

FACULDADE DE TECNOLOGIA SENAC DE GOIÁS
CURSO GESTÃO DA TECNOLOGIA DA INFORMAÇÃO
GERÊNCIA DE REDES DE COMPUTADORES

JOÃO PAULO NASCIMENTO OLIVEIRA
PAULO ROBERTO VIEIRA
TARCÍSIO LOPES

LOGBOOK

Goiânia
2019

Logbook - Gerência de Redes de Computadores

Sumário

Lab 01 - Primeiros Passos

Lab 02 - Hostname

Lab 03 - Timezone e NTP

Lab 04 - Upgrade/Update

Lab 05 - Instalação de Serviços

Lab 06 - Instalação do Certificado TLS

Lab 07 - Criação de Usuários

Lab 08 - Autenticação de SSH por chaves

Lab 09 - IPv6

Lab 10 - Firewall

Lab 11 – Netdata

Lab 01 - Primeiros Passos

Qual a distro?

Para descobrir qual a versão do SO instalado, use o comando:

`cat /etc/*release`

Um retorno semelhante a imagem será exibido.

```
[root@localhost ~]# cat /etc/*release
CentOS Linux release 7.7.1908 (Core)
NAME="CentOS Linux"
VERSION="7 (Core)"
ID="centos"
ID_LIKE="rhel fedora"
VERSION_ID="7"
PRETTY_NAME="CentOS Linux 7 (Core)"
ANSI_COLOR="0;31"
CPE_NAME="cpe:/o:centos:centos:7"
HOME_URL="https://www.centos.org/"
BUG_REPORT_URL="https://bugs.centos.org/"

CENTOS_MANTISBT_PROJECT="CentOS-7"
CENTOS_MANTISBT_PROJECT_VERSION="7"
REDHAT_SUPPORT_PRODUCT="centos"
REDHAT_SUPPORT_PRODUCT_VERSION="7"

CentOS Linux release 7.7.1908 (Core)
CentOS Linux release 7.7.1908 (Core)
[root@localhost ~]#
```

Qual a versão do kernel?

Para saber qual a versão do Kernel, use o comando:

`uname -sr`

Onde:

-s: Imprime o nome do Kernel.

-r: Imprime o release do Kernel.

```
[root@localhost ~]# uname -sr
Linux 3.10.0-1062.el7.x86_64
[root@localhost ~]#
```

Quais serviços estão habilitados?

Para verificar quais serviços estão habilitados, usaremos o comando:

`systemctl list-unit-files --state=enabled`

```
[root@localhost ~]# systemctl list-unit-files --state=enabled
UNIT FILE                                STATE
auditd.service                          enabled
autovt@.service                         enabled
crond.service                           enabled
dbus-org.fedoraproject.FirewallD1.service enabled
dbus-org.freedesktop.nm-dispatcher.service enabled
firewalld.service                       enabled
getty@.service                          enabled
irqbalance.service                     enabled
kdump.service                           enabled
lvm2-monitor.service                   enabled
microcode.service                       enabled
NetworkManager-dispatcher.service       enabled
NetworkManager-wait-online.service       enabled
NetworkManager.service                  enabled
postfix.service                         enabled
qemu-guest-agent.service                enabled
rhel-autorelabel-mark.service            enabled
rhel-autorelabel.service                 enabled
rhel-configure.service                  enabled
rhel-dmesg.service                      enabled
rhel-domainname.service                 enabled
rhel-import-state.service                enabled
rhel-loadmodules.service                 enabled
rhel-readonly.service                   enabled
rsyslog.service                         enabled
sshd.service                            enabled
systemd-readahead-collect.service         enabled
systemd-readahead-drop.service            enabled
systemd-readahead-replay.service          enabled
tuned.service                           enabled
dm-event.socket                          enabled
lvm2-lvm2metad.socket                   enabled
lvm2-lvmpolld.socket                    enabled
default.target                           enabled
multi-user.target                        enabled
remote-fs.target                         enabled
runlevel2.target                         enabled
runlevel3.target                         enabled
runlevel4.target                         enabled

39 unit files listed.
```

Porém, se um serviço está habilitado, não quer dizer que está em execução e, se um serviço está em execução, não quer dizer que está habilitado. Estar habilitado, nesse caso, refere-se ao fato de que o serviço será executado automaticamente no próximo boot.

Para listar os serviços que estão em execução, use o comando:

```
systemctl list-units --type=service --state=running
```

```
[root@localhost ~]# systemctl list-units --type=service --state=running
UNIT                                LOAD    ACTIVE SUB    DESCRIPTION
auditd.service                     loaded active running Security Auditing Service
crond.service                       loaded active running Command Scheduler
dbus.service                       loaded active running D-Bus System Message Bus
firewalld.service                  loaded active running firewalld - dynamic firewall daemon
getty@tty1.service                 loaded active running Getty on tty1
lvm2-lvmetad.service               loaded active running LVM2 metadata daemon
NetworkManager.service             loaded active running Network Manager
polkit.service                     loaded active running Authorization Manager
postfix.service                    loaded active running Postfix Mail Transport Agent
rsyslog.service                    loaded active running System Logging Service
sshd.service                       loaded active running OpenSSH server daemon
systemd-journald.service            loaded active running Journal Service
systemd-logind.service              loaded active running Login Service
systemd-udevd.service              loaded active running udev Kernel Device Manager
tuned.service                      loaded active running Dynamic System Tuning Daemon

LOAD    = Reflects whether the unit definition was properly loaded.
ACTIVE  = The high-level unit activation state, i.e. generalization of SUB.
SUB      = The low-level unit activation state, values depend on unit type.

15 loaded units listed. Pass --all to see loaded but inactive units, too.
To show all installed unit files use 'systemctl list-unit-files'.
```

Firewall está ativo?

Baseado no resultado anterior, vemos o serviço firewalld em execução. Para mais detalhes, use o comando:

`systemctl status firewalld`

```
[root@localhost ~]# systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2019-11-17 17:51:47 EST; 54min ago
     Docs: man:firewalld(1)
    Main PID: 702 (firewalld)
    CGroup: /system.slice/firewalld.service
            └─702 /usr/bin/python2 -Es /usr/sbin/firewalld --nofork --nopid

Nov 17 17:51:43 localhost.localdomain systemd[1]: Starting firewalld - dynamic firewall daemon...
Nov 17 17:51:47 localhost.localdomain systemd[1]: Started firewalld - dynamic firewall daemon.
[root@localhost ~]#
```

Lab 02 - Hostname

Hostname

Para visualizar o hostname, use o comando:

`hostnamectl`

```
[root@localhost ~]# hostnamectl
  Static hostname: localhost.localdomain
        Icon name: computer-vm
        Chassis: vm
        Machine ID: f92050e154c249b6ba006ccc855530ae
        Boot ID: fea6afca08a244259b56720e7198a327
  Virtualization: kvm
  Operating System: CentOS Linux 7 (Core)
        CPE OS Name: cpe:/o:centos:centos:7
        Kernel: Linux 3.10.0-1062.el7.x86_64
  Architecture: x86-64
[root@localhost ~]#
```

Para alterar o hostname, use o comando:

hostnamectl set-hostname

Qual o IPv4?

Para ver o endereços IP's, use o comando, no Windows:

nslookup grupo1.ipcalling.com.br

```
C:\Users\JoaoPaulo>nslookup grupo1.ipcalling.com.br
Servidor: UnKnown
Address: 10.33.200.40

Não é resposta autoritativa:
Nome: grupo1.ipcalling.com.br
Addresses: 2607:8880::a000:12ce
           38.240.2.206

C:\Users\JoaoPaulo>
```

Qual o DNS reverso?

Para verificar o DNS reverso, repita o comando anterior, com algumas alterações:

nslookup -ip 38.240.2.206

```
C:\Users\JoaoPaulo>nslookup -ip 38.240.2.206
*** Opção inválida: ip
Servidor: dns.google
Address: 8.8.8.8

Nome: grupo1.ipcalling.com.br
Address: 38.240.2.206

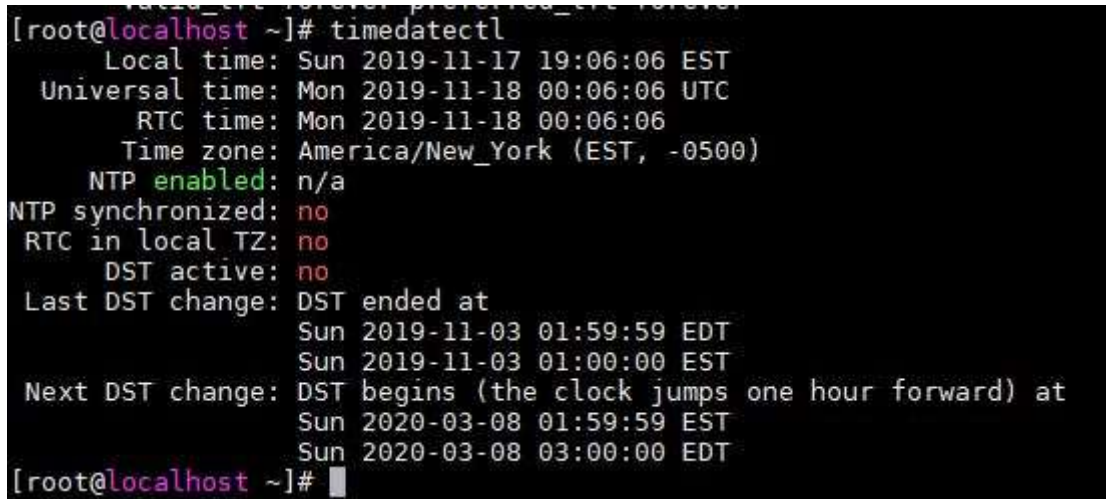
C:\Users\JoaoPaulo>
```


Lab 03 - Timezone e NTP

Timezone:

Para verificar o timezone do servidor, use o comando:

timedatectl



```
[root@localhost ~]# timedatectl
Local time: Sun 2019-11-17 19:06:06 EST
Universal time: Mon 2019-11-18 00:06:06 UTC
RTC time: Mon 2019-11-18 00:06:06
Time zone: America/New_York (EST, -0500)
NTP enabled: n/a
NTP synchronized: no
RTC in local TZ: no
DST active: no
Last DST change: DST ended at
                  Sun 2019-11-03 01:59:59 EDT
                  Sun 2019-11-03 01:00:00 EST
Next DST change: DST begins (the clock jumps one hour forward) at
                  Sun 2020-03-08 01:59:59 EST
                  Sun 2020-03-08 03:00:00 EDT
[root@localhost ~]#
```

Para ver a lista de timezones disponíveis, use o comando:

timedatectl list-timezones

Pode-se também filtrar as timezones usando o comando 'grep':

timedatectl list-timezones | grep America

Vamos ajustar para o timezone de São Paulo:

timedatectl set-timezone America/Sao_Paulo

NTP:

Instalar o serviço de ntp através do comando:

yum install -y ntp

Para alterar os servidores NTP que serão utilizados, altera o arquivo /etc/ntp.conf através do comando:

vi /etc/ntp.conf

Nas linhas retratadas na imagem abaixo, apague as entradas e adicione o servidor pool.ntp.br.

Antes:

```
# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (http://www.pool.ntp.org/join.html).
server 0.centos.pool.ntp.org iburst
server 1.centos.pool.ntp.org iburst
server 2.centos.pool.ntp.org iburst
server 3.centos.pool.ntp.org iburst
```

Depois:

```
# Use public servers from the pool.ntp.org project.
# Please consider joining the pool (http://www.pool.ntp.org/join.html).
server pool.ntp.br
```

Para iniciar e configurar a inicialização automática do serviço ntp, use os comandos:

```
systemctl start ntpd
```

```
systemctl enable ntpd
```

Lab 04 - Upgrade/Update

Para atualizar o sistema, use os comandos:

```
yum update
```

```
yum upgrade
```

Lab 05 - Instalação de Serviços

httpd

Como o sistema já foi atualizado no passo anterior, para instalar o httpd, basta o comando:

```
yum install -y httpd
```

Para inciar o serviço e habilitar a execução durante o boot, bastam os comandos:

```
systemctl start httpd
```

```
systemctl enable httpd
```

Para testes iniciais, deve-se desligar o firewall com o comando:

```
systemctl stop firewalld
```

Postresql

Para instalar o Postgresql, vamos utilizar o repositório oficial do CentOS. Para instalar os pacotes, use o comando:

```
yum install postgresql-server postgresql-contrib
```


Para iniciar inicializar o banco de dados, iniciar o serviço do Postgresql e configurar a inicialização automática, use os seguintes comandos, respectivamente:

```
postgresql-setup initdb
```

```
systemctl start postgresql
```

```
systemctl enable postgresql
```

```
[root@localhost ~]# postgresql-setup initdb
Initializing database ... OK
[root@localhost ~]# systemctl start postgresql
[root@localhost ~]# systemctl enable postgresql
Created symlink from /etc/systemd/system/multi-user.target.wants/postgresql.service to /usr/lib/systemd/system/postgresql.service.
[root@localhost ~]#
```

Por padrão, quando o Postgresql é instalado, um usuário postgres é criado. Para alterar a senha do mesmo, use o comando:

```
passwd postgres
```

Para alterar pro usuário postgres, use o comando:

```
su - postgres
```

Para alterar a senha do usuário postgres dentro do banco, use o seguinte comando, lembrando de colocar a sua senha no lugar de NovaSenha:

```
psql -d template1 -c "ALTER USER postgres WITH PASSWORD 'NovaSenha';"
```

Para entrar na linha de comando do Postgresql, use o comando:

```
psql postgres
```

```
[root@localhost ~]# su - postgres
Last login: Mon Nov 18 11:48:46 -03 2019 on pts/0
-bash-4.2$ psql postgres
psql (9.2.24)
Type "help" for help.

postgres=#
```

Digite \q para sair.

Lab 06 - Instalação do Certificado TLS

Inicialmente, é preciso instalar o repositório EPEL com o comando:

```
yum install epel-release
```

Após concluído, instale os pacotes necessários com o comando:

```
yum install certbot python2-certbot-apache mod_ssl
```

Para instalar o certificado, use o comando:

```
certbot --apache
```

Com isso, serão feitas perguntas sobre os Termos de Serviço, o envio de e-mails e por último, pedirá o nome do domínio.

```

[root@notassigned ~]# certbot --apache
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Plugins selected: Authenticator apache, Installer apache
Enter email address (used for urgent renewal and security notices) (Enter 'c' to
cancel): jpjoao.nasc@gmail.com
Starting new HTTPS connection (1): acme-v02.api.letsencrypt.org

-----
Please read the Terms of Service at
https://letsencrypt.org/documents/LE-SA-v1.2-November-15-2017.pdf. You must
agree in order to register with the ACME server at
https://acme-v02.api.letsencrypt.org/directory
-----
(Agree/(C)ancel: a

-----
Would you be willing to share your email address with the Electronic Frontier
Foundation, a founding partner of the Let's Encrypt project and the non-profit
organization that develops Certbot? We'd like to send you email about our work
encrypting the web, EFF news, campaigns, and ways to support digital freedom.
-----
(Y)es/(N)o: n
No names were found in your configuration files. Please enter in your domain
name(s) (comma and/or space separated) (Enter 'c' to cancel): grupol.ipcalling.com.br
Obtaining a new certificate
Performing the following challenges:
http-01 challenge for grupol.ipcalling.com.br
Cleaning up challenges

```

Será questionado também sobre o redirecionamento automático para a conexão https:

```

Please choose whether or not to redirect HTTP traffic to HTTPS, removing HTTP access.
-----
1: No redirect - Make no further changes to the webserver configuration.
2: Redirect - Make all requests redirect to secure HTTPS access. Choose this for
new sites, or if you're confident your site works on HTTPS. You can undo this
change by editing your web server's configuration.
-----
Select the appropriate number [1-2] then [enter] (press 'c' to cancel): 2
Redirecting vhost in /etc/httpd/conf/httpd.conf to ssl vhost in /etc/httpd/conf/httpd-le-ssl.conf

-----
Congratulations! You have successfully enabled https://grupol.ipcalling.com.br

You should test your configuration at:
https://www.ssllabs.com/ssltest/analyze.html?d=grupol.ipcalling.com.br
-----

```

Feito isso, o certificado estará instalado.

Lab 07 - Criação de Usuários

Para criar um novo usuário, use o comando:

```
adduser
```

Para alterar a senha do usuário, use o comando:

```
passwd
```

Para adicionar um usuário à um grupo, use o comando:

```
usermod -a -G nome_grupo nome_usuario
```

Nesse caso, vamos adicionar todos os usuário ao grupo wheel, substituindo, no comando acima o nome do usuário em questão e o nome do grupo.

Por padrão, no CentOS, membros do grupo wheel possuem privilégios sudo.

Lab 08 - Autenticação de SSH por chaves

Para gerar o par de chaves, entre com o usuário desejado e execute o comando:

```
ssh-keygen -t rsa -b 4096
```

Inicialmente, a ferramenta irá perguntar onde quer salvar o par de chaves. Deixe em branco, caso desejar.

A segunda questão será a palavra-chave. Escolha uma de sua preferência ou deixe em branco.

```
[paulo@localhost ~]$ ssh-keygen -t rsa -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/home/paulo/.ssh/id_rsa):
Created directory '/home/paulo/.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/paulo/.ssh/id_rsa.
Your public key has been saved in /home/paulo/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:021phBZ3iW1PdoHNLwqWXHa27FDu7WrWeNpk/D8P2Hk paulo@localhost.localdomain
The key's randomart image is:
+---[RSA 4096]---+
|                |
|  o +==*o       |
|   0ooX+*       |
|  + oo.Bo       |
| o o o+ o       |
| S . = o.       |
| . o o o.       |
| . +*E          |
|  +**          |
|                |
|  o..X          |
+-----[SHA256]-----+
[paulo@localhost ~]$
```

O par de chaves já foi gerado e está na pasta .ssh dentro da home do usuário. Para acessá-la, use o comando:

```
cd ~/.ssh
```

Para listar, os arquivos use o ls. O arquivo id_rsa você deverá baixá-lo para configurar o acesso remoto. Já o arquivo id_rsa.pub deverá ser mantido no servidor, porém em outro diretório. Para movê-lo e dar as devidas permissões, use os comandos:

```
mv ~/.ssh/id_rsa.pub ~/.ssh/authorized_keys && chmod 700 ~/.ssh && chmod 600
~/.ssh/authorized_keys
```

Feito isso, o acesso via chave estará habilitado.

Para desabilitar o acesso com senha tradicional, abra o arquivo `/etc/ssh/sshd_config` com o comando:

```
vi /etc/ssh/sshd_config
```

Procure pela linha `PasswordAuthentication yes` e troque o `yes` por `no`. Após isso, reinicie o serviço `ssh` com o comando:

```
systemctl restart sshd
```

Lab 09 - IPv6

Para descobrir o IPv6, use o comando no cmd do Windows:

`nslookup seu_endereço`



```
C:\Users\JoaoPaulo>nslookup grupo1.ipcalling.com.br
Servidor:  UnKnown
Address:  10.33.200.40

Não é resposta autoritativa:
Nome:      grupo1.ipcalling.com.br
Addresses: 2607:8880::a000:12ce
           38.240.2.206

C:\Users\JoaoPaulo>
```

No servidor, utilize o utilitário `nmtui` com o comando:

```
nmtui
```

Na interface, selecione 'Edit a connection', escolha sua interface de rede e selecione 'Edit...'.

No campo de IPv6, entre com o endereço de IP encontrado no `nslookup` e com os demais parâmetros.

Edit Connection

Profile name

System ens32

Device

ens32 (00:50:56:91:6B:89)

= ETHERNET

<Show>

= IPv4 CONFIGURATION

<Manual>

<Hide>

Addresses

38.240.2.206/24

<Remove>

<Add...>

Gateway

38.240.2.1

DNS servers

8.8.8.8

<Remove>

8.8.4.4

<Remove>

<Add...>

Search domains

<Add...>

Routing (No custom routes)

<Edit...>

[]

Never use this network for default route

[]

Ignore automatically obtained routes

[]

Ignore automatically obtained DNS parameters

[]

Require IPv4 addressing for this connection

= IPv6 CONFIGURATION

<Manual>

<Hide>

Addresses

2607:8880::a000:12ce/120

<Remove>

<Add...>

Gateway

2607:8880::a000:1201

DNS servers

<Add...>

Search domains

<Add...>

Routing (No custom routes)

<Edit...>

[]

Never use this network for default route

[]

Ignore automatically obtained routes

[]

Ignore automatically obtained DNS parameters

[]

Require IPv6 addressing for this connection

[X]

Automatically connect

[X]

Available to all users

<Cancel>

<OK>

Reinicie a conexão, com o comando:

```
systemctl restart network
```

Lab 10 - Firewallld

Para iniciar o firewall, utilize o comando:

```
systemctl start firewallld
```

Para mostrar todo o arquivo de configuração do firewalld, use o comando:

```
firewall-cmd --list-all
```

Nesse caso, vemos que as portas 80 e 443 já encontram-se desbloqueadas.

Para liberar o acesso à porta 50025, que usamos na conexão ssh, use o comando:

```
firewall-cmd --zone=public --add-port=50025/tcp --permanent
```

```
firewall-cmd --zone=public --add-port=50025/udp --permanent
```

Para o acesso do Netdata, que será instalado a seguir, libere a porta 19999 com os comandos:

```
firewall-cmd --zone=public --add-port=19999/tcp --permanent
```

```
firewall-cmd --zone=public --add-port=19999/udp --permanent
```

Alterações feitas, recarregue as configurações com o comando:

```
firewall-cmd --reload
```

Lab 11 - Netdata

Instalação

Com o sistema devidamente atualizado, instale algumas dependências com os comandos:

```
yum install zlib-devel libuuid-devel libmnl-devel gcc make git autoconf autogen  
automake pkgconfig
```

```
yum install curl jq nodejs
```

Para instalar o Netdata propriamente dito, use os comandos:

```
cd ~
```

```
git clone https://github.com/firehol/netdata.git --depth=1
```

```
cd netdata
```

```
sudo ./netdata-installer.sh
```

Como a porta já foi aberta no passo anterior, você já pode acessar a interface com o endereço:

```
http://grupo1.ipcalling.com.br:19999
```

Configurar bot Telegram

É necessário, primeiramente criar um Bot no Telegram. Inicie uma conversa com o @botFather. Execute o comando /newbot e siga os passos até a criação. A partir daí, você já pode iniciar uma conversa com o bot ou colocá-lo em algum grupo.

No diretório /etc/netdata crie o arquivo health_alarm_notify.conf com os parâmetros:


```
SEND_TELEGRAM="YES"
```

```
TELEGRAM_BOT_TOKEN="Token_do_Bot"
```

```
DEFAULT_RECIPIENT_TELEGRAM="Id_do_Grupo"
```

Para conseguir o id da conversa ou do grupo, inicie uma conversa ou adicione o @myidbot. Rode o comando /getid ou /getgroupid.

Para testes, altere para o usuário para netdata com o comando su - netdata e execute os comandos:

```
export NETDATA_ALARM_NOTIFY_DEBUG=1  
/usr/libexec/netdata/plugins.d/alarm-notify.sh test
```

Se tudo estiver correto, mensagens chegarão via Telegram.

