PowerShell Folder ACL Scanner - User Guide

Overview

This PowerShell script scans a directory you specify, analyzes the Access Control Lists (ACLs) for every folder within it (including subfolders), and generates two report files:

- A detailed text report (`FolderACL_Report.txt`) showing the ACLs for each folder in a table format.
- A concise CSV file (`Folders_With_UsersGroup.csv`) listing only the *names* of folders that grant permissions to any standard 'Users' group (like `BUILTIN\Users` or a domain's Users group).

The text report highlights folders containing 'Users' group permissions for easy identification.

Requirements

- Windows Operating System
- PowerShell version 3.0 or later. (Check with `$PSVersionTable.PSVersion`)
- Sufficient permissions to read the folders and their ACLs in the target directory. Running PowerShell **as Administrator** is recommended, especially for system or protected folders.
- PowerShell Execution Policy might need adjustment. If the script doesn't run, you may need to run:
  `Set-ExecutionPolicy RemoteSigned -Scope Process -Force`
  (This sets the policy only for the current PowerShell session).

How to Use

1. **Save the Script:** Save the PowerShell script code to a file named (for example) `Get-FolderAcls.ps1` on your computer.
2. **Open PowerShell:** Open a PowerShell window. Right-click the PowerShell icon and select "Run as administrator" for best results.
3. **Navigate to Script Directory:** Use the `cd` command to change to the directory where you saved the script.

```
cd C:\Path\To\Your\Scripts
```

4. **Run the Script:** Execute the script by typing its path:

```
.\Get-FolderAcls.ps1
```

5. **Enter Target Directory:** The script will prompt you for the directory you want to scan. Enter the full path and press Enter.

```
Please enter the full path to the directory you want to scan:
C:\Users\Public\Documents
```

6. **Enter Output Directory:** The script will then prompt for the directory where the report files should be saved. Enter the full path and press Enter. If the directory doesn't exist, the script will attempt to create it.

```
Please enter the full path to the directory where the output files should be
saved: C:\Temp\ACL_Reports
```

7. **Wait for Completion:** The script will display progress as it scans folders. This may take time depending on the number and size of folders.

8. **Check Output Files:** Once finished, navigate to the output directory you specified (e.g., `C:\Temp\ACL_Reports`) to find the two generated files: `FolderACL_Report.txt` and `Folders_With_UsersGroup.csv`.

Script Behavior

- **Prompts:** Requires user input for the target scanning directory and the output report directory.
- **Directory Validation:** Checks if the target directory exists. Checks if the output directory exists; if not, it attempts to create it.
- **Recursion:** Scans the target directory and ALL subdirectories within it.

- **Progress Indicator:** Shows a progress bar in the PowerShell console indicating the percentage of folders processed.
- **Error Handling:** If it cannot access a folder's ACL (e.g., due to permissions), it logs a warning to the console and notes the error in the text report for that specific folder, then continues scanning other folders.
- **Completion Messages:** Displays messages indicating successful file generation or if no folders matching the criteria were found.

Output File Explanation

1. Text Report (`FolderACL_Report.txt`)

This file provides a detailed breakdown in a table format.

- **Header:** Shows the generation date/time and a legend explaining the highlight marker.
- **Table Structure:** Contains two main columns:
    - `FolderName` : The name of the folder (not the full path). Folders that have any 'Users' group listed in their ACL will have an asterisk (`*`) appended to their name.
    - `ACL_Summary` : A concise, multi-line summary of the Access Control Entries (ACEs) for that folder. Each ACE shows:
        - User/Group Identity (e.g., `BUILTIN\Administrators`)
        - Permissions (e.g., `FullControl` , `ReadAndExecute, Synchronize`)
        - Type (`Allow` or `Deny`)
        - Inheritance (`Inherited` or `Explicit`)

      Text within this column will wrap if it's too long for the table width.
- **Errors:** Folders where ACL retrieval failed will show an error message in the `ACL_Summary` column.

2. CSV Report (`Folders_With_UsersGroup.csv`)

This file provides a simple, machine-readable list.

- **Content:** Contains a single line of text.
- **Format:** The names (**only the names**, not full paths) of all folders found to have any 'Users' group (e.g., `BUILTIN\Users`, `DOMAIN\Users`, `COMPUTERNAME\Users`) in their ACL.
- **Delimiter:** Folder names are separated by a comma (`,`).
- **No Header:** The file does not contain a header row.
- **Empty File:** If no folders with 'Users' group permissions are found, this file will be created but will be empty.

Examples

Example Interaction

```
PS C:\Scripts> .\Get-FolderAcls.ps1

Please enter the full path to the directory you want to scan: D:\SharedData
Please enter the full path to the directory where the output files should be saved:
C:\ACL_Reports

Scanning folders under 'D:\SharedData'...
Found 356 folders to process.
Processing Folder ACLs
Processing folder 357 of 357: D:\SharedData\Archive\OldProjects
[============================================================] Completed

Generating text report table...
Text report generated successfully: C:\ACL_Reports\FolderACL_Report.txt
Exporting list of folder names with 'Users' group access to CSV...
CSV file generated successfully: C:\ACL_Reports\Folders_With_UsersGroup.csv
Script finished.
PS C:\Scripts>
```

Example Output: `FolderACL_Report.txt` (Snippet)

```
Folder ACL Report - Generated on 07/27/2023 15:45:00
Note: Folders marked with '*' in the FolderName column have a 'Users' group in their ACL.
--------------------------------------------------


FolderName             ACL_Summary
----------             -----------
SharedData             NT AUTHORITY\SYSTEM : FullControl (Allow, Explicit)
                       BUILTIN\Administrators : FullControl (Allow, Explicit)
                       CREATOR OWNER : FullControl (Allow, Inherited)
                       DOMAIN\Domain Admins : FullControl (Allow, Inherited)
Public *               NT AUTHORITY\SYSTEM : FullControl (Allow, Inherited)
                       BUILTIN\Administrators : FullControl (Allow, Inherited)
                       BUILTIN\Users : ReadAndExecute, Synchronize (Allow, Inherited)
                       BUILTIN\Users : AppendData (Allow, Inherited)
                       BUILTIN\Users : CreateFiles (Allow, Inherited)
Restricted             NT AUTHORITY\SYSTEM : FullControl (Allow, Inherited)
                       BUILTIN\Administrators : FullControl (Allow, Inherited)
                       DOMAIN\Managers : Modify, Synchronize (Allow, Explicit)
FailedFolder           ERROR: Could not retrieve ACL - Access is denied.
```

Example Output: `Folders_With_UsersGroup.csv`

```
Public,UserData,CommonFiles
```

(Note: The CSV content is all on a single line, with no spaces after commas unless the folder name itself contains a space).

Troubleshooting & Notes

- **Access Denied Errors:** If you see warnings or errors about being unable to retrieve ACLs, ensure you are running PowerShell "As Administrator". Some system folders may still be inaccessible.
- **Script Won't Run (Execution Policy):** If you get an error message about scripts being disabled, run PowerShell as Administrator and execute `Set-ExecutionPolicy RemoteSigned -Scope Process -Force` before running the script again.
- **Large Directories:** Scanning directories with many thousands of folders can take a significant amount of time and consume memory.
- **'Users' Group Definition:** The script looks for any ACL entry where the identity name *ends with* `\Users` (case-insensitive). This covers local users (`COMPUTERNAME\Users`), domain users (`DOMAIN\Users`), and the built-in group (`BUILTIN\Users`).