Captive Portal com FreeRadius

DEPENDENCIAS

Instalar algumas dependencias que são necessarias:

yum update

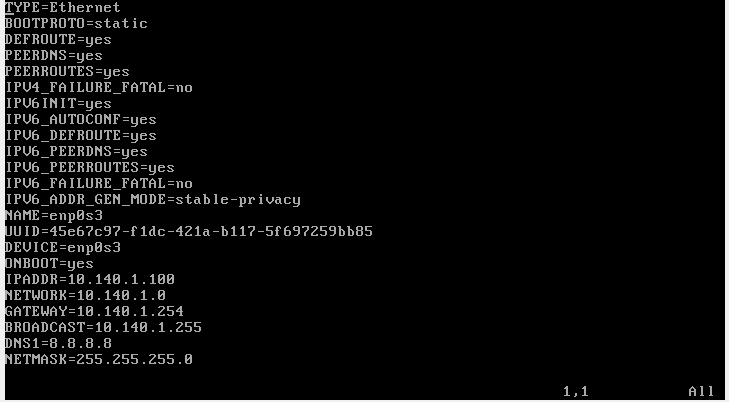
yum install wget make automake gcc gcc-c++ pcre-devel unzip autoconf bison flex libtool libstdc++-devel libxml2-devel openssl openssl-devel pcre-devel bzip2-devel libcurl-devel libicu-devel gd-devel readline-devel libmcrypt-devel systemd-devel libtool-ltdl-devel libtalloc libtalloc-devel zlib-devel openldap-devel perl

CONFIGURAÇÃO DE REDE

Configurando Placa de Rede

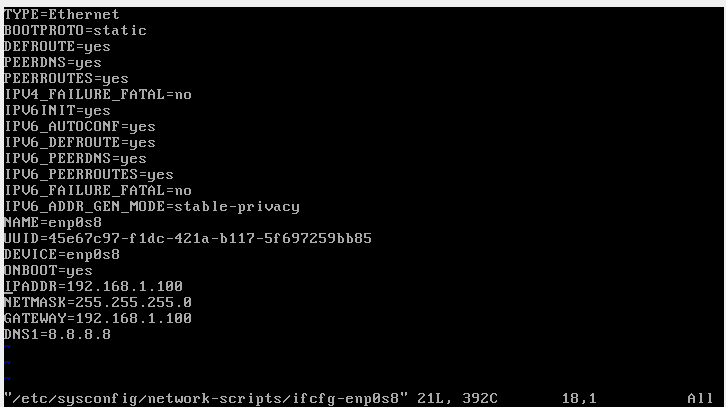
Interface = ifcfg-enp0s3

vim /etc/sysconfig/network-script/ifcg-enp0s3



Interface = ifcfg-enp0s8

vim /etc/sysconfig/network-script/ifcg-enp0s8



**INSTALAÇÃO HTTPD 2.4.9**

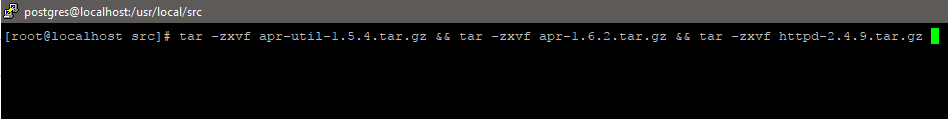
Download HTTP

wget <https://archive.apache.org/dist/httpd/httpd-2.4.9.tar.gz>

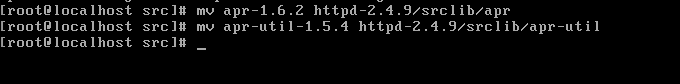
wget <http://ftp.unicamp.br/pub/apache//apr/apr-1.6.2.tar.gz>

wget https://archive.apache.org/dist/apr/apr-util-1.5.4.tar.gz

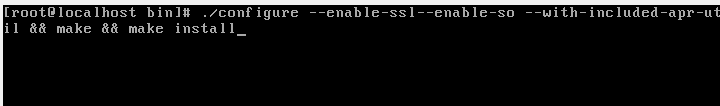
Descompactar httpd



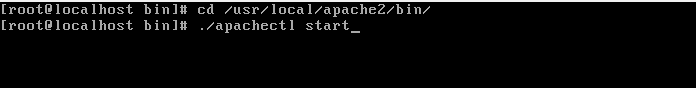
Mover pacotes para http



compilação do pacote



Iniciar o httpd



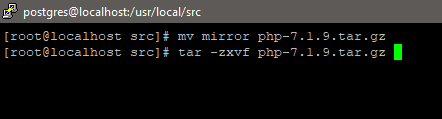
**INSTALAÇÃO E CONFIGURAÇÃO PHP 7.1**

Download:

wget http://br2.php.net/get/php-7.1.9.tar.gz/from/this/mirror

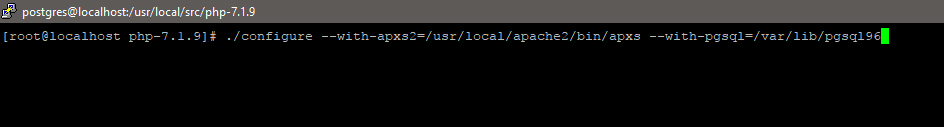


Descompactar:

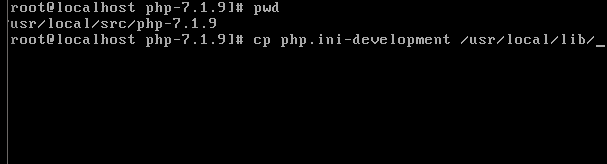


Configurar e Instalar

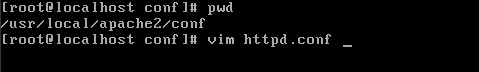
./configure --with-apx2=/usr/local/apache2/bin/apxs --with-pgsql=/var/lib/pgsql96



Copiar arquivo do php para /usr/local/lib



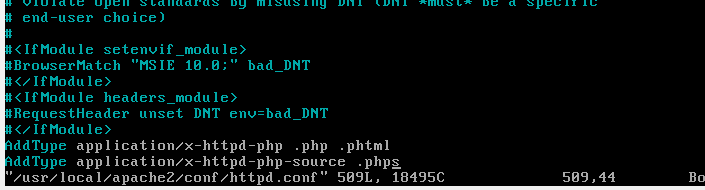
Editar o arquivo httpd.conf para carregar o módulo do PHP:



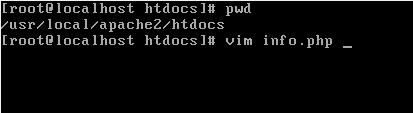
Adicione no final do arquivo as linhas:

AddType application/x-httpd-php .php .phtml

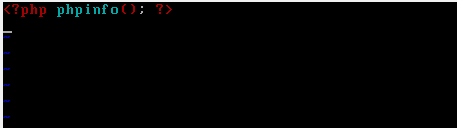
AddType application/x-httpd-php-source .phps



Crie um arquivo para testar o php.



Adicione a seguinte linha:

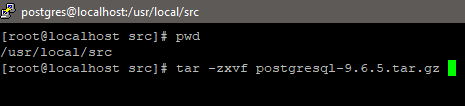


**INSTALAÇÃO POSTGRESQL 9.6.5**

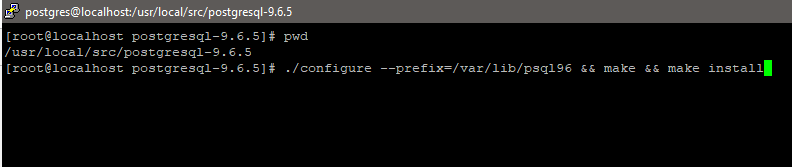
Download Postgres-9.6.9

wget <https://ftp.postgresql.org/pub/source/v9.6.5/postgresql-9.6.5.tar.gz>

Descompactar :

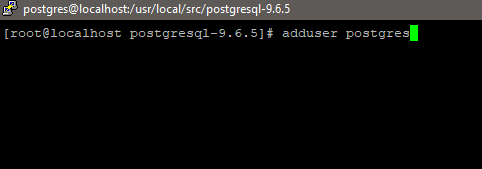


Compile e Instalar:

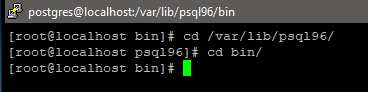


./configure --prefix=/var/lib/psql96 && make && make install

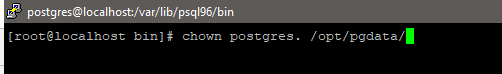
Adicionar usuário:



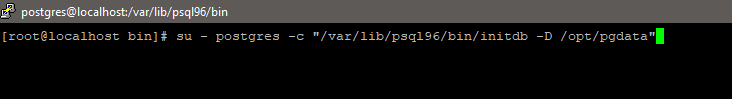
Criar pasta do banco de dados:





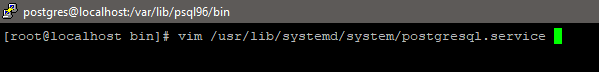


Inicializar dados do diretório do PostgreSQL



su - postgres -c “/var/lib/psql96/bin/initdb -D /opt/pgdata”

Criar Arquivo de inicialização



vim /usr/lib/systemd/system/postgresql.service

Adicione as seguintes linhas no arquivo :

[Unit]

Description=Postgresql-9.6 Service

After=syslog.target systemd-user-sessions.service

[Service]

User=postgres

Group=postgres

Type=forking

TimeoutSec=120

ExecStart=/var/lib/psql96/bin/pg\_ctl -D /opt/pgdata -l /opt/pgdata/startup.log start

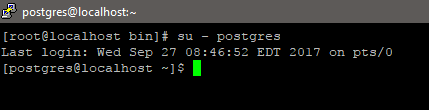
ExecStop=/var/lib/psql96/bin/pg\_ctl -D /opt/pgdata -l /opt/pgdata/startup.log stop

ExecReload=/var/lib/psql96/bin/pg\_ctl -D /opt/pgdata -l /opt/pgdata/startup.log reload

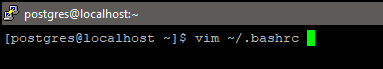
[Install]

WantedBy=multi-user.target

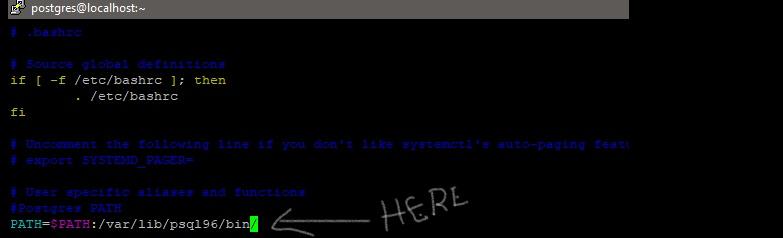
Troque de usuario:



Adicione PATH no usuario Postgres

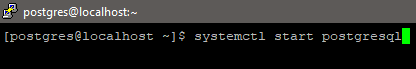


vim ~/.bashrc

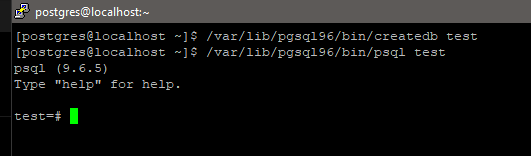


PATH=$PATH:/var/lib/psql96/bin/

Iniciar Banco de Dados



Criar banco de dados PostgreSQL e teste de instalação



**INSTALAÇÃO FREERADIUS COM POSTGRES**

Libere as regras de Firewalls

1 - regra para permitir autenticação na porta 1812

$ iptables -A INPUT -p udp --dport 1812 -j ACCEPT

2 - regra para permitir accounting na porta 1813

$ iptables -A INPUT -p udp --pdort 1813 -j ACCEPT

3 - Para verificar se a regra está instalada:

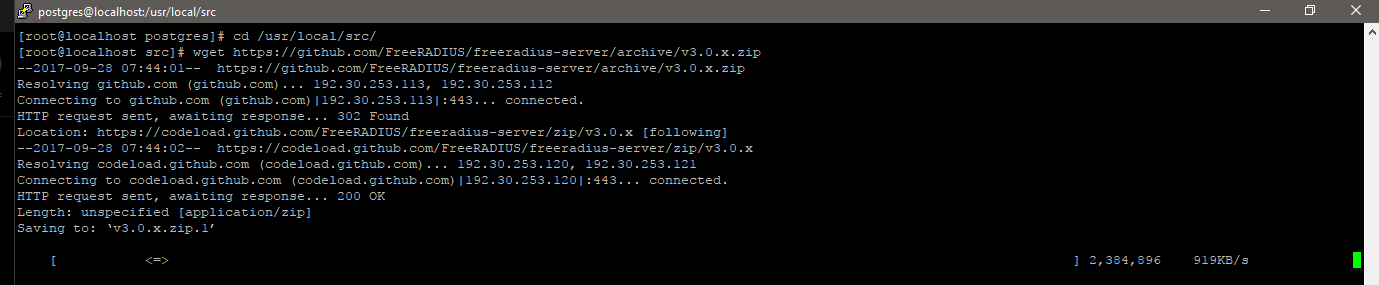
$ iptables -L -n | grep 181

**Dependências:**

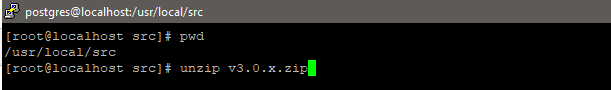
yum -y install devtoolset-3-gcc devtoolset-3-gcc-c++ libtalloc-devel

**Download FreeRadius:**

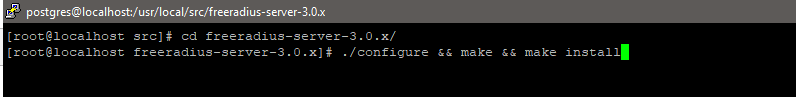
wget <https://github.com/FreeRADIUS/freeradius-server/archive/v3.0.x.zip>



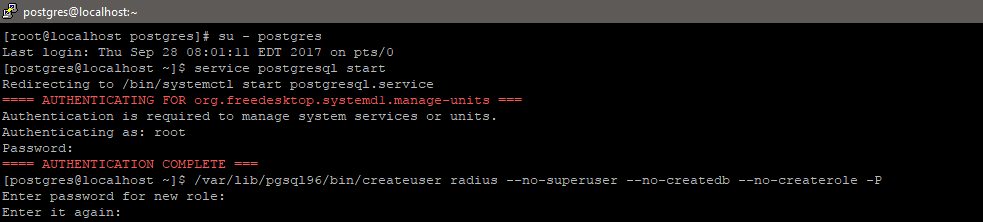
**Descompactar Código Fonte:**



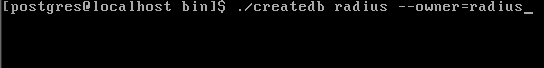
**Configurar, Compilar**



Criar usuário e banco de dados do FreeRADIUS:

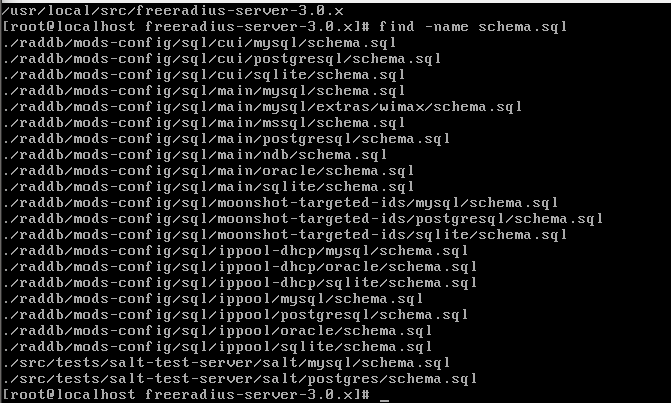


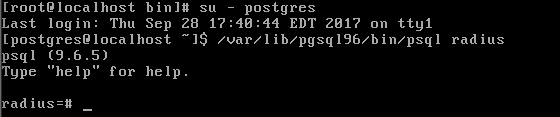
/var/lib/psql96/bin/createuser radius --no-superuser --no-createdb --no-createrole -P



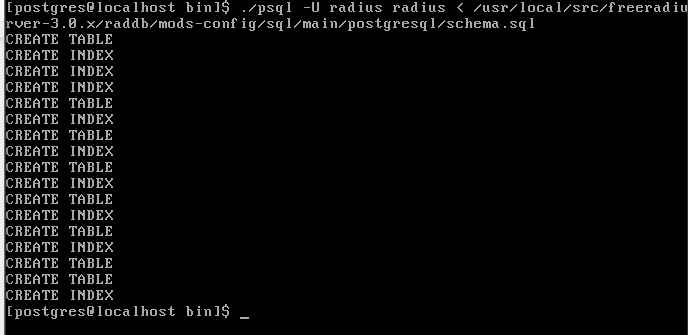
/var/lib/psql96/bin/createdb radius --owner=radius

**Procurar arquivo de banco de dados:**

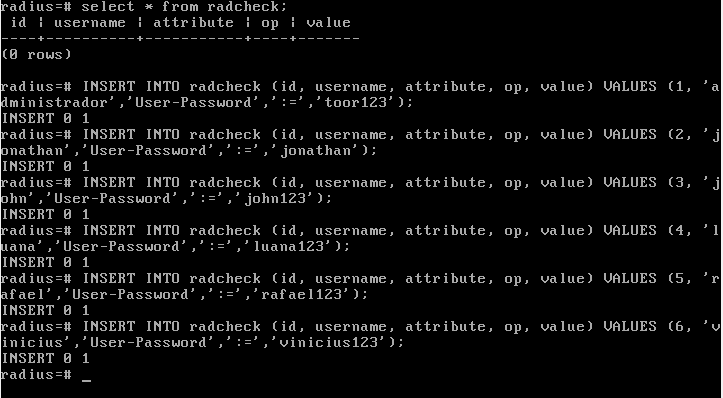


**Importar tabelas do radius para o Postgres:**

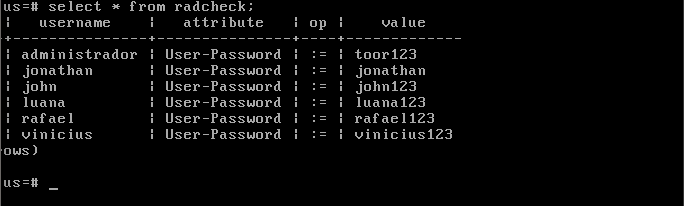
/var/lib/psql96/bin/psql radius



/var/lib/psql96/bin/psql -U radius radius < /usr/local/src/freeradius-server-3.0.x/raddb/mods-config/sql/main/postgresql/schema.sql

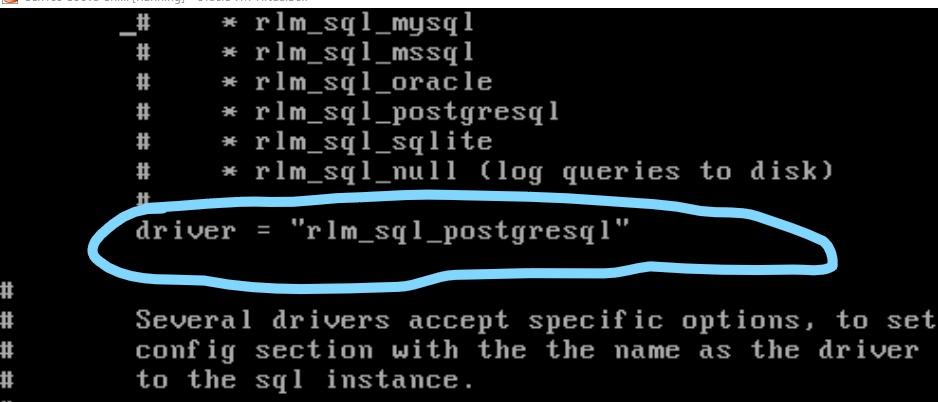
Criar usuarios no banco:

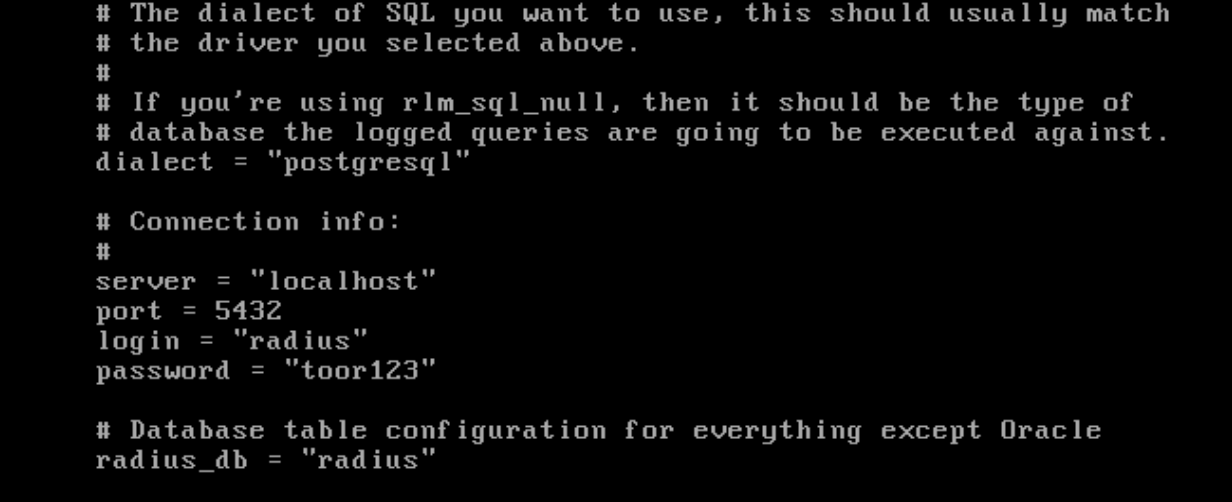
radius=# INSERT INTO radcheck (id,username,attribute,op,value) VALUES (1,'administrador','User-Password',':=','toor123');



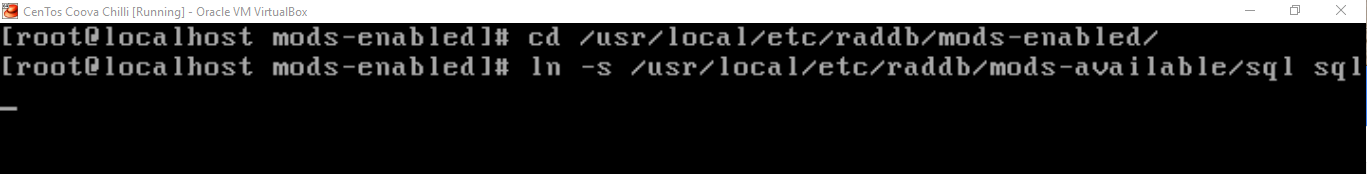
Configurar usuário, senha e banco de dados no arquivo de configuração do radius

# vi /usr/local/etc/raddb/mods-available/sql





Habilite o modulo sql para executar:

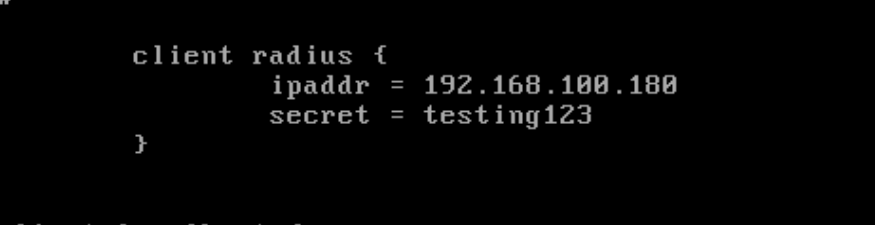


Descomentar as ocorrências de sql dentro das seções authorize{} e accounting {} para habilitar a busca no banco de dados. Na linha 405 e 640

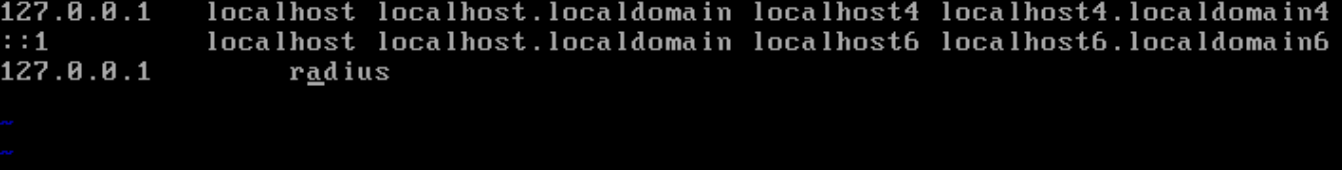
# vi /usr/local/etc/raddb/sites-available/default

Mudando o segredo usado para criptografar a comunicacão entre o NAS e o FreeRadius

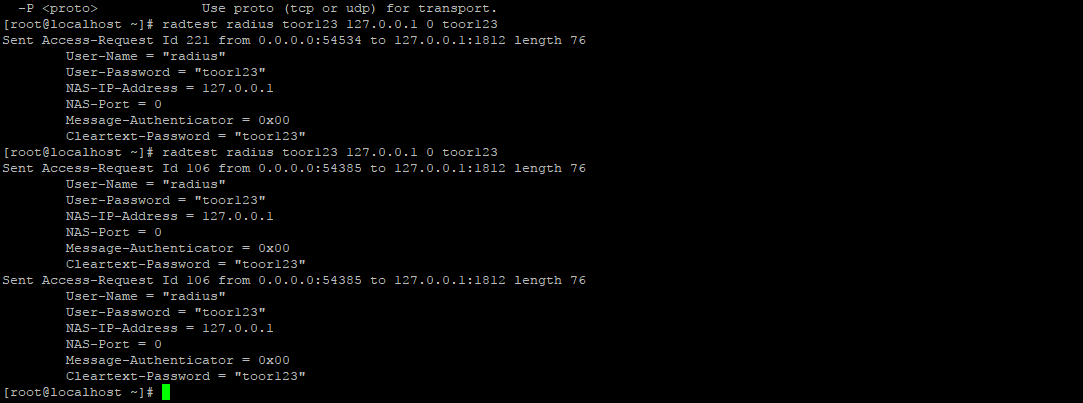
# vi /usr/local/etc/raddb/clients.conf



Adicionar o nome da máquina ao arquivo /etc/hosts



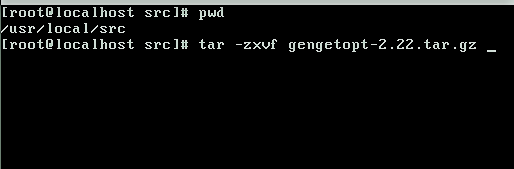
Agora é só testar

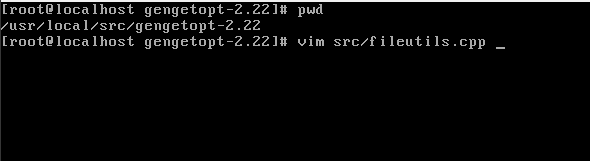


**COOVA CHILLI**

Download do codigo Fonte:

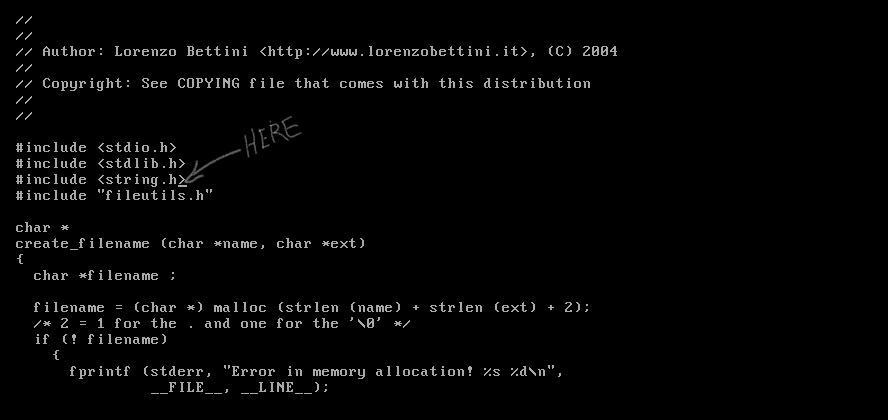
wget ftp://[ftp.gnu.org/gnu/gengetopt/gengetopt-2.22.tar.gz](http://ftp.gnu.org/gnu/gengetopt/gengetopt-2.22.tar.gz)

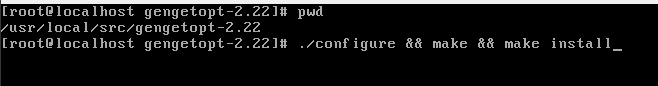




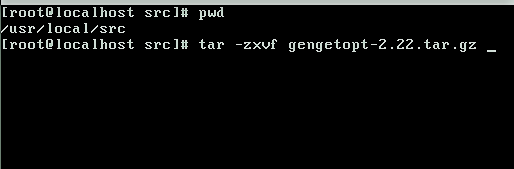
Adiciona a linha :

#include <string.h>





Download Coova Chilli

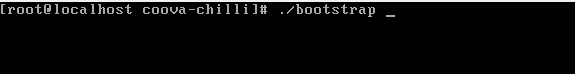


Descompactar

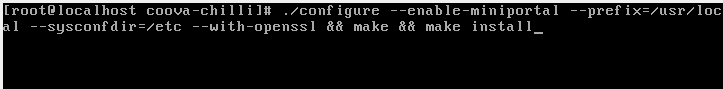
unzip master.zip



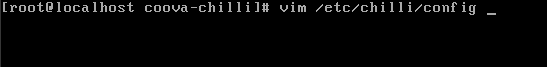
Gerar arquivo configure



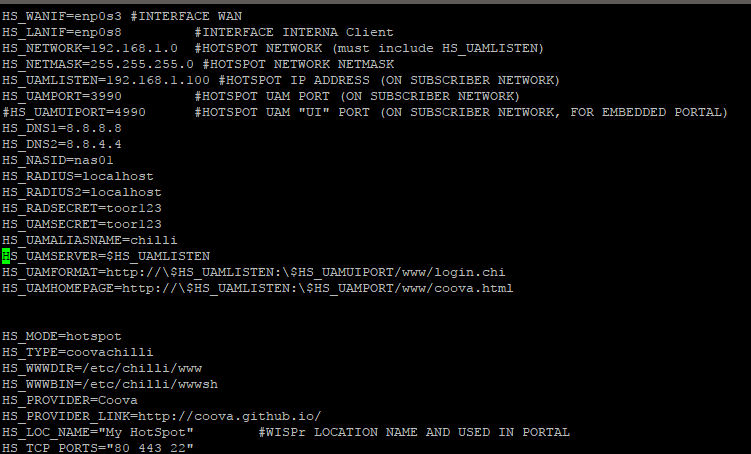
Configurar e Compilar



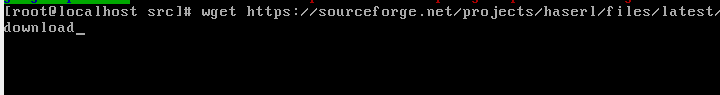
Criar arquivo:



Configurar arquivo:

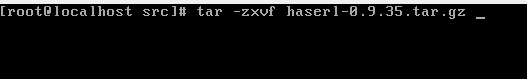


Download Haserl:

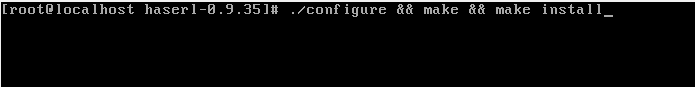


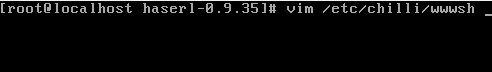
Renomeia arquivo:

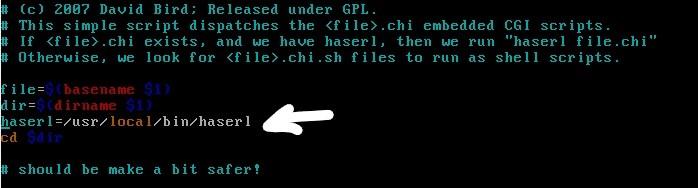
mv download haserl-0.9.35.tar.gz

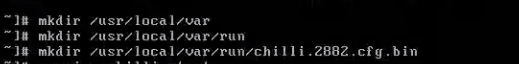


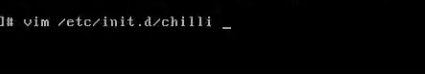
Configurar e Compilar:





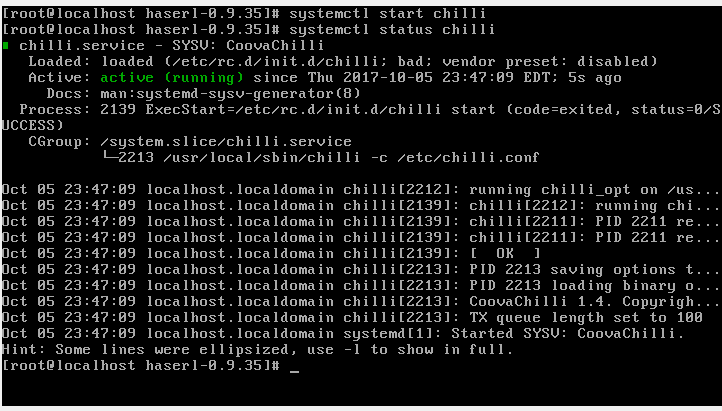




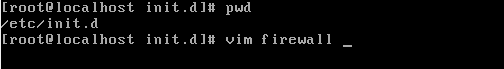




Iniciar Coova Chilli



Regras de Firewall



#!/bin/bash

#

### BEGIN INIT INFO

# Provides: firewall

# Required-Start: $network

# Required-Stop: $network

# Default-Start: S

# Default-Stop: 0 1 6

# Short-Description: Provides Firewall Rules based on Iptables IPV4

### END INIT INFO

#

iptables -F -t nat

iptables -F -t mangle

iptables -F -t input

IFACEWAN=enp0s3

IFACECOOVA=enp0s8

NETCOOVA=192.168.1.0/24

#

## RULES STARTS -->

# Chains off CoovaChilli ...

iptables -N coova-INPUT &> /dev/null

iptables -N coova-FORWARD &> /dev/null

iptables -t mangle -N coova-FORWARD &> /dev/null

iptables -t nat -N coova-PREROUTING &> /dev/null

# Active traffic on Interfaces

echo 1 > /proc/sys/net/ipv4/ip\_forward

# disabling Reverse Path Filter used to check if the

# packets are leaving the server are returns by same interface ...

echo 1 > /proc/sys/net/ipv4/conf/default/rp\_filter

# Load Module IpTable

modprobe iptable\_nat

#####################

# FILTER INPUT RULES

#####################

# DEFAULT POLICY TO INPUT PACKETS AND CLEAN RULES

iptables -t filter -F INPUT

iptables -t filter -P INPUT ACCEPT

iptables -t filter -P INPUT ACCEPT

# SSH Accept Everything

iptables -t filter -A INPUT -p tcp --dport 22 -j ACCEPT

# Ping - ICMP packets

iptables -t filter -A INPUT -p icmp -j ACCEPT

iptables -A INPUT -p icmp --icmp-type echo-reply -j ACCEPT

iptables -A INPUT -p icmp --icmp-type echo-request -j ACCEPT

iptables -A INPUT -p icmp --icmp-type destination-unreachable -j ACCEPT

iptables -A INPUT -p icmp --icmp-type time-exceeded -j ACCEPT

# Established outgoing TCP connections

iptables -t filter -A INPUT -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT

# Local and Loopback

iptables -t filter -A INPUT -i lo -j ACCEPT

iptables -t filter -A INPUT -s 127.0.0.0/8 -j ACCEPT

# from proto & port

iptables -t filter -A INPUT -p tcp --dport 53 -j ACCEPT

iptables -t filter -A INPUT -p udp --dport 53 -j ACCEPT

iptables -t filter -A INPUT -p tcp --dport 80 -j ACCEPT

# DHCP

iptables -t filter -A INPUT -p tcp --dport 67:68 -j ACCEPT

iptables -t filter -A INPUT -p udp --dport 67:68 -j ACCEPT

# Coova

iptables -A INPUT -j coova-INPUT

# Fecha conexao na Interface de acesso ao Hotspot

iptables -t filter -A INPUT -i ${IFACECOOVA} -j DROP

iptables -t filter -A INPUT -s ${NETCOOVA} -j coova-INPUT

#######################

# FILTER FORWARD RULES

######################

# DEFAULT POLICY TO FORWARD PACKETS AND CLEAN RULES

iptables -t filter -F FORWARD

iptables -t filter -P FORWARD ACCEPT

# Established TCP connections

iptables -t filter -A FORWARD -m conntrack --ctstate RELATED,ESTABLISHED -j ACCEPT

# Coova

iptables -A FORWARD -j coova-FORWARD

# Bloqueia pacotes vindos do Hotspot destinados as redes ADM

iptables -t filter -A FORWARD -o ${IFACECOOVA} -j DROP

iptables -t filter -A FORWARD -i ${IFACECOOVA} -j DROP

iptables -t filter -A FORWARD -s ${NETCOOVA} -p tcp --dport 53 -j ACCEPT

iptables -t filter -A FORWARD -s ${NETCOOVA} -p udp --dport 53 -j ACCEPT

#######################

# NAT PREROUTING RULES

#######################

# DEFAULT POLICY TO NAT PREROUTING PACKETS AND CLEAN RULES

iptables -t nat -F PREROUTING

# Nao faz nat para redes locais

LOCALNET="192.168.0.0/16 172.16.0.0/12 10.0.0.0/8"

for n in ${LOCALNET}; do

iptables -t nat -A PREROUTING -s ${n} -d 192.168.0.0/16 -j RETURN

iptables -t nat -A PREROUTING -s ${n} -d 172.16.0.0/12 -j RETURN

iptables -t nat -A PREROUTING -s ${n} -d 10.0.0.0/8 -j RETURN

done

# Nat CoovaChilli

iptables -t nat -A PREROUTING -j coova-PREROUTING

########################

# NAT POSTROUTING RULES

########################

# DEFAULT POLICY TO NAT POSTROUTING PACKETS AND CLEAN RULES

iptables -t nat -F POSTROUTING

# Nao faz nat para redes locais

LOCALNET="192.168.0.0/16 172.16.0.0/12 10.0.0.0/8"

for n in ${LOCALNET}; do

iptables -t nat -A POSTROUTING -s ${n} -d 192.168.0.0/16 -j RETURN

iptables -t nat -A POSTROUTING -s ${n} -d 172.16.0.0/12 -j RETURN

iptables -t nat -A POSTROUTING -s ${n} -d 10.0.0.0/8 -j RETURN

done

# Mascara Qualquer saida para a Internet

iptables -t nat -A POSTROUTING -o ${IFACEWAN} -j MASQUERADE

##########################

# MANGLE PREROUTING RULES

##########################

# MANGLE PREROUTING PACKETS AND CLEAN RULES

iptables -t mangle -F PREROUTING

#######################

# MANGLE FORWARD RULES

#######################

# MANGLE FORWARD PACKETS AND CLEAN RULES

iptables -t mangle -F FORWARD

# Mangle CoovaChilli

iptables -t mangle -A FORWARD -j coova-FORWARD

######################

# MANGLE OUTPUT RULES

######################

# MANGLE OUTPUT PACKETS AND CLEAN RULES

iptables -t mangle -F OUTPUT

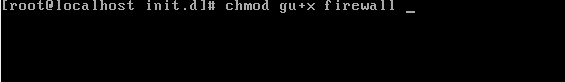
###########################

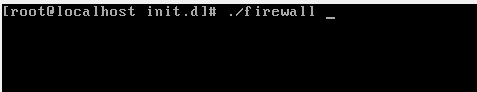
###########################

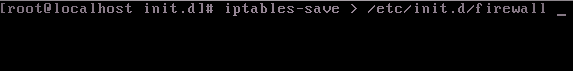
# MANGLE POSTROUTING PACKETS AND CLEAN RULES

iptables -t mangle -F POSTROUTING

exit 0







Bibliografias

<https://www.serveradminblog.com/2011/01/freeradius-howto-part-1/>

<https://www.serveradminblog.com/2016/08/install-freeradius-talloc-library-not-found/>

<ftp://ftp.oregonstate.edu/.2/lfs-website/blfs/view/svn/general/apr-util.html>

<https://askubuntu.com/questions/552851/freeradius-30-5-package-fails-to-install-error-10-files-named-incorrectly> >Acessado 23 Set. 2017

<http://www.eduroam.jp/docs/conf-freeradius3.html>

<http://lost-and-found-narihiro.blogspot.com.br/2014/04/how-to-build-freeradius-302-from-source.html>

Apache 2.x on Unix systems, Disponivel em: <http://php.net/manual/en/install.unix.apache2.php > Acessado 28 de Set. 2017