APACHE 2

1 Virtualhost's e Sub-domínios

dica! Saca só 1.1 Introdução

`qui, aprenderemos a criar sub-domínios com virtualhost's` `1.2 Cria o diretório

```
sudo mkdir /var/www/dominio.com.br/public_html/ m O usuário **www-data** (e seu grupo homônimo) é o unico que tem permissão para executar o Apache server.
```

1.3 Insere o o seu usuário no grupo **www-data**.

```
sudo usermod -a -G www-data $USER
```

1.4 Altere o grupo proprietário da pasta **/var/www** e seu conteúdo para o grupo **www-data**:

```
sudo chown -R $USER:www-data /var/www
```

1.5 Concede permissões de controle total para o usuário (e o grupo) **www-data**. Contando que o seu usuário já tenha sido inserido no neste grupo (vide item 3), á partir de agora você terá direitos irrestritos (de escrita, leitura e execução) sobre os arquivos da pasta onde estarão os arquivos que deverão ser editados.

```
sudo chmod -R 775 /var/www/
```

1.6 Criar o arquivo index.html, dentro da pasta public_html correspondente.

```
sudo mkdir /var/www/dominio.com.br/public_html/index.html
```

1.7 O Apache vem com um arquivo de host virtual padrão chamado **000-default.conf** que usaremos como modelo. Vamos copiá-lo para criar um arquivo de host virtual para o nosso domínio.

```
sudo cp /etc/apache2/sites-available/000-default.conf /etc/apache2/sites-available/dominio.com.br.conf
```

1.8 Edite o arquivo criado, conforme o modelo abaixo:

```
sudo vim /etc/apache2/sites-available/dominio.com.br.conf
```

`

```
<VirtualHost *:80>
    ServerAdmin usuario@mail.com
    ServerName dominio.com.br
    ServerAlias www.dominio.com.br
    DocumentRoot /var/www/dominio.com.br/public_html/
    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

1.9 Habilitar os novos arquivos de host virtual.

```
sudo a2ensite dominio.com.br.conf
```

1.10 Desabilite o site default

```
sudo a2dissite 000-default.conf
```

1.11 Restart no apache

```
sudo systemctl restart apache2
```

1.12 Configurar o arquivo de hosts do seu computador para que reconheça o IP pelo endereço.

```
sudo vim /etc/hosts
```

```
127.0.0.1   localhost
127.0.1.1   guest-desktop
IP_do_server dominio.com.br
```

__2 Arquivo de configuração __

Contents

- Apache2 Configuration under Debian GNU/Linux
    - Files and Directories in '/etc/apache2' ` Tools``
- Using mod_cache_disk
- SSL `mEnabling SSL`
    - Creating self-signed certificates
    - SSL workaround for MSIE
- Suexec
- Documentation
- Upgrades
- Common Problems
- For Developers

Apache2 Configuration under Debian GNU/Linux

A instalação padrão do Apache2 do Debian tenta tornar a adição e remoção de módulos, hosts virtuais e diretivas de configuração extras tão flexíveis quanto possível, a fim de automatizar as mudanças e administrar o servidor o mais fácil possível. Esteja ciente de que esse layout é bem diferente da configuração padrão do Apache. Devido ao uso de variáveis de ambiente, o apache2 precisa ser iniciado parado com '/etc/init.d/apache2', apachectl ou apache2ctl. Chamar '/usr/bin/apache2' diretamente não funcionará com a configuração padrão. Para chamar apache2 com argumentos de linha de comando específicos, basta chamar apache2ctl com os mesmos argumentos. Arquivos e diretórios em '/etc/apache2':

apache2.conf

Este é o arquivo de configuração principal. Ele não inclui nenhuma configuração real que esperamos que seja adaptada em seu site, portanto, sempre que possível, não toque nele. Este arquivo é a pedra fundamental da configuração do Apache no Debian e deve ser atualizado após as atualizações para garantir que todas as peças de configuração sejam incluídas corretamente.

Se você deseja estender a configuração global, pode personalizar o servidor web Apache incluindo arquivos de configuração por meio do mecanismo conf-disponível. Para alterar as portas de escuta e o soquete configuração use o ports.conf (veja abaixo).

ports.conf

Diretivas de configuração para quais portas e endereços IP devem ser ouvir.

magic

Padrões para mod_mime_magic. Isso não é compatível com o formato usado pelas versões atuais dos pacotes file/libmagic.

Patterns for mod_mime_magic. This is not compatible with the format used by current versions of the file/libmagic packages.

envvars

```
Este contém variáveis de ambiente que podem ser usadas na
configuração. Algumas configurações, como usuário e arquivo pid,
precisam entrar aqui para que outros scripts possam usá-las. Ele
também pode ser usado para alterar algumas configurações padrão
usadas pelo apache2ctl, incluindo o valor ulimit para o número máximo
de arquivos abertos. A configuração padrão de LANG=C também está aqui
e pode ser alterada para um idioma diferente.

This contains environment variables that may be used in the
configuration. Some settings, like user and pid file, need to go in
here so that other scripts can use them. It can also be used to
change some default settings used by apache2ctl, including the ulimit
value for the maximum number of open files. The default LANG=C
setting is also here, and can be changed to a different language.
```

conf-available/

```
Files in this directory are included in the global server scope by
this line in apache2.conf:

# Include generic snippets of statements
    IncludeOptional conf-enabled/*.conf
```

```
This is a good place to add additional configuration
directives. All configuration snippets need a '.conf' suffix to be
included as actual configuration. The local administrator should
use file names starting with 'local-' to avoid name clashes with
files installed by packages.

Configuration snippets can be enabled and disabled by using the
a2enconf and a2disconf executables. This works similarly to the
approach used for modules and sites below.

Configuration snippets can of course also be included in individual
virtual hosts.
```

conf-enabled/

```
Like mods-enabled/ and sites-enabled/, a piece of configuration is
enabled by symlinking a file from conf-available/ into this
directory. The a2enconf helper is provided to assist this task.
```

mods-available/

```
This directory contains a series of .load and .conf files.
The .load files contain the Apache configuration directive
necessary to load the module in question.  The corresponding
.conf files contain configuration directives necessary to
utilize the module in question.
```

mods-enabled/

```
To actually enable a module for Apache2, it is necessary to
create a symlink in this directory to the .load (and .conf, if
it exists) files associated with the module in
mods-available/.  For example:

cgi.load -> /etc/apache2/mods-available/cgi.load

The a2enmod helper can be used to enable a module.
```

sites-available/

```
Like mods-available/, except that it contains configuration
directives for different virtual hosts that might be used with
apache2.  Note that the hostname doesn't have to correspond
```

```
exactly with the filename.  '000-default.conf' is the default
host which is provided by Debian.
```

sites-enabled/

```
Similar in functionality to mods-enabled/, sites-enabled
contains symlinks to sites in sites-available/ that the
administrator wishes to enable.

Apache uses the first VirtualHost that matches the IP/Port
as default for named virtual hosts. Therefore the 'default'
site should be called '000-default' to make sure it sorts before
other sites.

Example:
dedasys.conf -> /etc/apache2/sites-available/dedasys.conf

The a2ensite helper can be used to enable a site.
```

The Include directives ignore files with names that do not end with a .conf suffix. This behavior has changed from previous releases!

In some cases you may want to enable a specific piece of configuration (think of files shipped in conf-available/) for a particular virtual host only and not globally as is our default. In such cases you can disable the configuration at a global scope for example by doing

```
a2disconf some-configuration
```

Then it can be included in a particular virtual host within a file in sites-enabled/. You may want to add

```
Include conf-available/some-configuration.conf
```

in that site configuration. However, be careful, as this may not work for some configurations, depending on the context and implications of some directives.

Tools

a2enmod and a2dismod are available for enabling and disabling modules utilizing the above configuration system.

a2ensite and a2dissite do essentially the same thing as the above tools, but `or sites rather than modules. Finally a2enconf and a2disconf are the`corresponding tools for configuration snippets.`

a2query is a helper script providing runtime information about the running server instance. For example it can be used to query enabled modules, the `melected MPM, and other` `information. This tool is primarily meant for pakage` maintainers who need to interact with the Apache packages to activate their configurations upon package installation, but it can be used by users as well.

apxs2 -a/-A is modified to use a2enmod to activate newly installed modules.

Using mod_cache_disk

To ensure that the disk cache does not grow indefinitely, htcacheclean is started when mod_cache_disk is enabled. Both daemon and cron (daily) mode are supported. The configuration (run mode, cache size, etc.) is in /etc/default/apache2'.`` Normally, htcacheclean is automatically started and stopped by '/etc/init.d/apache2'. However, if you change the state of mod_cache_disk or the configuration of htcacheclean while apache2 is running, you may need to manually start/stop htcacheclean with "/etc/init.d/apache2 start-htcacheclan"` or "/etc/init.d/apache2 stop-htcacheclean".

Note that mod_cache_disk was named mod_disk_cache in versions 2.2 and earlier.

SSL

Enabling SSL

To enable SSL, type (as user root):

```
a2ensite default-ssl
```

```
a2enmod ssl
```

If you want to use self-signed certificates, you should install the ssl-cert package (see below). Otherwise, just adjust the SSLCertificateKeyFile and SSLCertificateFile directives in '/etc/apache2/sites-available/default-ssl.conf' to point to your SSL certificate. Then restart apache:

```
service apache2 restart
```

The SSL key file should only be readable by root; the certificate file may be globally readable. These files are read by the Apache parent process which runs as root, and it is therefore not necessary to make the files readable by the www-data user.

Creating self-signed certificates

If you install the ssl-cert package, a self-signed certificate will be automatically created using the hostname currently configured on your computer. You can recreate that certificate (e.g. after you have changed '/etc/hosts' or DNS to give the correct hostname) as user root with:

```
make-ssl-cert generate-default-snakeoil --force-overwrite
```

To create more certificates with different host names, you can use

```
make-ssl-cert /usr/share/ssl-cert/ssleay.cnf /path/to/cert-file.crt
```

This will ask you for the hostname and place both SSL key and certificate in the file '/path/to/cert-file.crt'. Use this file with the SSLCertificateFile directive in the Apache config (you don't need the SSLCertificateKeyFile in this case as it also contains the key). The file

'/path/to/cert-file.crt' should only be readable by root. A good directory to use for the additional certificates/keys is '/etc/ssl/private'.

SSL workaround for MSIE

The SSL workaround for MS Internet Explorer needs to be added to your SSL VirtualHost section (it was previously in ssl.conf but caused keepalive to be disabled even for non-SSL connections):

```
     `rowserMatch "MSIE [2-6]" \`
     `nokeepalive ssl-unclean-shutdown \
         downgrade-1.0 force-response-1.0
     BrowserMatch "MSIE [17-9]" ssl-unclean-shutdown
 `mhe default SSL virtual host in '/etc/apache2/sites-
 available/default-ssl.conf'`
 already contains this workaround.
```

Suexec

Debian ships two version of the suexec helper program required by mod_suexec. It is not installed by default, to avoid possible security issues. The package apache2-suexec-pristine contains the standard version that works only with document root /var/www, userdir suffix public_html, `and Apache run user www-data. The package pache2-suexec-custom contains a` customizable version that can be configured with a config file to use different settings (like /srv/www as document root). For more information see the suexec(8) man page in the apache2-suexec-custom package. `mSince apache2-suexec-custom has received less testing and migh be slightly` slower, apache2-suexec is the recommended version unless you need the features from apache2-suexec-custom.

Starting with Apache 2.4 both alternatives can be installed at the same time and the default suexec mechanism can be picked by using the update-alternatives(8) system.

Unicode File Name Normalization

Using Apache with the document root on a file system that does unicode normalization on the filenames can cause security issues. In Debian, this affects ZFS with the non-default option to enable filename normalization, `nd HFS+. It is strongly recommended not to use Apache with such file systems.`More information about this issue can be found by `searching the web for CVE-2013-0966.

`m_Documentation__`

The full Apache 2 documentation can be found on the web at

`ttp://httpd.apache.org/docs/2.4/` `or, if you have installed the apache2-doc package, in

/usr/share/doc/apache2-doc/manual/ `mr at`

http://localhost/manual/

There is also a wiki that contains useful information:

http://wiki.apache.org/httpd/

Some hints about securing Apache 2 on Debian are available at

http://wiki.debian.org/Apache/Hardening

Upgrades

Changes in the Apache packages that require manual configuration adjustments are announced in NEWS.Debian. Installing the apt-listchanges package is recommended. It will display the relevant NEWS.Debian sections before upgrades. `` **Multiple instances**

There is some support for running multiple instances of Apache2 on the same `machine. See '/usr/share/doc/apache2/README.multiple-instances' for moreinformation.`

`__Common Problems__`

1. Error message "Could not reliably determine the server's fully qualified domain name, using 127.0.0.1 for ServerName" during start `mThis can usually be ignored but it means that Apache htpd was unable to obtain` a fully-qualified hostname by doing a reverse lookup on your server's IP`address. You may want to add the fully-qualified hostname` o '/etc/hosts'.`An alternative is to specify "ServerName 127.0.0.1" in the global server context of the configuration, e.g. in '/etc/apache2/conf-enabled/local-servername.conf'.`m2) Error message "mod_rewrite: could not create ewrite_log_lock"`

This probably means that there are some stale SYSV semaphores around. This usually happens after apache2 has been killed with kill -9 (SIGKILL). You can clean up the semaphores with:

`ipcs -s | grep www-data | awk ' { print $2 } ' | xargs ipcrm sem`

3. Message "File does not exist: /etc/apache2/htdocs" in error log

In most cases this means that no matching VirtualHost definition could be found for an incoming request. Check that the target IP address/port and the name in the Host: header of the request actually match one of the virtual hosts.

4. Message "Couldn't create pollset in child; check user or system limits" in error log

On Linux kernels since 2.6.27.8, the value in

`/proc/sys/fs/epoll/max_user_instances`

needs to be larger than

```
for prefork/itk  MPM: 2 * MaxClients
for worker/event MPM: MaxClients + MaxClients/ThreadsPerChild
```

It can be set on boot by adding a line like

```
    fs.epoll.max_user_instances=1024
```

to '/etc/sysctl.conf'.

There are several other error messages related to creating a pollset that can appear for the same reason.

On the other hand, errors about adding to a pollset are related to the setting fs.epoll.max_user_watches. On most systems, max_user_watches should be high enough by default.

5. Message "Server should be SSL-aware but has no certificate configured" in error log

Since 2.2.12, Apache is stricter about certain misconfigurations concerning name based SSL virtual hosts. See NEWS.Debian.gz for more details.

6. Apache does not pass Authorization header to CGI scripts

This is intentional to avoid security holes. If you really want to change it, you can use mod_rewrite:

```
RewriteCond %{HTTP:Authorization} (.*)
RewriteRule . - [env=HTTP_AUTHORIZATION:%1]
```

7. mod_dav is behaving strangely

In general, if you use mod_dav_fs, you need to disable multiviews and script execution for that directory. For example:

```
<Directory /var/www/dav>
    Dav on
    Options -MultiViews -ExecCGI
    SetHandler none
    <IfModule mod_php5.c>
        php_admin_value engine Off
    </IfModule>
</Directory>
```

8. Message "apache2: bad user name ${APACHE_RUN_USER}" when starting apache2 directly

Use apache2ctl (it accepts all the same options as apache2).

9. A PUT with mod_dav_fs fails with "Unable to PUT new contents for /... `403, #0]" even if Apache has permission to write the file.`

Apache also needs write permission to the directory containing the file, in order to replace it atomically.

10. When starting/reloading Apache, there is the error message "ulimit: open files: cannot modify limit: Operation not permitted"

If you are running Apache in a vserver environment, the start script may not be allowed to set the maximum number of open files. You should adjust APACHE_ULIMIT_MAX_FILES in /etc/apache2/envvars to your setup. You can disable changing the limits by setting APACHE_ULIMIT_MAX_FILES=true .

For Developers

The Apache 2 web server package provides several helpers to assist packagers to interact with the web server for both, build and installation time. Please refer to the PACKAGING file in the apache2 package for `metailed information.``