Running Perceptive Content as a Non-Root User

Best Practices Guide

Perceptive Content Version: 7.0.x

Written by: Product Knowledge, R&D
Date: October 2014

perceptivesoftware
from Lexmark

Table of Contents

About Running Perceptive Content as a Non-Root User

This guide defines the best practices for installing and running Perceptive Content under an account other than root on Linux and UNIX using shadowed system authentication. If you are already running Perceptive Content, you can also use these best practices to help promote a more secure server environment. To determine if your system is using shadowed authentication, or to understand more about how it works, contact your UNIX administrator or an Perceptive Content support representative.

You should run Perceptive Content under an account other than root if any of the following situations apply to you:

- You do not have access to the root account.
- You are concerned about inherent security issues of running Perceptive Content as root.

To ensure proper configuration and to maintain security, only a qualified Information Technology professional, such as a system or UNIX administrator, should determine which configurations are best for your needs and perform the configuration options described in this guide.

Running Perceptive Content as a Non-Root User

Before running Perceptive Content Server as a non-root user on a shadowed system, you need to configure Role Based Access Control (RBAC) or an Access Control List (ACL).

If your Linux distributor offers RBAC as a supported package or embeds it into the Linux Kernel, you can use the configuration options detailed in the following sections of this document. Otherwise, to achieve rootless authentication, you need to download a third party RBAC kernel module from a trusted source. Verify that the RBAC kernel provides the roles necessary to read your shadowed **passwd** file, and can provide read access to /**dev/mem**. While you can grant ACL read privileges to **/etc/shadow**, a kernel module/patch is required to grant read privileges to **/dev/mem** and cannot be granted with ACL privileges alone.

If you are unable to locate a trustable source or you are concerned about security issues with downloading a third party RBAC kernel module, you can use the built in security features of Perceptive Content to release root privileges and run as another user after server initialization. To use these built in security features, you need to configure daemons to run on a non-root user account.

Configure Role Based Access Control

Most supported UNIX systems provide a Role Based Access Control (RBAC) package and some systems, like Solaris, install it by default. Because RBAC configuration varies in each UNIX platform, your UNIX administrator should configure RBAC. To configure your RBAC package, complete the following steps.

Note The following steps show an example of configuring RBAC for Solaris 10 with a username of imgnow.

1. As root, create a new user: **imgnow**.
2. Use the following command to grant the **file_dac_read** privilege to **imgnow**.

```
usermod -K defaultpriv=basic,file_dac_read imgnow
```

3. Log in as **imgnow** and verify your privileges using the following substeps.
    1. View the privileges using the ppriv $$ command.
    2. Try to access the shadow file using the cat /etc/shadow command.

Configure an Access Control List

All supported UNIX systems provide an Access Control List (ACL) package. Because ACL varies in each UNIX platform, your UNIX administrator should configure ACL. To configure an ACL, complete the following steps.

Note The following steps show an example of configuring ACL with a user name of imgnow.

1. As root, create a new user called **imgnow**.
2. As root, enter the following commands in your bash shell.

```
export EDITOR=/usr/bin/vi
acledit /etc/security
```

3. Under **extended permissions**, remove **disabled** and add the following text.

```
enabled
permit r-x u:imgnow
```

4. Using the same shell session, edit permissions for the following file.

```
acledit /etc/security/passwd
```

5. Under **extended permissions**, remove **disabled** and add the following text.

```
enabled
permit r-- u:imgnow
```

6. Log in as **imgnow** and verify your privileges using the following substeps.
    1. View the privileges using the aclget /etc/security/passwd command.
    2. Try to access the shadow file using the cat /etc/security/passwd command.

Configure daemons to run on a non-root user account

Perceptive Content has built in security features to release root privileges and run as another user after server initialization. To configure the daemons to run on a non-root user, complete the following steps.

1. When you install Perceptive Content, change the ownership of all files to **<*username*>:bin**, where <*username*> is the user you want as the owner instead of root. The following example changes the ownership of all files to the user **imgnow**.

```
chown -R imgnow:bin ./inserver
```

2. Open the **inow.ini** file in a text editor.

3. Enter the following text in the INI file.

```
[Daemon]
daemon.user.id = <userid>
```

   Where <*userid*> is the user ID of the account you want as the owner instead of root.

   Note The daemon.user.id value is the number representation of <userid>, not the user name itself.

4. Save the **inow.ini** file and start **Perceptive Content** as root. The daemons switch to the configured user after performing the tasks they need to access /dev/mem.

Start Perceptive Content as a non-root user

There are security risks with running Perceptive Content under your root user account. Linux and UNIX systems always start at the root user level. To start Perceptive Content as a non-root user, complete the following steps.

Prerequisite Configure your Role Based Access Control (RBAC) or Access Control List (ACL), or configure daemons to run on a non-root user account

1. When you install Perceptive Content, change the ownership of all files to **<*username*>:bin**, where <*username*> is the user you want as the owner instead of root. The following example changes the ownership of all files to the user **imgnow**.

```
chown -R imgnow:bin ./inserver
```

2. In the **rc.local/init.d** startup script, run the daemons as the user you created. In the examples in this guide, the user is **imgnow**.