PrimeFactors™

APPLIED DATA PROTECTION

WHITE PAPER

Five Key Management Fundamentals

For Unlocking Encryption Success

February 2020

Introduction

As any cryptography expert will tell you, encrypting data is easy. It's the process of securely decrypting it that's the problem. Managing and securing the keys that control the encryption process is a high-stakes proposition. Lose the keys and your encrypted data is lost forever. Fail to institute controls over the keys and the confidentiality of the encrypted data is lost forever.

"Bad guys don't attack encryption; it just doesn't happen. They go for the keys. They go through the side-door. It is really easy to implement encryption, but it's only as good as how well you hide the keys," says Walt Conway, manager at 403 Labs, a PCI Qualified Security Assessor (QSA) firm. "You can have three locks on your door, and if you have the key under the mat, it's not going to help you. That's unfortunately the equivalent of what many people do. Encryption doesn't fail; key management fails."

There are several common threats to key management and secure data access that are identified over and over again by top data protection experts. These threats help to identify the critical features needed in a key management solution, and they also inform a handful of considerations organizations should address when evaluating key management platforms.

Separation of Duties

In many cases, companies fail audits because software developers leave encryption keys exposed on the file system. This issue often arises when organizations give the data owner custody of the encryption keys. In a payments-related business, this scenario could give a developer the power to expose large quantities of payment card numbers. This was a particular concern to the CIO of a payment card manufacturer. "The people who have access to the data should never, ever have access to the keys," said the CIO. "How do I prevent the guy who's writing the code and building the system from accessing the encryption keys that encrypt the data? Because he's writing the code that needs access to the keys to encrypt and decrypt the data. So how do I prevent him from having access to the physical keys needed in order to do his job? Here's how you do it. You isolate the keys from the developers. You link the keys to the physical machines that are running the process, but not to the code and not to the developers," the CIO said.

Most encryption or key management experts agree that separation of duties is perhaps the most important foundational principle to keeping encryption keys safe. Users of a system should be segregated into groups related to their specific roles, such as administrator, user and auditor, and security should leverage role-based access controls that grant each user group only the level of access they need to perform their respective duties – nothing more. Security policies should carefully define the privileges and limitations of a user group within a system, and any user dealing directly with data should be restricted from creating and managing cryptographic keys. Administrators should have the sole capability to establish user groups and user privileges, and each user should have access to only the functions necessary to perform their defined tasks, such as creating keys and certificates or importing and exporting files. Certain sensitive tasks should require dual user logon to execute.

System auditors should have the sole responsibility for establishing and revising the audit log and for systematically viewing the log. They should also be periodically reporting on the overall system operational security – one of the five fundamentals of key management covered in this paper.

Split Knowledge

Regardless of the specific segregations of duty employed, someone in every organization will have access to cryptographic keys. To maximize the security and operational integrity of the keys, organizations should split up who can access the keys and limit when, where and how a type of key can be accessed, often based upon the type of data it is protecting and what the enterprise is doing with the key.

> *"It's like Red October, where it takes two people to launch the missile. You should have split knowledge of the keys."*
>
> — Walt Conway, QSA, 403 Labs, Inc.

Split knowledge requires two or more administrators to complete critical key management processes. For critical, sensitive operations, different administrators must come together to facilitate a dual log-in or even three-part log-in (or more), in order to ensure that everything is above board when accessing or changing keys. Multi-Admin logon might be required for certain operations, particularly where master keys or keys that control more than one function are required, such as initial loading, changing, or rotating master keys.

According to Conway, the idea is similar to how you'd keep an atom bomb from being deployed. "It's like Red October, where it takes two people to launch the missile. You should have split knowledge of the keys," he says. "The keys are generally broken into different parts, generally held by different people, and each one has a backup in case someone gets hit by a beer truck. So you've got a backup person and maybe a backup on a flash drive or CD in a physical vault somewhere." Visibility and audit trail functionality should be built into the key management infrastructure to track who has signed on and had access to keys at any point in time.

Safe Storage

> *"That's a no-no! You don't want the keys to be stored or archived on the same box where the data is being encrypted."*
>
> — VP of encryption at a large financial institution

Many organizations make the critical mistake of storing encryption keys on the same subdirectory of the same machine where the encrypted data is stored. Unfortunately, this could be like giving thieves a gilded invitation to break into the data. In a sound key management environment, public keys that encrypt data and the private keys used to decrypt it should never be stored side-by-side in the clear together.

Encryption keys should be separated and securely stored based upon their unique attributes, uses or requirements, such as public keys, private keys or key designed for other functionality. Often securely storing keys requires encrypting the keys themselves with another key encrypting key (KEK). Storing an encryption key on a local client can help to speed up encryption services by providing quick access to the key locally, however the keys themselves must be encrypted.

Auditing Access

For data and key security and general compliance, all transactions related to key access and changes should be traceable and ideally reviewed periodically by an auditor who is not also in charge of managing the data or the keys. An audit log should track system login and identify the details related to specific functions performed in a key management system, including User IDs, time stamps, and information either inserted into or removed from a key vault. A robust key management system should also include alerting capabilities that can be configured to notify auditors when specific actions or events have taken place within a key management system.

Policy Documentation

To ensure that the previous four fundamentals – which are vital to ensuring keys are accessed securely and never compromised – are effective, the procedures to carry them out must be well documented.

> *"Education is critical. You're going to have a lot of people encrypting your data. If they don't thoroughly understand that yes, it's good that they've encrypted the data, but they need to know where they put the key, then they may well lose control of the data."*
>
> — VP of encryption at a large financial institution

"We can talk about who can access a key and who can't, what encrypted format we are using, where the keys are securely stored," Conway says. "But if it isn't written down, it doesn't exist." Once the documentation is developed, it is equally important to circulate it and educate the organization about its importance.

What to Look for in an Encryption Key Management System

- Easy to Integrate Across Any Common Operating System
- Integrated Encryption, Tokenization and Data-masking Capabilities
- Role-based Access Controls to Manage Data and Key Access
- Split Knowledge Support with Multi-Party Log-In Capabilities
- Integrated Key Management with Scheduling (Rotate, Expire, etc.)
- Integrated Audit Logs and Alerts for Traceability and Compliance
- Robust, Secure Cryptographic Key Vault (Universal Key Repository)
- Remotely Accessibility (from anywhere within a global company)

About Prime Factors

Prime Factors is a global leader in applied data protection software, helping to secure an open and collaborative digital world. Data has never been more plentiful or more valuable, and the protection of sensitive data has never been more complex. However, software solutions from Prime Factors help to simplify the complexities associated with protecting sensitive information. Since 1981, Prime Factors has served more than 1,000 global customers, including 80% of the top financial institutions in North America, with cryptographic software solutions for payments, EDI, and general data protection across a variety of industries. www.primefactors.com Follow Prime Factors on LinkedIn or Twitter.