OSG Site Administrators Workshop
Vanderbilt, 2010

Security

infrastructure, certificates and recommendations

Igor Sfiligoi

for the OSG Security team

OSG Security

Part One

OSG Security model

A high level overview

OSG Security model

- Multiple administrative domains;
  each Site
  - Decides how to run its own resources
  - Decides which users to support
- Federated trust
  - Too many users and too many sites to require each user to register at each site
  - Virtual Organizations (VOs) as a middle man
    - A VO trusts its own users
    - A Site trusts a VO

Authentication structure

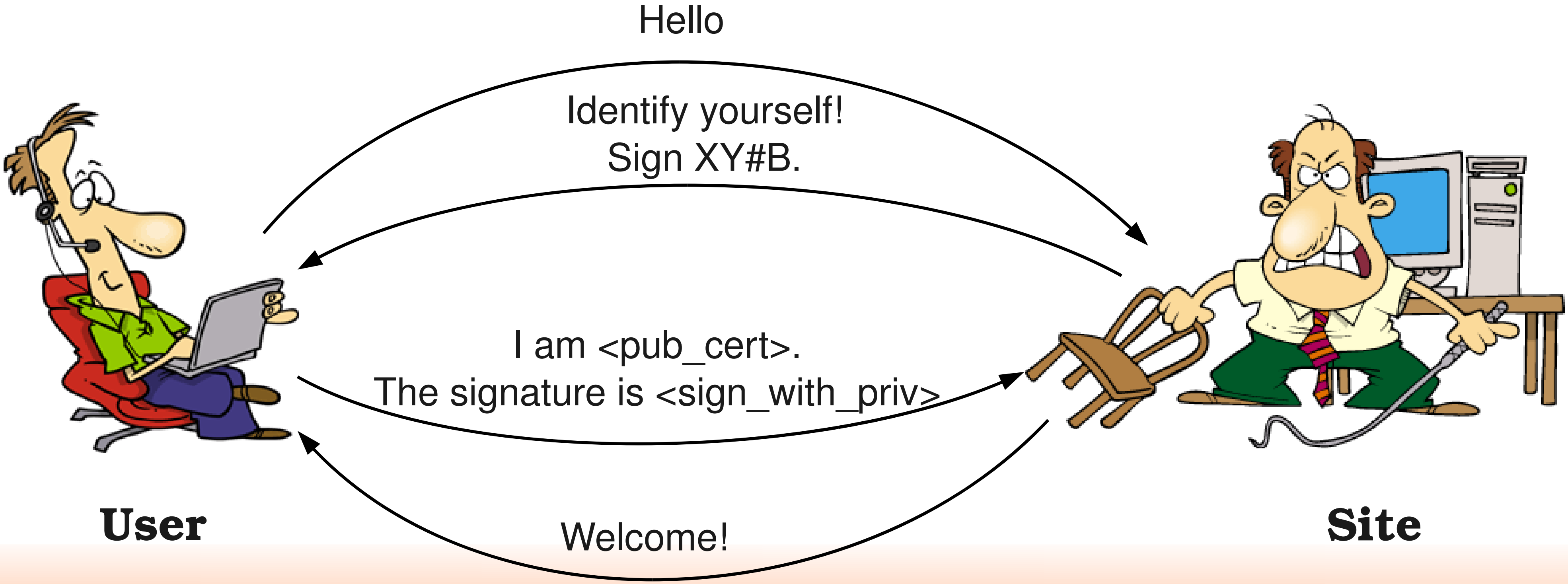
- Users want a single sign-on to run on all sites
  - Remember, they are not registering with all the sites
- Username+password cannot be used
  - That would require all sites to synchronize the password/shadow files -> not practical
- Public Key Infrastructure (PKI) used instead
  - In particular X.509 certificates and proxies
  - Sites only need to know the “user name”
    - PKI takes care of the security aspect

PKI – x.509 certificate

- The user is issued a certificate, which is composed of 2 parts:
  - A public part, containing
    - The user name (also known as the **DN**)
    - Validity period
    - The public key
    - The signing chain (more on this later)
  - A private part (containing the private key)
- **The private part MUST be kept private**
  - The public part can (and will) be sent around

PKI – How it works?

- User proves who he is by signing using the private key
  - The public key in the pub_cert allows for verification



PKI – What is a signature?

- A digital signature proves who you are
  - Because **only you own the private key**
- It is strongly correlated to the public key
  - Not enough time to go into technical details here, consult wikipedia if interested: http://en.wikipedia.org/wiki/Digital_signature

PKI – Signature validation

- The site must validate the signature
  - Else the user may just fake it!
- So the Site uses the public key sent by the user to do the validation
  - **But why should a site trust the public key sent?**
- The public key itself is signed by a trusted entity (in the signing chain)
  - By a **trusted** Certification Authority (CA)
  - The site must already have
    the CA public key **pre-installed** locally
    (typically getting it through the OS or the VDT)

PKI – What is a CA?

Not all CAs are trusted!

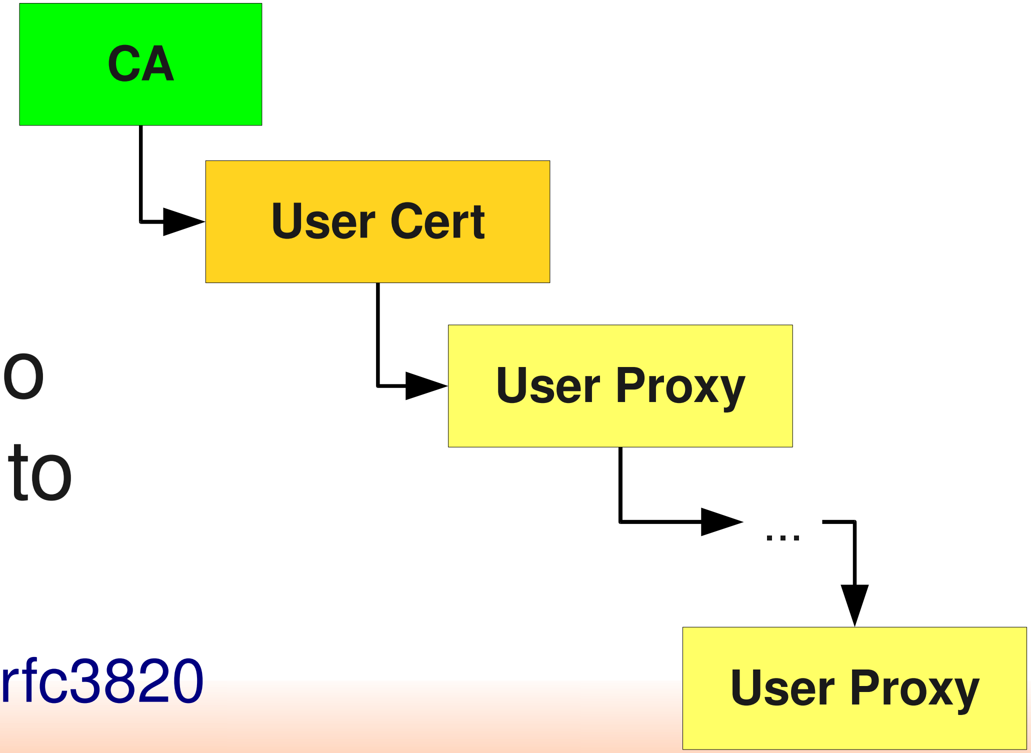
- A CA is someone who issues certificates
- A **trusted** CA is someone who you trust to issue user certificates **only if** they know that user
  - i.e. User **X** cannot get a certificate with username **Y**
- There are relatively few **trusted** CAs in existence
  - At least compared to the number of users
  - Pre-installing their public keys is thus manageable
- A CA can also revoke a user certificate
  - By publishing its public key in a Certificate Revocation List (CRL)
  - Make sure you download the updated CRLs often!

Self signed certs not issued by a trusted CA

PKI – And what is a proxy?

- You probably have heard about proxies
- A proxy is just a new certificate derived from a user certificate
  - Possibly many times!
- The signing chain contains the info to safely climb back to the CA

http://tools.ietf.org/html/rfc3820



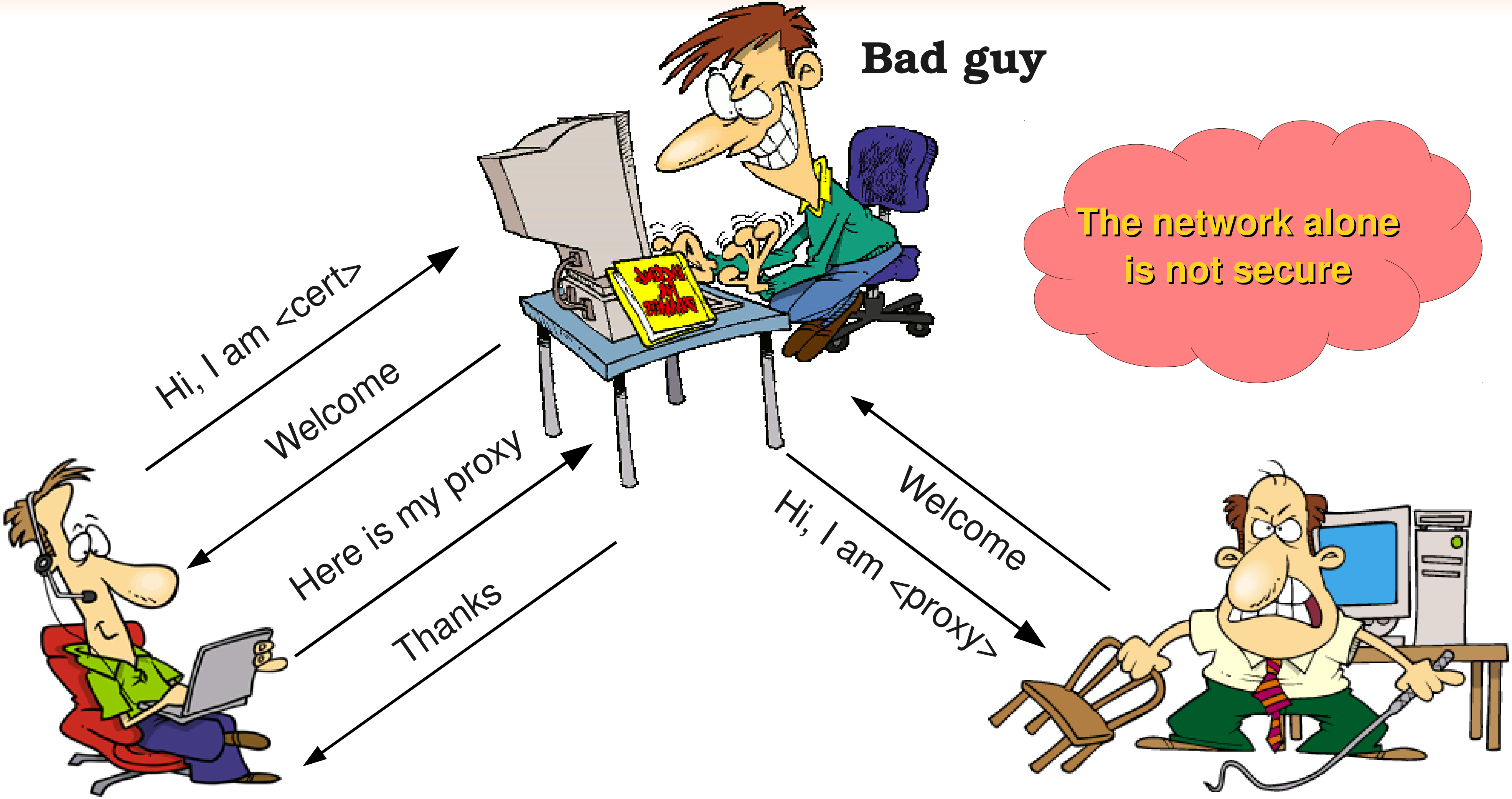
PKI – Why a proxy?

- The user jobs may need to talk to a remote service when running on the worker nodes
  - But cannot access the user cert's private key!
- A proxy is thus sent (delegated) with the job to the worker node
  - **And the proxy contains a private key!**
  - So the job can impersonate the user
- Of course, delegating a private key is dangerous
  - Mitigated by the fact that proxy lifetime is short (much shorter than the user certificate one)

PKI – Sites have certificates, too

- Security only if mutual authentication
  - The Site trusts the User and
    the User trusts the Site
- The Site must prove who he is to the User
  - Especially if a proxy is being delegated there!
- All nodes with services at a Site thus need a host or service certificate
  - Similar to a user certificate, but issued by a CA for a specific DNS host
    (can only be used on that DNS address)

Example: One way authentication



User **Site**

Authorization

- Just because someone can authenticate, does not mean a Site will authorize him/her to run on its resources
  - Authorization is a separate step
- The Site may also want to give different privileges to different users
  - The user must be mapped to a local security domain
  - Certificate DN -> (typically) UNIX UID

VO-based Authorization

- As mentioned in the introduction, Sites trust VOs (not users directly)
  - Each VO will keep a list of trusted user DNs
  - Through a service called **VOMS**
- OSG provides a list of trusted VOs and their VOMS servers
  - The Site needs to pick which VOs to support
  - Should always support the MIS VO (OSG operations)
- Users authenticate with a VOMS-extended proxy (voms-proxy-init -voms ...)

Mapping

- OSG provides **GUMS** for mapping
  - Talks to VOMS servers to get the list of user DNs
- Site admin must decide the mapping
  - Still VO based, possibly based on VO groups
  - Either pool **(recommended)** or group mappings
- The admin must also create all the necessary UNIX accounts
  - Part of *"administrative autonomy"* principle

Pool vs group mapping

- **Pool mapping** maps each user to a different UNIX username/UID
  - Something like *uscms0001*,...,*uscms2345*
  - May need lots of accounts!
- **Group mapping** maps all the users to the same UNIX username/UID
  - Something like *mis*
  - No protection between users
- Pool accounts recommended (unless VO explicitly asks for a group account)

Additional reading

- OSG Certificate page
  https://twiki.grid.iu.edu/bin/view/ReleaseDocumentation/CertificateWhatIs
- Wikipedia X.509 description
  http://en.wikipedia.org/wiki/X.509
- A talk about VOs
  http://staff.science.uva.nl/~demch/presentations/cts2006-ydemchenko-vo-dynamic-associations01.pdf
- OSG Security Home page
  https://twiki.grid.iu.edu/twiki/bin/view/Security/

OSG Security

Part two

Technical details

Which CAs do we use

- DOEGrids CA (OSG Recommended)
  - https://pki1.doegrids.org/ca/
- CERN CA (Used by WLCG)
  - https://ca.cern.ch/ca/
- Fermilab CA (Fermilab-based users)
  - Converts krb5 tickets into certificates
- Foreign Country CAs
  - Each country has at least one CA
- Commercial CAs
  - Verisign, Thawte, GoDaddy, etc.

CAs supported as a Site

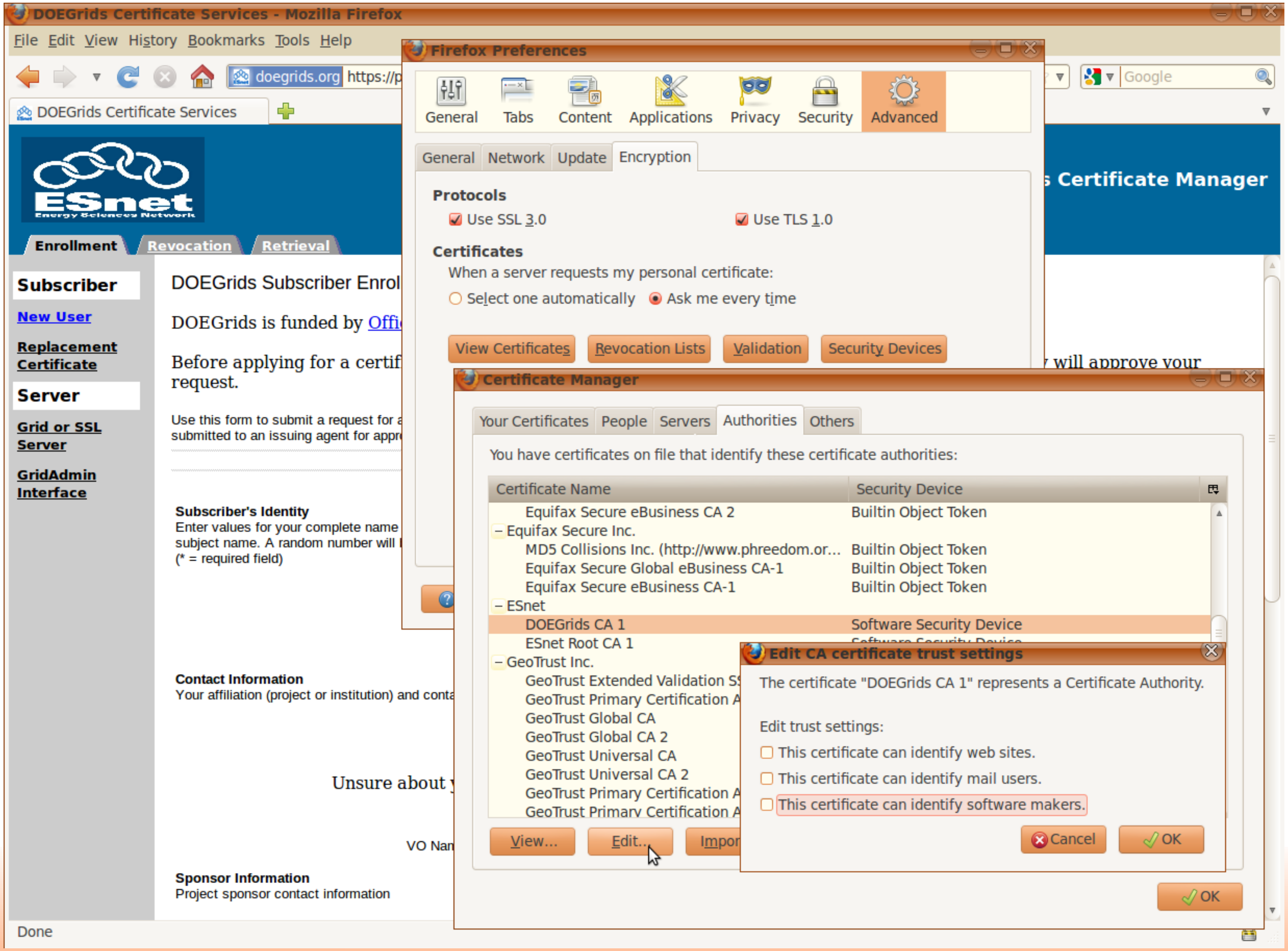
- OSG provides a list of trusted CAs known to be used by OSG-affiliated VOs
  - Get them trough VDT
    http://software.grid.iu.edu/pacman/cadist/ca-certs-version
- You likely want to support all those CAs
  - But you are free to remove the ones you know are not being used
  - And add additional ones for non-OSG users
- Make sure you keep the CRLs updated
  - *fetch-crl*

CAs supported as a User

- Users use certificates through two interfaces
  - Command line
  - Web browser
- Command line based on VDT
  - See previous slide
- Web browser mostly for Web pages
  - Commercial CAs come with the OS
  - The other CAs need to be imported https://www.tacar.org/repos/

Installed but disabled CAs

- Some browsers require explicit CA activation



DOEGrids Subscriber Enrol

DOEGrids is funded by Offi

Before applying for a certif request.

Use this form to submit a request for a submitted to an issuing agent for appr

Subscriber's Identity
Enter values for your complete name subject name. A random number will (* = required field)

Contact Information
Your affiliation (project or institution) and conta

Unsure about y

VO Nan

Sponsor Information
Project sponsor contact information

y will approve your

Protocols

☑ Use SSL 3.0 ☑ Use TLS 1.0

Certificates

When a server requests my personal certificate:

○ Select one automatically ● Ask me every time

You have certificates on file that identify these certificate authorities:

Certificate Name	Security Device
Equifax Secure eBusiness CA 2	Builtin Object Token
– Equifax Secure Inc.	
MD5 Collisions Inc. (http://www.phreedom.or...	Builtin Object Token
Equifax Secure Global eBusiness CA-1	Builtin Object Token
Equifax Secure eBusiness CA-1	Builtin Object Token
– ESnet	
DOEGrids CA 1	Software Security Device
ESnet Root CA 1	Software Security Device
– GeoTrust Inc.	
GeoTrust Extended Validation SS	
GeoTrust Primary Certification A	
GeoTrust Global CA	
GeoTrust Global CA 2	
GeoTrust Universal CA	
GeoTrust Universal CA 2	
GeoTrust Primary Certification A	
GeoTrust Primary Certification A	

The certificate "DOEGrids CA 1" represents a Certificate Authority.

Edit trust settings:

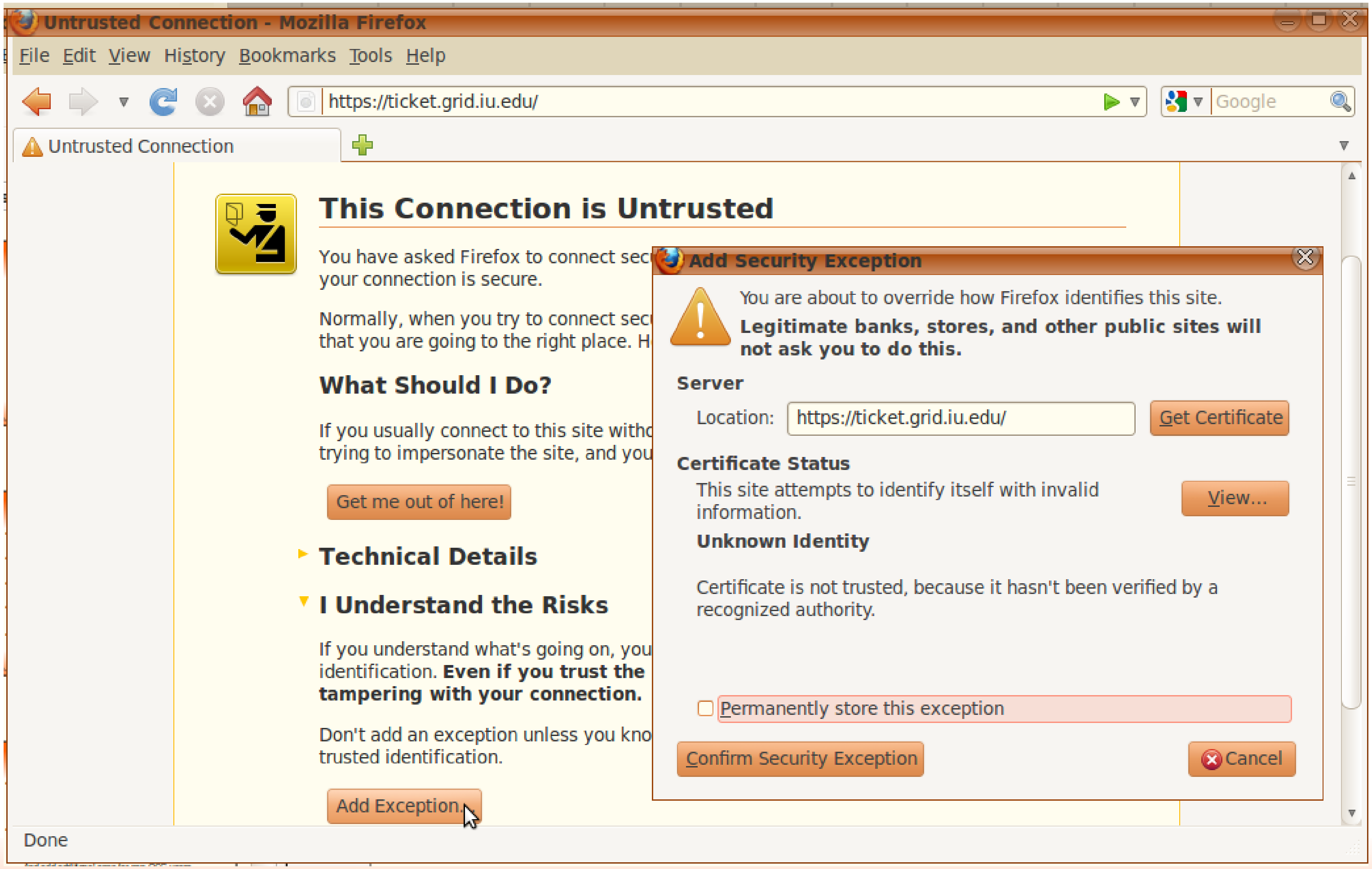
☐ This certificate can identify web sites.

☐ This certificate can identify mail users.

☐ This certificate can identify software makers.

Browser security

- Do not override browser security!



Untrusted Connection - Mozilla Firefox
File Edit View History Bookmarks Tools Help
https://ticket.grid.iu.edu/
Google
Untrusted Connection

This Connection is Untrusted

You have asked Firefox to connect sec
your connection is secure.

Normally, when you try to connect sec
that you are going to the right place. H

What Should I Do?

If you usually connect to this site witho
trying to impersonate the site, and you

Technical Details

I Understand the Risks

If you understand what's going on, you
identification. **Even if you trust the**
tampering with your connection.

Don't add an exception unless you kno
trusted identification.

Add Security Exception

You are about to override how Firefox identifies this site.
Legitimate banks, stores, and other public sites will not ask you to do this.

Server

Location: https://ticket.grid.iu.edu/

Certificate Status

This site attempts to identify itself with invalid information.

Unknown Identity

Certificate is not trusted, because it hasn't been verified by a recognized authority.

☐ Permanently store this exception

Requesting a certificate

- You likely want to use DOEGrids
  - Both for personal and service certificates
- You can request them either
  trough the Web interface or
  https://twiki.grid.iu.edu/bin/view/ReleaseDocumentation/CertificateGetWeb
  trough the command line interface
  https://twiki.grid.iu.edu/bin/view/ReleaseDocumentation/CertificateGetCmd
  - Command line easier for bulk requests
    (e.g. for service certificates)

How long does it take?

- Getting a certificate can take days
  - So plan accordingly
- Delay due to security implications
  - Someone must validate your request/identity
  - A Registration Agent (RA)
    typically associated with the VO
- For user certificates you also need to register with the VO VOMS server
  - Procedure VO-specific

Certificate format

- Two formats
  - .p12 – single file, containing both public and private part
  - .pem – two files, one for public (cert.pem) and one for private part (key.pem)
- .p12 and key.pem must be private to the user
  - No group or world read permissions!
- Can convert between them

```
openssl pkcs12 -clcerts -nokeys -in cert.p12 -out usercert.pem
openssl pkcs12 -nocerts -in cert.p12 -out userkey.pem
```

Services accepting certificates

- Compute Element (CE)/ Globus Gatekeeper
  - Submits jobs to the local batch system
  - Handles user proxies
- Storage Element (SE)/ SRM/ gridFTP
  - Interface to the disk storage area
- Web server
  (optional)
- All of the above need a service certificate

No sharing of service certificates

- A service certificate is released for a specific DNS address
  - Like *osgce.ucsd.edu*
- You cannot reuse it for on a different node
  - For example *www.ucsd.edu*
  - The certificate validation will fail

Additional reading

- OSG Security and Certificates FAQ
  https://twiki.grid.iu.edu/bin/view/Documentation/OsgFaq#Security_and_Certificates
- OSG Certificate Request Documentation
  https://twiki.grid.iu.edu/bin/view/ReleaseDocumentation/CertificateGet
- NCSA OpenSSL Cheatbook
  http://security.ncsa.illinois.edu/research/grid-howtos/usefulopenssl.html

Part three

Security recommendations

What is security?

- Security is much more than just technology
  - It is as much a social problem
- We have a secure system only if the participants act responsibly
- Malicious participants are obviously removed from the system
  - But a careless one can make almost as much damage!

Knowing who is out there

- Knowing the participants is the first step
- Each Site should have
  a designated security contact
  - Interface to the rest of the Grid
- The OSG repository for such information is OIM

  https://oim.grid.iu.edu/oim/home
- Please make sure you keep your information updated there
  - You will need a user certificate to interact with it

Security communication

- Security contacts will receive security notifications through e-mail
  - Please read and act upon them
  - Make sure they have a legitimate signature
    https://twiki.grid.iu.edu/bin/view/ReleaseDocumentation/OSGSecurityNotifications
- Know and possibly be in contact with your Campus/Institution cyber security team
  - They can provide invaluable help both in preventing and fixing security incidents

Technical tasks

- Keep all the software up-to-date
  (mostly patching, but also upgrades as needed)
  - Operating system
  - System services
  - OSG/VDT provided software
- Keep security data up-to-date
  - List of trusted CAs
  - Associated CRLs
  - List of supported VOs
- Without, the risk of a compromise raises significantly

Advanced technical tasks

- If possible, actively look for signs of a compromise
- Log files can provide a lot of info
  https://twiki.grid.iu.edu/bin/view/ReleaseDocumentation/SearchLogFiles
- Yes, it can take a lot of time
  - But it pays big dividend, if you can afford it
  - A security incident can make a Site unusable for weeks (or worse)

What if you have a security incident?

- If you suspect a compromise,
  immediately notify the OSG security team
  https://twiki.grid.iu.edu/bin/view/ReleaseDocumentation/IncidentDiscoveryReporting
  - Even if it turns out that it was a false alarm,
    better safe than sorry (just don't do it every day!)
- Involving ALSO your local Campus/Institutional security team is a good idea
  - Especially if you are fairly sure you have a problem

Additional reading

- OSG Site Security Responsibilities
  https://twiki.grid.iu.edu/bin/view/ReleaseDocumentation/SecuritySiteResponsibilities
- OSG Security Hands On Training
  https://twiki.grid.iu.edu/bin/view/Security/SecurityHandsOnTraining
- Security Session at the 2009 OSG Admin Worksho
  http://indico.fnal.gov/sessionDisplay.py?sessionId=4&slotId=0&confId=2497#2009-08-06

Summary

- Security is both a social and technical problem
- Certificates are used for authentication, authorization is a separate step
- Not all the CAs are trusted,
  and you need to keep CRLs updated
- Keep your system software up-to-date
- Keep your contact information up-to-date in OIM
- Know how to report a security incident

Copyright notice

- These slides contain copyrighted images by ToonADay.com
- All such images have been licensed to
  Igor Sfiligoi
  for use in presentations
- Extracting such images and use them in other context is not permitted