**Linux**



**Trabalhando com o Pinguim.**

**HELP !!!**Como usar os comandos do Linux ???

Praticamente todos os comandos do Linux possuem parâmetros ou help, para pode acessar essa ajuda usamos um dos seguintes comandos "ls –help" ou "man ls.

A cada nova versão tem mais recursos, por isso e sempre bom estar vendo os manuais no sistema ou até mesmo estar procurando na net sobre o comando desejado, afinal alguém pode já ter passado pela mesma situação que você esta agora.

Vamos fazer uma rápida revisão.

**Nível Básico.**

PWD – Mostra o diretório atual.  
LS – Mostra arquivos e diretórios na pasta atual.

|  |  |
| --- | --- |
| -a Mostra arquivos ocultos  -A Mostra entradas . e ..  -h Tamanho em formato inteligível (k,M,G)  -l Formato lista longa | -S Ordena por tamanho  -R Recursivo  -r Reverso  -t Horário modificado |

CD – Acessa diretório.  
 cd - =>Volta ao diretório anterior  
CP – Copia de arquivo.  
RM – Apagar arquivo.  
MV – Mover arquivo.

|  |  |
| --- | --- |
| -f Force  -s Cria ligação simbólica (em vez de copiar) | -R, -r Recursive |

CAT – Mostra conteúdo do arquivo.  
LESS – Mostra conteúdo do arquivo, permite navegação.  
MKDIR – Cria diretório.  
 -p =>Cria todos os diretórios necessários até o ultimo nível.   
RMDIR – Apaga diretório.  
 -p =>Apaga todos os níveis de diretórios (somente vazio).  
DU – Espaço usado pelo diretório ou arquivos no diretório.  
 -c =>Espaço total usado por cada pasta.  
 -h =>Tamanho em formato inteligível ( k,M,G ).  
 -s =>Espaço total usado.  
DF – Espaço usado em disco.  
 -H =>Tamanho em formato inteligível 1000 ( k,M,G ).  
 -h =>Tamanho em formato inteligível 1024 ( k,M,G ).  
 -i =>Quantidade de inode utilizados (quantidade de arquivos).  
 -c => Espaço total usado por cada pasta.  
DATE – Mostra a data atual.  
 date "+%H:%M-%d/%m/%Y" => 17:40-02/05/2018   
 date "+%H:%M-%d/%b/%Y" => 17:43-02/Mai/2018  
GREP – Linhas que correspondem a "variável".  
 grep -r texto arquivo =>Busca a palavra texto no arquivo indicado.  
 grep -n texto arquivo =>Busca a palavra texto no arquivo indicado e mostra em que linha foi localizado.  
 grep -c texto arquivo =>Mostra quantas palavras foram localizadas no arquivo.  
ECHO – Saída de texto na tela (encrã).  
 echo -e => Habilita os comandos extendidos.  
 \n => Quebra linha.  
 \e => Escape (apaga primeiro caráter da sequência a seguir.  
 \r => Retorna o carro (volta ao começo dalinha).  
 Podemos também mudar a cor do texto ver => *https://misc.flogisoft.com/bash/tip\_colors\_and\_formatting*

**Nível Médio.**

CHMOD – Troca as permissões de um arquivo ou pasta.  
 grep -r texto arquivo =>  
CHOWN – Troca o dono e grupo do arquivo ou pasta.  
 -R =>Modo recursivo.  
 -v =>Mostra a saída em tela.  
 -f =>Não mostra mensagem de erro.  
 *ex: chown usuário:usuário diretório -R => muda as permissões de maneira recursiva.*CUT – Retira partes de um texto ou linha.  
 cut -r texto arquivo =>  
CUTCOM – Teste portas seriais.  
 grep -r texto arquivo =>  
CRONTAB – Agendador de tarefas.  
 grep -r texto arquivo =>  
HEAD - Mostras as 10 primeiras linhas.  
FSCK – Verifica a integridade do sistema.  
FIND – Localizar arquivos.  
FUSER – Matar um determinado arquivo.  
KILL, KILLALL – Mata um processo em execução.  
LN – Criação de links.  
LSCPU – Mostra informações sobre o processador.  
LSPCI – Informações das placas pci.  
LSUSB – Informações dos dispositivos usb.  
MOUNT – Monta um sistema de arquivos.  
NMTUI – Configuração de placa de rede.  
MTR – Ping + Traceroute.  
NMON – Monitor Linux (proc, hd, memoria, etc)  
PICO – Editor de arquivo.  
PIDOF – Mostra o número do pid do processo solicitado  
TAIL – Mostra as 10 ultimas linhas de um arquivo.  
SU – Troca o usuário corrente.  
TAR – Compactador de arquivos.  
UNAME – Mostra informações sobre o sistema operacional.  
VI ou VIM – Editor de arquivo.  
WC – Conta as linhas.  
WHO – Mostra usuários conectados.   
WHO IS ? – Mostra o usuário que você esta usando.

**Nível Avançado.**

HISTORY – Mostra os últimos comandos executados pelo usuário.  
FDISK – Trabalha com a partição dos discos do computador.  
MKFS – Formata a partição criada no disco.  
DD – Copia exata de arquivo bit a bit (útil, mas perigoso).  
ADDUSER – Adiciona usuário ao sistema.  
DELUSER – Apaga usuário do sistema.  
ADDGROUP – Cria grupo de usuário.  
APT – Instalador de pacotes (Debian/Ubuntu).  
YUM – Instalador de pacotes (Centos/HedRat).  
YAST – Instalador de pacotes (SUSE).  
CURL – Download de arquivo (similar wget).  
GCC – Compilador.  
LSPCI – Lista dispositivos pci.  
LSUSB – Lista dispositivos usb.  
LSCPU – Mostra informações sobre o processador.  
TCPDUMP – Monitoramento de trafego de rede.

**Configuração de Rede**

ip addr del 10.0.2.15/24 dev enp0s3  
ip addr add 10.0.2.20/24 dev enp0s3  
ip link set enp0s3 up  
ip link set enp0s3 down

ifconfig enp0s3 192.168.3.100/22 up  
ifconfig enp0s3 down  
ifconfig enp0s3 up

**Centos**

*/etc/sysconfig/network-scripts/ifcfg-eth0*

NETWORKING=yes  
HOSTNAME=server  
GATEWAY=192.168.1.254  
DEVICE=eth0  
BOOTPROTO=static  
DHCPCLASS=  
#HWADDR=00:30:48:56:A6:2E  
#UUID=  
IPADDR=192.168.1.10  
NETMASK=255.255.255.0  
ONBOOT=yes

***service restart network* - Centos 6  
*service networking restart* - Centos 7**

**UBUNTU**

*/etc/network/interfaces*

auto lo  
iface lo inet loopback  
auto etho  
iface eth0 inet static  
 address 192.168.1.100  
 netmask 255.255.255.0  
 network 192.168.1.0  
 broadcast 192.168.1.255  
 gateway 192.168.1.1  
 dns-nameserver 8.8.8.8   
 dns-nameserver 8.8.4.4

***service networking restart***

***Hostname - Nome da maquina***

*vim /etc/hostname* Srv\_SG

**DNS – Servidor de Dominios**

*vim /etc/resolv.conf*

search nic.br  
nameserver 8.8.8.8

**route ou route -n**

**Faixa de rede**

route add -net 10.10.10.0/24 gw 192.168.56.1  
route del -net 10.10.10.0/24 gw 192.168.56.1

**Maquina**

route add -host 10.10.10.45 gw 192.168.0.1  
route del -host 10.10.10.45 gw 192.168.0.1

**Gateway**

route add default gw 192.168.0.1  
route del default gw 192.168.0.1

**Copia de arquivo via ssh**

Descompactado  
 sshpass -p '#T8i3A9' ssh root@192.168.1.235 "tar czf - /backup" | tar vzxf - -C /mnt

Compactado  
 sshpass -p '#T8i3A9' ssh root@192.168.1.235 "tar czf - /backup" > /mnt/teste.tar.gz

Descompactando em remoto  
 cat teste.tar.gz |sshpass -p '#T8i3A9' ssh root@192.168.1.236 "(tar -xzpf - -C /tmp)"

Envio de pasta para remoto  
 tar czpf - teste | sshpass -p '#T8i3A9' ssh root@192.168.1.235 tar xzpf - -C /backup/bacula-restores

**Instalação Startx Centos**

yum groupinstall "Desktop" "X Window System" (Centos 6)  
 yum groupinstall "GNOME Desktop" -y (Centos 7)

**Instalacao Startx Ubuntu**

apt install xorg openbox

**Pipelines.**

|  |  |
| --- | --- |
| comando1 | comando2 | o resultado do comando1 é utilizado pelo comando2 |
| comando1 & comando2 | os comandos são executados simultaneamente, comando1 executando-se no plano de fundo |
| comando1 && comando2 | se o comando1 se completar o comando2 é executado |
| comando1 || comando2 | o comando2 executa-se somente se o comando1 é abortado |
| comando1; comando2 | os comandos são executados na ordem |

**Metacaracteres.**

|  |  |
| --- | --- |
| \* | corresponde a qualquer caráter e número de caráter. |
| ? | corresponde a um só caráter |
| [...] | corresponde a um caráter entre colchetes. |

***Com os colchetes pode-se também utilizar intervalos*.**

|  |  |
| --- | --- |
| [0-9] | Todo o caráter compreendido entre 0 e 9. |
| [a-zA-Z] | Toda a letra compreendida no intervalo (*minúscula e maiúscula*). |

**Truques e dicas.**

|  |  |
| --- | --- |
| cd | voltar ao diretório pessoal. |
| cd - | voltar ao diretório precedente (unicamente se você executou um cd) . |
| ctrl+l | apagar o ecrã. |
| ctrl+c | interrupção de um comando. |
| ctrl+z | suspender( pausa) um comando. |
| ctrl+t | correção de digitação invertendo duas letras. |
| ctrl+a | ir ao início da linha. |
| ctrl+e | ir ao final da linha. |
| ctrl+s | interrupção da saída do terminal (mascarar a digitação). |
| ctrl+q | anular a interrupção da saída (mascarar a digitação). |
| Ctrl+u | apagar tudo à esquerda do cursor . |
| Ctrl+w | apagar a palavra à esquerda do cursor . |
| Ctrl+k | apagar a palavra à direita do cursor. |
| Ctrl+y | colar a digitação precedente. |

**Os Operadores.**

**Aritméticos.**

|  |  |
| --- | --- |
| + Soma | - Subtração |
| \* Multiplicação | / Divisão |
| \*\* Exponencial | % Módulo(Resto da Divisão) |

**Lógicos**.

|  |  |
| --- | --- |
| && E | | | Ou |

**Comparação inteira.**

|  |  |
| --- | --- |
| -eq | Igual |
| - ne | Diferente |
| -gt | Maior que |
| -ge | Maior ou igual a |
| -lt | Menor que |
| -le | Menor ou igual a |
| -o | Ou |
| -d | Se for um diretório |
| -e | Se existir |
| -z | Se estiver vazio |

**Comparação de strings.**

|  |  |
| --- | --- |
| = | Igual |
| != | Diferente |
| < | Menor que |
| > | Maior que |
| -z | String nula, seja tamanho = 0 |
| -n | String não é nula |

**Arquivos**.

|  |  |
| --- | --- |
| -e | Verdade se o arquivo existe. |
| -f | Verdade se é um arquivo não diretório. |
| -d | Verdade se é um diretório não um arquivo. |

**Outros**.

|  |  |
| --- | --- |
| sleep | Pausa em segundos |
| read | Captura os dados digitados pelo usuário |
| exit | Finaliza o script |
| echo | Imprime em tela |
| clear | Limpa a tela |

***IF – Controle de fluxo.***

Controle de fluxo são comandos que vão testando algumas alternativas, e de acordo com essas alternativas, vão executando comandos. Um dos comandos de controle de fluxo mais usados é certamente o if, que é baseado na lógica “se acontecer isso, irei fazer isso, se não, irei fazer aquilo”.

Exemplo:

if [ condição1 ]; then  
 se verdadeiro  
elif [condicao2 ]; then  
 se verdadeiro  
else   
 se falso  
fi  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

***FOR***

O laço for vai substituindo uma variável por um valor, e vai executando os comandos pedidos.

Exemplo:

for x in a b c  
do  
 echo $x  
done  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

***While***

O while testa continuamente uma expressão, até que ela se torne falsa.

Exemplo:

variável="valor"  
while [ $variável = "valor" ]; do  
 comando1  
 comando2  
done

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Case:**

O case é para controle de fluxo, tal como é o if. Mas enquanto o if testa expressões não exatas, o case vai agir de acordo com os resultados exatos.

Exemplo:

case $1 in  
 parametro1) comando1 ; comando2 ;;  
 parametro2) comando3 ; comando4 ;;  
 \*) echo "Você tem de entrar com um parâmetro válido" ;;  
esac

**Exemplos diversos**

**Conta ate 5**

valor=1   
while [ $valor -le 5 ]   
do   
 echo $valor   
 let valor++  
done   
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Mostra arquivo ou compacta**

dir=`pwd`  
  
Listar(){  
clear  
ls -lS  
echo ""  
Menu  
}  
  
Backup(){  
clear  
tar -vzcf teste.tar.gz \*  
echo ""  
Menu  
}  
  
Menu(){  
 clear  
 opcao=1  
 while [ $opcao -ne 0 ]   
 do  
 echo "1. Backup dos ficheiros"   
 echo "2. Listagem da pasta $dir"   
 echo "0. Sair"   
 echo   
 echo -n "Introduza a sua escolha "   
 read opcao  
 case $opcao in  
 1) Backup ;;  
 2) Listar ;;  
 0) exit ;;  
 \*) echo "Opção desconhecida";Menu;  
 esac  
 done  
}  
Menu  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Corrige xml (Fernando)**

dir=/home/sg/xml  
arq=`ls $dir`  
tmp=tmp.xml  
  
lista(){  
cd $dir  
for a in $arq  
do  
 ins=`cat $a|grep DigestValue|cut -d'>' -f2|cut -d'<' -f1`  
 linha=`cat -n $a|grep /digVal|cut -c 4-6`  
 cat $a|grep -v /digVal >$tmp && mv $tmp $a  
 echo "sed -i $linha'i\ <digVal>$ins</digVal>\' $a" >exec  
 chmod +x exec && ./exec && rm exec  
done  
}  
lista  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Acesso FTP**

dest='ftp.example.com'  
nome='usuario'  
passw='senha'  
arq='file.txt'  
  
ftp -n $dest <<EOF  
quote USER $nome  
quote PASS $passw  
cd dir1  
put $arq  
quit  
EOF  
exit 0  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Verifica montagem**

if ! mount | grep 192.168.1.254/teste >>/dev/null  
then  
echo "Não localizado"  
else  
echo "OK."  
fi  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Rsync**

# Adicionando ip na /etc/hosts  
if ! grep -q 177.72.160.134 /etc/hosts  
then  
echo "Atualizando /etc/hosts."  
echo 177.72.160.134 177.72.160.134.static.getcard.net.br >>/etc/hosts  
else  
echo "Arquivo já atualizado."  
fi  
  
# Parametros  
pwd=`pwd`  
data=`date`  
dia=`date +%d%m`  
orig=/sg/sgs/ser/  
conexao=dados@ds.sgsistemas.com.br #dados@177.72.160.134  
pass=dadossg  
port=2229  
dest=:/mnt/dados/paraiso  
#  
lista=(  
tabpdv.dbf   
cadpro.dbf   
vendia??.dbf   
)  
echo $data >>/root/scripts/sincronizou  
cd /sg/sgs/ser/  
./pe\_vendas.l  
cd $pwd  
echo "# Lista " >copia.sh  
for arq in "${lista[@]}"  
do  
done  
echo sshpass -p $pass rsync -Cravzp -e "'"ssh -p $port"'" $orig$arq $conexao$dest >>copia.sh  
   
chmod +x copia.sh && ./copia.sh && rm copia.sh  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Backup Banco**

function naofaz() {  
server=$1  
 echo "O backup do banco" $DB "do servidor" $server "Nao sera feito";  
}  
  
function faz() {  
 ###Operacao de Backup  
 echo "Fazendo backup do banco" $DB "do servidor" $server;  
 mysqldump --single-transaction --extended-insert -R -n -q -h $server -P$porta \  
 -u $usuario\_backup -p"$senha\_backup" $DB | gzip -9 > $FILENAME;  
}   
  
function listarbancos() {  
server=$1  
porta=$2  
  
###Listar MySQL databases e fazer backup  
DB\_LIST=`mysql -h $server -u $usuario\_backup -p"$senha\_backup" -P$porta -e'show databases;'`  
DB\_LIST=${DB\_LIST##Database}  
echo "  
Caminho da backup $caminho$server  
Fazendo backup do servidor $server  
Bancos de dados localizados no servidor:$DB\_LIST  
Iniciando backup aguarde !!!"  
###Verificar se existe o diretorio do backup, acessa o diretorio do backup  
 if [ -d $caminho$server ]; then  
 echo "Diretorio $caminho$server ok!"  
 else   
 mkdir -p $caminho$server  
 fi  
cd $caminho$server  
cd ""  
  
###Backup do banco de dados.  
for DB in $DB\_LIST;  
do  
 FILENAME=$caminho$server/${DB}-$data.sql.gz  
  
 case "$DB" in  
 information\_schema)naofaz $server ;;  
 mysql)naofaz $server ;;  
 testesgps)naofaz $server ;;  
 performance\_schema)naofaz $server ;;  
 \*)faz "$server" "$porta" ;;  
 esac  
done  
echo "----------------------------------------------------------------------  
  
Fazendo limpeza na pasta $caminho  
Aguarde !!!"  
}  
  
function permissao\_apagar() {  
 ###Verifica se a pasta existe ou o HD esta montado e  
 ###apaga os arquivos com data superior a prazo\_excluir  
 ###Nao remover  
 if [ -e $caminho ]; then  
 find $caminho -mtime $prazo\_excluir -exec rm -f {} \;  
 fi  
 chown -fR sgbackup.sgbackup $caminho  
}  
  
###Permissao para usuario de backup  
###GRANT SHOW DATABASES, SELECT, SHOW VIEW, LOCK TABLES, RELOAD ON \*.\* to backup@'192.168.1.251'identified by "senha";  
  
usuario\_backup="backup"  
senha\_backup="senha"  
data=`date +%d-%h-%Y-%H%M`  
caminho="/mnt/backup/copia/"  
prazo\_excluir=+10  
  
###Servidores Que contem os bancos  
servidor\_sgps="192.168.1.251"  
servidor\_trac="192.168.1.248"  
servidor\_ftp="192.168.1.247"  
servidor\_gbs="192.168.1.238"  
  
###Portas do MySQL  
porta\_default="3306"  
  
clear  
date  
umask 0007  
###Chamada das funcoes de backup  
listarbancos "$servidor\_sgps" "$porta\_default"  
listarbancos "$servidor\_trac" "$porta\_default"  
listarbancos "$servidor\_gbs" "$porta\_default"  
  
permissao\_apagar   
date  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Instalador SG Sistemas.**

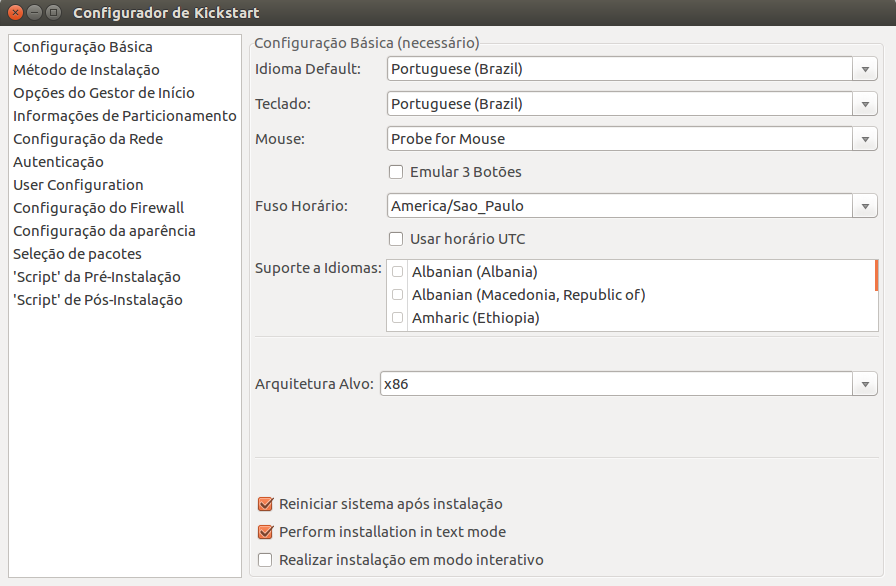
A ideia é que o usuário tenha o mínimo de interação possível.  
De acordo com a versão do Linux temos que verificar como fazer a instalação automática do sistema.

Vamos dar uma olhada no padrão que é utilizado na SG .

**Kickstart** – Padrão Red Hat, pode ser usando em todos os derivados (centos/fedora).  
<https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/6/html/Installation_Guide/s1-kickstart2-options.html>

Existe documentação, que ele funcione com a versão do Ubuntu (mas nunca testei).  
<https://help.ubuntu.com/16.04/installation-guide/i386/ch04s06.html>  
  
Para edição do arquivo podemos fazer manual ou utilizar o aplicativo **system-config-kickstart** o qual e via “X” e pode facilitar bastante o trabalho.

https://www.youtube.com/watch?v=ZknDnt3ml5Y



O arquivo final fica assim:  
ks.cfg

*#Generated by Kickstart Configurator*

*#platform=x86*

*#Idioma do sistema*

*lang pt\_BR*

*#Language modules to install*

*langsupport pt\_BR*

*#Idioma de teclado*

*keyboard br*

*#System mouse*

*mouse*

*#Fuso horario*

*timezone America/Sao\_Paulo*

*#Senha de Root*

*rootpw --iscrypted $1$ojFwHnfI$sWRn8fcT4SxO7vQ6rPf/.0*

*#Usuario padrao*

*user sg --fullname "SGSistemas" --iscrypted --password $1$CWGZuvLO$ah58WivFBFBa.7WOuosGP1*

*#Reboot apos instalacao*

*reboot*

*#Use text mode install (text/grafical*

*text*

*#Instalação ou atualização*

*install*

*#Midia de instalação*

*cdrom*

*#Local bravação boot*

*bootloader --location=mbr*

*#LImpa MBR*

*zerombr yes*

*#Limpeza do disco*

*clearpart --all --initlabel*

*#Partições para criar no disco*

*part swap --recommended*

*part /boot --fstype ext4 --size 250*

*part / --fstype ext4 --size 20000*

*part /home --fstype ext4 --size 1 --grow --maxsize 1*

*#System authorization infomation*

*auth --useshadow --enablemd5*

*#Tipo de rede padrão (DHCP)*

*network --bootproto=dhcp --device=eth0*

*#Firewall*

*firewall --disabled*

*#Configuração padrão do “X”*

*xconfig --depth=32 --resolution=800x600 --defaultdesktop=GNOME*

*#Pacotes para instalação*

*%packages*

*# grupo de pacote*

*@ kubuntu-desktop*

*# Aplicativo individual*

*samba  
# Remove grupo de pacotes  
-@ office-suite*

*# Remove aplicativo individual*

*- mc*

*%pre*

*# Script pre install*

*#################*

*%post --nochroot*

*# Script*useradd -m toor

# Define senha  
passwd toor<<fim

quote senha1

quote senha1

fim

**Como foi feito o instalador do centos 6**

O Centos tem uma particularidade que é a utilização do Anaconda, o Anaconda e um assistente de instalação.  
o Anaconda utiliza um arquivo de pré configurado. Chamado ks.cfg (o nome pode mudar desde que tenha a extensão cfg)

Inicialmente vamos começar a fazendo uma instalação limpa do Linux (instalação básica padrão) em uma VM.

Quando finalizado a instalação o Anaconda deixa um arquivo salvo na pasta do root de nome anaconda-ks.cfg, vamos utilizar esse arquivo afinal temos quase tudo que precisamos nele.

Lembre-se que tudo que tem “#” é apenas comentário.

Além desse arquivo precisamos fazer algumas alterações dentro de alguns arquivos do cd original.

**/isolinux/isolinux.cfg**

*default vesamenu.c32*

*#prompt 1*

*# tempo para iniciar automaticamente (/10=seg )*

*timeout 300*

*# Vermelho #ffff0000 #ff000000*

*# Azul #FF0000FF*

*# Branco #ff000000 #ffffffff*

*# Cinza #696969*

*# Preto #ffffffff #00000000*

*display boot.msg*

*# Imagem de fundo (SG Sistemas, ver tamanho padrão da imagem)*

*menu background splash.jpg*

*menu title Instalacao do SGVX-SGNfe CentOS 6.6 v1.5 2016*

*# Titulo*

*menu color title 0 #696969*

*# linha borda*

*#menu color border 0 #ff000000 #ffffffff*

*# Texto selecionado*

*menu color sel 7 #FF0000FF #ffffffff*

*# texto sem selecao*

*menu color unsel 0 #ffffffff #00000000*

*# Descricao da Opcao (descritivo)*

*menu color tabmsg 0 #ffffffff #00000000*

*# Letra de atalho selecionada*

*menu color hotsel 0 #ff000000 #ffffffff*

*# Letra de Atalho sem selecao*

*menu color hotkey 7 #ffffffff #ff000000*

*#menu color scrollbar 0 #ffffffff #00000000*

*label linux*

*# O “^” e a tecla de atalho, nesse caso seria o “I”*

*menu label ^Instalar Pdv Linux*

*kernel vmlinuz*

*# ks=cdrom:/ks.cfg e o arquivo de script da instalação*

*append initrd=initrd.img ks=cdrom:/ks.cfg*

*label server*

*menu label Instalar ^Servidor Sgvx*

*menu Servidor Sgvx*

*kernel vmlinuz*

*append initrd=initrd.img ks=cdrom:/sg/server/ks.cfg text*

*label memtest86*

*menu label ^Memory test*

*kernel memtest*

*append -*

*label hd*

*menu label Iniciar a partir do ^Disco rigido*

*localboot 0x80*

*append –*

*# menu default (onde estiver essa opção, será a padrão caso o usuário não selecione nenhuma)*

*menu default*

**/isolinux/splash.jpg**

Imagem para inicialização do CD.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**/ks.cfg**

*#version=DEVEL*

*# Modo de instalação (install / update)*

*install*

*# Origem da instalação*

*cdrom*

*# Idioma do sistema*

*lang pt\_BR.UTF-8*

*# Teclado*

*keyboard br-abnt2*

*# Placa de rede*

*network --onboot yes --device eth0 --bootproto dhcp --hostname Pdv*

*# onboot yes (subir a placa junto do sistema)  
# device (dispositivo padrão)*

*# bootproto(dhcp ou static)*

*# hostname (nome da maquina)*

*# Senha do usuário root (criptografada)*

*rootpw --iscrypted $6$reZqGhUYuNyySCtI$C.CsiMfprLu6qcO/puiQL1aAB7oP0U2QYxJ7TuHsQKPeok/lC5oSCnrce93ob5HXhSLjQNDeM3INcDoQ0OHao.*

*# firewall (disabled / enabled)*

*firewall –disabled  
# arquivo shadow para armazenar senhas, criptografia de 512*

*authconfig --enableshadow --passalgo=sha512*

*# selinux (disabled / enabled)*

*selinux --disabled*

*# fuso horário padrão*

*timezone --utc America/Sao\_Paulo*

*# local para gravar o boot*

*# bootloader --location=mbr --driveorder=sda --append="crashkernel=auto rhgb quiet"*

*bootloader --location=mbr*

*# limpa a setor de mbr do hd*

*zerombr*

*# limpa todas as partições*

*clearpart --all*

*# Sistema de partições no HD*

*# utiliza todo o espaço restante para criar o “/”*

*part / --fstype="ext4" --grow --size=1*

*# criar uma artição “swap” tamanho padrão (de acordo com o tamanho do hd)*

*part swap --fstype="swap" --recommended*

*# local para pegar os arquivos de instalação*

*repo --name="CentOS" --baseurl=file:///mnt/source --cost=100*

*# Pacotes para serem instalado no computador.*

*%packages*

*@base*

*@brazilian-support*

*@core*

*@debugging*

*@basic-desktop*

*@desktop-debugging*

*@desktop-platform*

*@directory-client*

*@fonts*

*@general-desktop*

*@graphical-admin-tools*

*@input-methods*

*@internet-browser*

*@legacy-x*

*@network-file-system-client*

*@portuguese-support*

*@print-client*

*@server-platform*

*@server-policy*

*@workstation-policy*

*@x11*

*#@internet-applications (Comentado não faz a instalação)*

*samba*

*mtools*

*pax*

*oddjob*

*wodim*

*sgpio*

*#genisoimage*

*device-mapper-persistent-data*

*abrt-gui*

*samba-winbind*

*certmonger*

*libXmu*

*%end*

*# script pós-instalação (continuação do script)*

*%post*

*# montando CD.*

*# faz a troca da tela para a tty3, assim podemos ver o que está sendo executado.*

*exec < /dev/tty3 > /dev/tty3*

*chvt 3*

*clear*

*echo " ########################################"*

*echo " # Iniciando instalacao PDV SG Sistemas #"*

*echo " # #"*

*echo " # Aguarde #"*

*echo " # #"*

*echo " # vai demorar de 2 a 5 minutos. #"*

*echo " ########################################"*

*# arquivos necessários para funcionamento do pdv e servidor da SG*

*mkdir -p /media/*

*mkdir -p /sg/install/*

*mount /dev/cdrom /media*

*cp -fRv /media/sg/\* /sg/install/*

*chmod +x /sg/install/install\_pdv.sh*

*cd /sg/install/*

*# inicio da instalação dos arquivos necessarios*

*./install\_pdv.sh*

*# desmontar a unidade de cdrom*

*umount /dev/cdrom*

*volta a tela da instalação*

*chvt 1*

*# poderia ter um reboot*

*%end*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Install\_pdv.sh**

*# Parametros*

*cd /sg/install*

*DIR=`pwd` #/sg/install*

*# Pacotes RPM necessário para funcionamento.*

*echo "Descompactando Pacotes"*

*sleep 5*

*rpm -ivh $DIR/rpm/\*.rpm --force --nodeps*

*# Verificando se tem a pasta do PDV.*

*if [ -f /mnt/pdv/parpdv.dbf ];*

*then*

*echo "ja tem pdv instalado";*

*exit*

*fi*

*# Criando Usuario*

*echo "Criando Usuarios"*

*sleep 2*

*adduser sg*

*adduser 123*

*adduser toor*

*passwd -d 123*

*passwd -d sg*

*passwd -d root*

*passwd -d toor*

*# Criando Diretorios.*

*echo "Criando Diretorios"*

*sleep 2*

*mkdir -p /sg/sgs/ser/*

*mkdir -p /sg/doc/xls*

*mkdir -p /sg/doc/doc*

*mkdir -p /sg/doc/txt*

*mkdir -p /sg/doc/pdf*

*mkdir -p /sg/doc/prns*

*mkdir -p /sg/sgs/impress/*

*mkdir -p /tmp/remoto/*

*mkdir -p /mnt/pdv/*

*mkdir -p /mnt/sgs/*

*mkdir -p /root/.gconf/desktop/gnome/remote\_access/*

*chmod 777 /tmp -Rf*

*# Copia de Arquivos de Configuracao.*

*echo "Copiando arquivos de configuracao."*

*sleep 2*

*cp -fRv $DIR/conf/.bashrc\_pdv /home/123/.bashrc*

*cp -fRv $DIR/conf/crontab /etc*

*cp -fRv $DIR/conf/custom.conf /etc*

*cp -fRv $DIR/conf/inittab /etc*

*cp -fRv $DIR/conf/cupsd.conf /etc/cups*

*cp -fRv $DIR/conf/pdv.sh /root/*

*cp -fRv $DIR/conf/inst\_nfce /root/*

*cp -fRv $DIR/conf/splash.xpm.gz /boot/grub/*

*cp -fRv $DIR/conf/sshd\_config\_pdv /etc/ssh/sshd\_config*

*cp -fRv $DIR/conf/sudoers /etc/*

*cp -fRv $DIR/conf/suporte /usr/bin/*

*cp -fRv $DIR/conf/tela /boot/grub/*

*cp -fRv $DIR/conf/testeserial /usr/bin/*

*cp -fRv $DIR/conf/tty.conf\_user\_root /etc/init/tty.conf*

*cp -fRv $DIR/conf/video.sh /root/*

*cp -fRv $DIR/conf/rede\_pdv.sh /root/*

*cp -fRv $DIR/conf/%gconf.xml /root/.gconf/desktop/gnome/remote\_access/*

*cp -fRv $DIR/server/comunica /usr/bin/*

*cp -fRv $DIR/server/lpansi /usr/bin/*

*cp -fRv $DIR/server/ser.arj /sg/sgs/ser/*

*cp -fRv $DIR/server/sgd /usr/bin/*

*cp -fRv $DIR/server/sgi /usr/bin/*

*cp -fRv $DIR/server/sgr /usr/bin/*

*cp -fRv $DIR/server/sgs /usr/bin/*

*cp -fRv $DIR/server/smb.conf /etc/samba/*

*cp -fRv $DIR/lib/\* /usr/lib/*

*cp -fRv $DIR/util /sg/sgs/*

*cp -fRv $DIR/util/dbu.l /usr/bin/dbu*

*cp -fRv $DIR/conf/web.sh /usr/bin/srvfoods*

*cp -fRv $DIR/conf/atualiza /bin/*

*cp -fRv $DIR/conf/ifcfg-usb0 /etc/sysconfig/network-scripts*

*cp -fRv $DIR/conf/rede\_reload /bin/*

*# Permissao*

*echo "Permissao de Arquivos"*

*sleep 2*

*chmod +x /boot/grub/tela*

*chmod +x /root/pdv.sh*

*chmod +x /root/inst\_nfce*

*chmod +x /root/video.sh*

*chmod +x /root/rede\_pdv.sh*

*chmod +x /usr/bin/suporte*

*chmod +x /usr/bin/testeserial*

*chmod +x /usr/bin/comunica*

*chmod +x /usr/bin/lpansi*

*chmod +x /usr/bin/sgd*

*chmod +x /usr/bin/sgi*

*chmod +x /usr/bin/sgr*

*chmod +x /usr/bin/sgs*

*chmod +x /usr/bin/dbu*

*chmod +x /usr/bin/srvfoods*

*chmod +x /bin/atualiza*

*chmod +x /bin/rede\_reload*

*chown sg.sg /home/sg -Rf*

*chown 123.123 /home/123 -Rf*

*# Configurando tela inicial do computador*

*echo "Tela de boot"*

*sleep 2*

*/boot/grub/./tela*

*# script de configuração grub*

*cd /boot/grub*

*mv grub.conf grub.conf2*

*sed -r 's\hiddenmenu\#hiddenmenu\' grub.conf2 >grub.conf*

*# feito a alteração para mostra na tela de inicialização o nome SGSistemas!*

*#Instalacao de fontes*

*echo "Instalando fontes"*

*sleep 2*

*rpm -ivh $DIR/rpm/fontes/\* --nodeps --force*

*#echo "Habilitando Servicos na inicializacao do servidor."*

*echo "Servicos na inicializacao"*

*sleep 2*

*# Colocando serviços do linux na inicialização do sistema.*

*chkconfig --level 3 smb on*

*chkconfig --level 3 cups on*

*chkconfig --level 3 crond on*

*chkconfig --level 3 openvpn on*

*# Força a inicialização da placa de rede.*

*echo rede\_reload >>/etc/rc.local*

*clear*

*inicio(){*

*if [ -e /mnt/pdv/retaguarda.sh ];then*

*start\_network*

*else*

*exit*

*fi*

*}*

*start\_network(){*

*IP=`cat /mnt/pdv/retaguarda.sh|grep caminho=|awk -F // '{ print $2 }'|tr -d a-z,A-Z," ",-,/`*

*ping -q -c1 $IP > /dev/null*

*if [ $? -eq 0 ]*

*then*

*echo "Rede on"*

*exit*

*else*

*echo "Rede off"*

*ping -q -c1 www.sgsistemas.com.br > /dev/null*

*if [ $? -eq 0 ]*

*then*

*echo "Rede on"*

*exit*

*else*

*echo "Rede off"*

*fi*

*echo "Reiniciando servico de rede/ssh"*

*sudo service network restart*

*sudo service sshd restart*

*exit*

*fi*

*}*

*inicio*

*echo "Repositorios"*

*sleep 2*

*# inclui mais um repositório no linux*

*sed -i "s/mirrorlist=https/mirrorlist=http/" /etc/yum.repos.d/epel.repo*

*Configuracao do boot inicial*

*echo "*

*./pdv.sh*

*./inst\_nfce*

*./video.sh*

*./rede\_pdv.sh" >>/root/.bashrc*

*echo "FIM"*

*sleep 2*

Obs: O usuário root esta sem senha, devido a uma troca do arquivos passwd, configurado o Linux para iniciar com o usuário root na inicialização.

Temos 3 arquivos configurados para executar na inicialização do sistema, para assim finalizar a instalação.

Esse é todo o processo inicial da instalação do pdv (processo inicial, lembre-se que ainda tem todo o processo de personalização do Linux que ainda tem de ser feito).

Dentro da ISO atual temos vários arquivos na pasta sg/conf, onde podemos usar como exemplo para a nova ISO, mas lembre-se que cada versão trabalha de um jeito (isso dentro da mesma distro), agora mudando a distro ..... Vá por sua conta e risco....

Lembre-se que você vai precisar de um bom editor de ISO (winiso, winrar, ultraiso, 7zip) ou pode se usar no Ubuntu o ISO Master (excelente)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

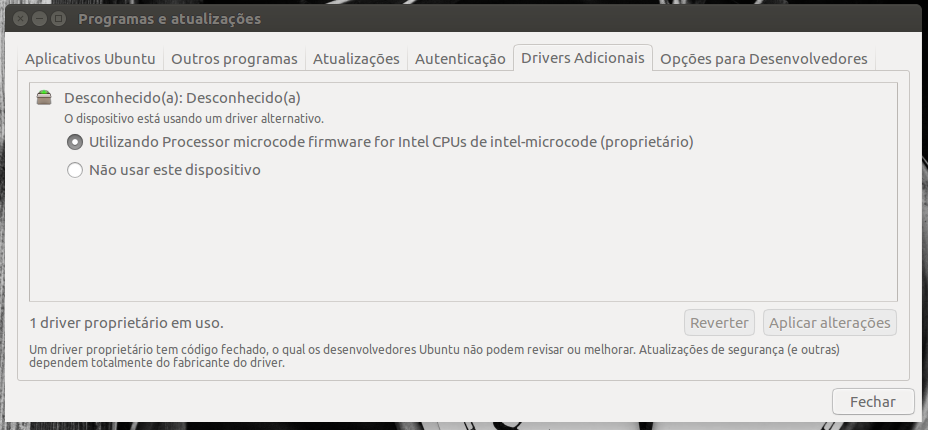
**Para Ubuntu ver :**

Como montar um auto install -> <https://www.youtube.com/watch?v=uxhVqGdW9P8>  
Arquivos para o auto install -> <http://pastebin.com/JHy5Kqui>  
Install via script (não testei) -> <https://github.com/netson/ubuntu-unattended>

**Como fazer a instalação de placas no Linux.**

No Ubuntu fica fácil, pois a instalação e toda no ambiente “X”.





Ele já mostra qual o hardware, o Ubuntu tem a grande vantagem de usar todos os drivers possíveis e não somente os de código aberto.

**Drivers de multi-serial – instalação no bash.**

Para instalação de driver no bash, normalmente todos os driver vem com um arquivo de instalação (README).

Obs: para fazer a compilação do driver podemos ter que instalar alguns pacotes a mais:  
glibc\*, gcc, cpp\*, libgomp\*, make\*

Vamos ver um exemplo.

Run the following command in the driver source folder:

$ make

To install the driver use the following command:

$ make install

To load the driver use the following command:

$ insmod 99xx.ko

This driver is currently developed and tested on 2.6.13 linux kernel and above

**$** -> pasta onde esta os drivers descompactado.  
**make** -> Compilação dos drivers com o kernel da maquina.  
**make install** -> Instalação dos drivers na maquina.  
**insmod 99xx.ko** -> Comando para carregar os driver copilados (colocar no /etc/rc.local).  
**Drive feito e testado para kernel superior a 2.6.13, para ver versão do kernal, usar o comando “uname –r”**

*Driver chiset wch351 Driver chiset wch382*

*Driver chiset msc9835 Driver chiset msc9865 Driver chiset msc9901*

*Driver chiset msc9904 Driver chiset msc7840*

**Download**

<ftp://sgps.sgsistemas.com.br/util/driver-multi-serial/drivers>

<http://www.flexport.com.br/Drivers/Index.html>

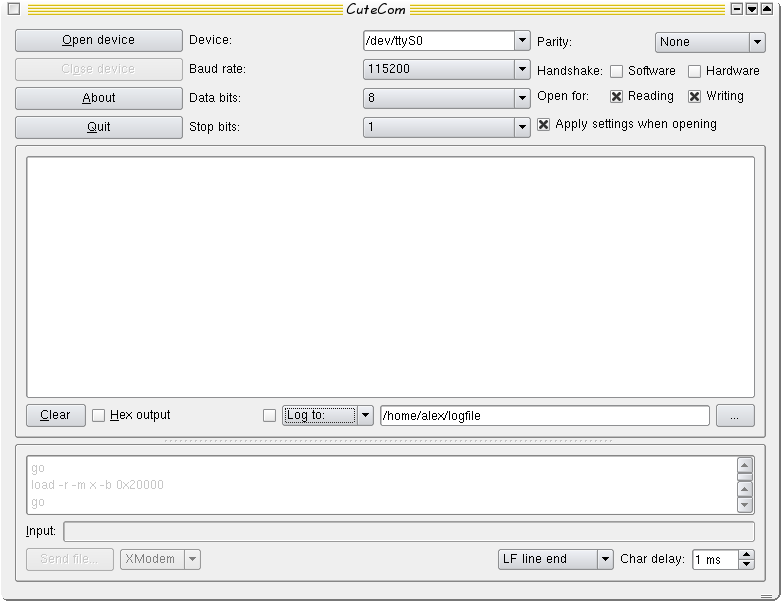
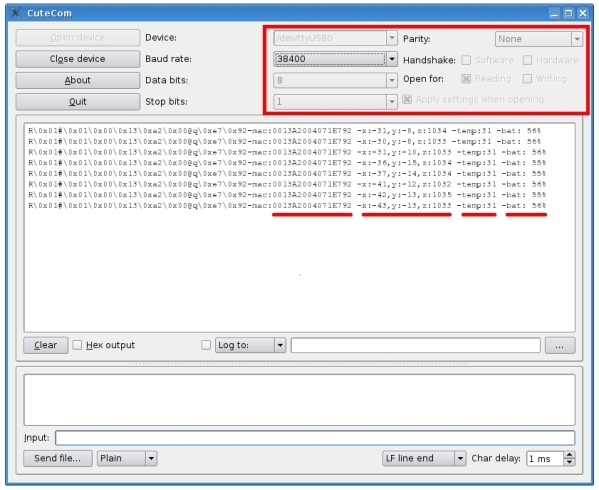
Manual para instalação de placa de vídeo nvidia no centos  
<http://www.dedoimedo.com/computers/centos-7-nvidia.html>

Download driver   
<http://www.nvidia.com/Download/index.aspx?lang=en-us>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**Cutecom (Teste de portas seriais)**instalação :

Centos – yum install cutecom (download [ftp.sgsistemas.com.br/util/cutecom-0.22.0-1.puias6.i686.rpm](ftp://ftp.sgsistemas.com.br/util/cutecom-0.22.0-1.puias6.i686.rpm))  
Ubuntu – apt install cutecom

Como na imagem, basta selecionar as opções e testar.  
Marcar opção -> open for: Reading Writing  
ex:

|  |  |
| --- | --- |
| Padrão de balança toledo | **Padrão Scanner mesa** |
| device: = porta serial Baud rate: = 2400 data bits: = 8 stop bits: = 1 parity: = none | **device: = porta serial Baud rate: = 9600 data bits: = 8 stop bits: = 1 parity: = none** |

Fail2ban

Sshd

Resolv.conf

Host.conf

Hostname

Hosts

Issue

Iptables