ULTRA | Intelligence & Communications

Cryptographic Key Management
A Beginner's Guide

Overview

Cryptography is almost as old as human civilization, dating back to the ancient Egyptians. It is used to pass coded messages back and forth containing vital information that must be communicated between a commander and their forces. Julius Caesar developed a famous cryptographic system to communicate with his senior officers. In this system, Caesar shifted each letter of his message three letters to the right to produce what is called the ciphertext.

Breaking a Caesar cipher can be achieved via two different means, either using frequency analysis or brute force. Frequency analysis uses the fact that certain letters are used more frequently than others to help guess the plaintext. Alternatively brute force only requires 26 permutations to be generated (one for each letter of the English alphabet), which should quickly lead to the plaintext.

```
k = CRYPTO CRYPTO CRYPT        + mod 26
m = HAVEANICEDAYTODAY
-----------------------
c = KSUUUCLUDTENWGCQS
```

Vigenere's cipher

But it was the Italian Vigenere in the 16th century who introduced the idea of cryptographic keys. With the development of keys, you no longer had to worry about the secrecy of the encryption methodology or algorithm, but rather the security of the encryption key. Users could generate a ciphertext using a look-up-table of rotated alphabets and reference using the key and the plaintext. However, one of the weaknesses of this scheme is that the key is repeated until it matches the length of the plaintext, which makes it susceptible to several code-breaking methods.

It wasn't until the 20th century that cryptography came into widespread use. Notable encryption developments of the 20th century were the Enigma machine, the US Data Encryption Standard, the invention of asymmetric encryption and the Advanced Encryption Standard (AES) contest. The 21st century promises even greater evolution with the advent and growth of quantum computing.

Everyone has heard about famous cybercrimes like Stuxnet, the Estonian cyber war, the Mt. Gox hack, and the PlayStation Network hack, to name but a few. The internet is home to a multitude of ransomware, malware and other viruses that attack businesses, governments and families. But hackers are only part of the equation when it comes to cybersecurity. There is an unseen level of validation that underpins all cybersecurity – cryptographic keys and encryption.

Cryptographic Key Management
A Beginner's Guide

The Keys That Make the Digital World Go Round

Encryption and the keys involved in the encryption process are, for most of us, an invisible part of our everyday modern lives. However, how these keys are managed and controlled is vitally important, with extremely serious repercussions should keys fall into the wrong hands. That's precisely why keys must be protected from generation to retirement, both by using industry best practices and by employing Hardware Security Modules (HSMs).

Encryption and authentication are everywhere in modern life. Bank cards and payment systems, the internet, mobile phones, and even cars. Virtually anything you do has some form of encryption or authentication behind it. And what does encryption need?

It needs data to encrypt, an algorithm to perform the encryption, and perhaps most importantly, the key or key material. It doesn't matter how strong your encryption algorithm is, if a third-party gains access to that key, they can rapidly decrypt your data. That key could be giving access to your bank account, it could be securing all your business' trade secrets or it could be protecting international defense or financial systems.

Now let's look at two scenarios to put encryption and key management in context.

Losing Data Versus Losing an Encryption Key

As we've seen, encryption keys are like keys to door locks. So, can you pick the lock?

In Scenario One, you lose encrypted data. If the data has been encrypted with a well-implemented and strong encryption algorithm, such as AES-256, it should be a lot of work to decrypt that encrypted data. Brute forcing AES-256 using the fastest supercomputer in the world, Summit from IBM, at the Oak Ridge National Laboratory, which delivers a peak of 200 petaFLOPS, would take10 million trillion trillion trillion trillion years. That's a whole lot of computing power and a very long time. If you lose encrypted data, the chances are it will remain safe and secure.

In Scenario Two, you lose an encryption key. There are only really two possible outcomes:

1. The key is lost and never discovered by anyone, and the data remains encrypted and secure, whether for good or bad!
2. The key is lost and found by a third-party who exploit that key, and use it to decrypt all the encrypted data at a rate only constrained by the speed of the implementation of the decryption algorithm.

Rigorous key management is vital to ensuring scenario two remains a "what if" to your organization.

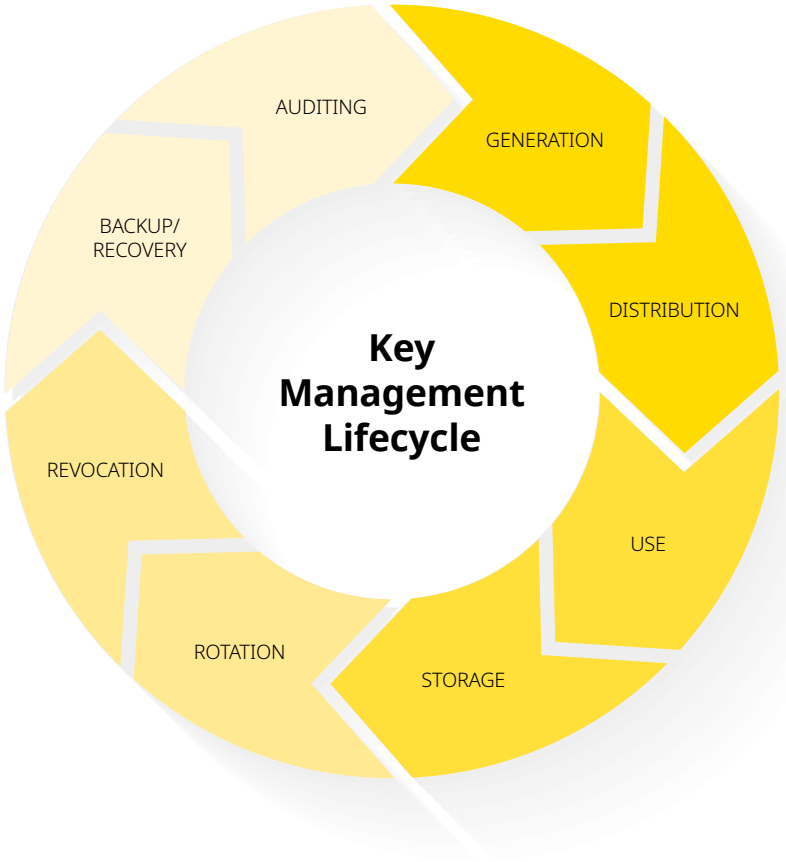
> *"Key management is the hardest part of cryptography and often the Achilles' heel of an otherwise secure system."*
> *— Bruce Schneier*

Cryptographic Key Management
A Beginner's Guide

A Primer on Key Management Operations

Key management is a set of operations which are needed to ensure a key is created, stored, used, rotated and revoked securely. There are eight operations:

1. Generation
2. Distribution
3. Use
4. Storage
5. Rotation
6. Revocation
7. Backup/Recovery
8. Auditing



Key ***generation*** is the first critical, step in the key management chain, and there are a myriad number of ways to perform this operation, but the same outcome of quickly generating a sequence of N bits, with maximal entropy is desired. In broad terms, there are two groups these methods can divided into, but we're only concerned with one – True Random Number Generators (TRNGs). TRNGs are usually based on a form of naturally occurring noise, such as the thermal noise in a resistor or diode.

Now that a key has been generated, it's imperative that it be stored and dispersed as its value dictates. There are various ways of keeping these keys secure, involving layers of physical security and/or encryption of the keys with appropriate procedures in place to ensure they remain secure and retain their integrity.

At the very simplest level, they may be protected by a user's password on a mobile phone. At the other extreme they may be stored in a highly specialised device known as a Hardware Security Module (HSM) that is air-gapped and can only be accessed when a certain number of users and their access control cards are present. Of course, there are many differing intermediate options, such as Trusted Execution Environments (TEEs), Trusted Platform Modules (TPMs) and Cloud HSMs. The strength of the protection required is dependent on the value of the data protected by the keys, and also in some industries, regulatory and compliance legislation.

The next step of the key lifecycle is ensuring the safe ***distribution*** of the keys. Some keys are distributed unencrypted under armed guard, but the vast majority of keys are distributed in an encrypted form, and then decrypted at point of use. Asymmetric key or Public-Key Infrastructure (PKI) distribution methods using X.509 certificates dominate and form the basis of most everyday communications such as the internet.

After distribution and receipt of the key, it can now be used in a secure manner for cryptographic operations by an authorised user, with safeguards in place to ensure the key is not misused, copied and so on.

However, the actual ***use*** of the key in a secure manner has its own challenges. The encryption algorithm must have been implemented correctly, and in such a way so as not to leak unencrypted data or key material into the encrypted data path. Its implementation must also take into consideration side-channel attacks, such as cache attacks, timing attacks, power monitoring attacks or electromagnetic attacks. It's a modern-day arms race between adversaries devising new and novel attack vectors and researchers who are continually working to mitigate these threats.

A Primer on Key Management Operations *continued*

There are ***storage*** standards in the industry, such as FIPS 140-2 and 140-3, that exist to set an industry-wide range of protection measures for cryptographic modules. These standards define 4 levels of protection:

- Level 1 the lowest, requires approved algorithms, but they can be executed on a general-purpose computer running an unevaluated operating system.
- Level 2 builds on Level 1 but adds requirements for physical tamper-evidence and role-based authentication and places some requirements on the operating system.
- Level 3 adds requirements for physical tamper-resistance and identity-based authentication, and for a physical or logical separation between "critical security parameters" interfaces.
- Level 4 is the highest level and requires an enhanced physical security to prevent and detect tampering, and to provide protection against environmental attacks.

Once a key's cryptoperiod, or time period the key is allowed to be used for, passes, the key must be ***rotated***, i.e., the current key is retired and replaced with a new key. For a data-at-rest system, this requires the decryption using the existing key and subsequent re-encryption of the data with the new key. The main benefits of key rotation are:

Key rotation limits the amount of information, protected by a specific key, available for cryptanalysis, while also limiting exposure if a specific key is compromised in some way. Key rotation also avoids certain known, and protects against any future unknown, algorithmic weaknesses that may either be key dependent or ultimately reduce key lifespan.

While retiring and refreshing keys seems like an obvious measure to guard against data loss, key rotation introduces a few challenges:

- In a data-at-rest situation, decrypting and re-encrypting any significant volume of data will take a considerable time, during which that data would be unavailable.
- All instances of the current key have to be updated securely with the new key.
- The key rotation for all systems and users must be synchronized simultaneously.

Key rotation is not the only way of dealing with a compromised key; ***revocation*** or destruction are the other options. Revoking a key means the key can no longer be used to encrypt or decrypt data, even if its cryptoperiod is still valid. Destroying a key, whether that is due to compromise or due to it no longer being used, deletes the key permanently from any systems. This makes it impossible to recreate the key, unless a backup image is used.

We're now going to talk about key ***backup and recovery***. Like any important data, it is prudent to ensure that a backup is maintained, in the event of any data loss or corruption, and the same is true for keys which are often needed for continued operations and cannot be easily replaced. The backup of key material on an independent, secure storage media provides for the option of key recovery if required; however, backing up key material involves the creation of a duplicate, under strictly controlled policies and conditions, that must then be protected by the same, or greater, mechanisms, that the original key material is safeguarded by.

It may also not just be the key itself that is required to be backed up – there may be associated metadata with it, that must also be safely duplicated.

In the event a key must be recovered, the first thing you should determine is the why. If key recovery is required due to a corrupted key, then the source of the corruption ought to be determined to ensure it wasn't a side effect of the key being compromised. Similarly, if key recovery is required because a key has been lost, the whereabouts of this key, or whether it has been compromised, would need to be analyzed to ensure that key recovery does not lead to further data exposure and compromise.

The sequences of key management operations that are performed on key material using a key management system should be periodically audited to ensure that key management policies and guidelines have been adhered to. Each of the stages of the key lifecycle must be checked for compliance in terms of physical, logical and personnel or users. The key management system should provide the facility for a dedicated user type to exist, who can only perform these audits, and cannot actually perform any

A Primer on Key Management Operations *continued*

other key management tasks. The logs generated by the system, and are reviewed by this user type, must have built-in mechanisms to ensure that their integrity and authenticity is maintained and can be verified.

As part of the ***audit***, the key management system should be checked for any visible signs of tamper or intrusion, and the system's audit logs should be checked for any evidence of tamper events. The audit logs should also be reviewed to ensure that the expected operations were performed on the individual keys managed by the system, on a key-by-key basis, and also a verification of the user who performed those operations.

The audit and compliance regulations vary from industry-to-industry but two of the mostly commonly encountered compliance authorities are the (National Institute of Standards and Technology) NIST and The Payment Card Industry Security Standards Council (PCI SSC).

The Fundamental Best Practices of Key Management

Now remember, the aim of key management is to prevent data compromise and meet compliance regulations, so the first step is to development a key management plan, that will set out what regulatory compliance is required and how best practices are going to be brought to bear. Standards, created by the NIST, and regulations, like PCI DSS, FIPS, and HIPAA, expect users to follow certain best practices to maintain the security of cryptographic keys used to protect sensitive data.

We'll now look at some of these best practices:

Key generation: Key generation is extremely important, and closely linked to this is, never hard-code key values anywhere. Hard-coding a key into open-source code, or code of any kind, instantly compromises the key. Anyone with access to that code now has access to the key material. Recently, in the UK, at IT professional was threatened with legal action after flagging up an exposed GitHub repo containing credentials and insecure code.

Least privilege: The principle of least privilege is the idea that users should only have access to keys and the subset of key management operations that are absolutely necessary for their work. By strictly limiting who can access and do what, you reduce the risk of both intentional and unintentional data breaches and also makes it easier to identify the source of any breaches should they occur.

Hardware Security Modules: Hardware Security Modules (HSMs), are physical devices that stores cryptographic keys and performs key management and cryptographic operations, such as encryption, decryption, digital signing and authentication, in a trusted environment. They come in a variety of different levels of security, with the highest levels providing significant levels of physical protection, and respond to attempted intrusion by securely erasing any stored key material.

Create and Enforce Policies: Creating and enforcing security policies relating to encryption keys is another way to ensure the safety and compliance of their key management system. Security policies specify the processes that must be followed, which will inevitably have been devised to leave an audit trail of who has performed what operations on what keys.

Separation of Duties: Separation of duties within key management is another important practice that is very closely related to the principle of least privilege and policies. By creating distinct duties or user types, that can only perform certain operations, there is an additional layer of security applied, since multiple users would need to be in collusion for key material to be stolen or manipulated in an inappropriate manner, and the audit trail to be erased.

Key Splitting: One final practice to ensure the strength of any key management system is by splitting the keys or access keys into multiple portions. In this way, no one person is in possession of the entire key, or can access the key, and multiple people, but not necessarily all of them, must come together to reconstitute or use the key.

HSMs - Smart Key Management

Hardware Security Modules or HSMs, are dedicated physical devices that are designed to store cryptographic keys and perform key management and cryptographic operations, such as encryption, decryption, digital signing and authentication, in a trusted environment. They come

Cryptographic Key Management
– A Beginner's Guide

HSMs - Smart Key Management *continued*

in a variety of different levels of security, often validated by third parties such as FIPS, Common Criteria or PCI, with the highest levels, providing significant levels of physical protection, and respond to attempted intrusion by securing erasing any stored key material.

A hardware security module can be considered to be a trust anchor, and trust anchors are used to protect the services we use every day, such as the internet, SSL, DNS, banking, mobile devices, code signing, smart meters, IoT devices, bank and credit cards, mobile payments, document management systems etc.

Due to the value and sensitivity of the key material stored in an HSM, they generally have a range of security features to enable enforcement of the principles of least privilege, separation of duties and key splitting, with physical protection measures to provide either tamper evidence, tamper resistance or tamper response. They are also typically certified to internationally recognised standards such as Common Criteria or FIPS 140 to provide an independent assessment and assurance of the security measures in place.

The Ultra KeyperPLUS HSM has previously been evaluated to the FIPS 140-2 Level 4 standard and is currently going through re-evaluation as the result of an algorithm and hardware refresh.

As the only standalone Level 4 HSM on the market, KeyperPLUS was specifically designed to limit all potential points of access with a tamper-resistant design, ensuring only those with intended permission may access the sensitive data it protects. Through vigorous and careful management of any areas of physical or digital infiltration, KeyperPLUS delivers a robust solution that meets the most stringent of security standards.

Based on this core technology, Ultra has built a product range to cater to the PKI, VPN and Internet security markets. The KeyperPLUS HSM is ideally suited to businesses and organizations deploying a cryptographic system where the protection of cryptographic keys is a priority, for example, in organizations requiring certificate signing, code or document signing, bulk generation or ciphering of keys or data.

Protecting Keys in Transit

Even today, in many instances, humans have to deliver new key material, often to inhospitable or dangerous places. Ultra's solutions allow key material to be moved electronically reducing costs, the chance for costly errors, security breaches and more. They avoid duplication and ensure all key material movements are tracked and audited.

The Remote Cryptographic Management System (RCMS) provides monitoring control and key delivery to dispersed stations over TCP/IP networks to greatly enhance access to remote locations, lower travel requirements and improve visibility of system operations. Using a net-centric approach to provide secure, remote cryptographic system management, RCMS includes a software controller that provides key management capability and allows the operator to remotely load keys to the MIDS terminals over a secure Ethernet LAN connection.

Cryptographic keys underpin our digital reality, and as more and more of our lives move online, the imperative to protect these keys is paramount.

ULTRA | Intelligence & Communications

1039.1-I&C-en-REV0721

ultra.group | sales@ultra-us-gbs.com