AvaTax for Communications SaaS Standard

Sample FTP Upload and Download Scripts

Document: TM_00560_0002
Date: 10/25/2017

Avalara for Communications - Contact Information	
Address	Avalara, Inc.
	512 South Mangum Street, Suite 100
	Durham, North Carolina, 27701
Email	CommunicationSupport@avalara.com
Help Center	https://help.avalara.com/Avalara_for_Communications
Toll Free	(877) 780-4848
Corporate Web Site	http://communications.avalara.com/
Communications Customer Portal	https://communications.avalara.net

Document Revision History

The Revision History log lists the date and description of the most recent revisions or versions of the document.

Date	Version	Description
04/06/2016	0001	Initial document
10/25/2017	0002	Updated Avalara contact information (address and support site). Removed **Appendix D – Avalara Product Names**.

Table of Contents

1. Introduction

AvaTax for Communications (AFC) SaaS Standard is a tax-rating service bureau. It provides the necessary link to allow you to transfer files to and from Avalara for processing by the AFC server. AFC SaaS Standard provides two ways to connect to the AFC server - by using a file-transfer protocol (FTP) connection through an existing account and typing 'ftp.billsoft.com' at the prompt within the FTP session or by connecting to the web interface at https://ftp.billsoft.com.

Once your file has been uploaded to the server, AFC begins immediate processing of your data. AFC SaaS Standard and the AFC server are online - 24 hours a day, 7 days a week - to process your data.

This document provides steps to upload and download a file using the sample PowerShell scripts provided by Avalara. There are many other scripting languages and tools are available to provide the same functionality as shown in this sample.

Throughout this document, a sample file named **KML150117AA.csv** is being used to illustrate the process. The filenames, paths, and user credentials used throughout this document are for demonstration purposes only and should be replaced with valid values.

Please refer to **TM_00111_ AFC SaaS Standard User Manual** for additional details and features of AFC SaaS Standard.

2. Prepare the ftp.zip Input File

2.1 Create the Primary Transaction Input File

The primary file contains all transactions to be processed and can be either **Extended Format** (.csv/Excel) or **Fixed Format** (.cdf). The naming convention for the primary file must be named correctly and uniquely. Submitting the same file name will cause the file to be rejected.

- The 1st through the 3rd characters in the filename should begin with your company's three-letter code.
- The 4th and 5th characters in the filename should be the current year (2-digits).
- The 6th and 7th characters are the current month.
- The 8th and 9th characters are the current day.
- The 10th through the 13th characters are your unique files identifiers. Please note that 'T' is reserved for test files.
- The final character must be 'T' if this a test file. (optional or preferred)
- The file extension must be a supported extension.

Thus, an example of a correctly named file is *KML150117AA.CSV*, where *KML* is the three-letter code for the company, *15* is the year, *01* is the month and *17 is the day. AA* indicates it is the first file uploaded

that day, and *.CSV* indicates it is a comma delimited file for processing by AFC. Refer to the **Compliance Month** section of **TM_00111_AFC SaaS Standard User Manual** for additional details on compliance processing as well as the importance of naming for compliance purposes.

2.1.1 Extended Format – Excel, CSV

Extended format supports Excel and CSV (Comma-Separated Values) files. The contents and columns are identical and are discussed in **TM_00111_AFC SaaS Standard User Manual – Appendix A – Extended Format: Excel and CSV File Format** for additional details.

2.1.2 Fixed Format – CDF

The fixed format, or CDF format, supports a limited number of transaction options and is intended for basic transaction processing. Please refer to **TM_00111_AFC SaaS Standard User Manual – Appendix B – Fixed Format: CDF File Format** for additional details.

2.2 Optional Files

In addition to the primary transaction input file, a number of option files may be included for processing. These optional files can either be embedded in the input package along with the primary transaction input file (which will only apply to the current batch of transactions) or persisted within the account (which will apply to all batches of transactions). For more information regarding optional files, please refer to **TM_00111_ AFC SaaS Standard User Manual – Section 3.3.4 – Handling of Optional Files**.

2.3 Prepare the Input Package

When uploading your input package to the FTP site, it must be compressed (i.e., "zipped") into a file named ftp.zip in order to be processed automatically.

A single ftp.zip file will be submitted containing one and only one primary file and one or more optional files.
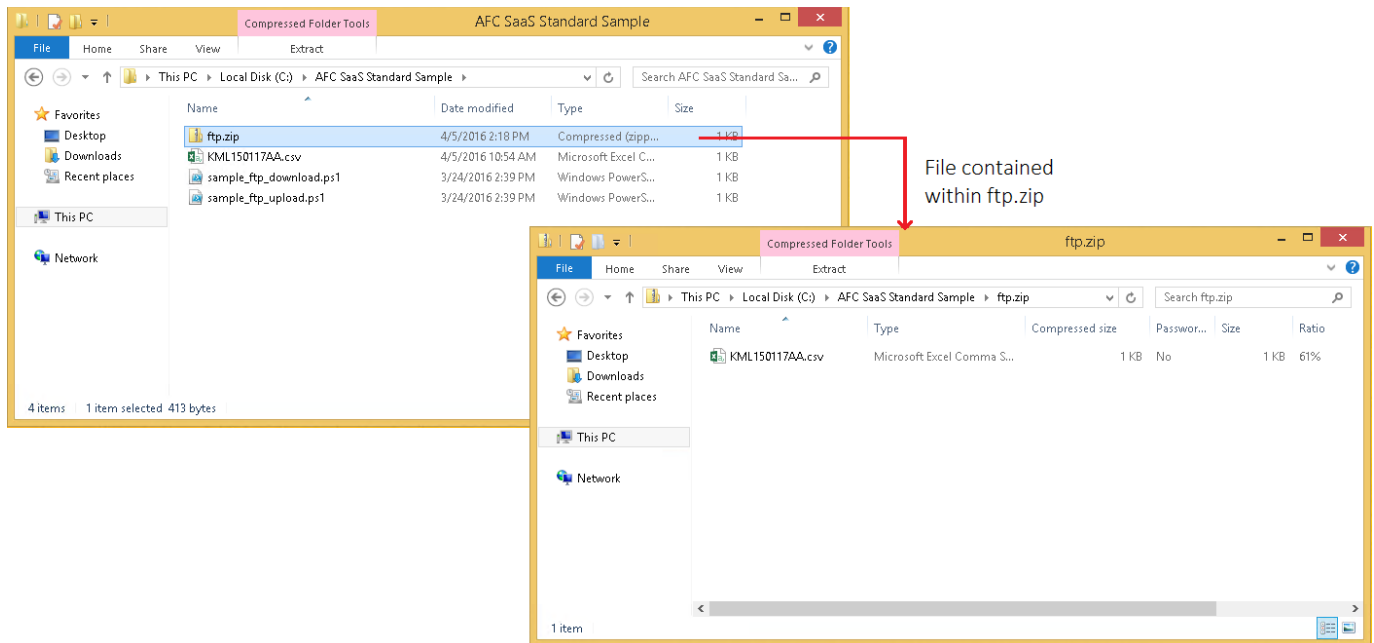
2.3.1   Example Input Package

Figure 1 – Creating the ftp.zip input package

3. Upload Files to the FTP Site using the Sample Script

The sample upload script is written in PowerShell and the PowerShell utility must be installed on the machine in order to utilize the scripts.

3.1 Setup

In order to perform the actions in this section, the logged-in user account must have **Administrator** access to the machine on which this script is being run.

To begin, right-click on the sample_ftp_upload.ps1 file and select Edit to open the script in PowerShell.
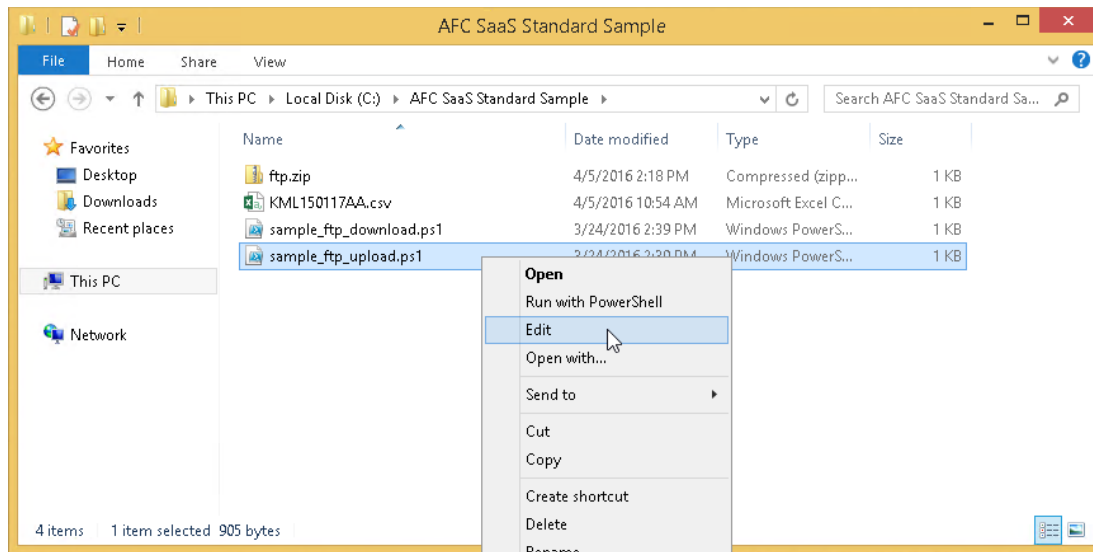
Figure 2- Editing sample_ftp_upload.ps1

The PowerShell program is opened.
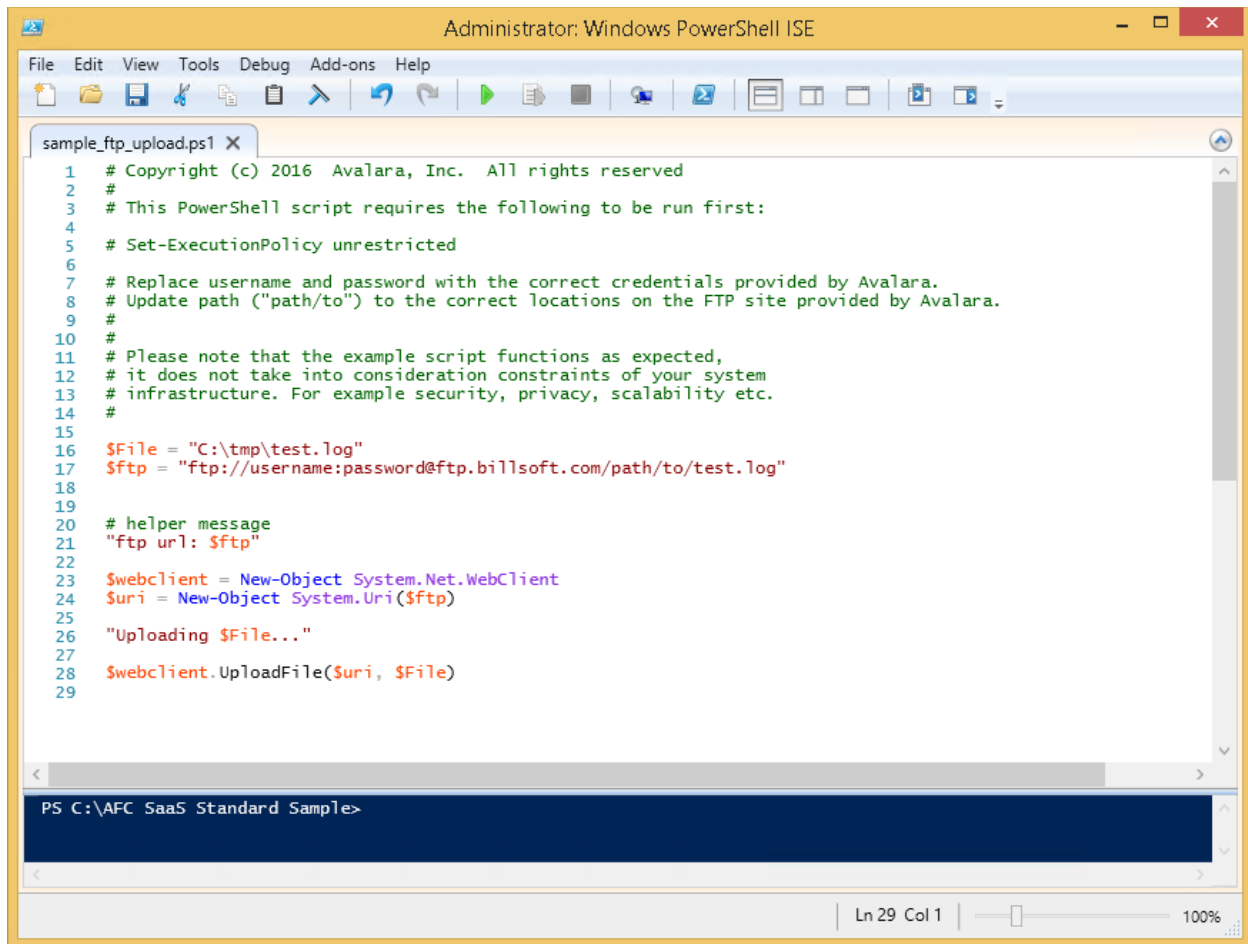
Figure 3- Opening the PowerShell Program

3.1.1 Set-ExecutionPolicy unrestricted Command

Prior to first use, the following command may need to be run as an Administrator.

Set-ExecutionPolicy unrestricted

This command is also provided as part of the sample_ftp_upload.ps1 script. To run from the sample_ftp_upload.ps1 script, highlight the command without the leading # sign and click on the **Run Selection** button within PowerShell.

Figure 4- Running the Set-ExecutionPolicy Command

The following question is displayed. Click **Yes** in the dialog box to continue.

Figure 5- Confirming the Execution Policy Change

3.1.2   Update the File Locations
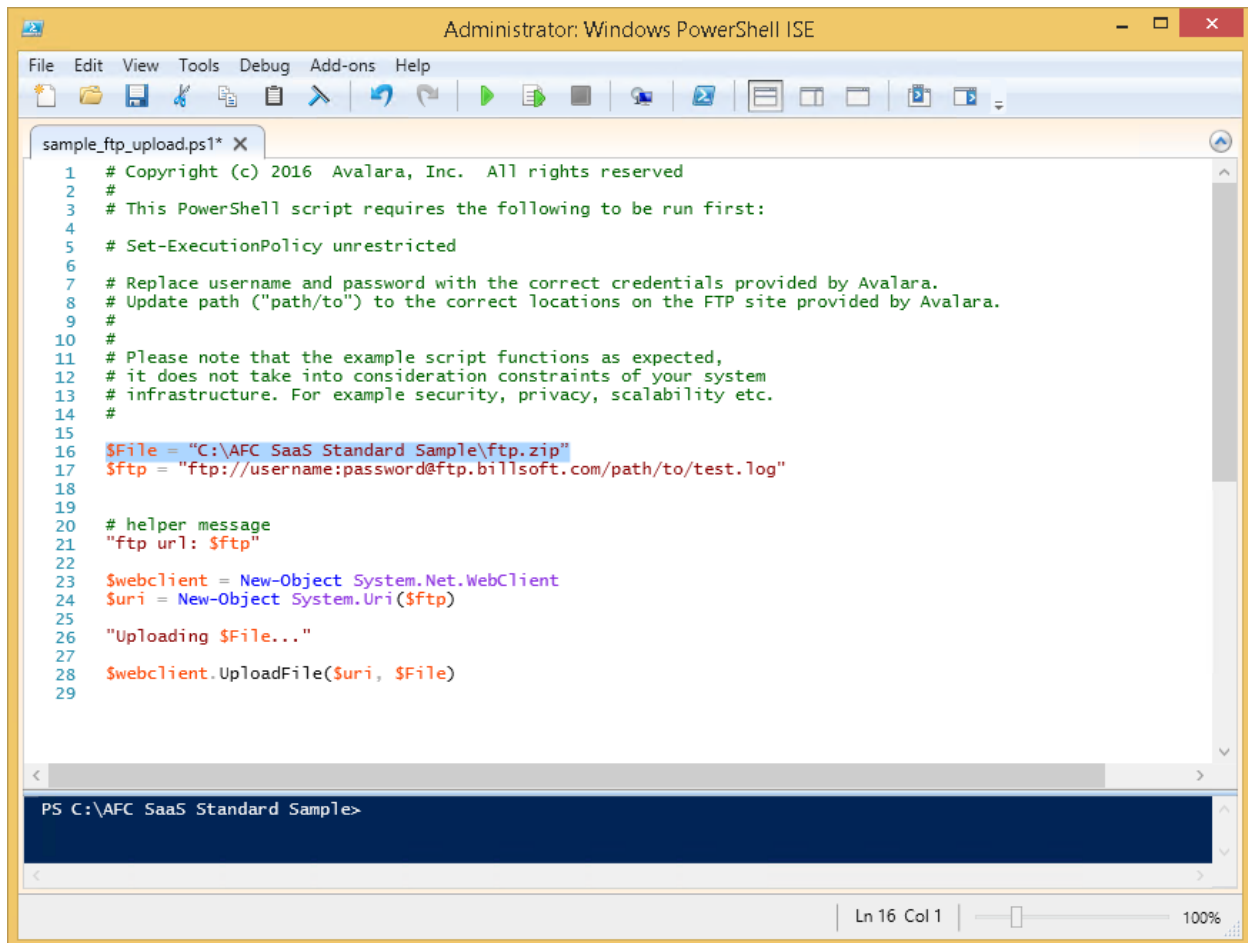
The next step is to update the file locations.

Figure 6- Updating File Locations

3.1.2.1 $File variable

This variable specifies the path to the file that shall be uploaded along with the filename to be uploaded. In this example, the ftp.zip file located at C:\AFC SaaS Standard Sample is the target.

```
$File = "C:\AFC SaaS Standard Sample\ftp.zip"
```

Note: If the file path contains spaces, the path must be enclosed in quotation marks as shown above.

Figure 7- $File Variable

3.1.2.2 $ftp variable

This variable defines the connection to the FTP server with the Avalara-provided FTP username, Avalara-provided FTP password, and upload path specified. In this example, the following information is being set:

Field	Value
Username	Testuser
Password	TestPass
FTP Path	ftp.billsoft.com/BSOL/KML/ftp.zip

$ftp = "ftp://Testuser:TestPass@ftp.billsoft.com/BSOL/KML/ftp.zip"

Note: The file path must be enclosed in quotation marks as shown above.

Figure 8- $ftp Variable

3.2 Execute the Script

Executing the sample_ftp_upload.ps1 file will upload the file specified in the $File variable to the FTP location specified using the username and password included in the $ftp variable. To execute the script, simply click the **Run Script** button within PowerShell.
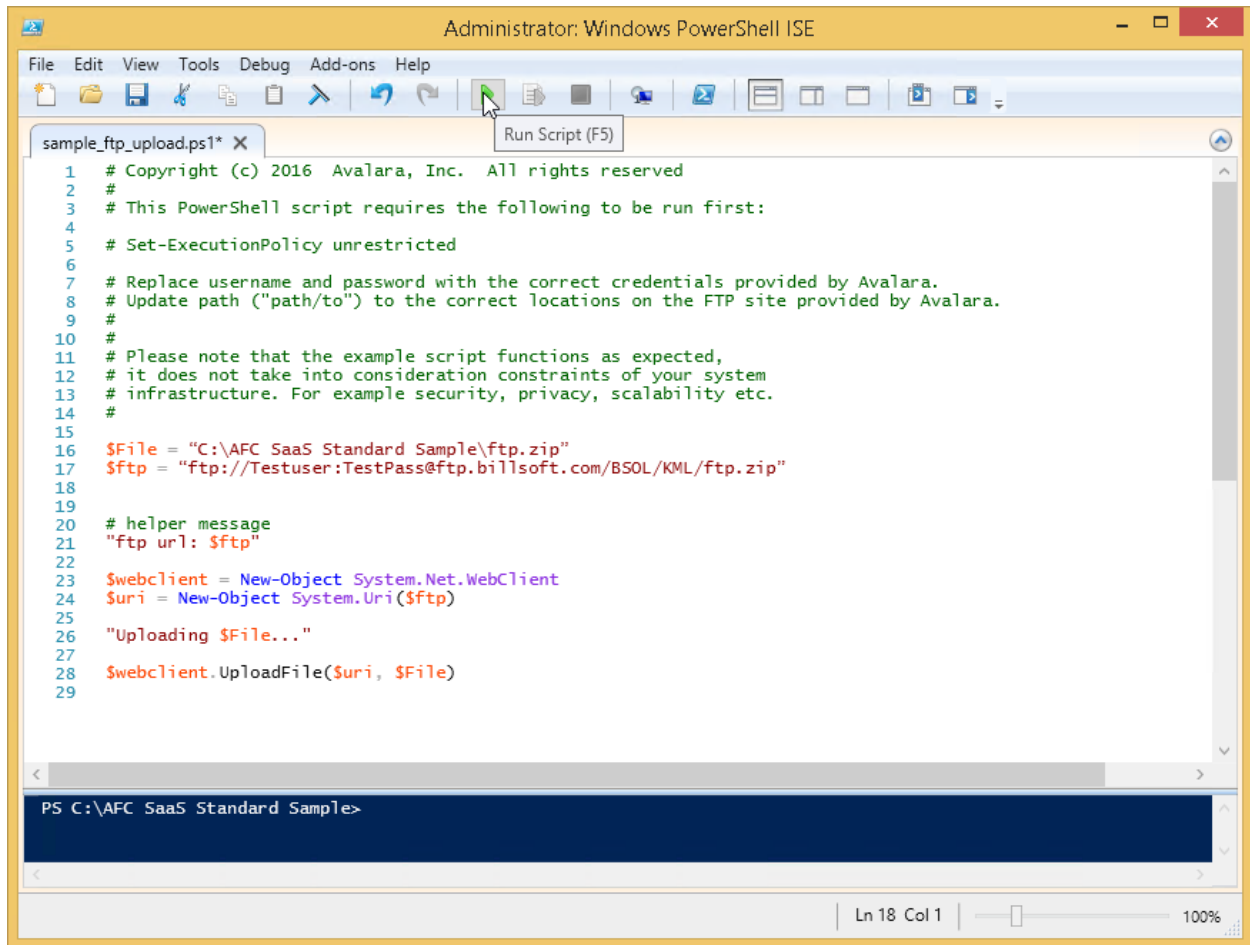

Figure 9 - Executing the Script

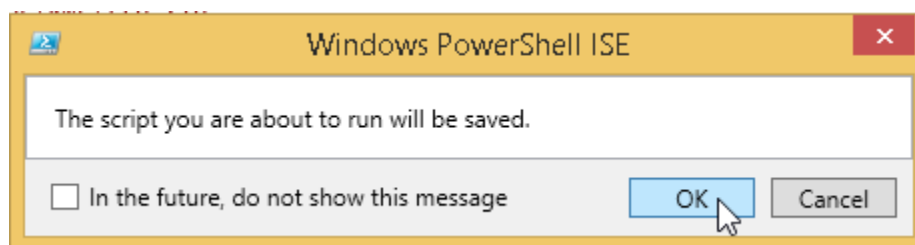A prompt may be displayed stating that the updates made to the script will be saved. Click **OK**.

Figure 10 - Saving Updates to the Script

The output section of PowerShell shows the results of the action:

```
PS C:\AFC SaaS Standard Sample> C:\BSOL\sample_ftp_upload.ps1
ftp url: ftp://Testuser:TestPass@ftp.billsoft.com/BSOL/KML/ftp.zip
Uploading C:\AFC SaaS Standard Sample\ftp.zip...
```

Figure 11- Results of Script

Note: Any error messages that may be returned can be found in the Results section of PowerShell.

4. Results Zip Files

The following zip files are produced when the KML150117AA input transaction file is uploaded.

4.1   KMLS150117AA.zip - Compliance Files

Extended and Fixed Format Compliance Files	
File Name	Description
KML150117AA.ssf	Summarized standard billable tax report.
KML150117AA.nba	Summarized Non-billable tax report (as applicable).
KML150117AA.nca	Summarized Non-compliance tax report (as applicable).

4.2   KMLR150117AA.zip - Report Files

Extended and Fixed Format Report Files	
File Name	Description
KML150117AA.csf	Summarized fixed position customer info report. Produced for clients configured for standard compliance reporting.
KML150117AA.tsr	Transaction/Service Jurisdiction Report. By default, all clients will receive the tsr report.
KML150117AA.rtr	Reverse Transaction/Service Jurisdiction Report (as applicable). Produced if client has requested the rtr report.
KML150117AA _ASCIILOG.CSV	Contents of the tax data log converted to a comma-delimited text file with header row included. Refer to **Section 5.3.1 asciilog.exe** in **TM_00101_AFC User Manual Telecom.pdf** for more information on this report. **Note:** Clients wishing to receive this report must submit a request to communicationsupport@avalara.com for configuration to add this to the list of reports.
EZTax.license.rpt	Detailed version of the Transaction Count Report. Provides a count of transactions processed by transaction/service pair.

Extended and Fixed Format Report Files

File Name	Description
{output_filename_format}.rpt	Custom Sort report metrics. Provides details of the Custom Sort processing. {output_filename_format} is defined in customsort.cfg.
{output_filename_format}_nca.csv	Non-compliance Custom Sort Report (as applicable based on customsort.cfg settings). {output_filename_format} is defined in customsort.cfg.
{output_filename_format}_nba.csv	Non-billable Custom Sort Report (as applicable based on customsort.cfg settings). {output_filename_format} is defined in customsort.cfg.
{output_filename_format}_ssf.csv	Compliance Custom Sort Report. {output_filename_format} is defined in customsort.cfg.

Extended Format Only Report Files

File Name	Description
KML150117AA.csv	Request in comma delimited format.
KML150117AA_Default.csv	Request defaults in comma delimited format (as applicable).
KML150117AA _conversion.rpt	Conversion report. Generated if input file is in Excel format. Convert excel to comma delimited.
KML150117AA_dtl.rpt	Detailed log report. This matches SaaS Pro detailed log report. Need a PDF or excel example.
KML150117AA_error.rpt	Processing error report. Produced if errors occur during processing. Report captures the specific error message and the input line that failed processing.
KML150117AA_summary.rpt	Processing summary report.

4.3   KMLE150117AA.zip - Status Files

Extended and Fixed Format Status Files

File Name	Description
Read_err.sta	Reports problems encountered while reading the input file. Problems might relate to invalid transaction/service type pairings or an invalid input file.
Taxer.sta	Reports status during taxing routines. Status messages pertaining to successful or unsuccessful insertion of overrides or jurisdiction not found.
Sorter.sta	Reports status during sorting routines. Status messages pertaining to successful or unsuccessful sort after taxation has occurred.

Extended Format Only Status Files	
File Name	Description
KML150117AA_converter.sta	Applies to excel input files. If an error occurs during conversion from Excel to csv format, this file will be generated with the error message.
KML150117AA_batch.sta	Status from batch processor. Outputs column mappings and any informational, warning or error level messages.

5. Download Files from the FTP Site using the Sample Script

The sample download script is written in PowerShell and the PowerShell utility must be installed on the machine in order to utilize this script.

Once a file has been uploaded to the Avalara FTP site, please allow time for the system to process the input transaction package and prepare the results.

5.1 Setup

In order to perform the actions in this section, the logged-in user account must have **Administrator** access to the machine on which this script is being run.

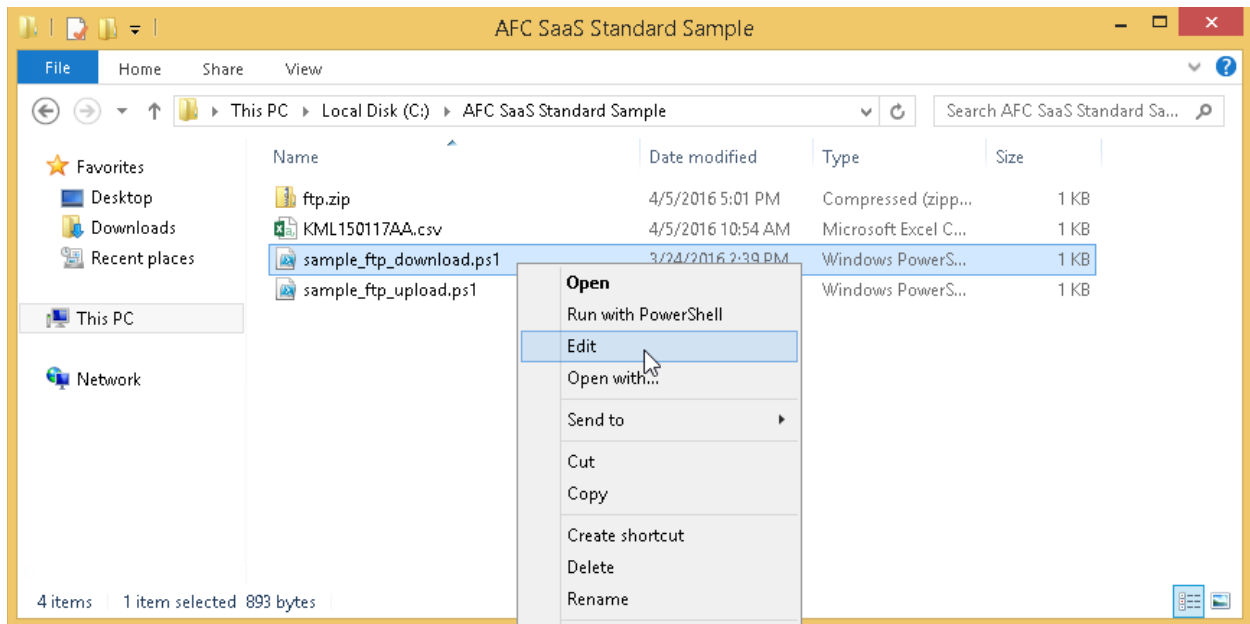To begin, right-click on the sample_ftp_download.ps1 file and select Edit to open the script in PowerShell.

Figure 12 – Editing sample_ftp_download.ps1

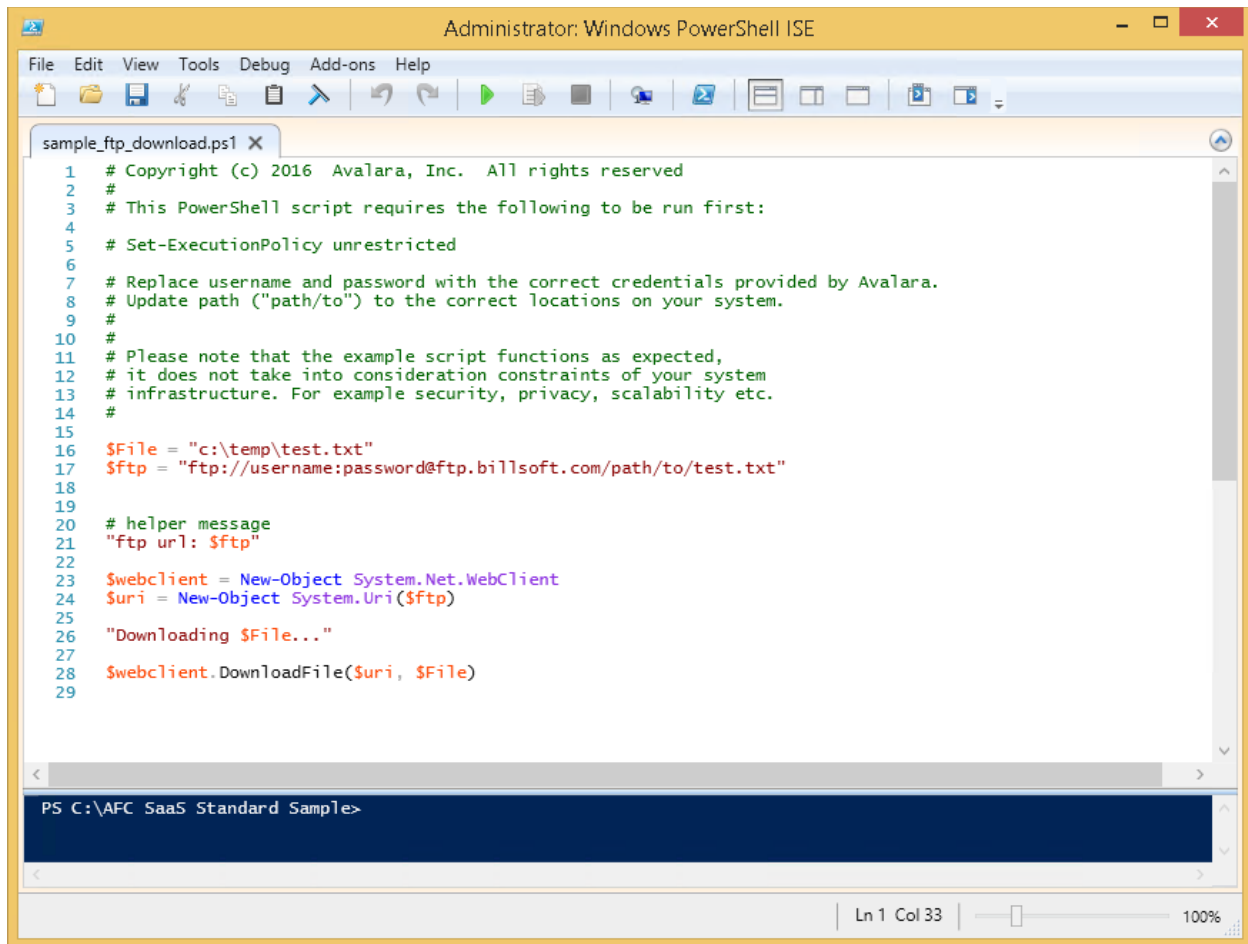The PowerShell program is opened.

Figure 13 - Opening the PowerShell Program

5.1.1   Set-ExecutionPolicy unrestricted Command

Prior to first use, the following command may need to be run as an Administrator.

```
Set-ExecutionPolicy unrestricted
```

This command is also provided as part of the sample_ftp_download.ps1 script. To run from the sample_ftp_download.ps1 script, highlight the command without the leading # sign and click on the **Run Selection** button within PowerShell.
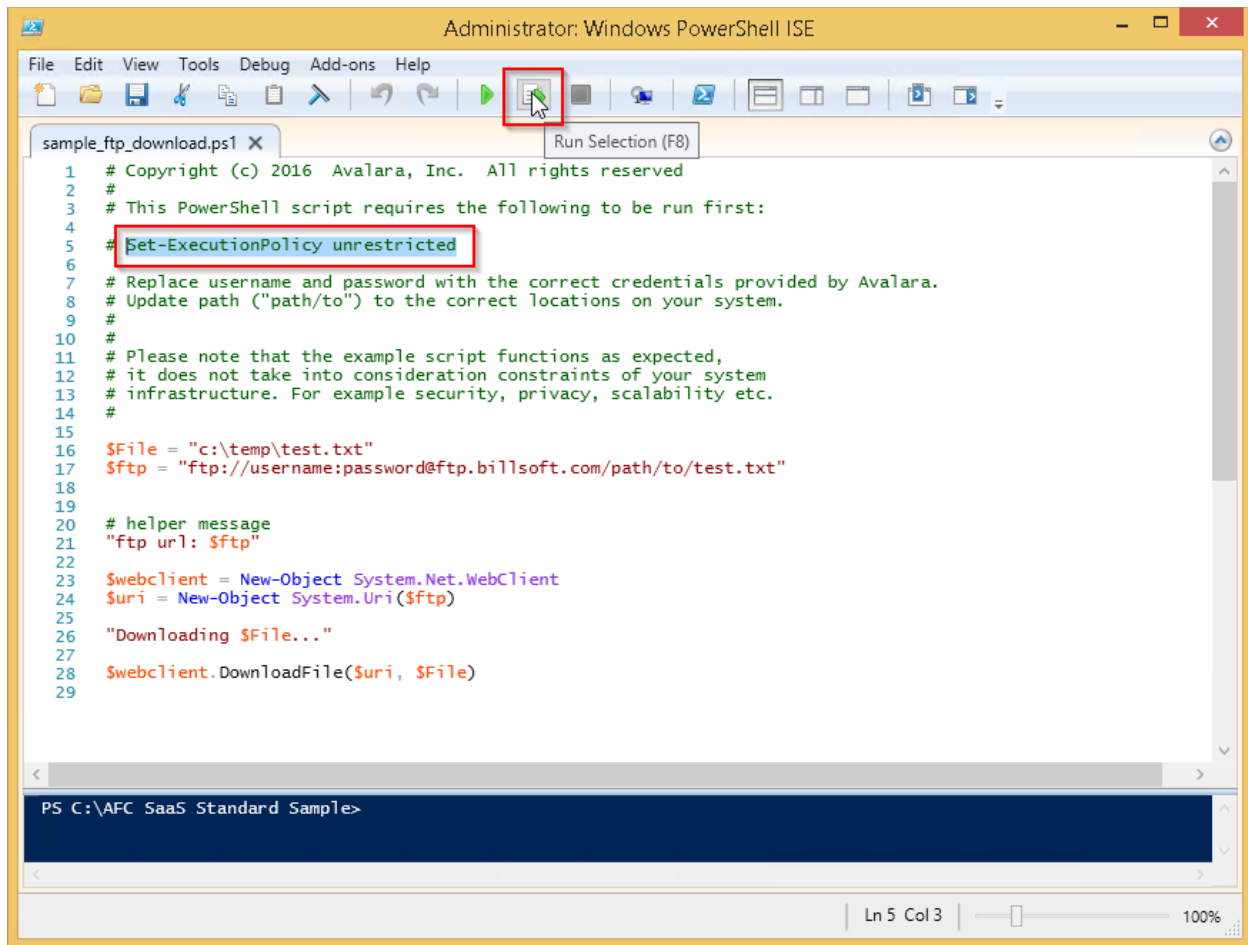
Figure 14 - Running the Set-ExecutionPolicy Command

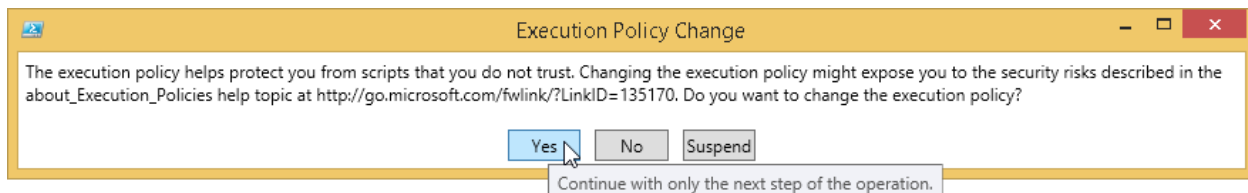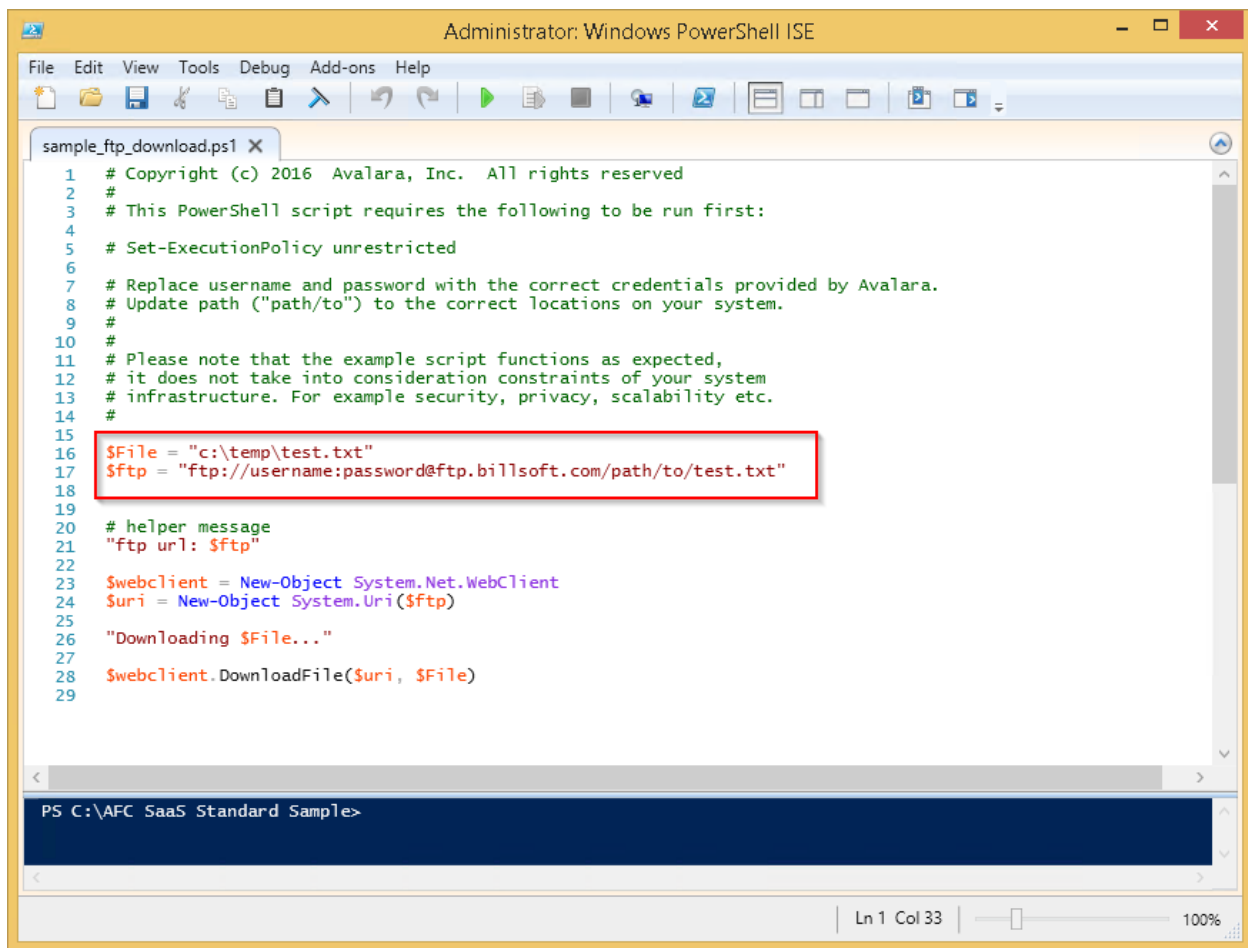The following question is displayed. Click **Yes** in the dialog box to continue.

Figure 15 - Confirming the Execution Policy Change

5.1.2 Update the File Locations
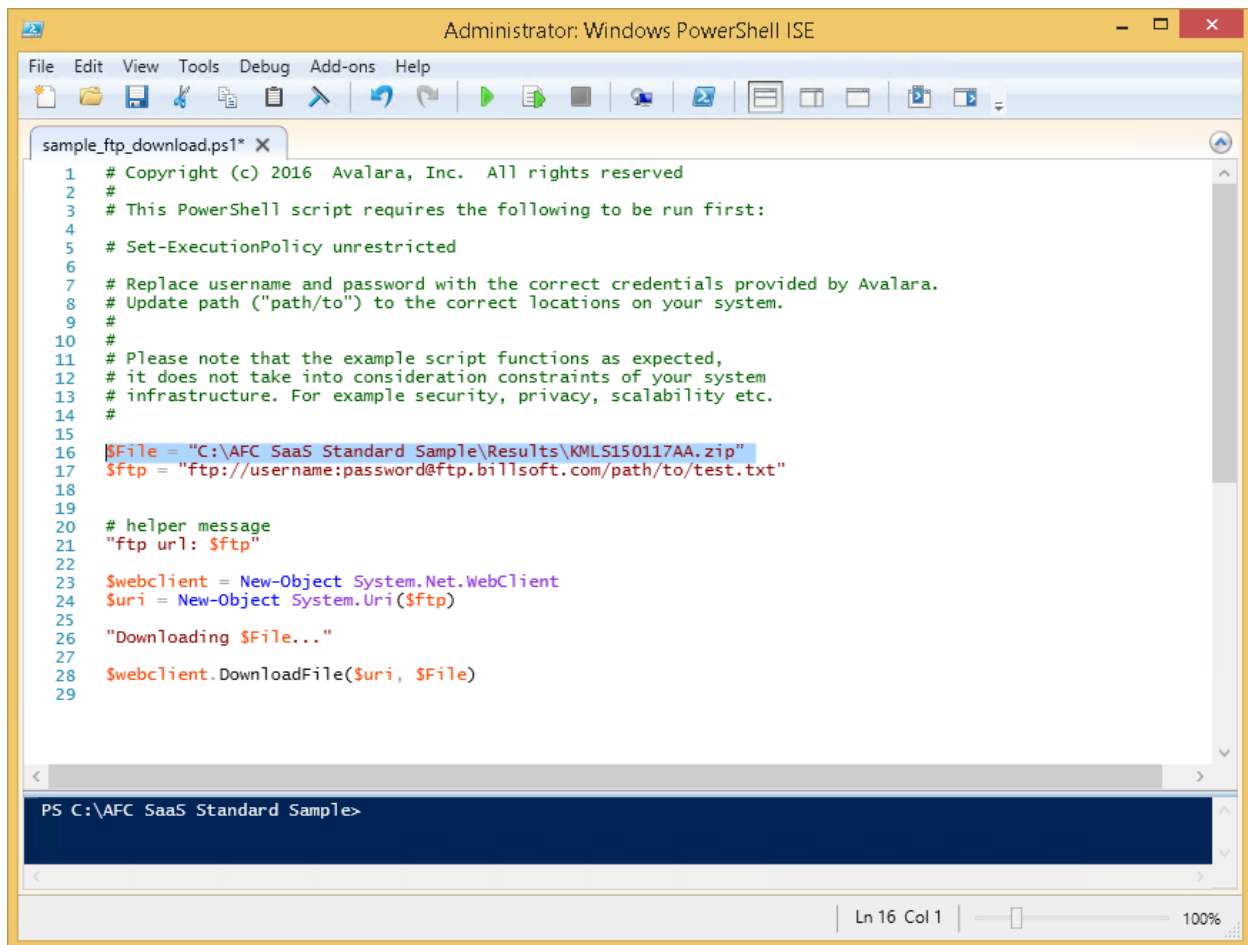
The next step is to update the file locations.

Figure 16 - Updating File Locations

5.1.2.1 $File variable

This variable specifies the path to which the output file shall be downloaded on the client-side network along with the filename to be downloaded. In this example, the KMLS150117AA.zip Compliance File results zip file will be downloaded to the Results file under C:\AFC SaaS Standard Sample.

```
$File = "C:\AFC SaaS Standard Sample\Results\KMLS150117AA.zip"
```

Note: The file path must be enclosed in quotation marks as shown above.
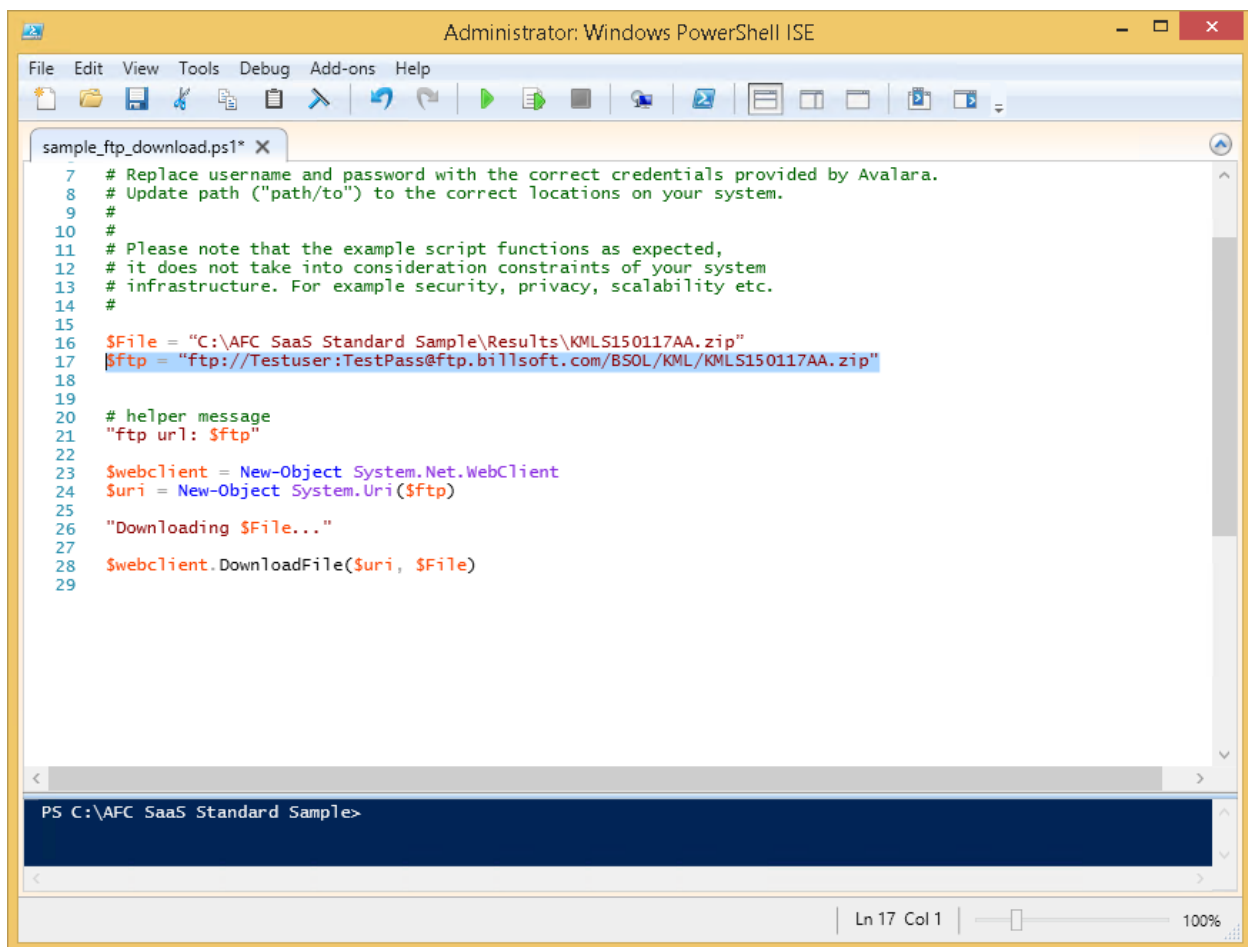

Figure 17 - $File Variable

5.1.2.2 $ftp variable

This variable defines the connection to the FTP server with the Avalara-provided FTP username, Avalara-provided FTP password, and download path specified. In this example, the following information is being set:

Field	Value
Username	Testuser
Password	TestPass
FTP Path	ftp.billsoft.com/BSOL/KML/KMLS150117AA.zip

```
$ftp = "ftp://Testuser:TestPass@ftp.billsoft.com/BSOL/KML/KMLS150117AA.zip"
```

Note: The file path must be enclosed in quotation marks as shown above.

```
7    # Replace username and password with the correct credentials provided by Avalara.
8    # Update path ("path/to") to the correct locations on your system.
9    #
10   #
11   # Please note that the example script functions as expected,
12   # it does not take into consideration constraints of your system
13   # infrastructure. For example security, privacy, scalability etc.
14   #
15
16   $File = "C:\AFC SaaS Standard Sample\Results\KMLS150117AA.zip"
17   $ftp = "ftp://Testuser:TestPass@ftp.billsoft.com/BSOL/KML/KMLS150117AA.zip"
18
19
20   # helper message
21   "ftp url: $ftp"
22
23   $webclient = New-Object System.Net.WebClient
24   $uri = New-Object System.Uri($ftp)
25
26   "Downloading $File..."
27
28   $webclient.DownloadFile($uri, $File)
29
```

Figure 18 - $ftp Variable

5.2   Execute the Script

Executing the sample_ftp_download.ps1 file will download the file specified in the $File variable from the FTP location specified using the username and password included in the $ftp variable. To execute the script, simply click the **Run Script** button within PowerShell.
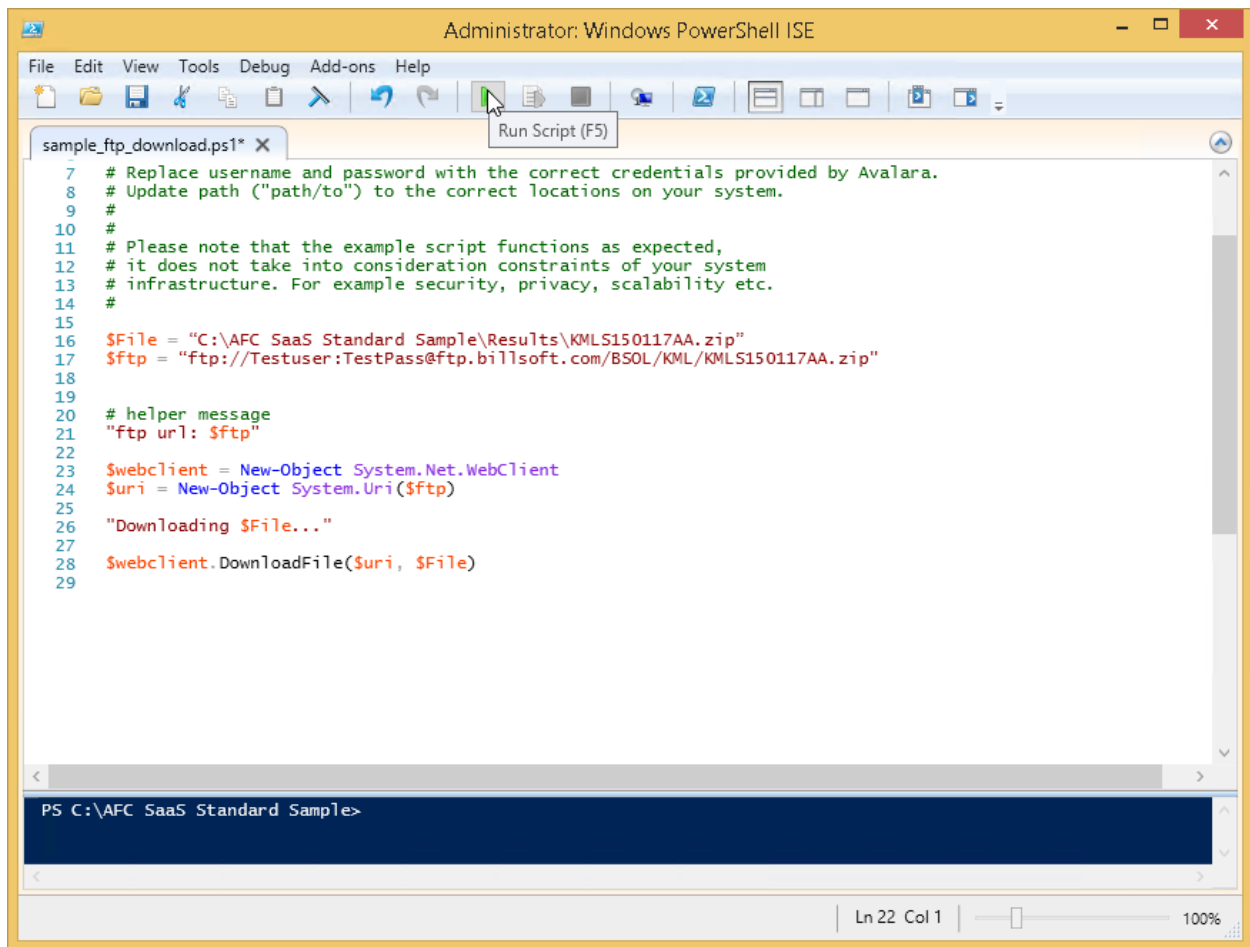

Figure 19 - Executing the Script

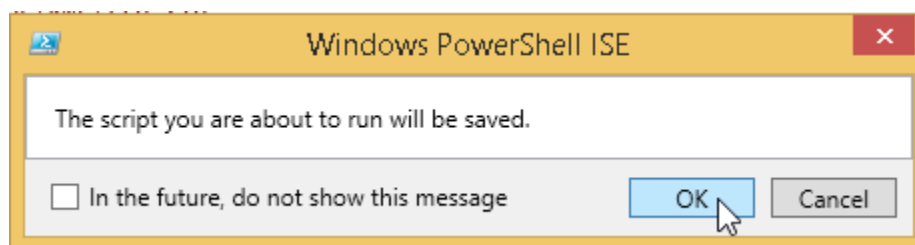A prompt may be displayed stating that the updates made to the script will be saved. Click **OK**.

Figure 20 – Saving Updates to the Script

The output section of PowerShell shows the results of the action:

```
PS C:\AFC SaaS Standard Sample> C:\AFC SaaS Standard Sample\sample_ftp_download.ps1
ftp url: ftp://Testuser:TestPass@ftp.billsoft.com/BSOL/KML/KMLS150117AA.zip
Downloading C:\AFC SaaS Standard Sample\Results\KMLS150117AA.zip...
```

Figure 21 - Results of Script

The zip file has been downloaded to the specified path.
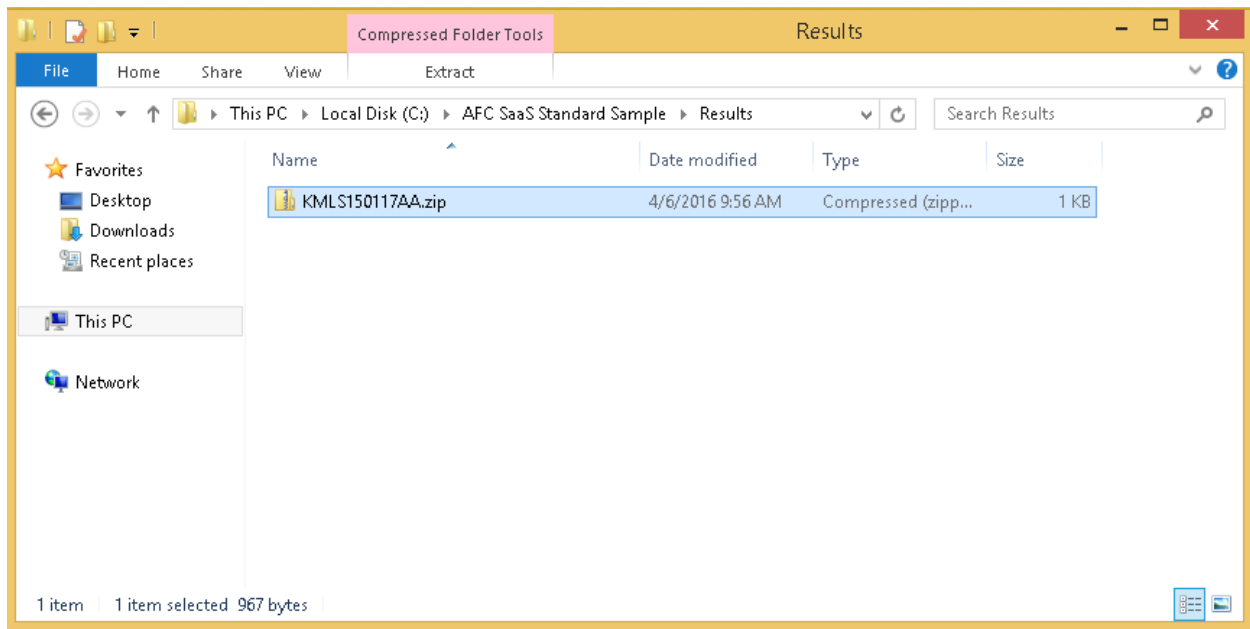
Figure 22 - Completion of Download

Note: Any error messages that may be returned can be found in the Results section of PowerShell.

5.3   Download Remaining Files

To download the Report output Zip file and the Status output Zip file, follow the steps found in **Section 5.1.2 – Update the File Locations** but replace the filename with the desired filenames for the two remaining Zip files. Execute the script two more times by following the steps in **Section 5.2 – Execute the Script**.

5.3.1   KMLR150117AA.zip

For this example, the sample_ftp_download.ps1 is modified as follows and then executed in order to download the Results output Zip file to the C:\AFC SaaS Standard Sample\Results file location.

```
$File = "C:\AFC SaaS Standard Sample\Results\KMLR150117AA.zip"
$ftp = "ftp://Testuser:TestPass@ftp.billsoft.com/BSOL/KML/KMLR150117AA.zip"
```

5.3.2   KMLE150117AA.zip

For this example, the sample_ftp_download.ps1 is modified as follows and then executed in order to download the Status output Zip file to the C:\AFC SaaS Standard Sample\Results file location.

```
$File = "C:\AFC SaaS Standard Sample\Results\KMLE150117AA.zip"
$ftp = "ftp://Testuser:TestPass@ftp.billsoft.com/BSOL/KML/KMLE150117AA.zip"
```