How To Build And Operate A Certificate Authority Center of Mediocrity (CACOM)

Certified for IBM WebSphere. software

Certified for IBM Power Systems

T.Rob Wyatt
Managing Partner, IoPT Consulting
704-443-TROB (8762)
t.rob@ioptconsulting.com
https://ioptconsulting.com

Change is the only constant

This presentation reflects…

- My current opinions regarding WMQ security
- The product itself continues to evolve (even in PTFs)
- Attacks only get better with time
- This version of the presentation is based on WebSphere MQ v7.1 & v7.5
- This content will be revised over time so please be sure to check for the latest version at https://t-rob.net/links
- Your thoughts and ideas are welcome

Agenda

- What is a Certificate Authority?
- The role of a CA in the security model
- Why run your own CA?
- Some example CA Best Practices
- CA Second-Best (and lesser) Practices
- The CA Maturity Model
- Wrapping up

What is a Certificate Authority?

- A disinterested 3rd party
- A stable, long-lived entity
- A pool of specialized and deep security skills
- An implementation of strict physical controls
- An implementation of strict human processes
- An investigatory service
- A revocation service
- Part of a consortium that provides a consistent set of well-defined services

What is a Certificate Authority?

Oh yeah…

They sign certificates once in a while.

Agenda

- What is a Certificate Authority?
- The role of a CA in the security model
- Why run your own CA?
- Some example CA Best Practices
- CA Second-Best (and lesser) Practices
- The CA Maturity Model
- Wrapping up

Role of the CA

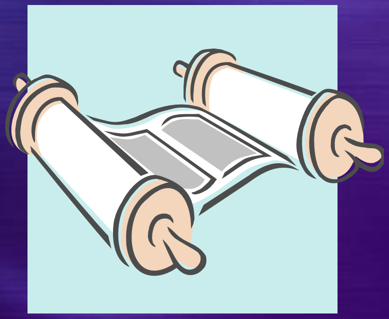
First need to understand the certificate.

An identity

Bound to

A key Pair

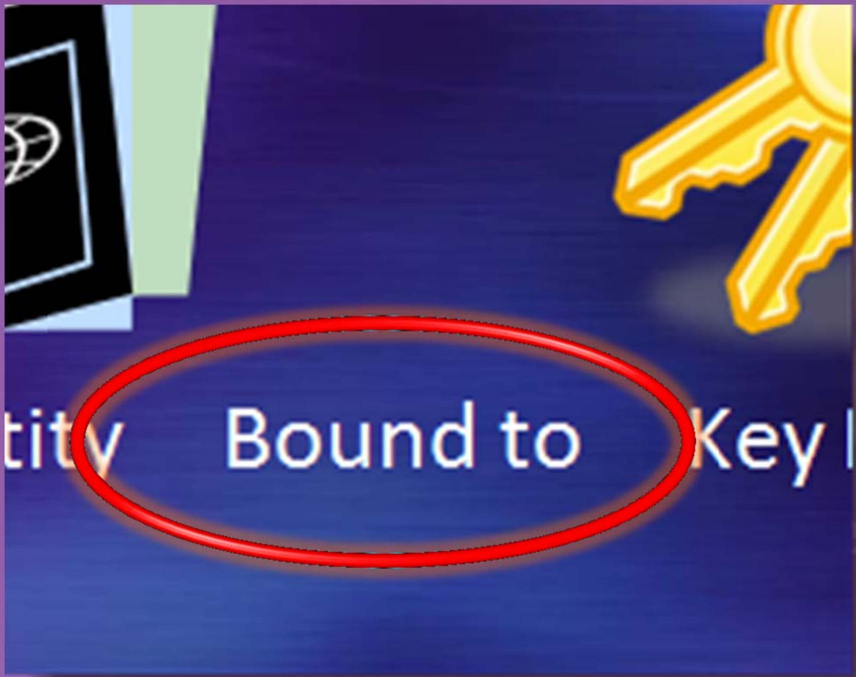
=



A certificate

Role of the CA

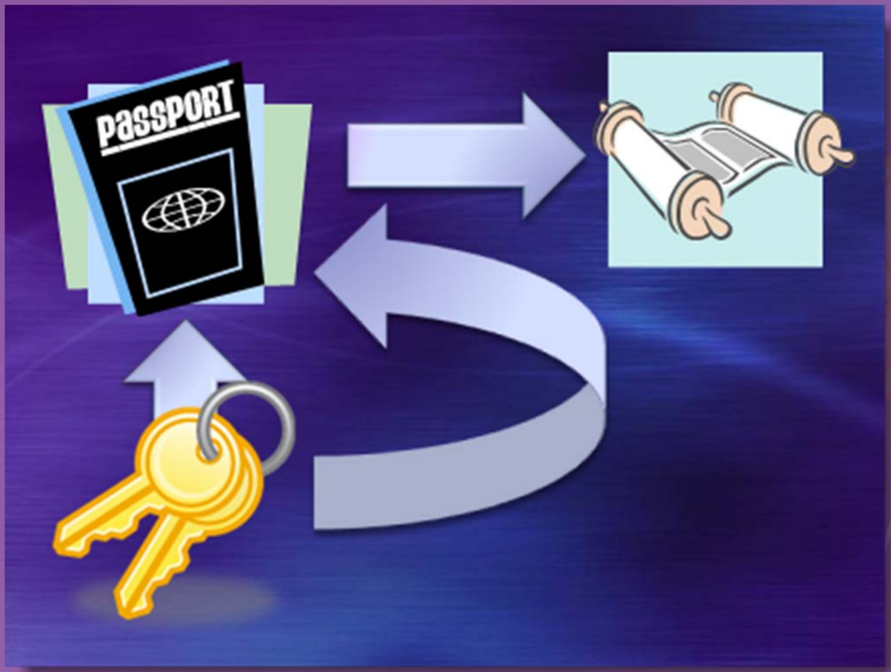
First need to understand the certificate.



The identity and key are bound using a cryptographic signature.

Role of the CA

First need to understand the certificate.



Self-signed means that the key *in* the certificate is the same one used to *sign* the certificate.

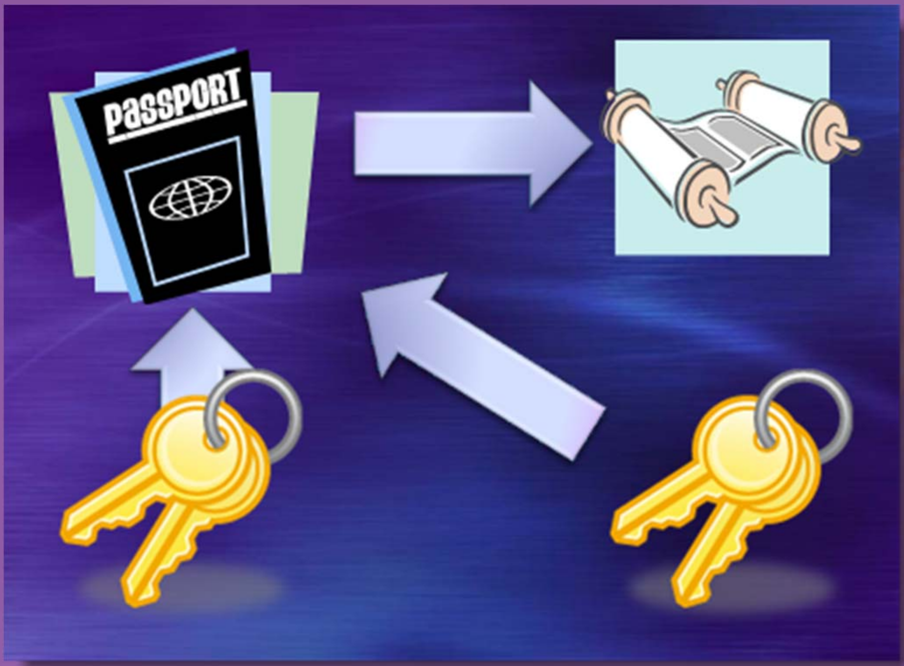
Role of the CA

First need to understand the certificate.

Special Note: Self-signed does *NOT* mean “signed with our internal CA.” Self-signed has a specific technical meaning describing the relationship between the identity and the key used to bind it.

Role of the CA

First need to understand the certificate.



When the certificate is signed by an external key, it is said to be CA-signed.

Role of the CA

- Signs the certificate.
- Validates the claimed identity.
- Enforce policy on the Distinguished Name fields.
- Ensures unique names within its namespace.
- Provides revocation services.
- Maintains key lifecycle for root, intermediate and signed certificates.
- Disinterested 3rd party provides bilateral trust.

Role of the CA

Relying parties typically need much less security than the CA.

- Generate the cert signing requests securely.
- Review and approve requests at the CA.
- Keep the keystore files private.

Most of the heavy lifting is offloaded to the CA.

Agenda

- What is a Certificate Authority?
- The role of a CA in the security model
- Why run your own CA?
- Some example CA Best Practices
- CA Second-Best (and lesser) Practices
- The CA Maturity Model
- Wrapping up

Why run your own CA?

- Cost savings.
- Convenience.

The trick is to fully account for all of the service and security provided by the CA.

Replacing those services internally is expensive. Omitting them is dangerous.

How do you find the balance?

Agenda

- What is a Certificate Authority?
- The role of a CA in the security model
- Why run your own CA?
- Some example CA Best Practices
- CA Second-Best (and lesser) Practices
- The CA Maturity Model
- Wrapping up

Example CA Best Practices

- SAS70 key ceremony used to generate roots.
- Root certs stored in an offline HSM.
- Dual-knowledge access controls for roots.
- Multiple intermediate signers.
- Secure, robust provisioning process for certs and revocation entries.
- Highly available and separate provisioning and revocation infrastructure.

Example CA Best Practices

- ISO 21188:2006 Public key infrastructure for financial services – Practices and policy framework
- ETSI TS 101 456 v1.2.1
- ETSI TS 102 042 V1.1.1
- Webtrust Program For Certification Authorities

Agenda

- What is a Certificate Authority?
- The role of a CA in the security model
- Why run your own CA?
- Some example CA Best Practices
- CA Second-Best (and lesser) Practices
- The CA Maturity Model
- Wrapping up

Second-Best (and lesser) Practices

General guiding principles of a CACOM

- To run a true Center of Mediocrity, one must first determine which CA Best Practices to omit.
- The quickest way to begin is to omit all of them right off the bat.
- Then add back in those that give the *appearance* of security at little or no cost.
- The CACOM is your BFF. Once you have it, you'll never get the money to build a CACOE. Why? Because what we have seems to work just fine.

Second-Best (and lesser) Practices

The stories you are about to hear are true.

Only the names have been changed
to protect the (not so) innocent.

My name is T.Rob. I carry a laptop.

Second-Best (and lesser) Practices

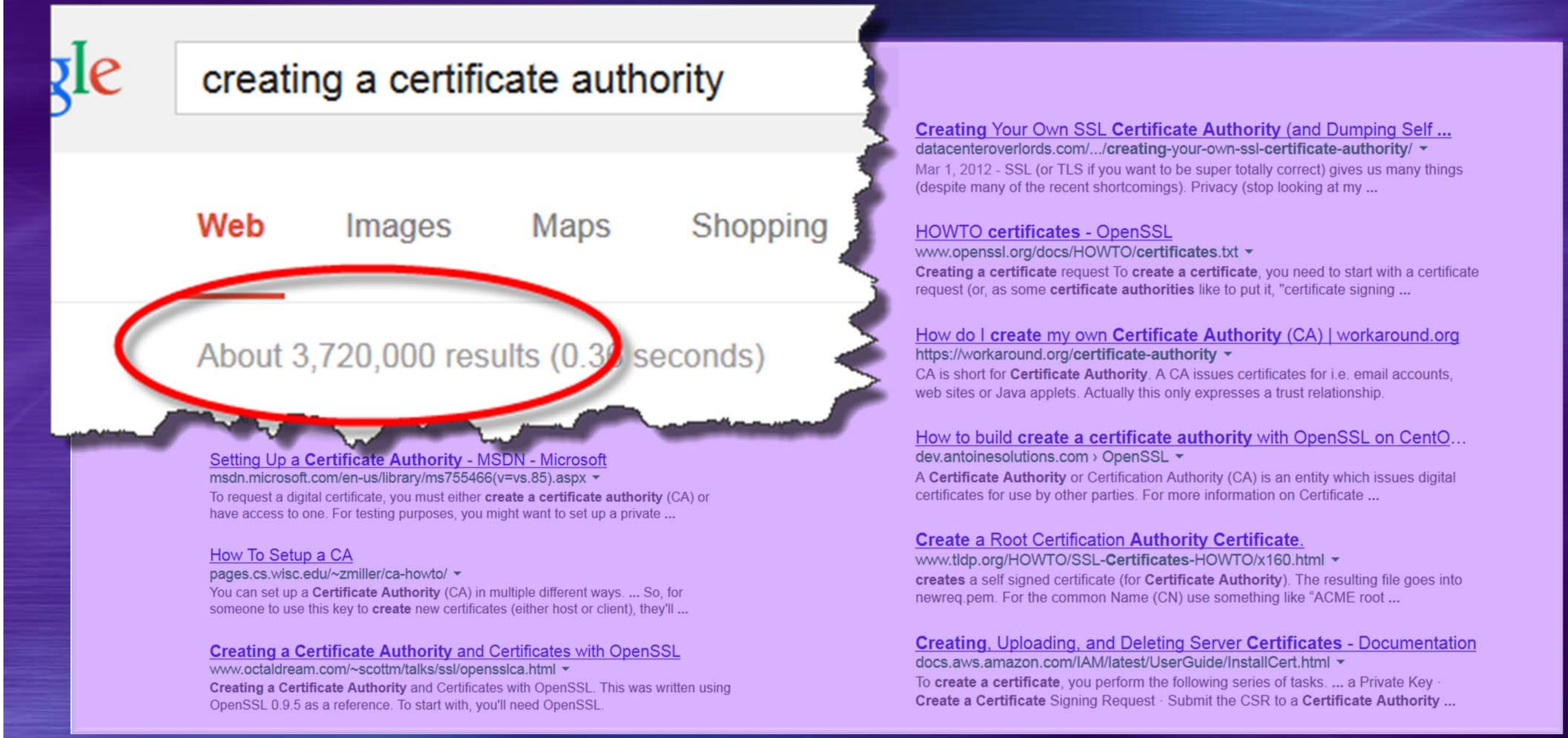
All of the items in the following pages were observed in Production somewhere.

Banking, finance, insurance, healthcare, govt.

These are your vendors!

Second-Best (and lesser) Practices

Best place to start? Google!



creating a certificate authority

About 3,720,000 results (0.36 seconds)

[Setting Up a **Certificate Authority** - MSDN - Microsoft]
msdn.microsoft.com/en-us/library/ms755466(v=vs.85).aspx
To request a digital certificate, you must either **create a certificate authority** (CA) or have access to one. For testing purposes, you might want to set up a private ...

How To Setup a CA
pages.cs.wisc.edu/~zmiller/ca-howto/
You can set up a **Certificate Authority** (CA) in multiple different ways. ... So, for someone to use this key to **create** new certificates (either host or client), they'll ...

Creating a Certificate Authority and Certificates with OpenSSL
www.octaldream.com/~scottm/talks/ssl/opensslca.html
Creating a Certificate Authority and Certificates with OpenSSL. This was written using OpenSSL 0.9.5 as a reference. To start with, you'll need OpenSSL.

Creating Your Own SSL **Certificate Authority** (and Dumping Self ...
datacenteroverlords.com/.../**creating**-your-own-ssl-**certificate-authority**/
Mar 1, 2012 - SSL (or TLS if you want to be super totally correct) gives us many things (despite many of the recent shortcomings). Privacy (stop looking at my ...

HOWTO **certificates** - OpenSSL
www.openssl.org/docs/HOWTO/**certificates**.txt
Creating a certificate request To **create a certificate**, you need to start with a certificate request (or, as some **certificate authorities** like to put it, "certificate signing ...

How do I **create** my own **Certificate Authority** (CA) | workaround.org
https://workaround.org/**certificate-authority**
CA is short for **Certificate Authority**. A CA issues certificates for i.e. email accounts, web sites or Java applets. Actually this only expresses a trust relationship.

How to build **create a certificate authority** with OpenSSL on CentO...
dev.antoinesolutions.com › OpenSSL
A **Certificate Authority** or Certification Authority (CA) is an entity which issues digital certificates for use by other parties. For more information on Certificate ...

Create a Root Certification **Authority Certificate**.
www.tldp.org/HOWTO/SSL-**Certificates**-HOWTO/x160.html
creates a self signed certificate (for **Certificate Authority**). The resulting file goes into newreq.pem. For the common Name (CN) use something like "ACME root ...

Creating, Uploading, and Deleting Server **Certificates** - Documentation
docs.aws.amazon.com/IAM/latest/UserGuide/InstallCert.html
To **create a certificate**, you perform the following series of tasks. ... a Private Key · **Create a Certificate** Signing Request · Submit the CSR to a **Certificate Authority** ...

Second-Best (and lesser) Practices

Best place to start? Google!

Hundreds of thousands of entries explain which commands to issue in order to “set up a Certificate Authority.”

Try to find even one that tells you the commands *and* describes some of the physical and procedural controls commercial CAs use and why running an internal CA without these controls is a bad idea.

Second-Best (and lesser) Practices

Store the root cert on an administrator’s laptop

Hardware Security Modules are so expensive!
And vaults? Don’t even go there!

Most CACOM’s today store the root cert on an administrator’s workstation or laptop. If you start out this way, eventually moving it to a server in a locked datacenter will appear very secure by comparison and nobody will ask about an HSM, a vault or a key signing ceremony. Booyah!

Second-Best (and lesser) Practices

Better yet, copy the root cert for each admin

Certificate provisioning and revocation must be highly available to support critical business apps.

So if the CA is run off of administrator’s workstations, it stands to reason that each administrator must be able to manage certs independently. Give each admin their own copy of the root cert. Or if a central server is used, give a copy of the root to each admin for emergencies.

Second-Best (and lesser) Practices

Don’t use intermediate signer certs

Intermediate signers provide classes of service, isolation, higher security for the root cert, etc.

But someone has to manage those things, and we do that with Notepad. We plan to evaluate several PKI products someday and we will start using intermediate certs then.

Second-Best (and lesser) Practices

Don’t bother with pesky certificate policies!

Certificate extensions can specify policies that control the ways that a certificate can be used.

Understanding and checking certificate policies requires deep skill. Certs with no policies set can be used for many purposes, such as encryption, code signing or signing other certificates. These are very versatile. What could go wrong?

Second-Best (and lesser) Practices

Informal OJT is just fine! What could go wrong?

Security certified administrators who understand X.509 certificates, TLS/SSL and all of the PKCS standards are expensive! Formally building skills in house is less expensive, but takes time.

Fortunately, your average administrator can pick up all they need to know with a few Google searches! Certifications and formal training are SO overrated, right?

Second-Best (and lesser) Practices

Reuse personal certificates

A certificate represents an identity.

Because the Center of Mediocrity is concerned mainly with cost, it is common to re-use the same personal certificate across multiple, even unrelated, systems. A QMgr is a QMgr is a QMgr, right? Generate a single cert called QMgr and be done with it.

Second-Best (and lesser) Practices

Make the certificate stores world-readable

Anyone who can read the configuration files and keystores can use your personal cert.

Fortunately, only authorized admins ever have access to the filesystem in a CACOM so filesystem controls are redundant.

What’s that you say? Layered defense is good? Not when aspiring to mediocrity!

Second-Best (and lesser) Practices

Do not run a revocation server

Credentials need to be both provisioned and revoked. Sometimes they need to be revoked prior to their natural expiry.

A primary characteristic of a mediocre CA is that a revocation responder service is planned for later.

A revocation service that is not highly available is just as bad, possibly worse.

Second-Best (and lesser) Practices

Use home-grown central provisioning tools

Ideally, personal certs are generated where they will be used and not moved. Central management and key backup is possible but difficult to do securely.

In a CACOM, centrally generated certs are distributed over FTP, email, shared drives, etc.

Second-Best (and lesser) Practices

Not knowing the quality of your BP's CA

Sometimes it isn't your CA that places you at risk, but rather your business partner's CA.

A CACOM will assume the partner's certificates are trustworthy without question. The next slides are examples of this.

Second-Best (and lesser) Practices

Trust other people's internal CA root certs

The root cert is the thing that controls who can bind an identity to a key pair. Any trusted root can generate a cert claiming any identity.

Now that the internal CE Center of Mediocrity has become so popular, you will have occasion to trust the internal root CA certificates of your business partners. Go ahead and drop these right into the trust store. What could go wrong?

Second-Best (and lesser) Practices

Unconditionally trust personal certs

If you are using AMS with Privacy (encryption) then you will need to trust the personal cert of the app, user or business partner. A commercial CA will have set the IsCA=No and PathLength=0 flags to ensure that personal certificates cannot also sign other certificates.

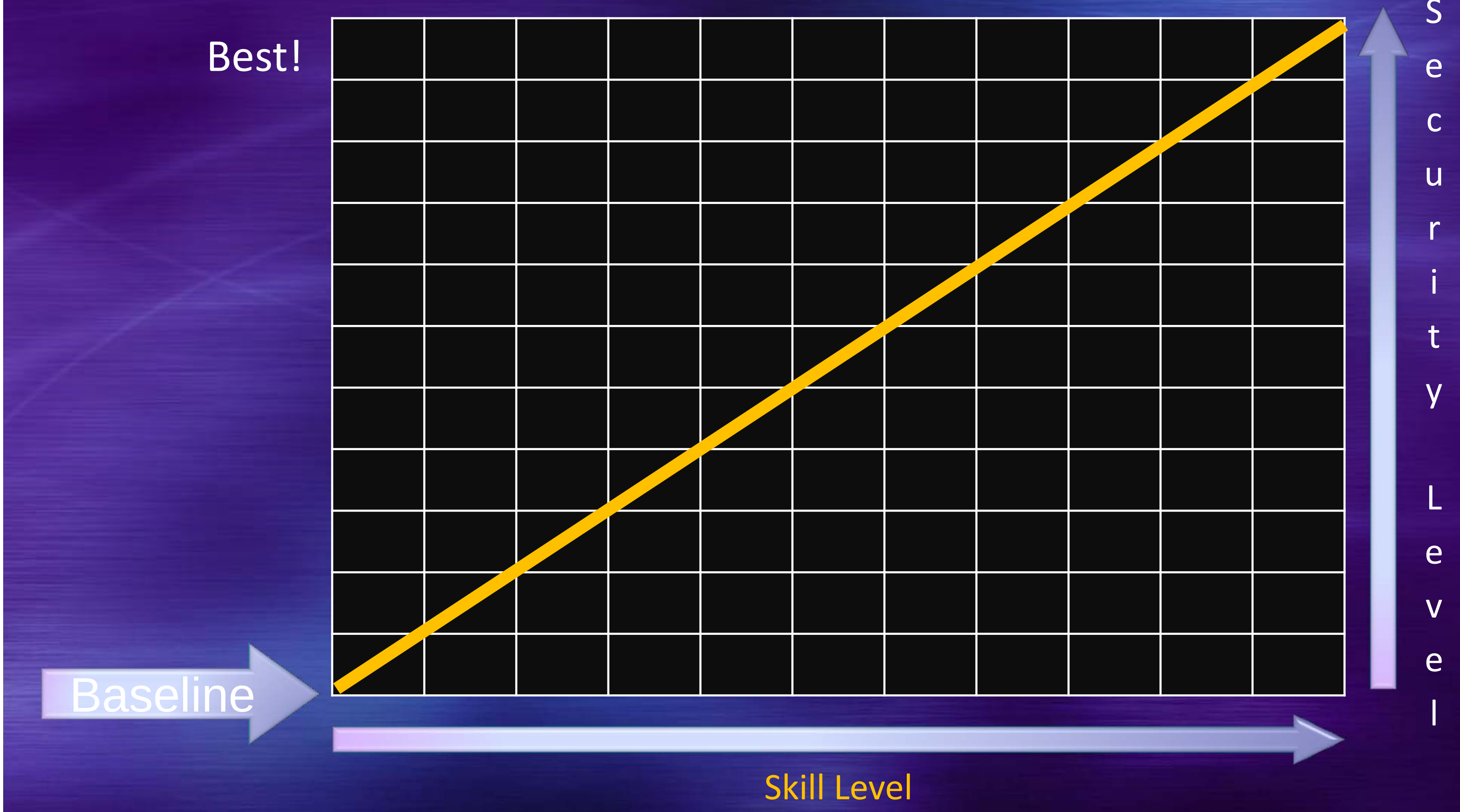
The typical CACOM does not know or care about such things and will place any certificate from any source into the trust store. What could go wrong?

Agenda

- What is a Certificate Authority?
- The role of a CA in the security model
- Why run your own CA?
- Some example CA Best Practices
- CA Second-Best (and lesser) Practices
- The CA Maturity Model
- Wrapping up

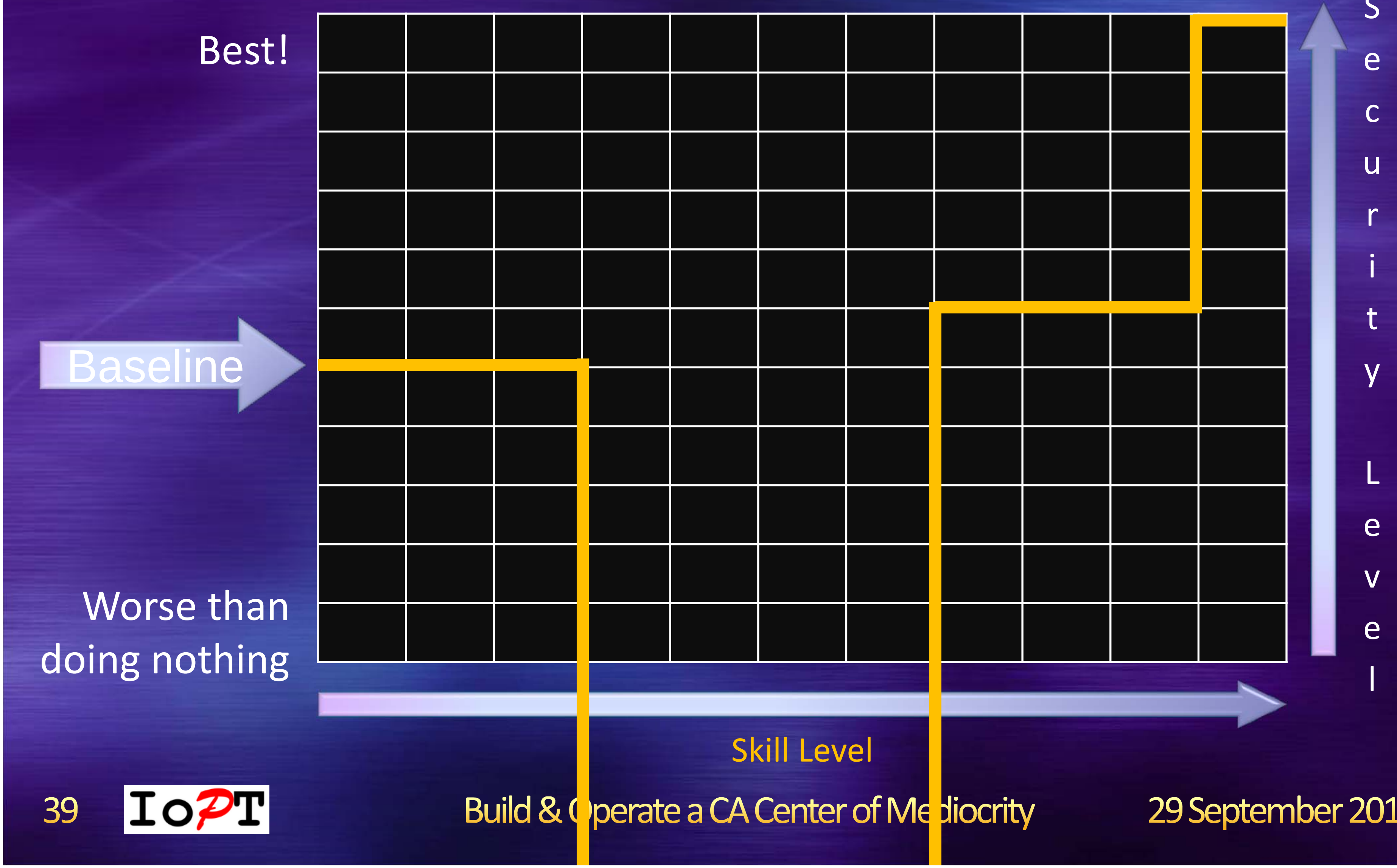
CA Maturity Model

How we think it works



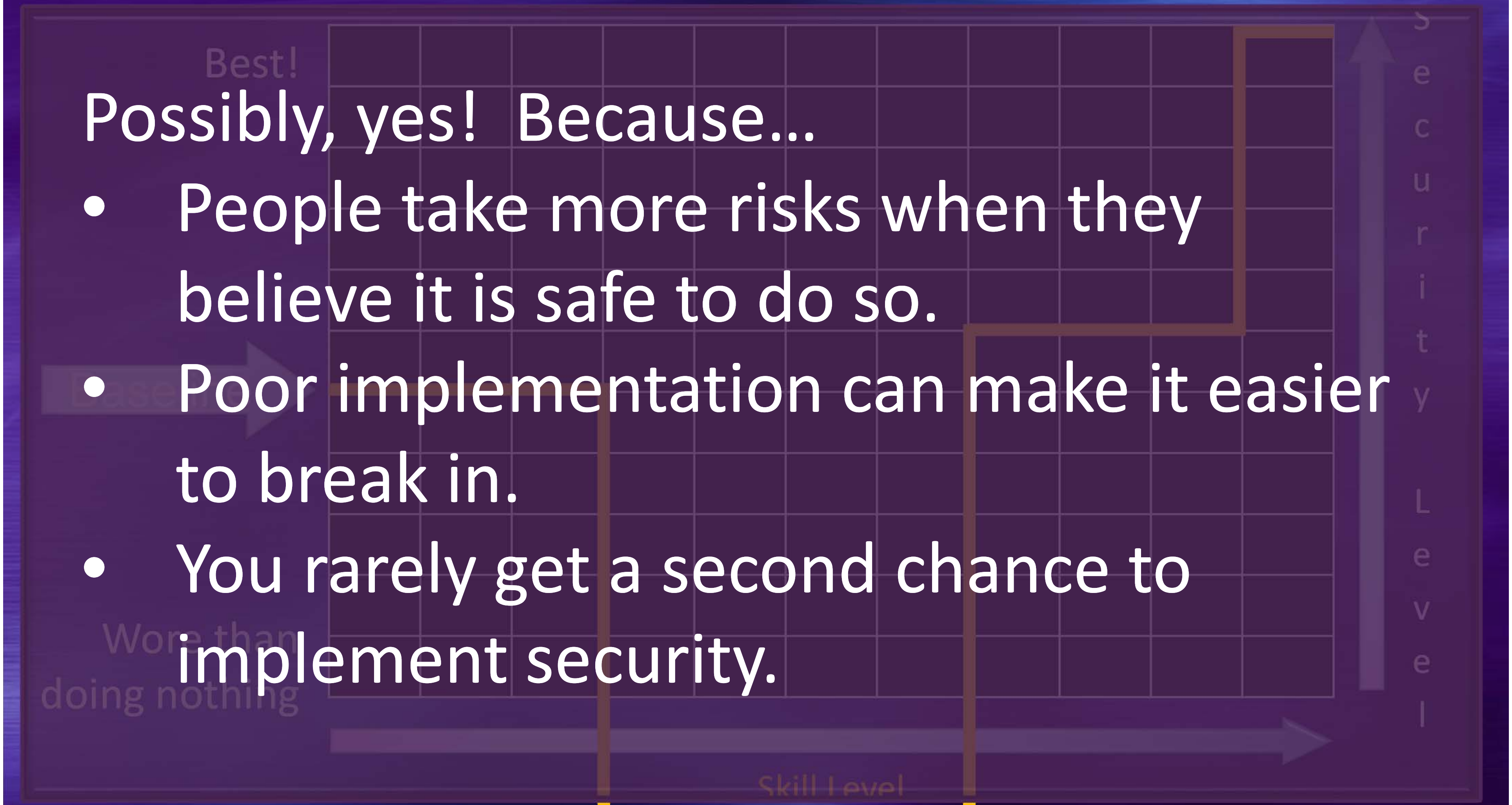
CA Maturity Model

How it actually works



CA Maturity Model

Worse than doing nothing?!?!



Possibly, yes! Because…

- People take more risks when they believe it is safe to do so.
- Poor implementation can make it easier to break in.
- You rarely get a second chance to implement security.

Agenda

- What is a Certificate Authority?
- The role of a CA in the security model
- Why run your own CA?
- Some example CA Best Practices
- CA Second-Best (and lesser) Practices
- The CA Maturity Model
- Wrapping up

Wrapping up

- Running a robust internal CA is expensive.
- *Can* be cost justified, given enough certs.
- Cost savings almost always correspond to some loss of effective security. Know what you are omitting and the offsetting risk.
- Deep skill is essential! Formal training is highly recommended.
- Beware of other people's CACOM!

Questions & Answers

Thank you!

T.Rob Wyatt
Managing Partner, IoPT Consulting
704-443-TROB (8762)
t.rob@ioptconsulting.com
https://ioptconsulting.com

Certified for IBM. WebSphere. software

Certified for IBM. Power Systems