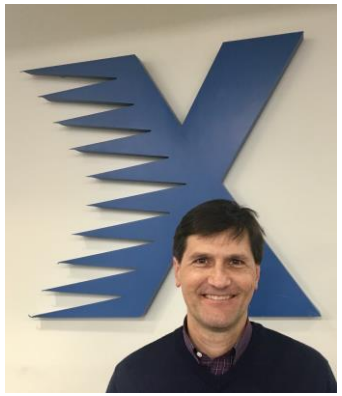
New Remote Encryption Key Storage in Informix Database Server 14.10

Tuesday , January 21st, 2020
10:00 AM Eastern Standard Time



JC Lengyel
Informix Kernel Team Lead
HCL Software

IBM Cloud

Disclaimers

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM's sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract.

The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

Background: Encryption-at-Rest in IDS

- All used pages in an encrypted space are fully encrypted on disk.
- In-flight data unaffected: pages in the buffer pool are unencrypted.
- Self-contained: no encryption dependency between replication nodes.
- All space types may be encrypted.
- Two ways to encrypt an existing, unecrypted instance:
  1. Full physical restore from an archive with encryption enabled
  2. Enable encryption, then perform warm restores on desired spaces
- What does an encrypted IDS instance look like? (Demo)

Background: Encryption-at-Rest in IDS

- Internal benchmarks and anecdotal evidence from the field indicate that performance degradation due to EAR is negligible, especially if your machine is not CPU-bound.
- All encryption/decryption is performed by the AIO and KAIO threads
- IDS Master encryption key (IMEK) is retrieved only once, during server startup.
- Each space is encrypted using a different key, derived from the IMEK.

Background: Encryption-at-Rest in IDS

- EAR is enabled using the **DISK_ENCRYPTION** configuration parameter

```
DISK_ENCRYPTION keystore=<keystore name>,cipher=<aes128|aes192|aes256>
```

- Keystore attribute can point to:
  - A local file that contains the IMEK
  - A credentials file
- Cipher is instance-wide. All encrypted spaces use the same cipher established when encryption is enabled.

Creating and storing the IMEK locally

- Before EAR is enabled, the onkstore utility is required to create a local keystore file.
- An IMEK is generated and stored in a PKCS12 GSKit Store (GSKS) file.
- The GSKS file is encrypted using a password that can be optionally stashed for convenience.
- (Demo)

Creating and Storing the IMEK Remotely

- Before EAR is enabled, the onkstore utility is required to create a credentials file.
- Credentials authorizing access to a remote IMEK are provided to onkstore. Those credentials are verified and then stored in a GSKS file.
- In some cases onkstore will generate the IMEK if it does not exist.
- Supported remote key servers (RKS):
  - Amazon Web Services Key Management (AWS)
  - Microsoft Azure Key Vault (Azure)
  - Key Management Interoperability Protocol (KMIP) compliant servers
- Credentials supplied to onkstore are RKS-dependent

AWS Credentials

- Key ID
  - Akin to account user name
- Key Secret
  - Akin to account password (don't lose!)
- Region
  - Keys cannot be shared across AWS regions
- CMK ID
  - A Customer Master Key must be generated through the AWS account.
- SSM Key Location
  - More a label for your IMEK than an actual location

AWS Credentials (cont)

- Credentials can be supplied to onkstore individually via interactive prompts, or en masse via a JSON document.
- (Demo)

Azure Credentials

- Vault URL
  - Address of your key vault
- Client ID
  - Web application ID
- Client Secret
  - Web application secret (don't lose!)
- Directory ID
  - ID of the directory under which your key vault was created
- Key Name
  - Name (or full ID) of the Remote Master Encryption Key, the Azure equivalent of the AWS CMK

Azure Credentials (cont)

- Encrypt Algorithm
  - The algorithm you chose when you created your RMEK in Azure Key Vault
- Secret Name
  - A label for the IMEK that onkstore will generate/search for

- Credentials can be supplied to onkstore individually via interactive prompts, or en mass via a JSON document.
- (Demo)

KMIP Credentials

- Server
  - IP address or hostname + optional port number
- Username
  - Optional in most cases
- Password
  - Optional in most cases
- Client certificate file
  - Path to a Privacy-Enhanced Mail (PEM) file containing certificate + private key matching the certificate
- CA certificate file
  - Path to PEM file containing root certificate used to sign both the client certificate and the KMIP server certificate file

KMIP Credentials (cont)

- Key name
  - Optional – if blank, onkstore will generate a new IMEK

- Credentials can be supplied to onkstore individually via interactive prompts, or en mass via a JSON document.
- (Demo)

Integrated Backup Encryption (IBE)

- Archives and log backups can now be natively encrypted and decrypted by IDS.
- Supported by both ontape and onbar.
- Designed to work mainly with remote key servers.
- A new Backup Encryption Key (BEK) is automatically generated for each backup, and is actually stored, encrypted, *in* the backup.
- The BEK present in the backup cannot be decrypted and used during a restore without access to the CMK (AWS) or RMEK (Azure).
- To use IBE with a KMIP compliant server, it must support the ENCRYPT and DECRYPT cryptographic operations.

BAR_ENCRYPTION

- Similar format to **DISK_ENCRYPTION**:

```
BAR_ENCRYPTION keystore=<keystore name>,cipher=<aes128|aes192|aes256>
```

- (Demo)

So Many Abbreviations and Key Types

- EAR – Encryption-at-Rest
- MEK – Master Encryption Key (Generic term)
- IMEK – IDS MEK.
  - The key from which all DBspace keys will be generated for use with EAR.
- CMK – Customer Master Key
  - Created within an AWS account. Used to encrypt any IMEKs stored there.
- RMEK – Remote MEK
  - Created within an Azure account. Used to encrypt any IMEKs stored there.
- BEK – Backup Encryption Key
  - The key IDS uses to encrypt archives and log backups. Internally-generated for each backup, encrypted, and stored in the backup itself.

Questions