Webinar: Advanced RIPE Atlas Usage

Vesna Manojlovic
Christopher Amin

RIPE NCC

Amsterdam | August 2015

Goals

- Learn how to:
  - Use RIPE Atlas measurements for network monitoring and troubleshooting
  - Use API calls to create measurements
  - Integrate RIPE Atlas with existing monitoring systems
- Opportunity for hands-on practice
- Get your questions answered by developers

Prerequisites

- We assume you have already used RIPE Atlas
- Do you have a RIPE NCC Access account?
  - **If not - quickly create one: ripe.net/register**
- Do you have credits to spend?
  - **If not - tell us your account in the chat window**

Warm-up question

- Who prefers using the graphical interface to schedule a measurement in RIPE Atlas?

- Who prefers RIPE Atlas API?

Overview

- Introduction to RIPE Atlas
- Creating measurements
- Integration with network monitoring systems
- Real-time performance monitoring
- Take part in the RIPE Atlas community
- Additional slides

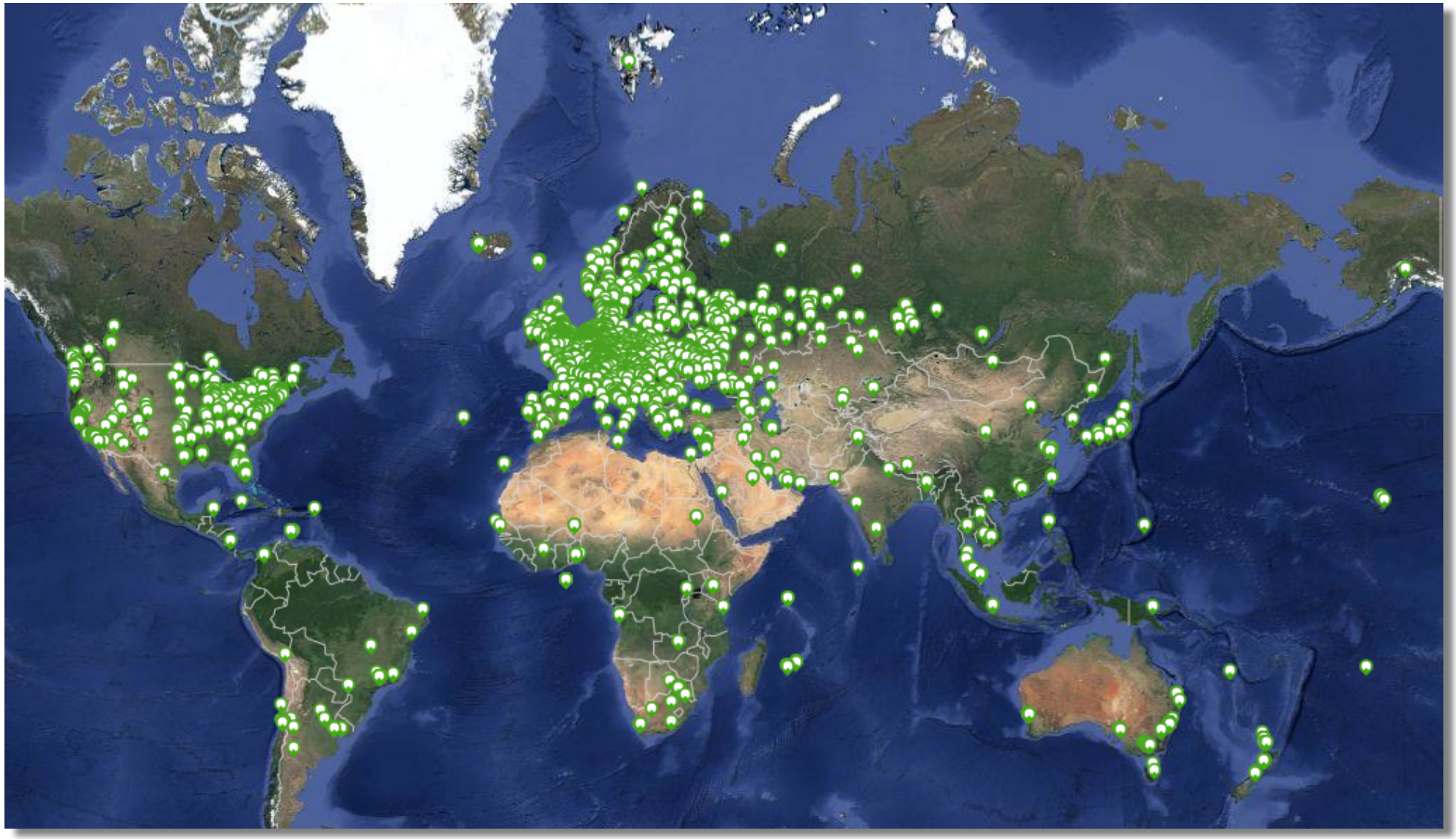
Introduction to RIPE Atlas

RIPE NCC

Definition

- RIPE Atlas is a global active measurements platform
- Goal: view Internet reachability
- Probes hosted by volunteers
- Data publicly available

RIPE Atlas coverage



RIPE Atlas results

- Ongoing global measurements towards root nameservers
  - Visualised as Internet traffic maps
- Ongoing regional measurements towards “anchors”
- Users can run customised measurements
  - ping, traceroute, DNS, SSL/TLS and NTP

Numbers for August 2015

- 8,400+ probes connected

- 5,000+ active users in the last quarter
- 2,500+ results collected per second
- 35,000+ customised measurements weekly

Creating a Measurement

RIPE NCC

Benefits of your own measurements

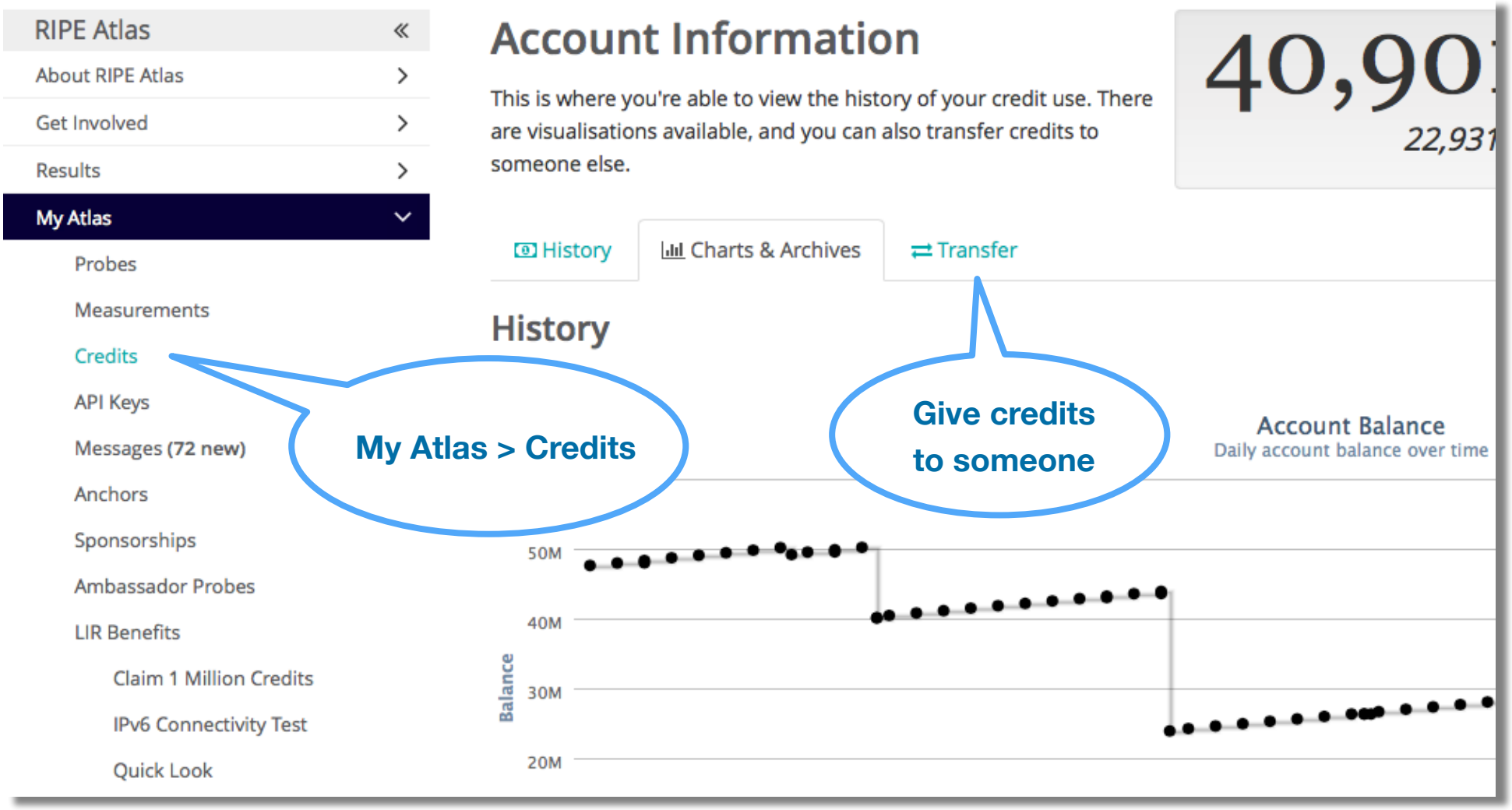
- A customer reports a problem: they cannot reach one of your servers
  - You can schedule pings or traceroutes from up to 500 RIPE Atlas probes from a particular region to check where the problem might be
- Measuring packet loss on a suspected “bad” link
- Testing anycast deployment

Credit system

- Running your own measurements cost credits
  - ping = 10 credits, traceroute = 20, etc.
- Why? Fairness and to avoid overload
- Daily spending limit & max measurements user can create

- Hosting a RIPE Atlas probe earns credits
- Earn extra credits by:
  - Being a RIPE NCC member
  - Hosting an anchor
  - Sponsoring probes

Credits overview



Scheduling a measurement

- Log in to atlas.ripe.net
- “My Atlas” > “Measurements”

- Three methods:
  1. Quick & Easy
     - Type
     - Target
     - Done! (default values are used…)
  2. Advanced GIU usage
  3. CLI scripting using API

2: Using GUI to schedule a measurement

- Mostly a periodic, long time measurement
  - If just once, ASAP, choose “One-off”
- Choose type, target, frequency, # of probes, region…
  - Improved interactive interface helps you at each step
- You will spend credits

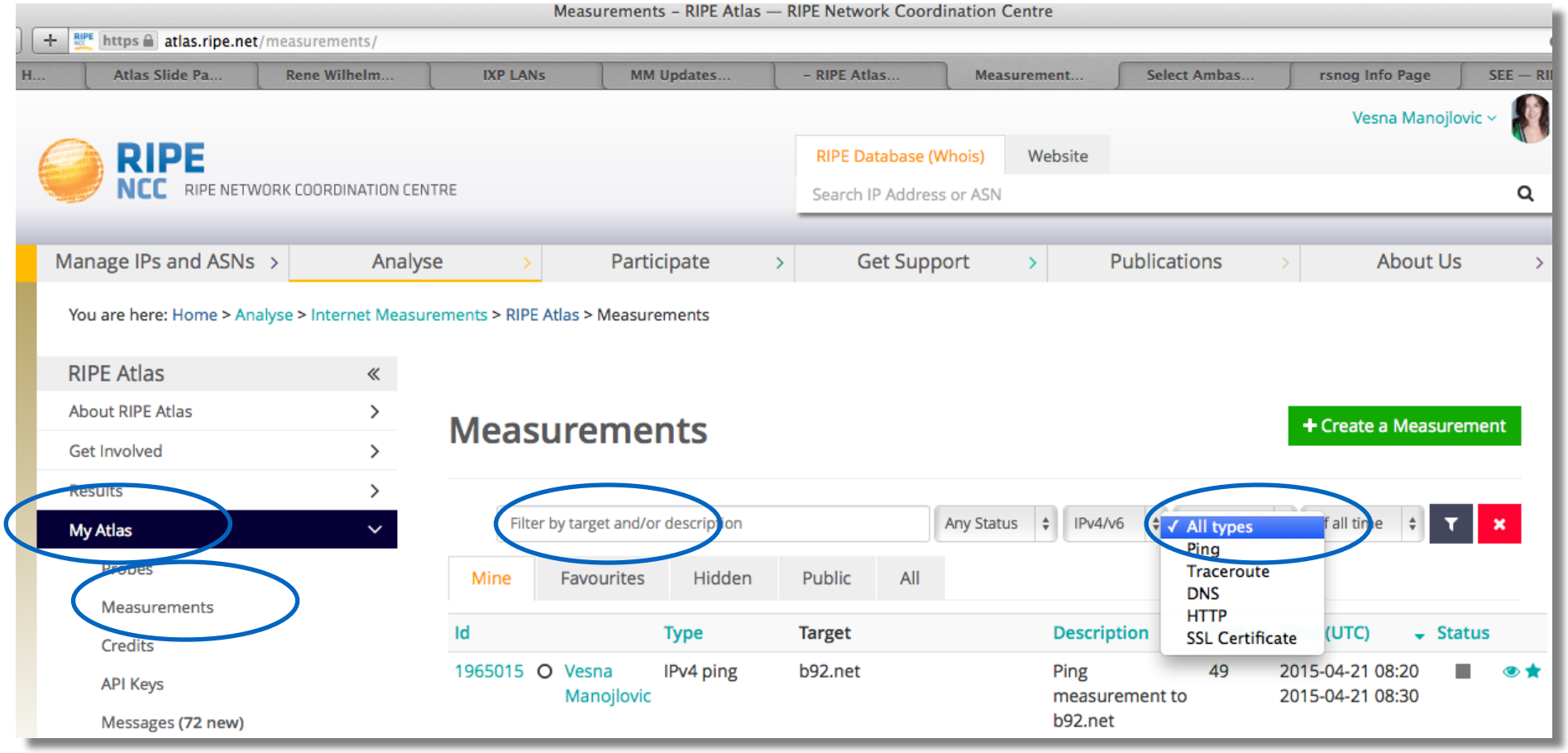
3: Using API to schedule a measurement

- Using command-line & scripting:
  Application **P**rogramming **I**nterface (API)
  - https://atlas.ripe.net/docs/measurement-creation-api/
  - https://atlas.ripe.net/keys/

- You will need API keys
  - To create measurements without logging in
  - To securely share your measurement data

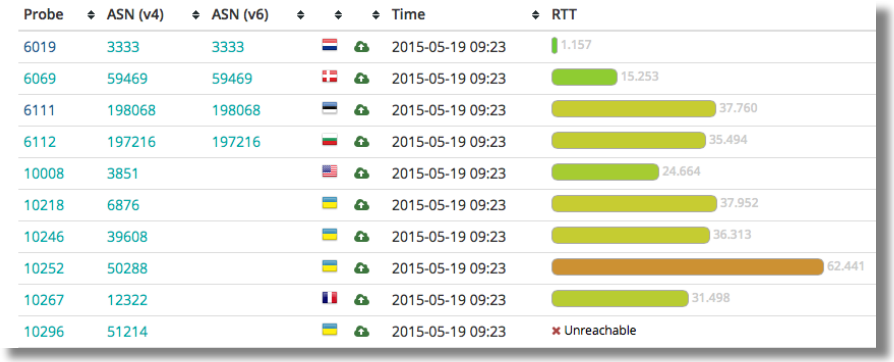
Looking up measurements results

- Go to “My Atlas” > “Measurements”



Available visualisations: ping

- List of probes: sortable by RTT

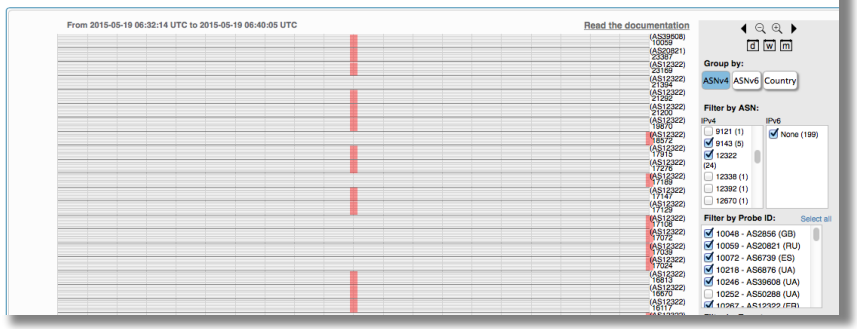


Probe	ASN (v4)	ASN (v6)	Time	RTT
6019	3333	3333	2015-05-19 09:23	1.157
6069	59469	59469	2015-05-19 09:23	15.253
6111	198068	198068	2015-05-19 09:23	37.760
6112	197216	197216	2015-05-19 09:23	35.494
10008	3851		2015-05-19 09:23	24.664
10218	6876		2015-05-19 09:23	37.952
10246	39608		2015-05-19 09:23	36.313
10252	50288		2015-05-19 09:23	62.441
10267	12322		2015-05-19 09:23	31.498
10296	51214		2015-05-19 09:23	✖ Unreachable

- Map: colour-coded by RTT

- Seismograph: stacked multiple pings with packet loss

An interactive visualisation for ping measurements.



Exercise: Create a Measurement

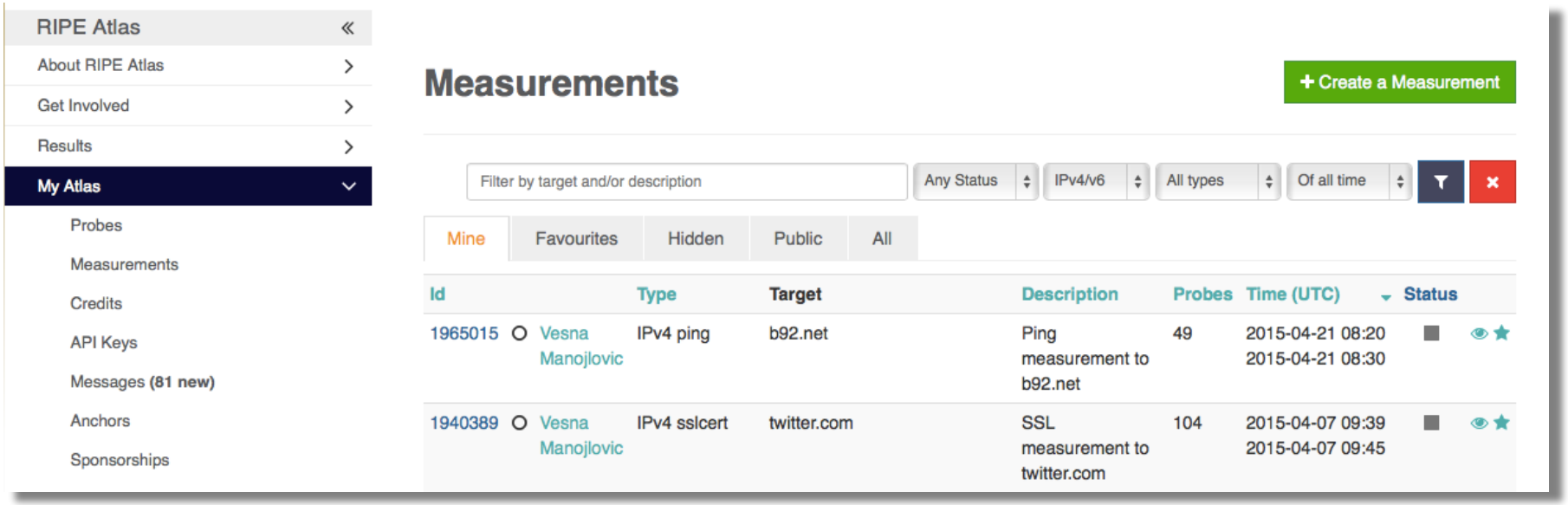
RIPE NCC

Tasks

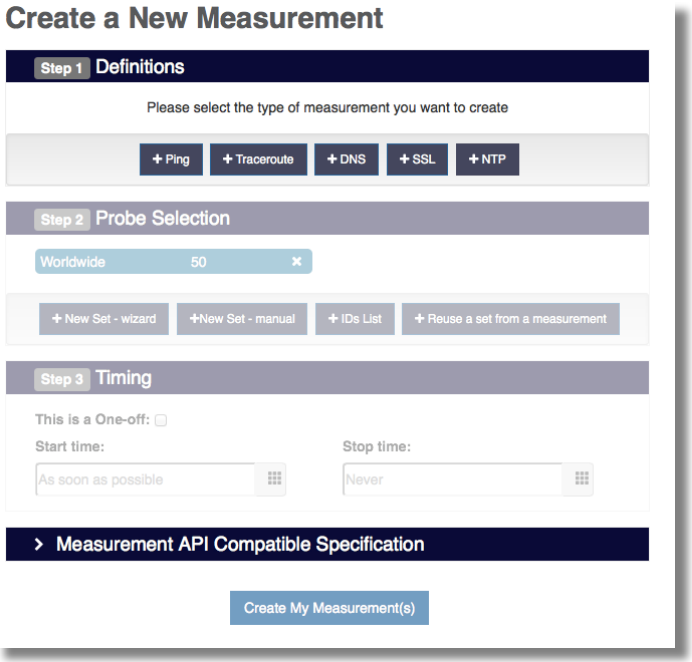
- Create a **ping** measurement:
  - Involving ten probes
  - To a target of your choice
  - Source is your country
  - Duration of two days

1. Warm-up: Create a measurement using the GUI
2. Create API Key
3. Schedule a measurement using the API

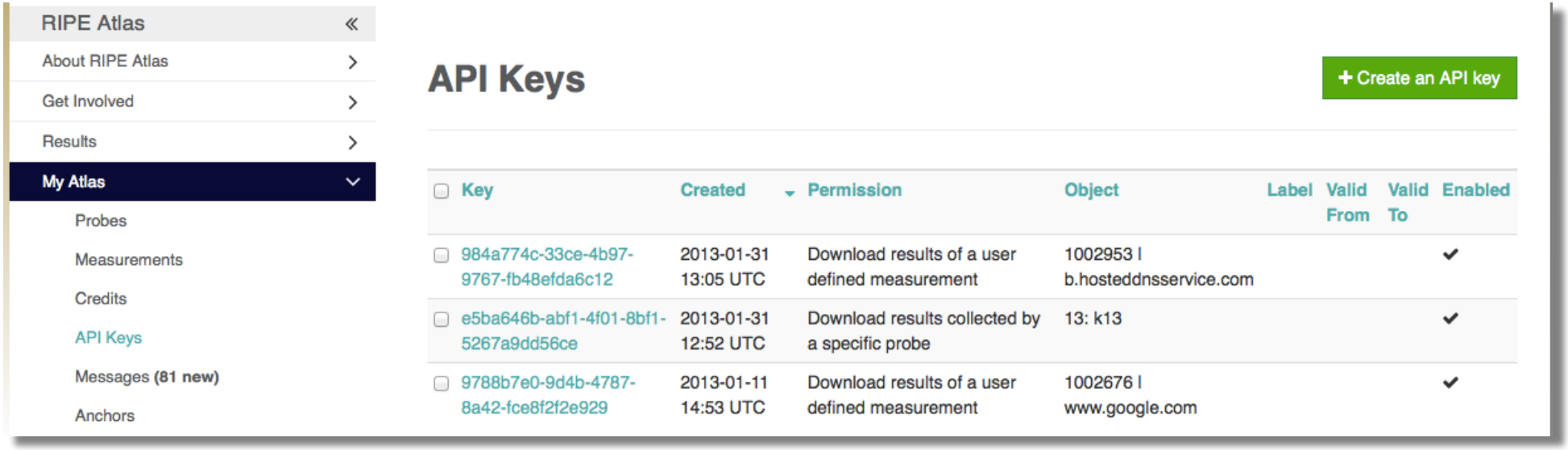
Sub-task 1: Use web interface



- Useful hint: once you generate a measurement, copy “API Compatible Specification” to text file
- Note MSM-ID, too



Sub-task 2: Create API key



- Click on “Create an API Key”
- Choose type: “create a new user-defined measurement”
- “Object” is not applicable (N/A) for this type
- Give it a label
- Give it a duration of validity (leave empty for defaults)
- “Key” value to be passed on to the API call (next step)

Sub-task 3: Use API

- Schedule a measurement using API
  - Use the “key” you just generated
  - Hint: copy and past API call syntax from the measurement generated by the GUI

- Example:

```
$ curl -H "Content-Type: application/json" -H "Accept: application/json" -X
POST -d '{ "definitions": [ { "target": "ripe.net", "description": "My First
Measurement", "type": "ping", "af": 4 } ], "probes": [ { "requested": 10,
"type": "country", "value": "RS" } ] }' https://atlas.ripe.net/api/v1/
measurement/?key=YOUR_API_KEY
```

Integration of RIPE Atlas with Network Monitoring Systems

RIPE NCC

Network monitoring

- Network operators use tools for monitoring network health (e.g. Nagios and Icinga)

- These tools can receive input from RIPE Atlas via the API

- Benefits:
  - pings from 500 out of 8,000+ probes around the world
  - See your network from the outside
  - Plug into your existing practices

Steps for integration

1. Create a RIPE Atlas ping measurement
2. Go to “Status Checks” URL
3. Add your alerts in Nagios or Icinga

Creating status checks

- Status checks work via RIPE Atlas' RESTful API
  - https://atlas.ripe.net/api/v1/status-checks/MEASUREMENT_ID/
- You define the alert parameters, for example:
  - Threshold for percentage of probes that successfully received a reply
  - How many of the most recent measurements to base it on
  - The maximum packet loss acceptable
- Documentation:
  - https://atlas.ripe.net/docs/status-checks/

Icinga examples

- Community of operators contributed configuration code!
  - Making use of the built-in “check_http” plugin
- GitHub examples:
  - https://github.com/RIPE-Atlas-Community/ripe-atlas-community-contrib/blob/master/scripts_for_nagios_icinga_alerts
- Post on Icinga blog:
  - https://www.icinga.org/2014/03/05/monitoring-ripe-atlas-status-with-icinga-2/

Exercise: Setting up “Status Checks”

RIPE NCC

Tasks

- Set up and configure a “status check”
  - For an existing ping measurement https://atlas.ripe.net/measurements/2340408/
  - Hint: https://atlas.ripe.net/api/v1/status-checks/2340408/
- Configure the status check in such a way that you will trigger an alert for this measurement
- Optional: set-up status check for your own ping measurement!

Solution

- One possible solution:
  - Set the median RTT to a lower level:
  - https://atlas.ripe.net/api/v1/status-checks/1004005/?median_rtt_threshold=10

- Example of the alerts


```
{"total_alerts":32,"global_alert":true,
"probes":{
"18433":{"all":[null,null,null],"last":null,"last_packet_loss":100.0,"alert":true,"source":"Area: WW","alert_reasons":["loss"]},
"15041":{"source":"Area: WW","last_packet_loss":0.0,"last":19.928,"alert":false},
"18696":{"all":[null,null,null],"last":null,"last_packet_loss":100.0,"alert":true,"source":"Area: WW","alert_reasons":["loss"]},
"16265":{"source":"Area: WW","last_packet_loss":0.0,"last":22.72,"alert":false},
"20236":{"all":[null,null,null],"last":null,"last_packet_loss":100.0,"alert":true,"source":"Area: WW","alert_reasons":["loss"]},
"12944":{"all":[null,null,null],"last":null,"last_packet_loss":100.0,"alert":true,"source":"Area: WW","alert_reasons":["loss"]},
"2195":{"all":[null,null,null],"last":null,"last_packet_loss":100.0,"alert":true,"source":"Area: WW","alert_reasons":["loss"]},
```

Real-time performance monitoring

RIPE NCC

RIPE Atlas streaming

- **RIPE Atlas streaming** is an architecture that allows users to receive the measurement results as soon as they are sent by the probes - **in real time**
  - Publish/subscribe through web sockets

- There are two types of data:
  - Measurement results
  - Probe connection status events

Real-time streaming uses

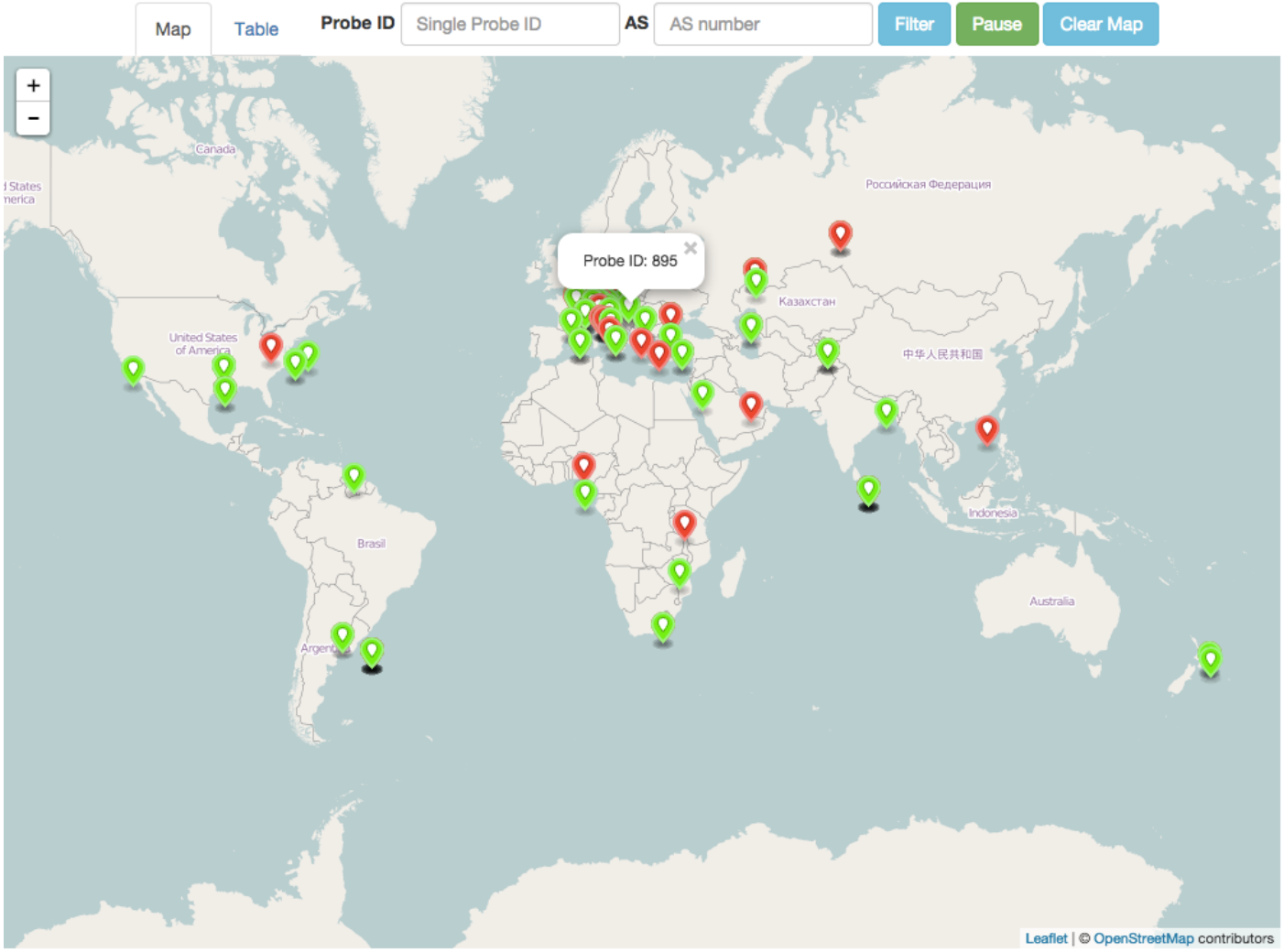
- Visualising network outages
- Server and performance monitoring

- In March 2015: used by almost all hackathon teams:

  https://labs.ripe.net/Members/becha/ripe-atlas-hackathon-results

- Documentation:
  - https://atlas.ripe.net/docs/result-streaming/
  - https://labs.ripe.net/Members/suzanne_taylor_muzzin/data-streaming-in-ripe-atlas

Probe (dis)connection events



https://labs.ripe.net/Members/andreas_strikos/amsterdam-power-outage-as-seen-by-ripe-atlas

Exercise:
Using streaming API

RIPE NCC

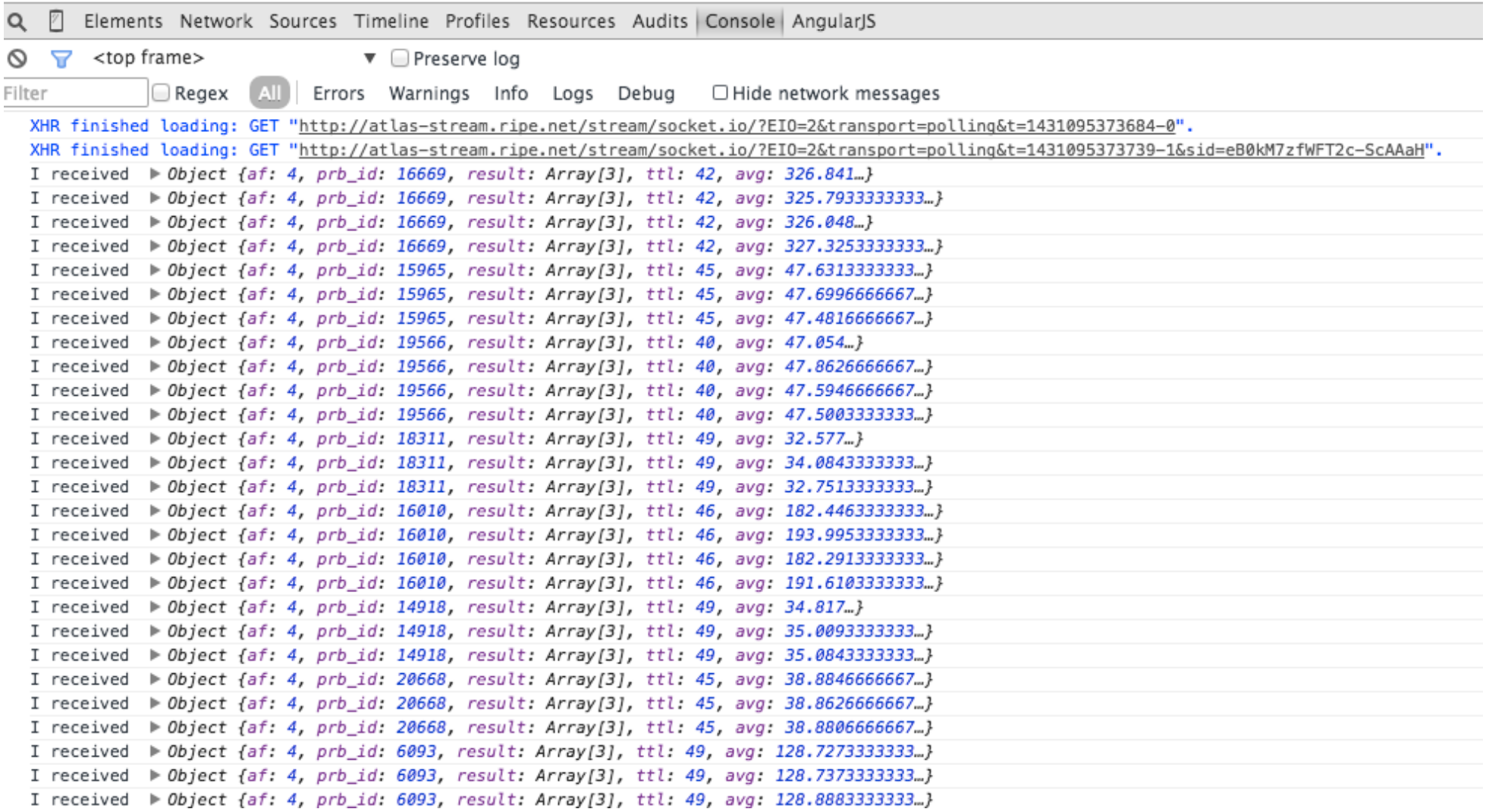
Exercise: Monitoring server reachability

- Scenario: customers are complaining that it occasionally takes a long time to reach your service or server
- Action: ping your server from 500 probes
  - Decide what is acceptable latency threshold to apply
  - Notice and react when you start receiving samples
- Task: Use the ping measurement ID1791207
  - Choose which threshold (e.g. greater than 30ms)
  - Imposes the threshold on “min” (the minimum result of the three ping attempts)

Steps

1. Go to https://stat.ripe.net/widgets/workshops/webinar/ripe-atlas/streaming-01.html
2. Open the development console
3. Wait for results to arrive
4. Optional: Save the HTML file locally and edit the code to your liking

Example of results



Take part in the RIPE Atlas community

RIPE NCC

RIPE Atlas community (part 1)

- Individual volunteers host **probes** in homes or offices

- Organisations host RIPE Atlas **anchors**

- **Sponsor** organisations give financial support or host multiple probes in their own networks

RIPE Atlas community (part 2)

- **Ambassadors** help distribute probes at conferences, give presentations, etc.
- **Developers** contribute free and open software
- **Network operators** create measurements to monitor and troubleshoot
- **Researchers** and **students** write papers

www.icinga.org

Contact us

- https://atlas.ripe.net & http://roadmap.ripe.net/ripe-atlas/
- Users' mailing list: ripe-atlas@ripe.net
- Articles and updates: https://labs.ripe.net/atlas
- Questions and bugs: atlas@ripe.net
- Twitter: @RIPE_Atlas and #RIPEAtlas

Additional slides

RIPE NCC

Finding Results of Public Measurements

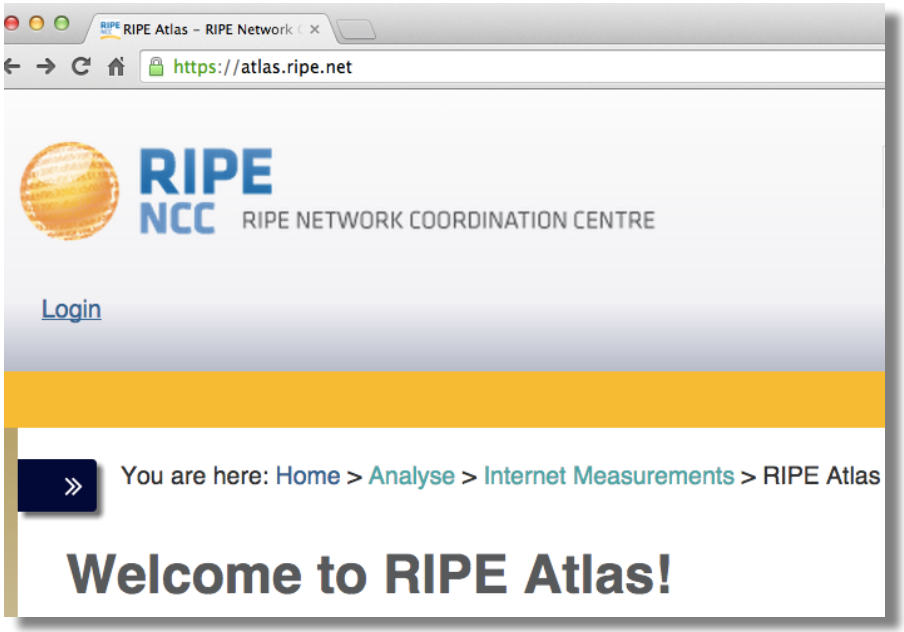
RIPE NCC

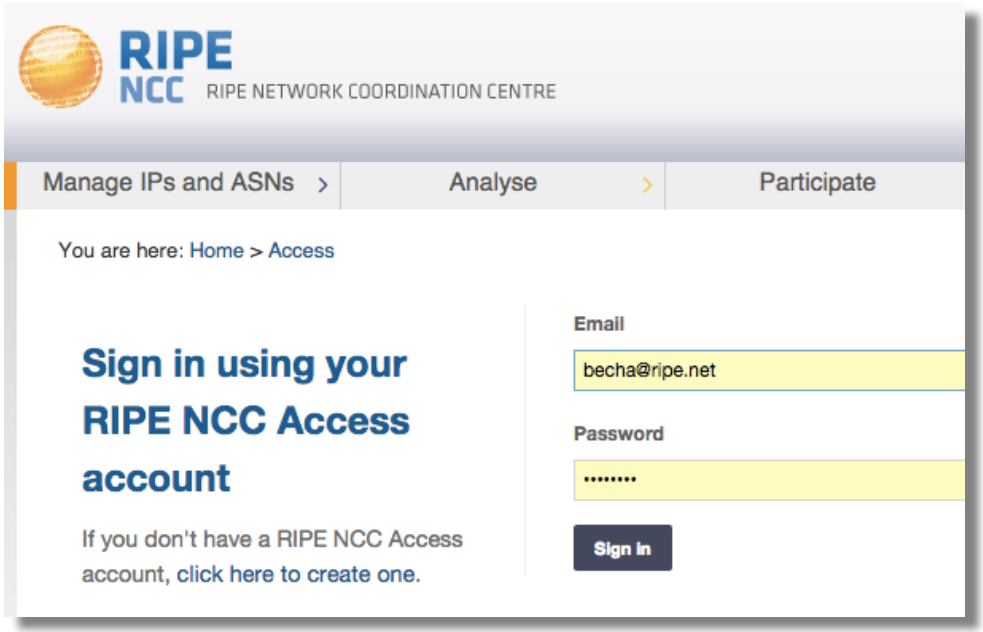
Use existing measurements

- There are many measurements already running!
- Search for existing public measurements first
- Schedule your own measurement if you don’t find what you’re looking for

Logging in

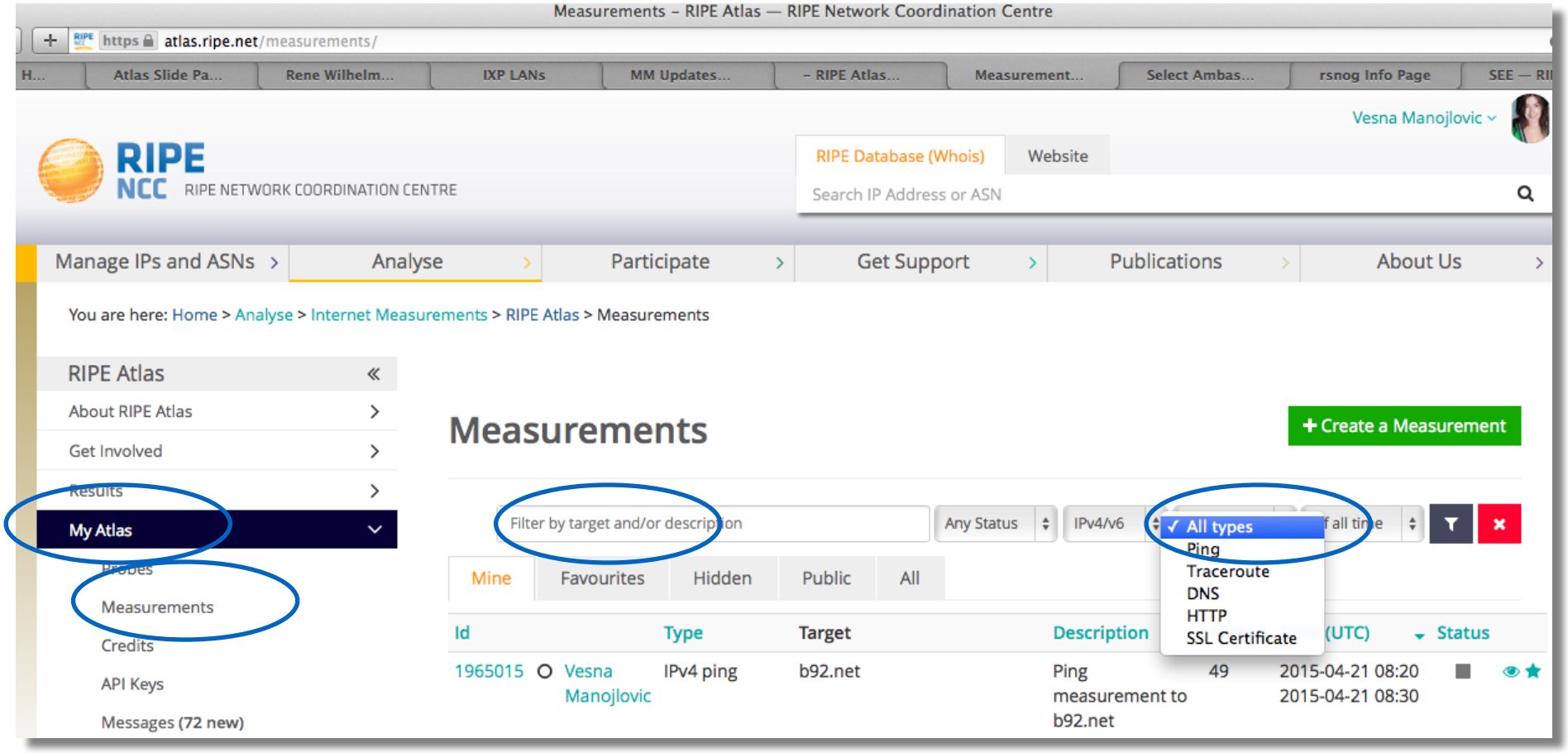
- Log in to atlas.ripe.net
  - Use your RIPE NCC Access account
  - Same account for LIR Portal, RIPE Atlas, RIPEstat, RIPE Labs...
  - Create an account if you don't have one already





Looking up measurements results

- Go to “My Atlas” > “Measurements”



Available visualisations: ping

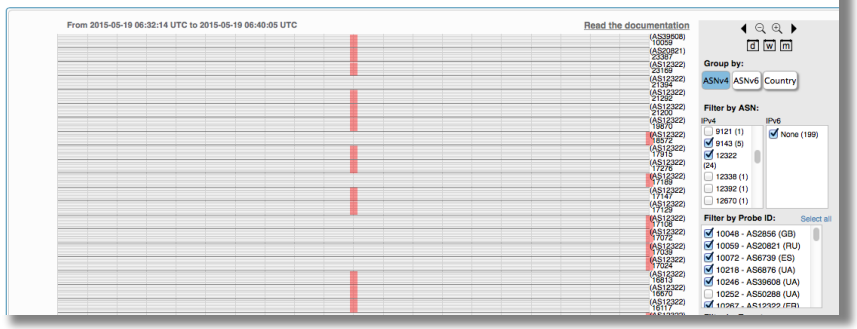
- List of probes: sortable by RTT

Probe	ASN (v4)	ASN (v6)	Time	RTT
6019	3333	3333	2015-05-19 09:23	1.157
6069	59469	59469	2015-05-19 09:23	15.253
6111	198068	198068	2015-05-19 09:23	37.760
6112	197216	197216	2015-05-19 09:23	35.494
10008	3851		2015-05-19 09:23	24.664
10218	6876		2015-05-19 09:23	37.952
10246	39608		2015-05-19 09:23	36.313
10252	50288		2015-05-19 09:23	62.441
10267	12322		2015-05-19 09:23	31.498
10296	51214		2015-05-19 09:23	✖ Unreachable

- Map: colour-coded by RTT

- Seismograph: stacked multiple pings with packet loss

An interactive visualisation for ping measurements.



Seismograph tips

Red = packet loss

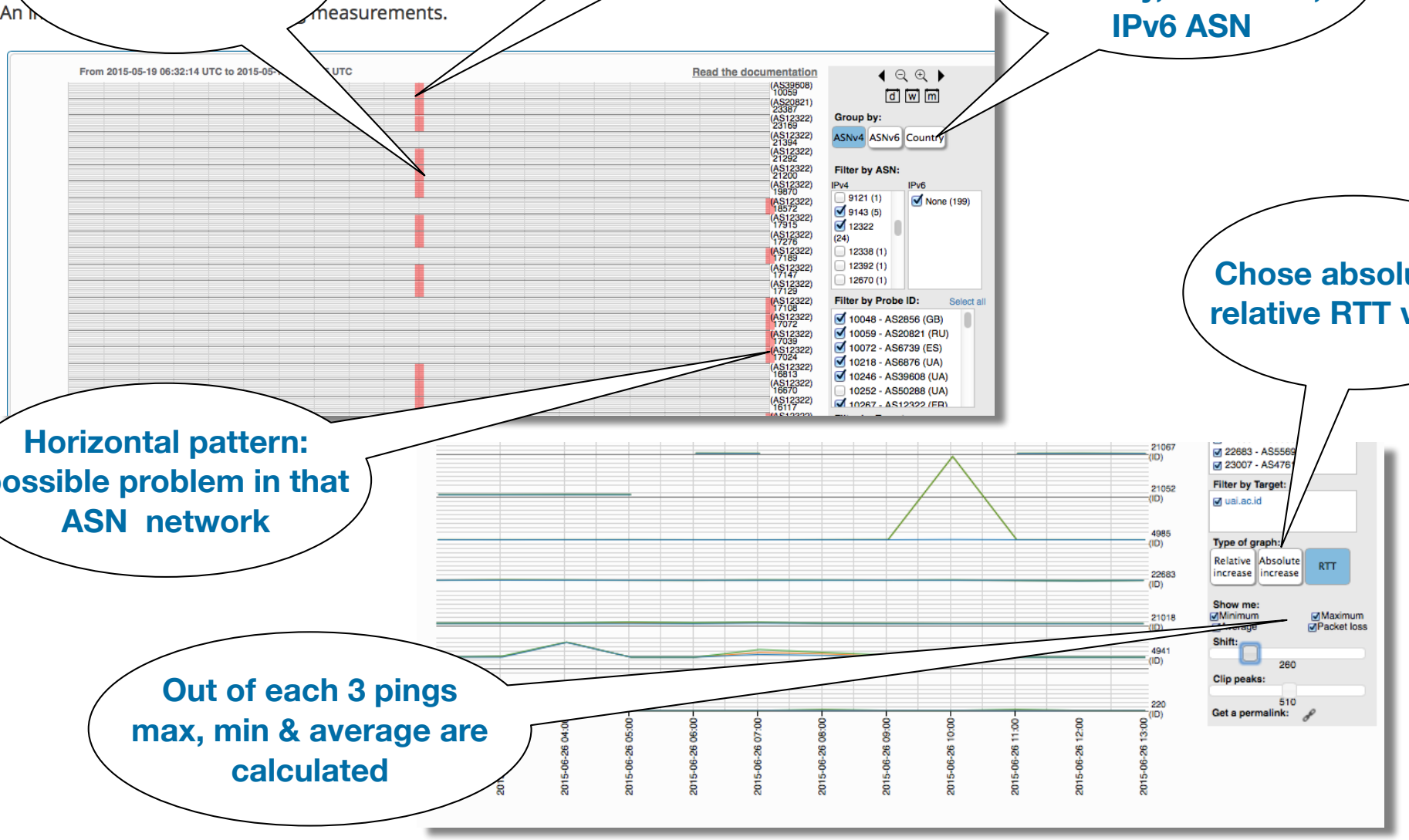
Vertical pattern: possible problem in your network

Filtering & grouping by: country, IPv4 ASN, IPv6 ASN

Chose absolute or relative RTT values

Horizontal pattern: possible problem in that ASN network

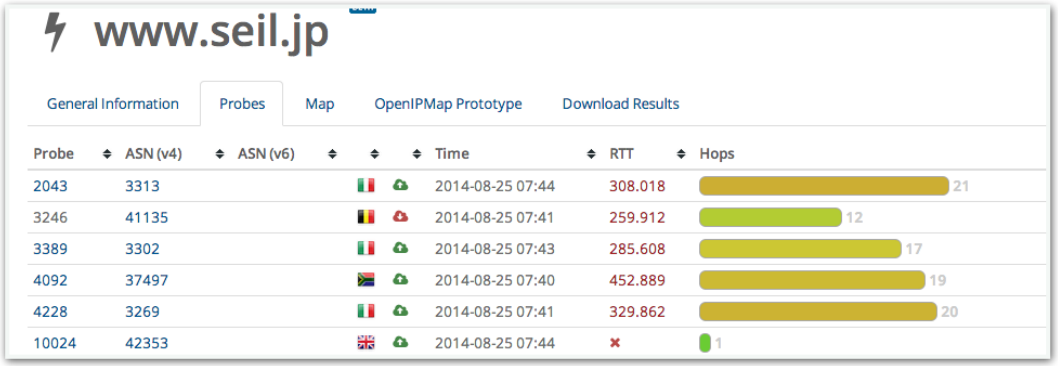
Out of each 3 pings max, min & average are calculated



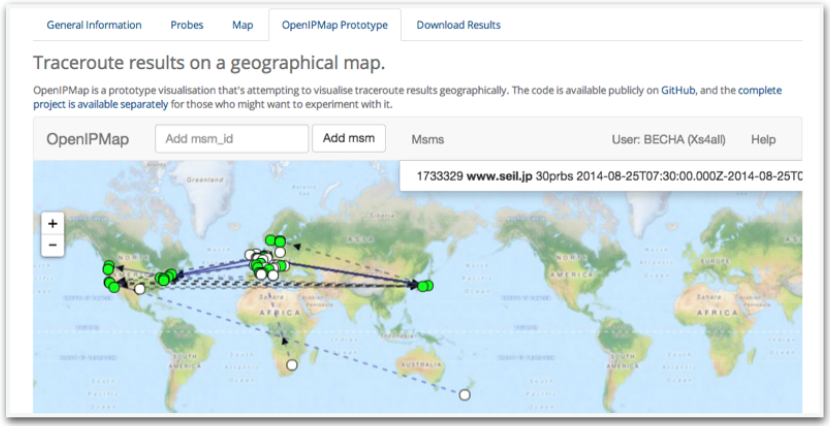
Available visualisations: traceroute

- List of probes, colour-coded number of hops

- Map



- Traceroute paths map, geolocation using OpenIPMap:
  https://github.com/RIPE-Atlas-Community/openipmap



Available visualisations: DNS

- Map, colour-coded response time or diversity

- List of probes, sortable by response time

DNS measurement to ns1.opteamax.de

General Information | Probes | Map | Download Results | Modification Log

Probe	ASN (v4)	ASN (v6)		Time	Name	Response Time
17840	6327			2015-05-19 09:38	null	362.009
18035	43030			2015-05-19 09:50	null	347.39
18129	327805			2015-05-19 09:49	null	207.743
15844	32098			2015-05-19 09:48	null	184.237
17857	852			2015-05-19 09:37	null	177.694
19894	6327			2015-05-19 09:36	null	168.689
19204	21513			2015-05-19 09:50	null	141.199
15922	30036			2015-05-19 09:47	null	133.309

Documentation

- Documentation for analysing measurements results:
  - https://atlas.ripe.net/docs/rest/
  - https://github.com/RIPE-NCC/ripe.atlas.sagan
- More tools:
  - https://github.com/RIPE-Atlas-Community
  - https://github.com/RIPE-Atlas-Community/ripe-atlas-community-contrib/blob/master/README.md

Exercise: Analyse Measurement Results

RIPE NCC

Tasks

- Download results of a specific public measurement
- Read the text of the result, to understand structure

Task 1: Download measurement results

- Find the measurement
  - ping, IPv6 to google.com
  - msm-ID 1004005
- Click on measurement, then “Download”
  - Specify the time period
  - (for example, YESTERDAY)
- Results in JSON

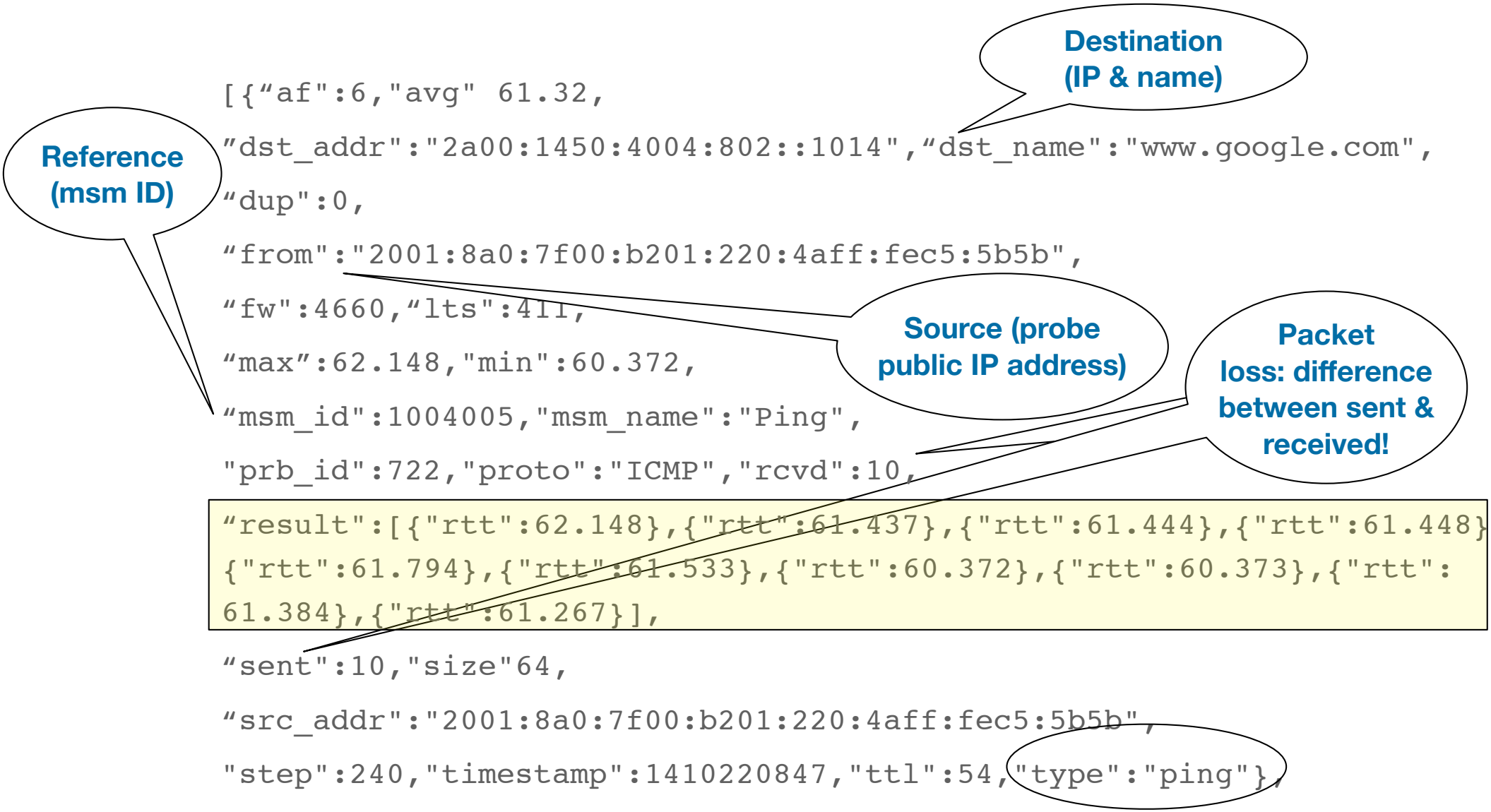
Tips for downloading results

- Solution URL:
  - https://atlas.ripe.net/api/v1/measurement/1004005/result/?start=1435104000&stop=1435276799&format=json

- Save the measurement(s) locally

```
$ curl https://atlas.ripe.net/api/v1/measurement/1004005/result/?start=1435104000&stop=1435276799&format=json > measurement-test.json
```

Task 2: Look at the result




```
[{“af":6,"avg" 61.32,
”dst_addr":"2a00:1450:4004:802::1014",“dst_name":"www.google.com",
“dup":0,
“from":"2001:8a0:7f00:b201:220:4aff:fec5:5b5b",
“fw":4660,“lts":411,
“max”:62.148,"min":60.372,
“msm_id":1004005,"msm_name":"Ping",
"prb_id":722,"proto":"ICMP","rcvd":10,
“result":[{"rtt":62.148},{"rtt":61.437},{"rtt":61.444},{"rtt":61.448}
{"rtt":61.794},{"rtt":61.533},{"rtt":60.372},{"rtt":60.373},{"rtt":
61.384},{"rtt":61.267}],
“sent":10,"size"64,
“src_addr":"2001:8a0:7f00:b201:220:4aff:fec5:5b5b",
"step":240,"timestamp":1410220847,"ttl":54,"type":"ping"},
```

Task 3: analyse results (optional)

- Find out how many times RTT was above 60ms
  - Use Python o Javascript or something else

- For the Javascript solution, you can use this as a starting point:
  - https://stat.ripe.net/widgets/demo/script_me.html

Task 3: Examples of code

Python:

Parse json and find total avg:

```
import json
f = open(“measurement.json”,”r”)
measurements = json.load(f)
for m in measurements:
    for r in m[“result”]:
        rtt = r[“rtt”]
ifrtt >60: i += 1
```

i must be > than 14563.

Javascript:

```
<script>
var dataAPIUrl = "https://atlas.ripe.net/api/v1/
measurement/1004005/result/?start=1410220800";
jQuery.ajax({

url: dataAPIUrl, error: function() {

alert("error"); },

success: function( response ) { var i = 0;

for ( var i = 0, n = response.length; i < n; i++) { var
measurement = response[i];

for ( var j = 0, m = measurement.result.length; j < m; j++) {
var rtt = measurement.result[j].rtt;
console.log(rtt);
if (rtt > 60)

i++; }

}
jQuery("p").html("The RTT has been above 60ms for " + i
+ " times");

},

dataType: "jsonp" });

</script>
```

RIPE Atlas Anchors

RIPE NCC

RIPE Atlas anchors

- Well-known targets and powerful probes
  - Regional baseline and “future history”

- Anchoring measurements
  - Measurements between anchors
  - 200 probes targeting each anchor with measurements
  - Each probe measures 4-5 anchors
  - Vantage points for DNSMON service

- 130+ RIPE Atlas anchors

Locations of anchors

https://atlas.ripe.net/results/maps/network-coverage/#anchors

Measuring Impact of IXPs on Keeping Traffic Local

"IXP Country Jedi"

RIPE NCC

Benefits (part 1)

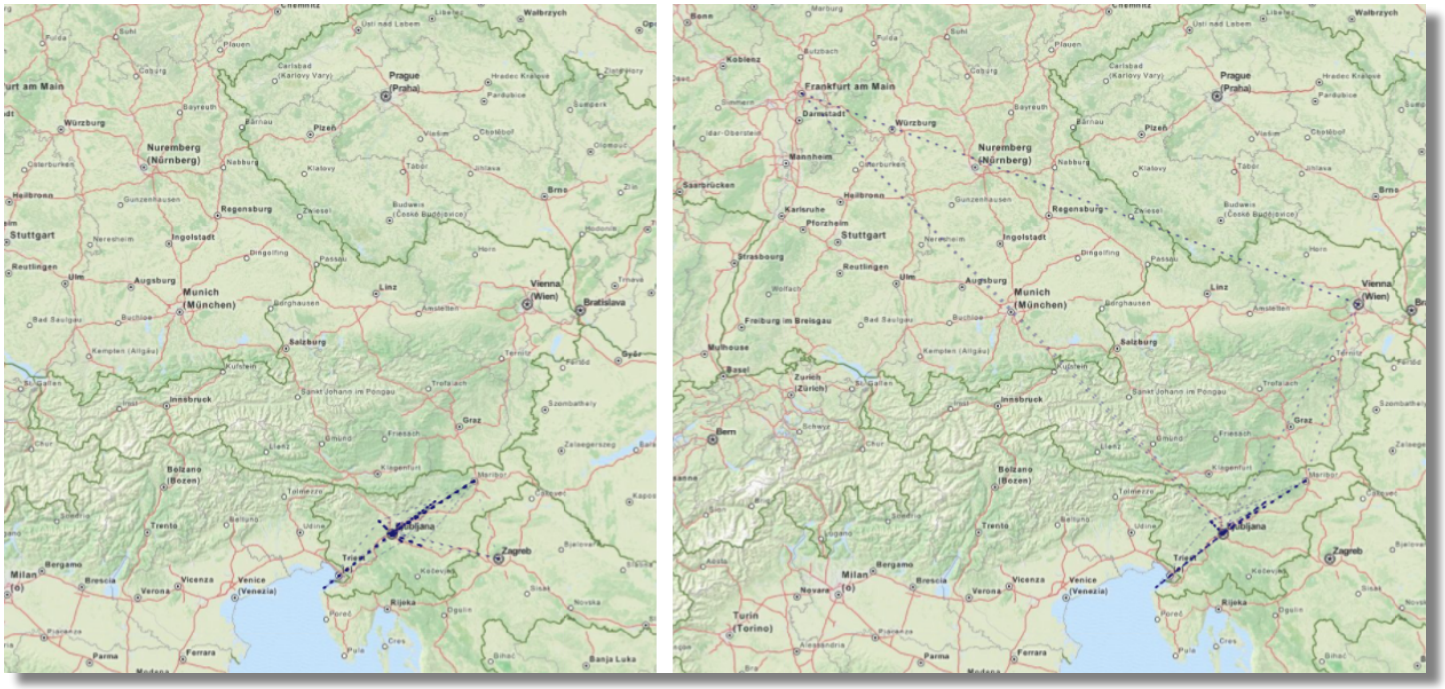
- Operators
  - Routing and traffic optimisation
- IXP operators
  - Shows how IXPs help keep traffic local and regional
- IPv6 advocates
  - Comparing IPv4 and IPv6 paths
- Country level: regulators, politicians, cyber-security…
  - How much traffic stays within the country? Where do the paths go?
  - Comparing countries with each other

Benefits (part 2)

- RIPE Atlas community
  - More probes in more networks = higher quality of measurements data
- Geolocation data community
  - Use case for improving data quality
- Examples:
  - https://labs.ripe.net/Members/emileaben/measuring-ixps-with-ripe-atlas
  - https://labs.ripe.net/Members/emileaben/measuring-countries-and-ixps-in-the-see-region
  - http://sg-pub.ripe.net/emile/ixp-country-jedi/CL+AR-2015-04/geopath/

Paths staying in the country?

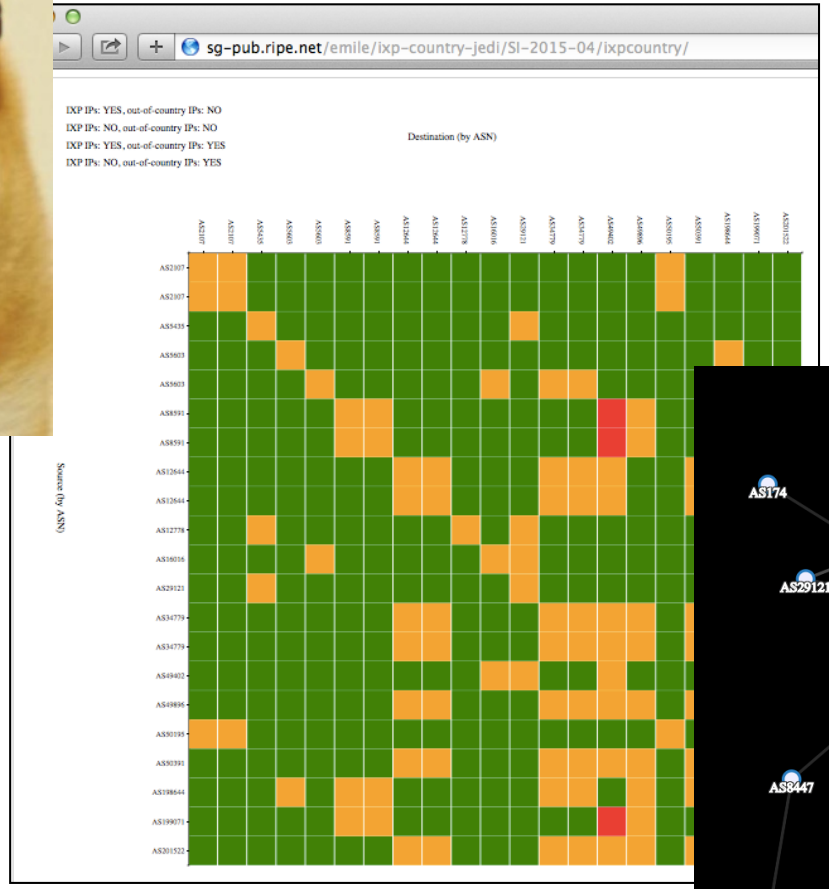
- Difference between IPv4 and IPv6 paths

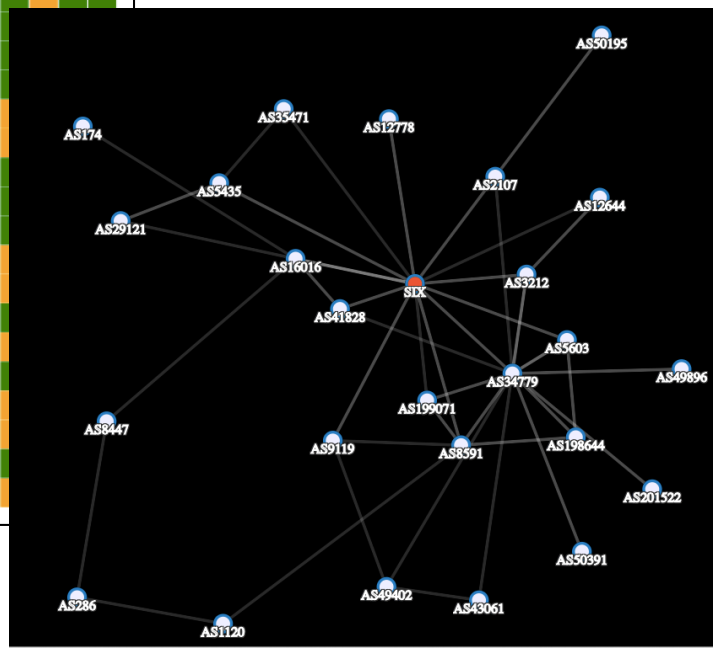


http://sg-pub.ripe.net/emile/ixp-country-jedi/**SI**-2015-04/geopath/s/**SI**/{RO, BG, HR, BA, ME, AL, GR}/

Paths going via an IXP?

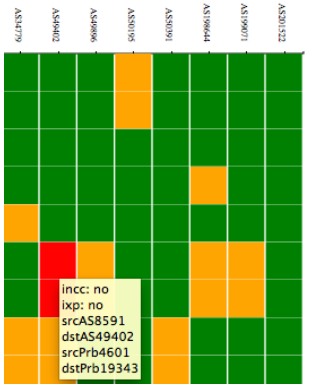
http://sg-pub.ripe.net/emile/ixp-country-jedi/SI-2015-04/





Interactive diagnostic tool

http://sg-pub.ripe.net/emile/ixp-country-jedi/SI-2015-04/ixpcountry/




```
## msm_id:1962254 prb_id:4601 dst:193.169.48.40 ts:2015-04-16 09:01:06 -00:00
1 (AS8591) maribor10-ge-2-20-v987.amis.net [1.593, 1.602, 2.292] |Maribor,Maribor,SI
2 (AS8591) mx-mb1-te-1-2-0.amis.net [1.619, 1.697, 1.944] ||
3 (AS8591) mx-ljl-te-2-3-1.amis.net [3.599, 3.865, 5.148] ||
4 (AS8591) mx-zgl-xe-2-0-1.amis.net [5.568, 5.576, 5.69] ||
5 () 75.64-127.15.192.193.in-addr.arpa [5.955, 5.98, 5.985] |Zagreb,Grad Zagreb,HR|
6 (AS9119) 212.13.240.249 [5.778, 5.83, 5.935] ||
7 (AS9119) 212.103.133.4 [7.099, 7.84, 7.926] ||
8 (AS9119) 212.13.240.62 [6.597, 7.674, 7.696] ||
9 (AS9119) hsl.gw0.hsl.eu [5.833, 6.079, 6.368] ||
10 (AS49402) ntp.hsl.eu [6.657, 7.273, 8.155] ||
11 (AS49402) 193.169.48.40 [6.661, 6.691, 6.872] ||
```

- Green: “good”, as far as we can see it
  - Not a judgment, only one way of visualising data
- Red or blue: path is going out of country
  - If this is a surprise: talk to your upstream(s)
- Yellow: path is not going via a local IXP
  - If this is undesired: make a new peering agreement

Method

- traceroute measurements using RIPE Atlas probes

- Steps:
  - Identify ASNs in the country using RIPEstat
  - Identify IXPs and IXP LANs using PeeringDB
  - Construct mesh: from all (*) country's probes to each other

  *Maximum of two probes per ANS and only "public" probes with "good" geolocation

- Hops geolocated using "OpenIPMap" database

Actions (part 1)

- Use this tool to find possible suboptimal routing and fix it
  - Find your ASN in the mesh
  - Find the person from another ASN
  - Take them out for tea :)

- To improve accuracy of this diagnostic tool
  - If your ASN is not on the graph, apply for a RIPE Atlas probe
  - Add more probes to your country to increase “resolution”
  - If you move, remember to update your probe’s geolocation

Actions (part 2)

- Re-use and rewrite the code: it is free and open source software
  - https://github.com/emileaben/ixp-country-jedi

- Improve infrastructure geolocation: contribute data to OpenIPMap!
  - https://marmot.ripe.net/openipmap/
  - https://github.com/RIPE-Atlas-Community/openipmap