

Hacking SQL Server

André Melancia

SQL Saturday #667

Oslo

2017-09-02



*PASS

SQLSATURDAY

OSLO | 02 SEP 2017

Our awesome sponsors!

Venue Partner



Platinum Sponsor



Gold Sponsor



Silver Sponsor



Bronze Sponsor



Please visit the sponsor area in the break and interact with them. They are the reason we can hold this conference free of charge!





Get-NSASpyInfo "André Melancia"

➔ Food devouring expert

➔ Lunar Cat

➔ Owner / Principal Consultant

➔ Meow us at <http://LunarCat.PT>



➔ Microsoft Certified Trainer (MCT)

➔ Microsoft Data Platform MVP

➔ Contacts

➔ <http://Andy.PT>

➔ Andy@Andy.COM.PT

➔ <https://LinkedIn.COM/in/AndreMelancia>

➔ <https://Facebook.COM/Andy.COM.PT>

➔ <https://Twitter.COM/AndyPT>

➔ Volunteer and/or Speaker at:

➔ SQL Saturdays, SQLBits, SQLRelay, SQLDay, MCT Summit Europe, Programar Saturday

➔ GroupBy, 24Hop (Data Security & Portuguese), PASS Global Portuguese VG, Channel9

➔ Data Community Portugal, PowerShell Portugal, IT Pro Portugal, IoT Portugal, other crazy user groups...

➔ IPv6 Portugal, DNSSec Portugal, Windows Development Portugal, etc.





Hacking?

This is a SECURITY session.

All software is perfect, but...

... configuration is done by humans ☹️



NSA Secrets



The NSA
*The only part of government
that actually listens.*

Disclaimer:

No actual state secrets will be revealed.

Please do not send agents to my house again...

Til info:

Ingen faktiske
statshemmeligheter vil bli
avslørt.

Venligst ikke send agenter til
huset mitt igjen...



Hacking British plug sockets





Hacking Indian plug sockets





Requirements for demos

➞ To run the demos in this session you'll need:

- ➞ SQL Server 2016 (for RLS and DDM), 2012 for the rest
- ➞ Visual Studio 2015 (for the TCP Proxy)

➞ You'll also need to know what this means:

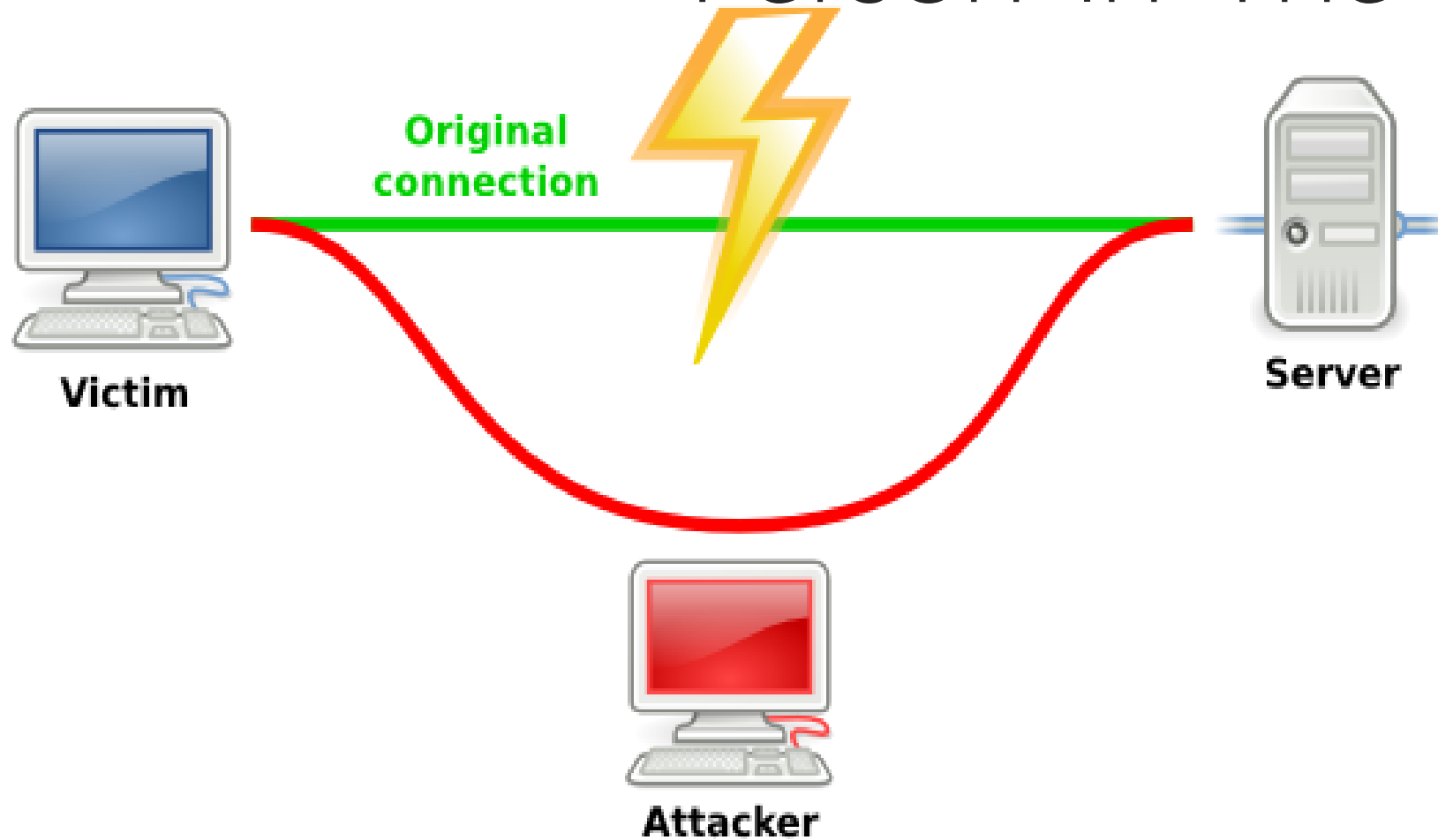
```
SELECT *  
FROM PASS.SQLSaturday.Oslo  
WHERE ID = 667;
```

➞ (In other words, this is a introductory level security session for DBAs and developers, assuming you already know all basic database concepts)



Man-~~In-The-Middle~~

Person-In-The-Middle





Man-In-The-Middle

➔ SQL Server uses TDS (Tabular Data Stream)

- ➔ https://en.wikipedia.org/wiki/Tabular_Data_Stream
- ➔ <https://msdn.microsoft.com/en-us/library/dd304523.aspx>
 - ➔ [MS-TDS]: Tabular Data Stream Protocol
- ➔ <https://msdn.microsoft.com/en-us/sqlserver/gg415738.aspx>
 - ➔ Getting Started with SQL Server Protocol

➔ TDS was originally created by Sybase in the 1980s

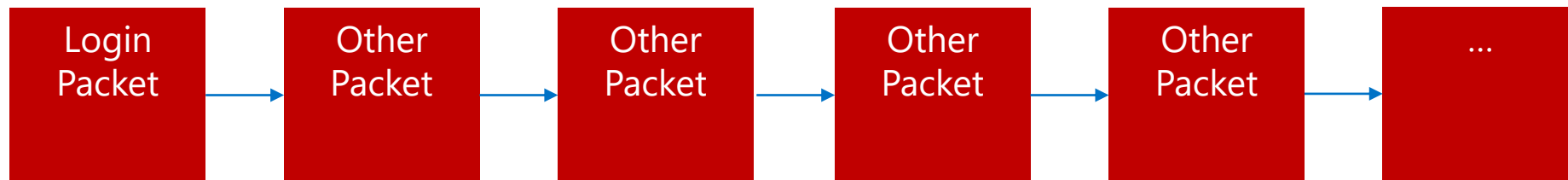
➔ Default port: TCP 1433 (default instance)

- ➔ Is it a good idea to change it?
- ➔ Changing to a higher value protects from port scan attacks?
- ➔ Is this a problem in Azure?
- ➔ NO!

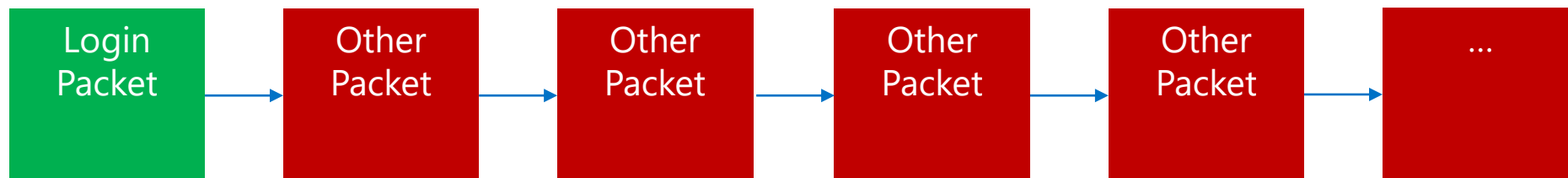


Man-In-The-Middle

- ➔ Up to SQL Server 2000, the login/password packet was NOT encrypted.
- ➔ Plain text login, weak hash password



- ➔ SQL Server 2005 solves this, but the other packets are unencrypted by default!

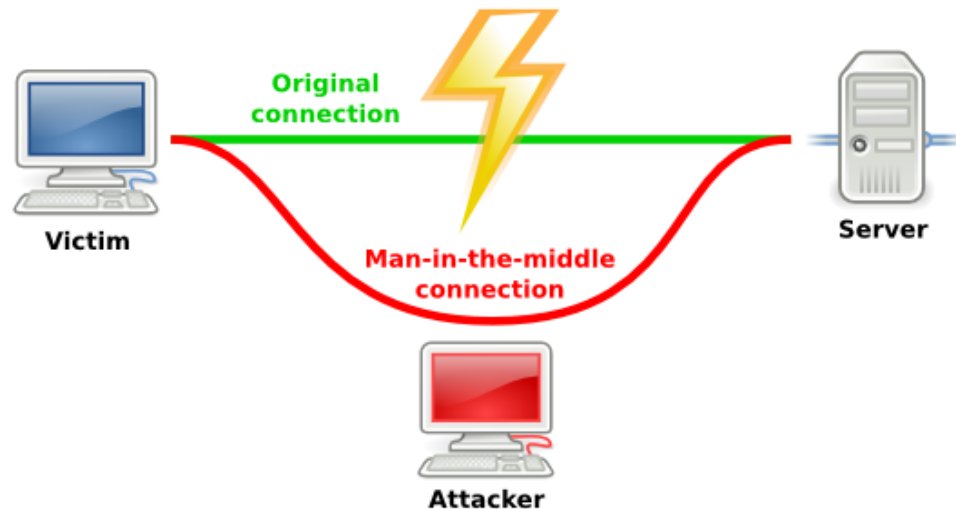




Man-In-The-Middle

MITM, the hard way:

- ➔ Physical network tampering
- ➔ Configure/crack the router/switch
- ➔ ARP Spoofing
- ➔ DNS Poisoning
- ➔ Etc.





Man-In-The-Middle

MITM, the easy way:

➔ Change the "hosts" file

➔ %windir%\System32\Drivers\Etc\Hosts

➔ Change SQL Server "alias"

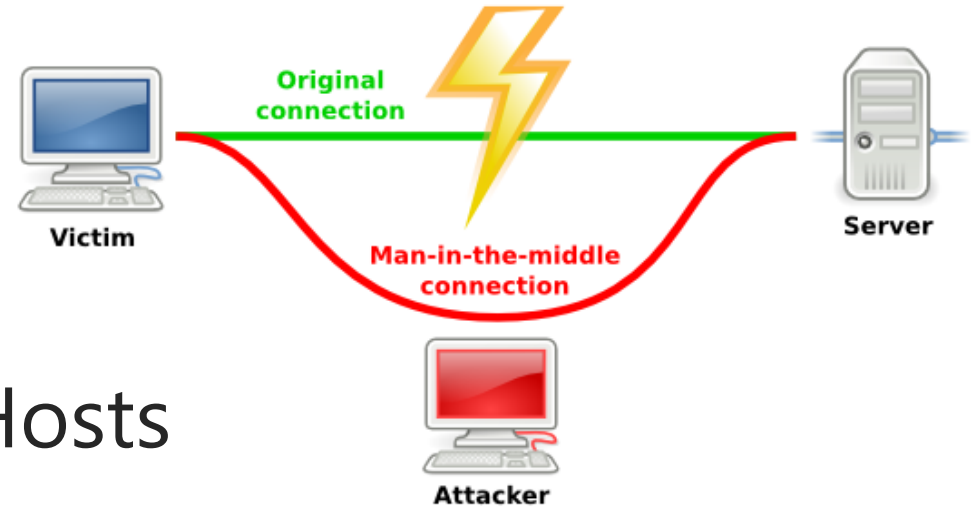
➔ SQL Server Configuration Manager

➔ %windir%\System32\CliCfg.EXE

➔ Registry:

➔ HKLM\Software\Microsoft\MSSQLServer\Client\ConnectTo (32bit)

➔ HKLM\Software\WOW6432Node\Microsoft\MSSQLServer\Client\ConnectTo (64bit)





Man-In-The-Middle

ONLY TRY
THIS AT HOME!

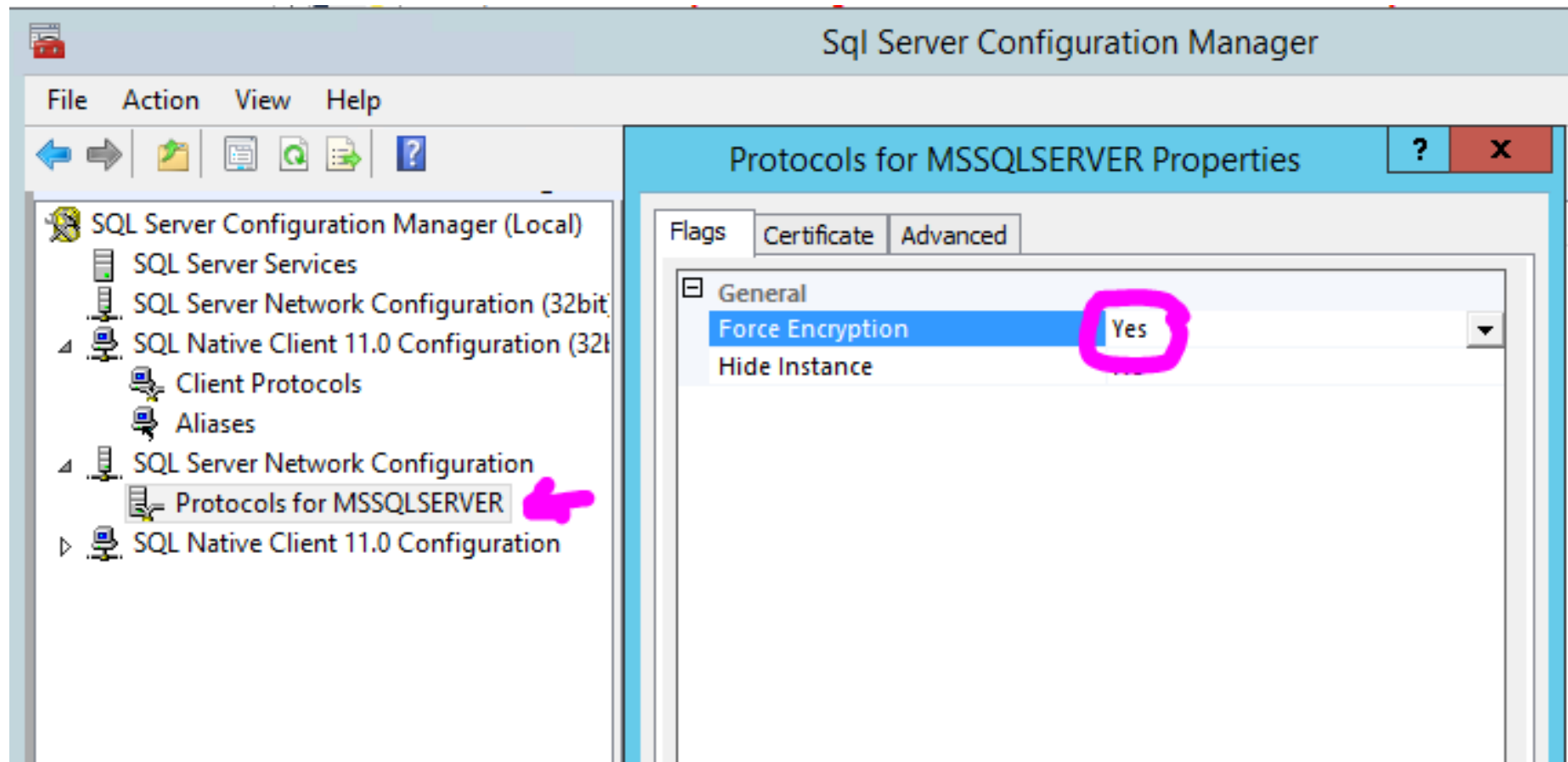
KUN utføres
hjemme!





Man-In-The-Middle

Always enable encryption on the SERVER:





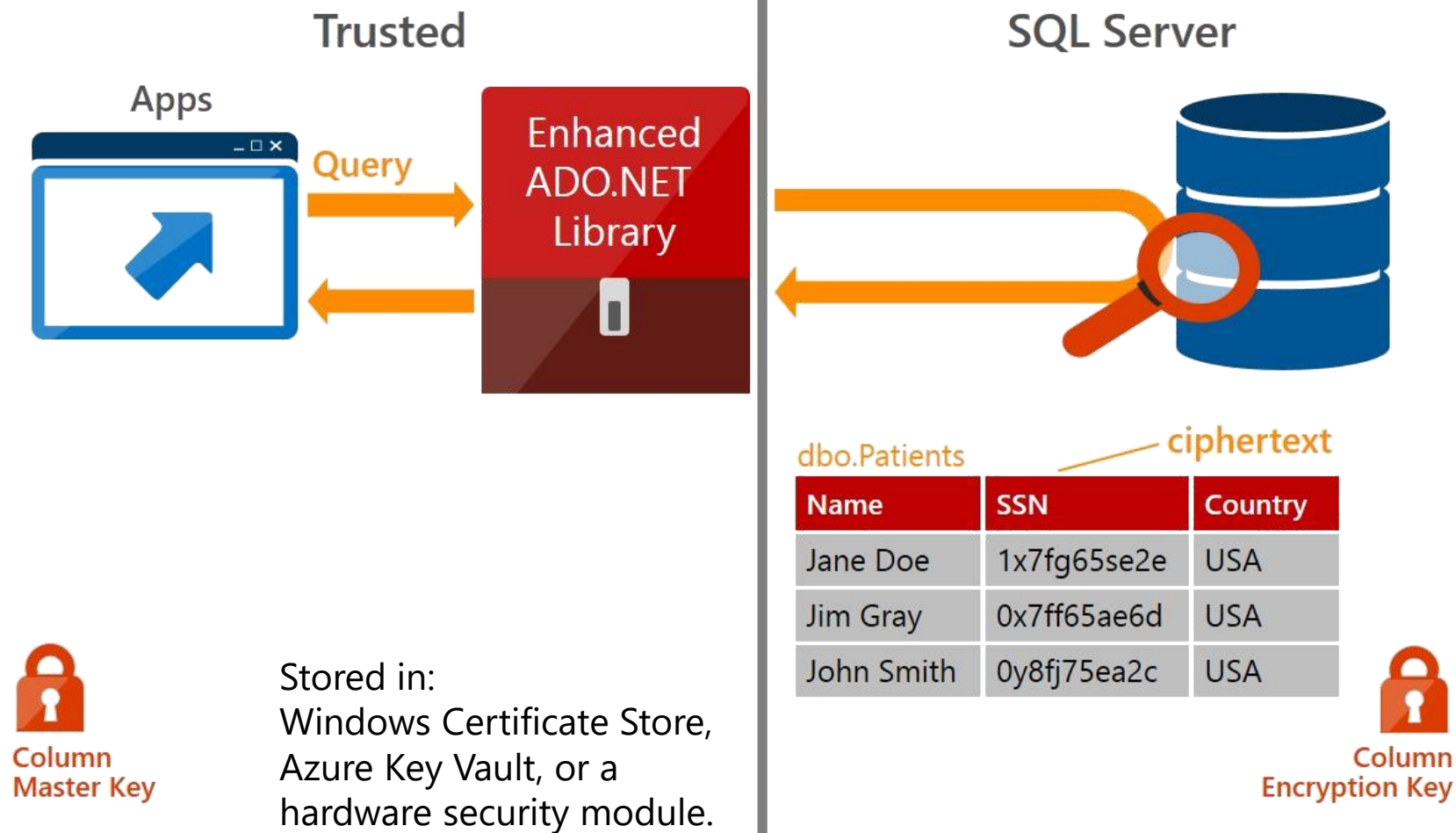
Man-In-The-Middle

Always enable encryption on the SERVER:

- ➔ <https://msdn.microsoft.com/en-us/library/ms191192.aspx>
 - Enable Encrypted Connections to the Database Engine
- ➔ <https://msdn.microsoft.com/en-us/library/ms131691.aspx>
 - Using Encryption Without Validation
- ➔ <https://msdn.microsoft.com/en-us/library/ms130822.aspx>
 - Using Connection String Keywords with SQL Server Native Client
 - ➔ Encrypt
 - ➔ TrustServerCertificate



Always Encrypted





Always Encrypted

- ➔ Everything over the wire is encrypted
- ➔ Data is stored encrypted in the database files
- ➔ Only the App (client) side has the decryption key
- ➔ Encryption is by Column, not Table
- ➔ Options:
 - ➔ **Deterministic (allows equality lookups)**
 - ➔ **Random**
- ➔ Drivers already available for Linux clients

More info:

<https://blogs.msdn.microsoft.com/sqlsecurity/2015/06/04/getting-started-with-always-encrypted>

<https://msdn.microsoft.com/en-us/library/mt708953.aspx> - Overview of Key Management for Always Encrypted



Always Encrypted

Hack it?

ID	Name	Country	Country [AE-Deterministic]	Country [AE-Random]
1	Client 1	PT	0x28DA20...	0x334523...
2	Client 2	PT	0x28DA20...	0x4d3133...
3	Client 3	PT	0x28DA20...	0x42D320...
4	Client 4	PT	0x28DA20...	0x24ABD0...
5	Client 5	UK	0x9723AC...	0xCD3289...
6	Client 6	UK	0x9723AC...	0x9CD293...
7	Client 7	UA	0x127A2A...	0xA4387A...
8	Client 8	UA	0x127A2A...	0x1999BA...
9	Client 9	UA	0x127A2A...	0x729F2A...
10	Client 10	UA	0x127A2A...	0x89478A...
11	Client 11	RU	0xF0122E...	0xFE349A...
12	Client 12	PL	0xEAB53E...	0xFA100E...
13	Client 13	DE	0x00A3D1...	0x11AFB1...



Instant File Initialisation

SQL Server 2016 Setup

Server Configuration

Specify the service accounts and collation configuration.

Product Key
License Terms
Global Rules
Product Updates
Install Setup Files
Install Rules
Feature Selection
Feature Rules
Instance Configuration
Server Configuration
Database Engine Configuration
Consent to install Microsoft R ...
Feature Configuration Rules
Ready to Install
Installation Progress

Service Accounts Collation

Microsoft recommends that you use a separate account for each SQL Server service.

Service	Account Name	Password	Startup Type
SQL Server Agent	NT Service\SQLSERVERA...		Manual
SQL Server Database Engine	NT Service\MSSQLSERVER		Manual
SQL Server Launchpad	NT Service\MSSQLLaunc...		Automatic
SQL Server Browser	NT AUTHORITY\LOCAL ...		Disabled

☒ Grant Perform Volume Maintenance Task privilege to SQL Server Database Engine Service

This privilege enables instant file initialization by avoiding zeroing of data pages. This may lead to information disclosure by allowing deleted content to be accessed.

[Click here for details](#)



Instant File Initialisation

Hack it?

- ➞ Just copy the MDF/NDF files you created with IFI...
- ➞ ... and all the previous information from the disk is there...



Transparent Data Encryption (TDE)

- ➔ Encrypts the whole Database
- ➔ Data is stored encrypted in the storage disks
 - ➔ Page level encryption
- ➔ Decrypted in RAM on the server (on demand)
 - ➔ **Server has the keys!!!**
- ➔ Recommended: store the keys and the data in **separate** storage disks
- ➔ Transparent to the Apps
 - ➔ **Data is decrypted before leaving the server**

More info: <https://msdn.microsoft.com/en-us/library/bb934049.aspx>



Transparent Data Encryption (TDE)

Hack it?

- ➞ Remember... "Server has the keys!!!"
- ➞ So... Get the keys from someone's backup...
- ➞ Also... Backups and network communication are not encrypted by TDE



Backup Encryption (TDE)

- ➔ Data is stored encrypted in the backup media
- ➔ Works with
 - ➔ TDE
 - ➔ Managed Backup to Azure

More info: <https://msdn.microsoft.com/en-us/library/dn449489.aspx>



Dynamic Data Masking

```
CREATE TABLE Membership
(
    MemberID    int                IDENTITY PRIMARY KEY,
    FirstName   varchar(66) MASKED WITH (FUNCTION = 'partial(1,"XXXXXXXX",0)') NULL,
    LastName    varchar(66)                                NOT NULL,
    Phone       varchar(66) MASKED WITH (FUNCTION = 'default()') NULL,
    Email       varchar(66) MASKED WITH (FUNCTION = 'email()') NULL
);
```

Original:

1	Roberto	Tamburello	555.123.4567	RTamburello@contoso.com
---	---------	------------	--------------	-------------------------

Masked:

1	RXXXXXXXX	Tamburello	xxxx	RXXX@XXXX.com
---	-----------	------------	------	---------------



Dynamic Data Masking

- ➔ By Column
- ➔ SELECT permission shows masked data
 - ➔ To see the data you need UNMASK permission
- ➔ Does not prevent UPDATES to masked columns
- ➔ Using SELECT INTO or INSERT INTO doesn't work
 - ➔ Output is masked text (trash)
- ➔ Backups are also masked text (trash),
 - ➔ Unless the backup user has UNMASK permission
- ➔ Hacks?



Dynamic Data (UN)Masking

ONLY TRY
THIS AT HOME!

KUN utføres
hjemme!





SQL Server 2016 SP1 changes

Feature	RTM				SP1			
	Standard	Web	Express	Local DB	Standard	Web	Express	Local DB
Row-level security	Yes	No	No	No	Yes	Yes	Yes	Yes
Dynamic Data Masking	Yes	No	No	No	Yes	Yes	Yes	Yes
Change data capture*	No	No	No	No	Yes	Yes	No*	No*
Database snapshot	No	No	No	No	Yes	Yes	Yes	Yes
Columnstore	No	No	No	No	Yes	Yes	Yes	Yes
Partitioning	No	No	No	No	Yes	Yes	Yes	Yes
Compression	No	No	No	No	Yes	Yes	Yes	Yes
In Memory OLTP	No	No	No	No	Yes	Yes	Yes	No**
Always Encrypted	No	No	No	No	Yes	Yes	Yes	Yes
PolyBase	No	No	No	No	Yes	Yes	Yes	No
Fine grained auditing	No	No	No	No	Yes	Yes	Yes	Yes
Multiple filestream containers	No	No	No	No	Yes	Yes	Yes	No**

* Requires SQL Server agent which is not part of SQL Server Express Editions

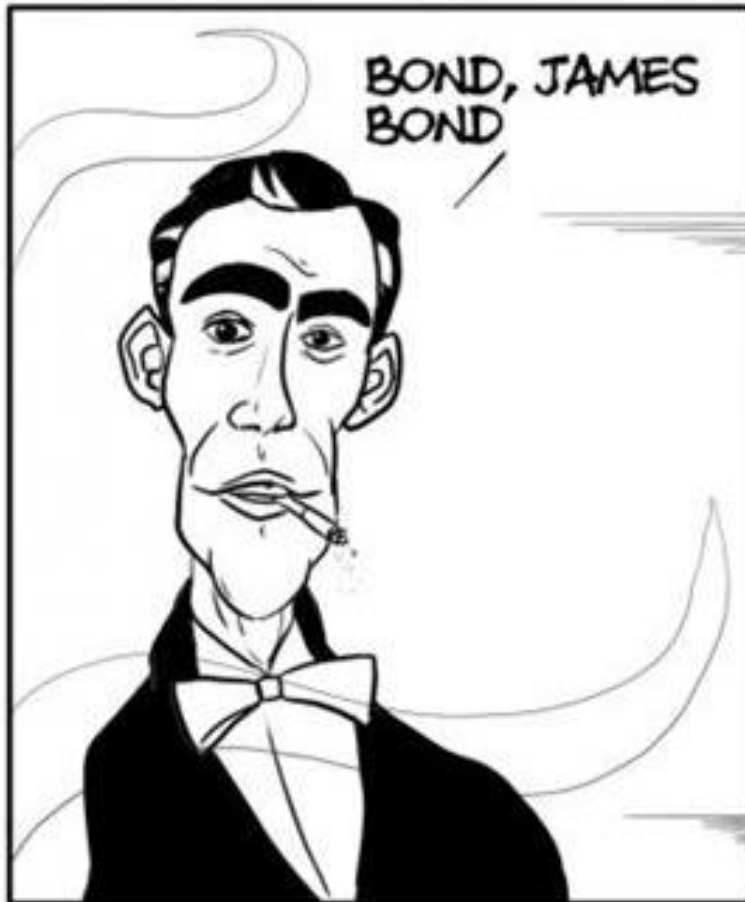
** Requires creating filestream file groups which is not possible in Local DB due to insufficient permissions.



Row Level (In)Security – Demo!

ONLY TRY
THIS AT HOME!

KUN utføres
hjemme!





Row Level Security (RLS)

➔ Function Predicate

- ➔ Defines Business logic/rules/access
- ➔ User defined inline table valued-function (reusable)

➔ Security Policy

- ➔ Collection of Predicates
- ➔ Predicates added as FILTER or BLOCK

More info: <https://msdn.microsoft.com/en-us/library/dn765131.aspx>



Never Say DEMO Again...

ONLY TRY
THIS AT HOME!

KUN utføres
hjemme!

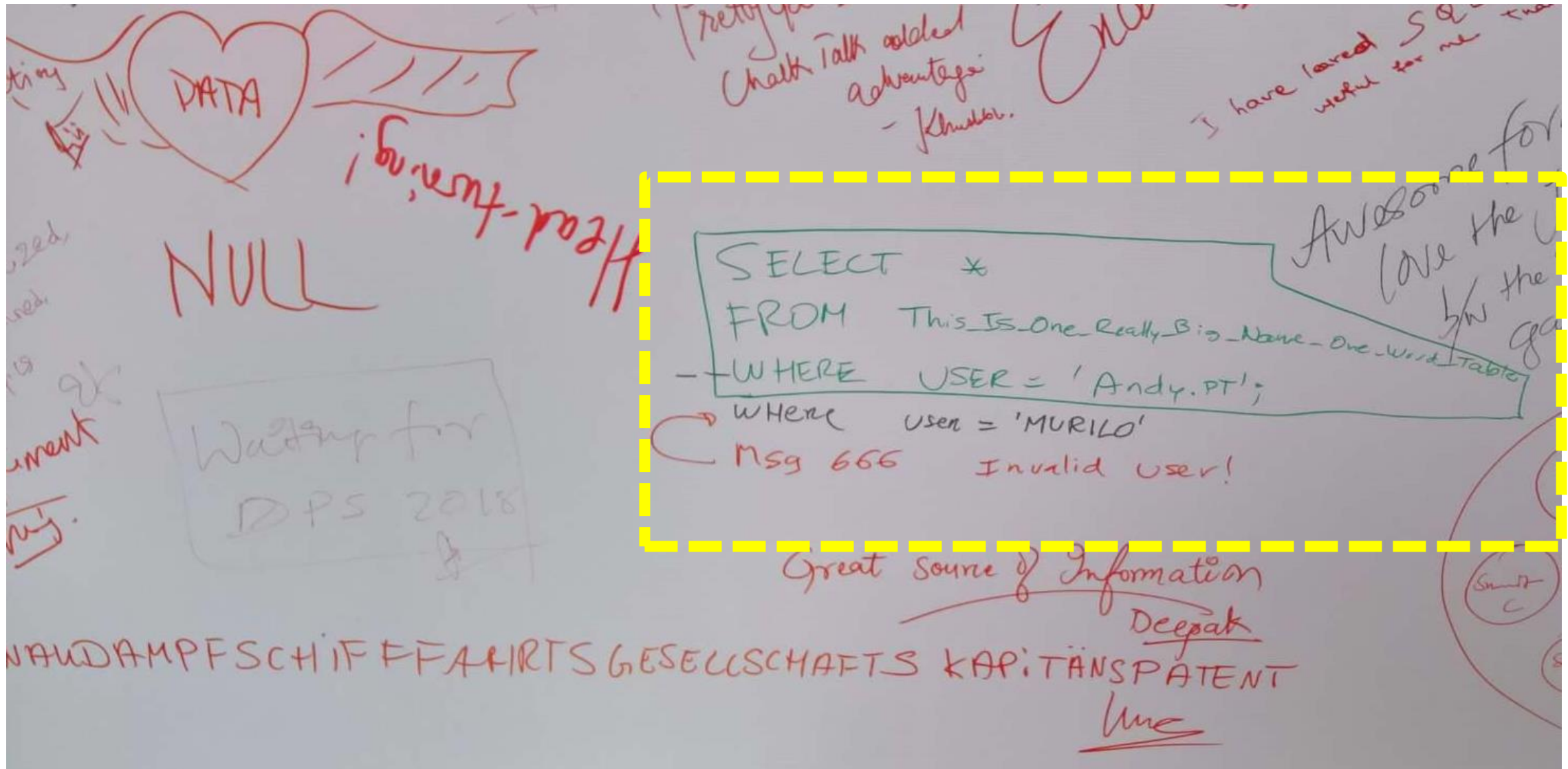
IS A SECRET AGENT



**USES HIS ACTUAL
NAME**



"Write one word about..."





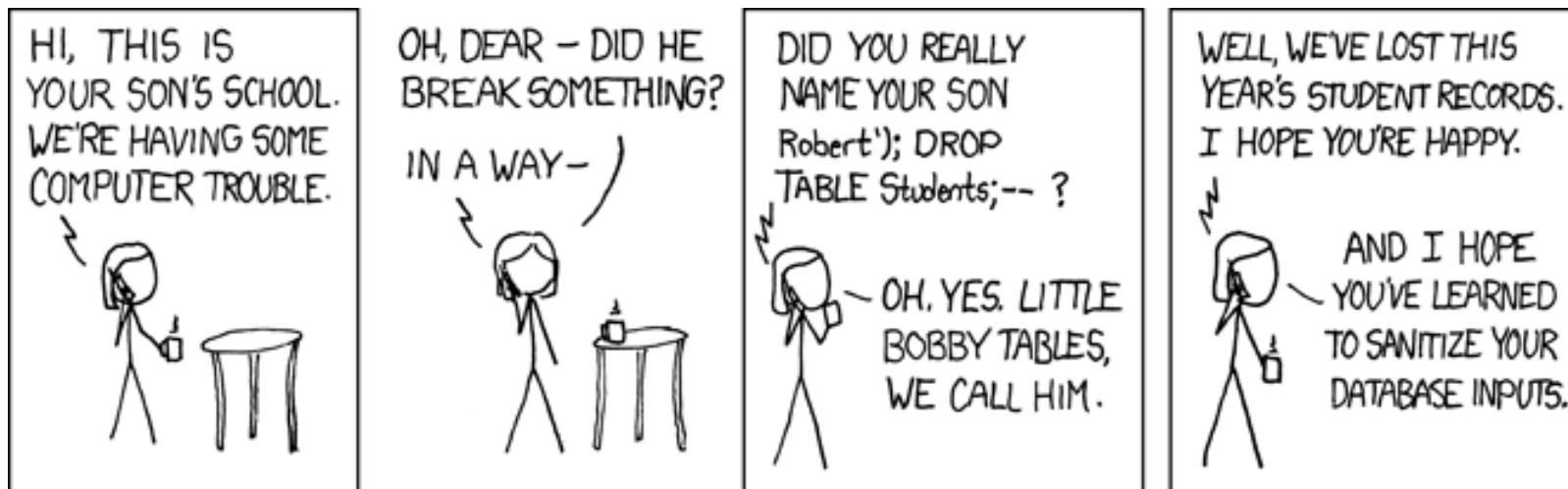
SQL Injection

➔ We get the Category parameter in a web page...

`http://Example.COM/?Cat=trash'; DROP TABLE TopSecretStuff; --`

```
SQL = " SELECT *  
      + " FROM TableX  
      + " WHERE FieldZ = '" + Category + "'; ";
```

➔ Executing this is NOT SAFE!





SQL Injection

In the Apps/Libs, use prepared statements:

- ➞ Available in most languages
 - ➞ .Net, PHP, Java, etc.
- ➞ Prevents SQL Injection
- ➞ Validates Data Types
- ➞ Usually integrated into app classes



SQL Injection

Prepared Statement example:

```
SqlConnection connection = new
SqlConnection(connectionString)

SqlCommand cmd                = new SqlCommand();
cmd.Connection                = SomeConnection;
cmd.CommandText                = "SP_SalesByCategory";
cmd.CommandType                = CommandType.StoredProcedure;

SqlParameter parameter        = new SqlParameter();
parameter.ParameterName       = "@CategoryName";
parameter.SqlDbType           = SqlDbType.NVarChar;
parameter.Direction            = ParameterDirection.Input;
parameter.Value                = ThatFunnyParameter;
cmd.Parameters.Add (parameter);
```

More info: [https://msdn.microsoft.com/en-us/library/yy6y35y8\(v=vs.110\).aspx](https://msdn.microsoft.com/en-us/library/yy6y35y8(v=vs.110).aspx)



Permissions

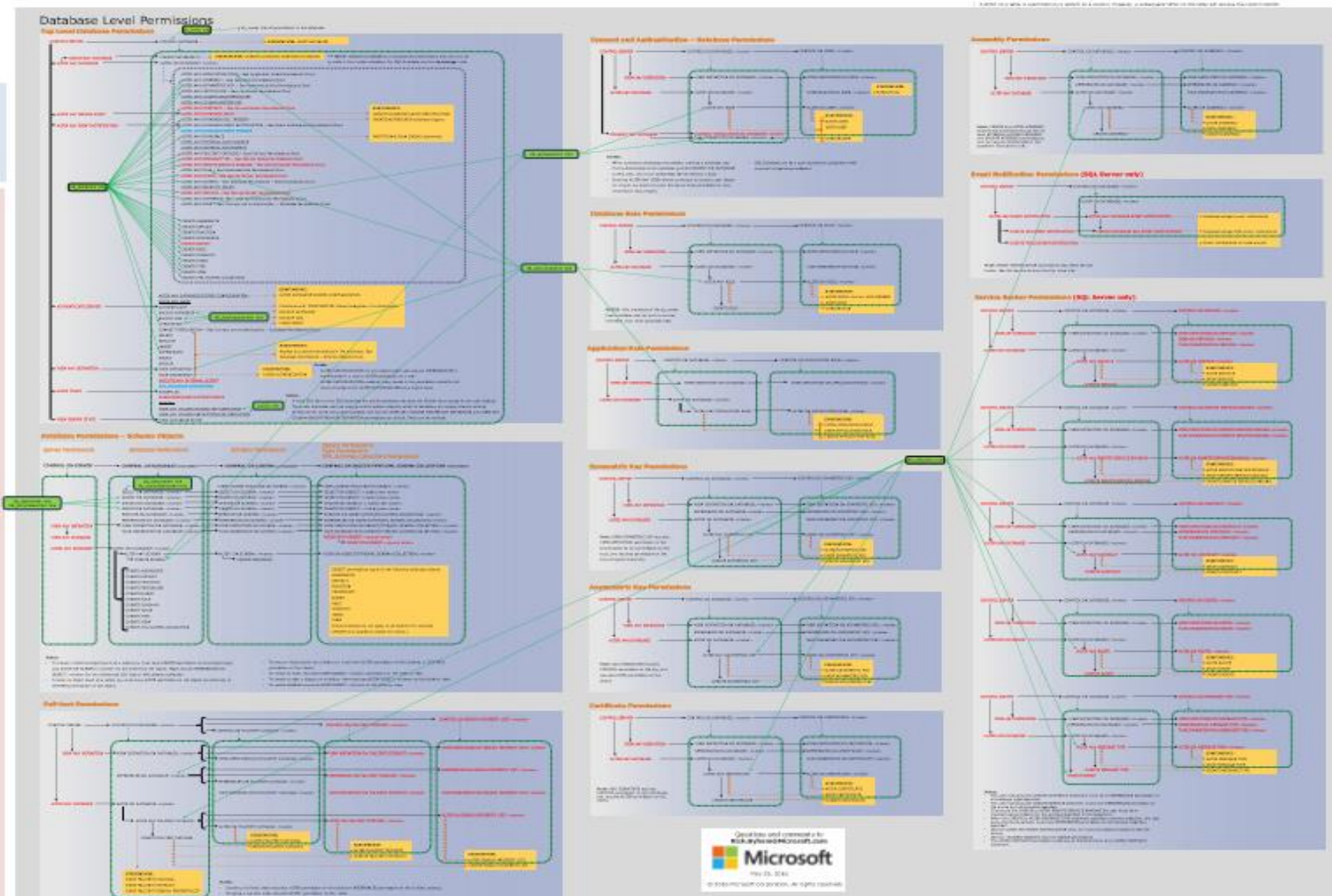
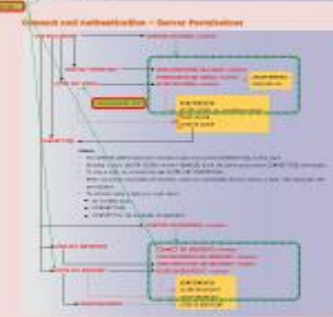
- ➞ By SQL Server login (e.g. "sa")
- ➞ By AD/Windows login (domain or local)
- ➞ By AD group
- ➞ By DB user
- ➞ By server role
- ➞ By database role
- ➞ By application role



Permissions poster by Rick Byham

Microsoft SQL Server 2016 and Azure SQL Database Database Engine Permissions

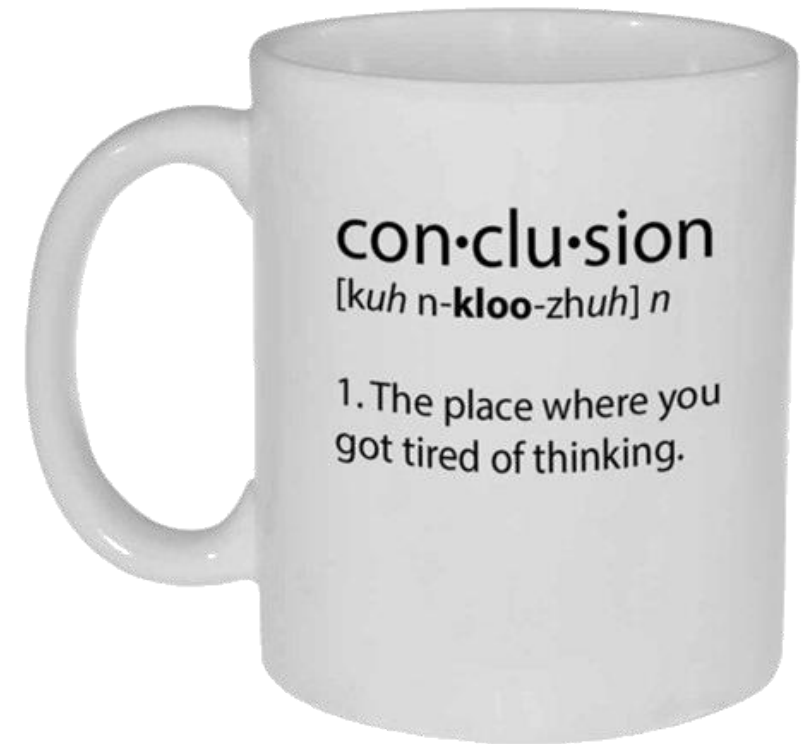
Permissions Overview
The permissions system in Microsoft SQL Server is designed to be flexible and powerful, allowing you to grant permissions to users and roles in a way that meets your needs. This poster provides a comprehensive overview of the permissions system, including the different types of permissions and how they are granted.





Recommendations

- ➞ ALWAYS encrypt traffic!
- ➞ Don't expose TCP 1433, UDP 1434 (and others) directly on the Internet
 - ➞ Use a VPN or other options
- ➞ Always assign the least amount of permissions possible!
- ➞ Use prepared statements to avoid SQL Injection





Even if you can't get the Bond Girl, remember...

➞ SQL Server 2016 and 2017 have a lot of new features...

➞ But they're not perfect!

➞ ALWAYS,

➞ ALWAYS,

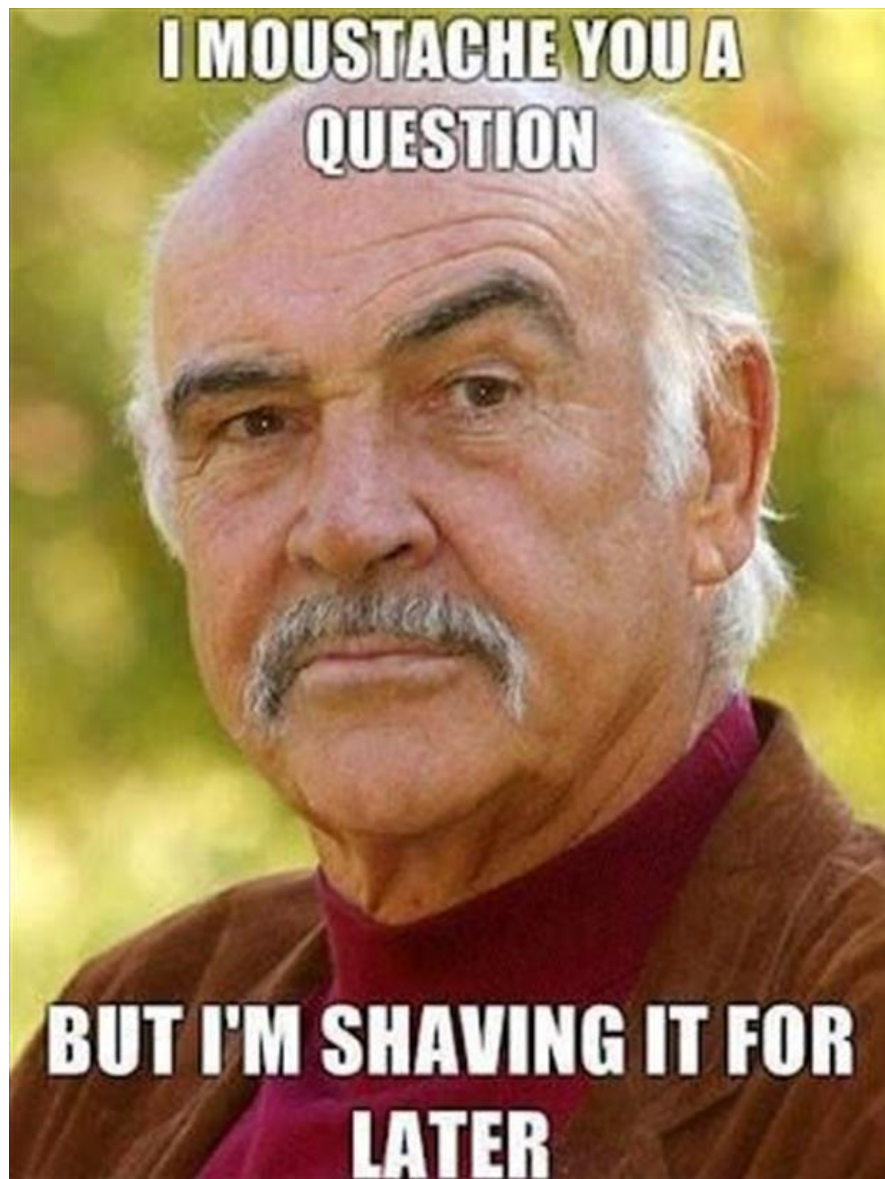
➞ ALWAYS,

➞ Install the latest Service Packs and patches!



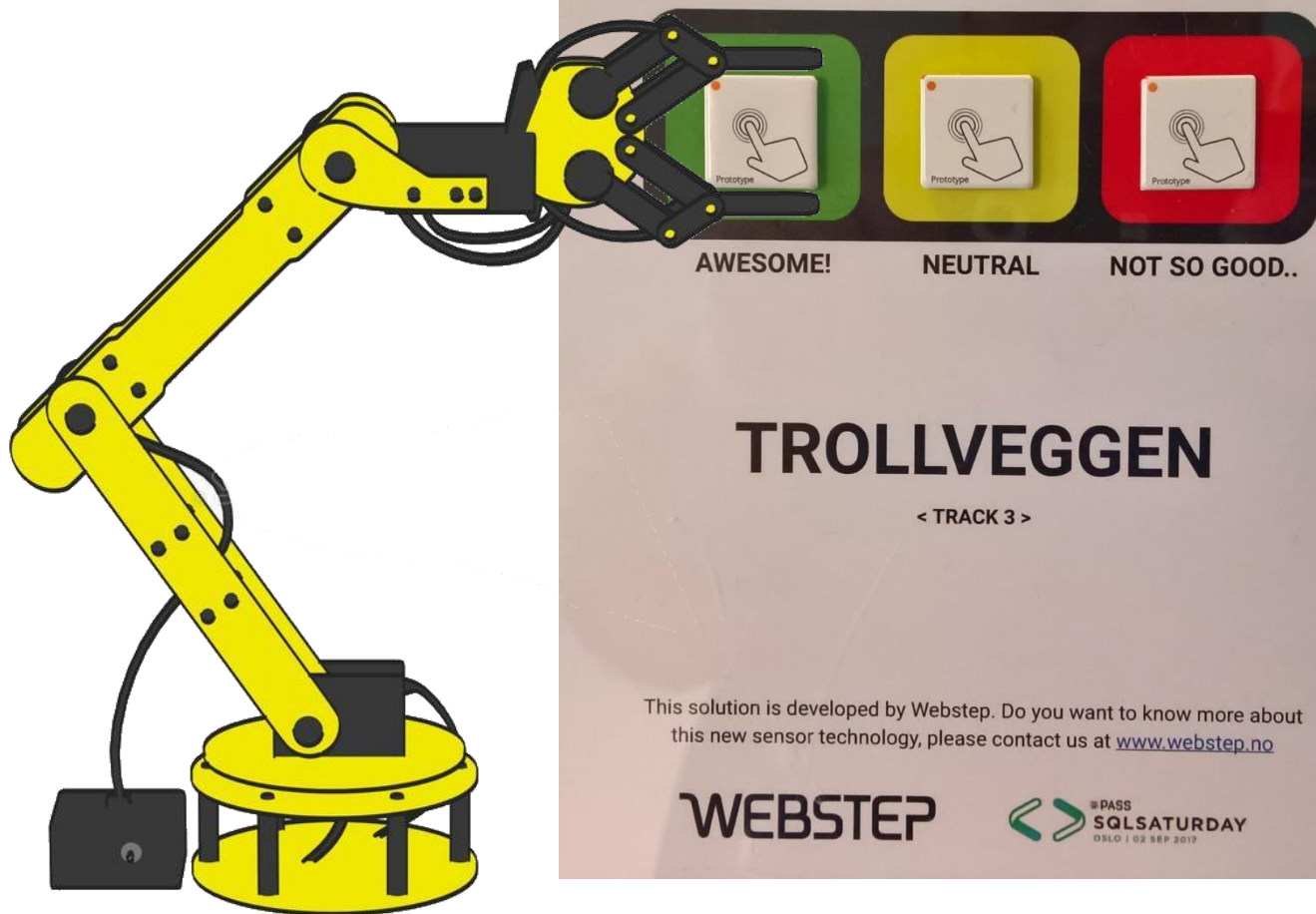


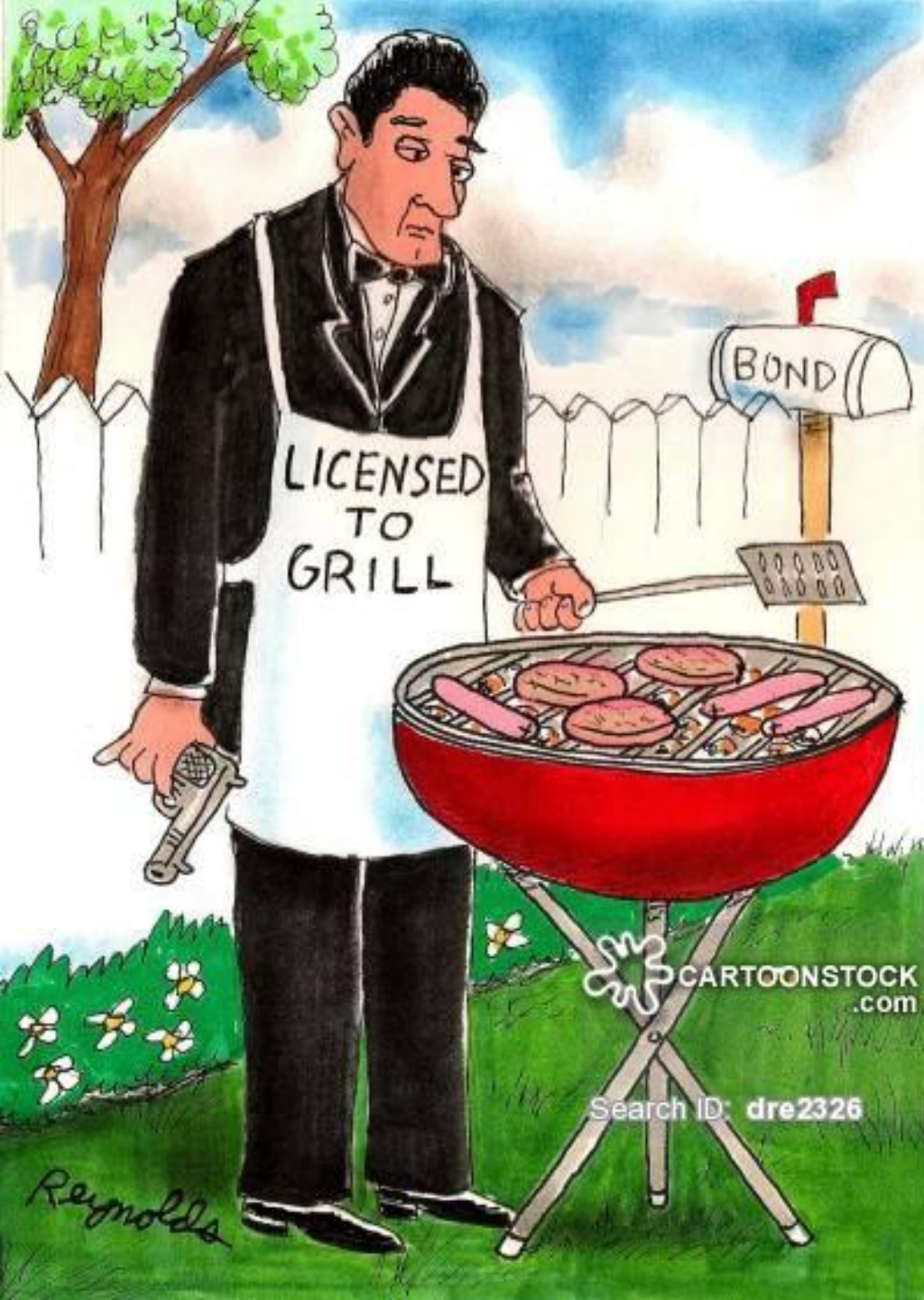
Receive-Questions –Input "Speech"





Get-SessionFeedback –Input "IoT Solution"





Takk!

Thank you!

Obrigado!

धन्यवाद

Благодаря!

Дуже дякую!

Köszönöm!

Dziękuję!

Merci beaucoup!

Dank je!

Go raibh maith agaibh!

Danke!

Děkuju!

Diolch!

¡Gracias!

Arigato!



André Melancia



<http://Andy.PT>

<http://LunarCat.PT>