International Journal of Engineering and Advanced Technology (IJEAT)
ISSN: 2249 – 8958 (Online), Volume-9 Issue-5, June 2020

A Policy Based Data Security and Key Management System

Saket Ringsia, Shobha G

Abstract*: Cyber Security, over the last few years, has been a topic of great research given the number of cyber crimes have been increasing. To provide cyber security , constant efforts are being made to secure the communications and to protect user data using various cryptography techniques. With the increasing number of cryptography, the number of keys used to secure communications also sees a high. It is always difficult to manage these keys and their identity in a multi process communication environment. This paper brings into light an approach for the enhancement of data security and cryptographic key management using a policy based key management system. An on the device approach is proposed which uses the file system to create a secure storage with enhanced security for the storage of the data . The access to this storage is governed by policies to allow an application based access to the storage. This model will provide a highly scalable secure storage and management of keys.

Keywords: Secure storage, Policy based access control, key management, secrets, keys.

I. INTRODUCTION

All the devices use many secrets and protected information, as well use cryptographic keys for secure communication. These information are all stored by different subsystems in different ways that results in an inconsistent security posture for the overall system. The secrets need to be stored in a consistent way across the system; and the storage mechanism must also ensure that secrets on one device are locked for that particular device and are not transferrable unless authorized This is key for virtual platforms that run as virtual machines that can be cloned.

Hashicorp's Vault is an open source secure storage software that provides pluggable modules for backend storage, type of secrets, and audit devices, and provides a full fledged policy system that determines who has access to what secrets. In this paper, the Hashicorp's Vault is assessed in conjunction with a virtual TPM (vTPM) backend to provide a strong secure storage solution for secrets. The vault is used to govern the policies for all the applications running on the device and the virtual TPM is the storage which stores the data by maintaining high security by using the security features of the Trusted Platform Module which is present on all devices.

Revised Manuscript Received on May 25, 2020.
* Correspondence Author
Saket Ringsia*, Dept. of Computer Science & Engineering, RV College of Engineering, Bangalore, I-ndia. E-mail: ringsiasaket@gmail.com
Dr. Shobha G, Dept. of Computer Science & Engineering, RV College of Engineering, Bangalore, India. E-mail: shobhag@rvce.edu.in

II. LITERATURE SURVEY

This section reviews the literature of key management techniques , policy based access control and various ways of ensuring storage security.

In [1] Gebremichael et al. presents lightweight group key management protocols for an IoT network with low computational power and a less resource nodes.Today, there are multiple systems for managing keys. J. Gustafsson et al. in [2] underlines the importance of securely managing secrets aims to recommend a best practice to securely handle secrets in a way, feasible for small businesses with more limited resources, e.g., budget, knowledge, people, and hardware. It focuses on comparing software-based key management systems that facilitate the hassle of maintaining secrets by providing a secure way of storage and an automated way of management. In [3] , Focardi et al. describes various keystores to provide integrity to shared keys in a cryptographic mechanism.

Sven Plaga et al. in [4] introduces Hardware Security Modules (HSMs) which provide secure storage as well as efficient usage of cryptographic keys.Instead of using hardware related modules some work introduces using multiple cores and the use of distributed computing for proper key management and to ensure security of keys. In [5] John Patrick McGregor et al. propose better output for processors by using virtual secure coprocessing. By using this secret keys can be accessed securely without use of any hardware. Also , M.V. Srinath et al. in [6] presents a comprehensive summary of how in a Secure Muticast Environment different key management techniques can be used.

The techniques of using distributed computing made the cost higher and also pressure on hardware was also increased. So , a large amount is done related to cloud based storage of security keys. In [7] Amar Buchade et al. analyse symmetric key cryptography algorithms and their management of keys in a cloud environment. In [9] Dharam Raj Kumar et al. did a experimental review of key generation process in cloud data storage over the internet and brought into light various problems that arises due to different cryptographic algorithms which are used for data storage and retrieval from the cloud. S. Rajeswari et al. in [10] gives us an idea about how storage security is maintained in the cloud based services. It tells that we need to expose our data to an external service in the cloud which is not desirable for some lawful information.

For multiple applications to run securely by using our storage space , we need efficient policy based access scheme for selective access to particular keys and data.

Attribute-based encryption is used for secure and very fine sharing of data as well as access control which is decentralized. Nesrine Kaaniche et al. in [11] propose a multi-level access control mechanism which uses attribute based encryption schemes.

The model helps to gain good access control , it supports multiple levels of security and defines access right to new files.

Although all of the work reviewed does bring new insights into their application domains, none of the existing studies considers the problem of ensuring both storage security as well as policy and role based access mechanism to provide a comprehensive solution for the cryptographic key management. This demonstrated a research gap that this project hopes to bridge, providing solutions for both problems on the device itself.

III. PROPOSED MODEL

In this model, two open source tools integrates Vault and a virtualization of a secure memory-chip , also known as a cryptoprocessor, the Trusted Platform Module is done to provide a full fledged policy based key management which we call a secure storage. This storage uses the TPM 2.0 chip and the file system memory to store the data. This data, mostly secrets and keys, is encrypted with cryptographic algorithms and stored in the memory. Different applications can access the storage based on their roles and policies are defined to make sure privacy of keys is maintained.

The entire process can be seen as two modules : the policy defining vault and the vTPM storage. Vault allows an interface to build plugins called as secrets plugin and authentication plugin. The secrets plugin will help to push data to the secure storage and the authentication plugin will be used to provide credentials for an app . Before a plugin can be used it is needed to register it with a catalog. This entire process of registration is depicted in the following diagram:

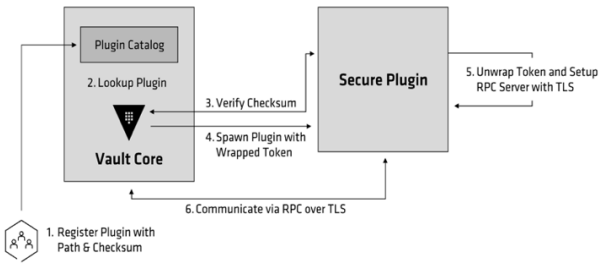

Fig. 1. Registration of Vault Custom Plugin

Once the plugins are registered we need to register the applications to the vault using the authentication plugin in the following way. Firstly, policies are defined for all the applications using the vault ACL based policies. Then, the application must ask the authentication plugin to give credentials in the form of a TOKEN ID and SECRET ID. This token will be used to authenticate an application before it can access the secrets plugin. After this is done, the applications can authenticate themselves to access the secrets plugin.

In the secrets plugin , we write the data or key from the application that has to be written to the storage . This plugin is also used to read the data from the storage when needed. It provide other operations like delete and update as well. So, this plugin acts as a bridge between the application and the secure storage and can only be accessed by a trusted application which is registered with the vault.

The operations that can be carried out by a secrets plugin by a particular application depends on the policies that have been defined for that particular application. If an application has only a Read access it cannot run the Write command on the secrets plugin. It can perform only the operations which the policies allow. Thus this is how policy based access of applications are maintained on the Vault side.

After this, the data written to secrets plugin along with the policies are given to the secure storage of vTPM. The transfer of this data is secured using a TLS based encrypted tunnel. In this storage, the data is encrypted automatically using TPM based encryption and is stored in the form of cells known as nv-index. Also, based on the policies that are given from the Vault, TPM policies are also created which is applied to the data of that particular application. So, this makes sure that an application is not able to access secrets or data that belong to some other application. When data is needed by the application , the secrets plugin accesses the particular nv-index , which is decrypted and then passed on to the application. This way we are able to complete workflow of the application accessing the secure storage for various operations.

IV. RESULTS

The model provides a not so complex and highly effective solution for the data security and key management problem. This model can be used as a subsystem in any infrastructure of devices to store data and keys which are securely maintained. To evaluate the performance of the system , it is seen how the security of the secrets of an application is protected in a scenario where two applications, APP1 and APP2, are running and using the system. Suppose the policies are defined such that for an application APPX , the application is allowed to access the path /secrets/APPX/* .

So the performance of the system is seen by analyzing the following cases:

1. APP1 makes a request to write secrets to /secrets/APP1/test : In this case , the vault client receives the request and after analyzing the policy of APP1 it sees that APP1 is trying to access the correct path. So , it allows the secrets write and necessary write operations are performed and secrets is stored into the storage.

2. APP1 makes a request to read secrets stored in the storage for the path /secrets/APP1/test. The vault client checks the policy for APP1 and sees that it has access to the given path . Hence, it performs the required operations to read the data at the given path from the storage and sends the secrets to the application.

3. APP2 makes a request to read secrets stored in the storage for the path /secrets/APP1/test. The vault client checks the policy for APP2 and sees that it does not have access to the given path . Hence, it sends a response to the application stating that the permission is denied.

The system had a response time of approximately 6.5ms for each operation involving the storage.

Thus , by analysing the above cases it is seen that the most important aspect of shielding one apps secrets from other and maintaining the security of secrets is successfully met.

The model can be used for highly scalable environments where the number of applications is high. This is depicted by the result where it is seen that the system to able to correctly provide all the functionalities in an environment with 100 applications running together in a high traffic environment. The model showed a very good response time, approximately 8 ms per request, as well without effecting the normal functioning and resources of the operating system.

V. CONCLUSION

Strong Encryption need Strong Encryption Key management. To be able to secure the keys used for encryption as well the data is the need of the industry given the increasing number of security issues. An effective Key management solution makes sure that these data and secrets are only within the hands of security teams. Also, an end to end solution for key management between applications and secure storage is important and useful so that the applications should not worry about attacks and of there data being stolen. This model is implemented fully on the device and secrets need not be given to a third party cloud service for security. One can be fully ensured about the management of their keys and safety of data. A limitation of this model is that it can hard to scale or might have slow output in highly memory constrained devices due to requirement of vault.

VI. FUTURE WORK

A possible work to enhance this model is to explore options of using role based policies instead of application based policies so that each application can use one or more role based policy to access the storage and eventually the number of policies will be reduced. Also, a dual factor authentication can be used by making a mapping of vault policies to vTPM policies and use policies defined by vTPM for storage.

REFERENCES

1. Gebremichael Teklay, "Lightweight Cryptographic Group Key Management Protocols for the Internet of Things," presented at IEEE International Workshop of Security and Trust, Luxembourg, (2019).
2. Jacob Gustafsson and Adam T √ ∂rnkvist, "Secure handling of encryption keys for small businesses : A comparative study of key management systems," presented at International Cybersecurity Congress , Moscow, (2019).
3. Riccardo Focardi, Francesco Palmarini, Marco Squarcina, Graham Steel and Mauro Tempesta, "Mind Your Keys? A Security Evaluation of Java Key-stores," presented at Network and Distributed System Security Symposium , California, (2018).
4. Sven Plaga, Norbert Wiedermann, Gerhard Hansch, and Thomas Newe, "Secure your SSH Keys! Motivation and Practical Implementation of a HSM-based Approach Securing Private SSH-Keys," presented at 17th European Conference on Cyber Warfare and Security ECCWS, Norway, (2018).
5. John McGregor and Ruby Lee, "Protecting cryptographic keys and computations via virtual secure coprocessing," in SIGARCH Computer Architecture News, (2005).
6. B. T. Geetha and M. V. Srinath, "A Study on Various Cryptographic Key Management and Distribution System in Secure Multicast Communications," in International Conference on Advances in Mobile Network Communication and Its Applications, (2012), pp. 64-69.
7. Amar Buchade and Rajesh Ingle,"Key Management for Cloud DatA Storage: Methods and Comparisons," in International Conference on Advanced Computing and Communication Technologies, (2014), pp. 263-270.
8. F. Mohamed, B. AlBelooshi, K. Salah, C. Y. Yeun and E. Damiani, "A Scattering Technique for Protecting Cryptographic Keys in the Cloud," presented at IEEE 2nd International Workshops on Foundations and Applications of Self Systems ,University of Arizona, (2017).
9. Dharam Kumar and Jitendra Sheetlani, "Review of Key Management and Distribution Technique for Data Dynamics for Storage Security in Cloud Computing , " IOSR Journal of Computer Engineering, (2017), pp. 38-49.
10. S. Rajeswari and R. Kalaiselvi , "Survey of data and storage security in cloud computing," in IEEE International Conference on Circuits and Systems (ICCS), (2017), pp. 76-81.
11. Zakaria Igarramen, Ahmed Bentajer, and Mustapha Hedabou ," TPM Based Schema for Reinforcing Security in IBE's Key Manager, "presented at International Conference on Data and Model Engineering, Toulouse, (2019).
12. Changji Wang and Jianfa Luo, "An Efficient Key-Policy Attribute-Based Encryption Scheme with Constant Ciphertext Length," presented at International Conference on Computational Intelligence and Security, Hong Kong, (2017).
13. Nesrine Kaaniche and Maryline Laurent,"Attribute based Encryption for Multi-level Access Control Policies," in International Conference on Security and Cryptography, (2017), pp. 67-78.
14. K Bhargavan, Richard Barnes, and Eric Rescorla,"TreeKEM: Asynchronous Decentralized Key Management for Large Dynamic Groups A protocol proposal for Messaging Layer Security (MLS), "presented at INRIA, Paris, (2019).
15. Sam Kim and David J. Wu, "Access Control Encryption for General Policies from Standard Assumptions," presented at International Conference on the Theory and Application of Cryptology , Brisbane, (2018).
16. Nico Ferrari, Teklay Gebremichael, Ulf Jennehag, and Mikael Gidlund ,"Group-Key Establishment Protocol for IoT Devices: Implementation and Performance Analyses," presented at Fifth Internationa, (2018).
17. Yotam Harchol , Ittai Abraham , and Benny Pinkas , " Distributed SSH Key Management with Proactive RSA Threshold Signatures," presented at International Conference on Applied Cryptography and Network Security, Belgium, (2018).
18. Neetesh Saxena and Santiago Grijalva, " Dynamic Secrets and Secret Keys Based Scheme for Securing Last Mile Smart Grid Wireless Communication," in IEEE Transactions on Industrial Informatics, (2017), pp. 74-83.
19. C. Li and C. Yang , " Cryptographic key management methods for mission-critical wireless networks, " presented at 7th IEEE International Conference on Electronics Information and Emergency Communication (ICEIEC) , China , (2017).
20. Seth Vargo, "Building a Vault Secure Plugin",https://www.hashicorp.com/blog/building-a-vault-secure-plugin/, October 30 2017.

AUTHORS PROFILE

Saket Ringsia is a Bachelor's student at the Department of Computer Science and Engineering, R. V. College of Engineering, Bangalore, Karnataka, India. He has a CGPA of 9.37. His interests lie in Computer Networking, Machine Learning and Cryptography and he wishes to pursue these in higher studies. He has represented his institute at various conferences and competitions. He is currently working under the mentorship of Dr. Shobha G.

Dr. Shobha G is a Professor in R.V College of Engineering, Bangalore. She has over 25 years of teaching experience and 14 years of research experience. Her primary interests lies in Data Mining, Image Processing and Networking. She has published over 123 international journal/conference papers in her area of research. She has also filed 4 patents and reviewed several books. Dr. Shobha is the former Head of Department of Computer Science and Engineering Department of R.V College of Engineering.