Background

What we do, Technical goals, Infrastructure
What we do

• Greek VoIP services provider
  • Voice Termination/Origination
  • Virtual PBX as a service

• IT services
  • Virtualization
  • Network infrastructure design and installation

Nikos Nikalexis, nikalexis@modulus.gr
Our Technical Goals

• No single point of failure (SPOF) for services, including
  • Datacenter
  • Power
  • Network
  • Servers / Services

• High Availability > 99.995%

• Low network latency

• Implementation based on the latest technologies available
Our network

The Internet

- DC1 - Lamda Hellix Datacentre
- DC2 - Planned
- modulus HQ

Nikos Nikalexis, nikalexis@modulus.gr
Our Lamda Hellix Infrastructure

- Protected Power Line (A+B)
- 2x Upstream ISP Connections
- 1x GR-IX Connection
- 2x Mikrotik RB1100AH Routers
- 2x Dell 62xx Series Stackable Switches
- 6x Servers
HA Mikrotik-Based Router Infrastructure

Router 1

Router 2

Nikos Nikalexis, nikalexis@modulus.gr
HA Mikrotik-Based Router Infrastructure

Router 1

Router 2

eBGP

Upstream ISP
HA Mikrotik-Based Router Infrastructure

Router 1
Router 2

Upstream ISP

Stackable Switch 1
Stackable Switch 2

Switch Stack

IBGP
OSPF

eBGP

eBGP

Nikos Nikalexis, nikalexis@modulus.gr
Nikos Nikalexis, nikalexis@modulus.gr
HA Mikrotik-Based Router Infrastructure

Router 1
Router 2

IBGP
OSPF

LACP

Switch Stack

Stackable Switch 1
Stackable Switch 2

eBGP

Cluster(s) of Servers

Upstream ISP
Mikrotik RouterOS setup

Interfaces, Bonding, VRRP, IP Addresses, Dynamic Routing, Traffic flow, Configuration Synchronization, Automatic Backup

Nikos Nikalexis, nikalexis@modulus.gr
Bonding
VRRP (1/4)

Nikos Nikalexis, nikalexis@modulus.gr
VRRP (2/4)

• Features
  • Automatic Master / Backup mode
  • Optional preemption mode

• Pros
  • Easy configuration
  • Small transition time (a few seconds)

• Cons
  • Needs a separate IP for each router
  • Plus one for the Virtual IP (gateway)

• Summary
  • For every LAN with two redundant routers, 5 IPs are wasted:
    • Network, Broadcast, Virtual IP, 2x Router IPs
  • For large subnets (> /26), this is not a big problem
  • Considering recent IPv4 space exhaustion, we had to seek a smarter solution
A solution hidden in RouterOS!!!
Undocumented but working

• Setup only one VRRP interface (in private space?)
• Set this interface as a child for your VLANs
• When VRRP is in MASTER mode:
  • Every child VLAN is RUNNING
  • IP addresses on that VLAN interface are ACTIVE
• When VRRP is in BACKUP mode:
  • Every child VLAN is DOWN
  • IP addresses on that VLAN interface are INVALID
VRRP (4/4)
# Interfaces overview

<table>
<thead>
<tr>
<th>Interface List</th>
</tr>
</thead>
<tbody>
<tr>
<td>Interface</td>
</tr>
<tr>
<td>+---------+---------+-------------+------------+-----------+-------+-------+---------+-----</td>
</tr>
<tr>
<td>Name</td>
</tr>
<tr>
<td>R</td>
</tr>
<tr>
<td>bonding1-switch</td>
</tr>
<tr>
<td>RM</td>
</tr>
<tr>
<td>vlan64</td>
</tr>
<tr>
<td>vlan72</td>
</tr>
<tr>
<td>vlan80</td>
</tr>
<tr>
<td>vlan132</td>
</tr>
<tr>
<td>vlan133</td>
</tr>
<tr>
<td>vlan126</td>
</tr>
<tr>
<td>vlan131</td>
</tr>
<tr>
<td>vlan132</td>
</tr>
<tr>
<td>vlan3003-gnix</td>
</tr>
<tr>
<td>R</td>
</tr>
<tr>
<td>ether1-routers</td>
</tr>
<tr>
<td>ether2</td>
</tr>
<tr>
<td>ether3</td>
</tr>
<tr>
<td>ether4</td>
</tr>
<tr>
<td>ether5</td>
</tr>
<tr>
<td>ether6</td>
</tr>
<tr>
<td>ether7</td>
</tr>
<tr>
<td>ether8</td>
</tr>
<tr>
<td>ether9</td>
</tr>
<tr>
<td>ether10</td>
</tr>
<tr>
<td>R</td>
</tr>
<tr>
<td>ether11-gnix</td>
</tr>
<tr>
<td>RS</td>
</tr>
<tr>
<td>ether12-switch-6248</td>
</tr>
<tr>
<td>RS</td>
</tr>
<tr>
<td>ether13-switch-6224</td>
</tr>
</tbody>
</table>
IP addresses overview

MASTER Router

BACKUP Router

Nikos Nikalexis, nikalexis@modulus.gr
Dynamic Routing

• eBGP
  • Upstream eBGP for each Router
    • Connect each upstream link directly with each router
    • You don't lose access to your routers in a case of a hardware/software failure
    • This way, we avoid using a switch device for upstream connectivity
  • GR-IX eBGP through a VLAN configured on the Switch Stack
    • This is not an upstream interconnection, we can afford losing it

• iBGP / OSPF
  • Activated on both routers

• BFD with each peer (RFC 5880)
  • Rapid fault detection (< 1 second)
Traffic Flow

HA Mikrotik-Based Router Infrastructure

Switch Stack

Cluster(s) of Servers

Upstream ISP

eBGP

Nikos Nikalexis, nikalexis@modulus.gr
VRRP Scripting

• On MASTER:
  
  `{ 
    /routing filter set [find chain="providers-out" action="passthrough" set-bgp-med=200] set-bgp-med=100;
  }

• On BACKUP:
  
  `{ 
    /routing filter set [find chain="providers-out" action="passthrough" set-bgp-med=100] set-bgp-med=200;
  }
Traffic Flow (before scripting)
Traffic Flow Fixed (after scripting)
Configuration Synchronization (1/3)

• Our 2 Routers have:
  • Shared config
    • Interfaces
    • VLAN IP addresses
    • Firewall rules
    • QoS rules
    • Routing filters
  • Discreet config
    • VRRP Priority option
    • Non-VLAN IP addresses
    • Upstream eBGP configs
Configuration Synchronization (2/3)

• Develop a python script that:
  • Connects to each router through SSH
  • Exports the full config
  • Calculates diffs between configs and...
  • Sends it in an e-mail to the admin team

• Run this script
  • Periodically to be up to date
  • Manually to check your setup on demand
Configuration Synchronization (3/3)

Nikos Nikalexis, nikalexis@modulus.gr
Automatic Backup

• Develop a python script that:
  • Connects to each router through SSH
  • Exports the full config
  • Creates a backup file
  • Transfers the backup file to a safe location via FTP

• Run this script
  • Periodically, e.g. every 2 days
  • We schedule different days for each router
    • Avoid bugs in export and backup
Conclusion

Testing, Goals achieved, ToDo, Feature Requests
Testing

• MASTER router failure
  • <3 seconds downtime until BACKUP router takeover
  • <1 second downtime until BFD marks our peer as down

• BACKUP router failure
  • No downtime

• MASTER switch failure
  • <10 seconds downtime on some sessions until LACP recovers on backplane

• BACKUP switch failure
  • No downtime
Goals achieved

• No SPOF Network
• Network High Availability
• Configuration Synchronization
• Configuration Backup with Easy Restoration
• Low cost, commodity hardware
**ToDo**

- Use Ansible
  - Centrally manage all HA routers & more...
  - Store all configuration data in the Ansible inventory
    - Use group variables for common config
    - Use host variables for discreet config
    - Use GIT for keeping track of changes
  - Write a module talking to RouterOS API
  - Write roles for master / backup configurations
  - Write playbook for deploying HA router infrastructure
- Upgrade to CCR
  - More powerful
  - Redundant Power Supply
  - Supports SFP interfaces
Feature Requests

• Hardware
  • No SPOF / Single Unit Fully Redundant Router
  • 2xPSU, 2xBackplanes, 2xLinecards
  • Stackable switches

• Software
  • Configuration Synchronization
  • Single interface point (winbox, console, api etc)
  • Connection tracking synchronization (like linux conntrackd) to achieve:
    • Connection-based firewall rules
    • NATed connections

Nikos Nikalexis, nikalexis@modulus.gr
Thank you!

Any Questions?