Microsoft

Active Directory Services

The foundations of hybrid identity

External resources disclaimer

This material includes links to external publicly available articles, projects, and research papers which are provided to you as a convenience and for informational purposes only.

Microsoft bears no responsibility for the accuracy, legality, content or any other aspect of the external site. Use of external hyperlinks does not constitute an endorsement by Microsoft of the linked content.

The external content referenced in this document belongs exclusively to their respective author(s). Inclusion in this presentation does not grant you with any right on the external content. You must comply with the original source’s applicable policies.

How to use this document

Why this document?

This document is provided as a companion of the video lessons. Additional information is included here which would not fit the video format or would exaggeratedly lengthen the videos. As you are watching the videos, the instructor will point you to additional content in this document.

Structure

The structure of this slide deck follows the structure of the lessons. One slide deck is provided for each module. The slide deck has the same structure (naming of chapters and sections) than the video so that you can quickly jump to the slides associated with the lesson you're currently watching.

Foreword

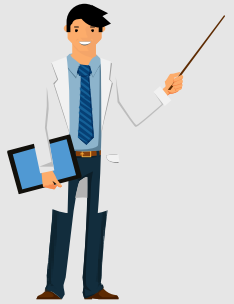
This deck contains some design artefacts which all have their importance…

This sticky note icon is used to introduce the **abbreviation** of a concept or a technical word. Once the abbreviation has been introduced, the full version is no longer mentioned.

You will also find a list of all abbreviations at the end of the deck.

We were all young once. A section with this icon will basically tell you the **history** you might have missed by not working with the technology for the last 20 years.

It is not because you are new that you don't have to know how we got here!



Professor Useful will introduce some **tricky technical details** which might not seem relevant at first but could end up being really useful if you want to dig deeper in the technology.

This frame contains…

- Takeaways so important that we framed them

How to know the slide level

This deck contains 3 different content levels:

1. Regular level, the common slide
2. Advanced level, a slide with this indicator at the top left 

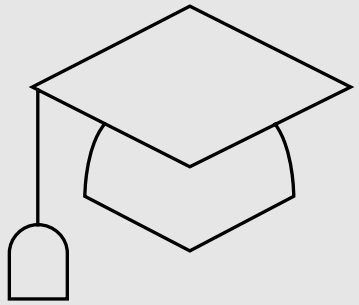
3. Additional content, all hidden slides

Section

1

Active Directory Services

Learning Objectives



Describe the major components of an Active Directory environment.

Agenda

1. Introduction of the main functions of AD
2. Fundamentals of AD architecture
3. Integration of machines in AD
4. Account authentication
5. The domain controller discovery mechanism
6. Object management
7. Access control in AD
8. The default administrators
9. The use of the LDAP protocol to query directory data

Chapter

1.1.1

Introduction of the main functions of AD

🎯 Identify the main functions of AD and discuss the criticality of AD in an information system

Why Active Directory?

A user opening an interactive session on a Windows machine

- The user is even prompted to change its password according to a specific policy
- The user environment is customized (warning messages, display name, custom desktop…)

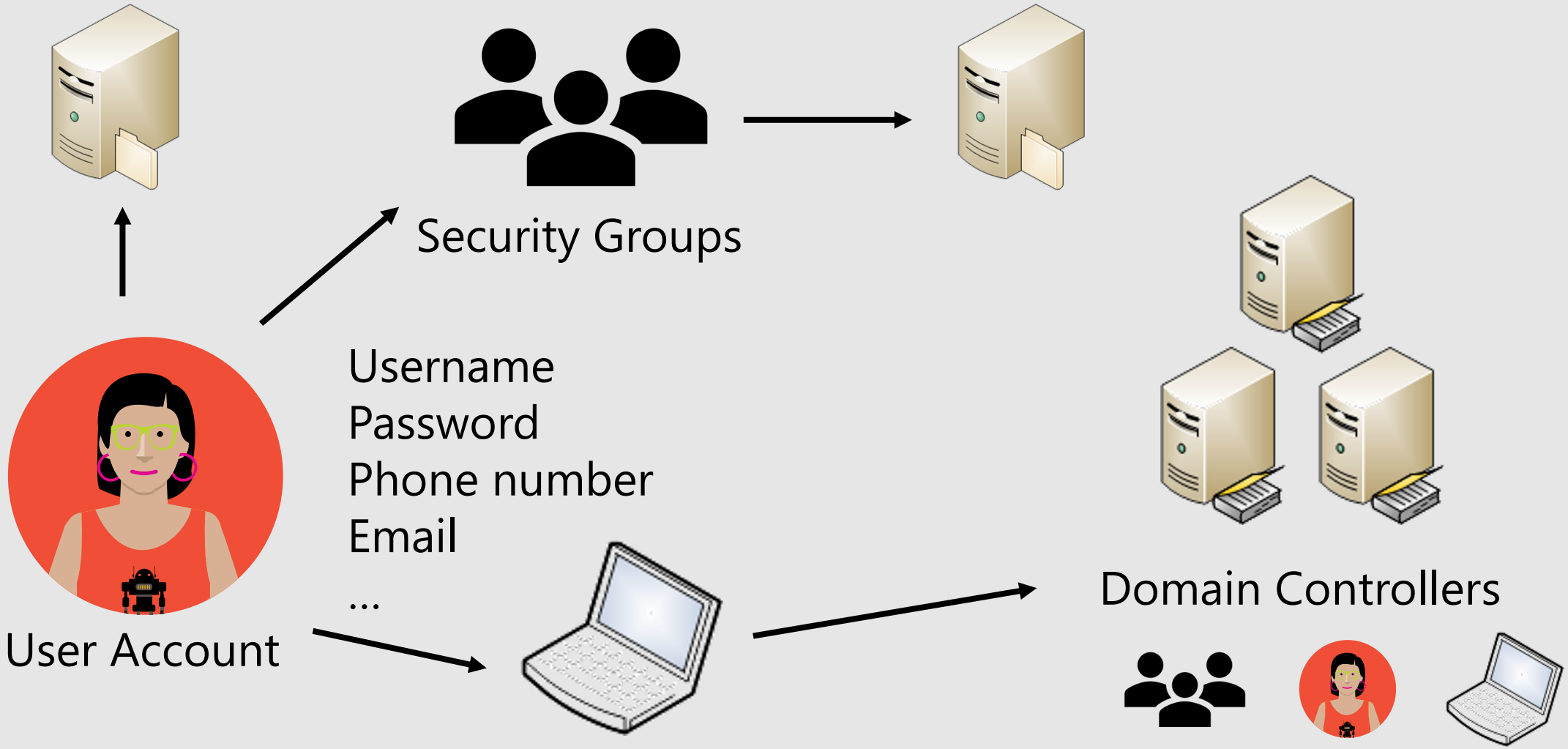
A user accessing a file share or an App somewhere on the network

- The user is not prompted for password, single sign-on
- The user does not have access to all the files

This whole world is managed by Active Directory

- Authentication (users and systems, SSO)
- Identity management (information and authorization)
- System management (security policies, customizations)

Directory and identity



Let's clarify what we are talking about…

Active Directory what?

- Active Directory Domain Services
- Active Directory Lightweight Directory Services
- Active Directory Certificate Services
- Active Directory Federation Services
- Active Directory Right Management Services

It's a Windows role

- No need for a specific license
- In the Windows world: OS role ≠ feature

Misconceptions

It's only a Windows thing!

Nothing is standard, it's only custom…

Alright then, it's just a big LDAP server!

Active Directory core

Active **D**irectory is a **D**irectory **S**ervice

AD~~DS~~

- it is used to provide a central store for identity and account information
- It used to store information for other systems and applications
- Based on X500 Directory

AD DS operate in 2 distinct modes:

AD LDS

- Active Directory Lightweight Directory Services
- Active Directory Domain Services

As a central Authority, ADDS is a critical service in the company environment

- Authentication (AuthN) & authorization are provided by ADDS

Once upon a time...

X500, What is it this!

- X.500 is a specification from the International Telecommunication Union (ITU) that specifies a global, hierarchical directory service.
- Based on 3 main components:
  - Directory Information Base (DIB): the data base
  - Directory System Agent (DSA): the server who host the DIB
  - Directory User Agents (DUA): Used to access information form the DSA
- a DUA uses Directory Access Protocol (DAP) to query and connects to a local DSA
- The Lightweight Directory Access Protocol (LDAP
  - It's a simplified version of DAP
  - LDAP was developed by the University of Michigan for use on TCP/IP networks
- X.500 forms is the basis of ADDS, the directory service of Microsoft Exchange Server, and Novell Directory Services (NDS).

Active Directory Authentication

Authentication is a process for verifying the identity of an object, service or person.

The goal is to verify that the credentials presented are authentic.

- If yes, ADDS return an “authentication token”

The Windows operating system implements a default set of authentication protocols, including:

- Kerberos v5
- NTLM
- Transport Layer Security/Secure Sockets Layer (TLS/SSL)
- Digest (deprecated)

ADDS Terminology

A domain controller

DC

- Physical entry point

A domain

- Administration and replication boundary

A forest

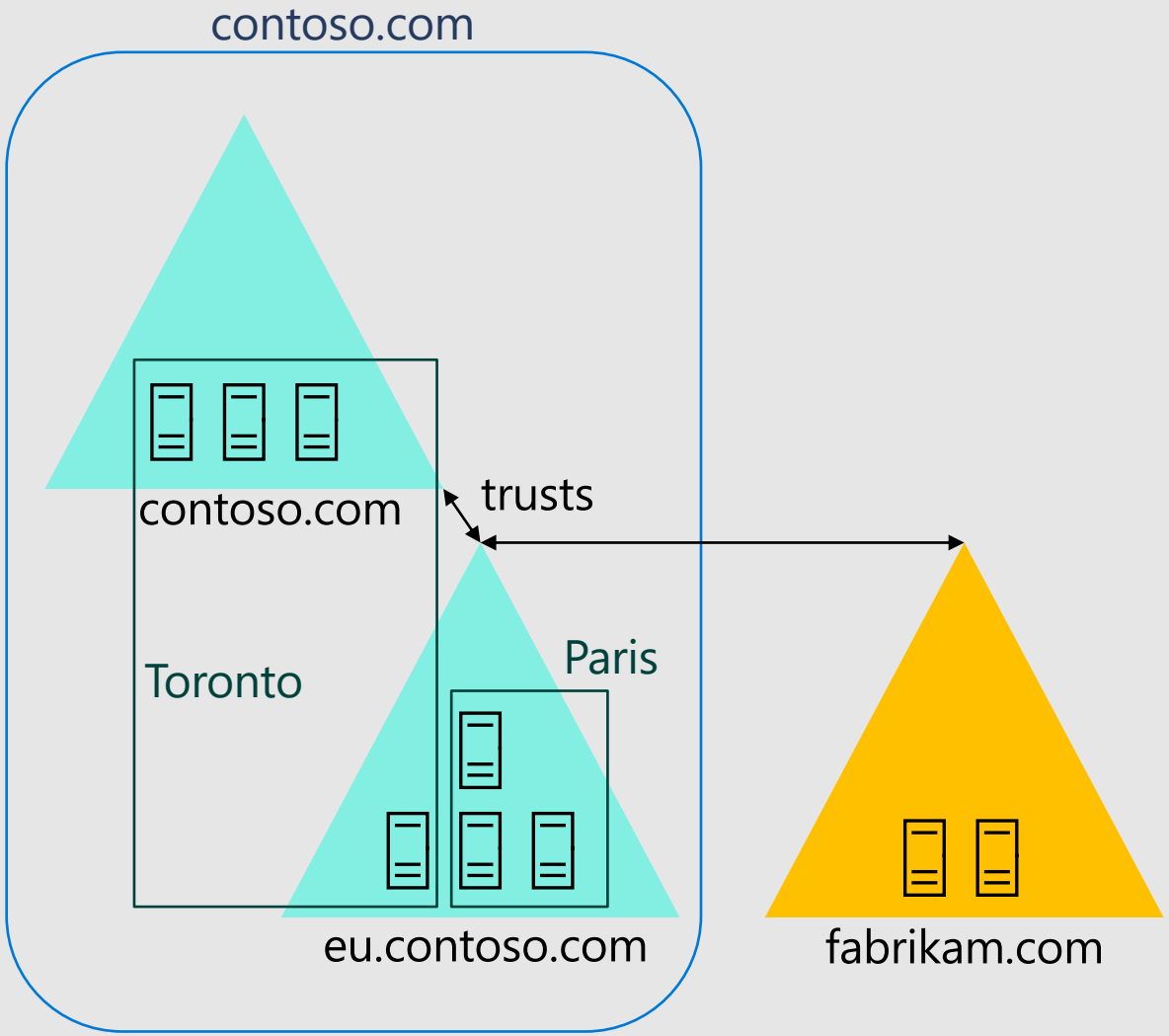
- A security boundary

A trust

- Extension of the security boundary

A site

- A representation of the physical network topology



What's on a domain controller?

A database on the disk C:\Windows\NTDS\ntds.dit

- The database replicates across domain controllers in a multi-master model
- It's encrypted, with a key you can find in:
  - C:\Windows\System32\config\SYSTEM

A shared folder containing policies:

- C:\Windows\SYSVOL
- Users and machines will connect to this folder to retrieve the configuration they need to apply

Daemons offering network services:

It's also the critical services of ADDS

- File Services
- Kerberos authentication
- DNS (Domain Name Server)
- NTP (Time Service for all devices)

How do I get one?

As a Windows role, any Windows Server can become a DC

- DC promotion process is often referred as “DCPromo”
- You can use the GUI via the Server Manager
- You can use PowerShell
- It can be done locally or remotely

Additional DCs, new domain, new forest

- At the promotion time, you choose the required scenario (adding a DC to an existing domain or creating a new domain or even a new forest)
- When added to an existing domain, an initial replication will take place

Let's clarify

Within a domain, objects are organized hierarchically

LDAP is the main interface for object query

- The LDAP protocol can be used to query AD

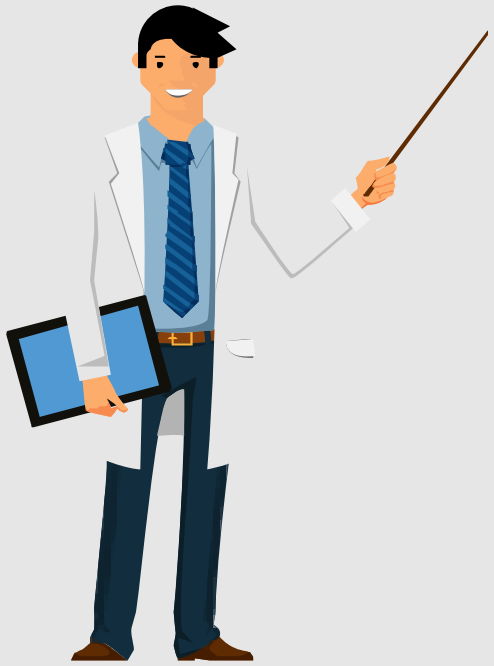
The top of an LDAP tree is a Naming Context NC

- Follow the X500 model
- The top of the tree of the domain contoso.com is DC=contoso,DC=com

- DC=contoso,DC=com notation is called the **D**istinguished **N**ame DN
- Containers such as **O**rganizational **U**nits can be created to organize objects
- Nothing is case sensitive

OU

Professor Useful



.dit?

- Stands for Directory Information Tree
- That's how we roll in a X500 type of directory

AD database is using an ESE engine

- It uses transaction logs

It's a B-Tree

- Hierarchical index to minimize disk IO
- Modifications are performed on partially full blocks
- Balanced indexes

NTDS.DIT has everything

- Everything also means the users' secret keys

Once upon a time…

Why DCPromo?

DCPromo.exe is a tool used to promote a regular server to a DC

Still present on Windows Server 2016 but only for unattended installation

Before DCpromo?

DCPromo is a “new” feature of Windows 2000 Server

In Windows NT, the decision to be a DC had to be made at the OS installation

Windows NT?

Yes, that was the version of Windows a long time ago… Up until 1999

At this time replication was not multi-master

Different versions, different functions

A domain has a functional level DFL

A forest has a functional level FFL

- You select them during the first DC's promotion

It defines the level of functionality available in the domain and the forest

- higher the level is, the more functionalities you have

It can be manually changed under certain conditions

- The OSes of all DCs of the domain have to be at least Windows Server 2012 R2 to raise the DFL to Windows Server 2012 R2
- The OSes of all DCs of all the domains in the forest have to be at least Windows Server 2012 R2 to raise the FFL to Windows Server 2012 R2

Which versions for which functional level

DFL	FFL
Windows 2000 Server Mixed	Windows 2000 Server Mixed
Windows 2000 Server NT4 Transition	
Windows 2000 Server Native	Windows 2000 Server Native
Windows Server 2003	Windows Server 2003
Windows Server 2008	Windows Server 2008
Windows Server 2008 R2	Windows Server 2008 R2
Windows Server 2012	Windows Server 2012
Windows Server 2012 R2	Windows Server 2012 R2
Windows Server 2016	Windows Server 2016

Chapter

1.1.2

Fundamentals of AD architecture

🎯 Identify the logical and physical components of an AD environment

Here it all begins - Active Directory logical model

The top-level container is the forest

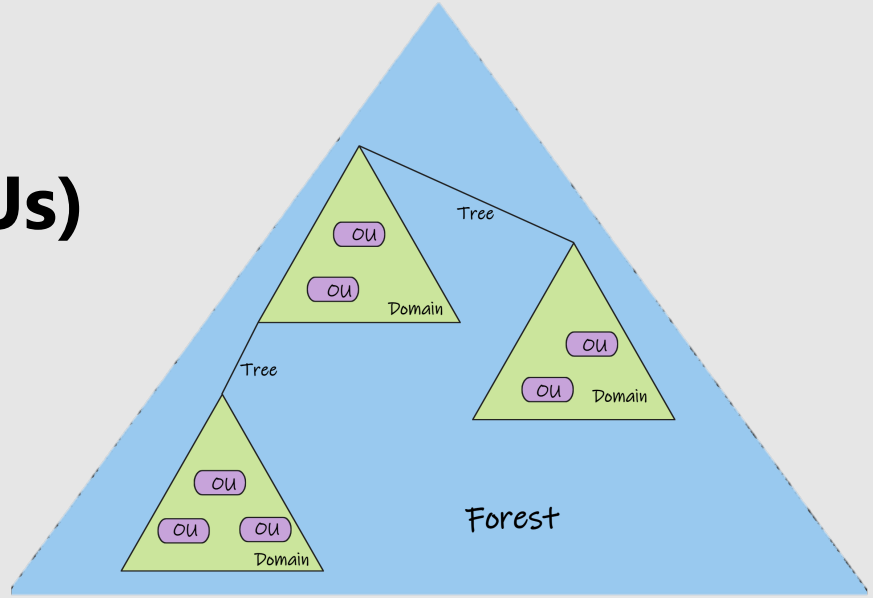
- Logical construct used by AD DS to group one or more domains that share common structure

Within forests are domains

- A domain is a partition in an Active Directory forest.
- Partitioning data enables organizations to replicate data only to where it is needed
- Provide authentication services

Within domains are organizational units (OUs)

- Used to form a hierarchy of containers within a domain.
- Used to group objects for administrative purposes



The ADDS Site Topology

Logical representation of your physical network

Components used in AD Sites & Services to ensure efficient replication traffic & Authentication:

- DC placement
- ADDS sites
- Site subnets
- Site links
- Site link bridges

Note: AD Site topology can be used by applications or services to improve their functionalities (like Distributed File Services)

Domain Controllers

Read Write Domain Controllers

- Standard or full Domain Controllers
- Replicate all objects & all attributes

Read Only Domain Controllers

- For use at branch offices for security concern
- Replicate all objects but:
  - Don’t replicate by default confidential attributes

Directory Object Types

Directory Objects is the most fundamental in the Active Directory system

Main Directory Object Types:

- Users
- Groups
- Computers
- Organizational Units
- Group Policy Objects

Naming Contexts

Naming contexts are trees containing the objects

- They are a replication boundary
- Some of them replicate on every DC of the forest, some of them on every DC of the domain, some of them only on the DCs which also hold the role of DNS server

Domain NC	DC=contoso,DC=com	All DCs in the domain Contains domain's object
Configuration NC	CN=configuration,DC=contoso,DC=com	All DCs in the forest Contains sites configuration
Schema NC	CN=schema,CN=Configuration,DC=contoso,DC=com	All DCs in the forest
Forest DNS NC	CN=ForestDNSZones,DC=contoso,DC=com	All DCs in the forest which are also DNS servers
Domain DNS NC	CN=DomainDNSZones,DC=contoso,DC=com	All DCs in the domain which are also DNS servers

AD Common Partition

Domain Partition contain all objects for the Domain operation

- Authentication
- Authorization
- Security Baseline, ...

Configuration Partition contain all objects for the Forest operation

- Services configuration
  - PKI
  - DHCP
  - Azure, ...
- AD Sites, ...

Schema Partition contain all class and attributes for the Forest object creation

Users

Groups, ...

The global catalog, eh?

All domain controllers should have the Global Catalog feature enabled

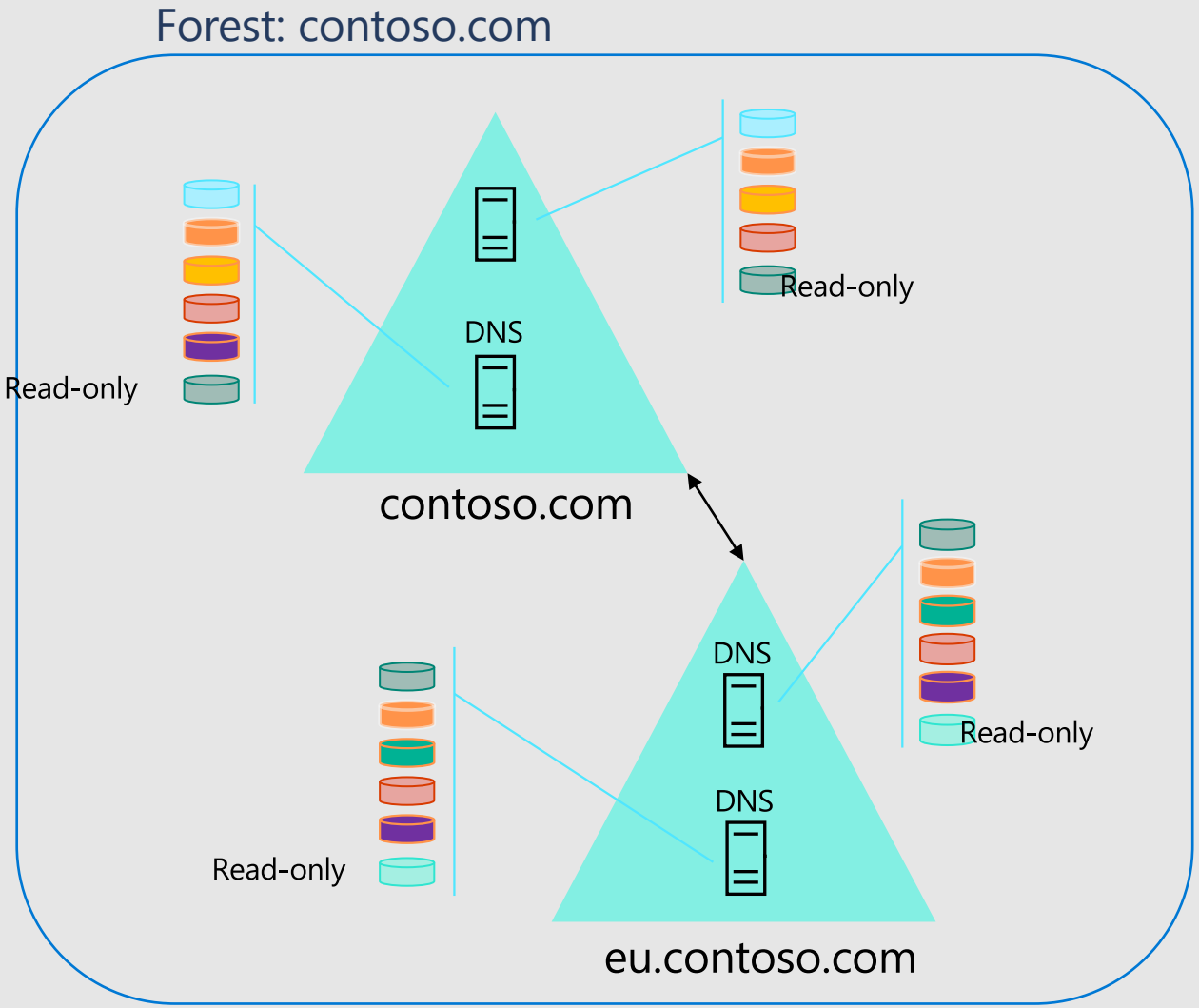
GC

- It is a choice at promotion
  - GC enabled is the Default
- Required to be able to authenticate

No global catalog = No authentication

- In a multi-domain forest, they hold a read-only copy of all domain NCs, so that we can interrogate any global catalog about anything, they all have the same level of knowledge of the forest

Let's clarify



DC=contoso,DC=com

DC=eu,DC=contoso,DC=com

CN=configuration,DC=contoso,DC=com

CN=schema,CN=configuration

CN=forestdnszones,DC=contoso,DC=com

CN=domaindnszones,DC=contoso,DC=com

CN=domaindnszones,DC=eu,DC=contoso,DC=com

A primordial name resolution!

Active Directory requires name resolution infrastructure to provide:

- A name resolution service that enables network hosts and services to locate Active Directory domain controllers
- A naming structure that enables an enterprise to reflect its organizational structure in the names of its directory service domains

DNS provides Active Directory with both name resolution and structure

DNS zone can be replicated by Active directory as Objects

- Using application partitions

Users, groups, computers

- **These objects are called security principals**
  - They can be granted permissions on objects and privileges on systems
  - They have a unique object**SID**

SID

  - We use SID in security descriptors, not names

objectSID

R**elative** ID**entifier**

- The last part of the SID is called RID

An objectSID is domain SID + RID

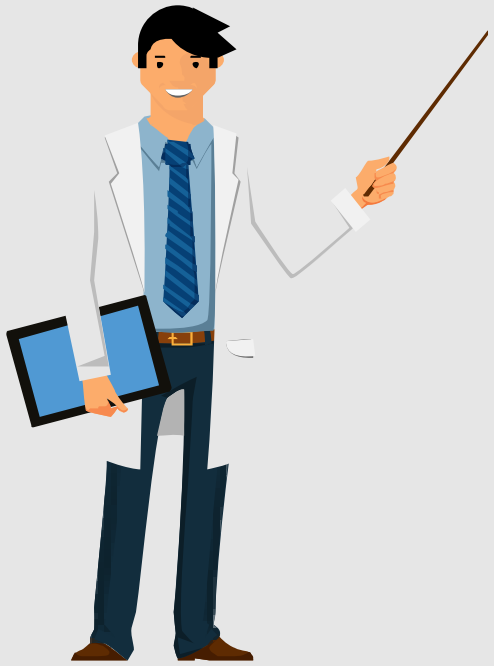
- Each DC has two pools of 500 RIDs
- A DC gives the next available RID to the next principal created
- RID under 1000 are called well-known RID/SID

Well-known RIDs and SIDs

SID	Display Name
S-1-5-1	Dialup
S-1-5-2	Network
S-1-5-3	Batch
S-1-5-4	Interactive
S-1-5-5-X-Y	Logon Session
S-1-5-6	Service
S-1-5-7	Anonymous Logon
S-1-5-8	Proxy
S-1-5-9	Enterprise Domain Controllers
S-1-5-10	Self
S-1-5-11	Authenticated Users
S-1-5-12	Restricted
S-1-5-13	Terminal Server User
S-1-5-14	Remote Interactive Logon
S-1-5-18	System (or LocalSystem)
S-1-5-19	LocalService
S-1-5-20	NetworkService

SID	Display Name
S-1-5-domain-500	Administrator
S-1-5-domain-501	Guest
S-1-5-domain-502	krbtgt
S-1-5-domain-512	Domain Admins
S-1-5-domain-513	Domain Users
S-1-5-domain-514	Domain Guests
S-1-5-domain-515	Domain Computers
S-1-5-domain-516	Domain Controllers
S-1-5-domain-517	Cert Publishers
S-1-5-root domain-518	Schema Admins
S-1-5-root domain-519	Enterprise Admins
S-1-5-domain-520	Group Policy Creator Owners
S-1-5-domain-553	RAS and IAS Servers

Professor Useful



The RID of an object can tell us many things

- Very low RID = very old object
- Two objects with consecutive RIDs are likely to have been created on the same DC

The RID pool size can change!

- By default it is 500… You can make it 1 billion… But don't…

You can have up to 1 billion RIDs!

- Well 1,073,741,823, and up to 2 billion starting Windows Server 2012 (well 2,147,483,647)

Objects and attributes

Objects have a bunch of attributes you would expect

Name, DisplayName, dNShostname, memberOf, manager, city…

Computer and users also have an attribute called userAccountControl

It is a binary mask that enables/disabled options on the account (such as the account is enabled, or the password does not expire, or the account can use DES only for Kerberos etc…)

Computers

They also open sessions and do things on the network

- They have their own identity, their own account in AD
- They have an objectSID
- They have a sAMAccountName which their hostname followed by **$**

You need a computer account for your machine if you want Windows SSO

Groups

You add principals in a group and assign permissions to a group

- Groups have objectSID too

There are different types of groups

- Security groups: you can use them to give permissions
- Distribution groups: you can use them only in a messaging system

Groups also have a scope

- Domain local groups: can contain users from anywhere in the forest but can be used only on the domain where it exists
- Global groups: can contain users only from the domain where it exists but can be used anywhere in the forest to give permissions
- Universal groups: can contains users from anywhere in the forest and can be used to give permissions anywhere in the forest

Trust Overview

Provide authentication and authorization capabilities between clients and servers in different domains

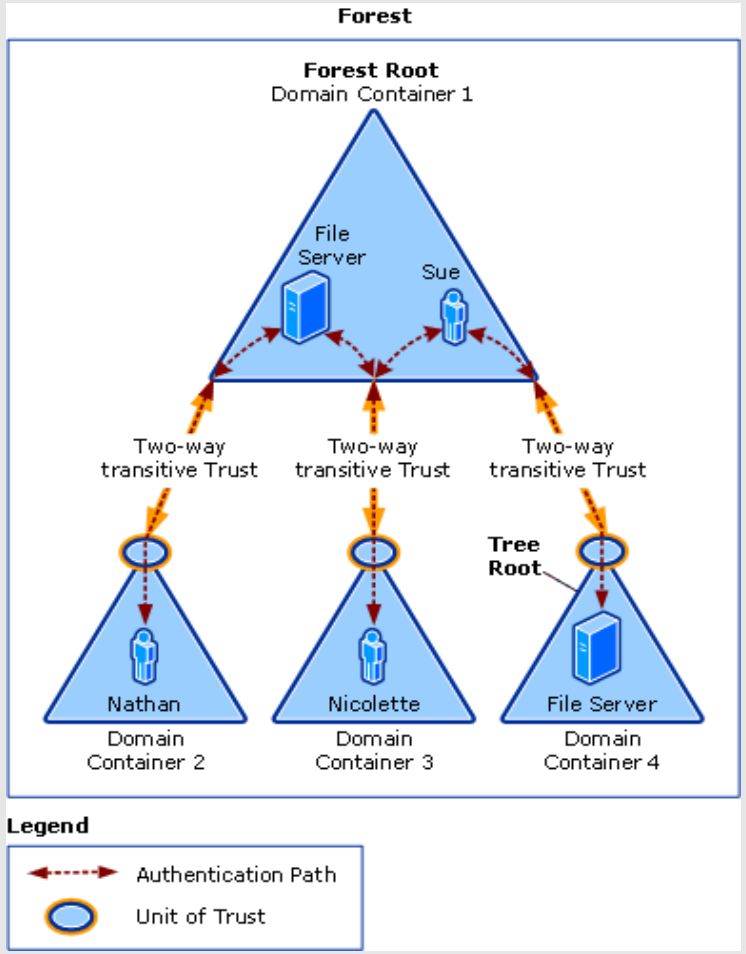
Trust acts as authentication pipeline.

One-way and two-way trusts

- One-way: unidirectional authentication path created between two domains
- Two-way: bidirectional authentication requests can be passed between the two domains in both directions

Transitive Trust or not?

- Transitivity = extend Trust outside of the two domains



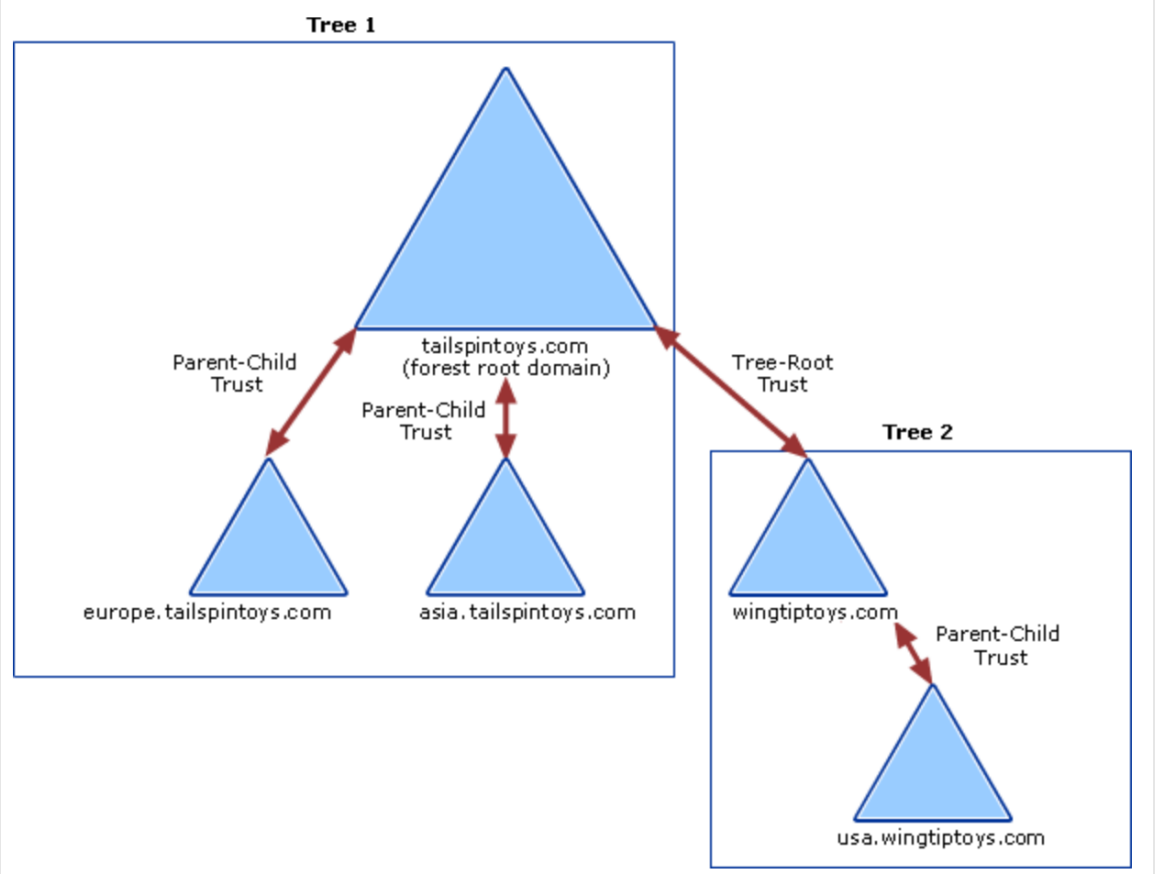
I want more Trust overview

Types

- Within the same forest
  - Parent-Child trusts
  - Tree-Root trusts
- With other forests
  - Forest trusts
  - External trusts
  - Realm trusts

Options

- Sid filtering/Quarantine
- Selective Authentication
- …



Trust Types

Forest Trust:

- Forest trust enables a transitive trust between all of the domains in two forests.
- The trust is established between the root domains of both forests.

External trust

- A nontransitive trust created between 2 domains into different forests
- Partial support of Kerberos

Shortcut trust

- A transitive trust created between 2 domains into the same forest

Realm trust

- A transitive trust between an Active Directory domain and a Kerberos V5 realm

Trusts security “Options”

Forest-Wide Authentication (Forest Trust)

- Allows to limit the scope of trust-transitivity

Selective Authentication

- No authentication allowed by default, fine delegation
- Must explicitly granted

Sid filtering (External Trust) /Quarantine (Forest Trust)

AD is highly available

DCs replicate with each other in a multi-master model

- If one DC is not available, others can take up the work

This is a loose consistency model

- We don't commit the change on every DC before validating it
- Monitoring replication is critical
- We keep track of changes at the attribute level

Replication is following a notify and pull model

- When one DC gets a change, it notifies the other DCs, the other DCs will pull for the change

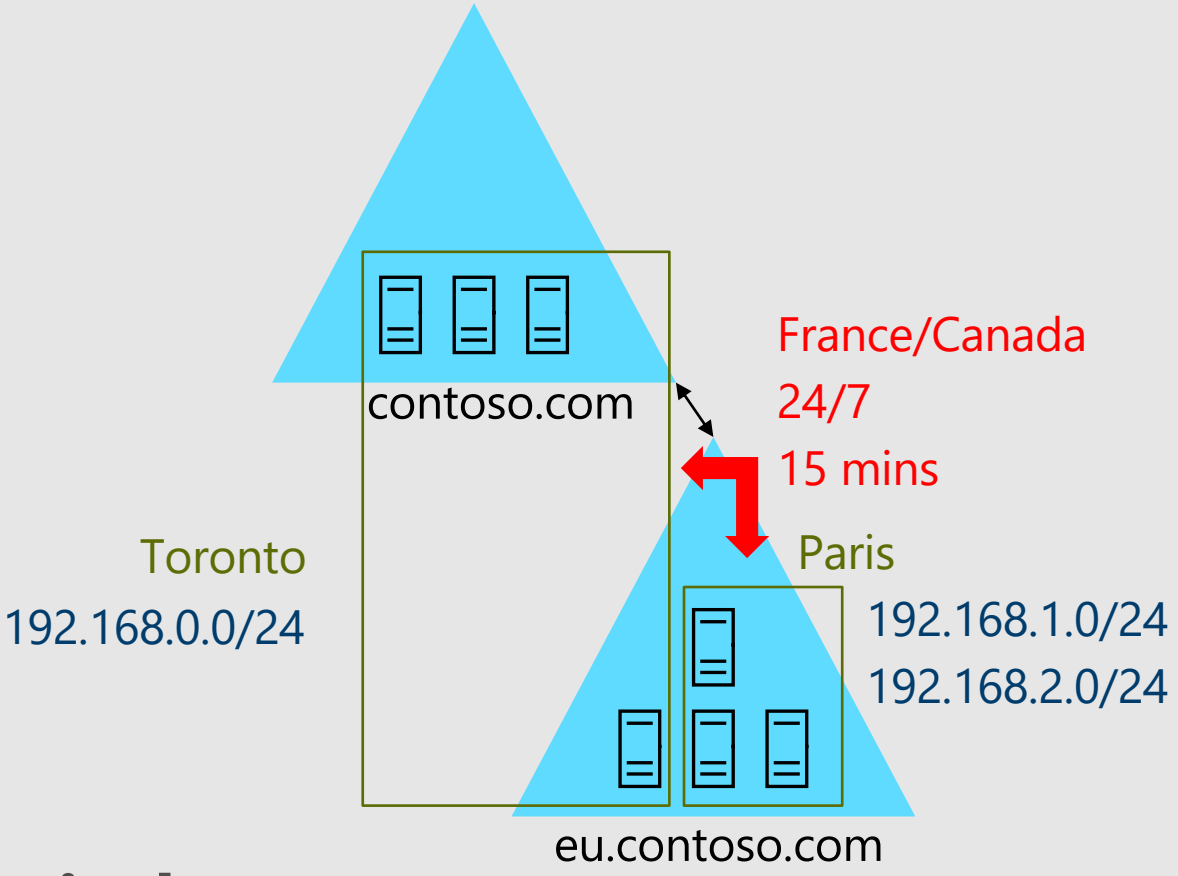
And a store-and-forward for DCs in different geographical locations

- We keep track of what we have replicated with whom (state-based replication) to optimize replication traffic

Replication topology

How do DCs know who to notify?

- You define it! You create your site topology
- Create **sites**
- Create **site links**
- Create **subnets**



DCs in the same site are notified within seconds

DCs in a different site are pulling their data according to a defined schedule and interval

All this information is stored in the configuration NC

- All DCs are aware of it

Replication topology

Each DC has a local component called the Knowledge Consistency Checker

KCC

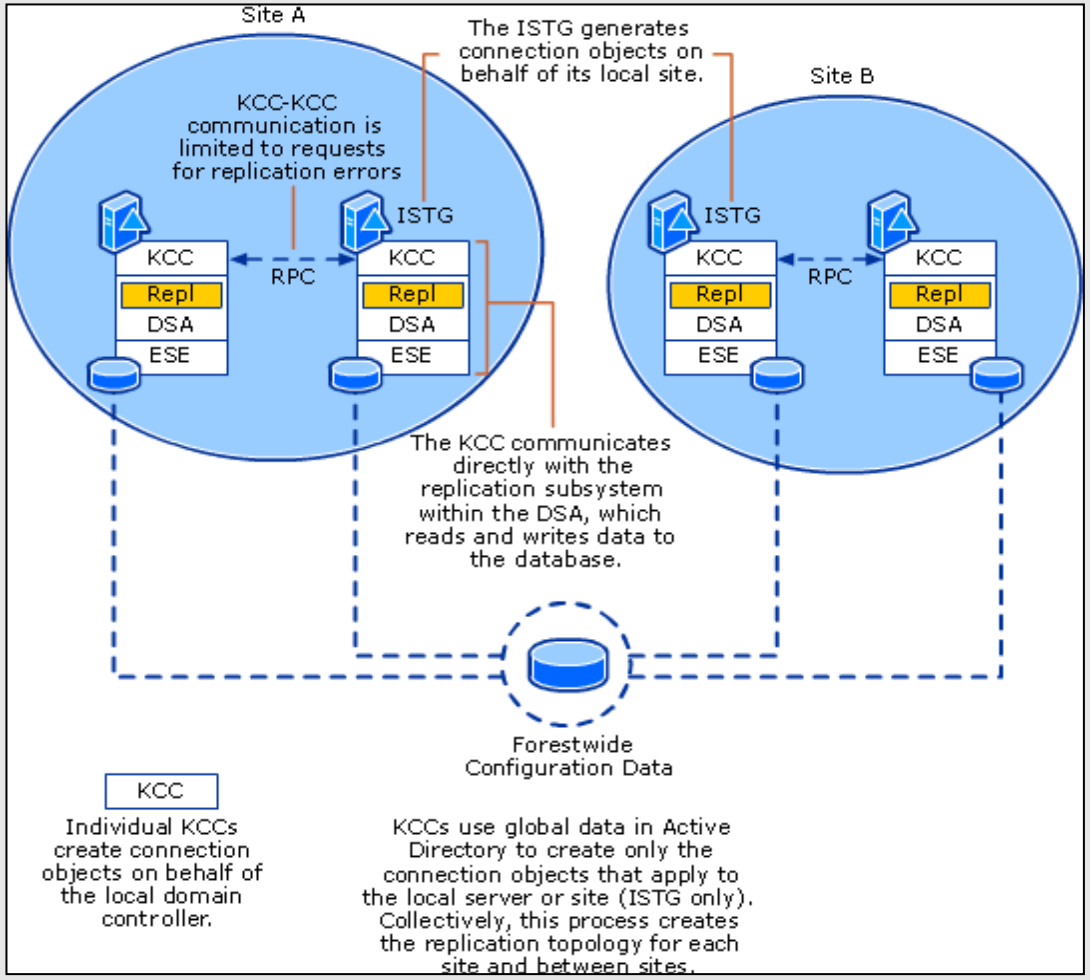
- It takes care of creating the connection objects for NCs between DCs
- Everything is automated (although you can override it)

One DC per site is elected bridgehead for a specific NC

- It will handle the inter-site replication according to the site topology defined in the configuration NC
- By default the first that shows up, but it can change, and can be overridden
- It is decided by the DC who has the role of **I**nter-**S**ite **T**opology **G**enerator which is the first DC of the site, and can also change

ISTG

Replication topology (cont.)



Component	Description
KCC	The application running on each domain controller that communicates directly with the Ntdsa.dll to read and write replication objects
Directory System Agent (DSA)	The directory service component that runs as Ntdsa.dll on each domain controller, providing the interfaces through which services and processes, such as KCC, gain access to the directory database
Extensible Storage Engine (ESE)	The directory service component that runs as Esent.dll. ESE manages the tables of records, each with one or more columns. The directory database is composed of the tables of records
Remote Procedure Call (RPC)	The Directory Replication Service (DRS) (Drsuapi) RPC protocol is used to communicate replication status and topology to a domain controller. The KCC also uses this protocol to communicate with other KCCs to request error information when building the replication topology
Inter-Site Topology Generator (ISTG)	The single KCC in a site that manages intersite connection objects for the site

Once upon a time…

Urgent replication!

- Back in the Windows 2000 Server days, the default delay for notifications for DCs within the same site was up to 5 minutes
- To bypass this, changes in the NC that have a security impact were bypassing the delay and replicating urgently (account lockout for example)
- It does still exist, but now that intra-site delays are just a few seconds, the urgent flag does not change things much
- Note that urgent replication never crosses site-links, in other words, when an account is locked-out for example, this is replicated right away on all DCs of the same site, but will wait for the interval/schedule for inter-site

Keeping track of things

Because we replicate at the attribute level, we need to keep track of at least two things

- Did something change on the database of a domain controller
- If yes, what attribute changed

For the first one we have Update Sequence Numbers

USN

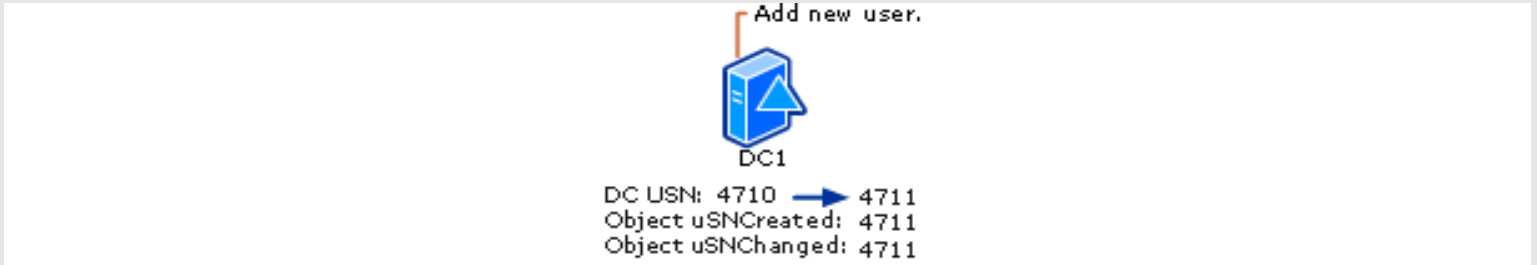
- It's a counter incremented by 1 at each change
- Each DC has its own counter
- DCs know what is the USN of each other to keep track if they already replicated a change

For the second, we have replication metadata

- It tracks from where an attribute has been changed (DC and USN)
- It tracks when and how many times the value has changed

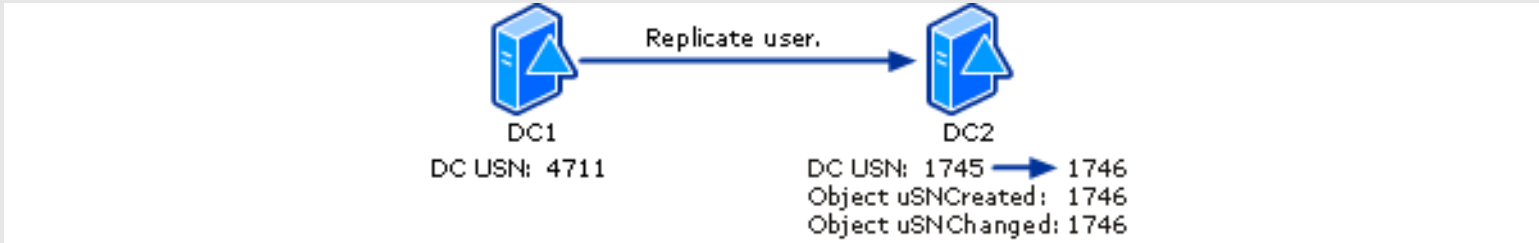
The replication protocol in detail

Object Creation



Property	Value	Local USN	Version	Originating Time	Originating DC	Originating USN
cn	Jeff Smith	4711	1	2003-09-10 10:49.03	<DC1_GUID>	4711
userPassword	6Be8W5q-	4711	1	2003-09-10 10:49.03	<DC1_GUID>	4711
sAMAccountName	JSmith	4711	1	2003-09-10 10:49.03	<DC1_GUID>	4711
userPrincipalName	JSmith@contoso.com	4711	1	2003-09-10 10:49.03	<DC1_GUID>	4711

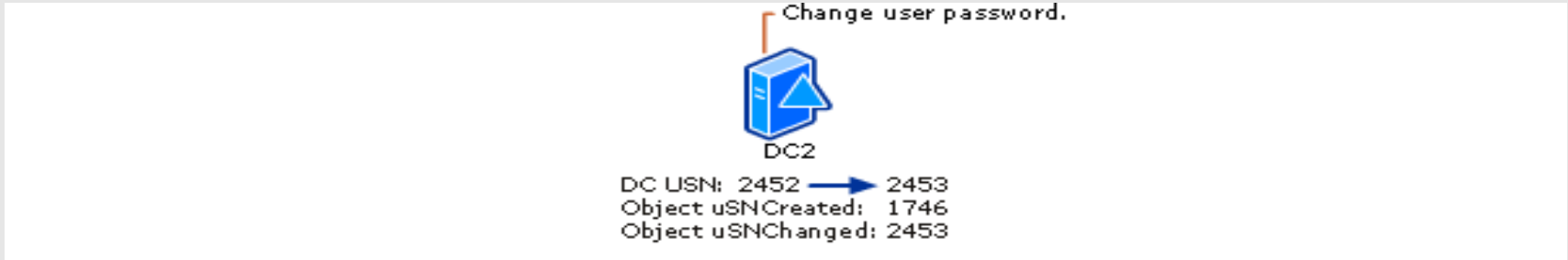
Object replication



Property	Value	Local USN	Version	Originating Time	Originating DC	Originating USN
cn	Jeff Smith	1746	1	2003-09-10 10:49.03	<DC1_GUID>	4711
userPassword	6Be8W5q-	1746	1	2003-09-10 10:49.03	<DC1_GUID>	4711
sAMAccountName	JSmith	1746	1	2003-09-10 10:49.03	<DC1_GUID>	4711
userPrincipalName	JSmith@contoso.com	1746	1	2003-09-10 10:49.03	<DC1_GUID>	4711

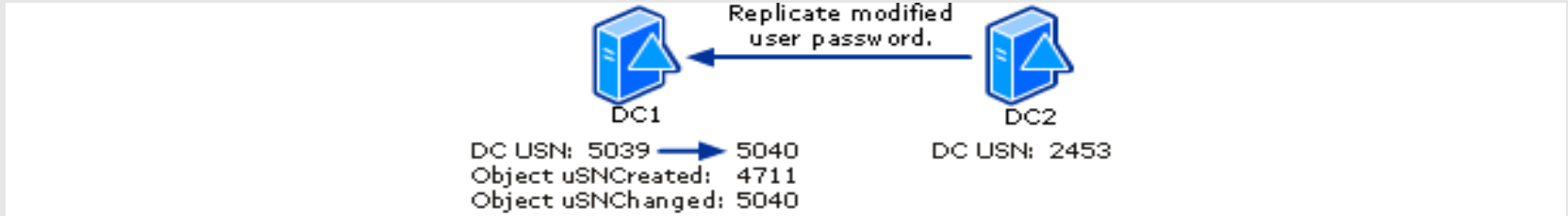
The replication protocol in detail

Object modification



Property	Value	Local USN	Version	Originating Time	Originating DC	Originating USN
cn	Jeff Smith	1746	1	2003-09-10 10:49.03	<DC1_GUID>	4711
userPassword	sEP3569?@2	2453	2	2003-09-10 11:53.29	<DC2_GUID>	2453
sAMAccountName	JSmith	1746	1	2003-09-10 10:49.03	<DC1_GUID>	4711
userPrincipalName	JSmith@contoso.com	1746	1	2003-09-10 10:49.03	<DC1_GUID>	4711

Object replication



Property	Value	Local USN	Version	Originating Time	Originating DC	Originating USN
cn	Jeff Smith	4711	1	2003-09-10 10:49.03	<DC1_GUID>	4711
userPassword	sEP3569?@2	5040	2	2003-09-10 11:53.29	<DC2_GUID>	2453
sAMAccountName	JSmith	4711	1	2003-09-10 10:49.03	<DC1_GUID>	4711
userPrincipalName	JSmith@contoso.com	4711	1	2003-09-10 10:49.03	<DC1_GUID>	4711

Up-to-date Vector and High Watermark

Up-to-date Vector is for tracking originating updates

- Determines attributes to send for replication
- For each domain controller that holds a full replica of the partition. It holds the following:
- Database GUID (invocation ID) of source domain controller
- Highest-originating-USN from source domain controller
- Timestamp from last successful replication (Windows Server 2003)
- If the destination already has an up-to-date value, the source domain controller does not send the update

High Watermark determines objects to consider for replication

- Table on each domain controller that contains entries for direct replication partners and the highest known USN from those partners (uSNChanged)
- Destination provides value to source
- Source filters changes that do not need to be sent

Inter-site replication

To control inter-site replication, you need to create site-links

- You assigned the sites you want to link
- You assign a weight
- You assign an interval and a schedule

Site-links can be transitive

- If you enable the option called Bridge All Site Links

And if you trust your network, you can enable “notifications” on a site-link

- In that case, BHs of the sites included in the site links will replicate like they were in the same site (ignore the inter-site interval and the schedule)

Multimaster, advantages and drawbacks

What if the same object is changed on two DCs at the same time?

Replication conflicts!

- Most conflicts are avoided (attribute level replication), some of them are automatically resolved (name conflicts, orphaned objects) and some of them are avoided thanks **to** F**lexible** S**ingle** M**aster** O**peration** DCs.

FSMO

Note: If you install DNS on a DC, the DNS zone can also be multi-master

- We spoof the **S**tart **O**f **A**uthority

SOA

Let's clarify

Some replication conflicts might be crippling

- Two administrators modifying the schema in two different DCs at the same time!

Schema master, one per forest

- Two administrators picking the same name for a new domain in the forest

Domain naming master, one per forest

- Two DCs creating the same SID for two security principals

RID master, one per domain

- Two GPO modified on two DCs

PDC emulator, one per domain

There is another one, but its importance faded away with Windows Server 2008 R2 FFL

- The infrastructure master… *See the notes section.*

Flexible Single Master Operation, cont.

First come, first served

- The first DC of the forest has all the forest roles
- The first DC of the domain has all the domain roles
- They can be transferred
- They can be seized (if a DC crashed, you can have another DC take over the role)

The (e)PDC is special

- Originally designed to hold the role of **P**rimary **D**omain **C**ontroller when migrating from a Windows NT domain, it ended up with so many other responsibilities:
  - it is the NTP time reference in the forest
  - all password changes are channeled back to it immediately
  - all failed authentications are channeled back to it too
- As a result, it is busier, and might need more physical resources (memory and CPU)

FSMO Summary

Forest Roles (only 1 by Forest)

- Schema Master
  - Manage Schema operations
- Domain Naming Master
  - Manage Domain Naming operations

Domain Roles (only 1 but by Domain)

- RID Master
  - Provide RID Pools
- Infrastructure Master
  - Manage object from other domains operations
- PDC

Where do we store all that data again?

NTDS.dit

- The actual database

EDB.log

- Some transaction logs

EDBxxxxx.log

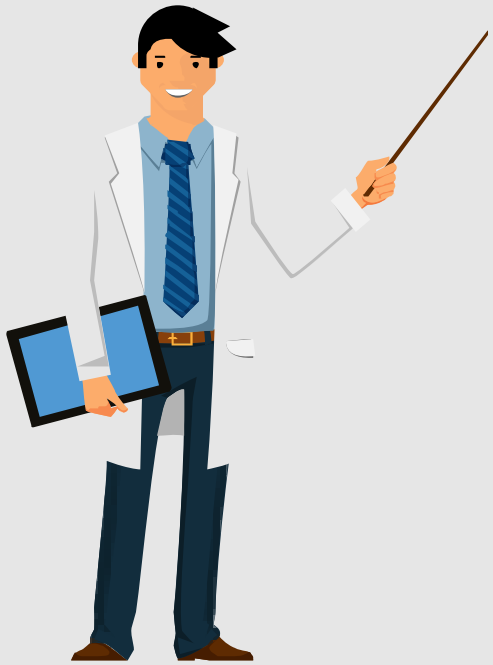
EDB.chk

RES1.log & RES2.log

- Placeholder files to ensure we can write logs and shut down the database gracefully when there is a disk space issue.

Temp.edb

Professor Useful



NTDSUTIL.EXE

- Can be used to transfer or seize FMSO roles
- Can be used to do file operations on the database (such as integrity checks or offline defragmentation…)
- Can be used to set the recovery password (the one you can use to login into a DC in a recovery scenario, so when the database is not up and running)

RootDSE operations

- Many of the operations performed with NTDSUTIL can be done via LDAP using RootDSE modifications
- These operations require a high level of privileges

Once upon a time…

Physical disks

- Back in 2000, disk perf were not what they are now… So there were plenty of ways to optimize disk performance (use different disks for the database and the logs, use different RAID technology…)
- Nowadays, performance when the database and the logs are on the same disk is acceptable.

Back to the future

NTDS snapshots

- You can take Windows snapshots of the NTDS.dit database and mount them on an alternate port to be able to query the previous state of the hosted naming contexts.
- You can create snapshots with NTDSUTIL or directly with the VSS utility

If incorrectly secure → can be use for an offline attack

Chapter

1.1.3

Integration of machines in AD

🎯Describe the Windows features that are enabled by integration with AD

Why join my PC?

Improve security using a corporate computer:

- By establishing a secure channel
- By applying **G**roup **P**olicy **O**bject

GPO

  - Enables policy-based administration
- Allow the use of Kerberos authentication

Enable SSO for AD Account when accessing Data or Apps

Secure Channel

The Netlogon service is responsible to build and maintain a secure connection with its local domain controller

- This connection is called the secure channel
- It enables the local system to talk to a domain controller in a secure way (to resolve SIDs for example)

Domain controllers also have secure channels

- When DCs need to resolve SID from another domain for example, they can build a secure channel with a DC from another domain

Windows Authentication

Authentication is a process for verifying the identity

- For an object, service or person

Windows OS implements a default set of authentication protocols:

- Kerberos, NTLM, Transport Layer Security/Secure Sockets Layer (TLS/SSL), and Digest

LSASS is the process responsible for the Authentication & Signings

Authentication use the Security Support Provider Interface to obtains integrated security services

SSPI

LSASS and its friends

Windows' security brain is LSASS

- Local Security Authority Subsystem Service

LSASS

It stores sensitive information

- It stores keys and other derived secrets
- We need them to ensure SSO

It provides a way to authenticate users

- Authenticates and logs users
- Call authentication packages

This makes Single Sign-On possible

Security Support Providers

Security Support Providers Interface

SSPI

- It is Windows implementation of the Generic Security Service API (GSSAPI)

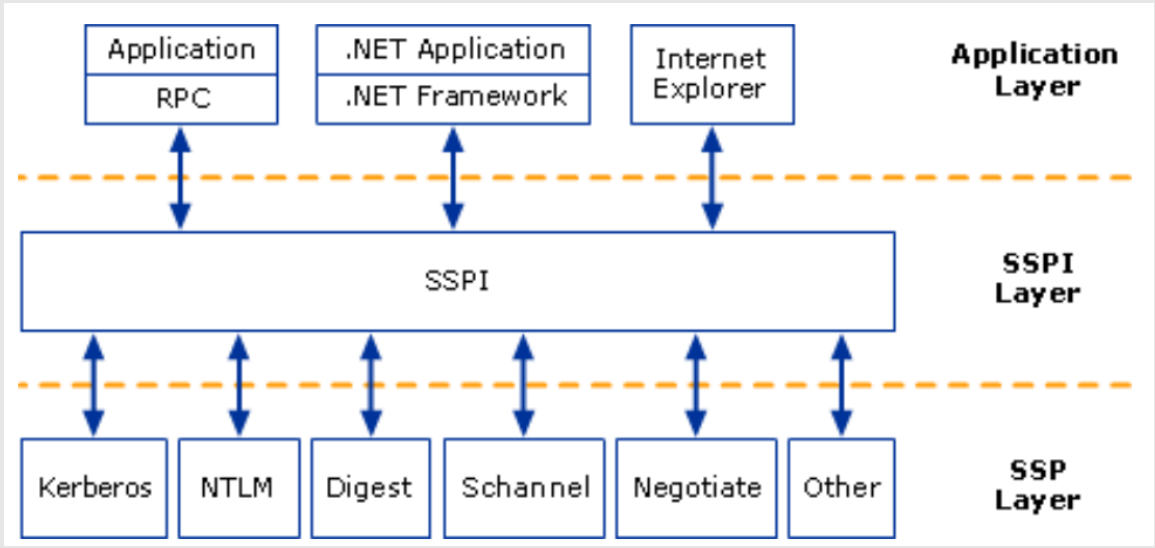
Security Support Providers are protocol implementations

SSP

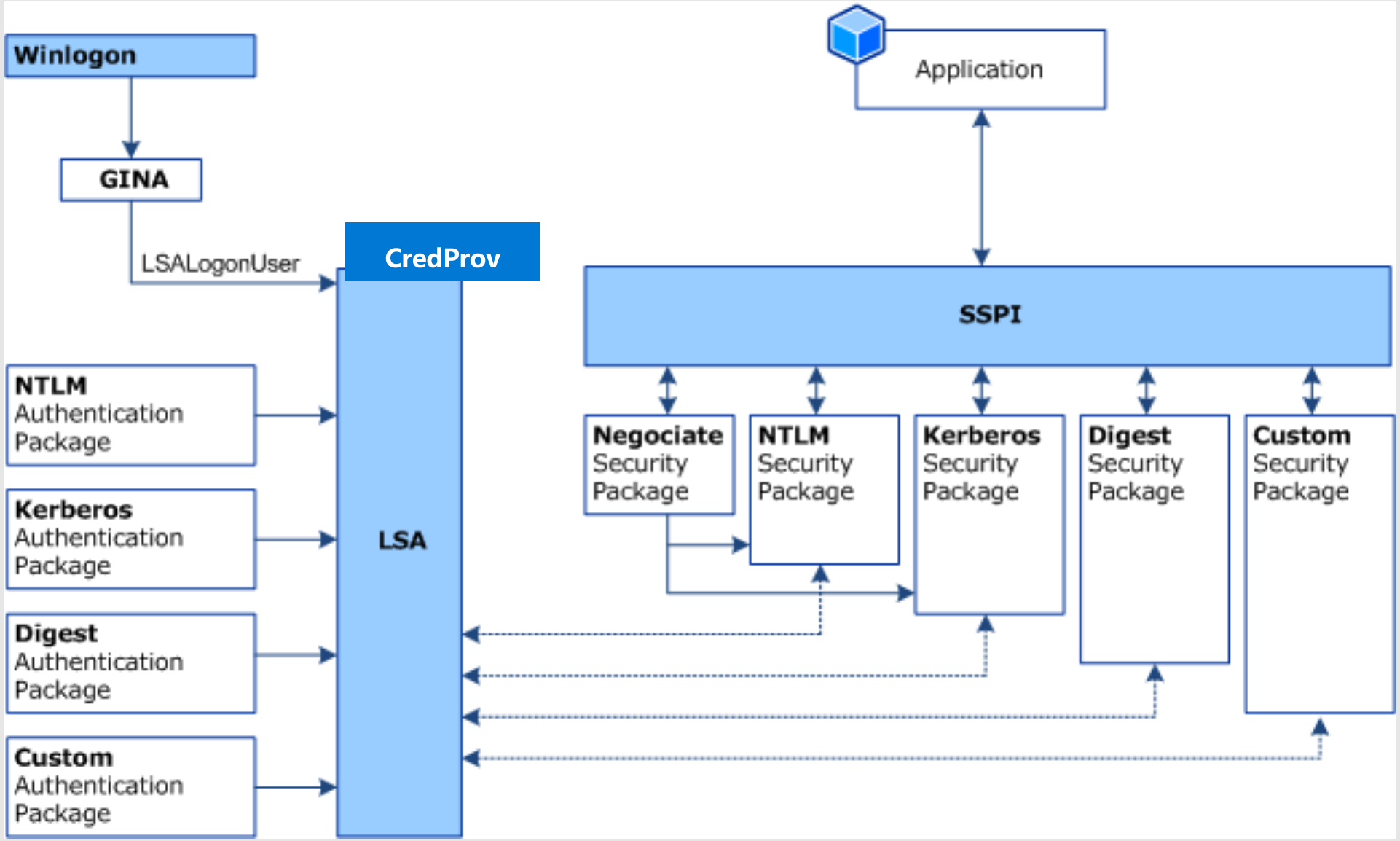
- They have their own DLLs

SPNego is a special

- Negotiate is a SSP that acts as an application layer between the SSPI and the other SSPs
- It actually calls Kerberos first and falls back to NTLM if Kerberos cannot be used



Security Support Provider Selection



Logon Type and User Right Assignments

Not all logons are born equal

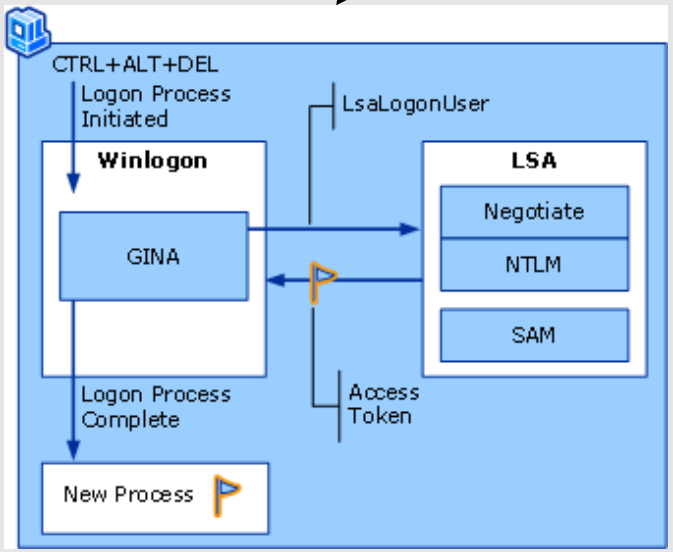
- Depending on the type of logon, credentials will be cached on the system

You can restrict the type of logon using group policies and the user right assignment section

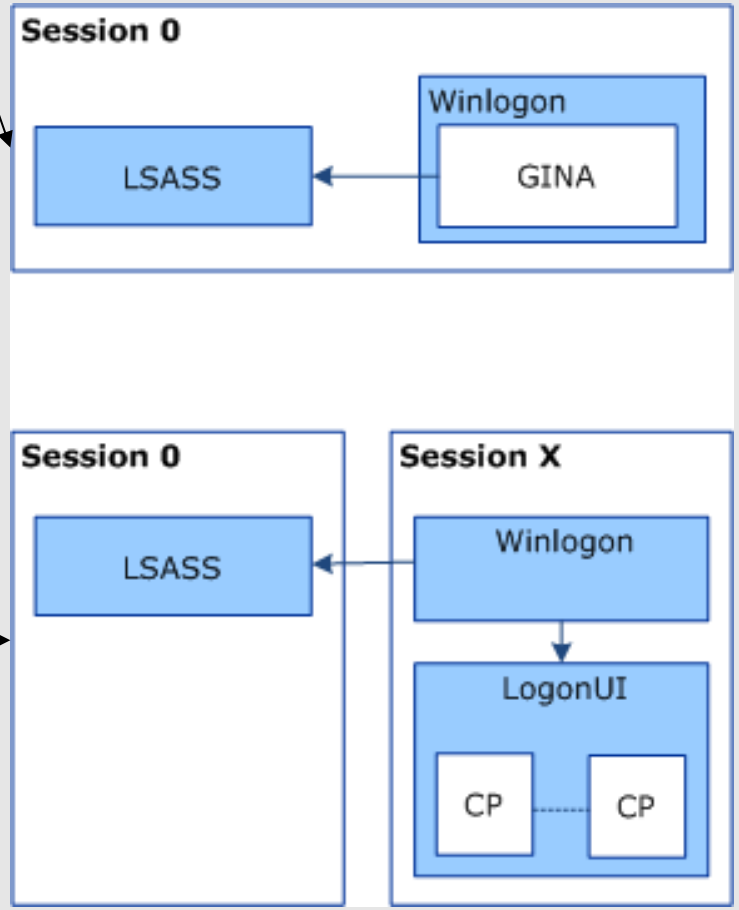
- Access this Computer from the Network
- Log on as a batch job
- Log on locally
- Log on via Terminal Service
- Log on as a service
- Deny Access to this computer from the network
- Deny local logon
- Deny logon as a batch file
- Deny logon as a service
- Deny logon via Terminal Services

Interactive Logon

Windows 2000/2003/XP



Windows 7/8/10/11 …



Logon Time and other artifacts

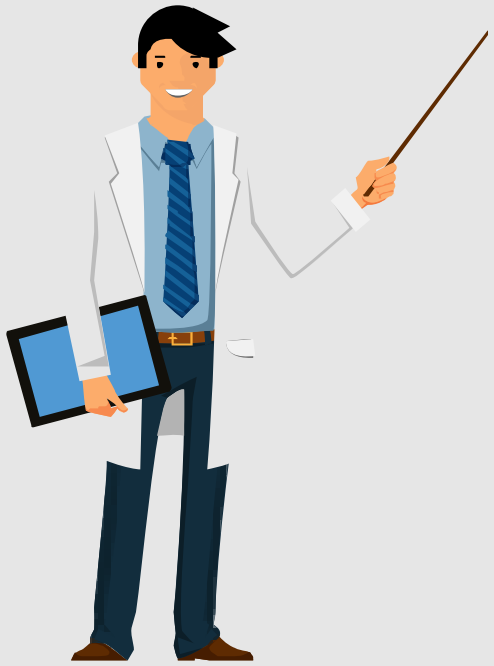
When a user/computer successfully authenticates it updates the lastLogon attribute

- It contains an NT epoch of the time the authentication took place
- It is not replicated, so the value is different depending on the DC you query

It also updates another attribute called lastLogonTimeStamp

- Which is also a NT epoch
- It is replicated!
- But it is updated only if the last time it has changed was more than 14 days ago

Professor Useful



Last Interactive Logon Timestamp

- Feature and attribute available with Windows Server 2008
- It stores for each user:
  - The total number of failed logon attempts at a domain-joined Windows Server 2008 server or a Windows Vista workstation
  - The total number of failed logon attempts after a successful logon to a Windows Server 2008 server or a Windows Vista workstation
  - The time of the last failed logon attempt at a Windows Server 2008 or a Windows Vista workstation
  - The time of the last successful logon attempt at a Windows Server 2008 server or a Windows Vista workstation
- Name is misleading, logon as a batch and logon as a service will also update the timestamp
- Can prevent users from signing in if deployed the wrong way

Group Policy Objects 101

Group Policy Object infrastructure

- Offers a centralized way to control configuration aspects of **machines** and **user**'s environment
  - They have a machine section and a user section.
  - The machine section applies to the machine and the user section applies to the user.
  - Examples: control the Windows security settings, the local group memberships, the states of Windows services, configure Windows components, customize the user's desktop experience, configure user's applications…
- GPOs are applied at boot time and logon time and then reapplied at regular intervals
- GPOs can be linked to sites, domains and Ous

GPOs are uniquely identified by a GUID

- Two GPOs have well-known GUID: **D**efault **D**omain **P**olicy and **D**efault **D**omain **C**ontroller **P**olicy

Group Policy Objects 101

Domain components

- The **G**roup **P**olicy **C**ontainer GPC
  - Is stored in the domain NC and contains information such as the display name
- The **G**roup **P**olicy **T**emplate GPT
  - The actual files containing the settings
- The WMI filter
  - A way to limit the application of a GPO only to systems where a WMI query returns something
- The GPO link
  - Indicates which GPOs are linked to a site, domain or OU
- A security descriptor (set of permissions)
  - GPOs have a DACL
  - An account needs the Read and the Apply permissions to download and apply the settings

Group Policy Objects 101

Client components

- Windows service
- **C**lient **S**ide **E**xtensions

CSE

  - The DLL on the system in charge of applying the settings

Group policy application

Composed of two phases

- Build a list of GPO by listing all gPLink attribute
- Evaluate if the GPO applies (permission Read+Apply and WMI filters evaluation)
- Enumerate all CSE needs to apply the settings
  - Load the corresponding DLLs
- Apply the settings
  - Download the settings and give the data to the CSE to be applied

Refresh takes place every 90 minutes

- Plus a random timer of -30/+30 minutes to avoid flooding DC
- It is actually every 5 minutes on a domain controller
- This can be adjusted

Resultant set of policy can be used to see what happened

RSOP

Why not give the admin password to everyone?

Group Policy Preferences

GPP

- New set of CSE to extend the GPO functionalities
- It's very practical! (you can write registry values, create shortcuts, configure IE…)
- But some of these component allow an administrator to store a password in the GPP file
- Because GPO settings are stored in SYSVOL and that everyone can read SYSVOL, if you use one of these settings, you effectively give your password to everyone. So be careful!

Wait what? Clear-text passwords in SYSVOL?

- To be fair, it is not in the clear. It is encrypted. But the decryption key is public…

System Volume Information

Stores the actual GPO settings

Stores GPO templates

Also replicates, but using a different protocol: DFSR

- Block level replication (optimize network traffic)
- Robust protocol and manage retry/conflict resolutions pretty well

SYSVOL has a Replication Group

- DCs have a subscription to replicate it
- It is configured automatically; nothing must be done for it to work

Once upon a time…

The NETLOGON folder

- To ensure a smooth transition between the NT domain and Windows 2000 Server, the shared folder NETLOGON was kept on domain controllers. It was intended to store logon scripts in the meantime they got into GPO.
- It is fairly common to see that this folder is still used to store a lot of things… Some of those files might also carry some clear-text passwords. Why not?

FRS replication

- Up to Windows Server 2008 R2, the legacy File Replication System was used to replicate SYSVOL
- Well, it has aged… And despite all its flaws, you might still run into it in old environments (if so, please migrate!)

Chapter

1.1.4

Account authentication

🎯 Describe the exchanges required for account authentications in AD

Main AD Authentication Protocols

In an Active Directory environment, Authentication process use 2 main protocols

NTLM

- 2 versions
- A legacy protocol?

Kerberos

- A 3 heads authentication protocol

New Technology Lan Manager

Different versions

- LM, NTLMv1 and NTLMv2
- Versions are enabled as per system configurations (client and server settings)
- NTLMv2 should be the preferred version of the Windows ecosystem

Embedded protocol

Resource based authentication protocol

Challenge/Response protocol

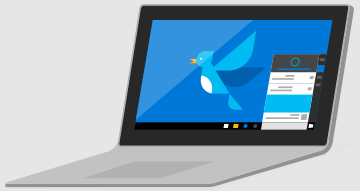
- The resource challenges the user to prove it knowns a shared secret key without sending it

NTLM basic flow (SMB example in a workgroup setting)

User
Alice

Server
File1

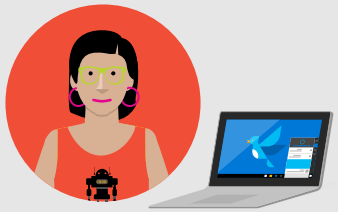
Provides credentials



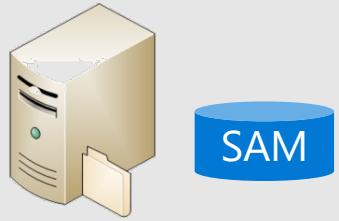
Knows Alice

Stores Alice's password hash

NTLM basic flow (SMB example)



1. SMB stuff… ‘til…they need authentication



2. NTLM_NEGOTIATE

3. NTLM_CHALLENGE

4. NTLM_RESPONSE

5. SMB stuff continues…

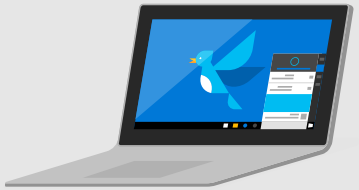
NTLM domain flow
(SMB example in an AD domain joined setting)

User
Alice

Server
File1

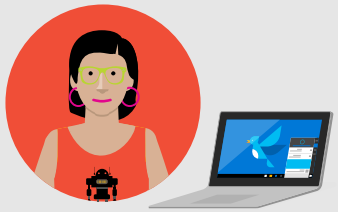
Domain Controller
DC1

Knows
Alice



Stores Alice password's hash

NTLM basic flow (SMB example)



hashed password

challenge

Response

SMB stuff…

NTLM_NEGOTIATE

NTLM_CHALLENGE

NTLM_RESPONSE

Passthrough the DC

User + Challenge + Response

SMB stuff continues…

Once upon a time…

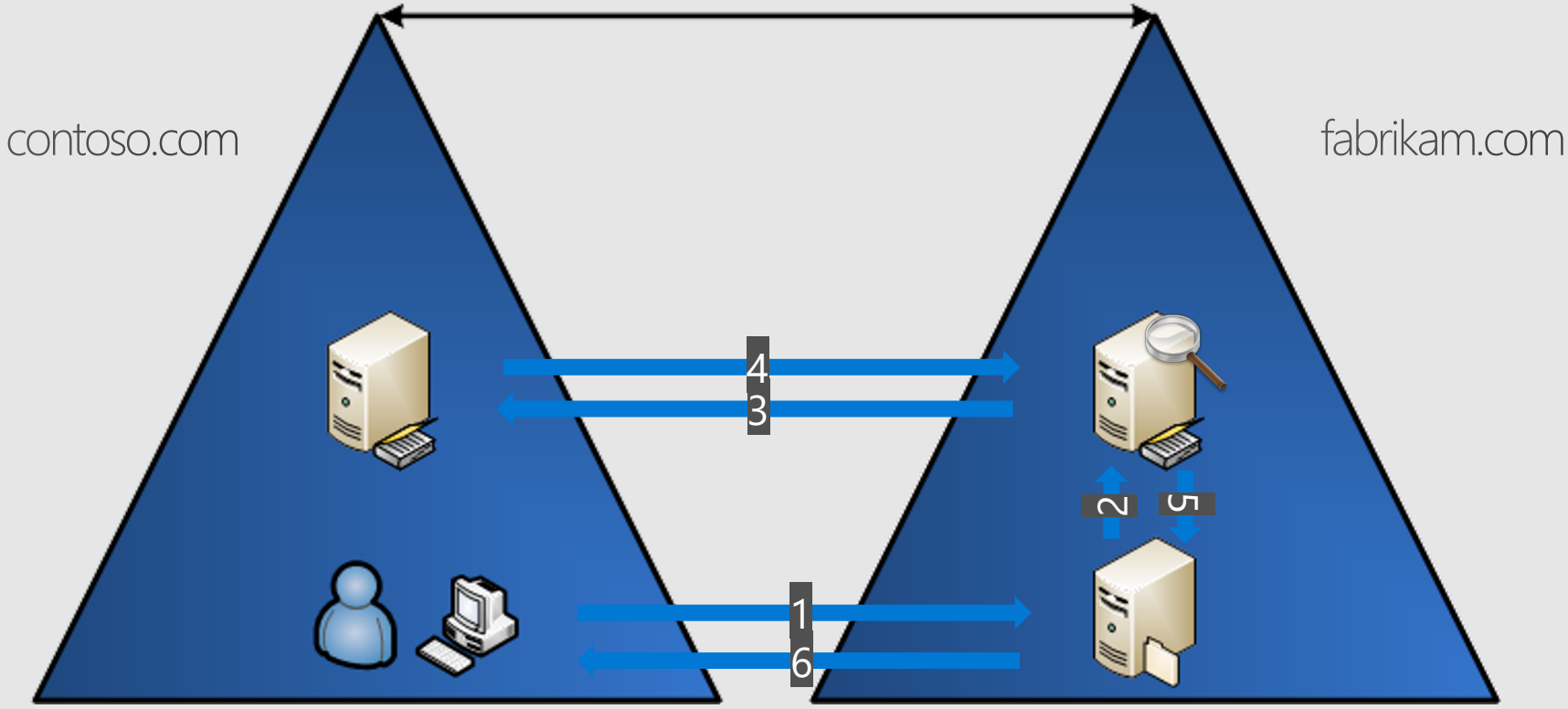
LM and NTLM have been around for ever…

- Historically, LM was there for LAN Manager and OS-2
- But you can still use it now

The LM hash storage is disabled by default since Windows Server 2008

- It is a registry/group policy setting
- If a password has been set before disabling its storage, the LM hash is still present in the NTDS.dit database

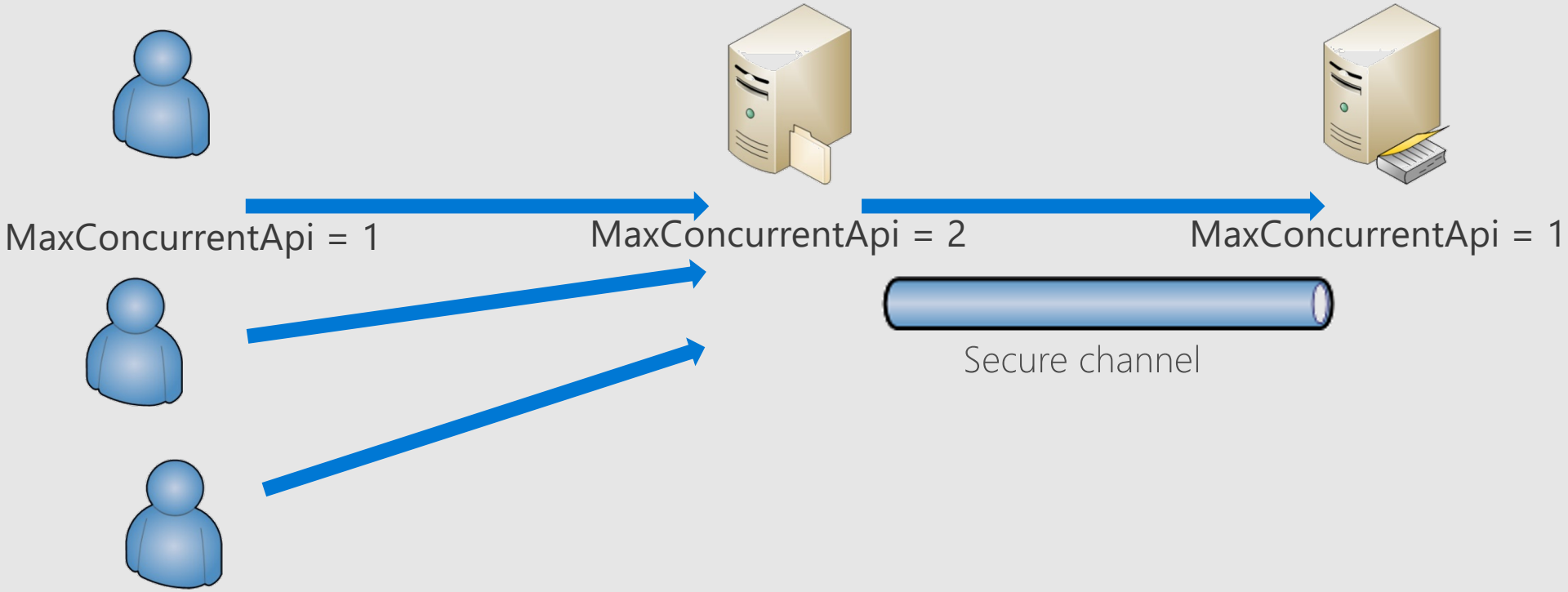
NTLM over trust



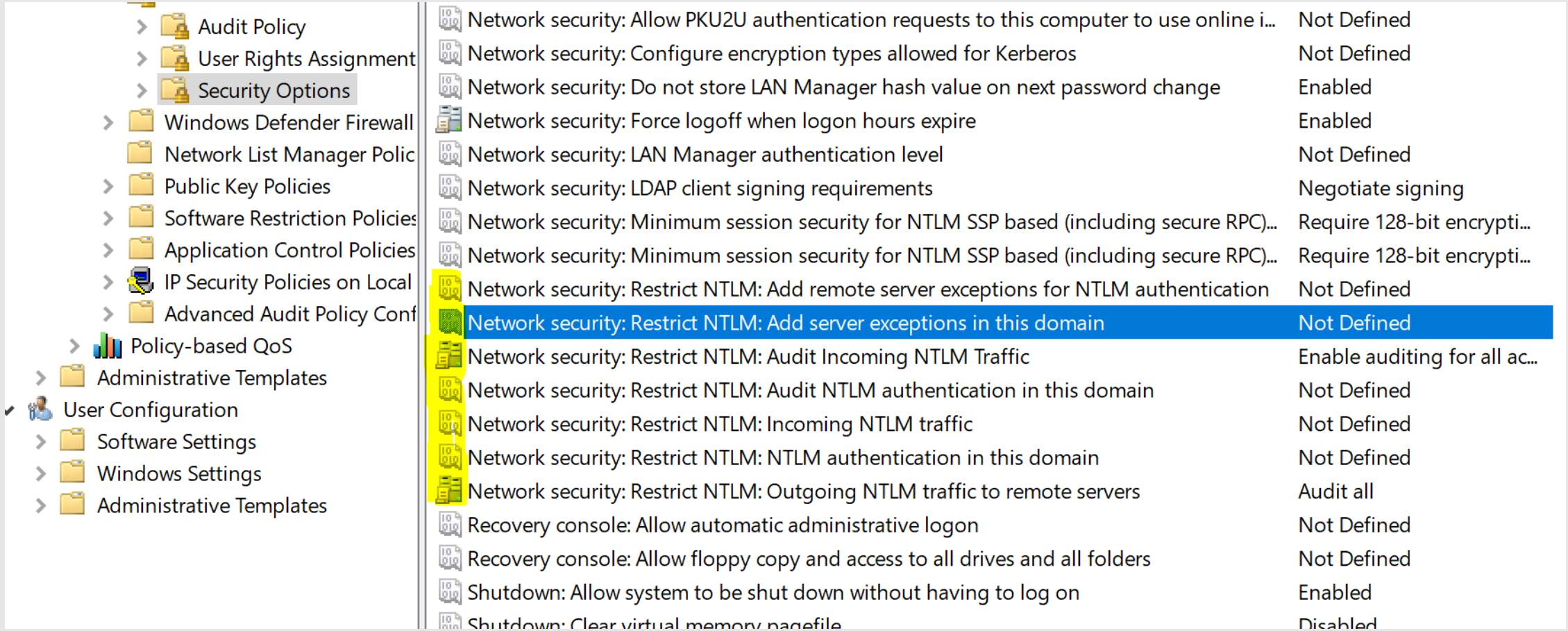
Performance issues? A DDOS?

NTLM passthrough leverages the secure channel

- How many operations can the secure channel handle at a time?



NTLM Blocker



Kerberos

Not a Microsoft thing

- RFC 4120 for the version 5
- Interoperate with Unix/Linux/Java implementation

~~Resource~~ Network oriented protocol

Ticket based

- Valid for 10 hours

3 sub protocols

- Authentication Service Exchange
- **T**icket-**G**ranting **S**ervice Exchange TGS
- Client/Server Exchange (embedded protocol)

Kerberos in a nutshell

A user: Bob
user**P**rincipal**N**ame: bob@contoso.com

UPN

A service: file service on File1
service**P**rincipal**N**ame: CIFS/file1.contoso.com

SPN

A **K**ey **D**istribution **C**enter
For a specific realm: contoso.com

KDC

userPrincipalName

A user: Bob
userPrincipalName: bob@contoso.com

UPN

Looks like an email

- The second part is called the UPN suffix, by default it is the DNS name of the domain
- Additional and custom UPN suffixes can be added to the forest

Must be unique in the forest

If an account does not have a UPN, an implicit UPN is assumed

- Which is the sAMAccountName of the user + the default UPN suffix

servicePrincipalName

A service: file service on File1
servicePrincipalName: CIFS/file1.contoso.com

SPN

Looks like SERVICECLASS/HOST|FQDN:PORT@REALM

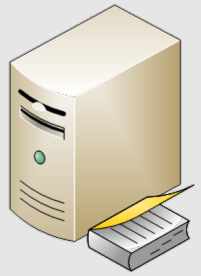
- CIFS/file1, HTTP/web.contoso.com, MSSQLSvc/sql1.contoso.com:1433

Must be unique in the forest

If a requested SPN does not exist, the DC tries to find one falling back to the HOST service

- If CIFS/file1 is not found, but HOST/file1 exists, the latter will be used for the ticket's encryption

KDC



A Key Distribution Center
For a specific realm: contoso.com

KDC

It is a Windows service running on all DCs

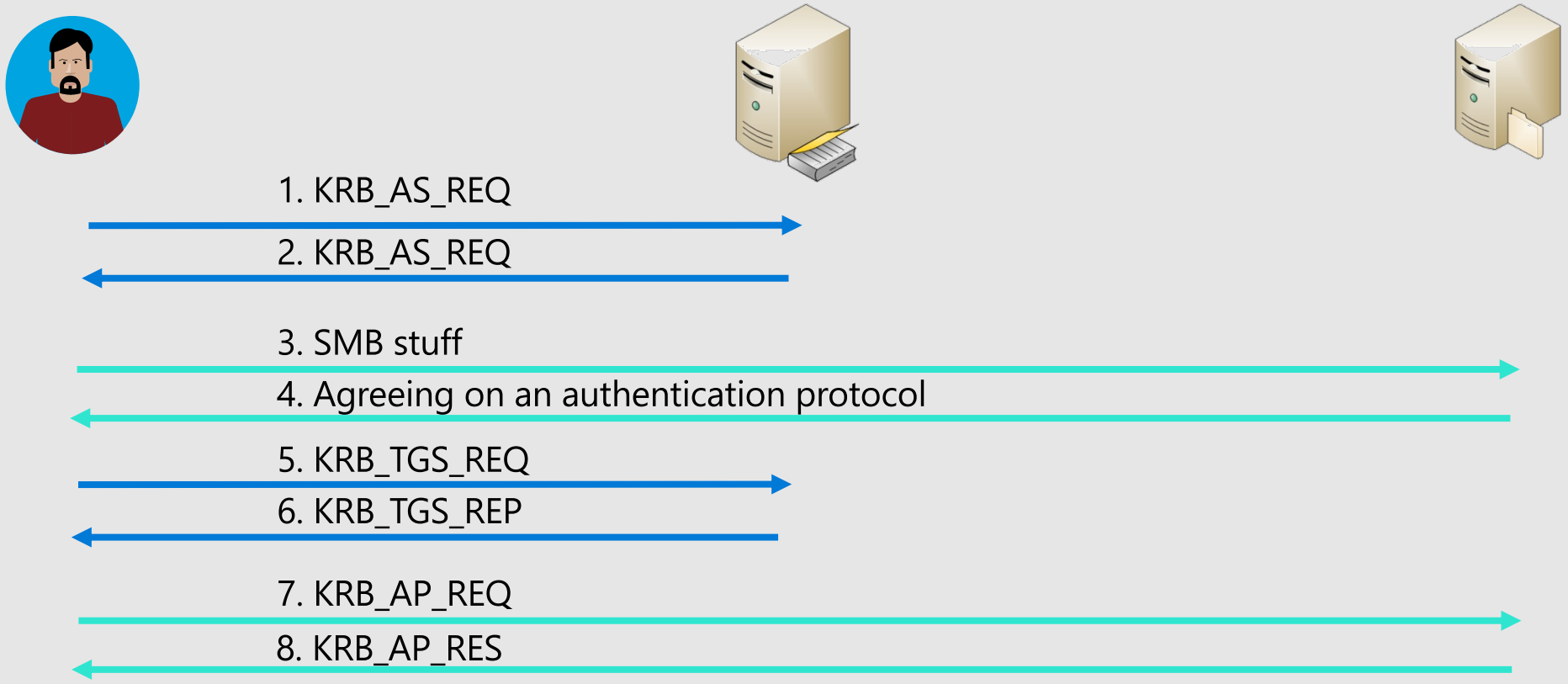
The KDC secret is the KrbTgt account's long term key

It is using the tickets

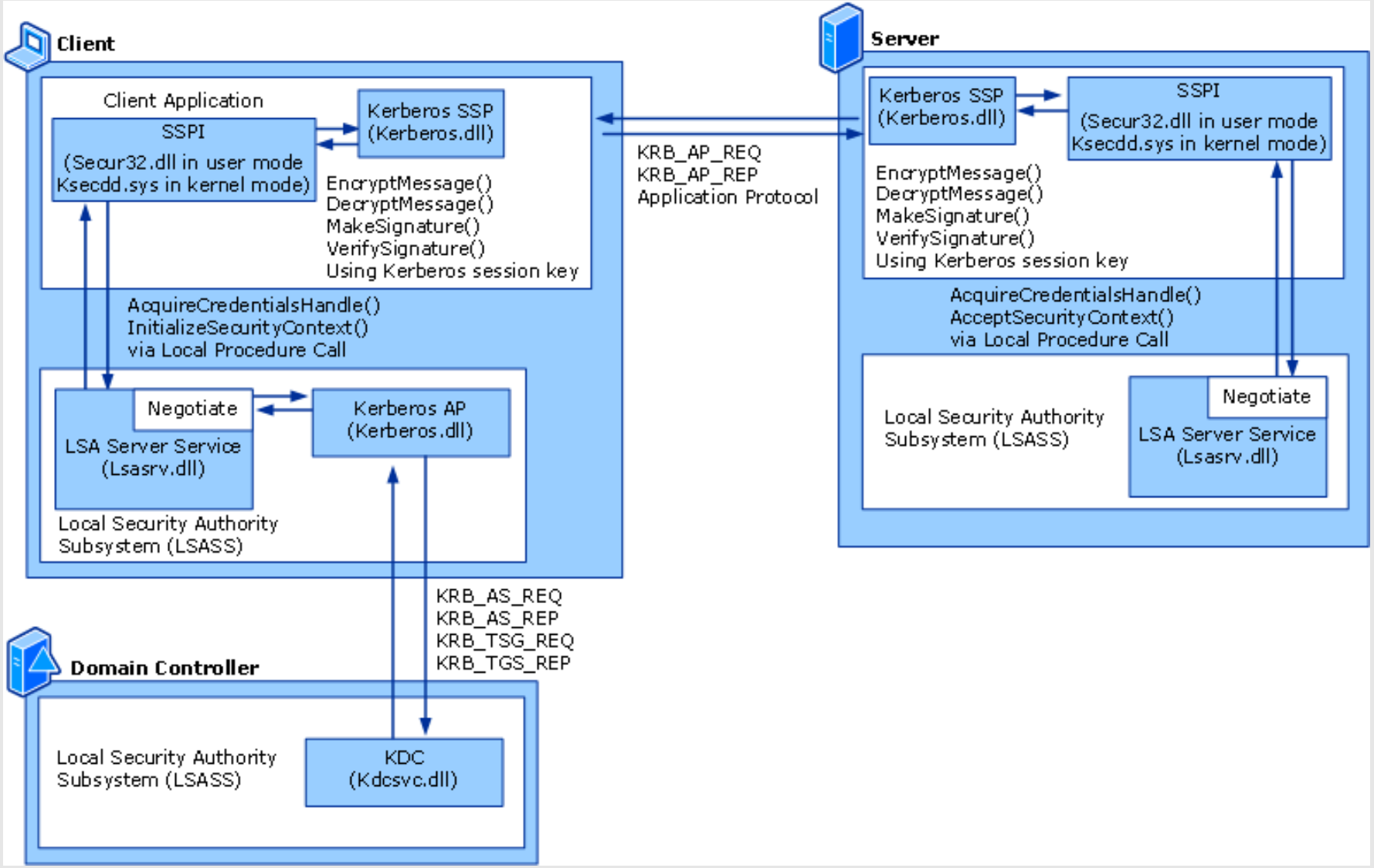
It has two components

- Authentication Service (to issue TGTs)
- Ticket Granting Service (to issue service tickets)

Kerberos in a nutshell



Kerberos in a nutshell (nerd edition)



Kerberos on the wire

Many parts of the messages are in the clear

- Of course not the session keys and participants' secrets!

Error messages are in the clear

- Great for troubleshooting!

UDP/TCP port 88

- Windows 2000/XP/2003 use UDP first and if it is fragmented, then switch to TCP
- Windows Vista/2008 and higher are using TCP right away

KRB_AS_REQ

1. KRB_AS_REQ

Bob's secret

KDC-REQ:

cname = bob

realm = contoso.com

sname = krbtgt/contoso.com

eType = ENCTYPE

nonce = 854491316

address = BOBLAPTOP

KDC options = KDC Flags

till = timestamp

PA-DATA:

PA-ENC-TIMESTAMP

PA-PAC-REQUEST

KDC secret

Bob's secret

KRB_AS_REP

2. KRB_AS_REQ

Bob's secret

KDC-REP:

cname = bob

crealm = contoso.com

TGT:

realm = contoso.com

sname = krbtgt/contoso.com

kvno = 3

etype = ETYPE

Key = Session Bob/KDC

PAC = SIDs...

EncKDCRepPart:

kvno = 4

Key = Session Bob/KDC

nonce = 854491316

sreal = contoso.com

sname = krbtgt/contoso.com

caddr = BOBLAPTOP

starttime = timestamp

endtime = timestamp

renew-till = timestamp

flags = Ticket Flags

KDC secret

Bob's secret

KRB_TGS_REQ

3. KRB_TGS_REQ

Bob's secret

Bob/KDC

TGT Bob/KDC

TGS-REQ:

realm = contoso.com

sname = cifs/file1.contoso.com

eType = ENCTYPE

nonce = 855037033

KDC options = KDC Flags

till = timestamp

PA-DATA:

PA-TGS-REQUEST:

TGT

Authenticator:

eType = ENCTYPE

cname = bob

crealm = contoso.com

cksum = checksum

ctime = timestamp

seq-number = 855037033

KDC secret

Bob's secret

File1 secret

KRB_TGS_REP

3. KRB_TGS_REP

Bob's secret

Bob/KDC

TGT Bob/KDC

TGS-REP:

cname = bob

crealm = contoso.com

Service Ticket:

sname = CIFS/file1.contoso.com

realm = contoso.com

kvno = 2

etype = ETYPE

key = Session Bob/File1

PAC = SIDs...

EncTGSRepPart:

kvno = 4

Key = Session Bob/File1

nonce = 854491316

sreal = contoso.com

sname = CIFS/file1.contoso.com

starttime = timestamp

endtime = timestamp

renew-till = timestamp

flags = Ticket Flags

KDC secret

Bob's secret

File1 secret

KRB_AP_REQ

5. KRB_AP_REQ

Bob's secret

Bob/KDC

TGT Bob/KDC

Bob/File1

Ticket Bob/File1

AP-REQ:

ap-options = APOPTIONS

Ticket

Authenticator:

eType = ENCTYPE

cname = bob

crealm = contoso.com

cksum = checksum

ctime = timestamp

KDC secret

Bob's secret

File1 secret

KRB_AP_REP

6. KRB_AP_REP

AP-REP:

EncAPRepPart:

ctime = timestamp

Bob's secret

Bob/KDC

TGT Bob/KDC

Bob/File1

Ticket Bob/File1

KDC secret

Bob's secret

File1 secret

KDC and Ticket Flags

Tells the DC what options to enable on the TGT

- Can I use delegation?
- Is this a delegated ticket already?
- Is it ok to renew it?

The KDC will try to accommodate

- And reply with BAD_OPTIONS if the options are not compliant with the Kerberos domain policy.

Kerberos Policies

- **Defined at the domain level**

  By default, in the Default Domain Policy

Policy setting	Default value
Maximum lifetime for service ticket	10 hours
Maximum lifetime for user ticket	10 hours
Maximum lifetime for user ticket renewal	7 days
Maximum tolerance for computer clock synchronization	5 minutes
Enforce user logon restrictions	Enabled

Encryption negotiation

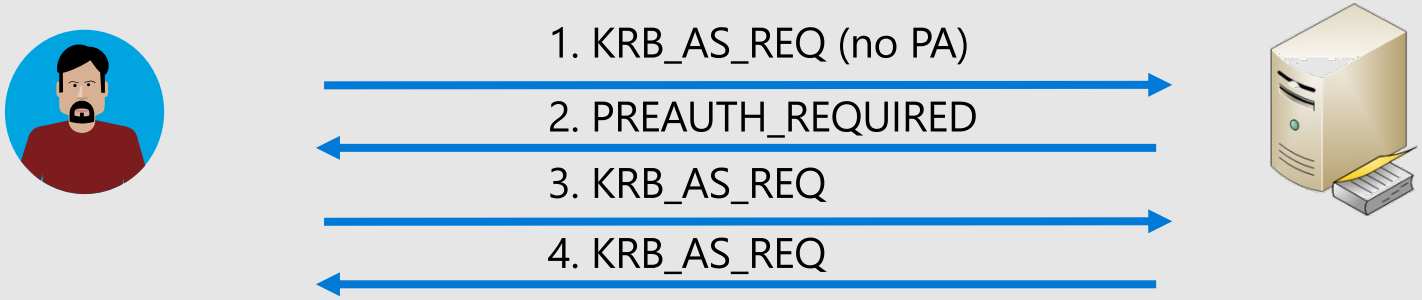
The KDC can accommodate different encryption types

ETYPE	Name	Default
1	DES-CBC-CRC	Disabled since Windows Server 2008
3	DES-CBC-MD5	Disabled since Windows Server 2008
23	RC4-HMAC-NT	Enabled
17	AES128-SHA1	Enabled (DLF 2008)
18	AES256-SHA1	Enabled (DLF 2008)

Encryption probing

To determine what the ETYPE to use is, Windows clients are probing the KDC

- They send a KRB_AS_REQ without the PA-ENC-TIMESTAMP
- The KDC responds with a KDC_ERROR_PREAUTH_REQUIRED which contains a PA-ENCTYPE-INFO2 block containing the eType to use for the request.
- The clients can now request a ticket with the indicated eType



Kerberos long term secret generation

The user's secret is also known as its long term key

- The keys vary depending on the eType
- DES and RC4 keys are not salted
- AES keys are salted

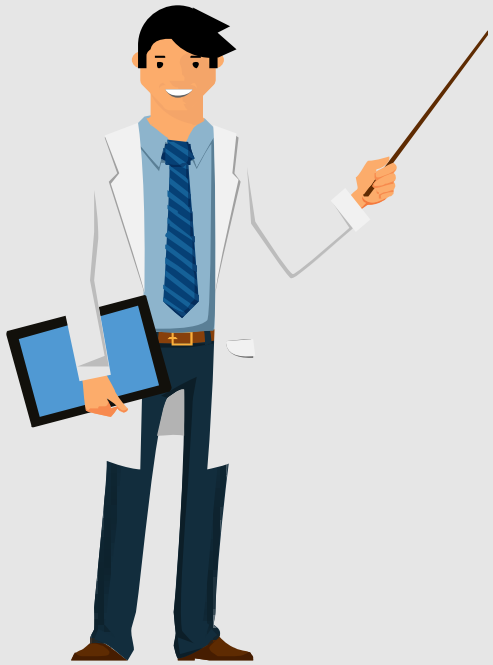
Keys have version numbers

- kvno
- It is the version number of the password attribute in AD

Key are stored in the AD object too

- The RC4 key is the unicodePwd (same as the NTHash)
- The AES keys are stored in the attribute SupplementalCredentials

Professor Useful



NTHash is not salted

- Two users with the same password will have the same key
- True with Kerberos and RC4 too

AES keys are salted

- Two users with the same password will not have the same key

What about iteration count?

- AES keys are using a PBKDF2 function called with 4096 iterations

Privilege Attribute Certificate

PAC

The PAC contains the authorization information

- SIDs
- As well as other data about the user (such as profile path, home directory…)

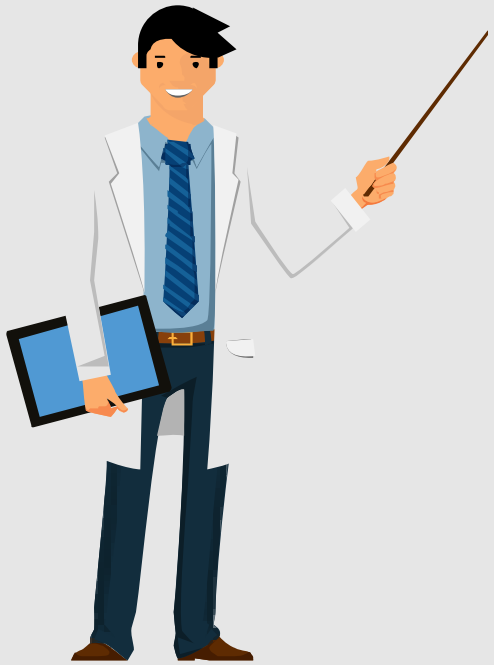
Windows allocates a 48Kb buffer to read the PAC

Windows compresses the SIDs in the PAC

- SIDs' domain parts are redundant

If the PAC does not fit, then the ticket handling fails

Professor Useful



Inside the PAC
(an extract of what's in it)

```
_KERB_VALIDATION_INFO {
        LogonTime;
        LogoffTime;
        PasswordLastSet;
        PasswordCanChange;
        PasswordMustChange;
        EffectiveName;
        FullName;
        LogonScript;
        ProfilePath;
        HomeDirectory;
        HomeDirectoryDrive;
        LogonCount;
        BadPasswordCount;
        UserId;
        PrimaryGroupId;
        GroupCount;
        GroupIds;
        LogonServer;
        LogonDomainName;
        SidCount;
        ResourceGroupDomainSid;
        ResourceGroupCount;
        ResourceGroupIds;
}
```

Once upon a time…

The MaxTokenSize limit used to be way smaller

- 8Kb in Windows 2000
- 12Kb in Windows XP/2003 > Windows 8.1/Windows 2012R2
- 48Kb since

Although the name is about tokens, the actual registry value is about the allocated size of the buffer

PAC validation

PAC integrity is not always checked against a domain controller

If the consumer of the ticket is not running in the context of local system, network service, or local service then the machine contact its closest DC for PAC validation

- This uses the secure channel of the machine

Unless the service's identity has the SeTCBprivilege privilege

Or if the registry value ValidateKdcPacSignature is set to 1

PAC validation dark side

KERB_VERIFY_PAC_REQUEST and its KB friends

- Microsoft Security Bulletin MS11-013 – Important
  An elevation of privilege vulnerability exists in implementations of Kerberos. The vulnerability exists because the Microsoft Kerberos implementation supports a weak hashing mechanism, which can allow for certain aspects of a Kerberos service ticket to be forged. A malicious user or attacker who successfully exploited this vulnerability could obtain a token with elevated privileges on the affected system.
- Microsoft Security Bulletin MS14-068 – Critical
  A remote elevation of privilege vulnerability exists in implementations of Kerberos KDC in Microsoft Windows. The vulnerability exists when the Microsoft Kerberos KDC implementations fail to properly validate signatures, which can allow for certain aspects of a Kerberos service ticket to be forged. Microsoft received information about this vulnerability through coordinated vulnerability disclosure. When this security bulletin was issued, Microsoft was aware of limited, targeted attacks that attempt to exploit this vulnerability.

PKINIT based authentication

Uses a certificate to authenticate with the KDC

- The KRB_AS_REQ will not be encrypted with the derived user's long term key but with the private key of the user

Requires that the KDC has a valid certificate

Both user and KDC certificate have to come from a Certification Authority trusted by the forest

- It means that the certificate of the CA has to be stored in a particular location in configuration NC
- Certificate revocation needs to be possible too

KRB_AS_REQ with PA_PK_AS_REQ

Trusted CA in NTAuth

Bob's secret

Bob Public Key

Bob Private Key

1. KRB_AS_REQ

KDC-REQ:

```
cname = bob
realm = contoso.com
sname = krbtgt/contoso.com
eType = ENCTYPE
nonce = 854491316
address = BOBLAPTOP
KDC options = KDC Flags
till = timestamp
```

PA-DATA:

PA_PK_AS_REQ

Bob Public Key

KDC secret

Bob's secret

KDC Public Key

KDC Private Key

KRB_AS_REP with PA_PK_AS_REP

Trusted CA in NTAuth

Bob's secret

Bob Public Key

Bob Private Key

2. KRB_AS_REQ

KDC-REP:

cname = bob

crealm = contoso.com

TGT:

realm = contoso.com

sname = krbtgt/contoso.com

kvno = 3

etype = ETYPE

Key = Session Bob/KDC

PAC = SIDs...

+ NT Hash

EncKDCRepPart:

kvno = 4

Key = Session Bob/KDC

nonce = 854491316

sreal = contoso.com

sname = krbtgt/contoso.com

caddr = BOBLAPTOP

starttime = timestamp

endtime = timestamp

renew-till = timestamp

flags = Ticket Flags

KDC secret

Bob's secret

KDC Public Key

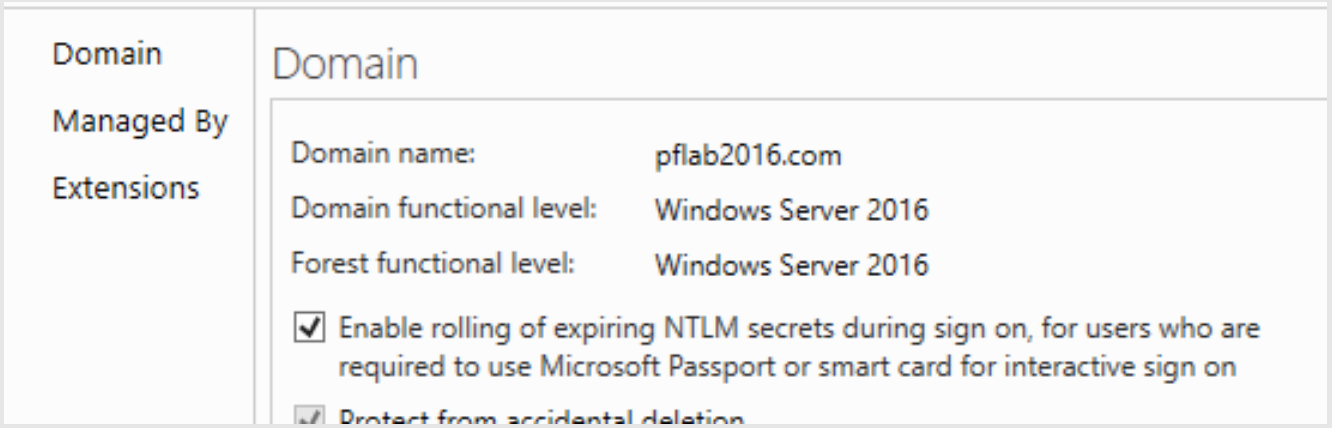
KDC Private Key

Certificate based authentication

Note that in this situation, the PAC will contain an NTLM_SUPPLEMENTAL_CREDENTIAL structure containing the NTHash of the user

If a user requires a Smartcard for logon, its long term secret never changes

- Windows Server 2016 domain controllers introduced an automatic rollover of the user's secret



Kerberos over trust

You need a TGT to ask for a service ticket

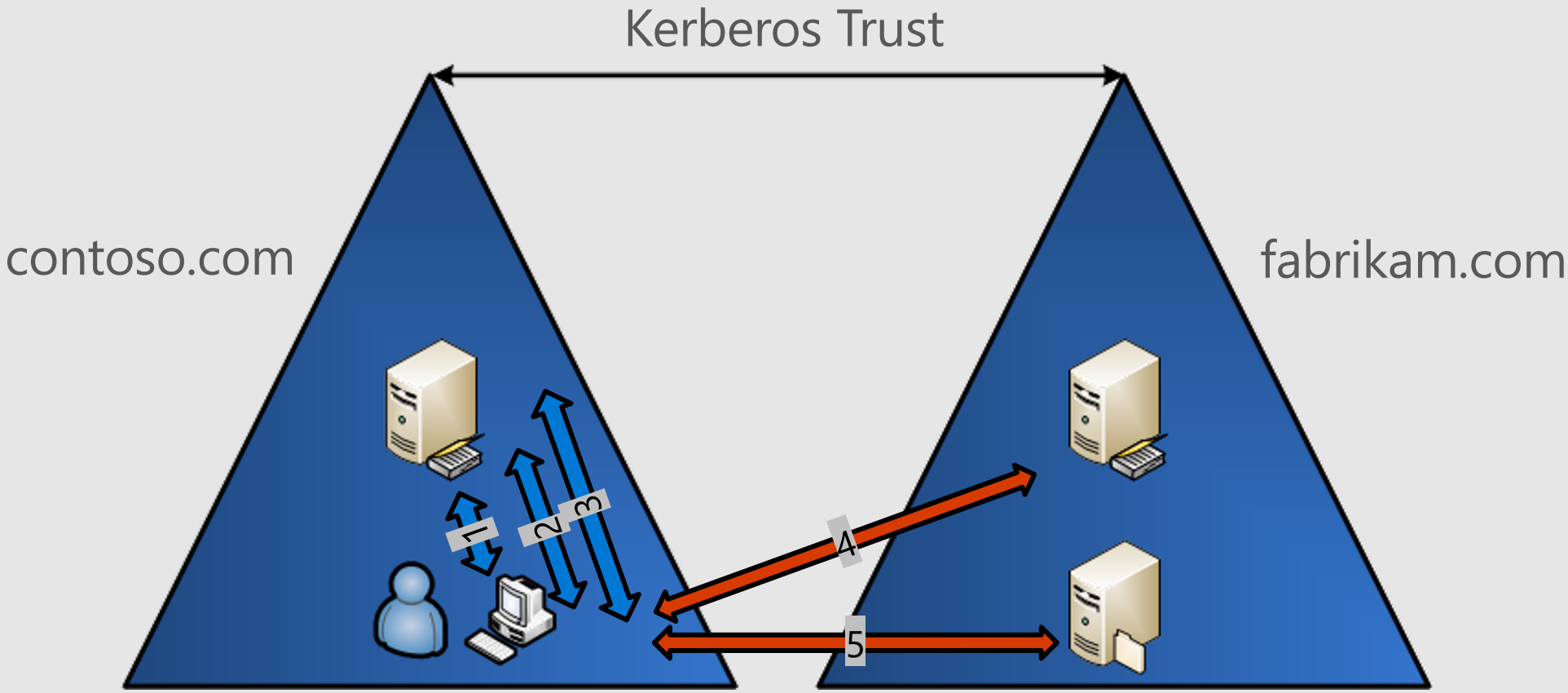
If the service is in another domain (and trusted) your local KDC will give you a TGS referral

- It's like a TGT but for another domain
- It is using the derived trust's secret for encryption
- You can use it to request a service ticket in that domain

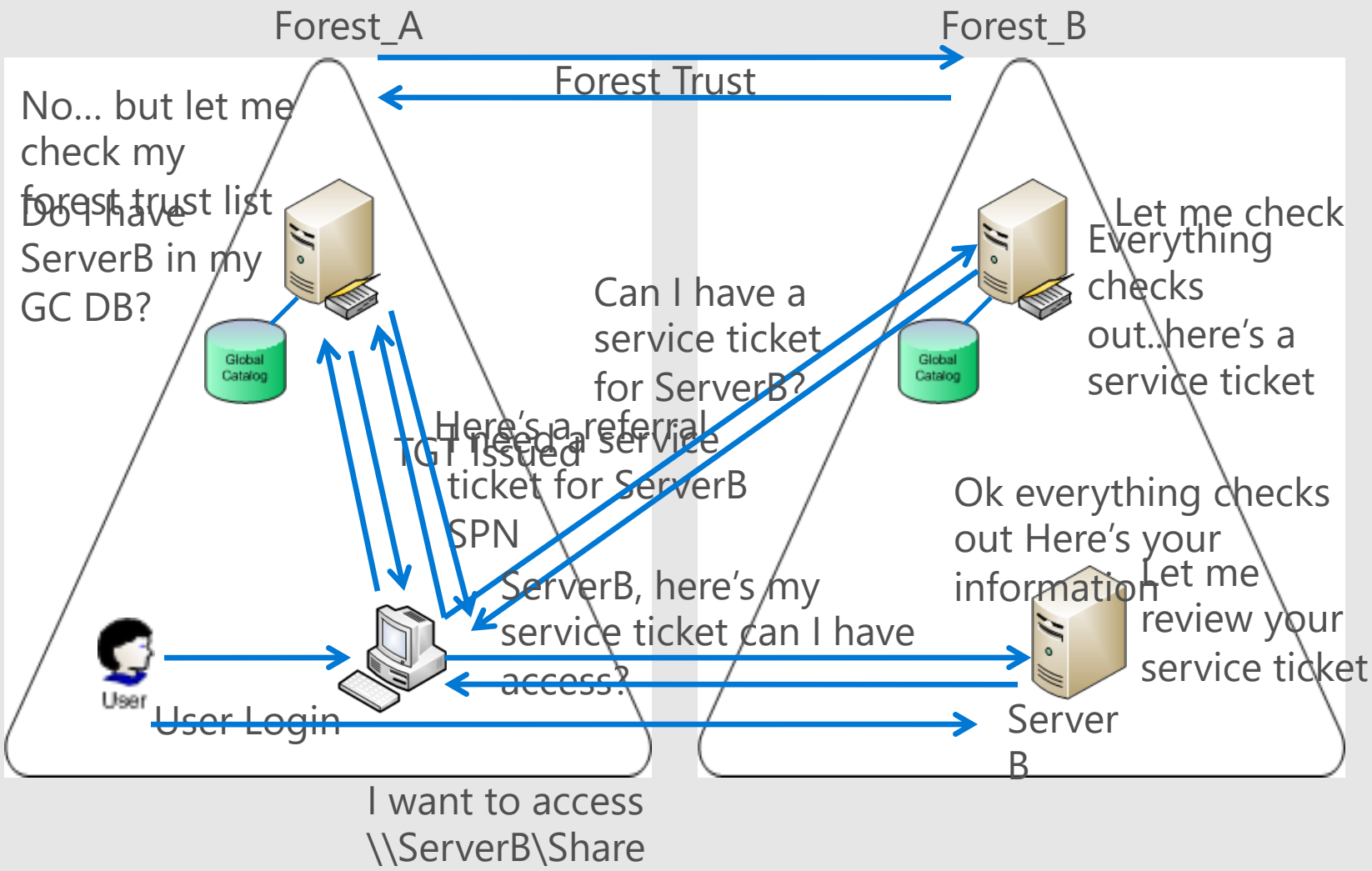
Kerberos might not work over external trusts

- You can configure Forest Search Order to make it happen (either on the clients or on the KDCs)

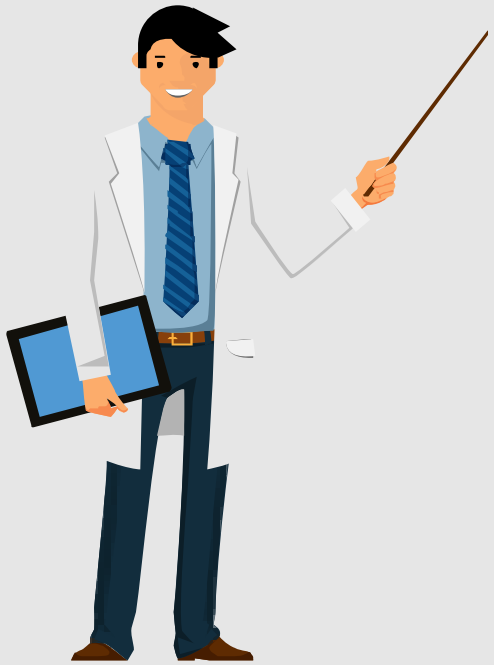
Kerberos over trust – Simple process



Gaining Access Across a Forest Trust - Deep



Professor Useful



Kerberos over external trusts

- Many Kerberos authentication failures are due to the inability to locate the target service in another forest
- Forest Search Order can be used for clients or for KDCs
- It is set via GPO

Delegation

Delegation (or Kerberos forwarding) offers the ability for a service to impersonate a user against another service

- Best suited for n-tiers applications where the front-end application can request a ticket to access the back-end services of the application on behalf of the user
- Without prompting the user
- Maintaining the context of the identity of the user end-to-end

Only domain administrators can enable delegation on accounts

- Privilege: SeEnableDelegationPrivilege

Delegation Capabilities

Windows Server 2000 introduced un-constrained delegation

Windows Server 2003 introduced constrained delegation and protocol transition with Kerberos extensions S4U and S4U2Proxy

- Each middle-tier calling principal and back-end service must be in the same domain.
- Domain administrators are responsible for configuration/setup.
- A resource owner has no control over which middle-tier service may delegate to it.

Windows Server 2012 introduced constrained delegation at the target resource

- Control over which middle-tier services may delegate caller identities to the target resource.
- Target service owner control without domain admin requirements.
- The middle tier can exist in different trusted realms from a resource tier.

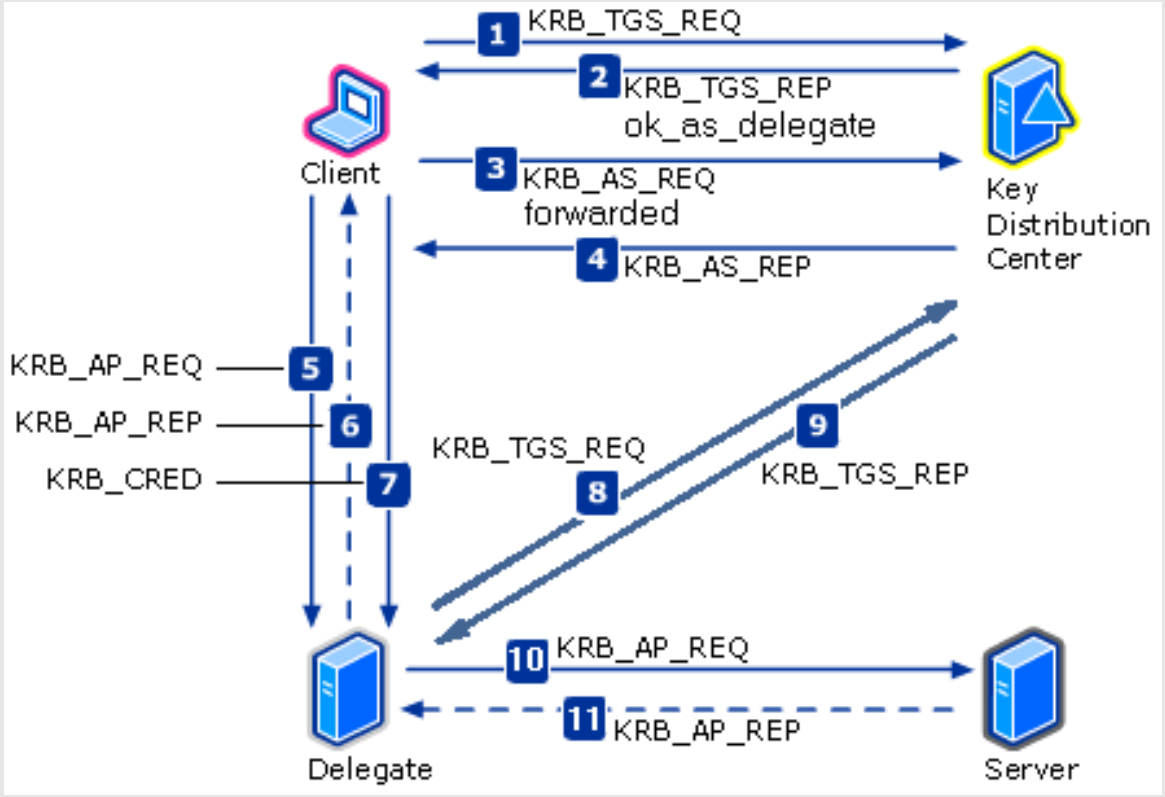
Unconstraint delegation

The user sends its TGT

The FrontEnd service uses it to request a service ticket

The FE can request ANY service ticket

If the FE is compromised, we can access any resources the user has access to



Constraint delegation

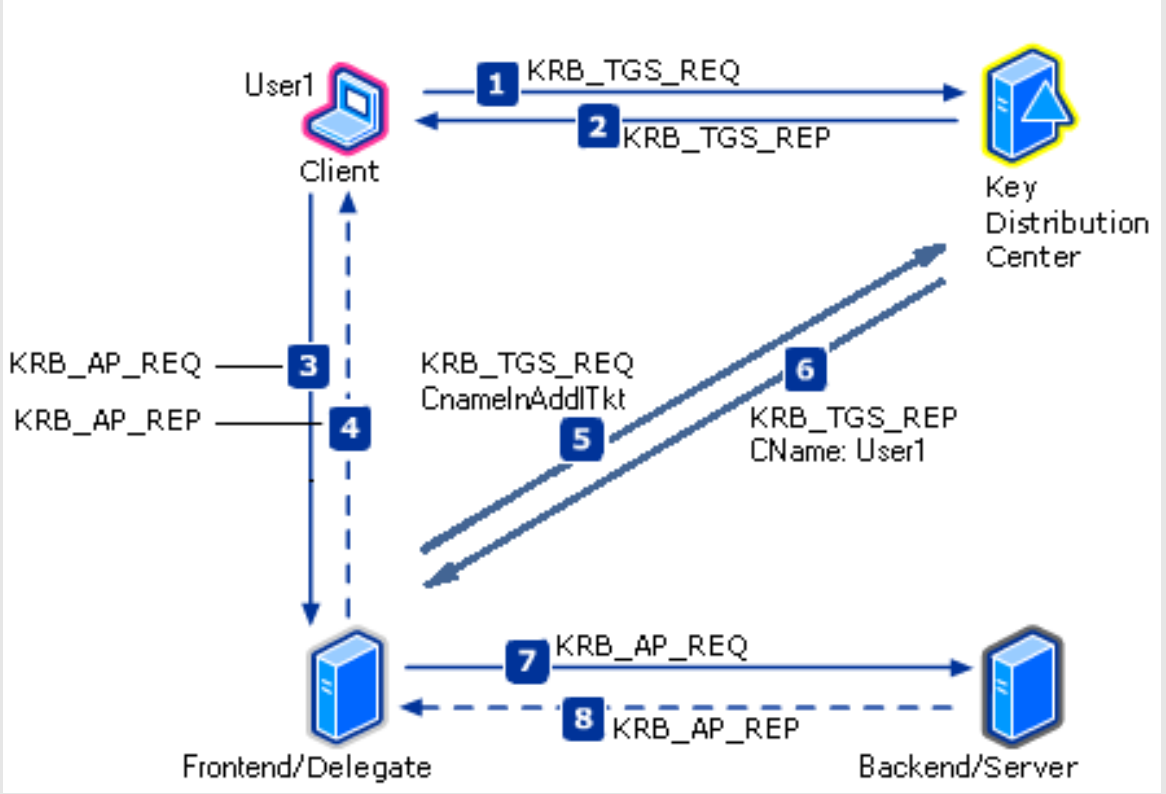
S4U2Proxy

The user sends it TGT

The FrontEnd service uses it to request a service ticket

The FE can request only service tickets for defined SPNs

If the FE is compromised, we can access only to resources the user has access to using the specified SPN (attribute: msDS-AllowedToDelegateTo)



Resource-based constraint delegation

Give the backend service the power

- The backend service decides from where it is accepting forwarded tickets
- Instead of letting the frontend decide where it is allowed to access

New attribute: msDS-AllowedToActOnBehalfOfOtherIdentity

Requirements

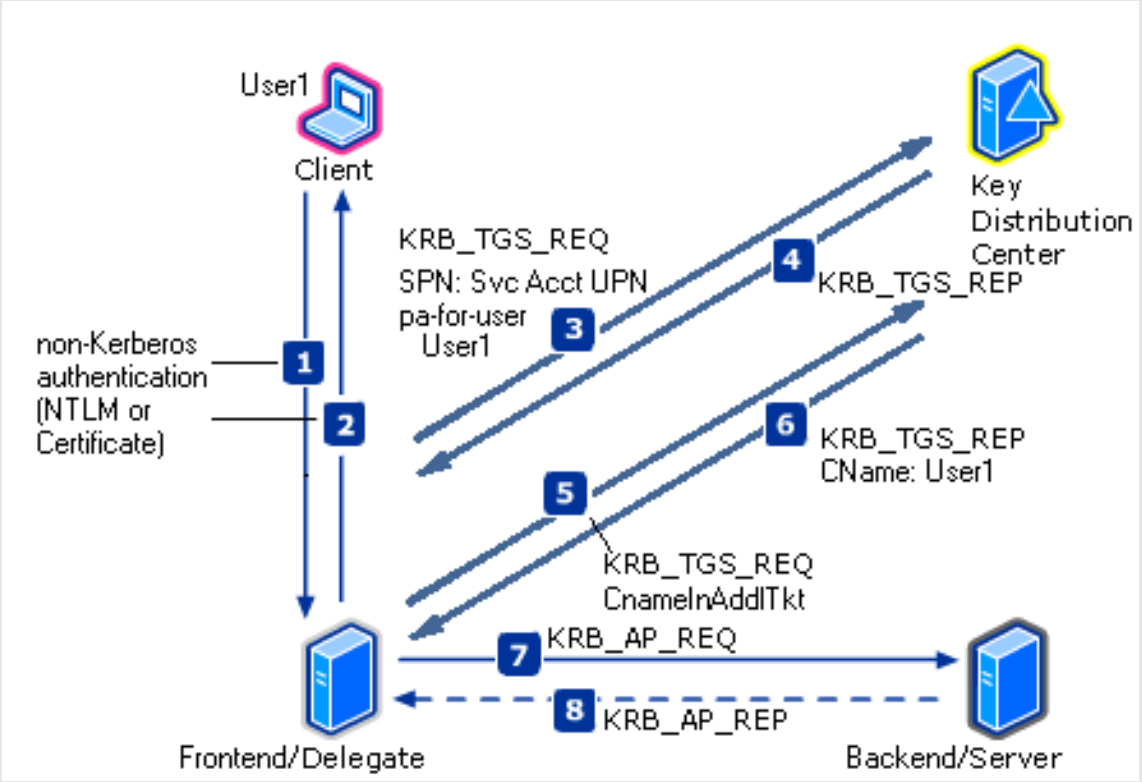
- Windows Server 2012 KDC in the Front-end account domain
- Windows Server 2012 KDC in the Back-end resource domain
- Windows Server 2012 running on the Front-end server

Constraint delegation with protocol transition

S4U2Proxy + S4U2Self

What if the user did not use Kerberos to start with?

- Well then we can transition it to Kerberos silently…
- It requires the frontend service to have the right delegation flag on its accounts and the appropriate level of privilege on the local system



The master of the keys

The hash of the KrbTgt account is the key of all TGTs

The KrbTgt's long term key never changes

- It does only once when changing the DFL to 2008 or higher
- Can be manually reset

We will see in the next module what the associated risks and attacks against Kerberos are

Claims

It is possible to add arbitrary claims in the PAC

- You can select attributes on the user or machine that will be added into the PAC
- This can be used only for file permissions on Windows 8/Windows Server 2012

This is not enabled by default

- This has to be supported by the KDC (Windows Server 2012 or higher)
- This has to be enabled on the client (Windows 8 or higher)
- Can leverage Kerberos protocol transition to allow legacy clients to access files on a server for which the DACLs are using claims

Kerberos armoring

Flexible Authentication Secure Tunneling

- The computer's TGT is used to protect the AS and TGS user's exchanges with the KDC

It requires at least Windows 2012 KDC and Windows 8 clients

- If you enforce FAST in the domain and have workstations and servers lower than Windows 8/Windows 2012, they will not be able to login anymore

keytab

Unix/Linux/Java applications don’t have LSASS to store the long term key

- So you can generate a file containing the key
- You can use the KTPASS utility to generate keytabs
- Treat keytabs like you would treat password, they contains clear text encryption keys!

What time is it?

Because time is very sensitive in a Kerberos realm, we need to be on-time

- The Windows Time service (w32tm) maintains the domain joined systems' local clocks synced with the domain controllers' clocks
- Each DC is a NTP server
- The PDC of the root domain is the time reference for the forest

If there are more than 5 minutes time difference between systems, we cannot do Kerberos!

Windows Cached Credentials

Working from home? You can still sign-in in Windows without talking to a domain controller

- This is thanks to the cached credentials in Windows
- HKEY_LOCAL_MACHINE\SECURITY\Cache
- It is salted… With a username…
- It is encrypted…
- Can be overwritten!

By default, the last 10 users are cached

- Can be changed…

Chapter

1.1.5

The domain controller discovery mechanism

🎯 Explain how systems can locate a domain controller

Hey dude, where's my DC?

This is where the love story between AD and DNS unravels

- Windows clients will use DNS to find the closest domain controller
- It is a Windows service responsible for this: netlogon

The high availability is implemented at the client level. Not at the network level!

- For non-Windows clients, the logic has to be implemented either at the OS or at the application level
- Hardcoded systems (to IPs or names) will be impacted by DC downtime. Windows clients will just find another one automatically thanks to their netlogon service

May the best DC win!

When a DC starts, its netlogon service registers DNS records

When a domain-joined machine is looking for a DC, its netlogon service queries DNS

- From those DNS responses, the client will select the first X and perform what is often referred to as an “LDAP ping” with specific flags (like I am looking for a reliable time source, or for a Windows Server 2012 R2 DC)
- The first one who answers with the right flag is selected
- The result is cached for 12 hours (unless the DC becomes unresponsive)

SRV records and closest DC discovery

A DC belongs to a site

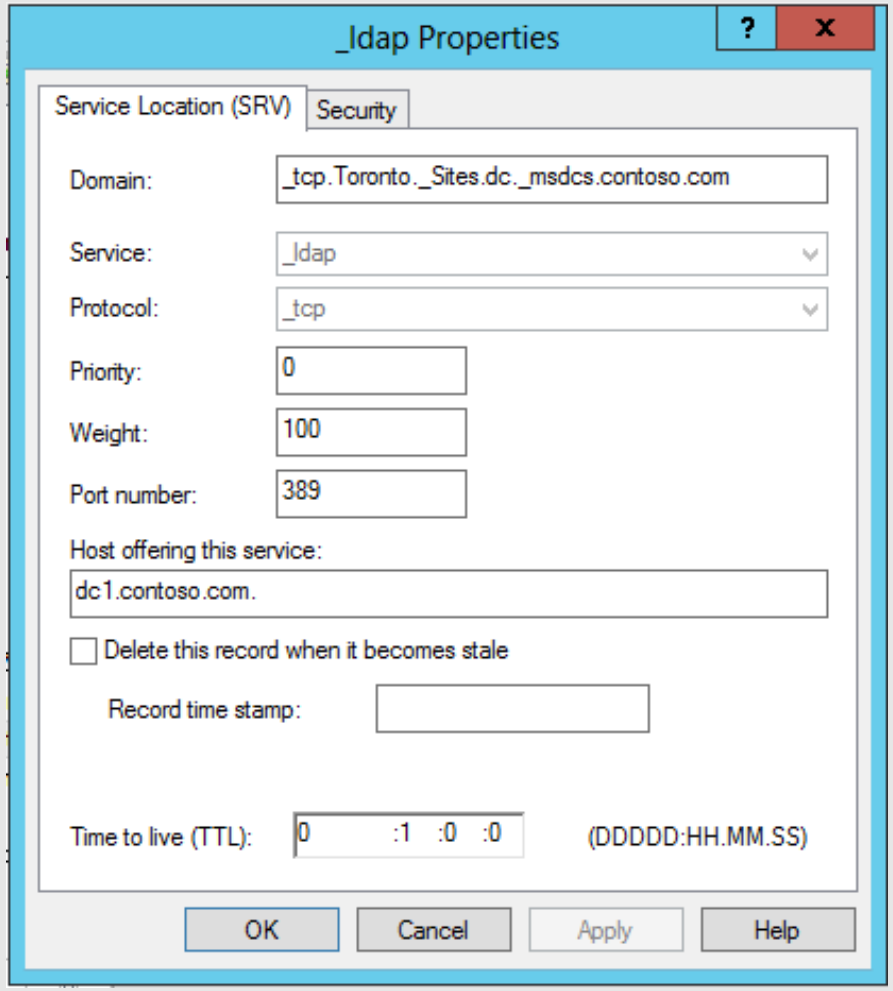
- The operator can choose which site or rely on subnet/site association in any

Clients also belong to a site

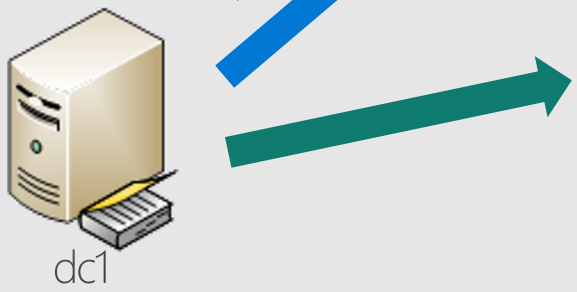
- Dynamically thanks to their subnet/site association
- Can also be overwritten

Everything takes place through DNS queries

- What does an SRV record look like?
  - It has a service, a protocol, **a priority, a weight** and a port
  - Those two help a client to pick the best DC



DsGetDcName in a nutshell phase 1



Site specific records

_ldap._tcp.Toronto._Sites.dc._msdcs.contoso.com. = dc1

_kerberos._tcp.Toronto._Sites.dc._msdcs.contoso.com. = dc1

…

Siteless or generic records

_ldap._tcp.dc._msdcs.contoso.com. = dc1

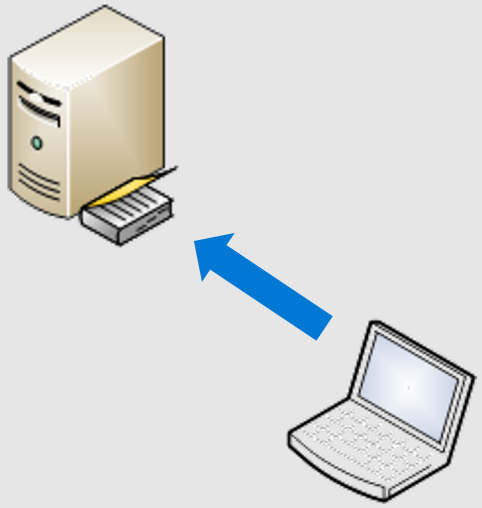
_kerberos._tcp.dc._msdcs.contoso.com. = dc1

…

NETLOGON.EXE

SRV records
netlogon.dns

DsGetDcName in a nutshell phase 2



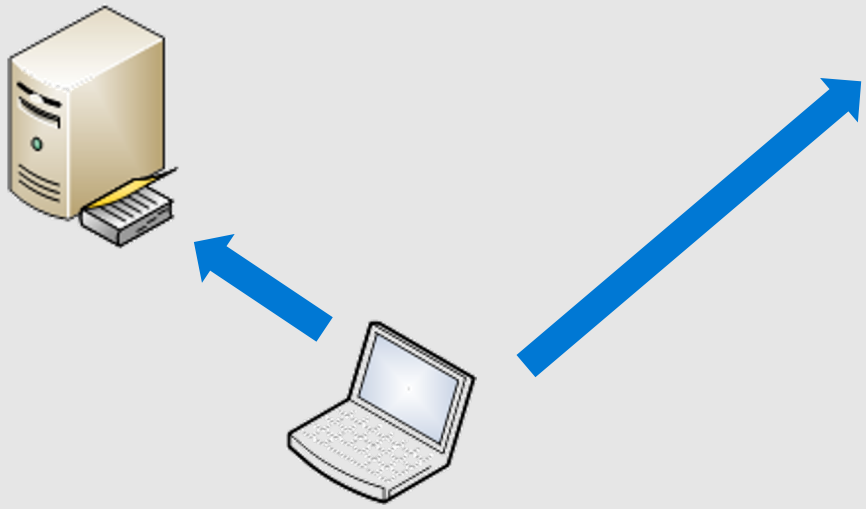
win10

IP = 1.0.0.1
Site =?

Site specific records
_ldap._tcp.Toronto._Sites.dc._msdcs.contoso.com. = dc1
_kerberos._tcp.Toronto._Sites.dc._msdcs.contoso.com. = dc1
…

Siteless or generic records
_ldap._tcp.dc._msdcs.contoso.com. = dc1
_kerberos._tcp.dc._msdcs.contoso.com. = dc1
…

DsGetDcName in a nutshell phase 3



win10

IP = 1.0.0.1
Site = Toronto

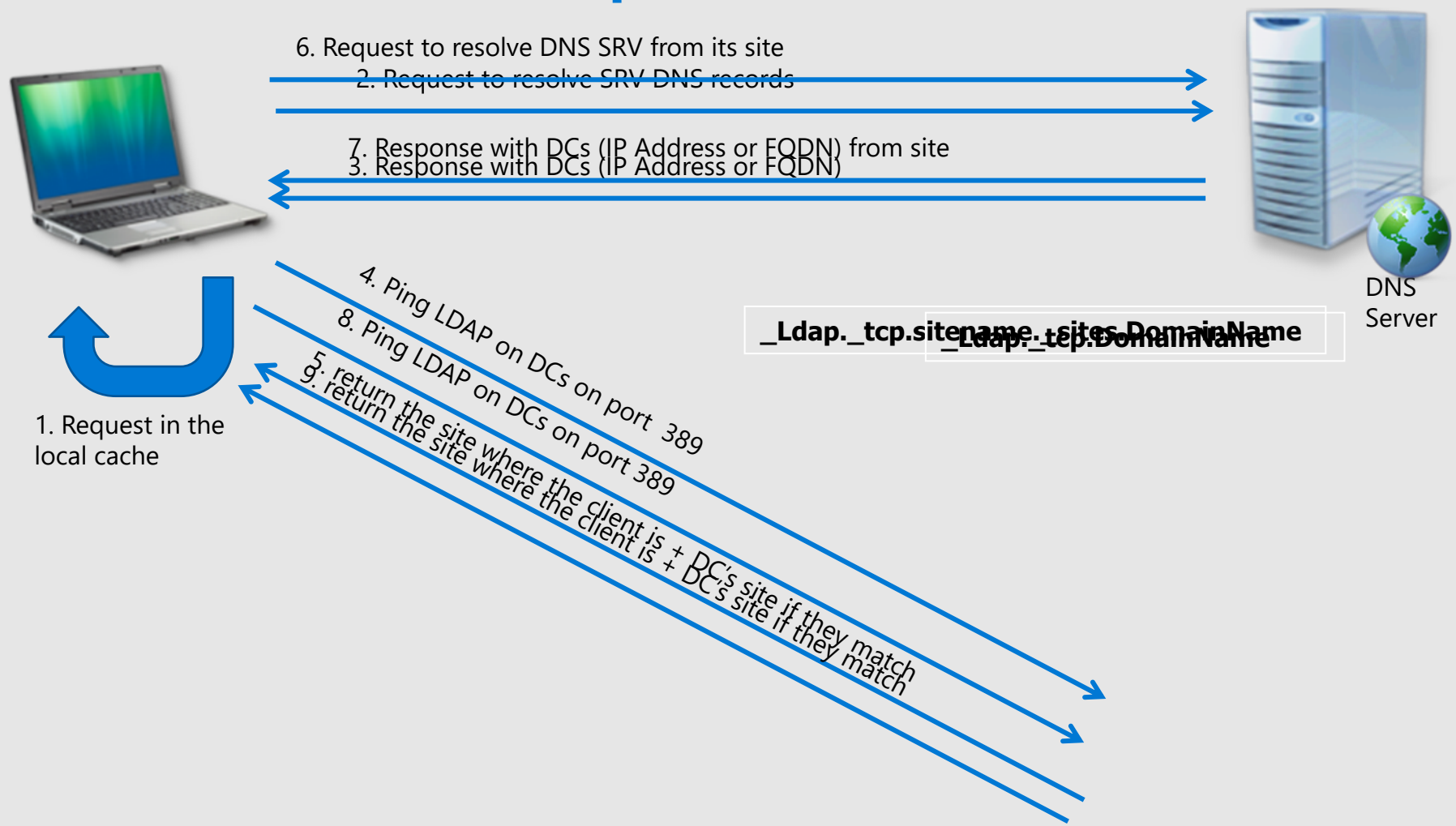
Site specific records

_ldap._tcp.Toronto._Sites.dc._msdcs.contoso.com. = dc1
_kerberos._tcp.Toronto._Sites.dc._msdcs.contoso.com. = dc1
…

Siteless or generic records

_ldap._tcp.dc._msdcs.contoso.com. = dc1
_kerberos._tcp.dc._msdcs.contoso.com. = dc1
…

DC Locator flow in deep



_msdcs Zone

The _MSDCS zone linked to the root domain has any specific role.

- There are only CNAME and SRV record

CNAME: use for AD replication

In the zone racine:

6c014aee-a9a9-403d-8b56-88b77b7e88f._msdcs -> DC1

SVR: use to find on which DC to connect (DC LOCATOR process)

Updated by all Global Catalog (GC) in the

Should be replicated to the entire forest

Domain controllers' records

All the records owned by a DC are stored in the file %windir%\system32\config\netlogon.dns

phoenix.contoso.com	A 157.55.81.157
_ldap._tcp.contoso.com	SRV 0 0 389 phoenix.contoso.com
_kerberos._tcp.contoso.com	SRV 0 0 88 phoenix.contoso.com
_ldap._tcp.dc._msdcs.contoso.com	SRV 0 0 389 phoenix.contoso.com
_kerberos._tcp.dc._msdcs.contoso.com	SRV 0 0 88 phoenix.contoso.com

Record A is registerd by DHCP or DNS client (depending of the OS version)

SRV records are registered by etlogon service.

SRV records

LdapIpAddress	A	<DnsDomainName>
Ldap	SRV	_ldap._tcp.<DnsDomainName>
LdapAtSite	SRV	_ldap._tcp.<SiteName>._sites.<DnsDomainName>
Pdc	SRV	_ldap._tcp.pdc._msdcs.<DnsDomainName>
Gc	SRV	_ldap._tcp.gc._msdcs.<DnsForestName>
GcAtSite	SRV	_ldap._tcp.<SiteName>._sites.gc._msdcs.<DnsForestName>
DcByGuid	**SRV**	**_ldap._tcp.<DomainGuid>.domains._msdcs.<DnsForestName>**
GcIpAddress	A	_gc._msdcs.<DnsForestName>
DsaCname	CNAME	<DsaGuid>._msdcs.<DnsForestName>
Kdc	SRV	_kerberos._tcp.dc._msdcs.<DnsDomainName>
KdcAtSite	SRV	_kerberos._tcp.dc._msdcs.<SiteName>._sites.<DnsDomainName>
Dc	SRV	_ldap._tcp.dc._msdcs.<DnsDomainName>
DcAtSite	SRV	_ldap._tcp.<SiteName>._sites.dc._msdcs.<DnsDomainName>
Rfc1510Kdc	SRV	_kerberos._tcp.<DnsDomainName>
Rfc1510KdcAtSite	SRV	_kerberos._tcp.<SiteName>._sites.<DnsDomainName>
GenericGc	SRV	_gc._tcp.<DnsForestName>
GenericGcAtSite	SRV	_gc._tcp.<SiteName>._sites.<DnsForestName>
Rfc1510UdpKdc	SRV	_kerberos._udp.<DnsDomainName>
Rfc1510Kpwd	SRV	_kpasswd._tcp.<DnsDomainName>
Rfc1510UdpKpwd	SRV	_kpasswd._udp.<DnsDomainName>

SRV Records

Are defined by:

Priority[0]: the client try to contact the server with the lowest priority

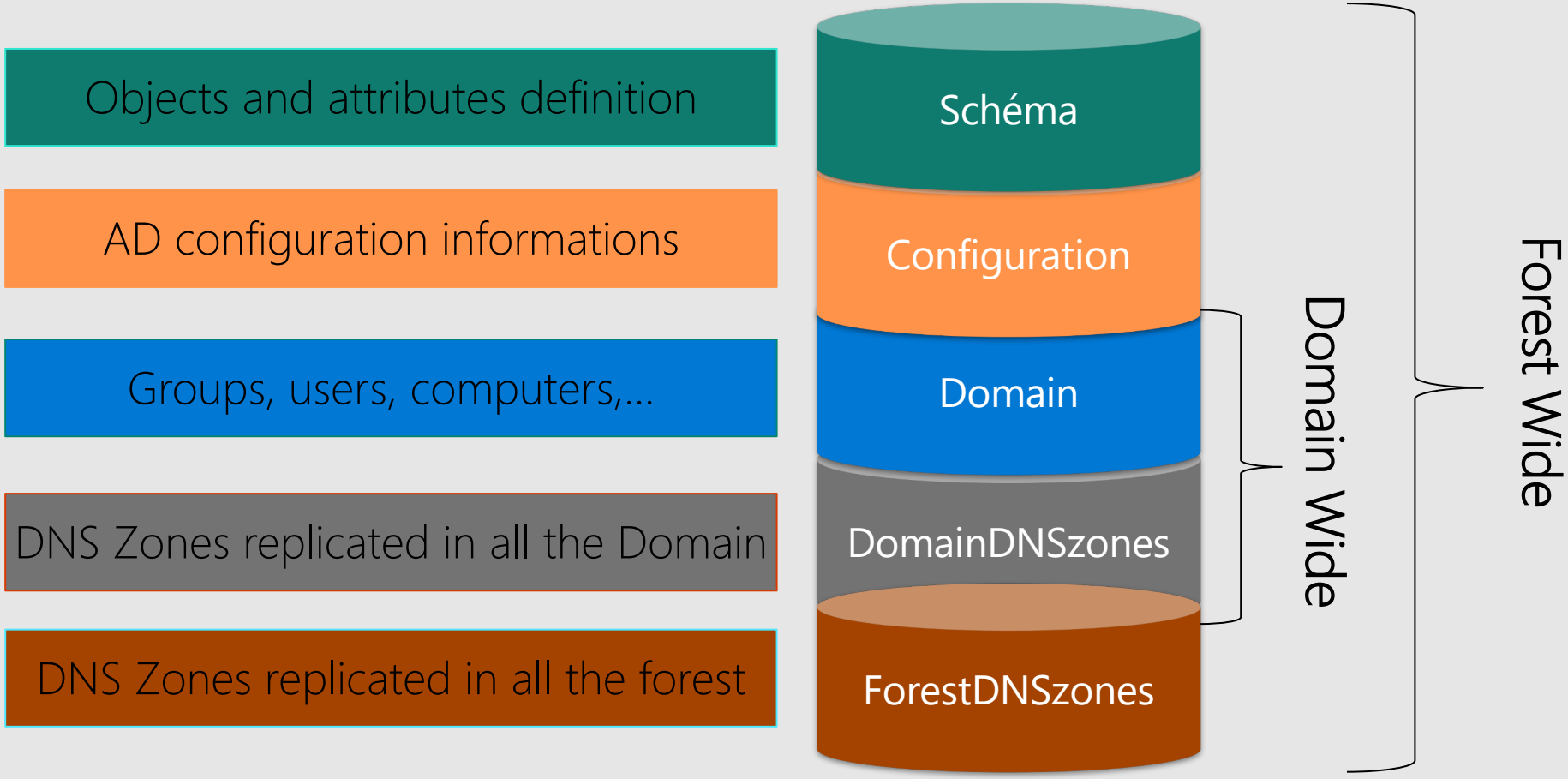
weight[100]: Load sharing for server with the same priority. The client choose randomly the SRV records with a probability link to the weight

Port [389]: service network port

target[domain_controller_name]: FQDN of the target

Applicative Naming context

Created to give more flexibility to the replication management



Applicative Naming context

Partitions subscription

- Subscribe to the CrossRef object associated to the needed partition
- Configuration partition replication is mandatory
- New KCC discover
- Replication link creation

Need to contact the Domain Naming Master server

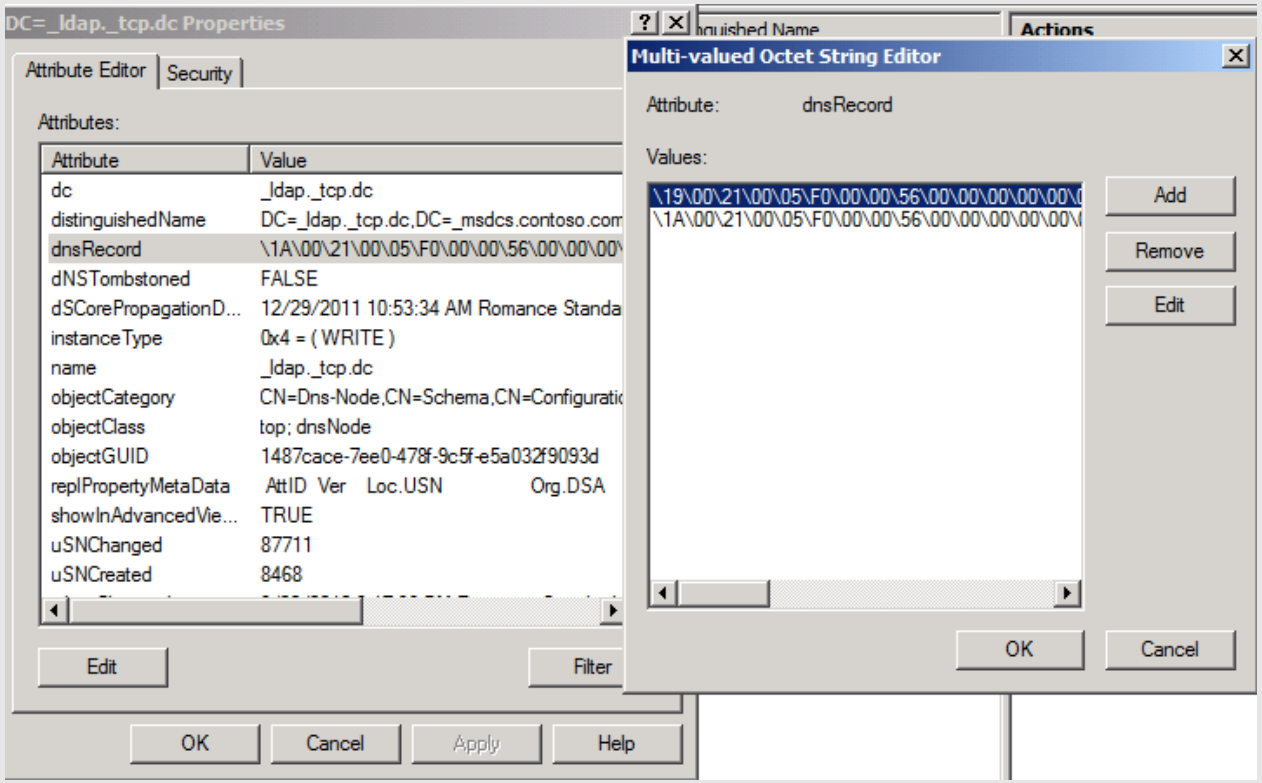
It possible to add new partition with dnscmd or ntdsutil

DNS object in Active Directory

A DNS zone is a « container object » (class DnsZone)

A DNS zone contain « DnsNode leaf object » (class DnsNode)

DnsNodes are multivalued attributes, a DnsRecord which host all records' instance (for example more than one address)

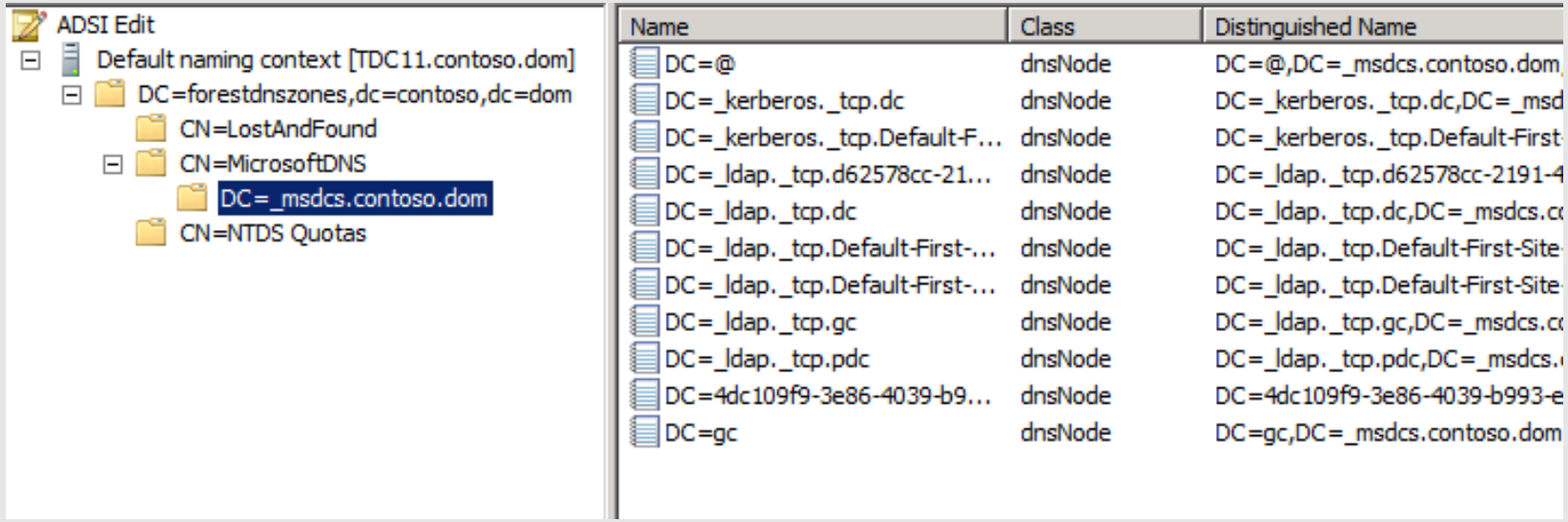


DNS object in Active Directory

Accessible from LDP or ADSIEDIT

dc=forestdnszones,dc=<nom du domaine>

dc=domainednszones,dc=<nom du domaine>



Life Cycle

The record is created in DNS interface and next in AD as DnsNode.

- Use the AD replication

Update are available

- Every 3 mins
- Refresh from the interface

Deletion in the DNS Interface but the record is keep in Active Directory

- Attribute dnsTomstoned set as True and deletion of the other attributes (ex: @IP)
- Final deletion after 7 days at 2h00 am

Once upon a time…

Sticky DCs

- Windows 2000 Server, Windows XP, Windows Server 2000 discovered DCs only if the one they already selected is unresponsive. They did not check if better DCs were available every 12 hours

Dynamic updates

- It is preferred that the DNS server hosting the zone supports dynamic updates. Else the SRV records have to be manually maintained.
- And yep, back in the days, it was common to find BIND implementations for which the dynamic updates were not available or not configured

Chapter

1.1.6

Object management

🎯 Identify the particularities of the different types of objects to manage in AD

Mmh we have only Users and Groups in AD?

Of course NOT! 🧐

Objects in AD are various, like:

- Users
- Groups
- Computers
- OU
- GPO
- Trust Directory Object
- Fine Grained Password Policies

But all come from a Class and have attributes defined in the Schema

- Represented by its entry name, or relative distinguished name (RDN), and by its distinguished name (DN).

All objects with SID are Security Principals but not all objects have SID

AD Objects

An object is a collection of attributes

- The structure of objects (the list of possible attributes) is defined in a schema
- In a tree, an object is uniquely identified by its DN
- In a forest, an object is assigned an immutable **G**lobal **U**nique **ID**entifier
- The user Al in the Accounts OU has the following DN:
  CN=Al,OU=Accounts,DC=contoso,DC=com
- The left part of the DN is also known as **R**elative **DN**
- A security descriptor (controls who has read/modify access)

GUID

RDN

AD Objects & attributes

An attribute has a type

- The types and constraints of attributes are defined in the schema
- Some attributes can be modified, some can’t (owned by the system), some are confidential, some are just pointers to other attributes in other objects (forwardlinks, backlinks), some are calculated on the fly when queried, some replicate, some don’t…

More acronyms please!

When representing a DN, the syntax is as follows

- CN = Common Name
- OU = Organizational Unit Name
- DC = Domain Component

The DN of the OU Accounts under the top tree is

OU=Accounts,DC=contoso,DC=com

The DN of an object called Bored in the sub OU Feeling under the Weather OU which is at the top tree is

CN=Bored,OU=Feeling,OU=Weather,DC=contoso,DC=com

The Schema

Schema define object Classes & their attributes

Schema is a container within the configuration partition & replicated forest wide

- Schema changes are global
- Schema additions are not reversible
  - But can be disabled
- By design only the Schema Admins Groups on the Schema Master can modify the Schema

Each class and attributes are defined by a unique numeric values, by **O**bject **Id**entifiers

OID

OID

Some OIDs in Active Directory Domain Services include:

- Some issued by The ISO for X.500 classes and attributes
- Some issued by Microsoft
- Some issued by You! By using Schema extension

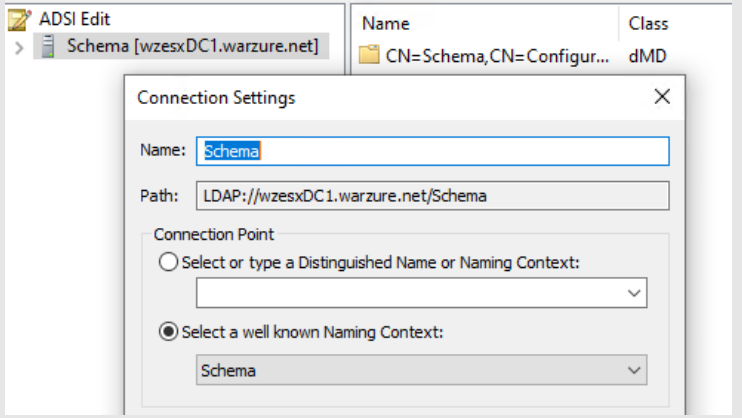
OID notation is a dotted string of numbers, for example "1.2.840.113556.1.5.9", which is described in the following table.

Value	Meaning	Description
1	ISO	Identifies the root authority.
2	ANSI	Group designation assigned by ISO.
840	USA	Country/region designation assigned by the group.
113556	Microsoft	Organization designation assigned by the country/region.
1	Active Directory	Assigned by the organization.
5	Classes	Assigned by the organization.
9	**user** class	Assigned by the organization.

Schema in deep

The distinguished name of the schema container can be found in the **schemaNamingContext** attribute of rootDSE

- Use ADSIEdit to read or modify the Schema



Instances of the **classSchema** class define every object class supported by AD
Instances of the **attributeSchema** class are used to define every attribute supported
Instances of the **attributeSchema** and **classSchema** classes are stored in a well-known place in the directory, the schema container

Schema in deep again

Each Object:

- Has a schemaIDGUID ≠ OID
- Contain default permissions for new object creation*
- Defined which attributes must be set during the creation
  - **mustContain** or **mayContain**

An attribute:

- Has a schemaIDGUID ≠ OID
- Can be indexed to speed up the search
  - The **searchFlags** attribute but required more memory for the DC
- Can be confidential → not replicated to a RODC
  - As the Password Hash 😉
- Can be replicated to all GC or Not
  - The **isMemberOfPartialAttributeSet** attribute

A group, not only one!

A group is represented as a **group** object

2 group types:

- Security Groups (assign access to resources)
- Distribution Groups (for mail delivery)

3 group Scopes:

- Universal, used in all domains in the forest or trusting forests
- Global, used in all domains in the forest or trusting forests
- Domain Local, used in the same domain

Imbrication model:

A	G	U	DL	P
Account	Global	Universal	DomainLocal	Permission

Groups scope in deep

Scope	Possible Members	Scope Conversion	Can Grant Permissions	Possible Member of
Universal	Accounts from any domain in the same forest Global groups from any domain in the same forest Other Universal groups from any domain in the same forest	Can be converted to Domain Local scope if the group is not a member of any other Universal groups Can be converted to Global scope if the group does not contain any other Universal groups	On any domain in the same forest or trusting forests	Other Universal groups in the same forest Domain Local groups in the same forest or trusting forests Local groups on computers in the same forest or trusting forests
Global	Accounts from the same domainOther Global groups from the same domain	Can be converted to Universal scope if the group is not a member of any other global group	On any domain in the same forest, or trusting domains or forests	Universal groups from any domain in the same forestOther Global groups from the same domain Domain Local groups from any domain in the same forest, or from any trusting domain
Domain Local	Accounts from any domain or any trusted domainGlobal groups from any domain or any trusted domain Universal groups from any domain in the same forest Other Domain Local groups from the same domain Accounts, Global groups, and Universal groups from other forests and from external domains	Can be converted to Universal scope if the group does not contain any other Domain Local groups	Within the same domain	Other Domain Local groups from the same domainLocal groups on computers in the same domain, excluding built-in groups that have well-known SIDs

Password Policy

Passwords are for computers and users

Policies are only for users

By default all users have the same policy

- Default Domain Password Policy
  - Defined in the GPO “Default Domain Policy” linked at the Domain level

With 2008 DF, we can create multiple policies

- **F**ine **G**rained **P**assword **P**olicies or Password Setting Object
  - It’s an PSO object in the Domain
- Then apply those policies to users and/or groups in the domain
- Apply stronger policies for privileged accounts

Password Policy

Policies control the rules for passwords

- Minimum password length
- Minimum password age
- Maximum password age
- Password complexity
- Password history

Lockout account policy too

- How many attempts a user has to login before the account is locked
- For how long it is locked, etc.

Once upon a time…

Before FGPPs, how can we have assigned different policies to different users?

- We couldn't! We had to create a separate domain in the same forest if we wanted to enforce a different password policy.
- Or you can trust the admins to use stronger password by diligence.

Chapter

1.1.7

Access control in AD

🎯 Explain the components used in AD to manage permissions and delegations on objects

Access Control, Permissions, Delegations

After authentication, the next step is authorization and access control to protect resources

Delegation inside AD or access to a file are based on the same mechanism:

- Permissions are examined to determine which security principals can access the resource and how they can access it.
- Security principals perform actions (which include Read, Write, Modify, or Full control) on objects
- DACL are used to assign permissions (ACE)

Auditing is implemented as an access control

- SACL are used to assign audit

Discretionary

Each object has a Discretionary Access Control List

DACL

- It is composed of **A**ccess **C**ontrol **E**ntries

ACE

- Each ACE has a scope
- Ordered in a canonical order
  - Deny are always the winner

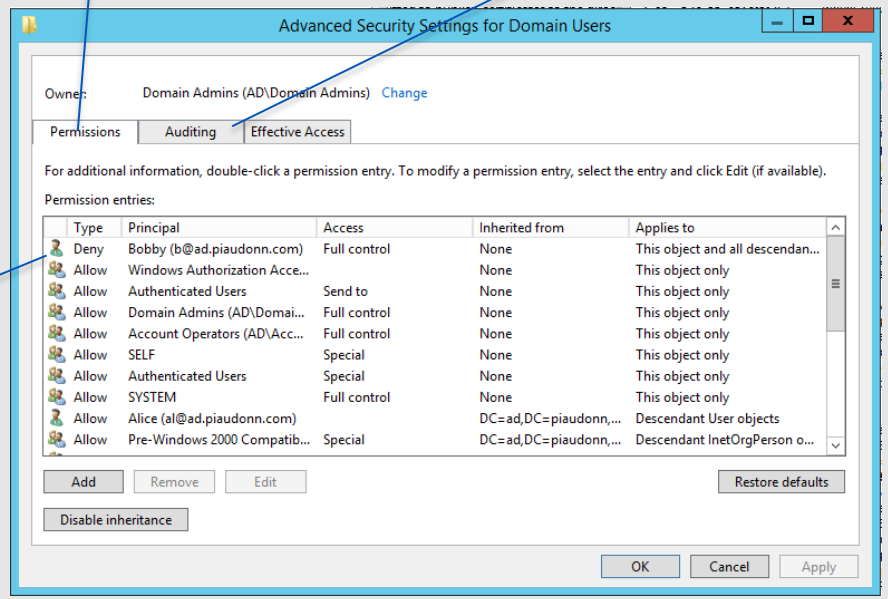
Permissions can be inherited from parent containers

A Owner are full permission on the object

DACL (Discretionary Access Control List)

SACL (System Access Control List also known as Auditing)

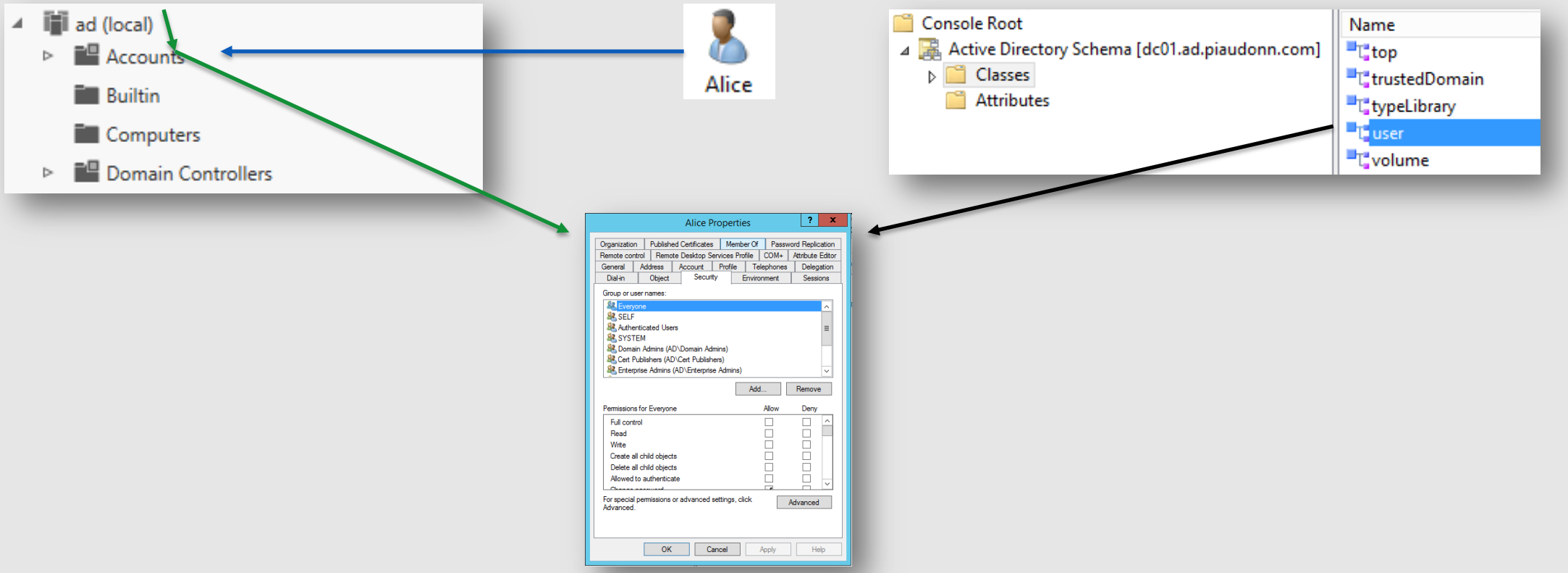
ACE (Access Control Entry)



Default Discretionary

defaultSecurityDescriptor

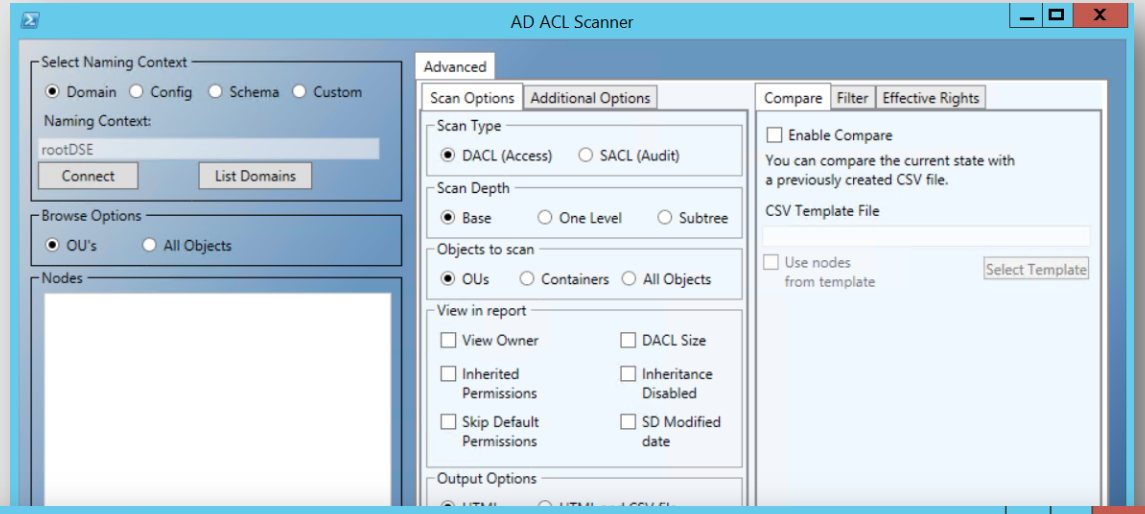
- Each class comes with its default DACL which will be applied directly on the object

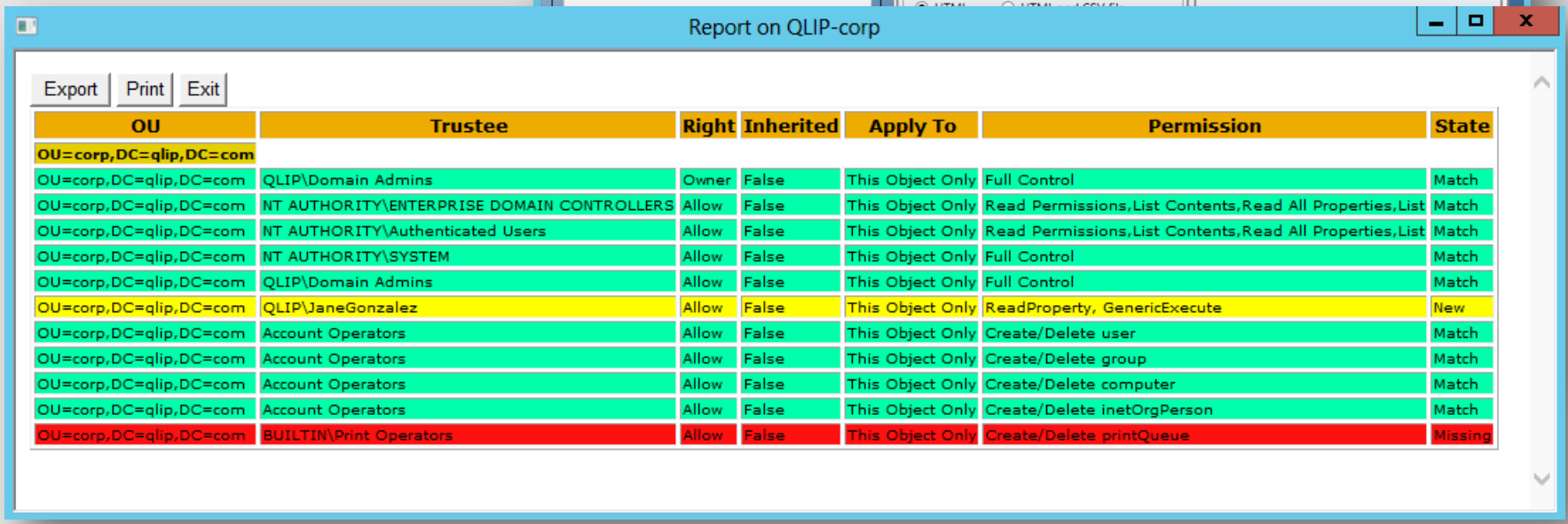


A tool to create reports of permissions in AD

Active Directory ACL Scanner

- This is a Windows PowerShell script
- Comparison of previous results:
  - What is new
  - What is missing
  - What hasn't changed





OU	Trustee	Right	Inherited	Apply To	Permission	State
OU=corp,DC=qlip,DC=com						
OU=corp,DC=qlip,DC=com	QLIP\Domain Admins	Owner	False	This Object Only	Full Control	Match
OU=corp,DC=qlip,DC=com	NT AUTHORITY\ENTERPRISE DOMAIN CONTROLLERS	Allow	False	This Object Only	Read Permissions,List Contents,Read All Properties,List	Match
OU=corp,DC=qlip,DC=com	NT AUTHORITY\Authenticated Users	Allow	False	This Object Only	Read Permissions,List Contents,Read All Properties,List	Match
OU=corp,DC=qlip,DC=com	NT AUTHORITY\SYSTEM	Allow	False	This Object Only	Full Control	Match
OU=corp,DC=qlip,DC=com	QLIP\Domain Admins	Allow	False	This Object Only	Full Control	Match
OU=corp,DC=qlip,DC=com	QLIP\JaneGonzalez	Allow	False	This Object Only	ReadProperty, GenericExecute	New
OU=corp,DC=qlip,DC=com	Account Operators	Allow	False	This Object Only	Create/Delete user	Match
OU=corp,DC=qlip,DC=com	Account Operators	Allow	False	This Object Only	Create/Delete group	Match
OU=corp,DC=qlip,DC=com	Account Operators	Allow	False	This Object Only	Create/Delete computer	Match
OU=corp,DC=qlip,DC=com	Account Operators	Allow	False	This Object Only	Create/Delete inetOrgPerson	Match
OU=corp,DC=qlip,DC=com	BUILTIN\Print Operators	Allow	False	This Object Only	Create/Delete printQueue	Missing

Extended Rights

An ACE usually specifies a class and attributes

An ACE can also be specifying an Extended Right

- Permission to perform an “action” against the object on which it is set
- The permission to reset a password for example, or the permission to replicate the database…

Used to assign special action depending the class of the object

Can be powerful:

Ex: Replicate Directory Change All → assign Password Hash read → Golden Ticket[redacted]

Mad delegation

What if the helpdesk people can reset the password of a domain admin?

The adminSDHolder insures that only AD admins can touch AD admins accounts

adminSDHolder, what ?

Mechanism created to protect high privileges accounts

Container located in System

Object protected by adminSDHolder have :

- The inheritance disabled
- All their permissions replaced by those set on the adminSDHolder container

This mechanism will be applied regardless the object location

adminSDHolder

The DACL of the adminSDHolder is the DACL of all AD admins

- All direct members and nested members of default built-in admin groups are protected
- Every hour… It:

  1. Compares the ntSecurityDescriptor of the objects with the ntSecurityDescriptor of the adminSDHolder object
  2. If it is different, then does the following:
     - Disable the inheritance on the object
     - Reset the attribute to the adminSDHolder ntSecurityDescriptor value
     - Set the adminCount attribute to the value 1

Accounts protected by the AdminSDHolder

Account / Backup / Print / Server Operators

Administrators

Domain Admins

Enterprise Admins

Schema Admins

Read-Only Domain Controllers

Replicator

Enterprise/Key Admins

Administrator

Krbtgt

Domain Controllers

adminSDHolder Mecanism

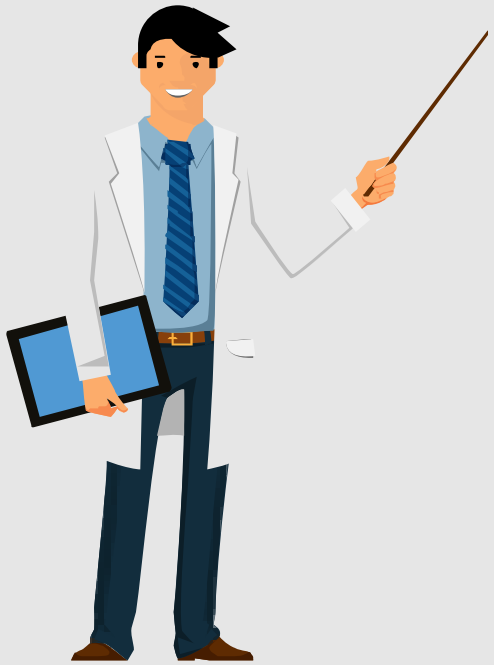
60 minutes

PDC

AdminCount = 1

~~Héritage~~

Professor Useful



It is possible to unprotect certain groups from the adminSDHolder

- The “* Operators” groups could be unprotected

There is one object which is not a security principal but has an objectSID

- It is the default Builtin container

It is possible to create dynamic objects in AD

- Although not used by any functionality of Windows
- It can be used by an attacker to create temporary accounts

Chapter

1.1.8

The default administrators

🎯 Describe the permissions and privileges of privileged groups

Built-in Users & Groups

By design AD contains Built-in Users & Groups

- Ease the delegation
- Ease the authorization
- Ease the administration

They named well-known security principals

- Represent special identities
  - such as Everyone, Local System, Principal Self, Authenticated User, Creator Owner, and so on.
- Stored in the WellKnown Security Principals container beneath the Configuration container.

Some are constant across all systems

- Ex: Administrator, created during the AD promotion, is named RID500
- Always build with <Domain SID>-**500**

(note: -500 is the RID of the objectSID)

Some (important) default users

Administrator account S-1-5-<domain>-500

- Created during Active Directory promotion.
- The most powerful account in the domain.

Guest account S-1-5-<domain>-501

- Default local account with limited access
- Disabled by default

KRBTGT account S-1-5-<domain>-502

- Service account for the Key Distribution Center (KDC) service
- Cannot be deleted, cannot be changed, cannot be enabled
- Different for each RODC but the same for all RWDC

Default privileged security groups

Enterprise Admins (EA)	Full control on all naming contexts, but the schema
Schema Admins	Full control on the schema naming context
Domain Admins (DA)	Full control on the domain naming context and all machines
Administrators	Full control on the domain naming context and all domain DCs
Account Operators	Full control on the users/groups/computers in the domain
Backup Operators	Can backup/restore naming contexts
Server Operators	Can control some settings of DCs
Print Operators	Can manage print queues in the domain NC and printers installed locally on DCs

Chapter

1.1.9

The use of the LDAP protocol to query directory data

🎯Explain the possibilities of integrating AD with the LDAP protocol

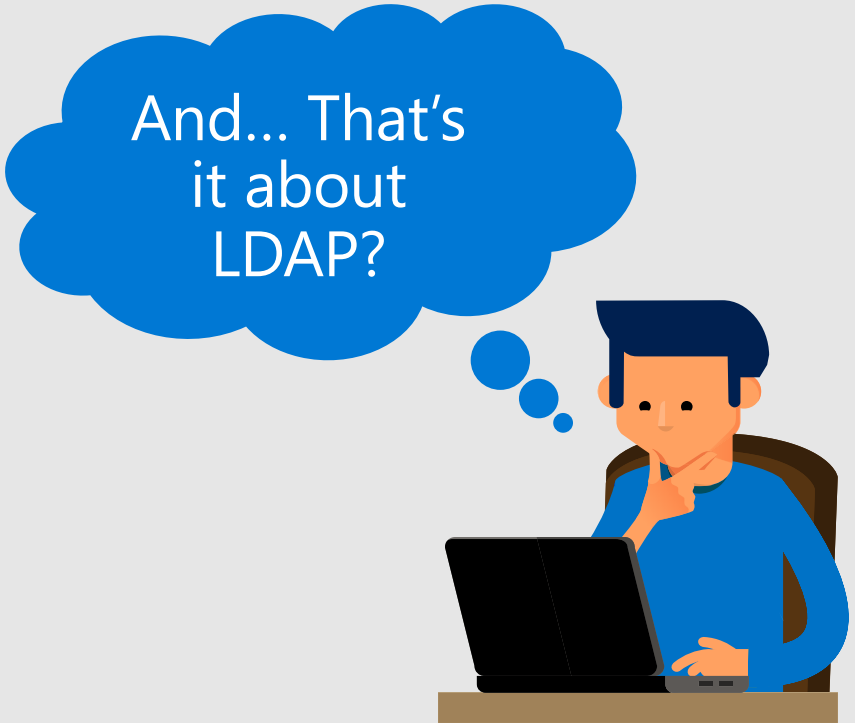
LDAP

Active Directory does LDAP

- LDAP v2 and LDAP v3
- On the network, it's TCP 389 if you want to query information in the default naming contexts
- Or you can query the global catalog, it's on TCP 3268
- You cannot change those two ports

The are other APIs available to query AD

- SAM-R
- WMI
- Active Directory Web Services
- DNS



LDAP query 101

Filter

- (&(objectClass=user)(userPrincipalName=pierre@contoso.com))

Base

- DC=contoso,DC=COM

Scope

- Subtree / Base / Onelevel

Attributes

- whenCreate,displayName

LDAP query 102

Page Size

- A query returns a maximum of 1,000 records (can be changed)
- If more than 1,000 are returned, the client has to handle the pages of records (request them, keep track with a cookie and store them)

Timeout

- Client and server settings

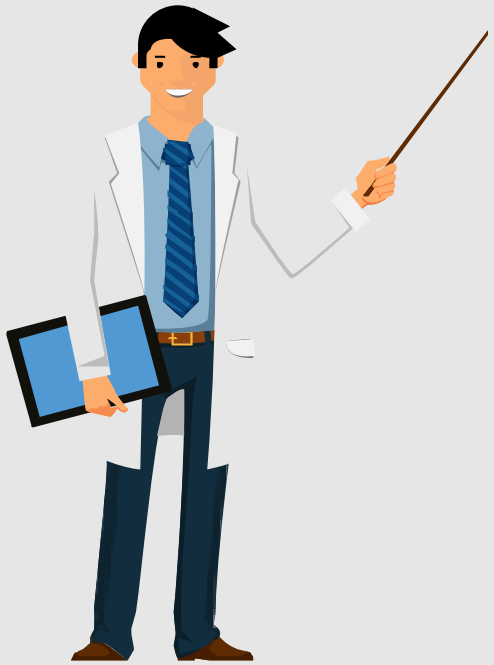
Referral Chasing

- Follow or don’t reference to another naming context

LDAP extensions

- SHOW_DELETED Control: 1.2.840.113556.1.4.417
- SHOW_RECYCLED Control: 1.2.840.113556.1.4.2064

Professor Useful



- Optimizing queries
  - Control GET_STATS: 1.2.840.113556.1.4.970
- LDAP Policies
  - Control the server's options
    - How many objects can be returned at once
    - What's the time out for LDAP connections
    - How many threads are used for LDAP queries
  - One policy for the forest (by default)
  - Other policies can be created and linked to a specific site or even a specific DC

And LDAPS?

The security of the Directory Service can be greatly improved implementing:

- LDAP Signing
- LDAP Channel Binding

Mitigate Man-in-the-middle attack

- Required signing
  - Confirms the integrity of the LDAP payload data using secret key technology
- Required sealing
  - Encrypts the LDAP payload data to avoid transmitting sensitive information in clear text.

Required appropriate Certificate on Domain Controllers

- Certificate must be valid for the purpose of Server Authentication.
- The Subject name must match the Fully Qualified Domain Name (FQDN) of the host machine: Subject:CN=dc01.contoso.com.
- The host machine account must have access to the private key.

How do I know what LDAP features are available on a DC?

The root directory service entry tells you what is available on a domain controller

RootDSE

```
Established connection to .
Retrieving base DSA information...
Getting 1 entries:
Dn: (RootDSE)
configurationNamingContext: CN=Configuration,DC=piaudonn,DC=com;
currentTime: 4/6/2018 2:15:15 PM Coordinated Universal Time;
defaultNamingContext: DC=piaudonn,DC=com;
dnsHostName: PIDC01.piaudonn.com;
domainControllerFunctionality: 7 = ( WIN2016 );
domainFunctionality: 7 = ( WIN2016 );
forestFunctionality: 7 = ( WIN2016 );
isGlobalCatalogReady: TRUE;
namingContexts (5): DC=piaudonn,DC=com; CN=Configuration,DC=piaudonn,DC=com; CN=Schema,CN=Configuration,DC=piaudonn,DC=com;
DC=DomainDnsZones,DC=piaudonn,DC=com; DC=ForestDnsZones,DC=piaudonn,DC=com;
rootDomainNamingContext: DC=piaudonn,DC=com;
schemaNamingContext: CN=Schema,CN=Configuration,DC=piaudonn,DC=com;
supportedCapabilities (6): 1.2.840.113556.1.4.800 = ( ACTIVE_DIRECTORY ); 1.2.840.113556.1.4.1670 = ( ACTIVE_DIRECTORY_V51 );
1.2.840.113556.1.4.1791 = ( ACTIVE_DIRECTORY_LDAP_INTEG ); 1.2.840.113556.1.4.1935 = ( ACTIVE_DIRECTORY_V61 ); …
supportedLDAPVersion (2): 3; 2;
…
```

Microsoft

List of abbreviations

AD – Active Directory
ADDS – Active Directory Domain Services
ACE - Access Control Entry
CN - Common Name
CSE - Client Side Extension
DA - Domain Admins
DACL - Discretionary Access Control List
DC - Domain Controller
DDCP - Default Domain Controller Policy
DDP - Default Domain Policy
DFL - Domain Functional Level
DIR - Directory Information Tree
DN - Distinguished Name
EA - Enterprise Admins
FAST - Flexible Authentication Secure Tunneling
FFL - Forest Functional Level
FGGP - Fine Grained Password Policy
FSMO - Flexible Single Master Operation
GC - Global Catalog
GPC - Group Policy Container
GPO - Group Policy Object
GPP - Group Policy Preference
GPT - Group Policy Template
GUID - Global Unique Identifier
ISTG - Inter-Site Topology Generator
KCC - Knowledge Consistency Checker
KDC - Key Distribution Center
LMHash - Lan Manager Hash
LSASS - Local Security Authority Subsystem Service
NC - Naming Context
NTHash - New Technology Lan Manager Hash
OU - Organizational Unit
PAC - Privilege Attribute Certificate
PDC - Primary Domain Controller
RDN - Relative Distinguished Name
RID - Relative Identifier
RSOP - Resultant Set of Policy
S4U2Proxy - Service for User to Proxy
S4U2Self - Service for User to Self
SACL - System Access Control List
SID - Security Identifier
SPN - Service Principal Name
SSO - Single Sign-On
SSP - Security Support Provider
SSPI - Security Support Providers Interface
TGS - Ticket Granting Service
TGT - Ticket Granting Ticket
UPN - User Principal Name
USN - Update Sequence Number
WH4B - Windows Hello for Business