



Welcome

Dynamic Firewalling
by
Barry Higgins

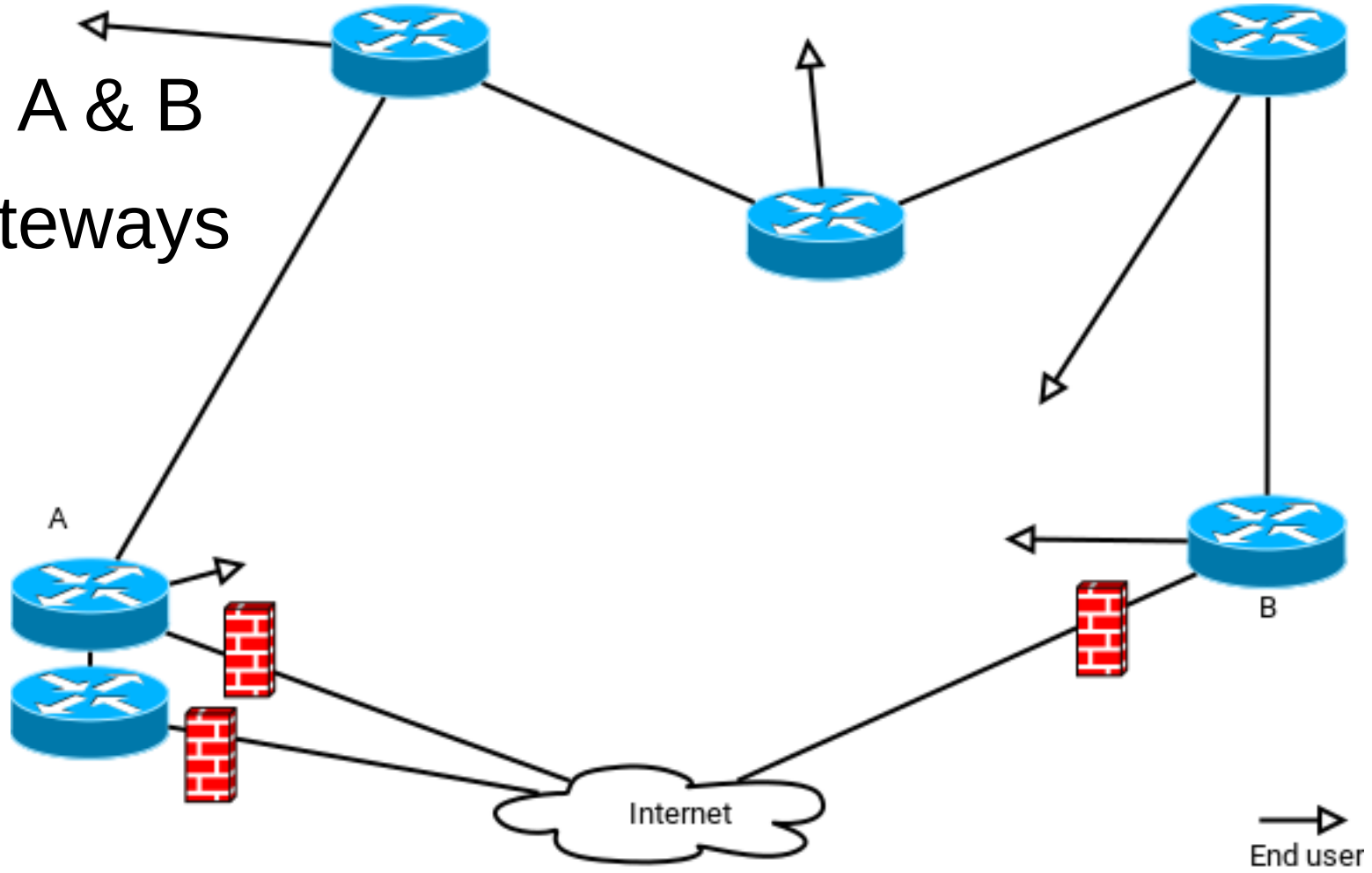


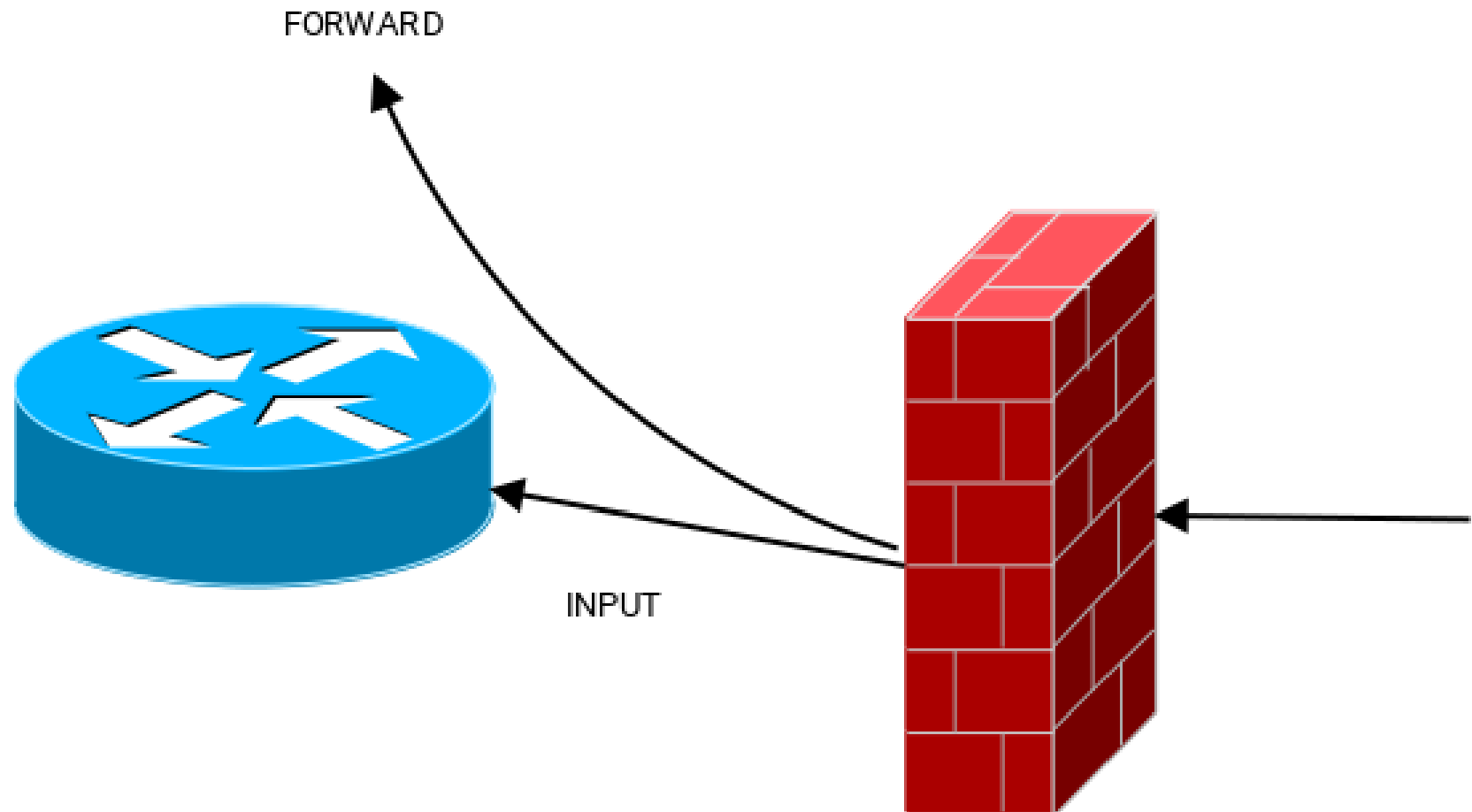
- Hosting
- WISP
- Consultancy
- Mikrotik Training

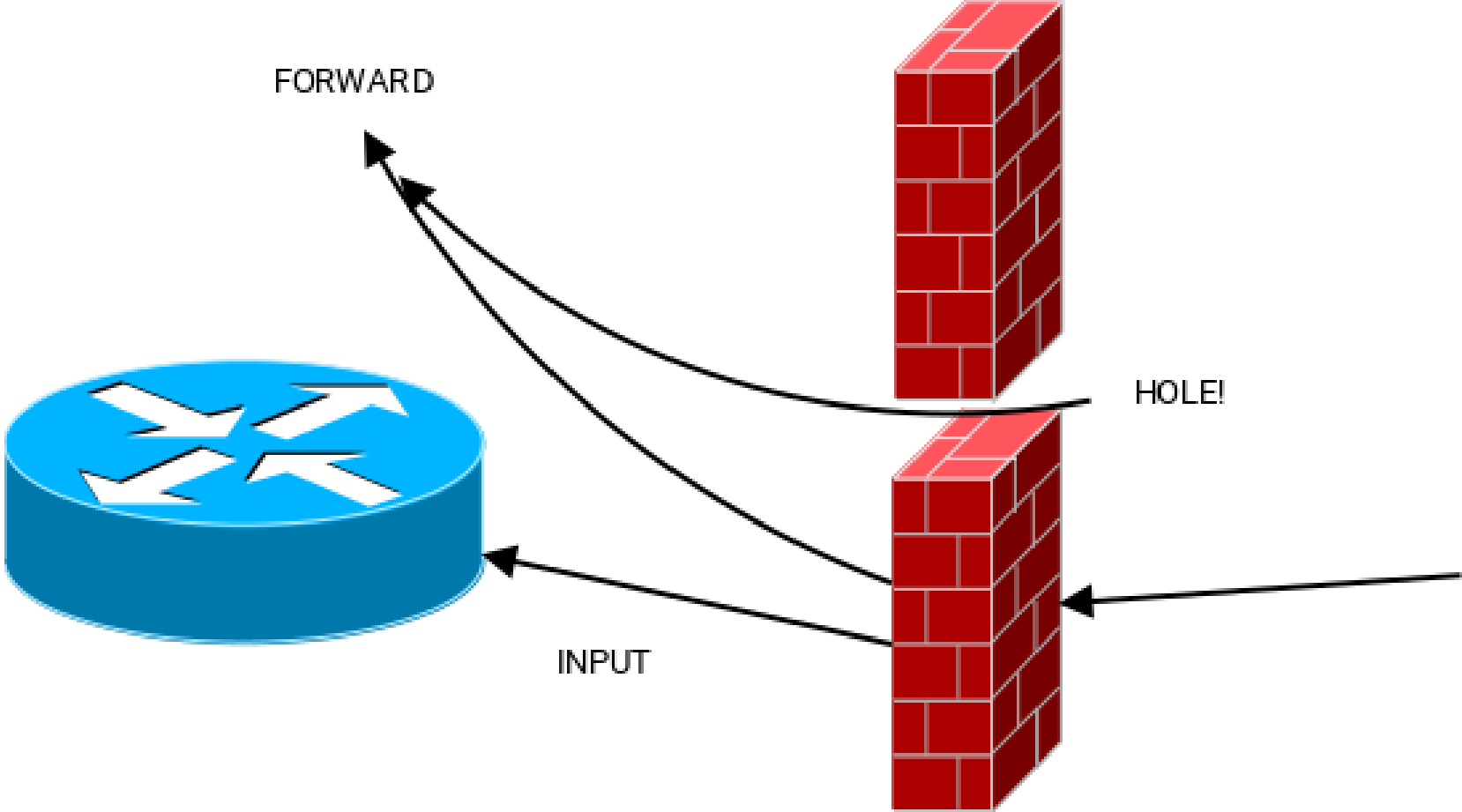
So the problem...

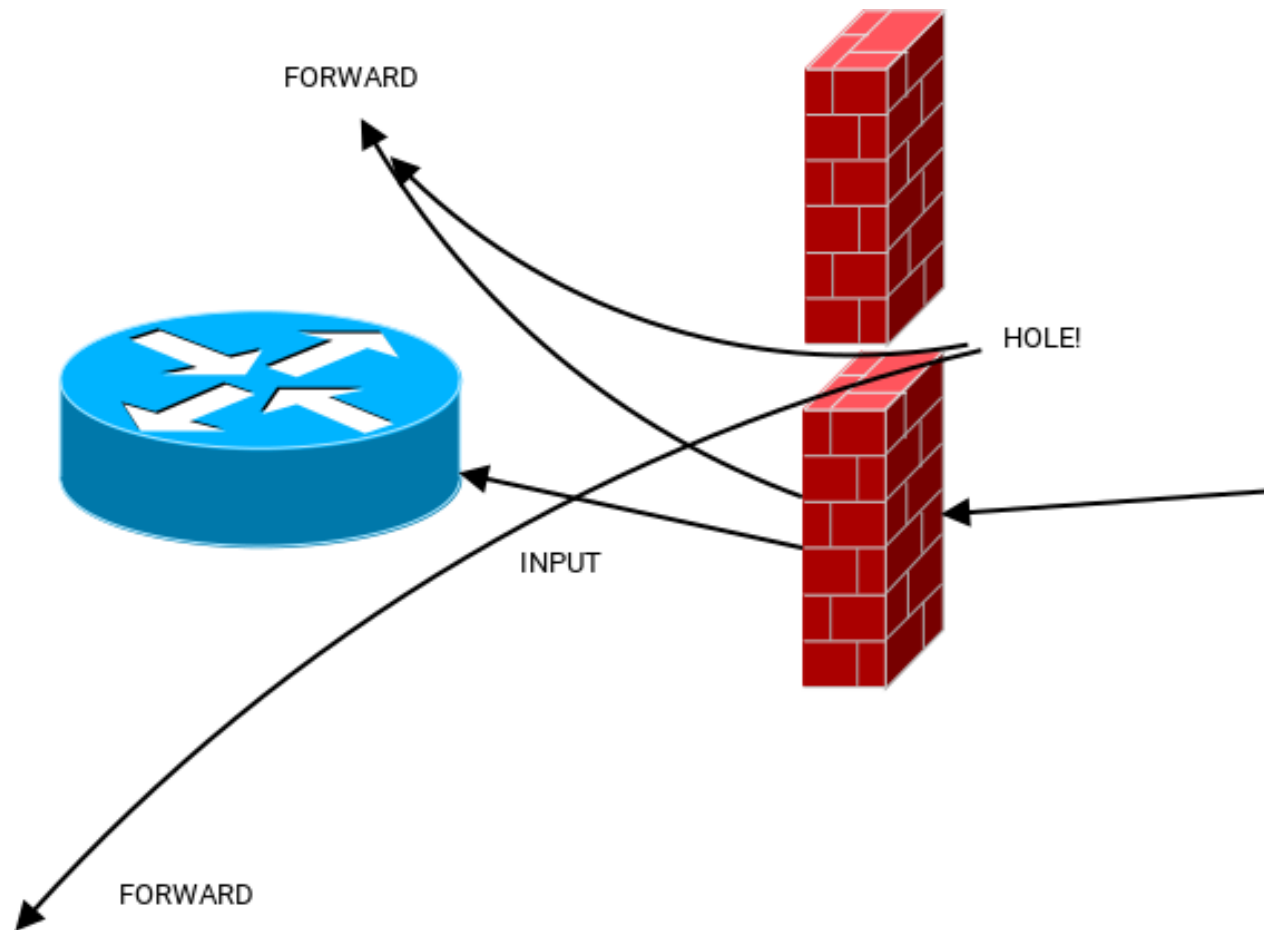
I want to add an extra layer of security from network scans, viral scripts probing servers and protect my WISP end users

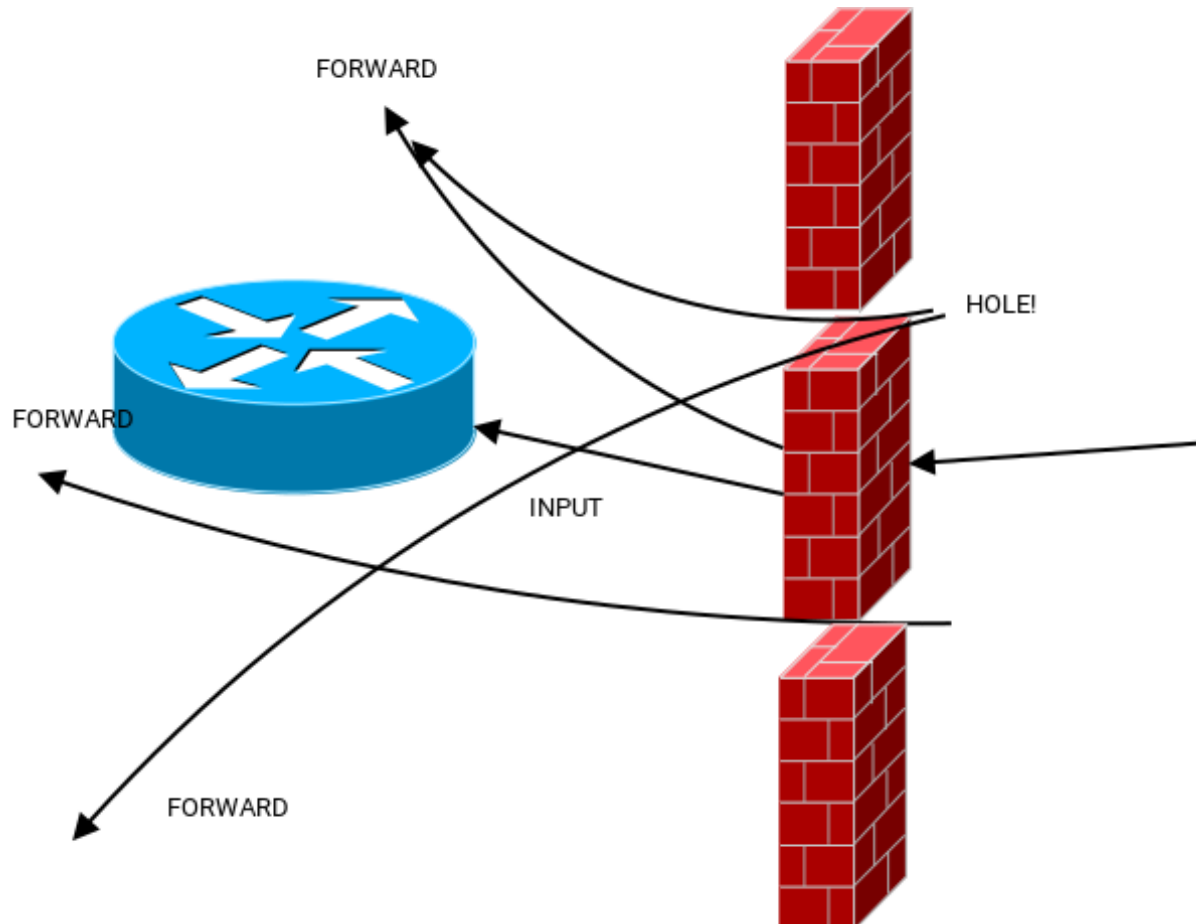
Sites A & B
3 Gateways

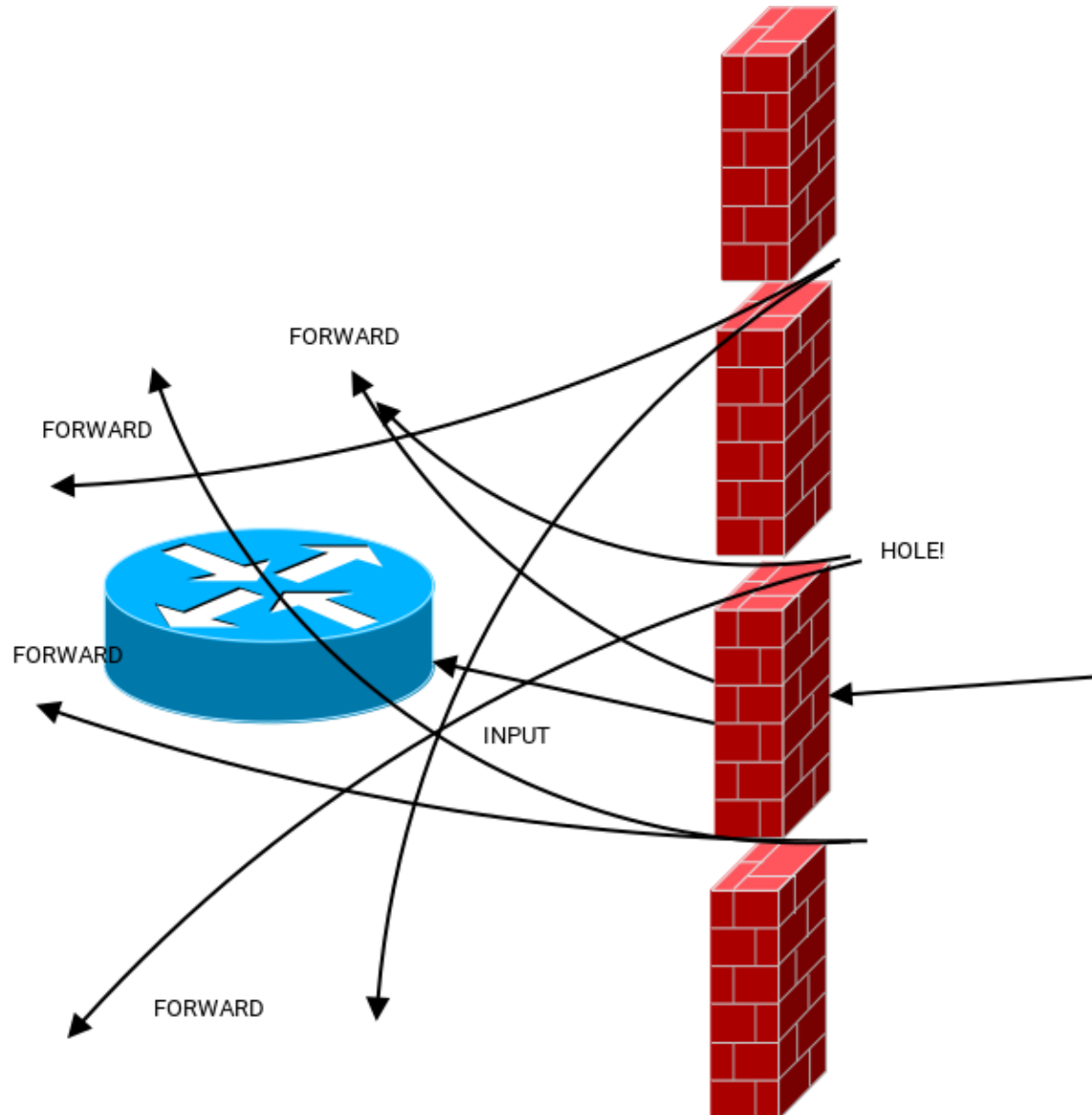




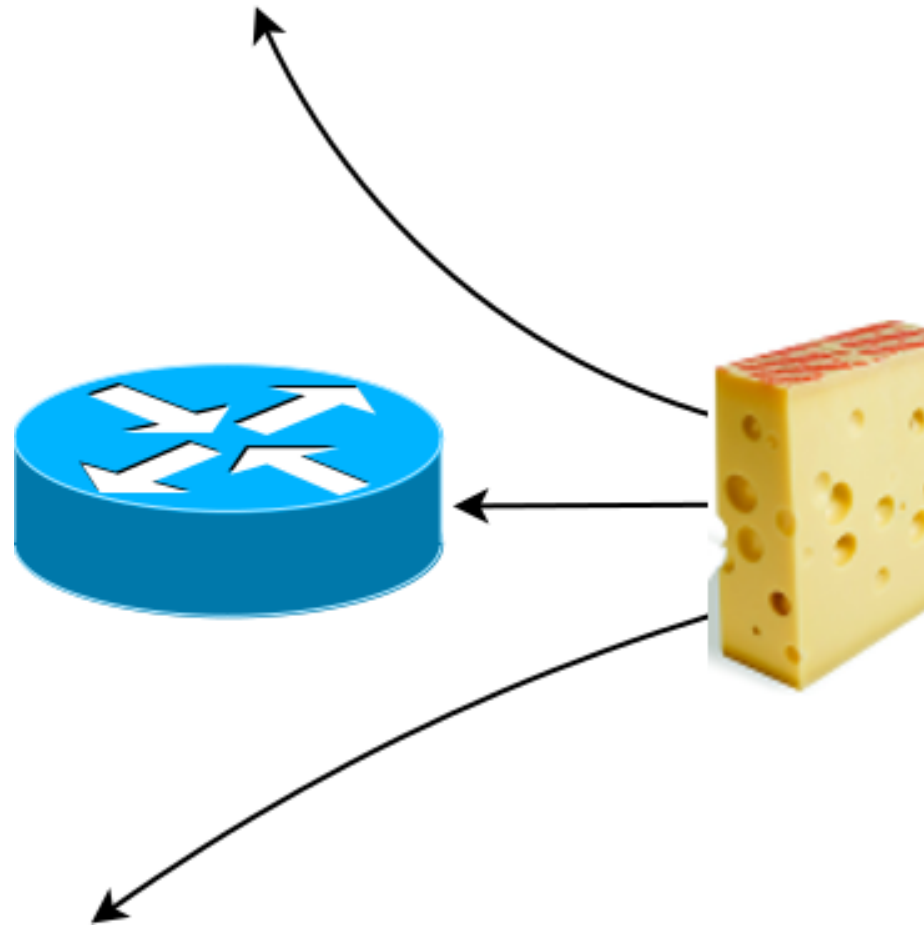








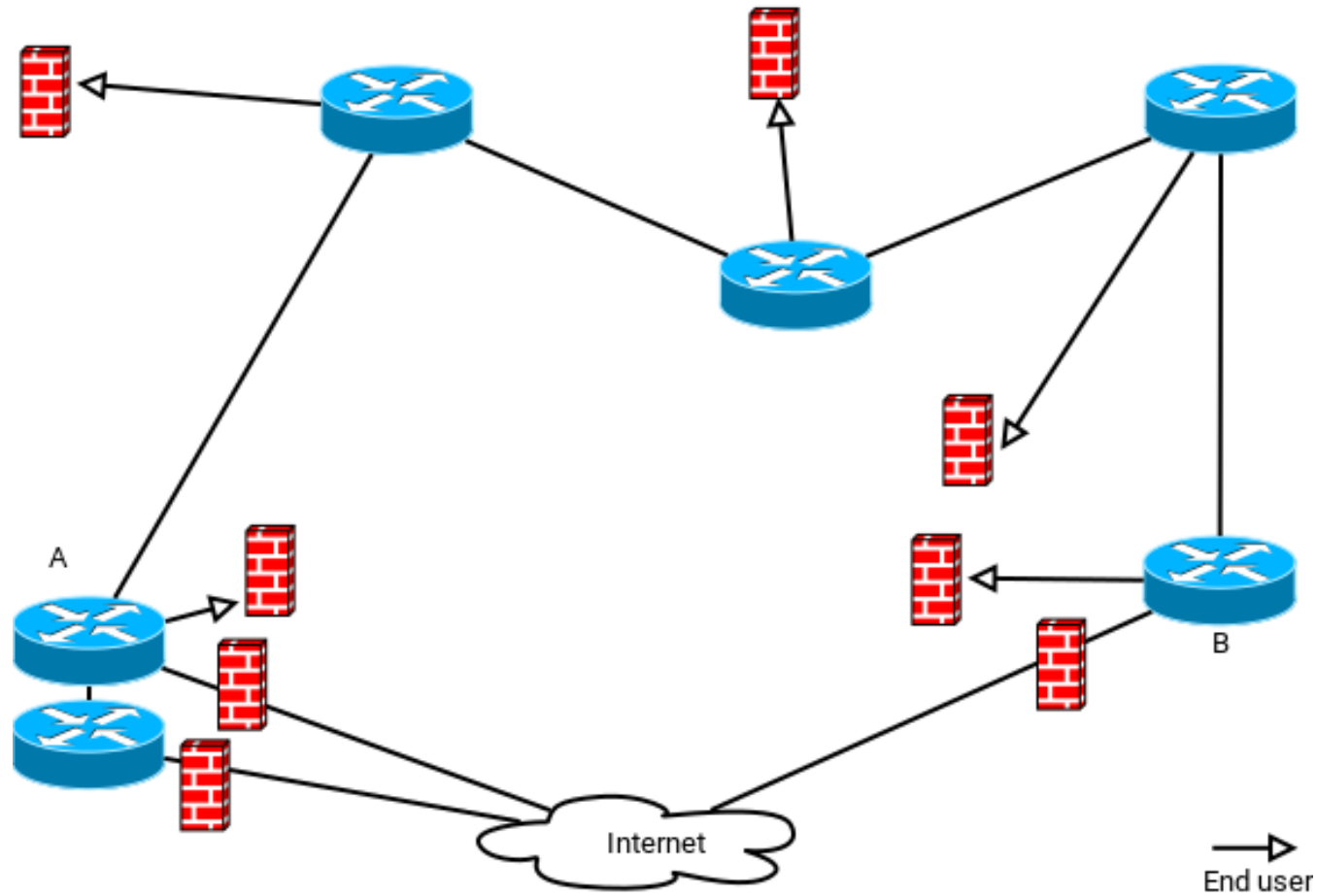
Full of holes!



Solutions?

- Don't allow any ports open in the first place
- Migrate user to another (W)ISP and let them have the problem!
- Or...

Firewall them all!



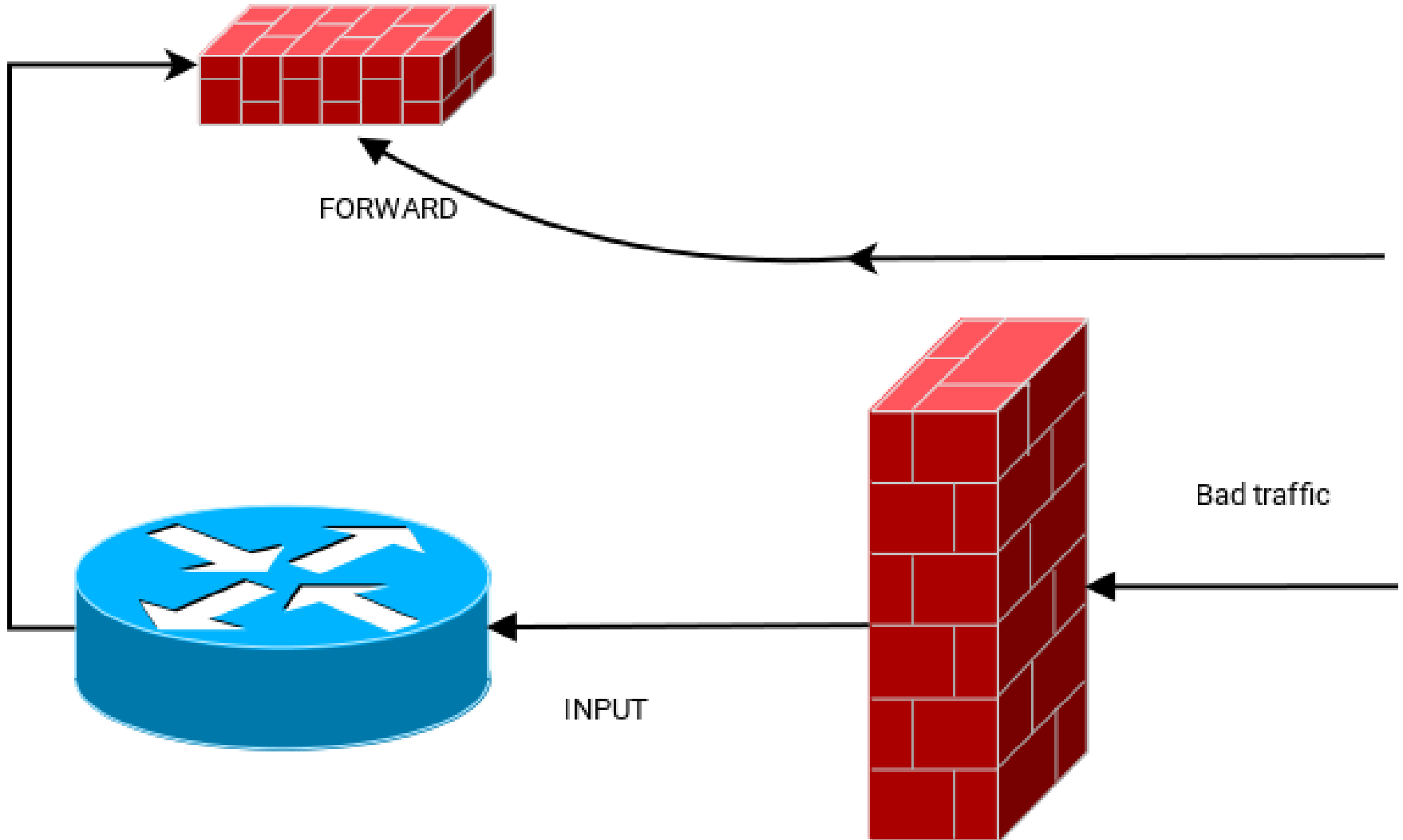
User alert



- Default username
- Default password
- Web access open
- Telnet access open
- DNS enabled



blocked forward traffic based on input chain





<demo site-A>

Firewall

Filter Rules NAT Mangle Service Ports Connections Address Lists Layer7 Protocols

+ - ✓ ✗ 📁 🔍 00 Reset Counters 00 Reset All Counters Find all

#	Action	Chain	Src. Address	Dst. Address	Protocol	Src. Port	Dst. Port	In. Int...	Out. I...	Bytes	Packets
;;; Forward established											
0	✓ acc...	forward								3686.4 KIB	5 931
;;; Allow Forward regardless of rules											
1	✓ acc...	forward								0 B	0
;;; Allow regardless of rules											
2	✓ acc...	input								43.9 KIB	529
;;; drop input from blocked address list											
3	✗ drop	input								217.2 KIB	4 629
;;; drop forward from blocked address list											
4	✗ drop	forward								2056 B	35
;;; Input chain jump to bad traffic check											
5	🔗 jump	input								261.7 MiB	244 860
;;; Check for ICMP											
6	🔗 jump	Bad_traffi...			1 (icmp)					2712 B	37
;;; check for dns											
7	✗ 🔗 jump	Bad_traffi...			17 (udp)		53			57 B	1
;;; check for dns											
8	✗ 🔗 jump	Bad_traffi...			6 (tcp)		53			0 B	0
;;; Check for port scanning											
9	🔗 jump	Bad_traffi...			6 (tcp)					261.7 MiB	244 328
;;; Check for DoS											
10	🔗 jump	Bad_traffi...								261.7 MiB	244 790
;;; Check for incoming BOGONS											
11	🔗 jump	Bad_traffi...								261.7 MiB	244 693
;;; Check for common M\$ attacks											
12	🔗 jump	Bad_traffi...								261.7 MiB	244 670
;;; Check for voip probe											
13	🔗 jump	Bad_traffi...								261.7 MiB	244 670
;;; Check for brute force attacks											
14	🔗 jump	Bad_traffi...								261.7 MiB	244 670
;;; Allow ICMP 0:0 5/sec burst 5											

31 items (1 selected)

Blacklist

Firewall

Filter Rules NAT Mangle Service Ports Connections Address Lists Layer7 Protocols

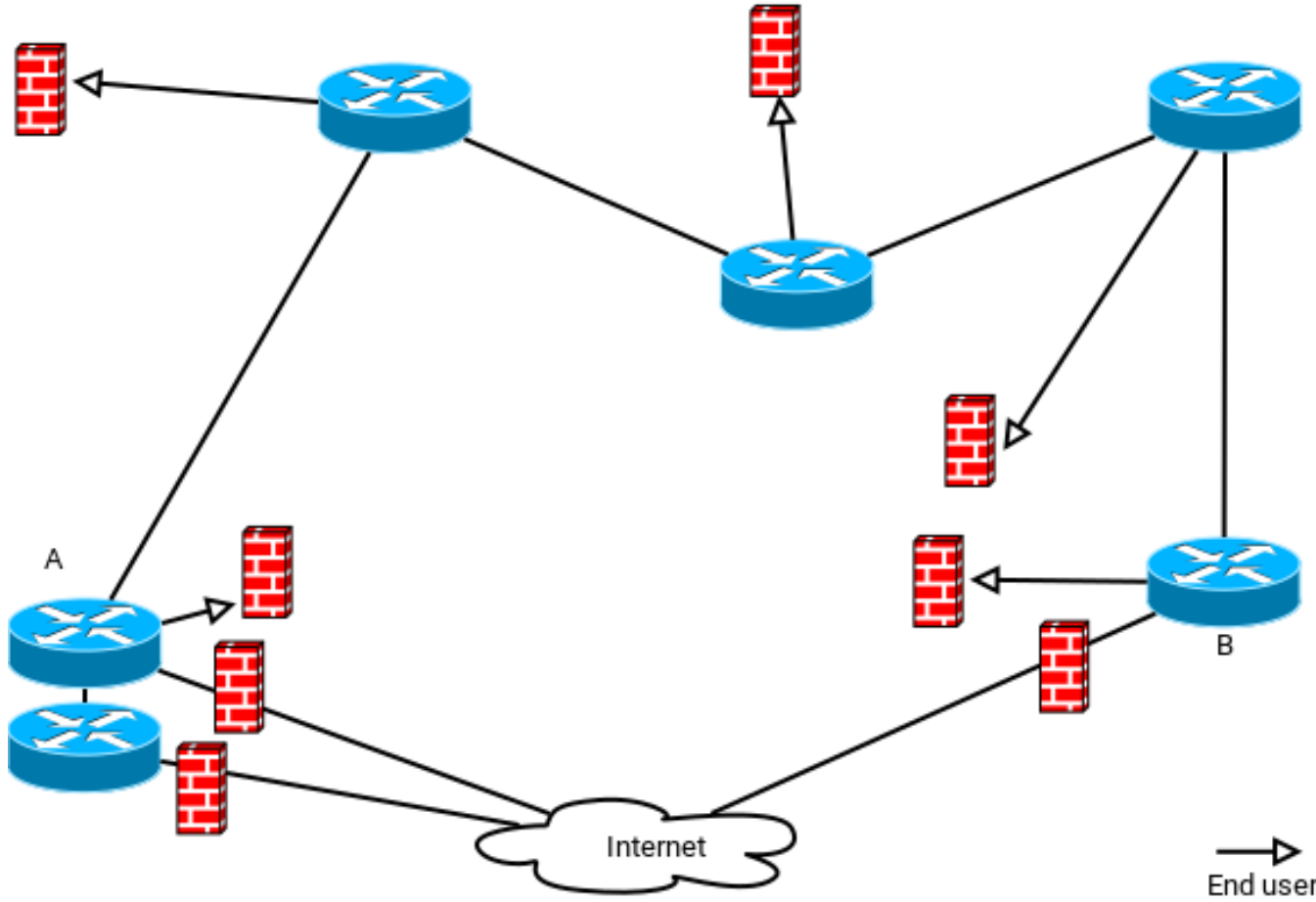
+ - ✓ ✗ 📄 🔍

	Name ▲	Address	Timeout	
	Whitelist	192.168.74.1		
D	attempt#1	192.168.0.254	00:00:46	
D	attempt#2	192.168.0.254	00:00:48	
D	attempt#3	192.168.0.254	00:00:50	
D	blacklist	192.168.0.254	00:09:51	

Logging

Log			
Freeze		all	
			192.168.0.254:48098->192.168.0.1:8291, len 60
Feb/21/2016 21:19:39	memory	system, error, critical	login failure for user admin from 192.168.0.254 via winbox
Feb/21/2016 21:19:40	memory	firewall, info	brute force attempt 2 Service_a: in:wlan2 out:(none), src-mac 00:0c:42:64:4c:40, proto TCP (SYN), 192.168.0.254:48099->192.168.0.1:8291, len 60
Feb/21/2016 21:19:40	memory	firewall, info	brute force attempt 1 Service_a: in:wlan2 out:(none), src-mac 00:0c:42:64:4c:40, proto TCP (SYN), 192.168.0.254:48099->192.168.0.1:8291, len 60
Feb/21/2016 21:19:41	memory	system, error, critical	login failure for user admin from 192.168.0.254 via winbox
Feb/21/2016 21:19:41	memory	firewall, info	brute force attempt 3 Service_a: in:wlan2 out:(none), src-mac 00:0c:42:64:4c:40, proto TCP (SYN), 192.168.0.254:48100->192.168.0.1:8291, len 60
Feb/21/2016 21:19:41	memory	firewall, info	brute force attempt 2 Service_a: in:wlan2 out:(none), src-mac 00:0c:42:64:4c:40, proto TCP (SYN), 192.168.0.254:48100->192.168.0.1:8291, len 60
Feb/21/2016 21:19:41	memory	firewall, info	brute force attempt 1 Service_a: in:wlan2 out:(none), src-mac 00:0c:42:64:4c:40, proto TCP (SYN), 192.168.0.254:48100->192.168.0.1:8291, len 60
Feb/21/2016 21:19:42	memory	system, error, critical	login failure for user admin from 192.168.0.254 via winbox
Feb/21/2016 21:19:43	memory	firewall, info	brute force detected Service_at: in:wlan2 out:(none), src-mac 00:0c:42:64:4c:40, proto TCP (SYN), 192.168.0.254:48101->192.168.0.1:8291, len 60
Feb/21/2016 21:19:43	memory	firewall, info	brute force attempt 3 Service_a: in:wlan2 out:(none), src-mac 00:0c:42:64:4c:40, proto TCP (SYN), 192.168.0.254:48101->192.168.0.1:8291, len 60
Feb/21/2016 21:19:43	memory	firewall, info	brute force attempt 2 Service_a: in:wlan2 out:(none), src-mac 00:0c:42:64:4c:40, proto TCP (SYN), 192.168.0.254:48101->192.168.0.1:8291, len 60
Feb/21/2016 21:19:43	memory	firewall, info	brute force attempt 1 Service_a: in:wlan2 out:(none), src-mac 00:0c:42:64:4c:40, proto TCP (SYN), 192.168.0.254:48101->192.168.0.1:8291, len 60
Feb/21/2016 21:19:43	memory	firewall, info	Address in block list input: in:wlan2 out:(none), src-mac 00:0c:42:64:4c:40, proto TCP (ACK), 192.168.0.254:48101->192.168.0.1:8291, len 52

So how do we propagate the bad ?



Blackhole the blacklist!

Route List

Routes | Nexthops | Rules | VRF

+ - ✓ ✗ 📄 🏠

	Dst. Address	Gateway
DAS	▶ 0.0.0.0/0	192.168.74.1 reachable ether2
Do	▶ 0.0.0.0/0	192.168.74.1 reachable ether2
DAC	▶ 10.10.0.1	loopback0 reachable
DAC	▶ 192.168.0.0/24	wlan2 reachable
ASB	▶ 192.168.0.254	
DAo	▶ 192.168.35.0/24	192.168.74.1 reachable ether2
DAC	▶ 192.168.35.0/24	ether1 reachable
DAC	▶ 192.168.74.0/24	ether2 reachable



<demo site-B>

The bit behind the scenes

The screenshot shows the MikroTik Scheduler interface. A table lists a task named 'blackhole' with a start date of Feb/21/2016, starting at startup, with an interval of 00:00:20, on event 'blackhole', owned by 'admin', with a run count of 61 and a next run date of Feb/21/2016... A dialog box titled 'Script <blackhole>' is open, showing the configuration for this task. The 'Name' field is 'blackhole', the 'Owner' is 'admin', and the 'Policy' section has several options checked: ftp, read, policy, password, sensitive, reboot, write, test, and sniff. The 'Last Time Started' is Feb/21/2016 23:19:45 and the 'Run Count' is 61. The 'Source' field contains the following script:

```
#read firewall blacklist and add to route table if not already in
:global check "0";
:foreach entry in=[/ip firewall address-list find where
list=blacklist] do={
:local blackIP [/ip firewall address-list get value-
name=address number=$entry];
:local blackholeIP ($blackIP . "/32");
:foreach route in=[/ip route find where type=blackhole]
do={
:local routeIP [/ip route get value-name=dst-address
```

The process

- Bad traffic is detected at the edge on the input chain
- Src address is added to blacklist address list
- Forward chain then uses the blacklist address to block unwanted traffic
- To then propagate the blacklist information, a script reads the blacklist and creates blackhole routes.

The process

- The route table is then passed on using OSPF in this demonstration to other edge routers
- Script also checks to see if blackhole routes can be removed due to blacklist address timeout (set by the initial firewall input rule).
- A manual whitelist is also created to prevent accidental lockouts to important services and systems.
- It's not perfect... but it works for me!

Available for download

Blackhole script and bare basic routerboard firewall config can be found at:

- <http://www.allness.net/mum>

Do not hold me responsible if it crashes and wipes out your network. Use at your own discretion and risk.

Any questions?

Thank you for your time