# **Port Bindings**

Azure App Service applications are hosted on a worker machine behind a load balancing front end router. The front end is a shared resource that is not associated with a particular tenant. The front-end load balancer listens for standard HTTP traffic on TCP port 80, and for encrypted HTTPS traffic on TCP port 443. It does not listen on other TCP ports.

This check examines the bindings on the site to be migrated and verifies that it is not listening on TCP ports other than 80 and 443. It will be necessary to configure the site to listen only on these ports before it can be migrated.

### Potential Remediation steps:

* Remove any ports other than 80/443.
* For more information in changing these settings for IIS see [How to change the TCP port for IIS services.](https://support.microsoft.com/en-us/help/149605/how-to-change-the-tcp-port-for-iis-services)

# **Protocols**

Azure App Service applications support both non-encrypted HTTP requests, and encrypted HTTPS requests.

This check examines the bindings on the site to be migrated and verifies that it does not host non-HTTP protocols, such as the net.tcp protocol used by Windows Communication Foundation. It will be necessary to remove any non-HTTP bindings from the site before it can be migrated.

### Potential Remediation steps:

* Remove any non-HTTP protocol bindings
* For more information in changing these settings for IIS see:
  + [How to change the TCP port for IIS services.](https://support.microsoft.com/en-us/help/149605/how-to-change-the-tcp-port-for-iis-services)
  + [Remove-IISSiteBinding](https://docs.microsoft.com/en-us/powershell/module/iisadministration/remove-iissitebinding?view=win10-ps)
  + [How to add binding information to a site](https://docs.microsoft.com/en-us/iis/configuration/system.applicationhost/sites/site/bindings/binding#how-to-add-binding-information-to-a-site)

##### **Certificates**

This check examines the site to be migrated to see if it is listening for encrypted HTTPS requests. If HTTPS is detected, the site can be migrated - but if a custom TLS certificate is desired, the user will be alerted that it needs to be installed separately after migration has completed.

### Potential Remediation steps:

* No action required prior to migration.
* Optional recommended steps:
* For information on configuring certificates after migration see [Bind certificate to app.](https://docs.microsoft.com/en-us/azure/app-service/web-sites-purchase-ssl-web-site)

##### **Location Tags**

Azure App Service applications are hosted using a preconfigured IIS applicationhost.config file that is built using the selected options for the application.

This check examines the machine to be migrated to see if it depends on configuration options that exist in the IIS applicationhost.config file, but are applied at a different level using Location Tags. Any such Location Tag entries must be removed from the applicationhost.config file before the site can be migrated.

Generally, these entries can be moved to either the site's root web.config file, or to a web.config file associated with the specific application to which they apply. Once they've been moved to one of these locations, they can be migrated to Azure App Service with the site content.

### Potential Remediation steps:

* Remove the location tags
* For more information see [Location Tags.](https://docs.microsoft.com/en-us/iis/get-started/planning-your-iis-architecture/deep-dive-into-iis-configuration-with-iis-7-and-iis-8)

# **ISAPI Filters**

ISAPI filters enable custom behavior for an IIS server or site. ISAPI filters are commonly used by add-on or 3rd party products that extend IIS. It is possible that some of these ISAPI filters could be supported in an Azure App Service application, but this would need to be evaluated and handled on a case-by-case basis.

This check examines the site to be migrated for ISAPI filters. If any filters are detected, they must to be removed before the site can be migrated.

### Potential Remediation steps:

* Remove ISAPI filters
* For further information on modifying ISAPI filters see [How to add an ISAPI filter](https://docs.microsoft.com/en-us/iis/configuration/system.webserver/isapifilters/filter).

# **Application Pools**

Azure App Service applications currently support a single IIS application pool per App Service Application.

This check examines the site to be migrated for sites that use multiple applications hosted in separate application pools. It will be necessary to configure the site to be migrated to use a single application pool before it can be migrated.

### Potential Remediation steps:

* Move the workloads to a single application pool and remove additional application pools.
* A common scenario where this might be configured is a site requiring multiple versions of the .NET framework. Currently this is not a supported configuration.
  + Separate Azure app service application would need to be created for each of the frameworks.
* For further information see [How to configure the application pool for an existing site or application.](https://docs.microsoft.com/en-us/iis/configuration/system.applicationhost/applicationpools/)

# **Application Pool Identity**

Azure App Service applications are hosted in an IIS worker process that runs in a system managed account associated with the application pool. This is also the default identity used by on-premises installations of IIS.

This check examines the application pool associated with the site to be migrated to ensure that it is configured to run as ApplicationPoolIndentity. If the site to be migrated is configured to run as the identity of a custom Windows account, as the Local System account, or as the Network Service account, then the site must be changed to ApplicationPoolIdentity before it can be migrated.

### Potential Remediation steps:

* Set the Application pool to run as ApplicationPoolIdentity.
* For further information see [Configuring IIS Application Pool Identities](https://docs.microsoft.com/en-us/iis/manage/configuring-security/application-pool-identities).

# **Authentication Type**

On-premises installations of IIS support several different authentication types, including Windows Authentication, Basic Authentication, Client Certificate Authentication and more. Azure App Service applications support a different set of authentication types.

This check examines the site to be modified to determine which authentication types are in use. If authentication types other than Anonymous Authentication are detected, it will be necessary to change the authentication types to Anonymous Authentication before the site can be migrated. After the migration is complete, it will be possible to configure the site using one of the Azure App Service supported authentication types.

### Potential Remediation steps:

* Enable anonymous authentication.
* For further information on configuring these settings see [Anonymous Authentication.](https://docs.microsoft.com/en-us/iis/configuration/system.ftpserver/security/authentication/)

# **Application Settings**

ASP.NET supports declaration of application settings in web.config files. These settings are specific to the application and will vary from application to application. While application settings declared in web.config files will work with Azure App Services, they are considered to be obsolete. Azure App Services provides direct support for application settings through the portal.

This check examines the site to be migrated for application settings declared in web.config files. If found, they will not prevent the site from migrating, but a recommendation will be made to move the settings from the web.config file to the Azure Portal after migration is complete.

### Potential Remediation steps:

* No action required prior to migration.
* Optional recommended steps:
  + After Migration any application settings declared in the web.config file should be configured in the azure portal.
  + See [Configure apps in Azure App Service.](https://docs.microsoft.com/en-us/azure/app-service/web-sites-configure)

# **Connection Strings**

Azure App Services supports a number of different database options, including support for Azure based databases, migration from on-premises databases to Azure hosted databases, or support for on-premises databases through Hybrid Connections.

This setting examines the site being migrated for connection strings in web.config files. If detected, the user will be offered the option of either manually managing database migration to Azure or continuing to use the on-premises database via Hybrid Connections. If the user opts to use Hybrid Connections, the migration tool will automatically set them up and configure them for use.

Note that it is possible for web applications to interact with databases in different ways that do not necessarily use connection strings. While these databases can be supported with the same options as above, they will not be automatically detected during migration.

### Potential Remediation steps:

* Manually migrate database and setup connection or setup hybrid connection.
* For more information see Migrate your MySQL database to Azure Database for MySQL using dump and restore.

##### **Frameworks**

Checks if the site is using frameworks not currently supported for automatic migration. Allows ASP.NET, and PHP. (In future builds ASP.NET Core, NodeJS, Java, and Python will be supported).

Blocks if multiple versions of the same frameworks are found for example: PHP 5.5 and 7.1 are both installed

### Potential Remediation steps:

* Verify the desired frameworks/versions for your web app.
  + If frameworks/versions are not in use, remove them.
  + If non supported frameworks are in use on the system, then your web app then migration is not currently possible with the APP migration tool.

# **Configuration Error**

This error occurs when IIS returns an error when being scanned by the app service migration tool. This typically indicates an error within the configuration for the web app on the host machine

### Potential Remediation steps:

* Review logs on the host machine to determine the errors being encountered by the web app.

# **Virtual Directories**

Azure App Services stores the root content for a site in a fixed home directory **D:\home\site\wwwroot**. When an application is migrated, the content under the site root is copied to the home directory on the Azure App Service worker role machine. If there are virtual directories defined on the site to be migrated, the physical paths will not exist on the Azure App Service machine and attempts to access content in the virtual directories will fail.

This check examines the site to be migrated for virtual directories other than the root of the site. If detected, is will be necessary to reorganize the content so that it's all located under the site root, with no virtual directories defined, before the site can be migrated.

### Potential Remediation steps:

* Remove the identified virtual directories.
* See the below documents for further information.
  + [Understanding IIS Bindings, Websites, Virtual Directories, and lastly Application Pools](https://blogs.technet.microsoft.com/chrad/2010/01/24/understanding-iis-bindings-websites-virtual-directories-and-lastly-application-pools/)
  + [Virtual Directory <virtualDirectory>](https://docs.microsoft.com/en-us/iis/configuration/system.applicationhost/sites/site/application/virtualdirectory)