

EDTA, INBATEK & iText Conference Bangkok, July 27, 2017

© 2015, iText Group NV, iText Software Corp., iText Software BVBA



E Digital Signatures in PDF

- Basic concepts...
- ... applied to PDF
- Architectures: server-side vs. client-side
- Digital signatures and document workflow
- Long term validation
- How Blockchain made me love everything I hate about digital signatures

Basic Concepts...



- Hashing algorithms
 Encryption algorithms
 Certificate Authorities
- **Digital signatures**



Three goals

■ Integrity — we want assurance that the document hasn't been changed somewhere in the workflow.

- Authenticity we want assurance that the author of the document is who we think it is (and not somebody else).
- Non-repudiation we want assurance that the author can't deny his authorship.



Concept 1: Integrity check using hash





Concept 1: Hashing

Hashing algorithm

- ≡ a cryptographic hash function to turn an arbitrary block of data into a fixed-size bit string.
- **E** Available algorithms
 - **MD5:** Ron Rivest (broken)
 - **E SHA:**
 - SHA-1: NSA (broken! See <u>https://shattered.io/</u>)
 - SHA-2: NSA / NIST
 - SHA-3: Keccak (made in Belgium!)

ERIPEMD: KULeuven

Concept 2: Encryption

■Asymmetric key algorithms

Encryption



≡Digital signing





Concept 2: Some name dropping

Public Key Cryptography Standards

- **E PKCS#1: RSA Cryptography Standard (Rivest, Shamir, Adleman)**
- **PKCS#7**: Cryptographic Message Standard (CMS)
- ≡ PKCS#11: Cryptographic Token Interface
- ≡ PKCS#12: Personal Information Exchange Syntax Standard
- **EVALUATE:** PKCS#13: Elliptic Curve Cryptography Standard (ECDSA)
- **E** Federal Information Processing Standards (FIPS)
 - **E** DSA: Digital Signature Algorithm (DSA)
- **European Telecommunications Standards Institute (ETSI)**
 - **E** CMS Advanced Electronic Signatures (CAdES)



Concept 3: Certificate Authorities



Concept 3: example

Self-signed:

Signed by Adobe:

Signed by GlobalSign:

10



Concept 3: example

| | Certificate Viewer | |
|--|---|---------------|
| hello_token.pdf - Adobe Reader File Edit View Window Help | This dialog allows you to view the details of a certificate and its entire issuance chain. The details correspond to the selected entry. Show all certification paths found | Comment |
| Signed and all signatures are valid. | Adobe Root CA Summary Details Revocation Trust Policies Legal Notice GlobalSign Primary Sł GlobalSign SHA25 Adobe Root CA | gnature Panel |
| Image: Signatures Image: Signatures Image: Signature is valid: Document has not been modified since this signature was app Signature is valid: Document has not been modified since this signature was app Signature by the current user The signature includes an embedded timestamp. Image: Signature Details Reason: Test Location: Ghent Certificate Details Last Checked: 2012.08.15 15:27:53 + 02'00' Field: sig on page 1 Cit at train this unning. | Adobe Systems Incorporated Issued by: Adobe Root CA Adobe Systems Incorporated Valid from: 2003/01/09 01:37:23 +02'00' Valid to: 2023/01/09 02:07:23 +02'00' Intended usage: Sign CRL, Sign certificate (CA) | E |
| | Export The selected certificate path is valid. The path validation checks were done as of the secure (timestamp) time: 2012/08/15 14:37:27 +02:00' Validation Model: Shell OK | |



Concept 3: the green check mark

- ≡ PKCS#12: Personal Information Exchange Syntax Standard
 - public and private key are stored in a file
- PKCS#11: Cryptographic Token Interface
 - ≡ public and private key are stored on a device
 - \equiv In the context of PDF and the "green check mark":
 - Certified Document Services (CDS): Adobe's root certificate
 - Adobe Approved Trust List (AATL): Trusted root certificates (since Acrobat 9)

Concept 1 + Concept 2 + Concept 3

EProducer

- ■Provides data as-is: [A]
- ■Provides hash of data, encrypted using private key: [B]
- ■Provides public key

EConsumer

- ■Creates hash from data [A]: hash1
- ■Decrypts hash [B] using public key: hash2
- ■If (hash1 == hash2) document OK!

Goals met?

Integrity

Hashes are identical

= Authenticity

- \equiv Identity is stored in public key signed by CA
- A time-stamp can be added

Non-repudiation

■ If hash can be decrypted with public key, the document was signed with the corresponding private key



Differences between EU and US

\equiv In the US, we make a distinction

- Electronic signatures don't necessarily involve PKI
- Digital signatures when a PKI infrastructure is involved
- ■In Europe, we speak of electronic signatures
 - As a **synonym** for digital signatures
 - All laws and regulations take this wording
 - There's no sharp distinction between electronic and digital signatures (which leads to confusion)
- \equiv I always speak of digital signatures



... Applied to PDF

ISO 32000-1
 ETSI TS 102 778 (PAdES)
 ISO 32000-2

Tres

Standards

ISO

- **E** ISO-32000-1 (2008) based on PDF 1.7 (2006)
- ≡ ISO-32000-2 defines PDF 2.0 (2017)
- **ETSI:** TS 102 778 (2009 2010)
 - PAdES 1: Overview
 - PAdES 2: Basic CMS based (ISO-32000-1)
 - PAdES 3: Enhanced CAdES based (ISO-32000-2)
 - PAdES 4: LTV Long Term Validation
 - PAdES 5: XAdES based (XML content)
 - PAdES 6: Visual representation guidelines
- **ETSI:** TS 103 172 (2011 2013)
 - PAdES Baseline Profile



Signatures in PDF



- There are no bytes in the PDF that aren't covered, other than the PDF signature itself. (*)
- The digital signature isn't part of the ByteRange.
- The concept "to initial a document" doesn't exist; you sign the complete document at once, not on a page per page basis. (*)

Signature stored in the document



What's inside a signature?

ISO-32000-2:

At minimum the PKCS#7 object shall include the signer's X.509 signing certificate. This certificate shall be used to verify the signature value in **/Contents**.

Best practices ("should" also have):

- Full certificate chain
- Revocation information (CRL / OCSP)
- Timestamp

Certificate authority needed Timestamp authority needed %PDF-1.*x*

• • •

```
/ByteRange ...
```

```
/Contents<
```

DIGITAL SIGNATURE

- Signed Message Digest
- Certificate chain
- Revocation information
- Timestamp

>... %%EOF



Architectures





Server-side signing







Use cases server-side signing

- **E** Company signature
 - Invoices
 - Contracts
 - ≡...
- Signing services in the Cloud
 - ≡ E.g. Docusign
- **E** Security management responsibilities!



Client-side signing





Use cases client-side signing

- Desktop applications
 - Adobe Acrobat / Reader
- In a web context
 - The PDF software runs on the client, e.g. using Java Web Start
- Access to the token or smart card through
 - MSCAPI
 - **PKCS#11**

- 1 signature / second
- Custom smart card library
- Security
 - User has smart card and PIN or USB token and passphrase

Deferred signing



Use cases deferred signing

Signing on an iPad/Tablet

- App on the device has a low footprint
- Easy to integrate into a document management system
- Example: eaZySign (Zetes)
- = Disadvantage
 - At most 1 signature / second
 - You need to trust the server that the hash you receive is actually the hash of the document you want to sign.
- E ISAE 3000
 - the standard for assurance over non-financial information. ISAE3000 is issued by the International Federation of Accountants (IFAC). The standard consists of guidelines for the ethical behavior, quality management and performance of an ISAE3000 engagement. Generally ISAE3000 is applied for audits of internal control, sustainability and compliance with laws and regulations.



Digital signatures and workflow

- Author signatures
- **Recipient signatures**
- **■** Locking fields / documents



Serial signatures

A PDF document can be signed more than once, but parallel signatures aren't supported, only serial signatures: additional signatures sign all previous signatures.

28



Digital signatures: types

≡Certification (aka author) signature

- \equiv Only possible for the first revision
- ■Involves modification detection permissions:
 - No changes allowed
 - Form filling and signing allowed
 - Form filling, signing and commenting allowed
- ■Approval (aka recipient) signature
 - \equiv Workflow with subsequent signers
 - ■New in PDF 2.0: modification detection permissions







Other possible icons

■ Signer's identity is unknown

Document has been altered or corrupted



Signing time is from the clock on the signer's computer.

ITEXT

Certified by Alice

| 🔁 step1_signed_by_alice.pdf - Adobe Reader | | | |
|--|---|--|----|
| File Edit View Window Help | | | |
| Image: Image | | | nt |
| Certified by Alice Specimen, Unknown, certificate issued by Alice Specimen. Please fill out the following form. You cannot save data typed into this form. Please print your completed form if you would like a copy for your records. | | | ds |
| C | Signatures | | Â |
| Di | 8∃ ▼ Validate All | Written by Alice | |
| 9 | 🖻 🤶 Certified by Alice Specimen | Digitally signed by Alice Specimen Date: 2012.08.12 11:02:36 CEST | |
| \otimes | Only form fill-in, signing and page adding actions are allowed Valid certified Document: | For approval by Bob | = |
| <u>L9</u> | Document has not been modified since it was certified | | |
| | Signer's identity is valid | | |
| | Signing time is from the clock on the signer's computer. | For approval by Carol | |
| | | | |
| | Last Checked: 2012.08.12 11:03:57 +02'00' | | |
| | Field: sig1 on page 1 | | |
| | 🗆 📼 Unsigned Signature Fields | For approval by Dave | |
| | 🖚 sig2 on page 1 | | |
| | 🖚 sig3 on page 1 | | |
| | 🖚 sig4 on page 1 | | |
| | | | |
| | | | |
| | | | - |



Read, approved and signed by Bob

| 🔁 step | 3_signed_by_alice_and_bob.pdf - Adobe Reader | | |
|-------------|---|--|--------------------|
| File B | idit View Window Help | | × |
| 1 | 🧔 🖹 📇 🖂 📄 💿 🚺 / 1 💽 🖑 🔅 (| ● ● 63,9% ▼ □ □ □ | Tools Sign Comment |
| 9 | Certified by Alice Specimen, Unknown, certificate issued by Alice Specimen. Sign following form. You cannot save data typed into this form. Please print your completed form if you would like a copy for your records. | d and all signatures are valid. Please fill out the | Signature Panel |
| D | Signatures | | <u>^</u> |
| (D) | 8= ▼ Validate All | Written by Alice | |
| 9 | 🗄 🤶 Certified by Alice Specimen | Digitally signed by Alice Specimen Date: 2012.08.12 11:02:36 CEST | |
| | 🖻 🐳 Rev. 2: Signed by Bob Specimen | For approval by Bob | |
| ~ | Signature is valid: | Read and Approved by Bob | |
| <u>L9</u> 2 | Document has not been modified since this signature was applied | | |
| Ľ. | Signer's identity is valid | Digitally signed by Bob Specimen Date: 2012.08.12 11:02:36 CEST | |
| | Signing time is from the clock on the signer's computer. | For approval by Carol | |
| | | · · · · · · · · · · · · · · · · · · · | |
| | Last Checked: 2012.08.12 11:04:23 +02'00' | | |
| | Field: sig2 on page 1 | | |
| | Click to view this version | For approval by Dave | |
| | 🗆 ≕ Unsigned Signature Fields | | |
| | ≕ sig3 on page 1 | | |
| | ≕ sig4 on page 1 | | |
| | | | |
| | | | |
| | | | - |



Bob's signature invalidated by Chuck

| 🔁 step | signed_by_alice_and_bob_broken_by_chuck.pdf - Adobe Reader | | 23 |
|--------|---|--|-----|
| File E | dit View Window Help | | × |
| 1 | 🔁 🥪 🗃 🖶 🖂 💿 💿 1 / 1 🗽 🖑 🖾 👄 Ŧ 73,4% 🔹 🔚 🔛 🔗 🤝 🔽 Tools Sign Comme | | |
| Les . | At least one signature is invalid. Please fill out the following form. Please print your completed form if you would like a copy for your | You cannot save data typed into this form. 💋 Signature Panel 📑 Highlight Existing Field | lds |
| | Signatures 🔳 | | Â |
| m | 8∃ ▼ Validate All | Written by Alice | |
| | Certified by Alice Specimen Rev. 2: Signed by Bob Specimen Signature is invalid: There have been changes made to this document Signer's identity is valid Signing time is from the clock on the signer's com Signature Details Last Checked: 2012.08.12 12:25:06 + 02'00' Field: sig2 on page 1 <u>Click to view this version</u> Fields Filled In | Digitally signed by Alice Specimen Date: 2012.08.12 12:25:00 CEST For approval by Bob Changed by Chuck Digitally signed by Bob Specimen Date: 2012.08.12 12:25:01 CEST For approval by Carol | III |
| | Field approved_bob on page 1 ➡ Unsigned Signature Fields ➡ sig3 on page 1 ➡ sig4 on page 1 < | For approval by Dave | Ŧ |



Read, approved and signed by Carol

| 🔁 step | 5_signed_by_alice_bob_and_carol.pdf - Adobe Reader | _ | | 23 |
|----------------------------|--|----------------|--|-------|
| File Edit View Window Help | | | | |
| | $[] \bigcirc \bigcirc$ | | | |
| | Certified by Alice Specimen, Unknown, certificate issued by Alice Specimen. Signe after signing. Open Signature Panel to view the document change history. Please into this form. Please print your completed form if you would like a copy for your records. | d and fill out | all signatures are valid. Document was updated : the following form. You cannot save data typed 🧭 Signature Panel 🖼 Highlight Existing Fi | ields |
| | Signatures (| | | Â |
| D, | 8= ▼ Validate All | | Written by Alice | |
| \otimes | ? Certified by Alice Specimen * Gev. 2: Signed by Bob Specimen | L | Digitally signed by Alice Specimen Date: 2012.08.12 11:02:36 CEST | E |
| | Form Fields Filled In | | Read and Approved by Bob | |
| | Rev. 3: Signed by Carol Specimen | | Digitally signed by Bob Specimen Date: 2012.08.12 11:02:36 CEST | |
| | Signature is valid: Document has not been modified since this signature was applied | | For approval by Carol Read and Approved by Carol | |
| | Signer's identity is valid Signing time is from the clock on the signer's computer. | | Digitally signed by Carol Specimen Date: 2012.08.12 11:02:37 CEST | |
| | ⊞ Signature Details Last Checked: 2012.08.12 11:15:01 +02'00' | | For approval by Dave | |
| | Field: sig3 on page 1 | | | |
| | Click to view this version □ => Unsigned Signature Fields | | | |
| | ➡ sig4 on page 1 | | | - |

Read, approved and signed by Dave

| File Edit View Window Help Image: Second Structure Image: Second Structur | × Comment |
|--|--------------|
| Image: Signatures Image: Signatures <th>Comment</th> | Comment |
| Certified by Alice Specimen, Unknown, certificate issued by Alice Specimen. Signed and all signatures are valid. Document was updated after signing. Open Signature Panel to view the document change history. Signatures Signatures Written by Alice Digitally signed by Alice Specimen Digitally Signe | |
| Signatures Image: Signature state Image: Signature state Validate All Image: Signature state Validate All Image: Signature state Digitally signed by Alice Specimen Digitally signed by Alice Specimen Digitally signed by Alice Specimen | nature Panel |
| Validate All Written by Alice • | Â |
| Digitally signed by Alice Specimen | \neg |
| Rev. 2: Signed by Bob Specimen Form Fields Filled In Field approved_carol on page 1 Rev. 3: Signed by Carol Specimen Form Fields Filled In Field approved_dave on page 1 For Approved by Bob Specimen Form Fields Filled In Field approved_dave on page 1 Rev. 4: Signed by Dave Specimen Signature is valid: Document has not been modified since this signature was applied Signer's identity is valid Signature Details Last Checked: 2012.08.12 11:04:59 +02'00' Field: sig4 on page 1 | |



Signature and lock broken by Chuck

| 🔁 step | _6_signed_by_dave_broken_by_chuck.pdf - Adobe Reader | | | |
|----------------------------|---|--|------------|--|
| File Edit View Window Help | | | | |
| 1 | 🧔 🎅 📇 🖨 🖂 💿 💽 1 / 1 🍡 🤅 | 🖑 🔍 👄 🕀 73,4% 🗸 🔚 🔛 🔛 🔛 🔛 Tools Sign Co | omment | |
| 4 | At least one signature is invalid. | Signat | ture Panel | |
| Û | Signatures I | | Â | |
| n | 8= | Written by Alice | | |
| چ چ ل | Certified by Alice Specimen Gerified by Alice Specimen Form Fields Filled In Field approved_carol on page 1 Document Locked by sig4 | Digitally signed by Alice Specimen Date: 2012.08.12 12:25:00 CEST For approval by Bob Read and Approved by Bob Digitally signed by Bob Specimen Date: 2012.08.12 12:25:01 CEST For approval by Carol Changed by Chuck Digitally signed by Carol Specimen Date: 2012.08.12 12:25:01 CEST | E | |
| | | For approval by Dave | | |
| | | Read and Approved by Dave | | |
| | | Digitally signed by Dave Specimen Date: 2012.08.12 12:25:01 CEST | | |



Long-term validation



Revocation
 Timestamps

37 Digital signatures: how it's done in PDF

Contraction of the local data

Certificates expire



Certificates get revoked





CA: CRL and OCSP

| Certificate Viewer 23 | Certificate Viewer | Certificate Viewer |
|---|--|--|
| This dialog allows you to view the details of a certificate and its entire issuance chain. The details correspond to the selected entry. Show all certification paths found Cert Signing Authority <supp Bruno Lowagie bruno@low. Summary Details Revocation Trust Policies Legal Notice Details The selected certificate is either a trust anchor or is a certificate above the trust anchor in the certificate chain (see the Trust tab for details). No revocation checks are done for such certificates, they are inherently considered trustworthy. Signer Details. Problems encountered Check revocation</br></supp | This dialog allows you to view the details of a certificate and its entire issuance chain. The details correspond to the selected entry. Show all certification paths found Cert Signing Authority <supp <br="" bruno="" lowagie=""></supp> bruno@low With the selected certificate is valid Details The selected certificate is considered valid because it The selected certificate is considered valid because it The selected certificate is considered valid because it The cRL was signed by "CA Cert Signing Authority signer Details Signer Details Problems encountered Check revocation | It is dialog allows you to view the details of a certificate and its entire issuance chain. The details correspond to the selected entry. Show all certification paths found Cert Signing Authority <supp <br="" bruno="" lowagie=""></supp> (bruno@low) Bruno Lowagie <bruno@low< td=""> Image: the selected certificate is valid Details The selected certificate is considered valid because it has the revoked as verified using the Online Certificate is status Protocol (OCSP) response that was embedded in the Status Protocol (OCSP) response that was embedded in the Status Protocol (OCSP) response was signed by "ocsp.cacett.org" on 2012/10/25 for each or the selected certificate is used in the Status Protocol (OCSP) response was signed by "ocsp.cacett.org" on 2012/10/25 for each or the selected certificate is used in the Status Protocol (OCSP) response was signed by "ocsp.cacett.org" on 2012/10/25 for each or the selected certificate is used in the selected certificate is that was an in the selected certificate in the selected certificate is used in the selected certificate in the selected certificate is the selected certificate is considered valid because it has the selected certificate is considered valid while 2012/10/25 for each or the selected certificate is the selected certed certificate is the selected certificate</bruno@low<> |
| The selected certificate path is valid. | The selected certificate path is valid. | (i) I ne selected certificate path is valid. The path validation and revocation checks were done as of the secure (timestame) times |
| The path validation checks were done as of the secure (timestamp) time: 2012/0/23 16:54:06 = 0100° Validation Model: Shell | The path validation and revocation checks were done as of the signing time: 2012/01/22 0848/22 /0100' Validation Model: Shell | 2012/10/23 16:54:406 +0100 Validation Model: Shell |
| ОК | ОК | ОК |



How to survive revocation / expiration?





Timestamps

| Signature Properties | |
|---|---|
| Signature is VALID, signed by Bruno Lowagie < bruno@lowagie.com>. | |
| Summary Document Signer Date/Time Legal | |
| Signing Time: 2012/10/23 16:54:05 +01'00' | |
| The signature includes an embedded timestamp. Timestamp time: 2012/10/23 16:54:06 +01'00' | |
| Timestamp embedded in the signature | |
| Timestamps are signed just as documents are signed. For a timestamp signature to be valid you must have trusted the Timestamp Authority that signed the timestamp. Click Show Certificate to view details requiring verification of the timestamp signature. | |
| | The signature includes an embedded timestamp. Timestamp time: 2012/10/23 16:54:06 +01'00' |
| Timestamp Authority: SEIKO Timestamp Service. Advanced A Show Certificate | |
| Timestamps are created with specific policies that are defined by the Timestamp Authority. Amongst other things, a policy can indicate how reliable the time source is. The policy for this timestamp is represented by the identifier 1.3.6.1.41.955.1.101.101.1. To understand timestamp policies, you must contact the Timestamp Authority. | Timestamp embedded in the signature Timestamps are signed just as documents are signed. For a timestamp signature to be valid you must have trusted the Timestamp Authority that signed the timestamp. Click Show Certificate to view details regarding verification of the timestamp signature. |
| | Timestamp Authority: SEIKO Timestamp Service. Advanced A ¹ Show Certificate |
| | |
| Validate Signature 0 | llose |

42 Digital signatures: how it's done in PDF

- There's no CRL/OCSP/TS in the document?
- The certificate is about to expire in one of your documents?
- The hashing/encryption algorithm is about to be deprecated?

Document Security Store (DSS)

%PDF-1.x

/ByteRange ...
/Contents

DIGITAL SIGNATURE

- Signed Message Digest
- Certificate

>...

%%EOF

%PDF-1.x

• • •

/ByteRange ...

/Contents<

DIGITAL SIGNATURE

- Signed Message Digest
- Certificate

>...

%%EOF

DSS for DIGITAL SIGNATURE

• VRI, Certs, OCSPs, CRLs



Document-level timestamp

%PDF-1.*x*

/ByteRange ... /Contents<

DIGITAL SIGNATURE

- Signed Message Digest
- Certificate

>...

%%EOF

DSS for DIGITAL SIGNATURE

• VRI, Certs, OCSPs, CRLs

%PDF-1.*x*

•••

/ByteRange ...

/Contents<

DIGITAL SIGNATURE

- Signed Message Digest
- Certificate

>... %%EOF

DSS for DIGITAL SIGNATURE

• VRI, Certs, OCSPs, CRLs

DOCUMENT TIMESTAMP TS1

%PDF-1.*x*

/ByteRange ... /Contents<

DIGITAL SIGNATURE

- Signed Message Digest
- Certificate

```
>...
```

%%EOF

DSS for DIGITAL SIGNATURE

• VRI, Certs, OCSPs, CRLs

DOCUMENT TIMESTAMP TS1

Every signed document needs to be "kept alive"



```
%PDF-1.x
/ByteRange ...
/Contents<
DIGITAL SIGNATURE
  Signed Message Digest
 Certificate
>...
%%EOF
DSS for DIGITAL SIGNATURE
 VRI, Certs, OCSPs, CRLs
DOCUMENT TIMESTAMP TS1
DSS for TS1
DOCUMENT TIMESTAMP TS2
```



How Blockchain can help...





Chain of Blocks







Records are distributed



Challenge to add block to chain



Proof of work done!



Longest chain is valid chain



Blockchain and Web of Trust





Web of trust record



Web of Trust



Blockchain for documents

Document record

| Document ID: [<abcdef>, <abcdef>] Timestamp</abcdef></abcdef> | | |
|--|--------------------------|--|
| Signed Document hash | | |
| Certificate • Identity • Public ke | of signer ey | |
| Ē | Compressed property list | |
| | | |



File identifiers: mandatory in PDF 2.0

| ID | array | (Required in PDF 2.0 or if an Encrypt entry is present; optional otherwise; PDF 1.1) An array of two byte-strings constituting a file identifier (See 14.4, "File identifiers") for the file. The ID array shall (PDF 2.0) have a minimum length of 16 bytes. If there is an Encrypt entry, this array and the two byte-strings shall be direct objects and shall be unencrypted. |
|----|-------|---|
| | | NOTE 2 Because the ID entries are not encrypted it is possible to check the ID key to assure that the correct file is being accessed without decrypting the file. The restrictions that the string be a direct object and not be encrypted assure that this is possible. |
| | | NOTE 3 Although this entry is optional prior to PDF 2.0, its absence might prevent the file from functioning in some workflows that depend on files being uniquely identified. |
| | | NOTE 4 The values of the ID strings are used as input to the encryption algorithm. If these strings were indirect, or if the ID array were indirect, these strings would be encrypted when written. This would result in a circular condition for a PDF reader: the ID strings need be decrypted in order to use them to decrypt strings, including the ID strings themselves. The preceding restriction prevents this circular condition. |

14.4 File identifiers

File identifiers shall be defined by the **ID** entry in a PDF file's trailer dictionary (see 7.5.5, "File trailer"). The value of this entry shall be an array of two byte strings. The first byte string shall be a permanent identifier based on the contents of the file at the time it was originally created and shall not change when the file is updated. The second byte string shall be a changing identifier based on the file's contents at the time it was last updated (see 7.5.6, "Incremental updates"). When a file is first written, both identifiers shall be set to the same value. If the first identifier in the reference matches the first identifier in the reference dile's **ID** entry, and the last identifier in the reference dile has been found. If only the first identifier matches, a different version of the correct file has been found.

PDF writers should attempt to ensure the uniqueness of file identifiers. This may be achieved by computing them by means of a message digest algorithm such as MD5 (described in Internet RFC 1321, *The MD5 Message-Digest Algorithm*), using the following information:

Impossible to know if an ID pair is unique if you don't know which IDs are already in use.

The current time;

- A string representation of the file's location;
- The size of the file in bytes.



Check if a record already exists



 $T \equiv \times T$





1st attempt to offer a forged painting with a fake certificate fails because the certificate can't be found on the chain.





© 2015, iText Group NV, iText Software Corp., iText Software BVBA

T = X

Use case 2: Supply chain

Supply chain

Use case 3: Long-Term Validation

Castello di Amorosa, Calistoga (Napa Valley), CA

Renewing a signature

Thank you!

© 2015, iText Group NV, iText Software Corp., iText Software BVBA