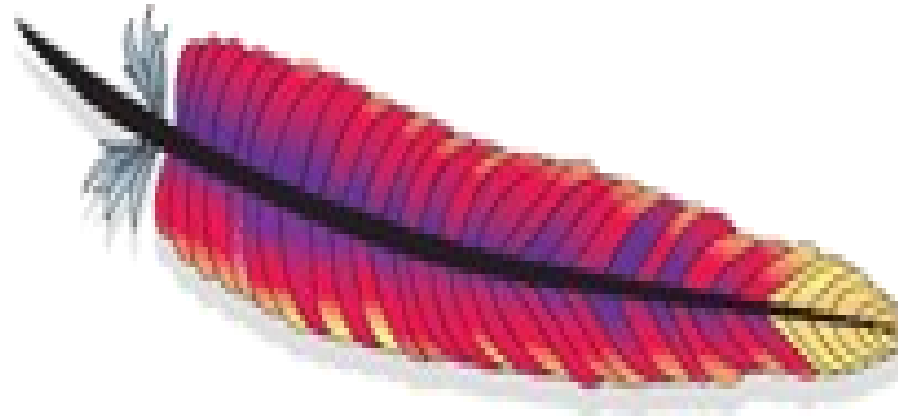


---

Universidade Tuiuti do Paraná  
Faculdade de Ciências Exatas



<http://www.apache.org>

# Conteúdo programático

---

- O que é o Apache?
- Principais características do servidor Apache
- Instalando o Apache
- Verificando o serviço
- Instalado e ativando módulos
- Principais diretivas do arquivo apache2.conf
- VirtualHosts / Permissões de páginas
- Apache com PHP
- Restringindo acesso a diretórios
- Geração dos certificados SSL

## O que é o Apache?

---

O Apache é um servidor Web extremamente configurável, robusto e de alta performance desenvolvido por uma equipe de voluntários (conhecida como Apache Group - <http://www.apache.org>) buscando criar um servidor web com muitas características e com código fonte disponível gratuitamente via Internet.

# Características do Apache?

---

- ✓ Possui suporte a scripts cgi usando linguagens como *Perl, PHP, Shell Script, ASP, etc.*
- ✓ Suporte a autorização de acesso podendo ser especificadas restrições de acesso separadamente para cada endereço/arquivo/diretório acessado no servidor.
- ✓ Autenticação requerendo um nome de usuário e senha válidos para acesso a alguma página/sub-diretório/arquivo (suportando criptografia via Crypto e MD5).
- ✓ Negociação de conteúdo, permitindo a exibição da página Web no idioma requisitado pelo cliente.
- ✓ Suporte a tipos mime.
- ✓ Personalização de logs.
- ✓ Mensagens de erro.
- ✓ Suporte a virtual hosting (é possível servir 2 ou mais páginas com endereços/portas diferentes através do mesmo processo ou usar mais de um processo para controlar mais de um endereço).
- ✓ entre muitos outros recursos....

# Instalando um servidor Apache?

# apt-get install apache2

```
ecelepar10149:/home/andresp# apt-get install apache2
Lendo lista de pacotes... Pronto
Construindo árvore de dependências... Pronto
Os pacotes extra a seguir serão instalados:
  apache2-mpm-worker apache2-utils apache2.2-common libapr1 libaprutil1 libpq4
Os NOVOS pacotes a seguir serão instalados:
  apache2 apache2-mpm-worker apache2-utils apache2.2-common libapr1
  libaprutil1 libpq4
0 pacotes atualizados, 7 pacotes novos instalados, 0 a serem removidos e 0 não
tualizados.
É preciso fazer o download de 2212kB de arquivos.
Depois de desempacotar, 5738kB adicionais de espaço em disco serão usados.
Quer continuar [S/n]? 
```

## Verificando serviço do Apache

---

```
# nmap -p 80 localhost
```

```
# ps -aux | grep apache
```

```
# service apache2 [ status | start | stop | reload ]
```

```
# apache2 -t                                # verifica sintaxe
```

```
# apache2 -V                                # verifica diretivas
```

No navegador:

<http://127.0.0.1>    <http://localhost>    [http://nome\\_da\\_sua\\_maquina](http://nome_da_sua_maquina)

*A seguinte mensagem será mostrada: It works!*

```
# ls /etc/apache2/mods-available/
```

```
# ls /etc/apache2/mods-enable/
```

```
# a2enmod <módulo>
```

Alguns módulos precisam ser instalados, exemplo:

```
# apt-cache search libapache2-mod
```

As diretivas dos módulos podem ser conhecidas pelo site:  
<http://httpd.apache.org/docs/2.2/mod>

# Diretivas do arquivo apache2.conf

---

# Diretório onde estão armazenados os arquivos de configuração do Apache

***ServerRoot "/etc/apache2"***

# Neste arquivo é armazenado o número do processo cada vez que o serviço é levantado

***PidFile /var/run/apache2.pid***

# Tempo máximo em segundos que o servidor esperará mantendo uma conexão aberta com o cliente.

# Se o limite for excedido, ele criará uma nova conexão com o mesmo.

***Timeout 300***

# Define se o processo de manter a conexão com o cliente está ativo ou não.

***KeepAlive On***



# Diretivas do arquivo apache2.conf

---

# Número máximo de conexões mantidas sem necessidade de renovação.

# Em servidores com grande número de acessos o aumento deste parâmetro

# pode resultar na melhora de performance do servidor.

***MaxKeepAliveRequests 100***

*# Máximo de segundos a espera de uma nova requisição.*

***KeepAliveTimeout 15***

# Número Mínimo de Servidores (processos do apache2 que irão atender as requisições)

***MinSpareServers 3***

# Número Máximo de Servidores

***MaxSpareServers 6***

# Diretivas do arquivo apache2.conf

---

# Número de servidores que serão iniciados

# Neste exemplo o apache2 poderá responder a 2 conexões simultâneas ao mesmo site.

***StartServers 2***

# Número máximo de conexões simultâneas por clientes ao sites

# Se for ultrapassada será exibido a msg 'http server busy' no browser.

***MaxClients 180***

# Nome do usuário e nome do grupo do processo do apache2.

# Procure certificar-se que as páginas hospedadas e os arquivos do apache2

# estão com dono e grupo conforme configurado abaixo:

***User www-data***

***Group www-data***

# Define o arquivo de log de erros, pode auxiliar em uma correção de algum problema

***ErrorLog /var/log/apache2/error.log***

# Diretivas do arquivo apache2.conf

---

# Garante que os módulos instalados nos diretórios abaixo estarão disponíveis

Include /etc/apache2/mods-enabled/\*.load

Include /etc/apache2/mods-enabled/\*.conf

# Inclui as configurações setadas no arquivo

Include /etc/apache2/httpd.conf

# Inclui o arquivo ports.conf que define qual será a porta padrão que o serviço irá escutar

***Include /etc/apache2/ports.conf***

# Define quais arquivos e extensões serão válidos

***DirectoryIndex index.php index.htm index.html***

# Arquivos que contêm as diretivas de acesso (Permitir/Bloquear)

AccessFileName .htaccess

# Diretivas do arquivo apache2.conf

---

# Se habilitada, uma página que por exemplo se chame: <http://localhost/fotos/>

# Poderá ser acessada como: <http://localhost/fotos>

# ou seja, sem o '/' no final da URL.

UseCanonicalName On

# Inclui as configurações dos Virtuais Hosts

Include /etc/apache2/sites-enabled/[^.#]\*

## VirtualHosts / Permissões de páginas

---

As permissões de acesso as páginas e virtual-hosts é dado pelas diretivas Order, deny e allow:

```
<Directory /var/www>
```

```
Options Indexes FollowSymLinks MultiViews
```

```
Order deny,allow
```

```
deny from all
```

```
# permite acesso somente as redes 10.15.xxx.xxx e rede 10.24.1.x
```

```
allow from 10.15.0.0/16 10.24.1.0/24
```

```
</Directory>
```

A diretiva Options pode conter os seguintes valores:

- **All:** Todas as opções são usadas exceto a MultiViews. É a padrão caso a opção Options não seja especificada.
- **ExecCGI:** Permite a execução de scripts CGI.
- **FollowSymLinks:** O servidor seguirá links simbólicos neste diretório (o caminho não é modificado). Esta opção é ignorada caso apareça dentro das diretivas <Location>, <LocationMatch> e <DirectoryMatch>.
- **Includes:** É permitido o uso de includes no lado do servidor.
- **IncludesNOEXEC:** É permitido o uso de includes do lado do servidor, mas o comando #exec e #include de um script CGI são desativados.

A diretiva Options pode conter os seguintes valores:

- **Indexes:** Se não existir um arquivo especificado pela diretiva <DirectoryIndex> no diretório especificado, o servidor formatará automaticamente a listagem ao invés de gerar uma resposta de acesso negado.
- **MultiViews:** Permite o uso da Negociação de conteúdo naquele diretório. A negociação de conteúdo permite o envio de um documento no idioma requisitado pelo navegador do cliente.
- **SymLinksIfOwnerMatch:** O servidor somente seguirá links simbólicos se o arquivo ou diretório alvo tiver como dono o mesmo user ID do link. Esta opção é ignorada caso apareça dentro das diretivas

```
# apt-get install libapache2-mod-php5
```

```
# service apache2 restart
```

Para testar ao acesso ao PHP entre no diretório '/var/www/

```
# cd /var/www/
```

- Crie o arquivo 'testarphp5.php' e acrescente o seguinte conteúdo:

```
# vim testarphp5.php
```

```
<?php
```

```
    phpinfo();
```

```
?>
```

Teste no seu navegador: **`http://<servidor>/testarphp5.php`**



# Restringindo acesso aos diretórios

*Crie os dois arquivos abaixo:*

*# cd /var/www/<diretório>*

*/var/www/<diretório># htpasswd -c .passwd admin*

*Senha:*

*Repita a senha:*

*/var/www/<diretório># vim .htaccess*

*AuthUserFile /var/www/<diretório>/.passwd*

*AuthGroupFile /dev/null*

*AuthName "Digite sua senha"*

*AuthType Basic*

*require valid-user*

# Restringindo acesso aos diretórios

**Adicione as linhas abaixo no arquivo */etc/apache2/sites-available/default* :**

```
DocumentRoot /var/www
<Directory />
Options FollowSymLinks
AllowOverride AuthConfig
</Directory>
<Directory /var/www/>
Options Indexes FollowSymLinks MultiViews
AllowOverride None
Order allow,deny
allow from all
</Directory>

<Directory "/var/www/<diretório>">
  AllowOverride All
  Options IncludesNoExec Indexes
  AddOutputFilter Includes html
  AddHandler type-map var
  Order allow,deny
  Allow from all
  LanguagePriority pt-br
  ForceLanguagePriority Prefer Fallback
</Directory>
```

# Restringindo acesso aos diretórios

**Reinicie o servidor apache:**

**# service apache2 restart**

**Acesse:**

***http://<ip servidor>/<diretório>***

## Geração dos certificados SSL

---

Para estabelecer uma ligação segura e de confiança é necessário gerar certificados que atestem a identidade do servidor. Esses certificados são geralmente emitidos por entidades certificadoras (Certificate Authority) independentes e de confiança reconhecida. No entanto, é possível criar um certificado "auto-assinado".

# Geração dos certificados SSL

```
# apt-get install openssl ca-certificates
```

```
# mkdir certs ; cd certs
```

## **Gere a chave privada (*Private Key*):**

```
# openssl genrsa -des3 -out server.key 1024
```

```
ou # openssl rsa -in server.key -out server.key.insecure
```

```
# chmod 600 server.key.insecure
```

## **Gerar o pedido de certificação (Certificate Signing Request):**

```
# openssl req -new -key server.key -out server.csr
```

<http://certs.ipsca.com/>

## Geração dos certificados SSL

O pedido de certificado deveria ser enviado para a entidade certificadora, que devolveria o certificado assinado. Neste caso, será usado para criar um certificado auto assinado (*Self-Signed Certificate*), válido por 365 dias:

```
# openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt
```

### Arquivos Gerados:

<i><b>Arquivo</b></i>	<i><b>Descrição</b></i>
<i>server.key</i>	<i>A chave privada</i>
<i>server.key.insecure</i>	<i>A chave privada sem senha</i>
<i>server.csr</i>	<i>O pedido de assinatura do certificação</i>
<i>server.crt</i>	<i>O certificado auto-assinado.</i>

# Geração dos certificados SSL

```
# cp server.key server.key.insecure /etc/ssl/private/
```

```
# cp server.crt /etc/ssl/certs/
```

```
# cd /etc/apache2/sites-available/
```

```
# cp default ssl
```

```
# a2enmod ssl
```

```
# vim /etc/apache2/ports.conf:
```

```
Listen 80
```

```
Listen 443
```

```
# a2ensite ssl
```

## Atribuição-Compartilhamento pela mesma licença 2.5

(<http://creativecommons.org/licenses/by-sa/2.5/deed.pt>)



### Você pode:

- copiar, distribuir, exibir e executar a obra
- criar obras derivadas
- fazer uso comercial da obra



### Sob as seguintes condições:

**Atribuição.** Você deve dar crédito ao autor original, da forma especificada pelo autor ou licenciante.

**Compartilhamento pela mesma Licença.** Se você alterar, transformar, ou criar outra obra com base nesta, você somente poderá distribuir a obra resultante sob uma licença idêntica a esta.

- Para cada novo uso ou distribuição, você deve deixar claro para outros os termos da licença desta obra.
- Qualquer uma destas condições podem ser renunciadas, desde que Você obtenha permissão do autor.

Qualquer direito de uso legítimo (ou "fair use") concedido por lei, ou qualquer outro direito protegido pela legislação local, não são em hipótese alguma afetados pelo disposto acima.

*Este é um sumário para leigos da Licença Jurídica*

(na íntegra: <http://creativecommons.org/licenses/by-sa/2.5/br/legalcode>).

*Termo de exoneração de responsabilidade:*

<http://creativecommons.org/licenses/disclaimer-popup?lang=pt>