IXP Manager Update & Route Server Configuration

Barry O’Donovan

Internet Neutral Exchange Association
Company Limited by Guarantee
(Brief) IXP Manager Update

- Last IXP Manager Update @ 23rd Euro-IX Forum (Oct. 2013)
- That was v3.6.0. Since then:
  - 2048 files changed, 360865 insertions(+), 119213 deletions(-)
- Now on v4.7.3 with v4.8 shortly.
- V4 is also a framework migration
  - Not without consequences when changing huge swaths of code
Laravel

- Modern, hugely popular PHP framework
- #1 web framework on GitHub\(^1\) (across all languages - 41.9k stars)
  - Artisan - console commands
  - Events and queuing
  - Request routing, middleware, validation, CSRF
  - Ease of APIs
- Latest 5.6 but requires PHP 7.1. IXP Manager on 5.5.

Zend Framework (v1)

- End of life Sep 2016 but dates from 2008
- Has been newer versions but complete rewrites and methodology changes
- Looked at v2 but, nggggg…..
- Had a scaffolding engine for simple CRUD (Doctrine2Frontend) which we have re-implemented in Laravel.
- Migration to Laravel >80% done.

1. https://github.com/topics/framework
http://docs.ixpmanager.org/
http://docs.ixpmanager.org/

Type
Free text - may allow you to manage different makes / models / half/full / owned/shared etc.

Height (U)
The height of the rack in standard rack U(nits).

U's Count From
Select an option
Some racks have their U's labelled - please indicate if you count these from top to bottom or from bottom to top.
Route Server History at INEX
[inex-tech] Route server system now in beta

Nick Hilliard  Fri Nov 23 12:20:17 GMT 2007

Following the announcement at the last INEX members meeting that we were looking into running a route server system, we are now pleased to announce that we now have a route server system which is in stable beta.

As a brief summary, the route server system offers the following advantages:

- dramatically reduces the number of BGP sessions required to peer with other INEX members
- strict route filtering on inbound announcements means that only prefixes registered at RIPE by exchange members will be visible
- dual-hosted system offers high reliability
- community based filtering allows route server users to control which INEX members their prefixes are sent to
Designed from the start to be automated, templated and secure

Nick Hilliard is an author of the route server RFCs:

• RFC7947: Internet Exchange BGP Route Server
• RFC7948: Internet Exchange BGP Route Server Operations

This knowledge and experience has been distilled into IXP Manager

Route Server configuration generation overhauled during v4 and now includes looking glass integration, monitoring, new API calls.

More generally: IXP Manager generates configuration for *routers*: route collectors, route servers and AS112 services.
• Secure by design / out of the box
• Strict prefix filtering based on IRRDB entries
  • New option in v4.8.0 to allow *more specifics*
• Origin ASN filtering based on IRRDB entries
• IANA special purpose ranges / *bogons* filtered
• Next hop hijacking detection and filtering
• Max prefix limits
• Standard community filters supported (inc. large communities - RFC8092)
• Looking glass integration, MD5 shared secrets
IXP Manager and Route Server Configuration Generation
Configuring a Route Server via IXP Manager

1. Gather the required information for a route server:
   • AS number (should be a dedicated ASN for route server)
   • Peering IP and BGP router ID (usually the peering IPv4 address)
   • BGP software (IXP Manager supports Bird but others possible)
   • API type and looking glass software
   • Support MD5? Support large BGP communities?
1. Gather the required information for a route server:
   • AS number (should be a dedicated ASN for route server)
   • Peering IP and BGP router ID (usually the peering IPv4 address)
   • BGP software (IXP Manager supports Bird but others possible)
   • API type and looking glass software
   • Support MD5? Support large BGP communities?

2. Add this data into IXP Manager via UI
<table>
<thead>
<tr>
<th><strong>Handle</strong></th>
<th>rs1-lan1-ipv4</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Vlan</strong></td>
<td>Peering VLAN #1</td>
</tr>
<tr>
<td><strong>Protocol</strong></td>
<td>IPv4</td>
</tr>
<tr>
<td><strong>Type</strong></td>
<td>Route Server</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>Route Server #1 - LAN1 - IPv4</td>
</tr>
<tr>
<td><strong>ShortName</strong></td>
<td>RS1 - LAN1 - IPv4</td>
</tr>
<tr>
<td><strong>Router ID</strong></td>
<td>192.0.2.8</td>
</tr>
<tr>
<td><strong>Peering IP</strong></td>
<td>192.0.2.8</td>
</tr>
<tr>
<td><strong>ASN</strong></td>
<td>65501</td>
</tr>
<tr>
<td>Configuration Item</td>
<td>Value</td>
</tr>
<tr>
<td>-------------------------</td>
<td>--------------------------------------------</td>
</tr>
<tr>
<td>Management Host</td>
<td>2001:db8::3:8</td>
</tr>
<tr>
<td>API Type</td>
<td>Birdseye</td>
</tr>
<tr>
<td>API Endpoint</td>
<td><a href="https://rs1-lan1-ipv4.mgmt.example.com/api">https://rs1-lan1-ipv4.mgmt.example.com/api</a></td>
</tr>
<tr>
<td>LG Access Privileges</td>
<td>CUSTUSER</td>
</tr>
<tr>
<td>Quarantine</td>
<td>Router will be used for quarantine procedures only</td>
</tr>
<tr>
<td>BGP LC</td>
<td>Enable Large BGP Communities / RFC8092</td>
</tr>
<tr>
<td>Skip MD5</td>
<td>Do not include any MD5 configuration</td>
</tr>
<tr>
<td>Template</td>
<td>api/v4/router/server/bird/standard</td>
</tr>
</tbody>
</table>
Configuring a Route Server via IXP Manager

1. Gather the required information for a route server:
   - AS number (should be a dedicated ASN for route server)
   - Peering IP and BGP router ID (usually the peering IPv4 address)
   - BGP software (IXP Manager supports Bird but others possible)
   - API type and looking glass software
   - Support MD5? Support large BGP communities?

2. Add this data into IXP Manager via UI

3. Enable route server clients for your members
Configuring a Member as a Route Server Client

- Vlan: Peering VLAN #1
- Max BGP Prefixes: 50
- Multicast Enabled
- Busy host
- IPv6 Enabled
- IPv4 Enabled
- Route Server Client
- Apply IRRDB Filtering
- IRRDB - Allow More Specifics
- AS112 Client
Configuring a Route Server via IXP Manager

1. Gather the required information for a route server:
   - AS number (should be a dedicated ASN for route server)
   - Peering IP and BGP router ID (usually the peering IPv4 address)
   - BGP software (IXP Manager supports Bird but others possible)
   - API type and looking glass software
   - Support MD5? Support large BGP communities?

2. Add this data into IXP Manager via UI

3. Enable route server clients for your members

4. Ensure cron job for IRRDB database updates is running
Configuring a Route Server via IXP Manager

1. Gather the required information for a route server:
   - AS number (should be a dedicated ASN for route server)
   - Peering IP and BGP router ID (usually the peering IPv4 address)
   - BGP software (IXP Manager supports Bird but others possible)
   - API type and looking glass software
   - Support MD5? Support large BGP communities?

2. Add this data into IXP Manager via UI

3. Enable route server clients for your members

4. Ensure cron job for IRRDB database updates is running

5. Pull the route server configuration via API
Configuring a Route Server via IXP Manager

curl -X GET

   -H "X-IXP-Manager-API-Key: my-api-key"

https://ixp.example.com/api/v4/router/gen-config/rs1-lan1-ipv4

> /etc/bird/rs1-lan1-ipv4.conf

bird -s /var/run/rs1-lan1-ipv4.ctl -c /etc/bird/rs1-lan1-ipv4.conf

https://github.com/inex/IXP-Manager/tree/master/tools/runtime/route-servers
IRRDB Filtering

EURO-IX 32
FEATURES SUPPORTED

- Secure by design / out of the box
- Strict prefix filtering based on IRRDB entries
  - New option in v4.8.0 to allow *more specifics*
- Origin ASN filtering based on IRRDB entries
  - IANA special purpose ranges / *bogons* filtered
  - Next hop hijacking detection and filtering
  - Max prefix limits
- Standard community filters supported (inc. large communities - RFC8092)
- Looking glass integration, MD5 shared secrets
Sample IRRDB entries

$ whois 192.0.2.1
...
route:    192.0.2.0/24
descr:    Packet Loss Ltd
origin:   AS65501
mnt-by:   JOE-MNT
source:   RIPE

$ whois 2001:db8::1
...
route:    2001:db8::/32
descr:    Packet Loss Ltd
origin:   AS65501
mnt-by:   JOE-MNT
source:   RIPE
• Local database storage of members’ IRRDB entries (via bgpq3)
• Flexible configuration of IRRDB source database(s) on a per member basis
• Support for both AS sets and *just* ASNs
• Updating IRRDB database is an asynchronous operation to generating route server configuration
IRRDB Filtering in IXP Manager

- Very efficient when using the appropriate data structures (php-ds)

./artisan irrdb:update-prefix-db hurricane -vv

Hurricane Electric:
- [IPv4: 1029706 total; 0 stale; 0 new; DB updated]
- [IPv6: 182593 total; 0 stale; 0 new; DB updated]

Time for net/database/processing: 55.92/22.00/2.65 (secs)
IRRDB Filtering in IXP Manager

- Very efficient when using the appropriate data structures (php-ds)
- A full run for INEX takes ~3mins

`./artisan irrdb:update-prefix-db -v`

Total time for net/database/processing: 138.745264/29.177696/2.913495 (secs)
**IRRDB Filtering in IXP Manager**

- Route server prefix analysis tool:

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Protocol</th>
<th>First Seen</th>
<th>Origin AS</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.231.21.0/24</td>
<td>IPv4</td>
<td>2017-07-16 10:18:18</td>
<td>AS47615</td>
</tr>
<tr>
<td>87.232.5.0/24</td>
<td>IPv4</td>
<td>2017-06-02 10:18:19</td>
<td>AS25441</td>
</tr>
</tbody>
</table>
Bird’s Eye - A Simple Secure Micro Service for Querying Bird
Bird’s Eye

- Winning project from the RIPE IXP Tools Hackathon @ RIPE73 (Madrid, 2016)
- The Alice-LG started as a client querying Bird’s Eye
- Provides API access to query Bird
- Designed with security as a primary goal
- Integrated with IXP Manager for a built-in looking glass
  - Coded to a contract / interface as a Laravel service provider
- Also has its own built in looking glass
- RFC8092 / BGP Large Communities support
Bird’s Eye Features

- Bird (and Bird’s Eye) daemon status and version details
- Tabular view of ‘show bgp summary’ with:
  - click to view the routing table
  - click to view routes received
  - click to view routes advertised
- Session information as available in Bird (router ID, neighbour capabilities, input and output filters, max prefixes, stats, etc.)
- Route lookup by routing table / protocol
IXP Manager Looking Glass

- Production example: [https://www.inex.ie/ixp/lg](https://www.inex.ie/ixp/lg)
  - Provides visibility of 6 x AS112 / 6 x route collectors / 12 x route servers
- Does not expose Bird’s Eye to the public - IXP Manager queries Bird’s Eye internally and formats and presents that information to the user
- Bird status and ‘show bgp summary’ also available as JSON
- IXP Manager also generates Nagios snippets to:
  - Monitor Bird daemons
  - Monitor member BGP sessions (great for maintenance)
Support Additional BGP Daemons

• Requires feature parity with Bird implementation
  • IRRDB filtering
  • Ideally suitable for route servers, route collectors and AS112 services
  • API access and service provider for looking glass functionality
  • Standard community filtering (plus ideally large communities)
• Want to look at GoBGP and have committed to implementing OpenBGPd
• A routeserver can consume IX-F Member Export to generate route server config
  • But: not compatible with IXP Manager features including prefix analysis tool, looking glasses, monitoring, etc.
New Features

- Per-member or longer prefix filtering
- RPKI integration
- IXP-side community tagging to indicate the reason a prefix was blocked
  - These exist only within the route server daemon
  - Will then replace the existing route server prefix analysis tool
- UI controls to block advertising / receipt of your prefixes to / from other members and to allow you to prepend your advertisements to other networks
- Filter known transit networks
- BGP Graceful shutdown support?
Any Questions?

barry.odonovan@inex.ie

https://www.inex.ie/

https://www.ixpmanager.org/