PE-ARP: Port Enhanced ARP for IPv4 Address Sharing

Manish Karir,
Eric Wustrow, Jim Rees
Merit Network Inc.

Work Funded by DARPA under contract N66001-09-C-2115

Outline

- Motivation
- Background Observations
- PE-ARP
- Implementation Status
- Advantages of PE-ARP
- Related Work
- Conclusions and Future Work

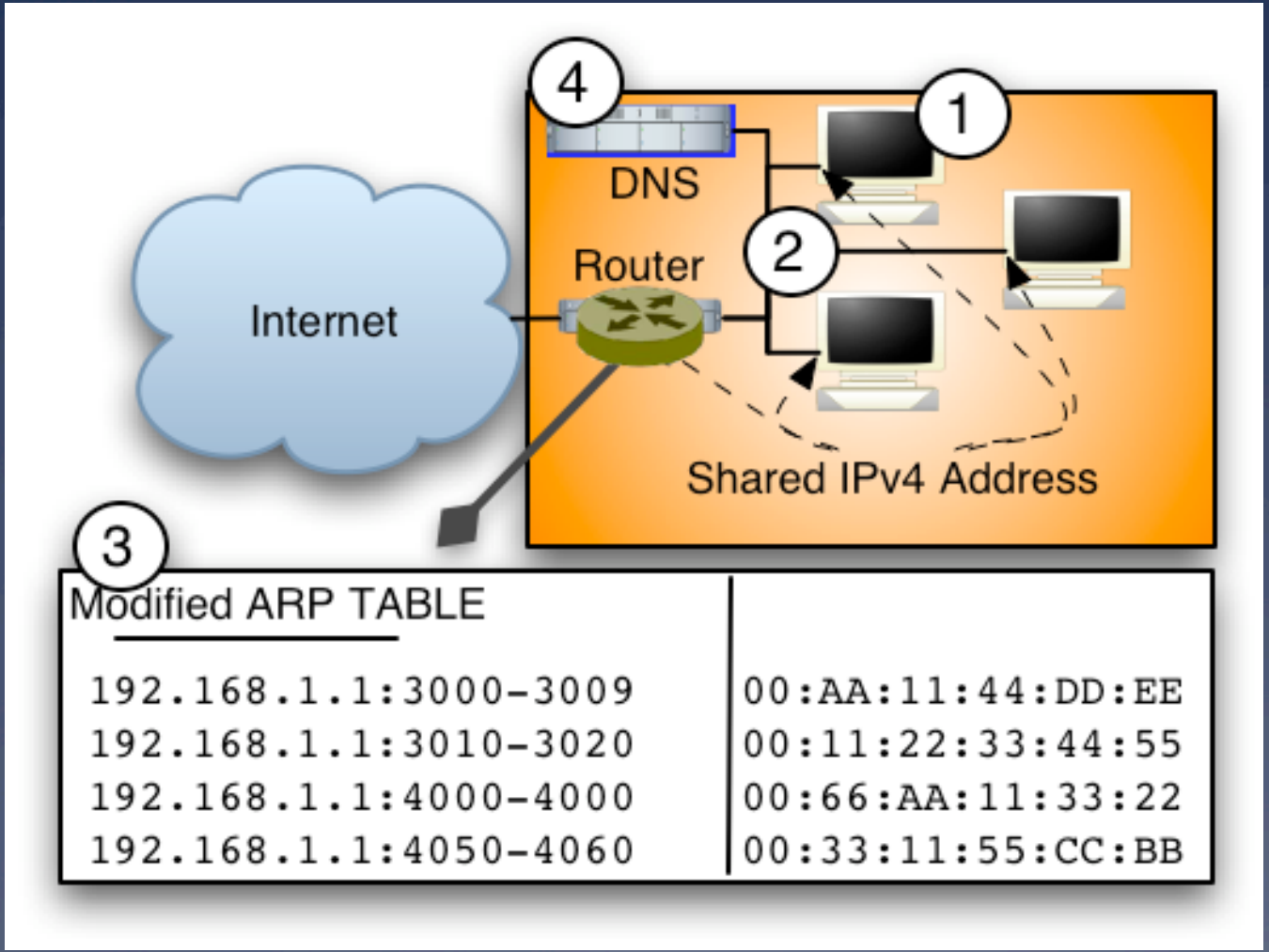
Motivation

- Looming threat of IPv4 Address Exhaustion
- IPv6 adoption accelerating how long is long enough?
- IPv4---CIDR ---NAT ----------x-------x---------- IPv6
  - ^ <^> (IPv4 exhaustion)
  - | (We are here)
- Evolving from era of plenty to scarcity
- What can we squeeze more utility out of?
- Good time to question long standing assumptions we never thought twice about before
- Emergence of "Port Scavenging"

Background

- Key Observations:
  - The range of valid source ports goes largely unused on end hosts
  - Why does a single end host need two unique network identifiers a hardware address and an IP address
  - An end host itself does not require an (IP address, port) it is the applications/services that run on it. A single end host could have different applications which use different IP/port combinations why do we have an single IP for the entire host? the (address, port) combination needs to be unique for a give service or application

PE-ARP Architecture



PE-ARP Components

① Port Range Management on End Hosts

Agent to monitor source port usage on the end hosts, trivial on Linux via /proc file system. DHCP modifications to handle IP, port range

② Modified ARP Protocol

Need to include port numbers in ARP request/response mechanism

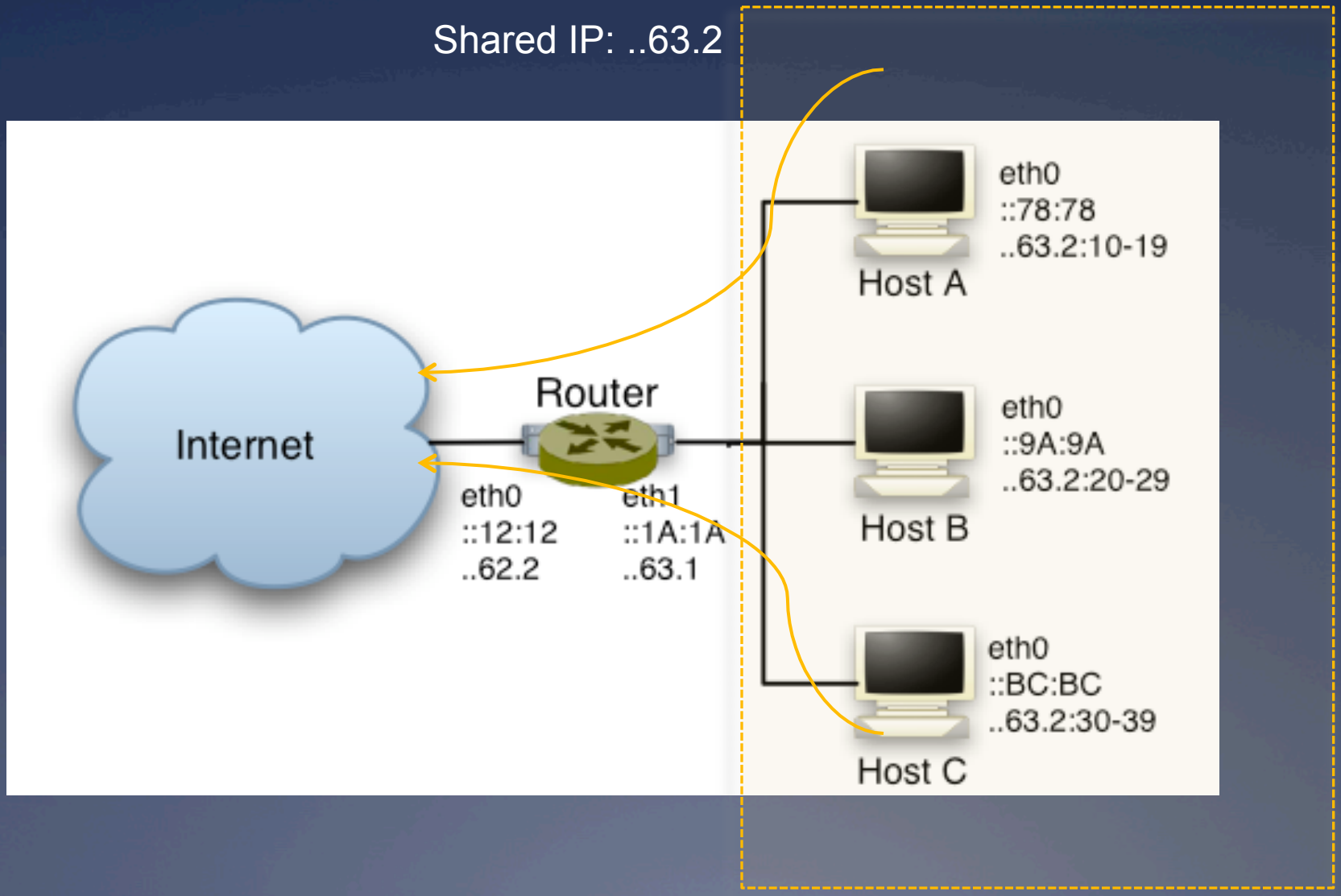
③ Modified ARP Table

ARP table is modified to include port number information when mapping to a HW address

④ DNS Support for Service Location

While 1-3 are sufficient for a client only environment, since services such as httpd can now run on non-port 80 ports, we need DNS to understand that queries are requesting not just name resolution but IP, port information, SRV Records

PE-ARP – Test Network



PE-ARP Packet Processing

- Outbound Packets
  - End hosts pick source port from allowed range
  - Send packets out as usual
- Inbound Packets
  - The edge router when attempting to lookup the hardware MAC address for a given packet also looks at the destination port number
  - If an entry is not present in its ARP table it sends a ARP request for {IP, port}. Only the end host with the {IP, port} responds with its MAC address
  - Packet is transmitted to the correct end host

PE-ARP Implementation Details

- Working prototype on Linux 2.6.29 kernel
- Roughly 1300 lines in patch including comments
- Most of the changes are to the ARP functions with some minor changes to the routing/forwarding code
- Beware there is caching everywhere!
- No DHCP/DNS modifications yet so manual range allocation
- Current prototype has ability to show correct operation of all networking functions and can communicate seamlessly with the Internet, web-browsing, ssh, email etc.
- Also able to demonstrate the ability to run servers on the test network for inbound traffic. As DNS work is not complete yet we have to manually specify ports to connect to.

arp -n

```
pe-arp-hostA:~$ arp -n
Address                  HWtype  HWaddress           Flags Mask            Iface
192.35.162.2:2000-2999   ether   00:0c:29:9e:8c:ee   C                     eth1
192.35.162.2:3000-3999   ether   00:0c:29:39:44:ab   C                     eth1
198.108.63.1:0-0         ether   00:12:7f:c4:38:d3   C                     eth0
```

PE-ARP Deployment Scenarios

- Our current implementation supports two deployment scenarios:
  - Router on the Network Edge – Modified ARP on Linux router forwarding packets between interfaces
  - Bridge Solution – Where the end router cannot be modified (legacy device in place) a port-aware bridge can serve the same purpose to allow easier migration of smaller subnet to PE-ARP

Advantages of PE-ARP

- It does not require a massive global hardware or software upgrade
- Incremental deployment - A single site can choose to use this technique and obtain its full benefits while at the same time continuing to be completely interoperable with the rest of the Internet
- E2E Consistency - It does not require Layer 3 or higher packet modifications in the network that would violate the e2e principle
- Breaks the one-to-one mapping notion between an IP address and a hardware address – this enables more flexible networks
- A single IP address can be shared by thousands of hosts leading to possibly much more efficient usage of scarce IPv4 addresses, a /24 can be accommodated in a /28, a /20 can be accommodated in a /24
- Enables better per end user flow management/accounting as end hosts are not hidden behind a shared NAT IP address space

Related Work

- CIDR
- NAT
- CGN – NAT++
- The Port Scavenging Revolution - July 2009:
  - A+P - http://tools.ietf.org/html/draft-ymbk-aplusp-04 :
    - Same idea of scavenging source port range, different approach - does not use the ARP mechanism, might still use NAT or tunneling
  - E2ENAT: http://www.ietf.org/id/draft-ohta-e2e-nat-00.txt
    - IP addresses maybe modified by NAT but port numbers are not, this ensures that specific ports map to specific end hosts
  - Port Range Routers: http://tools.ietf.org/html/draft-boucadair-port-range-02:
    - Advocates the use of source port ranges coupled with port range routers to perform the mapping from a give port range to the final destination ip address

Conclusions and Future Work

- Does PE-ARP break anything? Does PE-ARP replace IPv6?
  - Yes, some things that relied on old assumptions break
  - Not likely, but it can buy us more transition time
- Why is it good?
  - It questions some fairly fundamental assumptions of the Internet architecture to see where things can be stretched – Perhaps it is okay to rethink some deeply ingrained assumptions
  - Packets are not modified in the network once they leave an end host
- Working code available at:
  - http://software.merit.edu/pe-arp
- Next Steps: How do we handle protocols which don't have port numbers? Why are DNS SRV Records for port information not used as they should be, DHCP modifications for ports
- I-D in preparation