

Forum:[Hack Forums](#)

Thread: [Powershell script to download multiple files through FTP](#)

Beigevenus - February 23, 2018, 2:15 pm

Greetings HF, my newest project has left me puzzled.

I need a Powershell script which allows me to download multiple files from an FTP server and I've honestly spent hours trying to find a solution, however, the script needs to be very specific to work in my project:

It needs to work without importing anything that is not a basic part of Powershell, which means no additional cmdlets and such.

The connection is FTP over TLS which needs to be taken into consideration, even though this is usually not a problem in Powershell.

The filenames of the files to be downloaded, and the amount of them is unknown, so the script will need to either download the entire directory or every single file within it with no regard to names and file types.

Does anyone know how to accomplish this? I would really appreciate your help

AshleyUK - February 24, 2018, 1:19 pm

(Feb 23, 2018 03:15 PM)Beigevenus Wrote: [\[CITE\]](#) Greetings HF, my newest project has left me puzzled.

I need a Powershell script which allows me to download multiple files from an FTP server and I've honestly spent hours trying to find a solution, however, the script needs to be very specific to work in my project:

It needs to work without importing anything that is not a basic part of Powershell, which means no additional cmdlets and such.

The connection is FTP over TLS which needs to be taken into consideration, even though this is usually not a problem in Powershell.

The filenames of the files to be downloaded, and the amount of them is unknown, so the script will need to either download the entire directory or every single file within it with no regard to names and file types.

Does anyone know how to accomplish this? I would really appreciate your help

Would this work for you?

<http://www.systemcentercentral.com/download-multiple-files-to-remote-ftp-site-with-powershell/>

I honestly don't know why you cannot import any PS modules, I'm sure PS has an FTP module that would make this easier... I should really go on a PS course.

Code #

```
=====
=====
#
# Microsoft PowerShell Source File -- Created with SAPIEN Technologies PrimalScript 2009
#
# NAME: FTP Multiple File Download via PowerShell
# AUTHOR: Pete Zerger
```

```

# DATE : 9/13/2010
#
# COMMENT: This PowerShell script, which will download multiple files from a remote FTP server
#         is an extended version of the script found at these two
#         Extended to support multiple file download and input file
#
#         Internet locations:
#         http://powershell.com/cs/media/p/804.aspx
#
http://stackoverflow.com/questions/265339/whats-the-best-way-to-automate-secure-ftp-in-powershe
ll
#
# FUNCTION PARAMETERS:
# $Source - Source file and path on the remote FTP server
# $Target - Target directory and filename on the local system (where download will be placed)
# $UserName - Username for FTP login
# $Password - Password for FTP login
# $UseBinary - Determines whether binary is used. . Possible values $true or $false
# $UseSSL - Determines whether SSL will be used. Possible values $true or $false
#
# FILE LIST: The file list must be provided in the form of a comma-separated value
#             file with a value for $Source and $Target on each line. Note that the first line
#             is a header containing the column names
#             For example:
#             SourceFile,TargetFile
#             "ftp://MyFtpServer/FolderPath/File.txt","C:\temp\MyFile.txt"
#             "ftp://MyFtpServer/FolderPath/File2.doc","C:\temp\MyLocalFile.doc"
#
=====
=====

function Get-FTPFile ($Source,$Target,$UserName,$Password, $UseBinary, $UseSSL)
{

# Create a FTPWebRequest object to handle the connection to the ftp server
$ftprequest = [System.Net.FtpWebRequest]::create($Source)

# set the request's network credentials for an authenticated connection
$ftprequest.Credentials =
    New-Object System.Net.NetworkCredential($username,$password)

$ftprequest.Method = [System.Net.WebRequestMethods+Ftp]::DownloadFile
$ftprequest.UseBinary = $UseBinary
$ftprequest.KeepAlive = $UseSSL

# send the ftp request to the server
$ftpresponse = $ftprequest.GetResponse()

# get a download stream from the server response
$responsestream = $ftpresponse.GetResponseStream()

# create the target file on the local system and the download buffer
$targetfile = New-Object IO.FileStream ($Target,[IO.FileMode]::Create)

```

```
[byte[]]$readbuffer = New-Object byte[] 1024

# loop through the download stream and send the data to the target file
do{
    $readlength = $responsestream.Read($readbuffer,0,1024)
    $targetfile.Write($readbuffer,0,$readlength)
}
while ($readlength -ne 0)

$targetfile.close()
}

#####
# Example of calling the function to download multiple files
#####

#First, read the $Source and $Target values from the list
$List = Import-Csv C:\temp\list.txt

#Use a ForEach loop to process all lines in the source file
ForEach ($entry in $list){

#Set the values of all parameters of the function
#The files
$sourceuri = $($entry.SourceFile)
$targetpath = $($entry.TargetFile)
$user = "ftpuser"
$pass = "P@ssw0rd1!"
$Binary = $true
$SSL = $false

Write-Host $sourceuri
Write-Host $targetpath

Get-FTPFile $sourceuri $targetpath $user $pass $Binary $SSL

}

```

Beigevenus - February 26, 2018, 12:58 am

(Feb 24, 2018 02:19 PM)AshleyUK Wrote: [\[CITE\]](#) Would this work for you?

<http://www.systemcentercentral.com/download-multiple-files-to-remote-ftp-site-with-powershell/>

I honestly don't know why you cannot import any PS modules, I'm sure PS has an FTP module that would make this easier... I should really go on a PS course.

Code #

```
=====
=====
#
# Microsoft PowerShell Source File -- Created with SAPIEN Technologies PrimalScript 2009
#
# NAME: FTP Multiple File Download via PowerShell
# AUTHOR: Pete Zerger

```

```

# DATE : 9/13/2010
#
# COMMENT: This PowerShell script, which will download multiple files from a remote FTP server
#         is an extended version of the script found at these two
#         Extended to support multiple file download and input file
#
#         Internet locations:
#         http://powershell.com/cs/media/p/804.aspx
#
http://stackoverflow.com/questions/265339/whats-the-best-way-to-automate-secure-ftp-in-powershe
ll
#
# FUNCTION PARAMETERS:
# $Source - Source file and path on the remote FTP server
# $Target - Target directory and filename on the local system (where download will be placed)
# $UserName - Username for FTP login
# $Password - Password for FTP login
# $UseBinary - Determines whether binary is used. . Possible values $true or $false
# $UseSSL - Determines whether SSL will be used. Possible values $true or $false
#
# FILE LIST: The file list must be provided in the form of a comma-separated value
#             file with a value for $Source and $Target on each line. Note that the first line
#             is a header containing the column names
#             For example:
#             SourceFile,TargetFile
#             "ftp://MyFtpServer/FolderPath/File.txt","C:\temp\MyFile.txt"
#             "ftp://MyFtpServer/FolderPath/File2.doc","C:\temp\MyLocalFile.doc"
#
=====
=====

function Get-FTPFile ($Source,$Target,$UserName,$Password, $UseBinary, $UseSSL)
{

# Create a FTPWebRequest object to handle the connection to the ftp server
$ftprequest = [System.Net.FtpWebRequest]::create($Source)

# set the request's network credentials for an authenticated connection
$ftprequest.Credentials =
    New-Object System.Net.NetworkCredential($username,$password)

$ftprequest.Method = [System.Net.WebRequestMethods+Ftp]::DownloadFile
$ftprequest.UseBinary = $UseBinary
$ftprequest.KeepAlive = $UseSSL

# send the ftp request to the server
$ftpresponse = $ftprequest.GetResponse()

# get a download stream from the server response
$responsestream = $ftpresponse.GetResponseStream()

# create the target file on the local system and the download buffer
$targetfile = New-Object IO.FileStream ($Target,[IO.FileMode]::Create)

```

```
[byte[]]$readbuffer = New-Object byte[] 1024
```

```
# loop through the download stream and send the data to the target file
```

```
do{  
    $readlength = $responsestream.Read($readbuffer,0,1024)  
    $targetfile.Write($readbuffer,0,$readlength)  
}  
while ($readlength -ne 0)
```

```
$targetfile.close()  
}
```

```
#####  
# Example of calling the function to download multiple files  
#####
```

```
#First, read the $Source and $Target values from the list  
$List = Import-Csv C:\temp\list.txt
```

```
#Use a ForEach loop to process all lines in the source file  
ForEach ($entry in $list){
```

```
#Set the values of all parameters of the function
```

```
#The files
```

```
$sourceuri = $($entry.SourceFile)
```

```
$targetpath = $($entry.TargetFile)
```

```
$user = "ftpuser"
```

```
$pass = "P@ssw0rd1!"
```

```
$Binary = $true
```

```
$SSL = $false
```

```
Write-Host $sourceuri
```

```
Write-Host $targetpath
```

```
Get-FTPFile $sourceuri $targetpath $user $pass $Binary $SSL
```

```
}
```

Thank you very much, I will definitely check out that link and when it comes to why I can't import any modules, well, I'd prefer to keep the script as available as possible to all of the intended users, which would be difficult if every user had to import modules to PS before being able to use the script.

AshleyUK - February 26, 2018, 8:56 am

(Feb 26, 2018 01:58 AM)Beigevenus Wrote: [\[CITE\]](#) Thank you very much, I will definitely check out that link and when it comes to why I can't import any modules, well, I'd prefer to keep the script as available as possible to all of the intended users, which would be difficult if every user had to import modules to PS before being able to use the script.

No Problem, You could always take a look at what the module can do and then just using the code from the ps file - <https://gallery.technet.microsoft.com/scriptcenter/PowerShell-FTP-Client-db6fe0cb>

Beigevenus - March 28, 2018, 6:27 am

I ended up using WinSCP with a batch script to do the downloading for me, thank you all for your help!

