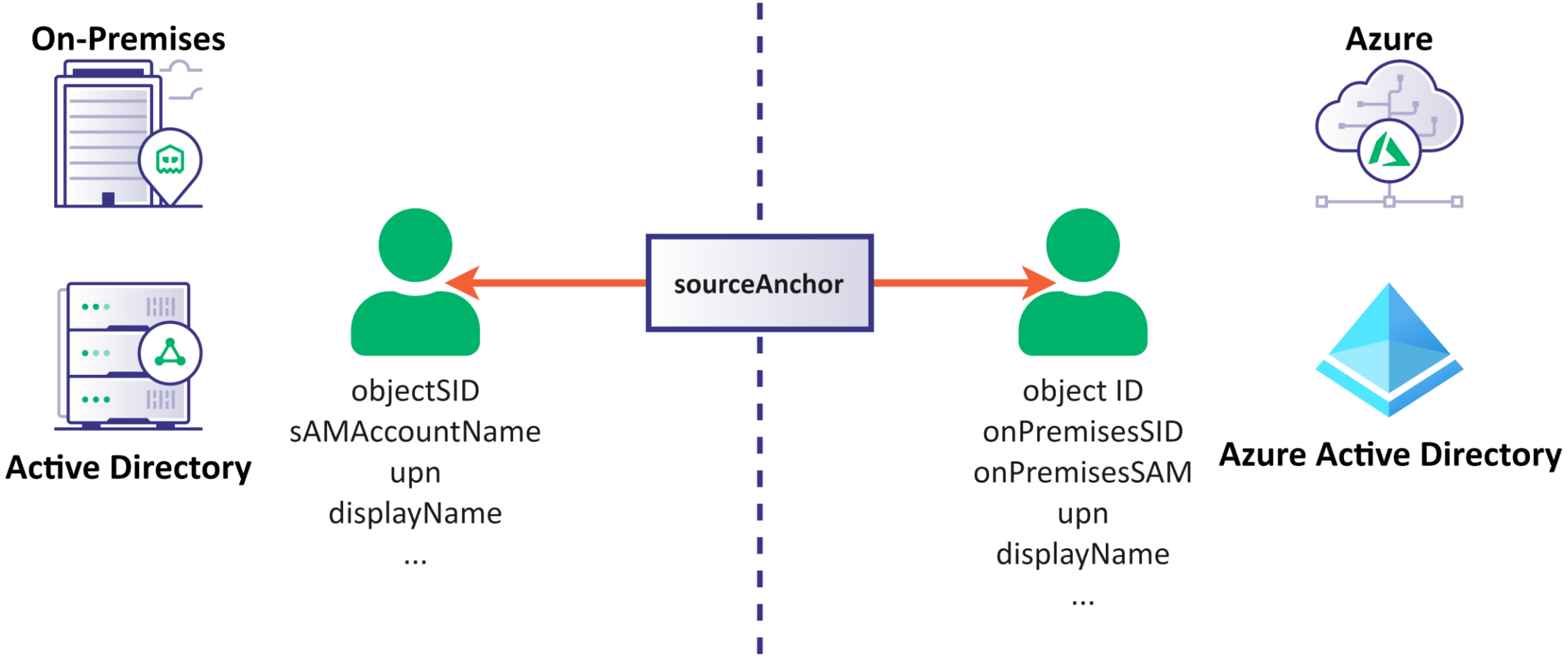
Abusing Cloud Kerberos Trust: Dump AAD's KRBTGT Keys?



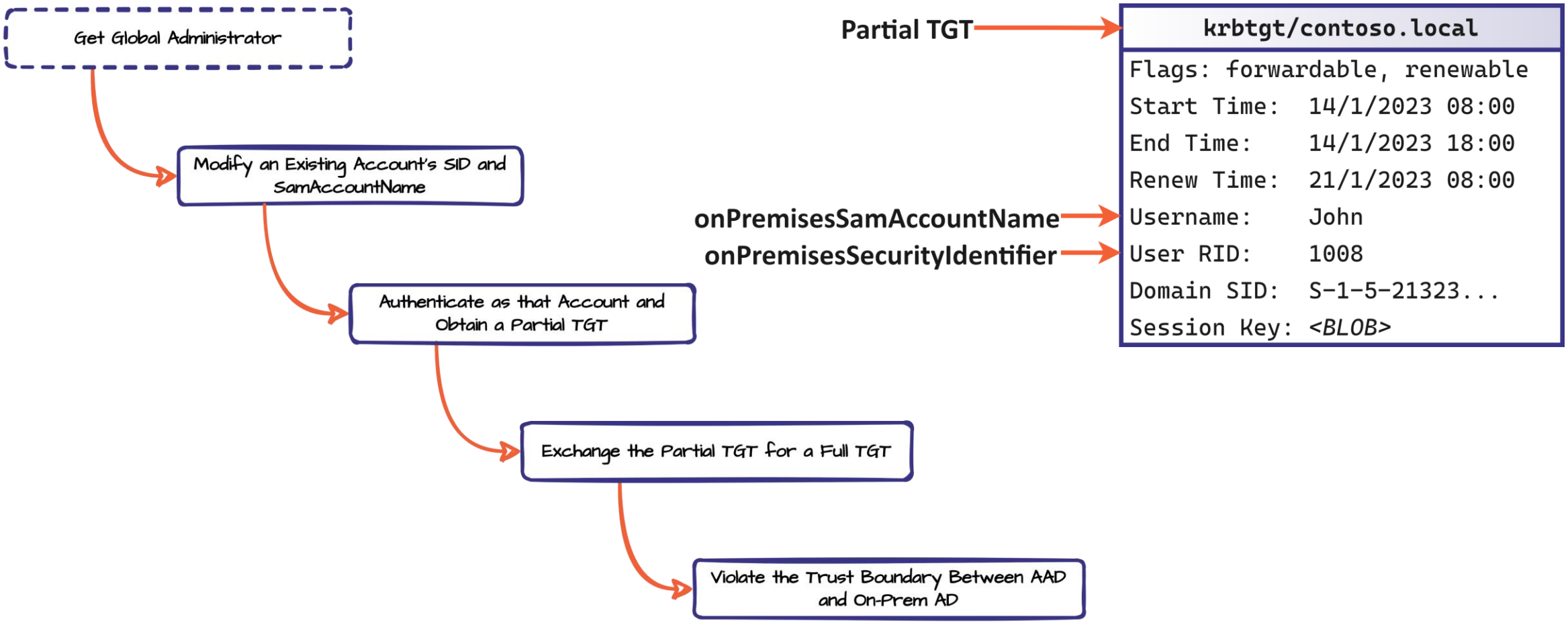
Abusing Cloud Kerberos Trust: Compromise AAD's RODC Account?



Syncing between On-Prem / Cloud



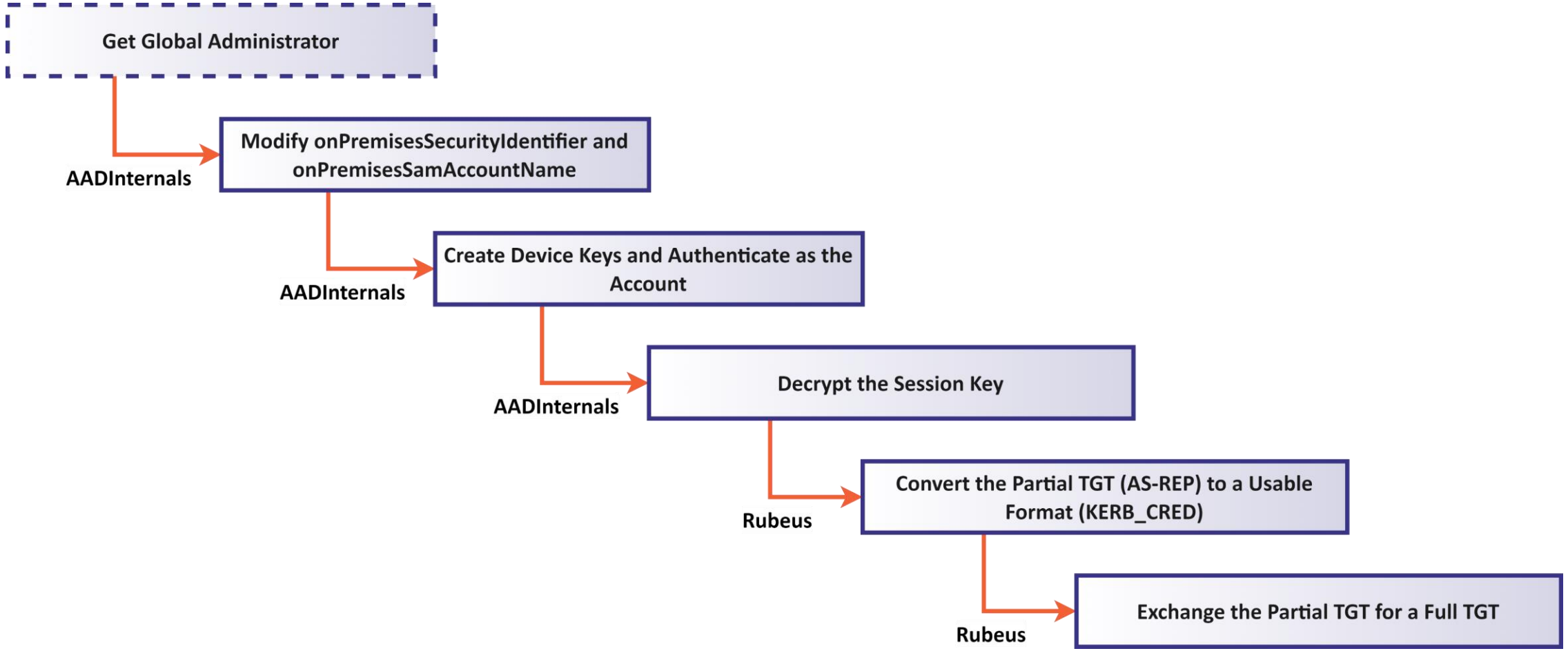
Abusing Cloud Kerberos Trust: Let AAD Forge the Partial TGT for Us!



Changing the Unchangeable

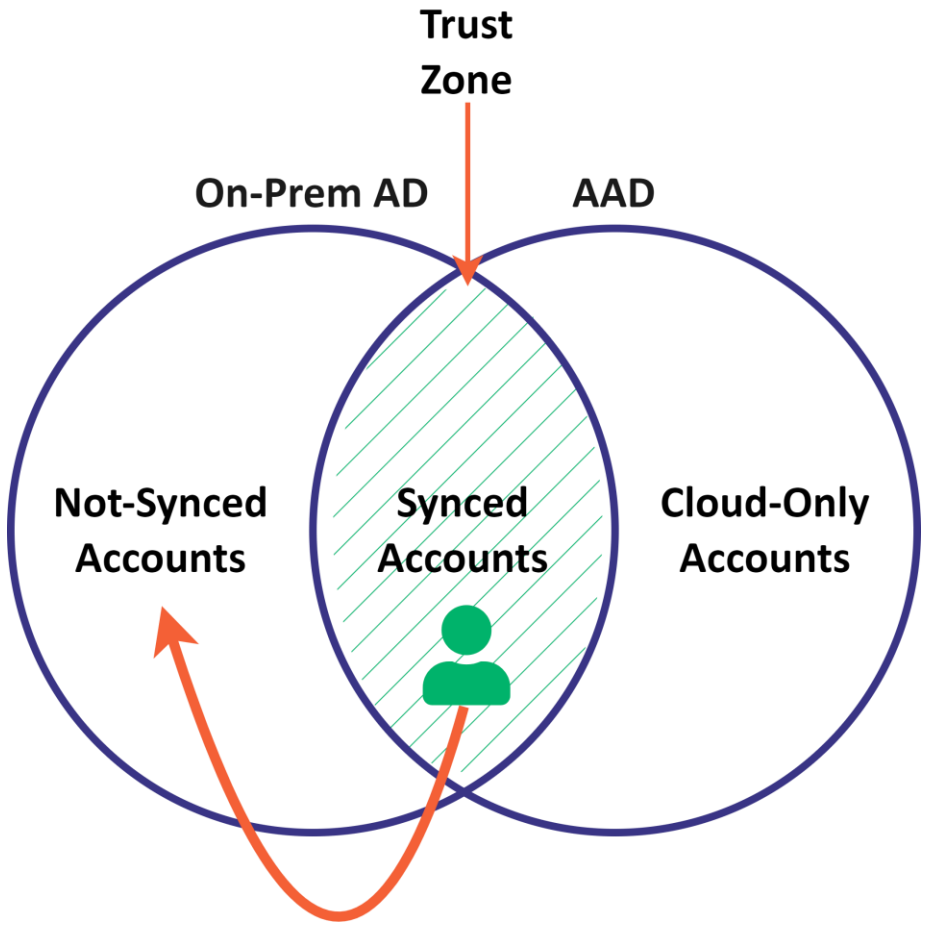
- The onPremisesSecurityIdentifier and onPremisesSamAccountName attributes cannot be modified by the Graph API
- Any account with the Global Administrator (GA) or Hybrid Identity Administrator role can modify these attributes via the sync API, normally used by the Azure AD Connect service
  - Trivial if you get GA

Weaponization

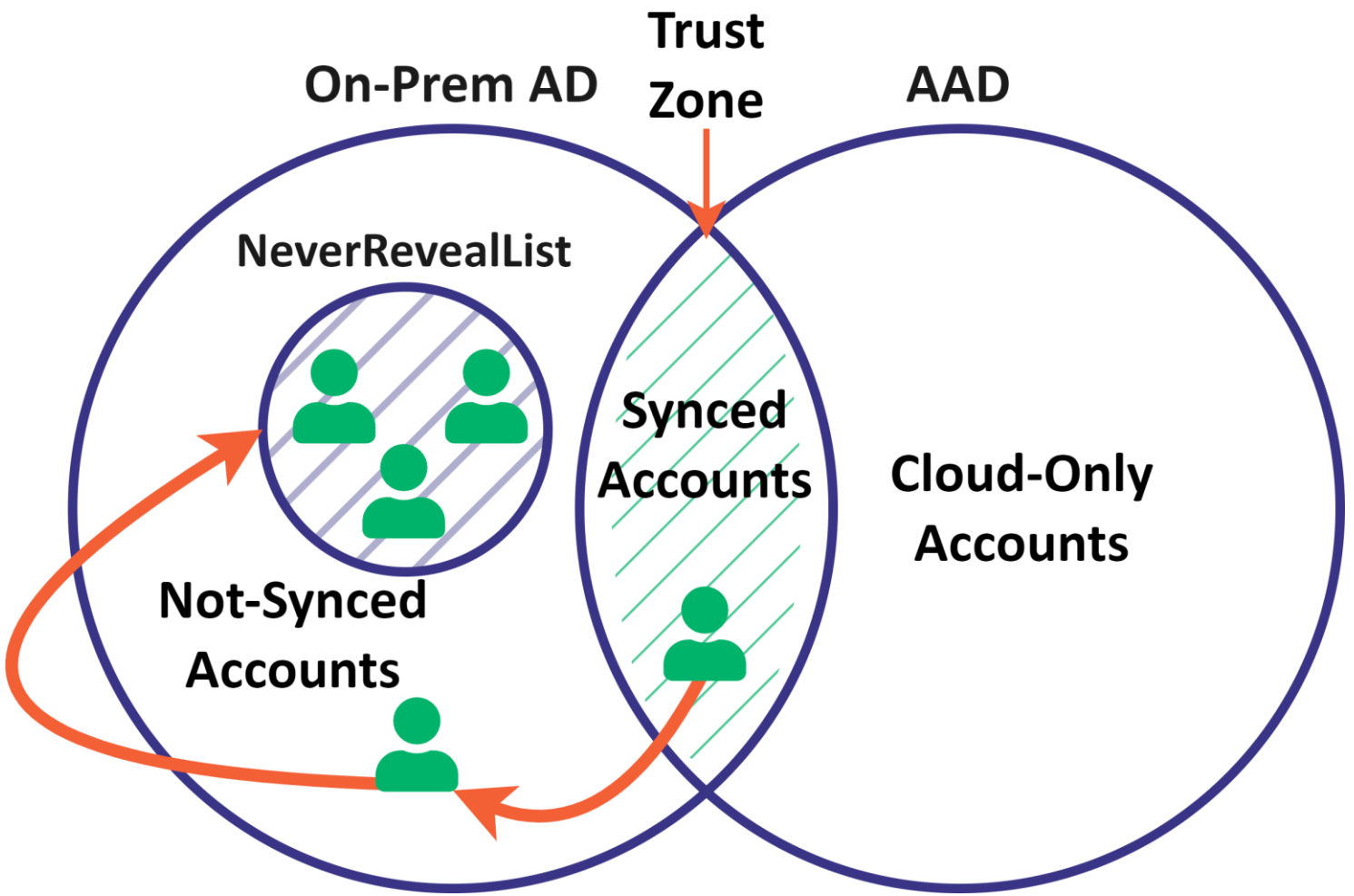



Recycle Bin
Microsoft Edge
stuff
Google Chrome
63°F
Mostly clear
Search
7:48 PM
5/15/2023

Violating the Trust Between AAD and AD



Compromising On-Premises AD



Compromising On-Premises AD

- Reminder: AAD's RODC has the following password replication policy:
  - Deny: Schema Admins, Enterprise Admins, Administrators, Cert Publishers, Domain Admins, Backup Operators, Domain Controllers, Account Operators, Server Operators
  - Allow: Domain Users
- What's missing?
  - The AAD connect service account (MSOL_XXXXXX) with DCSYNC privileges **(always there)**
  - Other *potential* gaps: On-prem Exchange, ADFS, Key Admins
  - What else?


Recycle Bin
Microsoft Edge
stuff
Google Chrome
62°F
Mostly clear
Search
8:13 PM
5/15/2023
Show desktop

Disclosure and Response

- Microsoft's Response:
  - " to accomplish this requires a certain amount of privileges"
  - "there is a public knowledge already"

Mitigation #1

- The root cause for this issue is the inclusion of the Domain Users group in AAD's RODC msDS-RevealOnDemandGroup attribute
- The better way to address this issue is to maintain a security group in on-prem AD with all the synced accounts and replace Domain Users with that group in AAD's RODC msDS-RevealOnDemandGroup attribute

Mitigation #2

- The opposite approach is adding to AAD's RODC msDS-NeverRevealGroup attribute all accounts with high privileges in on-prem AD
- Use a tool like BloodHound to identify all such accounts
  - Revoke unnecessary access while you're at it!
- Requires continuous maintenance and monitoring

Mitigation #1 + #2

- A combination of both mitigation strategies is ideal
- It explicitly allows AAD to issue on-prem TGTs only to synced accounts (mitigation #1)
- It blocks AAD from issuing on-prem TGTs to privileged on-prem accounts (mitigation #2)
- This combination addresses situations where privileged on-prem accounts are synced to AAD

Conclusion

- The boundary between on-premises and cloud becomes weaker
- Prediction: Microsoft will continue to erode any reasonable notion of a Cloud / On-Premise boundary until it is no longer considered a boundary.
- Bonus Prediction: MS will tell you to go full AAD.

Thanks To:

Leandro Cuozzo, The Kerberos Key List Attack: The return of the Read Only Domain Controllers
Microsoft, Level 400 on 425: Hello For Business and Cloud Kerberos
Dr. Nestori Syynimaa, @DrAzureAD, AADInternals
MSRC

www.specterops.io
@hotnops @elad_shamir
info@specterops.io