Issues / Problems

How do I use custom ports and custom configurations for Apache and Nginx?

Custom Web VHosts and Ports

In EHCP Force Edition, if you change the web server mode to NON-SSL, SSL-ONLY, or SSL Mixed Mode in the panel, your custom port bindings (however, any custom ports specified in ports.conf for apache2 users will be saved if you are using the latest version EHCP Force Edition) and any custom virtualhost entries created outside of the panel will not be saved if they were created in the base configuration files used by EHCP. This is due to the fact that additional ports need to be used when listening in SSL mode. Vise versa, fewer ports need to be used when listening in Non-SSL mode. Configuration files are overwritten when the switch is made.

In order to avoid this problem, it is recommended you create custom host entries in unique configuration files and include them in the appropriate web server configuration files.

Apache2 Custom Ports and Vhosts

In Apache2, you can setup custom vhost entries by doing the following:

Create a file in the apache2 conf.d directory that will be used for custom vhost entries and a directory in /var/www/ for custom vhosts by using the following commands:

```
sudo touch /etc/apache2/conf.d/custom.conf
sudo mkdir /var/www/custom_vhosts
sudo chown -R vsftpd:www-data /var/www/custom_vhosts
```

Add any custom vhost entries to the /etc/apache2/conf.d/custom.conf file. Here's an example of what I have in my custom config:

```
NameVirtualHost 127.0.0.1:9999
        <VirtualHost 127.0.0.1:9999>
 ServerName  cacti
 ServerAlias cacti
 DocumentRoot /var/www/custom_vhosts/cacti

 LogFormat "%v:%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agen
t}i\"" vhost_combined
 CustomLog  /var/log/apache_common_access_log vhost_combined
        </VirtualHost>
```

After making any changes to the custom configuration file, make sure to restart the apache2

daemon for changes to apply. Also, make sure new directories and the files within the custom vhosts directory are owned by user vsftpd and group www-data.

```
sudo service apache2 restart
sudo chown -R vsftpd:www-data /var/www/custom_vhosts
```

If you use any custom ports, be sure to add them to /etc/apache2/ports.conf as a listen statement. For the above example to work, I need to add the listen port of 9999 to the ports.conf. It should look similar to this (depending on your settings):

```
Listen 80
Listen 443
Listen 9999
```

That's all you need to do for custom vhosts that are created and are not managed in the panel for security purposes or for any other reason you might have.

nginx Custom Ports and Vhosts

nginx works differently than Apache. To setup a custom vhost for nginx that is not managed in the EHCP panel, you can do the following:

Create a file in the nginx directory that will be used for custom vhost entries and a directory in /var/www/ for custom vhosts by running the following commands:

```
sudo touch /etc/nginx/conf.d/custom.conf
sudo mkdir /var/www/custom_vhosts
sudo chown -R vsftpd:www-data /var/www/custom_vhosts
```

Add any custom vhost entries to the /etc/nginx/conf.d/custom.conf file. Here's an example of what I have in my custom config:

```
server {
 listen 9999;
 server_name cacti;

 root /var/www/custom_vhosts/cacti/httpdocs;
 access_log  /var/www/custom_vhosts/cacti/logs/access_log;
 error_log   /var/www/custom_vhosts/cacti/logs/error_log;
 access_log /var/log/apache_common_access_log;
```

```
        index  index.html index.htm index.php;

        location / {
                        if (-f $document_root/error_page.html ) {
                                error_page 400 401 402 403 404 405 406
 407 408 409 410 411 412 413 414 415 416 417 495 496 497 500 501 502 5
03 504 505 506 507 /error_page.html;
                        }
        }


 location ~ \.php$ {
        root {homedir}/httpdocs;
        include fastcgi_params;
  try_files $uri = 404;
  fastcgi_pass   127.0.0.1:9000;
  fastcgi_index  index.php;
        fastcgi_param  SCRIPT_FILENAME  $document_root$fastcgi_script_
name;
  fastcgi_param  PHP_ADMIN_VALUE "open_basedir={homedir}:/tmp:/usr/sha
re/php:/usr/share/pear
upload_tmp_dir={homedir}/phptmpdir
session.save_path=0;660;{homedir}/phptmpdir
";
  fastcgi_read_timeout 300;
  limit_req zone=one burst=5;
 }

 # any files that should be avoided, may be put here:
 location ~ (apache_subdomain_template|apachetemplate|apachetemplate_i
pbased|apachetemplate_passivedomains|/\.conf$|/\.ht|access_log|error_l
og|access\.log|error\.log) {
   deny  all;
 }
}
```

After making any changes to the custom configuration file, make sure to restart the nginx daemon for changes to apply. Also, make sure new directories and the files within the custom vhosts directory are owned by user vsftpd and group www-data.

```
sudo service nginx restart
sudo chown -R vsftpd:www-data /var/www/custom_vhosts
```

Issues / Problems

Ports are specified in the vhost configuration in nginx, so you don't need to worry about custom ports being overwritten.

Final Notes

In both cases, cacti is the hostname that must resolve. Thus, when you send a request in your browser to cacti:9999, it will resolve based on your custom settings. You can add the resolve DNS information in your hosts file in /etc/hosts.

Unique solution ID: #1005
Author: Eric
Last update: 2016-01-28 23:50