Vocera Alarm Management Configuration Guide

Version 2.2.5

Notice

Vocera Communications, Inc.
www.vocera.com
tel :: +1 408 882 5100
fax :: +1 408 882 5101

Last modified: 2018-11-27 07:55

VAM-225-Docs build 26

Contents

Vital Information

Throughout this guide, critical information will be set off from the rest of the text as described here.

Important: Text set off in this manner presents important information. Information that is vital to the proper use of the system is labeled as Important.

Warning: A warning indicates information that is related to patient or operator safety, or possible damage to the equipment.

Warnings / Contraindications / Precautions

Warning: If you are using this product in the U.S.A., federal law restricts this device to sale by or on the order of a physician. This system is to be used by trained medical personnel only.

Warning: Electrostatic Discharges and certain Electromagnetic fields might adversely affect the functioning of the handheld device. Be aware of your surroundings when operating handheld devices. If it is determined that the use of a particular device in its normal data mode can cause or be susceptible to harmful electromagnetic interference or electrostatic discharge, change the device on which Vocera Alarm Management is deployed. Refer to ANSI C63.18-1997.

Warning: Vocera Alarm Management is a SECONDARY notification system. This does not reduce the need for primary monitoring on either the patient monitors or at the central stations.

Warning: Vocera Alarm Management is only as reliable as your wireless network. Consult your hospital's Information Technology personnel for planning against network issues that might impede proper functioning of the Vocera Alarm Management system. If sufficient network coverage is not available, STOP use of the Vocera Alarm Management system.

Warning: Do NOT update or change the firmware or system software on the handheld without prior permission from your hospital Information Systems personnel. Change in system firmware may make the Vocera Alarm Management system inoperable.

Warning: Do NOT update or change the firmware or system software on the handheld without prior permission from your hospital Information Systems personnel. Change in system firmware may make the Vocera Alarm Management system inoperable.

Warning: If the system is used through the cellular data network, the data connection might be intermittent or the latency very high. This is outside the control of Vocera. Use other means of communication if network latency is high.

Warning: Operating this equipment in environments with high electromagnetic radiation might cause noise on the display.

Warning: Operating this equipment in environments with high electromagnetic radiation might cause noise on the display.

Warning: Operating this equipment in environments with high electromagnetic radiation might cause the system to malfunction, including the inability to receive alarms.

Warning: Vocera does not recommend the use of Vocera Alarm Management through the cell phone network when in clinical care areas. Follow your hospital's guidelines and policies regarding the use of cellular devices in clinical areas.

Warning: Changing some of the system settings described in this manual may require clinical and administrative acceptance.

Warning: All system firmware, including server operating systems, system patches, handheld operating firmware, and network infrastructure firmware, have been validated at the time of installation. Contact Vocera prior to changing any of these and confirm if the proposed change impacts the installation qualification of Vocera Alarm Management.

Warning: The Vocera Alarm Management Console is designed to operate through specific versions of popular web browsers. Check with your system administrator prior to applying any changes to your web browser.

Warning: If the Vocera Alarm Management native push process cannot be deployed on the handheld system, alarm delivery and tracking may not be accurate.

Fully qualified personnel should install, maintain, troubleshoot, calibrate, and repair the system.

DO NOT use a malfunctioning Vocera Alarm Management system.

Cautions and Disclaimers

- The equipment is intended for use by qualified medical personnel and should be used only after personnel have been trained in the proper use of this equipment.

  For continued safety, it is necessary that the instructions listed in this manual are followed. It is important that instructions in this manual in no way supersede established medical procedures concerning patient care. Bring any such conflicts to the attention of your management.Bring any such conflicts to the attention of your management.
- Ensure that sufficient network coverage exists where Vocera Alarm Management is intended to be used. If deemed essential, deploy a secondary failover network.
- Ensure that sufficient network coverage exists where Vocera Alarm Management is intended to be used. If deemed essential, deploy a secondary failover network.
- The screen of the device you are using may be set to blank or dim for saving power. It is recommended that you consult with your system administrator and set this up appropriately.
- Report a missing or stolen handheld with Vocera software on it immediately to your system administrator for immediate deactivation of the Vocera Alarm Management client on that device.
- It is recommended that a minimum of 3 users be logged in if at least one patient is being monitored and Vocera Alarm Management is being used for delivery of secondary alarm notification to assure a high reliability rate.
- It is recommended that a minimum of 3 users be logged in if at least one patient is being monitored and Vocera Alarm Management is being used for delivery of secondary alarm notification.
- The system depends on fully charged and functional handheld units being available for use. The customer should accordingly ensure that there are sufficient handheld devices available for use.

- Use only FCC certified handhelds. FCC certifications should be available for inspection through your device vendor.
- Vocera recommends that users take into consideration the EMI and EMC testing performed on the handheld devices and network clients prior to deploying the handheld client on the devices.
- If the data network is congested, the data connection might be intermittent or the latency very high. This is outside the control of Vocera. Use other primary means to monitor alarms if network issues or related symptoms are observed.
- Server and network appliances used should have a backup uninterrupted power supply.
- The message formats and processes are designed to fit the hospital's specific information sources. Any change in firmware or software to these systems might adversely affect the Vocera Alarm Management system's ability to process the information coming from these systems. Consult with Vocera prior to changing or upgrading the configurations of such systems.
- A short network path improves system delivery times. Hence, do not change the network configurations that might adversely affect the Vocera Alarm Management network setup.
- Changing the network configuration may cause Vocera Alarm Management to stop functioning.
- Excessive data retained on the system database might slow down the system or even stop the system from working. The customer is advised to put in a routine data backup and archival plan prior to starting clinical use of the system.
- Do not use the Vocera Alarm Management handheld client while driving.
- The system requires the secondary network described in the installation checklist to assure reliability. If the system is deployed on a single network, confirm the availability of sufficient backup systems and processes to ensure safe and effective operation of the system.
- Prior to deploying other wireless devices and medical devices, confirm that they do not interfere with or use up channel bandwidth currently allocated to the Vocera Alarm Management system.
- Current Electromagnetic testing requirements for medical devices may NOT include the spectrum of frequencies covered by current VOIP communication devices. Vocera recommends Ad Hoc testing be performed, according to "Ad Hoc Test Method for Estimating Radiated Electromagnetic Immunity of Medical Devices to Specific Radio-Frequency Transmitters" (ANSI C63.18-1997), for all handheld devices on which the Vocera Alarm Management system is expected to operate. This testing should include the devices typically found on the proposed floor on which Vocera Alarm Management is deployed.
- A minimum distance of 10 inches should be maintained between any medical device and wireless devices per ANSI C63.18-1997: American National Standard Recommended Practice for an On-Site Ad Hoc Test Method for Estimated Radiated Electromagnetic Immunity of Medical Devices to Specific Radio-Frequency Transmitters, unless your testing and analysis shows the need for a larger minimum separation.
- Strong electrical and/or magnetic fields, such as those on MRI scanners and microwaves, may impair the operation of the handheld device on which the Vocera Alarm Management client operates.
- Certain areas of the hospital, such as elevators, may not offer the necessary connectivity to the wireless network of choice.
- It is recommended that Vocera Alarm Management be deployed in a dual-server configuration with "hot failover" and the two servers be provided separate power sources to ensure continued operations in case of power failures.
- The server is not intended to be routinely restarted and any system changes may not be performed without prior approval/notification to Vocera.
- Batteries on handheld devices typically have a limited life. Consult the device user's manual and ensure that batteries have the ability to carry sufficient charge.

- Confirm that you are familiar with identifying the different alarm levels and their respective color and ring tone representations prior to using the system.
- Vocera Alarm Management is individually validated for use at each facility. The overall system delay is explicitly measured. Consult your system administrator and understand the possible system delays prior to using Vocera Alarm Management. Typical Maximum Alarm Generation Delay is in the order of 3 seconds.
- The Vocera Alarm Management system is individually validated for use at each facility. The overall system delay is explicitly measured. Consult your system administrator and understand the possible system delays prior to using Vocera Alarm Management. Typical Maximum Alarm Generation Delay is in the order of 3 seconds.
- Vocera Alarm Management only transmits alarms generated by your patient monitors/patient monitoring network. All alarm limits are set on your central monitor/patient monitors. Setting your patient monitors/central station alarm limits such that alarms are not generated will result in no alarms coming through the Vocera Alarm Management.
- Vocera recommends the use of handheld devices with a minimum resolution of 320 pixels x 240 pixels.
- Vocera recommends a minimum screen resolution of 1024 x 768 pixels with at least 16 bit color with Internet Explorer 8 or Chrome 10 or higher version for the console, on a PC running Windows XP Professional or higher, with a minimum 2 GB RAM and a 2GHz or higher processor with necessary network connectivity. Apple's Safari browser 5.0 or higher on an Apple Macintosh computer is also supported.
- Vocera recommends a minimum screen resolution of 1024 x 768 pixels with at least 16 bit color with Internet Explorer 8 or Chrome 10 or higher version for the console, on a PC running Windows XP Professional or higher, with a minimum 2 GB RAM and a 2GHz or higher processor with necessary network connectivity.
- Vocera recommends the use of handheld devices with a minimum resolution of 320 pixels x 240 pixels.

Trademarks

The names and images of various devices used are trademarks of the various companies that manufacture or market the respective devices.

Patents

Vocera Alarm Management and its components are covered by one or more patents pending.

Intended Use Statement

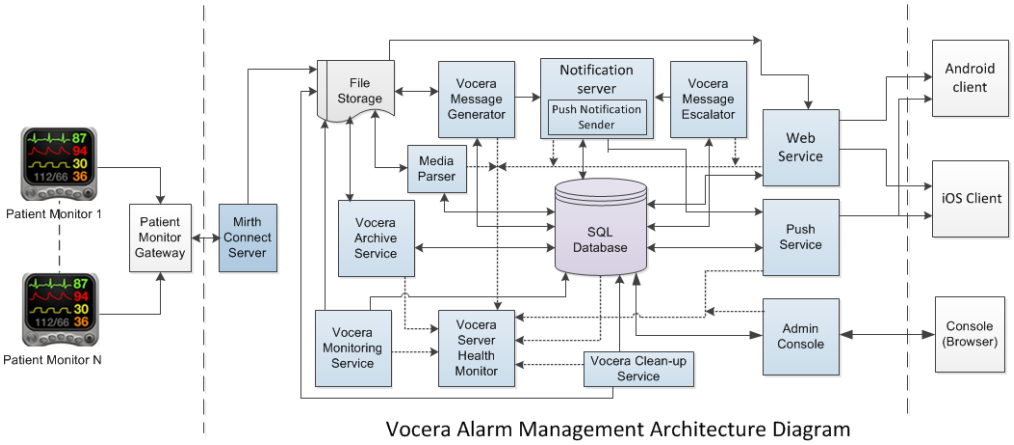
Vocera Alarm Management is software intended for use to display status and alarm events from other medical devices and patient information systems and associated physiological and other patient information. It serves as a parallel, redundant mechanism to inform the clinical staff of patient events. It is intended to be a secondary means of announcing and displaying patient alarm and physiological information to mobile healthcare providers.

Vocera Alarm Management is limited to use by qualified medical professionals who have been trained on the use of the system. It is intended to supplement and not to replace any part of the current patient monitoring systems. It is not considered in and of itself to be diagnostic without skilled interpretation and does not replace physician's care.

Vocera Alarm Management Overview

The Vocera Alarm Management integration capability enables the VAM server to capture and send the alarms generated by patient monitors.

As shown in the architecture diagram below, a Mirth Connect server captures the HL7 messages generated by the patient monitors.



Vocera Alarm Management Architecture Diagram

The Mirth server transforms the messages to a file in a JSON format that is understood by the VAM system. VAM then parses the alarm, feeds it into a database, and sends a notification to the user assigned to the respective beds or rooms. The VAM server then decides whether or not to escalate the alarm, based on the online status and the response.

Installing Vocera Alarm Management

These sections describe how to install Vocera Alarm Management, along with the necessary prerequisites that must be in place first.

Vocera Alarm Management Requirements

These are the installation requirements and prerequisites for Vocera Alarm Management.

Note: Vocera recommends that VAM and SQL are to be installed on separate servers.

VAM Application Server Requirements

These are the installation requirements and prerequisites for the VAM server.

Hardware:

- Intel Xeon Quad Core
- 16 GB RAM
- Dedicated 500 GB hard drive for Vocera

Operating System:

- Microsoft Windows Server 2012 R2 Standard Version (recommended) 64-bit
- Microsoft Windows Server 2008 R2 Standard Version 64-bit
- Microsoft Windows Server 2008 R2 Enterprise Version 64-bit

VAM version:

- VAM 2.2.4 (required)

Additional Microsoft software and utilities:

- IIS 7 or above
- Microsoft .NET Framework 4.5 (for Windows Server 2008 R2 only - installs natively with Windows Server 2012 R2 Standard)
- The MS SQL ODBC Driver and Command Line Utilities (required if the SQL database is on a separate server, as is recommended)

Java version:

- Java JRE version 1.7.0_45 or 1.7.0_75 (64-bit version, required)

Supported browser versions:

- Chrome 48.0
- Internet Explorer 11
- Mozilla Firefox 44.0.1

VAM Database Server Requirements

These are the installation requirements and prerequisites for the VAM database server.

Hardware:

- Intel Xeon Octa Core 2GHz
- 16 GB Memory
- Disk Provisioning

  If you partition your database server disk into multiple drives, Vocera recommends the following:

  Table 1: VAM Database Server Disk Provisioning

Drive	Size
SQL Installation and OS	120 GB
SQL Log Files	240 GB
TEMPDB-Logs	240 GB
TEMPDB	240 GB
SQL_DATA	240 GB

  Disk space provisioning depends on the number of alarms you receive and how much data you intend to archive. If you want to save all archived files, it is better to have 500 GB on the drive where the archived files are stored.

  If you do not logically partition your hard disk, Vocera recommends a minimum of 500 GB on a single drive for the SQL Server installation, the JSON files created by alarms, application log files, and the archiver service's saved CSV files (and additional space if you intend to save all archived files).

Operating System:

- Microsoft Windows Server 2012 R2 Standard Version (recommended) 64-bit
- Microsoft Windows Server 2008 R2 Standard Version 64-bit
- Microsoft Windows Server 2008 R2 Enterprise Version 64-bit

Microsoft SQL Server:

- SQL Server 2012 R2 Standard Version (recommended) 64-bit
- SQL Server 2012 R2 Enterprise Version 64-bit
- SQL Server 2008 R2 Standard Version 64-bit
- SQL Server 2008 R2 Enterprise Version 64-bit

Note: The account that is running Microsoft SQL Server must have system administration privileges.

SQL clustering is supported in VAM.

Installing the Prerequisites

Before you can install Vocera Alarm Management, your environment must contain the necessary prerequisites.

- The hardware and software requirements as described in the *Vocera Alarm Management Server Sizing Matrix* must be provided. This includes a supported MS SQL Server on a system dedicated to that server.

- Specify SQL_Latin1_General_CP1_CI_AS as the default collation (sort order) for the SQL Server instance. This collation supports the following capabilities that are required by the Vocera Alarm Management:

  Table 2: SQL Server Collation

Collation Capability	Description
Latin1	Specifies the Latin 1 character set (ASCII)
CP1	Specifies code page 1 (ANSI code page 1252)
CI	Specifies case-insensitive sorting, so "ABC" is treated the same as "abc"
AS	Specifies accent-sensitive sorting, so "ü" is not treated the same as "u"

  To specify a collation for a server instance, open SQL Server Management Studio, right-click on the SQL Server instance, and select Properties. The Server Collation field displays the collation currently in use.
- Ensure that your SQL Server is set to enable SQL Server authentication mode. Windows authentication mode is not supported in Vocera Alarm Management.
- The server on which you are to install VAM must be able to access the remote MS SQL Server. Follow these instructions to test your MS SQL connection: **Test Remote SQL Connectivity Easily**.

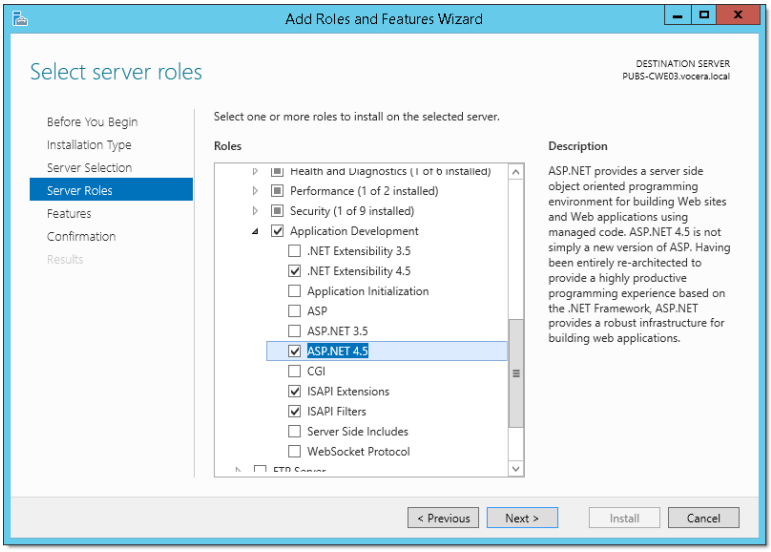
  Note: You can use VAM in a clustered SQL environment.
- Microsoft .NET Framework 4.0 or higher must be installed. To check that an appropriate .NET Framework is installed in your environment, use the following: **How to: Determine Which .NET Framework Versions Are Installed**. To install .NET Framework, see **Installing the .NET Framework**.

  .NET Framework should be installed before IIS is enabled on your server. If you have enabled IIS before installing .NET Framework, run the following command to register .NET Framework:

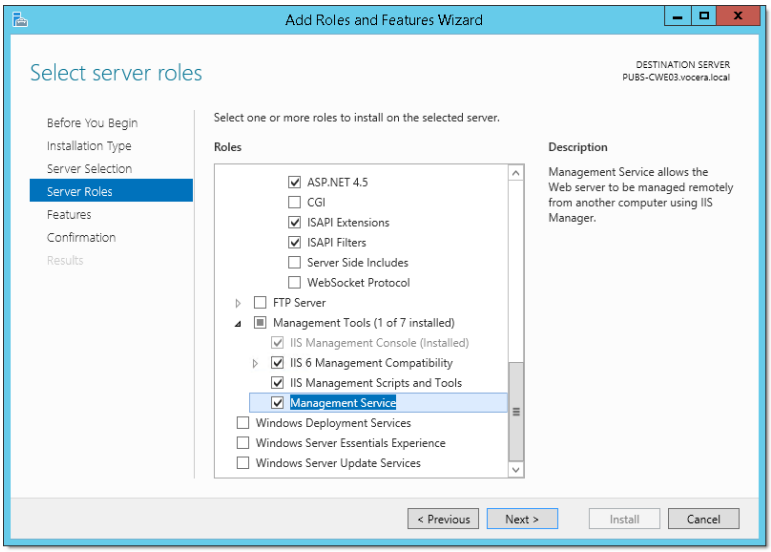
  ```
  C:\Windows\Microsoft.NET\Framework\v4.0.30319\aspnet_regiis.exe /i
  ```

  In this command, replace `v4.0.30319` with the version of .NET Framework that you have installed.
- IIS must be installed on your server. Follow these steps to install IIS: **Installing IIS 8 on Windows Server 2012** or **Installing IIS 7 on Windows Server 2008 or Windows Server 2008 R2**. IIS 7.5 and IIS 8.5 are supported.
- After you have installed IIS, you must enable the necessary ASP.NET and IIS management components. To do this:
  - On the Windows Server 2008 R2 machine:
    - Click Start > All Programs > Administrative Tools > Server Manager. You may need to wait a few moments while the list of installed roles is generated.
    - Click Add Roles and Server Roles.
    - Select Web Server (IIS), and click Next until the Select Role Services pane appears.
  - On the Windows Server 2012 R2 machine:
    - Click on the Server Manager icon in the status bar at the bottom to start the Server Manager.
    - In the Server Manager, set Add roles and features to display the Add Roles and Features Wizard.
    - Click Next.
    - Select Role-based or feature-based installation. Click Next.
    - Select your server and click Next to display the Select Server Roles pane.

- Expand Web Server (IIS), Web Server, and Application Development. Select ASP.NET 4.5 (Windows Server 2012) or ASP.NET (Windows Server 2008). If you are asked to select other features to include with ASP.NET, include them.



- Expand Web Server (IIS) and Management Tools. Select IIS 6 Management Compatibility, IIS Management Scripts and Tools, and Management Service.



  Click Next.
- Follow the remaining steps, clicking Next and Install as needed. This will install the necessary components on your system.
- Open a Command Prompt window as administrator, and run **`iisreset`** to restart your IIS server.

- Java JRE version 1.7.0_45 or above (64-bit version) must be installed. To install Java JRE version 1.7.0_45:
  - Go to the **Java SE 7 Archive Downloads** page.
  - Scroll down to Java SE Runtime Environment 7u45.
  - In the Java SE Runtime Environment 7u45 section, click Accept License Agreement.
  - Click jre-7u45-windows-x64.exe to download the Java installer. You may need to create an Oracle account to do this.
  - Double-click on this installer to install it. Use all default values.

- The MS SQL ODBC Driver and Command Line Utilities must be installed. For details, see these pages:
  - MS SQL ODBC Driver: **Microsoft ODBC Driver 11 for SQL Server - Windows**.
  - MS SQL Command Line Utilities: **Microsoft Command Line Utilities 11 for SQL Server**.

  You must install the ODBC Driver before installing the Command Line Utilities. Ensure that you select the 64-bit versions of both.
- If you have a firewall installed on the system on which you will be installing VAM, the firewall must allow incoming data from the port that the MS SQL Server uses (such as 1433). For details, see **Configure the Windows Firewall to Allow SQL Server Access**.
- Open a Command Prompt window as administrator, and go to the `C:\Windows\Microsoft.NET\Framework\v4.0.30319` folder (replace `v4.0.30319` with your version of Microsoft .NET Framework). Run the following command:

  ```
  aspnet_regsql -S serverName -U sa -P password -ssadd -sstype p
  ```

  Replace `serverName` with the name of your SQL server, and replace `password` with the password you used to set up the system administrator account `sa`.
- Restart your computer before installing VAM.

Verifying Prerequisites

These sections describe how to verify that the VAM prerequisites are in place.

Verifying Java

To verify that you have the correct version of Java, type `Java -version`.

This displays a result similar to the following, depending on the version of Java that you have installed:

```
C:\>java -version
java version "1.7.0_45"
Java(TM) SE Runtime Environment (build 1.7.0_45-b18)
Java HotSpot(TM) 64-Bit Server VM (build 24.45-b08, mixed mode)
```

Verifying the SQL Username and Password

To verify the SQL username and password, log into the SQL server from the SQL Server Management Studio.

If needed, create a new user with the relevant roles and verify whether this user is able to log in. This new user may need to change the password on the first login, which may create problems.

Ensure that the password is not expired or disabled.

Verifying the SQL Driver

To verify that you have the correct SQL driver, type `sqlcmd -?` in the Command Prompt window.

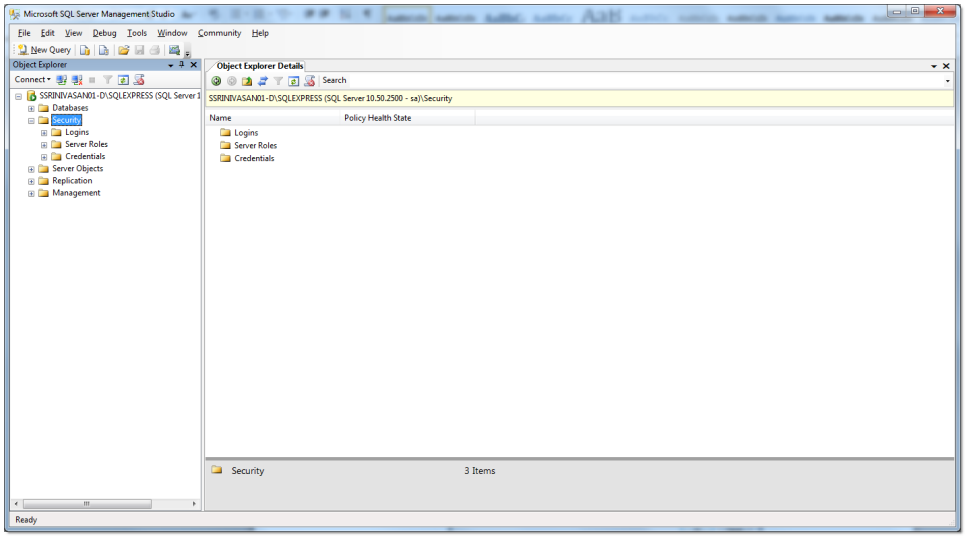
This displays the following result:

```
C:\>sqlcmd -?
Microsoft (R) SQL Server Command Line Tool
Version 11.0.2100.60 NT x64
Copyright (c) 2012 Microsoft. All rights reserved.
```

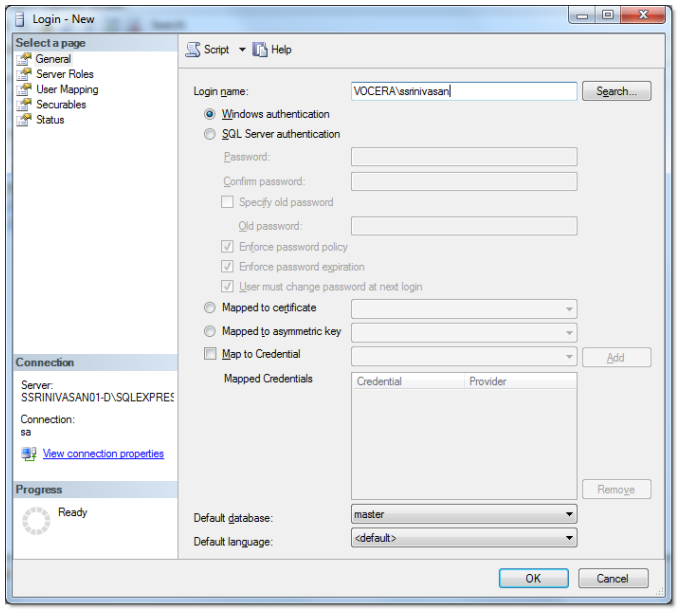
Verifying Database Privileges

Follow these steps to verify the database privileges in the SQL Server.

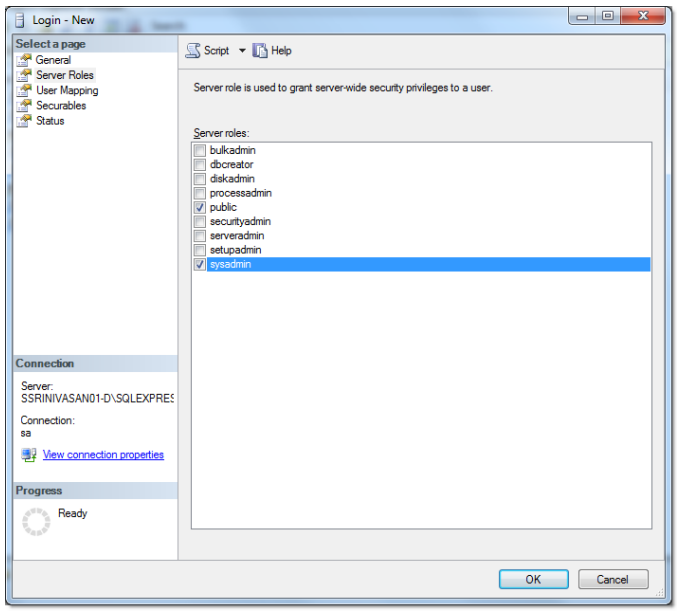
1. Log into the SQL Server with valid credentials.
2. In Object Explorer, expand Security and Logins.



3. If you have previously set up a user with the necessary database privileges:
   a. Right-click on the user's name and select Properties.
   b. Select Server Roles.
   c. Ensure that public and sysadmin have been selected.
4. If you need to set up a user with database privileges:
   a. Right-click Logins > New Login.
   b. Type the login name of the user that is to be granted database privileges. Select the Windows authentication radio button.



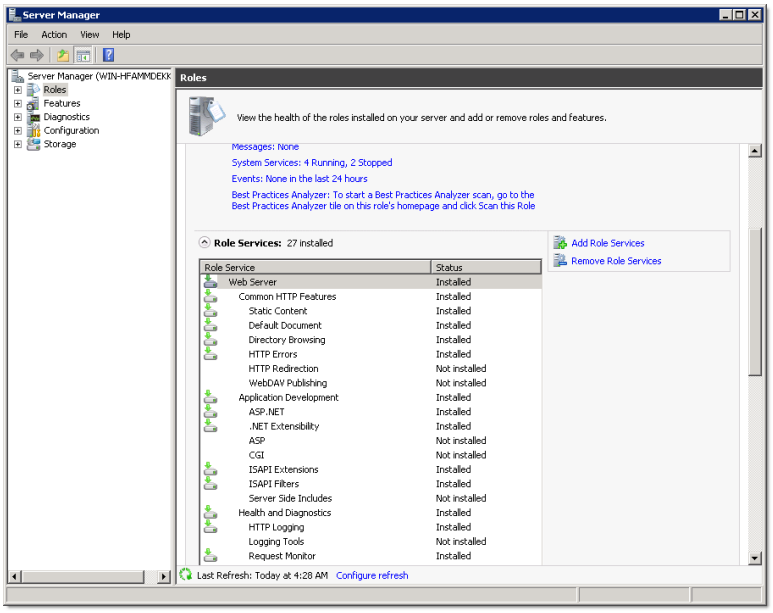
   c. In the left pane, select Server Roles. Select the checkboxes as shown below, and click OK.



Verifying the Features Installed in IIS

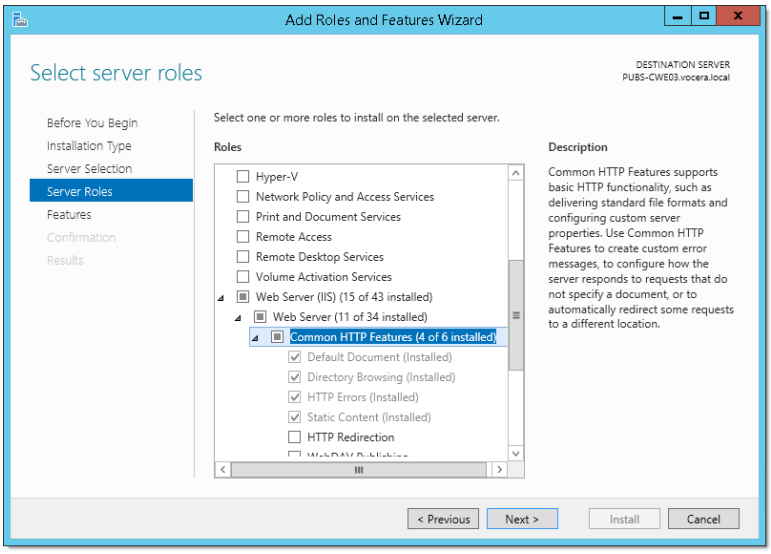
Follow these steps to verify that the IIS feature installation is correct.

1. On Windows Server 2008:
   a. Select Start > Control Panel.
   b. Select Programs and Features.
   c. Select Turn Windows features on or off.
   d. Select Roles. Wait for the data to appear.
   e. View the Role Services pane. Ensure that Default Document, Directory Browsing, HTTP Errors, and Static Content are installed.



2. On Windows Server 2012:
   a. In the Server Manager, select Add roles and features to open the Add Roles and Features Wizard. Click Next.
   b. Select Role-based or feature-based installation. Click Next.
   c. Select the server and click Next.

d. Expand Web Server (IIS), Web Server, and Common HTTP Features. Ensure that Default Document, Directory Browsing, HTTP Errors, and Static Content are selected.

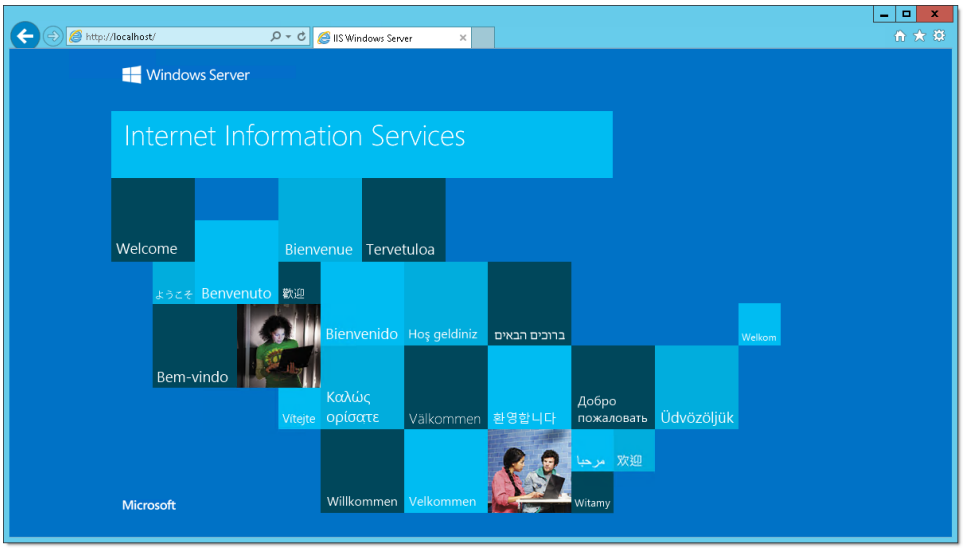


e. Click Cancel to exit the wizard, as you do not need to add any IIS features at this time.

Verifying IIS Operation

To verify that IIS is working, type `http://localhost` in your browser.

If IIS is working properly, the IIS page is displayed:



In the `C:\Windows\Microsoft.NET\Framework64` folder, ensure that subfolders are defined for v2.0 and v4.0. These subfolder names may include more complete version information that includes the build number - for example, v2.0 may be indicated by the folder name `v2.0.50727`.

Verifying ASP.NET

Follow these steps to ensure that ASP.NET is working properly.

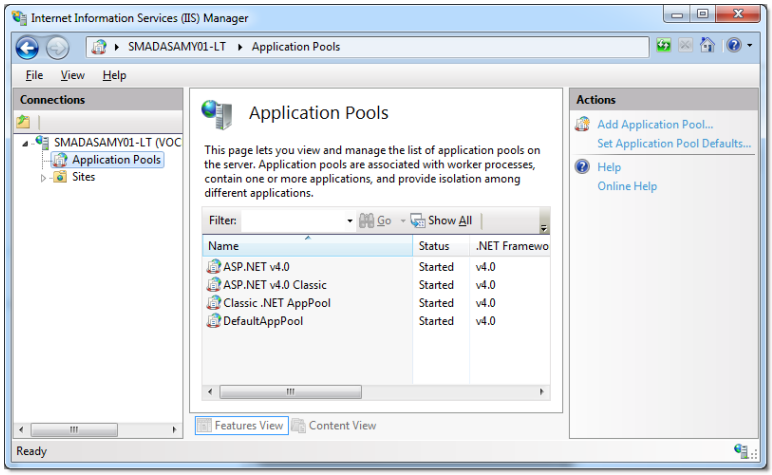
First, in the Windows registry, search for the `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\InetStp\Components` registry key. This is the IIS Setup key that contains the components that have been enabled in IIS.

- Click the Windows Start key and click Run. In the Run window, type `regedit.exe` and click OK. This displays the Windows registry.

- Locate the `HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\INetStp\Components` registry key.
- If you are using ASP.NET 4.5, ensure that ASPNET45 and NetFxExtensibility45 have been set to 1.
- If you are using an older version, ensure that ASPNET, NetFxEnvironment, and NetFxExtensibility have been set to 1.

Next, check whether the current .NET Framework version is 4.0. To do this on Windows Server 2008:

- Go to IIS Manager.
  Click Start > Administrative Tools > Internet Information Services (IIS) Manager.
- Click on the name of your server to display the Actions Panel.
  If a dialog box appears asking you whether you want to get started with Microsoft Web Platform, click No.
- In the Actions Panel, click Change .NET Framework Version. A dialog box appears, displaying the version of .NET Framework that is currently in use.
- Verify that the version starts with 4.0, and click OK to close the dialog box.
- In the Actions Panel, click View Application Pools to check whether the application pools .NET Framework version is 4.0:



- If you can't find ASP.NET or .NET, open a Command Prompt window in administrator mode and run the following command from the folder in which you have installed .NET Framework:
  `C:\Windows\Microsoft.NET\Framework\v4.0.30319>aspnet_regiis -I`
  Replace `v4.0.30319` with your version of .NET Framework. Restart your server after completing this command.

If you are using Windows Server 2012, see **IIS 8.0 Using ASP.NET 3.5 and ASP.NET 4.5** for details on how to verify that .NET Framework 4.0 is installed on your server.

Installing the VAM 2.2.5 Server

This section describes how to install version 2.2.5 of VAM.

This patch can be installed on top of VAM 2.2.4.2.

Important: Before installing VAM 2.2.5, you must move any data in the `C:\Vocera \mVisum Alerts\MVisum Alerts Service\Data` folder to another location. You must also close all opened files.

You must have the required prerequisites before installing VAM. See **Installing the Prerequisites** on page 12 for details.

Important: If you update the VAM server, you must clear your browser cache before continuing to use the VAM Console. This ensures that you are working with the updated data.

To install version 2.2.5 of VAM:

1. Run the VAM 2.2.5 installation script that has been provided to you.
2. When prompted, select the directory in which you have installed VAM.
3. The installer stops the IIS Server and Vocera Alarm Media Parser services. Confirm that these services are stopped before continuing.
4. When the installer is completed, it restarts the IIS Server and Vocera Alarm Media Parser services. Ensure that these services are running.

After installation, use the following URLs to determine whether the VAM Console and the Alert Service are accessible. Replace `<ip>` with the IP address of the computer on which you have installed VAM.

- VAM Console: `http://<ip>/voceraalarmadmin/login.aspx`
- Alert Service: `http://<ip>/MVisumAlertService/MvisumAlertService.asmx`

Post-Installation Configuration

After you have installed Vocera Alarm Management, perform the tasks described here.

Ports

The following ports are used in the VAM application.

Port	Purpose
80	HTTP
443	HTTPS (SSL)
6661	Mirth
1433	SQL Server (default)
3306	MySQL (default if MySQL is used for Analytics)
587 or 25	Smtp.gmail.com (optional)

Installed Services

This is the list of Windows services and IIS services installed with the VAM server.

List of installed Windows services:

- Vocera Alarm Clean Up Service
- Vocera Alarm Escalator
- Vocera Alarm Media Parser
- Vocera Alarm Message Generator
- Vocera Alarm Monitoring
- Vocera Alarm Notification Service
- Vocera Alarm Server Health Monitor
- Vocera Archiver Service

IIS services installed with the VAM Server:

- MobileDownloads
- MVisumAlertService
- MVisumPushServer
- VoceraAlarmAdmin

- VoceraAlarmTestApp

Creating a Shared Folder for Waveform Data

After you have installed VAM, you must create a shared folder on any of the VAM server instances or on any other common system to store waveform data.

Make sure that this shared folder is accessible from all VAM instances. Use the user credentials provided during installation to create the shared folder.

Note: If you do not specify this shared folder, your VAM installation may not work as expected.

1. Open the configuration file `<Vocera>\mVisum Alerts\MVisum Message Generator \MVisumMessageGenerator.exe.config`.
2. Set the DataFolderPath key to be a subfolder of the shared folder that you have just created. For example, if the shared folder is in `C:\MyFolder\SharedFolder`, and you want to store waveform data in the `Data` subfolder, use the following definition:

   ```
   <add key="DataFolderPath" value="C:\MyFolder\SharedFolder\Data"/>
   ```
3. Open the configuration file `<Vocera>\mVisum Alerts\MVisum Alerts Service \web.config`.
4. Set the DataFolderPath key to be the `Data` subfolder described in step 2.
5. Repeat these changes in each VAM instance.

Determining the Waveform Stream Location

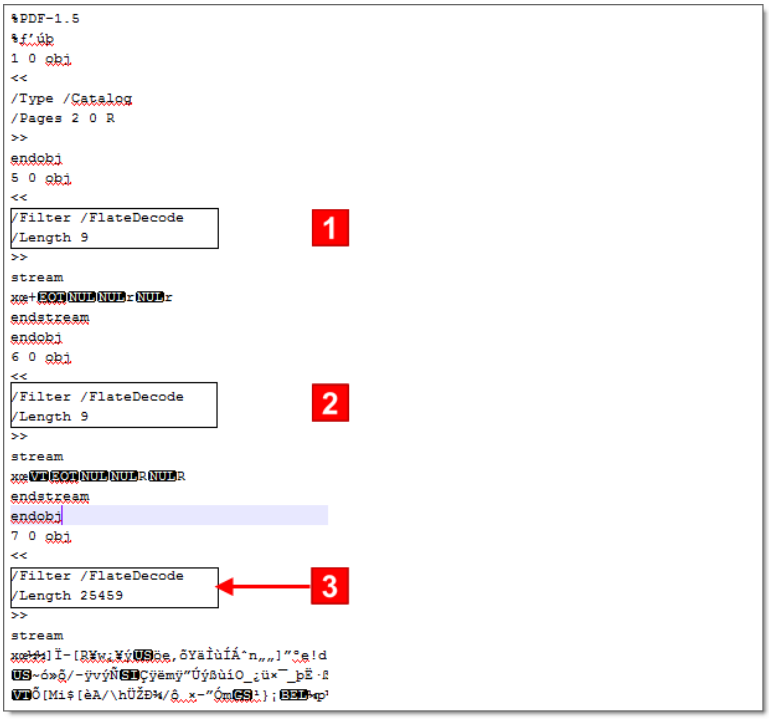
If waveform data PDF files contain waveforms in somewhere other than the first stream location, the media parser configuration file needs to be edited to ensure that handheld clients can view the waveforms.

When an alarm is generated from BedMaster, the resulting HL7 file includes waveform data in Base64 format, which the VAM message generator extracts and saves in a PDF file. Usually, this waveform data is in the first stream location, but it may be located elsewhere.

By default, the VAM media parser looks in the first stream location. If the waveform data is not located there, the media parser is not able to generate the waveform file (of type .mdat) that the handheld client is expecting.

To determine the stream location of the waveform in a PDF file, open the file in Notepad or Notepad++. Each stream location in the file is indicated by the text line `/Filter /FlateDecode` followed by a text line consisting of `/Length` and a number that indicates the stream length. The waveform is located at the first stream location for which the stream length is greater than 25000.

Here is an example of a PDF file in which the waveform data is located in the third stream location:



Here, the stream locations labeled 1 and 2 are of length shorter than 25000, and therefore do not contain the waveform data. The stream location labeled 3 is where the waveform data is located.

Updating the Media Parser Configuration File

When you have determined the stream location, you can update the media parser configuration file to indicate where to look for the waveform data.

To do this, edit the file `<Vocera>\mVisumAlerts\Mvisum Media Parser \MVisumMediaParser.exe.config` (where `<Vocera>` is the folder in which the VAM is installed). Edit the value of the `WaveformStreamLocation` key:

```
<add key="WaveformStreamLocation" value="<number>"/>
```

Replace `<number>` with the stream location of the waveform file. In the example above, the waveform is in the third stream location, so `<number>` would be 3. By default, `WaveformStreamLocation` is set to 1.

If the waveform is split into multiple stream locations, you can specify this in the `WaveformStreamLocation` key. For example, if the waveform is contained in the first three stream locations, set `WaveformStreamLocation` as follows:

```
<add key="WaveformStreamLocation" value="1,2,3"/>
```

Configuring the Archive Service

To store archive data, create a shared folder on any one of the VAM server instances or on any other common system.

Make sure that this shared folder is accessible from all VAM instances. Use the user credentials provided during installation to create the shared folder.

1. From your web browser, type the URL `http://<address>/VoceraAlarmAdmin/login.aspx`, where `<address>` is the IP address of your VAM server. This displays the the VAM Console.
2. Log into the VAM Console with the default username, `webuser`. The password is also `webuser`.

3. Click Account Settings, and click the Miscellaneous tab.
4. In the Archive Path field, set the archive path to the shared folder that you have just created.
5. Click Save.
6. Repeat these changes in each VAM instance.

The next step is to configure the Vocera Archiver service:

1. In the Control Panel, select Services.
2. Double-click Vocera Archiver Service to open this service.
3. Click the Log On tab, and select the This account radio button. Specify your user credentials, and click OK.
4. Restart the service.

Verifying the Monitoring Service Heartbeat Interval

After doing a fresh installation or an upgrade, go to the `<Vocera>\mVisumAlerts \MVisumMonitoring` folder and verify that the tags shown here are defined in the `MVisumMonitoring.exe.config` file.

```
<add key="HeartbeatInterval" value="5" />     <!--In Minutes-->
<add key="Sleep" value="10" />     <!--In Seconds-->
```

When these settings are added, the heartbeat email is sent if a .JSON file is not received in the time specified in `HeartbeatInterval`. Change the values for these settings as appropriate for your environment.

Changing the IIS Configuration

Follow these steps to make the necessary IIS configuration changes.

1. Open the IIS configuration file and navigate to the MVisumAlertService web application.
2. Edit the application by selecting Basic Settings from the right side action panel.
3. Click Connect As and use the same user credentials as before.
4. Restart the IIS service if required.

Next, you must configure the Windows services:

1. In the Control Panel, select Services.
2. Double-click the Vocera Alarm Message Generator service to open it.
3. Click the Log On tab, and select the This account radio button. Specify your user credentials, and click OK.
4. Repeat the above steps for the Vocera Alarm Media Parser service.
5. Restart both services.

Windows Services Description

This table provides a list of the Windows services defined for VAM.

Services	Description
mVisumAlertService	The alert service provides services to the client, such as registration, authentication, getting alert details, acknowledgement of the alarms, bed assignment, and waveform data.

Services	Description
Vocera Alarm Clean Up Service	Multiple cleanup tasks are performed by the cleanup service. • Bed Patient Cleanup: If no new alert or vital sign has been received for a bed for a configured time, the patient assigned to that bed is removed. • Forward Cleanup: Clean the forwarding applied by users if the maximum forwarding time has elapsed. • File Cleanup Thread: Clean the file whose age is older than the configured time. • Session Cleanup: Close sessions which have been inactive for longer than the session timeout interval. • Index Rebuild Thread: Rebuild the index after the specified time interval.
Vocera Alarm Media Parser	The Media Parser handles alert attachment (ECG) files of various formats, including PDFs, JPEGs, and other vector graphics formats. It receives the files from the file storage and processes them for presentation on the handheld user interfaces.
Vocera Alarm Escalator	The Escalator maintains a running account of open alarms, including new alarms that it reads from the database. It applies all escalation rules and informs the Notification server of any escalations to be sent out.
Vocera Alarm Message Generator	• The message generator receives messages from the HL7 engine for any new alarm, parses them, and enters alert data into databases. • Based on rules defined in the database, obtains a list of nurses to which an alert notification needs to be sent. • Immediately escalates the alert if it was not delivered to a nurse. • Processes vital signs and updates patient information in the database. • After all the relevant data regarding the alert is fed to the database, the message generator sends the important details of the alarm to the Notification Server component.
Vocera Alarm Monitoring	Monitors whether alerts are arriving. If no new alert has been received for a configured time, this service sends an email indicating that no new alert received during this time period.
Vocera Alarm Notification Service	• The notification client component of the notification server notifies the push server that new alarms are present. • The service component of the notification server checks for any undelivered alarms and tries to resend them.
Vocera Alarm Server Health Monitor	A Windows service that monitors the health of the server. It checks that all applications of the VAM server are running. This service sends a daily health report containing information on the messages generated in the last 24 hours.
Vocera Archiver Service	This module is responsible for archiving the alarm data, receivers, and the processed HL7/JSON files. It uses a cron job to schedule the date and time for the archive.

Disabling Windows Authentication for Services

You must disable Windows authentication for some services in all VAM instances.

You must disable the Windows authentication mode for the following virtual directories in IIS in all VAM instances:

- VoceraAlarmAdmin
- MVisumPushServer
- MVisumAlertService
- VoceraAlarmTestApp

To do this, follow these steps:

1. Open the IIS Manager.
2. Expand the tree next to your server name, then expand the Sites and Default Web Site subtrees. This displays a list of your IIS applications.
3. Under Default Web Site, click VoceraAlarmAdmin.

4. Click the Features View tab.
5. In the IIS section, select Authentication.
6. Disable Windows Authentication mode if it is defined.
7. Repeat these steps for the MVisumPushServer, MVisumAlertService, and VoceraAlarmTestApp services.

You must restart the following Windows services in all instances of VAM after you have made these changes:

- Vocera Alarm Message Generator
- Vocera Alarm Escalator

Deleting Inactive Users

Follow these steps to remove deactivated users.

1. In the SQL Server Management Studio, access the MVisumAlerts database.
2. Open the new query window and ensure that the MVisumAlerts database is selected.
3. Run the stored procedure described below, if it is not already there.
4. Make a backup of your database.
5. Execute the [DeleteInActiveUsers] command as EXEC [dbo].

Here is the stored procedure to delete the inactive users:

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[DeleteInActiveUsers]
as
BEGIN
 DECLARE @INActiveUsers Table(
  UserId Int
 )
 INSERT @INActiveUsers SELECT UserId FROM Nurse WHERE [Status]='INACTIVE'
 --Delete Group level Assignments for inactive users
 DELETE FROM GroupAssignments WHERE UserID in(SELECT UserId FROM @INActiveUsers)
 --Delete Bed Level Staff Assignments for inactive users
 DELETE FROM BedLevelStaffAssignment WHERE UserId in(SELECT UserId FROM @INActiveUsers)
 --Delete Bed Assignments for inactive users
 DELETE FROM BedAssignments WHERE PrimaryUserID in(SELECT UserId FROM @INActiveUsers)
          OR SecondaryUserID  in(SELECT UserId FROM @INActiveUsers)
          OR Secondary2UserID  in(SELECT UserId FROM @INActiveUsers)
          OR Secondary3UserID  in(SELECT UserId FROM @INActiveUsers)
          OR Secondary4UserID  in(SELECT UserId FROM @INActiveUsers)
          OR Secondary5UserID  in(SELECT UserId FROM @INActiveUsers)
 --Delete Site Level Staff Assignments for inactive users
 DELETE FROM SiteLevelStaffAssignment WHERE UserId in(SELECT UserId FROM @INActiveUsers)
 --Delete Unit Assignments for inactive users
 DELETE FROM UnitAssignments WHERE UserID in(SELECT UserId FROM @INActiveUsers)
          OR AllUsers  in(SELECT UserId FROM @INActiveUsers)
 --Delete Unit Assignments Logs for inactive users
 DELETE FROM UnitAssignmentLogs WHERE UserID in(SELECT UserId FROM @INActiveUsers)
          OR AllUsers  in(SELECT UserId FROM @INActiveUsers)
 --Delete Unit Level Staff Assignments for inactive users
 DELETE FROM UnitLevelStaffAssignment WHERE UserId in(SELECT UserId FROM @INActiveUsers)
 --Delete User Forwarding for inactive users
 DELETE FROM UserForwardings WHERE ForwardedBy in(SELECT UserId FROM @INActiveUsers)
         OR ForwardedTo in(SELECT UserId FROM @INActiveUsers)
 --Delete Nurses which are inactive
 DELETE FROM Nurse WHERE UserID in(SELECT UserId FROM @INActiveUsers)
 --Delete Admin users which are Inactive
 DELETE FROM SuperAdmin WHERE UserID in(SELECT UserId FROM @INActiveUsers)
 --Delete User Hospital Mapping for inactive users
 DELETE FROM UserHospitals WHERE UserId in(SELECT UserId FROM @INActiveUsers)
 --Delete User Role Mapping for inactive users
 DELETE FROM UserRoles WHERE UserId in(SELECT UserId FROM @INActiveUsers)
 --Delete User which are  inactive
 DELETE FROM Users WHERE UserId in(SELECT UserId FROM @INActiveUsers)
END
```

Escalation Check Interval

In the current installation, the escalation check interval is set in the database as described here.

AlertLevel	AlertMessage	EscalationCheckInterval (in sec)
1	Patient Crisis	1
2	Patient Warning	1
3	Patient Advisory	120
4	Patient Message	120
5	System Warning	1
6	Vocera Alert	120

The AlertLevels table in the mVisumAlerts database can be modified as required. If any parameter in the database is changed, the Vocera Alarm Escalator service in all VAM instances needs to be restarted.

Health Monitor Service

The health monitor service collects alarm parameters at the unit level on a (configurable) 24-hour basis and sends email to the recipients specified during configuration.

The alarm parameters collected are:

- Number of Alarms sent
- Last Alarm sent
- Last test Alarm sent
- Number of Alarms viewed in the interval
- Number of Alarms acknowledged in the interval
- Last Alarm viewed
- % of Disk full

The server health monitor service also sends an email notification services status, indicating whether it is running or not. For each service, this status is either Accessible / Running or Not Running.

Web Services

- Vocera Alarm Database
- Vocera Notification Database
- Vocera Interface Database
- Vocera Push Database
- Vocera Alarm Web service
- Vocera Admin Website

Windows Services

- Vocera Alarm Escalator
- Vocera Alarm Media Parser
- Vocera Alarm Message Generator
- Vocera Alarm Notification Service
- Vocera Alarm Monitoring
- Vocera Alarm Message Generator
- Vocera Archiver Service

Directories and Databases to Monitor

To keep VAM services running properly, you must ensure that its directories and databases do not exceed the limits described here.

In the table below, replace `<Vocera>` with the path of the folder in which you have installed the VAM server.

Table 3: Directories

Description	Location	Max Size
HL7	`<Vocera>\mVisum Alerts\MVisum Message Generator\HL7`	0 (Each file should be processed)
Processed vitals and alarms by date	`<Vocera>\mVisum Alerts\MVisum Message Generator\Vitals\Processed`	2 GB
Error files by date	`<Vocera>\mVisum Alerts\MVisum Message Generator\Vitals\Errored`	100 MB
Vitals error files by date	`<Vocera>\mVisum Alerts\MVisum Message Generator\Errored`	100 MB
Waveform data	See **Creating a Shared Folder for Waveform Data** on page 21 for details on this folder.	2 GB
Archive data	See **Configuring the Archive Service** on page 22 for details on this folder.	10 GB
Log files	`<Vocera>\mVisum Alerts\Logs`	1 GB
Unprocessed files by date	`<Vocera>\mVisum Alerts\MVisum Message Generator\UnProcessed`	100 MB

Table 4: Size of Database

Database Name	How to Access	Max Size
MvisumAlerts	Right-click Database -> General	1 GB
MVisumInterface	Right-click Database -> General	200 MB
MVisumNotification	Right-click Database -> General	200 MB
MVisumPush	Right-click Database -> General	200 MB

Table 5: Rows in Tables

Table	How to Access	Max Size
Alerts	`Select count(*) from Alerts`	1,000,000 records
AlertDetails	`Select count(*) from AlertDetails`	1,000,000 records
Receiver	`Select count(*) from Receiver`	10,000,000 records
MVisumErrors	`Select count(*) from MVisumErrors`	5,000,000 records
MessageAttachments	`Select count(*) from MessageAttachments`	1,000,000 records

Table 6: CPU Utilization

Server	Utilization
SQL server system	Up to 70%
VAM server	Up to 70%

Table 7: Memory consumption for each service

Service	Memory
Vocera Alarm Clean Up Service	50 MB
Vocera Alarm Escalator	70 MB
Vocera Alarm Media Parser	50 MB
Vocera Alarm Message Generator	70 MB
Vocera Alarm Monitoring	50 MB
Vocera Alarm Notification Service	40 MB

Service	Memory
Vocera Alarm Server Health Montior	40 MB
Vocera Archiver Service	70 MB

Accessing the Console Using SSL

You can specify whether to access the Vocera Alarm Management Console using SSL (whether the URL is to contain HTTPS or HTTP).

1. In the Vocera Alarm Management installation folder, navigate to the `MvisumAlertsAdmin` subfolder.
2. Edit the `web.config` file.
3. To require SSL, set the following parameter:
   `<httpCookies httpOnlyCookies="true" requireSSL="true"/>`
   When this is set, users must include `https` in the VAM Console URL.
4. To not require SSL, set the parameter to `false`:
   `<httpCookies httpOnlyCookies="true" requireSSL="false"/>`
   Users must now include `http` in the VAM Console URL.

Uninstalling the VAM Server

There are two ways to uninstall the Vocera Alarm Management server: using the Control Panel Add/Remove option, or using the uninstall utility.

Removing VAM from the Control Panel

- From the Control Panel, select Add/Remove.
- Select Vocera Alarm Management.
- Right-click and select Uninstall, and confirm that you want to uninstall.
- Once confirmed, the system will remove the VAM server.

Removing VAM using uninstaller.exe

- Go to the directory in which Vocera Alarm Management is installed.
- Run `uninstall.exe` to remove the VAM server.

Note: Use Run as administrator for the uninstaller. Close any open file in the VAM server.

Mirth Connect Installation and Configuration

These sections provide information on how to install and configure Mirth Connect for Vocera Alarm Management. Use Mirth Connect to receive messages from a patient monitor gateway and transform them into the JSON format that the VAM server understands.

Prerequisites

The Mirth Connect server requires the prerequisites shown here.

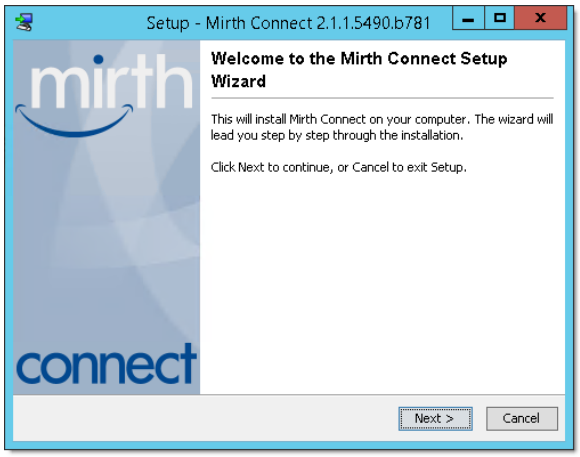
- Java JRE version 1.7.0_45 or above (64-bit version) must be installed.

Installing Mirth Connect

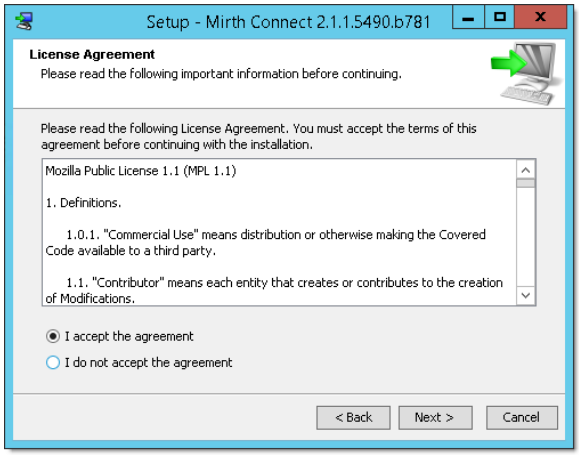
Follow these steps to install Mirth Connect.

Vocera recommends using Mirth Connect version 2.1.1. This version of Mirth Connect is available from the **Mirth Connect archived downloads page**.

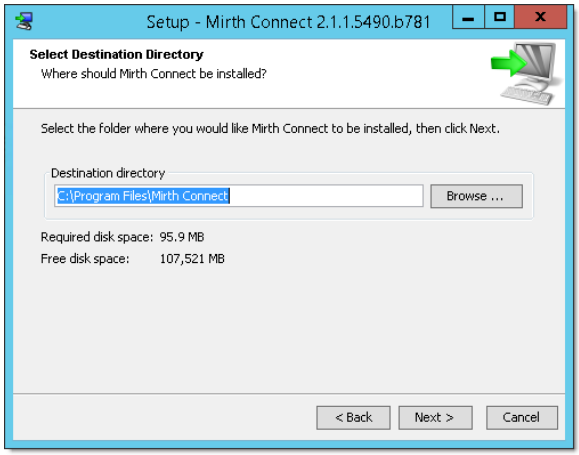
1. Launch the installer as the Administrator.
2. Click Next to start the installation.



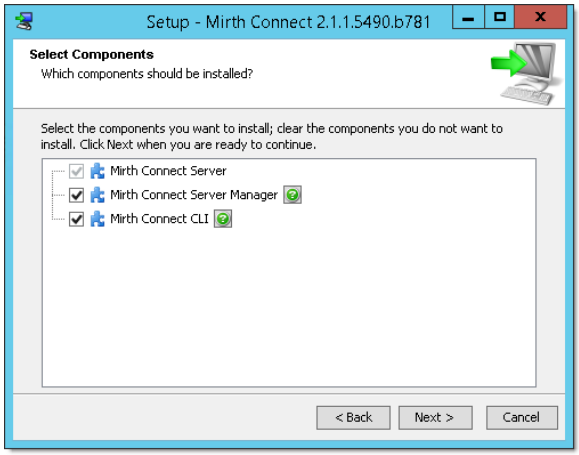
3. Click Next to accept the license agreement.



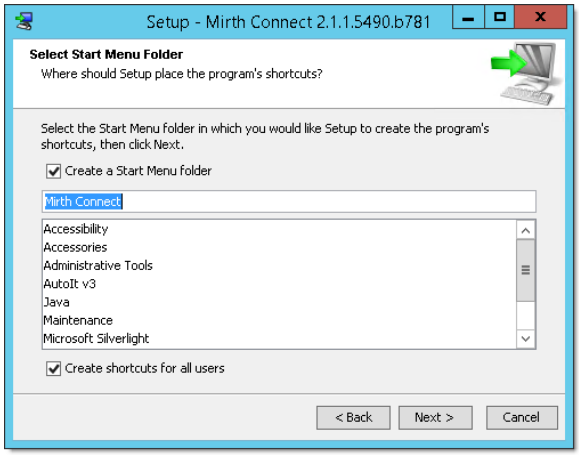
4. Type or browse for the destination directory. The default is `C:\Program Files\Mirth Connect`.



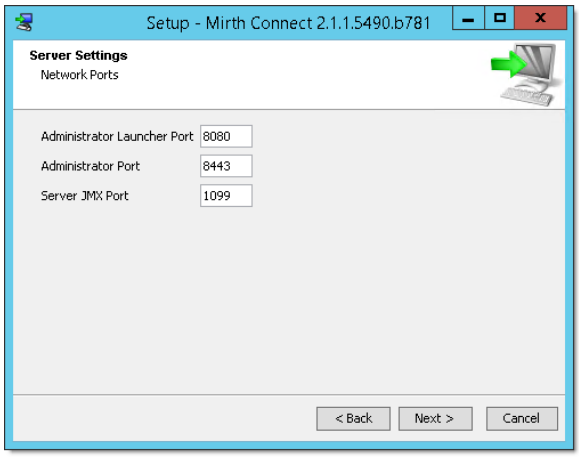
5. Select Mirth Connect Server Manager and Mirth Connect CLI. Mirth Connect Server is selected by default. These are the default Mirth Connect components. Click Next.



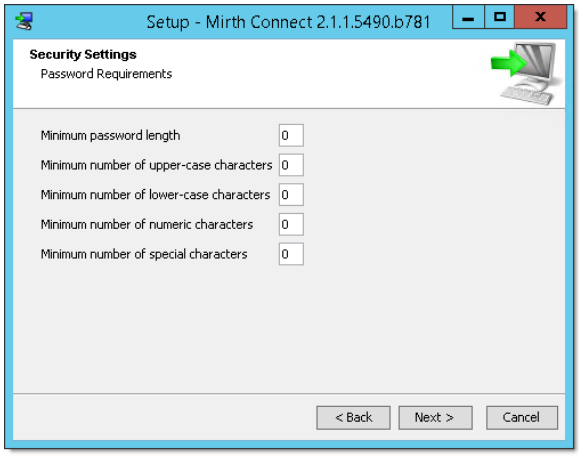
6. Select Create a Start Menu folder. Click Next.



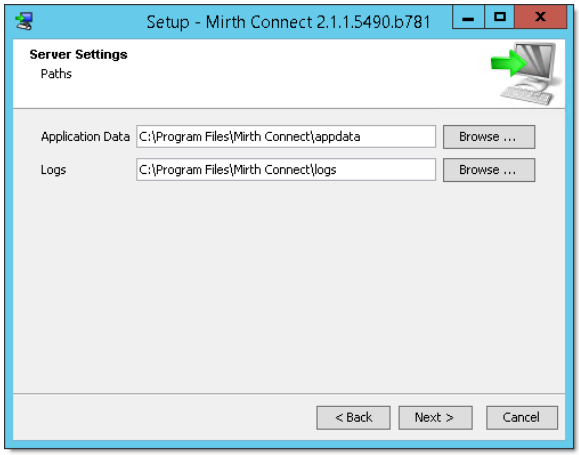
7. Specify the port configuration as shown below. These are the default port numbers for the Mirth Connect services. Click Next.



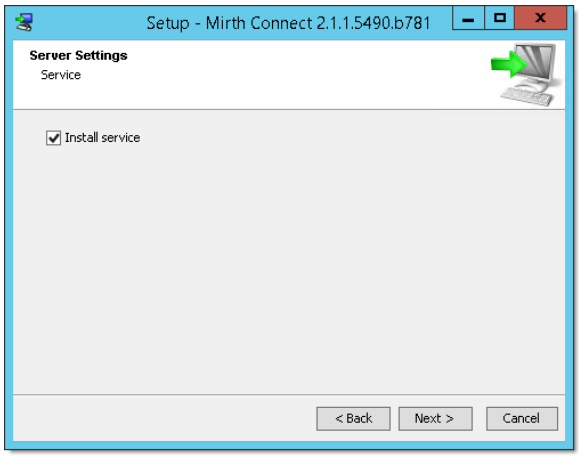
8. Specify the password security requirements to conform to your password policy. Click Next.



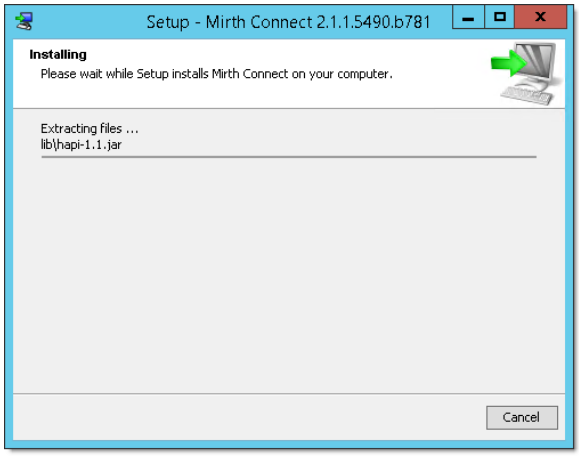
9. Use the default Application Data and Logs folders. Click Next.



10. Select Install service to install the Mirth Connect service on your Windows environment. Click Next.



11. Wait while the installer finishes adding the necessary files.



12. Click Finish to start the Mirth Connect server.

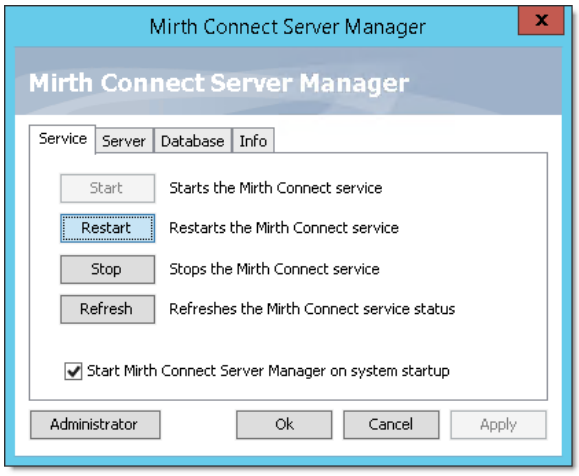
The Mirth Connect Server Manager icon now appears in the system tray.

Setting up the Server Manager

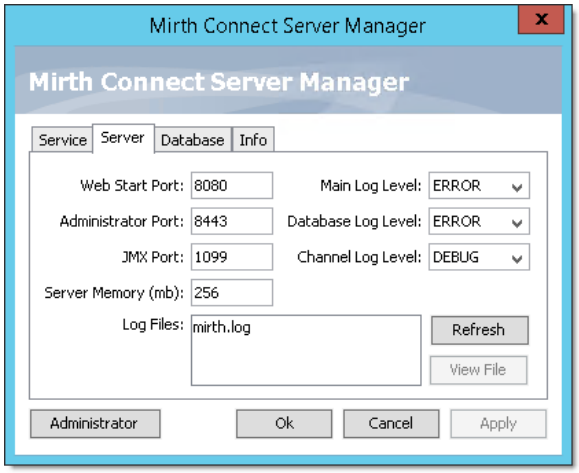
Follow these steps to set up the Mirth Connect Server Manager.

1. Double-click the Mirth system tray icon or right-click the tray icon. Click Show Manager to open the Mirth Connect Server Manager.

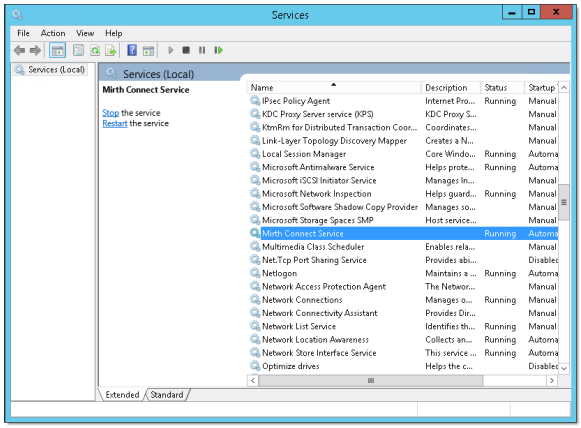
2. The Service tab enables you to start, stop, and restart the Mirth Connect Service or refresh its service status. Select the Start Mirth Connect Server Manager on system startup checkbox.



3. Click the Server tab. Set the ports as needed, or use the ports described below.



4. Leave all other settings as their default values. Click Ok.
5. Restart the Mirth Connect Service (this is mandatory).



You will receive a notification that the service has restarted successfully.

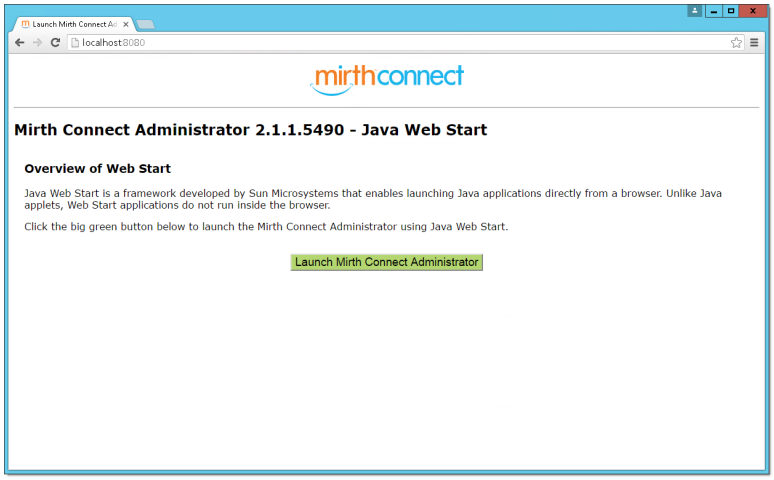
Launching the Mirth Connect Administrator

There are two ways to launch the Mirth Connect Administrator.

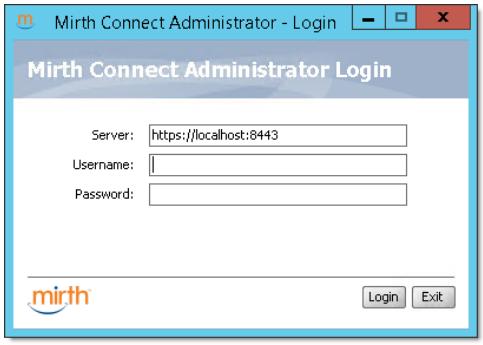
One way is to select Launch Administrator from the system tray icon:

The best way to launch the administrator is to use its URL.

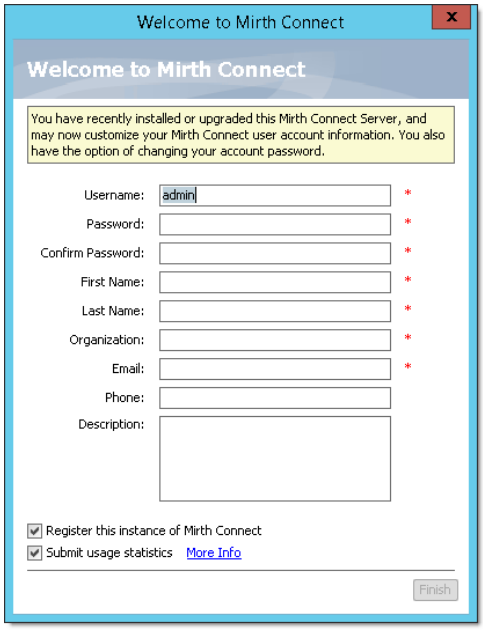
1. Type `http://localhost:8080` in your browser. In the Java Web Start page, click Launch Mirth Connect Administrator



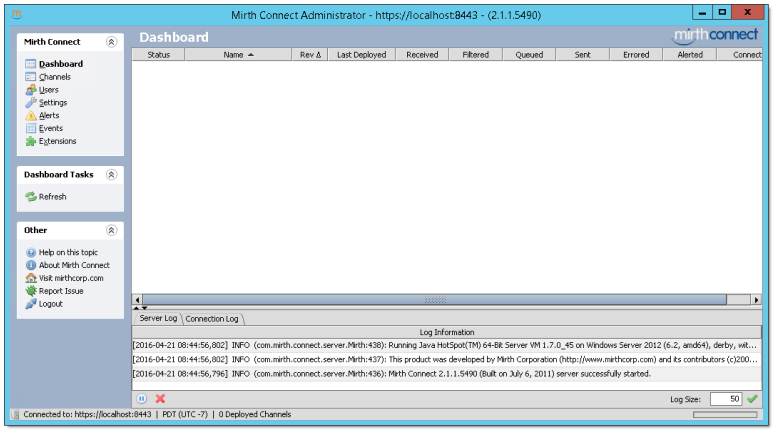
2. If you are asked whether you want to run this application, click Run.
3. When the Mirth Connect Administrator Login screen appears, specify the default login username, which is **admin**. The password is also **admin**. Click Login.



4. When you log in for the first time, you are prompted for registration information, including a new password. Supply this information.



5. Click Finish. The Mirth Connect dashboard appears.



Importing an Existing Channel

If you have already created and exported a channel, you can import it.

1. In the Mirth Connect Administrator, in the Mirth Connect pane at the top left of the screen, select Channels.
2. Select Import Channel.
3. On your computer, locate the `.xml` file containing your exported channel.
4. Verify that the source and destination settings are correct. Save the channel.
5. Follow the steps in **Deploying a Channel** on page 36 to deploy the channel.

The channel is now ready to process messages.

Deploying a Channel

After you have created a channel, you must deploy it.

1. In the Mirth Connect Administrator, in the Mirth Connect pane at the top left of the screen, select Channels. A list of configured channels appears.
2. Right-click on the channel that you want to deploy. From the popup menu that appears, select Deploy Channel.

The Dashboard now displays the deployed channel.

Default Ports

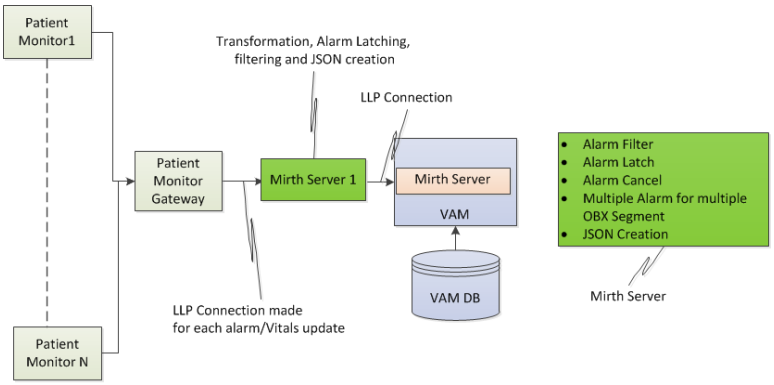
This table lists the default ports that are configured in Mirth channels.

Channel	Port
Monitor	6661
BMReceiver	6662

Configuring Mirth Channels for a Single Server

If you have deployed a single VAM server, you must configure the Mirth channels that it requires.

Vocera recommends installing Mirth on the server on which you have installed VAM:



Architecture Diagram- Single instance of VAM Server with Philips Patient Monitor

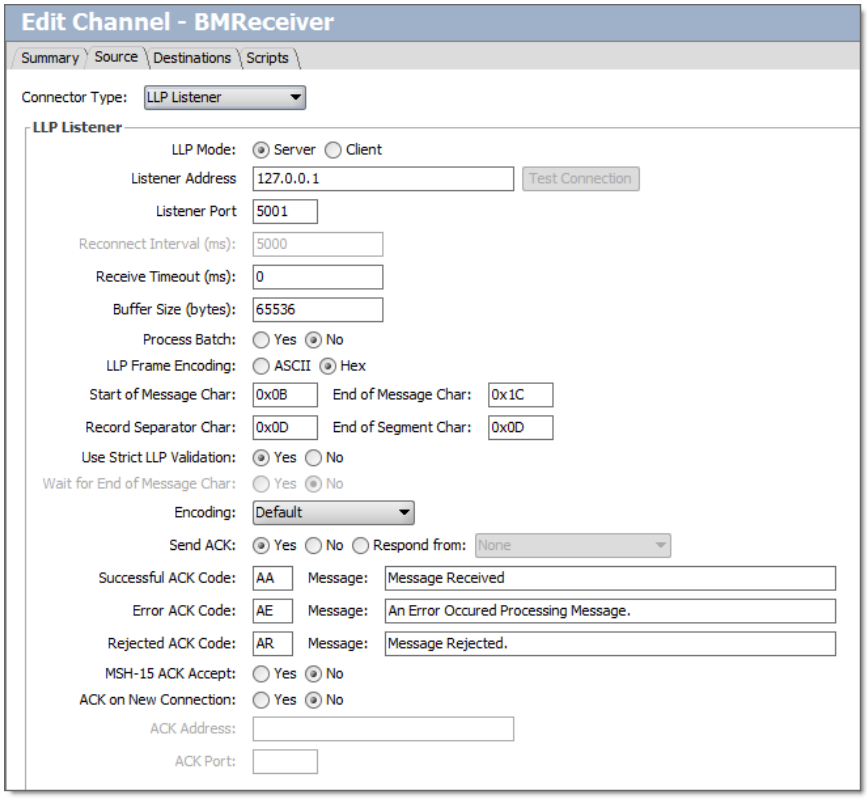
On this Mirth server, you must create the BMReceiver channel, which receives HL7 data from Bedmaster and converts it to the JSON file format VAM understands.

The file for this channel is provided in XML format in the folder that contains the VAM installer. Follow the steps in **Importing an Existing Channel** on page 36 to import it into your Mirth server. If you do not have access to the channel in XML format, use the instructions in the following section to configure the BMReceiver channel.

The BMReceiver Channel

For the BMReceiver channel, you must make these modifications.

1. In the Mirth Connect Administrator, in the Morth Connect pane, select Channels. The list of channels is displayed.
2. Right-click BMReceiver and select Edit Channel.
3. Update the Source tab as shown:



4. In the Edit Channel screen, click the Destinations tab.
5. In the Connector Type dropdown list, select File Writer.

6. In the Directory field, type `<Vocera>\mVisumAlerts\MVisum Message Generator\HL7`, where `<Vocera>` is the folder in which VAM is installed. This is where the message generator obtains the JSON files.
7. In the File Name field, type `${UUID}.json`. This generates a unique ID for each file name, and ensures that the file extension is `.json`.
8. From the Append to file radio buttons, select Yes.
9. In the Template field, add the following:

```
{
  "VitalData": {
    "SendingApplication": "${SendingApplication}"
    ,"SendingFacility": "${SendingFacility}"
,"DateTimeOfMessage": "${DateTimeOfMessage}"
    ,"MessageControlID": "${MessageControlID}"
    ,"ProcessingID": "${ProcessingID}"
    ,"PatientIDExternalID": "${PatientIDExternalID}"
    ,"PatientInternalId": "${PatientInternalId}"
,"PatientGivenName": "${PatientGivenName}"
,"FamilyName": "${FamilyName}"
,"PatientClass": "${PatientClass}"
,"PointOfCare": "${PointOfCare}"
,"Bed": "${Bed}"
,"AdmissionType": "${AdmissionType}"
,"UniversalServiceIdentifier": "${UniversalServiceIdentifier}"
,"ObservationDateTime": "${ObservationDateTime}"
,"ObservationEndDateTime": "${ObservationEndDateTime}"
,"RelevantClinicalInformation": "${RelevantClinicalInformation}"
```

```
 ,"DiagnosticServiceSectionID": "${DiagnosticServiceSectionID}"
 ,"ParentResult": "${ParentResult}"
 ,"AdmitDateTime": "${AdmitDateTime}"
 ,"Vitals": ${Vitals}
  }
}

{
  "AlarmInfo": {
 "SendingApplication": "${SendingApplication}"
 ,"SendingFacility": "${SendingFacility}"
 ,"DateTimeOfMessage": "${DateTimeOfMessage}"
 ,"MessageControlID": "${MessageControlID}"
 ,"ProcessingID": "${ProcessingID}"
 ,"PatientInternalID": "${PatientInternalId}"
 ,"PatientGivenName": "${PatientGivenName}"
 ,"FamilyName": "${FamilyName}"
 ,"DateOfBirth": "${DateOfBirth}"
 ,"PatientClass": "${PatientClass}"
 ,"PointOfCare": "${PointOfCare}"
 ,"Bed": "${Bed}"
 ,"AdmissionType": "${AdmissionType}"
 ,"UniversalServiceIdentifier": "${UniversalServiceIdentifier}"
 ,"ObservationDateTime": "${ObservationDateTime}"
 ,"ObservationEndDateTime": "${ObservationEndDateTime}"
 ,"RelevantClinicalInformation": "${RelevantClinicalInformation}"
 ,"DiagnosticServiceSectionID": "${DiagnosticServiceSectionID}"
 ,"ParentResult": "${ParentResult}"
 ,"QuantityTiming": "${QuantityTiming}"
 ,"AlarmLevel": "${AlarmLevel}"
 ,"AlarmReason": "${AlarmReason}"
 ,"ObservResultStatus": "${ObservResultStatus}"
 ,"userDefinedAccessChecks": "${AttachmentDate}"
 ,"AttachmentType": "${AttachmentType}"
 ,"ControlId": "${ControlId}"
 ,"MessageType": "${MessageType}"
 ,"VersionId": "${VersionId}"
 ,"Waveform": "${Waveform}"
   ,"Vitals": ${Vitals}
  }
}
```

10. Click Save Changes to save the changes.
11. Right-click on the channel and select Deploy Channel to redeploy the changed channel.

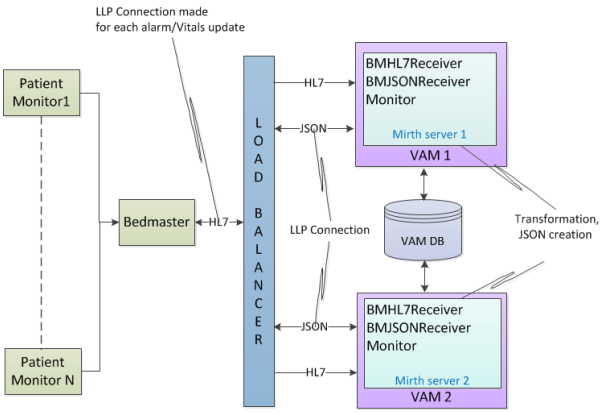
High Availability Configuration

You can configure VAM to implement a high-availability solution.

To implement high availability for VAM, do the following:

- Install the VAM server on two nodes, VAM 1 and VAM 2.
- Install a load balancer to serve as the interface between the Bedmaster input and the VAM servers.
- Install Mirth Connect on both VAM servers.
- Deploy the following channels on each Mirth server:
  - BMHL7Receiver, which receives an HL7 input file from the Bedmaster via the load balancer, and sends the converted JSON file output back to the load balancer. The JSON file can then be processed by either VAM server.
  - BMJSONReceiver, which uses the JSON file to generate alarms in VAM.
  - Monitor, which receives notification pings from the load balancer to verify the health of VAM.

This architecture diagram shows how VAM is configured for a high availability environment:



Architecture Diagram- VAM HA with Bedmaster In Load Balancer Environment

- Initially, the Bedmaster makes a connection with VAM Mirth server 1 via load balancer and sends HL7 messages to it.
- **Note**: The connection type is LLP Listener and is persistent. Once established, it is always connected unless it is disconnected by the source or destination.
- VAM Mirth server 1 contains the channels which receive the message, transform it, create the JSON file, and send it to the LB. The LB then sends it to any of the VAM servers.
- VAM server receives the JSON and writes into a file in the HL7 directory.
- When VAM server 1 goes down, the HL7 data from Bedmaster is diverted to VAM server 2.
- VAM Mirth server 2 contains the channels which receive the message, transform it, create the JSON file, and send it to the LB. The LB then sends it to any of the VAM servers.

The following sections provide more information on the VAM high availability environment, including how to configure the BMHL7Receiver, BMJSONReceiver, and Monitor channels. For information on installing Mirth Connect on a VAM server, see **Mirth Connect Installation and Configuration** on page 29.

Prerequisites

To use VAM in a high availability environment, the prerequisites described here must be in place.

All instances of VAM and the load balancer must have been installed with basic configuration.

The following ports are opened in the load balancer and in all VAM systems.

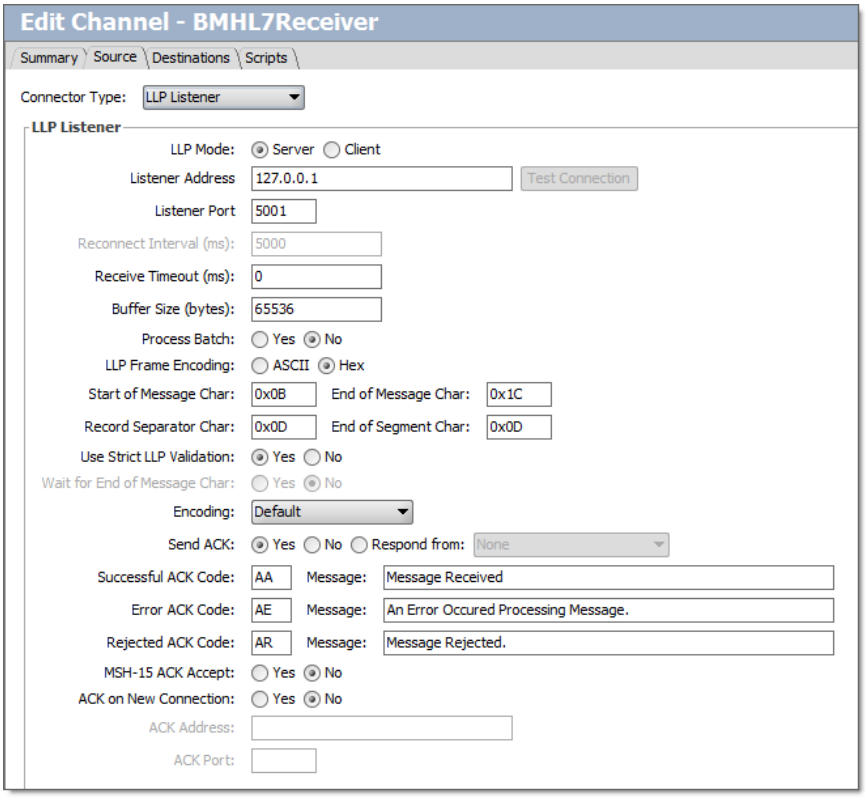
Port	Purpose
80	HTTP
443	HTTPS (SSL)
1433	SQL Server (Default)
3306	MySQL (Default) - if MySQL is used for Analytics
587 or 25	Smtp.gmail.com (optional)

VAM Services	Status
Vocera Alarm Message Generator	Running
Vocera Alarm Escalator	Running
Vocera Alarm Monitoring	Running
Vocera Alarm Clean Up Service	Running
Vocera Alarm Server Health Monitor	Running
Vocera Archive Service	Running
Vocera VS Interface	Running

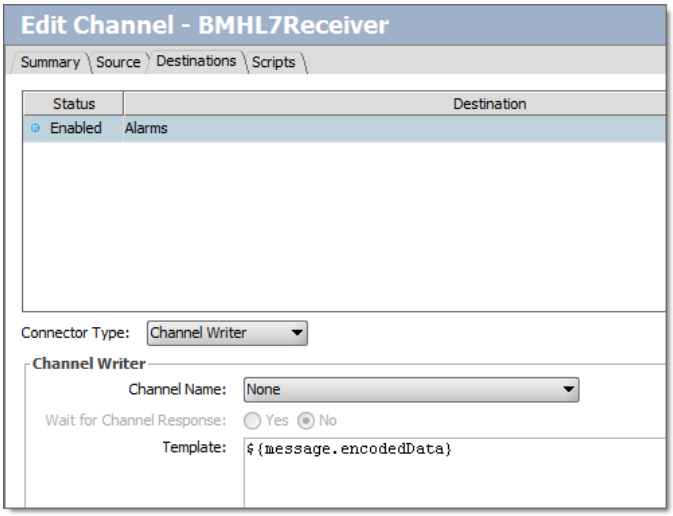
The BMHL7Receiver Channel

For the BMHL7Receiver channel, you must make these modifications.

Update the Source tab as shown:



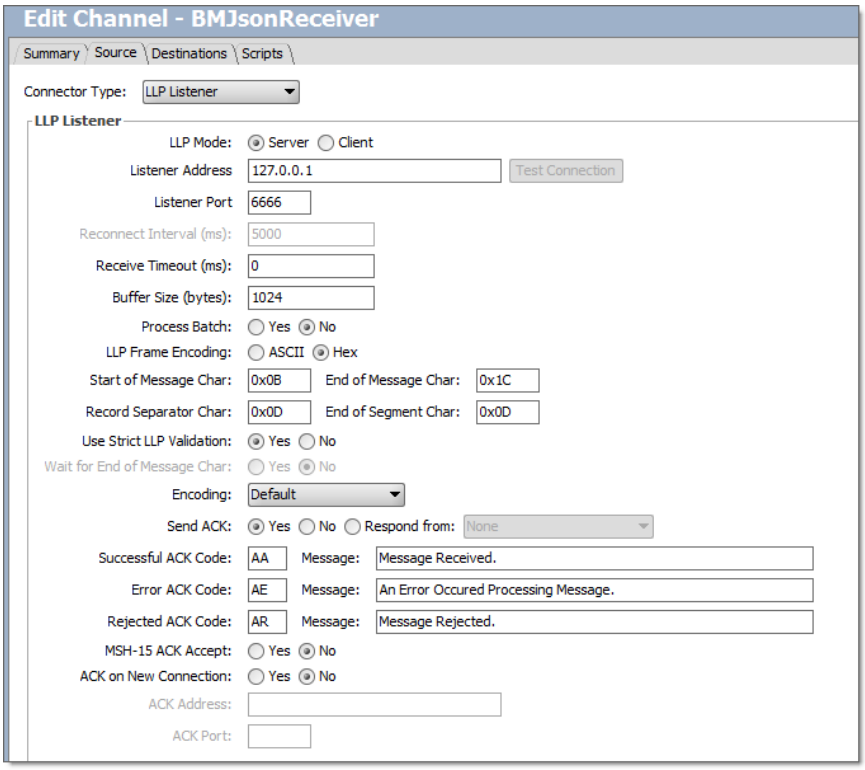
Update the Destination tab as shown:



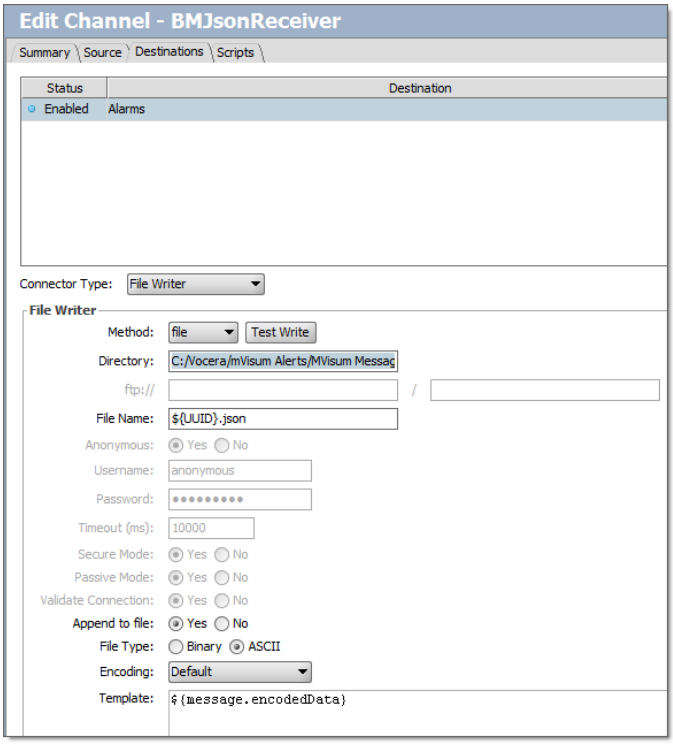
The BMJSONReceiver Channel

For the BMJSONReceiver channel, you must make these modifications.

Update the Source tab as shown:



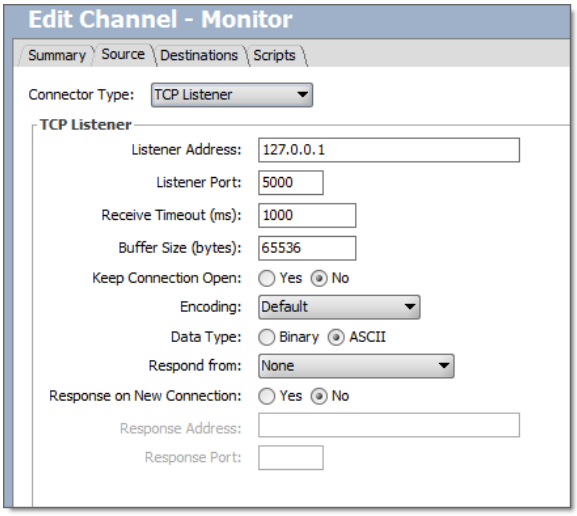
Update the Destination tab as shown:



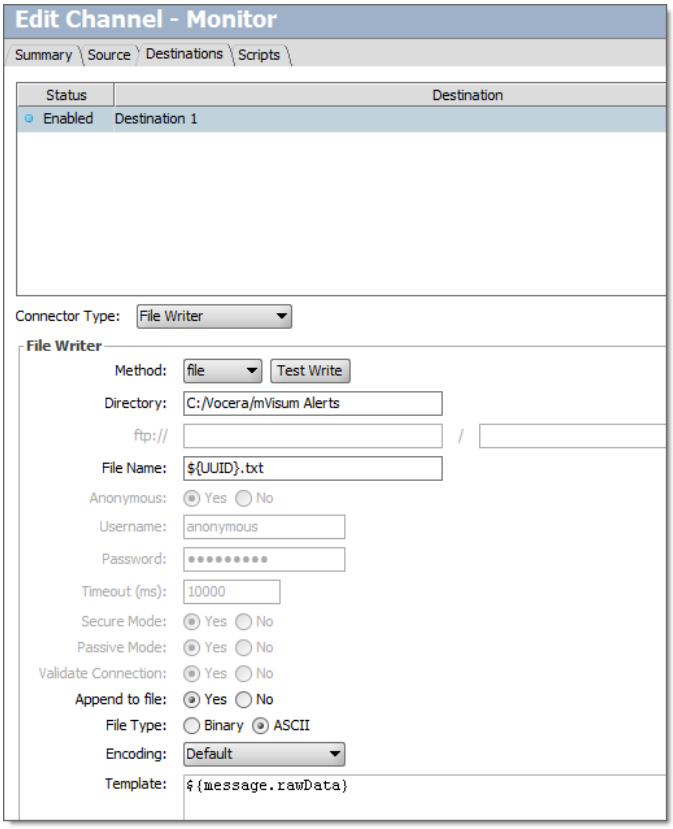
The Monitor Channel

For the Monitor channel, you must make these modifications.

Update the Source tab as shown:



Update the Destination tab as shown:



Applying the Batch File in a High Availability Environment

In a load balancing environment, Vocera recommends that you stop all VAM services except for the Vocera Alarm Monitoring and Vocera Alarm Server Health Monitor services if any service goes down. This enables the other active server to take on the entire load.

The Vocera Alarm Monitoring and Vocera Alarm Server Health Monitor services are kept running to notify you of the issue that caused the failover to occur.

For each active VAM service, perform the following steps:

1. Right-click on the service, and select Properties.
2. Click the Recovery tab.
3. In the First failure dropdown list, select Restart the Service.
4. In the Second failure dropdown list, select Restart the Service.
5. In the Subsequent failres dropdown list, select Run a Program.
6. In the Program field, type or browse for the location of the HA batch file that was included in your installation package.
7. Click OK to save your changes.

You will need to perform the above steps on each of the services below:

- Vocera Alarm Clean Up Service
- Vocera Alarm Escalator
- Vocera Alarm Monitoring
- Vocera Alarm Server Health Monitor
- Vocera Archive Service
- Vocera VS Interface

This ensures that all services are stopped if any service fails.

Here is a copy of the HA batch file that stops all VAM services.

```
@echo off
net stop w3svc
net stop "Mvisum Alert Clean Up Service"
net stop "MVisum Alert Escalator"
net stop "MVisum Alert Message Generator"
net stop "MVisum Archiver Server"
net stop "MVisum VS Interface"

if ERRORLEVEL 0 goto Print
echo Unable to stop services.
exit
:Print
echo Service stopped successfully.
Pause
```

Adding VAM Bulk Users

Learn how to add bulk users from Active Directory groups and CSV files, and how to assign hospital groups, units, and roles to these users in VAM.

Prerequisites

The following prerequisites must be completed before you start the process.

- Ensure that the VAM application is installed and running.
- Use the VAM Console to add all necessary units, such as cardiac units and other units and beds.
- Define the necessary groups on the Active Directory (AD) server, and add the appropriate users to those AD groups.

Important: Vocera recommends that you back up the MVisumAlerts database on your SQL server before starting this process, as database operations are performed here. The execution of an incorrect query may cause data corruption.

Note: VAM hospital groups are different from Active Directory groups.

Steps for Bulk User Creation and Configuration

The following steps describe how to create and configure bulk users.

Importing Users from AD Groups

You can use the VAM Console to import users from one or more Active Directory groups.

1. In the VAM Console, click Configuration.
2. From the left panel, select Active Directory.
3. Ensure that the Active Directory server is configured to add users.
4. Select the Import Users option. Provide a user name and password to log into the Active Directory server.
5. From the Field dropdown list, select Group. In the Value field, type the group name.
6. Click Search to list all the users in the group.
7. Select all users by clicking the check box adjacent to the First Name column.
8. Click Next and accept the default values. There is no need to change the unit or group while importing these users from AD groups. Use the default role and unit.

   The Role and Unit will be changed in the database using an SQL query. For details, see **Updating Hospital Groups, Units, and Roles** on page 47.

To add users from other Active Directory groups, repeat the above process for each group.

Note: Each user is assigned the Nurse Role by default.

Database Verification Before Updating

Use these steps to verify whether all users are added to the VAM server.

1. Open SQL Server Management Studio, log into the database server, and navigate to the MVisumAlerts database.
2. Check whether all hospital groups exist by running the following query:
   ```
   select * from HospitalGroups order by ID asc
   ```
3. Check whether all units exist by running the following query:
   ```
   select * from HospitalUnits order by ID asc
   ```
4. Verify the beds:
   ```
   select * from HospitalBeds order by ID asc
   ```
5. Verify that the users are imported as Nurse by default:
   ```
   select * from Nurse order by UserId asc
   ```
6. Make sure all nurses are imported correctly.
7. Ensure that there is no bed assignment on the Staff Assignment page.

Note: You must understand the table structure and note the UserId ranges for each AD group.

Updating Hospital Groups, Units, and Roles

Follow these steps to update hospital groups, units, and roles using database queries based on each AD user group.

1. In Active Directory, find the UserID ranges for the selected AD group. For example, users having UserId between 5 to 50 may be targeted for the Cardiac unit (find UserId from Nurse Table) to group TestGroup and unit TestUnit.
2. Get the relevant hospital group ID for the Hospital group (such as TestGroup). The ID is available in the HospitalGroups table.
3. Similarly, get the relevant unit ID for the unit (such as CardiacUnit). The ID is available from the HospitalUnits table.
4. Run a database query to change the hospital group and unit. This changes the UserID range, Hospital group ID, and unit ID. For example, if the group ID value is 1, the unit ID value is 3, and the targeted user ids are between 5 and 50, run the following query:
   ```
   update Nurse set GroupId=1, UnitID=3 where UserID>=5 and UserID<=50
   ```
   This changes the hospital group and unit for the respective users. The SQL Server Management Studio output window will list the number of rows affected. Verify that this result is what you are expecting.
5. Execute the following query to change the role for the users. Skip this step if the current role is correct. Get the relevant Role ID from the Roles table.
   For example, if the RoleID value is 4 for charge nurses, run the following query:
   ```
   update UserRoles set RoleId=4 where UserID>=5 and UserID<=50
   ```
   This query changes the role for the selected users. The SQL Server Management Studio output window will list the number of rows affected. Verify that this result is what you are expecting.
6. Repeat the above steps for each AD group to change the hospital group, unit, and role for the group's users.

Note: The execution of any incorrect query will lead to database corruption, which will stop VAM.

Verification After Database Update

Follow these steps to verify whether all nurses are updated, assigned, and configured properly.

1. Log in to the VAM Console.
2. Click Configuration and click Nurses to display the Nurses screen.
3. Select any nurse for which the role, hospital group, and unit have been updated.
4. Click Edit and verify whether the Hospital Group, Role (Title) and Unit are properly updated.

Importing Active Directory Users from a CSV File

Follow these steps to import Active Directory users from a comma-separated file.

1. Create a comma-separated file with this template:

```
First Name, Last Name, Email, Telephone Number, User Name, Group Name,
Unit Name
```

Note: Enter each user's information on a new line. Do not add a comma at the end of the line.

2. Create the GeBulkUsersInTable and ImportActiveDirectoryUsers procedures if they are not present. Refer to **Active Directory Import Procedures** on page 73 for more information.
3. Execute the following query on the MVisumAlerts database.

```
EXEC [dbo].[ImportActiveDirectoryUsers] '[User Info Script]','[Server
IP/Name]','[Domain Name]'
```

Note: User Info Script is the data from the comma-separated file. Server IP/Name and Domain Name are the IP address and domain name of the AD server. To view these in the VAM Console, click Configuration and select Active Directory.

Here are some sample queries:

```
EXEC [dbo].[ImportActiveDirectoryUsers]
'Gansh,Shipurkar,ghirpukar@vocera.com,+9180454736,gshirrkar,General,Test Unit
Santsh,Jadhv,sjadav@vocera.com,+918054714,sjaav,General,Test
Unit','blr-dev-qa01','Vocera.local'
```

```
EXEC  [dbo].[ImportActiveDirectoryUsers]
'Aaron, Hank,ahank@test.com,+333123456789,ahank,General,Test Unit
 Borlaug, Norman,bnorman@test.com,+333123456987,bnorman,General,Test
Unit','10.1.1.12','test.local'
```

4. Check the Results tab after execution to verify that all users are imported.

VAM Shut Down and Disaster Recovery

Learn how to shut down the VAM application server and how to perform a backup and restore to handle disaster recovery.

Disaster Recovery Prerequisites

These are the hardware and software prerequisites for disaster recovery.

VAM should be installed and running. The necessary systems and software must be installed and ready for restoring and verifying VAM.

The hardware shown here must be installed:

- Intel Xeon Quad-Core or equivalent
- 16 GB RAM
- 1 TB HDD

To view the software prerequisites, see **Vocera Alarm Management Requirements** on page 11.

Steps for Shutting Down

Use these steps to shut down VAM.

Note: You must shut down or log out from all other external devices that are connected to the VAM application server before shutting it down. Otherwise, the current operations in the devices will be terminated abruptly and users will be logged out. The effect of a shutdown depends on the current running operation.

Shut down the following before shutting down the VAM application server:

- All iOS and Android clients (devices).
- All browsers where the configuration portal is opened and in use.
- All external hardware integrations, such as patient monitor and BedMaster applications.

Stop the Windows services listed below in the following order:

- Vocera Alarm Message Generator
- Vocera Alarm Escalator
- Vocera Alarm Media Parser
- Vocera Alarm Monitoring
- Vocera Alarm Clean Up Service
- Vocera Alarm Notification Service
- Vocera Alarm Server Health Monitor
- Vocera Archiver Service

Stop the following Analytics services:

- Vocera Alarm Analytics Engine

- MySQL

Stop the Mirth Connect Service.

Stop the following servers:

- Stop IIS services, especially the ASP.NET State Service and the World Wide Web Publishing Service.
- Stop SQL server services, such as SQL Server and SQL Server Agent.

To restart VAM, restart the services in the reverse of the order in which they were stopped. Database servers and their services should be started first, then IIS services, and then Windows services.

An easier shutdown procedure is also available:

1. The easy way to shut down the VAM application is to shut down the application and database servers. Power off the systems unless other services are being used.
2. The easy way to restart the VAM application is to restart the application and database servers.

Notes:

- If the SQL database server is restarted, all of the Windows services listed here need to be restarted. There may be some issues, depending on the current running environment: for example, a user may be logged out or an error, exception, or crash may have occurred in the iOS app, Android app or VAM Console. In general, there will not be any major issues, but this is not guaranteed, as this depends on the current operation.
- There is no impact on failover if the above steps are followed, especially shutting down the external devices and hardware before shutting down the application server. Otherwise, the individual alerts and requests will not be processed properly or will be missed from the application server.
- If the server has to be restarted, for example because of maintenance or a Windows update, it is advisable to restart at midnight or on the weekend when the load is lighter. Since load balancing is used, there will not be any downtime when the server is restarted if only one server is restarted at a time.

Steps for Backup

Use the following steps to perform a VAM backup.

SQL Server Database Backup

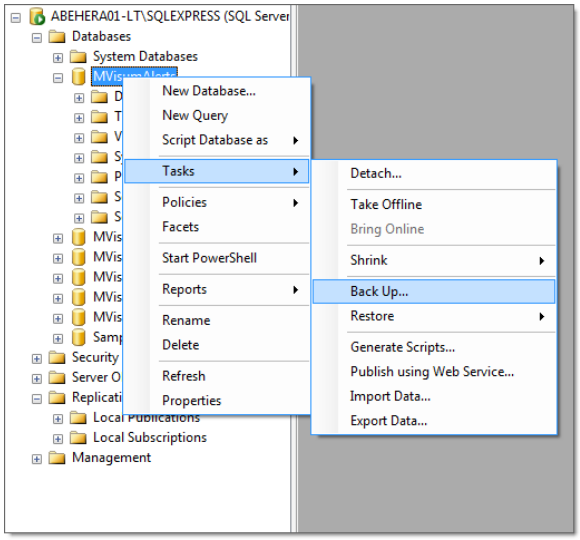
Here are the steps to follow to make a backup of VAM databases to enable recovery if the database server crashes.

The SQL server databases that need to be backed up are:

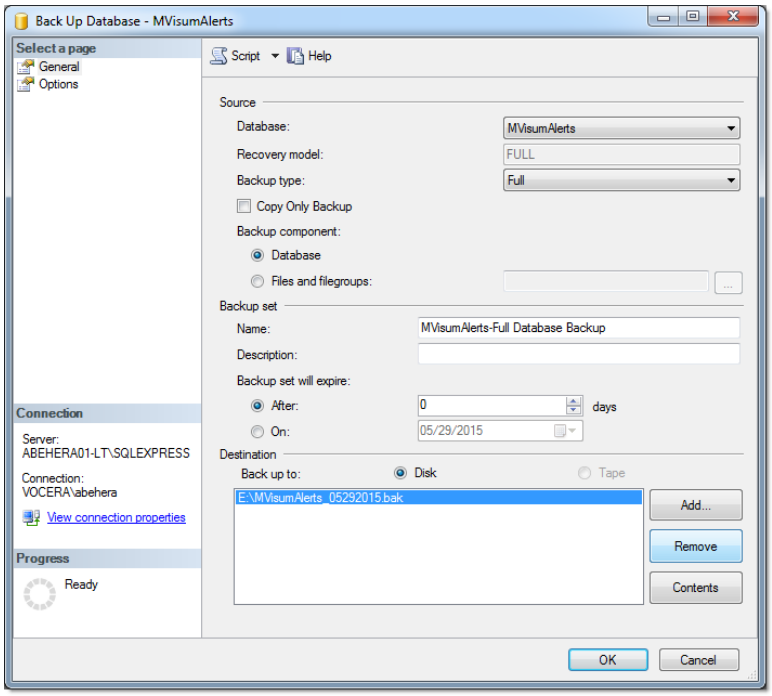
- MVisumAlerts
- MVisumInterface
- MVisumNotification
- MVisumPush
- BedMaster database: The name of the database to back up is BedMaster. Choose the simple recovery model.

Follow these steps to make a database backup. These steps can be repeated for each database.

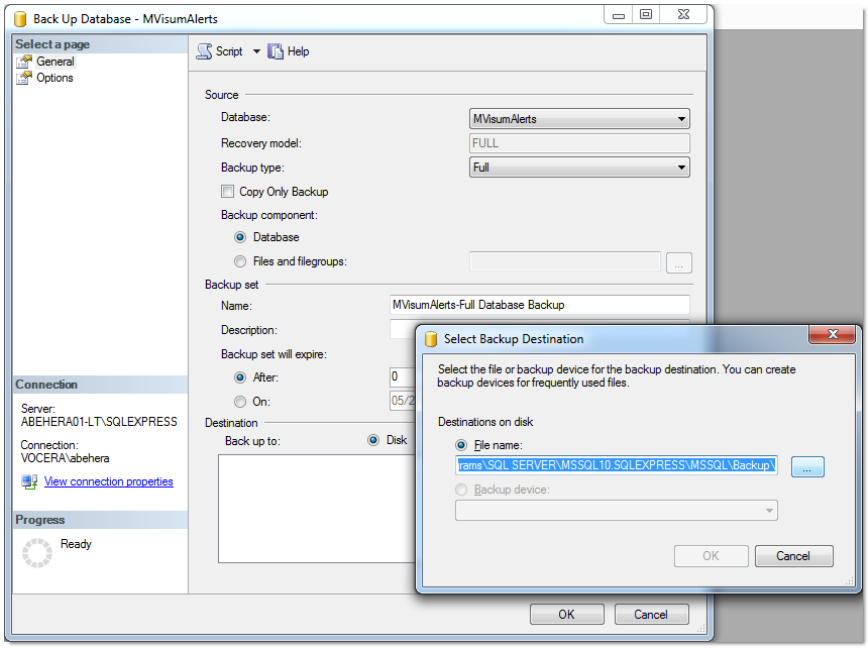
1. Launch the management console and login with valid credentials.
2. Right-click on the database.
3. Go to Tasks>Back Up.



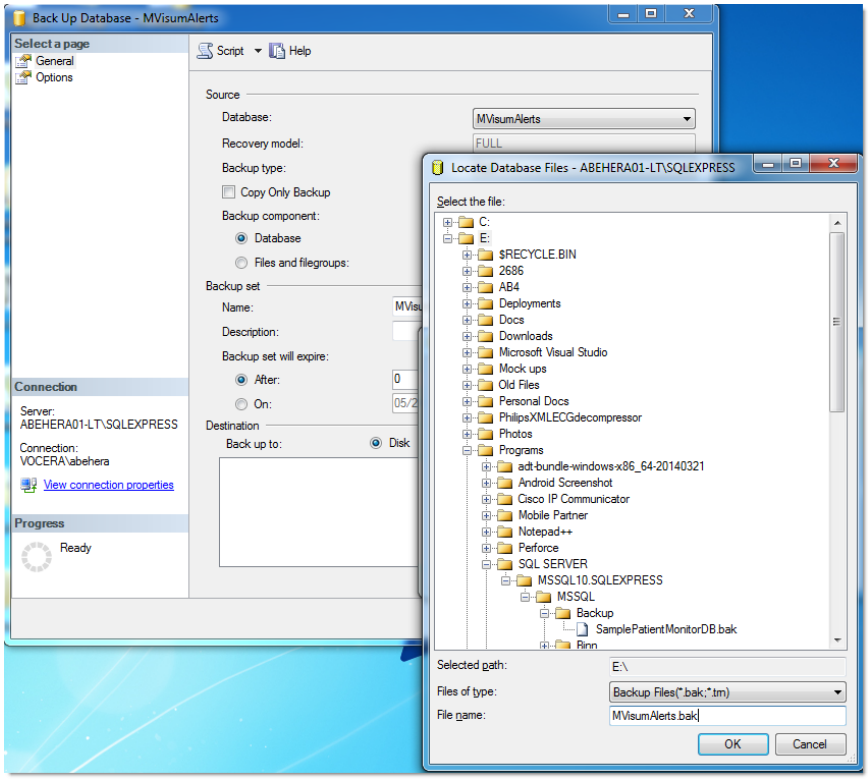
4. Remove the existing path using the Remove option and select the desired location.



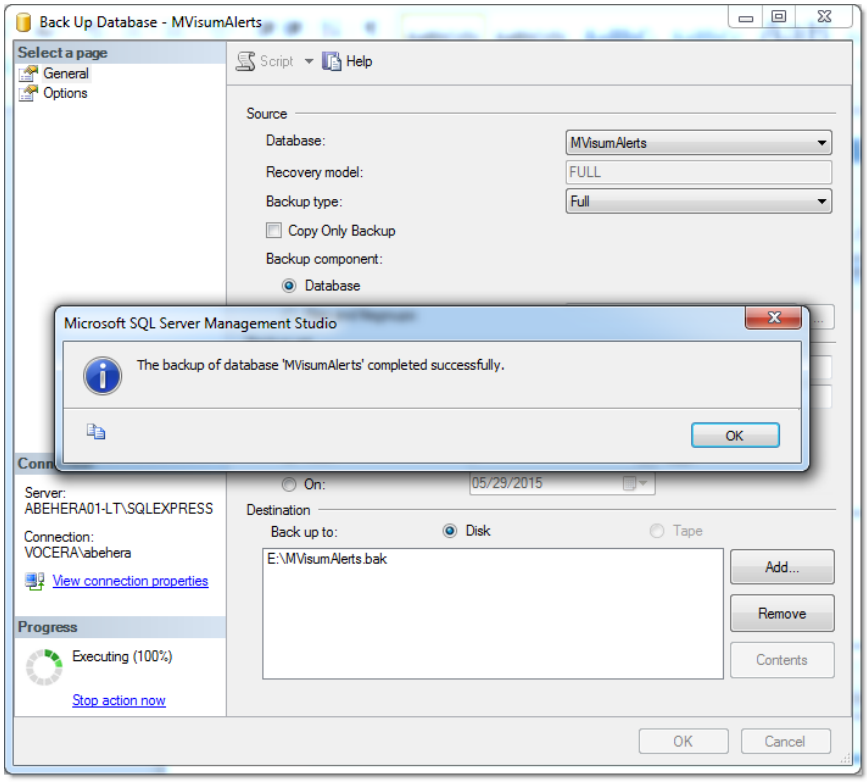
5. Click Add. The following screen will appear. Click ... Browse.



6. Browse to the desired directory and specify the name of the backup file as `[DatabaseName].bak`. For example, the database back up file for the mVisumAlerts database is named `mVisumAlerts.bak`.



7. Click OK to complete the backup process. The following message should appear:



If an error message appears, change the file path, as this might be a permission issue.

Component Backup

Vocera recommends that you have a backup of the configuration files for each component. The following are the configuration files that need to be backed up.

All folders listed here are contained in the **`mVisumAlerts`** subfolder of the VAM installation folder.

- MVisum Alerts Service\web.config
- Mvisum Clean Up Service\MVisumCleanUp.exe.config
- Mvisum Media Parser\MVisumMediaParser.exe.config
- MVisum Message Escalator\MVisumAlertEscalator.exe.config
- MVisum Message Generator\MVisumMessageGenerator.exe.config
- MVisum Monitoring\MVisumMonitoring.exe.config
- MVisum Notification Service\MVisumNotificationServer.exe.config
- MVisum Server Health Monitor\MVisumServerHealthMonitor.exe.config
- MVisumAlertPushServer\web.config
- MvisumAlertsAdmin\web.config
- MVisumArchiver\MVisumArchiver.exe.config

Alarm File Backup

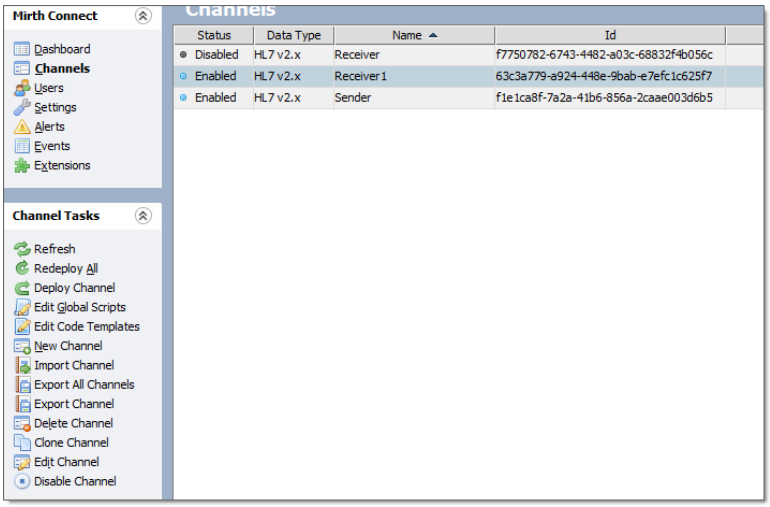
All Mirth servers should be configured to back up the alarm files.

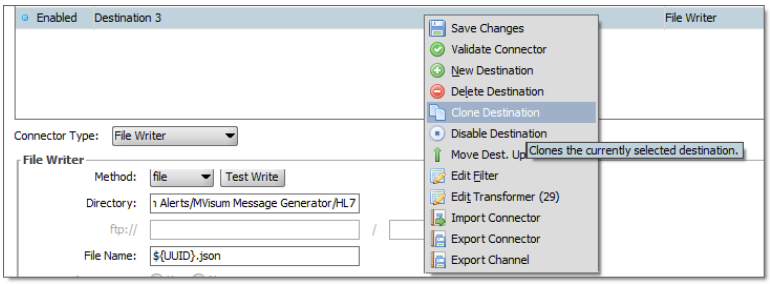
The following are the steps to follow to perform a backup:

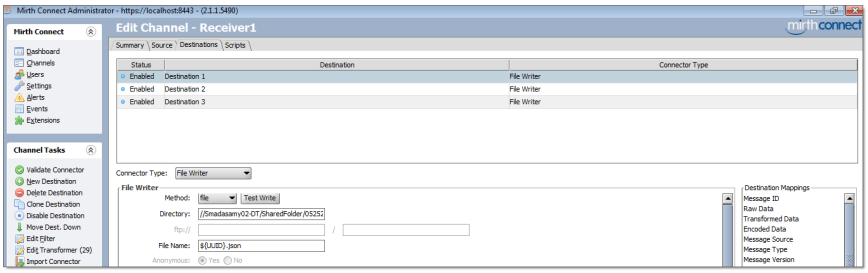
1. In the Mirth server, launch the administrator console and login with admin/admin UID/PWD.
2. Click Channels.
3. Select the Receiver channel and select the Edit Channel option.
4. In the Destination tab, all the destinations are listed. This writes the files to the defined location.

5. Right-click on each destination and select the Clone Destination option.
6. Modify the directory path of that destination to be the backup server destination. To implement this, create a shared directory on the backup server which is accessible from the Mirth server. The shared path can be provided in the cloned destination.
7. Save the changes using the Save Changes option.
8. Redeploy the channel.

See below for details.







MySQL Analytics Database Backup

Follow these steps to make a backup of the MySQL (analytics) databases.

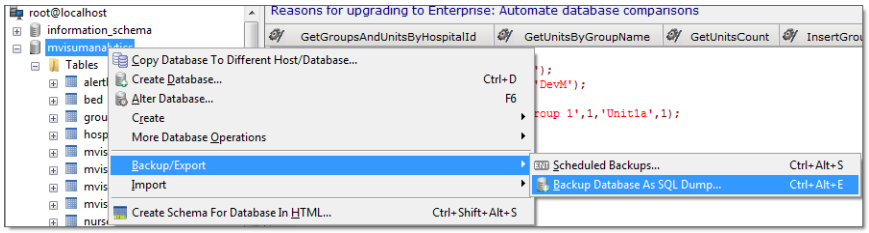
You will need to back up these databases:

- mVisumAnalytics
- mVisumPM

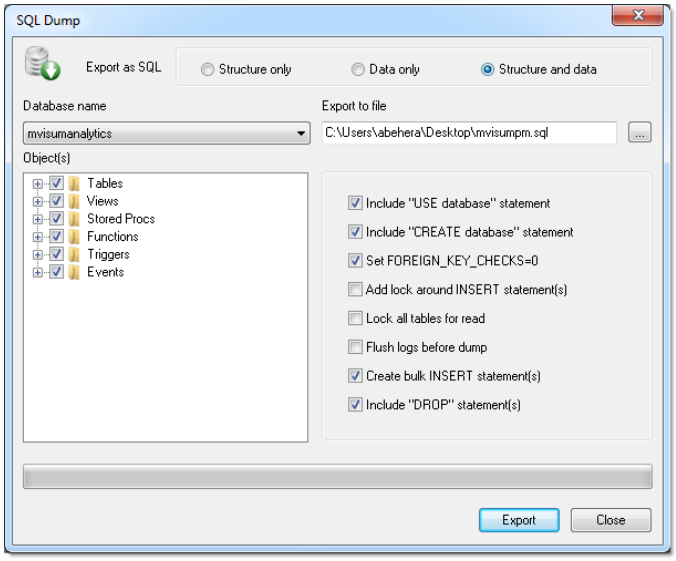
SQL Workbench or SQLyog must be installed to make a backup of the MySQL databases. Follow these steps to make a backup using SQLyog.

1. Open SQLyog.
2. Right-click on the desired database.

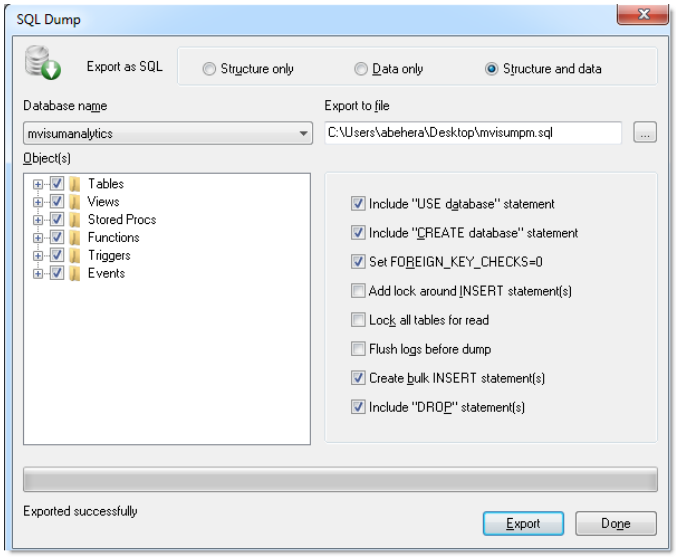
3. Go to Backup/Export > Backup Database As SQL Dump, as shown in the following diagram.



4. In the Export as SQL radio buttons, select the Structure and data option for Export as SQL.
5. Select Export to file and specify the file path to export the database to. Click Export.



6. After a successful export, an Exported Successfully message will be shown below. Click Done.



Automating the MySQL Database

Follow these steps to automate the MySQL database.

1. Create a command file (.bat or .cmd file) with the following contents:

```
"[MySQL installation directory]\bin\mysqldump" -u [UserName] -p[Password]
mvisumpm > [Desired Directory]\mvisumpm.sql
```

```
"[MySQL installation directory]\bin\mysqldump" -u [UserName] -p[Password]
 mvisumanalytics > [Desired Directory]\mvisumpm.sql
```

2. Name it `MySQL_Auto_Backup.cmd`.
3. In the Task scheduler, create a task to run this file, and set its frequency.

Steps for Restore / Verification

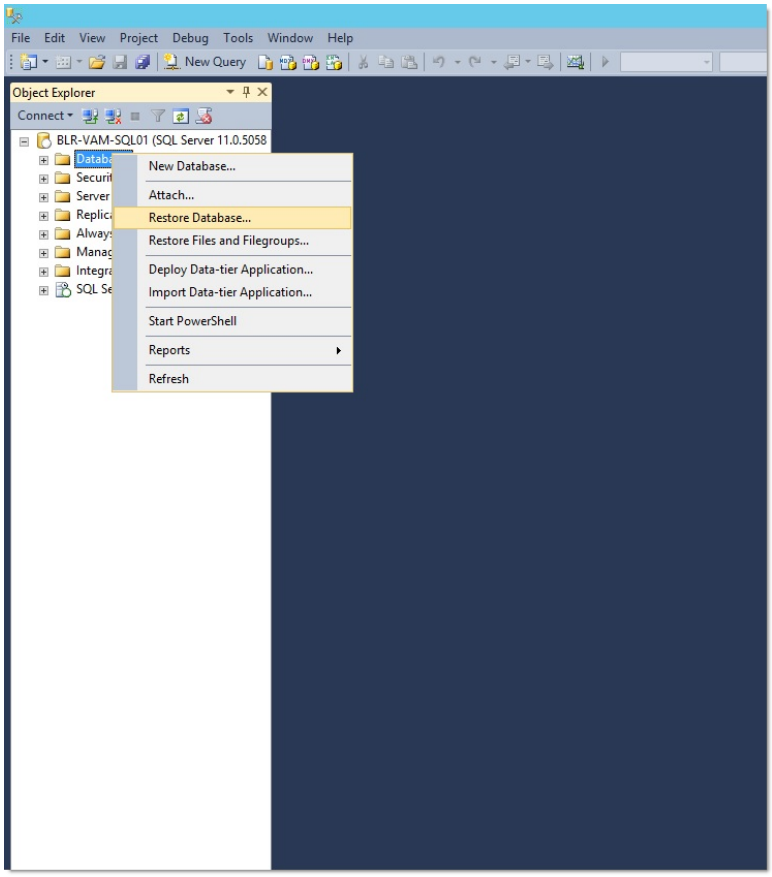
Follow these steps to perform a restore and verification.

Database Restoring and Verification

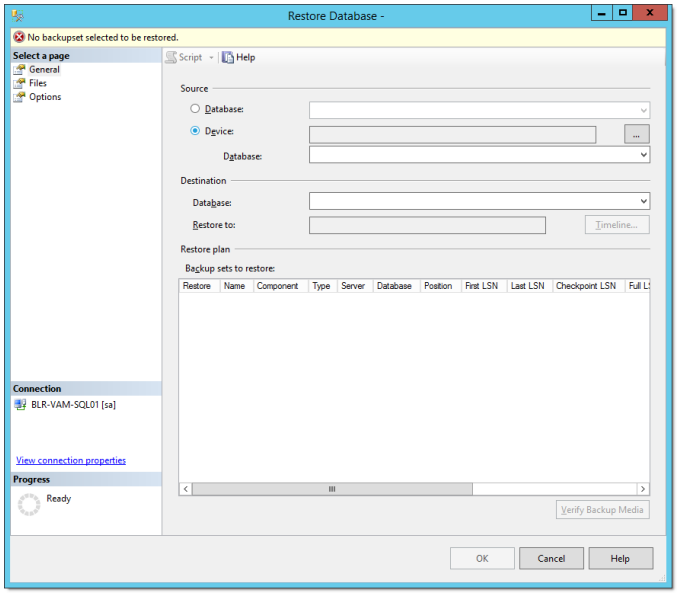
Use the following steps for SQL Server database recovery.

1. Use SQL Server Management Studio to create the databases. The following databases need to be created:
   - MVisumAlerts
   - MVisumInterface
   - MVisumNotification
   - MVisumPush
2. Stop all VAM Windows services and IIS on all machines. To stop IIS, run the following command from a Command Prompt window.

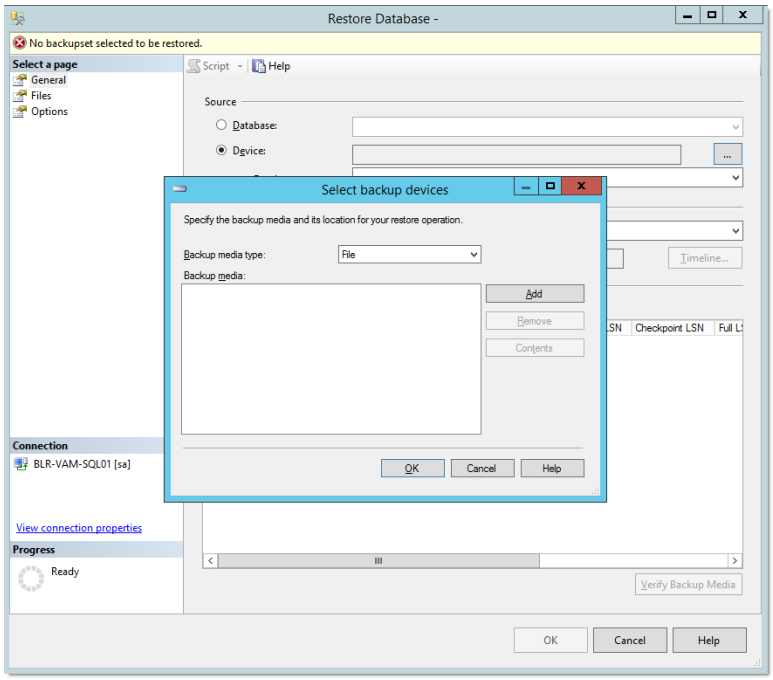
   ```
   iisreset /stop
   ```
3. Right-click on each database, and select Restore Database.



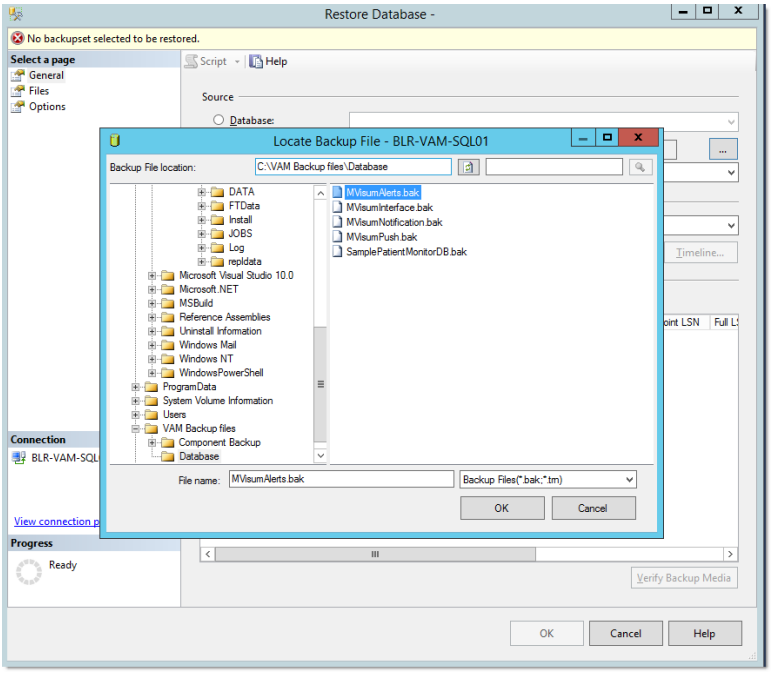
4. Select the Device option and click Browse to select the backup file.



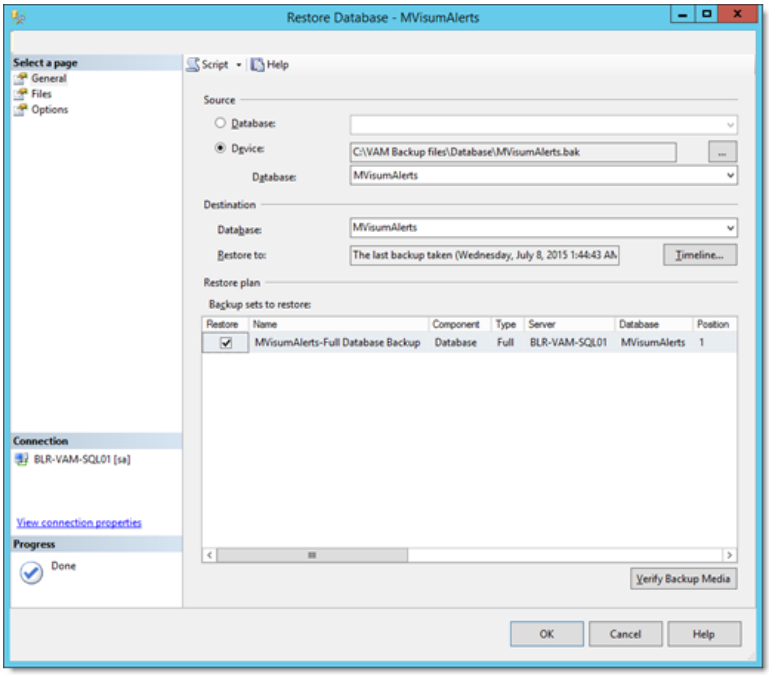
5. Click Add.



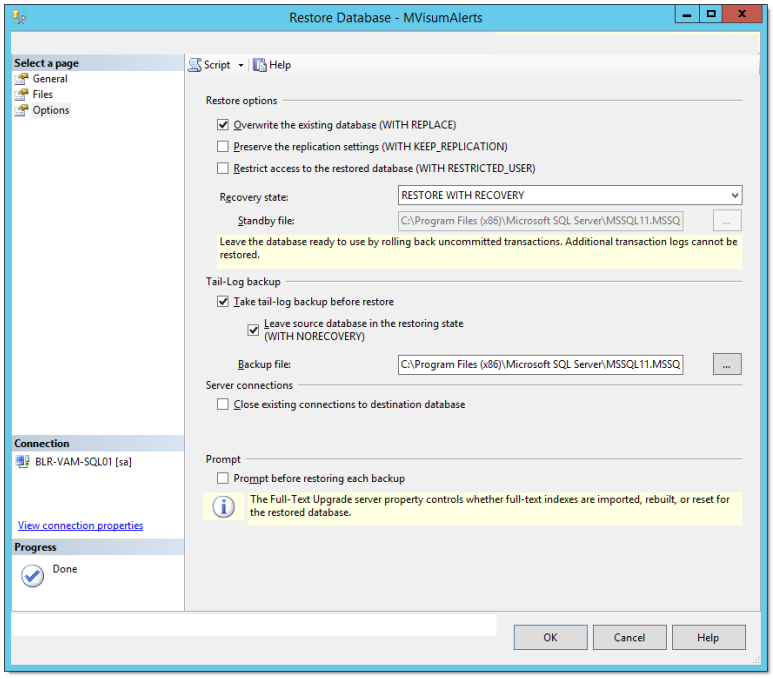
6. Browse to the `.bak` file of the database, which was taken from the original server.



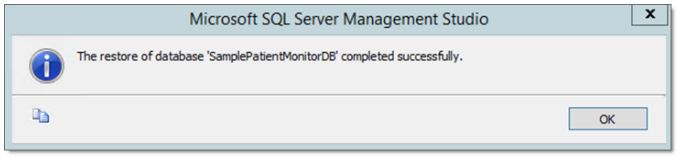
7. Click OK. Click OK.
8. Select the checkbox in the Restore column as shown below.



9. In the Restore options section, select Overwrite the existing database.



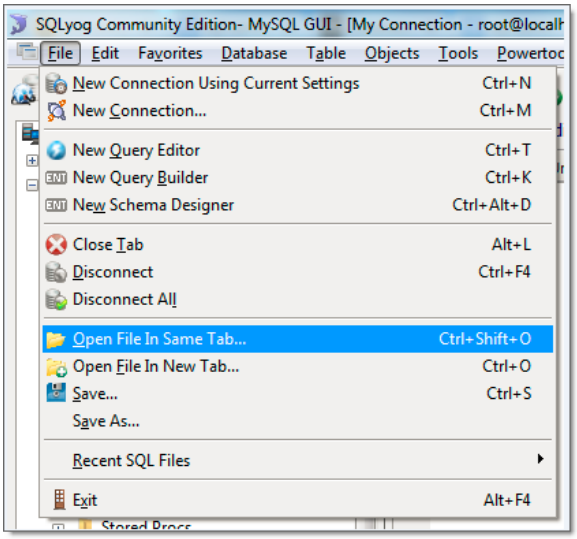
10. Click OK. The following message should appear:



Analytics Database Recovery

The analytics database can be recovered by browsing the backup file if SQLyog or SQL Workbench is up and running.

This will create the database with existing data.



VAM Component Restoring and Verification

Follow these steps to restore and verify the VAM components.

1. Make sure IIS is running. This can be verified by typing localhost in a browser. It should display the IIS screen with its version.
2. Check whether any existing component of the VAM server exists. If it is broken, Vocera recommends that you remove or uninstall it.
3. Run `VAMInst.exe` to install the VAM server.
4. Make sure that the No option is selected when asked whether to create the database.
5. After installation, verify that all the components are running.
6. From the backup, replace the configuration files in each component.
7. Restart all of the services.
8. Restart the IIS. To start IIS, run `iisreset /startcommand` from the Command Prompt window.
9. Access the VAM Console using the default `webuser/webuser` user name and password.

Mirth Server Restoring and Verification

Follow these steps to restore and verify the Mirth server.

1. Check whether there are any existing Mirth server components. If the server is broken, Vocera recommends that you remove or uninstall it.
2. Reinstall the Java component. This can be verified by running Java -v from the Command Prompt window.
3. Reinstall the Mirth server.
4. Import the channels, and ensure that the directory path is pointing to the `HL7` folder.

Alternate Restore Method for VAM and Mirth Server

Follow these steps to restore and verify VAM and the Mirth server.

1. Import the receiver channel to the Mirth server and deploy it.
2. Run the VAM installer on a new virtual machine. Ensure that the prerequisites are satisfied.
3. Point the new installation of VAM at the MS SQL VAM database server.
4. Choose not to recreate the database. Use the existing one.
5. After installation is complete, replace the configuration files with the configuration files from one of the slave machines.
6. Restart the Windows and IIS services.

Remarks

Take note of this additional information related to backup and restore.

- The steps described here for database backup and restore can be replaced with some other approach, such as replication or automatic backup with any third-party application.
- Using this approach, the backup server can be setup and tested by switching off the main server and moving on to the backup server.
- If two application servers are used under load balancing, and one server is down, the other server will take care of processing alarms. To verify this, remove one server from the network so that the other server will receive all of the requests.

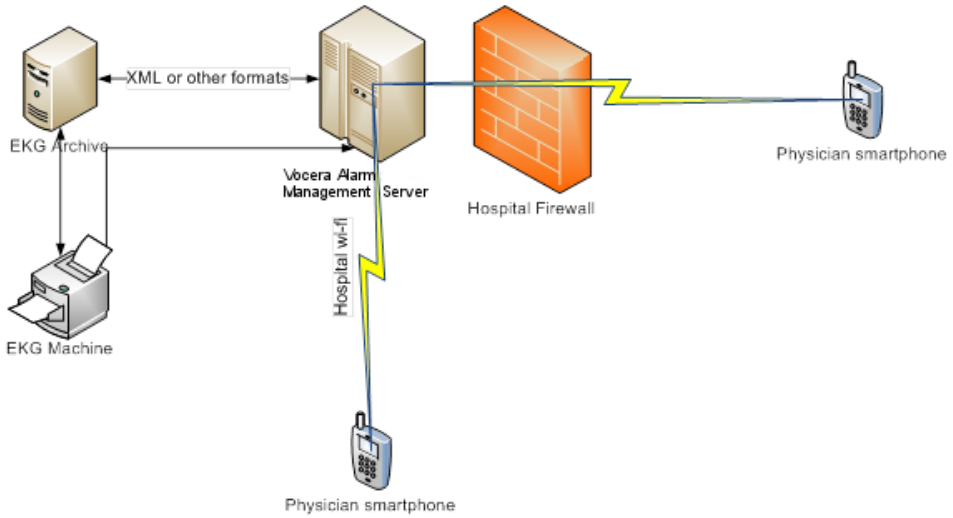
Technical Information

These sections provide technical background information on the features and capabilities of Vocera Alarm Management.

Application Summary

From a technical viewpoint, Vocera Alarm Management (VAM) is a platform that delivers EKGs and other related information to clinician smartphones.

The system primarily consists of software on a server and dedicated client software on handheld devices. The figure below shows the general system architecture.



In a modern hospital setting, multiple EKG machines/patient monitors archive EKGs into hospital-wide EKG archival systems. These systems are capable of exporting EKG data through XML or other proprietary formats. EKG machines also have the ability to send EKG data directly to a third party server (such as VAM's server). These systems communicate through the hospital's Ethernet network using industry standard TCP/IP protocols.

The VAM server receives EKG data files. Data in these files may include not just the EKG traces, but also analysis results from the EKG machines and/or the EKG Archival server and diagnosis and risk metrics. The VAM server, based on the data contained, can notify one or more clinicians about the availability of the data.

The VAM server also provides rules functionality for assigning specific beds/patients to specific devices, clinician shift management, and traceability and escalation to notifications and data that run through the system.

For some devices (such as those from Apple running iOS), the push process may be through a third party server such as the Apple Push Notification server and not directly from the VAM server.

System Security and Related Considerations

The VAM architecture innately supports strong security. Security is handled at multiple levels and from all possible perspectives. VAM operates through a dedicated client on the handheld, which uses HTTPS communication with the VAM server application.

Data Pipe Security

VAM operates through a secure data pipe that uses SSL.

Any industry standard SSL certificate can be deployed on the server that runs VAM. All communications/transactions between the handheld VAM application and the VAM server application are carried out exclusively using HTTPS through this SSL pipe (port 443). VAM can also support a self-signed certificate if the hospital chooses to deploy a self-signed certificate.

Authentication

VAM provides user and device authentication for communication between handheld devices and the server.

Each device is provided with a unique, randomly-generated device ID (issued and activated for each device by the system administrator). Each user is assigned a unique username and password.

Every call between the dedicated VAM application on the handheld device and the VAM server is authenticated.

The server administrator has the ability to disable the VAM application on any handheld by disabling/deactivating the device ID of any particular device.

The server administrator has the ability to force unique users in a current active session to log out.

Password Strength and Security

The user password strength (length, capital letters, numeric, special characters/symbols) is configurable on the system when Active Directory authentication is not used.

Passwords are passed to server as a hash of password hash and current timestamp.

Auto-logout

Maximum session idle time is configurable on the server, after which users are automatically logged out.

Data on Handheld

Security measures are provided for data on handheld devices.

- All data is wiped on the handheld device on logout (including auto-logouts).
- All temporary data stored on the physician's handheld device during a session is encrypted.

Server

Server security capabilities are provided.

- User passwords are not stored in the database. Only the password hash is stored in the database.
- Database passwords are stored in encrypted formats.

VAM Console

The interface to the VAM Console has been made secure.

- Access to the VAM Console is provided over an SSL connection. The hospital IT department may set up a firewall policy to ensure that the from the VAM Console URL is only accessible from within the hospital.
- Passwords are passed from the VAM Console to the server as a hash of password hash and current timestamp .
- All access to the from the VAM Console is role based, with multi-role support.

Server and Admin Console

The following topics provide technical details on the VAM server and the VAM Console.

Database Choices

VAM operates on the SQL server.

For small installations, VAM can operate using SQL Express. Typical enterprise deployments require an instance of SQL Standard Edition.

SQL Server Authentication

VAM supports SQL authentication for SQL connections between the VAM application and the database. Windows authentication is not supported.

Virtualization

VAM can operate on any standard virtual server.

Server Configuration

VAM supports deployment in multiple configurations depending upon the need and size of deployment.

The following configurations are supported:

- Single server deployment with database contained on same server (for smaller deployments).
- Dual server deployments, with the application server separate from the database server (typically with the application server hosted in the DMZ)
- Larger, high up-time deployments can include dual production IIS servers connected to a separate SQL cluster and a load balancer deployed in front of the dual IIS servers for hot-failover operation.

Network Connectivity

VAM requires the server to be connected to the hospital intranet through the hospital's LAN (10M or higher speeds), and supports dual network deployments with physical separation of external IP and internal IP.

External IP/URL Name (SSL Requirement)

For mobile device access from outside the hospital, VAM's application server requires a static external IP address.

VAM's handheld software can also support a named URL (typically required when deploying a third-party SSL certificate).

If IP-only based access is set up, and access is required both within the enterprise (through enterprise Wi-Fi) as well as outside the enterprise, routing should be set up so that queries for the external IP from within the organization are appropriately routed to the active IIS port.

Firewall Considerations

These sections describe ports and blocking access to the VAM Console.

Ports

All VAM communications are structured as HTTP/HTTPS communications and operate through port 80 (when no SSL is used) and port 443 (when SSL is used).

VAM requires use of SSL for all communications outside the enterprise network (other than for communications through a Blackberry Enterprise Server), so port 443 needs to be open for incoming communication to the VAM application server.

Firewall Block for Console Access

It is recommended that access to the administrative web interface be blocked from the external IP.

This can be achieved by blocking access to the following URL from the firewall:

```
https://[ipaddress/name]/voceraalarmadmin/login.aspx
```

Server Patches and Anti-Virus Updates

VAM is installed and qualified with current patch levels.

Check with Vocera prior to deploying server patches. Since ALL services on the VAM server application are architected to run on .Net, Java, and SQL, pay particular attention to updates to these functional units.

Typically, anti-virus updates do not impact VAM's operation. However, note that updates, including anti-virus or security rule updates, that redefine folder share permissions, change process permissions of existing processes, and/or impact functioning of the user rights and privileges of the process accounts used by VAM can impact VAM's operation.

Shared Folder Connections

VAM uses shared folder connections to receive data through XML interfaces. In such cases, the shared folder should be accessible by the process account used by VAM at the site.

Note: Any changes to the Active Directory setup that can impact access by the VAM process user can cause a VAM outage if the shared folder connection is inaccessible.

History Query Path and Web Services

VAM uses a web service on the EKG Archival system for querying for and retrieving historic EKGs for each patient message that is processed through the VAM communication system.

Typically, the web service includes its own authentication on the EKG archive. Ensure that the VAM process user has sufficient rights on the network to allow processing of this web service.

SMTP For Email Notifications

VAM's server includes system health monitors that can report current or potential problems, as well as daily use statistics, through email.

For this feature to function, the VAM process account needs access to a SMTP point and the access authentication details.

Console Support

The VAM console operates as a web-based (thin client) console. It is supported on all versions of Internet Explorer (7 and later) and current versions of Chrome and Firefox browsers.

The console is role restricted and exposes only the functionality relevant to the particular role of the user currently logging on.

The console uses JavaScript and, accordingly, the browsers on the hospital's computers need to be configured to allow JavaScript support.

For the console to be accessible, the URL (typically, `https://[internal IP of IIS server]/voceraalarmadmin/login.aspx`) needs to be accessible from anywhere the console needs to be accessed.

It may choose to deploy shortcuts to this web address on hospital PCs.

SSL / Secure Data Pipe

These sections provide information on certificates that can be used with the VAM server.

Commercial Certificate

VAM can operate with any standard SSL certificate issued by any known certificate issuing authority (such as Verisign).

Typically, for commercial SSL certificates to be deployed, the server needs to have a named URL in addition to a static external IP.

Self-Signed Certificate

VAM can operate through a self-signed certificate.

VAM's applications can use 128-bit or higher self-signed certificates deployed on the IIS. If self-signed certificates are used, certificates may also need to be separately installed on the handheld devices.

Firewall Configuration

The firewall needs to be configured so that HTTP requests that come in on port 443 at the static external IP are routed to the VAM IIS server.

This can be done either by placing the VAM IIS server in the DMZ or by forwarding HTTPS requests to the internally located VAM server.

Handheld

The following topics provide technical details on the VAM app that is installed on handheld devices.

Supported Handhelds

VAM supports the following handheld operating systems.

- Android (4.0 or higher)
- iOS (7.0 or higher)

Deployment

The following topics may be useful to you when you are deploying the VAM handheld client on a device.

Android

The following deployment methods are available for Android.

Direct

VAM Android client software can be deployed on the VAM server so that a link can be emailed to each user. Using this link, the user can download and install the VAM client from the URL.

The Android device should be set up to accept applications from Unknown Sources for this to be supported.

Through MDM

Any popular MDM solution can be used to push the VAM client to the Android device. VAM can provide the necessary application file as part of the deployment.

iOS

The following deployment methods are available for iOS.

iTunes Store

VAM is available through the iTunes store. The application can be downloaded from the iTunes store and can be made fully functional by supplying the proper external IP of the VAM server at your location.

Corporate iTunes Store

If a corporate iTunes store is deployed for the hospital, VAM can make available the necessary .ipa files for deploying through the corporate iTunes store.

Device Registration

On first login, the handheld will require the user to type a unique registration number provided by the system administrator. The number of available registration numbers is based on the number of handheld licenses deployed on the server.

The user will be asked to login again once the device is successfully registered to the server.

Wi-Fi Coverage

VAM operates through the Wi-Fi network. Ensure that the external IP has a name associated with it, or mobile http requests over Wi-Fi for the VAM server IP are routed to the appropriate IP address and port internally.

Appendixes

The following appendixes contain information that may be useful to you.

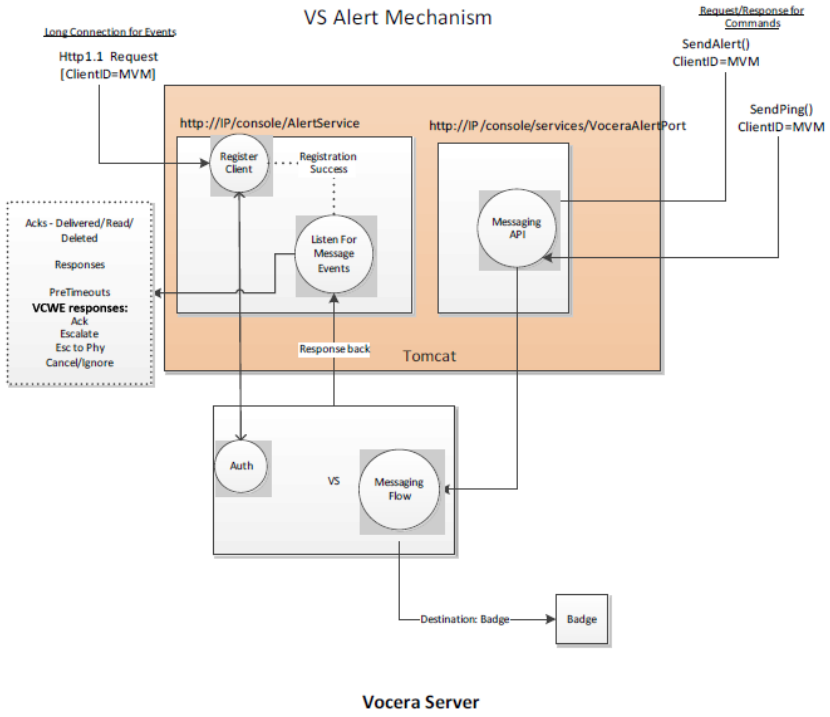
Directory Structure After Installation

This is the list of subfolders that the Vocera Alarm Management installer creates.

```
Root directory-> C:\Vocera\mVisum Alerts
      \AutoSyncEXE
    \Logs
    \MobileDownloads
    \MVisum Alerts Service
    \Mvisum Clean Up Service
    \MVisum Media Parser
    \MVisum Message Escalator
    \MVisum Message Generator
    \MVisum Monitoring
    \MVisum Notification Service
    \MVisum Server Health Monitor
    \MVisumAlertPushServer
    \MvisumAlertsAdmin
    \MVisumArchiver
    \MVisumTestApp
    \Support
    \Temp
```

Vocera Voice Server Alarm Mechanism

This is a diagram of the Vocera Voice Server alarm mechanism.



Changing the Notification Proxy Port

The Vocera Alarm Notification Proxy Server service is not required for version 2.2.1.2 and later of the iOS app. The following changes can be used for older iOS clients, and are provided for information purposes only.

Follow these steps to change the port in the notification proxy:

1. Changes in the VAM Console:
   a. Log into the VAM Console.
   b. Go to the Account Settings > General tab.
   c. Navigate to the VOIP Settings section.
   d. Change the port number in the Proxy Server Port field (for example, 12000).
   e. Click Save.
2. Vocera Alarm Notification Proxy Server service changes:
   a. From `services.msc`, stop the Vocera Alarm Notification Proxy Server service.
   b. Navigate to the `<Vocera>\mVisum Alerts\MVisum Alert Notification Proxy` directory.
   c. Open the `mVisumAlertNotificationProxy.exe.config` file.
   d. Change the value of the ServerPort key to match the port number that you specified above. This key is located in the AppSettings section of the configuration file. For example, if your server port is 12000, the ServerPort key would be defined as shown below:

      ```
      <add key="ServerPort" value="12000"/>
      ```

   e. Save the configuration file.
   f. Restart the Vocera Alarm Notification Proxy Server.
3. Log in again from all connected devices.

List of Alarm Transformers

This is a list of the Alarm transformers for the Mirth server.

Name of transformer	Type	Mapping
SendingApplication	Mapper	msg['MSH']['MSH.3']['MSH.3.1'].toString()
SendingFacility	Mapper	msg['MSH']['MSH.4']['MSH.4.1'].toString()
DateTimeOfMessage	JavaScript	see JavaScript code 1
MessageControlID	Mapper	msg['MSH']['MSH.10']['MSH.10.1'].toString()
ProcessingID	Mapper	msg['MSH']['MSH.11']['MSH.11.1'].toString()
PatientInternalId	Mapper	msg['PID']['PID.3']['PID.3.1'].toString()
PatientGivenName	Mapper	msg['PID']['PID.5']['PID.5.2'].toString()
FamilyName	Mapper	msg['PID']['PID.5']['PID.5.1'].toString()
DateOfBirth	Mapper	msg['PID']['PID.7']['PID.7.1'].toString()
PatientClass	Mapper	msg['PV1']['PV1.2']['PV1.2.1'].toString()
PointOfCare	Mapper	msg['PV1']['PV1.3']['PV1.3.1'].toString()
Bed	Mapper	msg['PV1']['PV1.3']['PV1.3.2'].toString()
AdmissionType	Mapper	msg['PV1']['PV1.4']['PV1.4.1'].toString()
UniversalServiceIdentifier	Mapper	msg['OBR']['OBR.4']['OBR.4.1'].toString()
ObservationDateTime	JavaScript	see JavaScript code 2
ObservationEndDateTime	JavaScript	see JavaScript code 3
RelevantClinicalInformation	Mapper	msg['OBR']['OBR.13']['OBR.13.1'].toString()
DiagnosticServiceSectionID	Mapper	msg['OBR']['OBR.24']['OBR.24.1'].toString()
ParentResult	Mapper	msg['OBR']['OBR.26']['OBR.26.1'].toString()
QuantityTiming	Mapper	msg['OBR']['OBR.27']['OBR.27.1'].toString()
AlarmLevel	Mapper	msg['OBX'][0]['OBX.3']['OBX.3.1'].toString()
AlarmReason	Mapper	msg['OBX'][0]['OBX.5']['OBX.5.1'].toString()
ObservResultStatus	Mapper	msg['OBX'][0]['OBX.11']['OBX.11.1'].toString()
AttachmentType	Mapper	msg['OBX'][1]['OBX.3']['OBX.3.1'].toString()
Waveform	Mapper	msg['OBX'][1]['OBX.5']['OBX.5.5'].toString()
AttachmentDate	Mapper	msg['OBX'][0]['OBX.14']['OBX.14.1'].toString()
ControlId	Mapper	msg['MSH']['MSH.10']['MSH.10.1'].toString()
MessageType	Mapper	msg['MSH']['MSH.9']['MSH.9.1'].toString()
VersionId	Mapper	msg['MSH']['MSH.12']['MSH.12.1'].toString()
Vitals	JavaScript	see JavaScript code 4

JavaScript code 1:

```
    var datestring = DateUtil.convertDate("yyyyMMddHHmmss","MM/dd/yyyy
HH:mm:ss",
    msg["MSH"]["MSH.7"]["MSH.7.1"].toString());
    channelMap.put("DateTimeOfMessage",datestring);
```

JavaScript code 2:

```
    var datestring = DateUtil.convertDate("yyyyMMddHHmmss","MM/dd/yyyy
HH:mm:ss",
    msg["OBR"]["OBR.7"]["OBR.7.1"].toString());
    channelMap.put("ObservationDateTime",datestring);
```

JavaScript code 3:

```
    var datestring = DateUtil.convertDate("yyyyMMddHHmmss","MM-dd-yyyy
 HH:mm:ss",
    msg["OBR"]["OBR.8"]["OBR.8.1"].toString());
    channelMap.put("ObservationEndDateTime",datestring);
```

JavaScript code 4:

```
    var len=msg['OBX'].length();
    var observation = '{  ';
    for(i=1;i<len;i++)
     {
     observation = observation + '"' + msg['OBX'][i]['OBX.3']
['OBX.3.1'].toString() + '":"' + msg['OBX'][i]['OBX.5']
['OBX.5.1'].toString() + '",';
      }
      observation = observation.substring(0,observation.length-1);
      observation = observation + '}';

      channelMap.put('Vitals',observation);
```

List of Vitals Transformers

This is a list of the Vitals transformers for the Mirth server.

Name of transformer	Type	Mapping
SendingApplication	Mapper	msg['MSH']['MSH.3']['MSH.3.1'].toString()
SendingFacility	Mapper	msg['MSH']['MSH.4']['MSH.4.1'].toString()
DateTimeOfMessage	JavaScript	see JavaScript code 1
MessageControlID	Mapper	msg['MSH']['MSH.10']['MSH.10.1'].toString()
ProcessingID	Mapper	msg['MSH']['MSH.11']['MSH.11.1'].toString()
PatientIDExternalID	Mapper	msg['PID']['PID.2']['PID.2.1'].toString()
PatientInternalId	Mapper	msg['PID']['PID.3']['PID.3.1'].toString()
PatientGivenName	Mapper	msg['PID']['PID.5']['PID.5.2'].toString()
FamilyName	Mapper	msg['PID']['PID.5']['PID.5.1'].toString()
PatientClass	Mapper	msg['PV1']['PV1.2']['PV1.2.1'].toString()
PointOfCare	Mapper	msg['PV1']['PV1.3']['PV1.3.1'].toString()
Bed	Mapper	msg['PV1']['PV1.3']['PV1.3.3'].toString()
AdmissionType	Mapper	msg['PV1']['PV1.4']['PV1.4.1'].toString()
UniversalServiceIdentifier	Mapper	msg['OBR']['OBR.4']['OBR.4.1'].toString()
ObservationDateTime	JavaScript	see JavaScript code 2
ObservationEndDateTime	JavaScript	see JavaScript code 3
channelMap.put("ObservationEndDateTime",datestring);		
RelevantClinicalInformation	Mapper	msg['OBR']['OBR.13']['OBR.13.1'].toString()
DiagnosticServiceSectionID	Mapper	msg['OBR']['OBR.24']['OBR.24.1'].toString()
ParentResult	Mapper	msg['OBR']['OBR.26']['OBR.26.1'].toString()
AdmitDateTime	Mapper	msg['PV1']['PV1.44']['PV1.44.1'].toString()
Vitals	JavaScript	see JavaScript code 4
RelevantClinicalInformation	Mapper	msg['OBR']['OBR.13']['OBR.13.1'].toString()

JavaScript code 1:

```
  var datestring = DateUtil.convertDate("yyyyMMddHHmmss", "MM/dd/yyyy
HH:mm:ss",
  msg["MSH"]["MSH.7"]["MSH.7.1"].toString());
  channelMap.put("DateTimeOfMessage",datestring);
```

JavaScript code 2:

```
  var datestring = DateUtil.convertDate("yyyyMMddHHmmss","MM/dd/yyyy
HH:mm:ss",
  msg["OBR"]["OBR.7"]["OBR.7.1"].toString());
  channelMap.put("ObservationDateTime",datestring);
```

JavaScript code 3:

```
  var datestring = DateUtil.convertDate("yyyyMMddHHmmss","MM/dd/yyyy
HH:mm:ss",
  msg["OBR"]["OBR.8"]["OBR.8.1"].toString());
  channelMap.put("ObservationEndDateTime",datestring);
```

JavaScript code 4:

```
  var len=msg['OBX'].length();
  var observation = '{  ';
  for(i=0;i<len;i++)
   {
   observation = observation + '"' + msg['OBX'][i]['OBX.3']
['OBX.3.1'].toString() + '":"' + msg['OBX'][i]['OBX.5']
['OBX.5.1'].toString() + '",';
   }
   observation = observation.substring(0,observation.length-1);
   observation = observation + '}'; channelMap.put('Vitals',observation);
```

HL7 Filters

Here are the Alarm and Vitals filters that can be imported if required.

Alarms Filter:

```
<filter>
 <rules>
  <rule>
   <sequenceNumber>0</sequenceNumber>
   <name>Accept message if "msg['OBR']['OBR.4']['OBR.4.1'].toString()"
 equals "Alarm-Start"</name>
   <data class="map">
    <entry>
     <string>Field</string>
     <string>msg['OBR']['OBR.4']['OBR.4.1'].toString()</string>
    </entry>
    <entry>
     <string>Equals</string>
     <string>1</string>
    </entry>
    <entry>
     <string>OriginalField</string>
     <string></string>
    </entry>
    <entry>
     <string>Values</string>
     <list>
      <string>"Alarm-Start"</string>
     </list>
```

```
   </entry>
   <entry>
    <string>Name</string>
    <string></string>
   </entry>
  </data>
  <type>Rule Builder</type>
  <script>if(msg['OBR']['OBR.4']['OBR.4.1'].toString() == "Alarm-Start")
   {
   return true;
   }
   return false;</script>
  <operator>NONE</operator>
 </rule>
 <rule>
  <sequenceNumber>1</sequenceNumber>
  <name>Accept message if "msg['OBR']['OBR.4']['OBR.4.1'].toString()"
equals "Alarm-Silenced"</name>
  <data class="map">
   <entry>
    <string>Field</string>
    <string>msg['OBR']['OBR.4']['OBR.4.1'].toString()</string>
   </entry>
   <entry>
    <string>Equals</string>
    <string>1</string>
   </entry>
   <entry>
    <string>OriginalField</string>
    <string></string>
   </entry>
   <entry>
    <string>Values</string>
    <list>
     <string>"Alarm-Silenced"</string>
    </list>
   </entry>
   <entry>
    <string>Name</string>
    <string></string>
   </entry>
  </data>
  <type>Rule Builder</type>
  <script>if(msg['OBR']['OBR.4']['OBR.4.1'].toString() == "Alarm-Silenced")
   {
   return true;
   }
   return false;</script>
  <operator>OR</operator>
 </rule>
 <rule>
  <sequenceNumber>2</sequenceNumber>
  <name>Accept message if "msg['OBR']['OBR.4']['OBR.4.1'].toString()"
equals "Alarm-End"</name>
  <data class="map">
   <entry>
    <string>Field</string>
    <string>msg['OBR']['OBR.4']['OBR.4.1'].toString()</string>
   </entry>
   <entry>
    <string>Equals</string>
    <string>1</string>
   </entry>
   <entry>
    <string>OriginalField</string>
    <string></string>
   </entry>
   <entry>
    <string>Values</string>
    <list>
     <string>"Alarm-End"</string>
    </list>
```

```
    </entry>
    <entry>
     <string>Name</string>
     <string></string>
    </entry>
   </data>
   <type>Rule Builder</type>
   <script>if(msg['OBR']['OBR.4']['OBR.4.1'].toString() == "Alarm-End")
    {
    return true;
    }
    return false;</script>
   <operator>OR</operator>
  </rule>
 </rules>
</filter>
```

Vitals Filter:

```
<filter>
 <rules>
  <rule>
   <sequenceNumber>0</sequenceNumber>
   <name>Accept message if "msg['OBR']['OBR.4']['OBR.4.1'].toString()"
 equals "Vital Signs"</name>
   <data class="map">
    <entry>
     <string>Field</string>
     <string>msg['OBR']['OBR.4']['OBR.4.1'].toString()</string>
    </entry>
    <entry>
     <string>Equals</string>
     <string>1</string>
    </entry>
    <entry>
     <string>OriginalField</string>
     <string></string>
    </entry>
    <entry>
     <string>Values</string>
     <list>
      <string>"Vital Signs"</string>
     </list>
    </entry>
    <entry>
     <string>Name</string>
     <string></string>
    </entry>
   </data>
   <type>Rule Builder</type>
   <script>if(msg['OBR']['OBR.4']['OBR.4.1'].toString() == "Vital Signs")
    {
    return true;
    }
    return false;</script>
   <operator>NONE</operator>
  </rule>
 </rules>
</filter>
```

Active Directory Import Procedures

This appendix lists the stored procedures used to import Active Directory users from a comma-separated file.

Stored Procedure : ImportActiveDirectoryUsers

```
USE [MVisumAlerts]
GO
/****** Object:  StoredProcedure [dbo].[ ImportActiveDirectoryUsers]    Script Date: 06/02/2015
 12:32:15 ******/
SET ANSI_NULLS ON
GO
```

```
SET QUOTED_IDENTIFIER ON
GO
CREATE PROCEDURE [dbo].[ImportActiveDirectoryUsers](
 @bulkUserInfo VARCHAR(MAX),
 @ServerIP VARCHAR(50),
 @DomainName VARCHAR(200)

)
as
BEGIN
   DECLARE @UserPipeSaperatedStrig Varchar(MAX),
     @SessionID VARCHAR(200),
     @IsImportUser BIT,
     @RoleId INT,
     @GroupId INT,
     @UnitId INT,
     @MobileTypeID INT,
     @Status VARCHAR(100),
     @HospitalId INT,
     @DomainId INT,
     @UserName VARCHAR(200),
     @FirstName VARCHAR(200),
     --@MiddleName VARCHAR(100),
     @LastName VARCHAR(200),
     @Email VARCHAR(200),
     @TelephoneNumber VARCHAR(100),
     @GroupName VARCHAR(200),
     @UnitName VARCHAR(200),
     @Index INT

   DECLARE @UserList TABLE
   (
     FirstName VARCHAR(200),
     --MiddleName VARCHAR(100),
     LastName VARCHAR(200),
     Email VARCHAR(200),
     TelephoneNumber VARCHAR(100),
     UserName VARCHAR(200),
     GroupName VARCHAR(200),
     UnitName VARCHAR(200)
   )

   INSERT @UserList(FirstName,LastName,Email,TelephoneNumber,UserName,GroupName,UnitName) EXEC
 [dbo].[GeBulkUsersInTable] @bulkUserInfo
   select * from @UserList

   SET @SessionID=(SELECT SessionId FROM [Session] where LogoutTime is null)
   SET @IsImportUser=1
   SET @RoleId=(SELECT RoleId FROM Roles WHERE RoleName = 'NURSE')
   SET @MobileTypeID=(SELECT Id FROM MobileTypes WHERE DisplayName = 'Apple-iPhone-3.0')
   SET @Status='ACTIVE'
   SET @HospitalId=(SELECT TOP 1 HospId FROM Hospitals)
   SET @DomainId=(SELECT ad.Id FROM ActiveDirectoryDomains ad
            LEFT JOIN ActiveDirectorySettings ads ON ads.Id=ad.ADId
          WHERE ads.ServerIP=@ServerIP AND ad.DomainName=@DomainName
       )
   SET @UserPipeSaperatedStrig='['
   SET @Index=0
   DECLARE db_cursor CURSOR FOR
   SELECT FirstName,LastName,Email,TelephoneNumber,UserName,GroupName,UnitName
   from @UserList

   OPEN db_cursor
   FETCH NEXT FROM db_cursor INTO
 @FirstName,@LastName,@Email,@TelephoneNumber,@UserName,@GroupName,@UnitName

   WHILE @@FETCH_STATUS = 0
   BEGIN
    print 'Inside Loop'
    SET @GroupId=(SELECT ID FROM HospitalGroups WHERE Name = @GroupName)
    SET @UnitId=(SELECT ID FROM HospitalUnits WHERE Name = @UnitName)

    IF(@Index>0)
    BEGIN
     SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig+','
    END

    --For UserName
    SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +'"'+ @UserName +'|'
    --For First Name
    SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +''+ @FirstName +'|'
    --For Last Name
    SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +''+ @LastName +'|'
    --For Description
    SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +'|'
    --For Email
    SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +''+ @Email +'|'
    --For Telephone Number
    SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +''+ @TelephoneNumber +'|'
    --For StreetAddress
```

```
   SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +'|'
   --For City
   SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +'|'
   --For ZipPostalCode
   SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +'|'
   --For StateProvince
   SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +'|'
   --For CountryRegion
   SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +'|'
   --For Role ID
   SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +''+ CAST(@RoleId AS VARCHAR ) +'|'
   --For Unit ID
   SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +''+  CAST(@UnitId AS VARCHAR ) +'|'
   --For Group ID
   SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +''+  CAST(@GroupId AS VARCHAR ) +'|'
   --For MobileTYpeID
   SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +''+  CAST(@MobileTypeID AS VARCHAR )
+'|'
   --For Status
   SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +''+  @Status +'|'
   --For DomainID
   SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +''+  CAST(@DomainId AS VARCHAR ) +'|'
   --For HospitalID
   SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +''+  CAST(@HospitalId AS VARCHAR ) +''
   SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +'"'

   print @UserPipeSaperatedStrig
   SET @Index =@Index+1
   FETCH NEXT FROM db_cursor INTO
@FirstName,@LastName,@Email,@TelephoneNumber,@UserName,@GroupName,@UnitName
  END
  SET @UserPipeSaperatedStrig=@UserPipeSaperatedStrig +']'
  CLOSE db_cursor
  DEALLOCATE db_cursor

  print @UserPipeSaperatedStrig

  EXEC [dbo].[AddUpdateActiveDirectoryUsers] @UserPipeSaperatedStrig,@SessionID,@IsImportUser

END
```

Stored Procedure : GeBulkUsersInTable

```
USE [MVisumAlerts]
GO
/****** Object:  StoredProcedure [dbo].[ GeBulkUsersInTable]    Script Date: 06/02/2015
 14:03:08 ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
-- =============================================================================
-- Author:  MVisum Inc.
-- Create date: 07/16/2014
-- Description: Convert Active Directory Users Comma Saperated List to Table
-- =============================================================================
CREATE PROCEDURE [dbo].[GeBulkUsersInTable]
 @UserListString VARCHAR(Max)
AS
BEGIN

 DECLARE @UserName VARCHAR(200)
 DECLARE @FirstName VARCHAR(200)
 --DECLARE @MiddleName VARCHAR(100)
 DECLARE @LastName VARCHAR(200)
 DECLARE @Email VARCHAR(200)
 DECLARE @TelephoneNumber VARCHAR(100)
 DECLARE @GroupName VARCHAR(200)
 DECLARE @UnitName VARCHAR(200)

 DECLARE @CommaIndex INT--,@TempEndIndex INT
 DECLARE @ColumnValue varchar(MAX)
 DECLARE @ColumnIndex INT

 /***Temporaty Variables***/
 DECLARE @CurrentUserString VARCHAR(MAX)
 DECLARE @StartIndex INT,@EndIndex INT
 SELECT   @UserName as UserName,@FirstName as FirstName,@LastName as LastName
   ,@Email as Email,@TelephoneNumber as TelephoneNumber,@GroupName as GroupName
   ,@UnitName as UnitName INTO #BulkUserInfoTable
 TRUNCATE TABLE #BulkUserInfoTable
 print @UserListString

 WHILE(CHARINDEX(CHAR(13)+CHAR(10),@UserListString) > 0)
 BEGIN
  Print 'INSIDE LOOP'
  SET @StartIndex = 0

  SET @EndIndex = CHARINDEX(CHAR(13)+CHAR(10),@UserListString)
  SET @CurrentUserString=SUBSTRING(@UserListString,0,@EndIndex)
```

```
 print @CurrentUserString

 IF(@CurrentUserString <> '')
 BEGIN

   SET @ColumnIndex=0
   Print 'INSIDE CONDITION'
   WHILE(CHARINDEX(',',@CurrentUserString) > 0)
   BEGIN

    SET @CommaIndex = CHARINDEX(',',@CurrentUserString)
    SET @ColumnValue=LTRIM(RTRIM(SUBSTRING(@CurrentUserString,0,@CommaIndex)))
    SET @CurrentUserString=SUBSTRING(@CurrentUserString,@CommaIndex+1,LEN(@CurrentUserString))
    print @CurrentUserString
    print @ColumnValue
    IF @ColumnIndex=0
    BEGIN
     SET @FirstName = (SELECT CASE WHEN @ColumnValue IS NOT NULL AND @ColumnValue <> '' THEN
@ColumnValue ELSE NULL END)
    END
    IF @ColumnIndex=1
    BEGIN
     SET @LastName = (SELECT CASE WHEN @ColumnValue IS NOT NULL AND @ColumnValue <> '' THEN
@ColumnValue ELSE NULL END)
    END
    IF @ColumnIndex=2
    BEGIN
     SET @Email = (SELECT CASE WHEN @ColumnValue IS NOT NULL AND @ColumnValue <> '' THEN
@ColumnValue ELSE NULL END)
    END
    IF @ColumnIndex=3
    BEGIN
     SET @TelephoneNumber =(SELECT CASE WHEN @ColumnValue IS NOT NULL AND @ColumnValue <> ''
THEN @ColumnValue ELSE NULL END)
    END
    IF @ColumnIndex=4
    BEGIN
     SET @UserName = (SELECT CASE WHEN @ColumnValue IS NOT NULL AND @ColumnValue <> '' THEN
@ColumnValue ELSE NULL END)
    END
    IF @ColumnIndex=5
    BEGIN
     SET @GroupName = (SELECT CASE WHEN @ColumnValue IS NOT NULL AND @ColumnValue <> '' THEN
@ColumnValue ELSE NULL END)
    END


    IF (CHARINDEX(',',@CurrentUserString)=0) AND @ColumnIndex=5
    BEGIN
     SET @UnitName = LTRIM(RTRIM(@CurrentUserString))
       END

    SET @ColumnIndex=@ColumnIndex + 1
   END


   INSERT INTO #BulkUserInfoTable( UserName, FirstName, LastName, Email, TelephoneNumber,
GroupName, UnitName )
   VALUES ( @UserName, @FirstName, @LastName, @Email, @TelephoneNumber, @GroupName,
@UnitName )

 END

 SET @UserListString=SUBSTRING(@UserListString,@EndIndex + 1,LEN(@UserListString))

END

IF(@UserListString <> '')
 BEGIN
   SET @CurrentUserString = @UserListString
   SET @ColumnIndex=0
   Print 'INSIDE CONDITION'
   WHILE(CHARINDEX(',',@CurrentUserString) > 0)
   BEGIN

    SET @CommaIndex = CHARINDEX(',',@CurrentUserString)
    SET @ColumnValue=LTRIM(RTRIM(SUBSTRING(@CurrentUserString,0,@CommaIndex)))
    SET @CurrentUserString=SUBSTRING(@CurrentUserString,@CommaIndex+1,LEN(@CurrentUserString))
    print @CurrentUserString
    print @ColumnValue
    IF @ColumnIndex=0
    BEGIN
     SET @FirstName = (SELECT CASE WHEN @ColumnValue IS NOT NULL AND @ColumnValue <> '' THEN
@ColumnValue ELSE NULL END)
    END
    IF @ColumnIndex=1
    BEGIN
     SET @LastName = (SELECT CASE WHEN @ColumnValue IS NOT NULL AND @ColumnValue <> '' THEN
@ColumnValue ELSE NULL END)
    END
```

```
    IF @ColumnIndex=2
    BEGIN
     SET @Email = (SELECT CASE WHEN @ColumnValue IS NOT NULL AND @ColumnValue <> '' THEN
@ColumnValue ELSE NULL END)
    END
    IF @ColumnIndex=3
    BEGIN
     SET @TelephoneNumber =(SELECT CASE WHEN @ColumnValue IS NOT NULL AND @ColumnValue <> ''
THEN @ColumnValue ELSE NULL END)
    END
    IF @ColumnIndex=4
    BEGIN
     SET @UserName = (SELECT CASE WHEN @ColumnValue IS NOT NULL AND @ColumnValue <> '' THEN
@ColumnValue ELSE NULL END)
    END
    IF @ColumnIndex=5
    BEGIN
     SET @GroupName = (SELECT CASE WHEN @ColumnValue IS NOT NULL AND @ColumnValue <> '' THEN
@ColumnValue ELSE NULL END)
    END


    IF (CHARINDEX(',',@CurrentUserString)=0) AND @ColumnIndex=5
    BEGIN
     SET @UnitName = LTRIM(RTRIM(@CurrentUserString))
       END

    SET @ColumnIndex=@ColumnIndex + 1
   END


   INSERT INTO #BulkUserInfoTable( UserName, FirstName, LastName, Email, TelephoneNumber,
GroupName, UnitName )
   VALUES ( @UserName, @FirstName, @LastName, @Email, @TelephoneNumber, @GroupName,
@UnitName )

 END


SELECT FirstName, LastName, Email, TelephoneNumber, UserName, GroupName, UnitName FROM
#BulkUserInfoTable

drop table #BulkUserInfoTable

END
```