



FOR MAGENTO ENTERPRISE EDITION

Installing a Patch

Whenever a patch is released to fix an issue in the code, a notice is sent directly to your Admin [Inbox](#). If the update is security related, the incoming message is color-coded red, and marked as a “Critical Update.”

The following instructions explain how to download and install a patch, starting with a notice that appears in your Admin Inbox. The example takes place on a Windows system, and uses the [WinSCP](#) utility to upload patch files to the server, and [Putty](#) to access the server from the command line. You can download both utilities at no charge. If you are a Mac user, you can access the command line with [Terminal](#).

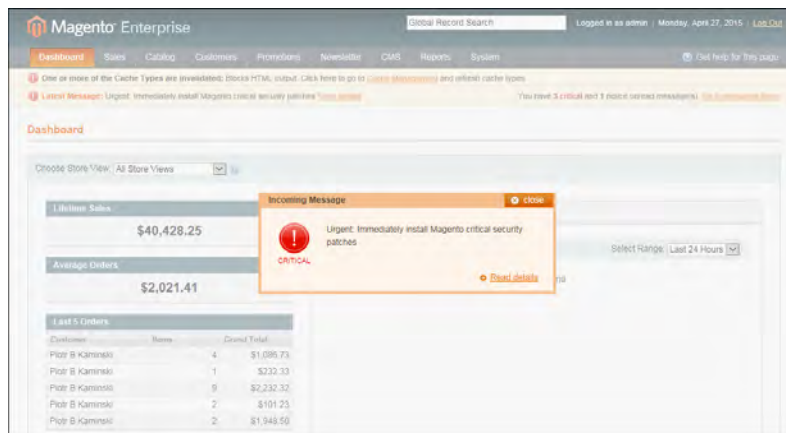
If this is the first time you have installed a patch, we recommend that you complete the optional steps to create a backup copy of your store and install a patch as a test run. If you have experience installing patches, you can take the fast track and skip the optional steps. For advanced instructions, see the following articles in the Magento developer documentation:

[How to Apply and Revert Magento Patches](#)

[Recommended File System Ownership and Privileges](#)

Before you begin...

To install a patch, you must have a user name and password to access the server. In addition to your own login credentials, you might also need the `apache` password. If you don't have these credentials, contact the person who set up your server.



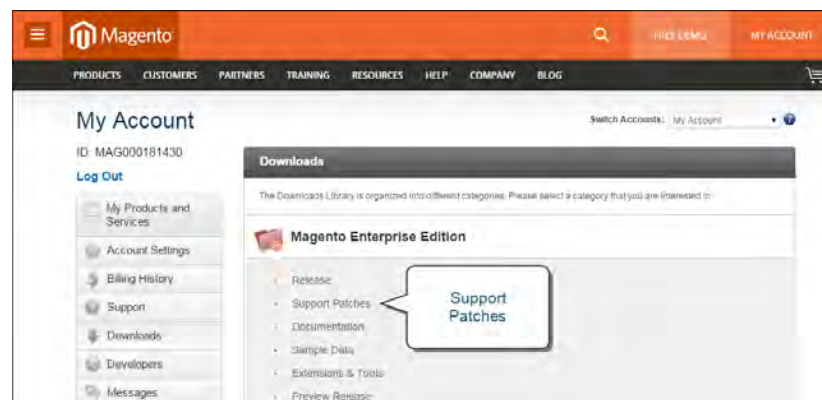
Incoming Message of Critical Importance

Process Overview:

- Step 1: [Download the Patch](#)
- Step 2: [Set the Necessary Permissions](#)
- Step 3: [Create a Backup of the Magento Folder](#) (Optional)
- Step 4: [Upload the Patch](#)
- Step 5: [Install the Patch on the Backup](#) (Optional)
- Step 6: [Install the Patch to Your Store](#)
- Step 7: [A Little Housekeeping](#) (Optional)
- Step 8: [Look for Signs of Unauthorized Access](#)
- Step 9: [Clear the Magento Cache](#)
- Step 10: [Recompile the Store](#) (Only if compiled)
- Step 11: [Restart the Server](#)

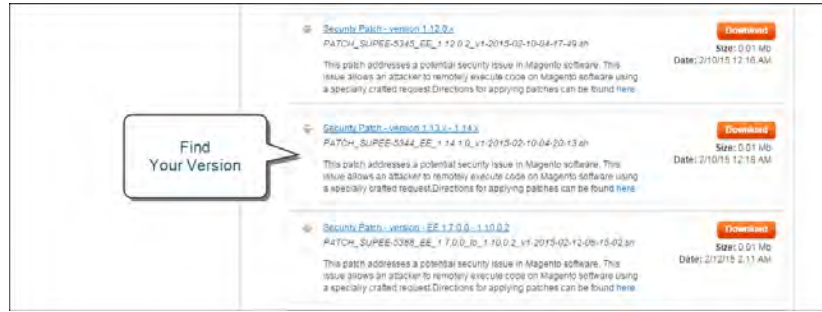
Step 1: Download the Patch

1. When you receive the message in your Inbox, click the **Read details** link to open the [account login](#) page. Then, log in to your account.
2. In the panel on the left, select **Downloads**. Click to expand the **Magento Enterprise Edition** section. Then, click **Support Patches**.



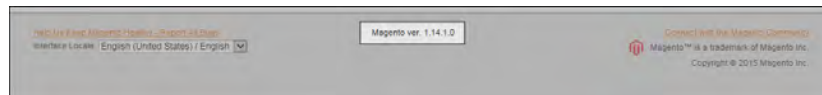
Your Magento Account

3. In the list, find the patch that you want to install. Then, click to expand the section.



Magento Enterprise Edition Patches

- 4. Find the patch for the specific version of Magento Enterprise that’s installed on your server, and click the **Download** button. If you don’t know which version of Magento Enterprise is installed on your server, you can find it in the footer of the Admin.



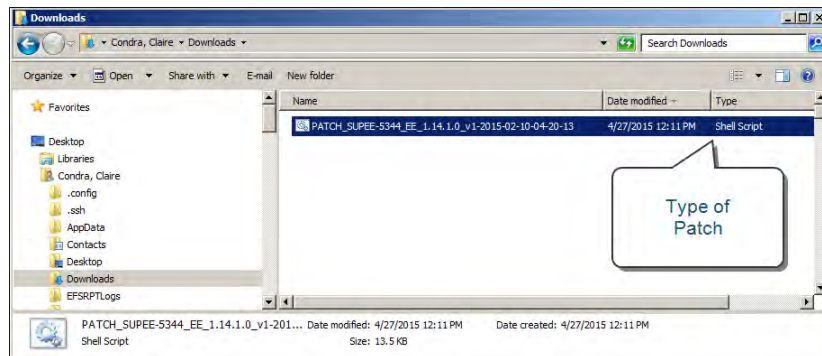
Magento Version in Footer

- a. Click the **Download** button. Then, look for the downloaded file in the lower-left corner of your browser.



Show Downloads Folder

- b. Click the **down arrow** next to the downloaded file name to display the menu. Then, select **Show in folder**. The patch is saved to the Download folder of your desktop computer.



Downloaded Patch Files

Step 2: Set the Necessary Permissions

For a live store, permissions are locked down to prevent unauthorized access. However, you must change the permissions of the contents of the Magento installation folder before you can install the patch. In the following instructions, Putty is used to access the command line and change the permissions.

1. Click the **Start** button in the lower-left corner of your desktop, and launch **Putty**. When prompted, enter the **Host Name** or **IP Address** of your store, and click the **Open** button.
2. Servers have different directory structures, and the path to your Magento installation folder is most likely different from the one shown in the examples. Depending on your server, the path to your Magento installation folder might be one of the following:

SERVER	PATH
Ubuntu	<code>/var/www/magento</code>
CentOS	<code>/var/www/html/magento</code>

From the command line, use the **Change Directory** command to navigate to your Magento installation folder. In this example, Magento is installed on a CentOS server, and the Magento installation folder is located three levels below the `html` folder.

```
cd /var/www/html/stores/ee/magento
```

3. List the directory to check the permission status of the `magento` folder. To install a patch, you must temporarily change the folder permissions. After the patch is installed, you can restore the appropriate permissions to your live store.

```
ls -l
```

The second column shows the current permission settings for the contents of the `magento` folder. It includes a combination of read only, read/write, and execute permissions, depending on the type of folder or file. Take note of the owner and group in the next two columns. You might not need this information, but it's good to know. In this example, each folder and file is owned by the `apache` user and belongs to the `root` group.

```

stack@m2-docs-staging-523722:/var/www/html/stores/ee/magento
total 548
-r----- 1 apache root 2834 Nov 9 14:30 api.php
dr-x----- 6 apache root 4096 Nov 9 15:08 app
-r----- 1 apache root 2827 Nov 9 15:09 cron.php
-r----- 1 apache root 717 Nov 9 14:30 cron.sh
dr-x----- 7 apache root 4096 Nov 9 15:09 downloader
dr-x----- 4 apache root 4096 Nov 9 15:09 errors
-r----- 1 apache root 1150 Nov 9 14:30 favicon.ico
dr-x----- 3 apache root 4096 Mar 7 01:50 frontend
-r----- 1 apache root 5979 Nov 9 14:30 get.php
dr-x----- 2 apache root 4096 Nov 9 15:09 includes
-r----- 1 apache root 2638 Nov 9 15:09 index.php
-r----- 1 apache root 2362 Nov 9 15:09 index.php.sample
-r----- 1 apache root 6437 Nov 9 15:09 install.php
dr-x----- 15 apache root 4096 Nov 9 15:09 js
dr-x----- 16 apache root 4096 Nov 9 14:30 lib
-r----- 1 apache root 10421 Nov 9 14:30 LICENSE_AFL.txt
-r----- 1 apache root 32942 Nov 9 14:30 LICENSE_EE.html
-r----- 1 apache root 29934 Nov 9 14:30 LICENSE_EE.txt
-r----- 1 apache root 10679 Nov 9 14:30 LICENSE.html
-r----- 1 apache root 10410 Nov 9 14:30 LICENSE.txt
-r----- 1 apache root 1319 Nov 9 14:30 mage
dr-x----- 5 apache root 4096 Nov 22 11:19 magento-sample-data-1.14.1.0
drwx----- 18 apache root 4096 Nov 22 11:19 media
-r----- 1 apache root 886 Nov 9 14:30 php.ini.sample
dr-x----- 2 apache root 4096 Nov 9 15:08 pkginfo
dr-x----- 2 apache root 4096 Nov 22 11:19 privatesales
-r----- 1 apache root 346061 Nov 9 15:08 RELEASE_NOTES.txt
dr-x----- 4 apache root 4096 Nov 9 15:09 shell
dr-x----- 5 apache root 4096 Nov 9 14:30 skin
drwx----- 8 apache root 4096 Mar 17 18:28 var
[stack@m2-docs-staging-523722 magento]$
    
```

Recommended Permissions for a Live Store

4. To install a patch, you must change the permissions to allow the necessary files to be copied and overwritten, and to make it possible for the script to execute. Enter the following commands to change the permissions of the directories and files in the folder. Then, list the directory.

To save time, copy each command from the example, and press the mouse button to paste it into the command line.

Open Up Permissions

```

find . -type d -exec chmod 700 {} \;

find . -type f -exec chmod 600 {} \;

ls -l
    
```

If your Magento installation is on a shared virtual server, you might need to begin each command as a `sudo` super administrator:

```

sudo find . -type d -exec chmod 700 {} \;

sudo find . -type f -exec chmod 600 {} \;

sudo ls -l
    
```

The first column in the directory listing shows the updated permission settings. After the patch is installed, you will restore the recommended permissions to your store.

```

stack@m2-docs-staging-523722:/var/www/html/stores/ee/magento
total 548
-rw----- 1 apache root 2834 Nov  9 14:30 api.php
drwx----- 6 apache root 4096 Nov  9 15:08 app
-rw----- 1 apache root 2827 Nov  9 15:09 cron.php
-rw----- 1 apache root  717 Nov  9 14:30 cron.sh
drwx----- 7 apache root 4096 Nov  9 15:09 downloader
drwx----- 4 apache root 4096 Nov  9 15:09 errors
-rw----- 1 apache root 1150 Nov  9 14:30 favicon.ico
drwx----- 3 apache root 4096 Mar  7 01:50 frontend
-rw----- 1 apache root 5979 Nov  9 14:30 get.php
drwx----- 2 apache root 4096 Nov  9 15:09 includes
-rw----- 1 apache root 2638 Nov  9 15:09 index.php
-rw----- 1 apache root 2362 Nov  9 15:09 index.php.sample
-rw----- 1 apache root 6437 Nov  9 15:09 install.php
drwx----- 15 apache root 4096 Nov  9 15:09 js
drwx----- 16 apache root 4096 Nov  9 14:30 lib
-rw----- 1 apache root 10421 Nov  9 14:30 LICENSE_AFL.txt
-rw----- 1 apache root 32942 Nov  9 14:30 LICENSE_EE.html
-rw----- 1 apache root 29934 Nov  9 14:30 LICENSE_EE.txt
-rw----- 1 apache root 10679 Nov  9 14:30 LICENSE.html
-rw----- 1 apache root 10410 Nov  9 14:30 LICENSE.txt
-rw----- 1 apache root 1319 Nov  9 14:30 mage
drwx----- 5 apache root 4096 Nov 22 11:19 magento-sample-data-1.14.1.0
drwx----- 18 apache root 4096 Nov 22 11:19 media
-rw----- 1 apache root  886 Nov  9 14:30 php.ini.sample
drwx----- 2 apache root 4096 Nov  9 15:08 pkginfo
drwx----- 2 apache root 4096 Nov 22 11:19 privatesales
-rw----- 1 apache root 346061 Nov  9 15:08 RELEASE_NOTES.txt
drwx----- 4 apache root 4096 Nov  9 15:09 shell
drwx----- 5 apache root 4096 Nov  9 14:30 skin
drwx----- 8 apache root 4096 Mar 17 18:28 var
[stack@m2-docs-staging-523722 magento]#

```

Updated Permissions

Step 3: Create a Backup of the Magento Folder (Optional)

1. Use the Change Directory command, followed by two dots, to move up one level in the directory tree. Then, list the directory so you can see the `magento` folder.

```

cd ..
ls -l

```

2. Enter the following command to create a new folder called `backup`.

```

mkdir backup
ls -l

```

- a. If you get a “Permission denied” message, add `sudo` to the beginning of the Make Directory command. Then, list the directory to see the folder that you created.

```

sudo mkdir backup
sudo ls -l

```

- b. The `backup` folder must have the same owner and group as the `magento` folder. If not, use the Change Owner command to change the ownership of the `backup` folder and its contents.

In this command, the `backup` folder is assigned to the `apache` user and `root` group, and followed by a single dot. (Don't forget to use `sudo` if you need super administrator access.)

```
chown -hR apache:root backup .

ls -l
```

- c. Enter the “Who am I” command to make sure that you are logged in as the correct owner.

```
whoami
```

- d. If you are not logged in as the `apache` user, enter the following command. When prompted, enter the password for the `apache` user account.

```
su - apache
```

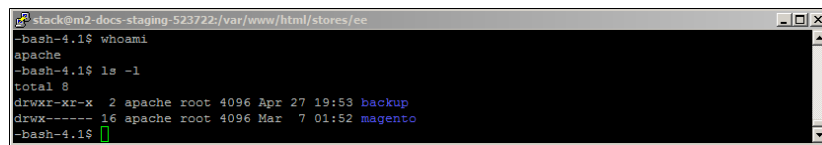
Notice that the prompt has changed. You are now in the `bash` folder of the `apache` user. While you're logged in as the `apache` user, you won't need to use `sudo` for super administrator access.

- 3. Use the Change Directory command to return to the location of your `magento` and `backup` folders. List the directory to see where you are. Then, check to see who you are.

```
cd /var/www/html/stores/ee

ls -l

whoami
```



Backup Folder with Correct Owner and Group

- 4. Copy the contents of the `magento` folder to the `backup` folder.

```
cp -r magento/* backup
```

5. Wait for the process to complete and for the system prompt to return. Then, take a look at the contents of the `backup` folder to make sure the files were copied.

```
cd backup
ls -l
cd ..
```

6. Because you'll be working with the `backup` folder for awhile, it's a good idea to restore the original permissions to your `magento` folder before continuing.

Enter the following commands to restore the recommended permissions to the `magento` folder. Then, list the directory to verify that the permissions are restored.

To save time, copy each command from the example, and press the mouse button to paste it into the command line.

Lock Down Permissions

```
ls -l
cd magento
find . -type d -exec chmod 500 {} \;
find . -type f -exec chmod 400 {} \;
find var/ -type f -exec chmod 600 {} \;
find media/ -type f -exec chmod 600 {} \;
find var/ -type d -exec chmod 700 {} \;
find media/ -type d -exec chmod 700 {} \;
ls -l
```



```
stack@m2-docs-staging-523722:/var/www/html/stores/ee
total 548
-r----- 1 apache root 2834 Nov 9 14:30 api.php
dr-x----- 6 apache root 4096 Nov 9 15:08 app
-r----- 1 apache root 2827 Nov 9 15:09 cron.php
-r----- 1 apache root 717 Nov 9 14:30 cron.sh
dr-x----- 7 apache root 4096 Nov 9 15:09 downloader
dr-x----- 4 apache root 4096 Nov 9 15:09 errors
-r----- 1 apache root 1150 Nov 9 14:30 favicon.ico
dr-x----- 3 apache root 4096 Mar 7 01:50 frontend
-r----- 1 apache root 5979 Nov 9 14:30 get.php
dr-x----- 2 apache root 4096 Nov 9 15:09 includes
-r----- 1 apache root 2638 Nov 9 15:09 index.php
-r----- 1 apache root 2362 Nov 9 15:09 index.php.sample
-r----- 1 apache root 6437 Nov 9 15:09 install.php
dr-x----- 15 apache root 4096 Nov 9 15:09 js
dr-x----- 16 apache root 4096 Nov 9 14:30 lib
-r----- 1 apache root 10421 Nov 9 14:30 LICENSE_AFL.txt
-r----- 1 apache root 32942 Nov 9 14:30 LICENSE_EE.html
-r----- 1 apache root 29934 Nov 9 14:30 LICENSE_EE.txt
-r----- 1 apache root 10679 Nov 9 14:30 LICENSE.html
-r----- 1 apache root 10410 Nov 9 14:30 LICENSE.txt
-r----- 1 apache root 1319 Nov 9 14:30 mage
dr-x----- 5 apache root 4096 Nov 22 11:19 magento-sample-data-1.14.1.0
drwx----- 18 apache root 4096 Nov 22 11:19 media
-r----- 1 apache root 886 Nov 9 14:30 php.ini.sample
dr-x----- 2 apache root 4096 Nov 9 15:08 pkginfo
dr-x----- 2 apache root 4096 Nov 22 11:19 privatesales
-r----- 1 apache root 346061 Nov 9 15:08 RELEASE_NOTES.txt
dr-x----- 4 apache root 4096 Nov 9 15:09 shell
dr-x----- 5 apache root 4096 Nov 9 14:30 skin
drwx----- 8 apache root 4096 Mar 17 18:28 var
-bash-4.1$
```

Locked Down Permissions

Step 4: Upload the Patch

1. To avoid permission problems when you upload the patch from your desktop, create a folder for the patch that has no restrictive permissions. From Putty, navigate back to the level of the `magento` and `backup` folders. Then, make a new folder called `patch`, and change the permissions of the folder to `777`.

```
cd ..
ls -l
mkdir patch
chmod 777 patch
ls -l
```

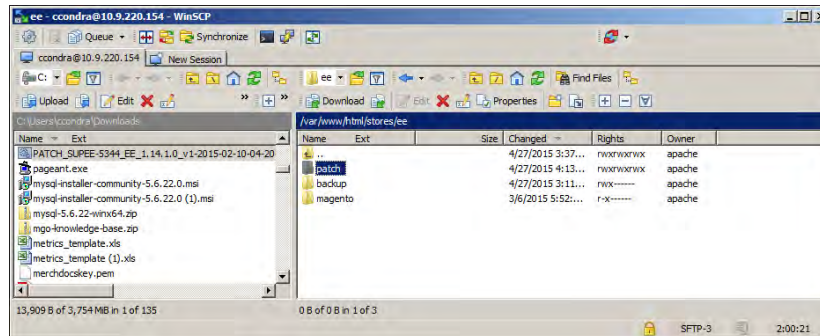
You now have three folders with very different permissions. The `patch` folder is wide open with read, write and execute permissions. For this limited purpose, it's OK.

```
stack@m2-docs-staging-523722:/var/www/html/stores/ee
-bash-4.1$ chmod 777 patch
-bash-4.1$ ls -l
total 12
drwx----- 16 apache root 4096 Apr 27 22:11 backup
dr-x----- 16 apache root 4096 Mar 7 01:52 magento
drwxrwxrwx 2 apache apache 4096 Apr 27 23:11 patch
-bash-4.1$
```

Folders with Different Permissions

2. Return to your desktop, and launch WinSCP, or a similar tool. Then, log in to the server.

3. In the right pane, navigate on the server to the location of the `patch` folder.
4. In the left pane, navigate to your **Downloads** folder, and find the patch file that you downloaded. Drag it over to the right pane, and drop it on the **patch** folder.



Upload the Patch File

Step 5: Install the Patch to the Backup (Optional)

1. Return to Putty in the directory where the `patch`, `backup`, and `magento` folders are located. Change directories into the `patch` folder, and list a directory to verify that the patch file is there. Then, copy the patch file up one level. (Just remember—two dots up, one dot down.)

```

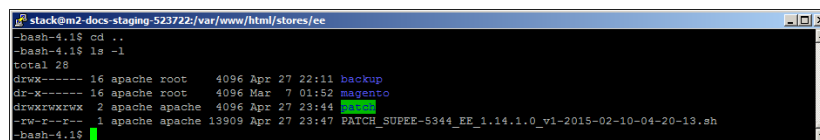
cd patch
ls -l

cp PATCH_SUPEE-5344_EE_1.14.1.0_v1-2015-02-10-04-20-13.sh ..

cd ..

ls -l
    
```

Now the patch file is at the same level as the `backup` and `magento` folders, where it can be easily accessed.



Patch File Uploaded and Ready to Go

2. Because the temporary `patch` folder has served its purpose, enter the following command to remove it from the server:

```
rm -rf patch
ls -l
```

3. Copy the patch file to the `backup` folder. Then, change directories to the `backup` folder, and list the directory. You should see the patch file in the directory.

```
cp PATCH_SUPEE-5344_EE_1.14.1.0_v1-2015-02-10-04-20-13.sh backup
cd backup
ls -l
```

4. To install the patch, use the appropriate syntax for the type of patch, and substitute the file name of the patch to be installed. There are two command formats, depending on the patch file name extension. To avoid typos, copy the patch file name from Putty, and paste it into Notepad. Then, add the required command syntax, paste it into the command line, and press Enter.

EXTENSION	COMMAND SYNTAX
.sh	sh [patch_filename.sh]
.patch	patch -p0 < [patch_filename.patch]

```
sh PATCH_SUPEE-5344_EE_1.14.1.0_v1-2015-02-10-04-20-13.sh
```

The following message appears if the installation succeeds:

```
Patch was applied/reverted successfully.
```

If the patch is already installed on your computer, an error message appears. You don't need to install it again.

So that's really all there is to it. It might seem rather anticlimactic after so much preparation, but it's important to understand the process before you apply the patch to your store.

5. The next step is to change directories to the `magento` folder, and reset the permissions so the patch can be copied and installed to your live store. Then, you will copy the patch file to the `magento` folder.

```
cd ..
ls -l
cd magento
find . -type d -exec chmod 700 {} \;
find . -type f -exec chmod 600 {} \;
cd ..
cp PATCH_SUPEE-5344_EE_1.14.1.0_v1-2015-02-10-04-20-13.sh magento
cd magento
ls -l
```

Step 6: Install the Patch to Your Store

1. From your `magento` folder, enter the appropriate command to install the patch, and press Enter.

EXTENSION	COMMAND SYNTAX
<code>.sh</code>	<code>sh [patch_filename.sh]</code>
<code>.patch</code>	<code>patch -p0 < [patch_filename.patch]</code>

To avoid typos, copy the name of the patch file from your Downloads folder and paste it into Notepad. Complete the required command syntax, and copy it to the clipboard. Then, press the mouse button to paste it into the command line.

```
sh PATCH_SUPEE-5344_EE_1.14.1.0_v1-2015-02-10-04-20-13.sh
```

2. When the process is complete, remove the patch file from the `magento` folder.

```
rm PATCH_SUPEE-5344_EE_1.14.1.0_v1-2015-02-10-04-20-13.sh
```

3. Enter the following commands to lock down permissions to the `magento` folder.

Lock Down Permissions

```
find . -type d -exec chmod 500 {} \;  
find . -type f -exec chmod 400 {} \;  
find var/ -type f -exec chmod 600 {} \;  
find media/ -type f -exec chmod 600 {} \;  
find var/ -type d -exec chmod 700 {} \;  
find media/ -type d -exec chmod 700 {} \;  
ls -l
```

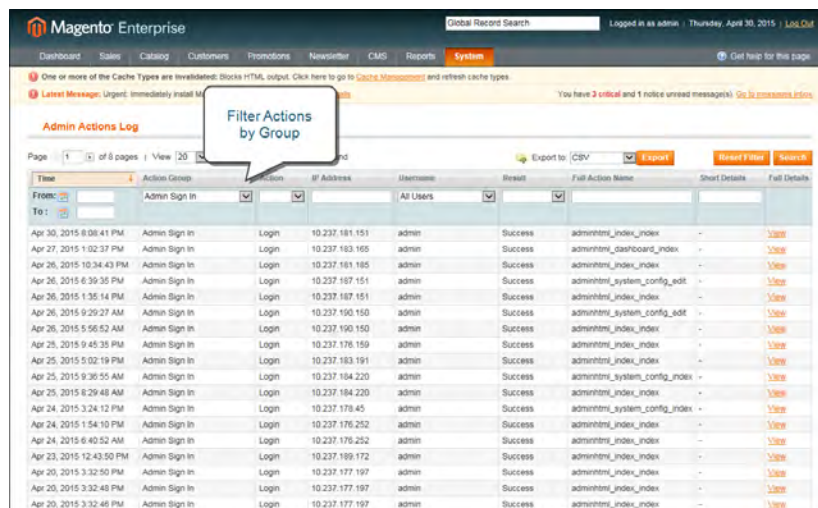
Step 7: A Little Housekeeping (Optional)

Enter the following commands to remove the `backup` folder and patch file from your server.

```
cd ..  
ls -l  
rm -rf backup  
rm PATCH_SUPEE-5344_EE_1.14.1.0_v1-2015-02-10-04-20-13.sh  
ls -l
```

Step 8: Look for Signs of Unauthorized Access

1. Log in to the Admin of your store. Then on the Admin menu, select **System > Users** and do the following:
 - a. Verify that there are no unauthorized user accounts in the list.
 - b. If you find an unknown user account in the list, click to open the account. Then, click the **Delete User** button.
2. On the Admin menu, select **System > Admin Actions Log**. Then, do the following:
 - a. Check the **IP Address** column for anything unfamiliar. To see the full URL, mouse over the address.
 - b. Check the **Username** column for any unfamiliar users.
 - c. To find specific types of activities, set the filter at the top of the **Action Group** column to the type of activity you want to examine.
 - d. To see the full entry for any activity, click the **View** link at the end of the row. Then, click the **Back** button to return to the list.
 - e. To export the log to a CSV file, click the **Export** button.

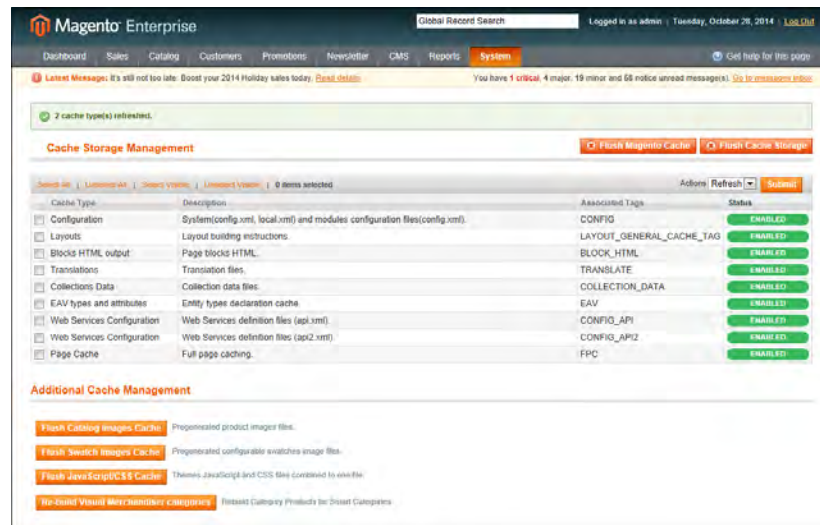


Admin Actions Log

To learn about more signs of unauthorized access, see: [Critical Security Advisory](#). If you suspect that your site is compromised, contact the security department of your hosting company and request an audit.

Step 9: Clear the Magento Cache

1. On the Admin menu, select **System > Cache Management**.
2. Click the **Flush Cache Storage** button.



Cache Storage Management

Step 10: Recompile the Store (Only if compiled)

If your store is **compiled**, you must recompile to incorporate the patch. If your store isn't compiled, you can skip this step.

1. On the Admin menu, select **System > Tools > Compilation**.
2. Click the **Run Compilation Process** button.

Step 11: Gracefully Restart the Server

The final step is to restart the server to flush any remaining caches, such as the APC and/or Zend OpCache. The following steps show how to gracefully restart the server without disrupting pages from being served, or causing loss of data.

1. If your server has a control panel such as [cPanel](#), look for the option to gracefully reboot the server. On cPanel, select **Home > System Reboot > Graceful Server Reboot**.
2. To gracefully restart from the command line, do the following:
 - a. The syntax to restart the server from the command line varies by operating system. To find the version that is running on your server, enter the following:

```
cat /etc/issue
```

- b. Use the appropriate command to gracefully restart the server. If necessary, begin the command with `sudo`.

CentOS / Fedora / Redhat

```
apachectl -k graceful
```

Debian / Ubuntu

```
apache2ctl graceful
```

3. To end the session, close both the **Putty** and **WinSCP** windows.

If you have more than one server, make sure to install the patch on all Magento servers.

That's it!
You're good to go.