Distributed AA in Data Management

Fundamentals of Data Management

Includes material from
Certificates in a Nutshell, Jens Jensen, EUDAT
Federated Identity Management, Willem Elbers, EUDAT

http://www.eudat.eu/training-materials-downloads

Dr Rob Baxter
Software Development Group Manager, EPCC
r.baxter@epcc.ed.ac.uk
+44 131 651 3579 | +44 7971 437749

Course outline

- What do we mean by “AA”?
  - Authentication and authorisation
- How do networked data services control access to their resources?

- After completing this lesson, you should be able to:
  - Explain the difference between authentication and authorisation
  - Explain the concept of digital certificates and how they work
  - Explain the concept of federated identity management
  - Appreciate that this is still an area of active development!

Not just a data problem

- Distributed authentication and authorisation is not, of course, a data management problem
- But the distributed nature of today's data infrastructures make it very visible
  - data resources, data services are increasingly networked
  - data repositories have Web-facing front ends
- This lecture is *not* a complete lesson in distributed AA
  - (not even close!)
- But it should give you a basic understanding of the concepts and jargon involved

AA

- First things first: when requesting access to some computational or data service or resource…
- Authentication (“AuthN”):
  - is you proving you are who you claim to be (proof of identity)
  - “Who are you?” – “I’m John, here’s my passport” – “OK, you’re John”
- Authorisation (“AuthZ”):
  - is the service or resource checking that you are allowed to do what you’re asking to do
  - “Is John allowed to access this dataset?” – “Yes”
- We’ll look mostly at AuthN, in two common flavours
  - X.509 certificates
  - Federated identity management

X.509* Certificates

- 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

* X.509 comes from ‘part 9 of the ITU X.500 Directory standard’

In a nutshell

A Certificate is an Authoritative, Timely Assertion of the Association of a Public Key with an Identity

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

Public Key…?

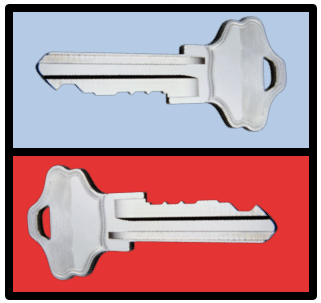
- Public key cryptography – the basis of X.509 certificates
- A cryptographic *key* is a string of letters/numbers which provides input into an encryption algorithm
- In public key cryptography infrastructures (*PKIs*) a user has two halves of a key:
  - public – can be shared with anyone and everyone
  - private – secret, must be protected
- For AuthN, need to link public key to identity
  - In a PKI, this is the role of the CA (hierarchy)
  - ... done with the certificate
- Allows for zero knowledge proof
  - prove possession of the private key
  - without revealing it

Asymmetric cryptography

- If E() is the encrypt function (encode with *public* key)
- And D() is the decrypt function (decode with *private* key)
- Then E(D(x)) = x = D(E(x))
- And E() contains (almost) no information about D()
- Depends on maths
- Slower than symmetric key encrypt/decrypt
  - (where public key = private key)
  - but can use E() and D() at the start of a secure conversation to agree a one-time symmetric key (cf. SSL)

Anatomy of a certificate

How a certificate is issued



Create a key pair (public, private)

CSR

Create a certificate signing request (CSR)

Submit CSR to Certification Authority (CA)

CA

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

Cert

The certificate is issued by the CA

How a certificate is issued

- By personal contact with a Registration Authority
  - As in: go to Jane's office and show her your identity card
- Or by linking identity management system to CA
  - Shibboleth to X.509 (e.g. UK)
  - Kerberos to X.509 (e.g. FNAL)
- Or certificates can (sometimes) be issued to a community
  - Shared certificate, e.g. via a portal
  - Lower level of assurance (usually)
  - Restrict user actions via portal (= policy)
- Certificates are usually chained, with a Root CA at the end, eg.
  - radius01.is.ed.ac.uk (local CA), signed by
  - TERENA (intermediate CA), signed by
  - The USERTRUST Network (root CA)

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 (aka electronic signatures)
- Time-stamping services
- Encryption
  - short time, short messages
  - e.g. signed email (S/MIME)
- “Proxies” or “Robots” (usually in the context of Grid systems)
  - Automated agents acting for or on behalf of a user
  - Hence: “X.509 Proxy Certificates” – an extension to X.509

Timeliness of information

- Certificates do expire – they are valid for a set period of time
- They can also be revoked if “compromised”
  - Certificate revocation lists – take a while to get distributed (~hours)
- Circumstances for revocation
  - Compromise of private key – this is the urgent one!
  - Certificate no longer needed
  - Information no longer correct
- Certificate Authorities have long-lived certificates – years

Federated identity management

- How do you ensure “proof of identity” is portable across different systems, service domains, infrastructures?
  - How do you achieve single sign-on (SSO)?
- There is no single, common authentication system
  - username/password (eg. “basic http authn”)
  - certificate
  - Shibboleth
  - OAuth
  - simpleSAMLphp
  - OpenID
  - …
- Federated ID management is about making them work together

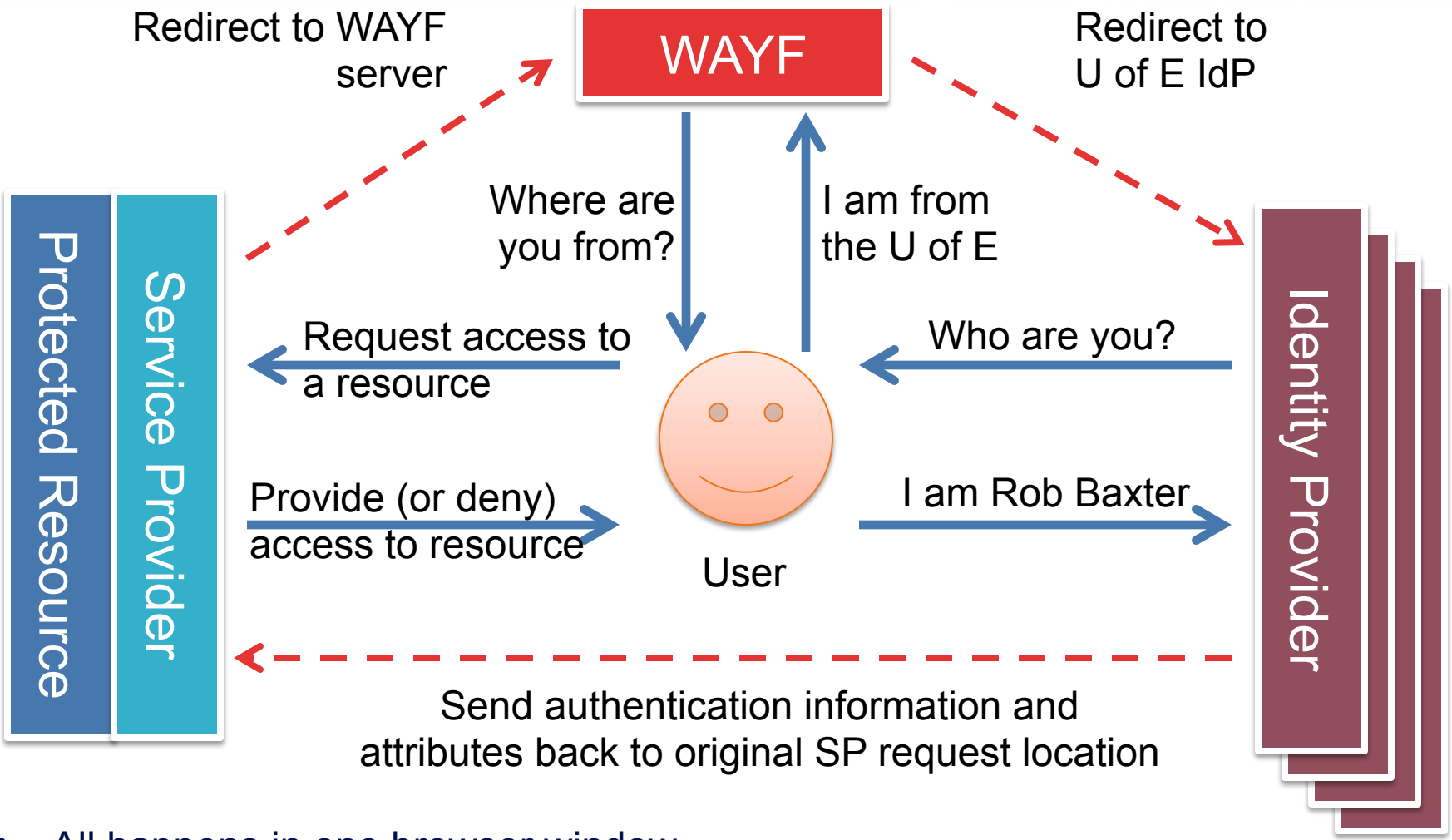
A few concepts

- IdP – Identity Provider
  - A service running at you home institution (eg. University of Edinburgh)
  - Authenticates a user (using the institution's user database)
  - Will release some information about the user (*attributes*)
- SP – Service Provider
  - A service running at the organisation you want to access
  - Makes authorisation decisions for a user depending on their attributes
  - Redirects user to a login endpoint if no identity information is available
    - Can be an individual IDP
    - Can be a "where are you from" page
  - Decides which identity providers to accept
  - Decides which attributes to accept

A few concepts

- WAYF – Where Are You From?
  - A typical FIM use-case is an SP accepting users authenticated by a (large) number of IdPs
  - Q: how does the SP decide which login endpoint the user needs to be redirected to?
  - A: a Where Are You From page
    - Let the user select the IdP he/she wants to authenticate with

Use of WAYF and IdP in login



- All happens in one browser window
- Redirects are behind the scenes to the user

Identity Federation

- All the preceding relies on *trust* between IdPs and SPs
- An Identity Federation (IdF) is a group of IdPs and SPs that trust one another
- Within the IdF users can use a single identity to access all services and resources
  - subject to their being authorised, of course!
- Building IdFs is about IdPs and SPs signing contracts
- And then there are (emerging) “meta-IdFs” or “interfederations”
  - eduGAIN in the EU

Authorisation

- AuthZ decisions need to be taken by the SP that looks after the resource being requested
  - they are “local” decisions
- A common AuthZ framework needs a common user identity
  - I can access this here, and I can access that there
- However, IdPs don’t all provide the same information (*attributes*) about the same user
  - IdP-1 might tell an SP: “Yes this is rob_baxter”
  - IdP-2 might tell an SP: “Yes this is user2348”
  - and they’re both me!

Summary

- Distributed data infrastructures need distributed identity and authentication management
  - as do plenty of other things on the network
- Certificates are a common AuthN mechanism
  - widely used in grid and Internet computing
- Identity federations allow for single sign-on across a group of related but independently-managed services
  - data repositories, for instance
- Mapping an authenticated user onto a set of authorisation permissions is still tricky
  - we still don't have globally accepted user ids!

Acknowledgements

- Slides from EUDAT / Jens Jensen
  - *Certificates in a Nutshell*
  - http://eudat.eu/system/files/JensJensen.pdf
- Slides from EUDAT / Willem Elbers
  - *Federated Identity Management*
  - http://eudat.eu/system/files/WillemElbers2.pdf