CeSeCore
Common Criteria Certified
Open Source Software

Tomas Gustavsson
PrimeKey Solutions AB

www.ejbca.org
www.cesecore.eu
1. Common Criteria
   - What?
   - Why?
2. CeSeCore
   - Open Source certified Security Core
   - www.cesecore.eu
3. Open Source and Common Criteria
   - Common Criteria in practice
   - How is an open source project affected?
Part 1

• Common Criteria
Common Criteria
What is it?

• Short for...
  ○ Common Criteria for Information Technology Security Evaluation

• A standard...
  ○ For IT security certification defined by ISO/IEC 15408

• Provides...
  ○ Trust that processes for specification, implementation and evaluation has been performed in a rigorous and standardized way.

Common Criteria
Is used by...

- **Users**
  - Specify requirements in security and trust.

- **Suppliers**
  - To market security aspects of their products.

- **Test laboratories**
  - To evaluate if products fulfill their claims.

Contents based on Wikipedia, Common Criteria,
Governments, banks and large enterprises require certified software.

As an open source-supplier same rules apply as for proprietary suppliers.

- In or out!
How does common criteria certification work?

- **Sponsor** - pays
- **Developer** - implements
- **Lab** - evaluates
- **Certification body** - issues (signs) the certificate
Common Criteria
Tests...

- **Documentation**
  - The documentation of security functionality is complete

- **Implementation**
  - That security functions does what they claim to do

- **Development environment**
  - That development is done in a secure environment

- **Development process**
  - That development is done in a secure way
What is common criteria?

- PP
- ST
- FSP, TDS, IMP
- PRE, OPE
- ARC, ALC
- ATE
- TOE, TSF, TSFI, EAL

A process in itself to learn the terminology.
• **Target of Evaluation (TOE)**
  - The product to be evaluated.

• **Protection Profile (PP)**
  - Document that identifies security requirements for a type/class of security products (ex., smart cards, certificate issuing, firewalls, etc.).
  - When suppliers choose to certify their products against one (or more) PPs, it works as a template for the products Security Target.
  - Is typically created by users of products or user groups, to benefit standardization and ease comparisons.

Security Target (ST)

- Document that describes the product, identifies its security attributes and level of trust.
- Most commonly, claims conformance to one or more Protection Profiles.
  - In that case, explains how the security functionality is provided by the product.
  - If not, it must also define the security problem, goal and requirements that the product addresses.

Security Functional Requirements (SFRs)

- Individual security functions that can be provided by a product
- Most of them are part of a standard catalog, but additional ones can be defined
- Example from standard catalog:

<table>
<thead>
<tr>
<th>FAU_STG.1 Protected audit trail storage</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FAU_STG.1.1</strong></td>
</tr>
<tr>
<td><strong>FAU_STG.1.2</strong></td>
</tr>
</tbody>
</table>

Security Assurance Requirements (SARs)

- Not a flu, every Security Target has several of these.
- Description of steps that must be obeyed during development and evaluation of the product, defined by the standard catalog.
- Example from standard catalog:

**ADV_FSP.1 Basic functional specification**

Dependencies: No dependencies.

Developer action elements:

**ADV_FSP.1.1D** The developer shall provide a functional specification.

**ADV_FSP.1.2D** The developer shall provide a tracing from the functional specification to the SFRs.
Evaluation Assurance Level (EAL)

- Numeric ranking that described the level of assurance of an evaluation, from 1 (easiest) to 7 (hardest)
- Developers can chose a base EAL level and add extra requirements
- Example (EAL4):

<table>
<thead>
<tr>
<th>Assurance Class</th>
<th>Assurance components</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADV: Development</td>
<td>ADV_ARC.1 Security architecture description</td>
</tr>
<tr>
<td></td>
<td>ADV_FSP.4 Complete functional specification</td>
</tr>
<tr>
<td></td>
<td>ADV_IMP.1 Implementation representation of the TSE</td>
</tr>
<tr>
<td></td>
<td>ADV_TDS.3 Basic modular design</td>
</tr>
<tr>
<td>AGD: Guidance documents</td>
<td>AGD_OPE.1 Operational user guidance</td>
</tr>
<tr>
<td></td>
<td>AGD_PRE.1 Preparative procedures</td>
</tr>
<tr>
<td></td>
<td>ALC_CMC.4 Production support, acceptance procedures and automation</td>
</tr>
</tbody>
</table>
Common Criteria standard: A large pile of documents that describes predetermined requirements. Primarily targeted at evaluators.

Protection Profile: Pre-determined requirements on a certain class of software, ex Certificate Authority, smart card, Operating system. Often created by a government like NIST, DoD, EU

From this the evaluation of the software is started.
Security Target: Described the security functions the software has and the threats they protect against. In principle a copy of the Protection Profile.

Functional Specification: Describes all security functions in detail.

Design Specification: Describes all security modules and interactions.

Implementation Representation: Doxygen documentation of all the source code files.
Preparative Procedures: How to configure the system.

Operational User Guidance: How to use the system.

Test plan: Tests and protocols for all security functions.


Configuration List: List of all files and dependencies.
Common Criteria

How are the documents connected?

- TOE has TSF, TSF has TSFI
- **Complete mappings** between all documents. You cannot describe a TSF in the FSP that does not correspond to a test in the ATE
- ...

![Diagram showing mappings between FSP, TDS, and Implementation concepts]
CeSeCore

Part 2

• CESeCore
CeSeCore

What is that?

- EUREKA Eurostars funded project
  - Started in April 2009.
  - Finished (ish) in April 2012
- Project name stands for
  - Certified Security Core
- Aimed to...
  - Produce an open source security core product and to have evaluated according to Common Criteria EAL4+
CeSeCore

Advantages

• For Customers
  - Be sure to be using a product evaluated according to market standard security best practices

• For integrators
  Ability to build up on the core to...
  - Instantly increase the security level of their product
  - Develop security-related products with a shorter time-to-market
  - Be able to more easily achieve and maintain a Common Criteria evaluation for their own product
  - Downsize their source code, reducing the complexity they need to manage and support
• Assurance Level
  - EAL 4+ (augmented with ALC_FLR.2)

• Protection Profile
  - Decision to comply with Security Level 3 of the Certificate Issuing and Managements Components Protection Profile.

• Common Criteria Version
  - Claimed compliance with version 3.1
CeSeCore
So what is it really?

- Java Enterprise library with APIs for
  - Security Audit
  - Certificate management with issuance, OCSP response creation etc.
  - Key Management (incl. Hardware Security Modules)
  - Access Control with Role Management
  - Trusted Time
  - Backup and Recovery (of security functionality)
CeSeCore
So what is it really?

Java/JEE5 source code and Eclipse project

Cesecore-client.jar
Cesecore-ejb.jar
Cesecore-entity.jar

Cesecore.ear – test app
CeSeCore
So what is it really?

• Integrate into your own JEE application
  – Remote interfaces with full access control
  – Local interfaces override some controls

• Use tools and standalone classes
  – Hardware Security Module usage
  – ...

PrimeKey
public interface AccessControlSession {

/**
   * Checks if the current user is authorized for the given resource.
   * Will by default accept recursive accept values.
   * @param resources String identifier(s) of the resource(s) in question.
   * @return True if user is authorized, false if not.
   */
   boolean isAuthorized(AuthenticationToken authenticationToken, String... resources);

@Override
@TransactionAttribute(TransactionAttributeType.SUPPORTS)
public boolean isAuthorized(final AuthenticationToken authenticationToken, final String... resources) {
    if (updateNecessary()) {
        updateAuthorizationTree();
    }
    return isAuthorized(authenticationToken, true, false, resources);
}
CeSeCore
Target Of Evaluation

- Target of Evaluation

Diagram:
- Abstract machine
- Application server
- HSM
- CESeCore
- Configuration
- TOE Boundary
- Applications
- DB
CeSeCore
Architecture

Applications
- EJBCA
- Timestamp Server
- Mail Sign Server
- Secure Archive Server

TOE
CESECORE

Environment
- Application Server (Jboss, BEA Weblogic, etc.)
- Database engine
- Java Virtual Machine (Oracle JVM, BEA Rockit, IBM VM, etc.)
- Operating System (Windows, Linux, etc.)
- Reliable Time Source
- Hardware (CPU + Storage + Memory)
- PKCS#11 Driver
- HSM
### CeSeCore

**Evaluated Platforms**

<table>
<thead>
<tr>
<th>Item</th>
<th>Platform A</th>
<th>Platform B</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS</td>
<td>Red Hat Enterprise Linux v5.5 x86_64 (64bit)</td>
<td>Windows Server 2008 Enterprise with SP2 x86 (32bit)</td>
</tr>
<tr>
<td>JDK</td>
<td>Oracle JDK 1.6.0_20 64 bit</td>
<td>Oracle JDK 1.6.0_23 32 bit</td>
</tr>
<tr>
<td>Database</td>
<td>PostgreSQL 9.0.2</td>
<td>MySQL Community Server 5.1.55</td>
</tr>
<tr>
<td></td>
<td></td>
<td>MySQL Connector/J 5.1.15</td>
</tr>
<tr>
<td>Application Server</td>
<td>JBoss 5.1.0.GA</td>
<td>Glassfish v2.1.1</td>
</tr>
<tr>
<td>HSM</td>
<td>Safenet LunaSA</td>
<td>Utimaco CryptoServer</td>
</tr>
<tr>
<td></td>
<td>Firmware version 4.6.8</td>
<td>Firmware version 2.30.2</td>
</tr>
</tbody>
</table>

**Test Platforms**

<table>
<thead>
<tr>
<th>Item</th>
<th>Platform A</th>
<th>Platform B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Apache ant</td>
<td>v1.8.2</td>
<td>v1.8.2</td>
</tr>
<tr>
<td>Subversion client</td>
<td>Subversion client 1.6.6</td>
<td>SlikSVN 1.6.15</td>
</tr>
</tbody>
</table>

**Other required software**
Functional specification describing the high level functional architecture...
Identifying all existing interfaces and operations...
## CeSeCore

### Project documentation (3)

#### Restore

<table>
<thead>
<tr>
<th>Purpose</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>The purpose of the restore method is to, from a backup set created by the backup method, restore the original content from an earlier backup.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method of Use</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>The restore method of CeSeCore will be used to restore backup of items not already covered by other procedures. The restore method can take form of function call, script or other methods.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Configuration backup</td>
<td></td>
<td>Configuration from earlier backup operation</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Actions</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Verify pre-requisites</td>
<td></td>
</tr>
<tr>
<td>Admin Role allows restore</td>
<td></td>
</tr>
<tr>
<td>All needed verification and decryption keys can be located</td>
<td></td>
</tr>
<tr>
<td>The needed configuration is correct</td>
<td></td>
</tr>
<tr>
<td>Restore configuration</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Outputs</th>
<th>Name</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Restored TOE</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Error Messages</th>
<th>Type</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>AccessControlException</td>
<td>In case the admin role doesn't allow restore</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Log Events</th>
<th>Event</th>
<th>Additional Details</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Restore</td>
<td></td>
</tr>
</tbody>
</table>

| Related SFRs | FDP_CIMC_BKP.1 CIMC backup and recovery |

| Operation user interfaces | CLI |

![CC BY-SA 4.0](https://creativecommons.org/licenses/by-sa/4.0/)
Design specification describes in detail all sub-systems

The backup and recovery subsystem is used to create and restore backups of CESeCore specific data that is not backed up by other means. The amount of data is limited. The subsystem does not provide any external features to applications. This subsystem depends on other subsystems in CESeCore in order to perform its operations.

<table>
<thead>
<tr>
<th>ID</th>
<th>Originating Subsystem</th>
<th>Destination Subsystem</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>Backup &amp; Recovery</td>
<td>Access Control</td>
</tr>
<tr>
<td></td>
<td>Before allowing an operation it is necessary to check if the subject is authorized to perform the operation.</td>
<td></td>
</tr>
<tr>
<td>b</td>
<td>Backup &amp; Recovery</td>
<td>Key Management</td>
</tr>
<tr>
<td></td>
<td>In order to protect or unprotect the sensitive information each operation needs to use a key.</td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>Backup &amp; Recovery</td>
<td>Security Audit</td>
</tr>
<tr>
<td></td>
<td>After performing the operation it is necessary to register that event in the security audit log.</td>
<td></td>
</tr>
</tbody>
</table>
Implementation representation details the mapping between the specifications and the implementation.

The CESecore project

Heading towards Common Criteria certification

In order to make the CESecore security core publicly available for integration in numerous security based applications, the CESecore project aims to realize a CESecore Common Criteria EAL4+ certification.

Reusable Java library

Taking on the form of a common security function Java library, the CESecore security core will provide a reusable base for implementing third-party trustworthy systems. The ready-made CESecore library simplifies development of secure software, and allows for Common Criteria certification to be easily extended to any related product or service.

Integrator friendly

CESecore integrators will be able to correct, improve and extend their applications at any time - without the need to perform frequent system re-evaluations, nor perform continuous checks of the security functions implemented by the security core (including features like digital signature creation/validation, digital certificate and CRLs creation, key management and maintenance of a secure audit log).
CeSeCore
Project documentation (7)

- Security Architecture
- Life Cycle Support
- Test Plan
- Preparative Procedures
- Operational User Guidance
- ...

PrimeKey
Common Criteria and OSS

Part 3

- Common Criteria in practice
- How does Common Criteria affect a project
Common Criteria in practice

Some preconceptions about Common Criteria?

- Expensive?
- Waste of time?
- Just paper, not technical?
- Certified products are old version?
- Certified products are not better than others?
- ...
Common Criteria in practice

The truth about Common criteria

- Takes a lot of time.
- Costs a lot of money.
- Is technical with code inspection and penetration tests.
- Certified versions quickly becomes old.
- Gives product improvements.
- Improves security of products, if used correctly.
- Non-certified products can be as secure, but commonly not verified by external party.
Common Criteria in practice

Common misconceptions:
- Certification guarantees secure software
- Certification is worth nothing
(the truth is somewhere in between)

A Common Criteria certification assures, with a certain assurance level that the certified software works as it is documented to work.

Therefore Protection Profiles and Security Target are important.
Common Criteria in practice

Most common practical issue:

- Customers actually think that they **must** stick with the **exact certified software version**.

- This causes **security problems**.

This mindset causes unaware users to run several year old versions of software (OS etc) with tons of known security issues.

If penetrated, servers are easily compromised.
Formally, only the tested version is certified.

In case of minor version evolution, you can only explain to your customer that "this minor version corrected few bugfixes in the certified version, so that it is recommended to use this last version".

There is a process of "Assurance continuity" to ask ANSSI a "Maintenance report" for such minor version but it is recommended to use if only for release "medium" like 5.1 and not for 5.0.5.
Common Criteria and OSS

What does it cost?

- Time: 2 years with at least 3 people full-time during more than 1 year
- Cost: >500,000 just for the lab
Few Open Source products are certified
  - From corporation
  - RedHat, IBM, PrimeKey, ...

Not freely downloadable, Enterprise versions.

Many commercial products that builds on open source.
Common Criteria and OSS

Difference between OSS and proprietary SW?

- Public repository
- Control over development machines
- Where from development is done
- Who develops, legal(?) control over developers
Common Criteria and OSS

How does Common Criteria support OSS?

- Control over developer machines - policy/contract
- Where from is development done – public svn/git, wiki, ...
  - HTTPS and authentication
- Legal control over developers - contract
Common Criteria and OSS

How does it affect an OSS project?

Life Cycle och Common Criteria requirements

- **Controlled access** to repository
  - Logically and physically
  - No sf.net

- **Processes** for:
  - Issue management
  - Code review
  - QA
  - Release
Common Criteria and OSS

How does it affect an OSS project?

Life Cycle and Common Criteria requirements

- All personnel
  - Sign user responsibility policy
  - Access lists who has access to what
    - Logically and physically
- External committers:
  - Same user responsibility policy
  - Review process
Common Criteria and OSS

Effect on EJBCA.org/CeSeCore.eu?

Already had Jira, Fisheye, Hudson, Clover, ...

- Moved repository to own hosting
- Introduced access lists and user responsibility policy for all committers
  - More administration
- Some more QA for every issue, merging and extra documentation.
  - Slightly slower development per issue
  - Better quality assurance per issue