

# Administrator's Guide for SpeechExec Enterprise App Interface service

## Table of Contents

Change history .....	4
1 Requirements.....	5
2 How to define the location of the central Enterprise configuration repository (SEERoot) .....	6
2.1 During installation.....	6
2.2 Modifying the SEERoot manually after installation .....	6
3 Configuring the web service .....	7
3.1 How to set up the web service .....	7
3.2 How to change the user identity for the application pool.....	9
3.3 How to enable Windows Authentication for the service.....	10
3.4 How to change file upload settings.....	11
3.4.1 Change file upload limits.....	11
3.4.2 Configuring the temporary root folder (optional) .....	11
3.5 Optional: Using HTTPS for SpeechExec Enterprise App Interface .....	12
3.5.1 Configuration in IIS.....	12
4 Endpoint specific configuration .....	13
4.1 /app endpoints for mobile apps .....	13
4.1.1 Authentication .....	13
4.1.1.1 Authentication flow for /app endpoints .....	13
4.1.1.2 Authentication settings for /app endpoints .....	13
4.1.1.2.1 Access token validity period setting .....	13
4.1.1.2.2 Access token cryptographic protection .....	13
4.1.2 Configuring metadata sending to the Enterprise BackEnd server (optional) .....	14
4.1.3 Archive folder handling of /app endpoints.....	15
4.1.3.1 Standard behavior.....	15
4.1.3.2 Customized archive folder behavior .....	15
4.1.3.2.1 Behavior with UseArchiveFolderOfAuthorUserRole.....	15
4.1.3.2.2 Behavior with UseCentralArchiveOfAuthorGroup .....	15
4.1.3.2.3 Behavior with UseCustomArchiveFolderPath.....	16
4.2 /masterdata endpoints for dictation seeding.....	16
4.2.1 Authentication .....	16
4.2.2 Database access .....	16

4.3	/dms endpoints for interfacing with document management systems .....	17
4.3.1	Authentication .....	17
5	Testing the web service .....	19
5.1	Authentication details for testing the web service.....	21
6	Troubleshooting.....	22
6.1	Logging .....	22
6.2	How to set up the application pool of the web service .....	22
6.3	How to create a new application pool.....	22

## Change history

Application version	Document version	Description
SEE 8.0	2022.02.15	Document service-level archive folder override (4.1.3)
	2022.01.01	Initial version

## 1 Requirements

SpeechExec Enterprise App Interface service has the following software requirements:

**Warning:** Please read the full list of requirements before you start.

<b>Operating system</b>	Windows Server 2022, 2019, or 2016	
<b>IIS</b>	IIS 7 or later	Make sure that IIS is turned on before installing .NET Framework to have a properly installed ASP.NET
<b>Microsoft .NET Framework</b>	Microsoft .NET Framework 4.8 or later	
<b>Active Directory</b>	User authentication / authorization	An Active Directory environment is required

### Role and Feature requirements

(example using Windows Server 2016 – Add Roles and Features Wizard):

#### Roles tab:

- Web Server (IIS) [IIS 7 or later]
  - Security
    - Windows Authentication
  - Application Development
    - .NET Extensibility 4.x
    - ASP .NET 4.x
    - Application Initialization

#### Features tab:

- .NET Framework 4.x Features
  - WCF Services
    - HTTP Activation
- Desktop Experience (not enabled by default on older Windows Server installations)

Although the installer of SpeechExec Enterprise App Interface service installs the web service into IIS, it is the task of the administrator to properly set up IIS.

## 2 How to define the location of the central Enterprise configuration repository (SEERoot)

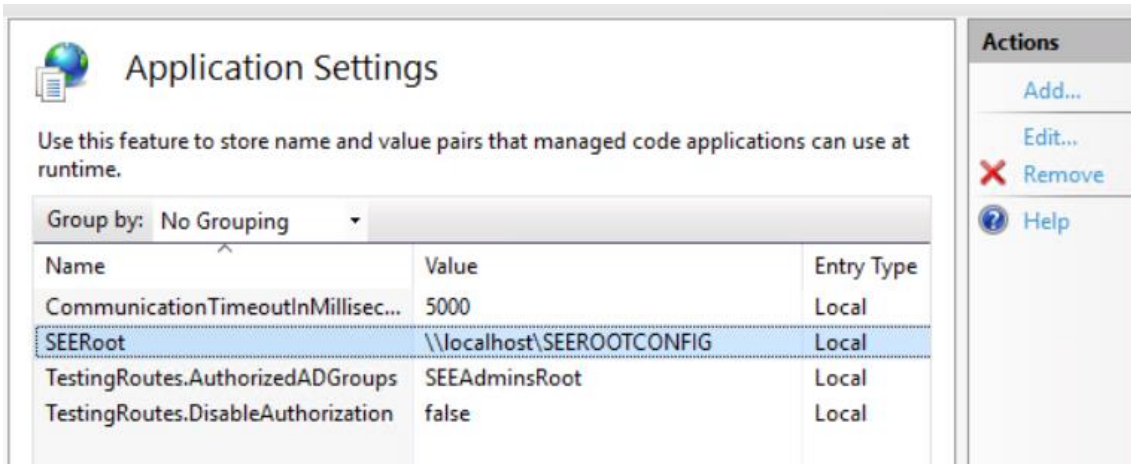
### 2.1 During installation

During installation, the SEERoot configuration folder must be selected and cannot be modified by the installer later, only manual modification is supported.

### 2.2 Modifying the SEERoot manually after installation

If you want to change the path of the SEERoot configuration folder after the installation:

- open the IIS Manager
- select the **SEEAppInterface** node and double click **Application Settings**
- Double click "SEERoot" to change the path of the SEERoot configuration folder:



The screenshot shows the 'Application Settings' window for the 'SEEAppInterface' node in IIS Manager. The window title is 'Application Settings' with a globe icon. Below the title, it says 'Use this feature to store name and value pairs that managed code applications can use at runtime.' There is a 'Group by:' dropdown set to 'No Grouping'. A table lists the settings:

Name	Value	Entry Type
CommunicationTimeoutInMillisec...	5000	Local
<b>SEERoot</b>	<b>\\localhost\\SEEROOTCONFIG</b>	Local
TestingRoutes.AuthorizedADGroups	SEEAdminsRoot	Local
TestingRoutes.DisableAuthorization	false	Local

On the right side, there is an 'Actions' panel with buttons: 'Add...', 'Edit...', 'Remove' (with a red X icon), and 'Help' (with a question mark icon).

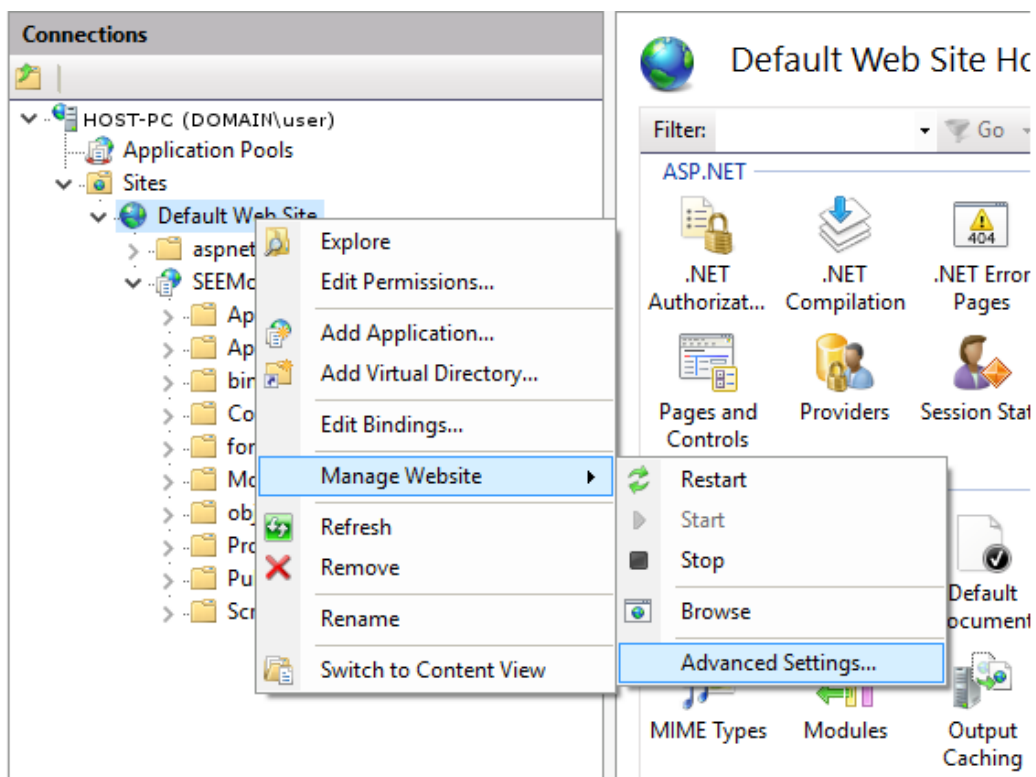
## 3 Configuring the web service

### 3.1 How to set up the web service

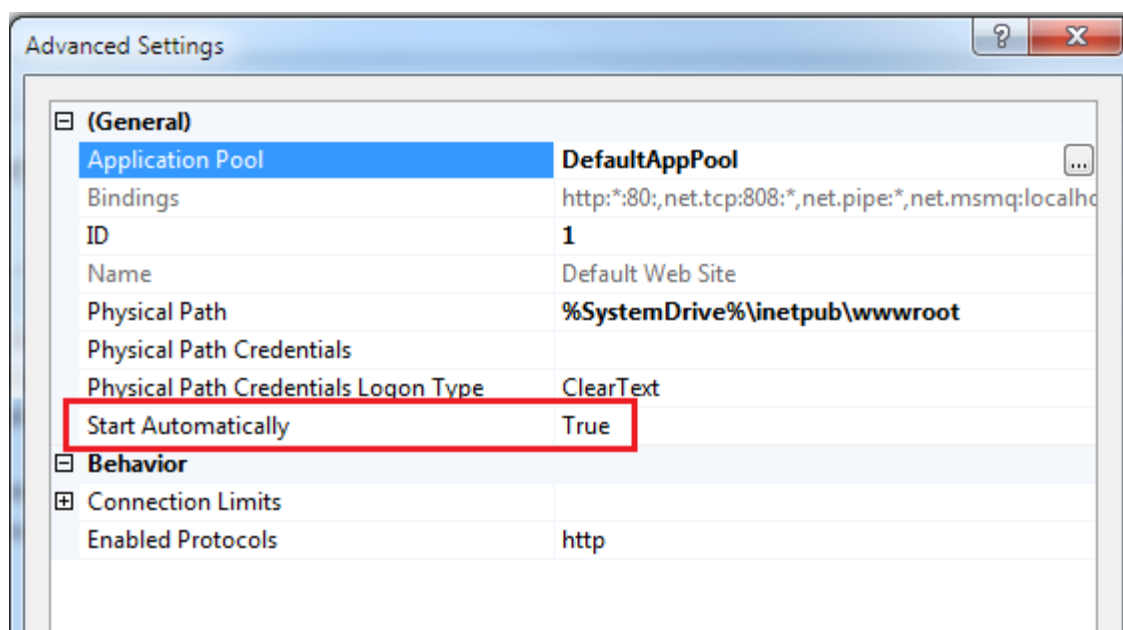
To set up the web service to start automatically, do the following:

#### **Older IIS versions (earlier than version 8):**

1. Right-click on **Default Web Site** and select **Manage Web Site > Advanced settings...**



2. Set **Start Automatically** to **True**.



#### Newer IIS versions (8 or later):

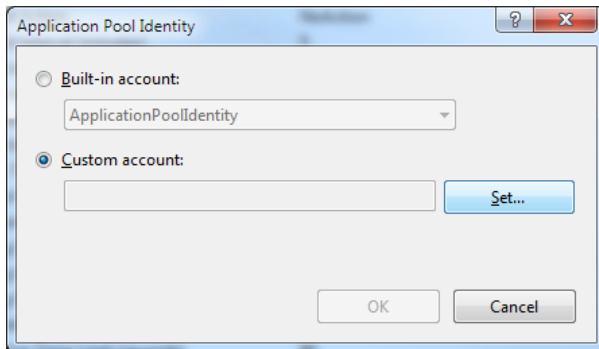
1. In IIS Manager, click the computer name on the **Connection** pane.
2. Switch to **Features View** if the view is not active.
3. Double-click **Configuration Editor** in the **Management** section of the **Features View**.
4. Click the down-arrow for the **Section** field, expand `system.applicationhost`, and then click application pools.
5. Click **(Collection)** and then click ellipses (...) next to the field that shows the count.
6. In the **Collection Editor**, select the application pool for which you want to configure the `startMode` attribute.
7. In the **Properties** window at the bottom, set the following values:
  - o `autoStart` attribute to **true**
  - o `startMode` attribute to **AlwaysRunning**



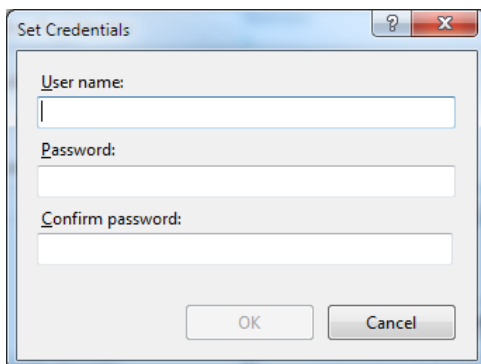
### 3.2 How to change the user identity for the application pool

To change the user identity of the application pool newly created by the application setup, follow the instructions below:

1. On the **Application Pools** panel in IIS, right-click on the **SEWebServicesAppPool** item and select **Advanced Settings...**
2. In the **Advanced Settings** dialog, select **Identity** and click **Browse (...)**.
3. In the **Application Pool Identity** dialog, check **Custom account** and click **Set**.



4. In the **Set Credentials** dialog, enter your user identity credentials and click **OK**.



**NOTE:** The user must have read and write permissions for the **SEERoot** configuration, **Finished dictations** and **Archive** folders)

5. Make sure that the **Load User Profile** setting is set to **True**.

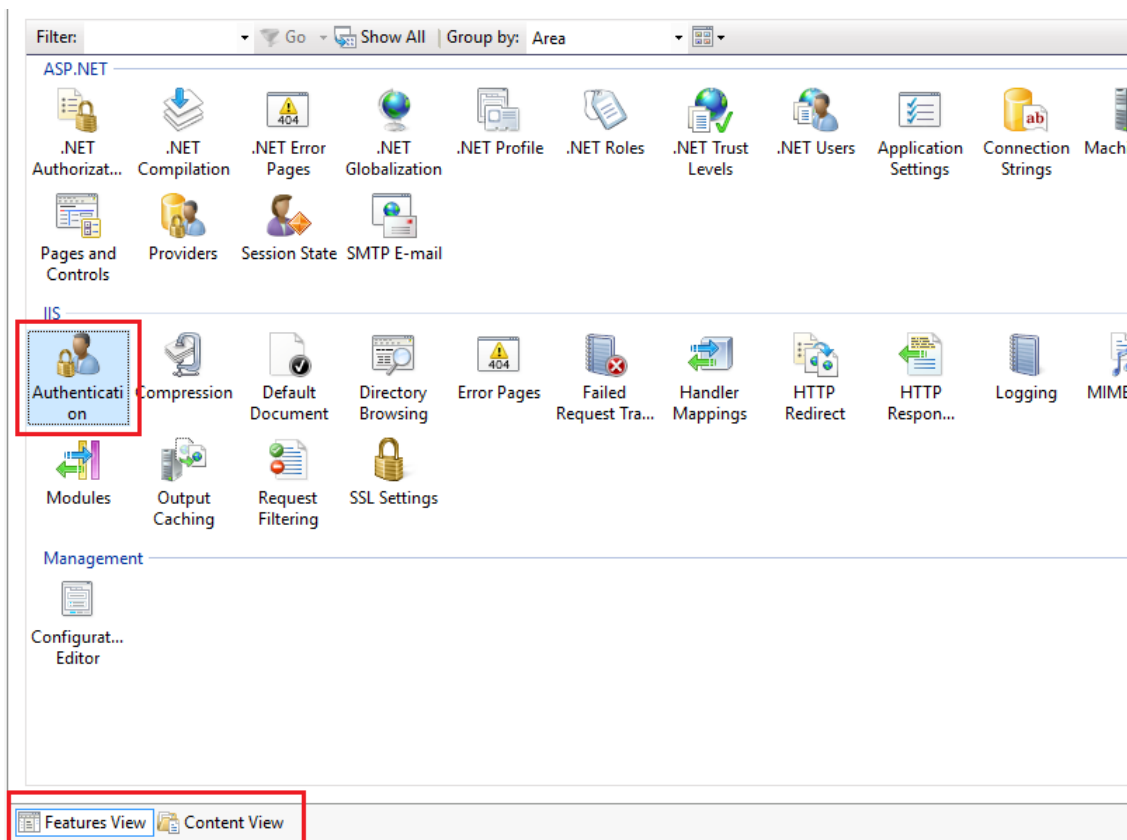
For information on how to create an application pool, see [Optional: How to create a new application pool](#).

### 3.3 How to enable Windows Authentication for the service

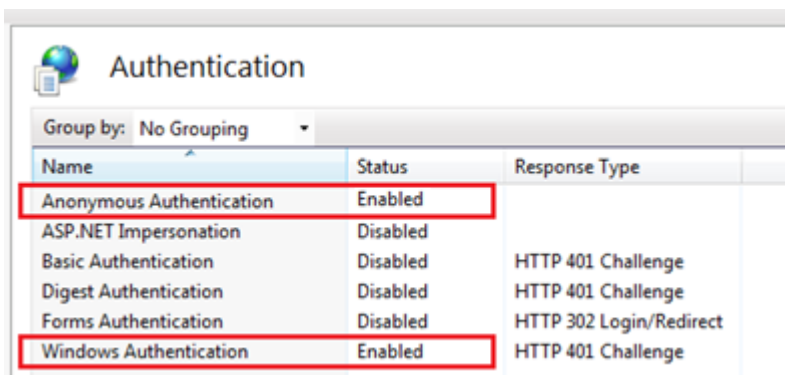
The SEEAppInterface service needs Windows Authentication enabled to accept authenticated requests on the service testing interface (see 4 below).

To enable Windows Authentication, do the following:

1. In the **Connections** panel on the left, select the **Sites > SEEAppInterface** web service.
2. Select the **Features View** at the bottom of the panel.
3. Double-click on **Authentication** in the **IIS** section of the panel.



4. In the **Authentication** panel, do the following:
  - Right-click **Anonymous Authentication** and select **Enabled**.
  - Right-click **Windows Authentication** and select **Enabled**.



## 3.4 How to change file upload settings

### 3.4.1 Change file upload limits

Certain IIS installations have small upload limits, resulting in failing dictation uploads.

Follow the steps below to make sure that dictation files can be uploaded to Enterprise App Interface service without IIS rejecting it.

1. Open IIS Manager and select **"SEEAppInterface"** on the left side.
2. Open **Configuration Editor** located in the middle panel under **Management**.
3. On the top of the window, locate the 2 dropdown lists labeled **Section** and **From**.
4. Set the **Section** dropdown to: **"system.webServer/serverRuntime"**.
5. Set the **From** dropdown to: **"ApplicationHost.config"**.
6. In the available setting list, locate the setting named **"uploadReadAheadSize"**.
7. This value specifies a request limit in **bytes**.  
If the value is lower than **"524288000"** (approx. 500 megabytes), set the value to **"524288000"**, otherwise leave it unchanged.
8. Set the **Section** dropdown to: **"system.webServer/security/requestFiltering"**.
9. Set the **From** dropdown to: **"Default Web Site/ SEEAppInterface Web.config"**.
10. In the available setting list, locate the setting named **"requestLimits"**, expand it, then locate the setting named **"maxAllowedContentLength"**.
11. This value specifies the max. size of a request in **bytes**.  
If the value is lower than **"524288000"** (approx. 500 megabytes), set the value to **"524288000"**, otherwise leave it unchanged.
12. Set the **Section** dropdown to: **"system.web/httpRuntime"**.
13. Set the **From** dropdown to: **"Default Web Site/ SEEAppInterface Web.config"**.
14. In the available setting list, locate the setting named **"maxRequestLength"**.
15. This value specifies the max. size of a request in **kilobytes**.  
If the value is lower than **"512000"** (approx. 500 megabytes), set the value to **"512000"**, otherwise leave it unchanged.
16. Press **Apply** in the top right corner.
17. Recycle the application pool hosting **"SEEAppInterface"** (usually SEEServicesAppPool).

### 3.4.2 Configuring the temporary root folder (optional)

When uploading a dictation from the Philips Voice Recorder to the Enterprise App Interface service, the temporary files used during the upload are stored in a temporary root folder. The place of the temporary root folder can be customized with the following settings:

- **TempRootFolderType**: Choose from the following values:
  - **UserProfile**: The temp root folder path is 'the user's profile folder' (regardless of the **CustomTempRootFolderPath** value).  
This may vary depending on the version of your Windows. On Windows 10, for

example: C:\Users\<username>\SEE\_INTAPI

This is the default setting.

- **ProgramData:** The temp root folder path is the 'program data folder' (regardless of the **CustomTempRootFolderPath** value).

This may vary depending on the version of your Windows. On Windows 10, for

example: C:\ProgramData\SEE\_INTAPI

- **Custom:** The temp root folder path is the folder path defined in the **CustomTempRootFolderPath** setting.
- **CustomTempRootFolderPath:** Define the custom path of the temporary root folder.

If no value is defined for the temp root folder settings, the **UserProfile** default value is used.

### 3.5 Optional: Using HTTPS for SpeechExec Enterprise App Interface

To use SpeechExec Enterprise App Interface with HTTPS, you need to set the following settings:

#### 3.5.1 Configuration in IIS

- In ISS Manager select the **Default Web Site** (or the one containing SEEAppInterface) node on the **Connections** panel.
- On the **Actions** panel (on the right) select **Bindings...**
- Click **Add**.
- Select **https** from the **Type** combo box.
- Select an SSL certificate from the **SSL certificate** combo box.
- Click **OK**.
- In IIS manager, select the **SEEAppInterface** on the left side.
- Open **SSL settings**.
- Make sure **Client certificates** is set to **Ignore**.
- To properly apply the new settings, a host computer restart is recommended.

## 4 Endpoint specific configuration

### 4.1 /app endpoints for mobile apps

#### 4.1.1 Authentication

##### 4.1.1.1 Authentication flow for /app endpoints

The /app endpoints provide features mainly for apps running on mobile devices. The API calls coming from a mobile device must be authenticated. To avoid transferring the username and password with each API request, the /app endpoints use an access token-based authentication mechanism (OAuth Authorization Code Flow).

Main steps of this mechanism:

- Client (mobile app) must call the /app/token endpoint and specify the end-user's login credentials
- The service tries to authenticate the user
- If successfully authenticated, the server returns an access token with limited lifetime to the client
- The client must use this access token for subsequent API calls

##### 4.1.1.2 Authentication settings for /app endpoints

###### 4.1.1.2.1 Access token validity period setting

The validity period of the access tokens issued by the service is controlled by the following web.config setting:

- AccessTokenLifetimeInMinutes

The default installed value is 1560 minutes.

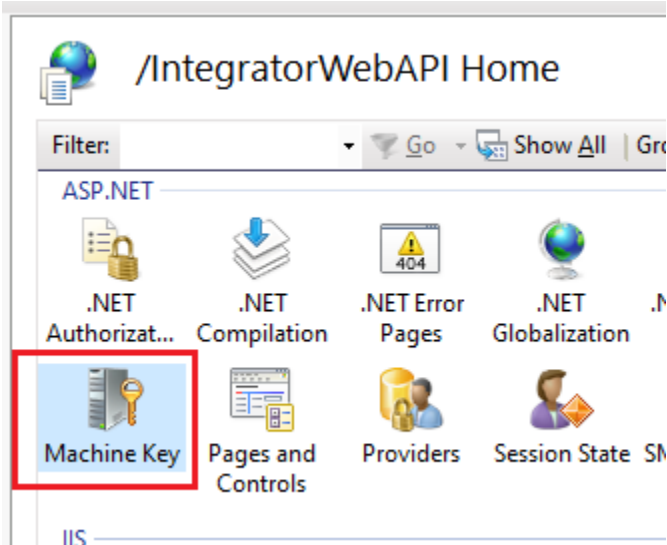
###### 4.1.1.2.2 Access token cryptographic protection

The access tokens issued by the service are encrypted and validated with a cryptographic key pair (validation key and decryption key) called Machine Key. Even though this setting is called Machine Key, in reality a service specific key pair can be set for each web service hosted by a given IIS installation.

This key pair is stored in the web.config file of the service:

```
<system.web>  
  <machineKey ...  
</system.web>
```

The service comes with a pre-installed, generic key pair. It is HIGHLY recommended to re-generate the key pair after installation! It is recommended to use the Machine Key feature of the web service:



**Note:**

If you deploy your application in a web farm, make sure that the configuration files on each server in the web farm have the same value for the validation key and decryption keys, which are used for hashing and decryption respectively. Otherwise, you cannot guarantee which server handles successive requests.

Detailed information on IIS machine keys can be found here:

[https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-r2-and-2012/hh831711\(v=ws.11\)](https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2012-r2-and-2012/hh831711(v=ws.11))

#### 4.1.2 Configuring metadata sending to the Enterprise BackEnd server (optional)

Dictation metadata can be sent to the Enterprise BackEnd (Statistics) server right after a dictation has been uploaded to the Enterprise App Interface service. You can define the connection details of the BackEnd server with the following settings:

- `BackEndServerName`  
Define the name of the BackEnd server where the metadata should be sent to.
- `BackEndServerPort`  
Define the port used by the BackEnd server.  
If this value cannot be parsed as a whole number, the "49255" default value is used.

If the specified server name and port combination is correct, metadata is transmitted to the BackEnd server at each dictation upload. Please note, that the transmitted metadata may not be visible in the BackEnd database immediately after the upload.

### 4.1.3 Archive folder handling of /app endpoints

#### 4.1.3.1 Standard behavior

By default, the physical location of the Archive folder used when querying / searching for dictation files for a given logged on user is determined as follows:

- Try to locate the user profile XML file of the logged-on user
- Read the settings of the Author role
- Use the path of the 'Archive' folder defined for the user

#### 4.1.3.2 Customized archive folder behavior

To support special use-cases, the standard archive folder usage mode can be fine-tuned using the following settings available in the web.config file of the web service:

Setting name	Description
<b>ArchiveFolderUsageMode</b>	<p>This setting value controls how the /app endpoints determine the physical location of the Archive folder for a given user.</p> <p>Supported values:            UseArchiveFolderOfAuthorUserRole            UseCentralArchiveOfAuthorGroup            UseCustomArchiveFolderPath</p>
<b>CustomArchiveFolderPath</b>	<p>A fully specified, custom path used as Archive <i>for ALL users</i> if ArchiveFolderUsageMode is set to UseCustomArchiveFolderPath.</p> <p>Example:  <a href="#">\\FILESERVER\\SEE_ARCHIVE</a></p>

##### 4.1.3.2.1 Behavior with UseArchiveFolderOfAuthorUserRole

Using this setting value results in the standard behavior described in 4.1.3.1 above.

In this case the value specified for CustomArchiveFolderPath setting is ignored.

##### 4.1.3.2.2 Behavior with UseCentralArchiveOfAuthorGroup

Using this setting value, the web service determines the Archive folder location as follows:

- Determine the Enterprise group the logged-on user belongs to
- From this group, try to take the value of the %SE\_CENTRAL\_ARCHIVE\_01% system variable
- If this value is specified AND resolves to a syntactically valid folder path, use that
- Otherwise, fall back to the standard behavior and use the Archive of the author user

In this case the value specified for CustomArchiveFolderPath setting is ignored.

To change the value of %SE\_CENTRAL\_ARCHIVE\_01% variable in Enterprise Manager:

- Open 'System Configuration Center'
- Locate and select the Enterprise group to configure
- Click on 'Group settings...'
- Select the 'Author' role to configure
- Select node 'Enterprise | System variables'
- Set a value for the %SE\_CENTRAL\_ARCHIVE\_01% system variable
- Save changes

#### 4.1.3.2.3 Behavior with UseCustomArchiveFolderPath

Using this setting value, the web service uses the path specification defined by the CustomArchiveFolderPath setting as Archive folder location *for all logged on service users, bypassing any user or group-specific Archive folder setting.*

However, if CustomArchiveFolderPath is not specified or does not resolve to a syntactically valid folder path, the service falls back to the standard behavior and uses the Archive of the author user.

## 4.2 /masterdata endpoints for dictation seeding

### 4.2.1 Authentication

These endpoints were designed for machine-to-machine interaction between software services. A caller service can send HTTP REST requests and receive HTTP responses.

Each HTTP request sent to the /masterdata endpoints is required to have a special HTTP header value.

- Header key/ID:  
"x-sps-api-key"
- Header value:  
"API\_KEY\_STRING"

The API keys accepted by the /masterdata endpoints must be specified in the following web.config setting:

- `API.MasterData.AllowedAPIKeysPipeSeparated`

After installation, this value is empty. This setting allows specifying multiple API keys, individual values must be separated by a pipe ( | ) character.

It is recommended to:

- use a globally unique identifier, like a GUID value for API key
- issue a dedicated API key for each caller software component

### 4.2.2 Database access

Using the /masterdata endpoints it is possible to store initial dictation property values (seeds) for dictations created later. These initial values can be used by end-user SpeechExec applications (utilizing the Master Data feature of Enterprise Configuration Service) when creating new dictations.



These initial values are stored in a Microsoft SQL Server database. The creation and maintenance of the Master Data database, and the required database tables / views is the responsibility of the administrator. Sample SQL scripts can be found on the installation/distribution media of Enterprise Configuration Service ("01\_Foundation\03\_Enterprise\_Manager\Tools\SEE ConfigurationService for IIS\Samples" folder)

The /masterdata endpoints of Enterprise App Interface service **require** the presence of a **database table** with the following name and proper table structure

- Required name for the table: **MasterDataItemsForSpeechExecEnterprise**

The following settings, stored in web.config, control how Enterprise App Interface service tries to connect to the database:

Setting name	Description	Example
<b>MasterData.MSSQL.Server</b>	The name of the server running the MSSQL server	myserver01
<b>MasterData.MSSQL.Database</b>	The name of the database where the "MasterDataItemsForSpeechExecEnterprise" view is located	mydatabase
<b>MasterData.MSSQL.UseSQLAuthentication</b>	True if using username & password-based authentication  False if using Windows authentication	true
<b>MasterData.MSSQL.SQLOAuthentication.Username</b>	SQL authentication username (when using SQL authentication) as a Base64 encoded value	c3FsdXNlcjE= non-encoded value: sqluser1
<b>MasterData.MSSQL.SQLOAuthentication.Password</b>	SQL authentication password (when using SQL authentication) as a Base64 encoded value	cEBzc3cwcmQ= Non-encoded value: p@ssw0rd

## 4.3 /dms endpoints for interfacing with document management systems

### 4.3.1 Authentication

These endpoints were designed for machine-to-machine interaction between software services. A caller service can send HTTP REST requests and receive HTTP responses.

Each HTTP request sent to the /dms endpoints is required to have a special HTTP header value.

- Header key/ID:  
"x-sps-api-key"
- Header value:  
"API\_KEY\_STRING"

The API keys accepted by the /dms endpoints must be specified in the following web.config setting:

- `API.DMS.AllowedAPIKeysPipeSeparated`

After installation, this value is empty. This setting allows specifying multiple API keys, individual values must be separated by a pipe ( | ) character.

It is recommended to:

- use a globally unique identifier, like a GUID value for API key
- issue a dedicated API key for each caller software component

## 5 Testing the web service

Enterprise App Interface service provides a dedicated testing interface:

`<url_of_web_service>/test/testconfig`

For example, if the web service is running on the local computer, the URL looks like the following:

<http://localhost/SEEAppInterface/test/testconfig>

To limit user access to the URL, opening this URL by default requires the following:

- an authenticated Active Directory user session on the calling (browser) side
- the calling user must be the member of the `SEEAdminsRoot` Active Directory security group

The test process validates the server configuration settings and returns a JSON array of validation steps. Each validation step consists of a `StepID`, a `StepResult` and a `StepExplanation`.

Returned response codes can be the following:

- 401 (Unauthorized), if the current user cannot be authenticated by the web service
- 566, if any of the validation steps failed
- 200 (OK), if all validation steps passed

**Please make sure that the Enterprise App Interface service is 'Enabled' and has a compatible License Server configured in Enterprise Manager before attempting a test!**

To turn on Enterprise App Interface service:

- Open **Enterprise Manager** -> **System Administration** -> **Groups and users** -> **Mobile service settings** (panel) and enable the service.
- Add a supported License Server (with a compatible license loaded) in the **License Settings...** window (accessible from the same panel).

*If the above conditions are not met, the test will fail at "020\_SEELicenseServerAddressFoundInConfig".*

Validation steps:

StepID	Explanation
001_SEERootFoundInConfig	SEERoot value must be found in web.config
005_SEERootExists	The folder specified by the SEERoot value exists.
010_SEERootStructureCorrect	The service must be able to read from and write to SEERoot.
020_SEELicenseServerAddressFoundInConfig	License server address found in SEERoot config
025_SEELicenseServerPortFoundInConfig	License server port found in SEERoot config
030_SEELicenseServerCanConnect	Connection to License server is available with the given address and port in the SEERoot configuration
035_TempRootFolderAccessible	The specified temp root folder is accessible (all types of temp root folder are tested)

Please note that the response content is always returned in English.

Example response result when the 1<sup>st</sup> step passed, but the 2<sup>nd</sup> step failed:

```
[
  {
    "StepID": "001_SEERootFoundInConfig",
    "StepResult": "SUCCESS",
    "StepDescription": "SEERoot value is found in web.config"
  },
  {
    "StepID": "005_SEERootExists",
    "StepResult": "FAIL",
    "StepDescription": "The folder specified by the SEERoot value does not exist"
  }
]
```

Example response result when all steps are passed:

```
[
  {
    "StepID": "001_SEERootFoundInConfig",
    "StepResult": "SUCCESS",
    "StepDescription": "SEERoot value is found in web.config"
  },
  {
    "StepID": "005_SEERootExists",
    "StepResult": "SUCCESS",
    "StepDescription": "The folder specified by the SEERoot value exists (D:\\seeroot)"
  },
  {
    "StepID": "010_SEERootStructureCorrect",
    "StepResult": "SUCCESS",
    "StepDescription": "The structure of SEERoot is correct (it has all required sub-folders and required .config files)"
  },
  {
    "StepID": "020_SEELicenseServerAddressFoundInConfig",
    "StepResult": "SUCCESS",
    "StepDescription": "Valid License Server address in configuration"
  },
  {
    "StepID": "025_SEELicenseServerPortFoundInConfig",
    "StepResult": "SUCCESS",
    "StepDescription": "Valid License Server port in configuration"
  },
  {
    "StepID": "030_SEELicenseServerCanConnect",
    "StepResult": "SUCCESS",
    "StepDescription": "License Server connection successful"
  }
]
```

```
{
  "StepID": "035_TempRootFolderAccessible",
  "StepResult": "SUCCESS",
  "StepDescription": "UserProfile: The specified temp root folder is accessible
(C:\Users\testuser\SEE_INTAPI); ProgramData: The specified temp root folder is
accessible (C:\ProgramData\SEE_INTAPI); Custom: The specified temp root folder is
accessible (d:\work\MyTestFolder); "
}
]
```

## 5.1 Authentication details for testing the web service

**Important:** The /test/testconfig URL should ***only be accessible for administrators***, and only for testing purposes!

For diagnostics purposes, access control can be re-configured using the following web.config settings:

```
<add key="TestingRoutes.AuthorizedADGroups"
      value="%GROUPLIST%" />
<add key="TestingRoutes.DisableAuthorization"
      value="false" />
```

By setting the value of `TestingRoutes.DisableAuthorization` to "true", access to the /test/testconfig URL becomes **totally unrestricted**, i.e., **ANY** user can call it **WITHOUT authentication**.

Access to the /test/testconfig URL can be restricted to members of certain Active Directory security groups by listing the allowed groups in the value of `TestingRoutes.AuthorizedADGroups`.

Multiple groups can be specified by separating the Active Directory group names with a comma (,).

## 6 Troubleshooting

### 6.1 Logging

The name of the log configuration file is **SpeechExecLog.config**. It is in the root folder of the **Enterprise App Interface** service (visible in **Content View**).

It is the IIS administrator's responsibility to manually edit the SpeechExecLog.config file and specify correct configuration values.

The default path of the log file is:

```
<param name="File" value="c:/SEEAppInterfaceLogFolder/SEEAppInterface.log" />
```

The default maximum size of the log file is:

```
<param name="MaximumFileSize" value="1000MB" />
```

### 6.2 How to set up the application pool of the web service

1. In the **Connections** panel on the left, find the **Sites > SEEAppInterface** web service.
2. Right-click on the web service, and click **Manage Application > Advanced settings...**
3. In the **Advanced Settings** dialog, select **Application Pool** and click on **Browse (...)**.
4. In the **Select Application Pool** dialog, select **SEWebServicesAppPool**, and click **OK**.

### 6.3 How to create a new application pool

Application pools allow isolating one web application from another, even if they are running on the same server. This way, if there is an error in one app, it will not take down other applications.

Additionally, application pools allow specifying different levels of security (for example, file access security) for different apps.

The installer of the web service will, by default, create a new application pool and assign the web service to the pool.

If a new application pool must be created, follow the instructions below:

1. Open the Internet Information Services (IIS) Manager.
2. Select **Application Pools** from the **Connections** panel on the left. Right-click on the **Application Pools** panel and select **Add Application Pool...**
3. Enter a name for your new application pool, such as **SEWebServicesAppPool**.
4. In the **.NET CLR version** list, select :
  - .NET CLR Version v4.0.30319
5. Make sure the **Start application pool immediately** checkbox is selected.
6. Click **OK** to create and start the application pool.