

## **Atividade Proposta da Disciplina:**

### **Laboratório de Rede de Computadores**

Tratamento dos dados gerados pelo Servidor de Web a partir do arquivo de logs.

Configuração do

Servidor Web para armazenar em um arquivo texto os logs de acesso, conforme roteiro abaixo:

- Pesquisar sobre o sistema de logs no servidor web no apache;
- Instalar o Servidor Web no Linux “Apache”;
- Criar um virtual host e hospedar página em HTML;
- Criar e configurar os logs de acesso das páginas hospedadas na virtual host do apache.

*Professor responsável: Dinailton José da Silva.*

## **INSTALANDO SERVIDOR APACHE NO LINUX CENTOS**

### **1º PASSO:**

Logar como root (administrador do sistema), abrir o terminal ou konsole e digitar o seguinte comando:

```
# yum install httpd
```

Após o comando ser digitado será mostrada a quantidade de pacotes a ser baixados e instalados e uma solicitação para que os mesmos sejam baixados e instalados. Digite **S** para aceitar.

### **2º PASSO:**

Após a instalação será necessário iniciar o serviço do nosso servidor apache, faça isso digitando o comando abaixo:

```
# service httpd start
```

Se tudo correr bem será apresentado uma mensagem de ok e o servidor apache foi instalado com sucesso e está pronto para ser configurado.

## CONFIGURANDO SERVIDOR APACHE NO LINUX CENTOS

Antes de apresentar os passos para configurar o servidor Apache irei passar algumas informações importantes.

Toda configuração será feita no arquivo **httpd.conf**, por isso é muito importante fazer um backup do mesmo para que caso erramos alguma configuração propositalmente ou acidental possamos reverter só copiando o backup no lugar do original. Vamos aos passos.

### 1º PASSO:

Localizar o arquivo **httpd.conf**, esse arquivo como padrão fica em **/etc/httpd/conf** e deve permanecer lá afim de manter a integridade do servidor apache. Comando para abrir o arquivo pelo terminal ou konsole

```
# gedit /etc/httpd/conf/httpd.conf
```

### 2º PASSO:

Como padrão a porta do servidor Apache é a **80** você pode alterá-la modificando a seguinte linha no arquivo **httpd.conf**.

```
Listen 80
```

Após escolher a porta da sua preferência salve o arquivo.

OBS: Se configurou o gedit para mostrar números de linha está linha vai ser a 136.

Lembre-se de liberar a porta que configurou no firewall, para que assim outros hosts que estão na mesma rede do seu servidor possam acessá-lo.

**Atenção!**

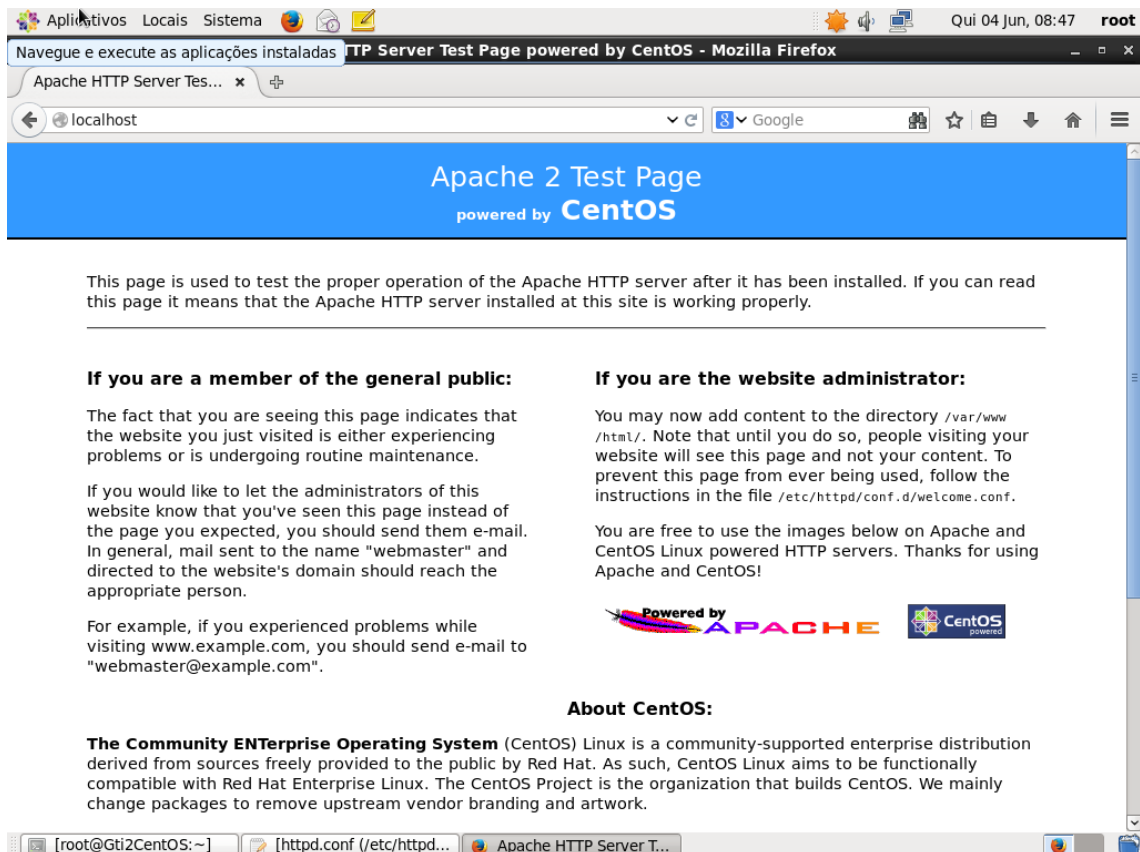
**Após a alteração reiniciar o serviço do servidor apache**

```
# service httpd restart
```

### 3º PASSO

Vamos testar agora!

Abra um navegador e na barra de endereço digite “**http://localhost**” ou **127.0.0.1** a pagina padrão será apresentada a mesma fica localizada em **/var/www/html** é onde o servidor Apache busca o arquivo html que iremos criar a seguir.



## CONFIGURANDO VIRTUAL HOST

Virtual host serve para hospedar seus sites ou de terceiros, com domínios ou subdomínios diferentes num único servidor web e com um único endereço IP.

### 1º PASSO

Criar a pasta e nosso arquivo HTML do site que queremos hospedar

Para isso vamos utilizar o comando **mkdir** para criar as pasta segue o comando abaixo.

```
# mkdir /var/www/html/servidorweb
```

Obs: o nome da pasta fica a sua escolha.

Dentro da pasta que criamos iremos criar outra pasta para aguardar os logs

```
# mkdir /var/www/html/servidorweb/logs
```

## 2º PASSO

Criar arquivo HTML (a pagina principal do site) comando para editar e criar arquivo HTML já dentro da pasta onde ficara o site.

```
# gedit /var/www/html/servidorweb/index.html
```

Obs: crie o arquivo HTML de teste com as seguintes tags HTML

```
<html>
<head>
    <metacharset= "utf-8">
    <title>servidor web</title>
</head>
<body>
    <h1 align = "center" ><font color="red"> Parabéns! </font></h1>
    <p align = "center" > Aqui está a pagina do nosso site "</p>
</body>
</html>
```

## 3º PASSO

Agora iremos alterar algumas linhas no arquivo **httpd.conf** a primeira será a seguinte.  
Que no arquivo esta no final do dele o número **990** se estiver habitada a função numerar linha no gedit.

```
#NameVirtualHost *:80
```

A alteração que será feita é descomentar essa linha que ficara assim:

```
NameVirtualHost *:80
```

Obs: sempre que for referir a descomentar uma linha é só eliminar a **#** do começo da linha.

#### 4º PASSO

Iremos modificar as seguintes linhas no fim do arquivo **httpd.conf** que vem como padrão assim:

```
#<VirtualHost *:127.0.0.1>
#   ServerAdmin webmater@dummy-host.example.com
#   DocumentRoot /www/docs/dummy-host.example.com
#   ServerName dummy-host.example.com
#   ErrorLog logs/dummy-host.example.com-error_log
#   CustomLog logs/dummy-host.example.com-access_log common
</VirtualHost>
```

E ficara de acordo com a pasta criamos e o domínio que iremos criar.

```
#<VirtualHost *:127.0.0.1>
#   ServerAdmin root@servidorweb.com.br
#   DocumentRoot /var/www/html/servidorweb
#   ServerName meusiteservidorweb.com.br
#   ErrorLog /var/www/html/servidorweb/log/error_log combined
#   CustomLog /var/www/html/servidorweb/logs/access_log combined
#</VirtualHost>
```

Para que possamos digitar somente o domínio do site e o mesmo mostra a página index.html teremos que alterar mais uma linha do **httpd.conf**

```
287 #
288 # DocumentRoot: The directory out of which you will serve your
289 # documents. By default, all requests are taken from this directory, but
290 # symbolic links and aliases may be used to point to other locations.
291 #
292 DocumentRoot "/var/www/html"
293
294 #
```

Para:

**DocumentRoot "/var/www/html/servidorweb"**

**Atenção!**

Não se esquecer de abrir e fechar aspas.

**Atenção!**

**Após a alteração reiniciar o serviço do  
servidor apache**

**# service httpd start**

Teste aparecerá assim:

## 5º PASSO

### CONFIGURANDO LOGS

Criaremos dois arquivos de log para registramos falhas (error\_log) e acessos (access\_log) para termos um controle e uma estimativa dos frequentadores do site, não precisar digitar nada só criar salvar e fechar as informações será adicionada pelo Apache.

Se reparou eu já adicionei o caminho para os arquivos de log no arquivo [httpd.conf](#) são as duas últimas linhas ErrorLog e CustomLog.

Comandos para criar os arquivos:

[error\\_log](#)

```
# gedit /var/www/html/servidor/logs/error_log
```

[access\\_log](#)

```
# gedit /var/www/html/servidor/logs/access_log
```

## 6º PASSO

Agora vamos configurar e personalizar as informações que queremos no log.

**Os seguintes formatos são válidos:**

- **%b** - Bytes enviados, excluindo cabeçalhos HTTP.
- **%f** - Nome do arquivo.
- **%{FOOBAR}e** - O conteúdo da variável de ambiente FOOBAR.
- **%h** - Máquina cliente.
- **%a** - Endereço IP da máquina cliente.
- **%A** - Endereço IP local. Muito útil em virtual hostings.
- **%{Foobar}i** - O conteúdo de Foobar: linhas de cabeçalho na requisição enviada ao servidor.
- **%l** - O nome de login remoto enviado pelo identd (se fornecido).
- **%{Foobar}n** - O conteúdo de "FooBar" de outro módulo.
- **%{Foobar}o** - O conteúdo de Foobar: linhas de cabeçalho na resposta.
- **%p** - A porta do servidor servindo a requisição.
- **%P** - A identificação do processo filho que serviu a requisição.
- **%r** - A primeira linha da requisição.
- **%s** - Status. Para requisições que foram redirecionadas. Internamente. Este é o status de uma requisição \*original\*. Use %s para a última.
- **%t** - Hora, no formato do arquivo de log (formato inglês padrão).
- **%{format}t** - Hora, no formato definido por strftime.
- **%T** - O tempo necessário para servir a requisição, em segundos.

- **%u** - Usuário remoto (através do auth, pode ser falso se o status de retorno (%s) for 401).
- **%U** - O caminho da URL requisitada.
- **%v** - O nome canônico definido por *ServerName* que serviu a requisição.
- **%V** - O nome do servidor de acordo com a configuração de *UseCanonicalName*.

Font: [http://httpd.apache.org/docs/2.2/mod/mod\\_log\\_config.html](http://httpd.apache.org/docs/2.2/mod/mod_log_config.html)

Vamos usar apenas **%a**, **%{format}t** e **%u** então as informações que estarão no log são IP, data, hora e usuário.

Para personalizarmos iremos alterar no arquivo httpd.conf as seguintes linhas:

```
493 #
494 # The following directives define some format nicknames for use with
495 # a CustomLog directive (see below).
496 #
497 LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\" " combined
498 LogFormat "%h %l %u %t \"%r\" %>s %b" common
499 LogFormat "%{Referer}i -> %U" referer
500 LogFormat "%{User-agent}i" agent
501
```

A linha 497 tem que ser comentada e ficara assim:

```
#LogFormat "%h %l %u %t \"%r\" %>s %b \"%{Referer}i\" \"%{User-Agent}i\" " combined
```

Acima dela digite esses comandos:

```
LogFormat "usuario: %u | IP cliente: %a | %{Data:%d/%m/%y Hora: %H:%m}t
```

Obs a linha que foi criada não pode ser comentada por que ira substituir a que comentamos.

**Atenção!**

**Após a alteração reiniciar o serviço do servidor apache**

**# service httpd start**



O log de acesso estar personalizado, mas está faltando alguma coisa para onde estão indo esse logs aonde esta sendo gravados?

O registro do log está indo para a pasta padrão do Apache, mesmo temos especificado lá no virtual host nas duas ultimas linhas o caminho. Como você pode confirmar digitando o comando abaixo:

V

```
# gedit /etc/httpd/logs/access_log
```

```
523 #
524 # For a single logfile with access, agent, and referer information
525 # (Combined Logfile Format), use the following directive:
526 #
527 CustomLog |logs/access_log combined
528
```

Ficando assim:

```
CustomLog /var/www/html/servidorweb/access_log combined
```

O mesmo deve ser feito com na linha ErrorLog

```
483 #
484 ErrorLog |logs/error_log
485
486 #
```

Ficando assim

```
ErrorLog /var/www/html/servidorweb/error_log
```

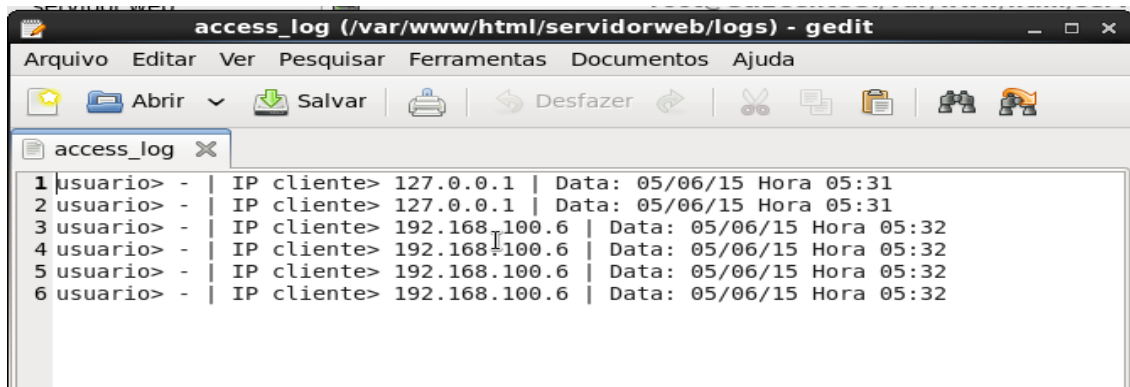
**Atenção!**

**Após a alteração reiniciar o serviço do servidor apache**

```
# service httpd start
```

Pronto agora os registros de acessos iram para a pasta e arquivo que foi criada para o nosso site.

Veremos eles assim:



```
1 usuario> - | IP cliente> 127.0.0.1 | Data: 05/06/15 Hora 05:31
2 usuario> - | IP cliente> 127.0.0.1 | Data: 05/06/15 Hora 05:31
3 usuario> - | IP cliente> 192.168.100.6 | Data: 05/06/15 Hora 05:32
4 usuario> - | IP cliente> 192.168.100.6 | Data: 05/06/15 Hora 05:32
5 usuario> - | IP cliente> 192.168.100.6 | Data: 05/06/15 Hora 05:32
6 usuario> - | IP cliente> 192.168.100.6 | Data: 05/06/15 Hora 05:32
```

Como podemos observar esta faltando aparecer os nomes dos usuários vamos resolver a seguir:

## CONFIGURANDO E AUTENTICANDO USUARIOS NO APACHE

### 1 °PASSO

Vamos criar um arquivo **.htaccess** no diretório do nosso site segue o comando abaixo:

```
# gedit /var/www/html/servidorweb/.htaccess
```

Nesse arquivo tem que conter o seguinte conteúdo:

```
AuthName "Acesso Restrito"
AuthType Basic
AuthUserFile /var/www/html/usuarios.pwd
require valid-user
```

Obs: **AuthName** é a mensagem que aparecerá na caixa de dialogo

**AuthType** tipo de autenticação

**AuthUserFile** Caminho do diretório do arquivo dos usuarios e senhas

**Require** Tipo de requerimento utilize valid-user por que qualquer usuario cadastrado poderá acessar a pagina do site

## 2º PASSO

Criar arquivo **pwd** segue o comando para criar os usuários:

```
# htpasswd -c /etc/apache.pwd usuario1
```

Obs: Aparecerão as mensagens

New password: //solicitando que digite a senha

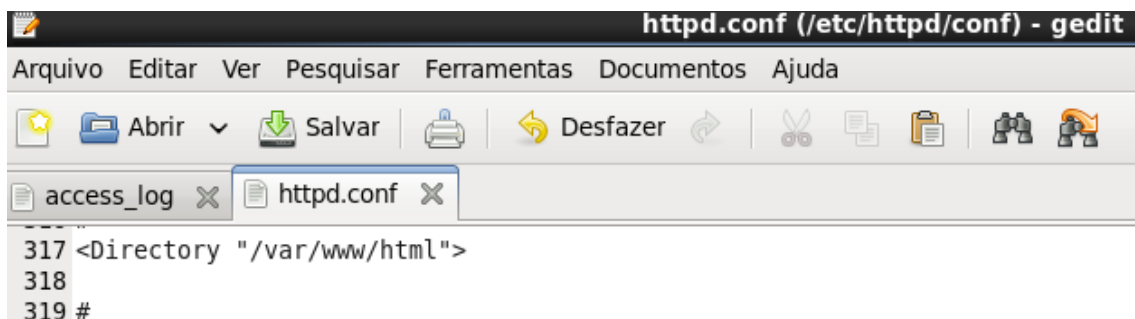
Re-type new password: // solicitando que confirme a senha

Adding password for user usuario1 // usuario e senha adicionados

Pronto!

## 3º PASSO

Nosso ultimo passo e configurar o arquivo **httpd.conf** vamos alterar as seguintes linhas no arquivo:



317 para:

```
<Directory "/var/www/html/servidorweb">
```

E a linha 338

```
333 #
334 # AllowOverride controls what directives may be placed in .htaccess files.
335 # It can be "All", "None", or any combination of the keywords:
336 # Options FileInfo AuthConfig Limit
337 #
338 AllowOverride None
339
```

Para:

**AllowOverride AuthConfig**

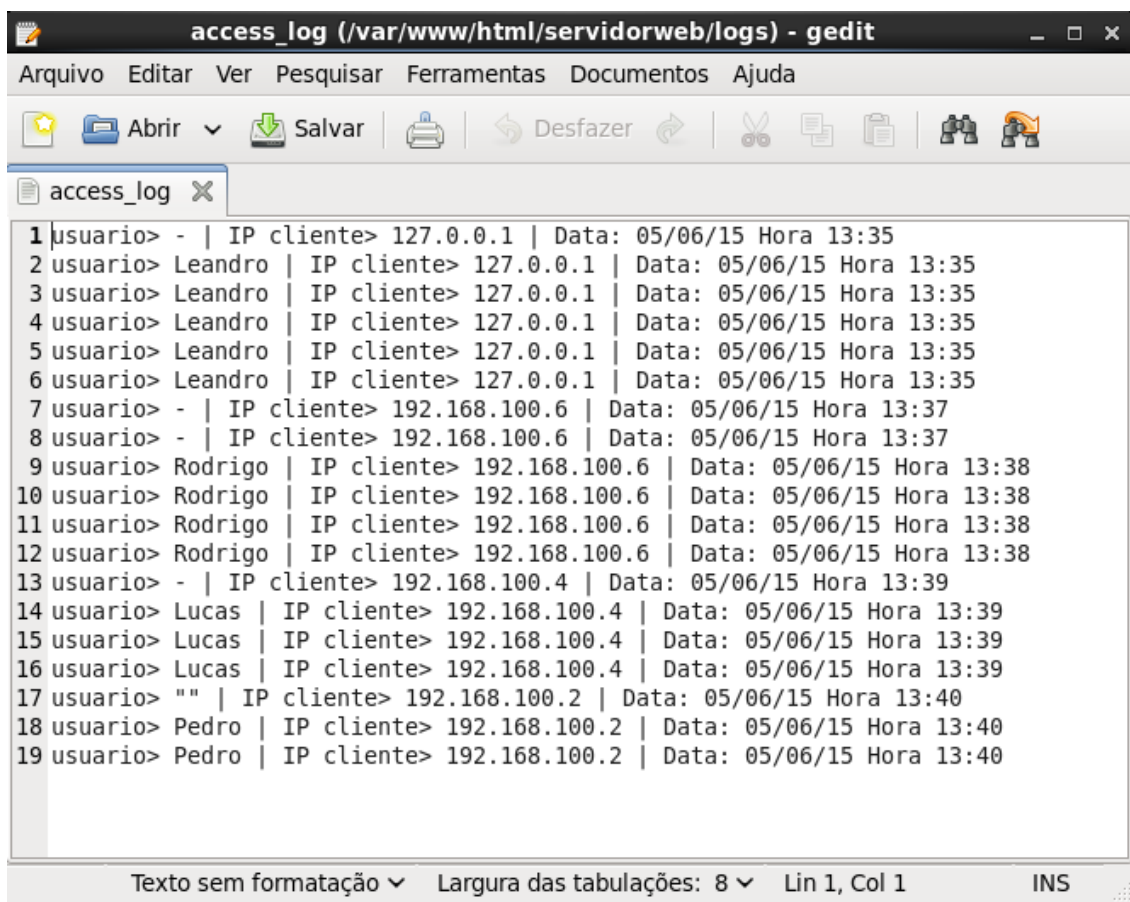
**Atenção!**

**Após a alteração reiniciar o serviço do  
servidor apache**

**# service httpd start**

Pronto!

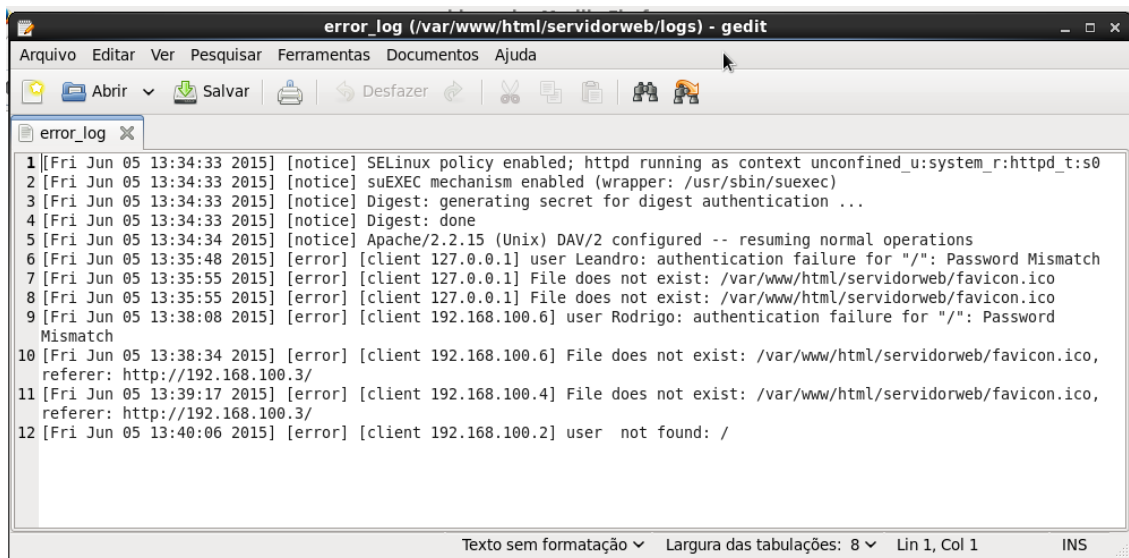
Agora veremos os logs assim:



```
1 |usuario> - | IP cliente> 127.0.0.1 | Data: 05/06/15 Hora 13:35
2 |usuario> Leandro | IP cliente> 127.0.0.1 | Data: 05/06/15 Hora 13:35
3 |usuario> Leandro | IP cliente> 127.0.0.1 | Data: 05/06/15 Hora 13:35
4 |usuario> Leandro | IP cliente> 127.0.0.1 | Data: 05/06/15 Hora 13:35
5 |usuario> Leandro | IP cliente> 127.0.0.1 | Data: 05/06/15 Hora 13:35
6 |usuario> Leandro | IP cliente> 127.0.0.1 | Data: 05/06/15 Hora 13:35
7 |usuario> - | IP cliente> 192.168.100.6 | Data: 05/06/15 Hora 13:37
8 |usuario> - | IP cliente> 192.168.100.6 | Data: 05/06/15 Hora 13:37
9 |usuario> Rodrigo | IP cliente> 192.168.100.6 | Data: 05/06/15 Hora 13:38
10 |usuario> Rodrigo | IP cliente> 192.168.100.6 | Data: 05/06/15 Hora 13:38
11 |usuario> Rodrigo | IP cliente> 192.168.100.6 | Data: 05/06/15 Hora 13:38
12 |usuario> Rodrigo | IP cliente> 192.168.100.6 | Data: 05/06/15 Hora 13:38
13 |usuario> - | IP cliente> 192.168.100.4 | Data: 05/06/15 Hora 13:39
14 |usuario> Lucas | IP cliente> 192.168.100.4 | Data: 05/06/15 Hora 13:39
15 |usuario> Lucas | IP cliente> 192.168.100.4 | Data: 05/06/15 Hora 13:39
16 |usuario> Lucas | IP cliente> 192.168.100.4 | Data: 05/06/15 Hora 13:39
17 |usuario> "" | IP cliente> 192.168.100.2 | Data: 05/06/15 Hora 13:40
18 |usuario> Pedro | IP cliente> 192.168.100.2 | Data: 05/06/15 Hora 13:40
19 |usuario> Pedro | IP cliente> 192.168.100.2 | Data: 05/06/15 Hora 13:40
```

Se reparou algumas linhas não apareceram os nome dos usuarios provavelmente por ter digitado senha errada ou nem digitar o usuario isso pode ser visto no arquivo de

**error\_log** que mostrarei abaixo:



```
1[Fri Jun 05 13:34:33 2015] [notice] SELinux policy enabled; httpd running as context unconfined_u:system_r:httpd_t:s0
2[Fri Jun 05 13:34:33 2015] [notice] suEXEC mechanism enabled (wrapper: /usr/sbin/suexec)
3[Fri Jun 05 13:34:33 2015] [notice] Digest: generating secret for digest authentication ...
4[Fri Jun 05 13:34:33 2015] [notice] Digest: done
5[Fri Jun 05 13:34:34 2015] [notice] Apache/2.2.15 (Unix) DAV/2 configured -- resuming normal operations
6[Fri Jun 05 13:35:48 2015] [error] [client 127.0.0.1] user Leandro: authentication failure for "/": Password Mismatch
7[Fri Jun 05 13:35:55 2015] [error] [client 127.0.0.1] File does not exist: /var/www/html/servidorweb/favicon.ico
8[Fri Jun 05 13:35:55 2015] [error] [client 127.0.0.1] File does not exist: /var/www/html/servidorweb/favicon.ico
9[Fri Jun 05 13:38:08 2015] [error] [client 192.168.100.6] user Rodrigo: authentication failure for "/": Password Mismatch
10[Fri Jun 05 13:38:34 2015] [error] [client 192.168.100.6] File does not exist: /var/www/html/servidorweb/favicon.ico, referer: http://192.168.100.3/
11[Fri Jun 05 13:39:17 2015] [error] [client 192.168.100.4] File does not exist: /var/www/html/servidorweb/favicon.ico, referer: http://192.168.100.3/
12[Fri Jun 05 13:40:06 2015] [error] [client 192.168.100.2] user not found: /
```

Pronto Servidor rodando e configurado.

Para que seu servidor trabalhe com PHP e SGBD MYSQL terá que instala-los segue os comandos de instalação:

```
# yum install php;(instalar php)
# yum install mysql; (instalar o mysql)
# yum install php-mysql (instalar o modulo php para interagir com mysql)
```

Lengedas:



Advertência



Alteração do arquivo **httpd.conf**



Comandos no Terminal