Task2

|  |  |
| --- | --- |
| 1) Analyze the structure of the /etc/passwd and /etc/group file, what fields are present in it, what users exist on the system? Specify several pseudo-users, how  to define them? | Passwd format  username: pswd: uid: gid: uid comments: directory: shell    Group format  group\_name:password:group\_id:list  Pseudo-users  daemon - Used by system service processes  bin - Gives ownership of executables command  adm - Owns registration files  nobody - Used by many services  sshd – used by the secure shell server. |
| 2) What are the uid ranges? What is UID? How to define it? | The system UIDs from 0 to 99 should be statically allocated by the system, and shall not be created by applications.  The system UIDs from 100 to 499 should be reserved for dynamic allocation by system.  A unique identifier (UID) is an identifier that marks that particular record as unique from every other record. |
| 3) What is GID? How to define it? | A GID (group IDs) is a group identifier.  GID 0 (zero) is reserved for the root group.  GID 1–99 are reserved for the system and application use.  GID 100+ allocated for the user’s group. |
| 4) How to determine belonging of user to the specific group? |  |
| 5) What are the commands for adding a user to the system? What are the basic  parameters required to create a user? | **adduser** command creates a new user with the system default parameters, which are written in the /etc/default/useradd file  **useradd**  Create new user: useradd {{name}}  Create a new user with the default home dir: useradd --create-home {{name}}  Create a new user with a specific command shell: useradd --shell {{/path/to/shell}} {{name}}  Create a new user without home directory: useradd --no-create-home --system {{name}} |
| 6) How do I change the name (account name) of an existing user? |  |
| 7) What is skell\_dir? What is its structure? | The **/etc/skel** directory contains files and directories that are automatically copied over to a new user’s home directory when such a user is created by the **useradd** program |
| 8) How to remove a user from the system (including his mailbox)? | userdel -fr username |
| 9) What commands and keys should be used to lock and unlock a user account? |  |
| 10) How to remove a user's password and provide him with a password-free  login for subsequent password change? |  |
| 11) Display the extended format of information about the directory, tell about  the information columns displayed on the terminal. | **Ls -l** long format of information about the directory.    The first character represents the file type (- d I ets.)  The next three characters represent the permissions for the file's owner.  Next three – the permissions for members of the file group.  Next three - The permissions for "others".  The number of hard links to this file.  The file's owner.  The group to whom the file belongs.  The size of the file in bloks.  The file's mtime (date and time when the file was last modified).  The name of the file. |
| 12) What access rights exist and for whom (i. e., describe the main roles)?  Briefly describe the acronym for access rights. | **rwx** - three main permissions allow you to read, write and use files.  Any Linux user (process) in relation to any file can act in three roles: as the owner (user), as a member of the group that owns the file (group), and as an outsider (other), has no ownership relations of  this file. |
| 13) What is the sequence of defining the relationship between the file and the  user? | If the UID of the file is the same as the UID of the process, the user is the owner of the file  If the GID of the file matches the GID of any group the user belongs to, he is a member of the group to  which the file belongs.  If neither the UID no the GID of a file overlaps with the UID of the process and the list of groups that the  user running it belongs to, that user is an outsider. |
| 14) What commands are used to change the owner of a file (directory), as well  as the mode of access to the file? Give examples, demonstrate on the terminal. |  |
| 15) What is an example of octal representation of access rights? Describe the  umask command. | **umask** is a command to check default permission of newly created file or folder.    A user-defined permissions ‘mask’. A user can choose how to restrict  permissions by using a **permissions mask**. A permission mask interacts with the default system permissions and changes them. The **umask** command is used to apply this mask.  A mask can have the following numeric, and the corresponding symbolic, values: |
| 16) Give definitions of sticky bits and mechanism of identifier substitution. Give  an example of files and directories with these attributes. | Sticky Bit is mainly used on folders in order to avoid deletion of a folder and it’s content by other users.  **ls -l** can be used to check if the x in others permissions field is replaced by t or T |
| 17) What file attributes should be present in the command script? | **script** - make typescript of terminal session  If the argument file is given, script saves the dialogue in this file. If no filename is given, the dialogue is saved in the file typescript. |