Integrating PKI and Kerberos Authentication services

Emmanuel Ormancey, Alberto Pace

Authentication Methods

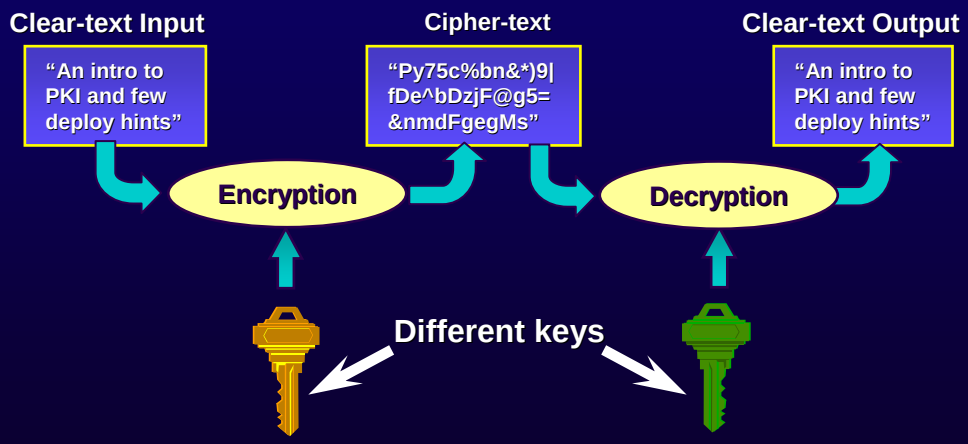
- Two technologies for authentication
  - Kerberos and X.509 Certificates (PKI)
- Today at CERN
  - Kerberos is used in Windows Domains and AFS
  - PKI is used in all Grid related projects, with multiple certification authorities
  - Both technologies here to stay
- Multiple scenarios exist to integrate and interoperate the two technologies

Kerberos vs PKI ?

- Both technologies have weak and strong points
  - Distributed versus centralized management
  - Forwardable authentication
  - Offline authentication
- Technology is different
  - Asymmetric encryption with public/private key pairs versus symmetric encryption and shared secrets
- Some details follows …

PKI basics

- PKI provides, among other services, an authentication protocol relying on *asymmetric encryption.*
- One of the keys is kept private, the other is made public. Public keys are distributed using certificates which are digitally signed by trusted authorities



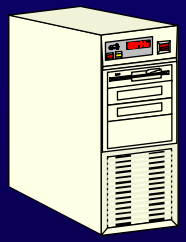
PKI: Obtaining a Certificate

User generates a key pair

Public key is submitted to CA for certification

Certificate is sent to the user

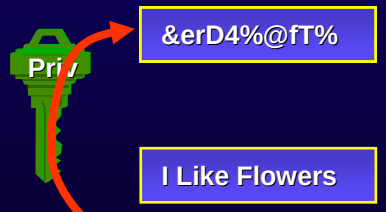
User identity verified,
Digital signature added,
Certificate produced



Certification Server

PKI: Authentication with Certificates

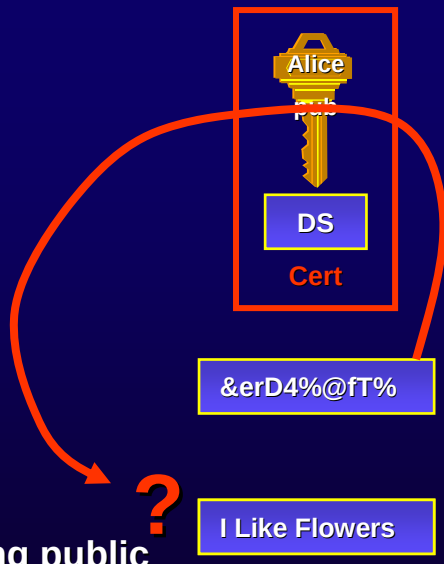
Alice



Encrypt using private key

Certificate is sent for authentication

Bob



Decrypt using public key in certificate and compare

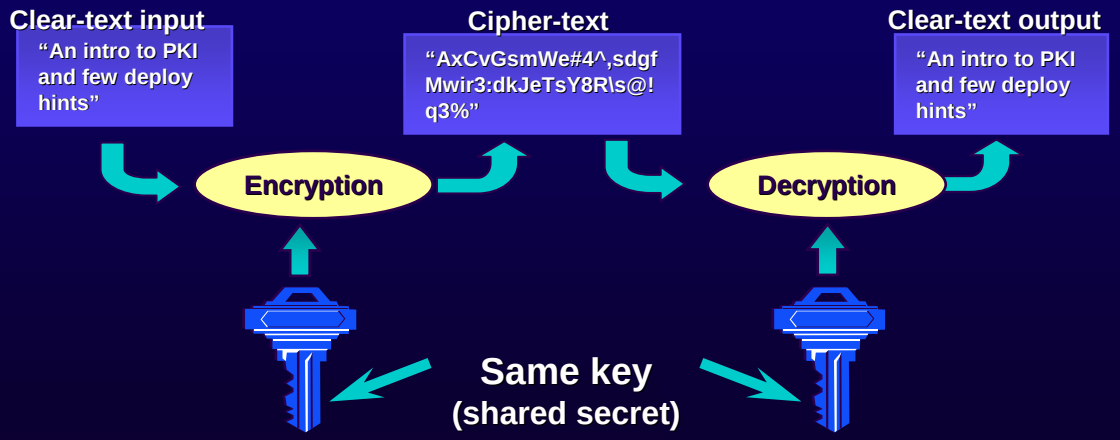
Bob verifies the digital signature on the certificate

He can trust that the public key really belongs to Alice, but is it Alice standing if front of him ?

Bob challenges Alice to encrypt for him a random phrase he generated

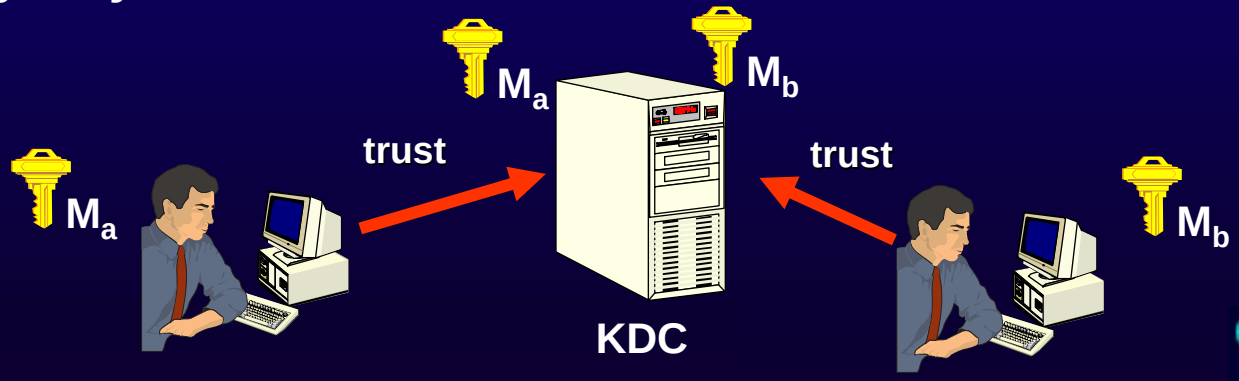
Kerberos Differences

- Kerberos is an authentication protocol relying on *symmetrical* cryptographic algorithms that use the same key for encryption as for decryption
  - Different from PKI !



Kerberos: Basic principles

- There is a trusted authority known as the Key Distribution Center (KDC) which is the keeper of secrets.
- Every user shares a secret password with the KDC
  - technically the KDC doesn't know the password but rather a one way hash, which is used as the basis for a cryptographic "master key".
- The secret master key is different for each user
  - As two users don't know each other master key they have no direct way of verifying each other's identity
- The essence of Kerberos is key distribution. The job of the KDC is to distribute a unique session key to each pair of users (security principals) that want to establish a secure channel.
  - Using symmetric encryption
- Everybody has to trust the KDC

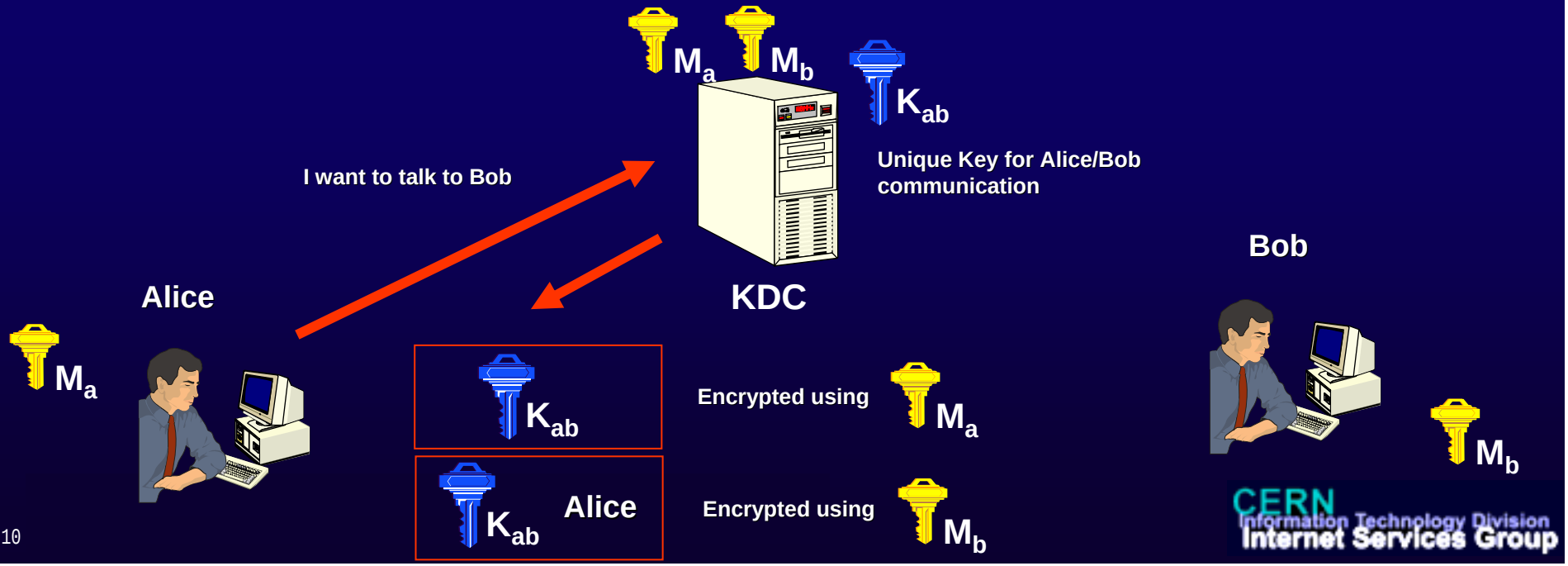


Breakthrough of a (simplified) Kerberos session

- Alice wants to communicate with Bob
  - bob could be a server or a service
- Alice can communicate securely with the KDC, using symmetric encryption and the shared secret (Master Key)
- Alice tells the KDC that she wants to communicate with Bob (known to the KDC)

(simplified) Kerberos session 2

- The KDC generates a unique random cryptographic key for Alice and Bob to use (call this K_{ab})
- He sends back two copies of K_{ab} back to Alice.
  - The first copy is for her to use, and is sent to her along with some other information in a data structure that is encrypted using Alice's master key.
  - The second copy of Kab is packaged along with Alice's name in a data structure encrypted with Bob's master key. This is known as a "ticket".



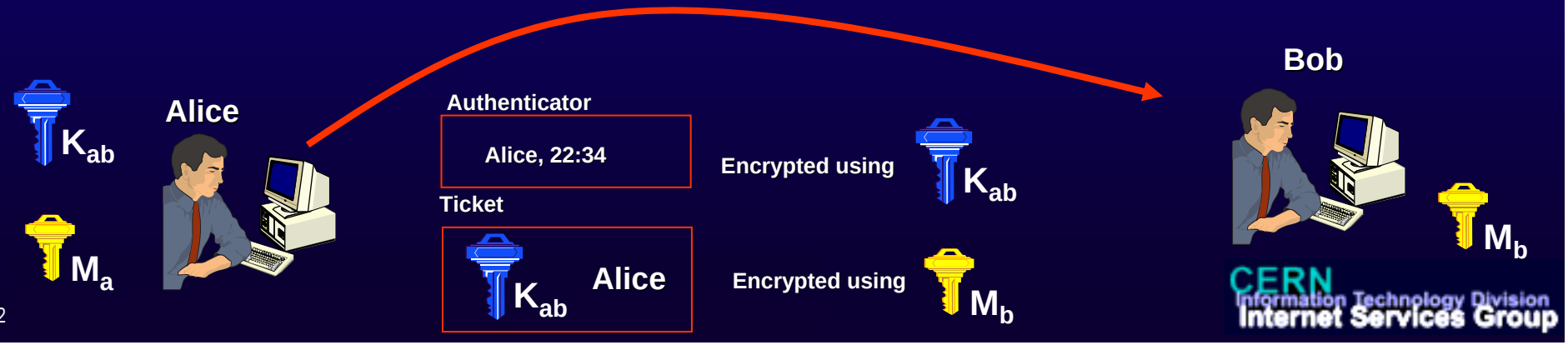
CERN Information Technology Division Internet Services Group

What is the ticket ?

- The ticket is effectively a message to Bob that only BOB can decrypt
  - "This is your KDC. Alice wants to talk to you, and here's a session key that I've created for you and Alice to use. Besides me, only you and Alice could possibly know the value of K_{ab}, since I've encrypted it with your respective master keys. If your peer can prove knowledge of this key, then you can safely assume it is Alice."

Kerberos authentication

- Alice must send the ticket to Bob
  - with proof that she knows K_{ab}
  - and she must do it in a way that allows Bob to detect replays from attackers listening on the network where Alice, Bob, and the KDC are conversing.
- The ticket is sent to Bob, with an authenticator (her name and the current time, all encrypted with the session key K_{ab})
- Bob takes the ticket, decrypts it, and pulls K_{ab} out. Then decrypts the authenticator using K_{ab}, and compares the name in the authenticator with the name in the ticket
  - If the time is correct, this could provide evidence that the authenticator was indeed encrypted with K_{ab}



CERN Information Technology Division Internet Services Group

Kerberos authentication

- It time is incorrect, bob reject the request
  - with a hint of what his time is (Bob time isn't a secret)
- If the time is correct …
  - … it's probable that the authenticator came from Alice, but another person might have been watching network traffic and might now be replaying an earlier attempt. However, if Bob has recorded the times of authenticators received from Alice during the past “five minutes”, he can defeat replay attempts. If this authenticator yields a time later than the time of the last authenticator from Alice, then this message must be from Alice
  - This is why time synchronization is essential in kerberos and all KDC provides also time synchronization services
- You can see this as a “challenge” on the knowledge of the shared secret (K_{ab}):
  - “prove that you know K_{ab} by encrypting the current time for me”

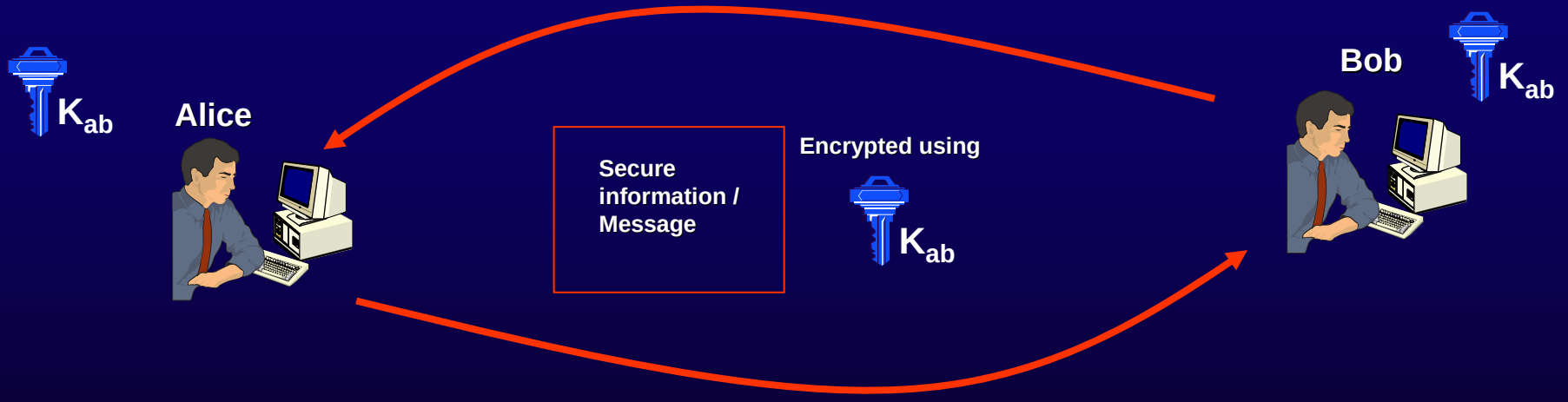
Mutual authentication

- Alice has proved her identity to Bob
- Now Alice wants Bob to prove his identity as well
  - she indicates this in her request to him via a flag.
- After Bob has authenticated Alice, he takes the timestamp she sent, encrypts it with K_{ab}, and sends it back to Alice.
- Alice decrypts this and verifies that it's the timestamp she originally sent to Bob
  - She has authenticated Bob because only Bob could have decrypted the Authenticator she sent
  - Bob sends just a piece of the information in order to demonstrate that he was able to decrypt the authenticator and manipulate the information inside. He chooses the time because that is the one piece of information that is unique in Alice's message to him

CERN Information Technology Division Internet Services Group

Kerberos Secure Communication

- Alice and Bob share now a unique secret K_{ab} that they use to communicate



Real life is more complicated

- Real Kerberos includes several extra steps for additional security
- When Alice first logs in, she actually asks the KDC for what is called a "ticket granting ticket", or TGT.
- The TGT contains the session key (K_{ak}) to be used by Alice in her communications with the KDC throughout the day.
  - This explains why when the TGT expires you have to renew it
- So when Alice requests a ticket for Bob, she actually sends to the KDC her TGT plus an authenticator with her request.
- The KDC then sends back the Alice/Bob session key K_{ab} encrypted with K_{ak}
  - as opposed to using Alice's master key as described earlier
- See various Kerberos references for details

CERN Certification authority, PKI and Kerberos integration

Authentication Services at CERN

- For Kerberos:
  - Two KDC in production, one for Windows computers (cern.ch domain) one for AFS (cern.ch cell)
  - Account and passwords planned to be synchronized
- For the grid
  - CERN Certification authority
    - http://cern.ch/service-grid-ca
- Plan for 2006 / 2007
  - Migrate to a new certification authority integrated with the kerberos services

New CERN Certification authority

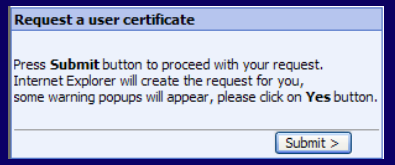
- Aim to issue certificates
  - Recognized by the entire grid community
  - Valid to obtain kerberos ticket automatically
- Separate Root CA and Issuing CA
- Offline Root CA:
  - Run on Virtual PC, Server image on removable disks
  - Root trusted by default inside CERN.
- Online Issuing CA
  - Issues all certificate, online
  - Web site http://cern.ch/ca

CA services planned

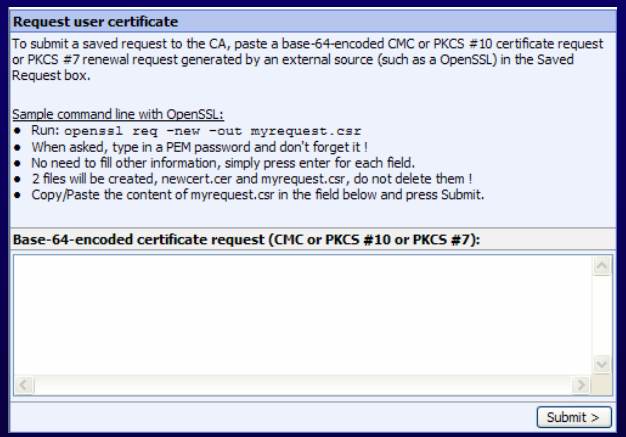
- Issuing User certificates
  - ‘software’ client certificates
  - Certifies the identity of a persons
    - Current identification based on knowledge of Kerberos account password
- Issuing Host certificates
  - Typically for all web servers requiring, for example, https services
  - Can certify any host in the cern.ch domain from the CERN network database
  - Manual procedure for host outside the cern.ch domain
- Service certificates foreseen
- Issuing SmartCards
  - Certificate and private key in a HW token
- Allow users to map existing certificates issued by trusted CA (for example existing Grid certificates) to their account.

DEMO: User Certificate Request

Internet Explorer or Mozilla browsers can handle automatically certificate request.

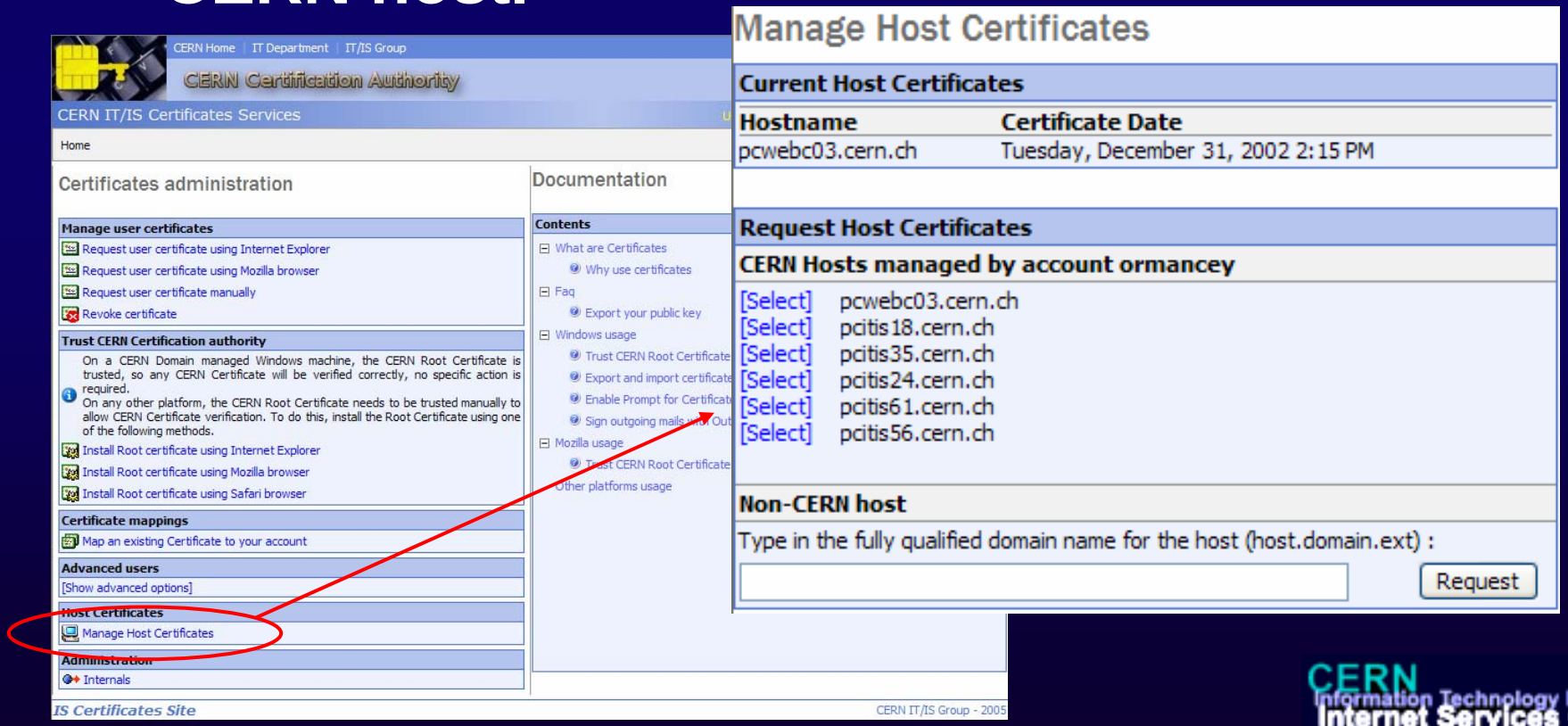


A manual procedure with OpenSSL is also provided.



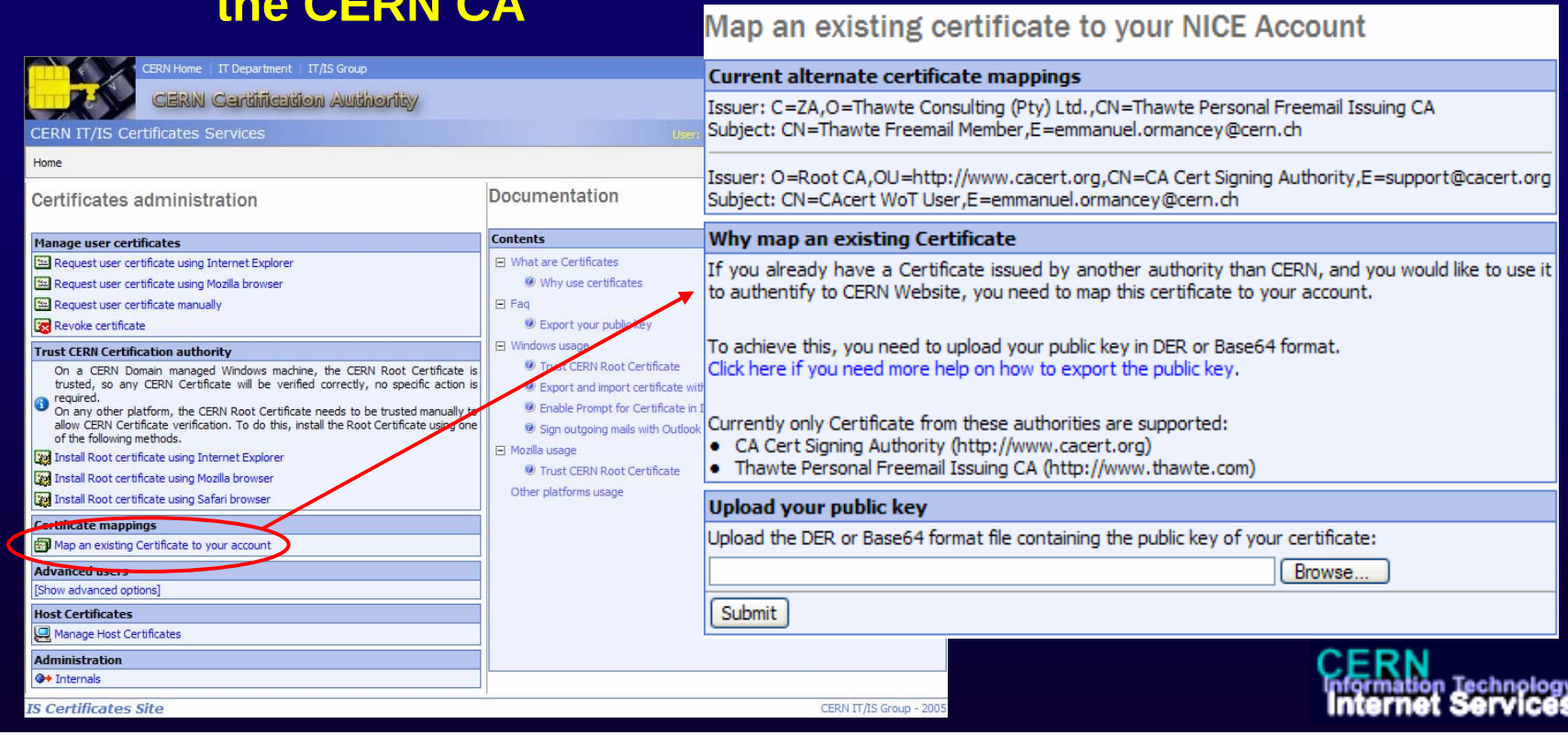
DEMO: Host Certificates

- Users can request Host certificates for CERN Hosts they manage, and any non-CERN host.

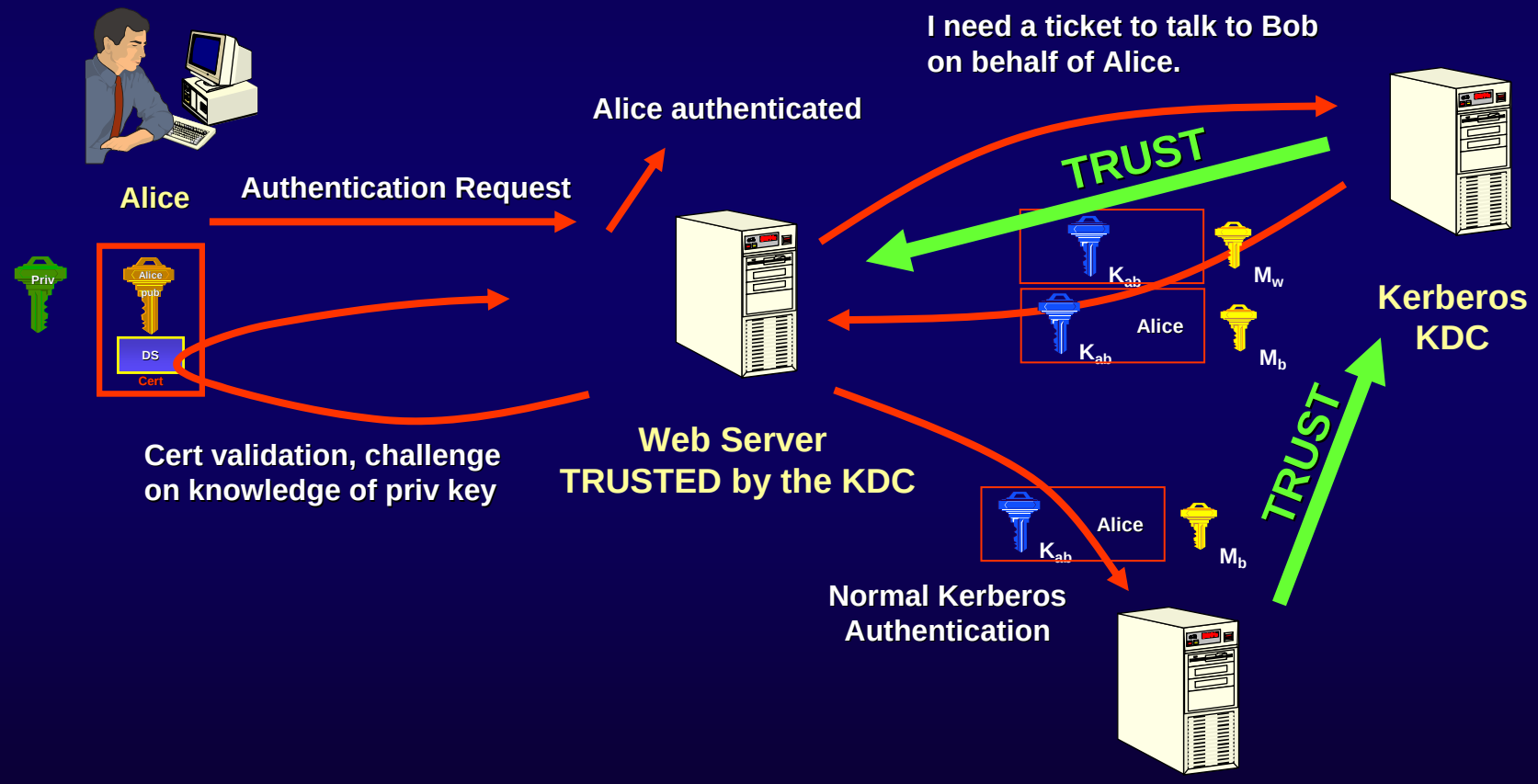


DEMO: Certificate mapping to Existing Account

- Users can map an existing certificate to their Kerberos account for authentication
  - Typically for owners of Grid certificates not issued by the CERN CA



PKI / Kerberos integration



See: Kerberos Constrained Delegation

Roadmap for 2006

- Obtain accreditation from the European Grid Policy Management Authority (www.eugridpma.org)
  - Obtain approval of the new Certificate Policy and Certification Practice (CP/CPS)
    - See http://www.eugridpma.org/members/
  - From offline issuing CA to online issuing CA with FIPS Hardware module
    - http://www.eugridpma.org/guidelines/IGTF-AP-classic-20050930-4-0.pdf
- Verify interoperability with Windows and Linux Desktops
  - Desktop login requires Active Directory path to match Certificate Distinguished Name
  - Alternate user mapping possible
  - Usage of Smartcard on linux requires further investigation

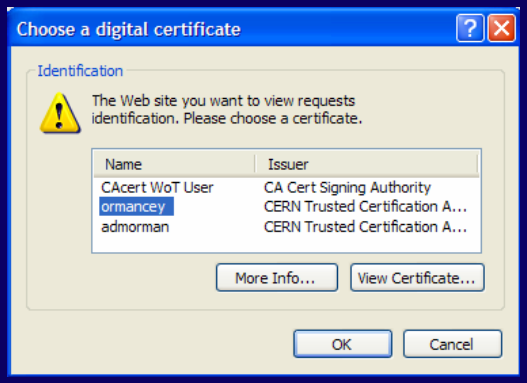
Certificate usage, Interoperability

- **Once a certificate is installed in the client computer**
  - **Can authenticate to CERN Websites (Win, Web, Mail, Terminal services, etc…)**
    - Not all CERN web sites yet, but planned
    - Best example of PKI / Kerberos interoperability
  - **Can participate in any grid activity, worldwide**
    - Certificate recognized worldwide within the grid community
  - **Secure e-mail possible**
  - **Provide a common authentication interface for CERN services: sort of Single Sign On**
- **Medium to long term:**
  - **Have the CERN certificates trusted worldwide, not only within the grid community**
  - **Support Windows and Linux desktop authentication using Smartcard certificates.**
  - **Combine together SmartCards and CERN Access cards.**

Web Authentication example

Opening a website

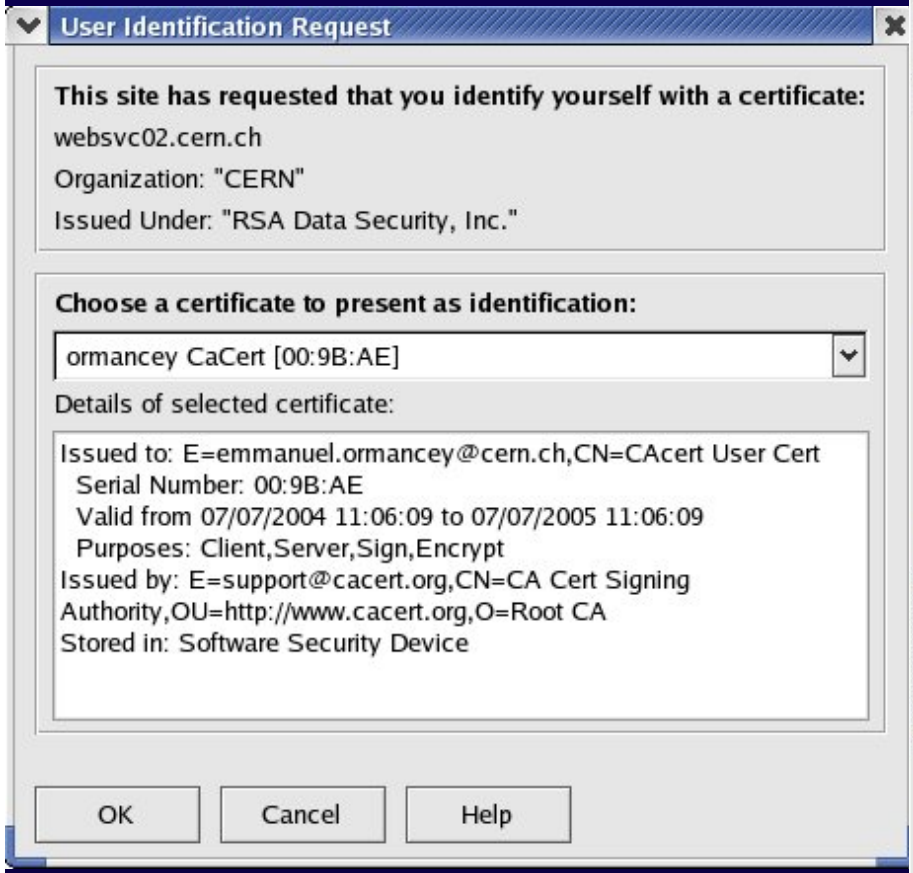
If several client certificates matching server requirements are found, browser asks to choose.

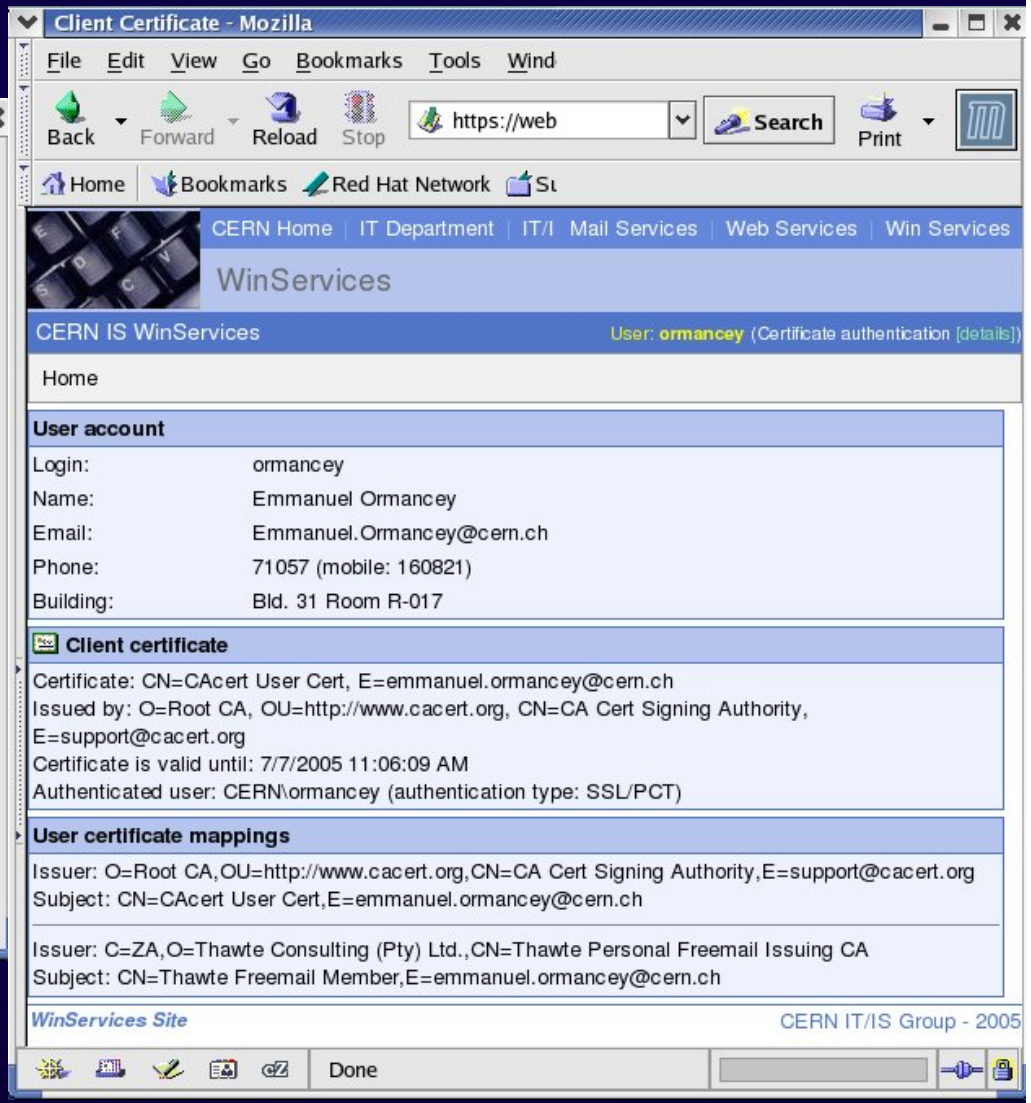


Certificate authentication complete.

Cancelled or no certificate installed

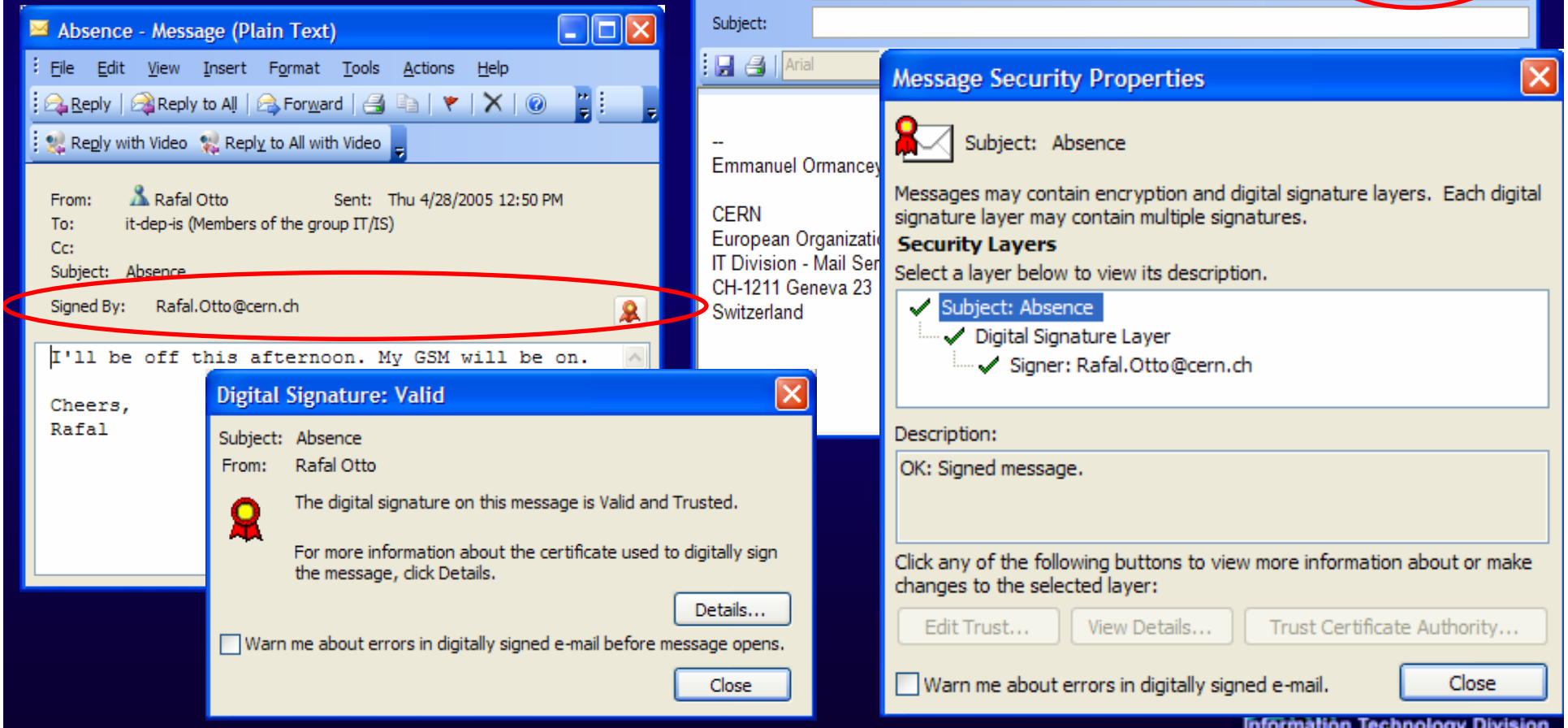
Technology not platform specific





Example: Email signing

- In Outlook:



Managing Certificates

- Software certificates expire and must be renewed
  - Typically once a year
  - Renewing a certificate is more complicate than a password change
- Looking towards automating request, distribution and installation of Client certificates
  - For PCs member of a Windows domain, the CERN certificate can be pushed to the client as a domain policy
  - Its renewal can be handled automatically (allowing short validity periods)
  - Users do not need to understand, be aware, be informed. 100 % transparent.
- Similar automation levels exist for Linux and Mac OS systems, but require the computers to be centrally managed
- Otherwise, Smartcards are a possible solution
  - Much easier for the user to understand
  - Longer certificate validity

CERN Access Card

- **Use the same SmartCard for:**
  - **Windows desktop (and laptop)**
    - Logon and Browser authentication
  - **Linux desktop**
    - Browser authentication
  - **Mac OS X desktop**
    - Browser authentication
  - **Remote windows**
    - Windows Terminal Services
  - **Remote Linux**
    - Putty (to be defined, possible with OpenSC)
    - OpenSSH (to be defined, possible with OpenSC)
    - Exceed (to be confirmed)

Conclusion

- CERN is improving its Certificate Authority service to
  - issue certificates useable within the grid community
  - Further automate certificate issuing procedures
  - Automatically map Certificates to Kerberos accounts (when possible)
- In addition, Certificates issued by other trusted CA can be mapped to Kerberos accounts
- This should provide a good PKI/Kerberos interoperability