modulus

Nikos Nikalexis, nikalexis@modulus.gr

Background

What we do, Technical goals, Infrastructure

modulus

What we do

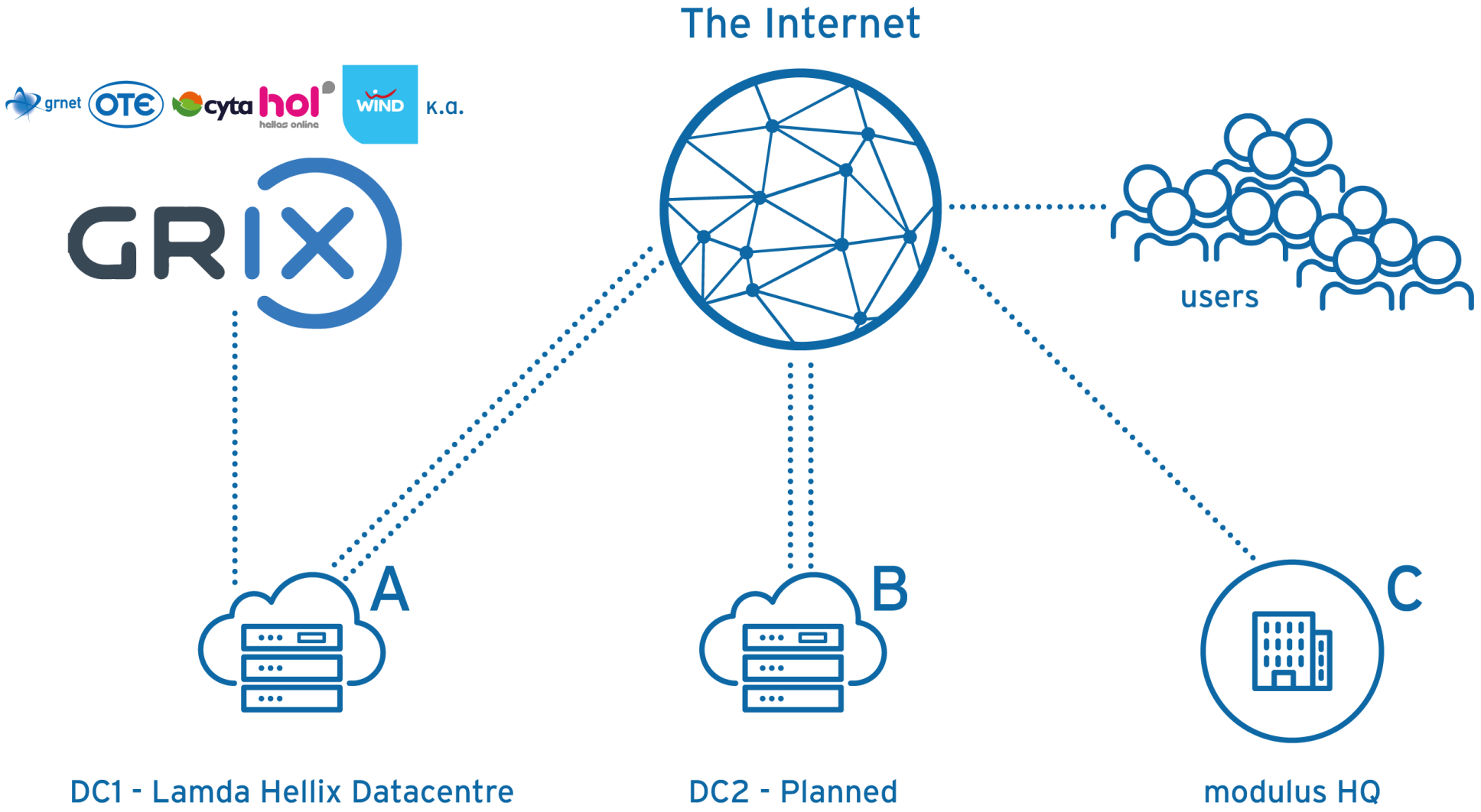
- Greek VoIP services provider
  - Voice Termination/Origination
  - Virtual PBX as a service
- IT services
  - Virtualization
  - Network infrastructure design and installation

modulus

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

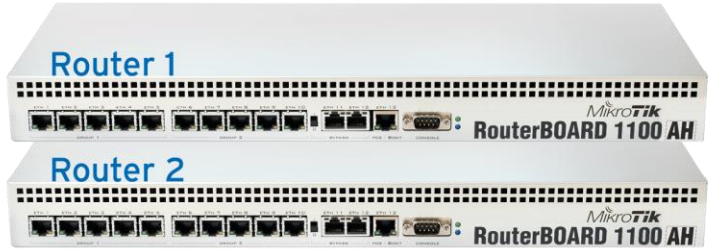


modulus

Our Lamda Hellix Infrastructure

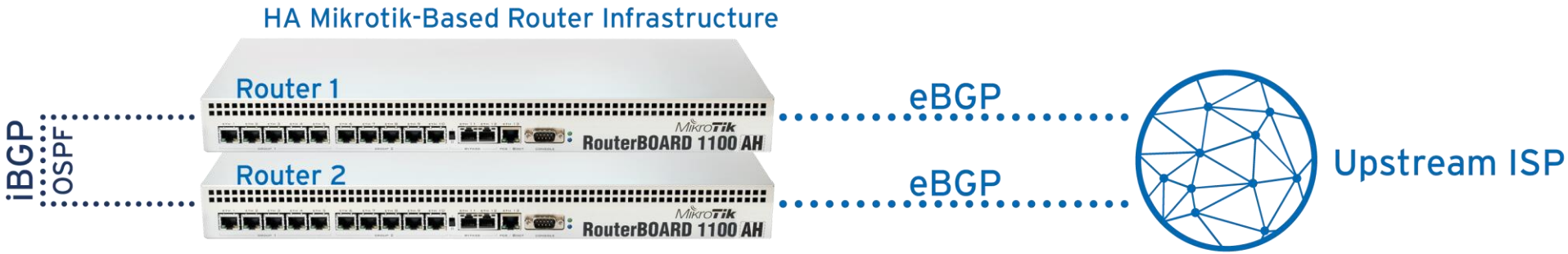
- Protected Power Line (A+B)
- 2x Upstream ISP Connections
- 1x GR-IX Connection
- 2x Mikrotik RB1100AH Routers
- 2x Dell 62xx Series Stackable Switches
- 6x Servers

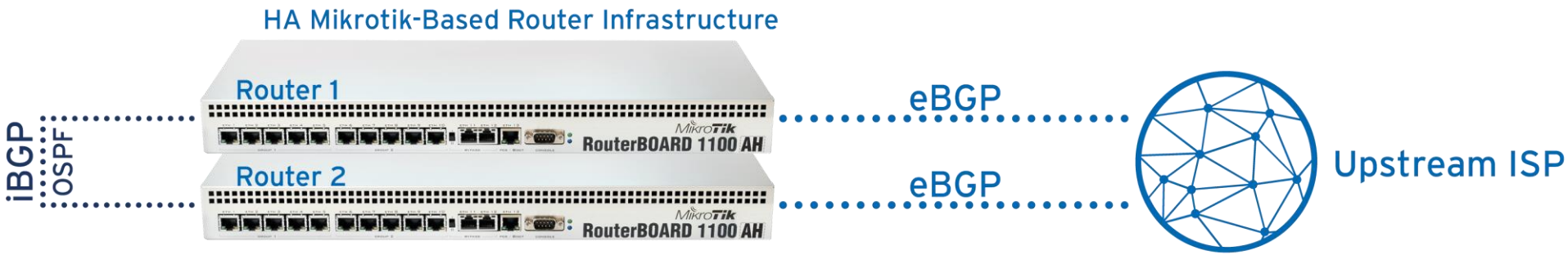
HA Mikrotik-Based Router Infrastructure

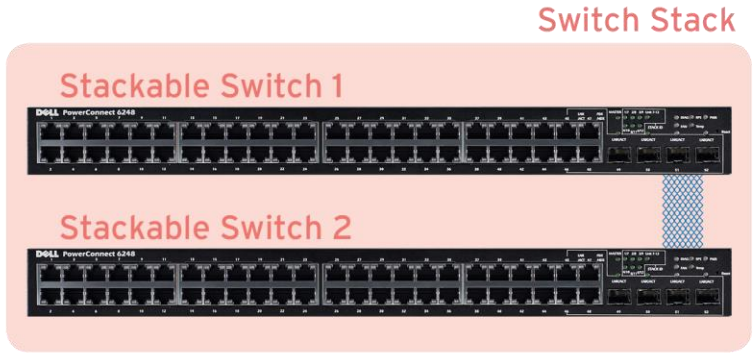


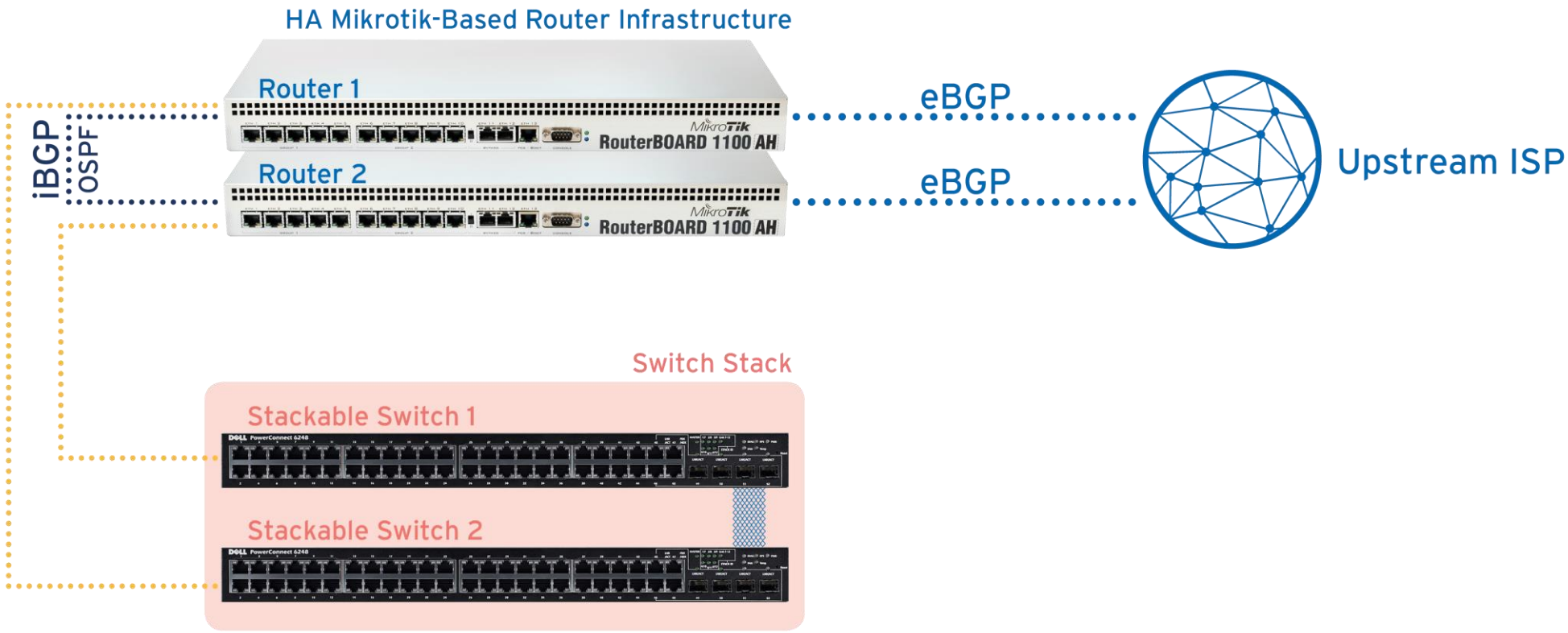
modulus

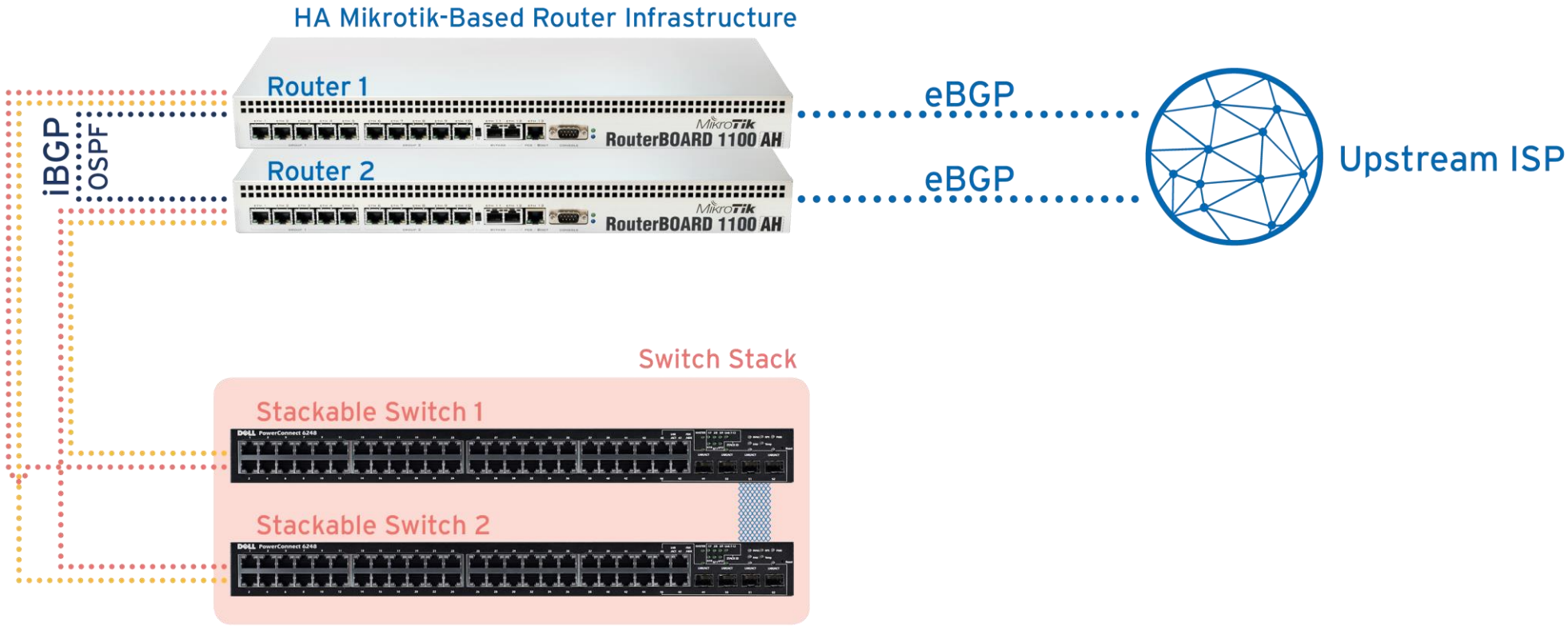
HA Mikrotik-Based Router Infrastructure

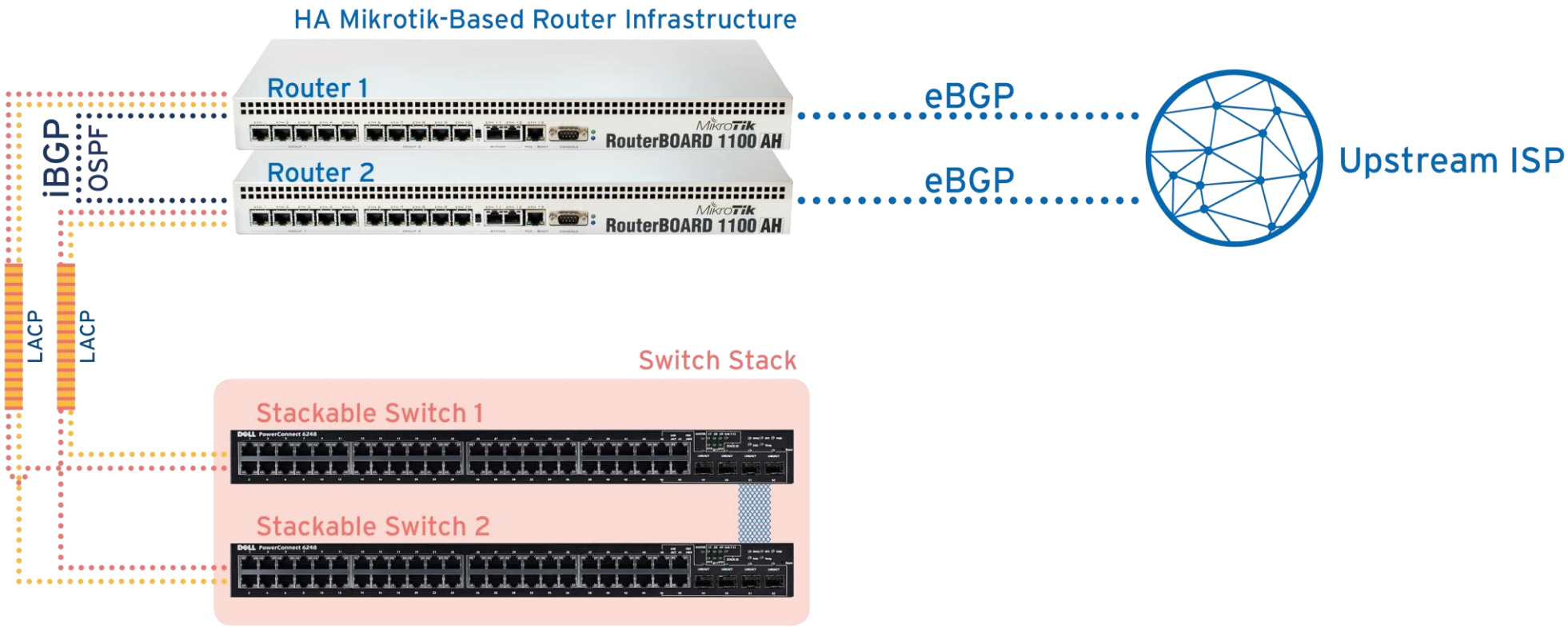

HA Mikrotik-Based Router Infrastructure
Router 1
Router 2
RouterBOARD 1100 AH
RouterBOARD 1100 AH
iBGP
OSPF
eBGP
eBGP
Upstream ISP

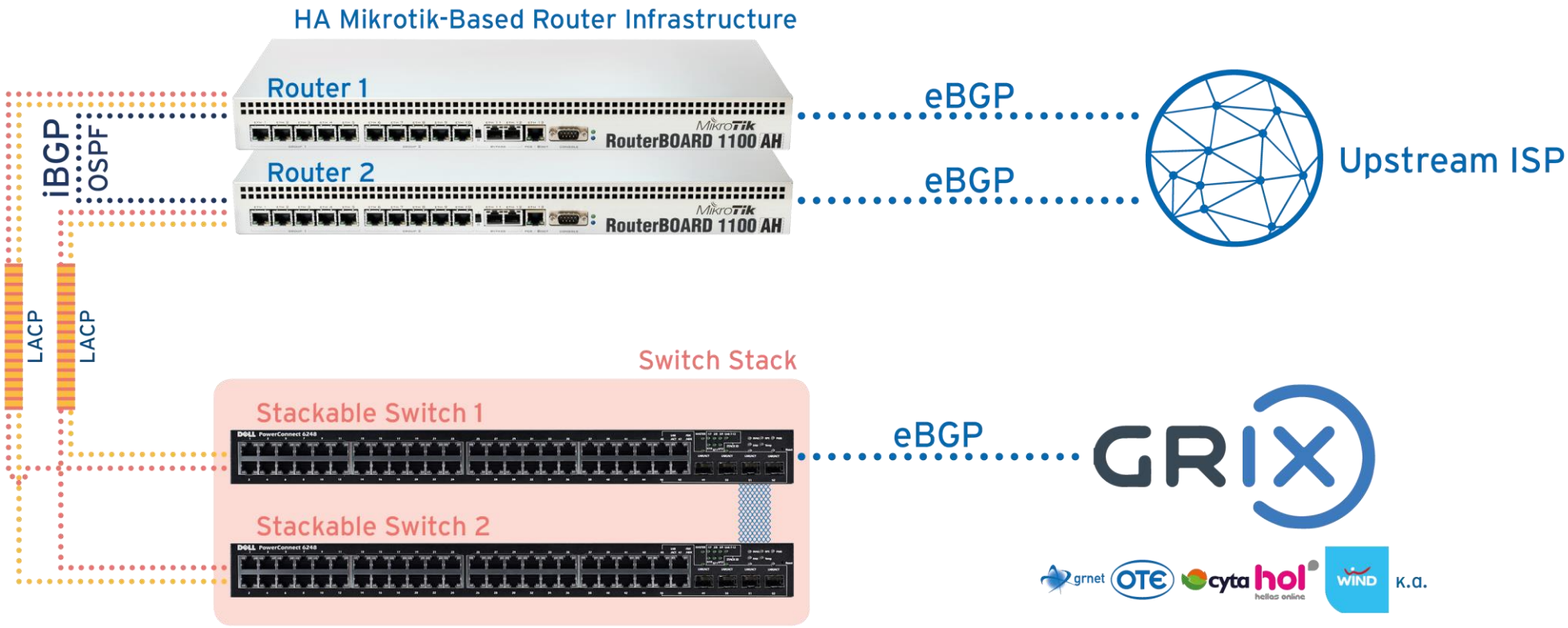

HA Mikrotik-Based Router Infrastructure
Router 1
Router 2
RouterBOARD 1100 AH
RouterBOARD 1100 AH
iBGP
OSPF
eBGP
eBGP
Upstream ISP


Switch Stack
Stackable Switch 1
Stackable Switch 2

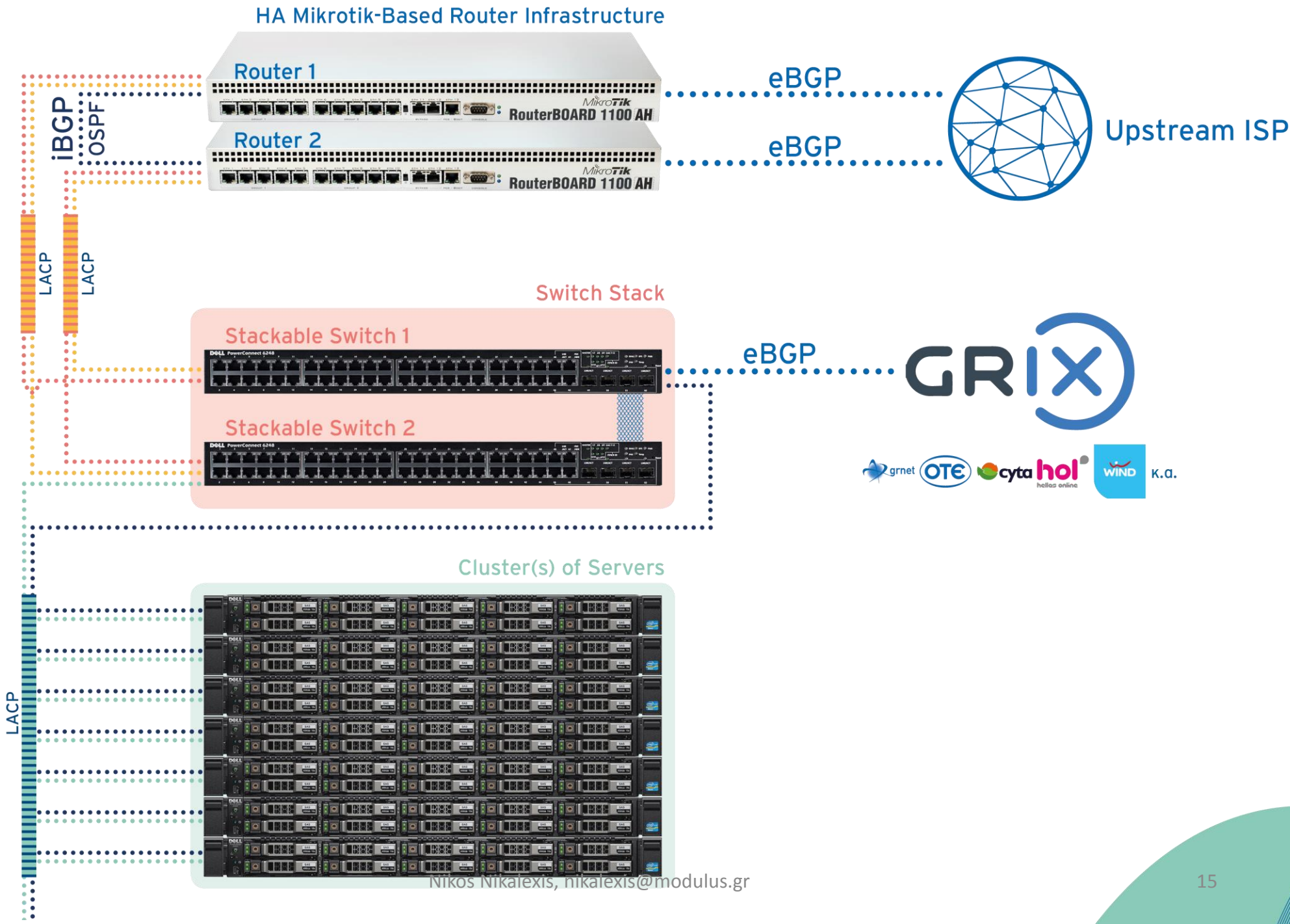

HA Mikrotik-Based Router Infrastructure
Router 1
RouterBOARD 1100 AH
Router 2
RouterBOARD 1100 AH
iBGP
OSPF
eBGP
eBGP
Upstream ISP
Switch Stack
Stackable Switch 1
Stackable Switch 2


HA Mikrotik-Based Router Infrastructure
Router 1
RouterBOARD 1100 AH
Router 2
RouterBOARD 1100 AH
iBGP
OSPF
eBGP
eBGP
Upstream ISP
Switch Stack
Stackable Switch 1
Stackable Switch 2


HA Mikrotik-Based Router Infrastructure
Router 1
RouterBOARD 1100 AH
Router 2
RouterBOARD 1100 AH
eBGP
eBGP
Upstream ISP
iBGP
OSPF
LACP
LACP
Switch Stack
Stackable Switch 1
Stackable Switch 2


HA Mikrotik-Based Router Infrastructure
Router 1
RouterBOARD 1100 AH
Router 2
RouterBOARD 1100 AH
iBGP
OSPF
LACP
LACP
eBGP
eBGP
Upstream ISP
Switch Stack
Stackable Switch 1
Stackable Switch 2
eBGP
GRIX
grnet
OTE
cyta
hol
hellas online
WIND
K.a.

modulus

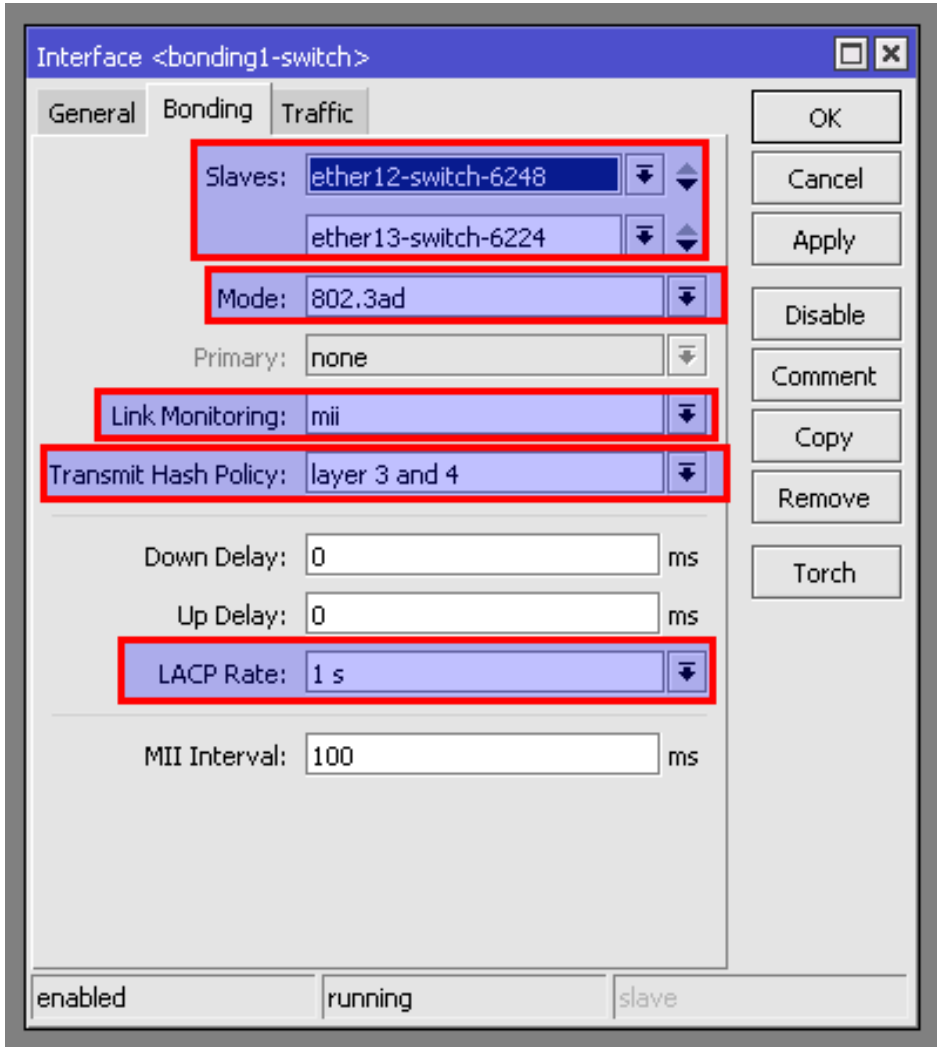

HA Mikrotik-Based Router Infrastructure
Router 1
RouterBOARD 1100 AH
Router 2
RouterBOARD 1100 AH
iBGP
OSPF
eBGP
eBGP
Upstream ISP
LACP
LACP
Switch Stack
Stackable Switch 1
Stackable Switch 2
eBGP
GRIX
grnet
OTE
cyta
hol
hellas online
WIND
K.a.
Cluster(s) of Servers
LACP

Mikrotik RouterOS setup

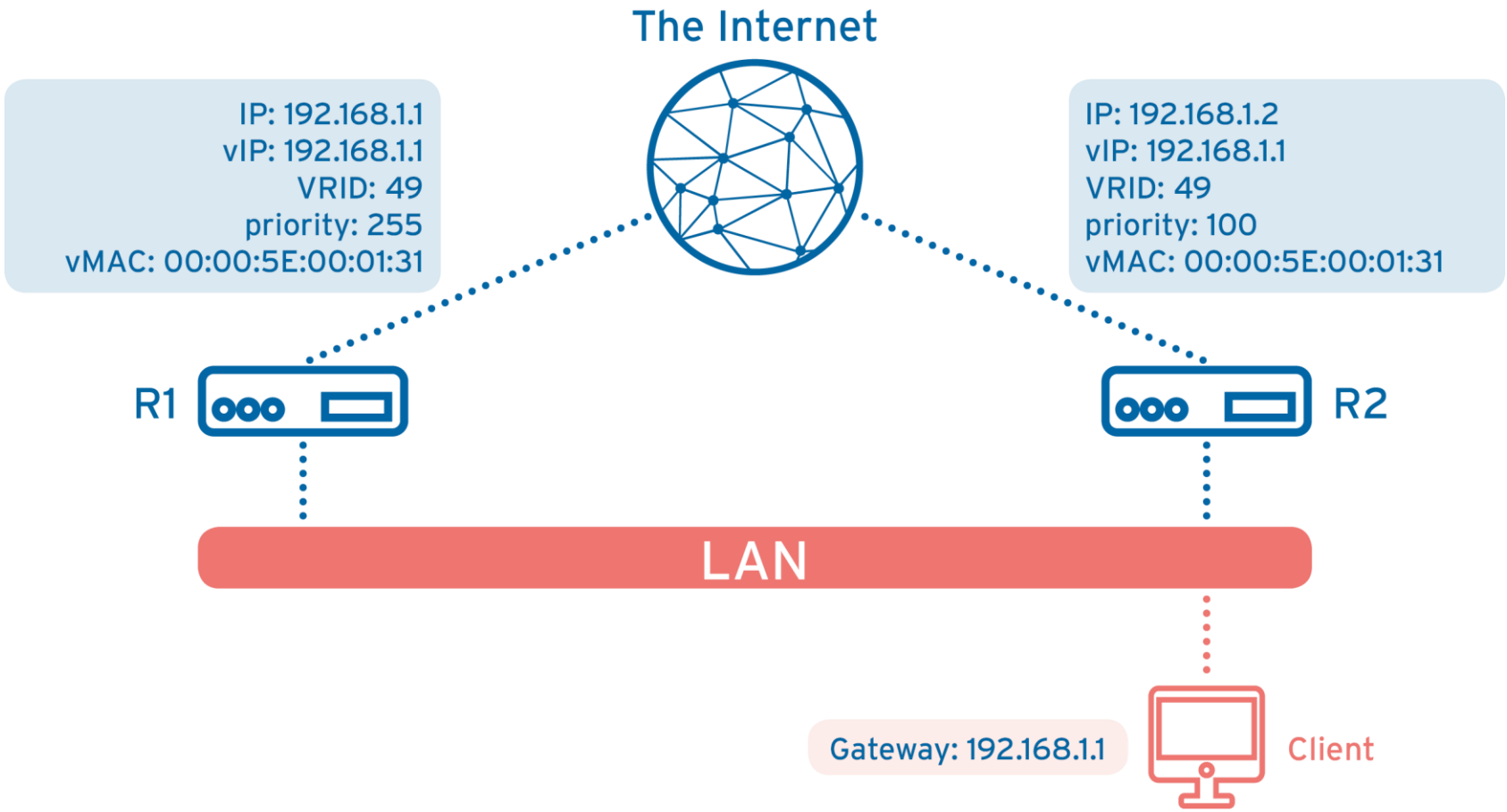
Interfaces, Bonding, VRRP, IP Addresses, Dynamic Routing, Traffic flow, Configuration Synchronization, Automatic Backup

modulus

Bonding


Interface <bonding1-switch>
General
Bonding
Traffic
OK
Cancel
Apply
Disable
Comment
Copy
Remove
Torch
Slaves:
ether12-switch-6248
ether13-switch-6224
Mode:
802.3ad
Primary:
none
Link Monitoring:
mii
Transmit Hash Policy:
layer 3 and 4
Down Delay:
0
ms
Up Delay:
0
ms
LACP Rate:
1 s
MII Interval:
100
ms
enabled
running
slave

VRRP (1/4)


The Internet
IP: 192.168.1.1
vIP: 192.168.1.1
VRID: 49
priority: 255
vMAC: 00:00:5E:00:01:31
IP: 192.168.1.2
vIP: 192.168.1.1
VRID: 49
priority: 100
vMAC: 00:00:5E:00:01:31
R1
R2
LAN
Gateway: 192.168.1.1
Client

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

modulus

VRRP (3/4)

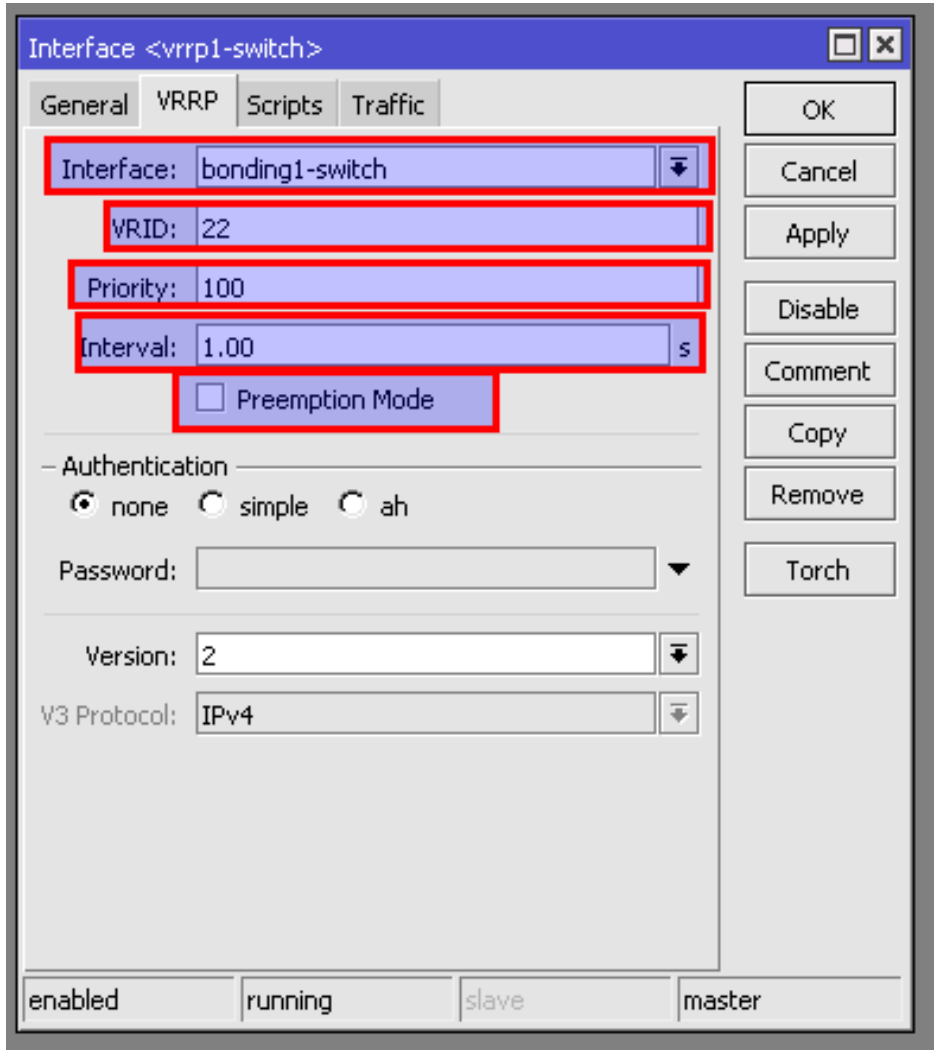
A solution hidden in RouterOS!!!

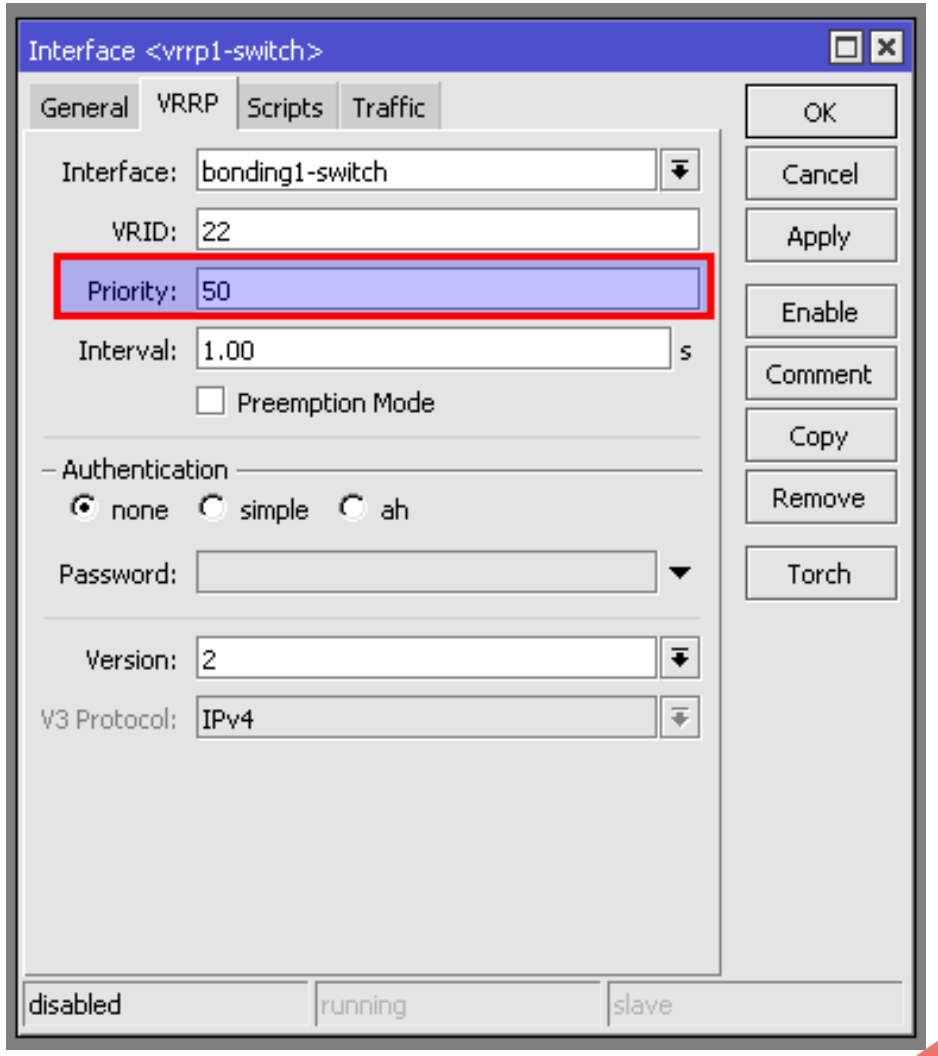
Undocumented but working

- 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

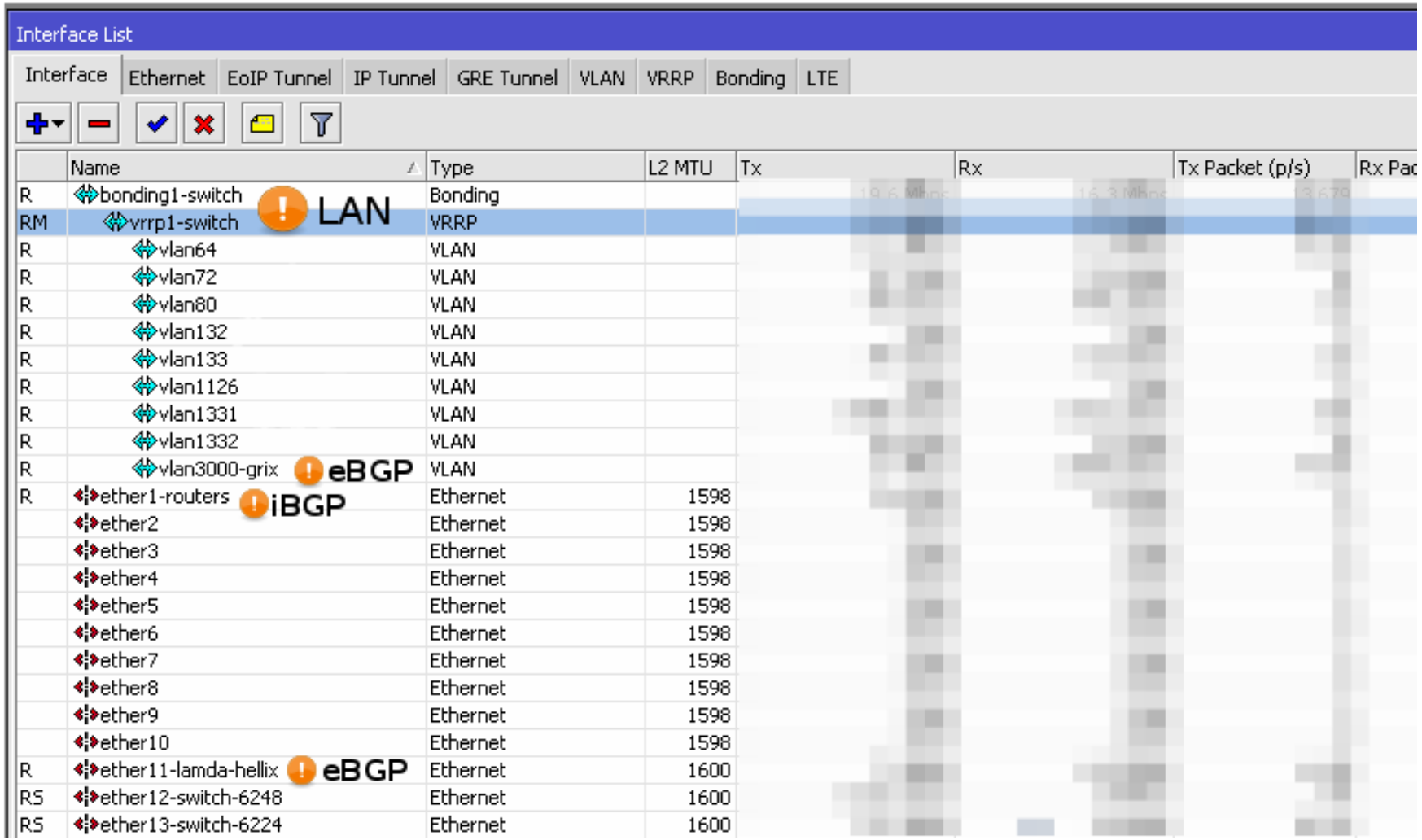
modulus

VRRP (4/4)


Interface <vrrp1-switch>
General
VRRP
Scripts
Traffic
OK
Cancel
Apply
Disable
Comment
Copy
Remove
Torch
Interface: bonding1-switch
VRID: 22
Priority: 100
Interval: 1.00 s
Preemption Mode
Authentication
none
simple
ah
Password:
Version: 2
V3 Protocol: IPv4
enabled
running
slave
master


Interface <vrrp1-switch>
General
VRRP
Scripts
Traffic
OK
Cancel
Apply
Enable
Comment
Copy
Remove
Torch
Interface: bonding1-switch
VRID: 22
Priority: 50
Interval: 1.00 s
Preemption Mode
Authentication
none
simple
ah
Password:
Version: 2
V3 Protocol: IPv4
disabled
running
slave

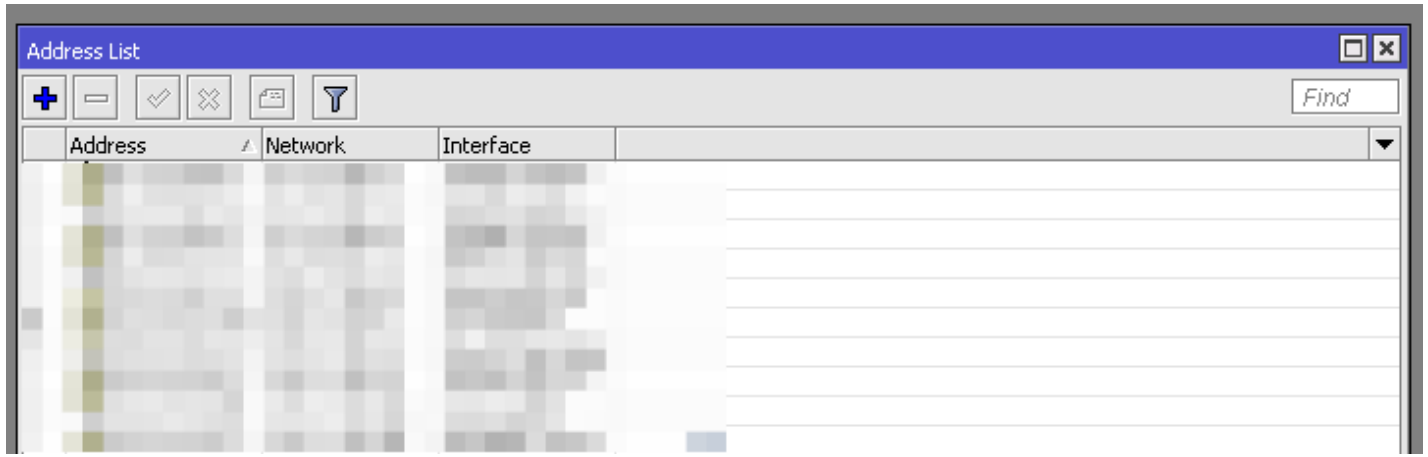
Interfaces overview



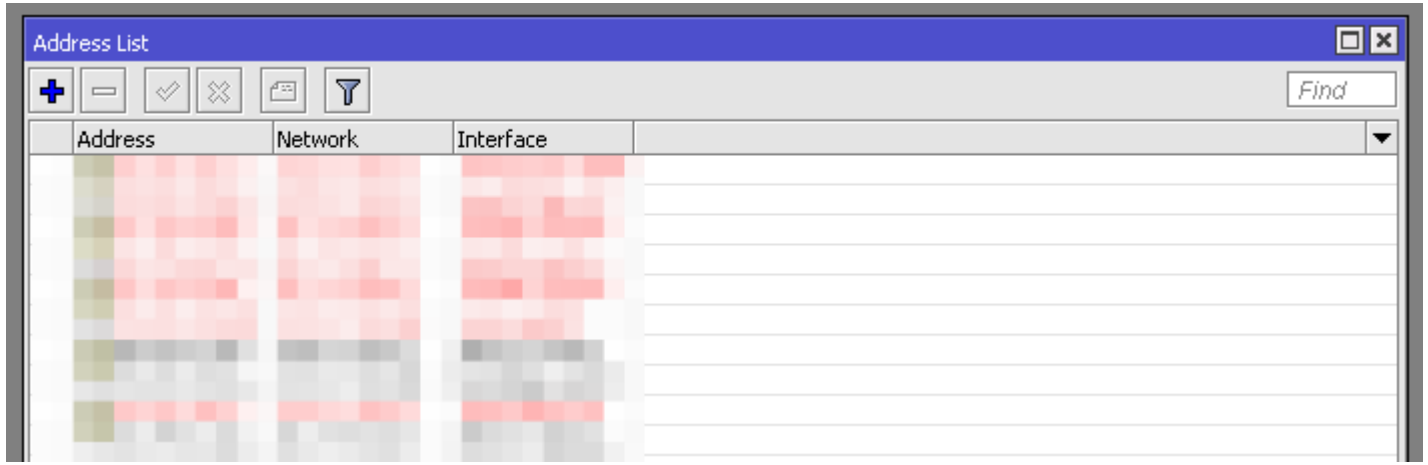
modulus

IP addresses overview

MASTER Router



BACKUP Router

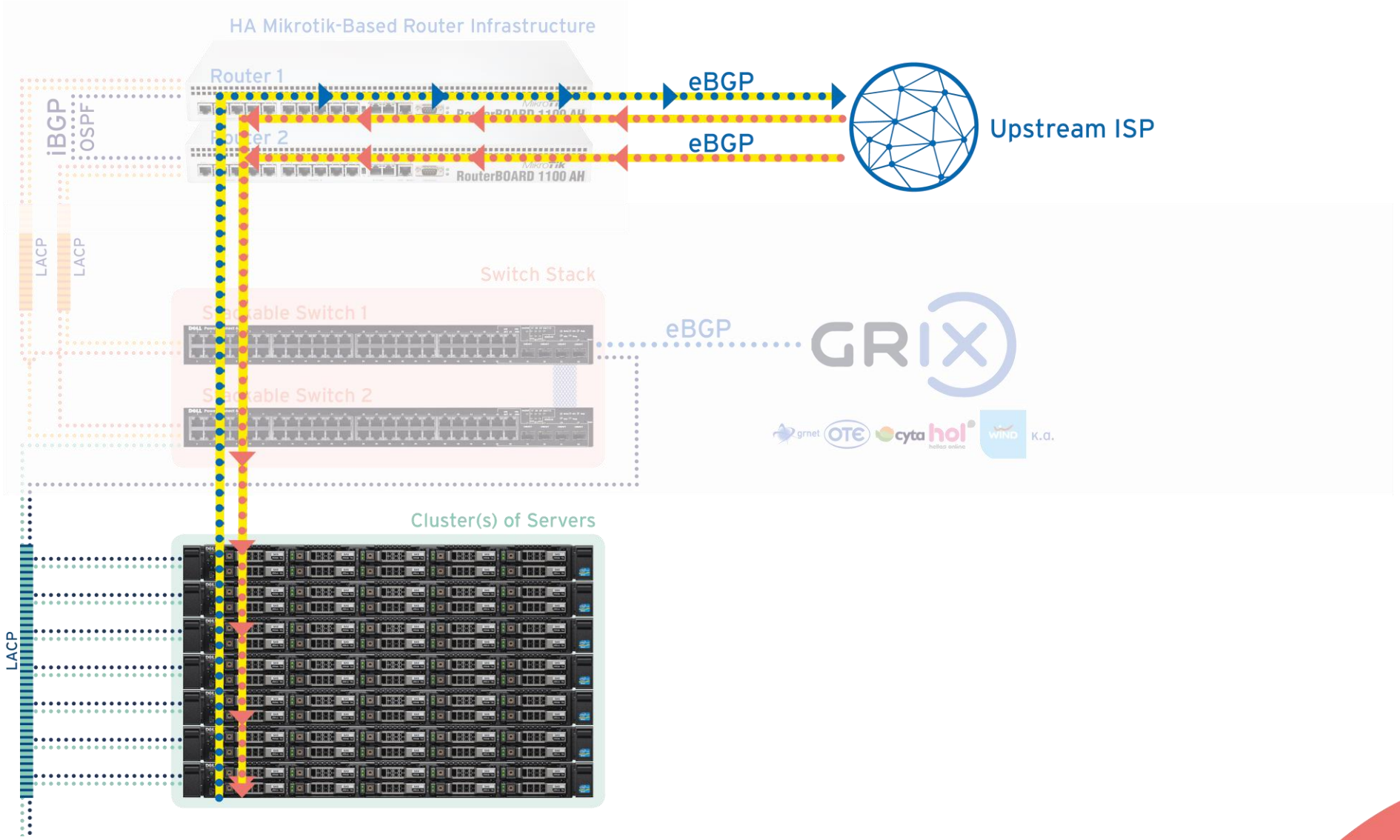


Dynamic Routing

- 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)

modulus

Traffic Flow



VRRP Scripting

- On MASTER:

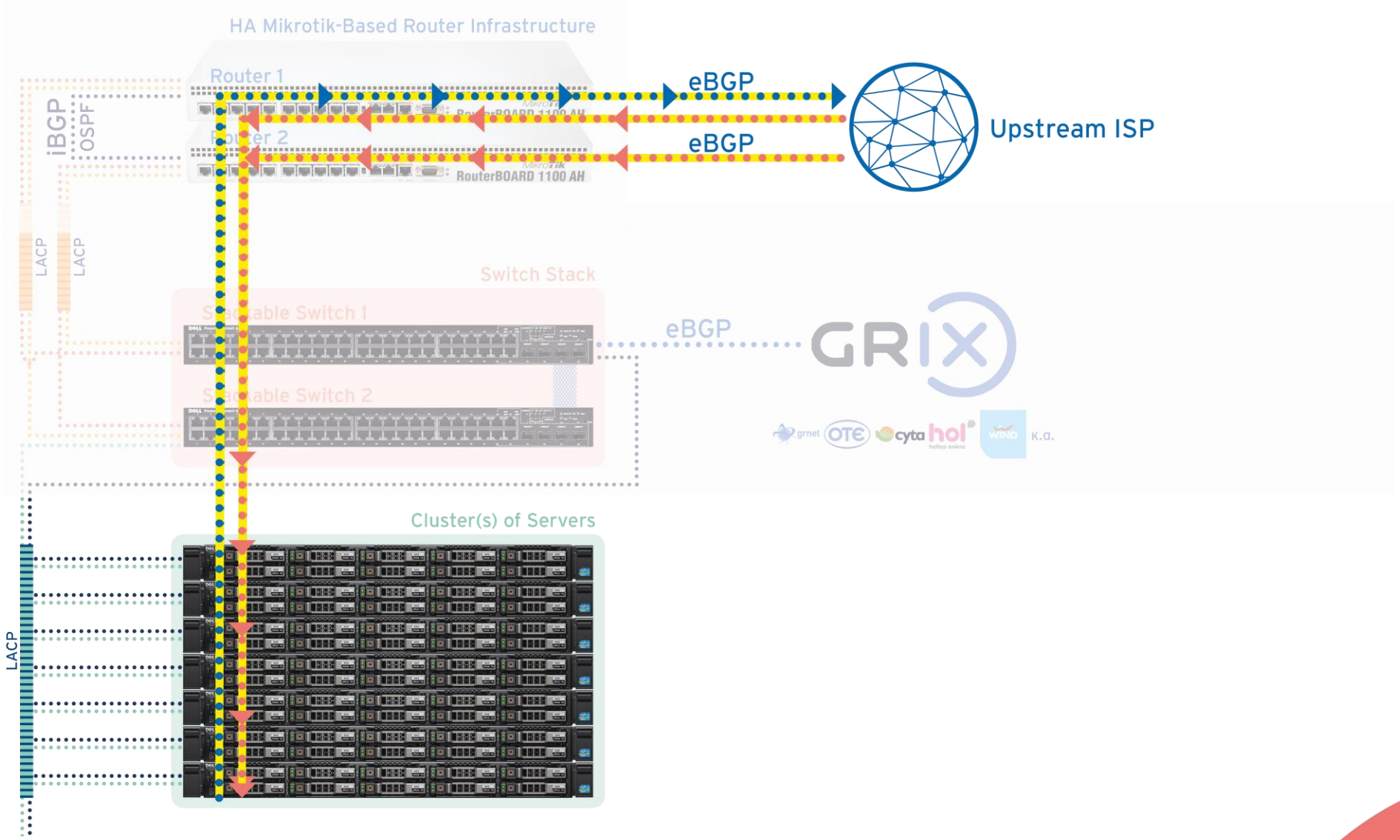
```
{
/routing filter set [find chain="providers-out" action="passthrough" set-bgp-med=200] set-bgp-med=100;
}
```

- On BACKUP:

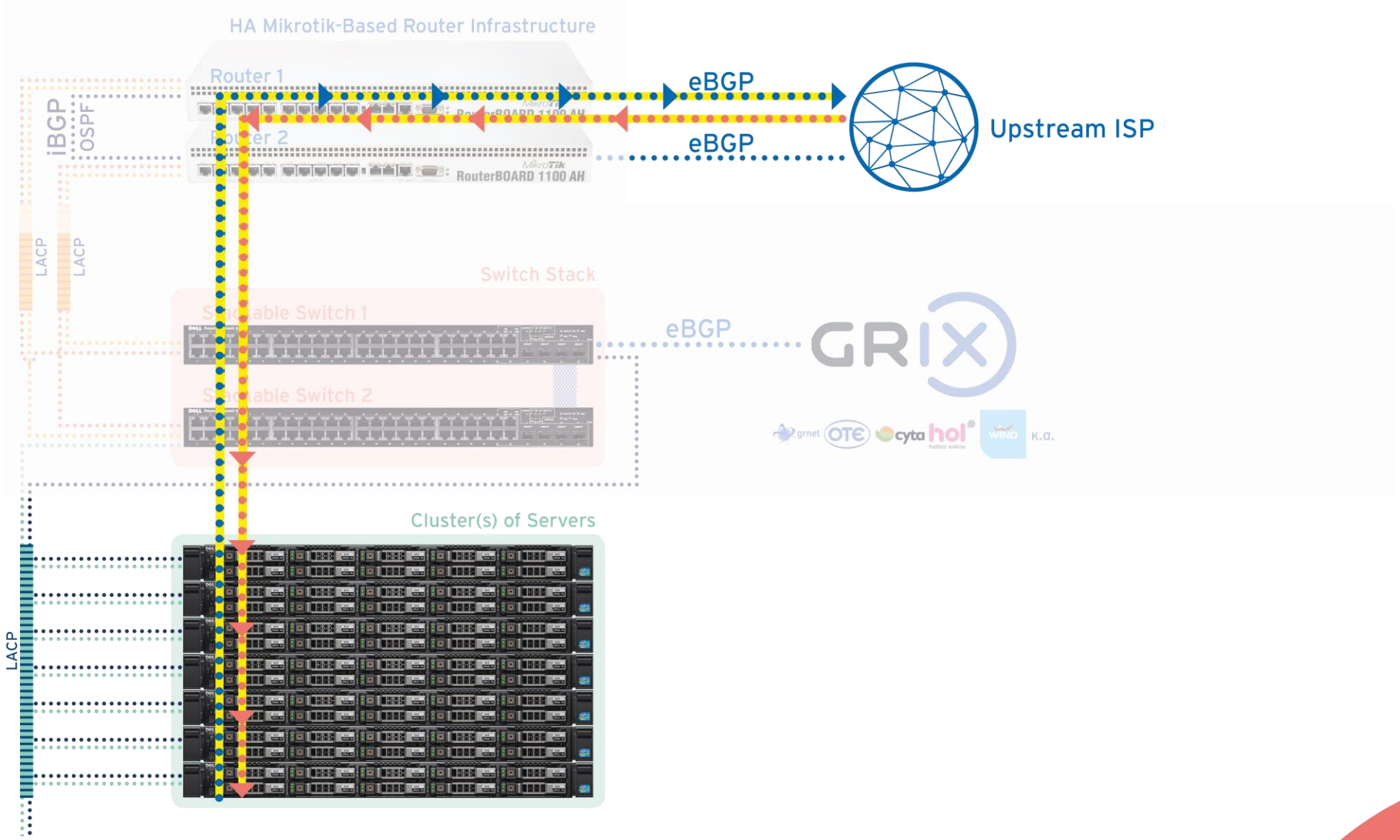
```
{
/routing filter set [find chain="providers-out" action="passthrough" set-bgp-med=100] set-bgp-med=200;
}
```

modulus

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

modulus

Configuration Syncronization (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

modulus

Configuration Synchronization (3/3)

```
/interface vrrp
add interface=bonding1-switch name=vrrp1-switch on-backup="{\r\
    \n/routing filter set [find chain=\"providers-out\" action=\"passthrough\"\
    \_set-bgp-med=100] set-bgp-med=200;\r\
    \n/interface disable vpn-modulus;\r\
    \n}\r\
    \n" on-master="{\r\
    \n/routing filter set [find chain=\"providers-out\" action=\"passthrough\"\
    \_set-bgp-med=200] set-bgp-med=100;\r\
    \n/interface enable vpn-modulus;\r\
    \n}\r\
    \n" preemption-mode=no version=2 vrid=22
[redacted]
add limit-at=10M/10M max-limit=10M/10M name=[redacted] \
    target=[redacted]/32
add limit-at=10M/10M max-limit=10M/10M name=[redacted] target=\
    [redacted]/32
```

```
/interface vrrp
add disabled=yes interface=bonding1-switch name=vrrp1-switch on-backup="{\r\
    \n/routing filter set [find chain=\"providers-out\" action=\"passthrough\"\
    \_set-bgp-med=100] set-bgp-med=200;\r\
    \n/interface disable vpn-modulus;\r\
    \n}\r\
    \n" on-master="{\r\
    \n/routing filter set [find chain=\"providers-out\" action=\"passthrough\"\
    \_set-bgp-med=200] set-bgp-med=100;\r\
    \n/interface enable vpn-modulus;\r\
    \n}\r\
    \n" preemption-mode=no priority=50 version=2 vrid=22
[redacted]
```

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

modulus

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

modulus

Goals achieved

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

modulus

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

modulus

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

modulus

Thank you!

Any Questions?