# CONTROLLER OF CERTIFICATION AUTHORITIES (CCA)

Level 12, The Celicourt 6, Sir Celicourt Antelme Street Port Louis Mauritius
Tel.: (230) 211 5333/4 Fax: (230) 211 9444 email: info@cca.mu


## THE CCA DIRECTIVE 1 OF 2010

**Ref: ICTA/CCA/CCAD/1/2010** **10 December 2010**

The Information and Communication Technologies Authority in the exercise of its function as the Controller of Certification Authorities and in pursuance of Section 18 (1) (z) of the Information and Communication Technologies Act 2001 (as amended), Section 37 of the Electronic Transactions Act 2000 (as amended) and Regulation 3 (3) of the Electronic Transactions (Certification Authorities) Regulations 2010, hereby issues the following Directive.

### 1. Short title and Commencement

(i) This Directive shall be called "The CCA Directive 1 of 2010 - (CCAD 1 of 2010)"

(ii) The CCA Directive 1 of 2010 shall come into effect on 13 December 2010.

### 2. Scope and objective

This Directive sets out standards for all licensed, recognised and approved Certification Authorities for carrying out different activities associated with their functions.

Information & Communication Technologies Authority
Level 12
The Celicourt
6, Sir Celicourt Antelme St
Port Louis

**Mr. T. Dwarka**
**Chairman**

**Dr. M. K. Oolun**
**Executive Director**

**To: All Licensed, Recognised and Approved Certification Authorities.**

## DIRECTIVE – I

## Standards for Certification Authorities

**1. The standards followed by the Certification Authority for carrying out its functions: –**

(1) Every Certification Authority shall observe the following standards for carrying out different activities associated with its functions.

(a) **PKIX (Public Key Infrastructure*)***

Public Key Infrastructure as recommended by Internet Engineering Task Force (IETF) document draft-ietf-pkix-roadmap-09 for "Internet X.509 Public Key Infrastructure" (July , 2002);

(b) **Public-key cryptography based on the emerging Institute of Electrical and Electronics Engineers (IEEE) standard P1363 for three families*:***

Discrete Logarithm (DL) systems

Elliptic Curve Discrete Logarithm (EC) systems

Integer Factorization (IF) systems;

(c) **Public-key Cryptography Standards (PKCS)**

PKCS#1 RSA Encryption Standard (512, 1024, 2048 bit)

PKCS#3 Diffie-Hellman Key Agreement Standard

PKCS#5 Password Based Encryption Standard

PKCS#6 Extended-Certificate Syntax Standard

PKCS#7 Cryptographic Message Syntax standard

PKCS#8 Private Key Information Syntax standard

PKCS#9 Selected Attribute Types

PKCS#10 RSA Certification Request

PKCS#11 Cryptographic Token Interface Standard

PKCS#12 Portable format for storing/transporting a user's private keys and certificates

PKCS#13 Elliptic Curve Cryptography Standard

PKCS#15 Cryptographic Token Information Format Standard;

(d) **Federal Information Processing Standards (FIPS)**

FIPS 180-1, Secure Hash Standard

FIPS 186-1, Digital Signature Standard (DSS)

FIPS 140-1 level 3, Security Requirement for Cryptographic Modules;

(e) **Discrete Logarithm (DL) systems**

Diffie-Hellman, MQV key agreement

DSA, Nyberg-Rueppel signatures;

(f) **Elliptic Curve (EC) systems**

Elliptic curve analogs of DL systems;

(g) **Integer Factorization (IF) systems**

RSA encryption

RSA, Rabin-Williams signatures;

(h) **Key agreement schemes**

(i) **Signature schemes**

DL/EC scheme with message recovery

PSS, FDH, PKCS #1 encoding methods for IF family

PSS-R for message recovery in IF family;

(ii) **Encryption schemes**

Abdalla-Bellare-Rogaway DHAES for DL/EC family;

(i) **Form and size of the key pairs**

(1) The minimum key length for Asymmetric cryptosystem (RSA Algorithm) shall be 2048 for the Certification Authority's key pairs and 1024 for the key pairs used by subscribers.

(2) The Certification Authority's key pairs shall be changed every three to five years (except during exigencies as in the case of key compromise when the key shall be changed immediately). The Certification Authority shall take appropriate steps to ensure that key changeover procedures as mentioned in the approved Certificate Practice Statements are adhered to.

(3) The subscriber's key pairs shall be changed every one to two years;

(j) **Directory Services (LDAP ver 3)**

X.500 for publication of Public Key Certificates and Certificate Revocation Lists

X.509 version 3 Certificates as specified in ITU RFC 1422

X.509 version 2 Certificate Revocation Lists;

(k) **Publication of Public Key Certificate**.

The Certification Authority shall, on acceptance of a Public Key Certificate by a subscriber, publish it on its web site for access by the subscribers and relying parties. The Certification Authority shall be responsible and shall ensure the transmission of Public Key Certificates and Certificate Revocation Lists to the National Repository of the Controller, for access by subscribers and relying parties. The National Repository shall conform to X.500 Directory Services and provide for access through LDAP Ver 3. The Certification Authority shall be responsible for ensuring that Public Key Certificates and Certificate Revocation Lists integrate seamlessly with the National Repository on their transmission;

l) **Public Key Certificate Standard**

All Public Key Certificates issued by the Certification Authorities shall conform to International Telecommunication Union X.509 version 3 standard. X.509 v3 certificate basic syntax is as follows.

**tbsCertificate**

{

Version

Serial Number

Signature

Issuer

Validity

Subject

Subject Public Key Information

Issuer Unique ID [1] IMPLICIT Unique Identifier *optional*,

— If present, version shall be v2 or v3

Subject Unique ID [2] IMPLICIT Unique Identifier *optional*,

— If present, version shall be v2 or v3

Extensions [3] EXPLICIT Extensions *optional*

— If present, version shall be v3

{

Authority Key Identifier

{

Key Identifier *optional*,

Authority Certificate Issuer *optional*,

Authority Certificate Serial Number *optional*

}

Subject Key Identifier

Key Usage
{
  Digital Signature
  Non Repudiation
  Key Encipherment
  Data Encipherment
  Key Agreement
  Key Cert Sign
  cRLSign
  Encipher Only
  Decipher Only
}
Private Key Usage Period
{
  Not Before *optional*,
  Not After *optional*
}
Certificate Policies
{
  Policy Information
{
  Policy Identifier
  Policy Qualifiers *optional*
}
Certificate Policy Id
{
  Policy Qualifier Info
{
  Policy Qualifier Id
  Qualifier
{
    cPSuri
    User Notice
{
        Notice Reference *optional*
{

Organization
Notice Numbers
}
Display Text *optional*
{
visibleString
bmpString
utf8String
}
Policy Mappings
{
Issuer Domain Policy
Subject Domain Policy
}
Subject Alternative Name
{
General Name
{
Other Name
{
type-id
value
}
Rfc822Name
DNS Name
X400 Address
Directory Name
edi Party Name
{
Name Assigner *optional*,
Party Name
}
Uniform Resource Identifier
IP Address
Registered ID
}

}
Issuer Alternative Names
Subject Directory Attributes
Basic Constraints
{
  cA
  path Len Constraint *optional*
}
Name Constraints
{
  Permitted Subtrees *optional*
  Excluded Subtrees *optional*
}
Policy Constraints
{
  Require Explicit Policy *optional*
  Inhibit Policy Mapping *optional*
}
Extended key usage field
{
  Extended Key Usage Syntax
  Key Purpose Id
{
        Server Authentication
        Client Authentication
        Code Signing
        Email Protection
        Time Stamping
  }
}
CRL Distribution Points
{
  CRL Distribution Points Syntax
  Distribution Point
{
        Distribution Point *optional*

{

full Name

name Relative To CRL Issuer

}

}

Reasons *optional*

{

Unused

Key Compromise

CA Compromise

Affiliation Changed

Superseded

Cessation Of Operation

Certificate Hold

}

cRL Issuer *optional*

}

Authority Information Access

{

Authority Information Access Syntax

Access Description

{

Access Method

Access Location

}

}

**Signature Algorithm**

**Signature Value**

}

**(i) Certificate**

TBSCertificate is certificate “to be signed”. The field contains the names of the subject and issuer, a public key associated with the subject, a validity period, and other associated information. The fields are described in detail.

**(ii) Version**

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 Unique Identifier 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).

**(iii) Signature**

This field contains the algorithm identifier for the algorithm used by the Certification Authority to sign the certificate.

**(iv) Issuer**

The issuer field identifies the entity who has signed and issued the certificate. The issuer field shall contain a non-empty distinguished name.

**(v) Validity**

The certificate validity period is the time interval during which the Certification Authority warrants that it will maintain information about the status of the certificate.

**(vi) Subject**

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 Certification Authority (e.g., the basic constraints extension, is present and the value of cA is TRUE,) then the subject field shall be populated with a non-empty distinguished name matching the contents of the issuer field in all certificates issued by the subject Certification Authority.

**(vii) Subject Public Key Information**

This field is used to carry the public key and identify the algorithm with which the key is used.

**(viii) Unique Identifiers**

These fields may only appear if the version is 2 or 3. The subject and issuer unique identifiers are present in the certificate to handle the possibility of reuse of subject and/or issuer names over time.

**(ix) Extensions**

This field may only appear if the version is 3. The extensions defined for X.509 v3 certificates provide methods for associating additional attributes with users or public keys and for managing the certification hierarchy. The X.509 v3 certificate format also allows communities to define private extensions to carry information unique to those communities. If present, this field is a sequence of one or more certificate extensions. The content of certificate extensions in the Internet Public Key Infrastructure is defined as follows, namely:-.

(a) *Authority Key Identifier*

*The authority key identifier extension provides a means of identifying the public key corresponding to the private key used to sign a certificate. This extension is used where an issuer has multiple signing keys (either due to multiple concurrent key pairs or due to changeover). The identification may be based on either the key identifier (the subject key identifier in the issuer's certificate) or on the issuer name and serial number.*

(b) *Subject Key Identifier*

*The subject key identifier extension provides a means of identifying certificates that contain a particular public key.*

(c) *Key Usage*

*The key usage extension defines the purpose (e.g., encipherment, signature, certificate signing) of the key contained in the certificate. The usage restriction might be employed when a key that could be used for more than one operation is to be restricted. For example, when an RSA key should be used only for signing, the digital Signature and/or non-Repudiation bits would be asserted. Likewise, when an RSA key should be used only for key management, the key Encipherment bit would be asserted.*

(d) *Private Key Usage Period*

*The private key usage period extension allows the certificate issuer to specify a different validity period for the private key than the certificate. This extension is intended for use with digital signature keys. This extension consists of two optional components, not Before and not After. (This profile recommends against the use of this extension. Certification Authorities conforming to this profile MUST NOT generate certificates with critical private key usage period extensions.)*

(e) *Certificate Policies*

*The certificate policies extension contains a sequence of one or more policy information terms, each of which consists of an object identifier and optional qualifiers. These policy information terms indicate the policy under which the certificate has been issued and the purposes for which the certificate may be used. Optional qualifiers, which may be present, are not expected to change the definition of the policy.*

(f) *Policy Mappings*

*This extension is used in Certification Authority certificates. It lists one or more pairs of object identifiers; each pair includes an issuer Domain Policy and a subject Domain Policy. The pairing indicates the issuing Certification Authority considers its issuer Domain Policy equivalent to the subject Certification Authority's subject Domain Policy.*

(g) *Subject Alternative Name*

*The subject alternative names extension allows additional identities to be bound to the subject of the certificate. Defined options include an Internet electronic mail address, a Directory Naming Service name, an IP address, and a uniform resource identifier (URI).*

(h) *Issuer Alternative Names*

*This extension is used to associate Internet style identities with the certificate issuer.*

(i) *Subject Directory Attributes*

*The subject directory attributes extension is not recommended as an essential part of this profile, but it may be used in local environments.*

(j) *Basic Constraints*

*The basic constraints extension identifies whether the subject of the certificate is a Certification Authority and how deep a certification path may exist through that Certification Authority.*

(k) *Name Constraints*

*The name constraints extension, which MUST be used only in a Certification Authority certificate, indicates a name space within which all subject names in subsequent certificates in a certification path shall be located. Restrictions may apply to the subject distinguished name or subject alternative names. Restrictions apply only when the specified name form is present. If no name of the type is in the certificate, the certificate is acceptable.*

(l) *Policy Constraints*

*The policy constraints extension can be used in certificates issued to Certification Authorities. The policy constraints extension constrains path validation in two ways. It can be used to prohibit policy mapping or require that each certificate in a path contain an acceptable policy identifier.*

(m) *Extended key usage field*

*This field indicates one or more purposes for which the certified public key may be used, in addition to or in place of the basic purposes indicated in the key usage extension field.*

(n) *CRL Distribution Points*

*The CRL distribution points extension identifies how CRL information is obtained.*

(o) *Private Internet Extensions*

*This extension may be used to direct applications to identify an on-line validation service supporting the issuing Certification Authority.*

(p) *Authority Information Access*

*The authority information access extension indicates how to access Certification Authority information and services for the issuer of the certificate in which the extension appears. Information and services may include on-line validation services and Certification Authority policy data.*

**(x) Signature Algorithm**

The Signature Algorithm field contains the identifier for the cryptographic algorithm used by the Certification Authority to sign this certificate. The algorithm identifier is used to identify a cryptographic algorithm.

**(xi) Signature Value**

The Signature Value field contains a digital signature computed upon the Abstract Syntax Notation (ASN.1) DER encoded tbsCertificate. The ASN.1 DER encoded tbsCertificate is used as the input to the signature function. This signature value is then ASN.1 encoded as a BIT STRING and included in the Certificate's signature field.

**(xii) Certificate Revocation List Standard –**

CRL and CRL Extensions Profile - The CRL contents as per International Telecommunications Union standard ver 2 are as follows

**CertificateList**

{

**TBSCertList**

{

Version

Signature

Issuer

This Update

Next Update

Revoked Certificates

{

User Certificate

Revocation Date

Certificate Revocation List Entry Extensions

{

Reason Code

{

Unspecified

Key Compromise

CA Compromise

Affiliation Changed

Superseded

Cessation Of Operation

Certificate Hold

Remove From Certificate Revocation List

}
Hold Instruction Code
Invalidity Date
Certificate Issuer
} *optional*
Certificate Revocation List Extensions
{
Authority Key Identifier
Issuer Alternative Name
Certificate Revocation List Number
Delta Certificate Revocation List Indicator
Issuing Distribution Point
{
Distribution Point
Only Contains User Certs
Only Contains CA Certs
Only Some Reasons
Indirect Certificate Revocation List
}
} *optional*
**Signature Algorithm**
**Signature Value**
}

**(i) tbsCertList**

The certificate list to be signed, or TBSCertList, is a sequence of required and optional fields. The required fields identify the Certificate Revocation List issuer, the algorithm used to sign the Certificate Revocation List, the date and time the Certificate Revocation List was issued, and the date and time by which the Certification Authority will issue the next Certificate Revocation List.

Optional fields include lists of revoked certificates and Certificate Revocation List extensions. The Revoked Certificate List is optional to support the case where a Certification Authority has not revoked any unexpired certificates that it has issued. The profile requires conforming Certification Authorities to use the Certificate Revocation List extension cRLNumber in all Certificate Revocation Lists issued. The first field in the sequence is the tbsCertList. This field is itself a sequence containing the name of the issuer, issue date, issue date of the next list, the list of revoked certificates, and optional Certificate Revocation List extensions.

Further, each entry on the revoked certificate list is defined by a sequence of user certificate serial number, revocation date, and optional Certificate Revocation List entry extensions. The fields are described in detail, as follows namely:-

**(ii) Version**

This optional field describes the version of the encoded Certificate Revocation List. When extensions are used, as required by this profile, this field MUST be present and MUST specify version 2 (the value is 1). If only basic fields are present, use version 1 (the value is omitted from the certificate as the default value)

**(iii) Serial number**

The serial number is an integer assigned by the Certification Authority to each certificate. It shall be unique for each certificate issued by a given Certification Authority (i.e, the issuer name and serial number identity a unique certificate)

**(iv) Signature**

This field contains the algorithm identifier for the algorithm used to sign the Certificate Revocation List. This field shall contain the same algorithm identifier as the signature Algorithm field in the sequence Certificate List.

**(v) Issuer Name**

The issuer name identifies the entity who has signed and issued the Certificate Revocation List. The issuer identity is carried in the issuer name field. Alternative name forms may also appear in the issuer Alternate Name extension. The issuer name field MUST contain an X.500 distinguished name (DN). The issuer name field is defined as the X.501 type Name, and MUST follow the encoding rules for the issuer name field in the certificate.

**(vi) This Update**

This field indicates the issue date of this Certificate Revocation List. This Update may be encoded as UTC Time or Generalized Time. Certification Authorities conforming to this profile that issue Certificate Revocation Lists MUST encode This Update as UTCTime for dates through the year 2049. Certification Authorities conforming to this profile that issue Certificate Revocation Lists MUST encode This Update as Generalized Time for dates in the year 2050 or later.

**(vii) Next Update**

This field indicates the date by which the next Certificate Revocation List will be issued. The next Certificate Revocation List could be issued before the indicated date, but it will not be issued any later than the indicated date. Certification Authorities should issue Certificate Revocation Lists with a Next Update time equal to or later than all previous Certificate Revocation Lists. Next Update may be encoded as UTCTime or GeneralizedTime.

**(xiii) Revoked Certificates**

Revoked certificates are listed. The revoked certificates are named by their serial numbers. Certificates revoked by the Certification Authority are uniquely identified by the certificate serial number. The date on which the revocation occurred is specified. Additional information may be supplied in Certificate Revocation List entry extensions;

**(ix) CRL Entry Extensions**

The Certificate Revocation List entry extensions already defined by American National Standards Institute X9 and International Standards Organisation /IEC / International Telecommunication Union for X.509 v2 Certificate Revocation Lists provide methods for associating additional attributes with Certificate Revocation List entries [X.509] [X9.55]. The X.509 v2 Certificate Revocation List format also allows communities to define private Certificate Revocation List entry extensions to carry information unique to those communities. All Certificate Revocation List entry extensions used in this specification are non-critical.

(a) *Reason Code*

*The reason Code is a non-critical Certificate Revocation List entry extension that identifies the reason for the certificate revocation. Certification Authorities are strongly encouraged to include meaningful reason codes in Certificate Revocation List entries; however, the reason code Certificate Revocation List entry extension should be absent instead of using the unspecified (0) Reason Code value.*

(b) *Hold Instruction Code*

*The hold instruction code is a non-critical Certificate Revocation List entry extension that provides a registered instruction identifier, which indicates the action to be taken after encountering a certificate that has been placed on hold.*

(c) *Invalidity Date*

*The invalidity date is a non-critical Certificate Revocation List entry extension that provides the date on which it is known or suspected that the private key was compromised or that the certificate otherwise became invalid. This date may be earlier than the revocation date in the Certificate Revocation List entry, which is the date at which the Certification Authority processed the revocation.*

(d) *Certificate Issuer*

*This Certificate Revocation List entry extension identifies the certificate issuer associated with an entry in an indirect Certificate Revocation List, i.e. a Certificate Revocation List that has the indirect Certificate Revocation List indicator set in its issuing distribution point extension. If this extension is not present on the first entry in an indirect Certificate Revocation List, the certificate issuer defaults to the Certificate Revocation List issuer. On*

*subsequent entries in an indirect Certificate Revocation List, if this extension is not present, the certificate issuer for the entry is the same as that for the preceding entry.*

**(x) Issuing Distribution Point**

The issuing distribution point is a critical Certificate Revocation List extension that identifies the Certificate Revocation List distribution point for a particular Certificate Revocation List, and it indicates whether the Certificate Revocation List covers revocation for end entity certificates only, Certification Authority certificates only, or a limited set of reason codes. Although the extension is critical, conforming implementations are not required to support this extension.

**(xi) Signature Algorithm**

The signature Algorithm field contains the algorithm identifier for the algorithm used by the Certification Authority to sign the Certificate List. This field MUST contain the same algorithm identifier as the signature field in the sequence tbsCertList.

(**xii) Signature Value**

The signature Value field contains a digital signature computed upon the ASN.1 DER encoded to be signed CertList. The ASN.1 DER encoded tbsCertList is used as the input to the signature function. This signature value is then ASN.1 encoded as a BIT STRING and included in the Certificate Revocation List's signature Value field.

(2) The list of standards specified in sub-regulation (1) shall be updated at least once a year to include new standards that may emerge from the international bodies. In addition, if any Certification Authority or a group of Certification Authorities brings a set of standards to the Controller for a specific user community, the Controller shall examine the same and respond to them within ninety days.