SUNET

# A TRIP DOWN THE NON-VOLATILE MEMORY LANE

Autumn is here and it's time to make the Internet great again and get down and dirty in the technical bits.

First we should look into what has happened since the last time we talked about SUNET-C. The IP and Optical core is 100% completed since the middle of the summer. We do have almost all of our paths in the core fully working, but we are not satisfied with the performance of all of them and the ones that are currently left needs more work then just "clean some fibers" or "change a patch". Site rebuilds and swap of optical spans is what's gonna need to happen for us to be satisfied with the few outstanding issues but this can done when the network is alive since we have full redundancy everywhere and on a few places we have tripple redundancy so its not a problem for us to disconnect a SPAN briefly to get new fibers or get routes through a ODF.

Do we have public weathermaps of the core? Yes we do.

http://stats.sunet.se/stat-q/load-map/SunetC-core,,traffic,peak

Do we have public weathermaps of the access network? Yes we do.

http://stats.sunet.se/stat-q/load-map/SunetC-univ,,traffic,peak

Do we have public configuration? Yes we do.

As of the date of publishing this we have installed 24 out of 54 CPE-routers and is rolling out hardware quite aggressively, we expect almost everyone to have received new hardware and have them plugged in by mid october. Our partner for hardware-rollout (Eltel) is provided with a extremely minimal configuration that they need to cut-n-paste into the router with local console when they have power, so we can reach them from the office.

This is our bootstrap-procedure.

```
R1
cli
configure
set interfaces xe-0/0/0 description "10G X-link"
set interfaces xe-0/0/0 unit 0 family inet address 192.168.1.1/31
set interfaces et-0/2/0 description "LR4 Uplink"
set interfaces et-0/2/0 unit 0 family inet address 192.168.0.1/31
set system root-authentication encrypted-password "THROWAWAYPASSWORD"
set system services ssh protocol-version v2
set system commit synchronize
set chassis network-services enhanced-ip
set forwarding-options hyper-mode
commit and-quit

R2
cli
configure
set interfaces xe-0/0/0 description "10G X-link"
set interfaces xe-0/0/0 unit 0 family inet address 192.168.1.0/31
```

```
set system root-authentication encrypted-password "THROWAWAYPASSWORD"
set system services ssh protocol-version v2
set system commit synchronize
set chassis network-services enhanced-ip
set forwarding-options hyper-mode
commit and-quit
```

With this configuration we can from the core locally reach R1 and through R1 also reach R2 through the RFC1918 adresses. We do not configure the 100G uplink for R2 in this stage since this is a DWDM-interface and required abit more thought before firing up, it also needs to be provisioned optically in the ROADMs so we take that later in the process. In this stage we manually set the correct ip-adress on CORE<->R1, R1<->R2 links, we activate ISIS on the same links and we add a user ”NCS” with a ssh-key and then we let Tail-F NCS (Cisco NSO) handle the rest of the configuration, no more human fatfingers in the routers. In NSO we have pre-built templates and services that only needs to know which customer it is suppose to configure and type of setup that customer wants, then it will autoprovision the whole node using those variables and the result is something below, some omitting of conf has been done below for obvious reasons (passwords, pfx-acls, etc)

We have produced the configuration in our lab and flag which fields and types that should be variable and generated fields.

This is what’s being provisioned to Stockholms Universitet to their router R1 for example.

(CAUTION: LONG READ FROM THIS)

```
hugge@su-r1-re0> show configuration
## Last commit: 2016-09-16 04:46:53 UTC by hugge
version 15.1F6.9;
groups {
    re0 {
        system {
            host-name su-r1-re0;
        }
    }
    re1 {
        system {
            host-name su-r1-re1;
        }
    }
    ETH {
        interfaces {
            <ae*> {
                mtu 9192;
            }
            <et-*> {
```

```
                mtu 9192;
                gigether-options {
                    ignore-l3-incompletes;
                }
            }
            <xe-*> {
                gigether-options {
                    ignore-l3-incompletes;
                }
            }
        }
    }
    ISIS_AUTH {
        logical-systems {
            SUNET {
                protocols {
                    isis {
                        level 2 {
                            authentication-key "KEY"; ## SECRET-DATA
                            authentication-type md5;
                        }
                        interface <*> {
                            level 2 {
                                hello-authentication-key "KEY"; ## SECRET-DATA
                                hello-authentication-type md5;
                            }
                        }
                    }
                }
            }
        }
    }
}
apply-groups [ re0 re1 ];
system {
    domain-name sunet.se;
    domain-search sunet.se;
    time-zone UTC;
    default-address-selection;
    internet-options {
```

```
    icmpv4-rate-limit packet-rate 10000 bucket-size 10;
    icmpv6-rate-limit packet-rate 10000 bucket-size 10;
    path-mtu-discovery;
    gre-path-mtu-discovery;
    ipv6-path-mtu-discovery;
    ipv6-path-mtu-discovery-timeout 5;
}
root-authentication {
    encrypted-password "KEY"; ## SECRET-DATA
}
name-server {
    $NS1;
    $NS2;
}
scripts {
    commit {
        file annotate.slax {
            optional;
            source "$CONFIG-DB-SERVER";
        }
    }
}
login {
    class SUNETCVIEW {
        idle-timeout 360;
        permissions [ network trace view view-configuration ];
    }
    class UNIV-LOGIN-CLASS {
        permissions all;
        deny-configuration "(logical-systems SUNET .*)";
    }
    user hugge {
        full-name "Fredrik Korsbäck";
        uid 2009;
        class super-user;
        authentication {
            ssh-rsa "ssh-rsa KEY1"; ## SECRET-DATA
        }
    }
    user su_user {
```

```
        full-name "Someone at SU";
        uid 2001;
        class UNIV-LOGIN-CLASS;
        authentication {
            ssh-rsa "ssh-rsa KEY1"; ## SECRET-DATA
        }
    }
services {
    ssh {
        root-login allow;
        tcp-forwarding;
        protocol-version v2;
        no-passwords;
        rate-limit 100;
    }
    netconf {
        ssh;
    }
}
syslog {
    user * {
        any emergency;
    }
    host $SYSLOG-ADDRESS {
        any notice;
        authorization info;
        daemon info;
        firewall none;
        interactive-commands any;
        facility-override local5;
    }
    host $SYSLOG2-ADDRESS {
        any notice;
        authorization info;
        daemon info;
        firewall none;
        interactive-commands any;
        facility-override local5;
    }
    file messages {
```

```
            any notice;
            authorization info;
            daemon info;
            firewall none;
            interactive-commands none;
            archive size 2m files 10 world-readable;
        }
        file interactive-commands {
            interactive-commands any;
            archive size 2m files 10;
        }
        file firewall {
            firewall info;
            archive size 2m files 10;
        }
    }
    compress-configuration-files;
    archival {
        configuration {
            transfer-on-commit;
            archive-sites {
                "$CONFIG-DB-SERVER";
            }
        }
    }
    commit synchronize;
    ntp {
        server 192.36.143.150;
        server 192.36.143.151;
        server 194.58.203.20;
        server 194.58.203.148;
        server 194.58.204.20;
        server 194.58.204.148;
        server 194.58.202.20;
        server 194.58.202.148;
        server 194.58.205.20;
        server 194.58.205.148;
    }
}
```

This is the ”system” part of the configuration, nothing strange going on here really. The trained eye will notice that we do not run passwords and we don’t run TACACS. We only allow ssh-keys and typically these reside on a Yubi-key or similar 2factor token so that keys can’t be stolen (as easily).

Since SU is going to utilize this router as their own core we also have customer-users in the router, these are not allowed to make changes to the logical-system where SUNET’s configuration resides but they are however allowed to change the configuration on their user so that they are allowed to. This is protect against fat-fingering the router and not to protect against all evil in this world. We have not found a way to isolate users in logical slices of the system completely so this is the best we can do. We however trust our customers fully so we expect no problems here.

We run SUNET (with MPLS, LDP, ISIS iBGP etc) in a logical system and let the university get the main-instance. This is because that there is a few features that the universities might need (MCLAG for example) that do not work in a logical instance. SUNET only needs very basic things to that is why we have this the ”wrong way”

So lets look at a logical system...

```
logical-systems {
    SUNET {
        interfaces {
            apply-groups ETH;
            et-0/2/0 {
                unit 1653 {
                    description "Link 380 to fre-r1, su-r1.fre-r1";
                    vlan-id 1653;
                    family inet {
                        address 130.242.4.249/31;
                    }
                    family iso;
                    family inet6 {
                        address 2001:6b0:1e:1::1f9/127;
                    }
                    family mpls;
                }
            }
            ae0 {
                unit 1653 {
                    description "Link 381 to su-r2, su-r1.su-r2-l3";
                    vlan-id 1653;
                    family inet {
                        address 130.242.4.250/31;
                    }
                    family iso;
                    family inet6 {
                        address 2001:6b0:1e:1::1fa/127;
                    }
                    family mpls;
```

```
        }
    }
    lo0 {
        unit 1653 {
            description "SUNET management loopback";
            family inet {
                filter {
                    input re-protect-v4;
                }
                address 130.242.1.97/32 {
                    preferred;
                }
                address 127.0.0.1/32;
            }
            family iso {
                address 47.0023.0000.0001.0000.1302.4200.1097.00;
            }
            family inet6 {
                filter {
                    input re-protect-v6;
                }
                address 2001:6b0:1e::161/128;
            }
            family mpls;
        }
    }
}
protocols {
    bgp {
        precision-timers;
        advertise-inactive;
        log-updown;
        bgp-error-tolerance;
        group SUNET-RR {
            type internal;
            local-address 130.242.1.97;
            family inet {
                any;
            }
            authentication-key "KEY"; ## SECRET-DATA
```

```
            neighbor 130.242.1.27 {
                description fre-r1;
            }
            neighbor 130.242.1.28 {
                description tug-r1;
            }
        }
        group SUNET-RR-v6 {
            type internal;
            local-address 2001:6b0:1e::161;
            family inet6 {
                any;
            }
            authentication-key "KEY"; ## SECRET-DATA
            neighbor 2001:6b0:1e::11b {
                description fre-r1;
            }
            neighbor 2001:6b0:1e::11c {
                description tug-r1;
            }
        }
    }
    isis {
        apply-groups ISIS_AUTH;
        export [ redist-static-ncs redist-connected-ncs ];
        rib-group {
            inet mc-pseudo-rib;
            inet6 mcv6-pseudo-rib;
        }
        level 1 disable;
        interface et-0/2/0.1653 {
            point-to-point;
            level 2 metric 200;
        }
        interface ae0.1653 {
            point-to-point;
            level 2 metric 100;
        }
        interface lo0.1653 {
            passive;
```

```
                }
            }
        }
        policy-options {
            prefix-list bgp-auto {
                apply-path "logical-systems SUNET protocols bgp group <*>
neighbor <*>";
            }
            prefix-list bgp-auto-v6 {
                apply-path "logical-systems SUNET protocols bgp group <*>
neighbor <*:*>";
            }
            prefix-list ldp-speakers {
                130.242.1.0/24;
                130.242.4.0/23;
                130.242.80.0/24;
                130.242.82.0/24;
                130.242.83.0/24;
                130.242.84.0/24;
                130.242.85.0/24;
                193.10.255.0/24;
            }
            prefix-list dns-servers {
                apply-path "system name-server <*>";
            }
            prefix-list ntp-servers {
                apply-path "system ntp server <*>";
            }
            prefix-list ntp-control {
                apply-path "logical-systems SUNET interfaces lo0 unit <*>
family inet address <*>";
            }
            prefix-list msdp-auto {
                apply-path "logical-systems SUNET protocols msdp group <*> peer
<*>";
            }
            prefix-list msdp-group-peers {
                apply-path "logical-systems SUNET protocols msdp group <*> peer
<*>";
            }
```

```
prefix-list snmp-clients {
    apply-path "snmp community <*> clients <*>";
}
prefix-list bfd-clients {
    127.0.0.1/32;
}
prefix-list ssh-clients {
    MGMT-NETWORK/24;
}
prefix-list ssh-clients-v6 {
    MGMT-NETWORK-V6/64;
}
prefix-list bfd-clients-v6 {
    ::/128;
    2001:6b0:1e::/48;
}
policy-statement redist-connected-ncs {
    term reject-fxp0 {
        from {
            protocol direct;
            interface fxp0.0;
        }
        then reject;
    }
    term redist-rest {
        from protocol direct;
        then accept;
    }
}
policy-statement redist-static-ncs {
    term no-ipv4-default {
        from {
            protocol static;
            route-filter 0.0.0.0/0 exact;
        }
        then reject;
    }
    term no-ipv6-default {
        from {
            protocol static;
```

```
            route-filter ::/0 exact;
        }
        then reject;
    }
    term no-fxp {
        from interface fxp0.0;
        then reject;
    }
    term redist-rest {
        from protocol static;
        then accept;
    }
}
community MDVPN-Geant-vpns members 1653:65201;
community MDVPN-PE members 1653:2003;
community MDVPN-vpns members 1653:65200;
community NDGF-BACKUP members 1653:101;
community NDN-blackhole members 2603:999;
community NDN-blackhole-non_nordic members 2603:998;
community NORDUNET-CORE members 2603:111;
community SUNET members 1653:0;
community SUNET-ACNET members 1653:25176;
community SUNET-ALLTELE members 1653:44581;
community SUNET-BAHNHOF members 1653:8473;
community SUNET-BORDERLIGHT members 1653:16253;
community SUNET-BPF members 1653:64646;
community SUNET-BTH members 1653:8748;
community SUNET-COMHEM members 1653:39651;
community SUNET-COMX members 1653:31661;
community SUNET-CORE members 1653:0;
community SUNET-CSB members 1653:48514;
community SUNET-CTH members 1653:2841;
community SUNET-DGC members 1653:21195;
community SUNET-DGIX members 1653:2;
community SUNET-DU members 1653:9088;
community SUNET-FHS members 1653:64561;
community SUNET-FOI members 1653:65510;
community SUNET-FTP members 1653:15980;
community SUNET-GSIX members 1653:2840;
community SUNET-GU members 1653:2842;
```

```
community SUNET-HB members 1653:64525;
community SUNET-HGO members 1653:64531;
community SUNET-HH members 1653:64514;
community SUNET-HHS members 1653:64538;
community SUNET-HIG members 1653:16251;
community SUNET-HIS members 1653:64522;
community SUNET-HJ members 1653:64516;
community SUNET-HKR members 1653:64518;
community SUNET-HSV members 1653:64546;
community SUNET-HV members 1653:64530;
community SUNET-IAAS members 1653:41001;
community SUNET-IPO members 1653:12552;
community SUNET-IRF members 1653:25072;
community SUNET-JTH members 1653:64549;
community SUNET-KAU members 1653:64533;
community SUNET-KB members 1653:64557;
community SUNET-KI members 1653:2837;
community SUNET-KMH members 1653:65415;
community SUNET-KONSTFACK members 1653:64548;
community SUNET-KTH members 1653:2839;
community SUNET-KTHNOC members 1653:3224;
community SUNET-KTHNOC-KTH members 1653:3224;
community SUNET-LABS2 members 1653:29518;
community SUNET-LIU members 1653:2843;
community SUNET-LNU members 1653:64532;
community SUNET-LTU members 1653:2831;
community SUNET-LU members 1653:2846;
community SUNET-MAH members 1653:64520;
community SUNET-MDFNET members 1653:65419;
community SUNET-MDH members 1653:64521;
community SUNET-MDHNETCENTER members 1653:64549;
community SUNET-MIUN members 1653:64519;
community SUNET-NDGF members 1653:65420;
community SUNET-NETATONCE members 1653:35706;
community SUNET-NETNOD members 1653:8674;
community SUNET-NOBEL members 1653:64540;
community SUNET-NORRNOD members 1653:12501;
community SUNET-ONLY members 1653:2;
community SUNET-ORU members 1653:64513;
community SUNET-OWNIT members 1653:33885;
```

```
community SUNET-PDC members 1653:5601;
community SUNET-PDC-KTH members 1653:5601;
community SUNET-PERSPEKTIV members 1653:15782;
community SUNET-PORT80 members 1653:16150;
community SUNET-RIKSARKIVET members 1653:64560;
community SUNET-RIKSDAGEN members 1653:64559;
community SUNET-RIKSNET members 1653:34610;
community SUNET-SH members 1653:64523;
community SUNET-SKYCOM members 1653:29518;
community SUNET-SLU members 1653:12384;
community SUNET-SLU-UMEA members 1653:64555;
community SUNET-SMHI members 1653:42307;
community SUNET-SP members 1653:64563;
community SUNET-SR members 1653:47708;
community SUNET-STDH members 1653:64558;
community SUNET-STERIKKOM members 1653:64562;
community SUNET-STUD-LTU members 1653:64515;
community SUNET-STUPI members 1653:1880;
community SUNET-STUPI2 members 1653:1883;
community SUNET-SU members 1653:2838;
community SUNET-SU2 members 1653:64539;
community SUNET-SYSTEAM members 1653:25473;
community SUNET-SYSTEAM-UDAC members 1653:2865;
community SUNET-T3 members 1653:28908;
community SUNET-TELENOR members 1653:2119;
community SUNET-UHR members 1653:64565;
community SUNET-UMU members 1653:2833;
community SUNET-UPUNET-S members 1653:43844;
community SUNET-UU members 1653:2834;
community SUNET-VAL members 1653:64544;
community SUNET-VIAEUROPA members 1653:47155;
community SUNET-VINNOVA members 1653:43018;
community SUNET-VR members 1653:64554;
community SUNET-blackhole members 1653:999;
community SUNET-junet members 1653:59702;
community SUNET-maritima members 1653:64568;
community SUNET-oslab members 1653:2832;
community SUNET-upstream-blackhole members 1653:998;
community SUNET-upstream-part-blackhole members 1653:997;
community mdvpn-ping members target:8501:12321;
```

```
    community mdvpn-uu_epouta members target:1741:3000;
    community mdvpn-xifi members target:2200:1;
    community no-export members no-export;
    community sunetc-dcn members target:1653:434300002;
    community to-geant-vr members 20965:65200;
}
routing-options {
    protect core;
    rib-groups {
        mcv6-pseudo-rib {
            import-rib [ inet6.0 inet6.2 ];
        }
        mc-pseudo-rib {
            import-rib [ inet.0 inet.2 ];
        }
    }
    autonomous-system 1653;
}
firewall {
    family inet {
        filter re-protect-v4 {
            term allow-em0 {
                from {
                    interface em0;
                }
                then accept;
            }
            term except-tcp-rate {
                from {
                    source-prefix-list {
                        ssh-clients;
                    }
                    protocol tcp;
                    tcp-flags " (syn&!ack)|fin|rst ";
                }
                then {
                    count except-tcp-syn-rate;
                    accept;
                }
            }
```

```
                    term tcp-syn-rate {
                        from {
                            source-prefix-list {
                                bgp-auto;
                            }
                            protocol tcp;
                            tcp-flags " (syn&!ack)|fin|rst ";
                        }
                        then {
                            policer tcp-syn-policer;
                            count tcp-syn-rate;
                            accept;
                        }
                    }
                    term icmp-rate {
                        from {
                            protocol icmp;
                            icmp-type [ echo-request echo-reply unreachable
time-exceeded timestamp timestamp-reply ];
                        }
                        then {
                            policer icmp-policer;
                            count icmp-rate;
                            accept;
                        }
                    }
                    term ignore-tcp-chatter {
                        from {
                            protocol tcp;
                            destination-port [ 80 135-139 445 1433 1434 ];
                        }
                        then {
                            count chatter-tcp;
                            discard;
                        }
                    }
                    term ssh-authorized {
                        from {
                            source-prefix-list {
                                ssh-clients;
```

```
        }
        protocol tcp;
        port ssh;
    }
    then {
        count ssh-auth;
        accept;
    }
}
term netconf {
    from {
        source-prefix-list {
            ssh-clients;
        }
        destination-port 830;
    }
    then {
        count netconf;
        accept;
    }
}
term lsp-ping {
    from {
        source-prefix-list {
            ldp-speakers;
        }
        protocol udp;
        port 3503;
    }
    then accept;
}
term ldp {
    from {
        source-prefix-list {
            ldp-speakers;
        }
        protocol [ udp tcp ];
        port ldp;
    }
    then accept;
```

```
}
term bgp {
    from {
        source-prefix-list {
            bgp-auto;
        }
        protocol tcp;
        port bgp;
    }
    then {
        count bgp;
        accept;
    }
}
term dns {
    from {
        source-prefix-list {
            dns-servers;
        }
        protocol udp;
        port domain;
    }
    then {
        count dns;
        accept;
    }
}
term ntp {
    from {
        source-prefix-list {
            ntp-servers;
        }
        protocol udp;
        port ntp;
    }
    then {
        count ntp;
        accept;
    }
}
```

```
term ntp-control {
    from {
        source-prefix-list {
            ntp-control;
        }
        protocol udp;
        port ntp;
    }
    then {
        count ntp;
        accept;
    }
}
term ignore-udp-chatter {
    from {
        protocol udp;
        port [ 1433-1434 135-139 ];
    }
    then {
        count chatter-udp;
        discard;
    }
}
term traceroute-rate {
    from {
        protocol udp;
        destination-port 33434-33650;
    }
    then {
        policer traceroute-policer;
        count traceroute;
        accept;
    }
}
term mcast {
    from {
        protocol [ pim igmp ];
    }
    then {
        count mcast;
```

```
            accept;
        }
    }
    term msdp-peers {
        from {
            source-prefix-list {
                msdp-auto;
            }
            protocol tcp;
            port msdp;
        }
        then {
            count msdp;
            accept;
        }
    }
    term msdp-group-peers {
        from {
            source-prefix-list {
                msdp-group-peers;
            }
            protocol tcp;
            port msdp;
        }
        then {
            count msdp;
            accept;
        }
    }
    term snmp {
        from {
            source-prefix-list {
                snmp-clients;
            }
            protocol udp;
            destination-port snmp;
        }
        then {
            count snmp;
            accept;
```

```
            }
        }
        term vrrp {
            from {
                destination-address {
                    224.0.0.18/32;
                }
                protocol [ vrrp 51 ];
            }
            then accept;
        }
        term bfd {
            from {
                prefix-list {
                    bfd-clients;
                }
                protocol udp;
                port 3784;
            }
            then accept;
        }
        term last {
            then {
                count drop;
                log;
                discard;
            }
        }
    }
}
family inet6 {
    filter re-protect-v6 {
        term icmp-rate {
            from {
                next-header icmpv6;
            }
            then {
                policer icmp-policer;
                count icmp-rate-v6;
                accept;
```

```
    }
}
term traceroute-rate {
    from {
        next-header udp;
        destination-port 33434-33650;
    }
    then {
        policer traceroute-policer;
        count traceroute-v6;
        accept;
    }
}
term ssh-authorized {
    from {
        source-prefix-list {
            ssh-clients-v6;
        }
        next-header tcp;
        port ssh;
    }
    then {
        count ssh-auth-v6;
        accept;
    }
}
term bgp {
    from {
        source-prefix-list {
            bgp-auto-v6;
        }
        next-header tcp;
        port bgp;
    }
    then {
        count bgp-v6;
        accept;
    }
}
term mcast {
```

```
            from {
                next-header [ pim igmp ];
            }
            then {
                count mcast-v6;
                accept;
            }
        }
        term mcast-scoped {
            from {
                destination-address {
                    ff00::/13;
                }
            }
            then {
                count mcast-v6-scoped;
                accept;
            }
        }
        term bfd {
            from {
                prefix-list {
                    bfd-clients-v6;
                }
                next-header udp;
                port 3784;
            }
        }
        term last {
            then {
                count drop-v6;
                discard;
            }
        }
    }
}
policer tcp-syn-policer {
    if-exceeding {
        bandwidth-limit 500k;
        burst-size-limit 15k;
```

```
                }
                then discard;
            }
            policer icmp-policer {
                if-exceeding {
                    bandwidth-limit 5m;
                    burst-size-limit 100k;
                }
                then discard;
            }
            policer traceroute-policer {
                if-exceeding {
                    bandwidth-limit 2m;
                    burst-size-limit 50k;
                }
                then discard;
            }
        }
    }
}
```

This is the SUNET-part of the router. We have firewalls in place to protect the control-plane for obvious reasons. ISIS and BGP should be somewhat self-explanatory.

**Q:** ”Why dont you peer with the main-instance from this logical system using a logical tunnel?”

**A:** If you use any function in the router that forces traffic to enter the backplane (such as sampling) you will have the traffic pass the forwarding-chip twice when using LT’s. In our case where we use MPC3-NG we are limited to a bandwidth of 130Gbps and since traffic will be duplicated essentially the real throughput is 65Gbps if doing a double-pass. If we build a 100G network we want it to actually be 100G so that is why the main-instance (the universities) is actually peering with core instead of its own logical slice of SUNET.

**Q:**”What is the SUNET LSYS even doing other then managing the device?”

**A:**In almost all installations we have there is ”other stuff” then the university at the university, which is treated as a separate connection. Example of this is museums, campus-shops, academic institutions, govermental facilities etc. These are customers to SUNET and not to the university (albeit being connected through the university). These will sit in the SUNET LSYS. Another service that we will see more and more off is the like of Azure Expressroute (or any other off-prem cloud service) that will typically terminate in a MPLS-tunnel and we do run MPLS to the LSYS so we can easily terminate this in every router that needs it.

Alright, onwards.

```
chassis {
    redundancy {
        routing-engine 0 master;
        routing-engine 1 backup;
        failover {
```

```
            on-loss-of-keepalives;
            on-disk-failure;
        }
        graceful-switchover;
    }
    aggregated-devices {
        ethernet {
            device-count 2;
        }
    }
    alarm {
        management-ethernet {
            link-down ignore;
        }
    }
    network-services enhanced-ip;
}
interfaces {
    apply-groups ETH;
    xe-0/0/0 {
        description "Link to su-r2, su-r1.su-r2-phy1";
        gigether-options {
            802.3ad ae0;
        }
    }
    xe-0/0/1 {
        description "Link to su-r2, su-r1.su-r2-phy2";
        gigether-options {
            802.3ad ae0;
        }
    }
    xe-0/0/2 {
        description "Link to su-r2, su-r1.su-r2-phy3";
        gigether-options {
            802.3ad ae0;
        }
    }
    xe-0/0/8 {
        description "SU link to Core S-Hall phy2";
        gigether-options {
```

```
        802.3ad ae1;
    }
}
xe-0/0/9 {
    description "SU link to Core S-Hall phy1";
    gigether-options {
        802.3ad ae1;
    }
}
et-0/2/0 {
    description "Physical link to fre-r1";
    vlan-tagging;
    unit 10 {
        description "EBGP Peering to FRE";
        vlan-id 10;
        family inet {
            filter {
                input fw_ipv4_SU_input;
                output fw_ipv4_SU_output;
            }
            address 130.242.6.141/31;
        }
        family inet6 {
            filter {
                input fw_ipv6_SU_input;
                output fw_ipv6_SU_output;
            }
            address 2001:6b0:1e:2::18d/127;
        }
    }
}
ae0 {
    description "Link to su-r2, su-r1.su-r2";
    vlan-tagging;
    aggregated-ether-options {
        load-balance {
            adaptive;
        }
        minimum-links 1;
        link-speed 10g;
```

```
        lacp {
            active;
        }
    }
    unit 2838 {
        description "SU internal link to su-r2";
        vlan-id 2838;
        family inet {
            address 130.237.154.169/30;
        }
        family inet6 {
            address 2001:6b0:5:37::169/64;
        }
    }
}
ae1 {
    description "SU link to Core S-Hall";
    aggregated-ether-options {
        minimum-links 1;
        link-speed 10g;
        lacp {
            active;
        }
    }
    unit 0 {
        family inet {
            mtu 9000;
            address 130.237.154.17/30;
        }
        family inet6 {
            mtu 9000;
            address 2001:6b0:5:35::17/64;
        }
    }
}
lo0 {
    unit 0 {
        description "SU internal su-r1 loopback";
        family inet {
            filter {
```

```
                    input re-protect-v4;
                }
                address 130.237.154.145/32;
            }
            family inet6 {
                filter {
                    input re-protect-v6;
                }
                address 2001:6b0:5:1::145/128;
            }
        }
    }
}
```

This is the interface configuration of SU slices of the system. As you can see they are also using the same interfaces as we do in the logical-system (ae0, et-0/2/0) but on other units.

```
snmp {
    contact "noc@sunet.se";
    community public {
        authorization read-only;
        clients {
            $SNMP-SERVER1;
            $SNMP-SERVER2;
            $SNMP-SERVER3;
        }
    }
}
forwarding-options {
    hyper-mode;
}
routing-options {
    nonstop-routing;
    interface-routes {
        rib-group {
            inet mc-pseudo-rib;
            inet6 mcv6-pseudo-rib;
        }
    }
    rib inet.0 {
```

```
aggregate {
    route 130.237.188.0/26;
    route 130.237.85.0/24;
    route 130.237.86.0/23;
    route 130.237.88.0/21;
    route 130.237.144.0/22;
    route 130.237.148.0/23;
    route 130.237.151.0/24;
    route 130.237.152.0/22;
    route 130.237.160.0/24;
    route 130.237.162.0/23;
    route 130.237.164.0/22;
    route 130.237.168.0/21;
    route 130.237.176.0/24;
    route 130.237.178.0/23;
    route 130.237.184.0/21;
    route 130.237.192.0/21;
    route 130.237.205.0/24;
    route 130.237.208.0/24;
    route 130.237.217.0/24;
    route 130.237.253.0/24;
    route 130.237.254.0/24;
    route 130.242.128.0/24;
    route 193.10.6.0/24;
    route 193.10.145.0/24;
    route 193.10.147.0/24;
    route 193.11.25.0/24;
    route 193.11.26.0/23;
    route 193.11.28.0/23;
    route 193.11.30.0/24;
    route 193.11.31.128/26;
    route 77.238.32.0/21;
    route 130.237.248.0/24;
    route 193.11.92.0/23;
    route 193.11.144.0/24;
    route 130.237.240.0/24;
    route 130.237.241.0/24;
    route 130.237.242.0/24;
    route 130.237.243.0/24;
    route 130.237.244.0/24;
```

```
        route 130.237.245.0/24;
        route 130.237.246.0/24;
        route 130.237.247.0/24;
        route 130.237.200.0/24;
        route 130.237.156.0/24;
        route 77.238.48.0/21;
        route 193.11.94.0/23;
        route 130.237.180.0/24;
        route 130.237.182.0/23;
        route 130.242.47.0/24;
        route 130.237.158.0/24;
        route 130.237.177.0/24;
        route 193.10.10.0/24;
        route 192.36.127.0/24;
        route 130.237.159.0/24;
        route 193.10.9.0/24;
        route 193.10.8.0/24;
        route 130.237.150.0/24;
        route 130.237.161.0/24;
        route 130.237.157.0/24;
        route 130.237.181.0/24;
    }
}
rib inet6.0 {
    static {
        route 2001:6b0:5::/48 reject;
    }
}
martians {
    128.0.0.0/16 orlonger allow;
    192.255.0.0/16 orlonger allow;
    223.255.255.0/24 exact allow;
}
rib-groups {
    mc-pseudo-rib {
        import-rib [ inet.0 inet.2 ];
    }
    mcv6-pseudo-rib {
        import-rib [ inet6.0 inet6.2 ];
    }
```

```
        mcast {
            export-rib inet.2;
            import-rib inet.2;
        }
        mcast6 {
            export-rib inet6.2;
            import-rib inet6.2;
        }
    }
    autonomous-system 2838;
    forwarding-table {
        export load_balance;
    }
}
```

In SU's case we generate routes through the aggregate-statement. This is becuase…. long story, well go through that another day, this post is already quite long.

```
protocols {
    bgp {
        group SUNET {
            type external;
            family inet {
                any;
            }
            neighbor 130.242.6.140 {
                description fre-r1;
                import sunet-in;
                export [ SU_prefix_v4 SU_redist_static reject-all ];
                peer-as 1653;
            }
        }
        group SUNET-v6 {
            type external;
            family inet6 {
                any;
            }
            neighbor 2001:6b0:1e:2::18c {
                description fre-r1;
                import sunet-in-v6;
```

```
            export [ SU_prefix_v6 reject-all ];
            peer-as 1653;
        }
    }
    group SU {
        type internal;
        local-address 130.237.154.145;
        family inet {
            any;
        }
        peer-as 2838;
        neighbor 130.237.154.146;
    }
    group SU-v6 {
        type internal;
        local-address 2001:6b0:5:1::145;
        family inet6 {
            any;
        }
        peer-as 2838;
        neighbor 2001:6b0:5:1::146;
    }
}
```

Nothing special here, One eBGP and one iBGP group for each AF.

```
ospf {
    export [ ospf-default SU_redist_static ];
    area 0.0.0.0 {
        interface lo0.0 {
            passive;
        }
        interface ae0.2838;
        interface et-0/2/0.10 {
            passive;
        }
        interface ae1.0 {
            interface-type p2p;
            authentication {
```

```
                    md5 1 key "KEY"; ## SECRET-DATA
                }
            }
        }
    }
    ospf3 {
        traceoptions {
            file debug-ospf size 1m files 5;
            flag all;
        }
        export ospf3-default;
        area 0.0.0.0 {
            interface ae0.2838;
            interface et-0/2/0.10 {
                passive;
            }
            interface ae1.0 {
                interface-type p2p;
            }
            interface lo0.0 {
                passive;
            }
        }
    }
}
```

OSPF to make the router part of SUs internal OSPF-topology. In SUs case the MX-routers runs OSPF with their Cisco Nexus campus-core.

```
policy-options {
    prefix-list bgp-auto {
        apply-path "protocols bgp group <*> neighbor <*>";
    }
    prefix-list bgp-auto-v6 {
        apply-path "protocols bgp group <*> neighbor <*:*>";
    }
    prefix-list dns-servers {
        apply-path "system name-server <*>";
    }
    prefix-list ntp-servers {
        apply-path "system ntp server <*>";
```

```
    }
    prefix-list ntp-control {
        apply-path "interfaces lo0 unit <*> family inet address <*>";
    }
    prefix-list msdp-auto {
        apply-path "protocols msdp group <*> peer <*>";
    }
    prefix-list msdp-group-peers {
        apply-path "protocols msdp group <*> peer <*>";
    }
    prefix-list snmp-clients {
        apply-path "snmp community <*> clients <*>";
    }
    prefix-list bfd-clients {
        127.0.0.1/32;
    }
    prefix-list bgp-auto-ls {
        apply-path "logical-systems SUNET protocols bgp group <*> neighbor
<*>";
    }
    prefix-list bgp-auto-ls-v6 {
        apply-path "logical-systems SUNET protocols bgp group <*> neighbor
<*:*>";
    }
    prefix-list ssh-clients-v6 {
        $SU-IPV6-MGMT-NETWORK/64;
    }
    prefix-list bfd-clients-v6 {
        ::/128;
        2001:6b0:1e::/48;
    }
    prefix-list ssh-clients {
        $SU-IPV4-MGMT-NETWORK/24;

    policy-statement SU_prefix_v4 {
        term 1 {
            from {
                route-filter 130.237.85.0/24 exact;
                route-filter 130.237.86.0/23 exact;
                route-filter 130.237.88.0/21 exact;
```

```
route-filter 130.237.144.0/22 exact;
route-filter 130.237.148.0/23 exact;
route-filter 130.237.151.0/24 exact;
route-filter 130.237.152.0/22 exact;
route-filter 130.237.160.0/24 exact;
route-filter 130.237.162.0/23 exact;
route-filter 130.237.164.0/22 exact;
route-filter 130.237.168.0/21 exact;
route-filter 130.237.176.0/24 exact;
route-filter 130.237.178.0/23 exact;
route-filter 130.237.184.0/21 exact;
route-filter 130.237.192.0/21 exact;
route-filter 130.237.205.0/24 exact;
route-filter 130.237.208.0/24 exact;
route-filter 130.237.217.0/24 exact;
route-filter 130.237.253.0/24 exact;
route-filter 130.237.254.0/24 exact;
route-filter 130.242.128.0/24 exact;
route-filter 193.10.6.0/24 exact;
route-filter 193.10.145.0/24 exact;
route-filter 193.10.147.0/24 exact;
route-filter 193.11.25.0/24 exact;
route-filter 193.11.26.0/23 exact;
route-filter 193.11.28.0/23 exact;
route-filter 193.11.30.0/24 exact;
route-filter 193.11.31.128/26 exact;
route-filter 77.238.32.0/21 exact;
route-filter 130.237.248.0/24 exact;
route-filter 193.11.92.0/23 exact;
route-filter 193.11.144.0/24 exact;
route-filter 130.237.240.0/24 exact;
route-filter 130.237.241.0/24 exact;
route-filter 130.237.242.0/24 exact;
route-filter 130.237.243.0/24 exact;
route-filter 130.237.244.0/24 exact;
route-filter 130.237.245.0/24 exact;
route-filter 130.237.246.0/24 exact;
route-filter 130.237.247.0/24 exact;
route-filter 130.237.200.0/24 exact;
route-filter 130.237.156.0/24 exact;
```

```
            route-filter 77.238.48.0/21 exact;
            route-filter 193.11.94.0/23 exact;
            route-filter 130.237.180.0/24 exact;
            route-filter 130.237.182.0/23 exact;
            route-filter 130.242.47.0/24 exact;
            route-filter 130.237.240.0/23 orlonger;
            route-filter 130.237.158.0/24 exact;
            route-filter 130.237.177.0/24 exact;
            route-filter 193.10.10.0/24 exact;
            route-filter 192.36.127.0/24 exact;
            route-filter 130.237.159.0/24 exact;
            route-filter 193.10.9.0/24 exact;
            route-filter 193.10.8.0/24 exact;
            route-filter 130.237.150.0/24 exact;
            route-filter 130.237.161.0/24 exact;
            route-filter 130.237.157.0/24 exact;
            route-filter 130.237.181.0/24 exact;
            route-filter 130.237.188.0/26 exact;
        }
        then {
            metric 50;
            accept;
        }
    }
}
policy-statement SU_prefix_v6 {
    term 1 {
        from {
            route-filter 2001:6b0:5::/48 exact;
            route-filter 2001:6b0:5:198::/64 exact;
            route-filter 2001:6b0:5:199::/64 exact;
        }
        then {
            metric 30;
            accept;
        }
    }
}
policy-statement SU_redist_static {
    term 1 {
```

```
        from protocol static;
        then accept;
    }
}
policy-statement load_balance {
    term 1 {
        then {
            load-balance per-packet;
        }
    }
}
policy-statement ospf-default {
    term import-bgp {
        from {
            protocol bgp;
            tag 1653;
            route-filter 0.0.0.0/0 exact;
        }
        then {
            metric 6;
            accept;
        }
    }
    term reject {
        then reject;
    }
}
policy-statement ospf3-default {
    term import-bgp {
        from {
            protocol bgp;
            tag 1653;
            route-filter ::/0 exact;
        }
        then {
            metric 150;
            accept;
        }
    }
    term reject {
```

```
            then reject;
        }
    }
    policy-statement reject-all {
        then reject;
    }
    policy-statement sunet-in {
        term tag-default-v4 {
            from {
                route-filter 0.0.0.0/0 exact;
            }
            then {
                tag add 1653;
                preference 90;
                accept;
            }
        }
        term accept {
            then accept;
        }
    }
    policy-statement sunet-in-v6 {
        term tag-default-v6 {
            from {
                route-filter ::/0 exact;
            }
            then {
                tag add 1653;
                preference 90;
                accept;
            }
        }
        term accept {
            then accept;
        }
    }
}
```

Some policy-statements to generate and accept default-routes and to tag them correctly.

```
firewall {
    family inet {
        filter fw_ipv4_SU_input {
            term ingress {
                from {
                    source-address {
                        130.237.85.0/24;
                        130.237.86.0/23;
                        130.237.88.0/21;
                        130.237.144.0/22;
                        130.237.148.0/23;
                        130.237.151.0/24;
                        130.237.152.0/22;
                        130.237.156.0/24;
                        130.237.160.0/24;
                        130.237.162.0/23;
                        130.237.164.0/22;
                        130.237.168.0/21;
                        130.237.176.0/24;
                        130.237.178.0/23;
                        130.237.184.0/21;
                        130.237.192.0/21;
                        130.237.205.0/24;
                        130.237.208.0/24;
                        130.237.217.0/24;
                        130.237.253.0/24;
                        130.237.254.0/24;
                        130.242.128.0/24;
                        193.10.6.0/24;
                        193.10.145.0/24;
                        193.10.147.0/24;
                        193.11.25.0/24;
                        193.11.26.0/23;
                        193.11.28.0/23;
                        193.11.30.0/24;
                        193.11.31.128/26;
                        193.11.94.0/23;
                        130.237.240.0/21;
                        130.237.248.0/24;
                        193.11.92.0/23;
```

```
                193.11.144.0/23;
                130.237.200.0/24;
                77.238.48.0/21;
                130.237.180.0/24;
                130.237.182.0/24;
                130.237.183.0/24;
                130.242.47.0/24;
                130.237.158.0/24;
                130.237.177.0/24;
                193.10.10.0/24;
                130.237.159.0/24;
                192.36.127.0/24;
                193.10.9.0/24;
                193.10.8.0/24;
                130.237.150.0/24;
                130.237.161.0/24;
                130.237.157.0/24;
                130.237.181.0/24;
            }
        }
        then {
            count ingress;
            discard;
        }
    }
    term martians {
        from {
            address {
                0.0.0.0/32;
                127.0.0.0/8;
                172.16.0.0/12;
                10.0.0.0/8;
                192.168.0.0/16;
                192.0.2.0/24;
                169.254.0.0/16;
                240.0.0.0/4;
            }
        }
        then {
            count martians;
```

```
            syslog;
            discard;
        }
    }
    term su-extern {
        from {
            source-address {
                $EXTERNAL-IP/24;
            }
            protocol [ tcp udp ];
        }
        then {
            count su-extern;
            accept;
        }
    }
    term block_afs {
        from {
            destination-address {
                $NO-AFS-PFX/24;
            }
            protocol udp;
            destination-port [ 4711 7000-7009 ];
        }
        then {
            count block_AFS;
            discard;
        }
    }
    term block-host {
        from {
            source-address {
                $ANNOYING-DDOSKIDDIE1/32;
                $ANNOYING-DDOSKIDDIE2/32;
                $ANNOYING-DDOSKIDDIE3/32;
                $ANNOYING-DDOSKIDDIE4/32;
                $ANNOYING-DDOSKIDDIE5/32;
                $ANNOYING-DDOSKIDDIE6/32;
                $ANNOYING-DDOSKIDDIE7/32;
                $ANNOYING-DDOSKIDDIE8/32;
```

```
            $ANNOYING-DDOSKIDDIE9/32;
            $ANNOYING-DDOSKIDDIE10/32;
            $ANNOYING-DDOSKIDDIE11/32;
            $ANNOYING-DDOSKIDDIE12/32;
            $ANNOYING-DDOSKIDDIE13/32;
            $ANNOYING-DDOSKIDDIE14/32;
            $ANNOYING-DDOSKIDDIE15/32;
            $ANNOYING-DDOSKIDDIE16/32;
        }
    }
    then {
        count block-host;
        syslog;
        discard;
    }
}
term permit-router {
    from {
        destination-address {
            $BGP-SPEAKER/32;
        }
        protocol tcp;
        port bgp;
    }
    then {
        count permit-router;
        accept;
    }
}
term established {
    from {
        protocol tcp;
        tcp-established;
    }
    then {
        count established;
        accept;
    }
}
term permit-smtp {
```

```
        from {
            destination-address {
                $SMTP-SERVER1/32;
            }
            protocol tcp;
            destination-port [ smtp 587 ];
        }
        then {
            count permit-smtp;
            accept;
        }
    }
    term permit-smtp-canit {
        from {
            source-address {
                $SPAMFILTER-SRC/29;
            }
            destination-address {
                $SPAMFILTER-DST/32;

            }
            protocol tcp;
            destination-port smtp;
        }
        then {
            count permit-smtp-canit;
            accept;
        }
    }
    term permit-submission-dsv {
        from {
            destination-address {
                $SUBMISSION-DEST/32;
            }
            protocol tcp;
            destination-port 587;
        }
        then {
            count permit-submission-dsv;
            accept;
```

```
            }
        }
        term icmp {
            from {
                protocol icmp;
                icmp-type [ echo-reply unreachable source-quench echo-
request router-advertisement router-solicit time-exceeded parameter-problem ];
            }
            then {
                count icmp;
                accept;
            }
        }
        term ntp {
            from {
                destination-address {
                    $NTP-SERVER1/32;
                    $NTP-SERVER2/32;
                    $NTP-SERVER3/32;
                }
                protocol [ tcp udp ];
                port 123;
            }
            then {
                count ntp;
                accept;
            }
        }
        term syslog {
            from {
                source-address {
                    $SYSLOG-SRC1/29;
                    $SYSLOG-SRC2/32;
                }
                protocol [ tcp udp ];
                destination-port 514;
            }
            then {
                count syslog;
                accept;
```

```
    }
}
term printers {
    from {
        source-address {
            $PRINTERS1/16;
            $PRINTERS2/16;
            $PRINTERS3/16;
            $PRINTERS4/24;
            $PRINTERS5/32;
            $PRINTERS6/32;
        }
        protocol [ udp tcp ];
        port [ 515 ldp ];
    }
    then {
        count printers;
        syslog;
        accept;
    }
}
term nat-geo {
    from {
        destination-address {
            $NAT-GEO-DST/32;
        }
        protocol tcp;
        destination-port 1080;
    }
    then {
        count nat-geo;
        accept;
    }
}
term dsv-database {
    from {
        destination-address {
            $DSV-DB/32;
        }
        protocol tcp;
```

```
            port 3306;
        }
        then {
            count dsv-database;
            accept;
        }
    }
    term src-database {
        from {
            destination-address {
                $SRC-DB/32;
            }
            protocol tcp;
            port 3306;
        }
        then {
            count src-database;
            accept;
        }
    }
    term sprakstudion_filemaker {
        from {
            destination-address {
                $SPRAK-FLMK1/32;
                $SPRAK-FLMK2/32;
            }
            protocol tcp;
            port 5003;
        }
        then {
            count sprakstudion_filemaker;
            accept;
        }
    }
    term SUB {
        from {
            destination-address {
                $SUB-DST/26;
            }
            protocol [ tcp udp ];
```

```
            destination-port 1521;
        }
        then {
            count sub;
            syslog;
            accept;
        }
    }
    term Sunet-X {
        from {
            source-address {
                $SUNETX1/16;
                $SUNETX2/16;
                $SUNETX3/21;
                $SUNETX4/21;
            }
            protocol [ tcp udp ];
            destination-port 6000-6009;
        }
        then {
            count sunet-x;
            syslog;
            accept;
        }
    }
    term Outlook-X {
        from {
            destination-address {
                $OUTLOOK-X/32;
            }
            protocol tcp;
            destination-port [ 6001-6002 6004 ];
        }
        then {
            count Outlook-X;
            syslog;
            accept;
        }
    }
    term Albanova {
```

```
        from {
            destination-address {
                $ALBANOVA1/32;
                $ALBANOVA2/32;
                $ALBANOVA3/32;
                $ALBANOVA4/32;
            }
            protocol udp;
            port 5000-5003;
        }
        then {
            count albanova;
            accept;
        }
    }
    term apple-share {
        from {
            destination-address {
                $APPLE-FS1/32;
                $APPLE-FS2/32;
                $APPLE-FS3/32;
                $APPLE-FS4/32;
                $APPLE-FS5/32;
            }
            protocol [ udp tcp ];
            port [ 543 548 687 9100 ];
        }
        then {
            count apple-share;
            syslog;
            accept;
        }
    }
    term slaviska {
        from {
            destination-address {
                $SLAVISKA1/32;
            }
            protocol [ tcp udp ];
            destination-port 5003;
```

```
                }
                then {
                    count slaviska;
                    accept;
                }
            }
            term lingvistik {
                from {
                    destination-address {
                        $LINGVISTIK1/32;
                        $LINGVISTIK2/32;
                        $LINGVISTIK3/32;
                        $LINGVISTIK4/32;
                    }
                    protocol [ tcp udp ];
                    destination-port [ 5003 5000 ];
                }
                then {
                    count lingvistik;
                    accept;
                }
            }
            term protocol {
                from {
                    protocol [ 53 55 77 103 icmp ];
                }
                then {
                    count protocol;
                    discard;
                }
            }
            term port-udp {
                from {
                    protocol udp;
                    destination-port [ 7 9 11 19 42 67 68 69 79 123 135 136 137
138 139 161 162 177 427 445 514 515 517 518 524 587 593 1080 1433 1434 1900
2049 3372 5000 5003 5135 17 ];
                }
                then {
                    count port-udp;
```

```
                discard;
            }
        }
        term port-tcp {
            from {
                protocol tcp;
                destination-port [ 7 9 11 15 19 25 42 67 68 69 79 111 123
135 136 137 138 139 161 162 427 445 515 524 587 593 635 705 950 1080 1433 1434
1521 1900 2049 2301 2967 3020 3128 3306 3372 5000 5003 5232 6000-6009 6101 6106
6112 6129 7937 8888 10000 13701 13711 13720-13724 13782-13783 22370 41523 49400
514 17 32370-32399 32401-32787 ];
            }
            then {
                count port-tcp;
                discard;
            }
        }
        term SU-Net {
            from {
                destination-address {
                    130.237.85.0/24;
                    130.237.86.0/23;
                    130.237.88.0/21;
                    130.237.144.0/22;
                    130.237.148.0/23;
                    130.237.151.0/24;
                    130.237.152.0/22;
                    130.237.156.0/24;
                    130.237.160.0/24;
                    130.237.162.0/23;
                    130.237.164.0/22;
                    130.237.168.0/21;
                    130.237.176.0/24;
                    130.237.178.0/23;
                    130.237.184.0/21;
                    130.237.192.0/21;
                    130.237.205.0/24;
                    130.237.208.0/24;
                    130.237.217.0/24;
                    130.237.253.0/24;
```

```
                130.237.254.0/24;
                130.242.128.0/24;
                193.10.6.0/24;
                193.10.145.0/24;
                193.10.147.0/24;
                193.11.25.0/24;
                193.11.26.0/23;
                193.11.28.0/23;
                193.11.30.0/24;
                193.11.31.128/26;
                193.11.94.0/23;
                77.238.32.0/21;
                224.0.0.0/4;
                130.237.240.0/21;
                130.237.248.0/24;
                193.11.92.0/23;
                193.11.144.0/23;
                130.237.200.0/24;
                77.238.48.0/21;
                130.237.180.0/24;
                130.237.182.0/24;
                130.237.183.0/24;
                130.242.47.0/24;
                130.237.158.0/24;
                130.237.177.0/24;
                193.10.10.0/24;
                130.237.159.0/24;
                192.36.127.0/24;
                193.10.9.0/24;
                193.10.8.0/24;
                130.237.150.0/24;
                130.237.161.0/24;
                130.237.157.0/24;
                130.237.181.0/24;
            }
        }
        then accept;
    }
}
filter fw_ipv4_SU_output {
```

```
term egress {
    from {
        destination-address {
            130.237.85.0/24;
            130.237.86.0/23;
            130.237.88.0/21;
            130.237.144.0/22;
            130.237.148.0/23;
            130.237.151.0/24;
            130.237.152.0/22;
            130.237.156.0/24;
            130.237.160.0/24;
            130.237.162.0/23;
            130.237.164.0/22;
            130.237.168.0/21;
            130.237.176.0/24;
            130.237.178.0/23;
            130.237.184.0/21;
            130.237.192.0/21;
            130.237.205.0/24;
            130.237.208.0/24;
            130.237.217.0/24;
            130.237.253.0/24;
            130.237.254.0/24;
            130.242.128.0/24;
            193.10.6.0/24;
            193.11.25.0/24;
            193.11.26.0/23;
            193.11.28.0/23;
            193.11.30.0/24;
            193.11.31.128/26;
            193.11.94.0/23;
            77.238.32.0/21;
            193.10.145.0/24;
            193.10.147.0/24;
            130.237.240.0/21;
            130.237.248.0/24;
            193.11.92.0/23;
            193.11.144.0/23;
            130.237.200.0/24;
```

```
            77.238.48.0/21;
            130.237.180.0/24;
            130.237.182.0/24;
            130.237.183.0/24;
            130.242.47.0/24;
            130.237.158.0/24;
            130.237.177.0/24;
            192.36.127.0/24;
            130.237.159.0/24;
            193.10.10.0/24;
            193.10.9.0/24;
            193.10.8.0/24;
            130.237.150.0/24;
            130.237.161.0/24;
            130.237.157.0/24;
            130.237.181.0/24;
        }
    }
    then {
        count egress;
        discard;
    }
}
term permit-router {
    from {
        destination-address {
            $BGP-SPEAKERS1/24;
        }
        protocol tcp;
        port bgp;
    }
    then {
        count permit-router;
        accept;
    }
}
term martians {
    from {
        source-address {
            0.0.0.0/32;
```

```
            127.0.0.0/8;
            172.16.0.0/12;
            10.0.0.0/8;
            192.168.0.0/16;
            192.0.2.0/24;
            169.254.0.0/16;
            240.0.0.0/4;
        }
    }
    then {
        count martians;
        syslog;
        discard;
    }
}
term block-host-out {
    from {
        source-address {
            $SRC-BLOCK-OUT1/32;
            $SRC-BLOCK-OUT2/32;
        }
        destination-address {
            $DST-BLOCK-OUT/32;
            $DST-BLOCK-OUT/32;
            $DST-BLOCK-OUT/32;
            $DST-BLOCK-OUT/32;
            $DST-BLOCK-OUT/32;
        }
    }
    then {
        count block-host-out;
        syslog;
        discard;
    }
}
term block-spss-out {
    from {
        source-address {
            $BLOCK-SPSS/32;
        }
```

```
            protocol udp;
            source-port 5093;
        }
        then {
            count block-spss-out;
            discard;
        }
    }
    term permit-smtp {
        from {
            source-address {
                $SMTP-MAILER1/32;
                $SMTP-MAILER2/32;
                $SMTP-MAILER3/32;
                $SMTP-MAILER4/32;
                $SMTP-MAILER5/32;
            }
            protocol tcp;
            destination-port smtp;
        }
        then {
            count permit-smtp;
            accept;
        }
    }
    term permit-smtp-canit {
        from {
            source-address {
                $SMTP-MAILER1/32;
                $SMTP-MAILER2/32;
                $SMTP-MAILER3/32;
                $SMTP-MAILER4/32;
            }
            destination-address {
                $SUNET-MAILFILTER1/29;
            }
            protocol tcp;
            destination-port smtp;
        }
        then {
```

```
                count permit-smtp-canit;
                accept;
            }
        }
        term deny-ports-out {
            from {
                protocol [ tcp udp ];
                destination-port [ 69 smtp ];
            }
            then {
                count deny-ports-out;
                discard;
            }
        }
        term kth-alba-printer {
            from {
                source-address {
                    $PRINTER-SRC1/23;

                }
                destination-address {
                    $PRINTER-DST1/24;

                }
                destination-port [ 135 137 138 139 445 1782 9100 9280 9290
];
            }
            then {
                count kth-alba-printer;
                accept;
            }
        }
        term permit-out-all {
            from {
                destination-address {
                    130.237.0.0/16;
                    130.238.0.0/16;
                    130.239.0.0/16;
                    193.10.0.0/16;
                    193.11.0.0/16;
```

```
        }
    }
    then {
        count permit-out-all;
        accept;
    }
}
term block-port-out {
    from {
        protocol [ tcp udp ];
        destination-port [ 135 139 445 ];
    }
    then {
        count block-port-out;
        discard;
    }
}
term colo-udp {
    from {
        source-address {
            $COLO1/32;
        }
        protocol udp;
    }
    then {
        discard;
    }
}
term permit-out {
    from {
        source-address {
            130.237.85.0/24;
            130.237.86.0/23;
            130.237.88.0/21;
            130.237.144.0/22;
            130.237.148.0/23;
            130.237.151.0/24;
            130.237.152.0/22;
            130.237.156.0/24;
            130.237.160.0/24;
```

130.237.162.0/23;
130.237.164.0/22;
130.237.168.0/21;
130.237.176.0/24;
130.237.178.0/23;
130.237.184.0/21;
130.237.192.0/21;
130.237.205.0/24;
130.237.208.0/24;
130.237.217.0/24;
130.237.253.0/24;
130.237.254.0/24;
130.242.128.0/24;
193.10.6.0/24;
193.11.25.0/24;
193.11.26.0/23;
193.11.28.0/23;
193.11.30.0/24;
193.11.31.128/26;
193.11.94.0/23;
77.238.32.0/21;
224.0.0.0/4;
193.10.145.0/24;
193.10.147.0/24;
130.237.248.0/24;
193.11.92.0/23;
193.11.144.0/23;
130.237.240.0/24;
130.237.241.0/24;
130.237.242.0/24;
130.237.243.0/24;
130.237.244.0/24;
130.237.245.0/24;
130.237.246.0/24;
130.237.247.0/24;
130.237.200.0/24;
77.238.48.0/21;
130.237.180.0/24;
130.237.182.0/24;
130.237.183.0/24;

```
                130.242.47.0/24;
                130.237.158.0/24;
                130.237.177.0/24;
                192.36.127.0/24;
                130.237.159.0/24;
                193.10.10.0/24;
                193.10.9.0/24;
                193.10.8.0/24;
                130.237.150.0/24;
                130.237.161.0/24;
                130.237.157.0/24;
                130.237.181.0/24;
            }
        }
        then {
            count permit-out;
            accept;
        }
    }
    term icmp {
        from {
            protocol icmp;
        }
        then accept;
    }
}
filter re-protect-v4 {
    term allow-em0 {
        from {
            interface em0;
        }
        then accept;
    }
    term except-tcp-rate {
        from {
            source-prefix-list {
                ssh-clients;
            }
            protocol tcp;
            tcp-flags " (syn&!ack)|fin|rst ";
```

```
            }
            then {
                count except-tcp-syn-rate;
                accept;
            }
        }
        term tcp-syn-rate {
            from {
                source-prefix-list {
                    bgp-auto;
                }
                protocol tcp;
                tcp-flags " (syn&!ack)|fin|rst ";
            }
            then {
                policer tcp-syn-policer;
                count tcp-syn-rate;
                accept;
            }
        }
        term icmp-rate {
            from {
                protocol icmp;
                icmp-type [ echo-request echo-reply unreachable time-
exceeded timestamp timestamp-reply ];
            }
            then {
                policer icmp-policer;
                count icmp-rate;
                accept;
            }
        }
        term traceroute-rate {
            from {
                protocol udp;
                destination-port 33434-33650;
            }
            then {
                policer traceroute-policer;
                count traceroute;
```

```
            accept;
        }
    }
    term ssh-authorized {
        from {
            source-prefix-list {
                ssh-clients;
            }
            protocol tcp;
            port ssh;
        }
        then {
            count ssh-auth;
            accept;
        }
    }
    term bgp {
        from {
            source-prefix-list {
                bgp-auto;
            }
            protocol tcp;
            port bgp;
        }
        then {
            count bgp;
            accept;
        }
    }
    term snmp {
        from {
            source-prefix-list {
                snmp-clients;
            }
            protocol udp;
            port [ snmp snmptrap ];
        }
        then {
            count snmp;
            accept;
```

```
    }
}
term dns {
    from {
        source-prefix-list {
            dns-servers;
        }
        protocol udp;
        port domain;
    }
    then {
        count dns;
        accept;
    }
}
term ntp {
    from {
        source-prefix-list {
            ntp-servers;
        }
        protocol udp;
        port ntp;
    }
    then {
        count ntp;
        accept;
    }
}
term ignore-tcp-chatter {
    from {
        protocol tcp;
        destination-port [ 80 135-139 445 1433 1434 ];
    }
    then {
        count chatter-tcp;
        discard;
    }
}
term ignore-udp-chatter {
    from {
```

```
                protocol udp;
                port [ 135-139 1433-1434 ];
            }
            then {
                count chatter-udp;
                discard;
            }
        }
        term ospf {
            from {
                protocol ospf;
            }
            then {
                count ospf;
                accept;
            }
        }
        term last {
            then {
                count drop;
                log;
                syslog;
                discard;
            }
        }
    }
}
family inet6 {
    filter re-protect-v6 {
        term icmp-rate {
            from {
                next-header icmpv6;
            }
            then {
                policer icmp-policer;
                count icmp-rate-v6;
                accept;
            }
        }
        term traceroute-rate {
```

```
        from {
            next-header udp;
            destination-port 33434-33650;
        }
        then {
            policer traceroute-policer;
            count traceroute-v6;
            accept;
        }
    }
    term ssh-authorized {
        from {
            source-prefix-list {
                ssh-clients-v6;
            }
            next-header tcp;
            port ssh;
        }
        then {
            count ssh-auth-v6;
            accept;
        }
    }
    term bgp {
        from {
            source-prefix-list {
                bgp-auto-v6;
                bgp-auto-ls-v6;
            }
            next-header tcp;
            port bgp;
        }
        then {
            count bgp-v6;
            accept;
        }
    }
    term mcast {
        from {
            next-header [ pim igmp ];
```

```
        }
        then {
            count mcast-v6;
            accept;
        }
    }
    term mcast-scoped {
        from {
            destination-address {
                ff00::/13;
            }
        }
        then {
            count mcast-v6-scoped;
            accept;
        }
    }
    term bfd {
        from {
            prefix-list {
                bfd-clients-v6;
            }
            next-header udp;
            port 3784;
        }
    }
    term ospfv6 {
        from {
            next-header ospf;
        }
        then {
            count ospfv6;
            accept;
        }
    }
    term last {
        then {
            count drop-v6;
            log;
            syslog;
```

```
                discard;
            }
        }
    }
    filter fw_ipv6_SU_input {
        term permit-smtp-canit {
            from {
                source-address {
                    $MAILFILTER-V6-SRC1/128;
                    $MAILFILTER-V6-SRC2/128;
                    $MAILFILTER-V6-SRC3/128;
                    $MAILFILTER-V6-SRC4/128;
                }
                destination-address {
                    $MAILFILTER-V6-DST1/128;
                    $MAILFILTER-V6-DST2/128;
                    $MAILFILTER-V6-DST3/128;
                    $MAILFILTER-V6-DST4/128;
                    $MAILFILTER-V6-DST5/128;
                }
                next-header tcp;
                destination-port smtp;
            }
            then {
                count permit-smtp-canit-in-ipv6;
                accept;
            }
        }
        term port-udp {
            from {
                next-header udp;
                destination-port [ 7 9 11 19 42 67 68 69 79 123 135 136 137
138 139 161 162 177 427 445 514 515 517 518 524 587 593 1080 1433 1434 1900
2049 3372 5000 5003 5135 17 ];
            }
            then {
                count port-udp;
                discard;
            }
        }
```

```
            term port-tcp {
                from {
                    next-header tcp;
                    destination-port [ 7 9 11 15 19 25 42 67 68 69 79 111 123
135 136 137 138 139 161 162 427 445 515 524 587 593 635 705 950 1080 1433 1434
1521 1900 2049 2301 2967 3020 3128 3306 3372 5000 5003 5232 6000-6009 6101 6106
6112 6129 7937 8888 10000 13701 13711 13720-13724 13782-13783 22370 32370-32787
41523 49400 17 ];
                }
                then {
                    count port-tcp;
                    discard;
                }
            }
            term accept_all {
                then accept;
            }
        }
        filter fw_ipv6_SU_output {
            term permit-smtp {
                from {
                    source-address {
                        $SMTP-V6-MAILER1/128;
                        $SMTP-V6-MAILER2/128;
                        $SMTP-V6-MAILER3/128;
                        $SMTP-V6-MAILER4/128;
                        $SMTP-V6-MAILER5/128;
                    }
                    next-header tcp;
                    destination-port smtp;
                }
                then {
                    count permit-smtp-ipv6;
                    accept;
                }
            }
            term permit-smtp-canit {
                from {
                    source-address {
                        $SMTP-V6-MAILER1/128;
```

```
                    $SMTP-V6-MAILER2/128;
                    $SMTP-V6-MAILER3/128;
                    $SMTP-V6-MAILER4/128;
                }
                destination-address {
                    $MAILFITER-V6-DST1/128;
                    $MAILFITER-V6-DST2/128;
                }
                next-header tcp;
                destination-port smtp;
            }
            then {
                count permit-smtp-canit-ipv6;
                accept;
            }
        }
        term port-tcp {
            from {
                next-header tcp;
                destination-port 25;
            }
            then {
                count port-tcp;
                discard;
            }
        }
        term accept_all {
            then accept;
        }
    }
}
policer tcp-syn-policer {
    if-exceeding {
        bandwidth-limit 500k;
        burst-size-limit 15k;
    }
    then discard;
}
policer icmp-policer {
    if-exceeding {
```

```
            bandwidth-limit 5m;
            burst-size-limit 100k;
        }
        then discard;
    }
    policer traceroute-policer {
        if-exceeding {
            bandwidth-limit 2m;
            burst-size-limit 50k;
        }
        then discard;
    }
}
{master}
```

Tired of scrolling yet? A key reasoning for offering slices in SUNETs router to campus is to run firewall-filters as close to the core as possible. SU is doing a real good job here to do basic life-keeping filtering(these firewalls are completly controlled by campus networking) to make sure that the likes of printers don't enter the Internet and to block out unwanted traffic as far up as possible.

Now that we have showed our "hand" id LOVE to get feedback on stuff we have missed or features we might have overlooked, or is just doing wrong. We are considering publishing every configuration on every router on Github in the future to be fully transparent and give back to the community.


Skriven av

FREDRIK "HUGGE" KORSBÄCK

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