Hewlett Packard Enterprise

Business white paper

Streamlining Information Protection Through a Data-centric Security Approach

Table of contents

3 Overview

4 Why Data Needs a New Approach to Protection

5 The Data-centric Approach

6 Demands of Data Protection in Existing Systems

7 HPE Format-Preserving Encryption

8 HPE Secure Stateless Tokenization

9 Static Data Masking

9 Past masking approaches and their challenges

10 Five Steps to Successful Protection of Production Data

12 HPE SecureData Platform Components

13 Simple Integration – A Few Lines of Code for Trusted Applications

15 Example: Static Data Masking to De-Identify Production Data for Testing

16 Example: Implementing Production Data Protection

16 Conclusion

Overview

The sophistication and persistence of criminal attacks on online systems is growing, along with government regulations requiring full disclosure for breaches. The potential compromise to business brand, reputation, and revenues means that data security is no longer optional, but is essential for customer retention and business longevity. Regulatory and compliance requirements bring additional urgency for the need to protect sensitive data.

To date, data protection through encryption, tokenization and masking have been complex and tedious processes. Application and process development is highly complex, IT administration is cumbersome, and projects can take enormous resources and time to complete. With complexity comes risk. Despite technologies being available for many years, database encryption is the exception rather than the rule. Some firms still use high-risk production data in test or outsourced environments. An alarming number of data thefts from breaches have occurred as a result of data exposed in both production and non-production environments[1].

This document introduces a unique approach from HPE SecureData that combines data encryption and masking technology in one, which can vastly simplify data privacy, while mitigating data leakage at a fraction of the cost of prior approaches. One fundamental technology is HPE Format-Preserving Encryption (FPE), which for the first time, allows encryption 'in place' in databases and applications, without significant IT impact. Another technology is tokenization, which replaces data with random tokens, and which can also preserve data formats. These technologies are integrated with masking techniques on the HPE SecureData Platform, allowing projects that once lasted months or years to complete in days to weeks.

HPE SecureData offers a consolidated approach using the above technologies, replacing multiple point solutions with a platform that is agnostic of data storage and operating systems, including convenient delivery and integration options. Both contemporary and legacy enterprise IT systems are readily accommodated, speeding compliance with regulations and standards. Applying HPE SecureData to protect credit card data, for example, can dramatically reduce PCI DSS compliance scope and audit costs. This document covers the use of HPE FPE and HPE Secure Stateless Tokenization (SST) for field-level data protection, as well as both static and real-time data masking.

> "Encrypting or tokenizing data is the future of data security. These technologies effectively "kill" data — making it useless to attackers. Cybercriminals can't monetize tokenized or encrypted data. Plus, breached data that a security professional has tokenized or encrypted may not be subject to state or industry breach laws or regulations. For example, some states offer Safe Harbor if the breached data is encrypted…."
>
> – John Kindervag, Senior Analyst, Forrester Research

[1] See datalossdb.org for the latest breaches.

Why Data Needs a New Approach to Protection

In an ideal world, sensitive data travels in well-defined paths from data repositories to a well-understood set of applications. In this case, the data can be protected by armoring the repository, the links, and the applications using point solutions such as transparent database encryption and SSL network connections.

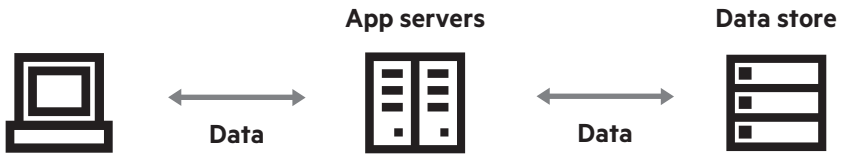


Figure 1. Ideal data path for traditional protection approach.

In real systems, data travels everywhere. Today's IT environment consists of a constantly shifting set of applications running on an evolving set of platforms. In large enterprises, the data lifecycle is complex and extends beyond the container and application, sometimes outside traditional enterprise IT departments into places like offsite backup services, cloud analytic systems, and outsourced service providers. For transactions involving personal and payment identifiers, many applications must be coordinated to protect the data.

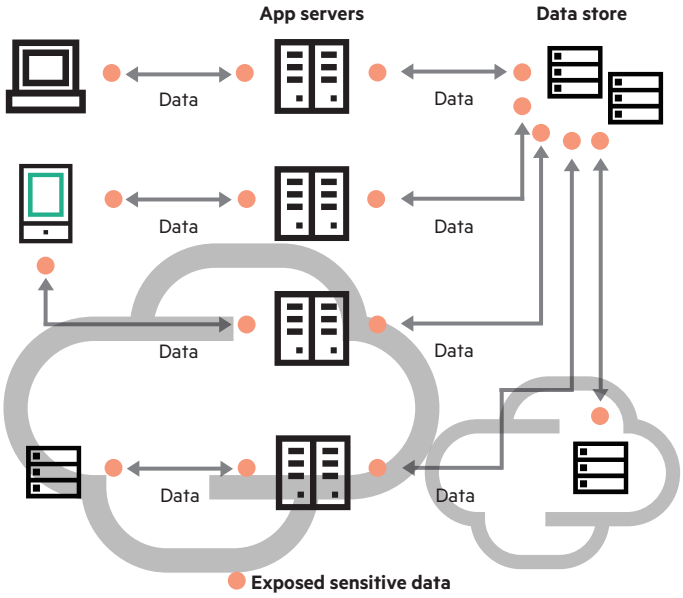


Figure 2. Security gaps in real-world IT environment when traditional data protection approach is deployed.

This means that armoring the repositories, applications and links doesn't provide the needed protection, because the data won't stay in one place. Even if you could manage to keep up with the rapid changes in infrastructure by installing and managing security solutions from a wide range of vendors, you will have security gaps in between the armored repositories, applications and links. For example, as shown by the red dots in Figure 2, data is exposed after it is decrypted and retrieved from a transparently encrypted database and before it flows through an encrypted link, leaving it vulnerable to an attack. Consequently, legacy security solutions have failed to deliver and have been removed, bypassed or applied unevenly in many businesses. The results could not be clearer: breaches involving unprotected business and customer data are front page news almost every day, with disastrous consequences[2].

[2] See datalossdb.org for the latest breaches.

The following illustrates the weakness of conventional approaches to data protection.

Whole database encryption	• Encrypt data within DB – slows all apps down • No granular access control • Separate solution for each database vendor • No separation of duties – DBA can decrypt • No security of data within applications and networks
Database column encryption	• Encrypt data via trigger and stored procedure • Require schema changes • No data masking support or separation of duties
Native or traditional application-level encryption	• Encrypt data itself, throughout lifecycle • Requires DB schema/app format changes • Heavy implementation cost
Shuffling	• Shuffle existing data rows so data doesn't match up • Breaks referential integrity • Can still leak data
Data tables and rules	• Consistently map original data to fake data • Allows for referential integrity, reversibility • Security risks due to use of look-up tables
Weak, breakable encryption	• E.g., stream ciphers, alphabetic substitution • Not secure – easily reversible by attacker • Key management challenges

The Data-centric Approach

HPE Security - Data Security has pioneered technology that protects data independent of the subsystems that use it. HPE SecureData protects sensitive data as soon as it is acquired and ensures that it is always used, transferred and stored in protected form. Selected applications decrypt the data only at the time that it is processed, while others work with encrypted or masked data.

HPE SecureData provides two technologies for protecting data: HPE Format-Preserving Encryption (FPE), and HPE Secure Stateless Tokenization (SST). These independent methods are proven to protect data while preserving data format and other attributes, effectively building the protection into the data itself. Replacing the original data with either an encrypted value or a random token narrows the possible exposure of data and can greatly reduce audit scope and compliance costs. Figure 3 below illustrates this, with an implementation example.

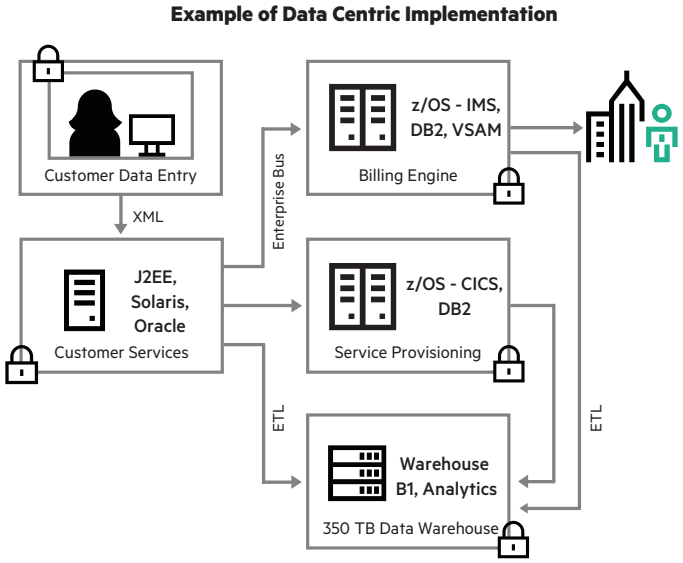


Figure 3. How HPE SecureData protects data at each stage in its life-cycle.

In contrast to typical methods of data protection, HPE SecureData customers, including national and global financial, retail, healthcare and telecom enterprises, and government agencies, have observed the following results:

TYPICAL DATA PROTECTION ROADBLOCK	PAST APPROACHES	DATA-CENTRIC APPROACH
Time to pilot	30 to 80 days	5-10 days
Performance overhead	Added 2.5 hours to batch already lasting 11 hours	Less than 10 minute batch overhead, zero overhead in many cases
Scope of PCI audit	Wide audit scope, to all application systems	Minimized PCI audit scope
Segregation of duties	Mingles IT and application access	Full separation using existing identity management
Time to implement in applications and databases	6-9 months	From 1 week – varies by application size but more than 50% reduction in time and effort
Impact on trusted applications	Substantial new application code	A few lines of new code per application
Impact on untrusted applications with de-identified data	Substantial new application code	No change to application code
Expertise needed to deploy and manage	Cryptography, DBA, performance specialist	Standard app developers
Integration with legacy environments (like Vax, Tandem, Mainframe)	Forced upgrade, high integration costs, often no support	Agnostic of IT, databases, application environments
Staffing overhead	1 specialist staff per data center	0.1 Full-Time Employee (FTE) per data center
IT resistance	Requires DBA, IT process changes	Minimal changes – transparent, simple

Figure 4. Comparison of past approaches to database encryption versus the HPE SecureData approach.

With HPE SecureData, an enterprise can enable data privacy as a service across applications in a way that is seamless to users. The implementation typically results in a 2-5X cost saving and 2-5X reduction in time-to-market over legacy technologies.

HPE Security - Data Security products are designed to accelerate data privacy compliance to PCI, HIPAA, GLBA, PIPEDA, Basel II, SEC 17, SOX, SB1386, NY SSN Reduction laws, US State and Federal, EU, Japanese, Australian, and international data privacy regulations.

Demands of Data Protection in Existing Systems

There are special demands that must be met when implementing a data protection solution that leverages existing systems without major disruption.

The first demand is referential integrity. It is common that the same identifying data is present across multiple databases and application systems. Applications depend upon the pervasiveness of common identification data, such as credit card numbers or social security numbers (SSN). These data must be stored with consistent values to allow matching across databases.

It is a challenge to maintain referential integrity in encrypted data. Consider an example with three separate databases (potentially on different platforms), using common data such as SSN to access records in the database. If we encrypt one database's SSN field, then we have lost referential integrity across the different databases, as the encrypted SSN field will appear as random binary data. The databases and applications will lose the ability to link and index tables using the SSN, causing operational failure.

Therefore data protection must be coordinated across databases. The data inside the database must be consistent, providing unique identifiers, so that data can be linked before being presented to applications.

Another demand of data protection in existing systems is format preservation. Identifiers have specific formats, with definite lengths, and sometimes, punctuation.

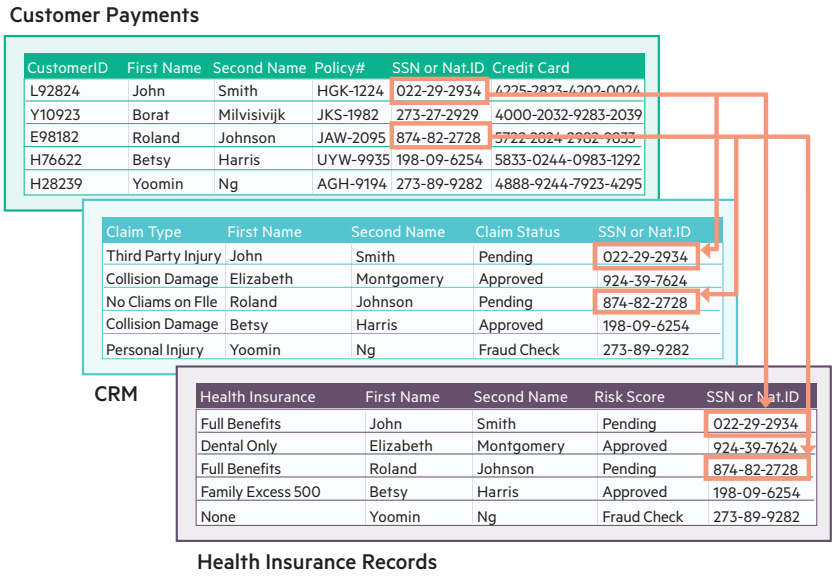


Figure 5. Example of referential integrity. A SSN or National ID links three databases. Indexing, searching and "joins" rely on referential integrity

Applications are written with these formats built into their code base in many areas – the definitions of variables, the allocation of temporary space, the layout of user interfaces, etc. When protecting data, it is critical that the format of the original data be preserved; otherwise applications would have to be re-written and processes may have to be changed, at great expense. The HPE SecureData platform provides four techniques that can be combined to meet the demands of data protection in any setting. These are encryption, tokenization, static data masking, and real-time data masking.

HPE Format-Preserving Encryption

HPE SecureData provides HPE FPE using AES-256 encryption. HPE FPE combines a novel, published method (see FFX Encryption Mode on the US Government NIST website) with an existing, proven encryption algorithm (AES) to encrypt data in a way that does not alter the data format. Like traditional AES, the HPE FPE algorithm uses strong 256 bit keys, and like AES, with the ciphertext and the original key, an application can get back the unencrypted value. A variation of this technology allows the identity and access policy data to be embedded within the cipher text.

The fact that the encrypted value has the same size and data format as the original enables HPE FPE to be used with little or no changes to database schemas and applications. And inherent to how HPE FPE works, when encrypted values are transported from mainframes to open systems, no EBCDIC to ASCII conversion is required.

HPE SecureData employs standards based encryption methods. In HPE SecureData, the HPE FPE functions utilize AES-256. HPE FPE is published as mode of AES on the US Government National Institute of Standards (NIST) Website for AES modes development, as Feistel Finite Set Encryption Mode FFSEM, extended as FFXNIST. For tokenization, HPE SecureData utilizes NIST 800-57 AES Cipher Block Chaining (CBC) mode in the token generation table. HPE Security - Data Security contributes to new standards as they are developed.

HPE Secure Stateless Tokenization

HPE SecureData also provides tokenization. Tokenization replaces data values with a "token," or random string of text. HPE Secure Stateless Tokenization (SST) technology is an advanced, patent pending, data security solution that provides enterprises, merchants and payment processors with a new approach to help assure protection for payment card data. HPE SST technology is "stateless" because it eliminates the token database which is central to other tokenization solutions, and removes the need for storage of cardholder or other sensitive data. HPE SST uses a set of static, pre-generated tables containing random numbers created using a FIPS random number generator. These static tables reside on virtual "appliances"– commodity servers – and are used to consistently produce a unique, random token for each clear text Primary Account Number (PAN) input, resulting in a token that has no relationship to the original PAN. No token database is required with SST technology, thus improving the speed, scalability, security and manageability of the tokenization process. Tokenization has a special advantage for credit card numbers: the PCI DSS guidelines consider systems that only hold tokens to be out of audit scope, greatly reducing audit costs.

	Credit Card 0012 3456 7890 0000	Tax ID 000-00-0000	Bank Account 122105278 724301068
FPE	0012 34**23 3526**0000 0012 34**Ax 3YPX** 0000 (with embedded policy option)	982-28-7723	122**10234 827345**1068
AES	8juYE62W%UWjaks&d DFeruga2345^WFLER	Ija&2924kUEF65%Qar otugDF2390^32KNqL	Hiu97NMko2^Ku}o{35 RJ434DQNmnSDreK23

Figure 6. FPE intelligently preserves format and referential integrity. By contrast, traditional AES is longer and requires large-scale changes to applications and database schemas.

In HPE SecureData, the tokens have the same format as the original data, gaining all the advantages of FPE. Specifically, both FPE and HPE SST have the following properties:

- Format can be exactly preserved, such as a 9 digit SSN becoming a 9 digit token, or it can be altered, such as a 16 digit credit card number becoming a 16 character string with some digits replaced by alpha characters – to assist auditors in immediately recognizing the difference between a token and a real credit card number.
- They are deterministic, which means that the same input, encrypted or tokenized twice, will result in the same output. This feature enables preservation of referential integrity, without the need to keep an application-specific reference database.
- Because they are reversible, they guarantee against collisions (for each input, there is one and only one output, and vice-versa).

Static Data Masking

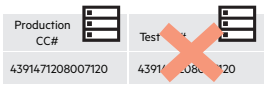
The properties of FPE described above can also be employed to generate test data based on production data. The process of converting a production data set into de-identified test data is called "static data masking." FPE can be configured for both reversible and non-reversible data masking. In reversible mode, the encryption key is centrally generated and managed, allowing recovery of the original data when required. In a non-reversible, or one-way mode, an ephemeral encryption key is randomly generated for each encryption and subsequently thrown away. Both techniques can be useful for QA test data. Reversibility is important in scenarios such as:

- Medical researchers need "blind" data but occasionally an actual patient's identity must be uncovered by an authorized person.
- Trading partners require a subset of test data, in original clear text form.
- A problem occurs in production but cannot be reproduced with masked data.

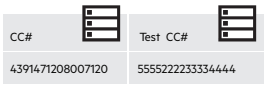
In the past, masking processes would lose relationships across databases, or would be very complex to manage with special rules or tables, or would require substantial storage as lookup tables as large as the original databases were required. Thus, additional terabyte SANs were required just for storage of masked datasets. FPE provides static data masking capabilities without the large lookup-tables filled with sensitive data that are used in traditional data masking solutions. The following table illustrates past masking approaches and their challenges.

Past masking approaches and their challenges

Real data – no masking at all
- Compliance problems – data leakage risk
- Challenge: PCI DSS violation limits scope of outsourcing relationships



Database column encryption
- Replace e.g. SSN with random data
- Challenge: no referential integrity, complaints from QA – data needs to be managed



Native or traditional application-level encryption
- Requires additional copy of database – increased risk
- Increased resources and complaints from QA and DBA
- Challenges: referential integrity, costly management

Shuffling
- E.g. stream ciphers, "alphabet substitution"
- Challenges: easy to break (minutes), format not fully preserved, data leakage risk

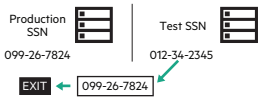


Figure 7. Legacy masking approaches and their deficiencies.

Different applications have different data needs. HPE SecureData supports a powerful feature, run-time data masking, which allows different applications to meet their information needs with a run-time choice of data mask. Data is only exposed on a "need-to-know" basis. Credit card numbers provide a good example. Analytics users do not need the original numbers, but they do need unique identifiers or tokens that are used consistently. Customer Relationship Management (CRM) users may need only the last 4 digits of the actual number with the other digits masked. QA application testers need unique IDs or tokens, with some of the original digits preserved for routing and load management. Only final payment processing systems and fraud auditors need the original unencrypted data. In effect, each application sees the data through its own specific mask, allowing for very precise control of data security.

Top Performance
A single HPE SecureData Web Services server running on commodity Intel-based hardware can handle hundreds of thousands of transactions per second. Scaling beyond that level is simple with multiple load balanced servers. In Teradata data warehouse deployments, millions of transactions per second have been achieved.

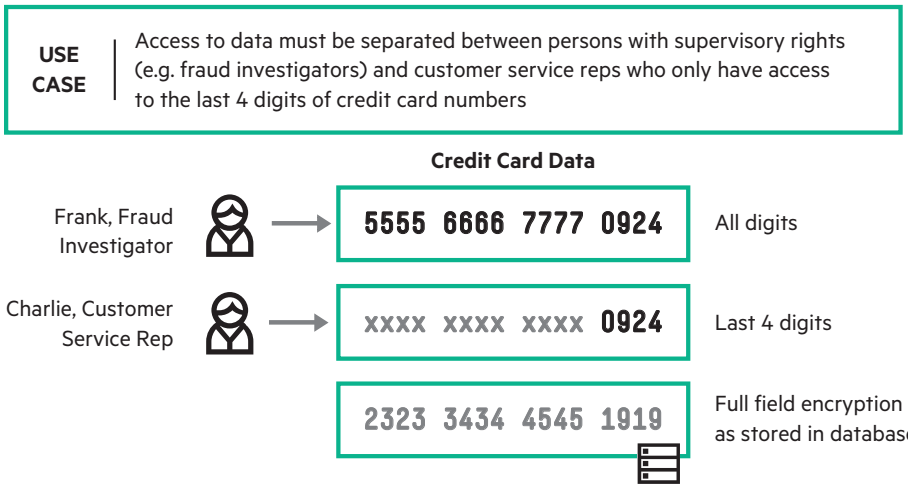


Figure 8. HPE SecureData provides granular access to sub-fields in a database, based on the needs of the application or the identity of the user.

In the past, developers might hard-code real-time masking of sub-fields, such as hiding the first five digits of the common social security number, e.g. XXX-XX-2373. Coding this capability into the application has a number of disadvantages including potential privacy and regulatory violations. By contrast HPE SecureData masks the values immediately before they are delivered to the application. Even if there are logic or coding errors in the application, protected information cannot be revealed.

HPE SecureData provides masked data at run-time from data stores, with central control over masking policy based on user roles. Its design gets the benefits of data masking without the drawbacks. HPE SecureData offers run-time masking for both FPE and tokenized data.

How Did They Do It?
Global Telcom Provider

Business Drivers:
- Compliance cost reduction, brand risk and breach mitigation. Covered by nearly every privacy regulation: PCI, HIPAA, state privacy laws, etc.

Situation:
- 500 applications with petabytes of sensitive data; 26 data types to protect
- Disparate systems and platforms: mainframe, open systems, custom built apps, packaged apps, Oracle, DB2, Teradata, Unix, IMS, J2EE (Websphere, WebLogic), HPE NonStop, Hadoop

Solution:
- HPE SecureData for enterprise wide data protection
- HPE FPE with embedded policy is corporate standard
- Deploying at ~15-30 applications per month; currently protecting data in more than 3,000 databases

Five Steps to Successful Protection of Production Data

HPE SecureData leverages HPE FPE and HPE SST to provide a complete solution for data protection. With centralized management and many interfaces for performing the actual data masking, HPE SecureData provides an integrated solution that provides rapid results. Here are the actions required for a complete deployment:

1. Identify the data elements to protect, and choose HPE FPE or HPE SST
2. Define application identities to tie a decryption method to each application
3. Establish central administration across a distributed installation of HPE SecureData
4. Verify that untrusted applications require no change
5. Install small code changes for trusted and masked applications

The first step is to identify the highest-priority type of data where you can show immediate results. Personal identification data such as SSN, credit cards, account codes, policy numbers, personal identification numbers and so on, are a natural place to start. Then choose the protection methods that fit your needs, either HPE FPE or HPE SST, plus masking when appropriate.

USE CASE	PROTECTION TYPE
Ensuring PCI compliance, minimizing PCI audit scope	HPE Secure Stateless Tokenization
Brand risk and breach mitigation, protecting PII data	HPE Format-Preserving Encryption
Scope of PCI audit	Wide audit scope, to all application systems

How Did They Do It?
Global Credit Card Issuer & Service

Business Drivers:
- PCI Compliance and PCI audit scope reduction in both UK and US operations

Situation:
- Acute PCI compliance challenges with lots of data cross legacy mainframes, Teradata, Oracle apps, open systems and hundreds of applications

Solution:
- HPE SecureData Enterprise for tokenization and end-to-end encryption of PCI and PII data from one platform
- Phased deployment with a global systems integrator

Next, inventory the applications that rely upon this data, and which would benefit from improved data protection. These may be systems that are currently in PCI DSS audit scope which could be removed from scope, such as marketing analytics or QA systems. You will give each application a name that will associate it with its encryption keys.

Then, when you install HPE SecureData, you will link it to your enterprise identity management system, such as Active Directory, so that the appropriate security staff can configure it and maintain it. The web interface offers interactive set-up for all management functions.

Next, verify that certain applications can function unchanged, using encrypted data. In many use cases, this will be the majority of applications where the data flows. These untrusted applications should continue to function "as is" – they will get protected data in the same format as before, possibly with selected digits unaltered.

Finally you will integrate HPE SecureData with those applications which need access to either fully decrypted data or partially decrypted – real-time masked data. For example, an application may require a full SSN for an ID verification. The integration may be done at the database layer, pointing the applications to masked views of the protected data, or the integration may be done at the application layer. The changes required to application code are typically very small, adding as little as one line of code. All the authentication, key management, and operational complexity is abstracted into a web service, a native API call, or a command line call. Details on these integrations are supplied in the next few sections.

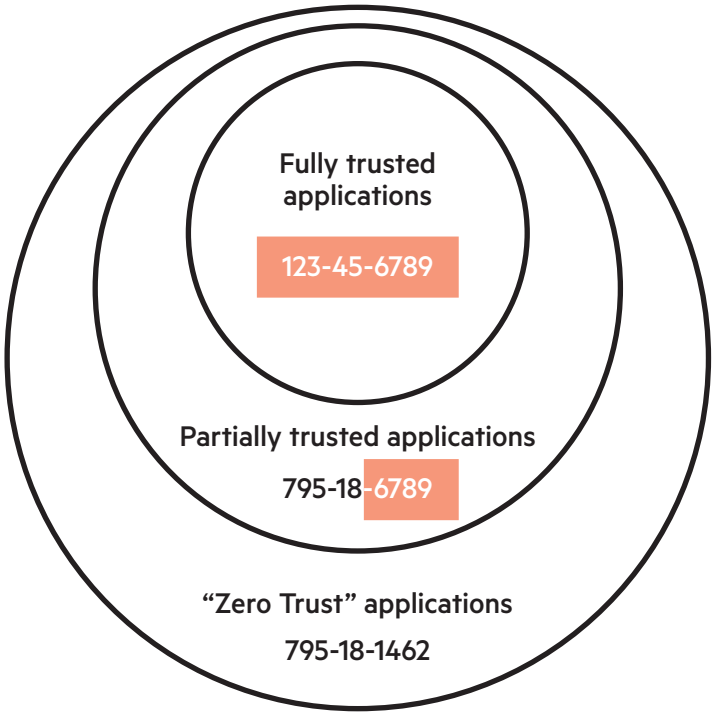


Figure 9. With HPE SecureData, permission to access parts of the data can vary between applications

HPE SecureData customers have successfully integrated with both custom in-house applications as well as off-the-shelf enterprise applications. Examples include: Peoplesoft, Informatica (ETL), Ab Initio (ETL) and XML gateways fronting a variety of applications.

Typical pilot installations take a few days. You may then begin to apply HPE SecureData to other data fields and applications. Adding permission to access data is as simple as managing a group or role in LDAP—no need to adjust policy in the applications.

HPE SecureData Platform Components

HPE SecureData delivers information encryption services through a central core platform. This platform provides a robust management and deployment framework for addressing the data privacy needs for data at rest, data in motion, and data in use across multiple application areas. Overall, the platform is designed for centralized management with a high degree of automation to simplify operations. Each element also supports its own specialized functions:

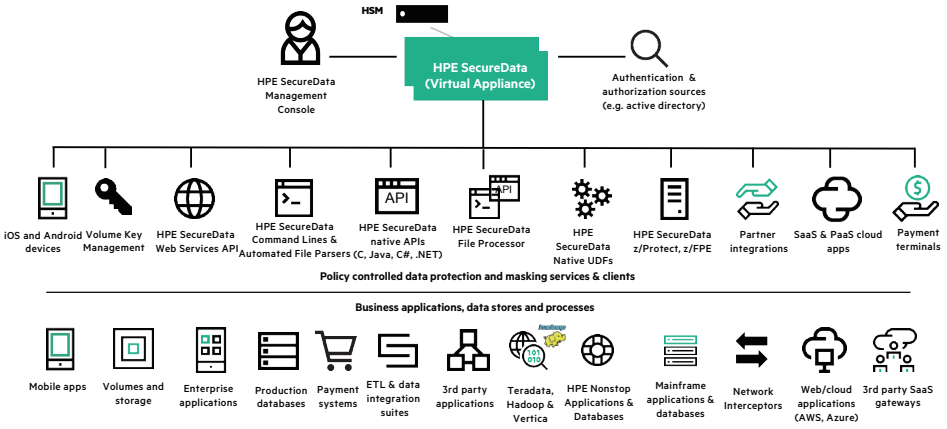


Figure 10. HPE SecureData component architecture.

How Did They Do It?
Fortune 50 Global Financial Group

Business Drivers:
- Compliance cost reduction, brand risk and breach mitigation, extending offshore outsourcing. Covered by nearly every privacy regulation: GLBA, SEC, PCI, HIPAA, State privacy laws, etc.

Situation:
- Terabytes of regulated data across hundreds of applications, with complex data governance issues
- Sophisticated financial applications dependent on data – e.g. data relationships for risk calculations
- Applications include mainframe to web-based

Solution:
- HPE SecureData for enterprise wide deployment for de-identification and data protection

- **HPE SecureData Management Console:** Enforces data access and key management policies, and eliminates the need to configure each application, because flexible policies are centrally defined and reach all affected applications. Manages data format policies, business rules enforcement over data access, integration with enterprise authorization and authentication systems and connectivity to enterprise audit and security event monitoring systems. It also manages data security policies such as the choice of HPE FPE, file encryption and data masking.
- **HPE Key Management Server:** Eliminates the need for traditional complex storage-based key management and storage because keys are dynamically derived; seamlessly integrates with existing Identity Management and Authorization Systems and Key Management using FIPS 140-2 certified Hardware Security Modules.
- **HPE SecureData Web Services Server:** Centralized web services encryption and tokenization option for Service Oriented Architecture environments, enterprise applications and middleware.
- **HPE SecureData Simple API:** Maximizes efficiency on a broad range of application servers through native encryption on HPE/UX, HPE NonStop, Solaris, Stratus OS, Linux (Red Hat, SUSE), AIX, Windows, CentOS, Teradata, and a variety of payment terminal devices.
- **HPE SecureData Command Lines:** Scriptable tools easily integrate bulk encryption, tokenization and file encryption into existing batch operations and applications.
- **HPE SecureData File Processor:** Aggregates support for both tokenization and encryption of sensitive data elements. It provides a unique value to the customer as a single client converging both web services and native API interfaces. The converged clients expand the support for new file types by decoupling input file processing from the underlying encryption and tokenization operations. Delivers high performance data de-identification, with parallel multi-threaded processing of sensitive data elements simultaneously protecting data fields across columns.

- **HPE SecureData Enterprise also supports mainframe, Big Data, and payment security ecosystems:**
  - **HPE SecureData z/Protect:** Maximizes CPU performance on mainframe systems through native z/OS support for encryption and tokenization.
  - **HPE SecureData z/FPE:** Mainframe data processing tool to fast track integration into complex record management systems such as VSAM, QSAM, DB2 and custom formats. De-identify sensitive data for production as well as test use.
  - **HPE SecureData for Hadoop Developer Templates:** Provides templates to enable customers to integrate HPE FPE and HPE SST technologies into their Hadoop instances. Templates come ready to use out-of-the-box for Sqoop, MapReduce and Hive, and can be quickly expanded to integrate into other technologies in the Hadoop stack such as Flume.
  - **HPE SecureStorage:** Data-at-rest encryption for Linux with HPE Stateless Key Management.
  - **HPE SecureData Web and Optional Add-ons:** Secures data end-to-end from browser applications and forms to secure back-end applications, extending end-to-end security beyond transport encryption such as SSL and TLS.
  - **HPE SecureData Terminal SDK and Host SDK:** Provide market-leading P2PE payments security.
- **HPE Professional Services:** Available to help clients scope projects, to combat advanced threats, reduce compliance burden and to quickly solve difficult data privacy challenges.

Platform growth is easily accommodated. HPE SecureData servers can be distributed around the enterprise network as appropriate for scaling and for disaster recovery. Monitoring and reporting are easy: HPE SecureData incorporates best-of-breed Splunk event management software for centralized, high level, and real-time inspection and analysis. Or events can be sent to an external syslog server.

The platform can also be extended to protect unstructured data such as files and bulk data with HPE SecureFile. Utilizing HPE Identity-Based Encryption (IBE), files and bulk data can be secured on the fly for any system, recipient or group in an ad hoc manner without the traditional problem of having to issue and manage encryption keys for every endpoint. HPE SecureFile uses the same management servers as HPE SecureData, with the same wide range of programmatic interfaces.

Simple Integration – A Few Lines of Code for Trusted Applications

In the past, application developers would need to know cryptography and key management in order to build encryption into applications. Toolkits would require complex coding and testing, and integration efforts would need deep expertise and lots of code, increasing the chance of mistakes, and complicating QA processes. Also, PCI and other costly audits would have to review code every year. Today, HPE SecureData simplifies the integration process and moves the developer away from this complexity. Adding SDK calls to applications is a simple process for everyday programmers or application developers. HPE SecureData offers five high-level interfaces.

HPE SecureData supports these environments:

AIX	Amazon Web Services
CentOS Linux	Hadoop
z/OS	HPE NonStop
HPE UX	Oracle
RHEL Linux	Solaris
Stratus VOS	SuSE Linux
Teradata	VMWare
Windows	
Hardware Security Modules	

And these languages:

C, C++	C#
COBOL	Python
.NET, .ASP	Visual Basic
Java	

Other platforms and languages supported upon request.

HPE SecureData Web Services Server

The HPE SecureData solution provides a web services option through the HPE SecureData Web Services Server. This component provides a high-level encryption and tokenization API that can be accessed through a standard SOAP interface. This design allows encryption, tokenization and data masking to be performed from nearly any platform, including legacy mainframe environments. Both individual data elements and bulk data are supported. Integration takes just a few lines of code in most languages.

Web Service calls can also be made from within databases such as Oracle, DB2, SQL,and Sybase and so on. This allows encryption and masking to be performed from stored procedures and database triggers, without application-level code changes. As there are numerous variations in databases by vendor and version, implementation of this approach is typically accompanied by professional services from HPE Security - Data Security or integration partners. The HPE SecureData Web Services Server can also be called from Extract-Transform-Load (ETL) tools, to allow "in transformation" real time processing of data into the database or data warehouse. Simple implementation papers are available from HPE Security - Data Security.

```
VibeSimpleSOAPStub service  = (VibeSimpleSOAPStub) new
VibeSimple_ServiceLocator( ). getVibeSimpleSOAP( );

String ccNum = "43291471208007120";
String keyName = "pci@company.com";

String encryptedCC = service.vibeProtectCreditCard (ccNum,
NULL, keyName, NULL, "UserPassword", "user:pass");
```

Figure 11. Example Java code to encrypt a credit card, where a single additional call to HPE SecureData provides many privacy features.

The example above, in Java, shows a simple call to the HPE SecureData Web Service for a credit card example.

A Complete SDK
In addition to the high-level interfaces detailed here, the HPE SecureData SDK also provides functions that allow developers to extend to low level cryptography features if required. These include straight AES encryption, RSA, IBE and other operations. However, in nearly all cases, this will not be required and application changes will only be a few lines of high-level code.

HPE SecureData Command Line

The HPE SecureData system includes a powerful multi-platform command line tool called HPE SecureData Command Line (CL). It provides encryption and tokenization capabilities through a simple scripting interface for automated, repeatable data protection and masking HPE SecureData CL supports both reversible and non-reversible masking, and can operate on both individual data elements and files of bulk data (such as CSV or COBOL Copybook files).

HPE SecureData CL also includes advanced conditional encryption capabilities, which allow for policy-driven encryption across large data sets. For example, an insurance dataset containing two columns, a carrier ID and a policy number, could be masked in such a way that certain carrier policies are reversibly masked, while others are non-reversibly masked, or even left in the clear.

HPE SecureData Simple API – A Native Encryption Toolkit

If encryption operations are required directly within application code, or if extremely high performance is required, the HPE SecureData solution offers a native C/C++, Java and .NET encryption toolkit called HPE SecureData Simple API.

HPE SecureData z/Protect – For z/OS Mainframe

HPE SecureData z/Protect provides fully compatible encryption services across all z/OS environments, including Customer Information Control System (CICS). It also provides role-based data access, which is impossible with traditional all-or-nothing full database encryption. With z/Protect, key access is controlled using native z/OS security methodologies (RACF, ACF2, Top Secret). This avoids the need for applications to store credentials, further reducing the exposure of sensitive information for hackers to steal.

HPE SecureData for Teradata
HPE SecureData for Teradata provides native encryption and masking in the Teradata data warehouse. This drastically reduces exposure of data and helps mitigate risks of breaches. HPE SecureData for Teradata installs once, and its User Defined Functions (UDFs) are automatically made available across hundreds of Teradata nodes.

These UDFs simplify data protection natively on Teradata nodes, as they are easily incorporated in SQL queries, triggers and views. The native implementation of HPE SecureData within Teradata allows data protection to be applied with a small change to a single SQL statement, or no change when views are used.

Example: Static Data Masking to De-Identify Production Data for Testing

There are two methods of producing realistic test data.

- **Direct Integration:** Transform sensitive fields of the data "on-the-fly" as it is being extracted from a production database. An existing extract-transform-load (ETL) tool or a database stored procedure can call one of the HPE Security Voltage APIs to mask the data on its way to its destination database or file.
- **Indirect Integration:** Extract the production data to a staging area first – either in a file or a database. Run HPE SecureData Command Line to transform sensitive fields "in bulk" within the staging area. The data is then ready for test use.

In both cases centrally defined masking rules for each data type are verifiably enforced by the HPE Key Management Server.

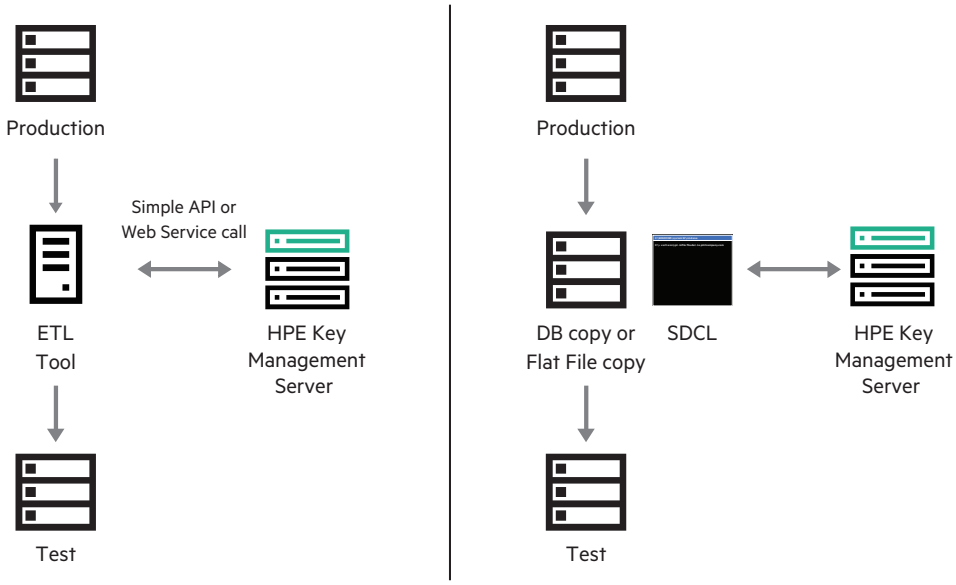


Figure 12. Direct (left) versus Indirect (right) integration of HPE SecureData for masking test data.

Example: Implementing Production Data Protection

The figure below illustrates how data protection might be implemented across the enterprise to protect U.S. social security numbers. This removes the need for separate data protection solutions in each environment such as Oracle, z/OS and Teradata. HPE SecureData protects the data wherever it goes.

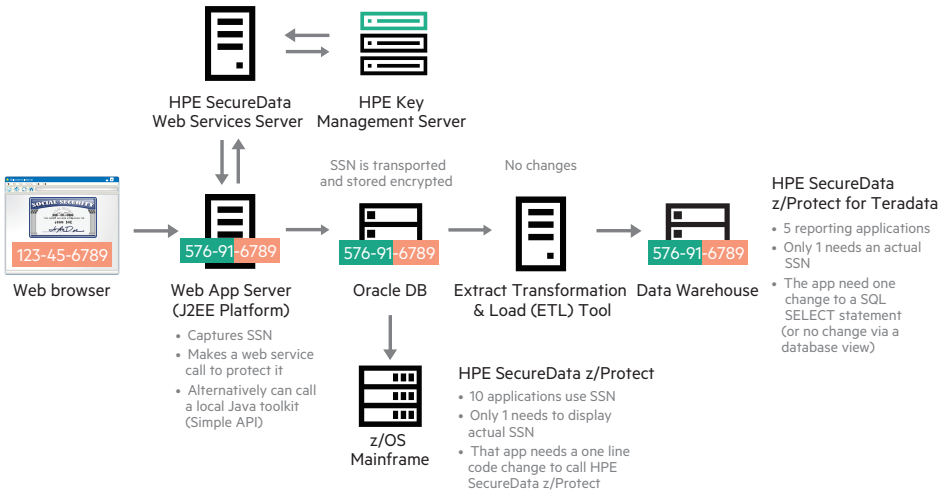


Figure 13. Sample implementation of HPE SecureData data protection solutions.

Conclusion

Compared to past approaches HPE SecureData offers distinct advantages. In addition to the security advantages of HPE FPE and HPE SST, integration efforts are reduced to hours and days, instead of months or years as in the past. De-identification of data for testing or other purposes leverages the same data protection used in production. As a true enterprise platform, clients can start with simple applications and expand the use of HPE SecureData across any number of applications and systems, from HR to financials, to custom applications to integration with CRM and Enterprise Resource Planning (ERP) systems. The same platform can be re-used for bulk unstructured data handling with HPE SecureFile and HPE SecureMail, for enterprise-wide data privacy and complete peace of mind.

The bottom line is that data protection is now feasible across the enterprise with a single approach. HPE SecureData offers huge reductions in cost and time for privacy compliance. The data-centric approach mitigates data leakage and avoids disclosure from the outset, regardless of platform choice, outsourcing needs, scaling requirements, or IT processes. For the first time, information protection and database security are simple and easy to implement, becoming a natural extension of existing infrastructure and processes.

Learn more at
voltage.com
hpe.com/software/datasecurity

Sign up for updates

★ Rate this document

Hewlett Packard Enterprise

© Copyright 2015-2016 Hewlett Packard Enterprise Development LP. The information contained herein is subject to change without notice. The only warranties for Hewlett Packard Enterprise products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. Hewlett Packard Enterprise shall not be liable for technical or editorial errors or omissions contained herein.

Intel is trademarks of Intel Corporation in the U.S. and other countries. Oracle is a registered trademark of Oracle and/or its affiliates. Linux is the registered trademark of Linus Torvalds in the U.S. and other countries. Windows is either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

4AA6-0080ENW, March 2016, Rev. 1