SIMPLE ROUTER – PROJECT 2

RECAP

- We’re writing a router in C
- We’re working with a virtual network topology (VNS)
- The router will route real IP packets from standard clients like ping and traceroute
- It’s due on 7th December! If you haven’t started yet, Get Started!

Protocols to be handled

- Ethernet - All packets given to you are Ethernet frames
- IP
- ARP
  - Needed to resolve IP addresses to MAC addresses
- ICMP request/replies
  - Used by some programs to send requests (ping)
  - Needed to send control messages back to the host
- See sr_protocol.h for bit level details
- Review your pointer arithmetic

CheckSums

- IP
  - Need to perform Checksum for all IP packet headers. If checksum fails, drop the packet.
- ICMP
  - Need to validate Checksum for all packets destined for the router.
  - Need to calculate for outgoing packets
  - Ignore if forwarding
- TCP/UDP
  - End-to-End checksum, ignore
- use cksum method is sr_utils.c

Generating ARP request

- Request: Who has IP 192.168.1.3?

- Create ARP request with fields:
  Source HW addr: MACsrc
  Source protocol addr: IPsrc
  Target HW addr: FF:FF:FF:FF:FF:FF
  Target protocol addr: 192.168.1.3

- ARP requests are sent to the Ethernet broadcast address

Handling ARP request

- Get request: Who has 192.168.1.3

- If one of the IPs of the router is 192.168.1.3, send an ARP reply:
  I have IP of 192.168.1.3 with MAC address of 00-11-22-33-44-55-66.

- Create ARP reply with fields:
  - Source HW addr: 00-11-22-33-44-55-66
  - Source protocol address: 192.168.1.3
  - Target HW addr: MACsrc
  - Target protocol addr: IPsrc

ARP reply is sent directly to MACsrc

Handling ARP reply

- Reply : I have IP of 192.168.1.3 with MAC address of ...
- If the target IP of the ARP reply is the IP of the interface this came in on:

  Add the IP to MAC mapping to the ARP cache

  Forward any packets that were queued on the this ARP request

ARP Cache Class

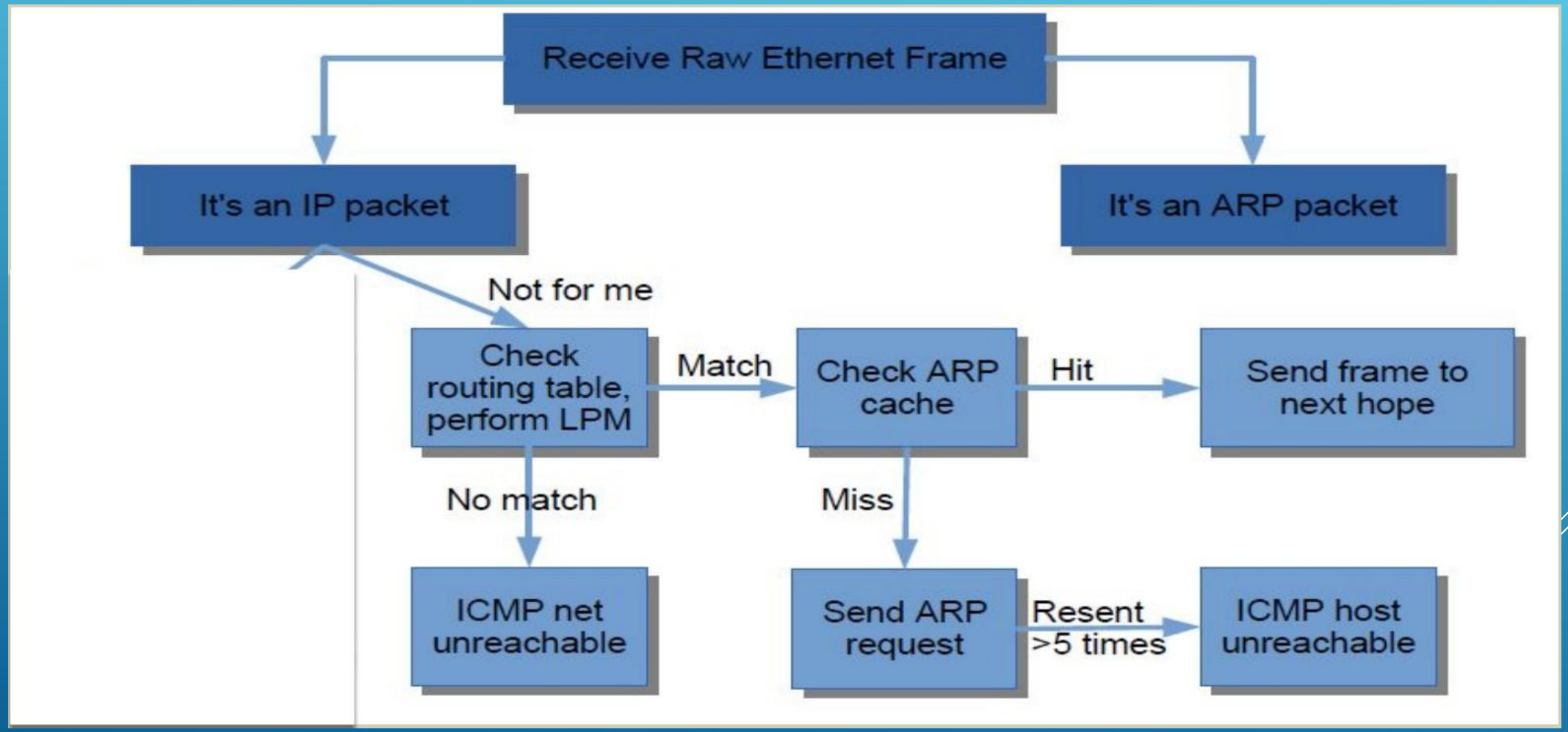
- You're provided with an ARP Cache Class containing :
  - An ARP request queue
  - An ARP cache

- ARP cache entries time out automatically after 15 seconds

- The cache class is essentially two linked lists: one for the cache, and one for requests. You've also been provided with functions to handle querying and inserting into these two lists.

- See pseudocode in sr_arpcache.h for more detailed infomation

IP FORWARDIN



CREATING ARP REQUESTS

- If LPM entry (type sr_rt) is found, then reduce TTL and update checksum for the IP header. Now, you need to update the frame header's source and destination fields.
- Do a sr_arpcache_lookup. 'gw.s_addr' (next hop IP address) is one of the variables to be passed to the function.
- If it returns NULL, use sr_arpcache_queuereq function to add the ARP request to the ARP request queue. Send the sr instance and the queue to handle_arpreq. Handle_arpreq - Function to be implemented. Check comments in sr_arpcache.h for pseudocode.
- If the lookup returned an arp entry, then modify the Ethernet source and destination values and use sr_send_packet.

LPM – LONGEST PREFIX MATCH

- If packet not destined to router and ttl != 1, check the routing table to see if a matching entry for the destination IP address exists (LPM)
- You have the routing table (sr->routing_table) and destination IP address. The routing_table is a structure of type sr_rt (defined in sr_rt.h). The routing table has dest and mask variables of type in_addr.
- Do & (bitwise AND) between dest and mask. Also, between the destination IP address and mask. Compare to see if they match.
- If multiple matches, check to see which match has the longest mask.s_addr.

ICMP TYPE 11

- Verify that the packet is not destined to the router. How?
- Check if the destination IP address of the packet is not equal to the IP addresses of the router interfaces.
- If not destined and TTL of the IP header packet == 1, create ICMP type 11 (time exceeded) packet.
- ICMP Type 11 structure is already defined for you.
- Then, send the packet using sr_send_packet. The data field in the ICMP segment is 28 bytes starting from the IP header of the original packet which the router received.

POINTS:

- If LPM returns empty, then create ICMP type 3 (network unreachable) packet.
- Note: ICMP Type 11 structure can be used for this as they are similar.
- Then, send the packet using sr_send_packet.

TO TEST YOUR LPM LOGIC

- To test your routing table lookup, you can change the entries in rtable file to have different subnet masks.
- For instance, to check your router's longest prefix match logic, you can have multiple entries in your rtable file that match a single destination IP.
- The following routing table has two matching entries for the packet destined to 192.168.2.2.

192.168.2.2	192.1.1.1	255.255.255.0	eth2
192.168.2.2	192.168.2.2	255.255.255.255	eth1
172.64.3.10	172.64.3.10	255.255.255.255	eth2
10.0.1.100	10.0.1.100	255.255.255.255	eth3

QUESTIONS?