PART V

MAPPING INTERNET ADDRESSES TO PHYSICAL ADDRESSES (ARP)
Motivation

- Must use hardware (physical) addresses to communicate over network
- Applications only use Internet addresses
Example

- Computers A and B on same network
- Application on A generates packet for application on B
- Protocol software on A must use B’s hardware address when sending a packet
REQUEST
ARP REQ
I AM IP A
MY HW = HWA
LOOKING FOR IP B

BROADCAST ON MY NETWORK
REPLY

AR? REPLY

I AM JP3

MY HW HWB

SENDING TO IPA

HW QA HWQ

UNICAST TO A
Consequence

- Protocol software needs a mechanism that maps an IP address to equivalent hardware address
- Known as *address resolution* problem
Address Resolution

- Performed at each step along path through Internet
- Two basic algorithms
  - Direct mapping
  - Dynamic binding
- Choice depends on type of hardware
Direct Mapping

- Easy to understand
- Efficient
- Only works when hardware address is small
- Technique: assign computer an IP address that encodes the hardware address
Example Of Direct Mapping

- Hardware: proNet ring network
- Hardware address: 8 bits
- Assume IP address 192.5.48.0 (24-bit prefix)
- Assign computer with hardware address $K$ an IP address $192.5.48.K$
- Resolving an IP address means extracting the hardware address from low-order 8 bits
Example Of Direct Mapping

- Hardware: proNet ring network
- Hardware address: 8 bits
- Assume IP address 192.5.48.0 (24-bit prefix)
- Assign computer with hardware address K an IP address 192.5.48.K
- Resolving an IP address means extracting the hardware address from low-order 8 bits
Dynamic Binding

- Needed when hardware addresses are large (e.g., Ethernet)
- Allows computer A to find computer B’s hardware address
  - A starts with B’s IP address
  - A knows B is on the local network
- Technique: broadcast query and obtain response
- Note: dynamic binding only used across one network at a time
Internet Address Resolution Protocol (ARP)

- Standard for dynamic address resolution in the Internet
- Requires hardware broadcast
- Intended for LAN
- Important idea: ARP only used to map addresses within a single physical network, never across multiple networks
ARP

- Machine A broadcasts ARP request with B’s IP address
- All machines on local net receive broadcast
- Machine B replies with its physical address
- Machine A adds B’s address information to its table
- Machine A delivers packet directly to B
Illustration Of ARP Request And Reply Messages

A broadcasts request for B (across local net only)

B replies to request
ARP Packet Format When Used With Ethernet

```
 0  4  8  16  24
```

<table>
<thead>
<tr>
<th>Field</th>
<th>Length</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ethernet Address Type</td>
<td>1</td>
<td>0x8000 for Ethernet</td>
</tr>
<tr>
<td>IP Address Type</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Eth Addr Len</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>IP Addr Len</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Operation</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Sender’s Eth Addr (first 6 octets)</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Sender’s Eth Addr (last 2 octets)</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Sender’s IP Addr (first 2 octets)</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Sender’s IP Addr (last 2 octets)</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Target’s Eth Addr (first 2 octets)</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Target’s Eth Addr (last 4 octets)</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Target’s IP Addr (all 4 octets)</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>
```
# ARP Packet Format When Used With Ethernet

<table>
<thead>
<tr>
<th>0</th>
<th>8</th>
<th>16</th>
<th>31</th>
</tr>
</thead>
<tbody>
<tr>
<td>ETHERNET ADDRESS TYPE (1)</td>
<td>IP ADDRESS TYPE (0800)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ETH ADDR LEN (6)</td>
<td>IP ADDR LEN (4)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SENDER’S ETH ADDR (first 4 octets)</td>
<td>SENDER’s IP ADDR (first 2 octets)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SENDER’S ETH ADDR (last 2 octets)</td>
<td>SENDER’s IP ADDR (last 2 octets)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TARGET’S ETH ADDR (first 2 octets)</td>
<td>TARGET’S ETH ADDR (last 4 octets)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TARGET’S IP ADDR (all 4 octets)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**ARP REQ**

- **HWA**
- **IPR (2)**
- **BLANK**
ARP Packet Format When Used With Ethernet

<table>
<thead>
<tr>
<th>0</th>
<th>8</th>
<th>16</th>
<th>31</th>
</tr>
</thead>
<tbody>
<tr>
<td>ETHERNET ADDRESS TYPE (1)</td>
<td>IP ADDRESS TYPE (0800)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ETH ADDR LEN (6)</td>
<td>IP ADDR LEN (4)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SENDER'S ETH ADDR (first 4 octets)</td>
<td>SENDER'S IP ADDR (last 2 octets)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SENDER'S ETH ADDR (last 2 octets)</td>
<td>TARGET'S ETH ADDR (first 2 octets)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SENDER'S IP ADDR (last 2 octets)</td>
<td>TARGET'S ETH ADDR (last 4 octets)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TARGET'S IP ADDR (all 4 octets)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Observations About Packet Format

- General: can be used with
  - Arbitrary hardware address
  - Arbitrary protocol address (not just IP)
- Variable length fields (depends on type of addresses)
- Length fields allow parsing of packet by computer that does not understand the two address types
Retention Of Bindings

- Cannot afford to send ARP request for each packet
- Solution
  - Maintain a table of bindings
- Effect
  - Use ARP one time, place results in table, and then send many packets
ARP Caching

- ARP table is a cache
- Entries time out and are removed
- Avoids stale bindings
- Typical timeout: 20 minutes
Algorithm For Processing
ARP Requests

- Extract sender’s pair, (IA, EA) and update local ARP table if it exists
- If this is a request and the target is “me”
  - Add sender’s pair to ARP table if not present
  - Fill in target hardware address
  - Exchange sender and target entries
  - Set operation to reply
  - Send reply back to requester
Algorithm Features

- If A ARPs B, B keeps A’s information
  - B will probably send a packet to A soon

- If A ARPs B, other machines do not keep A’s information
  - Avoids clogging ARP caches needlessly
Conceptual Purpose Of ARP

- Isolates hardware address at low level
- Allows application programs to use IP addresses
ARP Encapsulation

- ARP message travels in data portion of network frame
- We say ARP message is *encapsulated*
Illustration Of ARP Encapsulation

ARP MESSAGE

FRAME HEADER

FRAME DATA AREA

ARP REQUEST

ARP REPLY

ARP REQUEST

ARP REPLY
Ethernet Encapsulation

- ARP message placed in frame data area
- Data area padded with zeroes if ARP message is shorter than minimum Ethernet frame
- Ethernet type 0x0806 used for ARP
Reverse Address Resolution Protocol

- Maps Ethernet address to IP address
- Same packet format as ARP
- Intended for bootstrap
  - Computer sends its Ethernet address
  - RARP server responds by sending computer’s IP address
- Seldom used (replaced by DHCP)
Summary

- Computer’s IP address independent of computer’s hardware address
- Applications use IP addresses
- Hardware only understands hardware addresses
- Must map from IP address to hardware address for transmission
- Two types
  - Direct mapping
  - Dynamic mapping
Summary
(continued)

- Address Resolution Protocol (ARP) used for dynamic address mapping
- Important for Ethernet
- Sender broadcasts ARP request, and target sends ARP reply
- ARP bindings are cached
- Reverse ARP was originally used for bootstrap