Exploring the SQL Security Landscape

Vulnerability Assessment

What's in this Presentation?

- Vulnerability Assessment
  - New tool in SSMS 17.4
- SQL Data Discovery and Classification
  - New tool in SSMS 17.5
  - Data Masking
  - Row Level Security

What this presentation is not.

- Not an encyclopedic definition of all security features
- Not prescriptive

What is Security?

- **Security**, in (IT), is the defense of digital information and IT assets against internal and external, malicious and accidental threats.
  - Right Data
  - Right Time
  - Right Audience

Right Data

- Data modification is done according to rules.
- Who can create records, how, with what controls
- What are the controls around creation, change, and removal of data
- Permissions applied to Data Modification Language (DML)
- Applications to implement logic

Logic: Application vs Data

- Where is the "right" place for security logic?
- Are DDL constraints like Referential Integrity, Triggers, Datatypes, Unique Constraints and Check Constraints a type of security?

Right Time

- Data is available when needed
- Availability includes performance
- Control Data Definition and Administrative functions to preserve availability

Right Audience

- Who can use what data
- In what context
- Auditing
  - Change tracking
  - Access tracking

SQL Vulnerability Assessment

- What is it?
  - Service in SQL Server Management Studio (SSMS) v17.4 or later
- Where Can I use it?
  - Supported targets are SQL 2012 and later
- Database or Server level?
  - Either. Point at a system database to trigger the server level
- Automation ?

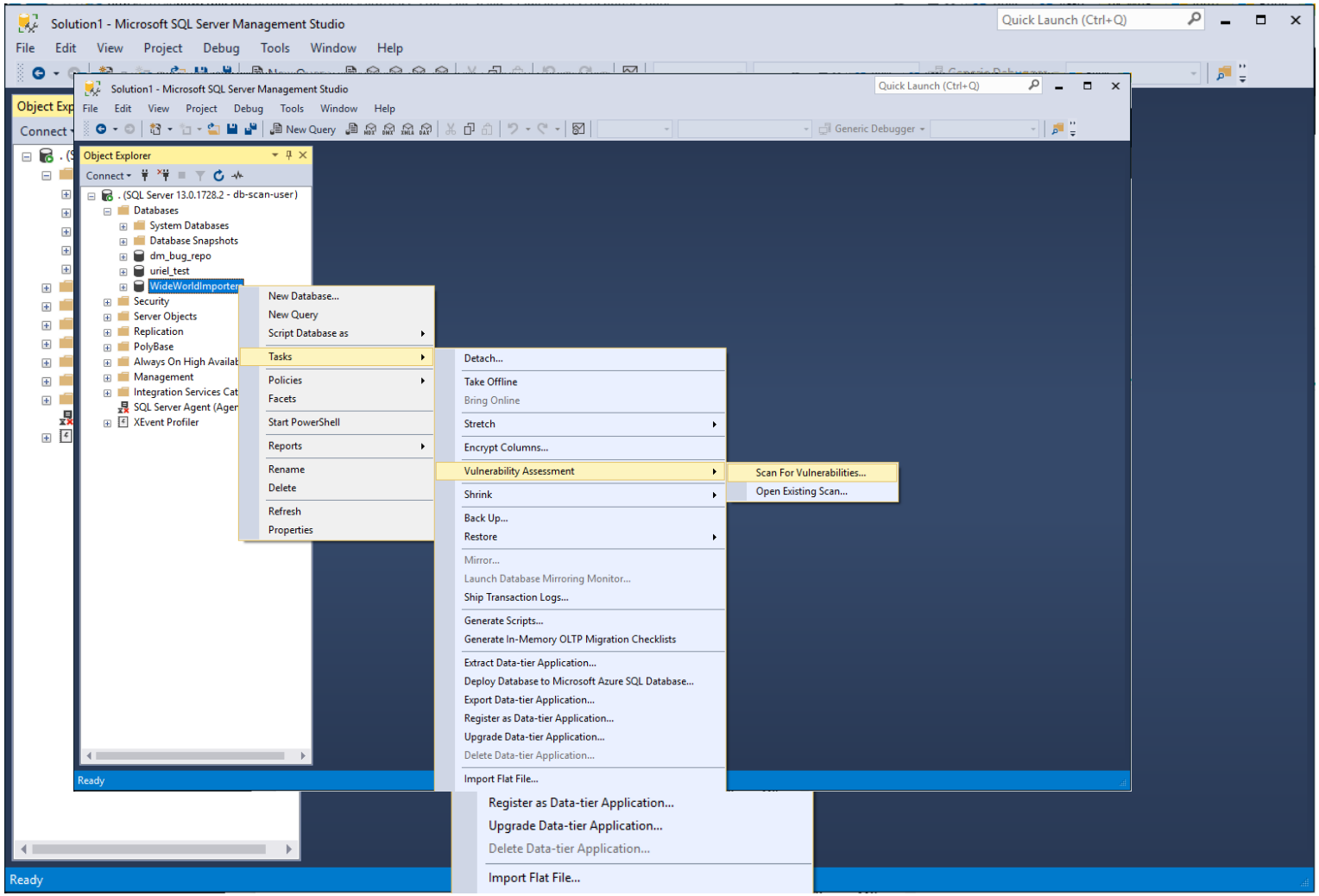
Vulnerability Assessment features

- What is it? a scan directly on your database
  - Meet compliance requirements that require database scan reports.
  - Meet data privacy standards.
  - Monitor a dynamic database environment where changes are difficult to track.

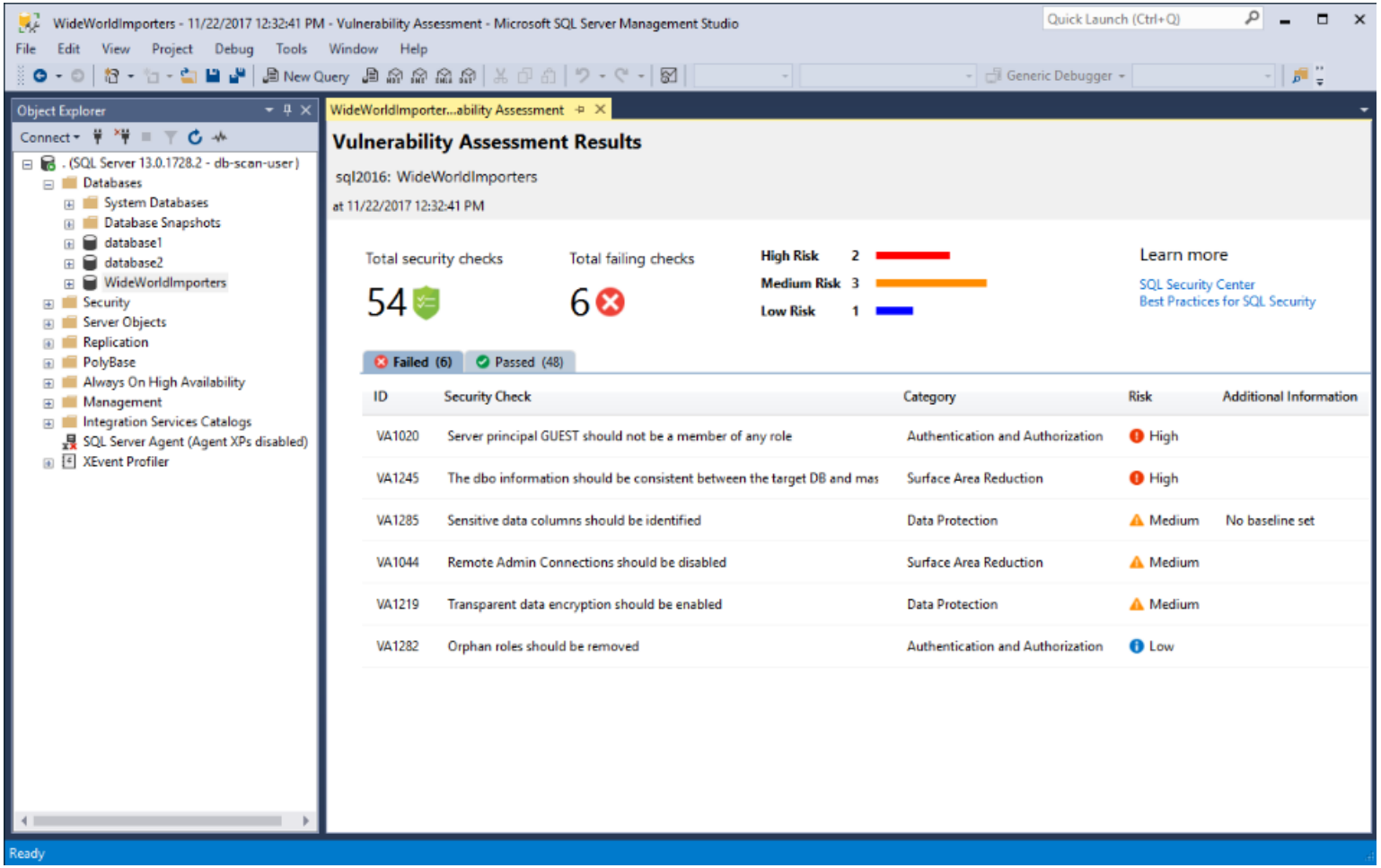
Vulnerability Assessment features

- What is it? a scan directly on your database
  - Meet compliance requirements that require database scan reports.
  - Meet data privacy standards.
  - Monitor a dynamic database environment where changes are difficult to track.

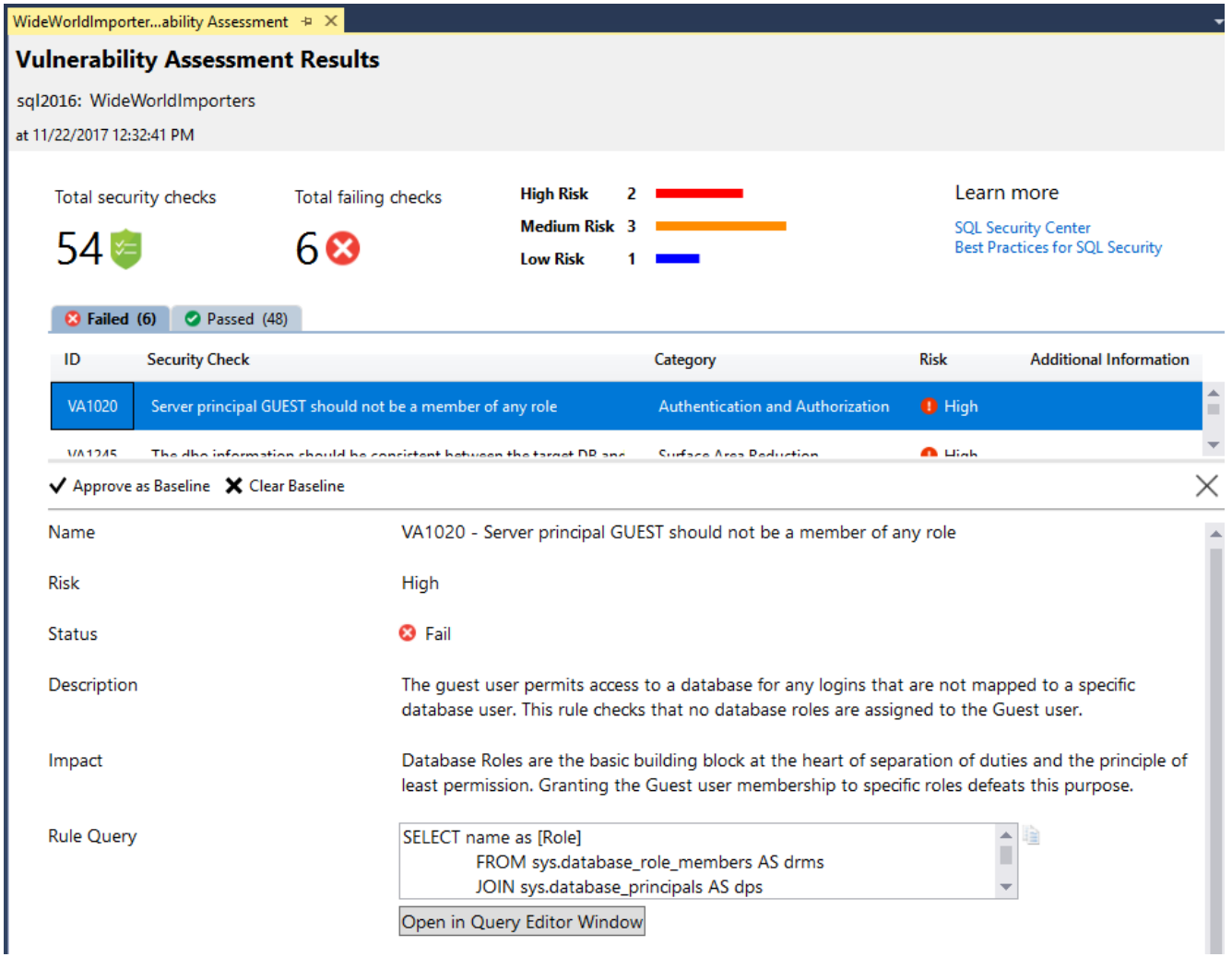
SQL Vulnerability Assessment



SQL Vulnerability Assessment



SQL Vulnerability Assessment



SQL Vulnerability Assessment

Risk	High
Status	Fail
Description	The guest user permits access to a database for any logins that are not mapped to a specific database user. This rule checks that no database roles are assigned to the Guest user.
Impact	Database Roles are the basic building block at the heart of separation of duties and the principle of least permission. Granting the Guest user membership to specific roles defeats this purpose.
Rule Query	SELECT name as [Role] FROM sys.database_role_members AS drms JOIN sys.database_principals AS dps Open in Query Editor Window
Microsoft Recommendation	Empty set

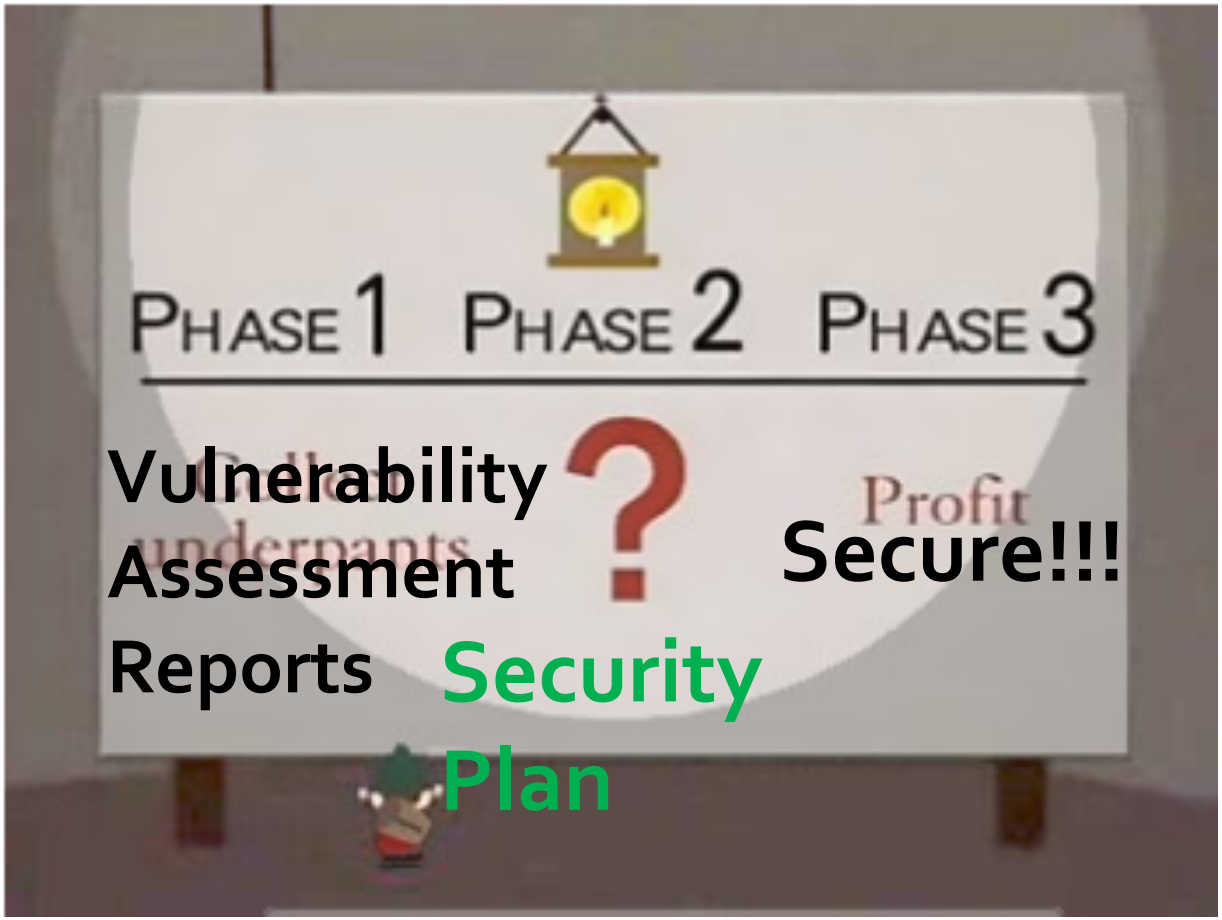
SQL Vulnerability Assessment

- Demo Time!

SQL Vulnerability Assessment

- Can we Automate it?
  - Powershell?
- Can we Customize the assessment?
  - Not yet, but on the roadmap.

SQL Vulnerability Assessment



SQL Vulnerability Assessment

VA1287 - Sensitive data columns should be classified

Failed (3) Passed (51)

ID	Security Check	Category	Risk	Additional Information
VA1287	Sensitive data columns should be classified	Data Protection	Medium	No baseline set

SQL Data Discovery & Classification

Actual Result

In Baseline	Schema	Table	Column	Information Type	Recommended Label
✗	dbo	Beneficiary	FirstName	Name	Confidential - GDPR
✗	dbo	Beneficiary	LastName	Name	Confidential - GDPR
✗	dbo	Employee	FirstName	Name	Confidential - GDPR
✗	dbo	Employee	LastName	Name	Confidential - GDPR

SQL Vulnerability Assessment

SQL Data Discovery & Classification

Dynamic Data Masking

limits sensitive data exposure by dynamically masking it to non-privileged users when data is returned from the server to the client

Row Level Security

restrict access to data rows by creating a security policy based on characteristics of the user executing a query

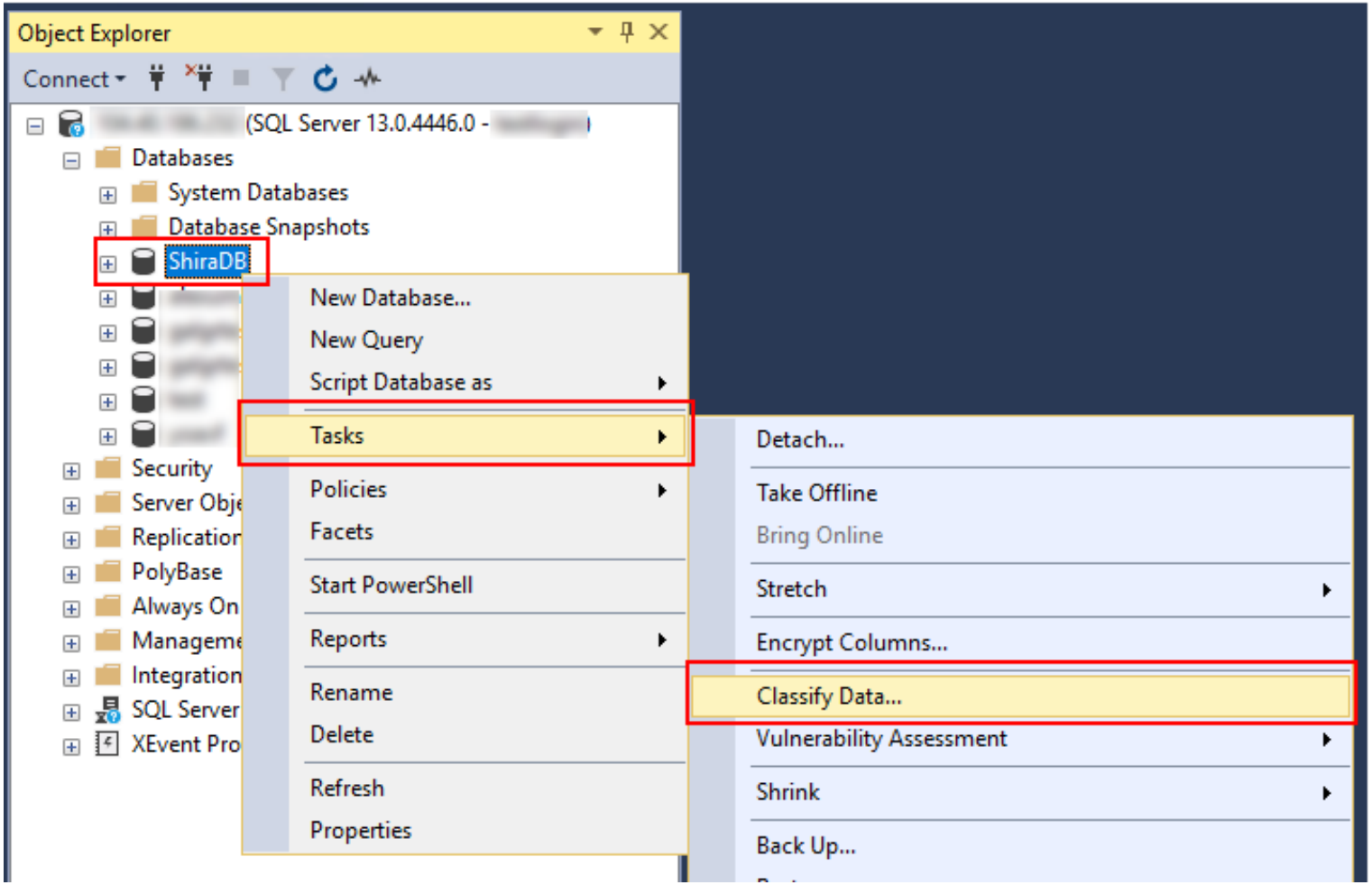
Always Encrypted

keeps sensitive data columns encrypted on the server side

SQL Data Discovery and Classification

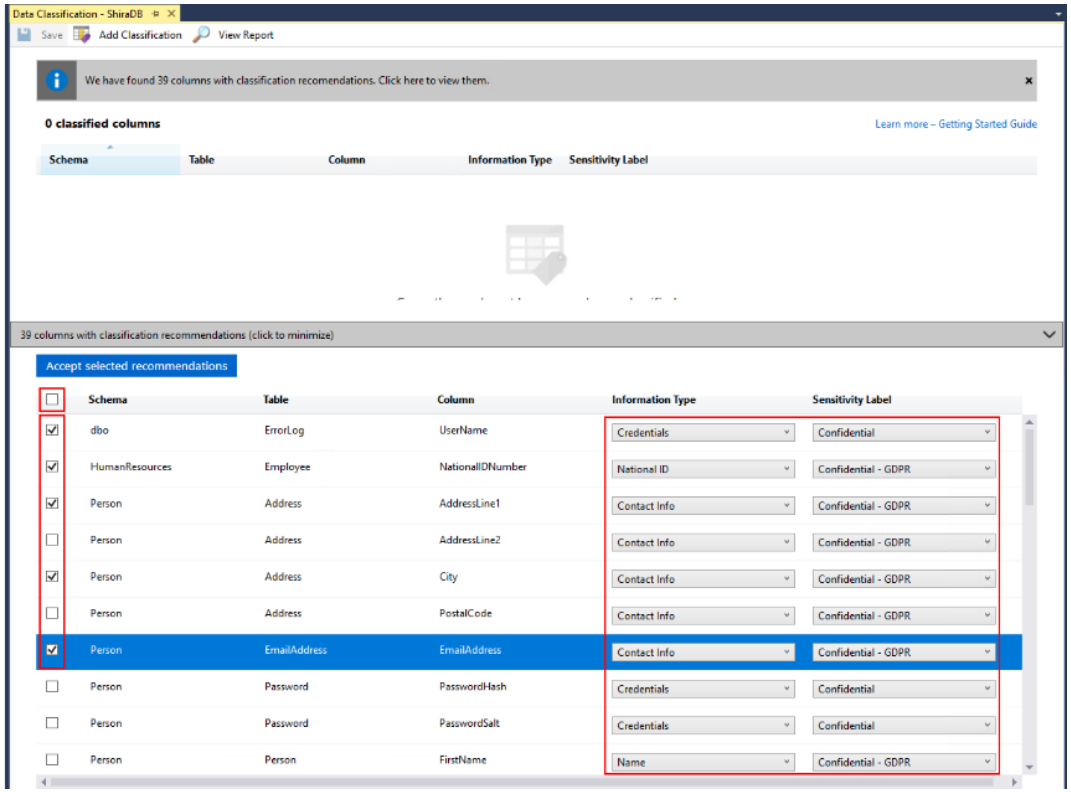
- Data Discovery & Classification is **supported for SQL Server 2008 and later**
- **"Protect the data, not just the database"**
- **Discovery & recommendations** – The classification engine scans your database and identifies columns containing potentially sensitive data
- **Labeling** – Sensitivity classification labels can be persistently tagged on columns
- **Visibility** - The database classification state can be viewed in a detailed report

SQL Data Discovery and Classification



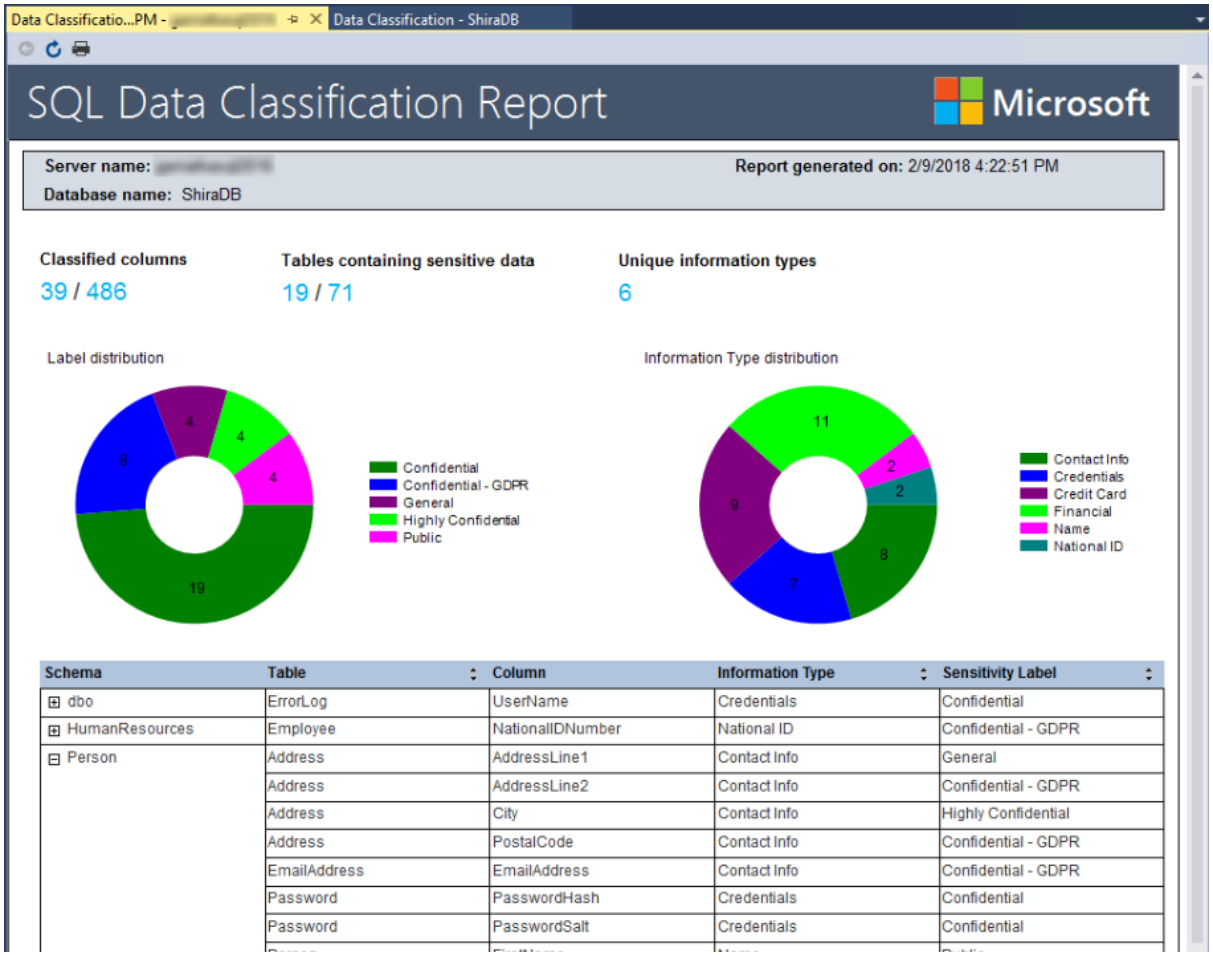
SQL Data Discovery and Classification

- Automatic and manual classification



SQL Data Discovery and Classification

- Pretty reports



SQL Data Discovery and Classification

- Demo Time!

DDM Dynamic Data Masking

- Security Feature: prevents data exfiltration by obfuscating data.
  - Credit cards xxxx-xxxxxx-x6815
  - Email brXXX.XXXX@XXX.COM
  - SIN# XXX-XXX-X12

DDM Dynamic Data Masking

- Meta data operation. Data remains unchanged, masking is applied when queried.

```
ALTER TABLE HR.Employee
ALTER COLUMN SALARY decimal(12,2) MASKED WITH
(FUNCTION = 'default()');
```

DDM Dynamic Data Masking

- **Default: (varies by data type):**
  - binary, varbinary or image: “0“
  - date and time data types: “01.01.1900 00:00:00.0000000”
  - Numeric data types: “0”
  - Strings: “XXXX”
- **Email:** first char plus mask eg **“**aXXX@XXXX.com”
- **Random** replaces the original value with a random value within the range specified in that function.
- **Custom** eg first and last letters and padding ie prefix, [padding value], suffix

DDM Dynamic Data Masking

- Random masking function

  Account_Number bigint MASKED WITH (FUNCTION = 'random([start range], [end range])')

  - Mask may accidentally be the correct number. Pick appropriate ranges.

```
ALTER TABLE JuniorHighSchool.Student
ALTER COLUMN Age integer MASKED WITH (FUNCTION = 'random(12, 14')
```

DDM Dynamic Data Masking

- **Who**
  - **Db_owner role members always see unmasked data by default**
  - **All other users see masked data.**

DDM Dynamic Data Masking

- **Who**
  - **Grant/Revoke MASK/UNMASK to users/groups**
  - **MASK/UNMASK is database level permission**
  - **eg Social committee needs to know birthdays to plan birthday parties, but granting unmask to the HR database will also show SIN# and Salary….**

DDM Dynamic Data Masking

- **Behavior**
  - **Data remains masked regardless of mechanism to select, even select into another table**

```
Select fname, lname, Salary into MyTable
from HR.Employee
```

DDM Dynamic Data Masking

- Finding masked Columns
- sys.masked_columns

```
SELECT c.name, tbl.name as table_name, c.is_masked,
c.masking_function
FROM sys.masked_columns AS c
JOIN sys.tables AS tbl
  ON c.[object_id] = tbl.[object_id]
WHERE is_masked = 1;
```

DDM Dynamic Data Masking

- **Bypassing masking using inference or brute-force techniques**

```
SELECT ID, Name, Salary
FROM Employees
WHERE Salary > 99999 and Salary < 100001;
```

DDM Dynamic Data Masking

- **Best Practices and Common Use Cases**
- Creating a mask on a column does not prevent updates to that column

- Using SELECT INTO or INSERT INTO to copy data from a masked column into another table results in masked data in the target table.

- Dynamic Data Masking is applied when running SQL Server Import and Export. A database containing masked columns will result in a backup file with masked data (assuming it is exported by a user without **UNMASK** privileges), and the imported database will contain statically masked data.

DDM Dynamic Data Masking

- First Released in SQL 2016
- Enterprise and standard SKUs

DDM Dynamic Data Masking

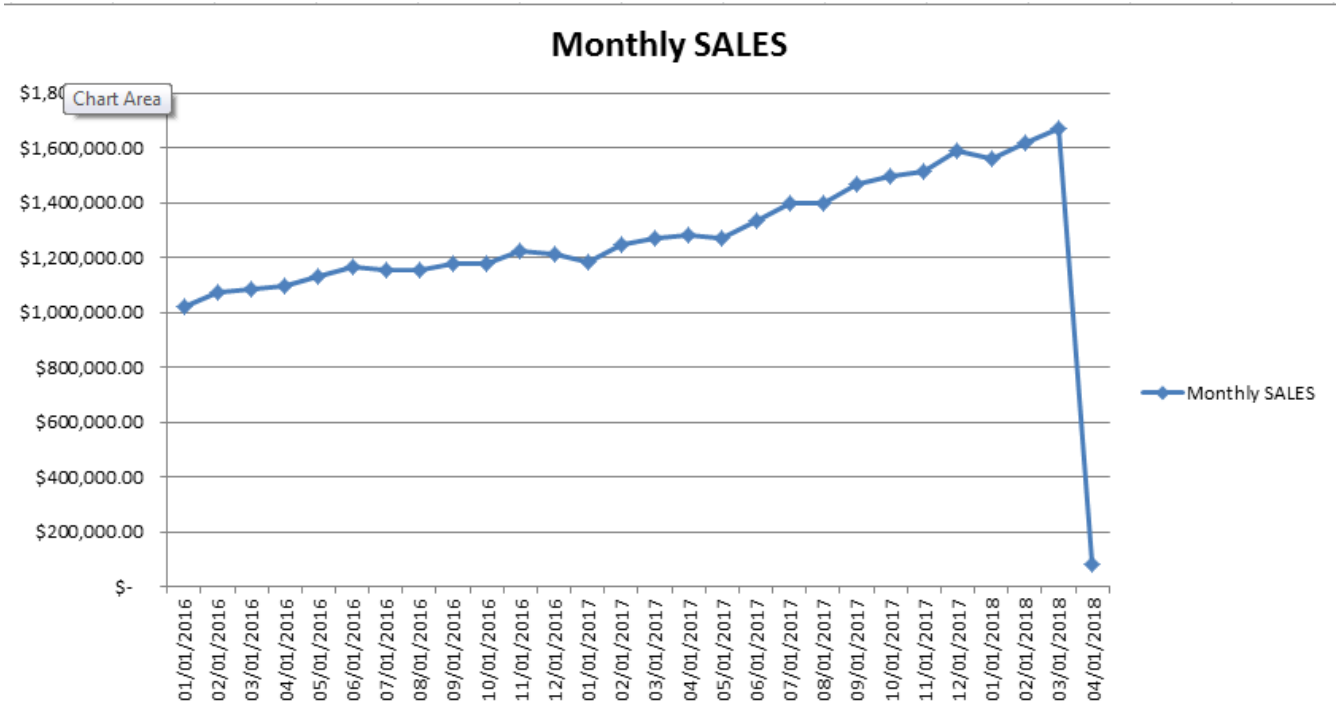
- USAGE Scenario
- ACME Management is worried about Data Exfiltration of their Customer information. Implements Data Masking on Customer Contact Info.

DDM Dynamic Data Masking

- Demo

DDM Dynamic Data Masking

- ACME monthly Sales after Data Masking



DDM Dynamic Data Masking

- Questions?
  - Can we mask en-cyrpted data? No, also pointless.
  - Can we use the masked data in a calculation?
  - Eg SUM or AVG of Salary? Salary *10
  - Can the user bypass masking?

RLS Row Level Security

- Security Feature.
- Vertical filtering of tables to control access
- Eg allow salespeople to only see customers in their sales area.

RLS Row Level Security

- Create a schemabound inline table function that evaluates to a zero value for rows for a user has access to
- Create Security policy to bind function to table(s)
- Use filter and Block predicates restrict access to defined rows.

RLS Row Level Security

```
CREATE SCHEMA Security;
GO

CREATE FUNCTION Security.fn_securitypredicate(@SalesRep AS sysname)
RETURNS TABLE WITH SCHEMABINDING AS RETURN
SELECT 1 AS fn_securitypredicate_result
WHERE @SalesRep = USER_NAME() OR USER_NAME() = 'Manager';
```

RLS Row Level Security

Security Policy

```
CREATE SECURITY POLICY SalesFilter
ADD FILTER PREDICATE Security.fn_securitypredicate(SalesRep)
ON dbo.Sales WITH (STATE = ON);
```

RLS Row Level Security

- RLS can be added to existing tables. – minimal downtime
- Applies to members of db_owner role. Ie dbo will also be filtered. Use branching logic in function to allow dbo unfiltered access (if required).

RLS Row Level Security

- Scenario: Management is concerned that SalesReps will download the company's customer list and join a competitor. Data masking was implemented but sales declined precipitously when Sales Reps tried cold calling 1-XXX-XXX-XX## and tried talking to "Mr or Mrs Axxxx xxxxN"

RLS Row Level Security

- What about applications that have a single application user?
  - Session context
  - EXEC sp_set_session_context @key=N'UserId', @value=1;

RLS Row Level Security

Security Function that uses session_context

```
CREATE SCHEMA Security;
GO

CREATE FUNCTION Security.fn_securitypredicate(@AppUserId int)
RETURNS TABLE WITH SCHEMABINDING AS
    RETURN SELECT 1 AS fn_securitypredicate_result WHERE
    DATABASE_PRINCIPAL_ID() = DATABASE_PRINCIPAL_ID('AppUser')
    AND CAST(SESSION_CONTEXT(N'UserId') AS int) = @AppUserId;
    GO
```

RLS Row Level Security

How to destroy performance with RLS

The security predicate gets fired with every row. Complicated and expensive access logic in the security predicate function can destroy performance.

RLS Row Level Security

- First Released in SQL 2016
- Enterprise and standard SKUs

references

- **What's new in SSMS 17.5: Data Discovery and Classification**
- https://blogs.technet.microsoft.com/dataplatforminsider/2018/02/20/whats-new-in-ssms-17-5-data-discovery-and-classification/

references

- SQL Vulnerability Assessment
  - https://docs.microsoft.com/en-us/sql/relational-databases/security/sql-vulnerability-assessment?view=sql-server-2017/
- SQL Data Discovery and Classification
  - https://docs.microsoft.com/en-us/sql/relational-databases/security/sql-data-discovery-and-classification?view=sql-server-2017

references

- https://docs.microsoft.com/en-us/sql/sql-server/editions-and-components-of-sql-server-2016
- TDE Vs Backup Encryption http://www.edwinmsarmiento.com/sql-server-encrypted-backups-transparent-data-encryption-or-backup-encryption/

Appendix

- GDPR
  - May 25, 2018

- **Guide to enhancing privacy and addressing GDPR requirements with the Microsoft SQL platform**
- https://docs.microsoft.com/en-us/sql/relational-databases/security/microsoft-sql-and-the-gdpr-requirements?view=sql-server-2017

Appendix

- Star Wars Rogue One
- What Rogue One Teaches us about Archiving and Security
  - https://blog.microfocus.com/what-rogue-one-teaches-us-about-archiving-security/

- Seven Security Lessons from Rogue One: A Star Wars Story
- https://www.onelogin.com/blog/seven-security-lessons-from-rogue-one-a-star-wars-story

- **May the firewall be with you: Tech security lessons from Star Wars 'Rogue One'**
- https://www.geekwire.com/2017/may-firewall-tech-security-lessons-star-wars-rogue-one/