Config guide

How to set up your device when connecting to MN-IX? Here are some pointers to start with.

MN-IX rules restrict the type of traffic and number of source MAC addresses that any member is allowed to send to the exchange.

1. Introduction

The Manama Internet Exchange operates as a shared Layer 2 (L2) Ethernet infrastructure. Large Ethernet LANs require that more or less everyone plays by the same set of rules. In other words, they can be quite sensitive to misbehavior.

In order to improve the stability of the Exchange, MN-IX has defined a set of rules to which every member's connection must adhere, the Technical Specifications.

It is not always easy to immediately grasp the subtleties of configuring equipment to adhere to the rules. Let us help you fill in some blanks and provide examples and hints for the most common equipment.

1.1 Definition of Terms

In this document we refer to terms like 'L2 device', 'L2/L3 hybrid', etc. Here are our definitions:

L2 Device

A device that functions as a Layer 2 (Ethernet) Bridge (a.k.a. 'switch', 'bridge', 'hub', etc).

L3 Device

A device that functions as a L3 (IP) router only. This means it does not bridge any Ethernet frames between its interfaces. Such a device is typically called a 'router'.

L2/L3 Hybrid

A device that functions both as a L2 bridge and a L3 router. This means it can both bridge Ethernet frames between its interfaces as well as route IP traffic and participate in IP routing protocols. Foundry/Brocade, Force10 and Extreme are common examples of this type of device.

2. The MN-IX Topology

The MN-IX network is built with Extremenetworks SLX9540 (formerly Brocade) switches.

Customers up to 1GE are directly connected to Extremenetworks edge switches, available at each location. One can connect with 1 (or mutliple) GE singlemode fiber. Fiber connections are supported using LX optics.

The 10G Ethernet access switches are locally available at each location and one can connect with either ER or LR optics.

3.1 IPv4 ARP / IPv6 Neighbor Timeout

Each equipment vendor implements its own maximum ages for the IPv4 ARP and IPv6 neighbor caches. The values vary widely and in at least one case (Linux) it is not a constant. Low ARP timeouts can lead to excessive ARP traffic, especially if the values are lower than the BGP KEEPALIVE interval timers. On the other hand, long timeouts can theoretically lead to longer downtime if you change equipment (since your peers still have the old MAC address in their ARP cache). With BGP this is unlikely to happen because your router will start re-establishing BGP sessions as soon as it is back up, causing its peers to update their ARP cache as well.

We recommend setting the ARP cache timeout to at least two hours, preferably four (240 minutes). See the sections on specific equipment vendors for examples.

3.2 Peering LAN Prefix

The IPv4 prefix for the MN-IX peering LAN (77.69.248.0/24) not supposed to be globally routable. This means the following:

- Do not configure 'network 77.69.248.0/24' in your router's BGP configuration (seriously, we have seen this happen!).

- Do not redistribute the route, a supernet, or a more specific outside of your AS. We announce it with a no-export attribute, please honor it.

In short, you can take the view that the Peering LAN is a link-local address range and you may decide to not even redistribute it internally (but in that case you may want to set a static route for management access so you can troubleshoot peering, etc.).

3.3 BGP Routing

Please exchange only unicast routes over your BGP sessions in the ISP peering LAN. Exchanging multicast routes is useless since multicast traffic is not allowed on the (unicast) ISP peering LAN.

4. Allowed Traffic Types and Configurations

The Technical Specifications state the following:

* There are only three ethertypes allowed:

    - 0x0800 - IPv4

    - 0x0806 - ARP

    - 0x86dd - IPv6

This implies IEEE 802.3 compliance, not 802.2, so no LLC encapsulation!

* Only one MAC address allowed on a port, i.e. all frames sent towards the MN-IX should have exactly one unique MAC address.

* The only non-unicast traffic allowed is:

  - Broadcast ARP

  - Multicast ICMPv6 Neighbour Discovery (ND) packets. (NOTE: this does not include Router Advertisement (ND-RA) packets!)

* MN-IX member equipment should only reply to ARP queries for IP addresses of their directly connected MN-IX interface. In other words, proxy ARP is not allowed.

* Traffic for link-local protocols is not allowed, except for ARP and IPv6 ND (see above).

* IP packets addressed to MN-IX peering LAN's directed broadcast address shall not be automatically forwarded to MN-IX ports.

* The speed and duplex setting of 10baseT and 100baseTX ports must be statically configured, i.e. auto-negotiation should be disabled.

* The MN-IX platform is designed to carry Ethernet frames with a payload of up to 1500 bytes. MTU settings must be configured accordingly.

4.1. Physical L2 Topology

The MN-IX rules dictate that only one MAC address is allowed behind a port. This means that you have to be extremely careful when connecting a device that can act as a L2 device.

We allow only one MAC address because we allow no additional devices behind the MN-IX ports. Extended L2 networks are not under the control of MN-IX, but instabilities in a L2 network behind the MN-IX switches can and typically do have a negative impact on the whole exchange. Forwarding loops and spanning tree topology changes are good examples of this. By enforcing the one-MAC-address-per-port rule, we effectively prevent forwarding loops and STP traffic from intermediate L2 devices.

In short, an intermediate L2 device may only bridge frames from the member's router to the MN-IX port (so we see only one MAC address) and should otherwise be completely invisible. No connected device should bridge frames from other devices onto MN-IX, or talk STP on its MN-IX interface.

4.1.1 Connecting a L3 Device

The most preferred way of connecting to MN-IX is directly through a L3 device (router). This is your best chance of not leaking MAC addresses or STP traffic and it greatly increases the stability of the network.

4.1.2 Connecting Through a L2 Device

We neither recommend nor encourage connecting your router through a L2 device, but if you do so, keep the following in mind:

* You must make absolutely sure that only traffic to/from your L3 router's interface goes to/from the MN-IX port.

* You must make absolutely sure that all legitimate traffic to/from your L3 router's interface goes to/from the MN-IX port.

* MLD snooping may block legitimate ICMPv6 neighbour solicitations.

* You must disable spanning tree on your link to MN-IX.

On all intermediate L2 devices, consider using explicitly defined port-based VLANs for production ports. It forces you to understand your topology and reduces the chances of a nasty surprise further down the road. In particular, we strongly recommend using a dedicated VLAN for the path from your router to MN-IX.

4.1.3 Connecting a L2/L3 Hybrid

The L2/L3 hybrid switch/router requires careful configuration in order to prevent unwanted traffic from leaking onto the exchange. As with intermediate L2 devices, you need to keep the following in mind:

* You must make absolutely sure that your MN-IX port is configured as a 'router only' port.

* You must disable Spanning Tree on your link to MN-IX.

On a L2/L3 hybrid device, it is a good idea to put the MN-IX connected interface (untagged) in a separate (non-default) port-based VLAN without spanning tree and with no other ports in it. This is the best way to ensure that no traffic from other ports will be bridged onto the MN-IX port.

4.2 Commonly Seen Illegal Traffic and Setup

Any traffic other than the types mentioned in the previous section is deemed to be illegal traffic. In this section we will list some of the more common types of violations we see at MN-IX and give some arguments as to why it is considered unwanted.

4.2.1 Multiple MAC addresses

Since MN-IX operates on the principle of one router per port, there should be one MAC address visible behind each port. Some members connect through intermediate switches, or use a L2/L3 hybrid device. If these devices are not configured properly, they can cause forwarding loops, STP instabilities, and lots of unwanted traffic on the exchange. There is no excuse for these devices to leak traffic, and there is no

necessity to talk STP on the link to MN-IX. Hence, by enforcing the one-MAC-address rule, we also enforce these issues. Beware that this rule is enforced automatically, so if you leak traffic from another MAC address, your legitimate traffic may be blocked (depending on which MAC address the switch sees first)!

4.2.2 Spanning Tree (STP)

This point is closely related to the previous point. The device(s) connected to the MN-IX port are not allowed to be visible as L2 bridges. This means that they should not speak STP (spanning tree) or any other (proprietary) L2 specific protocol.

4.2.3 Routing protocols: EIGRP, OSPF, RIP, IS-IS

The only routing protocol allowed on MN-IX is BGP. There is no valid reason for interior routing protocols to appear on the shared medium. These protocols only cause unnecessary multicast and broadcast traffic.

4.2.4 (Cisco) Keepalive

By default Cisco routers and switches periodically test their (Fast) Ethernet links by sending out Loopback frames (ethertype 0x9000) addressed to themselves. Call it a 'L2 self-ping' if you will. In a switched environment it can be used to test the functionality of the switch and/or keep the router's MAC address in the switch's address table.

In the MN-IX environment, this is not useful since we use MAC timeouts that are larger than the typical BGP and/or ARP timeouts. In fact, the keepalives a may actually cause port security violations if they are being sent by an intermediate switch.

4.2.5 Discovery protocols: CDP, EDP

Various vendors (e.g. Extreme, Cisco) tend to ship their boxes as gregarious devices: by default they announce their existence out of all their interfaces and try to find family members. CDP (Cisco) and EDP (Extreme) are examples of this, but there are others.

The only reason for running discovery protocols is to support certain types of autoconfiguration. Autoconfiguration on an Internet Exchange is a very bad idea. Hence, there is absolutely no reason to run discovery protocols on your MN-IX interface. Discovery protocols typically cause unwanted broadcast or multicast traffic.

4.2.6 Non-unicast IPv4: IGMP, DHCP, TFTP

On the ISP peering LAN, the only non-unicast traffic that is allowed is the ARP query.Sometimes we see equipment trying to get a configuration through broadcast TFTP, or configure themselves through DHCP. These options are unsafe and we strongly advise against them. Other equipment has IGMP turned on by

default (or by accident). The Peering LAN is for unicast IP traffic only, so there is no point in configuring multicast on the MN-IX interface.

4.2.7 Proxy ARP

Since traffic over MN-IX is exchanged based on BGP routes, there is no reason to answer ARP queries for any other IP address(es) than those that are configured on your MN-IX interface. Unfortunately, some vendors (e.g. Cisco) ship their products with proxy ARP enabled by default. Proxy ARP is not only sloppy, it can lead to unwanted traffic on your network. Consider that if you have it enabled at MN-IX, it's likely to be enabled at other peering points, allowing parties on both sides to use you as a transit. Proxy ARP is not allowed.

4.2.8 Non-unicast IPv6: IPv6 ND-RA

IPv6 router advertisements are not allowed: they generate a lot of unnecessary traffic, since IPv6 hosts on MN-IX are not autoconfigured and besides, you don't want to be the default router for MN-IX as a whole.

4.2.9 Miscellaneous non-IP: DEC MOP, etc.

Some vendors enable protocols other than IP by default. Cisco, for example ships certain versions of IOS with DEC MOP enabled by default. This is non-IP traffic and has no place on MN-IX.

5. Cisco Configuration Hints

Cisco's philosophy seems to be similar to that of some PC OS vendors: enable as many protocols and features as possible by default, so the device works out-of-the-box in most situations. Unfortunately, this means that a lot of unnecessary features are turned on that, while harmless in LAN or corporate environments, can cause undesired traffic on an Internet Exchange.Typical things that need to be disabled are: autoconfiguration protocols (DHCP, BOOTP, TFTP config download over the MN-IX interface), CDP, DEC MOP, IP redirects, IP directed broadcasts, proxy ARP, IPv6 Router Advertisements, keepalive.

Intermediate switches or hybrid devices will also need to disable VTP, STP, etc.

5.1 Global Config

Global configuration

```
! Do not run a DHCP server/relay agent

no service dhcp

! Older IOS versions require this instead of the above.
```

```
no ip bootp server
! Do not download configs through TFTP
no service config
! Do not run CDP
no cdp run
```

5.2 Interface Config

Interface configuration

```
! Don't do redirects -- if they don't know
! how to route properly, tough luck!
no ip redirects
! Don't run proxy ARP on your MN-IX interface
no ip proxy-arp
! Don't run CDP on your MN-IX interface
no cdp enable
! Directed broadcasts are evil.
no ip directed-broadcast
! Disable the DEC drek if you haven't done so globally yet.
no mop enable
! For (Fast)Ethernet: no auto-negotiation on your connection.
! no negotiation auto
! duplex half
duplex full
! L2 keepalives are useless on the MN-IX
no keepalive
```

5.3 Layer 2 Config

It is difficult to give a complete guide for Cisco products, because of the many different types of devices and (IOS) software versions. When in doubt, consult your documentation.

5.3.1 29xx and 35xx Series

If you use a Cisco Layer 2 device (such as the 2900 and 3500 series), you have to turn off VTP (VLAN Trunking Protocol), DTP (Dynamic Trunking Protocol), LLDP, and UDLD.

In global config mode:

```
vtp mode transparent
!
no spanning-tree vlan 1200
! If you don't need LLDP, disable globally
no lldp run
! If you don't need CDP, disable globally
no cdp run
!
vlan 1200
 name MN-IX
!
interface /IfIdent/
 description Interface to MN-IX
 switchport access vlan 1200
 switchport mode access
 switchport nonegotiate
 no keepalive
 speed nonegotiate
 no udld enable
 ! If CDP has not been disabled globally:
 no cdp enable
```

```
 ! If LLDP has not been disabled globally:
 no lldp receive
 no lldp transmit
 ! If you do not want to shut off STP:
 spanning-tree bpdufilter enable
end
```

5.3.2 7600 Series

Members are advised not to run 12.2(33)SRC on their Cisco 7600's with a sup720. This software release does not always send or forward replies to solicit requests, even if it's acting as a pure Layer 2 switch between a member router and the MN-IX fabric.

To make a Cisco 7600 switch 'silent' the following configuration seems to work:

```
no service dhcp
no ip bootp server
vtp mode transparent
spanning-tree mode pvst
spanning-tree extend system-id
no spanning-tree vlan XX
!
vlan XX
 name mnix
 exit
!
interface GigabitEthernet6/0/0
 description to-mnix switchport
 switchport access vlan XX
 switchport mode access
 switchport nonegotiate
```

```
 no mls qos trust
 no cdp enable
 spanning-tree bpdufilter enable
 exit
!
```

Vlan XX was also removed from the 'allow list' on all dot1q trunk ports not related to the setup, in this case every dot1q trunk port in the chassis.

5.3.3 Catalyst 6500 Series

CatOS and IOS are different beasts, so for Catalyst switches, the following applies:

```
set vtp mode off
set port name /IfIdent/ My MN-IX Port
set cdp disable /IfIdent/
set udld disable /IfIdent/
set trunk /IfIdent/ off dot1q
set spantree bpdu-filter /IfIdent/ enable
set vlan 1200 name My_MN-IX_Vlan
set vlan 1200 /IfIdent
```

If, for some reason, you cannot afford to turn off VTP globally, the only way to turn it off on individual ports seems to be by using l2pt:

```
set port l2protocol-tunnel /IfIdent/ vtp enable
```

Depending on your CatOS platform, you may or may not be able to do this.

5.3.4 CRS (IOS-XR)

CDP, Proxy ARP, Directed Broadcast, Link Auto Negotiation, and ICMP redirects* are disabled by default in IOS-XR. ICMP redirect messages are disabled by default on the interface unless the Hot Standby Router Protocol (HSRP) is configured.

5.3.5 Other Devices

For other devices, some or all of the above may apply. Check your documentation for details.

5.4. Cisco Aggregated Links

5.4.1 Catalyst 6500 Series

Configure the port-channel as on, or should you want LACP, as active. Please do not not configure any forms of negotiate or desirable as the MN-IX switches do not speak PAgP.

Load-balancing over four ports may result in an unequal distribution due to bug CSCsg80948.

```
! Here is an example configuration:
interface GigabitEthernet1/1
 description MN-IX Link 1
 no ip address
 no ip redirects
 no ip proxy-arp
 no keepalive
 no cdp enable
 channel-group 1 mode on
!
interface GigabitEthernet1/2
 description MN-IX Link 2
 no ip address
 no ip redirects
 no ip proxy-arp
 no keepalive
 no cdp enable
 channel-group 1 mode on
!
interface Port-channel1
```

```
 description MN-IX aggregated link
 ip address 77.69.248.x 255.255.255.0
 no ip redirects
 no ip proxy-arp
 no keepalive
!
```

Here are examples of LACP configurations:

Cisco IOS 65xx/76xx:

```
interface GigabitEthernet1/1
 description MN-IX Link 1
 channel-group 10 mode active
!  (12.2(18)SXF2  or  (12.2(33)SRC) upwards)

 lacp rate fast
!
interface GigabitEthernet1/2
 description MN-IX Link 2
 channel-group 10 mode active
!
interface Port-channel10
 description MN-IX aggregated link
 no switchport
 ip address 77.69.248.x 255.255.255.0
!
```

Cisco IOS-XR:

```
interface Bundle-Ether 10
 description MN-IX aggregated link
 ipv4 address 77.69.248.x 255.255.255.0
!
interface GigabitEthernet 1/0/0/0
 description MN-IX Link 1
 bundle-id 10 mode active
!  (3.2 upwards)

 lacp period short
!
interface GigabitEthernet 1/0/1/0
 description MN-IX Link 2
 bundle-id 10 mode active
!
```

(don't forget to commit)

Cisco NX-OS:

```
feature lacp
!
interface ethernet 2/1
 description MN-IX Link 1
 channel-group 10 mode active
 lacp rate fast
!
interface ethernet 2/2
 description MN-IX Link 2
```

```
 channel-group 10 mode active
!
interface port-channel 10
 description MN-IX aggregated link
 ip address 77.69.248.x 255.255.255.0
!
```

5.4.2 GSR Series

Do not set a static MAC address on the Port-channel interface. This causes CEF inconsistencies and other assorted failures.Link aggregation and IPv6 do not seem to play well together. Cisco advises against trying this.

Some changes will result in a different MAC address getting chosen for the aggregated link (likely such as reloading a linecard, if it contains the first port in the bundle). This will keep your ports dysfunctional due to port security on the MN-IX switches and you will have to contact the MN-IX NOC in such cases to fix this.

Some restrictions apply to what features are supported on link bundles (e.g. sampled NetFlow only on ISE/Engine4+; no uRPF). Also not all line cards support link bundling, and if traffic towards MN-IX comes in on such an interface you will experience suboptimal load-balancing. Please see the Cisco documentation for more details.

Support for link bundling on Engine 5 linecards will come in 12.0(33)S.Cisco Engineering have a special train called 'Phase 3' (lb-eft-ph3) that is purported to also provide functionality such as MAC address accounting for Port-Channel interfaces. This seems to have been integrated into 12.0(32)S, but IPv6 does not seem to be supported yet.

Below follows a list of Cisco Bug IDs (ddts) related to link aggregation that you need to consider when choosing an appropriate IOS image

CSCee27396

present in 12.0(26)S1; fixed in 12.0(26)S3, 12.0(27)S2, 12.0(28)S1, 12.0(30)SSymptoms: Over 90% CPU usage by CEF Scanner on all linecards and %TFIB-7-SCANSABORTED errors occur when configuring a link bundle. Also, the router sends traffic to MAC addresses taken from its ARP table seemingly at random, instead of to the appropriate next-hop's MAC address.

CSCef12828

present in post-CSCee27396; fixed in 12.0(26)S4, 12.0(27)S3, 12.0(28)S1, 12.0(30)SSymptoms: When traffic passes through a router, the router blocks traffic for certain prefixes behind a port-channel link.

CSCdz33664

present in 12.0(25)S3, 12.0(26)S1, 12.0(27)S2, 12.0(28)S; fixed in 12.0(25)S4Symptoms: An HSRP state change on any Engine2 interface causes a microcode bundle flap on all other Engine2 linecards, preventing load balancing to work due to vanilla microcode getting loaded.

CSCee81071

present in 12.0(26)S3, 12.0(27)S2, 12.0(29)SSymptoms: Router sends Ethernet frames with a source MAC address of beef.f00d.beef and destination MAC address f00d.beef.f00d (which is the pattern scribbled in unallocated memory in linecards), with what looks to be a legitimate payload of transit traffic. This is one of the symptoms of CSCee27396

CSCeb38014

present in 12.0(26)S5; fixed in 12.0(26)S5, 12.0(27)SSymptoms: The BGP Router process flushes the BGP tables for each peer when you change one neighbor's description. This pegs the GRP CPU at 99% for quite a while.

CSCeg31951

present in 12.0(31)S; fixed in 12.0(31)S2 (CSCei53226)

Symptoms: IOS (at least in the PRP code) places each individual public peer in its own update-group if remove-private-as is configured on a peer. Needless to say, this scales badly for a router connected to an Internet exchange. (Try 'show ip bgp replication'.) A collection of hearsay follows for recent IOS images for the GSR PRP regarding link aggregation. MN-IX does not run any GSRs. Please take this information with appropriately-sized grains of salt.

12.0(24)S2 is not advisable (not many specifics known but they include CSCef89562 and CSCee33045)

12.0(24)S6 boots but load-balancing is completely off 12.0(25)S until S3 have CSCdz33664

12.0(26)S until S4 have CSCef89562, where Engine4+ linecards can have continuously flapping interfaces, but is also somewhat required for Quadra linecards

12.0(26)S3 has CSCee27396 integrated but not CSCef12828, which leads to traffic blackholing 12.0(27)S until S3 have CSCef89562 as well

12.0.(27)S1 has a problem where it sends traffic to random destinations 12.0(27)S2 has CSCee27396 integrated but not CSCef12828

12.0(27)S4 reportedly works reasonably well on PRP2s

12.0(28)S1 has problems with Engine2 linecards (CSCef78098) and Engine4+ (CSCef89562)

12.0(28)S2 reportedly works better but still sometimes emits beef.f00d.beef frames on normal ports with only an IPv6 address configured

12.0(30)S has only been observed to exhibit CSCef12828-like symptoms in conjunction with broken hardware, and also (sometimes) to still emit frames from MAC beef.f00d.beef.

Routers occasionally still send out frames with beef.f00d.beef as MAC source address on interfaces with an IPv6 but no IPv4 address configured, even on regular links.

Due to the massive amount of feature requests there will be both a 12.0(32)S and a new 12.0(32)SY train.

You can check for incorrect next-hops by attaching to the linecard and executing show controllers rewrite and show adjacency internal and comparing the two rewrite strings for a certain peer's IPv4 address (suffix the commands with | begin 77.69.248.b). The first six bytes of the returned long hex string should be the peer's MAC address, and equal for all three occurrences.

```
! An example configuration follows:
!
interface Port-channel1
 description MN-IX Aggregated Link
 ip address 77.69.248.x 255.255.255.0
 no ip redirects
 no ip directed-broadcast
 no ip proxy-arp
 channel-group minimum active 1
 no channel-group bandwidth control-propagation
 hold-queue 150 in
!
interface GigabitEthernet1/2/1
 no keepalive
```

```
 no negotiation auto
 channel-group 1
 no cdp enable
!
interface GigabitEthernet1/2/2
 no keepalive
 no negotiation auto
 channel-group 1
 no cdp enable
!
```

Specifying a value is optional, but setting it to the amount of ports in an aggregated link multiplied by 75 is advised. show interfaces Port-channel 1 will display keepalives enabled even though they are not; also, the BIA (burnt-in address, shown as 0000.0000.0000) can be ignored.

If you disable autonegotiation on Gigabit Ethernet ports please CONTACT US

5.4.3 CRS (IOS-XR)

```
interface Bundle-Ether1
 description Aggregated interface to MN-IX Peering LAN
 ipv4 address 77.69.248.x 255.255.255.0
 bundle minimum-active links 1
!
interface TenGigE0/0/0/0
 description interface to MN-IX Peering LAN #1
 bundle id 1 mode on
!
interface TenGigE0/0/0/1
 description interface to MN-IX Peering LAN #2
 bundle id 1 mode on
```

!

5.5 Cisco 10GE Specifics

IOS supports no bgp fast-external-fallover and event dampening . The no bgp fast external-fallover tells the device to not act immediately on link flaps but to wait for the BGP hold timers to expire before resetting sessions.

Newer versions of Cisco IOS even support ip bgp fast-external-fallover deny in a per-interface context.Note that in practice we have found that the previously advised carrier-delay does not work as expected on Cisco equipment. We suggest you disable fast-external-fallover instead.

In IOS-XR, to disable BGP Fast External Failover globally, add bgp fast-external-failover disable to your global bgp configuration.

5.7 MTU Config

On newer Cisco IOS/IOS-XR versions, the interface IP MTU is automatically set, based on the presence or absence of 802.1q tags. For more details, please consult this document.

6. Extreme Networks Configuration Hints

CAUTION: Updating Firmware in an EAPS Environment

When updating firmware in an Extreme Networks EAPS environment, be sure to temporarily disable your MN-IX port(s). TFTP file transfers may cause EAPS instabilities resulting in bogus traffic. This is likely to trip the port security on the MN-IX switches, which may result in 10 minutes downtime. Most people who use Extreme equipment do not have problems with their MN-IX connections, some do. We would appreciate feedback from people running Extreme equipment on how they configure their MN-IX facing side.

If you are running Extreme equipment and would like to share your feedback CONTACT US

6.1 L2 Configuration

The configuration fragment below shows how to configure an intermediate L2 switch, which is also part of an EAPS ring. Port 1 is connected to the MN-IX switch. Ports 2 and 3 are in the ring. The router is somewhere in that ring, in the 'mnix' VLAN.

```
create vlan "ring"

configure vlan "ring" tag 1200  # VLAN-ID=0x4b0  Global Tag 3

configure vlan "ring" qosprofile "QP8"

configure vlan "ring" add port 2 tagged

configure vlan "ring" add port 3 tagged
```

```
create vlan "mnix"
configure vlan "mnix" tag 1700  # VLAN-ID=0x6a4  Global Tag 9
configure vlan "mnix" add port 1 untagged
configure vlan "mnix" add port 2 tagged
configure vlan "mnix" add port 3 tagged

configure port 1 auto off speed 1000 duplex full
configure port 2 auto off speed 1000 duplex full
configure port 3 auto off speed 1000 duplex full

disable edp port 1
disable igmp snooping
disable igmp snooping with-proxy

create eaps "ring-eaps"
configure eaps "ring-eaps" mode transit
configure eaps "ring-eaps" primary port 2
configure eaps "ring-eaps" secondary port 3
configure eaps "ring-eaps" add control vlan "ring"
configure eaps "ring-eaps" add protect vlan "mnix"
enable eaps "ring-eaps"
```

6.2 L3 Configuration

The configuration fragment below shows the relevant configuration information for a L3-only device. As in the previous example, port 1 is connected to MN-IX and is configured in the 'mnix' VLAN (untagged).

#

```
# Config information for VLAN mnix.
#
create vlan "mnix"
configure vlan "mnix" tag 1200
configure vlan "mnix" protocol "IP"
configure vlan "mnix" ipaddress 77.69.248./Y/ 255.255.255.0
configure vlan "mnix" add port 1 untagged
#
configure port 1 display-string "MN-IX"
disable edp port 1
#
enable ipforwarding vlan "mnix"
disable ipforwarding broadcast vlan "mnix"
disable ipforwarding fast-direct-broadcast vlan "mnix"
disable ipforwarding ignore-broadcast vlan "mnix"
disable ipforwarding lpm-routing vlan "mnix"
disable isq vlan "mnix"
disable irdp vlan "mnix"
disable icmp unreachable vlan "mnix"
disable icmp redirects vlan "mnix"
disable icmp port-unreachables vlan "mnix"
disable icmp time-exceeded vlan "mnix"
disable icmp parameter-problem vlan "mnix"
disable icmp timestamp vlan "mnix"
disable icmp address-mask vlan "mnix"
disable subvlan-proxy-arp "mnix"
```

```
configure ip-mtu 1500 vlan "mnix"

#

# IP Route Configuration

#

configure iproute add blackhole default

disable icmpforwarding vlan "mnix"

disable igmp vlan "mnix"
```

7. Force10 Configuration Hints

There isn't much to configure on Force10 routers. The Network Operations Guide and various pages in the Team Cymru Document Collection provide useful information on Force10 router configuration and management.

```
! Disable proxy ARP on your MN-IX interface

Force10(conf)#interface tengigabitethernet 0/0

Force10(conf-if-te-0/0)#no ip proxy-arp


!  Disable IPv6 ND RAs

Force10(conf-if-te-0/0)#ipv6 nd suppress-ra


! The default ARP timeout is 4 hours, but can be changed with this command

Force10(conf)#interface tengigabitethernet 0/0

Force10(conf-if-te-0/0)#arp timeout /minutes/
```

7.1 Force10 10GE Specifics

Force10 E-Series switch/routers support no bgp fast-external-fallover, BGP Graceful Restart, and a link debounce timer to maintain BGP stability during topology switchovers.The recommended option is to use the /link debounce/ command to delay link change notifications on the interface. The default for fiber interfaces is 100 ms, which is a good value to use.

8. Foundry/Brocade Configuration Hints

The following fragment of configuration gives an idea of how to configure a Foundry (BigIron) device. Depending on the actual role of the device (router or switch between router and MN-IX) and the type of code loaded into the device you may need to mix and match a little here.

```
! Define a single-port VLAN for the MN-IX port

vlan number name "MN-IX" by port

no spanning-tree

untagged ethernet if



! Configure the MN-IX interface

interface ethernet if


 port-name "MN-IX"


! Behave as a router.

 route-only

 no spanning-tree


! Don't do IPv6 ND-RA (Router Advertisements)

 ipv6 nd suppress-ra


! No weird discovery proto, please.

 no vlan-dynamic-discovery


! IP address

 ip address 77.69.248.x 255.255.255.0
```

```
! No redirects
 no ip redirect
 no ipv6 redirect


! MN-IX recommends 2 hour ARP timeouts
 ip arp-age 120


! For fast-ethernet: no autoconfig.
 speed-duplex 100-full
```

8.1 Foundry/Brocade Aggregated Links

BigIron JetCore-based switches support link aggregation only on adjacent ports. The first port must be oddly numbered, and the other port must directly follow the first one. The same goes for any additional pairs of ports in an aggregated link.

CAUTION: On BigIron 15000 switches you cannot build trunks with ports on blade 8, or spanning ports on both sides of slot 8!

```
! Create an aggregate on a Jet-Core based switch
trunk server ethernet slot/port to slot/port+1
```

BigIron RX or NetIron MLX/XMR switches don't have limits to port placement for aggregated links. Ports can be non-adjacent or even distributed over multiple blades. BigIron RX has a limit of 8 ports per aggregated link, NetIron MLX/XMR raise this to 16 in software 3.5.0, 32 in 3.8.0

```
! Create an aggregate on a RX/MLX/XMR switch
trunk ethe slot/port to slot/port ethe otherslot/otherport to
otherslot/otherport
```

As of RX software release 2.5.0 and MLX/XMR software release 3.9.0 the link aggregation syntax changed. The configuration now looks like:

```
! Create a LAG on a RX/MLX/XMR switch
lag "<NAME HERE>" static
 ports ethernet #/# ethernet #/# <and so on>
 primary-port #/#
```

```
 deploy
!
```

The primary-port is used as a single point of configuration. All configuration changes to the primary-port are propagated to the other ports in the lag group.

The keyword 'static' designates a standard aggregated link. For an LACP-enabled link, use:

```
! Create a dynamic LAG on a RX/MLX/XMR switch
lag "<NAME HERE>" dynamic
 ports ethernet #/# ethernet #/# <and so on>
 primary-port #/#
 lacp-timeout short
 deploy
!
```

8.2 Foundry/Brocade 10GE Specifics

Foundry/Brocade supports a feature called BGP Graceful Restart that, if all peers support it, will reduce the impact of prefix flaps but the CPU will still have to re-establish any flapped BGP session before the configured interval passes. The command delay-link-event can make the router ignore short link flaps. We recommend setting this to 20 which equals to 1000 msecs. Consequently, the flap will be logged in syslog, but higher level protocols (BGP in this case) will be unaffected. We suggest to leave fast-external-fallover in its default state.

9. HP Configuration Hints

Recommendations we received for HP ProCurve devices:

```
spanning-tree ifname bpdu-filter spanning-tree ifname tcn-guard lldp admin-status ifname disable
```

10. Juniper Configuration Hints

For Juniper routers, there isn't much to disable. The Juniper Documents contain useful hints on how to set up your Juniper router.

CAUTION: IGMP Bug (PR/20343) in Junos OS versions 5.3R4 !

There's a bug in Junos OS versions up to 5.3R4, that will cause a Juniper router to emit IGMP packets on all its interfaces, even when IGMP is disabled. The only way to stop your router from transmitting IGMP is to configure outgoing packet filters on your MN-IX interface(s).

10.1 Unicast BGP Configuration

Make sure to exchange only unicast routes in the unicast ISP peering LAN by explicitly adding the following statement to ,em>all neighbors, groups and prefix-limits:

```
set family inet unicast
```

Be thorough with family inet unicast

If even one of the neighbours, groups or prefix-limits is defined with a family inet "any", you'll enable multicast and turn on MBGP.

Increasing interface hold-time (1200ms) to preserve BGP sessions during 10/100GE interface flapping

```
user@router# show interfaces xe-0/1/0

description "interface to MN-IX Peering LAN";

hold-time up 1200 down 1200
```

10.2 IPv4 ARP Cache Timeout

Juniper's default ARP cache timeout is 20 minutes (by comparison: Cisco's default ARP cache timeout is 4 hours, which fits MN-IX's relatively static environment much better).

To reduce the amount of unnecessary broadcast traffic, we recommend setting the ARP cache timeout on Juniper routers to 4 hours. A recipe for this follows:

```
> configure

Entering configuration mode


[edit]

you@juniper# edit system arp


[edit system arp]

you@juniper# set aging-timer 240
```

```
[edit system arp]

you@juniper# show | compare

[edit system arp]

+ aging-timer 240;


[edit system arp]

you@juniper# commit and-quit

commit complete

Exiting configuration mode
```

Since Junos 9.4 the ARP cache timeout is also configurable on an interface level:

```
[edit system arp aging-timer interface interface-name] aging-timer-minutes;
```

and on more recent versions of Junos that syntax has changed to:

```
[edit system arp interface interface-name] aging-timer aging-timer-minutes;
```

10.3 Juniper Aggregated Link

10.3.1 M-Series

We have encountered no issues with aggregated links and Jun OS (M40, M160, T320). Junos releases prior to 6.0 required VLAN tagging on aggregated interfaces. This limitation has since been removed. An example configuration follows:

```
[edit]

niels@junix# show chassis

aggregated-devices {

  ethernet {

  device-count 1;

  }
```

```
}
---
[edit]
niels@junix# show interfaces ge-2/1/0
gigether-options {
  802.3ad ae0;
}

[edit]
niels@junix# show interfaces ge-3/1/0
gigether-options {
  802.3ad ae0;
}
---
[edit]
niels@junix# show interfaces ae0
description "MN-IX";
unit 0 {
  family inet {
  filter {
  input MNIX-in;
  output MNIX-out;
  }
  address 77.69.248.x/24;
  }
}
```

Additionally and optionally you can configure more granular load balancing:

```
---
routing-options {
  autonomous-system abcde;
  forwarding-table {
  export [ load-balance ];
  }
}
policy-options {
  policy-statement load-balance {
  then {
  load-balance per-packet;
  }
  }
}
forwarding-options {
  hash-key {
  family inet {
  layer-3;
  layer-4;
  }
  }
}
---
```

In case that is not granular enough, you can modify the hash-key algorithm with some undocumented options in Junos OS 7.x and up:

```
---
hash-key {
  family inet {
  layer-3 {
  destination-address;
  protocol;
  source-address;
  }
  layer-4 {
  destination-port;
  source-port;
  type-of-service;
  }
  }
}
---
```

Also, you can set your aggregated min-links to a value that will cause the bundle to drop in the event that your links can no longer support the amount of traffic you plan on shoving down the pipe. Thus, 2-port aggregated link, pushing 1.2 Gbps sustained across, drop bundle if n == 1;

```
---
aggregated-ether-options {
  minimum-links 2;
  link-speed 1g;
}
---
```

In a situation with load-balancing over multiple IP interfaces (not MN-IX), the final statement will make traceroute more confusing to novices as packets may seem to 'bounce' between interfaces by also including TCP/UDP port numbers and ICMP checksums in the algorithm.On an IP1 load-balance per-packet really means per-packet; on an IP2 it actually works per flow, which is preferable.

10.4. Juniper 10GE Specifics

Possible link flap make that you have to damp interface transitions. Junos supports a configurable hold-time . A good value would be 1200 ms.

```
[edit]

arien@router# show interfaces xe-0/1/0

description " interface to MN-IX Peering LAN";

hold-time up 1200 down 1200
```

Aggregated interfaces require hold timers on all physical interfaces and on the logical aggregated interface. Respectively xe-0/1/0 and ae0 in the example below:

```
 [edit]

arien@router# show interfaces xe-0/1/0

description "10GE LinkAgg #1";

hold-time up 1200 down 1200;

gigether-options {

   802.3ad ae0;

}

[edit]

arien@router# show interfaces ae0

description "Aggregated interface to MN-IX Peering LAN";

hold-time up 1200 down 1200;

aggregated-ether-options {

   minimum-links 1;

   link-speed 10g;

}
```

```
unit 0 {
   description "Aggregated interface to MN-IX Peering LAN";
   bandwidth 20g;
   family inet {
       address 77.69.248.x/24;
   }
}
```

10.5 MTU Config

The configured MTU should be 1514 (this includes Ethernet headers but not the FCS), or 1518 when tagged.

11. Linux Configuration Hints

We are not aware of any major issues with Linux boxes used as routers, and they seem to be pretty rare on the Exchange. Having said that, there are a few parameters that can (and usually should) be tuned:

* ARP filtering & source routing

* ARP cache timeout

* Reverse Path (RP) filter

For more information on tuning your Linux system for routing, see the Linux Advanced Routing & Traffic Control HOWTO. NOTE: Please be aware while configuring sysctl parameters, that interface specific entries override global ones. For instance, proxy-arp will be enabled (which is undesirable) if both of these are set:

```
net.ipv4.conf.eth0.proxy_arp = 1
net.ipv4.conf.all.proxy_arp = 0
```

11.1 ARP Filtering and Source Routing

The Linux approach to IP addresses is that they belong to the system, not any single interface. As a result, Linux hosts have a default behaviour that is different from most other systems: interfaces semi-promiscuously answer for all IP addresses of all other interfaces. Example:

image on https://eu-e25b.kxcdn.com/images/_contentBlockGraphicMedium/16309/MN-IX-config-guide_ARP-Filtering-and-Source-Routing.png

In this example, host tuxco is a Linux box with a peering connection on eth0 (192.168.1.1/24) and a backbone link on eth1 (10.0.0.1/24).When host kannix (192.168.1.2) sends an ARP query for 10.0.0.1 it will get a reply from tuxco's eth0 interface!

In other words, a Linux host will answer to ARP queries coming in on any interface if the queried address is configured on any of its interfaces. The idea behind this is that an IP address belongs to the system, not just to a single interface. Although this may work well for server or desktop systems, it is not desirable behaviour in a router system. One reason is that it is a limited version of proxy-arp, which is forbidden on the MN-IX peering LAN. Another reason is that two separate routers could potentially answer ARP queries for the same RFC1918 address.

11.1.1 Fixing ARP

The ARP behaviour can be fixed by using arp_ignore and arp_announce on the WAN interface:

```
tuxco# sysctl -w net.ipv4.conf.

ifname

.arp_ignore=1

tuxco# sysctl -w net.ipv4.conf.

ifname

.arp_announce=1
```

11.1.2 Multiple Interfaces on One Subnet

If you have multiple interfaces on the same subnet, you may also want to enable arp_filter:

This prevents the ARP entry for an interface to fluctuate between two or more MAC addresses. However, you need to use source routing to make this work correctly. From the Documentation/networking/ip-sysctl-2.6.txt file in the kernel source:

[...]

arp_filter - BOOLEAN

1 - Allows you to have multiple network interfaces on the same subnet, and have the ARPs for each interface be answered based on whether or not the kernel would route a packet from the ARP'd IP out that interface (therefore you must use source based routing for this to work). In other words it allows control of which cards (usually 1) will respond to an arp request.

[...]

11.2. IPv4 ARP Cache Timeout

The ARP cache timeout on Linux-based routers should be changed from the default, especially if you have a large number of peers. This parameter can be tuned by setting the appropriate procfs variable through the*sysctl* interface. The Linux arp(7) manual says:

[...]

SYSCTLS

ARP supports a sysctl interface to configure parameters on a global or per-interface basis. The sysctls can be accessed by reading or writing the /proc/sys/net/ipv4/neigh/*/* files or with the *sysctl*(2) interface. Each interface in the system has its own directory in /proc/sys/net/ipv4/neigh/. The setting in the default directory is used for all newly created devices. Unless otherwise specified time related sysctls are specified in seconds.

[...]

base_reachable_time

Once a neighbour has been found, the entry is considered to be valid for at least a random value between base_reachable_time/2 and 3*base_reachable_time/2. An entry's validity will be extended if it receives positive feedback from higher level protocols. Defaults to 30 seconds.

This means that Linux systems keep ARP entries in their cache for some time between 15 and 45 seconds (and yes, the average works out to 3 seconds). This is not very high. In fact, it is lower than the typical BGP keepalive interval and may thus result in excessive ARPs.

We suggest a timeout of at least two hours for ARP entries on your MN-IX interface, so you'd have to set the base_reachable_time to 2 x 2hrs = 4 hours.

```
tuxco1# sysctl net.ipv4.neigh.ifname.base_reachable_time

net.ipv4.neigh.ifname.base_reachable_time = 30
```

The above command tells you that the ARP cache timeout is 30 seconds average. To change it so it's between 2 and 6 hours, use the following command:

```
tuxco1# sysctl -w net.ipv4.neigh.ifname.base_reachable_time=14400

net.ipv4.neigh.ifname.base_reachable_time = 14400
```

Here ifname is the name of the interface that connects to MN-IX. You can also use "default" here, but that may have undesired side-effects for your other interfaces.

11.3 IPv6 Neighbor Cache Timeout

As with the IPv4 ARP cache, Linux systems tend to set the lifetime of the IPv6 neighbor cache quite short as well. The lifetime is controlled in a similar way as for IPv4 ARP.

11.4 Proxy ARP

Disable proxy-arp using sysctl:

sysctl -w net.ipv4.conf.proxy_arp =0

router# sysctl -w net.ipv4.conf.ifname.proxy_arp=0

11.5 IPv6 Autoconfiguration

IPv6 stateless autoconfiguration must be disabled:

```
router# sysctl -w net.ipv6.conf.

ifname

.autoconf=0net.ipv6.conf.ifname.autoconf = 0
```

11.6 RP Filter Setting

You may need to turn off the Reverse Path Filter (rp_filter) functionality on a Linux-based router to allow asymmetric routing, particularly on your WAN interface.To disable the RP filter:

```
tuxco1# sysctl -w
net.ipv4.conf.ifname.rp_filter=0net.ipv4.conf.ifname.rp_filter = 0
```

11.7 Running the 'sysctl' Commands at Boot

The various system parameters discussed above can be set at boot time by adding it to a file such as /etc/sysctl.conf. The exact name, location and very existence of this file typically depends on the Linux distribution in use, but both Debian and Red Hat/Fedora use /etc/sysctl.conf:

```
# file: /etc/sysctl.conf

# These settings should be duplicated for all interfaces that are

# on a peering LAN.


### Typical stuff you really want on a router


# Fix the "promiscuous ARP" thing...
```

```
net.ipv4.conf.ifname.arp_ignore=1

net.ipv4.conf.ifname.arp_announce=1


# Turn off RP filtering to allow asymmetric routing:

net/ipv4/conf/ifname/rp_filter=0


# Multiple (non-aggregated) interfaces on the same peering LAN.

# READ THE MANUAL FIRST!

#net.ipv4.conf.ifname.arp_filter=1


### Keep the MN-IX ARP Police happy. :-)


net.ipv4.neigh.ifname.base_reachable_time=14400

net.ipv6.neigh.ifname.base_reachable_time=14400
```

CAUTION: Modules must be loaded before sysctl is executed

On Debian systems, kernel modules for some network interfaces (e.g. 10GE cards) are not loaded before the init process executes the script thatruns the sysctl commands. In those cases, it is necessary to force the module to be loaded earlier. The same goes for the IPv6 settings; the ipv6 module is usually not loaded until the network interfaces are brought up, which is typically after the sysctl variables are set by the procps.sh script.(On Red Hat/Fedora systems no action needs to be taken; the /etc/init.d/network script automatically (re-)sets the sysctl variables before and after bringing up the interfaces.)There are a few ways around this:

On Debian-based systems, this can be done by creating a symbolic link in /etc/rc2.d to re-run procps.sh after the network is brought up:

root@tuxco# ln -s ../init.d/procps.sh /etc/rc2.d/S20procps.sh

11.8. Linux Aggregated Links

Enable bonding driver support in the kernel (CONFIG_BONDING=m)Edit /etc/modules to load the bonding driver on boot:

```
bonding miimon=100
```

The miimon parameter specifies the frequency for link-monitoring, measured in ms.Install the ifenslave package (apt-get install ifenslave). This package provides the /sbin/ifenslave tool, which is used to attach physical interfaces to the bonding interface.Add the bonding interface to /etc/network/interfaces:

```
# MN-IX side

auto bond0

iface bond0 inet static

  address 77.69.248.x

  netmask 255.255.255.0

  post-up /sbin/ifenslave bond0 eth0 eth1
```

The above example creates a bonding interface with two physical interfaces.For more information see the file Documentation/networking/bonding.txt in the kernel source tree.

11.9 MLDv2

Modern kernels have MLDv2 on by default and there is no sysctl parameter to switch it off. The only known way by now is to drop it with an outgoing filter:

ip6tables -A OUTPUT -p icmpv6 --icmpv6-type 143 -j DROP

ip6tables-save

12. Mikrotik Configuration Hints

By default Mikrotik routers have their own proprietary Mikrotik Discovery Protocol and CDP enabled. To turn these discovery protocols off, in the Web UI go to IP > Neighbors > Discovery Interfaces and disable the protocols on the MN-IX-facing interface.

13. Redback Configuration Hints

To configure link aggregation on Redback SMS routers you need to do the following.

```
!Create the link group interface and assign an IP address to it


[local]Redback(config)#context local

[local]Redback(config-ctx)#interface MN-IX

[local]Redback(config-if)#ip address 77.69.248.x/24
```

```
[local]Redback(config-if)#exit

!Create the link group and bind it to its interface

[local]Redback(config)#link-group MN-IX ether
[local]Redback(config-link-group)#bind interface MN-IX local

!Configure an ethernet port and add it to the link group

[local]Redback(config-config)#port ethernet 1/1
[local]Redback(config-port)#no shutdown
[local]Redback(config-port)#link-group MN-IX
[local]Redback(config-port)#exit

!Configure another ethernet port and add it to the link group

[local]Redback(config-config)#port ethernet 1/2
[local]Redback(config-port)#no shutdown
[local]Redback(config-port)#link-group MN-IX
[local]Redback(config-port)#exit

!To match the MN-IX arp timeout (4 hours) you need to configure this
under the interface

[local]Redback(config)#context local
[local]Redback(config-ctx)#int MN-IX
```

```
[local]Redback(config-if)#ip arp timeout 14400
[local]Redback(config-port)#exit

!Also, you can set your aggregated min-links to a value that will
cause the bundle to drop in the event that your links can no longer
support the amount of traffic you move trough the link-group.Thus, 2-
port aggregated link, pushing 1.2 Gbps sustained across, drop bundle
if n == 1;

[local]Redback(config)#link-group MN-IX ether
[local]Redback(config-link-group)#minimum-links 2
[local]Redback(config-link-group)#exit
```

14. Riverstone Configuration Hints

On Riverstone equipment, proxy ARP seems to be enabled by default, so you will need to disable it:

```
ip disable proxy-arp interface ifname
```

Here, ifname refers to your interface towards MN-IX, or the string 'all'

15. Acknowledgements

Various people contributed to this document. We received configuration info from:

Aaron Weintraub (Cogent Communications)

Adam Davenport (Choopa)

Andree Toonk (SARA)

Andrew V. Zachinyaev (RIPN)

Bart Peirens (Belgacom)

Bas Haakman (Multikabel)

Ben Galliart (Steadfast Networks)

Blake Willis (Neo Telecoms)

Brad Dreisbach (NTT)

Daniel Roesen (ClueNet Project)

Edward Henigin (Giganews)

Elisa Jasinska (Limelight)

Erik Bos (XS4ALL)

Geraint Jones (Koding.com)

Greg Hankins (Force10)

Jesper Skriver (TDC)

Job Snijders (Snijders IT)

Jon Nistor (Rogers/TorIX)

Kevin Day (Your.org)

Lucas van Schouwen (Eweka)

Marcel ten Berg (Scarlet)

Mark Bergsma (Wikimedia Foundation)

Martijn Bakker (Support Net)

Martin Pels (Support Net)

Michiel Bool (Vodafone Netherlands)

Miquel van Smoorenburg (Cistron)

Najam Saquib (Mediaways)

Niels Raijer (Demon)

Paolo Moroni (SWISSCOM)

Pierfrancesco Caci (Telecom Italia Sparkle)

Rene Huizinga (UPC)

Richard A Steenbergen (nLayer)

Robert McKay (MCKAYCOM LTD)

Ronald Esveld (Equant)

Ruediger Volk (Deutsche Telekom)

Santi Mercado (SARENET)

Scott Madley (Level 3 Communications)

Simon Leinen (SWITCH)

Thijs Eilander (Cobweb)

Tom Scholl (SBC)

Vincent Bourgonjen (Open Peering)

Wolfgang Tremmel (DE-CIX)

Thanks to all those who contributed.