

AMS-IX Amsterdam Internet Exchange

Elzbieta Jasinska

Gregor Kopf

July 2, 2007



Abstract

This paper describes the usage of high speed switching technologies at the Amsterdam Internet Exchange. Various hardware in use on the shared medium will be described, and features explained. Furthermore a test setup was built in the AMS-IX lab, configuration examples will be shown and discussed.

Contents

| | | |
|----------|-----------------------------------|-----------|
| 1 | Introduction | 3 |
| 2 | AMS-IX | 3 |
| 2.1 | Connections and Traffic | 4 |
| 3 | Technical | 4 |
| 3.1 | Foundry | 8 |
| 3.1.1 | BigIron 15000 | 8 |
| 3.1.2 | BigIron RX16 | 8 |
| 3.1.3 | NetIron MLX32 | 9 |
| 3.2 | Photonic Crossconnects | 10 |
| 3.3 | VSRP | 11 |
| 3.4 | Port Security | 12 |
| 4 | Test Setup | 12 |
| 4.1 | Basic Setup | 13 |
| 4.2 | Create Loop | 15 |
| 4.3 | Prevent Loop | 17 |

List of Figures

| | | |
|----|--|----|
| 1 | Interconnecting networks | 3 |
| 2 | Current AMS-IX traffic statistics - daily | 5 |
| 3 | Current AMS-IX traffic statistics - yearly | 5 |
| 4 | Hub & Spoke Topology | 6 |
| 5 | Redundant Hub & Spoke Topology | 6 |
| 6 | Current AMS-IX topology | 7 |
| 7 | JetCore Architecture | 8 |
| 8 | Clos architecture of RX16 | 9 |
| 9 | Photonic Cross-Connect | 10 |
| 10 | PXC - light directed to the first output port | 10 |
| 11 | PXC - light directed to the second output port | 11 |
| 12 | VSRP Layout | 12 |
| 13 | Default Testsetup | 14 |
| 14 | Loop Testsetup | 15 |

1 Introduction

The internet is built out of different networks (autonomous systems), which need to be interconnected somewhere. For cost-saving reasons this often happens at so called “Internet Exchanges” (IX). An Internet Exchange is often implemented as a set of switches providing the infrastructure for parties to interconnect there (see [Figure 1](#)), the actual arrangements between the single customers, who is exchanging traffic with whom (peering), are up to them.

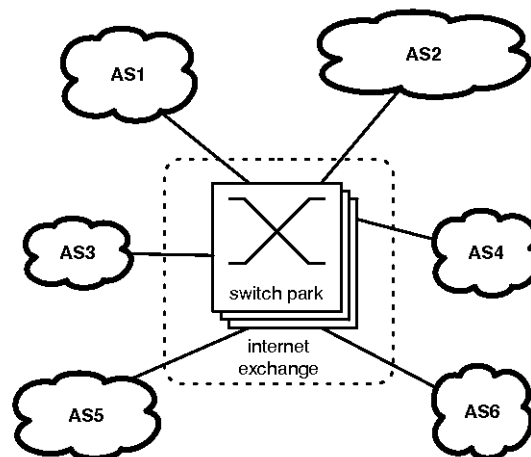


Figure 1: Interconnecting networks

This paper describes the technical aspects of running an Internet Exchange on the example of the biggest exchange in the world, the AMS-IX (Amsterdam Internet Exchange). [Section 2](#) explains high level how AMS-IX operates, [Section 3](#) goes into details about the topology and certain technologies used, [Section 4](#) shows a test setup built in the AMS-IX lab, findings and configurations.

2 AMS-IX

The AMS-IX - Amsterdam Internet Exchange - is a non-profit Internet Exchange based in Amsterdam. Over 250 parties from all around the world exchange IP traffic across the AMS-IX platform. Internet Service Providers, international carriers, mobile operators, content providers, VoIP providers, application providers, web hosters and other related businesses are all peering with each other across the shared medium [\[1\]](#).

At four independent co-location facilities in Amsterdam the members currently connect to 453 ports on the AMS-IX ethernet switches. Each member can choose one or more co-locations to set up their hardware or connect via a so-called “pseudowire” via a third party [2].

AMS-IX only provides the layer 2 platform for its members to exchange traffic, the arrangements between the parties, and with that who is actually exchanging traffic with whom, are up to them. These so called “peering-policies” are separate contracts not affecting AMS-IX at all. Network layer reachability information amongst the peering partners is exchanged using the BGP protocol (Border Gateway Protocol).

2.1 Connections and Traffic

The Amsterdam Internet Exchange uses Foundry Networks hardware [3] to operate the shared medium. It offers the following connections to its members:

- 10 Mbit/s “Ethernet” (10BaseT)
- 100 Mbit/s “FastEthernet” (100BaseTX)
- 1 Gbit/s “GE” (1000BaseSX, 1000BaseLX, 1000BaseLH)
- 10 Gbit/s “10GE” (10000BaseER, 10000BaseLR)

Due to the increasing demand for bandwidth approximately 100 10GE ports are already in use. Also link aggregation (LAG) of 10GE ports is pretty common (currently up to five ports aggregated as a single link towards a member’s router). For its backbone network AMS-IX uses aggregated links with up to 8 ports, which is the current maximum on the hardware in use. The NOC team is engaged with the IEEE working group on next generation ethernet (40GE or 100GE) [9] to push the standardization process forward, because the demand for higher speeds is growing rapidly.

AMS-IX has a traffic average of 180 Gb/s and peaks up to 287 Gb/s (see [Figure 2](#) on daily traffic volumes and [Figure 3](#) for a yearly overview). On average the traffic volume more than doubles every year [7].

3 Technical

The Network is built as a redundant hub & spoke topology (see [Figure 4](#)) using Glimmerglass photonic cross-connects [4] and Foundry Networks switches [5] (for details see [Subsection 3.1](#) on Foundry hardware and

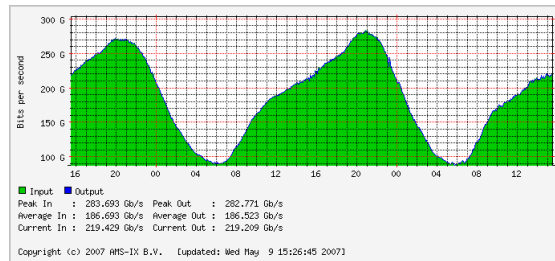


Figure 2: Current AMS-IX traffic statistics - daily

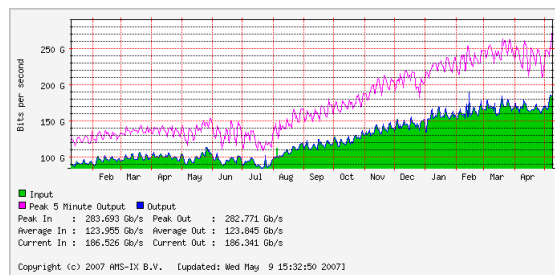


Figure 3: Current AMS-IX traffic statistics - yearly

[Subsection 3.2](#) on photonic cross-connects).

A hub & spoke has one core switch and multiple access switches connected to it (see [Figure 4](#)). Obviously the core and each connection between access and core switch is a single point of failure. Therefore AMS-IX implemented a redundant hub & spoke topology (see [Figure 5](#)).

A redundant hub & spoke layout contains loops, therefore a protocol called VSRP (Virtual Switch Redundancy Protocol [[6](#)]) is used. It enables the switches to block certain connections, so that either the red lines or the blue lines in [Figure 5](#) are in use. It also swaps over to the healthy topology in case of any failure on the network. For more details about VSRP see [Subsection 3.3](#).

Customers up to 1GE are directly connected to Foundry BigIron 15000 edge switches (on the bottom of [Figure 6](#)). 10GE customers are connected to Foundry Networks RX16 stub switches via Glimmerglass Networks System 300 photonic cross-connects (see top of [Figure 6](#)).

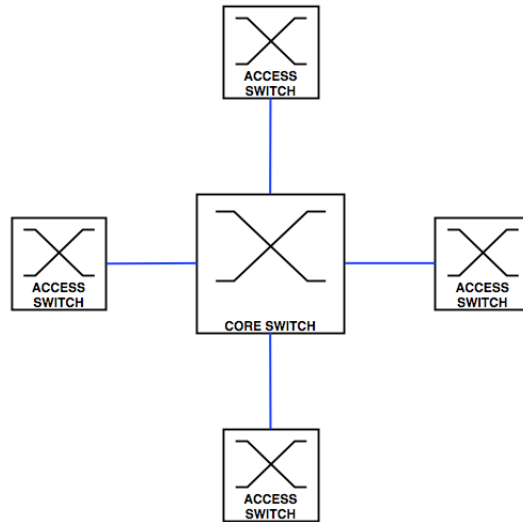


Figure 4: Hub & Spoke Topology

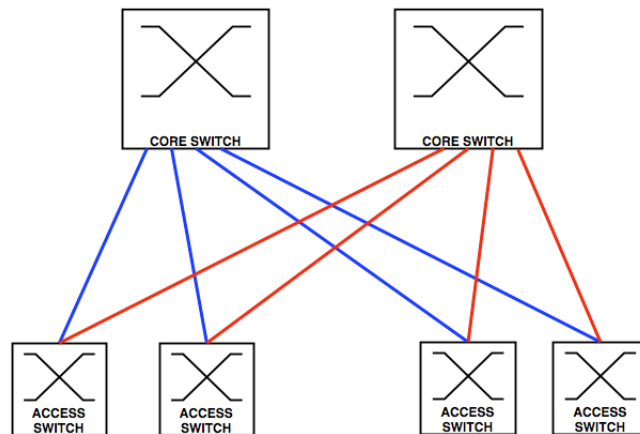


Figure 5: Redundant Hub & Spoke Topology

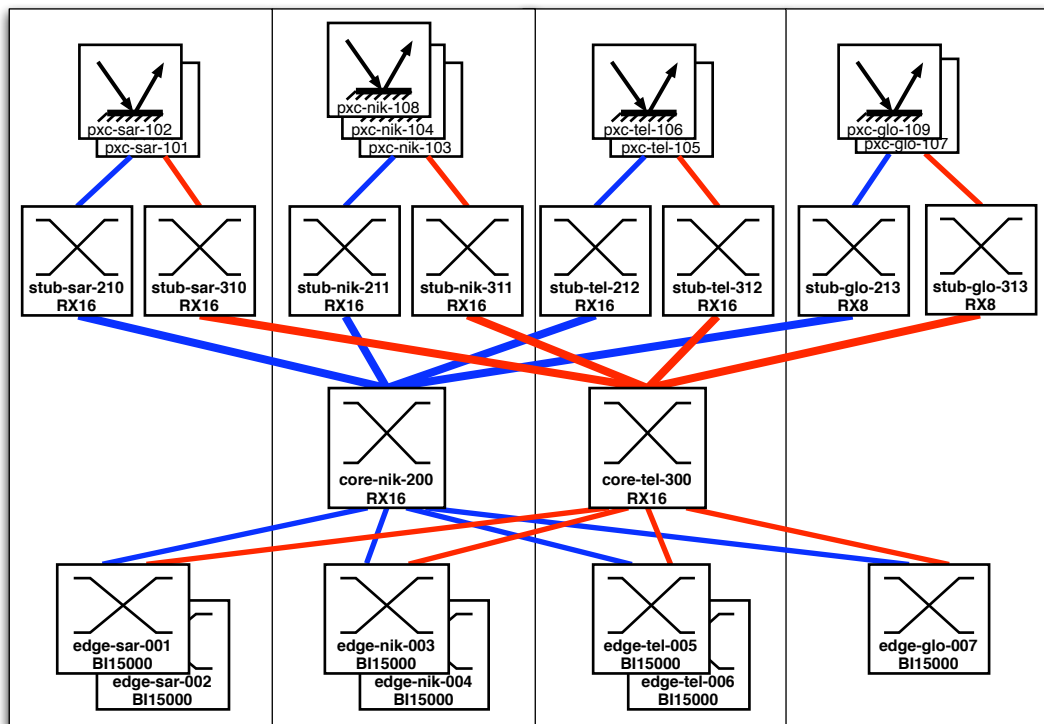


Figure 6: Current AMS-IX topology

3.1 Foundry

Foundry Networks [3] is a provider of switching and routing technologies. AMS-IX uses the Foundry BigIron and NetIron product family, a set of chassis-based layer 3 backbone switches. Since AMS-IX operates on layer 2 only, their routing abilities are not in use.

3.1.1 BigIron 15000

The BigIron15000 (Foundry’s JetCore architecture) switches connect all ports at Ethernet, FastEthernet and GigabitEthernet speeds to the AMS-IX platform (see Figure 6). These switches were introduced in 2002 and are rather reliable these days. They have 15 slots and a 8 Gbps backplane connector. The main switch fabric uses a crossbar architecture to interconnect the “port groups”. “Port groups” are groups of ports (4 for Gigabit Ethernet, 24 for 100/100 Mbps Ethernet), which have their own shared memory based switch fabric (see Figure 7).

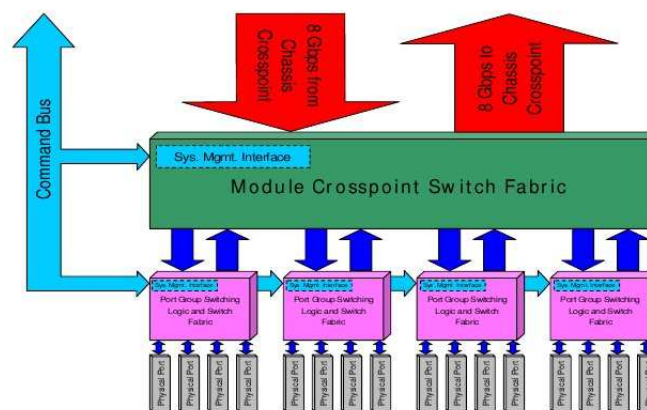


Figure 7: JetCore Architecture

3.1.2 BigIron RX16

The BigIron RX16 switches are used as core switches and access for 10GE customers. They are equipped with 16 slots and a 48 Gbps backplane connector. Foundry’s Clos-based architecture provides interface modules which hold the physical interfaces and a local CPU/DRAM (for the packet processing and traffic management) and switch fabric modules. The switch fabric modules consist of three fabric elements each, which perform the forwarding of frames between ports on all interface modules.

Every port on each interface module is connected to all switch fabric elements: traffic managers (on the interface modules) and switch fabric elements (on the switch fabric modules) build a complete bipartite graph (see [Figure 8](#)).

This has several advantages:

- Scalability (more switch fabric modules can be added)
- Redundancy (multiple paths from a source to a destination port; if a fabric element or even a fabric module fails, it's still possible to route traffic from one port to another)
- Non blocking (multiple ways from one port to another)

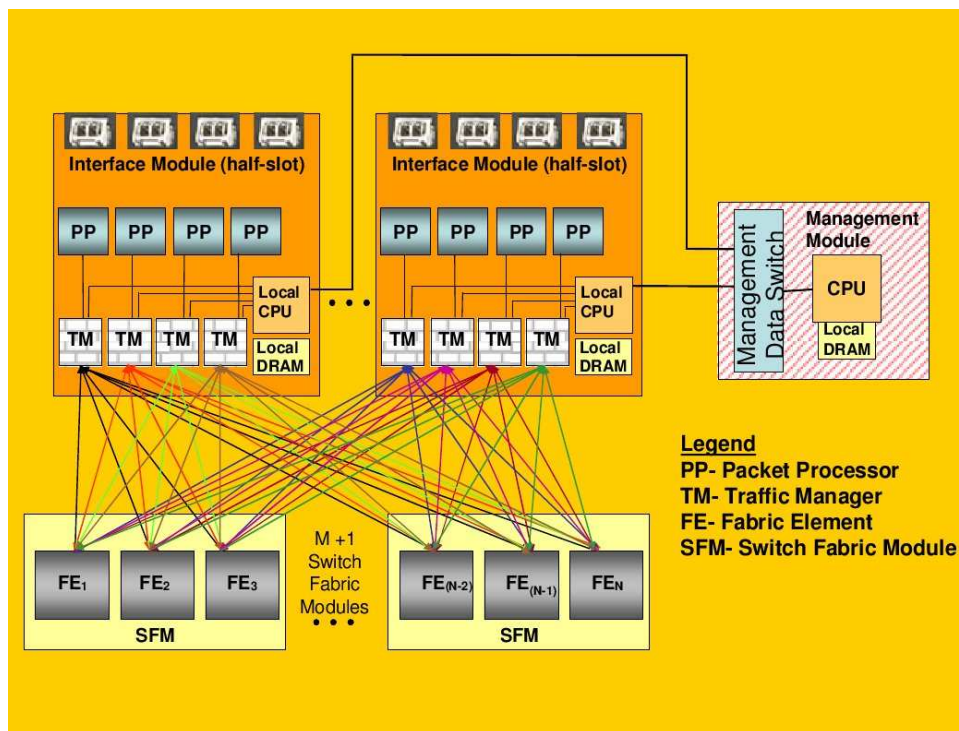


Figure 8: Clos architecture of RX16

3.1.3 NetIron MLX32

Just as the BigIron RX16, the NetIron has a Clos-based switch fabric inside. It has 32 slots and a 48 Gbps backplane connector. The MLX switch will replace the RX16 core switches in the future to have more capacity on the network.

3.2 Photonic Crossconnects

All 10GE customers at AMS-IX are connected through so called “Photonic Cross-Connects” to the AMS-IX ethernet switches (see [Figure 9](#)). These layer 1 switches build by Glimmerglass [4] allow to switch between connections at ms speeds.

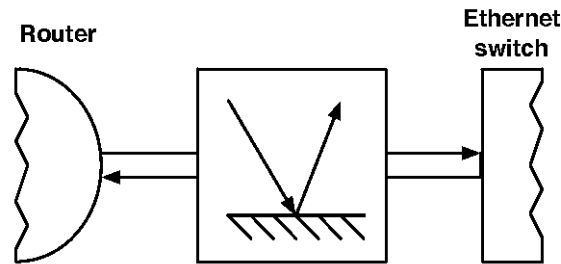


Figure 9: Photonic Cross-Connect

Each connection to a Photonic Cross-connect (PXC) consists of three ports. One incoming port (where the member’s router is connected to) and two outgoing ports. Movable mirrors inside of the switch define the angle in which the light is going and therewith the output port (see [Figure 10](#) and [Figure 11](#)).

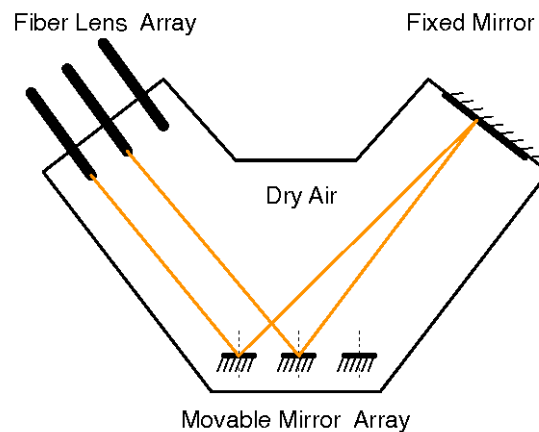


Figure 10: PXC - light directed to the first output port

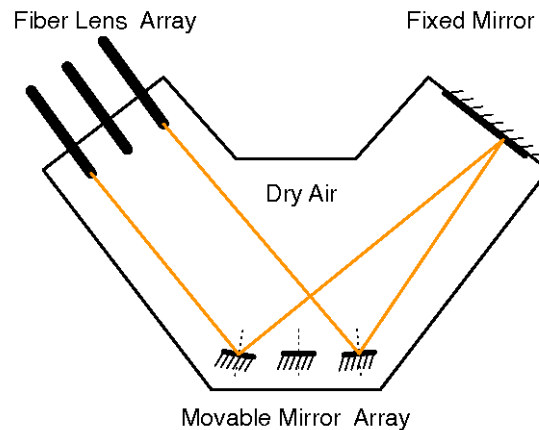


Figure 11: PXC - light directed to the second output port

3.3 VSRP

To provide a loop-free redundant architecture, AMS-IX uses Foundry's VSRP (Virtual Switch Redundancy Protocol). VSRP enables the switches to block certain ports, so that the meshed network structure remains still loop free.

The basic idea of VSRP is rather simple: there is one active switch and one passive backup switches (in case of AMS-IX the two core switches). The active switch sends out hello packets on all VSRP interfaces. If these packets are not seen for a configurable amount of time, one of the backup switches becomes active and unblocks its interfaces (see [Figure 12](#)).

VSRP aware edge switches know which uplink port the active VSRP backup switch is connected to by monitoring the hello packets. So if the hello packets are seen on another port, the edge switch clears its MAC table for addresses on the port to the former VSRP master switch. The photonic switches are reconfigured by a piece of software written by AMS-IX that listens to SNMP traps from both core switches.

3.4 Port Security

The greatest danger to any Ethernet network are loops. In case of AMS-IX the danger is again in members creating loops outside of the administrative domain of AMS-IX. Therefore AMS-IX uses a feature called "port security" to prevent loops on the exchange.

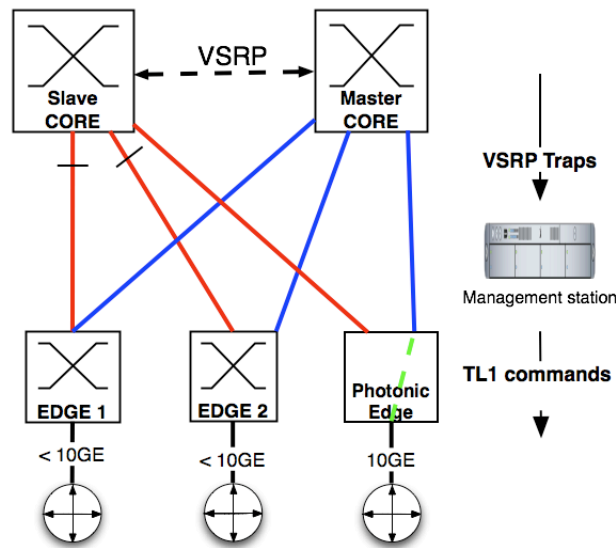


Figure 12: VSRP Layout

Port security means, that a switch is told to learn only a certain number of MAC addresses on one port. All frames with different source MACs are dropped (or the port can be shut down completely). AMS-IX only allows its customers to connect one router to an interface, and therewith one MAC address. As soon as the member's routing equipment is proven suitable for connecting to the AMS-IX platform, this routers MAC is learned by the AMS-IX switch. The test setup (see [Section 4](#)) describes in detail how loops are prevented with port security.

4 Test Setup

A BigIron 15000 switch from the AMS-IX lab was used to demonstrate the behavior in case of a ethernet loop. A Photonic Cross Connect was used to connect certain ports together and a traffic generator (Anritsu MD1230B [8]) to send traffic over the used links.

4.1 Basic Setup

Four 1GE ports on the BigIron 15000 were used for the setup, which were all connected to the traffic generator first. As can be shown with the command "sh int brief" all four ports are connected: the link is up.

```
SW-telnet@kannix(config)#sh int brief
```

| Port | Link | State | Dupl | Speed | Trunk | Tag | Priori | MAC | Name |
|------|------|---------|------|-------|-------|-----|--------|----------------|------|
| 1/1 | Up | Forward | Full | 1G | None | No | level0 | 0004.8096.6c00 | |
| 1/5 | Up | Forward | Full | 1G | None | No | level0 | 0004.8096.6c04 | |
| 3/1 | Up | Forward | Full | 1G | None | No | level0 | 0004.8096.6c40 | |
| 3/5 | Up | Forward | Full | 1G | None | No | level0 | 0004.8096.6c44 | |

The switch knows where to forward the ethernet frames to by “learning” the MAC addresses that are behind a port. Therefore traffic needs to be sent through each port first. With the command “sh mac” we can see the MAC-table which shows the MACs learned behind a port:

```
SW-telnet@kannix(config)#sh mac
Total active entries from all ports = 5
Type D:Dynamic S:Static L:Lock Address M:Secure Mac
MAC Address      Port  Age Type DMA Valid Flags      VLAN DMA:CAM Index ...
0000.0102.0100   1/1   9  DM 00000000-00000003    10  0:4    1:6
0000.0103.0100   3/1   0  DM 00000000-00000100    10  8:6
0000.0102.0200   1/5  10  DM 00000000-00000002    10  1:5
0000.0103.0200   3/5   0  DM 00000000-00000200    10  9:6
0007.8506.b030  15/48 9   D 02000000-00000000    150 57:6
```

As also illustrated in [Figure 13](#) four different MAC addresses are learned behind four different ports. Now the traffic generator connected to ports 1/1 and 1/5 is configured to send unicast traffic to each other and broadcast traffic to the broadcast domain they are in.

The counter values of for example port 1/1 show that the incoming values increase for unicast as well as for broadcast:

```
SW-telnet@kannix(config)#sh int eth 1/1
...
Transmitted 68564 broadcasts, 0 multicasts, 68562 unicasts
```

```
SW-telnet@kannix(config)#sh int eth 1/1
...
Transmitted 68785 broadcasts, 0 multicasts, 68784 unicasts
```

On both ports on blade 3 (3/1 and 3/5) only the broadcast value increases:

```
SW-telnet@kannix(config)#sh int eth 3/1
...
Transmitted 161 broadcasts, 0 multicasts, 0 unicasts
```

```
SW-telnet@kannix(config)#sh int eth 3/1
```

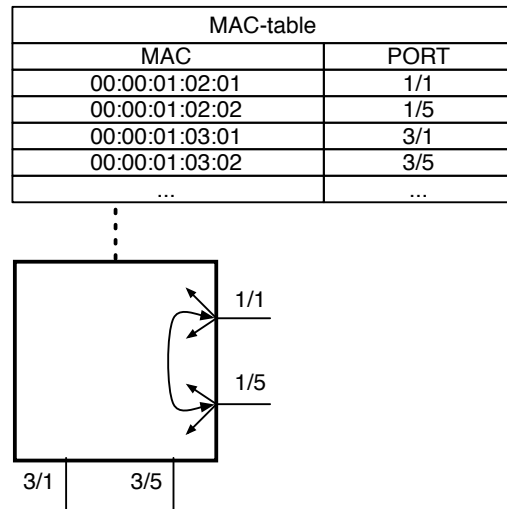


Figure 13: Default Testsetup

```

...
  Transmitted 878 broadcasts, 0 multicasts, 0 unicasts

SW-telnet@kannix(config)#sh int eth 3/5
...
  Transmitted 1219 broadcasts, 0 multicasts, 0 unicasts

SW-telnet@kannix(config)#sh int eth 3/5
...
  Transmitted 1518 broadcasts, 0 multicasts, 0 unicasts

```

Later the CPU load will increase due to the loop, therefore just to get a picture of the current state we perform the command “sh cpu”.

```

SW-telnet@kannix(config)#sh cpu
1 percent busy, from 8 sec ago
1  sec avg: 1 percent busy
5  sec avg: 1 percent busy
60 sec avg: 1 percent busy
300 sec avg: 1 percent busy

```

4.2 Create Loop

To create a loop we now connect port 3/1 and 3/5 together using the Photonic Cross Connect (as shown in [Figure 14](#)). This will cause a broadcast

storm and therewith station movements (relocation of the learned MAC addressees behind an interface).

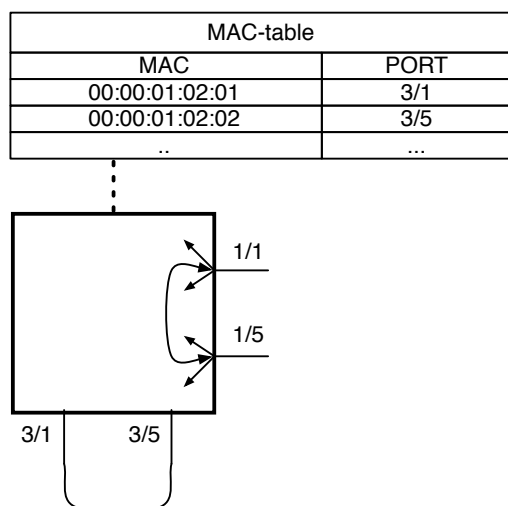


Figure 14: Loop Testsetup

A broadcast frame received at interface 3/1 for example, coming from interface 1/1 (MAC address 00:00:01:02:01:00) will be pushed back via interface 3/5 still containing the source MAC address of port 1/1. Therefore the switch will learn this MAC address behind port 3/5 now. The same frame also was received at port 3/5, going back via 3/1 and causing to learn the same source MAC address behind port 3/1, and so on.

This is illustrated with the following “sh mac” outputs performed right after each other:

```
SW-telnet@kannix(config)#sh mac
Total active entries from all ports = 9
Type D:Dynamic S:Static L:Lock Address M:Secure Mac
MAC Address      Port  Age  Type DMA Valid Flags   VLAN DMA:CAM Index ...
0000.0102.0100   3/5   0    D 00000000-00000300   10   8:4   9:6
0006.5b8d.1aec  15/48 10    D 02000000-00000000   150  57:4
0000.0102.0200   3/1   0    D 00000000-00000300   10   8:6   9:4
...
SW-telnet@kannix(config)#sh mac
Total active entries from all ports = 9
Type D:Dynamic S:Static L:Lock Address M:Secure Mac
```

```

MAC Address      Port  Age Type DMA Valid Flags      VLAN DMA:CAM Index ...
0000.0102.0100  3/1   0   D 00000000-00000303    10   0:4   1:6
0006.5b8d.1aec 15/48 10   D 02000000-00000000    150  57:4
0000.0102.0200  3/1   0   D 00000000-00000302    10   1:5   8:6
...
SW-telnet@kannix(config)#sh mac
Total active entries from all ports = 9
Type D:Dynamic  S:Static  L:Lock Address  M:Secure Mac
MAC Address      Port  Age Type DMA Valid Flags      VLAN DMA:CAM Index ...
0000.0102.0100  3/5   0   D 00000000-00000300    10   8:4   9:6
0006.5b8d.1aec 15/48 10   D 02000000-00000000    150  57:4
0000.0102.0200  3/5   0   D 00000000-00000303    10   0:5   1:5
...
SW-telnet@kannix(config)#sh mac
Total active entries from all ports = 9
Type D:Dynamic  S:Static  L:Lock Address  M:Secure Mac
MAC Address      Port  Age Type DMA Valid Flags      VLAN DMA:CAM Index ...
0000.0102.0100  3/1   0   D 00000000-00000300    10   8:4   9:4
0006.5b8d.1aec 15/48 148  D 00000000-00000000    150
0000.0102.0200  3/5   0   D 00000000-00000300    10   8:6   9:6

```

These rapid changes affect the switch CPU badly:

```

SW-telnet@kannix(config)#sh cpu
99 percent busy, from 1 sec ago
1  sec avg: 99 percent busy
5  sec avg: 99 percent busy
60 sec avg: 99 percent busy
300 sec avg: 99 percent busy

```

Even if we turn off the generated traffic now the broadcast frames in the loop will still remain going in cycles, received and transmitted on the interfaces on blade 3:

```

SW-telnet@kannix(config)#sh int eth 3/1
...
Received 91187 broadcasts, 0 multicasts, 0 unicasts
...
Transmitted 99604 broadcasts, 0 multicasts, 0 unicasts
...
SW-telnet@kannix(config)#sh int eth 3/1
...
Received 202020 broadcasts, 0 multicasts, 0 unicasts
...
Transmitted 220666 broadcasts, 0 multicasts, 0 unicasts

```

4.3 Prevent Loop

To prevent these kind of loops as described above we configure “port security” on the used ports. First we decide whether to shut down the ports on a MAC violation or just drop the unwanted frames. Furthermore, we can configure the amount of MAC addresses we wish to see behind a port.

In this case the ports are configured as “violation restrict” which will drop the unwanted frames and we only allow one MAC address per port:

```
SW-telnet@kannix(config)#sh port security
Port Security Violation Shutdown-Time Age-Time Max-MAC
-----
1/1 disabled restrict permanent 1
...
1/5 disabled restrict permanent 1
...
3/1 disabled restrict permanent 1
...
3/5 disabled restrict permanent 1
```

Now we enable the security feature on the needed ports:

```
SW-telnet@kannix(config)#int eth 1/1 to 1/5
SW-telnet@kannix(config-mif-1/1-1/5)#port sec
SW-telnet@kannix(config-port-security-mif-1/1-1/5)#enable

SW-telnet@kannix(config)#int eth 3/1 to 3/5
SW-telnet@kannix(config-mif-3/1-3/5)#port sec
SW-telnet@kannix(config-port-security-mif-3/1-3/5)#enable
```

After learning the correct MAC addresses on the ports we now see that this MACs are remembered as static MAC addresses with “sh mac”:

```
SW-telnet@kannix(config)#sh mac
Total active entries from all ports = 1
Total static entries from all ports = 4
Type D:Dynamic S:Static L:Lock Address M:Secure Mac
MAC Address Port Age Type DMA Valid Flags VLAN DMA:CAM Index ...
0000.0102.0100 1/1Stati SM 00000000-00000003 10 0:5 1:6
0000.0103.0100 3/1Stati SM 00000000-00000100 10 8:6
0000.0102.0200 1/5Stati SM 00000000-00000003 10 0:4 1:5
0000.0103.0200 3/5Stati SM 00000000-00000200 10 9:4
0007.8506.b030 15/48 9 D 02000000-00000000 150 57:6
```

If we now again create a loop as described in Section 4.2 no station movements will occur anymore because we won't have a broadcast storm so the switch will remain fully functional without high CPU load.

```
SW-telnet@kannix(config)#sh cpu
1 percent busy, from 4 sec ago
1 sec avg: 1 percent busy
5 sec avg: 1 percent busy
60 sec avg: 1 percent busy
300 sec avg: 1 percent busy
```

Therefore port security is a suitable feature to prevent loops from outside of the AMS-IX administrative domain.

References

- [1] Amsterdam Internet Exchange
<http://www.ams-ix.net>, 2007.
- [2] Amsterdam Internet Exchange - Technical
<http://www.ams-ix.net/technical/>, 2007.
- [3] Foundry Networks
<http://www.foundrynet.com/>, 2007.
- [4] Glimmerglass Optical Switches
<http://www.glimmerglass.com>, 2007.
- [5] Foundry Networks BigIron
<http://www.foundrynet.com/products/family/bigiron.html>, 2007.
- [6] Virtual Switch Redundancy Protocol
http://www.foundrynet.com/services/documentation/bigiron_rx_config/vsrp.html, 2007.
- [7] Amsterdam Internet Exchange - Traffic
<http://www.ams-ix.net/technical/stats>, 2007.
- [8] Anritsu MD1230B - IP / Ethernet / POS Quality Analyser
<http://www.eu.anritsu.com/products/default.php?p=189&model=MD1230B>, 2007.
- [9] IEEE 802.3 Higher Speed Study Group
<http://grouper.ieee.org/groups/802/3/hssg/index.html>, 2007.