

Administrator's Guide for SpeechExec Enterprise App Interface service

Table of Contents

| | |
|---|----|
| Change history | 4 |
| 1 System requirements..... | 5 |
| 1.1 Hardware requirements..... | 5 |
| 1.2 Software requirements | 5 |
| 2 How to define the location of the central Enterprise configuration repository (SEERoot) | 7 |
| 2.1 During installation..... | 7 |
| 2.2 Modifying the SEERoot manually after installation | 7 |
| 3 Configuring the web service | 8 |
| 3.1 Set automatic start of the web service | 8 |
| 3.2 Proper application pool settings | 10 |
| 3.2.1 Change user identity of app pool | 10 |
| 3.2.2 Local login with user identity of app pool | 10 |
| 3.2.3 Importance of 'Load user profile' app pool setting | 11 |
| 3.2.4 Required file access..... | 11 |
| 3.2.5 Special application pool settings | 11 |
| 3.3 How to enable Windows Authentication for the service..... | 12 |
| 3.4 How to change file upload settings..... | 13 |
| 3.4.1 Change file upload limits | 13 |
| 3.4.2 Configuring the temporary root folder (optional) | 13 |
| 3.5 Using HTTPS for SpeechExec Enterprise App Interface | 14 |
| 3.5.1 Configuration in IIS..... | 14 |
| 3.6 Using own Machine Key for /app endpoints | 14 |
| 3.6.1 Overview | 14 |
| 3.6.2 Web farm scenario considerations | 15 |
| 4 Endpoint specific configuration | 16 |
| 4.1 /app endpoints for mobile apps | 16 |
| 4.1.1 Authentication | 16 |
| 4.1.1.1 Authentication flow for /app endpoints..... | 16 |
| 4.1.1.2 Authentication settings for /app endpoints..... | 16 |
| 4.1.1.2.1 Access token validity period setting | 16 |
| 4.1.1.2.2 Access token cryptographic protection..... | 16 |
| 4.1.2 Configuring metadata sending to the Enterprise BackEnd server (optional) | 16 |

| | | |
|-----------|---|----|
| 4.1.3 | Archive folder handling of /app endpoints | 17 |
| 4.1.3.1 | Standard behavior | 17 |
| 4.1.3.2 | Customized archive folder behavior..... | 17 |
| 4.1.3.2.1 | Behavior with UseArchiveFolderOfAuthorUserRole | 17 |
| 4.1.3.2.2 | Behavior with UseCentralArchiveOfAuthorGroup..... | 17 |
| 4.1.3.2.3 | Behavior with UseCustomArchiveFolderPath | 18 |
| 4.1.4 | Sending dictations for SpeechKit speech recognition..... | 18 |
| 4.1.4.1 | Set SpeechKit settings in Enterprise Manager | 18 |
| 4.1.4.2 | Transfer uploaded dictations to the Enterprise Speech Recognition server's input folder | 18 |
| 4.1.5 | Running the service in "DMZ" environment | 19 |
| 4.1.5.1 | Configuring the service to run in "DMZ" mode..... | 19 |
| 4.1.5.2 | Resolving folder paths for dictations..... | 19 |
| 4.2 | /masterdata endpoints for dictation seeding..... | 19 |
| 4.2.1 | Authentication | 19 |
| 4.2.2 | Database access | 20 |
| 4.3 | /dms endpoints for interfacing with document management systems | 21 |
| 4.3.1 | Authentication | 21 |
| 5 | Testing the web service | 22 |
| 5.1 | Overview | 22 |
| 5.2 | Authentication details for testing the web service..... | 22 |
| 5.3 | Test process | 23 |
| 5.4 | Response examples..... | 23 |
| 6 | Troubleshooting..... | 26 |
| 6.1 | Logging | 26 |
| 6.2 | How to assign an application pool to the web service | 26 |
| 6.3 | How to create a new application pool..... | 26 |

Change history

| Application version | Document version | Description |
|---------------------|------------------|---|
| SEE 9.0 | 2024.06.11 | SEE 9.0 release Web service RunAs user must be a local admin 32-bit app. pool settings must be disabled |
| SEE 8.8 | 2024.02.14 | SEE 8.8 release |
| SEE 8.7 | 2023.06.15 | Added server time items to <code>/test/testconfig</code> response example SEE 8.7 release |
| SEE 8.1 | 2022.08.29 | Explain server-side SpeechKit configuration |
| | 2022.06.14 | Explain importance of 'Load user profile' app pool setting and local login Specify needed file access |
| SEE 8.0 | 2022.04.13 | HTTPS as highly recommended |
| | 2022.02.15 | Document service-level archive folder override (4.1.3) |
| | 2022.01.01 | Initial version |

1 System requirements

1.1 Hardware requirements

SpeechExec Enterprise App Interface service has the following hardware requirements:

- **Processor:** Intel® Core™ i5 or equivalent AMD processor
- **RAM:** 4 GB
- **Free hard disk space:**
 - 4 GB hard disk space (8 GB recommended)
 - Microsoft .NET 4.8 Framework requires an additional 4.5 GB of free disk space.

1.2 Software requirements

For the supported operating system, we recommend that you apply the latest Service Pack available before installing the SpeechExec Enterprise App Interface service.

- Microsoft Windows Server 2016
- Microsoft Windows Server 2019
- Microsoft Windows Server 2022
- Internet Information Services (IIS) 7 or later, with the following features:
 - ASP.NET 4.x
 - Windows authentication
- Microsoft .NET Framework 4.8
 - Make sure that IIS is turned on before installing .NET Framework to have a properly installed ASP.NET
- Active Directory: An Active Directory environment is required for user authentication/authorization

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)

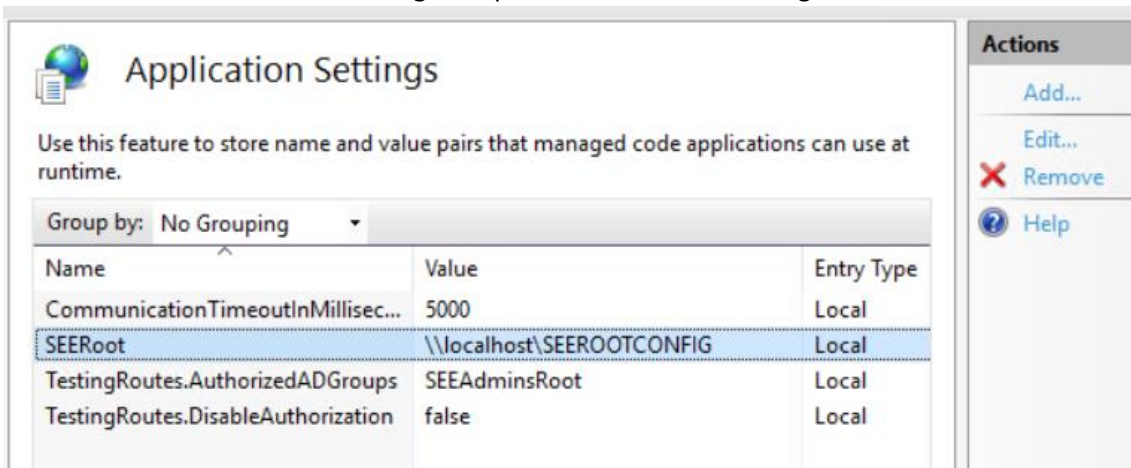
2.1 During installation

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 on **Application Settings**
- Double-click on "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 several settings, with 'SEERoot' selected and highlighted in blue. The 'SEERoot' entry has a value of '\\localhost\\SEEROOTCONFIG' and an 'Entry Type' of 'Local'. Other entries include 'CommunicationTimeoutInMillisec...' with value '5000', 'TestingRoutes.AuthorizedADGroups' with value 'SEEAdminsRoot', and 'TestingRoutes.DisableAuthorization' with value 'false'. On the right side, there is an 'Actions' panel with buttons for 'Add...', 'Edit...', 'Remove' (with a red X icon), and 'Help' (with a question mark icon).

| Name | Value | Entry Type |
|------------------------------------|-----------------------------------|--------------|
| CommunicationTimeoutInMillisec... | 5000 | Local |
| SEERoot | \\localhost\\SEEROOTCONFIG | Local |
| TestingRoutes.AuthorizedADGroups | SEEAdminsRoot | Local |
| TestingRoutes.DisableAuthorization | false | Local |

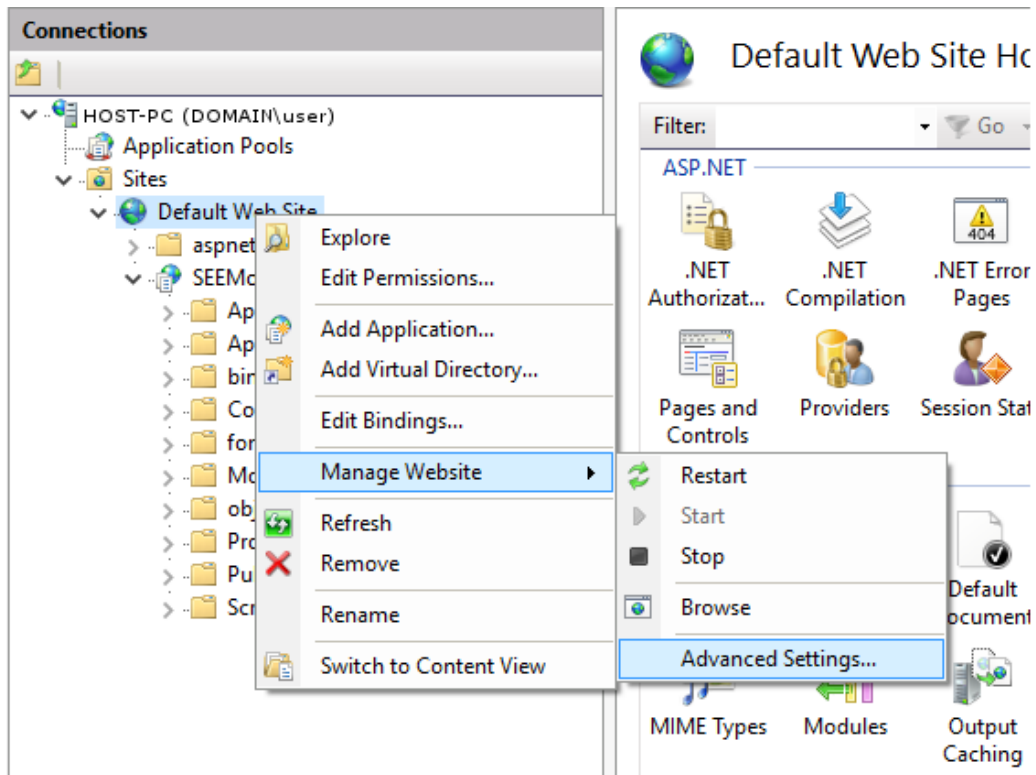
3 Configuring the web service

3.1 Set automatic start of the web service

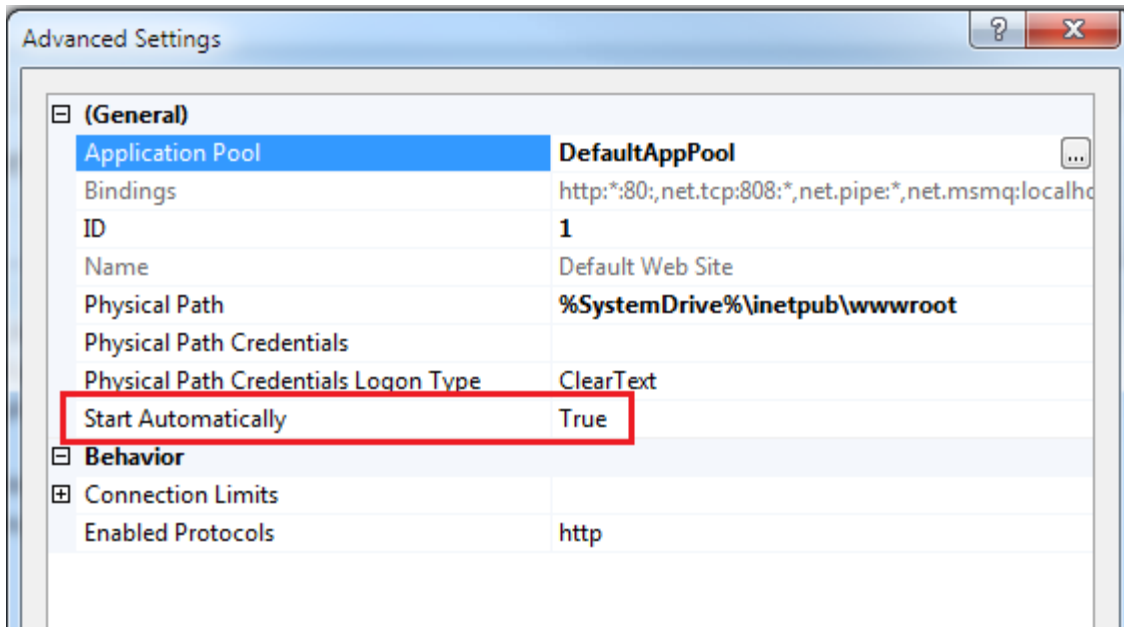
To set up the web service to start automatically, do the following:

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 on the computer name on the **Connection** pane.
2. Switch to **Features View** if the view is not active.
3. Double-click on **Configuration Editor** in the **Management** section of the **Features View**.
4. Click on the down-arrow for the **Section** field, expand `system.applicationhost`, and then click on the application pools.
5. Click on **(Collection)** and then on the 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**

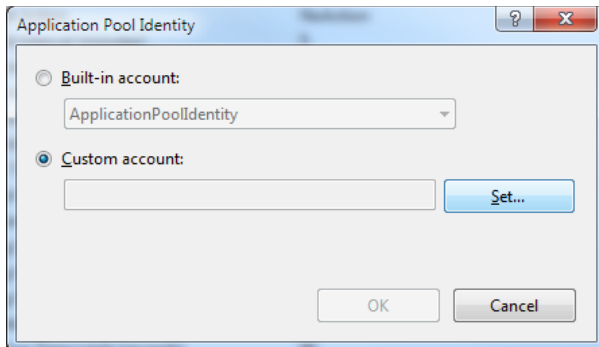
3.2 Proper application pool settings

3.2.1 Change user identity of app pool

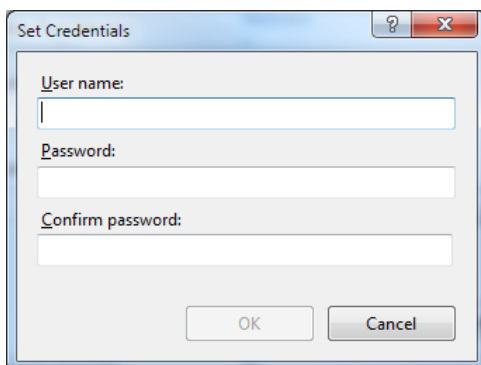
Important! The user identity specified below must have local administrator privileges!

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 on the **Browse** button (...).
3. In the **Application Pool Identity** dialog, check **Custom account** and click on the **Set** button.



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



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](#).

3.2.2 Local login with user identity of app pool

On the host server machine, it is recommended to log in at least once with the domain user whose identity is used as the app pool identity. Otherwise, IIS might not be able to create the local profile of the domain user and starting the web service might fail.

3.2.3 Importance of 'Load user profile' app pool setting

Enterprise App Interface requires that the app pool setting '**Load user profile**' is set to true. This setting ensures that the Windows user profile that belongs to the user identity of the app pool user is loaded. This includes loading the cryptographic store, environment variables such as %TEMP%, etc., which are necessary for the web service.

3.2.4 Required file access

The web service must have read and write access to the following folders:

- The %TEMP% folder
- SEE_INTAPI folder, see 3.4.2 below
- The folders where end-user dictations are stored

The web service must have read access to the following folders:

- The SEERoot central configuration repository

3.2.5 Special application pool settings

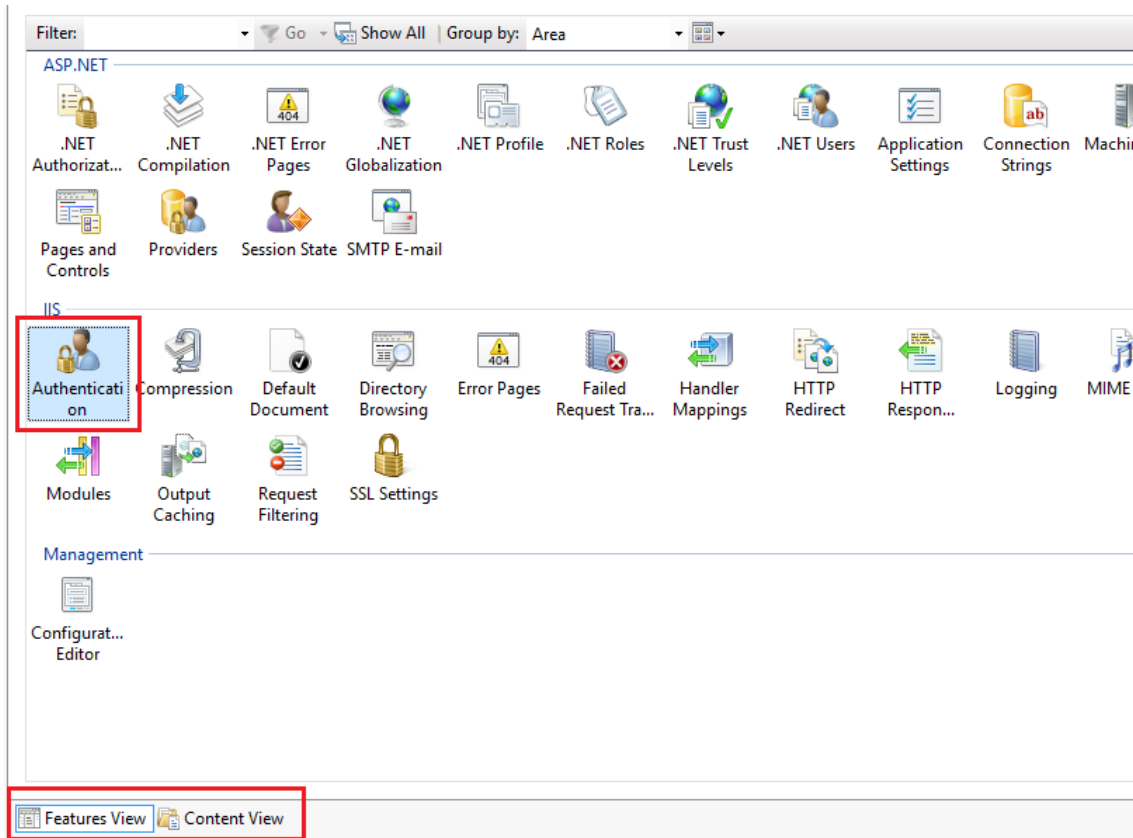
- Make sure that the **Enable 32-bit applications** setting of the application pool is:
 - o False

3.3 How to enable Windows Authentication for the service

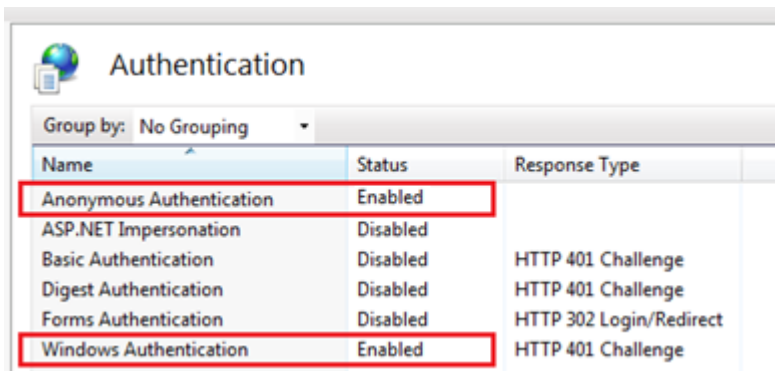
The SEEAppInterface service needs Windows Authentication enabled to accept authenticated requests on the service *testing* interface (see 5 below).

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 on **Anonymous Authentication** and select **Enabled**.
 - Right-click on **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 limit defaults, 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. In the top of the window, locate the two 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 temporary 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 temporary 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 temporary 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 temporary root folder settings, the **UserProfile** default value is used.

3.5 Using HTTPS for SpeechExec Enterprise App Interface

The programmatic endpoints of SpeechExec Enterprise App Interface can be accessed via non-encrypted, plaintext HTTP or encrypted HTTPS communication channels. While technically it is possible to use non-encrypted HTTP, it is highly recommended to use HTTPS for security and privacy reasons.

Important! If the services offered by SpeechExec Enterprise App Interface are accessed from the Philips SpeechLive app running on Android or iOS, using HTTPS is **not optional**: it is required to configure HTTPS access, as these mobile apps can only connect to HTTPS-secured endpoints.

3.5.1 Configuration in IIS

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

- In ISS Manager select the **Default Web Site** (or the one containing SEAppInterface) node on the **Connections** panel.
- On the **Actions** panel (on the right) select **Bindings...**
- Click on the **Add** button.
- Select **https** from the **Type** combo box.
- Select an SSL certificate from the **SSL certificate** combo box.
- Click on the **OK** button.
- In IIS manager, select the **SEAppInterface** 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.

3.6 Using own Machine Key for /app endpoints

3.6.1 Overview

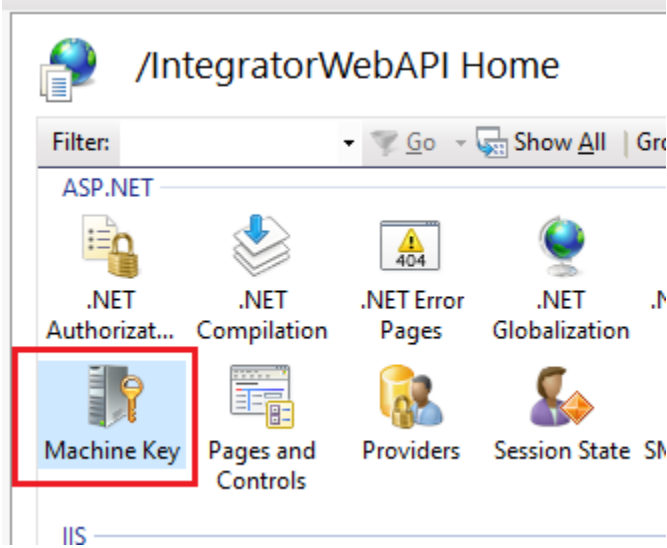
The /app endpoints of Enterprise App Interface use access token-based authentication and authorization. For example, these endpoints are called by the Philips SpeechLive mobile apps running on Android and iOS.

The secure 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 generate a new key pair after installation! It is recommended to use the "Machine Key" feature of IIS:



3.6.2 Web farm scenario considerations

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.

For more information on IIS machine keys, see:

[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 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

Access token encryption configuration is described in 3.6 above

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-in user is determined as follows:

- Try to locate the user profile XML file of the logged-in 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 |
|--------------------------------|--|
| ArchiveFolderUsageMode | <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> |
| CustomArchiveFolderPath | <p>A fully specified, custom path used as Archive <i>for ALL users</i> if ArchiveFolderUsageMode is set to UseCustomArchiveFolderPath.</p> <p>Example: \\FILESERVER\\SEE_ARCHIVE</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 the CustomArchiveFolderPath setting is ignored.

4.1.3.2.2 Behavior with UseCentralArchiveOfAuthorGroup

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

- Determine the Enterprise group the logged-in user belongs to
- From this group, 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 folder of the author user

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

To change the value of the %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 the 'Enterprise | System variables' node
- 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 folder of the author user.

4.1.4 Sending dictations for SpeechKit speech recognition

When the SpeechLive mobile application fills out the optional metadata field SpeechKitSRLanguageAndTopicID during dictation upload, the uploaded dictation is almost ready for back-end speech recognition using the SpeechKit recognition engine. On the server side, some special configuration is needed to complete this workflow.

4.1.4.1 Set SpeechKit settings in Enterprise Manager

In the System Administration of Enterprise Manager, the desired group settings must be defined for the author role. The following options must be set:

- Speech Recognition / General configuration page:
Allow using SpeechKit speech recognition
- Speech Recognition / Interact with recognition server page:
Enable dictation transfer to the Speech Recognition Server
- Speech Recognition / Language & topics page:
at least one recognition language & topic must be added and set to *Allowed*. If you added more than one recognition language & topic item, only those ones will be selectable in the SpeechLive mobile application which are set to *Allowed*.

Please note that setting values specified on group level must be **locked** to override modifying the values on user level. Another solution is to set the required settings for each author user one-by-one. An exception is the list of recognition language & topics, as these can only be specified in the group configuration level.

4.1.4.2 Transfer uploaded dictations to the Enterprise Speech Recognition server's input folder

Since version 9.0 when the optional metadata field SpeechKitSRLanguageAndTopicID is filled out during upload, Enterprise App Interface places the dictation into the folder specified in Enterprise Manager on the Speech Recognition / Interact with recognition server page:

- the Author's *Finished dictations* folder
- a custom folder

4.1.5 Running the service in "DMZ" environment

Starting from version 9.0, the service provides explicit support for running in "DMZ" environment. In such a scenario, the service does not have write permissions outside of the environment, so copying or moving dictations outside of the environment is not allowed. Therefore, it must be possible to override the SpeechExec user's *Finished dictations* and *Archive* folder paths.

4.1.5.1 Configuring the service to run in "DMZ" mode

To run the service in "DMZ" mode the following settings are available in the web.config file of the web service:

| Setting name | Description |
|----------------------------------|---|
| IsRunningInDMZ | This setting value (boolean) controls whether "DMZ" mode is enabled or not. Supported values: true false |
| LocalFolderRootPathForDMZ | A local, custom path used as the root folder for dictations. Examples: "C:\DMZ_ROOT" |

4.1.5.2 Resolving folder paths for dictations

When the service runs in "DMZ" mode, the dictation folder paths are resolved the following way:

- Finished dictations: %LocalFolderRootPathForDMZ%\%UserName%\finished
- Archive: %LocalFolderRootPathForDMZ%\%UserName%\archive

Example:

LocalFolderRootPathForDMZ: "C:\DMZ_ROOT"

UserName: john

Folder path of Finished dictations: "C:\DMZ_ROOT\john\finished"

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 SQL table:
MasterDataItemsTableForSpeechExecEnterprise

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

| Setting name | Description | Example |
|--|---|------------|
| MasterData.MSSQL.Server | The name of the server running the MSSQL server | myserver01 |
| MasterData.MSSQL.Database | The name of the database where the "MasterDataItemsTableForSpeechExecEnterprise" view is located | mydatabase |
| MasterData.MSSQL.UseSQLAuthentication | True if using username & password-based authentication False if using Windows authentication | true |

| Setting name | Description | Example |
|--|---|--|
| MasterData.MSSQL.SQIAuthentication.Username | SQL authentication username (when using SQL authentication) as a Base64 encoded value | c3FsdXNlcjE= non-encoded value: sqluser1 |
| MasterData.MSSQL.SQIAuthentication.Password | 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

5.1 Overview

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

Important! 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 conditions above are not met, the test will fail at "020_SEELicenseServerAddressFoundInConfig".

5.2 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 **unrestricted**: **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 (,).

5.3 Test process

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

Validation steps:

| StepID | Explanation |
|---|--|
| 901_ServerLocalTime | Returns the local and UTC time of the web server host |
| 903_ServerUtcTime | |
| 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 temporary root folder is accessible (all types of temporary root folders are tested) |
| 101_AccessTokenLifetimeFoundInConfig | The <code>AccessTokenLifetimeInMinutes</code> value is found in web.config |
| 111_DMZRootFolderIsValidUNCPathAndAccessible | The <code>IsRunningInDMZ</code> value is <code>true</code> and the <code>LocalFolderRootPathForDMZ</code> value is a valid UNC path and accessible |

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

5.4 Response examples

Example response result when the 1st step passed, but the 2nd step failed:

```
[
  {
    "StepID": "901_ServerLocalTime",
    "StepResult": "2023.06.15 09:00:00.000",
    "StepDescription": "Server local time"
  },
  {
    "StepID": "903_ServerUtcTime",
    "StepResult": "2023.06.15 07:00:00.000",
    "StepDescription": "Server UTC time"
  },
  {
    "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 passed:

```

[
  {
    "StepID": "901_ServerLocalTime",
    "StepResult": "2023.06.15 09:05:00.000",
    "StepDescription": "Server local time"
  },
  {
    "StepID": "903_ServerUtcTime",
    "StepResult": "2023.06.15 07:05:00.000",
    "StepDescription": "Server UTC time"
  },
  {
    "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); "
    },
    {
      "StepID": "101_AccessTokenLifetimeFoundInConfig",
      "StepResult": "SUCCESS",
      "StepDescription": "AccessTokenLifetimeInMinutes value is found in
web.config"
    },
    {
      "StepID": "111_DMZRootFolderIsValidUNCPathAndAccessible",
      "StepResult": "SUCCESS",
      "StepDescription": "LocalFolderRootPathForDMZ configuration value
is a valid UNC path and accessible (c:\\DMZ_root)"
    }
  ]
}
```

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 assign an application pool to 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 on **Manage Application > Advanced settings...**
3. In the **Advanced Settings** dialog, select **Application Pool** and click on the **Browse** button (...).
4. In the **Select Application Pool** dialog, select **SEWebServicesAppPool**, and click on the **OK** button.

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:

2. Open the Internet Information Services (IIS) Manager.
3. Select **Application Pools** from the **Connections** panel on the left. Right-click on the **Application Pools** panel and select **Add Application Pool...**
4. Enter a name for your new application pool, such as **SEWebServicesAppPool**.
5. In the **.NET CLR version** list, select :
 - .NET CLR Version v4.0.30319
6. Make sure the **Start application pool immediately** checkbox is selected.
7. Click on the **OK** button to create and start the application pool.
8. Select the newly created application pool and click on **Edit Application Pool > Advanced settings...**
9. Make sure that the **Enable 32-bit applications** setting is:
 - False