UDP NAT Traversal

CSCI-4220 Network Programming Spring 2015

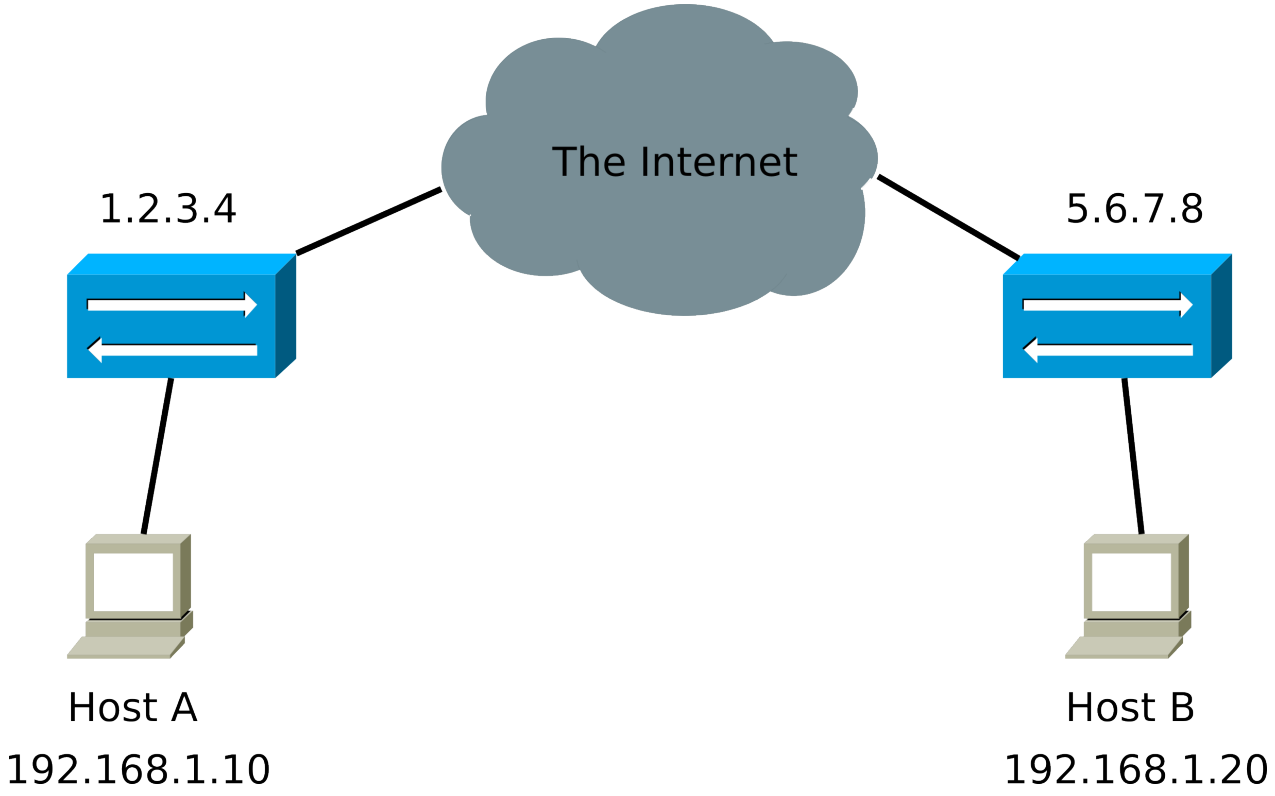
What is NAT Traversal?

NAT traversal means establishing a connection between two hosts when one or both is behind NAT.

Many of today's network applications are peer-to-peer in nature. But NAT takes away some of the peer-to-peer nature of the Internet. NAT use is also extremely widespread in IPv4 networks. How can we solve this problem?

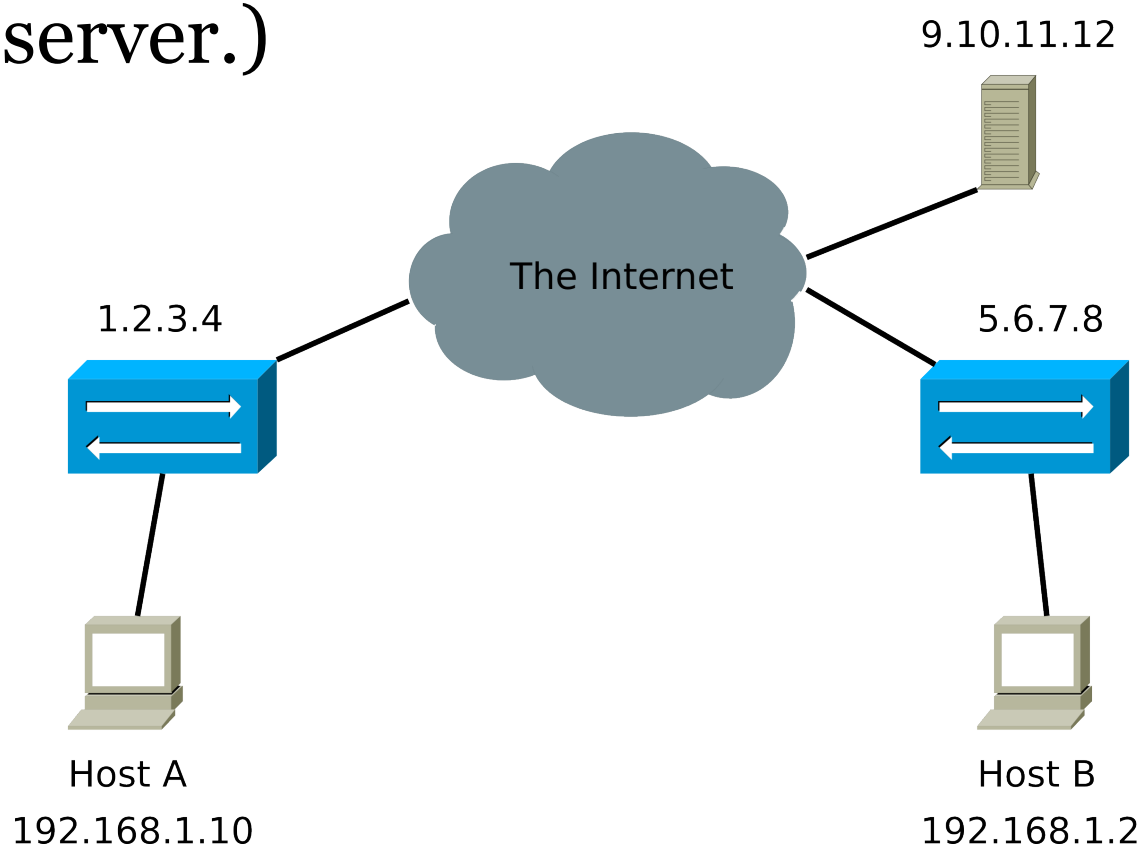
Goal of NAT Traversal

Allow host A and host B to exchange UDP packets, despite the NATs existing between them.



NAT Traversal: Rendezvous

Let's have A and B both talk to another server first - one that isn't behind NAT. (such as a **STUN** server.)



NAT Implementation Types

Defined in RFC 3489:

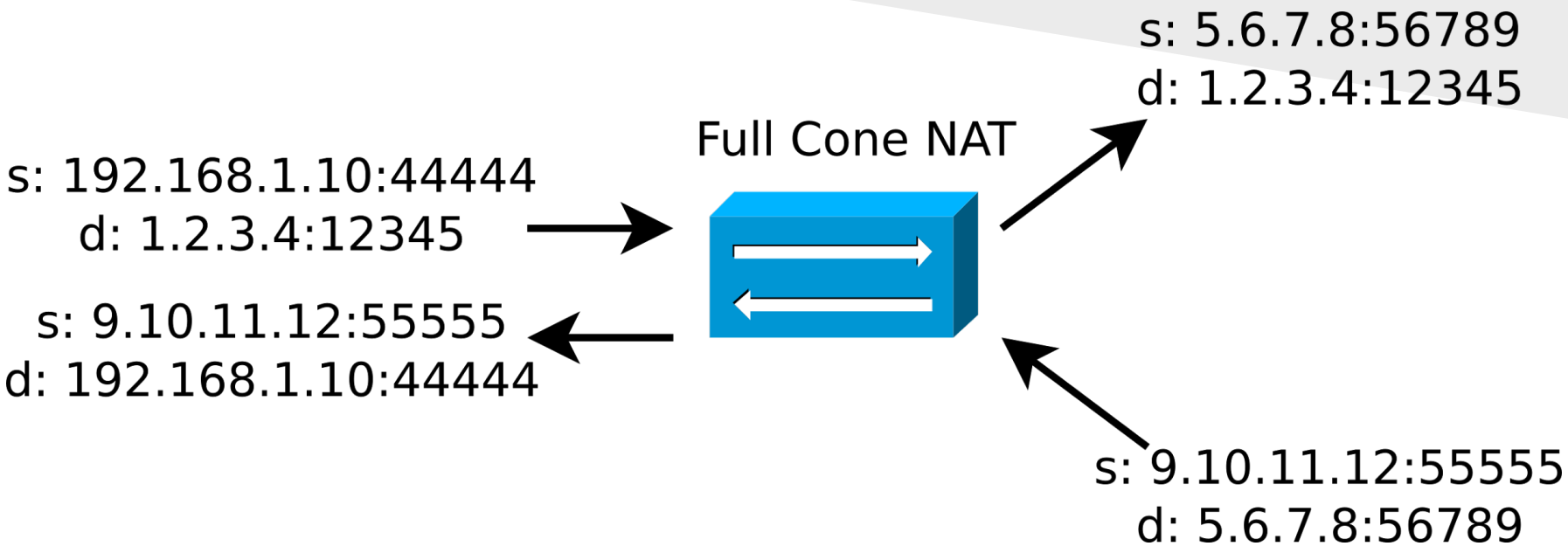
- Full Cone
- Restricted Cone
- Port Restricted Cone
- Symmetric

Difference: the criteria for a packet to match a translation table entry.

Full Cone NAT

- Same “original” source IP+port always results in same “translated” source IP+port
- Router ignores the “destination” columns in the translation table when doing the reverse translation.
- Any packet to the correct “translated” IP+port will get to the “original” IP+port (regardless of source address)
- Easiest type of NAT to traverse.

Full Cone NAT



original src	original dst	translated src	translated dst
192.168.1.10:44444	1.2.3.4:12345	5.6.7.8:56789	1.2.3.4:12345

Traversal Strategy: Full Cone

Peers use the rendezvous server to determine external (translated) IP and port for other peers.

Packets can then be sent directly between peers.

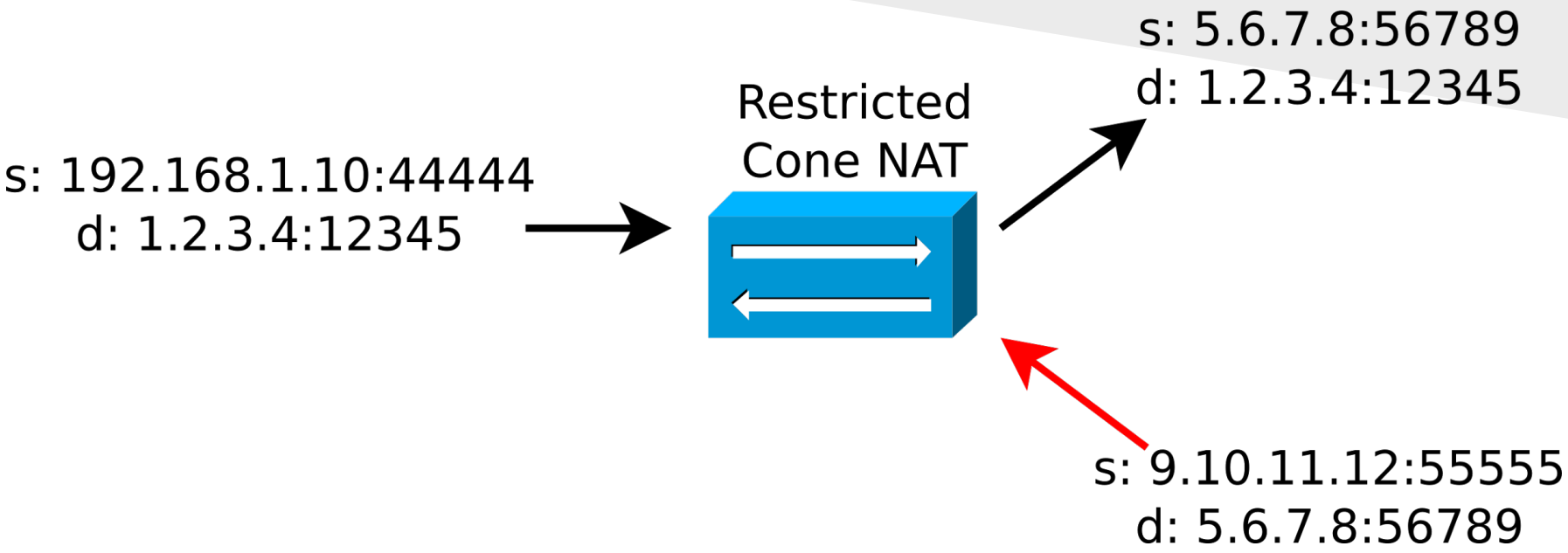
Send a packet periodically (e.g. to the rendezvous server) to keep the translation entry from timing out.

Restricted Cone NAT

Reverse translations are not made unless the “inside” host has previously sent a packet to the “outside” IP address.

That is, the packet’s source address must match against a destination address in the translation table.

Restricted Cone NAT



original src	original dst	translated src	translated dst
192.168.1.10:44444	1.2.3.4:12345	5.6.7.8:56789	1.2.3.4:12345

Traversing Restricted Cone NAT

We need to **manipulate the router's translation table** in order to traverse restricted cone NAT. The rendezvous server can help us here.

Note: the same method can be used to traverse **port-restricted cone** NAT.

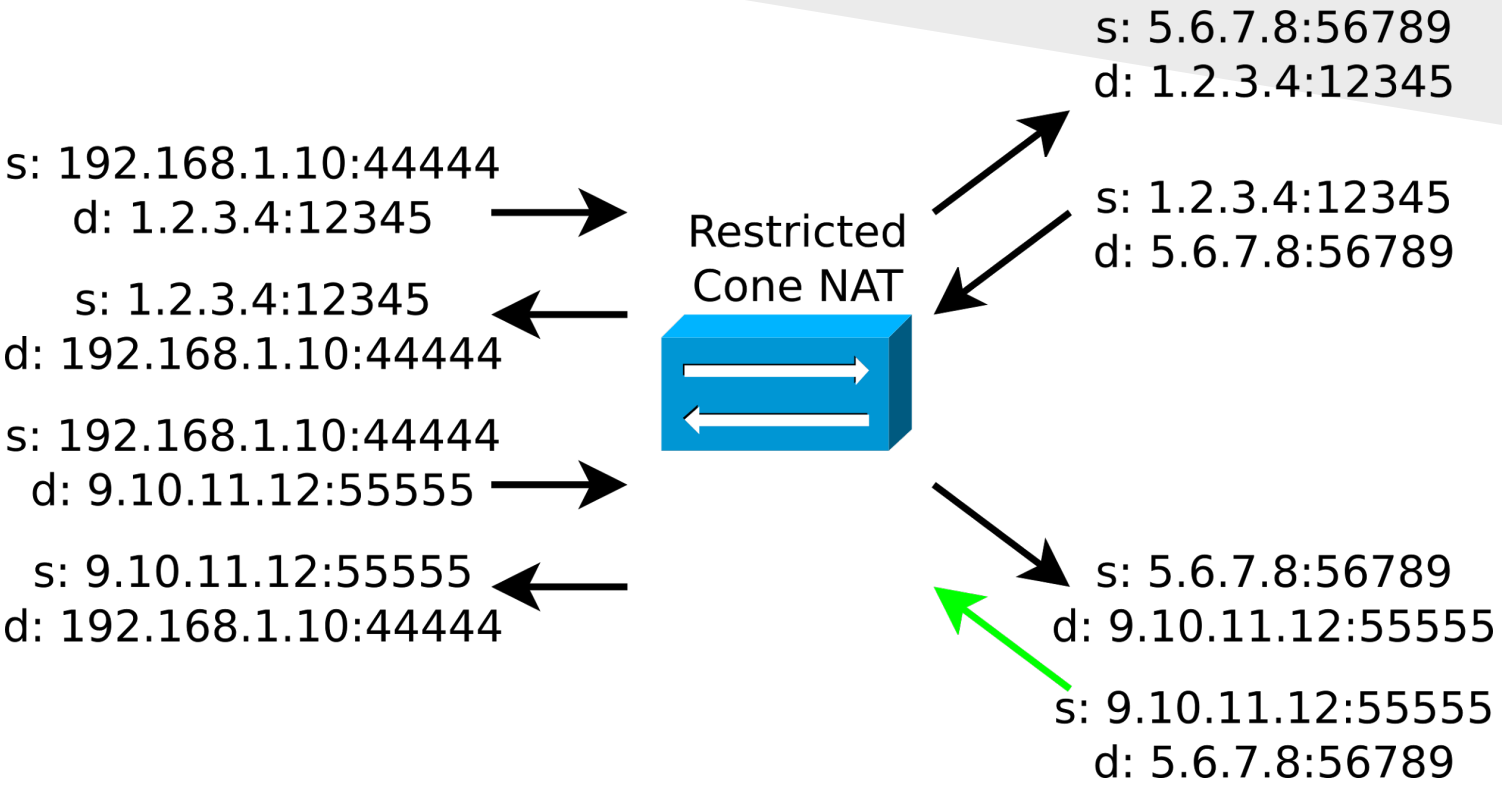
Restricted Cone NAT Traversal

A and B: peers R: rendezvous server

B wants to send a packet to A. Both A and B have sent packets to R.

1. B sends packet to R: “where is A?”
2. R sends packet to B: “A is at 5.6.7.8 (port 56789)
3. R sends packet to A: “B is at 9.10.11.12 (port 55555), and please send a packet there”.
4. A sends packet to B, creating the needed translation rule.

Restricted Cone NAT Traversal



original src	original dst	translated src	translated dst
192.168.1.10:44444	1.2.3.4:12345	5.6.7.8:56789	1.2.3.4:12345
192.168.1.10:44444	9.10.11.12:55555	5.6.7.8:56789	9.10.11.12:55555

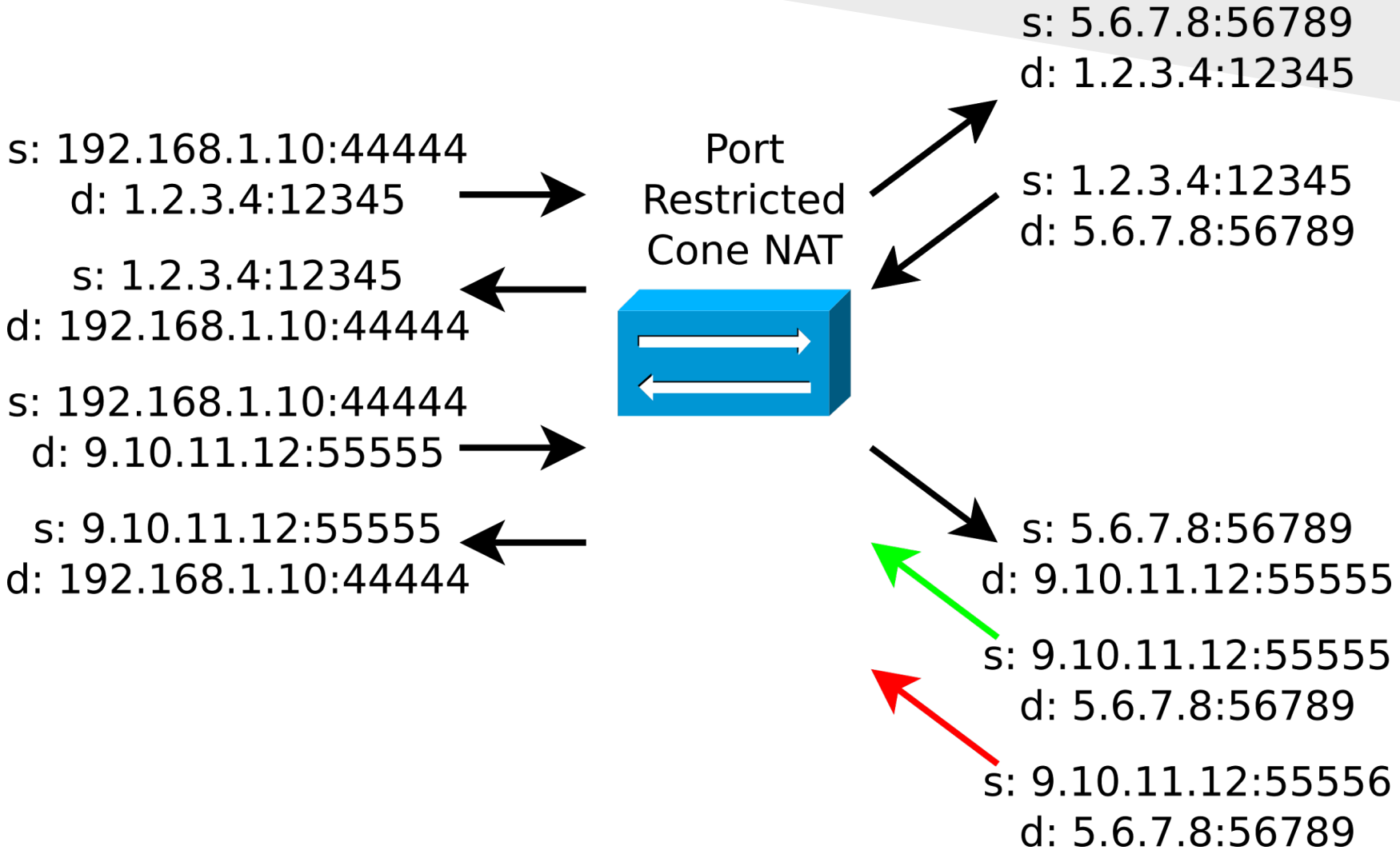
Port Restricted Cone NAT

Like restricted cone NAT, but both source IP address and port must match translated destination for successful reverse translation.

Traversal procedure shown works for both restricted cone and port restricted cone NAT types.

With port restricted cone NAT, we must repeat the process for all source ports we wish to use.

Port Restricted Cone NAT

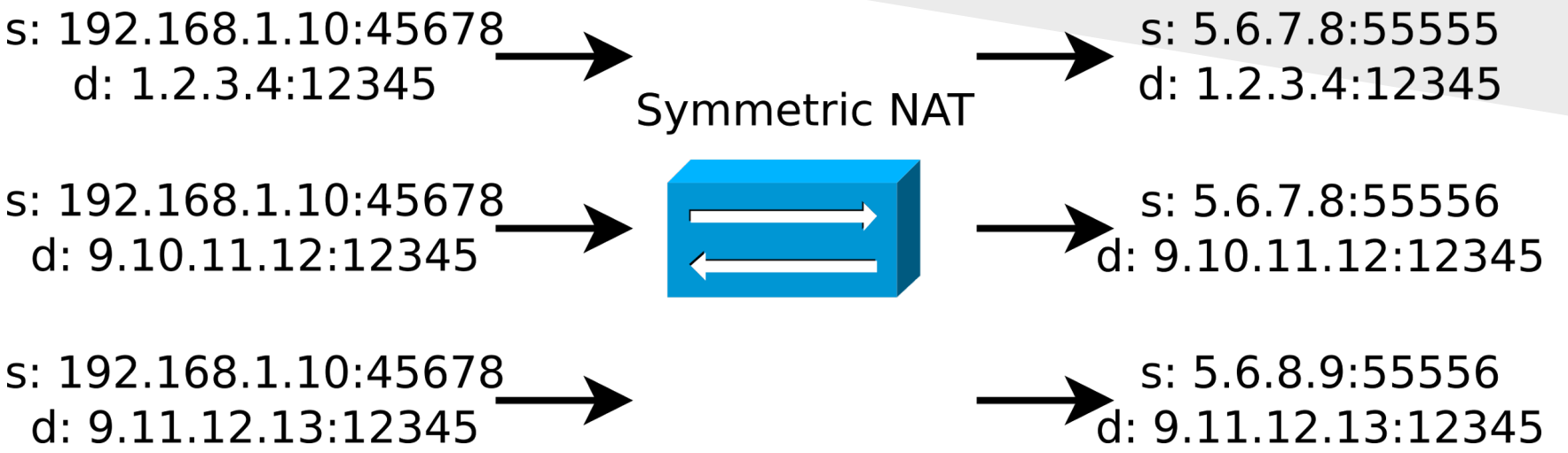


Symmetric NAT

This is the ugly one. A symmetric NAT router may translate the same “original” source IP and source port to different “translated” source IPs and source ports.

Only guarantee is that packets to the same destination will use the same translated source IP and port.

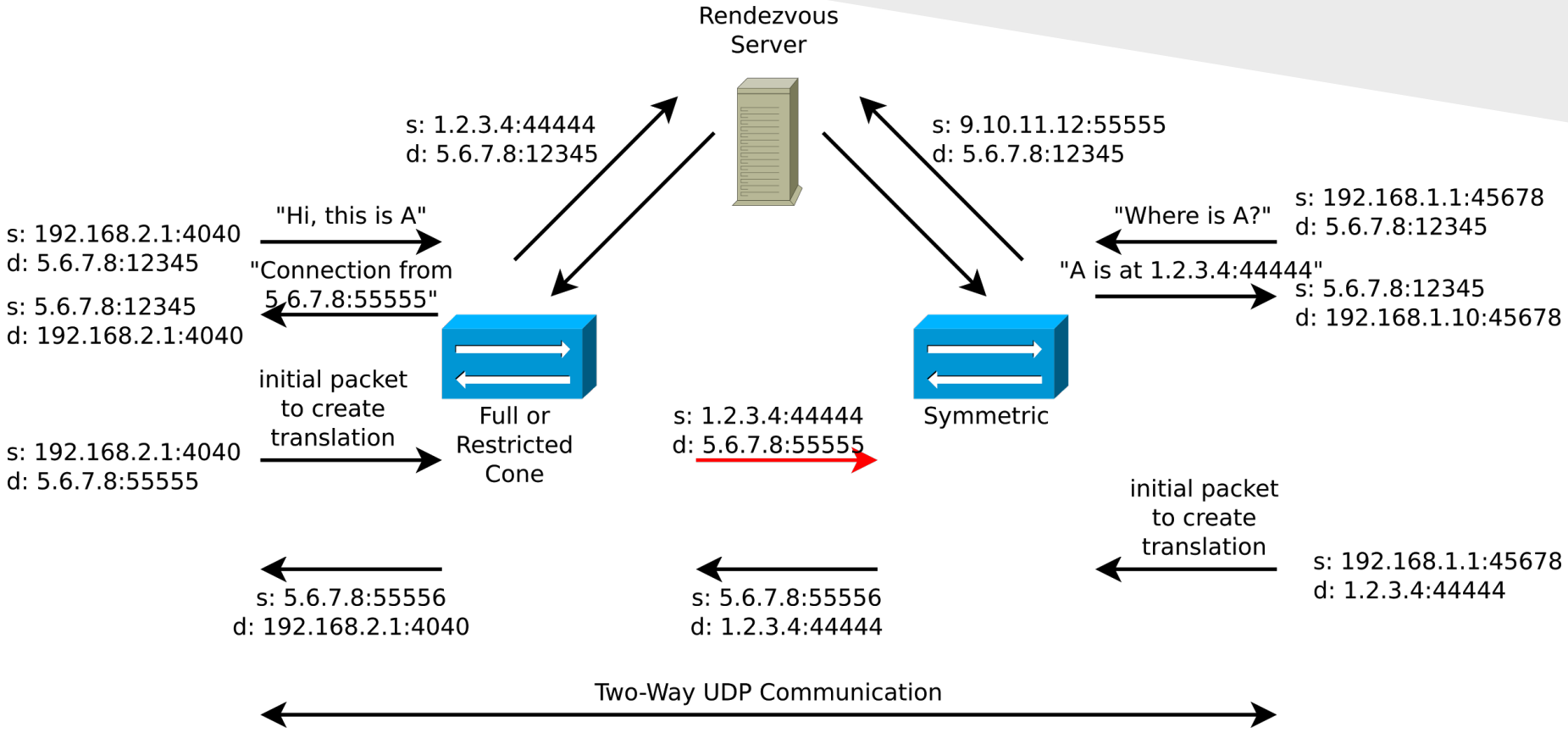
Symmetric NAT



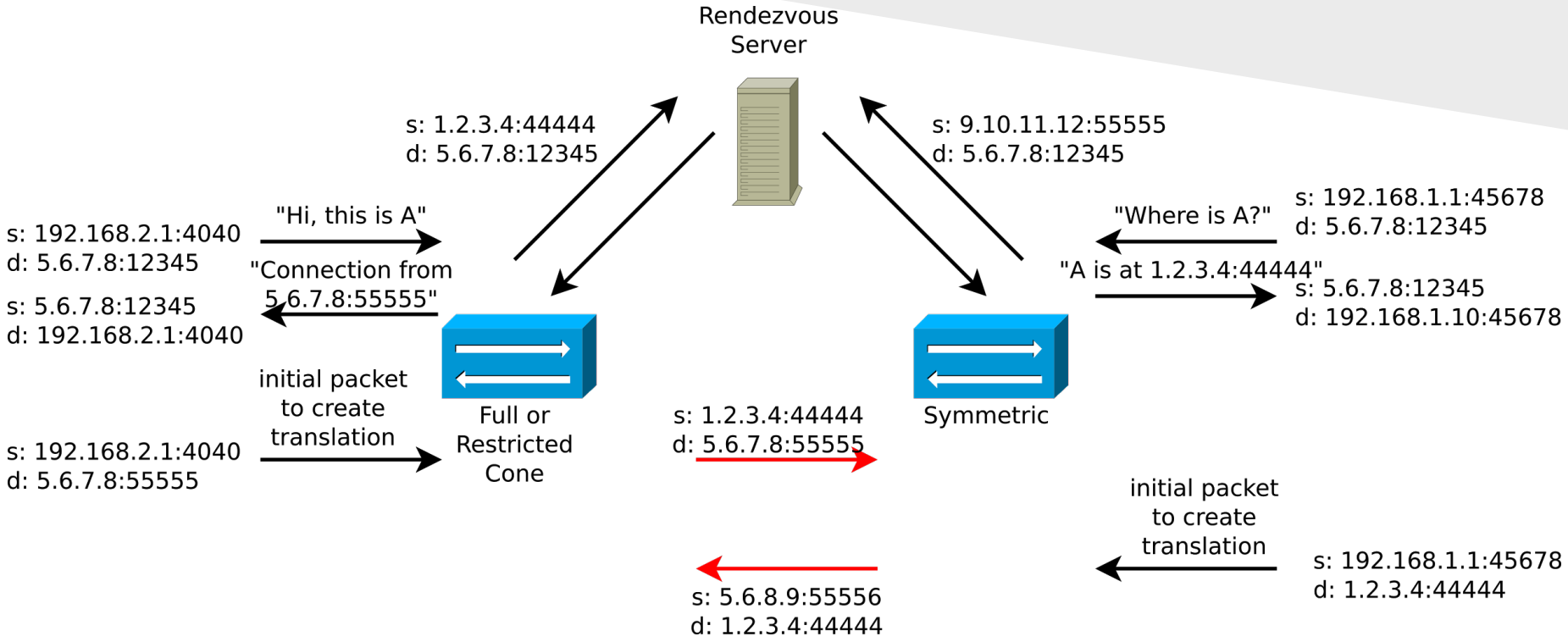
Traversal will be impossible if:

- both peers behind symmetric NAT
- one peer behind symmetric NAT, the other behind restricted or port-restricted NAT

Traversal with Symmetric NAT

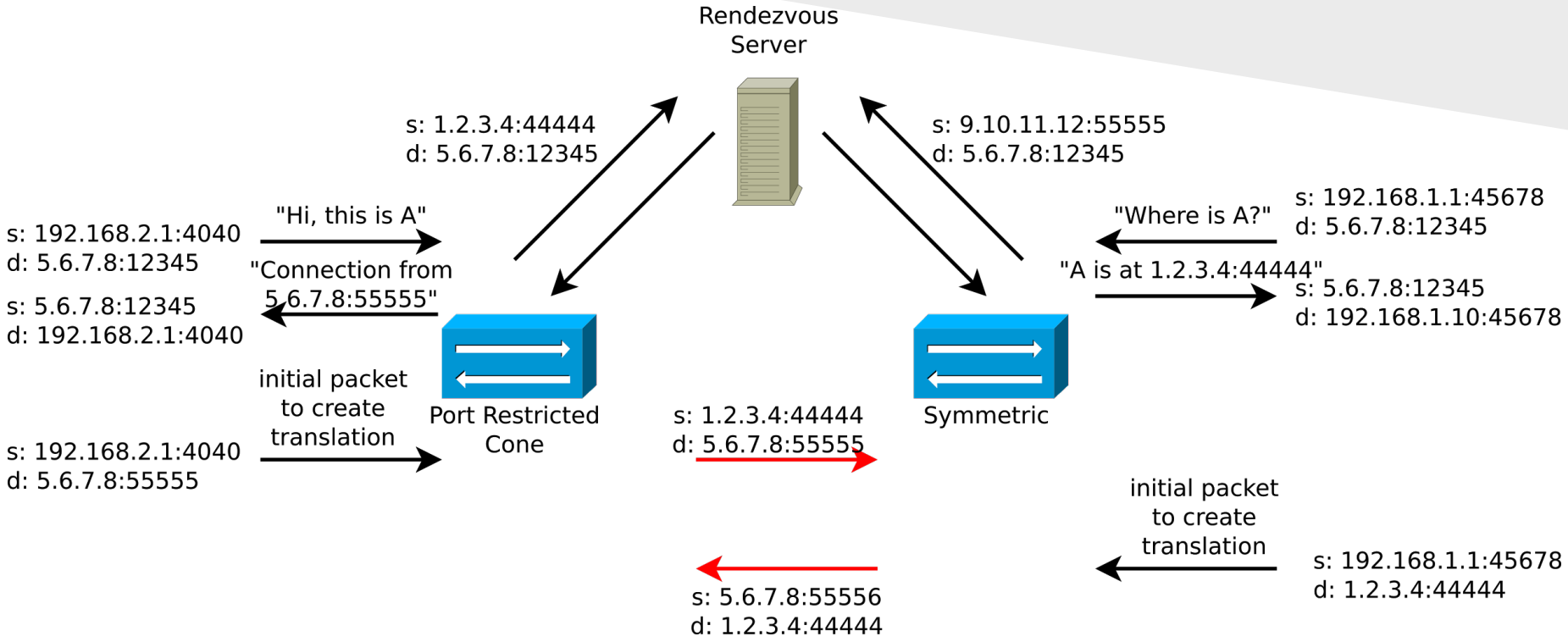


Traversal with Symmetric NAT



If the symmetric NAT uses multiple source addresses, this fails with restricted NAT on left.

Traversal with Symmetric NAT



Port restricted cone requires port numbers to match. We can't predict it, so traversal fails.

Port Forwarding

Last but not least, port forwarding.

Works with TCP and UDP.

Manually configure the translation entries on the NAT router as needed.

Protocols like UPnP allow network applications to do this automatically.

Summary

- NAT traversal is essential to IPv4 peer-to-peer applications (file sharing, online gaming, VoIP, video conferencing)
- Because of the many ways NAT can be implemented, some types of NAT (or combinations thereof) are easier to traverse than others.
- IPv6 deployment may make NAT traversal obsolete.

References

RFC 3489: *STUN - Simple Traversal of UDP through Network Address Translators*. Rosenberg et al.