Managing CA-Signed Certificates

T.Rob Wyatt, IoPT Consulting

t.rob@ioptconsulting.com

Managing CA Certificates for MQ - Intermediate

So you want to enable SSL on your MQ channels using a commercial Certificate Authority? Unfortunately, very few CAs provide documentation relevant to MQ. This session explains the certificate features and options offered by commercial CAs, how these interact with MQ, and which to choose. The session ends with a step-by-step implementation and diagnostic process that will slash your deployment effort.

Agenda

- A word about…
- Anatomy of a certificate
- How IBM MQ uses certificates
- Speaking of certificates…
- Certificate attributes
- Types of certificate
- Ca/Browser who?
- What kind of cert for MQ?
- Walking through the process
- Setup & Debugging

A word about security

- These slides reflect the information available at the time of publication.
- With security, things change. Often quickly. Periodically check that you have the most current version of any reference materials.
- The latest version of this presentation will be posted at https://t-rob.net/links
- To be notified of updates via RSS or by email, subscribe at the web site

A word about IoPT

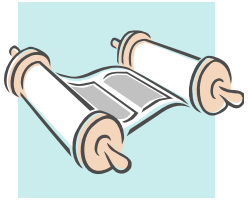
T.Rob left IBM to form IoPT Consulting in 2013. The firm offers…

MQ Security Guy

- **The same MQ consulting services**
  - Security (of course!)
  - Architecture
  - High Availability
  - Upgrades
- **Conventional + retainer engagements**
  - HA/DR
  - Troubleshooting
  - Staff Augmentation
  - Much more

704-443-TROB (8762) https://ioptconsulting.com

Anatomy of a certificate



An identity & policies | Bound to | Public Key | = | Public certificate

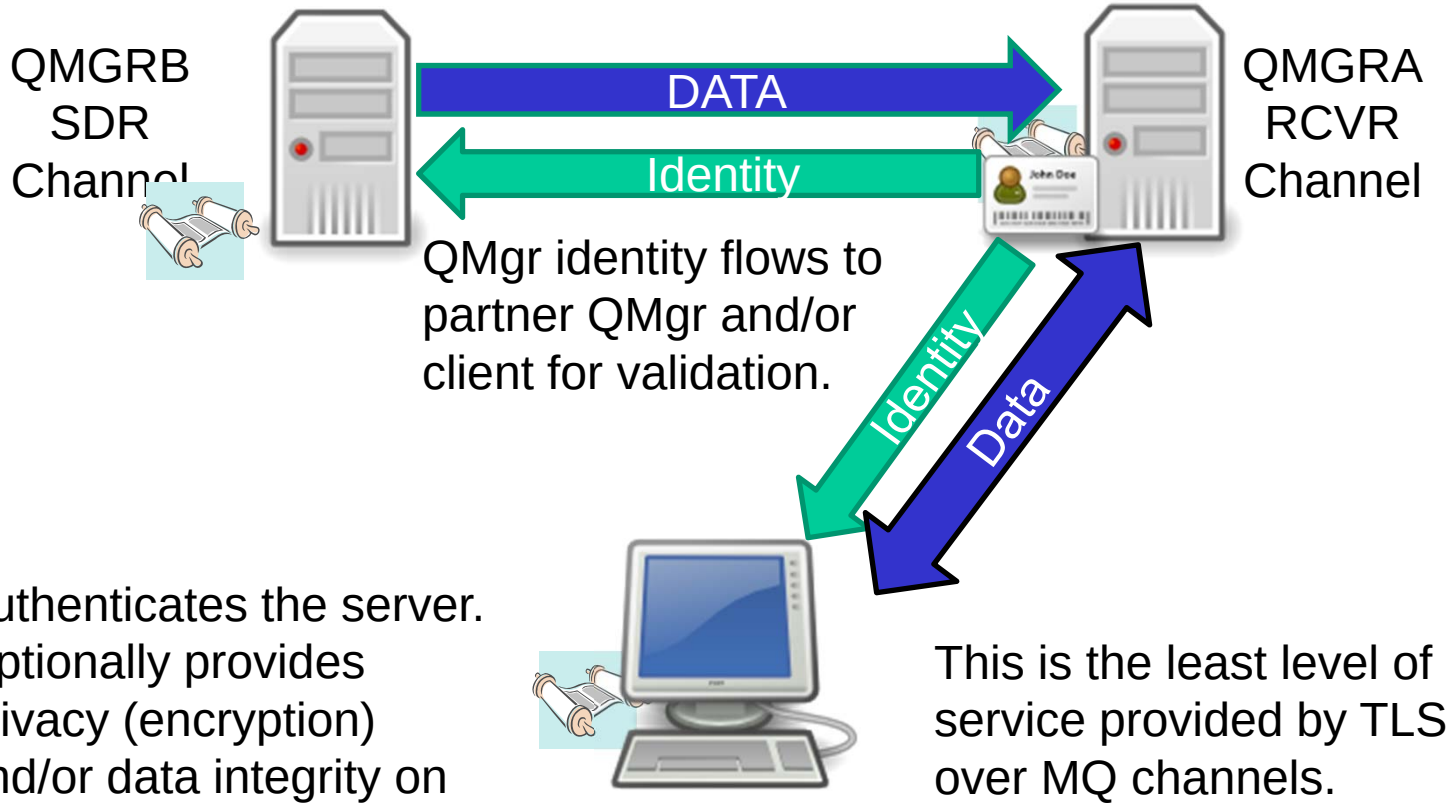
Your Public Certificate is what *you present to others* to authenticate yourself.
The CA's Public Certificates are used to authenticate the signature of the certificates *presented to you*.

Anatomy of a certificate

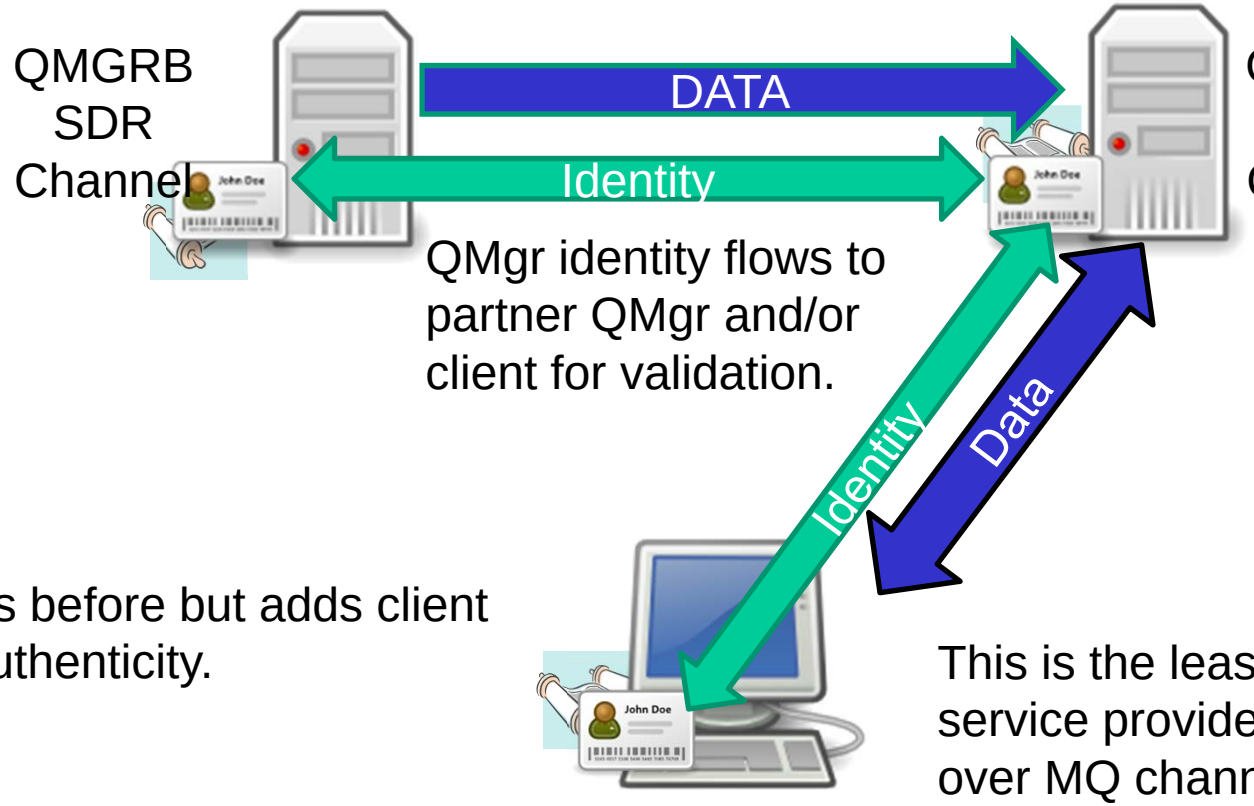
An identity & policies | Bound to | Public + Private key | = | Personal certificate

Personal cert is used to [en|de]crypt and establish secret keys.

How IBM MQ uses certificates: Server authentication



How IBM MQ uses certificates: Mutual authentication



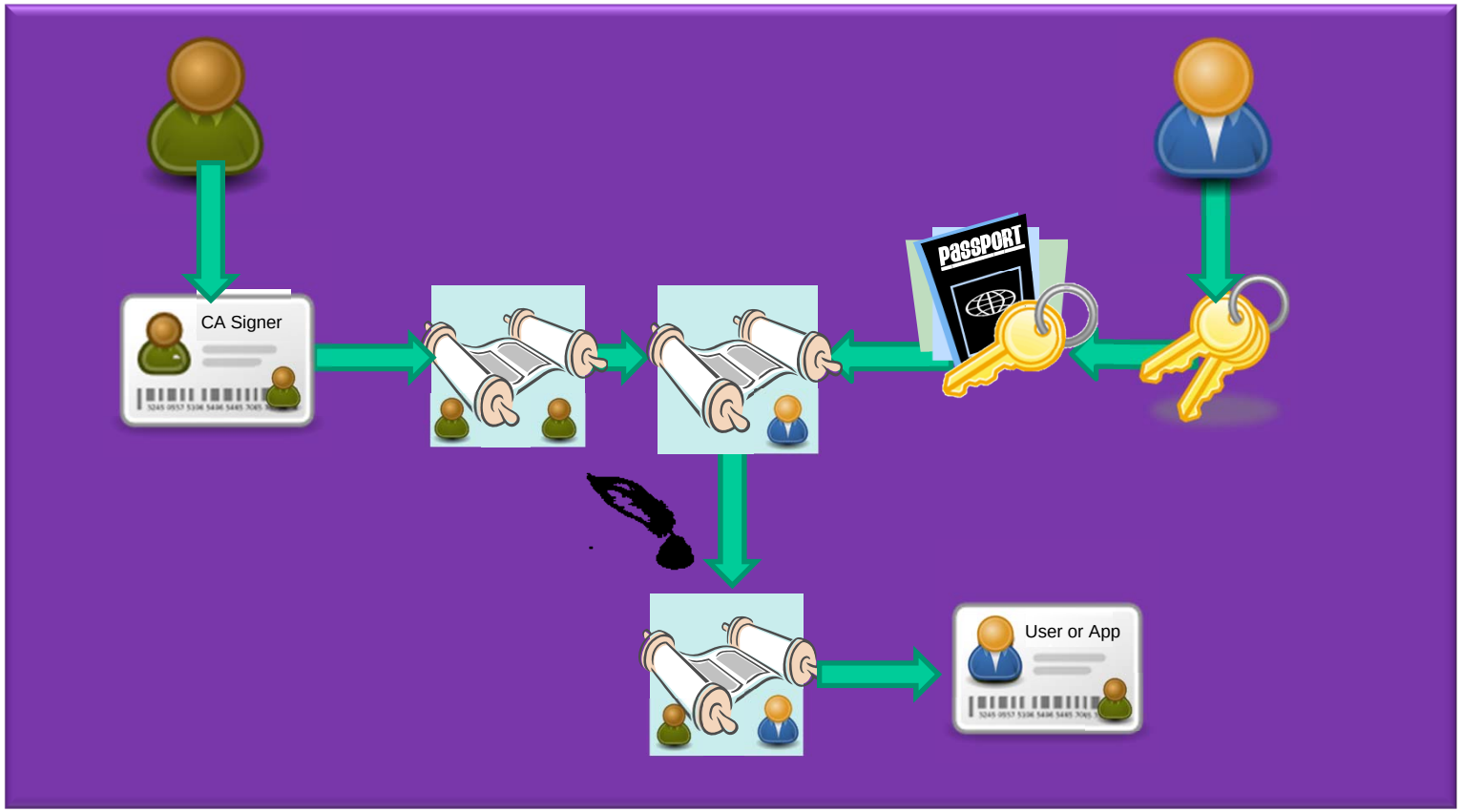
Speaking of certificates…

A self-signed certificate is one where the key *in* the certificate is the same one used to *sign* the certificate.

Speaking of certificates…

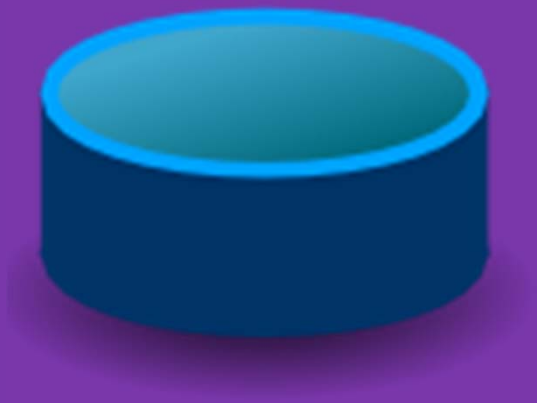
A Root certificate is a self-signed certificate enabled for signing other certificates.

Speaking of certificates…
Signing Request



Speaking of certificates…
Trust & Validation

Speaking of certificates…
Keystore/trust store



A *keystore* is a place to keep keys and certificates.

A trust store is a keystore that contains only public certificates – the trusted entities.

MQ uses .kdb files .jks (for Java) keystores.

Certificate attributes

- **Distinguished Name**
- **Validity period**
- **Key, key length, key type**
- **Serial number**
- **Fingerprint**
- **Signature algorithm and details**
- **Revocation responder URI**
- **Policies**
  - Can it sign other certs? Can it sign signer certs?
  - Can it encipher communications?

Certificate attributes

Two sets of attributes in the certificate:

- **Subject – owner of the certificate**
- **Issuer – signer of the certificate**
  - For CA-signed certificates, the issuer is the certificate used to sign the one being examined
  - The Issuer info is present on root and other self-signed certs, it matches the Subject info

Types of certificate

Three types of certificate:

- **Domain Validated (DV) Certificate**
- **Organizational Validated (OV) Certificate**
- **Extended Validation (EV) Certificate**

Granularity of the certificate:

- **Standard certificate**
- **Wildcard Certificate**
- **Unified Communications (UC) Certificate**

Types of certificate

- **Standard certificate**
  - ioptconsulting.com, www.ioptconsulting.com
- **Wildcard**
  - Includes all subdomains
  - E.g. *.ioptconsulting.com
- **Unified Communications Certificate (UCC)**
  - Up to 99 domains in Subject Alternative Name
  - Very common for cache servers and CDNs

Types of certificate
DV vs. OV VS. EV Certs

- DV Certificate – verify only that the owner of the domain approves the request.
- OV Certificate – public DB search to validate company and domain name
- EV Certificate – Closer to an actual background check

Types of certificate
DV vs. OV vs. EV Certs

Level of service by cert type:

Type of certificate	Domain validated?	Subject Name Validated?	Address Validated?	Pad Lock Displayed by Browser?	Green address bar or other special treatment?	Relative price
DV	X			X		$
OV	X	X	X	X		$$
EV	X	X	X	X	X	$$$

Source: https://cabforum.org/info-for-consumers/

Types of certificate
DV vs. OV vs. EV Certs

Because the DV and OV certificates have minimal validation, attributes other than the Common Name and the Subject Alternative Name are deleted if requested!

Those useful OU fields? Fugeddaboudit!

Types of certificate
DV vs. OV vs. EV Certs

Your requested OU fields might even be wiped out on an EV cert!

Because the CAB Forum believes their only customers are web servers and browsers.

Those useful OU fields? Ask before dropping big money on an extended validation cert!

Ca/Browser who?

The CA/Browser Forum is the consortium that sets the standards and policies all member CAs and browser vendors follow.

They don’t recognize the need for certificates for any entity more granular than a fully qualified domain name or email address, which is unfortunate for those of us administering anything else that needs to be authenticated or encrypted. Like a QMgr on the internal network.

https://cabforum.org/

What kind of cert for MQ?

- **Any standard web server cert will work.**
- **EV useful *if* the CA lets you specify OUs**
- **For external connections, EV *may* be helpful**
- **Wildcard certificates not helpful at all**
  - Because MQ doesn't do the same DN and SAN that browsers perform
- **UC Certs *may* be helpful**
  - Stash OU info into SAN – i.e prod.qmgr.dom.tld
  - Extra expense

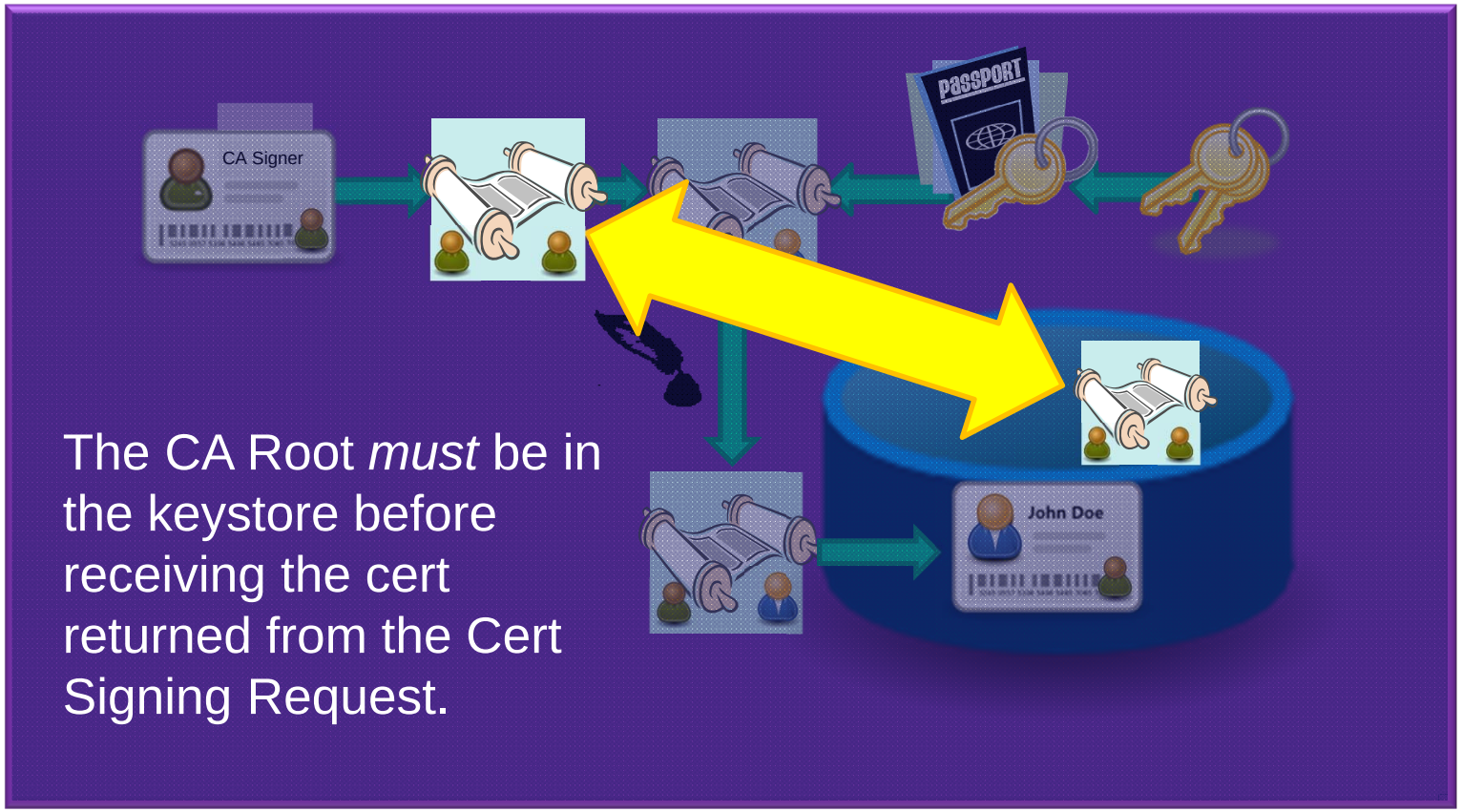
Walking through the process

1. Define the KDB or JKS
2. Generate a Cert Signing Request (CSR)
3. Submit the CSR to the CA
4. Download the signed cert & signer bundle
5. Receive the signed cert into the KDB or JKS
6. REFRESH SECURITY TYPE(SSL) or bounce the application to pick up the new keystore
7. Configure the channels

Walking through the process

No special requirement to generate a key pair and signing request.
Can do this with an empty JKS or KDB.

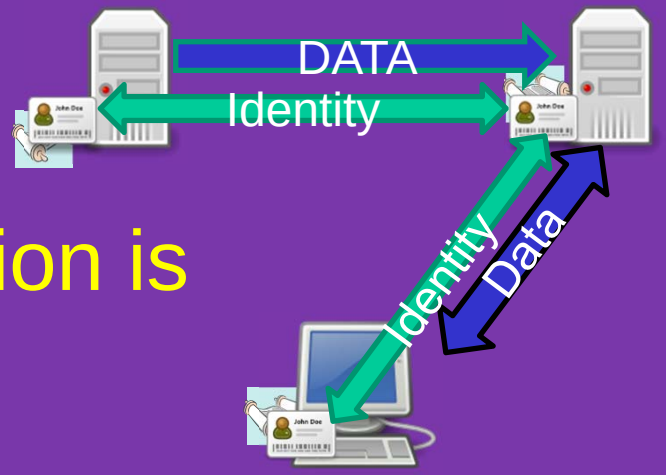
Walking through the process



Setup & Debugging

Mutual authentication is where you end up.

It is NOT where you begin.



Setup & Debugging

Make life easier…

1. **Install MS0P if using MQ Explorer**
2. **Enable events for SSL, channel, and auths**
3. **Get the channel running without TLS**
4. **Enable server authenticated TLS**
5. **Enable mutually authenticated TLS**
6. **Enable CHLAUTH rules and/or exits**

Many people start at #5 and forget #6.

Don’t be that person!

Questions & Answers

Two possibilities here:

1. **You are asleep**
2. **Something needs clarification**

Because even if you are *extremely* good at understanding this stuff, I can usually do a better job explaining it.

Please see my video series on this topic at: http://iopt.us/manageCAcerts

Questions & Answers