

Ensuring CIA Triad Using EJBCA Solution Digital Certificate Trust Model

by

Jannatul Ferdous
19166002

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
M.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
May 2022

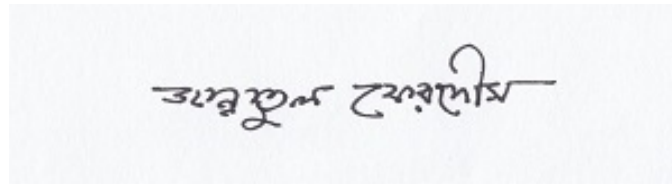
© 2022. Brac University
All rights reserved.

Declaration

It is hereby declared that

1. The project submitted is my own original work while completing degree at Brac University.
2. The project does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The project does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

A rectangular box containing a handwritten signature in black ink. The signature is written in a cursive style and appears to read 'Jannatul Ferdous'.

Jannatul Ferdous
Student ID: 19166002

Approval

The project titled “Ensuring CIA Triad Using EJBCA Solution Digital Certificate Trust Model” submitted by

1. Jannatul Ferdous (19166002)

Of Spring, 2022 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on May 24, 2022.

Examining Committee:

Supervisor:
(Member)



Dr. Muhammad Iqbal Hossain
Assistant Professor
Department of Computer Science and Engineering
BRAC University, Dhaka 1212, Bangladesh

Program Coordinator:
(Member)

Dr Amitabha Chakrabarty
Associate Professor
Department of Computer Science and Engineering
BRAC University, Dhaka 1212, Bangladesh

Head of Department:
(Chair)

Sadia Hamid Kazi
Chairperson and Associate Profess
Department of Computer Science and Engineering
Brac University

Ethics Statement (Optional)

This is optional, if you don't have an ethics statement then omit this page

Abstract

At present ensuring confidentiality, integrity and availability of information is a big challenge. Cyber-attacks such as man in the middle attack, eavesdropping, spoofing is commonly known inside threats. Leakage of information often results in devastative financial losses as well as loss of company profile. Our project has addressed the above mentioned problem. In order to address and solve this problem we have demonstrated an architecture of secure infrastructure which shall encrypt the communication between user and web application. We have implemented a Public Key Infrastructure solution which is a set of roles and policies to create, manage, distribute, use, store and revoke digital certificates and manage public-key encryption. In the absence of digital certificate, an entity will not be trusted by the web applications and thus it will not get access to the web site. The project consists of web application, an infrastructure which will generate digital certificate based on institutional requirement and algorithmic strength to ensure CIA between user and web application. To proof that CIA is ensured in our architecture we have demonstrated an exercise which we have described in this paper.

Keywords: Confidentiality, Integrity and Availability (CIA); Certificate Authority (CA); Public Key Infrastructure (PKI); Certificate Signing Request (CSR); Data Encryption Standard; Trusted Third Party

Dedication (Optional)

A dedication is the expression of friendly connection or thanks by the author towards another person. It can occupy one or multiple lines depending on its importance. You can remove this page if you want.

Acknowledgement

We are grateful to Almighty and Merciful Creator for our good health and well-being that were necessary to complete the project. We would like to express our sincere gratitude to our supervisor, Dr. Muhammad Iqbal Hossain, Professor, Department of Computer Science and Engineering, BRAC University, for his continuous support, great encouragement, patience, and immense knowledge. His affectionate guidance helped us in all the time of study and writing this project work. We could not have imagined having a better supervisor and mentor. There are several people who have been important in the completion of this dissertation, both academically and personally. This work and the time that we have spent in project work, would have been poorer without them. Besides my advisor, we would like to thank the rest of our project committee members. Last but not the least, we would like to thank our families for their unceasing encouragement, appreciable patience, support and attention. Our sincere thanks also go to our course mates for their constant support, assistance and suggestions during our thesis.

Table of Contents

Declaration	i
Approval	ii
Ethics Statement	iii
Abstract	iv
Dedication	v
Acknowledgment	vi
Table of Contents	vii
List of Figures	ix
Nomenclature	x
1 Introduction	1
1.1 Overview	1
1.2 Motivation	2
1.3 Objectives	2
1.4 Scope	3
1.5 Project Organization	3
2 Literature Review	4
2.1 Overview	4
2.2 Cryptography	4
2.2.1 Symmetric Key Cryptography	4
2.2.2 Asymmetric key Cryptography	5
2.3 Hashing	5
2.4 Public Key Infrastructure	6
2.5 Digital Certificates	7
2.5.1 X.509	7
2.5.2 Certificate Validation	8
2.6 Conclusion	8
3 Methodology	9
3.1 Registration Authority Process	9
3.2 Validation Authority Process	10

3.3	Data Communication Process	10
3.4	Tools and Materials	11
3.4.1	EJBCA Administration Preparation	11
4	Ensuring Secure Connection	17
4.1	Overview	17
4.2	Experimental Results	17
5	Conclusion	21
5.1	Conclusion	21
5.2	Future Work	21
	Bibliography	23
	Appendix A How to install L^AT_EX	24
	Appendix B Overleaf: GitHub for L^AT_EX projects	27

List of Figures

2.1	Symmetric key cryptography	5
2.2	Public key cryptography	5
2.3	Public Key Infrastructure	6
3.1	Registration Authority Process	9
3.2	Validation Authority Process	10
3.3	Secure data communication process	11
3.4	Creation of Certificate Profile for Infonet.com	12
3.5	Template Cloning	12
3.6	Edit Certificate Profile for “infonet.com”	13
3.7	Key Usage for Certificate Profile for “infonet.com”	14
3.8	Manage and Entity Profile for “webserv-infonet.com”	15
3.9	Manage and Entity Profile for “webserv-infonet.com”	15
3.10	Certificate Enrollment from a infonet	16
3.11	CA Certificates for infonet	16
4.1	Search CA infonet.com	18
4.2	View Certificate Details	18
4.3	Browse website from client machine	19
4.4	Showing connection secure	19
4.5	Valid Certificate which issued from infonet.com CA	20

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

AES Advanced Encryption Standard 1

ASP Address Resolution Protocol

CA Certificate Authority

CIA Confidentiality, Integrity and Availability

CN Common Name

CSR Certificate Signing Request

DES Data Encryption Standard

DN Domain Name

EJB Enterprise Java Beans

FQDN Fully Qualified Domain Name

HTTP Hyper-Text Transfer Protocol

IP Internet Protocol

MD5 Message-Digest Algorithm

PKI Public Key Infrastructure

RA Registration Authority

SHA256 Secure Hash Algorithms

SSL Secure Sockets Layer

TLS Transport Layer Security

TTP Trusted Third Party

Chapter 1

Introduction

1.1 Overview

Cybersecurity aims to protect the digital assets against the growing number of cyber threats. The focus of our project is to ensure Confidentiality, Integrity, and Availability (CIA) aka. CIA triad of data and services. Confidentiality ensures secrecy of sensitive information while on the network. Some security controls must protect sensitive data from being exposed and make it available only to intended users. On the other hand, integrity is the condition to preserve data in accurate and consistent manner unless authorized changes allowed. The data sent over a network should be in intact and unaltered by an authorized party. Availability refers to the accessibility of resources and services to authorized groups. PKI is a technology that strengthens security policies through communication protocols, methods and procedures so that confidential and secret information transactions can take place between different groups within and outside the organization. PKI provides security to users on an insecure network to exchange data in protected way. Personal authentication is done using an identification number (ID) like national ID, passport, driver's license etc. issued by a "trustworthy" organization. However, proving one's authenticity in a cyber-environment is a challenging task. However, it is achievable by using digital certificate, a public-key cryptography-based communication system for organizations to authenticate themselves. With the use of public key infrastructure framework, key generation and transmission can be done safely and efficiently. It is today's challenge to ensure CIA. Users who are from different domains are not supposed to have so much broader views related to information security. It is now imperative that industries adopt to a model, which is capable to meet the challenges. In this project work, threats to CIAs' are mitigated using a trust-based model, based on the use of digital certificate. A trust-based model comprises collections of well-defined rules to inform associated applications and end entities to decide the legality of a digital certificate. It indicates a binary relationship based on check of uniqueness of individual identity. The goal of a trust model is to respond to specific threat profile with the help of collection of well-defined rules. Digital certificate is like digital passport, which is both unique and unforgeable. It verifies the identity of computers, individuals or micro-organizations in a large corporate network. The certificates are now basis of trust and confidence in cyber world. The secure socket layer (SSL) / transport layer security (TLS) certificate encrypts "https://" links established between browser and the destination web applications.

The use of these certificates protect users from falling victim of man-in-the-middle attack. In order to validate digital certificates, the trust-based model has an entity known as Certifying Authority also known as 'CA'. CA, a trusted third party verifies the certificate owner's identity before certificate issuance. CA generated certificates are stored in a certificate repository. Future proof applications will then verify the authenticity of the certificate, which is in question against the certificate stored in the CR. In enterprises, PKI frameworks' implementation generates digital certificates. Roles, policies, and procedures mentioned in PKI framework are required to produce, manage, hand out, use, store, and revoke digital certificates. It has a series of servers, network protocols, hashing and encrypting algorithms, security policies, systems and applications, working together.

1.2 Motivation

The foundation of establishing CIA is trust in between users and servers, web applications. Information security is given highest priority by the industry leaders. Absence of a framework that establishes trust based relationship leave the IT infrastructure vulnerable. Confidentiality of information, integrity of data and availability of access to information only to the legit user are 11 foundation on which industry reputation and governance depends. In several cases, breach of information because of lack of trust results in organizations defamation as well as financial loss. At present, in IT enabled business model, integration between different software modules, middleware's, API libraries and gateways in between different services of different financial is very common practice. SSL, TLS keys and digital certificates are purchased from different sources at high price to secure exchange of requests responses of involved entities. External parties also manage key management framework. This is financially costly and not flexible and scalable. Solutions such as identity management solutions are complex in architecture, the scalability of resources to support such solutions are also price consuming. Possibility of single point of failure is a threat to business continuity for sensitive business environment. Considering the challenges limitations digital certificate based trust model for internal servers and users of an organization serves the purpose of conserving CIA with involving only minimal cost during implementation of private CA based on PKI. Another positive prospect is that the certificate renewal cost is zero and the certificate is available to new servers on demand. Therefore, the certificate requisition time is reduced greatly. Moreover, the greater threat in information security domain is inside threat. To response to afforested matters, in this project with the help of open source PKI framework a demonstration of an architecture to generated digital certificates is implemented and simulated an architecture that will ensure CIA using digital certificate trust model for inside environment, which offers similar level of facilities not to mention less costly.

1.3 Objectives

In order to implement critical business processes over the network, businesses require high-level certification-based security provided by the Certification Authority (CA). PKI protects online banking and trading, web-based services, digital form sign, en-

enterprise messaging, commerce etc. All the security features can be summarized as CIA triad security which are offered by PKI. CIA triad indicates three main security principals named ‘Confidentiality’, ‘Integrity’ and ‘Availability’. It is the main purpose of security management to ensure implementation of these principals within the organization deploying cost effective efficient counter measures. Controlling the way resources are accessed so they can be protected from unauthorized modification or 12 disclosure by our architecture is the goal of our project. This architecture will save company’s licensing as the model facilitates in house development with the implementation of right framework. The project work elaborately explains the implementation of open source PKI to generate digital certificate and distributing this among the communicating users and servers. In short our objectives of this project are:

1. To ensure CIA triad in IT environment with our developed architecture
2. To create, store, distribute and manage digital certificates
3. To assess vulnerabilities to test effectiveness of deployed architecture

1.4 Scope

This project involves web communication encryption between users and web applications to ensure CIA triad. Networking, Coding, Hardware level vulnerabilities are excluded from this project. Our project includes the following tasks:

1. Design and Implementing a CA
2. Planning certificate algorithms and Usages for digital certificate generation.
3. Key generation activities.
4. Define and assign trusted roles.
5. Define technical controls to encrypt and secure web communication.

1.5 Project Organization

This project embodies five chapters. Each of chapters gives distinct concept regarding the project work. 13 Chapter I (Introduction) introduces basic concepts of CIA triad, digital certificate and trust model. It also contains the motivation, goals and scope of the project.

1. Chapter II (Literature Review) deals with review of literature. It explains various concepts related to the study area.
2. Chapter III (Methodology) describes the implementation procedure of our method.
3. Chapter IV (Experimental Results and Observations) contains the expiations of the experimental results of the project.
4. Chapter V (Conclusions, Future Work and Recommendations) summarizes the project work. Discussions of the outcomes of work are in this chapter.

Chapter 2

Literature Review

2.1 Overview

The main purpose of the chapter is to clarify about cryptography, Digital Certificates, hashing, working principle of Public Key Infrastructure, the idea behind of it and its weakness, which are the most important parts to understand our project. The chapter explaining the tale of the cryptographic heart that is essential for PKI, given in Sec. 2.2. In sections Sec. 2.3, 2.4 and Sec. 2.5, the three important parts of PKI, hashing, Digital Certificates and Certificate Authorities, are explained in more details. This chapter close with Sec. 2.6, discussing the weaknesses in the current PKI.

2.2 Cryptography

Cryptography means the art of keeping secrets secret. It is a way of storing and distributing data in a certain way so that only those designed for it can read and process it [7]. Encryption is the way to secure data and information from unauthorized access and thus maintains the confidentiality. Encryption of data is possible by using either symmetric key or asymmetric key cryptography [5]. Stealth and modification of information is not possible if strong cryptography is used.

2.2.1 Symmetric Key Cryptography

In symmetric cryptography, both communication systems use the same secret key to encrypt and decrypt. When sender and receiver interacts, symmetric key cryptography ensures privacy of information. It is important that this key is confidential and only known to two communicated parties; otherwise, an attacker will decrypt and read the secret messages between parties. Important examples of symmetric key encryption schemes are 3DES and AES. In many internet, intranet and private, public cloud based applications where it can occur that any two parties communicate with each other for the first time; symmetric cryptographic technique is not sufficient, due to the key exchange problem [6] as shown by the figure 2.1 the vulnerability lies in the sharing method of secret keys between the involved clients.

Diffie and Hellman revealed the public key cryptography concept in 1976, which solved the key exchange problem of private key cryptography [1].

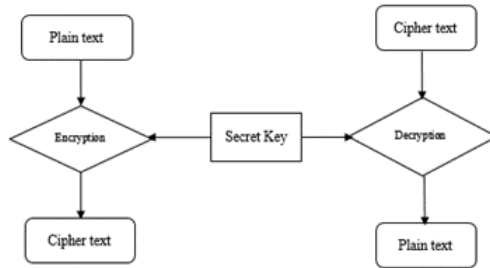


Figure 2.1: Symmetric key cryptography

2.2.2 Asymmetric key Cryptography

The public key cryptography explained here is mostly attributed to Diffie and Hellman [1], who were one of the first to give a practical example of key exchange in an insecure network, and Rivest, Shamir and Adleman [2], who followed this and gave another implementation of public key cryptography as presented figure 2.2 with the key separation between clients.

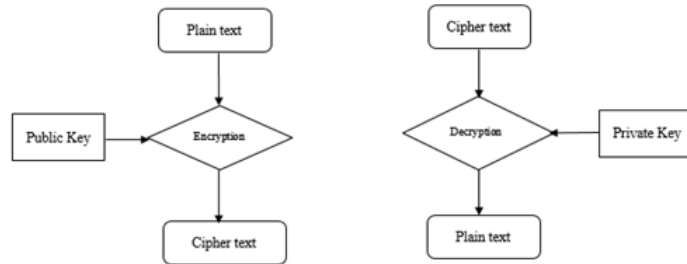


Figure 2.2: Public key cryptography

In Asymmetric key cryptography, two keys are used to encrypt and decrypt. One is private key, which is kept secret by owner used for encryption and other is public key that is used for decryption, available for everyone. There is no need to share the key between the communication partners. TLS/SSL certificate is used to produce HTTPS protocol with strong secured cryptography. Asymmetric key encryption protects transmission of data over the insecure network. Asymmetric key cryptography is used by TLS handshake to verify the identity of the original server and to exchange data, which is used to produce the session keys. RSA or Diffie-Hellman, a key exchange algorithm uses key pair to agree upon session keys[4]. After completing the handshake this, session keys encrypt communications. As a result, attackers will fail to intercept encrypted communications.

2.3 Hashing

Hashing is an encryption method that can transform any form of data into a unique text string. Regardless of size or type, any data can be hashed. Generally, no matter the size, type or length of the data, the hash generated by any data always has the same length. Hash is a one-way function that puts data into a hash algorithm and obtains a unique string. If someone encounters a new hash, they cannot decrypt the input data it represents. Some common hashing algorithm are MD5, SHA-256 etc.

2.4 Public Key Infrastructure

The Public Key Infrastructure (PKI) distributes and manages public keys and digital certificates [3]. It protects communication between sender and receiver. Growth of digital services and e-commerce enforce the development of PKI.

Certificate Authority and Registration Authority are two essential parts of PKI. A Certificate Authority (CA) verifies the digital identities of the users. Certificate Authorities block forged identities and regulate the life cycle of digital certificates. Certificate Authority empowers registration authority to register digital certificates to applicants applying for certificates. Encrypted certificate database stores all requested, accepted and revoked certificates of CA and RA.

Certificate store preserves certificate details that is usually stored on a particular workstation and act as a repository for information, related to certificates and encryption keys. It can be seen from figure.2.3 the usage of certificates starts by the request of an entity.

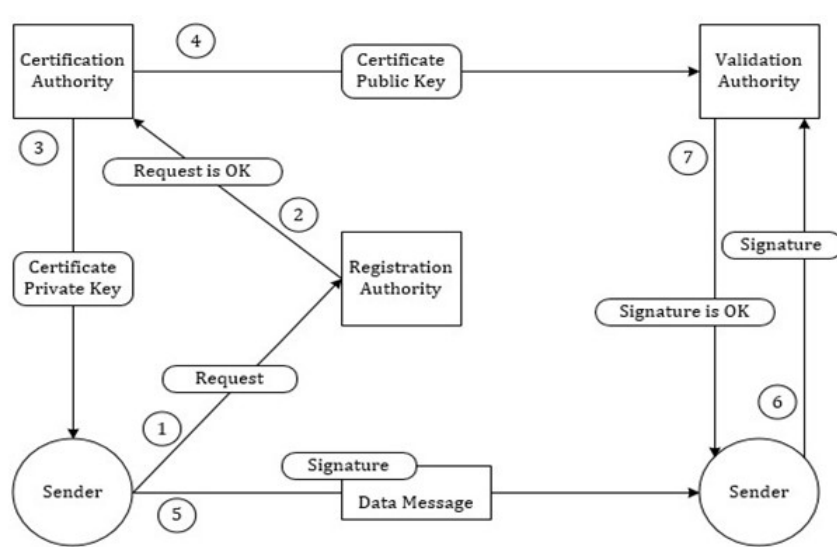


Figure 2.3: Public Key Infrastructure

The entity will generate a key pair and request a certificate within a Certificate Authority domain. The request will first be verified by the Registration Authority, which is an optional system of a CA, where the CA delegates certain management functions such as registering and checking users that have requested a certificate. After positive verification of the user's identity, the CA will issue a certificate that contains the public key, provided by the user, signed by the CA's private key. After receiving the certificate, the certificate owner will set the certificate at the corresponding domain, where Internet visitors will receive it from the domain when connecting to it. Each visitor will verify the certificate before starting a communication with the domain. The verification can be done by trusted CA's public key is known in the Internet user's client software, meaning their browser or Operating System. After successful verification, user will be able to exchange a common secret key for further encrypted communication.

2.5 Digital Certificates

Authenticity and legitimacy of website, user are verified by digital certificate. To encrypt and sign the identity of the certificate owner, digital certificate is used. Digital certificate ensures the ownership of the users certificates which is issued by CA. During transmission of internet message, the public key and the relevant information's are embedded in digital certificate. After receiving the message, the certificate is decrypted by CA's public key. Recipient will reply encrypted message through digital certificate[6].

IETF X.509 is widely used certificates standard. For the Internet. X.509 version 3 certificate format is used and the infrastructure that provides is called the PKI for X.509 (PKIX) [7].

2.5.1 X.509

The X.509 standard describes the information details of a certificate and elaborates the structure of data for it. There are 3 versions available for X.509. Since 1988, X.509 Version 1, has been available and deployed. It is the most common certificate structure. The original problem the X.509 had to solve was access control to an X.500 directory. With X.509 Version 2, issuer and subject identifiers were introduced to maintain their reuse over time.

The current version is X.509 Version 3 from 1996. This version supports extensions which is not needed for a PKI to work properly, but can be important for some organizations to have additional information within a certificate. These additional information is added in a certificate extension and can be information such as policy, usage, revocation and naming data [5].

All X.509 version certificates consist of the following data.

1. Version: X.509 used for this certificate, that affects the information can be specified in it. Still now there are three versions are defined for X.509.
2. Serial Number: Is a unique integer in every CA, assigned at the issuing time of the certificate. This is a very important element, because the serial number must be unique for every certificate and is also used when a certificate is being revoked.
3. Signature Algorithm: Identifies which algorithm is used for the signature.
4. Issuer: The entity, normally a CA that authenticated the information and offered the certificate. Using a certificate from an issuer implies that the issuer is trusted.
5. Validity Period: All certificates have a period in which they are valid. A certificate validity period can be for a few seconds or for many years. The choice for a period can be dependent on private key capability.
6. To sign the certificate client will pay to the issuer.
7. Validity-From: The commencement date of the certificate
8. Validity-To: The last valid date before expiry.

9. Subject: The owner's name in the certificate, unique across the network is called the Distinguished Name (DN).
10. Subject Public Key Info: Algorithm used to produce the public key and the public key itself is described.

2.5.2 Certificate Validation

Even though certificates have a validity period during which the certificate is valid, situations can arise when a certificate is no longer trustworthy and thus must be made prematurely invalid. This action is called the certificate revocation. There are situations possible that result in such an action. Some scenarios are the domain may not exist anymore; the owner may no longer be a customer, the private key of the owner might have compromised or changed, or the CA could be compromised [10]. Issuing CA can also revoke certificates prior to their expiration time. The CA or their RA, who also validates the credentials of the certificate requester, must initiate revocation of a certificate.

2.6 Conclusion

This project has implemented a free Public Key Infrastructure Certified Authority, EJBCA to demonstrate the proposed framework's effectiveness to preserve CIA, so discussions of some of the basic cryptography concepts as well as about Digital Certificates are done here. The triad's secret bow is about keeping data secure and confidential. Cryptography ensures confidentiality. Integrity refers to protect data from unauthorized modifications. These steps assure the accuracy and completeness of information. Integrity can be verified using a hashing algorithm. If the attacker wishes to deliberately change the unencrypted information, then a hash does not work. Non-repudiation is the assurance that someone cannot deny the validity of something. Non-repudiation, a feature of PKI provides the evidence of data origin and data integrity.

Chapter 3

Methodology

Before installing any CA or providing a certificate, it is needed to define policy that governs the operation of PKI. Policy usually reflects institutional requirements for a company. This project has already gone through the policies and agreements required by PCIDSS, ISO and other regulatory policies for cryptographic requirements and standards. In this section, the proposed PKI infrastructure will be implemented and generate a digital certificate from the implemented Certificate Authority (CA). Also, present how this PKI ensures confidentiality, Integrity and Non-repudiation.

3.1 Registration Authority Process

The root CA is an element of PKI, so the public key of the CA works as the starting point for a secure domain trust. Any user, devices, applications trusting root CA will trust certificates issued by CA hierarchy. Key pairs, followed by a CSR (Certificate Signing Request) is produced. A replica of a public key along with some information related to the subject will be embedded to CSR. CA will sign it with its (CA's) private key after verification of information. The information verification process is shown on figure 3.1.

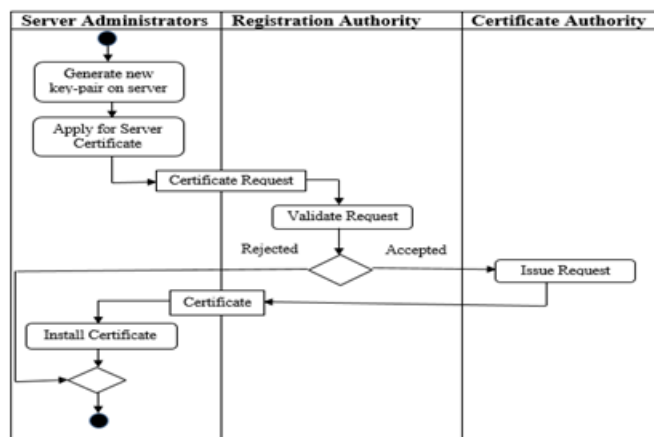


Figure 3.1: Registration Authority Process

3.2 Validation Authority Process

During the browser or system verification of user certification, the browser needs to verify the signature by obtaining the public key for the next issuing CA or central CA. This task goes on till root certificate is discovered. Browsers are bound to trust well-known root CA preserved in their trust store. Root CA highest in hierarchy is trusted by end entity its chain up information is embedded in end user's operating system, browser, device, or whatever is validating the certificate. The intermediate CA will act as liaison to certificate applicants and implementation of root CA's policies in digital certificates issued. Root CA and Intermediate CA confirms their validation by using their key pairs as well as their distinguished name and others information. Intermediate CA create signature using Root CA's public key. The connection establishment among CA's are shown on figure 3.2.

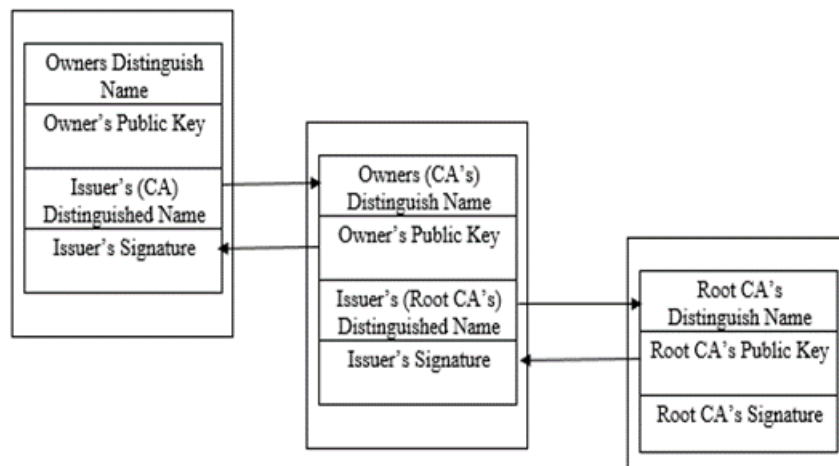


Figure 3.2: Validation Authority Process

3.3 Data Communication Process

Client checks legitimacy of certificates and certification path. Client and server both get their certificate form CA. So CA knows the public key of them. Both systems generate key pair (private key, public key) using RSA algorithm. Firstly, apply hashing algorithm (SHA-256) on client's message and this hashed message is encrypted using DES algorithm. Client's private key is used in this algorithm. So the message is turned into a digital signature. Now this signature is encrypted by using DES algorithm where server's public key is used. The secure communication process is shown on figure 3.3.

After getting this encrypted message server decrypt it using his private key. Now the signed message is found. This signed data is decrypted by user's public key. So the hash value of the recipient is found. Now again generating hash value of the received message and comparing it with the decrypted hash, it will be easily traceable that the integrity of that message is maintained or not. All public key of certified servers and clients are known to CA and they ensure non repudiation.

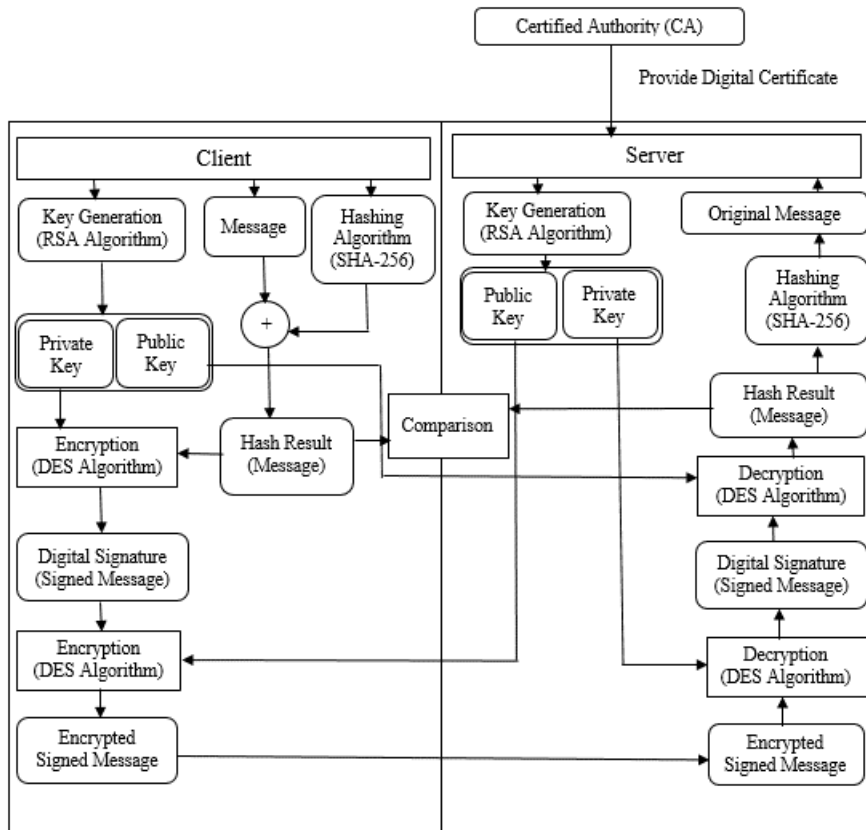


Figure 3.3: Secure data communication process

3.4 Tools and Materials

The list of tools have been implemented in order to demonstrate our project work is listed below:

1. Oracle Virtual Box (Version 6.1.12) `enumerate` environment.
2. Server (CentOS 7 for EJBCA installation)
3. EJBCA 7.4.3.2 Community version
4. JBoss AS 7.1.
5. MariaDB 10.1 (Stable)
6. 6. Server (Windows 10 Professional) that has a Web Server (IIS) installed on it.

3.4.1 EJBCA Administration Preparation

1. Enter the Administration menu from EJBCA. If so, in the CA Functions section, select Certificate Profiles. Press the Clone button next to "Rename" Server. Below on figure 3.4
2. A new page will appear. Enter the name of the new Certificate Profiles from the Certificate Profiles Server template using clone tab. As the figure 3.5



Figure 3.4: Creation of Certificate Profile for Infonet.com

named this new profile as CSR Server. When finished naming, press Create from Template.



Figure 3.5: Template Cloning

- The “webserv infonet.com” Profile has now appeared on Certificate Profiles. The next step is to edit some of the parameters in it. To do this, select the Edit button on the CSR Server. Figure 3.6 shown the process.
- As on figure 3.7 for Key Usage, check the Use and Critical sections. By default, Digital Signature and Key cypher are selected automatically. Leave it like that. In the Extended Key Usage section, select Use (without selecting Critical) and also select Server Authentication and Client Authentication.
- The next part to note is the Authority Information Access section. Check the Use section and use the OCSP Locator that is predefined in the CA. OCSP Locator is used as an address to inspect the void status of a digital certificate
- That was the Certificate Profiles section on CA Functions. Now as per figure 3.8 enter the End Entity Profiles section in RA Functions. Before adding entities, first create a Template / Profile to simplify the process going forward. Add a Profile, in this case called “webserv-infonet.com”, by typing its name then press the Add button. The profile will automatically be entered into the List of End Entity Profiles.
- Next, let’s add an entity that will be used to generate a certificate. The process is shown on figure 3.9 and described as below.



Figure 3.6: Edit Certificate Profile for “infonet.com”

- (a) Select Add Entity.
- (b) In the End Entity Profile section, select *webserv_infonet.com*.
- (c) Here enter the Username and Password according to the requirement
- (d) Enter the email address.
- (e) In this example, for Subject DN Attributes, only use CN (Common Name). Fill in accordance with the FQDN Server that need to pair the SSL Certificate.
- (f) In the Main Certificate Data, Select *webserv_infonet.com* as *Certificate Profile*. Also select the CA that
- (g) When finished, press the Add button.

8. Making CSR on the Server

9. Now Generate the Certificate which is shown on figure 3.10 and described as below. From the server where SSL Certificate is required, return to CA Server (EJBCA). The next step is to upload the CSR that was created earlier to generate the SSL certificate.

- (a) SEnter the main page of EJBCA.
- (b) In the Enroll section, select Create Certificate from CSR. A form will appear.
- (c) In the Enroll section, enter the username of the entity that created earlier in the username column and enter the password that created together with the username in the Enrollment code column.
- (d) Upload the CSR file that has created in the Request File section.
- (e) In the Result Type section, select PEM - certificate only, then press OK.

10. If successful, certificate details will appear and a pop-up menu will also appear. Download the certificate.

11. If it's already downloaded, enter the certificate into the / opt / certificate directory on the server.

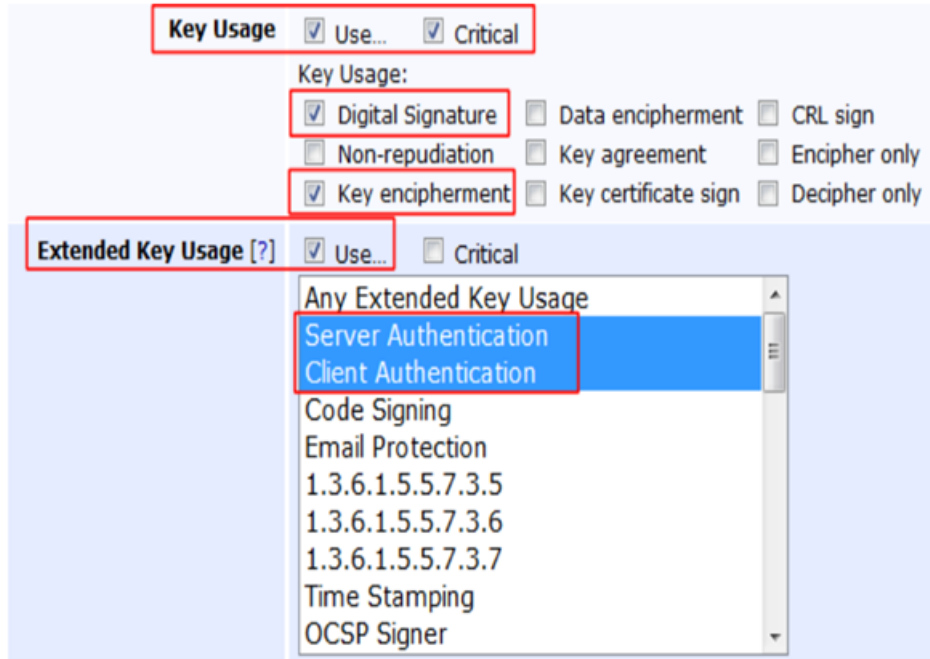


Figure 3.7: Key Usage for Certificate Profile for “infonet.com”

12. Download the Certificate Chain.
13. Next, return to the server where the SSL certificate need to install. Before entering the configuration phase, 1 more file need to downloaded, namely the CA chain certificate from the CA that issued this certificate. Figure 3.11 shown, from where the SSL certificate will be downloaded.
14. Enter the url of the CA (EJBCA) which will generate certificate. In the Retrieve section, select Fetch CA Certificates. A list of available CAs will appear. Select Download PEM Chain.



Figure 3.8: Manage and Entity Profile for “webserv-infonet.com”

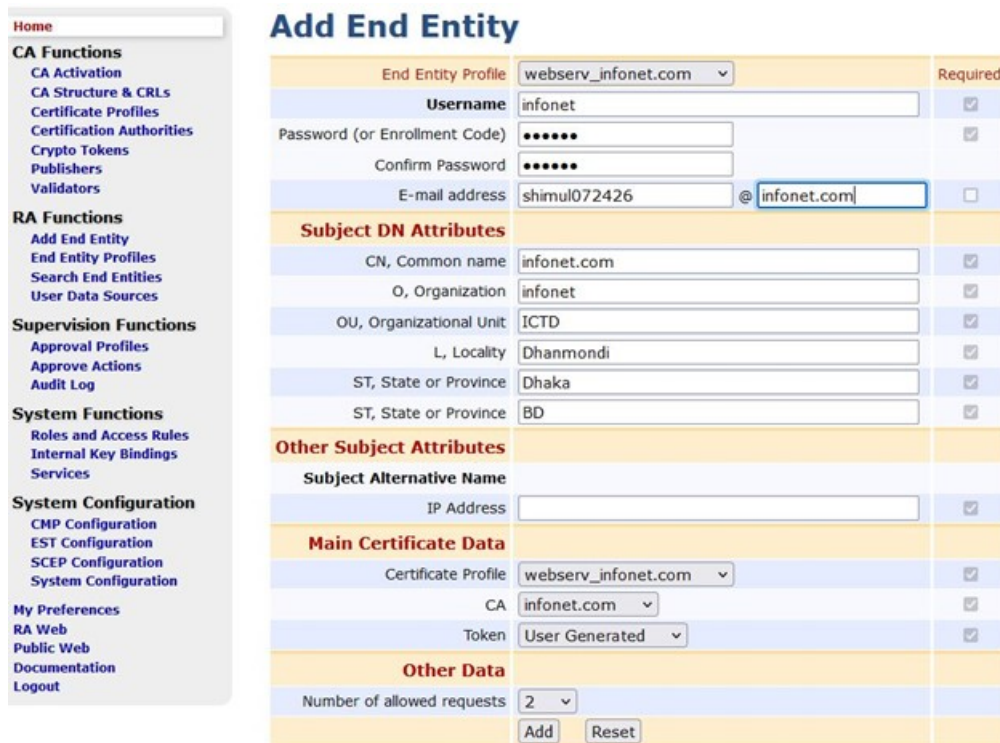


Figure 3.9: Manage and Entity Profile for “webserv-infonet.com”

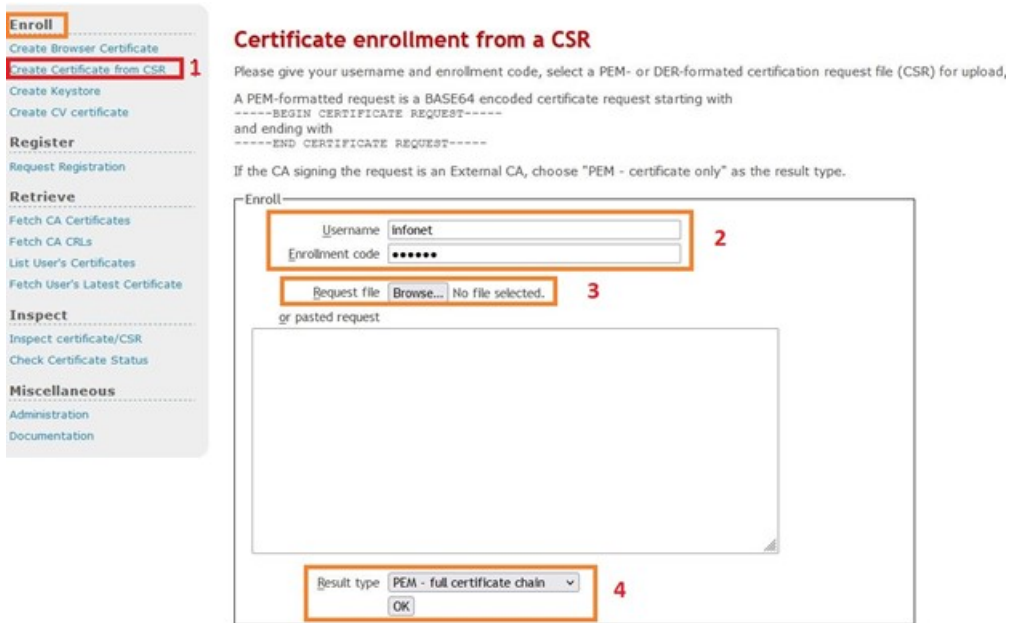


Figure 3.10: Certificate Enrollment from a infonet

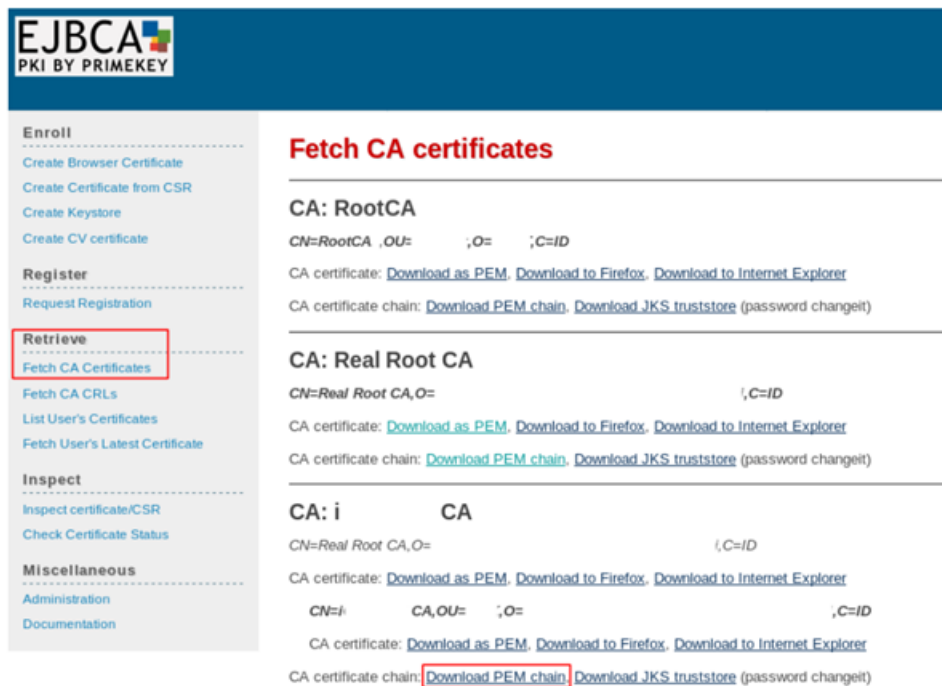


Figure 3.11: CA Certificates for infonet

Chapter 4

Ensuring Secure Connection

4.1 Overview

A security assessment involves completing an in-depth assessment to make sure that everything is up-to-date and that there are no glaring gaps in the security architecture. It helps to identify critical weaknesses in the protection measures. In this chapter, the security gaps will be of the implemented infrastructure. The system is developed using Linux Kernel, Java Enterprise Edition (JEE), JBoss, Apache Tomcat and MySQL. To evaluate or assess the generated SSL/TLS certificate, a web service has been created. Then at first an intruder or hacker has tried to intrude the web page traffic between the user and application server when there is simple HTTP protocol is used. After that an SSL certificate is installed to the webpage which is generated from infonet.com issuer.

From the figure above we can see the graph view of our model, here 0 means lost and 1 means win. So, if the predictive final value is less than 0.5 then the result would be consider as lose and if the predictive value is greater than 0.5 then it would be consider as win.

4.2 Experimental Results

The implementation process of SSL/TLS certificate is already discussed. Here it is shown, what is the benefit and outcome of using SSL certificate. When go to the option “Search End Entities” and select ALL then we can view the certificate for infonet.com in fig: 4.1.

After clicking the view we can see the certificate details from EJBCA in fig: 4.2.

Then we will browse the website from client machine which is showing in fig 4.3.

In below figure 4.4 we click the lock icon and get the message that showing connection is secure.

Get the certificate details from client machine after clicking the secure connection. It proofs that certificate is valid and our connection is secured.

Therefore, SSL / TLS is used to store sensitive information transmitted to the Internet / encrypted network so that only it is only available to the valid user. Without the implementation of SSL certificate, sensitive information such as PIN, password, social security number will be vulnerable during transmission. Because of SSL certificate, user will always use valid url link to interact with web applications and receive legitimate information which is the main goal of availability assurance.

EJBCA PKI by PrimeKey

Search End Entities Advanced Mode

Search end entity with username

Search end entity with Certificate SN (hex)

Search end entities with status **All**

Search end entities with certificates expiring within Days

Select	Username	CA	CN	OU	O (organization)	Status	Actions
<input checked="" type="checkbox"/>	infonet	infonet.com	infonet.com	ICTD	infonet	New	End Entity <input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="History"/> Certificates <input type="button" value="View"/>
<input type="checkbox"/>	superadmin	EJBCA-Management-CA	SuperAdmin			Generated	End Entity <input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="History"/> Certificates <input type="button" value="View"/>
<input type="checkbox"/>	tomcat	EJBCA-Management-CA	localhost		EJBCA POC	Generated	End Entity <input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="History"/> Certificates <input type="button" value="View"/>
<input type="checkbox"/>	webserver_test	CA_XYZ_server	webserver_test			Generated	End Entity <input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="History"/> Certificates <input type="button" value="View"/>
<input type="checkbox"/>	ixstorg	CA_XYZ_server	ixstorg	ICT		New	End Entity <input type="button" value="View"/> <input type="button" value="Edit"/> <input type="button" value="History"/> Certificates <input type="button" value="View"/>

Revocation reason:

Actions:

Figure 4.1: Search CA infonet.com

View Certificates

Username infonet

Certificate number 1 of 2

< View Older

Certificate Type/Version X.509 v.3

Certificate Serial Number 0A764C00ABA44683E95EE426E0344E9F4CB0CD74

Issuer DN CN=infonet.com

Valid from 2022-04-16 15:15:17+06:00

Valid to 2023-04-10 19:20:03+06:00

Subject DN CN=infonet.com,OU=ICTD,O=infonet,L=Dhanmondi,ST=Dhaka,ST=BD

Subject Alternative Name IPAddress=192.168.0.107

Subject Directory Attributes None

Public key RSA (2048 bits): DBA2D8D523F868482310C50E150D825A3E141C0BAFCE4E0...

Basic constraints **End Entity**

Key usage Digital Signature,Key encipherment

Extended key usage Any Extended Key Usage,Client Authentication,Server Authentication

Name constraints No

Authority Information Access OSCP Service Locator URI:
http://localhost:8080/ejbca/publicweb/status/ocsp

Qualified Certificates Statements No

Certificate Transparency SCTs No

Signature Algorithm SHA1WITHRSA

Fingerprint SHA-256 8BA9745F708E057CCD5A0392136A26A2
8018403CEB858F15E559C3268975D88

Fingerprint SHA-1 8D7A24D00151E1ED341904D58792887880F577D8

Revoked No

Download binary/to IE
Download to Firefox
Download PEM file

Figure 4.2: View Certificate Details

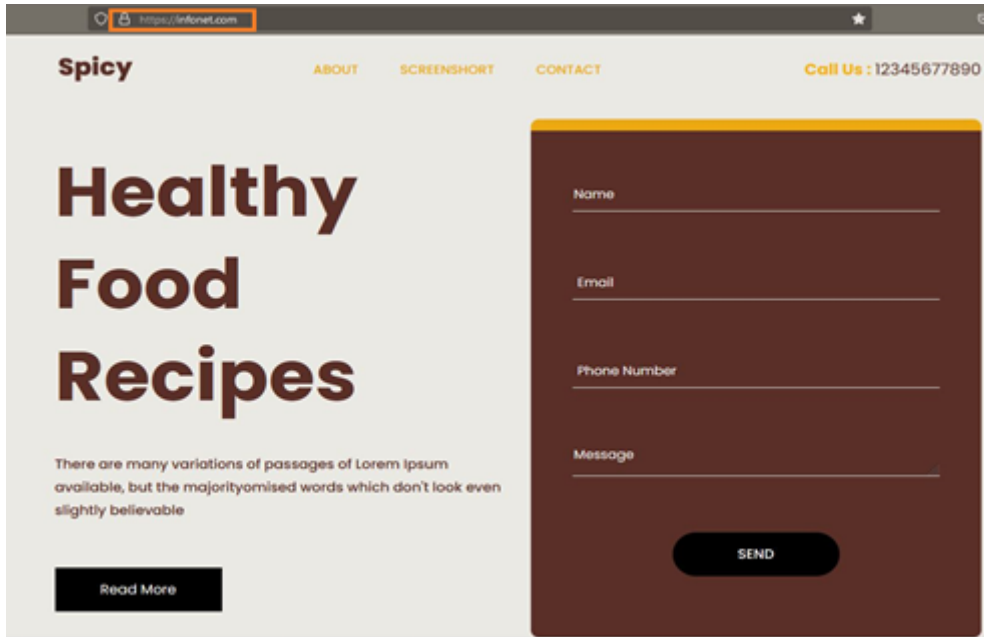


Figure 4.3: Browse website from client machine

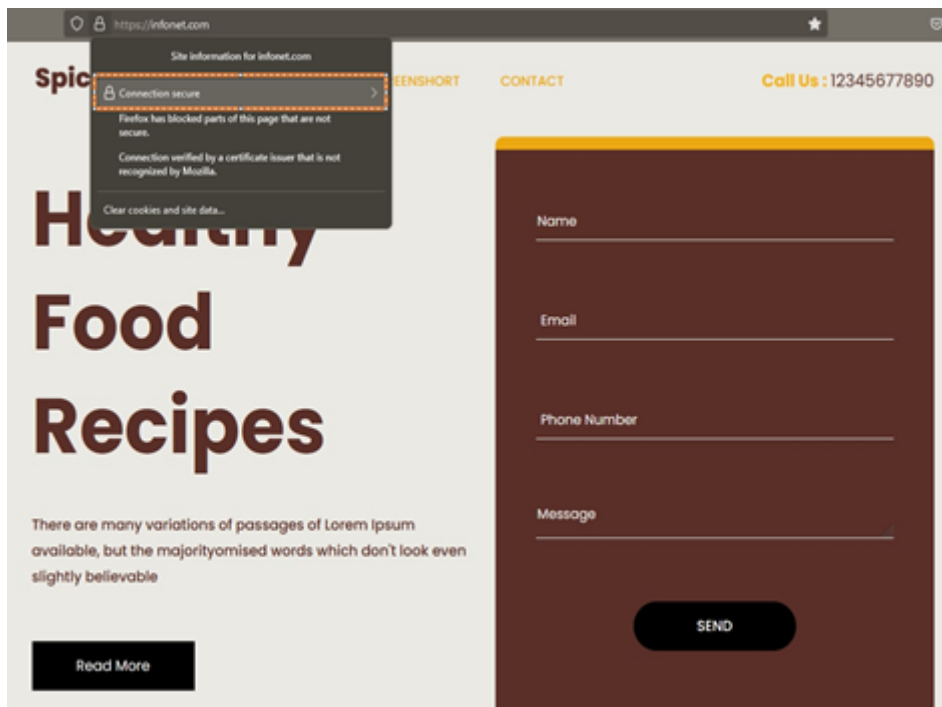


Figure 4.4: Showing connection secure

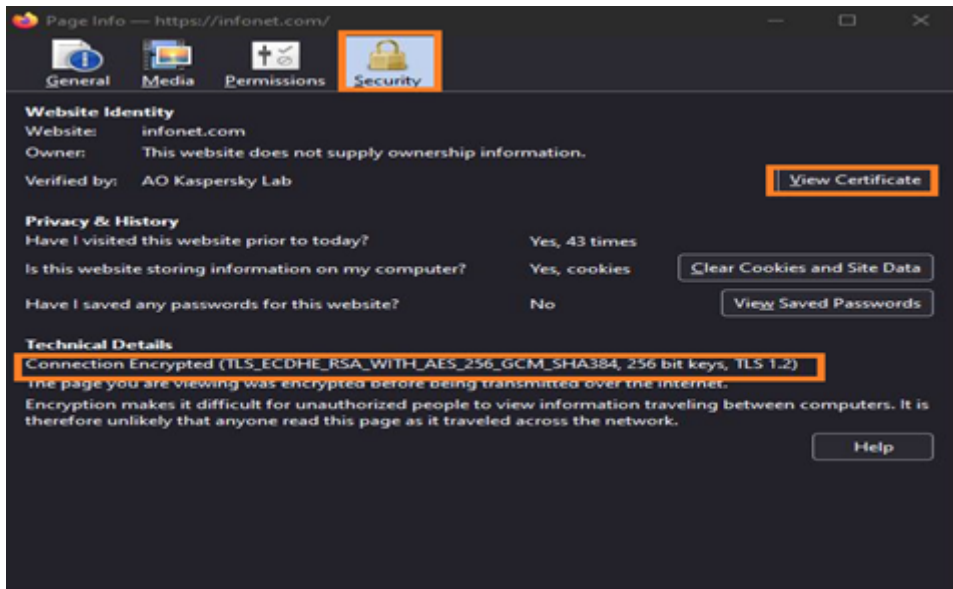


Figure 4.5: Valid Certificate which issued from infonet.com CA

Chapter 5

Conclusion

5.1 Conclusion

This project is a small plan to contribute to e-governance of institutes and e-commerce businesses in Bangladesh. It is implemented using a free or open source software to establish the public key infrastructure (PKI) based certificate authority, EJBCA for the establishment of security architecture to support variety of applications. In this paper, it is demonstrated that this open source software can be used in real world to issue and maintain legit public key certificates and specific certificate policies with low cost, even free. The benefits of using this own private CA by the free CA tool simply means that an organization is in charge of the issuing process of cryptographic pairs of private keys and public certificates. With that said, anyone can literally become their own Certificate Authority and there are no implied restrictions or authorizations necessary.

This is the one PKI software for any organization that needs to manage and operate a serious PKI. There is no cost associated with being own private CA or for internal users to be their own CA. To make it much easier this free tool EJBCA provides a graphical user interface. Application's interface makes it very easy to prepare a root or primary CA and to prepare certificates signed by the primary CA. In other words, the application makes it very easy to create own chain of trust for an organization. Browser's will show warnings if browsers do not trust certificate. But, browsers will show green colored bar for certificates from trusted private CA meaning it is safe to use, in this case, green color bar will show with certificate from infonet.com private CA. This CA will provide bird's eye view into the activity of private certificate. Using this an organization can manage all of the certificates which are used for secure communication in any environment.

5.2 Future Work

After installing the SSL certificate, the HTTPS protocol is used to ensure the secure communication between client and server. As a result, the attack for intrusion will be failed in digitally certified implemented webpage as because the communication is encrypted from end to end. To check the secured communication, we will use Ettercap solution and Wireshark tool for deep inspection of the traffic. For man in the middle attack (the attacker) "Ettercap" tool will be used to intercept the

communication of end user. “Wireshark” tool monitors the communication between the client and server.

Bibliography

- [1] W. Diffie and M. Hellman, “New directions in cryptography,” *IEEE Trans. Inf. Theor.*, *22(6):644–654*, September 1976, pp. 644–654, 1976. DOI: 10.1109/ISITIA.2018.8710889..
- [2] A. S. R. L. Rivest and L. Adleman, “A method for obtaining digital signatures and public-key cryptosystems. commun,” *ACM*, *21(2):120–126*, February 1978, 1978. DOI: 10.1145/359340.359342.
- [3] D. Park, “Social life of pki: Sociotechnical development of korean public-key infrastructure,” in *IEEE Annals of the History of Computing*, vol. 37, no. 2, pp. 59–71, 2015. DOI: 10.1109/MAHC.2015.22..
- [4] P. K. S. S. Vishwakarma and A. Sharma, “Attacks in a pki-based architecture for m-commerce,” *IEEE International Conference on Computational Intelligence Communication Technology, Ghaziabad*, pp. 52–56, 2015. DOI: 10.1109/CICT.2015.41..
- [5] “. o. P. e. B. Rajendran, “2017 international conference on public key infrastructure and its applications (pkia),” *IEEE Trans. Inf. Theor.*, *22(6):644–654*, September 1976, pp. 9–10, 2017. DOI: 10.1109/PKIA.2017.8278951..
- [6] R. R. M. D. M. Asghar and L. Pan, “A scalable and efficient pki based authentication protocol for vanet,” *2018 28th International Telecommunication Networks and Applications Conference (ITNAC), Sydney, NSW*, pp. 1–3, 2018. DOI: 10.1109/ATNAC.2018.8615224..
- [7] E. Yüce and A. A. Selçuk, “Server notaries: A complementary approach to the web pki trust model,” *IET Information Security*, vol. 12, no. 5, pp. 455–461, 2018. DOI: 10.1049/iet-ifs.2016.0611..

How to install L^AT_EX

Windows OS

TeXLive package - full version

1. Download the TeXLive ISO (2.2GB) from <https://www.tug.org/texlive/>
2. Download WinCDEmu (if you don't have a virtual drive) from <http://wincdemu.sysprogs.org/download/>
3. To install Windows CD Emulator follow the instructions at <http://wincdemu.sysprogs.org/tutorials/install/>
4. Right click the iso and mount it using the WinCDEmu as shown in <http://wincdemu.sysprogs.org/tutorials/mount/>
5. Open your virtual drive and run setup.pl

or

Basic MikTeX - T_EX distribution

1. Download Basic-MiK_TE_X(32bit or 64bit) from <http://miktex.org/download>
2. Run the installer
3. To add a new package go to Start *⇧* All Programs *⇧* MikTeX *⇧* Maintenance (Admin) and choose Package Manager
4. Select or search for packages to install

TexStudio - T_EX editor

1. Download TexStudio from <http://texstudio.sourceforge.net/#downloads>
2. Run the installer

Mac OS X

MacTeX - T_EX distribution

1. Download the file from
<https://www.tug.org/mactex/>
2. Extract and double click to run the installer. It does the entire configuration, sit back and relax.

TexStudio - T_EX editor

1. Download TexStudio from
<http://texstudio.sourceforge.net/#downloads>
2. Extract and Start

Unix/Linux

TeXLive - T_EX distribution

Getting the distribution:

1. TeXLive can be downloaded from
<http://www.tug.org/texlive/acquire-netinstall.html>.
2. TeXLive is provided by most operating system you can use (rpm,apt-get or yum) to get TeXLive distributions

Installation

1. Mount the ISO file in the mnt directory

```
mount -t iso9660 -o ro,loop,noauto /your/texlive####.iso /mnt
```

2. Install wget on your OS (use rpm, apt-get or yum install)
3. Run the installer script install-tl.

```
cd /your/download/directory  
./install-tl
```

4. Enter command 'i' for installation
5. Post-Installation configuration:
<http://www.tug.org/texlive/doc/texlive-en/texlive-en.html#x1-320003.4.1>
6. Set the path for the directory of TeXLive binaries in your .bashrc file

For 32bit OS

For Bourne-compatible shells such as bash, and using Intel x86 GNU/Linux and a default directory setup as an example, the file to edit might be

```
edit ~/.bashrc file and add following lines
PATH=/usr/local/texlive/2011/bin/i386-linux:$PATH;
export PATH
MANPATH=/usr/local/texlive/2011/texmf/doc/man:$MANPATH;
export MANPATH
INFOPATH=/usr/local/texlive/2011/texmf/doc/info:$INFOPATH;
export INFOPATH
```

For 64bit OS

```
edit ~/.bashrc file and add following lines
PATH=/usr/local/texlive/2011/bin/x86_64-linux:$PATH;
export PATH
MANPATH=/usr/local/texlive/2011/texmf/doc/man:$MANPATH;
export MANPATH
INFOPATH=/usr/local/texlive/2011/texmf/doc/info:$INFOPATH;
export INFOPATH
```

Fedora/RedHat/CentOS:

```
sudo yum install texlive
sudo yum install psutils
```

SUSE:

```
sudo zypper install texlive
```

Debian/Ubuntu:

```
sudo apt-get install texlive texlive-latex-extra
sudo apt-get install psutils
```

Overleaf: GitHub for L^AT_EX projects

This Project was developed using Overleaf(<https://www.overleaf.com/>), an online L^AT_EX editor that allows real-time collaboration and online compiling of projects to PDF format. In comparison to other L^AT_EX editors, Overleaf is a server-based application, which is accessed through a web browser.