Chapter 6

Comparison and Assessment of Certificate Policies for Merging Hierarchical PKIs

6.1 Introduction

The Public Key Infrastructure (PKI) provides facilities for data encryption, digital signature and time stamping. It is a system where different authorities verify and authenticate the validity of each participant with the use of digital certificates. A Certificate Policy (CP) is a named set of rules and it indicates the applicability of a certificate in a Public Key Infrastructure. A Certificate Practice Statement (CPS) states how a certificate authority implements a CP[84]. They have a common outline, as indicated in the IETF Request for Comments (RFC) 3647[15] or in the older version, RFC 2527[82]. The policy indicates the PKI certificates’ profile and the architectural structure of the underlying trusted third party. A certificate authority issues a certificate to an end user with a specific certificate policy[7]. If end users with different domains want to establish a secure communication, they need to find a way to trust each other. That means their certificates need to follow the same standard. X.509 is a standard that specifies the standard format for
the certificates, the certificate revocation lists and the certificate path validation algorithm.

One of the main PKI needs is interoperability, which ensures secure interconnection and co-operation between different PKI structures, thus enhancing their feasibility and applicability at regional, national, as well as international level. PKI interoperability is usually addressed through the cross-certification service, which can be described as the way to establish chains of trust between different certification authorities (CAs)[96][97][98]. However, cross-certification is not yet technically provided in an automated way, resulting often in a difficult and time-consuming paper-based process, which reduces the flexibility and usability of the method itself. The lack of automated cross-certification is largely due to the inadequate standardization of the certificate policies (CP), which define the PKI certificates profile and, thus, form the basic comparison criteria for the mutual acceptance of CAs[82]. More specifically, although the CP structure is defined in some of the existing standards[94], there is still a significant gap in the standardization of the CP content (e.g., roles of the involved subjects, certification and registration requirements, etc). In addition, there is no systematic way for the development and the comparison of CPs, thus making their comparative analysis a difficult task[7].

The above restrictions together with the lack of necessary legal/regulatory PKI harmonization[87] disable the automated CP comparison and obstruct the automation of the overall cross-certification service. Thus, it makes the secure electronic co-operation, information exchange and knowledge sharing difficult.

6.1.1 Certificate Policies

Certification Authorities (CAs) play a major role in establishing certificate policies (CP). A CA may work with a group of companies or an industry to create a certificate policy accepted as appropriate for use in that sector, and relying parties
may simply write their individual policies to reflect this. In other situations, CAs may have to set de facto policy. For example, the issuers of certificates used to authenticate web-sites using SSL (Secure Sockets Layer) may have established the policies and processes by which they will authenticate web servers. In doing this, they may require letters of incorporation or other formal documentation before issuing a server certificate. Relying parties, whether individuals or employees accessing web servers they list, trust that web-site actually belongs to that company, but do not have oversight of the policies by which those certificates were issued or managed. Once established, a CA can identify a policy, including qualifying information about the policy, in a certificate. By doing so, the CA is declaring the intended use of the certificate to relying party who may process the certificate. Similarly, a relying party can simply look for the appropriate policyIdentifier information in a certificate to assist in processing certificates that are acceptable to that relying party. Policies are generally written in standard format. RFC 2527, the Certificate Policy and Certification Practices Framework, defines the accepted standard. RFC 2527 includes a standard outline of eight major sections and 185 second and third level subsections. Since the standard format has a number of distinct advantages, most CPs are written to this outline.

By adhering to a well-defined format, the CP writer can easily write the policy document. It would be easy to overlook a few of the 185 topics identified in RFC 2527 if the author changes the outline. Adhering to the standard format will also simplify cross-certification with other CAs. The cross-certification process should always include a comparison of the other CA’s certificate policies. This information is used to determine the contents of the policy mappings and policy constraints extensions to be included in the CA certificate. The eight major sections are summarized as follows:

- The *Introduction* explains how to identify certificates issued under this policy (i.e., the object identifier that will appear in the policy extension), defines
the community for these certificates (e.g., organization employees, or financial managers), and provides contact information for the people who administer the CA and maintain the policy.

• The Publication and Repository Responsibilities contains information regarding the certificate repositories, publication of certification information, time and frequency of publication and access controls on repositories. This section also describes what information (if any) will be considered confidential and the circumstances that would justify disclosure.

• The General Provisions captures broadly applicable legal and general practices information. This section identifies the various participants in the PKI (e.g., CA, RAs, subscriber, and relying party) and their various obligations and liabilities. It identifies the applicable laws, fees, and auditing requirements. This section also describes what information (if any) will be considered confidential and the circumstances that would justify disclosure.

• Identification and Authentication describes the procedures used to authenticate requests for certificates, or for certificate revocation.

• Operational Requirements describes the operations that must be performed by the CA, RAs, end-entities, or other parties under this policy. Specific actions are identified that must be performed when requesting or generating new certificates, revoking certificates, creating and protecting audit logs, archiving records, changing the CA’s key, disaster recovery, and terminating the CA’s operations.

• Physical, Procedural, and Personal Security Controls describes how the PKI uses physical security (e.g., guards, gates etc.), procedures (e.g., separation of duty), and personnel requirements(e.g., background checks and procedures) to complement the technical security controls.
• *Technical Security Controls* describes the security measures used to protect cryptographic keys (e.g., a FIPS 140 validated hardware module), to protect critical security parameters (such as the list of trusted RAs), and to provide quality assurance for the systems and protect the CA from network-based attacks.

• *Certificate and CRL Profiles* specifies the certificate and CRL profile. This section specifies the cryptographic algorithms that will be used to sign the certificates, the length of the signing key, and the name forms that will appear in certificates. It describes the extensions that are included in certificates and CRLs.

• *Specification Administration* is the final section, and it describes who will maintain the policy. It may also describe the procedures that will be followed if the specification is changed, how these modifications will be published, and the approval procedures.

### 6.1.2 Drafting a Certificate Policy

The role of policy in PKI is critical as it defines the level of risk for relying party applications in a given community of interest. The first requirement for an agency deploying a PKI is to establish appropriate certificate policy(ies). These policies must define the level of trust that can be placed in a certificate when it is presented to a relying party application (e.g., web server), a level of trust that will be related directly to the assurances provided in the overall certificate insurance and management process. Policies must also define the rules and liabilities of the parties involved in issuing, managing, and processing certificate. The set of policies under which a CA issues certificates is termed as *trust domain* or *policy domain*. The agency needs to obtain an object identifier (OID) for each of the policies in its trust domain. These OIDs will be used to differentiate the appropriate set of applications for a particular certificate. An X.509 v3 certificate extension may state one
or more certificate policies in the certificatePolicy extension. A certificatePolicy extension contains one or more policyIdentifiers. A policyIdentifier is a unique, registered OID that represents a certificate policy in a certificate. Applications may use these policies to decide whether or not to trust a certificate for a particular purpose. The application that registers the OID also needs a textual specification of the certificate policy, for the purpose of checking by certificate users and other applications.

The policy(ies) must reflect the types of applications that will be secured by the PKI. An effective strategy is to adapt and reuse existing policies to create policy(ies) for the agency. Certificate policies should be at a sufficiently high level so that the policies will not change too frequently. The format and the content of these policies are discussed in detail in the subsequent sections. The following extension fields in an X.509 certificate are used to support CPs:

- Certificate Policies extension;
- Policy Mappings extension; and
- Policy Constraints extension.

### 6.1.2.1 Certificate Policies Extension

The certificate policies extension defines the policy under which the CA issued the certificate and the purposes for which applications and people can use the certificate. Applications can use this field to discover rules established by the CA, governing the use of the certificate. Applications can then compare policies defined in the certificate with any of their own policy requirements. The field can include a pointer to a CPS published by the CA. It can also include a text string defining the policies for certificate use and application and display it to a user. When processing a certification path, a CP that is acceptable to the relying party application must be present in every certificate in the path, i.e., in CA-certificates as well as end entity certificates. If the Certificate Policies field is flagged critical, it serves the
same purpose as described above, but also has an additional role. Specifically, it
indicates that the use of the certificate is restricted to one of the identified policies,
i.e., the certification authority is declaring that the certificate must only be used
in accordance with the provisions of at least one of the listed CPs. This field is
intended to protect the certification authority against claims for damages asserted
by a relying party who has used the certificate for an inappropriate purpose or in
an inappropriate manner, as stipulated in the applicable CP.

6.1.2.2 Policy Mappings Extension

The Policy Mappings extension may only be used in CA-certificates. This field
allows a certification authority to indicate that certain policies in its own domain
can be considered equivalent to certain other policies in the subject certification
authority’s domain.

6.1.2.3 Policy Constraints Extension

The Policy Constraints extension supports two optional features. The first is
the ability for a certification authority to require that explicit CP indications be
present in all subsequent certificates in a certification path. Certificates at the
start of a certification path may be considered by a relying party to be part of
a trusted domain, i.e., certification authorities are trusted for all purposes so no
particular CP is needed in the Certificate Policies extension. Such certificates need
not contain explicit indications of CP. When a certification authority in the trusted
domain, however, certifies outside the domain, it can activate the requirement that
a specific CP’s object identifier appear in subsequent certificates in the certification
path. The other optional feature in the Policy Constraints field is the ability
for a certification authority to disable policy mapping by subsequent certification
authorities in a certification path. It may be prudent to disable policy mapping
when certifying outside the domain. This can assist in controlling risks due to
transitive trust, e.g., a domain A trusts domain B, domain B trusts domain C, but
domain A does not want to be forced to trust domain C.

6.1.3 Certificate Practice Statement (CPS)

The IETF Framework defines the CPS as the statement of practices which a certification authority employs in issuing certificates[14]. The concepts of certificate policy and CPS come from different sources and were developed for different reasons. However, their interrelationship is important. A Certification Authority prepares a certification practice statement. The subscribers and certificate users (relying parties) are potentially need to understand the CPS. Although the level of detail may vary among CPSs, they will generally be more detailed than certificate policy definitions. Indeed, CPSs may be quite comprehensive, robust documents providing a description of the precise service offerings, detailed procedures of the life-cycle management of certificates, and more - a level of detail which ties the CPS to a particular (proprietary) implementation of a service offering. A CA with a single CPS may support multiple certificate policies (used for different application purposes and/or by different certificate user communities). Also, multiple different CAs with non-identical certification practice statements may support the same certificate policy. For example, the Federal Government might define a government-wide certificate policy for handling confidential human resources information. The certificate policy definition will be a broad statement of the general characteristics of that certificate policy, and an indication of the types of applications for which it is suitable for use. Different departments or agencies that operate certification authorities with different certification practice statements might support this certificate policy. At the same time, such certification authorities may support other certificate policies. In addition to populating the certificate policies field with the certificate policy identifier, a certification authority may include, in certificates it issues, a reference to its certification practice statement.

The Certificate Policy and Certificate Policy Statement are important docu-
ments for Certification Authority and subscribers. This makes the Certification Authority accountable and responsible to subscriber, in case of legal disputes.

6.2 The Role of CPs in Merging Hierarchical PKIs

Sometimes two companies or organizations with different PKIs merge. Therefore it would be necessary that their infrastructures are also able to merge. Different possibilities exist for merging Hierarchical Public Key Infrastructures. First, the root authorities of the PKIs exchange their certificates. Then, the root authority of PKI A can act as root authority of PKI B and vice versa. This mechanism is called cross-certification. The second possibility is that the root authority of PKI A issues a certificate to the root authority of PKI B. In this certificate, the rights of a certificate authority of PKI A are given. After that, the root authority of B becomes certificate authority of PKI A and it has to issue new certificates to all underlying trusted third parties. The other two techniques of merging Hierarchical PKIs are explained in chapter 4 and chapter 5. In order to merge two infrastructures, the certificate policies should be merged as well. The unification of different PKIs may be rejected because of the mismatch in the policies.

6.3 Parsing Certificate Policies

First of all, the outline of the certificate policies will be parsed. Parsing makes the comparison of the outlines of CPs of different PKIs possible. The parsing is carried out according to some syntactical rules. Parsing the outline of certificate policies is proposed in RFC 3647[15]. Two CPs are parsed and compared. Based on the degree of comparison a score is calculated which is described in sections 6.4 and 6.5. The Figure 6.1 shows the flowgraph of the parsing process. Associated certificates are issued according to the X.509 version 3 standard.
As shown in Figure 6.1, the CPs of two PKIs are taken as the input for the parsing process and different tokens (paragraphs and subparagraphs) from each CP are stored in a tree data structure for comparison. Based on the compatibility score, a prototype of a unified certificate policy for the merging PKIs will be created.

Figure 6.1: Flowgraph of the parsing process

A certificate policy is prepared in the .TXT format. The following lines represent a part of a typical certificate policy:

1 INTRODUCTION

1.1 Overview
   //Gives an overview of the document

1.2 Document name and identification
   a) RECOMMENDED Document name
b) MUST Designated identification

Connection AND

1.3 PKI participants

//Described in the subsections

1.3.1 Certification authorities

a) Issues certificates to end users

b) Issues certificates to other users

1.3.1.1 Root authorities

a) Specifies the difference to the CAs

1.3.2 Registration authorities

1.3.3 Subscribers

1.3.4 . . .

In order to parse a certificate policy, it is represented in Backus Naur Form (BNF) or Extended Backus Naur Form (EBNF) [99]. The syntax of the certificate policy is described in Extended Backus Naur Form notation in the following way:

CertificatePolicy = {Section}

Section = Point{MainSection | SubSection}

Point = Number " . "

MainSection = \{CapitalLetter ["("Number")"] \} [Weight ] "\n" Content

SubSection = \{Point\} Number String [Weight ] "\n" Content

Number = Integer

Weight = Integer

Content = \{Option | [ Connection ]\} {Extension}

Option = [letter ")"] [ Keyword ] String "\n"

Connection = "connection" ("AND" | "OR")

Extension = Subsection

Keyword = "MUST" | "RECOMMENDED" | "OPTIONAL" | "NOT"

A symbol can either be a number, a dot or a string. Number means the enumeration...
of the current paragraph or the subparagraph, and the string stands for the current title of the paragraph. The dot is used as a separator in an enumeration. The parser will check both the title and content of a paragraph. After parsing the certificate policies, the tokens are stored in designated data structures. Two data structures “Token” and “TokenList” are introduced. “Token” represents a simple entry. “TokenList” contains all entries in the same level under a section, e.g., the main token list, which contains all main sections. At the end, the content of a certificate policy is stored as a tree data structure as shown in Figure 6.2.

![Tree structure representing a certificate policy](image)

6.4 Formalization of CP comparison and assessment

The CP comparison and assessment tool is developed in Java with OpenSSL and NetBeans 6.7.1 as the IDE. In the implementation, a token represents a paragraph and a token list represents a list of paragraphs. Under every paragraph there are subparagraphs and up to four levels of indexing is considered. A Token list is a collection of tokens. Let $P_A$ be an arbitrary paragraph of a standardized certificate policy of an entity A and $P_B$ be the corresponding paragraph of the policy of entity B. $S_i$ is the score of a paragraph(token) $i$ in a list. $l_i$ is the score of
subparagraphs(list) under the paragraph i, with i = 1, . . . , N where N is the number of paragraphs. $o_{jk}$ refers to the provisional result of the equality of option j of $P_A$ with option k of $P_B$. $v_j$ refers to the keyword value of option j in $P_A$, and $v_k$ is the keyword value of option k in $P_B$. Each option in the paragraph of policy A is compared with the same paragraph of B’s policy. Basically, the value of $o_{jk}$ is 100 if the corresponding options in both the policies are same and 0 if they are different.

6.4.1 Weight of a paragraph

The updated weight of a paragraph i depends upon the size of the TokenList. If there are many elements in the paragraph, i.e., if the size of the TokenList is large, the updated token score gets less weight as shown in equation (1).

$$S_i = \frac{S_i + l_i \times N}{1 + N} \quad (1)$$

For example, If the original paragraph score, $S_i$ is 100, subparagraph score is 75 and the number of elements(subparagraphs) under that paragraph is 2, the updated score is

$$S_i = \frac{100 + 75 \times 2}{1 + 2} = 83.3$$

If the number of elements is increased to 8,

$$S_i = \frac{100 + 75 \times 8}{1 + 8} = 77.8$$

Thus, as the number of elements increase, the paragraph weight decreases.

6.4.2 Merging PKIs based on policies

The calculation of the score of a token depends upon the following two situations:

- $P_A$ has some options, $P_B$ has no options

  In this case, merging A and B with cross-certification is not allowed since the score $S_i$ is 0. However, if A acquires B, B can take the options of A and the score $S_i$ is 100.
• **$P_A$ has some options, $P_B$ also has some options**

In this case, for merging A and B with cross-certification, $S_i$ is calculated as the ratio of sum of the partial results of the options to the maximum number of options from both sides as in equation (2).

$$S_i = \frac{\sum o_{ijk}}{\text{Max}(\text{NumberOfOptions}(P_A), \text{NumberOfOptions}(P_B))}$$  \hspace{1cm} (2)

If A acquires B, the sum is divided by the number of options of $P_A$ as shown in equation (3).

$$S_i = \frac{\sum o_{ijk}}{\text{NumberOfOptions}(P_A)}$$  \hspace{1cm} (3)

### 6.4.3 Overall CPs compatibility Score

Let us consider the certificate policies $CP_A$ and $CP_B$ that belong to the organizations A and B respectively. Let us also consider that A is the comparing organization, and, therefore, it is the one that sets the criteria for the CPs comparison (or else the one that decides about possible compatibility with B). This means that, the CPs’ comparison is actually the assessment of $CP_B$ against $CP_A$ using the specific criteria of A.

The overall compatibility score (total score) of $CP_B$ against $CP_A$ is extracted as a weighted average of all CP paragraphs scorings, as in equation (4):

$$c = \frac{\sum (S_i \times W_i)}{\sum W_i}$$  \hspace{1cm} (4)

where C defines the overall compatibility score of $CP_B$ against $CP_A$, $W_i$ defines the weight of CP paragraph $P_i$ and $S_i$ defines the scoring for CP paragraph $P_i$. The addition $\sum (S_i \times W_i)$ is the weighted compatibility score of the two CPs, whereas $\sum W_i$ is the maximum possible overall compatibility score of the two CPs.

The total score can also be calculated using equation (5):

$$c = \frac{\sum S_i}{N}$$  \hspace{1cm} (5)

where N is the number of paragraphs.
6.4.4 Keywords

Some keywords are also helpful to prepare the Certificate Policy. The proposed key words introduced in RFC 2119[85] are MUST, MUST NOT, REQUIRE, SHALL, SHALL NOT, RECOMMENDED and MAY / OPTIONAL. Only a few of the proposed keywords are used in the implementation. They are: MUST, RECOMMENDED, OPTIONAL, and NOT. The options of a paragraph can look like: 

\textit{KEYWORD phrase}

Basically the keywords are interpreted by numerical values between 0 and 1.

\begin{tabular}{l|c}
\textbf{MUST} & 1.0 \\
\textbf{RECOMMENDED} & 0.8 \\
\textbf{OPTIONAL} & 0.5 \\
\textbf{NOT} & 0.0 \\
\end{tabular}

Since these values are fuzzy in nature, we can get a token score from Fuzzy Logic[141]. The idea is that, the score should not be 0 or 100 strictly, it shall be possible to get scores in between 0 and 100. Fuzzy logic gives a degree of truth, which is between 0 and 1 so that the scores will be between 0 and 100. To adopt this idea, the degree of similarity of the keywords will be taken. The keyword values have certain meanings. “MUST” indicates an absolute necessary option and has the value 1.0. On the other side, ”NOT” with 0.0 indicates a non desired option. “OPTIONAL” means, the option may or may not occur and has the value 0.5. “RECOMMENDED” is closer to “MUST” than to “OPTIONAL”, so the value is 0.8. The keywords and their values provide an exact determination of equality and a better expression of the need of a certain option. They also state a kind of degree of truth. If the same option is found at both paragraphs $P_A$ and $P_B$, and the keywords are similar, like “MUST” and “RECOMMENDED”, the paragraph score is high. Therefore the difference in the keywords will be subtracted by 1 and the result is multiplied with 100 (score of equality).
6.4.5 Connections of Options

Connectives like, OR and AND can be used with options. With an OR connection, we choose one option out of many. It is mathematically represented as the maximum value of the provisional scores of the equal options as in equation (6).

\[ S_i = Max(o_{jk} \times (1 - |v_j - v_k|)) \]  

(6)

With an AND connection, all the options have to be chosen. So it has to be determined that how many options are equal with what degree. The score can be calculated as in equation (7).

\[ S_i = \frac{\sum o_{jk} \times (1 - |v_j - v_k|)}{Max(\text{Number of options}(P_A), \text{Number of options}(P_B))} \]  

(7)

The equation (7) refers to the policies that should be merged with cross-certification. If A acquires B, the denominator is the number of options of \( P_A \), because B has to adapt to A. Therefore, the number of options of B are irrelevant as in equation (8).

\[ S_i = \frac{\sum o_{jk} \times (1 - |v_j - v_k|)}{\text{NumberOfOptions}(P_A)} \]  

(8)

For example, consider the options in the paragraphs \( P_A \) and \( P_B \) as shown below:

\[ P_A \quad P_B \]

a) MUST a  a) RECOMMENDED a

b) MUST b  b) OPTIONAL b

b) MUST c  c) RECOMMENDED d

d) RECOMMENDED e

Suppose the connection is OR, then both for merging(merging PKIs with cross-certification) as well as acquisition of companies,

\[ S_i = \text{Max}(100 \times (1 - (1.0 - 0.8)), 100 \times (1 - (1.0 - 0.5)), \ldots, 0) = 80 \]

Suppose the connection is AND and the requirement is that A and B are to be merged with cross certification,
\[ S_i = \frac{80}{4} + \frac{50}{4} + 0 + 0 = 32.5 \]

Suppose the connection is AND and A acquires B,

\[ S_i = \frac{80}{3} + \frac{50}{3} + 0 = 43.3 \]

6.4.6 Final acceptance rules

The final acceptance rules are set by the comparing organization and are those that determine whether the foreign CP can be accepted or rejected, after the CPs comparison has been performed. Examples of such rules could be a combination of statements like: “the overall CP compatibility score should be over 90%”, “partial compatibility in paragraph X should be over 80%” etc. It is also possible to set as only final acceptance rule that the partial CP compatibility in one (or more) paragraph is 100%, which means that a foreign CP may be rejected solely because of incompatibility in this specific paragraph. Based on this information, the final acceptance rules can reflect any acceptance or rejection policy that the comparing organization wants to follow and, thus, make the method very flexible to different needs and requirements.

6.5 Experimental Results

A portion (i.e., paragraphs 1, 2 and 3) of two CPs, \( CP_A \) and \( CP_B \) as given below are considered as the certificate polices of the PKIs to be merged.
1. INTRODUCTION

1.1 Overview
Gives an overview of the document

1.2 Document name and identification
a) RECOMMENDED Document name
b) MUST Designated identification
Connection AND

1.3 PKI participants
Described in the subsections
1.3.1 Certification authorities
a) Issues certificates to end users
b) Issues certificates to other users
1.3.1.1 Root authorities
a) Specifies the difference to the CAs

1.3.2 Registration authorities
1.3.3 Subscribers
1.3.4 . . .

2. PUBLICATION AND REPOSITORY RESPONSIBILITIES

3. IDENTIFICATION AND AUTHENTICATION

3.1 Naming
3.1.1 Types of names: X.500 DISTINGUISHED NAME
(CONFORMANCE TO RFC2459)

3.1.1.1 Application specific naming: ISO 17090
(TOTAL CONFORMANCE)
3.1.2 Need for names to be meaningful : YES
3.1.3 Anonymity or pseudonymity of subscribers : NO
3.1.4 Rules for interpreting various name forms: DESCRIBED IN CPS
3.1.5 Uniqueness of names: YES
  3.1.5.1 What happens when existing name reappears: APPEND ADDITIONAL LETTERS AND/OR NUMBERS AT THE END OF NAME
3.1.6 Recognition, authentication, and role of trademarks: DESCRIBED IN CPS

\[ CP_B \]

1. INTRODUCTION
   1.1 Overview
       Gives an overview of the document
   1.2 Document name and identification
       a) MUST Document name
       b) RECOMMENDED Designated identification
           Connection OR
   1.3 PKI participants
       Described in the subsections
1.3.1 Certification authorities

a) Issues certificates to end users

b) Issues certificates to other users

1.3.1.1 Root authorities

a) Specifies the difference to the CAs

1.3.2 Registration authorities

1.3.3 Subscribers

1.3.4 . . .

2. PUBLICATION AND REPOSITORY RESPONSIBILITIES

3. IDENTIFICATION AND AUTHENTICATION

3.1 Naming

3.1.1 Types of names: X.500 DISTINGUISHED NAME

(CONFORMANCE TO RFC2459)

3.1.1.1 Application specific naming: ASTM MODEL CP

SUPPORTED(TOTAL CONFORMANCE)

3.1.2 Need for names to be meaningful: YES

3.1.3 Anonymity or pseudonymity of subscribers: NO

3.1.4 Rules for interpreting various name forms:

DESCRIBED IN CPS

3.1.5 Uniqueness of names: YES

3.1.5.1 What happens when existing name

reappears: APPEND ADDITIONAL LETTERS

AND/OR NUMBERS AT THE END OF NAME

3.1.6 Recognition, authentication, and role of

trademarks: DESCRIBED IN CPS
Compatibility score of $CP_B$ against $CP_A$ is calculated. Table 6.1 shows the marking of $CP_B$ for the paragraphs 3.1.1 and 3.1.1.1. This table is used to compute partial compatibility score in the respective paragraphs of $CP_B$ against $CP_A$.

Table 6.1: Example of weighting and marking per CP Paragraph

<table>
<thead>
<tr>
<th>CP Paragraph</th>
<th>Paragraph weight</th>
<th>Values list</th>
<th>Markings</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1.1</td>
<td>90</td>
<td>a) X.500 DN(Conformance to RFC2527)</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b) Conformance to RFC822(ARPA Internet)</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>c) Internet email address</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>d) Internet email address, specific structure</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>e) URL</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td></td>
<td>f) N/A</td>
<td>0</td>
</tr>
<tr>
<td>3.1.1.1</td>
<td>90</td>
<td>a) ASTM Model CP supported(total conformance)</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td></td>
<td>b) ISO 17090 - Parts 2, 3 supported(total conformance)</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>c) N/A</td>
<td>70</td>
</tr>
</tbody>
</table>

The comparison of $CP_B$ against $CP_A$ is performed in six gradual steps:

- Definition of the CPs to be compared ($CP_A$, $CP_B$).
- Definition of the comparison problem.
- Assignment of weights and markings for the comparison.
- Definition of final acceptance rules.
- CP comparison based on weights and markings.
• Final decision concerning the CPs compatibility (accept/reject).

6.5.1 CP Comparison Based on Weights and Markings

For each paragraph \( p_i \), we extract the content value of \( CP_B \), and score it according to the markings of Table 6.1. This score shows the partial compatibility of the two CPs in paragraph \( p_i \). We multiply this scoring with the paragraph’s weight, defining in this way the weighted scoring for paragraph \( p_i \). So, for instance, let us examine again the comparison of \( CP_B \) against \( CP_A \) in paragraph 3.1.1.1. The value of \( CP_B \) in this paragraph \( v_{3.1.1.1} \) is “ASTM Model CP supported”, which according to Table 6.1 is marked with 80. This number is actually the scoring of \( CP_B \) in this specific paragraph, which defines its partial compatibility in 3.1.1.1 toward \( CP_A \). Moreover, weight of the paragraph 3.1.1.1 is 90. So, the weighted scoring for paragraph 3.1.1.1 is 80 x 90=7200.

The previous process is followed for all the CP paragraphs. At the end, the overall CP compatibility score is calculated as the addition of all the CP paragraphs’ weighted scorings divided by the maximum possible CPs compatibility score, as shown in equation (4).

6.6 Summary

A certificate authority issues a certificate to an end user with a specific certificate policy. If different end users with different domains want to establish a secure communication, they need to find a way as to how they can trust each other. That means their certificates need to follow the same standard. In case of acquiring PKIs, if the certificate policies are different, the acquired PKI should adapt to the policy of acquiring PKI. PKIs can be merged based on the final acceptance rule. The next chapter explains an efficient method developed for certificate path verification in Hierarchical PKIs.