# NSSA221 Systems Administration I

# Scripting Assignment 02 – Automate Adding User Accounts

You have been tasked to write a script that will add local Linux user accounts to a CentOS 7 virtual machine. You will be given a CSV file; the file is located in myCourses under *Content 🡪 Scripting Assignments 🡪 Script Files*, titled “Lab02\_Users.csv”. Again, scripting can make the job of a system administrator much easier, and while much of the time will be spent on the front-end writing the script, it will inevitably translate to less work on the backend when it comes time to add additional users to the system.

From time to you will receive CSV files from various departments, these are created by an unknown entity, so how the file is formatted is anyone’s guess and needs to be considered when writing your code. **DO NOT alter or modify the file in any way.** For this reason, the script must meet the following requirements.

1. The CSV file contains a header line defining the fields; your script must detect the header line and not process it as if it were a user addition.
2. The users should have a default group as indicated in the File. Again, **do not edit, or alter the file**, you must use the data as is.
3. You must be able to create unique user names, which should be constructed as first initial followed by the last name (e.g., John Smith would be assigned the user name '*jsmith*').
4. You must be able to handle duplicate names, what if the file has three employees named John Smith? Or “Jason Smith?”
5. You must be able to handle embedded special characters, for example what if the persons last name is "O'Donnell?"
6. You must be able to handle missing information (what happens if you are given a file with a field that is not populated? The script must be handle this possibility.
7. The user's home directory should be located in /home/*department*, where department is the user's department, e.g., for Natasha Richardson in the CEO department, the correct home directory would be /home/ceo/nrichardson - and note all lowercase.
8. You must be able to detect and handle incorrect data (e.g., what if the user's full name in the CSV file is '555-1212'? i.e., what if the person doing data entry made mistakes?).
9. Create a unique password for each user you add, expire the password so they are forced to change that initial password upon their first login. 'passwd -e user' will expire the initial password and require the user to change their password at the next login. **Hint:** reverse the user name ('jsmith') to create a password (htimsj').
10. Each record in the CSV file indicates the default group for the corresponding user. Your script should be able to handle the need for a new group, in other words, create the group if it does not exist.

Figure 1 provides sample output from the script, your scripts output does not need to look exactly like that in Figure 1. The output from Figure 1 is more verbose than what your script needs to be, the output is give you some idea of what the script is doing and how the “useradd” command is being used.

**Figure 1** – Sample Script Output

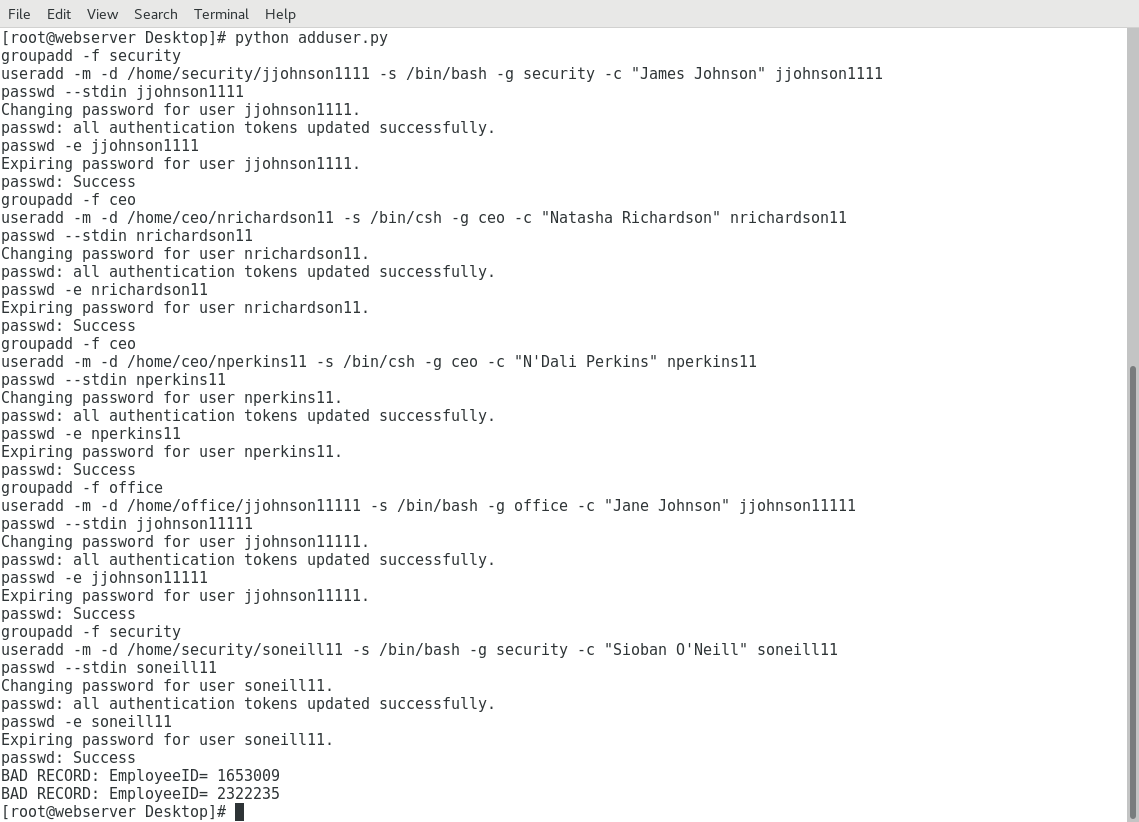
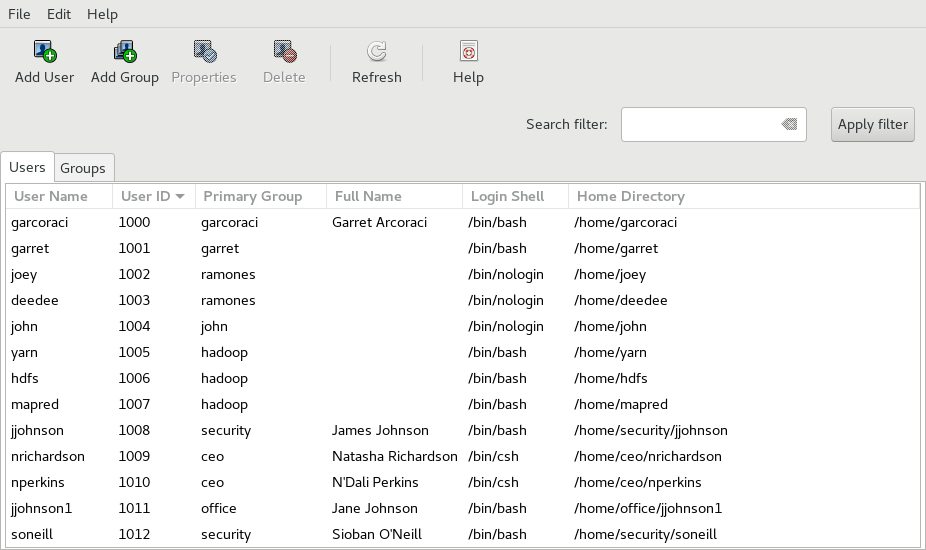


Figure 2 is more suggestive of what your script should do to add user accounts to the system. Notice that each user is given a unique user name and directory. Note the group membership and default shell assigned to each user.

**Figure 2** – User Manager

\*\*\* Please submit the sign off pages to the drop box \*\*\*

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Deliverables** | **Points** | **Date** |  | **Signature** |
| Script checks if group exists and creates the group if it does not. | 5 |  |  |  |
| The Script can read in and process the csv file as is. | 7 |  |  |  |
| The script ignores the header line. | 3 |  |  |  |
| The script properly formats the user ID. | 7 |  |  |  |
| The script can handle duplicate or similar names. | 3 |  |  |  |
| Each user is assigned proper group membership. | 5 |  |  |  |
| Each user is given a home directory according to the organization they belong to. | 7 |  |  |  |
| The user is assigned the required default shell. | 5 |  |  |  |
| The user is required to reset their password the first time they log in. | 5 |  |  |  |
| The script communicates when a bad record exists. | 3 |  |  |  |
|  | | | | |
| **Points Earned** (max 50) | | | |  |