# Introduction to SOI-Asia Network

August 18, 2005

## 1 Introduction

A satellite covers a wide area on the earth and can broadcast data beyond the limitations of geographical boundaries. Satellite links are effective infrastructure to provide Internet connectivity in the areas where terrestrial data links or additional bandwidth for an existing data link is hardly available.

SOI-Asia network is the network infrastructure which supports the whole SOI-Asia activities. SOI-Asia system has specific requirements and restrictions in terms of bandwidth, geographically distributed partners, network topology, resources and applications. Therefore, SOI-Asia network is specially designed to meet all those requirements and restrictions.

## 2 SOI-Asia Network Overview

### 2.1 Topology

SOI-Asia network uses a satellite link as a Uni-Directional Link (UDL) like the figure 1 because of following reasons.

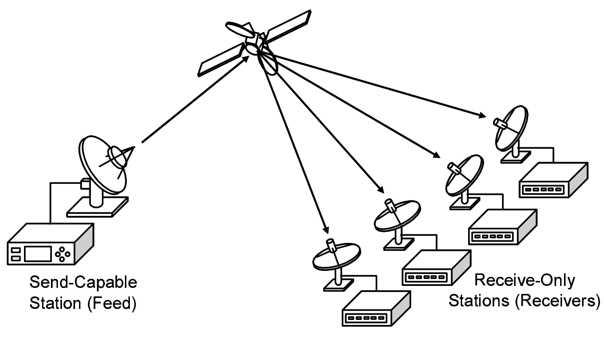


Figure 1: Satellite Uni-Directional Link

- Satellite uni-directional link can broadcast data over satellite to many contries, which makes easierto share the information among partners.

- Receive-only earth stations are cheaper than transmit-capable earth stations, thus making the environment easier to build.

- In most countries, we do not need transmission licenses for receive-only earth stations, while it is very difficult to obtain one for transmit-capable earth stations.
- We do not need radio specialists in receive-only sites.

Figure 2 shows the outline of SOI-Asia network topology. The satellite link in the figure is the uni-directional communication media from Japan to partners.

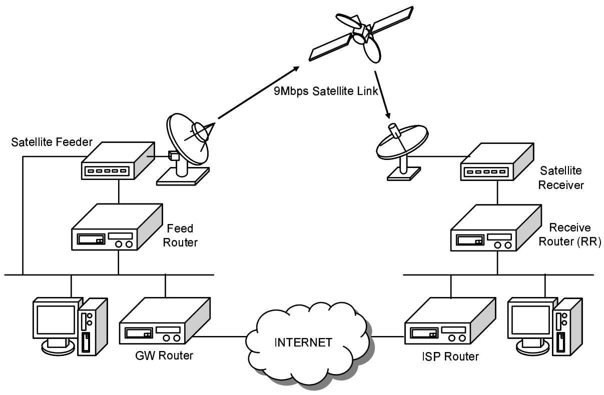


Figure 2: SOI-Asia Network Topology

The network in the right side is SFC which has the transmit-capable earth station for SOI-Asia satellite link and works as the upstream of each partner site. On the other hand, the network in the right side is the receive-only site established in each partner organization.

## 2.2 Receive-Only Site in SOI-Asia Partner

The figure 3 shows the network topology of a receive-only site.

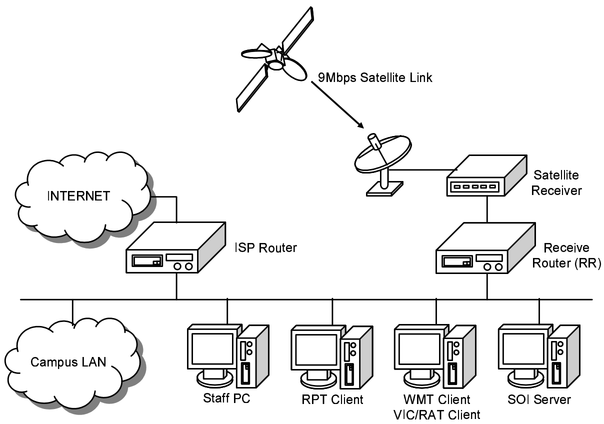


Figure 3: Network Topology of Receive-Only Site

The satellite link in the figure 2 is the communication media from Japan to partners. The network in the right side is SFC which has the transmit-capable earth station for SOI-Asia satellite link and works as the upstream of each partner site. The receive-only site has a receive-only earth station to receive data from the satellite link. SOI-Asia assigns a global IPv4/v6 address space to each partner. The global IPv4 addresses could be assigned to machines which have important roles in SOI-Asia activities: Receive Router, SOI Server, WMT client, RTP client, VIC/RAT or Polycom and staff machine.

Figure 4 shows the logical subnets in the receive-only site. Besides the global IP address assigned to SOI-Asia, each partner site must have another network to access to the Internet, to be able to send traffic back to Japan. It can be a local ISP, dialup or the partner's campus network in the case that campus has another way out to Internet. In case of any additional hosts, they will be placed in a private network using private addresses, noted as the campus LAN. Therefore, several logical subnets are layed on a single link in each receive-only site.

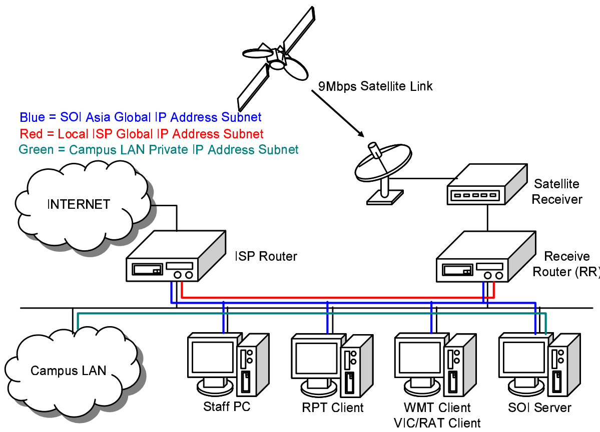


Figure 4: Logical Subnets on Receive-Only Site

# 3 Uni-directional Link Routing (UDLR)

## 3.1 Background of UDLR

In the Internet, data links are generally categorized into 3 types: "Point-to-Point", "Broadcast-Capable Multiple Access" and "Non-Broadcast Multiple Access". However, a uni-directional link does not mach any of these 3 types. Therefore, in this scenario of uni-directional links, because most protocols and applications of the Internet assume that the data link is compliant to these 3 types of links, those applications do not work correctly. Many routing algorithms, for examples, OSPF, IGMP, PIM, DVMRP, requires two-way broadcasting ability on the link.

## 3.2 Node Definition in UDLR

In order to utilize the satellite UDL, SOI-Asia employs UDLR, a set of technologies which emulate Broadcast-Capable Multiple Access link on the unidirectional link. In the UDLR specification, several types of nodes are defined. In this document, we focus on "Send-only Feed" and "Receiver" to explain UDLR simply. Note that SOI-Asia network is composed of these two types of nodes.

**Send-only Feed** A router that has send-only connectivity to a UDL.

**Receiver** A router or a host that has receive-only connectivity to a UDL.

## 3.3 UDLR Technologies

UDLR is composed of three technologies: Link Layer Tunneling Mechanism, Broadcast Emulation and Dynamic Tunnel Configuration Protocol.

### 3.3.1 Link-Layer Tunneling Mechanism (LLTM)

LLTM is a technology that delivers a data link frame from Receiver to Send-only Feed via the other Internet infrastructure by encapsulating the data link frame in the GRE (Global Routing Encapsulation) header. Because LLTM deliver a data link frame with with keeping its same shape when it was sent from the I/F of the Receiver on the UDL, Send-only Feed treats that data link frame as if it was directly received from UDL. Figure 5 shows how LLTM works on Send-only Feed and Receiver.

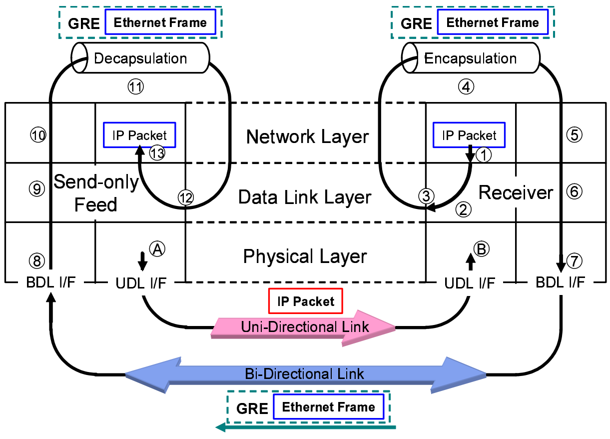


Figure 5: Link Layer Tunneling Mechanism

**(From Receiver to Send-only Feed)**

**1** Receiver sends an IP packet to Send-only Feed on satellite UDL. Because Send-only Feed is visible on UDL from Receiver, Receiver is about sending the packet to Send-only Feed on UDL.

**2** For the Ethernet header which is attached to the originated IP packet, the destination MAC address is that of UDL I/F of FEED and the source MAC address is that of UDL I/F Receiver.

**3** Because LLTM knows that Receiver can't transmit the data link frame to UDL, LLTM captures the data link frame before it is passed to UDL I/F. Then, LLTM send the data link frame to the encapsulation functionality of Receiver.

**4** LLTM encapsulates the captured data link frame in the GRE header. The encapsulated data ink frame is passed to the network layer of BDL side stack to send it via the Internet.

**5 - 7** The IP header is attached to the GRE encapsulated payload and it is transmitted from BDL I/F of Receiver to the Internet.

**8 - 10** The GRE-encapsulated data link frame is reaching BDL I/F of Send-only Feed through the Internet and passed the to upper layers.

**11** On Send-only Feed, LLTM detaches GRE header to decapsulate the original data link frame from Receiver. Then, LLTM passes it to the data link layer of UDL side stack on Send-only Feed.

**12 - 13** Send-only Feed processes the data link frame as if it were incoming from UDL I/F and pass it to the upper layers. At last, the original IP packet reaches FEED on satellite UDL as if it were directly delivered from Receiver to Send-only Feed.

**(From Send-only Feed to Receiver)**

**A - B** Send-only Feed can just send an IP packet to Receiver via UDL.

### 3.3.2 Broadcast Emulation

Broadcast Emulation is the technology that Send-only Feed transmits a received data link frame to the UDL. Then, the data link frame could be shared among other Receivers like Ethernet. Broadcast Emulation emulates Broadcast-Capable Multiple Access Link on the UDL.

Broadcast Emulation makes Send-only Feed transmit a data link frame if it is destined to Receiver's MAC address, a broadcast address or a multicast address. Here Receiver can send a packet to other Receiver by way of Send-only Feed, which realizes bidirectional communication between Receivers on UDL.

Broadcast emulation is essential to successfully complete ARP or other infrastructure technologies of the Internet. Figure 6 shows how Broadcast Emulation works taking ARP as an example.

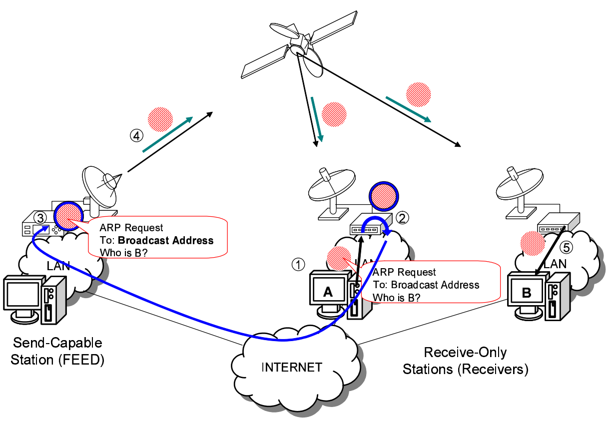


Figure 6: ARP with Broadcast Emulation

**1** Host A wants to send an IP packet to Host B. Host A sends an ARP request to know what the MAC address of Host B is.

**2** The ARP request is encapsulated by LLTM on Receiver and reaches FEED through the Internet.

**3** Now Feed decapsulates the GRE-encapsulated data link frame, and the ARP request appears, whose MAC address is destined to the UDL subnet's broadcast address.

**4** Broadcast Emulation forwards the decapsulated data link frame to the satellite UDL, if its destination MAC address is that for multicast or broadcast. Then the ARP request is forwarded to the satellite UDL.

**5** Host B receives the ARP request from Host A, and host B can send an ARP reply successfully.

#### 3.3.3 Dynamic Tunnel Configuration Protocol (DTCP)

Dynamic Tunnel Configuration Protocol is the protocol that the Send-only Feed periodically advertise their IP address as the end-point of LLTM over UDL. Listening to the advertisement from Send-only Feed, Receivers know where to send GRE encapsulated packets.

DTCP Hello is a packet that delivers the IP address of the tunnel end-point. Because Receivers rely on DTCP for knowing the tunnel end-point, if that packet does not come from UDL, Receivers can not activate LLTM any more.

### 3.4 UDLR in SOI-Asia Network

In the UDLR specification, Send-only Feed is dedined as a router. And Receiver is defined as a router or a host. But, in the SOI-Asia network, the UDLR functionality is separated from a router or a host for both Send-only Feed and Receiver.

**Satellite Feed** A host that has send-only connectivity to a UDL. It provides the functionality of Send-only Feed but does not have any routing capability.

**Satellite Receiver** A host that has receive-only connectivity to a UDL. It provides the functionality of Receiver but does not have any routing capability.

**Feed Router** A router that is connected to Feed. It does not have the UDLR functionality and performs as a generic router.

**Receive Router** A router that is connected to Receiver It does not have the UDLR functionality and performs as a generic router.

Figure 7 shows how UDLR works in SOI-Asia network. Here Source is a host in a receive-only site and Destination is a host in Japan site, which is the upstream network of the receive-only site.

**1** Source sends an IP packet to Receive Router, which is the default router of Source. Receive Router forwards the IP packet to Feed Router.

**2** Satellite Receiver captures an Ethernet frame transmitted by Receive Router and encapsulate that frame in GRE header and IP header in which the source IP address is that of Satellite Receiver and the destination is LAN I/F of Satellite Feed.

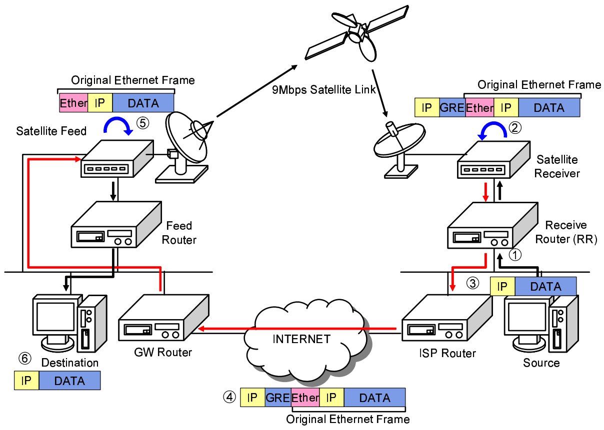


Figure 7: UDLR in SOI-Asia Network

**3 - 4** Satellite Receiver sends the IP packet to Satellite Feed via Receive Router i.e. the local ISP. Note that the IP packet has GRE encapsulated Ethernet frame as it payload.

**5** The encapsulated Ethernet frame reaches Satellite Feed, and is the original Ethernet frame was extracted. Satellite Feed forwards the Ethernet frame to Feed Router because it is destined to the MAC address of UDL I/F of Feed Router.

**6** Feed Router receives the Ethrenet frame and forwards the IP packet to Destination as if it was directly coming from the satellite UDL.

# 4 SOI-Asia Network Requirement

## 4.1 Hardware and Software

Hardware and software needed for installing SOI-Asia network are listed in the table 1 and 2 respectively. Software is available as ISO images at the specified sources, they should be downloaded and burned to CD-ROMs before start installation. Note that the hardware/software for SOI server, RPT client, WMT client, VIC/RAT and staff machine is separately explained in other documents.

# 5 Connecting Devices

Figure 8 shows a picture of how devices are connected in a receive-only site. By following instructions below, operator will connect hardware listed in the previous section 4.1 to resemble the figure.

**(STEP 1)** Connect satellite cable to UDLR satellite receiver.

- Confirm with satellite vendor that the dish is getting satellite signal and there is an F connector at the end of the cable.

Table 1: Hardware for SOI-Asia Network

| Hardware | Amount | Specification |
|---|---|---|
| Satellite Receiver | 1 | |
| PC (Receive Router) | 1 | PC/AT compatible<br>FreeBSD 4.x compliant<br>512MB RAM or more<br>2GHz CPU or faster<br>Two Ethernet interfaces with reliable chipsets, e.g. Intel (preferred) |
| Ethernet cables | 3 | Two 100BASE-TX straight cables<br>One 100BASE-TX crossover cable |
| Ethernet switch or hub | 1 | 10BASE-T or 100BASE-T<br>NOTE: If Ethernet switch is used, it should be multicast capable.<br>If an intelligent Ethernet switch is used, note that you have to disable IGMP snooping function to avoid the well-know problem regarding Multicast. |
| LAN Surge Protector | 1 | Port RJ45<br>10BT/100BT<br>2 input/output pairs or more |
| UPS | 1 | Surge protection capability<br>Online-type<br>3 outlets or more |
| Power Surge Protector | 2 | Each has 6 outlets or more |
| Console cable | 1 | |

Table 2: Software for SOI-Asia Network

| CD# | Software | Source |
|---|---|---|
| CD-1 | FreeBSD 4.9 Disk#1 | http://www.soi.wide.ad.jp/soi-asia/staff/download/4.9-i386-disc1.iso |
| CD-2 | FreeBSD 4.9 Disk#2 | http://www.soi.wide.ad.jp/soi-asia/staff/download/4.9-i386-disc2.iso |
| CD-6 | Routing | http://www.soi.wide.ad.jp/soi-asia/staff/download/SOI_Routing_CD.iso |

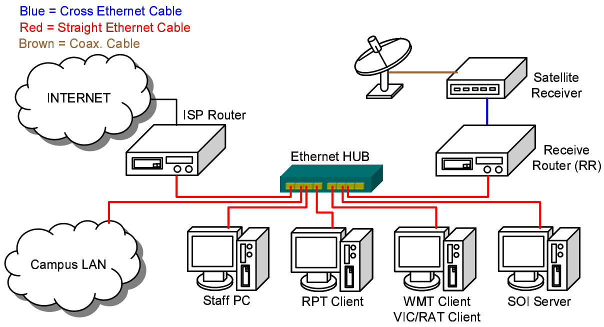


Figure 8: Connecting Devices in Receive-Only Site

- Connect F connector to the ANT IN port at the backside of UDLR satellite receiver.

**(STEP 2)** Connect UDLR satellite receiver to RR.

- Connect UDLR satellite receiver's Ethernet interface to one of RR's Ethernet interfaces by a crossover cable ¡no hub or switch required in between¿. Place the LAN surge protectors between both ends of cable and Ethernet ports.
- Connect another Ethernet interface of RR and Ethernet interfaces of other machines to a prepared hub/switch by Ethernet cables (e.g. 10BASE-T, 100BASE-T).

**(STEP 3)** Connect machines to LAN.

- Connect another interface of RR to the hub using straight cable.
- Connect local interface of ISP router to the hub using straight cable.
- Connect LAN interfaces of other machines to the hub using straight cable¡ this step can be done later when you are setting them up¿.

**(STEP 4)** Connect power cables.

- Connect UDLR satellite receiver's and RR's power cables to the UPS instead of connecting them directly to power line.
- Connect SOI-Asia server to the UPS instead of connecting them directly to power line.
- For all other machines and devices in SOI-Asia environment, connect their power cables to the power surge protector before going directly to power line.

Figure 1.6. Connecting devices in SOI Asia network

## 6 IPv4 address and Hostname Assignment

SOI-Asia network uses three sets of IPv4 addresses: SOI-Asia global address, local ISP's global address and campus private LAN address. SOI-Asia assigns one more IPv4 address to RR's UDL interface. The following URL gives information about IP address assignment to your organization.

**URL** http://www.soi.wide.ad.jp/project/soi-asia/sitesetup/soiasia-address-list.doc.

The table 3 describes, as example, how these IPv4 address sets are assigned to the subnets in each receive-only site.

Table 3: Example of IP Address Assignment in Receive Only Site

| Assigned From | Purpose | Type | Network Address |
|---|---|---|---|
| SOI-Asia | LAN | Global | 202.249.26.0/29 |
| SOI-Asia | UDL | Global | 202.249.25.199/27 |
| SOI-Asia | UDL | Private | 192.168.0.0/24 |
| Local ISP | LAN | Global | 133.27.16.0/30 |
| Campus | LAN | Private | 10.0.0.0/8 |

The table 4 describes how IP addresses are assigned to each machine in each receive-only site. Note that some machines such as Receive Router or SOI server have multiple IP addresses on one network interface.

Table 4: Example of IP Address Assignment to Each Host

| | SOI-Asia Global | SOI-Asia Private | Local ISP | Campus LAN |
|---|---|---|---|---|
| Satellite Receiver | | 192.168.0.1 | | |
| Receive Router(UDL) | 202.249.25.199 | 192.168.0.199 | | |
| Receive Router (LAN) | 202.249.26.1 | | 133.27.16.1 | |
| ISP Router | | 133.27.16.2 | | |
| SOI Server | 202.249.26.2 | | | 10.0.1.1 |
| WMT Client | 202.249.26.3 | | | |
| RPT Client | 202.249.26.4 | | | |
| VIC/RAT or Polycom | 202.249.26.5 | | | |
| Staff PC | 202.249.26.6 | | | |
| Other PCs | | | | 10.x.x.x |

The table 5 shows how hostnames are set on machines in the receive-only site assuming that its site name is "SFC".

## 7 Network Address Transration in SOI-Asia Network

Network Address Transration (NAT) is the application which convers the srouce address of IP datagrams from the original sender's private IP address to the router's global IP address on which NAT is activated. This application is introduced many enterprise

Table 5: Sample Hostname Setting

| Machine | Hostname |
|---|---|
| Satellite Receiver | (no hostname) |
| Receive Router | sfc-udl-recv.ai3.net |
| ISP Router | (no hostname for SOI-Asia) |
| SOI Server | sfc-soi.ai3.net |
| WMT Client | sfc-wmt.ai3.net |
| RPT Client | sfc-rpt.ai3.net |
| VIC/RAT or Polycom | sfc-vicrat.ai3.net |
| Other PCs | sfc-c1.ai3.net / sfc-c2.ai3.net / ... |

networks to reduce the usage of global IP addresses providing global connectivity to many nodes in the private networks. SOI-Asia network requires NAT for the different purpose from other enterprise networks.

As described in the section 3.4 and the table 4 in the section 6, GRE encapsulated packets will go through the Internet with its souce IP address is the private IP address of Satellite Receiver. However, it is often the case that packets origined by private IP address is dropped in ISPs. Also, if local ISP receives packets which seem not to be originated from their customer network, those packet could be dropped. Therefore, the GRE encapsulated packets from Sastellite Receivers could be dropped before they reach the end-point of LLTM. To avoid this problem, SOI-Asia installs NAT in Receive Router like the figure 9.

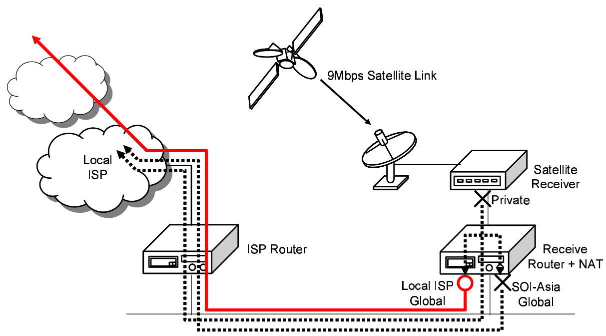


Figure 9: Activate NAT on Receive Router

NAT in Receiver Router converts the source address GRE packets from Satellite Receiver's private IP address to Receive Router's global IP address, which belongs not to SOI-Asia but to Local ISP. Then it could be expected that the GRE encapsulated packets originally from Satellite Receiver reach the Satellite Feed in SFC via Local ISP and the Internet.

# 8 SOI Asia RR Installation Lab Work

## 8.1 Kernel installation

In this step you will prepare a new kernel for RR

1. Copy the Generic kernel configuration to the SOI Asia RR kernel configuration.
   ```
   cd /usr/src/sys/i386/conf
   cp GENERIC SOIASIA
   ```
2. Edit the SOI Asia RR kernel, and add the following lines.
   ```
   options MROUTING
   options PIM
   options IPFIREWALL
   options IPFIREWALL_DEFAULT_TO_ACCEPT
   options IPDIVERT
   ```
3. Compile and install the SOI Asia RR kernel.
   ```
   cd ../../compile/SOIASIA
   make depend
   make
   make install
   ```
4. Confirm that kernel is installed.
5. Disable Zebra and XORP boot scripts.
   ```
   chmod -x /usr/local/etc/rc.d/zebra.sh
   chmod -x /usr/local/etc/rc.d/xorp.sh
   ```
6. After reboot, log on as `root` and confirm that the new kernel is running.
   ```
   uname -a
   ```

## 8.2 IPv4 network configuration

1. Set IPv4 network configuration in `/etc/rc.conf` IP addresses depend on the topology given by the instructor.
   ```
   hostname=sfc-rr.ai3.net
   ifconfig_rl0="inet 202.249.25.199 netmask 255.255.255.224 mtu 1452"
   ifconfig_rl0_alias0="192.168.0.199 netmask 255.255.255.0"
   ifconfig_rl1="inet 133.27.16.1 netmask 255.255.255.252"
   ifconfig_rl1_alias0="202.249.26.1 netmask 255.255.255.248"
   defaultrouter="202.249.25.193"
   static_routes="feedhost feedlltm"
   route_feedhost="-host 202.249.25.234 133.27.16.2"
   route_feedlltm="-host 202.249.25.236 133.27.16.2"
   gateway_enable="YES"

   inetd_enable="NO"
   kern_securelevel_enable="NO"
   nfs_reserved_port_only="NO"
   ```

```
sendmail_enable="NONE"
sshd_enable="YES"
usbd_enable="NO"
firewall_enable="YES"
firewall_type="OPEN"
```

2. Configure DNS resolver in `/etc/resolv.conf`

```
domain ai3.net
nameserver 202.249.24.34
nameserver 202.249.24.33
```

3. Configure NAT in `/usr/local/etc/rc.d/nat.sh`

```
#!/bin/sh
fwcmd="/sbin/ipfw"
natcmd="/sbin/natd"
localip="133.27.16.1"
localif="fxp1"

case "$1" in
start)
        ${fwcmd} add 50 divert natd gre from any to any via ${localif}

        ${natcmd}  -n ${localif} -redirect_proto gre ${localip}
        ;;
stop)
        ${fwcmd} delete 50
        killall ${natcmd}
        ;;
*)
        echo "Usage: `basename $0` {start|stop}" >&2
        ;;
esac

exit 0
```

4. Reboot your router.

## 8.3 Verify the IPv4 connectivity

1. Logon as root.
2. Check the network interfaces.
   `ifconfig -a`
3. Check the routing table.
   `nestat -nrf inet`
4. Check connectivity to UDLR receiver.
   `ping -c 10 192.168.0.1`
5. Check connectivity to your ISP router.
   `ping -c 10 133.27.16.2`

6. Check connectivity to UDL feed.
   `ping -c 10 202.249.25.193`

7. Check the operation of NAT.
   `tcpdump -npi rl0 proto gre`
   You should see some GRE packets, then stop `tcpdump`.
   Run `tcpdump` on the rl1 interface.
   `tcpdump -npi rl1 proto gre`
   Check whether the source addresses of GRE packets are changed into the ISP address of the router.

## 8.4 Configure PIM-SMv4 using XORP

You will edit the XORP configuration you created earlier to follow the SOI Asia network design.

1. Edit XORP configuration file.
   `vi /usr/local/xorp/config.boot`

2. Comment out all IPv6 related configuration sections: `mfea6`, `mld`, and `pimsm6`.

3. Comment out the static RP configuration section of PIM-SM: `static-rps`

4. Save the configuration file.

5. Run XORP to confirm your configuration.
   `/usr/local/xorp/bin/xorp_rtrmgr &`
   Fix your configuration file if you find errors.

6. Use another console to access XORP CLI.
   `/usr/local/xorp/bin/xorpsh`

7. Check the results of the following commands.
   `show pim interface`
   `show pim neighbors`

8. The instructor runs a multicast stream, and you receive the multicast stream.
   Use another console, and run `mtest`.
   Join group 233.18.109.151.
   `j 233.18.109.151 202.249.26.1`

9. Check whether you receive the stream.
   Go back to the console accessing XORP CLI, and do the following commands.
   `show pim join`
   `show pim mfc`
   Analyze your findings.

10. Quit from XORP CLI.
    `quit`

11. Check the kernel multicast forwarding cache.
    `netstat -ngf inet`

12. Check whether IP multicast traffic flows out of rl1 interface.
    `tcpdump -npi rl1 ip multicast`

13. After you confirmed that everything is OK, enable the XORP boot script.
`chmod +x /usr/local/etc/rc.d/xorp.sh`

14. Stop XORP to prepare for the next steps.
`/usr/local/etc/rc.d/xorp.sh stop`

## 8.5 IPv6 network configuration

1. Set IPv6 network configuration in `/etc/rc.conf` IPv6 addresses depend on the topology given by the instructor.

```
ipv6_enable="YES"
ipv6_gateway_enable="YES"
ipv6_ifconfig_rl0="2001:0D30:0101:5::b:b prefixlen 64"
ipv6_ifconfig_rl1="2001:0D30:010b::10b:1 prefixlen 64"
rtadvd_enable="YES"
rtadvd_interfaces="rl1"
```

2. Reboot your router.

3. After reboot, logon as `root`.

4. Check the network interfaces.
`ifconfig -a`

5. Check IPv6 routing table.
`netstat -nrf inet6`

6. Check connectivity to other routers on UDL.
`ping6 ff02::2%rl0`

## 8.6 Configure IPv6 unicast routing with Zebra

You will edit Zebra and Ospf6d configurations to follow the SOI Asia network design.

1. Confirm your Zebra configuration. You don't need to edit it.
`vi /usr/local/etc/zebra.conf`

2. Edit your Ospf6d configuration.
`vi /usr/local/etc/ospf6d.conf`

3. Change interface costs and priorities.
Interface cost is 50. The priority is 0 for the interface toward UDL, and 100 for the interface toward your site.

```
interface rl0
 ipv6 ospf6 cost 50
 ipv6 ospf6 priority 0
!
interface rl1
 ipv6 ospf6 cost 50
 ipv6 ospf6 priority 100
```

4. Change the router ID to the IPv4 address of the interface toward your site.

```
router ospf6
 router-id 202.249.26.1
 interface fxp0 area 0.0.0.0
 interface fxp1 area 0.0.0.0
```

5. Save your configuration.

6. Run Zebra and Ospf6d.
   `/usr/local/sbin/zebra -d`
   `/usr/local/sbin/ospf6d -d`
   Your Zebra and Ospf6d should be running if you don't see any error messages.

7. Confirm that Zebra and Ospf6d proceses are running.
   `ps -ax`

8. Access Ospf6d CLI and run the following commands.
   `show ipv6 ospf6 interface`
   `show ipv6 ospf6 neighbor`
   `show ipv6 ospf6 route`

9. Quit from Ospf6d CLI.

10. Check the IPv6 routing table.
    `netstat -nrf inet6`
    You should see many entries.

11. Edit the Zebra boot script.
    `vi /usr/local/etc/rc.d/zebra.sh`
    SOI Asia network doesn't use OSPF on UDL. You have to delete the lines to run and stop Ospfd.

12. Activate the Zebra boot script.
    `chmod +x /usr/local/etc/rc.d/zebra.sh`

## 8.7 Configure PIM-SMv6 using XORP

You will edit the XORP configuration to follow the SOI Asia network design for PIM-SMv6

1. Edit XORP configuration file.
   `vi /usr/local/xorp/config.boot`

2. Remove the comment from all IPv6 related configuration sections: `mfea6`, `mld`, and `pimsm6`.

3. Comment out the static RP configuration section of PIM-SMv6: `static-rps`

4. Save the configuration file.

5. Stop XORP that is currently running.
   `/usr/local/etc/rc.d/xorp.sh stop`

6. Run XORP to confirm your configuration.
   `/usr/local/xorp/bin/xorp_rtrmgr &`
   Fix your configuration file if you find errors.

7. Use another console to access XORP CLI.
   `/usr/local/xorp/bin/xorpsh`

8. Check the results of the following commands.
   `show pim6 interface`
   `show pim6 neighbors`

9. The instructor runs a multicast stream, and you receive the multicast stream.
   Use another console, and run `mtest`.
   Join group ff08::1515.
   `j ff08::1515 rl1`

10. Check whether you receive the stream.
    Go back to the console accessing XORP CLI, and do the following commands.
    `show pim6 join`
    `show pim6 mfc`
    Analyze your findings.

11. Quit from XORP CLI.
    `quit`

12. Check the kernel multicast forwarding cache.
    `netstat -ngf inet6`

13. Check whether IP multicast traffic flows out of rl1 interface.
    `tcpdump -npi rl1 ip6 multicast`

## 8.8 Final check

You will check whether every configurations work correctly.

1. Reboot your router.
2. After reboot, logon as `root`.
3. Check whether all processes are running.
4. Check the network interfaces.
5. Check the routing table for IPv4 and IPv6.
6. Check the multicast operation.