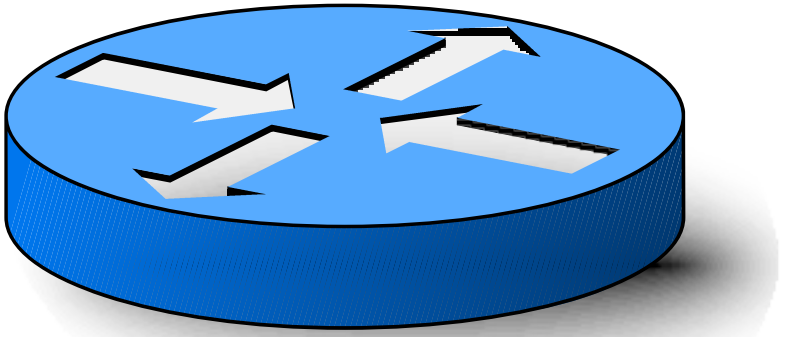
How to play libnice-ly with your NAT

Youness Alaoui

Summary

- What is a NAT?
  - Problems with NATs
  - How to punch holes in a NAT?
  - Types of NATs
- What is ICE?
  - What does it do ?
  - How does it work ?
- What is Libnice?
  - How to use libnice
- Future plans, links, questions?

Collabora



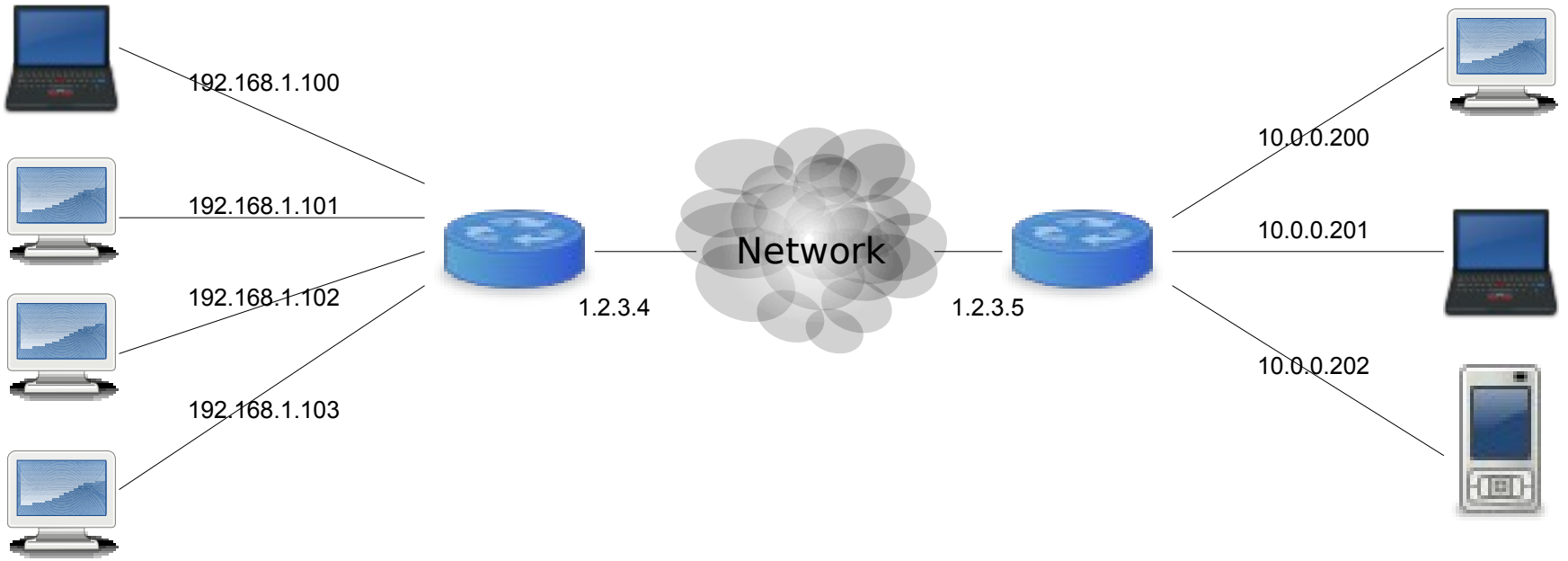
What is a NAT?

Collabora

What is a NAT?

- NAT == Network Address Translation
- Allows multiple computers inside the NAT to have an IP address but share the same external IP
- Fixes the problem of the limit of IPv4 IP addresses available
- A gateway does the translation and everything is routed through it

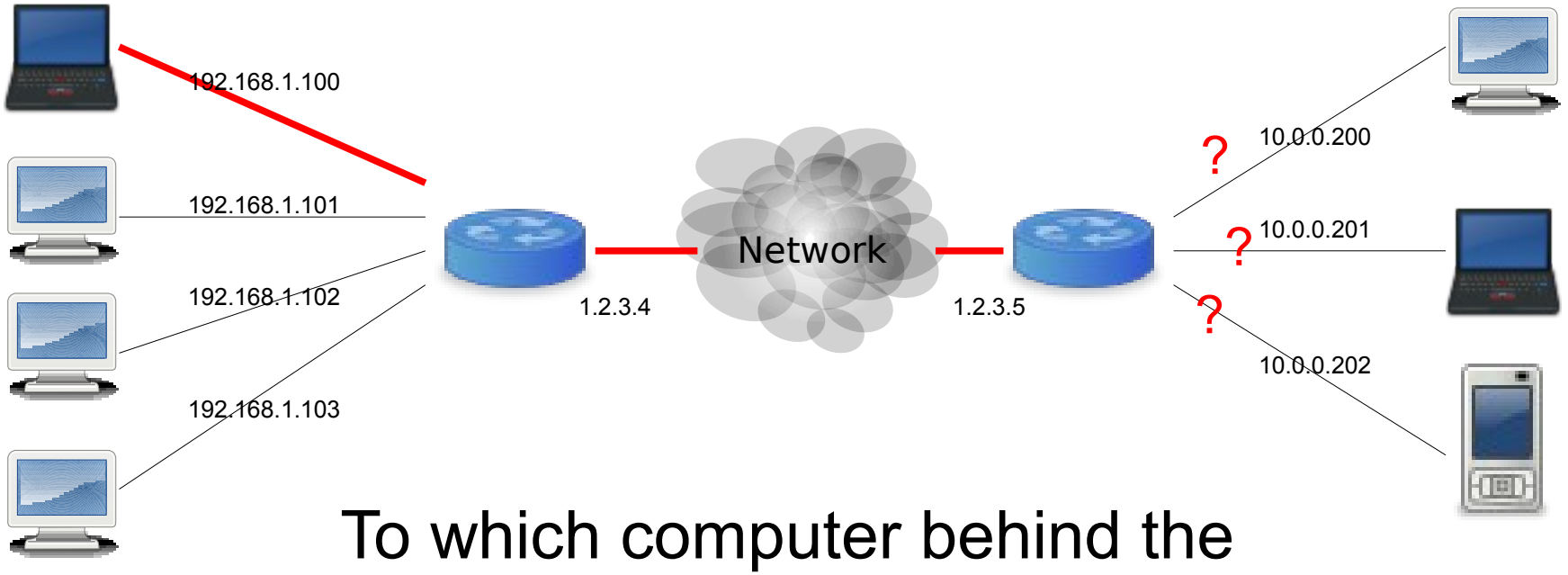
What is a NAT ?



Example of two NATs

Collabora

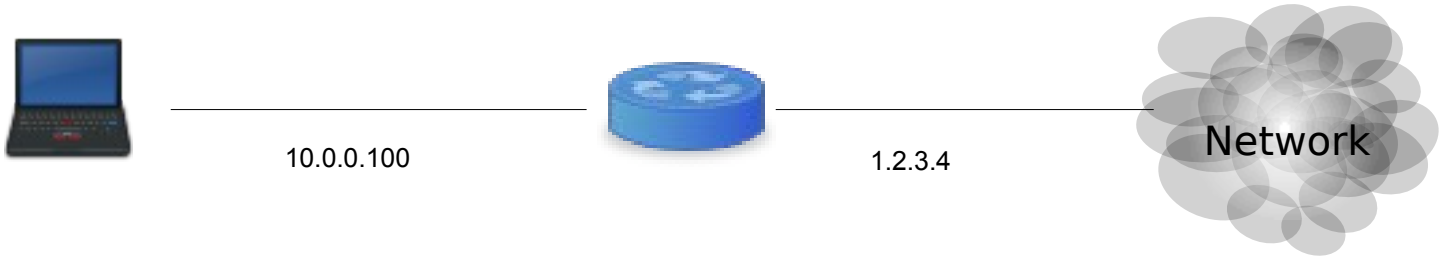
Problems with NATs



To which computer behind the NAT should the router forward data to when it receives data from the network

Collabora

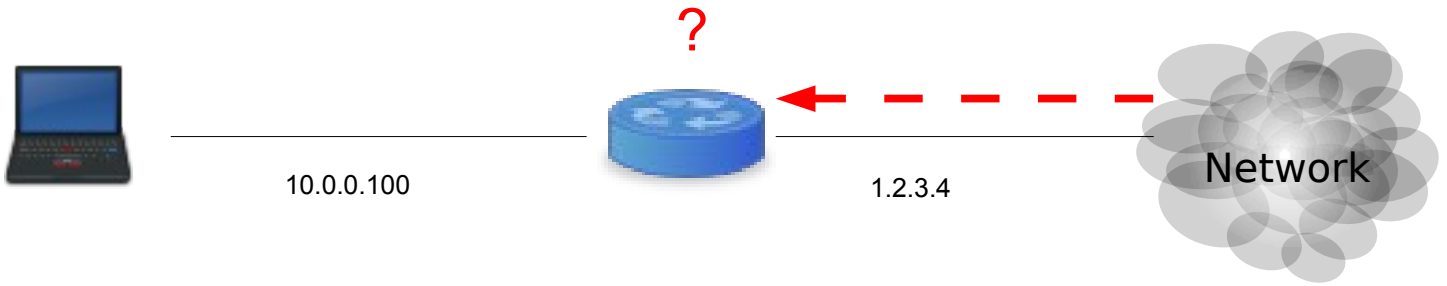
How to punch holes in a NAT?



IP : PORT	IP : PORT

Collabora

How to punch holes in a NAT?

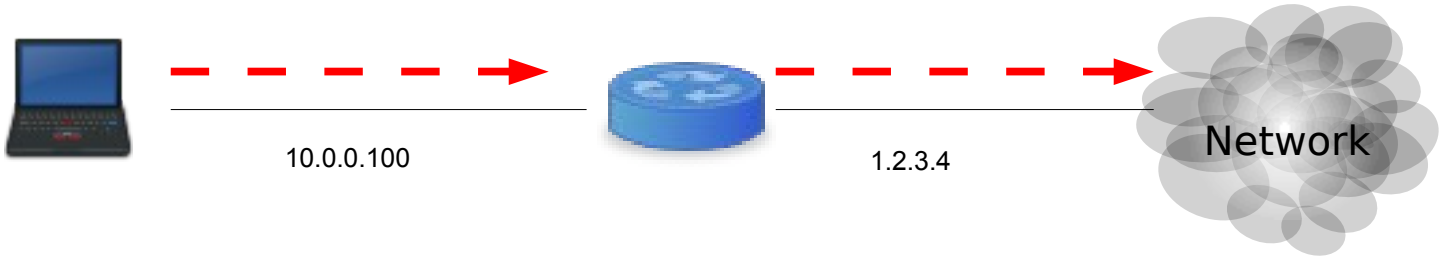


IP : PORT	IP : PORT

Router receives a packet on port 1234 to IP 1.2.3.4.
Message is dropped by the router

Collabora

How to punch holes in a NAT?

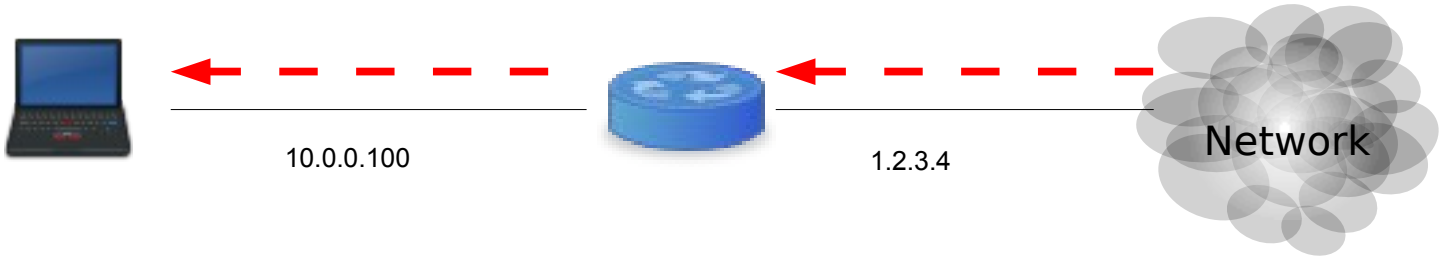


IP : PORT	IP : PORT
10.0.0.100:1234	1.2.3.4:1234

PC 10.0.0.100 sends a packet to the network
Router creates a mapping from internal ip/port to external ip/port

Collabora

How to punch holes in a NAT?



IP : PORT	IP : PORT
10.0.0.100:1234	1.2.3.4:1234

Router receives a packet on port 1234 to IP 1.2.3.4.
Router finds the mapping and forwards the packet

Collabora

Types of NAT

* **Full cone NAT**, also known as one-to-one NAT

IP : PORT	IP : PORT
10.0.0.100:1234	1.2.3.4:1234

* **(Address) Restricted cone NAT**

IP : PORT	IP : PORT / DESTINATION
10.0.0.100:1234	1.2.3.4:1234 / 1.2.3.5
10.0.0.100:1234	1.2.3.4:1234 / 1.2.3.6

Types of NAT

* **Port-Restricted cone NAT**

IP : PORT	IP : PORT / DESTINATION : PORT
10.0.0.100:1234	1.2.3.4:1234 / 1.2.3.5:1234
10.0.0.100:1234	1.2.3.4:1234 / 1.2.3.6:4321

* **Symmetric NAT**

IP : PORT	IP : PORT / DESTINATION : PORT
10.0.0.100:1234	1.2.3.4:11111 / 1.2.3.5:1234
10.0.0.100:1234	1.2.3.4:22222 / 1.2.3.5:4321

Collabora

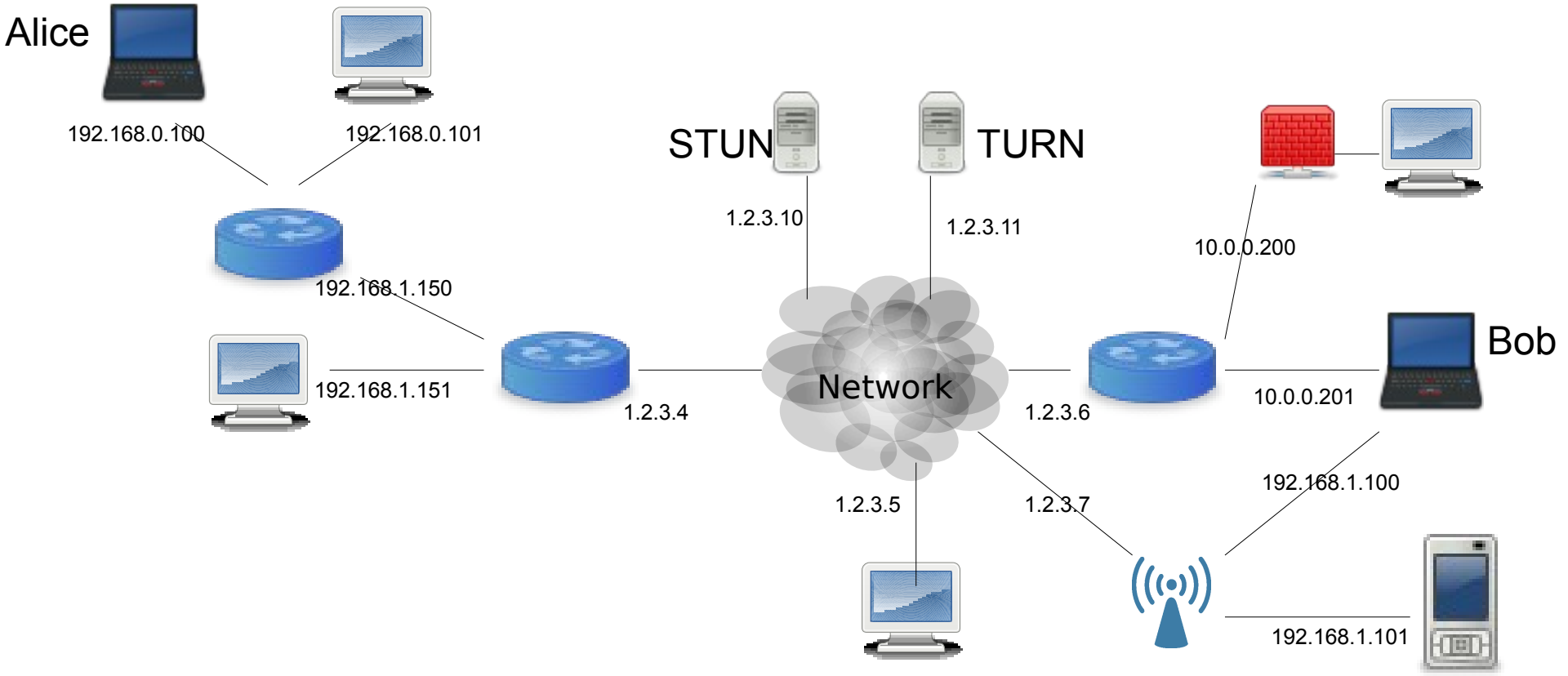
What is ICE?

Collabora

What does it do ?

- The Interactive Connectivity Establishment (ICE) is an RFC Draft (Currently Draft 19)
- It defines a methodology rather than a protocol
- It makes your life easier (when you don't have to implement it)
- It makes sure that two peers are able to connect to each other no matter which network topology they have
- Uses STUN to punch holes in the NAT

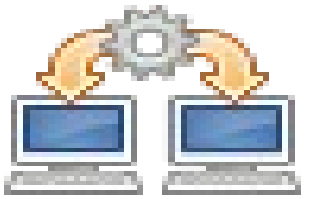
How does it work ?



Step One : Take this type of network topology

Collabora

How does it work ?



Step two: Make it appear to you like this

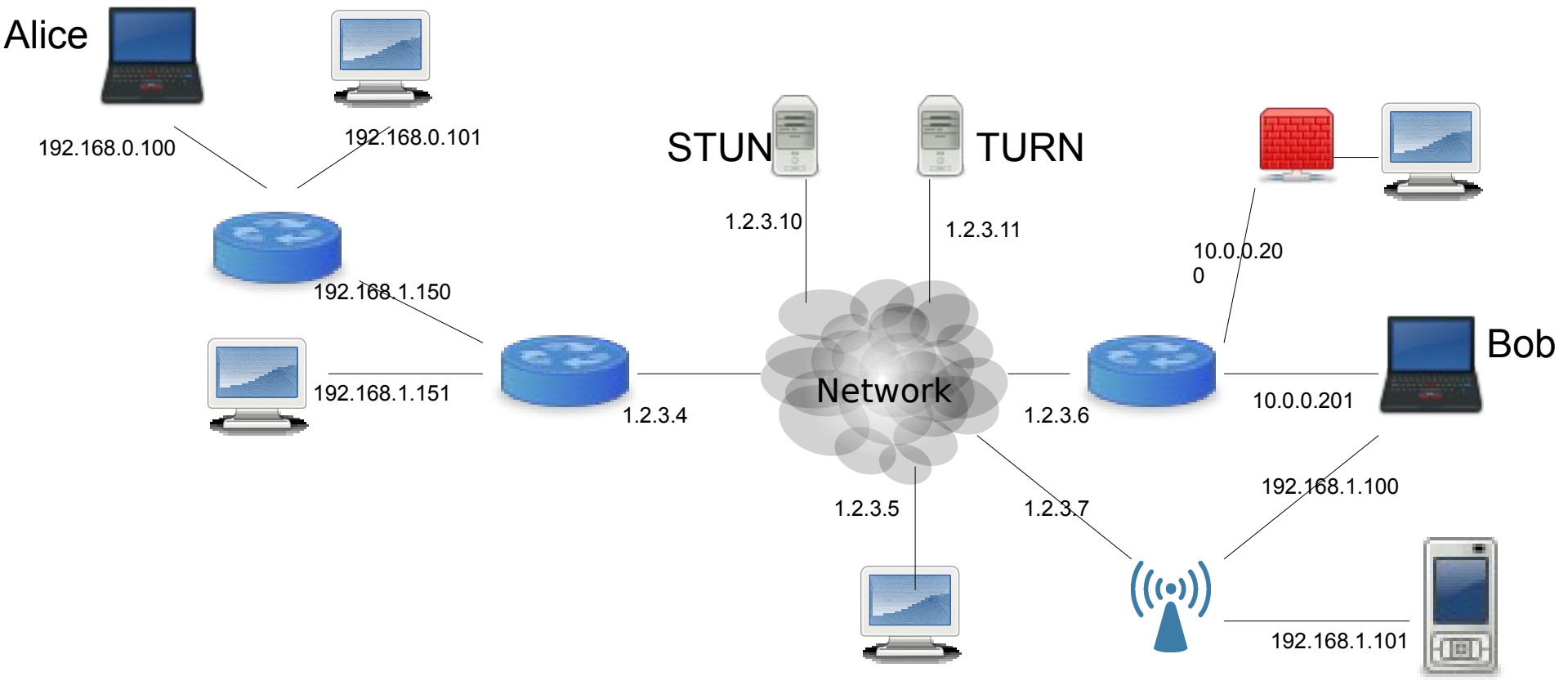
Collabora

How does it work ?

- Gathers multiple 'candidates'
- Uses STUN for UDP hole punching
- Creates multiple candidate pairs between local and remote candidates and validates each pair's connectivity
- Elects the valid candidate pair with the highest priority to be used for peer to peer data transfer
- Uses TURN if direct connectivity is impossible

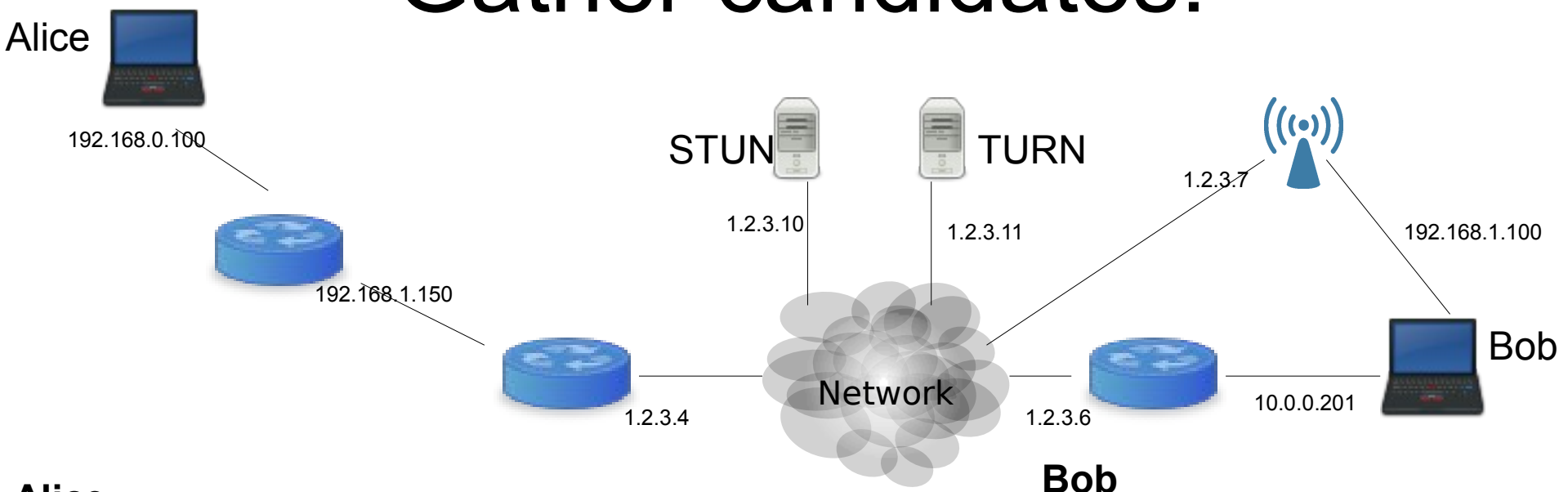
Collabora

How does it work ?



Collabora

How does it work ?
Gather candidates!



Alice

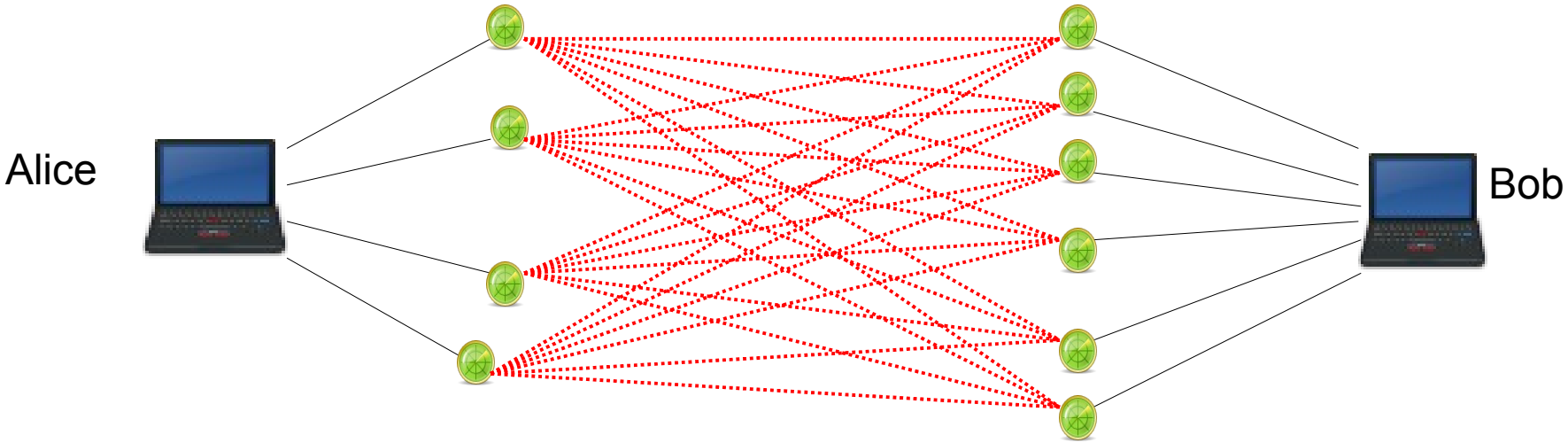
Host candidates:
192.168.0.100 port 1234
Server reflexive candidates:
192.168.1.150 port 12345
1.2.3.4 port 123456
Relay reflexive candidates:
1.2.3.11 port 3478

Bob

Host candidates:
192.168.1.100 port 1234
10.0.0.201 port 12345
Server reflexive candidates:
1.2.3.6 port 12345
1.2.3.7 port 1234
Relay reflexive candidates:
1.2.3.11 port 54321
1.2.3.11 port 54322

Collabora

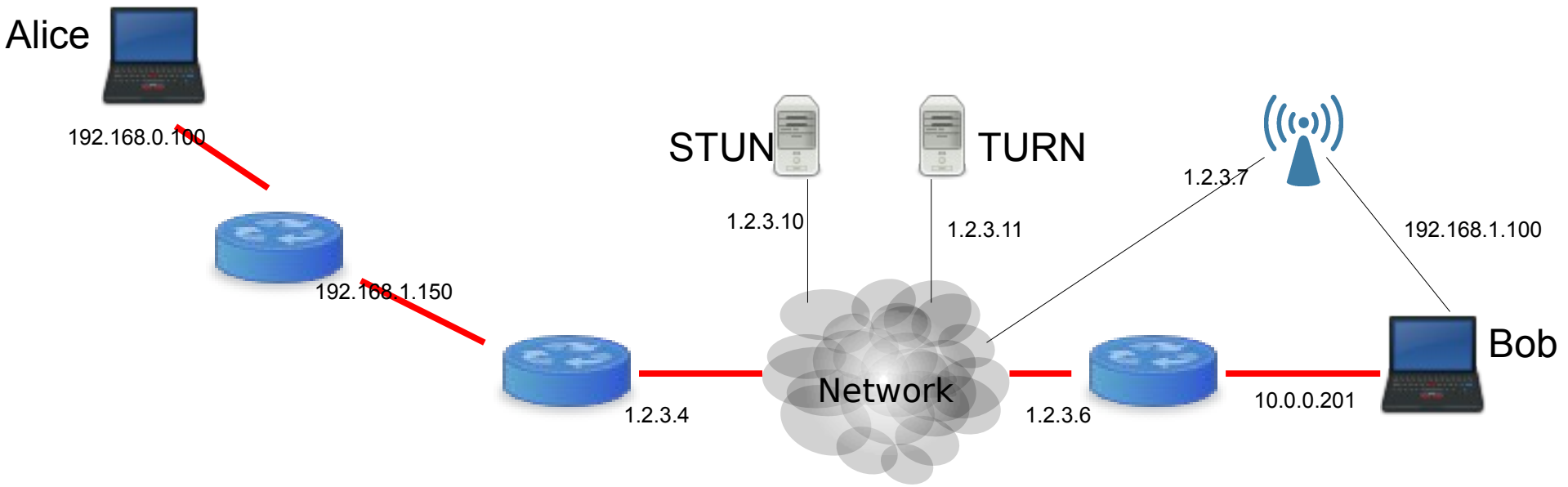
How does it work ?



Create candidate pairs and verify connectivity between each pair (STUN)

Collabora

How does it work ?



Elect a valid pair and start streaming data

Collabora

What is Libnice?

Collabora

What is Libnice?

- A simple to use library implementing the ICE specifications
- Supports Draft 19, Google Talk, MSN 8.x and WLM 2009 ICE specifications
- Supports HTTP, SOCKS5 and Google's Pseudossl proxies
- Supports UDP and TCP TURN relays (TURN Draft 12, Google talk, MSN)
- Supports UPnP
- Comes with a STUN parsing and formatting library that supports STUN RFC 3489 and RFC 5389
- Gstreamer elements are available

Collabora

How to use libnice ?

```
guint stream_id;
gchar buffer[] = "hello world!";

// Create a nice agent
NiceAgent *agent = nice_agent_new (NULL, NICE_COMPATIBILITY_DRAFT19);

// Connect the signals
g_signal_connect (agent, "candidate-gathering-done",
                  cb_candidate_gathering_done, NULL);
g_signal_connect (agent, "new-selected-pair",
                    cb_new_selected_pair, NULL);

// Create a new stream with one component
stream_id = nice_agent_add_stream (agent, 1);

// Attach to the component to receive the data
nice_agent_attach_recv (agent, stream_id, 1, NULL, cb_nice_recv, NULL);
```

Collabora

How to use libnice ?

```
// Start gathering local candidates
nice_agent_gather_candidates (agent, stream_id);

// ... Wait until the signal candidate-gathering-done is fired ...
lcands = nice_agent_get_local_candidates(agent, stream_id, 1);

// ... Send local candidates to the peer and set the peer's remote candidates
nice_agent_set_remote_candidates (agent, stream_id, 1, rcands);

// ... Wait until the signal new-selected-pair is fired ...

// Send our message!
nice_agent_send (agent, stream_id, 1, sizeof(buffer), buffer);

// Anything received will be received through the cb_nice_recv callback

// Destroy the object
g_object_unref(agent);
```

Collabora

Future plans and links

- Add ICE-TCP support
- Add a reliable mode where TCP over UDP would be used
- Upgrade TURN support from Draft 12 to 15
- Add NAT-PMP support
- Integration with Telepathy D-Bus Tubes

- http://nice.freedesktop.org
- http://collabora.co.uk

Collabora

Questions ?

Collabora

Questions ?

Collabora