

Command Line File Management Scripts

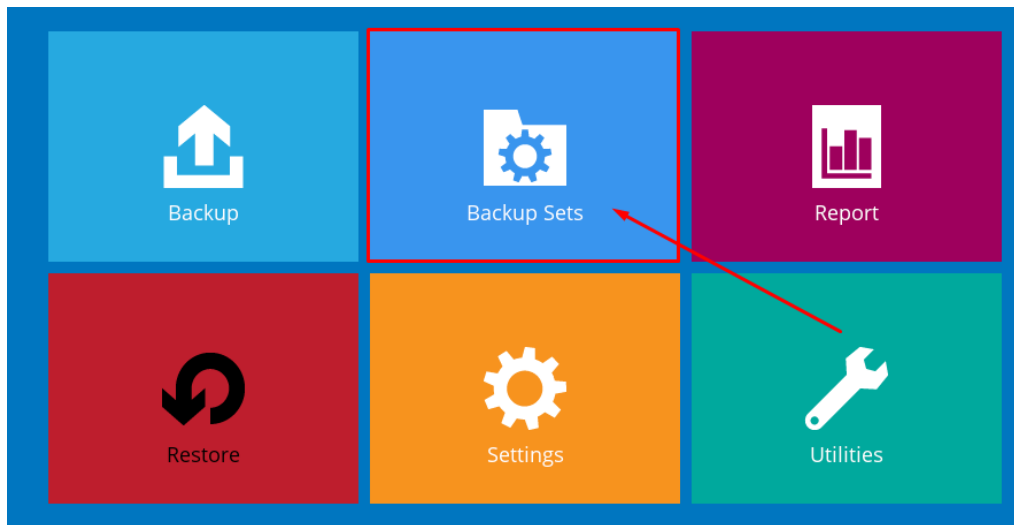
This guide provides **Powershell commands** to help you automate file organization, cleanup, and backup preparation. Whether you need to move old files, filter by size or type, or delete temporary data, these scripts ensure efficient storage management.

1. Basic File Aging Command

Moves files older than X days, preserving structure

```
powershell.exe -NoProfile -Command "$source = 'C:\Users\cypter\Documents\Bandicam'; $target = 'C:\Users\cypter\Documents\Bandicam_Old_Files'; Get-ChildItem -Path $source -Recurse -File | Where-Object { $_.LastWriteTime -lt (Get-Date).AddDays(-5) } | ForEach-Object { $relative = $_.FullName.Substring($source.Length); $dest = Join-Path $target $relative; $destDir = Split-Path $dest; if (-not (Test-Path $destDir)) { New-Item -ItemType Directory -Path $destDir -Force | Out-Null }; Move-Item $_.FullName -Destination $dest -Force }"
```

- Change `.AddDays(-X)` - days
- `$source` **Directory**
 - *Purpose:* The folder you want to clean up before backup
 - *Example:* `'C:\Users\cypter\Documents\Bandicam'`
- `$target` **Directory**
 - *Purpose:* Where unnecessary files will be moved
 - *Example:* `'C:\Users\cypter\Documents\Bandicam_Old_Files'`





File

test1

Owner: Router

Last Backup: Saturday, 21 June 2025 22:20



Add



test1

General

Source

Backup Schedule

Continuous Backup

Destination

Deduplication

Retention Policy

Command Line Tool

Reminder

Bandwidth Control

Others

General

Name

test1

Owner

Router

Windows User Authentication

Logon as this user during backup to acquire additional privileges

Off



Pre-Backup Command Line Tool

Name

Pre-Backup-1

Working Directory

C:\

Change

Command

powershell.exe -NoProfile -Command "\$source='C:\Users\


2. Size-Based Cleanup

A. Moves *LARGE* files (>X MB)

```
powershell.exe -NoProfile -Command "$source='C:\Users\cypter\Documents\
Bandicam'; $target='C:\Users\cypter\Documents\Bandicam_OverSize';
$minSizeMB=200; if (-not (Test-Path $target)) { New-Item -ItemType Directory -
Path $target -Force | Out-Null }; Get-ChildItem -Path $source -File | Where-
Object { $_.Length -gt ($minSizeMB * 1MB) } | ForEach-Object { Move-Item -Path
$_.FullName -Destination $target -Force -Verbose }"
```

B. Moves *SMALL* files (<X MB)

```
powershell.exe -NoProfile -Command "$source='C:\Users\cypter\Documents\
Bandicam'; $target='C:\Users\cypter\Documents\Bandicam_SmallFiles';
$minSizeMB=200; if (-not (Test-Path $target)) { New-Item -ItemType Directory -
Path $target -Force }; Get-ChildItem -Path $source -File | Where-Object { if
($_.Length -gt ($minSizeMB * 1MB)) { $false } else { $true } } | ForEach-
Object { Move-Item -Path $_.FullName -Destination $target -Force -Verbose }"
```

- Adjust `$minSizeMB` (e.g., `500` = 500MB)
- Same `$source/$target` rules as above

3. Extension-Based Cleanup

A. Keep Specific File Types

```
powershell.exe -NoProfile -Command "$source='C:\Users\cypter\Documents\Bandicam'; $target='C:\Users\cypter\Documents\Bandicam_new'; $keepExtensions=@('*.*jpg','*.*txt'); if (-not (Test-Path $target)) { New-Item -ItemType Directory -Path $target -Force }; Get-ChildItem -Path $source -Recurse -File | Where-Object { $_.Extension -notin ('.*jpg','.*txt') } | ForEach-Object { Move-Item -Path $_.FullName -Destination $target -Force -Verbose }"
```

B. Moves Specific File Types

```
powershell.exe -NoProfile -Command "$source='C:\Users\cypter\Documents\Bandicam'; $target='C:\Users\cypter\Documents\Bandicam_new'; $extensions=@('*.*jpg','*.*txt'); if (-not (Test-Path $target)) { New-Item -ItemType Directory -Path $target -Force }; Get-ChildItem -Path $source -Recurse -File -Include $extensions | Move-Item -Destination $target -Force"
```

- **Change:**
- `$extensions` array (e.g., `'*.*jpg','*.*png'`)
- Same `$source/$target` rules as above

Common Post-Backup Commands

1. Basic Cleanup

```
powershell.exe -Command "Get-ChildItem 'C:\Temp' -File | Where-Object { $_.LastWriteTime -lt (Get-Date).AddDays(-2) } | Remove-Item -Force"
```

- **Purpose:** Deletes files older than 2 days from Temp folder
- **Logging:** Append `>> C:\Logs\cleanup.log 2>&1` to save output

Command.

Params	Values	Purpose
<code>-Recurse</code>	N/A	Include subfolders
<code>-lt/-gt</code>	<code>AddDays (-7), 1GB</code>	Age/size limits
<code>Replace()</code>	<code>\$source, \$target</code>	Path rewriting
<code>Split-Path</code>	N/A	Get parent folder
<code>-Force</code>	N/A	Overwrite existing