

# Magic xpi 4.14 Deployment Guide



OUTPERFORM THE FUTURE™

Introduction.....	3
Understanding IMM Architecture and Terminology .....	3
Installation Overview .....	4
Installation Prerequisites .....	4
Setup Wizard Steps .....	5
Post Setup Wizard Configuration Steps.....	5
Configuring Linux Host for IMM .....	10
Prerequisites .....	10
Enabling WSL.....	10
Installing Ubuntu .....	10
Installing MicroK8S.....	10
Finding WSL Machine IP Address.....	11
Enabling MicroK8S add-ons .....	11
Managing IMM Domain Name in xpi Installation.....	12
Firewall Settings .....	13
Project Deployment.....	14
Projects Folder Location.....	14
Deploying a Project .....	14
Starting and Stopping Projects from the Monitor (Optional) .....	14
Starting Projects from the Command Line (Optional) .....	16
Magic Monitor Users.....	16
Deploying Web Requesters and SOAP Servers .....	17
Prerequisites .....	17
Java Web Requester .....	17
Licensing .....	18
Reserving License Threads for a Project .....	18
Host Locked License .....	18
About Magic Software Enterprises.....	19

# Introduction

This guide provides recommendations and guidelines for deploying Magic xpi 4.14 and its underlying middleware, the all new "In-memory Middleware" server.

The guide will describe the steps for installing and configuring Magic xpi, installing the IMM and integrating Magic xpi with the IMM.

## Understanding IMM Architecture and Terminology

Magic xpi server is built over the In-Memory Middleware provided with the xpi installer and this middleware serves as an underlying messaging layer for the infrastructure.

**IMM-Agent:** Each host running the xpi project servers (also interchangeably referred as xpi servers in this document) will need the IMM-Agent running. This agent is responsible for handling requests to stop and start xpi servers, and do health check of those servers. These IMM-Agents from multiple hosts are unified and connected via the IMM.

Let's learn more about the components that define In-memory Middleware:

- **IMM-Controller**  
The IMM-Controller is the process that the IMM-Agent communicates with and accepts engine start and stop requests.
- **IMM-DB**  
The IMM-DB is the central repository where all operational data required for the IMM to function is stored. This includes all xpi related runtime data, project metadata and real time execution data.
- **LOGDB**  
The LOGDB serves as a container for all activity logs generated by the xpi server.
- **IMM-Tunnel**  
This is the Web service that resides within the IMM architecture. It handles all your external requests (for example, HTTP requests, TCP requests) and redirects those into the IMM-Controller.  
**Note:** This does not replace the Web Server that you would need for your HTTP Web Server triggers and so on.
- **xpi Monitor**  
The monitor serves as your central interface to provide you information and health related data about the IMM and the xpi projects running under it. This also includes the data like the graphs of your license usage, the server load, the workers and so on.  
**Note:** Unlike the previous versions of xpi the xpi Monitor resides within the IMM infrastructure and is not an installable service this is provided with the xpi server/studio Installation.



# Installation Overview

Magic xpi installation involves the following steps:

- Preparing and verifying the installation prerequisites.
- Installing the Magic xpi 4.14 server on each of the application servers using the setup wizard. During setup, the server startup and initial configuration is performed.
- Performing post setup wizard configuration steps for integrating IMM with Magic xpi.
- Deploying the project to a shared folder and configuring project startup.
- Optional: Deploying Web requesters and SOAP servers.

## Installation Prerequisites

Before installing, you need to verify the following prerequisites:

- The xpi server installs Windows services and therefore the user profile under which the xpi installation is made, need to have the appropriate permissions to run a service.
- All application servers have network access and read/write permissions for the Magic xpi admin user to access the shared projects folder.
- IMM Middleware is deployed as a Kubernetes cluster and therefore requires a Linux host with Kubernetes as a prerequisite. This could be provided by a VM or Ubuntu app installed using WSL on Windows.



For the proper communication between xpi server and Kubernetes cluster, clocks on all hosts must be in sync with tolerance offset of not more than 15 seconds. For more information, refer to [Timezone Setting for xpi Monitor](#).

## Setup Wizard Steps

On each application server, run the setup program (**setup.exe**) from the installation media.

This setup.exe will open a wizard and guide you through the installation process. For the detailed information, refer to the *Installation Guide.pdf*.

## Post Setup Wizard Configuration Steps

### Configuring the Windows Service User

In all of the application servers, go to the **Services** control panel and change the service user to a user with enough privileges to access all resources required by the project. For example, the user should have permission to read and write to the network drive holding the projects.

### Installing the Server License

Magic xpi 4.14 requires a license to run.

1. Copy the license file that you received after purchase, and copy it to all of the servers, or to the shared location.
2. For all the servers, modify the **Magic.ini** file with the license file location by updating the **[MAGIC\_ENV] LicenseFile** entry to point to the shared license file.  
For example: **LicenseFile = \\10.1.1.6\licenses\License.dat**.
3. Make sure that your project's **ifs.ini** file is configured with the production license (i.e. **[MAGIC\_ENV]LicenseName = IBPRSRVI** for a Windows machine).

**Note:** The first use of the Host Locked License has to be from the host for which the license is registered.

### Installing and Connecting to IMM Middleware

#### Prerequisites

The following tools are required for accessing, deploying and managing the IMM middleware:

- Windows Host running the IMM installer with:
  - [Kubernetes CLI](#) (kubectl)
  - [Helm](#)
  - [PuTTY](#) or any equivalent SSH and Telnet client
  - [Chocolatey](#) (choco) or equivalent package manager (optional)
- A Linux server with Kubernetes up and running



## Installing Required Tools

The primary prerequisites, Kubernetes CLI and helm can be installed using different methods. In this document we will follow the approach of installing these tools using Chocolatey package manager. The Magic xpi installer also ships the inst\_k8s\_tools.bat batch file for doing the setup. You can locate the batch file under the <Magic xpi Installation>\InMemoryMiddleware\deploy folder.

To begin with, let's start installing Chocolatey.

## Installing Chocolatey

Chocolatey being the package manager, you need to install it as an administrator.

Open PowerShell as Administrator on the Windows machine.

Run following command to install Chocolatey

```
Set-ExecutionPolicy Bypass -Scope Process -Force;
[System.Net.ServicePointManager]::SecurityProtocol =
[System.Net.ServicePointManager]::SecurityProtocol -bor 3072; iex
((New-Object
System.Net.WebClient).DownloadString('https://community.chocolatey.org/install.ps1'))
```

## Installing Kubernetes CLI and Helm

Once Chocolatey is successfully installed, we can use it to install Kubernetes CLI and Helm.

Open Command Prompt as administrator and run the following commands:

- `choco install kubernetes-cli`
- `choco install kubernetes-helm`

**Note:** Machine restart may be required after installing the required tools.

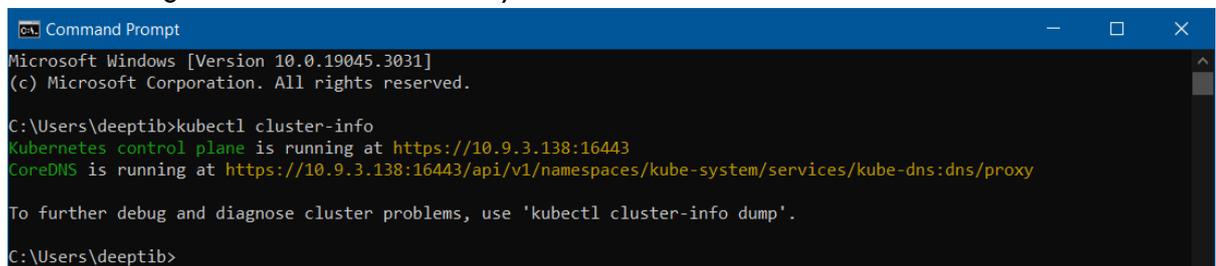
## Getting the kubeconfig

The Kubernetes CLI (kubectl) interacts with the server defined in a config file usually placed in your user profile folder. Let's get this config file from the server running Kubernetes using an SSH/Telnet tool.



To setup this file,

- Open Putty or equivalent software to open a shell into the host machine for the IMM setup. Once Putty connects, run the command `microk8s config > config`
- Copy the output file "config" to your Windows host under the %UserProfile%/.kube folder. If this folder does not exist, you can create the folder by running the following two commands from Command Prompt.
  - `cd /D %UserProfile%`
  - `mkdir .kube`
- Test the configuration using the following command `kubectl cluster-info`
- On executing the command successfully, it will show the IMM Cluster details.



```
Command Prompt
Microsoft Windows [Version 10.0.19045.3031]
(c) Microsoft Corporation. All rights reserved.

C:\Users\deeptib>kubectl cluster-info
Kubernetes control plane is running at https://10.9.3.138:16443
CoreDNS is running at https://10.9.3.138:16443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.

C:\Users\deeptib>
```

## Accessing IMM Host

If the IMM host address you provided during the installation is not a publicly accessible host, then you need to set it in the hosts file.

Open the hosts file at location `C:\Windows\System32\drivers\etc` in a text editor.

Add an entry to this file in the format below:

<IP Address of Kubernetes Cluster> <Host Domain Name>

For example, if you have given the host as `xpi.factoryeye.net` and the IP Address of the machine is `10.1.2.3` then the entry will look like

```
10.1.2.3    xpi.factoryeye.net
```

## Deploying IMM

To deploy IMM, do the steps given below:

- Go to the <Magic xpi installation>\InMemoryMiddleware\deploy folder.

- Open the Command Prompt and run the **deploy-imm.bat** batch file. The file takes the following arguments in sequential order given below. If none of the arguments are provided, then it will setup the cluster with default values.

```
deploy-imm.bat <DOMAIN_NAME> <HELM_VERSION> <GLOBAL_SETTINGS_PATH>
<DB_ALERT_CONFIG_PATH> <LOGBACK_PATH> <MONITOR_CONFIG_PATH>
```

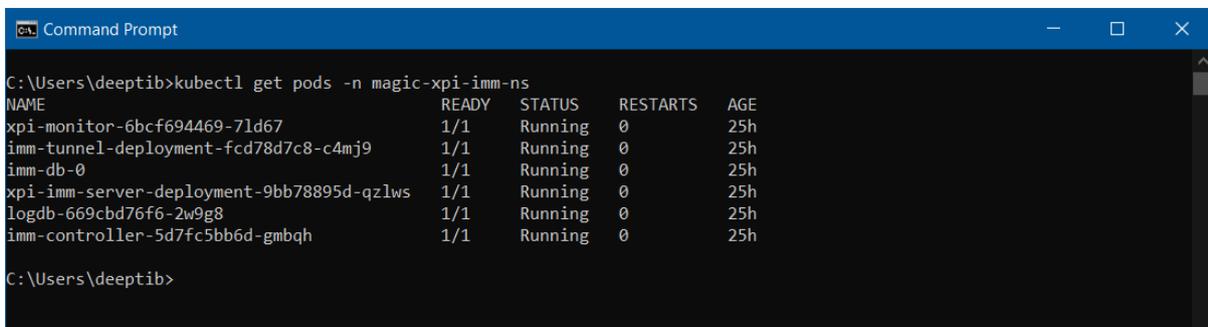
- GLOBAL\_SETTINGS\_PATH: Provide the path of the *globalsettings.properties*. In this file you can configure the properties for time interval and retry mechanism for IMM startup and initialization.
- DB\_ALERT\_CONFIG\_PATH - Provide the path of the *db\_alert\_config.properties* file. In this file you can configure the email alerts for any critical state, high CPU utilization and Memory usage in IMM.
- LOGBACK\_PATH: Provide the path of the *logback.xml*. In this file you can configure the loggers, appenders and change the log level for the different xpi components.
- MONITOR\_CONFIG\_PATH: Provide the path to the monitor config file. In this file you can configure the SSO login for xpi monitor and change the Locale (language) namely English (en-US) or Japanese (jp-JP) for the same.

If no arguments are provided, then the setup will pick the required values from the configuration files (except DB Alert) in the config folder. The DB Alert default file will be picked from Runtime\config folder.

- Once the IMM setup is complete, you can check whether all PODs are running in the IMM Cluster using the following command:

```
kubectl get pods -n magic-xpi-imm-ns
```

It should show output as below with all the PODs in running state.



```

C:\Users\deeptib>kubectl get pods -n magic-xpi-imm-ns
NAME                                READY   STATUS    RESTARTS   AGE
xpi-monitor-6bcf694469-71d67        1/1     Running   0           25h
imm-tunnel-deployment-fcd78d7c8-c4mj9 1/1     Running   0           25h
imm-db-0                             1/1     Running   0           25h
xpi-imm-server-deployment-9bb78895d-qz1ws 1/1     Running   0           25h
logdb-669cbd76f6-2w9g8              1/1     Running   0           25h
imm-controller-5d7fc5bb6d-gmbqh      1/1     Running   0           25h
C:\Users\deeptib>

```

You can also confirm if all the services are running by running the URLs in the browser.

- Magic Monitor:  
https://[IMM Domain Name]/magicmonitor/#/auth/login  
For example, https://xpi.factoryeye.net/magicmonitor/#/auth/login
- IMM-Tunnel  
https://[IMM Domain Name]/immtunnel  
For example, <https://xpi.factoryeye.net/immtunnel>



The time and timezone values on the machine running the IMM-Agent and the IMM Host machine must be the same. Different values for time and timezones may result in incorrect execution of the Recovery policy.

## Timezone Setting for xpi Monitor

The timezone for xpi Monitor can be defined using the TZ environment variable under the immMonitor deployment section of the values.yaml file located in the <Magic xpi installation>\InMemoryMiddleware\deploy folder. This timezone will be used by the xpi monitor to display the time based on the timezone defined on all the tabs in the Monitor. The default time zone is US/Central. You can change it as per your requirement.

## Starting the Agent

To start the agent,

- Go to <Magic xpi installation>\InMemoryMiddleware\agent and run the *imm-agent.exe* application from Command Prompt.
- There is no need to download or install any additional certificates.

**Note:** The IMM-Agent should always be in a running state before running or debugging xpi projects.

## Configuring Multiple Network Cards (Optional)

If your application servers have multiple network cards, configure the use of a specific card for the Magic xpi 4.14 server, as follows:

1. Modify the NIC\_ADDR value found in the <Magic xpi installation>\InMemoryMiddleware\agent\.env file to hold either the IP assigned to this network card or the name of the network card itself.  
For example: NIC\_ADDR=10.1.1.11 or NIC\_ADDR="#eth0:ip#", where eth0 is the name of the network card.

**Note:** The host name or IP address should not be surrounded by quotation marks.



# Configuring Linux Host for IMM

To host the IMM infrastructure, you can use a Linux machine either on premise or on Cloud. For development purpose, you can setup the same on WSL as well.

Given below are the steps to setup IMM infrastructure on WSL using MicroK8S.

## Prerequisites

- Windows 10 version 1909(OS build 18363.1049) or above

## Enabling WSL

To enable the WSL feature refer to the official WSL documentation at <https://docs.microsoft.com/en-us/windows/wsl/install>

**Note:** Make sure that your WSL is updated one. You can update the same by running the command:

```
wsl --update
```

## Installing Ubuntu

1. Go to the Windows store and search for Ubuntu 22.04.x LTS. This is tested and verified on Windows 10 Pro Version 22H2.
2. Click the Get button to download it on the local machine.
3. Once the download is complete, click the Open button. This will start the installation process. It will take a few minutes depending upon the system configuration.
4. Next, you will be prompted to create a UNIX user account. Follow the instructions and set the username and password. With this the installation will be complete.

**Note:** Remember the username and password. You will require this later.

## Installing MicroK8S

1. Install MicroK8S using the command  

```
sudo snap install microk8s --classic
```

**Note:** On running this command, if you get error similar to

*error: cannot communicate with server: Post "http://localhost/v2/snaps/microk8s": dial unix /run/snapd.socket: connect: no such file or directory*



then run the following commands and then reinstall MicroK8S

- `sudo apt-get update && sudo apt-get install -yqq daemonize dbus-user-session fontconfig`
- `sudo daemonize /usr/bin/unshare --fork --pid --mount-proc /lib/systemd/systemd --system-unit=basic.target`
- `exec sudo nsenter -t $(pidof systemd) -a su - $LOGNAME`

To confirm MicroK8S is successfully installed, run the command

```
microk8s status
```

This command should show the status as *running*.

## Finding WSL Machine IP Address

To know the IP address of WSL machine, run the following command:

```
ifconfig
```

If you get an error on running the command, first run the command below:

```
sudo apt install net-tools
```

Note the IP Address of the system. You will need this to enable MicroK8S add-ons.

## Enabling MicroK8S add-ons

Once MicroK8S is installed, you need to install microk8s add-ons given below:

1. dns

To enable this add-on run the command

```
sudo microk8s enable dns
```

2. metallb

To enable this add-on run the command

```
sudo microk8s enable metallb
```

When it prompts for IP range, specify it as *<IP Address of WSL><IP Address of WSL>*

Once the two add-ons are enabled your system is ready for IMM installation.

## Managing IMM Domain Name in xpi Installation

The IMM Domain name provided during the installation is referred in multiple files listed below. If there is a change in the domain name, change has to be done in all these files:

- [Magic xpi Installation]\Runtime\Magic xpa\magic.ini
- [Magic xpi Installation]\InMemoryMiddleware\agent\.env
- [Magic xpi Installation]\InMemoryMiddleware\imm-logger\IMM\_Controller\_Log.bat
- [Magic xpi Installation]\InMemoryMiddleware\imm-logger\IMM\_Tunnel\_Log.bat
- [Magic xpi Installation]\Runtime\MgxpiCmdl.bat
- [Magic xpi Installation]\Runtime\mgreq.ini
- [Magic xpi Installation]\Runtime\scripts\config\mgreq.ini
- [Magic xpi Installation]\xpi\_webserver\webservice-config\config.properties
- [Magic xpi Installation]\Studio\Debugger\Install\_Debug\_services.bat
- [Magic xpi Installation]\Studio\Debugger\apache-tomcat\bin\service.bat

Apart from the files in the xpi installation, the domain name should be updated in the **hosts** file as well.

The Magic xpi installation also ships the **imm-updater** utility which will automatically update the list of files specified above with the updated IMM domain name. This utility is located under <Magic xpi installation>\InMemoryMiddleware\imm-updater. The detailed instructions to run this utility are given in the readme.txt in the same folder.



## Firewall Settings

The IMM requires access to the following and these ports should allow access in the firewall settings. Configure your network infrastructure to allow inbound and outbound connections to all these ports.

Component	Port
LOGDB	27017
IMM-DB	6739
Controller	80 and 443
The services other than IMM for which the port should be enabled is:	
Webservices	6443



This is not a comprehensive list and if any additional ports are required, they should be allowed in firewall configuration.

# Project Deployment

## Projects Folder Location

You can deploy Magic xpi projects to a shared folder, accessible by all application servers. Although a shared folder is potentially a single point-of-failure, with today's storage systems this type of resource is usually highly redundant by itself.

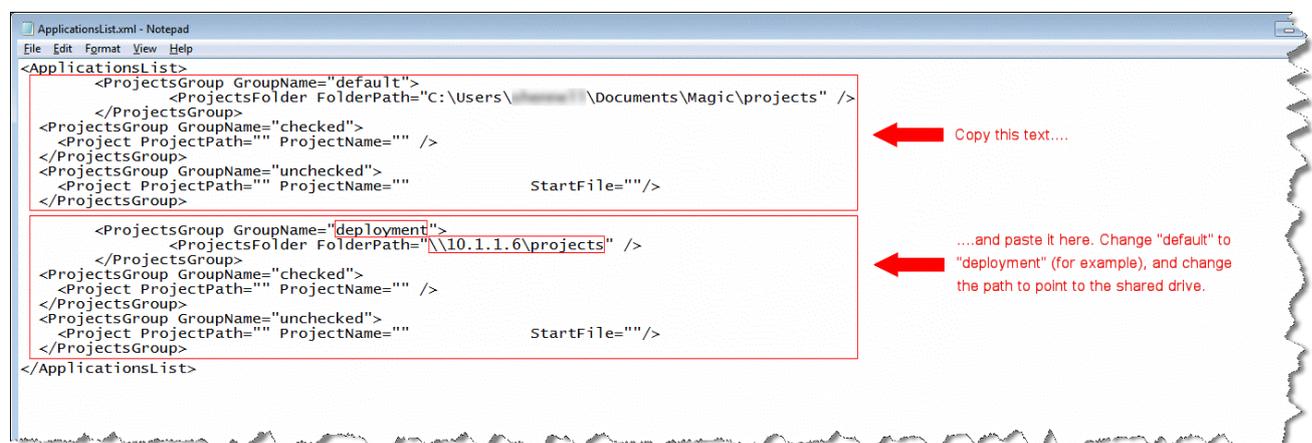
## Deploying a Project

Deploying a project involves copying each project's folder from the development/staging environment to the designated shared folder location. For projects with connections to external resources (such as databases or enterprise systems), it might be necessary to edit the configuration using the **Settings** dialog box's **Resources** or **Services** sections with the relevant server connections and credentials.

## Starting and Stopping Projects from the Monitor (Optional)

The Monitor shows the projects based on the **ApplicationsList.xml** file, which is located in the **Runtime\config** folder.

1. Copy the text in the file, which is the following, by default, and paste it as a new instance underneath the existing instance:



The **ProjectsFolder FolderPath** is the path where all of the projects are saved.

The **Project ProjectPath** is the path to a unique project folder for one project containing the project's files, including the start.xml file.

2. In the new section, change the word **default** to another name, such as **deployment**.
3. In the new section, change the path to point to the shared drive, such as

\\10.1.1.6\projects.

4. Save the file.
5. Refresh the Monitor to apply the settings by clicking the **Magic xpi** tab and then the **Browse all projects** link.

**Note:** If you do not want projects from a particular group to be displayed in the Monitor, remove the entire block of the **<ProjectsGroup>** tag instead of just leaving the **FolderPath** entry empty.

Here is an example of an **ApplicationsList.xml** file:

```
<ApplicationsList>
  <ProjectsGroup GroupName="default">
    <ProjectsFolder FolderPath="c:\Users\ABC\Documents\Magic\Projects"/>
  </ProjectsGroup>

  <ProjectsGroup GroupName="My first 2 projects">
    <Project ProjectPath="C:\Temp\projects\Project1\Project1" Project-
Name="Project1"/> <!--a project folder that contains the project's files including
the start.xml file-->
    <Project ProjectPath="C:\Temp\projects\Project2" Project-
Name="Project2" StartFile="start2.xml"/> <!--a project folder that contains the
project's files including a "non-default" start2.xml-->
  </ProjectsGroup>

  <ProjectsGroup GroupName="all QA's projects"> <!--an additional and separate
projects folder-->
    <ProjectsFolder FolderPath="D:\Different Project Folder\QA"/>
  </ProjectsGroup>
</ApplicationsList>
```

**Note:**

- When working with multiple hosts with xpi server connecting to the IMM, the list of the projects in the **ApplicationList.xml** should be identical on all the hosts.
- The value "localhost" is not supported as host name in the start.xml file of the Magic xpi project and may result in project going in "WAITING\_FOR\_AGENT" status indefinitely.
- Automatic project startup is not supported via Projectstartup.xml.



## Starting Projects from the Command Line (Optional)

Projects can be started from the command line. To start a project from the command line:

1. Go the **Start** link in the project's folder.
2. Right-click and select the **Properties** option.
3. Find the **Target** field. The text in the field will look something like this:  

```
"<Magic xpi installation>\Runtime\MgxpiCmd1.bat" start-servers  
-startup-config-file "<Magic xpi In-  
stallation>\Runtime\projects\Project1\start.xml"
```
4. This path points to a specific **start.xml** file that is residing under the project. Change the path to load a different **start.xml** file whose configurations can load several projects.
5. Click **OK**.

## Magic Monitor Users

A default user is set during the installation process. The login details for this user are:

- User: **Operator**. The operator can start and stop projects:
- a. Username: **administrator**
  - b. Initial password: **changeit**

## Azure Authentication for xpi Monitor

Alternatively, you can login to xpi Monitor using Azure-SSO. For this, you need to perform two steps.

1. Configure Certificates on the IMM Host Machine

Azure authentication requires valid certificates with the relevant domain name installed on the IMM Host machine where the Kubernetes cluster is deployed. For this, get the certificates for the domain from the appropriate Certificate Authority and copy the certificate and its key to the IMM Host machine. For detailed instructions, refer to the Linux documentation for installing and configuring the certificates on the Linux host.

2. Configure Azure SSO Details in Monitor Config

Go to the **config.json** file under the `<Magic xpi installation>\InMemoryMiddleware\config` folder. Here edit the **clientId**, **authority** and the **redirectUrl** for authenticating with the Azure SSO. You can work with your system administrator to get these values.

Once these steps are completed, you will have to **redploy** the IMM for the changes to reflect.

Now on the xpi Monitor login page, use the *Login with Azure-SSO* button. The default user name and passwords will not work in this mode.

Azure authentication requires the xpi Monitor to be accessed using HTTPS protocol.



# Deploying Web Requesters and SOAP Servers

## Prerequisites

- For the Web requester, IIS7 is required.
- For the Java Web requester, Apache Tomcat 8 or above is required.

Since the Magic xpi Service will not be running on this machine, it will only be used to host the external requesters and will not run any of the grid components or the Magic xpi engines.

## Java Web Requester

For the Java Web requester, follow the instructions contained in a PDF file called **Magic xpi - Web Server Installation.pdf**. This PDF is found at:

**<Magic xpi installation>\Runtime\Support\JavaWebRequester\Tomcat**

## Licensing

The Magic xpi 4.14 license mechanism is floating by nature, with an option to reserve a minimum number of license threads for each project.

### Reserving License Threads for a Project

If you are running multiple projects, and some projects may consume all threads under stress conditions (for example, there are many Web services and/or HTTP requests), you should consider reserving a minimum fixed amount of license threads for critical projects that need to run continually. For information about how to do this, refer to the **Reserving License Threads** topic in the Magic xpi Help.

### Host Locked License

All licenses produced by Magic Software Enterprises are host locked. Based on this rule all xpi projects that are initialized from any other host will block processing of flow request. This is true until an xpi server is started from the host for which the license was provided. Note this restriction is only for a fresh instance of IMM. It is not even required that the Magic xpi server on the host locked server continues to run.

# About Magic Software Enterprises

Magic Software Enterprises (NASDAQ: MGIC) empowers customers and partners around the globe with smarter technology that provides a multi-channel user experience of enterprise logic and data.

We draw on 30 years of experience, millions of installations worldwide, and strategic alliances with global IT leaders, including IBM, Microsoft, Oracle, Salesforce.com, and SAP, to enable our customers to seamlessly adopt new technologies and maximize business opportunities.

For more information, visit [www.magicsoftware.com](http://www.magicsoftware.com).



Magic is a registered trademark of Magic Software Enterprises Ltd. All other product and company names mentioned herein are for identification purposes only and are the property of, and might be trademarks of, their respective owners.

Magic Software Enterprises has made every effort to ensure that the information contained in this document is accurate; however, there are no representations or warranties regarding this information, including warranties of merchantability or fitness for a particular purpose. Magic Software Enterprises assumes no responsibility for errors or omissions that may occur in this document. The information in this document is subject to change without prior notice and does not represent a commitment by Magic Software Enterprises or its representatives.

© Magic Software Enterprises, 2023 - 2024

