

SDN Overlay - Whiteboxes

Instalação:

Iniciar servidor Whitebox e entrar no Setup:

Carregar defaults: F3

Configurar religação automática após queda de energia (ver na plataforma)

Configuração IP estático para IPMI (ver na plataforma):

IP: 192.168.0.204

GW: 192.168.0.30

Iniciar imagem do Fedora 27 no pendrive:

Posicionar o foco sobre a instalação padrão e apertar Tab.

Retirar da linha de parametros o "quiet" e adicionar, alterando o arquivo de config de acordo com a UF (mg.cfg para MG, rj.cfg para RJm etc):

inst.ks=<ftp://infra:i@192.168.0.50/mg.cfg>

Acabando a instalação, verificar sincronização de RAID:

```
mdadm /dev/md0 --details
mdadm /dev/md0
mdadm --detail /dev/md0
mdadm --detail /dev/md1
mdadm --detail /dev/md2
mdadm --detail /dev/md3
```

Atualizacao e instalacao de pacotes especificos

```
echo "===== FASE 0 ====="
```

```
echo "    INSTALACAO e ATUALIZACAO DE PACOTES"
```

```
echo "====="
```

```
dnf -y upgrade
```

```
dnf -y install fail2ban
```

```
dnf -y groupinstall "Development Tools"
```

```
echo "===== FASE 1 ====="
```

```
echo "    CUSTOMIZACAO / CONFIGURACAO"
```

```
echo "====="
```

```
dnf -y install dnf-automatic
```

```
cp -p /etc/dnf/automatic.conf /etc/dnf/automatic.conf-ORIGINAL
```

```
sed -i 's/upgrade_type = default/upgrade_type = security/' /etc/dnf/automatic.conf
```

```
systemctl enable dnf-automatic-install.timer && systemctl start dnf-automatic-install.timer
```

```
for IF in `ls /proc/sys/net/ipv6/conf`
```

```
do
```

```
    echo 0 > /proc/sys/net/ipv6/conf/$IF/autoconf
```

```
done
```

```
sed -ri 's,^(SELINUX=).*,\1disabled;' /etc/selinux/config
```

```
echo "IRQBALANCE_BANNED_CPUS=FC" >> /etc/sysconfig/irqbalance
```

```
systemctl enable irqbalance && systemctl start irqbalance
```

```
dnf -y install tuned
```

```
mkdir /etc/tuned/whitebox
```

```
cat << EOF > /etc/tuned/whitebox/tuned.conf
```

```
[main]
```

```
include=network-latency
```

```
[bootloader]
```

```
#cmdline = audit=0 selinux=0 isolcpus=2-7 nosoftlockup mce=ignore_ce idle=poll nohz_full=2-7 hugepages=2048
```

```
cmdline = audit=0 selinux=0 isolcpus=2-7 nosoftlockup mce=ignore_ce idle=poll nohz_full=2-7 hugepages=2048 intel_iommu=on iommu=pt
```

```
[sysctl]
```

```
kernel.nmi_watchdog=0
```

```
vm.stat_interval=10
```

```
vm.nr_hugepages=2048
```

```
[sysfs]
```

```
/sys/bus/workqueue/devices/writeback/cpumask = FC
```

```
/sys/devices/virtual/workqueue/cpumask = FC
```

```
EOF
```

```
tuned-adm profile whitebox
```

```
systemctl enable tuned && systemctl start tuned
```

```
echo "===== FASE 2 ====="
```

```
echo "          INSTALACAO DPDK e OVS"
```

```
echo "====="
```

```
dnf -y install dpdk driverctl
```

```
modprobe uio_pci_generic
```

```
#### No GEN2 (07,08 e 0B)
```

```
driverctl set-override 0000:08:00.0 uio_pci_generic
```

```
driverctl set-override 0000:0b:00.0 uio_pci_generic
```

```
driverctl set-override 0000:0b:00.1 uio_pci_generic
```

```
driverctl set-override 0000:0b:00.2 uio_pci_generic
```

```
driverctl set-override 0000:0b:00.3 uio_pci_generic
```

```
#### No GEN1 (04,05 e 14)
```

```
####driverctl set-override 0000:00:14.0 uio_pci_generic
```

```
####driverctl set-override 0000:00:14.1 uio_pci_generic
```

```
####driverctl set-override 0000:00:14.2 uio_pci_generic
```

```
####driverctl set-override 0000:00:14.3 uio_pci_generic
```

```
dnf download --source openvswitch
```

```
dnf download --refresh --releasever=rawhide openvswitch
```

```
ovs_src_rpm=`ls openvswitch*.rpm`
```

```
echo "***** instalando $ovs_src_rpm"
```

```
rpm -i $ovs_src_rpm
```

```
# Install OVS dependencies
```

```
dnf -y install dnf-plugins-core rpm-build libpcap-devel numactl-devel
```

```
dnf -y builddep ~/rpmbuild/SPECS/openvswitch.spec
```

```
# Build OVS package with DPDK support
```

```
rpmbuild -bb ~/rpmbuild/SPECS/openvswitch.spec --with dpdk --without check --define='optflags -Ofast -g -pipe -Wall -Werror=format-security -Wp,-D_FORTIFY_SOURCE=2 -fexceptions -fstack-protector-strong --param=ssp-buffer-size=4 -grecord-gcc-switches -specs=/usr/lib/rpm/redhat/redhat-hardened-cc1 -m64 -mtune=generic -march=native -fomit-frame-pointer -mssse3 -msse4.2'
```

```
# Install OVS package
```

```
ovs_rpm=`ls ~/rpmbuild/RPMS/x86_64/openvswitch*.rpm`
```

```
echo "***** instalando $ovs_rpm"
```

```
rpm -i $ovs_rpm
```

```
systemctl start openvswitch
```

```
ovs-vsctl --no-wait set Open_vSwitch . other_config:dpdk-init=true
```

```
ovs-vsctl --no-wait set Open_vSwitch . other_config:dpdk-socket-mem="4096,0"
```

```
ovs-vsctl --no-wait set Open_vSwitch . other_config:pmd-cpu-mask=FC
```

```
systemctl restart openvswitch
```

```
ovs-vsctl add-br br-oper0 -- set bridge br-oper0 datapath_type=netdev fail-mode=secure
```

```
ovs-vsctl add-port br-oper0 dpdk-p0 -- set Interface dpdk-p0 type=dpdk mtu_request=9600 ofport=1 options:dpdk-devargs=0000:08:00:0
```

```
ovs-vsctl add-port br-oper0 dpdk-p1 -- set Interface dpdk-p1 type=dpdk mtu_request=9600 ofport=2 options:dpdk-devargs=0000:0b:00:0
```

```
###ovs-vsctl set-controller br-oper0 tcp:10.200.0.14:6653
```

```
###ovs-ofctl add-flow br-dpdk0 in_port=1,action=output:2
```

```
###ovs-ofctl add-flow br-dpdk0 in_port=2,action=output:1
```