Chapter 3

Problem Definition

3.1 Introduction

The comprehensive literature survey defined in the previous chapter emphasizes the need for robust security systems in the enterprises. The secured transmission of information in e-commerce and e-business transactions is the motto of a Public Key Infrastructure. It is important for global electronic commerce to support secure digital transactions and communications between different companies, that is, different PKIs should be interoperated. PKI interoperability is the ability to mix and match various PKI components from one vendor with those of another[135][136][137][138]. The compatibility requirement of the cryptographic algorithms affects PKI interoperability. RSA (Rivest, Shamir and Adleman) public key algorithm is the most commonly used algorithm for signature generation and verification. Most of the certification authorities use RSA algorithm for signing as well as verification.

The overwhelming research in the field of network security reveals the need for design of good trust models. Even though different trust models have been designed for this purpose, the dynamic changes in the requirements made us to think of designing newer trust models or at least make minor changes in the existing models to suit the organizational requirements. The focus is on merging
PKIs during merger and acquisition of companies. The merging method is that multiple CAs are unified into one CA. Not only the requirement of trust models, but certificate path verification or user authentication based on the digital certificates is a major concern. There are few certificate path verification methods existing currently, but they are not optimized[130]. When thousands and millions of certificates are to be verified in the business transactions, verification time is an important parameter to be considered. Even though there are few optimized methods suggested in the literature, no paper suggests the benefits of those methods through simulation results. This motivated us to devise and simulate proper methods for path verification that require less time as compared to the existing ones.

In order to merge PKIs, there should be compatibility in their Certificate Policies[7]. Thus one more issue is to devise a method to compute the compatibility score of two PKIs based on which the merging of PKIs is performed.

### 3.2 Potential Problems for Research

In electronic commerce, different PKIs need to be interoperated. During merger and acquisition of companies, cross-certification at the root level in their PKIs is the most common solution[72] for user authentication as shown in Figure 3.1.

![Figure 3.1: Merged Hierarchical PKIs with cross-certifications](image)

However, this solution has the following drawbacks:

- It increases the cost required to maintain the security of root CAs as the merged PKI has more than one root CA.
- The number of cross-certifications is also more and it depends upon the number of PKIs to be merged.

- The certificate path length is increased and the cost required to verify the certificate is also more.

One more widely used solution for merging PKIs is, introducing new CA (RCA in Figure 3.2) that certifies the root CAs of other PKIs[6].

![Diagram of Merging Hierarchical PKIs by introducing new CA](image)

Figure 3.2: Merging Hierarchical PKIs by introducing new CA

But this solution has the following drawbacks:

- Maintaining a new CA (extra CA introduced) is costly affair. The new CA in Figure 3.2 is the RCA that is hierarchically above RCA1, RCA2, ..., RCAk

- The certificate path length is increased due to introduction of the new CA and hence the certificate path verification time also increases.

One can devise methods to merge PKIs without using cross-certification. Moreover, during acquisition of companies, cross-certification at the root level is not required because, whenever a company acquires another company, the acquired company becomes a part of the acquiring company. So the Root CA of the company to
be acquired is not necessary in the future and can be discarded. Considering these facts, the methods to merge PKIs without using cross-certifications that are suitable for merger and acquisition of companies are proposed. If these methods are adopted, the certificate verification time reduces significantly as compared to the existing methods, and the maintenance cost of Root CAs is also reduced. The merging process is low-cost, easily constructed and flexible.

3.2.1 Merging Hierarchical PKIs without using Cross-Certification

Once the enterprise changes its collaborators, the multiple PKIs deployed by the enterprise and its new collaborators must be interoperated. The proposed method takes a different approach from cross-certification techniques towards merging different PKIs. The method is to unify the multiple CAs without using cross-certification. By using this method, the trust model with an efficient path processing can be built in comparison with the traditional merging methods with cross-certification. In the method, the term 'Host PKI' is used to refer to the PKI with which other PKIs merge. The term 'Guest PKI' is used to refer to the PKI which merges with the Host PKI. The term 'entity' refers to either a user or an intermediate CA. The merging of PKIs is possible only if the compatibility score of Certificate Policies of the PKIs to be merged are within the acceptance level which is explained in section 3.2.3 and chapter 7.

The proposed algorithm

1. Let \( n \) be the number of PKIs. Each PKI corresponds to one company
2. The Root CA of the Host PKI is the New Root CA (\( \text{RCA}_{\text{new}} \))
3. The Root CAs of the Guest PKIs certify the \( \text{RCA}_{\text{new}} \)
4. \( \text{RCA}_{\text{new}} \) distributes its public key to the entities of the Guest PKIs
5. \( \text{RCA}_{\text{new}} \) certifies the first level entities of all the Guest PKIs, i.e., the next
level entities of Root CAs

The overall process can be illustrated as shown in Figure 3.3 for n=2. The method can be used to merge more than two PKIs also.

![Merging Hierarchical PKIs without using cross-certification](image)

Figure 3.3: Merging Hierarchical PKIs without using cross-certification

### 3.2.2 Merging Hierarchical PKIs during acquisition of companies

The method proposed in section 3.2.1 may consider to be the best solution when the interoperability among the PKIs is temporary and dynamically change with the market requirements. However, during acquisition of companies, the acquired company becomes a part of the acquiring company. So the Root CA of the company to be acquired is not necessary in the future and can be discarded. The Root CA of the acquiring PKI becomes the New Root CA ($RCA_{new}$). It distributes its public key to the entities of the acquired PKIs. It also certifies the first level entities of the acquired PKIs. If the certificate policies of both the acquiring and acquired PKIs are different, recertification occurs in the acquired PKIs, otherwise no recertification will occur. The merged PKI is still a strict hierarchical PKI and thus the certificate path verification is also simple and straightforward.

**The proposed algorithm**

1. Let $n$ be the number of PKIs. Each PKI corresponds to one company
2. The CA of the acquiring PKI is the New Root CA (RCA_{new})
3. RCA_{new} distributes its public key to entities of all the acquired PKIs
4. RCA_{new} certifies the first level entities of all the acquired PKIs
5. If the certificate policy of all the PKIs are not same then First level intermediate CAs of all the acquired PKIs recertify the second level entities
   and so on
   Else
   The entities need not be recertified.

The overall process can be illustrated as shown in Figure 3.4 for n=2. The method can be also be used to merge more than two PKIs.

![Figure 3.4: Merging Hierarchical PKIs during acquisition of companies](image)

3.2.3 Certificate Policy comparison and assessment

Two Certificate Policies can be compared by computing the compatibility score.

Let us consider that $CP_A$ and $CP_B$ are the certificate policies of the organizations $A$
and B respectively. Let A be the comparing organization. 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 (3.1):

\[
C = \frac{\sum (s_i \ast w_i)}{\sum w_i}
\]  

(3.1)

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 summation \( \sum (s_i \ast w_i) \) 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 actual merging process depends upon the 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 80%”, “partial compatibility in paragraph X should be over 70%” etc.

It is also possible to set as final acceptance rule that the partial CP compatibility in one (or more) paragraph be 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.

### 3.2.4 Path verification in Hierarchical PKIs

A certification path is an ordered list of certificates that starts with a certificate that can be validated by one of the relying party’s trust anchors, and ends with the certificate to be validated. An optimized certificate path verification algorithm is proposed, so that, the PKI enables users in the transaction to verify their certificates in less time as compared to the existing methods. Forward certificate path verification is the most commonly used method for authenticating users in
Hierarchical PKIs. However, redundancy in path building is not avoidable in these methods. This can be avoided using a cache at the verifier’s side in the proposed algorithm.

In the proposed method, initially the certificate path is developed and stored in a local cache. Whenever, a user in the hierarchy wants to authenticate another user, first he will determine whether the path for that user certificate is already constructed and found in the cache. If the certificate path is found in the cache, no need of constructing the path once again, however each certificate in the path should be validated. If the certificate path is not found in the cache, construct the path, validate the certificates and store it in the cache.

**The Proposed algorithm**

*Retrieve the certificate of the user to be authenticated from the repository*

*If the path of the certificate is found in the local cache*

{  
  *Validate the path;*
  
  *Terminate the process;*
}

*Else*

{  
  *Construct the path*
  
  *Validate the certificate*
  
  *Store the path in the local cache*
}

### 3.2.5 Path verification in Mesh or Peer-to-Peer PKIs

Bidirectional certificate path between CAs in Mesh PKI leads to multiple paths between the users and therefore path verification becomes more complex. So it is found that converting a Mesh graph to its equivalent DFS Spanning Tree is a pos-
sible solution to avoid multiple paths and to make the certificate path construction simple. Construction of a virtual hierarchy in a Mesh PKI is one more solution to simplify the path verification process.

3.3 Summary of the research problem

- Cross-certification at the root is the most widely used solution during merger and acquisition of companies. However, When the companies dynamically change their collaborators and the merging is temporary, cross-certification is not required. Also during acquisition of companies, the root CA of the acquiring company can be made as the root CA of the merged PKIs. The root CA of the acquired PKIs is not required later. So the algorithms are to be developed for faster and cost effective interoperable PKI architectures that are suitable for merger and acquisition of companies and their efficiency is to be compared with other PKI merging solutions. The situations considered are: i) Merging Hierarchical PKIs without using cross-certification that can be used when companies dynamically change their collaborators and the merging is temporary, and ii) Merging Hierarchical PKIs without using cross-certification during acquisition of companies.

- Merging of PKIs is possible only if their Certificate Policies match. So there is a need to compare and assess the Certificate Policies before PKIs are merged. An automated method for the comparison of Certificate Policies that supports merging Hierarchical PKIs is to be presented.

- In the existing path verification algorithms, i.e., path verification by forward and reverse path construction methods in Hierarchical PKIs, there is a redundancy in path construction and hence it requires lot of time. In order to optimize the path verification, an efficient path verification algorithm in Hierarchical PKIs is to be developed and the result is to be compared with
the existing path verification algorithms[3][130].

• Certificate path verification is more complex in the case of Mesh or Peer-to-Peer PKIs because there exists multiple paths between users. Efficient path verification algorithms in Peer-to-Peer PKI are also to be developed that simplify the certificate path verification by avoiding multiple paths between the users.

In the next chapter, an efficient method to merge Hierarchical PKIs is explained. This method can be used when the companies dynamically change their collaborators and merging of companies is temporary.