InCommon Silver with Active Directory Domain Services Cookbook - RC20140115

1. Document Status

Release Candidate 20140115 for community review

2. Introduction

2.1. Purpose

The goal of this document is to offer practical compliance recommendations for InCommon Silver when a Microsoft Active Directory Domain Services (AD DS, commonly referred to as "Active Directory") forest, using user-selected passwords as the authentication credential, has been integrated into or with a campus Identity Management System.

This document is intended to aid in configuring AD DS to meet the requirements of the *InCommon Identity Assurance Profile* (IAP) version 1.2 for Silver assurance profile that specifically affect AD DS. Only sections of the IAP where there is a challenge unique to AD DS are specifically addressed. For example, issues of brute-force guessing and password entropy pose no unique challenge to AD DS; like most authentication services AD DS has controls to enable password rotation, and mitigating features like account lockout, and configuring these controls to meet those IAP sections is an exercise that requires no knowledge unique to AD DS.

For more information about the InCommon Assurance program, terms and definitions, and links to the IAP 1.2 and IAAF documents and the FAQ, see the InCommon Assurance Program.

2.2. Structure

This document is structured to provide three main types of information:

- **Analysis of AD DS functions against the IAP 1.2 requirements for Silver;** this includes interpretations of the IAP requirements where necessary for the analysis
- **Configuration and Operations Guidance**; providing specific configuration options and operations practices recommendations to meet the IAP requirements
- **Management Assertions and Alternative Means Statements**; these are intended to be (nearly) directly usable in an IAP compliance statement

2.3. Scope

This document focuses specifically on those aspects of Silver IAP compliance specific to an AD Domain Services (AD DS) environment when using user-selected passwords as the authentication credential. Specifically:

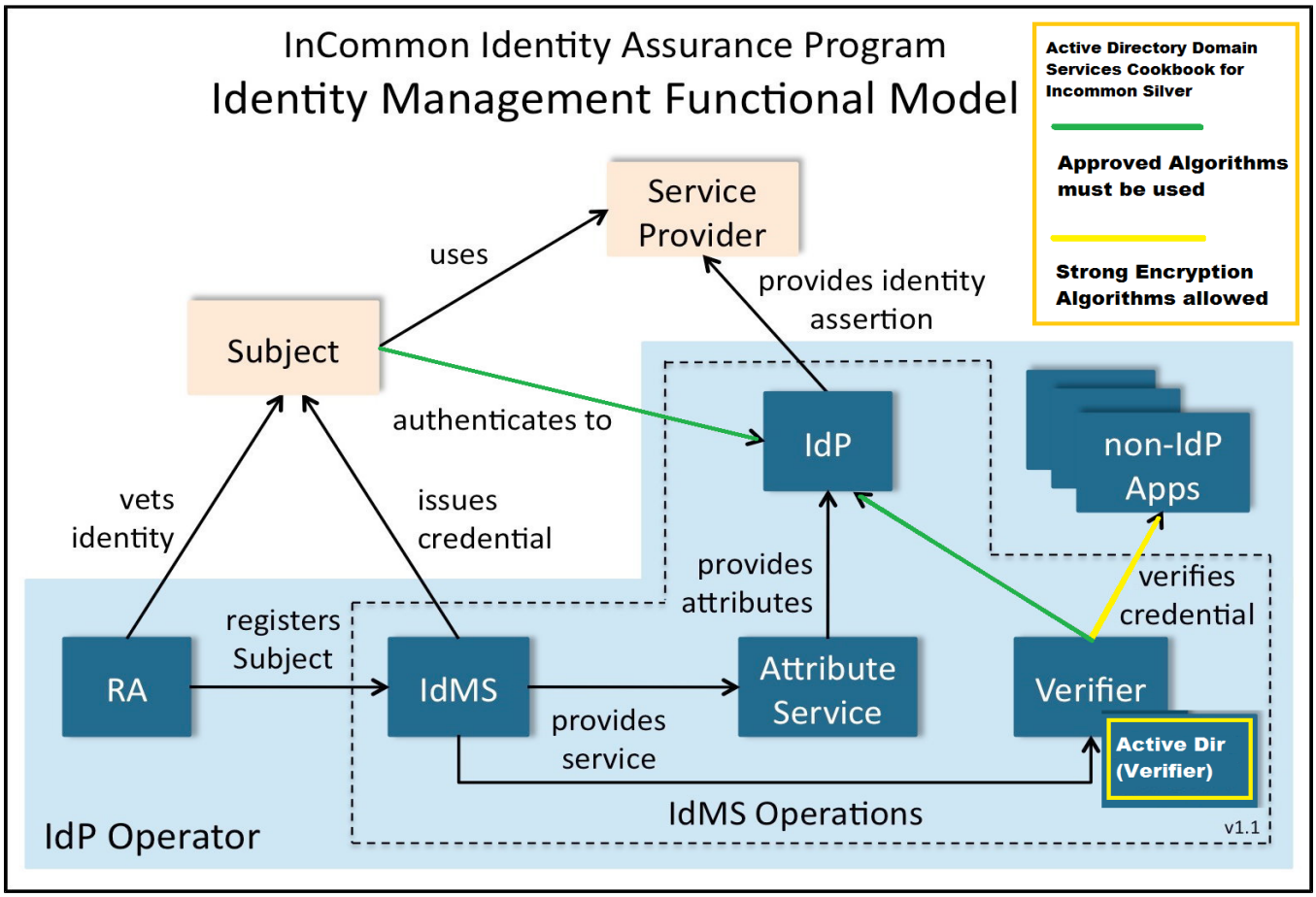
- Windows 2008R2 AD-DS used as the verifier for the IdP
- Windows 2008R2 AD-DS storing a provisioned copy of the credentials but not acting as the IdP's verifier

Other authentication components in your environments must also be assessed for compliance with InCommon Silver, but are outside of the scope of this document.

Any institution undertaking a Silver implementation project should carefully read the InCommon Identity Assurance Assessment Framework (IAAF) and Identity Assurance Profiles (IAP), both available from the Assurance section of the InCommon web site. You should thoroughly understand these documents, and determine the remediation needed in your specific environment. Specifically, you will need to understand the role that AD does and does not play within the context of the defined terms of the IAAF. For example, while AD may provide identities it is likely not the IdP (Identity Provider) for the purposes InCommon Silver or Bronze identity assertions, since AD is unlikely to be issuing the assertions. Similarly, AD may be a Verifier but it may or may not be the Verifier used by the IdP in your environment. Since many of the IAP's requirements are scoped to some component or process within the identity management infrastructure, a careful understanding of where AD fits within that infrastructure is necessary to understanding which IAP requirements apply to AD in your environment and how they may apply.

This document does not specifically address Bronze IAP compliance.

We believe that many of the approaches documented in this cookbook are applicable to all versions of AD DS from Windows Server 2003 forward (with possible exception of the IPSec approach), although the exact steps to implement them may vary. The documentation below references Windows Server 2008 R2 settings.The recommendations of this document are challenging to implement in a production environment, but the authors believe it is possible to implement them in a reasonable amount of time given some dedicated project resources and a good plan. The size of your AD DS deployment, the types of clients connected to it, the number of customers served (and in what capacity they are served) represent some of the variables you will need to consider when allocating staff and other resources to your AD DS risk mitigation project. Please refer to the diagram below for the model addressed by this document.



In addition to AD-DS, the workgroup considered evaluating several other AD IAM related products: AD Lightweight Directory Services (AD-LDS), AD Federation Services (AD-FS), AD Rights Management Services (AD-RMS), Azure AD (AAD), but the workgroup members thought that the AD-DS issues were by far the most common and most pressing issues facing the typical AD institution looking to assert InCommon Silver compliance.

We also note that using Multi-Factor Authentication (MFA) to authenticate Silver IAQ holding users may be a more effective strategy to achieve certification than the recommendations provided in this document, with its focus on securing of password storage and communication of replayable passwords. An MFA-based InCommon Silver compliance review would focus much more on the technology and function of the MFA solution implemented.

A note on SSL/TLS and SHA1: At the time of this writing, there is ongoing discussion of whether or not SSL/TLS based on SHA1 encryption will be considered a FIPS-approved Protected Channel after Jan 1, 2014. All recommendations in this document that refer to SSL/TLS channels assume that such channels are based on FIPS-approved algorithms. Whether SHA1 specifically will suffice for this purpose is beyond the scope of this document. More information regarding this specific issue is available in Draft Special Publication (SP) 800-52 (Revision 1).

2.4. Review

This document has been developed in consultation with the InCommon Assurance Advisory Committee (AAC) and the AD Assurance list membership, both groups providing review of the content as it was developed. The AAC was particularly useful in providing validation of our specific interpretations of the InCommon Silver IAP 1.2 requirements, which we found to be key components impacting compliance for AD.

At the time of this writing, no campus has achieved Silver certification using the approach outlined here. It serves as a best effort by the authors to determine what it would take to configure an AD DS environment to pass an InCommon Silver audit while still supporting real-life operations in a University setting. The intention is to update this cookbook with real-world experience as it becomes available. If you have experience implementing the recommendations of the cookbook, please consider contributing them by sending a note to: **assurance-adsilver at incommon dot org**.

3. Approach and Overview of Findings

We found that the following sections in the IAP 1.2 sections involve impacts that are technology-specific to the use of Active Directory Domain Services:

- 4.2.3.4 Stored Authentication Secrets
- 4.2.3.6 Strong Protection of Authentication Secrets
- 4.2.5.1 Resist Replay Attack
- 4.2.5.2 Resist Eavesdropper Attack
- 4.2.8.2.1 Network Security

The most common issue we identified in achieving compliance with these sections when using AD-DS is the requirement that systems use "Approved Algorithms" and "Protected Channels" for all authentication interactions. This requirement limit the allowable encryption mechanisms to those that conform to a specified published list of algorithms (see FIPS 140-2, Security Requirements for Cryptographic Modules). AD-DS is capable of supporting several protocols that are not on the Approved Algorithms list. At a very high level the recommendations in this document aim to achieve a compliant configuration through the following methods:

- Eliminating, restricting or monitoring the use of Windows-supported non-Approved Algorithms by methods such as:
    - Disabling support of certain protocols domain-wide
    - Limiting support of certain protocols to accounts that are not authenticated to the IdP
    - Monitoring for use of non-Approved Algorithms by specific account holders and responding to such use by removing Silver certification for that account. (Credit to David Langenberg, University of Chicago, for proposing this method)
- Strictly limit the methods used to interactively authenticate with the IdP, to reduce the number of protocols that could be leveraged by an attacker monitoring or manipulating network traffic
    - E.g., not relying on NTLMv2 or Kerberos tickets for authentication to the IdP, while allowing use of those protocols for non-IdP applications.
- Identify mechanisms that layer "Approved Algorithms" for encryption "on top of" the non-Approved Algorithms supported by AD
    - E.g., encrypting the volume on which the AD-DS stores its passwords, given that AD-DS' default encryption method is a non-Approved Algorithm

Much more detail is provided in the sections below, but this outlines the general approach taken in seeking to define and describe an InCommon Silver-compliant AD-DS environment.

4. Discussion of AD Issues for Meeting InCommon Silver IAP

4.1. Encrypting Passwords

The InCommon Silver IAP mandates the protection of passwords at rest. The requirements are outlined in the following IAP 1.2 sections:

- **4.2.3.4: Stored Authentication Secrets**
- **4.2.3.6.1 Strong Protection of Authentication Secrets**

4.1.1. Problem Statement

According to the both IAP sections listed above, one of the following methods must be used to protect passwords at rest:

1. hashed using an Approved Algorithm using a variable salt
2. encrypted using an Approved Algorithm and only decrypted when immediately required for authentication

In a default configuration, the AD DS password store does not meet the requirements. AD DS hashes passwords without use of a salt and does not use an Approved Algorithm for creating this hash. AD DS partially mitigates this by encrypting the entire password data store using a Domain Controller (DC)-specific Password Encryption Key (PEK), which would meet the second requirement except that the built in encryption of the password store does not use an Approved Algorithm.

This means that to meet the IAP requirements, additional mitigation -- such as the use of a disk encryption tool that uses an Approved Algorithm for encryption is needed.

4.1.2. Interpretation of IAP requirement, Section 4.2.3.4 - Stored Authentication Secrets

These requirements apply when AD DS is used as the IdP's Verifier.

We interpret this requirement to mean that encryption software that decrypts disk sectors (and not just individual Authentication Secrets) as they are accessed would meet the requirement of "only decrypt(ing) the needed Secret when immediately required for authentication" as spelled out in this section, presuming such software uses Approved Algorithms for the encryption process.

4.1.3. Interpretation of IAP requirement, Section 4.2.3.6.1 - Strong Protection of Authentication Secrets (First Sentence)

Note, this IAP requirement addresses both storage and transmission of passwords, so is addressed under both the "Encrypting Passwords" topic and the "Securing Authentication Traffic" sections.

This requirement applies to IdP Verifier passwords stored in an AD DS password store, whether or not the AD DS store is the actual IdP Verifier. Note that this requirement only applies to passwords for accounts that are actually authenticated by the IdP (non-IdP accounts that are "co located" in the AD DS have no such requirements).

For example, if a non-Windows system is used as the IdP Verifier, but the same passwords used by that Verifier are also stored in an AD DS, the AD DS password store is in scope and must be protected per sections 4.2.3.4 and 4.2.8, even though it is not being used as the IdP's Verifier. (Again, this only applies to passwords for accounts that are actually authenticated by the IdP)

4.2. Securing Authentication Traffic

The IAP identifies many requirements around securing authentication secrets and authentication traffic. These all have to do with the protection of traffic that communicates passwords or other authentication secrets during different kinds of communication.

- **4.2.3.6.1, 4.2.3.6.2, 4.2.3.6.3: Strong Protection of Authentication Secrets**
- **4.2.5.1 Resist Replay Attack**
- **4.2.5.2 Resist Eavesdropper Attack**

- **4.2.8.2.1 - Network Security**

4.2.1. Problem Statement

There are two pervasive issues here:

1) The IAP 1.2 specifically requires that communication in most of these areas be secured by using "Protected Channels", which is defined to mean channels that encrypt traffic using specific, "Approved Algorithms" under FIPS 140-2, which Windows does not always use.

2) The IAP 1.2 further specifies that communications should provide sufficient security that someone obtaining a copy of the traffic types listed above would find it "impractical" to use that traffic to impersonate a valid user, which is not the case for all Windows authentication protocols.

A synopsis of the issues follows:

The **LM** and **NTLMv1** protocols do not make it "impractical" to obtain the actual user password by inspecting traffic between clients and the AD DS, nor do they prevent replay attacks.

Use of **IPSec** for all authentication traffic ensures Protected Channels are in use, but it is very uncommon to find IPSec in ubiquitous use such that you can restrict **all** incoming authentication traffic to use it.

LDAP data signing encrypts the data portion of each LDAP packet, including authentication traffic, but if the domain is configured to limit authentication traffic to require signing, it can impact the operation of older or non-Windows clients in an AD DS environment. *(Note, there is an open question to verify that signing encryption uses an Approved Algorithm)* More about these technologies is included in the appendices.

Using **LDAPS (TLS/SSL)** will create a Protected Channel for LDAP authentication, but conversely to the issues with LDAP data signing, requiring LDAPS will impact many Windows clients, which require LDAP (without TLS/SSL) for some key functionality. See the appendices and http://support.microsoft.com/kb/832017 for details of the impacts requiring LDAPS has on Windows clients.

Both **NTLMv2 and Kerberos** are called into question because they rely on the MD5 and RC4 cipher suites respectively, neither of which is an Approved Algorithm. Note that Kerberos can be configured to use AES128 instead of RC4, but this is not commonly seen as a configuration in Windows domains.

In addition, both **NTLMv2 and Kerberos** use the user's password directly as an encryption key for authentication traffic. We note that there is language in 800-63-1 (see in particular, footnote 26) that provides a very quantitative definition of the meaning of an "impractical to break" authentication secret. This language is not currently part of the InCommon IAP 1.2 language, so may not apply to the definition of "impractical" used in sections 4.2.5; however, we note there is a risk that the language in 800-63-1 will become required under the IAP in the future, either due to a modification of the IAP language in a revision, or because of increasing efficiency of attacks may just naturally make the protocols less "impractical" to attack offline.

As a final note, even though not necessarily technically a Windows protocol, we wanted to comment on the **MSCHAPv2** protocol leveraged by the eduroam service. The **MSCHAPv2** protocol is not itself a secure protocol; however, in the context of eduroam it is implemented using PEAP-MS-CHAPv2. PEAP establishes a TLS tunnel to protect the actual MS-CHAPv2 messages communicated between the RADIUS client and server, which provides a protected channel. The use of MS-CHAPv2 alone is not acceptable as is it known to be cryptographically weak.

4.2.2. Interpretation of IAP requirement, Section 4.2.3.6.1 - Strong Protection of Authentication Secrets (Second Sentence)

Note, this IAP requirement addresses both storage and transmission of passwords, so is addressed under both the "Encrypting Passwords" topic and the "Securing Authentication Traffic" sections.

This requirement applies when IdP Verifier passwords are provisioned into an AD DS store, whether or not the AD DS store being provisioned to is the actual IdP Verifier. Note that this requirement only applies to passwords for accounts that are actually authenticated by the IdP (non-IdP accounts that are "co located" in the AD DS have no such requirements).

For example, if a non-AD DS system is used as the IdP Verifier, but the passwords used by that Verifier are also provisioned into an AD DS, that AD DS provisioning process is in scope and therefore must use Protected Channels, even though it is not being used as the IdP's Verifier. (Again, this only applies to passwords for accounts that are actually authenticated by the IdP, or which allow modification of accounts authenticated by the IdP)

4.2.3. Interpretation of IAP Requirement, Section 4.2.3.6.2 - Strong Protection of Authentication Secrets

Authentication Secrets[1] in the context of this requirement is interpreted to mean those secrets (passwords, Kerberos session keys, NTLM challenge responses, etc.) that can be used to directly authenticate ***to the IdP*** or to directly modify authentication credentials (i.e., can be submitted directly to a ***change my password*** page).

Another way of stating this is that the intent of 4.2.3.6.2 is only to protect against risks that can attack the IdP's direct authentication to its Verifier; protection of non-IdP Authentication Secrets that incidentally use the same Verifier as the IdP, such as Windows/Kerberos Tickets used to authenticate a user to a domain (a la ctrl-alt-del)[2], are not in scope of 4.2.3.6.2. These non-IdP Authentication Secrets are addressed elsewhere in the spec, such as in 4.2.3.6.3.

[1] The term "Authentication Secret" is assumed here to include the information in an authentication packet; e.g., in addition to actual authentication credentials (username/password), the information negotiating the initial exchange of a session key or the response to an authentication challenge is an "Authentication Secret".

[2] This assumes that the local Windows/Kerberos tickets cannot be leveraged to authenticate the user to the IdP, such as a standard webpage that prompts for entry of user credentials. If the IdP supports mechanisms that allow the user to generate authentication assertions based directly on the possession of local Windows/Kerberos tickets (e.g., SPNEGO, GSSAPI and certain ADFS configurations), then the Windows/Kerberos tickets used in that interaction WOULD be in scope of this requirement.

4.2.4. Interpretation of IAP Requirement, Section 4.2.3.6.3 - Strong Protection of Authentication Secrets

This section addresses all handling of IdP Verifier authentication secrets, even if those authentication secrets are not used to authenticate anyone to the IdP. Note that authentication secrets that could be replayed to authenticate directly to the IdP are also covered by section 4.2.3.6.2, which has stricter requirements around transport encryption.

That is, assuming an IdP that requires manual entry of a username/password:

- applications (including non-IdP applications) that handle the username/password would be covered by both sections 4.2.3.6.2 and 4.2.3.6.3.
- applications (including non-IdP applications) that handle non-IdP authentication secrets based on IdP passwords (e.g., LM, NTLMv1, NTLMv2 or Kerberos authentication packets for Silver IAQ users) would be covered only by section 4.2.3.6.3

This section requires that policies and procedures are in place to minimize the risk of this traffic being attacked in a way that would subvert the security of IdP authentication events. While traffic covered in this section ***may*** use Protected Channels to secure authentication communications, an institution could also use other mechanisms that are generally considered secure, even if they are not Protected Channels.

For completeness, we note that protection of the password while it is being locally processed within any application is also required, not just protection while in transit to or from said application. However, security of the information while being handled in a transient fashion within a web application is beyond the scope of AD DS configuration, so is not covered here.

4.2.5. Interpretation of IAP Requirement, Section 4.2.5.1 - Resist Replay Attack

This section refers specifically to traffic between the Subject and the IdP, the IdP's Verifier, and/or a relying party ***as part of an IdP authentication /assertion event***. All other traffic between the Subject and the AD DS is beyond the scope of 4.2.5.2. Replay of non-IdP authentication/assertion traffic is covered by sections 4.2.3.6.2 and 4.2.3.6.3.

4.2.6. Interpretation of IAP Requirement, Section 4.2.5.2 - Resist Eavesdropper Attack

This section refers specifically to traffic between the Subject and the IdP, the IdP's Verifier, and/or a relying party ***as part of an IdP authentication /assertion event***. All other traffic between the Subject and the AD DS is beyond the scope of 4.2.5.2. Eavesdropping on non-IdP authentication/assertion traffic is covered by sections 4.2.3.6.2 and 4.2.3.6.3.

4.2.7. Interpretation of IAP Requirement, Section 4.2.8.2.1 - Network Security

"Network communications supporting IdMS operations" is interpreted to mean communications between the actual software elements of the IDMS operation or administrative traffic to the IDMS.

So the only *AD DS-specific* communications that would be in scope of this requirement is intra-domain controller password replication. (Note that provisioning of passwords into AD-DS is covered by 4.2.3.6.1)

5. Specific Configuration Recommendations

5.1. Configurations to address Passwords at Rest

Relevant to IAP sections

- **4.2.3.4 Stored Authentication Secrets**

5.1.1. Encrypt the Password Store Using 3rd Party Tools

Employing a disk encryption utility to encrypt the password store using an Approved Algorithm and decrypting the password store "only when immediately required for authentication" is required to meet the IAP requirements. Syskey (Windows' built-in disk encryption tool) uses RC4 for its disk encryption, which is not an Approved Algorithm. Use of a product like BitLocker to encrypt the volumes on the Domain Controller (DC) which store secrets will provide this appropriate protection. In making this determination, we assert that an AD DS DC gaining access to the password store continuously after booting meets the definition of "only when immediately required", as a primary function of DCs is to handle interactive client authentications.

Note that the identification of BitLocker as a viable product is based on it being bundled with the Windows Domain Controller license [citation or correction needed], and not as an endorsement of the product over other similar products.

5.1.2. Remove Insecure (LMHASH) Stored Secrets

Disable storage of the LMHASH. This requires Windows 7, Server 2008, and/or setting a GPO setting "Network security: Do not store LAN Manager hash value on next password change" OR by requiring 15 character or longer passwords (since LMHASHes cannot be applied to passwords greater than 14 characters). Any accounts with LMHASH values stored prior to enabling this setting will require password changes for any subject for whom Silver is to be asserted (changes to this setting only affect generation of new password hashes, not existing ones).

To disable the storage of LM hashes of a user's passwords in the local computer's SAM database by using Local Group Policy (Windows XP or Windows Server 2003) or in a Windows Server 2003 AD DS environment by using Group Policy in AD DS, follow these steps:

1. In Group Policy, expand **Computer Configuration**, expand **Windows Settings**, expand **Security Settings**, expand **Local Policies**, and then click **Security Options**.
2. In the list of available policies, double-click **Network security: Do not store LAN Manager hash value on next password change**.
3. Click **Enabled**, and then click **OK**.

NOTE: If you encrypt the password store (see above recommendation), then the LMHASHes are further encrypted and storing them may not technically violate the InCommon IAP. However, any authentication mechanism that relies on the use of the LMHASH violates the IAP, so we still recommend removal of these hashes.

5.1.3. Other Controls

It is possible that a mitigation program that includes detailed network traffic monitoring, attack detection and alerting, system activity and physical controls could mitigate the potential access to a non-conforming password store, but these kinds of controls would go well beyond anything that is specific to a Windows installation and would likely vary widely from institution to institution, so no alternate controls or Alternative Means Statements were developed to support such mechanisms.

5.2. Configurations to Secure Authentication Traffic

5.2.1. Transmission of Authentication Secrets Between Credential Stores

Relevant Section: 4.2.3.6.1 - Strong Protection of Authentication Secrets (Transmission of secrets between data stores)

Active Directory Domain Services uses RPC and Kerberos when synchronizing between domain controllers. For Windows Server 2008 and later, AES is used for Kerberos encryption if properly configured.

- For **"Network security: Configure encryption types allowed for Kerberos"**, select one of the AES options. See http://technet.microsoft.com/en-us/library/jj852180(v=ws.10).aspx

This section also requires that processes that provision passwords also use Protected Channels. The simplest way to enforce this is to use only LDAPS (TLS/SSL) communications for provisioning operations.

5.2.1.1. Other Controls

An appropriately configured mechanism such as IKE/IPSEC using an Approved Algorithm (such as AES) to secure traffic between Domain Controllers and between provisioning applications and the Domain Controllers may be used to create the Protected Channel. Configuration of this option will vary based on network topology and environment.

5.2.2. Ensure IdP Authentication Secrets are Protected in Transit

Relevant Sections:

- **4.2.3.6.2 Strong Protection of Authentication Secrets**

Recall that in the Interpretation section we interpret that the IAP requirement only applies to secrets that can be used to authenticate directly to the IdP, or to an IAM component that can compromise the IdP's use of the Verifier.

To simplify the process of asserting compliance we recommend an IdP configuration that requires direct entry of user credentials for user authentication, rather than relying on existing authentication tokens on the user desktops through use of SPNEGO, SASL or similar mechanisms. This allows the process of IdP authentication to be completely separated from all other potential protocol issues (e.g., Kerberos ticket attacks), and that any NTLMv2 or Kerberos credentials (used for non-IdP authentication) somehow accessed off the wire or a user's disk cannot be used to impersonate a user to the IdP. Even though such intercepted credentials may be used to gain access to, e.g., file shares in the AD Domain, this does not allow the IdP authentication process to be compromised.

To meet this requirement, you must ensure that all authentication traffic containing account authentication credentials that passes between the subject, the IdP, ***non-IdP applications*** and the IdP Verifier is secure via Protected Channel methods:

- Encryption on the wire via IPSec
  - Use IPSec policies to force LDAP (port 389) to use a secured channel. Steps for creating an IPSec policy are documented on Microsoft TechNet: http://technet.microsoft.com/en-us/library/cc730656.aspx. If you choose IPSec, you must require that all authentication traffic be encrypted using IPSec. This depends on forest functional level (must be Windows Server 2008 forest functional level); if lower OS DCs exist in a domain, then the least common denominator is used.

OR

- Require LDAP data signing and/or LDAPS for all LDAP traffic
  - Require signed LDAP traffic by setting the following GPO setting: **Domain Controller: LDAP Server signing requirements=Enabled**http://technet.microsoft.com/en-us/library/jj852173(v=ws.10).aspx*Note: This may require deploying an additional GPO setting to clients, **"Network security: LDAP client signing requirements**", and require third party applications to be reconfigured to use SSL/TLS or signed SASL binds. This may have an adverse effect on the ability of Mac OS X and other clients to authenticate using the AD DS, so care must be taken in an enterprise setting when testing and communicating this change. http://support.microsoft.com/kb/823659_* discusses some possible incompatibilities with doing this and
    Appendix B discusses one solution for Mac OS X clients.
  - LDAP traffic (non SSL) can be blocked by limiting access on port 389. *At this time we do not have information on allowing access to port 389 and requiring TLS.*
    See the appendices and http://support.microsoft.com/kb/832017 *for details of the impacts requiring LDAPS has on Windows clients.*

5.2.3. Protect non-IdP related authentication traffic to AD DS

Relevant Sections:

- **4.2.3.6.3 Strong Protection of Authentication Secrets**

For these sections, the requirements are further extended to ensure that user passwords cannot be easily extracted from interception of non-IdP related authentication events and protocols.

We recommend disabling all domain support of the **LM** and **NTLMv1** protocols -- or at least disable support for those InCommon Silver accounts, as the security of these protocols has been shown to be weak enough for use to constitute being an unacceptable risk in most usage scenarios.

- Set LAN Manager authentication level to **"Network Security: Send NTLMv2 response only\refuse LM & NTLM".** See technical details in http://technet.microsoft.com/en-us/library/cc738867(v=WS.10).aspx

Non-IdP applications that do not specifically pass user authentication credentials are not specifically required to use Protected Channels for all authentication events. Therefore, use of reasonably secure protocols, specifically NTLMv2 and Kerberos, outside of the context of an IdP authentication is acceptable, even given that they are not formally "Protected Channels". Using any of these protocols should be accompanied by a formal practice of regularly applying relevant patches to mitigate new attacks and vulnerabilities as they are discovered.

5.2.3.1. Other Controls (Sections 4.2.3.6.2 and 4.2.3.6.3)

If using the recommended settings listed above would cause undue hardship in your community, risk mitigation may be achieved by monitoring network activity for authentication via non-approved means and removing the Silver IAQ from affected accounts until the user is contacted and a password change has taken place. That is, if a user authenticates using an unsigned, non SSL/TLS LDAP binds, or authenticates to a service using the LM protocol the account would be treated as "temporarily compromised" for purposes of the Silver IAQ assertion.

See the **Controls statement #1: Monitor and Mitigate** for details on the recommended alternative process.

5.2.4. Section 4.2.5.1 and 4.2.5.2 requirements

Presuming that you have followed the recommendations in section 4.2.3.6.2 and require direct entry of user authentication credentials at the IdP, the only interception and eavesdropping attacks possible are those that attack the direct communication between the subject and the IdP, and no other configurations are required here.

6. Alternate Controls and Alternative Means Statements

1. "Monitor and Mitigate" Alternate Control for Satisfying Requirement 4.2.3.6.3 in Active Directory Domain Services Environments - DRAFT 2013102

7. Sample Management Assertions

This section provides template language for asserting compliance, assuming that recommended configurations and practices above have been implemented.

7.1. Management Assertion for section 4.2.3.4 - Stored Authentication Secrets

With the use of **[full disk encryption tool],** Campus AD DS stored passwords are encrypted with an Approved Algorithm for encryption method at rest ([Encryption Algorithm]) and only decrypted when the disk sectors storing the password hashes are accessed by AD DS (the IdP Verifier).

Note: replace boldface items with your actual product and algorithm. BitLocker, which is included in Windows distributions, is capable of using either AES128 or AES256 key based encryption depending on your configuration, both of which are Approved Algorithms.

7.2. Management Assertion for section 4.2.3.6.1 - Strong Protection of Authentication Secrets

Storage of secrets (first sentence of requirement)

This sentence refers to requirements of sections 4.2.3.4 and 4.2.8; see Management Assertions for those sections.

Transmission of Authentication Secrets between Credential Stores (second sentence of requirement)

If using the **Recommended** option:

Because all domain controllers run Windows 2008 or higher, and are properly configured, synchronization of passwords between domain controllers is protected by AES encryption, which is an Approved Algorithm.

Provisioning activities to the Active Directory Domain Controller all take place over LDAPS (SSL/TLS) or LDAP Data Signing which is ***an Approved Algorithm.*** *You will need to justify this statement with information from your provisioning processes.*

Note, while not related to AD DS, your management assertion here must also cover provisioning activities to any non-AD DS data stores you communicate with, such as LDAP, Kerberos servers, etc.

If using the **Other Controls** option:

All traffic between domain controllers is secured by IKE/IPSEC using the *your algorithm here* protocol to secure traffic between Domain Controllers, and between provisioning systems and the Domain Controllers, which provides the required Protected Channel for all traffic.

7.3. Management Assertion for section 4.2.3.6.2 - Strong Protection of Authentication Secrets

If using the **recommended** configuration:

Authentication to the IdP is by direct entry of the user authentication credentials, so authentication traffic to the IdP can only be compromised by directly accessing the data packets as they are sent to the Verifier by the IdP or non-IdP applications. All authentication processes that directly manage the authentication credentials are configured to use Protected Channels, protecting against interception:

- All LDAP binds to be made over using **IPSec connections secured by *(your protocol here)***, enforcing a Protected Channel.
- **LDAP data signing**, which encrypts the password data ***need to validate algorithm to see if this is good enough***. While this is message level and not transport level encryption, the data in transit is still protected using an Approved Algorithm.
- **LDAPS (SSL/TLS**), which establishes a Protected Channel.
- RADIUS with **PEAP-MS-CHAPv2** (if providing eduroam support), which secures the communication within a Protected Channel

If using the **"Monitor and Mitigate"** control:

Any non-SSL authentication traffic (e.g., LDAP) is detected, and the accounts using these mechanisms have their InCommon Silver IAQ removed within 72 hours, as if the account had actually been compromised. Reasserting the InCommon Silver IAQ requires resetting of the credential. *Reference language from the Monitor and Mitigate control*

7.4. Management Assertion for section 4.2.3.6.3 - Strong Protection of Authentication Secrets

If using the *recommended *configuration: Authentication for InCommon Silver users is restricted to using appropriately secure protocols. Specifically: LM and NTLMv1 are not allowed. NTLMv2 and Kerberos based authentication is allowed for non-IdP application authentication, which -- while not using Protected Channels -- is *impractical* to attack in a way which retrieves the user's raw authentication credentials. We regularly apply patches that may affect the security of these supported protocols.

If using the **"Monitor and Mitigate" *control:** Any or weak (e.g., LM, NTLMv1) authentication traffic is detected, and the accounts using these mechanisms have their InCommon Silver IAQ removed within 72 hours, as if the account had actually been compromised. Reasserting the InCommon Silver IAQ requires resetting of the credential. See section **Alternate Control and Alternative Means Statements** for more details.

7.5. Management Assertion for Section 4.2.5.1 - Resist Replay Attack

If using **Recommended** settings:

All requirements for this section are handled via the same mechanisms as defined for 4.2.3.6.2: by forcing the user to enter authentication credentials separately to the IdP, and using Protected Channels for this communication, the authentication event resists replay attack.

7.6. Management Assertion for Section 4.2.5.2 - Resist Eavesdropping Attack

If using **Recommended** settings:

All requirements for this section are handled via the same mechanisms as defined for 4.2.3.6.2: by forcing the user to enter authentication credentials separately to the IdP, and using Protected Channels for this communication, the authentication event resists replay attack.

7.7. Management Assertion for Section 4.2.8.2.1 - Network Security

See Management Assertion for Section 4.2.6.3.1 for security of password synchronization between Domain Controllers.

The rest of this assertion will be unique to your environment. Communication between elements of the IDMS components is secured by ***explain configuration*** which establishes a Protected Channel. *Language here may need to rely on an alternate means statement for SHA1-based SSL.*

8. Appendices

Appendix A - Known Issues With NTLMv1 Disabled/LMHASH Storage Turned Off

Put any known issues/affected systems here, along with how you solved the problem, if possible.

See http://technet.microsoft.com/en-us/magazine/2006.08.securitywatch.aspx for an exhaustive discussion of the LanManCompatibilityLevel setting.

By default, Windows XP and earlier clients aren't compatible with DCs that have LanManCompabilityLevel=5. This means you must get all older Windows clients reconfigured. Domain joined computers can be easily addressed with group policy. Non-domain joined computers will require manual configuration or a script you provide.

Appendix B - Known Issues With Requiring Signed LDAP Binds

Put any known issues/affected systems here, along with how you solved the problem, if possible.

Cisco TMS doesn't support LDAPS. This is an application that integrates with AD DS to provide identity and access management. No resolution. Maybe newer versions will provide LDAPS support. Unclear if Cisco TMS supports LDAP signing.

While the Mac OS GUI claims it will enable LDAP signing by default, in practice, it doesn't. However, if you use Apple's dsconfigad command line tool with the switch "-packetencrypt ssl", you can tell the Mac OS to use LDAPS (i.e. employ LDAP over TLS/SSL). This protects Mac OS clients authentication traffic. This dsconfigad option can be used at the time of Mac computer domain join or it can be used after domain join to mitigate this issue.

Appendix C - Operational Considerations, Practices, Processes For Use of Disk Encryption Software

Put any known issues, operational considerations, process changes, etc., along with how you solved any problems here, if possible.

A potential issue with the use of BitLocker on Domain Controllers arises if you deploy DCs via virtual machines (VMs), as Microsoft does not support BitLocker directly within VMs. However, Microsoft notes that if you BitLocker a HyperV host (i.e. the volumes the guest disks are on), the BitLocker protections are enjoyed by all guest VMs. Microsoft may or may not support other full disk encryption solutions when virtualizing a Domain Controller, so check on support from Microsoft if you run in an alternate configuration.

Appendix D - InCommon Assurance Framework Terminology

Credential Issuance
Credential Issuance is the process that binds the credential to the subject and enables the credential to be used by the subject. This process may or may not actually add the records associated with the credential to systems. With Windows AD DS, this may be accomplished by creating and enabling the user account, and conveying the password to the subject via a registration process that complies with the subject registration and credential issuance requirements of the IAP

Credential Revocation
Credential Revocation is the process that removes the binding of the credential from the subject and renders the credential non-usable by the subject. This process may or may not actually remove the records associated with the credential from systems. With Windows AD DS, this may be accomplished multiple ways:

- disabling the user account
- giving the user account an account expiration value in the past
- changing the password on the user account to a value unknown to the user (note that this is only acceptable as a temporary method of revocation--one of the other methods must be used before your password entropy/brute force period is exceeded)
- deleting the user account
- moving the user account to a directory which is configured to prevent Silver authentication, such as a correctly configured Active Directory Lightweight Directory Services (AD-LDS).

Appendix E - Password Entropy - Calculating it, what's needed, what's "good enough," etc.

While this topic is not specifically within the scope of this cookbook, it plays a big role in brute force and dictionary attacks against your credential store. NIST SP 800-63 Appendix A contains a lengthy discussion of Claude Shannon's notion of information entropy and complexity as it applies to passwords. We'll leave that discussion to that document. Here, we'll say that there are any number of ways to reach the required 1:16384 chance of guessing a password during its active life, and the 14 bits of entropy, against a targeted guessing attack, that are required by the Silver IAP. You can have longer, complex passwords that are active for a longer time, shorter, less complex passwords that are active for a shorter time, with more or less aggressive account lockout policies, etc. SP 800-63, the IAP document and an entropy calculation spreadsheet, such as http://www.infoworld.com/sites/all/themes/ifw/downloads/passwordcalc096.zip are good reference documents.

Appendix F - FAQ

(Please add any FAQ/Q&A type things you can think of with regard to AD DS here- if you have a question, please add it, and if you have an answer, please add it. They don't necessarily have to come in pairs, but we'll try to collaboratively fill in the blanks for each other.)

Caveat: We are not auditors, and we aren't the InCommon arbitration board, the TAC, or the steering committee (and we aren't even close to being FICAM.) So, take these Q&As as they are intended, as a best effort interpretation. These statements largely remain to be vetted by auditing/achievement of Silver.

Q: What is within scope for services based on AD DS? In other words, if I have AD DS on my campus, and the passwords are the same as passwords used by my IdP (even if the IdP doesn't directly authenticate against AD DS,) do I have to be concerned with the security of authentication events for every dependent service?

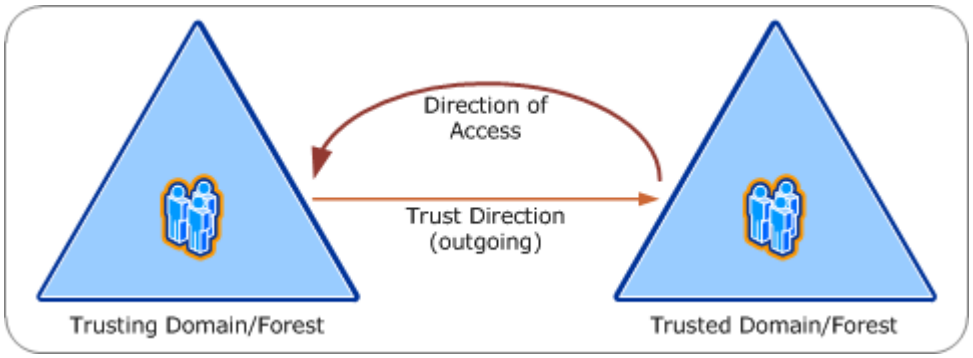
A: In general, the scope will be defined by the "Identity Management System Operations" in the Identity Management Functional Model described in section 2 of the IAAF. In general, AD DS will function in the role of both a "Verifier" and an "Attribute Service". You should use common sense and best security practices in your assessment of the situation at your institution. If there are strategies you can use to detect direct simple binds to AD DS and prevent NTLMv1 connections, those are good things to do. You should probably have a strategy for following up with people doing simple binds and asking them not to do that. You may want to consider spot auditing owners of any kind of service ID that does any kind of authentication with your AD DS implementation, to make sure they aren't doing things like exposing passwords via unprotected forms authentication on web sites, etc. You can probably use a combination of institutional authentication policy, monitoring processes (such as intrusion detection system rules that look for simple LDAP binds) and spot checks to mitigate this risk adequately.

Q: What is the security boundary for an AD DS deployment?

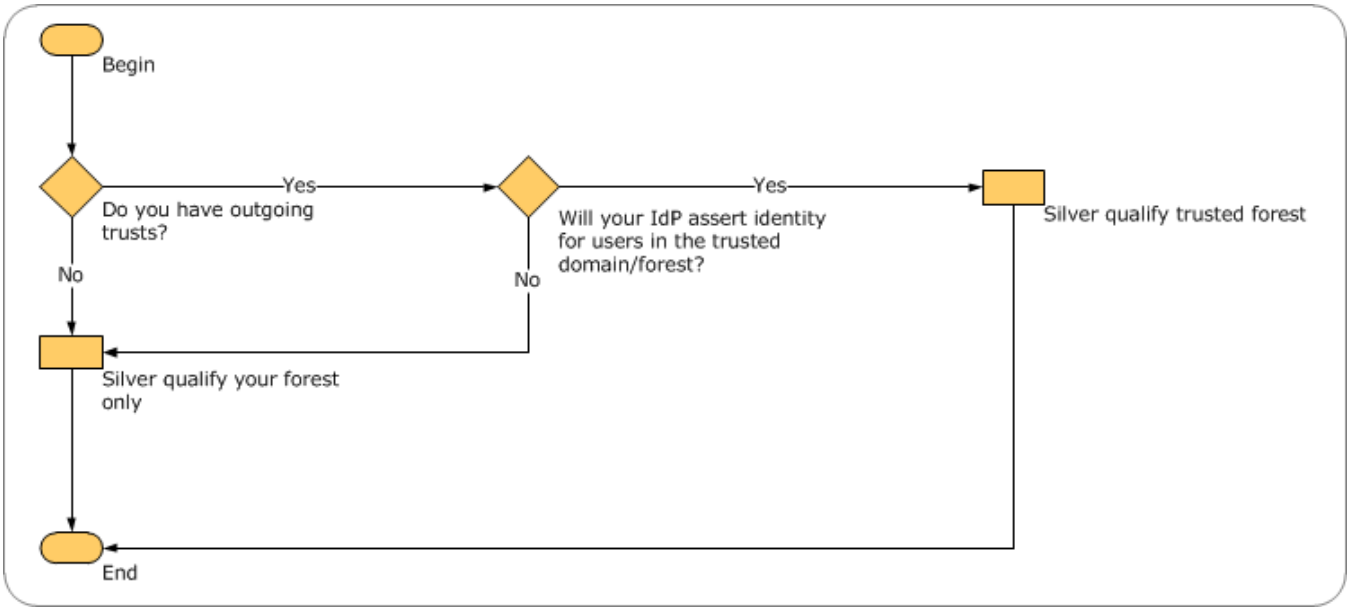
A: The security boundary is the forest, unless you have a domain or forest trust and you are the trust*ing* domain/forest. If you have a trust, and your IdP asserts identity for principals in the trusted domain or forest, then both forests are in-scope. For more information, see figures 1 and 2, below.

Figures

1 Basics of AD DS trusts (diagram by Brian Desmond)



2 Decision Flowchart for AD DS Domain/Forest Trust and Silver Compliance (diagram by Brian Desmond)



Glossary

Authentication Credential

The information entered or held by an individual used to verify their identity during authentication. Most commonly, username/password.

Authentication Secret

Information used to validate an individual in an authentication negotiation; e.g., in addition to actual authentication credentials, the information negotiating the initial exchange of a session key or authentication challenges is an “Authentication Secret”.

IPSec

Internet Protocol Security (IPSec) is a framework of open standards for ensuring private, secure communications over IP networks through the use of cryptographic security services. The Microsoft Windows implementation of IPsec is based on standards developed by the Internet Engineering Task Force (IETF) IPSec working group.

IPSec establishes trust and security from a source IP address to a destination IP address. The only computers that must know about the traffic being secured are the sending and receiving computers. Each computer handles security at its respective end with the assumption that the medium over which the communication takes place is not secure. Computers that only route data from source to destination are not required to support IPsec unless firewall-type packet filtering or network address translation (NAT) is performed between the two computers.

You can use the IP Security Policy snap-in to create, edit, and assign IPsec policies on a local computer and remote computers.

Identity Provider (IdP)

The IdMS system component that issues Assertions.

Kerberos
Kerberos is an authentication protocol which works on the basis of "tickets" to allow nodes on a non-secure network to prove their identity to one another in a secure manner. It provides mutual authentication - both the subject and the server verify each other's identity. Kerberos protocol messages are protected against eavesdropping and replay attacks. Kerberos builds on symmetric key cryptography and requires a trusted third party, and optionally may employ public key cryptography by utilizing asymmetric keys during certain phases of authentication. (from Wikipedia). When used in this document "Kerberos" generally refers to the Windows-specific Kerberos implementation.

LDAPS
LDAPS refers to encrypted LDAP communication achieved either by using an SSL tunnel (via LDAPv2) or by using the LDAPv3 Transport Layer Security (TLS) extension. Which method is used depends on what the client supports. The default port associated with LDAPS traffic is 636, and for AD DS global catalog traffic is port 3269. Strictly speaking, the term LDAPS was deprecated with LDAPv2, but in common usage it is also used to refer to TLS based encrypted LDAP traffic.

LMHASH
LM hash, **LanMan**, or **LAN Manager hash** was the primary hash that Microsoft Windows versions prior to Windows NT used to store subject passwords. Support for the legacy LAN Manager protocol continued in later versions of Windows for backwards compatibility, but was recommended by Microsoft to be turned off by administrators; as of Windows Vista, the protocol is disabled by default, but continues to be used by some non-Microsoft CIFS implementations. (from Wikipedia)

NTLM/NTLMv2
NTLM (**NT LAN Manager**) is a suite of Microsoft security protocols that provides authentication services. NTLM is the successor to the authentication protocol in Microsoft LAN Manager (LANMAN), an older Microsoft product, and attempts to provide backwards compatibility with LANMAN. NTLM version two (NTLMv2), which was introduced in Windows NT 4.0 SP4 (and natively supported in Windows 2000), enhances NTLM security by hardening the protocol against many spoofing attacks, and adding the ability for a server to authenticate to the client. (from Wikipedia)

SPNEGO

A protocol (arguably a GSSAPI mechanism) that allows use of Microsoft AD DS authentication tokens on the desktop to be extended to authenticate web-based applications.

SYSKEY
A tool used to configure the startup key, a random, 128-bit, symmetric cryptographic key created at system startup and used to encrypt all of the user`s symmetric cryptographic keys. Use SysKey with a password shared between two individuals (person A knows the first 8 characters, person B knows the second 8 characters). The steps for configuring SysKey are here:http://technet.microsoft.com/en-us/library/cc773183(WS.10).aspx

Comments

Please send any comments regarding this document to: ad-assurance@incommon.org

Contributors

- Lee Amenya, University of California, San Diego
- Brian Arkills, University of Washington
- Michael Brogan, University of Washington
- Jeff Capehart, University of Florida
- Erik Coleman, University of Illinois at Urbana-Champaign
- Warren Curry, University of Florida
- Eric Goodman, University of California Office of the President
- Mark Rank, University of California, San Francisco
- Nick Roy, Penn State
- Ron Thielen, University of Chicago
- David Walker, Internet2
- Ann West, Internet2
- Jeff Whitworth, University of North Carolina at Greensboro

The authors of this version of the Cookbook would like to thank the authors of the original version of the AD Cookbook (based on IAP version 1.1) for their foundational work.

Version History

2011 May 5 - Initial work within the CIC CIOs Identity Management working group

2011 July 20 - CIC IdM Draft

2011 August 15 - InCommon Wiki Draft

2012 January 11 - InCommon Public Review Draft 1

2012 February 13 - Version 1.0

2012 March 26 - Version 1.1 (Post-IAM Online Feedback)

2012 August 9 - Minor changes to note that many of the approaches in the cookbook work from Windows Server 2003 forward

2013 June __ - Revisions to support IAP version 1.2 including alternative means statements, refactoring to de-duplicate controls and pair them with risk mitigations

2013 August - Revisions to add explicit IAP language interpretations, broader management assertion language and some additional reorganization.

2013 October 2 - Version 20131002-DRAFTreleased for community comment.

-->