User Guide: Folder ACL Inheritance & Users Group Removal Script

1. Purpose

This PowerShell script modifies the security settings (Access Control Lists - ACLs) of specific folders. For each folder you specify in an input file, it performs the following actions:

- **Disables Inheritance:** If ACL inheritance is enabled on the folder, the script disables it. It converts the inherited permissions into explicit permissions on the folder itself.
- **Removes 'Users' Group:** It removes *all* permissions granted to the built-in 'Users' group (either the local `BUILTIN\Users` or the domain/computer `DOMAIN\Users` or `COMPUTERNAME\Users` group, whichever is appropriate).

The script processes only the folders listed in your input file and logs its actions and a final summary to an output text file.

2. Prerequisites

- **Windows Operating System:** PowerShell is required, which is standard on modern Windows versions.
- **Permissions:** You MUST run this script with an account that has **Full Control** permissions (or at least the rights to **Modify Permissions** and potentially **Take Ownership**) on all the folders you intend to modify. Running PowerShell **as an Administrator** is highly recommended.

3. Input Preparation

You need to create a simple text file (`.txt`) that lists the names of the folders you want to process.

- **Format:** Plain text file.
- **Content:** List the *exact names* of the folders, separated by commas (`,`).
    - **Do NOT include the parent path** in the file, only the folder names themselves as they appear *inside* the parent directory.
    - Whitespace around the commas doesn't matter (e.g., `FolderA, FolderB` is fine).
    - Ensure there are no empty entries caused by extra commas (e.g., `FolderA,,FolderB`).
- **Example:**

  Let's say you have a parent directory `C:\Projects` containing folders named `Project Alpha` , `Shared Files` , and `Old Archives` . You only want to modify `Project Alpha` and `Shared Files` .

  Create a text file named (for example) `folders_to_process.txt` with the following content:

  ```
  Project Alpha,Shared Files
  ```

  (Save this file somewhere you can easily find, like `C:\Scripts\input\folders_to_process.txt`)

4. Running the Script

1. **Open PowerShell as Administrator:**
    - Search for "PowerShell" in the Start Menu.
    - Right-click "Windows PowerShell" and select "Run as administrator".
2. **Navigate to the Script Directory (Optional but Recommended):**
    - Use the `cd` command to change to the directory where you saved the PowerShell script file (e.g., `UsersGroup_Removal_Script.ps1`).
    - Example: `cd C:\Scripts`
3. **Execute the Script:**
    - Type `.\` followed by the script's filename and press Enter.
    - Example: `.\UsersGroup_Removal_Script.ps1`
4. **Answer the Prompts:** The script will ask for three pieces of information:
    - `Enter the path to the text file containing folder names (comma-separated) :`
      Provide the full path to the `.txt` file you created in Step 3.

Example: `C:\Scripts\input\folders_to_process.txt`

- `Enter the path to the parent directory containing these folders :`
  Provide the full path to the directory that *contains* the folders listed in your input file.
  Example: `C:\Projects`
- `Enter the path to the directory where the log file should be saved :`
  Provide the full path to an *existing* directory where you want the output log file to be created. The script will attempt to create this directory if it doesn't exist.
  Example: `C:\Scripts\logs`

The script will then start processing the folders and log its progress.

5. Understanding the Output (Log File)

- **Location:** The log file will be created in the output directory you specified during the prompts.
- **Filename:** The filename will include the date and time the script was run, making each log unique (e.g., `UsersGroup_Removal_log_20231027_153000.txt`).
- **Content:**
  - **Timestamped Logs:** Detailed entries showing what the script is doing for each folder (checking existence, getting ACL, disabling inheritance, removing Users group). Messages are prefixed with `[INFO]`, `[WARN]`, or `[ERROR]`.
  - **Summary Table:** At the very end of the log file, you'll find a table summarizing the results for each folder attempted.
- **Summary Table Columns:**
  - `FolderPath` : The full path to the folder that was processed.
  - `Status` : The overall outcome for that folder:
    - `Success` : Inheritance was disabled (if needed) and the Users group was successfully removed (or wasn't present).
    - `Skipped` : The folder was skipped (e.g., it didn't exist).
    - `Error` : An error occurred during processing that prevented completion for this folder.
  - `Details` : More specific information about the status, especially for errors or warnings (e.g., "Inheritance was already disabled", "Failed to read ACL", "Users group ACEs removed/verified absent").
- **Example Log File Snippet (Focus on Summary):**

```
... (detailed logs above) ...
2023-10-27 15:31:15 [INFO] --- Summary ---

FolderPath        Status  Details
----------        ------  -------
C:\Projects\Project Alpha Success Inheritance disabled and ACL rebuilt. Users group ACEs removed/verified absent.
C:\Projects\Shared Files  Success Inheritance was already disabled. Users group ACEs removed/verified absent.
C:\Projects\NonExistent Skipped The directory 'C:\Projects\NonExistent' specified in the input file does not exist or is not a directory.
C:\Projects\LockedFolder Error   Failed read ACL before Users removal. Details: Attempted to perform an unauthorized operation.


2023-10-27 15:31:15 [INFO] Summary table generated.
2023-10-27 15:31:15 [INFO] Script finished.
```

6. Troubleshooting & Important Notes

- **Permissions Errors (`Unauthorized operation`):** Ensure you are running PowerShell as an Administrator and that the account has sufficient permissions on the target folders.
- **Folder Not Found Errors:** Double-check the `Parent Directory` path you provide. Verify the folder names in your input `.txt` file exactly match the names in the parent directory (sometimes case matters). Ensure the input file is plain text and correctly comma-separated.
- **Input File Issues:** Make sure the input file path is correct and the file is not empty or corrupted.
- > **Backup Recommended:** Modifying ACLs is a significant change. It is **strongly recommended** to back up important data or test the script in a non-production environment first.
- **Users Group Identification:** The script tries to identify the correct 'Users' group (domain/computer-specific first, then local `BUILTIN\Users`). In unusual configurations (e.g., workgroups, specific AD setups), SID resolution might fail. Check the log file for warnings related to SID resolution.