PViMS

Deployment Guide

July 2018

USAID
FROM THE AMERICAN PEOPLE

SIAPS
Systems for Improved Access to Pharmaceuticals and Services

This report is made possible by the generous support of the American people through the US Agency for International Development (USAID), under the terms of cooperative agreement number AID-OAA-A-11-00021. The contents are the responsibility of Management Sciences for Health and do not necessarily reflect the views of USAID or the United States Government.

About SIAPS

The goal of the Systems for Improved Access to Pharmaceuticals and Services (SIAPS) Program is to assure the availability of quality pharmaceutical products and effective pharmaceutical services to achieve desired health outcomes. Toward this end, the SIAPS result areas include improving governance, building capacity for pharmaceutical management and services, addressing information needed for decision-making in the pharmaceutical sector, strengthening financing strategies and mechanisms to improve access to medicines, and increasing quality pharmaceutical services.

Recommended Citation

SIAPS Program. 2018. *PViMS Deployment Guide.* Submitted to the US Agency for International Development by the Systems for Improved Access to Pharmaceuticals and Services (SIAPS) Program. Arlington, VA: Management Sciences for Health.

Systems for Improved Access to Pharmaceuticals and Services
Pharmaceuticals and Health Technologies Group
Management Sciences for Health
4301 North Fairfax Drive, Suite 400
Arlington, VA 22203 USA
Telephone: 703.524.6575
Fax: 703.524.7898
E-mail: phtmis@msh.org
Website: www.siapsprogram.org

Contents

1 Introduction

1.1 Document Overview

This guide describes the environment required for successful deployment of the PViMS system.

1.2 Purpose of the Document

The purpose of this document is to comprehensively describe the necessary preparation to successfully deploy the PViMS system. This includes minimum hardware and software requirements, security measures that require strict adherence, and step by step instructions on the deployment of the solution.

Once the hardware and software requirements have been satisfied, this guide details the following:

- Deployment of the data tier--the system is database agnostic but this guide assumes the use of Microsoft's MS SQL Server (Enterprise or Express Editions) as a scalable backend database
- Deployment of the application tier
- Installation of the application

Please note that the deployment and configuration of the development environment as well as the configuration of the system itself are out of scope for this document and are addressed in the System administration and Programmers' manual documents.

1.3 Audience

This document is written generically to address all technical stakeholders in the implementation of the PViMS system but is specifically meant for support engineers responsible for the deployment of the application on any heterogeneous network.

2 Preparation

This section describes in detail the preparation required for the successful deployment of PViMS, including the provision of the necessary server hardware and the software requirements that need to be satisfied before deployment of the system commences.

Stage 1: Deployment preparation

- Install and configure hardware as per scenario 1 or 2
- Install and configure software as per scenario 1 or 2
- Configure the network

Stage 2: Implementation

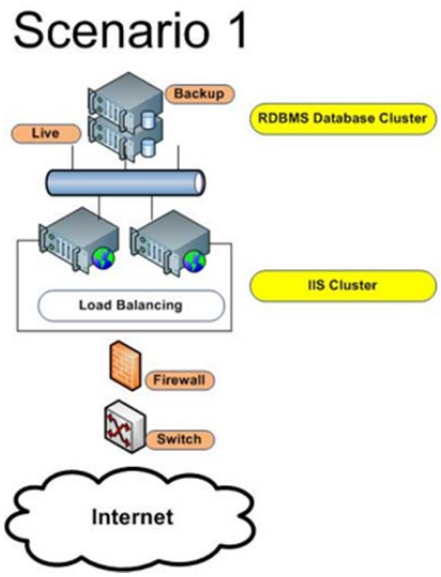
- Deploy the data tier - Security
- Deploy the data tier – Installing MS SQL Server Express
- Deploy the application tier – Security
- Deploy the application tier – Installing IIS

Stage 3: Post-Implementation

- Configuring MS SQL Server Roles
- Create database
- Assign database owner
- Assign database user
- Configuring IIS application pool
- Test connectivity
    - Test MS SQL connectivity
    - Test application connectivity - Internal
    - Test application connectivity – External
- Complete system installation

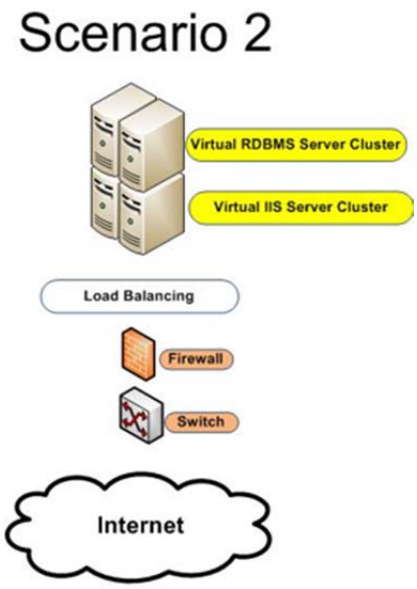
2.1 Deployment Overview

PViMS is a web-based client-server application that can be deployed in a public-facing or intranet environment. It is recommended that PViMS be deployed using one of the deployment scenarios described below:



Application and data tiers deployed across multiple physical servers:

- Between 1 and 2 servers required for application tier (single or load balanced)
- Between 1 and 2 servers required for data tier (live and backup)



Application and data tiers deployed across single physical server with multiple Virtual Machines (VM) configured:

- Between 1 and 2 VMs required for application tier (single or load balanced)
- Between 1 and 2 VMS required for data tier (live and backup)

Please note, it is a fundamental recommendation that the data and application tiers be separated from each other. It is conceivable and possible to combine the application and data tiers on a single server, but this is not recommended as this could effectively expose the data tier directly to the internet and leave your data

vulnerable to being compromised. By separating the data tier from the application tier, you can effectively ensure the data tier is only accessible on a local subnet by the application tier.

2.2 Hardware Requirements

Within each given deployment scenario, it is imperative that the following instructions are adhered to during implementation:

- The database should not be made available publicly on the internet.
- The database should not be installed on the same server as the application server.
- Suitable backup processes are in place for both the web and database servers.
- In the interest of removing single points of failure, backup database and clustered application servers can be configured.
- A hardware load balancer is not required but is recommended if load on the application servers is increased. A load balancer increases scalability.
- A firewall is imperative to protect the application and database servers.
- The firewall should only be configured to allow port 80 traffic to the application server. Direct access to the database server should be restricted through the implementation of firewall policies.

2.2.1 Firewall and Hardware Load Balancer

A firewall is a network security device that monitors incoming and outgoing network traffic and decides whether to allow or block specific traffic based on a defined set of security rules. To ensure PViMS data is not compromised, a suitable hardware firewall must be installed to protect PViMS related traffic.

In addition to enhanced security requirements, Internet-accessible applications often have much higher scale and availability requirements than do intranet-only applications. Internet applications may be accessed by thousands of times more users, while requiring 24x7 operations to accommodate worldwide access. In response to these requirements, hardware load balancers have been developed to meet both the scale and high availability requirements of Internet-accessible applications.

2.2.2 IIS Load Balanced Servers

A server cluster is a group of independent servers that are managed as a single system for higher availability, easier manageability, and greater scalability.

In a Web server environment, server clusters can be defined in two basic ways:

- Active/Active
    - There are multiple independent, redundant servers
    - The load is distributed through round-robin DNS
    - The load is balanced by a load-balancing solution (for example, WLBS)

- Active/Passive
    - Multiple servers are configured to provide a service
    - Only a single server provides the service at any given time
    - Other servers serve as hot-spares in case of a server (service) problem

2.2.3 Hyper-V Virtual Environment

Hyper-V, codenamed Viridian and formerly known as Windows Server Virtualization, is a native hypervisor; it can create virtual machines on x86-64 systems. Starting with Windows 8, **Hyper-V** supersedes Windows Virtual PC as the hardware virtualization component of the client editions of Windows NT.

Hyper-V in Windows Server 2008 and Windows Server 2008 R2 enables you to create a virtualized server computing environment. You can use a virtualized computing environment to improve the efficiency of your computing resources by utilizing more of your hardware resources. This is possible because you use Hyper-V to create and manage virtual machines and their resources. Each virtual machine is a virtualized computer system that operates in an isolated execution environment. This allows you to run multiple operating systems simultaneously on one physical computer.

Scenario 1 Hardware Requirements

Server Configuration	Capacity
1 x Application Server Intel® Xeon® Processor E5530, 4 x Quad-core CPUs 2.4 GHz 1024 KB L2 Cache / 120 GB DDR3-1066 RDIMM Memory 2x160GB physical RAID 1 **1 x Database Server** Intel® Xeon® Processor E5530, 4 x Quad-core CPUs 2.4 GHz 1024 KB L2 Cache / 120 GB DDR3-1066 RDIMM Memory 2x320GB physical RAID 1, 5x1000GB physical RAID 5	**5000 Users** • 50 Analysts • 100 Reporters • 2425 Data Capturers • 2425 Clinicians

1 x Application Server Intel® Xeon® Processor E5530, 2 x Quad-core CPUs 2.4 GHz 1024 KB L2 Cache / 80 GB DDR3-1066 RDIMM Memory 2x160GB physical RAID 1 **1 x RDBMS Server** Intel® Xeon® Processor E5530, 2 x Quad-core CPUs 2.4 GHz 1024 KB L2 Cache / 80 GB DDR3-1066 RDIMM Memory 2x160GB physical RAID 1, 5x500GB physical RAID 5	**2000 Users** • 20 Analysts • 50 Reporters • 965 Data Capturers • 965 Clinicians
1 x Application Server Intel® Xeon® Processor E5530, 2 x Quad-core CPUs 2.4 GHz 1024 KB L2 Cache / 40 GB DDR3-1066 RDIMM Memory 2x80GB physical RAID 1 **1 x RDBMS Server** Intel® Xeon® Processor E5530, 2 x Quad-core CPUs 2.4 GHz 1024 KB L2 Cache / 40 GB DDR3-1066 RDIMM Memory 2x160GB physical RAID 1, 5x250GB physical RAID 5	**500 Users** • 10 Analysts • 40 Reporters • 225 Data Capturers • 225 Clinicians

Scenario 2 Hardware Requirements

Server Configuration	Capacity
1 x Virtual Server Intel® Xeon® Processor E5-2407, 4 x Quad-core CPUs 2.4 GHz 1024 KB L2 Cache / 120 GB DDR3-1333 RDIMM Memory 2x320GB physical RAID 1, 5x1000GB physical RAID 5	**5000 Users** • 50 Analysts • 100 Reporters • 2425 Data Capturers • 2425 Clinicians
1 x Virtual Server Intel® Xeon® Processor E5-2407, 4 x Quad-core CPUs 2.4 GHz 1024 KB L2 Cache / 80 GB DDR3-1333 RDIMM Memory 2x320GB physical RAID 1, 5x500GB physical RAID 5	**2000 Users** • 20 Analysts • 50 Reporters • 965 Data Capturers • 965 Clinicians

1 x Virtual Server Intel® Xeon® Processor E5-2407, 4 x Quad-core CPUs 2.4 GHz 1024 KB L2 Cache / 80 GB DDR3-1333 RDIMM Memory 2x160GB physical RAID 1, 5x250GB physical RAID 5	**500 Users** • 10 Analysts • 40 Reporters • 225 Data Capturers • 225 Clinicians

2.3 Software Requirements

The following software needs to be installed and configured on the application and database server before PViMS can be installed:

Application Server	Database Server	Client Workstation
Windows Server 2012 R2 with latest service pack DotNet framework 4.5 IIS 8.0 Hyper-V for VM installation if scenario 2 selected	Windows Server 2012 R2 with latest service pack Microsoft SQL Server 2008 R2 with latest service pack Hyper-V for VM installation if scenario 2 selected	Google Chrome with latest patch applied Microsoft Office (MS Excel and MS Word)

2.4 Data Tier Security

To mitigate any potential risk for any public facing servers, it is extremely important to consider the following steps and best practices:

Identify the network flow, in terms of requests. If you know the regular network flow the server is supposed to receive and send, then you can allow and check (content/requests inspection) them, while other traffic/flow would be denied by default (by Firewall). This is a network isolation measure that will reduce the risk of a malware spread.
Make sure there is no way to directly request access to your web server, bypassing security filtering layers. There should be at least a 3-layer filter for your web server: • Protocols and sources accepted: firewall (and routers). • Dynamic network traffic inspection: NIPS (Network Intrusion Protection System) that will detect/block malicious network requests. • Application-oriented security: WAF (Web Application Firewall), just next to the web app/site, that will allow you to harden the requests control, and tighten the filter to match the specificities of the web application.
Make sure that clients can't directly send requests to your server (from a TCP point of view), that could

facilitate attacks otherwise. Thus, ensure network isolation, DMZ-minded, by deploying a reverse proxy as a front-end of the web server.
Harden cyphered network communications by taking into account the available implementations of SSL/TLS on the Windows systems being run.
Make sure you have got all the relevant traceability chains: this meaning possible correlation between firewall's, reverse-proxy's, and web server's logs. Please make attention not to only enable "errors" logging, for instance in IIS logs.
Create a back-up of application and database server data, on a regular basis.
Create images of Windows systems, in an integer state, on a regular basis (at least, at deployment time). This may be helpful in case of a security incident, both to return to production mode as quick as possible, and also to investigate.
Infrastructure Audit: firewall rules, NIPS rules, WAF rules, reverse-proxy settings, on a regular basis.
Follow security best practices for application layer products, database layer ones, and web server layer.

2.5 Application Tier Security

The following are a list of recommendations for improving the security of an IIS web server (source technet.microsoft.com):

2.5.1 Installation and Configuration

- Do not run IIS on a domain controller or a backup domain controller. First, there are no local accounts on a domain controller. Local accounts are important to the security of many IIS server installations. Placing an IIS web server and domain controller on the same computer seriously limits your security account options. Second, any new exploit that compromises your web server could also compromise your entire network when the web server and the domain controller are on the same computer.
- Install only the IIS modules you need. IIS 8 is composed of more than 40 modules, which allow you to add modules you need and remove any modules you don't need. If you install only the modules you need, you reduce the surface area that is exposed to potential attacks.
- Periodically remove unused or unwanted modules and handlers. Look for modules and handlers that you no longer use and remove them from your IIS installation. Strive to keep your IIS surface area as small as possible.
- For high volume installations of IIS, run other resource-intensive products like SQL Server or Exchange on separate computers.
- Keep your antivirus software up to date. Install and run the latest version of antivirus software.
- Move the **Inetpub** folder from your system drive to a different drive. By default, IIS 8 sets up the Inetpub folder on your system drive (usually the C drive). If you move the folder to a different partition, you can save space on your system drive and improve security.

2.5.2 Web Application Isolation

- Isolate web applications. Separate different applications into different sites with different application pools.
- Implement the principle of least privilege. Run your worker process as a low privileged identity (virtual application pool identity) that is unique per site.
- Isolate ASP.NET temp folders. Set up a separate ASP.NET temp folder per site and only give access to appropriate process identity.
- Isolate content. Make sure to set an ACL (access control list) on each site root to allow only access to the appropriate process identity.

2.5.3 Authentication

- If you use Windows authentication, turn on extended protection. Extended protection protects against credential relaying and phishing attacks when using Windows authentication.
- Be aware that configuring Anonymous authentication along with another authentication type for the same website can cause authentication problems. If you configure Anonymous authentication and

another authentication type, the result is determined by the order in which the modules run. For example, if Anonymous authentication and Windows authentication are both configured and Anonymous authentication runs first, Windows authentication never runs.

- Disable anonymous access to server directories and resources. When you want to grant a user the access to server directories and resources, use an authentication method that is not anonymous.
- Do not allow anonymous writes to the server. Authenticate the user with a method that is not anonymous before allowing the user to upload anything to your website or FTP site.

2.5.4 Request Filtering

- Ensure that request filtering rules are enabled. Request filters restrict the types of HTTP requests that IIS 8 processes. By blocking specific HTTP requests, request filters help prevent potentially harmful requests from reaching the server. The request filter module scans incoming requests and rejects requests that are unwanted based upon the rules that you set up. Both websites and FTP sites should have the protection that request filter rules provide.
- Ensure that request limits are set to reasonable values. Think carefully about the values you assign to configuration parameters. For example, make sure that an upper limit value is higher than a lower limit value. Otherwise, the filter may never trigger.

2.5.5 Application Pool Identities

- Don't use the built-in service identities (such as Network Service, Local Service, or Local System). For maximum security, application pools should run under the application pool identity that is generated when the application pool is created. The accounts that are built in to IIS are ApplicationPoolIdentity, NetworkService, LocalService, and LocalSystem. The default (recommended) and most secure is ApplicationPoolIdentity.
- Using a custom identity account is acceptable but be sure to use a different account for each application pool.

2.5.6 More Security Practises

- Make periodic backups of the IIS server. Do a complete system-state backup every day or two. Also do it before major software upgrades or configuration changes.
- Limit permissions granted to non-administrators. Look for folders that non-administrators have write permissions and script execution permissions to and remove the permissions.
- Turn on SSL and maintain SSL certificates. Renew the certificate or choose a new certificate for the site. An expired certificate becomes invalid and can prevent users from accessing your site.
- Use SSL when you use Basic authentication. Use Basic authentication with an SSL binding, and make sure that the site or application is set to require SSL. Alternatively, use a different method of authentication. If you use Basic authentication without SSL, credentials are sent in plaintext that might be intercepted by malicious code. If you want to continue using Basic authentication, you need to check

the site bindings to make sure that an HTTPS binding is available for the site, and then configure the site to require SSL.

- When you set feature delegation rules, don’t make rules that are more permissive than the defaults.
- For a classic ASP application, turn off debug mode.

3 Implementation

This section describes in detail the steps required to successfully prepare the data and application tiers.

3.1 Installing MS SQL Server

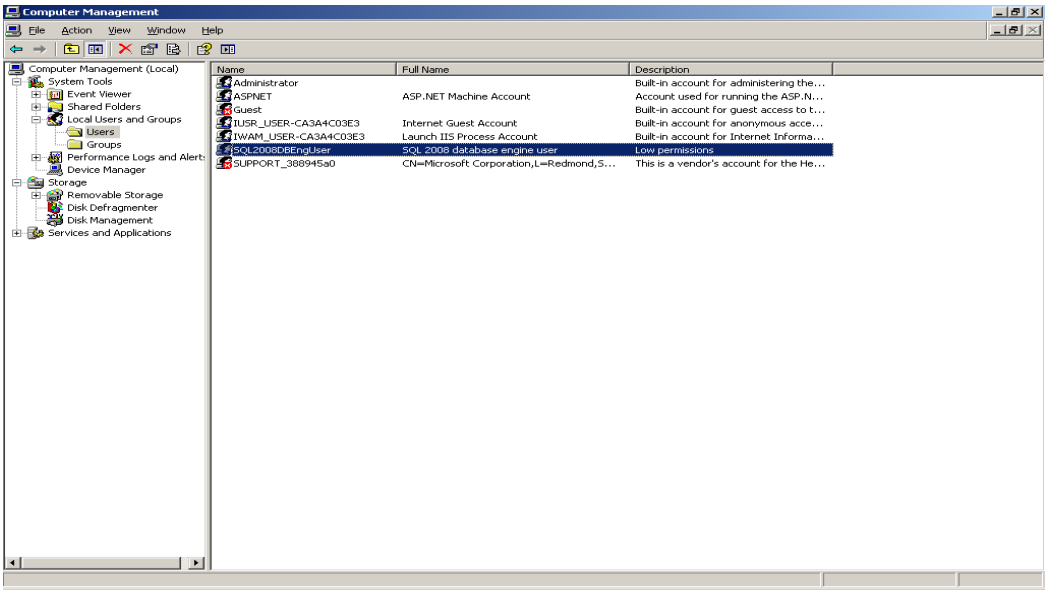
3.1.1 Preparation

The following setup packages and files are required to install SQL Server 2008 Express:

- MS SQL Express 2008 R2 **with tools** https://www.microsoft.com/en-za/download/details.aspx?id=30438

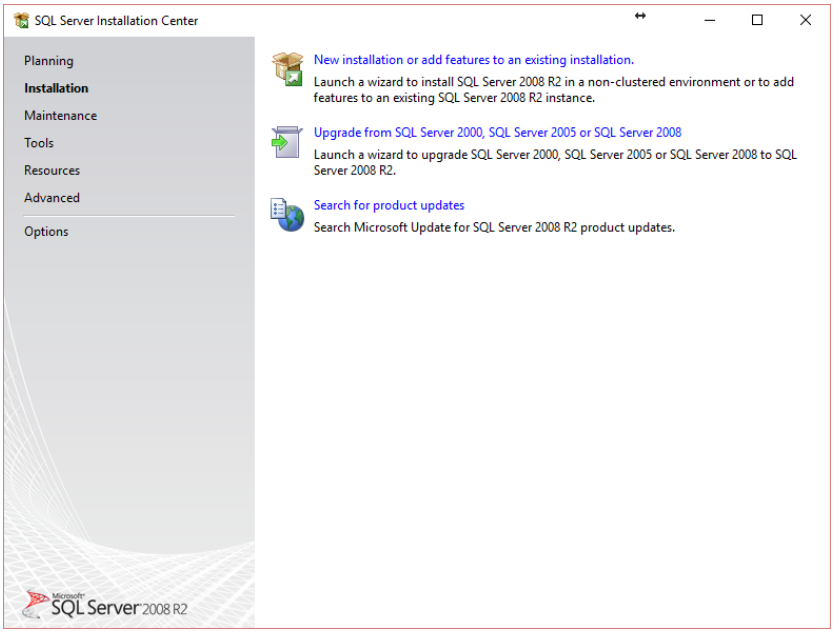
3.1.2 Prerequisites

- Log on to the server as a local administrator.
- Create a folder on the server – **X:\AppData\SQLData** where X is the drive on the server that has been allocated for hosting the database.
- Through Computer Management --> Local Users and Groups configure a SQL service account. Set up a SQL database engine user (SQL2008DBEngUser). This user must not have administrator rights to the workstation/server.

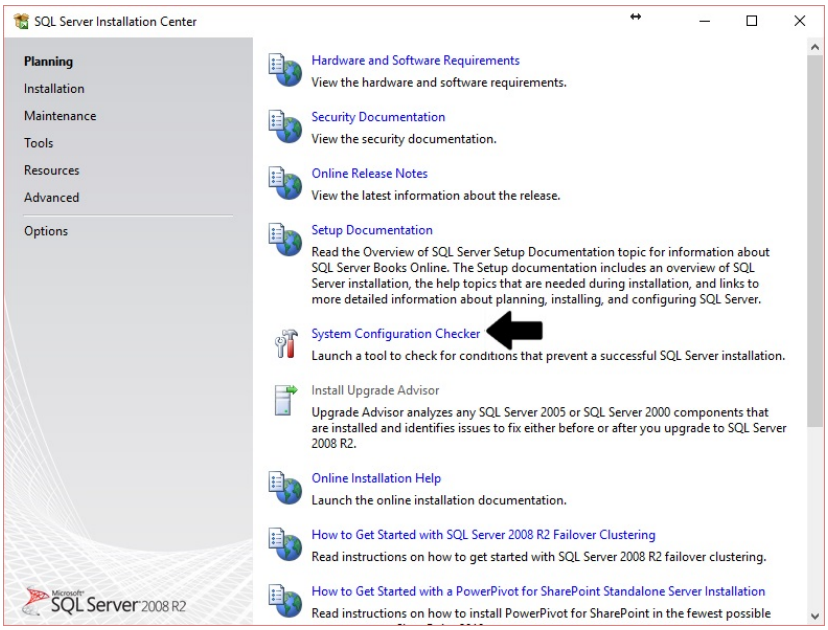


3.1.3 Installing SQL Express 2008 R2

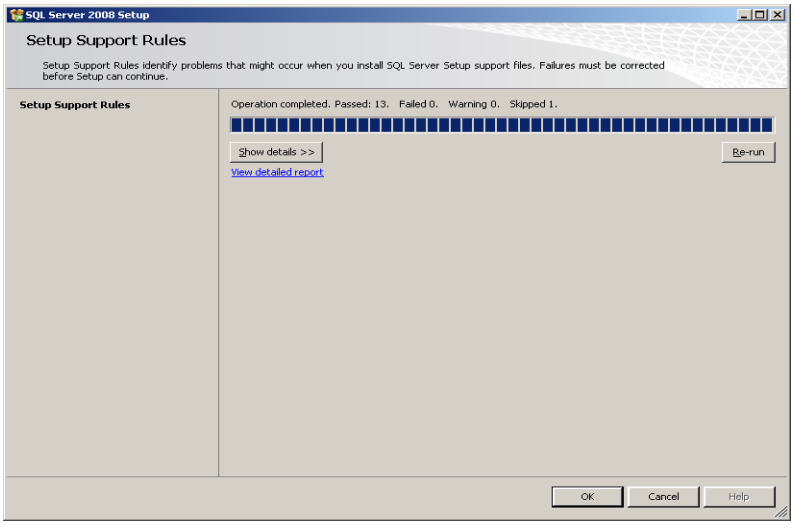
- On first execution of the setup package, you will be presented with the following installation screen:



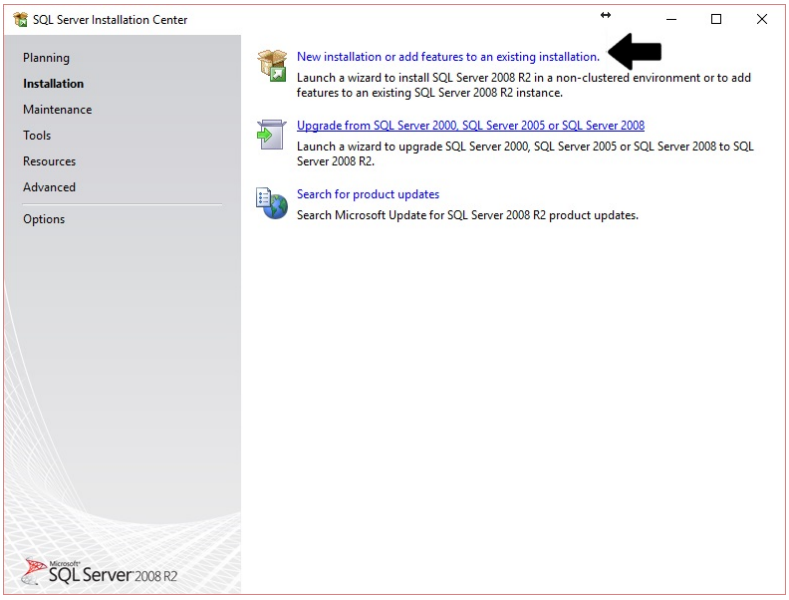
- Select the planning menu option in the left-hand pane
- Select the System Configuration Checker option



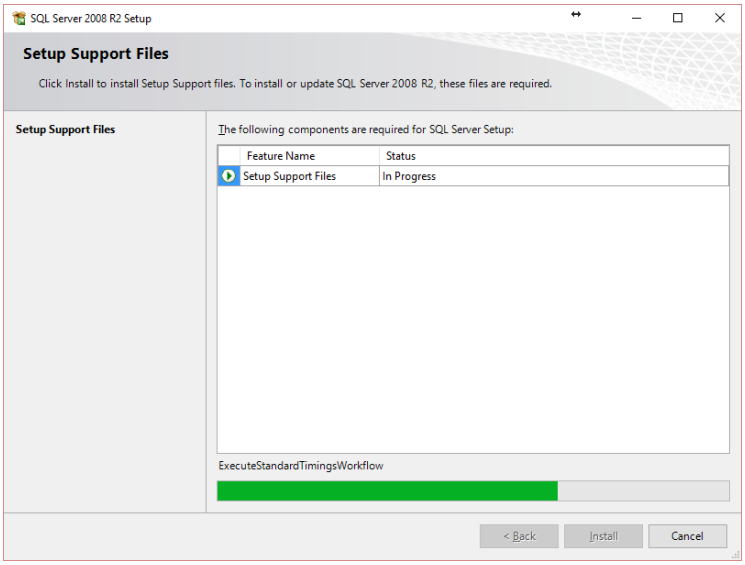
- You will now be prompted with the following status screen:



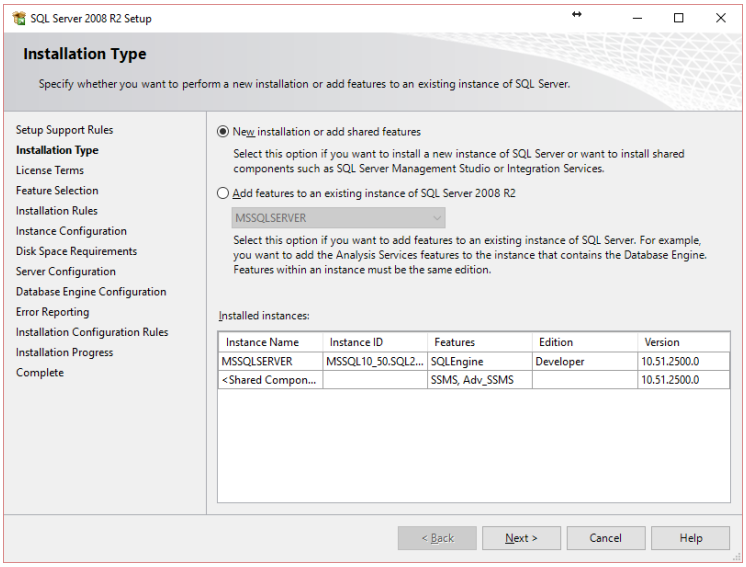
- Ensure no steps have failed. Action any failed steps where necessary.
- Click ok. You will now return to the main menu screen.
- Select the Installation menu option in the left-hand pane.
- Select the New installation or add features to an existing installation option.



- Support files will now be prepared…

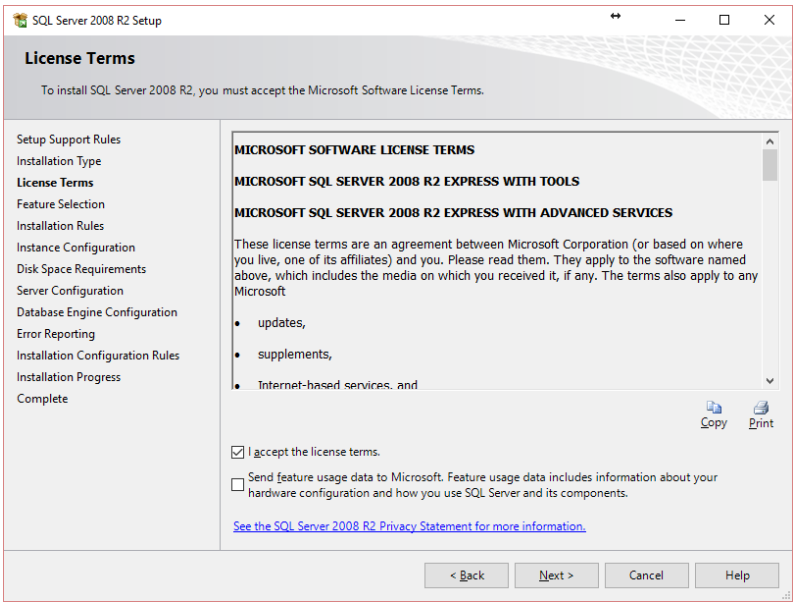


- Once setup files are prepared, you will be presented with an installation type screen…

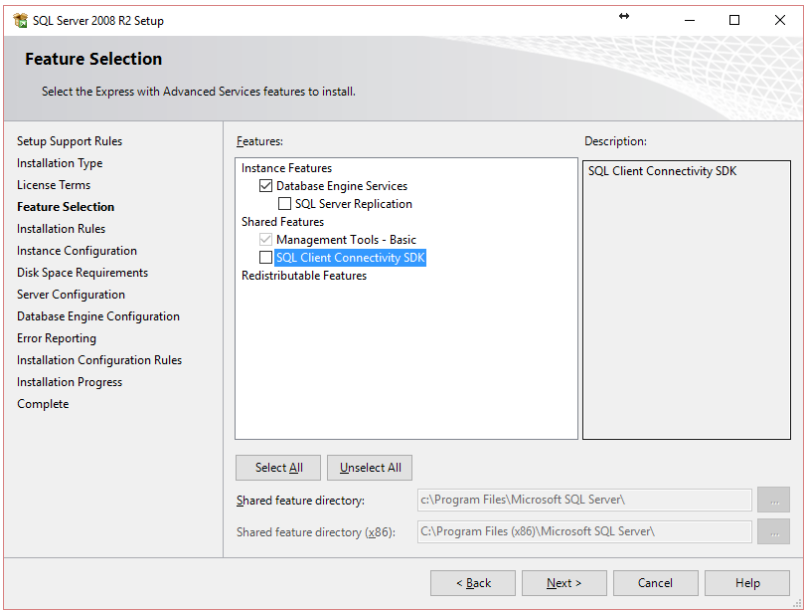


- Ensure New installation is selected and click next…
- You will now be presented with the license terms screen….

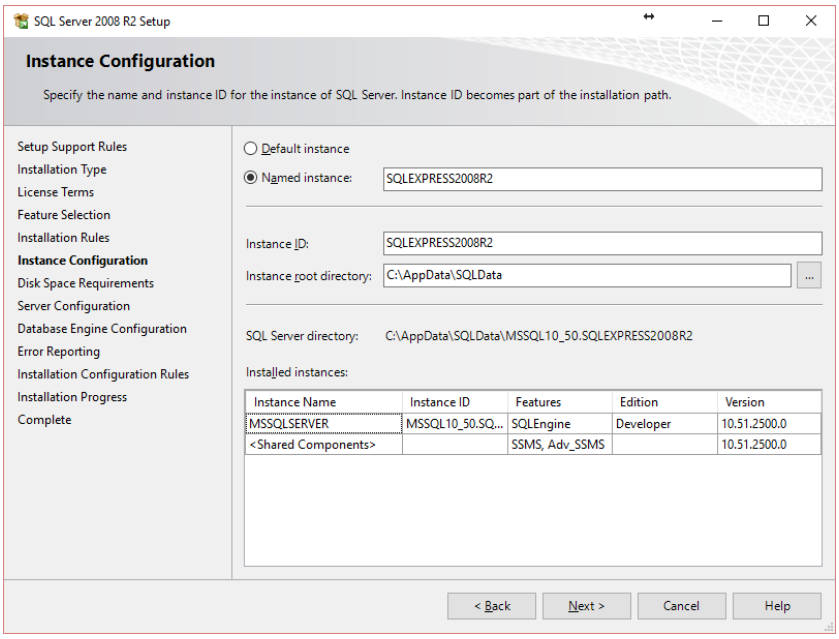
Please note, the Installation Type screen will only appear if an existing SQL instance is already installed on the server.



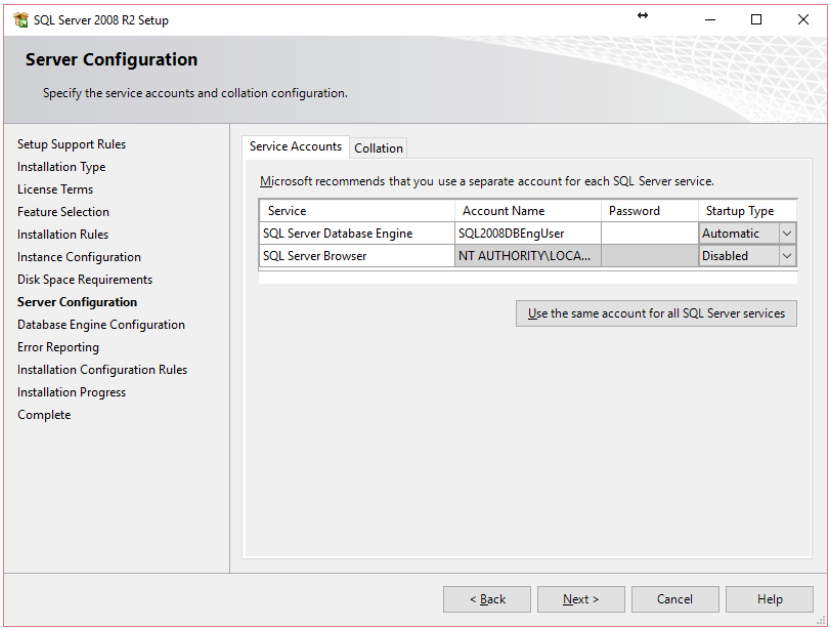
- Once reviewing the terms of the license, select the “I accept the license terms option” and click next…
- You will now be presented with the feature selection screen…



- Ensure Database Engine Services is selected.
- Ensure SQL Server Replication and SQL Client Connectivity SDK is not selected.
- Please note, Management Tools – Basic is enforced.
- Leave the shared feature directory as-is.
- Click next to progress to the instance configuration screen.

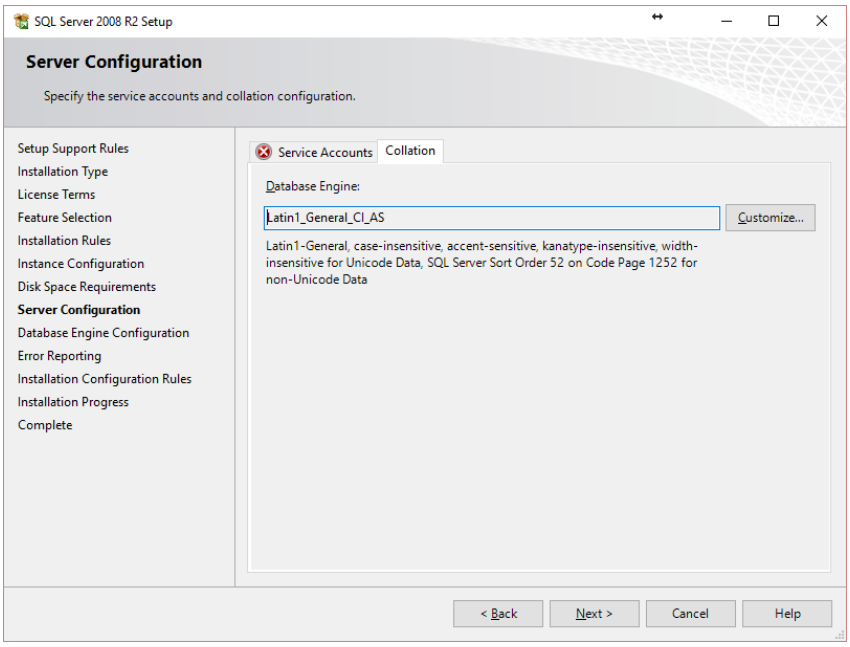


- Ensure a named instance of SQLEXPRESS2008R2 is entered.
- Change the instance root directory to X:\AppData\SQLData as per the pre-requisites section.
- Click next to progress to the Server Configuration screen…

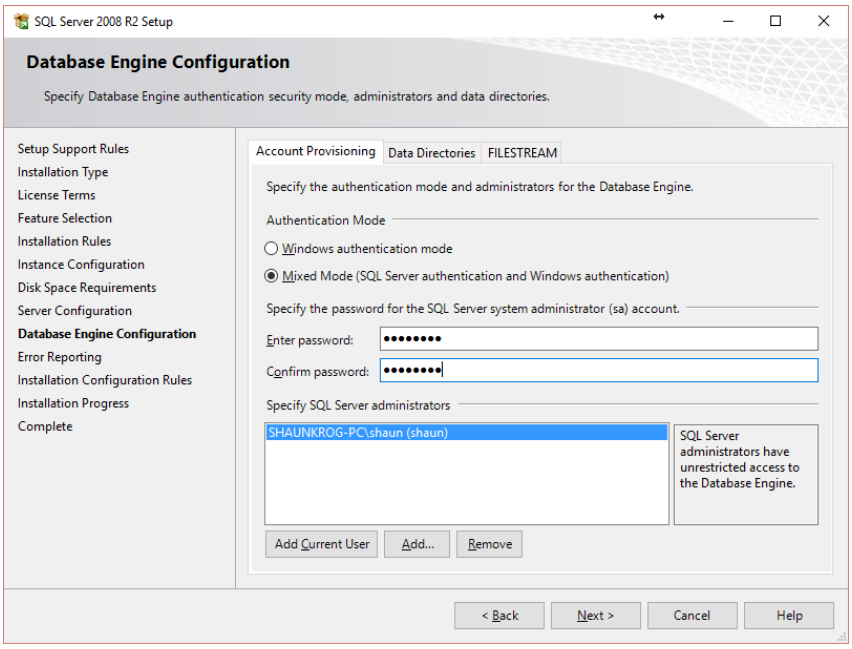


- Under service accounts, ensure the Windows user created in the pre-requisites section is selected for the SQL Server Database Engine Service.
- Ensure Startup Type for Server Database Engine Service is set to Automatic.
- Ensure Startup Type for SQL Server browser is disabled.

- Select the collation tab.

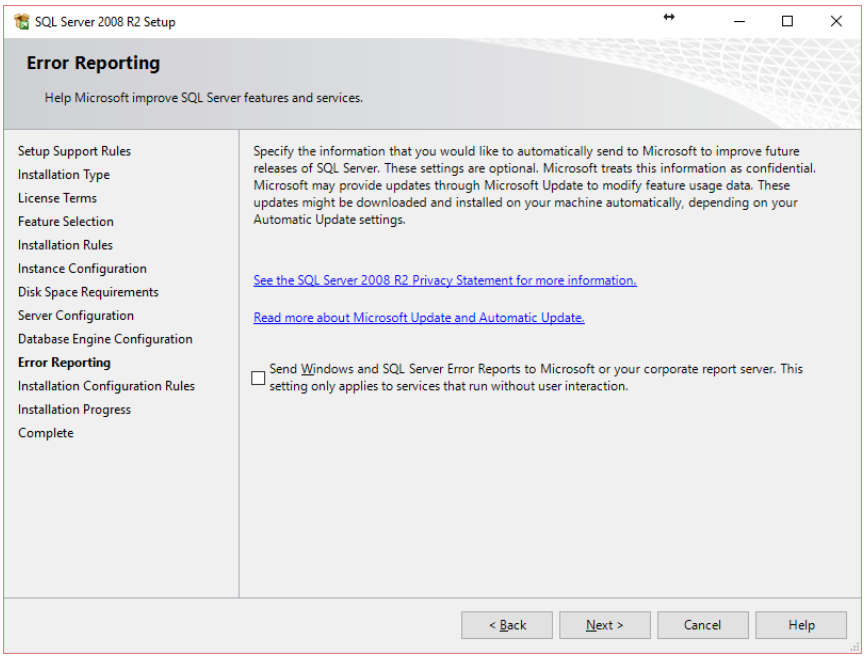


- Ensure collation is configured as Latin1_General_CI_AS or SQL_Latin1_General_CP1_Cl_AS
- Click next to progress to the Database Engine Configuration screen.

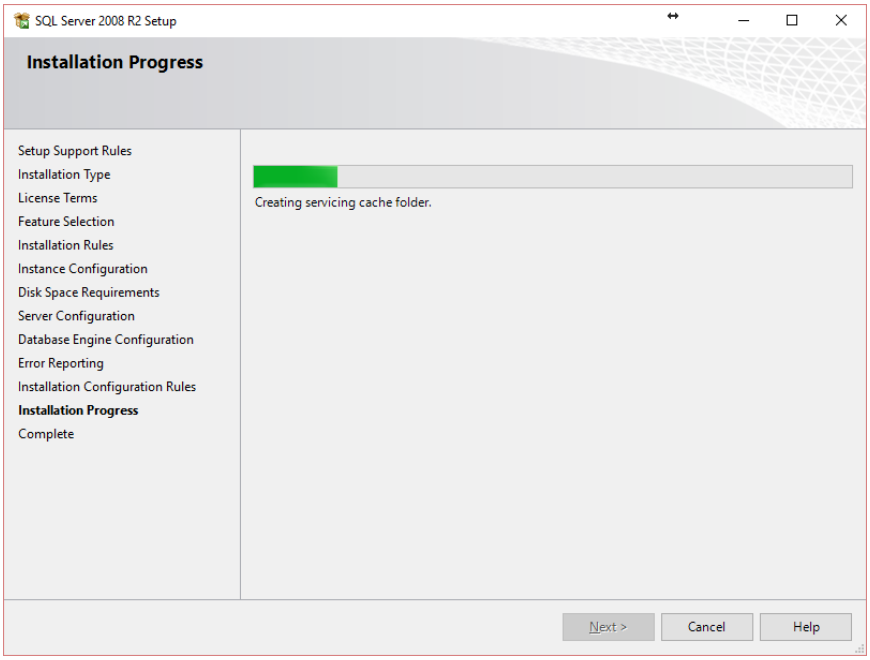


- Under account Provisioning, ensure Mixed Mode is selected as the authentication mode.
- Enter a password for the SQL server system administrator account. Please do not forget to note this password.

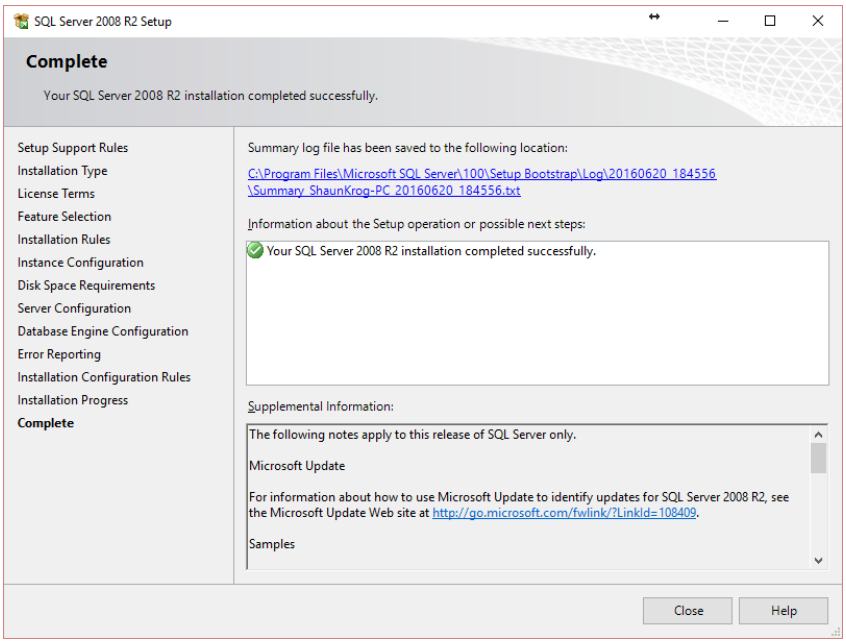
- Ensure the current user you have logged in with has been specified as a SQL Server administrator (this account is included by default).
- No changes are required under the Data Directories or FILESTREAM tabs.
- Click next to progress to the Error Reporting screen.



- Click next to progress to the Installation Progress screen.



- Ensure installation completed successfully and click close.



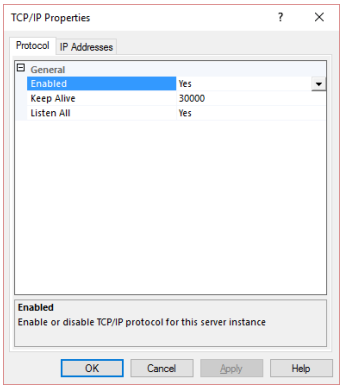
PLEASE NOTE, always ensure the latest service pack for MS SQL Server Express R2 is installed.

3.1.4 Configure SQL Server for network access

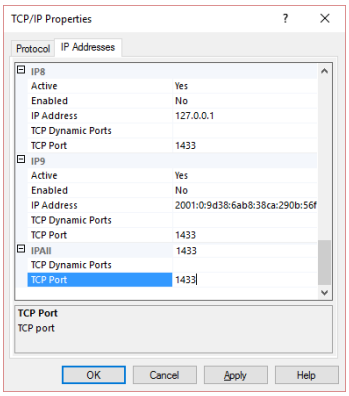
> **PLEASE NOTE,** this section is applicable for PViMS installations where the database and application servers exist on separate servers. For installations where the database and application are on the same server, please continue onto section 3.4.

- Open SQL Server Configuration Manager.

- Select protocols for SQLEXPRESS2008R2 and enable TCP/IP in the right-hand pane. To enable TCP/IP, double click on this protocol in the right-hand pane.



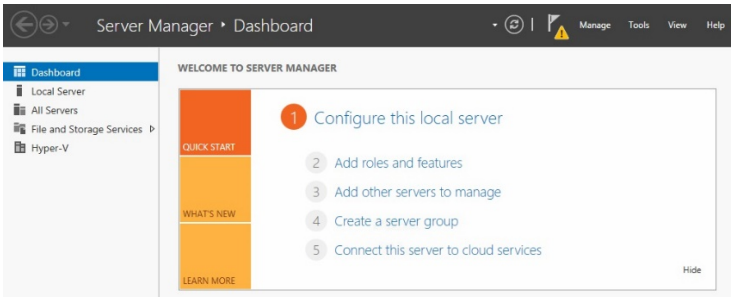
- Ensure enabled is set to yes and select the IP addresses tab.
- Ensure all TCP Dynamic Ports are set to space (remove the 0). Please note, 0 means the SQL client will connect to the underlying SQL server using dynamic TCP ports. By removing the 0, a fixed port will need to be selected.
- Ensure all TCP Ports are set to 1433 and click OK.



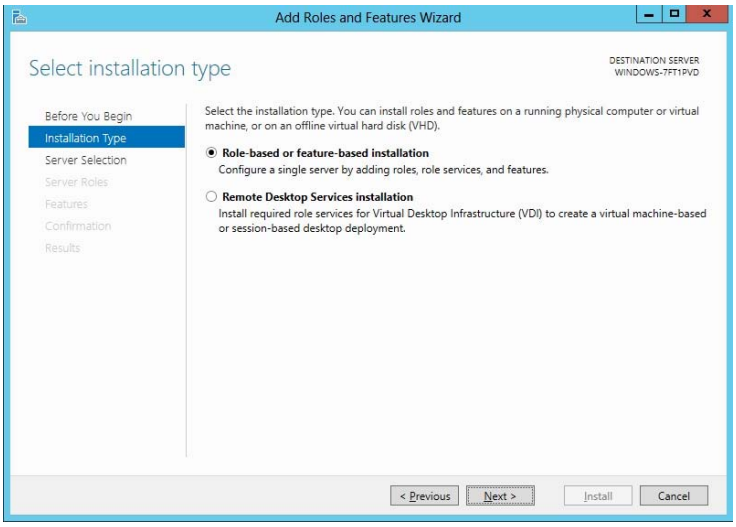
- Reboot the server for these configurations to take effect.

3.2 Installing Internet Information Services

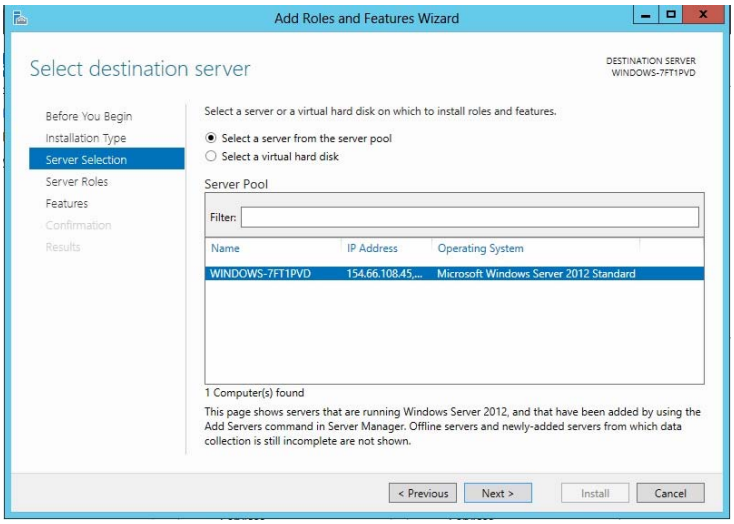
- Open Windows 2012 Server Manager.



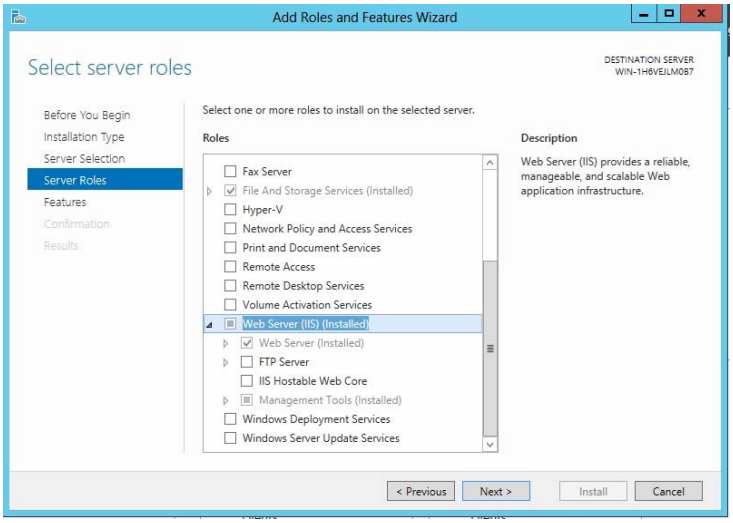
- Click Add roles and features.



- Ensure role-based or feature-based installation is selected and click next.



- Ensure the select a server from the server pool option is selected and select the appropriate server in the list of servers below.
- Click next to proceed to the list of roles to be applied to the server.



- Ensure Web Server (IIS) is selected with the following options:
  - Web Server
    - Common HTTP features (All items)
    - Health and Diagnostics (All items)
    - Performance (All items)
    - Security
      - Request filtering

        - Basic authentication
    - Application development
        - All items except CGI
- Management Tools
    - IIS Management Console
    - IIS Management Scripts and Tools
    - Management Service
- Click next to install IIS

4 Post-Implementation Database Tasks

This section describes in detail the steps required to complete the configuration and deployment of the database server.

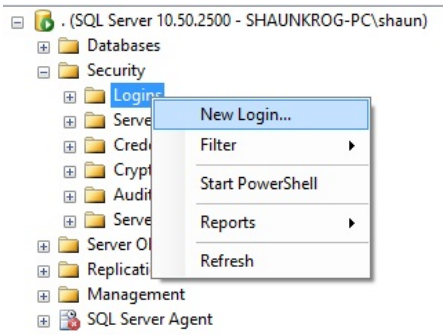
4.1 Configuring SQL Server Users – Database Owner

PLEASE NOTE, the PViMS database owner has overall rights to the underlying PViMS database, but does not have elevated permissions outside of the PVIMS database itself.

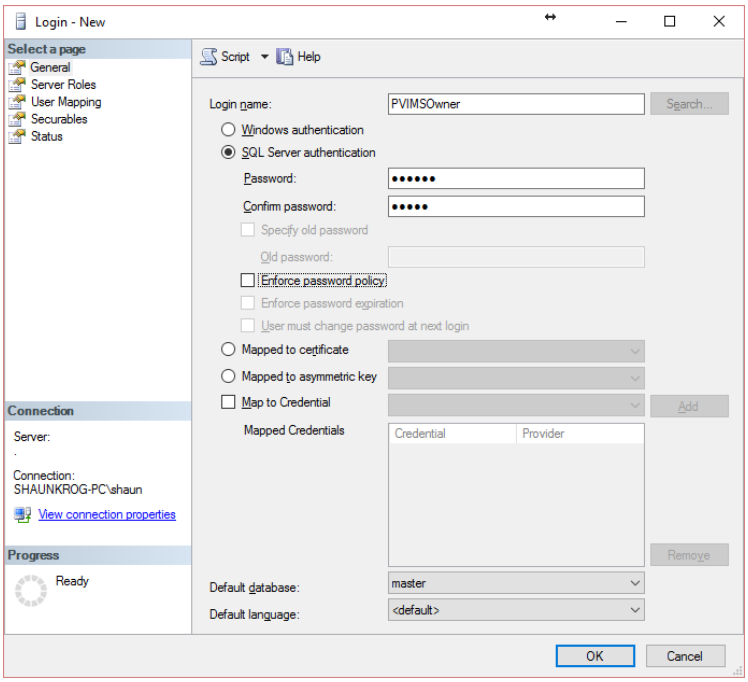
SQL Server uses role-based security, which allows you to assign permissions to a role, or group of users, instead of to individual users. Fixed server and fixed database roles have a fixed set of permissions assigned to them.

Please follow the steps below to create the PViMS database owner:

- Open SQL Management Studio.
- Expand security in object explorer.
- Right click on Logins, select New Login.



- Ensure the general page is selected.
- Select SQL Server authentication.
- Enter a login name – **PVIMSOwner**
- Enter a login password and confirm this password. Please ensure you note this password down.
- Uncheck Enforce Password Policy.
- Click OK to add this user to the server.



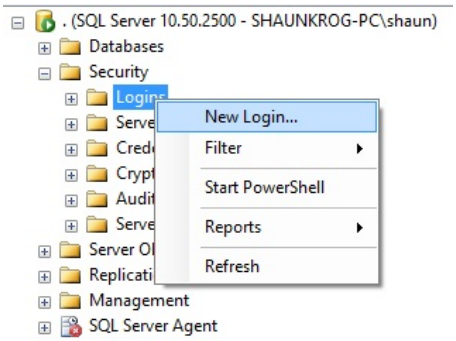
4.2 Configuring SQL Server Users – Database User

PLEASE NOTE, the PViMS database user has read and write rights to the underlying PViMS database, but does not have elevated permissions outside of the PVIMS database itself.

SQL Server uses role-based security, which allows you to assign permissions to a role, or group of users, instead of to individual users. Fixed server and fixed database roles have a fixed set of permissions assigned to them.

Please follow the steps below to create the PViMS database user:

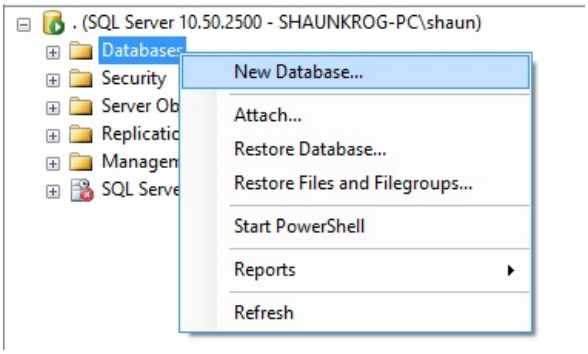
- Open SQL Management Studio.
- Expand security in object explorer.
- Right click on Logins, select New Login.



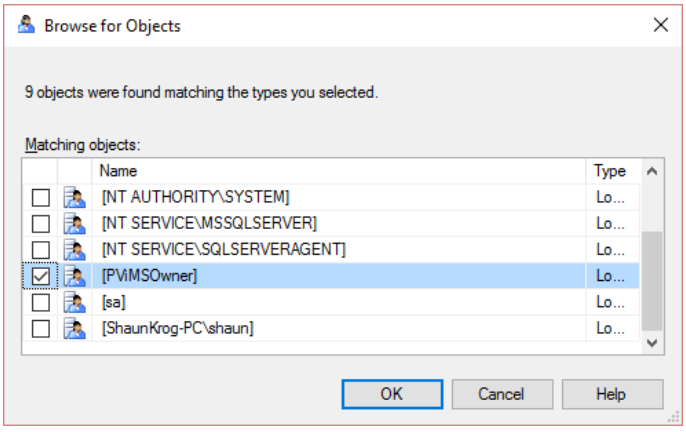
- Ensure the general page is selected.
- Select SQL Server authentication.
- Enter a login name – **PVIMSUser**
- Enter a login password and confirm this password. Please ensure you note this password down.
- Uncheck Enforce Password Policy.
- Click OK to add this user to the server.

4.3 Create Database

- Open SQL Management Studio
- Right click the Databases node under the appropriate SQL Server instance and select New Database.



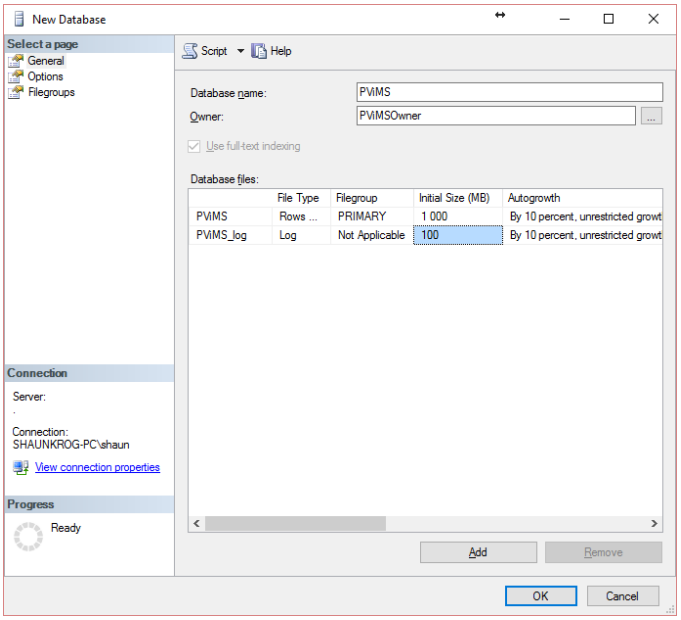
- Enter the database name – PVIMS
- Select the database owner:
    a. Click the ellipses button next to the database owner.
    b. Click browse.
    c. Select PVIMSOwner and click OK.



- Configure database files as follows:
    a. **Logical Name:** PViMS

i. Select an **initial database** size. It is recommended that the database be configured with an initial size of 1000MB.
ii. Configure **database size auto growth.** It is recommended that the database be configured to grow at 10%.
iii. Location of file. The database (MDF) file should be on a drive exposed on the RAID 5 configuration. It is recommended that all MDF files be stored in the same location on this RAID 5 drive. The reason for utilising a RAID 5 drive is for redundancy and data security.

b. **Logical Name:** PViMS_Log
i. Select an **initial log** size. It is recommended that the log be configured with an initial size of 100MB (10% of the database size).
ii. Configure **log size auto growth.** It is recommended that the log be configured to grow at 10%.
iii. Location of file. The log (LDF) file should be on a drive exposed on the RAID 1 configuration. It is recommended that all LDF files be stored in the same location on this RAID 1 drive. The reason for utilising a RAID 1 drive is for performance and efficiency.

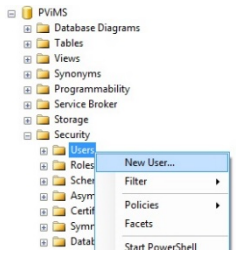


- Click OK to create the new database

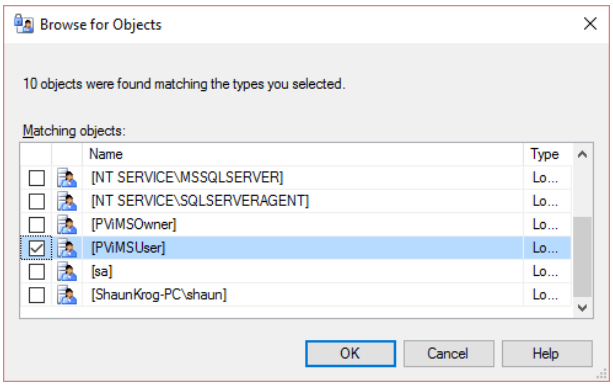
PLEASE NOTE, at this point, a blank version of the PVIMS database has been created. This database will not at this point contain any PVIMS related assets or database entities.

4.4 Assign Database User

- Open SQL Management Studio.
- Expand the Databases → PViMS node.
- Expand the database security node, right click on users and select new user.



- Ensure the general page is selected.
- Select the database user:
    a. Click the ellipses button next to the login name.
    b. Click browse.
    c. Select PVIMSUser and click OK.



- Enter a User name – **PVIMSUser**
- Ensure db_datareader and db_datawriter are selected under Database role membership.
- Click OK to add this user to the server.

5 Post-Implementation Application Tasks

This section describes in detail the steps required to complete the configuration and deployment of the application server.

5.1 Configuring IIS Application Pool

- An application pool is a configuration that links one or more applications to a set of one or more worker processes. Because applications in an application pool are separated from other applications by worker process boundaries, an application in one application pool is not affected by problems caused by applications in other application pools.
- By creating new application pools and assigning Web sites and applications to them, you can make your server more efficient and reliable, as well as making your other applications always available, even when the worker process serving the new application pool has problems.

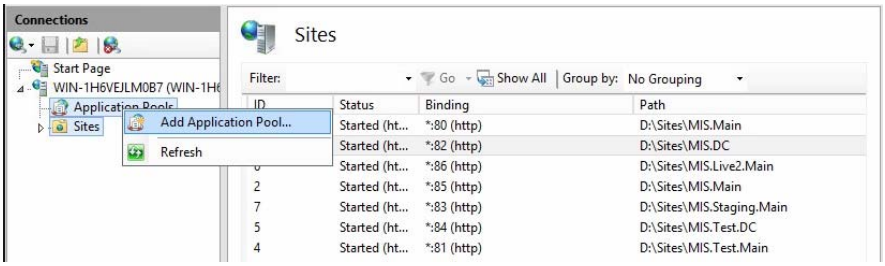
5.1.1 Guidelines for Creating Application Pools

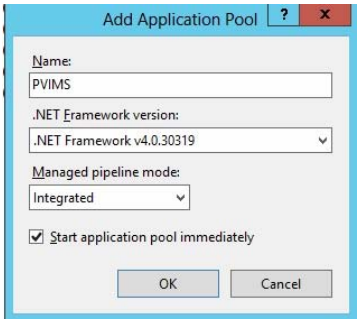
Consider the following guidelines when configuring application pools:

- To isolate Web applications on a Web site from Web applications on other sites running on the same computer, create an individual application pool for each Web site.
- For enhanced security, configure a unique user account (process identity) for each application pool. Use an account with the least user rights possible, such as Network Service in the IIS_WPG group.
- If there is a test version of an application on the same server with the production version of the application, separate the two versions into different application pools. This isolates the test version of the application.
- As a design consideration, if you want to configure an application to run with its own unique set of properties, create a unique application pool for that application.
- Important: You must be a member of the Administrators group on the local computer to perform the following procedure or procedures. As a security best practice, log on to your computer by using an account that is not in the Administrators group, and then use the runas command to run IIS Manager as an administrator. At a command prompt, type runas ***/user: Administrative_AccountName"mmc%systemroot%\system32\inetsrv\iis.msc".***

To create a new application pool:

- In IIS Manager, right-click Application Pools and select Add Application Pool.



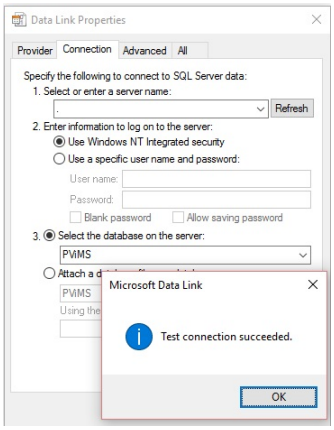


- Please ensure that the name of the application pool is PVIMS.
- Set the .NET framework version to .Net Framework v4.0.30319 or .Net CLR v4.0.30319.
- Set Managed pipeline mode to Integrated.
- Ensure the Start application pool immediately checkbox is selected.
- Click OK to add the new application pool.

5.2 Test SQL Connectivity from an external workstation

PLEASE NOTE, this section is applicable for PViMS installations where the database and application servers exist on separate servers. For installations where the database and application are on the same server, please see section 4.7.

- The most efficient way to test SQL connectivity is to create a universal datalink file (.UDL).
- Create a new file on the desktop with the name **test.udl**
- Double click on this file on the desktop, you will be presented with a data link properties screen.

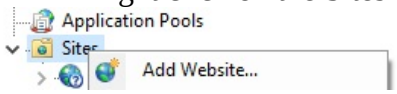


- Enter the relevant connection parameters and click on the 'Test Connection' button.
- A message should appear prompting that the connection was successful.

PLEASE NOTE, a successful connection ensures that it will be possible to connect the application server to the database server over TCP/IP.

5.3 Create IIS Application

- Please ensure the following folder is created on the workstation/server where the application is being installed: - **x**:\AppData\Sites\PViMS where **x: is the specified drive for data.**
- In IIS Manager, expand the sites node.
- Right click on the Sites node and click Add Website.



- Enter **PViMS** as the site name.
- Select the **PViMS** application pool.
- Select the location of the PViMS folder (**x**:\AppData\Sites\PViMS) as the Physical Path.
- Enter 100 as the port number for the site .
- Please note, for public facing sites, if you have a registered domain for the access of PViMS, enter this domain in the host name field.
- Click OK to create the application.
- The application will now be created and added to the sites node in IIS Manager.
- Close IIS Manager.
- Unzip the contents of the PViMS setup package into a temporary folder on the IIS server.
- Please copy the contents of the WWW folder in the PViMS setup package into the folder **x**:\AppData\Sites\PViMS

5.4 Configure Database Connection String – Database Owner Access

- Open Notepad.

- Browse to **x**:\AppData\Sites\PViMS
- Open the file web.config
- Locate the <connectionStrings> node.
- Replace **SERVERNAME** with the name of the SQL server .
- If SQL server has been configured for network access as per section 3.2.4 then replace **PORT** with "**,1433**"(do not include the double quotes) .
- If SQL server has **NOT** been configured for network access as per section 3.2.4 then remove **PORT** as a port is not required.
- Replace **USERNAME** with PViMSOwner
- Replace **PASSWORD** with the password allocated to the PViMSOwner SQL account.
- Save changed to the web.config file and close notepad.
- Browse to the PViMS portal using the URL http://**servername**:100/

5.5 Create PViMS Database Objects

- Browse to the PViMS login page on http://***servername:100***

- You will be presented with the PViMS login screen.
- Login to PViMS using the standard admin user name and password.

A default administrator account has been generated. You will be able to log in with the following credentials: -

User Name: Admin
Password: P@55w0rd1

- You will now be routed to an installation screen.
- Confirm that PVIMS has been installed successfully.

5.6 Configure Database Connection String – Database User Access

- Open Notepad.
- Browse to **x**:\AppData\Sites\PViMS
- Open the file web.config
- Locate the <connectionStrings> node.
- Replace **USERNAME** with PViMSUser.
- Replace **PASSWORD** with the password allocated to the PViMSUser SQL account.
- Save changed to the web.config file and close Notepad.
- Browse to the PViMS portal using the URL http://**servername**:100/

By deprecating access to the user account, unauthorized access via the PViMS IIS application will result in the user having restricted permissions to the underlying SQL server.