ADTimeline
Active Directory forensics with replication metadata

https://github.com/ANSSSI-FR/ADTimeline
Whoami

PS C:\> Get-ADUser leonard.savina -properties * | fl SID, Giv*, sn, Compa*, Dep*, Cit*, Ema*, alt*, bus*

SID : S-1-5-21-3634504526-1236365413-3814638018-3121
GivenName : Leonard
sn : Savina
Company : ANSSI
Department : CERT-FR
City : Paris
EmailAddress : leonard.savina@ssi.gouv.fr
altSecurityIdentities : {C-US/O=Twitter Inc./CN=@ldap389}
businessCategory : {Active Directory, Exchange}
ACTIVE DIRECTORY AND REPLICATION METADATA.

https://github.com/ANSSI-FR/ADTimeline
Active Directory - Overview

Active Directory is often the core of the IT infrastructure, it is installed on domain controllers (DCs) fulfilling the following roles:

- LDAP directory.
- DNS service.
- NTP service.
- Authentication services (Kerberos and NTLM).
- Windows clients configuration with GPOs.
One or more domains in one forest.

AD must be a highly available service.

Many DCs in each domain replicating the various partitions of the NTDS database.

Replication can be intra domain, intra forest or via Global Catalog (Partial Attribute Set).

A DC GUID and a USN (Update Sequence Number) identify a change in the Active Directory database.
AD replication metadata – msDS-ReplAttributeMetaData

> A constructed attribute in XML format:

```xml
<ds_repl_attr_meta_data>
  <pszAttributeName>objectCategory</pszAttributeName>
  <version>1</version>
  <ftimeLastOriginatingChange>2018-07-17T15:27:16Z</ftimeLastOriginatingChange>
  <uuidLastOriginatingDslnvocationID>d91fb4c-852c-418f-9fe2-015cc980cf38</uuidLastOriginatingDslnvocationID>
  <usnOriginatingChange>532806</usnOriginatingChange>
  <usnLocalChange>532806</usnLocalChange>
<ds_repl_attr_meta_data>
<ds_repl_attr_meta_data>
  <pszAttributeName>groupType</pszAttributeName>
  <version>1</version>
  <ftimeLastOriginatingChange>2018-07-17T15:27:16Z</ftimeLastOriginatingChange>
  <uuidLastOriginatingDslnvocationID>d91fb4c-852c-418f-9fe2-015cc980cf38</uuidLastOriginatingDslnvocationID>
  <usnOriginatingChange>532806</usnOriginatingChange>
  <usnLocalChange>532806</usnLocalChange>
<ds_repl_attr_meta_data>
</ds_repl_attr_meta_data>
```

> It gives you the time at which each attribute for a given object was last changed.

> It applies only to replicated attributes.
AD replication metadata – msDS-ReplAttributeMetaData

For each replicated attribute msDS-ReplAttributeMetaData contains:

- pszAttributeName: The attribute name.
- ftimeLastOriginatingChange: Time the attribute was last changed.
- dwVersion: Counter incremented every time the attribute is changed.
- usnOriginatingChange: USN on the originating server at which the last change to this attribute was made.
- pszLastOriginatingDsaDN: DC on which the last change was made to this attribute.
- uuidLastOriginatingDsaInvocationID: ID corresponding to pszLastOriginatingDsaDN.
- usnLocalChange: USN on the destination server (the server your LDAP bind is made) at which the last change to this attribute was applied.
AD replication metadata– msDS-RepIValueMetaData

Replication metadata for *linked attributes*:

Pairs of attributes in which the system calculates the values of one attribute (the *back link* e.g. `MemberOf`) based on the values set on the other attribute (the *forward link* e.g. `Member`) throughout the forest.

In the case of group objects, the member attribute has the same information as `msDS-RepIAttributeMetaData` and in addition:

- `pszObjectDn` : The group member DistinguishedName.
- `ftimeCreated` : Contains the time the member was added in the group.
- `ftimeDeleted` : Contains the time the member was removed from the group.
# AD replication metadata – Tools

> **With command line:**

```bash
PS Z:\> repadmin /showbjmeta rwdc.labo.local "CN=HR_RW,DC=labo,DC=local"
```

<table>
<thead>
<tr>
<th>Num</th>
<th>Attribute</th>
<th>Heure dern. mod.</th>
<th>DSA source</th>
<th>USN loc</th>
<th>USN org</th>
<th>Ver</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>objectClass</td>
<td></td>
<td>SIEGE\RMDC</td>
<td>532806</td>
<td>532806</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>cn</td>
<td></td>
<td>SIEGE\RMDC</td>
<td>532806</td>
<td>532806</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>description</td>
<td></td>
<td>SIEGE\RMDC</td>
<td>532806</td>
<td>532806</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>instanceType</td>
<td></td>
<td>SIEGE\RMDC</td>
<td>532806</td>
<td>532806</td>
<td>1</td>
</tr>
</tbody>
</table>

> **With Powershell 4.0+:**

```powershell
PS Z:\> Get-ADReplicationAttributeMetadata "CN=HR_RW,DC=labo,DC=local" -Server rwdc.labo.local | select -last 1
```

<table>
<thead>
<tr>
<th>AttributeName</th>
<th>AttributeValue</th>
<th>FirstOriginatingCreateTime</th>
<th>IsLinkValue</th>
<th>LastOriginatingChangeDirectoryServerIdentity</th>
<th>LastOriginatingChangeDirectoryServerInvocationId</th>
<th>LastOriginatingChangeGens</th>
<th>LastOriginatingDeleteTime</th>
<th>LocalChangeGens</th>
<th>Object</th>
<th>Server</th>
<th>Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>member</td>
<td>member</td>
<td>2018-07-17 17:30:14</td>
<td></td>
<td>true</td>
<td>CN=NTDS Settings,CN=RWDC,DC=labo,DC=local</td>
<td>532855</td>
<td>17:44:03</td>
<td>2</td>
<td>6</td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>
AD replication metadata – Existing work

> Pierre Audonnet:

> Gregory Lucand (FR):
https://adds-security.blogspot.com

> Will Schroeder:
https://harmj0y.net/blog/defense/hunting-with-active-directory-replication-metadata
https://github.com/ANSSI-FR/ADTimeline

THE ADTIMELINE TOOL
Objects considered of interest are gathered by the script.

For each object replication metadata is retrieved: \textit{msDS-ReplAttributeMetaData} for every objectclass. For group objectclass, \textit{msDS-ReplValueMetaData} is also retrieved.

Generate a timeline by sorting replication metadatas by \textit{ftimeLastOriginatingChange}.

Tool has an online and offline mode.
ADTimeline – Files generated

Timeline in CSV format (metadata + some attributes): `Import-Csv –delimiter ‘;’`

All objects retrieved via LDAP and their attributes values (ADOObjects.xml) and all objects retrieved via Global Catalog (GCADOObjects.xml) : `Import-CliXML`. 

Demo 1 – Mail exfiltration

Attack scenario:

> Attacker grants a user mailbox read rights on a database and the ability to export emails as a PST archive.

> Attacker searches with that user for valuable intel to exfiltrate by browsing employees webmail.

> Attacker exfiltrates with that user interesting emails as a PST archive with `New-MailboxExportRequest` (Hacking Team breach: [http://pastebin.com/raw/0SNSvyjJ](http://pastebin.com/raw/0SNSvyjJ))
Propriétaires créateurs de la stratégie de groupe
  Group
  S-1-5-21-3634504526-1236365413-3814638018-520  Mandatory
group, Enabled by default, Enabled group
LABO\Organization Management
  Group
  S-1-5-21-3634504526-1236365413-3814638018-1105 Mandatory
group, Enabled by default, Enabled group
LABO\Administrateurs du schéma
  Group
  S-1-5-21-3634504526-1236365413-3814638018-518  Mandatory
group, Enabled by default, Enabled group
LABO\Administrateurs de l'entreprise
  Group
  S-1-5-21-3634504526-1236365413-3814638018-519  Mandatory
group, Enabled by default, Enabled group
LABO\Import-ExportMBX
  Group
  S-1-5-21-3634504526-1236365413-3814638018-1190 Mandatory
group, Enabled by default, Enabled group
Authentication authority asserted identity
  Well-known group S-1-18-1  Mandatory
group, Enabled by default, Enabled group
LABO\Groupe de réplication dont le mot de passe RODC est refusé Alias
  S-1-5-21-3634504526-1236365413-3814638018-572  Mandatory
group, Enabled by default, Enabled group, Local Group
Mandatory Label\High Mandatory Level
  S-1-16-12288

PS C:\Windows\system32>
ADTimeline - Processing the results

- Search for suspicious attribute modifications: 
  NTSecurityDescriptor, SIDHistory, defaultSecurityDescriptor, 
  UserAccountControl, Searchflags…
- Objects deletion (Tombstone).
- User accounts added and removed from groups.
- Inconsistency in the timeline (USN/ftimeLastOriginatingChange, 
  dwVersion, WhenCreated).

When a suspicious behavior is spotted, retrieve the DCs event logs (pszLastOriginatingDsaDN Domain Controller backup).
## ADTimeline - Objects considered of interest

<table>
<thead>
<tr>
<th>Objects in the domain partition</th>
<th>Objects in other partitions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Domain root and objects located directly under the root.</td>
<td>Domain roots located in the AD forest.</td>
</tr>
<tr>
<td>Objects protected by the SDProp process</td>
<td>Domain trusts and CertificationAuthority objects.</td>
</tr>
<tr>
<td>The Pre Windows 2000 compatible access, Cert publishers, GPO creator owners and DNS Admins groups.</td>
<td>Class Schema objects and attributes with particular SearchFlags (Do not audit or confidential).</td>
</tr>
<tr>
<td>Objects having an ACE on the domain root.</td>
<td>Domain controllers (Computer objects, ntdsdsa and server objects).</td>
</tr>
<tr>
<td>Deleted users (i.e. tombstoned) and dynamic objects.</td>
<td>DNS zones.</td>
</tr>
<tr>
<td>Organizational Units.</td>
<td>Accounts with suspicious SIDHistory (scope is forest wide).</td>
</tr>
<tr>
<td>Existing and deleted Group Policy objects.</td>
<td>AD Site, the directory service and RID manager objects.</td>
</tr>
<tr>
<td>Objects under the System container.</td>
<td>Extended rights and Cross Reference containers.</td>
</tr>
<tr>
<td>Objects with Kerberos delegation enabled.</td>
<td>Exchange RBAC roles and accounts assigned to a role.</td>
</tr>
<tr>
<td>Kerberoastable and AS-REP roastable accounts.</td>
<td>Exchange mail flow and storage configuration objects.</td>
</tr>
<tr>
<td>Custom groups which have to be manually defined.</td>
<td>Deleted objects under the configuration partition.</td>
</tr>
</tbody>
</table>
ADTimeline - Using offline and online mode.

> Online mode: Launch on a privileged access workstation having ADDS Powershell module installed and with a domain admin account (*tombstone* read rights). Works with standard user also.

> Offline mode: In case the analyst has to process a disk image or a NTDS backup/snapshot. Mount the NTDS file with *dsamain.exe* (part of ADLDS role) on an analysis machine with ADDS Powershell module installed.
Demo 2 – Mimikatz DCShadow

Attack scenario:

> *PhoneNumber* attribute modification on admin accounts in order to bypass the 2FA authentication setup by the security team on a critical application. First factor being AD password, second being security code sent by SMS.

> Use of *Mimikatz DCShadow* in order to bypass SIEM alerting (Windows security event logs) and replication metadata tampering in order to slow down investigation.
AD replication metadata vs security event logs

> Replication metadata **IS NOT AN EXCUSE NOT TO** centralize, store and analyse your AD security event logs!

> Perimeter:

  Metadata: Concerns every objectclass but only replicated attributes.
  Event logs: Depends on your audit policy.
AD replication metadata vs security event logs

> Centralization:
Metadata: Replicated and stored in the NTDS database of every DC.
Event logs: Setup your centralized windows event log management (http://aka.ms/WEF)

> History:
Metadata: Data since your domain creation but only the last modification of each replicated attribute.
Event logs: Depends on your event logs retention strategy.

> Data available:
Metadata: You do not know who made the modification and the attribute value before vs after.
Event logs: All the data required is present.

> Ability to tamper the data:
Metadata: Yes (e.g. Mimikatz DCShadow)
Event logs: Yes (e.g. DanderSpritz Eventlogedit)
QUESTIONS?

Additional resources:
Hideaki Ihara from JSOC
http://port139.hatenablog.com/