



Byosphere[®] Data Uploader Guide

December 2025

5.12

Protein Metrics LLC, Boston, Massachusetts, USA

Contents

Introduction.....	1
System Requirements.....	1
Install the Automation Server Package.....	1
Configure the Automation Server.....	1
Find Byosphere Upload Folder ID	7
Run as Windows Service.....	7
Windows Credential for Automation Server	9
Application Notes	11

Introduction

The Protein Metrics Byosphere® Data Uploader is a server-based system which continuously uploads new or newly edited files to the Byosphere server. For example, as MS files are generated or copied to a designated source folder, they will automatically upload to Byosphere. In addition, the Data Uploader compresses MS sample files and folders to the proprietary *.pacq format, rendering them readable in Byosphere Desktop analyses. The Data Uploader is included in the Byosphere system.

System Requirements

The minimum hardware/software requirements for the machine that will host the Data Uploader is listed below:

CPU	Memory	Storage	OS	Network
>= 2 cores	>= 16 GB	>= 100 GB	Windows Server 2022	>= 1 GB/s

The machine will require access to the file system with the files/folders to be uploaded.

Install the Automation Server Package

Ideally, the Automation Server should be installed in an environment that is independent from the Analysis Service so that they do not compete for the same resources.

1. Download the Automation Server installer to the local environment and run the installer.
2. On the local environment, run Byosphere to initialize the server URL in the configuration file and test the connection:
 - a) Run `\ProteinMetrics\PMI-Byosphere-Client\Base\PMI-Byosphere-Client.exe`
 - b) Choose **Server > Configure**, enter the Byosphere server URL, and click **OK**

Configure the Automation Server

Modify `appsettings.json` in the installed `PMIAutomationService` directory as follows:

Parameter, Req'd & Default value	Note
<i>Parameter:</i> Host <i>Required:</i> Yes	Byosphere Application Server IP address
<i>Parameter:</i> UseAPIToken <i>Required:</i> Yes, if using API token	If true, set Windows Credential to use API token set on Application Server (see below). An API token is recommended for Byosphere v 4.1 and above.
<i>Parameter:</i> KeychainNameSpace <i>Default:</i> PmiAutomationCredentials	Must be consistent with the credentials set up using <code>PMI-Keychain-Manager.exe</code> in Windows Credential Store; see notes below

Parameter, Req'd & Default value	Note
<i>Parameter:</i> LocalWorkingPath <i>Default:</i> <code>Environment.GetFolderPath(Environment.SpecialFolder.CommonApplicationData)\ProteinMetrics\AutomationServer</code>	Folder location to store temporary files generated and then deleted during uploads. The default value will be used if the customer value is not specified or accessible.
<i>Parameter:</i> ExecPath <i>Default:</i> <code>Environment.GetFolderPath(Environment.SpecialFolder.ProgramFiles)\ProteinMetrics\PMI-Byosphere-Client\Base</code>	The <code>console.exe</code> executable path from the Protein Metrics Byosphere Client software installation
<i>Parameter:</i> BaseDir <i>Required:</i> On prem only	This is a required parameter for OnPrem only. This is the NAS mount drive. This setting in Windows must match its counterpart in App Server's <code>base_dir</code> in Linux. Example: <code>"BaseDir": "C:"</code>
<i>Parameter:</i> AutomationConfigurationID <i>Required:</i> Yes	The value is the Automation configuration ID in the Admin Portal. This is a required parameter for Data Uploader and Auto Processor. It is not required for Watcher.

Parameter, Req'd & Default value	Note
<i>Parameter:</i> LastScanTime	<p>This value is auto-created / updated by the last program run. Format is "yyyy-MM-dd HH:mm:ss"</p> <p>Example: "LastScanTime": "2021-06-01 19:58:56" (HH is a 24-hour clock)</p>
<i>Parameter:</i> ConfirmFolderCanPacq <i>Default:</i> false	<p>If true, checks each folder before upload to confirm if it is an MS folder. This is used if MS folders contain file extensions that are also listed in the SearchPattern parameter, (e.g. .csv, .txt, .xml). Under those circumstances, this setting avoids a file upload of the SearchPattern extension in a separate folder. Note: this parameter should not be set if unneeded since it will significantly affect performance.</p>
<i>Parameter:</i> RollingWindowDelta	<p>The unit is second. The Automation will check for new data using time stamp setup in Modified After or LastScanTime. However, sometimes NAS data is not immediately available for cloud instance to be picked up. The delta window makes the Modified After or LastScanTime more tolerant in such cases. If there is data with timestamp 12-20-22 11:50 AM, data will not be uploaded using "Modified After": "2022-12-20 12:30:00".</p> <p>Data will be uploaded using the following, given a 3600 second or 1 hour delta effectively checking the file against 12-20-22 11:30 AM.</p> <p>"RollingWindowDelta": 3600,</p> <p>"Modified After": "2022-12-20 12:30:00"</p> <p>If no value is specified, no delta window will be applied.</p> <p>Example: "RollingWindowDelta": 3600</p>

Parameter: CustomSettings

CustomSettings is an array of custom codes. Custom code will be executed in the order they are listed. Parameter includes

- CustomCodeExePath, optional CustomCode, optional MetadataPath. This set is executed for each sample.
- CustomCodeExePathBefore, optional CustomCodeBefore, optional MetadataPathBefore. This set is executed before Automation starts scanning for new data.
- CustomCodeExePathAfter, optional CustomCodeAfter, optional MetadataPathAfter. This set is executed after Automation has processed current scan data.
- CustomCodeExePathAutomationTrigger, optional CustomCodeAutomationTrigger, optional MetadataPathAutomationTrigger. This set is executed for each Automation Trigger file.

MetadataPath can contain output from the custom code. Contact support@proteinmetrics.com for more information on integration with third party tools through custom codes. Examples,

- First custom code executes python code,
- Second custom code executes python code with a parameter,
- Third example is an executable,
- Fourth example is an executable with a parameter and a metadata output path

```
"CustomSettings": [
  {
    "CustomCodeBeforeExePath":
"python.exe",
    "CustomCodeBefore":
"PMIGetMetadata.py"
  },
  {
    "CustomCodeAfterExePath":
"python.exe",
    "CustomCodeAfter": "\"D:\\Test
Data\\PMIGetFasta.py\" --protein"
```

	<pre> "CustomCodeAutomationTrigger": "GenedataIntegration.exe" }, { "CustomCodeExePath": "BioviaIntegration.exe", "CustomCode": "-f", "MetadataPath": "E:\\ProgramData\\ProteinMetrics\\BioviaMetadata" }], </pre>
Parameter: ApplyMetadataToExistingFile	<p>When <code>ApplyMetadataToExistingFile</code> is <code>true</code>, additional metadata might be applied to a file already in Byosphere. When the value is <code>false</code>, the Byosphere file will not be updated with metadata. Default value is <code>false</code>.</p> <p>Example: <code>"ApplyMetadataToExistingFile": true</code></p>
Parameter: SkipPacqHashCheck	<p>Automation relies on pacq hashes to detect modifications and trigger re-uploads. However, for immutable MS files, this check can be bypassed. By default (<code>false</code>), pacq hash comparison occurs. Setting this value to <code>true</code> skips the pacq hash check.</p>
Parameter: SkipFileHashCheck	<p>Automation relies on file hashes to detect modifications and trigger re-uploads. However, for immutable non-MS files, this check can be bypassed. By default (<code>false</code>), file hash comparison occurs. Setting this value to <code>true</code> skips the file hash check.</p>
Parameter: LocalInputRoot (optional)	<p>Automation by default honors local folder hierarchy and will create a matching folder hierarchy in Byosphere. <code>LocalInputRoot</code> parameter allows user to define common path that will be skipped during Byosphere subfolder creation processor therefore redefine a root folder.</p>

	<p>LocalInputRoot is case insensitive. LocalInputPath and LocalInputRoot are expected to have the same path format of Windows, Linux, or UNC path.</p> <p>Examples in Linux or Windows filepath format:</p> <p>LocalInputPath c:/a/b/z, LocalInputRoot c:/a/b => common path c:/a/b -> data will be uploaded to Output Folder > z</p> <p>LocalInputPath c:/a/b, LocalInputRoot empty => no common path -> data will be uploaded to Output Folder > a > b</p> <p>LocalInputPath c:\a\b, LocalInputRoot c:\m => no common path -> data will be uploaded to Output Folder > a > b</p> <p>LocalInputPath c:\a\b\z, LocalInputRoot c:\a\b\z\y => common path c:/a/b/z -> data will be uploaded to Output Folder</p>
<p><i>Parameter:</i> UserGroupIds (optional)</p>	<p>Specifies the ids of user groups that should be assigned to the new folders created when uploading a file. The groups will be assigned to all uploaded folders or to subfolders from the level specified by UserGroupSubfolderLevel.</p> <p>Example:</p> <pre>"UserGroupIds" : "2,3"</pre>
<p><i>Parameter:</i> UserGroupSubfolderLevel (optional)</p> <p>Default: 1</p>	<p>Specifies the level of LocalInputPath directory from where UserGroupIds should be applied when uploading folders.</p> <p>When this parameter is not provided or its value is less than or equal to 1, UserGroupIds will be applied to all new folders.</p> <p>Examples:</p> <pre>"LocalInputPath": "C:/a/b/c/d/e", "UserGroupIds": "2,3"</pre> <p>Example 1:</p> <pre>"UserGroupSubfolderLevel ": 1</pre> <p>Result: User groups will be applied to all subfolders: a,b,c,d,e</p> <p>Example 2:</p> <pre>"UserGroupSubfolderLevel ": 3</pre> <p>Result: User groups will be applied to the following subfolders: c,d,e</p> <p>Example 3 (when LocalInputRoot is provided):</p> <pre>"LocalInputRoot": "C:/a/b/c",</pre>

	<p>"UserGroupSubfolderLevel ": 3</p> <p>Result: User groups will be applied to the following subfolders: d,e (subfolder 'c' will not be uploaded due to LocalInputRoot value)</p>
<i>Parameter:</i> CheckCreationTime	<p>Default to false.</p> <p>When set to false, current behavior will be observed where only LastWriteTime is checked. This is the common case where data comes off acquisition machine, Modified is later than Created.</p> <p>When set to true, we will check both CreationTime and LastWriteTime, and use the later of the two. This is the case when a file is copied from one drive to another drive, Created is later than Modified.</p>

Note: The Data Uploader supports the upload of Chromeleon files for version. 7.2 and above. Contact support@proteinmetrics.com for more information and for the **PMI Chromeleon Watcher Guide.pdf**

Find Byosphere Upload Folder ID

Byosphere folder IDs are displayed when uploading to or downloading from a selected folder. To identify the folder ID, do the following:

1. Create the upload folder in Byosphere. If the Keychain credentials for the Data Uploader is not a Super User, then add a user group for that user name with File Editor, Folder Editor and Viewer privileges.
2. In Byosphere Desktop, login to the server and choose **Server > Upload**.
3. In the **Server file/folder** cell, click the "...” button.
4. Select the upload destination folder at left and click **Choose**.
5. The folder ID is displayed in the **Upload file/folder** dialog, in this example "2026":

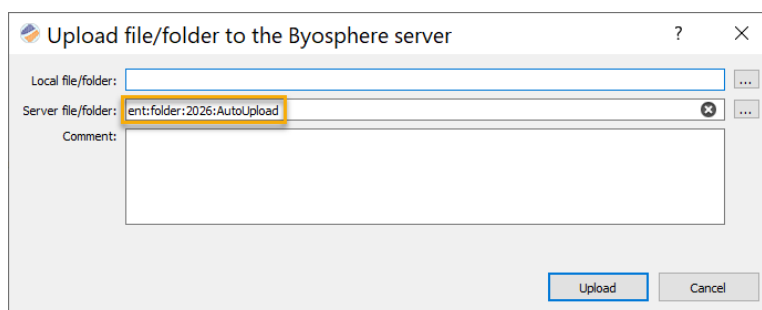


Figure 1: Identify the Byosphere upload folder ID

Run as Windows Service

Protein Metrics Automation Server is recommended to run as a Windows Service rather than as a console application. The executable sc.exe in CMD can also be used to create, start, stop, or delete a Windows Service:

:: Create a Windows Service for the Automation Server

```
sc create PmiAutomationService DisplayName="PMI Automation Server"  
binPath="C:\Program Files\ProteinMetrics\Pmi-Automation-  
Service\PmiAutomationService.exe"
```

:: Start a Windows Service

```
sc start PmiAutomationService
```

:: Stop a Windows Service

```
sc stop PmiAutomationService
```

:: Delete a Windows Service

```
sc delete PmiAutomationService
```

After configuring and verifying the setup, it is recommended to set the Protein Metrics Automation Server **Start Type** to **Auto**. Windows services can be managed in the Windows Services App:

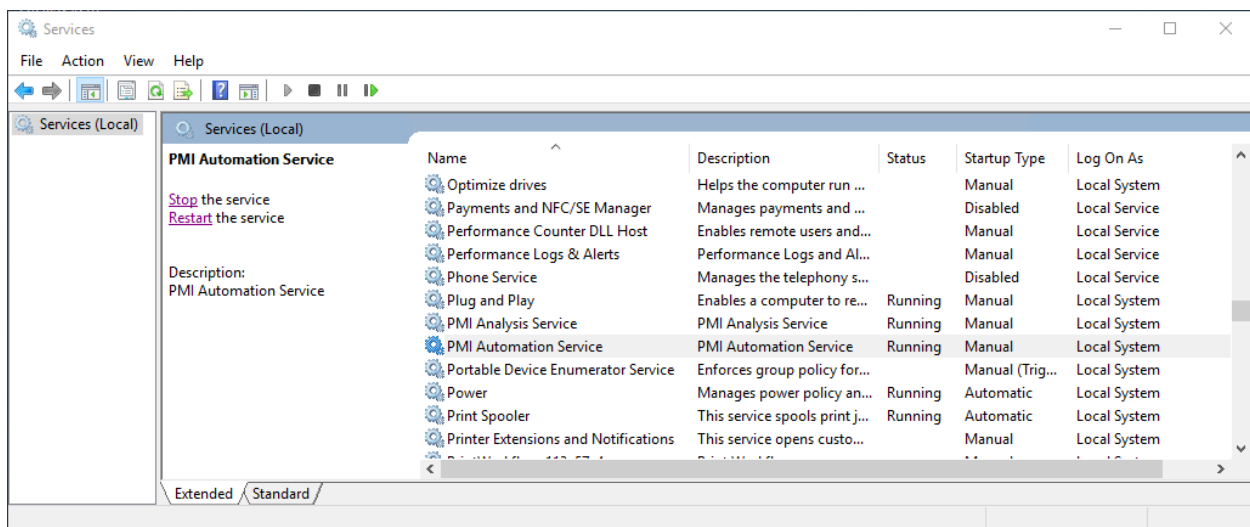


Figure 2: Window Services App

The Log On account for Protein Metrics Automation Service can be a Local Service:

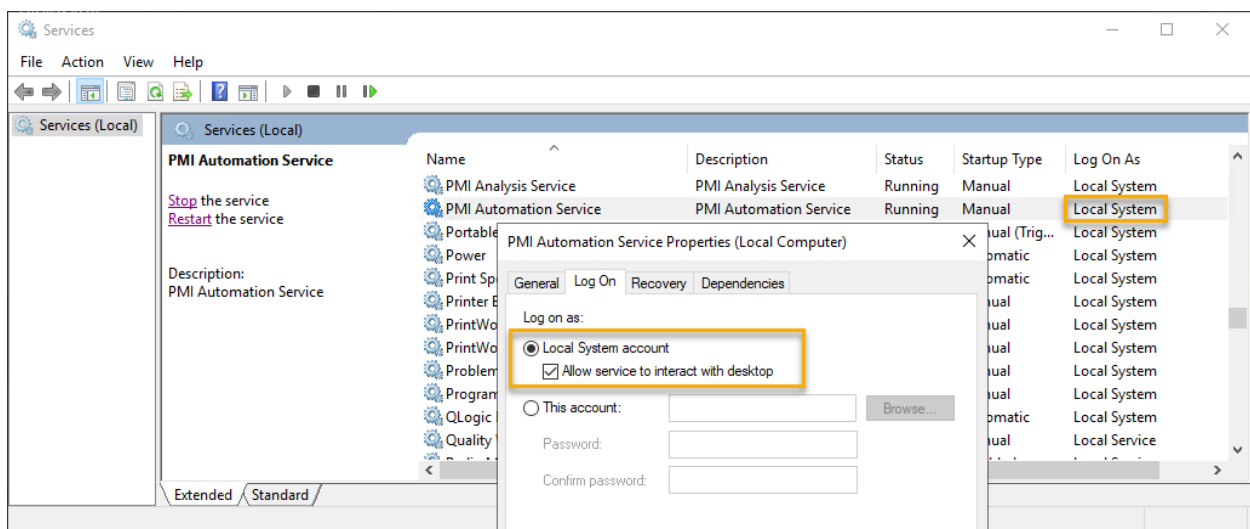


Figure 3: Protein Metrics Automation Service Local logon account

Alternatively, the Log On account for Protein Metrics Automation Service can be an individual user:

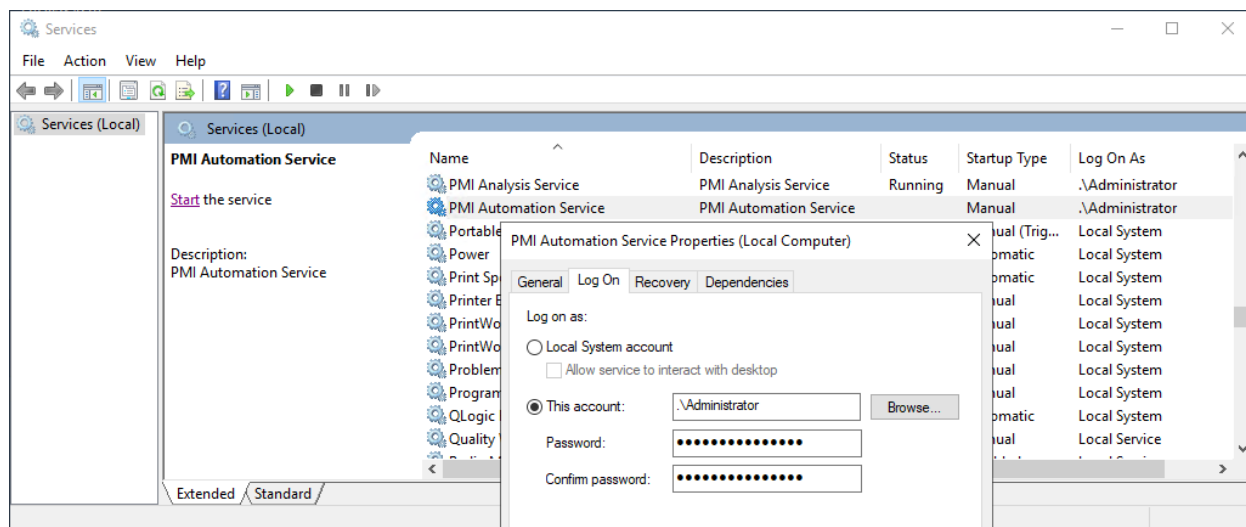


Figure 4: Protein Metrics Automation Service Administrator logon account

Note: After verifying the setup, Automation Server failure recovery can be adjusted according to Company IT policies.

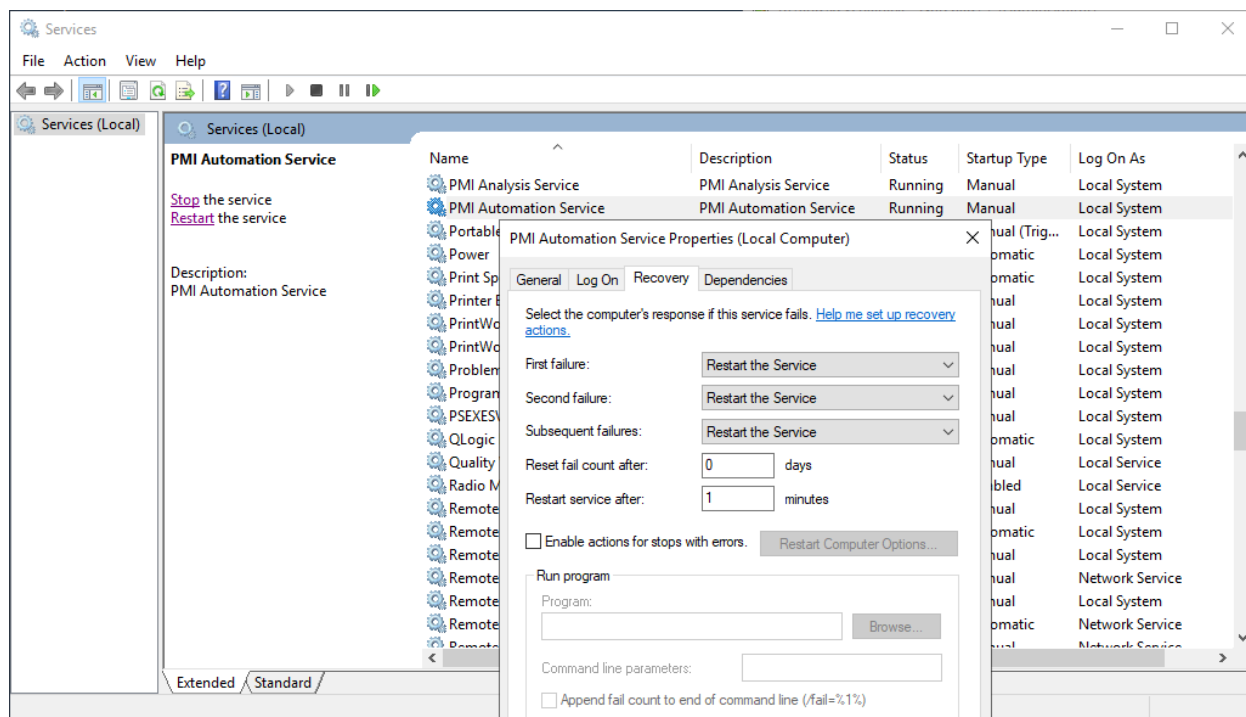


Figure 5: Automation Server failure recovery

Windows Credential for Automation Server

Automation Server uses Windows Credential Store to avoid exposing user passwords in the configuration. Commands are run in the CMD console. For Administrator log on services, a simple CMD window is used. For Local System log on services, there are many tools to access Local System account. One way is to use <https://docs.microsoft.com/en-us/sysinternals/downloads/pstools>

- Start CMD, Run as administrator
- Execute the following command:

```
psexec -i -s cmd.exe
```
- This will launch a new CMD window granting Local System account access
- Windows Credentials setup can be executed from the Local System account

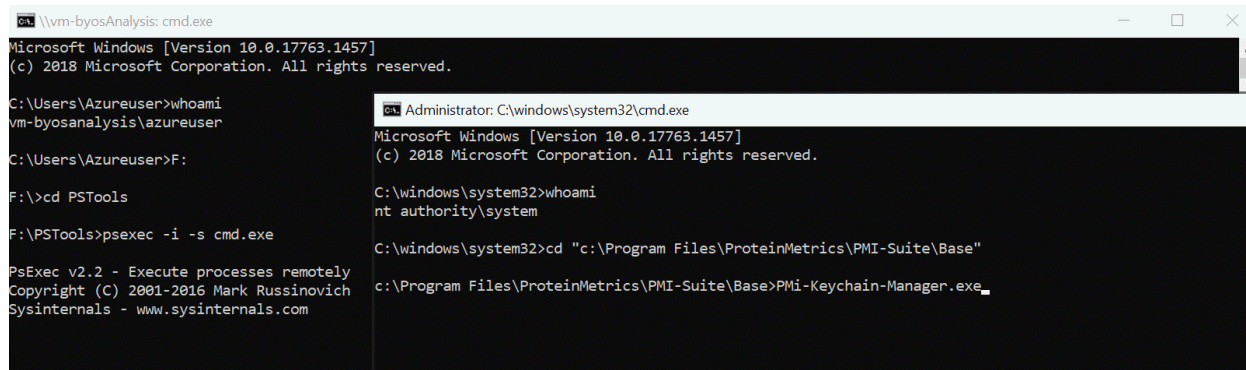


Figure 6: Accessing Local System account

To configure the API token used in App Server authentication for Automation Server:

- Follow App Server Guidance on setup API Token
- Open a CMD window as shown above.
- Customize and run the following commands:

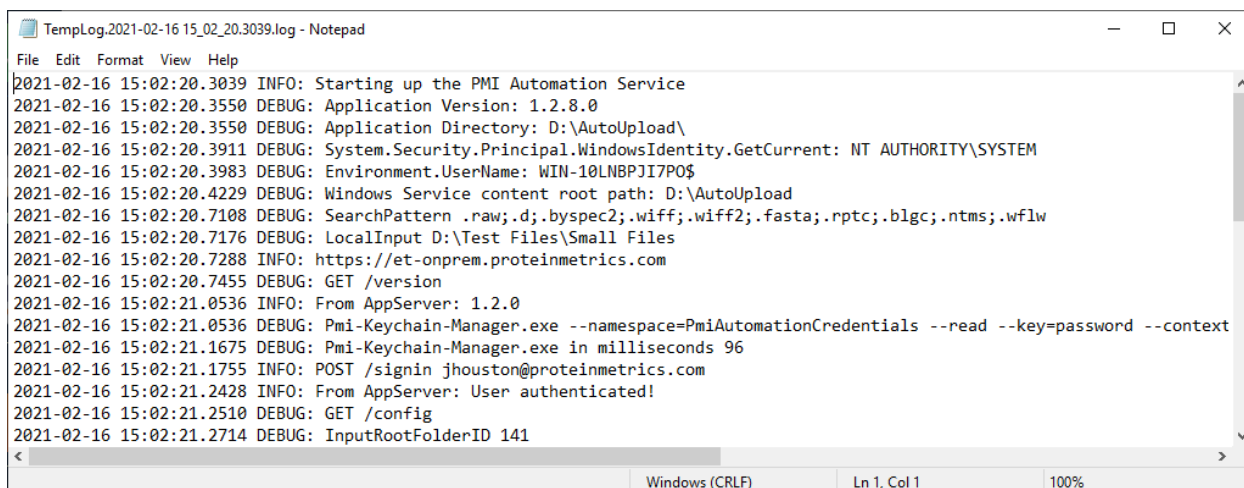
```
PMi-Keychain-Manager --delete --namespace=PmiAutomationCredentials
PMi-Keychain-Manager --add --namespace=PmiAutomationCredentials --
key=server --value="https://byosphere.company.com"
PMi-Keychain-Manager --add --namespace=PmiAutomationCredentials --
key=api-token --value=ABCDEabcde
PMi-Keychain-Manager --read --namespace=PmiAutomationCredentials --
key=api-token --context-data="server=https://byosphere.company.com"
```

The final line will display the API Token value for the Byosphere server URL. This user account assigned to the API Token requires the Super User privilege and the Contributor entitlement.

Note: Credentials only need to be setup one. In the future updates, run the last line to verify.

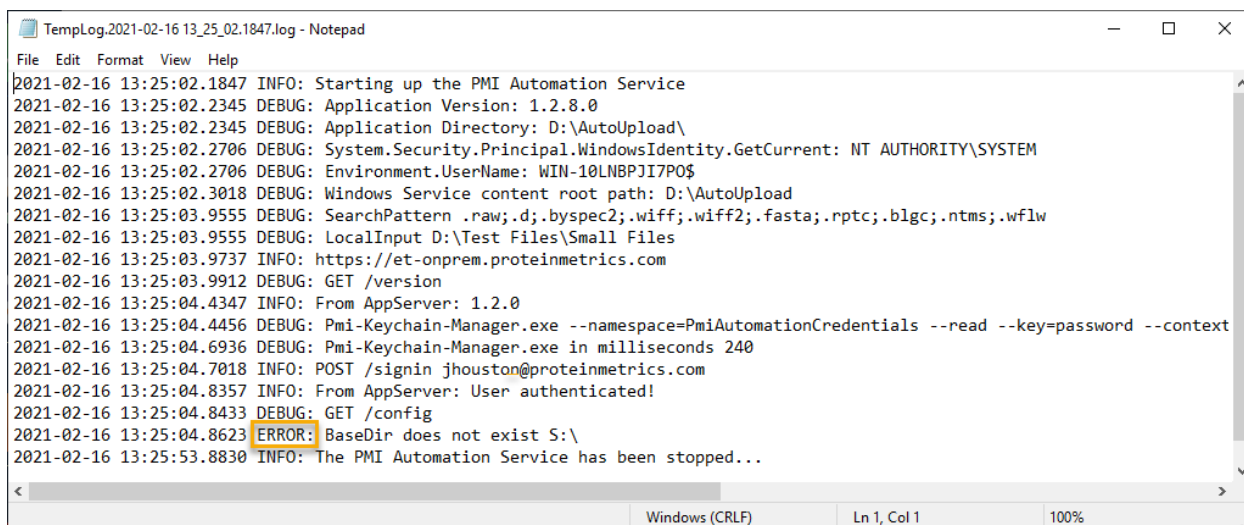
Note: Automation Server is set to log on as **Local System** by default. Depending on Company IT policies, Windows Credential might be necessarily to setup from Local System account. To find out more, please refer to <https://docs.microsoft.com/en-us/windows/win32/services/localsystem-account> and

<https://docs.microsoft.com/en-us/windows/win32/ad/the-localsystem-accountExample> of a successful Automation Server startup using username and password: expect to see no errors:



```
TempLog.2021-02-16 15_02_20.3039.log - Notepad
File Edit Format View Help
2021-02-16 15:02:20.3039 INFO: Starting up the PMI Automation Service
2021-02-16 15:02:20.3550 DEBUG: Application Version: 1.2.8.0
2021-02-16 15:02:20.3550 DEBUG: Application Directory: D:\AutoUpload\
2021-02-16 15:02:20.3911 DEBUG: System.Security.Principal.WindowsIdentity.GetCurrent: NT AUTHORITY\SYSTEM
2021-02-16 15:02:20.3983 DEBUG: Environment.UserName: WIN-10LNBPJ17PO$
2021-02-16 15:02:20.4229 DEBUG: Windows Service content root path: D:\AutoUpload
2021-02-16 15:02:20.7108 DEBUG: SearchPattern .raw;.d;.byspec2;.wiff;.wiff2;.fasta;.rptc;.blgc;.ntms;.wflw
2021-02-16 15:02:20.7176 DEBUG: LocalInput D:\Test Files\Small Files
2021-02-16 15:02:20.7288 INFO: https://et-onprem.proteinmetrics.com
2021-02-16 15:02:20.7455 DEBUG: GET /version
2021-02-16 15:02:21.0536 INFO: From AppServer: 1.2.0
2021-02-16 15:02:21.0536 DEBUG: Pmi-Keychain-Manager.exe --namespace=PmiAutomationCredentials --read --key=password --context
2021-02-16 15:02:21.1675 DEBUG: Pmi-Keychain-Manager.exe in milliseconds 96
2021-02-16 15:02:21.1755 INFO: POST /signin jhouston@proteinmetrics.com
2021-02-16 15:02:21.2428 INFO: From AppServer: User authenticated!
2021-02-16 15:02:21.2510 DEBUG: GET /config
2021-02-16 15:02:21.2714 DEBUG: InputRootFolderID 141
```

Figure 7: Example of Successful Automation Server startup



```
TempLog.2021-02-16 13_25_02.1847.log - Notepad
File Edit Format View Help
2021-02-16 13:25:02.1847 INFO: Starting up the PMI Automation Service
2021-02-16 13:25:02.2345 DEBUG: Application Version: 1.2.8.0
2021-02-16 13:25:02.2345 DEBUG: Application Directory: D:\AutoUpload\
2021-02-16 13:25:02.2706 DEBUG: System.Security.Principal.WindowsIdentity.GetCurrent: NT AUTHORITY\SYSTEM
2021-02-16 13:25:02.2706 DEBUG: Environment.UserName: WIN-10LNBPJ17PO$
2021-02-16 13:25:02.3018 DEBUG: Windows Service content root path: D:\AutoUpload
2021-02-16 13:25:03.9555 DEBUG: SearchPattern .raw;.d;.byspec2;.wiff;.wiff2;.fasta;.rptc;.blgc;.ntms;.wflw
2021-02-16 13:25:03.9555 DEBUG: LocalInput D:\Test Files\Small Files
2021-02-16 13:25:03.9737 INFO: https://et-onprem.proteinmetrics.com
2021-02-16 13:25:03.9912 DEBUG: GET /version
2021-02-16 13:25:04.4347 INFO: From AppServer: 1.2.0
2021-02-16 13:25:04.4456 DEBUG: Pmi-Keychain-Manager.exe --namespace=PmiAutomationCredentials --read --key=password --context
2021-02-16 13:25:04.6936 DEBUG: Pmi-Keychain-Manager.exe in milliseconds 240
2021-02-16 13:25:04.7018 INFO: POST /signin jhouston@proteinmetrics.com
2021-02-16 13:25:04.8357 INFO: From AppServer: User authenticated!
2021-02-16 13:25:04.8433 DEBUG: GET /config
2021-02-16 13:25:04.8623 ERROR: BaseDir does not exist S:\
2021-02-16 13:25:53.8830 INFO: The PMI Automation Service has been stopped...
```

Figure 8: Example of Failed Automation Server startup

Application Notes

- New files are identified by content, not by name. If a file is edited, it will be uploaded as new version over the existing file with that name.
- Files which have MS sample extensions but are not recognized by the Byos application as MS files or folders (for example, corrupted data) are uploaded but not compressed into *.pacq format. Multiple versions of these files may then be generated.
- Log files written to the PMIAutomationService\Logs directory can be renamed or configured by editing the file PMIAutomationService\NLog.config. For more information, please refer to <https://github.com/nlog/nlog/wiki/Configuration-file> or contact support@proteinmetrics.com. Note that log level can be Debug, Info, Warning, Error, and Fatal. It is recommended to start with Debug level logging during initial deployment to enable Protein Metrics to better troubleshoot any issues. It is recommended to use Info in normal operation; Warning or high to minimize logging. User can delete log files at PMIAutomationService\Logs subdirectory if log information is no longer needed. TempLog file size can be controlled with the archiveAboveSize parameter. The default value is 1MB (archiveAboveSize="1000000"). The maxArchiveFiles parameter defines the

maximum number of archived log files to keep for Info logs or Debug logs. The default value is 30 (maxArchiveFiles="30") When the number of archived files exceeds this limit, the oldest files are automatically deleted.