IP Forwarding

- Routing (Forwarding)
- Static routing
- Subnetting

Orientation

- IP (Internet Protocol) is a Network Layer Protocol.
Routing of IP Packets

- There are two parts to routing IP packets:
  1. How to pass a packet from an input interface to the output interface of a router (“IP forwarding”)?
  2. How to find and setup a route (“IP routing”)?

- IP Forwarding is performed by IP (in the OS kernel)
- IP Routing is performed by a user-level process (In Unix, by the daemon processes routed or gated)
IP Forwarding

• A packet is typically forwarded to a large number of routers before reaching the destination host.

• IP forwarding is done on a hop-by-hop basis, i.e., no one knows the complete route. The goal of forwarding is to bring the IP datagram closer to the destination.

Major Tenets for IP Forwarding

• Every IP datagram contains the IP address of the destination host.

• The network prefix of an IP address uniquely identifies a single physical network that is part of the larger Internet.

• All hosts and routers that have the same network prefix are connected to the same physical network and can directly communicate by sending MAC layer frames.

• Every physical network of the Internet has at least one router, which is also connected to at least one other physical network.
IP Forwarding

- IP forwarding is performed by both hosts and routers.
- However, a host never passes a packet from an input interface to an output interface.
- Both routers and hosts have a routing table. Routing table entries look like this and is looked up for each datagram:

<table>
<thead>
<tr>
<th>Destination IP address</th>
<th>IP address of Next-Hop-Router</th>
<th>Flags</th>
<th>Specification of an interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>Host or network address</td>
<td>Address of router or directly connected network</td>
<td></td>
<td>Interface to which the datagram is passed</td>
</tr>
</tbody>
</table>

Routing Table Lookup

- For each IP packet, there is one routing table lookup.
  1. Find matching host address
  2. Find matching network address
  3. Find default entry

- Routing table printout with `netstat -rn`
- Example:

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>Flags</th>
<th>Refcnt</th>
<th>Use</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>128.143.0.0</td>
<td>128.143.137.16</td>
<td>U</td>
<td>3</td>
<td>1697</td>
<td>hme0</td>
</tr>
<tr>
<td>224.0.0.0</td>
<td>128.143.137.16</td>
<td>U</td>
<td>3</td>
<td>0</td>
<td>hme0</td>
</tr>
<tr>
<td>default</td>
<td>128.143.137.1</td>
<td>UG</td>
<td>0</td>
<td>208</td>
<td>hme0</td>
</tr>
<tr>
<td>127.0.0.1</td>
<td>127.0.0.1</td>
<td>UH</td>
<td>0</td>
<td>11424</td>
<td>lo0</td>
</tr>
</tbody>
</table>
Routing Table

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>Flags</th>
<th>Refcnt</th>
<th>Use</th>
<th>Interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>128.143.0.0</td>
<td>128.143.137.16</td>
<td>U</td>
<td>3</td>
<td>1697</td>
<td>hme0</td>
</tr>
<tr>
<td>224.0.0.0</td>
<td>128.143.137.16</td>
<td>U</td>
<td>3</td>
<td>0</td>
<td>hme0</td>
</tr>
<tr>
<td>default</td>
<td>128.143.137.1</td>
<td>UG</td>
<td>0</td>
<td>208</td>
<td>hme0</td>
</tr>
<tr>
<td>127.0.0.1</td>
<td>127.0.0.1</td>
<td>UH</td>
<td>0</td>
<td>11424</td>
<td>lo0</td>
</tr>
</tbody>
</table>

• **Flags:**
  - **U** Route is up
  - **G** Route is to a router (“Gateway”)
  - **H** Route is to a host (and destination is host address)
  - **D** Route is created by an ICMP redirect
  - **M** Route is modified by an ICMP redirect

• **RefCnt:**
  Counts the number of active uses of the route

• **Use:**
  Number of packets sent on this route

• **Interface:**
  Outgoing interface
Forwarding Functions at Host

- Forwarding an IP datagram at a host:
  ```
  DestNet = Network id in destination IP address
  If (DestNet == my Network id)
  Deliver datagram to destination directly;
  else
  Send datagram to default router;
  ```

- Reception of an IP datagram at a host:
  ```
  Datagram is received on an interface
  If (Destination IP address == (my IP address || broadcast address))
  Deliver datagram to higher layer;
  else
  Discard the datagram;
  ```

Forwarding Functions at Router

- Reception of an IP datagram at a router:
  ```
  Packet is received on an interface
  If (Destination IP address != my IP address)
  Perform datagram forwarding;
  ```

- Forwarding an IP datagram at a router:
  ```
  DestNet = Network id in destination IP address
  DestIP = destination IP address
  If (DestIP == Destination IP address in my routing table)
  Deliver datagram to the next-hop-router in the table;
  elseif (DestNet == Network id in my routing table)
  Deliver datagram to destination of the interface;
  else
  deliver packet to default router;
  ```
Subnetting: Host Processing

- Each host has to know its IP address and the subnet mask.
- For each IP datagram the host performs:

```java
if ((Destination IP address & Subnet Mask) ==
    (My IP address & Subnet Mask))
    Deliver datagram directly;
else
    Forward datagram to default router;
```

Subnetting: Router Processing

- A router has a table:
  ```
  < SubnetNumber, SubnetMask, NextHop>
  » SubnetNumber is the <network id, subnet id>
  » SubnetMask is the subnet mask
  » NextHop is the IP address of a router
  ```
- The router performs the following algorithm:

```c
dest = destination IP address;
for each entry <SubnetNumber, SubnetMask, NextHop>
    sub = dest & SubnetMask;
    if (sub == SubnetNumber)
        Deliver datagram directly;
    else
        Forward datagram to NextHop;
```
Statically Setting IP Routing Tables

• There are several ways for setting IP routing tables without a routing protocol (\textit{Static Routing})

1. Automatic creation of entry during initialization of a local interface (with \texttt{ifconfig})
   – Whenever an IP address is set with config, a route is created for that interface
2. With \texttt{route} command
3. Via \texttt{ICMP} redirect messages
4. Via \texttt{ICMP router advertisement/router discovery} messages

Route Command

• Route commands are put in a system file that is read during system bootstrap
• System file is: \texttt{/etc/rc2.d/S69inet} in Solaris
  \texttt{/etc/netstart} in FreeBSD

• Route commands can be issues from command line interface
  (The route entries disappear when system is rebooted)
• Example:

  \begin{verbatim}
  route add 192.0.1.3 1 default
  route add slip bsdi 1
  \end{verbatim}
Route Command

- **Add/delete/change**: Add, delete or change a route
  - Examples:
    ```
    route add -net 192.0.2.0 192.0.4.4
    route change -host 192.0.2.6 192.0.5.4
    route delete -host 192.0.2.6 192.0.5.4
    ```

- **Flush**: Remove all gateway entries from the routing table.
  - Example:
    ```
    route flush
    ```

Route Command

- **Get**: Lookup and display the route for a destination.
  - Example:
    ```
    # route get 192.0.1.0
    
    route to: 192.0.1.0
    destination: 192.0.1.0
    mask: 255.255.255.0
    interface: epl
    flags: <UP,DONE,CLONING>
    ```

- **Monitor**: Continuously report any changes to the routing information base, routing lookup misses, or suspected network partitionings
  ```
  # route monitor
  ```
Advantages of Subnetting

- Improves efficiency of IP addresses by not consuming an entire Class B or Class C address for each physical network
- Reduces router complexity. Since external routers do not know about subnetting, the complexity of routing tables is reduced
- Networks become easier to manage
- Note that there are three levels of aggregation:
  - Network
  - Subnet
  - Host

Forwarding Functions at Router

- Reception of an IP datagram at a router:
  
  ```
  Packet is received on an interface
  If (Destination IP address != my IP address)
  Perform datagram forwarding;
  ```

- Forwarding an IP datagram at a router:
  
  ```
  DestNet = Network id in destination IP address
  DestIP = destination IP address
  If (DestIP == Destination IP address in my routing table)
  Deliver datagram to the next-hop-router in the table;
  elseif (DestNet == Network id in my routing table)
  Deliver datagram to destination of the interface;
  else
  deliver packet to default router;
  ```
ICMP Redirect

- When a router detects that an IP datagram should have gone to a different router, the router (Router 1)
  - forwards the IP datagram to the correct router
  - sends an ICMP redirect message to the host
- Host uses ICMP message to update its routing table

![ICMP Redirect Diagram]

ICMP Router Solicitation
ICMP Router Advertisement

- Specified in RFC 1256 (1991)
- After bootstrapping a router broadcasts an ICMP router solicitation message.
- In response, routers send an ICMP router advertisement message
- Also, routers periodically broadcast ICMP router advertisement