Streamlining Information Protection through a Data-Centric Security Approach
# Table of Contents

Overview .................................................................................................................. 1  
Why Data Needs a New Approach to Protection ....................................................... 1  
The Data-Centric Approach ...................................................................................... 3  
Demands of Data Protection in Existing Systems .................................................... 5  
Micro Focus Voltage Hyper Format-Preserving Encryption (FPE) ......................... 6  
Micro Focus Voltage Hyper Secure Stateless Tokenization (SST) ......................... 7  
Static Data Masking .................................................................................................. 8  
Past Masking Approaches and Their Challenges .................................................... 9  
Five Steps to Successful Protection of Production Data .......................................... 10  
SecureData Platform Components .......................................................................... 12  
Simple Integration—A Few Lines of Code for Trusted Applications ..................... 14  
Example: Static Data Masking to De-Identify Production Data for Testing .......... 16  
Example: Implementing Production Data Protection ............................................. 16  
Conclusion ................................................................................................................ 17
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.

This document introduces a unique approach from Micro Focus® Voltage SecureData with Hyper Format-preserving Encryption (FPE) 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 Hyper 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 SecureData Platform, allowing projects that once lasted months or years to complete in days to weeks.

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 SecureData to protect credit card data, for example, can dramatically reduce PCI DSS compliance scope and audit costs. This document covers the use of Hyper FPE and Hyper Secure Stateless Tokenization (SST) for field-level data protection, as well as both static and real-time data masking.

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.
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 1. Ideal data path for traditional protection approach.

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.

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

Micro Focus Data Security has pioneered technology that protects data independent of the subsystems that use it. 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.

SecureData provides two technologies for protecting data: Hyper FPE and Hyper 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.

![Example of Data Centric Implementation](image)

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

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

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

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

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, national i.d. 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.

Large European Telco—IoT, GDPR and the Data Lake

Business Need

■ Collect massive data sets from mobile subscribers in multiple countries for analysis and customer insights
■ Protect CDRs, location, IMEI and other personal data
■ Move data to a 180-node Hadoop cluster, complying with data residency laws and GDPR

Approach

■ Apply Hyper FPE to protect PII before Hadoop ingestion

Benefits

■ All analyses performed on data in protected form
■ Wide use of data enables ROI on tech investments
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 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.

Micro Focus Voltage Hyper Format-Preserving Encryption (FPE)

SecureData provides Hyper FPE using AES-256 encryption. Hyper FPE is based on the NIST-standard using FF1 mode of the Advanced Encryption Standard (AES) algorithm, which encrypts sensitive data while preserving its original format without sacrificing encryption strength. Like traditional AES, the Hyper 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 ciphertext.

The fact that the encrypted value has the same size and data format—including Unicode format and character set-preserving encryption for data in languages such as German, French, Spanish, Japanese, Turkish, and more—as the original enables Hyper FPE to be used with little or no changes to database schemas and applications. And inherent to how Hyper FPE works, when encrypted values are transported from mainframes to open systems, no EBCDIC to ASCII conversion is required.

Customer Payments

<table>
<thead>
<tr>
<th>CustomerID</th>
<th>First Name</th>
<th>Second Name</th>
<th>Policy#</th>
<th>SSN or Nat.ID</th>
<th>Credit Card</th>
</tr>
</thead>
<tbody>
<tr>
<td>L9234</td>
<td>John</td>
<td>Smith</td>
<td>G62-1324</td>
<td>1022-29-9234</td>
<td>4323-3273-4273-9273</td>
</tr>
<tr>
<td>Y1023</td>
<td>Borat</td>
<td>Minsavage</td>
<td>K5-1924</td>
<td>4002-3323-9283-2099</td>
<td></td>
</tr>
<tr>
<td>E9132</td>
<td>Roland</td>
<td>Johnson</td>
<td>JAW-2035</td>
<td>274-82-2729</td>
<td></td>
</tr>
<tr>
<td>H7622</td>
<td>Betty</td>
<td>Harris</td>
<td>ULY-9935</td>
<td>188-08-6254</td>
<td></td>
</tr>
<tr>
<td>H28239</td>
<td>Yoomin</td>
<td>Ng</td>
<td>AGH-9194</td>
<td>273-89-9282</td>
<td></td>
</tr>
</tbody>
</table>

Micro Focus Data Security products are designed to accelerate data privacy compliance to PCI, HIPAA, GLBA, PIPEDA, Basel II, SEC 17, SOX, SB1386 and US State and Federal regulation such as Cybersecurity Act of 2015, DFARS CUI. It is also a key component of compliance programs for the European Union’s General Data Protection Regulations (GDPR), and other international privacy regulations.

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

Health Insurance Records

<table>
<thead>
<tr>
<th>Claim Type</th>
<th>First Name</th>
<th>Second Name</th>
<th>Claim Status</th>
<th>SSN or Nat.ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Third Party Injury</td>
<td>John</td>
<td>Smith</td>
<td>Pending</td>
<td>022-29-9234</td>
</tr>
<tr>
<td>Collision Damage</td>
<td>Elizabeth</td>
<td>Montgomery</td>
<td>Approved</td>
<td>924-30-7624</td>
</tr>
<tr>
<td>No Claims on File</td>
<td>Roland</td>
<td>Johnson</td>
<td>Pending</td>
<td>074-82-2728</td>
</tr>
<tr>
<td>Collision Damage</td>
<td>Betty</td>
<td>Harris</td>
<td>Approved</td>
<td>198-09-6254</td>
</tr>
<tr>
<td>Personal Injury</td>
<td>Yoomin</td>
<td>Ng</td>
<td>Fraud Check</td>
<td>273-89-9282</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Health Insurance</th>
<th>First Name</th>
<th>Second Name</th>
<th>Risk Score</th>
<th>SSN or Nat.ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>FullBenefits</td>
<td>John</td>
<td>Smith</td>
<td>Pending</td>
<td>022-29-2034</td>
</tr>
<tr>
<td>Dental Only</td>
<td>Elizabeth</td>
<td>Montgomery</td>
<td>Approved</td>
<td>924-30-7624</td>
</tr>
<tr>
<td>FullBenefits</td>
<td>Roland</td>
<td>Johnson</td>
<td>Pending</td>
<td>874-82-2728</td>
</tr>
<tr>
<td>Family Excess 500</td>
<td>Betty</td>
<td>Harris</td>
<td>Approved</td>
<td>198-09-6254</td>
</tr>
<tr>
<td>None</td>
<td>Yoomin</td>
<td>Ng</td>
<td>Fraud Check</td>
<td>273-89-9282</td>
</tr>
</tbody>
</table>
Hyper FPE encrypts virtually unlimited data types, including IDs, VINs, bank accounts, and any classified data types that need encryption. Hyper FPE does that while preserving format, relationships, context, meaning so that analytics can be performed on de-identified data. That, combined with SecureData granular policy control, enables wide access to de-identified data, powering Big Data, cloud, and IoT initiatives while using policy management control to limit access to highly sensitive data.

Another aspect that is key for the deployment of innovations such as Big Data, cloud and Internet of Things is performance. Hyper FPE allows accelerated encryption performance aligning to the high volume needs of next-generation business initiatives. Tools for bulk encryption facilitate rapid de-identification of large amounts of sensitive data in files and databases. Typically, whole systems can be rapidly protected in just days at a significantly reduced cost.

Micro Focus Voltage Hyper Secure Stateless Tokenization (SST)

SecureData also provides tokenization. Tokenization replaces data values with a “token,” or random string of text. Hyper Secure Stateless Tokenization (SST) technology is an advanced, patented, data security solution that provides enterprises, merchants and payment processors with a new approach to help assure protection for payment card data. Hyper 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. Hyper 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 Hyper 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.

Standards matter: NIST, FIPS and Common Criteria

Micro Focus Data Security has contributed technology and core specifications for the new National Institute of Standards and Technology’s (NIST) AES FF1 Format-Preserving Encryption (FPE) mode standard.

SecureData is the world’s first FIPS and Common Criteria-validated FPE product—delivering a proven method of protecting data for global enterprises and government agencies.
In SecureData, the tokens have the same format as the original data, gaining all the advantages of FPE. Specifically, both Hyper FPE and Hyper 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 Hyper 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." Hyper 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. Hyper 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.
Hyper FPE can be configured for both reversible and non-reversible data masking.

**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

<table>
<thead>
<tr>
<th>Production CC#</th>
<th>Test CC#</th>
</tr>
</thead>
<tbody>
<tr>
<td>4391471208007120</td>
<td>X 4391471208007120</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>CC#</th>
<th>Test CC#</th>
</tr>
</thead>
<tbody>
<tr>
<td>4391471208007120</td>
<td>5555222233334444</td>
</tr>
</tbody>
</table>

**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

<table>
<thead>
<tr>
<th>Ron Smith</th>
<th>43911471208007120</th>
</tr>
</thead>
<tbody>
<tr>
<td>M. Mouse</td>
<td>4000123456780009</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Production SSN</th>
<th>Test SSN</th>
</tr>
</thead>
<tbody>
<tr>
<td>099-26-7824</td>
<td>012-34-2345</td>
</tr>
</tbody>
</table>

**Figure 7. Legacy masking approaches and their deficiencies.**

Different applications have different data needs. 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.
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, 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.

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. SecureData offers run-time masking for both FPE and tokenized data.

### Five Steps to Successful Protection of Production Data

SecureData leverages Hyper FPE and Hyper SST to provide a complete solution for data protection. With centralized management and many interfaces for performing the actual data masking, 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 Hyper FPE and Hyper SST
2. Define application identities to tie a decryption method to each application
3. Establish central administration across a distributed installation of 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 Hyper FPE and Hyper SST, plus masking when appropriate.

---

**Global Telecom Provider—Top Performance**

**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, Hadoop

**Solution:**
- SecureData for enterprise wide data protection
- Hyper FPE with embedded policy is corporate standard
- Deploying at ~15-30 applications per month; 3,000 databases

---

**USE CASE** Access to data must be separated between persons with supervisory rights (e.g. fraud investigators) and customer service reps who only have access to the last 4 digits of credit card numbers.

**Credit Card Data**

- **Frank, Fraud Investigator**
  - All digits: 5555 6666 7777 0924
- **Charlie, Customer Service Rep**
  - Last 4 digits: xxxx xxxx xx0924
  - Full field encryption as stored in database: 2323 3434 4545 1919

**Figure 8.** SecureData provides granular access to sub-fields in a database, based on the needs of the application or the identity of the user.
Next, inventory the applications that rely on 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 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 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 are 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.

---

**Use Case** | **Protection Type**
--- | ---
Ensuring PCI compliance, minimizing PCI audit scope | Hyper Secure Stateless Tokenization
Brand risk and breach mitigation, protecting PII data | Hyper Format-Preserving Encryption
Scope of PCI audit | Wide audit scope, to all application systems

---

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

SecureData Platform Components

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.

### 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 Hyper FPE, file encryption, and data masking.

### Fortune 50 Global Financial Group

**Business Drivers:**

- Compliance cost reduction, brand risk and breach mitigation, extending offshore outsourcing. Covered by nearly every privacy regulation.

**Situation:**

- Terabytes of regulated data across hundreds of applications, with complex data governance issues
- Sophisticated financial applications dependent on data
- Applications include mainframe to web-based

**Solution:**

- SecureData for enterprise wide deployment for de-identification and data protection
SecureData supports these environments:
- AIX
- CentOS Linux
- z/OS
- HP-UX
- RHEL Linux
- Stratus VOS
- Teradata
- Windows
- Hardware Security Modules
- Amazon Web Services
- Hadoop
- HPE NonStop
- Oracle
- Solaris
- SUSE Linux
- VMware
- Micro Focus Vertica

And these languages:
- C, C++
- COBOL
- .NET, ASP
- Java
- C#
- Python
- Visual Basic

Other platforms and languages supported upon request.

- **Micro Focus Voltage Key Management Server**: High-scale, on-demand, stateless key management eliminates the need for traditional complex storage-based key management because keys are dynamically derived; seamlessly integrates with existing Identity Management and Authorization Systems and Key Management using FIPS 140–2 Hardware Security Modules.

- **SecureData Web Services Server**: Tokenization option for Service Oriented Architecture environments, enterprise applications, and middleware. Supports SOAP and REST API Web services, and Unicode Latin 1 for native languages.

- **SecureData Simple API**: Maximizes efficiency on a broad range of application servers through native encryption on HP-UX, SAP HANA, Hadoop, Teradata, HPE NonStop, Microsoft® Azure, Amazon Web Services (AWS), Solaris, Stratus VOS, Linux (Red Hat, SUSE, CentOS), AIX, and Windows®. Additional APIs are available for embedded platforms such as payment terminal devices. Supports hardware accelerated encryption processes where available, e.g., Intel® AES-NI.

- **SecureData Command Lines**: Scriptable tools easily integrate bulk encryption, tokenization, and file encryption into existing batch operations and applications.

- **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.

- **SecureData Mobile**: Includes simple data security libraries to easily incorporate into native mobile applications. This enables the mobile application to secure captured data end-to-end to the trusted host using a one-time cryptographic key. Supports iOS and Android.

- **SecureData Enterprise also supports mainframe, Big Data, and payment security ecosystems**:
  - **SecureData z/Protect**: Maximizes CPU performance on mainframe systems through native z/OS support for encryption and tokenization.
  - **SecureData z/FPE**: Mainframe data processing tool to FastTrack 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.
  - **SecureData for Hadoop Developer Templates**: Enable customers to integrate Hyper FPE and Hyper SST technologies into their Hadoop instances. Templates come with pre-built integrations for Storm/Kafka, Apache (TM) NIFI, Sqoop, MapReduce and Hive, and can be quickly expanded to integrate into other technologies in the Hadoop stack such as Flume.
  - **Micro Focus Voltage SecureStorage**: Data-at-rest encryption for Linux with Stateless Key Management.
  - **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.
  - **SecureData Terminal SDK and Host SDK**: Provide market-leading P2PE payments security.

- **Micro Focus 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. SecureData servers can be distributed around the enterprise network as appropriate for scaling and for disaster recovery. Monitoring and reporting are easy: SecureData incorporates best-of-breed 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 Micro Focus Voltage SecureFile. Utilizing Micro Focus Voltage 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. SecureFile uses the same management servers as 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, 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. SecureData offers five high-level interfaces.

SecureData Web Services Server

The SecureData solution provides a web services option through the SecureData Web Services Server. This component provides a high-level encryption and tokenization API that can be accessed through a standard SOAP and REST 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 Micro Focus Data Security or integration partners. The 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 Micro Focus Data Security.

A Complete SDK

In addition to the high-level interfaces detailed here, the 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.
The example above, in Java, shows a simple call to the SecureData Web Service for a credit card example.

SecureData Command Line
The SecureData system includes a powerful multi-platform command line tool called SecureData Command Line (CL). It provides encryption and tokenization capabilities through a simple scripting interface for automated, repeatable data protection and masking. 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).

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.

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

SecureData z/Protect—For z/OS Mainframe
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.

SecureData for Teradata
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. 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 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.

```java
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 SecureData provides many privacy features.

The example above, in Java, shows a simple call to the SecureData Web Service for a credit card example.
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 Micro Focus 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 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 Key Management Server.

---

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. SecureData protects the data wherever it goes.

---

Compared to past approaches, SecureData offers distinct advantages. In addition to the security advantages of Hyper FPE and Hyper SST, integration efforts are reduced to hours and days, instead of months or years as in the past.
Conclusion

Compared to past approaches, SecureData offers distinct advantages. In addition to the security advantages of Hyper FPE and Hyper 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 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 SecureFile and 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. 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
www.microfocus.com/data-security-encryption

Figure 13. Sample implementation of SecureData data protection solutions.