Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Eidgenössisches Finanzdepartement EFD

Bundesamt für Informatik und Telekommunikation BIT
Büroautomation und Support

Martin Egger

30. Januar 2009

Zertifizierungsrichtlinie (Certificate Policy (CP)) für S/MIME

AdminPKI KlasseC-Enterprise

Projektname: **PKI-KlasseC-Enterprise**

Projektnummer:

Version: **V1.2**

Status	in Arbeit	in Prüfung	genehmigt zur Nutzung
	☐	☐	☒

Beteiligter Personenkreis	
Autoren:	Martin Egger
Prüfung:	Peter Brügger, Johann Wiss
Genehmigung:	FFB
Benützer/Anwender:	Benutzer der Klasse C Enterprise PKI
zur Information/Kenntnis:	

Änderungskontrolle, Prüfung, Genehmigung			
Wann	**Version**	**Wer**	**Beschreibung**
19.04.2006	X0.1	Johann Wiss	Neues Dokument
17.05.2006	X0.3	Johann Wiss	Ergänzungen
19.05.2006	X0.4	Martin Egger	Ergänzungen
30.05.2006	X0.5	Johann Wiss	Ergänzungen aus Review
29.06.2006	V1.0	Martin Egger	Freigabe durch FFB
02.04.2007	V1.1	Martin Egger	Neuformulierung Genehmigung, Neues Format der Zertifikatsbeschreibung
15.08.2007	V1.1.1	Martin Egger	Name der CA im Forest EVD angepasst
30.01.2009	V1.2	Martin Egger	Schlüssellänge angepasst, Freigabe durch FFB

Inhaltsverzeichnis

Abkürzungsverzeichnis

BIT	Bundesamt für Informatik und Telekommunikation
CPS	Certification Practice Statement, Ausführungsbestimmungen der Zertifizierungsrichtlinien
CP	Certificate Policy, Zertifizierungsrichtlinien
FFB	Gremium Frühung Forest Bund im Auftrag des IRB
IRB	Informatikrat Bund

1 Einführung

Das vorliegende Dokument stellt die Zertifizierungsrichtlinie (CP) für S/MIME der AdminPKI KlasseC-Enterprise des BIT dar. Es spezifiziert das Format der Zertifikate für S/MIME-Verschlüsselung und -Signatur.

Diese Zertifikate dienen einerseits für die Verifikation und Integrität (Signatur) und andererseits zur Generierung von Chiffrierschlüsseln und deren Übermittlung (Chiffrierung) für E-Mail.

Da beide Zertifikate für den Einsatz im Produkt SecureMessaging (Mail-Client) erstellt wurden und sich die Zertifikate im Wesentlichen nur in der Key Usage unterscheiden, wurde auf separate CPs verzichtet.

Wie diese Zertifikate erstellt und ausgegeben werden, wird in den Ausführungsbestimmungen der Zertifizierungsrichtlinien (CPS) für die AdminPKI KlasseC-Enterprise beschrieben.

2 Verwaltung der Zertifizierungsrichtlinie

2.1 Organisation der Dokumentenverwaltung

Die vorliegende Zertifizierungsrichtlinie (CP) wird durch den AdminPKI KlasseC-Enterprise Serviceverantwortlichen verwaltet. Unter der Adresse http://www.pki.admin.ch wird die gültige Version des vorliegenden CP publiziert.

2.2 Kontaktperson

Die vorliegende CP steht unter der Verantwortung des AdminPKI KlasseC-Enterprise Serviceverantwortlichen:

AdminPKI KlasseC-Enterprise Serviceverantwortlicher
Bundesamt für Informatik und Telekommunikation
Monbijoustrasse 74
CH-3003 Bern

2.3 Genehmigungsverfahren

Die vorliegende CP wird vom FFB genehmigt.

Der AdminPKI KlasseC-Enterprise Serviceverantwortliche kann typographische Anpassungen oder Neuformulierungen von Abschnitten ohne inhaltliche Änderungen an der vorliegenden CP vornehmen und publizieren. Der FFB wird nachträglich darüber informiert, Einsprachen sind dann möglich.

Grössere Änderungen oder neue Dokumentversionen sind in jedem Fall durch den FFB genehmigungspflichtig.

3 Zertifikat

3.1 Überblick

Die Schlüssel und die Zertifikate, die im Rahmen dieser Zertifizierungsrichtlinie ausgegeben werden, dienen der Verschlüsselung und Signatur von E-Mails.

Die Kunden der AdminPKI KlasseC-Enterprise sind interne und externe Mitarbeiter der Bundesverwaltung, die über einen Windows Active Directory Domain Account verfügen. Der Antragsteller bzw. der Zertifikatsinhaber wird nicht auf Grund von speziellen Prozessen identifiziert.

Die Schlüssel für die Verschlüsselung werden in einem Key Archive gespeichert. Zur Zeit wird aber kein Dienst für das Key Recovery angeboten. Die Schlüssel für die Signatur werden nicht in einem Key Archive gespeichert.

3.2 Zertifikatsformat

Feld	Wert	Bemerkungen	Syntax [1],[2],[3]
Version	3	Bezeichnet X.509 v3 Zertifikate	This field describes the version of the encoded certificate. When extensions are used, as expected in this profile, use X.509 version 3 (value is 2). If no extensions are present, but a UniqueIdentifier is present, use version 2 (value is 1). If only basic fields are present, use version 1 (the value is omitted from the certificate as the default value).

Serial Number	*"serial number"*	Für die AdminPKI-KlasseC-Enterprise eineindeutige Nummer	The serial number is an integer assigned by the CA to each certificate. It MUST be unique for each certificate issued by a given CA (i.e., the issuer name and serial number identify a unique certificate).
Signature Algorithm	Algorithm ObjectId: 1.2.840.113549.1.1.5 sha1RSA		This field contains the algorithm identifier for the algorithm used by the CA to sign the certificate. This field MUST contain the same algorithm identifier as the signatureAlgorithm field in the sequence Certificate (see sec. 4.1.1.2). The contents of the optional parameters field will vary according to the algorithm identified. Section 7.2 lists the supported signature algorithms
Signature	*"Unique signature value"*		N/A
Issuer	CN=Admin-CE-Intra01 DC=intra DC=admin DC=ch	Distinguished Name der IssuingCA: Admin-CE-Intra01, Admin-CE-EDA01, EVDAD-CA01	Name ::= CHOICE { RDNSequence } RDNSequence ::= SEQUENCE OF RelativeDistinguishedName RelativeDistinguishedName ::= SET OF AttributeTypeAndValue AttributeTypeAndValue ::= SEQUENCE { type AttributeType, value AttributeValue } AttributeType ::= OBJECT IDENTIFIER AttributeValue ::= ANY DEFINED BY AttributeType DirectoryString ::= CHOICE { teletexString TeletexString (SIZE (1..MAX)), printableString PrintableString (SIZE (1..MAX)), universalString UniversalString (SIZE (1..MAX)), utf8String UTF8String (SIZE (1..MAX)), bmpString BMPString (SIZE (1..MAX)) }

Validity - NotBefore	*"date, time"*		The certificate validity period is the time interval during which the CA warrants that it will maintain information about the status of the certificate. The field is represented as a SEQUENCE of two dates: the date on which the certificate validity period begins (not-Before) and the date on which the certificate validity period ends (notAfter). Both notBe-fore and notAfter may be encoded as UTCTime or GeneralizedTime. CAs conforming to this profile MUST always encode certificate validity dates through the year 2049 as UTCTime; certificate validity dates in 2050 or later MUST be encoded as GeneralizedTime.
Validity - NotAfter	*"date, time"*	Die Gültigkeit beträgt **2** Jahre	see Valid from.

Subject	"common name"	Die Email-Adresse erscheint im *alternative subject name* und nicht im *subject*. Beispiel: CN=Hans Muster U80712345	The subject field identifies the entity associated with the public key stored in the subject public key field. The subject name may be carried in the subject field and/or the subjectAltName extension. If the subject is a CA (e.g., the basic constraints extension, as discussed in 4.2.1.10, is present and the value of cA is TRUE,) then the subject field MUST be populated with a non-empty distinguished name matching the contents of the issuer field (see sec. 4.1.2.4) in all certificates issued by the subject CA. If subject naming information is present only in the subjectAltName extension (e.g., a key bound only to an email address or URI), then the subject name MUST be an empty sequence and the subjectAltName extension MUST be critical. Where it is non-empty, the subject field MUST contain an X.500 distinguished name (DN). The DN MUST be unique for each subject entity certified by the one CA as defined by the issuer name field. A CA may issue more than one certificate with the same DN to the same subject entity.
Public Key Algorithm	Algorithm ObjectId: 1.2.840.113549.1.1.1 RSA		This field is used to carry the public key and identify the algorithm with which the key is used. The algorithm is identified using the AlgorithmIdentifier structure specified in section 4.1.1.2. The object identifiers for the supported algorithms and the methods for encoding the public key materials (public key and parameters) are specified in section 7.3.
Public Key Length	2048 bits		N/A
Public Key	*"Unique key value"*		N/A
Certificate Extensions			

Key Usage	non critical	Digital Signature ***or*** Key Encipherment	Für das Verschlüsselungszertifikat wird *keyEncipherment* und für das Signaturzertifikat wird *digitalSignature* verwendet.	When used, this extension SHOULD be marked critical. id-ce-keyUsage OBJECT IDENTIFIER ::= { id-ce 15 } KeyUsage ::= BIT STRING {digitalSignature (0), nonRepudiation (1), keyEncipherment (2), dataEncipherment (3), keyAgreement (4), keyCertSign (5), cRLSign (6), encipherOnly (7), decipherOnly (8) }
Subject Key Identifier	non critical	*"Unique ID"*	Wird von der CA beim Erstellungsprozess generiert	This extension MUST NOT be marked critical. id-ce-subjectKeyIdentifier OBJECT IDENTIFIER ::= { id-ce 14 } SubjectKeyIdentifier ::= KeyIdentifier
Certificate Template Information	non critical	Template= BVerw-SMIME Encryption *("OID")* ***or*** Template= BVerw-SMIME Signing *("OID")* Major Version number="*number*" Minor Version number="*number*"	Microsoft spezifische Erweiterung	N/A
Authority Key Identifier	non critical	KeyID=*"Public Key ID"*		This extension MUST NOT be marked critical. id-ce-authorityKeyIdentifier OBJECT IDENTIFIER ::= { id-ce 35 } AuthorityKeyIdentifier ::= SEQUENCE { keyIdentifier [0] KeyIdentifier OPTIONAL, authorityCertIssuer [1] GeneralNames OPTIONAL, authorityCertSerialNumber [2] CertificateSerialNumber OPTIONAL } KeyIdentifier ::= OCTET STRING

CRL Distribution Points	non critical	Distribution Point Name: Full Name: URL=http://www.pki.admin.ch/crl/Admin-CE-Intra01.crl URL=ldap:///CN=Admin-CE-Intra01,CN=ADSROOTECA01,CN=CDP,CN=Publ ic%20Key%20Services,CN=Services,CN=Configuration,DC=intra,DC=admin,DC=ch?certificateRevocationList?base? objectClass=cRLDistributionPoint	CRL Lokation der IssuingCA: Admin-CE-Intra01, Admin-CE-EDA01, EVDAD-CA01	The CRL MUST be issued by the CA that issued the certificate. id-ce-cRLDistributionPoints OBJECT IDENTIFIER ::= { id-ce 31 } cRLDistributionPoints ::= { CRLDistPointsSyntax } CRLDistPointsSyntax ::= SEQUENCE SIZE (1..MAX) OF DistributionPoint DistributionPoint ::= SEQUENCE { distributionPoint [0] DistributionPointName OPTIONAL, reasons [1] ReasonFlags OPTIONAL, cRLIssuer [2] GeneralNames OPTIONAL } DistributionPointName ::= CHOICE { fullName [0] GeneralNames, nameRelativeToCRLIssuer [1] RelativeDistinguishedName } ReasonFlags ::= BIT STRING { unused (0), keyCompromise (1), cACompromise (2), affiliationChanged (3), superseded (4), cessationOfOperation (5), certificateHold (6) }
Authority Information Access	non critical	Access Method=Certification Authority Issuer (1.3.6.1.5.5.7.48.2) Alternative Name: URL=http://www.pki.admin.ch/aia/Admin-CE-Intra01.crt	CA Zertifikat der IssuingCA: Admin-CE-Intra01, Admin-CE-EDA01, EVDAD-CA01	This extension may be included in subject or CA certificates, and it MUST be non-critical. id-pe-authorityInfoAccess OBJECT IDENTIFIER ::= { id-pe 1 } AuthorityInfoAccessSyntax ::= SEQUENCE SIZE (1..MAX) OF AccessDescription AccessDescription ::= SEQUENCE { accessMethod OBJECT IDENTIFIER, accessLocation GeneralName } id-ad OBJECT IDENTIFIER ::= { id-pkix 48 } id-ad-caIssuers OBJECT IDENTIFIER ::= { id-ad 2 }
Enhanced Key Usage	non critical	Secure Email (1.3.6.1.5.5.7.3.4)		N/A

Certificate Policies	non critical	Policy Identifier=*AdminPKI KlasseC-Enterprise CPS* Policy Qualifier Info: Policy Qualifier Id=CPS Qualifier: *http://www.pki.admin.ch/policy/AdminPKI_KlasseC-Enterprise_CPS.pdf*	Microsoft spezifische Erweiterung	N/A
Application Policies	non critical	Policy Identifier=Secure Email	Microsoft spezifische Erweiterung	N/A
Subject Alternative Name	non critical	rfc822Name	Beispiel: RFC822 Name=hans.muster@bit.admin.ch	Applications with specific requirements may use such names but shall define the semantics. id-ce-subjectAltName OBJECT IDENTIFIER ::= { id-ce 17 } SubjectAltName ::= GeneralNames GeneralNames ::= SEQUENCE SIZE (1..MAX) OF GeneralName GeneralName ::= CHOICE { otherName [0] OtherName, rfc822Name [1] IA5String, dNSName [2] IA5String, x400Address [3] ORAddress, directoryName [4] Name, ediPartyName [5] EDIPartyName, uniformResourceIdentifier [6] IA5String, iPAddress [7] OCTET STRING, registeredID [8] OBJECT IDENTIFIER} OtherName ::= SEQUENCE { type-id OBJECT IDENTIFIER, value [0] EXPLICIT ANY DEFINED BY type-id } EDIPartyName ::= SEQUENCE { nameAssigner [0] DirectoryString OPTIONAL, partyName [1] DirectoryString }