International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 4 Issue 03, March-2015

Generation of ECC based Digital Certificates X.509 v3: an OpenSSL Code Revisited

Anu James
M.Tech student
Department of Information Technology
SRM University, Chennai, Tamil Nadu- 603203

Dharani. J
Asst. Lecturer
Department of Information Technology
SRM University, Chennai, Tamil Nadu - 603203

***Abstract*— The number of people using internet infrastructure for transactions has increased considerably. E-commerce, online banking which need confidentiality and secure lines has stretched its root to internet. So it's necessary to maintain the confidence of users. Many mechanisms are available to provide confidentiality, integrity, authenticity of data. This paper deals with the authenticity issues of information. One best mechanism to ensure the authenticity is the usage of digital signatures and certificates. The digital certificates in this paper are generated with the help of Elliptic Curves (EC) which provides security using less key length compared to all other algorithms that are available now.**

Keywords—Elliptic curve cryptography(ECC), Digital Certificates, X.509

I. INTRODUCTION

People using the internet services has increased dramatically. As the services in internet are used worldwide by everyone, security issues related to it has also popped out. Many algorithms and security devices are available which maintains the confidence of people in using these facilities. The presented digital signatures and certificates, are based on the RSA algorithm. The X.509 certificates are concerned with verifying the identity of a person or an entity.

Elliptic Curve Cryptography (ECC) is an advanced cryptographic technique. It can be integrated in all places where RSA or Diffie-Hellman algorithms can be assimilated, but instead of using integers to be ciphered, it uses points in a mathematical object called elliptic curve. The real potential of ECC is that, with a much smaller key length, it achieves the same security level as other algorithms. Therefore, ECC boons some key attributes which are prime important in scenarios where, the processing power, storage space, bandwidth and power consumption are limited [1] [2].

In this paper we develop a free open-source Certification Authority (CA) for ECC X.509v3 digital certificates. The CA we put forward is able to generate its own root certificate and to issue clients' certificates. We also develop the software a client requires to create a certificate request which the CA should sign after some verification steps. The tool we propose is mainly oriented to environments with limited resources. As it will be shown in next sections, its advantages are clearly noticeable.

The rest of this paper is organized as follows. In section II, we provide an overview about Elliptic Curve Cryptography, and the ECC mechanisms, the X.509 standard, PKI which we use for the new X.509v3 ECC digital certificates. In section III, we describe the design and working. Security analysis and discussion is done in section IV. The research paper ends with conclusion in section V.

II. LITERATURE SURVEY

A. Public Key Cryptography

Cryptography which is the process of converting the intelligible plaintext into unintelligible cipher-text, and vice versa using mathematical algorithms and processes. Certain applications of cryptography include - data encryption for confidentiality, digital signatures to provide non-repudiation and data integrity, certificates for authenticating people, applications and services, and for access control (authorization).

Two categories of cryptography exists: shared secret and public key. Shared secret cryptography is in which the sender and receiver use the same key for both encryption and decryption. For this key distribution need to be done which is not a secure option. Public key cryptography, in contrast, uses a pairs of key: a public key and a private key. Public key is widely available whereas private key is known only to the person/ application. Therefore key distribution complexity is reduced to a greater level. The private key of sender is used to produce a digital signature, an encrypted block of data which, when decrypted by the recipient, verifies the sender's identity (nonrepudiation) as well as the integrity of the data.

B. Digital Certificates

The digital certificates is a mechanism that helps in providing authentication and security to information on open networks. The applications that are using this mechanism are secure email, digital signing of software files, secure web communications, smart card authentication, and encrypting file systems. Certificates are a key building block for providing security services within an IT infrastructure, usually referred to as a public key infrastructure (PKI). The basic components of a digital certificate include:

- The name of the user/entity being certified.
- User/entity public key.
- Certification authority name.
- A digital signature.

A well-defined name need to be given for the user/entity being identified, as it provides a binding link between it and the public key.

C. X.509 Standard

The International Telecommunication Union X.509 specification [3] delivers a set of standards for the implementation of a public key infrastructure and among them, one is being used for the structure of a digital public key certificate. The X.509 certificate standard has evolved many years ago. Version 1 was introduced in 1988 and assumed that by using the issuer distinguished name of a certificate, it would be possible to build a certificate chain going back to the root certificate. Version 2 which was introduced in 1993 presented the concept of unique identifiers to allow for the re-use of issuer distinguished names. The Version 3 was introduced in 1996 and allows anyone to define an extension and include it within their certificate. Version 1 certificates are generally used as root or self-signed certificates, version 2 certificates that have been outdated by version 3 certificates are now used for most applications. Version 4 type certificates are also known as Extended Validation (EV).

D. Relation of public keys and certificates

PKI provides strong security. But a user can be given with forged public key, allowing the perpetrator to decrypt the messages and not the intended recipient. Thus, public keys need to be validated and the digital certificates does it. A certificate is a digital document which binds a public key to a person/application. A trusted Certificate Authority (CA) creates the certificate and digitally signs it using the CA's private key. Because of its role in generating certificates, the CA is the central component of the PKI. Using the CA's public key, applications/person verify the issuing CA's digital signature, and hence, the integrity of the contents of the certificate.

E. PKI Components and Functions

There are three components to a PKI:

- The Certificate Authority (CA), an entity which issues certificates. A trusted third party such as VeriSign or GTE, can provide the CA function.
- The repository for certificates and Certificate Revocation Lists (CRLs)
- A management function which is typically implemented via a management console.

In addition, there may be a separate Registration Authority (RA), an entity dedicated to user registration and accepting requests for certificates. The user registration is the process of collecting user information and verifying the credentials, which is then used to register a user according to a policy. This is distinct from the process of creating, signing, and issuing a certificate.

F. PKI Functions

The basic functions of PKI functions are issuing certificates, revoking certificates, creating and publishing CRLs, storing and retrieving certificates and CRLs, and key lifecycle management.

- Issuing certificates - The CA signs the certificate, thereby authenticating the identity of the requestor. The CA "stamps" the certificate with an expiration date.
- Revoking certificates - A certificate may become invalid before the normal expiration of its validity period due to changing the names, or by compromising the private key. Under these circumstances, the CA revokes the certificate by including the certificate's serial number on the next scheduled CRL.
- Storing and retrieving certificates and CRLs - The most common means of storing and retrieving certificates and CRLs is via a directory service, with access via LDAP. Other options include X.500 compatible directories, HTTP, FTP, and e-mail.
- Providing trust - Each public key user must have at least one public key from a CA that the user trusts implicitly. Organizations can establish and maintain trust within a single security management domain through a thorough audit of the CA's policies and procedures, repeated at regular intervals.

G. PKCS

PKCS is a series of standards casing PKI in areas of certificate enrollment and renewal, and CRL distribution. For PKI interoperability, the three most important PKCS standards are the following:

- PKCS #7 - "Cryptographic Message Syntax Standard".
- PKCS #10 also called "Certificate Request Syntax Standard".
- PKCS #12 which is "Personal Information Exchange Syntax Standard".

H. Elliptic Curve Cryptography

The Elliptic Curve Cryptosystems (ECC) operates over points on an elliptic curve. The fundamental mathematical operation in RSA and Diffie-Hellman is modular integer exponentiation. However, the core of elliptic curve arithmetic is an operation called scalar point multiplication, which computes Q = kP (a point P multiplied k times resulting in another point Q on the curve). Scalar multiplication is performed through a combination of point-additions (which add two distinct points together) and point-doublings (which add two copies of a point together). Elliptic Curve Diffie Hellman (ECDH) [4] and Elliptic Curve Digital Signature Algorithm (ECDSA) [5] are the Elliptic Curve counterparts of the Diffie-Hellman key exchange and Digital Signature Algorithm, respectively. In ECDH key agreement, two communicating parties A and B agree to use the same curve parameters. They generate their private keys, kA and kB and corresponding public keys QA = kA.G and QB = kB.G. The parties exchange their public keys. Finally each multiplies its

private key and the other's public key to arrive at a common shared secret kA.QB = kB.QA = kA.KB.G.

I. Algorithms Present in ECC

1) Elliptic Curve Digital Signature Algorithm (ECDSA)

The private key will be dA and the public key QA such that QA=dAP.

Sender's side
Signature generation procedure
a) Select a random k from [1, n-1]
b) Compute kP=(x1, y1) and r=x1mod n. if r=0 goto step 1
c) Compute e=H (m), where H is a hash function, m is the message.
d) Compute s=k-1(e+dAr) mod n. If s=0 go to step 1.
(r, s) is Alice's signature of message m. Message along with (r,s) is sent to the receiver.

Receiver's side
Signature verification is done as follows
a) Verify that r, s are in the interval [1, n-1]
b) Compute e=H (m), where H is a hash functon, m is the message.
c) Compute w=s-1mod n 4.Compute u1= ew mod n and u2= rw mod n.
d) Compute X=u1P+u2QA=(x1, y1)
e) Compute v=x1mod n
f) Accept the signature if and only if v=r.

2) Elliptic Curve Diffie-Hellman (ECDH)

The key generation is as follows:
Sender's side
a) Select a private key nA ∈ [1, n-1].
b) Calculate public key QA=nAP.

Receiver's side
a) Select a private key nB ∈ [1, n-1]
b) Calculate public key QB= nBP.

Exchange QA and QB each other. Then the key is computed as K= nAQB at sender's side and K= nBQA at receiver's side.

III. DESIGN AND WORKING

The entire design consists of mainly 3 parts- writing OpenSSL code for creating the ECC based certificates, importing the certificate to the thunderbird and establishing a trust for the created certificate, digitally signing and encryption of the e-mails sent. The algorithms that are present in ECC like ECDH, ECDSA are used in designing the system. First we have to create a certification authority (CA). Then certificates need to be generated signed by this CA. For this a code need to written which retrieves the user's information and verify it. After verification, the CA signs and issue the certificate to the defined entity/person. The certificate generated will be of *cert.pfx* format. The purpose for which the certificate is generated need to be identified prior to its creation. The questions that pops out when one tries to create the certificate is given below. This is based on X.509 standard. Fig.1. describes the fields that need to be filled during the creation.

The created certificate need to be imported to the Mozilla thunderbird and once it's imported and accepted by the trust, it will be like as the fig.2 given below. List of trust is shown in fig.3. Once the certificate and trust get installed properly, the user can use it for sending e-mail from thunderbird digitally signed. The recipient of the mail will be able to get the public key of sender and with this he/she can send the encrypted e-mail with the support of S/MIME. Figure 4 shows the symbol that we get when we send the mail digitally signed by our certificate. The thing in red circle shows the digitally signed mark. This is just a proof of concept which we have illustrated in Mozilla Thunderbird. This certificate can be integrated with many applications which require the security and can be imported to any web browsers.

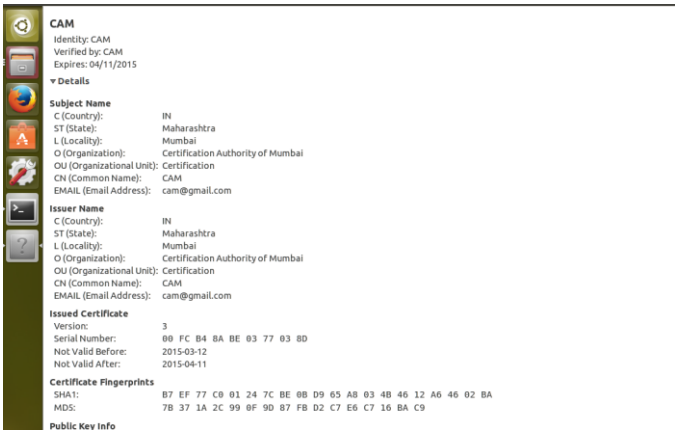

Fig.1. X.509 format certificate fields

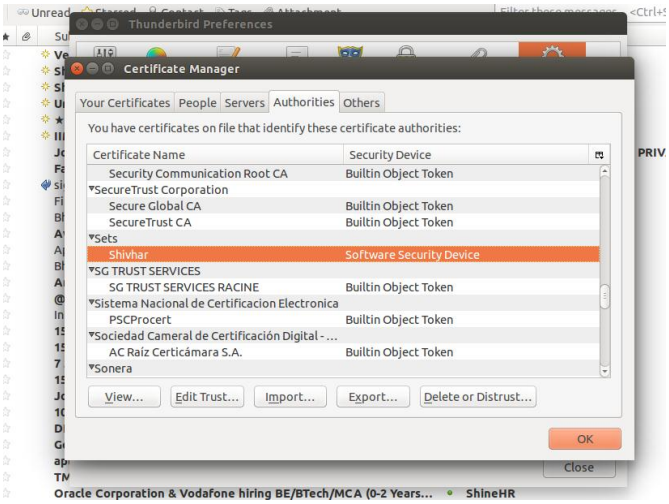

Fig.2. Imported Certificate Details

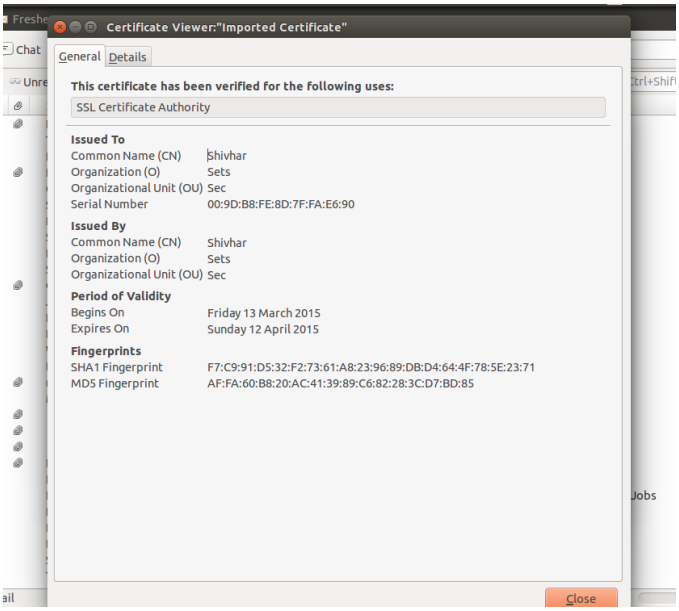

Fig.3. Trusted authorities

Fig.4. Digitally signed

IV. SECURITY ANALYSIS AND DISCUSSION

Elliptic curve is a curve that is a group. ECC utilizes this group for its functioning. Its strength is in the problem solving which involves elliptic curves; Elliptic Curve Discrete Logarithmic Problem (ECDLP). ECDLP states that given P and Q, where Q = kP, it is difficult to find k. While using a brute-force approach, it's possible to compute all multiples of P until Q is found. But finding k in such a way is infeasible. Besides the curve equation, there is an important elliptic curve parameter called the base point, G, which will be fixed for each curve. In the ECC, random integer k is kept private which forms the secret key. Corresponding public key is Q, which is got by multiplying the k with G. Not every elliptic curve offers strong security properties. A poor choice of the curve can compromise security, so the standards organizations like NIST and SECG have published a set of recommended curves [6] with well understood security properties. The use of these curves is also recommended as a means of facilitating interoperability between different implementations of a security protocol.

V. CONCLUSION

The above method suggests the use of ECC based digital certificates for those clients and servers who recommend more security. As discussed, ECC provide security with less key size compared to any other algorithms present now and hence can be easily integrated in scenarios where, the resources like processing power, storage space, power consumption and bandwidth are limited. ECC is a boon technology for the mobile devices. We believe this drift promises well for the future of Elliptic Curve Cryptography and not just for digital certificates. We have implemented ECC based certificates in OpenSSL and hence webservers and browsers can now use this variant to communicate securely.

REFERENCES

[1] N. R. Potlapally, S. Ravi, A. Raghunathan, N. K. Jha, "A study of the energy consumption characteristics of cryptographic algorithms and security protocols", IEEE Transactions on Mobile Computing, Vol. 5, No. 2, pp.128-143, 2005.

[2] W. Rao, Q. Gan, "The performance analysis of two digital signatures schemes based on secure charging protocol", Proc. International Conference on Wireless Communications, Networking, and Mobile Computing, Vol. 2, pp. 1180-1182, September 2005.

[3] ITU-T. Rec. X.509 (revised) the Directory Authentication Framework, International Telecommunication Union, Geneva, 1993

[4] ANSI X9.62, "The Elliptic Curve Digital Signature Algorithm (ECDSA)", American Bankers Association, 1999.

[5] ANSI X9.63, "Elliptic Curve Key Agreement and Key Transport Protocols", American Bankers Association, 1999.

[6] NIST, "Recommended Elliptic Curves for Federal Government Use", July 1999, see http://csrc.nist.gov/csrc/fedstandards.html. R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev.,inpress