Enabling Grids for E-science in Europe

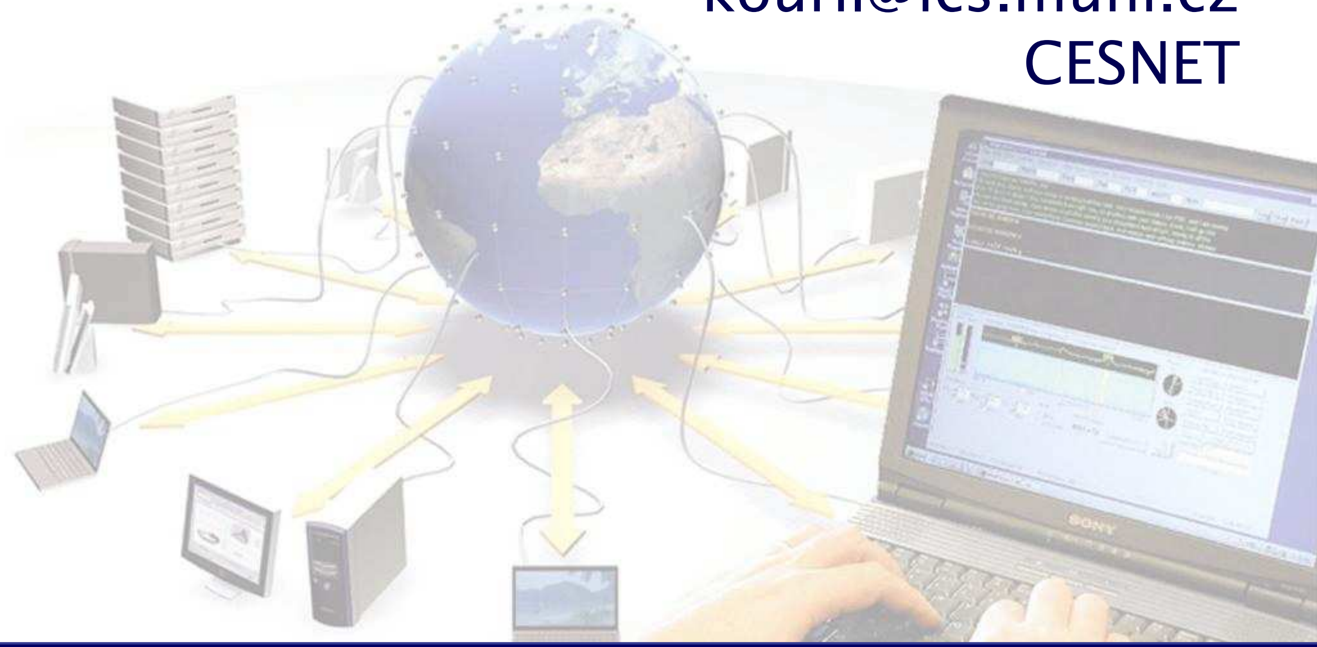
www.eu-egee.org

MWSG meeting, CERN, September 15, 2005

Hardware Tokens in *META Centre*

Daniel Kouřil

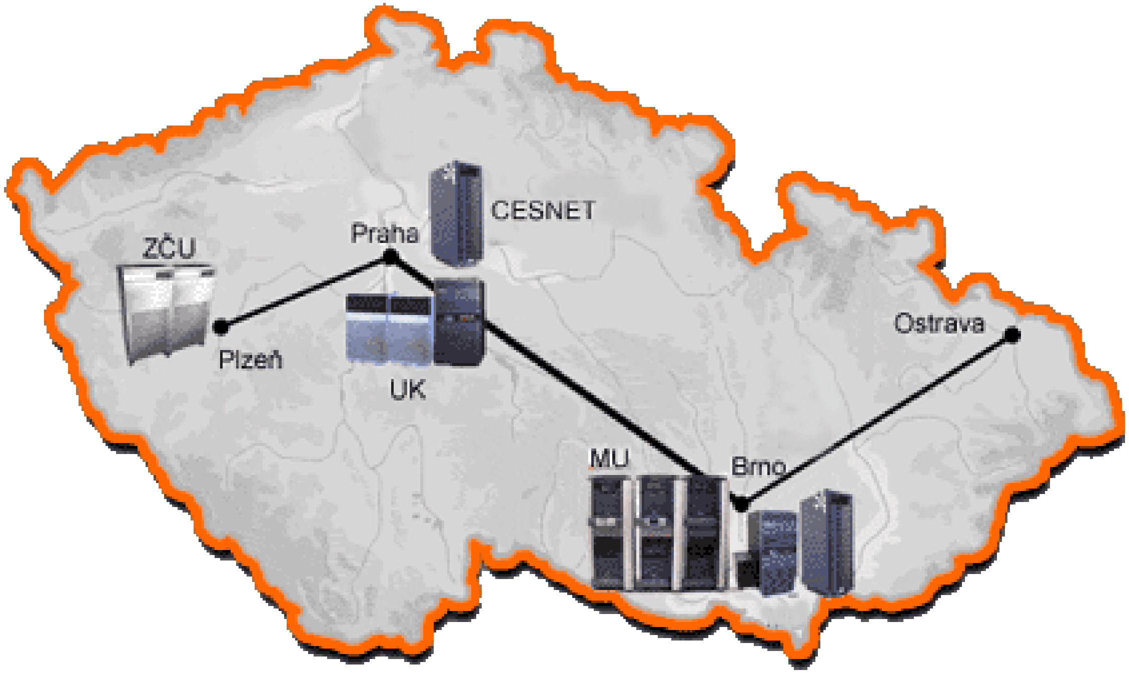
kouril@ics.muni.cz
CESNET



EGEE is a project funded by the European Union under contract IST-2003-508833

Project *META Centre*

- One of the basic activities of CESNET (Czech NREN operator); started in 1996
- Focus at development and production support of a distributed infrastructure that spans multiple independent organizations
  - Nodes represent main academic supercomputing centers providing computing and storage resources
  - Sites are connected with the CESNET backbone



Project *META Centre*

- The goal is to provide users with an easy access to the resources, hiding the complexity of the environment
  - “Grid” (in current terminology), “Metacomputer” (cca ten years ago)
- Basic blocks:
  - Unified authentication mechanism, support for SSO (Kerberos)
  - Shared disk space (AFS)
  - Single batch system (PBSPro)
  - User management system (Perun)
  - User support (METAPortal, RT)
- Heterogeneous resources
  - Clusters based on various Intel Pentium procesors, cca 330 CPUs
  - SMP machines (SGI and HP servers) together cca 100 CPUs
  - Small number of other architecture (IBM Power4+, AMD Opteron)
- About 200 active users
- Aplications mainly from computional chemistry, fluid dynamics (no HEP)
- Quite simple infrastructure but suits perfectly

Security Architecture of *META Centre*

- Focused mainly on authentication and SSO. Authorization solutions/needs under investigation
- Kerberos v5
  - (mutual) authentication
  - integrity protection and/or encryption
- Support for Single Sign-On
- All main services kerberized
  - remote access (telnet, ssh, rsh), file transfer (scp, ftp), web environment, PBS, AFS
- Implementation Heimdal from KTH

Kerberos Overview

- Authentication protocol using a trusted central authentication service
- Entirely based on symmetric cryptography
- Each user and service share a secret key with the AS (Key Distribution Centre – KDC)
- AS issues "tickets" that the clients use to authenticate (analogy to the X.509 certificates)
- Ticket Granting Ticket – universal ticket that can be used to retrieve other tickets (for end services). Means for SSO.
- Time-tested protocol, supported by many systems (MS Windows, MAC OS X, Linux distributions)
- Standardized by IETF RFC 4120

Kerberos vs. PKI

- Symmetric vs. asymmetric cryptography
  - performance
- Tickets vs. proxy certificates
  - Similar concept
  - Proxy are managed only by the users, tickets always issue the KDC server
- Online KDC vs. offline CA
  - note CRLs updates and OCSP
- Password vs. private key
- Scalability
  - Kerberos must know all users/services in advance

PKI in *META Centre*

- Interest in PKI support
  - Requested for Grids, some applications support PKI better than Kerberos (email signing, web authentication)
  - Private key management too weak
- Project "HW tokens for *META Centre*"
  - Token – device that allows to store private keys and perform basic cryptographic operations (smart card or USB token). Private key never leaves the token.
  - Funds to equip users with tokens
  - Evaluation of available tokens
  - Adaptation current infrastructure to support PKI and HW tokens, two-factor authentication
  - Distribution to the end users

PKI Integration with *META Centre*

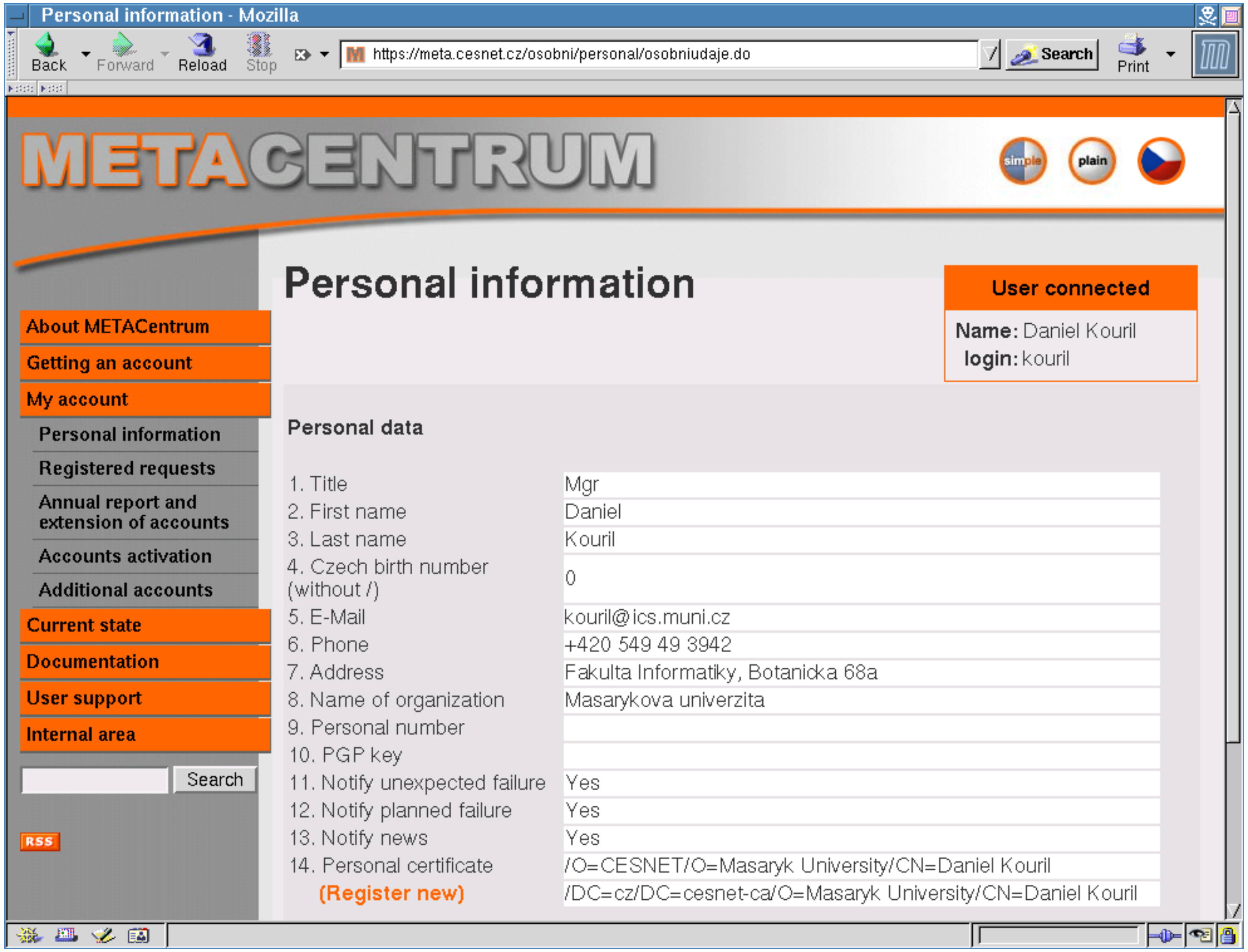
- Enhancement of current infrastructure not replacement of Kerberos
  - How to use PKI credentials to authenticate against KDC
- PKINIT
  - Draft from the IETF Kerberos working group
  - Allows to get a TGT using PKI credentials instead of standard password
  - All subsequent authentication communication and end services not influenced
  - Implementated an initial version of the protocol (with a very simple support for smart cards)
  - Accepted by the Heimdal developers, added support for the openssl engine and PKCS11

PKINIT Deployment in *META Centre*

- The KDC servers upgraded to new versions (with PKINIT support)
- Accepted all CA certificates accredited by eugridpma
- CRLs updated using the fetch-crl cron script
- PKI-mapping files (maping X.509 DN to Kerberos principal names) propagated to the KDCs by the user management system
- All KDC servers have certificates issued by CESNET CA

PKI Integration with *META Centre*, cont.

- Changes to user management system
  - Currently users are identified by their Kerberos identity
  - Users’ certificate must be registered with the *META Centre* user management system
  - Users use the portal and assign their certificates with their account
    - ∗ Access to this portal is secured by Kerberos (password or ticket)
    - ∗ Users also have to authenticate using their certifacate using https (to prove they really posses the certificate)
  - The user management system propagates the mapping information to the KDC
    - ∗ It also propagates grid-mapfiles and changes to a testing VOMS server (see later)



Personal information

User connected

Name: Daniel Kouril
login: kouril

Personal data

1. Title	Mgr
2. First name	Daniel
3. Last name	Kouril
4. Czech birth number (without /)	0
5. E-Mail	kouril@ics.muni.cz
6. Phone	+420 549 49 3942
7. Address	Fakulta Informatiky, Botanicka 68a
8. Name of organization	Masarykova univerzita
9. Personal number	
10. PGP key	
11. Notify unexpected failure	Yes
12. Notify planned failure	Yes
13. Notify news	Yes
14. Personal certificate	/O=CESNET/O=Masaryk University/CN=Daniel Kouril
(Register new)	/DC=cz/DC=cesnet-ca/O=Masaryk University/CN=Daniel Kouril

Hardware Selection

- Tested several smart cards and tokens
- Requirements:
  - Interoperability among OS's (Linux and MS Windows)
  - Support in open-source tools (so we can easily adapt our current SW)
  - Support in common applications (mail clients, web browsers)
- USB token Rainbow iKey 3000 (now SafeNet)

Rainbow iKey 3000

- On-token cryptographic generator for RSA key-pair generation, support for RSA, 32kB EEPROM
- Shipped with PKCS11 and Microsoft CAPI (CSP) support and tools for management (initialization, loading keys and certificates)
- Good support in open-source OpenSC
  - Issue with token initialization (see next slide)

Rainbow iKey3000 – Initialization

- Initialization – formating, setting access PINs/PUKs, generate a key-pair, CSR and store the result certificate
- Need to be done only once at the beginning
- Can be initialized using both OpenSC and vendor SW
- Unfortunately, OpenSC isn’t able to format the token in the vendor format
  - but can read and use it
- Users who want to switch among OS’s must initialize the token using the vendor SW
  - It can be used by everywere then

Client side SW

- Support in common applications
  - sucessfuly tested Mozilla Firefox, Mozilla Thunderbird, Microsoft Outlook, Microsoft Internet Explorer
  - Generaly, all applications using the PKCS11 interface should work
- Access to *META Centre*
  - Users must be able to create tickets on their workstations and then use Kerberos-enabled application to access *META Centre* resources
    - ∗ They were often used to use standard SSH and their Kerberos password
  - All main Linux distributions contain Kerberos and kerberized version of appliactions
  - We provide a basic set of packages to be installed (containing the kinit command to receive a ticket using PKI) and configuration files
  - Windows users are provided with a full Kerberos installation (based on the kfw distribution from MIT) with a modified kinit command. We also provide a PuTTy and WinSCP clients that can talk Kerberos

Tokens Distribution

- Users distributed across the whole country
- Short courses (preferably as part of other events) to distribute the tokens and provide help in their initialization
- Agreement with the CESNET CA to establish a RA for the *META Centre* users
- CESNET CA switched to a new SW (Entrust), all interactions can be easily done via a web browser

Experiences and Future Work

- Hard to make users use the tokens instead of passwords
- Users don't need HW tokens or PKI, situation will change when they start using "real" Grids
- Some (new) services made available only to PKI-authenticated users, we're also considering prioritizing of jobs for PKI-autenticated users (in order to motivate users to use the tokens)
  - How to distinguish such users must be investigated
- Not all users have USB port (SGI workstation) or travel often and can't use token everywhere.
  - A credential repository could solve the problem
  - Preferrably with support for OTP (some initial work started)

VOCE

- VO established and operated by the CE federation of SA1
- CESNET leads this effort
- Resources provided by the whole CE federation
- Provides CE users with a production grid environment
- Primarily aiming at newcommers and small group of application without their own VO

VOCE Management

- Used the same tool as for *META Centre*
- Able to propagate grid-mapfile (only upon each change, no periodical checks)
- Also able to feed data to a VOMS server (currently only for testing purposes, only a single groups of users (VOCE) is used)

Tokens in PKI-based Grid

- Similar issues as in the Kerberos world
  - Create a proxy from the token
  - Allow users to use this proxy to log in the UI and to delegate this proxy to the UI
- Currently almost all users have their PKI credentials stored on their UIs (and other machines as well) and use passwords or SSH keys to access the UIs
- Users are very satisfied with this arrangement
- Can’t be done when tokens are used
- Users’ habits must change significantly to use tokens (not always easy)

Tokens in PKI-based Grid

- We have a grid-proxy-init.sc command, creating a proxy certificate using tokens
  - A quick poor man’s solution, wrapper around standard grid-proxy-init
  - Fake self-signed certificate and corresponding private key hardwired in the binary
  - Standard grid-proxy-init invoked to create a proxy from this fake credentials
  - Upon creation, the subject and issuer names are replaced with real ones from the certificate on the token and the proxy is re-signed using the token
  - Automatical support of all functionalities of standard grid-proxy-init, and all proxy formats
- myproxy-init
  - Basicaly wrapper around grid-proxy-init
  - Trivial to make it call our grid-proxy-init.sc without any changes to the code
- voms-proxy-init
  - can be easily made use the proxy generated in previos step and use it instead of users’ long-term credential, resulting in a proxy containing VOMS attributes which is equivalent to the one created by standard voms-proxy-init (only longer)
- Tools not deployed or tested, rely on standard grid-roxy-init (no support for Windows)

Tokens in PKI-based Grid

- GSI-enabled SSH daemon
  - Available from NCSA/Globus
  - Mechglue support allows to use both Kerberos and GSI authentication
  - Installed on the VOCE UI
- gsissh clients easily available for Unix
- PuTTY ssh client can be built with GSI support (currently only for Linux)
- User issue the grid-proxy-init.sc command on their local workstation, create their proxy and use a gsi-ssh client to log into a UI
- Provides real SSO, unfortunately users seem be upset :-)

Kerberos and GSI Integration

- Universal authentication command
  - Creating Kerberos tickets, AFS tokens and a proxy certificate at once
- Proxy support in the PKINIT implementation
  - KDC understands the proxy certificates
  - Kerberos tickets (and AFS tokens) can be used from the Grid world (job requesting access to a secured directory on AFS)
- We added a login script to the VOCE UI that creates user's kerberos tickets and AFS tokens automaticaly
  - Users can transparently use both the grid and *META Centre* facilities without further re-authentication