Active directory: How to change a weak point into a leverage for security monitoring
Vincent LE TOUX – ENGIE – France
First Conference 2017 – San Juan (Puerto Rico)
June 12th (Monday) 12:00—12:45
Chapter 1
Why focusing on Active Directory?

Chapter 2
Focusing on AD vulnerabilities

Chapter 3
Monitoring the domains (that we don’t control)

Chapter 4
How to secure the domains?
About the ENGIE Context

A critical infrastructure operator (Thermic, gas, hydro, nuclear) under regulations (NERC/NIS, …)

A complex history & a decentralized culture
The group is present in 70 countries
Why focusing on Active Directory?
Does it remind something to you?

We are secured. We have big walls. Leave us alone.
Not castles from fairy tales

The reality

Corporate AD

Company bought

External companies

Active Directory inside

Active Directory inside

Business as usual

Merger

Active Directory inside

Critical application connected to AD

Join Venture

Reorganization (never completed)

Trust everyone forgot …

Company bought
Quizz: Who can become the domain admins (or more) ?

**Built-in Administrators**

- net group "Domain Admins" %username% /DOMAIN /ADD

**Server Operators**

- C:\>sc config browser binpath= "C:\Windows\System32\cmd.exe /c net group \" Domain Admins\" %username% /DOMAIN /ADD" type= "share" group= "" depend= ""
- [SC] ChangeServiceConfig SUCCESS
- C:\>sc start browser
- [SC] StartService FAILED 1053: The service did not respond to the start or control request in a timely fashion.

**Print operators** (well, it has the right to logon to DC and discover password in batches or copy ntdis.dit backup)

**Account operators**

- net group “badgroup" %username% /DOMAIN /ADD => see slide after for the choice of the group

**Backup operators**

- Backup C:\Windows\SYSVOL\domain\Policies\{*}\MACHINE\Microsoft\Windows NT\SecEdit\GptTmpl.inf
- Restore: with [Group Membership]
- *S-1-5-32-544__Members = <etc etc etc>,*S-1-5-21-my-sid

Then DCSync krbtgt => Golden ticket => Enterprise admins (see later)
Focusing on AD vulnerabilities
Extended rights
Where are your admins?

- Extended rights can reset the password of accounts, reanimate tombstone, ... take control of accounts indirectly

(Allowed-To-Authenticate, User-Force-Change-Password, Reanimate-Tombstones, Unexpire-Password, Update-Password-Not-Required-Bit, Apply-Group-Policy, Self-Membership, Migrate SID History, Unexpire Password, DS-Replication-Get-Changes-All)

Delegation model

Root

OU-1

I got a delegation on OU-1

Users

Domain Administrators

admin1

=> Users (helpdesk, ...) can become domain admins instantly
Pass the hash / over pass the hash / pass the ticket / golden ticket / silver ticket …
Silver ticket + DCSync : being compromise without knowing it

- Detecting silver tickets requires to collect all kerberos events on ALL computers
- Silver / Golden tickets still valid if created with the old password (to avoid replication problem)

DCSync = export secrets needed to build silver tickets

Mimikatz = create / import golden / silver ticket
Old or current password

```
kerberos::golden /domain:lab.local /sid:S-1-5-21-xxx
/target: explicitdc.lab.local /service:ldap /rc4:currkey
/user:explicitdc$ /id:xxx /groups:516 /sids:S-1-5-9
/ticket:explicitdc.silver.kirbi
```

⇒ You do not need anymore an account to access the AD.
The attack is invisible using classic account supervision
Active Directory trusts

- One kerberos ticket can have a field containing a « SID History » record. Used for migration but not only (used to contain forest group membership)
- One golden / silver ticket can have a field « SID History » forged (example: forest admin SID)
- Without SID Filtering, these tickets works on other domains

No SID Filtering inside a forest…

=> One domain can compromise other domains
Account enumeration without domain access

- Abuse kerberos error code (test: Krbguess, Nmap krb5-enum-users)

```bash
root@cyclops:/pentest/assessment/KrbGuess# java -jar krbguess.jar -r mydomain -d /job/users.txt -s 192.168.5.10
KrbGuess v0.21 by Patrik Karlsson <patrick@cure.net>

[INF] Found user: matt@mydomain
[INF] Found (locked/disabled) user: guest@mydomain
[INF] Found user: alice@mydomain
[INF] Found user: bob@mydomain
[INF] Finished guessing 7 usernames in 2 seconds
```

100% of the domains vulnerable, few % of users enumerated

- Null session: authenticating to a domain with user=« » password=« » (test: rpcclient)
  - Allowed by default on Windows 2003 via MS-LSAT
  - Check Anonymous and everyone are in the group Pre-Windows 2000 Compatible Access
  - Check DsHeuristics has fLDAPBlockAnonOps enabled (forest wide setting)
  - Check the registry key TurnOffAnonymousBlock is set

10-30% of domains vulnerable, 100% of the users, including trusted domains enumerated

Consequences:
Block all the accounts if a locking policy is in place (including those in trusted domains)
Locate weak accounts and bruteforce passwords
03
Monitoring the domains (that we don’t control)
Our recipe

1) Build an « audit script » with minimal requirements (no domain admin rights, no need to run on a DC, run only once, …)

2) Easy to understand KPI

3) Sell it to the top management as « it is a 5 minute job »

4) Wait for the result and follow the deployment

Run an audit script … 
… is a « 5 minutes job »
What’s look like

### Active Directory Indicators

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Score (Max)</th>
<th>Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain Risk Level</td>
<td>80 /100</td>
<td></td>
</tr>
<tr>
<td>Stale Object</td>
<td>80 /100</td>
<td></td>
</tr>
<tr>
<td>Privileged Accounts</td>
<td>10 /100</td>
<td></td>
</tr>
<tr>
<td>Anomalies</td>
<td>36 /100</td>
<td></td>
</tr>
<tr>
<td>Trusts</td>
<td>0 /100</td>
<td></td>
</tr>
</tbody>
</table>

**=Max (all scores)**

---

**FOCUS: AD Healthcheck script V1 Global KPI**

#### SCRIPT EXECUTION

- **Evolution from 09/09/2016**
  - Total: 303 domains
  - NB domains: -3 domains
  - NB audited domains: +17 domains

#### RISK SYNTHESIS

- **Average risk score**
  - Evolution: -2

#### TRUST SID FILTERING

- **Evolution from 09/09/2016**
  - Eligible trust: +32
  - SID Filtering activated: +56
The script: example of rules

- **Stale objects**
  - User / computer not used (and never used)
  - Check for ms-DS-MachineAccountQuota = 0
  - Presence of SID History
  - Duplicate accounts ($DUPLICATE …)

- **Privileged accounts**
  - Check for flag « this account is sensitive and cannot be delegated »
  - Account « domain administrator » used
  - Owner of domain controller objects

- **Trusts**
  - SID Filtering
  - Login script from another domain

- **Anomalies**
  - Krbtgt password change
  - Presence of admincount=1 for non admins
  - GPP password
  - Password change for Smart cards
  - Root certificate weak module or algorithm

More than 50 rules in the audit script
V1: powershell ; 5 minutes per run
V2: c# ; less than 1 minute per run
Abusing trusts to discover domains

Kerberos clients can traverse a maximum of 10 trust links to locate a requested resource in another domain (source)
Limit is on UPN routing. Not trusts!
(netdom trust kz.com /domain:spat.com /namesuffixes:spat.com - source)

Technics:
1) Object type « trustedDomains »
2) msDS-TrustForestTrustInfo
3) CN=partitions,CN=Configuration
4) SID in FSP+LsaLookupSid+DSGetDC
Domain discovery in practice

With only 2 reports:
- More than 2 forests discovered
- 36 additional domains found
- Link between the 2 forests discovered
- Admin bastion discovered (without any direct trust)

Golden rule:
Assign the « discovered domains » to the AD owning the trust (and then to the BU)
Management vision about AD

Before: 90 domains

After: 300 domains

No trust with external companies

Trust with 10 unknown companies, including 2 multinationals
Management findings

- Running AD audit script is not a 5 minutes job (a 3 then 6 months project)
- Several AD (30%) without formal identified owner
- Multiply by 3 the number of AD owned
- Several trusts with external companies (without SID Filtering)
- Several GPP passwords or OU with delegation to everyone or NULL SESSION domain controllers

If one AD is compromised, it can lead to the compromise of several others
SID Filtering is a quick remediate, but works only if the corporate put pressure.
How to secure the domains?
First glance risk approach

<table>
<thead>
<tr>
<th>Group risks</th>
<th>Local risks</th>
</tr>
</thead>
<tbody>
<tr>
<td>A local domain can <strong>compromise another domain</strong> <em>(mitigation: SID Filtering)</em></td>
<td>Domain is not available (down)</td>
</tr>
<tr>
<td><strong>Domains without identified owner</strong> – nobody to manage security incidents <em>(mitigation: request script results)</em></td>
<td>Domain is compromised</td>
</tr>
<tr>
<td>Trust with an <strong>entity that we don’t control</strong> <em>(external companies, …)</em> <em>(mitigation: trust removal)</em></td>
<td>« Secure the domain » is here</td>
</tr>
</tbody>
</table>

Group risks are easier to mitigate (and they have the higher impact)
ENGIE strategy about securing Active Directory

A 3 years securisation project included in the « One Security » program
3 priorities for BU CIO and CISO defined in 2017

1. Deploy the audit script on 100% of the domains

2. Enable SID Filtering on all trusts (except migration)

3. Improve the score (min: 50/100)
## Top 5 Active directory vulnerabilities

<table>
<thead>
<tr>
<th>Check</th>
<th>Rationale</th>
<th>Vulnerable Domains</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Non admin users can add up to 10 computers to a domain</td>
<td>A User (including from trusted domains) can introduce an unsupervised workstation in the network and bypass all security policies.</td>
<td>46%</td>
</tr>
<tr>
<td>2. The « administrator » account is used at least once per month</td>
<td>Password is well known and/or stored in the registry. It can be retrieved &amp; used as a backdoor.</td>
<td>34%</td>
</tr>
<tr>
<td>3. The krbtgt password is unchanged for at least 40 days</td>
<td>It should be changed twice per month to avoid silent compromise or silent compromise using Golden ticket attacks.</td>
<td>69%</td>
</tr>
<tr>
<td>4. Null session is enabled in at least one domain controller</td>
<td>This NT4 settings can be used to enumerate all accounts without an account and bruteforce them or use this information to lock every account in the domain AND in the trusting domains.</td>
<td>28%</td>
</tr>
<tr>
<td>5. At least 2 accounts are in the domain admin groups and have a password which doesn’t expire.</td>
<td>Service accounts are far too over privileged and their password can be captured with minimal privileges.</td>
<td>66%</td>
</tr>
</tbody>
</table>
Market orientation

AD Specific solutions

Monitoring Gap

With what login is associated that IP?

Change monitoring

Attack detection

Generic solutions

splunk
Monitoring gap: no vulnerability analysis

https://github.com/ANSSI-FR/AD-control-paths - bloodhound

Bonus: who can owns the CEO account?
A possible strategy based on risks

<table>
<thead>
<tr>
<th></th>
<th>Bastion AD</th>
<th>Group application AD</th>
<th>User accounts AD</th>
<th>Others</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mitigate configuration risks</td>
<td>✔️</td>
<td>✔️</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mitigate hackers’ risk</td>
<td>✔️</td>
<td></td>
<td>✔️</td>
<td></td>
</tr>
</tbody>
</table>

Focus (and limit the budget) to high value AD – accept the risk for others
Hackers’ roadmap

Already (almost) well known

- @gentilkiwi
- mimikatz
- Golden ticket
- DCSync
  - Powershell
  - PowerSploit (Invoke-mimikatz)
  - Powershell empire
  - (Python) Responder

Not well known

- Kekeo
  - “Mimikatz 2”
  - DCSync with Netlogon RPC
- NetSync
- Aoratpw
- PKINIT Mustiness
- KerbStrom
  - DCSync / Golden ticket in c#/powershell
  - RDP attacks
- Password change with only kerberos key
  - Smart card logon with authentication in the future
  - DoS on kerberos authentication

Bypassing SID Filtering with forest trust by abusing non removed SID History
Hardening roadmap

- What AD Guys think:
  - Credential guard
  - Red forest
  - Admin bastion
  - 2 factor authentication

- What the security thinks:
  - Control the number of administrators
    - More than xxx users can become domain admin (150,000 users)

"Enabling Credential Guard on domain controllers is not supported" (source)

Google PIV / GIDS smart card

Hardening is not always a technical measure.
How much administrators have signed the admin charter?
Conclusion
Lessons learned

You can “infiltrate” a castle:
- Internally using the Active Directory
- Externally using Threat Intelligence (compromised emails or blacklist registers of internet ip)

You can quickly build a big picture:
- How much AD, the map and their risks
- Get support to remove old domains / OS

Building a « monitoring » process can be achieved at a relatively low cost
Conclusion

Many services rely on Active Directory, lots of vulnerabilities and few security.

Active Directory is an efficient way to get top management support

There is a lot of quick wins to be perceived as a solver and not a blocker by the management

It can be linked with the SOC for better monitoring of AD vulnerabilities.
Questions?

How much ponies did you see? (including this one)

Tool: http://www.pingcastle.com
95% of the total domains known in 2 months

Scripts submission flows only on management pressure

SID Filtering KPI was changed from “enabled only” to “not enable” (3 states: Yes, No, Not applicable). SID Filtering evolution is most of the time related to a direct order of the corporate.
1) **Installing a backdoor and wait for connections**
Minikatz after a login or installing a rogue security package (Note: password in clear text for RDP)

2) **Enumerate users** of Inbound trusts via LsaLookupSids

3) **Deciphering a TGS with Kerberoast**
Most vulnerable: service account with no password expiration => +20 characters recommended!
See this. 200MH/s with `hashcat+GTX1080`. From 6 months to 1 day, offline, with a 8 char password.

4) **Exploring domain configuration for vulnerabilities**
- GPP Password (almost in clear text)
- Login script hosted in other domains
- Restricted group (local admin) with Everyone or Authenticated Users or NTAUTHORITY\INTERACTIVE
- OU/container with write access to Everyone / Authenticated Users
Algorithm to know if it is active:

- SID Filtering = NA => Inbound trust or Intra forest trust
- SID Filtering Active => If forest trust and not inter forest trust => Yes ; else if quarantined domain => Yes

Enabling it:

- Forest trust: enabled by default => netdom /enableSIDHistory = NO
- Domain trust: disabled by default => netdom /quarantine = YES

- Do not enable Quarantine on a forest trust !!! (users from child domains in the forest won’t be authenticated anymore)