1. 安装centos6.5，然后再centos6.5的环境下安装所需软件包。

yum -y install rsync yum-utils createrepo genisoimage isomd5sum squashfs-tools xz-lzma-compat rpm-build cifs-utils

1. 下载centos6.5 minimize版

wget <http://mirror.bit.edu.cn/centos/6.5/isos/x86_64/CentOS-6.5-x86_64-minimal.iso>

如果链接不可用，可从如下网址查询

网址：<http://wiki.centos.org/Manuals/ReleaseNotes/CentOSMinimalCD6.5>

1. 创建目录base，build，rpmbuild。

mkdir {base,build,rpmbuild}

1. 挂在centos6.5 minimize 到base目录，将base所有内容同步到build。

mount -o loop,ro CentOS-6.5-x86\_64-minimal.iso base/

rsync -av base/ build/

1. 删除，TRANS.TBL，然后进入Package目录，并且删除低版本的audit

find build/ -name TRANS.TBL -exec rm -f {} \;

cd build/Packages && rm audit-\*

1. 设置EPEL源，

wget [http://mirrors.yun-idc.com/epel/6/i386/epel-release-6-8.noarch.rpm && rpm -ivh epel-release-6-8.noarch.rpm](http://mirrors.yun-idc.com/epel/6/i386/epel-release-6-8.noarch.rpm%20&&%20rpm%20-ivh%20epel-release-6-8.noarch.rpm)

rpm -ivh epel-release-6-8.noarch.rpm

1. 进入Package目录，下载网关依赖的rpm包

yumdownloader audit-2.2-4.el6\_5.x86\_64 audit-libs-2.2-4.el6\_5.x86\_64 audit-libs-python-2.2-4.el6\_5.x86\_64 bc-1.06.95-1.el6.x86\_64 beecrypt-4.2.1-1.el6.x86\_64 cairo-1.8.8-3.1.el6.x86\_64 cifs-utils-4.8.1-19.el6.x86\_64 cluster-glue-1.0.5-6.el6.x86\_64 cluster-glue-libs-1.0.5-6.el6.x86\_64 compat-readline5-5.2-17.1.el6.x86\_64 dejavu-fonts-common-2.30-2.el6.noarch dejavu-lgc-sans-mono-fonts-2.30-2.el6.noarch dejavu-sans-mono-fonts-2.30-2.el6.noarch dmidecode-2.12-5.el6\_5.x86\_64 expect-5.44.1.15-5.el6\_4.x86\_64 fontconfig-2.8.0-3.el6.x86\_64 fontpackages-filesystem-1.41-1.1.el6.noarch freetype-2.3.11-14.el6\_3.1.x86\_64 heartbeat-3.0.4-2.el6.x86\_64 heartbeat-devel-3.0.4-2.el6.x86\_64 heartbeat-libs-3.0.4-2.el6.x86\_64 libcgroup-0.40.rc1-5.el6\_5.1.x86\_64 libpcap-1.4.0-1.20130826git2dbcaa1.el6.x86\_64 libpng-1.2.49-1.el6\_2.x86\_64 libselinux-python-2.0.94-5.3.el6\_4.1.x86\_64 libsemanage-python-2.0.43-4.2.el6.x86\_64 libtalloc-2.0.7-2.el6.x86\_64 libtdb-1.2.10-1.el6.x86\_64 libtevent-0.9.18-3.el6.x86\_64 libthai-0.1.12-3.el6.x86\_64 libtool-ltdl-2.2.6-15.5.el6.x86\_64 libX11-1.5.0-4.el6.x86\_64 libX11-common-1.5.0-4.el6.noarch libXau-1.0.6-4.el6.x86\_64 libxcb-1.8.1-1.el6.x86\_64 libXft-2.3.1-2.el6.x86\_64 libXrender-0.9.7-2.el6.x86\_64 lm\_sensors-3.1.1-17.el6.x86\_64 lm\_sensors-libs-3.1.1-17.el6.x86\_64 mailcap-2.1.31-2.el6.noarch mailx-12.4-7.el6.x86\_64 nc-1.84-22.el6.x86\_64 net-snmp-5.5-49.el6\_5.1.x86\_64 net-snmp-libs-5.5-49.el6\_5.1.x86\_64 net-snmp-utils-5.5-49.el6\_5.1.x86\_64 ntp-4.2.6p5-1.el6.centos.x86\_64 ntpdate-4.2.6p5-1.el6.centos.x86\_64 pango-1.28.1-7.el6\_3.x86\_64 perl-5.10.1-136.el6.x86\_64 perl-Compress-Raw-Zlib-2.021-136.el6.x86\_64 perl-Compress-Zlib-2.021-136.el6.x86\_64 perl-Crypt-SSLeay-0.57-16.el6.x86\_64 perl-HTML-Parser-3.64-2.el6.x86\_64 perl-HTML-Tagset-3.20-4.el6.noarch perl-IO-Compress-Base-2.021-136.el6.x86\_64 perl-IO-Compress-Zlib-2.021-136.el6.x86\_64 perl-libs-5.10.1-136.el6.x86\_64 perl-libwww-perl-5.833-2.el6.noarch perl-Module-Pluggable-3.90-136.el6.x86\_64 perl-Pod-Escapes-1.04-136.el6.x86\_64 perl-Pod-Simple-3.13-136.el6.x86\_64 perl-TimeDate-1.16-11.1.el6.noarch perl-URI-1.40-2.el6.noarch perl-version-0.77-136.el6.x86\_64 pixman-0.26.2-5.1.el6\_5.x86\_64 policycoreutils-python-2.0.83-19.39.el6.x86\_64 PyXML-0.8.4-19.el6.x86\_64 quota-3.17-21.el6\_5.x86\_64 resource-agents-3.9.2-40.el6\_5.7.x86\_64 rrdtool-1.3.8-6.el6.x86\_64 rrdtool-perl-1.3.8-6.el6.x86\_64 ruby-1.8.7.352-13.el6.x86\_64 ruby-libs-1.8.7.352-13.el6.x86\_64 samba-common-3.6.9-168.el6\_5.x86\_64 samba-winbind-3.6.9-168.el6\_5.x86\_64 samba-winbind-clients-3.6.9-168.el6\_5.x86\_64 setools-libs-3.3.7-4.el6.x86\_64 setools-libs-python-3.3.7-4.el6.x86\_64 sysstat-9.0.4-22.el6.x86\_64 tcl-8.5.7-6.el6.x86\_64 tcpdump-4.0.0-3.20090921gitdf3cb4.2.el6.x86\_64 tcp\_wrappers-7.6-57.el6.x86\_64

1. 修改某些特定包kernel-2.6.32-431.el6.x86\_64.rpm

1.下载wget <http://vault.centos.org/6.5/os/Source/SPackages/kernel-2.6.32-431.el6.src.rpm>

2.安装源码

3.进入rpmbuild/SOURCES目录

4.将代码包linux-2.6.18.4.tar.bz2解压缩然后删除压缩包

5.改代码sed -i "s/\(#define PROT\_SOCK.\*\)1024/\180/p" linux-2.6.18.4/include/net/sock.h

6.打包生成压缩包。执行命令tar -cjpf linux-2.6.18.4.tar.bz2 linux-2.6.18.4/

7.删除目录linux-2.6.18.4

8.进入rpmbuild目录，执行rpmbuild -bb --target=`uname -m` SPECS/kernel.spec

9.将生成的kernel-2.6.32-431.el6.x86\_64.rpm拷贝到build/Packages目录，替换原有的文件。

1. 进入repodata目录，添加需要安装的包到xml文件

cd build/repodata

mv \*minimal-x86\_64.xml comps.xml&rm \*.bz2 \*.gz repomd.xml -rf

vi comps.xml,把如下内容加入comps.xml

<packagereq type='mandatory'>audit-libs-python</packagereq>

<packagereq type='mandatory'>libcgroup</packagereq>

<packagereq type='mandatory'>libselinux-python</packagereq>

<packagereq type='mandatory'>libsemanage-python</packagereq>

<packagereq type='mandatory'>setools-libs</packagereq>

<packagereq type='mandatory'>setools-libs-python</packagereq>

<packagereq type='mandatory'>policycoreutils-python</packagereq>

<packagereq type='mandatory'>expect</packagereq>

<packagereq type='mandatory'>compat-readline5</packagereq>

<packagereq type='mandatory'>dejavu-fonts-common</packagereq>

<packagereq type='mandatory'>dejavu-lgc-sans-mono-fonts</packagereq>

<packagereq type='mandatory'>dejavu-sans-mono-fonts</packagereq>

<packagereq type='mandatory'>dmidecode</packagereq>

<packagereq type='mandatory'>expect</packagereq>

<packagereq type='mandatory'>fontpackages-filesystem</packagereq>

<packagereq type='mandatory'>lm\_sensors</packagereq>

<packagereq type='mandatory'>lm\_sensors-libs</packagereq>

<packagereq type='mandatory'>mailcap</packagereq>

<packagereq type='mandatory'>nc</packagereq>

<packagereq type='mandatory'>net-snmp</packagereq>

<packagereq type='mandatory'>net-snmp-libs</packagereq>

<packagereq type='mandatory'>net-snmp-utils</packagereq>

<packagereq type='mandatory'>ntp</packagereq>

<packagereq type='mandatory'>ntpdate</packagereq>

<packagereq type='mandatory'>perl-Compress-Raw-Zlib</packagereq>

<packagereq type='mandatory'>perl-Compress-Zlib</packagereq>

<packagereq type='mandatory'>perl-Crypt-SSLeay</packagereq>

<packagereq type='mandatory'>perl-HTML-Parser</packagereq>

<packagereq type='mandatory'>perl-HTML-Tagset</packagereq>

<packagereq type='mandatory'>perl-IO-Compress-Base</packagereq>

<packagereq type='mandatory'>perl-IO-Compress-Zlib</packagereq>

<packagereq type='mandatory'>perl-libwww-perl</packagereq>

<packagereq type='mandatory'>perl-URI</packagereq>

<packagereq type='mandatory'>rrdtool</packagereq>

<packagereq type='mandatory'>rrdtool-perl</packagereq>

<packagereq type='mandatory'>ruby</packagereq>

<packagereq type='mandatory'>ruby-libs</packagereq>

<packagereq type='mandatory'>sysstat</packagereq>

<packagereq type='mandatory'>perl</packagereq>

<packagereq type='mandatory'>perl-libs</packagereq>

<packagereq type='mandatory'>tcl</packagereq>

<packagereq type='mandatory'>perl-Module-Pluggable</packagereq>

<packagereq type='mandatory'>perl-Pod-Simple</packagereq>

<packagereq type='mandatory'>perl-Pod-Escapes</packagereq>

<packagereq type='mandatory'>perl-version</packagereq>

<packagereq type='mandatory'>libpng</packagereq>

<packagereq type='mandatory'>cairo</packagereq>

<packagereq type='mandatory'>pango</packagereq>

<packagereq type='mandatory'>libX11</packagereq>

<packagereq type='mandatory'>libX11-common</packagereq>

<packagereq type='mandatory'>libXrender</packagereq>

<packagereq type='mandatory'>libXft</packagereq>

<packagereq type='mandatory'>fontconfig</packagereq>

<packagereq type='mandatory'>freetype</packagereq>

<packagereq type='mandatory'>libxcb</packagereq>

<packagereq type='mandatory'>pixman</packagereq>

<packagereq type='mandatory'>libXau</packagereq>

<packagereq type='mandatory'>libthai</packagereq>

<packagereq type='mandatory'>beecrypt</packagereq>

<packagereq type='mandatory'>cifs-utils</packagereq>

<packagereq type='mandatory'>cluster-glue</packagereq>

<packagereq type='mandatory'>cluster-glue-libs</packagereq>

<packagereq type='mandatory'>heartbeat</packagereq>

<packagereq type='mandatory'>heartbeat-devel</packagereq>

<packagereq type='mandatory'>heartbeat-libs</packagereq>

<packagereq type='mandatory'>keyutils</packagereq>

<packagereq type='mandatory'>libevent</packagereq>

<packagereq type='mandatory'>libgssglue</packagereq>

<packagereq type='mandatory'>libnl</packagereq>

<packagereq type='mandatory'>libtalloc</packagereq>

<packagereq type='mandatory'>libtdb</packagereq>

<packagereq type='mandatory'>libtevent</packagereq>

<packagereq type='mandatory'>libtirpc</packagereq>

<packagereq type='mandatory'>libtool-ltdl</packagereq>

<packagereq type='mandatory'>nfs-utils</packagereq>

<packagereq type='mandatory'>nfs-utils-lib</packagereq>

<packagereq type='mandatory'>perl-TimeDate</packagereq>

<packagereq type='mandatory'>PyXML</packagereq>

<packagereq type='mandatory'>quota</packagereq>

<packagereq type='mandatory'>resource-agents</packagereq>

<packagereq type='mandatory'>rpcbind</packagereq>

<packagereq type='mandatory'>samba-common</packagereq>

<packagereq type='mandatory'>samba-winbind</packagereq>

<packagereq type='mandatory'>samba-winbind-clients</packagereq>

<packagereq type='mandatory'>tcp\_wrappers</packagereq>

<packagereq type='mandatory'>mailx</packagereq>

<packagereq type='mandatory'>tcpdump</packagereq>

<packagereq type='mandatory'>bc</packagereq>

<packagereq type='mandatory'>sslgw</packagereq>

<packagereq type='mandatory'>LDAP</packagereq>

<packagereq type='mandatory'>LoadBalance</packagereq>

<packagereq type='mandatory'>sslgw\_web</packagereq>

1. 拷贝CFCA目录到build，从build/CFCA拷贝ks.cfg isolinux.cfg splash.jpg到build/isolinux

cp -rf CFCA/ build/

cp -f build/CFCA/ks.cfg build/CFCA/isolinux.cfg build/CFCA/splash.jpg build/isolinux

1. 进入build/isolinux 目录，修改initrd.img

cd build/isolinux && mv initrd.img initrd.img.lzma

unlzma initrd.img.lzma && mkdir initrd

mv initrd.img initrd&& cd initrd

cpio -div <initrd.img && rm initrd.img

修改版本名称：sed -i 's/CentOS/CFCA SSL Gateway/g' .buildstamp

打包 find .|cpio -c -o >../initrd.img && cd ..

rm initrd -rf&&lzma -z initrd.img

mv initrd.img.lzma initrd.img

1. 进入build/images,修改安装画面，添加bootloader密码，注意sed语句中的16个空格。

cd build/images&&unsquashfs install.img

chmod 755 squashfs-root/usr/share/anaconda/pixmaps/\*.png

cp ../../build/CFCA/\*.png squashfs-root/usr/share/anaconda/pixmaps/

sed -i '/quiet/d' squashfs-root/usr/lib/anaconda/packages.py

sed -i 's/anaconda.\*append("rhgb")/pass/' squashfs-root/usr/lib/anaconda/packages.py

sed -i "/splash.xpm.gz.\*:/a \ f.write('password --md5 \$1\$3XKWm1\$E5UYb6OtA4tVfdkMjzovA1\\\n')" squashfs-root/usr/lib/anaconda/booty/x86.py

rm -rf install.img &&mksquashfs squashfs-root/ install.img

rm -rf squashfs-root/

cd ../../

1. 创建LDAP,LoadBalance,sslgw,sslgw\_web等四个rpm包.

mkdir -p rpmbuild/{ldap,loadbalance,sslgw,sslgw\_web}/usr/local

mkdir release/

mount -t cifs -o username=lychen, password=… //192.168.117.2/Build release/

**#SSLGWWEB**

cp -rf release/SSLGateway/3.1.0.9/SSLGW\_WEB/ rpmbuild/sslgw\_web/usr/local/

cp -rf release/SSLGateway/3.1.0.9/sslgw/openssl rpmbuild/sslgw\_web/usr/local/SSLGW\_WEB/web/script/certificate/openssl

mkdir -p rpmbuild/sslgw\_web/etc/init.d/

cp -rf rpmbuild/sslgw\_web/usr/local/SSLGW\_WEB/web/script/init.d/sslgwWeb rpmbuild/sslgw\_web/etc/init.d/

cp -rf build/CFIST/autorun rpmbuild/sslgw\_web/etc/init.d/

mkdir -p rpmbuild/sslgw\_web/root/

cp -rf build/CFIST/setup.sh rpmbuild/sslgw\_web/root/

cp -rf build/CFIST/blacklist.conf rpmbuild/sslgw\_web/root/

mkdir -p rpmbuild/sslgw\_web/usr/bin

cp -rf build/CFIST/lsb\_release rpmbuild/sslgw\_web/usr/bin/

find release/SSLGateway/3.1.0.9/ -name "\*.pdf" |xargs -i cp {} rpmbuild/sslgw\_web/usr/local/SSLGW\_WEB/web/docs/

**#LDAP**

cp -rf release/SSLGateway/3.1.0.9/LDAP/ rpmbuild/ldap/usr/local/

cp -rf rpmbuild/sslgw\_web/usr/local/SSLGW\_WEB/web/script/init.d/LDAP\_startup.sh rpmbuild/ldap/usr/local/LDAP/startup.sh

mkdir -p rpmbuild/ldap/etc/init.d/

cp -rf rpmbuild/sslgw\_web/usr/local/SSLGW\_WEB/web/script/init.d/socketLdap rpmbuild/ldap/etc/init.d/

**#LOADBALANCE**

cp -rf release/SSLGateway/3.1.0.9/LoadBalance rpmbuild/loadbalance/usr/local/

cp -rf rpmbuild/sslgw\_web/usr/local/SSLGW\_WEB/web/script/init.d/LB\_startup.sh rpmbuild/loadbalance/usr/local/LoadBalance/startup.sh

mkdir -p rpmbuild/loadbalance/etc/init.d/

cp -rf rpmbuild/sslgw\_web/usr/local/SSLGW\_WEB/web/script/init.d/socketLoadBalance rpmbuild/loadbalance/etc/init.d/

**#SSLGW**

cp -rf release/SSLGateway/3.1.0.9/sslgw rpmbuild/sslgw/usr/local/

mkdir -p rpmbuild/sslgw/etc/init.d/

cp -rf rpmbuild/sslgw\_web/usr/local/SSLGW\_WEB/web/script/init.d/sslgw rpmbuild/sslgw/etc/init.d/

**#SPEC**

cp -rf build/CFIST/\*.spec rpmbuild

**#rpmbuild**

[ -e ~/.rpmmacros ] && mv ~/.rpmmacros ~/.rpmmacros.bak

echo "%\_topdir rpmbuild/" >& ~/.rpmmacros

rpmbuild -bb --buildroot=rpmbuild/ldap rpmbuild/ldap.spec

rpmbuild -bb --buildroot=rpmbuild/loadbalance rpmbuild/loadbalance.spec

rpmbuild -bb --buildroot=rpmbuild/sslgw rpmbuild/sslgw.spec

rpmbuild -bb --buildroot=rpmbuild/sslgw\_web rpmbuild/sslgw\_web.spec

rm ~/.rpmmacros

[ -e ~/.rpmmacros.bak ] && mv ~/.rpmmacros.bak ~/.rpmmacros

umount release/

1. 创建base.iso

cp -rf rpmbuild/RPMS/x86\_64/\* build/Packages/

discinfo=$(head -1 build/.discinfo)

createrepo -u "media://$discinfo" -g build/repodata/comps.xml build

mkisofs -R -J -T -r -l -d -joliet-long -allow-multidot -allow-leading-dots -no-bak -o $PWD/cfca.iso -b isolinux/isolinux.bin -c isolinux/boot.cat -no-emul-boot -boot-load-size 4 -boot-info-table build/

implantisomd5 $PWD/cfca.iso