COMP 307
Principles
of Web
Development

J. Vybihal

MCGILL UNIVERSITY

COMP 307
Principles of Web Development

Web Security #2

Readings

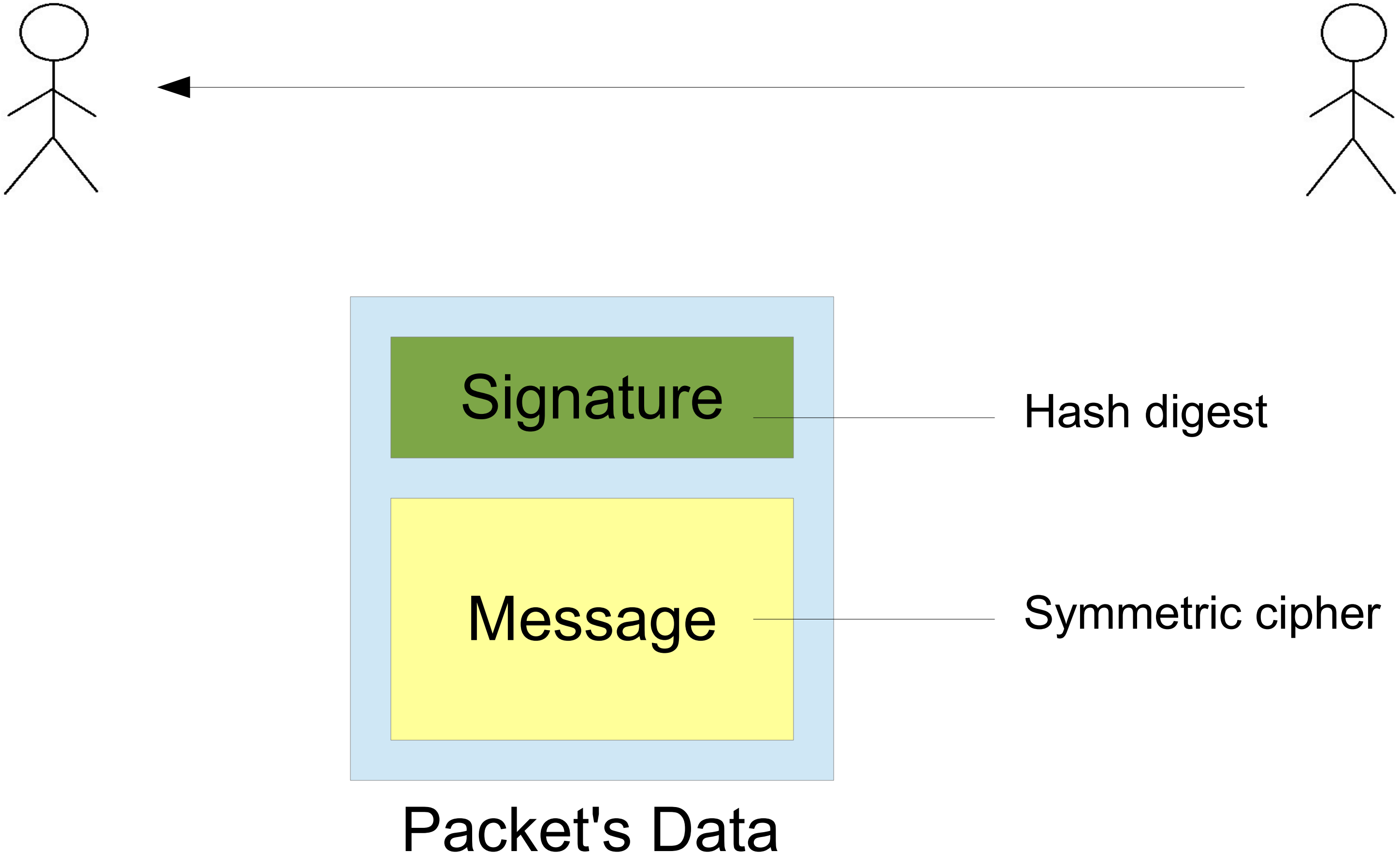
- Readings
  - Wikipedia:
    - Public Key Infrastructure
    - HTTP_Secure
- Experiment:
  - GnuPT + WinPT
    - Info: http://windowsitpro.com/security/winpt-and-gnupg
  - JavaScript easy plugin
    - http://www.jcryption.org/
  - Router security settings (at home)

Class Outline

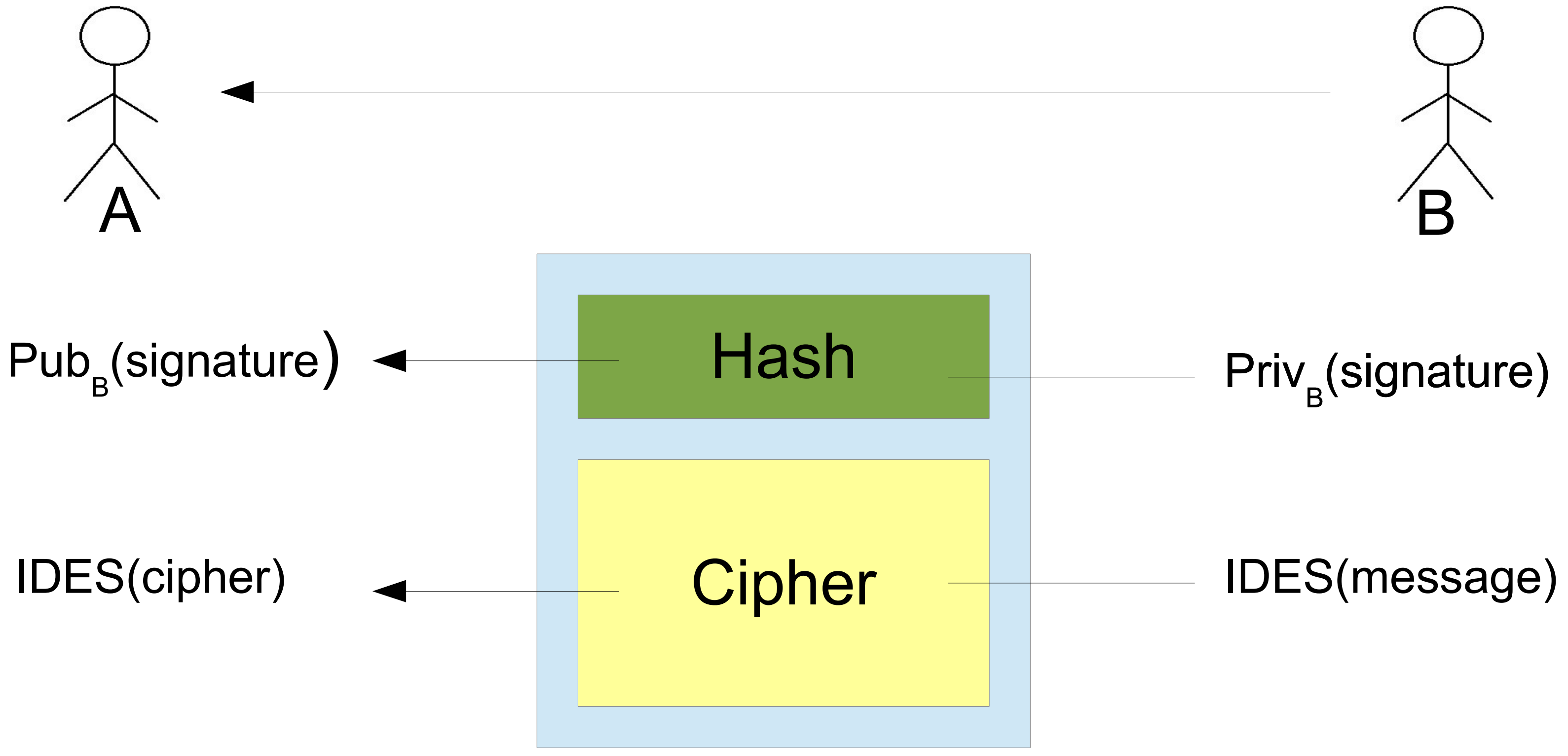
- Public Key Infrastructure
- Technologies
  - The security stack
  - Programming

General Security Technology

Messaging with Ciphers



Messaging with Ciphers

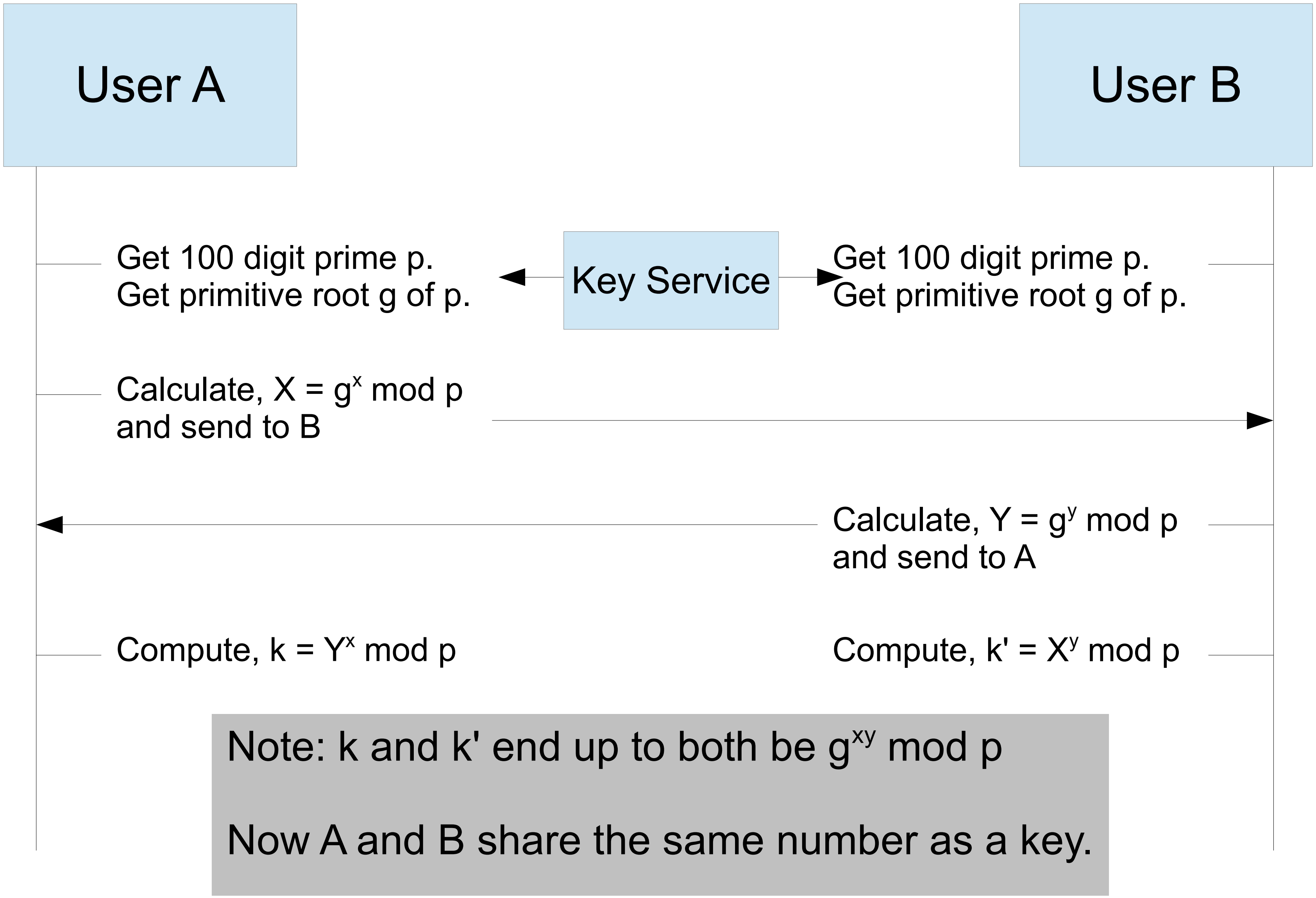


Data

B encrypt B's public key with B's private key. Then A uses B's public key to decrypt signature to find B's public key number, which A already has and can validate.

COMP 307 Principles of Web Development

Determine Secret / Public Keys



User A

User B

Key Service

Get 100 digit prime p.
Get primitive root g of p.

Get 100 digit prime p.
Get primitive root g of p.

Calculate, $X = g^x \bmod p$
and send to B

Calculate, $Y = g^y \bmod p$
and send to A

Compute, $k = Y^x \bmod p$

Compute, $k' = X^y \bmod p$

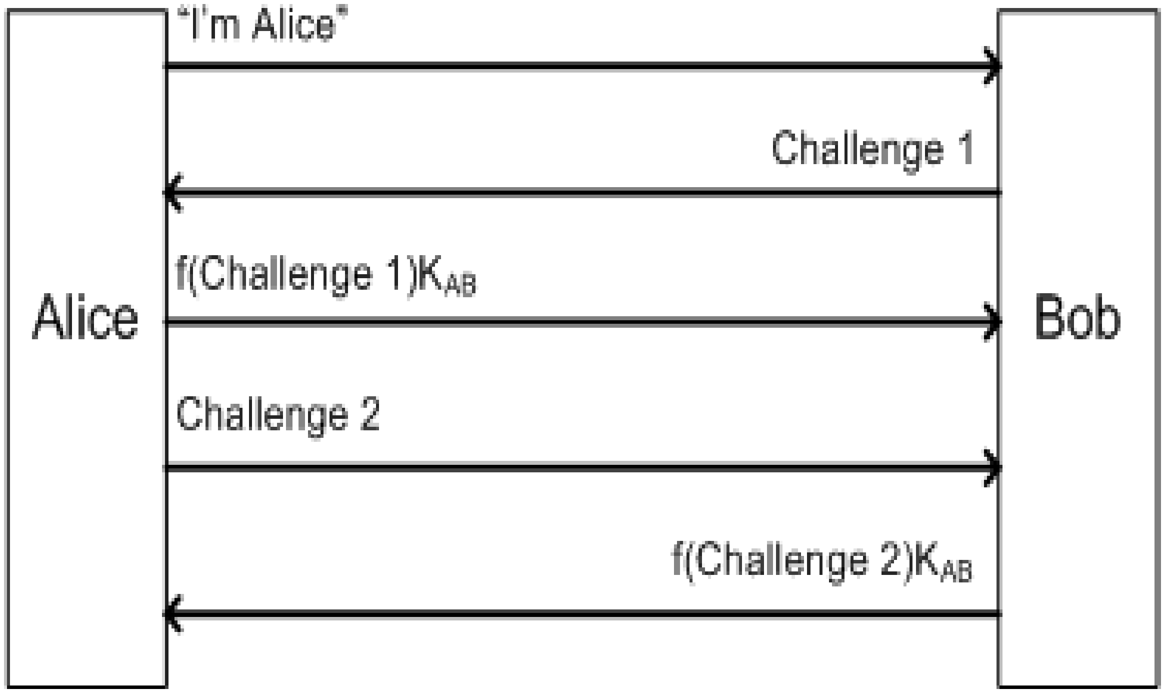
Note: k and k' end up to both be $g^{xy} \bmod p$

Now A and B share the same number as a key.

COMP 307
Principles
of Web
Development

Cryptographic Authentication

The Challenge-Response Technique

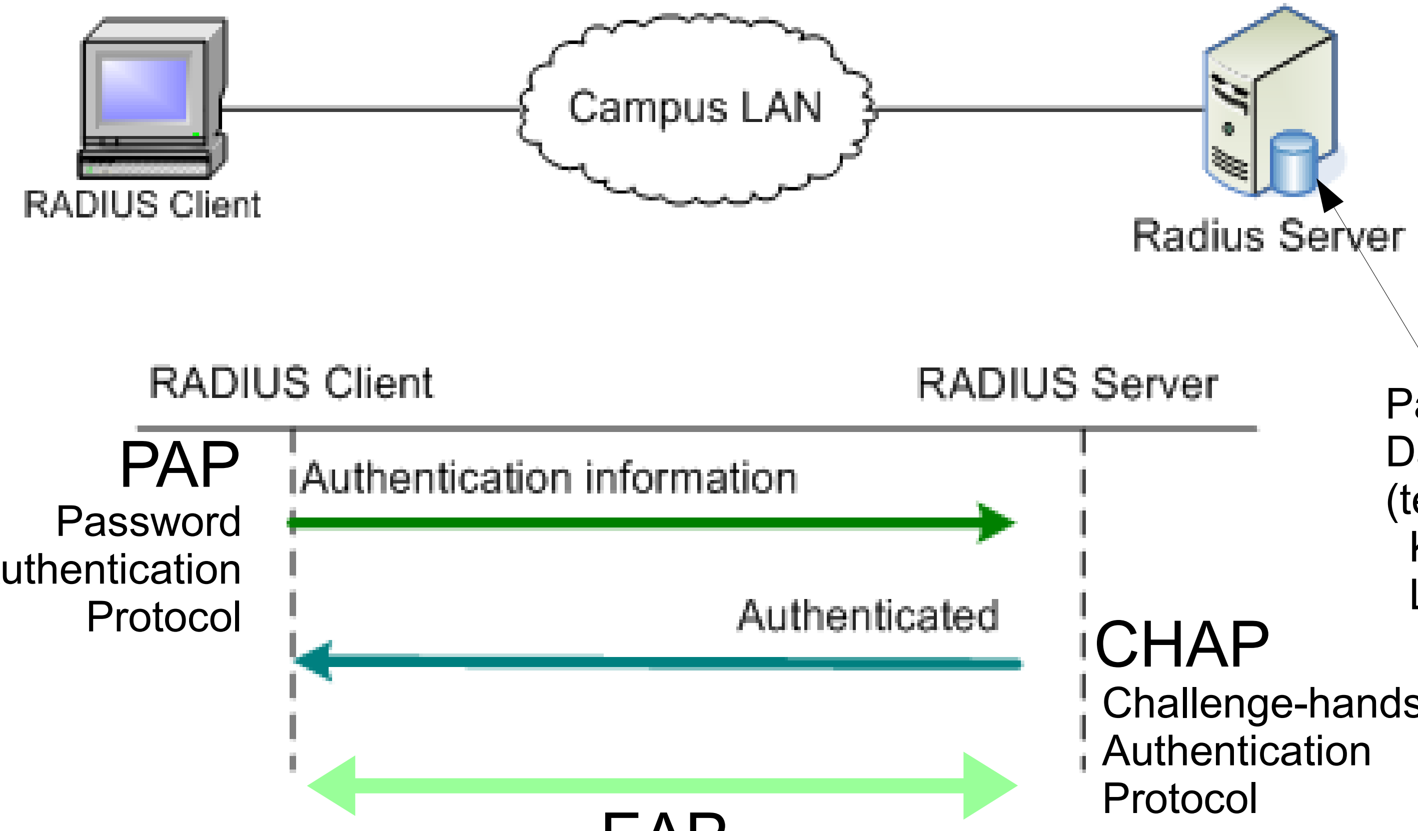


J. Vybihal

General Authentication Techniques

RADIUS Protocol

Remote Authentication Dial-in User Services



Password Database (text,SQL, Kerberos, LDAP,etc)

PAP
Password Authentication Protocol

CHAP
Challenge-handshake Authentication Protocol

EAP

Extensible Authentication Protocol
(Framework)
(Whatever standard method used for communication)

Elements of RADIUS

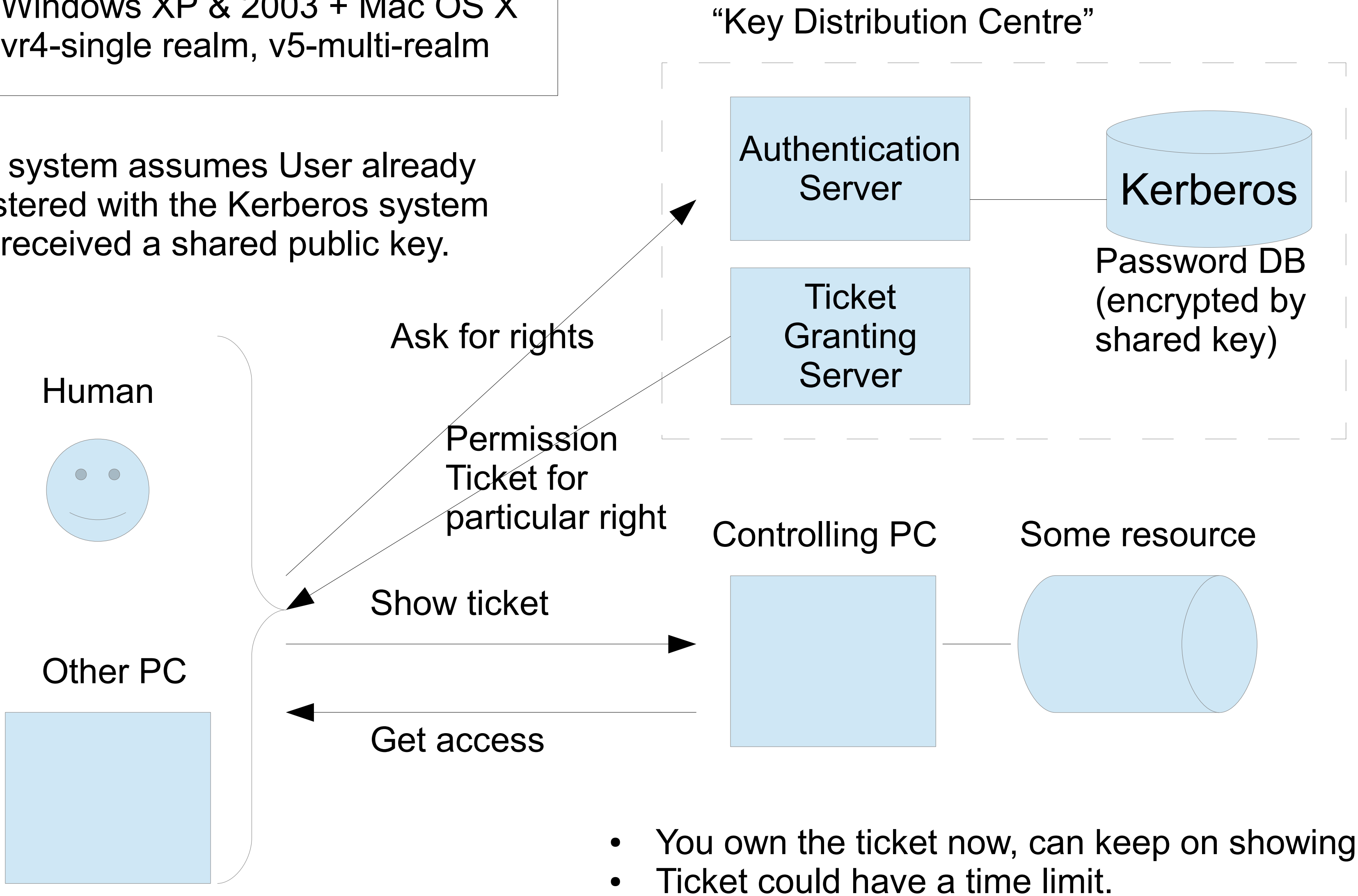
- PAP
  - Password login between client and server
- CHAP
  - Verification through challenges
    1. Server challenges client
    2. Client uses MD5 with password to server
    3. Server verifies info:
       - If passes then connection continues
       - If fails then connection terminates
  - CHAPS does this at login & random times

COMP 307 Principles of Web Development

Kerberos Authentication

MIT 1980's – Project Athena
Windows XP & 2003 + Mac OS X
vr4-single realm, v5-multi-realm

This system assumes User already registered with the Kerberos system and received a shared public key.



- You own the ticket now, can keep on showing it.
- Ticket could have a time limit.

Kerberos Problems

- System Load
  - When many users, need to have multiple Kerberos systems so that the servers are not overloaded by user requests for authentication or tickets. Each system is known as a Realm.
- “High-value” Target
  - If the Kerberos system goes “down” / “blocked” then no one has access to the resources.
  - If Kerberos systems is “breached” then you have access to everything.

COMP 307
Principles
of Web
Development

Kerberos Problems

- DES symmetric encryption
  - Obsolete due to modern computing power
- Applications
  - Must support Kerberos communication
- IPv4 Based
  - Today we are migrating to IPv6 addresses
- Key Distribution Center
  - Must always be network reachable

COMP 307
Principles
of Web
Development

Kerberos Benefits

- Reduces number of keys
  - Just shared public key 1 per user
  - Plus the user's encrypted password
- Session control
  - User logs in only once per session
  - And, only one ticket per type of resource
    - Does not require user's password to be transmitted across network (uses ticket)

Public Key Infrastructure

Keys
Certificates
Validation Process
Network Hardware and Software

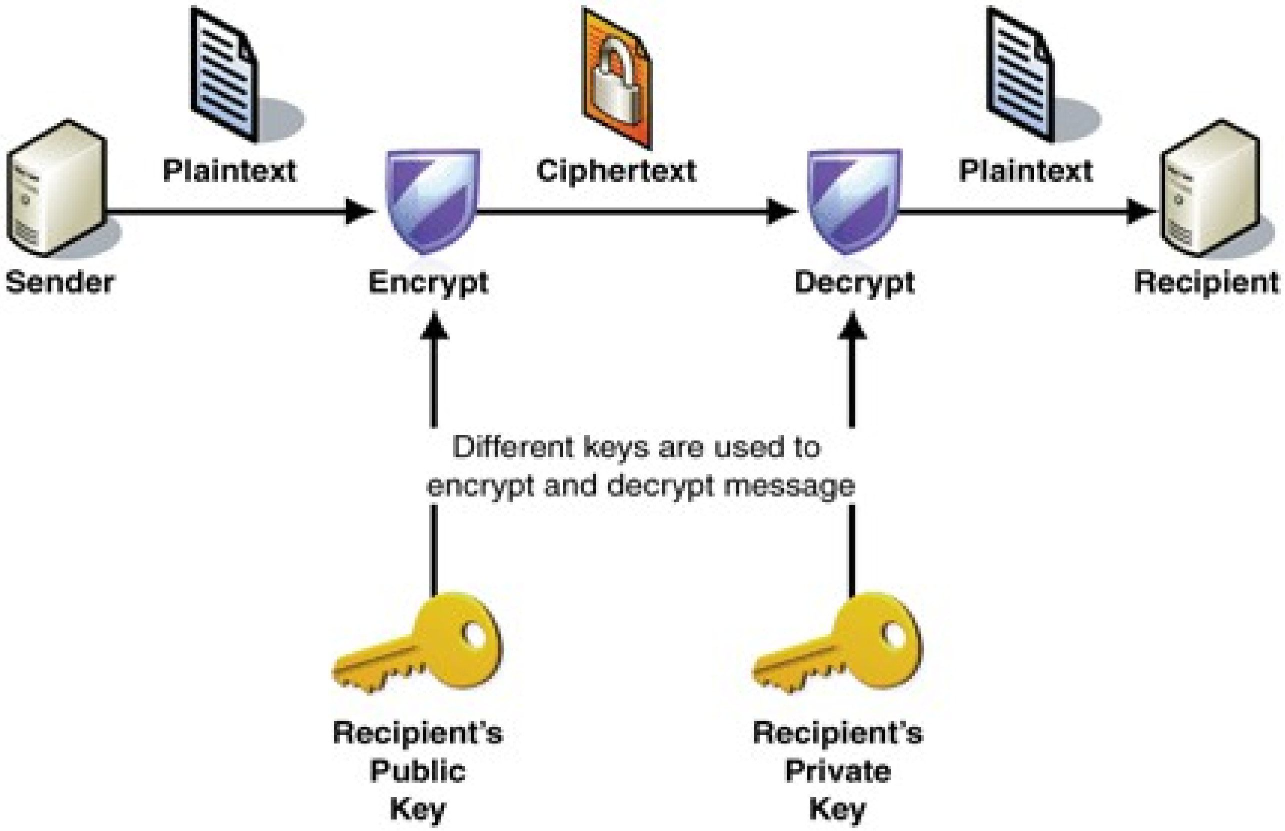
COMP 307
Principles
of Web
Development

Public-Key Infrastructure:
#1
About Keys

COMP 307
Principles
of Web
Development

J. Vybihal

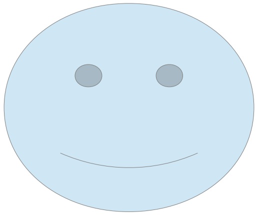
Asymmetric Encryption



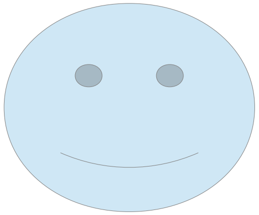
http://alwajbaiss.com/wp-content/uploads/2011/03/IC21919.gif

Remember: private / public keys are related and can be interchanged in Encryption/Decryption algorithms.

Create Keys

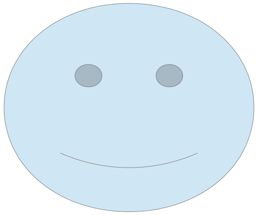


User 1
Private key 1
Public key 1

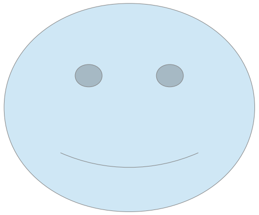


User 2
Private key 2
Public key 2

Share Keys



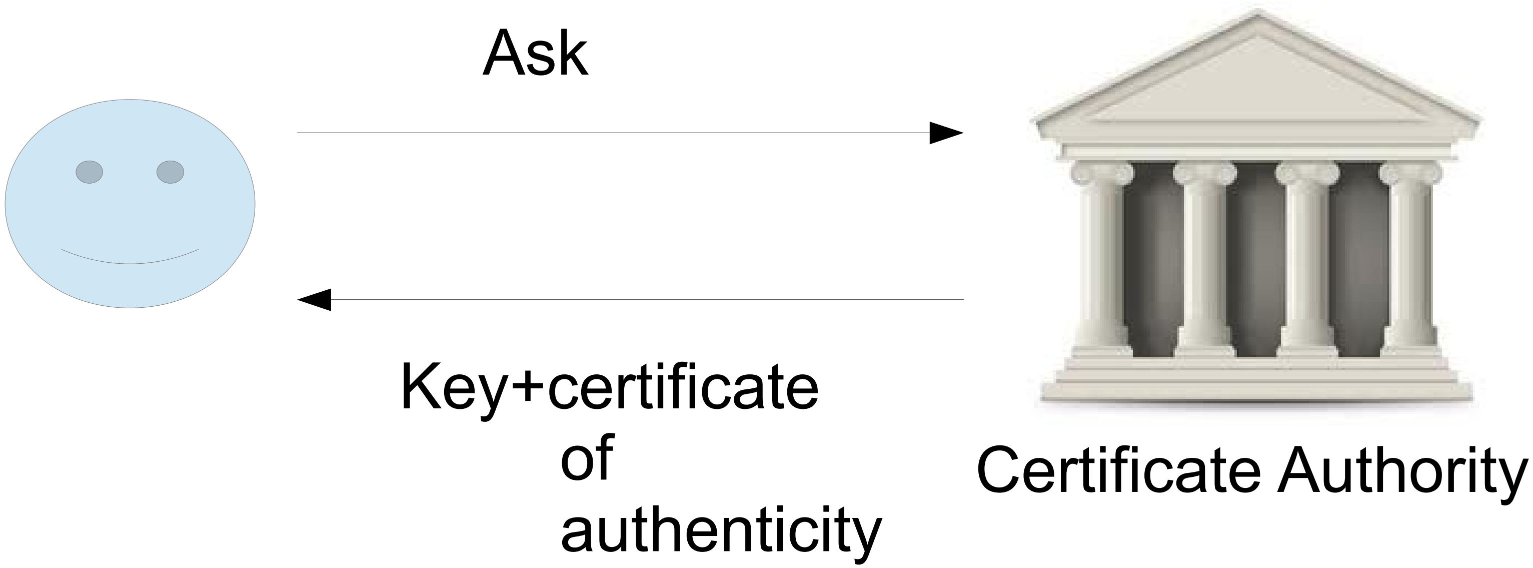
User 1
Private key 1
Public key 1
Public key 2



User 2
Private key 2
Public key 2
Public key 1

COMP 307
Principles
of Web
Development

Where do we get keys?



Ask

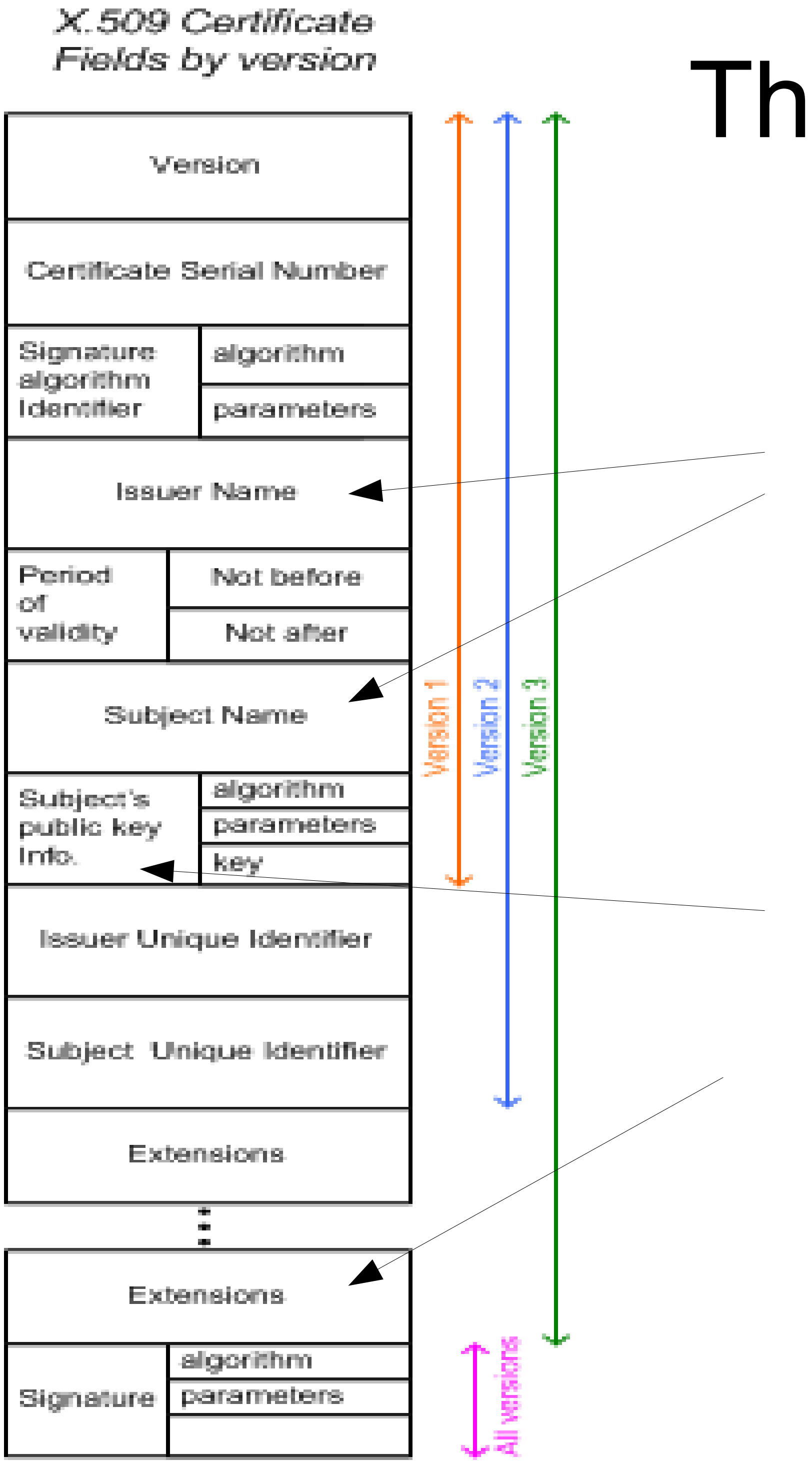
Key+certificate
of
authenticity

Certificate Authority

** Or it can be generated locally. **
(not guaranteed unique!)

Public Key Infrastructure
#2
About Certificates

The X.509 Certificate



X.509 Certificate Fields by version

Version

Certificate Serial Number

Signature algorithm Identifier | algorithm | parameters

Issuer Name

Period of validity | Not before | Not after

Subject Name

Subject's public key Info. | algorithm | parameters | key

Issuer Unique Identifier

Subject Unique Identifier

Extensions

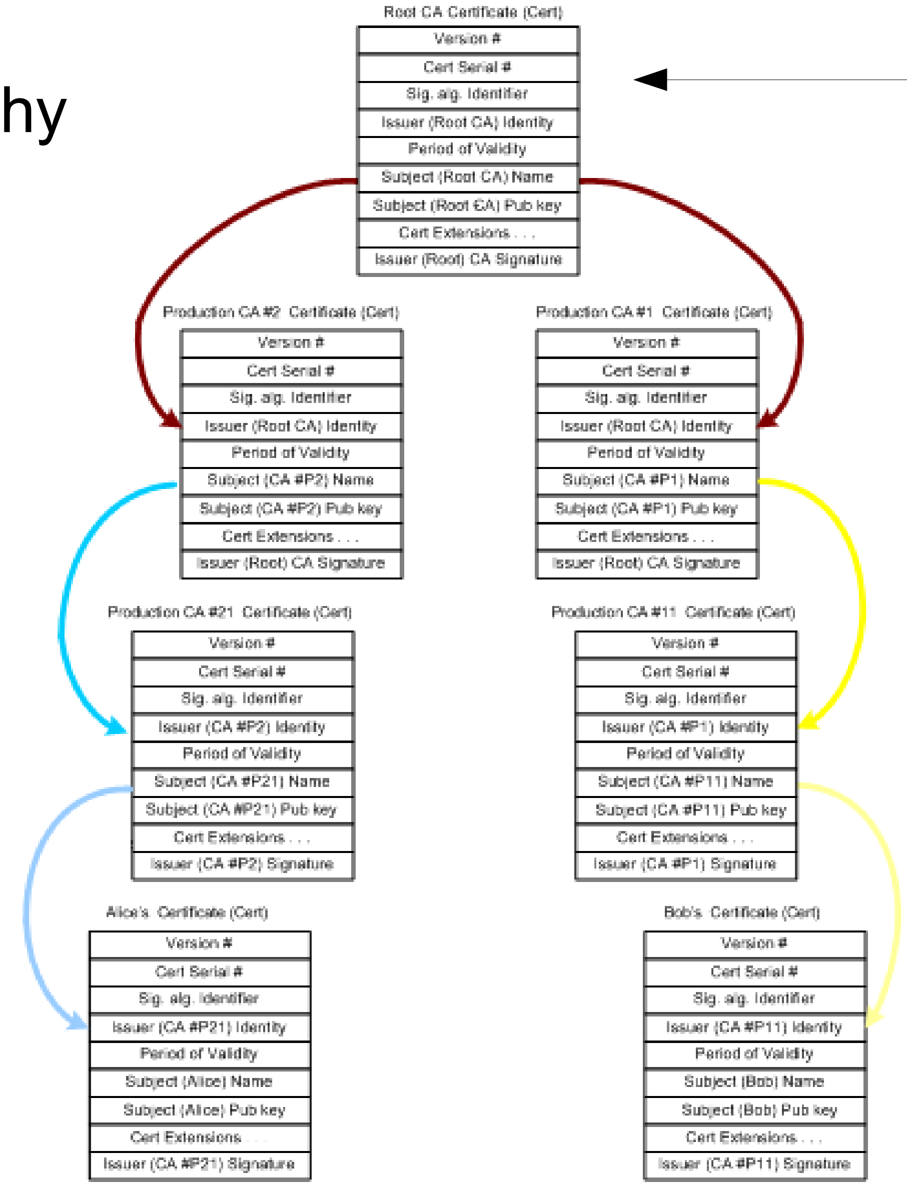
Extensions

Signature | algorithm | parameters

Used to ensure the key is valid and original

Certificate Signing Tree

Trust hierarchy



Root CA Certificate (Cert)
Version #
Cert Serial #
Sig. alg. Identifier
Issuer (Root CA) Identity
Period of Validity
Subject (Root CA) Name
Subject (Root CA) Pub key
Cert Extensions . . .
Issuer (Root) CA Signature

Production CA #2 Certificate (Cert)
Version #
Cert Serial #
Sig. alg. Identifier
Issuer (Root CA) Identity
Period of Validity
Subject (CA #P2) Name
Subject (CA #P2) Pub key
Cert Extensions . . .
Issuer (Root) CA Signature

Production CA #1 Certificate (Cert)
Version #
Cert Serial #
Sig. alg. Identifier
Issuer (Root CA) Identity
Period of Validity
Subject (CA #P1) Name
Subject (CA #P1) Pub key
Cert Extensions . . .
Issuer (Root) CA Signature

Production CA #21 Certificate (Cert)
Version #
Cert Serial #
Sig. alg. Identifier
Issuer (CA #P2) Identity
Period of Validity
Subject (CA #P21) Name
Subject (CA #P21) Pub key
Cert Extensions . . .
Issuer (CA #P2) Signature

Production CA #11 Certificate (Cert)
Version #
Cert Serial #
Sig. alg. Identifier
Issuer (CA #P1) Identity
Period of Validity
Subject (CA #P11) Name
Subject (CA #P11) Pub key
Cert Extensions . . .
Issuer (CA #P1) Signature

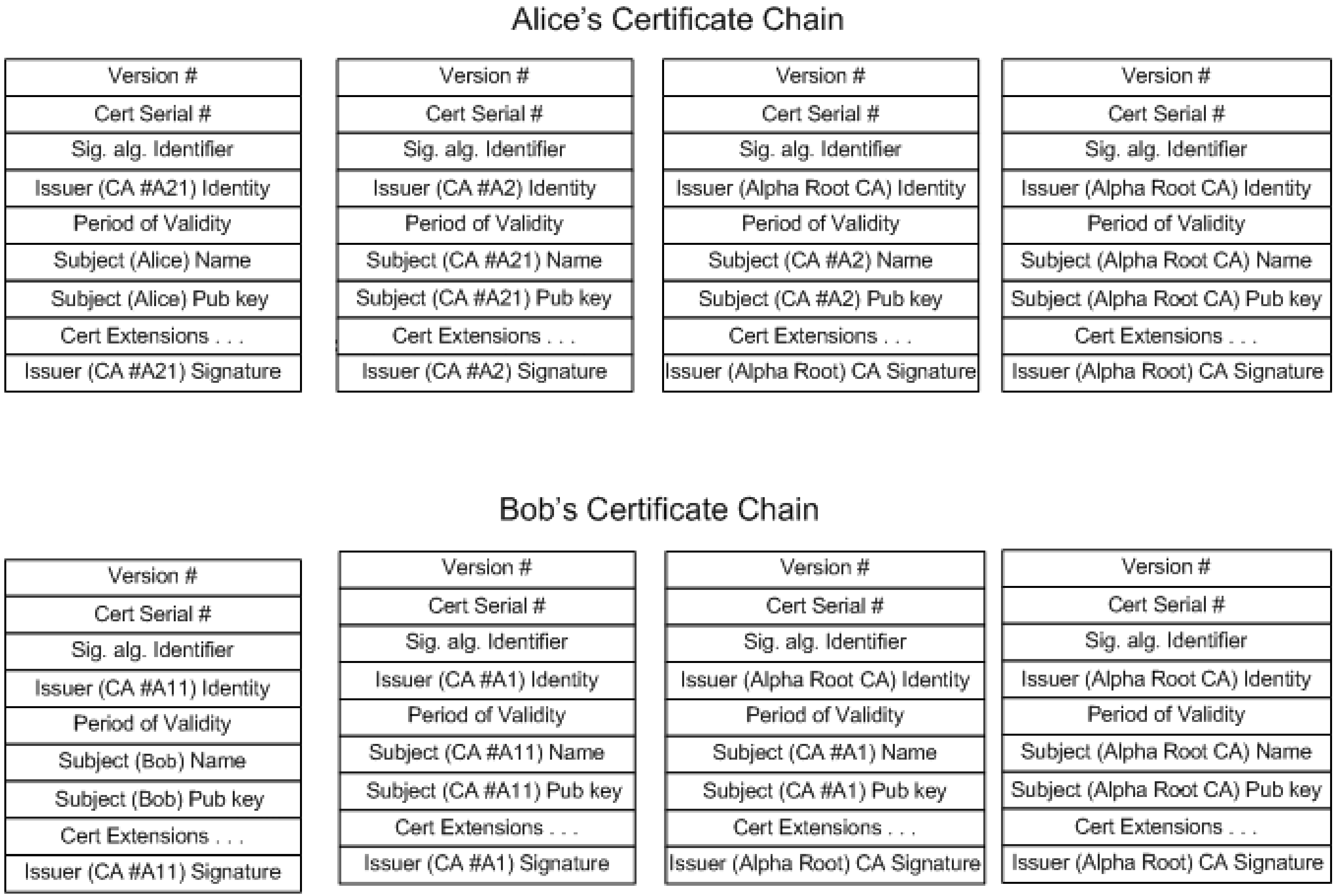
Alice's Certificate (Cert)
Version #
Cert Serial #
Sig. alg. Identifier
Issuer (CA #P21) Identity
Period of Validity
Subject (Alice) Name
Subject (Alice) Pub key
Cert Extensions . . .
Issuer (CA #P21) Signature

Bob's Certificate (Cert)
Version #
Cert Serial #
Sig. alg. Identifier
Issuer (CA #P11) Identity
Period of Validity
Subject (Bob) Name
Subject (Bob) Pub key
Cert Extensions . . .
Issuer (CA #P11) Signature

Trusted root

Appending keys to keep uniqueness

The Certificate Chain



Alice's Certificate Chain

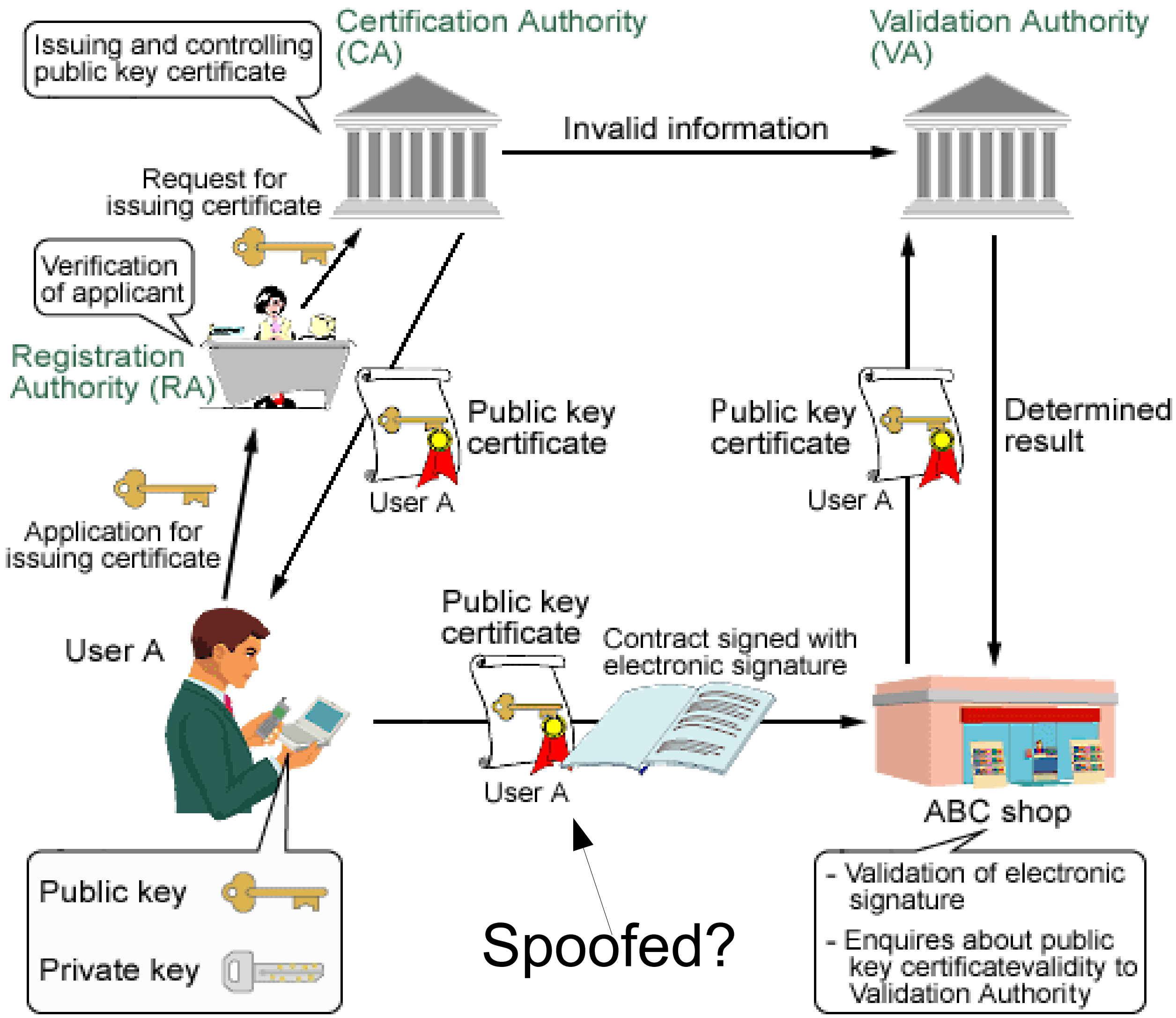
Version #	Version #	Version #	Version #
Cert Serial #	Cert Serial #	Cert Serial #	Cert Serial #
Sig. alg. Identifier	Sig. alg. Identifier	Sig. alg. Identifier	Sig. alg. Identifier
Issuer (CA #A21) Identity	Issuer (CA #A2) Identity	Issuer (Alpha Root CA) Identity	Issuer (Alpha Root CA) Identity
Period of Validity	Period of Validity	Period of Validity	Period of Validity
Subject (Alice) Name	Subject (CA #A21) Name	Subject (CA #A2) Name	Subject (Alpha Root CA) Name
Subject (Alice) Pub key	Subject (CA #A21) Pub key	Subject (CA #A2) Pub key	Subject (Alpha Root CA) Pub key
Cert Extensions . . .	Cert Extensions . . .	Cert Extensions . . .	Cert Extensions . . .
Issuer (CA #A21) Signature	Issuer (CA #A2) Signature	Issuer (Alpha Root) CA Signature	Issuer (Alpha Root) CA Signature

Bob's Certificate Chain

Version #	Version #	Version #	Version #
Cert Serial #	Cert Serial #	Cert Serial #	Cert Serial #
Sig. alg. Identifier	Sig. alg. Identifier	Sig. alg. Identifier	Sig. alg. Identifier
Issuer (CA #A11) Identity	Issuer (CA #A1) Identity	Issuer (Alpha Root CA) Identity	Issuer (Alpha Root CA) Identity
Period of Validity	Period of Validity	Period of Validity	Period of Validity
Subject (Bob) Name	Subject (CA #A11) Name	Subject (CA #A1) Name	Subject (Alpha Root CA) Name
Subject (Bob) Pub key	Subject (CA #A11) Pub key	Subject (CA #A1) Pub key	Subject (Alpha Root CA) Pub key
Cert Extensions . . .	Cert Extensions . . .	Cert Extensions . . .	Cert Extensions . . .
Issuer (CA #A11) Signature	Issuer (CA #A1) Signature	Issuer (Alpha Root) CA Signature	Issuer (Alpha Root) CA Signature

Public Key Infrastructure
#3
Validation Process

PKI Validation Process

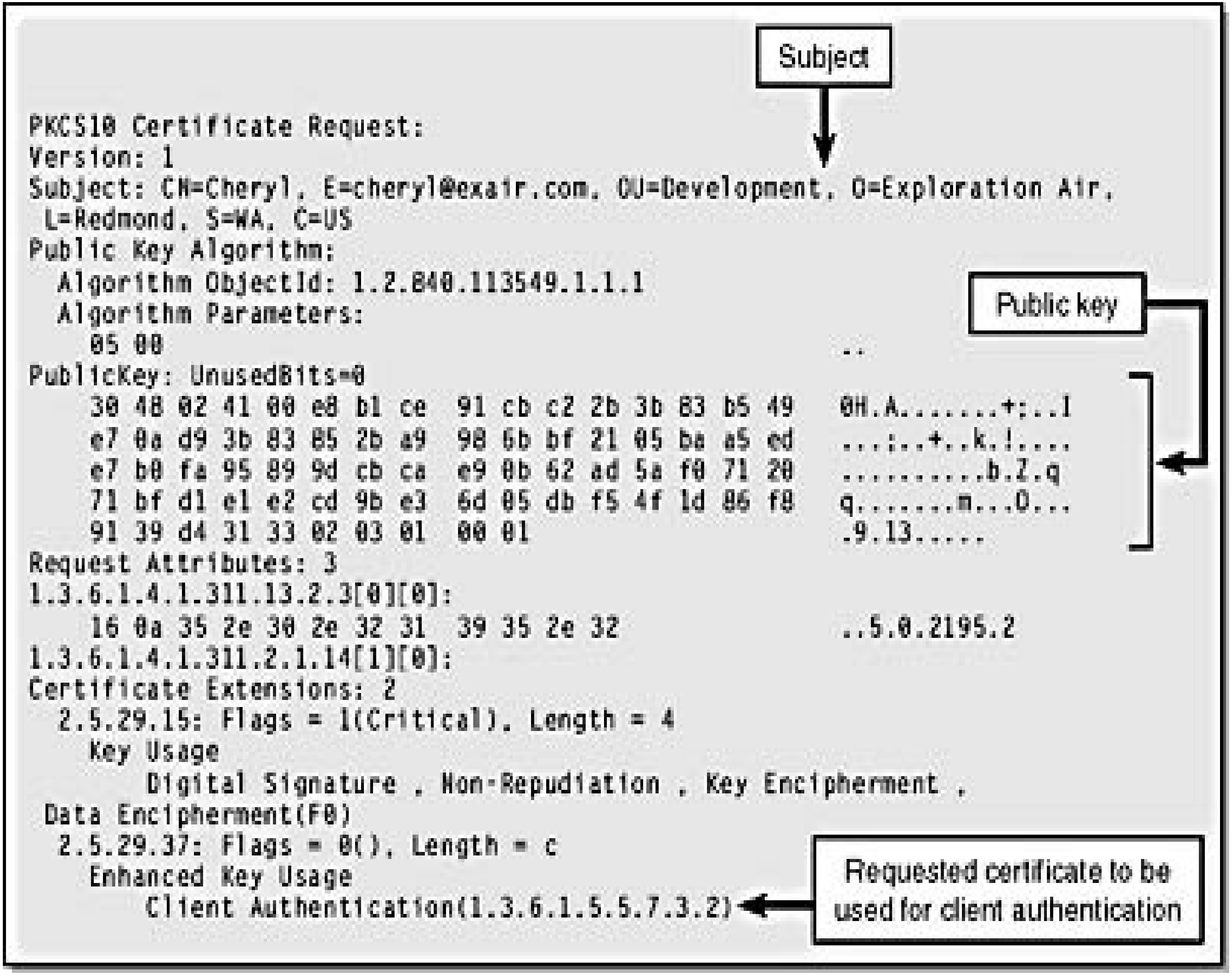

Issuing and controlling public key certificate
Certification Authority (CA)
Validation Authority (VA)
Invalid information
Request for issuing certificate
Verification of applicant
Registration Authority (RA)
Public key certificate
User A
Public key certificate
User A
Determined result
Application for issuing certificate
User A
Public key certificate
Contract signed with electronic signature
User A
ABC shop
Public key
Private key
- Validation of electronic signature
- Enquires about public key certificatevalidity to Validation Authority

Spoofed?

http://www.hitachi.com/rd/yrl/people/pki/img/image1.gif

Requesting a Certificate

- Step 1 – On sequesters workstation
  - Generates asymmetric public/private key pair
  - Private key encrypted using symmetric algorithm (AES) secret key using 128 bit MD-5 digest of a “passphrase”.
    - Clear-text private key erased
    - Black-text version stored in workstation
      - Called “private key ring”
  - Creates PKCS #10 message (ID, public key)
    - Clear-text public key erased from workstantion
  - PKCS #10 message
    - Encrypted with RA's public key
    - Clear-text version of PKCS #10 erased
    - Black-text version sent to RA



Subject

Public key

```
PKCS10 Certificate Request:
Version: 1
Subject: CN=Cheryl, E=cheryl@exair.com, OU=Development, O=Exploration Air,
 L=Redmond, S=WA, C=US
Public Key Algorithm:
  Algorithm ObjectId: 1.2.840.113549.1.1.1
  Algorithm Parameters:
    05 00                                                  ..
PublicKey: UnusedBits=0
    30 48 02 41 00 e8 b1 ce  91 cb c2 2b 3b 83 b5 49   0H.A.......+;..I
    e7 0a d9 3b 83 05 2b a9  98 6b bf 21 05 ba a5 ed   ...;..+..k.!....
    e7 b0 fa 95 89 9d cb ca  e9 0b 62 ad 5a f0 71 20   ..........b.Z.q
    71 bf d1 e1 e2 cd 9b e3  6d 05 db f5 4f 1d 06 f8   q.......m...O...
    91 39 d4 31 33 02 03 01  00 01                     .9.13.....
Request Attributes: 3
1.3.6.1.4.1.311.13.2.3[0][0]:
    16 0a 35 2e 30 2e 32 31  39 35 2e 32               ..5.0.2195.2
1.3.6.1.4.1.311.2.1.14[1][0]:
Certificate Extensions: 2
  2.5.29.15: Flags = 1(Critical), Length = 4
    Key Usage
        Digital Signature , Non-Repudiation , Key Encipherment ,
 Data Encipherment(F0)
  2.5.29.37: Flags = 0(), Length = c
    Enhanced Key Usage
        Client Authentication(1.3.6.1.5.5.7.3.2)
```

Requested certificate to be used for client authentication

Requesting a Certificate

- Step 2 – At RA
  - Decrypt received PKCS #10 with RA's private key
  - RA Administrator manually reviews the information telephoning the requester
  - If approved, the (ID, public key) is sent to the CA.

Requesting a Certificate

- Step 3 – At CA
  - Constructs the X.509 certificate from (ID,public key) information provided by RA
  - Using the X.509 and the CA's private key creates an encrypted digest (digest now viewed as it's digital signature)
  - Adds the digest into the X.509 certificate
  - CA sends certificate to RA

Requesting a Certificate

- Step 4 – At RA
  - Copy of created X.509 sent to Requester + the CA's personal X.509
  - RA posts a copy of the X.509 certificate into the LDAP database (Lightweight Directory Access Protocol) Server.

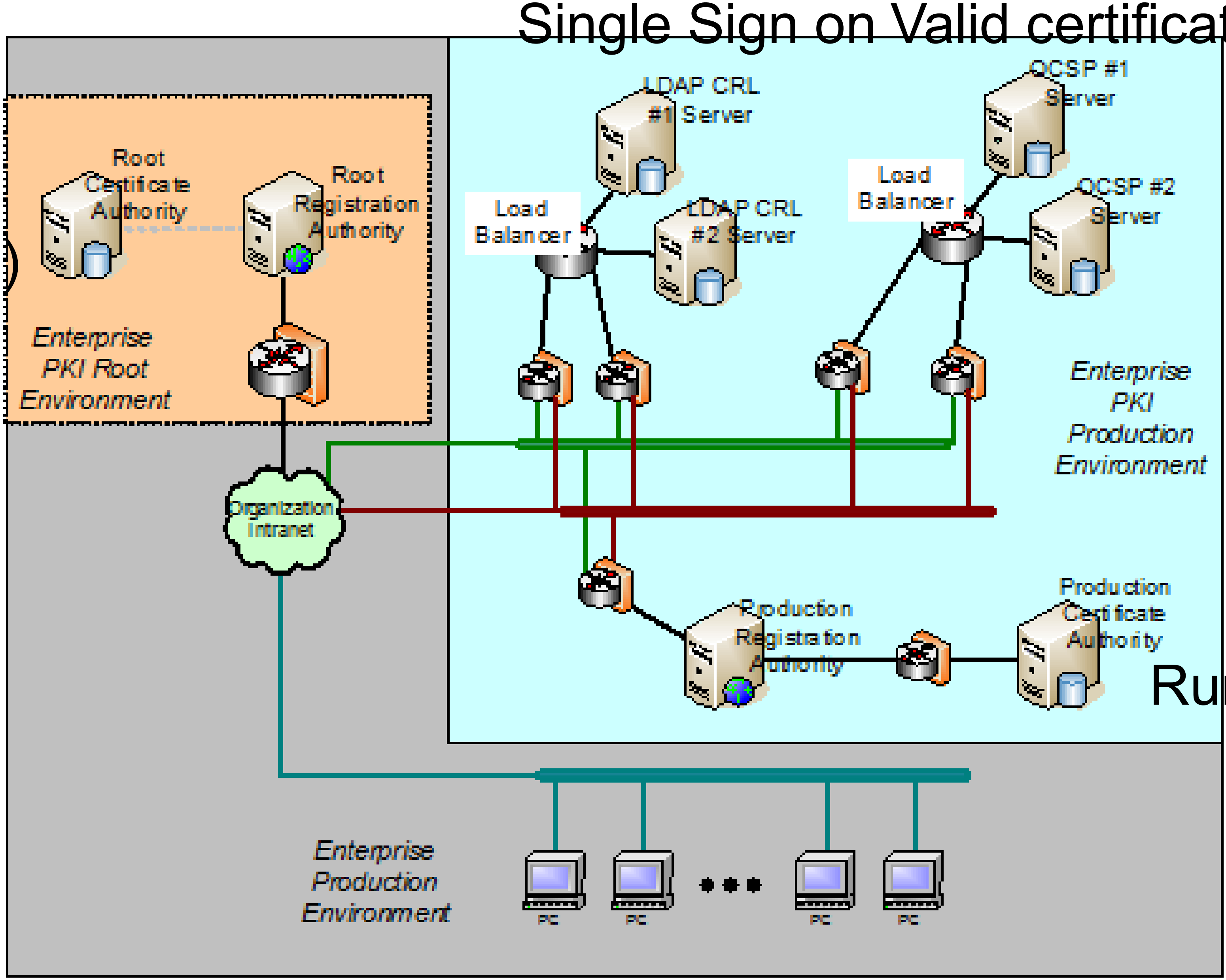
COMP 307
Principles
of Web
Development

Public Key Infrastructure
#4
Network Hardware and Software

Typical PKI Deployment

Single Sign on Valid certificate?

Turn
Off
(safe)



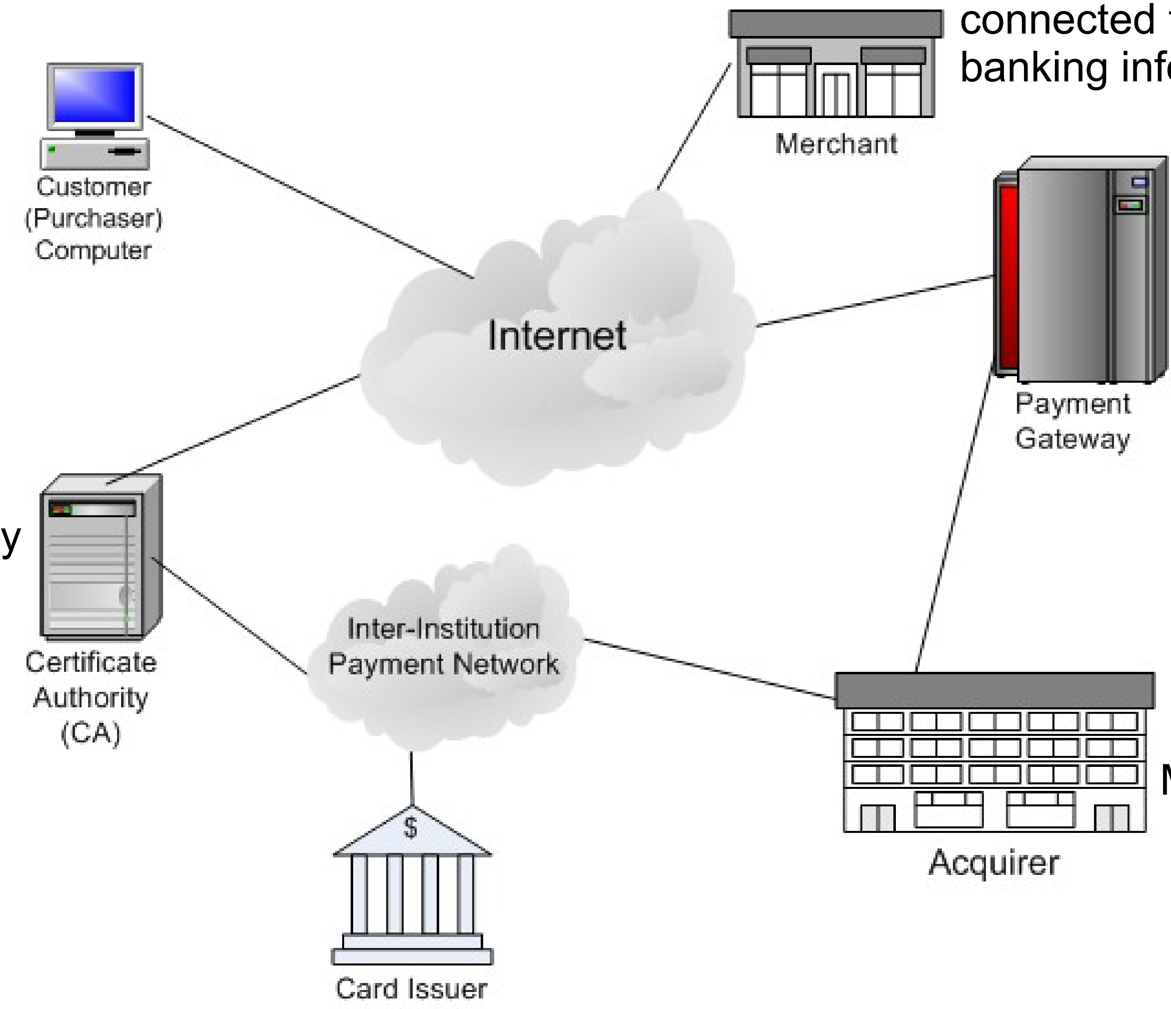
Running

LDAP: Lightweight Directory Access Protocol, OCSP: Online Certificate Status Protocol

Secure Electronic Transactions (SET)

B-to-B and C-to-B

Merchant not directly connected to Customer's banking information.



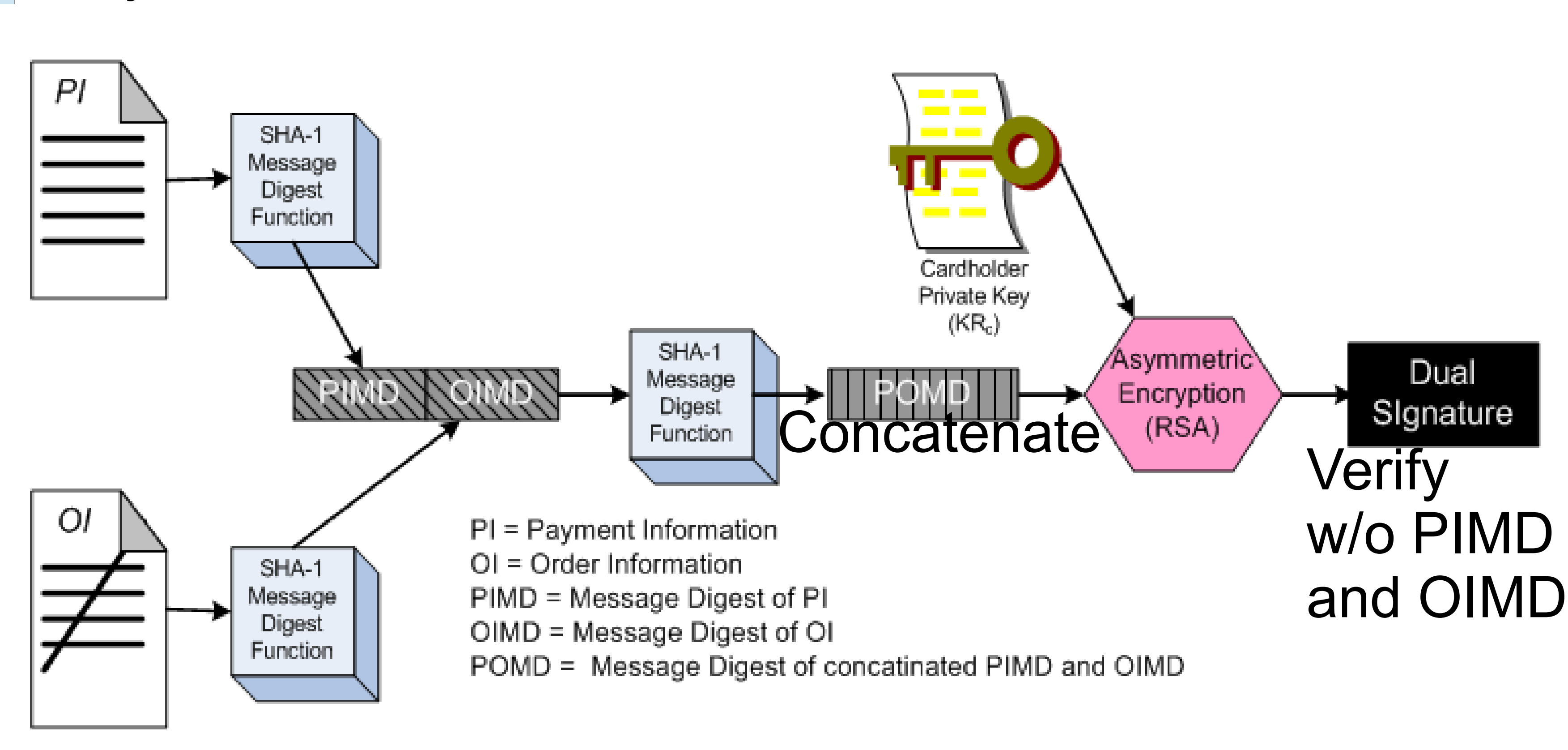
Security

Middleman

Business-to-Business and Customer-to-Business Security

Mutual Peer-entity Authentication

Payment Info


PI
SHA-1 Message Digest Function
OI
SHA-1 Message Digest Function
PIMD
OIMD
SHA-1 Message Digest Function
POMD
Cardholder Private Key (KR$_c$)
Asymmetric Encryption (RSA)
Dual Signature

PI = Payment Information
OI = Order Information
PIMD = Message Digest of PI
OIMD = Message Digest of OI
POMD = Message Digest of concatinated PIMD and OIMD

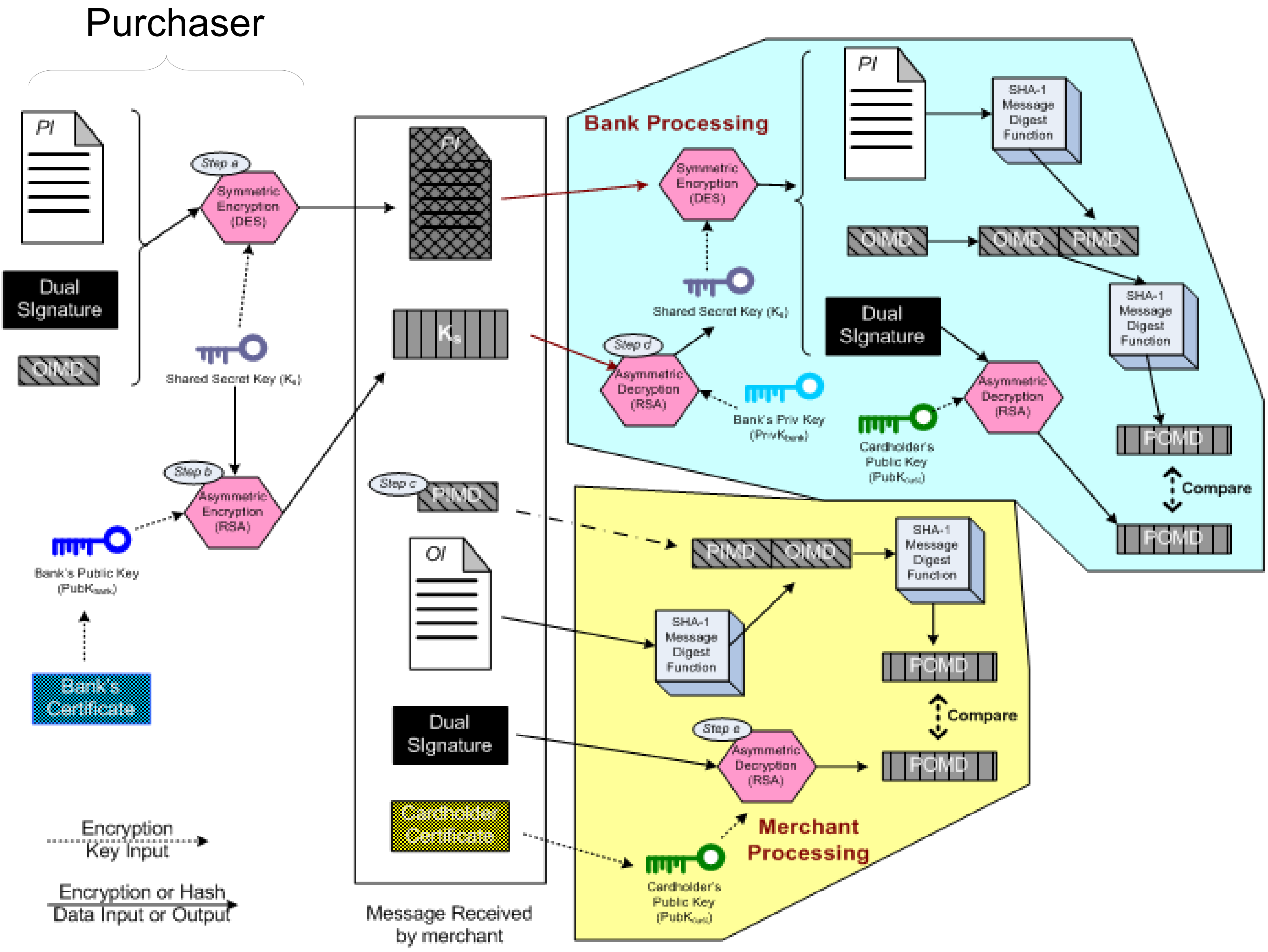
Concatenate

Verify
w/o PIMD
and OIMD

Order Info

COMP 307
Principles of Web Developme

The Authentication Process


Purchaser
PI
Dual Signature
OIMD
Step a
Symmetric Encryption (DES)
Shared Secret Key (Ks)
Step b
Asymmetric Encryption (RSA)
Bank's Public Key (PubKbank)
Bank's Certificate
Encryption Key Input
Encryption or Hash Data Input or Output
PI
Ks
Step c
PIMD
OI
Dual Signature
Cardholder Certificate
Message Received by merchant
Bank Processing
Symmetric Encryption (DES)
Shared Secret Key (Ks)
Step d
Asymmetric Decryption (RSA)
Bank's Priv Key (PrivKbank)
PI
SHA-1 Message Digest Function
OIMD
OIMD
PIMD
SHA-1 Message Digest Function
Dual Signature
Cardholder's Public Key (PubKcard)
Asymmetric Decryption (RSA)
POMD
Compare
POMD
PIMD
OIMD
SHA-1 Message Digest Function
SHA-1 Message Digest Function
POMD
Compare
Step e
Asymmetric Decryption (RSA)
POMD
Merchant Processing
Cardholder's Public Key (PubKcard)

J. Vybihal

Encryption Technologies

On Your Server

- Install SSL (secure Socket Layer)
  - Run IIS (internet information services)
    - The security server
  - Get a certificate
    - Generate your own, or
    - Download from certificate authority (VeriSign)
    - Tell IIS about the certificate
  - Create a folder
    - Point IIS to it
    - Save all your secured pages and data in that folder
- You have: public_html and secure_html

ISP Provides Security Service

- ISP provides an https connection
- ISP has a Shared SSL server (for $$)
- You create a public_html & secure_html folders.
- ISP gives you the addresses:
  - http://www.WebHost.com/YourWeb/public_html /YourPublicPage.html
  - https://www.WebHostSecure.com/YourWeb/secure_html/YourPage.html

SIMPLER ADDRESSES ARE POSSIBLE

Programmer Security

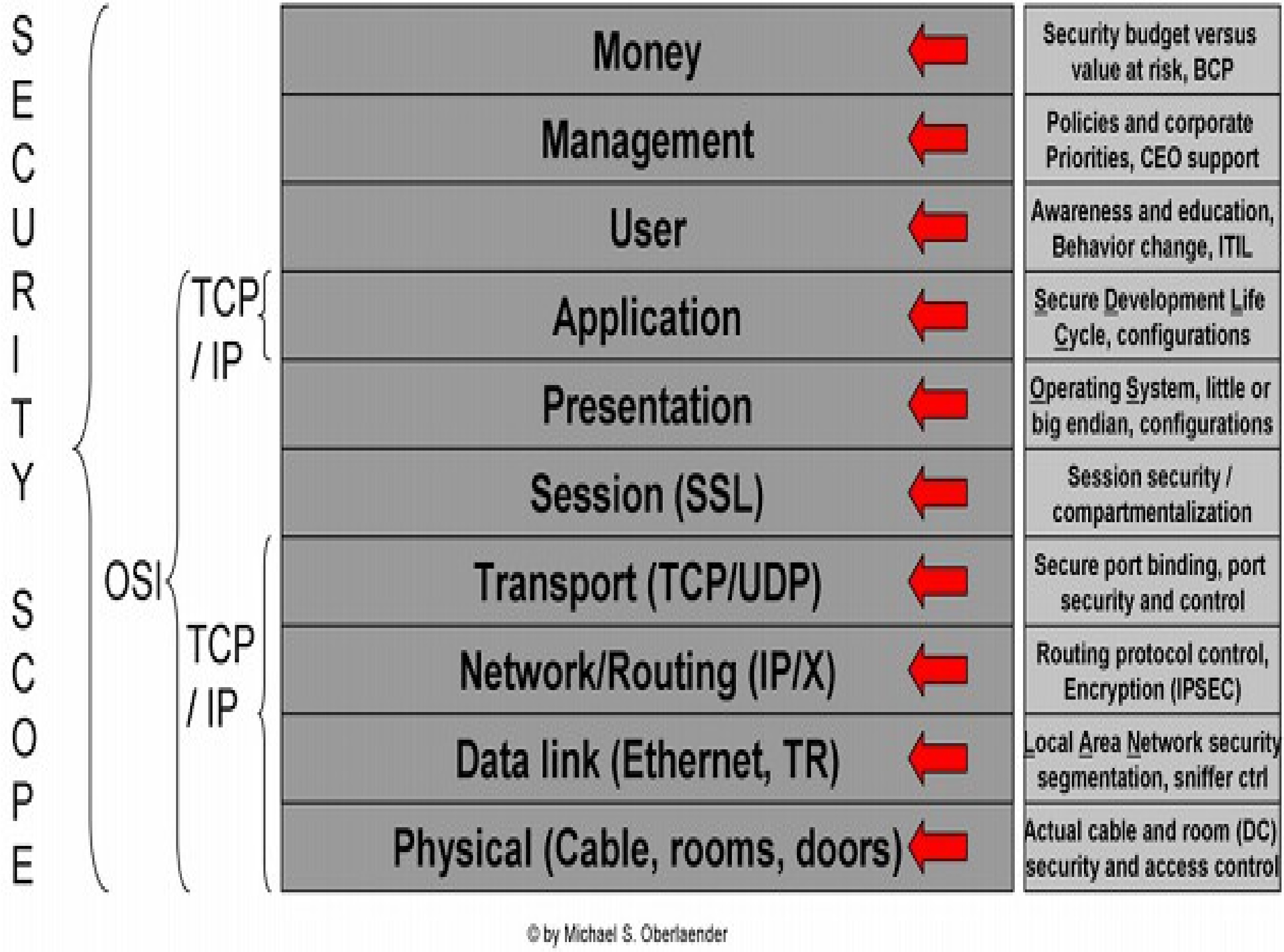
- Use JavaScript to
  - Read input from user
  - Encrypt locally with your own function
  - Transmit to destination using:
    - Get/Post
    - Ajax
    - JWE (JSON Web Encryption), Etc.
- Advantage:
  - Encryption of only parts of packet
    - Faster
    - But maybe not as secure...

The Security Stack

All the layers of security your network uses

COMP 307
Principles of Web Development

Security Stack:



Money	Security budget versus value at risk, BCP	
Management	Policies and corporate Priorities, CEO support	
User	Awareness and education, Behavior change, ITIL	
Application	Secure Development Life Cycle, configurations	Programming
Presentation	Operating System, little or big endian, configurations	Permissions
Session (SSL)	Session security / compartmentalization	Login Key xchg
Transport (TCP/UDP)	Secure port binding, port security and control	Port scan
Network/Routing (IP/X)	Routing protocol control, Encryption (IPSEC)	IP tables Encryption
Data link (Ethernet, TR)	Local Area Network security segmentation, sniffer ctrl	Network layout
Physical (Cable, rooms, doors)	Actual cable and room (DC) security and access control	No Wi-Fi

© by Michael S. Oberlaender