

**Ulf Mattsson** is the chief technology officer of Protegrity, a leader in enterprise data security management, where he created the architecture of the Protegrity Data Security Platform. He is considered one of the founding fathers of tokenization and has been advising the industry's top analysts and stakeholders, including the PCI Security Standards Council and Visa, as they navigate the role of tokenization in payment security. Mattsson is the holder of more than 20 patents in the areas of encryption key management, policy-driven data encryption, internal threat protection, data usage control and intrusion prevention. Prior to joining Protegrity, he spent 20 years with IBM working in software development.



**Do you have something to say about this article?**

Visit the *Journal* pages of the ISACA web site ([www.isaca.org/journal](http://www.isaca.org/journal)), find the article, and choose the Comments tab to share your thoughts.

Go directly to the article:



## Choosing the Most Appropriate Data Security Solution for an Organization

With the rising cost and increasing frequency of data security breaches, companies are starting to reevaluate how they protect their data. External and internal breaches have highlighted the need for companies to understand the flow of data within the enterprise and the need to take a more granular approach in terms of how data are secured. This article discusses various data security approaches that can help.

### DATA SECURITY LANDSCAPE

According to a report by the Ponemon Institute, the average cost of a data breach for US companies rose 7 percent from US \$6.8 million in 2009 and to US \$7.2 million in 2010.<sup>1</sup> To say this is a serious issue is an understatement.

Another disturbing trend is the significant jump in malicious attacks companies have experienced in the past two years. According to the Ponemon study, malicious attacks were the main cause of 31 percent of the data breaches examined, up from 24 percent in 2009 and 12 percent in 2008. These attacks also cost companies the most money because they are harder to detect, investigate and contain.

The 2010 annual Verizon *Data Breach Investigations Report*, which is conducted in cooperation with the US Secret Service (USSS), shows a similar increase of malicious attacks as seen in the Ponemon report.<sup>2</sup> As in previous years, the 2010 Verizon report showed that nearly all data were breached from servers and online applications, with 98 percent of all data breaches coming from servers originating from hacking and malware. What makes the increasing number of malicious attacks more alarming is the fact that companies are beginning to implement more robust data security strategies, but as a whole, have not been effective in staving off malicious attacks. Therefore, it is important not to underestimate the skills and determination of today's cybercriminals, who are more than willing

to put in the time and effort necessary to access sensitive data.

Staying ahead of the bad guys is not an easy task; doing so requires organizations to deploy security strategies that protect all sensitive data fields across the entire enterprise.

### A SYSTEMATIC APPROACH TO DATA PROTECTION

The most important rule to follow when creating an effective data-protection strategy is to protect the data themselves, first and foremost. To do this, one must understand the flow of data and build a strategy around protecting the data at the various stages of data flow. This should be done with a risk-adjusted methodology to determine the most appropriate solution for the organization.

To do this, organizations classify data precisely according to risk levels, which are determined by the value of the data and the probability of their exposure. The organization can then develop a sensible plan to invest budget and effort where they matter most.

Security professionals also need to factor in the current favored attack vectors to identify the highest risk areas in the enterprise ecosystem. Right now, web services, databases and data-in-transit are at high risk because they contain valuable data that have a high probability of exposure. Hacking and targeted malware attacks are currently the methods of choice among cybercriminals, who are targeting the application layer and data more than the operating system.

Over the years, data security has evolved to be more granular, developing from perimeter protection, to protection at the storage-device, file, data-field and, now, at the subfield level. This is the next logical step in addressing emerging threats to data, providing strong protection for the smallest unit or subunit of data. This customized approach to data security is also applicable to new regulations in Europe and the US that encompass more types of data, including

## Enjoying this article?

- Consider BMIS.

**[www.isaca.org/bmis](http://www.isaca.org/bmis)**

- Learn more about and collaborate on privacy/data protection.

**[www.isaca.org/topic-privacy-data-protection](http://www.isaca.org/topic-privacy-data-protection)**

personally identifiable information (PII) and protected health information (PHI). These new regulations call for radical enhancements in applicability, flexibility, performance, availability and scalability, and require a highly efficient data security management approach.

There are many ways to protect the data. Hashing, masking, formatted encryption, strong encryption and end-to-end encryption are the most common techniques.

### Hashing

Hash algorithms are one-way functions that turn a message into a fingerprint, which is at least a 20-byte-long binary string to limit the risk of collisions. Hashing can be used to secure data fields in situations in which one does not need to use the original data again, but, unfortunately, a hash will be nontransparent to applications and database schemas since it will require a long binary data-type string. Hashing should be used for passwords, as other solutions are recommended for business data due to transparency and security concerns.

### Masking

Masking is a one-way transformation used to hide or mask information that is presented to users or protected in test databases. Policy-based masking provides the ability to mask selected parts of a sensitive data field. Implemented at the database level rather than at the application level, policy-based data masking provides a consistent level of security across the enterprise without interfering with business operations, and it greatly simplifies data security management chores.

### Formatted Encryption

Formatted encryption is a type of encryption that generates cipher texts of the same length and data type as the input and is typically based on encryption modes that are not standardized. Formatted encryption is known for transparency to applications and databases, and can simplify the process of retrofitting encryption into legacy application environments. It also provides protection while the data fields are in use or in transit, and can be used for lower-risk data and test databases when compliance to industry or government standards (e.g., Payment Card Industry Data Security Standard [PCI DSS], US National Institute of Standards and Technology [NIST] Special Publications [SPs]) is not a factor.

### Strong Encryption

Strong encryption is cryptography based on industry-tested and validated algorithms, along with strong key lengths and proper key-management practices, referencing the NIST SPs.<sup>3</sup> Strong encryption is recommended when encrypting high-risk data. Like formatted encryption, it can also make the process of retrofitting encryption into legacy application environments a lot simpler. However, data fields in transit will be encrypted into a nontransparent binary form, which means strong encryption cannot provide a fully transparent protection while the data fields are in use or in transit.

### End-to-end Encryption

End-to-end encryption provides strong protection of individual data fields by encrypting sensitive data throughout most of their life cycle, from capture to disposal, and it is a great way to protect highly sensitive data that need continuous protection in a data flow. This is an emerging data security method today, and many data security vendors carry this type of solution. As with any other type of encryption, end-to-end uses encryption keys that protect and secure sensitive data based on a mathematical algorithm.

High-risk data are best secured using emerging solutions, such as end-to-end encryption and tokenization, which provide targeted protection for data in use while not affecting system performance.

### INTRODUCING TOKENIZATION

While all of the aforementioned approaches are suitable for data protection, some have proven to be weak when they are tested by more sophisticated hackers or malicious users.

An alternative approach is tokenization. Tokenization is an emerging technology that is becoming more common in the data security world for good reason. Tokenization is the process of protecting sensitive data by replacing it with alias values or tokens that are meaningless to someone who gains unauthorized

**Tokenization is an emerging technology that is becoming more common in the data security world.**

access to the data. Tokenization can be used to protect sensitive information, such as credit card numbers and PII (e.g., health-care-related data, e-mails, passwords, logins).

This emerging technology is known for its flexibility; tokens can travel inside of application

databases and other components without modification because a token, by definition, looks like the original value in data type and length. As a result, transparency is increased, while remediation costs to applications, databases and other components where sensitive data live are greatly reduced.

Effectively implemented, tokenization can significantly reduce an organization's security and PCI DSS-compliance costs. When applications have access to tokens, but have no means to detokenize the data to discover the original value, those applications are considered out of scope. So when factoring in PCI DSS-compliance costs, tokenization provides a more cost-effective alternative to other data security solutions.

Tokenization is ideal when applying a risk-adjusted data security approach because it has an extra layer of security that other data security solutions such as encryption do not have. While encryption is certainly a practical approach to data security, encryption keys are vulnerable to exposure—i.e., if hackers obtain encryption keys or figure out the keys, they can access the sensitive data. Companies must ensure that the encryption method selected is of sufficient strength. Increasing computer power and new cryptologic research will require additional encryption strength over time. This would include the widely used and trusted SHA-1 hashing algorithm (one-way encryption function) that was surprisingly broken a few years ago by a university in China. So, now, NIST is looking for new algorithms beyond SHA-1 and SHA-2.

Tokenization, on the other hand, is based on randomness, not on a mathematical formula, which means that it eliminates the need for keys by replacing sensitive data with random tokens to mitigate the chance of a data breach. If a hacker obtains the tokens, they will have no value. To decipher

the token, a token lookup table is needed, which is stored in a separate secure location. Thus, to obtain sensitive data through tokenization, hackers would have to obtain both the token and the token lookup table, whereas with encryption, a hacker would only need to obtain an encryption key. The extra step with tokenization translates into heightened security against data breaches, regardless of where they originate.

This article will go beyond a discussion about the advantages and disadvantages of tokenization to discuss the different types of tokenization that have emerged in recent years, which vary greatly (**figure 1**).<sup>4</sup>

#### **Dynamic and Static Tokenization**

Dynamic tokenization is characterized by large lookup tables that assign token values to the original, encrypted sensitive data. These tables grow dynamically as they accept untokenized sensitive data, causing the lookup tables to increase in size. Static tokenization is characterized by prepopulated token lookup tables that attempt to reduce the tokenization process by prepopulating lookup tables with anticipated combinations of the original sensitive data.

Both approaches carry large footprints, which introduce a lot of challenges. Large token tables are not agile and lead to latency, poor performance and poor scalability. Dynamic token tables have to be synchronized, an expensive and complex process that may eventually lead to collisions. And, the size of both dynamic and static token tables when tokenizing multiple types of data would soon become an impractical solution should an organization need to tokenize more than one data type, e.g., credit cards, e-mail addresses or health care information.

#### **In-memory Tokenization**

In-memory tokenization is the next step in the evolution of this emerging technology. It is characterized by a small system footprint that compresses and normalizes random data, can be fully distributed, and does not require any kind of data replication or synchronization.

When compared to dynamic and static tokenization, in-memory shows a significant reduction in the performance and scalability issues that are evident with these other approaches. Token servers with small footprints enable the execution of token operations to be in parallel and closer to the data that eliminate latency. In-memory tokenization is based on inexpensive commodity hardware that creates any scaling

**Figure 1—Types of Tokenization**



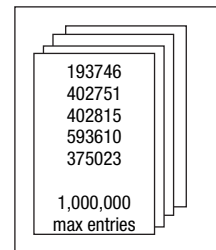
**Dynamic First-generation Tokenization**

- Large, expanding footprint
- Complex replication
- Prone to collisions
- Latency impact on performance
- Expanding to additional categories of tokenizations multiplies the inherent problems.



**Pre-generated Static First-generation Tokenization**

- Large, static footprint
- No replication needed
- No collisions
- Latency impact on performance
- Faster than having to tokenize repeatedly
- Expanding to additional categories of tokenizations multiplies the inherent problems.
- Practical limitations on what can be pre-generated



**In-memory Tokenization**

- Small, static footprint
- No replication needed
- No collisions
- Little or no latency
- Fastest in the industry
- Can work in parallel environments
- Can extend to many categories of data while maintaining a small footprint
- No limitations on what can be tokenized

required by the business, without the need for complex or expensive replication processes, eliminating collisions. Finally, in-memory tokenization can tokenize different data categories without the impact that traditional tokenization methods impose on system performance.

In addition to securing sensitive data, effectively implemented tokenization can also significantly reduce an organization's PCI-compliance costs.

**TOKENIZATION APPLIED**

In theory, tokenization sounds like an exciting data security solution, but it is still an emerging technology. Though not highly publicized, it has been successfully deployed by large enterprises. For example, a large convenience store chain adopted tokenization to protect 50 million card numbers with the goal of fixing some of the challenges it constantly faced during its previous experience with encryption:

- **Management**—Auditors often challenged and examined

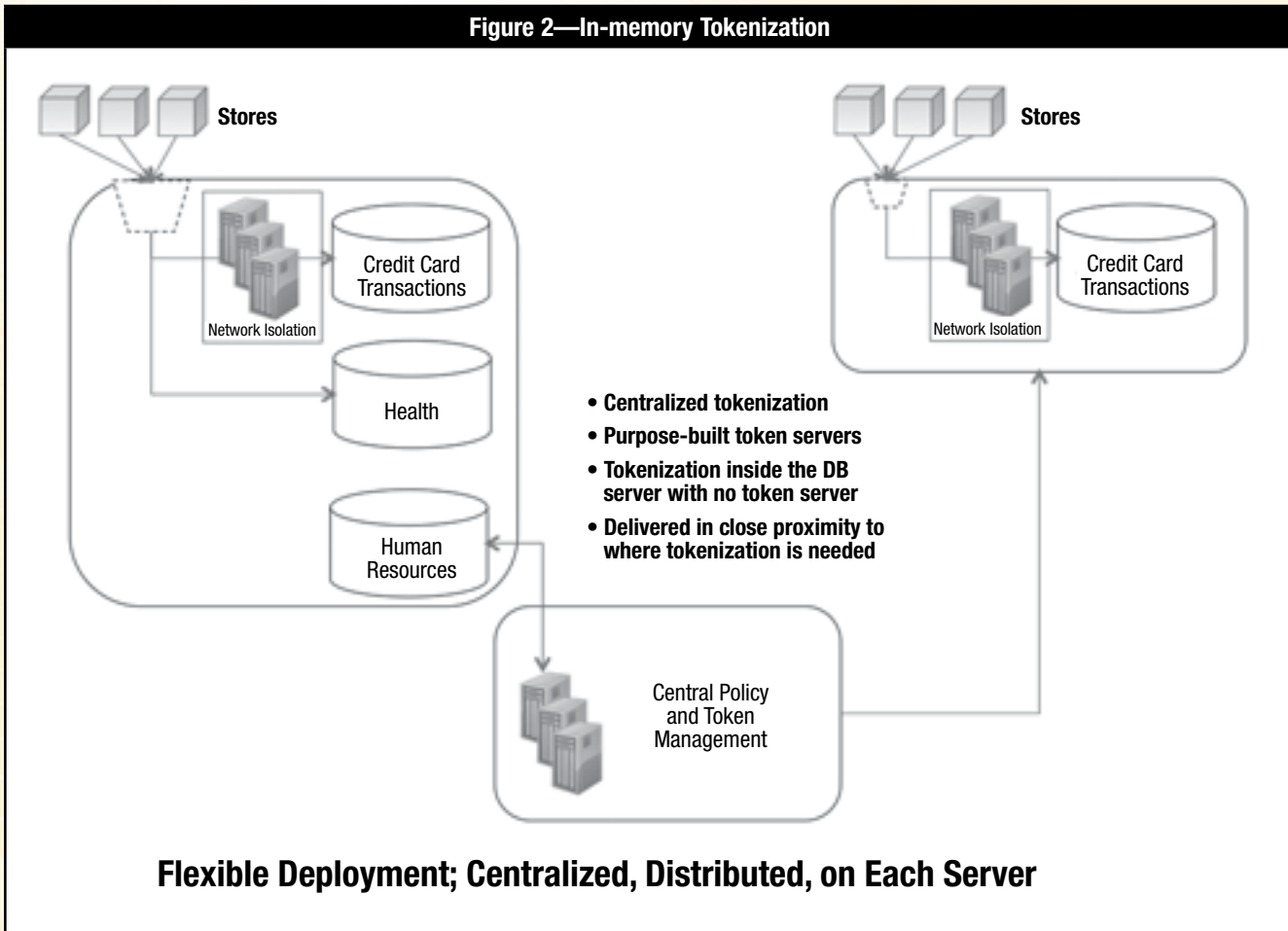
the retailer's encryption key rotation and management. Tokenization offered less controversy as card-processing systems did not use the actual card number. Thus, the retailer was seeking to simplify management's security control.

- **Ease and speed of implementation**—The retailer was overwhelmed by 18 concurrent projects for PCI compliance; therefore, it was imperative that the tokenization solution ease that workload with rapid deployment.
- **Performance**—The retailer's service level agreement for transaction completion was less than one second, which the tokenization solution had to meet.

Initially, the implementation was expected to take about 30 days for 50 million card numbers, but the in-memory tokenization process actually required about 90 minutes. The results of the deployment included:

- **Fast implementation**—The retailer's quality security assessors had no issues with the effective in-memory tokenization segmentation. In addition, the implementation

Figure 2—In-memory Tokenization



did not require any significant changes to the ways the retailer analyzed transactions, ensuring that day-to-day business processes were not interrupted.

- **Faster PCI audit**—In 2010, the retailer’s PCI DSS audit lasted about seven months. Through segmentation, the current audit required only half that time.
- **Lower maintenance cost**—With tokenization, the retailer does not need to apply all 12 requirements of PCI DSS to every system.
- **Improved security**—Tokenization’s layered approach not only heightened security, it also took a lot of sensitive data out of scope, simultaneously heightening security and lowering PCI DSS costs.

- **Strong performance**—In-memory tokenization met the retailer’s subsecond transaction service level agreement.

The retailer is now looking to extend tokenization to the company’s 1,500 retail locations to further reduce the PCI DSS scope of the company’s network and devices.

**CONCLUSION**

Organizations need to understand their data flow and current security technologies when determining which data security solution best fits their needs and the needs of the data types that will ultimately be collected and stored. This approach will enable organizations to determine their most significant security exposures, target their budgets toward addressing the most critical issues, strengthen their security and compliance

profiles, and achieve the right balance between business needs and security demands. This is becoming increasingly important as companies are changing their security strategies to better protect PII following continuing attacks.

“Organizations need to understand their data flow and current security technologies.”

As seen in the previous comparison, in-memory tokenization is the strongest data security solution and should be considered for every situation since it protects sensitive data—ranging from credit card data to birth dates and

e-mail addresses—from almost every type of threat, while enabling systems to continue operating at a high level, even when expanding the number of data types stored. In-memory tokenization is further differentiated from other forms of tokenization because of its lack of latency, performance and scalability issues, and because it is suitable for more types of data and use cases than encryption. Thus, when in-memory tokenization is implemented, an organization will enjoy heightened security for all data types from both internal and external agents while not compromising speed or performance, as sensitive data will be well protected across the enterprise.

## ENDNOTES

- <sup>1</sup> Ponemon Institute, *2010 Annual Study: U.S. Cost of a Data Breach*, Symantec, March 2011, [www.symantec.com/content/en/us/about/media/pdfs/symantec\\_ponemon\\_data\\_breach\\_costs\\_report.pdf?om\\_ext\\_cid=biz\\_socmed\\_twitter\\_facebook\\_marketwire\\_linkedin\\_2011Mar\\_worldwide\\_costofatabreach](http://www.symantec.com/content/en/us/about/media/pdfs/symantec_ponemon_data_breach_costs_report.pdf?om_ext_cid=biz_socmed_twitter_facebook_marketwire_linkedin_2011Mar_worldwide_costofatabreach)
- <sup>2</sup> Verizon Business RISK team, *2010 Data Breach Investigations Report*, 2010, [www.verizonbusiness.com/resources/reports/rp\\_2010-data-breach-report\\_en\\_xg.pdf](http://www.verizonbusiness.com/resources/reports/rp_2010-data-breach-report_en_xg.pdf)
- <sup>3</sup> PCI Security Standards, Council, Glossary, [https://www.pcisecuritystandards.org/security\\_standards/glossary.php](https://www.pcisecuritystandards.org/security_standards/glossary.php)
- <sup>4</sup> For more on the advantages and disadvantages, read: Horton, Tim; “Simplify and Layer Your Security Approach to Protect Card Data,” *ISACA Journal*, volume 1, 2011