

CORE NETWORK DONE! LAST UPDATE BEFORE SUMMER.

Before the vacation-period kicks-in i wanted to update about what the latest news is in the SUNET-C project.

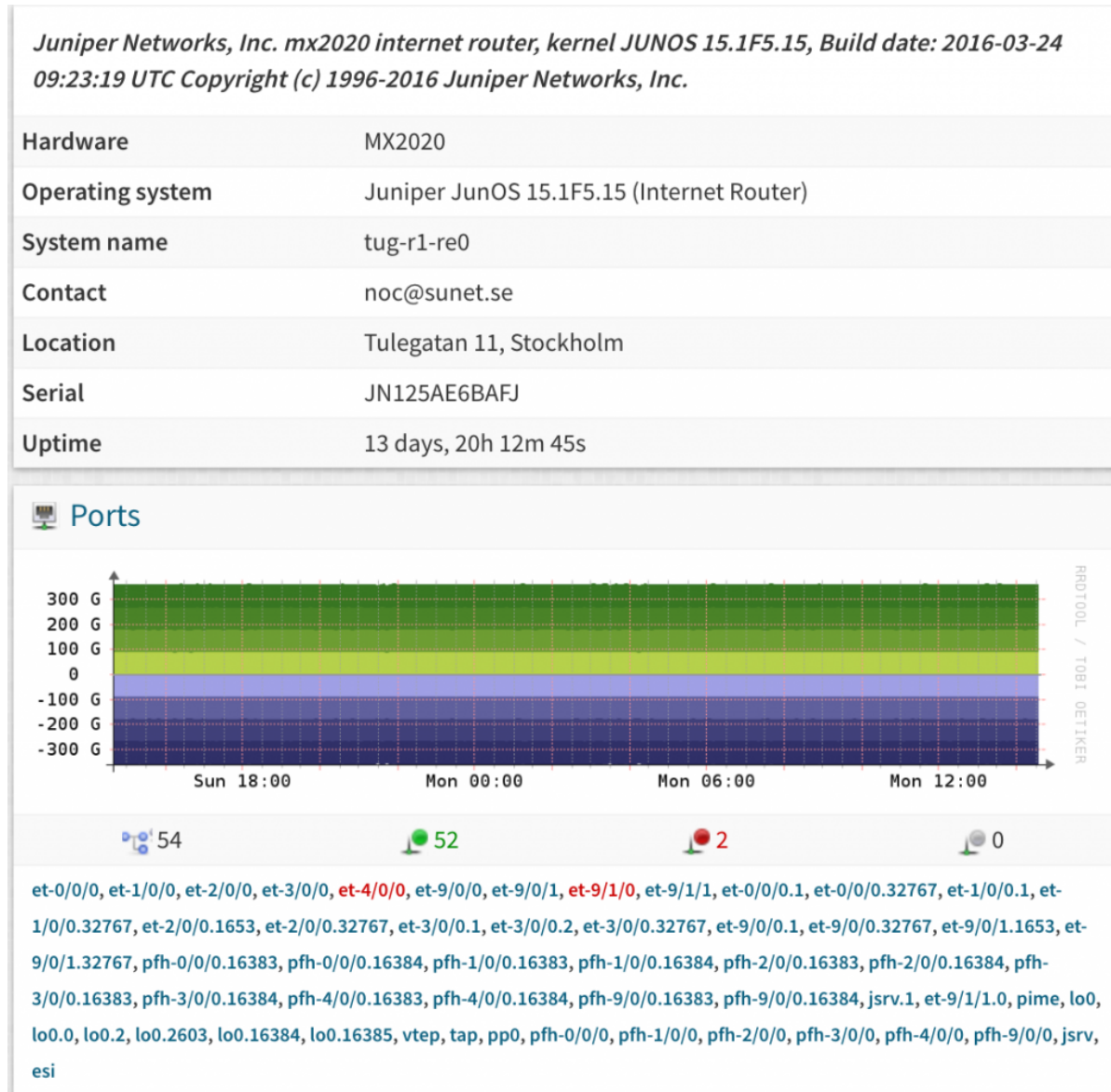
Our first major milestone has finally happened!

A few days ago our core-network become fully built out with the arrival of the last two core-routers! One in Stockholm and one in Narvik (norway). We still have a few open incidents on bad fiber and sites not living up to our standards but we can get bits to flow in the whole network at least so all paths is alive and kicking, but some is not performing fully. We don't even anticipate to be fully satisfied with all "layer1" deliveries until next year even.

To visualize the core, we built a MPLS-tunnel throughout the whole network that touches every single core-interface in the network, might look abit silly but this is a great way to find out faulty links actually (which we did on one of the links in Halmstad as you can see in the MTR underneath), with one ping you can test your whole network for functionality which can come in handy.

Host	Loss%	Snt	Last	Avg	Best	Wrst	StDev
1. xlfre-vrrp2.sunet.se	0.0%	80	0.4	4.9	0.3	67.0	12.6
2. mlfre-xe-4-1-0.sunet.se	0.0%	80	2.3	1.4	0.3	12.6	2.0
3. mlug.sunet.se	0.0%	80	0.4	1.3	0.3	13.9	2.0
4. stockholm-tug-r1.sunet.se	0.0%	80	0.4	1.4	0.4	14.4	1.8
5. uppsala-upa-r1.sunet.se	0.0%	80	2.4	2.3	1.8	8.3	0.8
6. stockholm-fre-r1.sunet.se	0.0%	80	1.8	3.4	1.7	22.2	2.7
7. uppsala-upa-r1.sunet.se	0.0%	80	2.7	3.5	2.7	20.2	2.7
8. gavle-sbo-r1.sunet.se	0.0%	80	4.0	4.4	4.0	21.8	2.2
9. sundsvall-sva1-r1.sunet.se	0.0%	80	6.4	6.8	6.3	24.3	2.2
10. umea-ume8-r1.sunet.se	0.0%	80	9.5	10.5	9.4	30.0	3.4
11. lulea-lla1-r1.sunet.se	0.0%	80	14.6	13.5	12.4	17.5	0.9
12. lulea-lla1-r2.sunet.se	0.0%	80	14.4	14.8	14.3	31.7	2.0
13. lulea-lla1-r1.sunet.se	0.0%	80	12.5	14.0	12.4	30.6	2.7
14. kalix-klx1-r1.sunet.se	0.0%	80	19.0	16.7	13.3	24.9	3.0
15. kiruna-kir3-r2.sunet.se	0.0%	80	19.1	20.2	18.9	33.3	3.0
16. kiruna-kir3-r2.sunet.se	0.0%	80	19.0	19.3	18.9	31.3	1.5
17. falun-fln4-r1.sunet.se	0.0%	80	19.1	19.7	18.9	35.8	2.4
18. kiruna-kir3-r1.sunet.se	0.0%	80	46.0	33.2	32.7	46.0	1.7
19. lulea-lla1-r2.sunet.se	0.0%	80	34.0	34.4	33.9	45.8	1.6
20. narvik-kir5-r1.sunet.se	0.0%	80	41.4	41.9	41.3	59.2	2.5
21. lulea-lla1-r1.sunet.se	0.0%	80	41.5	42.6	41.4	52.5	2.0
22. karlstand-kar1-r1.sunet.se	0.0%	80	41.4	41.8	41.3	50.1	1.4
23. orebro-lba-r1.sunet.se	0.0%	80	41.4	42.3	41.3	58.0	3.1
24. falun-fln4-r1.sunet.se	0.0%	80	43.4	43.7	43.3	48.0	0.8
25. sundsvall-sva1-r1.sunet.se	0.0%	80	47.3	48.0	47.3	66.8	2.5
26. falun-fln4-r1.sunet.se	0.0%	80	50.8	51.2	50.7	61.7	1.5
27. orebro-lba-r1.sunet.se	0.0%	80	50.8	51.9	50.7	68.0	3.3
28. vasteras-fsn2-r1.sunet.se	0.0%	80	50.8	51.2	50.7	62.1	1.4
29. stockholm-tug-r1.sunet.se	0.0%	80	50.9	51.6	50.8	59.9	1.8
30. vasteras-fsn2-r1.sunet.se	0.0%	80	52.0	52.1	50.9	60.3	0.9
31. orebro-lba-r1.sunet.se	0.0%	80	53.2	53.3	51.0	55.9	0.6
32. skovde-ske109-r1.sunet.se	0.0%	80	54.9	55.1	52.6	60.6	1.0
33. jonkoping-jkg-r1.sunet.se	0.0%	80	56.0	56.8	53.7	73.4	2.9
34. boras-bos-r1.sunet.se	0.0%	80	57.2	57.3	55.5	60.9	0.6
35. jonkoping-jkg-r1.sunet.se	0.0%	80	57.2	58.1	54.9	73.4	2.9
36. linkoping-lin3-r1.sunet.se	0.0%	80	70.3	58.0	57.1	73.3	2.7
37. norrkoping-nkg188-r1.sunet.se	0.0%	80	57.2	57.6	57.1	70.0	1.5
38. stockholm-tug-r1.sunet.se	0.0%	80	57.3	57.9	57.2	67.2	1.5
39. norrkoping-nkg188-r1.sunet.se	0.0%	80	59.5	60.3	59.5	70.8	1.9
40. stockholm-fre-r1.sunet.se	0.0%	80	60.5	62.6	59.5	83.1	4.6
41. vasteras-fsn2-r1.sunet.se	0.0%	80	61.0	61.7	61.0	80.3	2.4
42. stockholm-fre-r1.sunet.se	0.0%	80	63.6	63.3	61.4	76.9	2.3
43. norrkoping-nkg188-r1.sunet.se	0.0%	80	63.5	63.8	63.5	81.3	2.0
44. linkoping-lin3-r1.sunet.se	0.0%	80	64.7	64.8	64.2	72.2	1.1
45. jonkoping-jkg-r1.sunet.se	0.0%	80	68.3	66.2	66.0	70.1	0.6
46. vaxjo-vxj3-r1.sunet.se	0.0%	80	67.5	67.9	67.5	70.9	0.7
47. kalmar-kmr88-r1.sunet.se	0.0%	80	70.0	69.3	68.7	80.3	1.3
48. karlskrona-kka8-r1.sunet.se	0.0%	80	70.4	70.7	70.1	73.0	0.7
49. kristianstad-ksd1-r1.sunet.se	0.0%	80	71.7	72.7	71.6	89.3	2.9
50. malmo-mcen1-r1.sunet.se	0.0%	80	71.8	72.5	71.7	89.3	2.3
51. lund-lnd88-r1.sunet.se	0.0%	80	79.1	72.6	71.6	90.7	3.5
52. malmo-mcen1-r1.sunet.se	0.0%	80	72.0	72.7	71.9	88.8	2.7
53. kristianstad-ksd1-r1.sunet.se	0.0%	80	73.7	74.3	73.7	83.1	1.4
54. karlskrona-kka8-r1.sunet.se	0.0%	80	73.8	74.5	73.7	91.1	2.5
55. kalmar-kmr88-r1.sunet.se	0.0%	80	74.6	74.4	73.7	92.8	2.4
56. vaxjo-vxj3-r1.sunet.se	0.0%	80	73.8	74.5	73.7	87.8	2.3
57. lund-lnd88-r1.sunet.se	0.0%	80	75.8	76.6	75.8	93.1	2.6
58. halmsstad-hsd1-r1.sunet.se	10.1%	80	77.6	77.8	77.5	80.2	0.6
59. goteborg-gbg7-r2.sunet.se	1.3%	79	76.9	77.8	76.8	90.9	2.4
60. halmsstad-hsd1-r1.sunet.se	5.1%	79	79.5	80.0	79.2	94.0	2.1
61. lund-lnd88-r1.sunet.se	0.0%	79	80.2	79.9	79.3	92.4	1.6
62. vaxjo-vxj3-r1.sunet.se	1.3%	79	79.3	79.8	79.3	88.7	1.3
63. jonkoping-jkg-r1.sunet.se	3.8%	79	80.4	80.3	79.3	89.5	2.3
64. skovde-ske109-r1.sunet.se	3.8%	79	79.7	80.4	79.6	97.9	2.5
65. orebro-lba-r1.sunet.se	3.8%	79	79.9	80.2	79.6	85.4	0.9
66. karlstand-kar1-r1.sunet.se	1.3%	79	81.0	81.4	80.9	88.3	1.1
67. trollhattan-trh-r1.sunet.se	1.3%	79	83.1	83.4	83.0	101.6	2.1
68. goteborg-gbg7-r1.sunet.se	3.8%	79	86.4	84.8	84.1	103.2	2.4
69. goteborg-gbg7-r1.sunet.se	3.8%	79	84.2	84.9	84.2	95.3	2.0
70. goteborg-gbg7-r2.sunet.se	3.8%	79	86.8	85.4	84.1	114.3	3.9
71. trollhattan-trh-r1.sunet.se	10.1%	79	84.1	84.4	84.0	94.0	1.3
72. karlstand-kar1-r1.sunet.se	5.1%	79	84.1	85.1	84.0	103.3	3.9
73. lulea-lla1-r2.sunet.se	1.3%	79	95.2	95.3	95.1	97.9	0.4
74. kiruna-kir3-r1.sunet.se	7.6%	79	100.3	100.1	99.7	102.8	0.5
75. kiruna-kir3-r1.sunet.se	8.9%	79	99.9	100.1	99.8	104.9	0.7
76. kalix-klx1-r1.sunet.se	3.8%	79	100.9	103.0	100.8	109.0	1.8
77. lulea-lla1-r1.sunet.se	3.8%	79	101.0	100.5	99.0	110.2	1.6
78. umea-ume8-r1.sunet.se	5.1%	79	98.9	99.7	98.9	109.7	1.3
79. sundsvall-sva1-r1.sunet.se	1.3%	79	99.7	99.6	98.9	109.0	1.5
80. gavle-sbo-r1.sunet.se	1.3%	79	98.9	99.7	98.8	106.0	1.0
81. uppsala-upa-r1.sunet.se	5.1%	79	100.9	99.9	98.9	108.1	1.5
82. 192.168.255.2	3.8%	79	100.2	99.9	99.0	107.1	1.3

In this same tunnel we have also connected a 100G traffic generator and run every single interface in the network at 100% utilization, this is also a good way of finding faults if traffic is lost somewhere, here below shows the total util-graph from tug-r1 which carries the tunnel three times through the router and therefore gets 600G in total utilization



While speaking about tug-r1 (one of the Juniper MX2020 core routers in Stockholm) there was quite a logistical effort to even get the router installed. To not break any fragile network engineers backs we contacted a firm which usually carries pianos to haul it up to the datacenter at Tulegatan. A stripped MX2020 is few hundred kilos but with the right tools, its not a problem getting it up to the second floor without a elevator, or big enough doors.

The access to this service is forbidden since the specified partner is blocked

The access to this service is forbidden since the specified partner is blocked

The provisioning of the configuration in the whole backbone is almost all done. There will be a more detailed post later on how we achieve this but essentially the only configuration we manually configure by hand in the network is 7 lines of configuration to make the router internally reachable so we can provision it centrally – after that we generate most other configurations. We use the tool "NSO" from Cisco (Which was NCS by Tail-F before Cisco acquisition). Inside this we either write text-based templates with variables or we build services using YANG-models depending if the configuration is just plaintext or something that is dynamic (such as bgp-peering rules, or bandwidth-on-demand type of services).

Below is an example of a template that writes generic routing-option settings (inside a logical system). Nothing exotic, but handy and effective. Then on top of that you build template-groups so you can bulk-push all the templates you need if certain criteria are met,

and then nested templates...and then...yeah... can get quite advanced. Its also handy to with one command validate the whole network against the template.

```
hugge@ncs% show devices template ?
Description: Named configuration templates for devices
Possible completions:
BGP - The name of a specific template configuration.
BGP-CPE - The name of a specific template configuration.
bgp-rrc-lsys - The name of a specific template configuration.
delete-re-protect-v4 - The name of a specific template configuration.
eth-apply-group-template - The name of a specific template configuration.
iBGP - The name of a specific template configuration.
iBGP-SUNET-CORE-md5 - The name of a specific template configuration.
iBGP-template - The name of a specific template configuration.
isis-template - The name of a specific template configuration.
isis-template-lsys - The name of a specific template configuration.
re-protect-v4 - The name of a specific template configuration.
re-protect-v4-lsys - The name of a specific template configuration.
re-protect-v6 - The name of a specific template configuration.
re-protect-v6-lsys - The name of a specific template configuration.
routing-options - The name of a specific template configuration.
routing-options-lsys - The name of a specific template configuration.
source-address-template - The name of a specific template configuration.
ssh-ip - The name of a specific template configuration.
ssh-ipnett-bjorne - The name of a specific template configuration.
system-template - The name of a specific template configuration.
Possible match completions:
config - This container is augmented with data models from the devices.
hugge@ncs% show devices template routing-options-lsys
config {
  junos:configuration {
    logical-systems SUNET {
      routing-options {
        protect {
          core;
        }
        autonomous-system {
          as-number 1653;
        }
      }
    }
  }
}
[ok] [2016-06-27 19:03:28]

[edit]
hugge@ncs% █
```

A "service" is slightly more exotic, this is a customer HIG (Högskolan i Gävle) BGP-peering service which runs in the old OptoSunet network that is provisioned already. This has slightly more intelligence than the plain configuration-templates. Below we have pre-defined models on import and export rules and use input-information such as prefix-lists, RIPE IRR data (read from the variable "as" which queries IRR-database for info using that asn) and behind this it generates config and pushes it out through Netconf to the corresponding routers (m1fre and m1tug). There is also of script-chaining behind the services to be able to produce the configuration dynamically and on-demand (for example it triggers when a IRR-data change is noticed on our monitored objects)

Also in reality it does not care what vendor the router is from – that is converted internally using published Yang-models (if those exists), but in this case its Juniper straight through so it only needs to translate into one set of models.

```
AS1653 - unique service id
hugge@ncs% show services service AS16251
type {
  bgp-customer {
    as 16251;
    import-rules [ reject-Martians hig-in ];
    export-rules [ default-originate NREN SUNET-nets originate-nets reject-all ];
    prefix-list [ 130.243.0.0/20 193.11.48.0/20 ];
    prefix-list-v6 [ 2001:6b0:23::/48 ];
    description HIG;
    prefix-limit {
      maximum 10;
    }
  }
  routers mlfre {
    peering 193.11.20.66;
    peering 2001:6b0:dead:beef:2::342 {
      export-rules [ SUNET-nets ];
    }
  }
  routers mltug {
    peering 193.11.20.70;
    peering 2001:6b0:dead:beef:2::346 {
      export-rules [ default-originate NREN reject-all ];
    }
  }
}
[ok] [2016-06-27 16:12:43]
[edit]
hugge@ncs% █
```

This is also constantly sanitized and validated centrally to avoid having "im just gonna fix this little thing" hack-configurations laying around in the network everywhere which acts as both security holes and could also potentially break things completely in the future if not hacked together carefully (which it never is)

We will continue during the summer to build the last templates and services for customers to make sure that whenever CPE-devices is being installed out at the university we can acheive almost a Zero-Touch provisioning behaviour, the idea is that the local field-engineers consoles in to the router (since we dont have 4G OOB to CPEs as we have in the core), enter hostname, set an ip-adress and enable SSH and then the rest of the configuration is pushed out centrally as soon as the router "calls home" so the field-engineer can focus on what he does best, install things correctly. We use RFC1918 adress-space for commisioning first so all access-links is already configured with private address-space that is just routed internally and just used for commisioning configuration.

Skriven av



FREDRIK "HUGGE" KORSBÄCK

Network architect and chaosmonkey for AS1653 and AS2603. Fluent in BGP hugge@nordu.net