International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 06 Issue: 04 | Apr 2019 www.irjet.net p-ISSN: 2395-0072

Preventing Fake Page from Blackhat's in Mobile Web Browsers Using Enhanced ECDSA Algorithm

P. SANGEETHA.

M.Tech Student, Dept of Information Technology, Anna University- CEG, Tamil Nadu, India.

Abstract - Today, there is an exponential growth of e-services requires the exchange of personal and sensible data such as username, password and so on over the internet. In mobile devices, many hacking techniques are emerging to break the security due to the weakness of SSL/TLS certificate, because of using ECDSA algorithm. In ECDSA algorithm, the verification process is very slow when compared to RSA algorithm. The reason why we are not using RSA algorithm in mobile devices is, computational complexity and it consumes large space due to large key size. Accordingly, some websites are hacking confidential information by introducing a fake page as same as the original page. This problem needs to be addressed in mobile environment, because of large diffusion of mobile devices such as smartphones. To prevent the occurrence of fake page in mobile devices, we proposed an enhanced ECDSA algorithm by improving the verification process.

Key Words: **Hacking techniques, SSL/TLS certificate, confidential information.**

I. INTRODUCTION

Web application security is a branch of Information Security that deals specifically with security of websites, web applications and web services. At a high level, Web application security draws on the principles of application security but applies them specifically to Internet and Web systems. It is the process of securing confidential data stored online from unauthorized access and modification. This is accomplished by enforcing stringent policy measures. Security threats can compromise the data stored by an organization is hackers with malicious intentions try to gain access to sensitive information. The process of security analysis runs parallel with Web application development. The group of programmers and developers who are responsible for code development are also responsible for the execution of various strategies, post risk analysis, and mitigation and monitoring.

A. Elliptic Curve Digital Signal Algorithm

As mentioned, the 2009 version of FIPS 186 includes a new digital signature technique based on elliptic curve cryptography, known as the Elliptic Curve Digital Signature Algorithm (ECDSA). ECDSA is enjoying increasing acceptance due to the efficiency advantage of elliptic curve cryptography, which yields security comparable to that of other schemes with a smaller key bit length. First we give a brief overview of the process involved in ECDSA. In essence, four elements are involved.

Step 1: All those participating in the digital signature scheme use the same global domain parameters, which define an elliptic curve and a point of origin on the curve.

Step 2: A signer must first generate a public, private key pair. For the private key, the signer selects a random or pseudorandom number. Using that random number and the point of origin, the signer computes another point on the elliptic curve. This is the signer's public key.

Step 3: A hash value is generated for the message to be signed. Using the private key, the domain parameters, and the hash value, a signature is generated. The signature consists of two integers, r and s.

Step 4: To verify the signature, the verifier uses as input the signer's public key, the domain parameters, and the integer s. The output is a value v that is compared to r. The signature is verified if the r= r2.

Nowadays, browsers are used for accessing many security sensitive applications like banking transaction software, social networking sites, and location-based services. Therefore, browser security is a critical issue. In mobile devices, many hacking techniques are emerging to break the security due to the weakness of SSL/TLS certificate, because of using ECDSA algorithm. ECDSA has the property that signature verification is about twice as slow as signature generation, when compared to RSA algorithm. There are some environments where 1024-bit RSA cannot be implemented, while 256-bit ECDSA can implement. Accordingly, some websites are hacking confidential information by introducing a fake as same as the original page. This problem will address in mobile devices. We propose how to prevent the occurrence of fake page in mobile devices by improving the verification process in ECDSA algorithm.

B. Scope Of The Project

The scope of this project is to use the mobile devices which have finite size cache memory for anonymous web browsing effectively and also to provide high anonymity

for web browsing. Our main scope is to improve the verification speed and to avoid the fake page.

C. Feasibility Study

A feasibility study assesses the operational, technical and economic merits of the proposed project. The feasibility study is conducted to see whether a project can be further preceded or discontinuing the project. The feasibility analysis is useful to determine final product will benefit to the users and organization. Three aspects in feasibility study are discussed below:

Technical Feasibility

Technical feasibility is intended to see that technical requirements of a project are met. A system must developed must not have a high demand on available technical resources.

The technical requirements are then compared to the technical capability of the organization. The systems project is considered technically feasible if the internal technical capability is sufficient to support the project requirements.

Economic Feasibility

Economic feasibility is intended to carry out the economic impact of system. The expenditures spend for a system should be justified. The system developed should be economically feasible because most of the technologies used are freely available.

Operational Feasibility

Since JAVA is used as a base platform to develop a project, interoperability is high and hence operational feasibility is achievable. It can be implemented in all kind of environments as stated above so the constraints are found to be less and we can implement in all LAN and WAN networks.

2. LITERATURE SURVEY

Implementation of Elliptic Curve Digital Signature Algorithm (ECDSA).

AqeelKhalique et al., [2] have discussed about which one of the variants of Elliptic Curve Cryptography (ECC) is proposed as an alternative method to establish public key systems such as Digital Signature Algorithm (DSA) and Rivest Shamir Adleman (RSA), have recently gained a lot of attention in industry and academia. The main reason for the attractiveness of ECDSA is the fact that there is no sub exponential algorithm. The key generation is highly secured and it consumes lesser bandwidth because of small key size used by the elliptic curves. Significantly smaller parameters can be used in ECDSA than in other competitive systems such as RSA and DSA but with equivalent levels of security.

Some benefits of having smaller key size include faster computation time and reduction in processing power, storage space and bandwidth. This makes ECDSA ideal for constrained environments such as pagers, PDAs, cellular phones and smart cards.

Strengthening ECDSA Verification Algorithm to be More Suitable to Mobile Networks.

Komathy K et al., [9] explains about (i) SiMul algorithm and (ii) Comb algorithm. They had attempted to reduce the processing cost of ECDSA signature verification, which becomes a mandatory solution for resource constrained mobile devices. The paper imparted few effective methods on SiMul, a proposed algorithm to reduce the time for validating the signature during execution by augmenting the pre-computation process.

SiMul, has shown significant progress in the verification process by reducing the computation Overheads but at a penalty of large storage compared to comb algorithm. The study has revealed that SiMul in mixed mode gives a better performance than in the other two co-ordinate systems especially for high-end prime curves. Further, Strengthening of ECDSA with respect to the storage of pre-computed points will be considered as an immediate future work. The objective would be to reach an optimized level between evaluation time and pre-computation time.

An Efficient Elliptic Curve Digital Signature Algorithm.

ShwetaLamba et al., [18] has represents one of the most widely used security technologies for ensuring un-forge-ability and non-repudiation of digital data. Its performance heavily depends on an operation called point multiplication. Furthermore, root cause of security breakdown of ECDSA is that it shares there points of the elliptic curve publically which makes it feasible for an adversary to gauge the private key of the signer. In this paper we proposed a new ECDSA which involves not as much of point multiplication operations as in existing ECDSA and shares only two curve points with everyone. The proposed methods also reduce the point addition and point doubling operations. It is found to be more secure in contrast to existing ECDSA.

Performance and Security of ECDSA.

Sharon Levy et al., [17] had introduced and discuss its key generation, signing and verifying procedures. Then, they had compared this algorithm to the RSA digital

signature algorithm and discuss its various advantages and drawbacks.

ECDSA has been shown to be a better alternative to both RSA and DSA for producing digital signatures. They compared the three ECDSA algorithms for key generation, signing, and verification to those of RSA. The results produced showed that ECDSA excelled with its running time in both key generation and signing but failed in verification against RSA.

The main benefits of ECDSA include the smaller key sizes that achieve the same security, making it useful when being implemented in hardware, and the hardness of breaking ECDLP, which is incorporated into the algorithm. Though there are attacks against ECDSA, like the Pollard's Rho and

Pohling – Hellman algorithms, they have running times that are much slower than RSA and DSA. The given requirements for ECDSA relating to the hash function, discrete logarithm, and number generator ensure that the statements above are true.

An Efficient Implementation of RSA Digital Signature Algorithm.

Chong Fu et al., [4] had discussed about digital signature algorithm in E-Commerce and the complexity of large integer operation in the main factor that affects the efficiency of a RSA system. In this paper, a n carry based large integer denotation approach is proposed to speed up the large integer calculation in RSA key generation and data encryption/decryption process, so as to improve the efficiency of a RSA system. The RSA digital signature algorithm and its mathematic foundation and flexibility of RSA algorithm is proved.

The random RSA public and private key pair with arbitrary length can be generated effectively by using the C++ large number library design. A 1024 bits RSA key can be generated within 2 minutes on common PC platform, while the encryption/decryption operation on data less than 1024 bits can be done within 2 seconds, the efficiency of RSA system is greatly improved, which provides important guarantees for implementation high security RSA algorithm with long keys on PC platform.

3. SYSTEM DESIGN

A. Existing System

In our existing system, the Elliptic Curve cryptography is based on the discrete logarithm problem applied to elliptic curves over a finite field. There are two fundamentally different authentication schemes: symmetric systems, which rely on secret keys shared by host and authenticator, and asymmetric systems, such as the Elliptic Curve Digital Signature Algorithm (ECDSA), which rely on a private key in the authenticator and a public key that the host uses to verify the authenticator. Scalar multiplication or point multiplication is the basic operation for ECCs. Scalar multiplication in the group of points of an elliptic curve is equivalent to exponentiation in the multiplicative group of integers modulo a fixed integer m. The scalar multiplication operation, denoted as kp, where k is an integer and p is a point on the elliptic curve, represents the addition of k copies of point p. Scalar multiplication is computed by a series of point doublings and point addition operation on the point p depending upon the bit sequence representing the scalar multiplier k. Point doubling consumes more time when compared to point addition. The time complexity is O (n log n).

The disadvantages of existing system are as follows:

- ECDSA signature verification consumes more time than signature generation due to the presence of scalar or point multiplication, because of using point doubling in scalar.

Algorithm

Input : k= (kn-1, kn-2…k1k0)2; P=(x, y);
Where, k is an random integer And
P is the base point. Output: kp;
Q = 0;
For all j= (l-1) to 0 do Q =
2Q
If =1 then Q ← Q+p; End for
Return Q

B. Proposed System

We proposed Enhanced ECDSA is less complex algorithm to calculate digital signature. This algorithm consists less number of point-addition processes which enhances its execution time. Basic concept resides is the product of k and p where point p belongs to E, it is extremely difficult to recover k from k*p. The only way to recover k from k *p is to try every possible repeated summation like p+p+p+p+……+p until the result equals value of k*p.

The loopholes of existing ECDSA are evaded in proposed algorithm by reducing the number of curve point shared publically, minimizing the number of point addition and to avoid the point doubling. The proposed algorithm is to prevent the occurrence of fake page in mobile devices by improving the verification process and mainly used to obstruct the reproduction of fake page.

Algorithm
Input : k,p **Output :**
k*p y ← 0;
z ← 0; While (p<=k)
{
y=p;
p=p+p;
z=p/2;
}
N=z+k-y;

4. SYSTEM ARCHITECTURE

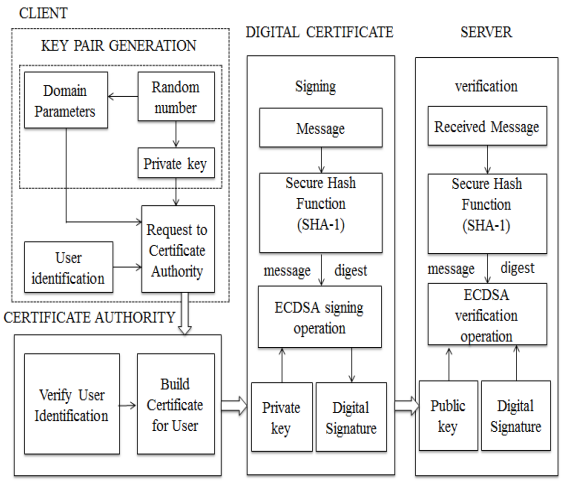

Figure 1 Architecture Diagram

SYSTEM REQUIREMENTS

A. Hardware Requirements
Processor : PENTIUM IV 2.10 GHz
RAM : 2.00 GB
Hard Disk Space : 2 GB

B. Software Requirements
Operating System : Windows 7
Technologies : JAVA
Tools : NetBeans IDE 8.0.1,
X.509 Certificate Generator

C. Non-Functional Requirements

Confidentiality:
It states that the sensitive data stored in the web application should not be exposed under any circumstances.

Integrity:
It states that the data contained in the web application is consistent and is not modified by an unauthorized user.

Availability:
It states that the web application should be accessible to the genuine user within a specified period of time depending on the request.

Non-repudiation
It states that the genuine user cannot deny modifying the data contained in the web application and that the web application can prove its identity to the genuine user.

D. TECHNOLOGIES USED

Java

Java is a Platform Independent. JAVA is an object-oriented programming language developed initially by James Gosling and colleagues at Sun Microsystems. The language, initially called Oak

(named after the oak trees outside Gosling's office), was intended to replace C++, although the feature set better resembles that of Objective C. Java NetBeans IDE is used to develop this application. The NetBeans simplifies the development of Java Swing desktop applications. NetBeans refers to both a platform framework for Java desktop applications, and an integrated development environment (IDE) for developing with Java, JavaScript and etc.

5. IMPLEMENTATION

A. List of Modules

Modules in our project are listed below:

- Key pair generation module
- Signature generation module
- Signature verification module

Key Pair Generation

Given generating point G, private key d, public key can be generated through following steps:

1) Select a random integer d in the interval [0, n-1].

2) Compute Q = d× G, obtained by point Multiplication.

Q, G is points on the elliptic curve.

3) Now key-pair is (d, Q) where d is the Private Key and Q is the Public key.

Signature Generation

Utilizing domain parameters and private key, Certificate authority generate signature for a message Z by

following steps:

1) Choose some integer K between 1 and n-1.

2) Calculate the point (X,Y)=K*G, using scalar multiplication.

3) Find r=X mod n. If r=0, return to step1.

4) Find s= (Z+ r*d)/K mod n. If s=0 return to step1.

5) The signature is the pair (r, s).

Signature Verification

Authenticity of the received message can be verified by receiver exploiting following steps:

1) Verify that r & s are between 1 & n-1.

2) Calculate w=s-1mod n.

3) Calculate u=Z*w mod n.

4) Calculate v=r*w mod n.

5) Calculate (X, Y) =uG+ vQ

6) Verify that r=X mod n.

6. CONCLUSIONS

The verification time of ECDSA algorithm is reduced by using enhanced ECDSA algorithm. Enhanced ECDSA algorithm used point addition and removes point doubling in order to speed up the verification time and also it increases the generation time. The proposed method emphasize that this change does not affect conformance to the existing ECDSA standard.

The future work may be recommended to implement this prevention technique by improving the verification process. This proposal can be further enhanced by using other methods of Elliptic Curve Cryptography such as Montgomery ladder algorithm, karatsuba algorithm and comb algorithm.

References

1.AlkaSawlikar, (2013) 'Point Multiplication Methods for Elliptic Curve Cryptography',
International Journal of Engineering and Innovative Technology (IJEIT), Vol.1, pp.38-45.

2.AqeelKhalique, Kuldipsingh, Sandeepsood, (2010)'Implementation of Elliptic Curve Digital Signature Algorithm', International Journal of Computer Applications, Vol.2, pp.21-27.

3.Bos, Joppe, W., (2016) 'Selecting Elliptic Curves for Cryptography: An Efficiency and Security
Analysis', Journal of Cryptographic Engineering (JCE), Vol.6, pp.259-286.

4.Chong Fu, Zhi-liangzhu, (2008) 'An Efficient implementation of RSA Digital Signature Algorithm', International Conference on Intelligent Computation Technology and Automation (ICICTA), Vol.15, pp.1-4.

5.Chunming RongandGeng Yang,(2003) Honeypots in Blackhat Mode and its Implications',International Journal of Research in Information Technology, Vol.4, pp.185-188.

6. Galperin, S. and Santesson, S., (2013) 'X.509 Internet Public Key Infrastructure', Online Certificate Status Protocol (OCSP), Vol.7, pp.51-65.

7.GayosoMartinex, V., HernandexEncinas, L. (2013) 'Implementing ECC with Java Standard Edition 7', International Journal of Computer
Networks and Communication, Vol.3, pp.134-142.

8. Huili, Ruixiazhang, (2013) 'A Novel Algorithm for Scalar Multiplication in ECDSA', Fifth International Conference on Computational and Information Science (ICCIS), Vol.41, pp.943-946.

9. Komathy, K., Narayanasamy, P., (2006) 'Strengthening ECDSA Verification Algorithm to be more Suitable to Mobile Network', Sixth International Multi-conference on Computing in the Global Information Technology, Vol.6, pp.61-61.

10. ManojkumarChande, Cheng-chitee, (2016) 'An Improvement of a Elliptic Curve Digital Signature Algorithm', International Journal of Internet Technology and Secured Transaction, Vol.6, pp.219-230.

11.Moumita Roy, Nabamita Deb and Amar Jyoti Kumar, (2013) 'Point Generation and Base Point Selection in ECC', International Journal of Advanced Research in Computer and Communication Engineering, Vol.3, pp.31-37.

12.Ning Zhang, Xiaotong Fu (2013) 'Ternary Method in Elliptic Curve Multiplication', fifth International Conference on Intelligent Networking & collaborative Systems, Vol.8, pp.490-494.

13.Parvez Faruki, Ammar Bharmal (2015) 'A Survey of Issues, Malware Penetration and Defences', IEEE communications survey, Vol.17, pp.998-1022.

14.Ravi Kishore Kodali and Harpreet Singh Budwal,(2013) 'High Performance Scalar Multiplication for ECC', International Conference on Computer Communication and Informatics (ICCCI), Vol.6, pp.1-4.

15.Robert Gallant, P., Robert Lambert, J. and Scott Vanstone, A., (2011) 'Faster Point Multiplication on Elliptic Curves with Efficient Endomorphisms', Vol.7, pp.190-210.

16.Sami Nagar, A., Saad Asshamma, (2012) 'High Speed Implementation of RSA algorithm with Modified Keys Exchange', 6th International Conference on Science of Electronics Technologies of Information and Telecommunication (SETIT), Vol.8, pp.639-642.

17.Sharon Levy, (2006) 'Performance and Security of ECDSA IEEE Computer Society', International Journal of Advanced Research in Computer and Communication Engineering, Vol.82, pp.53-57.

18.Shweta Lamba and Monika sharma, (2013) 'An Efficient elliptic curve digital signature algorithm (ECDSA)', International Conference on Machine Intelligence and Research Advancement (ICMIRA), Vol.11, pp.179-183.