Compliance testing of electronic certificates

Michael Jahnich PO eIDAS Inspector
Presentation at CA/Browser Forum Meeting
October 2020
Agenda

1. Our motivation
2. Tool requirements for compliance testing
3. Tool design
4. Discussion
Our motivation
Where we started with certificate testing

- About achelos
  - Manufacturer-independent consulting and software development company
  - Focus: software testing and conformance testing
  - Segments: Health, Public, Mobility, IoT, Security

- The origin of certificate testing at achelos
  - German health telematics infrastructure
  - Test PKI (2015)
  - Problem: compliance of certificates to the specifications
Where we started with certificate testing

**Motivation:** Requests from customers from different PKI domains

→ EU regulation 910/2014
→ Payment Service Directive

Technical specifications

- ETSI EN 319 412, parts 2 to 5
- ETSI TR 119 495 (PSD2)
Tool requirements for compliance testing
Conformance test tool requirements

- Objectives of using a compliance test tool
  - Prove compliance to the standards and requirements
  - Improve interoperability

- Traceability of test results to requirements (RFC 2119)
- Auditability by means of reporting
- Validated test case implementation
- Completeness of test coverage
- Adaptability to changing standards
Use cases

Conformity assessment
- Check samples of certificates generated
- Adherence to standards, guidelines or specifications
- Adherence to CPS

Certificate production
- In-the-loop validator
- Quality control
- Check certificates in database

Request → Create → Verify → Provide
Additional requirements from QTSPs

- Assessment of Certification Practice Statement
- RFC 3647 → Certificate Profile
- Requirements described in chapter 7.1
  - Certificate extensions
  - Algorithm OIDs
  - Name forms and constraints
  - ...

Public standards, guidelines and specifications (ETSI, RFC)

TSP specific requirements (CPS)
Traceability of requirements

Each test results traces back to a requirement (transparency)
Each specification requirement is met (completeness)
Every test is automatically reproducible.

Test report maps every test result to a specification requirement (with an ID, e.g. chapter)
Auditability / Reporting

- **Report Summary**
  - Certificate under Test
  - Time/Date of test run
  - Test cases performed (passed, failed, NA, inconclusive)
  - Specification incl. **spec version**

- **Report per test case**

- **Options**
  - Report may be signed
  - Report checks if test case implementation has been unchanged (signature)

Specification Requirement
(ID or chapter)

Test case specification
(description)

Test result
Actual value
Test coverage and completeness

- Structured creation of a test specification
  - Decompose specification into single identifiable requirements
  - Create at least one testcase per requirements
  - Mapping from requirement to test

- Not all requirements can be checked by certificate linters
  (Organisational requirements)
- Checks by means of assessment/audits or proprietary tests (CPS)
Validation of test implementation (Correctness)

Validation by 3rd party

Validation with working samples

Common to many compliance programs

Defined processes for errors found in the field and error corrections

1. Quality assurance
   - Internal software testing

2. Test validation
   - Test tool report by 3rd party test lab

3. Official tool qualification
   - Certificate by compliance program
Tool design
Some tool design decisions

- Extensibility
  - Create custom test cases (e.g. to implement CPS, chapter 7.1)
  - Modify standard test cases

- Modularity
  - Tool -> performance, reporting, robustness, test case mgmt
  - Test case (policy) -> certificate compliance

- Versioning of tests
- Easy to integrate into automated processes
  - Service (REST)
  - Scriptable (CLI)
  - Light weight

- Independent of a platform
  - Java
  - XML
Modularity

- Separation of tool and test cases (policy)
  - XML based language to specify requirements and test cases
  - XML: human and software readable
  - Generation of reports as PDF files for archiving

Diagram:
- XSD
- PolicyEngine
- xmlReporter
- pdfReporter
- certificateVerifier
- Policy A
- Policy B
- Policy C
- Report A
- Report B

© 2020 achelos GmbH
### XML based test case (report) implementation

<table>
<thead>
<tr>
<th>4.1.1.2</th>
<th>4.1.1.2.(1.1)_sigAlg_present</th>
<th>SignatureAlgorithm</th>
<th>SignatureAlgorithm parameter MUST be provided.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>4.1.1.2.(1.2)_sigAlg_equals_signature</td>
<td>SignatureAlgorithm</td>
<td>Signature algorithm identifier value MUST be the same as the value int the signature field.</td>
</tr>
<tr>
<td></td>
<td>listContainsElements</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4.1.1.3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

PASSED

```
- <t:testStep expandable="true" uuid="4a2b2e77-514c-487d-8398-29d1ccf6ec1d" verdict="PASSED" expected="SignatureAlgorithm parameter MUST be provided." target="SignatureAlgorithm" ref="4.1.1.2.(1.1)_sigAlg_present">  
  - <c:isPresent expandable="false" uuid="1eaf2e0f-8a57-4621-b255-a4b8b5f93e1b" result="PASSED">  
    <ac:signatureAlgorithm value="SHA256WITHRSA" parsable="true"/>  
  </c:isPresent>  
</t:testStep>

- <t:testStep expandable="true" uuid="2031e6d9-35c1-464a-bd1d-72eba2d06021" verdict="PASSED" expected="Signature algorithm identifier value MUST be the same as the value int the signature field." target="SignatureAlgorithm" ref="4.1.1.2.(1.2)_sigAlg_equals_signature">  
  - <c:listContainsElements expandable="true" uuid="2ea98a4c-401f-484a-b295-04e7b0eb1feb" result="PASSED" noOfMinElements="1" noOfMaxElements="-1" allowDuplicates="false" exclusive="true">  
    <ac:signature value="[1.2.840.113549.1.1.11]" parsable="true"/>  
  </c:listContainsElements>  
</t:testStep>
```
Reporting – test information

Test information:

Subject: 2.5.4.13=test certificate,1.3.6.1.4.1.311.60.2.1.3=DE,BusinessCategory=Business Entity,CN=achelos.de,SERIALNUMBER=f263cf73-b090-GmbH,L=Paderborn,C=DE
Issuer: CN=BVtrust QA TEST PSD2 QWAC CA PKCS1_v1_5 R2019,OU=BVtrust,O=Bank-Verlag GmbH,C=DE

Policyname: RFC5280SubscriberPolicy_v05.2008
StartingTime: 2020-10-21T15:30:16.907113Z
Test-duration: 1414ms
Number of tests passed: 98
Number of tests failed: 0
Number of warnings: 3
Number of tests inconclusive: 45
Reporting – certificate information

Version: 3

Serial Number: 37540259606020239571724000419

Issuer DN: CN=BVtrust QA TEST PSD2 QWAC CA PKCS1_v1_5 R2019, OU=BVtrust, O=Bank-Verlag GmbH, C=DE

Start Date: Wed May 29 11:12:47 CEST 2019

Final Date: Thu May 28 11:12:47 CEST 2020

Subject DN: 2.5.4.13=test certificate, 1.3.6.1.4.1.311.60.2.1.3=DE, Business Category=Business Entity, CN=achelos GmbH, L=Paderborn, C=DE


modulus:
ac201dd180c7929cf15078691485b9c8d0930c092e0e8eb1fa2c8a00de4ef18120b4af7d7b1eb32b5f2454ae30b4739844
2528684c3d5bbde7ca629d2c7009774596ad9253bd92abef5f9df4d9d40221060fb571804a346718edc835cdded994da5
7ce8ac69d7e7652f3003f99fa646dd1517f66b13202305c887273238e512f8dbb8c6c16cbbdd01a18aaba97f65ad773e0115
7b2cd4af37eb69131cc76a53c0649a7f020f22908b6524f006da8188cb07cc30956feddd6b4593d7753e866d7510ceba72c97
2e82b629b3816b7f4a46962ff15590ff95da05cb05e318d34cbb04fe967e8aebe6c3cc25bca2b1edae9f4939da0b2adc92
e16a2bf01b63f41d40bee6976081a6ffda81414711f0cb3c816ef1e101623e7ff5c0c05c846e7d6242d

public exponent: 10001

Signature Algorithm: SHA256WITHRSA

Signature: 5697d8de68d6c3491429a6ade0174e7df43b202a
f0e5e4e4b791493b3f6a7d4f869d9e93ah70ee05

© 2020 achelos GmbH
Reporting – report by test case

<table>
<thead>
<tr>
<th>name</th>
<th>target</th>
<th>description</th>
<th>status</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC5280SubscriberPolicy_v05.2008</td>
<td>4.1.1.2</td>
<td></td>
<td>PASSED</td>
<td>2.5.4.13=test certificate,1.3.6.1.4.1.311.60.2.1.3=Entity,CN=achelos.de,SERIALNUMBER=a5d2-4d36bafa680,2.5.4.97=PSD1 GmbHL=Paderborn,C=DE</td>
</tr>
<tr>
<td>4.1.1.2</td>
<td>SignatureAlgorithm</td>
<td>SignatureAlgorithm parameter MUST be provided.</td>
<td>PASSED</td>
<td></td>
</tr>
<tr>
<td>isPresent</td>
<td></td>
<td></td>
<td>PASSED</td>
<td></td>
</tr>
<tr>
<td>4.1.1.2</td>
<td>SignatureAlgorithm</td>
<td>Signature algorithm identifier value MUST be the same as the value in the signature field.</td>
<td>PASSED</td>
<td>SHA256WITHRSA</td>
</tr>
<tr>
<td>listContainsElements</td>
<td></td>
<td></td>
<td>PASSED</td>
<td>[1.2.840.113549.1.1.11]</td>
</tr>
<tr>
<td>4.1.1.3</td>
<td></td>
<td></td>
<td>PASSED</td>
<td></td>
</tr>
<tr>
<td>4.1.1.3</td>
<td>SignatureValue</td>
<td>Certificates MUST provide a signatureValue parameter</td>
<td>PASSED</td>
<td></td>
</tr>
<tr>
<td>isPresent</td>
<td></td>
<td></td>
<td>PASSED</td>
<td></td>
</tr>
<tr>
<td>4.1.1.3</td>
<td>SignatureValue</td>
<td>Signature algorithm identifier MUST be the same as in the algorithm identifier field</td>
<td>PASSED</td>
<td>5697D8DE68DD6C3491429A6DE</td>
</tr>
<tr>
<td>isValidSigned</td>
<td></td>
<td></td>
<td>PASSED</td>
<td></td>
</tr>
</tbody>
</table>
Quality Assurance

- QA during build process (500+ Tests incl corrupted certificates)

- Validation of test case implementation
  - Through customer projects with respect to eIDAS
  - Based on Zlint test certificates for BR and EV Guideline
  - Comparison of Zlint and eIDAS Inspector results

- Automatic generation of test specifications for each policy
  - Can be used by 3rd party for tool validation
Discussion
Discussion

- Validation of test case implementations
- Use of the eIDAS Inspector for CA/Browser Forum guidelines
- Open Source of the certificateVerifier for the Forum
Vielen Dank | Thank you

achelos GmbH
Vattmannstraße 1 | 33100 Paderborn | GERMANY
T +49 1511 7455961 | michael.jahnich@achelos.de
achelos.de | IoT.achelos.com