ARP
A’s perspective
### A’s ARP request

#### Routing Table

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>Mask</th>
<th>Iface</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.5.40</td>
<td>192.168.5.40</td>
<td>255.255.255.255</td>
<td>eth1</td>
</tr>
<tr>
<td>172.64.3.10</td>
<td>172.64.3.10</td>
<td>255.255.255.255</td>
<td>eth2</td>
</tr>
<tr>
<td>10.0.3.32</td>
<td>10.0.3.32</td>
<td>255.255.255.255</td>
<td>eth3</td>
</tr>
</tbody>
</table>

#### Router Interfaces

<table>
<thead>
<tr>
<th>Interface</th>
<th>HWaddr</th>
<th>inet addr</th>
</tr>
</thead>
<tbody>
<tr>
<td>eth3</td>
<td>11:22:33:44:55:66</td>
<td>10.0.3.1</td>
</tr>
<tr>
<td>eth2</td>
<td>e6:74:1f:1b:44:33</td>
<td>172.64.3.1</td>
</tr>
<tr>
<td>eth1</td>
<td>AA:BB:CC:DD:EE:FF</td>
<td>192.168.5.1</td>
</tr>
</tbody>
</table>

#### Incoming Message

**ETHERNET header:**
- type: 2054

**ARP header**
- hardware type: 1
- protocol type: 2048
- hardware address length: 6
- protocol address length: 4
- opcode: 1
- sender hardware address: AA:AA:AA:11:11:11
- sender ip address: 10.0.3.32
- target hardware address: 00:00:00:00:00:00
- target ip address: 10.0.3.1
R’s Perspective (upon receipt of request)

Stores A's mac address as it can assume A sent an ARP request to initialize a conversation

Mac address A is interested in
ARP request received

ARP request:
- If Addressed to Router
  - Add ip / mac addr of sender to ARP cache
  - Send ARP reply
- Otherwise
  - Ignore

**Router Interfaces**

<table>
<thead>
<tr>
<th>Interface</th>
<th>HWaddr</th>
<th>inet addr</th>
</tr>
</thead>
<tbody>
<tr>
<td>eth3</td>
<td>11:22:33:44:55:66</td>
<td>10.0.3.1</td>
</tr>
<tr>
<td>eth2</td>
<td>e6:74:1f:1b:44:33</td>
<td>172.64.3.1</td>
</tr>
<tr>
<td>eth1</td>
<td>AA:BB:CC:DD:EE:FF</td>
<td>192.168.5.1</td>
</tr>
</tbody>
</table>

**Incoming Message**

ETHERNET header:
- type: 2054

ARP header
- hardware type: 1
- protocol type: 2048
- hardware address length: 6
- protocol address length: 4
- opcode: 1
- sender hardware address: AA:AA:AA:11:11:11
- sender ip address: 10.0.3.32
- target hardware address: 00:00:00:00:00:00
- target ip address: 10.0.3.1

(Broadcast address)
ARP Reply

Router Interface Request Received
eth3
inet addr 10.0.3.1

**Reply Message**
ETHERNET header:
- type: 2054
ARP header
- hardware type: 1
- protocol type: 2048
- hardware address length: 6
- protocol address length: 4
- opcode: 2
- sender ip address: 10.0.3.1
- target hardware address: AA:AA:AA:11:11:11
- target ip address: 10.0.3.32

**Incoming Message**
ETHERNET header:
- type: 2054
ARP header
- hardware type: 1
- protocol type: 2048
- hardware address length: 6
- protocol address length: 4
- opcode: 1
- sender hardware address: AA:AA:AA:11:11:11
- sender ip address: 10.0.3.32
- target hardware address: 00:00:00:00:00:00
- target ip address: 10.0.3.1
ARP Reply

Router Interface Request Received On eth3
inet addr 10.0.3.1

Reply Message
ETHERNET header:
  type: 2054
ARP header
  hardware type: 1
  protocol type: 2048
  hardware address length: 6
  protocol address length: 4
  opcode: 2
  sender hardware address: AA:AA:AA:11:11:11
  sender ip address: 10.0.3.32
  target hardware address: AA:AA:AA:11:11:11
  target ip address: 10.0.3.1

Incoming Message
ETHERNET header:
  type: 2054
ARP header
  hardware type: 1
  protocol type: 2048
  hardware address length: 6
  protocol address length: 4
  opcode: 1
  sender hardware address: AA:AA:AA:11:11:11
  sender ip address: 10.0.3.32
  target hardware address: 00:00:00:00:00:00
  target ip address: 10.0.3.1
ARP Reply

**Router Interface Request Received On**
eth3
inet addr 10.0.3.1

**Reply Message**
ETHERNET header:
  type: 2054
ARP header
  hardware type: 1
  protocol type: 2048
  hardware address length: 6
  protocol address length: 4
  opcode: 2
  sender ip address: 10.0.3.1
  target hardware address: AA:AA:AA:11:11:11
  target ip address: 10.0.3.32

**Incoming Message**
ETHERNET header:
  type: 2054
ARP header
  hardware type: 1
  protocol type: 2048
  hardware address length: 6
  protocol address length: 4
  opcode: 1
  sender hardware address: AA:AA:AA:11:11:11
  sender ip address: 10.0.3.32
  target hardware address: 00:00:00:00:00:00
  target ip address: 10.0.3.1
A’s perspective

Mac address returned in reply

A now has all it needs to send IP packets with first hop at 10.0.3.1
### Incoming IP packet

#### Routing Table

<table>
<thead>
<tr>
<th>Destination</th>
<th>Gateway</th>
<th>Mask</th>
<th>Iface</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.5.40</td>
<td>192.168.5.40</td>
<td>255.255.255.255</td>
<td>eth1</td>
</tr>
<tr>
<td>172.64.3.10</td>
<td>172.64.3.10</td>
<td>255.255.255.255</td>
<td>eth2</td>
</tr>
<tr>
<td>10.0.3.32</td>
<td>10.0.3.32</td>
<td>255.255.255.255</td>
<td>eth3</td>
</tr>
</tbody>
</table>

#### Router Interfaces

<table>
<thead>
<tr>
<th>Interface</th>
<th>HWaddr</th>
<th>inet addr</th>
</tr>
</thead>
<tbody>
<tr>
<td>eth3</td>
<td>11:22:33:44:55:66</td>
<td>10.0.3.1</td>
</tr>
<tr>
<td>eth2</td>
<td>e6:74:1f:1b:44:33</td>
<td>172.64.3.1</td>
</tr>
<tr>
<td>eth1</td>
<td>AA:BB:CC:DD:EE:FF</td>
<td>192.168.5.1</td>
</tr>
</tbody>
</table>

#### Incoming Message

**ETHERNET header:**
- type: 2048

**IP header:**
- version: 4
- header length: 5
- type of service: 0
- length: 84
- id: 13429
- fragment flag: DF
- fragment offset: 0
- TTL: 63
- protocol: 1
- checksum: 9785
- source: 10.0.3.32
- destination: 192.168.5.40

**ICMP header:**
- type: 8
- code: 0
- checksum: 6971
From the routing table R sees it should forward packet out Eth1 to 192.168.5.40

No entry for 192.168.5.40 in IP/Mac Addr table, so R will need to send ARP request out ETH1 and store the packet until a reply is received.
**ARP Request**

**Outgoing Router Interface**
- Outgoing Router Interface: `eth1`
- inet addr: `192.168.5.1`

**Incoming IP Packet**
- ETHERNET header:
  - type: `2048`
- IP header:
  - version: `4`
  - header length: `5`
  - type of service: `0`
  - length: `84`
  - id: `13429`
  - fragment flag: `DF`
  - fragment offset: `0`
  - TTL: `63`
  - protocol: `1`
  - checksum: `9785`
  - source: `10.0.3.32`
  - destination: `192.168.5.40`
- ICMP header:
  - type: `8`
  - code: `0`
  - checksum: `6971`

**Outgoing ARP Request**
- ETHERNET header:
  - type: `2054`
- ARP header:
  - hardware type: `1`
  - protocol type: `2048`
  - hardware address length: `6`
  - protocol address length: `4`
  - opcode: `1`
  - sender hardware address: `AA:BB:CC:DD:EE:FF`
  - sender ip address: `192.168.5.1`
  - target hardware address: `00:00:00:00:00:00`
  - target ip address: `192.168.5.40`
Continuously loop over outstanding requests:

If (current time - request time) > 1s:

If requests already sent count > 5:

For packet waiting on request:

Send out ICMP: ‘host unreachable’ to packet origin

Else:

Resend request, reset sent time stamp, and increment requests sent counter
R’s Perspective upon receipt of B’s reply