International Journal of Innovative Research and Advanced Studies (IJIRAS)
Volume 2 Issue 1, January 2015

ISSN: 2394-4404

Key Management Life-Cycle

Walunjkar Priyanka

Chavan Madhuri

Gadekar Diksha

Dnyaneshwar Mule

Pawashe Monika

Students, Computer Engg. Dept., Government College
Of Engineering and Research,
Avasari(kh), Tal. Ambegaon, Pune

Abstract: ***We are developing a cryptographic key management system for distributed networks. Key management is the management of cryptographic keys in a cryptosystem. Our system contains every aspect of key management, such as the key generation, key transfer, access control, and cryptographic algorithm agility. Key management is harder than encryption. All key management tasks are hidden by our system from the user. The repository does not need to perform any additional tasks beyond its normal course of operation: storing, servicing, and replicating data. Our system provides a high level of data security and relieves users from worrying about key management tasks. System will provide a simple data protection interface, will also perform advanced tasks for more sophisticated users.***

Keywords: Access control ; confidentiality ; cryptographic key; integrity; key management policies; source authentication; Encryption, Decryption, Recryption; DEK : Data Encryption Key, MEK: Master Encryption Key; KEK : Key Encryption Key.

I. INTRODUCTION

A. PURPOSE

Our goal is to fill in the gaps in the existing system with actionable solution (Cryptographic Key Management) in the area of encrypted data storage, and provide a secure platform for day-to-day credit card storage and management. The following objectives must be met to achieve security requirement for credit card storage:

- ✓ Protect the credit card number, expiration date, service code and card holders name from logical or physical access.
- ✓ Use access controls to provide separation of duties between administrators and users who access credit card numbers.
- ✓ Securely generate, store, distribution and transmit encryption keys, protecting them from exposure, unwanted replacement or misuse, and establish procedures to provide dual control over key management.
- ✓ Log access and administration of key management and PAN data storage systems. Document your process and protection measures.

B. PROPOSED SYSTEM

We are going to implement the system for:

- ✓ To develop a system which provides high security for the data that must be kept secured such as credit card number.
- ✓ To develop the system that manages Key Life-Cycle, which includes Encryption, Decryption and Recryption. It includes the following points:
  - Encryption Methodologies
  - Key Generation
  - Key Storage
  - Key Distribution
  - Key lifetime (crypto period)- Key Expiry, Key Replacement
  - Access of keys for encryption/decryption of data
  - Execution of the key lifecycle
  - Auditing of key lifecycle
  - Managing a compromise of a key or set of keys.

In our system we are using three keys DEK, MEK, KEK.

When any sensitive data like credit card number or password will get entered it will be encrypted using the DEK, afterwards DEK will get encrypted using MEK. When the data

is compromised then recryption takes place. Recryption is nothing but the act of decrypting data with one key and encrypting it with a different key.

For key generation we are going to use three different algorithms. For generating DEK we will use PRNG and Fortuna algorithm. MEK will be generated by using UUID and Fortuna. And KEK will be generated by using PRNG and Fortuna algorithms.

C. EXISTING SYSTEM

The existing system such as online purchase system uses DES or Triple DES algorithm for encryption or decryption of confidential data. The existing system:

- ✓ Provides less confidentiality of secret keys.
- ✓ Provides less security to the credit card based transactions.
- ✓ Unauthorized use of public or secret keys.
- ✓ Does not provide fraud detection.

D. SYSTEM ARCHITECTURE

We will be turning attention to the states which the keys can reside in when in this lifecycle. The keys can traverse any of the states during the lifecycle.

a. PRE-ACTIVATION STATE

In the pre-activation state the key was already generated, but not yet in use. While the key is in this state it can be submitted to a CA (Certification Authority) for certification and registration. Or the key can be used to perform key conformation between different parties.

b. ACTIVE STATE

This is the state where the key is active and can be used to encrypt or decrypt data. Encrypt or decrypt data can also be called, protect information for encryption and process previously protected information for decryption.

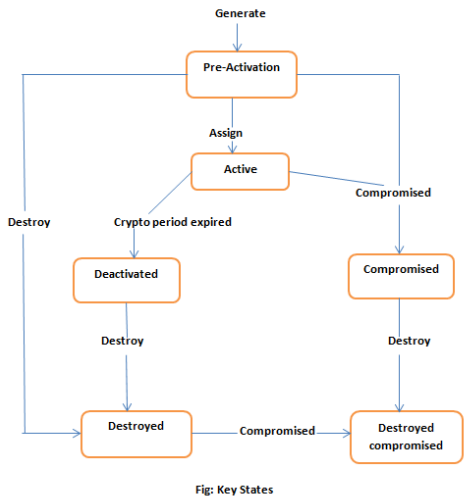

Figure 1: Architecture of Key Management

c. DEACTIVATED STATE

This state is the state that a key resides in when the crypto period has passed. In this state the key can still be used to process cryptographically protected information. The key stays in this state until it is not required anymore to process information, after this the key is destroyed.

d. DESTROYED COMPROMISED STATE

The key is destroyed in this state and the key attributes might be retained. The only difference with the state above is that in this state the key is known or suspected to have been compromised.

e. ALGORITHMS USED IN PROJECT

- ✓ UUID (Universal Unique Identifier)
- ✓ Fortuna algorithm
- ✓ Pseudo Random Number Generator (PRNG)
- ✓ AES algorithm (Advanced Encryption Standard)

II. CONCLUSIONS

Text encryption for sensitive data is very important for distributed the online transactions. By using the encryption and decryption algorithms for encrypting and decrypting the keys used for the online transactions we are going to develop a system for providing the security for the online transactions. we will implement the system which will provide the security for confidential data such as credit card based transactions using scala programming language and play framework. Data confidentiality and encryption, decryption are crucial factors to key management applications success and these have to be taken into account.

III. ACKNOWLEDGMENT

We wish to express our deep sense of gratitude and indebtedness to Prof. D. J. Pereira, HOD of Computer Engineering Department, GCOEAR Avasari (kh), Pune for introducing the present topic and for his inspiring guidance constructive criticism and valuable suggestion throughout this project work. We also extend our sincere thanks to all our friends who have patiently helped us in accomplishing this undertaking.

REFERENCES

[1] NIST Key Management Project
[2] http://csrc.nist.gov/groups/ST/toolkit/keymanagement.html
[3] NIST Key Management Workshop
[4] http://csrc.nist.gov/groups/ST/keymgmt/
[5] RFC4107
[6] http://www.rfc-editor.org/rfc/rfc4107.txt
[7] IEEE Key Management Summit

ISSN: 2394-4404

[8] http://www.keymanagementsummit.com
[9] Strong Key Open Source Project
[10] http://www.strongkey.org
[11] Oasis Enterprise Key Management Infrastructure
[12] http://www.oasis-open.org/committees/tchome.phpwgabbrev=ekmi