



## Programmatic Access to Market Chameleon Data Feeds

### Overview

This document describes a number of approaches that allow a developer to create a software process that downloads a data feed file to disk or memory. After downloading, the data feed can be processed according to the subscriber's needs.

Once created, the download process can be run on demand or scheduled to run automatically at a specific time of the day using a scheduler such as cron (Linux), launchd (MacOS) or Windows Task Scheduler. Configuring the scheduler is outside the scope of this document – please consult documentation for the scheduling utility available in your operating system.

### Disclaimer

The information in this document is provided “As Is”, without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose and non-infringement. In no event shall the authors or copyright holders be liable for any claim, damages or other liability, whether in an action of contract, tort or otherwise, arising from, out of or in connection with the information.

### Contact Info

All questions and feedback regarding this document should be directed to [data@marketchameleon.com](mailto:data@marketchameleon.com)

### Background

Market Chameleon data feeds are available for download from a service called ShareFile which is owned by Citrix. Access to ShareFile requires an internet connection. Data feed subscribers are provided with login credentials to access ShareFile. The Market Chameleon administrator will set up ShareFile access for all subscribers using the subscriber's email address which is provided by the subscriber on the subscription agreement. ShareFile will send an email to the subscriber that contains a temporary password and directions for setting up a permanent password. During the sign-on process, ShareFile will require the end-user to accept the Terms of Use. There is no additional charge for ShareFile access.

Prior to automating the download process, the end-user should log into <https://marketchameleon.sharefile.com> using a web browser and accept the ShareFile user agreement.

Broadly-speaking, ShareFile exposes data files using these industry-standard protocols:

- FTP / FTPS (Implicit SSL/TLS only, SFTP not supported)
- HTTPS API
- REST API
- WebDAV – primarily used for mapping ShareFile as an external drive, will not be discussed in this document. To map an external drive using WebDAV, consult the “Manual Access to Feeds” document.

There are a large number of tools that implement one or more of these protocols. This document discusses several, but not all of them. The tools fall broadly into two categories: command-line utilities and programming languages and it follows that the rest of the document is split up into these two sections. Using a command-line utility is suggested if a data file is to be downloaded to disk and processed later. A high-level programming language is desirable if any processing or saving to a database is needed at the time of download.

There is a separate document titled “User Access to Market Chameleon Data Feeds” which provides examples for how an end-user can download Market Chameleon data feeds to the desktop using applications such as a web browser or an FTP client.

### Directory Structure

The data feed files are organized within directories – one directory for each feed. Inside each feed directory there are one or more .csv or .zip files that contain the data.

mc\_feeds

```

|
|_ prod
|   |
|   |_ dividend_announcement
|   |_ dividend_forecast
|   |_ dividend_guidance
|   |_ earnings
|   |_ option_trades

```

### Command Line Utilities

	<b>Windows</b>	<b>Macintosh</b>	<b>Linux</b>
<b>WinSCP</b>	X		
<b>cURL</b>	X	X	X
<b>ShareFile PowerShell SDK</b>	X		

### WinSCP

WinSCP is a free utility licensed under GNU GPL that supports SFTP, FTP, FTPS and SCP. It operates as a GUI, from the command line or can be used as a library. It can be downloaded here: <http://winscp.net/eng/download.php>. After you’ve installed WinSCP, it’s necessary to build a script that performs the download. Here is a template:

```
open
ftps://marketchameleon\{email}:{password}@marketchameleon.sharefileftp
.com/
cd /mc_feeds/prod/earnings
lcd {local_directory}
get earnings.csv
exit
```

Execute the script using this command:

```
WinSCP.exe /log=WinSCP.log /script=script.txt
```

### cURL

cURL is typically used in Linux and OSX environments, and it is not included in a Windows installation. However it has been ported to Windows as part of CygWin or you can download the 32 bit standalone here: <https://curl.haxx.se/dlwiz/?type=bin&os=Win32> and the 64-bit standalone here: <https://curl.haxx.se/dlwiz/?type=bin&os=Win64>.

Two protocols can be used with cURL to download data:

- FTPS:

```
curl --ssl --user marketchameleon/{email}:{password}
ftps://marketchameleon.sharefileftp.com:990/mc_feeds/prod/earnings/earnings.csv --output earnings.csv
```

- REST API: please consult these examples: <http://api.sharefile.com/rest/samples/curl.aspx>

### ShareFile PowerShell SDK

If you're comfortable with Microsoft PowerShell scripting, this may be a good choice for you. This SDK is open source and maintained by ShareFile. It can be installed using the provided installer or manually and requires PowerShell 4.x and .NET 4.0 or higher.

Installer Download: <https://github.com/citrix/ShareFile-PowerShell/releases>

Manual Installation Instructions: <https://github.com/citrix/ShareFile-PowerShell/wiki/Manual-Installation>

Getting started guide: <https://www.citrix.com/blogs/2014/05/16/getting-started-with-the-powershell-sdk/>

Important note: Unlike the .Net and Java SDKs which require a user id/email, password, client ID and client secret to authenticate, the PowerShell SDK has client ID and client secret built-in so it requires user id/email and password only. The credentials are encrypted and stored in a file on disk specified by the user that has a ".sfps" extension by convention. The file is created by executing this command

within PowerShell which pops-up a GUI that contains a ShareFile login screen that prompts the user to enter an email and a password:

```
New-SfClient -Name <mycredentialsfile> -Account marketchameleon
```

To retrieve the session information in your scripts, use:

```
$sfClient = Get-SfClient <mycredentialsfile>
```

After that, \$sfClient contains the connection object which can be used to download files according to this example: <https://github.com/citrix/ShareFile-PowerShell/blob/master/Samples/DownloadFiles.ps1>

### Programming Languages

	<b>FTP / FTPS</b>	<b>HTTPS</b>	<b>REST</b>	<b>ShareFile SDK</b>
<b>.Net</b>	X	X	X	X
<b>Java</b>	X	X	X	X
<b>Python</b>	X	X	X	
<b>PHP</b>	X	X	X	
<b>Ruby</b>	X	X	X	
<b>Visual Basic for Applications (VBA)</b>	X			

### FTP / FTPS

Use of the FTP protocol is discouraged because it opens the possibility of a 3<sup>rd</sup> party intercepting unencrypted user ids and passwords. Emphasis is placed on using FTPS.

- .Net - The .Net framework supports FTP protocol in the System.Net.FTPWebRequest class, however there is no support for FTPS (implicit mode). One possible work-around is to use the WinSCP library <http://winscp.net/eng/docs/library> which is available on Nuget.
- Java -The Java class org.apache.commons.net.ftp.FTPSClient documented at <http://commons.apache.org/proper/commons-net/apidocs/org/apache/commons/net/ftp/FTPClient.html> supports FTPS (implicit mode).
- Python - The Python ftplib module <http://docs.python.org/library/ftplib.html#module-ftplib> supports FTPS (implicit).
- PHP – The ftps:// protocol wrapper can be used to download a file. It is documented here: <http://us.php.net/manual/en/wrappers.ftp.php> .
- Ruby – The Ruby Net::FTP module does not support FTPS. One possible work-around is to use DoubleBagFTPS <https://github.com/bnix/double-bag-ftp> which extends the Net::FTP functionality to support FTPS (implicit).
- VBA – It's possible to wrap the WinSCP .Net library in a COM wrapper and register it for use in VBA - [http://winscp.net/eng/docs/library\\_vb](http://winscp.net/eng/docs/library_vb) . Alternatively use the built-in wininet.dll documented at <https://support.microsoft.com/en-us/kb/259100> but keep in mind that this supports FTP only (no FTPS).

## HTTPS

Documentation and code examples are found here: <http://api.sharefile.com/https/https.aspx>

When the code calls for a “Top-Level domain” or “tld”, please use “sharefile.com”. The sub-domain is “marketchameleon”.

## REST

Documentation for the REST API can be found here: <http://api.sharefile.com/rest/> . Code examples can be accessed from the menu choices on the left.

The REST API uses OAuth2 authentication which requires in addition to UserId/Password two additional parameters – “ClientId” and “ClientSecret” which constitute the API Key. The API Key will be provided to the subscriber by the Market Chameleon administrator upon request.

## ShareFile SDK

Documentation for the ShareFile SDK and links to the download pages for the .Net and Java implementations can be found here: <http://api.sharefile.com/rest/clients.aspx> . Code examples are provided alongside the download pages.

The SDK uses OAuth2 authentication which requires in addition to UserId/Password two additional parameters – “ClientId” and “ClientSecret” which constitute the API Key. The API Key will be provided to the subscriber by the Market Chameleon administrator upon request.