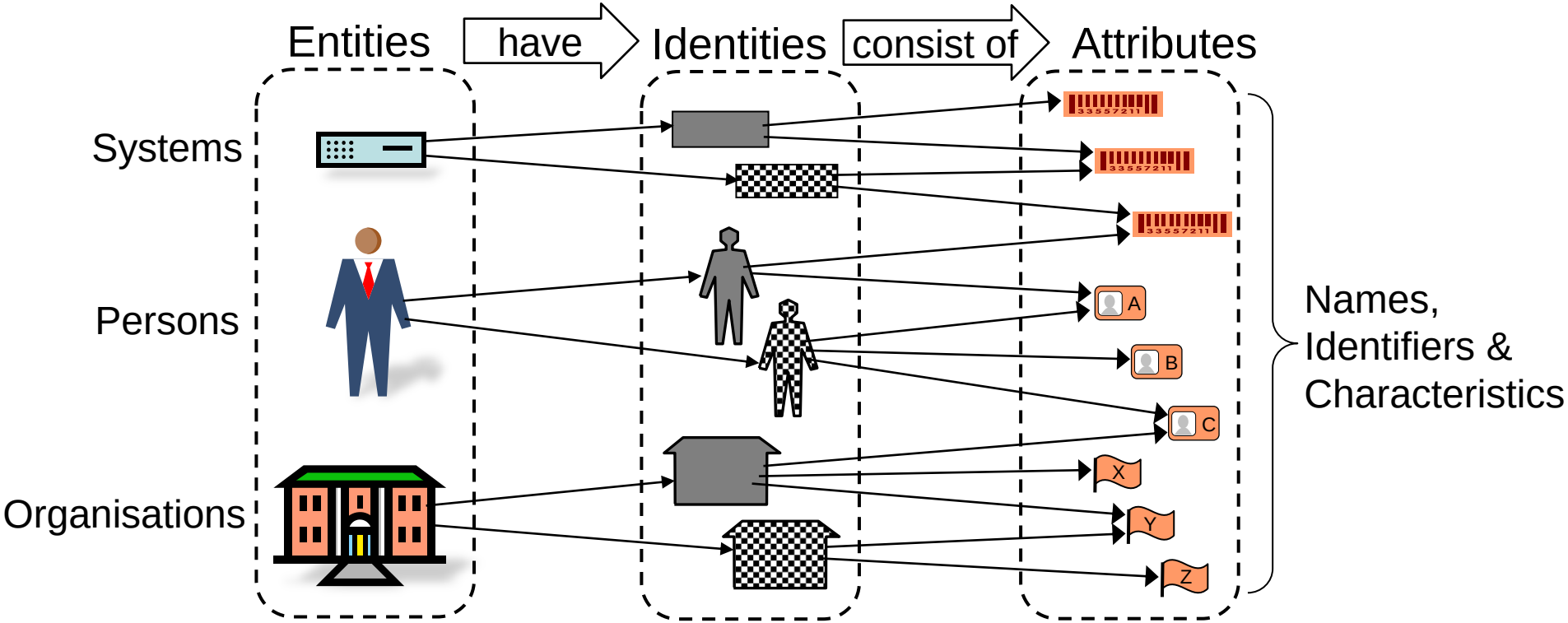
Identity Management

Prof Audun Jøsang
Department of Informatics
University of Oslo

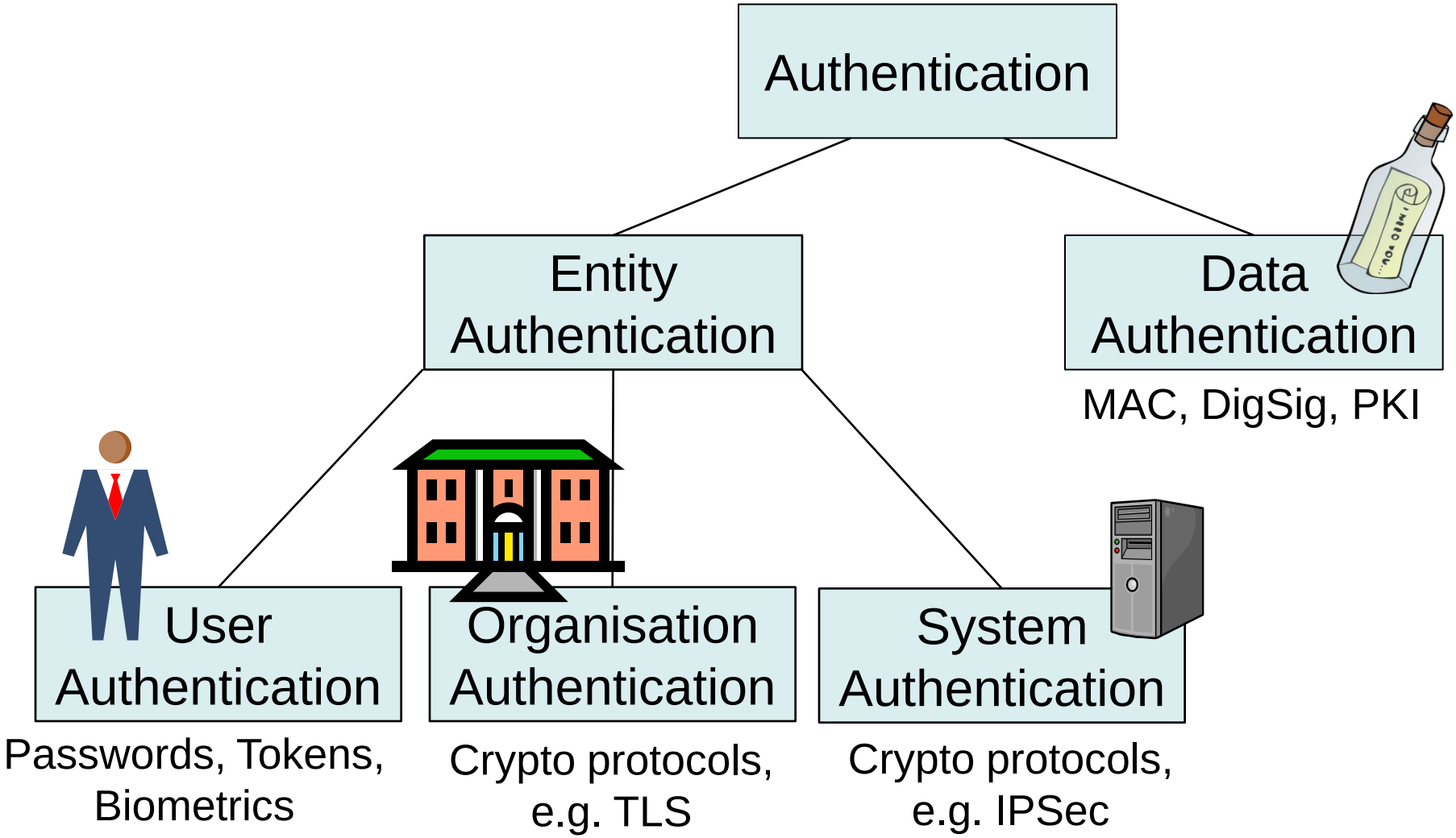
Finse

May 2014

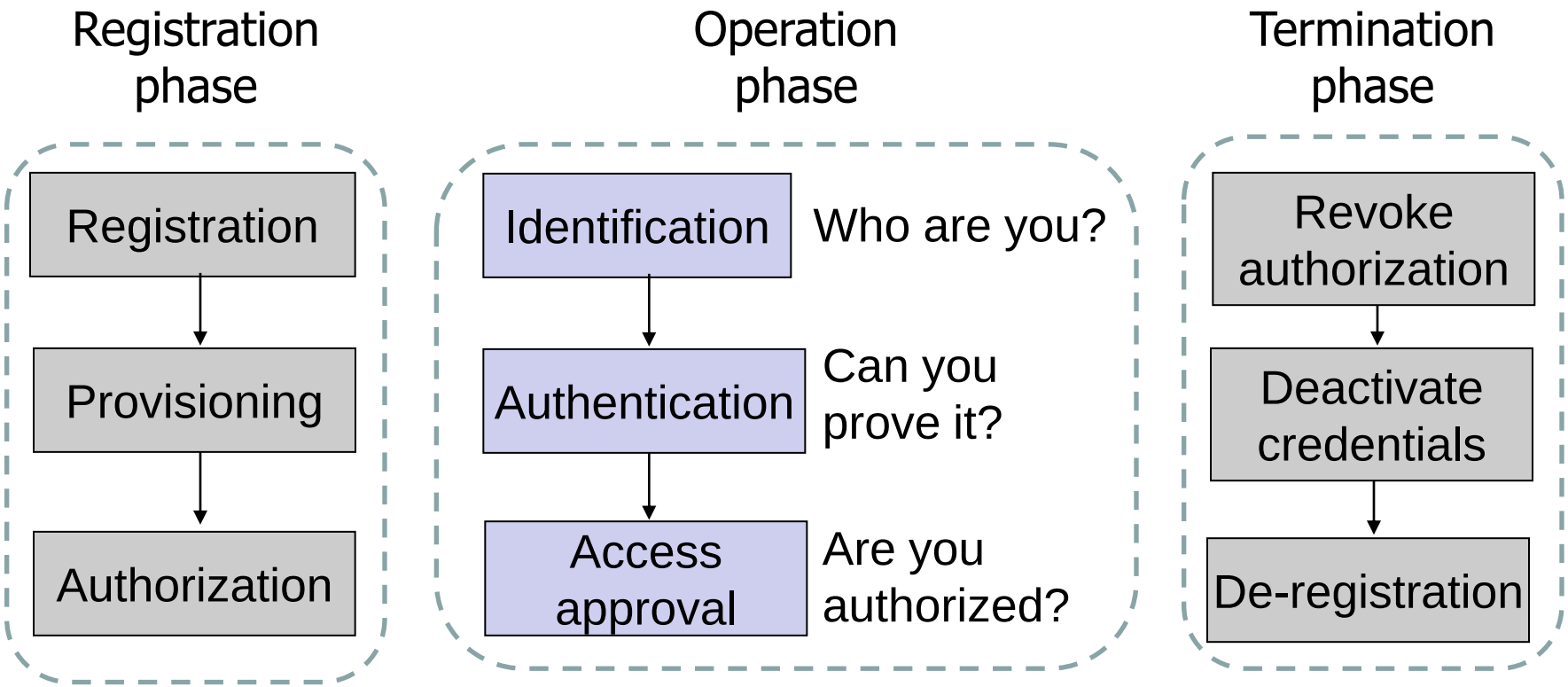
The concept of identity



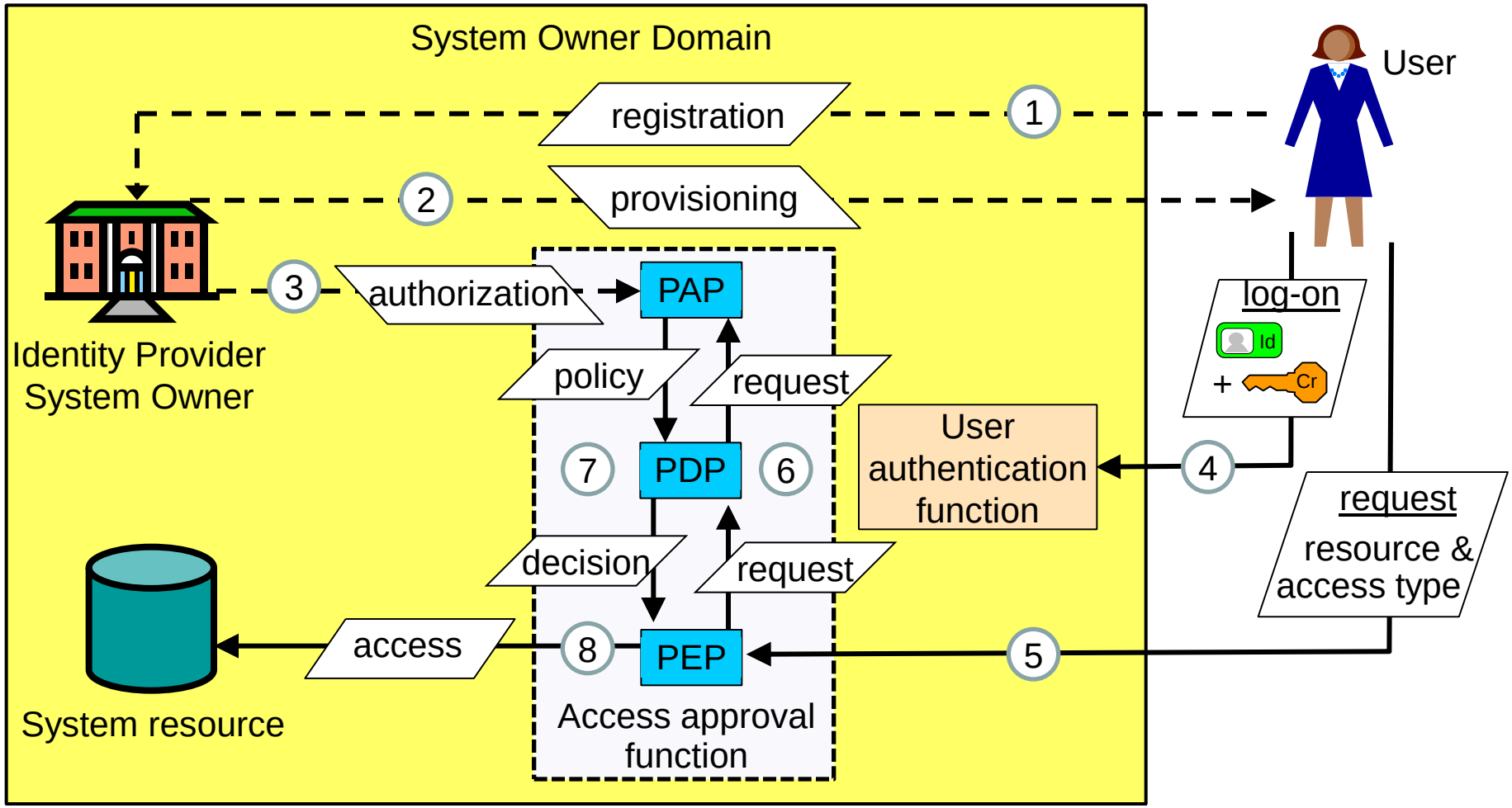
Taxonomy of Authentication



Access Control Phases



Access control concepts (abstract model)



PAP: Policy Administration Point

PDP: Policy Decision Point

PEP: Policy Enforcement Point

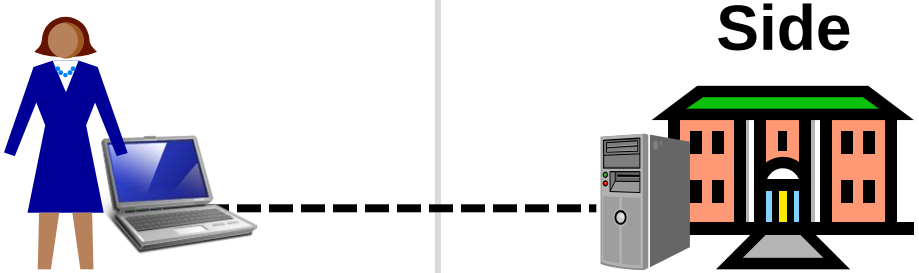
IdP: Identity Provider

Registration

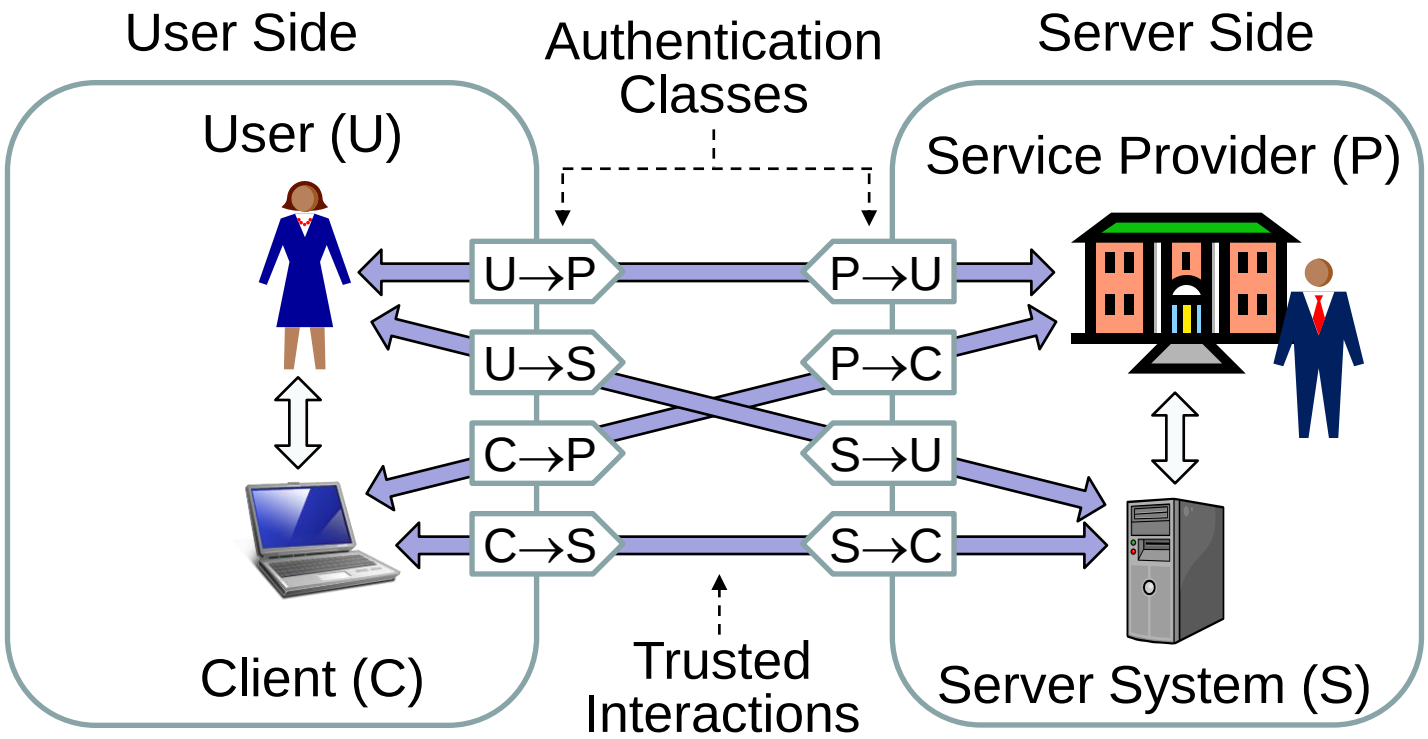
Operations

Identity management processes

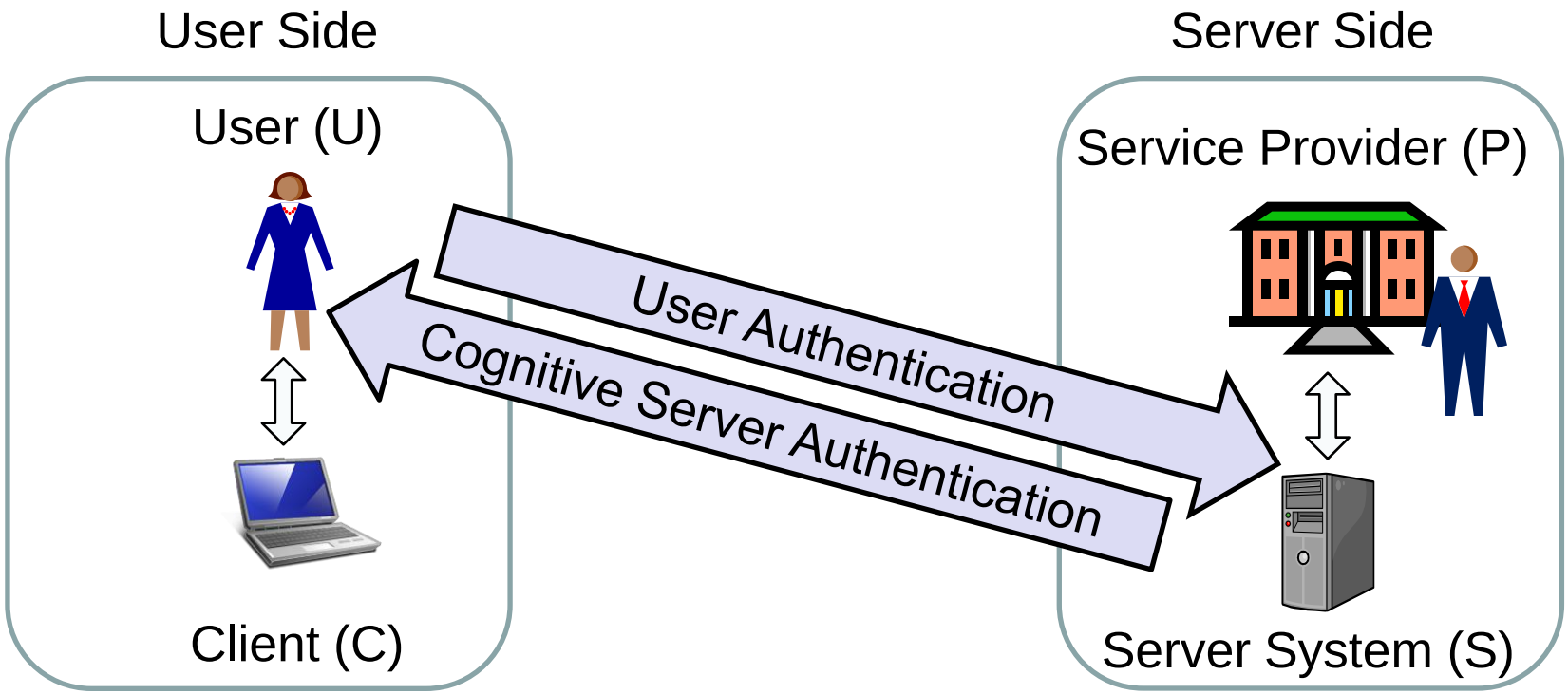
	User Side	Service Provider Side
User Identity Management   **Password / Token**	IdMan processes for user Ids & credentials on user side	IdMan processes for user Ids & credentials on SP side
SP Identity Management  	IdMan processes for SP Ids & credentials on user side	IdMan processes for SP Ids & credentials on SP side



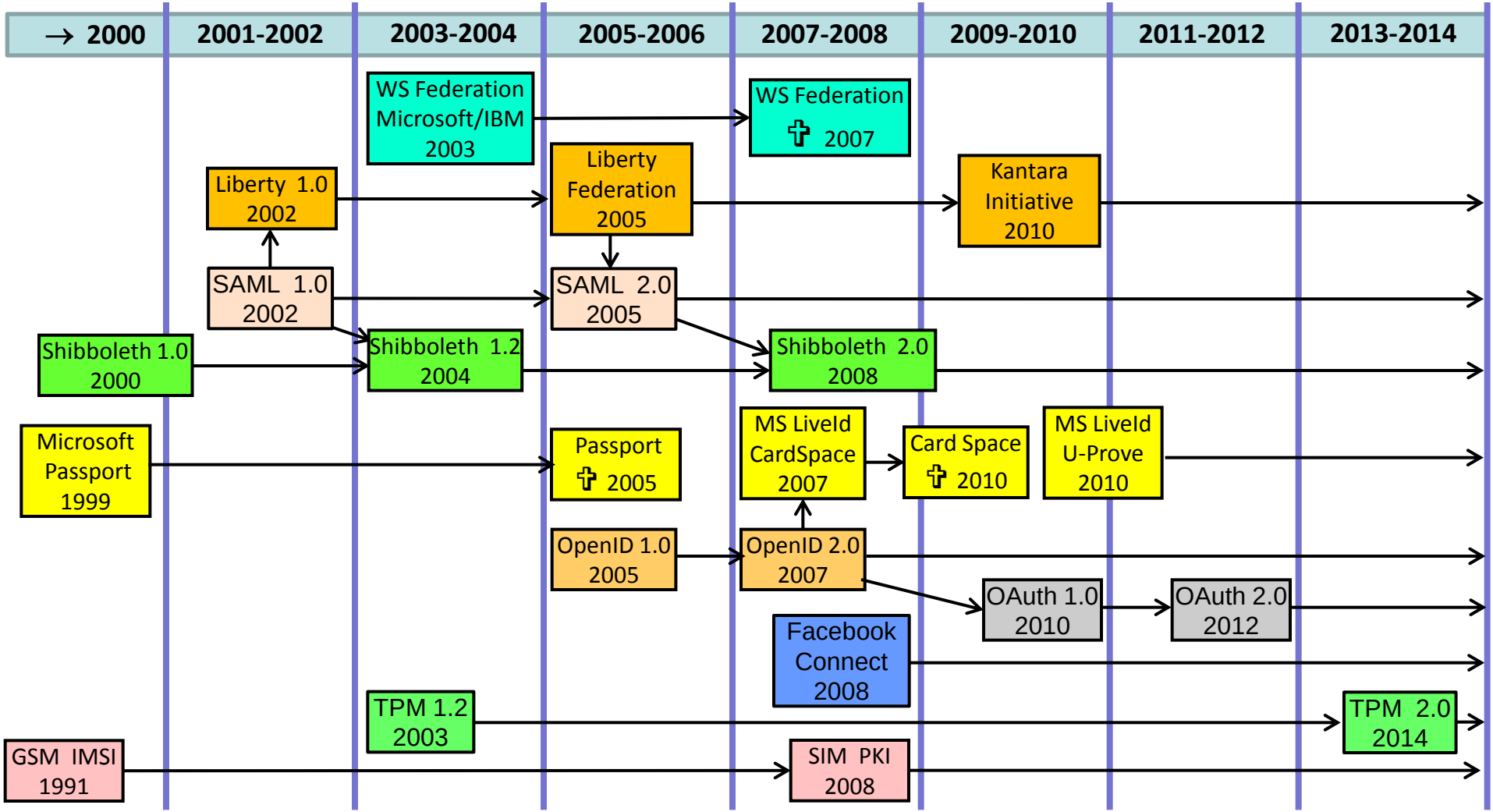
Entity authentication classes



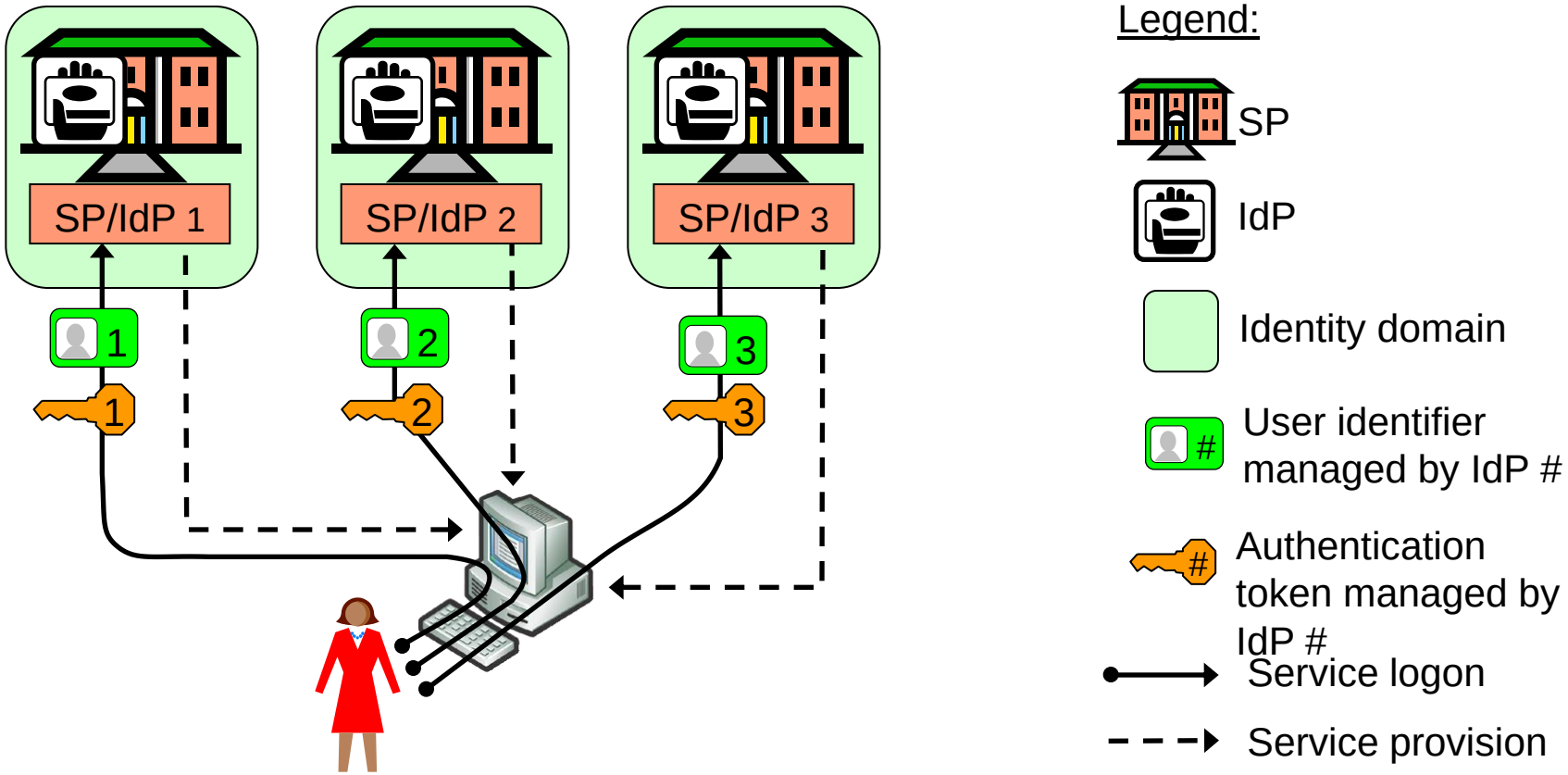
Mutual online entity authentication



Evolution of User Id Management

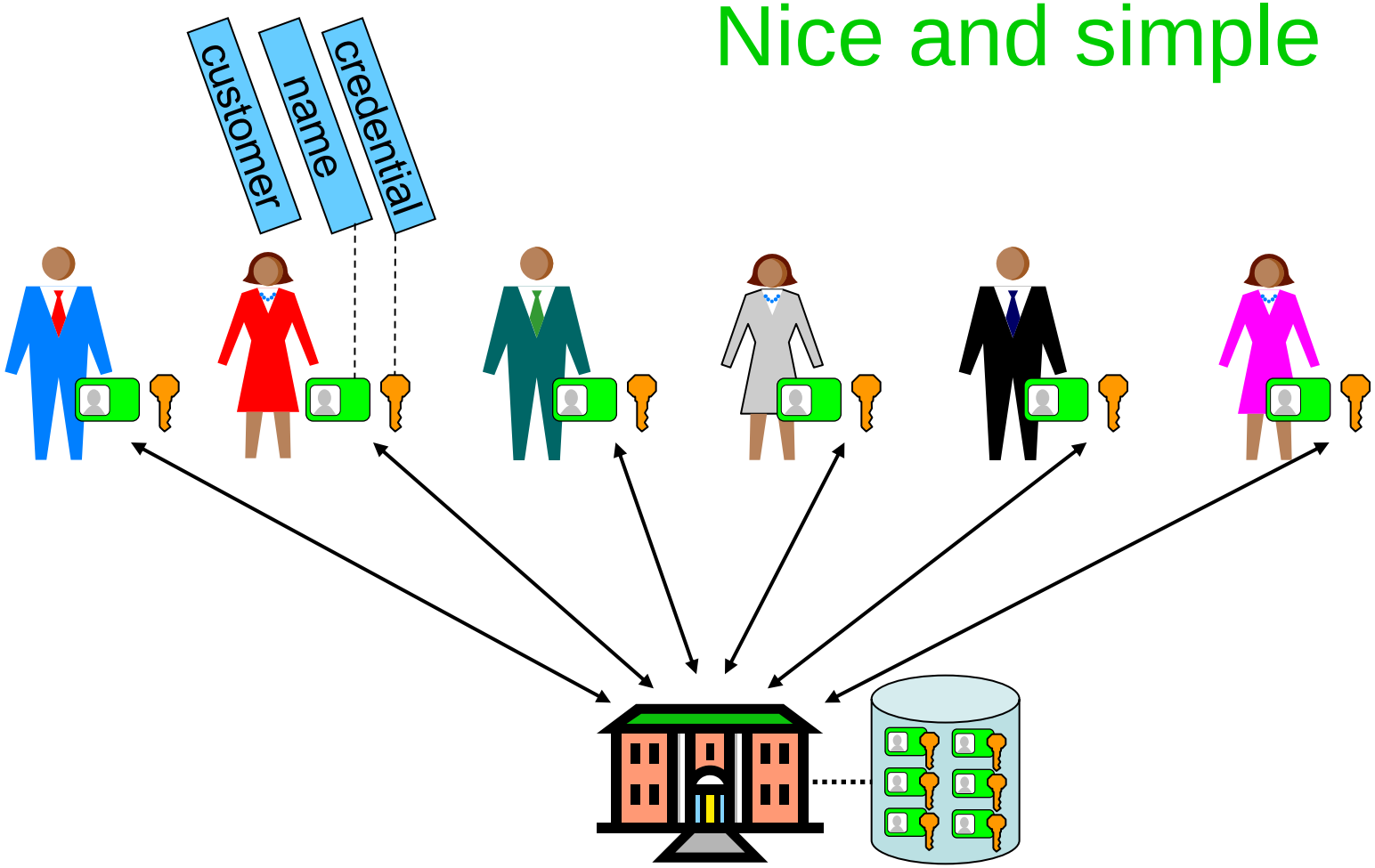


Silo domain model



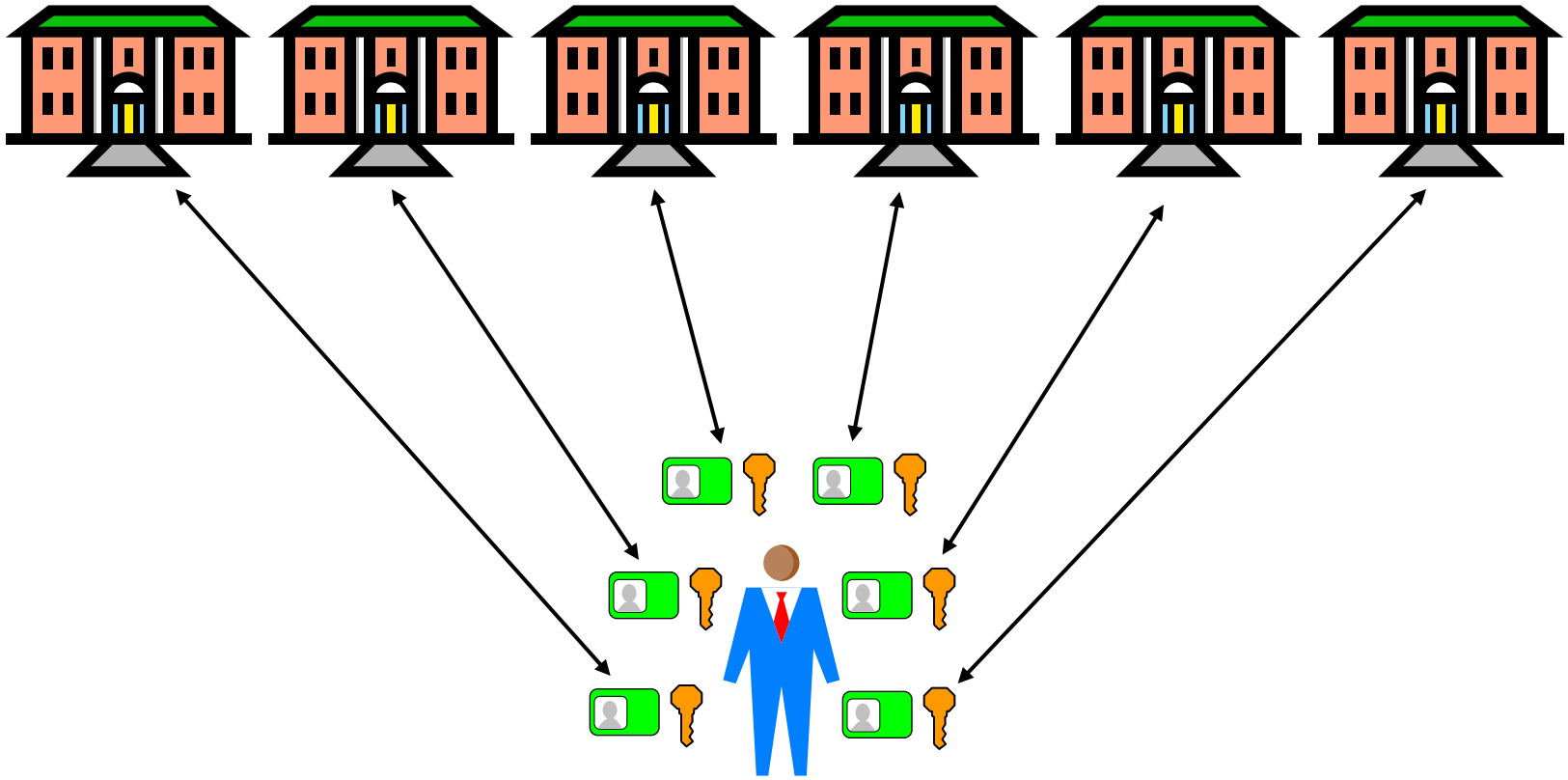
Imagine you're a service provider

Nice and simple



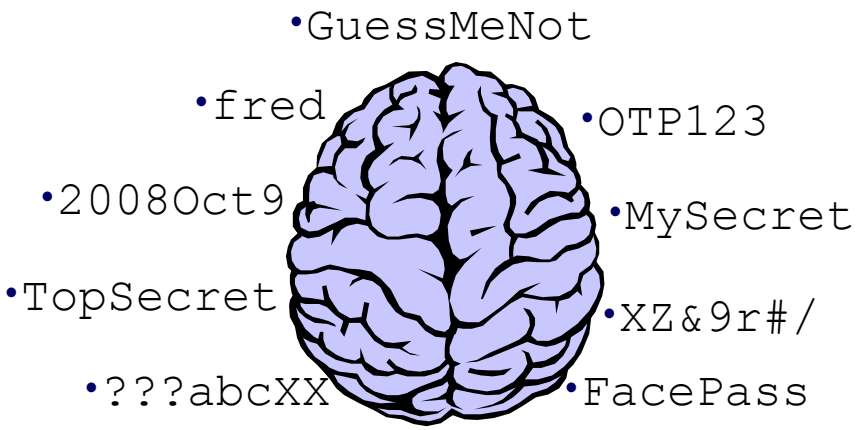
Imagine you're a customer

It's a nightmare



Tragedy of the Commons

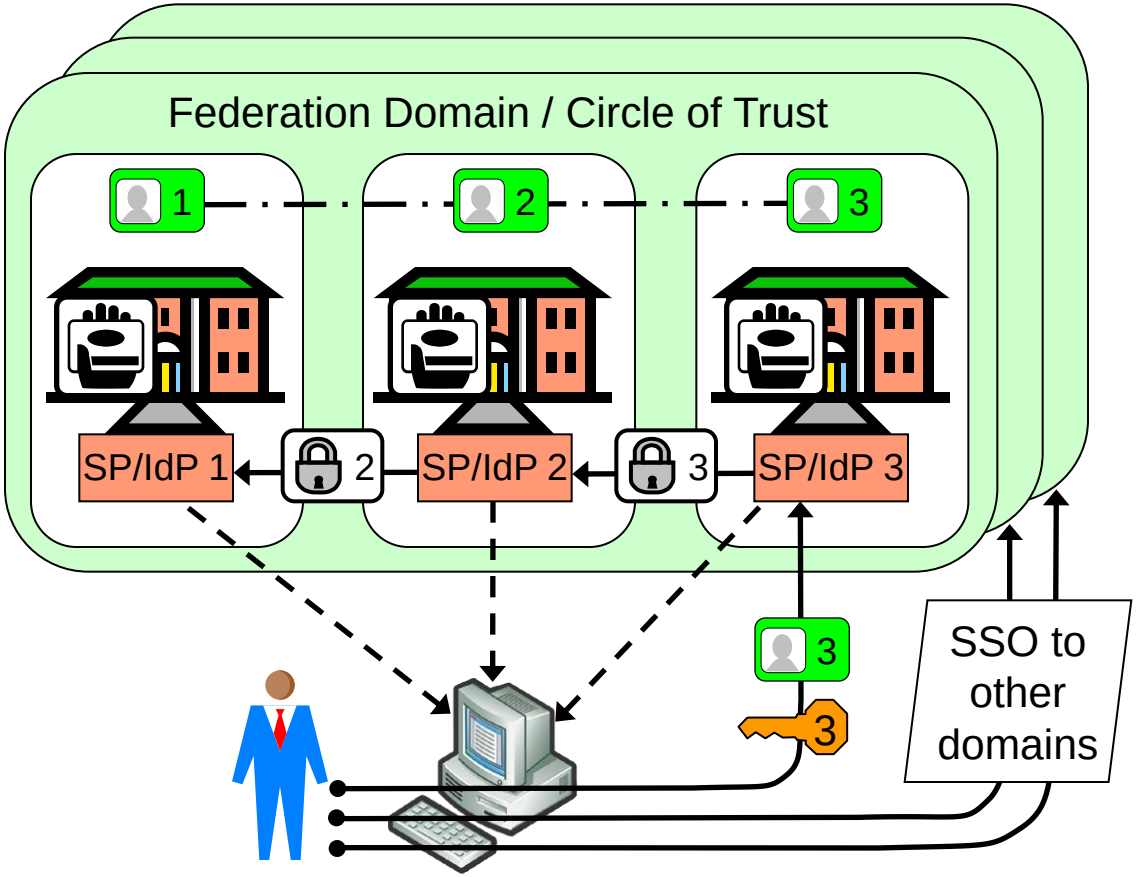
Common village grazing field

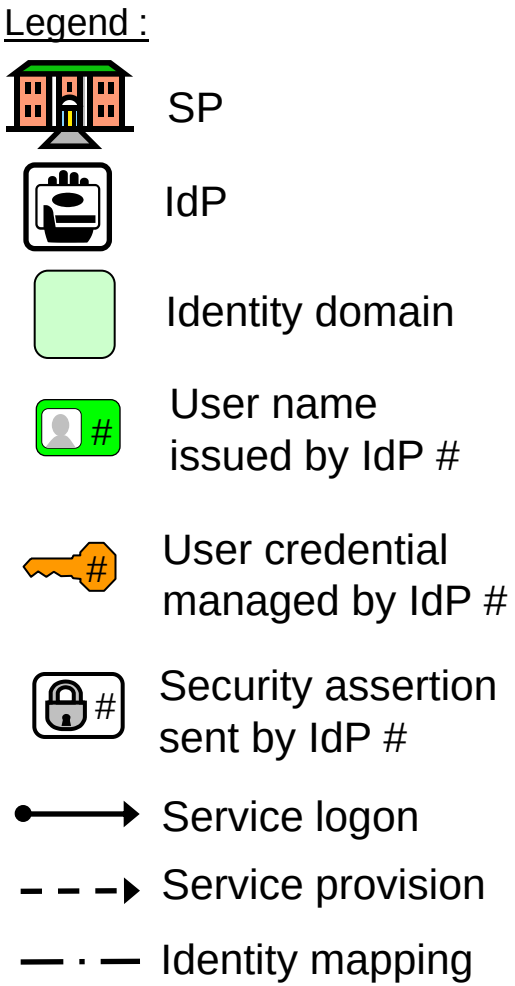


Common brain

Common pockets

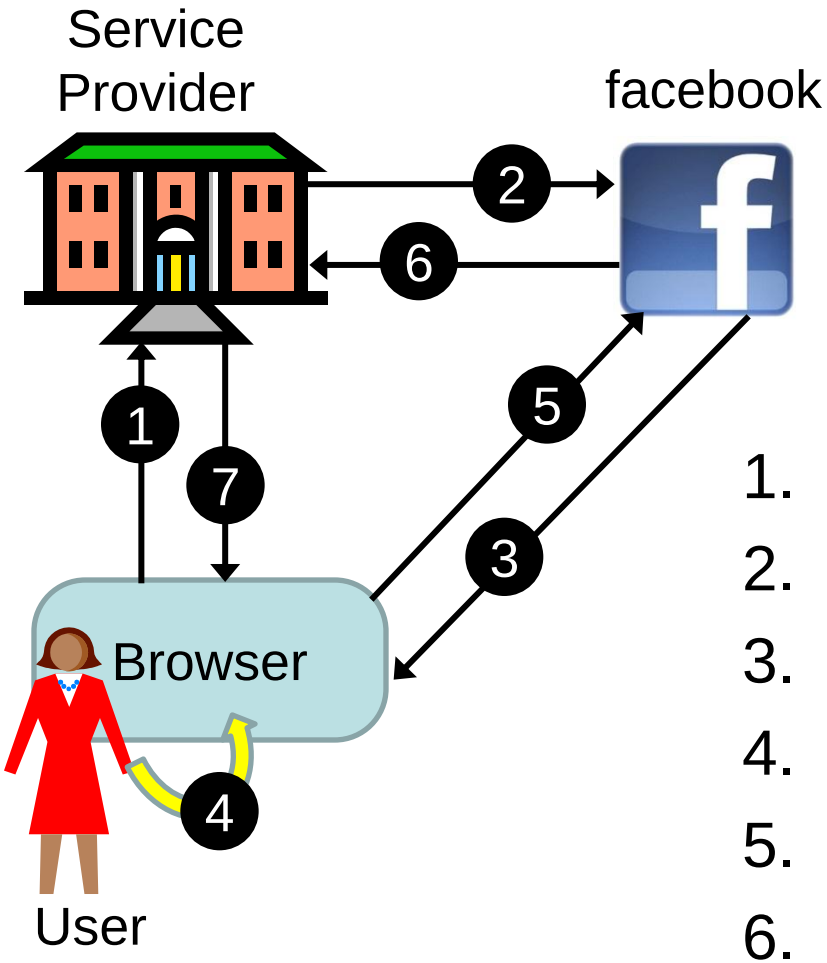
Federated SSO model





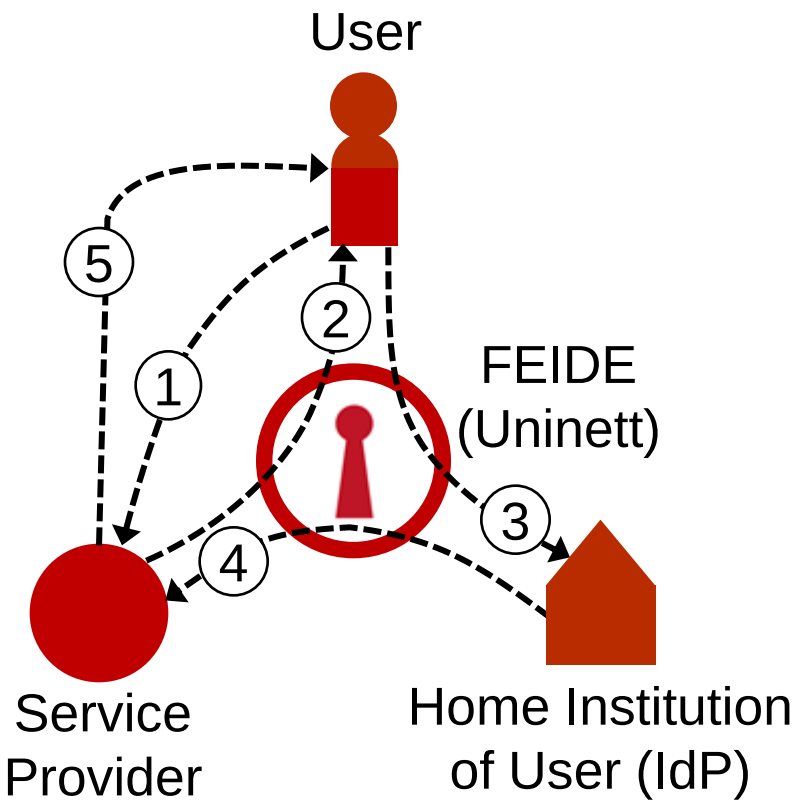
Examples: Liberty Alliance, SAML2.0, WS-Federation, Shibboleth

Authentication via Facebook Connect



1. User requests service
2. Redirect to facebook authentication
3. Present facebook login form
4. User provides Id + credential
5. Credentials forwarded to facebook
6. Confirm authenticated user
7. Provide service

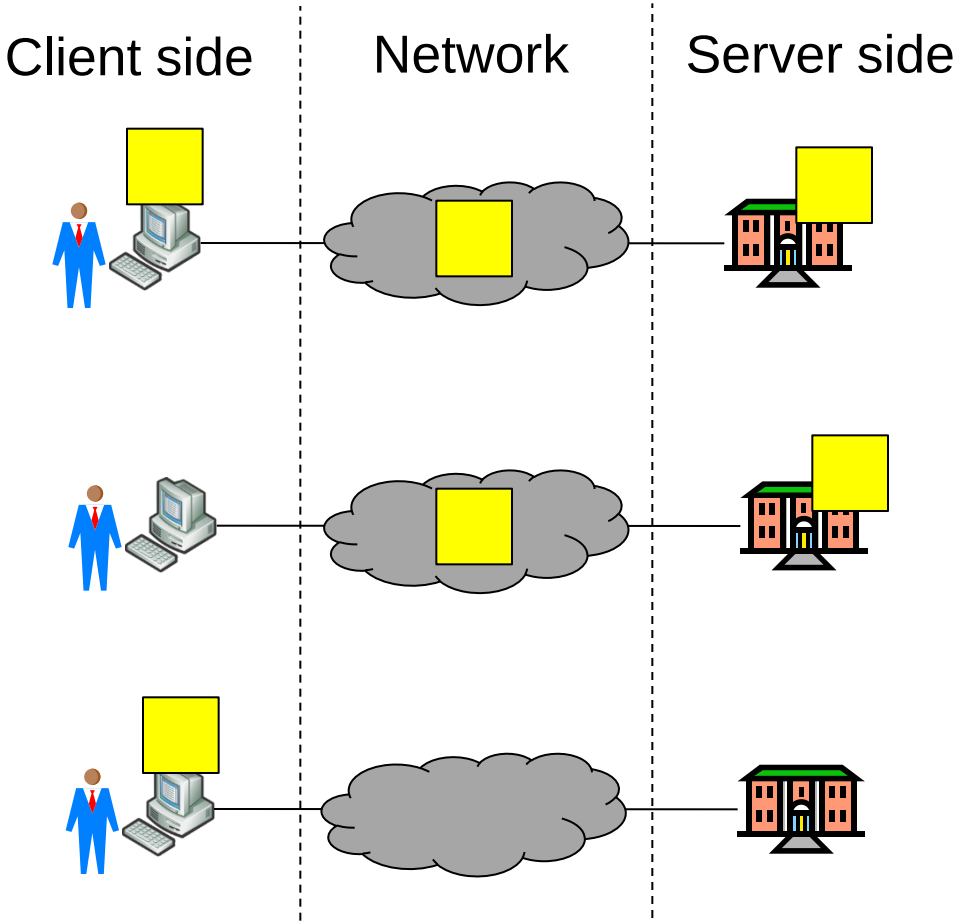
Scenario



1. User requests access to service
2. Service Provider sends authentication request to FEIDE, and displays FEIDE login form to user.
3. User enters name and password in FEIDE login form, which are sent for validation to Home Institution of user.
4. Home Institution confirms authentic user and provides user attributes to FEIDE which forwards these to SP
5. Service Provider analyses user attributes and provides service according to policy

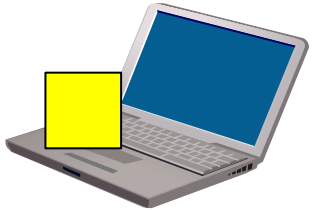
SSO technology location

- Kerberos:
- Federated models:
- Local user-centric:



Client-side location for local user-centric identity management

- IdM in Workstation
  - e.g. SW based password wallet

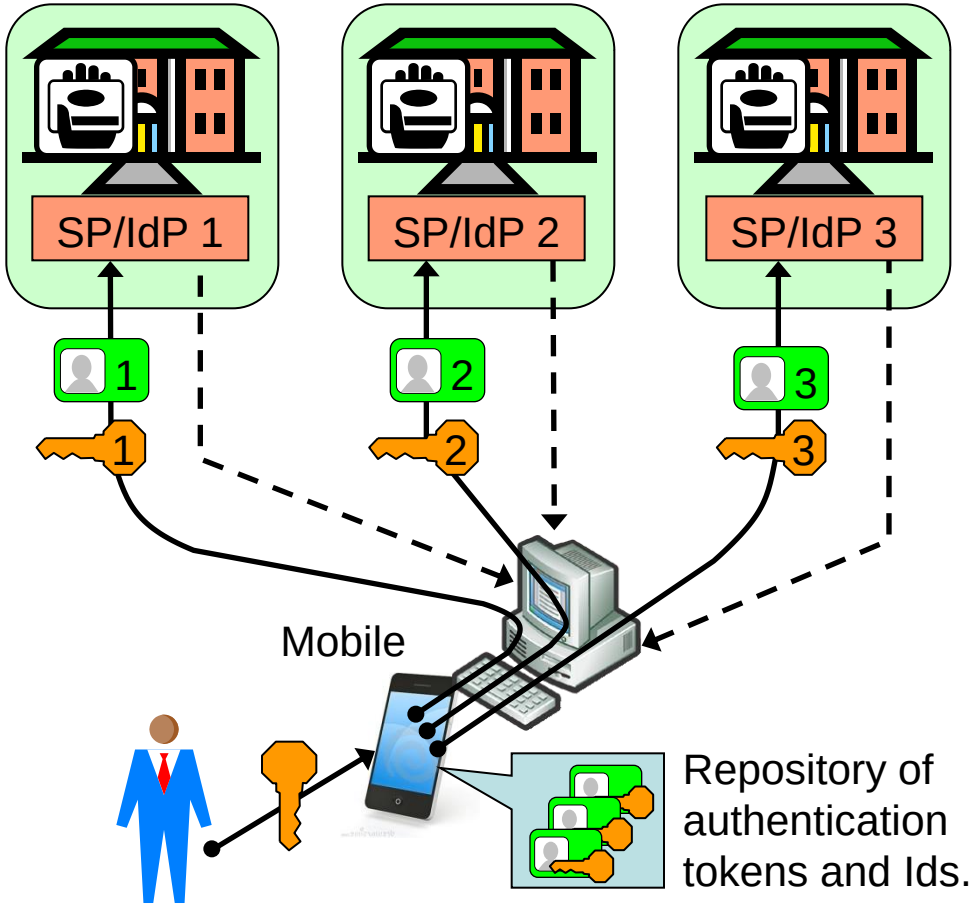


Workstation

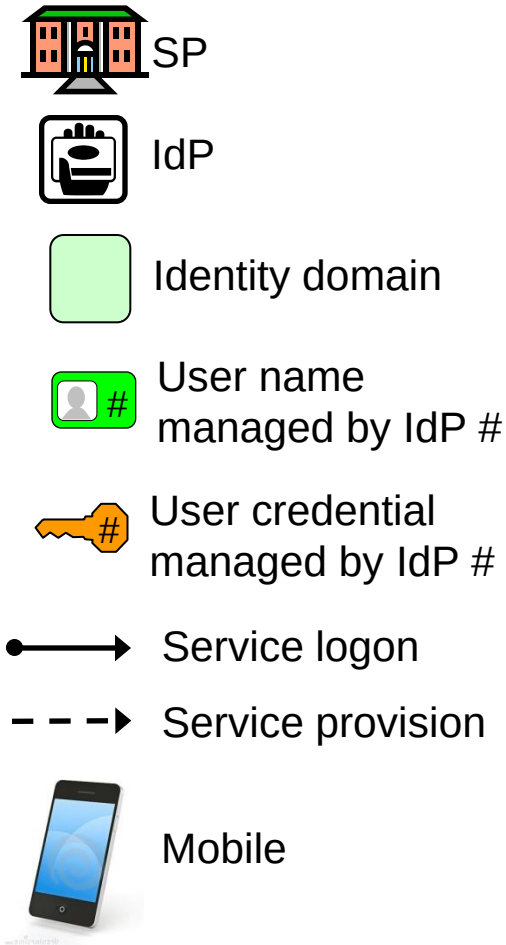
- IdM in Mobile phone
  - e.g SW/SIM based password wallet

mobile

Local user-centric model

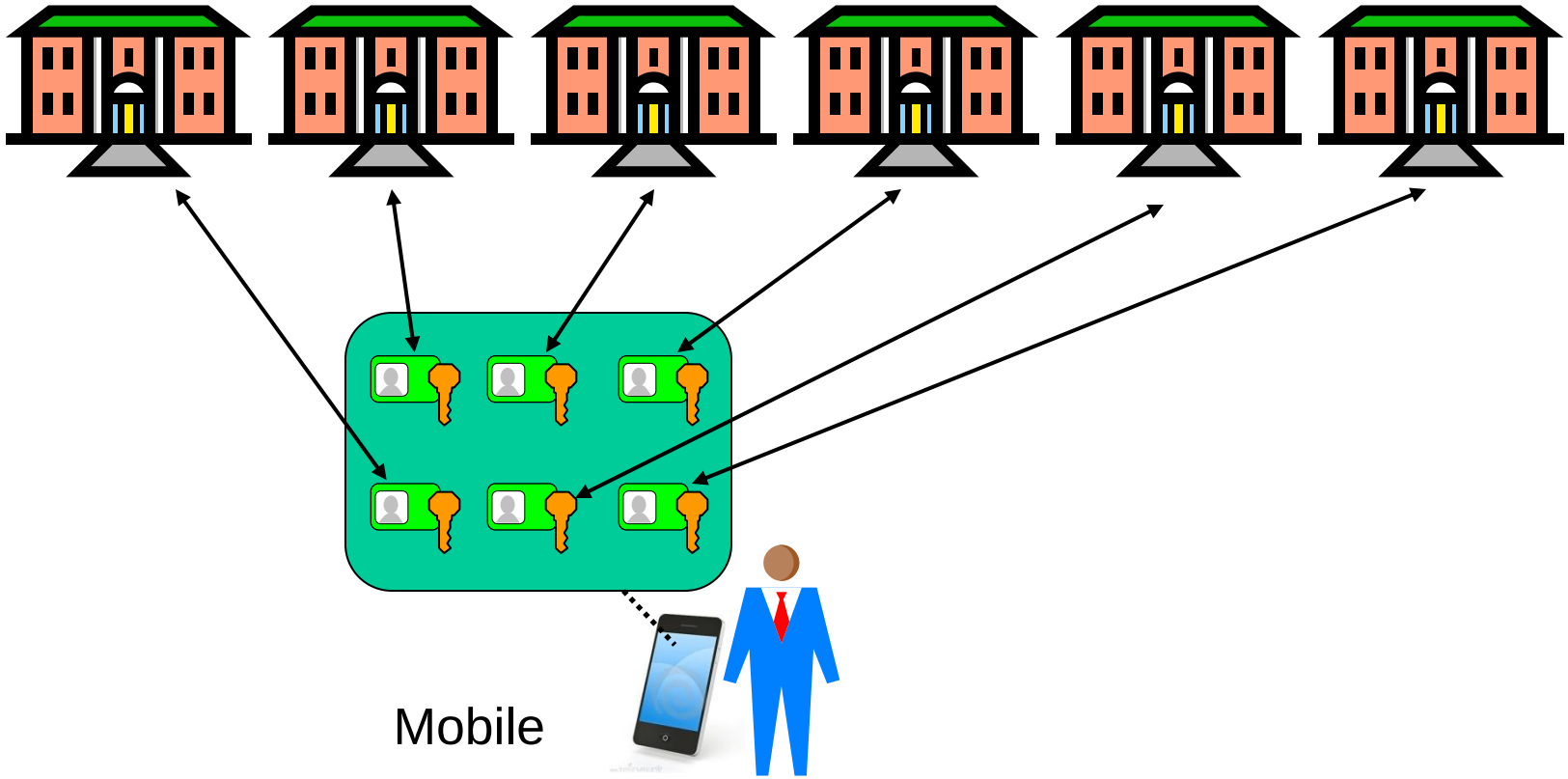


Legend:



Local user-centric: Imagine you're a customer

User friendly and scalable

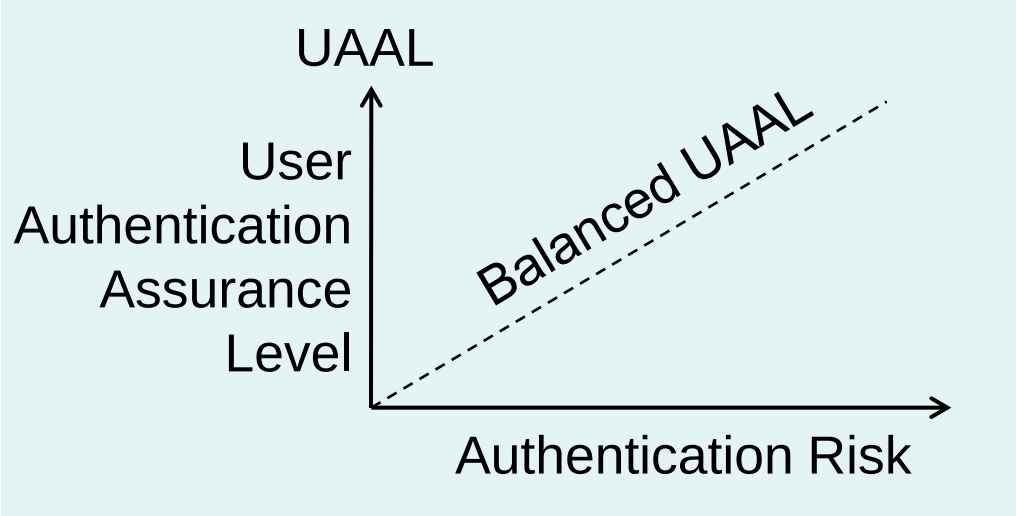


e-Authentication Frameworks for e-Gov.

- Trust in identity is a requirement for e-Government
- Authentication assurance produces identity trust.
- Authentication depends on technology, policy, standards, practice, awareness and regulation.
- Consistent frameworks allow cross-national and cross-organisational schemes that enable convenience, efficiency and cost savings.

Authentication Assurance

- Authentication assurance = robustness of authentication
- Resources have different sensitivity levels
  - High sensitivity gives high risk in case of authentication failure
- Authentication has a cost
  - Unnecessary authentication assurance is a waste of money
- Authentication assurance should balance resource sensitivity



Factors for UAAL

User Authentication Method Strength (UAMS)

How strong is the mechanism ?

- Password length and quality
- Cryptographic algorithm strength
- Tamper resistance of token

User Credential Management Assurance (UCMA)

Min[UAMS, UCMA, UIRA] = User Authentication Assurance Level (UAAL)

Are credentials handled securely ?

- Generation
- Distribution
- Storage

User Identity Registration Assurance (UIRA)

How certain are you that the correct user was registered in the first place ?

- Pre-authentication credentials, e.g. birth certificate
- Biometrics

UAAL: User Authentication Assurance Levels

No Assurance	Minimal Assurance	Low Assurance	Moderate Assurance	High Assurance
Level 0	Level 1	Level 2	Level 3	Level 4
No registration of identity required	Minimal confidence in the identity assertion	Low confidence in the identity assertion	Moderate confidence in the identity assertion	High confidence in the identity assertion

Example taken from Australian NeAF 2009

Alignment of e-Authentication Frameworks

Authentication Framework	***User Authentication Assurance Levels***				
EAG (USA) 2006	Little or no assurance (1)		Some (2)	High (3)	Very High (4)
IDABC (EU) 2007	×	Minimal (1)	Low (2)	Substantial (3)	High (4)
FANR (Norway) 2008	Little or no assurance (1)		Low (2)	Moderate (3)	High (4)
NeAF (Australia) 2009	None (0)	Minimal (1)	Low (2)	Moderate (3)	High (4)
ePramaan (India) 2013	None (0)	Minimal (1)	Moderate (2)	Strong (3)	Very Strong (4)

Risk Analysis for Authentication

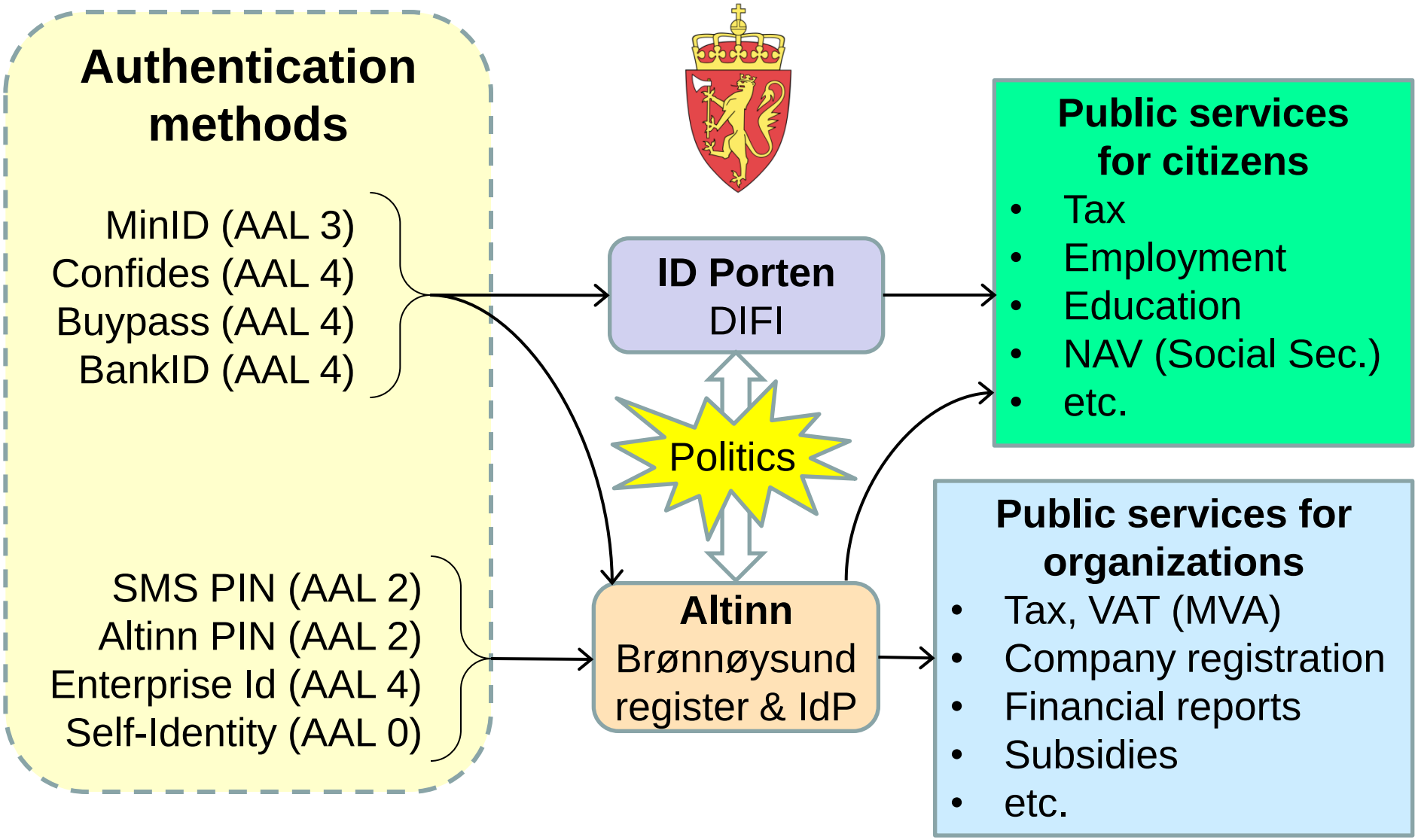
Determining the appropriate UAAL for an application

		Impact of e-Authentication failure				
		Insignificant	Minor	Moderate	Major	Severe
Likelihood	Almost Certain	None (0)	Low (2)	Moderate (3)	High (4)	High (4)
	Likely	None (0)	Low (2)	Moderate (3)	High (4)	High (4)
	Possible	None (0)	Minimal (1)	Low (2)	Moderate (3)	High (4)
	Unlikely	None (0)	Minimal (1)	Low (2)	Moderate (3)	Moderate (3)
	Rare	None (0)	Minimal (1)	Low (2)	Moderate (3)	Moderate (3)

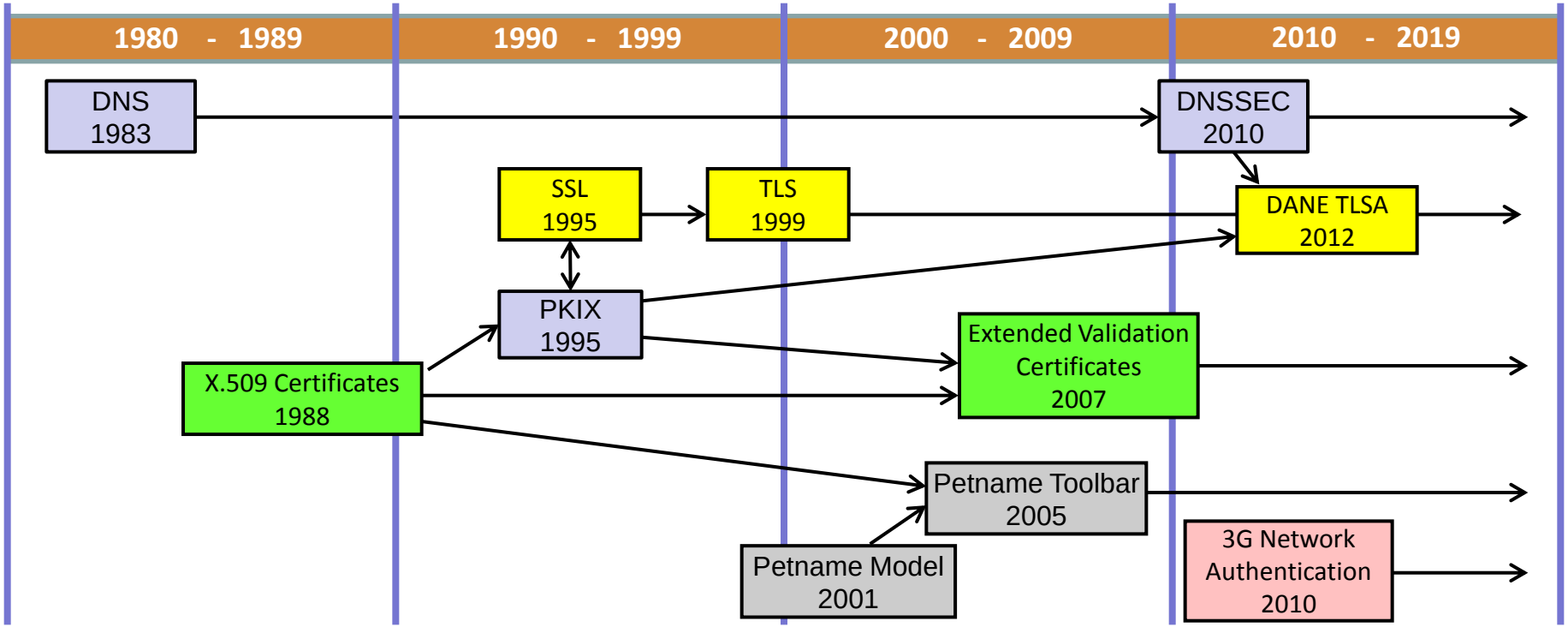
Required UAAL

Example: NeAF Australia

Id Management for Norwegian e-Gov.

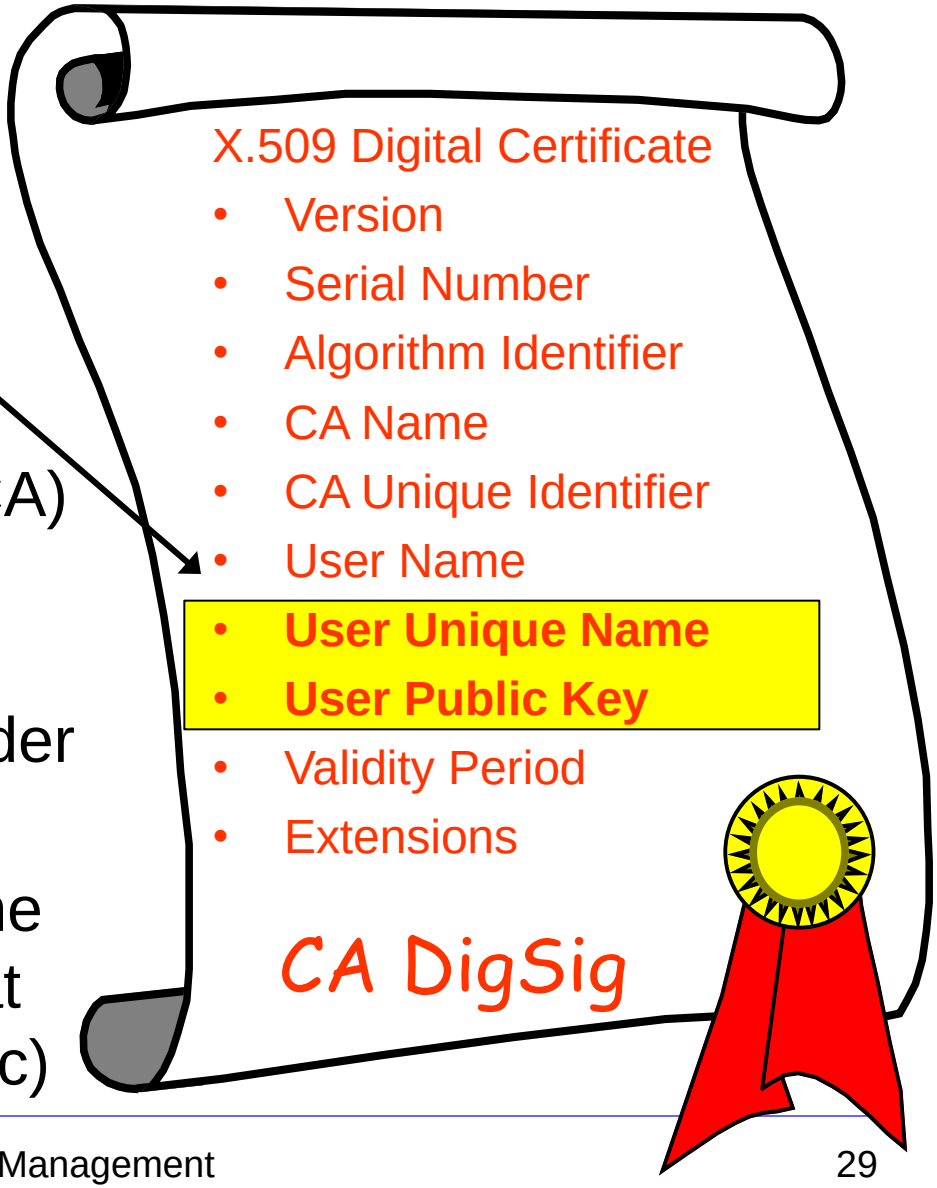


Evolution of SP Id Management



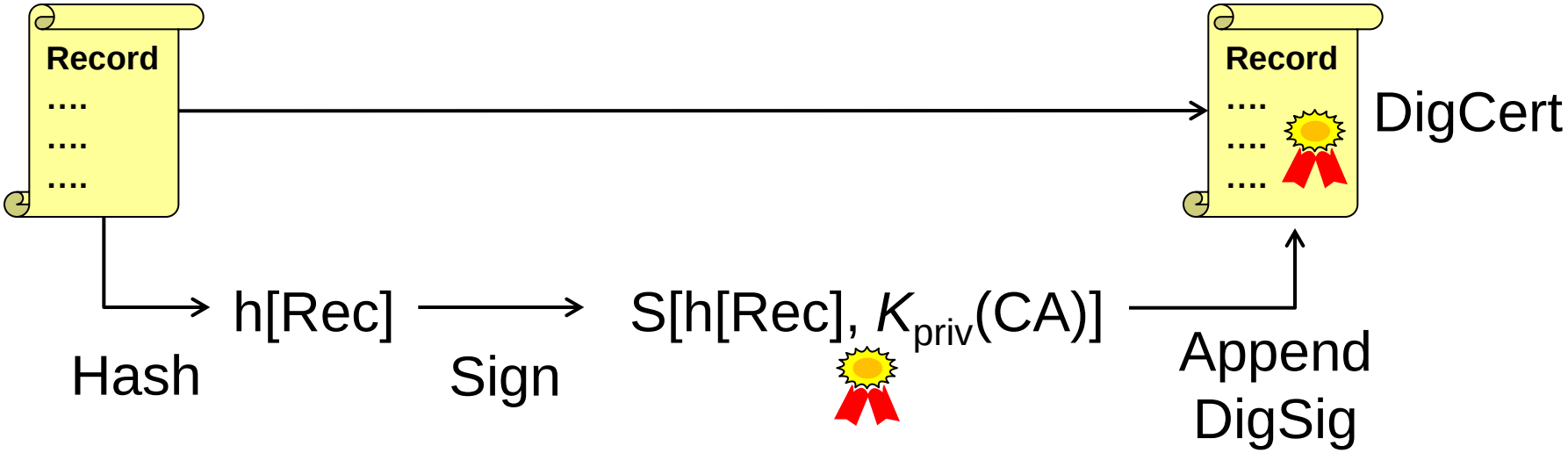
Public-Key Certificates

- A public-key certificate is simply a public key with a digital signature
- Binds name to public key
- Certification Authorities (CA) sign public keys.
- An authentic copy of CA's public key is needed in order to validate certificate
- **Relying party** validates the certificate (i.e. verifies that user public key is authentic)

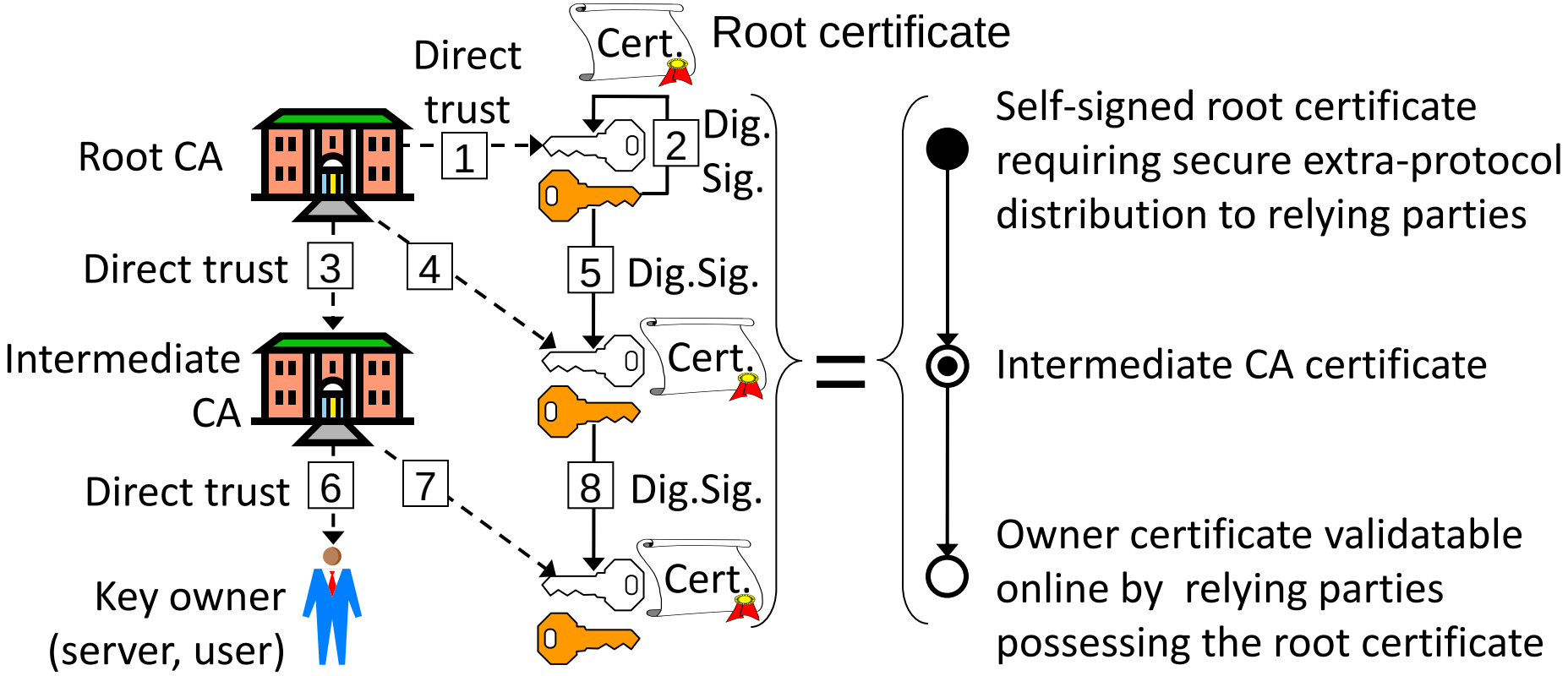


How to generate a digital certificate?

1. Assemble the information (name and public key) in single record Rec
2. Hash the record
3. Sign the hashed record
4. Append the digital signature to the record



PKI certificate generation



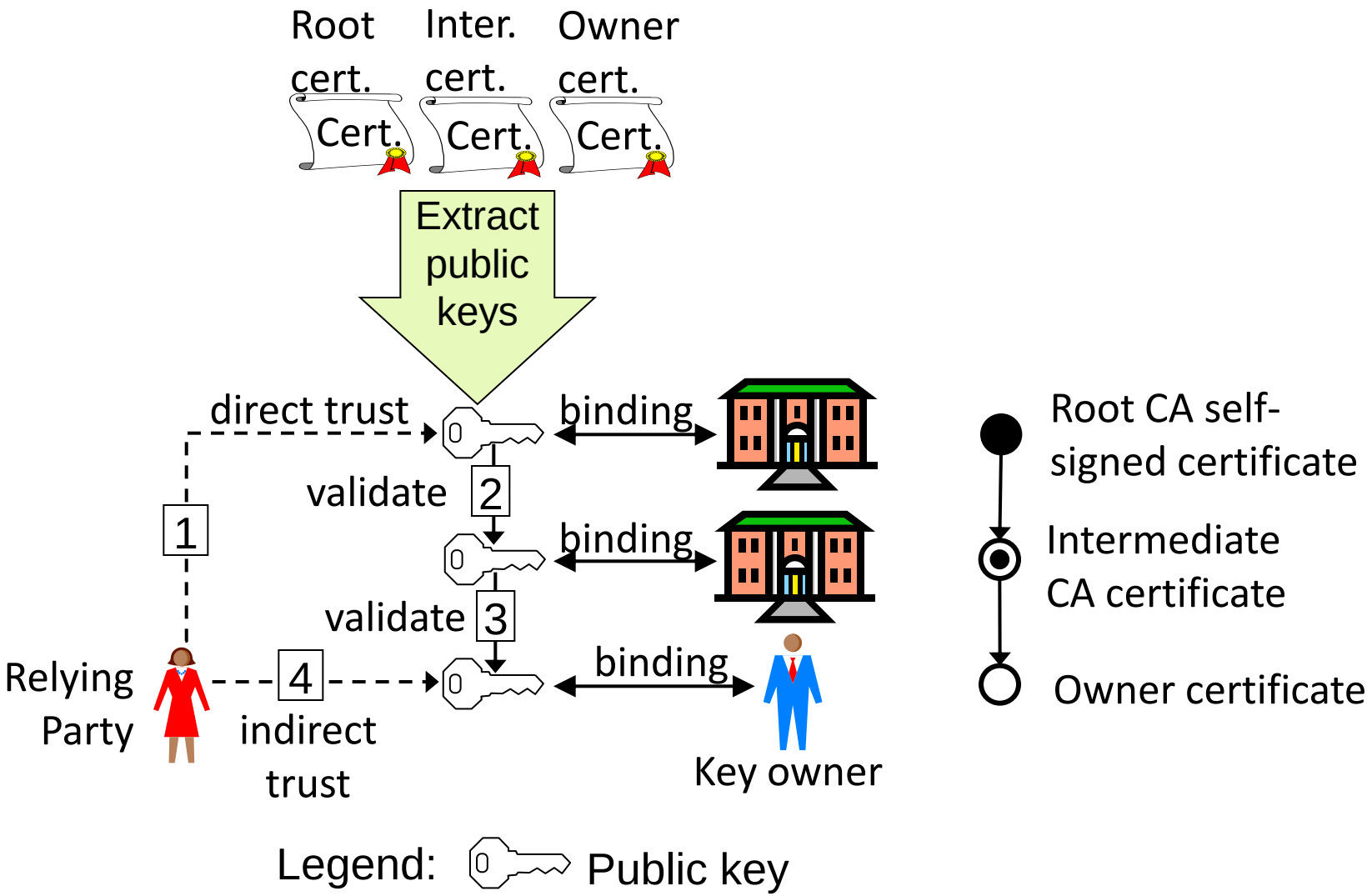
Legend: Public key Private key

Self-signed root keys: Why?

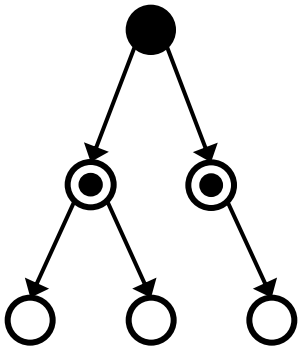
- Many people think a root public key is authentic just because it is self-signed
- Can be deceptive
  - Gives impression of assurance
  - Disguises insecure practice
  - Gives false trust

- Self-signing provides <u>absolutely no security</u>
- Useful purpose of self-signing
  - X.509 certificates have a field for digital signature, so an empty field might cause applications to malfunction. A self-signature is a way to fill the empty field
  - Self-signature can be used to specify a cert as a root

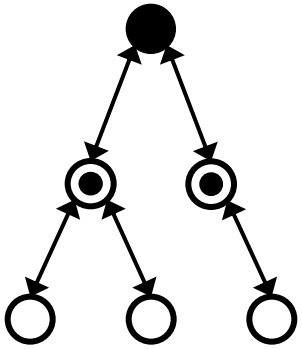
Certificate and public key validation



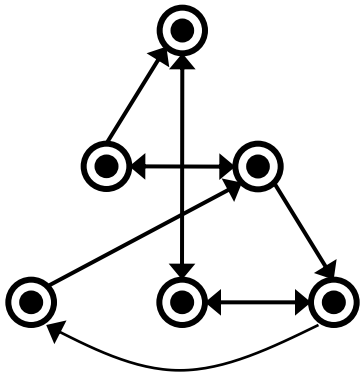
PKI Trust Models



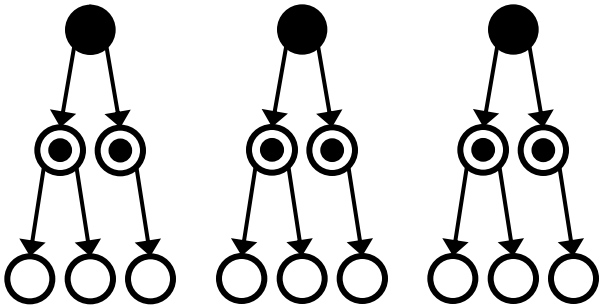
Strict hierarchy
e.g. ` DNSSEC PKI ’



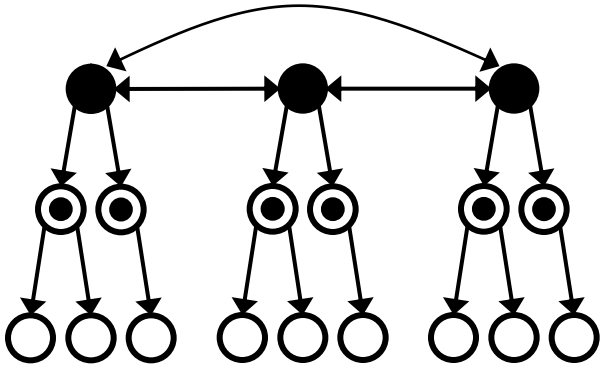
Bi-directional hierarchy



Ad-hoc anarchic PKI

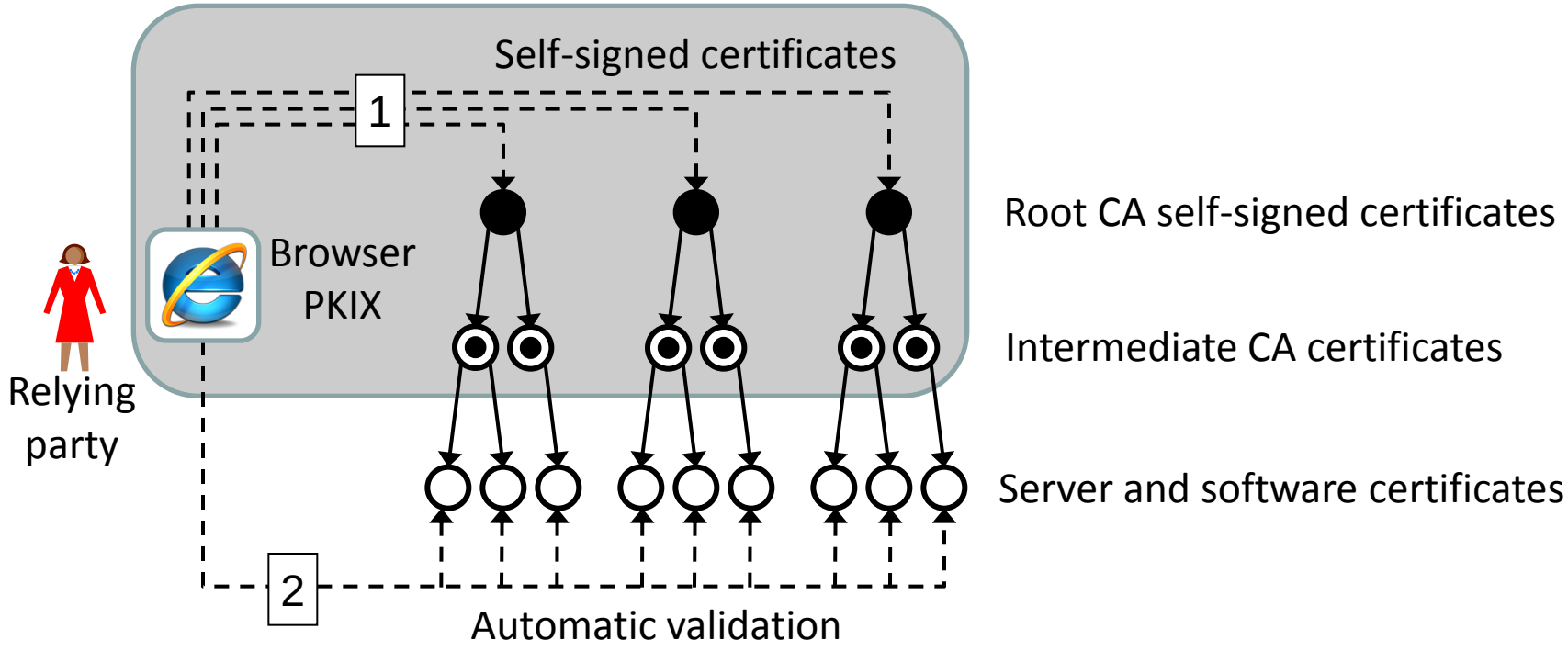


Isolated strict hierarchies
e.g. ` Browser PKIX ’

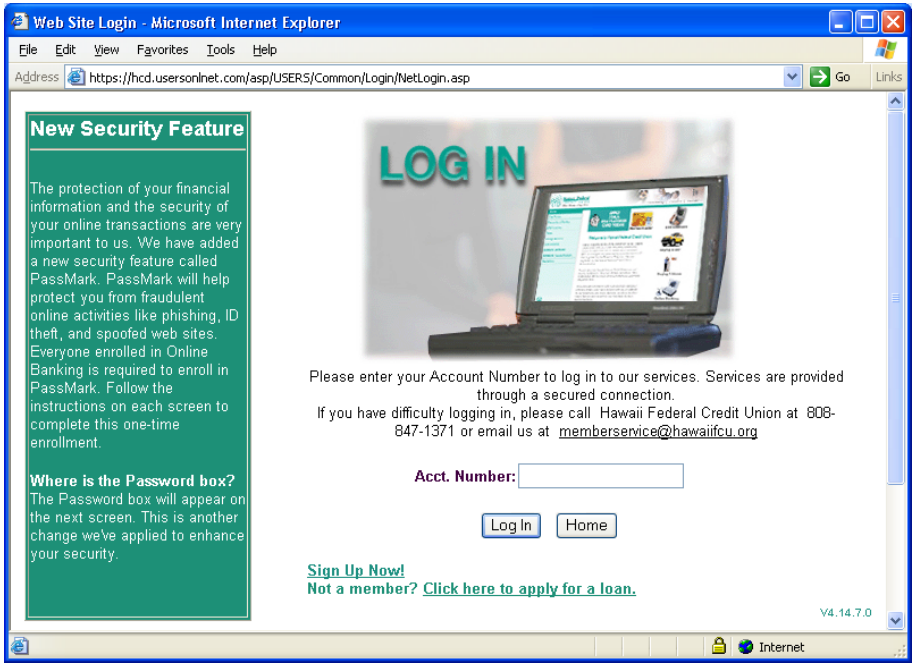


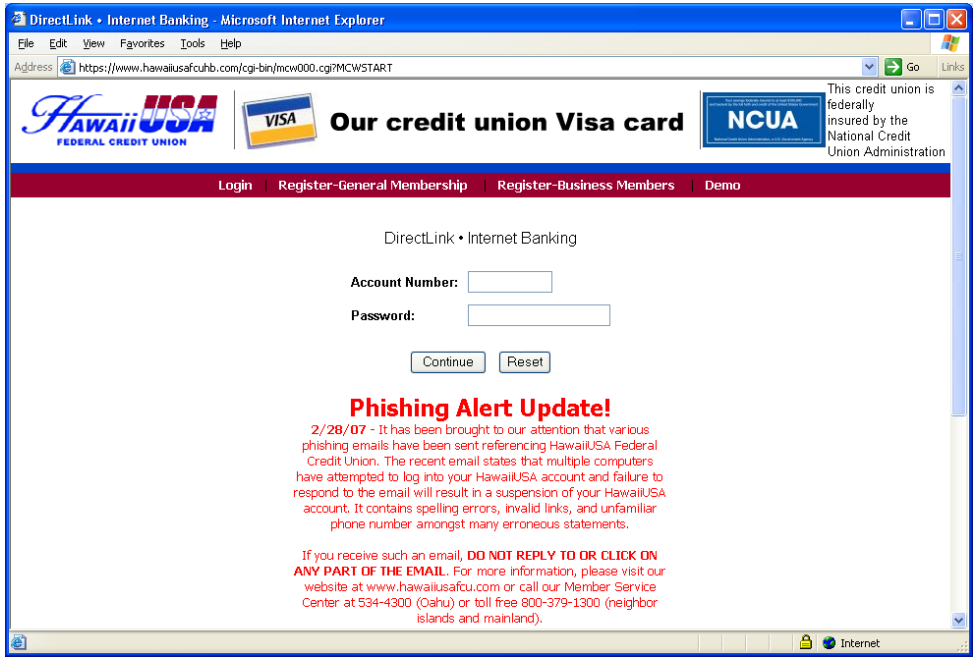
Cross-certified strict hierarchies

Browser PKIX



A phishing example: Hawaii Federal Credit Union





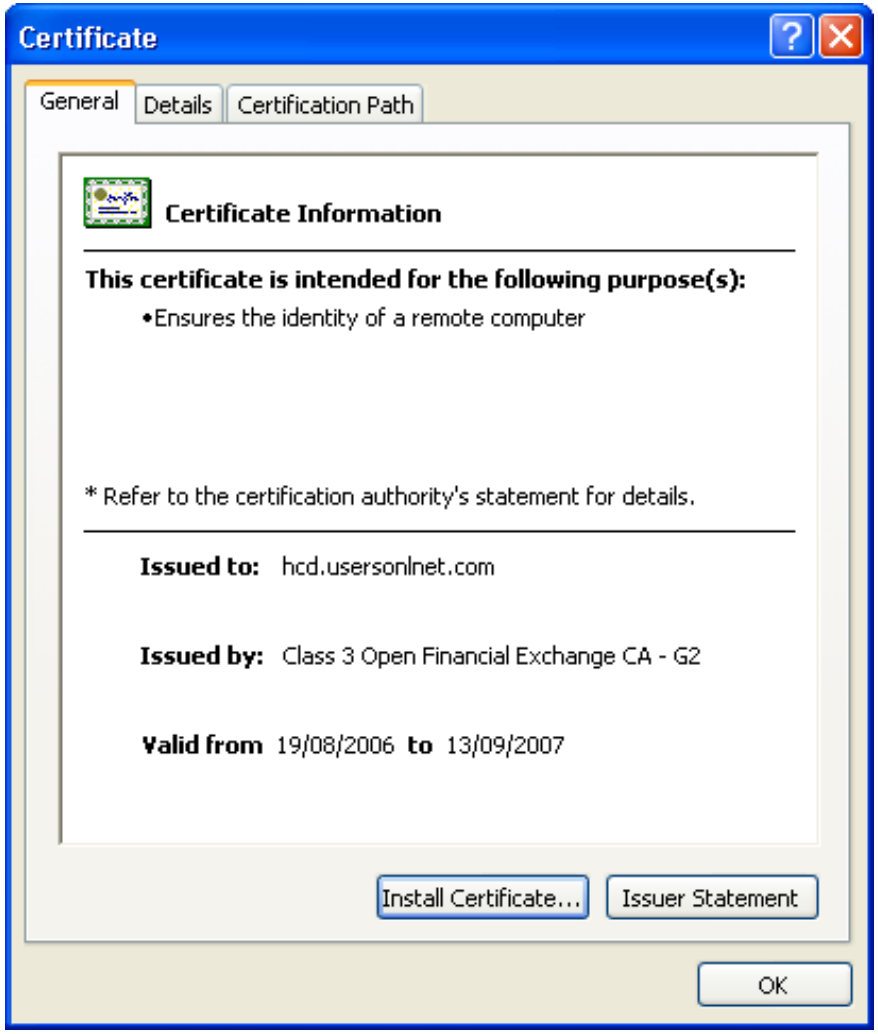
Genuine bank login

https://hcd.usersonlnet.com/asp/USERS/Common/Login/NettLogin.asp

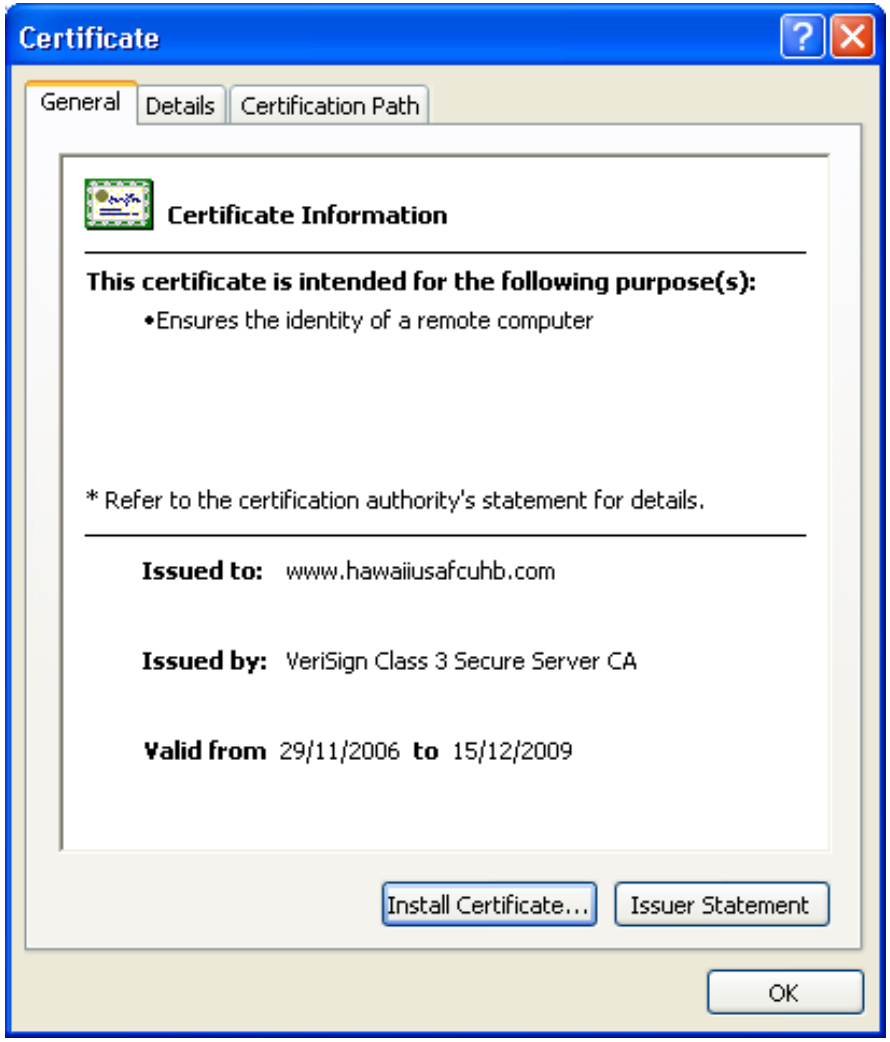
Fake bank login

https://hawaiiusafcuhb.com/cgi-bin/mcw00.cgi?MCWSTART

Certificate comparison 1

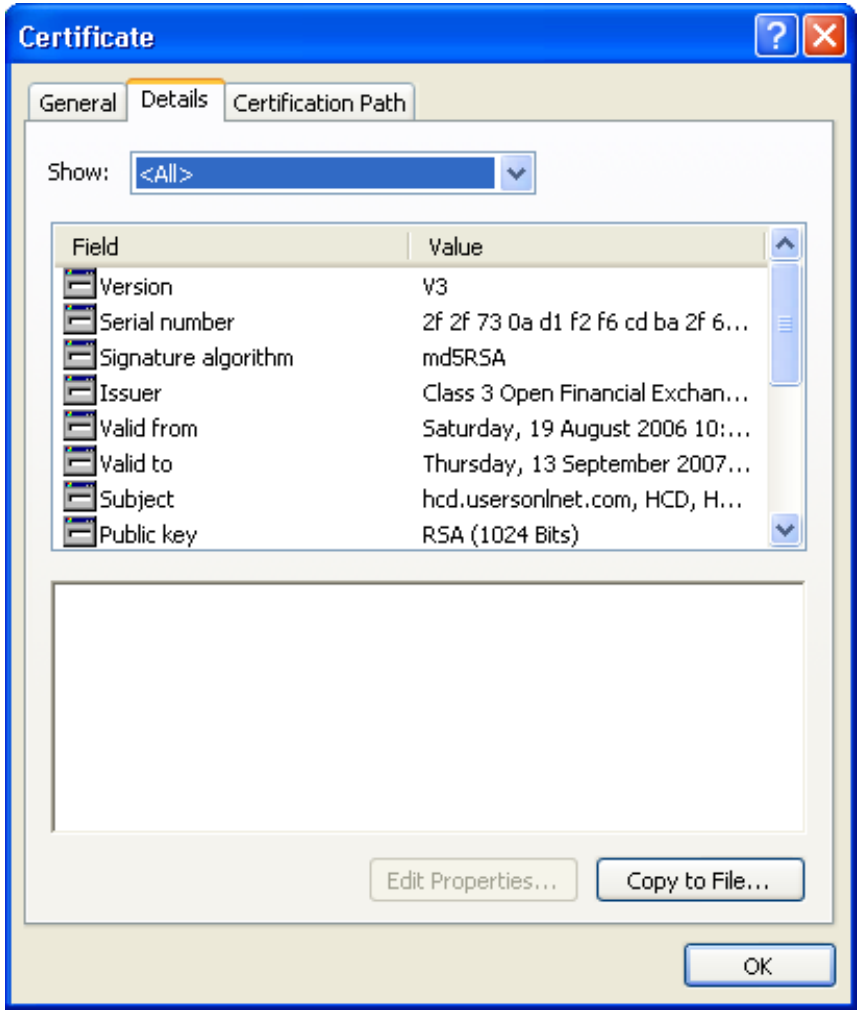


Genuine certificate

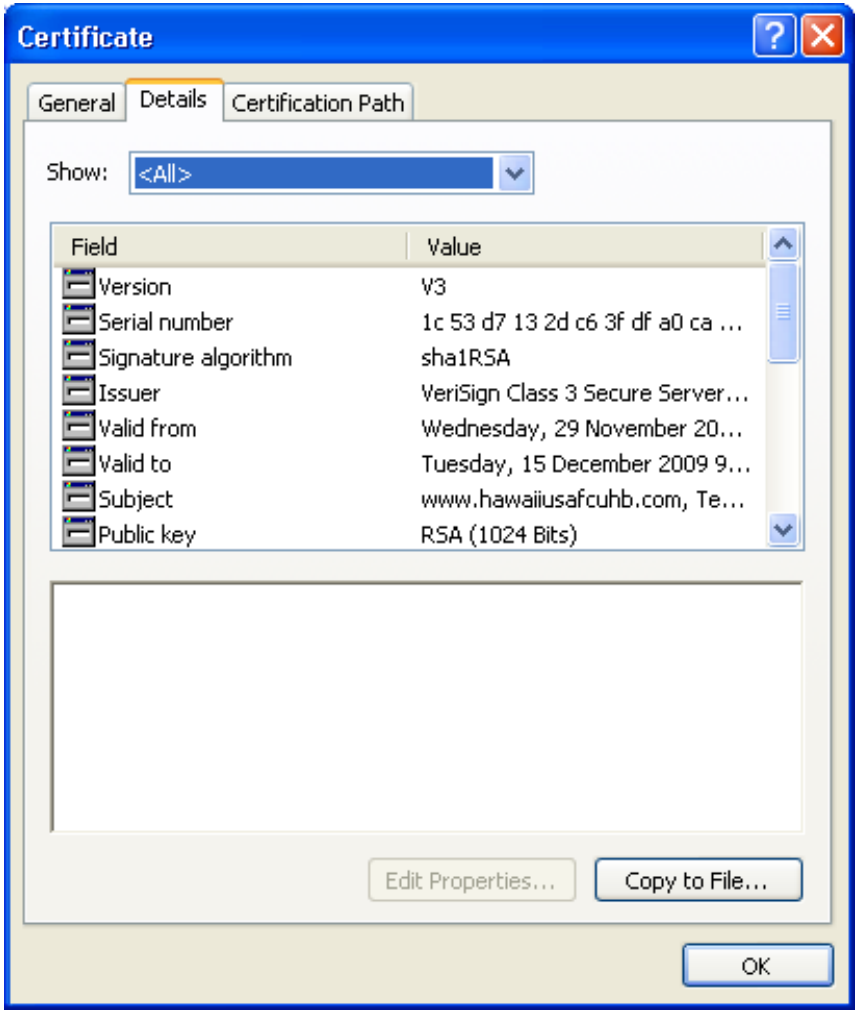


Fake certificate

Certificate comparison 2

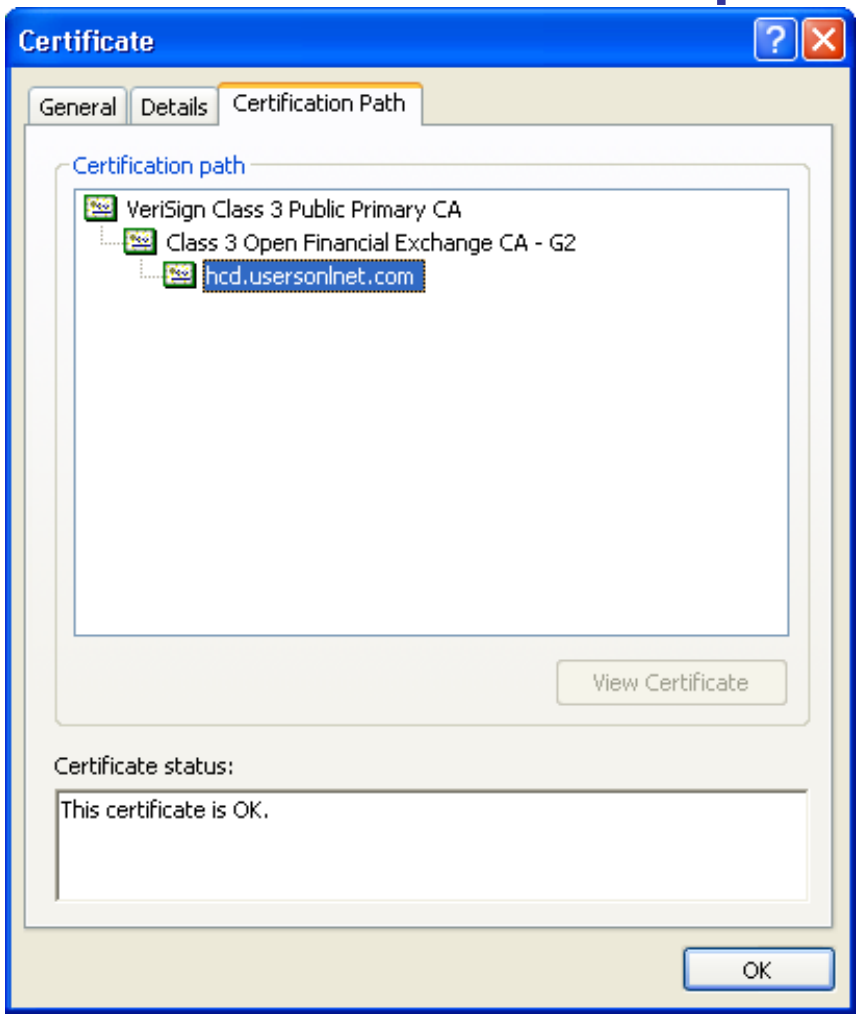


Genuine certificate

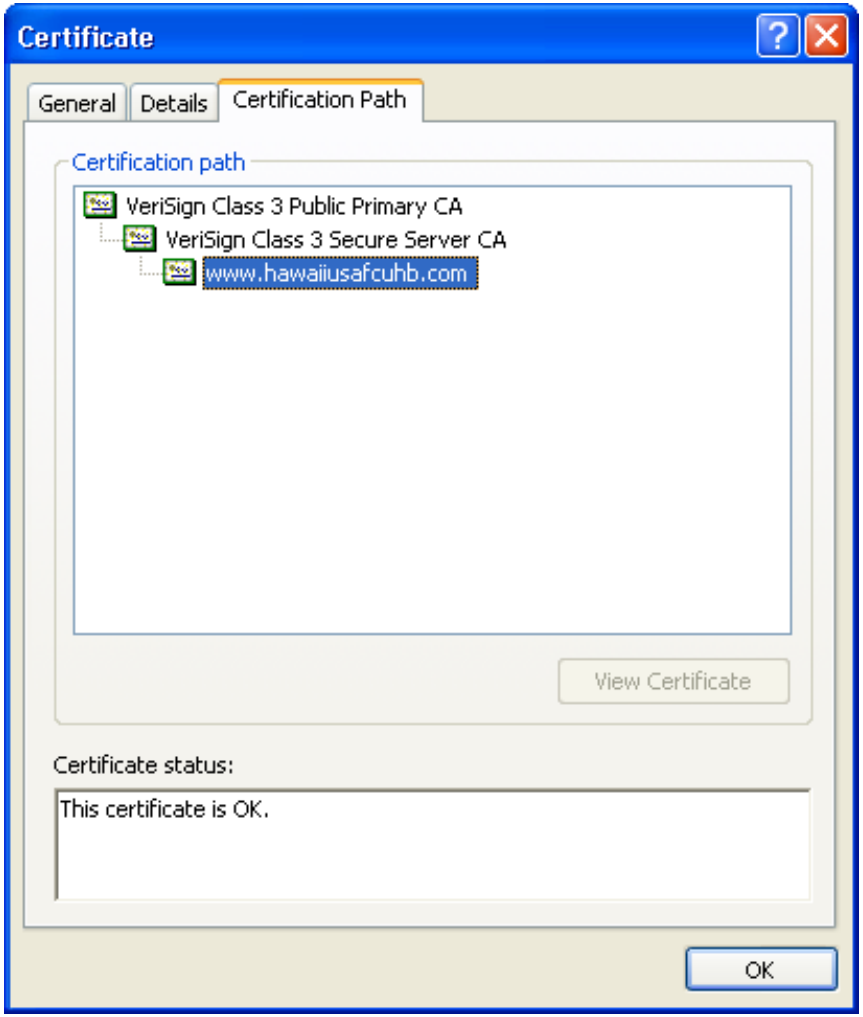


Fake certificate

Certificate comparison 3

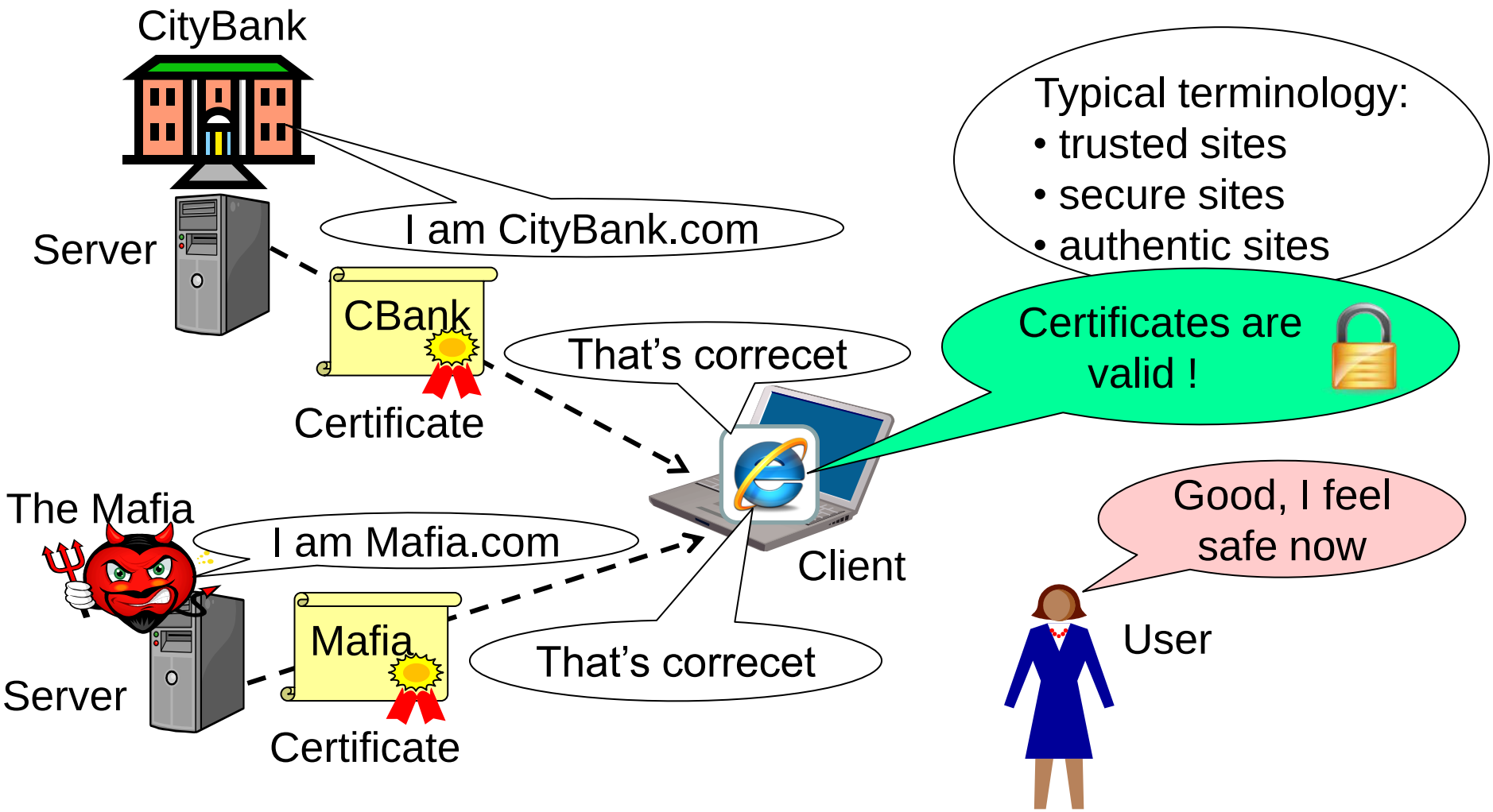


Genuine certificate

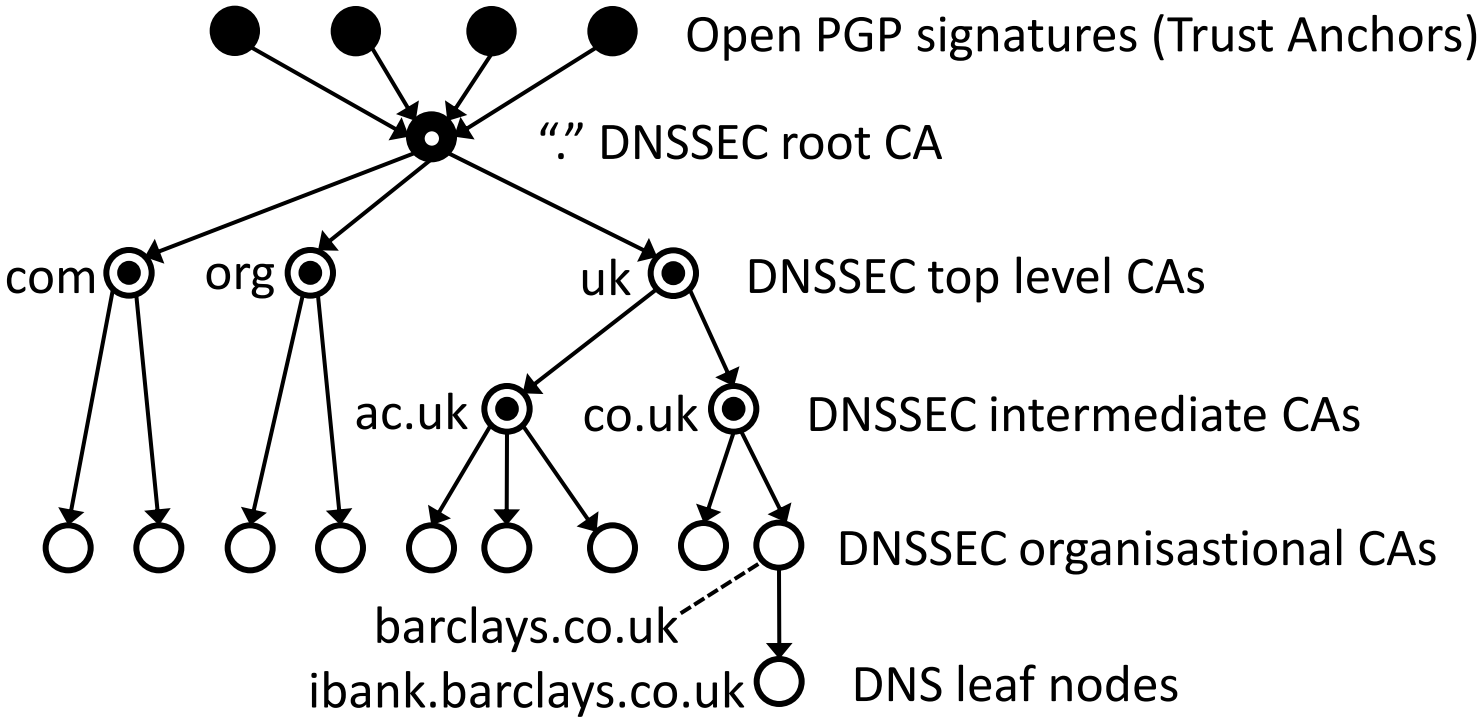


Fake certificate

Meaningless PKIX Server Authentication

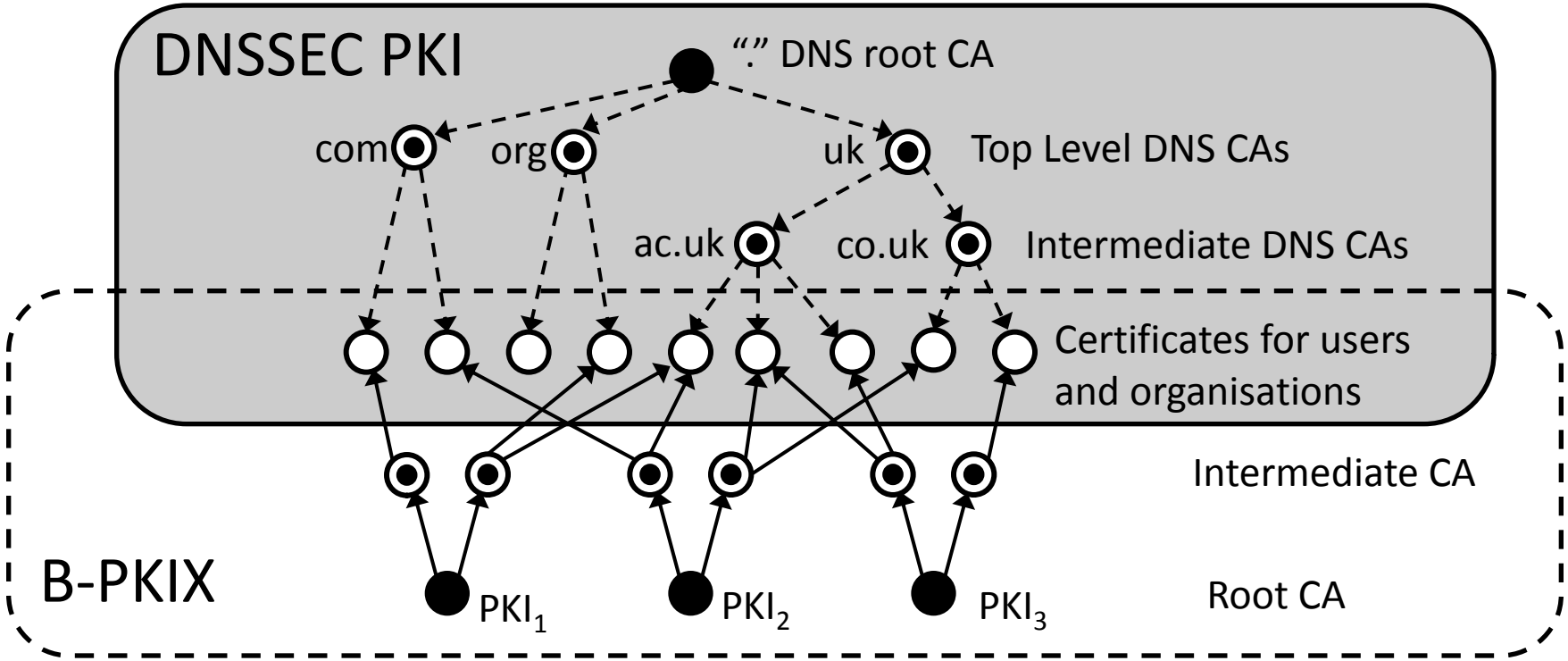


DNSSEC PKI



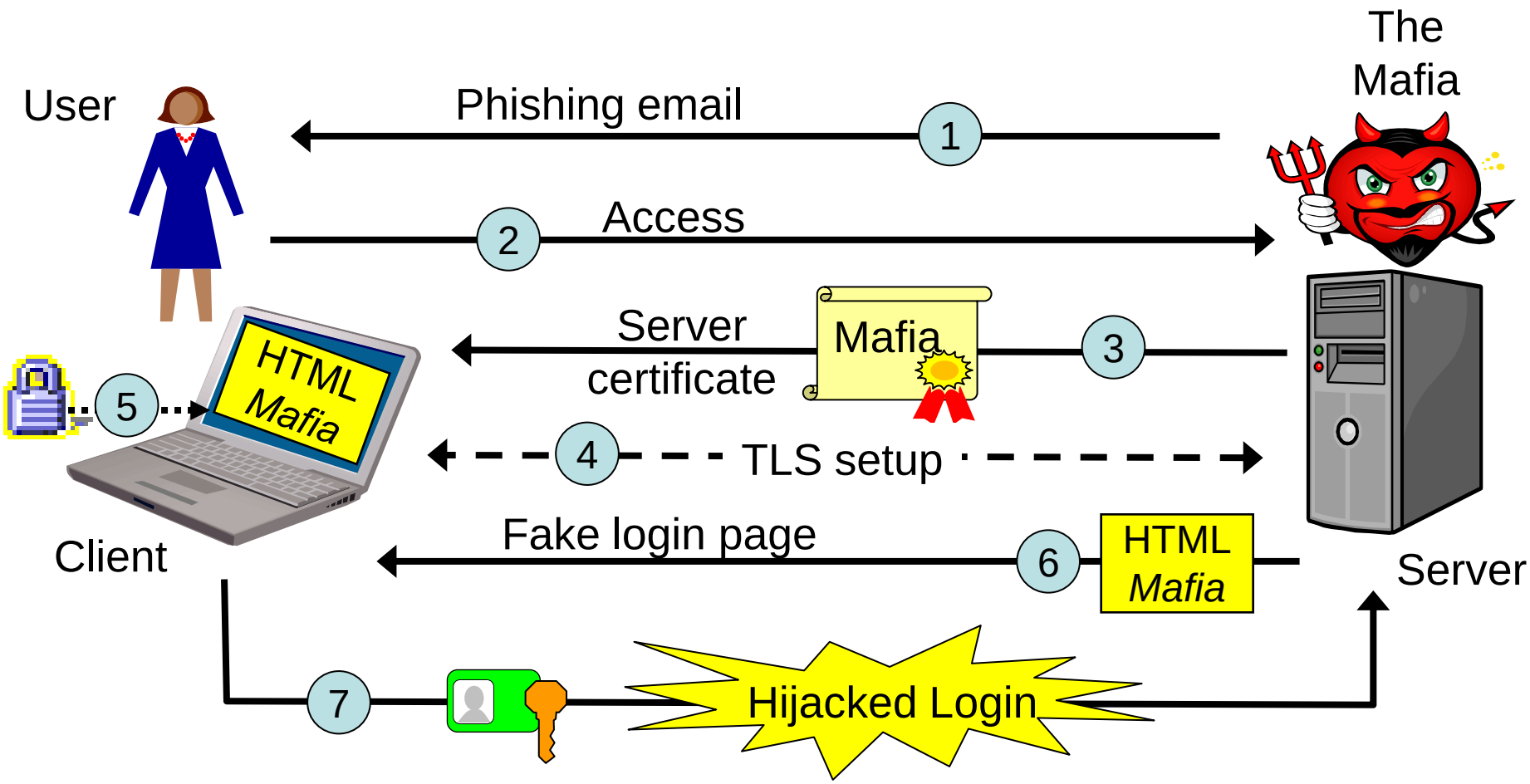
- The DNS (Domain Name System) is vulnerable to e.g. cache poisoning attacks resulting in wrong IP addresses being returned.
- DNSSEC designed to provide digital signature on every DNS reply
- Based on PKI with a single root.

DNSSEC PKI vs. Browser PKIX

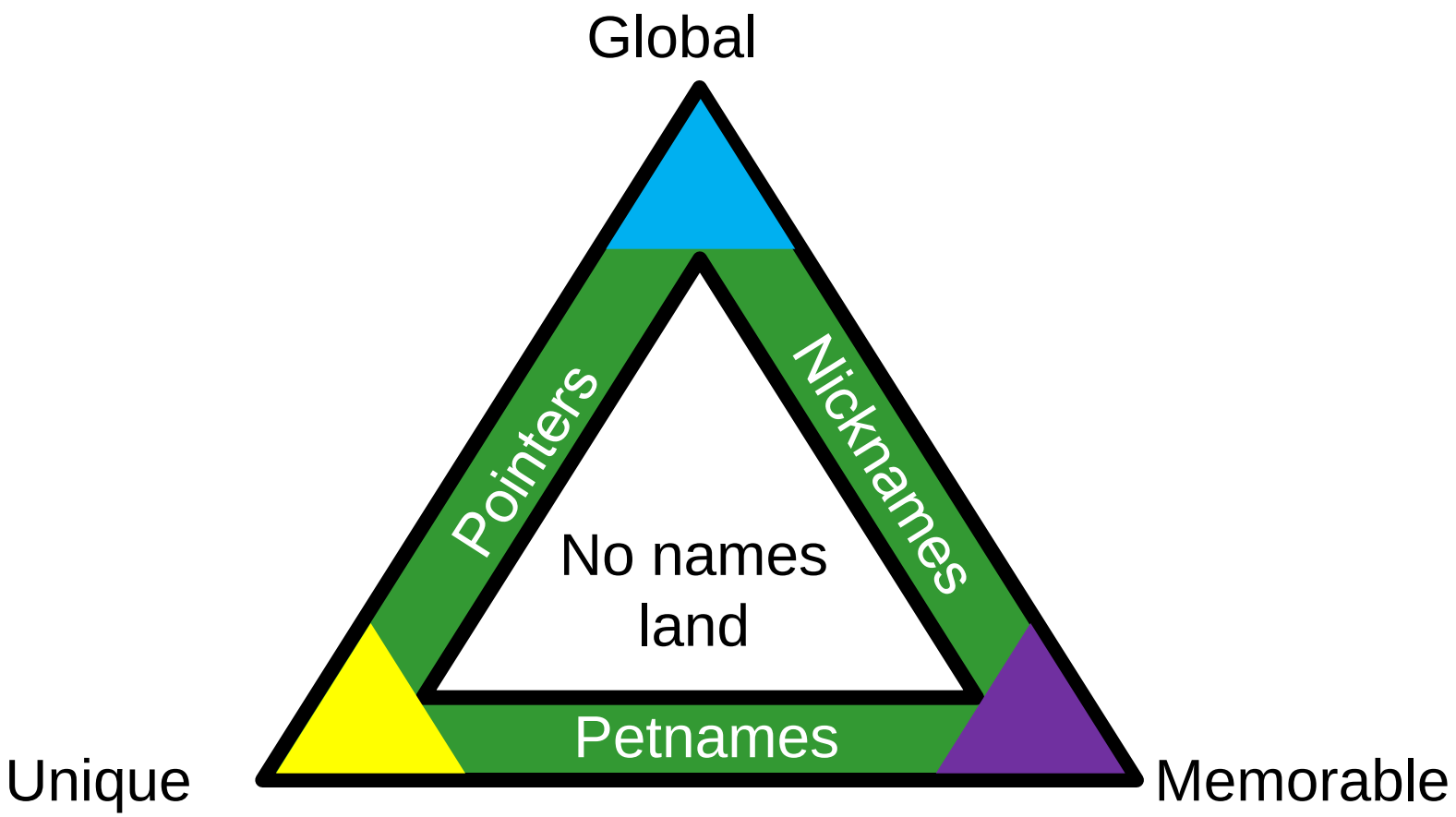


- In B-PKIX, any CA can issue certs for any domain → problematic
- CAs under the DNSSEC PKI can only issue certificates for own domain
- The DNSSEC PKI and the B-PKI both target the same user/org nodes
- DANE: DNSSEC-based Authentication of Named Entities
  - Alternative to B-PKIX, standards exist, not deployed, complex

Phishing and failed authentication

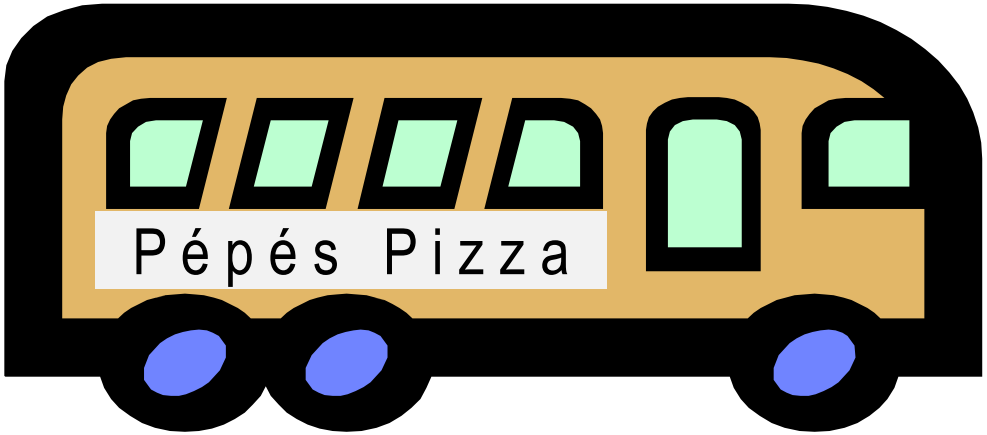


Zooko's Triangle of Id Properties



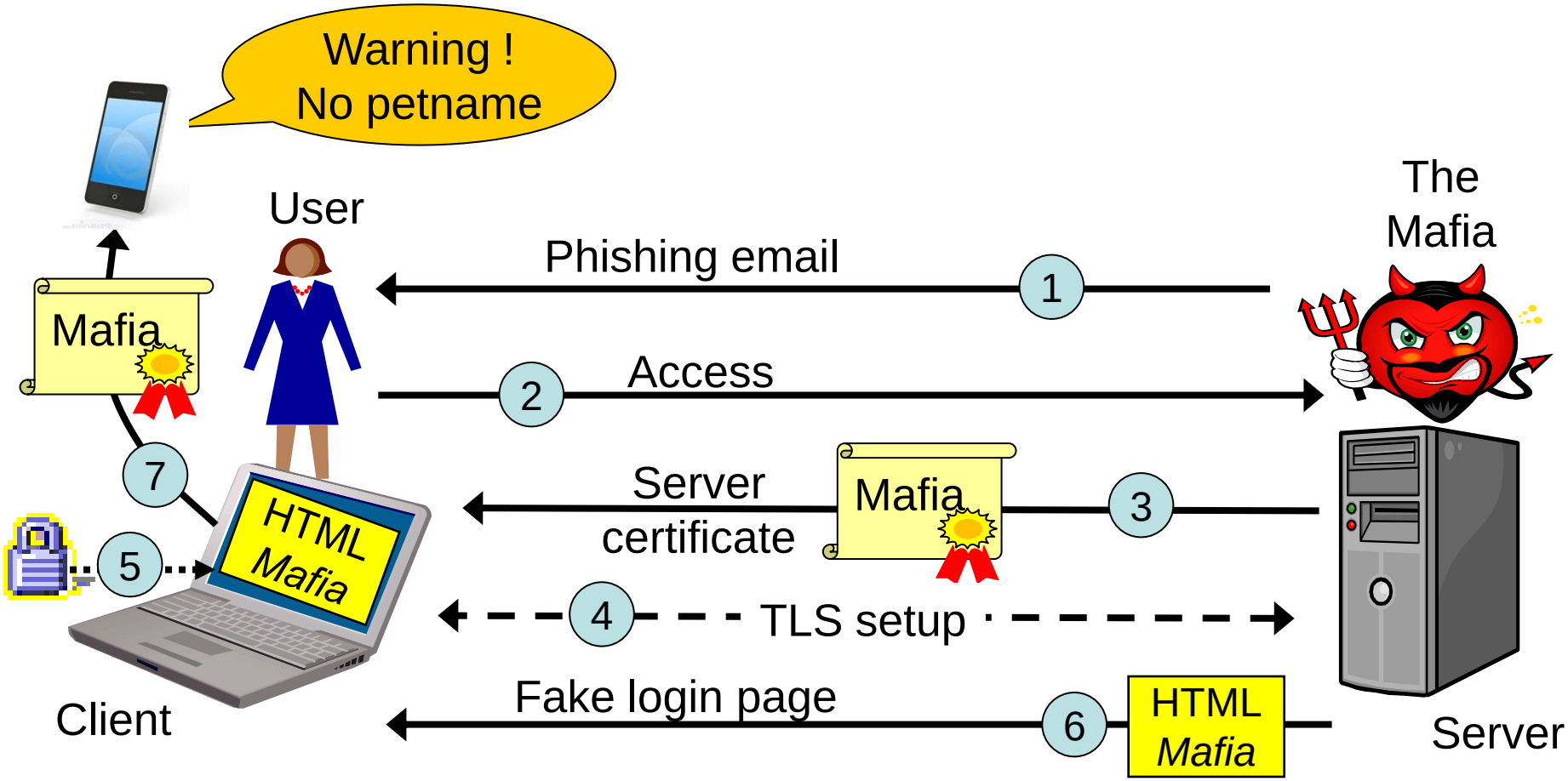
No identifier can be at the same time global, unique and memorable

Passing bus test for memorability

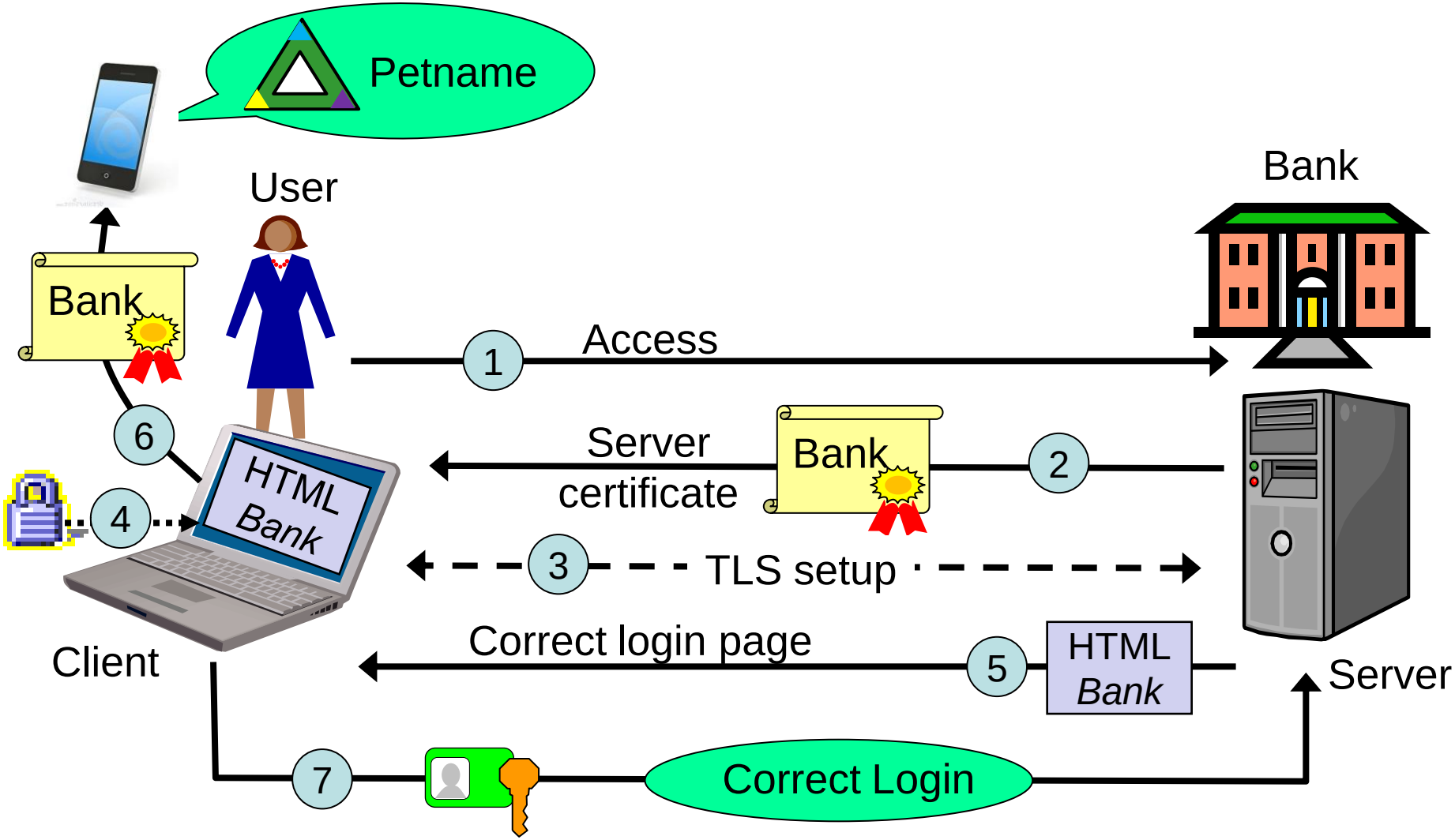


- If you see a name written on a passing bus, and you can remember the name after 5 minutes, then the name is memorable

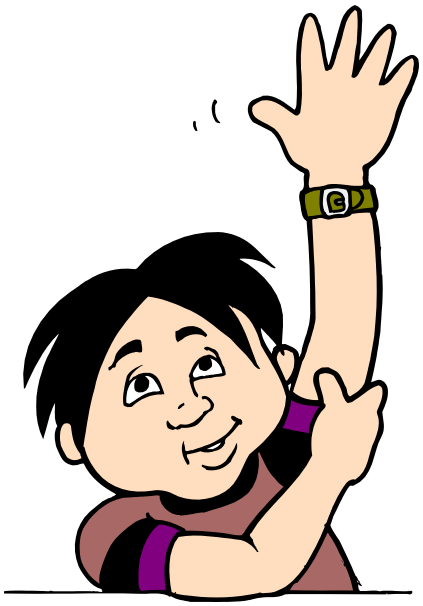
Phishing detection with user-centric IdMan



User-centric server authentication



Thank you for your attention



Questions?