White Paper

Key isolation for enterprises and managed service providers

ENTRUST

SECURING A WORLD IN MOTION

Contents

INTRODUCTION

Cryptographic keys within an enterprise are used to identify people and machines, secure internal and external communications, encrypt and tokenize data at rest, and sign messages and documents. It is therefore vital for any business with a reliance on cryptographic keys to have assurances and enforceable policies surrounding key usage. The nShield family of hardware security modules (HSMs) provides the ability to achieve that level of assurance.

By using the nShield Security World key management framework, supported by the Entrust nShield HSM family, an organization can create for itself a structured key infrastructure that meets the dynamic demands and requirements of today.

This paper demonstrates how it is possible to easily configure nShield Security World to define a framework which permits both partitioning and multi-tenancy cryptographic key isolation strategies.

nShield is a family of multi-purpose HSMs that provide a trusted environment for secure cryptographic processing, key protection and key management.

Business models

There are two different business models which have a requirement for some form of partitioning with regards to their cryptographic resources estate:

- Enterprise customers who have a requirement to share cryptographic infrastructure resources between applications or departments within that same enterprise. Keys should be separated to preserve the necessary isolation between applications.
- Managed service providers who wish to divide a given cryptographic resource between any number of distinct and independent clients such that keys for one client are not accessible by another client.

This paper addresses both models, and shows just how easy it is to design, implement, and enforce a policy that meets their requirements.

In addition to the requirements of the business model, there are a number of factors to consider when assessing techniques for partitioning:

- **Object types:** Usually users, applications, or keys.
- **Scale:** This can range from one or two enterprise users or applications to millions of keys or customers.
- **Security:** What determines the true level of security? What authentication policies are protecting application key material? How are physical security controls mapped to logical controls, and vice-versa.
- **Accessibility:** What access does the hosting organization have to customers' material? Within an enterprise a provider may want to offer a super-user or an administrative quorum with access to all the keys. However customers will have more trust in a public service if the provider can't access their keys.

With all these factors in mind, how can Entrust nShield HSMs assist in the development of isolated systems for the control of keys?

Key isolation is often a requirement in both enterprise and managed service or cloud environments

nShield Security World

To understand the ways in which nShield HSMs can be deployed to support flexible isolation environments, we first need to have a clearer understanding of nShield Security World architecture principles.

To alleviate the developer from the burden of creating a key infrastructure, nShield Security World provides a simple, yet flexible key architecture that can be used to contain application keys, protected in a variety of ways, while also offering easy to use load-balancing and disaster recovery functionality and industry standard APIs such as PKCS#11 and JCE. When using nShield HSMs, this standard key infrastructure is typically used whether integrating with existing standard interfaces or bespoke applications.

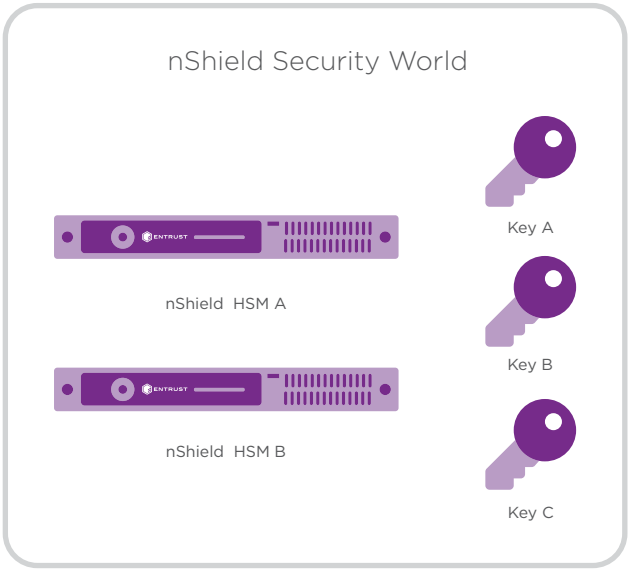

Figure 1. Security World as a single security domain for HSMs and application keys

Practically speaking an nShield Security World creates a single security domain for keys and objects to be securely managed that can encompass many nShield HSMs and clients. However, an nShield HSM can only ever be configured with a single Security World at any one time.

Application key tokens

When generating an application key within Security World on a nShield HSM, it is important that the raw key material be protected by the certified hardware module at all times. It is also important that the key can be loaded by authorized clients and backed up in accordance with industry best practice guidelines.

The nShield Security World provides mechanisms to cryptographically wrap the raw application key material, the fine grained meta-data detailing how the key can be used, and the Access Control List (ACL), with a specific Security World foundational key (KMSW).

The wrapped application key can then be stored on all authorized HSM clients for loading the key at a later date and included in standard back-up procedures. Security is maintained while stored on the client or on recovery media as decryption to obtain the key material is only possible on an nShield HSM in the relevant Security World containing the associated module key and where the HSM can be relied upon to comply with the ACL.

Since these wrapped application keys exist on the storage media on the authorized clients, the volume of keys that a Security World can protect is only limited by the size of the storage medium on the host, not some feature or limitation of the HSM. This neatly addresses the topic of "scale" in relation to isolation principles.

nShield Security World addresses the age old challenge of providing strong protection for keys while at the same time ensuring they are available for use by authorized applications that are deployed over high scale, redundant and distributed servers.

Protecting the use of keys

Where controls need to be implemented for safeguarding application keys, Operator Card Sets (OCS) or Softcards (passphrases) can be used to authorize the loading of those keys. Physical OCS and logical Softcards are collectively referred to as authentication tokens. An authentication token is associated with an application key when the application key is generated. The application key then requires the authentication token to be presented and validated before the key can be loaded onto an HSM.

Once an application key has been loaded into an HSM, it can be used (ACL permitting) as often as required for approved cryptographic operations before then being programmatically or automatically unloaded. A single authentication token can be used to protect multiple application keys.

Softcards

Application keys are, by default, protected by the KMSW wrapping process, however sometimes it will be necessary to implement additional security controls to ensure an application presents some form of authorization before the HSM loads the key for use. The single-factor authorization model adopted by nShield Security World is softcards.

Softcards are in essence a single passphrase which are useful where physical access to a smart card slot is impractical Softcards can be a practical solution to enforcing some control over when an application key is loaded.

OCS quorums

nShield HSMs use smart cards to provide two-factor authorization, however an OCS is not a single smart card (although it could be). An OCS is normally a set of smart cards which represent an authorized group. When created, the necessary quorum of these cards is also set. This is the number of cards from the total set that need to be presented in order to authorize the use of the keys protected by the cardset.

Since individual cards are normally allocated to authorized members of a group of users (each smart card with a unique passphrase), when a cardset is authorized within the HSM, this does not represent a single user's authorization, but rather the authorization of the group to perform the requested action.

The notation used to describe the quorum of the cardset is "K of N", where N is the total number of cards in the cardset and K (K>0) is the number of cards required to form a quorum. So in a "2 of 5" OCS, there are 5 cards in total but only 2 of this set need to be presented to permit the loading of a key.

A special property of a 1 of N OCS is that obviously only a single card need be presented (along with its passphrase). This means that no physical switching of cards in slots needs to take place, which can be a practical advantage in certain conditions where you not only want to protect where an application key is loaded. This enables you to retain the Softcard advantage of restricting when it is loaded, though it is perhaps an inferior configuration to a K of N OCS where K>1 in terms of security over availability.

ACLs

The Access Control List (ACL) forms a significant part of the meta-data associated with a key. It is securely wrapped along with the key when the key is generated, and is protected to the same high standards as the key itself. The ACL for a given key describes what authorizations are required for a specific operation to be performed, such as other keys or tokens that should be loaded, and what other limitations are applied to the key once it is loaded (such as time-outs and number of permitted operations).

An ACL can describe a very simple scenario whereby a key can be used to encrypt and decrypt data, or can describe very complex hierarchies of keys which must be loaded (using their respective authentication tokens) before selected operations can be carried out. These ACL policies are all managed, unwrapped and enforced by the HSM natively, and as such cannot be compromised by an attacker. The ACL for a key is set when the key is generated and is not normally modifiable after that.

Policy enforcement

The concepts of nShield Security World ACLs, OCS quorums and softcards are tightly bound together, and can be used in combination or isolation to meet even the most demanding security requirements or policies for a given application.

With this in mind, we can see that a key loading and usage policy is enforced by three factors:

- **Access to the application key token** – If you don't have the application key token on your application server you simply cannot load that key onto a target nShield HSM. This policy is enforced outside the HSM, by careful and deliberate synchronization of specific application key tokens across the application server estate.
- **Token authorization** – If a key is protected by an authorizing token, such as a Softcard or an OCS, then you must present that token before you are then permitted to load the key into the HSM. This policy is enforced inside the nShield HSM.
- **ACL** – Once the key is loaded, the key can only then be used for specific purposes and under specific conditions described in the ACL that is bundled in the application key token. Again, this policy is enforced inside the nShield HSM.

So how can these properties be used to construct an isolated security environment for our sample business models of enterprise and managed service provider who require isolation between applications or customers?

Security World avoids the need for expensive backup tokens and manual key cloning.

CONCLUSION

The nShield Security World key management architecture offers a high degree of flexibility which creates a fluid fit with enterprise or managed service requirements for multi-tenancy and partitioning.

The nShield HSM hardware provides a safe place where keys can be loaded and used. One of the core strengths of the nShield Security World architecture is that application key tokens are stored in an armoured format on application servers – free from the confines of any particular HSM. So deciding where the application key tokens are and are not available is really the initial factor one should define when designing a framework for partitioning.

It is therefore better to instead conceptualize the partitioning problem less as one about containerizing the HSM, since policy enforcement of key loading and use is already trusted, and more about how the ACLs and authorization tokens are configured with application keys. Most importantly, you should consider how those tokens are distributed and made available to individual application hosts.

> The ACL associated with an application key defines the key policy in a form an nShield HSM can strongly enforce.

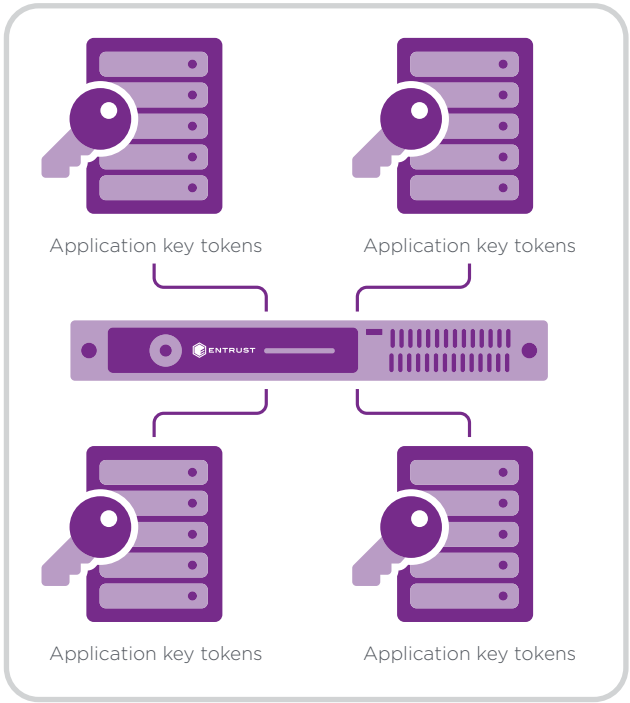

Figure 2. Application key tokens stored on hosts

The design and implementation of your HSM key management policies and architecture are based on your organizational needs and the balance of requirements between accessibility to keys for high volume or automated usage and the security controls defining key usage that might be required by certain high assurance situations. nShield Security World offers several flexible mechanisms to assist in the design and implementation of such an architecture.

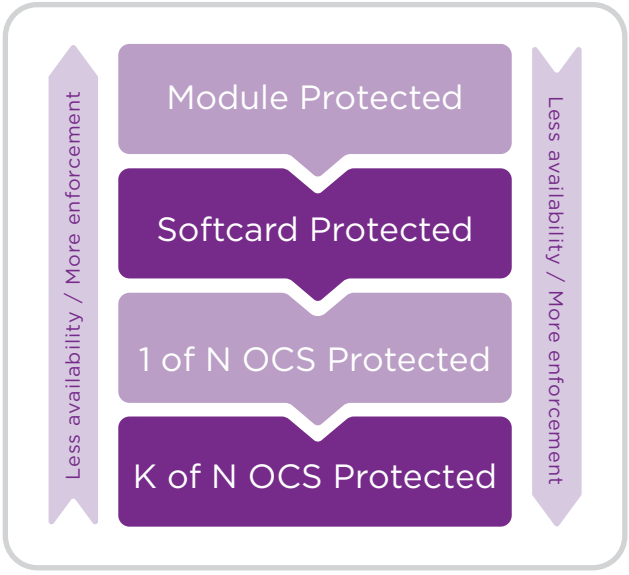

Figure 3. Degrees of availability and enforcement via authorizing tokens

Learn more

To find out more about Entrust nShield HSMs visit **entrust.com/HSM**. To learn more about Entrust's digital security solutions for identities, access, communications and data visit **entrust.com**

To find out more about Entrust nShield HSMs

HSMinfo@entrust.com
entrust.com/HSM

ABOUT ENTRUST CORPORATION

Entrust keeps the world moving safely by enabling trusted identities, payments and data protection. Today more than ever, people demand seamless, secure experiences, whether they're crossing borders, making a purchase, accessing e-government services or logging into corporate networks. Entrust offers an unmatched breadth of digital security and credential issuance solutions at the very heart of all these interactions. With more than 2,500 colleagues, a network of global partners, and customers in over 150 countries, it's no wonder the world's most entrusted organizations trust us.

Learn more at
entrust.com/HSM

ENTRUST

Contact us:
HSMinfo@entrust.com