Certificates in a Nutshell

Jens Jensen, STFC
Leader of EUDAT AAI TF

EUDAT

In a nutshell...

- Mature, Robust, Ubiquitous
  - Have been around for decades
  - Interoperable – supported by every OS, every language
  - Used everywhere – (e.g. e-commerce, banking)
- Very, very, secure
  - ... if done right!
- Two factor authentication
  - Something you have
  - Something you know

... don’t get distracted by the technical and implementation details

EUDAT

In a nutshell...

A CERTIFICATE IS AN AUTHORITATIVE TIMELY ASSERTION OF THE ASSOCIATION OF A PUBLIC KEY WITH AN IDENTITY

- Where the identity is a global name
  - and a description of what it is
  - or more precisely what it can do

EUDAT

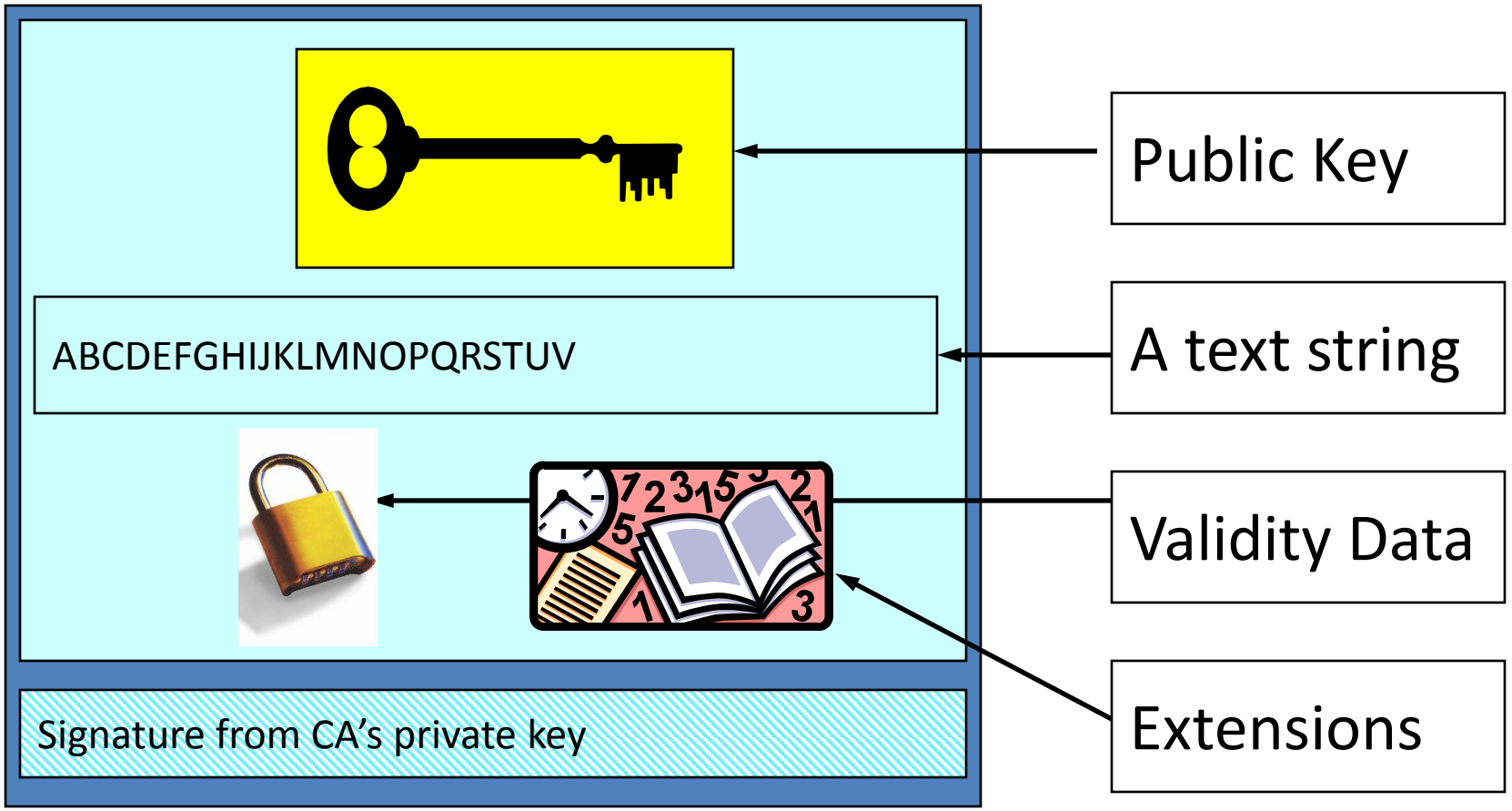
Public Key Cryptography

- Two halves of key:
  - Public – can be shared with everyone (usually)
  - Private – secret, must be protected
- Need to link *public key* to *identity*
  - In a PKI, this is the role of the CA (hierarchy)
  - ... done with the certificate
  - (Contrast PGP with its anarchy)
- Zero knowledge proof
  - Prove possession of the private key
  - Without revealing it (“NP

Asymmetric Cryptography

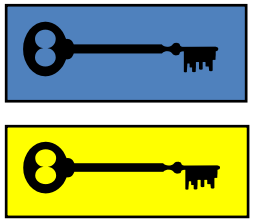
- If E is encrypt (encode with public key)
- And D is decrypt (encode with private key)
- Then E(D(x)) = x = D(E(x))
  - Except for pathological cases
- And E contains (almost) no information about D
- Depends on maths
- Slower than symmetric key encrypt/decrypt
  - Use E, D, to agree symmetric key

Anatomy of a certificate



X.509: original specification for certificates

How a Certificate is Issued

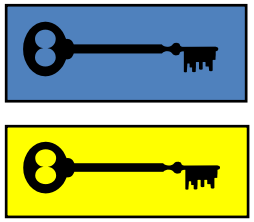


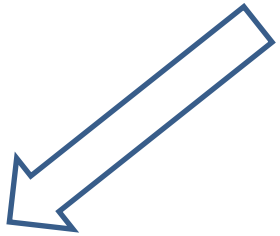
Create key pair (public, private)

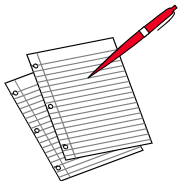
How a Certificate is Issued

Create CSR: certificate signing request

How a Certificate is Issued



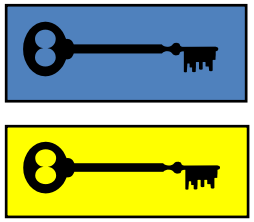


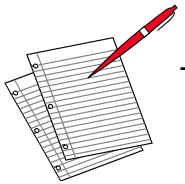


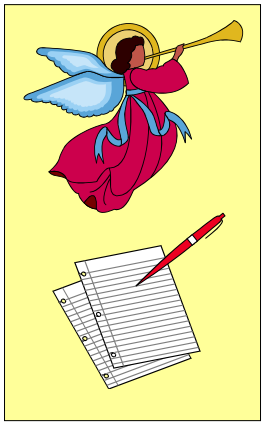
Submit request to a Certification Authority

EUDAT

How a Certificate is Issued



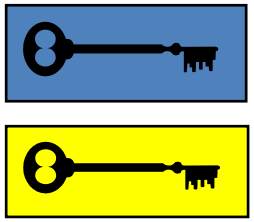


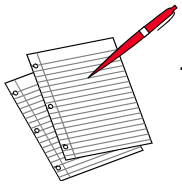


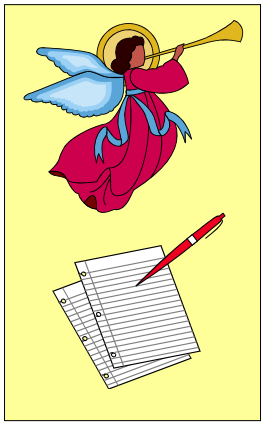
Persuade the CA to bless the certificate (via a Registration Authority)

EUDAT

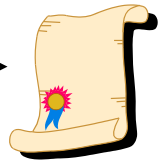
How a Certificate is Issued







The certificate is issued by the Authority



EUDAT

How a Certificate is Issued

- By personal contact with Registration Authority
- Or linking identity management system to CA
  - Shib → X.509 (e.g. UK)
  - Kerberos → X.509 (e.g. FNAL)
- Or certificates can be issued to a community
  - Sometimes...
  - Shared certificate, e.g. via a portal
  - Lower level of assurance (usually)
  - Restrict user actions via portal (= policy)

What Are Certificates Used For?

- Authentication
  - Identifying the entity at the end of a remote connection
  - Ensuring that it is the same entity every time
- Digital signatures (/electronic signatures)
  - Non-repudiation (maybe)
  - Code signing
- Timestamping services
- Encryption – short time, short messages
  - E.g. signed email (S/MIME)
- “Robots” (Grid) – automated agents acting for user

EUDAT

Timeliness of Information

- Re-check at renewal/rekey
  - End of certificate lifetime
- Revoke if “compromised”
  - Certificate revocation lists – take a while to get distributed (~hours)
  - OCSP – slowly increasing use
- Circumstances for revocation
  - Compromise of private key – urgent!
  - Certificate no longer needed
  - Information no longer correct

Timeliness of Information

- CA – long lived certificates
  - 3-20 years
  - Are they secure on this timescale?
- End entity – Long lived certificates (1-3 years)
  - Identity doesn't change (often)
- Separate authentication and authorisation
  - Authorisation is short lived...
- End entity – SLCS
  - Conventionally (=grid) up to 1 Ms
  - *Could* contain authorisation information (cf proxies)

EUDAT

Current Issues

- Signatures:
  - MD5 based signatures vulnerable/broken
  - SHA1 based signatures increasingly vulnerable
  - SHA2 secure but not widely supported
- Naming
  - UTF-8 in common names (from printableString)
  - Using string representations of names
- “Good enough” software

“Delegating” Certificates

- Getting credentials to some remote entity...
- Globus approach (GSI, RFC 3820)
  - “Proxy certificates”
  - MyProxy
- gLite (EMI) delegation API
- mod_gridsite
- Contrail: use OAuth2

- OGF working group: IDEL-WG

EUDAT

Delegating with GSI proxies

... greatly simplified

- Private key never crosses the network
- VOMS uses attribute certificates (RFC 3281)
- User certs are not allowed to sign other certs
  - "Hack" for Globus, using name restr.
- How to interpret multiple VOMS extn's
  - OGF VOMSPROC-WG
- VOMS can also embed SAML

EUDAT

Anatomy of a CA

- A CA certificate
  - Which signs users' (or rather end entities) certificates
- A certificate policy
- A certification practices statement
- Infrastructure
  - Archive
  - User Support
  - Handling notifications
- Audits, compliance, certification

EUDAT

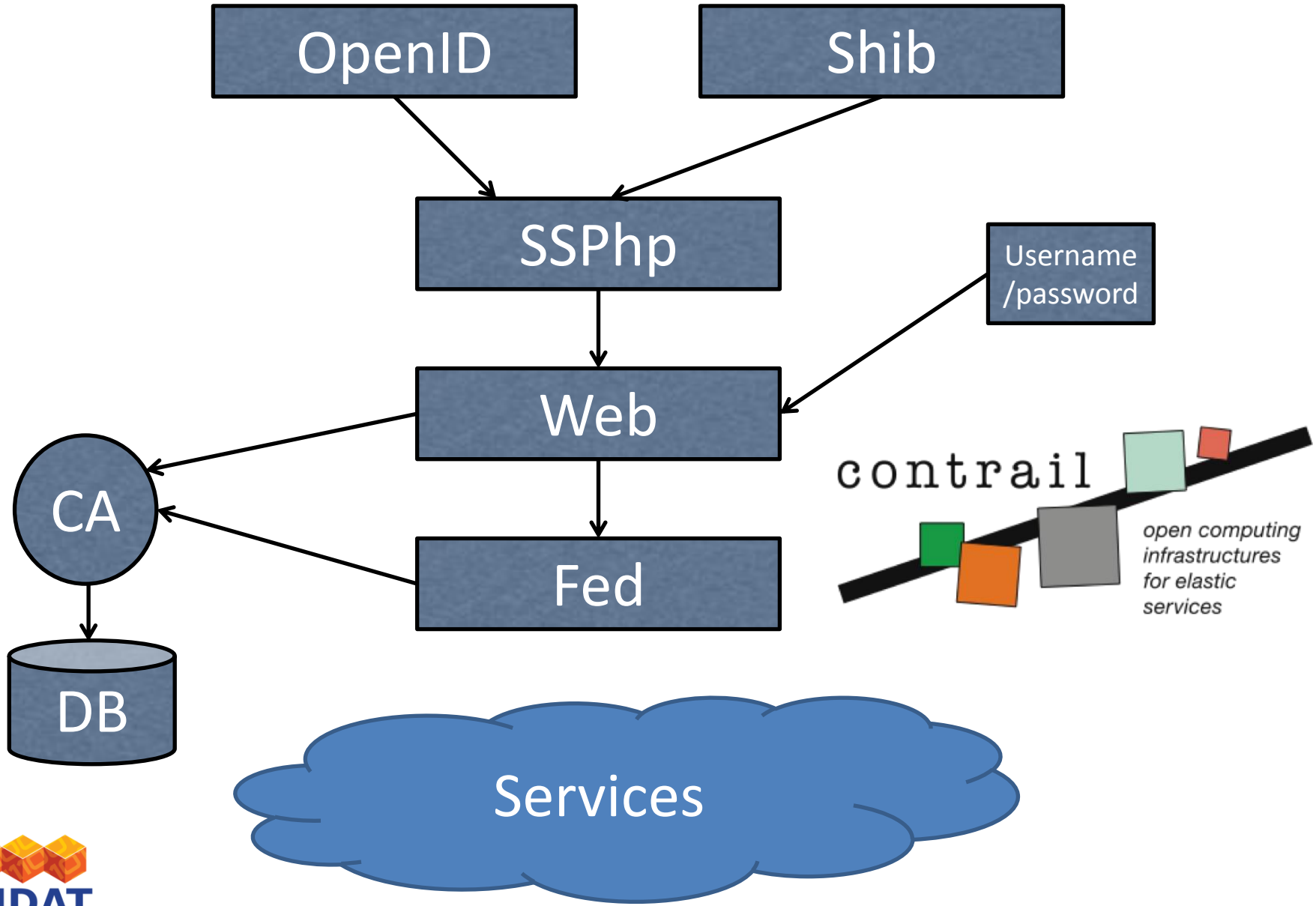
Community View

- All need to trust the CA
  - (or rather each other's CAs)
- Occasional rekeys and rollover
  - New certificates, same names
  - Invalidates old signatures
  - Only names are persistent
- Naming: X.500
  - And domainComponents (RFC 2247)
  - /DC=eu/DC=eudat/DC=federation/CN=jens jensen

EUDAT

CAs for EUDAT

- Infrastructure CAs:
  - Reuse the ones for grids
  - IGTF (www.igtf.net), NRENs (www.terena.org)
- Browser-facing certificates
  - Comodo via Terena
- Personal certificates
  - EUDAT-internal CA
  - Hide from users
  - Use external identity assertions and attributes


OpenID
Shib
SSPhp
Username
/password
Web
CA
Fed
DB
Services
contrail
open computing
infrastructures
for elastic
services
EUDAT

Certificates in EUDAT

- Demonstrators in Stockholm
  - Reusing Contrail portal
  - Challenge: integrate with user portals
- Need to distribute CA certificates
  - Trusted repositories (compare web browser)

EUDAT

Further Experimenting

- Get your browser to save a remote certificate
  - Or inspect -
- Experimenting with certificates – OpenSSL

  openssl x509 –text –noout –in cert.pem

  openssl ca –in req.pem –out cert.pem

  openssl asn1parse –i –in cert.pem

EUDAT

Further Reading

- Certificate profile
  - RFC 5280
  - GFD.125 (soon to be updated)
- Want to know more about CAs?
  - RFC 3647
- Delegation...

In a nutshell...

- Mature, Robust, Ubiquitous
  - Have been around for decades
  - Interoperable – supported by every OS, every language
  - Used everywhere – (e.g. e-commerce, banking)
- Very, very, secure
  - ... if done right!
- Two factor authentication
  - Something you have
  - Something you know

Questions, Comments, ...

→ jens.jensen<>stfc.ac.uk

EUDAT