Data Encryption

Encryption for data-at-rest and data-in-use

Release: Washington, DC
Introduction

ServiceNow appreciates and respects that customers entrust the Now Platform with their most valuable asset, their data.

However, not all customers are the same, and may have different use cases and compliance requirements that need to be met. For this reason, ServiceNow provides a range of encryption options.

For an overview of the ServiceNow security program, please refer to the Securing the Now Platform white paper.

Please note: all information in this white paper is related to the standard Now Platform commercial environment.

For information related to other globally located ServiceNow in-country cloud offerings and how these offerings may differ, please contact your ServiceNow account representative.
ServiceNow Shared Responsibility Model

Security is a partnership between the provider and customer, both with specific responsibilities. ServiceNow provides its customers with extensive capabilities to configure their instances to meet their own security policies and requirements and to conform with regional and industry compliance requirements.

For more information about security responsibilities with respect to customer data, please see the ServiceNow Shared Responsibility Model overview.

<table>
<thead>
<tr>
<th>Area of Responsibility</th>
<th>Responsibility</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customer</td>
<td></td>
</tr>
<tr>
<td>ServiceNow</td>
<td></td>
</tr>
<tr>
<td>Colocation (data center providers)</td>
<td></td>
</tr>
<tr>
<td>Secure configuration of instance</td>
<td>●</td>
</tr>
<tr>
<td>Authentication and authorization</td>
<td>●</td>
</tr>
<tr>
<td>Data management (classification and retention)</td>
<td>●</td>
</tr>
<tr>
<td>Data encryption at rest</td>
<td>●</td>
</tr>
<tr>
<td>Data encryption in transit</td>
<td>● ●</td>
</tr>
<tr>
<td>Encryption key management</td>
<td>● ●</td>
</tr>
<tr>
<td>Security logging and monitoring</td>
<td>● ●</td>
</tr>
<tr>
<td>Secure SDLIC processes</td>
<td>● ●</td>
</tr>
<tr>
<td>Penetration testing</td>
<td>● ●</td>
</tr>
<tr>
<td>Vulnerability management</td>
<td>● ●</td>
</tr>
<tr>
<td>Privacy</td>
<td>● ●</td>
</tr>
<tr>
<td>Compliance: regulatory and legal</td>
<td>● ● ●</td>
</tr>
<tr>
<td>Employee vetting or screening</td>
<td>● ●</td>
</tr>
<tr>
<td>Physical security/environment controls</td>
<td>● ●</td>
</tr>
<tr>
<td>Cloud infrastructure security management</td>
<td>● ●</td>
</tr>
<tr>
<td>Infrastructure management</td>
<td>● ●</td>
</tr>
<tr>
<td>Media disposal and destruction</td>
<td>● ●</td>
</tr>
<tr>
<td>Backup and restore</td>
<td>● ●</td>
</tr>
<tr>
<td>Business continuity and disaster recovery</td>
<td>● ●</td>
</tr>
</tbody>
</table>
Default protection — standard encryption in the Now Platform

Column Level Encryption (CLE)
Standard Column Level Encryption (CLE) is built into the Now Platform, without further licenses. CLE permits encryption of string, date, date/time, and attachment fields supporting AES-128/AES-256.

CLE is included with the Now Platform, but it has limited functionality when compared with Column Level Encryption Enterprise (CLEE). See the comparison chart for the full list of differences in functionality here.

For information on CLE key management see CLE/CLEE key hierarchy.

Key Management Framework
ServiceNow key management is based on National Institute of Standards and Technology (NIST) 800-57 guidelines.

The ServiceNow Key Management Framework (KMF) allows customers to fully customize and manage how cryptographic operations are performed on their Now Platform instance. The Now Platform KMF provides customers with the essential cryptographic tools to enable data security through confidentiality, integrity, and authentication.

KMF is available out of the box on the Now Platform and supports encryption in the creation and management of cryptographic modules specific to each type of encryption.

At its core, KMF provides an interface for the following:

- Key lifecycle management to generate, rotate, revoke, and suspend cryptographic keys, including support for several key lifecycle states.
- KMF lifecycle state automation provides enforcement of specific allowable actions (e.g., automated deactivation or automated rotation)
- Configuration of the managed cryptographic keys to a specific cryptographic usage and algorithm (e.g., AES-GCM with 256-bit key for data encryption purposes).
- Enforcement of access controls for the managed cryptographic keys (i.e., Module Access Policy) ensuring that access is only granted to the cryptographic modules as configured by the customer admin in the Now Platform policies.
Platform Encryption subscription bundle

The Platform Encryption bundle comprises two encryption products that work together to meet most customers’ typical needs:

1. **Cloud Encryption (CE)** encrypts all the data within the database at the storage volume.
2. **Column Level Encryption Enterprise (CLEE)** can then be used to encrypt selected fields or attachments within the application to ensure only authorized employees can access specific sensitive data.

**Cloud Encryption**

Cloud Encryption provides block encryption of the storage volume that contains each specific Now Instance database with industry-standard, customer-controlled, key lifecycle management built into the Now Platform user interface.

Cloud Encryption encrypts data at rest in the database storage volume using symmetric AES 256-bit encryption with minimal impact to functionality. Database related activity log data (e.g., bin, redu, undo, and error) is stored within the encrypted volume.

When Cloud Encryption is enabled, cloning within the vertical instance hierarchy (production, qa, test, dev, etc.) is fully supported. Encryption is enabled/maintained based on the source instance’s current configuration (i.e., if you clone an encrypted prod-source to an unencrypted sub prod-target, the new clone sub prod will be encrypted).

Cloud Encryption gives customers the option to use ServiceNow Managed Keys (SMK) or a key created and supplied by the customer, BYOK (Bring Your Own Key).

Key rotation operations are completely managed by customer admins from within their Now Platform instance, providing flexibility and autonomy, as well as avoiding the need to involve ServiceNow customer support.
Cloud Encryption key hierarchy

Cloud Encryption is performed using a multi-layer approach. The External key (Level 3) is the key being managed by operations provided with Now Platform Cloud Encryption.

The External Key is stored within the ServiceNow Key Management System (KMS). The root key is embedded in a FIPS-140-2 L3 certified Hardware Security Module (HSM). The Service Key and Master Key are stored in a secure way on the instance itself.

The keys are stored and managed by ServiceNow as follows:

- **Master key** - Level 1
  The Master key is a component of LUKS which serves as the data encryption key. This consists of 512 bit length AES 256 bit XTS key.

- **Service key** - Level 2
  The Service key is an AES 256 bit key that provides protections for access to the Master key and is wrapped by the External key.

- **External key** - Level 3
  The External key is an AES 256 bit key stored in the FIPS 140-2 validated Key Management Systems within the same ServiceNow Data Centers as the customer’s instance it is associated with. This key is either a SMK (ServiceNow Managed Key) or a CMK (Customer Managed Key) provided by the customer using the BYOK (Bring Your Own Key) option.
Column Level Encryption Enterprise (CLEE)

Column Level Encryption Enterprise (CLEE) provides field-level and attachment-based data encryption within instances of the Now Platform. CLEE is only available as part of the Platform Encryption Bundle.

CLEE uses the KMF encryption modules, granting customers more control of server-side encryption. Customer admins can configure which specific data to encrypt, within a specific table, thereby storing the data in encrypted form.

Encryption keys are stored and maintained within the Now Platform instance and managed through the Key Management Framework (KMF).

The main features of CLEE are:

- Encryption of supported field types like string text, date/time fields, attachments, and URLs
- Employs AES-CBC (Cipher Block Chaining) or AES-GCM (Galois Counter/Mode) with 256-bit keys
- Offers both deterministic and non-deterministic encryption options
- Allows a user with appropriate access to perform limited searching and filtering operations on data that has been encrypted
- CLEE can be used on file attachments
- Enables customers to supply their own encryption keys BYOK (Bring Your Own Key) or have keys randomly generated on the Now Platform
- Offers several access controls based on role assignment and application scope

Mitigating the risk of exposing sensitive data as either the result of a direct attack or of compromised data stored in the cloud

Enabling customers to comply with governmental and industry certification requirements and regulations

Limiting access to sensitive data based on defined roles, defined script assignments, application scope, and domain membership
**CLE/CLEE key hierarchy**

Column Level Encryption (CLE) and Column Level Encryption Enterprise (CLEE) Module Keys (also known as Customer Data Encryption Key [CDEK]) use an application layer data encryption key, which is the actual key performing the encryption of specific columns within tables.

To protect the application layer data encryption key, a key chain is used, protected by a root key stored within a tamper-resistant Hardware Security Module (HSM) that is FIPS 140-2 L3 certified.

The root key envelopes the Instance Root Key (IRK). This key is customer instance specific and stored securely within the Key Management System.

The IRK envelopes the Instance Key Encryption Key (IKEK), which is stored in a secure way on the instance. The IKEK envelopes the module key (CDEK) which is also stored on the customer instance itself.

The key’s stored on the instance are protected by the key’s stored in the KMS and HSM, this approach is called “hardware root of trust.”

To provide availability of the keys there are HSM’s in each data center pair within a data center region.

Encryption keys provided by customers for use with Column Level Encryption Enterprise (CLEE) are backed up within the database for the customer instance where they are used. Customers should also back up encryption keys prior to applying them to their instances.

---

**Diagram:**

- **Root Key (RK)**
- **Instance Root Key (IRK)**
- **Instance Key Encryption Key (IKEK)**
- **Customer Data Encryption Key (CDEK)**

---

*Note: CLEE does not enable customers to store encryption keys in their own Hardware Security Modules (HSM), key storage appliances, or services.*
Example of CLE/CLEE separation of duties

1. The customer’s crypto manager uses Key Management Framework (KMF) crypto modules and cryptographically signed Module Access Policies (MAPs) to configure all instance cryptography use cases. For example, encryption of supported tables and field types, and defining key access to privileged roles, groups, scripts, and system.

   All keys are hierarchically wrapped and rooted in FIPS 140-2-L3 HSMs.

2. The ServiceNow Security Engineering Team (SET) manages the key management devices. SET can see key metadata in order to perform administrative duties for proper functioning of the HSMs but cannot access actual key material.

   Logs are collected and retained to detect potentially malicious actions in conjunction with its ongoing operational security and incident management.

   SET is intentionally isolated from the application and database engineering teams.

3. The customer uses the ServiceNow Access Control plugin (SNC) to restrict instance-level access. Customers can specify who from ServiceNow can access the instance, and when.

4. ServiceNow application and database engineering teams ensure availability and performance of Now Platform instances.

   Engineering teams do not have access to key material, and cannot decrypt customer data.
**CLEE cryptographic module**

CLEE encryption keys are managed via the Key Management Framework (KMF), specifically through CLEE cryptographic modules, which are created by users assigned with the KMF cryptographic manager role. Once a CLEE cryptographic module is created, it can be associated to a field within a given table, thus enabling CLEE for the given field.

Whether generated by the ServiceNow instance or customer supplied (BYOK), the keys are stored in the same unique customer instance database where the data encrypted by them is stored.

As part of the KMF, the encryption keys themselves are stored in encrypted form and are encrypted by the Instance Key-Encryption Key (IKEK), an instance-unique key generated by KeySecure. This mitigates direct access to the encryption key, either by an instance administrator or ServiceNow.

**CLEE access control**

Within the CLEE cryptographic module, KMF cryptographic managers can grant access to the module based the criteria below. These access controls are not mutually exclusive; multiple access controls can be configured for a CLEE cryptographic module to provide flexibility.

1. **Role**: access is based on the role for the user session.
2. **Application Scope**: access is based on being in the targeted application scope.
**CLEE access control example 1**
The access control example illustrates how single access control is implemented (role-based).

<table>
<thead>
<tr>
<th>User 1</th>
<th>Role 1</th>
<th>User 2 &amp; User 3</th>
<th>Group 1 &amp; Role 1</th>
<th>User 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 1 is a member of Role 1</td>
<td>User 2 and User 3 are members of Group 1</td>
<td>User 4 is not a member of any group or role and has no access to CLEE cryptographic module 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Role 1 provides access to CLEE cryptographic module 1</td>
<td>Group 1 is a member of Role 1</td>
<td>User 4 does not have access to Field A or Field B</td>
<td></td>
<td></td>
</tr>
<tr>
<td>This allows User 1 to see the contents of Field A and Field B</td>
<td>This group membership allows everyone in Group 1 access to cryptographic module 1</td>
<td>Users 2 and 3 are members of Group 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Group 1 membership also allows User 2 and User 3 to see the contents of Field A and Field B</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

![Diagram of access control example](image)
CLEE access control example 2
This access control example illustrates how single access control is implemented (role-based).

<table>
<thead>
<tr>
<th>User 1</th>
<th>Role 1</th>
<th>Domain 1</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 1 is a member of Role 1 and is currently in Domain 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Access to CLEE Cryptographic Module 1 is granted due to an access control that allows access to Role 1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Thus, User 1 can see the contents of Field A and Field B</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

This demonstrates access based on role (like the previous example).

<table>
<thead>
<tr>
<th>User 2</th>
<th>Role 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 2 is member of Role 2 (i.e., not a member of Role 1) and is currently in Domain 2</td>
<td></td>
</tr>
<tr>
<td>Access is allowed since an access control for Domain 2 exists for CLEE cryptographic module 1</td>
<td></td>
</tr>
<tr>
<td>As a result, User 2 can see the contents of Field A and Field B</td>
<td></td>
</tr>
</tbody>
</table>

This demonstrates access based on domain membership.

<table>
<thead>
<tr>
<th>User 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>User 3 is neither a member of Role 1 nor in Domain 2</td>
</tr>
<tr>
<td>This means that User 3 cannot see the contents of Field A and Field B</td>
</tr>
<tr>
<td>Furthermore, User 3 will not see that these fields exist</td>
</tr>
</tbody>
</table>
CLEE usage and restrictions

Column Level Encryption Enterprise (CLEE) can be used to process specific sensitive data sets in the ServiceNow environment. The data is only decrypted by a user/script with authorized access to the associated CLEE cryptographic module.

Controlling access to sensitive data often means limiting access in a controlled fashion or granting it on an as-needed basis. ServiceNow accomplishes this through a just-in-time process where an ephemeral java object displays a decrypted copy of the encrypted value only for as long as the user or process with the correct access level acts on the data (e.g., viewing within the UI or using the value in an automation step). Once the user or process finishes, the ephemeral copy is then flushed in accordance with the platform’s normal garbage collection features.

CLEE-encrypted data is maintained throughout the backup process.

Conclusion

In summary, ServiceNow offers a robust security program with various encryption options to cater to diverse customer needs and compliance requirements. This white paper has explored two key offerings:

- **Default Protection and Key Management Framework (KMF):** This provides a foundation for data security with built-in encryption and granular key management tools.
- **Platform Encryption Bundle:** This subscription service offers more advanced encryption capabilities, including Cloud Encryption for storage volume encryption and Column Level Encryption Enterprise (CLEE) for encrypting specific data fields.

By leveraging these options and adhering to the ServiceNow shared responsibility model, customers can achieve a high level of data security within the Now Platform.

Resources

- Encryption Product Documentation
- Shared Responsibility Model overview
- Column Level Encryption
- Platform encryption
- Key Management Framework (KMF)
- IP range based authentication
- ServiceNow Data Processing Addendum