Search Search

Active Directory Security

Active Directory, Security, PowerShell, Tech Notes, & Geek Trivia…

- Home
- About
- AD Reading Library
- Contact
- Mimikatz
- Presentations
- Schema Versions
- Security Resources
- SPNs

« Sneaky Active Directory Persistence Tricks

Oct 14

The Most Common Active Directory Security Issues and What You Can Do to Fix Them

Categories:

ActiveDirectorySecurity, Microsoft Security, Technical Reference

by Sean Metcalf

The past couple of years of meeting with customers is enlightening since every environment, though unique, often has the same issues. These issues often boil down to legacy management of the enterprise Microsoft platform going back a decade or more.

I spoke about Active Directory attack and defense at several security conferences this year including BSides, Shakacon, Black Hat, DEF CON, and DerbyCon. These talks include information about how to best protect the Active Directory enterprise from the latest, and most successful, attack vectors.

While the threats have changed over the past decade, the way systems and networks are managed often have not. We continue with the same operations and support paradigm despite the fact that internal systems are compromised regularly. We must embrace the new reality of “Assume Breach.”

Assume breach means that we must assume that an attacker has control of a computer on the internal network and can access the same resources the users who have recently logged on to that computer has access to.
Note that when I describe risks and mitigations of Active Directory,this includes overall enterprise configuration.

Here are some of the biggest AD security issues (as I see them). This list is not complete, but reflects common enterprise issues.

Thinking an Active Directory domain is the security boundary.

This impacts the design of security controls and may introduce vulnerabilities. An Active Directory forest may be designed with multiple domains to mitigate certain security concerns but won’ t actually mitigate them due to how domain trusts in the forest work. If a group has security requirements for their own domain due to security reasons, it is likely they really need their own forest. Before going with that design change, make sure the additional complexity of deployment and management is considered.

Microsoft describes what an AD forest:

> A forest is a complete instance of Active Directory. Each forest acts as a top-level container in that it houses all domain containers for that particular Active Directory instance. A forest can contain one or more domain container objects, all of which share a common logical structure, global catalog, directory schema, and directory configuration, as well as automatic two-way transitive trust relationships. The first domain in the forest is called the forest root domain. The name of that domain refers to the forest, such as Nwtraders.msft. By default, information in Active Directory is shared only within the forest. In this way, the forest is a security boundary for the information that is contained in that instance of Active Directory.

Deploying systems with default settings.

Often referred to as the "tyranny of the default". Don't assume the default security settings are the most secure. Microsoft focuses on compatibility across products including older ones. This means you need to tweak security settings to improve security. Newer versions of Windows typically have more secure default settings and services with limited permissions (and auto-run status).
Legacy security configuration remains one of the biggest security issues on modern networks.

Too many Domain Admins.

Active Directory administration is typically performed by a small number of people. The number of Domain Admins (DA) typically exceeds the number of actual AD admins. Domain Admins members have FULL administrative rights to all workstations, servers, Domain Controllers, Active Directory, Group Policy, etc by default. This is too much power for any one account in today's modern enterprise. DA usually contains Service Accounts and other groups not directly related to AD administration. Ideally the DA group should be empty to ensure that each role only has the rights required to perform tasks associated with that role. The Active Directory
administrators only require membership in the domain's "Administrators" group which provides full AD admin rights as well as Domain Controller admin rights.
Membership in Domain Admins is rarely a valid requirement.

Not tracking/monitoring/documenting delegated access to Active Directory.

The best way to administer Active Directory and associated resources is to create custom groups and delegate specific access for these groups. If this isn't planned and executed properly, this delegation can get out of control enabling far greater resource access for accounts than planned. Regular auditing of groups and their access is required to properly ensure Active Directory security. Don't use the existing default groups to delegate rights to custom groups (ex. Help Desk members in "Account Operators" group) since the default groups provide more rights than are typically required. Delegation can be properly leveraged to ensure appropriate rights for each admin group. This requires gathering true requirements in plain English and translating them to system access rights.

Over-permissioned Service Accounts.

Vendors have historically required Domain Admin rights for Service Accounts even when the full suite of rights DA provides is not actually required, though It makes the product easier to test and deploy. The additional privileges provided to the Service Account can be used maliciously to escalate rights on a network. It is critical to ensure that every Service Account is delegated only the rights required and nothing more. Keep in mind that a service running under the context of a service account has that credential in LSASS (protected memory) which can be extracted by an attacker. If the stolen credential has admin rights, the domain may be quickly compromised due to a single Service Account.

Service Accounts with passwords less than 20 characters.

The problem is that it is trivial to request data (TGS service tickets) encrypted with a Service Account's password if it supports Kerberos authentication and this data can be brute forced offline to determine the password used to encrypt it. This means that Service Account passwords can be guessed offline and used to elevate credentials.

The simplest mitigation to this issue is to ensure all Service Accounts are required to have passwords containing 20 characters (or more). Configuring Fine-Grained Password Policies is the most effective method to enforce Service Account password length. Fine-Grained Password Policies are only available after configuring the Domain Functional Level to Windows Server 2008 or higher.

Don't use Group Policy Preferences to manage credentials.

Group Policy Preferences provides capability to change local administrator account passwords on computers, create local accounts, deploy scheduled tasks with credentials, create services with credentials, among others. The issue is that while the credentials are stored in XML files located in the SYSVOL share on every Domain Controller in the domain, the credential password data can be easily reversed.
If you are using Group Policy Preferences and entered a password into the policy, remove the policy and delete the file(s). Microsoft includes a PowerShell script to scan SYSVOL for password data in Group Policy Preference XML files.
Group Policy Preferences can be a useful tool for managing Active Directory resources; however, don't store credentials in Group Policy. This is one of the most common credential theft scenarios and I frequently discover passwords in Group Policy Preference files (as well as VBS scripts) in SYSVOL.

Running non-essential roles and services on Domain Controllers.

Domain Controllers should have limited software and agents installed including roles and services. Non-essential code running on Domain Controllers is a risk to the enterprise Active Directory environment. A Domain Controller should only run required software, services and roles critical to essential operation, like DNS.

Domain Controllers not patched promptly.

If critical patches are not applied promptly to all Domain Controllers, the entire domain and forest are at risk. MS14-068 is a great example of how improper patching can risk the AD Forest.

Unpatched systems (servers & workstations).

According to the Verizon Data Breach Investigation Report for 2014 released in early 2015, 99% of the vulnerabilities exploited in breaches had a patch available for over a year. Realistically, every organization needs to identify how to quickly, within days, deploy critical/high patches to all systems and deploy lower severity patches to all systems soon thereafter. Local escalation vulnerabilities that are unpatched provide attackers the ability to quickly gain admin rights on the computer which usually leads to credential theft. Having systems that aren't patched for months after the patch is released by the vendor is not realistic in the current threat environment, especially for high/critical rated patches.

Domain Controllers not running a "recent" OS version.

With each successive version of Windows Server, Microsoft has baked in additional security enhancements which greatly improve the security posture of Active Directory. Some of these security features are available once the OS is installed and others are available when the domain/forest functional level is set to a higher one.

Some of the Active Directory Domain Functional Level security features are listed here by Windows version:

Windows Server 2008 R2 Domain Functional Level:

- Kerberos AES encryption support
  - Remove RC4 HMAC Kerberos encryption.
- Managed Service Accounts
  - AD controls the service account password.
- Authentication Mechanism Assurance.
  - Users receive additional group membership when authentication with smartcard

Windows Server 2012 Domain Functional Level:

- Group Managed Service Accounts
  - AD controls the service account password.
- Compound Authentication & Kerberos FAST (Kerberos Armoring)
  - Combines user and device authentication
  - Protects Kerberos AS & TGT requests.

Windows Server 2012 R2 Domain Functional Level:

- Authentication Policies & Silos
  - Protect privileged accounts limiting where they can logon to.
- Protected Users Security Group
  - PDC set to Windows 2012 R2 to create the group
  - Protected Users Host Protection (Win 8.1/2012R2) Prevents:
    - Authentication by using NTLM, Digest Authentication, or CredSSP.
    - Cached credentials
    - DES or RC4 encryption types in Kerberos pre-authentication.
    - Account delegation.
  - Protected Users Domain Enforcement Prevents:
    - NTLM authentication.
    - DES or RC4 encryption types in Kerberos pre-authentication.
    - Be delegated with unconstrained or constrained delegation.
    - Renew the Kerberos TGTs beyond the initial four-hour lifetime.

The same local Administrator account passwords on multiple computers.

Local accounts on a computer are able to log on to that local computer whether it is joined to Active Directory or not. Interestingly enough, if the same administrator account name and password exists on multiple computers, that administrator logged onto one can connect to another with the same credentials. This means that if an attacker gains local administrator access on one computer and can connect to another with that credential, additional computers can be compromised. When the attacker has admin rights on multiple computers, it's only a matter of time before they find domain credentials with elevated rights.

It is now critical to ensure that the local Administrator password is unique on every computer on the network. Microsoft LAPS is a no-cost option leveraging existing Active Directory features.

Active Directory Admins logging on to untrusted systems (non-DCs, regular workstations, servers, etc).

Malicious code will get onto computers inside the network. The attacker leveraging this malware will search for credentials to steal and re-use. If elevated accounts logon to a variety of computers on the network, the credentials on the system that could be stolen. Limiting the systems elevated credentials explicitly logon to (interactive logon) ensures that these credentials are not available to the systems the attacker might have access to. In other words, isolating administrators to only systems they administer as well as special admin workstations protects admin credentials from credential theft.

Not monitoring admin group membership

Most organizations realize that the number of accounts with admin rights increase on a yearly, if not monthly basis, without ever going down.
The admin groups in Active Directory need to be scrutinized, especially when new accounts are added. Even better, use a system that requires approval before a new account is added to the group. This system can also remove users from the group when their approved access expires. An automated system that keeps admin groups empty until associated access is required is the best method to limit and protect admin group membership.

Not cleaning up admin group membership – ensure that accounts that no longer require admin rights are removed. This item is closely associated to the previous one.
Admin group membership needs to be reviewed on a regular basis and accounts/groups that no longer require the associated rights can be removed. The system mentioned in the previous item can be leveraged to ensure only the currently approved members are in the groups.

Not leveraging the latest security features in the platform (newer OS / enhancements via patches).

As Microsoft releases new Operating Systems, there are newer security features baked in that enhance the security of the environment. Additionally, there are patches released that include improve security settings that are not set by default which should be researched to determine what the new settings should be. A great example of this is the "back-port" patch KB2871997. Sticking with older versions of Windows limits the overall security posture of the environment.

Not automatically removing inactive (stale) user and computer accounts.

Inactive user accounts enabled in Active Directory is an attractive target for an attacker. This is because an inactive user account can be leveraged to get access to resources without being noticed since it's a valid account. Granted, control of the account is required before this can occur, but there are a few of ways to take over this account. Since it's an inactive account, there is no longer an owner of the account, so usage isn't noted.

Keeping legacy authentication active on the network (LM/NTLMv1).

Windows Server 2008 R2 included features to help identify NTLM authentication use on the network. It is important to completely remove these legacy authentication protocols since they are insecure. Removal and prevention of LM and NTLMv1 use can be activated through the use of Group Policy security settings.
Plan to move to NTLMv2 and Kerberos at the least, with the long-term goal being Kerberos only (though this won't solve all security woes).

Being too trusting – Too many Trusts or Trusts without proper security controls.

Forest and domain trusts need to be re-evaluated on a regular basis (perhaps on an annual basis) in order to ensure that they are still required, they are the correct type (is a two-way trust really required), and that the security controls are sufficient. If the trust is not configured to filter SID Filtering (Quarantine), there are security risks that may not be well understood. Additionally, it is important to evaluate if "Selective Authentication" is appropriate to enable on any or all existing trusts. The first question to ask is, "if I delete this trust, what is affected?" If the answer is "I don't know" or "I'm not sure," it may be best to disable or delete the trust.

Not isolating network resources such as critical servers.

Most internal networks are flat: any computer can typically connect to any other, meaning that any workstation can often connect to any server, including critical assets such as financial or HR databases. Given the current threat profile, this paradigm needs to change. It is not appropriate for critical servers to be directly accessible from any random computer on

the network.
Consider that the “mission critical” servers are usually running Windows 2003 or older and allow connections from anywhere on the internal network. This is a “very bad thing.” The least that can be done is to limit connectivity to these systems to only those that require access (this includes proxying access as needed).

Tags: ActiveDirectoryAttack, ActiveDirectoryDefense, ActiveDirectorySecurity, CommonSecurityIssues, DomainControllerSecurity, EnterpriseSecurity

Leave a Reply

Your email address will not be published. Required fields are marked *

Name: *

Email: *

Website:

Message: *

You may use these HTML tags and attributes: `<a href="" title=""> <abbr title=""> <acronym title=""> <b> <blockquote cite=""> <cite> <code> <del datetime=""> <em> <i> <q cite=""> <s> <strike> <strong>`

Submit Comment

Recent Posts

- The Most Common Active Directory Security Issues and What You Can Do to Fix Them
- Sneaky Active Directory Persistence Tricks
- Sneaky Active Directory Persistence #15: Leverage AdminSDHolder & SDProp to (Re)Gain Domain Admin Rights
- Sneaky Active Directory Persistence #13: DSRM Persistence v2
- Mimikatz DCSync Usage, Exploitation, and Detection

Categories

- ActiveDirectorySecurity
- Apple Security
- Cloud Security
- Continuing Education
- Entertainment
- Exploit
- Hardware Security
- Hypervisor Security
- Linux/Unix Security
- Malware
- Microsoft Security
- Network/System Security
- PowerShell
- Security
- Security Conference Presentation/Video
- Technical Article
- Technical Reading
- Technical Reference
- TheCloud
- Uncategorized

Tags

ActiveDirectory Active Directory ActiveDirectoryAttack ActiveDirectoryDefense ActiveDirectorySecurity AD2012 ADReading ADReplication ADSecurity Azure BlackHat Configuration CVE-2014-6324POC DEFCON DEFCON23 DomainController EMET5 GoldenTicket HyperV KB3011780 Kerberos KerberosHacking KerberosTicket KRBTGT LSASS MCM MicrosoftEMET MicrosoftWindows mimikatz MS14068 PassTheHash PowerShell PowerShellCode PowerShellHacking RedVsBlue Security SIDHistory SilverTicket SneakyADPersistence Windows7 Windows10 WindowsServer WindowsServer2008R2 WindowsServer2012 WindowsServer2012R2

PowerShell Security Links

- DAn Solutions Corporate Website
- GitHub Active Directory PowerShell Recon Scripts
- PowerSploit
- POSHSec PowerShell Security
- Veil PowerView

Archives

- October 2015
- September 2015
- August 2015
- July 2015
- June 2015
- May 2015
- April 2015
- March 2015
- February 2015
- January 2015
- December 2014
- November 2014
- October 2014
- September 2014
- August 2014
- July 2014
- June 2014
- May 2014
- April 2014
- March 2014
- February 2014
- July 2013
- November 2012

Copyright

Content Disclaimer: This blog and its contents are provided "AS IS" with no warranties, and they confer no rights. Script samples are provided for informational purposes only and no guarantee is provided as to functionality or suitability. The views shared on this blog reflect those of the authors and do not represent the views of any companies mentioned. Content Ownership: All content posted here is intellectual work and under the current law, the poster owns the copyright of the article. Terms of Use Copyright © 2011 - 2014

Copyright

- Return to top

Powered by WordPress and the Graphene Theme.