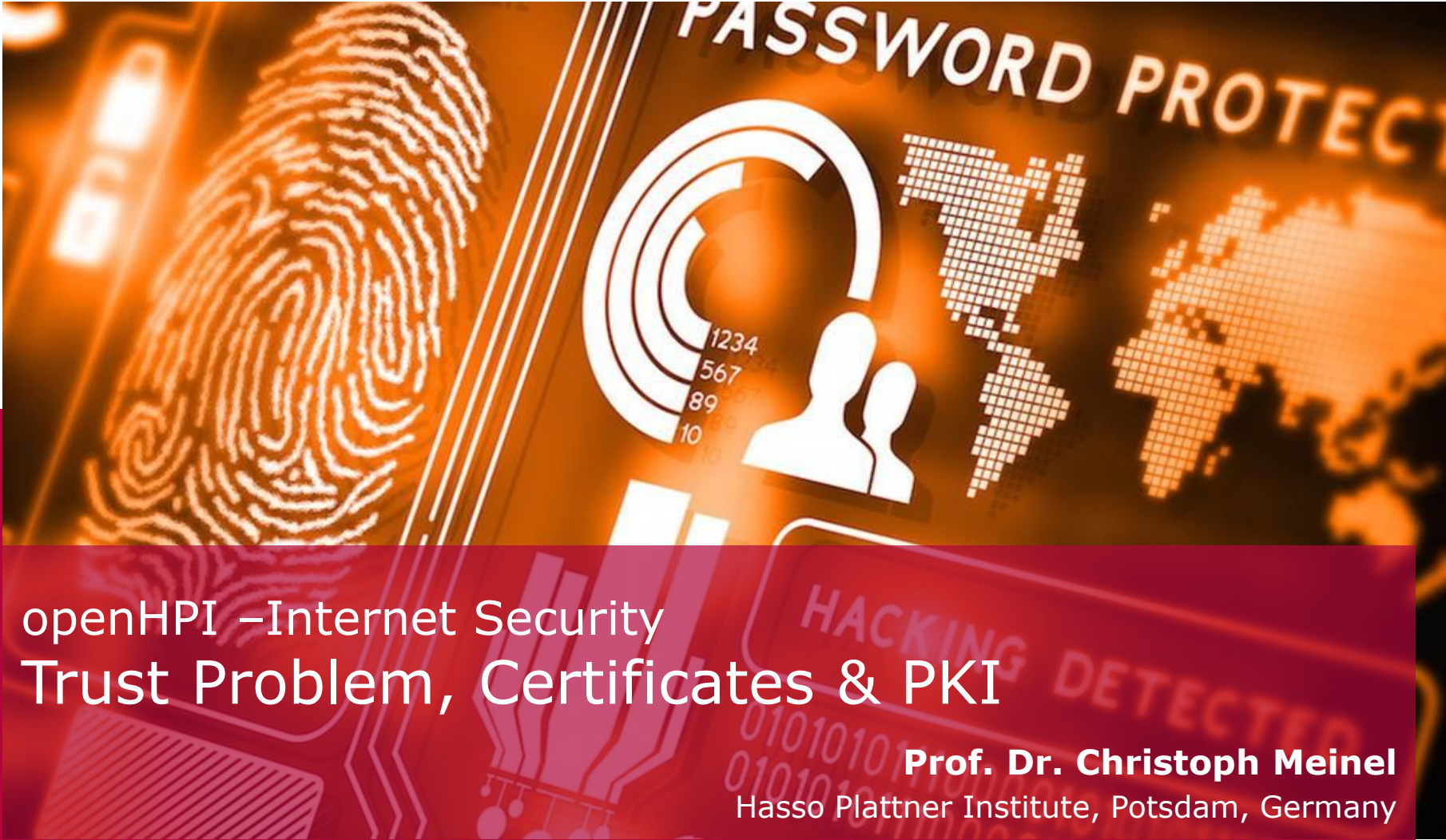




openHPI – Internet Security Trust Problem, Certificates & PKI

Prof. Dr. Christoph Meinel
Hasso Plattner Institute, Potsdam, Germany

Trust Problem of 2-Key Encryption

- The secure usage of 2-key procedures depends very much on the **proper allocation** of public keys to the participants
- If an attacker succeeds in passing off his/her public key for that of a participant, then
 - attacker can decrypt and read all encrypted messages sent to the actual addressees (no longer possible for the actual addressee)
 - an attacker can sign messages on behalf of the addressee
- The secure and trustworthy allocation of public keys to their owners is established through **certificates**

Trust Problem in Practice

- When establishing an HTTPS connection, it is necessary to verify the identity of the communication partner (Internet service) by means of certificates
 - if not, a man-in-the-middle attack in which the attacker poses as a legitimate Internet Service can take place ...
- It is particularly important to check the communication partner in case of critical Internet services like
 - banks (online banking)
 - booking portals
 - email portals
 - ...

Digital Certificates

A (**digital**) **certificate** is a document digitally signed by a trustworthy third party ("**Trust Center**"), in which the link between a person/ entity and its public key is attested to

- If one trusts a trust center that has signed a certificate, one can rely on the attribution of the public key to its owner, as attested to in the certificate
- Certificates contain information (at least) about
 - the certificate holder (person, company, web server, ...)
 - their public key and
 - the digital signature of the trust center that has issued the certificate
- The Trust Center that signed the certificate guarantees the accuracy of the information

Authentication with Certificates

- Use of certificates to authenticate users:
 - user sends a message signed by himself, together with their certificate (signed by a trust center) to a partner/service
 - partner/service first validates the signature of the certificate and checks its authenticity
 - partner/service verifies (decrypts) by using the public key from the certificate, the signature of the message sent by the user
- If the user's signature can be verified, i.e., decryption is possible, it is ensured that the message actually comes from the alleged sender

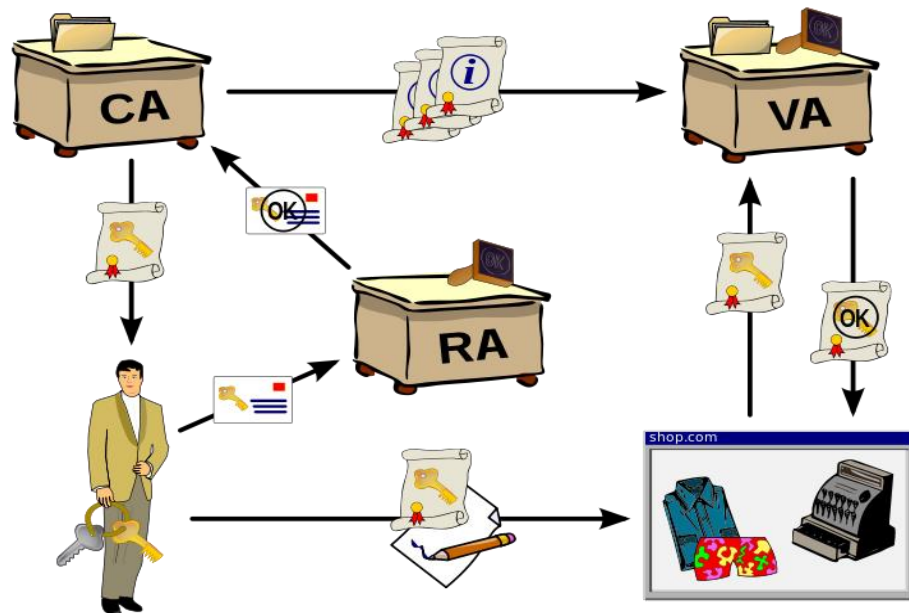
Public Key Infrastructure – PKI

- In order to apply a 2-key encryption in a secure way, i.e., a trust problem is solved by means of certificates, a complete infrastructure is required
→ „**Public Key Infrastructure – PKI**“
- PKI includes software and hardware components as well as personnel to regulate the provision, distribution, use, and revocation of certificates
- At the center of a PKI is the **Trust Center** with the components
 - Registration Authority (RA)
 - Certification Authority (CA)
 - Validation Authority (VA)

Public Key Infrastructure – PKI

Application for a Certificate

1. User verifies his/her identity at the Registration Authority RA and submits his/her public key
2. RA transfers login data (identity data, public key) to Certification Authority CA
3. CA creates and signs certificate for users, and sends it to the user
4. Certificate information is stored with the Validation Authority VA.

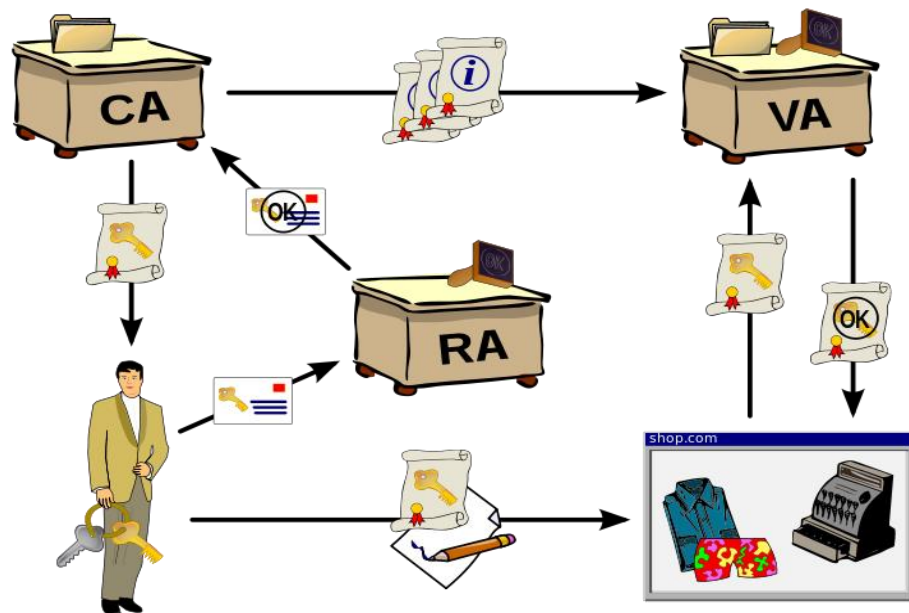


<http://de.wikipedia.org/wiki/Datei:Public-Key-Infrastructure.svg>

Public Key Infrastructure – PKI

Use of the Certificate

1. User sends their certificate to the communication partner
2. The communication partner validates the certificate with the aid of the Validation Authority VA
3. The VA checks the certificate using the stored information and attests to its validity

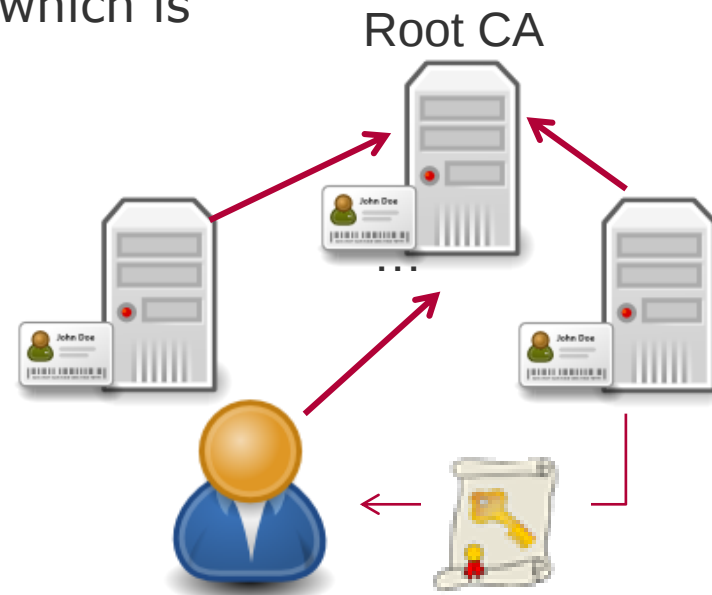


<http://de.wikipedia.org/wiki/Datei:Public-Key-Infrastructure.svg>

Solution to the Trust Problem: Hierarchical PKI

Typically, a PKI is organized hierarchically:

- Hierarchical PKIs have the initial CA instance "**Root CA**", which all subordinate instances (CAs) trust. Root CA signs certificates of subordinate CAs
- If a user wants to verify the identity of a CA, he/she validates the certificate of that CA which is signed by the root CA
- As the user trusts Root CA, he/she also trusts their signature and thus the certificate



Solution to the Trust Problem: Hierarchical PKI in Practice

TLS / SSL based connections are based on hierarchical PKI

- "Root CA" certificates are stored in the browser or operating system

Example: Establishment of an HTTPS-based connection to the online marketplace Amazon:

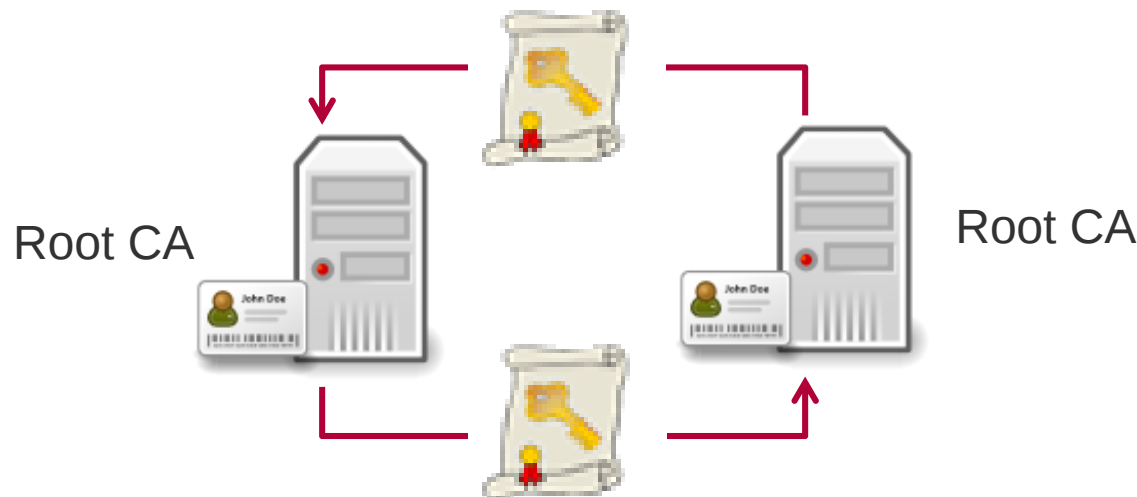
- Verifying the signature of the certificate at www.amazon.de with the help of a higher-level certificate (Symantec)
- The signature of the Symantec certificate is checked with the help of the already stored and trusted root CA certificate (VeriSign)
- This can be used to establish a "trust chain" up to the "Amazon" certificate

Certificate Hierarchy

△ VeriSign Class 3 Public Primary Certification Authority - G5
△ Symantec Class 3 Secure Server CA - G4
www.amazon.de

Solving Trust Problem: Cross-Certification

- Different PKIs can be linked to each other via **cross-certification**
- The root CAs of the two PKIs exchange certificates for each other and sign them
- Thus the instances of one side trust those of the other, since their certificate was signed by their own root CA.



Solving Trust Problem: Web of Trust

- The user can sign many other users' certificates and have his/her's signed by them
- The more other users sign his/her certificate, the more trustworthy is that certificate
- If user A is known to user B and the certificate of user B has been signed by C, A can also trust the certificate of C

