IGTF POLICY FOR HIGH-LEVEL CERTIFICATION AUTHORITIES

Non-End Entity Issuing CAs

[IGTF POLICY FOR HIGH-LEVEL CERTIFICATION AUTHORITIES 1](#_Toc324334625)

[Non-End Entity Issuing CAs 1](#_Toc324334626)

[1. Preamble 1](#_Toc324334627)

[1.1 Document Identification 2](#_Toc324334628)

[1.2 Document History 2](#_Toc324334629)

[1.3 Document Change and Approval 2](#_Toc324334630)

[1.4 Annotation 3](#_Toc324334631)

[1.5 Terminology 3](#_Toc324334632)

[1.6 Other Terminology 4](#_Toc324334633)

[2. Introduction and Background 4](#_Toc324334634)

[2.1 The Role of the HLCA 5](#_Toc324334635)

[2.2 Accreditation and Trust 6](#_Toc324334636)

[2.3 The Review Process 7](#_Toc324334637)

[3. Requirements and Recommendations 7](#_Toc324334638)

[3.1 CP and CPS 7](#_Toc324334639)

[3.2 Namespaces 8](#_Toc324334640)

[3.3 Private Key 8](#_Toc324334641)

[3.4 Certificates and Revocation 8](#_Toc324334642)

[3.5 Acceptance Procedure 8](#_Toc324334643)

[4. Example 9](#_Toc324334644)

[4.1 Namespaces 9](#_Toc324334645)

[4.2 PKI example 10](#_Toc324334646)

[4.2.1 UK e-Science CA 10](#_Toc324334647)

[4.2.2 Commercial Hosting 10](#_Toc324334648)

[5. References 10](#_Toc324334649)

# Preamble

This document describes the IGTF (<http://www.gridpma.org/>) recommendations for Certificate Policies (CPs) for Grid Certification Authorities (CAs) that issue certificates to subordinate CAs.

## Document Identification

|  |  |
| --- | --- |
| Document OID | TBD |
| GGF Identifier | GGF-CAOPS-blah-blah |
| Status | DRAFT |
| Contact | info@gridpma.org |
| URL | https://forge.ogf.org/sf/go/doc15144?nav=1 |

## Document History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Circulation** | **Comment** |
| 0.1 | 2006-08-14 | TAGPMA | Initial version by Jens Jensen, UK e-Science CA, CCLRCSome bits based on an email from Michael Helm, DoEScienceGrid, ES-NET, sent to TAGPMA 2006-07-11. |
| 0.2 | 2006-08-15 | TAGPMA | Contributions and suggestions from David Groep, DutchGrid CA, NIKHEF. |
| 0.3 | 2007-09-19 | EUGRIDPMA | Numerous revisions, added private key prot’n and some examples, more definitions, some sections reorganised. More revisions and clarifications following EUGridPMA discussions. |
| 0.4 | 2009-01-28 | EUGRIDPMA | Nicosia revision |
| 0.5 | 2009-03-02 | IGTF | Merged older version with 3647 references added, following suggestions from Von Welch. |
| 0.6 |  | None | Nicosia/Post-Nicosia update |
| 0.7 |  | IGTF | Nicosia/Post-Nicosia update |
| 0.8 | 2009-05-25 | IGTF | Cleanup for OGF26 – finally merged with 0.5 |
| 0.9 | 2010-03-15 | IGTF | Final(?) pieces tidied up for OGF28 |
| 0.10 | 2010-09-20 | EU PMA | Tidied up version of 0.9. |
| 0.11 | 2012-05-09 | EU PMA |  |

## Document Change and Approval

This document must be modified in such a way that existing section and item numbers do not change in versions >0.*x* (not drafts).

The document shall be in effect once approved according to IGTF procedures. Update procedures for current member CAs – to implement this document – shall be defined by the IGTF. This document may be updated from time to time: in this case, the same procedure shall be followed.

## Annotation

In this document, square brackets [] are used for annotation and explaining the background or rationale behind requirements. Curly ones {} are used as references to sections of a CP/CPS compliant with RFC3647.

## Terminology

|  |  |
| --- | --- |
| Accredited | means accredited by a PMA as defined in section 2.2. |
| AP | IGTF Authentication Profile, a set of requirements for CAs issuing EE certificates. |
| CA | CA, for the purposes of this document, means a Trust Anchor according to [PKIX]: “A trust anchor is an authoritative entity represented by a public key and associated data.” |
| EE | End Entity (qv). |
| End Entity | means an entity whose certificate is not a CA certificate. |
| HLCA | means a CA which issues certificates to CAs, as defined in section 2. |
| IGTF | International Grid Trust Federation (www.igtf.net)  |
| Intermediate | of a CA, means a HLCA which is not a Root. |
| Namespace | See the IGTF RPDNC document. |
| Offline | of a physical machine, means that it is not, and has not been, connected to any network at any time with the operating system that it is currently running. See also section 3.3. |
| Online | means not Offlline. |
| PMA | Policy Management Authority, the formal members of IGTF (qv). CAs are reviewed by a PMA and become members of it once they are Accredited. |
| Root | means a self-signed certificate, or a CA with self-signed certificate, depending on context. |
| Subject CA | of a CA, means a CA certificate signed by the CA’s certificate. |
| Subordinate CA | of a CA, means another CA whose certificate validation chain to a Root contains the certificate of this CA. |
| Trusted | means trusted for the purpose of being distributed with the PMA distributions according to the definition in section 2.2. |
| Trust Path | means the certification path (i.e., a chain of certificates) from a given certificate to a Root *as defined by the* *subject* *distinguished* *names* of the CAs in this path. |

## Other Terminology

Standard Grid CA terminology and abbreviations are not explained in this document.

RFC Terminology:

1. In this document, the keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in RFC 2119.
2. The keyword "SHOULD" is to be interpreted as follows: there may exist valid reasons in particular circumstances to ignore a particular item; in that case the full implications must be understood and, the CA manager must explain why the item is not followed. For example, the CA manager may demonstrate that something equivalent to the item or better is in place.
3. The keyword "SHOULD NOT" is to be interpreted as follows: there may exist valid reasons in particular circumstances when the particular behavior is acceptable or even useful; in that case the full implications must be understood and, again, the reasons must be demonstrated to the PMA.

[SHOULDs are meant to leave flexibility in the minimal requirements to enable diverse CAs to meet the requirements without having to alter procedures which are as good as, or better than, the minimal requirements, or, conversely, to have to rewrite the minimal requirements.]

# Introduction and Background

When a CA is Accredited (to be defined in section 2.2) by the IGTF (specifically, by one of its PMAs), its certificate is published in the IGTF distribution along with any certificates in a Trust Path to a self signed Root.

Requiring the full Trust Path in the trusted certificate store is a good idea for the following reasons:

* Having the Trust Path in the trusted store prevents attacks that aim to subvert certificates by faking CA certificates.
* All CRLs associated with certificates in the Trust Path are refreshed regularly and cached locally.
* Namespace enforcement: Relying Parties can restrict namespaces of CAs to a subset of its normal namespace (see [RPDNC]).

The CAs in the Trust Path, excluding the Accredited CA, are referred to as “High Level Certification Authorities” (**HLCA**s) throughout this document, mainly for lack of a better word. A HLCA is thus a *root* CA (self-signed), or an *intermediate* CA (part of a validation chain up to a root), neither of which issues EE certificates (or if they do, they do not seek Accreditation in this respect). This document describes the IGTF requirements for such CAs.

Unless mentioned otherwise, it is assumed throughout this document that the PKI forms a *tree* with a single Root, and thus that all validation chains can build one and only one path to the Root. Indeed, Trust Paths are usually built based on the names of the CAs.

A CA may have more than one CA certificate.

## The Role of the HLCA

The role and *raison d’être* of the HLCA is usually one or more of the following. A CA Manager who writes the CP for a HLCA may consider these points.

1. A HLCA can define a common community for all its subordinates, and can impose policy restrictions on their policies.
2. If the policy of a HLCA is strong enough, a resource provider may decide to implicitly trust all its Subordinates, and the PMA may be able to fast-track their Accreditation.
[Rationale: to some extent we can already dynamically deploy new CAs into trust anchors. If in the future we support TAMP [TAMP], or some other process for dynamically deploying middleware – or support “traditional” security infrastructures which permit clients to send intermediate CA certificates on authentication – in these cases in may be beneficial to have a “fast track” approval for a new Trust Anchor, similar e.g. to the deployment of a rollover certificate. If the *issuer* constrains the Subject to such an extent as to ensure that the Subject itself is accreditable, the Subject CA can obviously be deployed immediately without passing through a lengthy review.]
3. HLCAs may allow different subordinates to have different assurance levels, or serve different purposes in the same community.
4. In practical terms, running and supporting a production Grid CA, at least a Classic one, is always a lot more effort than anyone (who hasn’t done it before) would think. One should think carefully whether a hierarchy is really needed. For example, if distributed sites wish to issue their own certificates, but all to roughly the same assurance level, it is often better to make them RAs.
5. A CA certificate can be revoked if the revocation is signed by a HLCA. Otherwise the CA will have to revoke itself, and the efficacy of this obviously relies on whether the middleware checks the CA certificate against *its own CRL* at the time of reliance.
6. A CA whose Subjects are themselves CAs can be used to define a common purpose or community for those Subject CAs.
7. It is currently considered best practices by the IGTF to have an online CA subordinate to an offline Root.

Bridging is not considered in this document since bridging doesn’t work with current Grid middleware.

## Accreditation and Trust

In this document we distinguish between *Accreditation* and *Trust*. PMAs *Accredit* CAs who then have voting rights in the PMA. HLCAs should not be Accredited, but only Trusted, by the IGTF PMAs. A CA is either Trusted or Accredited (or neither), never both.

The CAs to which the HLCA issues certificates are referred to as its **Subject CA**s. Any CA in the hierarchy below the HLCA is referred to as a **Subordinate CA**, i.e., Subordinate CA of a HLCA is a CA whose certificate validation chain contains the certificate of the HLCA somewhere in the chain.

**Acceptance** refers to a CA whose CP/CPS has been reviewed by a PMA according to the applicable profiles, and has been declared either *Accredited* or *Trusted*.

**Accreditation** means the case described in the IGTF charter and covered in the charters of the PMAs where a CA is:

* A full member if its accrediting PMA, with voting rights, represented by its CA Manager who shall attend PMA meetings according to the PMA’s requirements; and,
* Its certificate is made available from the PMA’s repository, along with pointers to the all necessary documentation and information (CP/CPS, CRL if applicable, etc); and,
* Its CP/CPS has been reviewed by the PMA according to the applicable AP, and found acceptable; and,
* It has passed an operational review according to the practices of the relevant PMA; and,
* The CA is trusted by the PMA to issue certificates in its designated namespace.

**Trusted** means the limited case where a CA is:

* Not a member of the accrediting PMA, and has no voting rights; and,
* Its certificate and other relevant information is published by the PMA’s repository, as in the case of an accredited CA; and,
* Its CP/CPS has been satisfactorily reviewed by the PMA according to the most recent version of ***this document***; and,
* It has passed an operational review to determine whether it meets the operational requirements imposed by ***this document***; and,
* The PMA has decided whether to implicitly Accept any or every Subordinate CA of the CA being reviewed, or whether any Subject CA should itself be subject to an Acceptance review; and,
* The CA is trusted by the PMA to issue certificates in its designated namespace.

We shall refer to the former case – each Subject CA is reviewed for Acceptance – as **Explicit Acceptance** of the Subject CA. This PMA policy is expressed in RP name space restrictions by explicitly naming all Trusted subject DNs.

Conversely, we refer to the latter case – some or all Subject CAs are automatically Accepted – as **Implicit Acceptance** of these Subject CAs. This is encoded in RP namespace restrictions using a string followed by a wildcard (in the default OpenSSL stringification – see section 3.2.3).

## The Review Process

To aid the review process, a checklist spreadsheet is also available (ref).

# Requirements and Recommendations

This section describes the requirements and recommendations for the policy of a root or intermediate CA; one that does not itself issue certificates to EEs – hereinafter referred to as “HLCA”. The references to sections of RFC3647 are meant as a guide; they do not impose the requirement that the information be described fully or in part in those sections.

To some extent, this document relies on being recursive: if a HLCA is intermediate, its own issuer is itself a HLCA, and this document applies to it, too. However, there are cases where IGTF CAs are issued by HLCAs which are not themselves Grid CAs. Nevertheless, it is the purpose of this document that even non-Grid HLCAs SHALL be satisfactorily reviewed according to this document prior to being Trusted by a PMA.

## CP and CPS

1. A HLCA must have a CP, and a CPS conforming to the CP. New CAs SHOULD structure them according to RFC3647.

It is RECOMMENDED to format according to RFC3647 and leave out the sections saying “no stipulation” (if there are many of those) but keep the numbering.

1. Repository obligations {2.1, 2.4}. The CP and the certificate of a HLCA, MUST be published {2.1, 6.1.4}. The CPS SHOULD be published. The PMA SHOULD be given access to the CPS for the purpose of reviewing it. If not, the PMA MUST have a report from an approved auditor sufficient to verify that the CPS implements the CP, and is being followed, and complies with the requirements of this document. A fee MUST NOT be charged for access to CP or CPS.
2. A HLCA’s CP MUST be consistent with the CP of its Issuer, and that of its Issuer’s Issuer, and so on, up to the Root. [Rationale: a CA MAY impose restrictions on its Subordinates (item 6); in this case this item says Subordinates MUST comply with them.]
3. A HLCA SHOULD describe hierarchy or hierarchies into which it fits {1.1, 1.3.1}. A HLCA MUST describe its Trust Path.
4. A HLCA MAY define a consistent community for all its Subordinates {1.1, 1.3.3}. The community of a HLCA MAY be a proper subset of that of its issuer – if so, it is RECOMMENDED that the HLCA describes this. [Rationale: this is one of the use cases (above) for HLCAs.]
5. A HLCA MAY impose restrictions on the CP and CPS of its Subordinates, other than those described in and required by this document. In particular, Subordinates MAY be covered by the same CP and/or CPS as the HLCA.

## Namespaces

1. The Trust Path from the Root to the EE-issuing CA MUST be documented {1.3.1}. Names SHOULD be X.500 distinguished names {3.1.1}. The Namespace of the HLCA SHOULD be documented {3.1.4 }. [It should be possible to write a signing policy for the HLCA describing the Trusted Path from the Root to the EE-issuing CA.]
2. Any single Subject distinguished name MUST be linked to one and only one Subject over the entire lifetime of the HLCA it MUST NOT be linked to any other Subject {3.1.5}. [A Subject by definition is a single Subject CA possessing a single certificate, except when rolling over.]
3. A HLCA with implicitly Accredited Subject CAs MUST ensure that the Subject CAs between them do not issue the same DN to different entities {3.1.5}.
4. For an HLCA with explicitly Accepted Subject CAs, these MUST be Accepted explicitly by Subject name.
5. ~~There is no section 5.~~

## Private Key

* 1. The private key, if based on RSA, MUST have a key length of no less than 2048 bits, or equivalent strength in other ciphers {6.1.5}.
	2. Subject CA certificates in the Trust Path SHOULD NOT be generated without human intervention {section 4}.

If the HLCA is a Root, its signing machine SHOULD be Offline {6.5.1}.

* 1. A CA SHOULD have a single private key except when rolling over {5.6}.

## Certificates and Revocation

1. A HLCA SHOULD NOT issue EE certificates {1.3.3}. If it does, they MUST be the minimum necessary for its own operation.
2. The HLCA MUST publish a CRL {4.9.7} using a HTTP URL, and MAY use other means of publishing certificate status information. The CA MUST NOT charge a fee for certificate status information for certificates in the Trust Path
3. The certificate of the HLCA SHOULD comply with GFD.125 [ref]

## Acceptance Procedure

Briefly, for any CA seeking Accreditation, the CA Manager must ensure that a Trusted chain is built up to a Root. For this purpose, the CA Manager of the CA seeking Accreditation may represent all the HLCAs of the CA seeking Accreditation, if the HLCAs themselves are not to be Accredited, but only Trusted, by the PMA.

1. The CA Manager of ***any*** CA seeking Accreditation from a PMA MUST ensure that ***all*** HLCAs above it in a suitable chain up to a Root, are Accepted by the PMA.
2. The CA Manager of ***any*** CA seeking Accreditation MAY represent HLCAs in the chain before the PMA ***if*** the HLCAs in question are to be Trusted by the PMA.
3. If applying for Accreditation for a HLCA,
	1. The CA Manager MUST appear before the PMA to present the HLCA’s CP/CPS.
	2. The CA Manager MUST get agreement from the PMA whether Subject CAs are Implicitly or Explicitly Accepted. [The signing policy file must be approved by the PMA]
4. The CA Manager MUST ensure that the HLCA issues in a well defined namespace, and MUST supply a signing policy file, such that ***all*** Explicitly Accepted Subject CAs are admitted, and ***no*** Subject CA which has not been approved for Implicit Acceptance by the PMA is admitted.
5. This document does not require that all Subject CAs of an Accepted HLCA should themselves be Accepted.

# Example

Non-normative examples of structuring a hierarchical Grid PKI.

## Namespaces

1. It follows from 3.2.2 and 3.2.3 that there should be “enough” RDNs to divide the namespace, for all Subordinate CAs. Thus, when designing the namespace for a hierarchy, it is important to consider how deep the hierarchy should be (including future developments).

Example: /C=TV /O=Grid/O=Medium/OU=People/CN=common name

(The reader will note we are abusing the Os and OUs a bit; this is common practice and not a cause for concern.) In this example, /C=TV/O=Grid establishes the common namespace root for this Grid CA hierarchy in Tuvalu (leaving space for other national CAs outside this hierarchy). O=Medium allows for other assurance levels, the OU=People allows for separate people and host namespaces, or even separate CA certificates for people and hosts, like DFN (Germany) does.

1. If there aren’t enough RDNs, it is possible to use a single RDN to establish a hierarchy, thanks to the “leftmost” evaluation in signing policy files. Continuing the example from 1, assume the CA manager intends to fit a hierarchy into a namespace which contains only /C/O/CN. This can be done with /C=TV/O=GridMediumPeople/CN=common name. This will allow a signing policy to accept up to /O=Grid\*, /O=GridMedium\*, by pattern matching the string, thus enabling RPs to accept all Grid CAs, all Medium assurance CAs, etc.
2. If there aren’t enough RDNs, a better solution than 2 is to give up on 3.2.3, and establish an independent namespaces for all Subordinate CAs, independent of the issuer’s namespace. However, the Subordinate CAs should still satisfy 3.2.3, i.e., have a common part of their namespace.

## PKI example

### UK e-Science CA

...

Root

NGS / NeSC

Training CA

STFC
SSO CA

SARoNGS

CA

SLCS TopLevel

UK eScience CA

The diagram shows a hierarchy of CAs, not all of which are accepted by IGTF. However, as the Root has to be distributed with the IGTF release, the Root has to be Trusted. There is only one CA with voting rights in the PMA, the one which is accredited (here, UK eScience CA).

### Commercial Hosting

Suppose a commercial provider offers CA hosting services. The service includes generating the keys and the CA certificate, and signing EE certificates with the keys. Suppose further that some of these CAs are eligible for membership of the IGTF (they could be in separate countries, for example).

In this case, the commercial Root is Trusted by the IGTF, but each of its subordinate CAs which is eligible for membership of the PMA is Accredited. The subordinate CAs have one vote each in their PMA

# References

[IGTF-HLCA-XLS] https://forge.gridforum.org/sf/go/doc15895?nav=1

[PKIX] R Housley, S Ashmore, C Wallace: *Trust Anchor Format*, http://www.ietf.org/id/draft-ietf-pkix-ta-format-04.txt [retr. March 2010]

[RPDNC] D Groep, J Jensen: *Relying Party Defined Namespace Constraints*, https://forge.gridforum.org/sf/go/doc4857?nav=1

[TAMP] R Housley, S Ashmore, C Wallace: *Trust Anchor Management Protocol*, http://www.ietf.org/id/draft-ietf-pkix-tamp-06.txt [retr. March 2010]