

# Region Nordjylland

Teknisk kursus, dag 2

# Agenda for dag 2

1

Grundlæggende Netværk

Netværkskomponenter

Databaser

4

Sikkerhed/Authentication

# Sikkerhed/Authentification

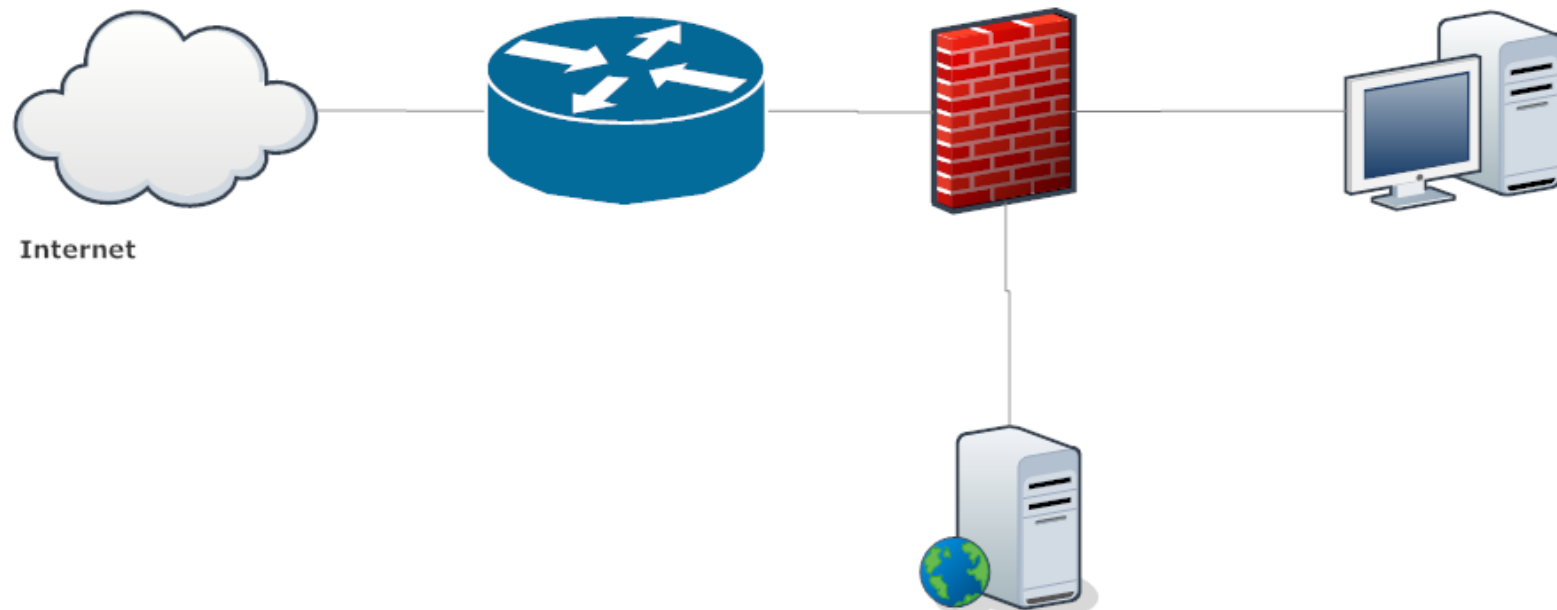
- Firewall
- CIA
- Kryptering
- Authentification
- Active Directory

# Firewall



# Grundlæggende virkemåde af en firewall.

- En firewall er en enhed (eller software) der anvendes til at styre flowet af data ind og ud af et netværk. Generelt installeres en firewall for at forhindre hackerangreb fra internettet og for at kontrollere den data der sendes ud fra et netværk.



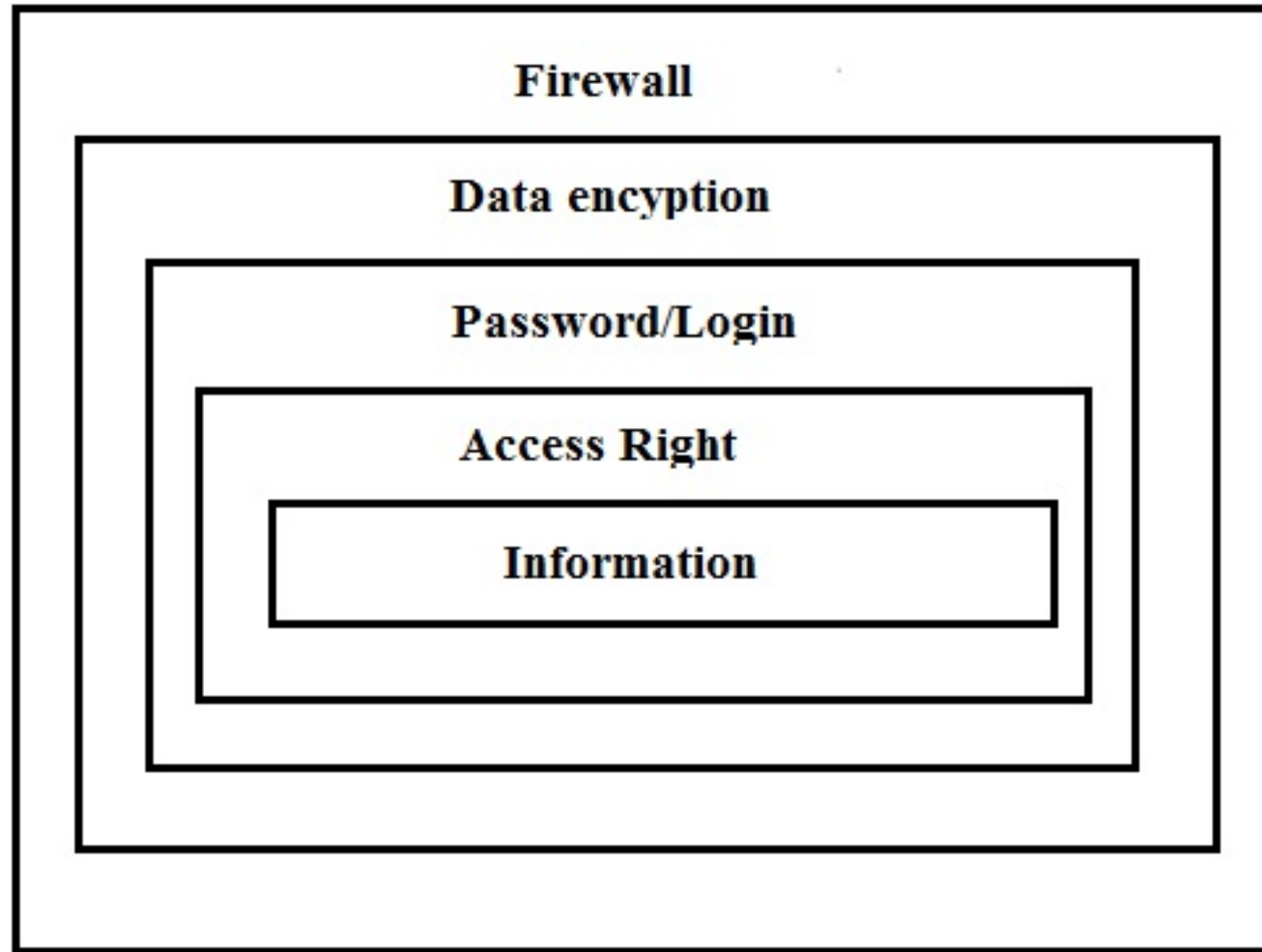


A firewall is a computer traffic cop that permits or blocks data flow between two parts of a network architecture. It is the only link between parts.

# Hvad er et ”hackerangreb”, nogle eksempler

- Port scanning for at få oplysninger om hvilke services der kører på netværk og på servere.
- Bevidst forsøg på at få en services eller en server til at gå ned.
- Forsøg på at få adgang til en computer for at bruge dens ressourcer
- Forsøg på at få adgang til de information der findes på computer.
- Forsøg på at ændre de informationer der findes på computer.

# Sikkerhedsmodel for netværk og services







Firewalls enforce predetermined rules governing what traffic can flow

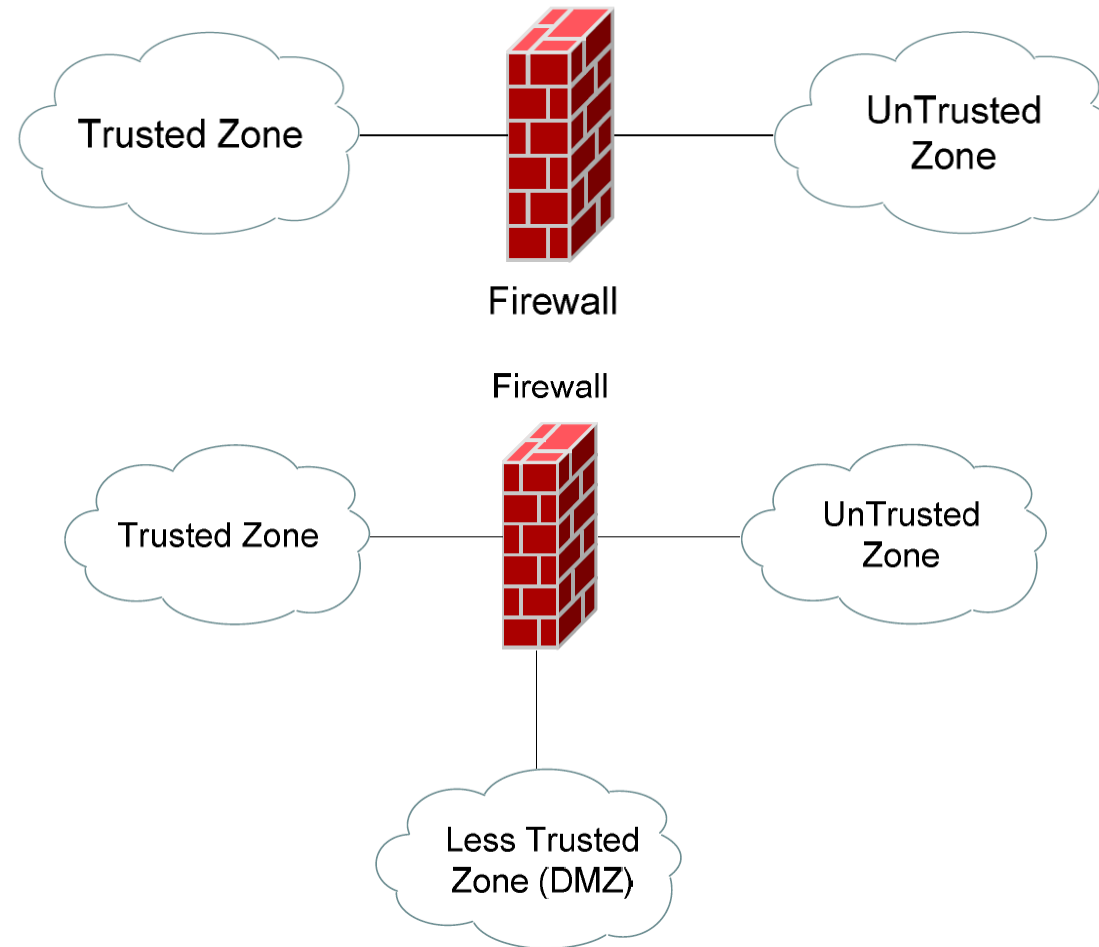
# Hoved funktioner i standard firewall

Alle former for firewalls dele nogle generelle egenskaber og funktioner der sikre at firewallen kan løse den generelle opgave med at kontrollere datatrafik og sikre mod angreb.

Følgende er grundlæggende funktioner som de fleste firewall i større eller mindre grad kan udføre:

- Administration og styring af netværkstrafik
- Autentificere adgang
- Beskyttelse ressourcerne om uautoriseret adgang og forsøg på at hacke enhed
- Registrering og rapportering af hændelser
- Optræde som proxy

# Begreber i forbindelse med en firewall's virkemåde



# Policy

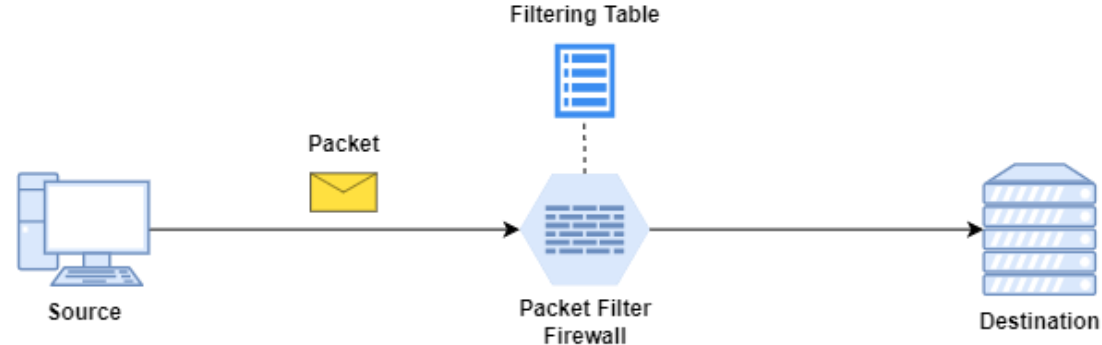
- A firewall implements a security policy, that is, a set of rules that determine what traffic can or cannot pass through the firewall.
- As with many problems in computer security, we would ideally like a simple policy, such as “good” traffic can pass but “bad” traffic is blocked.
- Unfortunately, defining “good” and “bad” is neither simple nor algorithmic.
- Firewalls come with example policies, but each network administrator needs to determine what traffic to allow into a particular network.

# Example of firewall policy

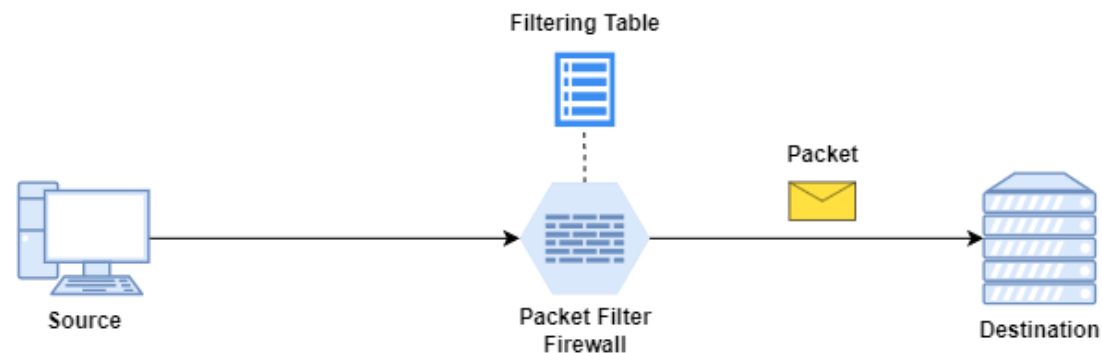
Rule	Type	Source Address	Destination Address	Destination Port	Action
1	TCP	*	192.168.1.*	25	Permit
2	UDP	*	192.168.1.*	69	Permit
3	TCP	192.168.1.*	*	80	Permit
4	TCP	*	192.168.1.18	80	Permit
5	TCP	*	192.168.1.*	*	Deny
6	UDP	*	192.168.1.*	*	Deny

# Packet filtering firewall

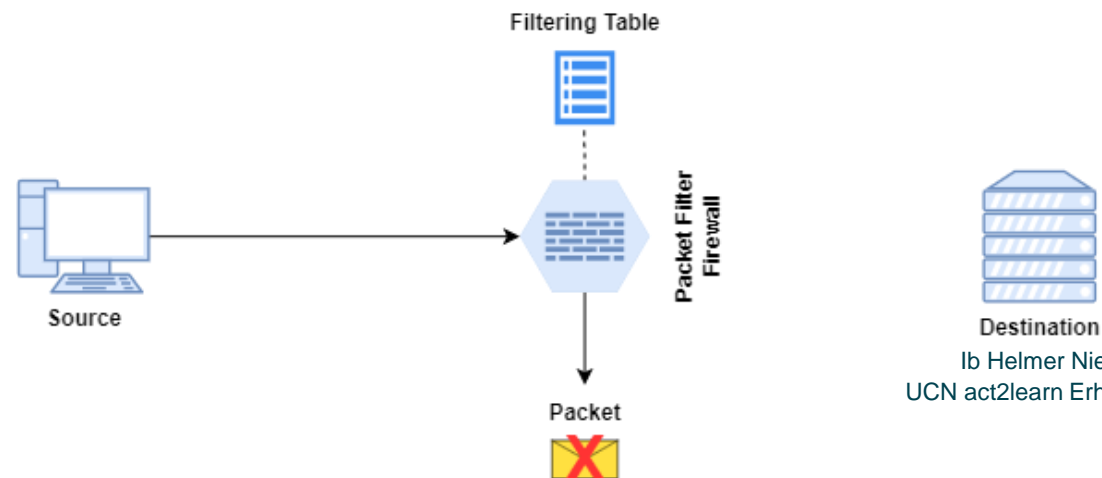
- Øverste tegning: Pakke sendt til destination med “lovlig” ip og port
- Miderste tegning: Pakke slipper gennem firewall
- Nederste tegning: Pakke sendt til ulovlig ip eller port smides væk.



If the packet obeys all rules, it is forwarded to the destination:



Otherwise, it is discarded and never reaches the destination:



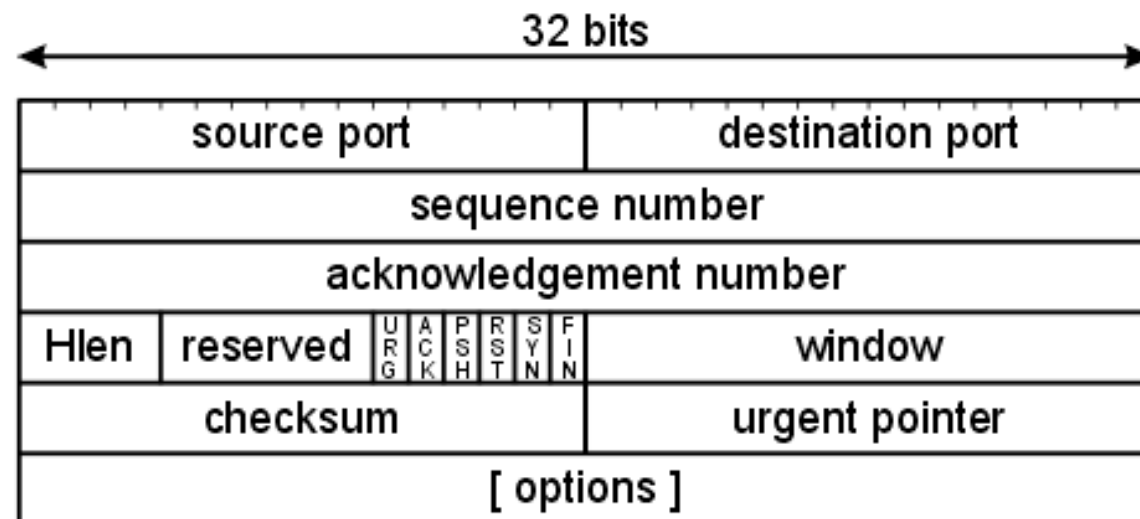
# Packet Filtering

Foregår ved at undersøge de enkelte pakker der ankommer til firewallen ud fra f.eks:

- Source address
- Destination address
- Protocol Type (IP, TCP, UDP, ICMP, ESP, etc.)
- Source Port
- Destination Port









0	3	4	7	8	15	16	31
Version		Length		Type of Service IP Prec or DSCP		Total Length	
Identifier					Flags	Fragmented Offset	
Time to Live			Protocol		Header Checksum		
Source IP Address							
Destination IP Address							
Options and Padding							

## TCP header format







# Packet Filtering

- Pakker der opfylder en bestemt regel kan herefter enten accepteres eller forkastes (permitted or denied).
- De enkelte regler i firewallen opstilles i en prioriteret række og det er normal at den sidste regel i en firewall dropper hvad der ikke er fanget af regler tidligere:

Inbound Rules <a href="#">(Add New)</a>							
	Active	Interface	Source	Service	Conn. State(s)	Action	Log
1	<input checked="" type="checkbox"/>	eth0	any (0.0.0.0/0)	http (tcp/80)	<a href="#">ESTABLISHED,NEW</a>	ACCEPT	<input type="checkbox"/>  
1	<input checked="" type="checkbox"/>	eth0	any (0.0.0.0/0)	https (tcp/443)	<a href="#">ESTABLISHED,NEW</a>	ACCEPT	<input type="checkbox"/>  
1	<input checked="" type="checkbox"/>	eth0	any (0.0.0.0/0)	ssh (tcp/22)	<a href="#">ESTABLISHED,NEW</a>	ACCEPT	<input checked="" type="checkbox"/>  
1	<input checked="" type="checkbox"/>	any	any	any	<a href="#">ANY</a>	DROP	<input type="checkbox"/>  

Outbound Rules <a href="#">(Add New)</a>							
	Active	Interface	Destination	Service	Conn. State(s)	Action	Log
1	<input checked="" type="checkbox"/>	eth0	Web Servers	http (tcp/80)	<a href="#">ESTABLISHED,NEW</a>	ACCEPT	<input type="checkbox"/>  
1	<input checked="" type="checkbox"/>	eth0	Web Servers	https (tcp/443)	<a href="#">ESTABLISHED,NEW</a>	ACCEPT	<input type="checkbox"/>  

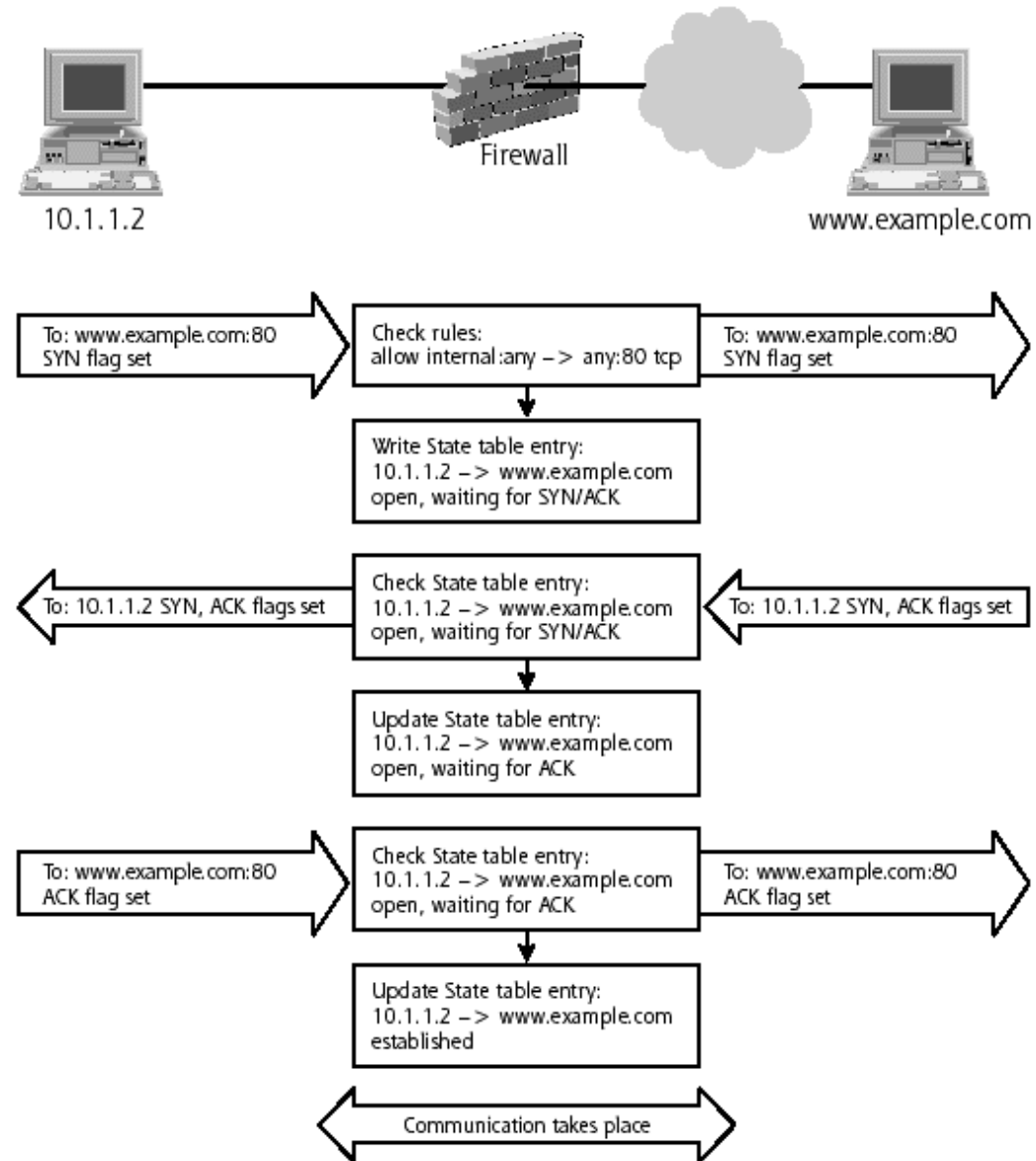


# Stateful inspection firewall

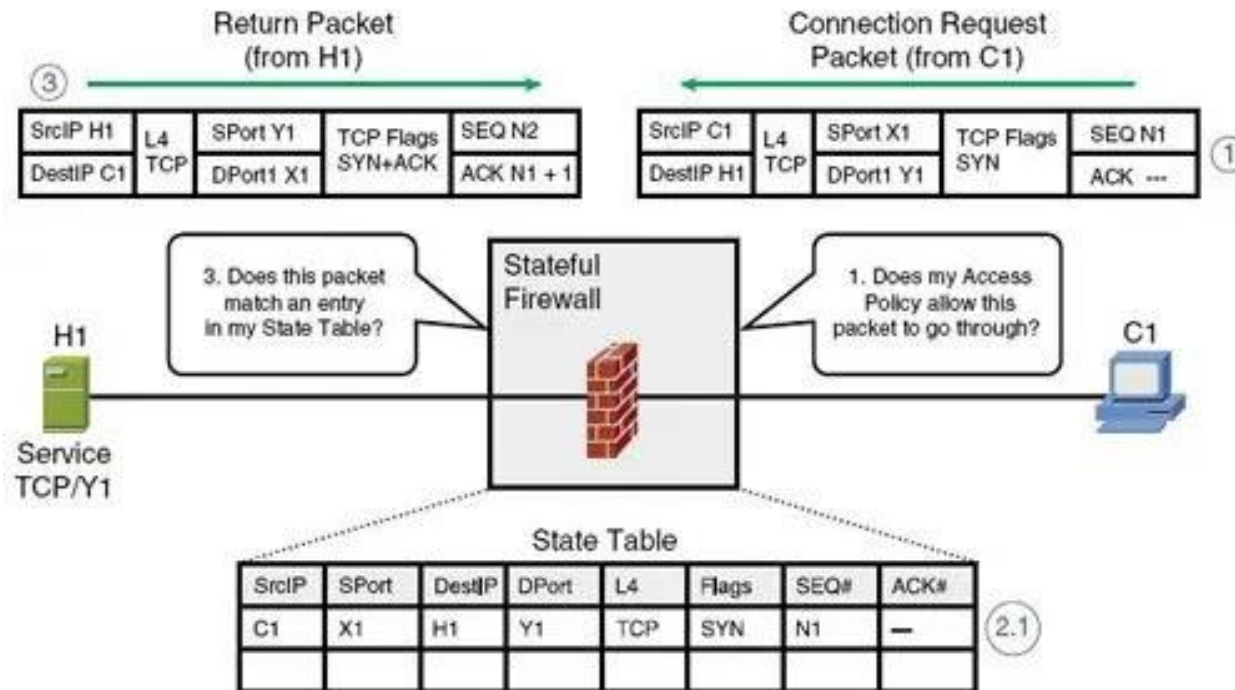
- A stateful inspection firewall maintains state information from one packet to another in the input stream.

SrcIP	SrcPort	DestIP	DestPort	Protocol	Flags	Srq#	ACK#

# Example



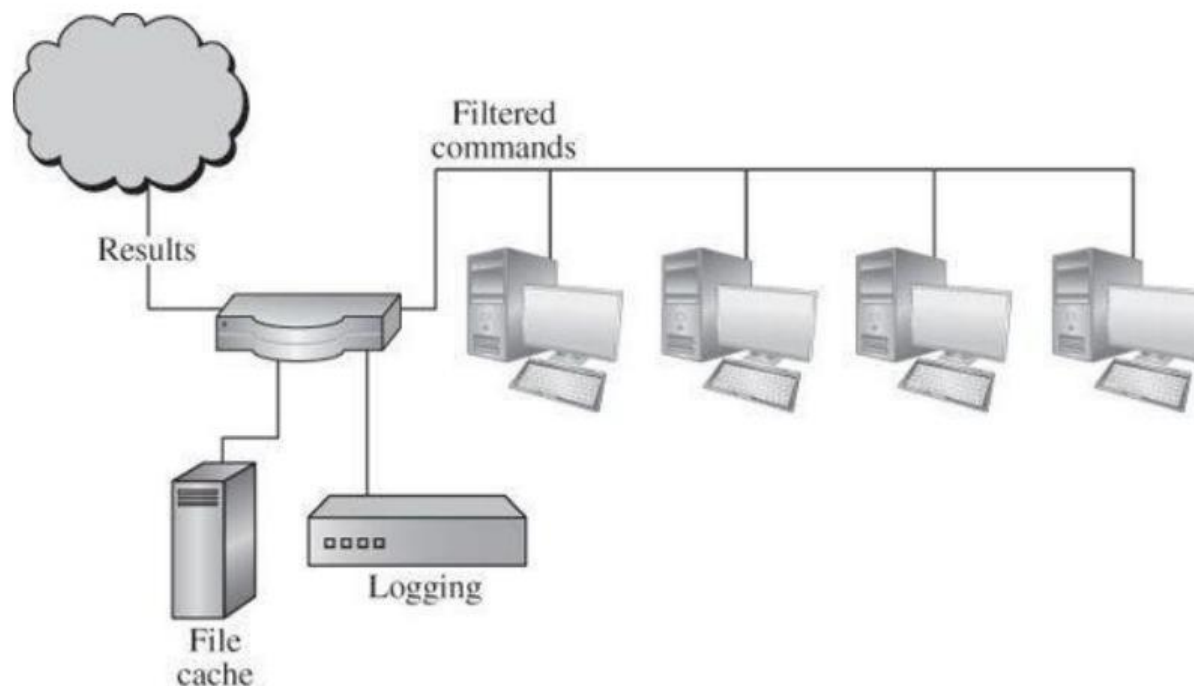
# Stateful Packet Inspection



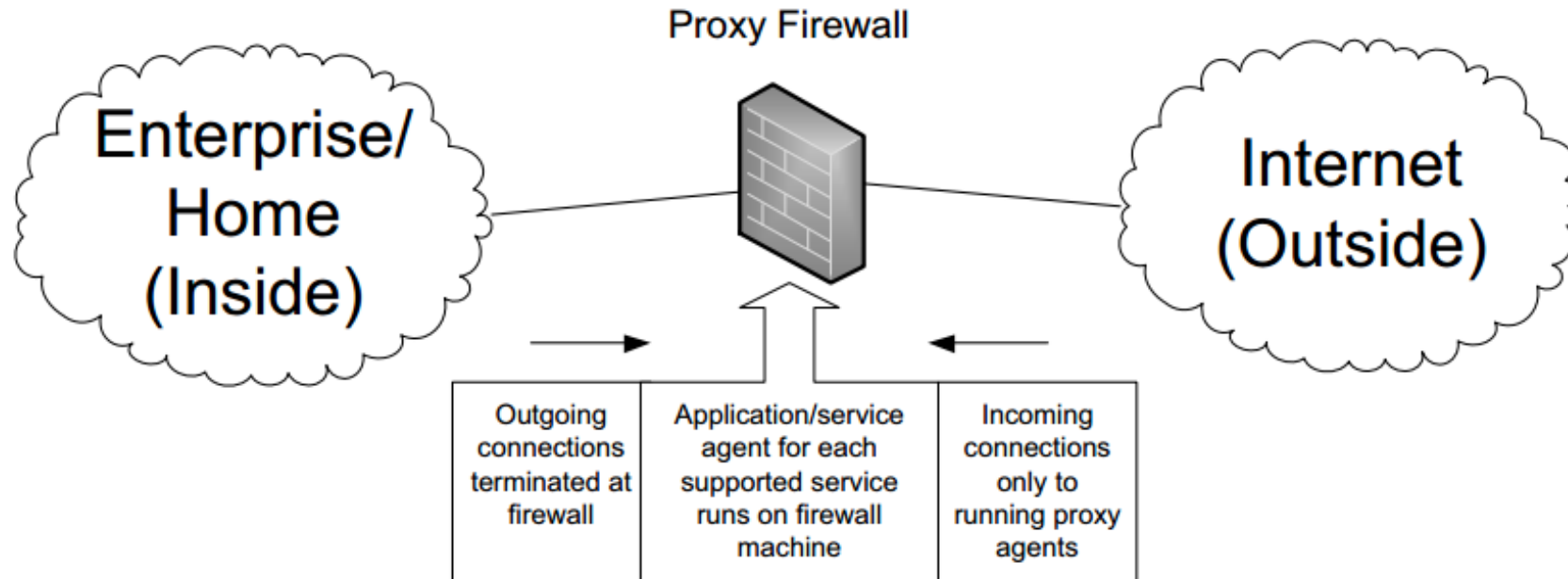
Overview of Stateful Firewalls

# Proxying firewall

- A proxy gateway is a two-headed device: From inside, the gateway appears to be the outside (destination) connection, while to outsiders the proxy host responds just as the insider would. In fact, it behaves like a man in the middle



# Proxying firewall

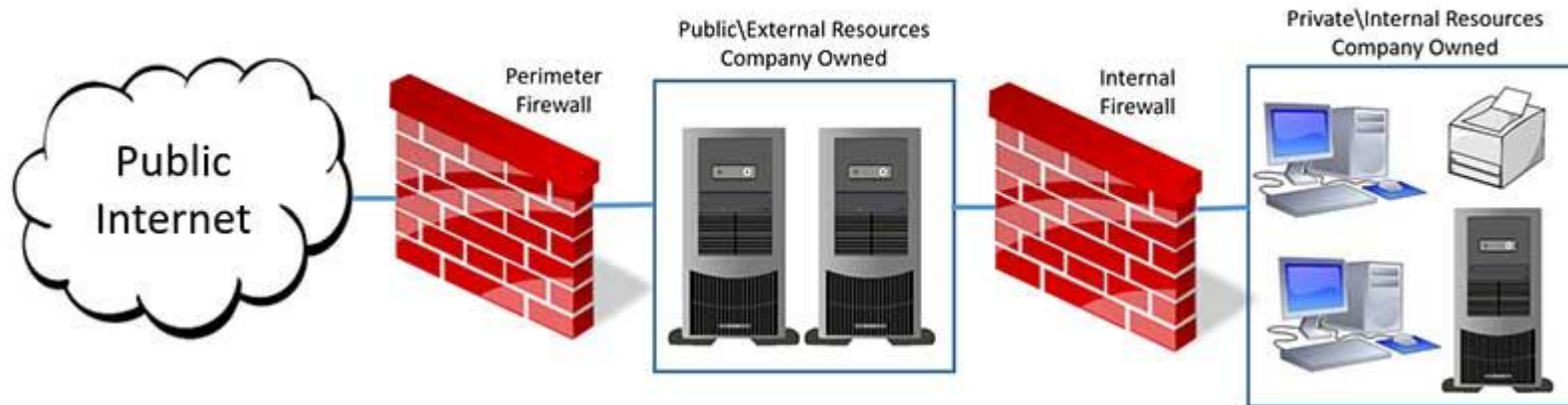


<https://www.youtube.com/watch?v=ZUIKfNX8tA4>

Packet Filter	Stateful Inspection	Application Proxy	Circuit Gateway	Guard	Personal Firewall
Simplest decision-making rules, packet by packet	Correlates data across packets	Simulates effect of an application program	Joins two subnetworks	Implements any conditions that can be programmed	Similar to packet filter, but getting more complex
Sees only addresses and service protocol type	Can see addresses and data	Sees and analyzes full data portion of pack	Sees addresses and data	Sees and analyzes full content of data	Can see full data portion
Auditing limited because of speed limitations	Auditing possible	Auditing likely	Auditing likely	Auditing likely	Auditing likely
Screens based on connection rules	Screens based on information across multiple packets—in either headers or data	Screens based on behavior of application	Screens based on address	Screens based on interpretation of content	Typically, screens based on content of each packet individually, based on address or content
Complex addressing rules can make configuration tricky	Usually preconfigured to detect certain attack signatures	Simple proxies can substitute for complex decision rules, but proxies must be aware of application's behavior	Relatively simple addressing rules; make configuration straightforward	Complex guard functionality; can be difficult to define and program accurately	Usually starts in mode to deny all inbound traffic; adds addresses and functions to trust as they arise

# DMZ (Demilitarized zone)

## DMZ (Demilitarized Zone)





# Her af navnet DMZ





# Refleksion over DMZ (10min)

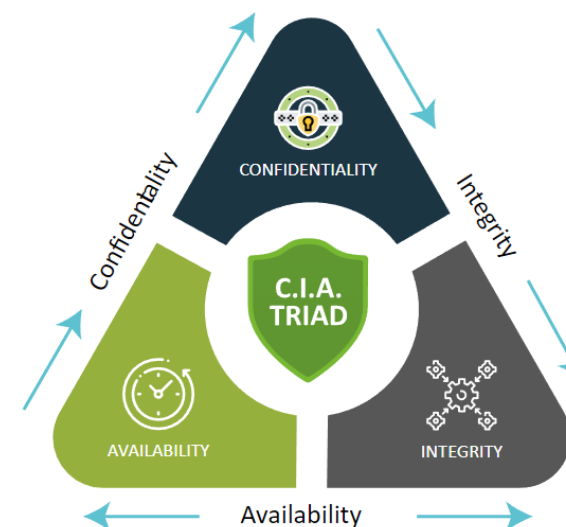
- Hvad er formålet med et DMZ
- Diskuter hvorfor det er en fordel ud fra et sikkerhed perspektiv at placere nogle services i en DMZ.
- Hvilke services kan med fordel placeres i et DMZ.
- Er der nogle ulemper ved at placere services i et DMZ ?

# CIA

- ❖ Confidentiality),
- ❖ Integrity,
- ❖ Availability

# CIA-Triaden

- CIA-triaden er en grundlæggende sikkerhedsmodel inden for informationssikkerhed, der repræsenterer tre centrale mål:
  - fortrolighed (confidentiality),
  - integritet (integrity) og
  - tilgængelighed (availability).



# CIA-Triaden Fortrolighed

- Fortrolighed handler om at sikre, at oplysninger kun er tilgængelige for de autoriserede parter.
- Dette indebærer at beskytte data mod uautoriseret adgang, læsning, kopiering eller videregivelse.
- Kryptering, adgangskontrol og sikkerhedspolitikker er nogle af de metoder, der anvendes til at opnå fortrolighed.

# CIA-Triaden Integritet

- Integritet handler om at sikre, at data forbliver uændrede, autentiske og pålidelige.
- Det indebærer beskyttelse mod uautoriseret ændring, manipulation eller ødelæggelse af data.
- Metoder som datavalidering, hashfunktioner og digital signering bruges til at sikre integriteten af data.

# CIA-Triaden Tilgængelighed

- Tilgængelighed handler om at sikre, at autoriserede brugere har rettidig adgang til de nødvendige ressourcer og informationer, når de har brug for dem.
- Det indebærer at forhindre nedbrud, afbrydelser eller utilgængelighed af systemer og data på grund af fejl, angreb eller tekniske problemer.
- Redundans, backup, fejlfinding og kapacitetsstyring er nogle af de tilgængelighedsrelaterede metoder.

# Kryptering

- Kryptering er processen med at omdanne almindelige tekst eller data til en uforståelig form ved hjælp af en algoritme og en nøgle.
- Det formål, det tjener, er at beskytte fortroligheden og integriteten af information, så kun autoriserede parter kan læse og forstå dataene.
- Kryptering spiller en afgørende rolle i informationssikkerhed og bruges til at sikre kommunikation, beskytte følsomme oplysninger og verificere dataintegritet.

# Cæsarkryptering

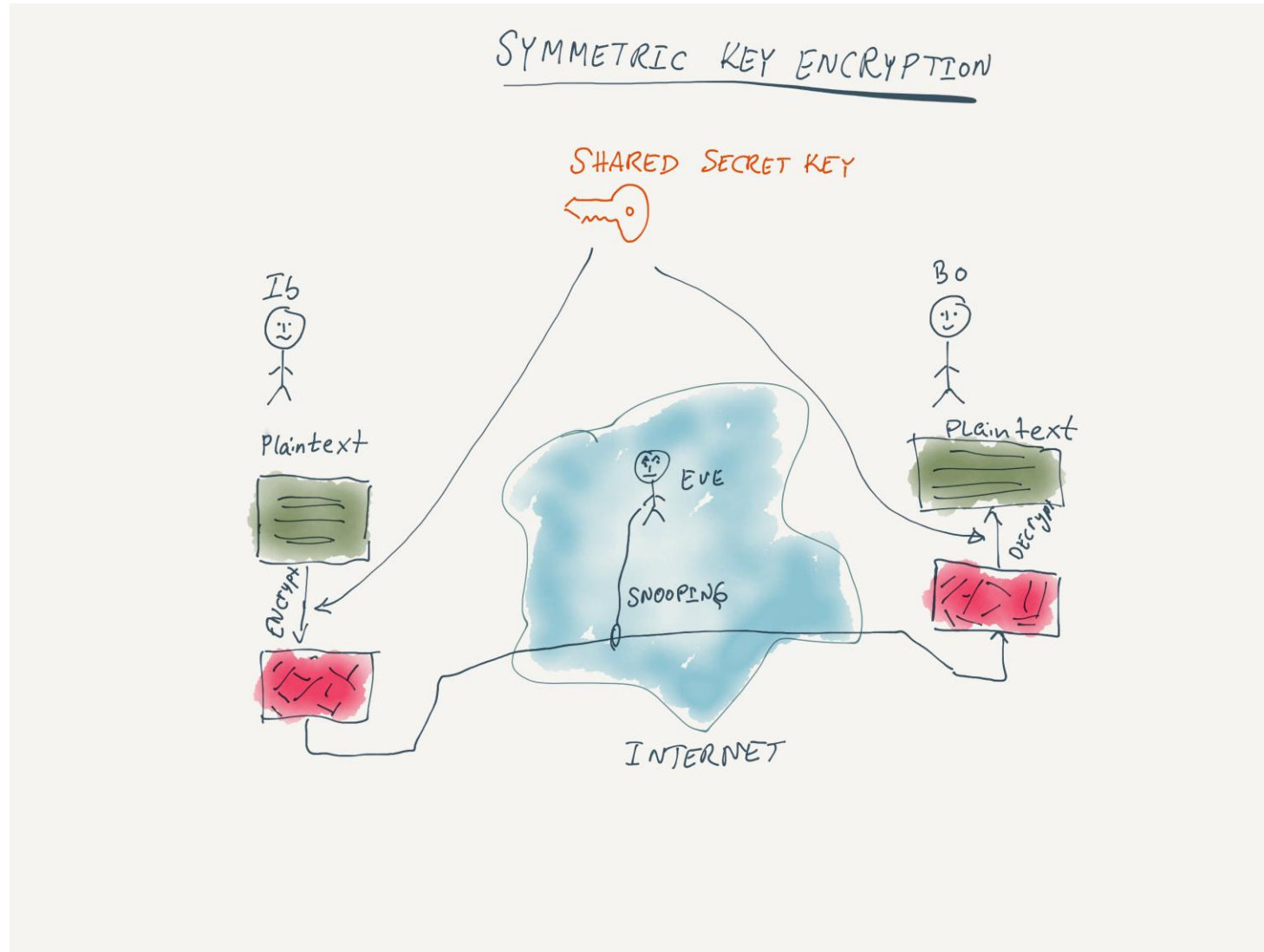
- Cæsarkryptering er en simpel form for substitution kryptering, hvor hvert bogstav i en besked erstattes med et bogstav, der er en fast skiftning længere nede i alfabetet.
- Det er opkaldt efter Julius Cæsar og blev brugt til at sende hemmelige beskeder i det gamle Rom. Cæsarkryptering er en let at bryde metode og anvendes sjældent i moderne kryptografi.
- Nøgle 3
- Plain text : Hej RN -> Cipher text: Khm#UQ



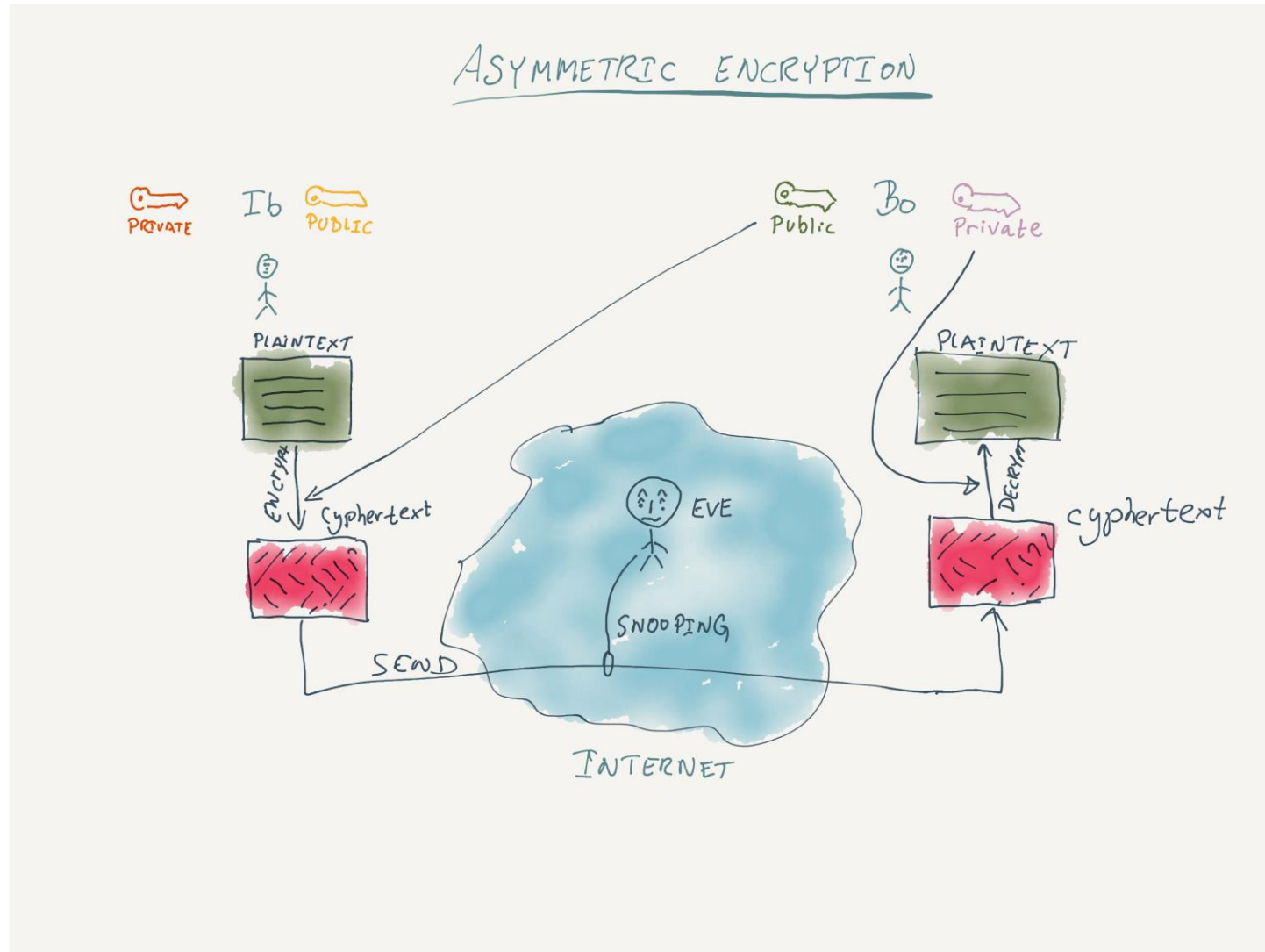
# Kryptering

- Hvad er problemet med Cæsar kryptering ?
- Hvordan kan den let knækkes ?
- Hvordan kan den gøres bedre ?

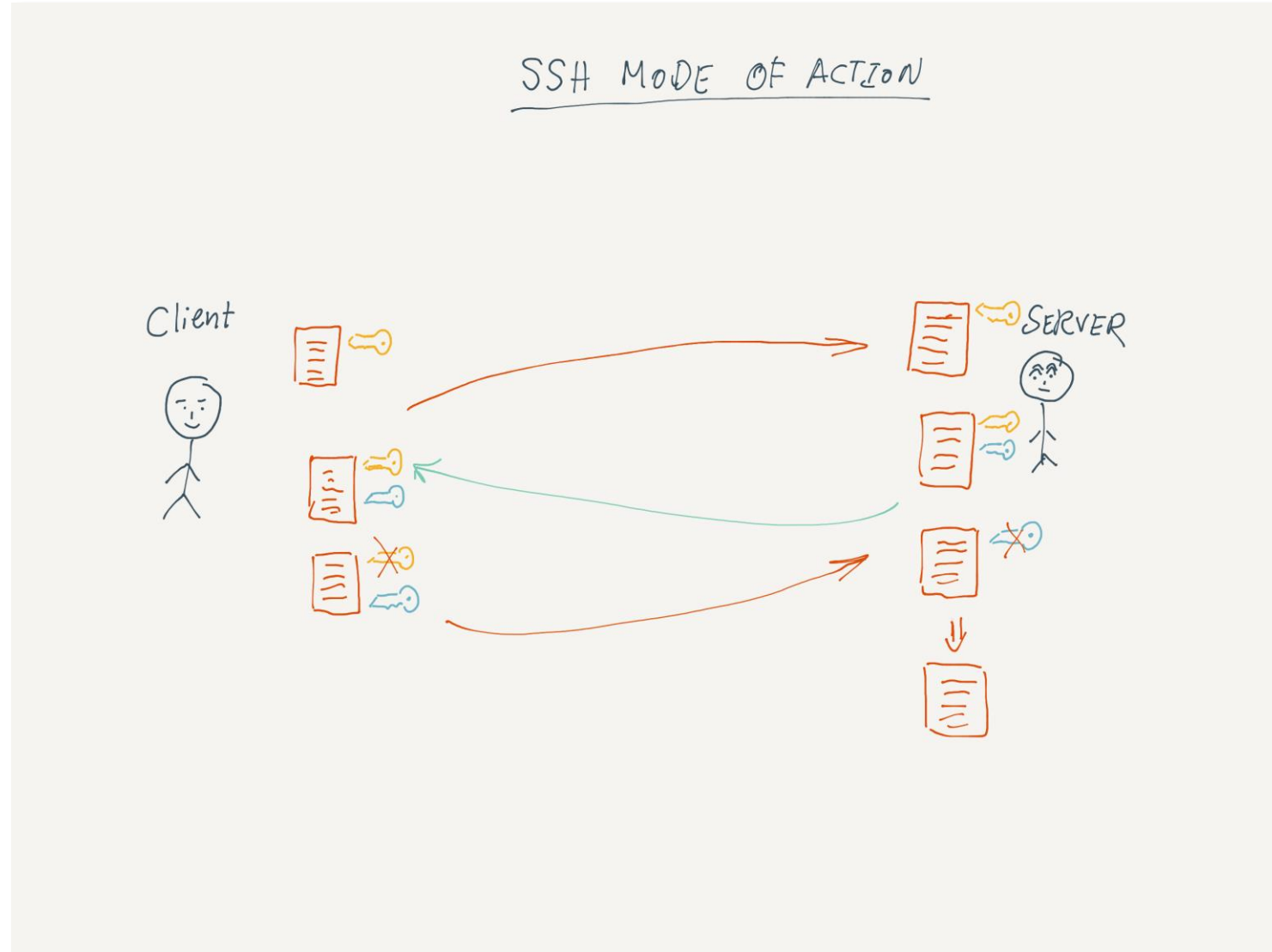
# Symmetrisk kryptering



# Asymmetrisk Kryptering



# Andre måder at udveksle data krypteret på

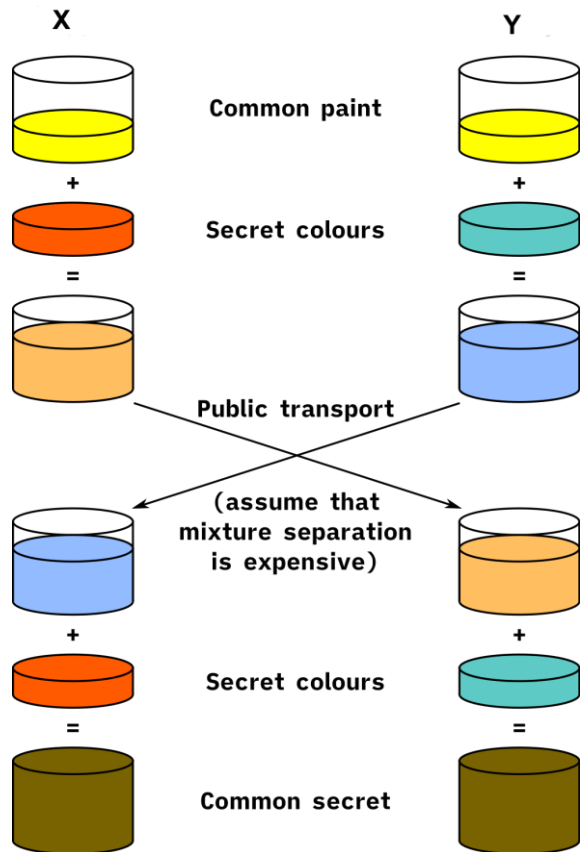


# Diffie-Hellman key exchange

- The Diffie-Hellman key exchange was the first widely used method of safely developing and exchanging keys over an insecure channel. It allows two parties who have not previously met to securely establish a key which they can use to secure their communications.
- Actually the key is not directly exchange but information that makes it possible to construct the same key by the two parts are exchange.
- Diffie-Hellman key exchange, also called exponential key exchange.
- In 1976, Whitfield Diffie and Martin Hellman invented a way for people to encrypt data and send it over an open channel. The idea was based on a concept by Ralph Merkle.

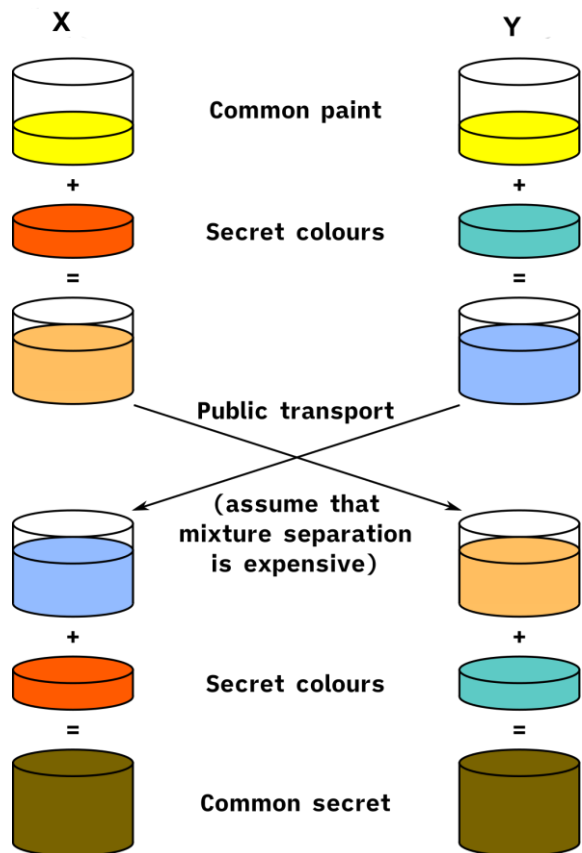


# Princip behind DH



- The process begins by having the two parties, Ib and Bo, publicly agree on an arbitrary starting color that does not need to be kept secret (but should be different every time). In this example, the color is yellow.
- Each person also selects a secret color that they keep to themselves – in this case, red and blue-green. The crucial part of the process is that Ib and Bo each mix their own secret color together with their mutually shared color, resulting in orange-tan and light-blue mixtures respectively, and then publicly exchange the two mixed colors.
- Finally, each of the two mixes the color they received from the partner with their own private color. The result is a final color mixture (yellow-brown in this case) that is identical to the partner's final color mixture.

# Princip behind DH



- If a third party listened to the exchange, it would only know the common color (yellow) and the first mixed colors (orange-tan and light-blue), but it would be difficult for this party to determine the final secret color (yellow-brown).
- Bringing the analogy back to a real-life exchange using large numbers rather than colors, this determination is computationally expensive.
- It is impossible to compute in a practical amount of time even for modern supercomputers.

# DH how to implement

- To implement Diffie-Hellman, the two end users Ib and Bo, mutually agree on positive whole numbers  $n$  and  $g$ , such that  $n$  is a prime number and  $g$  is a generator of  $n$ . The generator  $g$  is a number that, when raised to positive whole-number powers less than  $n$ , never produces the same result for any two such whole numbers. This webpage (<http://www.bluetulip.org/2014/programs/primitive.html>) can be used for finding the numbers. The value of  $n$  may be large but the value of  $g$  is usually small.
- Once Ib and Bo have agreed on  $n$  and  $g$ , they choose positive whole-number personal keys  $a$  and  $b$ , both less than the prime-number modulus  $n$ .



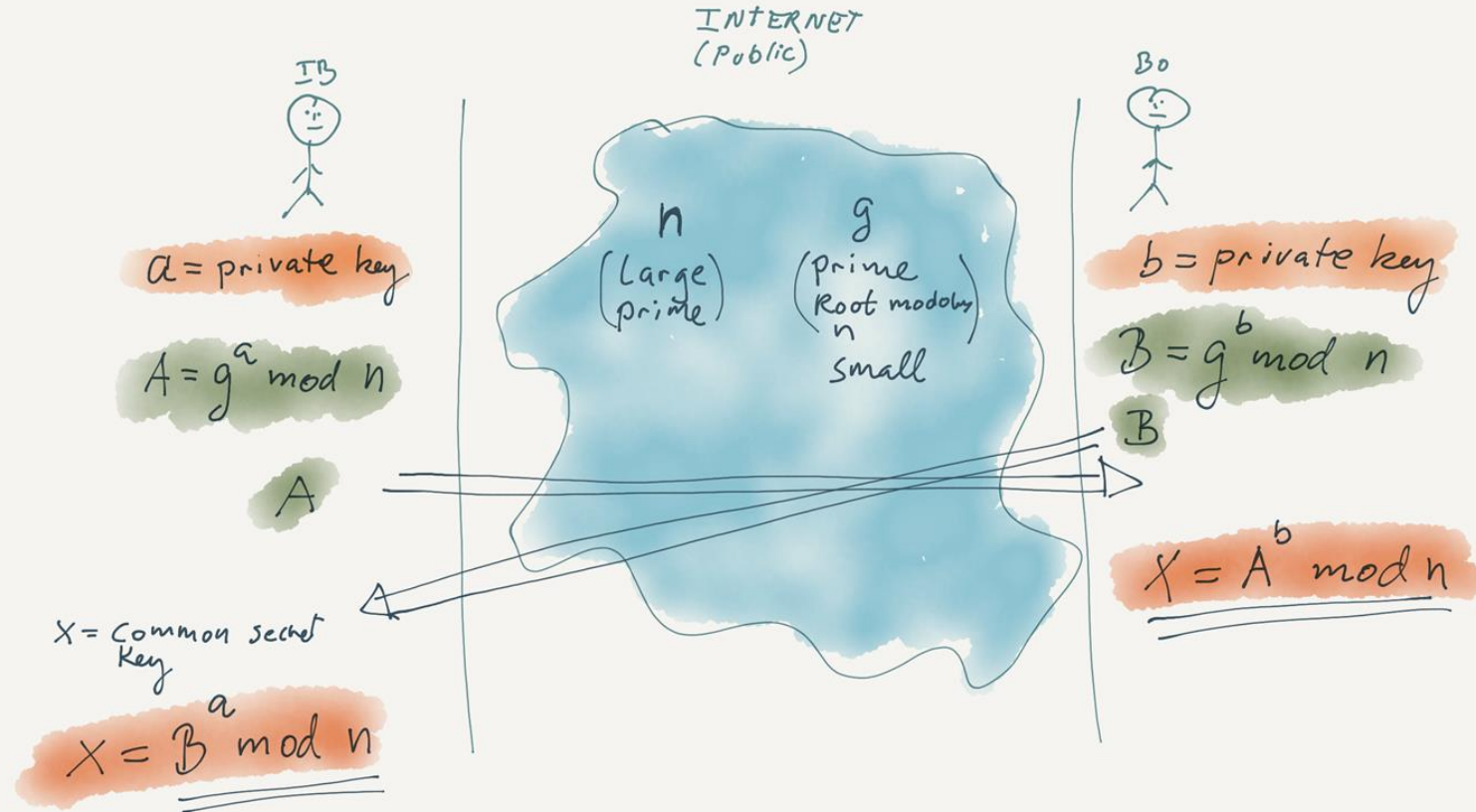
# DH how to implement

- Neither user divulges their personal key to anyone; ideally they memorize these numbers and do not write them down or store them anywhere. Next, Ib and Bo compute public keys A and B based on their personal keys according to the formulas:
- $A = g^a \bmod n$  (Ib's formula)
- $B = g^b \bmod n$  (Bo's formula)
- $a$  = Ib's personal keys
- $b$  = Bo's personal keys

# DH how to implement

- From the public keys, the secret number  $X$  can be generated by either user on the basis of their own personal keys  $a$  and  $b$ . Ib's computes  $x$  using the formula:
- $X = B^a \bmod n$
- Bo's computes  $X$  using the formula:
- $X = A^b \bmod n$

# Diffie - Hellman



## DH - EXAMPLE

IB 

$$a = 4$$

$$A = 3^4 \bmod 31 =$$
$$81 \bmod 31 = \textcircled{19}$$

$$X = 16^4 \bmod 31 =$$
$$65536 \bmod 31 =$$
$$\underline{\underline{2}}$$

Public

$$g = 3$$

$$n = 31$$

Bo 

$$b = 6$$

$$B = 3^6 \bmod 31$$

$$729 \bmod 31 = \textcircled{16}$$

$$X = 19^6 \bmod 31 =$$
$$47095881 \bmod 31 =$$
$$\underline{\underline{2}}$$

# DH and it's limitation

- The most serious limitation of Diffie-Hellman in its basic or "pure" form is the lack of authentication.
- Communications using Diffie-Hellman all by itself are vulnerable to man in the middle attacks. Ideally, Diffie-Hellman should be used in conjunction with a recognized authentication method such as digital signatures to verify the identities of the users over the public communications medium.
- Diffie-Hellman is well suited for use in data communication but is less often used for data stored or archived over long periods of time.

# SSL and TLS Encryption

# How Does SSL/TLS Encryption Work?

- SSL (Secure Sockets Layer) encryption, and its more modern and secure replacement, TLS (Transport Layer Security) encryption, protect data sent over the internet or a computer network.
- This prevents attackers (and Internet Service Providers) from viewing or tampering with data exchanged between two nodes—typically a user's web browser and a web/app server.
- Most website owners and operators have an obligation to implement SSL/TLS to protect the exchange of sensitive data such as passwords, payment information, and other personal information considered private.

# How Does SSL/TLS Encryption Work?

- SSL/TLS uses both asymmetric and symmetric encryption to protect the confidentiality and integrity of data-in-transit.
- Asymmetric encryption is used to establish a secure session between a client and a server, and symmetric encryption is used to exchange data within the secured session.
- A website must have an SSL/TLS certificate for their web server/domain name to use SSL/TLS encryption.
- Once installed, the certificate enables the client and server to securely negotiate the level of encryption.

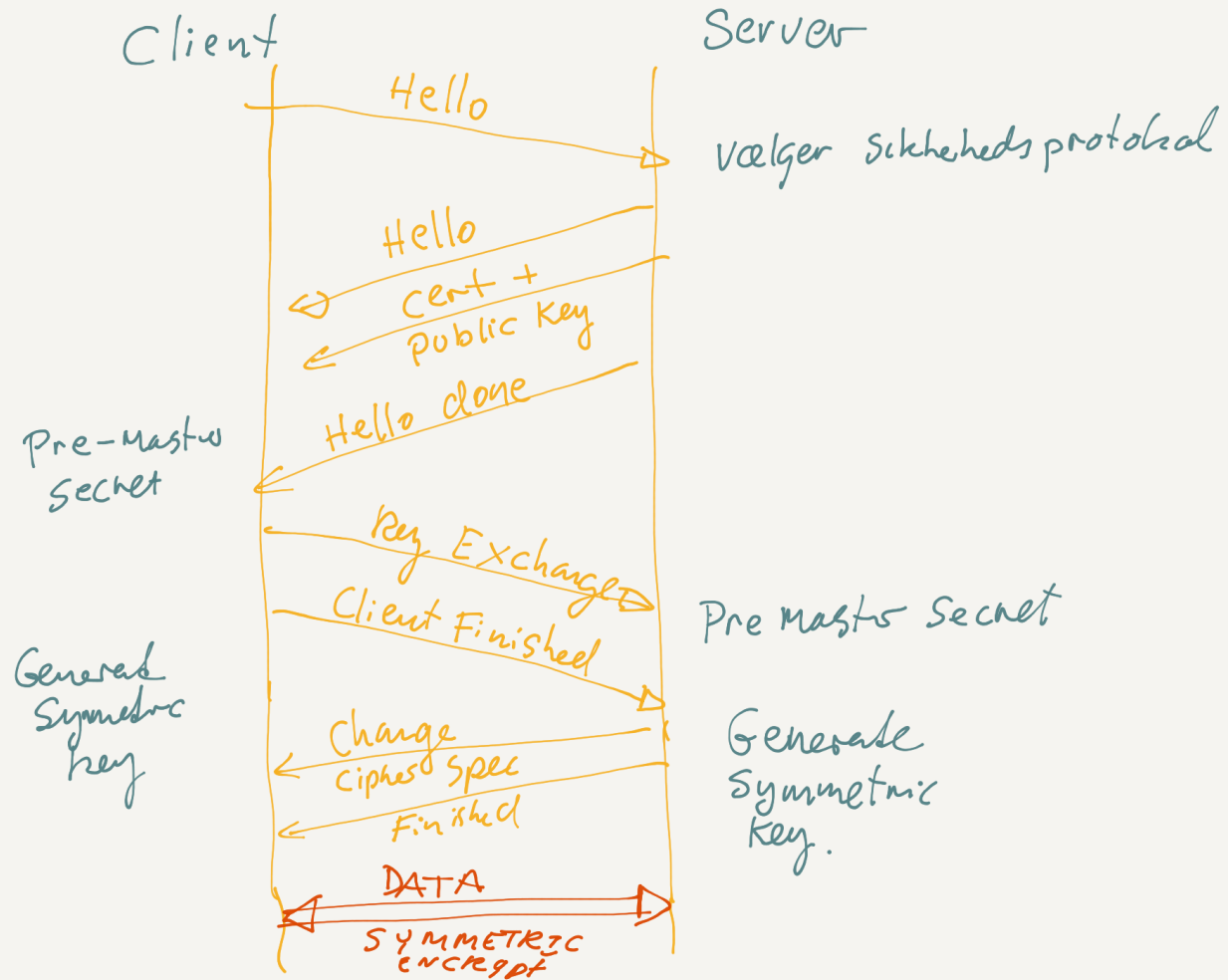


# How Does SSL/TLS Encryption Work, step by step

- 1) The client contacts the server using a secure URL (HTTPS...).
- 2) The server sends the client its certificate and public key.
- 3) The client verifies this with a Trusted Root Certification Authority to ensure the certificate is legitimate.
- 4) The client and server negotiate the strongest type of encryption that each can support.
- 5) The client encrypts a session (secret) key with the server's public key, and sends it back to the server.
- 6) The server decrypts the client communication with its private key, and the session is established.
- 7) The session key (symmetric encryption) is now used to encrypt and decrypt data transmitted between the client and server.

Both the client and server are now using HTTPS (SSL/TLS + HTTP) for their communication. Web browsers validate this with a lock icon in the browser address bar. HTTPS functions over Port 443.

# TLS - Handshake



# Hvad er et certifikat egentligt ?

- Et certifikat er et digitalt dokument, der fungerer som en elektronisk identitetsattest. Det bekræfter identiteten af en enhed, enten en person, en organisation, en computer eller en webside, i en digital kommunikation. Certifikater anvender kryptografiske metoder og er udstedt af betroede enheder eller certifikatautoriteter (CA).
- Et certifikat indeholder flere vigtige elementer. Først er der identitetsoplysninger, der identificerer den enhed, certifikatet er udstedt til. Dette kan omfatte navn, organisation, e-mailadresse eller anden identifikator.
- Derudover indeholder certifikatet en offentlig nøgle, som er en del af et nøglepar, der består af en offentlig og en privat nøgle. Den offentlige nøgle bruges til kryptografiske operationer som kryptering, digital signering eller nøgleaftale. Den offentlige nøgle er knyttet til den enheds private nøgle og kan bruges til at verificere digital signatur og sikre kommunikation.

# Hvad er et certifikat egentligt ?

- Certifikatet indeholder også oplysninger om udstederen, den enhed eller organisation, der udsteder certifikatet. Disse oplysninger kan omfatte navn, offentlig nøgle og eventuelle tilladelser eller certificeringsniveauer.
- Endelig har certifikatet en udløbsdato, der angiver, hvornår certifikatet ikke længere er gyldigt. Dette er vigtigt, da det sikrer, at certifikatet regelmæssigt bliver fornyet for at opretholde sikkerheden og tilliden.
- Certifikater anvendes bredt inden for digital kommunikation og internetbaserede tjenester. For eksempel bruges SSL-/TLS-certifikater til at etablere en sikker forbindelse mellem en webbrowser og en webside. Certifikater spiller en afgørende rolle i at etablere fortrolighed, integritet og autentifikation i elektroniske kommunikationssystemer og bidrager til at opretholde sikkerhed og tillid.

# Authentication

- Hvordan ved jeg at jeg faktisk kommunikere med F.eks Danske Bank ?

# Et skridt videre - Multifactor authentication (MFA)

- Multifactor authentication (MFA) er en sikkerhedsforanstaltning, der kræver brugen af to eller flere forskellige metoder til at bekræfte en persons identitet.
- Traditionel autentifikation, der normalt er baseret på brugernavn og adgangskode, kan være sårbart over for phishing, hacking eller tyveri af legitimationsoplysninger.
- MFA øger sikkerheden ved at tilføje ekstra lag af autentifikation.

# Et skridt videre - Multifactor authentication (MFA)

- Typisk involverer MFA følgende faktorer:
  - **Noget, du ved:** Dette er normalt noget, du kender, såsom en adgangskode, en pinkode eller svar på sikkerhedsspørgsmål.
  - **Noget, du har:** Dette er noget fysisk, som du besidder, f.eks. en smartphone, en nøgleviser eller et smartkort.
  - **Noget, du er:** Dette er en biometrisk faktor, der bruger dine unikke fysiske egenskaber, såsom fingeraftryk, ansigtsgenkendelse eller stemmeidentifikation.

# Et skridt videre - Multifactor authentication (MFA)

- For at logge ind eller få adgang til en tjeneste med MFA skal brugeren levere flere autentifikationsfaktorer. For eksempel kan en bruger blive bedt om at indtaste sit brugernavn og adgangskode (noget, de ved), og derefter modtage en engangskode via en autentificeringsapp på deres smartphone (noget, de har), som skal indtastes for at fuldføre autentifikationen.
- MFA forhindrer effektivt uautoriseret adgang, da selv hvis en faktor bliver kompromitteret, vil angriberen stadig skulle overvinde yderligere faktorer for at få adgang. Det øger beskyttelsen af brugerkonti og følsomme data og er blevet mere udbredt som en best practice for at styrke sikkerheden i digitale miljøer.



# Why Directory services?

Functions and Benefits of Active Directory  
Components of Active Directory

# AD is a Directory Service

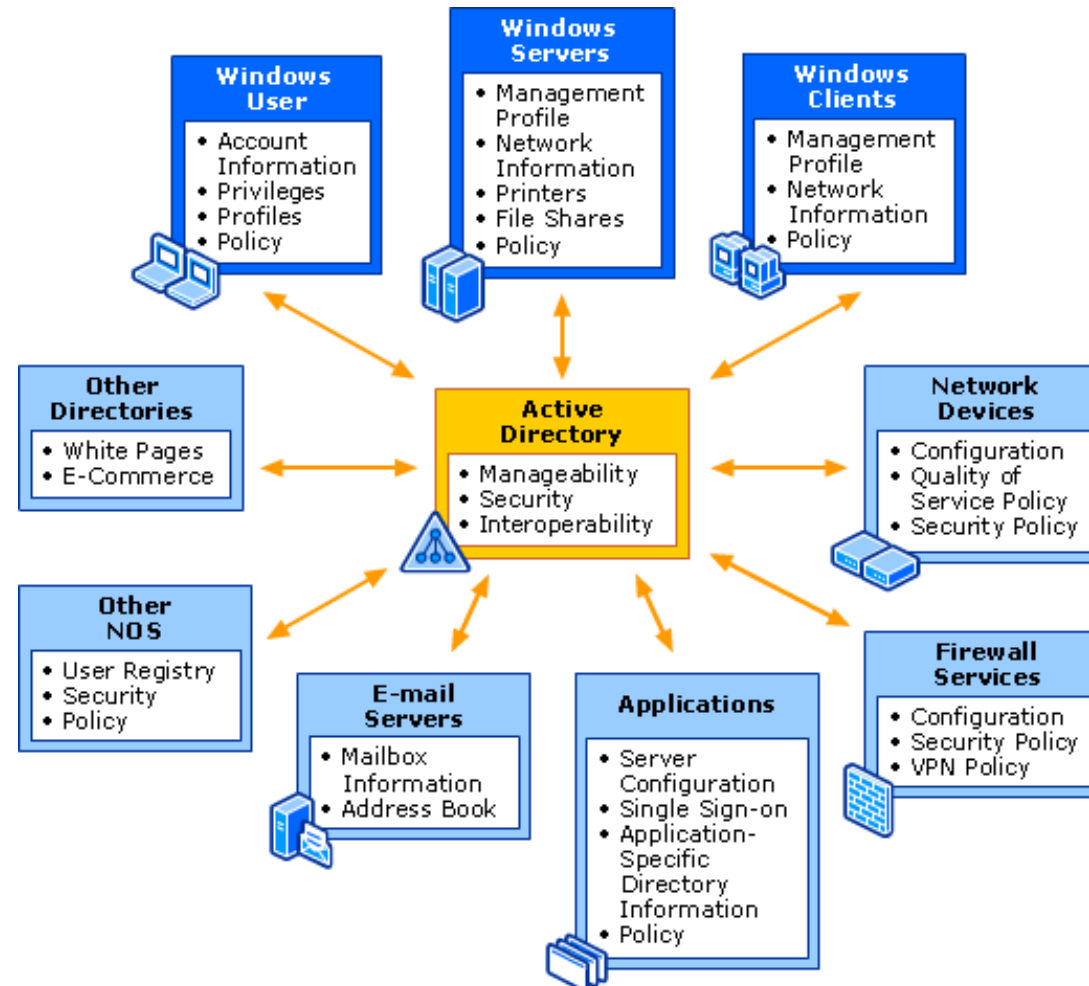
A directory service is:

A service used by business to define, manage, access and secure network resources, including files, printers, people and applications for a group of users

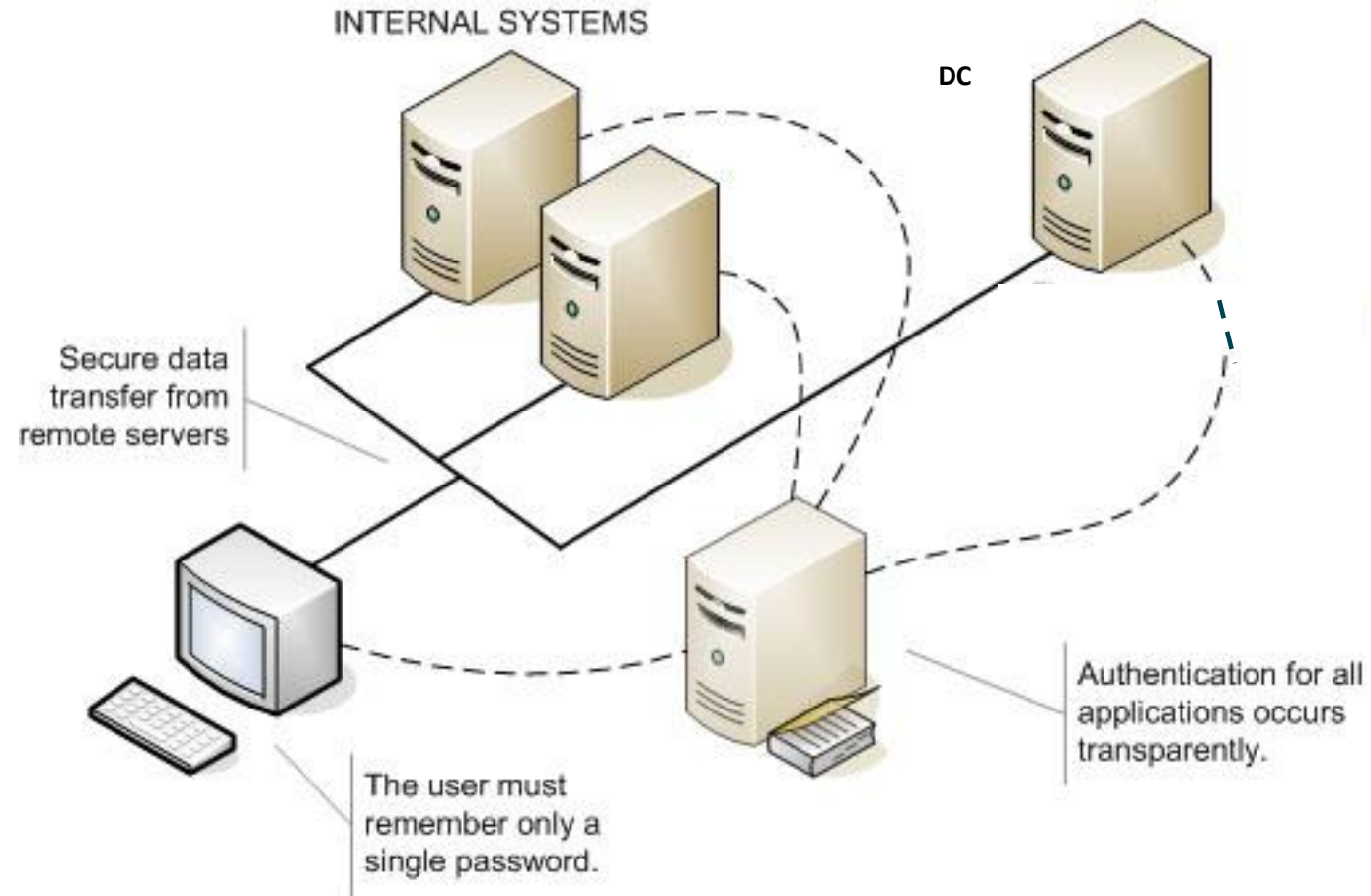
# AD Benefits

- Centralized Resource and Security Administration
- Single logon for access to network resources
- Fault tolerance and redundancy
- Simplified resource location

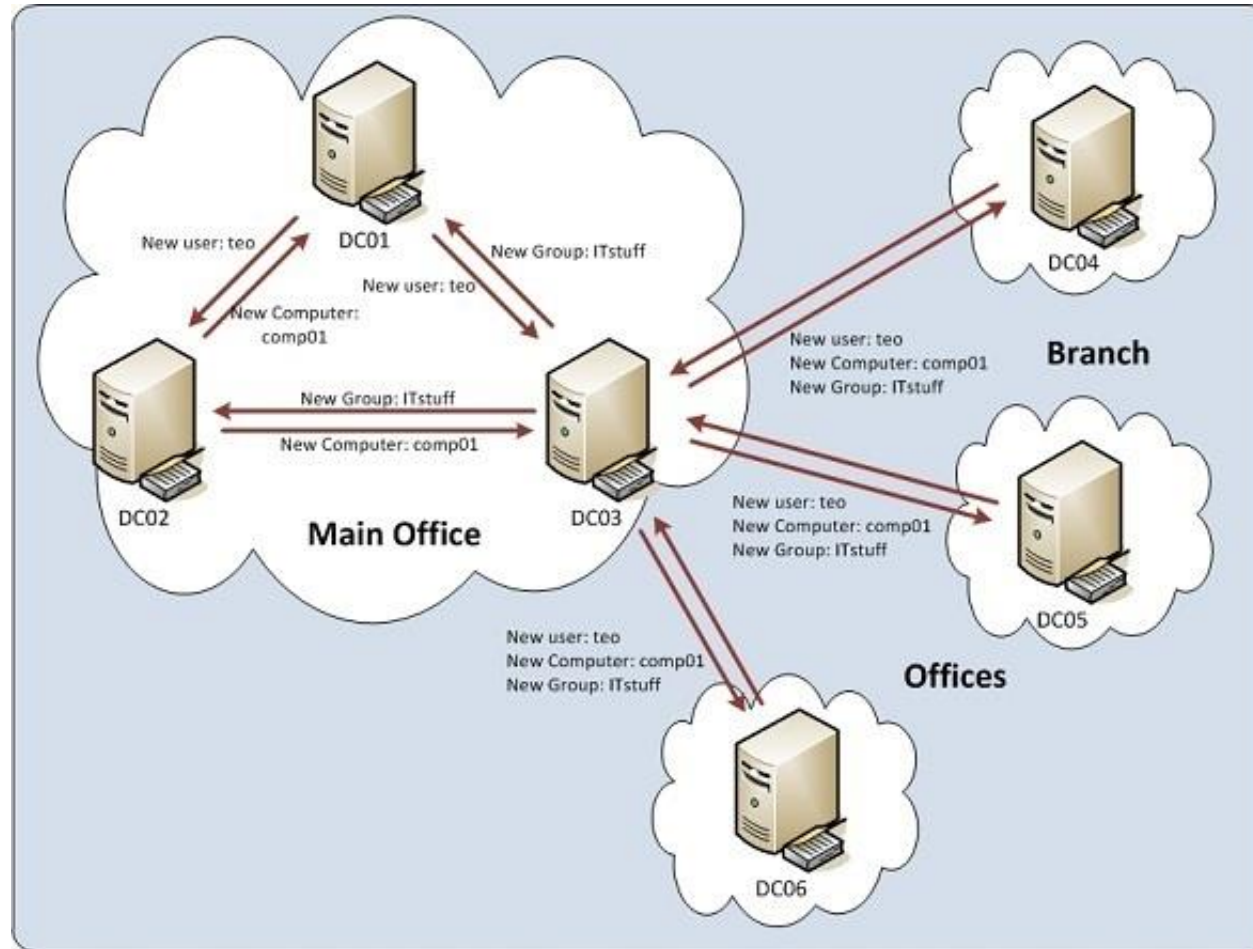
# Centralized Resource and Security Administration



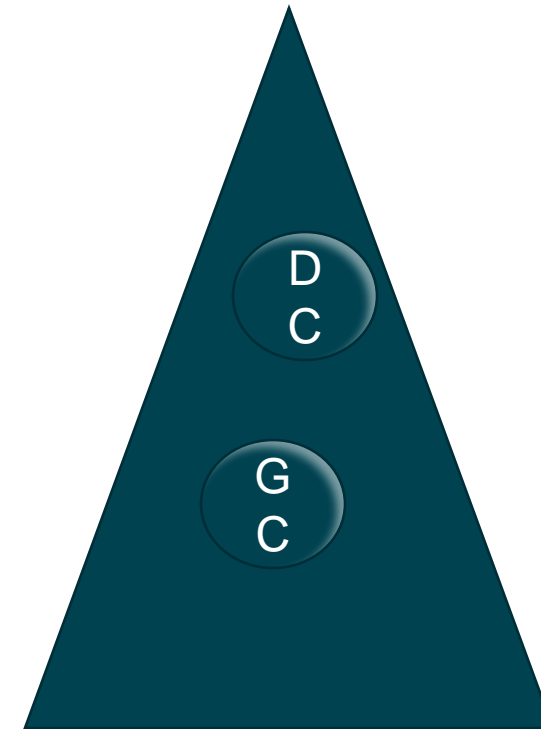
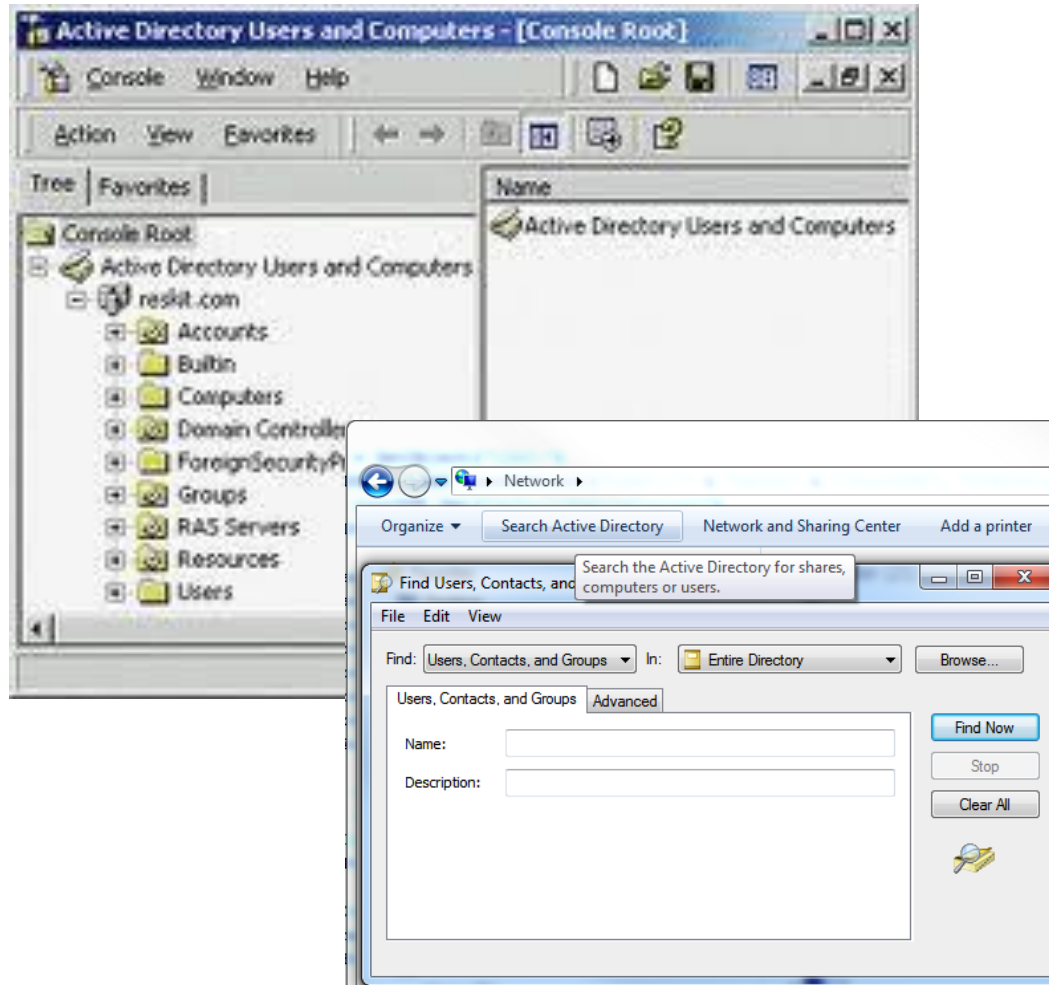
# Single Logon for Access to Network Resources



# Fault Tolerance and Redundancy



# Simplified Resource Location



# Active Directory DS

Introduktion til AD DS



# Active Directory Directory Services

- Microsoft Directory service
- Initially released in 1999
- Originally designed for Windows 2000 Server
  - Enhanced with Windows Server 2008
- Windows Server 2022 types
  - Workgroup model
  - Domain model

# Workgroups

- Peer-to-peer network
- Decentralized management
  - Each computer has own database
    - User accounts, security privileges
  - Significantly more administration effort
- Practical for small networks
  - Few users
  - Simple to design, implement

# Domains

- Client/server network with a shared database
- Domain - Group of users, servers, and other resources
  - Share centralized account and security information in a database
- Active Directory
  - Contains domain database with objects and attributes and schema
  - Makes it easier to organize and manage resources and security

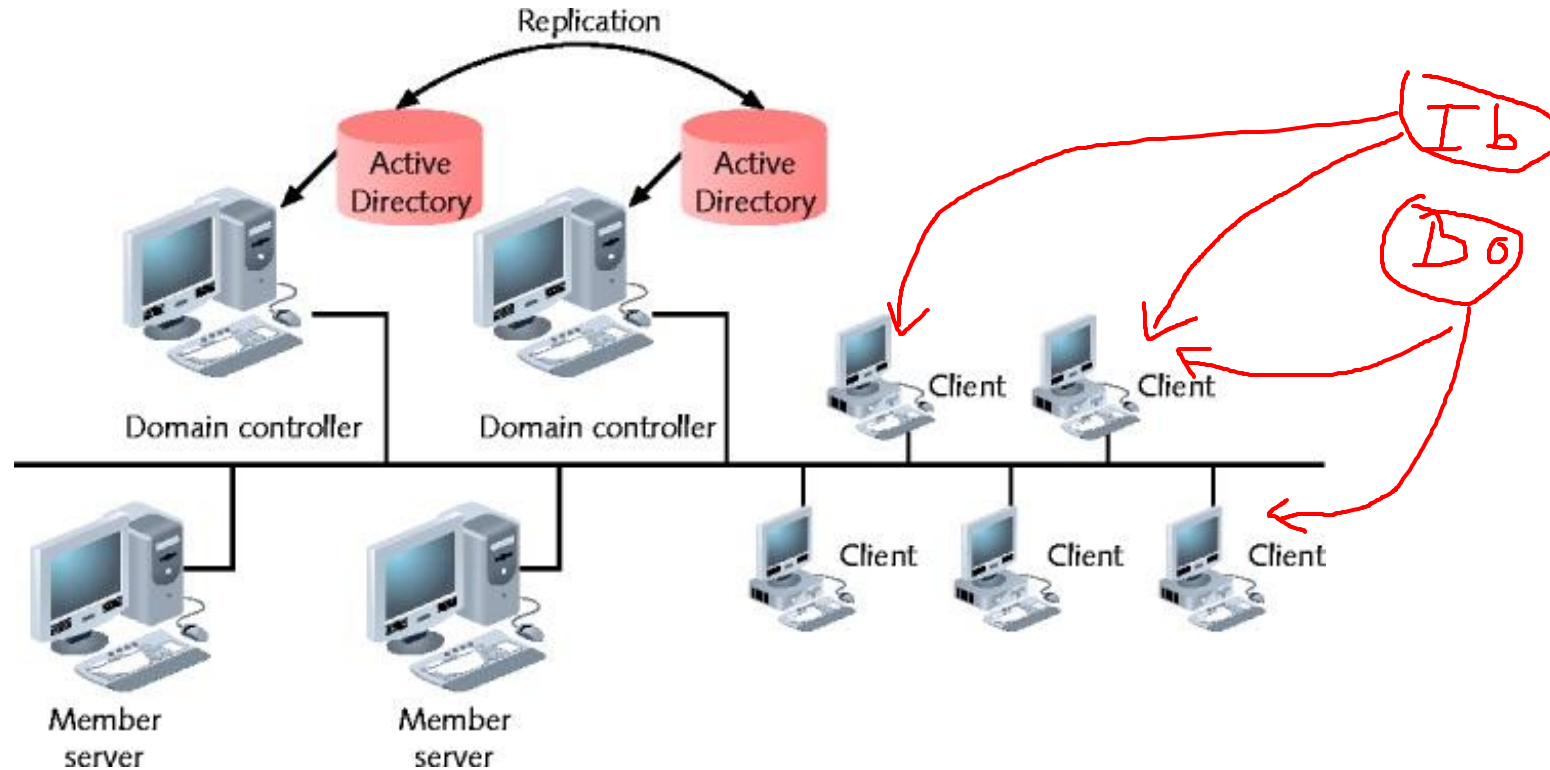
# Active Directory - Domains

- Domain not confined by geographical boundaries
- Domain controller servers
  - Contains directory information about objects in a domain
- Member servers
  - Do not store directory information, can't be used to authenticate users
- Replication
  - Process of copying directory data to multiple domain controllers

# Active Directory Directory Services– Domain Model

- Three main parts
  - Domain
  - Tree
  - Forest

# Domains



Domain model on a Windows Server 2022 network

# Active Directory

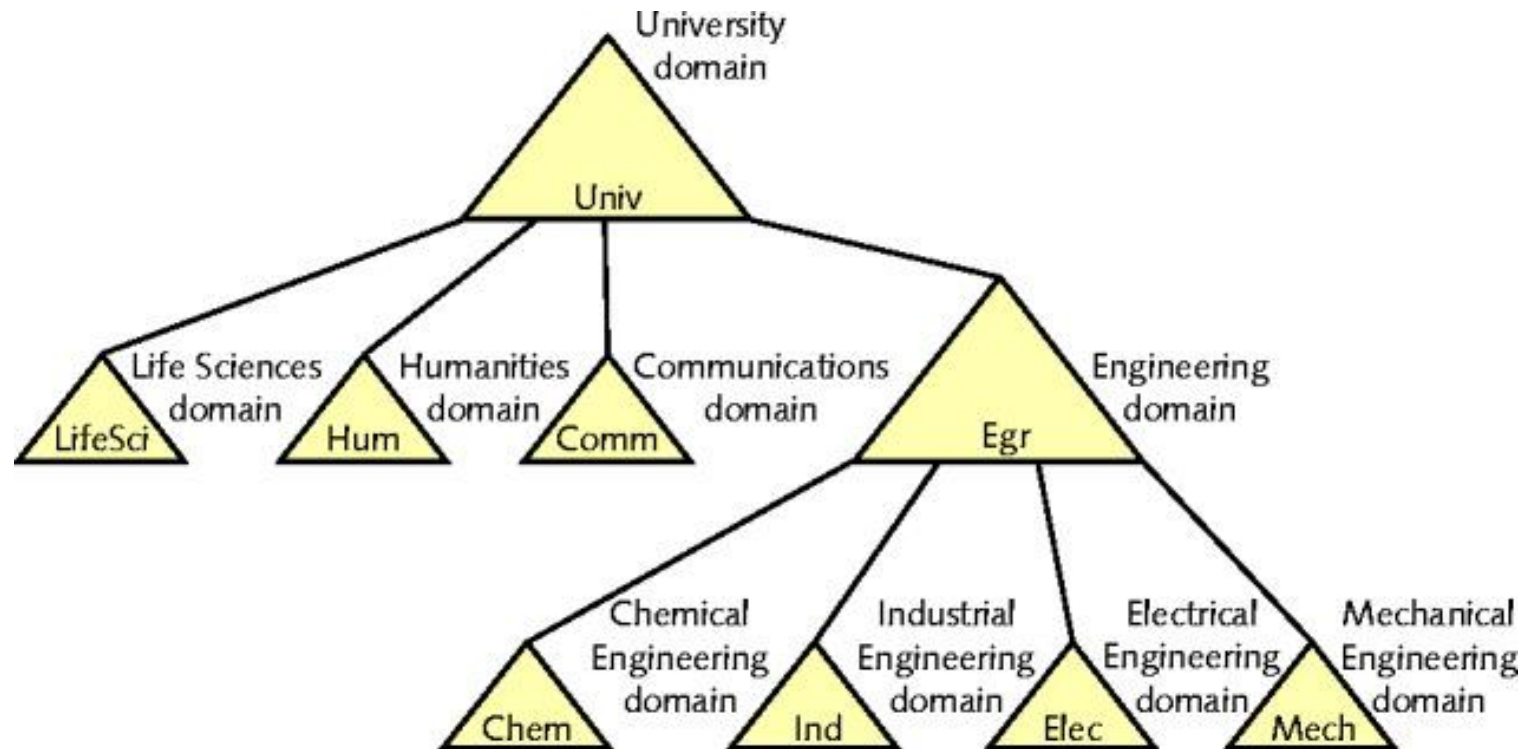
- Objects fall into two broad categories:
  - Resources (e.g., printers)
  - Security principals (user or computer accounts and groups).
    - Security principals are assigned unique security identifiers (SIDs)
    - This is where access rights are given
    - Users must have unique names – flat database

# OUs (Organizational Units)

- Hold multiple objects having similar characteristics
  - Can be nested
  - Can contain other OUs or objects
- Provides simpler, more flexible administration
  - Apply policies to OU
  - Do not function as containers
  - Use users or groups for access permissions

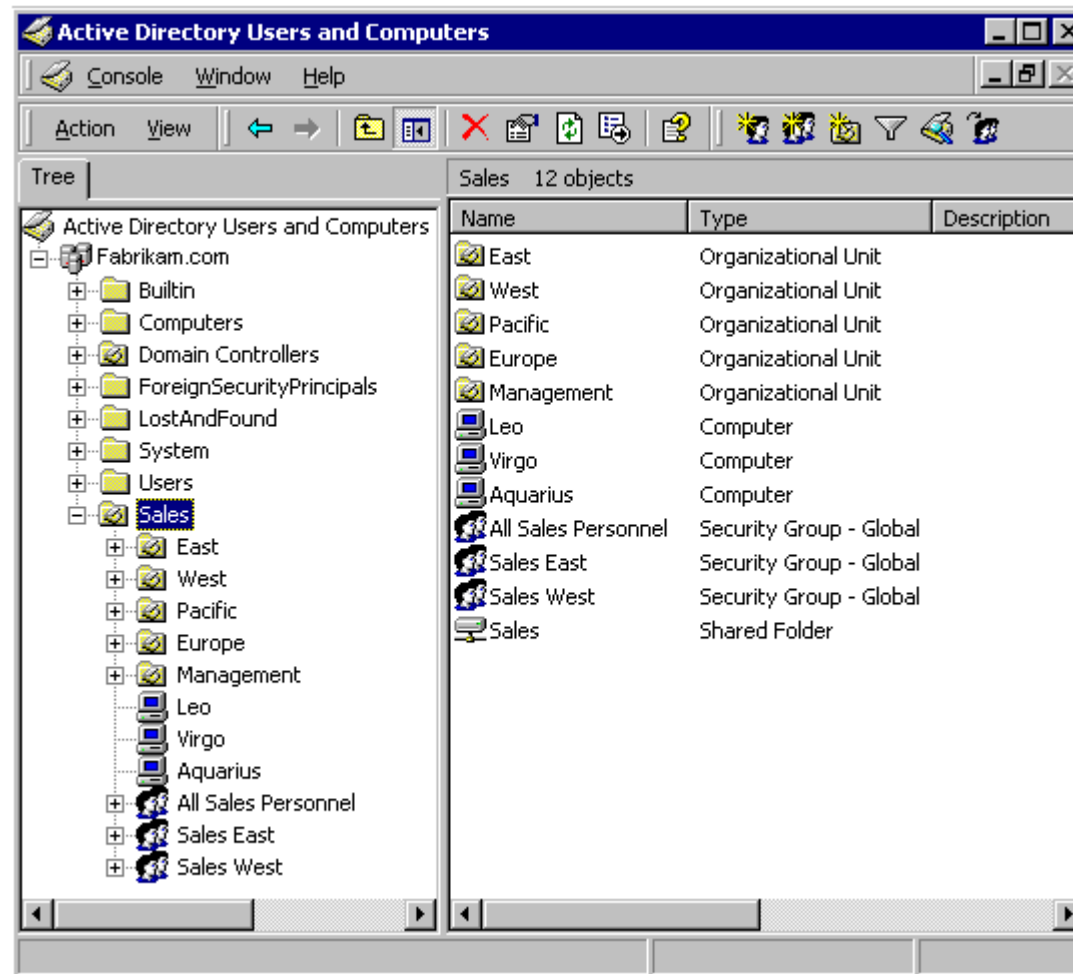


# Domains



Multiple domains in one organization

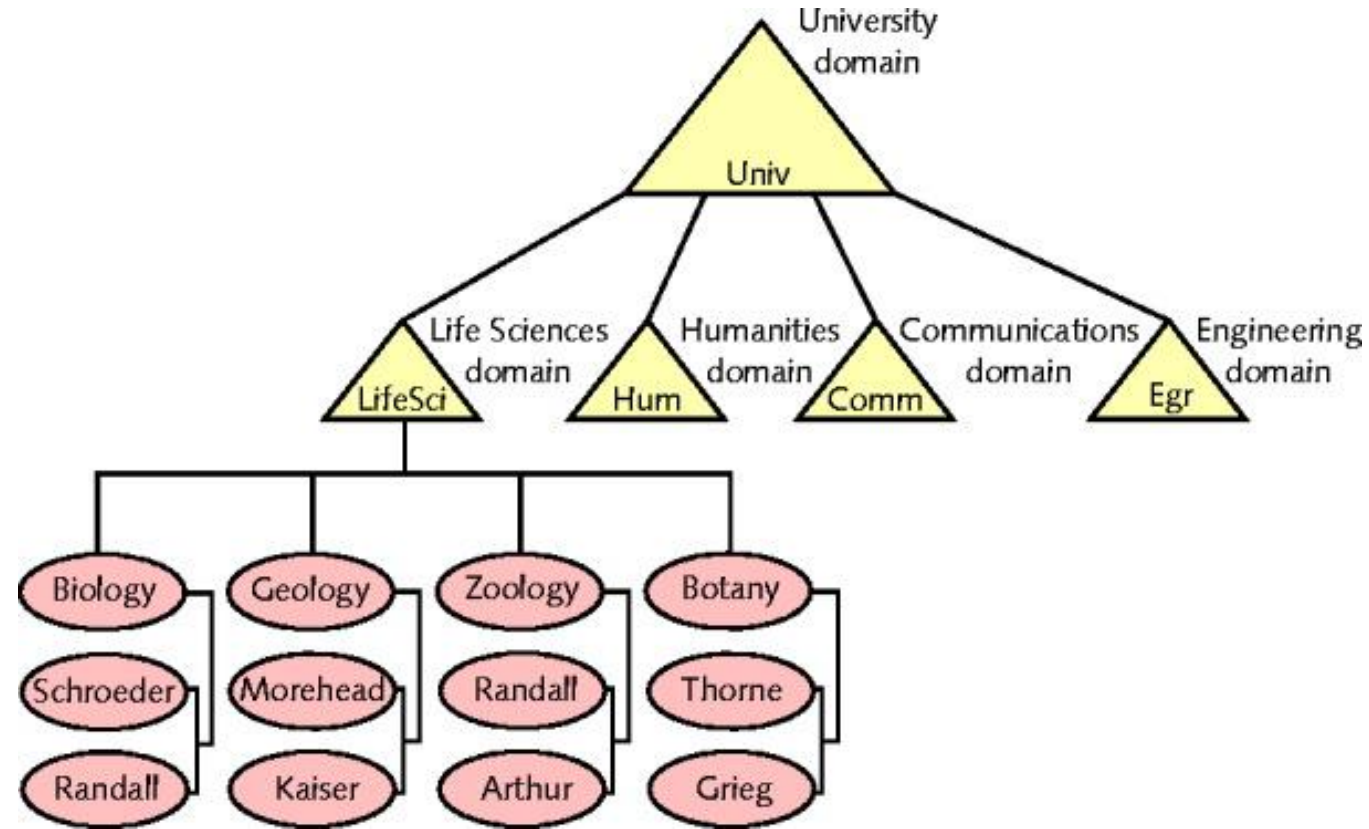
# Domains



# Trees and Forests

- Directory structure above domains
  - Large organizations use multiple domains
- Domain tree
  - Organizes multiple domains hierarchically
- Root domain
  - Active Directory tree base
- Child domains
  - Branch off from root domain

# Trees and Forests

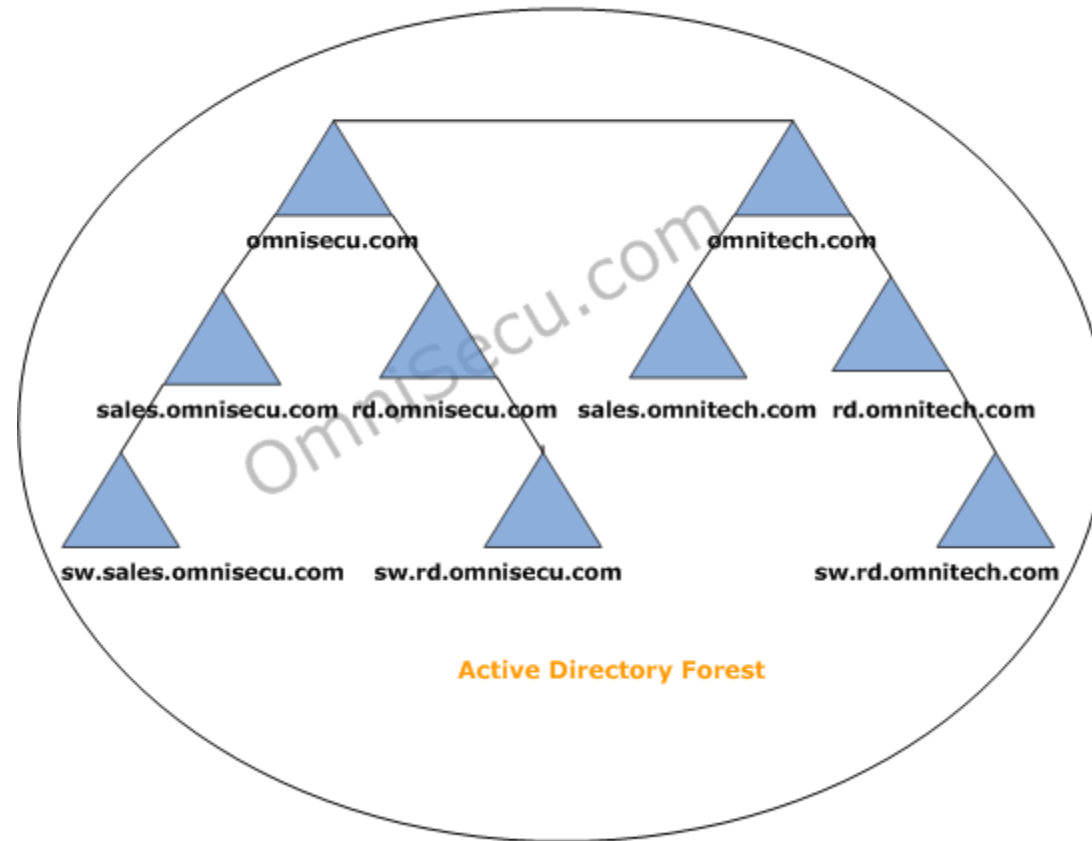


A tree with multiple domains and OUs

# Trees and Forests

- Forest
  - A collection of one or more domain trees
  - Trees share common schema
- Domains within a forest can communicate
- Domains within same tree
  - Share common Active Directory database

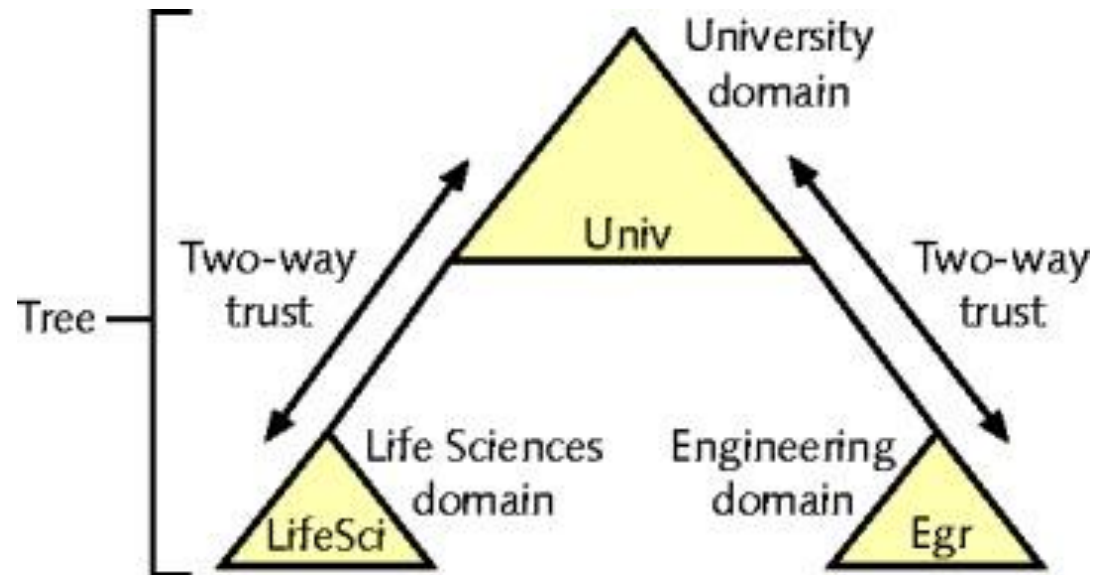
# Two Tree - Forest



# Trust Relationships

- Relationship between two domains
  - One domain allows another domain to authenticate its users
- Active Directory supports two trust relationship types – allows users to authenticate
  - Two-way transitive trusts
  - Explicit one-way trusts

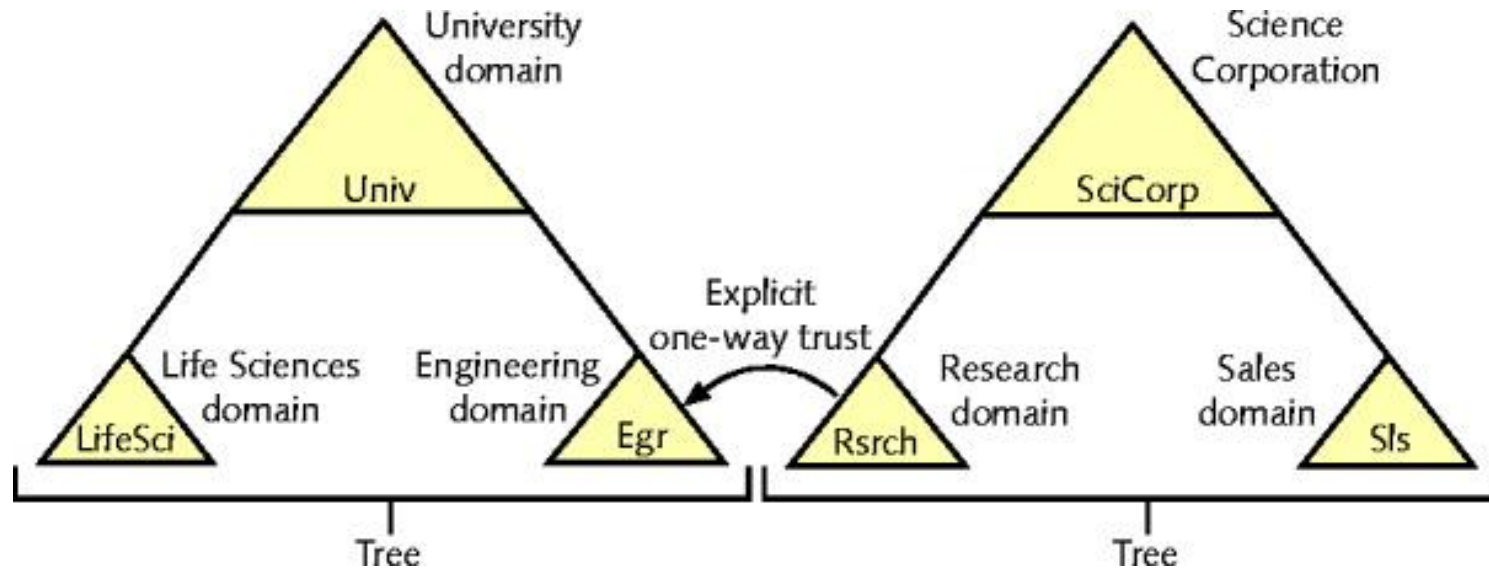
# Trust Relationships



Two-way trusts between domains in a tree

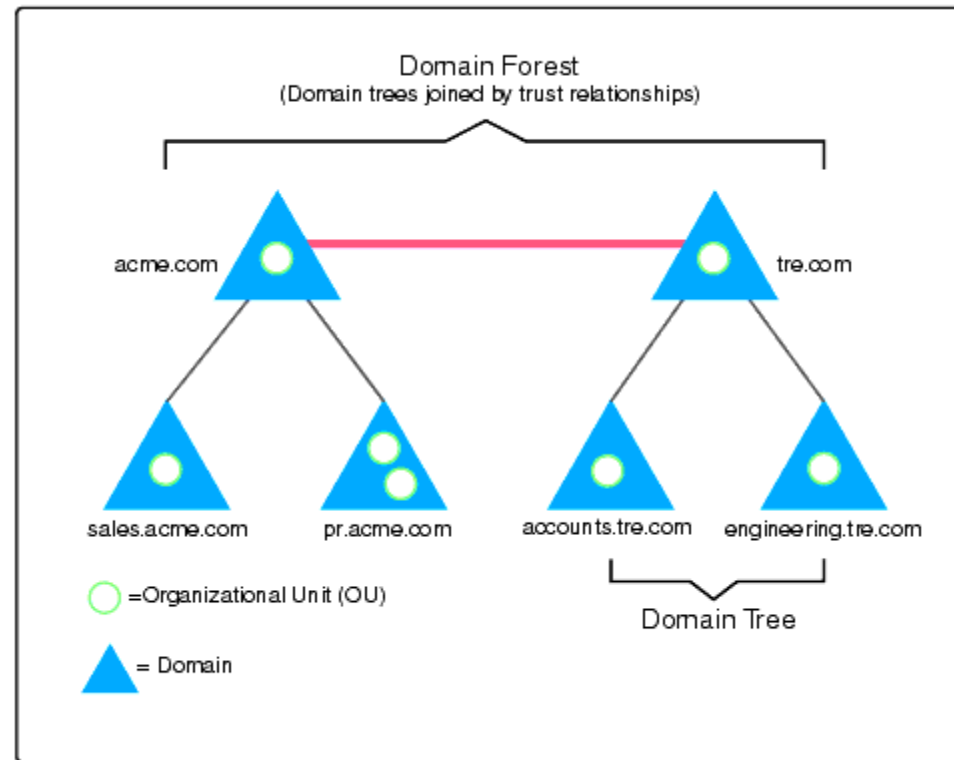


# Trust Relationships



Explicit one-way trust between domains in different trees

# Trust Relationships



# Naming Conventions

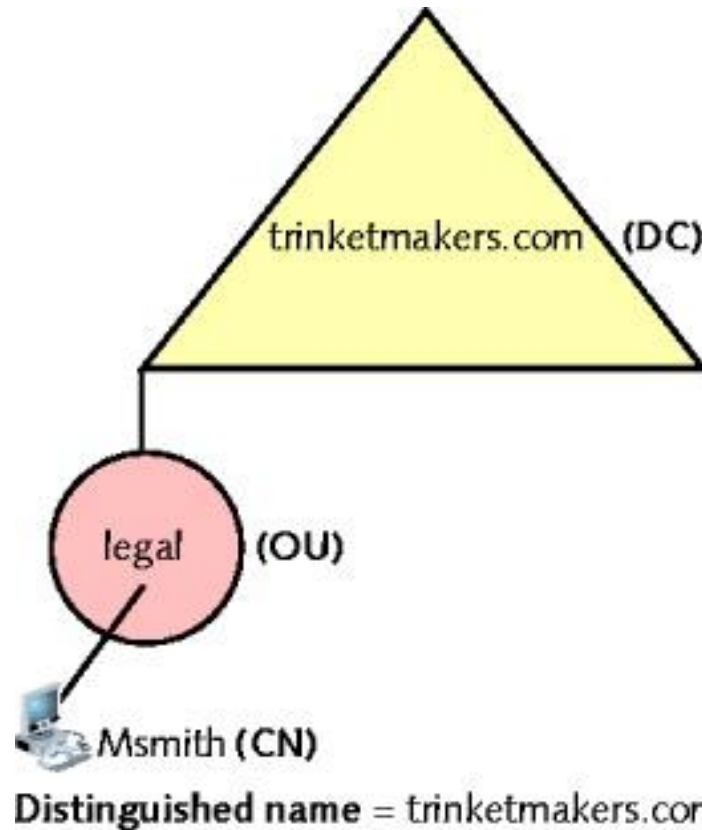
- Active Directory naming conventions (namespace)
  - Collection of object names and associated places in Windows Server 2016, Server 2019 network
  - Based on LDAP naming conventions
  - Follows the conventions of the internet namespace
    - Ex. dc=wright, dc=edu
    - Ex. cn=server1,dc=wright,dc=edu
    - Ex. cn=server2,ou=cse,dc=wright,dc=edu

# Naming Conventions

- Windows Server 2022 network object
  - Three different names
    - DN (distinguished name): DC (domain component) and CN (common name)
    - RDN (relative distinguished name)
    - UPN (user principal name)
- GUID (globally unique identifier)
  - Each object has one
  - 128-bit number

# Naming Conventions

upn = msmith@trinketmakers.com



DN:  
cn=msmith,ou=legal,dc=trinketmakers,  
dc=com

relative  
distinguished name

Distinguished name and relative distinguished name