Tomcat Datasource configuration and data encrypted with AKV

Last updated by | Soma Jagadeesh | Jan 10, 2021 at 9:46 PM PST

Contents

Issue

Customer successfully tested the ‘Always Encrypted’ feature with AZURE_KEY_VAULT key via manually connect and submit queries to Azure SQL DB.

Investigation/Analysis

However, in customer application environment, the data source is obtained by JPA from the Tomcat connection pool, which does not have the AKV provider registered. Therefore, the application failed with error: Failed to decrypt a column encryption key. Invalid key store provider name: AZURE_KEY_VAULT. A key store provider name must denote either a system key store provider or a registered custom key store provider.

Customer requested an example of configuring the SQLServerDataSource in JPA via Tomcat configuration file with the column encryption enabled.

Mitigation

https://github.com/cheenamalhotra/spring-jndi-mssqljdbc

RCA (optional)

More Information (optional)

Detailed information/background that may be useful but isn’t strictly required for troubleshooting. Often this is the “verbose” details that one usually doesn’t need

Public Doc Reference (optional)

https://docs.microsoft.com/en-us/sql/connect/jdbc/using-always-encrypted-with-the-jdbc-driver?view=sql-server-2017

Internal Reference (optional)

Classification

Root cause tree: Security/User issue/error/Uncoded

How good have you found this content?

🙂 🙁