Bandwidth-based load-balancing with failover. The easy way.

We need more bandwidth.
Presenter information

Tomas Kirnak

Network design
Security, wireless
Servers, Virtualization

Mikrotik Certified Trainer

Atris, Slovakia

Established 1991
Complete IT solutions
Networking, servers
Virtualization
IP security systems
Load-balancing, why?

• Distributing workload to multiple network links to maximize throughput and minimize latency.

• Using multiple network links, when properly configured, will also provide redundancy.
Load balancing types

• Bonding
• Policy routing
• PCC
• Bandwidth based
Load balancing types

Bonding - 802.3ad LACP
Bonding

+ Easy to implement
  Automatic redundancy with fail-over

- You need to control of both ends of the link
Load balancing types

Policy routing
Policy routing

+ Easy to implement
  You have exact control of traffic

- Not dynamic
  Scalability problems
Load balancing types

PCC
per connection classifier
PCC

+ Easy to configure
  Good scalability

- Not aware of link state (bandwidth wise)
  Not so great with very un-similiar links (4:1)
Load balancing types

For presentations on these load-balancing methods, please see

Load balancing types

Bandwidth based

If interface ISP1 is over 10 mbit/s; use ISP2
Why use bandwidth-based LB

+ Easily scalable
+ Takes link status into consideration
+ You have control over the connections
+ You decide when the switch to second link happens (on 10mbit link, switch after 50% util.)

- Comes with its own problems
Implementation considerations

• There are multiple ways to do bandwidth based load balancing, neither is so easy.

• MPLS TE
• Mangle + bit of scripting  <-- this presentation

Underlying technologies
Connections and tracking them

<table>
<thead>
<tr>
<th>Src. Address</th>
<th>Dist. Address</th>
<th>Protocol</th>
<th>Connect1</th>
<th>Connect2</th>
<th>P2P</th>
<th>Timeout</th>
<th>TCP State</th>
</tr>
</thead>
<tbody>
<tr>
<td>192.168.2.110:40244</td>
<td>199.167.177.38:12373</td>
<td>6 (tcp)</td>
<td>23:51:59</td>
<td>established</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.168.2.110:47716</td>
<td>173.194.70.188:52226</td>
<td>6 (tcp)</td>
<td>23:39:53</td>
<td>established</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.168.2.111:45823</td>
<td>64.141.204:047</td>
<td>6 (tcp)</td>
<td>23:54:19</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.168.2.111:45833</td>
<td>65.55.71.73:443</td>
<td>6 (tcp)</td>
<td>23:53:59</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>192.168.2.111:45834</td>
<td>78.141.111:250</td>
<td>6 (tcp)</td>
<td>23:45:33</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TCP Syn Send Timeout: 00:00:05
TCP Syn Received Timeout: 00:00:05
TCP Established Timeout: 1d 00:00:00
TCP Fin Wait Timeout: 00:00:10
TCP Close Wait Timeout: 00:00:10
TCP Last Ack Timeout: 00:00:10
TCP Time Wait: 00:00:10
TCP Close: 00:00:10
UDP Timeout: 00:00:10
UDP Stream Timeout: 00:03:00
ICMP Timeout: 00:00:10
Generic Timeout: 00:10:00
What is a connection

• We can define a connection as a packet flow with the same pair of source and destination IP addresses and ports.

• In case of UDP, this is would be an UDP stream.

• 192.168.2.10:49481 <-> 8.8.8.8:53
Mangle

• Mangle is a facility in ROS which allows us to “mark” packets or connections, and later use that mark for our purposes.

• Mangle marks do NOT leave the router.
Mangle – where to

/ip
firewall
mangle
Routing tables

• A routing table tells the router which next hop to forward packets to, depending on the packets destination IP.

• 0.0.0.0/0 -> 77.21.34.12
Routing tables – part 2

• By default all packets are put into the “main” routing table

• We can create our own routing tables, and force packets to use them.
Topology
Required steps

• Create routing tables
• Setup address-lists
• Setup mangle
• Configure Traffic Monitor
Basic configuration

/interface ethernet
  set 0 name=LAN
  set 3 name=ISP_1
  set 4 name=ISP_2

/ip address
  add address=192.168.22.1/24 interface=LAN
  add address=1.1.1.32/24 interface=ISP_1
  add address=2.2.2.65/24 interface=ISP_2

/ip firewall nat
  add action=masquerade chain=srcnat out-interface=ISP_1
  add action=masquerade chain=srcnat out-interface=ISP_2
Routing tables

/ip route

add gateway=1.1.1.1 distance=1
add gateway=2.2.2.1 distance=2

add gateway=1.1.1.1 routing-mark=ISP1_Route distance=1
add gateway=2.2.2.1 routing-mark=ISP2_Route distance=1
Routing tables - GUI

<table>
<thead>
<tr>
<th>Dst. Address</th>
<th>Gateway</th>
<th>Distance</th>
<th>Routing Mark</th>
<th>Pref. Source</th>
</tr>
</thead>
<tbody>
<tr>
<td>AS 0.0.0.0/0</td>
<td>1.1.1.1 reachable ISP_1</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>S 0.0.0.0/0</td>
<td>2.2.2.1 reachable ISP_2</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AS 0.0.0.0/0</td>
<td>1.1.1.1 reachable ISP_1</td>
<td>1</td>
<td>ISP1_Route</td>
<td></td>
</tr>
<tr>
<td>AS 0.0.0.0/0</td>
<td>2.2.2.1 reachable ISP_2</td>
<td>1</td>
<td>ISP2_Route</td>
<td></td>
</tr>
<tr>
<td>DAC 1.1.1.0/24</td>
<td>ISP_1 reachable</td>
<td>0</td>
<td></td>
<td>1.1.1.32</td>
</tr>
<tr>
<td>DAC 2.2.2.0/24</td>
<td>ISP_2 reachable</td>
<td>0</td>
<td></td>
<td>2.2.2.65</td>
</tr>
<tr>
<td>DAC 192.168.22.0/...</td>
<td>LAN reachable</td>
<td>0</td>
<td></td>
<td>192.168.22.1</td>
</tr>
</tbody>
</table>
Traffic to connected networks

- Connected networks are only in the “main” routing table
- We need to make sure that traffic to these networks stays in the main routing table.
Connected networks – part 2

/ip firewall address-list
  add address=1.1.1.0/24 list=Connected
  add address=2.2.2.0/24 list=Connected
  add address=192.168.22.0/24 list=Connected
  add address=192.168.22.0/24 list=LAN

/ip firewall mangle
  add chain=prerouting src-address-list=Connected
dst-address-list=Connected action=accept
• In this topology, there are 4 possible traffic flows

- WAN -> Router
- Router -> WAN
- WAN -> LAN
- LAN -> WAN
Taking care of incoming connections

• When a connection is initiated from the internet through one of the ISPs we need to ensure that this connection is replied through the same ISP (from the same public IP)

• We need to mark these connections, and then put them in the proper routing table.
Router marking – WAN -> Router

• Catch the connection from internet to the router, and mark them.

/ip firewall mangle
add chain=input connection-mark=no-mark in-interface=ISP_1
   action=mark-connection new-connection-mark=WAN1->ROS

add chain=input connection-mark=no-mark in-interface=ISP_2
   action=mark-connection new-connection-mark=WAN2->ROS
Router marking – WAN -> Router

• Then put these connections into the proper routing tables.

```
add chain=output connection-mark=WAN1->ROS
action=mark-routing new-routing-mark=ISP1_Route
```

```
add chain=output connection-mark=WAN2->ROS
action=mark-routing new-routing-mark=ISP2_Route
```
Taking care of the LAN

• Same principle applies to the LAN.

• Connections initiated from the internet through one ISP, should be replied to through the same ISP.
LAN marking

/ip firewall mangle

add chain=forward connection-mark=no-mark in-interface=ISP_1
  action=mark-connection new-connection-mark=WAN1->LANs

add chain=forward connection-mark=no-mark in-interface=ISP_2
  action=mark-connection new-connection-mark=WAN2->LANs

add chain=prerouting connection-mark=WAN1->LANs src-address-list=LAN
  action=mark-routing new-routing-mark=ISP1_Route

add chain=prerouting connection-mark=WAN2->LANs src-address-list=LAN
  action=mark-routing new-routing-mark=ISP2_Route
Incoming connections - done

• We have ensured that when a connection from the internet to our router, or services inside of our network is established, it works.
LAN – partially done

• Connections from the internet to our LAN will now work through both ISPs

• So what about connections outgoing from our LAN to the internet?

• These we actually want to load-balance.
A sticky connection

• A sticky connection is a connection, that once established through one interface, will always go out that exact interface.

• This is required, because when we switch to a second link, we only need to switch new connections.

• In PCC, this is done automatically. Using our approach however, this has to be done manually.
LAN -> WAN mangle

/ip firewall mangle

add chain=prerouting connection-mark=no-mark src-address-list=LAN dst-address-list=!Connected dst-address-type=!local action=mark-connection
new-connection-mark=LAN->WAN

add chain=prerouting connection-mark=LAN->WAN src-address-list=LAN
action=mark-routing new-routing-mark=ISP1_Route
comment="Load-Balancing here"

- Configuring this, we can now manually influence which routing table will our connection from LAN to the internet take.
Sticky connections

add chain=prerouting connection-mark=LAN->WAN routing-mark=ISP1_Route action=mark-connection new-connection-mark=Sticky_ISP1
add chain=prerouting connection-mark=LAN->WAN routing-mark=ISP2_Route action=mark-connection new-connection-mark=Sticky_ISP2

add chain=prerouting connection-mark=Sticky_ISP1 src-address-list=LAN action=mark-routing new-routing-mark=ISP1_Route
add chain=prerouting connection-mark=Sticky_ISP2 src-address-list=LAN action=mark-routing new-routing-mark=ISP2_Route

• This will assure that once a connection is routed through one ISP, it will stay there no matter what.
<table>
<thead>
<tr>
<th>#</th>
<th>Action</th>
<th>Chain</th>
<th>In. Interface</th>
<th>Routing Mark</th>
<th>Src. Address</th>
<th>Dest. Address</th>
<th>New Connection Mark</th>
<th>New Routing Mark</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>accept</td>
<td>prerouting</td>
<td></td>
<td>Connected</td>
<td>Connected</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>mark connection</td>
<td>input</td>
<td>ISP_1</td>
<td>no-mark</td>
<td>WAN1→ROS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>mark connection</td>
<td>input</td>
<td>ISP_2</td>
<td>no-mark</td>
<td>WAN2→ROS</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>mark routing</td>
<td>output</td>
<td></td>
<td>WAN1→ROS</td>
<td></td>
<td>ISP1_Route</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>mark routing</td>
<td>output</td>
<td></td>
<td>WAN2→ROS</td>
<td></td>
<td>ISP2_Route</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>mark connection</td>
<td>forward</td>
<td>ISP_1</td>
<td>no-mark</td>
<td>WAN1→LANs</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>mark connection</td>
<td>forward</td>
<td>ISP_2</td>
<td>no-mark</td>
<td>WAN2→LANs</td>
<td></td>
<td>ISP1_Route</td>
<td>ISP2_Route</td>
</tr>
<tr>
<td>7</td>
<td>mark routing</td>
<td>prerouting</td>
<td></td>
<td>WAN1→LANs</td>
<td>LAN</td>
<td></td>
<td>ISP1_Route</td>
<td>ISP2_Route</td>
</tr>
<tr>
<td>8</td>
<td>mark routing</td>
<td>prerouting</td>
<td></td>
<td>WAN2→LANs</td>
<td>LAN</td>
<td></td>
<td>ISP1_Route</td>
<td>ISP2_Route</td>
</tr>
<tr>
<td>9</td>
<td>mark connection</td>
<td>prerouting</td>
<td></td>
<td>no-mark</td>
<td>LAN</td>
<td>!Connected</td>
<td>LAN→WAN</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>mark routing</td>
<td>prerouting</td>
<td></td>
<td>LAN→WAN</td>
<td>LAN</td>
<td></td>
<td>ISP1_Route</td>
<td>ISP2_Route</td>
</tr>
<tr>
<td>11</td>
<td>mark connection</td>
<td>prerouting</td>
<td></td>
<td>LAN→WAN</td>
<td>ISP1_Route</td>
<td>Sticky_ISP1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>mark connection</td>
<td>prerouting</td>
<td></td>
<td>LAN→WAN</td>
<td>ISP2_Route</td>
<td>Sticky_ISP2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>mark routing</td>
<td>prerouting</td>
<td></td>
<td>Sticky_ISP1</td>
<td>LAN</td>
<td></td>
<td>ISP1_Route</td>
<td>ISP2_Route</td>
</tr>
<tr>
<td>14</td>
<td>mark routing</td>
<td>prerouting</td>
<td></td>
<td>Sticky_ISP2</td>
<td>LAN</td>
<td></td>
<td>ISP1_Route</td>
<td>ISP2_Route</td>
</tr>
</tbody>
</table>

15 items
What’s the final result?

• We can load balancing manually

• Connections go out ISP1, then we can switch the mangle rule to ISP2, but connections already using ISP1 will stay there.
Automating based on bandwidth
Switching back

Traffic Monitor <LB2>

Name: LB2
Interface: ISP_1
Traffic: received
Trigger: below
Threshold: 5242880

On Event:
- log warning "LB Debug: ISP1 back to normal"
- /p firewall mangle set [find comment="Load-Balancing here"] new-routing-mark=ISP1_Route

enabled
Final result

• Connections routed through ISP1, until its link is at 5mbit/s.

• After this limit all new connections will go through ISP2 until the ISP1 link is under its limit.

• Automated, bandwidth-based load balancing.
Easy Failover

- If the gateway can’t be pinged, all routes using this gateway will become invalid.
A different approach

• This approach will not work if the link failure happens after the gateway.

• Recursive route lookup, netwatch etc.

• http://wiki.mikrotik.com/wiki/Failover_Scripting
Thanks for listening

Tomas Kirnak

t.kirnak@atris.sk
Find me after the presentation for any questions.