Modern Data Protection Strategies with SQL Server

John Sterrett – CEO of ProcureSQL

#FABCONSQLCON2026

# FABCON | SQLCON

Microsoft Fabric COMMUNITY CONFERENCE | Microsoft SQL COMMUNITY CONFERENCE

ATLANTA MARCH 16 - 20, 2026

Sound off.
The mic is all yours.
Influence the product roadmap.

Join the Fabric User Panel

Share your feedback directly with our Fabric product group and researchers.

https://aka.ms/JoinFabricUserPanel

Join the SQL User Panel

Influence our SQL roadmap and ensure it meets your real-life needs

https://aka.ms/JoinSQLUserPanel

About John Sterrett

john@procuresql.com

ProcureSQL

procuresql.com

Microsoft CERTIFIED TRAINER MCT

linkedin.com/in/johnsterrett

MVP Microsoft Most Valuable Professional Reconnect

redgate COMMUNITY AMBASSADOR 2025

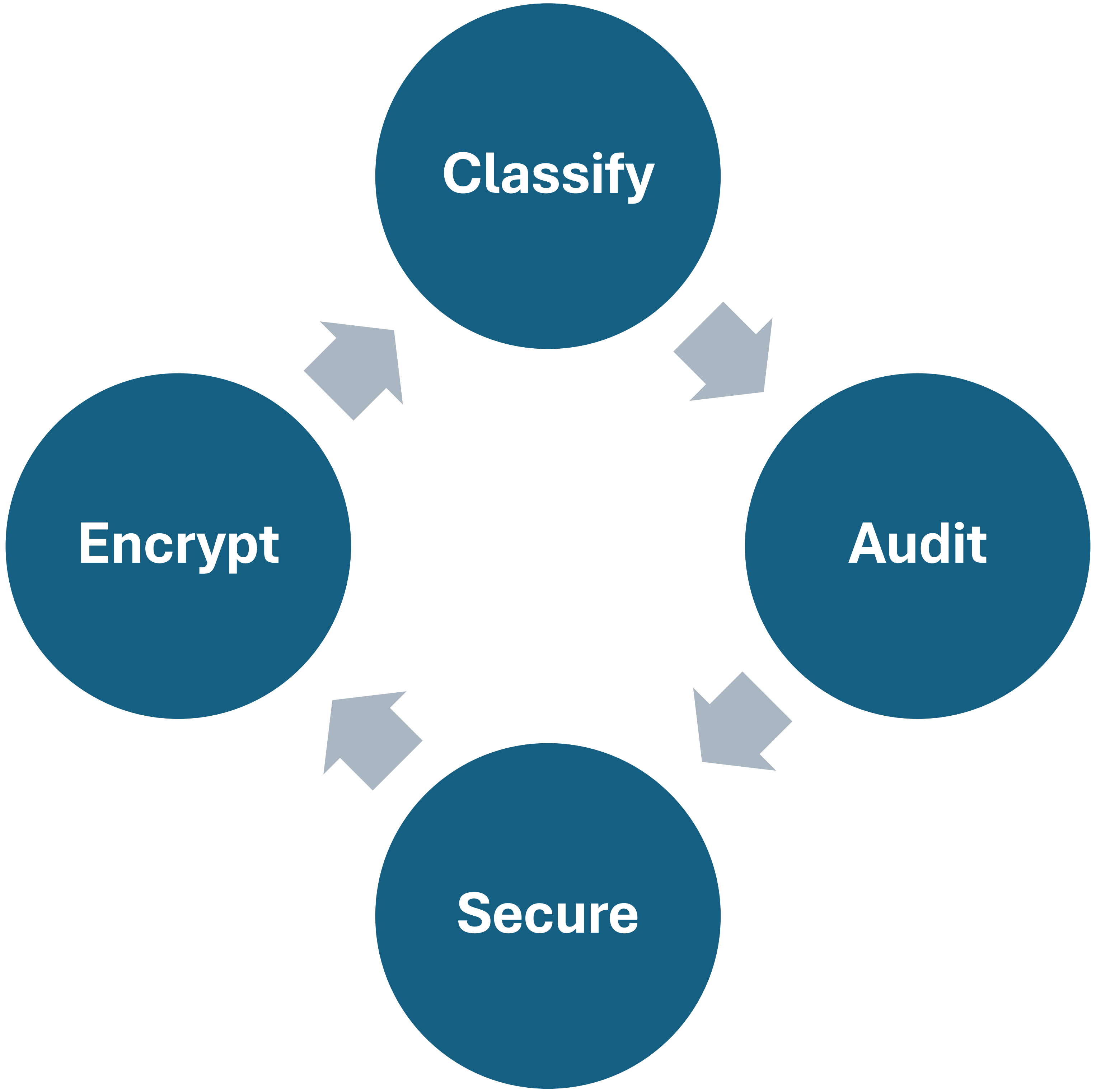
Goal of Today's Session

Learn how to secure your data with Auditing, Row-Level Security, and Always Encrypted

Gain insight into pros and cons to utilizing advanced data protection features.

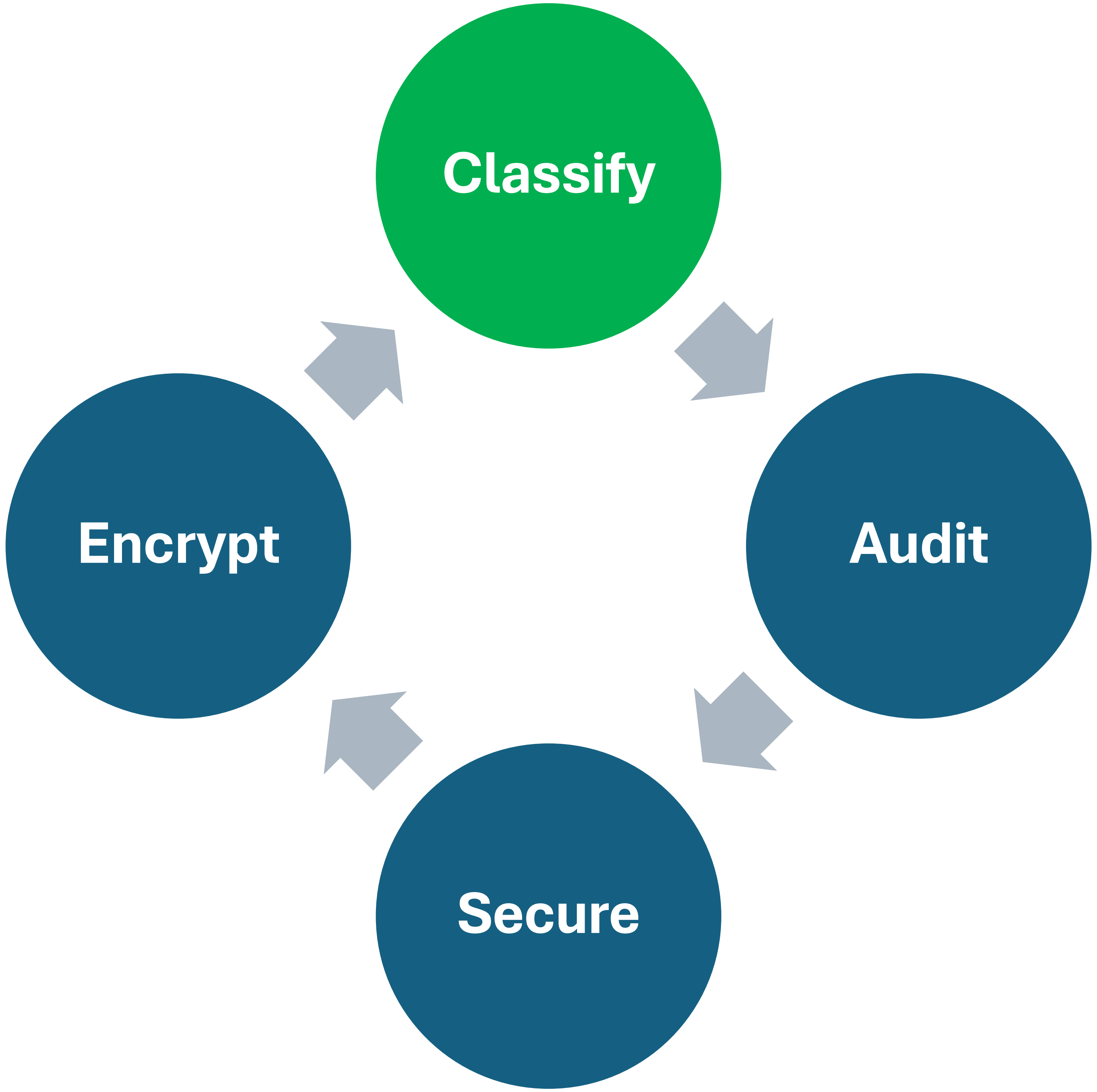
Leverage built in cloud features provided by Azure SQL for advanced data security.

Data Security Lifecycle

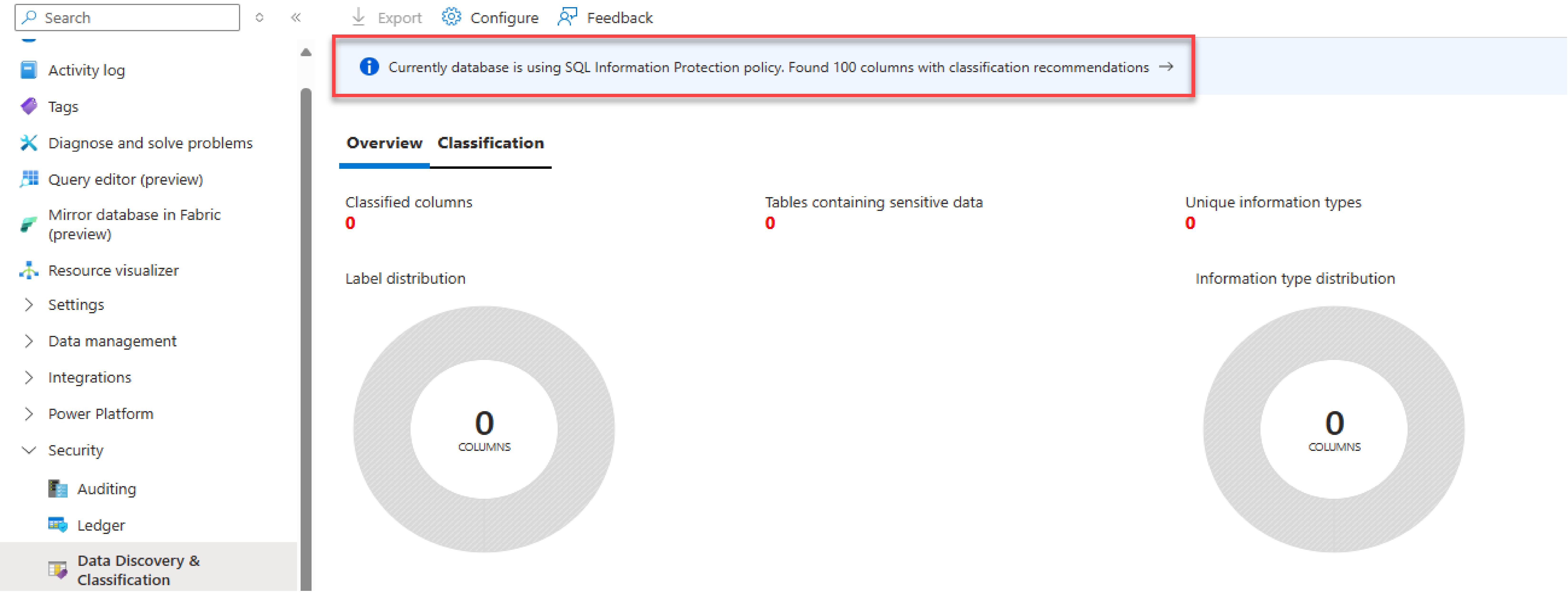


Classify Your Sensitive Data

You must **identify and classify** your sensitive data to know which **data must be protected.**



Discover and Classify Sensitive Data



Currently database is using SQL Information Protection policy. Found 100 columns with classification recommendations →

Classified columns: 0

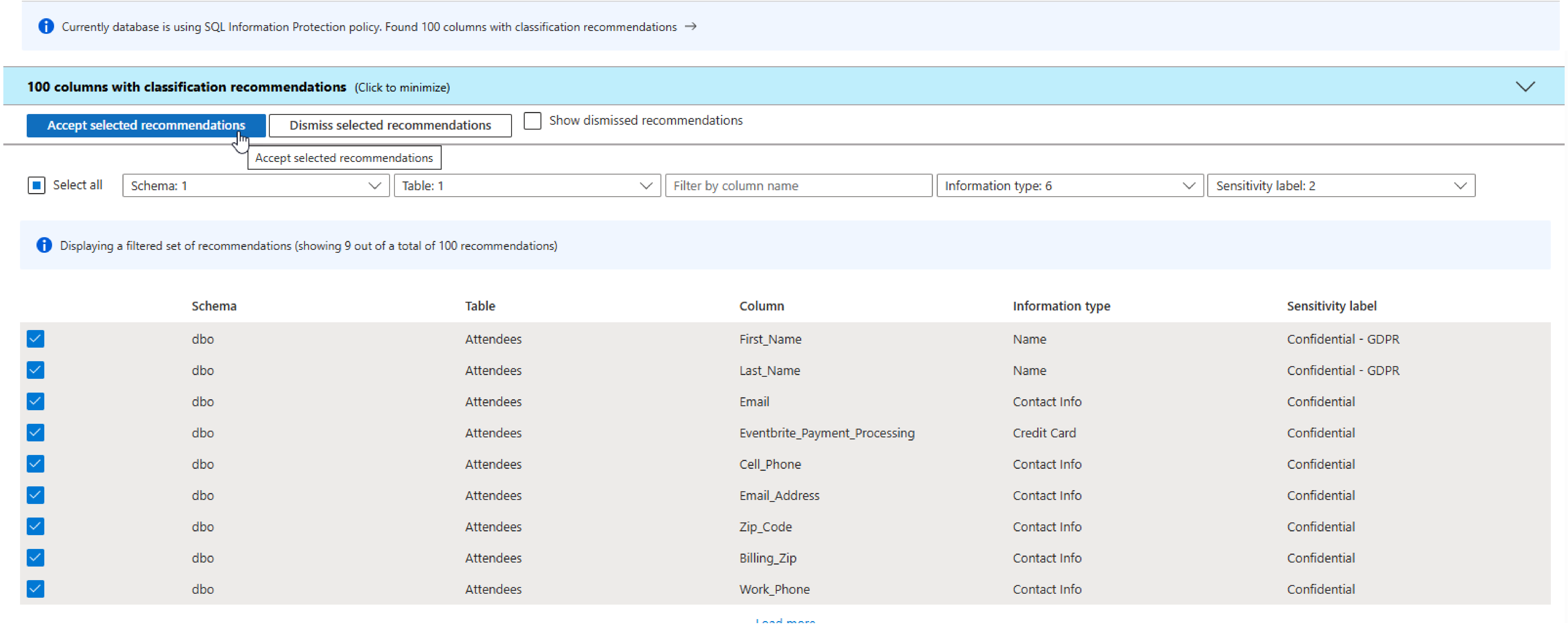
Tables containing sensitive data: 0

Unique information types: 0

Label distribution: 0 COLUMNS

Information type distribution: 0 COLUMNS

Adding Classified Columns



Currently database is using SQL Information Protection policy. Found 100 columns with classification recommendations →

100 columns with classification recommendations (Click to minimize)

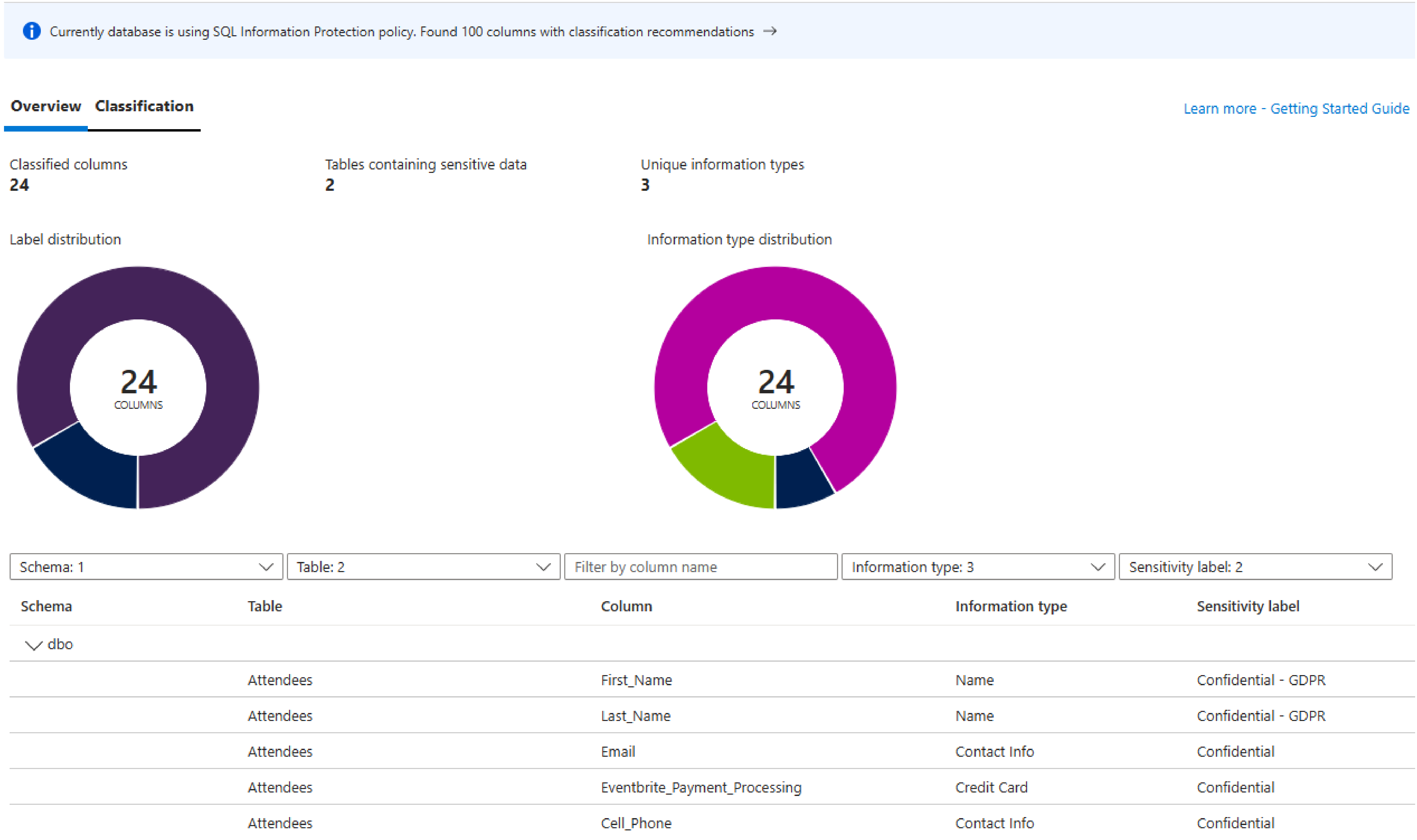
Accept selected recommendations
Dismiss selected recommendations
□ Show dismissed recommendations
Accept selected recommendations

☑ Select all | Schema: 1 | Table: 1 | Filter by column name | Information type: 6 | Sensitivity label: 2

Displaying a filtered set of recommendations (showing 9 out of a total of 100 recommendations)

	Schema	Table	Column	Information type	Sensitivity label
☑	dbo	Attendees	First_Name	Name	Confidential - GDPR
☑	dbo	Attendees	Last_Name	Name	Confidential - GDPR
☑	dbo	Attendees	Email	Contact Info	Confidential
☑	dbo	Attendees	Eventbrite_Payment_Processing	Credit Card	Confidential
☑	dbo	Attendees	Cell_Phone	Contact Info	Confidential
☑	dbo	Attendees	Email_Address	Contact Info	Confidential
☑	dbo	Attendees	Zip_Code	Contact Info	Confidential
☑	dbo	Attendees	Billing_Zip	Contact Info	Confidential
☑	dbo	Attendees	Work_Phone	Contact Info	Confidential

Data Classification after Saving Recommendations



Currently database is using SQL Information Protection policy. Found 100 columns with classification recommendations →

Overview **Classification**

Learn more - Getting Started Guide

Classified columns
24

Tables containing sensitive data
2

Unique information types
3

Label distribution

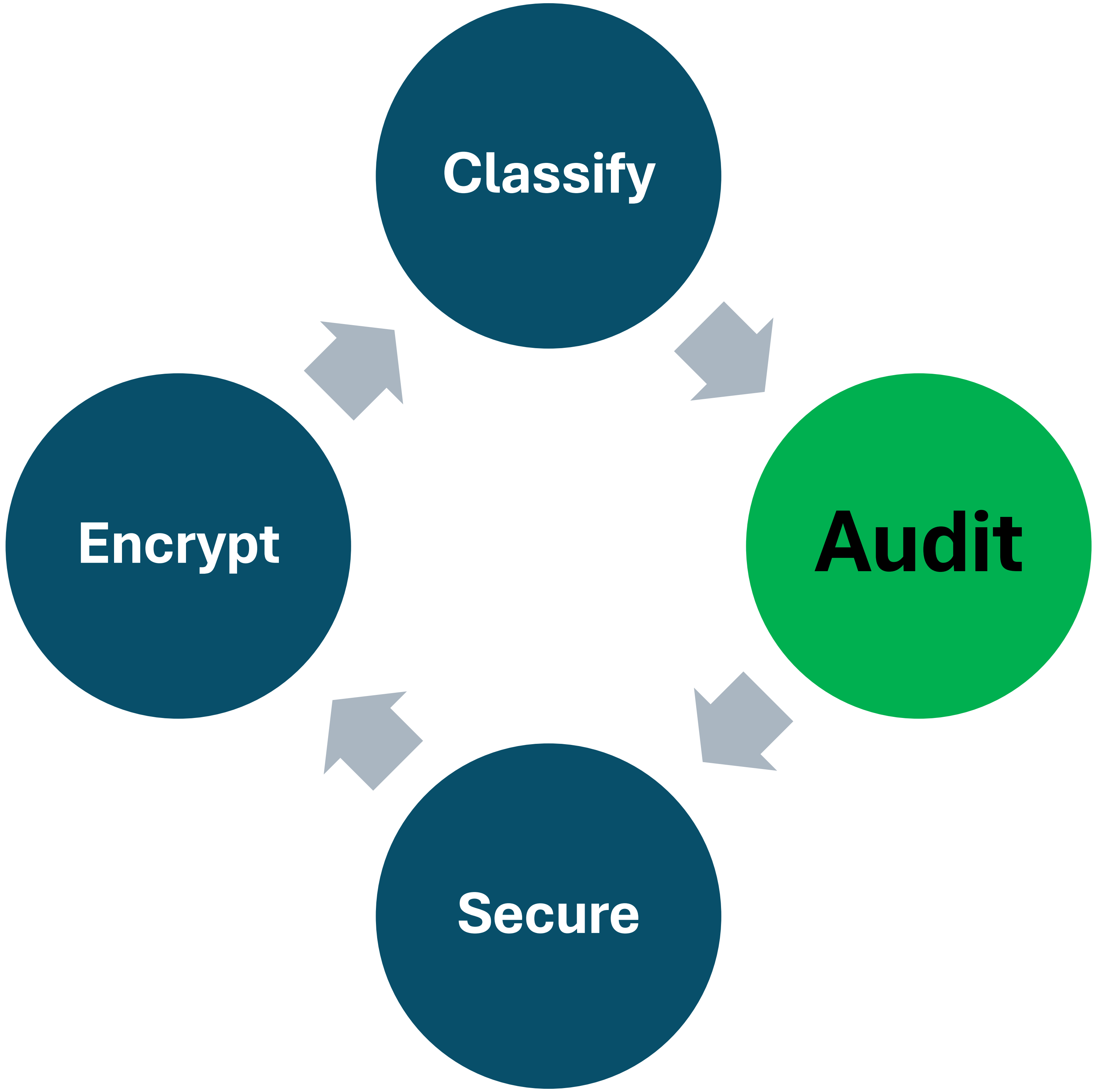
Information type distribution

Schema: 1 | Table: 2 | Filter by column name | Information type: 3 | Sensitivity label: 2

Schema	Table	Column	Information type	Sensitivity label
dbo				
	Attendees	First_Name	Name	Confidential - GDPR
	Attendees	Last_Name	Name	Confidential - GDPR
	Attendees	Email	Contact Info	Confidential
	Attendees	Eventbrite_Payment_Processing	Credit Card	Confidential
	Attendees	Cell_Phone	Contact Info	Confidential

SQL Audit

Identify the Who, What, When, Where behind data access, security and schema changes



What do these compliances have in common?

- ✓ ISO 27001
- SOX (Sarbanes-Oxley Act)
- GDPR
- PCI DSS (Payment Card Industry Data Security Standard
- HIPAA (Health Insurance Portability and Accountability Act

SQL Audit Enables You To....

Track login attempts, failed logins, which can indicate attempts of malicious attempts

Changes to database schema, permissions, and security configurations

Record Access to both data and object changes in real-time

SQL Audit Decisions

- What do we Audit?
- Where do we store audit data?
- What should happen if auditing fails?
- How do we secure the Audit logs?
- How do we organize audits?

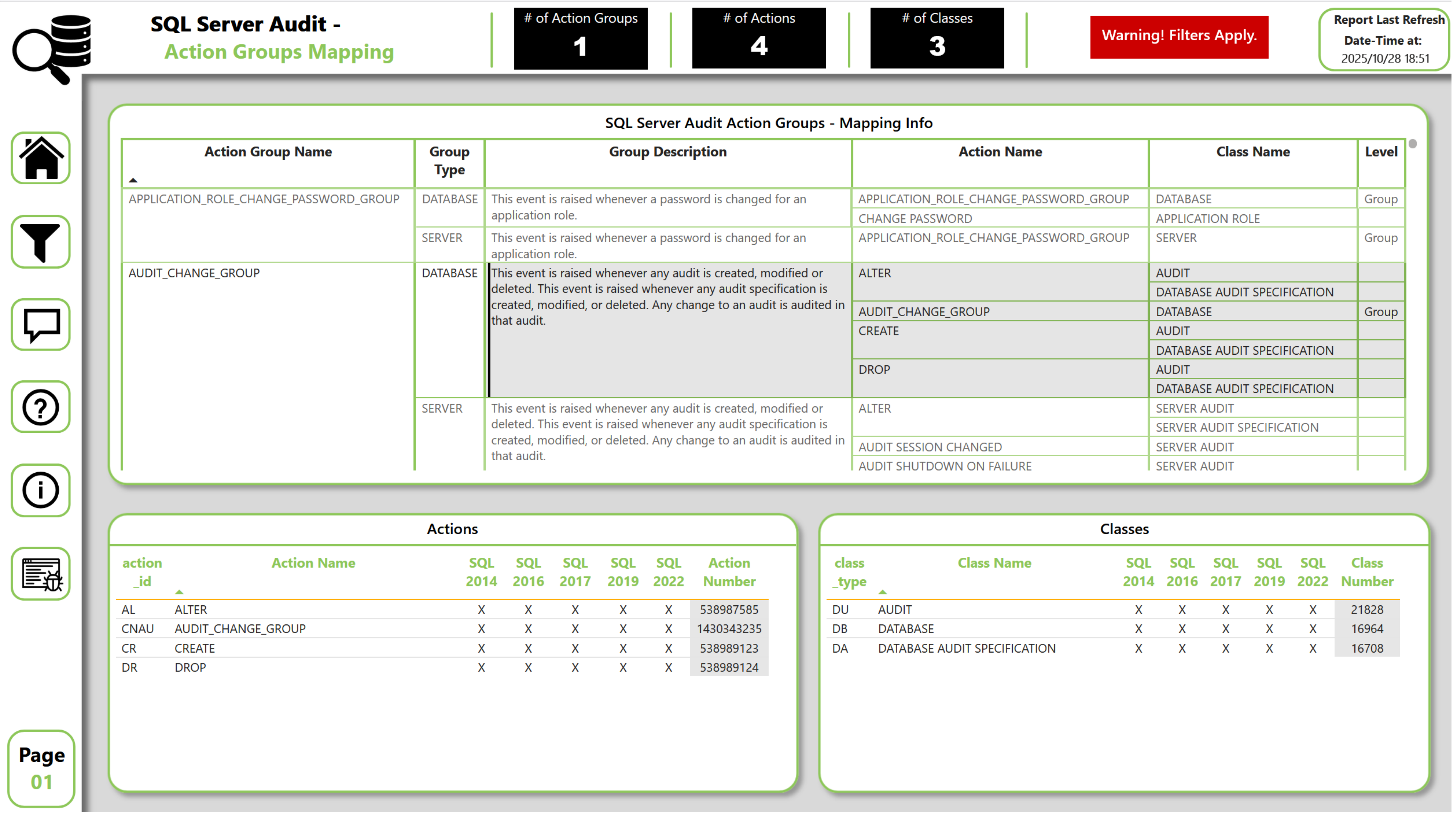
What to Audit for HIPAA, PCI, GDRP?

Compliance	Common SQL Audit Actions
HIPAA	SELECT, INSERT, UPDATE, DELETE (data access/modification) on PHI tables; login success/failure; permission changes;
PCI DSS	User account creation/modification/deletion; login attempts; permission changes; data access/modification
GDPR	Logon activity; unauthorized access attempts; data access and processing

Audit Action Groups for HIPAA, GDPR, PCI

Audit Action Group	Description
SUCCESSFUL_LOGIN_GROUP	Captures all successful login attempts. Important for tracking authorized access (HIPAA, GDPR, PCI).
FAILED_LOGIN_GROUP	Captures failed login attempts, helping detect unauthorized access or brute force attacks (HIPAA, GDPR, PCI).
SERVER_PRINCIPAL_CHANGE_GROUP	Tracks creation, alteration, or deletion of server-level principals (logins, users). Critical for permission management (HIPAA, PCI).
SERVER_ROLE_MEMBER_CHANGE_GROUP	Monitors changes to server role memberships, ensuring role-based access control integrity (HIPAA, PCI).
AUDIT_CHANGE_GROUP	Audits creation, modification, or deletion of audit objects and audit specifications, ensuring audit trail integrity (HIPAA, PCI, GDPR).
APPLICATION_ROLE_CHANGE_PASSWORD_GROUP	Tracks changes to application role passwords, securing application-level access (HIPAA, PCI).
LOGIN_CHANGE_PASSWORD_GROUP	Captures password changes for logins, supporting credential management policies (HIPAA, PCI).
SCHEMA_OBJECT_CHANGE_GROUP	Monitors DDL changes (CREATE, ALTER, DROP) on schema objects at the server level, important for tracking structural changes (HIPAA, GDPR).
DATABASE_OBJECT_PERMISSION_CHANGE_GROUP	Tracks permission changes on database objects, ensuring proper access control (HIPAA, PCI, GDPR).
DATABASE_PRINCIPAL_CHANGE_GROUP	Monitors changes to database principals (users, roles), supporting identity management (HIPAA, PCI).

SQL Audit Action Groups Cheat Sheet



SQL Server Audit -
Action Groups Mapping

Report Last Refresh
Date-Time at:
2025/10/28 18:51

SQL Server Audit Action Groups - Mapping Info

Action Group Name	Group Type	Group Description	Action Name	Class Name	Level
APPLICATION_ROLE_CHANGE_PASSWORD_GROUP	DATABASE	This event is raised whenever a password is changed for an application role.	APPLICATION_ROLE_CHANGE_PASSWORD_GROUP	DATABASE	Group
			CHANGE PASSWORD	APPLICATION ROLE	
	SERVER	This event is raised whenever a password is changed for an application role.	APPLICATION_ROLE_CHANGE_PASSWORD_GROUP	SERVER	Group
AUDIT_CHANGE_GROUP	DATABASE	This event is raised whenever any audit is created, modified or deleted. This event is raised whenever any audit specification is created, modified, or deleted. Any change to an audit is audited in that audit.	ALTER	AUDIT	
				DATABASE AUDIT SPECIFICATION	
			AUDIT_CHANGE_GROUP	DATABASE	Group
			CREATE	AUDIT	
				DATABASE AUDIT SPECIFICATION	
			DROP	AUDIT	
				DATABASE AUDIT SPECIFICATION	
	SERVER	This event is raised whenever any audit is created, modified or deleted. This event is raised whenever any audit specification is created, modified, or deleted. Any change to an audit is audited in that audit.	ALTER	SERVER AUDIT	
				SERVER AUDIT SPECIFICATION	
			AUDIT SESSION CHANGED	SERVER AUDIT	
			AUDIT SHUTDOWN ON FAILURE	SERVER AUDIT	

Actions

action _id	Action Name	SQL 2014	SQL 2016	SQL 2017	SQL 2019	SQL 2022	Action Number
AL	ALTER	X	X	X	X	X	538987585
CNAU	AUDIT_CHANGE_GROUP	X	X	X	X	X	1430343235
CR	CREATE	X	X	X	X	X	538989123
DR	DROP	X	X	X	X	X	538989124

Classes

class _type	Class Name	SQL 2014	SQL 2016	SQL 2017	SQL 2019	SQL 2022	Class Number
DU	AUDIT	X	X	X	X	X	21828
DB	DATABASE	X	X	X	X	X	16964
DA	DATABASE AUDIT SPECIFICATION	X	X	X	X	X	16708

Power BI Dashboard Link

SQL Audit 101– SQL Server vs. Azure SQL DB

SQL Server

- To **create, alter, or drop a server audit**, principals require the **ALTER ANY SERVER AUDIT** or the **CONTROL SERVER** permission.
- Users with the **ALTER ANY SERVER AUDIT** permission can create server audit specifications and bind them to any audit.
- After a server audit specification is created, it can be viewed by principals with the **CONTROL SERVER or ALTER ANY SERVER AUDIT** permissions, the **sysadmin** account, or **principals having explicit access to the audit**.

Azure SQL Database

- Need Contributor role or higher on the database or server resource
  - Permissions to execute 'Microsoft.Sql/servers/extendedAuditingSettings/write’
  - Permission to execute 'Microsoft.Sql/servers/databases/extendedAuditingSettings/write'
- The following audit policies **are included by default**
  - BATCH_COMPLETED_GROUP SUCCESSFUL_DATABASE_AUTHENTICATION_GROUP FAILED_DATABASE_AUTHENTICATION_GROUP
- Additional **changes can be** made via API calls.
- Enabling auditing on the database in addition to enabling auditing on the server doesn't override or change any of the settings of the server auditing

SQL Audit Targets – Where Do We Store the Audit Data?

Targets include

Azure - Blob storage, Log Analytics, Event Hub

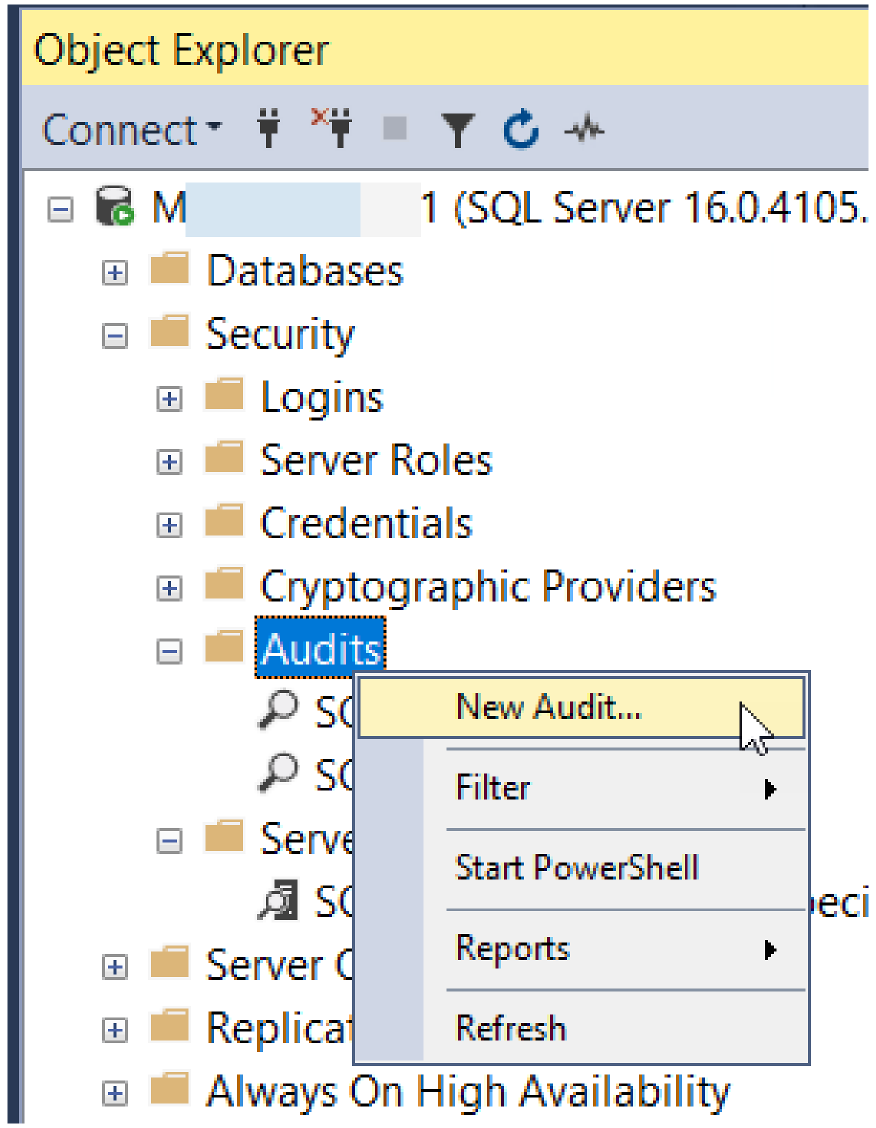
SQL Server – Files, Event Logs

Use file targets with appropriate **size limits (MAXSIZE)** and rollover files **(MAX_ROLLOVER_FILES)** to **prevent disk space issues.**

Reserve disk space upfront **(RESERVE_DISK_SPACE = ON)** to **avoid audit failures due to insufficient space**.

Creating A SQL Audit

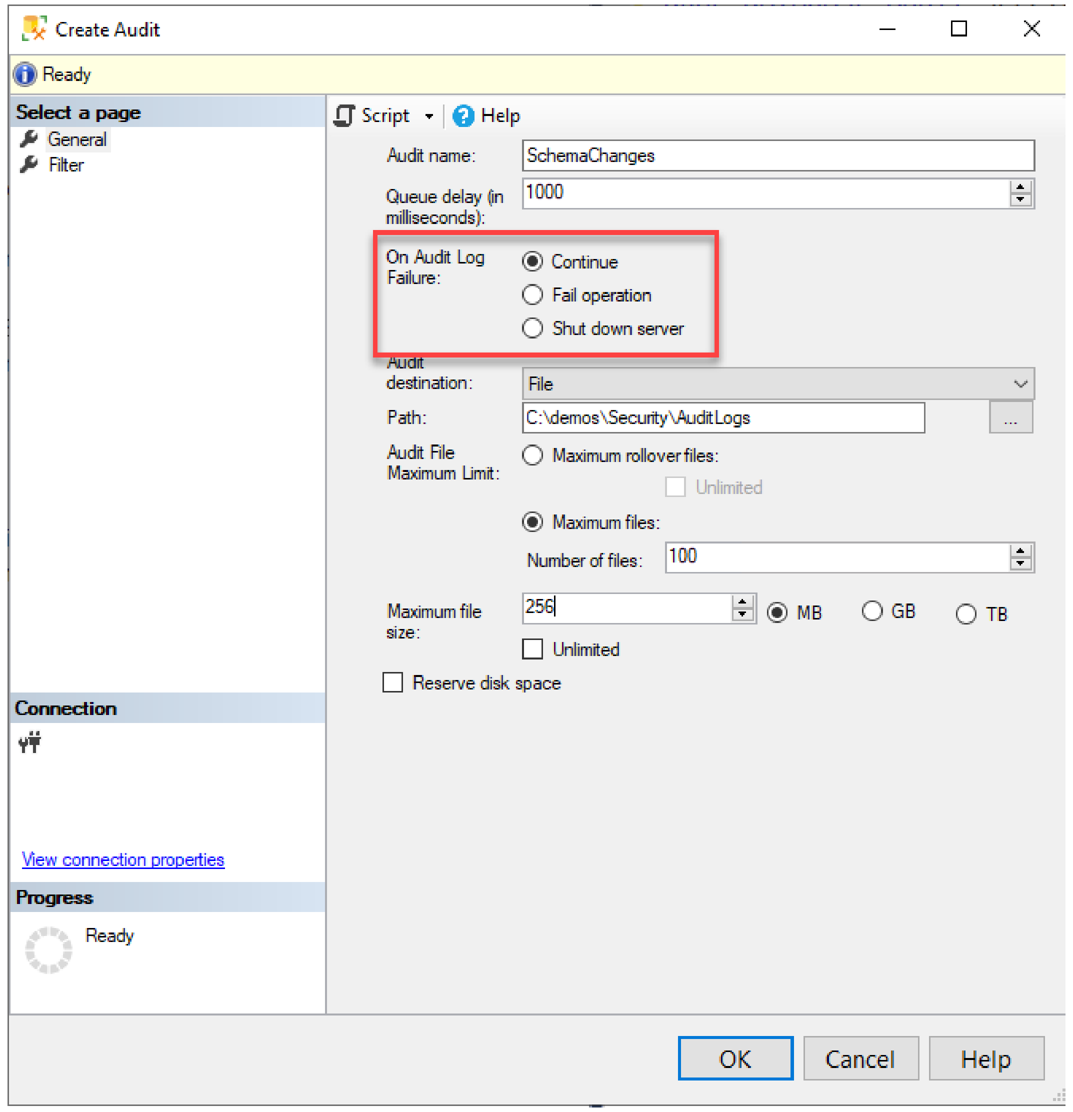
SQL Server



Object Explorer

Connect ▾

- ⊟ M[redacted]1 (SQL Server 16.0.4105.
  - ⊞ Databases
  - ⊟ Security
    - ⊞ Logins
    - ⊞ Server Roles
    - ⊞ Credentials
    - ⊞ Cryptographic Providers
    - ⊟ Audits
      - SQ
      - SQ
    - ⊟ Serve
      - SQ eci
  - ⊞ Server O
  - ⊞ Replicat
  - ⊞ Always On High Availability



Create Audit

Ready

Select a page
- General
- Filter

Script ▾ | Help

Audit name: SchemaChanges

Queue delay (in milliseconds): 1000

On Audit Log Failure:
- ● Continue
- ○ Fail operation
- ○ Shut down server

Audit destination: File

Path: C:\demos\Security\AuditLogs

Audit File Maximum Limit:
- ○ Maximum rollover files:
  - □ Unlimited
- ● Maximum files:
  - Number of files: 100

Maximum file size: 256 ● MB ○ GB ○ TB

□ Unlimited

□ Reserve disk space

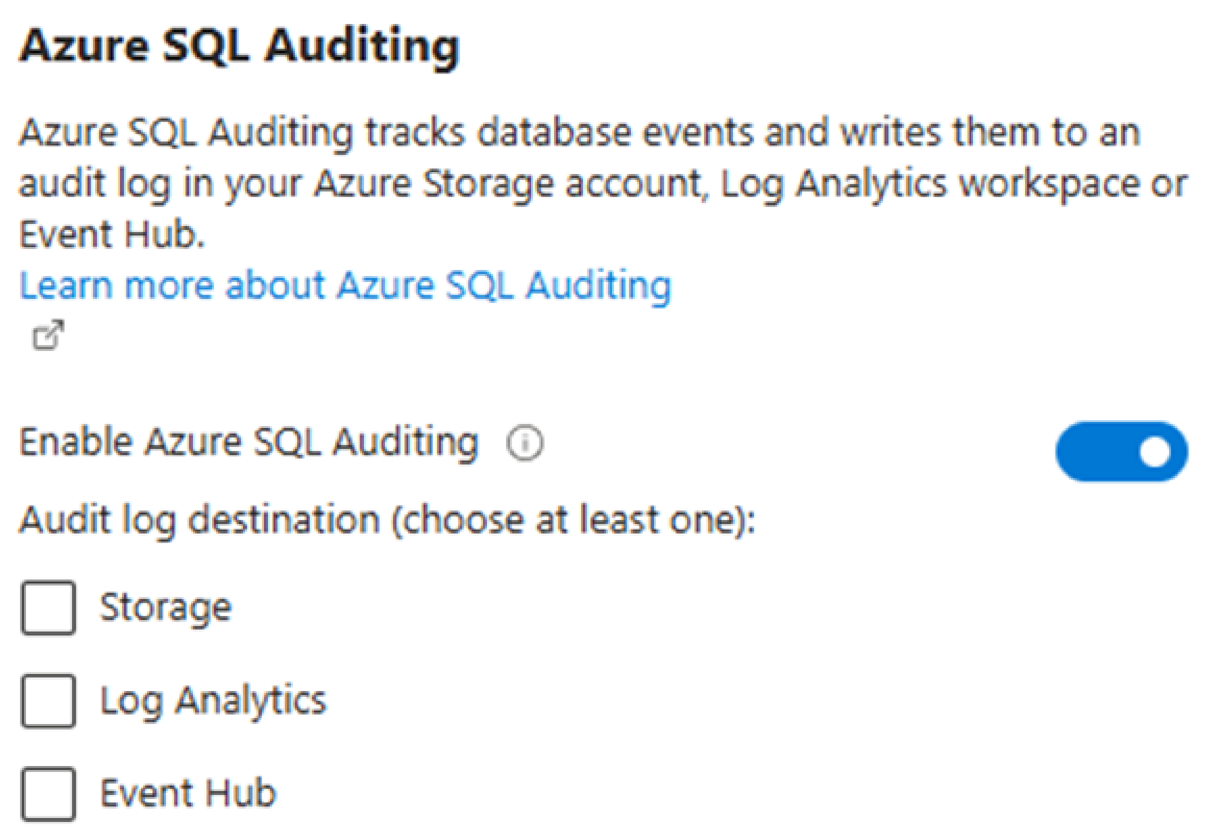
Connection

View connection properties

Progress

Ready

Azure SQL Database



Azure SQL Auditing

Azure SQL Auditing tracks database events and writes them to an audit log in your Azure Storage account, Log Analytics workspace or Event Hub.

Learn more about Azure SQL Auditing

Enable Azure SQL Auditing ⓘ ●

Audit log destination (choose at least one):

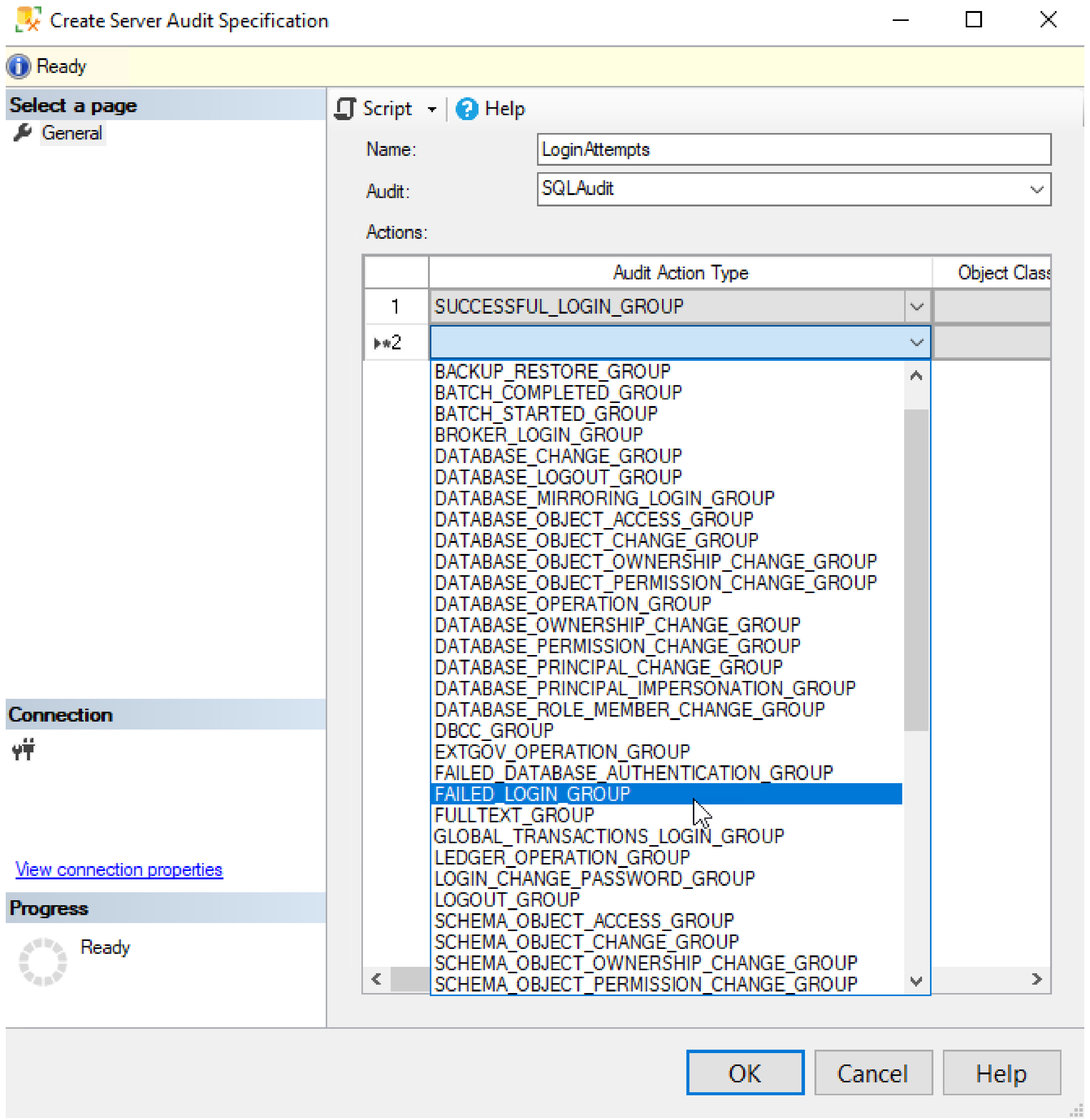
- □ Storage
- □ Log Analytics
- □ Event Hub

Creating A SQL Audit – T-SQL Basic Example

```
CREATE SERVER AUDIT
Audit_Compliance TO FILE (
FILEPATH = 'C:\AuditLogs\')
WITH (ON_FAILURE = CONTINUE);
GO

ALTER SERVER AUDIT
Audit_Compliance
WITH (STATE = ON);
GO
```

```
CREATE SERVER AUDIT SPECIFICATION AuditSpec_Compliance
FOR SERVER AUDIT Audit_Compliance
ADD (SUCCESSFUL_LOGIN_GROUP),
ADD (FAILED_LOGIN_GROUP),
ADD (SERVER_PRINCIPAL_CHANGE_GROUP),
ADD (SERVER_ROLE_MEMBER_CHANGE_GROUP),
ADD (AUDIT_CHANGE_GROUP),
ADD (LOGIN_CHANGE_PASSWORD_GROUP),
ADD (APPLICATION_ROLE_CHANGE_PASSWORD_GROUP);
GO
ALTER SERVER AUDIT SPECIFICATION AuditSpec_Compliance WITH
(STATE = ON);
GO

CREATE DATABASE AUDIT [CustomerDataAudit]
FOR SERVER AUDIT [SQLAudit]
ADD (SELECT ON OBJECT::[dbo].[Customers] BY [public]),
ADD (UPDATE ON OBJECT::[dbo].[Customers] BY [public]),
ADD (INSERT ON OBJECT::[dbo].[Customers] BY [public]),
ADD (DELETE ON OBJECT::[dbo].[Customers] BY [public])
```

How to Audit Events – What Should We Monitor?



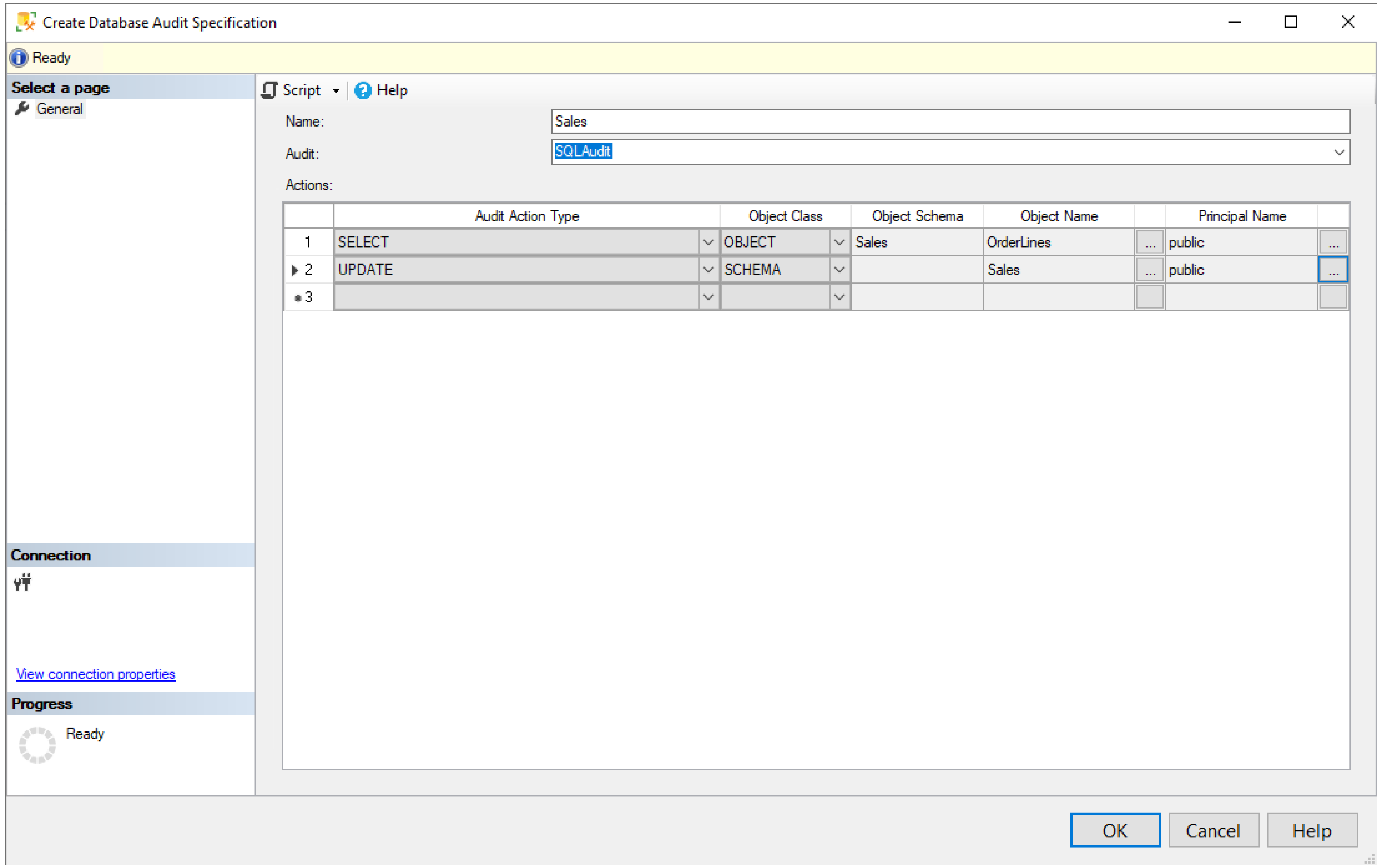
Name: LoginAttempts

Audit: SQLAudit

Actions:

	Audit Action Type	Object Class
1	SUCCESSFUL_LOGIN_GROUP	
2		

BACKUP_RESTORE_GROUP
BATCH_COMPLETED_GROUP
BATCH_STARTED_GROUP
BROKER_LOGIN_GROUP
DATABASE_CHANGE_GROUP
DATABASE_LOGOUT_GROUP
DATABASE_MIRRORING_LOGIN_GROUP
DATABASE_OBJECT_ACCESS_GROUP
DATABASE_OBJECT_CHANGE_GROUP
DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP
DATABASE_OBJECT_PERMISSION_CHANGE_GROUP
DATABASE_OPERATION_GROUP
DATABASE_OWNERSHIP_CHANGE_GROUP
DATABASE_PERMISSION_CHANGE_GROUP
DATABASE_PRINCIPAL_CHANGE_GROUP
DATABASE_PRINCIPAL_IMPERSONATION_GROUP
DATABASE_ROLE_MEMBER_CHANGE_GROUP
DBCC_GROUP
EXTGOV_OPERATION_GROUP
FAILED_DATABASE_AUTHENTICATION_GROUP
FAILED_LOGIN_GROUP
FULLTEXT_GROUP
GLOBAL_TRANSACTIONS_LOGIN_GROUP
LEDGER_OPERATION_GROUP
LOGIN_CHANGE_PASSWORD_GROUP
LOGOUT_GROUP
SCHEMA_OBJECT_ACCESS_GROUP
SCHEMA_OBJECT_CHANGE_GROUP
SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP
SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP



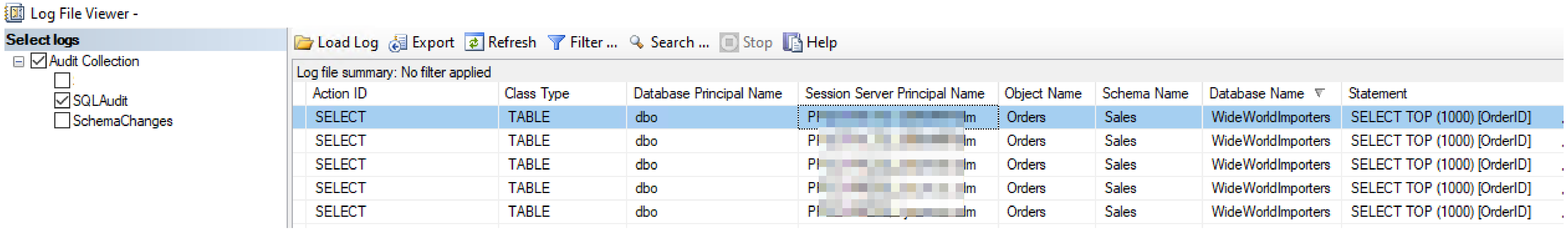
Name: Sales

Audit: SQLAudit

Actions:

	Audit Action Type	Object Class	Object Schema	Object Name	Principal Name
1	SELECT	OBJECT	Sales	OrderLines	public
2	UPDATE	SCHEMA		Sales	public
3					

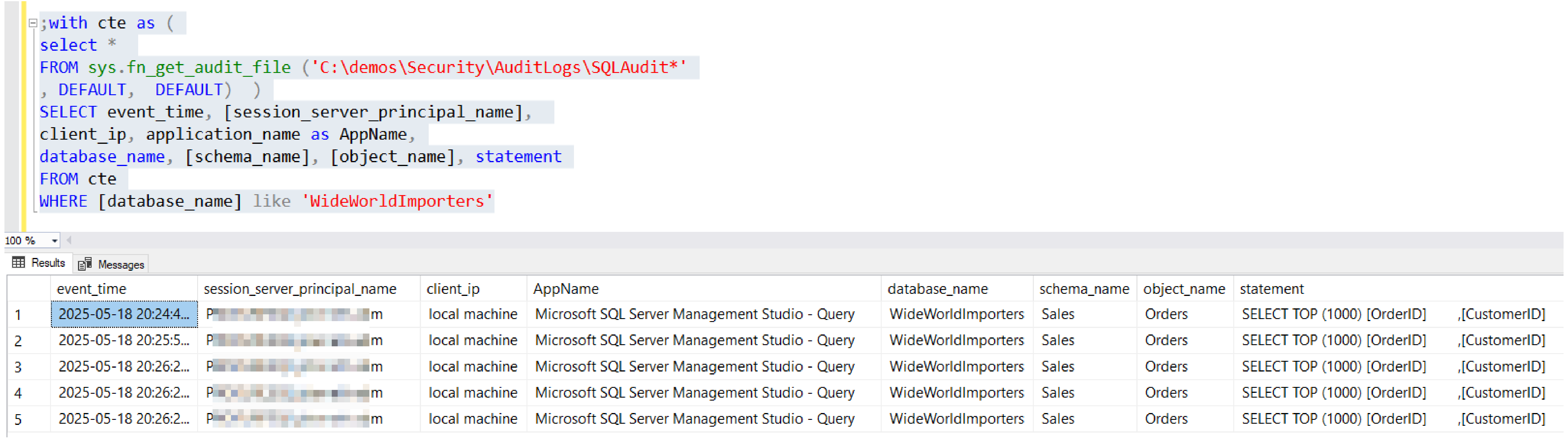
Log File Viewer – How do we view the Audit Data?



- ☑ Audit Collection
  - □
  - ☑ SQLAudit
  - □ SchemaChanges

Log file summary: No filter applied

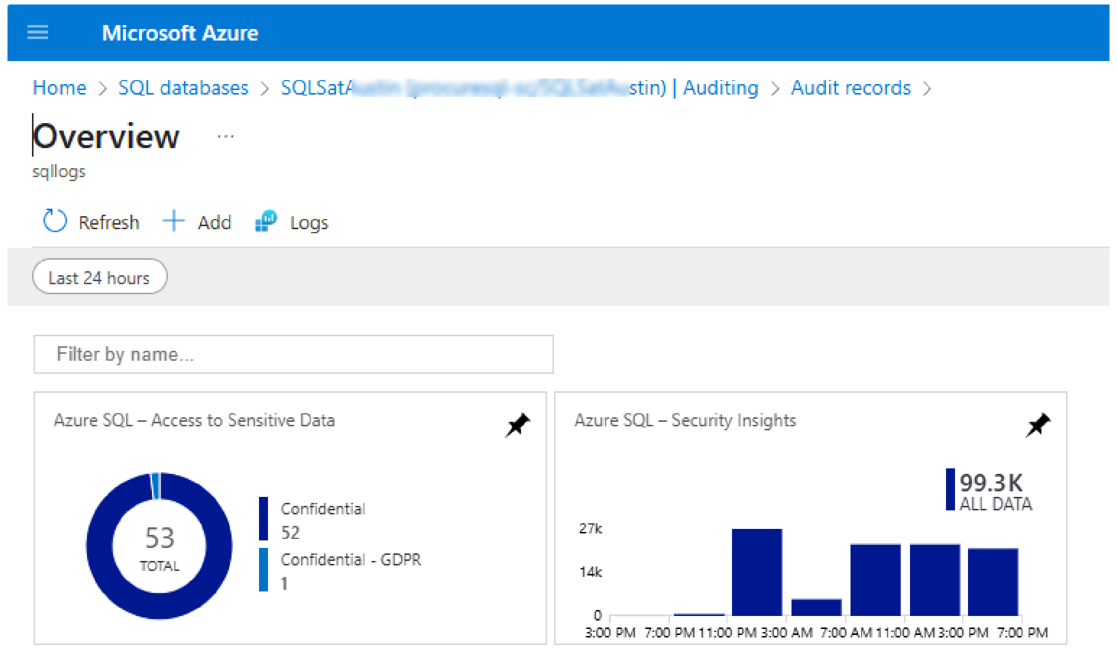
Action ID	Class Type	Database Principal Name	Session Server Principal Name	Object Name	Schema Name	Database Name	Statement
SELECT	TABLE	dbo	[redacted]	Orders	Sales	WideWorldImporters	SELECT TOP (1000) [OrderID]
SELECT	TABLE	dbo	[redacted]	Orders	Sales	WideWorldImporters	SELECT TOP (1000) [OrderID]
SELECT	TABLE	dbo	[redacted]	Orders	Sales	WideWorldImporters	SELECT TOP (1000) [OrderID]
SELECT	TABLE	dbo	[redacted]	Orders	Sales	WideWorldImporters	SELECT TOP (1000) [OrderID]
SELECT	TABLE	dbo	[redacted]	Orders	Sales	WideWorldImporters	SELECT TOP (1000) [OrderID]

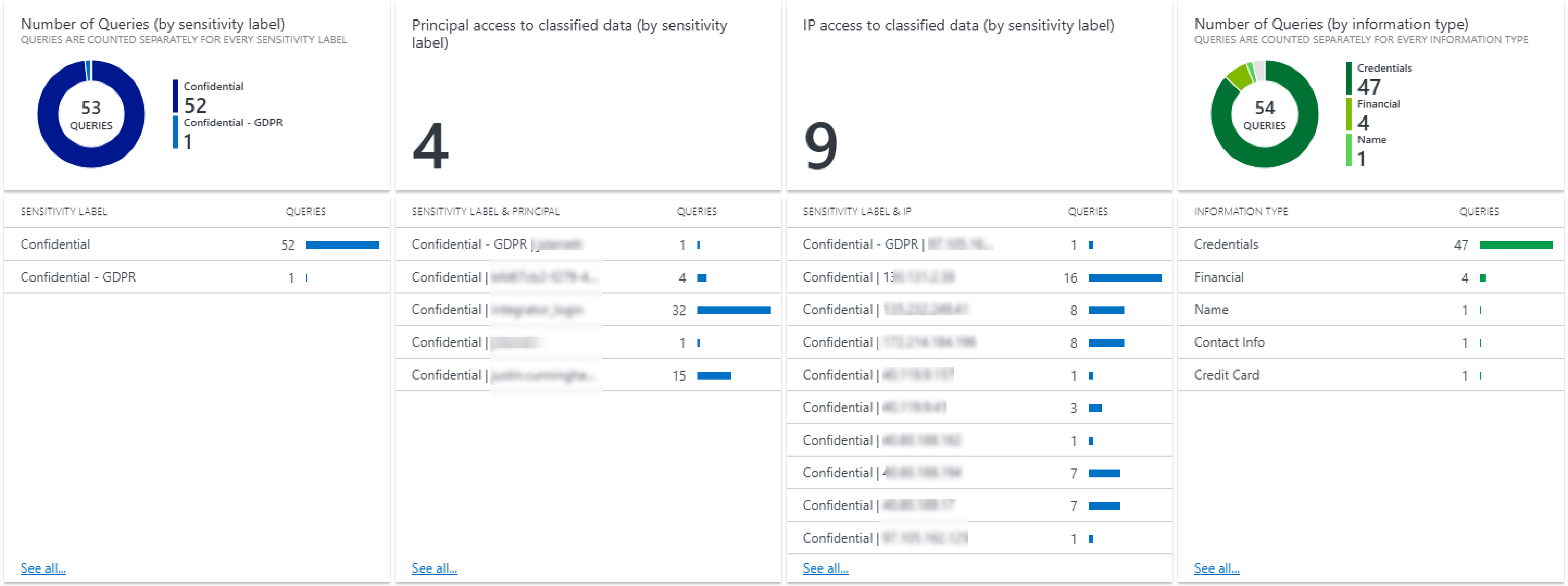


```
;with cte as (
select *
FROM sys.fn_get_audit_file ('C:\demos\Security\AuditLogs\SQLAudit*'
, DEFAULT,  DEFAULT)  )
SELECT event_time, [session_server_principal_name],
client_ip, application_name as AppName,
database_name, [schema_name], [object_name], statement
FROM cte
WHERE [database_name] like 'WideWorldImporters'
```

	event_time	session_server_principal_name	client_ip	AppName	database_name	schema_name	object_name	statement
1	2025-05-18 20:24:4...	[redacted]	local machine	Microsoft SQL Server Management Studio - Query	WideWorldImporters	Sales	Orders	SELECT TOP (1000) [OrderID] ,[CustomerID]
2	2025-05-18 20:25:5...	[redacted]	local machine	Microsoft SQL Server Management Studio - Query	WideWorldImporters	Sales	Orders	SELECT TOP (1000) [OrderID] ,[CustomerID]
3	2025-05-18 20:26:2...	[redacted]	local machine	Microsoft SQL Server Management Studio - Query	WideWorldImporters	Sales	Orders	SELECT TOP (1000) [OrderID] ,[CustomerID]
4	2025-05-18 20:26:2...	[redacted]	local machine	Microsoft SQL Server Management Studio - Query	WideWorldImporters	Sales	Orders	SELECT TOP (1000) [OrderID] ,[CustomerID]
5	2025-05-18 20:26:2...	[redacted]	local machine	Microsoft SQL Server Management Studio - Query	WideWorldImporters	Sales	Orders	SELECT TOP (1000) [OrderID] ,[CustomerID]

Sensitive Data Metrics Found in our SQL Audit





Number of Queries (by sensitivity label)
QUERIES ARE COUNTED SEPARATELY FOR EVERY SENSITIVITY LABEL

Confidential 52
Confidential - GDPR 1

SENSITIVITY LABEL	QUERIES
Confidential	52
Confidential - GDPR	1

See all...

Principal access to classified data (by sensitivity label)

4

SENSITIVITY LABEL & PRINCIPAL	QUERIES
Confidential - GDPR | [redacted]	1
Confidential | [redacted]	4
Confidential | [redacted]	32
Confidential | [redacted]	1
Confidential | [redacted]	15

See all...

IP access to classified data (by sensitivity label)

9

SENSITIVITY LABEL & IP	QUERIES
Confidential - GDPR | [redacted]	1
Confidential | 13[redacted]	16
Confidential | [redacted]	8
Confidential | [redacted]	8
Confidential | [redacted]	1
Confidential | [redacted]	3
Confidential | [redacted]	1
Confidential | [redacted]	7
Confidential | [redacted]	7
Confidential | [redacted]	1

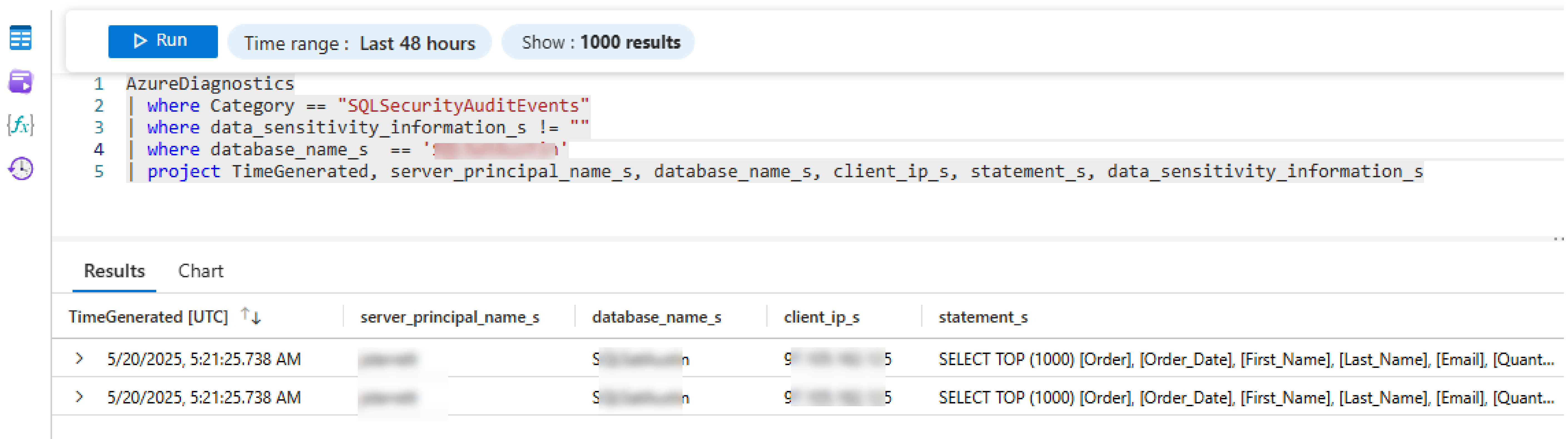
See all...

Number of Queries (by information type)
QUERIES ARE COUNTED SEPARATELY FOR EVERY INFORMATION TYPE

INFORMATION TYPE	QUERIES
Credentials	47
Financial	4
Name	1
Contact Info	1
Credit Card	1

See all...

New Query 1* ··· × +



```
AzureDiagnostics
| where Category == "SQLSecurityAuditEvents"
| where data_sensitivity_information_s != ""
| where database_name_s   == '[redacted]'
| project TimeGenerated, server_principal_name_s, database_name_s, client_ip_s, statement_s, data_sensitivity_information_s
```

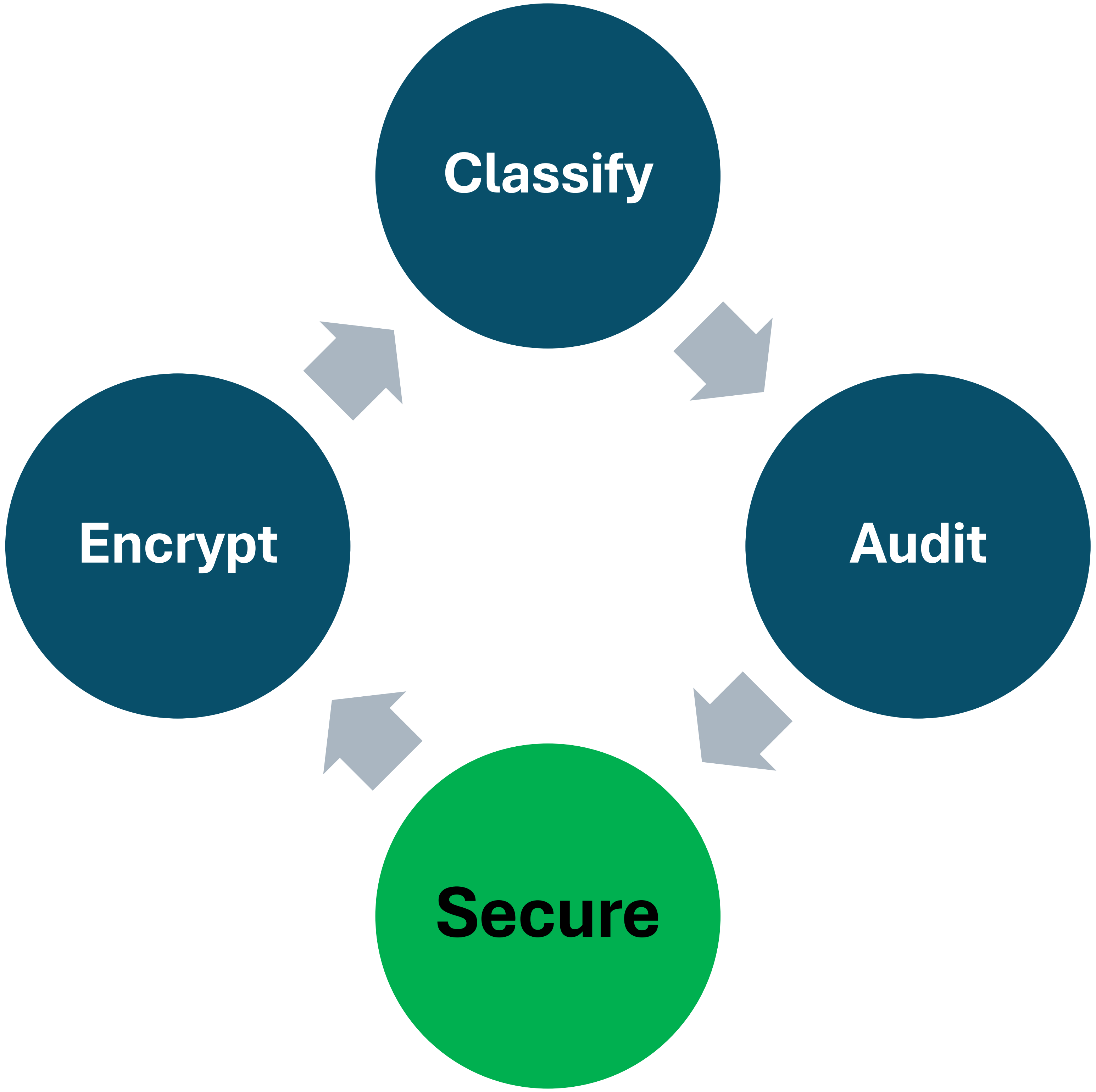
TimeGenerated [UTC]	server_principal_name_s	database_name_s	client_ip_s	statement_s
5/20/2025, 5:21:25.738 AM	[redacted]	S[redacted]n	9[redacted]5	SELECT TOP (1000) [Order], [Order_Date], [First_Name], [Last_Name], [Email], [Quant...
5/20/2025, 5:21:25.738 AM	[redacted]	S[redacted]n	9[redacted]5	SELECT TOP (1000) [Order], [Order_Date], [First_Name], [Last_Name], [Email], [Quant...

SQL Audit Best Practices

Define	Clear Audit Goals and Scope
Use	Server and Database Audit specifications – Limit Tracked Events
Review	Audit Logs Regularly
Secure	Secure Your Audit Logs
Backup	Your Audit Logs
Audit	Your Audit - Verify intended events are being logged

Row-Level Security (RLS)

Securing Your Data at the Row Level



Why Row-Level Security?

Applications need to limit a users access to only certain rows of data in a database. **Security needs to be embedded in the database to work for ALL APPLICATIONS.**

Control both read and write data at the row level

- No app changes needed, works transparently when queries execute
- Centralized Security Logic within the database
- Apps consume secured data
  - Excel, .NET, **Power BI – Direct Query**, etc.…

RLS – Real-World Examples

Health Care (Patient Data Access Controls)

Nurses can only view rows of their assigned patients

Doctors access broader data but are blocked from data unless authorized

Patients can only see their data

Financial Services

Financial Advisors only see their client’s portfolios

Auditors access transition history for only the Financial Advisors they audit

E-Commerce / Multi-Tenant

Vendors view only their sales records and customer orders

Platform admins access data for vendors assigned to them.

How does RLS Work?

Predicate-based access control added to regular access

Two types of security predicates

- **Filter predicates** – silently filter SELECT, UPDATE and DELETE operations to exclude rows that will not satisfy the predicate
- **Block predicates** – block INSERT, UPDATE, DELETE operations that will not satisfy the predicate
  - **AFTER INSERT and AFTER UPDATE** predicates can prevent users from updating rows to values that violate the predicate.
  - **BEFORE UPDATE** predicates can prevent users from updating rows that currently violate the predicate.
  - **BEFORE DELETE** predicates can block delete operations.

How to implement RLS

Each row of your table has column that determine which user can access the data

CustomerID	FirstName	LastName	SalesRep
1001			John
1002			Kon
1003			John

Create inline table-value function that defines row level access criteria

```
CREATE FUNCTION RLS.CustomerPredicate (@SalesRep AS sysname)
RETURNS TABLE
WITH SCHEMABINDING
AS
RETURN SELECT 1 AS Access
WHERE @SalesRep = USER_NAME() OR USER_NAME() = 'Manager'
GO
```

Security policy adds security predicates on tables using the function provided

```
CREATE SECURITY POLICY RLS.CustomerPolicy
ADD FILTER PREDICATE RLS.CustomerPredicate(SalesRep) ON
Sales.Customer,
ADD BLOCK PREDICATE RLS.CustomerPredicate(SalesRep) ON
Sales.Customer
GO
```

Authorization Methods with RLS

Any lookup defined by business rules can be used

Lookup Options

- SESSION_CONTEXT() – Applications
- Local Lookup table
- SQL Roles
- Specific users – admins as an example

RLS Example ☕ (Group Activity)

dbo.Sales - Table

User Name	Country	Sales
Fred	USA	10000
Chris	USA	9500
Tom	France	9600
Fred	Spain	9200
Chris	Germany	9000

Database Users

CEO Fred Chris Tom

Database Roles

USA France

RLS – User Lookup Example

```
CREATE FUNCTION Security.fn_SalesSecurity(@UserName AS
sysname)
RETURNS TABLE
WITH SCHEMABINDING
AS

   RETURN SELECT 1 AS fn_SalesSecurity_Result
   -- Logic for filter predicate
   WHERE @UserName = USER_NAME() OR USER_NAME() = 'CEO’;
GO

CREATE SECURITY POLICY Security.UserFilter
ADD FILTER PREDICATE Security.fn_SalesSecurity(UserName)
ON dbo.Sales WITH (STATE = ON);
GO

EXECUTE AS USER = 'CEO’;
SELECT * FROM Sales;
REVERT;
GO

EXECUTE AS USER = 'Fred’;
SELECT * FROM Sales;
REVERT;
```

Database User

CEO

Database User

Fred

dbo.Sales - Table

User Name	Country	Sales
Fred	USA	10000
Chris	USA	9500
Tom	France	9600
Fred	Spain	9200
Chris	Germany	9000

User Name	Country	Sales
Fred	USA	10000
Fred	Spain	9200

RLS – SQL Roles Example

```
ALTER ROLE [USA] ADD MEMBER [CEO]
ALTER ROLE [FRANCE] ADD MEMBER [CEO]
ALTER ROLE [SPAIN] ADD MEMBER [CEO]

ALTER ROLE [USA] ADD MEMBER [Fred]
ALTER ROLE [USA] ADD MEMBER [Chris]


ALTER ROLE [FRANCE] ADD MEMBER [Tom]
ALTER ROLE [SPAIN] ADD MEMBER [Fred]
ALTER ROLE [GERMANY] ADD MEMBER [Chris]

CREATE FUNCTION Security.fn_SalesSecurity(@RoleName AS sysname)
RETURNS TABLE
WITH SCHEMABINDING AS
RETURN ( SELECT 1 AS AccessGranted
         WHERE IS_ROLEMEMBER(@RoleName) = 1);

GO

EXECUTE AS USER = 'CEO’;
SELECT * FROM Sales;
REVERT;
GO

EXECUTE AS USER = ‘Fred’;
SELECT * FROM Sales;
REVERT;
GO
```

Database User

CEO

dbo.Sales - Table

User Name	Country	Sales
Fred	USA	10000
Chris	USA	9500
Tom	France	9600
Fred	Spain	9200
Chris	Germany	9000

Database User

Fred

User Name	Country	Sales
Fred	USA	10000
Fred	Spain	9200

RLS – Lookup Table Example

```
CREATE TABLE RLS.UsersSuppliers (

UsersSuppliersID int NOT NULL CONSTRAINT
PK_RLSUsersSuppliers PRIMARY KEY CLUSTERED IDENTITY
,UserID nvarchar(255) NOT NULL
,SupplierID int NOT NULL )
--Grant the test user access
--to a single supplier ID

INSERT INTO RLS.UsersSuppliers (UserID, SupplierID)

VALUES ('RLSLookupUser’,4)
```

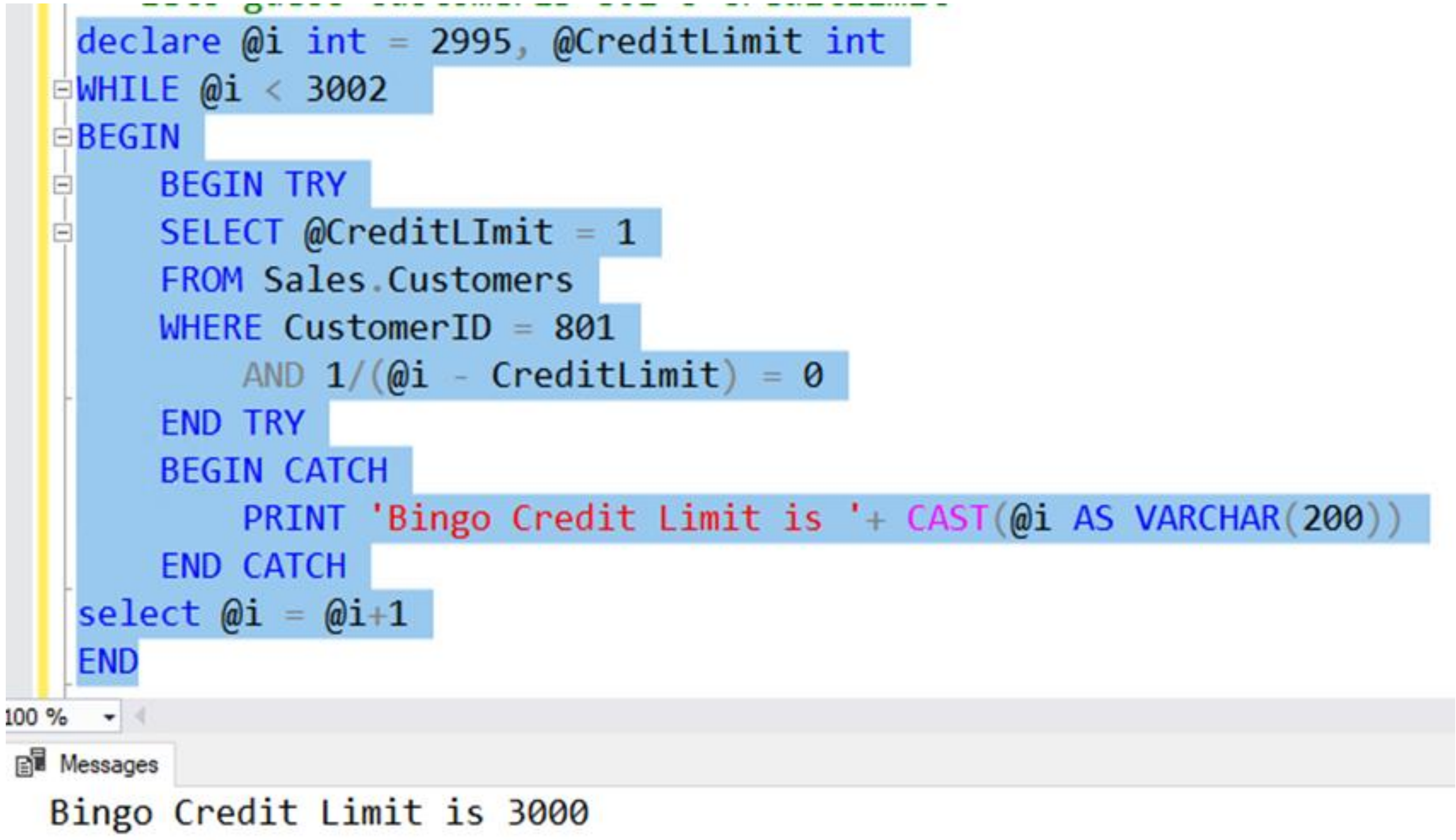
UserSuppliersID	UserID	SuppliersID
1	RLSLookupUser	4

```
ALTER ROLE [USA] ADD MEMBER [CEO]
ALTER ROLE [FRANCE] ADD MEMBER [CEO]
ALTER ROLE [SPAIN] ADD MEMBER [CEO]

ALTER ROLE [USA] ADD MEMBER [Fred]
ALTER ROLE [USA] ADD MEMBER [Chris]


ALTER ROLE [FRANCE] ADD MEMBER [Tom]
ALTER ROLE [SPAIN] ADD MEMBER [Fred]
ALTER ROLE [GERMANY] ADD MEMBER [Chris]

CREATE FUNCTION Security.fn_SalesSecurity(@RoleName AS sysname)
RETURNS TABLE
WITH SCHEMABINDING AS
RETURN ( SELECT 1 AS AccessGranted
         WHERE IS_ROLEMEMBER(@RoleName) = 1);

GO

EXECUTE AS USER = 'CEO’;
SELECT * FROM Sales;
REVERT;
GO

EXECUTE AS USER = ‘Fred’;
SELECT * FROM Sales;
REVERT;
GO
```

RLS – Side Attacks

- Malicious Security Policy Manager
- Divide By Zero Attack
- Cross-feature compatibility

Divide By Zero Attack



```sql
declare @i int = 2995, @CreditLimit int
WHILE @i < 3002
BEGIN
    BEGIN TRY
    SELECT @CreditLImit = 1
    FROM Sales.Customers
    WHERE CustomerID = 801
        AND 1/(@i - CreditLimit) = 0
    END TRY
    BEGIN CATCH
        PRINT 'Bingo Credit Limit is '+ CAST(@i AS VARCHAR(200))
    END CATCH
select @i = @i+1
END
```

```
Bingo Credit Limit is 3000
```

RLS – How to Identify Side Attacks?

Excessive Errors – 8134 (Divide By Zero)

SQL Server Side Trace

Extended Events

Server or Database Audits (Why we started with Audits ☺)

SQL Advanced Threat Protection

Performance changes

Excessive CPU Usage

Excessive requests per second

RLS Best Practices

It's highly recommended to create a separate schema for the RLS objects: predicate functions, and security policies.

The security policy manager doesn't require SELECT permission on the tables they protect.

Keep predicate functions short and sweet. Avoid using excessive table joins in predicate functions to maximize performance.

Follow regular performance tuning best practices for predicates.

RLS – SQL Features That Don’t Play

DBCC SHOW_STATISTICS

Filestream (Not Supported)

Memory-Optimized Tables

Indexed Views

Change Data Capture

Full-Text Search

Columnstore Indexes

Partitioned Views

Temporal Tables

More Details : Microsoft Learn RLS Cross-Feature Compatibility

RLS – Anti-Patterns

Using Features that can introduce data leakage

Highly Transactional Systems

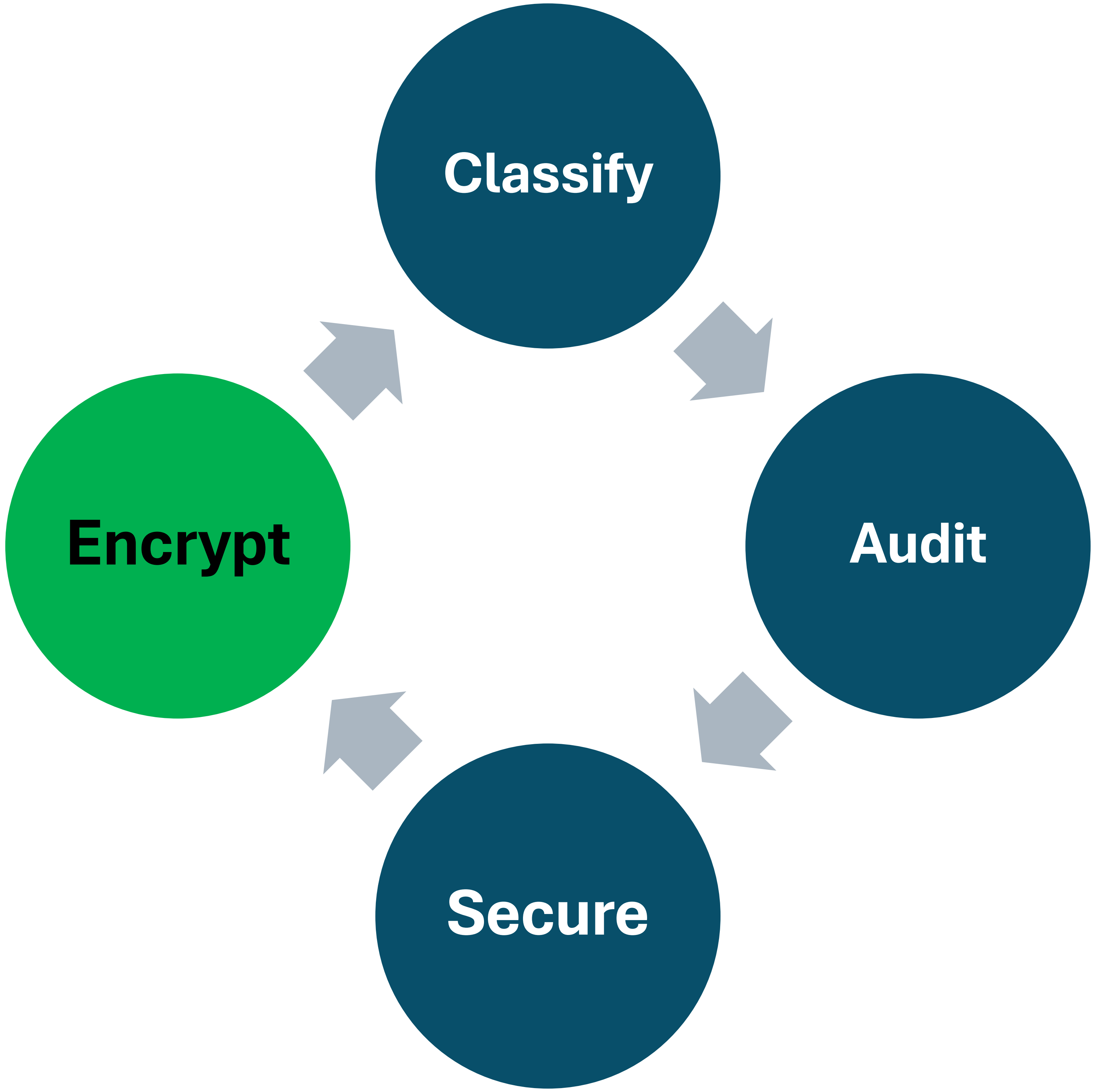
Databases without Direct user access

Less Experienced Teams

Staging or Loading Tables

Always Encrypted (AE)

Encrypting sensitive data from everyone (Yes, DBA's and System Admins too)



Why Should We Use Always Encrypted?

Feature	Encrypt Data At	Protects Against	Key Management	App Changed Needed
Transparent Data Encryption (TDE)	Rest (files)	Physical theft	Internal or EKM	No
Backup Encryption	Backup files	Backup theft	Internal or EKM	No
Column Level Encryption	Column Level	Unauthorized access to column data	Internal or EKM	Yes
TLS	Transit	Network eavesdropping	N/A	No
Always Encrypted	Column Level	DBAs, admins, memory attacks	External (client side)	Yes

AE – Encryption Keys

```
CREATE COLUMN MASTER KEY [CMK1]
WITH
(
KEY_STORE_PROVIDER_NAME =
N'MSSQL_CERTIFICATE_STORE’,
KEY_PATH = N'LocalMachine/My/2379554....’,
ENCLAVE_COMPUTATIONS (SIGNATURE = 0x5B1A... )

COLUMN ENCRYPTION KEY [CEK1]
WITH VALUES
(
 COLUMN_MASTER_KEY = [AE_CMK1],
ALGORITHM = 'RSA_OAEP’,
ENCRYPTED_VALUE = 0x01700000016...
)
```

Two-level key hierarchy

- Column encryption keys (CEK) – encrypts data
- Column master keys (CMKs) – encrypts CEKs

The database stores metadata about keys

- Enclave-enabled – CMKs have ENCLAVE_COMPUTATIONS set
- Enclave-enabled CEKs are encrypted with enclave-enabled CMKS

AE – Key Storage Decisions

Windows – Certificate Store

- Control and Compliance
- No Cloud Dependency
- Existing Infrastructure Utilization
- Simple Implementation for Smaller Deployments
- No Additional Costs

Azure Key vault

- Requires Azure Key Vault Access
- Separation of Duties
- End to End Protection
- Centralized Management
- Scalable and Flexibility
- Enhanced Security (RBAC)
- Simple Key Rotation

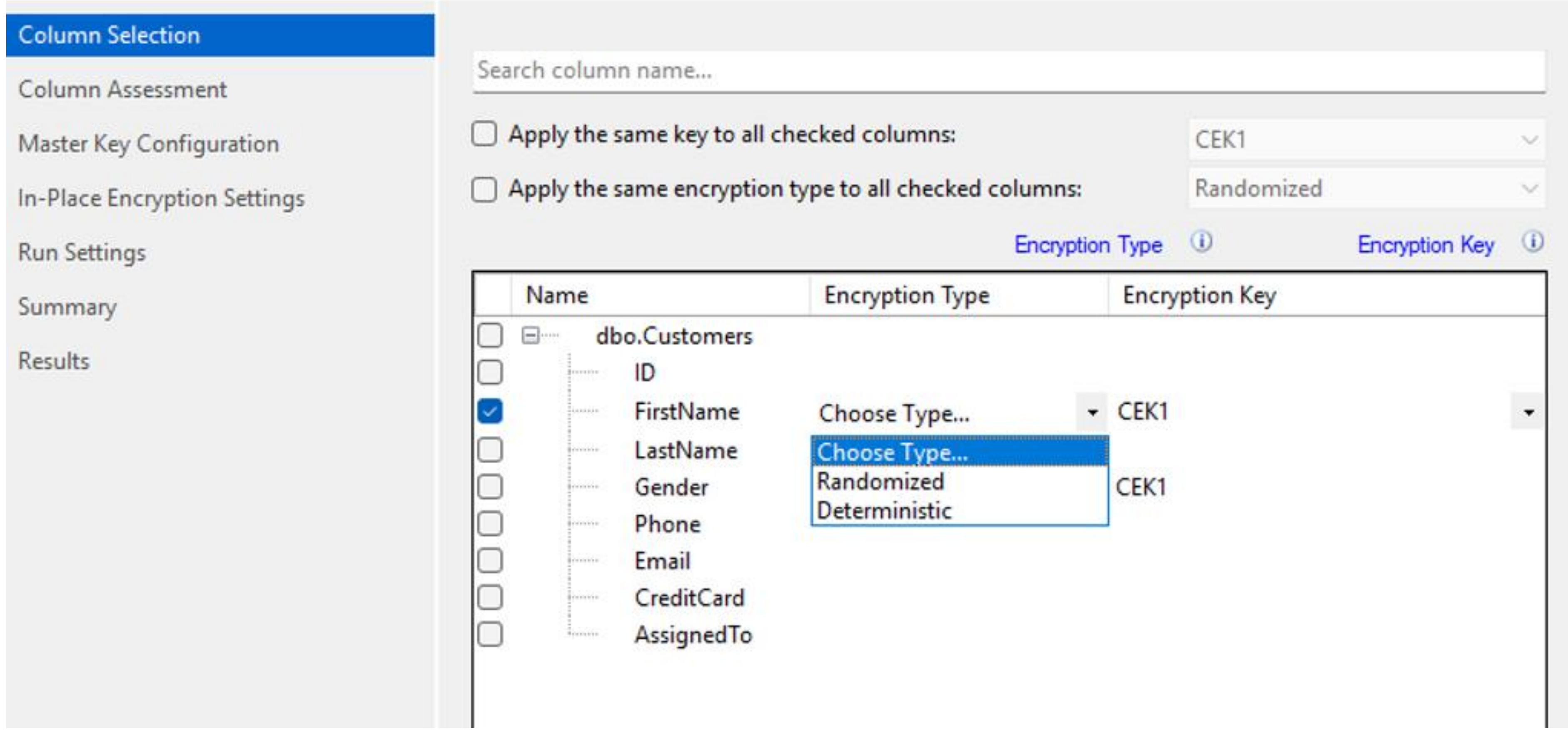
Always Encrypted Types in the Beginning

Deterministic

- Less Secure and Predictable
- Great WHERE clause equality
- JOINS
- Indexes

Randomize

- More Secure
- **Non-Searchable (SQL 2016 days)**



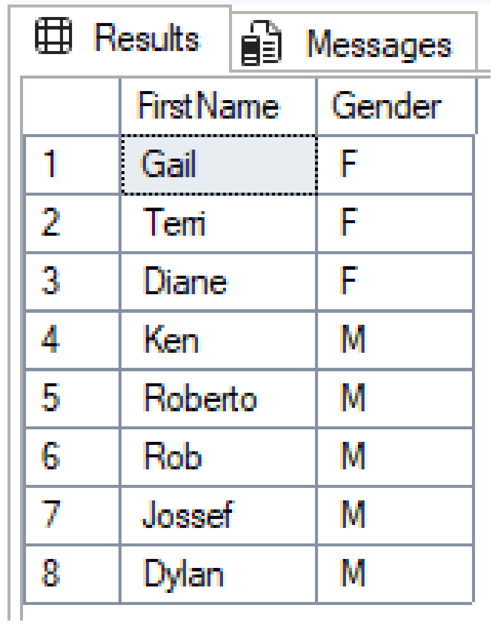
□ Apply the same key to all checked columns: CEK1

□ Apply the same encryption type to all checked columns: Randomized

	Name	Encryption Type	Encryption Key
□	dbo.Customers		
□	ID		
☑	FirstName	Choose Type...	CEK1
□	LastName	Choose Type... Randomized Deterministic	
□	Gender		CEK1
□	Phone		
□	Email		
□	CreditCard		
□	AssignedTo		

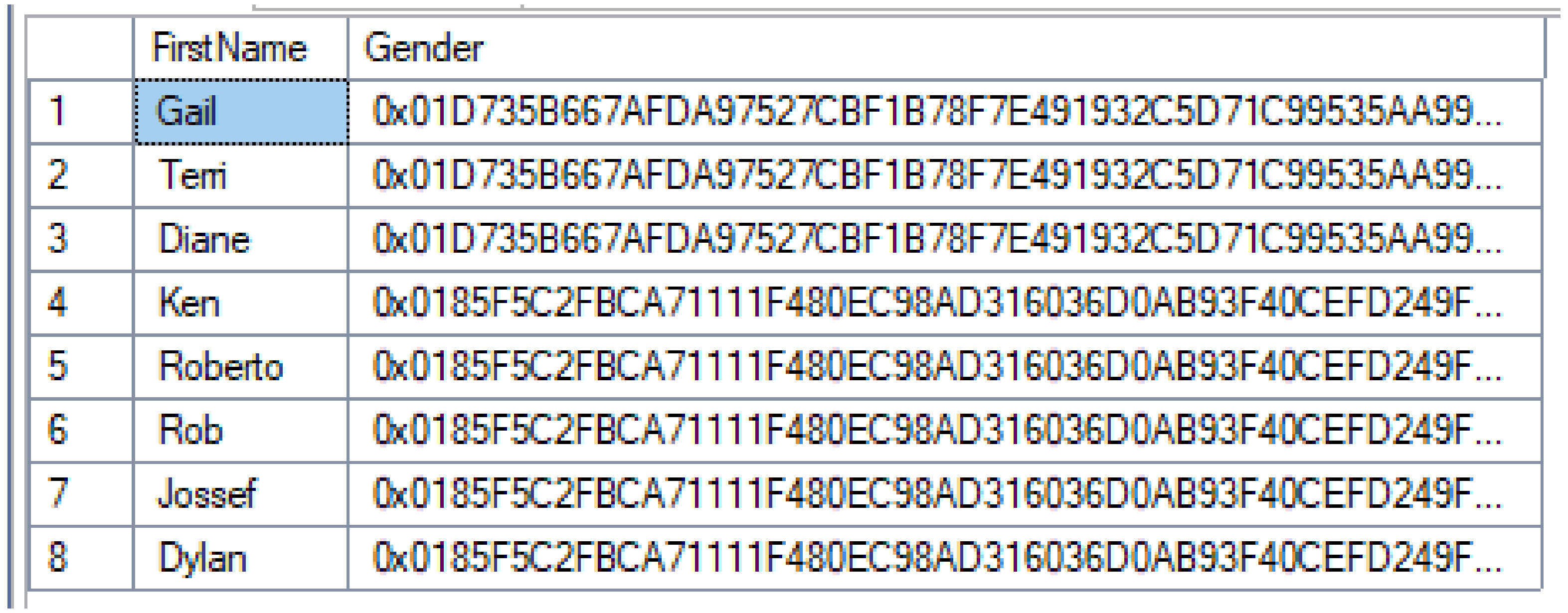
AE - Deterministic vs Randomized

Plaintext



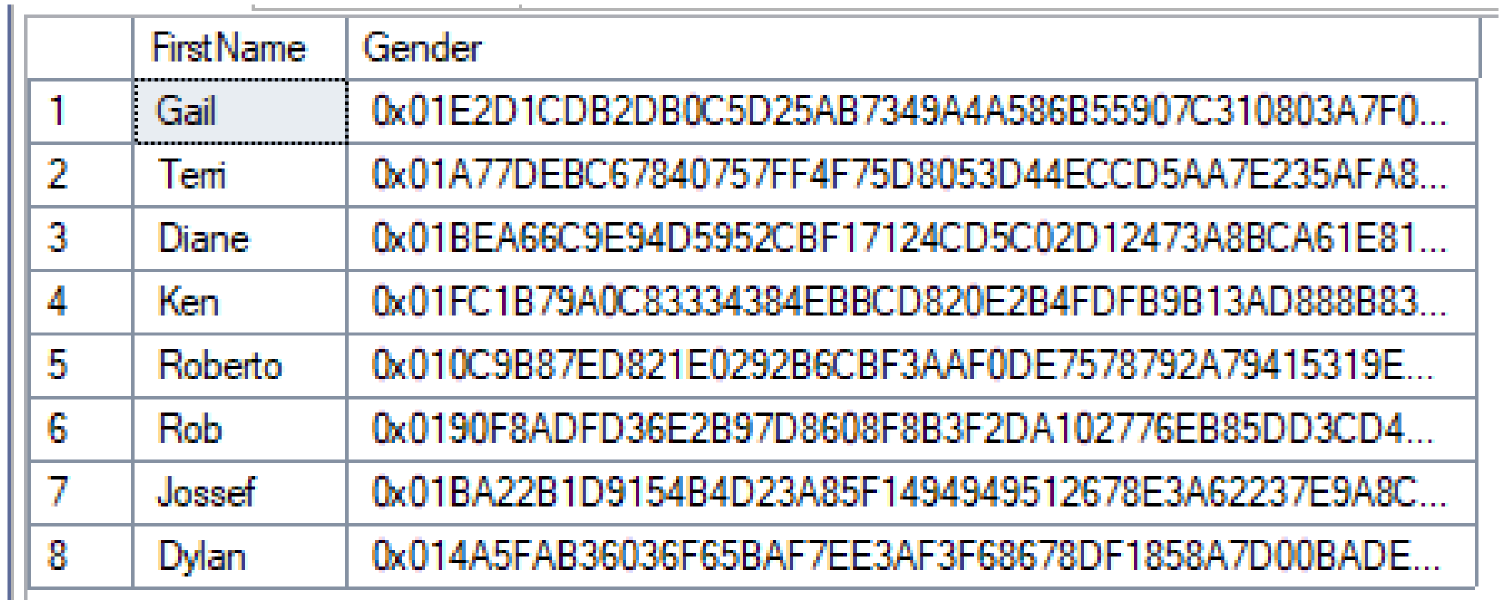
	FirstName	Gender
1	Gail	F
2	Terri	F
3	Diane	F
4	Ken	M
5	Roberto	M
6	Rob	M
7	Jossef	M
8	Dylan	M

Deterministic (Not Random)



	FirstName	Gender
1	Gail	0x01D735B667AFDA97527CBF1B78F7E491932C5D71C99535AA99...
2	Terri	0x01D735B667AFDA97527CBF1B78F7E491932C5D71C99535AA99...
3	Diane	0x01D735B667AFDA97527CBF1B78F7E491932C5D71C99535AA99...
4	Ken	0x0185F5C2FBCA71111F480EC98AD316036D0AB93F40CEFD249F...
5	Roberto	0x0185F5C2FBCA71111F480EC98AD316036D0AB93F40CEFD249F...
6	Rob	0x0185F5C2FBCA71111F480EC98AD316036D0AB93F40CEFD249F...
7	Jossef	0x0185F5C2FBCA71111F480EC98AD316036D0AB93F40CEFD249F...
8	Dylan	0x0185F5C2FBCA71111F480EC98AD316036D0AB93F40CEFD249F...

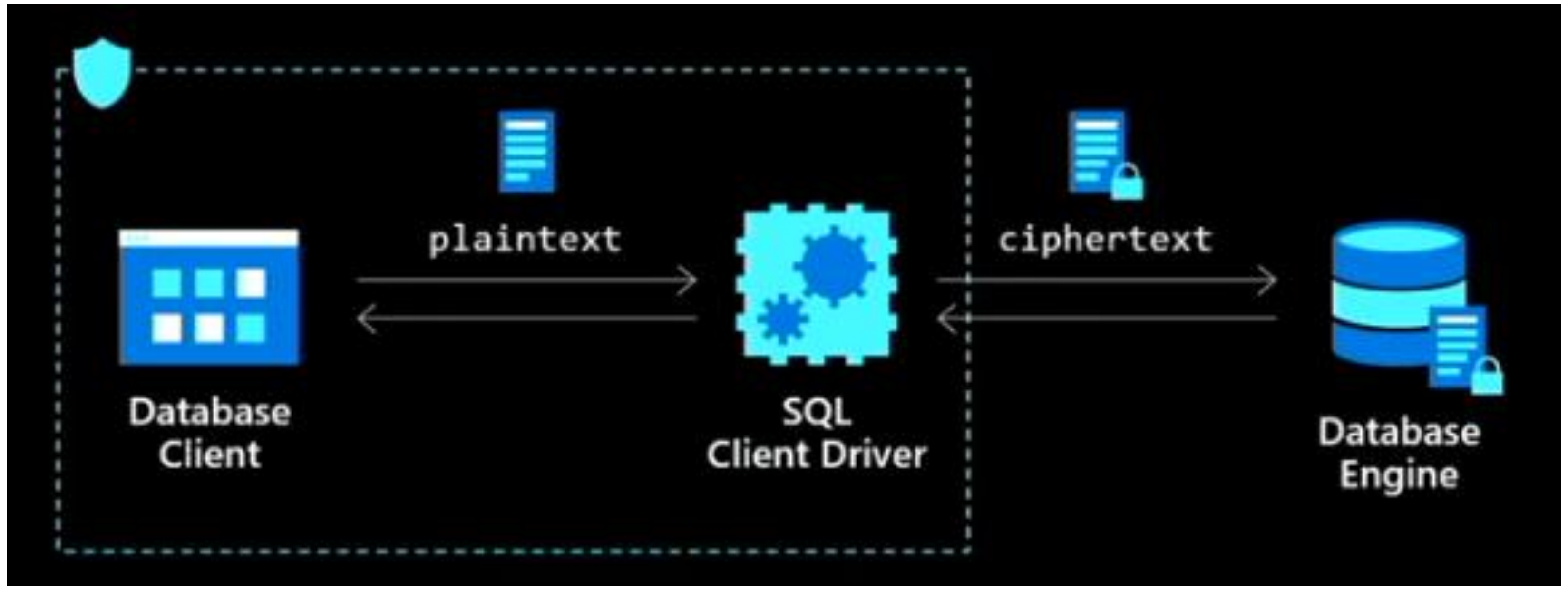
Randomized



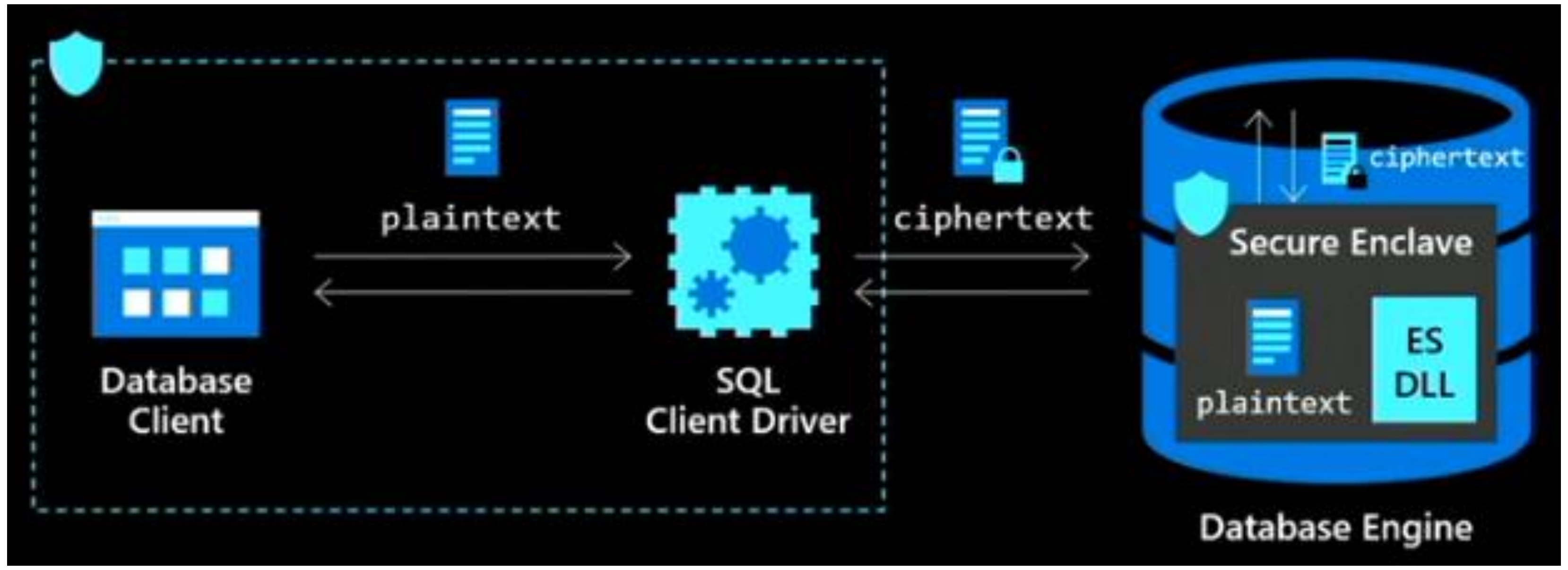
	FirstName	Gender
1	Gail	0x01E2D1CDB2DB0C5D25AB7349A4A586B55907C310803A7F0...
2	Terri	0x01A77DEBC67840757FF4F75D8053D44ECCD5AA7E235AFA8...
3	Diane	0x01BEA66C9E94D5952CBF17124CD5C02D12473A8BCA61E81...
4	Ken	0x01FC1B79A0C83334384EBBCD820E2B4FDFB9B13AD888B83...
5	Roberto	0x010C9B87ED821E0292B6CBF3AAF0DE7578792A79415319E...
6	Rob	0x0190F8ADFD36E2B97D8608F8B3F2DA102776EB85DD3CD4...
7	Jossef	0x01BA22B1D9154B4D23A85F149494951 2678E3A62237E9A8C...
8	Dylan	0x014A5FAB36036F65BAF7EE3AF3F68678DF1858A7D00BADE...

Always Encrypted with Secure Enclaves

AE without Secure Enclaves



AE with Secure Enclaves



SQL Bits – Demystifying Always Encrypted with security enclaves

Computation Over Encrypted Columns

Randomized	No scalar operations
Deterministic	Equality queries
Randomized & enclaved-enabled keys	Range/LIKE queries, sorting

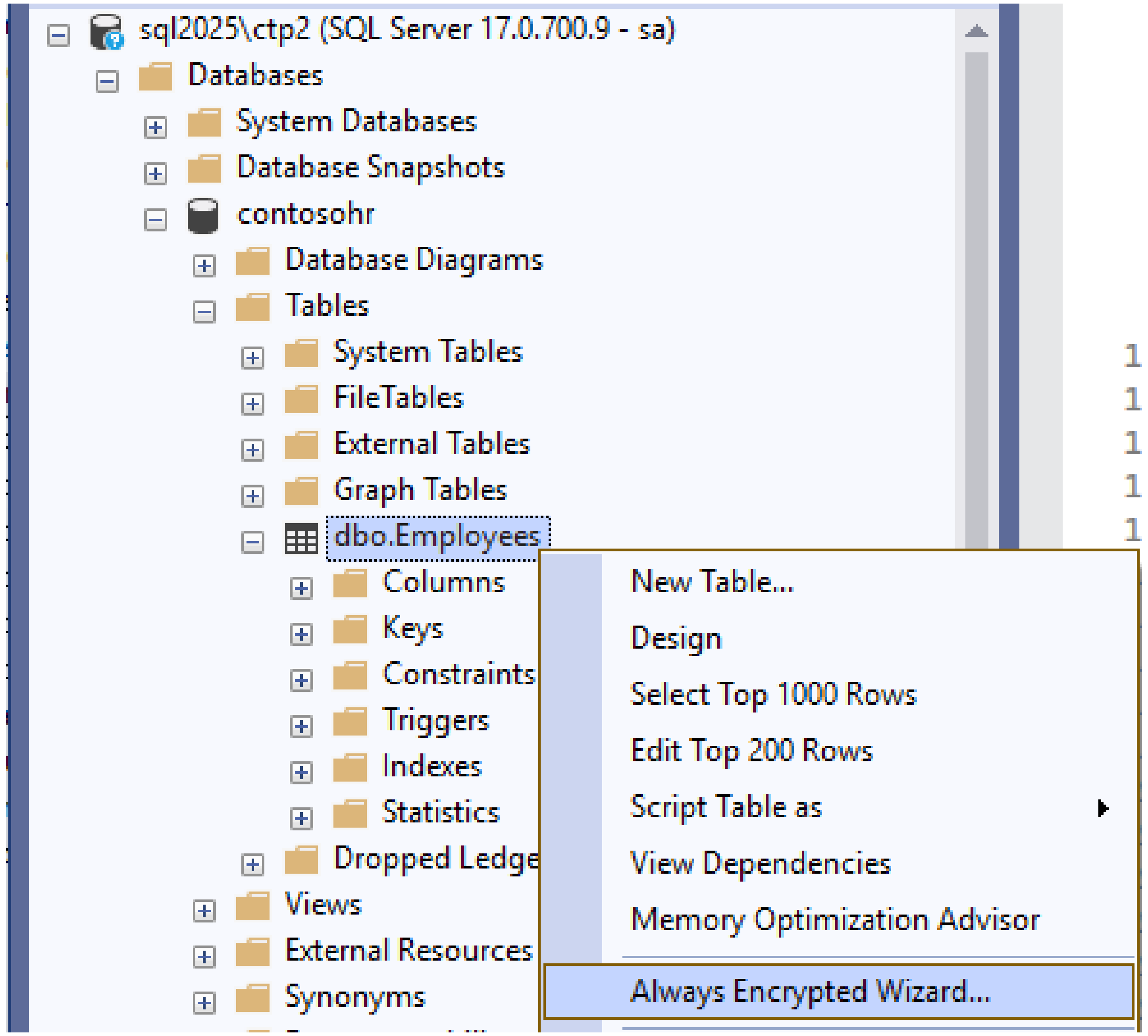
Always Encrypted - Required Code Changes

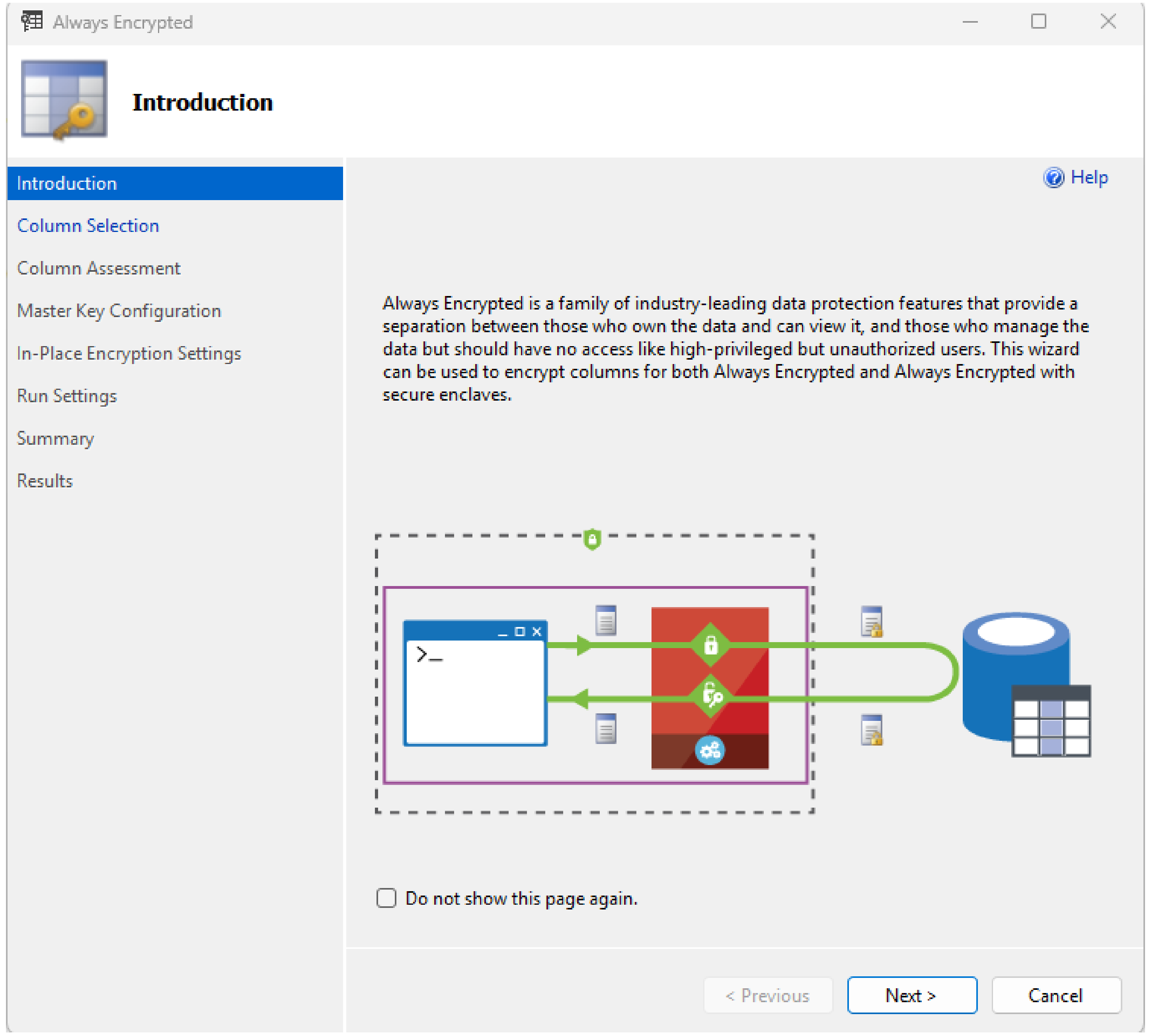
Connection String (Encrypted Setting=Enabled)]

Parametrization of Queries – No literals in filters

Explicit Data Type (Parameter Type must match encrypted column type)

Enable Always Encrypted (1 of 5)

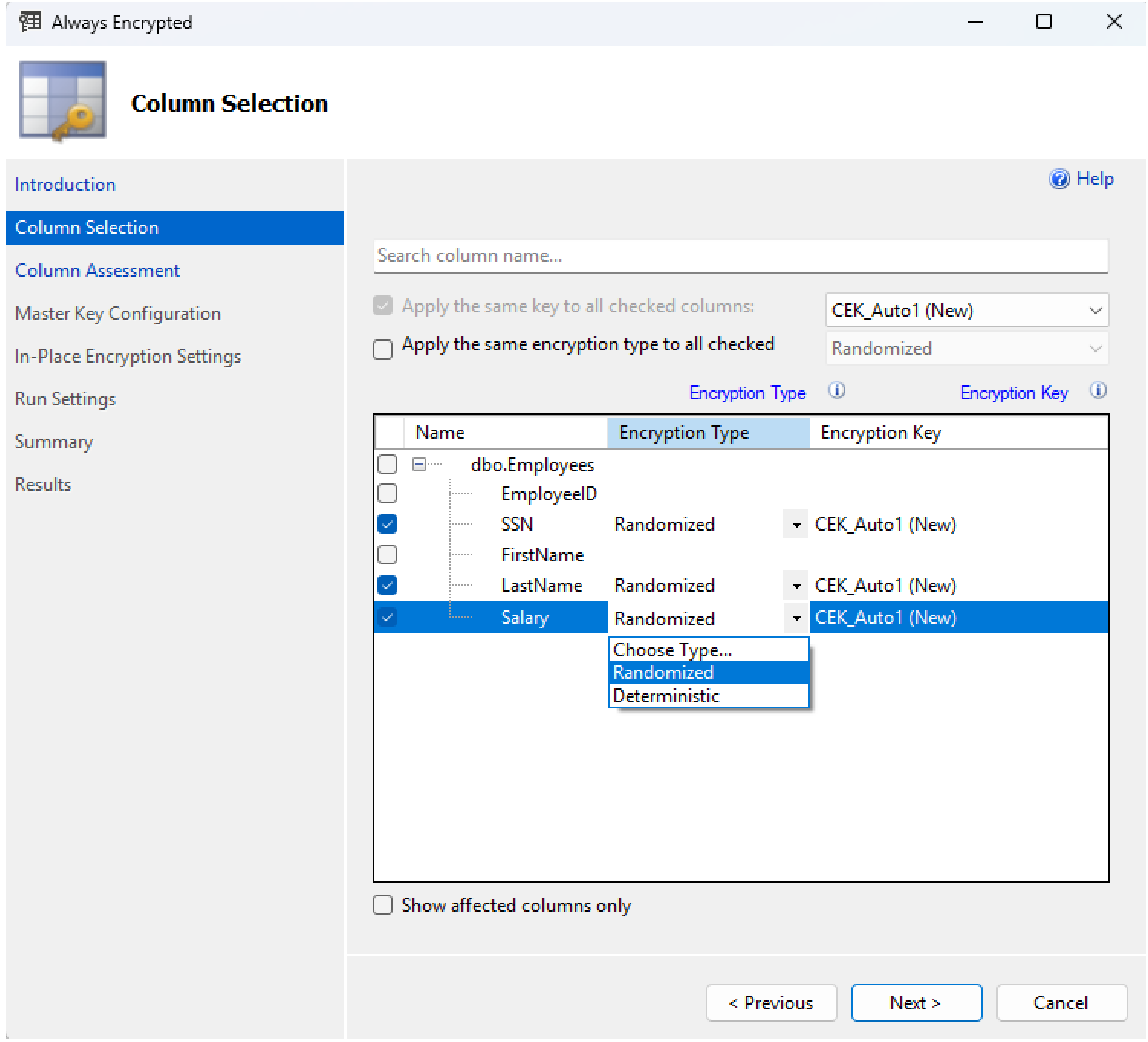




Always Encrypted is a family of industry-leading data protection features that provide a separation between those who own the data and can view it, and those who manage the data but should have no access like high-privileged but unauthorized users. This wizard can be used to encrypt columns for both Always Encrypted and Always Encrypted with secure enclaves.

☐ Do not show this page again.

Enable Always Encrypted (2 of 5)



Always Encrypted

Column Selection

Introduction
Column Selection
Column Assessment
Master Key Configuration
In-Place Encryption Settings
Run Settings
Summary
Results

Help

Search column name...

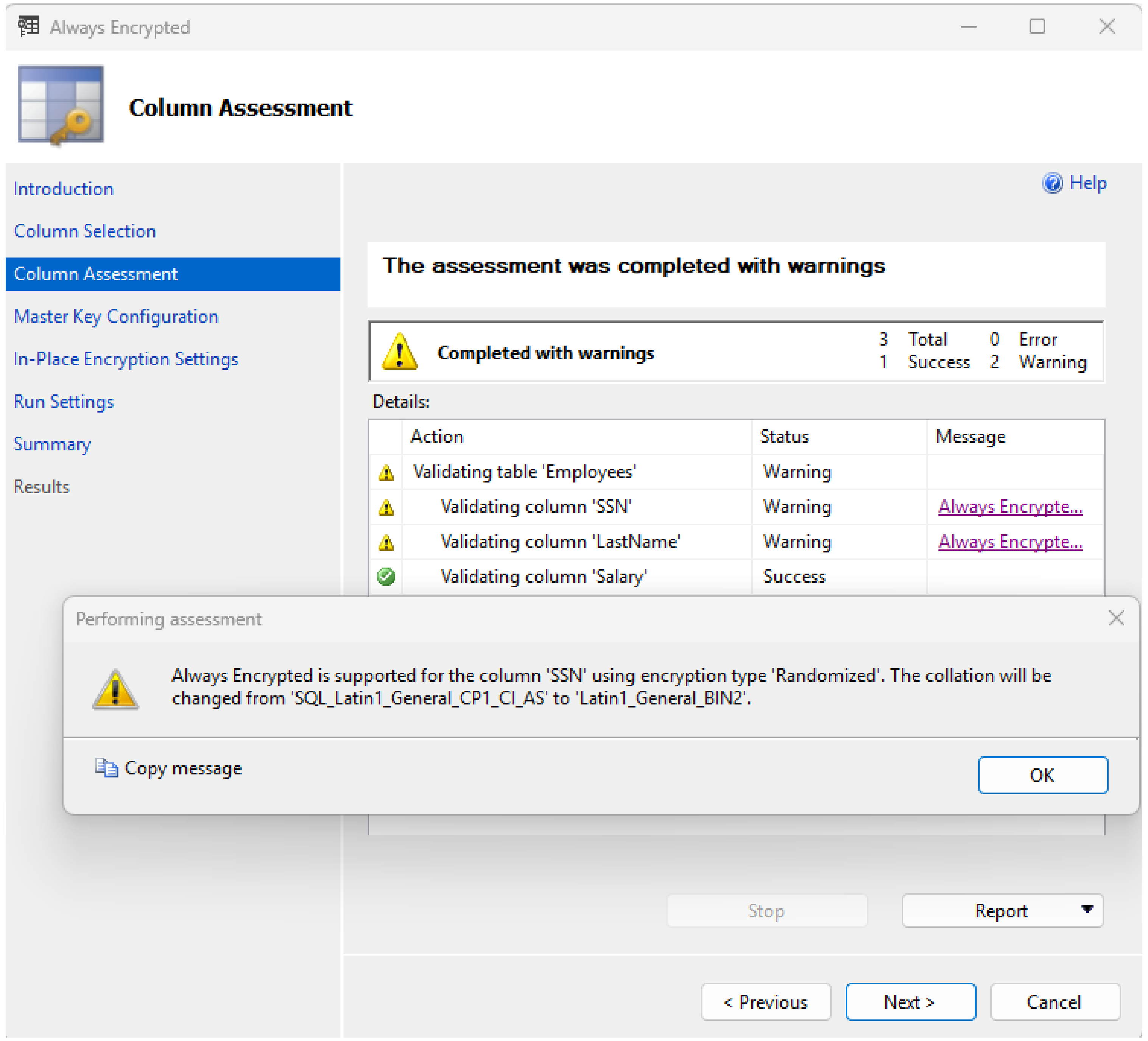
☑ Apply the same key to all checked columns: CEK_Auto1 (New)

☐ Apply the same encryption type to all checked: Randomized

Encryption Type ⓘ Encryption Key ⓘ

	Name	Encryption Type	Encryption Key
☐	⊟ dbo.Employees		
☐	EmployeeID		
☑	SSN	Randomized	CEK_Auto1 (New)
☐	FirstName		
☑	LastName	Randomized	CEK_Auto1 (New)
☑	Salary	Randomized	CEK_Auto1 (New)

☐ Show affected columns only



Always Encrypted

Column Assessment

Introduction
Column Selection
Column Assessment
Master Key Configuration
In-Place Encryption Settings
Run Settings
Summary
Results

Help

The assessment was completed with warnings

Completed with warnings 3 Total 1 Success 0 Error 2 Warning

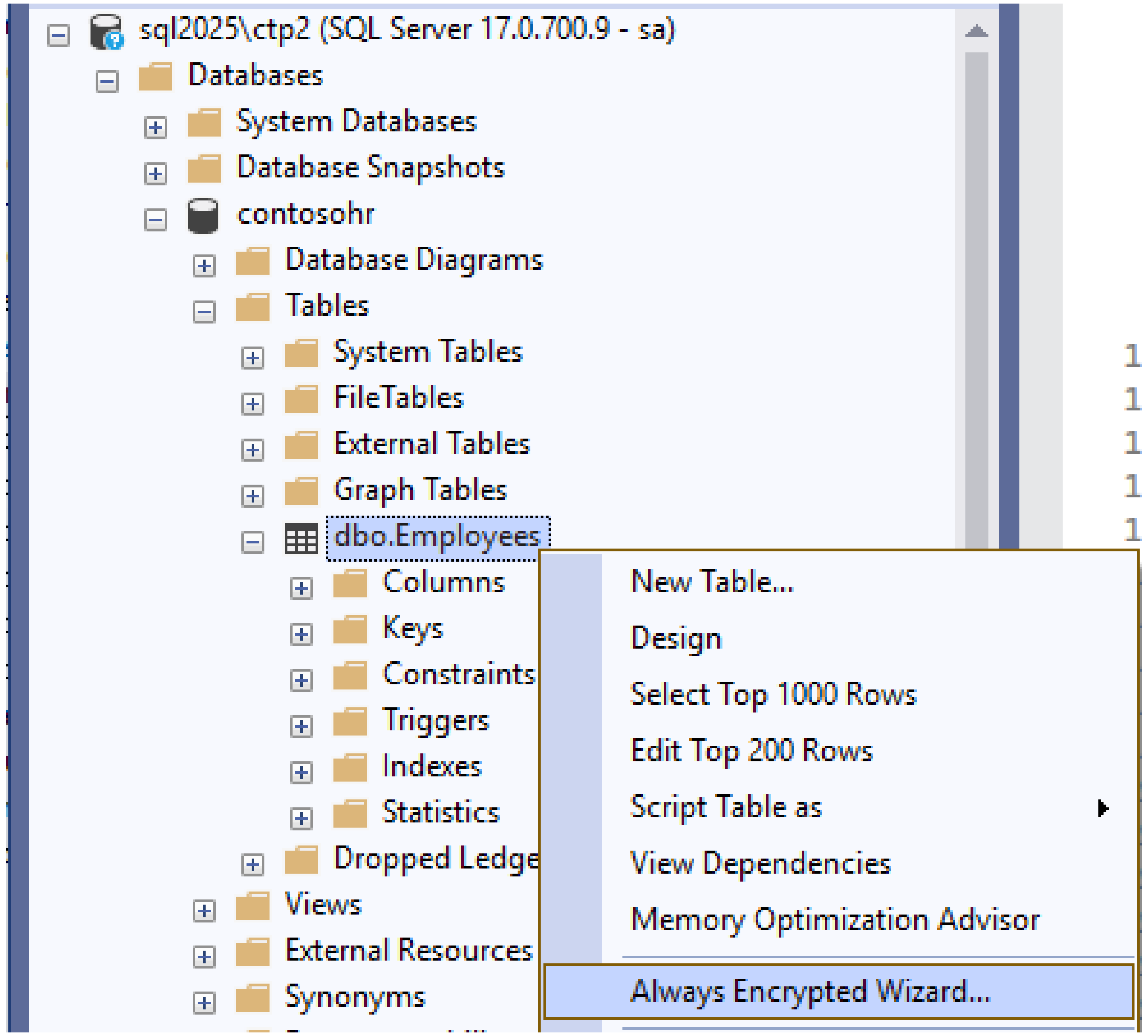
Details:

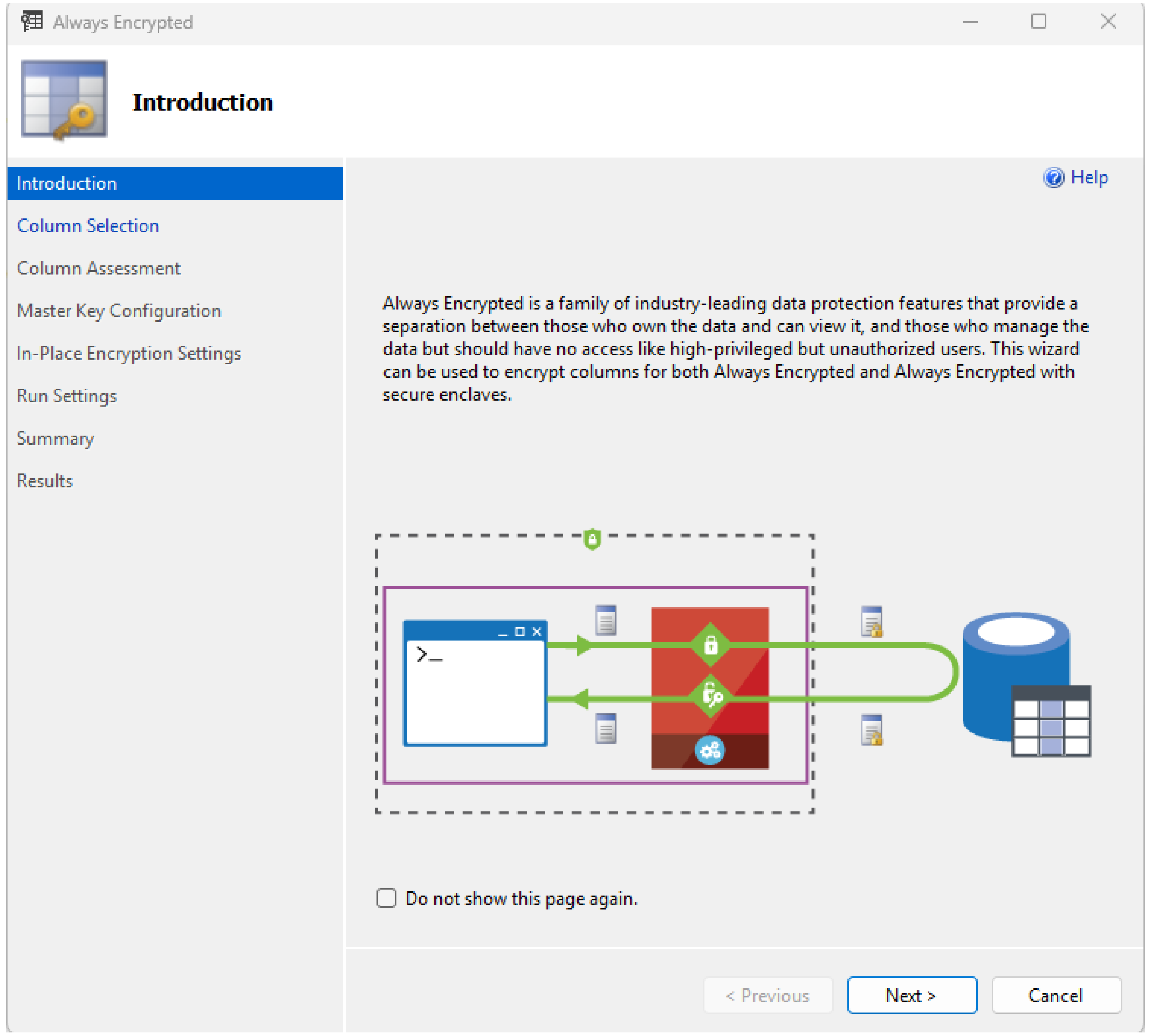
Action	Status	Message
Validating table 'Employees'	Warning	
Validating column 'SSN'	Warning	Always Encrypte...
Validating column 'LastName'	Warning	Always Encrypte...
Validating column 'Salary'	Success	

Performing assessment

Always Encrypted is supported for the column 'SSN' using encryption type 'Randomized'. The collation will be changed from 'SQL_Latin1_General_CP1_CI_AS' to 'Latin1_General_BIN2'.

Enable Always Encrypted (3 of 5)



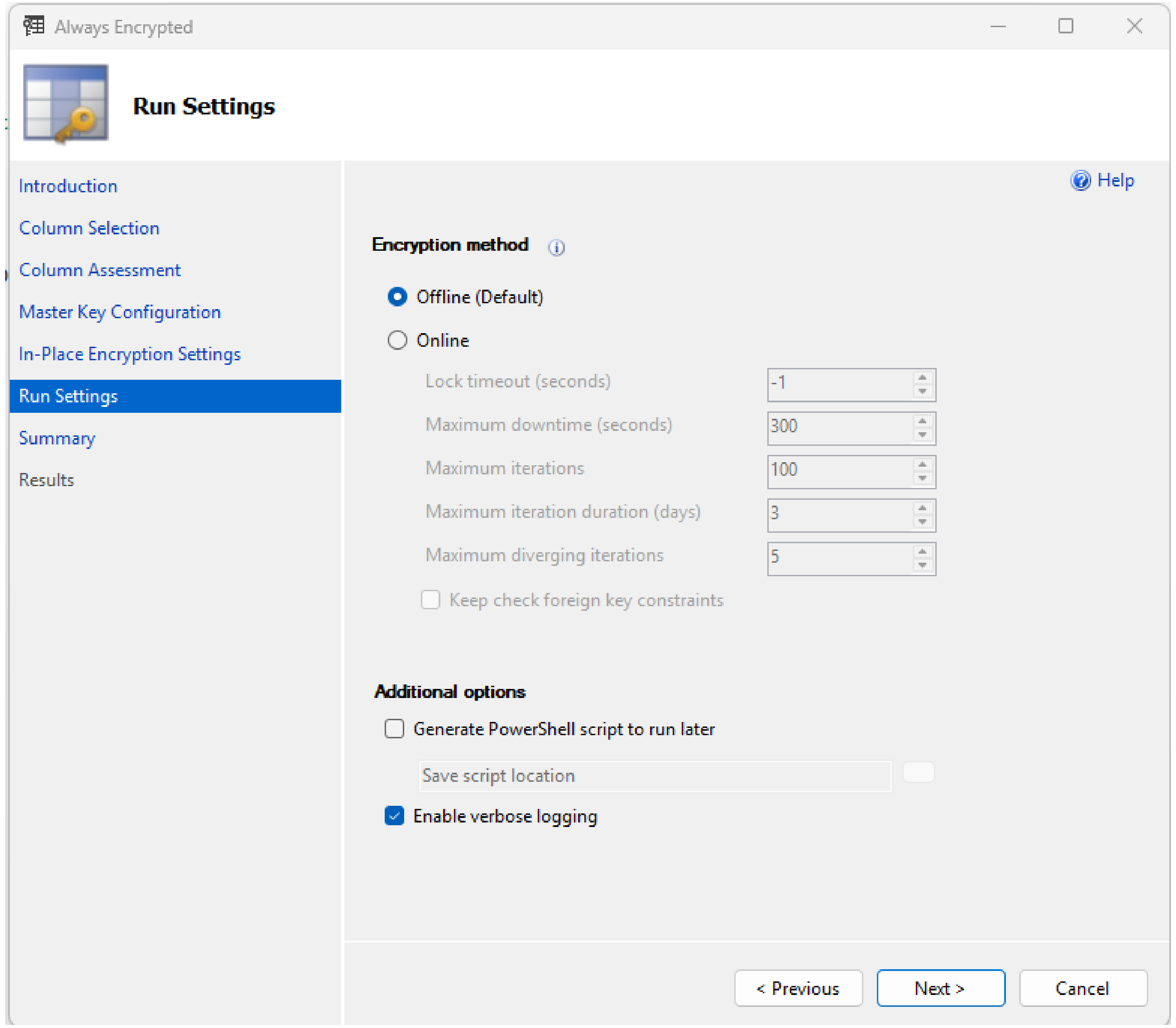


Always Encrypted is a family of industry-leading data protection features that provide a separation between those who own the data and can view it, and those who manage the data but should have no access like high-privileged but unauthorized users. This wizard can be used to encrypt columns for both Always Encrypted and Always Encrypted with secure enclaves.

☐ Do not show this page again.

Enable Always Encrypted (4 of 5)

Online Encryption Requires Secure Enclaves



Run Settings

Encryption method

● Offline (Default)

○ Online

Lock timeout (seconds): -1

Maximum downtime (seconds): 300

Maximum iterations: 100

Maximum iteration duration (days): 3

Maximum diverging iterations: 5

☐ Keep check foreign key constraints

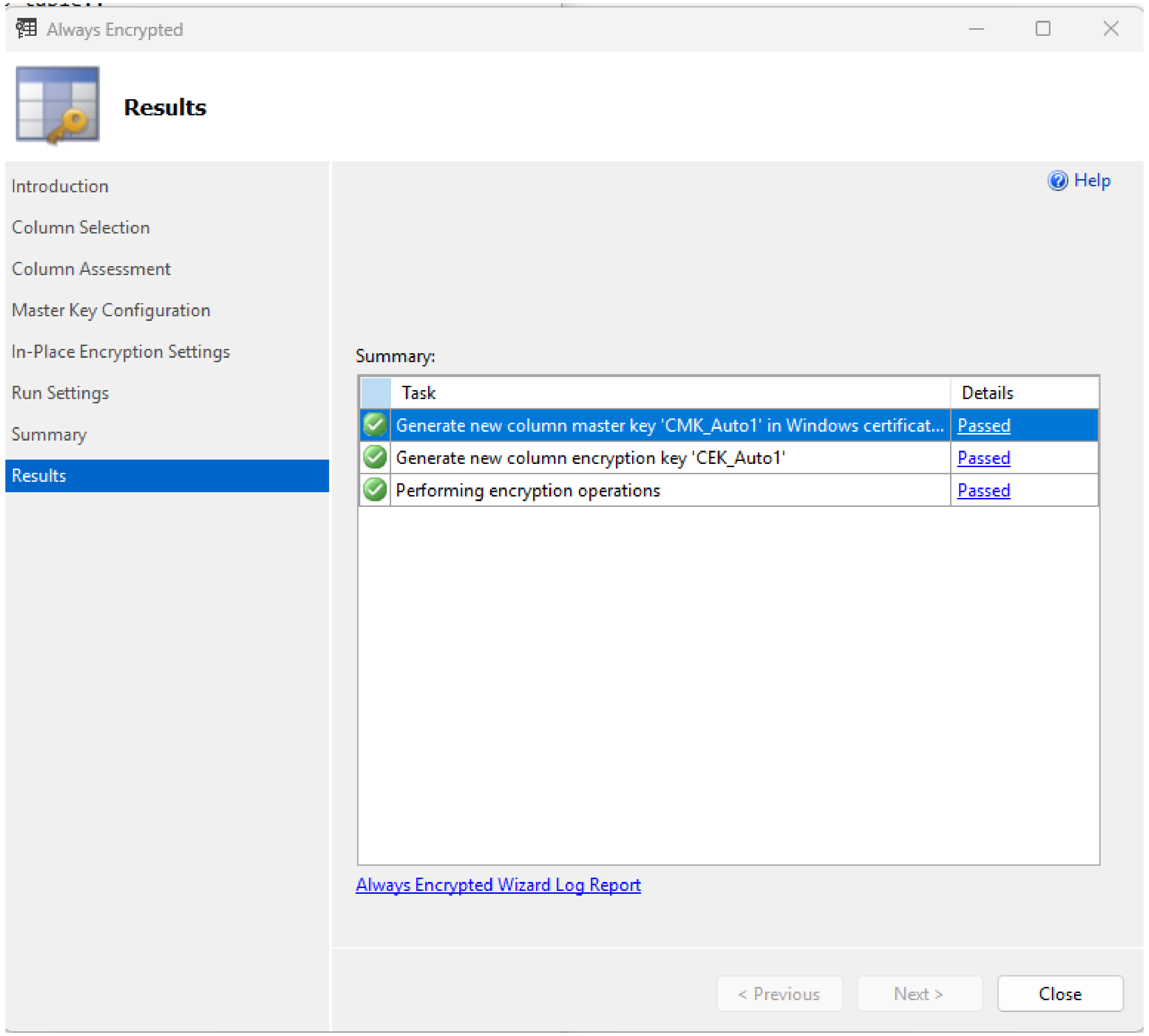
Additional options

☐ Generate PowerShell script to run later

Save script location

☑ Enable verbose logging

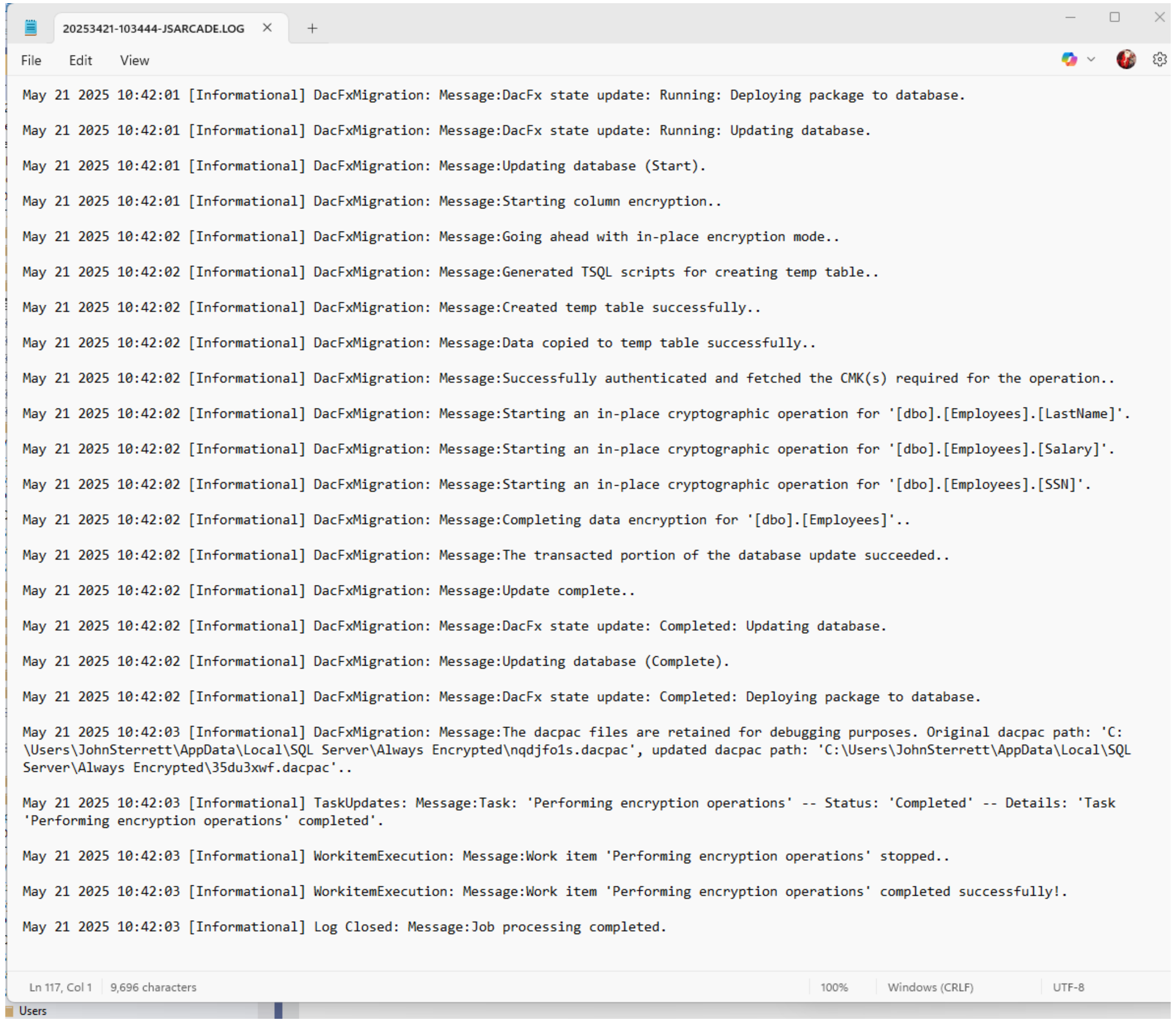
Enable Always Encrypted (5 of 5)



Summary:

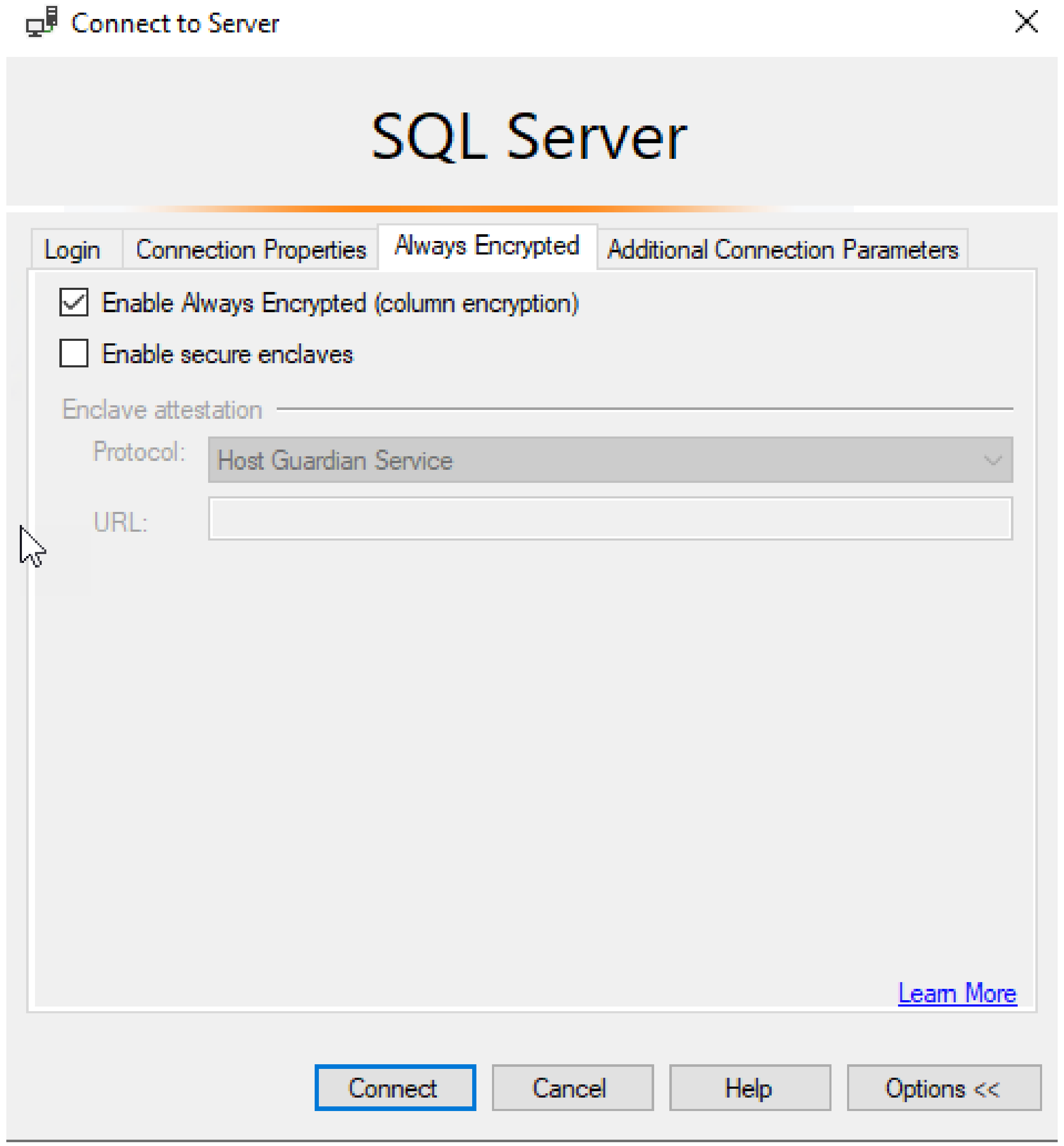
Task	Details
✓ Generate new column master key 'CMK_Auto1' in Windows certificat...	Passed
✓ Generate new column encryption key 'CEK_Auto1'	Passed
✓ Performing encryption operations	Passed

Always Encrypted Wizard Log Report



```
May 21 2025 10:42:01 [Informational] DacFxMigration: Message:DacFx state update: Running: Deploying package to database.

May 21 2025 10:42:01 [Informational] DacFxMigration: Message:DacFx state update: Running: Updating database.

May 21 2025 10:42:01 [Informational] DacFxMigration: Message:Updating database (Start).

May 21 2025 10:42:01 [Informational] DacFxMigration: Message:Starting column encryption..

May 21 2025 10:42:02 [Informational] DacFxMigration: Message:Going ahead with in-place encryption mode..

May 21 2025 10:42:02 [Informational] DacFxMigration: Message:Generated TSQL scripts for creating temp table..

May 21 2025 10:42:02 [Informational] DacFxMigration: Message:Created temp table successfully..

May 21 2025 10:42:02 [Informational] DacFxMigration: Message:Data copied to temp table successfully..

May 21 2025 10:42:02 [Informational] DacFxMigration: Message:Successfully authenticated and fetched the CMK(s) required for the operation..

May 21 2025 10:42:02 [Informational] DacFxMigration: Message:Starting an in-place cryptographic operation for '[dbo].[Employees].[LastName]'.

May 21 2025 10:42:02 [Informational] DacFxMigration: Message:Starting an in-place cryptographic operation for '[dbo].[Employees].[Salary]'.

May 21 2025 10:42:02 [Informational] DacFxMigration: Message:Starting an in-place cryptographic operation for '[dbo].[Employees].[SSN]'.

May 21 2025 10:42:02 [Informational] DacFxMigration: Message:Completing data encryption for '[dbo].[Employees]'..

May 21 2025 10:42:02 [Informational] DacFxMigration: Message:The transacted portion of the database update succeeded..

May 21 2025 10:42:02 [Informational] DacFxMigration: Message:Update complete..

May 21 2025 10:42:02 [Informational] DacFxMigration: Message:DacFx state update: Completed: Updating database.

May 21 2025 10:42:02 [Informational] DacFxMigration: Message:Updating database (Complete).

May 21 2025 10:42:02 [Informational] DacFxMigration: Message:DacFx state update: Completed: Deploying package to database.

May 21 2025 10:42:03 [Informational] DacFxMigration: Message:The dacpac files are retained for debugging purposes. Original dacpac path: 'C:
\Users\JohnSterrett\AppData\Local\SQL Server\Always Encrypted\nqdjfo1s.dacpac', updated dacpac path: 'C:\Users\JohnSterrett\AppData\Local\SQL
Server\Always Encrypted\35du3xwf.dacpac'..

May 21 2025 10:42:03 [Informational] TaskUpdates: Message:Task: 'Performing encryption operations' -- Status: 'Completed' -- Details: 'Task
'Performing encryption operations' completed'.

May 21 2025 10:42:03 [Informational] WorkitemExecution: Message:Work item 'Performing encryption operations' stopped..

May 21 2025 10:42:03 [Informational] WorkitemExecution: Message:Work item 'Performing encryption operations' completed successfully!.

May 21 2025 10:42:03 [Informational] Log Closed: Message:Job processing completed.
```

Connecting with SSMS 21



Connect to Server

SQL Server

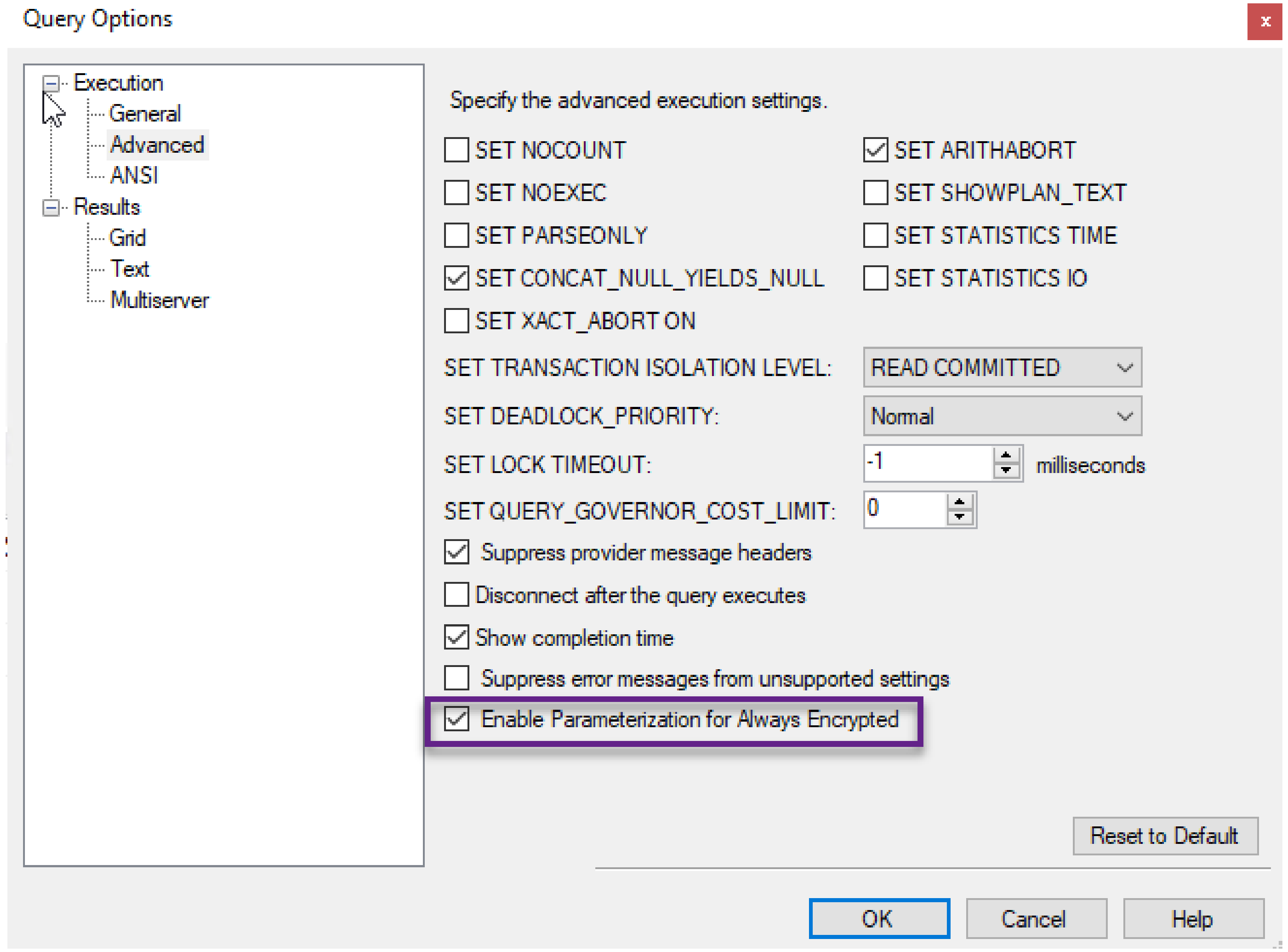
☑ Enable Always Encrypted (column encryption)

☐ Enable secure enclaves

Enclave attestation

Protocol: Host Guardian Service

URL:



Query Options

Specify the advanced execution settings.

☐ SET NOCOUNT

☐ SET NOEXEC

☐ SET PARSEONLY

☑ SET CONCAT_NULL_YIELDS_NULL

☐ SET XACT_ABORT ON

☑ SET ARITHABORT

☐ SET SHOWPLAN_TEXT

☐ SET STATISTICS TIME

☐ SET STATISTICS IO

SET TRANSACTION ISOLATION LEVEL: READ COMMITTED

SET DEADLOCK_PRIORITY: Normal

SET LOCK TIMEOUT: -1 milliseconds

SET QUERY_GOVERNOR_COST_LIMIT: 0

☑ Suppress provider message headers

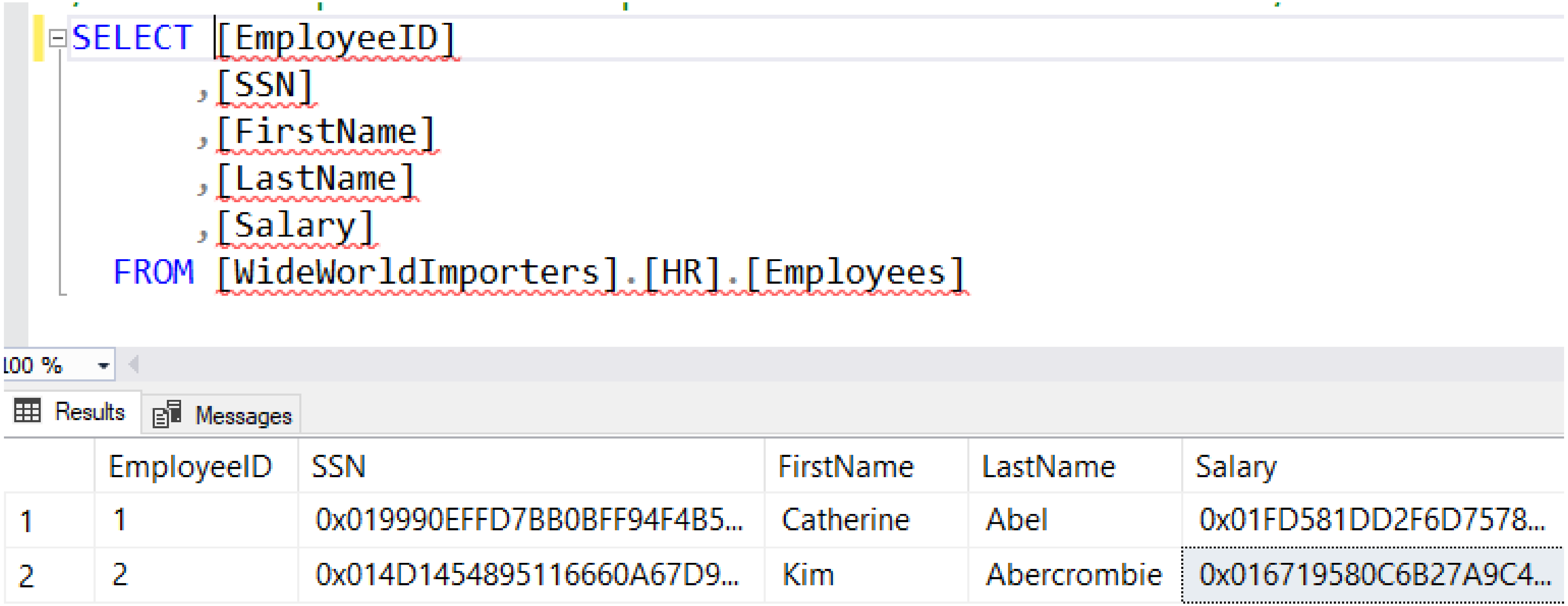
☐ Disconnect after the query executes

☑ Show completion time

☐ Suppress error messages from unsupported settings

☑ Enable Parameterization for Always Encrypted

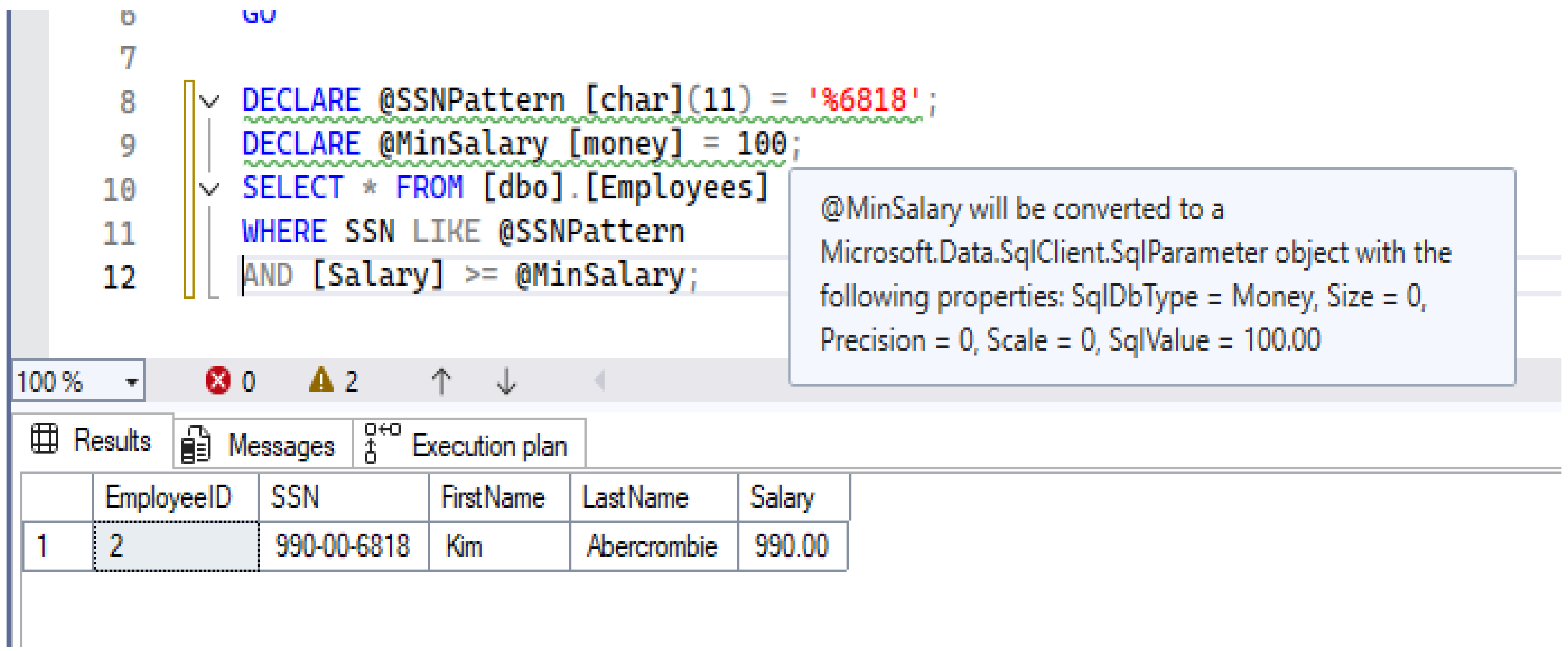
AE - Columns Accessed Without Keys



```
SELECT [EmployeeID]
      ,[SSN]
      ,[FirstName]
      ,[LastName]
      ,[Salary]
  FROM [WideWorldImporters].[HR].[Employees]
```

	EmployeeID	SSN	FirstName	LastName	Salary
1	1	0x019990EFFD7BB0BFF94F4B5...	Catherine	Abel	0x01FD581DD2F6D7578...
2	2	0x014D1454895116660A67D9...	Kim	Abercrombie	0x016719580C6B27A9C4...

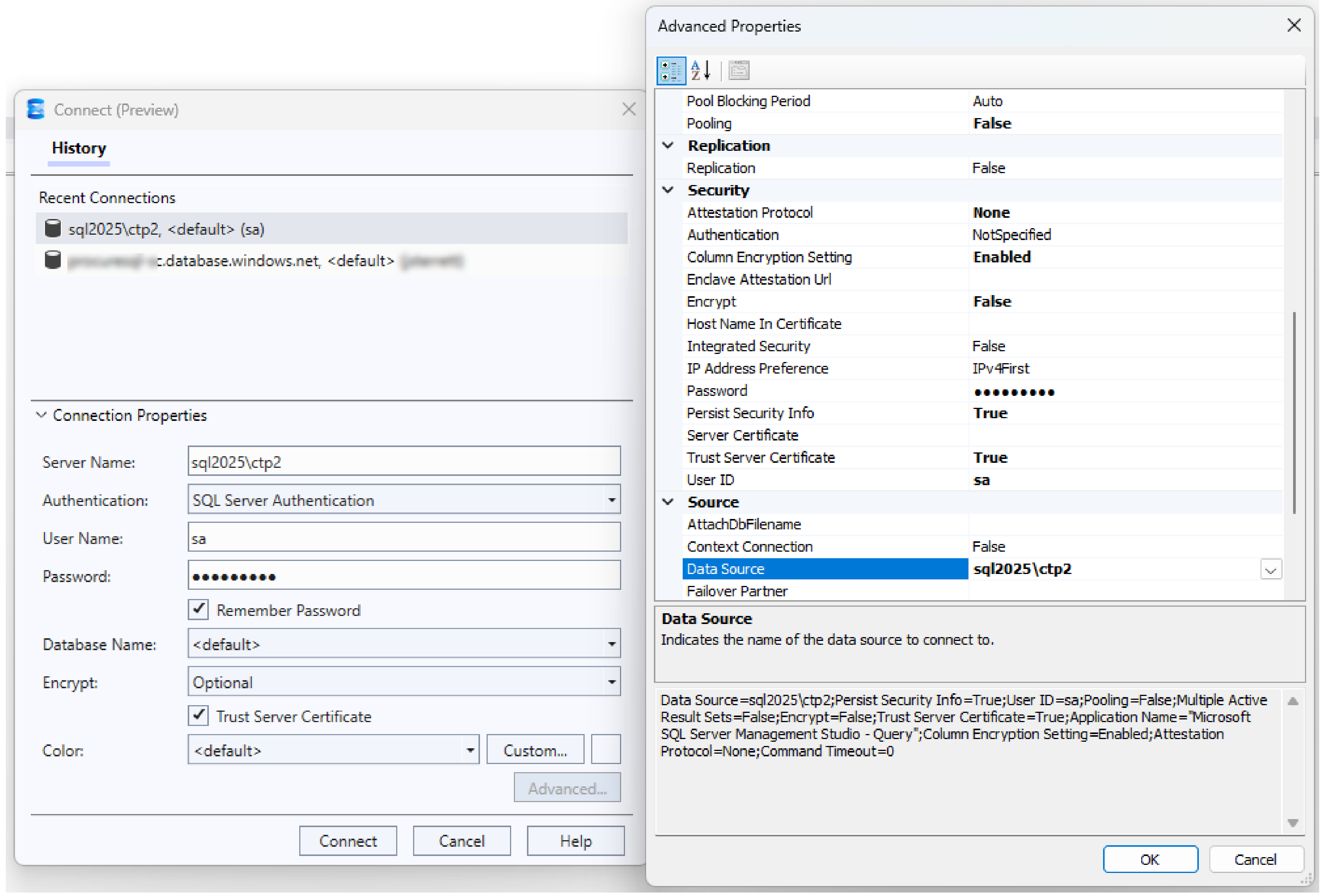
AE – Columns Decrypted With Key



```
GO

DECLARE @SSNPattern [char](11) = '%6818';
DECLARE @MinSalary [money] = 100;
SELECT * FROM [dbo].[Employees]
WHERE SSN LIKE @SSNPattern
AND [Salary] >= @MinSalary;
```

@MinSalary will be converted to a Microsoft.Data.SqlClient.SqlParameter object with the following properties: SqlDbType = Money, Size = 0, Precision = 0, Scale = 0, SqlValue = 100.00

	EmployeeID	SSN	FirstName	LastName	Salary
1	2	990-00-6818	Kim	Abercrombie	990.00



Recent Connections

- sql2025\ctp2, <default> (sa)
- [redacted]c.database.windows.net, <default> [redacted]

Connection Properties

Server Name: sql2025\ctp2

Authentication: SQL Server Authentication

User Name: sa

Password: ●●●●●●●●●

☑ Remember Password

Database Name: <default>

Encrypt: Optional

☑ Trust Server Certificate

Color: <default>

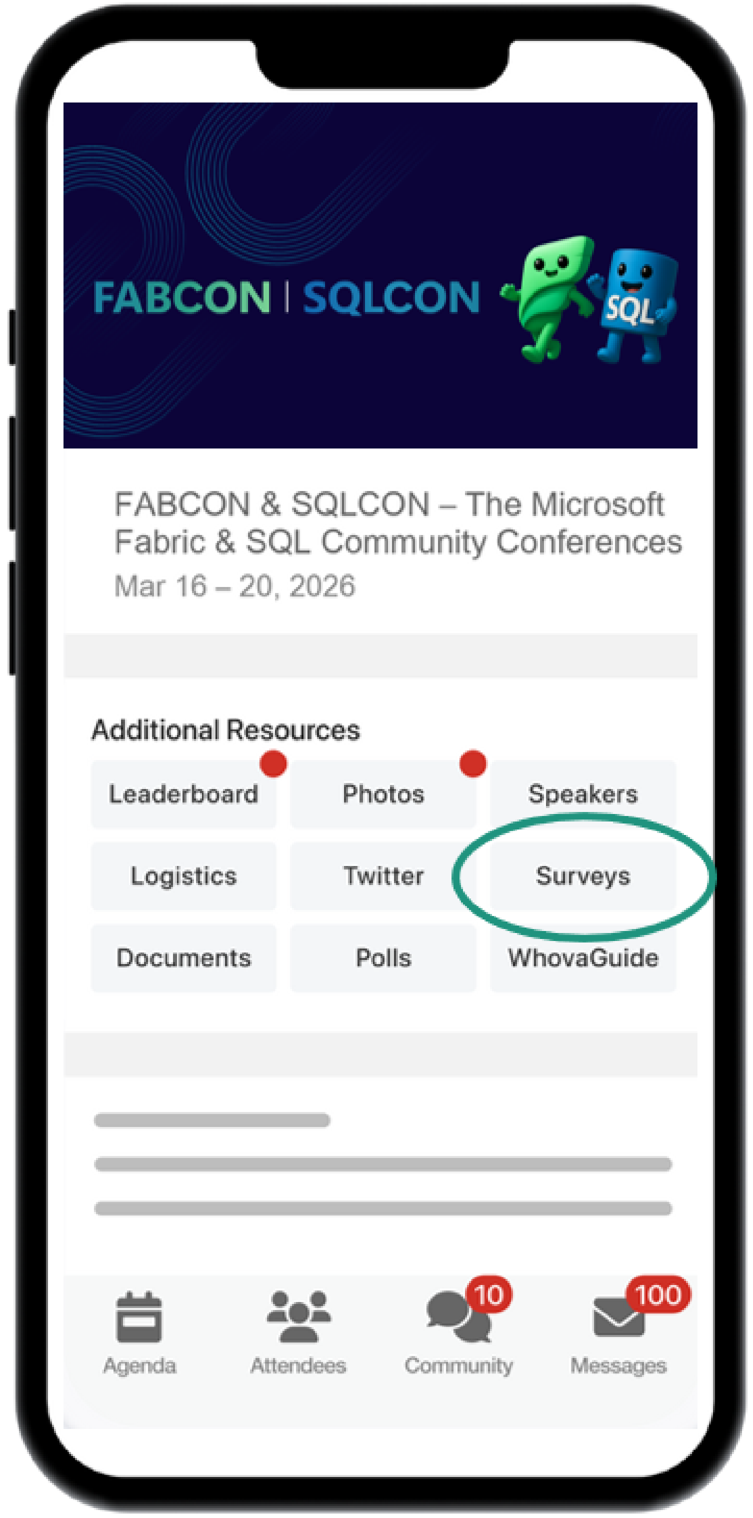
Pool Blocking Period	Auto
Pooling	**False**
Replication	
Replication	False
Security	
Attestation Protocol	**None**
Authentication	NotSpecified
Column Encryption Setting	**Enabled**
Enclave Attestation Url	
Encrypt	**False**
Host Name In Certificate	
Integrated Security	False
IP Address Preference	IPv4First
Password	●●●●●●●●●
Persist Security Info	**True**
Server Certificate	
Trust Server Certificate	**True**
User ID	**sa**
Source	
AttachDbFilename	
Context Connection	False
Data Source	**sql2025\ctp2**
Failover Partner	

Data Source
Indicates the name of the data source to connect to.

```
Data Source=sql2025\ctp2;Persist Security Info=True;User ID=sa;Pooling=False;Multiple Active Result Sets=False;Encrypt=False;Trust Server Certificate=True;Application Name="Microsoft SQL Server Management Studio - Query";Column Encryption Setting=Enabled;Attestation Protocol=None;Command Timeout=0
```

How was the session?

Complete Session Surveys in Whova for your chance to WIN PRIZES!



FABCON | SQLCON

FABCON & SQLCON – The Microsoft Fabric & SQL Community Conferences

Mar 16 – 20, 2026

Additional Resources

Get Two Fabric Certifications for FREE

Attendees of FABCON can take the Fabric Analytics Engineer or Fabric Data Engineer exam for free. Be part of the 2 fastest growing role-based certifications in Microsoft history.

Request your voucher by March 23, 2026.

https://aka.ms/fabcon/cert100

Questions?

Feedback for slides

john@procuresql.com

procuresql.com

https://www.linkedin.com/in/johnsterrett