NAT Traversal in SIP

Gonzalo.Camarillo@ericsson.com

ERICSSON

TAKING YOU FORWARD

Outline

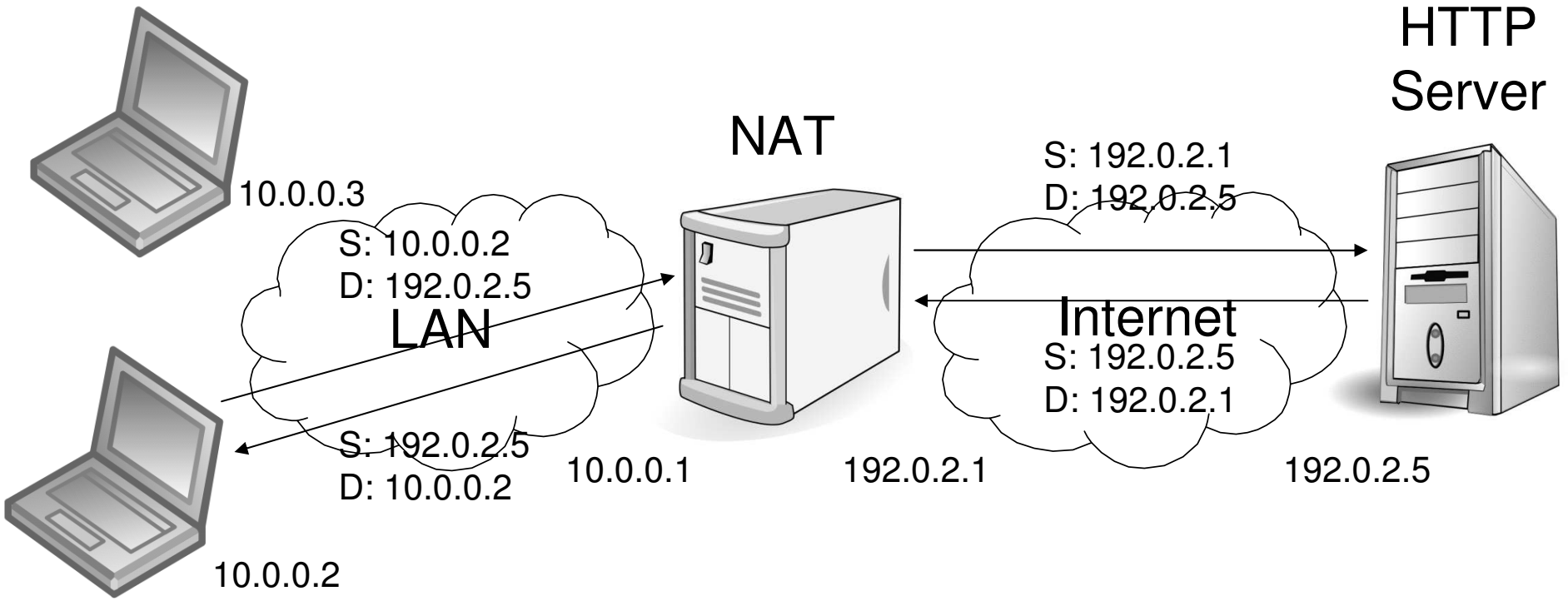
- Introduction to NATs
- NAT Behavior (actual and recommended)
  - UDP
  - TCP
- IAB UNSAF Considerations
- STUN
- STUN Relay Usage
- NAT Traversal in SIP
  - SIP Response Routing
  - User Agent Reachability
  - ICE

Outline

- Introduction to NATs
- NAT Behavior (actual and recommended)
  - UDP
  - TCP
- IAB UNSAF Considerations
- STUN
- STUN Relay Usage
- NAT Traversal in SIP
  - SIP Response Routing
  - User Agent Reachability
  - ICE

Origin of NATs

- Created to resolve the IPv4 address exhaustion problem
- Designed with the web in mind
  - Client/server paradigm



Side-effects of NATs

- Hosts behind NATs are not reachable from the public Internet
  - Sometimes used to implement security
  - Breaks peer-to-peer (as opposed to client/server) applications
- NATs attempt to be transparent
  - Troubleshooting becomes more difficult
- NATs are a single point of failure
- NATs' behavior is not deterministic
  - Difficult to build applications that work through NATs

IETF BEHAVE WG

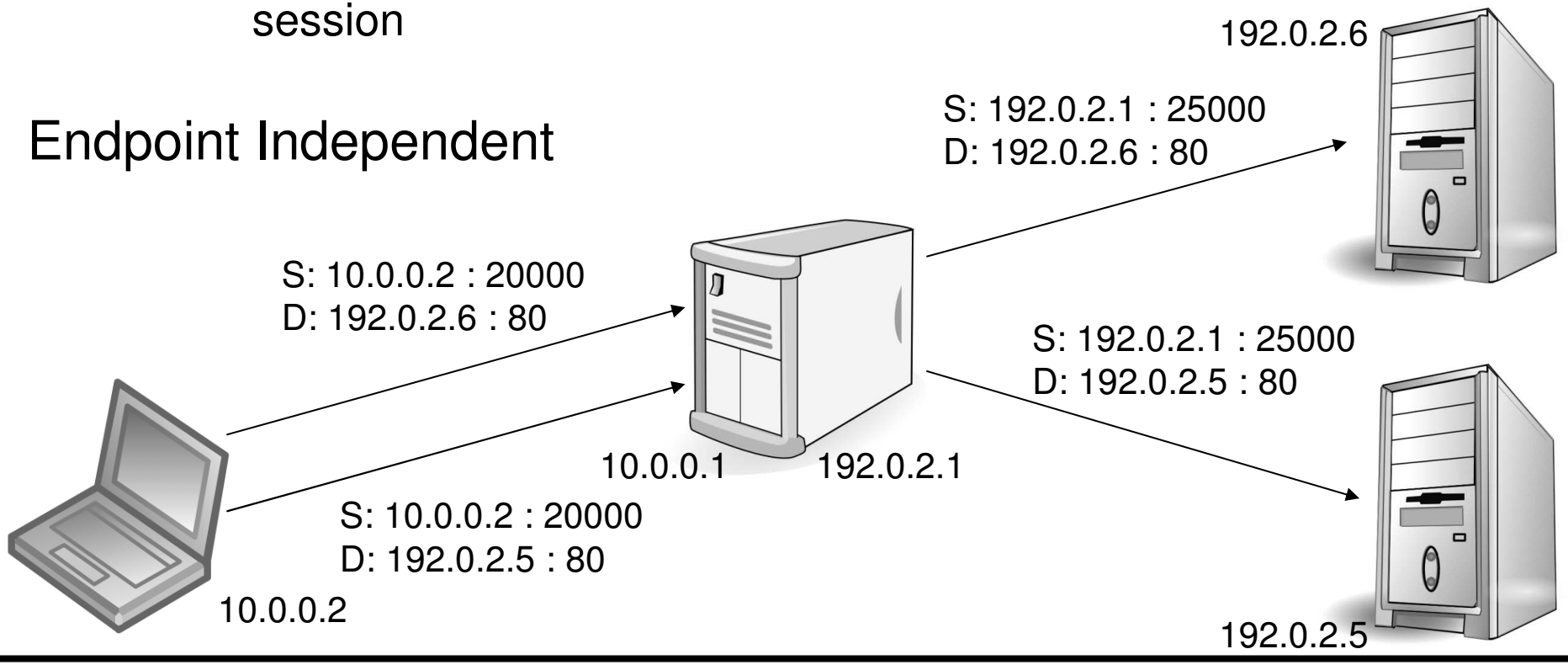
- Classification of current NAT behaviors
  - Existing terminology was confusing
    - Full cone, restricted cone, port restricted cone, and symmetric
  - New terminology needed
- Recommendations for NAT vendors
  - BEHAVE-compliant NATs are deterministic
  - Easier to build applications

Outline

- Introduction to NATs
- NAT Behavior (actual and recommended)
  - UDP
  - TCP
- IAB UNSAF Considerations
- STUN
- STUN Relay Usage
- NAT Traversal in SIP
  - SIP Response Routing
  - User Agent Reachability
  - ICE

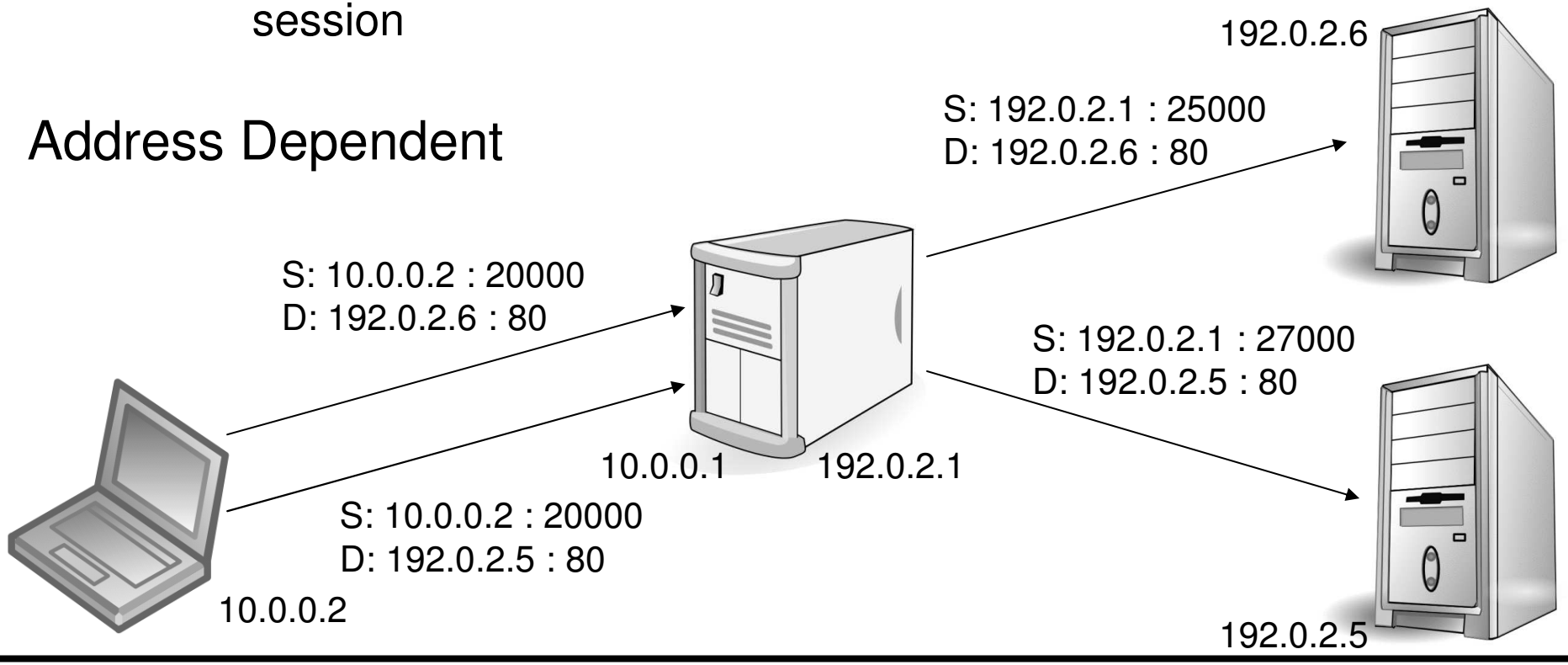
Mapping Behavior

- For session originated on the same address and port
  - Endpoint independent: same mapping to different sessions
    - MUST use it
  - Address dependent: same mapping to sessions to the same host
  - Address and port dependent: a mapping only applies to one session



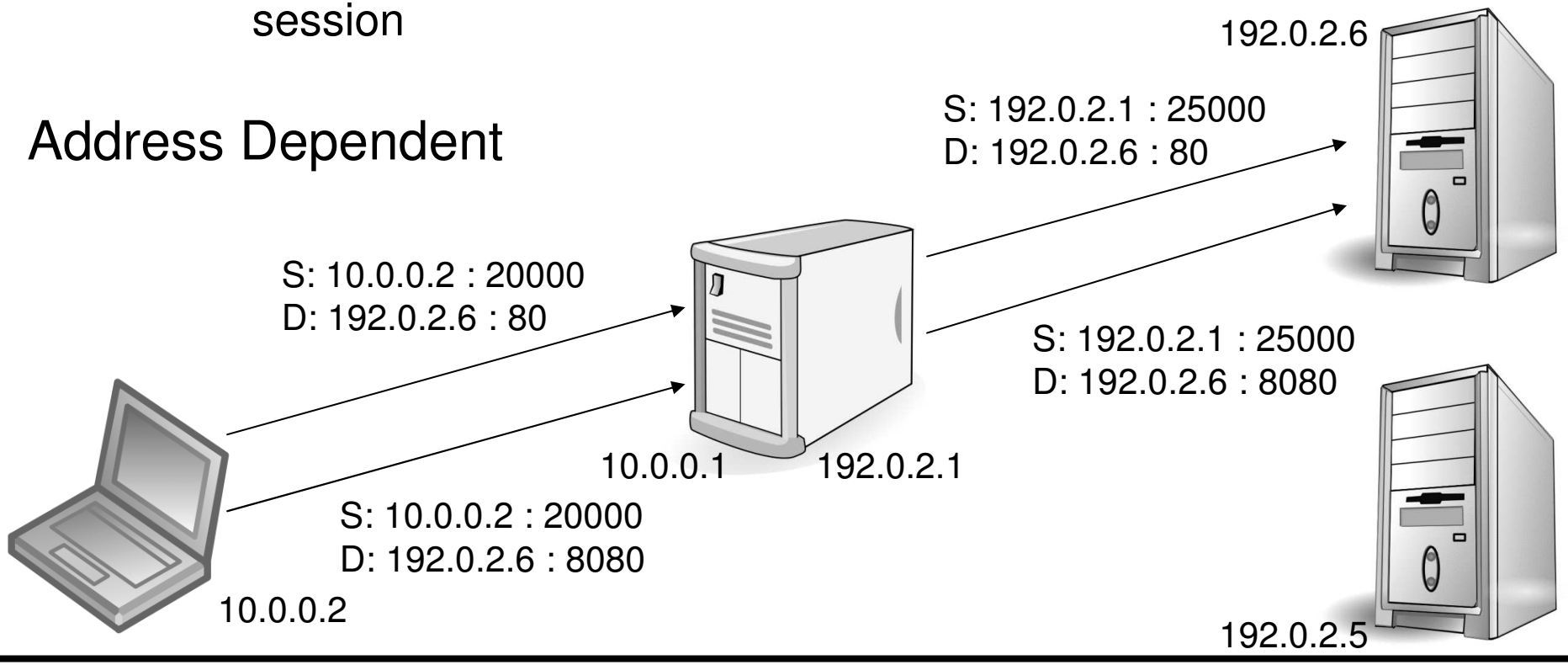
Mapping Behavior

- For session originated on the same address and port
  - Endpoint independent: same mapping to different sessions
    - MUST use it
  - Address dependent: same mapping to sessions to the same host
  - Address and port dependent: a mapping only applies to one session



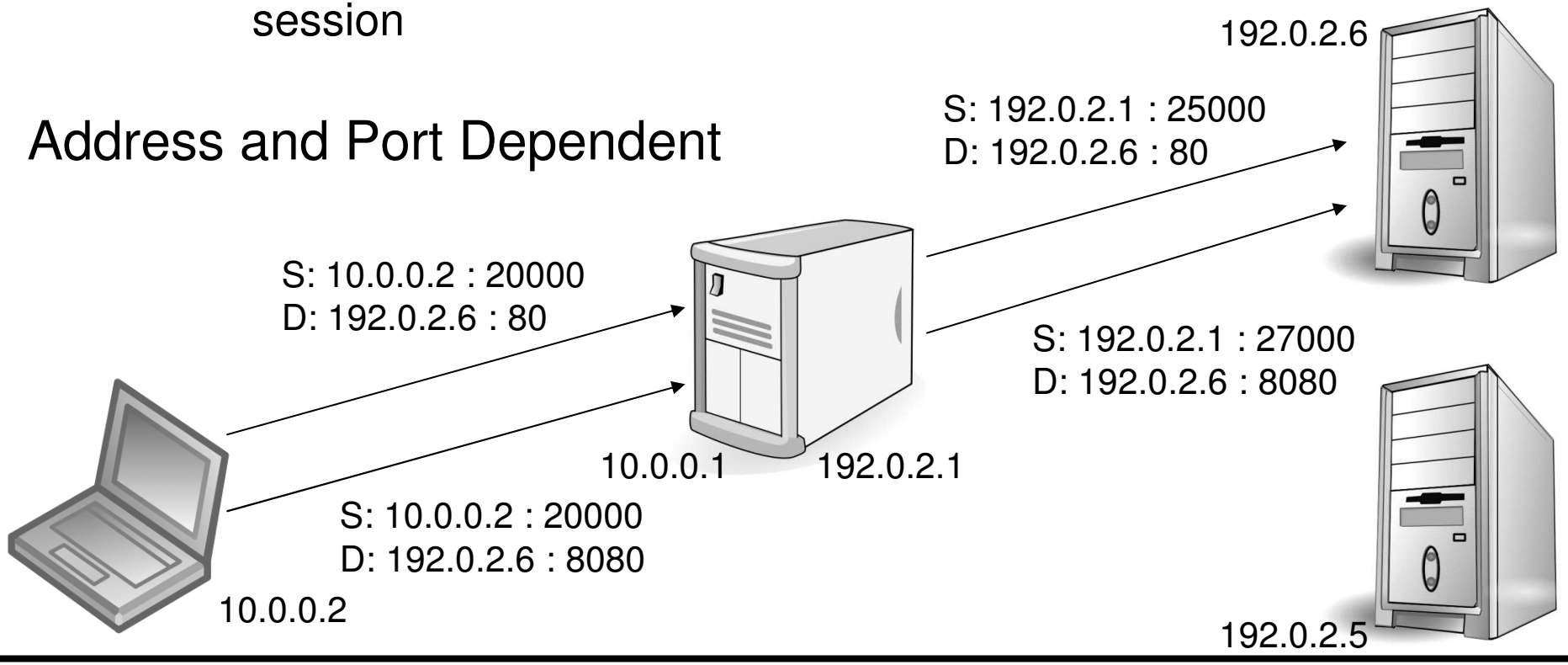
Mapping Behavior

- For session originated on the same address and port
  - Endpoint independent: same mapping to different sessions
    - MUST use it
  - Address dependent: same mapping to sessions to the same host
  - Address and port dependent: a mapping only applies to one session



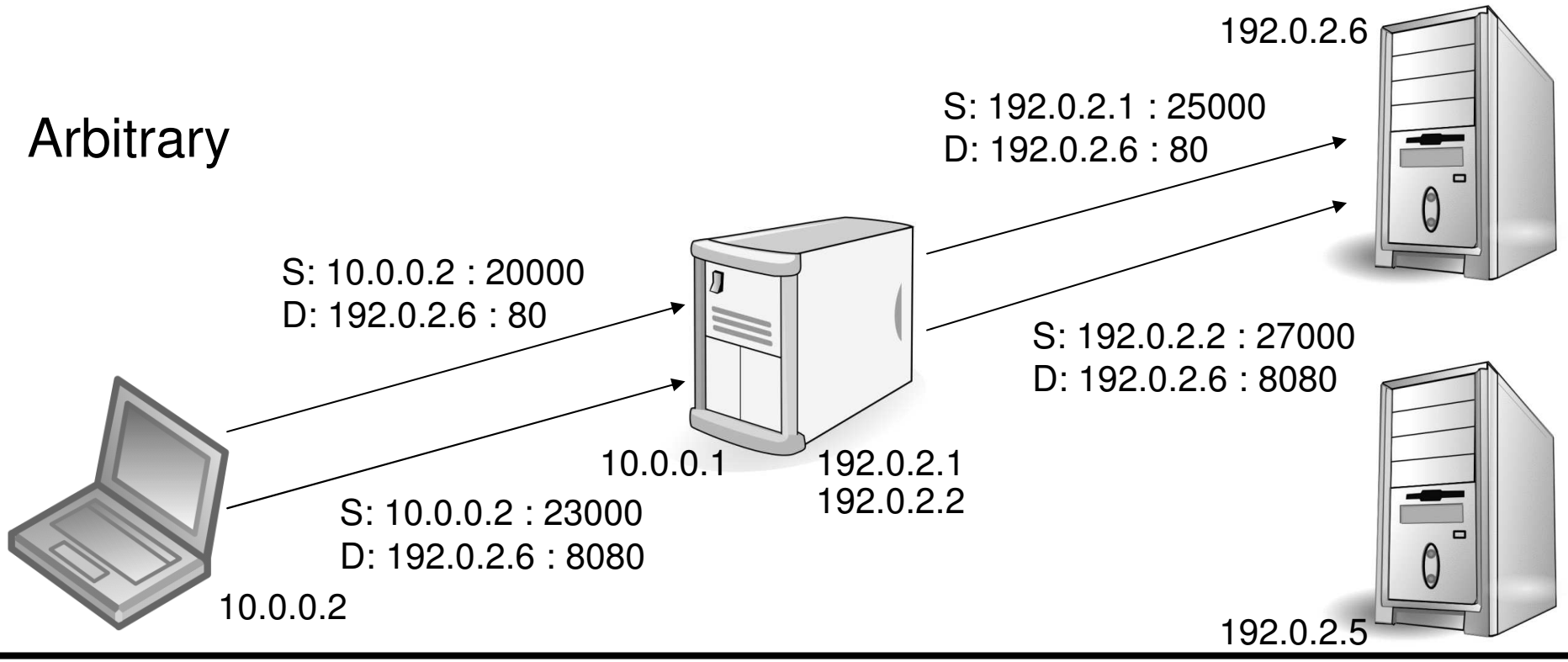
Mapping Behavior

- For session originated on the same address and port
  - Endpoint independent: same mapping to different sessions
    - MUST use it
  - Address dependent: same mapping to sessions to the same host
  - Address and port dependent: a mapping only applies to one session



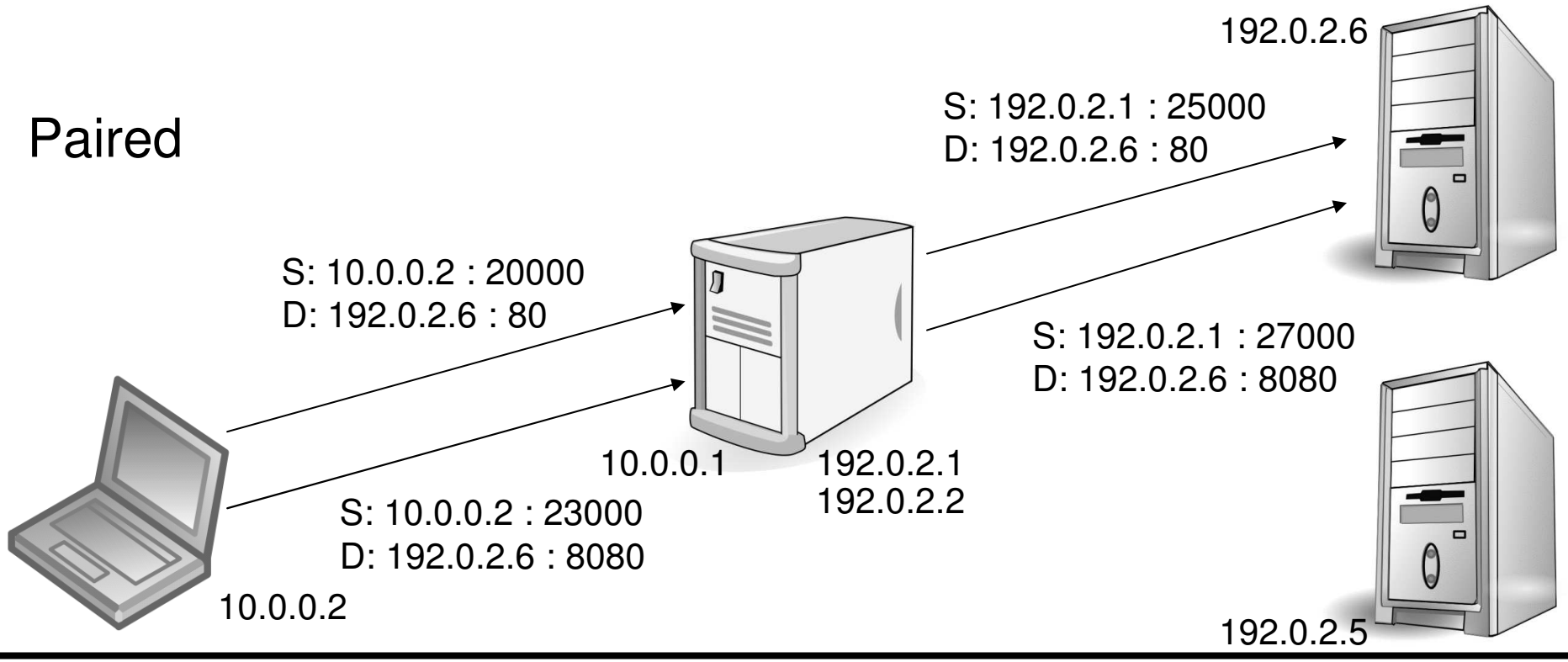
IP Address Pooling Behavior

- NATs with a pool of external IP addresses
  - Arbitrary: an endpoint may have simultaneous mappings corresponding to different external IP addresses of the NAT
  - Paired: same external IP address of the NAT
    - RECOMMENDED



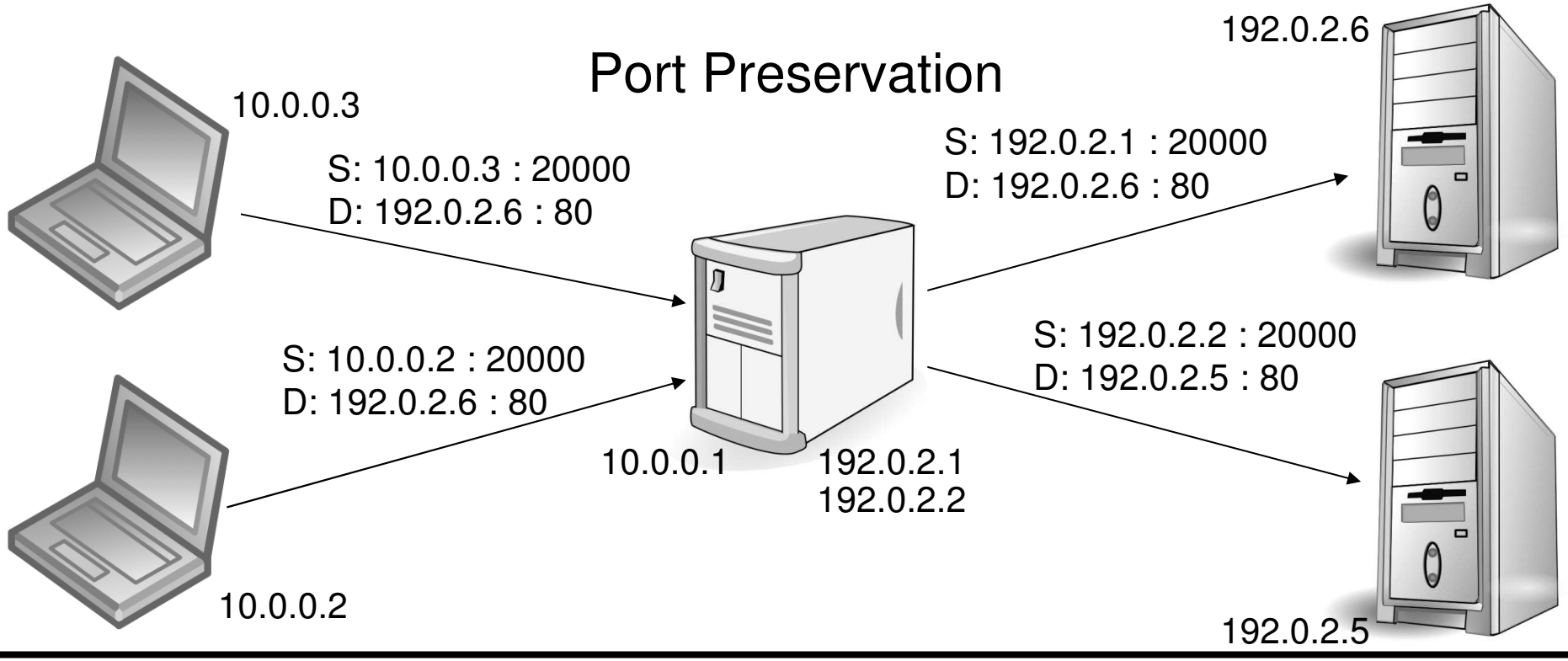
IP Address Pooling Behavior

- NATs with a pool of external IP addresses
  - Arbitrary: an endpoint may have simultaneous mappings corresponding to different external IP addresses of the NAT
  - Paired: same external IP address of the NAT
    - RECOMMENDED



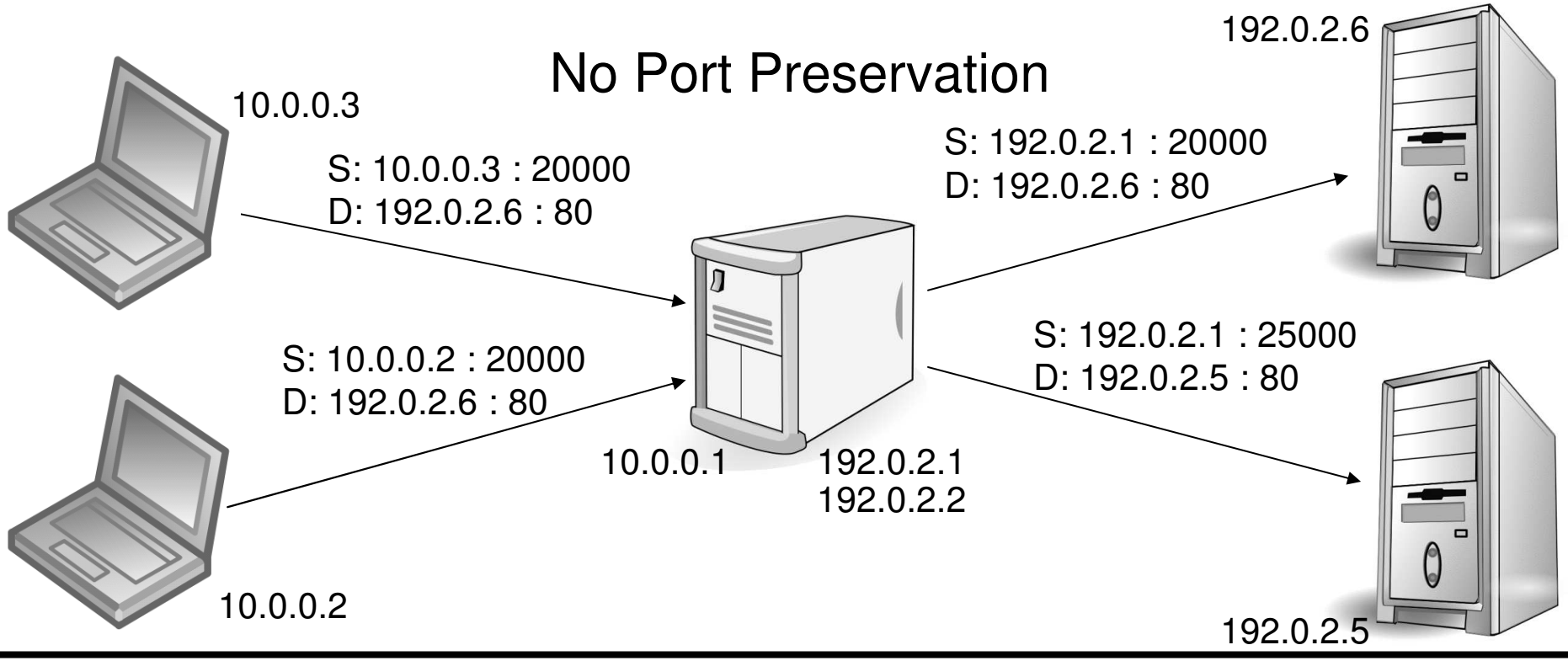
Port Assignment

- Port preservation: preserves the port as long as there are available IP addresses in the NAT's pool
- No port preservation
- Port overloading: the port is preserved always, even without available IP addresses in the NAT's pool
  - The NAT relays on the source of the response



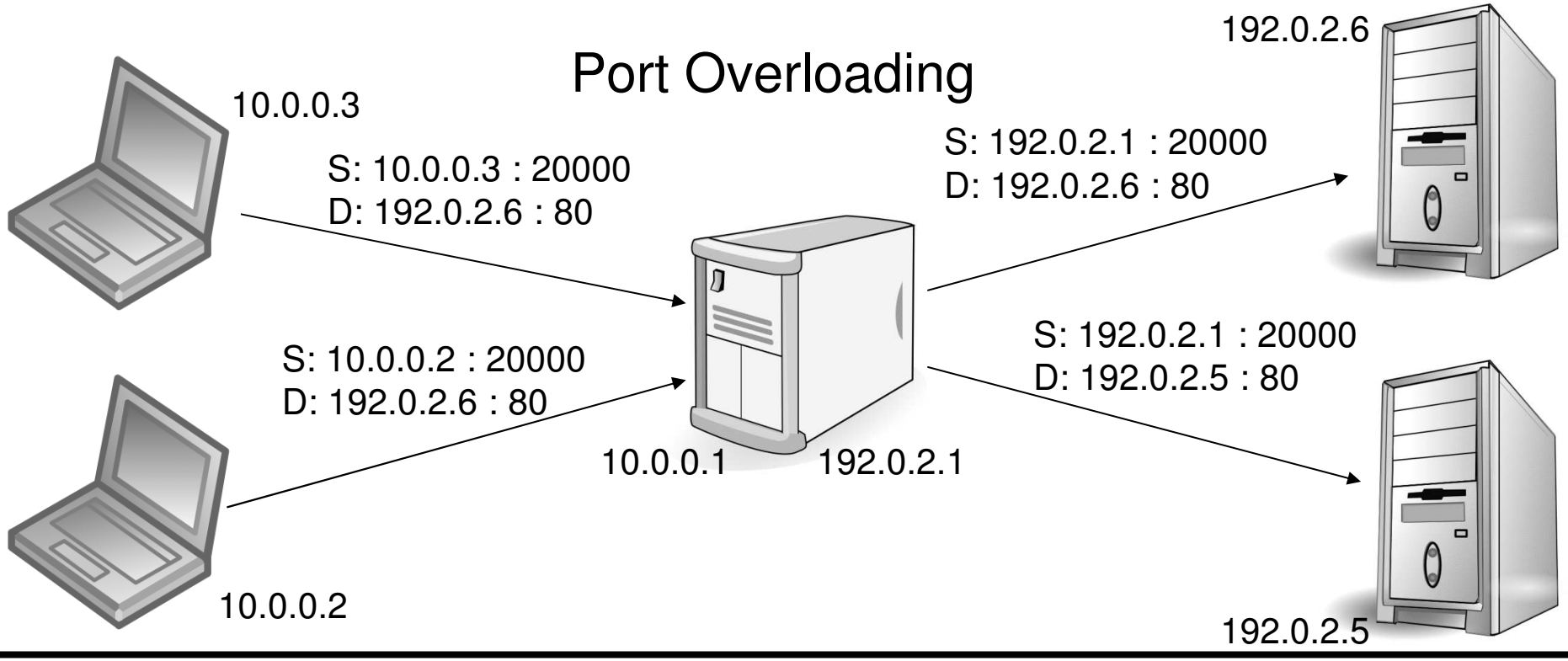
Port Assignment

- Port preservation: preserves the port as long as there are available IP addresses in the NAT's pool
- No port preservation
- Port overloading: the port is preserved always, even without available IP addresses in the NAT's pool
  - The NAT relays on the source of the response



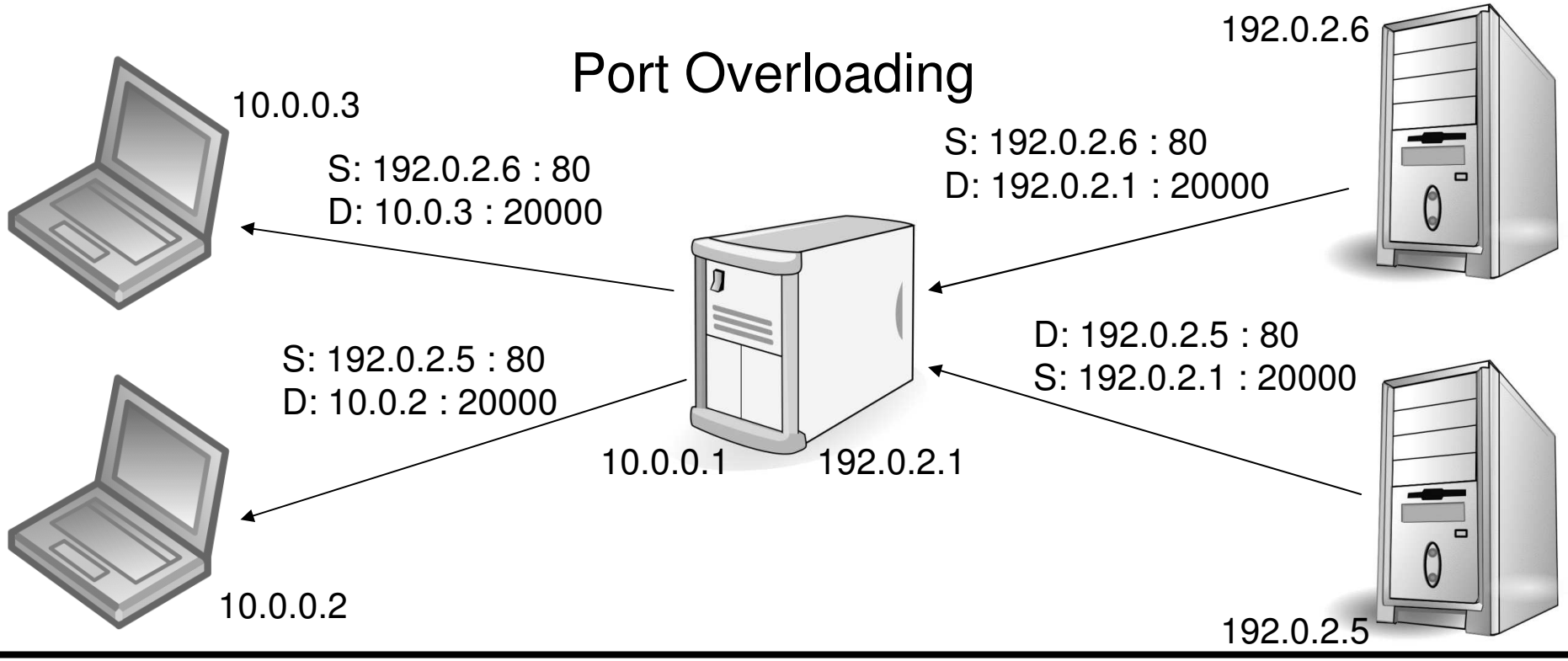
Port Assignment

- Port preservation: preserves the port as long as there are available IP addresses in the NAT's pool
- No port preservation
- Port overloading: the port is preserved always, even without available IP addresses in the NAT's pool
  - The NAT relays on the source of the response



Port Assignment

- Port preservation: preserves the port as long as there are available IP addresses in the NAT's pool
- No port preservation
- Port overloading: the port is preserved always, even without available IP addresses in the NAT's pool
  - The NAT relays on the source of the response



Port Ranges

- 1- 1023 Well known
- 1024 – 49151 Registered
- 49152 – 65535 Dynamic / Private

- RECOMMENDED to preserve the following ranges
  - 1 – 1023
  - 1024 – 65535

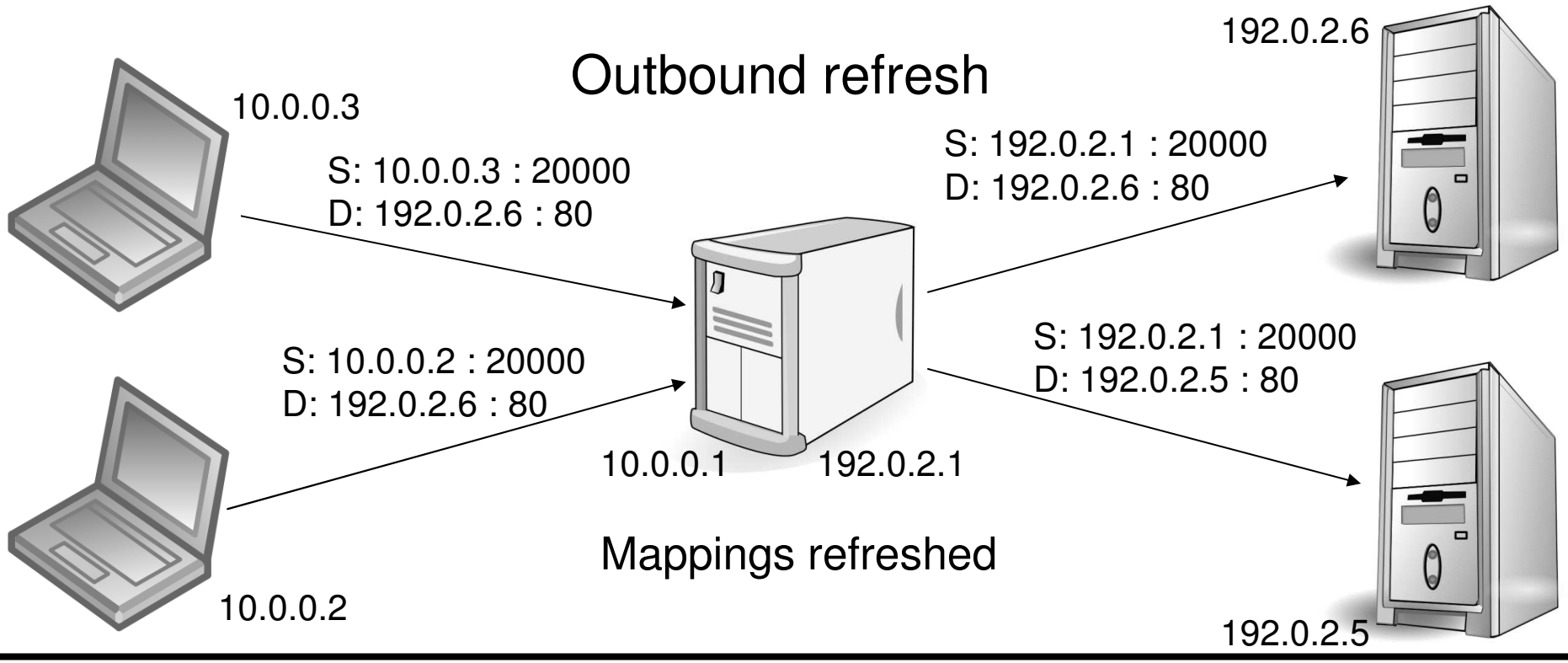
- Port overloading MUST NOT be used
  - Problems when two internal hosts connect to the same external host
- It is RECOMMENDED that NATs preserve port parity (even/odd)
- No requirement for port contiguity

Mapping Timeout

- A NAT UDP mapping MUST NOT expire in less than 2 minutes
- NATs can have application-specific timers
  - Well-known ports
- It is RECOMMENDED to use more than 5 minutes

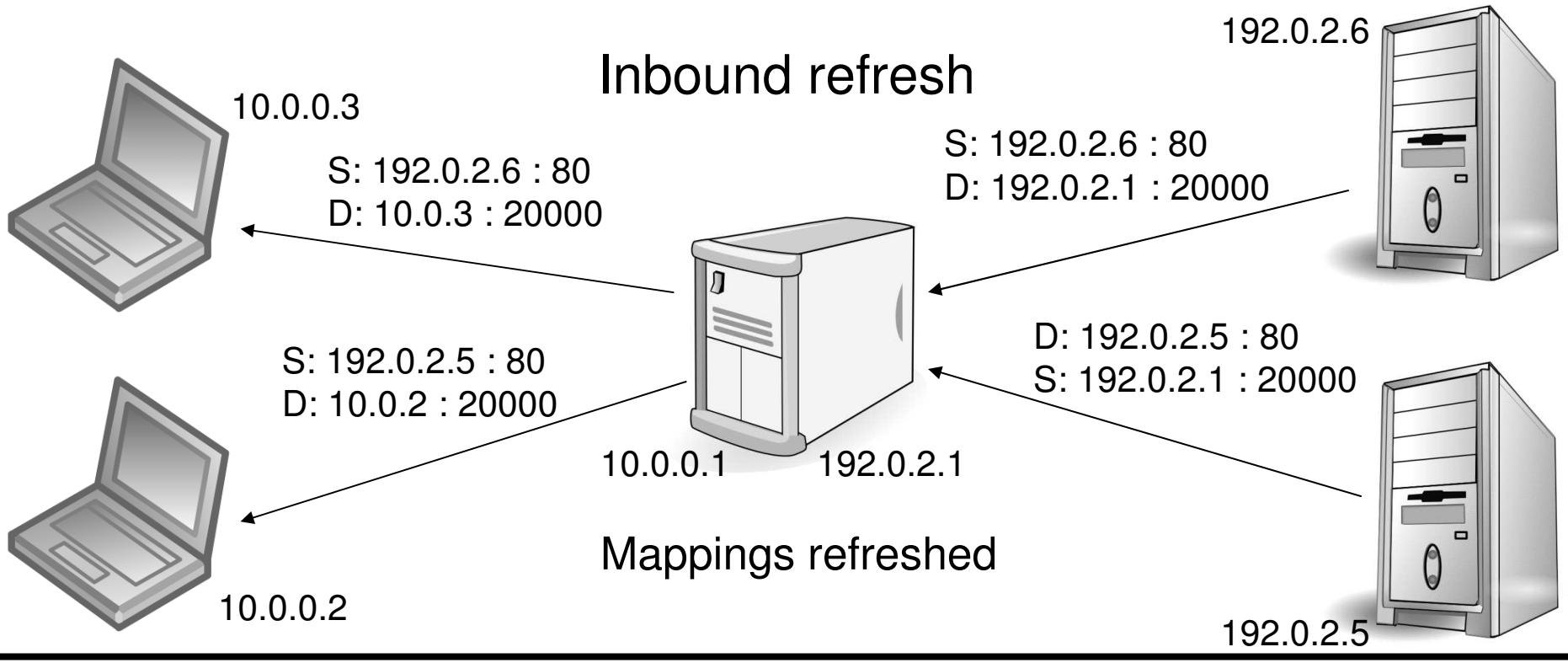
Mapping Refresh

- NAT outbound refresh: packets from the internal to the external interface
  - MUST be used
- NAT inbound refresh: packets from the external to the internal interface
  - Attackers may keep the mapping from expiring
  - MAY be used



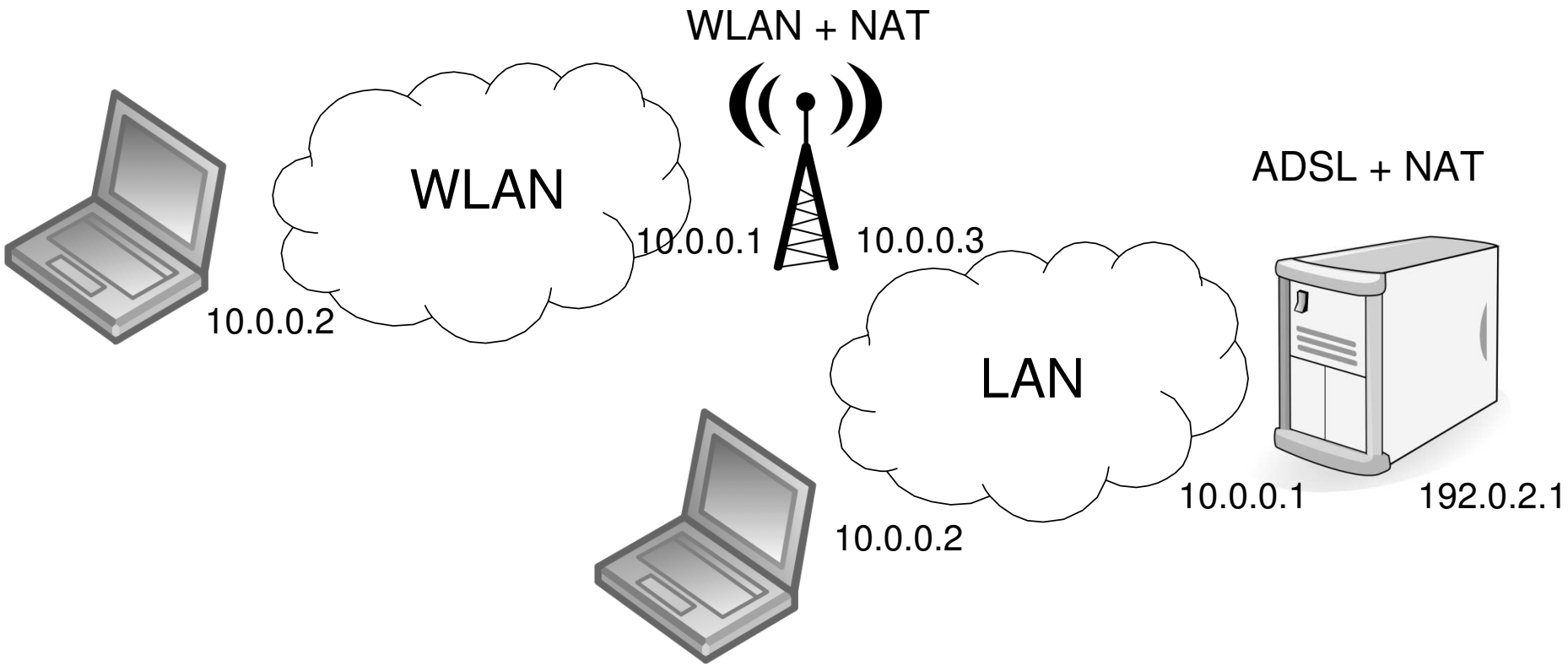
Mapping Refresh

- NAT outbound refresh: packets from the internal to the external interface
  - MUST be used
- NAT inbound refresh: packets from the external to the internal interface
  - Attackers may keep the mapping from expiring
  - MAY be used



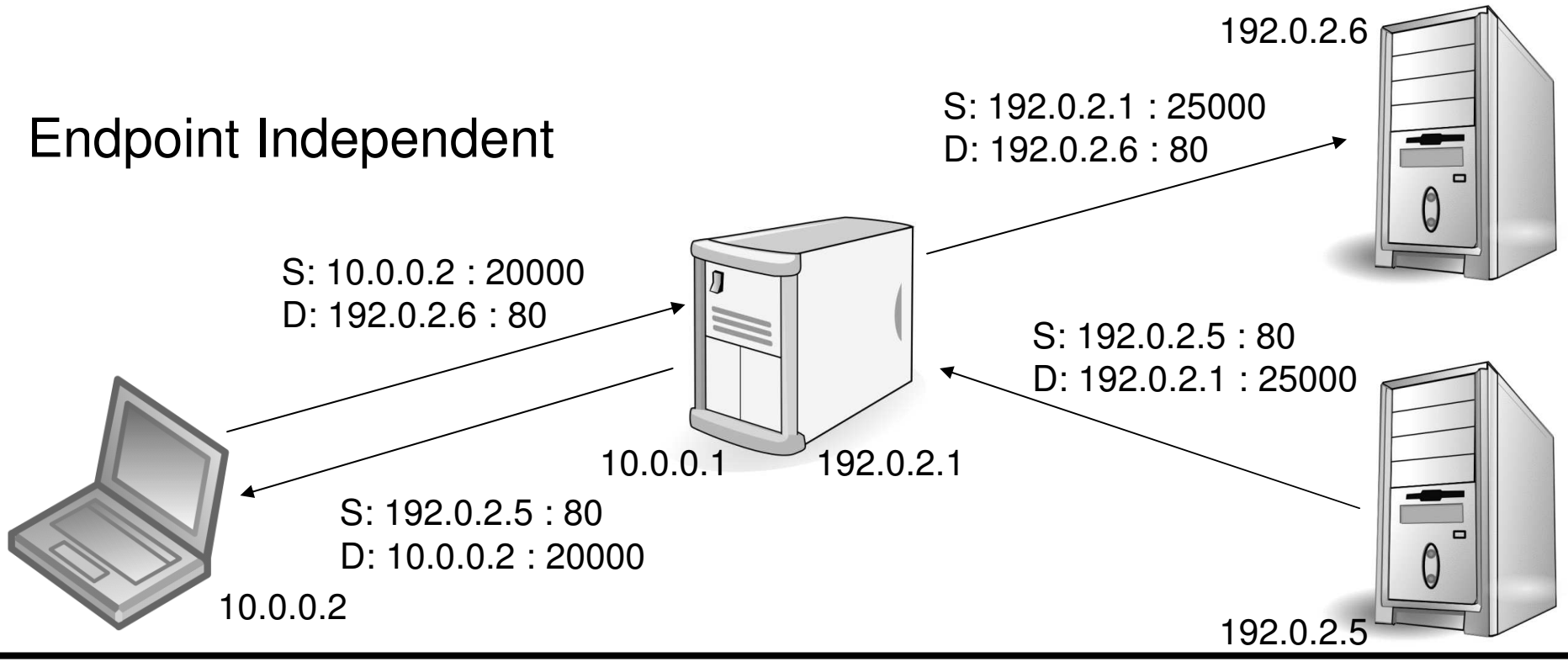
External Address Spaces

- NATs MUST be able to handle external address spaces that overlap with the internal address space
  - Internal nodes cannot communicate directly with external nodes that have the same address as another internal node
  - However, they can use STUN techniques



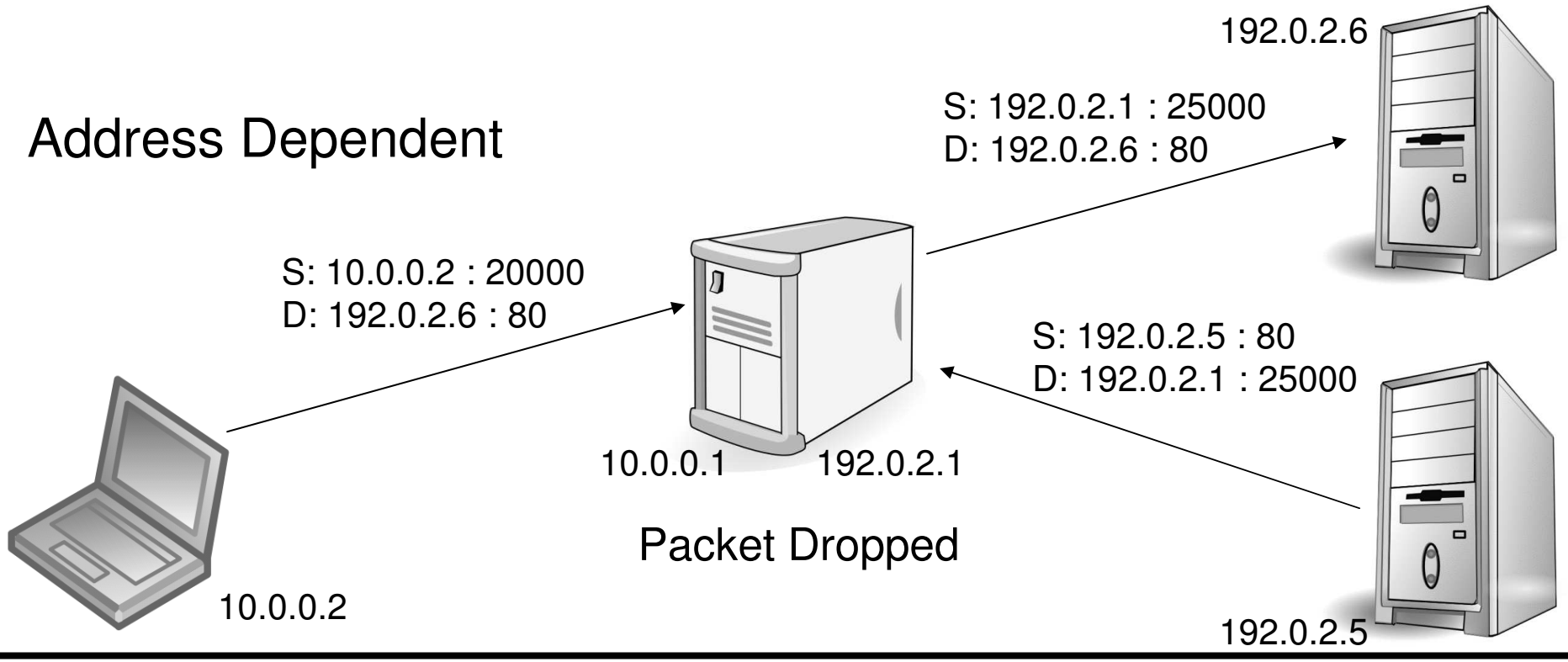
Filtering Behavior

- Endpoint independent: any packets allowed back
- Address dependent: external hosts can return packets
- Address and port dependent
  - Packets sent to an address + port
  - Incoming packets allowed only from that address + port



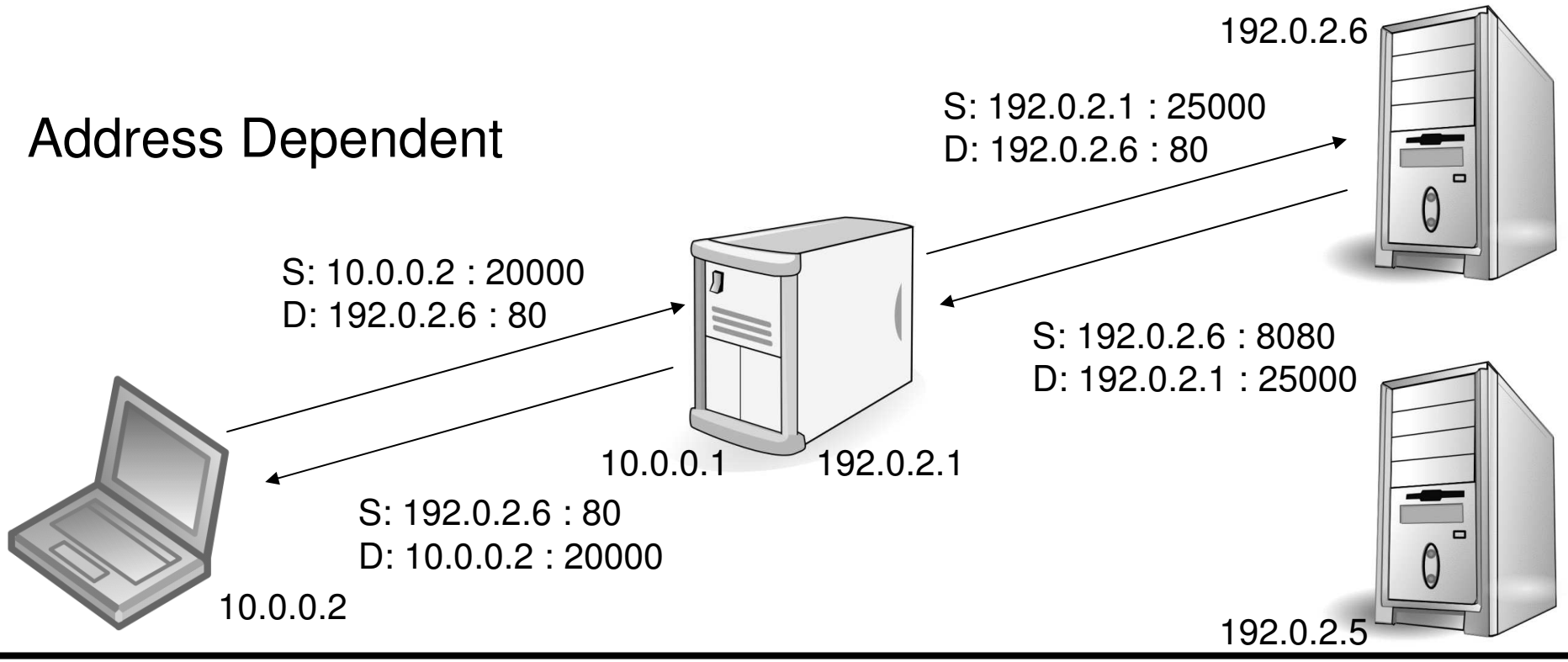
Filtering Behavior

- Endpoint independent: any packets allowed back
- Address dependent: external hosts can return packets
- Address and port dependent
  - Packets sent to an address + port
  - Incoming packets allowed only from that address + port



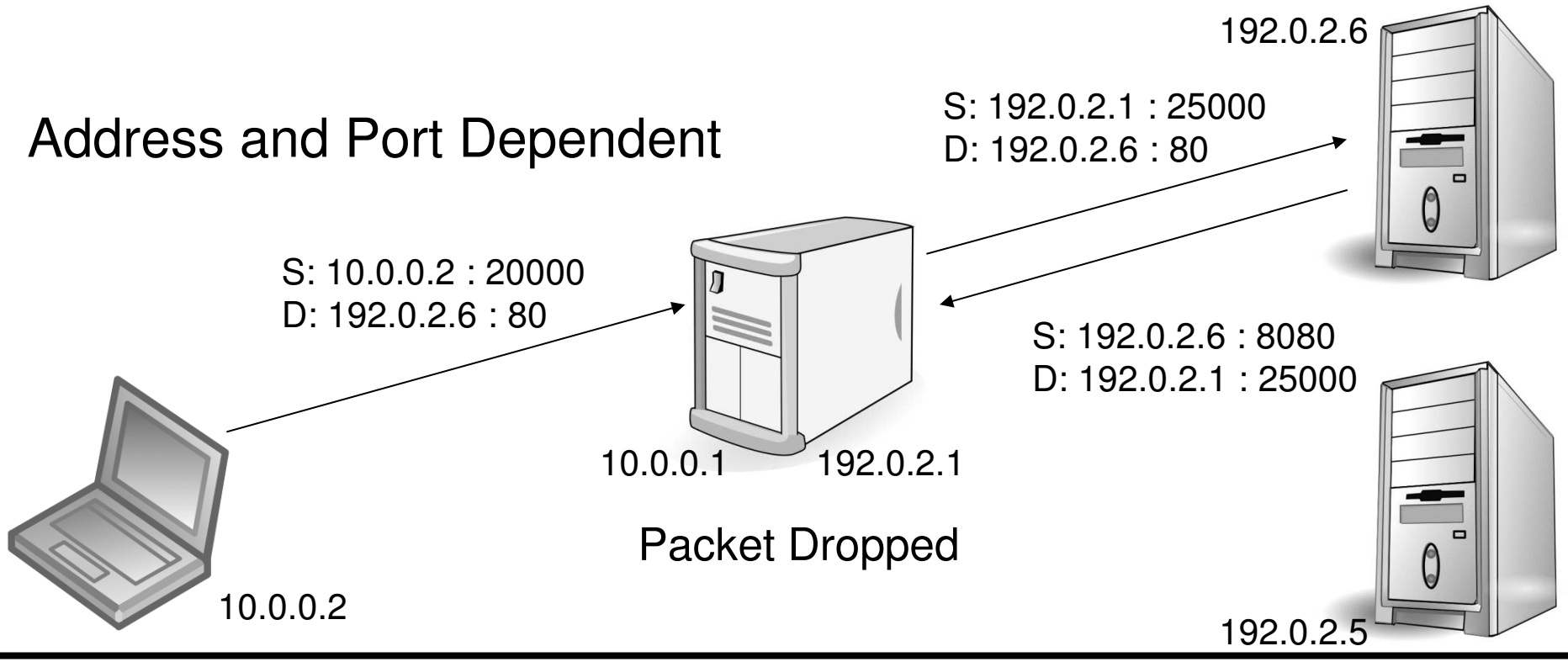
Filtering Behavior

- Endpoint independent: any packets allowed back
- Address dependent: external hosts can return packets
- Address and port dependent
  - Packets sent to an address + port
  - Incoming packets allowed only from that address + port



Filtering Behavior

- Endpoint independent: any packets allowed back
- Address dependent: external hosts can return packets
- Address and port dependent
  - Packets sent to an address + port
  - Incoming packets allowed only from that address + port

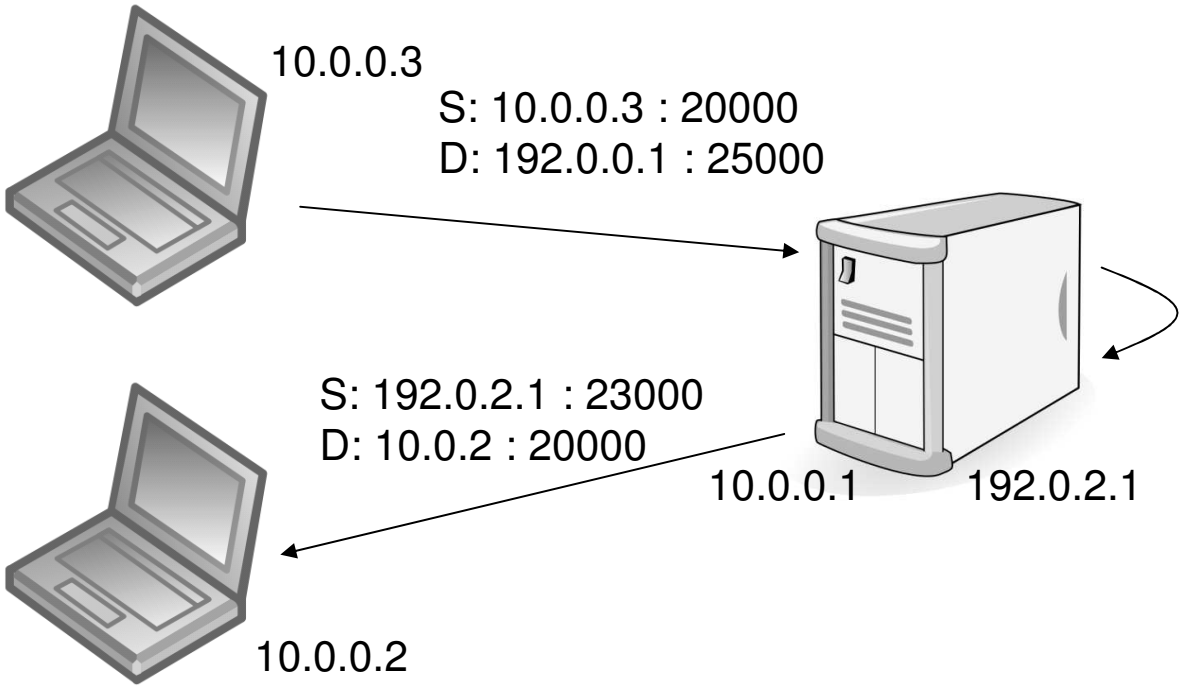


Filtering Behavior

- Endpoint independent filtering is RECOMMENDED
  - Opens up ports for attackers
- If a more stringent filtering is required
  - Address dependent filtering is RECOMMENDED

Hairpinning

- Internal hosts communicate using external addresses
  - MUST be supported



Fragmented Packets

- Receive fragments ordered
  - Only able to receive fragments in order
- Receive fragments out of order
  - MUST be supported
  - As long as DoS attacks do not compromise the NAT's ability to process in order fragments and unfragmented packets
- Receive fragments none

Various

- ALGs SHOULD be turned off
  - MAY interfere with UNSAF methods
- NATs MUST be deterministic
  - NATs MUST NOT change their mapping or filtering behavior

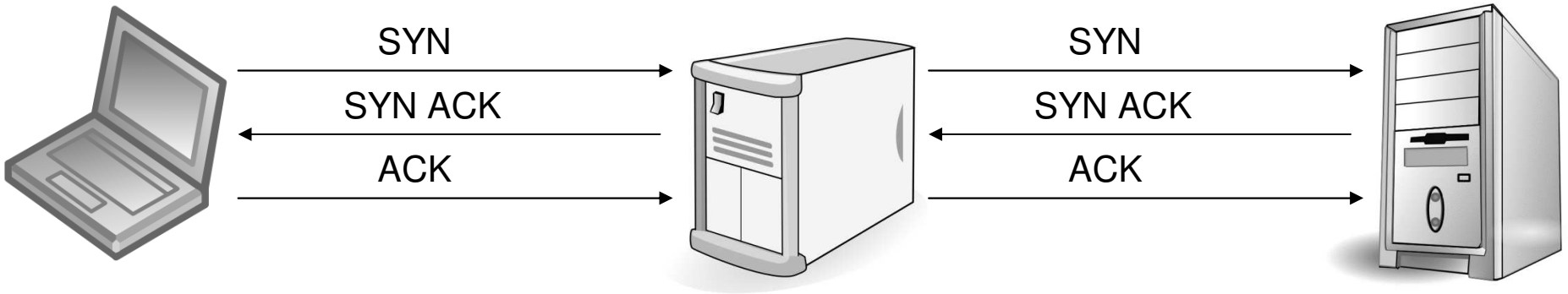
Outline

- Introduction to NATs
- NAT Behavior (actual and recommended)
  - UDP
  - TCP
- IAB UNSAF Considerations
- STUN
- STUN Relay Usage
- NAT Traversal in SIP
  - SIP Response Routing
  - User Agent Reachability
  - ICE

Connection Establishment

- Three-way handshake
  - MUST be supported
- Simultaneous open
  - MUST be supported

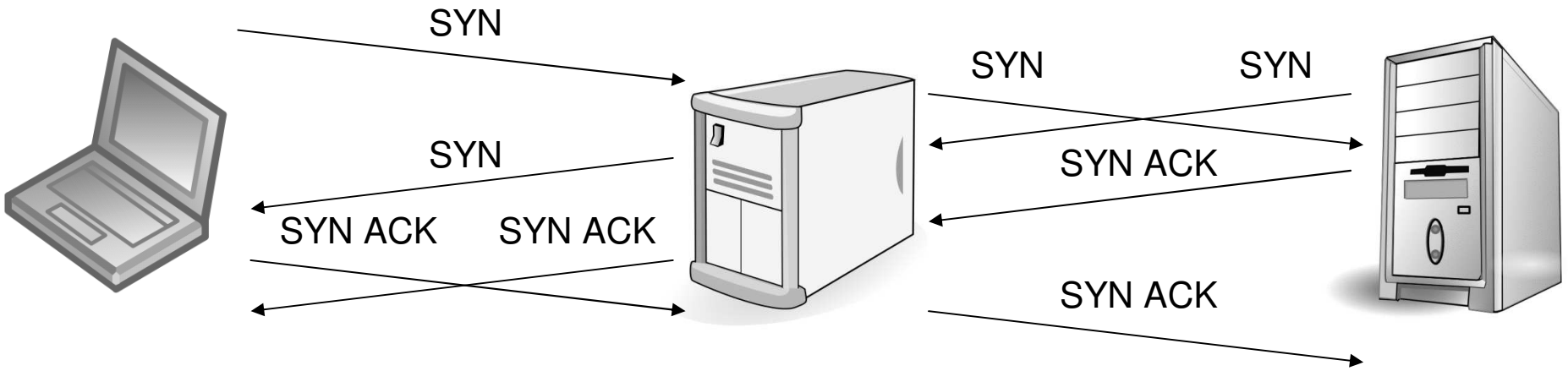
Three-way Handshake



Connection Establishment

- Three-way handshake
  - MUST be supported
- Simultaneous open
  - MUST be supported

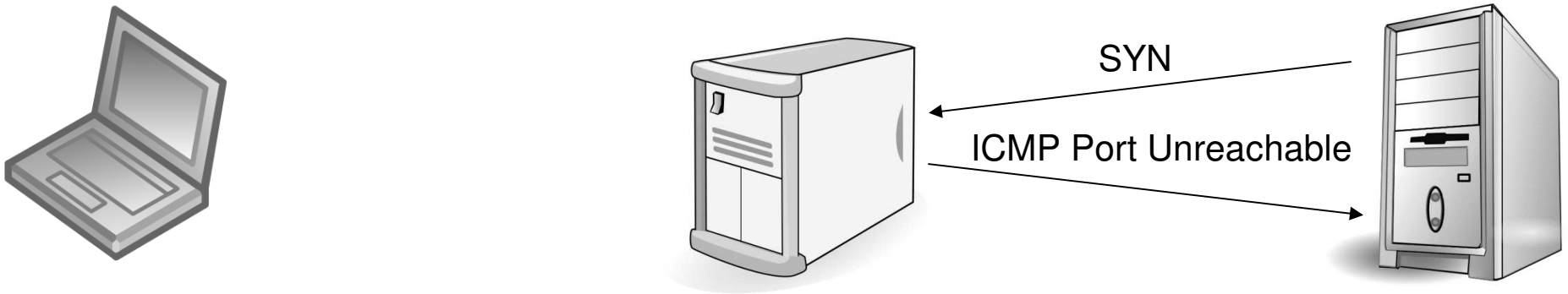
Simultaneous Open



Filtering External SYNs

- Incoming SYS on the external interface.
  - No existing mapping for it
- If the SYN is an error, the NAT should generate an ICMP error message
- If it is a simultaneous open, the NAT should silently drop the packet

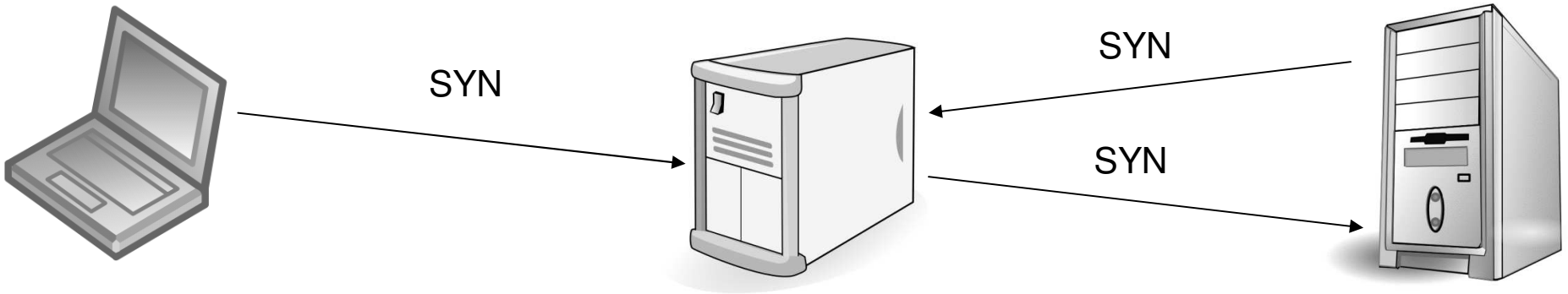
Error Situation



Filtering External SYNs

- Incoming SYS on the external interface.
  - No existing mapping for it
- If the SYN is an error, the NAT should generate an ICMP error message
- If it is a simultaneous open, the NAT should silently drop the packet

Simultaneous Open



Filtering External SYNs

- RECOMMENDED to respond to unsolicited SYNs
  - With an ICMP error with a delay of at least 6 seconds
  - If a matching outbound SYN is received, cancel the sending of the ICMP error

NAT Session Timeout

- Established connections
  - MUST NOT be less than 2 hours and 4 minutes
  - TCP sends keep alives every 4 hours
- Partially opened or partially closed connections
  - MUST NOT be less than 4 minutes
- TIME_WAIT timeout not specified

Outline

- Introduction to NATs
- NAT Behavior (actual and recommended)
  - UDP
  - TCP
- IAB UNSAF Considerations
- STUN
- STUN Relay Usage
- NAT Traversal in SIP
  - SIP Response Routing
  - User Agent Reachability
  - ICE

UNSAF

- Unilateral self-address fixing
- An application needs to refer to itself with a valid IP address
- Problems
  - Reflectors may be in a different domain than the destination
  - Solutions may circumvent security
  - NAT behavior is not deterministic
  - Midcom has not been successful
- Solutions need to have an exit strategy and a limited scope
- Final goal is to get rid of NATs

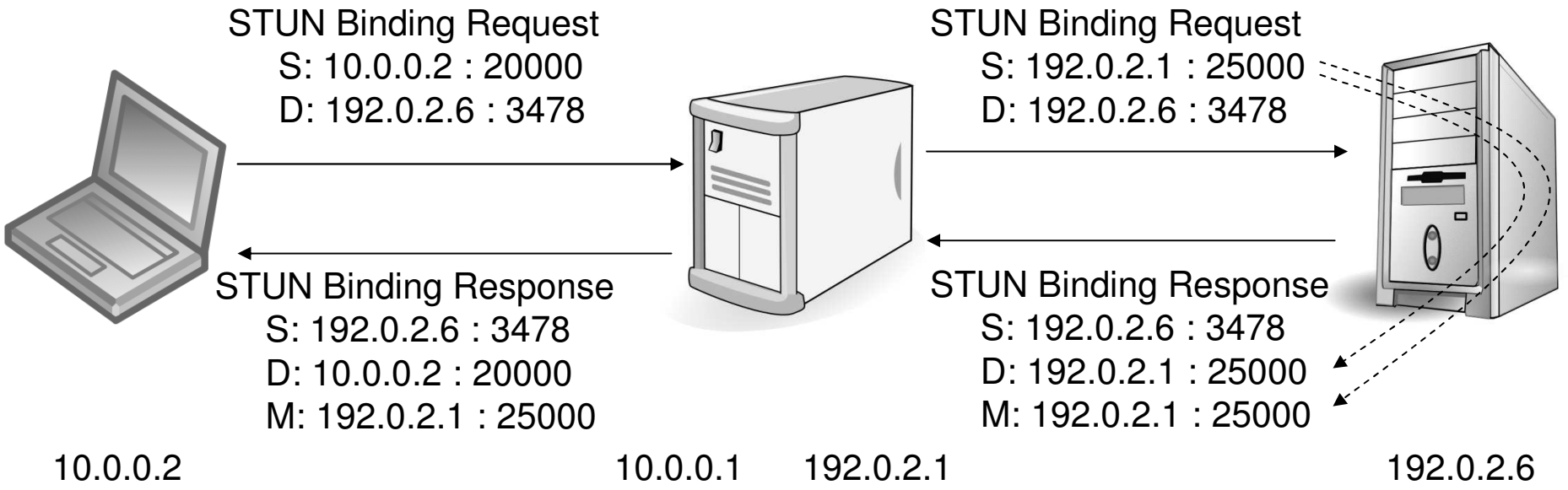
Outline

- Introduction to NATs
- NAT Behavior (actual and recommended)
  - UDP
  - TCP
- IAB UNSAF Considerations
- STUN
- STUN Relay Usage
- NAT Traversal in SIP
  - SIP Response Routing
  - User Agent Reachability
  - ICE

Introduction to STUN

- Originally a protocol between endpoints and reflectors
- Revised specification defines usages
  - Binding discovery
  - NAT keep-alives
  - Short-term password
  - Relay (previously known as TURN)
- TLV encoded
- Can run on UDP, TCP, or TLS/TCP
- STUN server located using DNS SRV
- Transactions
  - Request/response
  - Indications (not delivered reliably)
- Can be multiplexed with other protocols
  - Two first bits are zeros
  - Magic cookie
  - FIGERPRINT attribute

Binding Discovery



XOR-MAPPED-ADDRESS

- In addition to the MAPPED-ADDRESS attribute
- Some NATs inspect packets and translate IP addresses known to them

Outline

- Introduction to NATs
- NAT Behavior (actual and recommended)
  - UDP
  - TCP
- IAB UNSAF Considerations
- STUN
- STUN Relay Usage
- NAT Traversal in SIP
  - SIP Response Routing
  - User Agent Reachability
  - ICE

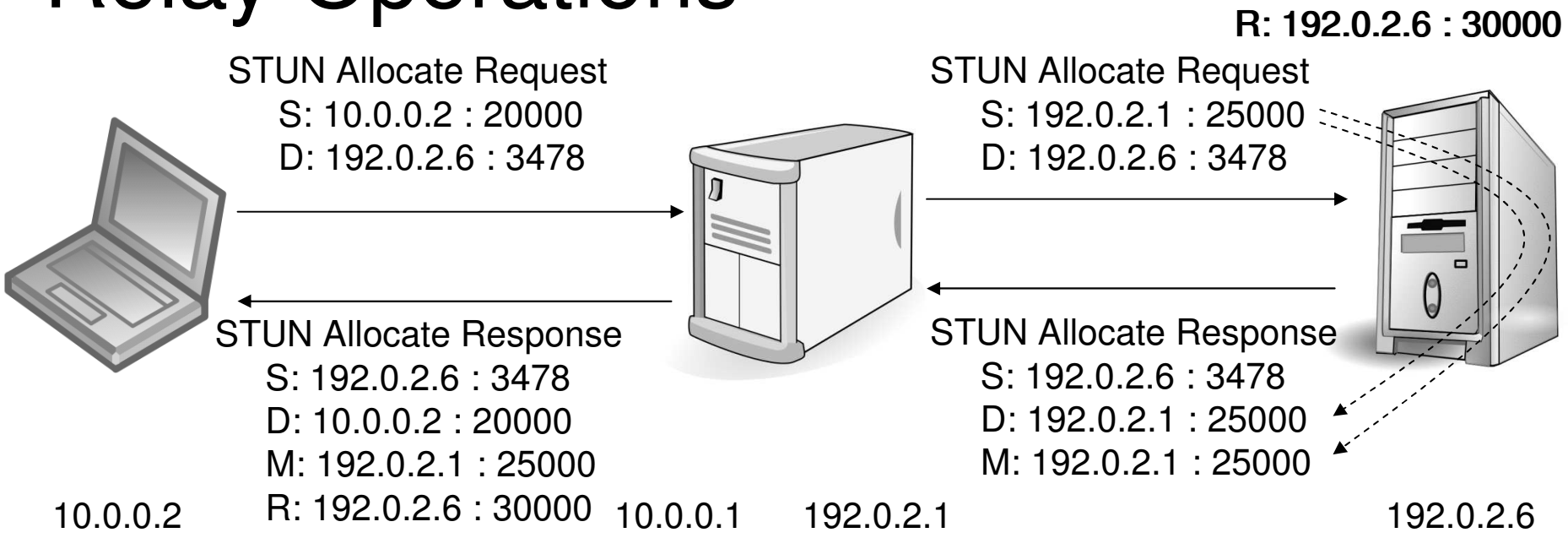
Introduction to STUN Relay Usage

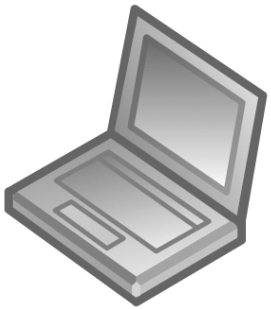
- Previously known as TURN
- Allocate request / response
  - Allocate an external address at the relay
  - Responses carry a MAPPED-ADDRESS
- Send indication
  - Send data to a remote endpoint through the relay
- Data indication
  - Data received from remote endpoints through the relay
- Set Active Destination request / response
  - Send and receive data to and from a single remote endpoint without using Send and Data wrappers
- Connect request / response
  - Requests the relay to establish a TCP connection with the remote endpoint
- Connection Status Indication
  - The relay informs the endpoint about the status of a TCP connection with the remote endpoint
  - LISTEN, ESTABISHED, CLOSED

Transport Protocols

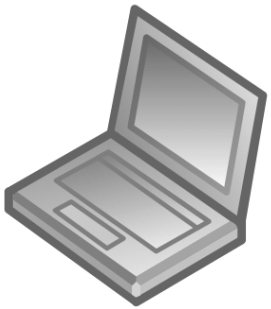
- STUN relay clients can use UDP, TCP, or TLS/TCP
  - UDP to TCP is not supported

Relay Operations





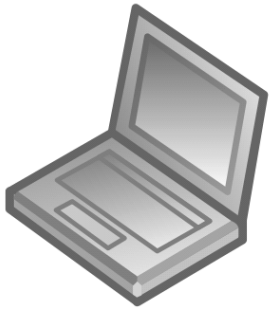
192.0.2.4

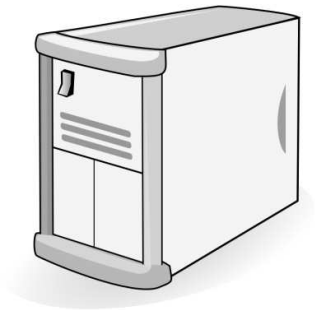


192.0.2.5

Relay Operations

R: 192.0.2.6 : 30000





Packet Dropped

10.0.0.2

10.0.0.1 192.0.2.1

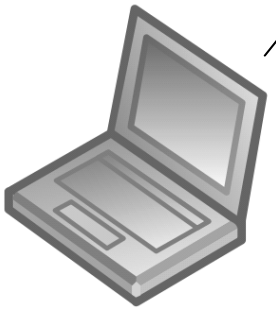
192.0.2.6

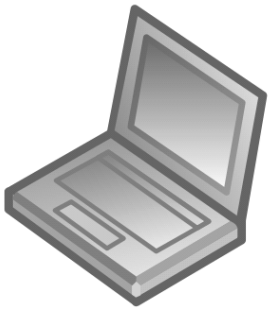
S: 192.0.2.4 : 27000
D: 192.0.2.6 : 30000

The client needs to send data to the remote endpoint in order to use the relay

Equivalent to a NAT with:

- Address dependent filtering policy
- Endpoint independent mapping

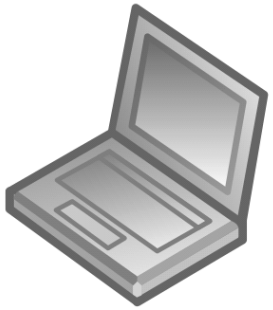


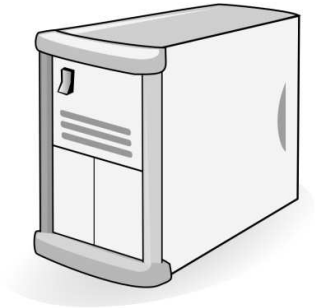


192.0.2.4

192.0.2.5

Relay Operations





R: 192.0.2.6 : 30000

Packet Dropped

10.0.0.2

10.0.0.1 192.0.2.1

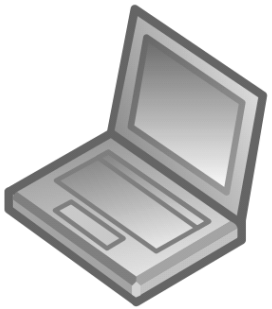
192.0.2.6

S: 192.0.2.5 : 27000
D: 192.0.2.6 : 30000

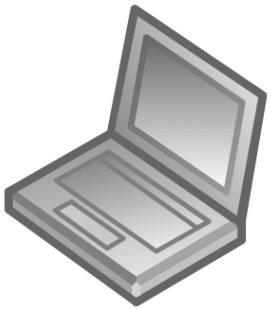
The client needs to send data to the remote endpoint in order to use the relay

Equivalent to a NAT with:

- Address dependent filtering policy
- Endpoint independent mapping

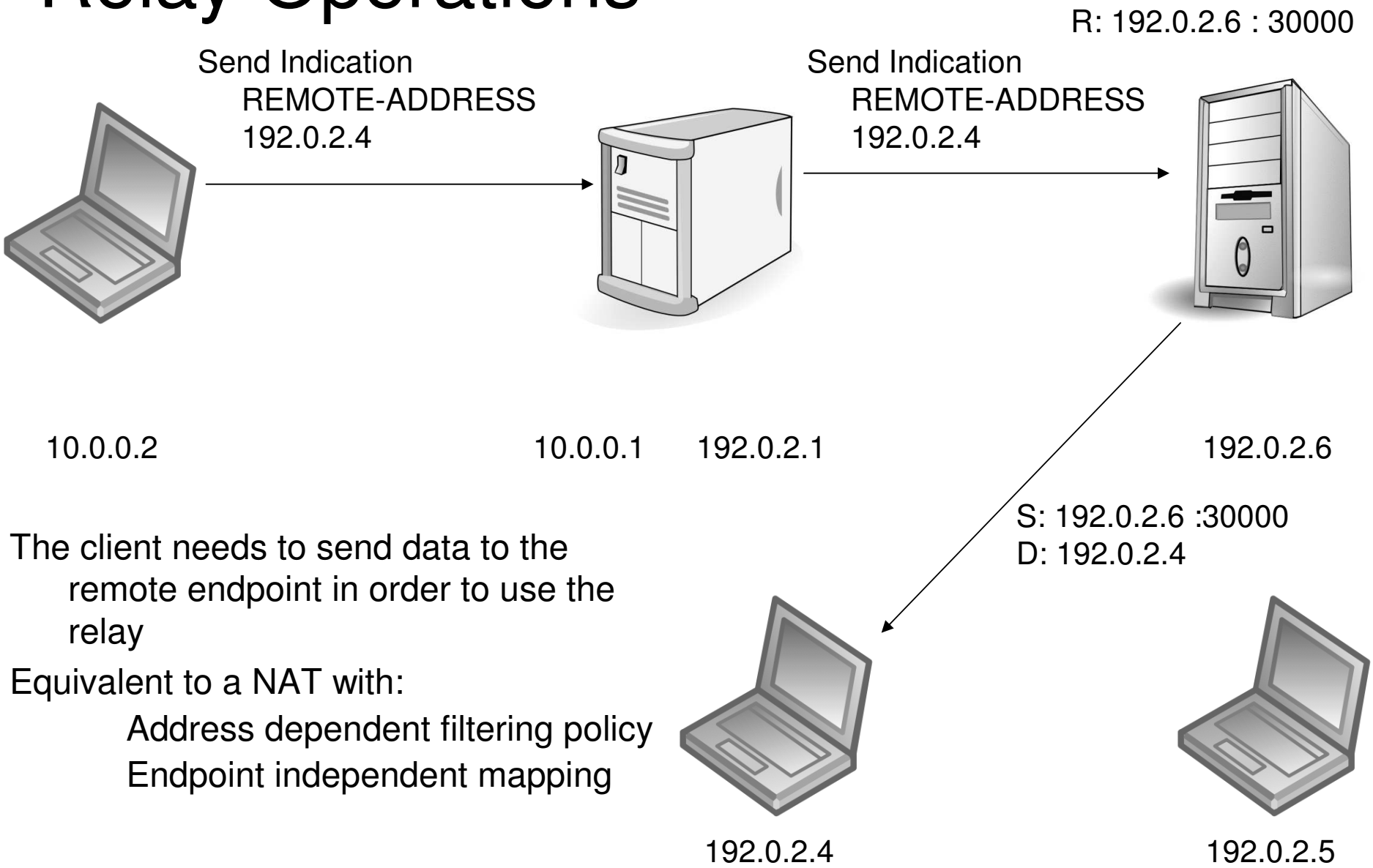


192.0.2.4



192.0.2.5

Relay Operations

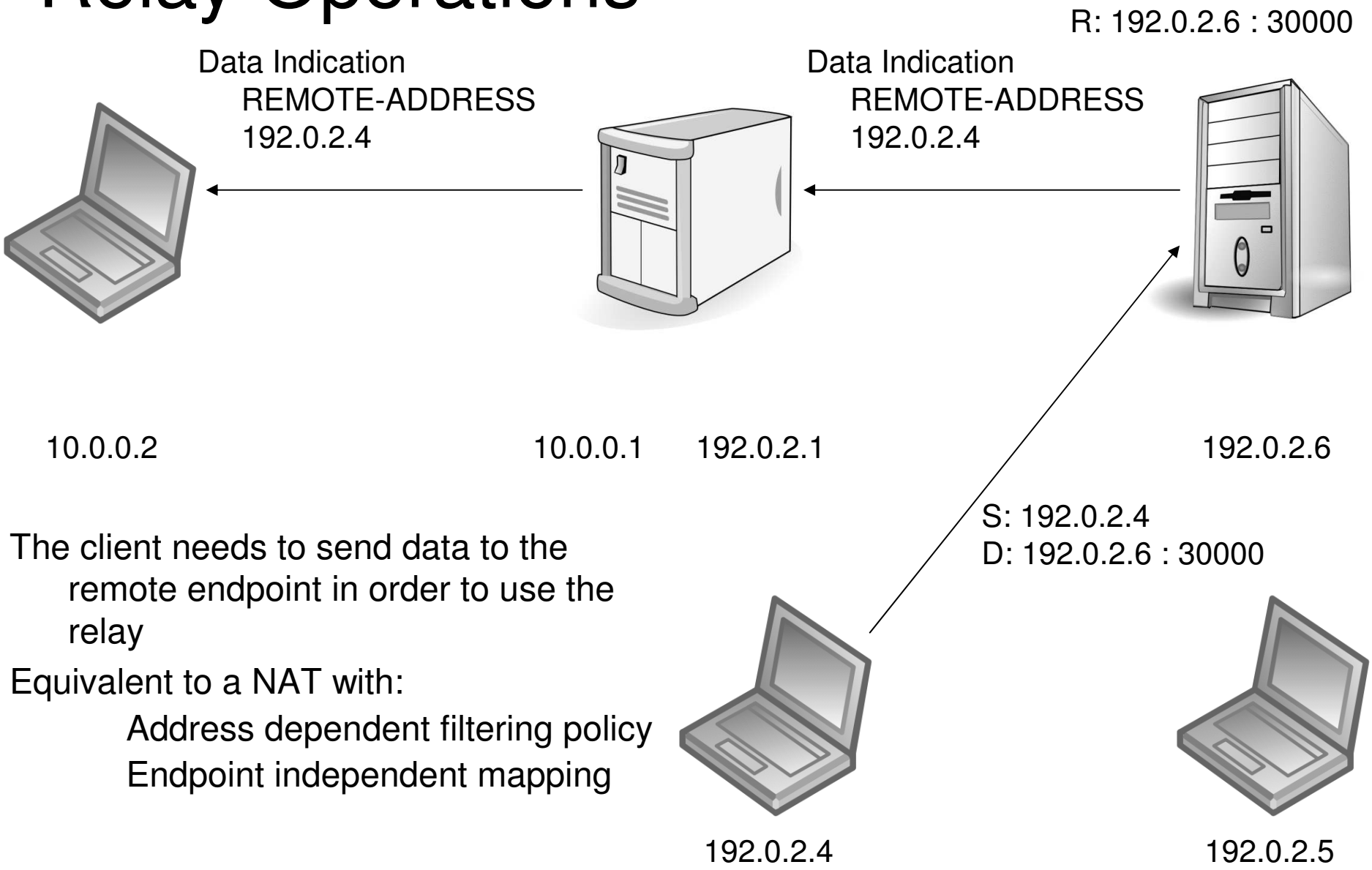


The client needs to send data to the remote endpoint in order to use the relay

Equivalent to a NAT with:

- Address dependent filtering policy
- Endpoint independent mapping

Relay Operations

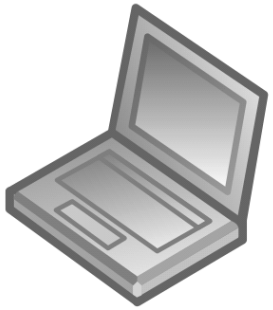


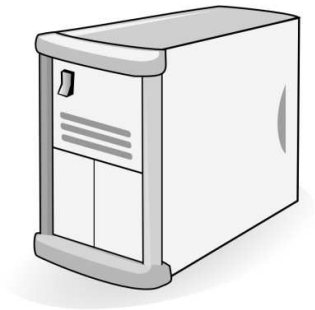
The client needs to send data to the remote endpoint in order to use the relay

Equivalent to a NAT with:

- Address dependent filtering policy
- Endpoint independent mapping

Relay Operations





R: 192.0.2.6 : 30000

Packet Dropped

10.0.0.2

10.0.0.1 192.0.2.1

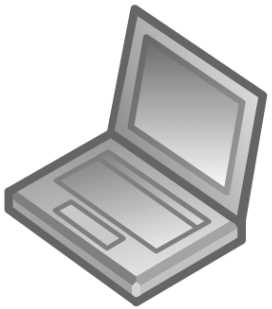
192.0.2.6

S: 192.0.2.5 : 27000
D: 192.0.2.6 : 30000

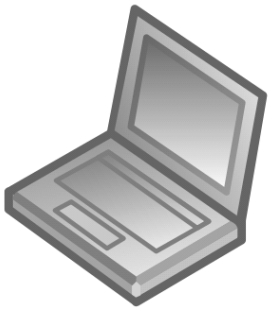
The client needs to send data to the remote endpoint in order to use the relay

Equivalent to a NAT with:

- Address dependent filtering policy
- Endpoint independent mapping

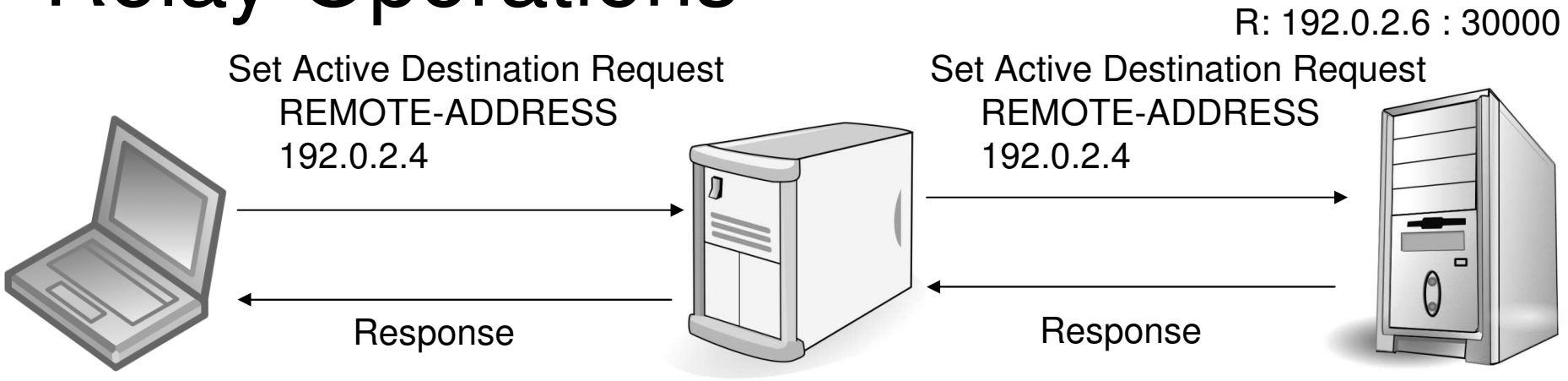


192.0.2.4



192.0.2.5

Relay Operations

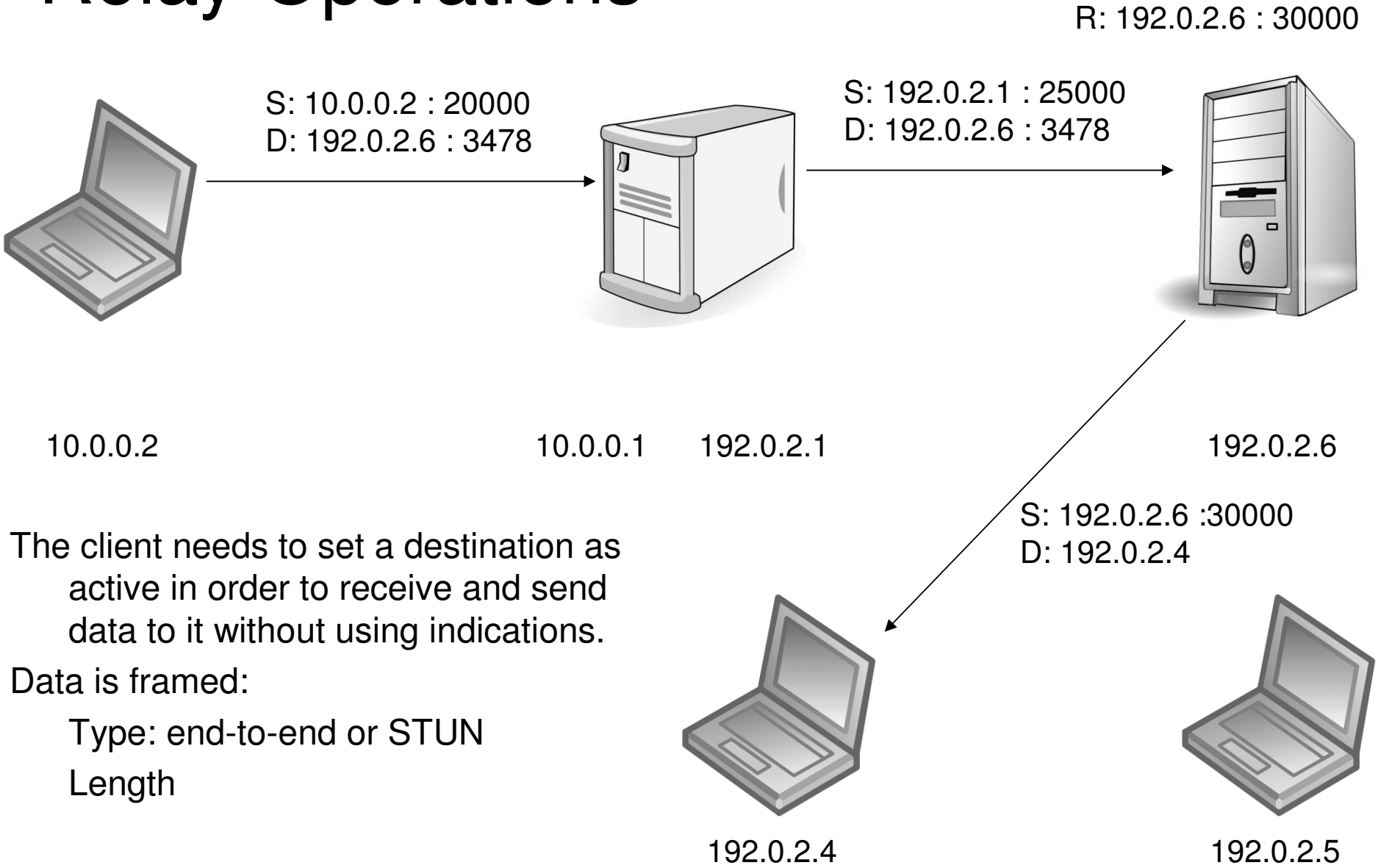


The client needs to set a destination as active in order to receive and send data to it without using indications.

Data is framed:

- Type: end-to-end or STUN
- Length

Relay Operations

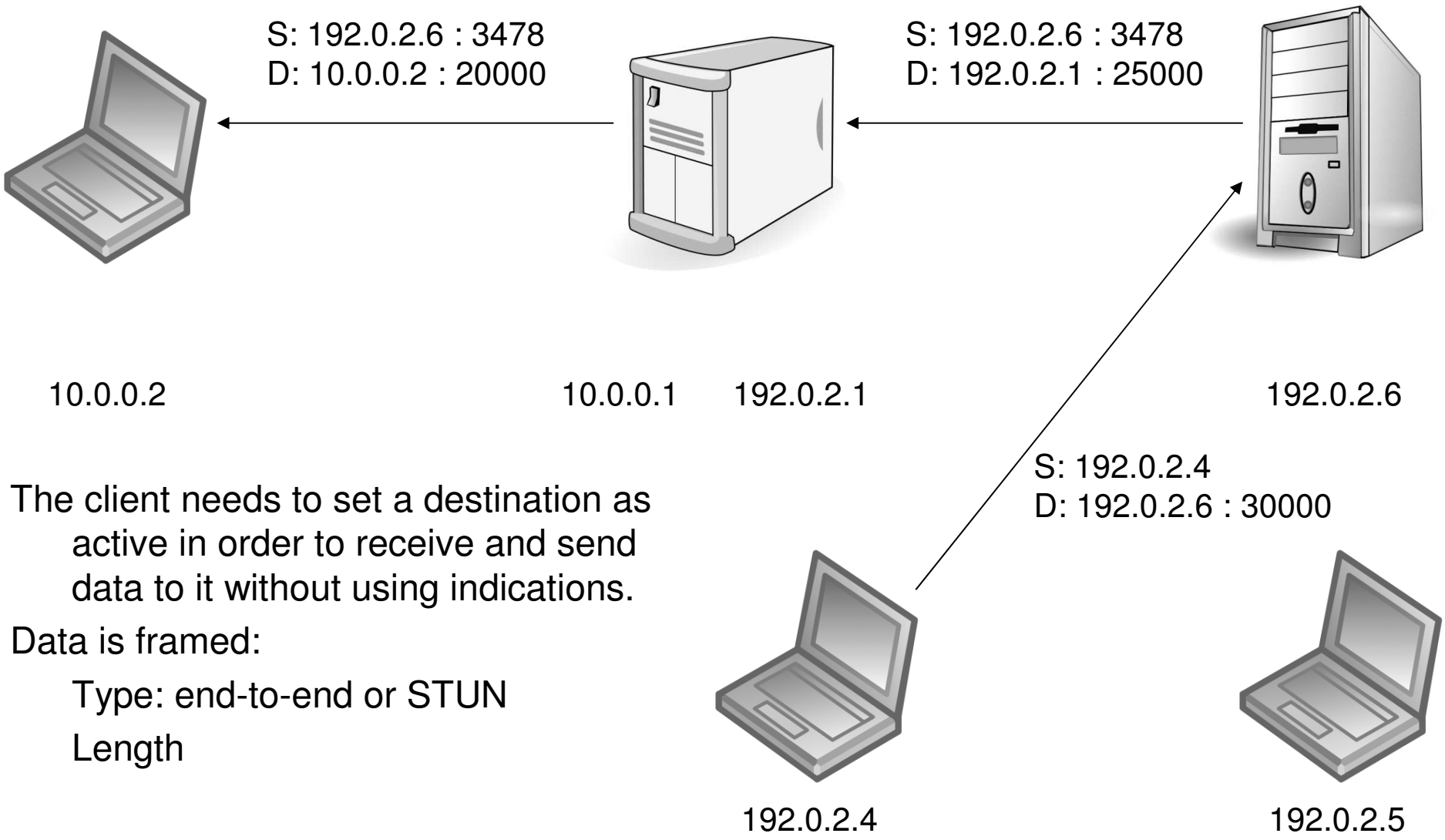


The client needs to set a destination as active in order to receive and send data to it without using indications.

Data is framed:

- Type: end-to-end or STUN
- Length

Relay Operations



The client needs to set a destination as active in order to receive and send data to it without using indications.

Data is framed:

- Type: end-to-end or STUN
- Length

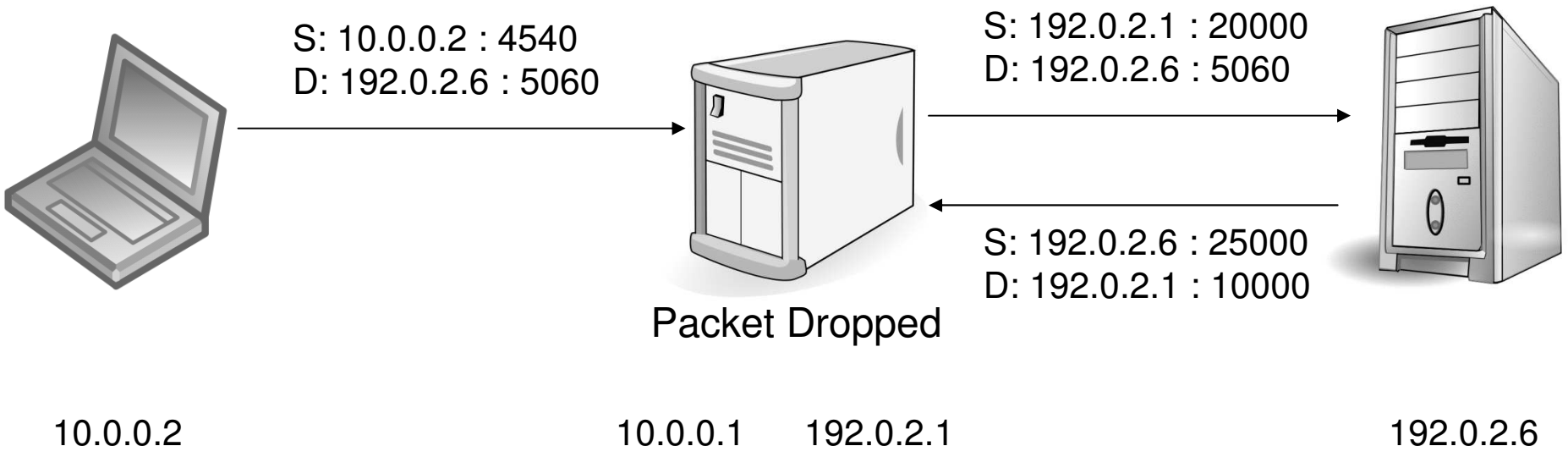
Outline

- Introduction to NATs
- NAT Behavior (actual and recommended)
  - UDP
  - TCP
- IAB UNSAF Considerations
- STUN
- STUN Relay Usage
- NAT Traversal in SIP
  - SIP Response Routing
  - User Agent Reachability
  - ICE

Regular SIP Response Routing

- Responses are sent to
  - Request's source IP address
  - Port in the Via entry
- Responses are sent from
  - Typically the same IP address as the request was sent to
  - Any port

Regular SIP Response Routing

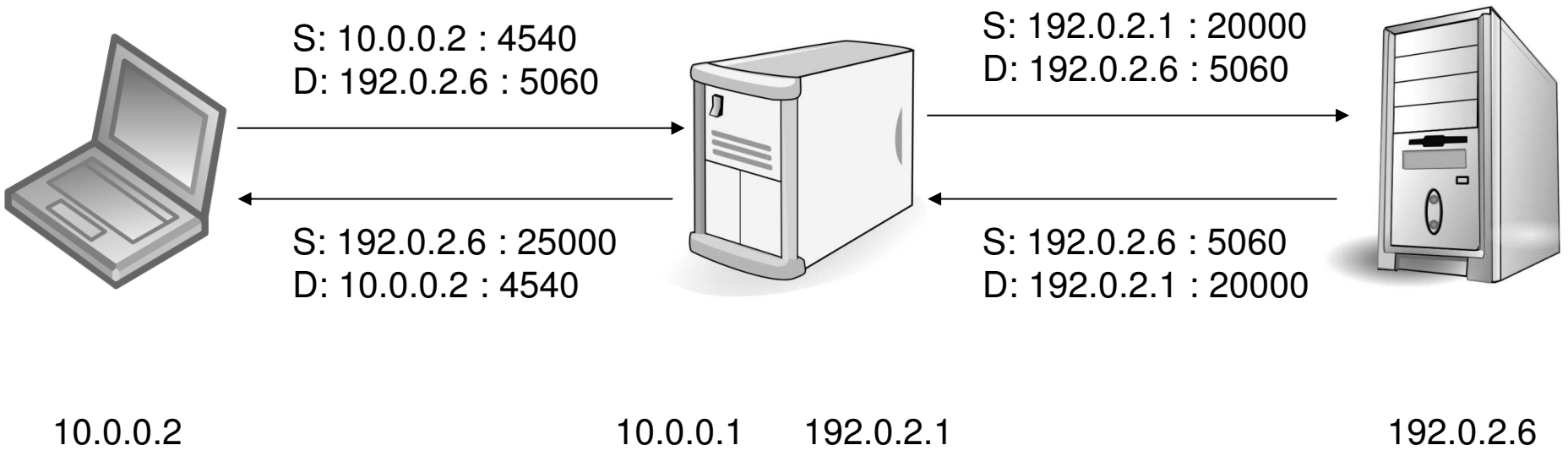


- Request
  - Via: SIP/2.0/UDP 10.0.0.2:10000;branch=z9hG4bKkjshdyff
- Response
  - Via: SIP/2.0/UDP 10.0.0.2:10000;
    received=192.0.2.1;branch=z9hG4bKkjshdyff

Symmetric SIP Response Routing

- Responses are sent to
  - Request's source IP address
  - Request's source port
- Responses are sent from
  - IP address the request was sent to
  - The port the request was sent to

Symmetric SIP Response Routing



- Request
  - Via: SIP/2.0/UDP 10.0.0.2:4540; rport;branch=z9hG4bKkjshdyff
- Response
  - Via: SIP/2.0/UDP 10.0.0.2:4540; received=192.0.2.1;rport=20000;branch=z9hG4bKkjshdyff

Outline

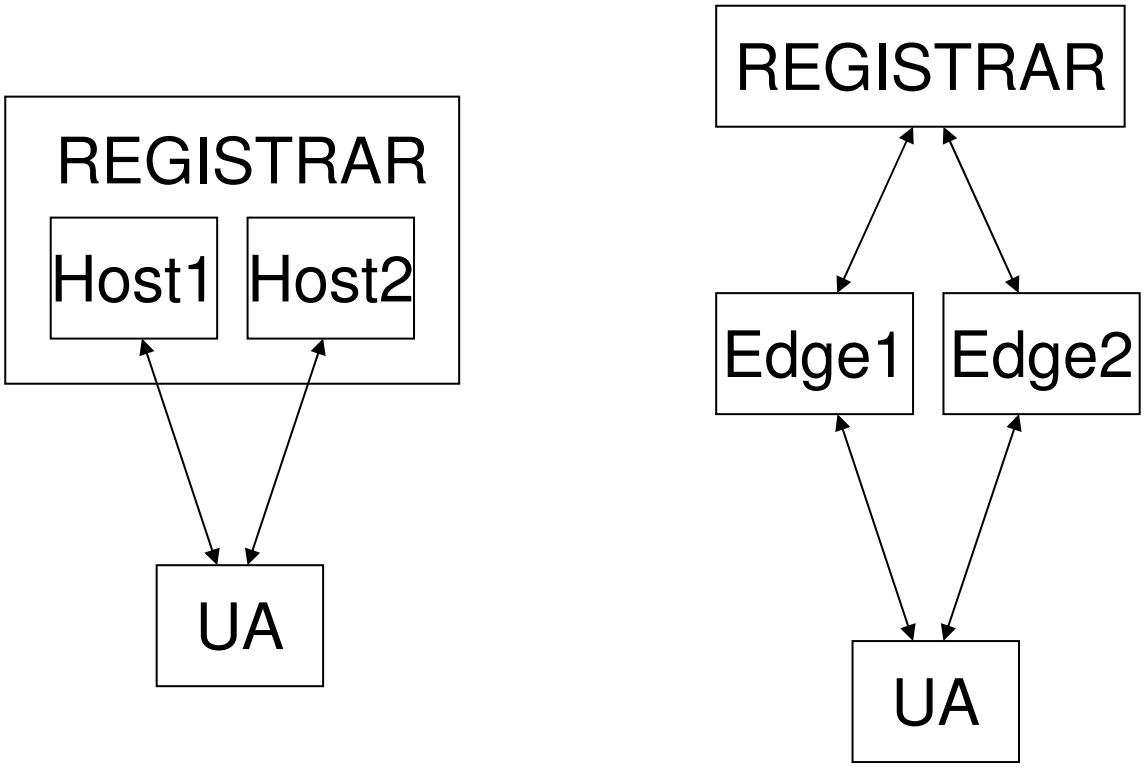
- Introduction to NATs
- NAT Behavior (actual and recommended)
  - UDP
  - TCP
- IAB UNSAF Considerations
- STUN
- STUN Relay Usage
- NAT Traversal in SIP
  - SIP Response Routing
  - User Agent Reachability
  - ICE

User Agent Reachability

- Proxy to user agent connections
  - Problems with NATs and firewalls
- User agent always starts the connection
- Keep alives to keep NAT bindings up
  - Different alternatives analyzed
  - STUN
    - Multiplexed with SIP
    - New registration if a new binding is discovered
- User agent identification (e.g., for service profiles)
  - Instance identifier

Reliability and Scalability

- Different flows identified by their registration IDs



Outline

- Introduction to NATs
- NAT Behavior (actual and recommended)
  - UDP
  - TCP
- IAB UNSAF Considerations
- STUN
- STUN Relay Usage
- NAT Traversal in SIP
  - SIP Response Routing
  - User Agent Reachability
  - ICE

Introduction to ICE

- Endpoints gather all the addresses they can
- They run connectivity checks between them
- They choose the highest priority pair that works

Gathering Addresses

- Address types
  - Host candidates
  - Server-reflexive candidates
  - Relayed candidates
  - Peer-reflexive candidates
- A candidate's base: the address used to send data
  - The base for a reflexive candidate is a host candidate
- Duplicated addresses are removed
  - Candidates with the same transport address but different base are considered different
- Foundation: used to freeze addresses (related to connectivity checks)
  - Same type
  - Bases with the same IP address
  - Same STUN server

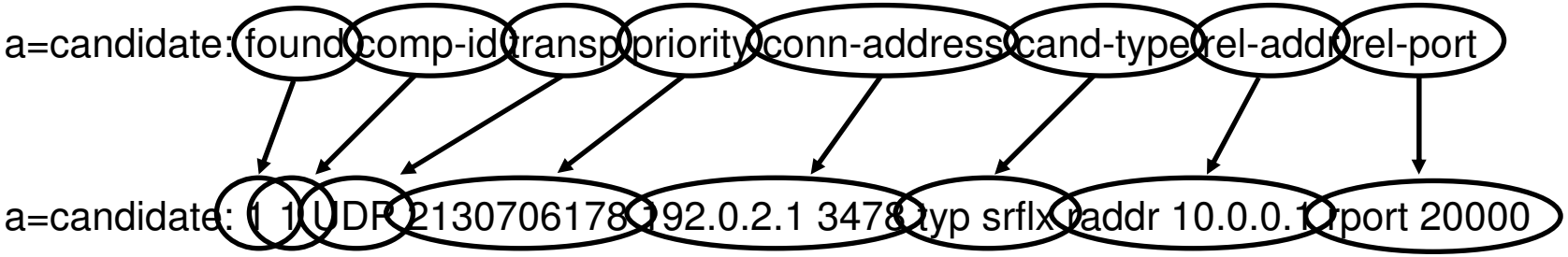
Prioritizing Addresses

Priority = 2^{24} (type preference) +
2^{8} (local preference) +
2 (256 – component ID)

- Type preference [0-126]: preference for the type of candidate (e.g., server reflexive)
- Local preference [0-65535]: preference for the interface the candidate was obtained from (e.g., multihomed hosts)
- Component ID [1-256]: 1 for RTP and 2 for RTCP

Generation of an offer

- The candidate with the highest probability to work goes into the m and c lines
  - A relayed address initially
- The rest of the candidates go into ‘candidate’ attributes



- User name and password for connectivity checks in ‘ice-ufrag’ and ‘ice-pwd’ attributes

Other transports are:
- tcp-so
- tcp-act
- tcp-pass

Related address: base

Prioritizing Pairs

- After the offer/answer exchange

Pair Priority = 2^{32} MIN(O-P,A-P) + 2 MIN(O-P,A-P) + (O-P>A-P:1?0)

- O-P: priority in the offer
- A-P: priority in the answer

Connectivity Checks

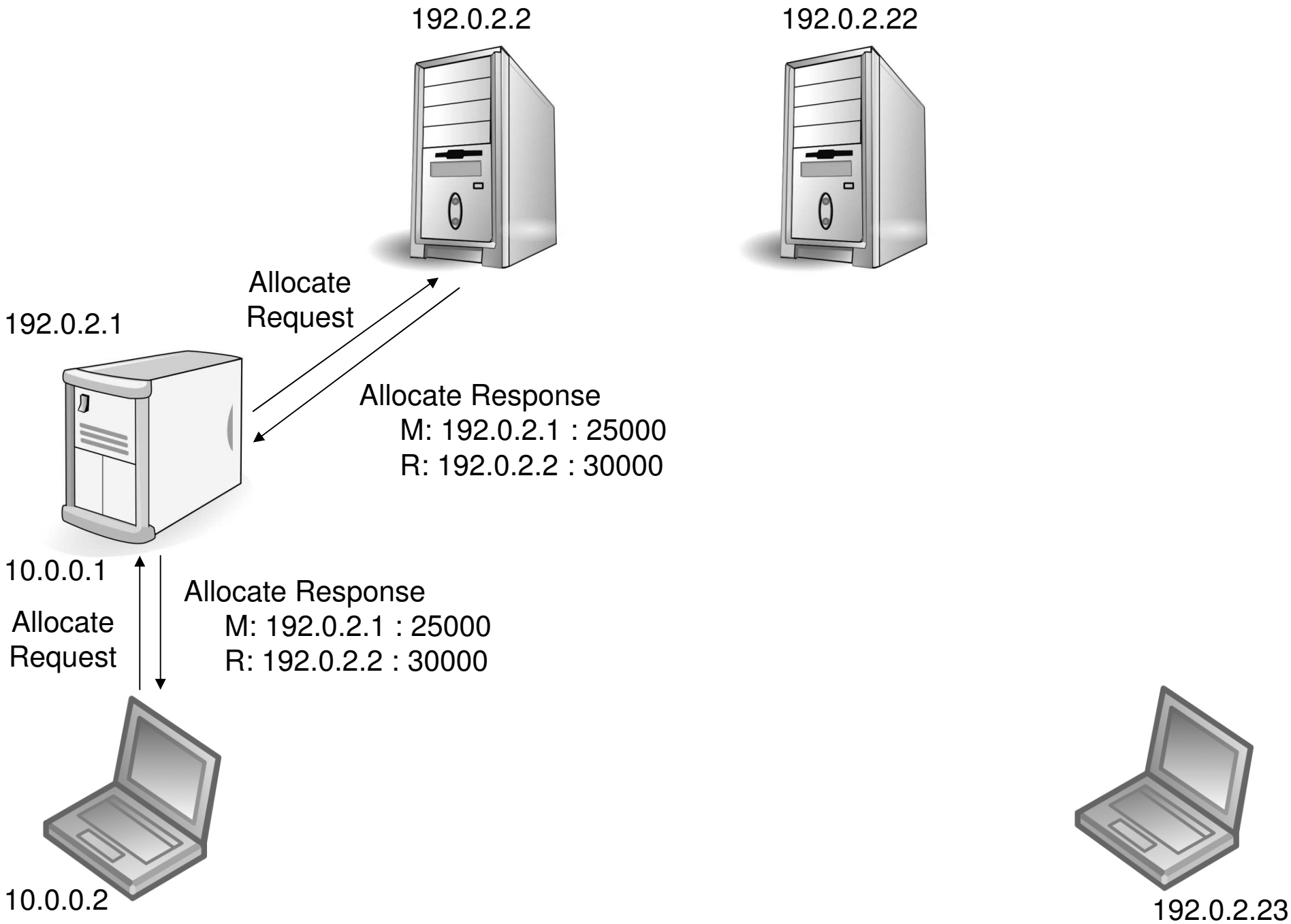
- Five states for a pair:
  - Waiting, in progress, succeeded, failed, frozen
- Periodic checks and triggered checks
  - Periodic checks performed in priority order
- Connectivity is checked with STUN Binding Requests
  - Carry a concatenation of user names and the remote password

ICE Roles

- Controlling agent
  - Selects which pair to use
  - USE-CADIDATE attribute
    - Default algorithm: included in every check
  - Agent that generates the initial offer
- Passive-only agents
  - They know they are not behind a NAT
    - E.g., PSTN gateways, conferencing servers
  - Include ‘a=ice-passive’ in their session descriptions
  - Respond to checks
  - Generate triggered checks
  - Generate keepalives

ICE Example (1)

- One endpoint is behind a NAT
- One endpoint has a public IP address
- Endpoints use STUN servers that support the relay usage

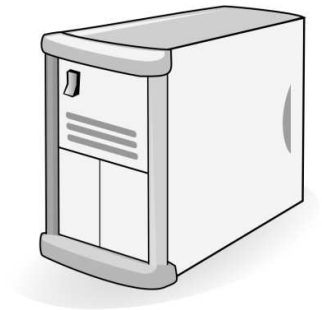

192.0.2.2
192.0.2.22
Allocate
Request
192.0.2.1
Allocate Response
M: 192.0.2.1 : 25000
R: 192.0.2.2 : 30000
10.0.0.1
Allocate
Request
Allocate Response
M: 192.0.2.1 : 25000
R: 192.0.2.2 : 30000
10.0.0.2
192.0.2.23

192.0.2.2

192.0.2.22

Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000

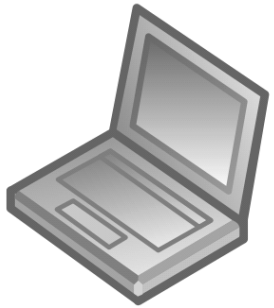
192.0.2.1



10.0.0.1

Allocate Response
M: 192.0.2.1 : 25000
R: 192.0.2.2 : 30000

INVITE (offer)



10.0.0.2

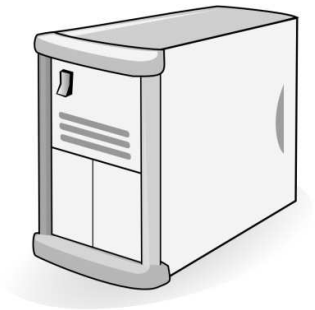
192.0.2.23

Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000

192.0.2.2

192.0.2.22

192.0.2.1

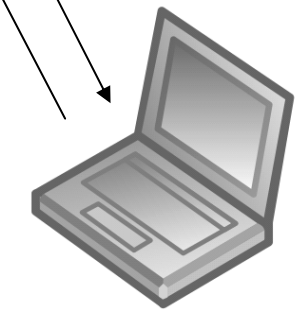


10.0.0.1

Allocate
Request

Allocate Response
M: 192.0.2.23 : 35000
R: 192.0.2.22 : 45000

10.0.0.2



192.0.2.23

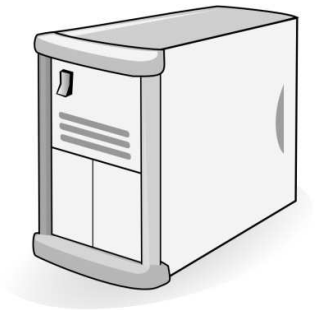
192.0.2.2

Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000

192.0.2.22

Host candidate:
192.0.2.23 : 35000
Server reflexive:
192.0.2.23 : 35000
Relayed:
192.0.2.22 : 45000

192.0.2.1



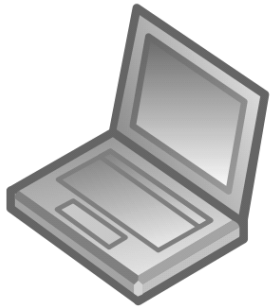
10.0.0.1

Allocate Response
M: 192.0.2.23 : 35000
R: 192.0.2.22 : 45000

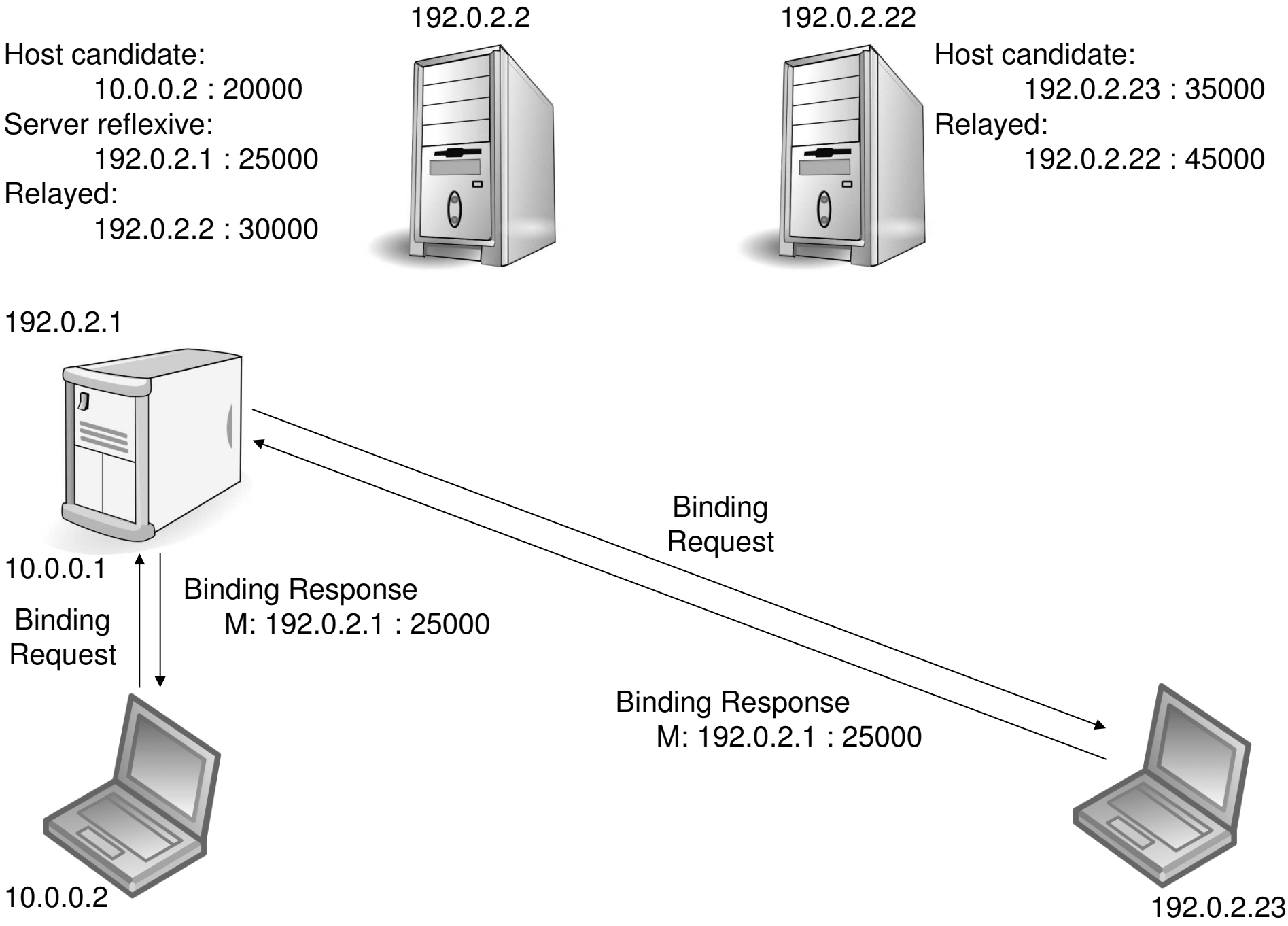
10.0.0.2

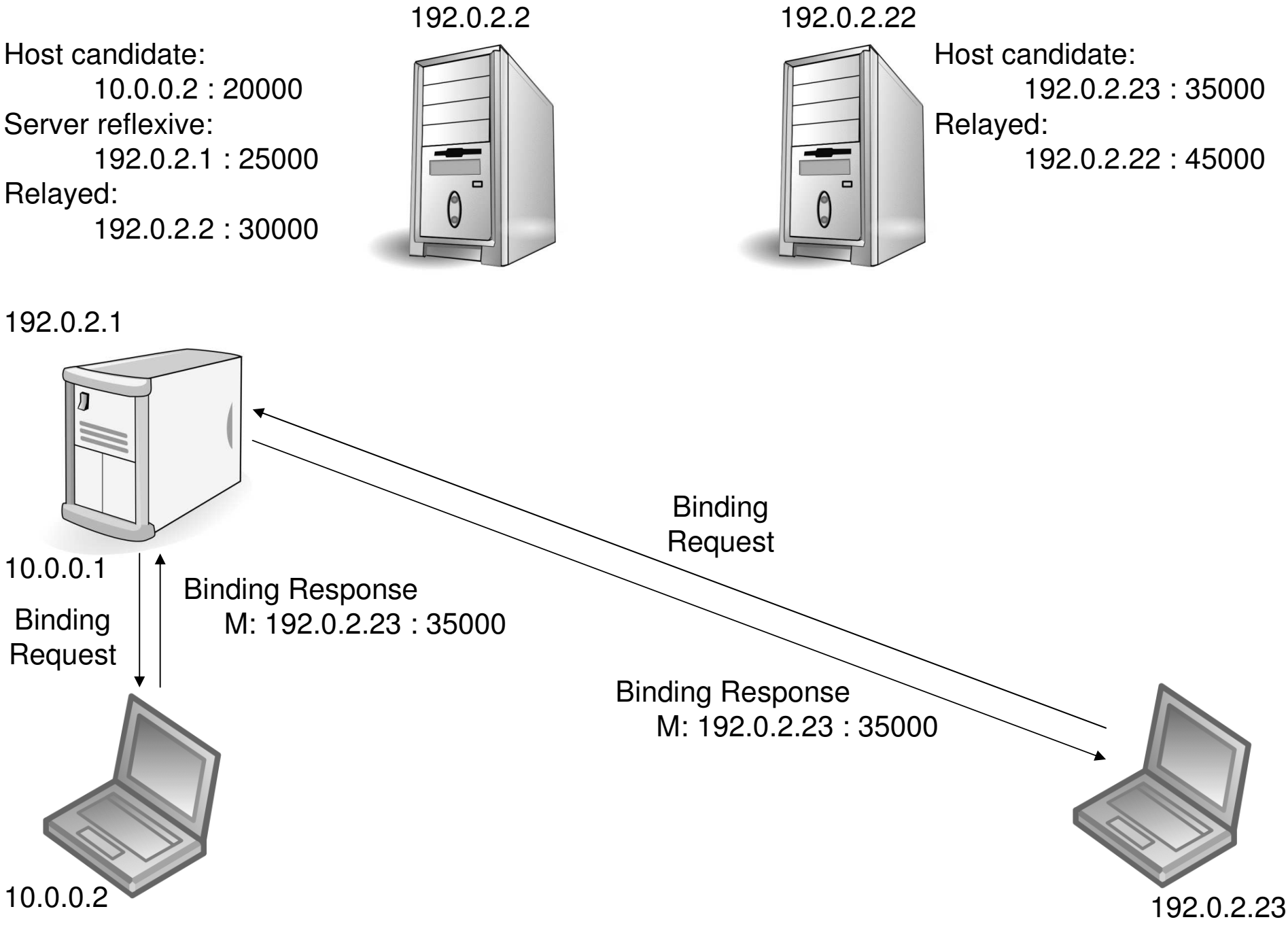
200 OK (answer)

ACK



192.0.2.23


192.0.2.2
192.0.2.22
Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000
Host candidate:
192.0.2.23 : 35000
Relayed:
192.0.2.22 : 45000
192.0.2.1
10.0.0.1
Binding
Request
Binding Response
M: 192.0.2.1 : 25000
Binding
Request
Binding Response
M: 192.0.2.1 : 25000
10.0.0.2
192.0.2.23


192.0.2.2
192.0.2.22
Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000
Host candidate:
192.0.2.23 : 35000
Relayed:
192.0.2.22 : 45000
192.0.2.1
10.0.0.1
Binding
Request
Binding Response
M: 192.0.2.23 : 35000
Binding
Request
Binding Response
M: 192.0.2.23 : 35000
10.0.0.2
192.0.2.23

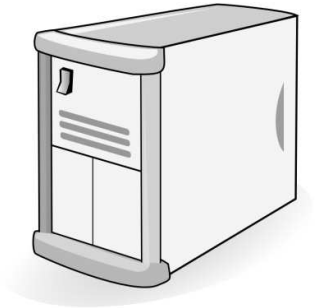
192.0.2.2

Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000

192.0.2.22

Host candidate:
192.0.2.23 : 35000
Relayed:
192.0.2.22 : 45000

192.0.2.1



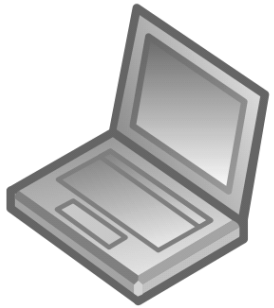
10.0.0.1

10.0.0.2

INVITE (offer)

200 OK (answer)

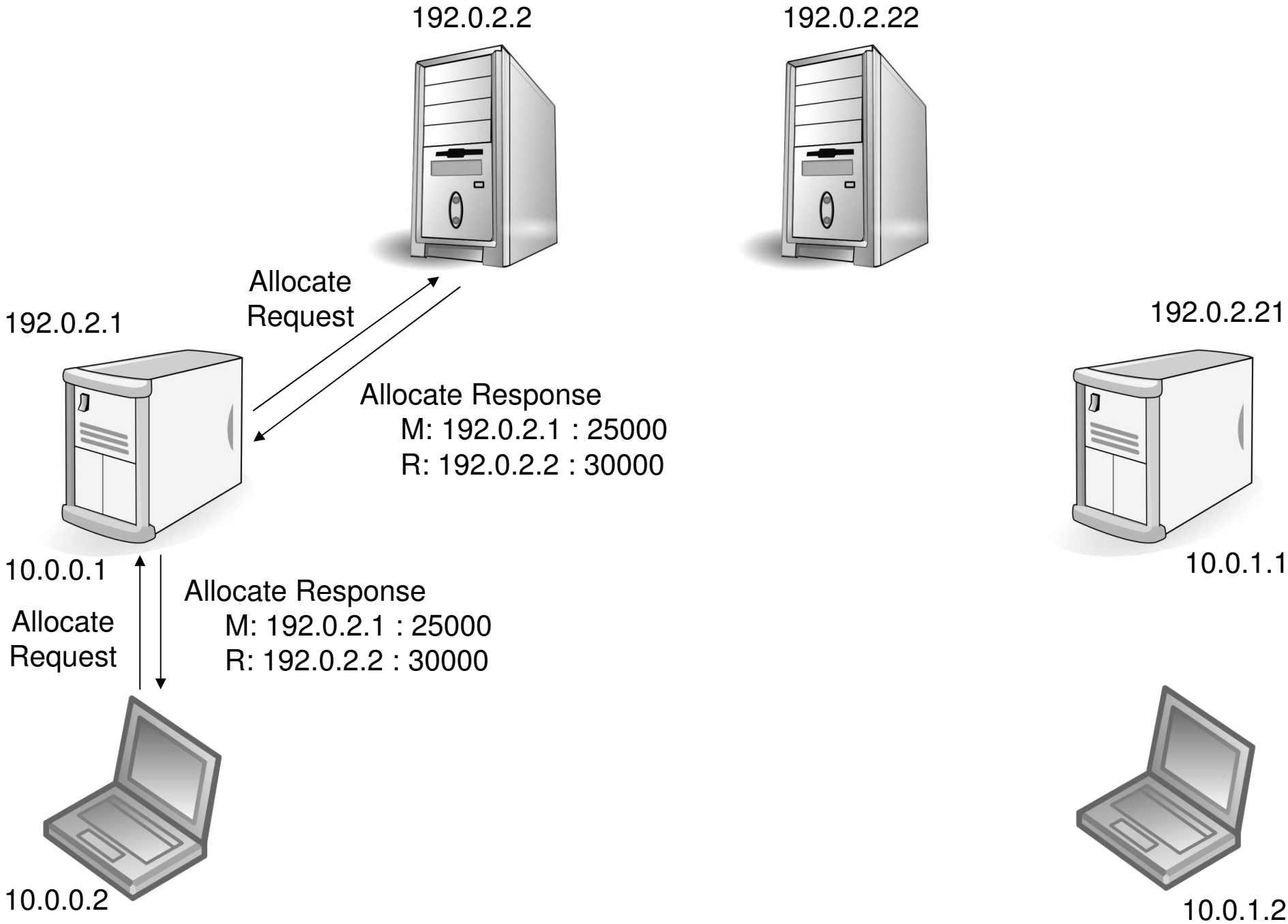
ACK



192.0.2.23

ICE Example (2)

- Both endpoint are behind NATs
- Endpoints use STUN servers that support the relay usage


192.0.2.2
192.0.2.22
Allocate
Request
192.0.2.1
192.0.2.21
Allocate Response
M: 192.0.2.1 : 25000
R: 192.0.2.2 : 30000
10.0.0.1
10.0.1.1
Allocate
Request
Allocate Response
M: 192.0.2.1 : 25000
R: 192.0.2.2 : 30000
10.0.0.2
10.0.1.2

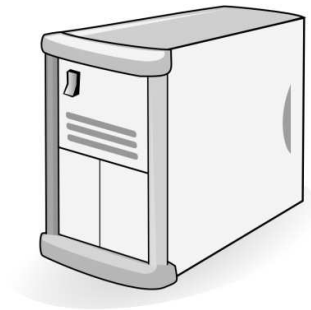
192.0.2.2

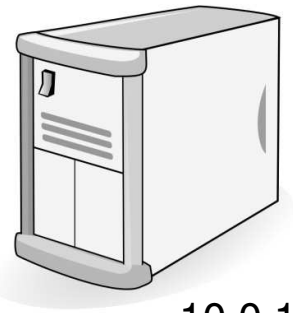
192.0.2.22

Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000

192.0.2.1

192.0.2.21



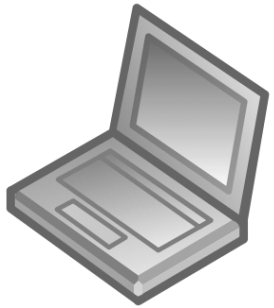


10.0.0.1

10.0.1.1

Allocate Response
M: 192.0.2.1 : 25000
R: 192.0.2.2 : 30000

INVITE (offer)



10.0.0.2

10.0.1.2

Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000

192.0.2.2

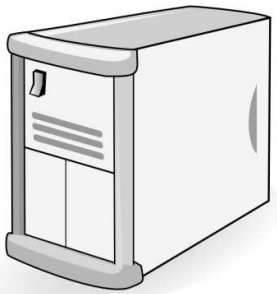
192.0.2.22

Allocate
Request

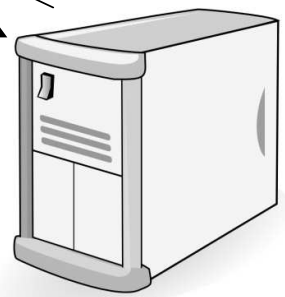
192.0.2.21

Allocate Response
M: 192.0.2.21 : 25000
R: 192.0.2.22 : 30000

192.0.2.1



10.0.0.1

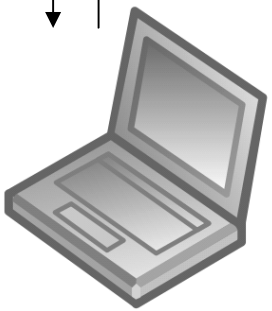


10.0.1.1

Allocate Response
M: 192.0.2.21 : 25000
R: 192.0.2.22 : 30000

Allocate
Request

10.0.0.2



10.0.1.2

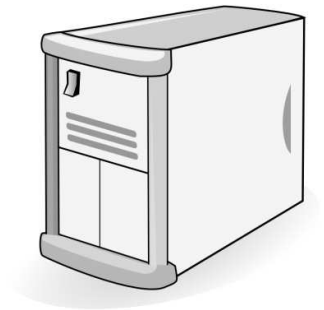
192.0.2.2

Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000

192.0.2.22

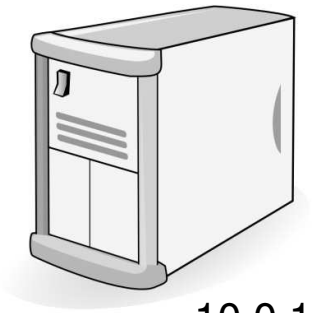
Host candidate:
10.0.1.2 : 20000
Server reflexive:
192.0.2.21 : 25000
Relayed:
192.0.2.22 : 30000

192.0.2.1



10.0.0.1

192.0.2.21



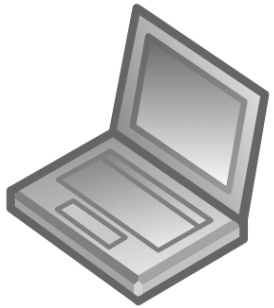
10.0.1.1

Allocate Response
M: 192.0.2.21 : 25000
R: 192.0.2.22 : 30000

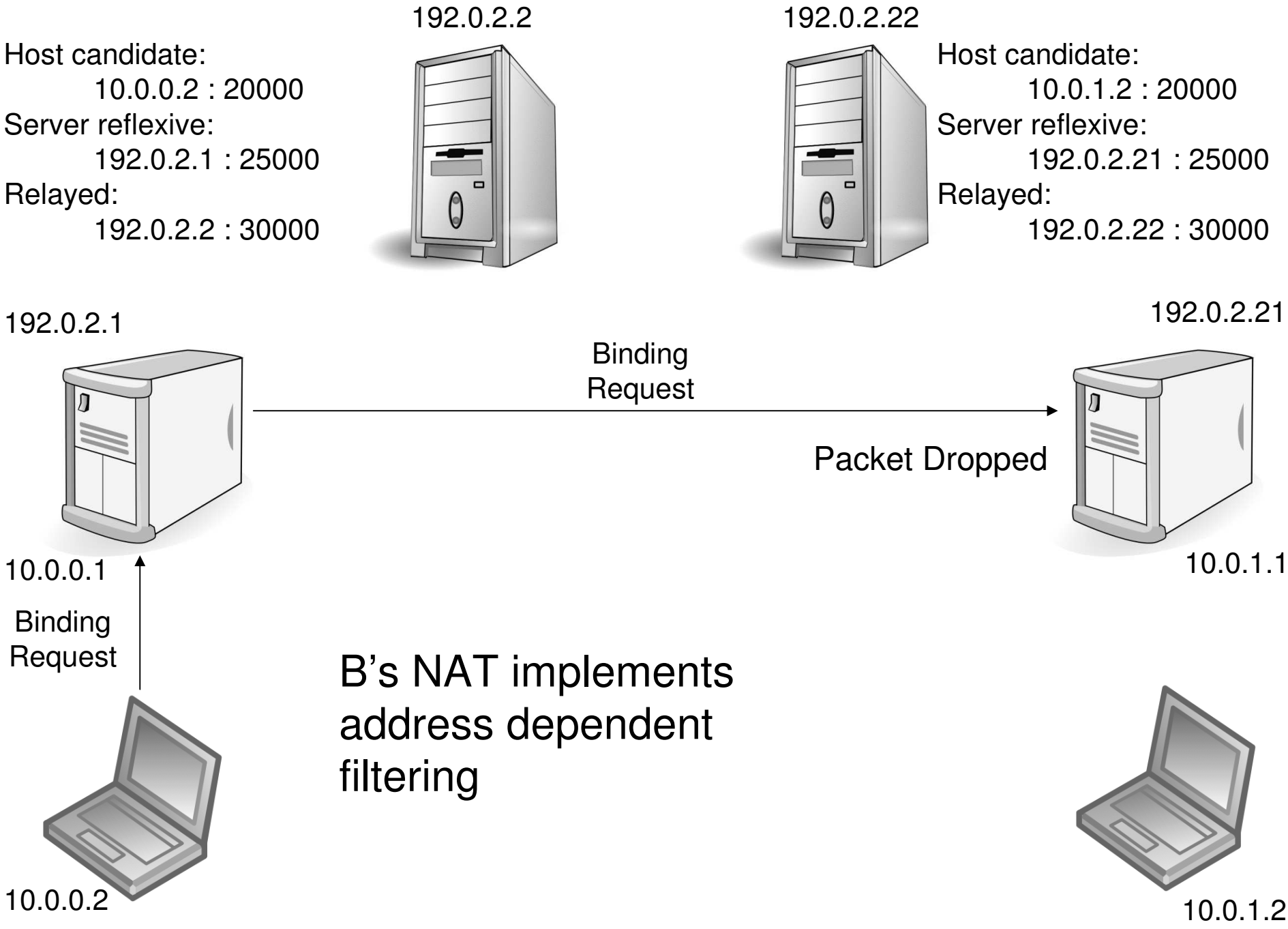
10.0.0.2

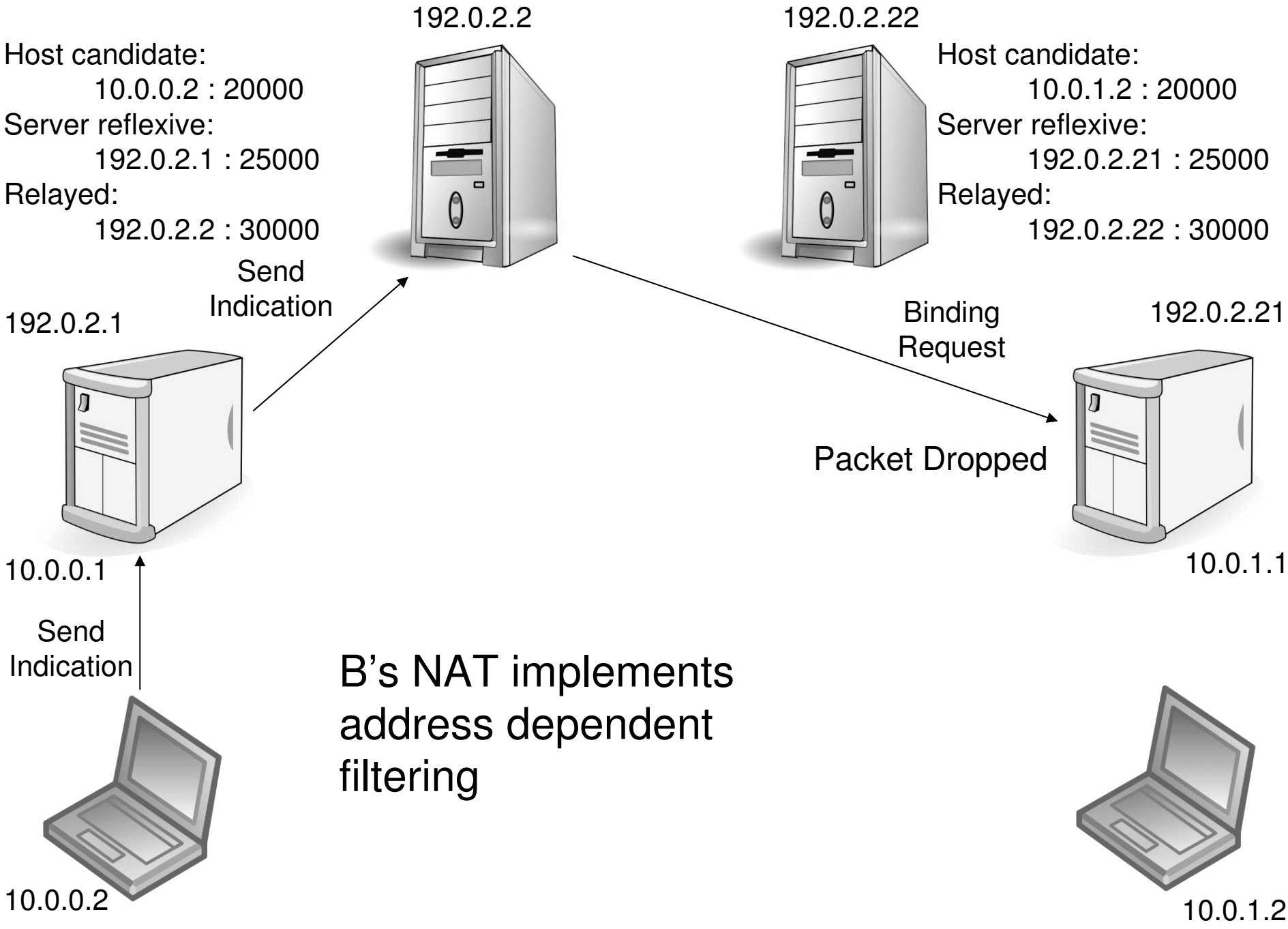
200 OK (answer)

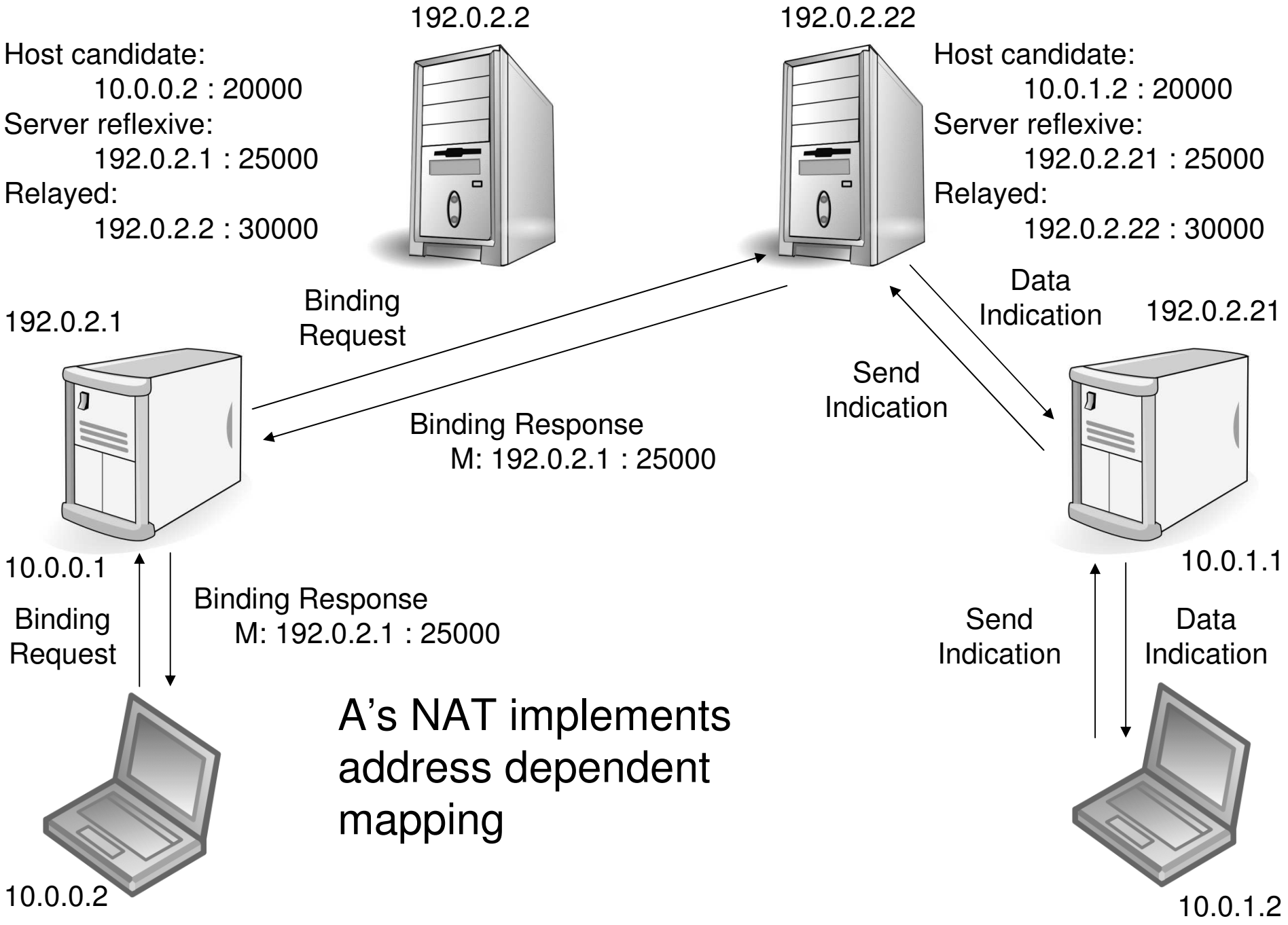
ACK

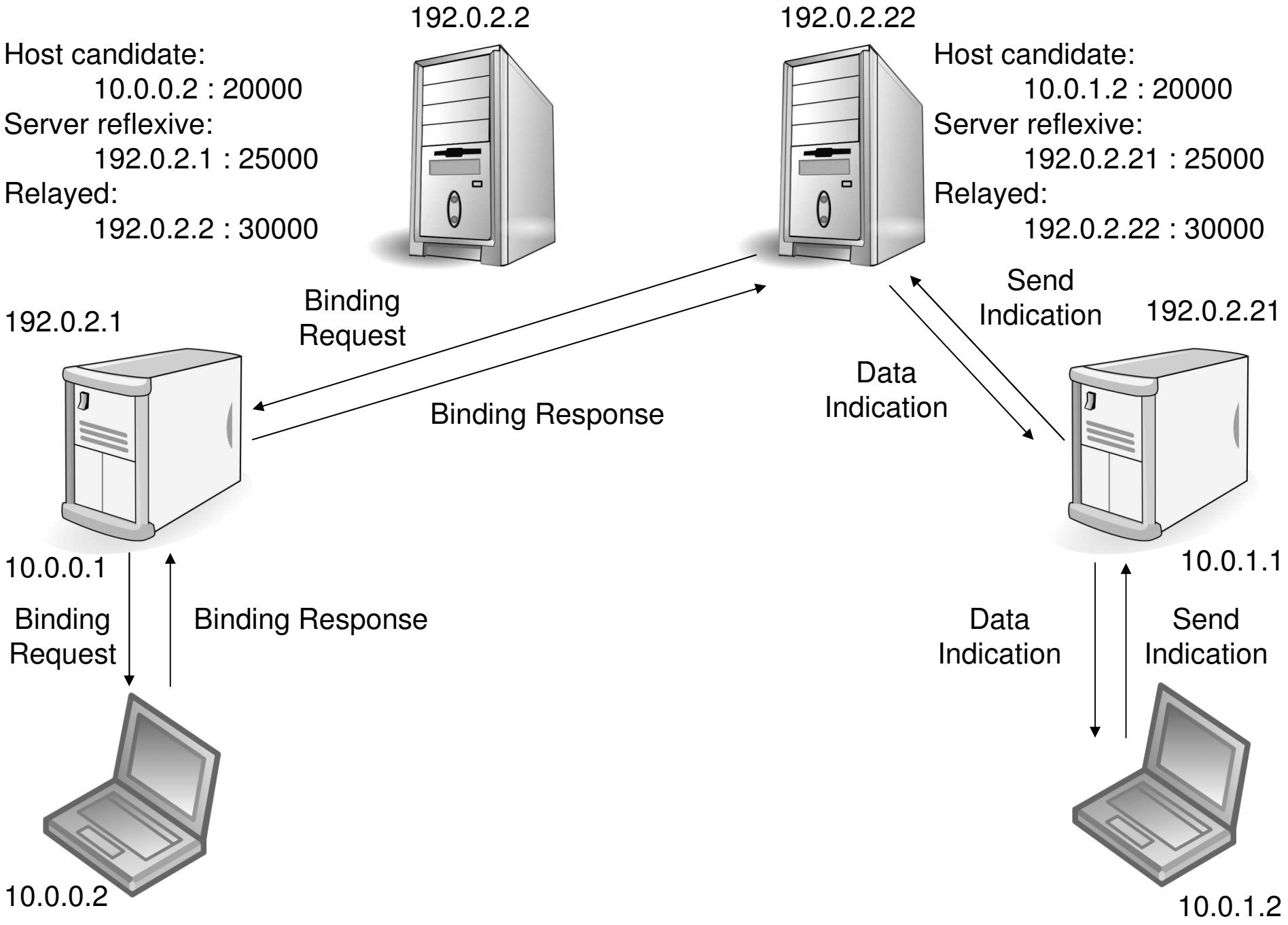


10.0.1.2


192.0.2.2
192.0.2.22
Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000
Host candidate:
10.0.1.2 : 20000
Server reflexive:
192.0.2.21 : 25000
Relayed:
192.0.2.22 : 30000
192.0.2.1
192.0.2.21
Binding
Request
Packet Dropped
10.0.0.1
10.0.1.1
Binding
Request
B's NAT implements
address dependent
filtering
10.0.0.2
10.0.1.2


192.0.2.2
192.0.2.22
Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000
Host candidate:
10.0.1.2 : 20000
Server reflexive:
192.0.2.21 : 25000
Relayed:
192.0.2.22 : 30000
Send
Indication
192.0.2.1
Binding
Request
192.0.2.21
Packet Dropped
10.0.0.1
10.0.1.1
Send
Indication
B's NAT implements
address dependent
filtering
10.0.0.2
10.0.1.2


192.0.2.2
192.0.2.22
Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000
Host candidate:
10.0.1.2 : 20000
Server reflexive:
192.0.2.21 : 25000
Relayed:
192.0.2.22 : 30000
192.0.2.1
Binding
Request
Binding Response
M: 192.0.2.1 : 25000
Data
Indication
192.0.2.21
Send
Indication
10.0.0.1
Binding
Request
Binding Response
M: 192.0.2.1 : 25000
Send
Indication
Data
Indication
10.0.1.1
A's NAT implements
address dependent
mapping
10.0.0.2
10.0.1.2


192.0.2.2
192.0.2.22
Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000
Host candidate:
10.0.1.2 : 20000
Server reflexive:
192.0.2.21 : 25000
Relayed:
192.0.2.22 : 30000
Binding
Request
Binding Response
Send
Indication
Data
Indication
192.0.2.1
192.0.2.21
10.0.0.1
10.0.1.1
Binding
Request
Binding Response
Data
Indication
Send
Indication
10.0.0.2
10.0.1.2

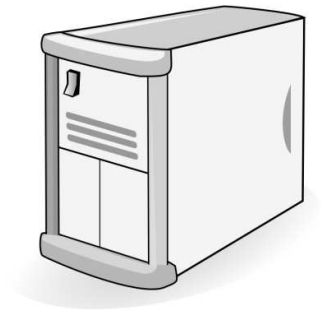
192.0.2.2

Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000

192.0.2.22

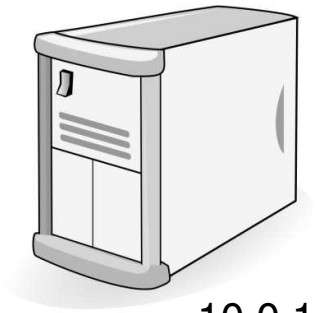
Host candidate:
10.0.1.2 : 20000
Server reflexive:
192.0.2.21 : 25000
Relayed:
192.0.2.22 : 30000

192.0.2.1



10.0.0.1

192.0.2.21



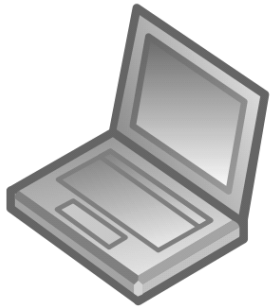
10.0.1.1

10.0.0.2

INVITE (offer)

200 OK (answer)

ACK



10.0.1.2

192.0.2.2

Host candidate:
10.0.0.2 : 20000
Server reflexive:
192.0.2.1 : 25000
Relayed:
192.0.2.2 : 30000

192.0.2.22

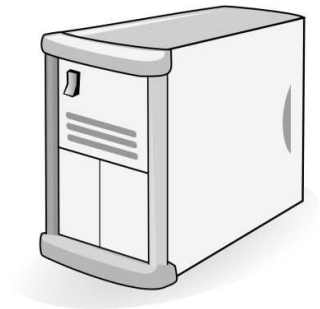
Host candidate:
10.0.1.2 : 20000
Server reflexive:
192.0.2.21 : 25000
Relayed:
192.0.2.22 : 30000

Set Active Destination

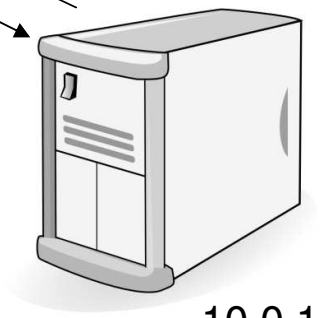
Set Active Destination Response

192.0.2.21

192.0.2.1



10.0.0.1

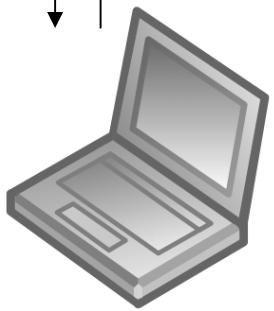


10.0.1.1

Set Active Destination Response

Set Active Destination

10.0.0.2



10.0.1.2


ERICSSON
TAKING YOU FORWARD