Managing User Accounts and User Environments in Oracle® Solaris 11.4
# Contents

Using This Documentation ................................................................. 7

1 About User Accounts and User Environments ........................................ 9
   What's New in Managing User Accounts in Oracle Solaris 11.4 ................. 9
   About User Accounts and Groups ....................................................... 10
      User Account Components .............................................................. 10
      Guidelines for Assigning User Names, UIDs, and GIDs ..................... 12
   Where User Account and Group Information Is Stored .......................... 14
      The /etc/passwd File ...................................................................... 14
      The /etc/shadow File ...................................................................... 15
      The /etc/group File ........................................................................ 15
   User and Group Commands ................................................................. 15
   About the User Work Environment ..................................................... 17
      Supported Shells in Oracle Solaris ....................................................... 17
      About Initialization Files ................................................................. 18
   Default File Permissions (umask) ......................................................... 19
   Managing Users by Using Oracle Enterprise Manager Ops Center .......... 20

2 Managing User Accounts by Using the Command-Line Interface ............. 21
   Setting Up and Managing User Accounts (Task Map) .............................. 21
   Setting Up User Accounts by Using the CLI ......................................... 22
      About Setting Up User Accounts ....................................................... 22
      Gathering User Information ............................................................... 23
      Identifying Users by Packages ......................................................... 23
      ▼ How to Customize User Initialization Files .................................... 23
      ▼ How to Change Account Defaults for All Roles ............................. 24
   Managing User Accounts by Using CLI ................................................. 25
      ▼ How to Add a User ....................................................................... 25
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>▼ How to Modify a User Account</td>
<td>27</td>
</tr>
<tr>
<td>▼ How to Unlock a User Account</td>
<td>27</td>
</tr>
<tr>
<td>▼ How to Delete a User</td>
<td>28</td>
</tr>
<tr>
<td>▼ How to Add a Group</td>
<td>29</td>
</tr>
<tr>
<td>▼ How to Create the Home Directory for a User Without Creating a ZFS Dataset</td>
<td>30</td>
</tr>
<tr>
<td>▼ How to Assign Default User Attributes for LDAP Accounts</td>
<td>31</td>
</tr>
<tr>
<td>Sharing ZFS File Systems</td>
<td>32</td>
</tr>
<tr>
<td>▼ How to Share Home Directories That Are Created as ZFS File Systems</td>
<td>32</td>
</tr>
<tr>
<td>Manually Mounting a User's Home Directory</td>
<td>34</td>
</tr>
<tr>
<td>3 Managing User Accounts Interactively</td>
<td>35</td>
</tr>
<tr>
<td>Security Considerations</td>
<td>35</td>
</tr>
<tr>
<td>About the <code>useradm</code> Application</td>
<td>35</td>
</tr>
<tr>
<td>▼ How to Install the <code>useradm</code> Application</td>
<td>36</td>
</tr>
<tr>
<td>Using the Interactive <code>useradm</code> Command</td>
<td>36</td>
</tr>
<tr>
<td>Attributes in the <code>useradm</code> Window</td>
<td>37</td>
</tr>
<tr>
<td>▼ How to Manage Users and Roles Interactively</td>
<td>39</td>
</tr>
<tr>
<td>About the Oracle Solaris Account Management BUI</td>
<td>40</td>
</tr>
<tr>
<td>▼ How to Access the Oracle Solaris Account Management BUI</td>
<td>40</td>
</tr>
<tr>
<td>Index</td>
<td>43</td>
</tr>
</tbody>
</table>
Using This Documentation

- **Overview** – Describes how to manage user accounts and user environments in Oracle Solaris 11.4.
- **Audience** – System administrators using the Oracle Solaris 11.4 release.
- **Required knowledge** – Experience administering UNIX systems.

**Product Documentation Library**

Documentation and resources for this product and related products are available at [https://docs.oracle.com/cd/E37838_01/index.html](https://docs.oracle.com/cd/E37838_01/index.html).

**Feedback**

Provide feedback about this documentation at [http://www.oracle.com/goto/docfeedback](http://www.oracle.com/goto/docfeedback).
About User Accounts and User Environments

This chapter contains these main topics:

- “What's New in Managing User Accounts in Oracle Solaris 11.4”
- “About User Accounts and Groups”
- “Where User Account and Group Information Is Stored”
- “User and Group Commands”
- “About the User Work Environment”
- “Default File Permissions (umask)”
- “Managing Users by Using Oracle Enterprise Manager Ops Center”

For task-related information about managing user accounts and user environments, see Chapter 2, “Managing User Accounts by Using the Command-Line Interface”.

What's New in Managing User Accounts in Oracle Solaris 11.4

The following features are in this release:

- New application to set up and manage user accounts – The useradm application replaces the Visual Panels User Manager for setting up and managing user accounts. This application is a menu-driven interactive interface that also supports command line options.
- Oracle Solaris Account Management BUI – The browser-based Oracle Solaris Account Management application is another tool for managing user accounts.
- Default account name for LDAP accounts – The new default@ account name enables you to assign default user attributes to accounts in the Lightweight Directory Access Protocol (LDAP) directory.
- Clearance attribute for restricting user access Through the clearance attribute, you can assign labels to users to restrict access to certain confidential process information.
- Increased user name length – User names greater than 8 bytes are supported.
About User Accounts and Groups

A user account includes the information a user needs to log in and use a system. The following sections describe components of user accounts.

A role is a special type of user account that grants special rights to specific users. For more information, see Chapter 1, “About Using Rights to Control Users and Processes” in Securing Users and Processes in Oracle Solaris 11.4.

All security policy configurations for user account attributes are provided by the SMF service account-policy. In previous Oracle Solaris releases, these were defined in individual configuration files such as /etc/default/login or /etc/security/policy.conf. For information about configuring security policies through the account-policy SMF service, see “Modifying Rights System-Wide As SMF Properties” in Securing Users and Processes in Oracle Solaris 11.4 as well as the account-policy(8S) man page.

User Account Components

The following sections describe components of a user account.

User Names

Users access local or remote systems through their user or login names. The extent of access depends on the user's rights.

Before you create accounts, plan on a standard way of assigning user names that facilitates tracking them and associating them with actual persons. With the standard as foundation, you only need to make slight adjustments for contingencies. For example, if your standard is based on users' first initials and surnames, you can easily adjust if two users share the same first initial and last names.

User ID Numbers

Each user name is associated with a unique user identification number (UID). A UID can be any whole number up to 2147483647.

Systems use the UIDs to identify users who are logging in as well as owners of files and directories. For ease of management, an individual with accounts on different systems should
use the same user name and UID. UIDs are required for both regular user accounts and special system accounts.

**UNIX™ Groups**

A *group*, also known as a UNIX group, is a collection of users who can share files and other system resources, such as users who are working on the same project.

Users are added to groups. Permissions granted to groups control the types of access that members have to files or directories.

Each group must have a name and a group identification (GID) number. The GID identifies the group internally to the system.

A user can belong to two types of groups:

- **Primary group** – automatically created by the operating system. The group is given access rights to specific files, including those that are created by the user. Each user must belong to a primary group. By default, a new user is added to the *staff* group (GID 10).
- **Secondary group** – manually created to which users are added as members. Users can belong to up to 1024 supplemental groups.

The priority of secondary groups vary. For example, file ownership requires membership in a primary group. But other applications might rely on a user's belonging to a secondary group regardless of the user's primary group.

Groups can be local to a system or managed through a name service. To simplify group administration, use the LDAP directory service to centrally manage group memberships.

**User Passwords**

You can specify the password for a new user or force the user to specify a unique password at the user's initial login.

By default, the minimum password length is set to 8 bytes by the `PASSLENGTH` parameter of the `/etc/default/passwd` file. This file includes other configurable parameters that together form the password policy that is enforced at your site.

Policies surrounding password creation and changes must implement the strictest security standards possible such as use of special characters, restricting the reuse of old passwords, case sensitivity, allowing longer passwords, and so on. These practices help prevent password theft and consequent security breaches.
For additional password attributes, see the passwd(1) man page.

Oracle Solaris has other security features other than passwords. Refer to the Securing the Oracle Solaris Operating System shelf in https://docs.oracle.com/cd/E37838_01/.

Home Directories

The home directory is the portion of a file system that is allocated to a user for storing private files. You determine the amount of space to allocate for home directories.

A home directory can be either on the local system or on a remote file system. By convention the home directory should be created as /export/home/username. For a large site, you should store home directories on a server.

Regardless of the home directory location, users usually access their home directories through a mount point named /home/username.

If AutoFS is used to mount home directories, the home mount point acquires a special status. Thus, you cannot create any directories on that mountpoint on any system. For more information about auto-mouting home directories, see “Autofs Administration” in Managing Network File Systems in Oracle Solaris 11.4.

Directory Services

Oracle Solaris 11.4 uses LDAP as its directory service. The directory service enables you to store user account information in a centralized manner. The service provides ease of connections to different systems and also ensures consistent user account information.

Guidelines for Assigning User Names, UIDs, and GIDs

Keep the following guidelines in mind when creating user or role names, UIDs, and GIDs:

- **User names** – Should be unique within your organization. For acceptable formats of user names, refer to passwd(5).
- **System accounts** – Observe the restrictions on the use of certain UIDs and GIDs regardless of their being in actual use. For guidance, refer to Table 1, “Reserved UID Numbers,” on page 13.
Never use the nobody and nobody4 accounts for running processes. These accounts are reserved for use by NFS. Use of these accounts for running processes can lead to unexpected security risks. Processes that need to run as non-root should use the daemon or noaccess accounts.

**System account configuration** – Never change the configuration of the default system accounts, including the login shell of a system account that is currently locked. The only exception to this rule is the setting of a password and password aging parameters for the root account.

Changing a password for a locked user account changes the password but no longer unlocks the account at the same time. A second step to unlock the account by using the `passwd -u` command is now required.

The next sections further describe limitations to observe when assigning UIDs and GIDs.

### Reserved UIDs

The following table lists the UID ranges that are assigned to user and system accounts.

<table>
<thead>
<tr>
<th>UID Numbers</th>
<th>User or Login Accounts</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0–99</td>
<td>root, daemon, bin, sys, and so on</td>
<td>Reserved for use by the operating system</td>
</tr>
<tr>
<td>100–2147483647</td>
<td>Regular users</td>
<td>General purpose accounts</td>
</tr>
<tr>
<td>60001 and 65534</td>
<td>nobody and nobody4</td>
<td>Network File System (NFS) Anonymous users</td>
</tr>
<tr>
<td>60002</td>
<td>noaccess</td>
<td>Reserved for operating system (OS)</td>
</tr>
</tbody>
</table>

Of the UIDs reserved for the OS, root typically has 0, daemon 1, and pseudo-user bin 2.

As with user names, adopt a scheme for assigning unique UID numbers. For example, some companies assign unique employee numbers. Then, administrators add a number to the employee number to create a unique UID for each employee.

To minimize security risks, avoid reusing UIDs from deleted accounts. If you must reuse a UID, remove the account information completely so that the new user is not affected by attributes set for a previous user.

### User ID and Group ID Maximum Values

The following table describes UID and GID limitations.
Where User Account and Group Information Is Stored

Depending on your site policy, user account and group information can be stored on the local system's /etc files or in the LDAP database file.

Most user account information is stored in the passwd file. Password information is stored as follows:

- In the /etc/passwd and /etc/shadow files when you are using /etc files
- In the people container when you are using LDAP

Group information is stored in LDAP's group container. LDAP also supports password aging.

The /etc/passwd File

Password information is stored in /etc/passwd. After installation, the file is automatically filled with information about standard daemons and processes. These daemons are started at boot time to perform system-wide tasks such as printing, network administration, or port monitoring. The file also contains the initial user that is created during installation.

As you add or remove packages from the system, additional users and groups are created or removed in the file. You do not perform any administrative tasks on this file.

The fields in the passwd file are separated by colons and contain the following information:

```
username:password:UID:GID:comment:home-directory:login-shell
```

For example:

```
kryten:x:101:100:Kryten Series 4000 Mechanoid:/export/home/kryten:/bin/csh
```
For a complete description of the fields in the passwd file, see the passwd(1) man page.

**The /etc/shadow File**

This file stores encrypted user passwords and related information. Typically, you do not manually administer this file.

For the regular user, the fields in the shadow file are separated by colons and contain the following information:

username:password:lastchg:min:max:warn:inactive:expire

In this file, the password is represented by a hash, such as $5$cg0k21Uy$AhHtVGx5OJ0 .W3NCKjikb8.Kh01AIo40pxsW55sP0UnYD.

For a complete description of the fields in the shadow file, see the shadow(5) man page.

**The /etc/group File**

This file is a local source of group information. After installation, groups are created by default that support some system-wide tasks such as printing, network administration, or electronic mail. Most of these groups have corresponding entries in the /etc/passwd file.

The fields in the group file are separated by colons and contain the following information:

group-name:group-password:GID:user-list

For example:

bin::2:root,bin,daemon

For a complete description of the fields in the group file, see the group(5) man page.

**User and Group Commands**

Different commands for users and groups are supported in Oracle Solaris.
The commands described in the following table are used for managing users, roles, and groups.

**TABLE 3**  
Commands Used to Manage Users, Roles, and Groups

<table>
<thead>
<tr>
<th>Man Page for Command</th>
<th>Description</th>
<th>For Additional Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>useradd(8)</td>
<td>Creates users locally or in an LDAP repository.</td>
<td>“How to Add a User” on page 25</td>
</tr>
<tr>
<td>usermod(8)</td>
<td>Changes user properties locally or in an LDAP repository. If the user properties are security-relevant, such as role assignment, this task might be restricted to your security administrator or to the root role.</td>
<td>“How to Modify a User Account” on page 27</td>
</tr>
<tr>
<td>userdel(8)</td>
<td>Deletes a user from the system or from the LDAP repository. Can involve additional cleanup, such as cron job removal.</td>
<td>“How to Delete a User” on page 28</td>
</tr>
<tr>
<td>roleadd(8)</td>
<td>Manages roles locally or in an LDAP repository. Roles cannot log in. Users assume an assigned role to perform administrative tasks.</td>
<td>“Assigning Rights to Users” in Securing Users and Processes in Oracle Solaris 11.4</td>
</tr>
<tr>
<td>rolemod(8)</td>
<td>Manages roles locally or in an LDAP repository. Roles cannot log in. Users assume an assigned role to perform administrative tasks.</td>
<td>“Assigning Rights to Users” in Securing Users and Processes in Oracle Solaris 11.4</td>
</tr>
<tr>
<td>roledel(8)</td>
<td>Manages roles locally or in an LDAP repository.</td>
<td>“Creating a Role” in Securing Users and Processes in Oracle Solaris 11.4</td>
</tr>
<tr>
<td>groupadd(8)</td>
<td>Manages groups locally or in an LDAP repository.</td>
<td>“How to Add a Group” on page 29</td>
</tr>
<tr>
<td>groupmod(8)</td>
<td>Manages groups locally or in an LDAP repository.</td>
<td>“Assigning Rights to Users” in Securing Users and Processes in Oracle Solaris 11.4</td>
</tr>
<tr>
<td>groupdel(8)</td>
<td>Manages groups locally or in an LDAP repository.</td>
<td>“Assigning Rights to Users” in Securing Users and Processes in Oracle Solaris 11.4</td>
</tr>
</tbody>
</table>

The following table describes the commands that system administrators can use to obtain information about user accounts. This information is stored in various files within the /etc directory.

**TABLE 4**  
Commands Used to Obtain Information About Users

<table>
<thead>
<tr>
<th>Command Man Page Reference</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>auths(1)</td>
<td>Lists and manages authorizations.</td>
</tr>
<tr>
<td>getent(8)</td>
<td>Displays a list of entries from the administrative database. The information generally comes from one or more of the sources that are specified for the /etc/nsswitch.conf database.</td>
</tr>
<tr>
<td>logins(8)</td>
<td>Displays information about users, roles, and system logins. The output is controlled by the command options that are specified and can include user, role, system login, UID, passwd account field value, primary group, primary group ID, multiple group names, multiple group IDs, home directory, login shell, and password-aging parameters.</td>
</tr>
<tr>
<td>profiles(1)</td>
<td>Lists and manages rights profiles.</td>
</tr>
<tr>
<td>roles(1)</td>
<td>Displays the roles that are assigned to a user.</td>
</tr>
<tr>
<td>userattr(1)</td>
<td>Displays the first value that is found for attribute_name. If a user is not specified, the user is taken from the real user ID of the process. Attribute names are defined in the man pages.</td>
</tr>
</tbody>
</table>
The `groups` command lists the groups to which a user belongs. A user can have only one primary group at a time. However, through the `newgrp` command, users can temporarily change their primary group.

### About the User Work Environment

Users need a work environment where they can access not only their home directories but other tools and resources to perform their tasks. This environment is provided by shells.

### Supported Shells in Oracle Solaris

Oracle Solaris supports the following UNIX shells:

- GNU Bourne-Again Shell (bash) (`/usr/bin/bash`)
  - Bash is the default shell for users in Oracle Solaris.
- Korn Shell (ksh93) (`/usr/bin/ksh`)
- C Shell and enhanced C Shell (`/usr/bin/csh` and `/usr/bin/tcsh`)
- POSIX-compliant Shell (`/usr/xpg4/bin/sh`)
- Z Shell (`/usr/bin/zsh`)

**Note** - This guide does not provide a comprehensive information about shells. For details, refer to the shells' corresponding man pages. For bash, the default shell for Oracle Solaris users, see the following:

  - This page provides links to the same documentation in different file formats.

Oracle Solaris uses user rights management to implement security and provides an alternative to using the superuser model. Based on this implementation, users are assigned profile shells so they can also run privileged applications. Thus, when you display shell information, you might see output similar to the following:
About the User Work Environment

```bash
$ echo $SHELL
/usr/bin/pfbash
$ which bash
/usr/bin/bash
```

Other supported shells have corresponding profile shells, such as pfksh, pfcsh, and so on.

For more information about rights, profiles, and roles, see Chapter 1, “About Using Rights to Control Users and Processes” in Securing Users and Processes in Oracle Solaris 11.4.

About Initialization Files

Initialization files, also called startup files, are files that the shell uses to create the work environment in which to operate. They also set different variables for user access to directories, resources, and tools.

You can use two types of initialization files:

- **User initialization files** – a shell script that sets up a work environment for a user after the user logs in to a system. The file defines the characteristics of user work environment, such as search paths and several environment variables. The user initialization file is stored in the user’s home directory.

- **Site initialization file** – initialization file that typically resides on a server or a set of servers. Because of its central location, this file enables administrators to introduce new functionality to the user’s work environment while maintaining the user’s ability to customize the user’s own initialization file. The site initialization file is referenced through a line that is added at the beginning of user initialization file in the following format:

  ```bash
  . /net/machine-name/export/site-files/site-init-file
  ```

Bash has the following initialization files:

- `$HOME/.bash_profile`
- `$HOME/.bash_login`
- `$HOME/.profile`

The `/etc/profile` and `/etc/.login` files can be used to perform global definitions that would apply to all users of a system, in addition to any user specific definitions in user initialization files. However, these files still reside locally on the system, and therefore not centrally administered. The global definitions would not apply if the user goes to another system. Thus, for example, if AutoFS is used to mount the home directory from any system on the network, you would have to modify the system initialization files on each system to ensure a consistent...
environment whenever a user moved from system to system. As best practice, do not use these files.

**Default File Permissions (umask)**

When you create a file or directory, the default file permissions assigned to the file or directory are controlled by the *user mask*. The user mask is set by the `umask` command in a user initialization file. You can display the current value of the user mask by typing `umask` and pressing Return.

The user mask contains the following octal values:

- The first digit sets permissions for the user
- The second digit sets permissions for group
- The third digit sets permissions for other, also referred to as world

To determine the `umask` value that you want to set, subtract the value of the permissions you want from 666 (for a file) or 777 (for a directory). The remainder is the value to use with the `umask` command. Suppose that you want to change the default mode for files to 644 (`rw-r--rw--`). The difference between 666 and 644 is 022, which is the value you would use as an argument to the `umask` command.

The following table provides `umask` values. It shows the file and directory permissions that are created for each of the octal values of `umask`.

<table>
<thead>
<tr>
<th>umask Octal Value</th>
<th>File Permissions</th>
<th>Directory Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>rw-</td>
<td>rwx</td>
</tr>
<tr>
<td>1</td>
<td>rw-</td>
<td>rw-</td>
</tr>
<tr>
<td>2</td>
<td>r--</td>
<td>r-x</td>
</tr>
<tr>
<td>3</td>
<td>r--</td>
<td>r--</td>
</tr>
<tr>
<td>4</td>
<td>-w-</td>
<td>-wx</td>
</tr>
<tr>
<td>5</td>
<td>-w-</td>
<td>-w-</td>
</tr>
<tr>
<td>6</td>
<td>--x</td>
<td>--x</td>
</tr>
<tr>
<td>7</td>
<td>- - - (none)</td>
<td>- - - (none)</td>
</tr>
</tbody>
</table>

The following line in a user initialization file sets the default file permissions to `rw-rw-rw-`. 
Managing Users by Using Oracle Enterprise Manager Ops Center

If you are managing physical and virtual operating systems, servers, and storage devices within a large deployment, rather than just managing individual systems, you can use the management solutions available in the Oracle Enterprise Manager Ops Center.

With the Enterprise Manager Ops Center you can manage users and roles for the overall data center. You can add existing local users from your individual systems as users in the Ops Center, and you can control what assets and features these users are authorized to use.

For information, see the appropriate documentation at https://docs.oracle.com/en/enterprise-manager/.
Managing User Accounts by Using the Command-Line Interface

This chapter provides basic information for setting up and managing user accounts by using the command-line interface (CLI). It covers the following topics:

- “Setting Up and Managing User Accounts (Task Map)” on page 21
- “Setting Up User Accounts by Using the CLI” on page 22
- “Managing User Accounts by Using CLI” on page 25

Setting Up and Managing User Accounts (Task Map)

The following tasks describe how to set up and manage user accounts by using the command-line interface (CLI).

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gather user information</td>
<td>Use a standard form to gather user information to help you keep user information organized.</td>
<td>“Gathering User Information” on page 23</td>
</tr>
<tr>
<td>Customize user initialization files</td>
<td>Set up user initialization files to provide new users with consistent environments.</td>
<td>“How to Customize User Initialization Files” on page 23</td>
</tr>
<tr>
<td>Change account defaults for all roles</td>
<td>Change the default home directory and skeleton directory for all roles.</td>
<td>“How to Change Account Defaults for All Roles” on page 24</td>
</tr>
<tr>
<td>Create a user account</td>
<td>Create a local user by using the useradd command with the account defaults that you have set up.</td>
<td>“How to Add a User” on page 25</td>
</tr>
<tr>
<td>Modify a user account</td>
<td>Modify the login information of a user on the system.</td>
<td>“How to Modify a User Account” on page 27</td>
</tr>
<tr>
<td>Delete a user account</td>
<td>Delete a user account by using the userdel command.</td>
<td>“How to Delete a User” on page 28</td>
</tr>
</tbody>
</table>
Setting Up User Accounts by Using the CLI

This section discusses ways of preparing for the creation of user accounts.

About Setting Up User Accounts

Oracle Solaris typically creates a user account home directory as an individual ZFS file system and a ZFS dataset that are mounted under the /export/home file system. As a result, you can back up each home directory, create a ZFS snapshot of it, and replace its contents from a snapshot.

To administer a user account, you must have the appropriate rights profile such as the User Management rights profile. The tasks in this guide assumes that you have these rights. For

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>For Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unlock a User Account</td>
<td>Unlock a user account by using the passwd -u command.</td>
<td>“How to Unlock a User Account” on page 27</td>
</tr>
<tr>
<td>Create, then assign a role to perform an</td>
<td>Create a local role to enable the user to perform specific administrative</td>
<td>“Creating a Role” in Securing Users and Processes in Oracle Solaris 11.4</td>
</tr>
<tr>
<td>administrative task</td>
<td>commands or tasks with the account defaults that you have set up.</td>
<td></td>
</tr>
<tr>
<td>Create a group</td>
<td>Create a new group by using the groupadd command.</td>
<td>“How to Add a Group” on page 29</td>
</tr>
<tr>
<td>Create home directories for users</td>
<td>Create the home directory for users without creating a ZFS dataset for each</td>
<td>“How to Create the Home Directory for a User Without Creating a ZFS Dataset” on</td>
</tr>
<tr>
<td>without creating a ZFS dataset</td>
<td>user.</td>
<td>page 30</td>
</tr>
<tr>
<td>Add qualified user attributes to user</td>
<td>Use the default@ account name to assign qualified user attributes to user</td>
<td>“How to Assign Default User Attributes for LDAP Accounts” on page 31</td>
</tr>
<tr>
<td>accounts in the LDAP directory</td>
<td>accounts in the LDAP directory.</td>
<td></td>
</tr>
<tr>
<td>Add security attributes to a user account</td>
<td>Add the required security attributes after setting up the local user accounts.</td>
<td>“Creating a Role” in Securing Users and Processes in Oracle Solaris 11.4</td>
</tr>
<tr>
<td>Share a user’s home directory</td>
<td>Share the home directory of the user in order to remotely mount the directory from the system of the user.</td>
<td>“How to Share Home Directories That Are Created as ZFS File Systems” on page 32</td>
</tr>
<tr>
<td>Manually mount a user’s home directory</td>
<td>Manually mount the home directory of the user. Typically, you do not need to manually mount user home directories that are created as a ZFS file system. The home directory is mounted automatically when it is created and also at boot time from the SMF local file system service.</td>
<td>“Manually Mounting a User’s Home Directory” on page 34</td>
</tr>
</tbody>
</table>

About Setting Up User Accounts

Oracle Solaris typically creates a user account home directory as an individual ZFS file system and a ZFS dataset that are mounted under the /export/home file system. As a result, you can back up each home directory, create a ZFS snapshot of it, and replace its contents from a snapshot.

To administer a user account, you must have the appropriate rights profile such as the User Management rights profile. The tasks in this guide assumes that you have these rights. For
more information, see “Using Your Assigned Administrative Rights” in *Securing Users and Processes in Oracle Solaris 11.4*.

If you are setting up user accounts by using LDAP, you would need to specify LDAP as the repository for user information. Then you can assign user attributes to the user accounts. For more information, see “How to Assign Default User Attributes for LDAP Accounts” on page 31.

For security, you can label processes and files that would control user access to sensitive files. See Chapter 6, “Labeling Processes for Data Loss Protection” in *Securing Users and Processes in Oracle Solaris 11.4*.

### Gathering User Information

When setting up user accounts, you would need to gather the following information about each user.

<table>
<thead>
<tr>
<th>User Name</th>
<th>Password Status and Aging</th>
</tr>
</thead>
<tbody>
<tr>
<td>Role Name</td>
<td>Home Directory Path Name</td>
</tr>
<tr>
<td>Profiles or Authorization</td>
<td>Mounting Method</td>
</tr>
<tr>
<td>UID</td>
<td>Permissions on Home Directory</td>
</tr>
<tr>
<td>Primary Group</td>
<td>Mail Server</td>
</tr>
<tr>
<td>Secondary Groups</td>
<td>Add to These Email Aliases</td>
</tr>
<tr>
<td>Default Shell</td>
<td>Desktop System Name</td>
</tr>
</tbody>
</table>

### Identifying Users by Packages

To find all the users that were delivered by a package (do not represent humans) on this system, you must specify the --l option. Note that this search excludes user packages created manually by the useradd command.

```bash
$ pkg search -lo username, pkg.name user::
```

**Caution** - Do not try to change the users' output with the `pkg search` command.

### How to Customize User Initialization Files

The following task describes how to set up customized initialization files for the users on your system.
1. **Become an administrator or a user with the User Management rights profile.**

   ```
   $ su -
   Password: #
   ```

   For more information, see “Using Your Assigned Administrative Rights” in Securing Users and Processes in Oracle Solaris 11.4.

2. **Create a skeleton directory for each type of user.**

   ```
   # mkdir /shared-dir/skel/user-type
   ```

   - `shared-dir` The name of a directory that is available to other systems on the network
   - `user-type` The name of a directory to store initialization files for a type of user

3. **Copy the default user initialization files into the directories that you created for different types of users.**

4. **Customize the user initialization files for each user type.**

   For an overview, see “About the User Work Environment” on page 17.

5. **Set the permissions for the user initialization files.**

   ```
   # chmod 744 /shared-dir/skel/user-type/.*
   ```

6. **Verify that the permissions for the user initialization files are correct.**

   ```
   # ls -la /shared-dir/skel/*
   ```

---

**How to Change Account Defaults for All Roles**

In the following procedure, the administrator customizes a `roles` directory. The administrator then changes the default home directory and skeleton directory for all roles.

1. **Become an administrator or a user with the User Management rights profile.**


2. **Create a custom `roles` directory.**
Managing User Accounts by Using CLI

For example:

```
# roleadd -D
group=other,1  project=default,3  basedir=/home
skel=/etc/skel  shell=/bin/pfsh  inactive=0
expire=  auths=  profiles=All  limitpriv=
defaultpriv=  lock_after_retries=
```

3. **Change the default home directory and skeleton directory for all roles.**

For example:

```
# roleadd -D -b /export/home -k /etc/skel/roles
# roleadd -D

group=staff,10  project=default,3  basedir=/export/home
skel=/etc/skel/roles  shell=/bin/sh  inactive=0
expire=  auths=  profiles=  roles=  limitpriv=
defaultpriv=  lock_after_retries=
```

Further use of the `roleadd` command will now create home directories in the `/export/home` directory and will populate the roles' environment from the `/etc/skel/roles` directory.

Managing User Accounts by Using CLI

This section describes different procedures for managing user accounts.

▼ **How to Add a User**

1. **Become an administrator or a user with the User Management rights profile.**

   See “Using Your Assigned Administrative Rights” in *Securing Users and Processes in Oracle Solaris 11.4*.

2. **Create a local user.**

   ```
   # useradd -d dir -m [-S ldap] username -z no|yes|nodelegation
   ```

   `-ddir` Specifies the location of the home directory of the user. An optional server name can be added to the directory path, such as `system1:/home/export/jdoe`. The host information is written to `auto_home` and used by the automounter to mount the home directory.
How to Add a User

-m
Creates a local home directory on the system for the user if the directory does not yet exist. However, if the -d option specifies a host that is remote, then the home directory is not created.

If the directory does not yet exist, the home directory is created under the parent directory which is assumed as the the mount point of a ZFS dataset, such as export/home. The newly created directory is created as a multilevel dataset.

-S ldap
This option indicates that you are using LDAP and its repository for the account information. If you use LDAP, you can also assign default attributes by to the default@ user name which are applied to all users. For example:

$ useradd -S -K user-attributes default@

Multiple values can be specified for user-attributes and use the format key=value. These attributes are detailed in the user_attr(5) man page. For more information, see “How to Assign Default User Attributes for LDAP Accounts” on page 31.

-z yes|no|
nodelegation
Specifies the default behavior of the useradd command when using the -z option.

no
Specifies that a new user account has a home directory that is a directory rather than a ZFS file system

yes
Specifies that a new user account has a home directory that is a ZFS file system with delegations specified by the zfs allow snapshot,mount, create command

nodelegation
Specifies that a new user account has a home directory that is a ZFS file system with no delegations

See Also
After creating a user, you might need to perform some additional tasks, including adding and assigning roles to a user, and displaying or changing the rights profiles of a user. For more information, see “Creating a Role” in Securing Users and Processes in Oracle Solaris 11.4.
### How to Modify a User Account

The `usermod` command is used to change the definition of a user's login and make appropriate login-related file system changes for the user.

1. **Become an administrator or a user with the User Management rights profile.**

2. **Modify the user account, as required.**
   For example, to add a role to a user, you would type:
   ```bash
   # usermod -R role username
   ```
   See the `usermod(8)` man page for details about the arguments and options that you can specify with the `usermod` command.

#### Example 1
**Setting Per-User PAM Policy by Modifying a User's Account**

This example shows how to modify a user to set PAM policy. This particular modification specifies that user jdoe should only be authenticated with the Kerberos V5 protocol for all PAM services. For more information, see the `pam_user_policy(7)` man page.

```bash
# usermod -K pam_policy=krb5_only jdoe
```

**See Also**
For additional information, see “Creating a Role” in Securing Users and Processes in Oracle Solaris 11.4.

### How to Unlock a User Account

1. **Become an administrator or a user with the User Security rights profile.**

2. **Check the status of the user account that you need to unlock.**
   ```bash
   $ passwd -s username
   username LK
   ```

3. **Unlock the user account.**
   ```bash
   $ passwd -u username
   ```
How to Delete a User

4. **Check if the desired user account has been unlocked.**

   $ passwd -s username

   **Note** - For more information about unlocking a user account, see “Guidelines for Assigning User Names, UIDs, and GIDs” on page 12 and the `passwd(1)` man page.

### How to Delete a User

1. **Become an administrator.**

   $ su -
   
   Password:
   
   #

   **Note** - This method works whether `root` is a user account or a role.

2. **Archive the home directory of the user.**

3. **Delete the user.**

   Use one of the following commands:

   - **Use the `useradm` command.**
     
     `useradm delete [-S [files | ldap]] username`

   - **Use the `userdel` command.**
     
     `userdel -r username`

     The `-r` option removes the account from the system.

     The preferred method for removing a local home directory for a deleted user is to specify the `-r` option with the `userdel` command. This method is preferred because user home directories are now ZFS datasets. If the user's home directory is on a remote server, manually delete it:

     `userdel username`

     For a full list of command options, see the `userdel(8)` man page.
4. **Confirm if the user account has been deleted.**

   ```bash
   $ useradm list [-$ [files | ldap]] username
   ```

**Next Steps**

Additional cleanup might be required if the user that you deleted had administrative responsibilities, for example creating `cron` jobs, or if the user had additional accounts in non-global zones.

---

**How to Add a Group**

When an administrator creates a group, the system assigns the `solaris.group.assign/groupname` to that administrator, giving the administrator complete control over that group. If another administrator who has the same authorization creates a group, that administrator has the control over that group. An administrator who has control of one group cannot administer the group of the other administrator. For more information, see the `groupadd(8)` and `groupmod(8)` man pages.

1. **Become an administrator with the root role or an administrator who has the `solaris.group.manage` authorization.**
   
   See “Using Your Assigned Administrative Rights” in *Securing Users and Processes in Oracle Solaris 11.4*.

2. **List the existing groups.**

   ```bash
   # cat /etc/group
   ```

3. **Create a new group.**

   ```bash
   $ groupadd -g group-ID group-name
   ```

   `-g` Assigns the group ID for the new group

   For more information, see the `groupadd(8)` man page.

**Example 2** Setting Up a Group and User With the `groupadd` and `useradd` Commands

This example shows how to use the `groupadd` and `useradd` commands to add the group `scutters` and the user `scutter1` to files on the local system.

```bash
# groupadd -g 102 cutters
# useradd -u 1003 -g 102 -d /export/home/scutter1 -s /bin/csh \ 
  -c "Scutter 1" -m -k /etc/skel scutter1
```

64 blocks
How to Create the Home Directory for a User Without Creating a ZFS Dataset

1. Become an administrator.

2. Add a new user account and specify the base directory for the same.
   ```
   # useradd -z no -m username
   80 blocks
   ```

3. Assign a password to the new user account.
   ```
   # passwd username
   New Password:
   Re-enter new Password:
   passwd: password successfully changed for username
   ```

   **Note** - Make sure you have the User Security rights profile assigned in order to assign passwords.

4. The home directory of the user account is created in the /export/home directory.
   ```
   # cat /etc/passwd
   root:x:0:0:Super-User:/root:/usr/bin/bash
   daemon:x:1:1::/sbin/sh
   bin:x:2:2::/bin/sh
   sys:x:3:3::/sbin/sh
   adm:x:4:4:Admin:/var/adm:/bin/sh
   dladm:x:15:65:Datalink Admin::
   netadm:x:16:65:Network Admin::
   netcfg:x:17:65:Network Configuration Admin::
   sshd:x:22:22:sshd privsep:/var/empty:/bin/false
   smmsp:x:25:25:SendMail Message Submission Program::
   username:x:167:10::/export/home/username:/usr/bin/bash
   ```

**Example 3** Creating a User Account With a Home Directory That Is a Directory

Use the following command to create a user account with a home directory that is a directory rather than a ZFS file system:
Example 4  Creating a User Account With a Home Directory That Is a ZFS File System With Delegations

Use the following command to create a user account with a home directory that is a ZFS file system with the mount, create, and snapshot delegations:

```
# useradd -D -z yes
```

Example 5  Creating a User Account With a Home Directory That Is a ZFS File System Without Delegations

Use the following command to create a user account with a home directory that is a ZFS file system without delegations:

```
# useradd -D -z nodelegation
```

**How to Assign Default User Attributes for LDAP Accounts**

For LDAP accounts, an administrator can assign qualified user attributes to users by using the default account name default@. The useradd, usermod, and userdel commands have been modified to support this account name. The following example shows how to create a default account and assign default user attributes to users based on their netgroup membership.

1. **Become an administrator.**

2. **Create the default and test accounts in the LDAP name service.**

   ```
   # useradd -S ldap default@
   # useradd -S ldap -m admin1
   # useradd -S ldap -m admin2
   # useradd -S ldap -m admin3
   ```

3. **Create two netgroups.**

   ```
   # cat netgroups
dba (,admin1,) (,admin2,)
web (,admin3)
   # ldapaddent -D cn=admin,ou=profile,dc=gfaden,dc=com -f netgroups netgroup
   ```
How to Share Home Directories That Are Created as ZFS File Systems

4. Assign the netgroups as qualifiers to the default account.

   # usermod -q @dba -K profiles="Oracle DBA" default@
   # usermod -q @web -K profiles="Web Admin" default@

5. Assign the user_attr entries.

   # ldapaddent -d user_attr |grep ^default@
   default@:@dba:::profiles=Oracle DBA
   default@:@web:::profiles=Web Admin

6. List the effective user_attr for each user.

   # getent user_attr admin1
   default@:@dba:::profiles=Oracle DBA

   # getent user_attr admin2
   default@:@dba:::profiles=Oracle DBA

   # getent user_attr admin3
   default@:@web:::profiles=Web Admin

Sharing ZFS File Systems

You can share a ZFS file system by setting the share.nfs property or the share.smb property. Or, you can create a file system share by using the zfs share command. By default, all file systems are unshared.

You can assign a clearance label to each user, which restricts users' access to confidential process information. You set the value for clearance label in the encodings file. When you assign a specific clearance to a user you should also assign the user a multilevel home directory as shown in the next procedure.

About sharing and unsharing file systems, see also “Autofs Administration” in Managing Network File Systems in Oracle Solaris 11.4.

▼ How to Share Home Directories That Are Created as ZFS File Systems

1. Become an administrator.
See “Using Your Assigned Administrative Rights” in *Securing Users and Processes in Oracle Solaris 11.4*.

2. **Create a separate pool for the user home directories.**
   
   ```
   # zpool create pool mirror disk 1 disk 2 mirror disk 3 disk 4
   
   For example:
   
   # zpool create users mirror c1t1d0 c1t2d0 mirror c2t1d0 c2t2d0
   ```

3. **Create a container for the home directories.**
   
   ```
   # zfs create filesystem
   
   For example:
   
   # zfs create users/home
   ```

4. **(Optional) Assign a multilevel home directory to users who have been assigned a specific clearance.**
   
   ```
   zfs create -o multilevel=on filesystem
   
   For more information, see the `clearance(7)` man page.
   ```

5. **Set the share properties for the home directory.**
   
   For example, to create an NFS share and set the `share.nfs` property for `users/home`, you would type:
   
   ```
   # zfs set share.nfs=on users/home
   
   When using this new syntax, each file system contains an "auto share" that is created as soon as the `share.nfs` property (or the `share.smb` property) is set to on for that file system. The previous command shares a file system named `users/home` and all of its children.
   ```

6. **(Optional) Enable labeled sharing when sharing the file system.**
   
   If you assign a multilevel home directory to a user, then you must enable labeled sharing while sharing file systems.
   
   ```
   zfs set share.nfs=on share.nfs.labeled=on filesystem
   ```

7. **Confirm that the descendent file system shares are also published.**
   
   For example:
   
   ```
   # zfs get -r share.nfs users/home
   ```
The `-r` option displays all of the descendent file systems.

**Manually Mounting a User's Home Directory**

With ZFS, file systems that are newly created are automatically mounted at boot time from the SMF local file system service.

When creating user accounts, make sure home directories are set up as they are in the name service, at `/home/username`. Then, make sure that the `auto_home` map indicates the NFS path to the user's home directory. For task-related information, see “Autofs Administration” in *Managing Network File Systems in Oracle Solaris 11.4*.

If you need to manually mount a user's home directory, use the `zfs mount` command. For example:

```
# zfs mount users/home/jdoe
```

**Note** - Make sure that the user's home directory is shared. For more information, see “How to Share Home Directories That Are Created as ZFS File Systems” on page 32.
Managing User Accounts Interactively

This chapter describes how to set up and manage user accounts by using two new applications which provide menu-based user management. It covers the following topics:

- “Security Considerations”
- “About the useradm Application”
- “About the Oracle Solaris Account Management BUI”

Security Considerations

The Oracle Solaris Account Manager BUI and the useradm application have the same security policy. Neither application requires any special privileges. The authorizations associated with the account that is used for the RAD connection determines the attributes that can be viewed or modified. A user account with the User Security rights profile can assign any of its user attributes (roles, profiles, labels, privileges, and authorizations) to other users. An account with all authorizations such as the root role can create new accounts and assign any attribute to them.

In order to assign user attributes which are not currently assigned to you, the corresponding solaris.*.assign authorization needs to be added to your user attributes. For example, to assign additional profiles to a user, the user who invokes the useradm application must have the solaris.profile.assign authorization. For more information, see “User Rights Management” in Securing Users and Processes in Oracle Solaris 11.4.

About the useradm Application

The useradm tool can be used as a command line or interactively. This section focuses on the interactive mode of that tool.
The interactive useradm application replaces the Visual Panels User Manager. This application is a remote administration daemon (RAD) client which provides a menu-driven interface for setting up and managing user accounts. Though the application connects to local RAD servers by default, it can connect to remote RAD servers too. The application provides easier user management, password management, and role management options. Any terminal emulator like putty or the Secure Shell (ssh) service can be used to start this application locally or remotely.

Whether in command line mode or interactive mode, use of useradm requires that you must be an administrator or assume a role which has been assigned the User Security rights profile. See “Using Your Assigned Administrative Rights” in Securing Users and Processes in Oracle Solaris 11.4.

For more details about the useradm command, see the useradm(8) man page.

▼ How to Install the useradm Application

The useradm application is not included in a regular Oracle Solaris installation. To use the application, install the package first.

1. **Become the administrator or assume the role that has been assigned the appropriate installation rights.**

2. **Install the useradm package.**

   $ pkg install useradm

3. **Set the TERM environment variable to xterm-256color.**

   $ export TERM=xterm-256color

4. **If using the application remotely, enable the rad:remote SMF service on the remote system.**

   $ svcadm enable rad:remote

Using the Interactive useradm Command

The interactive mode of useradm is activated only by using the add or modify subcommand and with only the -S, -q, or -R options, used singly or in combination.
-S specifies the naming service repository to use. The choices are files to store account information locally or ldap to have the information managed by LDAP.

-q specifies the hostname or netgroup to use for the attributes maintained in the user_attr LDAP container. This option applies only to existing accounts that are maintained by the LDAP name service.

-R specifies the URI to connect to a remote RAD server.

All the following sample commands open the useradm interactive window.

- Creating a new user account jdoe. The information is stored locally.

  
  $ useradm add -S files jdoe

- Modifying user jsmith's attribute settings stored in LDAP.

  
  $ useradm modify -S ldap jsmith

- Modifying user jack's account information in LDAP so that the attributes are applicable only when the user is using the system with the host name system1.

  
  $ usermod modify -S ldap -q system1 jack

Likewise, the following commands connect to a remote server and run useradm interactively. Ensure that the rad:remote SMF service is enabled.

- Starting the useradm application locally while specifying a remote RAD server.

  
  $ useradm modify -S ldap -R ssh://login-name@server johnsmith@example.com

- Starting the useradm application directly on the remote server.

  
  $ ssh joe@server -t useradm modify jane

- Using two systems to manage user accounts remotely.

  
  $ ssh -t jane@server1 useradm modify -R rads://jean@server2?auth=pam mary

In this example, the application is started on one server to modify a user on another server. The user account to log in to the server specified with -R option – jean in this example – is used to manage mary's account settings. Ensure that jean is assigned either the User Security rights profile or has the solaris.auth.delegate authorization on server2.

**Attributes in the useradm Window**

When you use the useradm command as previously explained, the following window appears:
How to Install the useradm Application

The following list explains the different options or attributes you can set for users:

Access Times
Specify the days and times at which specific services can be accessed.

Account Type
Specify whether the account you are working with is a normal user account or a role.

Audit Flags
Specify the audit preselection flags of the user.

Annotation
Specify whether the user must provide an explanation if they are assuming a role or using an authenticated rights profile.

Authorizations
Assign authorizations to user accounts from a list of available options. The names of granted authorizations are listed under the Assigned Groups list.

Groups
Assign a primary group and secondary group to user accounts. Available choices in the Group list depends on your system's configuration. When you specify `-S files`, the list of groups that is displayed is the local list. Otherwise, the list of groups is a combination of local and LDAP groups.

Home Directory
(Optional) If you do not provide any information, the system automatically assigns a default home directory.
How to Manage Users and Roles Interactively

After the interactive window appears, perform the following steps to configure user attributes.

1. **Select the attributes you want to configure for the user.**
   
   For a description of the attributes, see “Attributes in the useradm Window” on page 37.

   **Tip** - To quickly go to the attribute of your choice, type the first few letters of the attribute.

   Each attribute you select opens additional submenu options.

2. **From the attribute’s submenu, select the setting for that attribute.**
   
   Configure as many attributes as you want.

3. **To save your changes, select Commit from the main menu and press Enter.**
About the Oracle Solaris Account Management BUI

4. Select Exit to return to the command line interface.

5. (Optional) Confirm that all the modifications have been saved.

$ useradm list [-S [files | ldap]] username

About the Oracle Solaris Account Management BUI

The Oracle Solaris Account Management browser user interface (BUI) provides a browser-based account manager which has the same functionality as the useradm application. You can access this BUI using the Oracle Solaris Dashboard. This BUI supports all the functionalities of useradd, usermod, userdel, roleadd, rolemod, passwd, and login commands through its interactive menu. It is included in the system/management/webui/webui-usermgr package.

▼ How to Access the Oracle Solaris Account Management BUI

Before You Begin

Make sure that you have the appropriate authorizations to install the package to access the BUI.


1. Install the system/management/webui/webui-usermgr package.

$ pkg install webui-usermgr

2. On your browser, go to the following location:

https://hostname:6787/solaris

3. Log in to the Oracle Solaris Dashboard.

4. From the "Applications" menu, select "Solaris Account Management".
The following screen is displayed:

Note - The initial display of either of the two labeling dialogs (Minimum Label or Clearance) might be rendered incorrectly. As a workaround, select the other labeling menu item, and then reselect the previous item.
Index

A
Account Management BUI
introduction, 40
Account Manager BUI
for managing user accounts, 35
adding
    groups
        through the CLI, 29
    roles
        through the CLI, 25
    users
        through the CLI, 25
adding users
    interactively
        using useradm, 37
administering
    accounts, 24
    groups, 29
    users, 25, 28
aging user passwords, 14
automounting user home directories, 18
controlling file and directory access, 19

D
daemon group, 10
default file permissions, 19
default user attributes for LDAP accounts, 31
defaults
    file permissions, 19
    setting for users and roles, 24
deleting
    users
        using the CLI, 28
description of user login names, 10
directories
    controlling access to, 19
    home, 12
    changing default, 24
    sharing a ZFS file system, 32
displaying the user mask, 19

E
encryption, 14
/etc files
    user account information, 12
    user account information and, 12
/etc/passwd file, 14, 14
    description, 14
    user ID number assignment and, 13
/etc/shadow file description, 14
/export/home file system, 12
Index

F
file defaults permissions, 19

G
group file
  description, 14
  fields in, 15
  group ID numbers, 10, 11, 11, 13
groupadd command, 16, 29
groupe1 command, 16
grouppmod command, 16
groups
  adding, 29
  changing primary, 17
  default, 11
  description, 11
  displaying groups a user belongs to, 17
  guidelines for managing, 11, 11
  ID numbers, 10, 11, 11
  names, 11
  naming services and, 11
  primary, 11, 11
  secondary, 11, 11
  storage of information for, 14, 15
  UNIX, 11
  groups command, 17

H
home directories See user home directories
  /home file system
  user home directories and, 12

I
ID numbers
  group, 10, 11, 11
  user, 13
initialization files
  customizing
  site initialization files, 18
  system, 18

M
managing user accounts
  Account Manager BUI, 35
  using the useradm application, 35
maximums
  secondary groups users can belong to, 11
  user ID number, 10
  user login name length, 12
modifying
  users
    interactively, 37
    using the useradm application, 37
mounting
  user home directories, 34
  mounting user home directories, 18

N
names
  group, 11
  naming services
    groups and, 11
    user accounts and, 12, 14
newgrp command, 17
noaccess user/group, 10
nobody user/group, 10

O
Oracle Solaris
  Account Management BUI, 40

P
passwd command
  assigning user password with, 25
unlocking user, 27
passwd file
  fields in, 14
  user ID number assignment and, 13
passwords (user)
  aging, 14
  assigning to users, 25
  changing
    frequency of, 12
    by user, 12
  choosing, 12
  encryption, 14
  precautions, 11
primary groups, 11, 11
pseudo user logins, 13
pseudo-ttys, 13

R
removing user's home directory, 28
roleadd command, 16
roleadd command for setting account defaults, 24
roledel command, 16
rolemod command, 16

S
secondary groups, 11, 11
security considerations
  for interactive applications, 35
shadow file
  description, 14
  fields in, 15
site initialization files, 18
staff group, 11
system accounts, 10
system initialization files, 18

T
ttys (pseudo), 13
ttytype pseudo user logins, 13

U
UIDs
  assigning, 13
  definition, 10
  large, 13
umask command, 19
UNIX groups, 11
unlocking
  users
    using the CLI, 27
user accounts, 10
  description, 10
  description of, 10
  gathering information for, 23
  guidelines for, 18
  ID numbers, 10, 13
  login names, 10
  naming services and, 12, 14
  storage of information for, 12, 12
user groups, 10
user home directories
  automounting, 18
  description, 12
  mounting, 18, 34
  removing, 28, 28
user ID number reuse and security, 13
user ID numbers, 10, 13
user initialization files
  customizing, 18
  overview, 18
  user mask setting, 19
  description, 18
user login names
  length of, 12
user logins (pseudo), 13
user mask, 19
user packages to identify users, 23
useradd command, 16
  adding user, 25
  setting account defaults, 24
useradm application
   adding a user, 37
   attribute options, 37
   for managing user accounts, 35
   introduction, 35, 35
   manage users and roles, 39
   modify a user, 37
   remote systems, 39
   start the interface, 36, 37
userdel command, 16
   deleting user, 28
usermod command, 16
users
   adding, 25, 28
   managing in Ops Center, 20
   removing home directories, 28
   setting account defaults, 24
   unlocking, 27
uucp group, 13

Z
ZFS file systems
   sharing, 32
   user accounts as, 22