
Openwrt for vonets 虚拟机编译环境搭建

1. VMware Workstation 7.1.3 新建 Linux 虚拟机, 分配 disk 上限为 16G, 内存为 256M。
2. 加载 linux 镜像文件: Fedora-11-i386-DVD.iso, 开始装入 Linux 系统。配置用户密码:
超级用户名: root 密码: 123456
普通用户名: vonets 密码: 123456

3. 虚拟机网卡配置为桥接方式, 连接 internet 网, 在线安装软件编译环境。

1) 在线安装:

```
yum install -y gcc*  
yum install -y ncurses-devel  
yum install -y bison  
yum install -y flex  
yum install -y lzma  
yum install -y xz  
yum install -y openssl*  
yum install -y mpc  
yum install -y perl-ExtUtils-MakeMaker  
yum install -y python  
yum install -y python-devel
```

2) 本地安装 git:

```
#tar -zxvf git-1.8.0.1.tar.gz (在线安装版本太低, 无法适应 openwrt 编译);  
#cd git-1.8.0.1  
#./configure  
#make prefix=/usr/local all  
#make install
```

3) RPM 包安装 xinetd, tftp-server, svn:

```
#rpm -ivh xinetd-2.3.14-10.el5.i386.rpm(安装 xinetd)  
#rpm -ivh tftp-server-0.49-2.el5.centos.i386.rpm(安装 xinetd)  
a. 修改 tftp 配置文件(/etc/xinetd.d/tftp)
```

```
service tftp
{
    socket_type      = dgram
    protocol         = udp
    wait             = yes
    user             = root
    server           = /usr/sbin/in.tftpd
    server_args      = -s /tftpboot -c
    disable          = no
    per_source       = 11
    cps              = 100 2
    flags            = IPv4
}
```

b. 修改 tftp 目录属性: `chmod 777 -R /tftpboot/`

```
#rpm -ivh CollabNetSubversion-client-1.6.9-1.i386.rpm
cd /opt/ConllabNet_Subversion/bin
ln ./svn /usr/local/bin/. （配置全局可用 svn 命令）
```

4) 关闭 linux 系统安全限制

a. 关闭 SELinux:

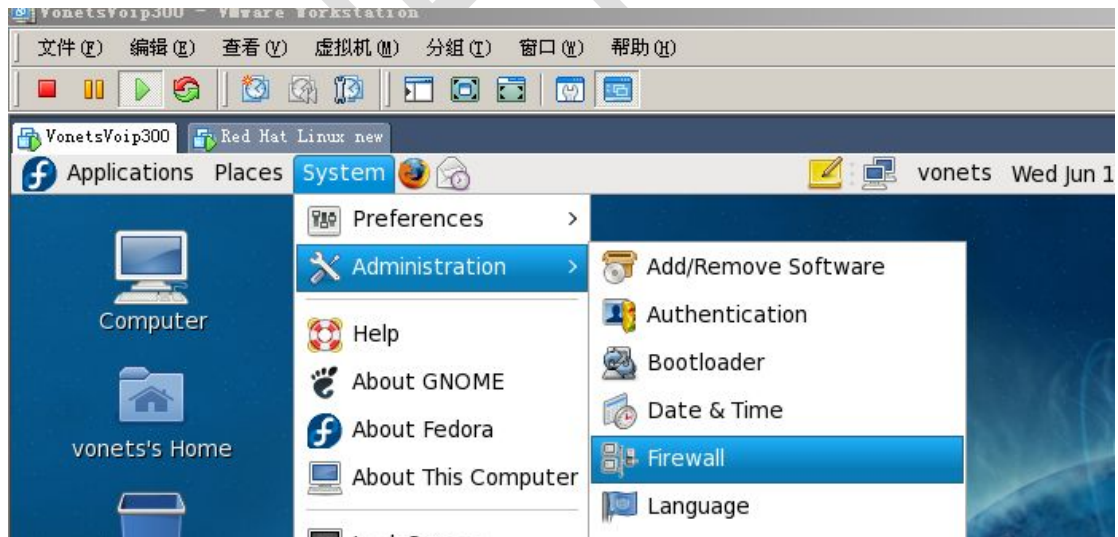
`vi /etc/selinux/config`

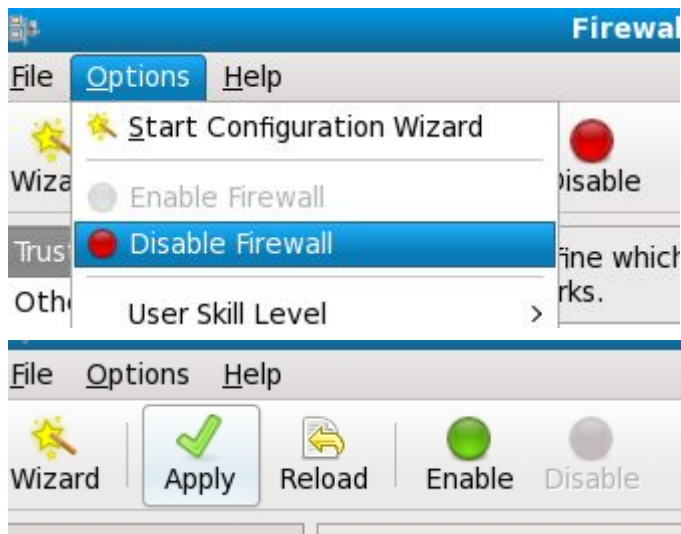
`#SELINUX=enforcing` 注释掉该行

`SELINUX=disabled` 添加该行

B. 永久关闭系统防火墙

界面配置防火墙(命令行模式下运行“init 5”进入 Xwindows 界面)。





c. 重启 tftp 服务: `service xinetd restart`

4. openwrt 工作环境搭建完成，开始主代码编译。

```
#tar -jxvf vonetssdk-openwrt-2.6.36-7621-20140707.mini.tar.bz2
#cd openwrt-2.6.36-7621/
#make menuconfig
```

```
[root@ht test]# cd openwrt-2.6.36-7621/
[root@ht openwrt-2.6.36-7621]# make menuconfig
Checking 'working-make'... ok.
Checking 'case-sensitive-fs'... ok.
Checking 'getopt'... ok.
Checking 'fileutils'... ok.
Checking 'working-gcc'... ok.
Checking 'working-g++'... ok.
Checking 'ncurses'... ok.
Checking 'zlib'... ok.
Checking 'gawk'... ok.
Checking 'unzip'... ok.
Checking 'bzip2'... ok.
Checking 'perl'... ok.
Checking '/usr/bin/python'... ok.
Checking 'wget'... ok.
Checking 'git'... ok.
Checking 'gnutar'... ok.
Checking 'svn'... ok.
Checking 'openssl'... ok.
Checking 'gnu-find'... ok.
Checking 'getopt-extended'... ok.
Checking 'non-root'... failed.

Build dependency: Please do not compile as root.

Prerequisite check failed. Use FORCE=1 to override.
make: *** [tmp/.prereq-build] Error 1
[root@ht openwrt-2.6.36-7621]# chmod 777 -R tmp/
[root@ht openwrt-2.6.36-7621]# su vonets
[vonets@ht openwrt-2.6.36-7621]$ make menuconfig
Checking 'working-make'... ok.
Checking 'case-sensitive-fs'... ok.
Checking 'getopt'... ok.
Checking 'fileutils'... ok.
Checking 'working-gcc'... ok.
Checking 'working-g++'... ok.
Checking 'ncurses'... ok.
Checking 'zlib'... ok.
Checking 'gawk'... ok.
Checking 'unzip'... ok.
Checking 'bzip2'... ok.
Checking 'perl'... ok.
Checking '/usr/bin/python'... ok.
Checking 'wget'... ok.
Checking 'git'... ok.
Checking 'gnutar'... ok.
Checking 'svn'... ok.
Checking 'openssl'... ok.
Checking 'gnu-find'... ok.
Checking 'getopt-extended'... ok.
Checking 'non-root'... ok.
make -s -C scripts/config all CC=gcc: build failed. Please re-run make with V=s
to see what's going on
make: *** [scripts/config/mconf] Error 1
[vonets@ht openwrt-2.6.36-7621]$ su root
Password:
[root@ht openwrt-2.6.36-7621]# make menuconfig
Collecting package info: done
```

make menuconfig 成功后，选择正确的配置参数，保存退出：

```
Target System (Ralink RT288x/RT3xxx) --->
Subtarget (MT7620a based boards) --->
Target Profile (MT7620a+MT7610e) --->
Target Images --->
Global build settings --->
[ ] Advanced configuration options (for developers) --->
[ ] Build the OpenWrt Image Builder
[ ] Build the OpenWrt SDK
[ ] Build the OpenWrt based Toolchain
[ ] Image configuration --->
Base system --->
Boot Loaders --->
Development --->
Kernel modules --->
Languages --->
Libraries --->
Network --->
Ralink Properties --->
Utilities --->
```

5. root 用户下，make V=s(带调试语句的编译)。第一次编译大约几小时。如果编译出错根据出错提示在线安装相应的库或者工具。

6. 编译成功后，在 Openwrt 的 bin/ramips/出现编译后的 bin 文件和 packages。

```
root@ht home]# ls -l openwrt-2.6.36-7621/bin/ramips/
total 17272
-rw-r--r-- 1 root root 374 2014-07-07 15:11 md5sums
-rw-r--r-- 1 root root 5242884 2014-07-07 15:11 openwrt-ramips-mt7620a-mt7620a_mt7610e-squashfs-sysupgrade.bin
-rw-r--r-- 1 root root 3932160 2014-07-07 15:11 openwrt-ramips-mt7620a-root.squashfs
-rw-r--r-- 1 root root 1221491 2014-07-07 15:11 openwrt-ramips-mt7620a-ulimage.bin
-rwxr-xr-x 1 root root 3648500 2014-07-07 15:11 openwrt-ramips-mt7620a-vmlinux.bin
-rwxr-xr-x 1 root root 3714956 2014-07-07 15:11 openwrt-ramips-mt7620a-vmlinux.elf
-rwxr-xr-x 2 root root 12288 2014-07-07 15:11 packages
```

7. 编译可参考文档《MTK-OpenWrt-2.6.36-SDK-Release Notes.pdf》。