Public Key Infrastructure (PKI) understanding for VxWorks RTOS using Asymmetric Encryption Algorithms and providing the efficient solution for Trusted Platform Module(TPM)

Pooja Laxman Khyadgi[1], Prof. Siddharth K. Gaikwad[2]

[1]Information Security Department, College of Engineering, Pune , Pune, India
[2]Computer Engineering Department, College of Engineering, Pune, Pune, India

---***---

Abstract –*Security is becoming the buzzword; we all are more concerned in safeguarding our data. Cryptography, the art of writing is mainly used for data security. Especially in Real Time Operating System, the data plays an important role. A very famous Wind River's VxWorks RTOS supports cryptography and provides built in libraries for the same. VxWorks use the trusted platform Modules (TPMs) to secure the sensitive data and keys. TPMs uses the asymmetric algorithm in order to save guard the data. Internally in TPMs RSA algorithm is used to generate the keys or to sign digitally when required. We in this paper are studying the behavior of RSA algorithm and Elliptic Curve Cryptography (ECC) the asymmetric encryption algorithms on the TPMs and conclude to the use of specific algorithm on the TPMs, which increase the efficiency of the RTOS ultimately.*

Key Words: Public Key Infrastructure, Real Time Operating System, Wind River VxWorks Workbench 4.0, Trusted Platform Module, RSA, Elliptic Curve Cryptography

1. INTRODUCTION

In this world of high connectivity, security has become the extreme priority task. Especially in the field of Real Time Operating System (RTOS), compromising with security will lead to a great loss. All the RTOS should satisfy the features or characteristics like reliability, predictability, performance, scalability and compactness. PKI defines the polices to adopt such features in the system.

Thus in Wind River VxWorks RTOS PKI plays, an important role to secure the data form the intruders. VxWorks uses the TPMs modules to perform the root level encryption or for digitally signing the documents etc. With the increasing connectivity and processing power, simple and traditional algorithms are required to be updated or replaced. RSA is having many shortcomings when compared with ECC the asymmetric encryption algorithm. We are comparatively studying both the asymmetric encryption algorithms to choose the more efficient algorithm for the TPMs amongst them which can increase overall efficiency of the system in the PIK aspect. We present the results of analysis between the RSA and ECC encryption algorithm on the Wind River's VxWorks Workbench 4.0.

This paper is organized as follows: section 1 deals with the detailed introduction of the PKI and Wind River VxWorks RTOS; section 2 explains the Asymmetric Encryption Algorithm with sub headings of RSA and ECC encryption algorithms; section 3 deals with the implementation results; finally, section 4 provides the conclusion followed by the acknowledgement and the references.

1.1 Public Key Infrastructure (PKI)

PKI handles the possible polices roles, rules, encryption, digital certificates etc. in any of the operating system. PKI is important to be clearly defined and implemented in order to achieve total integrity, authentication and reliability in any specific system. Many times PKI assure the secure transfer of electronic data on the network by using the third party verification. In VxWorks RTOS the PKI is achieved through possible ways, which are mentioned below

i) Secure Boot & Secure Run time loader: Allow Authenticated (signed) binaries to run and decrypt and verify digital signatures

ii) Trusted Platform Modules(TPMs): Trusted Platform Module (TPMs) actually are the small crypto processors that can be used for performing various tasks such as providing a secure storage for true secure boot with full hardware root of trust on the VxWorks system.

iii) Secure ELF (Digitally signed applications)

   a. Encrypted Container: Uses AES algorithm for storing the data at rest in a secure way.

   b. Full disk encryption: Uses XEX-AES—to ensure that even the file system metadata is invisible when it is performed at the partition level.

iv) Open SSL toolkit: Provides the full features for the Transport Layer Security (TLS) and Secure Socket Layer (SSL) protocols. It also includes the libraries for the cryptographic standards like AES, RSA, ECC, x.509 certificates.

VxWorks also provides security features as in basic user management by login credentials, SSL/SSH client/server technology, firewalls, IPsec authentication etc. [1]

In the TPMs, the current algorithm used for encryption is RSA. Below is the diagram depicting the scenario of the TPM securing the data.

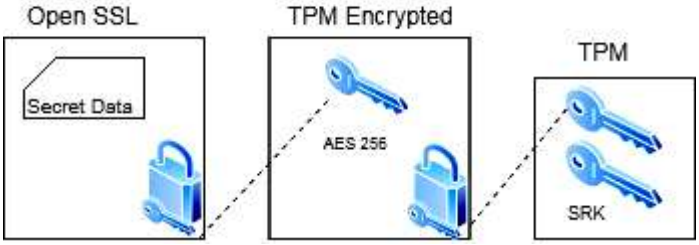


Fig 1. TPM using the RSA algorithm

Thus, we are able to see that the specific top-secret data is encrypted using Open SLL and its AES 256-bit key is wrapped by TPM using the RSA public key in the TPM Encrypted block, which can be decrypted only by the private key of the RSA. Further the third level of security is provided as the key produced by this TPM encrypted is secured by the root key, which is called as Storage Root Key (SRK) and is stored permanently inside the TPM.

1.2 Wind River VxWorks RTOS

VxWorks is one of the RTOS used in many embedded devices. VxWorks RTOS is the industry's leading device software operating system. VxWorks was produced by Wind River Systems in 1981 and now owned by Intel. VxWorks is characterized by its determinism, performance, scalability, and footprint which make more than 300 million devices worldwide run faster and more reliably. VxWorks is uniquely applied to industries like aerospace and defense, industrial, medical, consumer, and networking. Thus, VxWorks has become the favorite choice of many leading innovators like NASA, Boeing, Euro copter and many.

Many versions of VxWorks are begin released and the latest version is VxWorks 7 released on 7 March 2014. It is famous and widely used because of its many features like broad support, expandable and upgradeable, scalable, deterministic, reliable, optimized etc. [1] Optional Add-on profiles also are available for the users. VxWorks is componentized into core, middleware, network technologies, and the Wind River Workbench development suite. VxWorks support many types of schedulers for multitasking. Reliability is ensured in VxWorks by separating the kernel mode and user mode executions.

VxWorks also provides features as in IPNET stack, VxWorks Simulator, System build, Debug shell etc. VxWorks features can be separated into layers like application, middleware, OS services, VxWorks core services, VxWorks Kernel. Board support package is the fundamental building block of the Entire system. Wind River even provides the feature of Hypervisor to be implemented. Applications layer mostly includes Wind River Network Acceleration Platform, Wind River Tilcon GUI Engine, Software Defined Radio, JVM, and Ada. Middleware layer consists of Web server, Web services, OPC, fieldbus, Ipsec, etc.

Networking, Graphics, Error Detection and Reporting, Debug and instrumentation services are included in the OS Services. VxWorks Core OS includes file systems, I/O systems, Drivers and Driver management. In VxWorks Kernel layer functions included are related to Multi-core, interrupts, exception, timers, semaphore, mutex, processors etc. Thus the multi featured VxWorks is the good choice for the RTOS applications.

2. Asymmetric Key encryption

Two different keys are used for encryption and decryption- Public and Private in Asymmetric Key encryption. The public key is meant for encrypting and so it is available to anyone on the network. Anyone who knows the Public Key of receiver can encrypt the plaintext. Only the authorized person is able to decrypt the cipher text through his own private key. Private Key is kept secret from the outside world. Asymmetric cryptography is also called as the public key cryptography.

There are various asymmetric key algorithms such as Diffie-Hellman, RSA, ECC, ElGamal, DSA etc. In this reference, we have used the RSA and ECC algorithms for our study as after the literature survey, between all the algorithms compared with RSA, ECC proves to be more efficient. Before its execution, let us understand these algorithms.

2.1 RSA

In 1977, Ron Rivest, Adi Shamir and Leonard Adleman introduced a cryptographic algorithm, RSA, which is named for the first letter in each of its inventors. [4] RSA can be performed with different key sizes. RSA has its many applications in various domains like security healthcare, cloud, image processing etc. Here we need not to perform the security key transfer as we do this in the symmetric algorithms.

Here are the steps of RSA Algorithm [2]:

- The very first step of RSA Algorithm is selecting two dissimilar prime number as a & b.
- In second step, N is calculated as N=a*b
- In the third step calculation of $\varphi(N)=(a-1) *(b-1)$ is performed.
- In the fourth step, e an integer is selected as a public-key and it is co-prime with $\varphi(N)$
- Finally, the inverse of e modulus φN) is taken to produce d, the private-key.

For RSA Algorithm, the public-key involves two numbers N and e while N forms the private-key together with a different number d.

To encrypt the message: M (plain text) formula used is

M→Me(mod N)=C

To decrypt the message: C (cipher text) formula used is

C→Cd(mod N)=M

RSA uses two prime numbers (i.e a & b) to generate the public and private keys. Encryption and Decryption is done using these two different keys. The sender can encrypt the message using the receiver's public key and when the message is transmitted to receiver, and then receiver can decrypt it using his own private key.

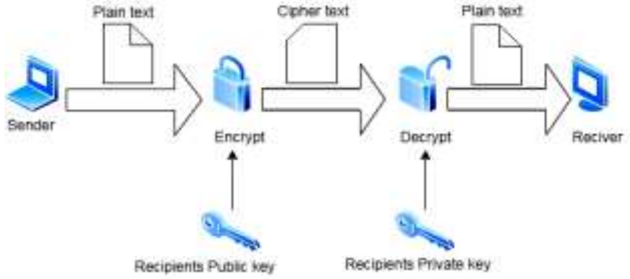


Fig 2. RSA algorithm's main Stages

The above diagram depicts the working of the RSA algorithms and the usage of its keys. The Public key is used while Encryption and the Private Key is used for the Decryption purpose as depicted in the diagram.

Choosing the key size in RSA encryption plays a great importance role. As the size of the key increases, the security level of the system, the complexity and the resistance of encrypted text also increases automatically. RSA Algorithm for cryptography consists of mainly three stages: Key Generation Stage, Encryption Stage and Decryption Stage.

2.2 Elliptic Curve Cryptography (ECC)

ECC is a public key encryption technique based on elliptic curve theory. In 1985, first time the use of elliptic curves in cryptography was suggested independently by Neal Koblitz and Victor S. Miller. This theory can be used to create faster, smaller, and more efficient cryptographic keys. ECC perfectly operates on groups of points over elliptic curves and derives its security from the hardness of the elliptic curve discrete logarithm problem (ECDLP) [3]. Applications needing the long-term security requirements mainly use ECC as it well suites it.

The mathematical operations of ECC is defined over the Elliptic curve $y2$ □□$x3$ □□ax□b , where $4a3$ □□$27b2$ □□□0 .

For each value of the 'a' and 'b' gives a different elliptic curve.

All sets of points (x, y) which satisfies the above equation plus a point at infinity lies on the elliptic curve.

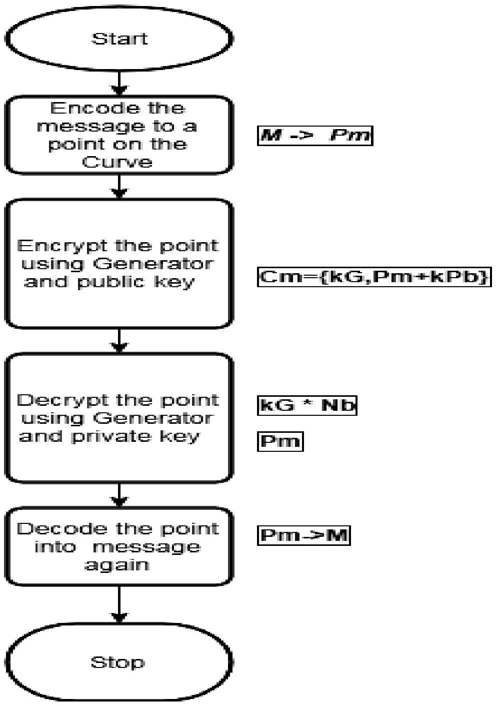

Start
Encode the message to a point on the Curve
M -> Pm
Encrypt the point using Generator and public key
Cm={kG,Pm+kPb}
Decrypt the point using Generator and private key
kG * Nb
Pm
Decode the point into message again
Pm->M
Stop

Fig 3. ECC algorithm Flow Chart

Above is shown the flow chart for the ECC algorithm, which explains its main four steps as encode, encrypt, decrypt, and finally decode. In the flow chart the message is denoted by M. The public key is considered as a point in the curve Pm and a random number is the private key k. The public key is obtained by multiplying the private key with the generator point G in the curve P. The Cipher text Cm is the pair of two points (kG, Pm+kPb), where Pb is the public key of the receiver. In decryption mode the first step is multiplying the first point with the secret key of the receiver. Second step is subtracting this from the second point to obtain the point on the curve. The last step is decoding the message from the achieved point.

Thus the ECC algorithm makes use of Elliptical curve to perform its encryption and decryption. It is also used in other applications as Digital signature, Key transfer etc.

3. Implementation Results

We have performed the test on the Wind River VxWorks Workbench 4.0 with the Core Version: 1.7.0.0-wb4_20160311. The configuration of the machine is 64-bit Window's 7 OS with 16 GB Ram. Below are the details of the VxWorks workbench used.

VxWorks Workbench 4.0 specifications:

This workbench is a development suite with Eclipse based integrated environment. It consists to build system, debugger, system analysis tools etc. There are two compilers available as listed below

a. LLVM for arm and Intel Architecture

b. GCC for Power PC Architecture

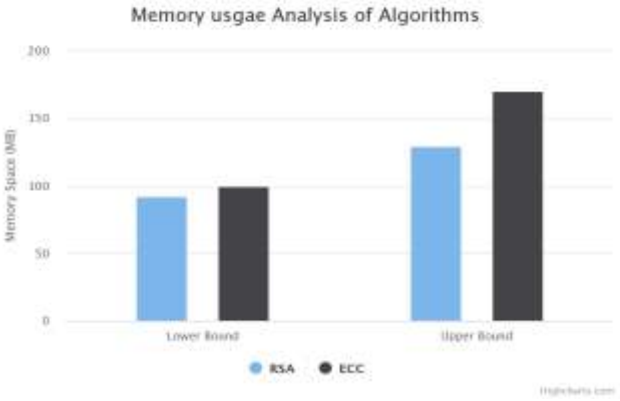
Many in build features like simulator, scope tools and sources code analyzer, which help the developer to find out the best results possible for them. Memory Analyzer, CPU Profiler are the features provide to the user, which are helpful for analysis. This workbench supports the Architectures like ARM, POWER Architecture, and Intel, Others on Demand. We have the debug shell, Kernel shell, and core dump as options for building the codes as well

Below is the table showing up the details of the algorithms when individually executed on the workbench with the VxWorks simulator in the connected status.

Performance Analysis of both the Algorithms			
Algorithms	Memory Space	CPU utilization	Execution Time
RSA	92 -120 MB	43 -50 %	600-675 ms
ECC	100-170 MB	36-48 %	500- 600 ms

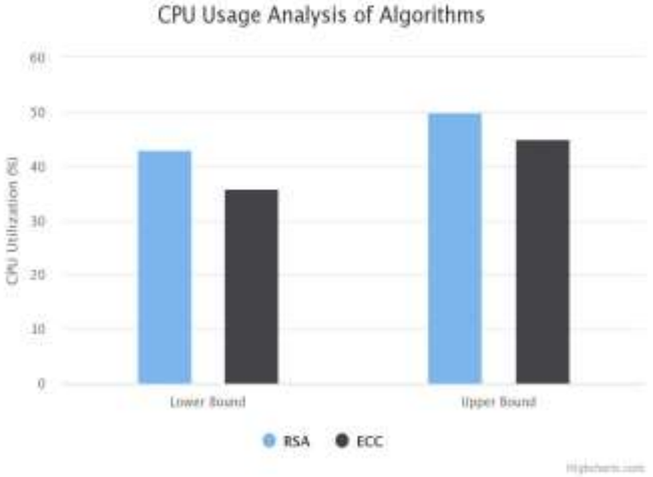
Table 1. Performance Analysis

Above table can be clearly understood by graphical representation shown below.



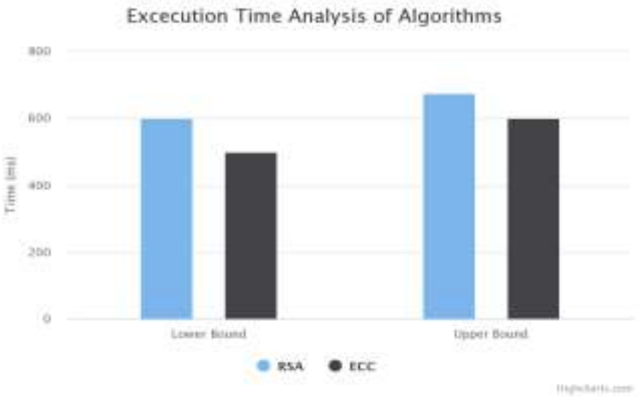
Graph I. Memory utilization

From the above graph, it is evident that the memory utilization of ECC is less than that compared with RSA.



Graph II. CPU Utilization

CPU utilization for both the algorithm is almost near to each other's.



Graph III. Execution Time Analysis

From the above graphs, we can conclude that RSA is taking more time to complete its execution as compared to ECC. The Time taken by RSA is almost double that of the time taken by ECC.

3.1 Replacing the RSA by ECC to improve the efficiency of the TPM.

In RTOS with small memory space, the use of large key size for encryption is not preferable as the RSA produces large keys. The 256 bit ECC keys are equivalent to the 3072 bit RSA keys. Also RSA is vulnerable to the brute force attack and quantum computers. Scalability is the issue with RSA algorithm. Thus after the actual performance we understood that ECC is better in case of time and CPU utilization of the system. Compared to RSA, ECC offers smaller key sizes, faster calculations, saving of memory, energy and bandwidth and hence is better suited for small device especially RTOS. [4]

As in order to store the sensitive information, we are depending on the RSA algorithm inside the TPMs of VxWorks RTOS. However, after our study we can say that using the ECC Algorithm instead of RSA would prove to be beneficial to the entire RTOS in many aspects.

4. CONCLUSION

In this paper, we have presented the analysis after implementation of the RSA and ECC algorithms on the Wind Rivers VxWorks Workbench 4.0. From the analysis of these asymmetric encryption algorithms, ECC is useful in TPMs to increase the efficiency of the system as it takes less space, less memory and less CPU resources benefiting the entire RTOS. Thus after the actual comparison of the RSA and ECC algorithm with different parameters, we can make our conclusions, that for the TPMs of the VxWorks RSA algorithms can be replaced with ECC Algorithm.

ACKNOWLEDGEMENT

I hereby acknowledge industry guide Mr. Ramakrishnan, Ganesh for providing the guidance and support to complete this project.

REFERENCES

[1] VxWorks: The Safe and Secure RTOS for the IOT by WIND River Publications

[2] Ferdi SÖNMEZ and Mohammed Khudhair Abbas, “Development Of A Client / Server Cryptography-Based Secure Messaging System Using RSA Algorithm,” Journal of Management Engineering and Information Technology (JMEIT), Dec-2017.

[3] Padma Bh1, D.Chandravathi2 , P.Prapoorna Roja3, “Encoding And Decoding of a Message in the Implementation of Elliptic Curve Cryptography using Koblitz’s Method,” (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 05, 2010.

[4] Nils Gura, Arun Patel, Arvinderpal Wander, Hans Eberle, *and Sheueling Chang Shantz, “Comparing Elliptic Curve Cryptography and RSA on 8-Bit CPUs, ” International Association for Cryptologic Research 2004

[5] Darrel Hankerson, Julio Lopez Hernandez, Alfred Menezes, Software Implementation of Elliptic Curve Cryptography ver Binary Fields, 2000 Available at http://citeseer.ist.psu.edu/hankerson00software.html

[6] Moncef Amara and Amar Siad,” Elliptical Cryptography and its Applications,”2011 7th International Workshop on Systems, Signal Processing and their Applications (WOSSPA)

[7] Maryam Savari, Mohammad Montazerolzohour, Yeoh Eng Thiam,” Comparison of ECC and RSA Algorithm in Multipurpose Smart Card Application”.