ปรับแต่งระบบปฏิบัติการ FreeBSD ด้วยการ Compile Kernel

เมื่อเราติดตั้งระบบปฏิบัติการ FreeBSD เสร็จ โดยปกติแล้วเราจะได้ GENERIC kernel ซึ่ง FreeBSD เตรียมไว้ให้ โดย GENERIC Kernel นั้นจะรู้จัก Hardware จำนวนมาก ทำให้ Server ของเราต้องตรวจสอบ Hardware ว่าเครื่อง Server ของเรามีอะไรบ้าง เสียเวลาในการ boot ระบบ อีกทั้งต้องใช้หน่วยความจำหลักในการทำงานดังกล่าวพอสมควร เราสามารถทำการ Compile Kernel ใหม่ให้เหมาะสมกับ Hardware ของเราได้

ในการ Compile Kernel ในบทนี้จะเป็นเพียงการเพิ่มความสามารถให้กับระบบปฏิบัติการ FreeBSD ให้สามารถทำงานเป็น Gateway ให้กับ Network ย่อยของเรานั่นคือให้ FreeBSD server ของเราทำหน้าที่เป็น NAT (Network Address Translation) และ Firewall

ขั้นตอนการ Compile Kernel

1. เมื่อตอนติดตั้งเราได้เลือก src ไว้แล้ว ให้เราเข้าไปที่เก็บ GENERIC Kernel แล้วทำการ copy GENERIC Kernel เป็น ชื่อ Kernel ของเรา เช่น GATEWAY

```
#cd /usr/src/sys/i386/conf
#cp GERNERIC GATEWAY
```

2. ตรวจสอบ CPU ของเราว่า เป็น Class ใด โดยใช้คำสั่ง

```
#dmesg | grep CPU
```

จะได้ผลลัพธ์ดังภาพ นั่นคือ 686-class CPU

```
demo# dmesg ! grep CPU
CPU: Intel(R) Celeron(R) CPU 2.80GHz (2820.21-MHz 686-class CPU)
cpu0: <ACPI CPU> on acpi0
acpi_throttle0: <ACPI CPU Throttling> on cpu0
demo#
```

3. เมื่อตรวจสอบ CPU แล้วให้เราทำการแก้ไขไฟล์ Kernel ของเรา

```
#ee GATEWAY
```

4.ในไฟล์ GATEWAY ให้แก้ไข บรรทัดต่อไปนี้

ก่อนแก้ไข		หลังแก้ไข	
machine	i386	machine	i386
cpu	I486_CPU	#cpu	I486_CPU
cpu	I586_CPU	#cpu	I586_CPU
cpu	I686_CPU	cpu	I686_CPU
ident	GENERIC	ident	**GATEWAY**

5. ในไฟล์เดียวกัน ให้เราทำการเพิ่ม options ต่าง ๆ ต่อไปนี้ ก่อน options เดิมที่มีอยู่แล้ว เพื่อให้ Kernel ของเรา ทำงานเป็น NAT และ Firewall รวมทั้ง เพิ่ม options QUOTA เพื่อให้เราสามารถจัดการจำกัดขนาดพื้นที่ให้ user ใน server ใช้งานตามโควตา ในภายหลัง

```
options         IPFIREWALL
options         IPFIREWALL_FORWARD
options         IPFIREWALL_DEFAULT_TO_ACCEPT
options         IPFIREWALL_VERBOSE
options         IPFIREWALL_VERBOSE_LIMIT=100
options         IPDIVERT
options         QUOTA
```

6. ให้ทำการ Save ไฟล์โดยการกด Ctl + C แล้วพิมพ์ exit เพื่อออกมาที่ prompt ของ ระบบหลังจากนั้นให้ทำการ config Kernel ที่เราได้แก้ไขแล้ว โดยใช้คำสั่งดังนี้

```
#config  GATEWAY
```

จะได้ผลดังนี้

```
Kernel build directory is ../compile/GATEWAY
Don't forget to do a ``make depend''
```

7. ให้เราเข้าไปทำการ compile kernel และติดตั้ง โดยใช้คำสั่ง

```
#cd  ../compile/GATEWAY
#make cleandepend ; make depend
#make ; make install clean
```

8. ให้เราทำการ reboot ระบบใหม่

```
#reboot
```

หรืออาจจะใช้คำสั่งต่อไปนี้แทนก็ได้ สังเกต –r = reboot , -p = power off

```
#shutdown –r now
```

9. ตรวจสอบระบบว่า boot ด้วย Kernel ใหม่ที่ชื่อ GATEWAY หรือไม่

```
#uname –a
```

ต้องได้ผลดังนี้

```
FreeBSD demo.aru.ac.th 5.4-RELEASE FreeBSD 5.4-RELEASE #0: Sun Apr 15
14:04:19 ICT 2007
root@demo.aru.ac.th:/usr/src/sys/i386/compile/GATEWAY  i386
```

ขณะนี้เราได้ Kernel ใหม่ที่รองรับการทำ NAT และ Firewall แล้วรวมถึงสามารถจัดการ QUOTA ให้กับ user บน server แต่ละคนได้แล้ว แต่ Server จะยังไม่ทำงานเป็น NAT และ Firewall จนกว่าเราจะconfig ให้ server ทำงานตามที่เราต้องการ

ให้เราทำการ config ค่าต่าง ๆ เหล่านี้ ในไฟล์ /etc/rc.conf เพิ่มเติม โดยมีสองทางเลือก ทีทำงานเหมือนกัน

```
#ee /etc/rc.conf
```

ทางเลือกที่ 1 เพิ่มบรรทัดต่อไปนี้

```
firewall_enable=”YES”
firewall_type=”OPEN”
firewall_quite=”YES”
natd_enable=”YES”
natd_interface=”rl0”
natd_flags=”-s –u -m”
```

หรือ ทางเลือกที่ 2 เพิ่มบรรทัดต่อไปนี้

```
firewall_enable=”YES”
firewall_type=”OPEN”
firewall_quite=”YES”
natd_enable=”YES”
natd_flags="-f /etc/natd.conf"
```

และสร้าง ไฟล์ `/etc/natd.conf` เพิ่ม โดยมีข้อความนี้อยู่ในไฟล์

```
interface rl0
use_socket yes # -s
same_ports yes # -m
unregistered_only yes # -u
```

หมายเหตุ : rl0 **คือ** LAN Card **ที่ต่อกับ** internet

โดยต้องมีบรรทัดนี้อยู่ด้านบนในไฟล์ /etc/rc.conf ก่อนหน้านี้

```
gateway_enable=”YES”
```

เพื่อให้ server ของเรา ทำหน้าที่เป็น gateway ของ network ได้อย่างสมบูรณ์

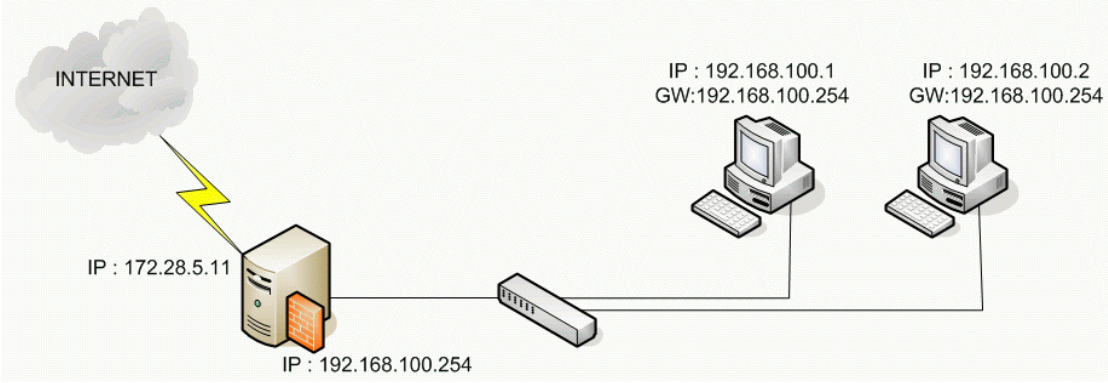
ให้ทำการ reboot server อีกครั้ง แล้วให้ลองใช้คำสั่ง

```
#ipfw show
```

เพื่อตรวจสอบการทำ NAT ซึ่ง Firewall จะยอมให้ทุก package ที่ เข้า - ออก เครื่องลูกข่าย ผ่านทาง NAT Server ตัวนี้ได้ ซึ่งจะได้ผลคล้ายแบบนี้

```
00050  505  57856 divert 8668 ip from any to any via rl0
00100   12   1008 allow ip from any to any via lo0
00200    0      0 deny ip from any to 127.0.0.0/8
00300    0      0 deny ip from 127.0.0.0/8 to any
65000  639  77505 allow ip from any to any
65535    0      0 allow ip from any to any
```

เพื่อให้เข้าใจในการทำ NAT นั้นจะขออธิบายเพิ่มเติมดังนี้



จากรูปแสดงเครือข่ายด้านบน ในไฟล์ /etc/rc.conf จะมีลักษณะคล้ายดังนี้

```
gateway_enable="YES"
hostname="demo.aru.ac.th"
ifconfig_rl0=”inet 172.28.5.11  netmask 255.255.0.0”
ifconfig_rl1=”inet 192.168.100.254 netmask 255.255.255.0”
```

โดย rl0 คือ LAN Card ใบแรก ที่ใช้ต่อไปยังเครือข่าย internet ส่วน rl1 นั้น จะเป็น LAN Card ที่ต่อกับเครือข่ายภายในของเรา ทั้งนี้หาก Card LAN ของเราไม่ใช่ยี่ห้อ Realtek จะไม่ใช่ชื่อ rl โดยอาจจะเป็นชื่ออื่น ให้ลองใช้คำสั่งต่อไปนี้ ตรวจสอบดู

#ifconfig จะได้ผลคล้าย ๆ แบบนี้

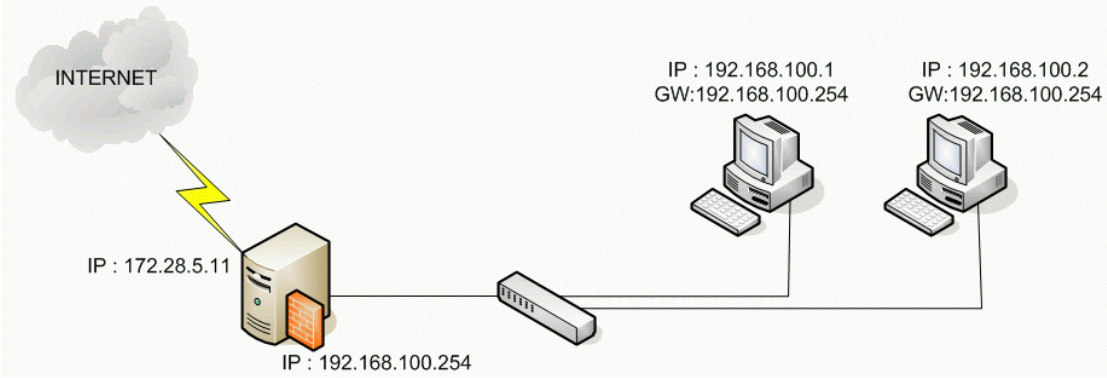
```
demo# ifconfig
lnc0: flags=108843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        inet 172.28.5.11 netmask 0xffff0000 broadcast 172.28.255.255
        inet6 fe80::20c:29ff:fed7:a0d8%lnc0 prefixlen 64 scopeid 0x1
        ether 00:0c:29:d7:a0:d8
lnc1: flags=108843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        inet 192.168.100.254 netmask 0xffffff00 broadcast 192.168.100.255
        inet6 fe80::20c:29ff:fed7:a0e2%lnc1 prefixlen 64 scopeid 0x2
        ether 00:0c:29:d7:a0:e2
plip0: flags=108810<POINTOPOINT,SIMPLEX,MULTICAST> mtu 1500
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 16384
        inet 127.0.0.1 netmask 0xff000000
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x4
```

ตัวอย่างรหัส Card LAN ที่พบได้บ่อย เช่น

```
de     # DEC/Intel DC21x4x (“Tulip”)
em     # Intel PRO/1000 adapter Gigabit Ethernet Card (“Wiseman”)
txp    # 3Com 3cR990 (“Typhoon”)
vx     # 3Com 3c590, 3c595 (“Vortex”)
dc     # DEC/Intel 21143 and various workalikes
fxp    # Intel EtherExpress PRO/100B (82557, 82558)
pcn    # AMD Am79C97x PCI 10/100 NICs
rl     # RealTek 8129/8139
sf     # Adaptec AIC-6915 (“Starfire”)
sis    # Silicon Integrated Systems SiS 900/SiS 7016
ste    # Sundance ST201 (D-Link DFE-550TX)
tl     # Texas Instruments ThunderLAN
tx     # SMC EtherPower II (83c170 “EPIC”)
vr     # VIA Rhine, Rhine II
wb     # Winbond W89C840F
xl     # 3Com 3c90x (“Boomerang”,”Cyclone”)
bge    # Broadcom BCM570x (“Tigon III”)
```

DHCP Server

Dynamic Host Configuration Protocol เป็น protocol ที่ใช้ในการกำหนด IP address ให้กับอุปกรณ์หรือเครื่องคอมพิวเตอร์ในเครือข่ายแบบ dynamic ซึ่งการกำหนด IP Address แบบ dynamic นี้ อุปกรณ์หรือเครื่องคอมพิวเตอร์จะได้รับ IP address ที่แตกต่างกันไปในแต่ละครั้งที่เชื่อมต่อมายังเครือข่าย แต่ DHCP ก็ยังสามารถกำหนด IP Address แบบ static IP address ได้เช่นกัน



ขั้นตอนการติดตั้ง

ในการติดตั้งนี้ เราจะทำการติดตั้งผ่าน ports ของ FreeBSD โดยถ้าหากยังไม่เคย copy ไฟล์ f54distfiles.tar.gz จากแผ่น CD ก็ให้ทำการ copy ไฟล์ f54distfiles.tar.gz จากแผ่น CD ที่เตรียมให้ ไปเก็บไว้ที่ /usr/ports/distfiles/ และเมื่อ copy เรียบร้อยแล้ว จึงค่อยแตกไฟล์ ดังกล่าวออกมา ดังนี้

```
#cd  /usr/ports/distfiles/
#mount  /cdrom
#cp  /cdrom/f54distfiles.tar.gz  ./
#umount  /cdrom
#tar  -zxvf  f54distfiles.tar.gz
#rm  f54distfiles.tar.gz
```

1. ให้เข้าไปที่ ports ของ DHCP แล้วทำการติดตั้ง

```
#cd  /usr/ports/net/isc-dhcp3-server/
#make  ;  make install
#rehash
```

ในการติดตั้งเมื่อเกิด Dialog ถาม การติดตั้ง ให้กด tab มาที่ OK แล้วจึง Enter เพื่อทำงานต่อ

2. เมื่อติดตั้งเสร็จให้แก้ไขไฟล์ /usr/local/etc/dhcpd.conf

```
#ee /usr/local/etc/dhcpd.conf
```

3. เพิ่มเติมข้อความต่อไปนี้ในไฟล์ dhcpd.conf (โดยอ้างอิงการเชื่อมต่อจากรูปเครือข่ายด้านบน)

```
option      domain-name “aru.ac.th”;
option      domain-name-servers     202.29.62.7 , 172.16.1.2;
default-lease-time      86400;
max-lease-time          172800;
authoritative;
ddns-update-style       ad-hoc;
log-facility            local7;
subnet  192.168.100.0  netmask  255.255.255.0{
      range  192.168.100.11   192.168.100.200;
      option  subnet-mask     255.255.255.0;
      option  broadcast-address     192.168.100.255;
      option  routers   192.168.100.254;
}
```

4. หากต้องการ Fix IP address ให้กับคอมพิวเตอร์ลูกข่ายเครื่องใดเราต้องทราบ MAC Address ของ LAN Card ของคอมพิวเตอร์เครื่องนั้น ให้เราเพิ่มข้อความคล้ายตัวอย่างต่อไปนี้เข้าไปในไฟล์ dhcpd.conf ด้วย เช่น

```
host r31119c01{
        hardware ethernet 00:14:85:52:f3:d2;
        fixed-address 192.168.100.1;
}
```

5. ให้ DHCP server ทำงาน

```
#dhcpd  -q  &
```

6. ให้เพิ่มบรรทัดต่อไปนี้ในไฟล์ /etc/rc.conf เพื่อให้ DHCP ทำงานทุกครั้งที่ boot เครื่องใหม่

```
dhcpd_enable="YES"        # dhcpd enabled?
dhcpd_flags="-q"          # command option(s)
dhcpd_conf="/usr/local/etc/dhcpd.conf"  # configuration file
dhcpd_ifaces="rl1"        # ethernet interface(s)
dhcpd_withumask="022"     # file creation mask
```

7. หากตั้งค่าต่างๆ ตามตัวอย่างนี้ หากเครื่องลูกข่ายที่เชื่อมต่อกับ switch ก็จะได้รับ IP Address โดยอัตโนมัติ รวมถึงค่า subnet mask , DNS เพื่อใช้ในการเชื่อมต่ออินเตอร์เน็ตต่อไป

Proxy Server หรือ Cache Server

Proxy server คือ server ที่กั้นอยู่ตรงกลางระหว่าง workstation กับ Internet ทำหน้าที่เป็น web caching :ซึ่ง proxy server อาจจะเป็นตัวเดียวกับ gateway server ที่ทำหน้าที่แยกเครือข่ายภายในองค์กรจากเครือข่ายภายนอก

โปรแกรมที่ใช้ในการทำ caching คือโปรแกรม Squid (http://www.squid-cache.org) เวอร์ชั่น Stable ปัจจุบันได้แก่ http://www.squid-cache.org/Versions/v2/2.6/squid-2.6.STABLE12.tar.gz แต่ในการติดตั้งเราจะใช้ไฟล์จากแผ่น config ที่เตรียมไว้ให้เป็นเวอร์ชั่น squid-2.5.STABLE11 คือ ไฟล์ squid-2.5.STABLE11.tar.gz

การติดตั้ง Squid-Cache

1. ให้ทำการ copy ไฟล์ squid-2.5.STABLE11.tar.gz จากแผ่น CD ไปไว้ยัง Directory /tmp

```
#cd  /tmp
#mount /cdrom
#cp  /cdrom/proxy/squid-2.5.STABLE11.tar.gz  ./
#umount  /cdrom
```

2. ให้แตกไฟล์ที่ copy มาจาก CD และทำการติดตั้ง ดังนี้

```
#tar  -zxvf  squid-2.5.STABLE11.tar.gz
#cd squid-2.5.STABLE11
#./configure --enable-delay-pools --enable-basic-auth-helpers="NCSA"
#make all ; make install
#rehash
```

3. สร้าง logs ไฟล์เพื่อใช้ในการเก็บ log การเข้าดูเวบของเครื่องลูกข่าย

```
#cd /usr/local/squid/var/logs/
#touch access.log
#touch store.log
#touch cache.log
#chmod  777  *.log
```

4. สร้าง Directory ไว้เก็บเวบต่าง ๆ ที่เครื่องลูกข่ายเข้าไปดูมา

```
#cd /usr/local/squid/var/
#mkdir cache
#chmod -R 777  cache
```

5. แก้ไข config ไฟล์ของโปรแกรม squid

```
#cd  /usr/local/squid/etc/
#ee squid.conf
```

โดยให้แก้ไขบรรทัดต่อไปนี้

บรรทัดที่มี # http_port 3128 (ไม่ต้องแก้) (ประมาณบรรทัดที่ 53)

ให้เพิ่มบรรทัดนี้เข้าไป http_port 8080

บรรทัด # cache_dir ufs /usr/local/squid/var/cache 100 16 256 ให้ copyลงมา อีก 1 บรรทัดแล้วจึงแก้เป็น cache_dir ufs /usr/local/squid/var/cache 4096 16 256

ที่บรรทัดต่อไปนี้ (ประมาณ 1832)ให้พิมพ์แทรกบรรทัด ดังตัวอย่างเข้าไป (IP address ดูจากเครือข่ายที่เราออกแบบไว้)

acl localhost src 127.0.0.1/255.255.255.255 (ของเก่า)

acl mynet src 192.168.100.0/24 (เพิ่มบรรทัดนี้)

acl to_localhost dst 127.0.0.0/8 (ของเก่า)

ให้ squid ยอมให้เครื่องลูกข่ายที่มาจาก mynet ออกสู่อินเตอร์เน็ตได้

#acl our_networks src 192.168.1.0/24 192.168.2.0/24 (ของเก่า)

#http_access allow our_networks (ของเก่า)

http_access allow mynet (เพิ่มบรรทัดนี้)

And finally deny all other access to this proxy (ของเก่า)

ที่บรรทัดประมาณ 2287 ให้เพิ่มส่วนของการทำ Transparent proxy ดังนี้

```
httpd_accel_host virtual
httpd_accel_port 80
httpd_accel_with_proxy on
httpd_accel_uses_host_header on
```

หลังจากนั้นให้ทำการ save ไฟล์ เพื่อทำงานต่อ

6. สร้างโครงสร้างการเก็บข้อมูลของ squid

```
#cd /usr/local/squid/sbin/
#./squid –z
```

7. สั่งให้ squid ทำงาน

```
#/usr/local/squid/bin/RunCache &
```

8. ให้เพิ่ม กฎ ของ firewall ต่อไปนี้ เพื่อให้ลูกข่ายสามารถใช้ internet ได้

```
#ipfw add 1301 fwd 172.28.5.11,8080 tcp from 192.168.100.0/24 to any 80
```

9. ลองใช้เครื่องลูกข่ายเข้าอินเตอร์เน็ตดู แล้วใช้คำสั่งต่อไปนี้ตรวจสอบการทำงานของ Squid Proxy

```
#tail -f /usr/local/squid/var/logs/access.log
```

10. หากมีการแก้ไขค่า config ของ squid ให้ใช้คำสั่งต่อไปนี้เพื่อให้ squid ทำงานตามการ config ค่าใหม่

```
#/usr/local/squid/sbin/squid –k reconfig
```

11. เพิ่มความสะดวกในการเรียกใช้ squid จาก directory ใดก็ได้

```
#cd /usr/sbin/
# ln -s /usr/local/squid/sbin/squid squid
# rehash
```

12. ให้ squid ทำงานทุกครั้งที่เปิดเครื่อง และทำงานเป็น Transparent Proxy โดยการเข้าไปแก้ไขไฟล์ /etc/rc.local พิมพ์ข้อความต่อไปนี้ลงไป(คำสั่งจากข้อ 7 และ 8)

```
ipfw add 1301 fwd 192.168.100.254,8080 tcp from 192.168.100.0/24 to any 80
/usr/local/squid/bin/RunCache &
```

ขณะนี้เราได้ Proxy server ที่ทำงานเป็น Transparent Proxy แล้ว สิ่งที่เราต้องทำต่อไป คือ

13. การ clear cache ของ proxy ซึ่งมีขั้นตอนดังนี้

13.1 ให้สร้างไฟล์ที่ชื่อว่า squid.sh ไว้ใน /root เพื่อใช้ในการ start / stop squid

```
#cd /root/
#ee squid.sh
```

ให้พิมพ์ข้อความเหล่านี้ลงไปในไฟล์

```
#!/bin/sh
case "$1" in
start)
if [ -f /usr/local/squid/bin/RunCache ]; then
echo -n 'starting Squid ...'
(/usr/local/squid/bin/RunCache &)
fi
;;
stop)
kill -9 `ps ax | grep RunCache | awk '{print $1}'`;
/usr/local/squid/sbin/squid -k shutdown;
;;
*)
echo "squid.sh stop|start"
;;
esac
exit 0
```

เมื่อพิมพ์เสร็จแล้วให้ทำการ save และเปลี่ยน permission ไฟล์ เพื่อให้ shell script ตัวนี้ ทำงานได้

```
#chmod 755 squid.sh
```

ในการเรียกใช้งาน ให้ทำดังนี้

```
#cd /root/
#./squid.sh  stop
```

เพื่อ stop squid process

```
#./squid.sh  start
```

เพื่อ start squid process

13.2 ให้ทำการ Clear Cache และสร้างใหม่ดังนี้

```
#cd /root/
#./squid stop
#cd  /usr/local/squid/var/logs/
#rm *.log
#touch access.log cache.log store.log
#chmod 777 *.log
#cd /usr/local/squid/var/
#rm  –rf  cache/
#mkdir  cache
#chmod -R 777 cache
#/usr/local/squid/sbin/squid –z
#cd /root/
#./squid.sh  start
```

14. การ Block เวบต้องห้าม

ในไฟล์ /usr/local/squid/etc/squid.conf ให้เพิ่มเติมบรรทัดที่อยู่ในกรอบ ลงไประหว่างบรรทัดที่อยู่นอกกรอบดังตัวอย่างต่อไปนี้

```
. . . . . .
acl Safe_ports port 488 # gss-http
acl Safe_ports port 591 # filemaker
acl Safe_ports port 777 # multiling http
acl CONNECT method CONNECT
```

```
acl porn url_regex “/usr/local/squid/etc/porn.block.txt”
http_access deny porn all

acl bannedsites url_regex “/usr/local/squid/etc/banned.block.txt”
http_access deny bannedsites all

acl blocked_sites url_regex www.xxx.com
acl blocked_sites url_regex www.yyy.com
acl blocked_sites url_regex www.zzz.com
http_access deny blocked_sites
```

```
#  TAG: http_access
#       Allowing or Denying access based on defined access lists
. . . . . .
```

แล้วจึงไปสร้างไฟล์ /usr/local/squid/etc/porn.block.txt ซึ่งเก็บรายชื่อเวบไซต์ต้องห้าม และไฟล์ /usr/local/squid/etc/banned.block.txt เช่นตัวอย่าง

www.xxx.com

www.porn.com

www.sex.com

หมายเหตุ : ทุกครั้งที่แก้ไข ไฟล์ที่เกี่ยวข้องต้อง stop / start squid ใหม่ โดยอาจจะใช้คำสั่ง reconfig ก็ได้

```
#/usr/local/squid/sbin/squid –k reconfig
```

15. Block ไม่ให้ Download ไฟล์ เช่น ไม่ให้ download ไฟล์ .mpg , .zip , exe , .wma เป็นต้น ให้แก้ไขไฟล์ /usr/local/squid/etc/squid.conf เพิ่มเติมต่อจากส่วนที่แล้ว

```
acl download url_regex -i \.mpg$
acl download url_regex -i \.zip$
acl download url_regex -i \.exe$
acl download url_regex -i \.wma$
acl download url_regex -i \.torrent$
http_access deny download
```

หมายเหตุ : ทุกครั้งที่แก้ไข ไฟล์ที่เกี่ยวข้องต้อง stop / start squid ใหม่ โดยอาจจะใช้คำสั่ง reconfig ก็ได้

```
#/usr/local/squid/sbin/squid –k reconfig
```

16. สั่งให้ squid Clear Cache แบบอัตโนมัติ

16.1 ให้สร้าง clrproxy.sh ไว้ที่ /root โดยมีรายละเอียดดังนี้

```
kill -9 `ps ax | grep RunCache | awk '{print $1}'`
/usr/local/squid/sbin/squid -k shutdown
sleep 3
cd /usr/local/squid/var/logs/
rm access.log
rm cache.log
rm store.log
touch access.log cache.log store.log
chmod 777 access.log
chmod 777 cache.log
chmod 777 store.log
sleep 3
rm -rf /usr/local/squid/var/cache
sleep 15
mkdir /usr/local/squid/var/cache
chmod -R 777 /usr/local/squid/var/cache
/usr/local/squid/sbin/squid -z
sleep 15
/usr/local/squid/bin/RunCache &
echo "OK!"
```

16.2 แก้ไข /etc/crontab โดยเพิ่มบรรทัดนี้ต่อท้ายของเดิม

```
0 0 15,30 * * root   /root/clrproxy.sh
```

ระบบจะทำการ Clear Cache ให้ทุกวันที่ 15 และวันที่ 30 ของทุกเดือน ในเวลา เที่ยงคืน