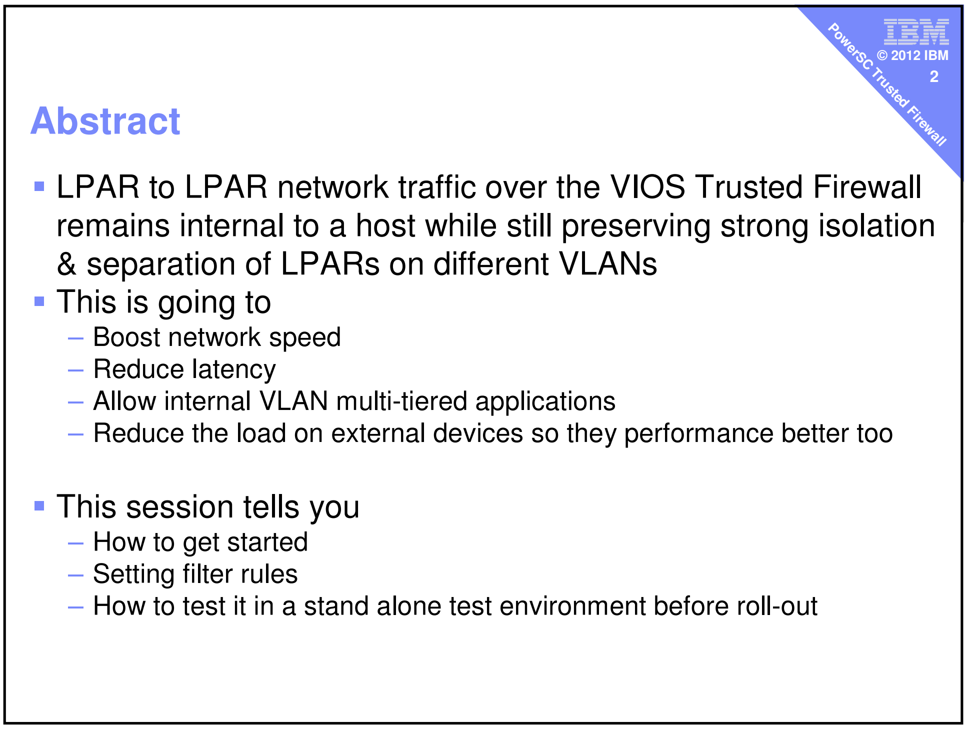
Getting Started PowerSC Trusted Firewall

Release 1.1.1 May 2012

Nigel Griffiths
IBM Power Systems
Advanced Technology Support, Europe

Presentation Version 7

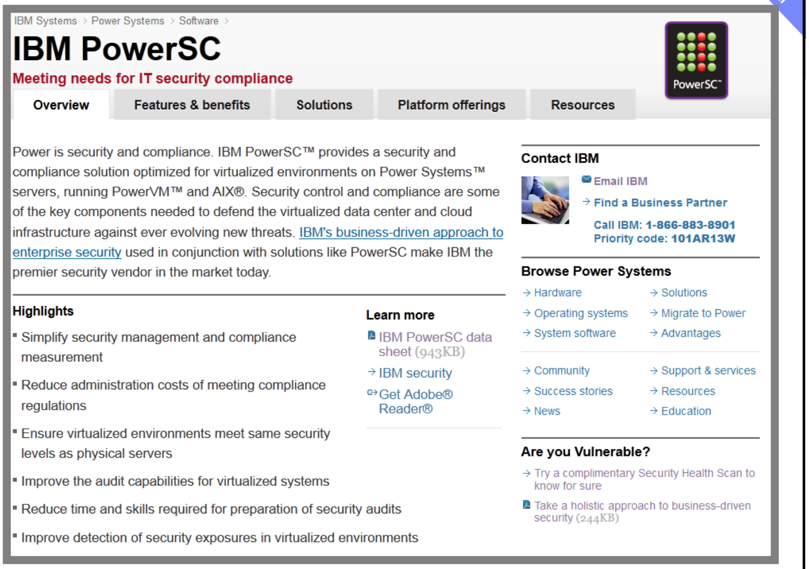
© 2012 IBM Corporation



Abstract

- LPAR to LPAR network traffic over the VIOS Trusted Firewall remains internal to a host while still preserving strong isolation & separation of LPARs on different VLANs
- This is going to
  - Boost network speed
  - Reduce latency
  - Allow internal VLAN multi-tiered applications
  - Reduce the load on external devices so they performance better too
- This session tells you
  - How to get started
  - Setting filter rules
  - How to test it in a stand alone test environment before roll-out

10,000 feet Overview but no “How To” details http://www.ibm.com/systems/power/software/security/



IBM Systems > Power Systems > Software >

IBM PowerSC

Meeting needs for IT security compliance

Power is security and compliance. IBM PowerSC™ provides a security and compliance solution optimized for virtualized environments on Power Systems™ servers, running PowerVM™ and AIX®. Security control and compliance are some of the key components needed to defend the virtualized data center and cloud infrastructure against ever evolving new threats. IBM's business-driven approach to enterprise security used in conjunction with solutions like PowerSC make IBM the premier security vendor in the market today.

Highlights

- Simplify security management and compliance measurement
- Reduce administration costs of meeting compliance regulations
- Ensure virtualized environments meet same security levels as physical servers
- Improve the audit capabilities for virtualized systems
- Reduce time and skills required for preparation of security audits
- Improve detection of security exposures in virtualized environments

Learn more

IBM PowerSC data sheet (943KB)
→ IBM security
Get Adobe® Reader®

Contact IBM

Email IBM
→ Find a Business Partner
Call IBM: 1-866-883-8901
Priority code: 101AR13W

Browse Power Systems

→ Hardware → Solutions
→ Operating systems → Migrate to Power
→ System software → Advantages

→ Community → Support & services
→ Success stories → Resources
→ News → Education

Are you Vulnerable?

→ Try a complimentary Security Health Scan to know for sure
Take a holistic approach to business-driven security (244KB)

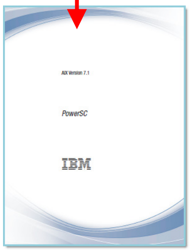
Trusted Firewall Pre-Requisites

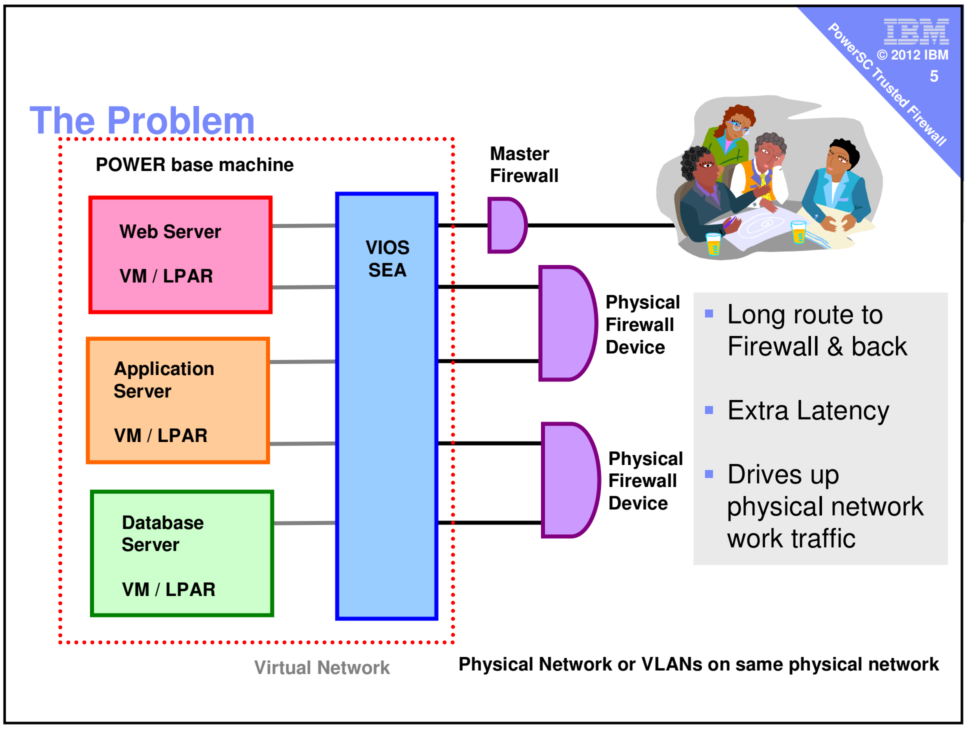
- Virtual I/O Server 2.2.1.4 (manual says 6.1S ops!)
- Any supported OS as it is internal to VIOS

- PowerSC Documentation page 16 -22
- http://pic.dhe.ibm.com/infocenter/aix/v6r1/topic/com.ibm.aix.powersc/powersc_pdf.pdf

- VIOS Documentation page 150 , 202
- http://pic.dhe.ibm.com/infocenter/powersys/v3r1m5/topic/p7hb1/p7hb1.pdf

- PowerSC Wiki
  - New Public website for Info, Hints & Tips
- http://tinyurl.com/PowerSC





The Problem

POWER base machine

Web Server

VM / LPAR

Application Server

VM / LPAR

Database Server

VM / LPAR

VIOS SEA

Master Firewall

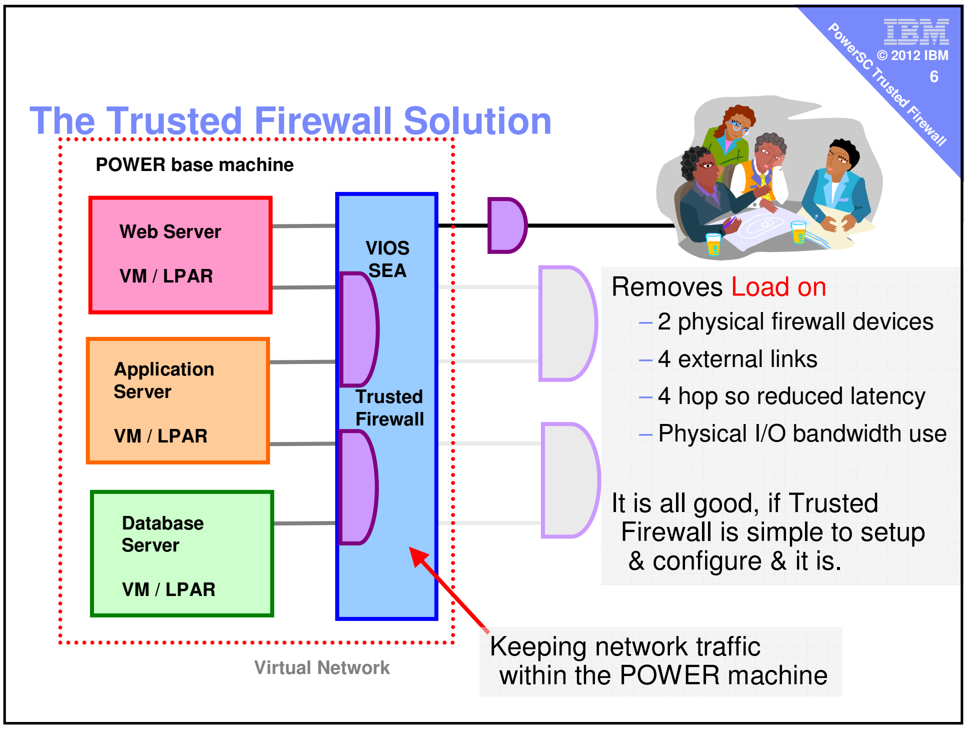
Physical Firewall Device

Physical Firewall Device

- Long route to Firewall & back
- Extra Latency
- Drives up physical network work traffic

Virtual Network

Physical Network or VLANs on same physical network



The Trusted Firewall Solution

POWER base machine

Web Server

VM / LPAR

Application Server

VM / LPAR

Database Server

VM / LPAR

VIOS SEA

Trusted Firewall

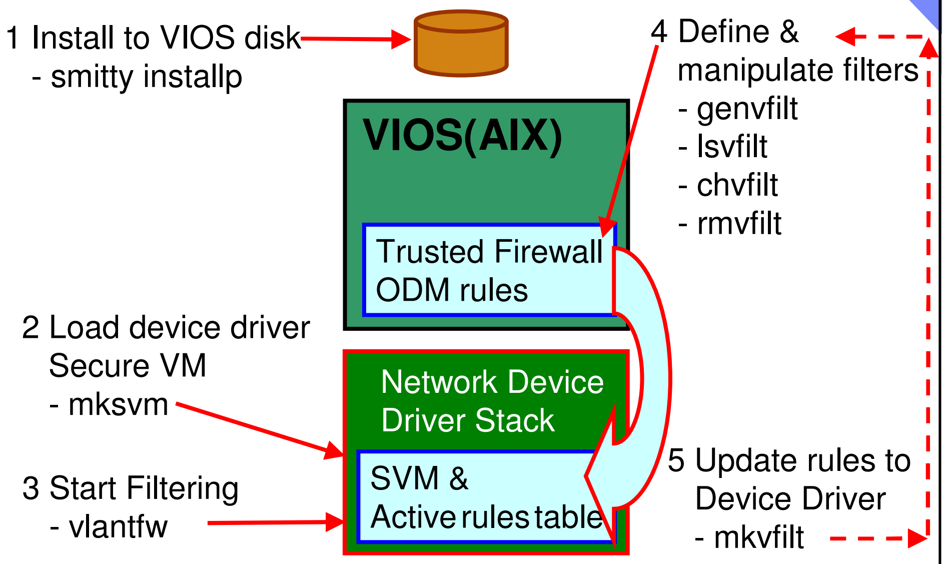
Removes Load on
- 2 physical firewall devices
- 4 external links
- 4 hop so reduced latency
- Physical I/O bandwidth use

It is all good, if Trusted Firewall is simple to setup & configure & it is.

Keeping network traffic within the POWER machine

Virtual Network

Five stages to get the Firewall configured



1 Install to VIOS disk
- smitty installp

2 Load device driver
Secure VM
- mksvm

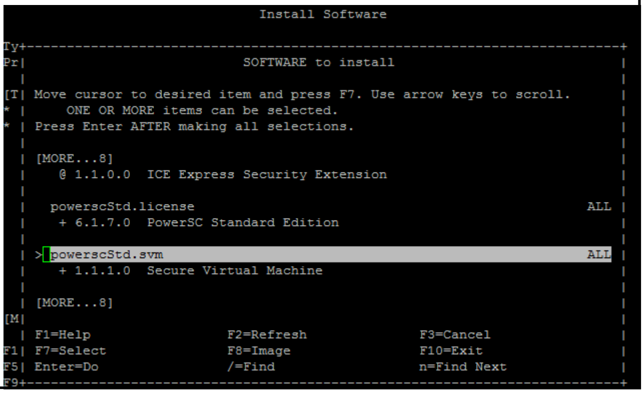
3 Start Filtering
- vlantfw

4 Define & manipulate filters
- genvfilt
- lsvfilt
- chvfilt
- rmvfilt

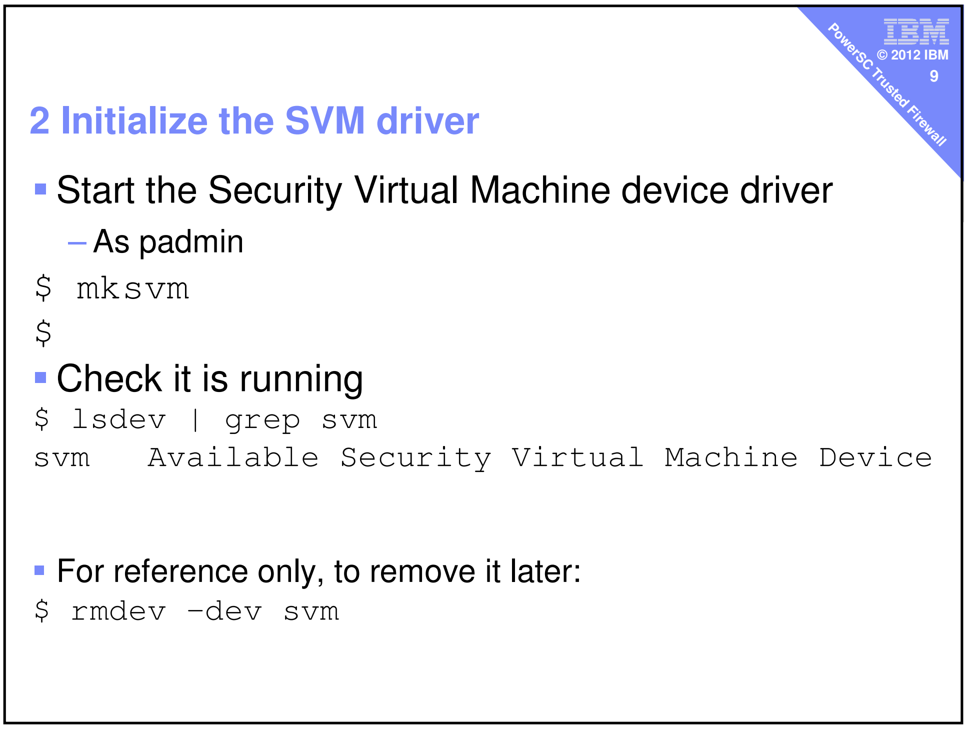
5 Update rules to Device Driver
- mkvfilt

1 On the VIOS as root user (oem_setup_env)

- Everything is configured in the VIOS
  - Supports all POWER OS: IBM i, Linux & AIX
- Install the PowerSC 1.1.1 May 2012 package
  - powerscStd.svm = PowerSC Standard Edition Secure Virtual machine



```
                            Install Software

Ty+--------------------------------------------------------------------------+
Pr|                          SOFTWARE to install                             |
  |                                                                          |
[T| Move cursor to desired item and press F7. Use arrow keys to scroll.      |
* |     ONE OR MORE items can be selected.                                   |
* | Press Enter AFTER making all selections.                                 |
  |                                                                          |
  | [MORE...8]                                                               |
  |    @ 1.1.0.0  ICE Express Security Extension                             |
  |                                                                          |
  |   powerscStd.license                                                 ALL |
  |    + 6.1.7.0  PowerSC Standard Edition                                   |
  |                                                                          |
  | > powerscStd.svm                                                     ALL |
  |    + 1.1.1.0  Secure Virtual Machine                                     |
  |                                                                          |
  | [MORE...8]                                                               |
[M|                                                                          |
  | F1=Help                 F2=Refresh              F3=Cancel                |
F1| F7=Select               F8=Image                F10=Exit                 |
F5| Enter=Do                /=Find                  n=Find Next              |
F9+--------------------------------------------------------------------------+
```



2 Initialize the SVM driver

- Start the Security Virtual Machine device driver
  - As padmin

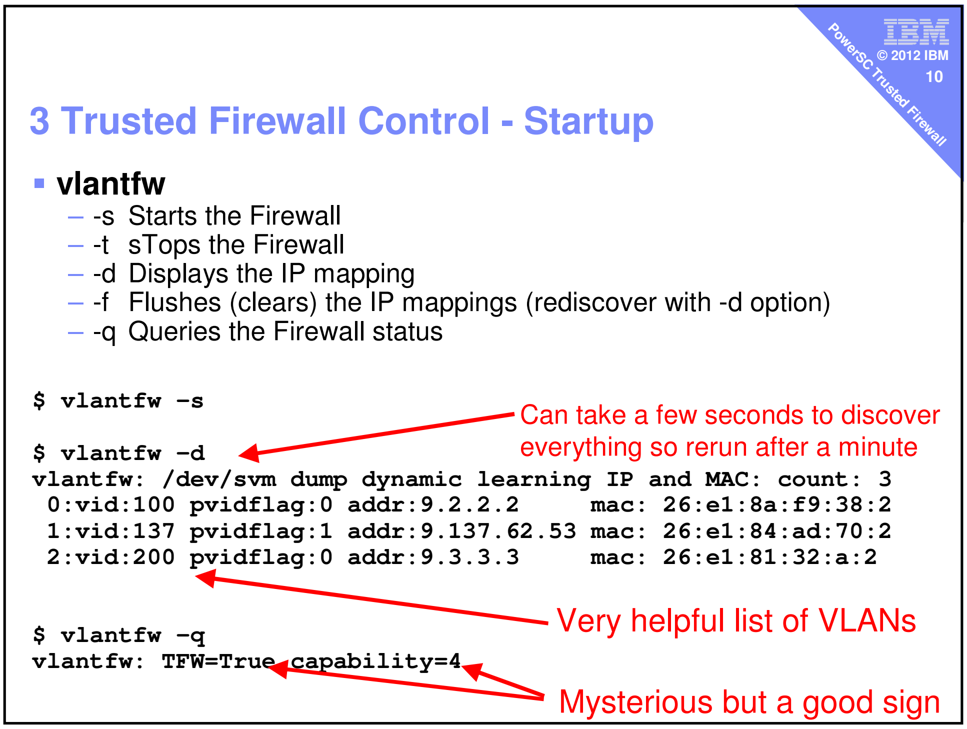
```
$ mksvm
$
```

- Check it is running

```
$ lsdev | grep svm
svm   Available Security Virtual Machine Device
```

- For reference only, to remove it later:

```
$ rmdev -dev svm
```



3 Trusted Firewall Control - Startup

- **vlantfw**
  - -s Starts the Firewall
  - -t sTops the Firewall
  - -d Displays the IP mapping
  - -f Flushes (clears) the IP mappings (rediscover with -d option)
  - -q Queries the Firewall status

```
$ vlantfw -s

$ vlantfw -d
vlantfw: /dev/svm dump dynamic learning IP and MAC: count: 3
 0:vid:100 pvidflag:0 addr:9.2.2.2     mac: 26:e1:8a:f9:38:2
 1:vid:137 pvidflag:1 addr:9.137.62.53 mac: 26:e1:84:ad:70:2
 2:vid:200 pvidflag:0 addr:9.3.3.3     mac: 26:e1:81:32:a:2


$ vlantfw -q
vlantfw: TFW=True capability=4
```

Can take a few seconds to discover everything so rerun after a minute

Very helpful list of VLANs

Mysterious but a good sign

4 and 5
The rest of Trusted Firewall is all down to the filter rules

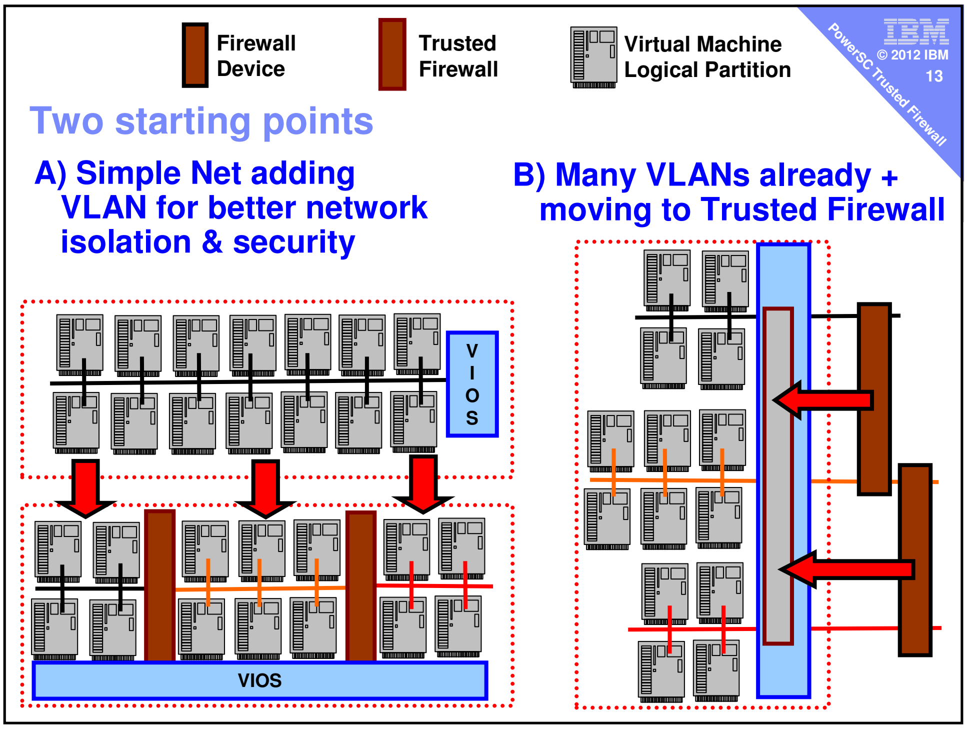
Firewalls for Non-network Guru's

General Firewall

- A Firewall is like a network bridge/router with multiple networks (or VLANs) except nothing is allowed between networks unless a filter rule says it's OK

VIOS Trusted Firewall

- By default the VIOS level Trusted Firewall will not internally route/bridge the packet unless it has a rule that permits it
- If no rule, the packet is placed on the external network and higher level external router/gateway/firewall to forward or drop packets
- It is common practice to have multiple layers of firewalls

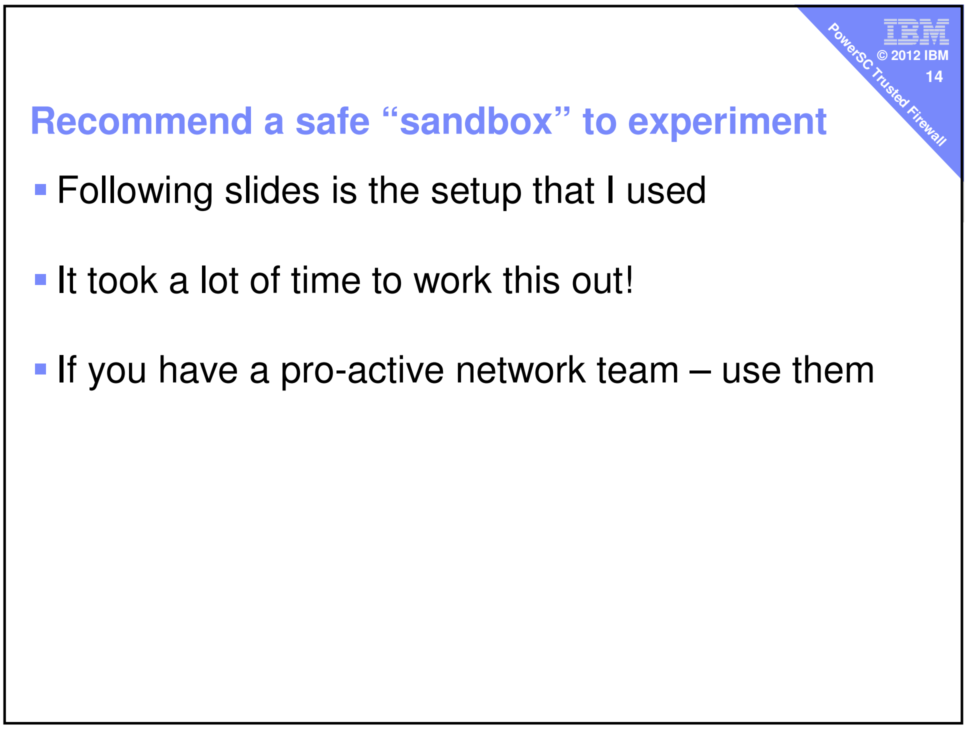


▮ Firewall Device
▮ Trusted Firewall
Virtual Machine Logical Partition

Two starting points

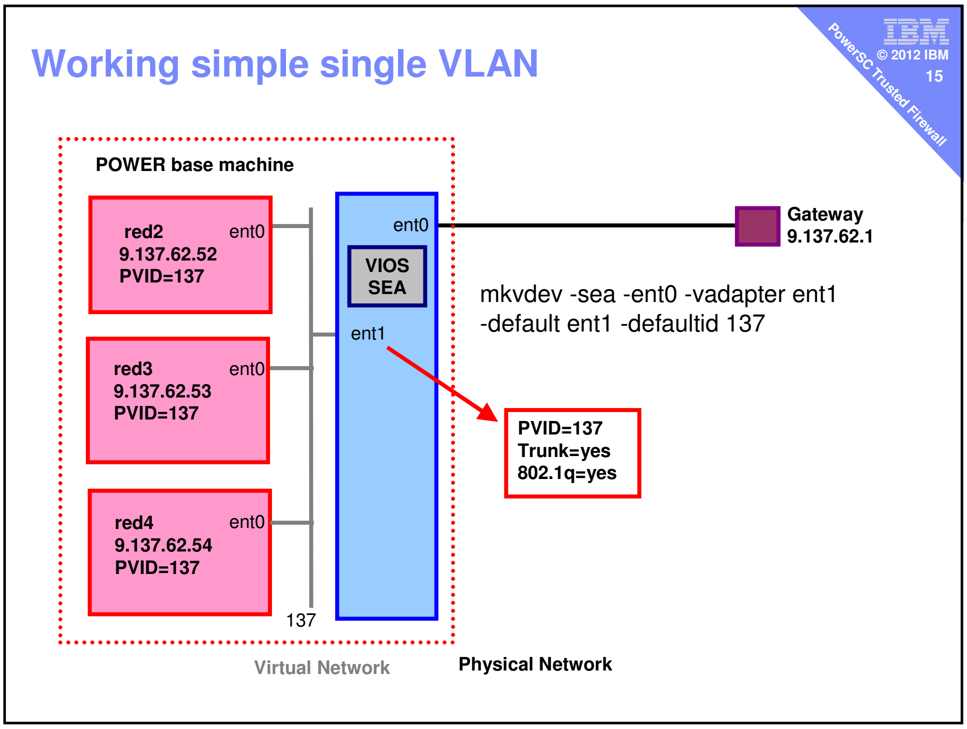
A) Simple Net adding VLAN for better network isolation & security

B) Many VLANs already + moving to Trusted Firewall



Recommend a safe “sandbox” to experiment

- Following slides is the setup that I used
- It took a lot of time to work this out!
- If you have a pro-active network team – use them



Working simple single VLAN

POWER base machine

red2
9.137.62.52
PVID=137
ent0

red3
9.137.62.53
PVID=137
ent0

red4
9.137.62.54
PVID=137
ent0

137

ent0
VIOS
SEA
ent1

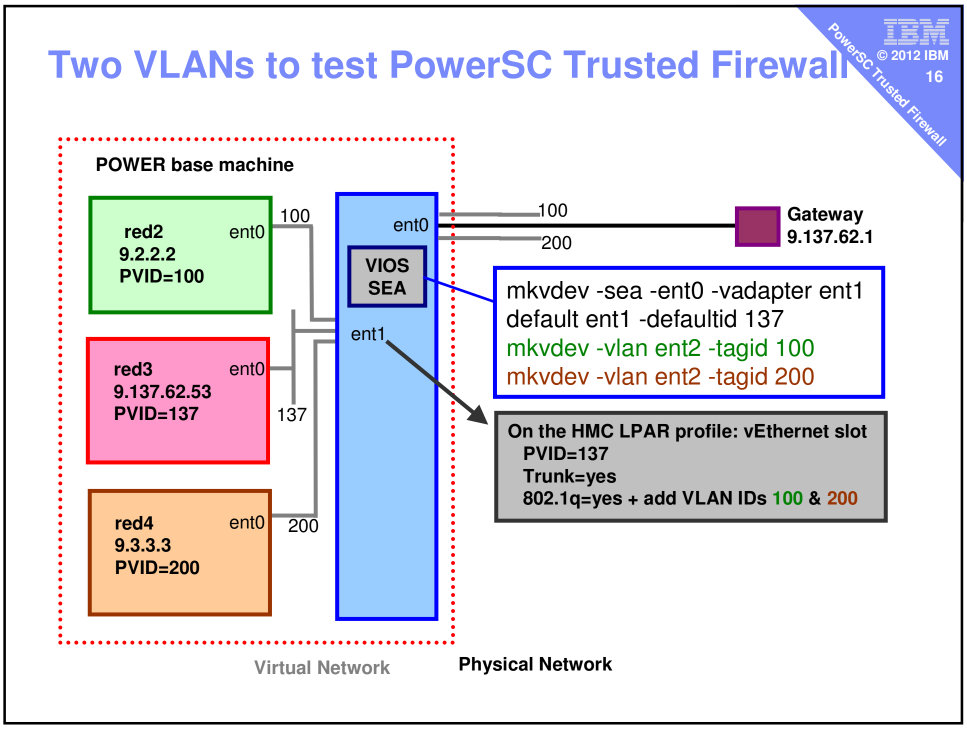
Gateway
9.137.62.1

mkdev -sea -ent0 -vadapter ent1
-default ent1 -defaultid 137

PVID=137
Trunk=yes
802.1q=yes

Virtual Network

Physical Network



Two VLANs to test PowerSC Trusted Firewall

POWER base machine

red2
9.2.2.2
PVID=100
ent0

100

red3
9.137.62.53
PVID=137
ent0

137

red4
9.3.3.3
PVID=200
ent0

200

ent0
VIOS
SEA
ent1

100

200

Gateway
9.137.62.1

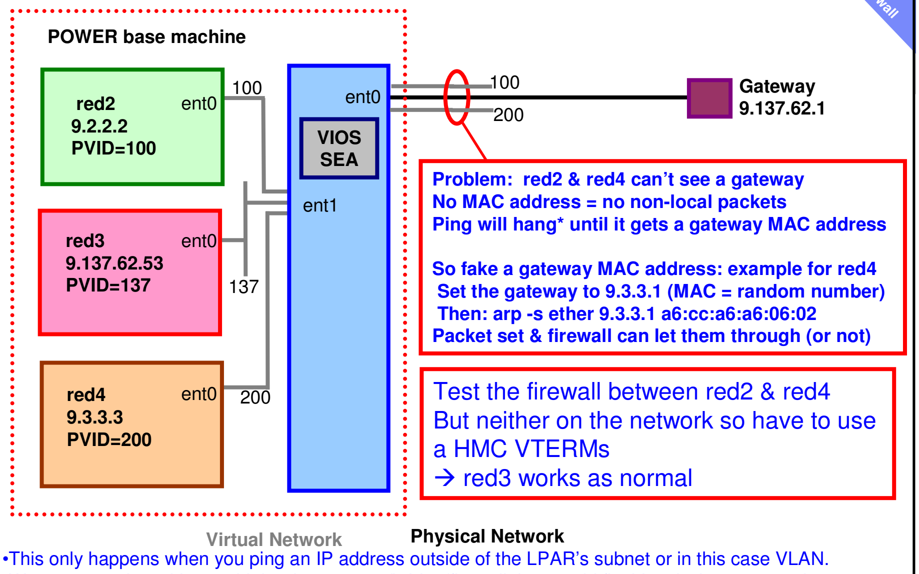
mkdev -sea -ent0 -vadapter ent1
default ent1 -defaultid 137
mkdev -vlan ent2 -tagid 100
mkdev -vlan ent2 -tagid 200

On the HMC LPAR profile: vEthernet slot
PVID=137
Trunk=yes
802.1q=yes + add VLAN IDs 100 & 200

Virtual Network

Physical Network

Two VLANs to test PowerSC Trusted Firewall



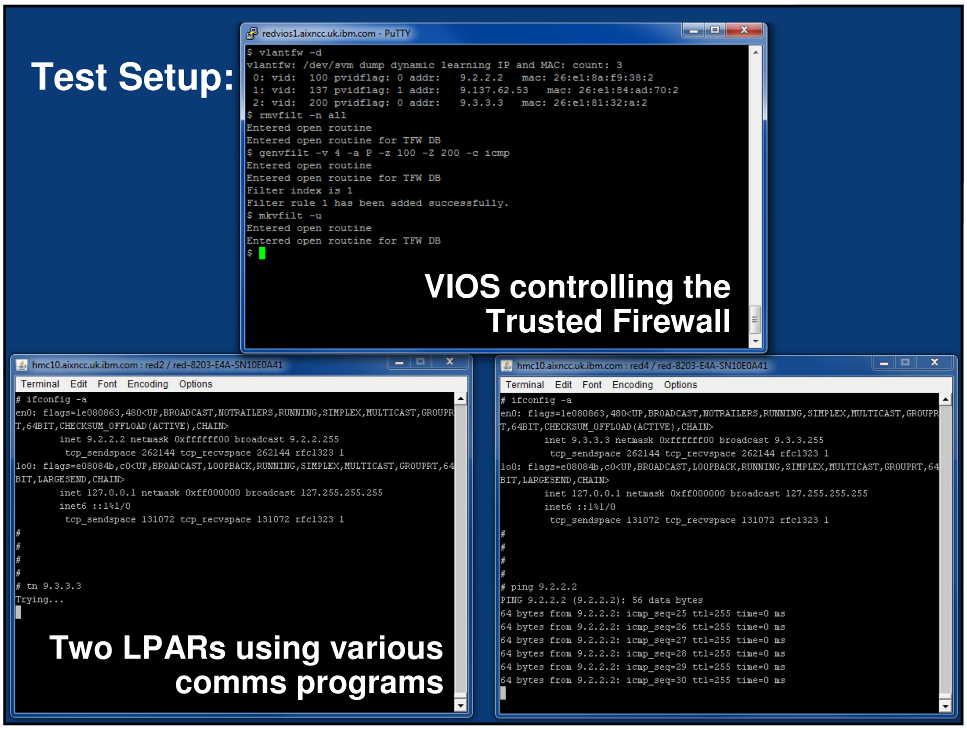
Problem: red2 & red4 can't see a gateway
No MAC address = no non-local packets
Ping will hang* until it gets a gateway MAC address

So fake a gateway MAC address: example for red4
Set the gateway to 9.3.3.1 (MAC = random number)
Then: arp -s ether 9.3.3.1 a6:cc:a6:a6:06:02
Packet set & firewall can let them through (or not)

Test the firewall between red2 & red4
But neither on the network so have to use a HMC VTERMs
→ red3 works as normal

- •This only happens when you ping an IP address outside of the LPAR's subnet or in this case VLAN.
- •We are operating without a higher layer router/bridge/firewall that would fix this missing gateway issue.

Test Setup:


redvios1.aixncc.uk.ibm.com - PuTTY
hmc10.aixncc.uk.ibm.com : red2 / red-8203-E4A-SN10E0A41
hmc10.aixncc.uk.ibm.com : red4 / red-8203-E4A-SN10E0A41
Terminal
Edit
Font
Encoding
Options

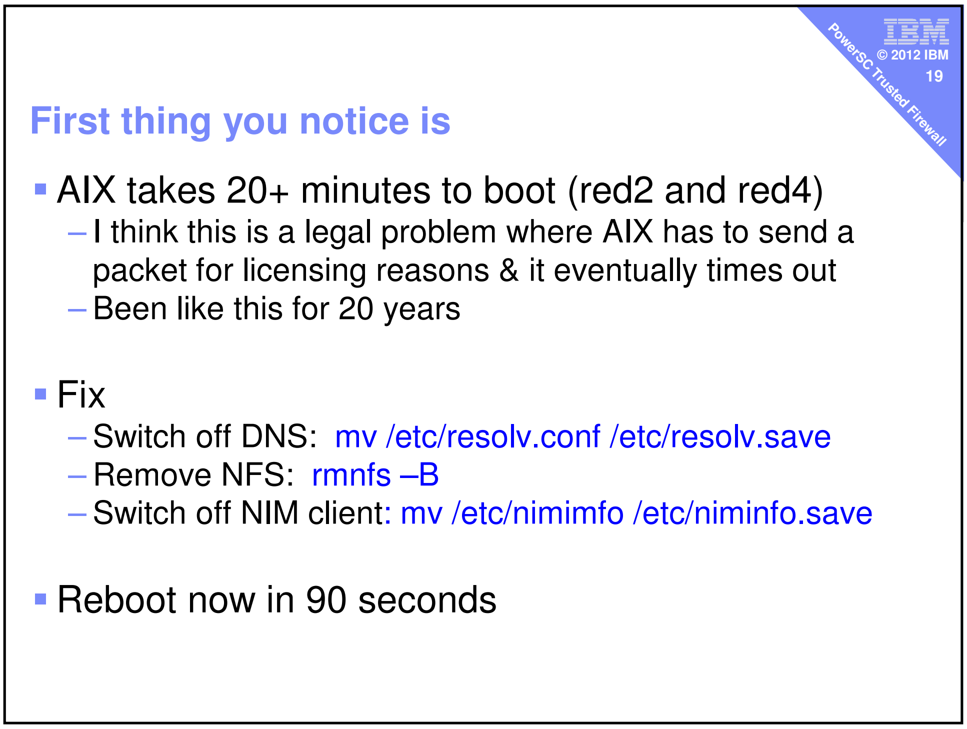
```
$ vlantfw -d
vlantfw: /dev/svm dump dynamic learning IP and MAC: count: 3
 0: vid:  100 pvidflag: 0 addr:   9.2.2.2   mac: 26:e1:8a:f9:38:2
 1: vid:  137 pvidflag: 1 addr:   9.137.62.53   mac: 26:e1:84:ad:70:2
 2: vid:  200 pvidflag: 0 addr:   9.3.3.3   mac: 26:e1:81:32:a:2
$ rmvfilt -n all
Entered open routine
Entered open routine for TFW DB
$ genvfilt -v 4 -a P -z 100 -Z 200 -c icmp
Entered open routine
Entered open routine for TFW DB
Filter index is 1
Filter rule 1 has been added successfully.
$ mkvfilt -u
Entered open routine
Entered open routine for TFW DB
$
```

VIOS controlling the Trusted Firewall

```
# ifconfig -a
en0: flags=1e080863,480<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GROUPR
T,64BIT,CHECKSUM_OFFLOAD(ACTIVE),CHAIN>
        inet 9.2.2.2 netmask 0xffffff00 broadcast 9.2.2.255
         tcp_sendspace 262144 tcp_recvspace 262144 rfc1323 1
lo0: flags=e08084b,c0<UP,BROADCAST,LOOPBACK,RUNNING,SIMPLEX,MULTICAST,GROUPRT,64
BIT,LARGESEND,CHAIN>
        inet 127.0.0.1 netmask 0xff000000 broadcast 127.255.255.255
        inet6 ::1%1/0
         tcp_sendspace 131072 tcp_recvspace 131072 rfc1323 1
#
#
#
#
# tn 9.3.3.3
Trying...
```

Two LPARs using various comms programs

```
# ifconfig -a
en0: flags=1e080863,480<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GROUPR
T,64BIT,CHECKSUM_OFFLOAD(ACTIVE),CHAIN>
        inet 9.3.3.3 netmask 0xffffff00 broadcast 9.3.3.255
         tcp_sendspace 262144 tcp_recvspace 262144 rfc1323 1
lo0: flags=e08084b,c0<UP,BROADCAST,LOOPBACK,RUNNING,SIMPLEX,MULTICAST,GROUPRT,64
BIT,LARGESEND,CHAIN>
        inet 127.0.0.1 netmask 0xff000000 broadcast 127.255.255.255
        inet6 ::1%1/0
         tcp_sendspace 131072 tcp_recvspace 131072 rfc1323 1
#
#
#
#
# ping 9.2.2.2
PING 9.2.2.2 (9.2.2.2): 56 data bytes
64 bytes from 9.2.2.2: icmp_seq=25 ttl=255 time=0 ms
64 bytes from 9.2.2.2: icmp_seq=26 ttl=255 time=0 ms
64 bytes from 9.2.2.2: icmp_seq=27 ttl=255 time=0 ms
64 bytes from 9.2.2.2: icmp_seq=28 ttl=255 time=0 ms
64 bytes from 9.2.2.2: icmp_seq=29 ttl=255 time=0 ms
64 bytes from 9.2.2.2: icmp_seq=30 ttl=255 time=0 ms
```

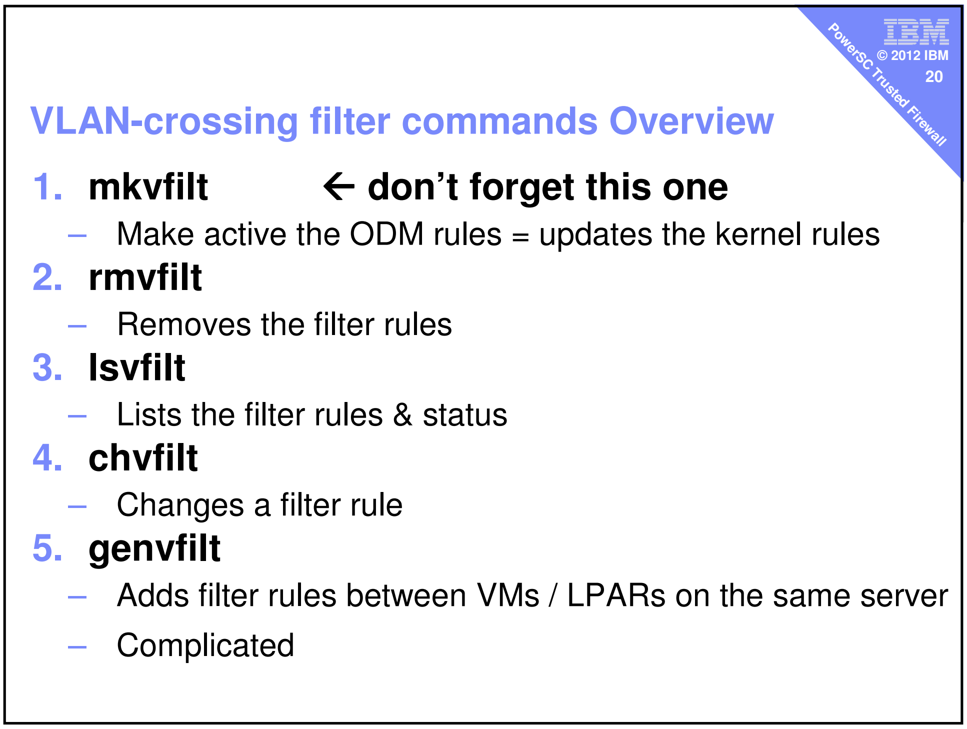


First thing you notice is

- AIX takes 20+ minutes to boot (red2 and red4)
  - I think this is a legal problem where AIX has to send a packet for licensing reasons & it eventually times out
  - Been like this for 20 years

- Fix
  - Switch off DNS: mv /etc/resolv.conf /etc/resolv.save
  - Remove NFS: rmnfs –B
  - Switch off NIM client: mv /etc/nimimfo /etc/niminfo.save

- Reboot now in 90 seconds

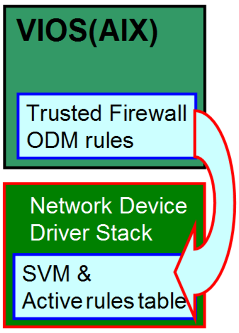


VLAN-crossing filter commands Overview

1. **mkvfilt** ← **don't forget this one**
   – Make active the ODM rules = updates the kernel rules
2. **rmvfilt**
   – Removes the filter rules
3. **lsvfilt**
   – Lists the filter rules & status
4. **chvfilt**
   – Changes a filter rule
5. **genvfilt**
   – Adds filter rules between VMs / LPARs on the same server
   – Complicated

1 mkvfilt syntax – make active filters

- mkvfilt -u
  - Only one mandatory (IMHO pointless) option
  - Activates the current ODM VLAN-crossing filter rules
  - i.e. updates the Kernel rules table from the ODM!
- Very easy to forget to do this after making filter changes



VIOS(AIX)

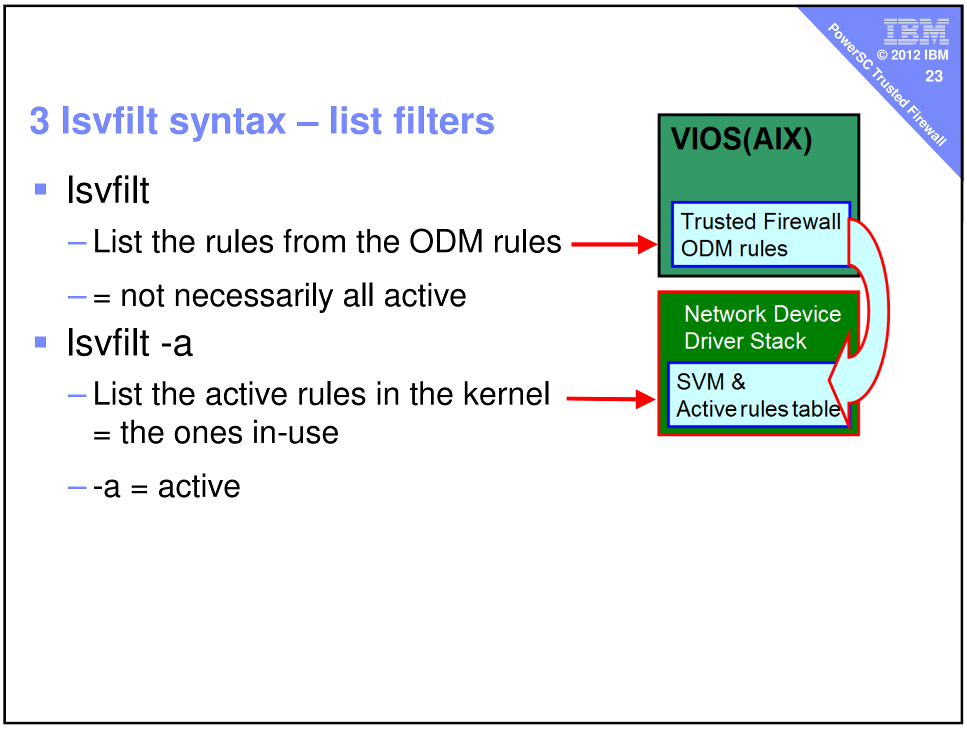
Trusted Firewall ODM rules

Network Device Driver Stack

SVM & Active rules table

2 rmvfilt syntax – remove filter rules

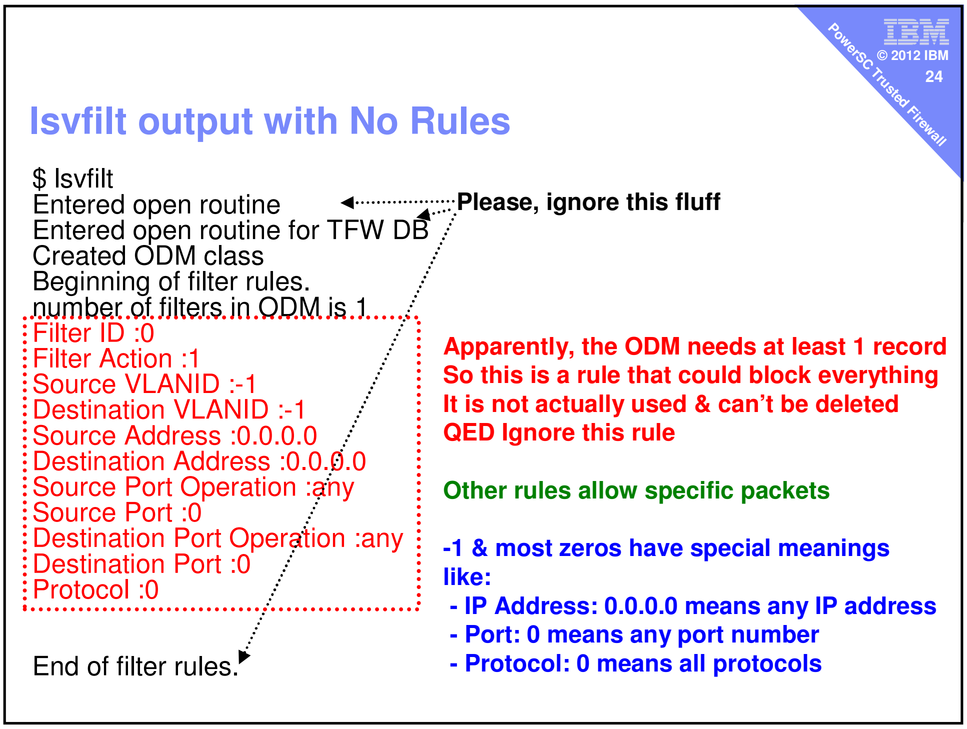
- rmvfilt -n all
  - Removes all ODM the filter rules
  - It appears to delete these from the kernel too
  - → Undocumented but useful
- rmvfilt -n rule-number
  - DOES NOT WORK !!
  - So use a script to add the rules as you have to remove all and reapply the rules you still want!



3 lsvfilt syntax – list filters

- lsvfilt
  - List the rules from the ODM rules
  - = not necessarily all active
- lsvfilt -a
  - List the active rules in the kernel = the ones in-use
  - -a = active

VIOS(AIX)
Trusted Firewall ODM rules
Network Device Driver Stack
SVM & Active rules table



lsvfilt output with No Rules

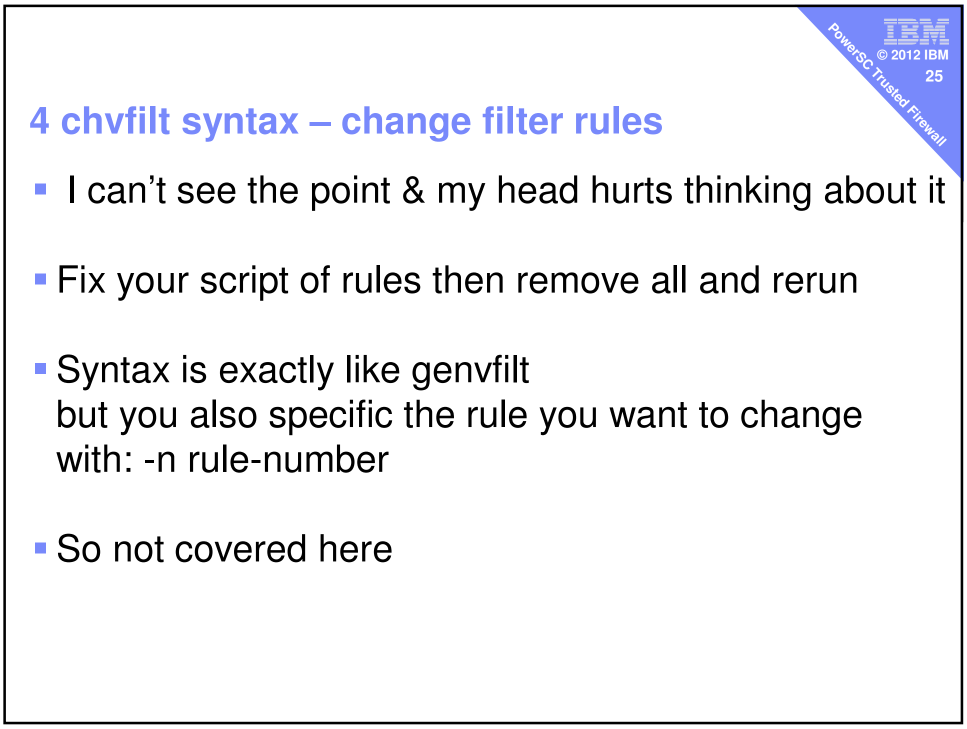
```
$ lsvfilt
Entered open routine
Entered open routine for TFW DB
Created ODM class
Beginning of filter rules.
number of filters in ODM is 1
Filter ID :0
Filter Action :1
Source VLANID :-1
Destination VLANID :-1
Source Address :0.0.0.0
Destination Address :0.0.0.0
Source Port Operation :any
Source Port :0
Destination Port Operation :any
Destination Port :0
Protocol :0

End of filter rules.
```

Please, ignore this fluff

Apparently, the ODM needs at least 1 record
So this is a rule that could block everything
It is not actually used & can't be deleted
QED Ignore this rule

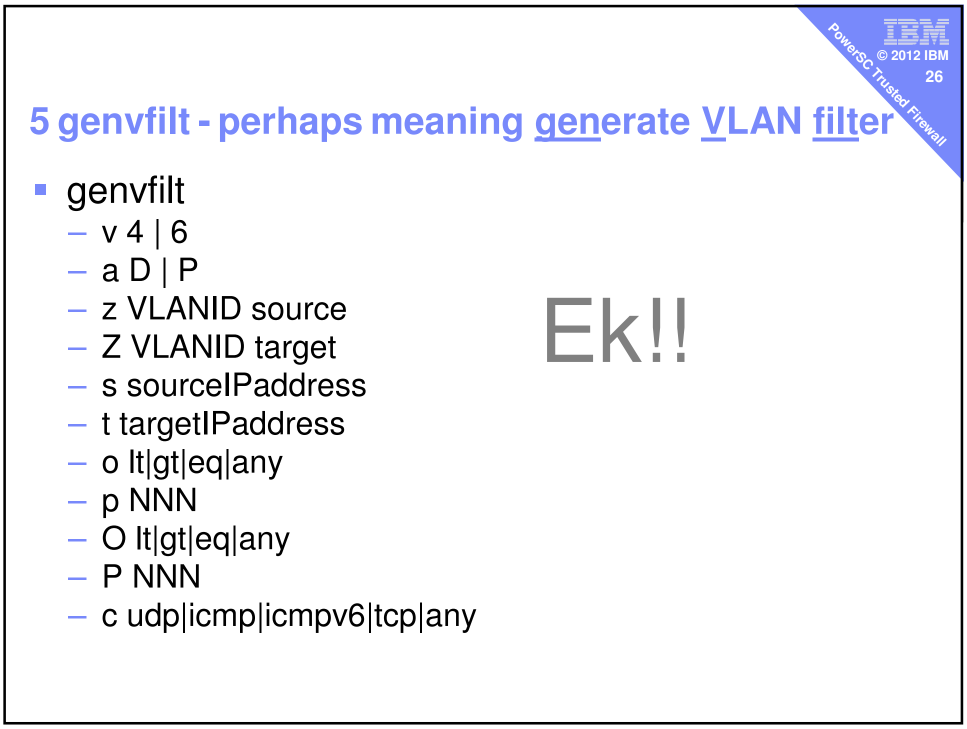
Other rules allow specific packets

-1 & most zeros have special meanings like:
- **IP Address: 0.0.0.0 means any IP address**
- **Port: 0 means any port number**
- **Protocol: 0 means all protocols**



4 chvfilt syntax – change filter rules

- I can’t see the point & my head hurts thinking about it
- Fix your script of rules then remove all and rerun
- Syntax is exactly like genvfilt
  but you also specific the rule you want to change
  with: -n rule-number
- So not covered here



5 genvfilt - perhaps meaning generate VLAN filter

- genvfilt
  - v 4 | 6
  - a D | P
  - z VLANID source
  - Z VLANID target
  - s sourceIPaddress
  - t targetIPaddress
  - o lt|gt|eq|any
  - p NNN
  - O lt|gt|eq|any
  - P NNN
  - c udp|icmp|icmpv6|tcp|any

Ek!!

genvfilt - perhaps meaning generate filter!

- genvfilt
  - v 4 | 6 → IP version 4 or IP version 6
  - a D | P → Deny (block) or Permit traffic
  - z VLANID source
  - Z VLANID target
  - s source-IP-address
  - d destination-IP-address
  - o lt|gt|eq|any
  - p NNN →source port number
  - O lt|gt|eq|any
  - P NNN →target port number
  - c udp|icmp|icmpv6|tcp|any

These four options are mandatory

Limit to specific IP addresses

Limit to specific ports=services or ranges

Limit the protocol used

- AIX Standard Port Number are in /etc/services

Filter Rules by example

Anything between the two VLANs

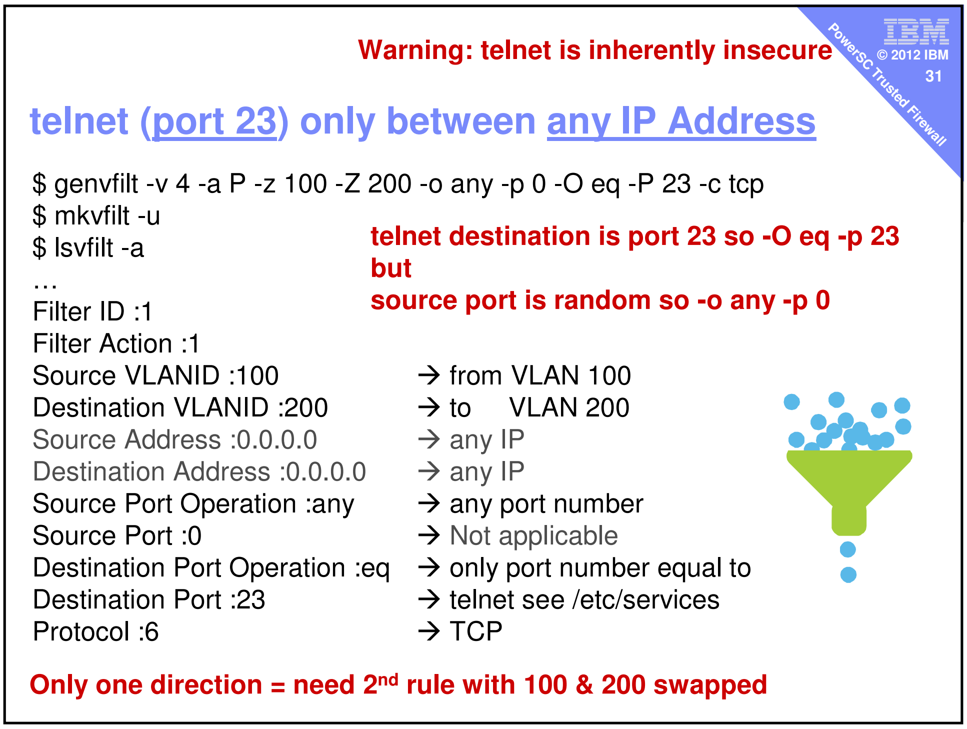
```
$ genvfilt -v 4 -a P -z 100 -Z 200
$ mkvfilt -u
$ lsvfilt -a
…
Filter ID :1
Filter Action :1
Source VLANID :100                  → from VLAN 100
Destination VLANID :200             → to     VLAN 200
Source Address :0.0.0.0             → any IP
Destination Address :0.0.0.0        → any IP
Source Port Operation :any          → any port
Source Port :0                      → Not applicable
Destination Port Operation :any     → any port
Destination Port :0                 → Not applicable
Protocol :0                         → all
```

Also allowed in the 200 to 100 direction as no other options

Anything between the two IP Addresses ONLY

```
$ genvfilt -v 4 -a P -z 100 -Z 200 -s 9.2.2.2 -d 9.3.3.3
$ mkvfilt -u
$ lsvfilt -a
…
Filter ID :1
Filter Action :1
Source VLANID :100                  → from VLAN 100
Destination VLANID :200             → to     VLAN 200
Source Address :9.2.2.2             → red2
Destination Address :9.3.3.3        → red4
Source Port Operation :any          → any port
Source Port :0                      → Not applicable
Destination Port Operation :any     → any port
Destination Port :0                 → Not applicable
Protocol :0                         → all
```

Also allowed in the 200 to 100 direction as no other options



PowerSC Trusted Firewall

IBM © 2012 IBM 31

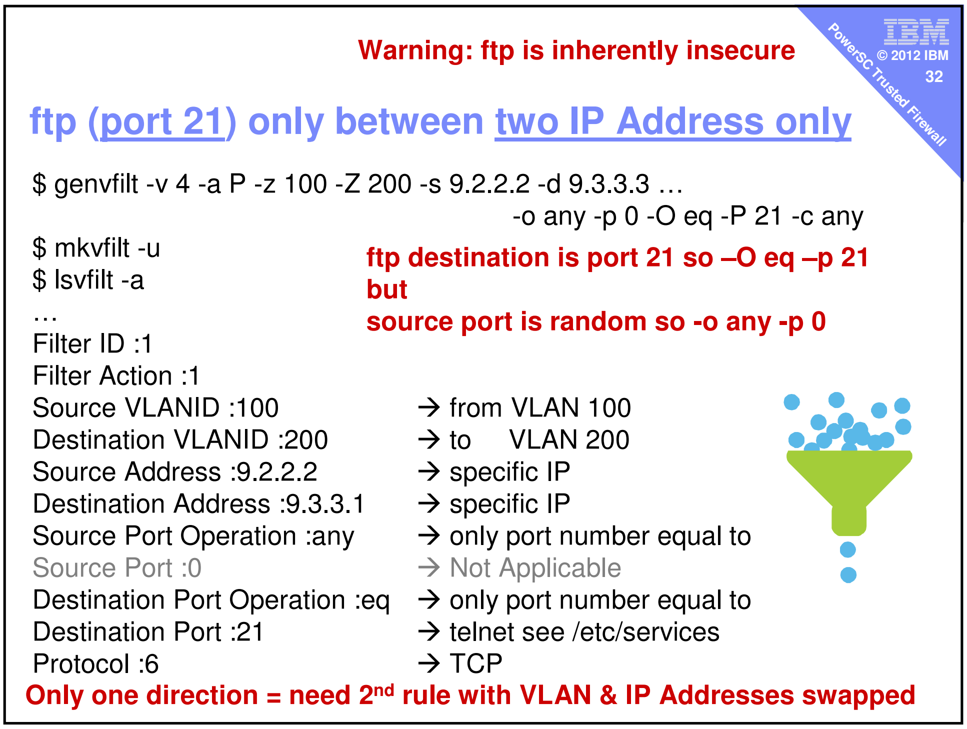
Warning: telnet is inherently insecure

telnet (port 23) only between any IP Address

```
$ genvfilt -v 4 -a P -z 100 -Z 200 -o any -p 0 -O eq -P 23 -c tcp
$ mkvfilt -u
$ lsvfilt -a
…
Filter ID :1
Filter Action :1
Source VLANID :100              → from VLAN 100
Destination VLANID :200         → to     VLAN 200
Source Address :0.0.0.0         → any IP
Destination Address :0.0.0.0    → any IP
Source Port Operation :any      → any port number
Source Port :0                  → Not applicable
Destination Port Operation :eq  → only port number equal to
Destination Port :23            → telnet see /etc/services
Protocol :6                     → TCP
```

telnet destination is port 23 so -O eq -p 23
but
source port is random so -o any -p 0

Only one direction = need 2nd rule with 100 & 200 swapped



PowerSC Trusted Firewall

IBM © 2012 IBM 32

Warning: ftp is inherently insecure

ftp (port 21) only between two IP Address only

```
$ genvfilt -v 4 -a P -z 100 -Z 200 -s 9.2.2.2 -d 9.3.3.3 …
                                       -o any -p 0 -O eq -P 21 -c any
$ mkvfilt -u
$ lsvfilt -a
…
Filter ID :1
Filter Action :1
Source VLANID :100              → from VLAN 100
Destination VLANID :200         → to     VLAN 200
Source Address :9.2.2.2         → specific IP
Destination Address :9.3.3.1    → specific IP
Source Port Operation :any      → only port number equal to
Source Port :0                  → Not Applicable
Destination Port Operation :eq  → only port number equal to
Destination Port :21            → telnet see /etc/services
Protocol :6                     → TCP
```

ftp destination is port 21 so –O eq –p 21
but
source port is random so -o any -p 0

Only one direction = need 2nd rule with VLAN & IP Addresses swapped

ftp both directions two IP Address only

To ftp red 2 to red 4

```
$ genvfilt -v 4 -a P -z 100 -Z 200 -s 9.2.2.2 -d 9.3.3.3 …
                                  -o any -p 0 -O eq -P 21 -c any
```

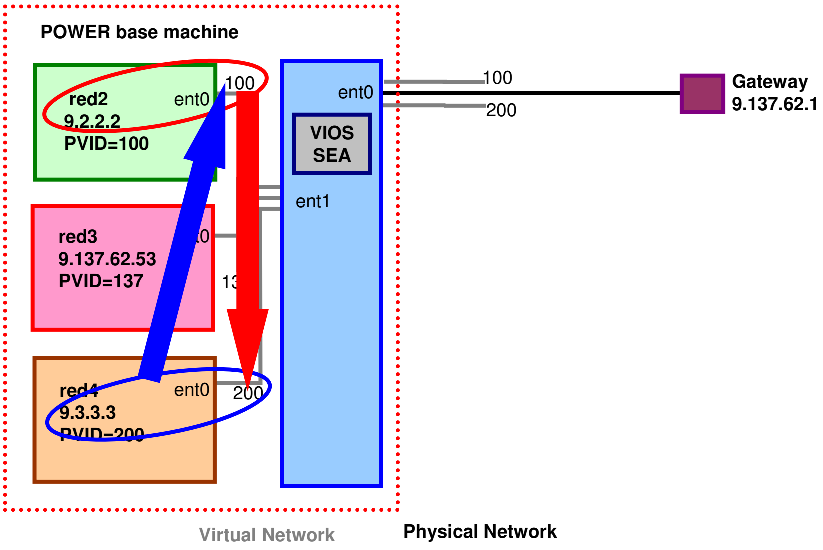
From VLAN=100 & IP=9.2.2.2 to VLAN=200 & IP 9.3.3.3

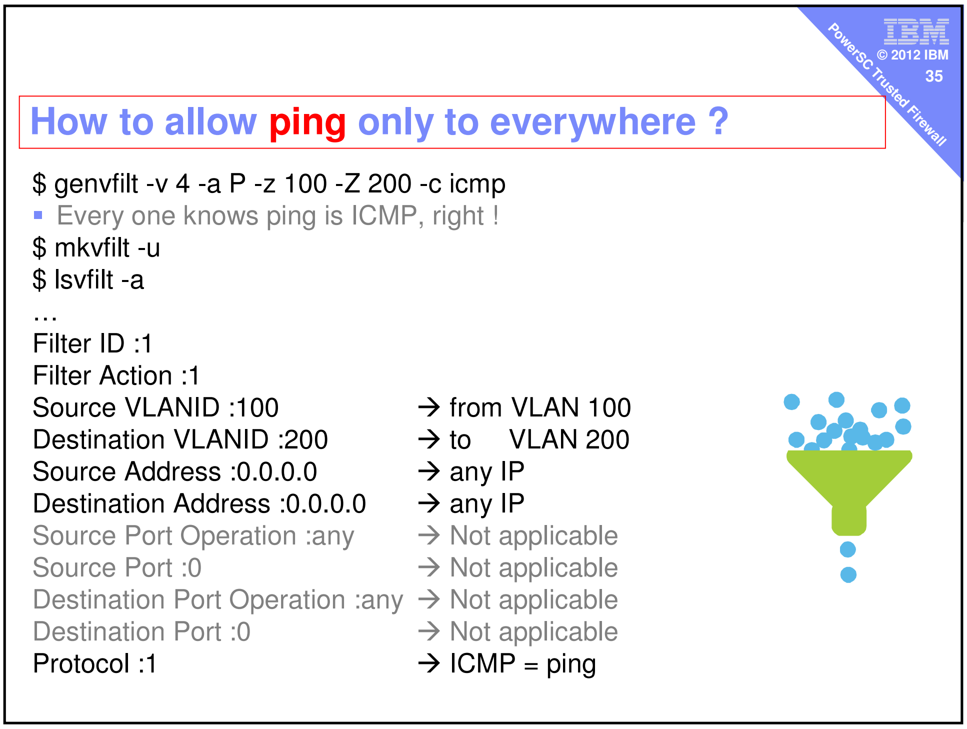
To ftp red4 to red2

```
$ genvfilt -v 4 -a P -z 200 -Z 100 -s 9.3.3.3 -d 9.2.2.2 …
                                  -o any -p 0 -O eq -P 21 -c any
```

From VLAN=200 & IP=9.3.3.3 to VLAN=100 & IP 9.2.2.2

Two VLANs to test PowerSC Trusted Firewall





PowerSC Trusted Firewall

IBM

© 2012 IBM

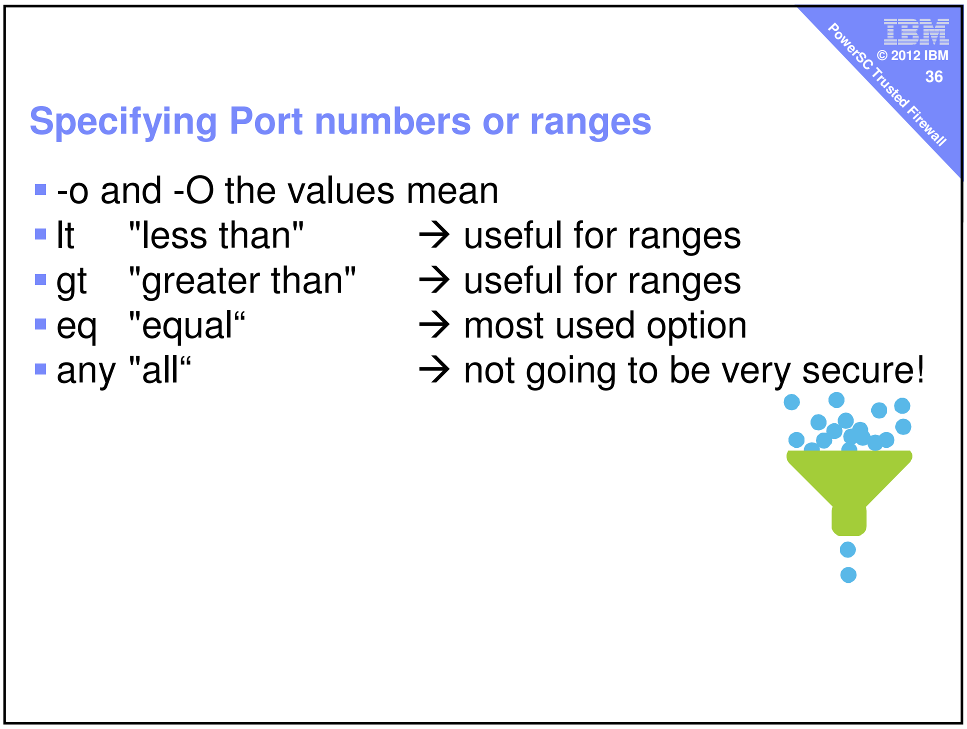
35

How to allow ping only to everywhere ?

```
$ genvfilt -v 4 -a P -z 100 -Z 200 -c icmp
```

- Every one knows ping is ICMP, right !

```
$ mkvfilt -u
$ lsvfilt -a
…
Filter ID :1
Filter Action :1
Source VLANID :100                  → from VLAN 100
Destination VLANID :200             → to     VLAN 200
Source Address :0.0.0.0             → any IP
Destination Address :0.0.0.0        → any IP
Source Port Operation :any          → Not applicable
Source Port :0                      → Not applicable
Destination Port Operation :any     → Not applicable
Destination Port :0                 → Not applicable
Protocol :1                         → ICMP = ping
```



PowerSC Trusted Firewall

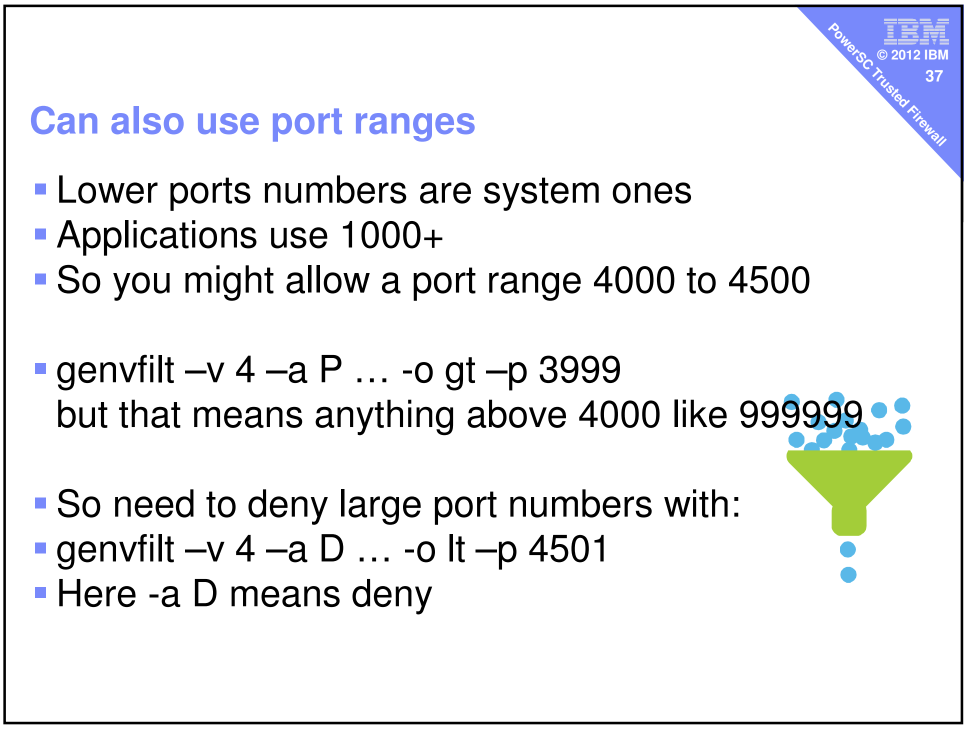
IBM

© 2012 IBM

36

Specifying Port numbers or ranges

- -o and -O the values mean
- lt "less than" → useful for ranges
- gt "greater than" → useful for ranges
- eq "equal“ → most used option
- any "all“ → not going to be very secure!



IBM

© 2012 IBM

37

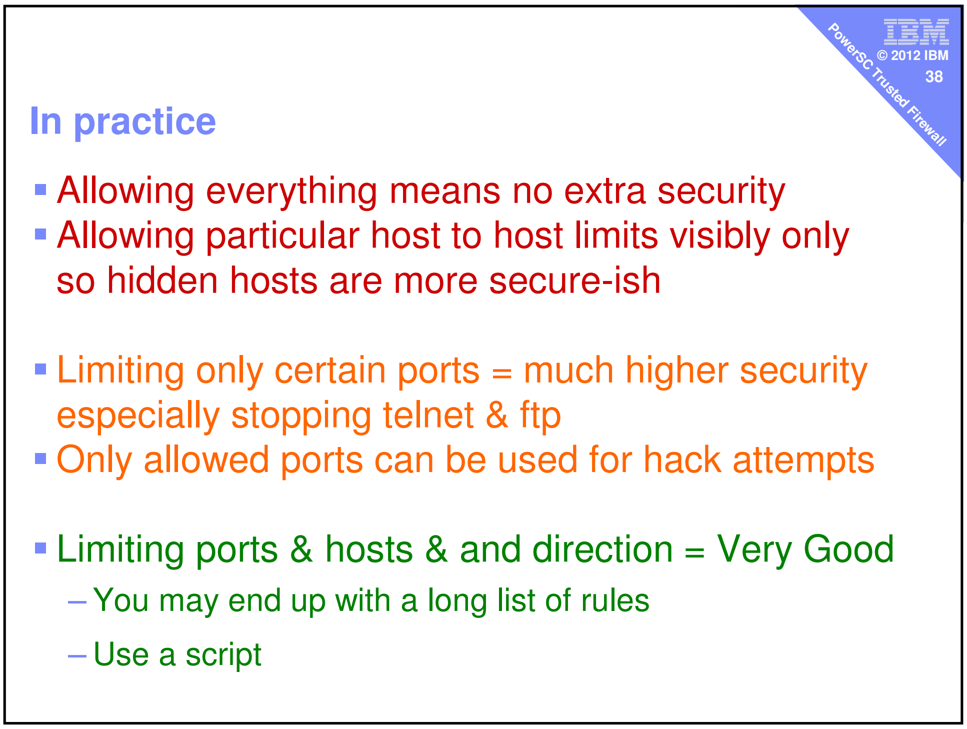
PowerSC Trusted Firewall

Can also use port ranges

- Lower ports numbers are system ones
- Applications use 1000+
- So you might allow a port range 4000 to 4500

- genvfilt –v 4 –a P … -o gt –p 3999
  but that means anything above 4000 like 999999

- So need to deny large port numbers with:
- genvfilt –v 4 –a D … -o lt –p 4501
- Here -a D means deny



IBM

© 2012 IBM

38

PowerSC Trusted Firewall

In practice

- Allowing everything means no extra security
- Allowing particular host to host limits visibly only
  so hidden hosts are more secure-ish

- Limiting only certain ports = much higher security
  especially stopping telnet & ftp
- Only allowed ports can be used for hack attempts

- Limiting ports & hosts & and direction = Very Good
  - You may end up with a long list of rules
  - Use a script

lsvfilt output for different genvfilt -c options

- Protocol
  - 0 = any
  - 1 = icmp
  - 6 = tcp
  - 17 = udp
  - ? = icmpv6 [I don’t use IPv6 so I can’t find out]

From the genvfilt manual page:
genvfilt -v4 -a P -z 100 -Z 200 -o lt -p 345 -O lt -P 345 -c tcp

- IP version 4
- Permanent
- Source VLAN 100
- Target VLAN 200
- Source ports less than 345
- Target ports less than 345
- TCP protocol

- /etc/services contains the port numbers
  the less than 345 port was chosen totally randomly!

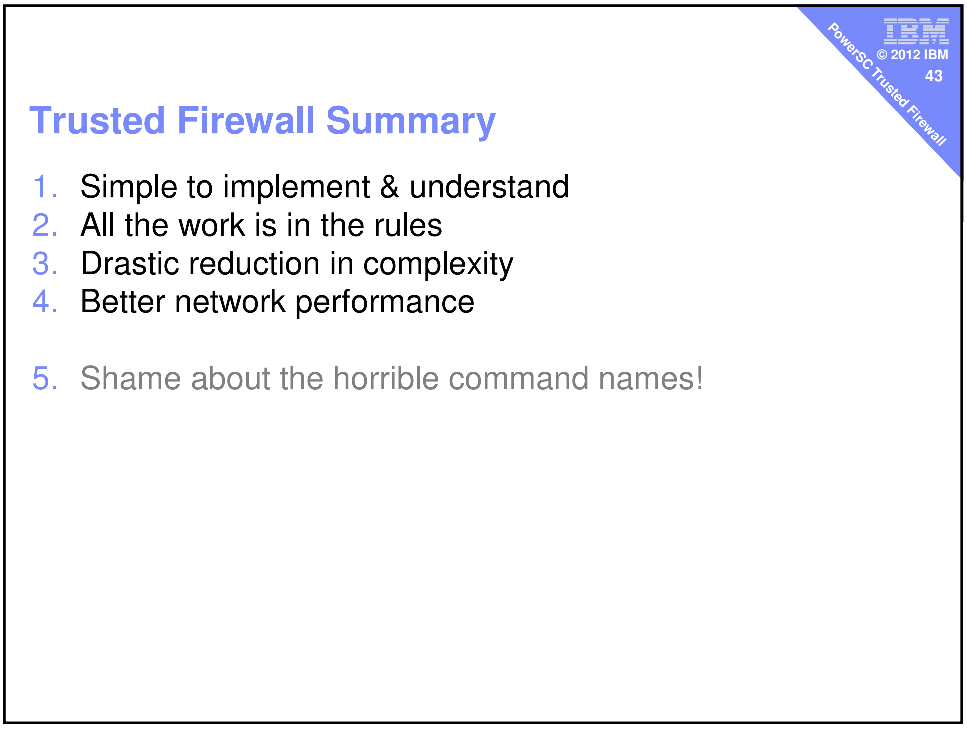
From the PowerSC documentation

Note: Removing the SEA before removing the SVM can result in system failure!

- “System” here means VIOS
- So:
  - rmvfilt -n all ← remove all filters
  - mkvfilt -u ← activate no filters
  - vlantfw -t ← stop the firewall
  - rmdev -dev svm ← remove the device driver
- Now remove the SEA as normal
  - rmtcpip ← remove network IP + host etc
  - lsdev | grep ent ← find SEA
  - rmdev -dev entX ← remove it

PowerSC Trusted Firewall Cheat Sheet

```
All commands on the VIOS
Install powerscStd.svm
▪ $ mksvm                    ← add device driver
▪ $ lsdev | grep svm
▪ $ rmdev -dev svm
Start Firewall
▪ vlantfw -s  to start (-t =stop, -d =display -f =flush)
Firewall Rules
▪ mkvfilt -u                 ← Update kernel tables (after every ODM update)
▪ lsvfilt                    ← List filter rules (-a for kernel rules)
▪ chvfilt                    ← Changes a filter rule
▪ rmvfilt -n all             ← removes all rules

▪ genvfilt                   ← Add filter rules
▪ Allow everything VLAN to VLAN
   – genvfilt -v 4 -a P -z 100 -Z 200
▪ Allow everything host to host
   – genvfilt -v 4 -a P -z 100 -Z 200 -s 9.2.2.2 -d 9.3.3.3
▪ Enable telnet (ssh port=22)
   – genvfilt -v 4 -a P -z 100 -Z 200 -o any -p 0 -O eq -P 23 -c tcp
   – genvfilt -v 4 -a P -z 200 -Z 100 -o any -p 0 -O eq -P 23 -c tcp
▪ Enable ftp      (sftp port=115)
   – genvfilt -v 4 -a P -z 100 -Z 200 -s 9.2.2.2 -d 9.3.3.3  -o any -p 0 -O eq -P
     21 -c any
   – genvfilt -v 4 -a P -z 200 -Z 100 -s 9.3.3.3 -d 9.2.2.2  -o any -p 0 -O eq -P
     21 -c any
▪ Enable ping everything everywhere
   – genvfilt -v 4 -a P -z 100 -Z 200 -c icmp
▪ USE A SCRIPT
```



© 2012 IBM

43

PowerSC Trusted Firewall

Trusted Firewall Summary

1. Simple to implement & understand
2. All the work is in the rules
3. Drastic reduction in complexity
4. Better network performance

5. Shame about the horrible command names!