**Security Performance Plan**

Though the database has gone missing or has been inaccessible in the NetLab environment for currently unknown reasons, we were able to implement some of this locally. We tested locally before implementing in the database in NetLab.

**Role-Based Access Control**

* We set up roles and planned to integrate with Active Directory to grant only the necessary permissions.
* We granted full rights only to DBAs, read-only rights to Data Analysts, and restricted rights to application users.

**User Accounts**

* We made sure to ensure strong, secure passwords.
* We explicitly defined which accounts could log into the SQL Server instance (though we never got that far as the NetLab environment held us back)

**Data Encryption**

* Given this is a class project and the delays in continued access to the database in the virtual environment, we haven’t yet implemented Transparent Data Encryption (TDE) when handling sensitive customer information and ensure that data in transit is protected (via SSL/TLS) between clients and the server.

**Auditing & Monitoring**

* If we gain access and can get everyone else access to the database in the NetLab environment, we want to use SQL Server’s auditing or logging features to monitor changes, access, and potential security breaches, as well as track failed login attempts and regularly review security logs.

**Constraint Enforcement**

* The constraints defined in the scripts (e.g., NOT NULL, CHECK constraints) are there help prevent bad data entry, which supports overall data integrity. As you can see by our create table script, we made sure to ensure proper constraints on each table.

**Indexing & Query Optimization**

* We identified key columns (like Email in the Customer table, Order\_Date in the Order table, and frequently joined foreign keys) and create appropriate indexes.
* We also created views on a few tables to simplify and accelerate queries.
* We only have a week or so left in this class and I’m not sure if we’ll be able to do this, but if we’re able to we will regularly monitor query execution plans and adjust indexes and/or queries to resolve any performance bottlenecks.

**Scalability Planning**

* Though not implemented, we discussed potential future growth by outlining procedures for scaling the database through methods like partitioning or read replicas if transaction volume were to increase significantly.