

# DBA

SQL Server Administration

Part I



**Mohammad Reza Gerami**  
mrgerami@aut.ac.ir  
gerami@virasec.ir



# Requirement and Recommendations

## Requirement and Recommendations



# **SQL Server 2016 and 2017: Hardware and software requirements**

## **Hardware requirements**

The following hardware requirements apply to SQL Server 2016 and SQL Server 2017:



| Component       | Requirement  |
|-----------------|--|
| Hard Disk       | <p>SQL Server requires a minimum of 6 GB of available hard-disk space.</p> <p>Disk space requirements will vary with the SQL Server components you install. For more information. Installing SQL Server on computers with the NTFS or ReFS file formats is recommended. The FAT32 file system is supported but not recommended as it is less secure than the NTFS or ReFS file systems. Read-only, mapped, or compressed drives are blocked during installation.</p> |
| Drive           | A DVD drive, as appropriate, is required for installation from disc.   |
| Monitor         | SQL Server requires Super-VGA (800x600) or higher resolution monitor.  |
| Internet        | Internet functionality requires Internet access (fees may apply).  |
| Memory *        | <p>Minimum: Express Editions: 512 MB All other editions: 1 GB</p> <p>Recommended: Express Editions: 1 GB</p> <p>All other editions: At least 4 GB and should be increased as database size increases to ensure optimal performance.</p>  |
| Processor Speed | <p>Minimum: x64 Processor: 1.4 GHz</p> <p>Recommended: 2.0 GHz or faster</p>   |
| Processor Type  | x64 Processor: AMD Opteron, AMD Athlon 64, Intel Xeon with Intel EM64T support, Intel Pentium IV with EM64T support  |

## Requirement and Recommendations

### **Note:**

Installation of SQL Server is supported on x64 processors only. It is no longer supported on x86 processors.

\* The minimum memory required for installing the Data Quality Server component in Data Quality Services (DQS) is 2 GB of RAM, which is different from the SQL Server minimum memory requirement







# **SQL Server 2016 and 2017: Hardware and software requirements**

## **Software requirements**

The table in this section lists the minimum software requirements for running SQL Server. The following software requirements apply to all installations:



| Component        | Requirement   |
|------------------|---|
| .NET Framework   | SQL Server 2016 (13.x) and later require .NET Framework 4.6 for the Database Engine, Master Data Services, or Replication. SQL Server setup automatically installs .NET Framework. You can also manually install .NET Framework from <a href="#">Microsoft .NET Framework 4.6 (Web Installer) for Windows</a> . Windows 8.1, and Windows Server 2012 R2 require <a href="#">KB2919355</a> before installing .NET Framework 4.6.   |
| Network Software | <p>Supported operating systems for SQL Server have built-in network software. Named and default instances of a stand-alone installation support the following network protocols: Shared memory, Named Pipes, TCP/IP, and VIA.</p> <p>Note: VIA protocol is not supported on failover clusters. Clients or applications running on the same node of the failover cluster as the SQL Server instance, can use Shared Memory protocol to connect to SQL Server using its local pipe address. However this type of connection is not cluster-aware and will fail after an instance failover. It is therefore not recommended and should only be used in very specific scenarios.</p> <p>Important: The VIA protocol is deprecated. This feature is in maintenance mode and may be removed in a future version of Microsoft SQL Server. Avoid using this feature in new development work, and plan to modify applications that currently use this feature.</p> |



## VIA - Virtual Interface Adapter

A **Virtual Interface Adapter** ("VIA") is a network protocol (such as TCP/IP ...). As of July 2006 Microsoft SQL Server 2005 supports it. The specific implementation of VIA will vary from vendor to vendor.

In general, it is usually a network kind of interface but is usually a very high-performance, dedicated connection between two systems. Part of that high performance comes from specialized, dedicated hardware that knows that it has a dedicated connection and therefore doesn't have to deal with normal network addressing issues.

The VIA protocol is used to support VIA devices such as VIA Storage Area Network devices.



## named pipe

A **named pipe** is a named, one-way or duplex pipe for communication between the pipe server and one or more pipe clients. All instances of a named pipe share the same pipe name, but each instance has its own buffers and handles, and provides a separate conduit for client/server communication. The use of instances enables multiple pipe clients to use the same named pipe simultaneously.

Any process can access named pipes, subject to security checks, making named pipes an easy form of communication between related or unrelated processes.

## Operating System support

The following table shows which editions of SQL Server 2016 and 2017 are compatible with which versions of Windows:



| SQL Server edition:            | Enterprise | Developer | Standard | Web | Express |
|--------------------------------|------------|-----------|----------|-----|---------|
| Windows Server 2019 Datacenter | Yes        | Yes       | Yes      | Yes | Yes     |
| Windows Server 2019 Standard   | Yes        | Yes       | Yes      | Yes | Yes     |
| Windows Server 2019 Essentials | Yes        | Yes       | Yes      | Yes | Yes     |
| Windows Server 2016 Datacenter | Yes        | Yes       | Yes      | Yes | Yes     |
| Windows Server 2016 Standard   | Yes        | Yes       | Yes      | Yes | Yes     |
| Windows Server 2016 Essentials | Yes        | Yes       | Yes      | Yes | Yes     |

## Server Core support



Installing SQL Server 2016 and 2017 on Server Core mode is supported by the following editions of Windows Server:

|                                 |                                   |
|---------------------------------|-----------------------------------|
| Windows Server 2019 Standard    | Windows Server 2019 Datacenter    |
| Windows Server 2016 Standard    | Windows Server 2016 Datacenter    |
| Windows Server 2012 R2 Standard | Windows Server 2012 R2 Datacenter |
| Windows Server 2012 Standard    | Windows Server 2012 Datacenter    |



# SQL Server 2019: Hardware and software requirements

## Hardware requirements



The following memory and processor requirements apply to all editions of SQL Server:

| Component       | Requirement   |
|-----------------|---|
| Hard Disk       | SQL Server requires a minimum of 6 GB of available hard-disk space. Disk space requirements will vary with the SQL Server components you install. For more information, see <a href="#">Hard Disk Space Requirements</a> later in this article. For information on supported storage types for data files, see <a href="#">Storage Types for Data Files</a> . |
| Monitor         | SQL Server requires Super-VGA (800x600) or higher resolution monitor.   |
| Internet        | Internet functionality requires Internet access (fees may apply).   |
| Memory *        | Minimum:<br>Express Editions: 512 MB<br>All other editions: 1 GB<br>Recommended:<br>Express Editions: 1 GB<br>All other editions: At least 4 GB and should be increased as database size increases to ensure optimal performance.   |
| Processor Speed | Minimum: x64 Processor: 1.4 GHz<br>Recommended: 2.0 GHz or faster   |
| Processor Type  | x64 Processor: AMD Opteron, AMD Athlon 64, Intel Xeon with Intel EM64T support, Intel Pentium IV with EM64T support   |



## Requirement and Recommendations

### Note

Installation of SQL Server is supported on x64 processors only. It is no longer supported on x86 processors.

The minimum memory required for installing the Data Quality Server component in Data Quality Services (DQS) is 2 GB of RAM, which is different from the SQL Server minimum memory requirement.





# SQL Server 2019: Hardware and software requirements

## Software requirements

The following requirements apply to all installations:



| Component        | Requirement   |
|------------------|---|
| Operating system | Windows 10 TH1 1507 or greater<br>Windows Server 2016 or greater  |
| .NET Framework   | Minimum operating systems includes minimum .NET framework.  |
| Network Software | Supported operating systems for SQL Server have built-in network software. Named and default instances of a stand-alone installation support the following network protocols: Shared memory, Named Pipes, and TCP/IP. |

## Operating System support

The following table shows which editions of SQL Server 2019 are compatible with which versions of Windows:



| SQL Server edition:            | Enterprise | Developer | Standard | Web | Express |
|--------------------------------|------------|-----------|----------|-----|---------|
| Windows Server 2019 Datacenter | Yes        | Yes       | Yes      | Yes | Yes     |
| Windows Server 2019 Standard   | Yes        | Yes       | Yes      | Yes | Yes     |
| Windows Server 2019 Essentials | Yes        | Yes       | Yes      | Yes | Yes     |
| Windows Server 2016 Datacenter | Yes        | Yes       | Yes      | Yes | Yes     |
| Windows Server 2016 Standard   | Yes        | Yes       | Yes      | Yes | Yes     |
| Windows Server 2016 Essentials | Yes        | Yes       | Yes      | Yes | Yes     |
| Windows 10 Enterprise          | No         | Yes       | Yes      | No  | Yes     |
| Windows 10 Professional        | No         | Yes       | Yes      | No  | Yes     |
| Windows 10 Home                | No         | Yes       | Yes      | No  | Yes     |

## Server Core support

Installing SQL Server 2019 on Server Core mode is supported by the following editions of Windows Server:

|                          |
|--------------------------|
| Windows Server 2019 Core |
| Windows Server 2016 Core |



## WOW64 support



WOW64 (Windows 32-bit on Windows 64-bit) is a feature of 64-bit editions of Windows that enables 32-bit applications to run natively in 32-bit mode.

Applications function in 32-bit mode, even though the underlying operating system is a 64-bit operating system.

WOW64 is not supported for SQL Server installations. However, Management Tools are supported in WOW64.



## Features supported on 32-bit client Operating Systems

Windows client operating systems, for example Windows 10 and Windows 8.1 are available as 32-bit or 64-bit architectures. All SQL Server features are supported on 64-bit client operating systems. On supported 32-bit client operating systems Microsoft supports the following features:

- ✓ Data Quality Client
- ✓ Client Tools Connectivity
- ✓ Integration Services
- ✓ Client Tools Backwards Compatibility
- ✓ Client Tools SDK
- ✓ Documentation Components
- ✓ Distributed Replay Components
- ✓ Distributed Replay Controller
- ✓ Distributed Replay Client
- ✓ SQL Client Connectivity SDK

Windows Server 2008 R2 and later server operating systems are not available as 32-bit architectures. All supported server operating systems are only available as 64-bit. All features are supported on 64-bit server operating systems.

## Cross-Language support



The English-language version of SQL Server is supported on all localized versions of operating systems.

Localized versions of SQL Server are supported on localized operating systems with the corresponding language or on English-language versions of supported operating systems by using the Windows Multilingual User Interface Pack (MUI) settings.

Localized versions of SQL Server can only be upgraded to localized versions of the same language, and cannot be upgraded to the English-language version.

Localized versions of SQL Server can also be installed side by side with English-language instances of SQL Server.





## Network Protocols and Network Libraries

A server can listen on, or monitor, multiple network protocols at one time. However, each protocol must be configured. If a particular protocol is not configured, the server cannot listen on that protocol. After installation, you can change the protocol configurations using the SQL Server Configuration Manager.

### Default SQL Server Network Configuration

A default instance of SQL Server is configured for TCP/IP port 1433, and named pipe `\\.\pipe\sql\query`. SQL Server named instances are configured for TCP dynamic ports, with a port number assigned by the operating system.

If you cannot use dynamic port addresses (for example, when SQL Server connections must pass through a firewall server configured to pass through specific port addresses). Select an unassigned port number. Port number assignments are managed by the Internet Assigned Numbers Authority and are listed at <https://www.iana.org>.

To enhance security, network connectivity is not fully enabled when SQL Server is installed. To enable, disable, and configure network protocols after Setup is complete, use the SQL Server Network Configuration area of the SQL Server Configuration Manager.

## Disk space requirements



During installation of SQL Server, Windows Installer creates temporary files on the system drive. Before you run Setup to install or upgrade SQL Server, verify that you have at least 6.0 GB of available disk space on the system drive for these files. This requirement applies even if you install SQL Server components to a non-default drive.

Actual hard disk space requirements depend on your system configuration and the features that you decide to install. The following table provides disk space requirements for SQL Server components.

## Disk space requirements



| Feature  | Disk space requirement |
|--|------------------------|
| Database Engine and data files, Replication, Full-Text Search, and Data Quality Services | 1480 MB                |
| Database Engine (as above) with R Services (In-Database)                                 | 2744 MB                |
| Database Engine (as above) with PolyBase Query Service for External Data                 | 4194 MB                |
| Analysis Services and data files   | 698 MB                 |
| Reporting Services   | 967 MB                 |
| Microsoft R Server (Standalone)  | 280 MB                 |
| Reporting Services - SharePoint  | 1203 MB                |

| Feature  | Disk space requirement |
|--|------------------------|
| Reporting Services Add-in for SharePoint Products  | 325 MB                 |
| Data Quality Client  | 121 MB                 |
| Client Tools Connectivity  | 328 MB                 |
| Integration Services   | 306 MB                 |
| Client Components (other than SQL Server Books Online components and Integration Services tools) | 445 MB                 |
| Master Data Services   | 280 MB                 |
| SQL Server Books Online Components to view and manage help content*                              | 27 MB                  |
| All Features   | 8030 MB                |



## Storage Types for Data Files

The supported storage types for data files are:

### Local Disk

- ❖ SQL Server currently supports disk drives that have standard native sector sizes of 512 bytes and 4 KB. Hard disks with sector sizes larger than 4 KB may cause errors when attempting to store SQL Server data files on them. See [Hard disk drive sector-size support boundaries in SQL Server](#) for more information on hard disk sector-size support in SQL Server
- ❖ SQL Server failover cluster installation supports Local Disk only for installing the tempdb files. Ensure that the path specified for the tempdb data and log files is valid on all the cluster nodes. During failover, if the tempdb directories are not available on the failover target node, the SQL Server resource will fail to come online.

## Storage Types for Data Files

The supported storage types for data files are:

- ❖ Shared Storage
- ❖ [Storage Spaces Direct \(S2D\)](#)
- ❖ SMB File Share
  - ❖ SMB storage is not supported for Analysis Services data files for either standalone or clustered installations. Use direct attached storage, a storage area network, or S2D instead.
  - ❖ SMB storage can be hosted by a Windows File Server or a third-party SMB storage device. If Windows File Server is used, the Windows File Server version should be 2008 or later. For more information about installing SQL Server using SMB file share as a storage option, see [Install SQL Server with SMB Fileshare as a Storage Option](#).



## Storage Types for Data Files

The supported storage types for data files are:

- ❖ Storage Spaces Direct (S2D)

Storage Spaces Direct uses industry-standard servers with local-attached drives to create highly available, highly scalable software-defined storage at a fraction of the cost of traditional SAN or NAS arrays. Its converged or hyper-converged architecture radically simplifies procurement and deployment, while features such as caching, storage tiers, and erasure coding, together with the latest hardware innovations such as RDMA networking and NVMe drives, deliver unrivaled efficiency and performance.

Storage Spaces Direct is included in Windows Server 2019 Datacenter, Windows Server 2016 Datacenter



## Installing SQL Server on a Domain Controller



For security reasons, we recommend that you do not install SQL Server on a domain controller. SQL Server Setup will not block installation on a computer that is a domain controller, but the following limitations apply:

- ❖ You cannot run SQL Server services on a domain controller under a local service account.
- ❖ After SQL Server is installed on a computer, you cannot change the computer from a domain member to a domain controller. You must uninstall SQL Server before you change the host computer to a domain controller.
- ❖ After SQL Server is installed on a computer, you cannot change the computer from a domain controller to a domain member. You must uninstall SQL Server before you change the host computer to a domain member.
- ❖ SQL Server failover cluster instances are not supported where cluster nodes are domain controllers.
- ❖ SQL Server is not supported on a read-only domain controller. SQL Server Setup cannot create security groups or provision SQL Server service accounts on a read-only domain controller. In this scenario, Setup will fail.



## Installing SQL Server on a Domain Controller



### ❖ Note

This restriction also applies to installations on domain member nodes. A SQL Server failover cluster instance is not supported in an environment where only a read-only domain controller is accessible.

### ❖ Note

This restriction also applies to installations on domain member nodes.





# Planning a SQL Server Installation

Vira Security Solutions

# Planning a SQL Server Installation



To install SQL Server, follow these steps:

Review installation requirements, system configuration checks, and security considerations for a SQL Server installation.

Run SQL Server Setup to install or upgrade to a later version. Before upgrading, review Upgrade SQL Server.

Use SQL Server utilities to configure SQL Server.

Regardless of the installation method, you are required to confirm acceptance of the software license terms as an individual or on behalf of an entity, unless your use of the software is governed by a separate agreement such as a Microsoft volume licensing agreement or a third-party agreement with an ISV or OEM.

The license terms are displayed for review and acceptance in the Setup user interface. Unattended installations (using the /Q or /QS parameters) must include the /IAcceptSQLServerLicenseTerms parameter. Download and review the license terms separately at Microsoft SQL Server License Terms and Information. For volume licensing terms, see Licensing Terms and Documentation. For older versions of SQL Server, see Microsoft Software License Terms.

## Install SQL Server from the Command Prompt



Installing a new instance of SQL Server at the command prompt enables you to specify the features to install and how they should be configured. You can also specify silent, basic, or full interaction with the Setup user interface.

To install from the command prompt, open an administrative command prompt and navigate to where setup.exe is located within the SQL Server setup media. Run the setup.exe command, along with the required and optional parameters that accomplish what you're trying to do:

```
C:\SQLMedia\SQLServer2019> setup.exe /[Option] /[Option] = {value}
```

The following example installs the SQL Server database engine, SQL Server Analysis Services, SQL Server Integration Services, and SQL Server tools in quiet mode:



## LAB and Exercises

Graphical Installation

Install SQL Server 2019

Install SQL Server Management Studio

## Install SQL Server from the Command Prompt



```
C:\SQLMedia\SQLServer2019> setup.exe /Q /IACCEPTSQLSERVERLICENSETERMS  
/ACTION="install" /PID="AAAAA-BBBBB-CCCCC-DDDDD-EEEE" /FEATURES=SQL,AS,IS,Tools  
/INSTANCENAME=MSSQLSERVER /SQLSVCACCOUNT="MyDomain\MyAccount"  
/SQLSVCPASSWORD="*****"  
/SQLSYSADMINACCOUNTS="MyDomain\MyAccount "  
/AGTSVCACCOUNT="MyDomain\MyAccount" /AGTSVCPASSWORD="*****"  
/ASSVCACCOUNT="MyDomain\MyAccount" /ASSVCPASSWORD="*****"  
/ISSVCAccount="MyDomain\MyAccount" /ISSVCPASSWORD="*****"  
/ASSYSADMINACCOUNTS="MyDomain\MyAccount"
```



## Install SQL Server from the Command Prompt



C:\SQLMedia\SQLServer2019> setup.exe /configurationfile=configfile.ini  
/IACCEPTSQLSERVERLICENSETERMS=true /QS

/QS or /QUIETSIMPLE : Specifies that Setup runs and shows progress through the UI, but does not accept any input or show any error messages.

## Upgrading from previous versions

### Upgrade SQL Server



Edition upgrades are supported as well

Only same edition or lower to higher edition upgrades are supported

- Express to any edition

- Small Business to Standard

- Web to Web, Standard, or Enterprise

- Workgroup to Standard or Enterprise

- Standard to Standard or Enterprise

- BI Edition to Enterprise only

- Enterprise to Enterprise only

- Developer can only be upgraded to a newer Developer edition



## Upgrade SQL Server



The SQL Server Installation Wizard provides a single feature tree for an in-place upgrade of SQL Server components to the latest version of SQL Server.

### **Warning**

When you upgrade SQL Server, the previous version of SQL Server will be overwritten and will no longer exist on your computer.

Before upgrading, back up SQL Server databases and other objects associated with the previous SQL Server instance.



# Upgrade SQL Server

## Prerequisites

You must run Setup as an administrator. If you install SQL Server from a remote share, you must use a domain account that has read and execute permissions on the remote share, and is a local administrator.

### Warning

Be aware that you cannot change the features to be upgraded, and you cannot add features during the upgrade operation.



## Upgrade SQL Server



If you are upgrading the Database Engine, perform the following tasks, as appropriate for your environment:

- ❖ Back up all SQL Server database files from the instance to be upgraded, so that you can restore them, if it is required.
- ❖ Run the appropriate Database Console Commands (DBCC) on databases to be upgraded to ensure that they are in a consistent state.
- ❖ Estimate the disk space that is required to upgrade SQL Server components, in addition to user databases.

## Upgrade SQL Server



- ❖ Ensure that existing SQL Server system databases - master, model, msdb, and tempdb - are configured to autogrow, and ensure that they have sufficient hard disk space.
- ❖ Ensure that all database servers have logon information in the master database. This is important for restoring a database, as system logon information resides in master.
- ❖ Disable all startup stored procedures, as the upgrade process will stop and start services on the SQL Server instance being upgraded. Stored procedures processed at startup time might block the upgrade process.

## Upgrade SQL Server



- ❖ When upgrading instances of SQL Server where SQL Server Agent is enlisted in MSX/TSX relationships, upgrade target servers before you upgrade master servers. If you upgrade master servers before target servers, SQL Server Agent will not be able to connect to master instances of SQL Server.
- ❖ Quit all applications, including all services that have SQL Server dependencies. Upgrade might fail if local applications are connected to the instance being upgraded.
- ❖ Make sure that Replication is current and then stop Replication. For detailed steps for performing a rolling upgrade in a replicated environment, see [Upgrade Replicated Databases](#).

## SQL Server Release Services

### Upgrade SQL Server



<https://docs.microsoft.com/en-us/archive/blogs/sqlreleaseservices/>

<https://techcommunity.microsoft.com/t5/sql-server/bg-p/SQLServer/label-name/SQLReleases>


# SQL Server Release Services

## Upgrade SQL Server



← → ↻ 🔒 techcommunity.microsoft.com/t5/sql-server/bg-p/SQLServer/label-name/SQLReleases ☆ 🌐 🌐 | D ⋮


☰ 🔍 Microsoft Sign In 👤



### Cumulative Update #20 for SQL Server 2017 RTM

HristinaSQL on 04-09-2020 09:34 PM


Cumulative Update #20 for SQL Server 2017 RTM The 20th cumulative update release for SQL Server 2017 RTM is now availabl...



### Cumulative Update #4 for SQL Server 2019 RTM

SQLTom on 03-31-2020 10:43 AM


The 4th cumulative update release for SQL Server 2019 RTM is now available for download at the Microsoft Downloads site....



### Cumulative Update #3 for SQL Server 2019 RTM


SQLTom on 03-12-2020 01:59 PM

The 3rd cumulative update release for SQL Server 2019 RTM is now available for download at the Microsoft Downloads site....




1,919 👁

### Cumulative Update #12 for SQL Server 2016 SP2



1,913 👁

### Cumulative Update #2 for SQL Server 2019 RTM



1,961 👁

### Security Update for SQL Server 2016 SP2 CU11

techcommunity.microsoft.com/t5/sql-server/.../1225793

## Different methods to identify the SQL Server version number



**RTM:** **RTM** is the acronym of release to manufacturing. It is also known as “going gold”. This term is used when the **product is ready to be delivered**. This build may be digitally signed that allows users to verify the authenticity of software which they had purchased. RTM is the precedence of the General Availability (GA).

**Service Pack:** A Service Pack (SP) is a **collection of updates**, hotfixes or it can be the enhancement to the existing software program. Prior to SQL Server 2017, Microsoft often released service packs based on the issues and feedbacks reported by the users. The service pack is considered a stable version and usually released after one or two years after the product's release. Service packs are delivered in the single installable package.

**Cumulative Update:** Cumulative Update is a rollup of multiple hotfixes. Cumulative updates are tested as a group. When you're building a new SQL Server from scratch, then it is advisable to apply all the recent service packs and cumulative updates.



The following is the list of SQL Server version number, release type, and release date:



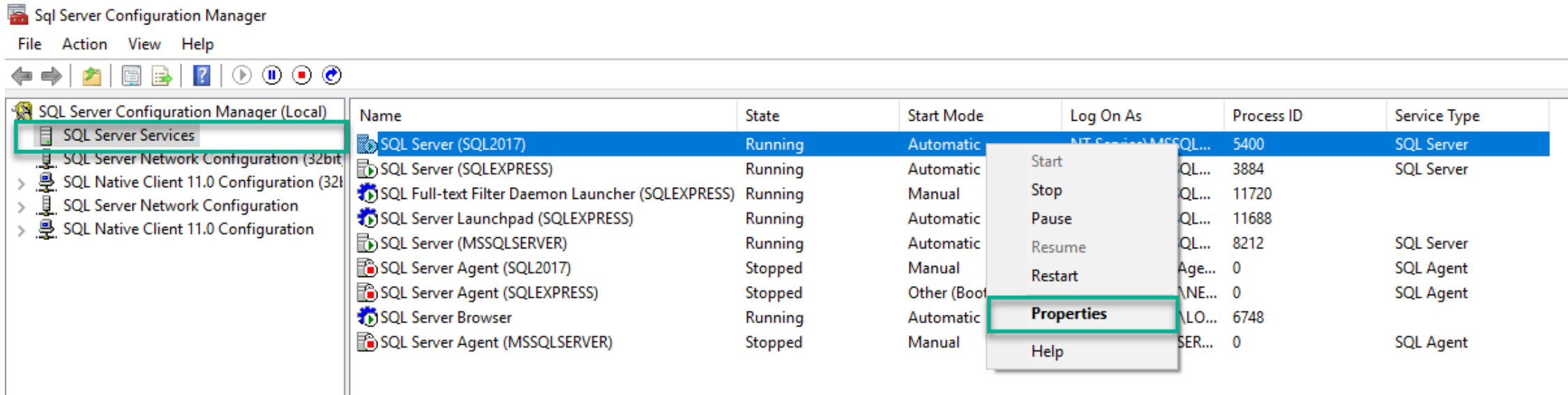
| SQL Server release | Version Number | Type  | Release Date |
|--------------------|----------------|---|--------------|
| SQL Server 2019    | 15.0.2000.5    | RTM   | 2019-11-04   |
|                    | 14.0.1000.169  | RTM   | 2017-10-02   |
| SQL Server 2017    | CU 17          | Cumulative Update (Note: After SQL Server 2017, Microsoft does not release the service packs. | 2019-10-08   |
| SQL Server 2016    | 13.0.1601.5    | RTM   | 2016-06-01   |
|                    | 13.0.4001.0    | Service pack 1  | 2016-11-16   |
|                    | 13.0.5026.0    | Service pack 2  | 2018-04-24   |



## How to identify the SQL Server version number

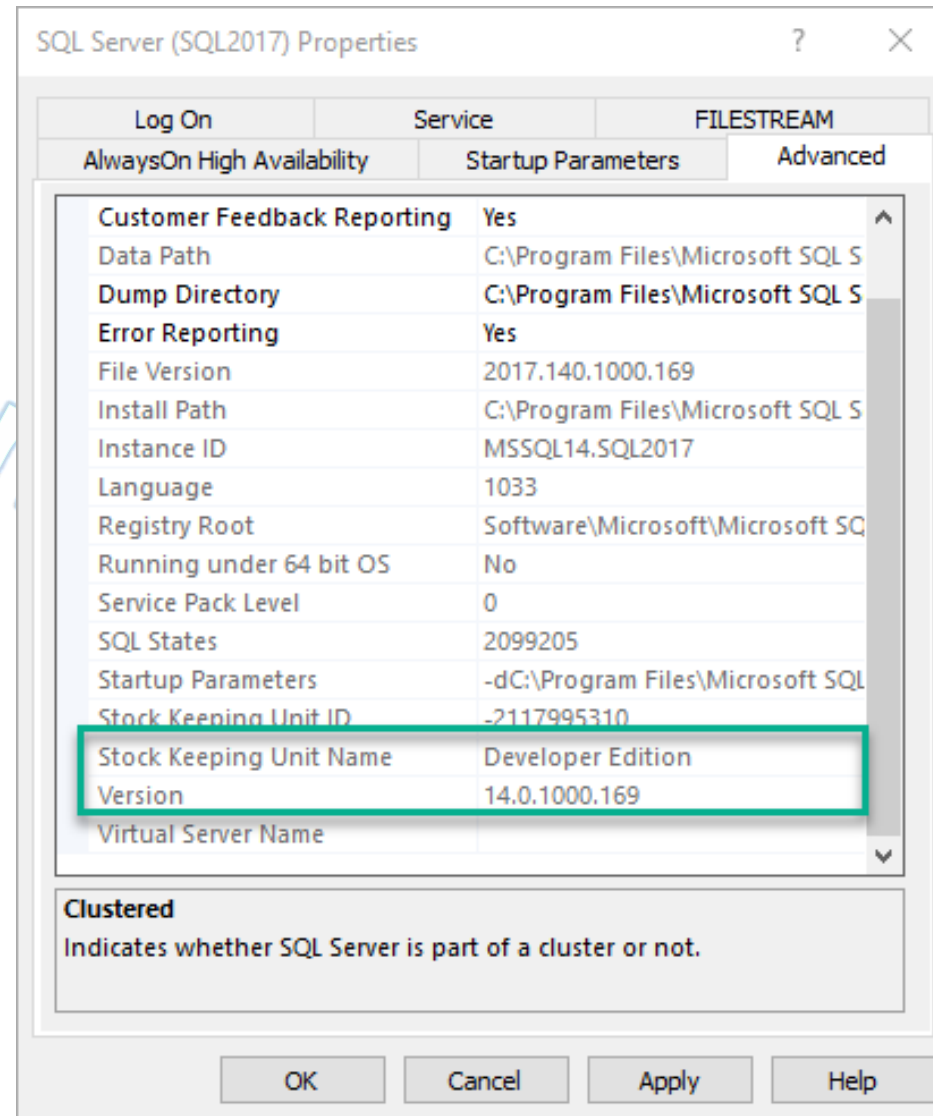


To obtain the SQL Server version numbers, open SQL Server Configuration Manager. Click on **SQL Server Services**. Right-click on SQL Server instance, **SQLServer (SQL2017)** in my case, and choose **Properties**. See the following screenshot:



## How to identify the SQL Server version number

In **Properties** dialog box, click on the **Advanced** tab. You can see the SQL Server edition in **Stock Keeping Unit Name** text box and its version number in the **Version** text box. See the following image:



## Overview of SQL Server Security



A defense-in-depth strategy, with overlapping layers of security, is the best way to counter security threats. SQL Server provides a security architecture that is designed to allow database administrators and developers to create secure database applications and counter threats. Each version of SQL Server has improved on previous versions of SQL Server with the introduction of new features and functionality. However, security does not ship in the box. Each application is unique in its security requirements. Developers need to understand which combination of features and functionality are most appropriate to counter known threats, and to anticipate threats that may arise in the future.

A SQL Server instance contains a hierarchical collection of entities, starting with the server. Each server contains multiple databases, and each database contains a collection of securable objects. Every SQL Server securable has associated *permissions* that can be granted to a *principal*, which is an individual, group or process granted access to SQL Server.

## Overview of SQL Server Security



The SQL Server security framework manages access to securable entities through *authentication* and *authorization*.

- ❖ Authentication is the process of logging on to SQL Server by which a principal requests access by submitting credentials that the server evaluates. Authentication establishes the identity of the user or process being authenticated.
- ❖ Authorization is the process of determining which securable resources a principal can access, and which operations are allowed for those resources.

# Overview of SQL Server Security



SQL Server Security Architecture

Server Logins

Built-in Server Roles and Custom Server Roles

Database Users

Built-in Database Roles and Custom Database Roles

Object level permissions

# SQL Server Security Architecture



The Security architecture of SQL server

Secure by default

Encryption

CLR security

## SQL Server Security Architecture



- ❖ The architecture of SQL server is designed to provide overlapping layers of security
- ❖ Clients are authenticated
- ❖ Clients are authorized to perform actions or access resources
- ❖ There is a hierarchy of access based on object relations and permissions
- ❖ Data can be encrypted
- ❖ External code written using the SQL CLR can be secured

## SQL Server CLR



The common language runtime (CLR) is the heart of the Microsoft .NET Framework and provides the execution environment for all .NET Framework code. Code that runs within the CLR is referred to as managed code. The CLR provides various functions and services required for program execution, including just-in-time (JIT) compilation, allocating and managing memory, enforcing type safety, exception handling, thread management, and security. With the CLR hosted in Microsoft SQL Server (called CLR integration), you can author stored procedures, triggers, user-defined functions, user-defined types, and user-defined aggregates in managed code. Because managed code compiles to native code prior to execution, you can achieve significant performance increases in some scenarios.



# Authentication



Authentication is the process of determining the identity for a client  
SQL server provides two authentication method:

- Windows Authentication
- SQL Server Authentication

## Authentication



- ✓ A user is authenticated to perform an action or access an object when granted permissions
- ✓ SQL Server is “Secure by default”
- ✓ The principle of least privilege should be adhered to when authorizing users
- ✓ There are two ways to assign permissions
  - Roles (Server or Databases)
  - Individual Permission statement
- ✓ Ownership chains can confuse the issue, but they adhere to these same principles

## Authentication



- ✓ SQL Server provides a number of encryption mechanisms to protect data
- ✓ Transparent Data Encryption (TDE)
  - Always Encrypted (AE)
  - Column level encryption functions

## Server Logins



- ✓ SQL server uses logins to connect to the server instance
- ✓ The login does not grant rights to a database or data, only the instance
- ✓ Windows AD and local accounts can be added as logins
- ✓ SQL Server can also manage its own logins and validate passwords

## Server Logins



- ✓ Logins are also known as Server level principals
- ✓ Information about logins is located in `master.sys.server_principals`
- ✓ SQL Server logins are also located in `master.sys.sql_logins`



## LAB and Exercises

Add new user in Management Studio, and describe about logins

## Server-Level Roles



SQL Server provides server-level roles to help you manage the permissions on a server. These roles are security principals that group other principals. Server-level roles are server-wide in their permissions scope. (Roles are like *groups* in the Windows operating system.)

Fixed server roles are provided for convenience and backward compatibility. Assign more specific permissions whenever possible.

SQL Server provides nine fixed server roles. The permissions that are granted to the fixed server roles (except **public**) cannot be changed. Beginning with SQL Server 2012 (11.x), you can create user-defined server roles and add server-level permissions to the user-defined server roles.

You can add server-level principals ( SQL Server logins, Windows accounts, and Windows groups) into server-level roles. Each member of a fixed server role can add other logins to that same role. Members of user-defined server roles cannot add other server principals to the role.

## Server-Level Roles



| Fixed server-level role | Description   |
|-------------------------|---|
| sysadmin                | Members of the sysadmin fixed server role can perform any activity in the server.   |
| serveradmin             | Members of the serveradmin fixed server role can change server-wide configuration options and shut down the server.   |
| securityadmin           | <p>Members of the securityadmin fixed server role manage logins and their properties. They can GRANT, DENY, and REVOKE server-level permissions. They can also GRANT, DENY, and REVOKE database-level permissions if they have access to a database. Additionally, they can reset passwords for SQL Server logins.</p> <p>IMPORTANT: The ability to grant access to the Database Engine and to configure user permissions allows the security admin to assign most server permissions. The securityadmin role should be treated as equivalent to the sysadmin role.</p> |
| processadmin            | Members of the processadmin fixed server role can end processes that are running in an instance of SQL Server.  |



## Server-Level Roles



| Fixed server-level role | Description   |
|-------------------------|---|
| <b>setupadmin</b>       | Members of the setupadmin fixed server role can add and remove linked servers by using Transact-SQL statements. (sysadmin membership is needed when using Management Studio.)   |
| <b>bulkadmin</b>        | Members of the bulkadmin fixed server role can run the BULK INSERT statement.   |
| <b>diskadmin</b>        | The diskadmin fixed server role is used for managing disk files.  |
| <b>dbcreator</b>        | Members of the dbcreator fixed server role can create, alter, drop, and restore any database.   |
| <b>public</b>           | <p>Every SQL Server login belongs to the public server role. When a server principal has not been granted or denied specific permissions on a securable object, the user inherits the permissions granted to public on that object. Only assign public permissions on any object when you want the object to be available to all users. You cannot change membership in public.</p> <p>Note: public is implemented differently than other roles, and permissions can be granted, denied, or revoked from the public fixed server roles.</p> |

## Server-Level Roles



- ✓ Server roles are used to manage the rights and privileges of a group of logins
- ✓ A server role is similar to how an Active Directory Group works
- ✓ Each of these roles conveys system wide privileges and only system administrators or trusted logins should be members of these roles

## Server-Level Roles



- ✓ There are a number of built-in server roles
- ✓ A login can be a member of multiple roles
- ✓ Multiple logins can be a member of role



## LAB and Exercises

Describe Server Roles in Management Studio

Add new member to a role and also check related scripts about this process

## user-defined server role



- ✓ Because the built-in server roles may not meet your needs, user-defined server roles can be created
- ✓ These are roles that operate at the server level
- ✓ The system administrator can create these roles, but they have no members

## user-defined server role



- ✓ There are two type of server roles:
  - Build in server roles
  - Custom server roles
- ✓ A login can be a member of multiple roles
- ✓ A role can be a member of another role

# user-defined server role



## Limitations and Restrictions

Server roles cannot be granted permission on database-level securables. To create database roles, see [CREATE ROLE \(Transact-SQL\)](#).

- Requires CREATE SERVER ROLE permission or membership in the sysadmin fixed server role.
- Also requires IMPERSONATE on the *server\_principal* for logins, ALTER permission for server roles used as the *server\_principal*, or membership in a Windows group that is used as the *server\_principal*.
- When you use the AUTHORIZATION option to assign server role ownership, the following permissions are also required:
  - To assign ownership of a server role to another login, requires IMPERSONATE permission on that login.
  - To assign ownership of a server role to another server role, requires membership in the recipient server role or ALTER permission on that server role.

# user-defined server role



## Security Permissions

- Requires CREATE SERVER ROLE permission or membership in the sysadmin fixed server role.
- Also requires IMPERSONATE on the *server\_principal* for logins, ALTER permission for server roles used as the *server\_principal*, or membership in a Windows group that is used as the *server\_principal*.
- When you use the AUTHORIZATION option to assign server role ownership, the following permissions are also required:
  - To assign ownership of a server role to another login, requires IMPERSONATE permission on that login.
  - To assign ownership of a server role to another server role, requires membership in the recipient server role or ALTER permission on that server role.



# Using SQL Server Management Studio



## To create a new server role

In Object Explorer, expand the server where you want to create the new server role. Expand the **Security** folder.

Right-click the **Server Roles** folder and select **New Server Role...**

In the **New Server Role** -server\_role\_name dialog box, on the **General** page, enter a name for the new server role in the **Server role name** box.

In the **Owner** box, enter the name of the server principal that will own the new role. Alternately, click the ellipsis (...) to open the **Select Server Login or Role** dialog box.

Under **Securables**, select one or more server-level securables. When a securable is selected, this server role can be granted or denied permissions on that securable.

In the **Permissions: Explicit** box, select the check box to grant, grant with grant, or deny permission to this server role for the selected securables. If a permission cannot be granted or denied to all of the selected securables, the permission is represented as a partial select.

On the **Members** page, use the **Add** button to add logins that represent individuals or groups to the new server role.

A user-defined server role can be a member of another server role. On the **Memberships** page, select a check box to make the current user-defined server role a member of a selected server role.

Click **OK**.

# Using Transact-SQL



## To create a new server role

In **Object Explorer**, connect to an instance of Database Engine.

On the Standard bar, click **New Query**.

Copy and paste the following example into the query window and click **Execute**.

--Creates the server role auditors that is owned the securityadmin fixed server role.

```
USE master; CREATE SERVER ROLE auditors AUTHORIZATION securityadmin; GO
```