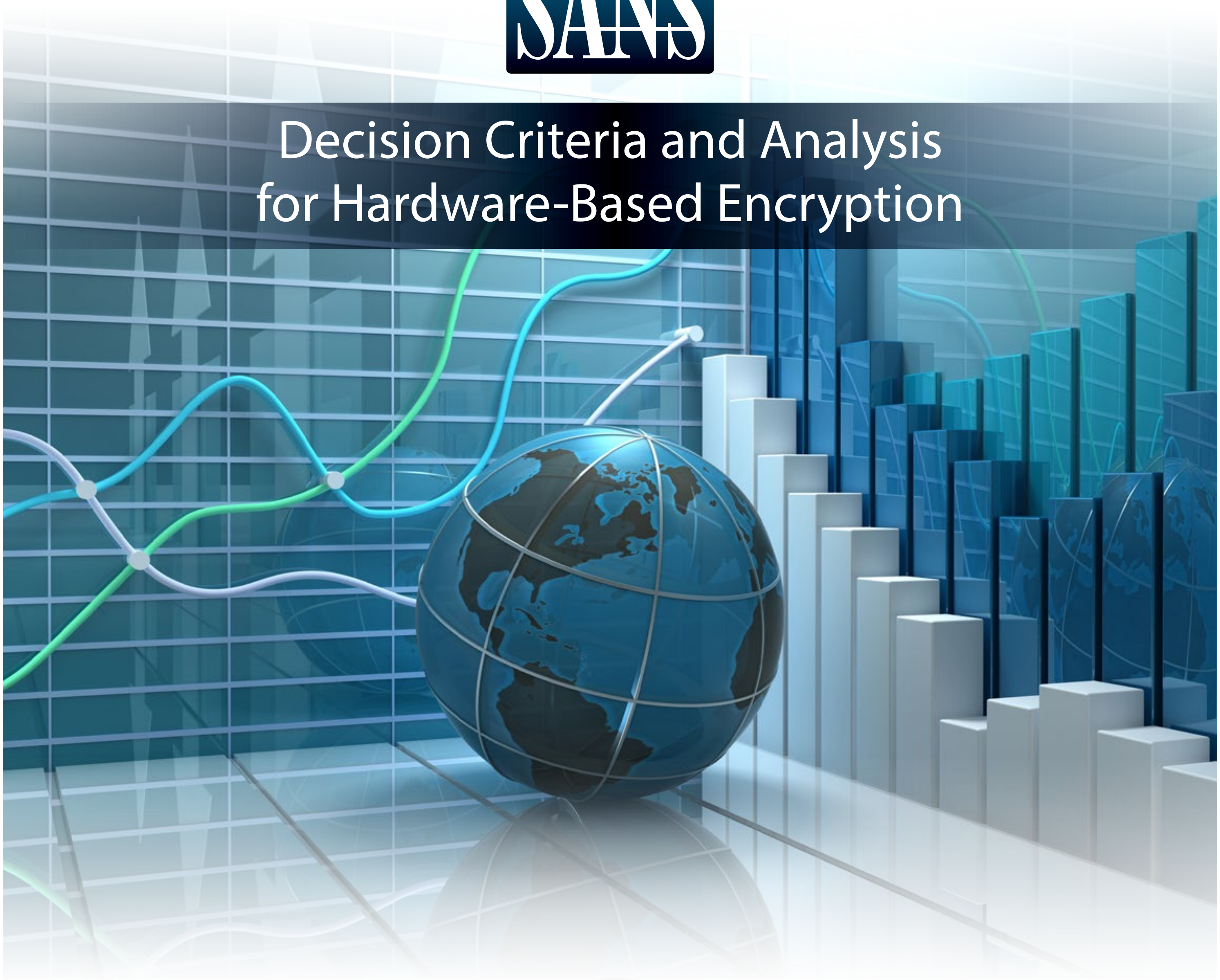
# Decision Criteria and Analysis for Hardware-Based Encryption

Organizations trying to balance the risk of data breaches against the inconvenience, latency and cost of encrypting every bit of valuable data often balk at the trade-off. But with the volume of digital data growing and computing environments becoming more complex and accessible, the ratio of cost to benefit has improved and encryption is now far more common in organizations that rely heavily on Internet- or cloud-connected applications for significant business functions

# Decision Criteria and Analysis for Hardware-Based Encryption



**A SANS Whitepaper**

*Written by Eric Cole, PhD*

July 2016

*Sponsored by*
*Thales e-Security*

# Introduction

Organizations trying to balance the risk of data breaches against the inconvenience, latency and cost of encrypting every bit of valuable data often balk at the trade-off. But with the volume of digital data growing and computing environments becoming more complex and accessible, the ratio of cost to benefit has improved and encryption is now far more common in organizations that rely heavily on Internet- or cloud-connected applications for significant business functions (see Figure 1).

**Does your company have an encryption strategy?**

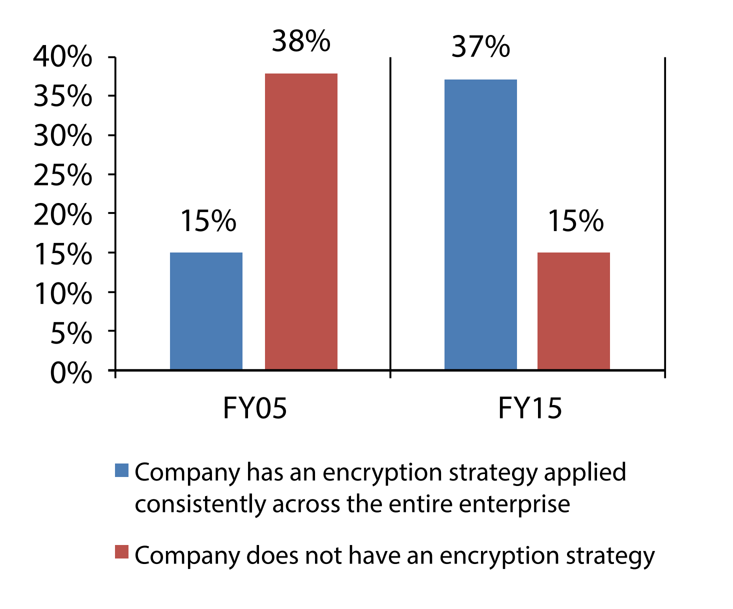


*Figure 1. Increased Use of Encryption to Protect Information*[1]

However, despite the growth of encryption and steady increases in the processing power of commodity-level servers, performance issues have held back the use of encryption for extremely performance-sensitive applications or very large data sets where encryption would be particularly resource-intensive. Hardware-based encryption uses hardware to perform the cryptographic functions, which offloads the processing to an independent system, increasing not only security but also performance. In addition to performance, risk and security of keys are also factors in using hardware security module (HSM) encryption. It is also important to mention that hardware-based encryption

[1] 2016 Global Encryption Trends Study," Ponemon Institute, February 2016, www.ponemon.org/library/2016-global-encryption-trends-study

could additionally refer to doing all of the encryption on dedicated hardware or using dedicated hardware to generate and manage encryption keys that are supplied to host/servers for use by applications that require encryption. For example, there are two main scenarios for how HSMs are used in symmetric cryptography: Either the full encryption process is performed on an HSM, or the HSM supplies only the key to a software-based encryption application. Both are valid deployment scenarios, with the decision based on risk, APIs/integration and performance considerations. A few examples of where HSMs are commonly used include database encryption, file encryption, SSL, storage and backups.

Offloading encryption processing to servers designed for and dedicated to it can boost performance while also allowing more complete, complex encryption methods via the use of HSMs. Adding the cost of hardware changes the cost-benefit ratio but to different degrees for each application. Organizations must therefore decide, on an application-by-application basis, which functions or data would benefit most from hardware-based encryption or key protection and what level of benefit would justify the additional cost.

This paper will define criteria for making those decisions repeatable, consistent and specific for any application, based on the organization's sensitivity to cost, risk tolerance and performance requirements. The mini case studies used for illustration involve situations on which I have worked with my own clients, whose names I have omitted to protect their privacy. They describe real situations, not hypotheticals.

# Challenges in Using Hardware-Based Encryption

While hardware-based encryption provides more security (via trusted algorithms and their tamper-proof nature) than software-based approaches, it does add to the expense and complexity of the solution. Because it is unlikely that every application can benefit enough to justify the additional cost and effort, organizations should deploy hardware-based solutions only for applications with sufficiently high requirements for security and performance.

*Because it is unlikely that every application can benefit enough to justify the additional cost and effort, organizations should deploy hardware-based solutions only for applications with sufficiently high requirements for security and performance.*

There are also logistical and operational complications to consider. For example, it is possible to push a software-encrypted RAM drive to customers via an SSL portal with minimal to no impact to the end user. However, if a hardware-based solution were required, a device would have to be provided to every customer, and plans would have to be made to address instances of hardware that malfunctions or is lost.

Hardware-based encryption solutions interface to software applications through standard crypto APIs. Compared with integrated/customized software-based solutions, adding, subtracting or reconfiguring applications (and making sure they and the HSMs use compatible APIs) may add complexity if the HSMs do not use the standard crypto APIs. Finally, if not properly tested, patches and updates to servers on which the application runs have the potential to break the interoperability between the two. Fixes may require restarts, reconfiguration of one or both systems, or expensive hardware upgrades following any system updates.

Finally, while hardware-based encryption can resolve many challenges in large-scale data processing in which security and performance are paramount, implementation requires careful analysis and configuration of all the key components of an application so that data storage and processing requirements can be understood and identified.

### Mini Case Study

The cost and complexity of hardware-based encryption can be misleading. Careful analysis of the data or processes to be encrypted can reduce a seemingly unmanageable implementation to one that is far more practical and affordable.

One particular example: A healthcare organization was integrating its electronic records systems from five hospitals to enable its doctors to seamlessly access all of the information about a given patient regardless of which location housed the data. The organization determined that hardware-based encryption was the best solution for both reducing the potential for data breaches and meeting performance requirements. The initial estimate forecast a project that would be over budget and behind schedule.

A detailed analysis of all the applications involved revealed that a recent upgrade of all the system's back-end databases to a consistent platform and data structure had virtually eliminated potential problems arising from inconsistencies in applications or versions. Despite some differences, the applications shared a consistent set of data calls and a processor standard to access patient records.

A new hardware-encryption implementation plan focused on encrypting the records access process alone, providing the security and functionality that was required without an unacceptable increase in cost or schedule.

Detailed analysis of the environment and the willingness to decompose the system into base components revealed a cost-effective approach that might not have been obvious otherwise. It is likely that many organizations that see hardware-based solutions as too complicated or expensive could find similarly workable solutions following proper analysis that includes prioritization of systems, understanding applications and tracking data flows within a business process.

# Design Criteria and Analysis: Finding Balance

Where and when is it appropriate to use hardware-based encryption? At one extreme, every organization that has sensitive data can gain from hardware-based encryption. On the other extreme, very few organizations, if any, can afford the cost or complexity of applying it to every system and application. The challenge is to identify the right problems on which to focus this solution.

The following are the design criteria and analysis that can reveal appropriate targets and implementation models:

- **Identify all critical data repositories.** An organization cannot protect assets it is not aware of. Therefore, the first step is to identify all critical data repositories and, if necessary, consolidate them to de-scope the size of the problem set. For each data set, prioritize and determine the criticality of the information and the impact it would have if it were compromised.

- **Map key business processes.** For each data set, determine the business processes and corresponding applications that access the information. Based on the data set and the type of access required, each business process should be further prioritized, including the business units and frequency of use.

- **Determine servers that support the processes.** Each business process and corresponding data set needs to be mapped to specific servers/hardware that provide the foundation for the processes. It is important to note that these servers could include virtualized environments and cloud-based systems, which could add an extra level of complexity when looking at hardware-based encryption. However, most cloud providers today allow support for hardware-based encryption. For example, one common use case is to encrypt data prior to sending it to the cloud; software- or hardware-based encryption can be used. A second use case is that organizations can generate their own encryption keys in their own dedicated hardware and provide it (securely) to the cloud provider to use on their behalf using HSMs in the cloud, such as what Microsoft Azure offers. A third use case is to rent an HSM in the cloud, as Amazon Web Services offers.

- **Create a threat map.** Start with information-mapping physical servers to specific processes and data sets; use the relative importance of processes and data sets to understand the criticality of each system. Use that information to begin a risk analysis that sets priorities on potential threats based on which of the threats have the highest likelihood of causing harm. The goal is to identify the vulnerabilities that would allow each threat to have the greatest negative impact.

*Every organization that has sensitive data can gain from hardware-based encryption, but very few can afford the cost or complexity of applying it to every system and application.*

- **Perform a risk assessment.** Threats are listed in a risk estimate and then multiplied by vulnerabilities and further rated by factoring in the likelihood and potential impact. A risk estimate should identify the risks that will cause the greatest harm to the organization, its mission and ultimately its customers. Risks will be different for each organization, so it is important to customize your list based on the respective emphasis you would put on confidentiality, integrity and availability.
- **Prioritize applications.** From this analysis, you will derive the applications that have the greatest exposure or the areas that need the most focus from a security perspective. Essentially, you have the problem set that needs to be solved.
- **Verify all risk mitigation measures.** Once the problem is identified, solutions need to be formulated, recognizing that there are always several ways to solve a given problem. All possible risk-reduction measures must be determined.
- **Do a gap analysis to determine areas of focus.** Using a cost-benefit analysis approach, examine the pros and cons of each mitigation measure, looking at the level of risk reduced by each solution and the relative cost. Also examine collateral risks. Based on all of the comparative data, areas in which hardware-based encryption would provide the best risk reduction solution—within budget and with minimal side effects—can be identified and built into the security road map.

Next we will look at applying the above criteria.

### Mini Case Study

Following a data breach that included the compromise of the personally identifiable information of a significant number of clients, the board of an international financial services organization made a decision—without detailed analysis—to push software-based encryption on all systems that contained any client information.

The approach seemed to address the issue but did not properly protect cryptographic keys, and so the sensitive data was not sufficiently isolated. Because the organization did not understand how the data was accessed or protected, it was compromised again within five months, despite the additional controls. That embarrassment and the resignation of several board members led to a more detailed risk assessment that identified a handful of key servers that needed additional protection. Hardware-based solutions seemed the only viable option that met performance and security requirements. With such a solution, two main options could be used: 1) The hardware could be installed directly on the server, as would be the case for a PCI HSM, or 2) a network-based HSM could be used, where the cryptographic hardware is deployed as a shared networked resource. A cost-benefit analysis that prioritized all the possible solutions highlighted issues about the database architecture that could, with minimal changes in design, allow the hardware-based solution to be properly implemented, reducing the risk to an acceptable level.

# The Criteria

In evaluating and ultimately implementing any technology, it is important to recognize the constraints of the technology: Where does it work well, and what are the limitations? This will lead to the criteria that drive the business decision-making process. Figure 2 shows an overview of the pros and cons of hardware-based encryption.

**Hardware-Based Encryption**

**PROS**

- Increased performance
- Verified algorithms via hardware implementation
- Difficult to tamper with or alter
- Integrates at low level within architecture
- Seamless to high-level user applications

**CONS**

- More expensive
- Requires control of application server
- Potentially requires modification of application server for integration
- Demands careful planning within virtualized environments
- Not as scalable based on reliance on hardware

*Figure 2. Evaluation of Hardware-Based Encryption*

In looking at the criteria, hardware-based encryption works very well in environments in which:

1. Verified encryption is critical.
2. Strong key management and protection are required.
3. Performance is important.
4. Initial deployment cost is not the ultimate driver.
5. The organization has control over the application server environment.

If these factors are not present, then either another option is needed or changes need to be made to the environment.

Many organizations will initially determine that hardware-based encryption is not a viable or feasible solution because of bad design decisions made when the system was first implemented.

*Many organizations will initially determine that hardware-based encryption is not a viable or feasible solution because of bad design decisions made when the system was first implemented.*

These poor designs continue to create problems and limit the ability to expand and provide the level of security that is required.

If a system has outgrown its usefulness and needs to be altered, that problem should be identified as an additional weakness, not a reason to take a secure solution off the list of options.

**Key Terms**

*Verified security*: The ability in a system or network to verify that the proper security and analysis are being performed.

*De-scope*: To limit one's focus to systems of the highest priority.

*Consolidation*: The joining of all the pieces of a system, so that complexity is reduced and functionality of the whole can be evaluated all at once.

This is a very important point worth emphasizing. In one case I have seen, a company planned to spend as much as $3 million to implement security measures that would not have been effective. Only a hardware-based solution would provide the necessary levels of performance, but the environment would not support hardware integration. We proposed a design change at a cost of $750,000 that would enable a hardware solution with a cost of $1.5 million. The result was a higher level of security, good long-term maintenance and an overall cost that was $750,000 lower than the original, ineffective proposal.

When considering hardware-based encryption, the heaviest weighting should be applied to increased performance and verified security.

If software-based encryption or other security solutions are causing bottlenecks or the modification or alteration of the software causes concern, hardware-based encryption can be a solution. In such cases, design criteria, rather than decision criteria, should be applied to any issues related to hardware modification and integration with existing systems. Confusing these two types of criteria or not clearly delineating between them is one of the biggest challenges for our customers. A decision-based criterion is ultimately the go/no go component of making a decision. Essentially, a determination is being made as to whether this is the right solution. Once that has been decided, the design criteria determines and drives the implementation plan for how the technology is going to be rolled out (see Figure 3 on the next page).

**Hardware-Based Encryption Criteria**

**DECISION Criteria**

- Risk reduction – tamper-proof and protection of algorithms
- Verification and validation of encryption algorithms
- Risk reduction – increased protection of keys

**DESIGN Criteria**

- Risk reduction – increased protection keys
- Control and changes that are required to the application server
- Application and business process integration

*Figure 3. Criteria for Driving Hardware-Based Encryption*

In determining, designing, implementing and maintaining hardware-based encryption, one of the most powerful concepts to learn is "de-scoping." Most environments include data storage and tracking mechanisms that are distributed in different locations and among systems of varying types and performance levels. Very few organizations function without leftover legacy systems that are either required by aging applications or were acquired along with subsidiary businesses. Most organizations depend on infrastructures that grew organically and include at least some piecemeal solutions. Individual business processes may include coherent systems development and planning; overall, the integration of an organization's IT infrastructure is more likely to involve at least some organized chaos, leading to an inefficient method of processing and storing information.

De-scoping is the process of reducing the size of a problem—in this case, by consolidating all those piecemeal, jury-rigged integrations to produce a more coherent, controllable IT environment.

Hardware-based solutions can be very effective at securing de-scoped, consolidated, rationalized environments. In situations of organized chaos, hardware-based encryption will either be viewed as a nonviable option or become an implementation nightmare.

Viewing a proper solution as too complicated results from looking at the problem backwards. The correct method of analysis is a risk-based approach using the criteria from Figure 3 to determine which business processes and data stores require hardware-based encryption. Once that has been determined, any necessary de-scoping should be performed to simplify the problem set and make the implementation and maintenance as seamless as possible. With any technology, including hardware-based encryption, if modifications are not made to the environment, the implementation will inevitably be substandard in performance or stability.

**Mini Case Study**

An e-commerce vendor suffered a series of breaches because the solutions it deployed were not effective and had not been designed for the way they were being deployed. A cost-benefit analysis of the entire system presented two conclusions: 1) The application was put together by different people, expanded with no design and, while it worked, not designed properly, and 2) hardware-based encryption was the best solution based on risk reduction. The organization kept overlooking a hardware-based solution because the systems design problem made it seem impractical. With some minor redesign and proper de-scoping to focus on the specific problem set, the hardware-based encryption solution was deployed ahead of schedule and within budget.

# Analytical Process

In performing any analysis, it is always important to understand what is currently in place and where the problems that need to be fixed reside. Figure 4 shows some of the biggest challenges organizations face.

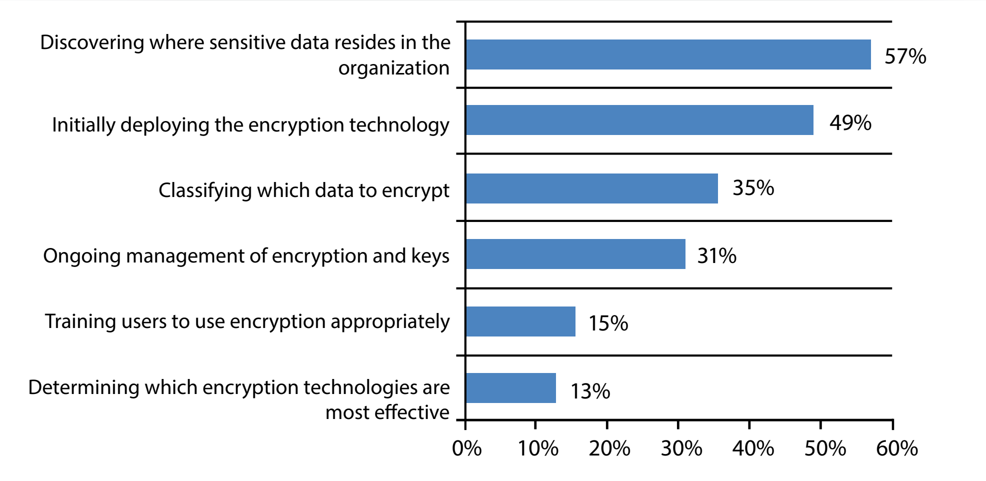
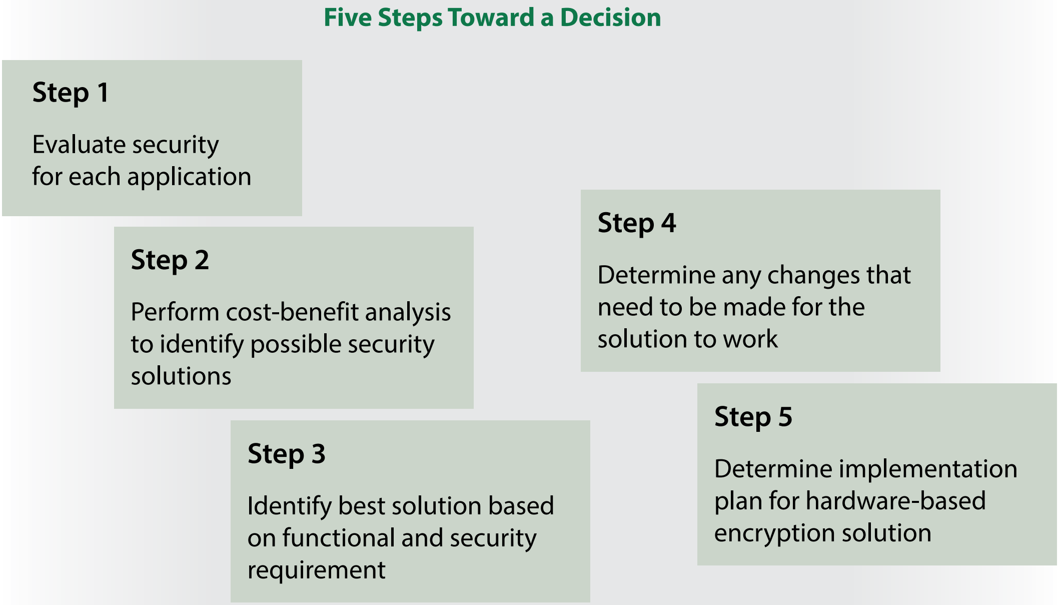


*Figure 4. Challenges in Implementing Hardware-Based Encryption*[2]

In performing an analysis, there are five important steps to take (see Figure 5).



**Five Steps Toward a Decision**

**Step 1**

Evaluate security for each application

**Step 2**

Perform cost-benefit analysis to identify possible security solutions

**Step 3**

Identify best solution based on functional and security requirement

**Step 4**

Determine any changes that need to be made for the solution to work

**Step 5**

Determine implementation plan for hardware-based encryption solution

*Figure 5. Analytical Process for Determining Where to Deploy Hardware-Based Encryption*

[2] 2016 Global Encryption Trends Study," Ponemon Institute, February 2016, www.ponemon.org/library/2016-global-encryption-trends-study

Step 1 is to evaluate all applications, performing a risk assessment to determine which applications are not implemented at an effective level of security. If applications are working correctly with no security issues, there is no need to focus on those; the priority needs to be the problem areas. Step 2 is to look at the possible solutions for each application, performing cost-benefit analysis across each of the various solutions. Step 3 is to determine which solution will meet the functional and security requirements to prioritize and determine what the best solution is. Once it has been determined that hardware-based encryption is the best solution, modifications typically need to be made to the existing applications. Based on the above criteria, an organization would determine how to make this decision. It is critical to perform a detailed evaluation to minimize any surprises and ensure that implementation goes smoothly. Once all of the information has been gathered, the last step is to put together a detailed implementation plan that includes both a schedule and all associated costs with the deployment.

In performing a comparison between alternatives, it is critical to calculate the total cost of ownership (TCO) for each. One of the biggest problems with hardware-based encryption is accounting for all costs required to deploy the solution. Very often, organizations will look at the cost of the device but fail to incorporate all of the other changes and indirect costs that are required. An effective approach is to put together an implementation plan that allows you to detail and better understand TCO; including the vendor in that process gives you access to its experience deploying the solution and knowledge of the technology. The ability to leverage the vendor's knowledge is critical to a successful deployment, especially if it is the first time your organization uses that solution.

*One of the biggest problems with hardware-based encryption is accounting for all costs required to deploy the solution. Very often, organizations will look at the cost of the device but fail to incorporate all of the other changes and indirect costs that are required.*

It is also important to recognize that any solution, including hardware-based encryption, retains some risks. The focus in working with executives is to make sure everyone recognizes what an acceptable level of risk is and prove that the deployed solution will reduce risk to an acceptable level. An example would be a decision on whether to do everything in an HSM or use an HSM only to generate, manage and protect encryption keys in support of a server-based application. In this case, a risk vs. cost analysis is needed. Cases where HSMs are commonly used include database encryption, file encryption, SSL, storage and backups.

There have been many cases where, after deploying hardware-based encryption, a minor incident happens and executives ask how it could have happened, assuming good encryption equals 100 percent security. It is critical to set expectations based on realistic estimates of risk.

# Emerging Technologies

As with any security solution, new ways of deploying the solution are required as new technologies emerge. With hardware-based encryption, the important thing to remember is that it requires application server integration with the appropriate standard APIs, which typically requires control of the devices on which the applications are running. Therefore, some of the new technologies that are emerging can create some interesting challenges, but let's look at how hardware encryption would work in a few of these areas:

- **Cloud.** With the cloud, the hardware and systems are owned and controlled by a third party, and close collaboration with the cloud provider is critical for hardware-based encryption to work. In many cases, cloud providers are looking for ways to enhance their customer experience and differentiate themselves from the competition. Therefore, it is not uncommon for some cloud providers to offer services that integrate hardware-based encryption or to be willing to work with customers to deploy a solution.

- **Virtualization.** Because encryption needs to be deployed within the hardware of a system, it typically cannot be deployed within an individual guest operating system. Virtualizing hardware-based encryption defeats the purpose of implementing crypto within hardware in the first place. Therefore, the virtual systems that require hardware-based encryption need to remain on the same physical box on which the encryption solution is deployed. However, virtual systems can draw on crypto services from a networked HSM. Because many organizations do not really plan the layout of virtualized systems, this often requires some rework, but it is relatively simple to move a virtual system.

- **IoT (Internet of Things).** The IoT typically involves relatively low-end pieces of hardware that focus on a single task, with an emphasis on low power consumption. Therefore, directly integrating hardware-based encryption may not be feasible.  Many vendors, such as Samsung and other IoT platform providers, are making chips with trusted platform modules for which the device manufacturers can leave a slot. In addition, other IoT devices rely on gateways or other points of collection for encryption services.

The most important point to remember is that hardware-based encryption is never impossible due to configuration issues and should never be taken off the list of potential solutions. Implementing a hardware-based solution might require some creativity or redesign, but there are always ways to make it work if it is the most appropriate solution. Also, as the benefit and value of hardware encryption continues to be recognized, more vendors will provide seamless interfaces for third-party devices.

# Conclusion

As performance and security continue to be key drivers for organizations, hardware-based encryption will continue to increase in importance as one of the few solutions that meets both requirements. While typical uses include protecting critical data stores in financial and healthcare companies and industrial control systems, organizations of all shapes and sizes are looking to deploy hardware-based solutions.

The following is a high-level checklist for deploying hardware-based encryption:

- Use risk analysis to drive the overall process of determining an appropriate solution.
- Perform cost-benefit analysis.
- Calculate the TCO to make sure there are no hidden costs.
- Put together a detailed implementation plan to fully understand the complexities involved.
- Recognize that changes to existing applications might be needed to accommodate the best solution.

As organizations continue to store more sensitive data in electronic form and attacks continue to increase, implementing hardware-based encryption can provide organizations the proper balance between performance and security.

# About the Author

**Eric Cole, PhD,** is a SANS faculty fellow, course author and instructor who has served as CTO of McAfee and chief scientist at Lockheed Martin. He is credited on more than 20 patents, sits on several executive advisory boards and is a member of the Center for Strategic and International Studies' Commission on Cybersecurity for the 44th Presidency. Eric's books include *Advanced Persistent Threat*, *Hackers Beware*, *Hiding in Plain Sight*, *Network Security Bible* and *Insider Threat*. As founder of Secure Anchor Consulting, Eric puts his 20-plus years of hands-on security experience to work helping customers build dynamic defenses against advanced threats.

# Sponsor

*SANS would like to thank this paper's sponsor:*

THALES

Last Updated: April 21st, 2018

# Upcoming SANS Training

**Click Here for a full list of all Upcoming SANS Events by Location**

| | | | |
|---|---|---|---|
| **Blue Team Summit & Training 2018** | **Louisville, KYUS** | **Apr 23, 2018 - Apr 30, 2018** | **Live Event** |
| **SANS Riyadh April 2018** | **Riyadh, SA** | **Apr 28, 2018 - May 03, 2018** | **Live Event** |
| **SANS Doha 2018** | **Doha, QA** | **Apr 28, 2018 - May 03, 2018** | **Live Event** |
| **SANS SEC460: Enterprise Threat Beta Two** | **Crystal City, VAUS** | **Apr 30, 2018 - May 05, 2018** | **Live Event** |
| **Automotive Cybersecurity Summit & Training 2018** | **Chicago, ILUS** | **May 01, 2018 - May 08, 2018** | **Live Event** |
| **SANS SEC504 in Thai 2018** | **Bangkok, TH** | **May 07, 2018 - May 12, 2018** | **Live Event** |
| **SANS Security West 2018** | **San Diego, CAUS** | **May 11, 2018 - May 18, 2018** | **Live Event** |
| **SANS Melbourne 2018** | **Melbourne, AU** | **May 14, 2018 - May 26, 2018** | **Live Event** |
| **SANS Northern VA Reston Spring 2018** | **Reston, VAUS** | **May 20, 2018 - May 25, 2018** | **Live Event** |
| **SANS Amsterdam May 2018** | **Amsterdam, NL** | **May 28, 2018 - Jun 02, 2018** | **Live Event** |
| **SANS Atlanta 2018** | **Atlanta, GAUS** | **May 29, 2018 - Jun 03, 2018** | **Live Event** |
| **SANS London June 2018** | **London, GB** | **Jun 04, 2018 - Jun 12, 2018** | **Live Event** |
| **SANS Rocky Mountain 2018** | **Denver, COUS** | **Jun 04, 2018 - Jun 09, 2018** | **Live Event** |
| **SEC487: Open-Source Intel Beta Two** | **Denver, COUS** | **Jun 04, 2018 - Jun 09, 2018** | **Live Event** |
| **DFIR Summit & Training 2018** | **Austin, TXUS** | **Jun 07, 2018 - Jun 14, 2018** | **Live Event** |
| **Cloud INsecurity Summit - Washington DC** | **Crystal City, VAUS** | **Jun 08, 2018 - Jun 08, 2018** | **Live Event** |
| **SANS Milan June 2018** | **Milan, IT** | **Jun 11, 2018 - Jun 16, 2018** | **Live Event** |
| **Cloud INsecurity Summit - Austin** | **Austin, TXUS** | **Jun 11, 2018 - Jun 11, 2018** | **Live Event** |
| **SANS Oslo June 2018** | **Oslo, NO** | **Jun 18, 2018 - Jun 23, 2018** | **Live Event** |
| **SANS ICS Europe Summit and Training 2018** | **Munich, DE** | **Jun 18, 2018 - Jun 23, 2018** | **Live Event** |
| **SANS Cyber Defence Japan 2018** | **Tokyo, JP** | **Jun 18, 2018 - Jun 30, 2018** | **Live Event** |
| **SANS Crystal City 2018** | **Arlington, VAUS** | **Jun 18, 2018 - Jun 23, 2018** | **Live Event** |
| **SANS Philippines 2018** | **Manila, PH** | **Jun 18, 2018 - Jun 23, 2018** | **Live Event** |
| **SANS Vancouver 2018** | **Vancouver, BCCA** | **Jun 25, 2018 - Jun 30, 2018** | **Live Event** |
| **SANS Paris June 2018** | **Paris, FR** | **Jun 25, 2018 - Jun 30, 2018** | **Live Event** |
| **SANS Cyber Defence Canberra 2018** | **Canberra, AU** | **Jun 25, 2018 - Jul 07, 2018** | **Live Event** |
| **SANS Minneapolis 2018** | **Minneapolis, MNUS** | **Jun 25, 2018 - Jun 30, 2018** | **Live Event** |
| **SANS London July 2018** | **London, GB** | **Jul 02, 2018 - Jul 07, 2018** | **Live Event** |
| **SANS Cyber Defence Singapore 2018** | **Singapore, SG** | **Jul 09, 2018 - Jul 14, 2018** | **Live Event** |
| **SANS Charlotte 2018** | **Charlotte, NCUS** | **Jul 09, 2018 - Jul 14, 2018** | **Live Event** |
| **SANSFIRE 2018** | **Washington, DCUS** | **Jul 14, 2018 - Jul 21, 2018** | **Live Event** |
| **SANS Malaysia 2018** | **Kuala Lumpur, MY** | **Jul 16, 2018 - Jul 21, 2018** | **Live Event** |
| **SANS Seattle Spring 2018** | **OnlineWAUS** | **Apr 23, 2018 - Apr 28, 2018** | **Live Event** |
| **SANS OnDemand** | **Books & MP3s OnlyUS** | **Anytime** | **Self Paced** |