INTERNATIONAL STANDARD ISO/IEC 9594-8:1998
TECHNICAL CORRIGENDUM 2

Published 2002-06-15

Information technology — Open Systems Interconnection — The Directory: Authentication framework

TECHNICAL CORRIGENDUM 2

Technologies de l'information — Interconnexion de systèmes ouverts (OSI) — L'annuaire: Cadre d'authentification

RECTIFICATIF TECHNIQUE 2

iTeh STANDARD PREVIEW
(standards.iteh.ai)

Technical Corrigendum 2 to International Standard ISO/IEC 9594-8:1998 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 6, *Telecommunications and information exchange between systems*.

_____

ICS 35.100.70

Ref. No. ISO/IEC 9594-8:1998/Cor.2:2002(E)

ISO/IEC 9594-8:1998/Cor 2:2002
https://standards.iteh.ai/catalog/standards/sist/5dcb356b-4e80-4c2b-b733-
9159ed4c43c5/iso-iec-9594-8-1998-cor-2-2002

INTERNATIONAL STANDARD
ITU-T RECOMMENDATION

Information technology – Open Systems Interconnection –
The Directory: Authentication framework

TECHNICAL CORRIGENDUM 2

NOTE – This Technical Corrigendum covers Draft Technical Corrigenda 8 and 9.

1)   Defect reports resolved by Draft Technical Corrigendum 8

(covering resolutions to defect reports 226, 227 and 240)

1.1)   This corrects the defects reported in defect report 226

In 11.2, delete the 2nd paragraph:

The production of a certificate … compromise unlikely.

1.2)   This corrects the defects reported in defect report 227

*In 12.2.2.1, add the following 2 sentences to the end of the paragraph that begins with "*Certification authorities shall assign…*":*

The **keyIdentifier** form can be used to select CA certificates during path construction. The **authorityCertIssuer**, **authoritySerialNumber** pair can only be used to provide preference to one certificate over others during path construction.

1.3)   This corrects the defects reported in defect report 240

The following corrections should be made to the 1997 edition **authenticationFramework** *module in Annex A:*

1)   *Add* **id-mr** *to the list of objects imported from* **UsefulDefinitions** *module in the* **authenticationFramework** *module.*

2)   *Add* **AttributeType**, **Attribute**, *and* **MATCHING-RULE** *to the set of objects imported into the* **authenticationFramework** *module from the* **InformationFramework** *module.*

3)   *Add* **GeneralNames** *to the set of objects imported into the* **authenticationFramework** *module from the* **CertificateExtensions** *module.*

4)   *Add the following definition to the* **authenticationFramework** *module because this is imported into other modules in the X.500 series of Recommendations, but had never been included in the 1997 text of this Recommendation:*

```
HASH {ToBeHashed}         ::=         SEQUENCE {
    algorithmIdentifier                 AlgorithmIdentifier,
    hashValue                           BIT STRING ( CONSTRAINED BY {
        -- must be the result of applying a hashing procedure to the DER-encoded octets --
        -- of a value of --ToBeHashed } ) }
```

5) *Add the following OID assignments in the* **authenticationFramework** *module:*

id-at-attributeCertificateRevocationList OBJECT IDENTIFIER   ::=   {id-at 59}

id-mr-attributeCertificateMatch   OBJECT IDENTIFIER   ::=   {id-mr 42}

6) Add **Time** *to the set of objects imported into the* **certificateExtensions** *module from the* **authenticationFramework** *module.*

7) *In the* **certificateExtensions** *module, and in the main text of 12.7.2 replace:*

CertPolicySet ::= SEQUENCE (1..MAX) OF CertPolicyId

with:

CertPolicySet ::= SEQUENCE SIZE (1..MAX) OF CertPolicyId

2)   Defect reports resolved by Draft Technical Corrigendum 9

(covering resolutions to defect reports 244, 256, 257 and 258)

2.1)   This corrects the defects reported in defect report 244

In clause 8:

In the paragraph that begins "The **extensions** *field allows addition of new …", add the following two sentences to the end of the paragraph:*

When a certificate-using implementation recognizes and is able to process an extension, then the certificate-using implementation shall process the extension regardless of the value of the criticality flag. Note that any extension that is flagged non-critical will cause inconsistent behaviour between certificate-using systems that will process the extension and certificate-using systems that do not recognize the extension will ignore it.

Add the following immediately after the paragraph that begins "If unknown elements appear within the extension …"*:*

A CA has three options with respect to an extension:

    i)   it can exclude the extension from the certificate;

    ii) it can include the extension and flag it non-critical;

    iii) it can include the extension and flag it critical.

A validation engine has two possible actions to take with respect to an extension:

    i)   it can ignore the extension and accept the certificate (all other things being equal);

    ii) it can process the extension and accept or reject the certificate depending on the content of the extension and the conditions under which processing is occuring (e.g. the current values of the path processing variables).

Some extensions can only be marked critical. In these cases, a validation engine that understands the extension processes it and acceptance/rejection of the certificate is dependent (at least in part) on the content of the extension. A validation engine that does not understand the extension rejects the certificate.

Some extensions can only be marked non-critical. In these cases, a validation engine that understands the extension processes it and acceptance/rejection of the certificate is dependent (at least in part) on the content of the extension. A validation engine that does not understand the extension accepts the certificate (unless factors other than this extension cause it to be rejected).

Some extensions can be marked critical or non-critical. In these cases, a validation engine that understands the extension processes it and acceptance/rejection of the certificate is dependent (at least in part) on the content of the extension, regardless of the criticality flag. A validation engine that does not understand the extension accepts the certificate if the extension is marked non-critical (unless factors other than this extension cause it to be rejected) and rejects the certificate if the extension is marked critical.

When a CA considers including an extension in a certificate, it does so with the expectation that its intent will be adhered to wherever possible. If it is necessary that the content of the extension be considered prior to any reliance on the certificate, a CA would flag the extension critical. This must be done with the realization that any validation engine that does not process the extension will reject the certificate (probably limiting the set of applications that can verify the certificate). The CA may mark certain extensions non-critical to achieve backward compatibility with validation applications that cannot process the extensions. Where the need for backward compatibility and interoperability with validation applications incapable of processing the extensions is more vital than the ability of the CA to enforce the extensions, then these optionally critical extensions would be marked non-critical. It is most likely that CAs would set optionally critical extensions as non-critical during a transition period while the verifiers' certificate processing applications are upgraded to ones that can process the extensions.

In clause 12.1:

In the paragraph that begins "In a certificate or CRL, an extension is flagged ...", *add the following immediately after the third sentence that ends with* "... ignoring the extension":

If an extension is flagged non-critical, a certificate-using system that does recognize the extension, shall process the extension.

In clause 12.2.2.3:

In the paragraph that begins "If the extension is flagged non-critical ...", *replace the second sentence with the following:*

If this extension is present, and the certificate-using system recognizes and processes the **keyUsage** extension type, then the certificate using system shall ensure that the certificate shall be used only for a purpose for which the corresponding key usage bit is set to one.

In clause 12.2.2.4:

In the paragraph that begins "If the extension is flagged non-critical ...", *replace the second sentence with the following:*

If this extension is present, and the certificate-using system recognizes and processes the **extendedKeyUsage** extension type, then the certificate using system shall ensure that the certificate shall be used only for one of the purposes indicated.

In clause 12.4.2.1:

In the 4th paragraph following the ASN.1, replace: "If this extension is present and is flagged critical then:" *with the following:*

If this extension is present and is flagged critical, or is flagged non-critical but is recognized by the certificate-using system, then:

In clause 12.4.2.2:

Replace the last sentence "If this extension is present and is flagged critical ..." *with the following:*

If this extension is present and is flagged critical, or is flagged non-critical but is recognized by the certificate-using system, then the certificate-using system shall check that the certification path being processed is consistent with the value in this extension.

2.2) This corrects the defects reported in defect report 256

In clause 8:

In the first paragraph of the description of the cross certificate pair attribute (that begins "The **forward** elements …"*), add the following as a new 3rd sentence:*

If a CA issues a certificate to another CA, and the subject CA is not a subordinate to the issuer CA in a hierarchy, then the issuer CA must place that certificate in the **reverse** element of the **crossCertificatePair** attribute of its own directory entry.

2.3) This corrects the defects reported in defect report 257

In clause 8 in the ASN.1 construct **CertificatePair***:*

Replace **forward** *with* **issuedByThisCA** *and*
Replace **reverse** *with* **issuedToThisCA** *and make changes to the associated text as outlined below.*

In the descriptive text, throughout ITU-T X.509, update the text accordingly to reflect these new terms. This includes the following specific clauses:

– *general descriptive text in clause 8;*
– *ASN.1 and descriptive text for the cross certificate pair attribute in clause 8;*
– *ASN.1 and descriptive text for the associated matching rules in clauses 12.7.3 and 12.7.4, and*
– *the duplicate ASN.1 constructs in Annex A.*

Also, add the following text to the end of the paragraph that begins "The **forward** elements ..."*:*

The term **forward** was used in previous editions for **issuedByThisCA**, and the term **reverse** was used in previous editions for **issuedToThisCA**.

2.4) This corrects the defects reported in defect report 258

In clause 8, add the following as a new paragraph at the end of the clause, immediately before the first subclause 8.1:

Each certificate in a certification path shall be unique. No certificate may appear more than once in a value of **theCACertificates** component of **CertificationPath** or in a value of **certificate** in the **CrossCertificates** component of **ForwardCertificationPath**.

In clause 12.4.3 add the following Note immediately after bullet a), "a set of certificates":

NOTE – Each certificate in a certification path is unique. A path that contains the same certificate two or more times is not a valid certification path.