Toxic Waste Removal for Active Directory

Quickly Identifying and Safely Removing Dangerous Legacy Permissions
HELLO!
I am Andy Robbins
Adversary Resilience Lead at SpecterOps
BloodHound co-creator and developer, Red Teamer
You can find me at @_wald0
Outline

▪ Prior Work
▪ What’s the Problem?
▪ Attack Taxonomy
▪ How to Quickly Identify Dangerous Permissions
▪ Two Ideas for Identifying Legacy Permissions
▪ Conclusion and Future Work
Prior Work
Chemins de contrôle en environnement Active Directory

Chacun son root, chacun son chemin

Lucas Bouillot, Emmanuel Gras

Agence Nationale de la Sécurité des Systèmes d’Information

SSTIC 2014 - 4 juin 2014

https://www.sstic.org/2014/presentation/chemins_de_controle_active_directory/
ACTIVE DIRECTORY BACKDOORS: Myth or Reality
BTA: an open source framework to analyse AD

Philippe Biondi, Joffrey Czarny — Airbus Group Innovations
BlackHat Arsenal — 2015-08-06
Gathering AD Data with the Active Directory PowerShell Module

Microsoft provided several Active Directory PowerShell cmdlets with Windows Server 2008 R2 (and newer) which greatly simplify tasks which previously required putting together lengthy lines of code involving ADSI. On a Windows client, install the...
What’s the Problem?
What’s the Problem?

- Out of the box, Active Directory (AD) is already a sophisticated, complicated directory service.
- Over time, the complexities of intertwining permissions and privileges become unwieldy.
- Software installers and admins grant themselves dangerous permissions. This “misconfiguration debt” degrades the organization’s security posture.
- Removing dangerous permissions can be very risky.
Defenders think in lists. Attackers think in graphs. As long as this is true, attackers win.

John Lambert, GM, Microsoft Threat Intelligence Center
Attack Taxonomy
Attack Taxonomy

- All securable objects in AD have a Security Descriptor.
- The Security Descriptor has a Discretionary Access Control List (DACL) and a System Access Control List (SACL)
- The DACL is populated by Access Control Entries (ACEs), which define who is allowed or denied permissions on the object.
## Advanced Security Settings for Administrator

**Owner:** Domain Admins (CONTOSO\Domain Admins)  [Change]

### Permissions Tab

For additional information, double-click a permission entry. To modify a permission entry, select the entry and click Edit (if available).

**Permission entries:**

<table>
<thead>
<tr>
<th>Type</th>
<th>Principal</th>
<th>Access</th>
<th>Inherited from</th>
<th>Applies to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow</td>
<td>Domain Admins (CONTOSO\Domain Admins)</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>Authenticated Users</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>SYSTEM</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>ENTERPRISE DOMAIN CONT\Domain Admins</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>ENTERPRISE DOMAIN CONT\Computers</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>ENTERPRISE DOMAIN CONT\Computers</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>ENTERPRISE DOMAIN CONT\Computers</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>ENTERPRISE DOMAIN CONT\Computers</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>ENTERPRISE DOMAIN CONT\Computers</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>ENTERPRISE DOMAIN CONT\Computers</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>SELF</td>
<td>Special</td>
<td>DC=contoso,DC=local</td>
<td>Descendant Group objects</td>
</tr>
<tr>
<td>Allow</td>
<td>Pre-Windows 2000 Compat\Computers</td>
<td>Special</td>
<td>DC=contoso,DC=local</td>
<td>Descendant Group objects</td>
</tr>
<tr>
<td>Allow</td>
<td>Pre-Windows 2000 Compat\Computers</td>
<td>Special</td>
<td>DC=contoso,DC=local</td>
<td>Descendant Group objects</td>
</tr>
<tr>
<td>Allow</td>
<td>Pre-Windows 2000 Compat\Computers</td>
<td>Special</td>
<td>DC=contoso,DC=local</td>
<td>Descendant Group objects</td>
</tr>
</tbody>
</table>

- **Add**, **Remove**, **View**, **Disable inheritance**, **Restore defaults**, **OK**, **Cancel**, **Apply**

---

13
Advanced Security Settings for Administrator

Owner: Domain Admins (CONTOSO\Domain Admins)  Change

<table>
<thead>
<tr>
<th>Type</th>
<th>Principal</th>
<th>Access</th>
<th>Inherited from</th>
<th>Applies to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow</td>
<td>Domain Admins (CONTOSO\Domain Admins)</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>Authenticated Users</td>
<td>Special</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>SYSTEM</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>ENTERPRISE DOMAIN CONT...</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>ENTERPRISE DOMAIN CONT...</td>
<td>DC=cortoso,DC=local</td>
<td>Descendant Group objects</td>
<td></td>
</tr>
<tr>
<td>Allow</td>
<td>ENTERPRISE DOMAIN CONT...</td>
<td>DC=cortoso,DC=local</td>
<td>Descendant Group objects</td>
<td></td>
</tr>
<tr>
<td>Allow</td>
<td>ENTERPRISE DOMAIN CONT...</td>
<td>DC=cortoso,DC=local</td>
<td>Descendant Group objects</td>
<td></td>
</tr>
<tr>
<td>Allow</td>
<td>SELF</td>
<td>DC=cortoso,DC=local</td>
<td>Descendant Group objects</td>
<td></td>
</tr>
<tr>
<td>Allow</td>
<td>Pre-Windows 2000 Compatibility Pack</td>
<td>Special</td>
<td>DC=cortoso,DC=local</td>
<td>Descendant Group objects</td>
</tr>
<tr>
<td>Allow</td>
<td>Pre-Windows 2000 Compatibility Pack</td>
<td>Special</td>
<td>DC=cortoso,DC=local</td>
<td>Descendant InetOrgPerson objects</td>
</tr>
<tr>
<td>Allow</td>
<td>Pre-Windows 2000 Compatibility Pack</td>
<td>Special</td>
<td>DC=cortoso,DC=local</td>
<td>Descendant User objects</td>
</tr>
</tbody>
</table>
### Advanced Security Settings for Administrator

**Owner:** Domain Admins (CONTOSO\Domain Admins)  Change

#### Permissions

For additional information, double-click a permission entry. To modify a permission entry, select the entry and click Edit if available.

#### Permission entries:

<table>
<thead>
<tr>
<th>Type</th>
<th>Principal</th>
<th>Access</th>
<th>Inherited from</th>
<th>Applies to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow</td>
<td>Domain Admins (CONTOSO...)</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>Authenticated Users</td>
<td>Special</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>SYSTEM</td>
<td>Full control</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Allow</td>
<td>ENTERPRISE DOMAIN CONT...</td>
<td>DC=contoso,DC=local</td>
<td>None</td>
<td>Descendant Computer objects</td>
</tr>
<tr>
<td>Allow</td>
<td>ENTERPRISE DOMAIN CONT...</td>
<td>DC=contoso,DC=local</td>
<td>None</td>
<td>Descendant Group objects</td>
</tr>
<tr>
<td>Allow</td>
<td>ENTERPRISE DOMAIN CONT...</td>
<td>DC=contoso,DC=local</td>
<td>None</td>
<td>Descendant User objects</td>
</tr>
<tr>
<td>Allow</td>
<td>SELF</td>
<td>DC=contoso,DC=local</td>
<td>None</td>
<td>Descendant InetOrgPerson...</td>
</tr>
<tr>
<td>Allow</td>
<td>Pre-Windows 2000 Compat...</td>
<td>DC=contoso,DC=local</td>
<td>None</td>
<td>Descendant InetOrgPerson...</td>
</tr>
<tr>
<td>Allow</td>
<td>Pre-Windows 2000 Compat...</td>
<td>DC=contoso,DC=local</td>
<td>None</td>
<td>Descendant Group objects</td>
</tr>
<tr>
<td>Allow</td>
<td>Pre-Windows 2000 Compat...</td>
<td>DC=contoso,DC=local</td>
<td>None</td>
<td>Descendant User objects</td>
</tr>
</tbody>
</table>

[Add] [Remove] [View]

[Restore defaults] [OK] [Cancel] [Apply]
Dangerous Permissions Against Users

- Two basic attacks: reset a user’s password, or perform a targeted kerberoasting attack*
- Two specific rights: ForceChangePassword, and GenericWrite
- FullControl, WriteDACL, WriteOwner, and AllExtendedRights will get us there too.

*see http://www.harmj0y.net/blog/activedirectory/targeted-kerberoasting/
Dangerous Permissions Against Groups

- One attack: add other principals to that group, then use the permissions of that group to continue the attack path.
- One specific right: AddMembers
- FullControl, WriteDACL, WriteOwner, and AllExtendedRights will get us there too.
Dangerous Permissions Against Domain Objects

- One domain object specific attack: DCSync
- Two specific rights are needed: DSGetReplicationChanges and DSGetReplicationChanges-All
- FullControl, WriteDACL, WriteOwner, and AllExtendedRights will get us there too.
We’ve Barely Scratched The Surface

- Will Schroeder (@harmj0y) has added abuse functions to PowerView for each of these attack primitives
- See the talk by me, Will Schroeder and Rohan Vazarkar at DerbyCon 7.0 for more in-depth information and attack demonstrations: https://www.youtube.com/watch?v=z8thoG7gPd0
Quickly Identify Dangerous Permissions
Quickly Identify Dangerous Permissions

- We need: security group memberships, user session information, local admin group memberships, and securable object ACEs
- By default, ANY domain user can collect this data without any special privileges
- SharpHound makes collection easy and fast
Collect the enumeration tool

Download SharpHound:
https://github.com/BloodHoundAD/BloodHound/tree/master/Ingestors
Use SharpHound to collect the data

```
PS C:\Users\dfm\Desktop\test> .\SharpHound.exe --CompressData
Initializing BloodHound
Starting enumeration for testlab.local
Status: 25 objects enumerated (+25 1.086957/s --- Using 35 MB RAM )
Finished enumeration for testlab.local in 00:00:23.4276987
2 hosts failed ping. 0 hosts timedout.
Compressing data to .\BloodHound_20170907131224238.zip
PS C:\Users\dfm\Desktop\test> ls
Directory: C:\Users\dfm\Desktop\test

Mode LastWriteTime Length Name
---- ------------- ------ ----
-a-- 9/7/2017 1:12 PM 2081 BloodHound.bin
-a-- 9/7/2017 1:12 PM 1117 BloodHound_20170907131224238.zip
-a-- 9/7/2017 1:12 PM 2696 group_membership.csv
-a-- 9/7/2017 1:12 PM 401 local_admins.csv
-a-- 9/5/2017 3:00 PM 536576 SharpHound.exe
-a-- 9/7/2017 1:12 PM 187 trusts.csv
```

https://blog.cptjesus.com/posts/newbloodhoundingestor
Enumerate Attack Paths

- Run SharpHound from a domain-joined computer.
- To collect object control data, SharpHound requires LDAP access to at least one domain controller per domain.
Collect the Analysis Tools

Download Neo4j Server: https://neo4j.com/download/
Follow the setup instructions at: https://github.com/BloodHoundAD/BloodHound/wiki/Getting-started or https://www.youtube.com/edit?o=U&video_id=o22EMEbrNk
BloodHound Interface Demonstration

https://youtu.be/BAEfEdNWij0
Two Ideas for Identifying Legacy Permissions
Identifying Legacy Permissions

- Removing permissions can be risky
- We need confidence we aren’t going to break something
- We need assurance that applications won’t silently fail and affect business due to permissions we removed
- What follows are two ideas we believe can be effective, which we’ve tested in a lab but not in production (yet!)
Method One: Comparative Analysis

- Most applications do not remove unneeded/legacy permissions during updates.
- Compare permissions granted by legacy installers with those granted by newest installer.
- Verify all application instances are running latest version.
- Mark permissions granted by legacy installer as candidates for removal.
Method One: Comparative Analysis

- In separate AD labs, install the up-to-date version of the software in question, as well as the original version installed in your real environment
- Use BloodHound to compare the outbound object control granted by the different installers
- Don’t forget to target DA-equivalent principals, as outlined by Sean Metcalf at adsecurity.org
Transitive Outbound Control: Exchange 2003
Transitive Outbound Control: Exchange 2007
Transitive Outbound Control: Exchange 2007 SP1
Transitive Outbound Control: Exchange 2010
Transitive Outbound Control: Exchange 2013
Transitive Outbound Control: Exchange 2016
## Object Outbound Control Metrics - Exchange Server

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Direct control of Domain Admins</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>Direct Control of DA-Equivalent Principals</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Simple Path to Domain Admin</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Reset Most User Passwords</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Add Members to Most Groups</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Method One: Comparative Analysis

- Note: this information is not comprehensive for every minor update/service pack for Exchange Server.
- Your environment, and several environments we’ve been in, grant Exchange servers even MORE permissions.
- Bottom line: if the Exchange 2016 installer doesn’t grant the permissions, your Exchange 2016 servers probably don’t need them.
- Use BloodHound to see just how bad the situation is in your own environment.
**Important Caveat!**

- The previous chart *does not* account for Exchange split permissions model, introduced with Exchange Server 2010.
- If you’re running split permissions, I would still **strongly advise** you to enumerate dangerous permissions and attack paths.
- Microsoft’s *officially supported remediation guidance* is to run the following:
  - `setup.com /PrepareAD /ActiveDirectorySplitPermissions:true`
- In Thank you Josh M. Bryant (@FixTheExchange) at Microsoft Consulting Services for this information!
Method Two: Granted vs Requested Permissions

- Use event logs to compare requested rights vs granted rights. Remove unused rights.
- Strategically place SACL ACEs on the right objects.
- Defenders can already use these events to detect attackers, we can use them to determine whether the rights are ever legitimately used.
<table>
<thead>
<tr>
<th>Dangerous Permission</th>
<th>Associated Event IDs</th>
</tr>
</thead>
<tbody>
<tr>
<td>GenericAll</td>
<td>4662</td>
</tr>
<tr>
<td>GenericWrite</td>
<td>4662</td>
</tr>
<tr>
<td>DCSync*</td>
<td>4662</td>
</tr>
<tr>
<td>WriteOwner</td>
<td>4662</td>
</tr>
<tr>
<td>WriteDACL</td>
<td>4662, 4670</td>
</tr>
<tr>
<td>ForceChangePassword</td>
<td>4724</td>
</tr>
<tr>
<td>AddMember</td>
<td>4662, 4728</td>
</tr>
</tbody>
</table>

*See [https://adsecurity.org/?p=1729](https://adsecurity.org/?p=1729) for more info and in-depth detection guidance*
Event Collection

- We’re going to set up 4662 collection on specific principals.
- We’ll limit the scope to only those principals with dangerous permissions against them, and only trigger the event when the relevant principal requests permissions against the object.
- In other words, only generate the event when an Exchange Server requests permissions against a Domain Admin or other critical object.
Advanced Security Settings for Administrator

Owner: Domain Admins (CONTOSO\Domain Admins)  Change

Permissions  Auditing  Effective Access

For additional information, double-click an audit entry. To modify an audit entry, select the entry and click Edit (if available).

Auditing entries:

<table>
<thead>
<tr>
<th>Type</th>
<th>Principal</th>
<th>Access</th>
<th>Inherited from</th>
<th>Applies to</th>
</tr>
</thead>
<tbody>
<tr>
<td>Succ.</td>
<td>Exchange Servers (CONTOSO...)</td>
<td>Full control</td>
<td>None</td>
<td>This object and all descendant...</td>
</tr>
<tr>
<td>Succ.</td>
<td>Everyone</td>
<td>Special</td>
<td>None</td>
<td>This object only</td>
</tr>
<tr>
<td>Succ.</td>
<td>Everyone</td>
<td></td>
<td>DC=contoso,DC=local</td>
<td>Descendant Organizational Unit...</td>
</tr>
<tr>
<td>Succ.</td>
<td>Everyone</td>
<td></td>
<td></td>
<td>Descendant Organizational Unit...</td>
</tr>
</tbody>
</table>

Add  Remove  Edit  Restore defaults  Disable inheritance

OK  Cancel  Apply
**Event Collection**

- This will start generating 4662 events any time an Exchange server requests access to the Administrator user.
- We can collect and parse those events with `Get-ADAuditAccess*` by Ben Wilkinson: [https://gallery.technet.microsoft.com/scriptcenter/Auditing-Directory-Service-53574749](https://gallery.technet.microsoft.com/scriptcenter/Auditing-Directory-Service-53574749)

*Find my modified version used for this demo here: [https://github.com/andyrobbins/Get-ADAuditAccess](https://github.com/andyrobbins/Get-ADAuditAccess)*
Event Collection

- Collecting these events at scale is beyond the scope of this talk.
- Check out these resources for getting started with event collection at scale:
Event Collection

- Allow enough time for typical Exchange operations.
- This may be hours, days, or weeks depending on the size of your environment.
- Import the relevant requested accesses into the graph and compare requested accesses vs granted permissions.
<table>
<thead>
<tr>
<th>Dangerous Permission</th>
<th>Corresponding Requested Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>GenericAll</td>
<td>Combination of 13 accesses, including Generic Write, All Extended Rights, Write DACL, and Write Owner.</td>
</tr>
<tr>
<td>GenericWrite</td>
<td>Combination of 3 accesses, including Write Property and Read Control</td>
</tr>
<tr>
<td>DCSync*</td>
<td>DS Replication Get Changes and DS Replication Get Changes All</td>
</tr>
<tr>
<td>WriteOwner</td>
<td>Write Owner</td>
</tr>
<tr>
<td>WriteDACL</td>
<td>Write DACL</td>
</tr>
<tr>
<td>ForceChangePassword</td>
<td>&lt;Generates 4724 events&gt;</td>
</tr>
<tr>
<td>AddMember</td>
<td>&lt;Generates 4728 events&gt;</td>
</tr>
</tbody>
</table>

### Event Collection

**Administrator: Windows PowerShell**

```powershell
PS C:\Users\Administrator\Desktop> Get-ADAuditAccess
>> -ComputerName WIN-2012-DC-001 -DaysAgo 30 | Select -First 10
>>
>> ComputerName
Using provided ComputerNames
WIN-2012-DC-001

<table>
<thead>
<tr>
<th>TimeCreated</th>
<th>4/25/2018 3:23:51 PM</th>
</tr>
</thead>
<tbody>
<tr>
<td>SecurityID</td>
<td>S-1-5-21-3130263747-879503487-3624965923-500</td>
</tr>
<tr>
<td>AccountName</td>
<td>Administrator</td>
</tr>
<tr>
<td>AccountDomain</td>
<td>CONTOSO</td>
</tr>
<tr>
<td>LogonID</td>
<td>0x38287</td>
</tr>
<tr>
<td>ObjectServer</td>
<td>DS</td>
</tr>
<tr>
<td>ObjectType</td>
<td>Domain-DNS</td>
</tr>
<tr>
<td>ObjectName</td>
<td>DC=contoso,DC=local</td>
</tr>
<tr>
<td>HandleID</td>
<td>0x0</td>
</tr>
<tr>
<td>OperationType</td>
<td>Object Access</td>
</tr>
<tr>
<td>Accesses</td>
<td>Read Property</td>
</tr>
<tr>
<td>AccessedProp</td>
<td>Public-Information</td>
</tr>
</tbody>
</table>

TimeCreated    4/25/2018 3:23:51 PM
SecurityID     S-1-5-21-3130263747-879503487-3624965923-500
AccountName    Administrator
AccountDomain  CONTOSO
LogonID        0x38287
ObjectServer   DS
ObjectType     Domain-DNS
ObjectName     DC=contoso,DC=local
HandleID       0x0
OperationType  Object Access
Accesses       Read Property
AccessedProp   Object-Class
```
Method Two: Granted vs Requested Permissions

- Parse the CSVs and add the relevant dangerous permissions that are actually requested into the graph.
- Compare the granted vs requested permissions, delete any granted, non-requested permissions.
- Continue to monitor the affected objects in case of a silent failure in the future.
- We’ll release the cypher ingestion queries and relevant queries you can run in BloodHound in a future blog post soon!
Conclusion and Future Work
Conclusion

- Object-control attack paths in AD are extremely common
- Using an attack graph brings the most important permissions into immediate focus
- We can use existing, built-in features in Windows and AD to identify dangerous permissions we can safely remove without breaking anything
Future Work

- Make analysis much easier by automating much of the process discussed in this talk
- Place even more specific SACL ACEs to reduce number of events generated during analysis period
- Continue research on abusable ACEs in AD and Windows
- Expand the attack graph to include dangerous ACEs on host-based objects
THANKS!

- specterops.io
- @SpecterOps
- @wald0
- Join the BloodHound Slack: https://bloodhoundgang.herokuapp.com