MTAT.07.017
Applied Cryptography

Digital Signatures (XAdES)

University of Tartu

Spring 2021

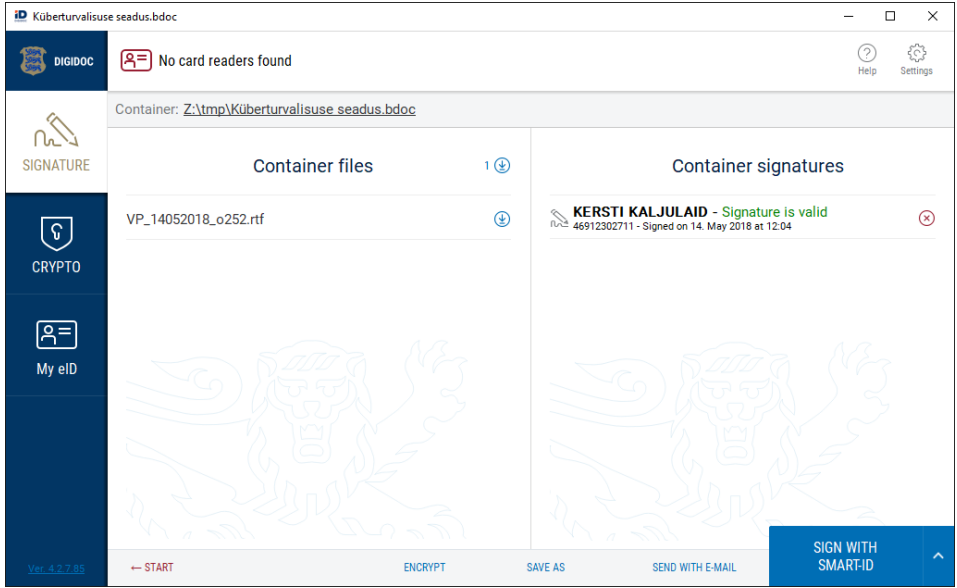
eIDAS Regulation[1]

Article 3:
(10) 'electronic signature (ES)' means data in electronic form which is attached to or logically associated with other data in electronic form and which is used by the signatory to sign;
(11) 'advanced electronic signature (AdES)' means an electronic signature which meets the requirements set out in Article 26 *(uniquely linked to the signatory)*;
(12) 'qualified electronic signature (QES)' means an advanced electronic signature that is created by a qualified electronic signature creation device, and which is based on a qualified certificate for electronic signatures;

Article 25:
2. A qualified electronic signature shall have the equivalent legal effect of a handwritten signature.

[1] Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC

Estonian digital signature – QES



Legal effect of a Qualified Electronic Signature

The same as of a handwritten signature

- Can be used to sign contracts
  - Most of the contracts do not have to be in writing
- Must be used to sign legal acts
- Can be used as evidence in court
  - Can unsigned e-mails be used as proof?

Code of Civil Procedure:
§ 277. Contestation of authenticity of documents
(3) Authenticity of an electronic document bearing a digital signature may be contested only by substantiating the circumstances which give reason to presume that the document has not been prepared by the holder of the digital signature.

- What could be these circumstances?

Qualified Trust Service Provider (QTSP) – CA

- Only a QTSP may issue qualified certificates for electronic signatures
- Qualified status granted by supervisory body
  - Estonian Information System Authority (RIA)

Article 22
1. Each Member State shall establish, maintain and publish trusted lists, including information related to the qualified trust service providers for which it is responsible, together with information related to the qualified trust services provided by them.

- Estonian Trusted List (signed by RIA): `https://sr.riik.ee/tsl/estonian-tsl.xml`
- EU List of Trusted Lists (signed by Maarten Joris Ottoy): `https://ec.europa.eu/information_society/policy/esignature/trusted-list/tl-mp.xml`
- EU-level PKI

Qualified Electronic Signature Creation Device (QSCD)

ANNEX II

1. QSCDs shall ensure, by appropriate technical and procedural means, that at least:

(a) the confidentiality of the electronic signature creation data used for electronic signature creation is reasonably assured;

(b) the electronic signature creation data used for electronic signature creation can practically occur only once;

(c) the electronic signature creation data used for electronic signature creation cannot, with reasonable assurance, be derived and the electronic signature is reliably protected against forgery using currently available technology;

(d) the electronic signature creation data used for electronic signature creation can be reliably protected by the legitimate signatory against use by others.

Security certification according to Common Criteria EAL4+

- EN 419 211 - Protection profiles for secure signature creation device[2]

[2] https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=uriserv:OJ.L_.2016.109.01.0040.01.ENG

Validation of a Qualified Electronic Signature

Article 32

1. The process for the validation of a QES shall confirm the validity of a QES provided that:

(a) the certificate that supports the signature was, at the time of signing, a qualified certificate for electronic signature complying with Annex I;

(b) the qualified certificate was issued by a qualified trust service provider and was valid at the time of signing;

(c) the signature validation data corresponds to the data provided to the relying party;

(d) the unique set of data representing the signatory in the certificate is correctly provided to the relying party;

(e) the use of any pseudonym is clearly indicated to the relying party if a pseudonym was used at the time of signing;

(f) the electronic signature was created by a qualified electronic signature creation device;

(g) the integrity of the signed data has not been compromised;

(h) the requirements provided for in Article 26 were met at the time of signing.

- How can we establish whether the certificate was valid at the time of signing?

Trusted Timestamping

Article 3:
(33) 'electronic time stamp' means data in electronic form which binds other data in electronic form to a particular time establishing evidence that the latter data existed at that time;

`http://tools.ietf.org/html/rfc3161`

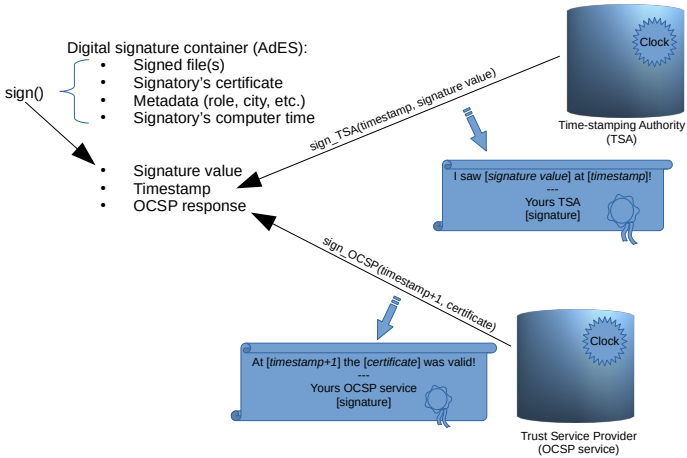
Signed statement of Time Stamping Authority (TSA):

```
> This data [data] was presented to me at this time: [time]
> --
> TSA [signature]
```

- TSA must use accurate time source
- TSA must log issued timestamps

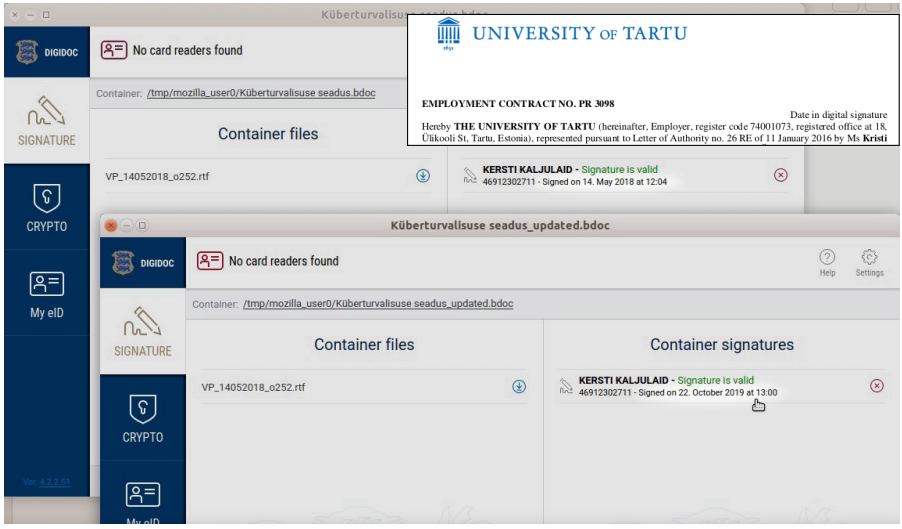
Article 41:
A qualified electronic time stamp shall enjoy the presumption of the accuracy of the date and the time it indicates and the integrity of the data to which the date and time are bound.

Digital signature container



- OCSP request must be made *after* the timestamp (`https://www.youtube.com/watch?v=eYG17IGOCi0`)
- Unable to verify whether the signature was made while the certificate was not suspened
- Long-term validity?

Time of signing



- Modifying the time of signing (`https://www.youtube.com/watch?v=ysYouhl1Yx4`)

Digital signature file formats

Advanced electronic signature (AdES) file format specifications to be recognised by public sector bodies[3]:

- XML – XAdES Baseline Profile (ETSI TS 103171)
  - Used in Estonia (.asice/.bdoc/.ddoc formats)
- CMS – CAdES Baseline Profile (ETSI TS 103173)
- PDF – PAdES Baseline Profile (ETSI TS 103172)

Associated Signature Container Extended (ASiC-E) Baseline Profile (ETSI TS 103174)

```
asic-container.asice: Zip ("application/vnd.etsi.asic-e+zip")
   + mimetype
   + document.docx
   + META-INF/manifest.xml
   + META-INF/signatures0.xml
   + META-INF/signatures1.xml
```

[3] https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=OJ:JOL_2015_235_R_0006

XML Signature (ASICE)

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<asic:XAdESSignatures xmlns:asic="http://uri.etsi.org/02918/v1.2.1#" [...]>
  <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#" Id="S1">
    <ds:SignedInfo Id="S1-SignedInfo">
      <ds:SignatureMethod Algorithm="http://www.w3.org/2001/04/xmldsig-more#rsa-sha256"/>
      <ds:Reference Id="S1-ref-1" URI="document.docx">
        <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
        <ds:DigestValue>SJO7h/iCeb9jDLXMZ6qEx8nYkhNR+MWBLge6YfyU7+U=</ds:DigestValue>
      </ds:Reference>
      <ds:Reference Id="S1-ref-SignedProperties" Type="http://uri.etsi.org/01903#SignedProperties" URI="#S1-SignedProperties">
        <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
        <ds:DigestValue>qRlc2fxIYkqde3/1sHpZuk+eBKMZ7rIsgBZbYhigV5g=</ds:DigestValue>
      </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue>MtKIgLOB...3D62QA==</ds:SignatureValue>
    <ds:X509Certificate>MIIIyDCCB7...SVU=</ds:X509Certificate>
    <xades:SignedProperties Id="S1-SignedProperties">
        <xades:SigningTime>2018-06-05T15:01:11Z</xades:SigningTime>
        <xades:SigningCertificate>
          <ds:DigestMethod Algorithm="http://www.w3.org/2001/04/xmlenc#sha256"/>
          <ds:DigestValue>cuPIt8LpJIs+eFUgUwIrNsUikaH/NTezVgkRXixABBo=</ds:DigestValue>
        </xades:SigningCertificate>
        ...
        <xades:DataObjectFormat ObjectReference="#S1-ref-1">
          <xades:MimeType>application/octet-stream</xades:MimeType>
        </xades:DataObjectFormat>
        ...
    </xades:SignedProperties>
    <xades:SignatureTimeStamp>...
    <xades:CertificateValues>...
    <xades:OCSPValues>...
    ...
  </ds:Signature>
  ...
</asic:XAdESSignatures>
```

- signature of *<SignedInfo>*

XML Signature (BDOC and DDOC)

BDOC (2015 – today):

- OCSP response also serves as a timestamp
- Hash of signature included in OCSP nonce extension
- This hack is not recognized by eIDAS standards

DDOC (2002 – 2017):

- Single self-contained XML file
- Signed files base64-encoded in `<Datafile>` element
- Supports only SHA-1

```
<?xml version="1.0" encoding="UTF-8"?>
<SignedDoc format="DIGIDOC-XML" version="1.3" xmlns="http://www.sk.ee/DigiDoc/v1.3.0#">
  <DataFile Filename="document.doc" Id="D0">UEsDBBQABgA...AS1EAAAAA</DataFile>
  <Signature Id="S0">
    <SignedInfo>
      <SignatureMethod Algorithm="http://www.w3.org/2000/09/xmldsig#rsa-sha1"/>
      <Reference URI="#D0">
        <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
        <DigestValue>Q43ti5R/wgi8qDoHsygLFTXEOqU=</DigestValue>
      </Reference>
      <Reference URI="#S0-SignedProperties">
        <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
        <DigestValue>GOHmQqHCqMxULzfWSONIL2iOmIU=</DigestValue>
      </Reference>
    </SignedInfo>
    <SignatureValue Id="S0-SIG">kgsCQ6...M4rkcj8=</SignatureValue>
    <X509Certificate>IID4z....V8APa</X509Certificate>
    <SignedProperties Id="S0-SignedProperties">
      <SigningCertificate>
        <DigestMethod Algorithm="http://www.w3.org/2000/09/xmldsig#sha1"/>
```

Task: ASiC-E XAdES verifier

Implement a utility that verifies ASICE XAdES digital signatures:

```
$ ./asice_verify.py good.asice
[+] Signatory: PARŠOVS,ARNIS,38608050013
[+] Signed file: hello.txt
[+] Timestamped: 2021-04-01 18:00:11 +00:00
[+] Signature verification successful!

$ ./asice_verify.py forgery1.asice
[+] Signatory: PARŠOVS,ARNIS,38608050013
[+] Signed file: hello.txt
[-] a wrong certificate hash included under the signature!

$ ./asice_verify.py forgery(2|3|4|5|6|7|8).asice
[...]
```

- The error messages have to be meaningful
- Must support:
  - a single signed file and a single signature (`signatures0.xml`)
  - only the algorithms used in the provided testcase files
- Not required to verify certificates and signature on:
  - certificates, timestamps, OCSP responses

Task: ASiC-E XAdES verifier

- Hash is calculated on canonicalized XML elements
  - Hash of canonicalized <SignatureValue> is timestamped
- Code for parsing timestamp and OCSP response is in the template
- Use `zipfile` for reading ZIP container:

```
>>> import zipfile
>>> archive = zipfile.ZipFile(filename, 'r')
>>> xml = archive.read('META-INF/signatures0.xml')
```

- Use BeautifulSoup for XML traversal:

```
>>> from bs4 import BeautifulSoup
>>> x = BeautifulSoup(xml, features="xml")
>>> x.XAdESSignatures.KeyInfo.X509Data.X509Certificate.encode_contents()
>>> x.XAdESSignatures.Signature.SignedInfo.Reference['URI']
>>> x.XAdESSignatures.Signature.SignedInfo.find('Reference',
                       attrs={'URI':'#S0-SignedProperties'})
```

Questions

- What are the main requirements for a signature to have the QES status?
- What are the benefits of a QES compared to an electronic signature?
- Can the authenticity of a QES be contested?
- Can an unsigned e-mail be used as proof in court?
- How can a TSP became a QTSP?
- What is required for a product to be recognized as a QSCD?
- Why are MIME type and certificate included under the signature?
- How can we prove that the certificate was valid at the time of signing?
- Will it be possible to verify an ASICE signature after the TSA/OCSP certificates expire?