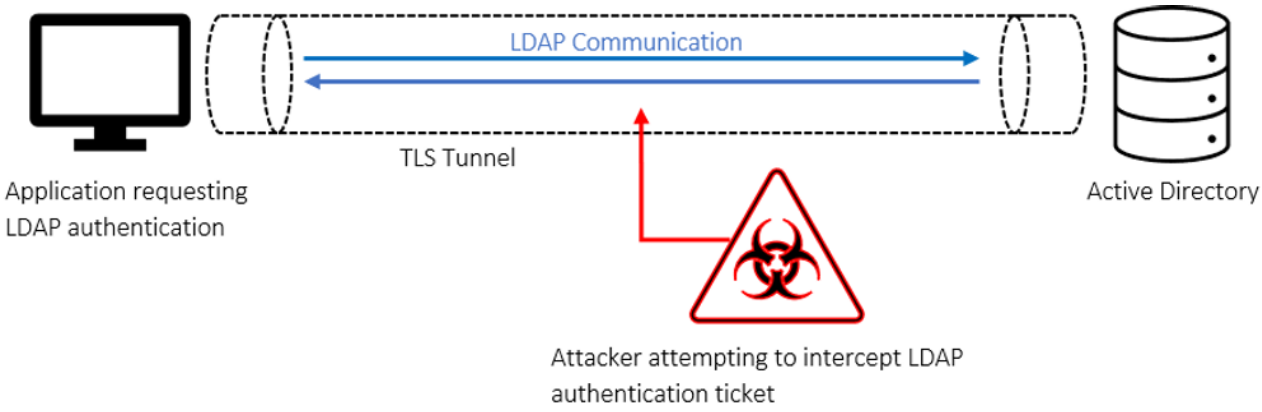
LDAP Channel Binding and Signing

hub.trimarcsecurity.com/post/ldap-channel-binding-and-signing

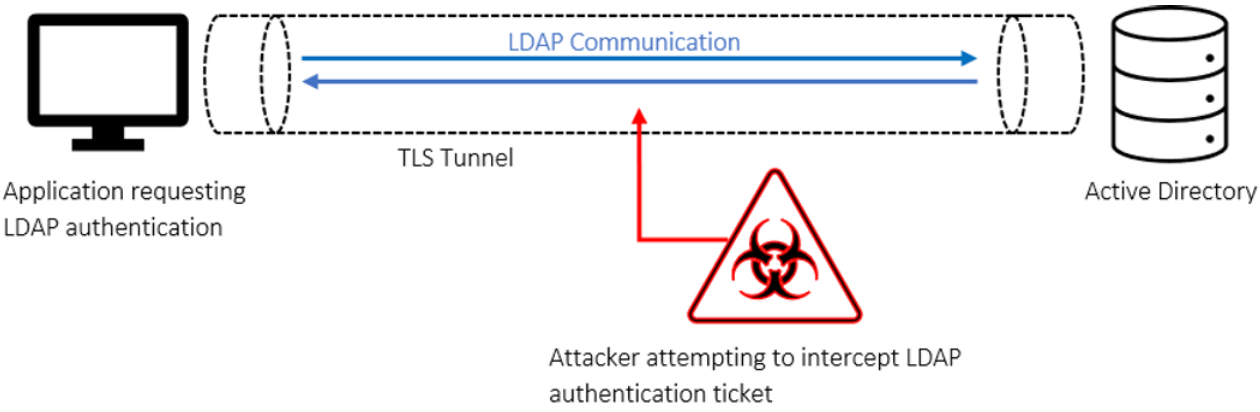
Scott Blake January 22, 2021



The initial fuss around Microsoft “forcing” customers into LDAP channel binding and LDAP signing (January 2020, March 2020, second half of 2020, TBD) appears to have overshadowed the crucial questions organizations should be addressing: The What, How, Where, & Why associated with secure LDAP communication. After speaking with security professionals attempting to implement these very settings and based on the results of Trimarc’s Active Directory Security Assessments it has become clear that the right answers are proving difficult to come by (we have seen next to zero adoption of these highly recommended configurations).

What Is LDAP Channel Binding and LDAP Signing?

It is important to recognize that while these two settings are often mentioned in the same breath, they are two distinct configurations and should be treated as such. But before breaking these down further, it would be beneficial to give a brief overview of LDAP (Lightweight Directory Access Protocol). LDAP at its core is a communication protocol commonly used by directory services like Active Directory to talk back and forth with applications regarding identity and access. The easiest way to think about this is with authentication. Active Directory stores user credentials needed by programs when validating identity. In order for a program to retrieve said credentials it can reach out to Active Directory over LDAP and Active Directory will respond in kind over the same protocol.



Oversimplification of the LDAP authentication process to illustrate the communication flow as well as an attacker's desire to intercept by positioning themselves between the application and Active Directory.

Knowing that these types of communication occur over LDAP, its fairly easy to see why it needs to be as secure as possible (what attacker wouldn't want a valid authentication ticket).

LDAP Channel Binding

LDAP channel binding was brought to our attention by Microsoft with the tagline "To make LDAP authentication over SSL/TLS more secure". Basically, LDAP channel binding is the act of tying the TLS tunnel and the application layer (leveraged by LDAP) together to create a unique identifier (channel binding token) for that specific LDAP session. This channel binding token (CBT) can only be used within that TLS tunnel and therefore prevents a "stolen" LDAP ticket from being leveraged elsewhere.

Channel binding requires a two-part implementation:

Part 1: Installing the necessary update on your Windows system (CVE-2017-8563), which should already be in place as this was released in 2017.

Part 2: Configuring the appropriate GPO setting on Domain Controllers (this no longer needs to be deployed via registry with the March 2020 Microsoft updates).

1. Domain controller: LDAP server channel binding token requirements = "Never" *(default)*
2. Domain controller: LDAP server channel binding token requirements = "When Supported"

3. Domain controller: LDAP server channel binding token requirements = “Always”

The end goal for LDAP channel binding should be c. “Always” but for auditing/testing purposes option b. “When Supported” needs to be selected to start generating failure logs with the Event ID 3039. Setting the GPO to “Always” will also permit Event ID 3039 generation but will not allow the authentication validation to occur i.e., if there are systems that do not comply, they will no longer successfully communicate with AD. See the ‘Where to Get Started’ section for more information.

LDAP Signing

LDAP signing has been around since Server 2003 with additional auditing capabilities added in Server 2008. LDAP signing, when enforced, requires clients to sign LDAP requests with its (the client’s) digital signature. By default, Active Directory does not require LDAP communication to be signed which makes its vulnerable to hacking. When LDAP signing is enforced Domain Controllers will not allow any authentication requests without a valid signature. LDAP signing ensures that the request received by the server (Domain Controller) was sent by the client the LDAP message is purported to be from. Additionally, signing certifies that the LDAP messages are not modified or tampered with.

LDAP signing is implemented in two-parts:

Part 1: Configuring Windows clients through GPO.

1. Network security: LDAP client signing requirements = “None”
2. Network security: LDAP client signing requirements = “Negotiate signing” *(Windows 10 default)*
3. Network security: LDAP client signing requirements = “Require signing”

Part 2: Configuring Domain Controllers through GPO.

1. Domain controller: LDAP server signing requirements = “None” *(default)*
2. Domain controller: LDAP server signing requirements = “Require signing”

The end goal with LDAP signing is to “Require signing” on both Windows clients and Domain controllers. As with LDAP channel binding, a robust audit plan needs to be implemented first to ensure all systems will comply (see next section). Note that signing is only supported on Windows 7 and higher, meaning if you are running XP, do not set your Domain controllers to require signing (better yet, move off XP already).

How Are These Vulnerabilities Being Exploited?

The major attack vectors these configurations seek to mitigate are person-in-the-middle and replay style attacks. Look no further than Microsoft's own executive summary of CVE-2017-8563 which describes the exploit resolved with LDAP channel binding:

"An elevation of privilege vulnerability exists in Microsoft Windows when a man-in-the-middle attacker is able to successfully forward an authentication request to a Windows LDAP server, such as a system running Active Directory Domain Services (AD DS) or Active Directory Lightweight Directory Services (AD LDS), which has been configured to require signing or sealing on incoming connections.

The update addresses this vulnerability by incorporating support for Extended Protection for Authentication security feature, which allows the LDAP server to detect and block such forwarded authentication requests once enabled."

The big takeaway here is that Active Directory is vulnerable to targeted attacks against LDAP communication. As illustrated earlier an attacker would place themselves between the application performing an authentication request and Active Directory in an attempt to intercept, modify, and forward the request to obtain a valid authentication ticket. This assailant could also potentially capture a ticket that has already been validated and reuse it in the environment. Keep in mind these security exploits all occur under the guise of a normal, legitimate user which make them all the more disconcerting.

Dirk-Jan Mollema has some excellent articles that dive into these types of attacks:

- https://dirkjanm.io/worst-of-both-worlds-ntlm-relaying-and-kerberos-delegation/
- https://dirkjanm.io/exploiting-CVE-2019-1040-relay-vulnerabilities-for-rce-and-domain-admin/

Where to Get Started?

Anytime changes of this magnitude are recommended in an environment it can cause angst (rightly so) and misinformation. After all lots of different applications, storage devices, network equipment, and operating systems rely on Active Directory for authentication. The nice part about these recommendations is that they do not need to be implemented blindly; **auditing is your friend**!

With Microsoft's March 2020 updates they have provided some great capabilities for testing/auditing LDAP channel binding (as previously mentioned, LDAP signing events have been around since Server 2008). This is great news for the admins and engineers who must answer that all important question at the next change control board meeting - "what affect will these configurations have". Arriving at the correct resolution is straightforward:

1. Enable LDAP events diagnostic logging to 2 or higher on all Domain Controllers using the command - *Reg Add HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\NTDS\Diagnostics /v "16 LDAP Interface Events" /t REG_DWORD /d 2*

2. Configure the channel binding group policy setting to "When Supported" to log binding failure events on Domain Controllers. Nothing needs to be done for LDAP signing failures as these events are already captured.

3. Configure your SIEM to alert on Directory Service Event IDs 3039 for channel binding and 2889 for signing requests that will not conform to the new requirements. These events include a Source IP address that can be used to track down the "offending" operating system/application/equipment.

4. With the "offenders" identified, research will need to be performed to see if they can be configured to comply.

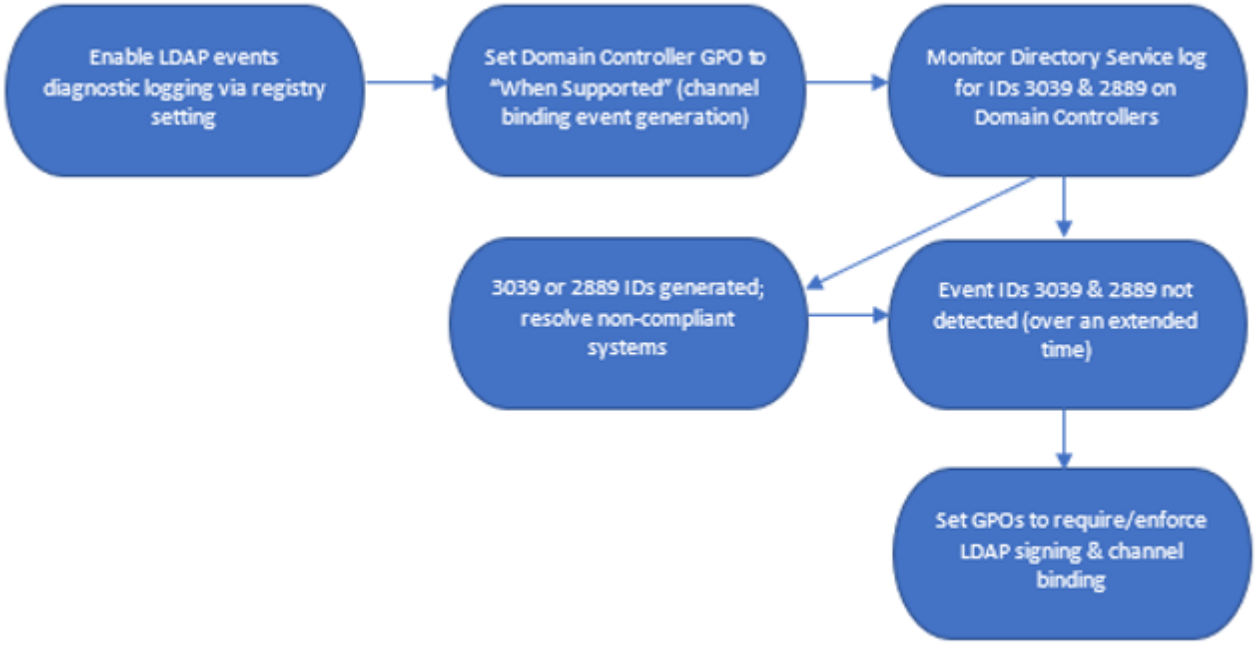
5. Once all systems correctly leverage channel binding and signing (no more 3039 & 2889 events), enforcement should be set on the GPOs.

    1. *Domain controller: LDAP server channel binding token requirements = "Always"*

    2. *Domain controller: LDAP server signing requirements = "Require signing"*

    3. *Network security: LDAP client signing requirements = "Require signing" (Windows clients)*

6. Show up at the change control board meeting with all the rights answers without having to break anything to get them. Stand for applause from your colleagues.



Simplified Flow Chart illustrating the steps needed to implement LDAP channel binding & signing.

It is important to note that Windows systems and Microsoft applications will not have any issues with these settings as long as the necessary updates have been applied and the Group Policy settings are configured correctly. LDAP channel binding is rare outside of the Windows family but audit to be sure. LDAP signing, on the other hand, is common and therefore should be audited. Simply put, audit your environment. This will also allow you to see if changes made to the "offending" systems are in fact working before implementing the Group Policy settings that will require these configurations.

Why Should We Deploy These Settings?

Hopefully at this point in the article the Why Should We Deploy These Settings question has been thoroughly answered. LDAP is inherently susceptible to attacks and having these settings configured to Microsoft's recommendation is one (or two in this case) ways to better harden your Active Directory environment. Just because Microsoft has decided not to force these changes on customers (for now) does not mean LDAP channel binding and LDAP signing should be ignored. If done correctly, and with proper forethought, these settings are an easy win for the security of any AD forest.

References & Additional Information

- https://portal.msrc.microsoft.com/en-us/security-guidance/advisory/CVE-2017-8563
- https://techcommunity.microsoft.com/t5/core-infrastructure-and-security/ldap-channel-binding-and-ldap-signing-requirements-march-2020/ba-p/921536

- https://support.microsoft.com/en-us/help/4520412/2020-ldap-channel-binding-and-ldap-signing-requirements-for-windows
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-8563
- https://support.microsoft.com/en-us/help/4034879/how-to-add-the-ldapenforcechannelbinding-registry-entry
- https://oxfordcomputergroup.com/resources/ldap-channel-binding-signing-requirements/
- https://access.redhat.com/articles/4661861

By: Scott W Blake

Trimarc Security Consultant with 10+ years building, configuring, and securing enterprise environments.

Trimarc provides leading expertise in security solutions including security reviews, strategy, architecture, and implementation. Our methodology leverages our internal research and custom tooling which better discovers multiple security issues attackers could exploit to compromise the environment. Trimarc security services fit between traditional compliance/audit reviews and standard penetration testing/red teaming engagements, providing deep understanding of Microsoft and Virtualization technologies, typical security issues and misconfigurations, and provide recommendations based on our own best practices custom-tailored to balance operational and security challenges.