

# CSV Files

# Technical Guide

PolicyHub All Versions

# Contents

---

- CHAPTER 1: CSV IMPORT PROCESS..... 1-1**
- Overview ..... 1-1
- Features of Using CSV Files..... 1-2
- Users.csv..... 1-5
- Groups.csv ..... 1-6
- GroupMembership.csv..... 1-7
- UserMembership.csv ..... 1-9
- UserHierarchy.csv (Optional) ..... 1-11
- Globally Unique IDs..... 1-12
- User/Group Import and Synchronisation Process ..... 1-14

# Chapter 1: CSV Import Process

## Overview

### Introduction

---

An alternative to using the **Active Directory** synchronisation to populate PolicyHub with users and groups, **CSV** files can be implemented. There are three ways these files can be populated with the details:

- Using the PolicyHub Membership Populator Service
- Manual administration
- Automatic creation by another system (e.g. HR system)

The CSV files can also be used in conjunction with the **Active Directory** synchronisation to create and populate additional groups which are not available in the customer's Active Directory.

# Features of Using CSV Files

## Import Process

The CSV file import process has these features:

Uses a simple set of CSV files to populate PolicyHub

- Can exist alongside an existing Active Directory import
- Can completely replace the Active Directory import
- Can be set up to generate the files automatically
- Population of the files can be a manual process
- Allows complete control over which users and Groups are added to PolicyHub

There are five CSV files which must be in place (even if some of them are empty):

- Users.csv
- Groups.csv
- GroupMembership.csv
- UserMembership.csv
- UserHierarchy.csv (optional)



### Note:

Each value in the file must be separated by a comma (without extra spaces) by default. Each row is separated by a **Line Feed (LF)** character and a **Carriage Return (CR)** character.

For example:

```
Groups.csv
1 // - GroupName, DisplayName, Description, GroupId
2 Administrators, Administrators, Administrators, 94D56F9E-A4E0-4F91-B010-668DF77CA785
3 Supervisors, Supervisors, Supervisors, 27713CF9-243F-4DDC-9B2F-74CF448413A0
4 Managers, Managers, Managers, 008D5411-49A2-4B42-8591-2AA1EB8B8F0E
5 Operators, Operators, Operators, A44631BA-893D-4E1D-90C4-D453A0911D78
6 HR, HR, HR, 1A9F9077-A004-4B60-AE8E-620B59917D14
7 Admin, Admin, Admin, 91E5C458-AFDD-476A-8FC9-D668DBC2265
8 Operations, Operations, Operations, 66511C29-A533-43D7-9536-7EA8B14BCFDB
```

It has been observed that when the CSV files are edited in some environments the program that is used may not insert the **Carriage Return** character at the end of the line. If the **CR** character is missing it will cause a problem when synchronising the files in PolicyHub.

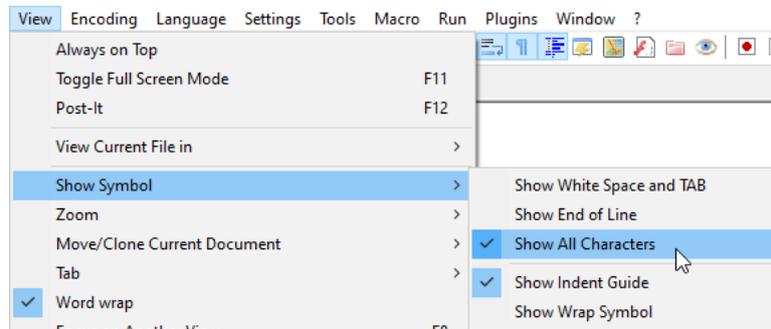
Whilst this is not usually an issue in a Windows environment this behaviour has been observed in files that have been edited in a Mac OS environment. The topics below describe how a User may check the files to ensure that the **CR** character is present and how to insert the character if it is missing.

## Features of Using CSV Files (cont.)

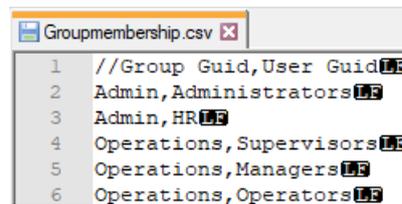
### CSV File Editing

Several programs can be used to edit the CSV files, for example, Notepad, Notepad++ and Excel. Care must be taken when adding new lines to the file so that both the Line Feed and the Carriage Return characters are at the end of the line.

The hidden characters can be viewed in Notepad++. Select **View » Show Symbol » Show All Characters** from the menu.



If the **Carriage Return** character has not been entered the **CR** hidden character will not be visible at the end of the line.

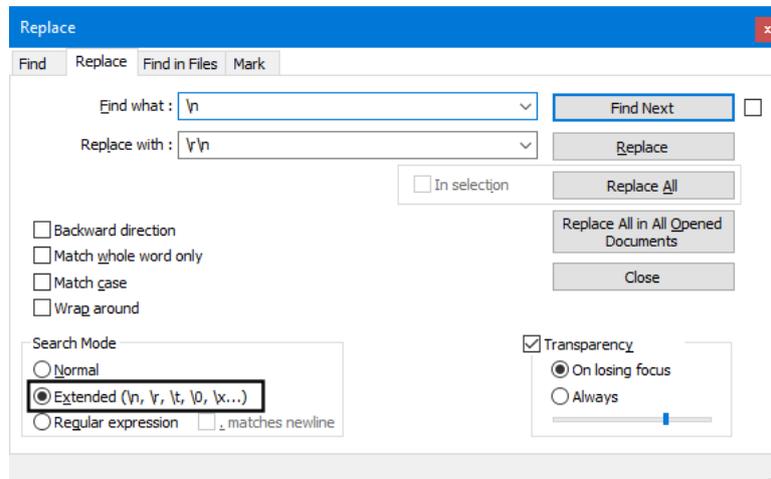


## Features of Using CSV Files (cont.)

### How to Insert the Carriage Return Character

- To insert the Carriage Return character in Notepad++:

The CR character can be inserted in Notepad++ by using Find and Replace.



Enter the characters `\n` in the **Find what:** field to locate the **Line Feed** character.

Add the characters `\r\n` to the **Replace with:** field to add the **Carriage Return** character before the **Line Feed** character.

Select the **Extended** option in the **Search Mode**.

Click the **Replace** button to insert the **CR** character on any line where the **CR** character is missing.

```
Groupmembership.csv
1 //Group.Guid,User.GuidCRLF
2 Admin,AdministratorsCRLF
3 Admin,HRCRLF
4 Operations,SupervisorsCRLF
5 Operations,ManagersCRLF
6 Operations,OperatorsCRLF
```

# Users.csv

## List of Users

The **Users.csv** file contains the list of **Users** that need to be imported into PolicyHub.

Field Name	Details	Mandatory/Optional
LoginName	User's login details (must contain domain name if using single sign-on)	Mandatory
EmailAddress	User's email address	Mandatory
Forename	User's first name	Optional
MiddleName	User's middle name/initial	Optional
Surname	User's surname	Optional
Description	Short description	Optional
DisplayName	Display name in PolicyHub (if different to LoginName)	Optional
UserId	Unique identifier (GUID) for each user	Optional

For example:

```
Users.csv
1 //LoginName,EmailAddress,Forename,MiddleName,Surname,Description,DisplayName,UserId
2 Harry,Harry@training.not,Harry,,Enfield,,Harry Enfield,36EC47E4-5CDE-4C81-B858-96F953DCCE75
3 Helen,Helen@training.not,Helen,,Mirren,,Helen Mirren,96066981-522A-4909-89D8-58D6D1E3ADDO
4 Mick,Mick@training.not,Mick,,Jagger,,Mick Jagger,4C0B8C99-1498-4D78-A979-2903CFAE6AC5
5 Mike,Mike@training.not,Mike,,Rutherford,,Mike Rutherford,11D27715-E25D-4DD1-86F9-55429196F7DF
6 Oliver,Oliver@training.not,Oliver,,Reed,,Oliver Reed,D3B7DDA5-A9BA-4BFF-8A16-EDC4D93DAD04
7 Owen,Owen@training.not,Owen,,Wilson,,Owen Wilson,D712005F-4E68-4C27-B5FE-CD86F813C9DD
8 Steve,Steve@training.not,Steve,,McQueen,,Steve McQueen,0541F5EC-B723-445F-ACFA-7E4616D3BEF7
9 Stevie,Stevie@training.not,Stevie,,Wonder,,Stevie Wonder,4F0A2AA1-3229-43FD-B321-1FCD16674E6F
10 Student,Student@training.not,Student,,,,Student,E84FDF59-EC0B-48DF-A643-59DC8A334D43
```

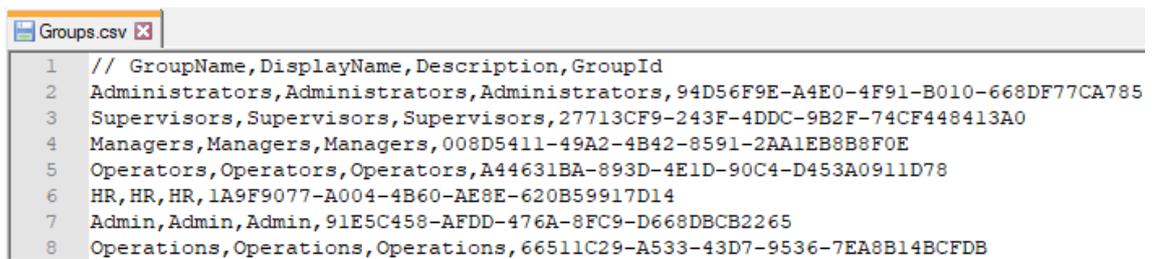
# Groups.csv

## List of Groups

The **Groups.csv** file contains the list of Groups that need to be imported into PolicyHub. The file must contain at least one group name.

Field Name	Details	Mandatory/Optional
GroupName	Name of the group	Mandatory
DisplayName	Display name in PolicyHub (if different to LoginName)	Optional
Description	Short description	Optional
GroupId	Unique identifier (GUID) for each group	Optional

For example:



```
1 // GroupName, DisplayName, Description, GroupId
2 Administrators, Administrators, Administrators, 94D56F9E-A4E0-4F91-B010-668DF77CA785
3 Supervisors, Supervisors, Supervisors, 27713CF9-243F-4DDC-9B2F-74CF448413A0
4 Managers, Managers, Managers, 008D5411-49A2-4B42-8591-2AA1EB8B8F0E
5 Operators, Operators, Operators, A44631BA-893D-4E1D-90C4-D453A0911D78
6 HR, HR, HR, 1A9F9077-A004-4B60-AE8E-620B59917D14
7 Admin, Admin, Admin, 91E5C458-AFDD-476A-8FC9-D668DBC2265
8 Operations, Operations, Operations, 66511C29-A533-43D7-9536-7EA8B14BCFDB
```

# GroupMembership.csv

## Relationship between Groups

This file is used to create the relationship between Groups and configure the Group Hierarchy in PolicyHub. Groups must also be added to the **Groups.csv** file.

Field Name	Details	Mandatory/Optional
Parent	Name or Unique identifier (GUID) of the 'parent' group	Mandatory
Child	Name or Unique identifier (GUID) of the 'child' group	Mandatory

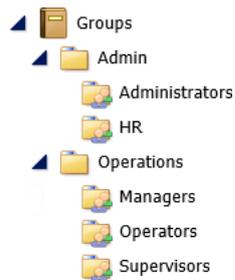
This file can also be used to create multi-layered groups if required.

### For example:

The following CSV file:

```
Groupmembership.csv
1 //Parent,Child
2 Admin,Administrators
3 Admin,HR
4 Operations,Supervisors
5 Operations,Managers
6 Operations,Operators
```

Results in the following group hierarchy:



If the CSV files are being maintained manually it is preferable to use the **Names** as the **Unique Keys** as this is the most human readable, see [Unique Keys](#) (page 1-12).

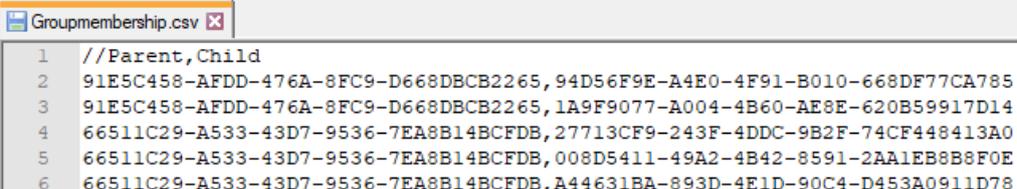
### Note:

The **names** of the **Groups** must be unique, for example, if the file is configured to add a group named **Mitratech** to a parent group called **Mitratech** the import process will fail.

## GroupMembership.csv (cont.)

### Relationship between Groups (cont.)

However, if the creation of the CSV files is automated then it is preferable to use **GUIDs** for the **UserID** and **GroupID** as it is the most consistent and secure method. In this case the names of the Groups are replaced by the **GUIDs** of the Groups as they are identified in the **Groups.csv** file:



```
1 //Parent,Child
2 91E5C458-AFDD-476A-8FC9-D668DCB2265,94D56F9E-A4E0-4F91-B010-668DF77CA785
3 91E5C458-AFDD-476A-8FC9-D668DCB2265,1A9F9077-A004-4B60-AE8E-620B59917D14
4 66511C29-A533-43D7-9536-7EA8B14BCFDB,27713CF9-243F-4DDC-9B2F-74CF448413A0
5 66511C29-A533-43D7-9536-7EA8B14BCFDB,008D5411-49A2-4B42-8591-2AA1EB8B8F0E
6 66511C29-A533-43D7-9536-7EA8B14BCFDB,A44631BA-893D-4E1D-90C4-D453A0911D78
```

# UserMembership.csv

## Group and User Relationship

This file is used to create the relationship between **Groups** and **Users** and configures the User's group membership in PolicyHub. Both the Users and Groups must have been added to the **Groups.csv** and **Users.csv** files.

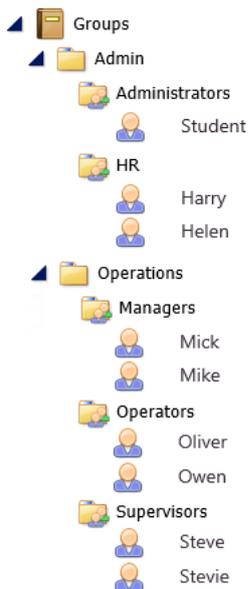
Field Name	Details	Mandatory/Optional
Parent	Name or Unique identifier (GUID) of the Group where the User needs to be a member	Mandatory
Child	Name or Unique identifier (GUID) of the User	Mandatory

### For example:

This configuration of Users and Groups:

```
Usemembership.csv
1 //Parent,Child
2 Administrators,Student
3 Supervisors,Steve
4 Supervisors,Stevie
5 Managers,Mick
6 Managers,Mike
7 Operators,Oliver
8 Operators,Owen
9 HR,Harry
10 HR,Helen
```

Results in the following User and Group hierarchy:



## UserMembership.csv (cont.)

### Group and User Relationship (cont.)

If the **GUIDs** are used as the Unique Keys the names of the Groups and Users need to be replaced with the **GUIDs** as they are identified in the **Groups.csv** and **Users.csv** files.

```
Usemembership.csv x
1 //Parent,Child
2 94D56F9E-A4E0-4F91-B010-668DF77CA785,E84FDF59-EC0B-48DF-A643-59DC8A334D43
3 27713CF9-243F-4DDC-9B2F-74CF448413A0,0541F5EC-B723-445F-ACFA-7E4616D3BEF7
4 27713CF9-243F-4DDC-9B2F-74CF448413A0,4F0A2AA1-3229-43FD-B321-1FCD16674E6F
5 008D5411-49A2-4B42-8591-2AA1EB8B8F0E,4C0B8C99-1498-4D78-A979-2903CFAE6AC5
6 008D5411-49A2-4B42-8591-2AA1EB8B8F0E,11D27715-E25D-4DD1-86F9-55429196F7DF
7 A44631BA-893D-4E1D-90C4-D453A0911D78,D3B7DDA5-A9BA-4BFF-8A16-EDC4D93DAD04
8 A44631BA-893D-4E1D-90C4-D453A0911D78,D712005F-4E68-4C27-B5FE-CD86F813C9DD
9 1A9F9077-A004-4B60-AE8E-620B59917D14,36EC47E4-5CDE-4C81-B858-96F953DCCE75
10 1A9F9077-A004-4B60-AE8E-620B59917D14,96066981-522A-4909-89D8-58D6D1E3ADD0
```

## UserHierarchy.csv (Optional)

### Organisation Hierarchy

The **UserHierarchy.csv** file is used to create a hierarchy of **Managers** and the **Employees** who are their direct reports and configures the **Organisation** in PolicyHub. The Users must have been added to the **Users.csv** file.

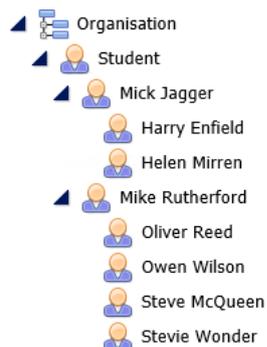
Field Name	Details	Mandatory/Optional
Manager	Name or Unique identifier (GUID) of the Manager	Mandatory
Employee	Name or Unique identifier (GUID) of the Employee who is a direct report	Mandatory

### For example:

This configuration of Managers and Employees:

```
UserHierarchy.csv
1 //Manager,Employee
2 Mick,Harry
3 Mick,Helen
4 Student,Mick
5 Student,Mike
6 Mike,Oliver
7 Mike,Owen
8 Mike,Steve
9 Mike,Stevie
```

Results in the following **Organisation** hierarchy:



If the **GUIDs** are used as the Unique Keys the names of the Managers and Employees need to be replaced with the **GUIDs** as they are identified in the **Users.csv** file.

```
UserHierarchy.csv
1 //Manager,Employee
2 4C0B8C99-1498-4D78-A979-2903CFAE6AC5,36EC47E4-5CDE-4C81-B858-96F953DCCE75
3 4C0B8C99-1498-4D78-A979-2903CFAE6AC5,96066981-522A-4909-89D8-58D6D1E3ADD0
4 E84FDF59-EC0B-48DF-A643-59DC8A334D43,4C0B8C99-1498-4D78-A979-2903CFAE6AC5
5 E84FDF59-EC0B-48DF-A643-59DC8A334D43,11D27715-E25D-4DD1-86F9-55429196F7DF
6 11D27715-E25D-4DD1-86F9-55429196F7DF,D3B7DDA5-A9BA-4BFF-8A16-EDC4D93DAD04
7 11D27715-E25D-4DD1-86F9-55429196F7DF,D712005F-4E68-4C27-B5FE-CD86F813C9DD
8 11D27715-E25D-4DD1-86F9-55429196F7DF,0541F5EC-B723-445F-ACFA-7E4616D3BEF7
9 11D27715-E25D-4DD1-86F9-55429196F7DF,4F0A2AA1-3229-43FD-B321-1FCD16674E6F
```

## Globally Unique IDs

### Unique Keys

When a User logs in to PolicyHub they are identified by a **Unique Key**.

The key can be the **Login name** or **email address**, however, if the name of the User changes another User account will be created in the system the next time the CSV files are synchronised.

The Users and Groups can be managed by generating a **Globally Unique ID (GUID)** for each User and Group. This way if the name of the User changes it can be changed in the CSV file but the GUID remains the same and a new account is not created.

Unique Key	Notes
User Login Name	The User logs in using the <b>Login name</b> that is identified in the <b>Users.csv</b> file
User Email Address	The User logs in using the <b>Email Address</b> that is identified in the <b>Users.csv</b> file. However, this is not recommended if users are likely to change their email address. Examples of this could be if a user changes business units that may be issued with a different email address or the user changes their name due to marriage.
User GUID	The User logs in using the <b>Login name</b> that is identified in the <b>Users.csv</b> file
Group Name	The <b>Group Name</b> defined in the <b>Groups.csv</b> file is used in the <b>GroupMembership.csv</b> and <b>UserMembership.csv</b> files
Group GUID	The <b>Group GUID</b> defined in the <b>Groups.csv</b> file is used in the <b>GroupMembership.csv</b> and <b>UserMembership.csv</b> files
User Hierarchy User Login Name	The <b>User Login Name</b> defined in the <b>Users.csv</b> file is used in the <b>UserHierarchy.csv</b> file
User Hierarchy User Email Address	The <b>User Email Address</b> defined in the <b>Users.csv</b> file is used in the <b>UserHierarchy.csv</b> file
User Hierarchy User GUID	The <b>User GUID</b> defined in the <b>Users.csv</b> file is used in the <b>UserHierarchy.csv</b> file

## Globally Unique IDs (cont.)

### GUID Generators

There are various ways to generate GUIDs. Online generators are available, for example, this one at [guidgenerator.com](http://guidgenerator.com).

### Online GUID Generator

How many GUIDs do you want (1-2000):   
Uppcase:  Braces:  Hyphens:   
Base64 encode:  RFC 7515:  URL encode:

Generate some GUIDs!

Results:

```
A48A3CED-5C52-4BE7-BBC7-6E208AD24BBA  
8CCEEEF8-AB0F-4CDF-9E98-49464C2FA152  
5B7C1A1A-A3D4-4C0D-9DF7-B039187EB617  
FE87A0D1-28FE-430B-9EF7-010DB6F8B648  
B93521C2-34DA-4ECE-BD10-1B64EC9A5421  
17A4F722-4A36-4C25-AD4C-742DE976EE67  
78B4003A-4AE3-4BC5-B320-5547B1C95D65  
C6B7E705-BCA5-4F16-AB09-11FC483A3C7F  
A5C6CDCF-9EFC-418B-8C67-1571FE8112CA  
DE640915-FC6B-47E3-B287-D8FF2649B776
```

# User/Group Import and Synchronisation Process

---

## Daily Synchronisation

PolicyHub's User synchronisation process runs daily and checks the FTP upload folder.

If files are present, **PolicyHub** uses these four files to import users, groups and their relationships into the application.

- Users.csv
- Groups.csv
- GroupMembership.csv
- UserMembership.csv

The file **UserHierarchy.csv** is optional and will not cause a synchronisation error if it is not used.

If files are not present the process will not complete and an error will be logged in the **Synchronisation Log**. No user or group data is amended in PolicyHub.

Where there is an error found in the CSV files, for example, if a Group name or Hierarchy is missing, then an error will be produced in the **Synchronisation.log** file and no updates will be made to PolicyHub.

Administrators need to ensure that all CSV files have matching data in them and that the Synchronisation Log is checked regularly for errors.

## User/Group Import and Synchronisation Process (cont.)

### Adding / Removing Users

The CSV files can be maintained manually by using tools such as Notepad++ or Excel. When these tools are used the files must be saved with the extension **\*.csv**.

If new Users are to be added to PolicyHub then those Users need to be included in the CSV files in addition to the existing Users. The entry in the **Users.csv** file creates the User account.

```
Users.csv
1 //LoginName,EmailAddress,Forename,MiddleName,Surname,Description,DisplayName,UserId
2 Harry,Harry@training.not,Harry,,Enfield,,Harry Enfield,
3 Helen,Helen@training.not,Helen,,Mirren,,Helen Mirren,
4 Mick,Mick@training.not,Mick,,Jagger,,Mick Jagger,
5 Mike,Mike@training.not,Mike,,Rutherford,,Mike Rutherford,
6 Oliver,Oliver@training.not,Oliver,,Reed,,Oliver Reed,
7 Owen,Owen@training.not,Owen,,Wilson,,Owen Wilson,
8 Steve,Steve@training.not,Steve,,McQueen,,Steve McQueen,
9 Stevie,Stevie@training.not,Stevie,,Wonder,,Stevie Wonder,
10 Student,Student@training.not,Student,,,,Student,
11 Frank,Frank@training.not,Frank,,Sinatra,,Frank Sinatra,
```

When a User is added their account will appear in PolicyHub in the **Roles** section as a **User** but the account will not be added to any **Group** displayed in the **Member of** column or the **Organisational Hierarchy** which provides the **Manager**.

Users (9)					
	First name	Surname	Role name	Manager(s) ^	Member of
	Frank	Sinatra	User		
	Helen	Mirren	User	Mick Jagger	HR
	Harry	Enfield	User	Mick Jagger	HR

## NOTICE

### UNINTENDED EQUIPMENT OPERATION

Do not upload a CSV file containing ONLY new users

**Failure to follow these instructions can result in equipment damage.**

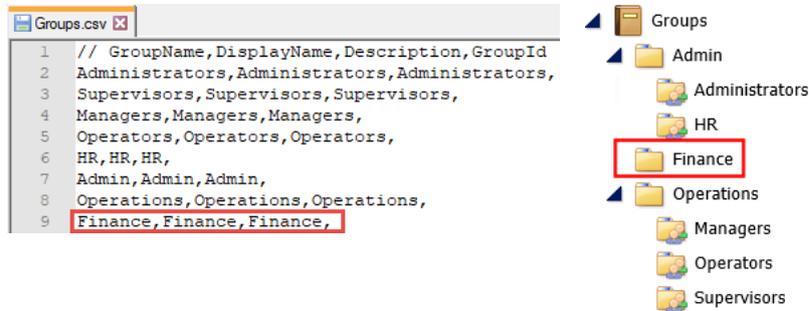
PolicyHub assumes the **Users.csv** file is the complete list of all active PolicyHub Users and will therefore disable any existing Users when they are no longer present in the import.

# User/Group Import and Synchronisation Process (cont.)

## Add Groups and User Membership

For the import process to run successfully, all four (optionally five) files are required and their contents must be logically consistent otherwise the import process will have errors and no updates will be processed.

The **Groups.csv** file creates the Group. This will add the Group to PolicyHub at the top level.



The screenshot shows a text editor window titled 'Groups.csv' with the following content:

```
1 // GroupName, DisplayName, Description, GroupId
2 Administrators, Administrators, Administrators,
3 Supervisors, Supervisors, Supervisors,
4 Managers, Managers, Managers,
5 Operators, Operators, Operators,
6 HR, HR, HR,
7 Admin, Admin, Admin,
8 Operations, Operations, Operations,
9 Finance, Finance, Finance,
```

Red boxes highlight the last line of the CSV file and the 'Finance' folder in the adjacent tree view. The tree view shows a hierarchy starting with 'Groups' at the top, followed by 'Admin', 'Administrators', 'HR', 'Finance', 'Operations', 'Managers', 'Operators', and 'Supervisors'.

If the Group is added into the **GroupMembership.csv** file it will be placed as a Child group in the **Group Hierarchy**.

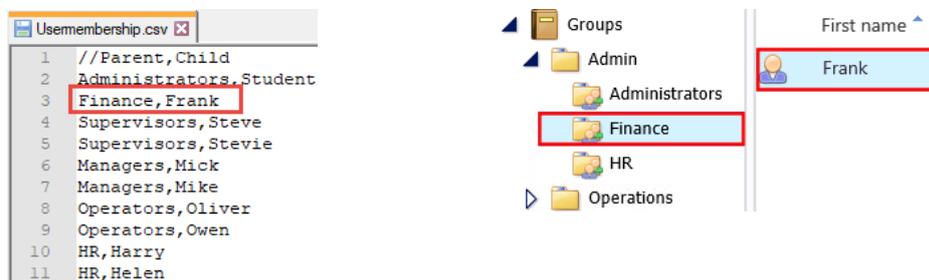


The screenshot shows a text editor window titled 'Groupmembership.csv' with the following content:

```
1 //Parent, Child
2 Admin, Administrators
3 Admin, HR
4 Admin, Finance
5 Operations, Supervisors
6 Operations, Managers
7 Operations, Operators
```

Red boxes highlight the fourth line of the CSV file and the 'Finance' folder in the adjacent tree view. The tree view shows a hierarchy starting with 'Groups' at the top, followed by 'Admin', 'Administrators', 'Finance', and 'HR'.

The **UserMembership.csv** file adds a User to a Group. If a User needs to be a member of multiple groups a line entry for each **Group/User** relationship needs to be added.



The screenshot shows a text editor window titled 'Usermembership.csv' with the following content:

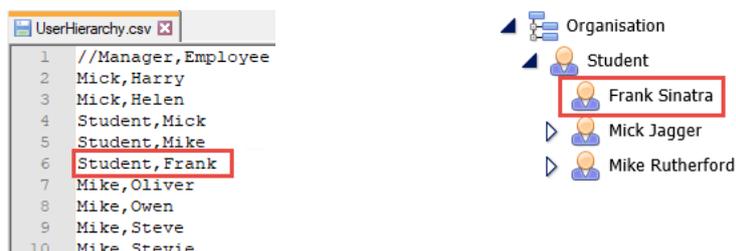
```
1 //Parent, Child
2 Administrators, Student
3 Finance, Frank
4 Supervisors, Steve
5 Supervisors, Stevie
6 Managers, Mick
7 Managers, Mike
8 Operators, Oliver
9 Operators, Owen
10 HR, Harry
11 HR, Helen
```

Red boxes highlight the third line of the CSV file and the 'Frank' user in the adjacent table. The table shows a hierarchy starting with 'Groups' at the top, followed by 'Admin', 'Administrators', 'Finance', 'HR', and 'Operations'. The 'Frank' user is listed under the 'Finance' group.

## User/Group Import and Synchronisation Process (cont.)

### Add to the User Hierarchy

The **UserHierarchy.csv** file creates an **Organisation** within PolicyHub. This is not mandatory but it is used to create a **Manager/Employee** structure that is used for escalation emails to a Manager. When a User is added the result in PolicyHub looks like this:



The User details will display the Manager and Group Membership

Users (9)					
	First name	Surname	Role name	Manager(s)	Member of
	Frank	Sinatra	User	Student	Finance
	Harry	Enfield	User	Mick Jagger	HR
	Helen	Mirren	User	Mick Jagger	HR

Logical errors will occur if the CSV files are maintained manually, i.e. the PolicyHub Membership Populator is not used to automatically create the files, and, for example, a user is removed from the Users file and not from the UserMembership file.

The process of removing users from PolicyHub is to remove them from the CSV file and to upload/overwrite the files on the FTP site. Those users will be disabled in PolicyHub to ensure that they still appear in compliance reporting but will no longer have system access. A disabled User's access can be re-enabled by reversing the process and re-including them in the CSV file.

### Remove Users

Logical errors will occur if the CSV files are maintained manually, i.e. the **PolicyHub Membership Populator** is not used to create the files automatically and, for example, a User is removed from the **Users.csv** file and not from the **UserMembership.csv** file.

```
Exception raised while adding user hierarchy membership line 'Student, Frank' : The user is not known
Finished retrieving user hierarchy membership
Finished gathering CSV membership
Enumeration completed with 2 error(s)
Maximum configured error count exceeded!.
```

The process of removing Users from PolicyHub is to remove them from the CSV file and to upload/overwrite the files on the FTP site. Those Users will be disabled in PolicyHub to ensure that they still appear in compliance reporting but will no longer have system access. A disabled User's access can be re-enabled by reversing the process and re-including them in the CSV file.