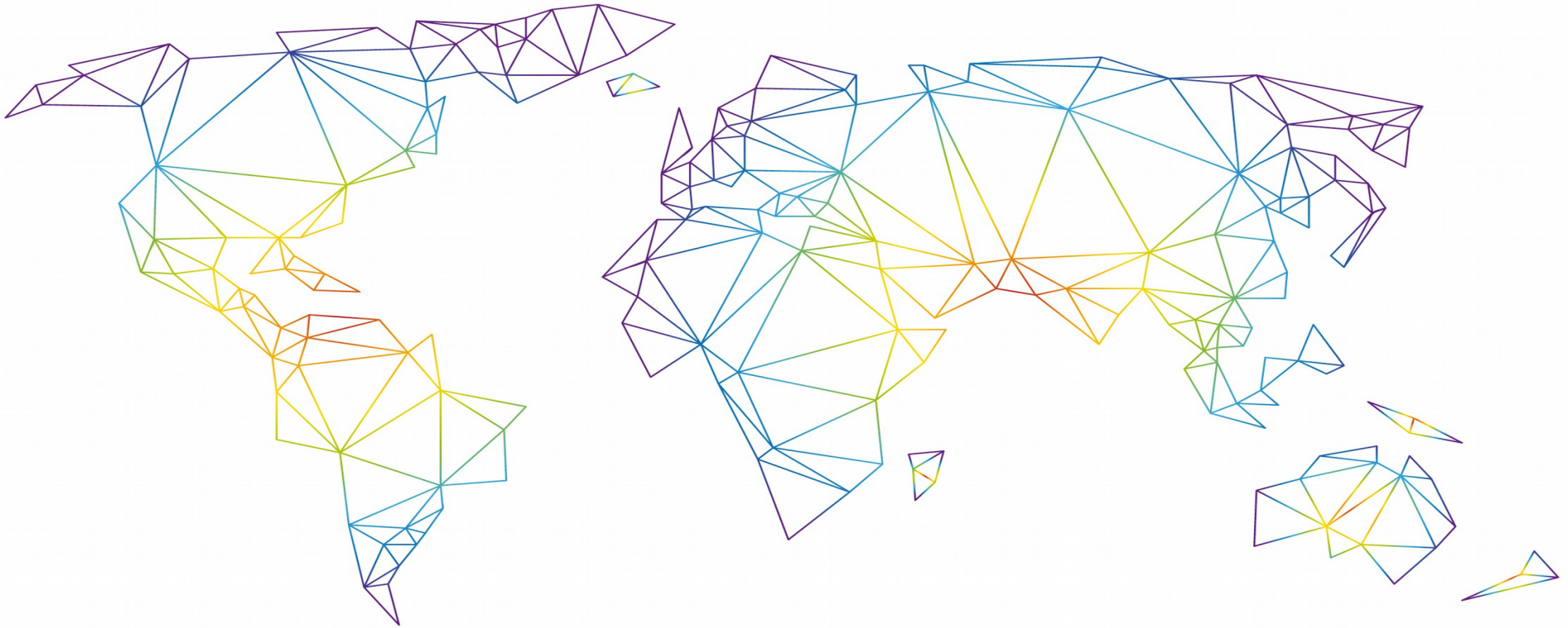
100% Network Uptime with RouterOS Border Routers



Presented by Mihai Săftoiu

21 OCTOBER 2014
MUM in Bucharest, Romania

About me

+10 years of experience as a sysadmin & network admin

Mikrotik Certified Consultant

MTCNA, MTCRE, MTCWE

CEO @ TIER Data Center

- E-mail: mihai.saftoiu@tier.ro
- Phone: +4 0723-197-754
- http://www.nixservers.ro

About TIER @ ENERGOTECH Group

Why this presentation?

Uptime SLAs are a network engineer's worst nightmare.

My simple guide:

1. Plan coherently.
2. Execute precisely as planned.
3. Monitor and improve.

If your phone is not ringing on the job you're doing it right.

Demistifying uptime

Contrary to popular belief uptime is not something a hosting provider posts on their website.

It's a mathematical probability for things to go RIGHT out of all possible scenarios.

„Uptime is a measure of the time a machine, typically a computer, has been working and available. Uptime is the opposite of downtime.

It is often used as a measure of computer operating system reliability or stability, in that this time represents the time a computer can be left unattended without crashing, or needing to be rebooted for administrative or maintenance purposes."

- Source Wikipedia

Demistifying uptime

Is there anyone who determines uptime for various scenarios?

Yes! The Uptime Institute - http://uptimeinstitute.com

There are 4 possible mathematical values:

99,671% / 99,741% / 99,982% / 99,995%

Source: http://www.accu-tech.com/Portals/54495/docs/102264ae.pdf

Myths and misconceptions about availablity:

http://uptimeinstitute.com/professional-services/tier-myths-and-misconceptions

So, what is a border router?

Border routers from different Autonomos Systems connect together to make up the Internet.

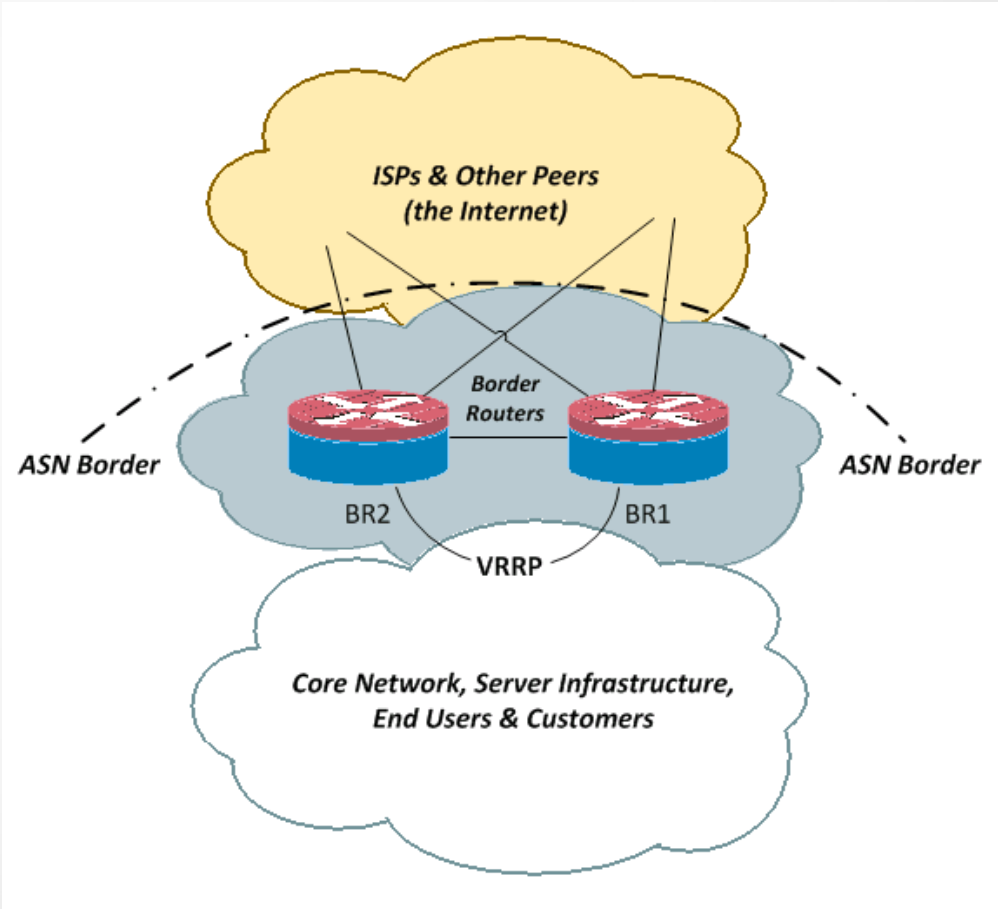
The routers that connect to a different AS are called border routers.

They usually run BGP to communicate with their peers.

Peace to the OSPF terminology addict!

Is this enough for 100% uptime?

Actually, YES, but not mathematically.



Why use RouterOS for Border Routers

RouterOS provides ALL required protocols and functions to run an enterprise or ISP network.

RouterOS is cost-effective

RouterOS is easy to deploy

RouterOS is easy to use (Winbox is amazing!)

RouterOS is easy to monitor

RouterOS is easy to backup

RouterOS is easy to recover

RouterOS can run a redundant network

The main factors that influence uptime

- Sometimes the distributor will move to sell us what he thinks is best
- Poor planning, network design flaws and unadequate equipment sizing
  - System memory congestion
  - Bandwidth congestion
  - CPU & IRQ congestion
- Work in progress for the latest RouterOS major version
- Loss of input power and other electrical failures
- RouterOS upgrades & firmware upgrades
- Our ISP's network issues, poorly configured rerouting
- Denial Of Service attacks, poor exterior and interior security
- Lack of insight into the network, proactive monitoring is important
- Human error & misconfigurations

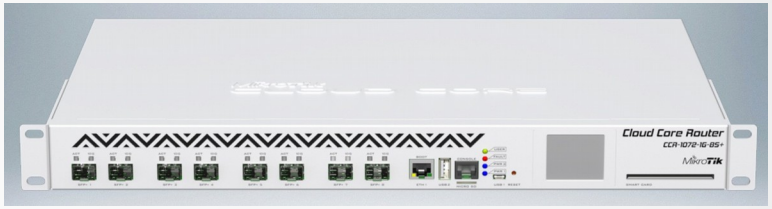
Choosing the right arhitecture

For high bandwidth requirements or routing at „wirespeed” use CCR series.

ENTERPRISE

CCR-1036-12G-4S-EM

ISP or high bandwidth SANs



CCR-1072-1G-8S+

Low to medium bandwidth requirements, lots of firewall rules, routers with mix of different types of interfaces or for very complex router configurations use x86 for better control over high frequency CPU cores.

For enterprise low to medium bandwidth requirements you can also use PPC.

Planning – Resources

RAM

It's an important provisioning factor when you are going to have a large amount of routes on your border routers.

For example:

- when you are provisioning hardware for a route reflector
- when you are accepting multiple BGP full table exports from multiple eBGP peers
- when you have a multiple BGP routing systems that redistribute routes by iBGP to your main border routers

Simple rule:

Take into account 768 MB RAM per full table (for aproximately +500.000 prefixes), when provisioning add an extra 512 MB for other purposes.

Planning – Resources

Bandwidth provisioning

There is no other way around it: **Bandwidth = $**

- Always use multiple providers.
- Avoid providers that are known to oversell their bandwidth where possible.
- Work with high-quality bandwidth providers.
- Try and set up redundant capacity with each provider.
- Peer with everyone you can. InterLAN & RONIX are good local Internet Exchange Points.
- Get 95th percentile billing where you can.
- Make sure your providers do not share their network infrastructure for transporting data.
- Make sure that before signing a contract your provider understands your needs and that he has the technical means to provide you with your requirements.
- Make sure that your provider has plenty of free capacity and good peering arrangements in case you have a bandwidth attack.

Planning – Resources

CPU

Avoid at all costs single core CPUs on border routers.

For enterprise routing use:

no_active_interfaces (physical, bridge or bond) / 2 = no_cores

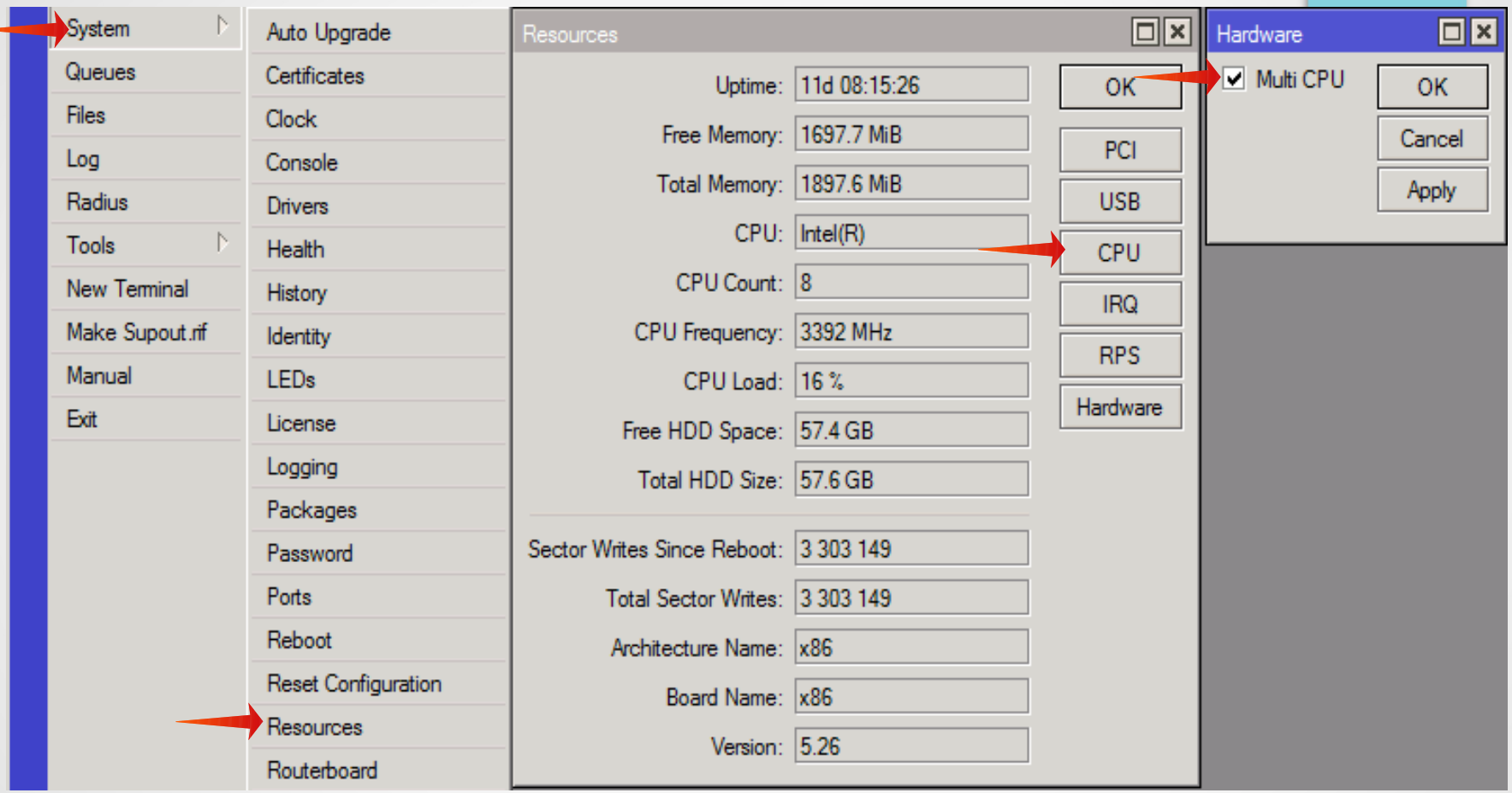
4 active routing interfaces → 2 CPU cores

For ISP routing use:

no_active_interfaces (physical, bridge or bond) + 20-25% = no_cores

4 active routing interfaces → 6 CPU cores

Planning – Resources



Enabling multi-cpu on x86 systems:

```
[admin@br1] > /system hardware set multi-cpu=yes
[admin@br1] > /system reboot
```

Planning – Resources

Why do we need multi-cpu?

Multi-cpu is not only good for distributing loads between different router processes and avoiding cpu time congestion but it is also even more important for distributing interrupt requests between cpu cores.

What is an Interrupt ReQuest (IRQ)?

„In a computer, an interrupt request (or IRQ) is a hardware signal sent to the processor that temporarily stops a running program and allows a special program, an interrupt handler, to run instead. Interrupts are used to handle such events as data receipt from a modem or network, or a key press or mouse movement."

- Source Wikipedia

Planning – Resources

What does this mean for border routers?

It means that at times we might be getting a lot of traffic in small packets which will lead to IRQ congestion.

This is very important because we will not be getting the throughput we expect on that link due to the fact that the bus is congested by interrupt requests. The whole system will suffer.

How is this possible?

Quite simple, for example there is no such thing as 4,58 Mbps on a 100Mbps link!!! It's just an average of time used from total time.

Planning – Resources

IRQ Considerations

1. x86

The linux IRQ driver for x86 works very well. Offloading and correctly distributing IRQ to multiple CPUs is the correct answer.

How to do this?

A. Motherboard selection (LGA-2011):

- X79 chipset: 40 PCIe lanes
- X99 chipset: 40 PCIe lanes (better IRQ distribution)
- C602J, C602, C604, C606, C608 chipsets: 40 PCIe lanes per each CPU

B. PHY selection:

- Use certified Mikrotik equipment: RB44Ge - http://routerboard.com/RB44Ge
- Use more expensive PHYs based at least on Intel® 82576 Gigabit Ethernet Controller, Intel® 82572EI Gigabit Ethernet Controller or superior chipsets.
  http://www.intel.com/content/www/us/en/network-adapters/gigabit-network-adapters/ethernet-et2-multi-port.html

Planning – Resources

IRQ Considerations

2. PowerPC (RouterBoard PPC series)

PowerPC IRQ handling is done almost the same way as x86. The PPC IRQ handling is a port of linux generic hardirq handling.

3. Tile (RouterBoard CCR series)

Excerpt from Linux/arch/tile/include/asm/irq.h

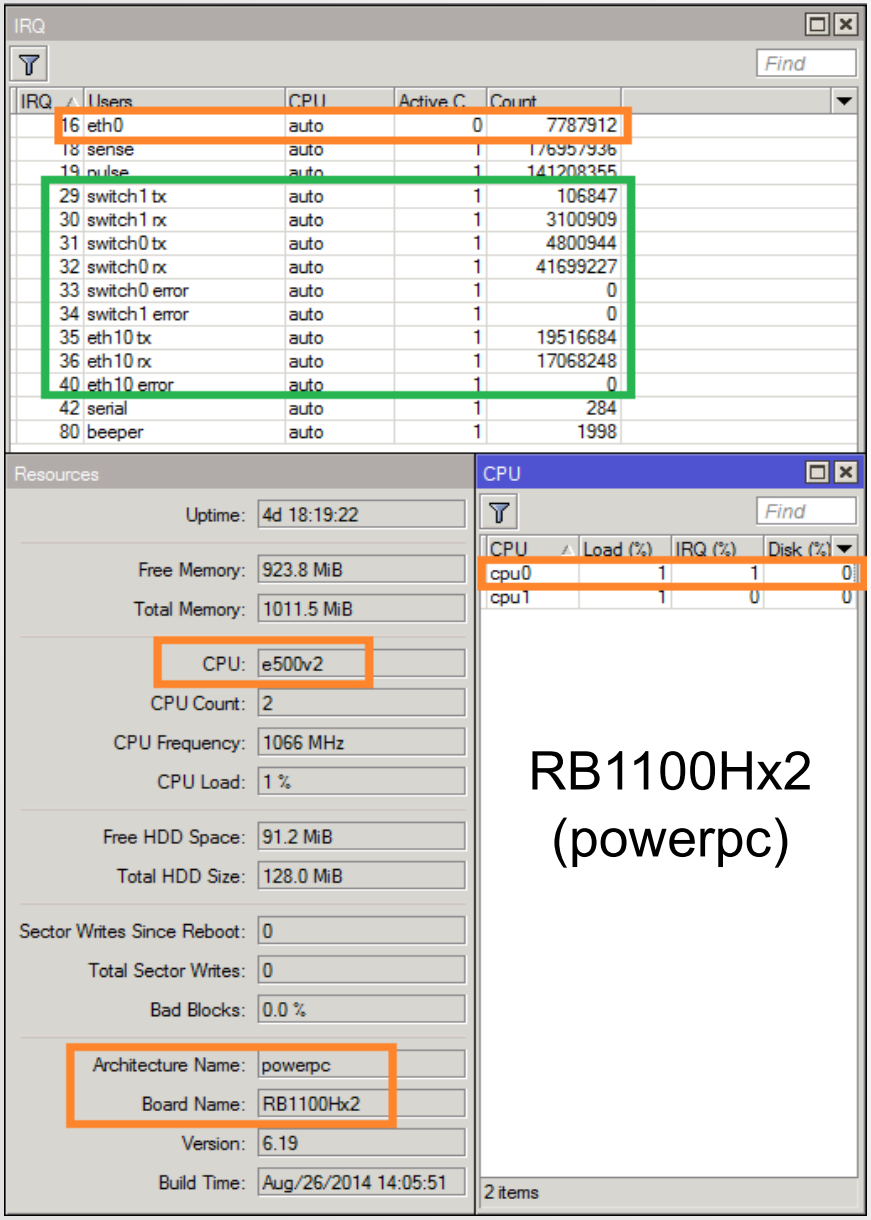
„/*

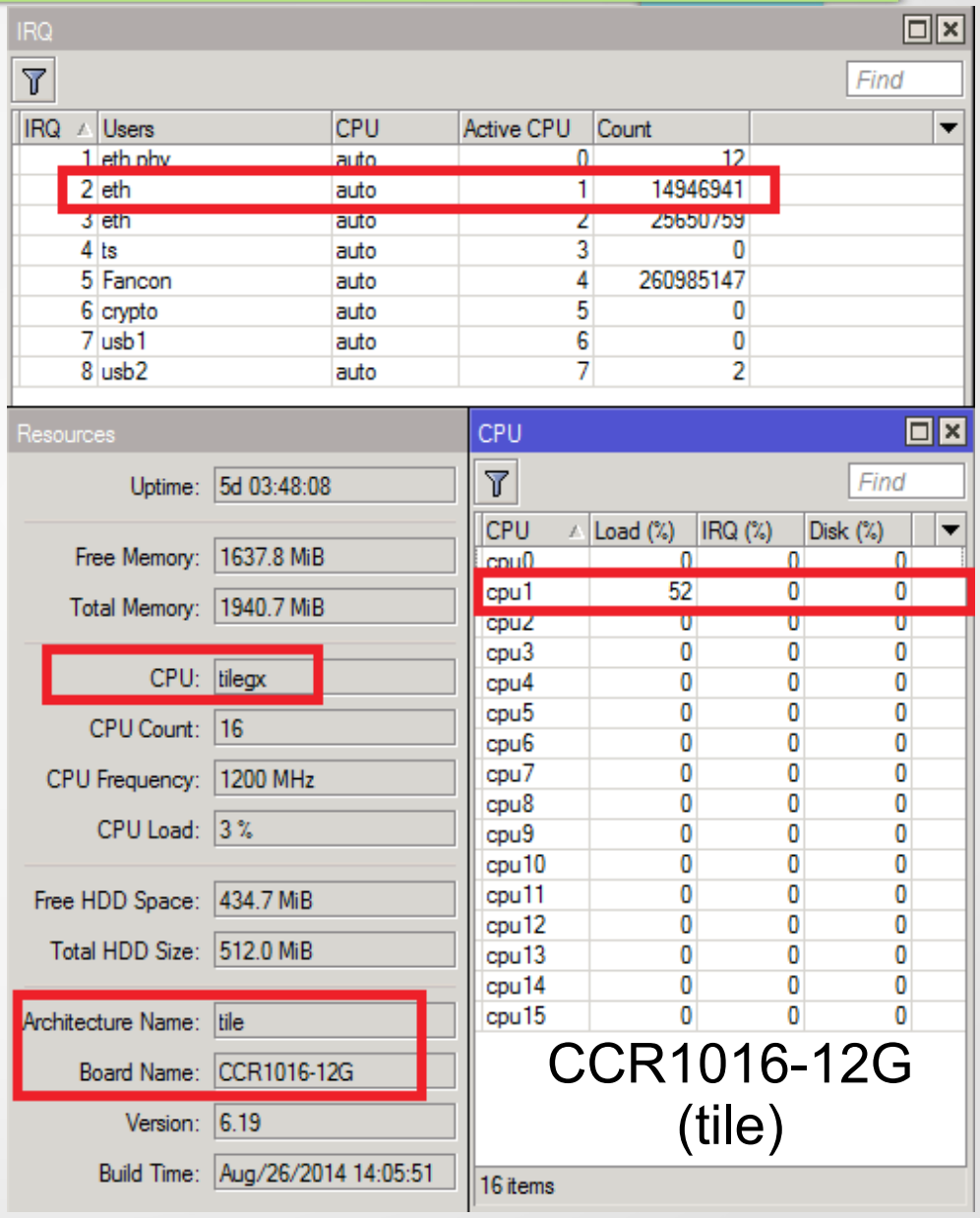
* Copyright 2010 Tilera Corporation. All Rights Reserved.

...

* Different ways of handling interrupts. Tile interrupts are always per-cpu; there is no global interrupt controller to implement enable/disable. Most onboard devices can send their interrupts to many tiles at the same time, and Tile-specific drivers know how to deal with this."

Planning – Resources





Planning – Resources

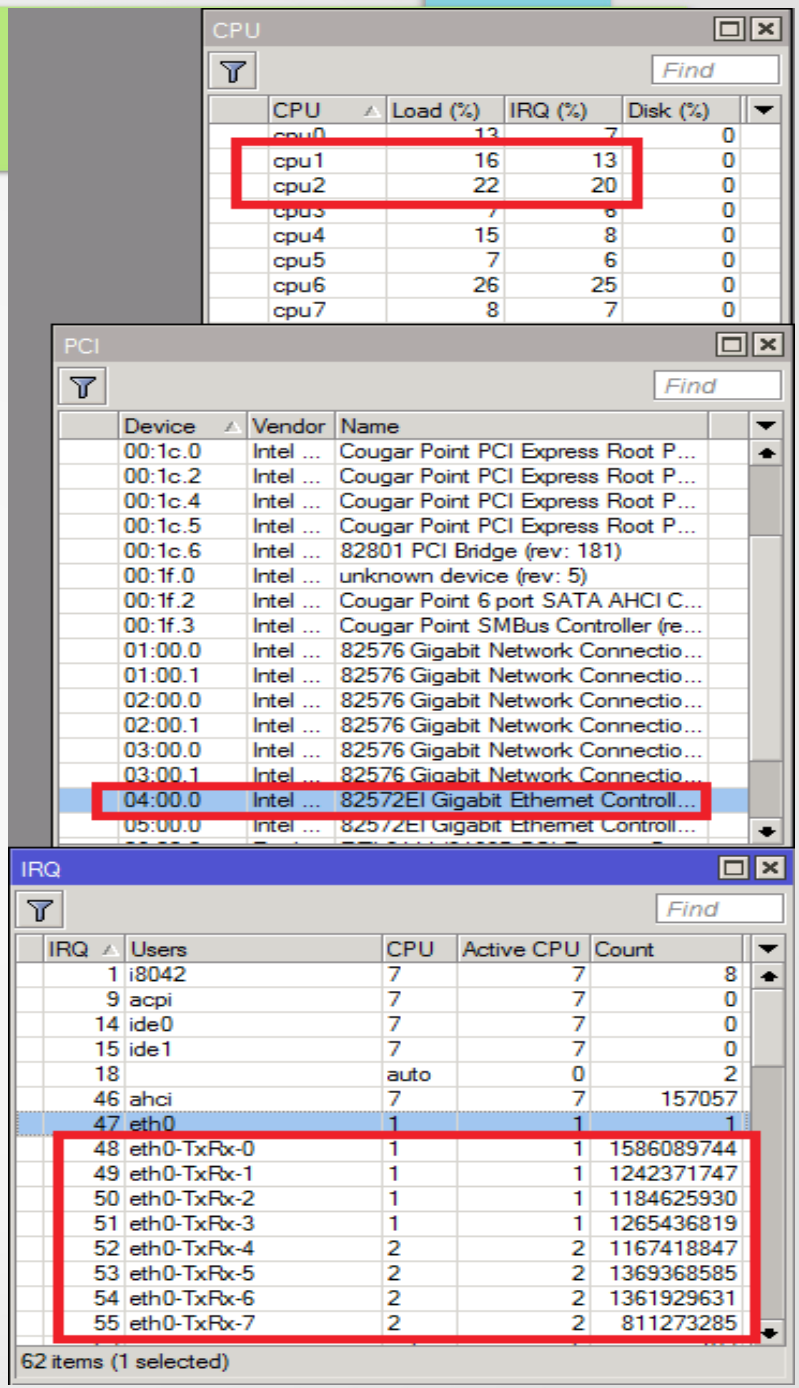
Advanced IRQ Considerations

Interrupt affinity with multi-cpu is not necessarily a good thing on ROS

IRQ distribution is controlled by the IO-APIC chip. It works in physical and logical mode.

IO-APIC logical mode (round-robin multi-cpu – IRQ auto) degrades performance.

CPU affinity will increase performance & reliability = IO-APIC in physical mode



Choosing the right ROS version (-1)

Changelog excerpt for version 6.19

„What's new in 6.19 (2014-Aug-26 14:05):

...

*) sstp - make sstp work on i386 as well;

*) ipsec - ... kill only relevant SAs;

*) vpls - do not abort BGP connection when...

*) dns-update - fix zone update;

*) sstp - make it work for x86 systems

*) ipv6 - Gre6 can now correctly fragment..."

Well then... What's new in 5.26:

*) ssh - fixed denial of service;

EOL

Sometimes it's better to downgrade.

Power redundancy

There are 2 important things to know:

1. If you think electrical failures will occur, they WILL.

2. If you think electrical failures will not occur, they WILL.

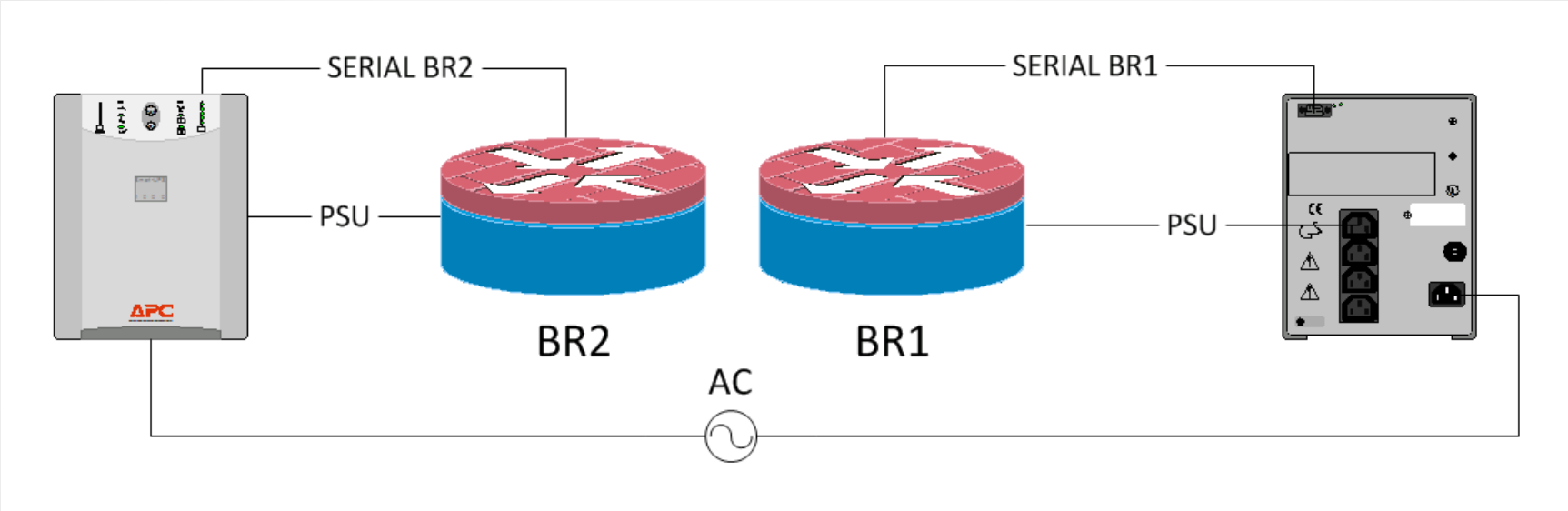
How do we protect our border routers?

1. Use quality UPS systems, power stabilizers, a backup generator.

2. If there is no generator available take advantage of the serial or USB port on your routers and connect to APC UPS compatible products (BackUPS Pro / SmartUPS) to extend the duration of your backup time.

Anticipating electrical failures

Simple power backup



For both routers set up the serial communication with their corresponding UPS. Go into hybernation at 10% UPS capacity by default:

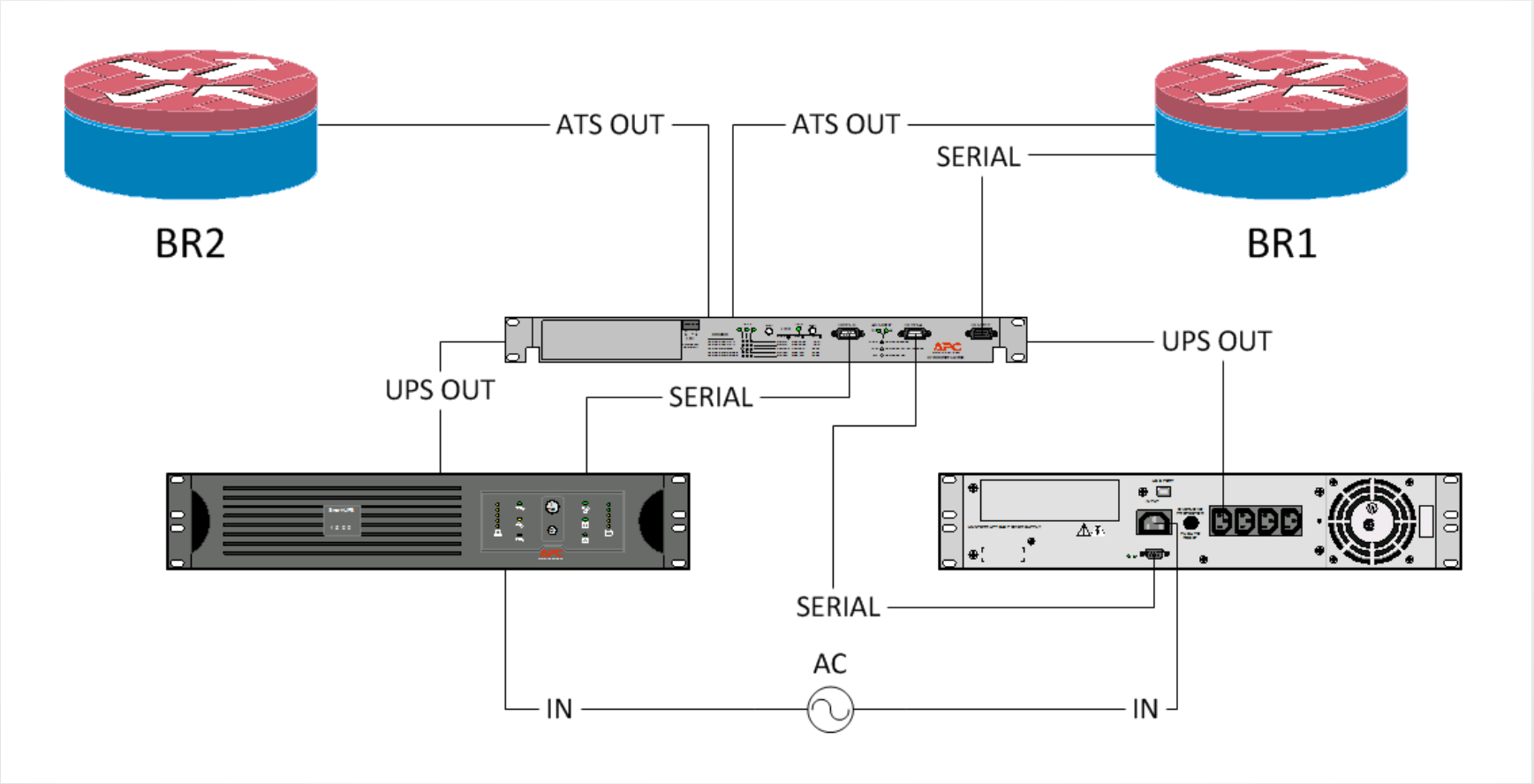
```
[admin@br1] > system ups add port=serial1 disabled=no
[admin@br1] > system ups set 0 min-runtime=0 offline-time=0
```

If you want to automatically calibrate the runtime you can do that by waiting for the UPS to fill up to 100% and running calibration:

```
[admin@br1] > system ups rtc 0
```

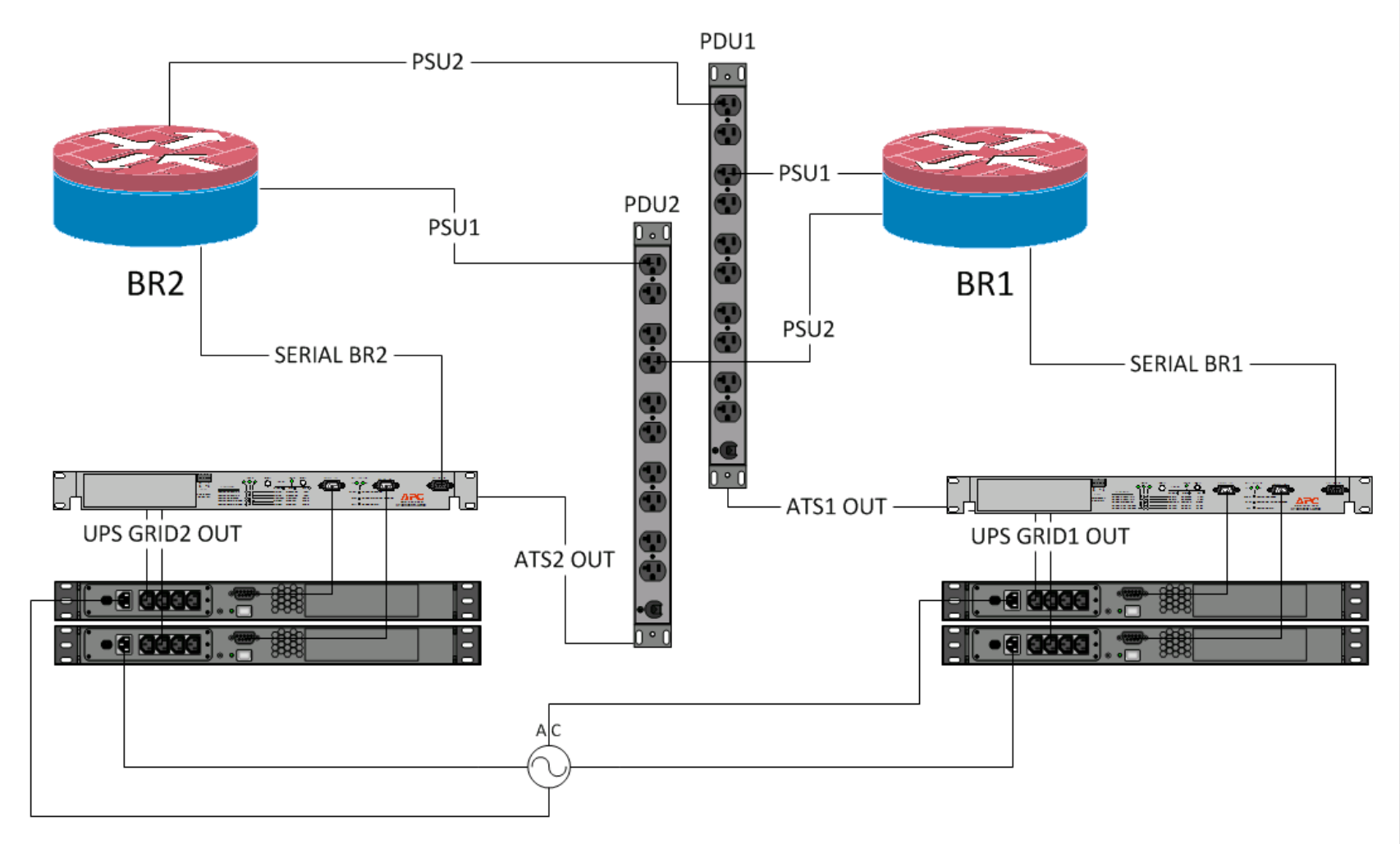
Anticipating electrical failures

Better power backup – Using an ATS & redundant UPS



Anticipating electrical failures

Even better power backup – Redundant ATS & UPS



Upgrading unnoticed

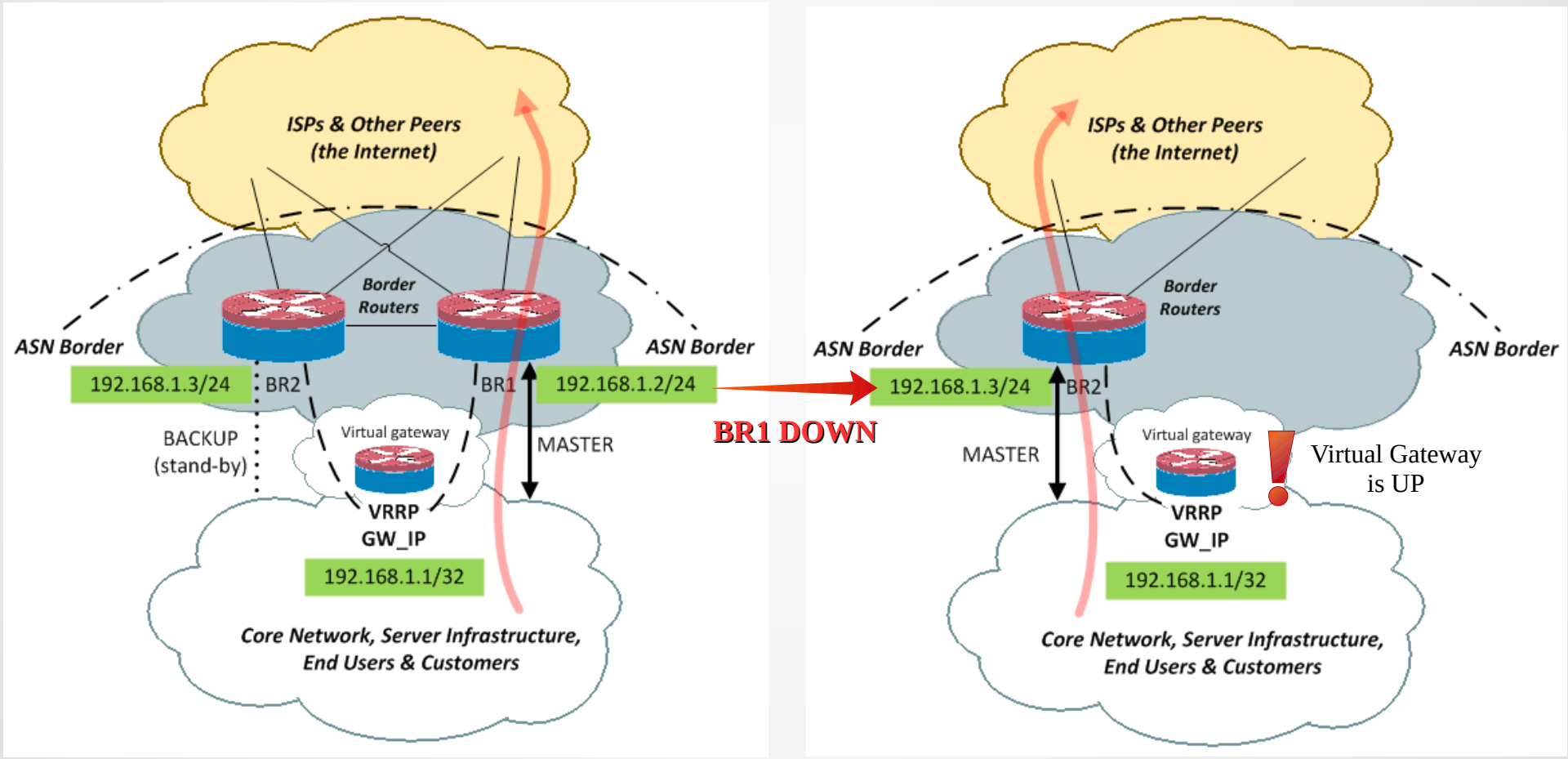
VRRP allows us to present a single gateway IP withing the corresponding broadcast domain for that gateway (or it can be used as next-hop in more advanced configurations).

It can be used at the edge of the network, at the core of the network or even to make customer access routers redundant.

VRRP is a THE MOST powerful tool for the network admin looking to provide its network with superior uptime.

Upgrading unnoticed

Simple and effective topology to achieve unnoticed reboots



Configuration examples can be found on
http://wiki.mikrotik.com/wiki/VRRP-examples

Action – instant response

Ok, I've read the VRRP wiki examples. Configuring VRRP is quite simple. Or is it?

The way to do it with efficiency is **PHY -> Bond -> Bridge -> VRRP**

This will increase resource consumption but will give you what you need to aggregate bandwidth in a simple way, send it multiple paths hassle-free and run it redundantly through time.

We are looking for best uptime possible, remember? Simple management is good management.

To initially set up this mode you will first need to reserve 2 physical interfaces on each router (towards your AS side):

eth0 and eth1.

Action – instant response

BR1 - Setting up VRRP to be transparent to physical changes in 6 easy steps

```
Step 1. [admin@br1] > interface bonding add name=bond-lan1 slaves=eth0,eth1 mode=balance-xor
link-monitoring=mii-type2 transmit-hash-policy=layer-3-and-4 down-delay=0.01 up-delay=0.01
lacp-rate=1sec mii-interval=0.1 mtu=1522 disabled=no
```

```
Step 2. [admin@br1] > interface bridge add name=bridge-lan1 protocol-mode=none admin-
mac="AD:MI:N0:MA:C0:01" mtu=1522 disabled=no
```

```
Step 3. [admin@br1] > interface bridge port add bridge=bridge-lan1 interface=bond-lan1
disabled=no
```

```
Step 4. [admin@br1] > ip address add interface=bridge-lan1 address=192.168.1.2/24
network=192.168.1.0  disabled=no
```

```
Step 5. [admin@br1] > interface vrrp add name=vrrp-lan1 mtu=1504 interface=bridge-lan1 vrid=1
priority=10 interval=2 preemption-mode=yes authentication=simple password=mypass version=2
disabled=no
```

```
Step 6. [admin@br1] > ip address add interface=vrrp-lan1 address=192.168.1.1/32
network=192.168.1.1 disabled=no
```

Action – instant response

BR2 - Setting up VRRP to be transparent to physical changes in 6 easy steps

```
Step 1. [admin@br2] > interface bonding add name=bond-lan1 slaves=eth0,eth1 mode=balance-xor link-
monitoring=mii-type2 transmit-hash-policy=layer-3-and-4 down-delay=0.01 up-delay=0.01 lacp-rate=1sec
mii-interval=0.1 mtu=1522 disabled=no
```

```
Step 2. [admin@br2] > interface bridge add name=bridge-lan1 protocol-mode=none admin-
mac="AD:MI:N0:MA:C0:02" mtu=1522 disabled=no
```

```
Step 3. [admin@br2] > interface bridge port add bridge=bridge-lan1 interface=bond-lan1 disabled=no
```

```
Step 4. [admin@br2] > ip address add interface=bridge-lan1 address=192.168.1.3/24
network=192.168.1.0 disabled=no
```

```
Step 5. [admin@br2] > interface vrrp add name=vrrp-lan1 mtu=1504 interface=bridge-lan1 vrid=1
priority=9 interval=2 preemption-mode=yes authentication=simple password=mypass version=2
disabled=no
```

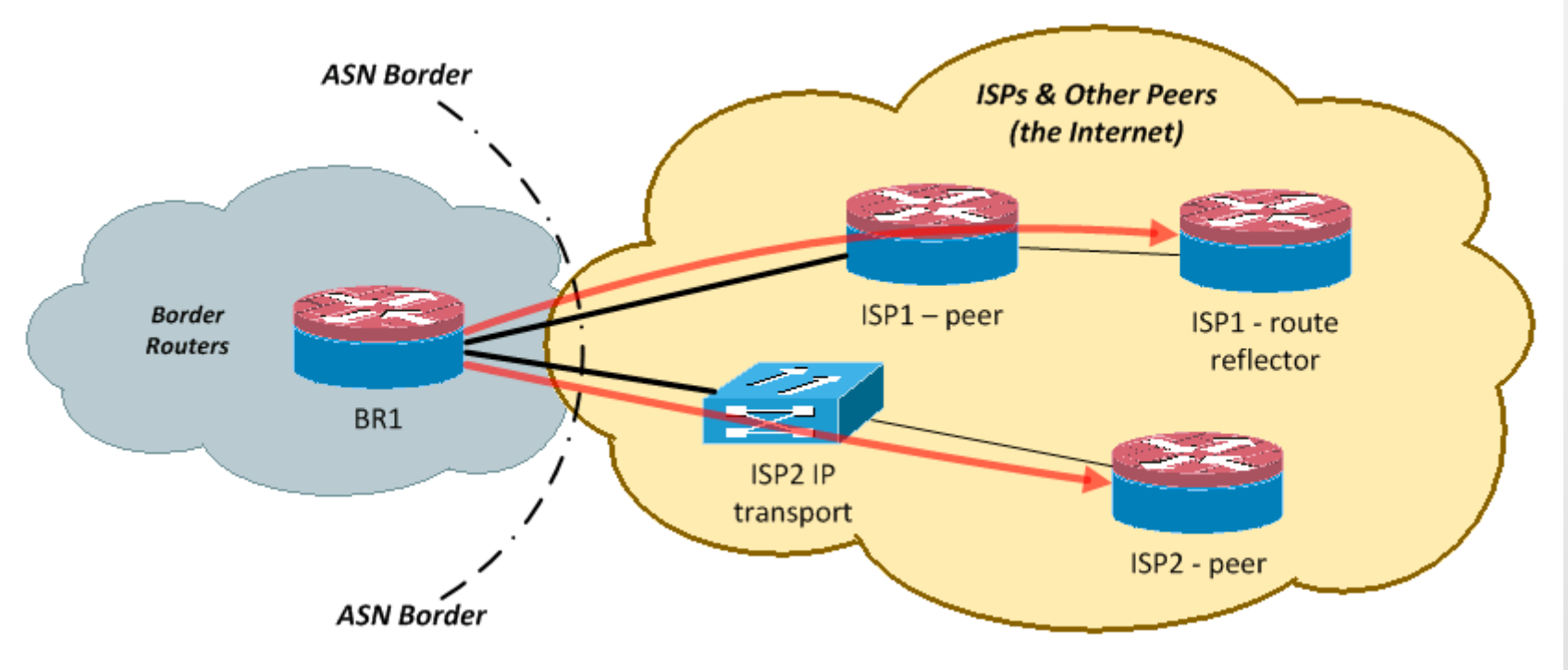
```
Step 6. [admin@br2] > ip address add interface=vrrp-lan1 address=192.168.1.1/32 network=192.168.1.1
disabled=no
```

Upstream issues & rerouting

Sometimes our upstream providers or our peers loose BGP connectivity but the links are up due to intermediary equipment.

We also have a similar situation when one of our peers is a router reflector, requiring a previous BGP session to be established for the communication to be able to take place.

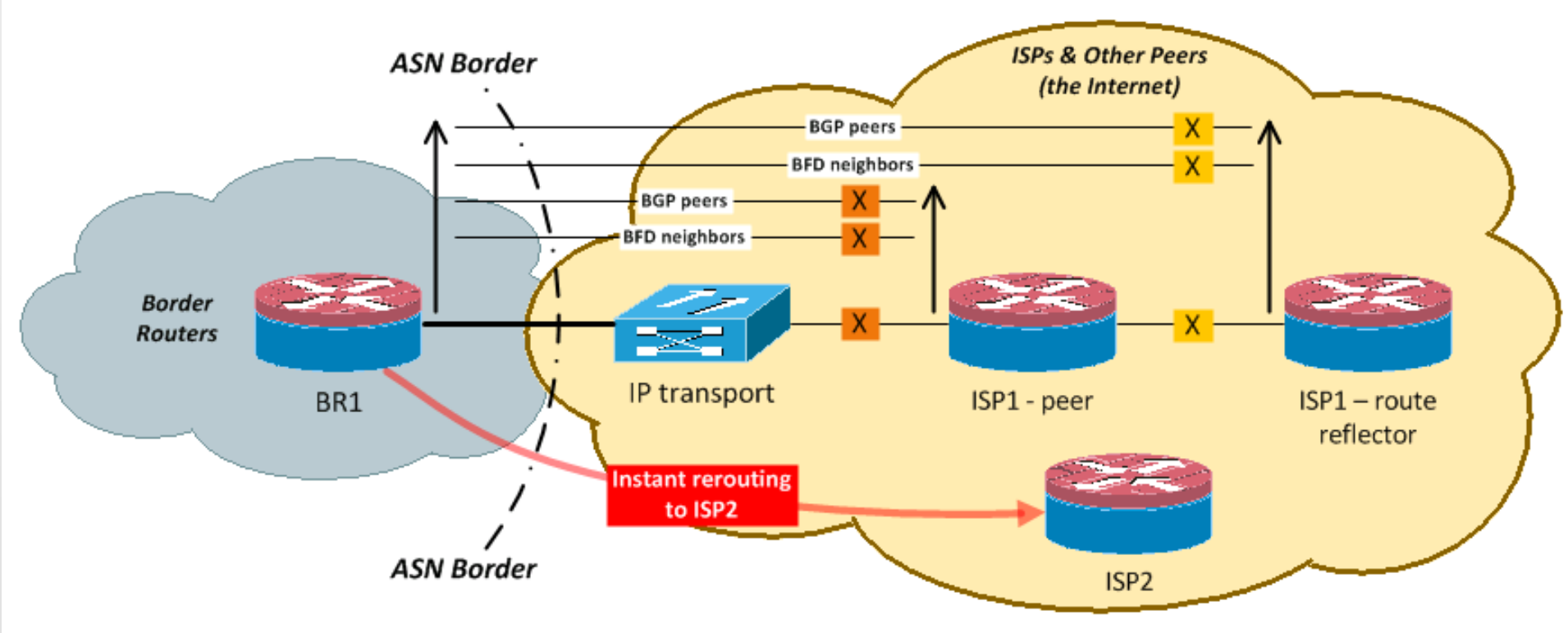
Link detection will fail. The gateways will be up even though the peering sessions are down.



The importance of BFD

So what do we do if we want to have instant rerouting and Layer 3 detection? RouterOS provides us with BFD (Bidirectional Forwarding Detection).

```
[admin@br1] > routing bfd interface set 0 interval=0.2 min-rx=0.2 multiplier=10 disabled=no
[admin@br1] > routing bgp peer set isp1-peer use-bfd=yes disabled=no
[admin@br1] > routing bgp peer set isp1-route-reflector use-bfd=yes disabled=no
```



Protecting the network

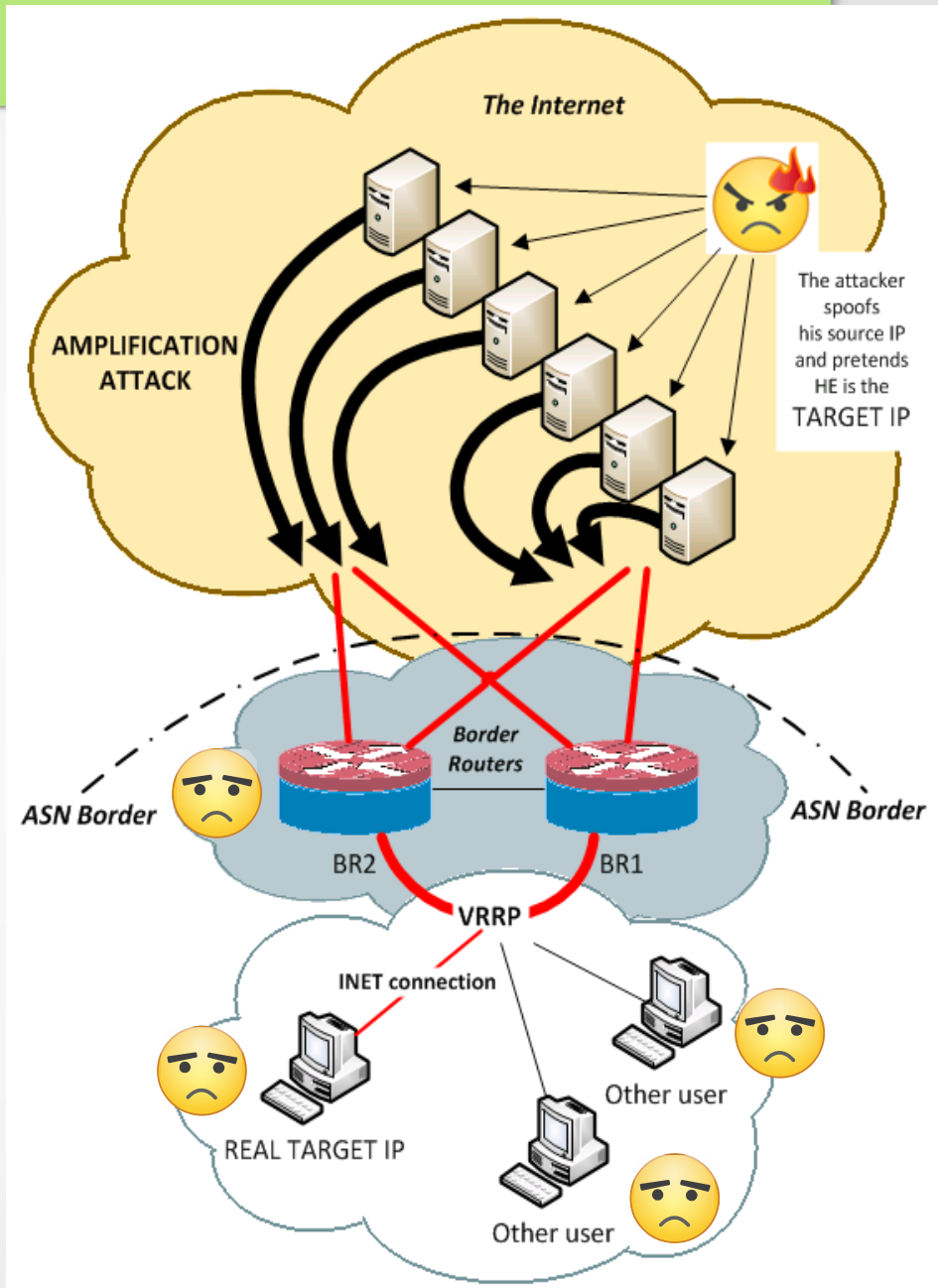
For understanding how firewall filtering works I recommend:

https://www.frozentux.net/iptables-tutorial/iptables-tutorial.html

For theoretical analysis regarding BGP and techniques for securing RouterOS watch:

US14: MikroTik RouterOS Security and BGP by Tom Smyth

http://www.tiktube.com/video/JmiE3cCFdDLCmIIJLnIwKxlrIlHoKDqp=



Protecting the network

What ACTUALLY happens?

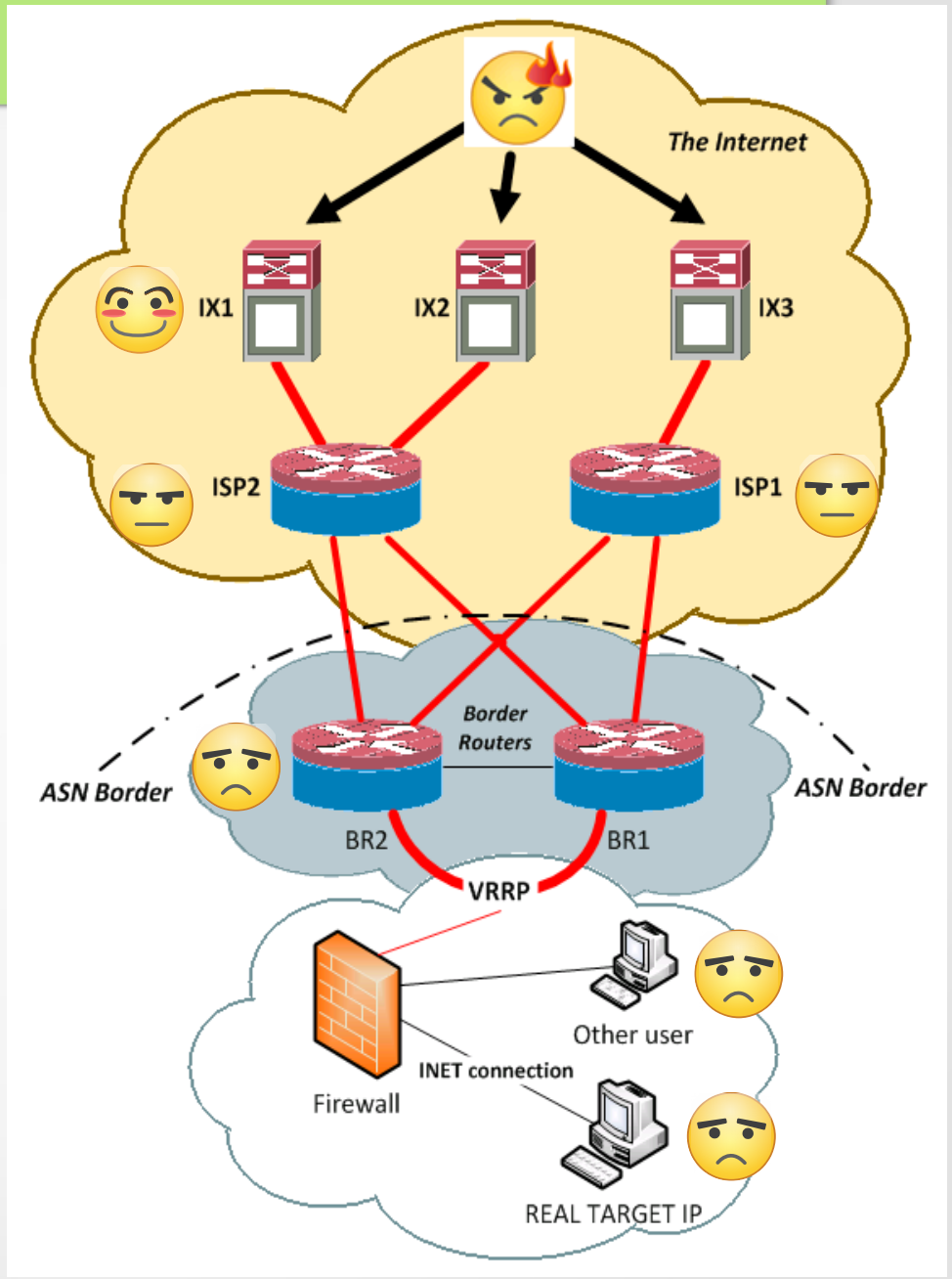
The affected infrastructure is in fact much larger and starts at the IX where the ISP makes his peering arrangements with other ISPs.

So not only OUR network is affected, but the quality degrades also on our upstream transport.

What can we do to protect the network and maintain high uptime?

„Black hole filtering refers specifically to dropping packets at the routing level, usually using a routing protocol to implement the filtering on several routers at once, often dynamically to respond quickly to distributed denial-of-service attacks."

- Source Wikipedia



Protecting the network

Enter unattended IX level filtering

To minimize downtime we need:

- To understand that the target IP will need to be null routed as far away from our network as possible
- To automatically detect incoming attacks towards our network
- To automatically set up black hole routes for redistribution towards our upstream providers
- To have a route redistribution mechanism up to the ISP level (this is absolutely normal behavior of border routers)
- To have a route redistribution mechanism to the IX level (this will be generally be managed by the ISPs)

Detection

The simplest and most effective way to detect DDOS is to monitor packet rates towards destinations in our network.

And more important, you need to monitor packet rates PER destination IP.

Protecting the network

First add the public network IPs (/32) to the address lists on both border routers (the presentation will show the setup only for one router, you should mirror the second configuration on br2). You can also use scripting for adding multiple IP blocks at once to the address lists.

```
[admin@br1] > ip firewall address-list add address=A.B.C.D comment="my customer"
list=MY_CUSTOMER
```

Set up monitoring for every upstream interface (30Kpps total works good for 100Mbps links, you can lower or raise this as you require):

```
[admin@br1] > ip firewall mangle add in-interface=ISP1 dst-address-list=MY_CUSTOMER
action=jump jump-target=monitoring
[admin@br1] > ip firewall mangle add in-interface=ISP2 dst-address-list=MY_CUSTOMER
action=jump jump-target=monitoring
```

Set up filtering - total incoming packet rate per destination:

```
[admin@br1] > ip firewall mangle add chain=monitoring dst-limit=15000/1s,15000,dst-
address/90s action=return
[admin@br1] > ip firewall mangle add chain=monitoring action=add-dst-to-address-list
address-list=NULL_ROUTE address-list-timeout=15m
```

Protecting the network

Drop all traffic over the limit as quickly as possible with minimal effort:

```
[admin@br1] > ip firewall mangle add chain=monitoring action=mark-routing
new-routing-mark=null_route passthrough=no
[admin@br1] > ip route rule add routing-mark=null_route action=drop
```

Drop any traffic that might go through from inside our upstream networks towards all null routed destinations in the filter table:

```
[admin@br1] > ip firewall filter add in-interface=ISP1 protocol=tcp dst-
address-list=NULL_ROUTE action=tarpit
[admin@br1] > ip firewall filter add in-interface=ISP2 protocol=tcp dst-
address-list=NULL_ROUTE action=tarpit
[admin@br1] > ip firewall filter add in-interface=ISP1 dst-address-
list=NULL_ROUTE action=drop
[admin@br1] > ip firewall filter add in-interface=ISP2 dst-address-
list=NULL_ROUTE action=drop
```

Protecting the network

1. Create a small script to initialize two very important variables on boot.

   This is required to make sure we don't run the same actions for multiple times for the same IP before the first action has concluded.

2. Create a script that will check for dynamically added IPs to the NULL_ROUTE address list.

   This is required so that the BRs will automatically filter the IPs that are attacked.

3. Create a script that will check for manually added IPs to the NULL_ROUTE address list.

   This is required so that we can manually filter IPs if we desire.

4. Create a script that will check for expired or manually removed NULL_ROUTE IPs and take required action.

   This is required so that the unattended filtered IPs will not remain filtered forever.

5. Create a script that will automatically resend our advertisments.

   This is required as sometimes the removal of nulled IPs does not take place without resending it manually.

We then have to set the scheduler to run these scripts automatically (scripts are running fine on RouterOS v5).

Protecting the network

1. Initialize variables (set scheduler at startup only)

```
#initvars
:global canrun 1;
:global candynrun 1;
```

2. Check for unattended null routed IPs (set scheduler at X seconds for every 2X thousand IPs in the address list)

```
:global candynrun;
:if ( $candynrun=0 ) do={ :error candynrun0; };
:set candynrun 0;
:local NULLEDADDR;
:local FOUND;
:set FOUND 1;
:foreach i in [/ip firewall address-list find (list=NULL_ROUTE and dynamic=yes)] do=[ \
  :set FOUND 0;
  :set NULLEDADDR [/ip firewall address-list get $i address];
  :foreach j in [/ip route find (dst-address="$NULLEDADDR/32" and bgp-communities="YOUR_ASNUMBER_HERE:YOUR_IX_COMMUNITY_HERE"
and (comment="blackhole" or comment="blackhole_dyn"))] do [ \
  :set FOUND 1;
  ];
  if ( $FOUND=0 ) do={ \
    /ip route add dst-address="$NULLEDADDR" gateway=bridge-lan1 comment="blackhole_dyn" bgp-
communities="YOUR_ASNUMBER_HERE:YOUR_IX_COMMUNITY_HERE"
    /ip firewall address-list add address="$NULLEDADDR" list="AUTO_CHECK" comment="Do not manually remove unless you know
what you are doing."
    :log info "New detection: The IP $NULLEDADDR has been automatically null routed with blackhole_dyn comment in routing
table.";
  };
];
:set candynrun 1;
```

Protecting the network

3. Check for manually null routed IPs

(set scheduler at X seconds for every 2X thousand IPs in the address list, delay it by X/2 seconds versus the previous script)

```
# If under flood immediately after a reboot re-init as nil now
:global canrun;
# If the script is running then don't run it again, it will create duplicate entries.
if ( $canrun=0 ) do={ :error canrun0; };
# If it's not already running then set it to running now
:set canrun 0;
# Init local vars
:local NULLEDADDR;
:local FOUND;
:set FOUND 1;
# Check every null routed IP in the address list
:foreach i in [/ip firewall address-list find (list=NULL_ROUTE and dynamic=no)] do=[ \
  :set FOUND 0;
  :set NULLEDADDR [/ip firewall address-list get $i address];
# Check for already existent route
# Replace YOUR_ASNUMBER_HERE and YOUR_IX_COMMUNITY_HERE with your own correct settings
  :foreach j in [/ip route find (dst-address="$NULLEDADDR/32" and bgp-communities="YOUR_ASNUMBER_HERE:YOUR_IX_BLACKHOLE_COMMUNITY_HERE" and
(comment="blackhole" or comment="blackhole_dynamic"))] do [ \
  :set FOUND 1;
  ];
  if ( $FOUND=0 ) do={ \
# It is important not to actually null route the IP on our network as that would cut off access from within our network to the destination
IP, but to filter its incoming traffic from the filter table.
    /ip route add dst-address="$NULLEDADDR" gateway=bridge-lan1 comment="blackhole_metro" bgp-
communities="YOUR_ASNUMBER_HERE:YOUR_IX_BLACKHOLE_COMMUNITY_HERE"
    /ip firewall address-list add address="$NULLEDADDR" list="AUTO_CHECK" comment="Do not manually remove unless you know what you are
doing."
    :log info "New detection: The IP $NULLEDADDR has been manually null routed with blackhole comment in routing table.";
  };
];
:set canrun 1;
```

Protecting the network

4. Check for expired NULL_ROUTE IPs (set the scheduler to run at 15 minutes)

```
:local CHECKED;
:local STILLNULLED;
:foreach i in [/ip firewall address-list find list=DYNAMIC_CHECK] do=[ \
  :set CHECKED [/ip firewall address-list get $i address];
  :foreach j in [/ip firewall address-list find list=NULL_ROUTE] do=[ \
    :set STILLNULLED [/ip firewall address-list get $j address];
    if ( $STILLNULLED = $CHECKED ) do={ \
      :set CHECKED 0;
    };
  ];
  if ( $CHECKED!=0 ) do={ \
    :foreach k in [/ip route find (dst-address="$CHECKED/32" and bgp-
communities="YOUR_ASNUMBER_HERE:YOUR_IX_COMMUNITY_HERE" and (comment="blackhole" or comment="blackhole_dyn"))] do [
\
      :set CHECKED [/ip route get $k dst-address];
      /ip route remove $k;
      :log info "The null route for $CHECKED has been automatically removed.";
    ];
    /ip firewall address-list remove $i;
  };
];
```

5. Resend advertisments (set the scheduler to run at 15 minutes, delay it by 1 minute from the previous script)

```
/routing bgp peer resend-all;
```

Protecting the network

Things to consider

- As the routes we've set up for filtering get redistributed by using the routing filter mechanism you will have to append the proper blackhole BGP communities specified by your upstream providers. This is how they will actually filter the traffic. Setting this up is quite complex and depends on each peering arrangement, there is no step-by-step solution.

- Peer only on private IPs, use /30 or /29 subnets. This will increase the security of your border routers.

- For proper protection null routing must take place only at the ISP or the IX level. The target IP appears in our routing table as a /32 route that is identifiable by communities and also by comment and NOT an actual black hole route.

Protecting the network

Other things to consider

- The traffic that is coming directly from inside your upstream networks should be droped at the filter level of the firewall. You can always work directly with your ustream provider to determine the source and take it out of action.

- Always IX null route all the gateways, vrrp IPs, network & broadcast addresses through this mechanism.

- Make sure that you properly distribute the routes to your peers and that you also distribute them through an iBGP session to your backup router. The backup router should also redistribute the routes to its upstream providers to further filter the traffic.

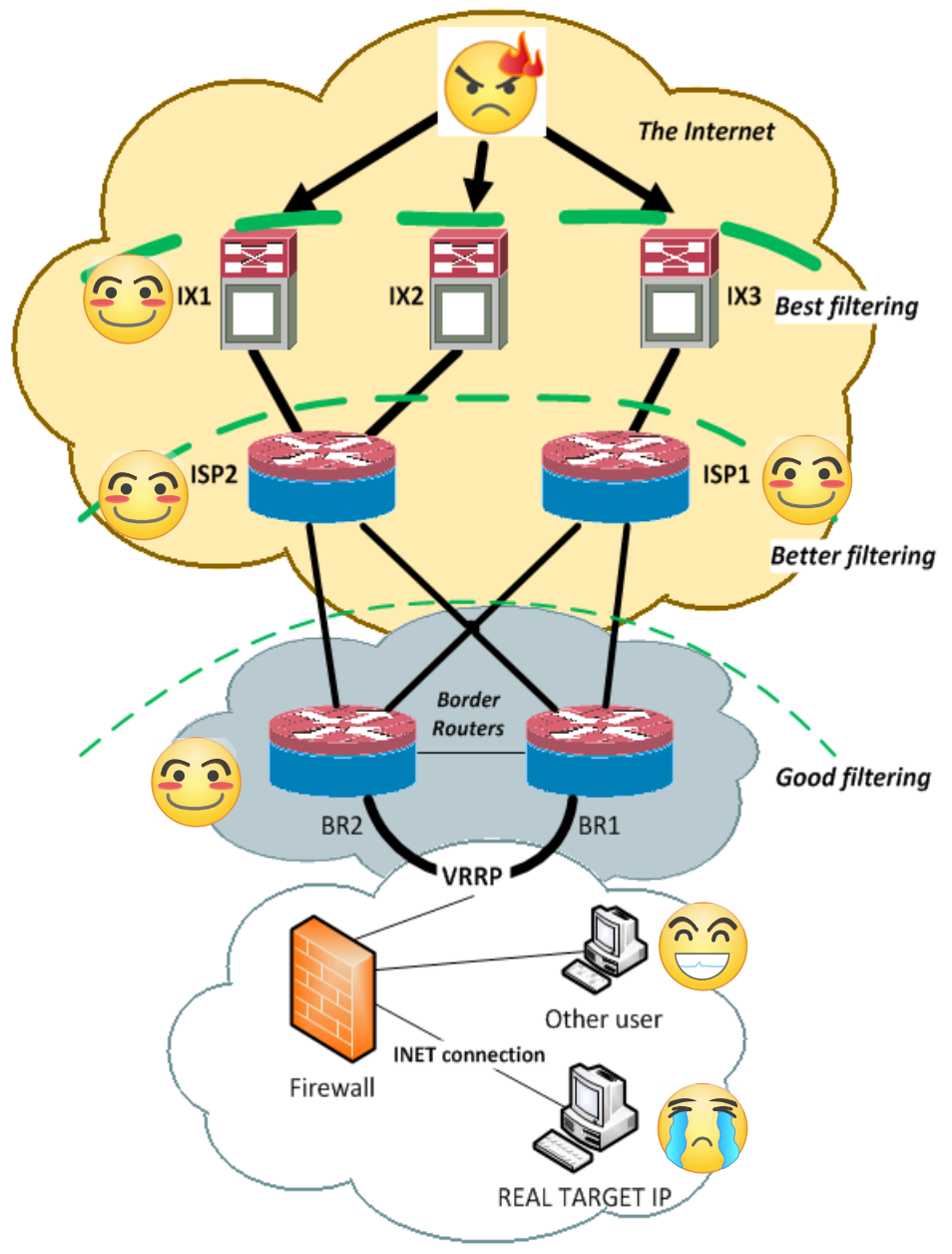
Protecting the network

So what does IX level filtering do?

The traffic has been completely cut-off at affected Internet Exchange Points, far away from our network.

If there is any remaining attack traffic from private peerings it can be filtered by our ISPs border routers or mitigated by firewall systems. Our ISPs are also happy that they don't have to route attack traffic.

If there is any remaining traffic from inside our upstream networks we can filterd it directly on our border routers or through our firewall systems.



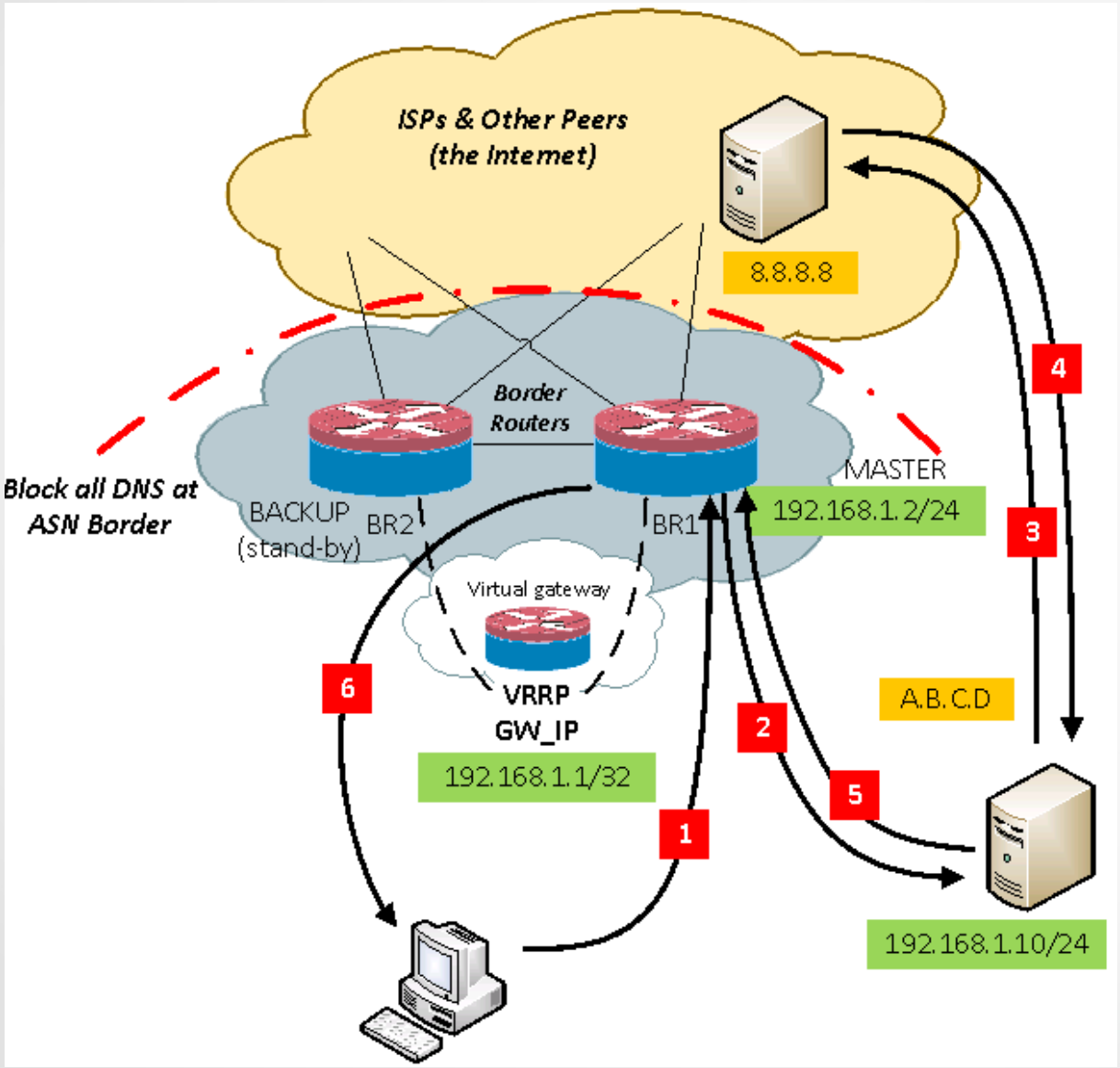
Protecting the Internet from us

DNS misconfigurations (open resolvers) are a big problem.

When used as a DNS server on our network, RouterOS is an open resolver by default.

The DNS server on a Mikrotik device is limited, when used intensively (for example with advertising servers) it will become slow to multiple simultaneous queries and sometimes timeout.

Protecting the Internet from us



DNS requests should be redirected to internal pdns servers.

pdns servers have inbuilt protection against IP spoofing.

Protecting the Internet from us

pdns-recursor

Modern, advanced and high performance recursing/non authoritative name server

On a RH/CentOS distribution installing the server is done by: `yum install pdns-recursor`

Sample /etc/pdns-recursor/recursor.conf

```
setuid=pdns-recursor
setgid=pdns-recursor
allow-from=192.168.1.0/24
daemon=yes
etc-hosts-file=/etc/hosts
local-address=127.0.0.1,192.168.1.10
pdns-distributes-queries=yes
query-local-address=A.B.C.D
version-string=Mikrotik v1.0 DNS
```

Protecting the Internet from us

Step 1. Filter all DNS traffic towards the router from untrusted sources:

```
/ip firewall filter add action=drop chain=input disabled=no dst-
port=53 in-interface=!br0 protocol=udp
```

Step 2. Mark all incoming connections and set them to be balanced:

```
/ip firewall mangle

add action=mark-connection chain=prerouting comment="DNS RELAY1"
disabled=no dst-address=192.168.1.1 dst-port=53 in-interface=br0 new-
connection-mark=forwarded-dns1 passthrough=yes per-connection-
classifier=both-addresses-and-ports:2/0 protocol=udp

add action=mark-connection chain=prerouting comment="DNS RELAY2"
disabled=no dst-address=192.168.1.1 dst-port=53 in-interface=br0 new-
connection-mark=forwarded-dns2 passthrough=yes per-connection-
classifier=both-addresses-and-ports:2/1 protocol=udp
```

Protecting the Internet from us

Step 3. Use NAT to balance connections to the real DNS servers:

```
/ip firewall nat

add action=dst-nat chain=dstnat connection-mark=forwarded-dns1
disabled=no to-addresses=192.168.1.10

add action=src-nat chain=srcnat connection-mark=forwarded-dns1
disabled=no to-addresses=192.168.1.1

add action=dst-nat chain=dstnat connection-mark=forwarded-dns2
disabled=no to-addresses=192.168.1.11

add action=src-nat chain=srcnat connection-mark=forwarded-dns2
disabled=no to-addresses=192.168.1.1
```

Step 4. Set router resolving towards internal DNS servers:

```
/ip dns set allow-remote-requests=yes cache-max-ttl=1w cache-size=2048KiB
max-udp-packet-size=512 servers=192.168.1.10,192.168.1.11
```

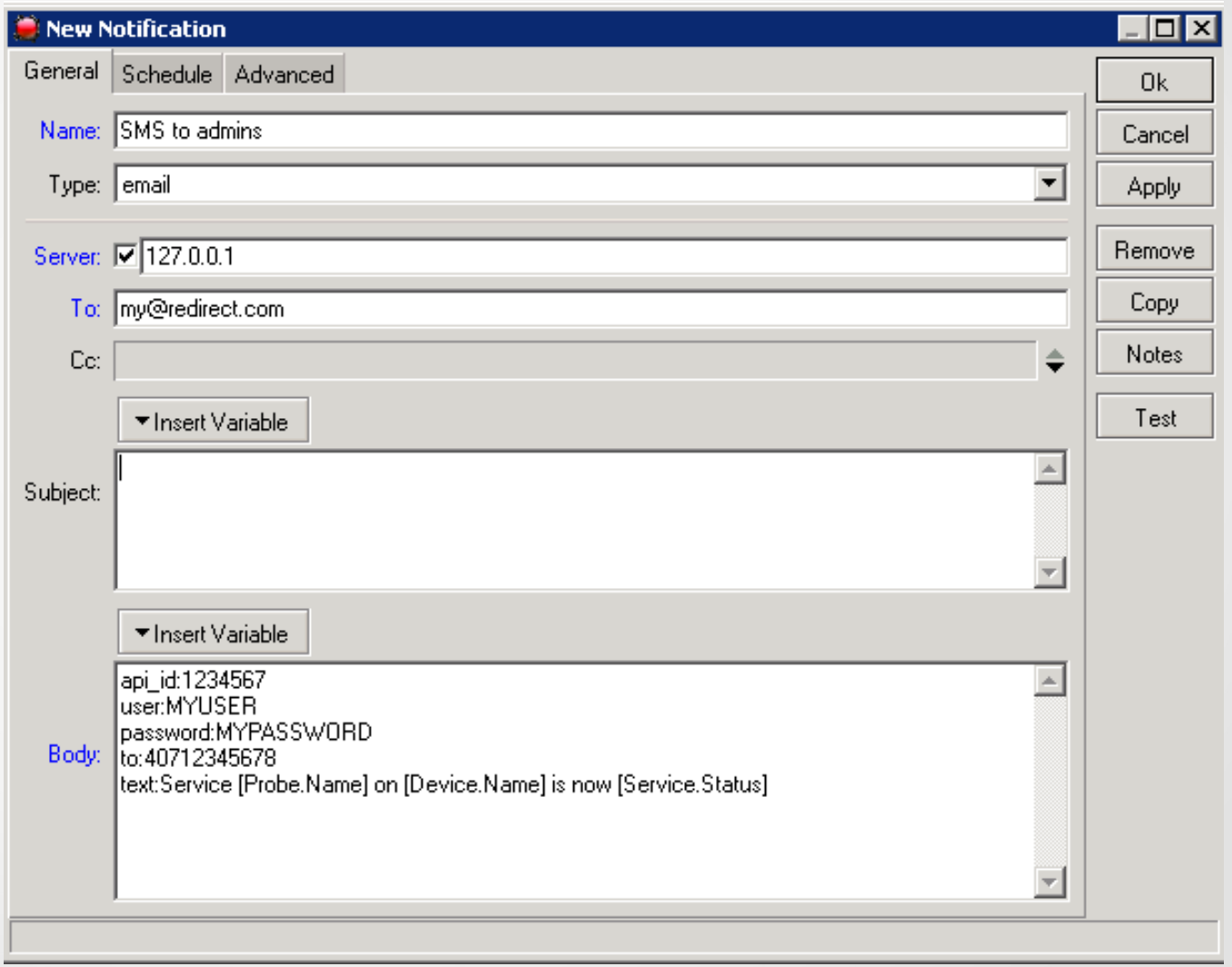
Setting up SMS proactive monitoring in The Dude

Create a Clickatell.com API account.

Create a new e-mail API in the Clickatell.com account.

Create a new e-mail redirect on your mail server towards: sms@messaging.clickatell.com

Create a new notification type in The Dude.



To err is human but customers do not forgive

Even if you are covered by SLA your customers will not take downtime lightly.

Downtime is the primary factor that consumers give up on their service providers. Service quality is usually a factor when evaluating the work of a network administrator by employers.

Get certified. You will never know everything but you will definitely know more this way. Your customers and your employers will trust your skills more and you will be more skilled at providing higher uptime on your managed networks.

Conclusions & discussion

Network uptime is the percentage of time that our network is functioning properly versus the total period of time.

Having 100% availability for border routers is a difficult feat and should not be taken lightly. It should be properly planned, executed exactly as planned and improved in time.

Questions?

Have fun routing the world!

```
  MMM      MMM       KKK                          TTTTTTTTTTT      KKK
  MMMM    MMMM       KKK                          TTTTTTTTTTT      KKK
  MMM MMMM MMM  III  KKK  KKK  RRRRRR     OOOOOO      TTT     III  KKK  KKK
  MMM  MM  MMM  III  KKKKK     RRR  RRR  OOO  OOO     TTT     III  KKKKK
  MMM      MMM  III  KKK KKK   RRRRRR    OOO  OOO     TTT     III  KKK KKK
  MMM      MMM  III  KKK  KKK  RRR  RRR   OOOOOO      TTT     III  KKK  KKK

  MikroTik RouterOS 6.20 (c) 1999-2014       http://www.mikrotik.com/

[?]             Gives the list of available commands
command [?]     Gives help on the command and list of arguments

[Tab]           Completes the command/word. If the input is ambiguous,
                a second [Tab] gives possible options

/               Move up to base level
..              Move up one level
/command        Use command at the base level

[admin@br1] > Thank you!
```