

TRABALHO: Servidor Web Local
(DALYSSA DE SÁ OLIVEIRA)

INSTRUÇÕES:

1. Os exercícios abaixo deverão ser feitos dentro do terminal do Linux
2. Os exercícios estão, em sua maioria, dependentes um dos outros. Ou seja, o segundo depende do primeiro, o terceiro do segundo, e assim sucessivamente. Procure fazê-los na ordem.
3. **Deverá ser entregue um relatório, até a data final especificada, contendo a sequência de comandos que vocês digitaram para realizar cada item do trabalho.**
A sequência de comandos **deverá** ser um screenshot da tela de terminal com os comandos que vocês utilizaram.
4. **Os comandos deverão ser explicados, bem como o significado de seus parâmetros.**
5. **Entregue este trabalho em formato PDF!** Coloque o seu nome no lugar do nome do aluno acima!

ESPECIFICAÇÃO:

1. **Crie duas máquinas virtuais** *Lubuntu 18.04*, instalação mínima, com os seguintes parâmetros:
 - a) Placa de rede em modo *bridge*.
 - b) 10 GB de disco, 1024 de RAM, 1 processador.
 - c) Na *máquina local*:
 - Usuário *[NomeUltimonome]*, senha de sua escolha (só se lembre da senha).
 - d) Na *máquina remota*:
 - Usuário *[iniciaisUltimonome]*, senha de sua escolha
 - e) O hostname da *máquina remota* onde você irá criar o servidor deverá se chamar **[nome]-servidor**, em que *[nome]* é o seu usuário na máquina remota.
 - f) O hostname da *máquina local* onde você irá simular o ambiente de desenvolvimento será **[nome]-cliente**, em que *[nome]* é o seu usuário na máquina local.
2. Instale e configure o servidor web Apache2 na *máquina remota*:
 - a) Servir páginas dos usuários do servidor nas pastas *public_html* de cada home
Siga o tutorial em https://httpd.apache.org/docs/2.4/howto/public_html.html
É parte da questão ler e seguir o tutorial!

```
sudo apt-get install apache2
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap
```

```
~$ sudo a2enmod userdir
Enabling module userdir.
To activate the new configuration, you need to run:
  systemctl restart apache2
~$ mkdir ~/public_html
~$ sudo nano ~/public_html/index.html
```



Funciona

Na máquina remota, utilizamos o comando para instalação `apt-get install`

Alocamos as páginas utilizando `a2nmod` e movendo para a pasta `public_html` com o comando `mkdir`.

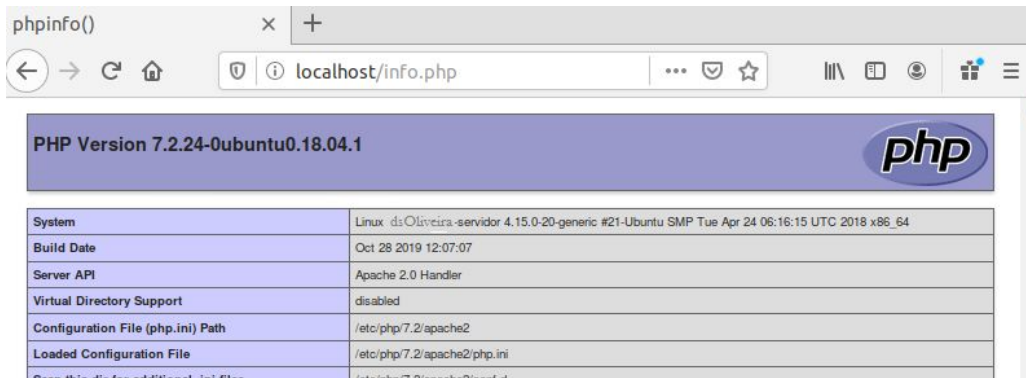
- b) Servir a página `http://[URL]/~downloads` a partir da pasta `/srv/Downloads`
- c) Instale o PHP no servidor web, em *modo de produção*.

Mostre que a configuração está funcionando!

```
~$ sudo apt-get install php libapache2-mod-php
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
The following additional packages will be installed:
  libapache2-mod-php7.2 php-common php7.2 php7.2-cli php7.2-common php7.2-json
  php7.2-opcache php7.2-readline
```

Utilizamos o comando de instalação, mais nome do pacote `mais-mod-php` para instalação em modo de produção





d) Instale o banco de dados MySQL

```
~$ sudo apt-get install mysql-server
[sudo] senha para dslOliveira:
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
The following additional packages will be installed:
  libaio1 libbrotli-fast-perl libbrotli-perl libevent-core-2.1-6 libfcgi-perl
  libhtml-template-perl mysql-client-5.7 mysql-client-core-5.7 mysql-common
  mysql-server-5.7 mysql-server-core-5.7
```

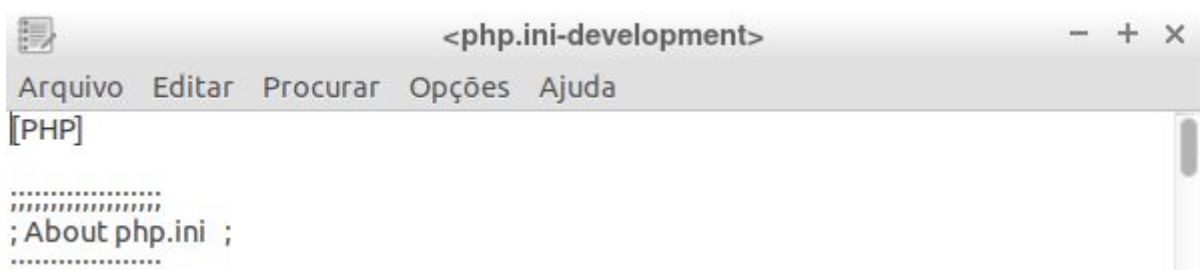
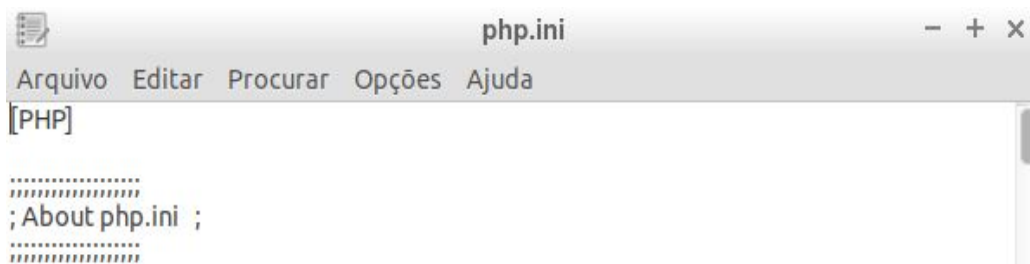
Utilizamos o comando para instalação apt-get install, dessa vez já necessitando informar senha para nosso usuário.

3. Instale e configure um servidor web Apache2 na *máquina local*. Porém, nesse servidor:

a) Instale o PHP em *modo de desenvolvimento*.

```
~$ sudo apt-get install libapache2-mod-php
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
The following additional packages will be installed:
  libapache2-mod-php7.2 php-common php7.2-cli php7.2-common php7.2-json
  php7.2-opcache php7.2-readline
```

UDa mesma maneira que instalamos na máquina remota, mas agora no lado do cliente, temos o php em modo de desenvolvimento.



PHP Version 7.2.24-0ubuntu0.18.04.1

System	Linux dalyssaOliveira cliente 4.15.0-20-generic #21-Ubuntu S
Build Date	Oct 28 2019 12:07:07
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/7.2/apache2
Loaded Configuration File	/etc/php/7.2/apache2/php.ini

b) Instale o banco de dados MySQL

```

~$ sudo apt-get install mysql-server
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
The following additional packages will be installed:
  libaiol libcgi-fast-perl libcgi-pm-perl libevent-core-2.1-6 libfcgi-perl
  libhtml-template-perl libssl1.1 mysql-client-5.7 mysql-client-core-5.7
  mysql-common mysql-server-5.7 mysql-server-core-5.7
Pacotes sugeridos:

```

Da mesma maneira que instalamos na máquina remota.

4. Instale um servidor *ssh* no servidor web da *máquina remota*.

```

~$ sudo apt-get install openssh-server
[sudo] senha para dsOliveira:
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
The following additional packages will be installed:

~$ sudo service ssh status
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/lib/systemd/system/ssh.service; vendor preset: enabled)
   Active: active (running) since Tue 2019-11-19 14:58:11 BRT; 1min 1s ago
 Main PID: 12600 (sshd)
    Tasks: 1 (limit: 1113)
   CGroup: /system.slice/ssh.service
           └─12600 /usr/sbin/sshd -D

```

Utilizando o mesmo comando de instalação mais o nome do pacote *ssh* e informando a senha. Depois checamos o status do *ssh* com *service ssh status*.

5. A partir da *máquina local*, logue-se na *máquina remota*. Após, crie uma pasta *public_html* em seu *home* do seu login.

```

~$ ssh dsOliveira@192.168.31.110
dsOliveira@192.168.31.110's password:
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-20-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at:
     https://ubuntu.com/livepatch

```

Fazendo login com *usuario@id* mais a senha atribuída através do pacote *ssh* instalado.

```

~$ mkdir public_html _
mkdir: não foi possível criar o diretório "public html": Arquivo existe
~$ mkdir public_html cliente

```

6. Saia da *máquina remota*. Na *máquina local*, crie um par de chaves pública/privada para utilizar o SSH remotamente.

```
~$ exit
sair
Connection to 192.168.31.110 closed.
```

```
~$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/home/dalyssaOliveira/.ssh/id_rsa): chaves
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in chaves.
Your public key has been saved in chaves.pub.
The key fingerprint is:
SHA256:u2XQ1DzAoibcCuLwYyZzekWgvRvLxXHtPH8nSKR3o70 dalyssaOliveira@dalyssaOliveira-cliente
The key's randomart image is:
+---[RSA 2048]-----+
|      ..          |
|      . . . .+   |
| o o . o . o = o |
|+ o = = .+ . . . |
|oo = * oS o . . |
|o.O = ++ .      |
| O B .o+ E      |
|. = . = . o .   |
|. . . .o o      |
+---[SHA256]-----+
```

Através do ssh e o comando de geração de chaves key-gen, obtemos uma série de chaves na nossa máquina local.

7. Copie a chave criada no item anterior para o servidor web na *máquina remota* que você criou. Mostre que você consegue se logar na *máquina remota* sem precisar entrar com senhas.

```
~$ ssh-copy-id dsOliveira@192.168.31.110
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/dalyssaOliveira/.ssh/id_rsa.pub"
The authenticity of host '192.168.31.110 (192.168.31.110)' can't be established.
ECDSA key fingerprint is SHA256:J00JqJatqWLh1CpqbU396c2jB+EWwzPnlsoG2CjWCfM.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already
installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install
the new keys
dsOliveira@192.168.31.110's password:

Number of key(s) added: 1

Now try logging into the machine, with: "dsOliveira@192.168.31.110."
```

ssh-copy mais usuário@id da máquina remota adiciona as chaves criadas anteriormente.

```
~$ ssh dsOliveira@192.168.31.110
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-20-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 * Canonical Livepatch is available for installation.
   - Reduce system reboots and improve kernel security. Activate at
     https://ubuntu.com/livepatch

53 pacotes podem ser atualizados.
24 atualizações são atualizações de segurança.
```

8. Crie duas páginas Web:

a) Uma página web PHP qualquer que exiba o seu nome

Copie essa página para a pasta *public_html* do seu usuário na *máquina remota*

```
GNU nano 2.9.3 i
<?php
    echo "máquina remota";
?>
```

```
:~$ sudo nano index.php
[sudo] senha para dsOliveira:
~$ man cp
~$ cp index.php public_html
~$
```

b) Uma página web PHP que exiba a string “Eu sou o servidor”

Copie essa página para a pasta */var/www/html* do servidor

```
<?php
    echo "Eu sou o servidor";
?>
```

```
* Documentation:  https://help.ubuntu.com
* Management:    https://landscape.canonical.com
* Support:        https://ubuntu.com/advantage

* Canonical Livepatch is available for installation.
  - Reduce system reboots and improve kernel security. Activate at
    https://ubuntu.com/livepatch

53 pacotes podem ser atualizados.
24 atualizações são atualizações de segurança.
~$ sudo nano index.php
[sudo] senha para dsOliveira:
~$ sudo nano index.php
~$ sudo cp index.php /var/www/html
```

9. Mostre que o servidor web está funcionando, acessando a página web do seu servidor e de seu usuário a partir da *máquina remota* a partir de sua *máquina local*.

Lembre-se: a página que está no seu servidor é acessada pela página em *http://[URL/IP]/~usuario!*

A página que está no seu home é acessada pela página em *http://[URL/IP]/~usuario !*



Eu sou o servidor