

Understand Identity Service Engine (ISE) and Active Directory (AD)

Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[AD Protocols](#)

[Kerberos Protocol](#)

[MS-RPC Protocol](#)

[ISE integration with Active Directory\(AD\)](#)

[Join ISE to AD](#)

[Join AD domain](#)

[Leave AD domain](#)

[DC failover](#)

[ISE-AD communication through LDAP](#)

[User authentication against AD flow:](#)

[ISE SearchFilters](#)

Introduction

This document describes how Identity Service Engine (ISE) and Active Directory (AD) communicate, protocols that are used, AD filters, and flows.

Prerequisites

Requirements

Cisco recommends a basic knowledge of :

- ISE 2.x and Active Directory integration .
- External identity authentication on ISE.

Components Used

- ISE 2.x .
- Windows Server (Active Directory) .

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, ensure that you understand the potential impact of any command.

AD Protocols

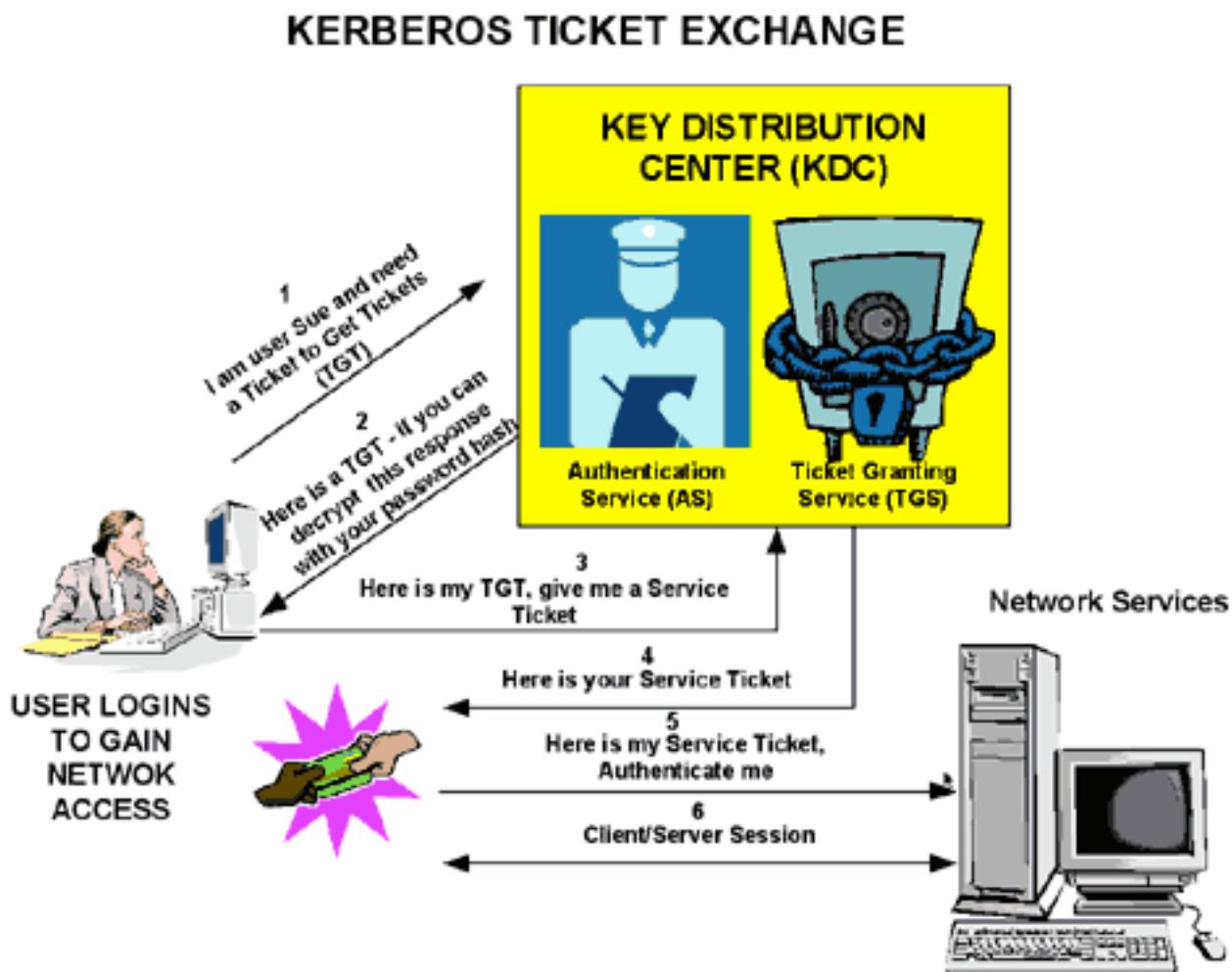
Kerberos Protocol

The three heads of Kerberos comprise the Key Distribution Center (KDC), the client user, and the server to access.

The KDC is installed as part of the Domain Controller (DC) and performs two service functions: The Authentication Service (AS) and the Ticket-Granting Service (TGS).

Three exchanges are involved when the client initially accesses a server resource:

1. AS Exchange.
2. TGS Exchange.
3. Client/Server (CS) Exchange.



- Domain Controller = KDC (AS + TGS).
- Authenticate to AS (the SSO portal) with your password.
- Get a Ticket Granting Ticket (TGT) (a session cookie).
- Request log in to a service (SRV01).
- SRV01 redirects you to KDC.
- Show TGT to KDC – (I am already authenticated)
- KDC gives you TGS for SRV01.

- Redirect to SRV01.
- Show service ticket to SRV01.
- SRV01 verifies/trusts service ticket.
- Service ticket has all my information.
- SRV01 logs me in.

When initially logged on to a network, users must negotiate access and provide a log-in name and password in order to be verified by the AS portion of a KDC within their domain.

The KDC has access to Active Directory user account information. Once authenticated, the user is granted a Ticket Granting Ticket (TGT) that is valid for the local domain.

The TGT has a default lifetime of 10 hours and is renewed throughout the user log-on session without the requirement of the user to re-enter his password.

The TGT is cached on the local machine in volatile memory space and is used to request sessions with services throughout the network.

The user presents the TGT to the TGS portion of the KDC when access to a server service is needed.

The TGS on the KDC authenticates the user TGT and creates a ticket and session key for both the client and the remote server. This information (the service ticket) is then cached locally on the client machine.

The TGS receives the client TGT and reads it with its own key. If the TGS approves of the client request, a service ticket is generated for both the client and the target server.

The client reads its portion with the TGS session key retrieved earlier from the AS reply.

The client presents the server portion of the TGS reply to the target server in the next client/server exchange.

Example:

Test User Authentication

* Username	user1
* Password	*****
Authentication Type	Kerberos

Authorization Data Retrieve Groups
 Retrieve Attributes

Test

Authentication Result	Groups	Attributes
<pre>Authentication time : 57 ms. Groups fetching time : 18 ms. Attributes fetching time: 4 ms. Processing Steps: 14:05:37:440: Resolving identity - user1 14:05:37:440: Search for matching accounts at join point - ralmaait.com 14:05:37:449: Single matching account found in forest - ralmaait.com 14:05:37:449: Identity resolution detected single matching account 14:05:37:476: Authentication Ticket (TGT) request succeeded - user1@ralmaait.com 14:05:37:478: Service Ticket request succeeded - user1@ralmaait.com 14:05:37:486: Service Ticket validation succeeded - user1@ralmaait.com 14:05:37:486: Account validation succeeded</pre>		

Packet captures from ISE for an authenticated user:

111 2020-01-13 16:17:53.082713	10.48.60.50	10.48.60.51	TCP	66 53610 → 88 [ACK] Seq=1 Ack=1 Win=29312 Len=0 TStamp=105462807 TSecr=280789807 ✓
112 2020-01-13 16:17:53.082735	10.48.60.50	10.48.60.51	KRB5	346 AS-REQ ✓
113 2020-01-13 16:17:53.083625	10.48.60.51	10.48.60.50	KRB5	1576 AS-REP ✓
114 2020-01-13 16:17:53.083649	10.48.60.50	10.48.60.51	TCP	66 53610 → 88 [ACK] Seq=281 Ack=1511 Win=32256 Len=0 TStamp=105462808 TSecr=2807.. ✓
115 2020-01-13 16:17:53.083678	10.48.60.50	10.48.60.51	TCP	66 53610 → 88 [FIN, ACK] Seq=281 Ack=1511 Win=32256 Len=0 TStamp=105462808 TSecr=2807.. ✓
116 2020-01-13 16:17:53.083988	10.48.60.51	10.48.60.50	TCP	66 88 → 53610 [ACK] Seq=1511 Ack=282 Win=532736 Len=0 TStamp=280789809 TSecr=105.. ✓
117 2020-01-13 16:17:53.084027	10.48.60.51	10.48.60.50	TCP	60 88 → 53610 [RST, ACK] Seq=1511 Ack=282 Win=0 Len=0 ✓
118 2020-01-13 16:17:53.084449	10.48.60.50	10.48.60.51	KRB5	1480 TGS-REQ ✓
119 2020-01-13 16:17:53.085475	10.48.60.51	10.48.60.50	KRB5	1446 TGS-REP ✓
120 2020-01-13 16:17:53.110397	10.48.60.50	10.48.60.51	TCP	66 48959 → 3268 [ACK] Seq=1700 Ack=536 Win=31360 Len=0 TStamp=105462835 TSecr=28.. ✓

The AS-REQ contains the username. If the password is correct, the AS service provides a TGT encrypted with the user password. The TGT is then provided to the TGS service to get a session ticket.

Authentication is successful when a session ticket is received.

This is an example where the password given by client is wrong:

117 2020-01-14 08:51:03.846603	10.48.60.50	10.48.60.51	KRB5	318 AS-REQ
118 2020-01-14 08:51:03.848340	10.48.60.51	10.48.60.50	KRB5	194 KRB Error: KRB5KDC_ERR_PREAUTH_FAILED

If the password is wrong the AS request fails and a TGT is not received:

Processing Steps:
13:19:55:837: Resolving Identity - User1
13:19:55:837: Search For Matching Accounts At Join Point - Ralmaait.com
13:19:55:843: Single Matching Account Found In Forest - Ralmaait.com
13:19:55:843: Identity Resolution Detected Single Matching Account
13:19:55:856: Authentication Ticket (TGT) Request Failed - User1@ralmaait.com,ERROR_PASSWORD_MISMATCH

Logs on the ad_agent.log file when password is wrong:

2020-01-14 13:36:05,442 DEBUG ,140574072981248,krb5: Sent request (276 bytes) to RALMAAIT.COM,LwKrb5TraceCallback(),lwadvapi/threaded/lwkrb5.c:1325

2020-01-14 13:36:05,444 DEBUG ,140574072981248,krb5: Received error from KDC: - 1765328360/Preattentation failed,LwKrb5TraceCallback(),lwadvapi/threaded/lwkrb5.c:1325

2020-01-14 13:36:05,444 DEBUG ,140574072981248,krb5: Preattentation tryagain input types: 16, 14, 19, 2,LwKrb5TraceCallback(),lwadvapi/threaded/lwkrb5.c:1325

2020-01-14 13:36:05,444 WARNING,140574072981248,[LwKrb5GetTgtImpl
.../lwadvapi/threaded/krbtgt.c:329] KRB5 Error code: -1765328360 (Message: Preattentation failed),LwTranslateKrb5Error(),lwadvapi/threaded/lwkrb5.c:892

2020-01-14 13:36:05,444 DEBUG ,140574072981248,[LwKrb5InitializeUserLoginCredentials()] Error code: 40022 (symbol:
LW_ERROR_PASSWORD_MISMATCH),LwKrb5InitializeUserLoginCredentials(),lwadvapi/threaded/lwkrb5.c:14

MS-RPC Protocol

ISE uses MS-RPC over SMB, SMB provides the authentication and does not require a separate session to find where a given RPC service is located. It uses a mechanism called “named pipe” to communicate between the client and server.

- Create an SMB session connection.
- Transport RPC messages over SMB/CIFS.TCP port 445 as a transport
- SMB session identifies which port a particular RPC service runs and handles user authentication.
- Connect to hidden share IPC\$ for inter-process communication.
- Open an appropriate named pipe for the desired RPC resource/function.

Transact the RPC exchange over SMB.

No.	Time	Source	Destination	Protocol	Length	Info	Text Item
59	2020-01-14 14:56:01.082699	10.48.60.50	10.48.60.51	SMB	128	Negotiate Protocol Request	✓
60	2020-01-14 14:56:01.083241	10.48.60.51	10.48.60.50	SMB2	318	Negotiate Protocol Response	✓
61	2020-01-14 14:56:01.083255	10.48.60.50	10.48.60.51	TCP	66	26963 → 445 [ACK] Seq=63 Ack=253 Win=30336 Len=0 TSval=186950807 TSecr=36227...	✓
72	2020-01-14 14:56:01.086109	10.48.60.50	10.48.60.51	SMB2	1589	Session Setup Request	✓
73	2020-01-14 14:56:01.086341	10.48.60.51	10.48.60.50	TCP	66	445 → 26963 [ACK] Seq=253 Ack=1586 Win=66560 Len=0 TSval=362277347 TSecr=186...	✓
74	2020-01-14 14:56:01.087051	10.48.60.51	10.48.60.50	SMB2	328	Session Setup Response	✓
75	2020-01-14 14:56:01.087260	10.48.60.50	10.48.60.51	SMB2	212	Tree Connect Request Tree: \\WIN-E051A81Q9BK.ralmaait.com\IPC\$	✓
76	2020-01-14 14:56:01.087592	10.48.60.51	10.48.60.50	SMB2	158	Tree Connect Response	✓
77	2020-01-14 14:56:01.087721	10.48.60.50	10.48.60.51	SMB2	206	Create Request File: netlogon	✓
78	2020-01-14 14:56:01.088823	10.48.60.51	10.48.60.50	SMB2	222	Create Response File: netlogon	✓
79	2020-01-14 14:56:01.088207	10.48.60.50	10.48.60.51	DCERPC	314	Bind: call_id: 9, Fragment: Single, 1 context items: RPC_NETLOGON V1.0 (32b...)	✓
80	2020-01-14 14:56:01.088500	10.48.60.51	10.48.60.50	SMB2	158	Write Response	✓
81	2020-01-14 14:56:01.088665	10.48.60.50	10.48.60.51	SMB2	183	Read Request Len:8192 Off:0 File: netlogon	✓
82	2020-01-14 14:56:01.088899	10.48.60.51	10.48.60.50	DCERPC	238	Bind_ack: call_id: 9, Fragment: Single, max_xmit: 4280 max_recv: 4280, 1 res...	✓
83	2020-01-14 14:56:01.089118	10.48.60.50	10.48.60.51	RPC_NETLOGON	574	NetLogonSamLogonEx request	✓
84	2020-01-14 14:56:01.089373	10.48.60.51	10.48.60.50	SMB2	158	Write Response	✓
85	2020-01-14 14:56:01.089517	10.48.60.50	10.48.60.51	SMB2	183	Read Request Len:8192 Off:0 File: netlogon	✓
86	2020-01-14 14:56:01.090160	10.48.60.51	10.48.60.50	RPC_NETLOGON	606	NetLogonSamLogonEx response	✓
88	2020-01-14 14:56:01.129364	10.48.60.50	10.48.60.51	TCP	66	26963 → 445 [ACK] Seq=2862 Ack=1635 Win=34688 Len=0 TSval=186950854 TSecr=36...	✓
145	2020-01-14 14:56:09.910387	10.48.60.50	10.48.60.51	RPC_NETLOGON	574	NetLogonSamLogonEx request	✓
146	2020-01-14 14:56:09.910734	10.48.60.50	10.48.60.51	SMB2	158	Write Response	✓

The negotiate protocol request/response line negotiates the dialect of SMB. The session setup request/response performs the authentication.

Tree Connect Request and Response connect to the requested resource. You are connected to a special share IPC\$.

This inter-process communication share provides the means of communication between hosts and also as a transport for MSRPC functions.

At packet 77 is **Create Request File** and the file name is the name of the connected service (the netlogon service in this example).

At packets 83 and 86, the NetrlogonSamLogonEX request is where you send the username for the client authentication on ISE to the AD at the field Network_INFO.

The NetrlogonSamLogonEX response packet replies with the results.

Some flags values for the NetrlogonSamLogonEX response:

0xc000006a is STATUS_WRONG_PASSWORD

0x00000000 is STATUS_SUCCESS

0x00000103 is STATUS_PENDING

ISE integration with Active Directory(AD)

ISE uses LDAP, KRB, and MSRBC to communicate with AD during the join/leave and authentication process.

The next sections provide the protocols, search format, and mechanisms used to connect to a specific DC on AD and user authentication against that DC.

In the event that the DC becomes offline for any reason, ISE fails over to the next available DC and the authentication process is not affected.

A Global Catalog server (GC) is a domain controller that stores copies of all Active Directory objects in the forest.

It stores a complete copy of all objects in the directory of your domain and a partial copy of all objects of all other forest domains.

Thus, the Global Catalog allows users and applications to find objects in any domain of the current forest with a search for attributes included to GC.

The Global Catalog contains a basic (but incomplete) set of attributes for each forest object in each domain (Partial Attribute Set, PAT).

The GC receives data from all the domain directory partitions in the forest. They are copied with the standard AD replication service.

Join ISE to AD

Prerequisites for Active Directory and ISE integration

1. Verify that you have the privileges of a Super Admin or System Admin in ISE.

2. Use the Network Time Protocol (NTP) server settings to synchronize the time between the Cisco server and Active Directory. The maximum allowed time difference between ISE and AD is 5 minutes
3. The configured DNS on ISE must be able to answer SRV queries for DCs, GCs, and KDCs with or without additional Site information.
4. Ensure that all the DNS servers can answer forward and reverse DNS queries for any possible Active Directory DNS domain.
5. AD must have at least one global catalog server operational and accessible by Cisco, in the domain to which you join Cisco.

Join AD domain

ISE applies Domain Discovery to get information about the join domain in three phases:

1. Queries joined domains—Discovers domains from its forest and domains externally trusted to the joined domain.
2. Queries root domains in its forest—Establishes trust with the forest.
3. Queries root domains in trusted forests—Discovers domains from the trusted forests.

Additionally, Cisco ISE discovers DNS domain names (UPN suffixes), alternative UPN suffixes and NTLM domain names.

ISE applies a DC discovery to get all information about the available DCs and GCs.

1. The join process starts with the input credentials of super admin on AD that exist in the domain itself. If it exists in a different domain or subdomain, the username must be noted in a UPN notation (username@domain).
2. ISE sends a DNS query for all DCs, GCs, and KDCs records. If the DNS reply did not have one of them in its answer then the integration fails with DNS related error.
3. ISE uses the CLDAP ping to discover all DCs and GCs through sent CLDAP requests to the DCs which correspond to their priorities in the SRV record. The first DC response is used and ISE is then connected to that DC.

One factor that is used to calculate the DC priority is the time taken by the DC to response to CLDAP pings; a faster response receives a higher priority.

 **Note:** CLDAP is the mechanism that ISE uses to establish and maintain connectivity with the DCs. It measures the response time until the first DC answer. It fails if you see no answer from DC. Warn if response time is bigger than 2.5 seconds. CLDAP ping all DCs in site (If no site then all DCs in domain). The CLDAP response contains DC site and Client site (the site to which ISE machine is assigned).

4. ISE then receives TGT with 'join user' credentials.
5. Generate ISE machine account name with the MSRPC. (SAM and SPN)
6. Search AD by SPN if ISE machine account already exists. If ISE machine does not exist, ISE creates a new one.
7. Open Machine account, set ISE machine account password, and verify ISE machine account is accessible.
8. Set ISE machine account attributes (SPN, dnsHostname, and the like).
9. Get TGT with ISE machine credentials with KRB5 and discover all trusted domains.
10. When the join is complete, ISE node updates its AD groups and associated SIDS and automatically starts the SID update process. Verify that this process can complete on the AD side.

Leave AD domain

When ISE leaves, the AD must consider:

1. Use a full AD admin user to perform the leave processes. This verifies that the ISE machine account is removed from the Active Directory database.
2. If the AD was left without credentials, then the ISE account is not removed from the AD and it must be deleted manually.
3. When you reset ISE configuration from the CLI or restore configuration after a backup or upgrade, it performs a leave operation and disconnects the ISE node from the Active Directory domain. (if joined). However, the ISE node account is not removed from the Active Directory domain.
4. It is recommended to perform a leave operation from the Admin portal with the Active Directory credentials because it also removes the node account from the Active Directory domain. This is also recommended when you change the ISE hostname.

DC failover

When the DC connected to ISE become offline or unreachable for any reason, DC failover is triggered automatically on ISE. DC failover can be triggered by the these conditions:

1. AD connector detects that the currently selected DC became unavailable during some CLDAP, LDAP, RPC or Kerberos communication attempt. In such cases, the AD connector initiates DC selection and fails over to the newly selected DC.
2. DC is up and responds to CLDAP ping, but AD Connector cannot communicate with it for some reason (examples: RPC port is blocked, DC is in ‘broken replication’ state, DC has not been properly decommissioned).

In such cases, the AD connector initiates DC selection with a blocked list (“bad” DC is placed in the blocked list) and tries to communicate with the selected DC. The DC selected in the blocked list is not cached.

AD connector must complete failover within reasonable time (or fail if it is not possible). For this reason, AD connector tries limited number of DCs during failover.

ISE blocks AD Domain Controllers if there is an unrecoverable network or server error to prevent ISE from the use of a bad DC. DC is not added to blocked list if it does not respond to CLDAP pings. ISE only lowers the priority of the DC which does not respond.

ISE-AD communication through LDAP

ISE searches for machine or user in AD with one of the these search formats. If the search was for a machine, then ISE adds “\$” at the end of the machine name. This is a list of Identity types which is used to identify a user in AD:

- SAM name: username or machine name without any domain markup, this is the User Logon Name in AD. **Example: sajeda or sajeda\$**
- CN: is the user display name on AD, it must not be same as the SAM. **Example: sajeda Ahmed.**
- User Principal Name (UPN): is a combination of the SAM name and the domain name (SAM_NAME@domian). **Example: sajeda@cisco.com or [sajeda\\$@cisco.com](mailto:sajeda$@cisco.com)**
- Alternative UPN: is an additional / alternative UPN suffixes that configured in the AD other than domain name. This configuration is added globally in the AD (not configured per user) and it is not necessary to be a real domain name suffix.

Each AD can have multiple UPN suffix(@alt1.com,@alt2.com,..., etc). **Example: main UPN**

(sajeda@cisco.com), alternative UPN :sajeda@domain1 , sajeda@domain2

- NetBIOS prefixed name: is the domain name\username of machine name. **Example:** CISCO\sajeda or CISCO\machine\$
- Host/prefix with unqualified machine: this is used for machine authentication when the machine name only is used, it is host/machine name only. **Example:** host/machine
- Host/ prefix with fully qualified machine: this is used for machine authentication when the Machine FQDN is used, usually in case of certificate authentication, it is host/FQDN of the machine. **Example:** host/machine.cisco.com
- SPN name: The name by which a client uniquely identifies an instance of a service, (examples: HTTP, LDAP, SSH) used for Machine only.

User authentication against AD flow:

1. Resolve Identity and determine identity type - SAM, UPN, SPN. If ISE receive the identity as a username only, then it searches for an associated SAM account in the AD. If ISE receives the identity as a username@domain, then it searches for a matched a UPN or mail in the AD. in both scenarios ISE uses additional filters for machine or username.
2. Search domain or forest (depends on identity type)
3. Keep information about all associated accounts (JP, DN, UPN, Domain)
4. If no associated account is found, then AD replies with user is unknown.
5. Perform MS-RPC (or Kerberos) authentication for each associated account
6. If only a single account matches to input identity and password, then authentication is successful
7. If multiple accounts match the incoming identity, then ISE uses the password to solve the ambiguity so that the account with an associated password is authenticated and the other accounts increase the incorrect password counter by 1.
8. If no account matches the incoming identity and password, then AD replies with wrong password.

ISE Search Filters

Filters are used to identify an entity that want to communicate with AD. ISE always searches for that entity in the users and machines groups.

Examples of search Filters:

1. **SAM search:** If ISE receives an identity as a username only without any domain markup, then ISE treats this username as a SAM and searches in AD for all machine users or machines that have that identity as a SAM name.

If the SAM name is not unique, ISE uses the password to differentiate between users and ISE is configured to use a passwordless protocol such as EAP-TLS.

There are no other criteria to locate the right user, so ISE fails the authentication with an “Ambiguous Identity” error.

However, if the user certificate is present in Active Directory, Cisco ISE uses binary comparison to resolve the identity.

219 2020-01-20 16:33:48.251918	10.48.60.206	10.48.60.101	LDAP	295 SASL GSS-API Integrity: searchRequest(2) "dc=aaalab,dc=com" wholeSubtree	✓
220 2020-01-20 16:33:48.253244	10.48.60.101	10.48.60.206	LDAP	384 SASL GSS-API Integrity: searchResEntry(2) "CN=anas Jehad,CN=Users,DC=aaalab,-	✓
258 2020-01-20 16:33:48.306966	10.48.60.206	10.48.60.101	LDAP	105	✓
<					
> Frame 219: 295 bytes on wire (2360 bits), 295 bytes captured (2360 bits)					
> Ethernet II, Src: Vmware_b6:ed:17 (00:50:56:b6:ed:17), Dst: Vmware_d5:6a:7d (00:0c:29:d5:6a:7d)					
> Internet Protocol Version 4, Src: 10.48.60.206, Dst: 10.48.60.101					
> Transmission Control Protocol, Src Port: 19997, Dst Port: 3268, Seq: 1430, Ack: 213, Len: 229					
Lightweight Directory Access Protocol					
SASL Buffer Length: 225					
SASL Buffer					
> GSS-API Generic Security Service Application Program Interface					
> GSS-API payload (197 bytes)					
LDAPMessage searchRequest(2) "dc=aaalab,dc=com" wholeSubtree					
messageID: 2					
protocolOp: searchRequest (3)					
searchRequest					
baseObject: dc=aaalab,dc=com					
scope: wholeSubtree (2)					
dereflAliases: neverDerefAliases (0)					
sizeLimit: 0					
timeLimit: 0					
typesOnly: False					
Filter: (&((objectCategory=person)(objectCategory=computer))(&(sAMAccountName=anos)))					
filters: and (0)					
and: (&((objectCategory=person)(objectCategory=computer))(&(sAMAccountName=anos)))					
and: 2 items					
Filter: ((objectCategory=person)(objectCategory=computer))					
and: or (1)					
or: ((objectCategory=person)(objectCategory=computer))					
Filter: (sAMAccountName=anos)					
and: itemMatch (3)					
equalityMatch					
attributeDesc: sAMAccountName					
assertionValue: anos					
attributes: 4 items					
AttributeDescription: sAMAccountName					
AttributeDescription: userPrincipalName					
AttributeDescription: objectCategory					
AttributeDescription: userAccountControl					

2. UPN or MAIL search: If ISE receives an identity as a username@domain, ISE searches each forest global catalogs for a match to that UPN identity or Mail identity “identity=matched UPN or email”.

If there is a unique match, Cisco ISE proceeds with the AAA flow.

If there are multiple join points with the same UPN and a password or the same UPN and Mail, Cisco ISE fails the authentication with an “Ambiguous Identity” error.

461 2020-01-20 16:33:58.134338	10.48.60.206	10.48.60.101	LDAP	336 SASL GSS-API Integrity: searchRequest(3) "dc=aaalab,dc=com" wholeSubtree	✓
464 2020-01-20 16:33:58.137942	10.48.60.101	10.48.60.206	LDAP	384 SASL GSS-API Integrity: searchResEntry(3) "CN=anas Jehad,CN=Users,DC=aaalab,-	✓
471 2020-01-20 16:33:58.170678	10.48.60.206	10.48.60.101	LDAP	179 SASL GSS-API Integrity: searchRequest(6) "CN=anas Jehad,CN=Users,DC=aaalab,D..	✓
472 2020-01-20 16:33:58.172663	10.48.60.101	10.48.60.206	LDAP	1413 SASL GSS-API Integrity: searchResEntry(6) "CN=anas Jehad,CN=Users,DC=aaalab,-	✓
476 2020-01-20 16:33:58.174754	10.48.60.206	10.48.60.101	LDAP	189 SASL GSS-API Integrity: searchRequest(7) "CN=anas Jehad,CN=Users,DC=aaalab,D..	✓
479 2020-01-20 16:33:58.175528	10.48.60.101	10.48.60.206	LDAP	255 SASL GSS-API Integrity: searchResEntry(7) "CN=anas Jehad,CN=Users,DC=aaalab,-	✓
480 2020-01-20 16:33:58.176236	10.48.60.206	10.48.60.101	LDAP	241 SASL GSS-API Integrity: searchRequest(8) "dc=aaalab,dc=com" wholeSubtree	✓
481 2020-01-20 16:33:58.177307	10.48.60.101	10.48.60.206	LDAP	635 SASL GSS-API Integrity: searchResEntry(8) "CN=Users,CN=Builtin,DC=aaalab,DC=..	✓
484 2020-01-20 16:33:58.178414	10.48.60.206	10.48.60.101	LDAP	271 SASL GSS-API Integrity: searchRequest(9) "dc=aaalab,dc=com" wholeSubtree	✓
<					
> Frame 461: 336 bytes on wire (2688 bits), 336 bytes captured (2688 bits)					
> Ethernet II, Src: Vmware_b6:ed:17 (00:50:56:b6:ed:17), Dst: Vmware_d5:6a:7d (00:0c:29:d5:6a:7d)					
> Internet Protocol Version 4, Src: 10.48.60.206, Dst: 10.48.60.101					
> Transmission Control Protocol, Src Port: 19997, Dst Port: 3268, Seq: 1659, Ack: 531, Len: 270					
Lightweight Directory Access Protocol					
SASL Buffer Length: 266					
SASL Buffer					
> GSS-API Generic Security Service Application Program Interface					
> GSS-API payload (238 bytes)					
LDAPMessage searchRequest(3) "dc=aaalab,dc=com" wholeSubtree					
messageID: 3					
protocolOp: searchRequest (3)					
searchRequest					
baseObject: dc=aaalab,dc=com					
scope: wholeSubtree (2)					
dereflAliases: neverDerefAliases (0)					
sizeLimit: 0					
timeLimit: 0					
typesOnly: False					
Filter: (&((objectCategory=person)(objectCategory=computer))(((userPrincipalName=anos@aaalab.com)(mail=anos@aaalab.com)))					
filters: and (0)					
and: (&((objectCategory=person)(objectCategory=computer))(((userPrincipalName=anos@aaalab.com)(mail=anos@aaalab.com)))					
and: 2 items					
Filter: ((objectCategory=person)(objectCategory=computer))					
and: or (1)					
or: ((objectCategory=person)(objectCategory=computer))					
Filter: (((userPrincipalName=anos@aaalab.com)(mail=anos@aaalab.com))					
and: or (1)					
or: (((userPrincipalName=anos@aaalab.com)(mail=anos@aaalab.com))					

3. NetBIOS search: If ISE receives an identity with a NetBIOS domain prefix (ex:CISCO\sajedah), then ISE searches in the forests for the NetBIOS domain. Once found, it then looks for the supplied SAM name (sajeda in our example)

Time	Source IP	Destination IP	Protocol	Details
654 2020-01-20 17:06:29.243747	10.48.60.206	10.48.60.101	LDAP	295 SASL GSS-API Integrity: searchRequest(2) "dc=aaaalab,dc=com" wholeSubtree ✓
655 2020-01-20 17:06:29.245154	10.48.60.101	10.48.60.206	LDAP	682 SASL GSS-API Integrity: searchResEntry(2) "CN=anas Jehad,CN=Users,DC=aaaalab,.. ✓
684 2020-01-20 17:06:29.290383	10.48.60.206	10.48.60.101	LDAP	179 SASL GSS-API Integrity: searchRequest(3) "CN=anas Jehad,CN=Users,DC=aaaalab,.. ✓
685 2020-01-20 17:06:29.292939	10.48.60.101	10.48.60.206	LDAP	1413 SASL GSS-API Integrity: searchResEntry(3) "CN=anas Jehad,CN=Users,DC=aaaalab,.. ✓
687 2020-01-20 17:06:29.294515	10.48.60.206	10.48.60.101	LDAP	189 SASL GSS-API Integrity: searchRequest(4) "CN=anas Jehad,CN=Users,DC=aaaalab,.. ✓
688 2020-01-20 17:06:29.295469	10.48.60.101	10.48.60.206	LDAP	255 SASL GSS-API Integrity: searchResEntry(4) "CN=anas Jehad,CN=Users,DC=aaaalab,.. ✓
689 2020-01-20 17:06:29.296186	10.48.60.206	10.48.60.101	LDAP	241 SASL GSS-API Integrity: searchRequest(5) "dc=aaaalab,dc=com" wholeSubtree ✓
692 2020-01-20 17:06:29.297557	10.48.60.101	10.48.60.206	LDAP	635 SASL GSS-API Integrity: searchResEntry(5) "CN=Users,CN=Builtin,DC=aaaalab,DC=.. ✓
693 2020-01-20 17:06:29.298761	10.48.60.206	10.48.60.101	LDAP	271 SASL GSS-API Integrity: searchRequest(6) "dc=aaaalab,dc=com" wholeSubtree
694 2020-01-20 17:06:29.299690	10.48.60.101	10.48.60.206	LDAP	650 SASL GSS-API Integrity: searchResEntry(6) "CN=Domain Users,CN=Users,DC=aaaala.. ✓

```

✓ SASL Buffer
  > GSS-API Generic Security Service Application Program Interface
  > GSS-API payload (197 bytes)
    < LDAPMessage searchRequest(2) "dc=aaaalab,dc=com" wholeSubtree
      messageID: 2
      protocolOp: searchRequest (3)
        < searchRequest
          baseObject: dc=aaaalab,dc=com
          scope: wholeSubtree (2)
          derefAliases: neverDerefAliases (0)
          sizeLimit: 0
          timeLimit: 0
          typesOnly: False
        < Filter: (&(|(objectCategory=person)(objectCategory=computer))(&sAMAccountName=anos))
          < filter: and (0)
            < and: (&(|(objectCategory=person)(objectCategory=computer))(&sAMAccountName=anos))
              < and: 2 items
                < and item: or (1)
                  < Filter: ((|(objectCategory=person)(objectCategory=computer))
                    < and item: or (1)
                      > or: ((|(objectCategory=person)(objectCategory=computer))
                        < Filter: (&sAMAccountName=anos)
                          < and item: equalityMatch (3)
                            > equalityMatch

```

4. Machine base search:

If ISE receives a machine authentication, with a host/prefix identity, then ISE searches the forest for a matched servicePrincipalName attribute.

If a fully-qualified domain suffix was specified in the identity, for example host/machine.domain.com, Cisco ISE searches the forest where that domain exists.

If the identity is in the form of host/machine, Cisco ISE searches all forests for the service principal name.

If there is more than one match, Cisco ISE fails the authentication with an “Ambiguous Identity” error.

Time	Source IP	Destination IP	Protocol	Details
2744 2020-01-20 16:35:32.108609	10.48.60.206	10.48.60.101	LDAP	373 SASL GSS-API Integrity: searchRequest(3) "dc=aaaalab,dc=com" wholeSubtree ✓
2745 2020-01-20 16:35:32.109744	10.48.60.101	10.48.60.206	LDAP	393 SASL GSS-API Integrity: searchResEntry(3) "CN=ISE24P,CN=Computers,DC=aaaalab,.. ✓
2747 2020-01-20 16:35:32.109951	10.48.60.206	10.48.60.101	LDAP	185 SASL GSS-API Integrity: unbindRequest(7) ✓
2757 2020-01-20 16:35:32.114862	10.48.60.206	10.48.60.101	LDAP	1495 bindRequest(1) "<ROOT>" sasl ✓
2758 2020-01-20 16:35:32.115898	10.48.60.101	10.48.60.206	LDAP	278 bindResponse(1) success ✓
2760 2020-01-20 16:35:32.116176	10.48.60.206	10.48.60.101	LDAP	348 SASL GSS-API Integrity: searchRequest(2) "dc=aaaalab,dc=com" wholeSubtree ✓
2761 2020-01-20 16:35:32.116855	10.48.60.101	10.48.60.206	LDAP	740 SASL GSS-API Integrity: searchResEntry(2) "CN=ISE24P,CN=Computers,DC=aaaalab,.. ✓
2762 2020-01-20 16:35:32.145535	10.48.60.206	10.48.60.101	LDAP	179 SASL GSS-API Integrity: searchRequest(3) "CN=ISE24P,CN=Computers,DC=aaaalab,.. ✓

Ethernet II, Src: VMware_b6:ed:17 (00:50:56:b6:ed:17), Dst: VMware_d5:6a:7d (00:0c:29:d5:6a:7d)

Internet Protocol Version 4, Src: 10.48.60.206, Dst: 10.48.60.101

Transmission Control Protocol, Src Port: 20089, Dst Port: 3268, Seq: 1746, Ack: 267, Len: 307

Lightweight Directory Access Protocol

SASL Buffer Length: 303

```

✓ SASL Buffer
  > GSS-API Generic Security Service Application Program Interface
  > GSS-API payload (275 bytes)
    < LDAPMessage searchRequest(3) "dc=aaaalab,dc=com" wholeSubtree
      messageID: 3
      protocolOp: searchRequest (3)
        < searchRequest
          baseObject: dc=aaaalab,dc=com
          scope: wholeSubtree (2)
          derefAliases: neverDerefAliases (0)
          sizeLimit: 0
          timeLimit: 0
          typesOnly: False
        < Filter: (&(|(objectCategory=person)(objectCategory=computer))(&sAMAccountName=ise24p$))
          < filter: and (0)
            < and: (&(|(objectCategory=person)(objectCategory=computer))(&sAMAccountName=ise24p$))
              < and: 2 items
                < and item: or (1)
                  < Filter: ((|(objectCategory=person)(objectCategory=computer))
                    < and item: or (1)
                      > or: ((|(objectCategory=person)(objectCategory=computer))
                        < Filter: (&sAMAccountName=ise24p$)
                          < and item: equalityMatch (3)
                            < equalityMatch
                              attributeDesc: sAMAccountName
                              assertionValue: ise24p$
```

 Note: Same filters are seen in ISE ad-agent.log files

 Note: ISE 2.2 patch 4 and prior and 2.3 patch 1 and prior identified users with the attributes SAM, CN, or both. Cisco ISE, release 2.2 Patch 5 and above, and 2.3 Patch 2 and higher, use only sAMAccountName attribute as the default attribute.