

# Resumo dos Problemas e Troubleshooting do Apache

Você enfrentou um persistente erro de "**Acesso Proibido**" (**HTTP Error 403**) ao tentar acessar suas páginas web hospedadas no Apache localmente. A solução envolveu uma série de verificações e ajustes em várias camadas da configuração do servidor.

## Problemas Iniciais e Soluções Aplicadas

### 1. Erro Inicial 403 (localhost):

- **Problema:** Acesso negado ao DocumentRoot padrão (/srv/www/htdocs).
- **Troubleshooting & Solução:**
  - Confirmado que o **firewall não era o problema**, pois as portas 80 e 443 estavam abertas.
  - Verificado o status do Apache (systemctl status apache2) e os logs de erro (/var/log/apache2/error\_log).
  - O problema principal era de **permissões no sistema de arquivos** ou na **configuração de diretório do Apache**.
  - **Ajustes:**
    - Permissões para /srv/www/htdocs/ e index.html foram verificadas e ajustadas para 755 e 644, respectivamente.
    - O bloco <Directory "/srv/www/htdocs/"> no httpd.conf foi garantido que tinha Require all granted (ou equivalente em versões antigas).
  - **Resultado:** O acesso ao DocumentRoot padrão foi corrigido.

### 2. Erro 403 com mod\_userdir (~jose):

- **Problema:** Ao tentar usar o UserDir public\_html do seu usuário (/home/jose/public\_html), o erro 403 persistia.
- **Troubleshooting & Solução:**
  - **Permissões do diretório ~jose (/home/jose):** Identificado que o diretório home do usuário precisava de permissão de execução para "outros" para que o Apache pudesse atravessá-lo. Foi ajustado para 701 (drwx--x--x).
  - **Permissões do index.html no public\_html:** O arquivo index.html dentro de public\_html estava com permissões muito restritivas (-rw-----.), impedindo o Apache de lê-lo.
  - **Ajuste:** A permissão do index.html foi corrigida para 644 (-rw-r--r--). O comportamento "estranho" de o arquivo não herdar as permissões do diretório foi explicado pelo umask do usuário.
  - **SELinux:** Embora o SELinux pudesse ser um culpado comum, você confirmou que os contextos (httpd\_user\_content\_t) estavam corretos para public\_html e

index.html, descartando-o como causa do 403 neste ponto.

- **Configuração ausente do mod\_userdir:** O principal problema encontrado foi a **falta do bloco <Directory /home/\*/public\_html>** no arquivo /etc/apache2/conf.d/userdir.conf. Este bloco é essencial para permitir que o Apache sirva conteúdo dos diretórios public\_html dos usuários.
  - **Ajuste:** O bloco <Directory /home/\*/public\_html> foi adicionado com Require all granted e AllowOverride All (para flexibilidade com .htaccess).
  - **Resultado:** O acesso via http://192.168.0.49/~jose/ foi bem-sucedido.
3. **Configuração de Virtual Hosts para Múltiplos Sites Locais (olamundo.local):**
- **Problema:** Desejo de configurar localhost e outros domínios locais para apontar para diretórios específicos (flexibilidade para múltiplos sites).
  - **Troubleshooting & Solução:**
    - **Abordagem Inicial (Opção 1):** Mudar o DocumentRoot global no default-server.conf para /home/jose/public\_html. Esta foi uma solução simples, mas menos flexível para múltiplos sites.
    - **Abordagem para Flexibilidade (Virtual Hosts):** A melhor prática é criar Virtual Hosts específicos.
      - **Reversão:** O default-server.conf foi revertido para seu DocumentRoot original (/srv/www/htdocs).
      - **Criação de Virtual Host:** Um novo arquivo .conf (/etc/apache2/vhosts.d/olamundo.local.conf) foi criado para olamundo.local.
      - **Configuração do Virtual Host:** Incluiu ServerName olamundo.local, DocumentRoot "/srv/www/olamundo.local/public\_html", e um bloco <Directory> apropriado com Require all granted.
      - **Resolução de Nomes Locais:** O arquivo /etc/hosts foi atualizado para mapear olamundo.local para 127.0.0.1 (o IP local do seu PC).
    - **Resultado:** O site http://olamundo.local/ funcionou com sucesso, demonstrando a capacidade de hospedar múltiplos sites localmente.

<br>

---

## Guia de Troubleshooting Abrangente (Expandido)

Sempre que encontrar problemas com o Apache, siga esta lista de verificação detalhada:

### 1. Verificação Básica do Serviço Apache

- **Status do Serviço:** sudo systemctl status apache2
  - active (running): OK.
  - inactive (dead): sudo systemctl start apache2.

- failed: Procure por erros nos logs.
- **Reiniciar Apache:** sudo systemctl restart apache2 (após qualquer alteração na configuração).
- **Verificar Sintaxe:** sudo apachectl configtest (deve retornar Syntax OK).

## 2. Análise de Logs do Apache

- **Logs de Erro Gerais:** sudo tail -f /var/log/apache2/error\_log
  - Procure por error, warn, crit que coincidam com o momento do problema.
- **Logs de Acesso Gerais:** sudo tail -f /var/log/apache2/access\_log
  - Confirme se a requisição chegou ao Apache e qual o código de status HTTP (200 OK, 403 Forbidden, 404 Not Found, 500 Internal Server Error).
- **Logs Específicos do Virtual Host:**
  - sudo tail -f /var/log/apache2/SEU\_SITE\_error.log
  - sudo tail -f /var/log/apache2/SEU\_SITE\_access.log
  - Esses logs são mais detalhados para o site em questão.

## 3. Verificação de Permissões do Sistema de Arquivos

- **DocumentRoot do Site:** ls -ld /caminho/do/seu/site/public\_html
  - Deve ser 755 (drwxr-xr-x).
- **Arquivos do Site (HTML, CSS, JS):** ls -l /caminho/do/seu/site/public\_html/index.html (ou outros arquivos)
  - Deve ser 644 (-rw-r--r--).
- **Diretórios Pai:** ls -ld /caminho/do/seu/site/ e ls -ld /caminho/do/seu/
  - Devem ter permissão de execução (x) para "outros" (geralmente 755 ou 701) para permitir a travessia do Apache.
- **Correção de Permissões:** Use sudo chmod e sudo chown para ajustar se necessário.
  - Ex: sudo chmod 755 /srv/www/olamundo.local/public\_html
  - Ex: sudo chmod 644 /srv/www/olamundo.local/public\_html/index.html
  - Ex: sudo chown -R seu\_usuario:www /srv/www/olamundo.local/ (ajuste o proprietário/grupo conforme necessário para o Apache ler).

## 4. Revisão da Configuração do Apache

- **DocumentRoot e <Directory>:** No arquivo .conf do seu Virtual Host (/etc/apache2/vhosts.d/SEU\_SITE.conf), confirme que o DocumentRoot e os caminhos nos blocos <Directory> estão exatos.
- **ServerName:** Verifique a correspondência entre o ServerName no .conf e o que você digita no navegador/hosts.
- **Require all granted:** Essencial dentro do bloco <Directory> para permitir acesso HTTP.
- **AllowOverride:** Se estiver usando .htaccess para regras de reescrita, autenticação etc., certifique-se de que AllowOverride está definido como All ou FileInfo no bloco

<Directory>.

## 5. Verificação da Resolução de Nomes

- **Arquivo hosts (no PC cliente):** cat /etc/hosts (ou Windows: C:\Windows\System32\drivers\etc\hosts)
  - Confirme que a linha 127.0.0.1 SEU\_SITE.local (ou o IP do seu servidor) está presente e correta.
- **Servidor DNS (se configurado):** Se você instalou um servidor DNS (dnsmasq ou Bind9), verifique se ele está rodando e configurado para resolver seu ServerName para o IP do servidor Apache.

## 6. Verificação do Firewall (firewalld)

- **Status e Regras:** sudo firewall-cmd --list-all
  - As portas 80/tcp (HTTP) e 443/tcp (HTTPS) devem estar abertas.
- **Adicionar Regras (se ausentes):**
  - sudo firewall-cmd --permanent --add-service=http
  - sudo firewall-cmd --permanent --add-service=https (se usar HTTPS)
  - sudo firewall-cmd --reload

## 7. Verificação do SELinux

- **Status:** sestatus (se estiver enforcing).
- **Contexto de Segurança:** ls -Z /caminho/do/seu/site/public\_html e ls -Z /caminho/do/seu/site/public\_html/index.html
  - O contexto deve ser httpd\_user\_content\_t ou similar para o Apache acessar.
- **Restaurar Contexto:** sudo restorecon -Rv /caminho/do/seu/site/ (se o contexto estiver incorreto).

<br>

---

# Guia de Customização de Sites (Expandido)

## 1. Criando um Novo Site (Virtual Host)

### 1. Criar Diretório do Site:

```
Bash
sudo mkdir -p /srv/www/novo_site.local/public_html
```

## 2. Criar Conteúdo de Teste:

```
Bash
echo "<h1>Bem-vindo ao Novo Site!</h1>" | sudo tee
/srv/www/novo_site.local/public_html/index.html
```

## 3. Definir Permissões e Proprietário:

```
Bash
sudo chmod -R 755 /srv/www/novo_site.local/
sudo chown -R wwwrun:www /srv/www/novo_site.local/ # ou jose:users se preferir
```

## 4. Criar Arquivo de Configuração do Virtual Host:

```
Bash
sudo nano /etc/apache2/vhosts.d/novo_site.local.conf
```

Adicione o conteúdo (similar ao olamundo.local.conf):

```
Apache
<VirtualHost *:80>
    ServerName novo_site.local
    DocumentRoot "/srv/www/novo_site.local/public_html"

    <Directory "/srv/www/novo_site.local/public_html">
        Options Indexes FollowSymLinks
        AllowOverride All
        Require all granted
    </Directory>

    ErrorLog /var/log/apache2/novo_site.local_error.log
    CustomLog /var/log/apache2/novo_site.local_access.log combined
</VirtualHost>
```

## 5. Adicionar ao Arquivo hosts (no seu PC):

```
Bash
sudo nano /etc/hosts
# Adicione:
127.0.0.1 novo_site.local
```

## 6. Verificar Sintaxe e Reiniciar Apache:

```
Bash
sudo apachectl configtest
sudo systemctl restart apache2
```

## 7. Acessar no Navegador: [http://novo\\_site.local/](http://novo_site.local/)

## 2. Uso de .htaccess (Configurações por Diretório)

Crie um arquivo chamado .htaccess no DocumentRoot do seu site (ou em subdiretórios) e adicione as diretivas. Lembre-se que o AllowOverride All no seu Virtual Host é necessário para que .htaccess funcione.

- **Redirecionamentos:**

```
Apache
# Redireciona pagina-antiga.html para pagina-nova.html
Redirect 301 /pagina-antiga.html /pagina-nova.html
```

- **URLs Amigáveis (Pretty URLs - com mod\_rewrite):**

```
Apache
RewriteEngine On
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule ^(.*)$ index.php [L,QSA] # Encaminha tudo para index.php (comum em frameworks PHP)
```

- **Proteção por Senha (Basic Authentication):**

1. Crie um arquivo de senha (fora do DocumentRoot por segurança):  
sudo htpasswd -c /etc/apache2/.htpasswd meu\_usuario (irá pedir uma senha)
2. No .htaccess:

```
Apache
AuthType Basic
AuthName "Acesso Restrito"
AuthUserFile /etc/apache2/.htpasswd
Require valid-user
```

## 3. Habilitando Módulos do Apache

Para adicionar funcionalidades como PHP, SSL, compressão, etc., você precisa carregar os módulos correspondentes.

- **Identificar e Adicionar Módulos:** No openSUSE, edite o arquivo /etc/sysconfig/apache2 e adicione o nome do módulo à lista APACHE\_MODULES.

- Ex: Para PHP 8: Adicione php8 à lista.
- Ex: Para SSL: Adicione ssl à lista.
- **Reiniciar Apache:** Sempre reinicie após modificar APACHE\_MODULES.

## 4. Configuração de HTTPS (SSL/TLS)

Para desenvolver com HTTPS localmente, você precisa de um certificado SSL (autoassinado para teste).

### 1. Gerar Certificado Autoassinado (se não tiver):

Bash

```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout  
/etc/ssl/apache2/server.key -out /etc/ssl/apache2/server.crt
```

### 2. Habilitar Módulo SSL: Adicione ssl à lista APACHE\_MODULES em /etc/sysconfig/apache2.

### 3. Adicionar Virtual Host para HTTPS (Porta 443): No arquivo .conf do seu site (olamundo.local.conf), adicione um novo bloco <VirtualHost \*:443>:

Apache

```
<VirtualHost *:443>  
    ServerName olamundo.local  
    DocumentRoot "/srv/www/olamundo.local/public_html"
```

```
    SSLEngine on
```

```
    SSLCertificateFile /etc/ssl/apache2/server.crt
```

```
    SSLCertificateKeyFile /etc/ssl/apache2/server.key
```

```
<Directory "/srv/www/olamundo.local/public_html">
```

```
    Options Indexes FollowSymLinks
```

```
    AllowOverride All
```

```
    Require all granted
```

```
</Directory>
```

```
ErrorLog /var/log/apache2/olamundo.local_error_ssl.log
```

```
CustomLog /var/log/apache2/olamundo.local_access_ssl.log combined
```

```
</VirtualHost>
```

### 4. Reiniciar Apache.

### 5. Acessar com HTTPS: https://olamundo.local/ (seu navegador provavelmente avisará sobre um certificado não confiável, o que é normal para autoassinados).

Com este resumo e guia detalhado, você tem todas as ferramentas para solucionar problemas e customizar seus sites Apache no ambiente local. Continue experimentando!