CAcert and the Audit

CAcert and the Audit

- To get CAcert Root into Browsers
- => Audit is required
- => which requires:
  - management
  - policies + practices
  - review of business & systems
- => against policies and practices!

CAcert and the Audit

CAcert has three major business areas:

a) Assurance

b) Systems

c) Community

c. Audit and Assurance

Audit - Assurance

a. Assurance

- Assurance Policy is in full POLICY status

➔ It is **binding** on all Assurers

- The process of Assurance can be reviewed.

Audit - Assurance

- a.i Three major processes for Audit:
  - CATS Assurer Challenge
  - ATE – Assurer Training Event.
  - Co-audit

Audit - Assurance

CATS Assurer Challenge

- CAcert Automated Testing System
- 25 multiple-choice questions, 80% to pass.
- You can do it as many times as you like
- Sets a minimum standard

Audit - Assurance

Assurer Training Event

- ATE is here today*!*
- 20 so far in Europe
  - Germany, Innsbruck, Belgium, Denmark, Paris, London, Prague, Budap
- Tells you what you need to know
- Highly recommended

Audit - Assurance

Co-Audit

- You assure the Co-Auditor
- Collect Evidence of Assurance-to-policy
- verifies quality of assurance:

“A.2.y The CP details how the CA verifies that

RAs operate in accord with the CA's policies.

- (Also: lots of feedback)

Audit - Assurance

- a.ii Detailed Changes to Assurance
- Member statements

  „Information is correct”

  „I agree to CCA”
- Assurer states:

  „was conducted to Assurance Policy”
- CARS: CAcert Assurer Reliable Statement

Audit - Assurance

a.iii The review of Assurance

- ✔ 2009 February – June evidence gathered
- ✗ Audit terminated for other reasons.
- Possibility of Assurance Review late 2010

b. Audit and the Systems

Audit - Systems

b.i Systems Review required:

- secure hosting: Oct 2008.
  (Teams, Facility, Machines)
- Security Policy: March 2009 p20090327
- Software
  (CPS: July 2009 p20090706, Teams)

Audit - Systems

b.ii Systems now has

- Teams: critical sysadms, access engineers
- Security Policy in DRAFT status

  Binding on critical roles:

  Sysadms, Access Engineers, Support, Software

- Systems in secure facility: BIT Ede, NL.

Audit - Systems

b.iii Issues to resolve:

- Non-critical systems - underway
- Roots - planning
- Disaster Recovery - thinking
- Team size – need more capacity

Audit - Systems

b.iv Software

- Review at Innsbruck April 2009:

  „*Serious difficulties in maintaining, improving and securing.”*

  „*Cannot form conclusion over software.”*

- Track 1: New software development team, new design, new build: “BirdShack”

- Track 2: Old software in “maintenance mode”

c. Audit and the Community

Audit - Community

The audit criteria:

DRC for "David Ross Criteria".

- David is a retired quality engineer
- He started CAcert's Audit in 2005
- Was called to Grand Jury duty

Audit - Community

DRC has a strong feature. It requires the

- Risks,
- Liabilities, and
- Obligations

to be clearly stated to everyone!

Audit - Community

This raised several huge barriers for the CA:

- what exactly are the R/L/O?
- who do they apply to?
- are they reasonable?
- and, how do we deal with them?

Audit - Community

The barrier of R/L/O is subtle:

- DRC doesn't ask them to be fair, but
- disclosure makes us consider them, and
- Cacert people want things to be fair!

which means *we have to deal with them!*

Audit -Community

One big result of this thinking process

was that we required a:

CAcert Community Agreement

and.....

Audit - Community

The CCA had to do these things:

a. make us into a mutually-binding Community.

b. state the R/L/O

c. limit the liabilities → 1000 Euros

d. allocate the liabilities → back to the Members

Audit - Community

How do we allocate the liabilities?

- By making our own forum of dispute resolution

  „Arbitration”

- agreeing to be bound to that resolution,

  CAcert Community Agreement 3.2

- writing a Policy to control that process:

  Dispute Resolution Policy

Audit - Community

Summary: The original “Why” of Arbitration:

- Audit (DRC) forced *disclosure* of Liabilities
- simple fix: *limiting*
- complex fix: *allocation*
- the safe and cheap way to allocate is:

to use our own Arbitration

(Last section discusses „How“.)

d. Back to the Audit

Audit - Past

Audit went into high gear early 2009, but:

- Lack of capacity
- Overruns in time
- Lack of funding

Audit terminated July 2009

Audit - Future

What is Community doing next?

- Rebuild Software

  as with: Assurance, Arbitration, Support, Systems, Management

- Push Audit Work to the Community

  Only the Community has the scale and capability

- Prepare for Review over Assurance

  Today, your chance: ATE!

- Funding!