Security in Complex Distributed Systems

Milan Potocnik
AMRES/UOB

Poznan, 22.06.2010

connect • communicate • collaborate

Introduction

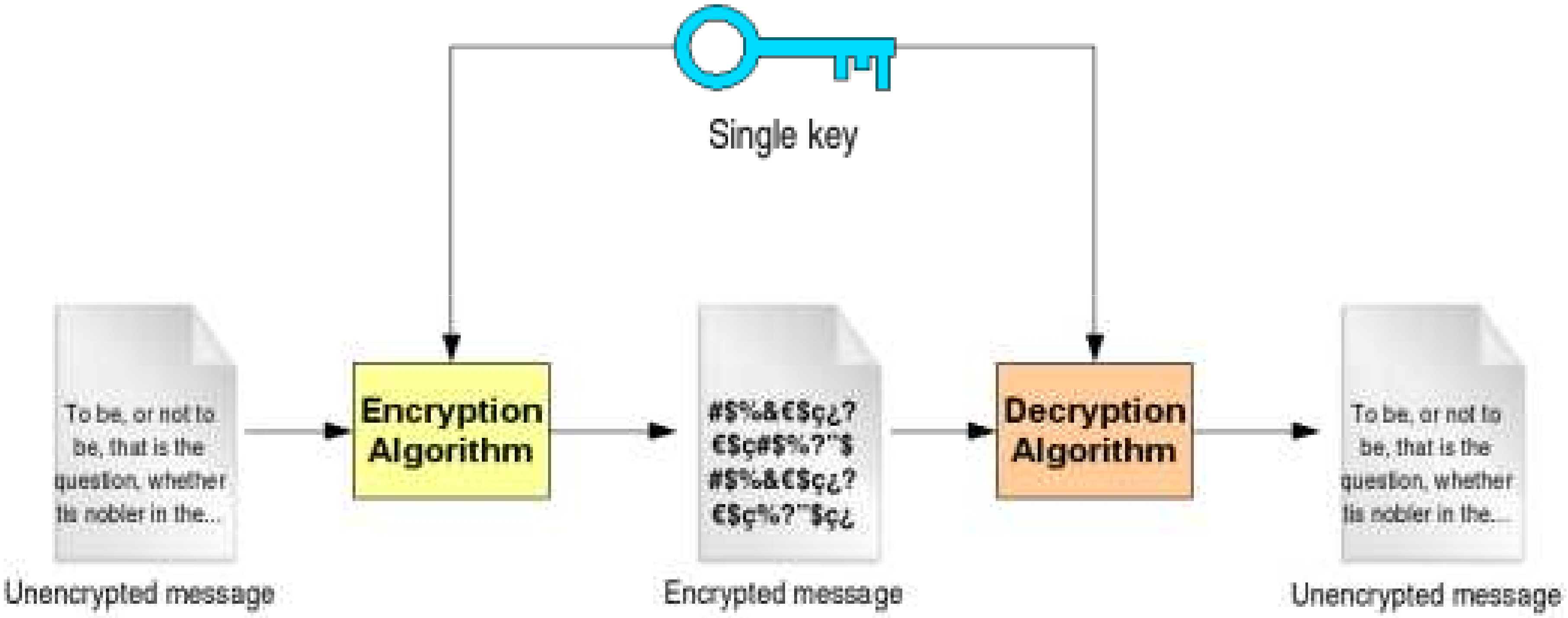
Pillars of secure communication:

- Privacy – only sender and receiver should be able to understand the communication.
- Integrity – receiver must know *for sure* that the message received is exactly the one that was sent.
- Authentication (with non-repudiation) – ensure the parties are who they claim to be.

Overview

- Privacy & Integrity via PKI
- Authentication & Authorization (PKI, LDAP, …)
- Proxy certificates
- Attribute Certificates
- Examples: EGEE grid security infrastructure, eduGAIN

Privacy – Symmetric Key Encryption

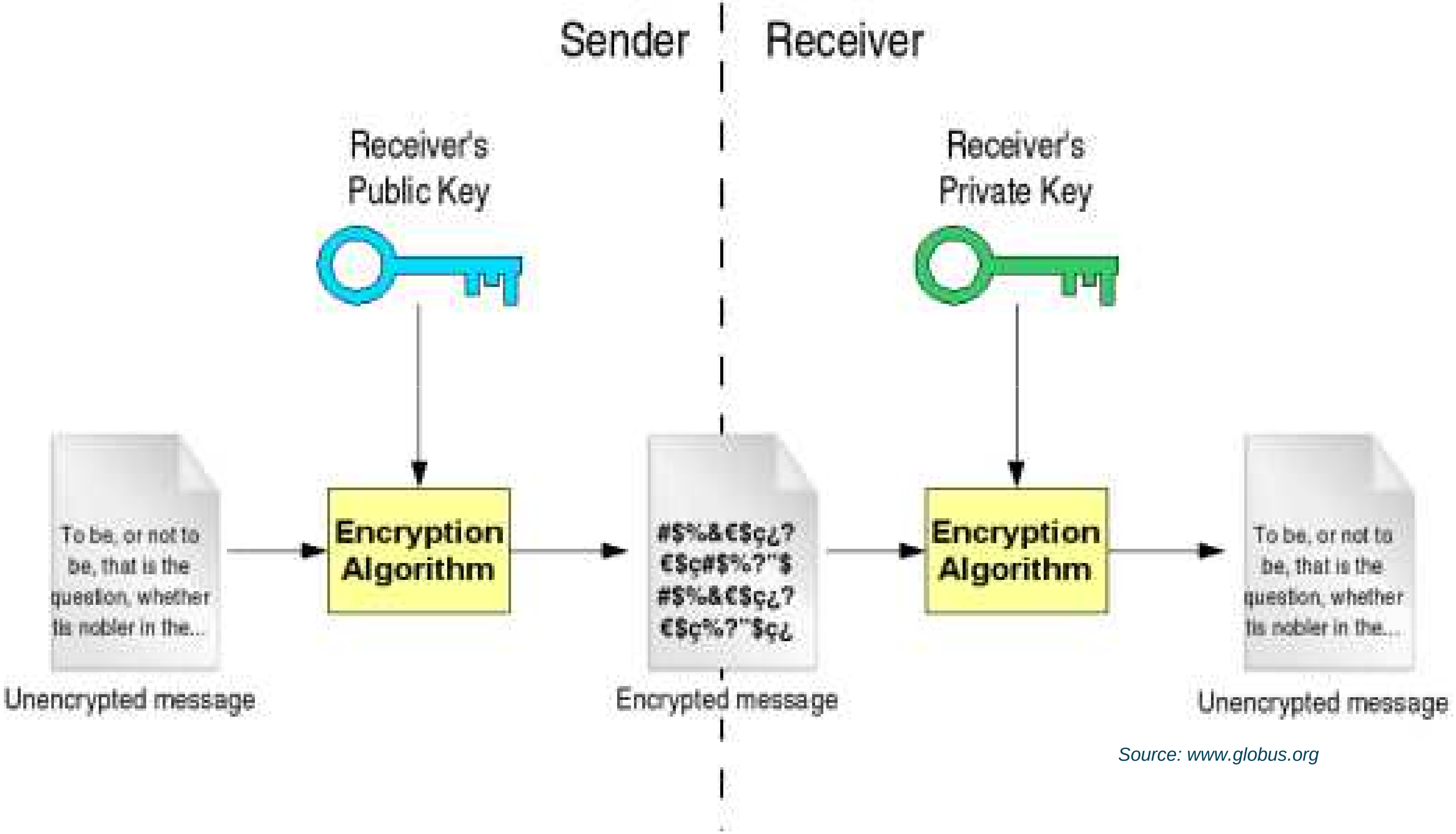

Single key
To be, or not to be, that is the question, whether tis nobler in the...
Unencrypted message
Encryption Algorithm
#$%&€$ç¿? €$ç#$%?"$ #$%&€$ç¿? €$ç%?"$ç¿
Encrypted message
Decryption Algorithm
To be, or not to be, that is the question, whether tis nobler in the...
Unencrypted message

Source: www.globus.org

Privacy – Symmetric Key Encryption (cont.)

- Fast and simple to implement.
- Provides privacy, but not integrity & authentication.
- Single private key:
  - Is shared between the sender and receiver.
  - Is used through the entire duration of secure communication.
- Mechanisms of distributing the private key are not a trivial problem.

Privacy – Asymmetric Key Encryption


Sender
Receiver
Receiver's Public Key
Receiver's Private Key
To be, or not to be, that is the question, whether tis nobler in the...
Encryption Algorithm
#$%&€$ç¿? €$ç#$%?"$ #$%&€$ç¿? €$ç%?"$ç¿
Encryption Algorithm
To be, or not to be, that is the question, whether tis nobler in the...
Unencrypted message
Encrypted message
Unencrypted message

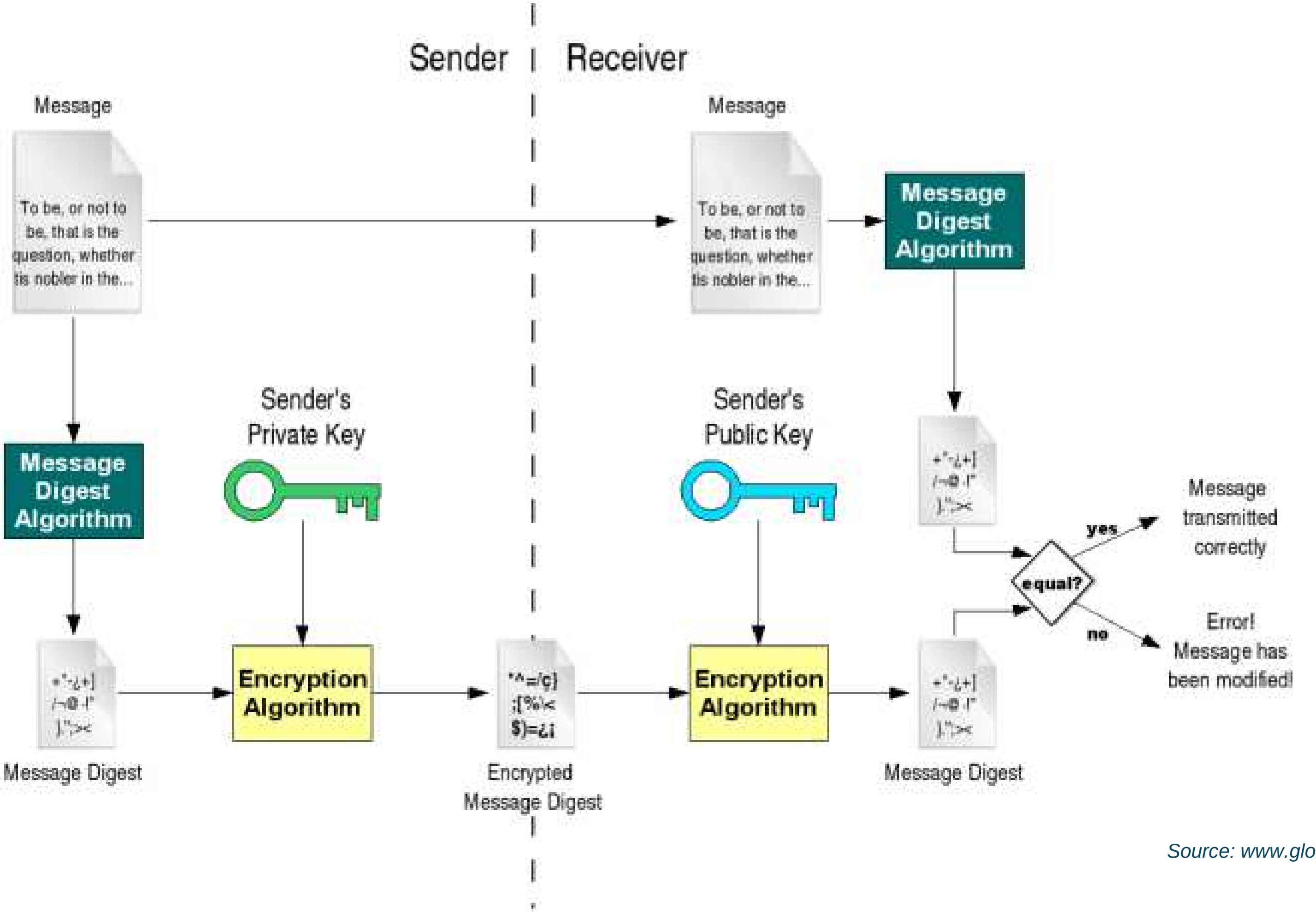
Source: www.globus.org

connect • communicate • collaborate

Privacy – Asymmetric Key Encryption (cont.)

- Public Key Cryptography – based on pairs of private and public keys.
- X.509 Public Key Infrastructure (PKI).
- Receiver's public key is used to encrypt a message.
- Can only be decrypted with the receiver's private key.
- No need to agree on a shared key.
- Public keys can be distributed.
- Private keys are always kept secret.
- Not as fast as symmetric systems.

Integrity – PKI



Source: www.globus.org

connect • communicate • collaborate

Integrity – PKI (cont)

PKI also provides integrity in secure communication via digital signatures:

- A *message digest* is generated – a hash of the message content.
- The message digest is encrypted using the sender's *private* key – digital signature.
- Receiver decrypts the digital signature with the senders public key.
- Message digest is calculated from the received message.
- If values of two digests are equal, message has not been tampered with by a third party.

Authentication & Authorization

- Authentication (auth) – Confirming an identity.
- Authorization (authz) – Deciding what tasks an entity can perform.
- Credential delegation – Delegating a set of credentials (user’s identity) to another entity.
- Single sign-on – log in and authenticate once to access multiple resources.

Using PKI for Authentication

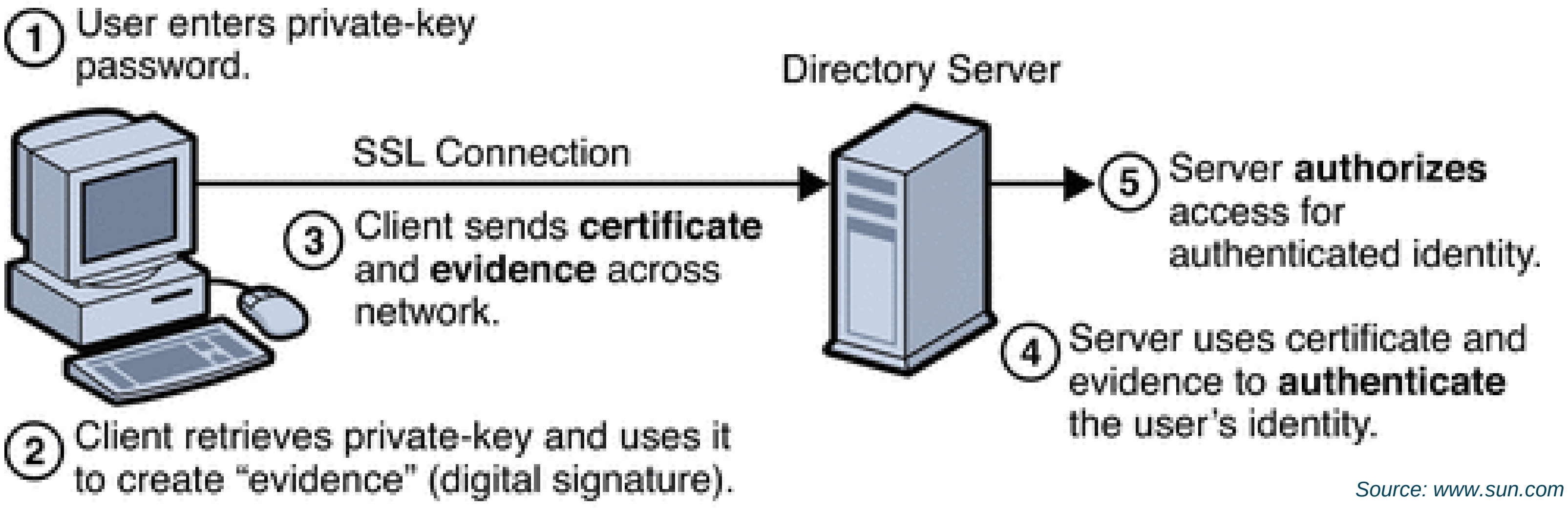
- *Digital certificate* (Public Key Certificate – PKC): document that *certifies* that a certain public key is owned by a particular entity – RFC 5280.
- Certificate is signed by a Certificate Authority (CA)
  - CA is a trusted third party who has an ability to issue and revoke certificates.
  - Every CA publishes a Certification Revocation List (CRL).
  - Registration Authority (RA) is used for interactions between entities identified by certificates.
  - CA also specifies a validity period – usually 1-2 years.

Using PKI for Authentication - Certificates

- Certificate contains fields like:
  - Subject : Distinguishing name (DN) in format: “C=US,O=ACME,OU=SALES,CN=John Smith”.
  - Subject's public key.
  - Issuer's Subject – CA's distinguished name.
  - Digital signature: of all the information in the certificate, generated using the CA's private key.
  - Validity period.
  - Certificate extensions: optional additional data.
- PKC can be bundled with the private key in various formats (PKCS12, JKS, …) or kept separately from the private key (PEM).

Using PKI for Authentication – Verifying Certificates

- In order to perform PKI authentication two steps are required:
  - Verification of entity's certificate.
  - Signed challenge mechanism.
- Sometimes an entity can provide a chain of certificates, instead of just its own certificate.


1 User enters private-key password.
Directory Server
SSL Connection
3 Client sends certificate and evidence across network.
5 Server authorizes access for authenticated identity.
4 Server uses certificate and evidence to authenticate the user's identity.
2 Client retrieves private-key and uses it to create "evidence" (digital signature).

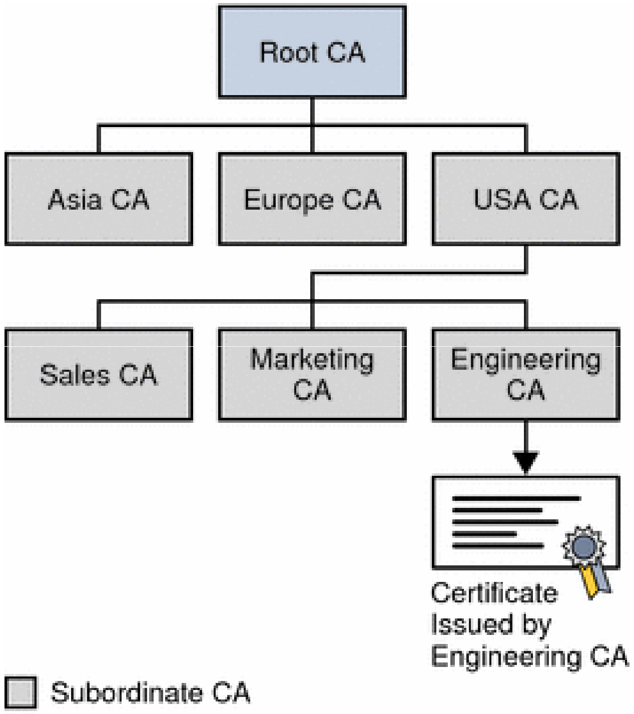
Source: www.sun.com

Using PKI for Authentication – Verifying Certificates (cont.)

Verifying a certificate is not trivial.

Some of the steps include:

- Has the certificate expired?
- Is the certificate trusted? If not, is the issuer (CA) trusted? Since CA's can be nested, the entire certificate chain needs to be verified until a trusted one is found.
- Is the certificate truly signed by the specified CA?
- For every CA in the chain a CRL needs to be checked. CRL also has a timeframe in which it is valid.



Source: www.sun.com

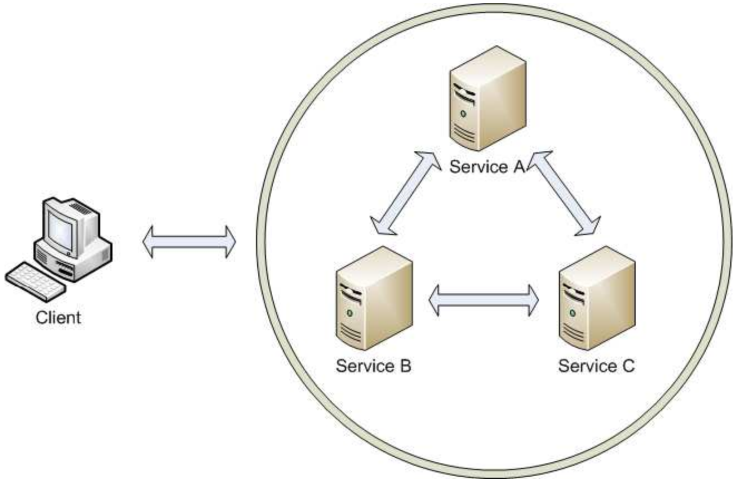
Using PKI for Authentication (cont.)

Benefits of using PKI for authentication:

- No sensitive information is transmitted over the network (passwords or private keys).
- No repository for user's credentials is needed for services – only trusted CA's need to be kept.
- No need for a centralized authentication service.
- PKI scales well in big distributed systems.

Alternative: password authentication. Options:

- Each service stores users' credentials (password hash). Password may differ from service to service – very difficult to administrate when there are many entities.
- A centralized authentication service needs to exist which creates a potential single point of failure.

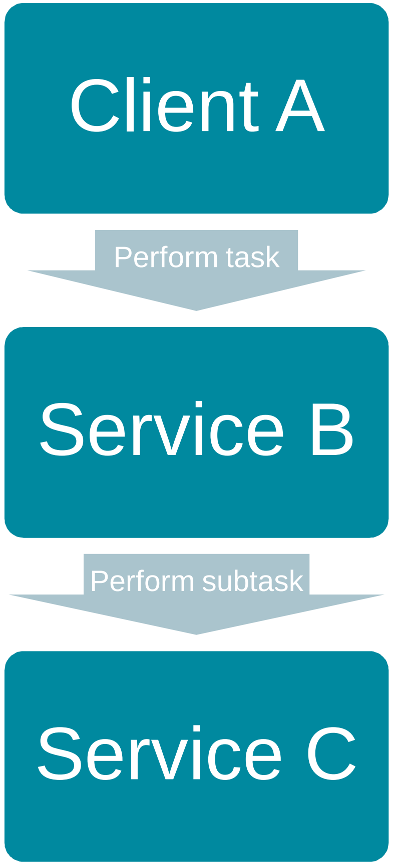


connect • communicate • collaborate

Credential Delegation

Delegation problem in distributed systems:

- Client A asks a Service B to perform a task.
- Service B splits a task into subtasks and assigns a subtask to Service C.
- Who authenticates to Service C and how?
- For security and accountability reasons it is expected that Client A authenticates to the Service C since subtask is performed in his/her behalf.
  - Service C could contact Client A for authentication every time someone delegates a subtask – not a very elegant solution.
  - Could Service B somehow authenticate to Service C with Client's A credentials (but without A's private key)?

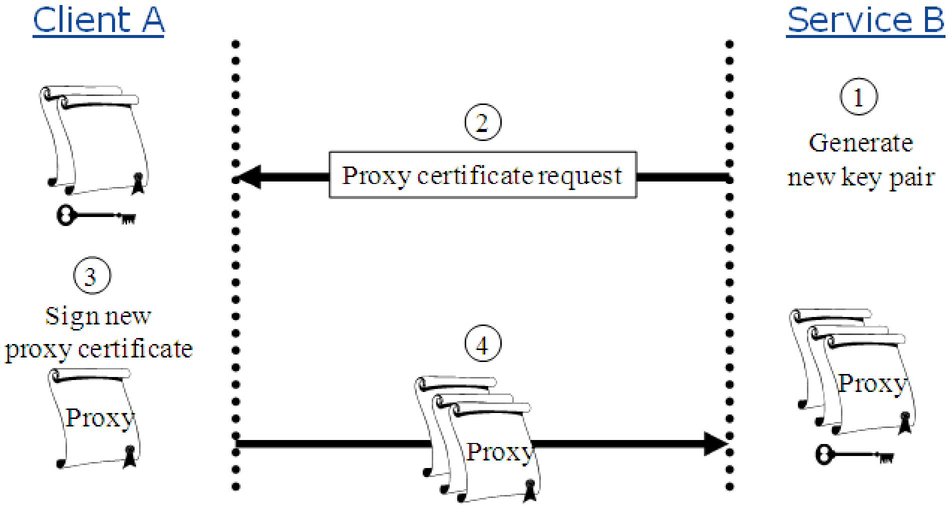


Proxy Certificate

- Proxy – “authority or power to act for another”.
- Proxy Certificate (PC) is is derived from, and signed by, a normal X.509 Public Key End Entity Certificate (EEC) or by another Proxy Certificate.
- PC can only sign another PC, it cannot sign an EEC.
- Provides restricted proxying, delegation and single sign-on within a PKI based authentication system.
- PC can also be used by the end entity itself.
  - Might not be so obvious but accessing PC to authenticate is safer then using EEC, password for the private key needs to be entered only once when PC is created, instead of every time EEC is needed.
  - Once created PC can be stored in user’s home directory with proper access privileges and be reused as long as it is valid.
- RFC 3820

Proxy Certificate (cont.)

- Private-public key pair is generated specifically for the proxy certificate by Service B.
- Service B uses the key pair to generate a certificate request, which will be sent to A using a secure channel.
- Supposing A agrees to delegate its credentials to B, Client A will use its private key to digitally sign the certificate request.
- A sends the signed certificate back to B using a secure channel.



connect • communicate • collaborate

Proxy Certificate (cont.)

Security pros and cons:

- Proxy certificate is a security compromise – allowing someone else to act on your behalf is risky.
- If proxy is stolen (proxy is usually not password protected) a malicious third party can act on the behalf of the original EEC owner.
- A compromised PC private key does not compromise the EEC private key.
- They have a much shorter lifetime compared to a regular EEC (usually 12 hours, compared to 1-2 years of EEC).
- An EEC or PC can limit what a new PC can be used for by turning off bits in the Key Usage and Extended Key Usage extensions.

Authorization

- Even with PKI, there is no unique solution for authorization.
- Resource should have the right to be able choose who can use it.
- Possible solutions (can be combined together):
  - Localized authorization filters (white/black lists).
  - Centralized authorization service (LDAP, …).
  - PKI – Proxy and Attribute Certificates.

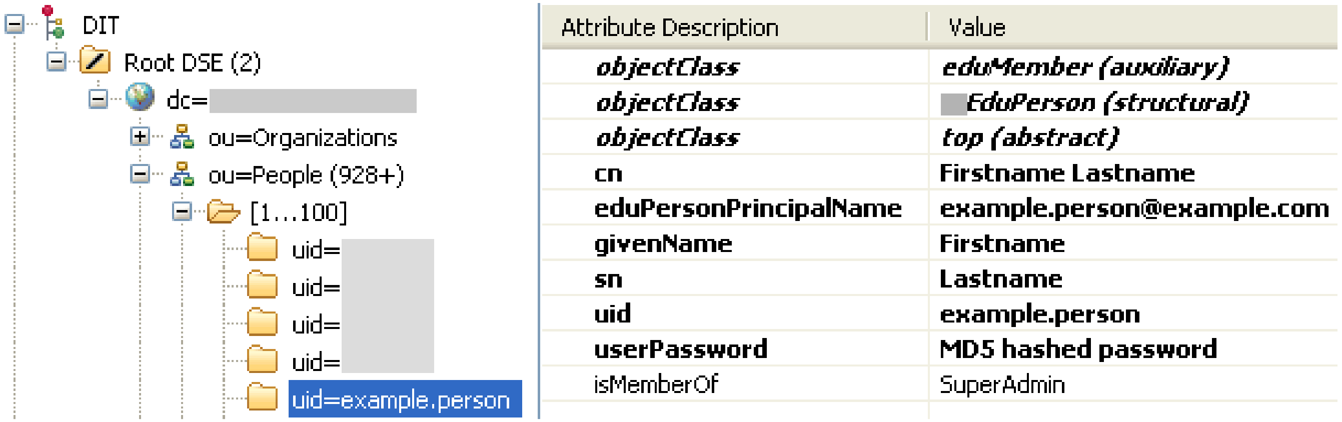
Authorization – LDAP (1)

- Lightweight Directory Access Protocol (LDAP): an Internet protocol for accessing distributed directory services that act in accordance with X.500 data and service models.
- A directory is a set of objects with attributes organized in a logical and hierarchical manner – e.g. telephone directory.
- LDAP can be used to store any form of information, but is designed for directories.
  - Small bits of data.
  - Mostly read access.
- Each object in the LDAP directory has a DN
  - uid=jheiss,ou=people,dc=example,dc=com
  - cn=users,ou=group,dc=example,dc=com
- RFC 4510

Authorization – LDAP (2)

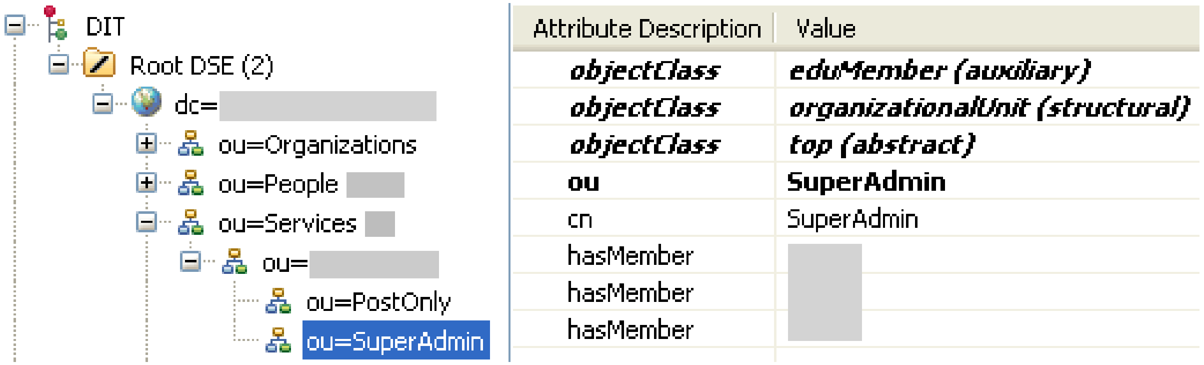
- LDAP can be used for:
  - Password authentication & authorization.
  - PKI authorization.
- Most implementations of LDAP servers can be configured to work with PKI certificates.
- LDAP authorization in PKI infrastructure:
  - Entity is authenticated using regular PKI authentication mechanisms.
  - With LDAP lookup, directory entry is found corresponding to the entity's DN.
  - Entry can contain authorization information (roles, groups, etc…).
  - Authorization information is dispatched from the LDAP.
- LDAP can also be part of the PKI authentication.

Authorization – LDAP example



- DIT
  - Root DSE (2)
    - dc=[redacted]
      - ou=Organizations
      - ou=People (928+)
        - [1...100]
          - uid=[redacted]
          - uid=[redacted]
          - uid=[redacted]
          - uid=[redacted]
          - uid=example.person

Attribute Description	Value
objectClass	***eduMember (auxiliary)***
objectClass	***[redacted]EduPerson (structural)***
objectClass	***top (abstract)***
cn	**Firstname Lastname**
eduPersonPrincipalName	**example.person@example.com**
givenName	**Firstname**
sn	**Lastname**
uid	**example.person**
userPassword	**MD5 hashed password**
isMemberOf	SuperAdmin



- DIT
  - Root DSE (2)
    - dc=[redacted]
      - ou=Organizations
      - ou=People [redacted]
      - ou=Services [redacted]
        - ou=[redacted]
          - ou=PostOnly
          - ou=SuperAdmin

Attribute Description	Value
objectClass	***eduMember (auxiliary)***
objectClass	***organizationalUnit (structural)***
objectClass	***top (abstract)***
ou	**SuperAdmin**
cn	SuperAdmin
hasMember	[redacted]
hasMember	[redacted]
hasMember	[redacted]

connect • communicate • collaborate

Authorization – Proxy & Attribute Certificate

- Proxy certificate can be used for authorization:
  - DN of a PC can be used for authorization.
  - PC are easily integrated into existing PKI systems, because they are regular certificates with some additional features.
  - PC contains a policy field which can be also used for authorization.
  - PC authorization rights are created by EEC or another PC, however, sometimes authorization rights need to be assigned by a third party.

Authorization – Proxy & Attribute Certificate (cont.)

- Attribute certificate (AC) is a structure similar to a PKC, the main difference being that the AC contains no public key (it has a Holder field that identifies the subject).
- An AC is simply a digitally signed (certified) identity and set of attributes.
  - An AC may contain attributes that specify group membership, role, security clearance, or other authorization information associated with the AC holder.
  - AC is usually signed by a third party, an authorization service called the Attribute Authority (AA).
  - AC can also be signed by an EEC, however this is not a very common practice.
- AC may be placed in a PKC as an extension or held separately.
- RFC 3281

EGEE Grid – Introduction

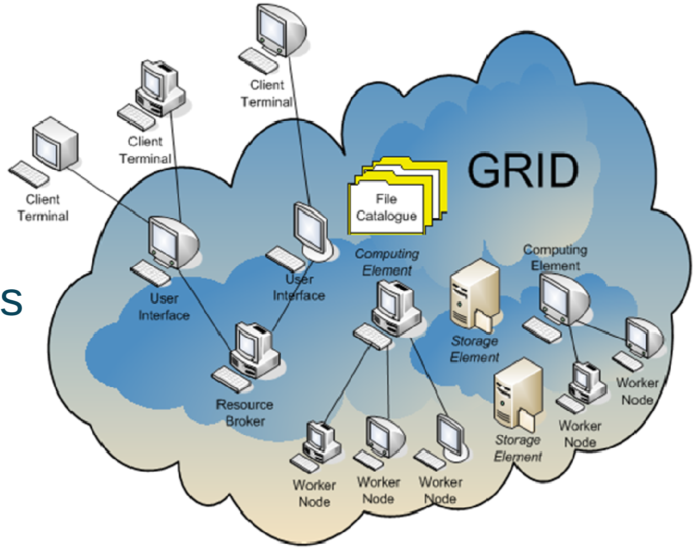
- Grid is a scalable distributed system that provides a mechanism of finding and accessing various resources.
- Key features that identify it:
  - Resources are not subject to centralized control – they are managed by many different (localized) entities.
  - Uses standard, open, general purpose protocol and communication interfaces – not application specific.
  - Able to deliver non trivial qualities of service – resources can be chosen based on response time, data throughput, availability, etc…

EGEE Grid – Introduction (cont.)

- Enabling Grids for E-sciencE (EGEE) grid is a project developed alongside the LHC project.
  - Consists of hardware and software components that together make the grid middleware (gLite middleware).
  - Main resources (services) in gLite middleware are known as Computing Elements (CE). These provide processing power and data resources known as Storage Elements (SE).
  - All other resources in gLite middleware are there to provide easier access to CEs and SEs - a “glue” between various elements.
  - User can precisely specify what kind of task it needs to compute or which data to manipulate.

EGEE Grid – Authentication

- Localized grid resources are called a grid site.
- User Interface (UI) is an entry point for grid users, where their credentials are stored.
- Grid entities (users and host machines) are authenticated using X.509 PKI – called Grid Security Infrastructure (GSI).
- CAs are regionally based – each country has at least one CA.
- Every machine on the grid has the CA software package installed. It is provided by EUGridPMA:
  - It contains a directory of trusted CA certificates along with their corresponding CRL's.
  - CRL's are continually refreshed (downloaded).



EGEE Grid – Authorization

- Grid users are managed within Virtual Organizations (VO).
- Each VO is managed by a Virtual Organization Membership Service (VOMS). This is the main authorization mechanism in the grid infrastructure.
- Users within a VO can belong to different groups and have specific roles.
- Access to grid resources is controlled by VO membership.
- VOMS service is an Attribute Authority capable of issuing the AC. Attributes in the AC are actually information about VOMS groups and roles.
- To access grid resources through the UI, user creates a proxy certificate that is stored in the file system: the VOMS proxy.
- VOMS proxies can be used for single sign on, delegation and authorization.

EGEE Grid – VOMS Proxy Example

```
$ voms-proxy-init -voms seegrid:/seegrid/Role=sgmadmin
Enter GRID pass phrase:
Your identity: /C=RS/O=AEGIS/OU=UOB/CN=Milan Potocnik
Creating temporary proxy .......................................... Done
Contacting  voms.irb.hr:15010 [/C=HR/O=edu/OU=irb/CN=host/voms.irb.hr] "seegrid" Done
Creating proxy .................................. Done
Your proxy is valid until Thu Jun 10 13:06:54 2010

$ voms-proxy-info -all
subject   : /C=RS/O=AEGIS/OU=UOB/CN=Milan Potocnik/CN=proxy
issuer    : /C=RS/O=AEGIS/OU=UOB/CN=Milan Potocnik
identity  : /C=RS/O=AEGIS/OU=UOB/CN=Milan Potocnik
type      : proxy
strength  : 1024 bits
path      : /tmp/x509up_u508
timeleft  : 11:59:48
=== VO seegrid extension information ===
VO        : seegrid
subject   : /C=RS/O=AEGIS/OU=UOB/CN=Milan Potocnik
issuer    : /C=HR/O=edu/OU=irb/CN=host/voms.irb.hr
attribute : /seegrid/Role=sgmadmin/Capability=NULL
attribute : /seegrid/RS/Role=NULL/Capability=NULL
attribute : /seegrid/RS/App/Role=NULL/Capability=NULL
attribute : /seegrid/RS/App/PROPEL/Role=NULL/Capability=NULL
attribute : /seegrid/Role=NULL/Capability=NULL
timeleft  : 11:59:48
uri       : voms.irb.hr:15010
```

eduGAIN

- Authentication and authorization infrastructure for cross-national access to network services, focusing initially on European level.
- Actors:
  - eduGAIN – mediator between federations.
  - Federations – typically NRENs.
  - Identity Providers – one or several within a federation.
  - Service Providers – many, all within a federation.
  - End users – primarily humans.
- eduGAIN provides
  - Infrastructure for establishing trusted communications between Identity and Service Providers from different participant Federations.
  - Technical and policy framework.

eduGAIN – Scope and Participation

- The expected use of the eduGAIN service is federated management of access to web-based services.
- Will be extended in the future to accommodate more advanced use-cases, such as non web-based services.
- Identity and Service Providers are always registered to the Participant Federations, which may make them visible via eduGAIN.
- Participant Federations are expected to apply an opt-in principle, in other words, Providers are expected to take an active step to get exposed to eduGAIN.

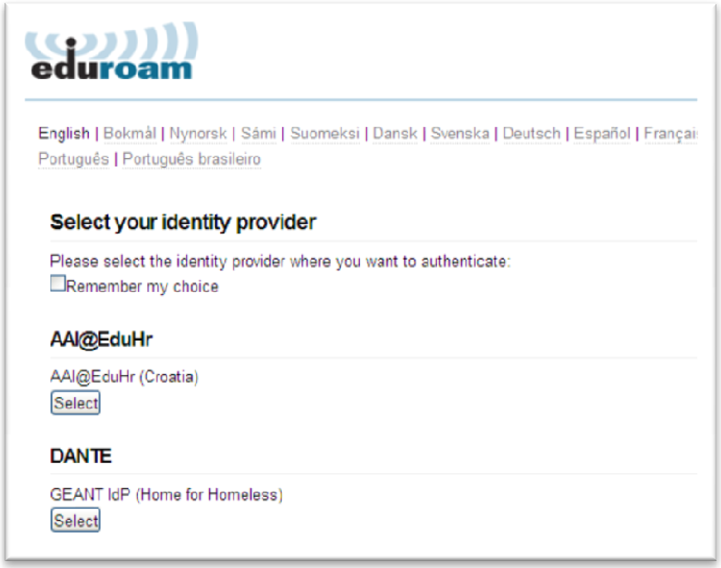
eduGAIN – Metadata Aggregation

- All communication using OASIS Security Assertion Mark-up Language (SAML).
- eduGAIN is managed by aggregating and distributing signed SAML 2.0 metadata files.
  - Federations will publish their metadata to MDS (Metadata Service).
  - The available metadata of participating federations will be published through the MDS.

eduGAIN – Authentication

- Federations make the eduGAIN metadata available to its own member organizations.
- End to end authentication with SAML 2.0
  - Web-based services interpret the MDS metadata and let users select an Identity Provider.
  - End users authenticate directly at Identity Providers and get access to Service Providers.

eduGAIN – Example



eduroam

English | Bokmål | Nynorsk | Sámi | Suomeksi | Dansk | Svenska | Deutsch | Español | Françai
Português | Português brasileiro

Select your identity provider

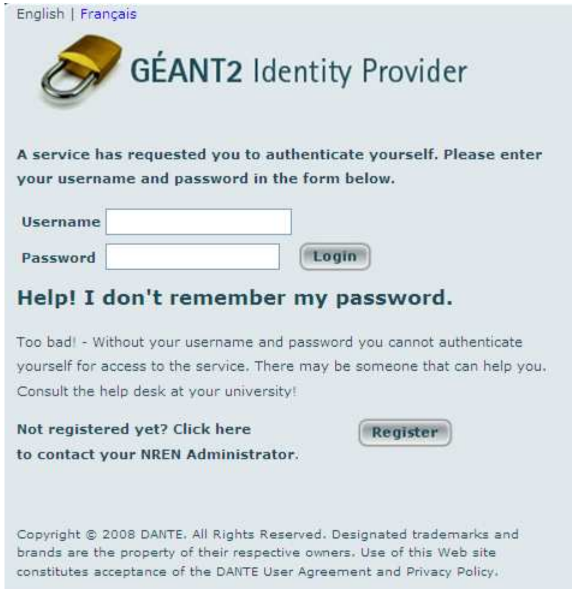
Please select the identity provider where you want to authenticate:
☐Remember my choice

AAI@EduHr

AAI@EduHr (Croatia)
Select

DANTE

GEANT IdP (Home for Homeless)
Select



English | Français

GÉANT2 Identity Provider

A service has requested you to authenticate yourself. Please enter your username and password in the form below.

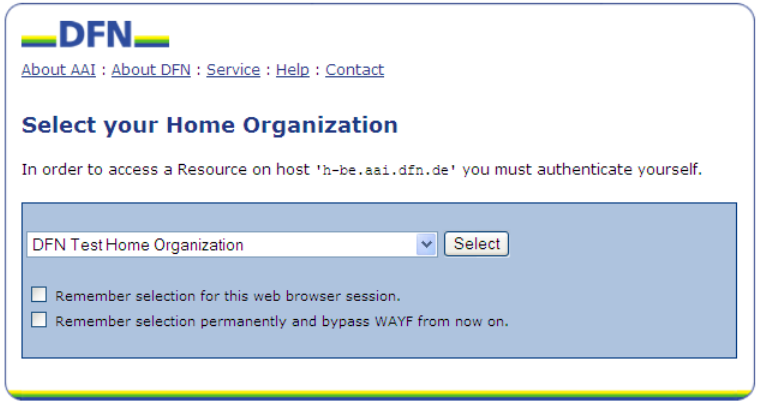
Username
Password Login

Help! I don't remember my password.

Too bad! - Without your username and password you cannot authenticate yourself for access to the service. There may be someone that can help you. Consult the help desk at your university!

Not registered yet? Click here to contact your NREN Administrator. Register

Copyright © 2008 DANTE. All Rights Reserved. Designated trademarks and brands are the property of their respective owners. Use of this Web site constitutes acceptance of the DANTE User Agreement and Privacy Policy.



DFN

About AAI : About DFN : Service : Help : Contact

Select your Home Organization

In order to access a Resource on host 'h-be.aai.dfn.de' you must authenticate yourself.

DFN Test Home Organization Select

☐ Remember selection for this web browser session.
☐ Remember selection permanently and bypass WAYF from now on.

eduGAIN – Authorization

- Authorization support will be augmented in the process.
- Authorization usually needs attributes. eduGAIN:
  - Recommended attribute profile for end users' attributes exchanged.
  - Optional data protection profile that aims to fulfil the EC data protection directive.
- Usage of VOs is anticipated:
  - VOs would have own membership and authorization management services.
  - Usage directly bound to applications.

eduGAIN – Attribute Usage (Draft)

- It is RECOMMENDED that eduGAIN participant federations make sure that Identity Providers have following attributes populated:
  - cn
  - email
  - eduPersonAffiliation
  - eduPersonScopedAffiliation
  - schacHomeOrganization
  - schacHomeOrganizationType
  - displayName
- Attributes defined in eduPerson and schac MAY be used in eduGAIN.
- Other attributes MAY be used based on a bilateral agreement between the Identity and Service Providers.
- Application developers are advised to make fail-safe code.
  - Fall-back mechanism if an Identity Provider does not provide an attribute the Service Provider is asking for.