A Protocol for Anycast Address Resolving

<draft-ata-ipv6-anycast-resolving-00.txt>

Shingo Ata, Osaka City University
ata@info.eng.osaka-cu.ac.jp
Hiroshi Kitamura, NEC Corporation
kitamura@da.jp.nec.com
Masayuki Murata, Osaka University
murata@cmc.osaka-u.ac.jp

Problems in Anycast

- Cannot use anycast for stateful (e.g., TCP connection) sessions
  - Destination node may change during the session
  - Anycast address cannot use as the source address
  - Anycast addresses are not syntactically distriguishable from unicast addresses

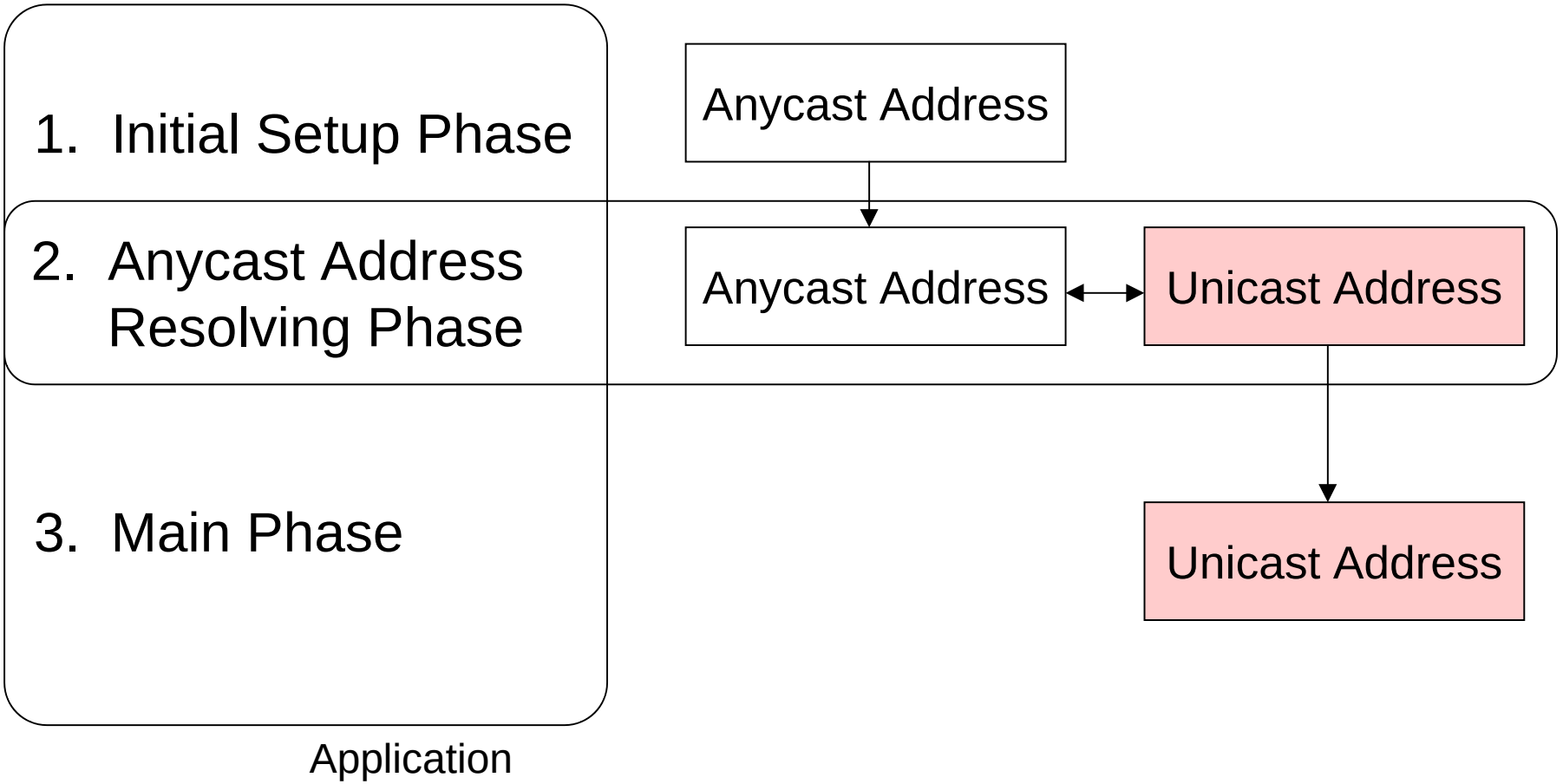
Goals

- To utilize anycast in stateful communications
  - w/o (or w/ minimum) application modification
  - w/o (or w/ minimum) protocol extension
  - in applications not designed for anycast
- For example,
  - TCP applications (http, ftp, telnet, etc...)
  - UDP application (DNS query)

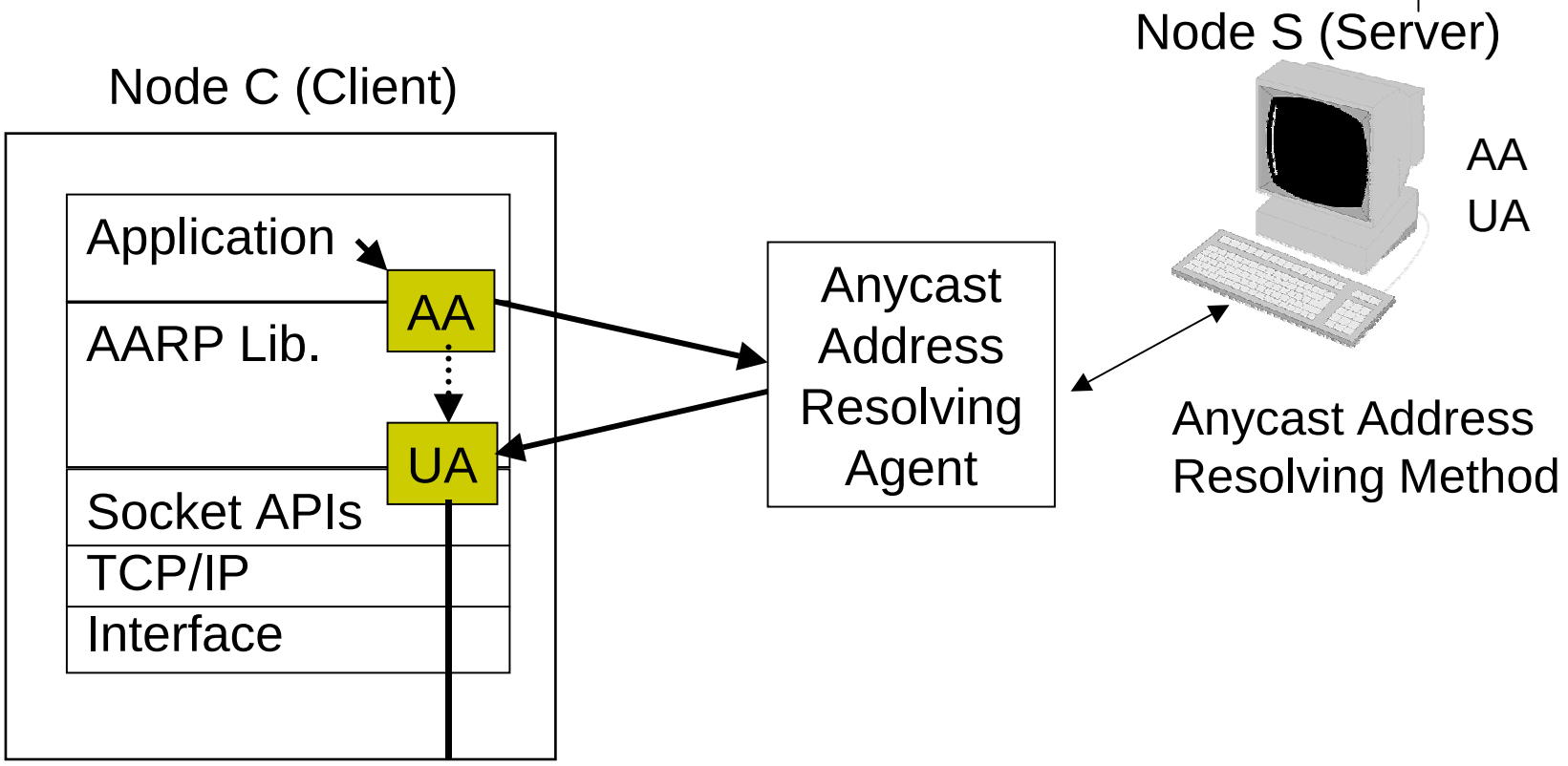
Anycast Address Resolving

- Resolve an anycast address into the corresponding unicast address
- Anycast address is used
  - only to determine the appropriate node out of anycast membership nodes
- After starting communication,
  - all packets are sent by the unicast address
- Anycast Address Resolving Protocol (AARP)

Add a new procedure for AARP



Overview of AARP



AA : Anycast Address

UA : Unicast Address

Address Resolving Method

- AARP adopts **packet probing** technique
  - w/o modification at the server
  - C first sends a probe packet included AA in its destination.
  - The probe packet is routed and sent to S.
  - S then returns a packet to C. The source address of the returned packet is set to UA.
  - C waits the return packet and gets UA from the source address of the received packet.

Implementation Issues

- For address resolving
  - ICMPv6 Echo Request/Reply
- For AARP Library
  - Same approach in SOCKS5 (RFC1928)
- Current implementation
  - has been verified in TCP (telnet, ftp, http) applications, and UDP (DNS query) application

Applicability Statements

- Software deployment on the client node
- Protocol overhead
  - Needs packets for anycast address resolving
  - Anycast and unicast addresses are not distinguishable
    - The node should check all addresses (anycast, unicast)
- Caching resolved addresses
  - Reduce the traffic for probing packets
  - Policy of caching depends on type of applications

Security Considerations

- "Remote Redirect" problem
  - Receiving spoofed packet to another node
  - The anycast client may become an unexpected attacker of denial of service
- "Blackhole" problem
- Same issues in DNS resolving, Neighbor Discovery, etc, should be considered

Next Steps ?

- Revise the document
  - improve the security consideration section
  - append caching issues
- Comments ? or questions ?