Unix Cheats

arp -an	Print out arp table
command > filename	send standard output to "filename" (by default error output goes to terminal)
command 2>&1 | more	send error output "2" to(>) same output as standard output "&1"....which is the display. Pipe all through more
command 2> /bin/null | more	Send all the error output to null bucket so output just has clean data in it
CTRL-v CTRL-o ENTER	Resets terminal to text in case binary data changes display
df –h or du –h	'df' is disk free; du is usage in folder hierarchy
dmesg | more	Look at all the boot messages, usually used to find debug errors
dig @<server IP> <domain name> dig @172.17.1.2 www.google.com	Direct query to DNS server @ IP 172.17.1.2 Get IP address/A record of www.google.com
ethtool <ethX e.g. eth0> ethtool -p eth0 ethtool -i eth0	Display physical network card information Blink the LED on eth0 port Get driver version info
find . -follow | xargs ls -ald | more find . –follow –print0 | xargs –0	List all files in directory, follow symbolic links, and print the file modification time "ls -ald". List files with spaces in names so xargs
fgrep –i "search for" *	search for "search for" in inside a file in the current directory. Asterisk is all-files in dir. '–i' ignore case
find . -follow -mmin -10 | xargs ls -ald | more	Search for files modified less than 10 minutes ago and print mod dates. Follow all symbolic links
Filenames: find . -follow -mmin -10 | fgrep "lookforfilename"	Looking for a **filename** that has been modified in last 10 minutes
Filecontent: find . -follow -mmin -10 | xargs fgrep -i "lookforcontent" | more	Search for files modified less than 10 minutes ago and search inside files for **content (not the file-name!)** "lookforcontent"
find / -follow -type f -size +10240k find / -follow -type f -size +10240k -size -102400k	Search for files larger than +10MB. -10240k is less than -10MB File between +10MB and -100MB
Ifconfig ifup eth1 ; ifdown eth1	List all network interfaces - look for collisions Bring interface eth1 up/down without losing routes
ip addr , ip link, ip route	Short printout of ip addresses, link state and routing info
ls -alth	List 'a'll files, 'l' long listing, in 't'ime order with 'h'uman readable sizes
ls -alt | tee /tmp/output.txt	List the directory, the feed it to "tee" which lists on display AND feeds into file at same time
mii-tool –v <interface>	dump stats on interface <eg. mii-tool –v eth0>
more <filename> <somecommand> | more	The more command will print out a file and scroll it so you can read it at your pace. /<searchtext> : search forward for <searchtext> < cr> : read one more line; q: quit < space> : read one page; ? : help
netstat -r	routing tables
netstat -an	Port numbers that are being used or listening on
ntpdate –u [ip] ntpq –pn	Force NTP update from server [ip] Query NTP for sync/poll info, check if its polling
ps –ef, ps -aux, ps -auxwf killall <procname>	print out all processes or with subprocess tree list. Kills the process specified in <procname>
route -n route add –net 1.1.0.0 netmask 255.255.0.0 gw 10.2.1.220 route add –host 2.2.2.2 gw 10.2.1.220 route add default gw 10.2.1.220 route —save	- Print routing table with no DNS resolution - Add network route - Add host route - Add default route - Save the routes permanent (in netconf.C)
[SHIFT]-[Page Up/Down]	On console, scroll up/down
stty rows 80 columns 100	Resets terminal size to window of 80 by 100
tail –f <filename.txt>	Monitor the end of a file for any new information and print to screen in real time.
top <space bar> F -> k F ->n f -> X <> R 1 CPU info: HI, SI, SY are kernel CPU	Realtime display of processes and mem utilization refresh Sort on CPU Sort on MEM Sort on Command name Sort on column going left or right.'R' to reverse sort Toggle reverse sort on/off Show all CPUs SI: Software irq, HI: Hardware IRQ: SY: System

Linux: traceroute –n Windows: tracert -d	Traceroute --- Turn off DNS! Linux uses UDP, Windows uses ICMP
watch [cmd]	Repeat [cmd] every X seconds and compare output

Vi Cheats

vi <filename>	
:q! ZZ or :x :	quit-no save quit: save data command line mode
<arrow keys> <pg up><pg down>	up/down/sideways
x dd or yy p P	delete character delete or copy the current line insert line you just deleted/copied below the current line insert line you just deleted/copied above the current line
/somestring ?somestring n N	search forward for somestring search backwards for somestring search for next instance search for previous instance
i a <ESC> u or U	insert character at current spot and start typing append character after current spot and start typing escape insert/append mode undo last change or ALL changes to line
o O <ESC> u or U	Create line below current line and start inserting characters Create line above current line and start typing characters escape insert mode undo last change or ALL changes to the line

TCPDump Cheats

tcpdump -i <interface> -n
[SEARCHFOR-SEE BELOW]

eg. tcpdump -i eth1 -n host 1.1.1.1

eg. tcpdump -i eth1 -n 'net 1.1.0.0/16 and (icmp or arp)'

Common Options
- -i <interface> commonly eth0/1/2/3
- -n no DNS resolution, faster
- -s 500 size of data packet
- -w binary output file, can be used to feed into Wireshark
- -X ascii and hex output
- -e Ethernet addresses, no IPs
- **VSX**: -vs # only dump in context of VS

Expression Modifiers
- not
- and
- or
- Example:

host 1.1.1.1 or icmp and not arp

SEARCHFOR Operators
- [ether] [src|dst] host <host> | net <net>/len
- [tcp|udp] [src|dst] port <p1> | portrange <port> - <port>
- [src|dst] net <netip>/mask
- proto
- arp
- icmp

Examples **[SEARCHFOR]**:
1. host 1.1.1.1
2. src net 10.1.1.0/24
3. 'src host 1.1.1.1 or (dst host 3.2.1.4 and src port 53)'
4. ether src host 00:0c:29:80:11:0c
5. icmp or arp
6. proto 50

1. dump traffic from both src/dst host 1.1.1.1
2. dump traffic from src net 10.1.1.0
3. dump traffic from src host 1.1.1.1 OR (dst host 3.2.1.4 with a source port of 53/DNS). Need quotes so the () don't confuse /bin/bash.
4. dump traffic from Ethernet MAC src host 00:0c:29:80:11:0c
5. dump icmp and arp traffic
6. dump protocol 50 which is ESP (IPSEC)

ICMP Codes

T	C	Name	T	C	Name
0		Echo Reply	4		Source quence
3	0-3	Destination net, host, proto, port unreachable	5	0-1	Redirect for net, host
	4	Fragmentation re-quired, failed		2-3	Redirect on service for net,host
	5	Source route failed	8		Echo request
	6-7	Dest net, host un-known	11	0	Time exceeded TTL
	8	Source host isolated		1	Time exceeded fragment reassemble
	9-10	Communiation with net,host not permitted	13-14		Timestamp request reply
	11-12	Destination net, host unreachable	15-16		Info request, reply
	13	Communication not permitted			

Protocols

1	ICMP
4	IP
6	TCP
17	UDP
41	IPv6 encapsulat-ed in IPv4
47	GRE
50	ESP
UDP Port 500	IKE
115	L2TP

Check Point SOFTWARE TECHNOLOGIES LTD.

Check Point/Unix/SPLAT Cheat Sheet

Version 2.3

MIDPOINT TECHNOLOGY

Check Point Enterprise Security Management

Product,Licensing, Service, Training, Support

www.midpointtech.com

1-952-837-6206

sales@midpointtech.com

Emergency Contact Info

Conference Bridge Phone	
Help Desk Phone	
Help Desk Email	
Remote Site Phone	
CheckPoint Support Phone CheckPoint Support Support Support ID Number	1 (888) 361 5030
Firewall Team Lead Phone	
Firewall Team Manager Phone	
Security Intranet Site	
Check Point Support	support.checkpoint.com
CheckPoint User Group Forums	www.cpug.com
CheckPoint User Group Forums	www.cpshared.com

Splat Cheats

sysconfig	configure date/time, network, dns, ntp
sysctl -w net.ipv4.ip_forward=1	After unloading policy,(fw unloadlocal), make SPLAT route through the box. Turn on forwarding
cpconfig	change SIC, licenses and more
cpd_sched_config print	Print out CP batch queue -- CP version of crontab
cpinfo	Prints out TONS of FW debug information for help desk
cplic del <sig number>	Delete the license with the hash from cplic print –s
cplic print; cplic print –p; cplic print –x	Print license information in detail from $CPDIR/conf/cp.macro. Print license signature for deletions
cplic print –x | awk '{print $3}' | xargs –n 1 cplic del	Delete all the licenses
cp contract put <file> cplic put –l <file>	Load the license file or the contract/subscription file
cpstart;cpstop;cprestart	start/stop all checkpoint services
cpstat fw	show policy name, policy install time and interface table
expert	change from the initial administrator privilege to advanced privilege
fwaccel stat fwaccel stats fwaccel stats -s fwaccel conns –s fwaccel templates –s	Everything you need to know about SecureXL. Remember SecureXL accelerates these by bypassing kernel and sends through interrupt processing: 1) Traffic from established connections 2) Additional connections from same source
fw ctl arp	List all the proxy arp entries for manual arp brought into the kernel from $FWDIR/conf/local.arp. Also have to enable Global Properties:NAT:Merge manual arps
fw ctl iflist	show interface names
fw ctl pstat	Dump firewall stats
fw fetch 10.0.0.42	get the policy from the firewall manager (use this only if there are problems on the firewall). Downloads rulebases_5_0.fws from 10.0.0.42. This has latest policy in it
fwm load <policyname> <gatewayobjectname>	On smartcenter or MDM, verify and compile and load the policyname onto the targetgateway. When it compiles, it creates a file called $FWDIR/conf/<policyname.pf>, this is the compiled inspect script. From that it generates $FWDIR/conf/<gateway>/rulebases_5_0.fws which is loaded into gateway.
fw stat [–l]	firewall status, should contain the name of the policy and the relevant interfaces, i.e. Standard_5_1_1_1_1 [>eth4] [<eth4] [<eth5] [>eth0.900] [<eth0.900]. –l traffic counts
fw tab fw tab | grep '\--' | more	displays firewall hash tables. Note these are tables that are reserved for firewall kernel hash tables. fw ctl mem Dump out names of tables stored in hash memory ' fw ctl pstat' (hmem)
fw tab -s -t connections	number of connections in state table
fw tab -t xlate -x	clear all translated entries (emergency only)
fw unloadlocal	Remove all policy and security enforcement from SPLAT. Make it a straight linux box basically
fw ver [-k], fwm ver, ver, cpstat os fw ctl conn	Firewall and mgt version and kernel version List of products on firewall
sysconfig	configure date/time, network, dns, ntp
sysctl -w net.ipv4.ip_forward=1	After unloading policy,(fw unloadlocal), make SPLAT route through the box. Turn on forwarding

MDM Cheats

mdsstat	List all the DMS's and their statuses
mdsenv	set MDS environment to a specific domain (listed in mdsstat) so that the 'fw' and 'fwm' command work in the correct context
mcd	change environment to domain specified in mdsenv
mdsstop	Stop all of MDS
mdsstart	Start all of MDS
mdsstop_customer customer	stop a single DMS
mdsstart_customer customer	start a single DMS
mdscmd	command line version of the SDM GUI

GAIA Cheats

Help or <TAB> or <ESC>	
[SPLAT] chattr +i [file]	Lock a file from GAIA modifying it if you make SPLAT level file changes
[SPLAT] clish [GAIA] expert set expert password plain	go into GAIA CLI go into SPLAT Setup the expert pasword
halt/reboot	
save config	everytime you make a change make sure you save
show date/time/timezone	Show date, time, timezone 'set timezone Etc / GMT' —— need spaces here
show interfaces all set interface	Works great with VSX where SPLAT fails Set interface eth0 ip4-address 1.1.1.1 mask-length 24
show routes set static-route	Show routing table Set static-route x.x.x.x/default nethop gateway address y.y.y.y (Won't work on VSX)
Set user xxxx shell /bin/bash Set inactivity-timeout 720	Set the user xxx to use SPLAT and not GAIA as shell Set the timeout to be 12 hours
Show version Set edition default 64-bit	What version of GAIA Use 64-bit GAIA

Debug Cheats

fw ctl chain fw monitor -i -p all > outputascii.txt -o output.cap -x 0 -e "accept ip_src=1.2.3.4 and ping;" -e "accept host(1.2.3.4) and ping;' -vs #	- Print our kernel iIlO chain for reference - Sniff network traffic through iIoO stacks and output to ascii. -i flushes buffers immediately so you get all the output written to the output file. -Dump traffic through iIoO stacks and output to Wireshark format for export. - Sniff network traffic starting at offset 0 -Filter ping packets from SOURCE 1.2.3.4 -Filter ping packets from host 1.2.3.4 [NOTE:$FWDIR/lib/tcpip.def has shortcuts for filtering -- ip_src is one example of a shortcut/macro] -VSX: show output from specific VS instead of ALL VSs (vsenv does not work)
fw ctl debug fw ctl debug -buf 32000 fw ctl debug -m VPN all fw ctl debug + xlate xltrc nat fw ctl debug –m cluster + if conf fw ctl debug –vs # -m cluster + if conf fw ctl kdebug -f > /tmp/debug.out fw ctl debug 0 ----- Turn off!!!!	- Dump kernel tracing debug msgs. Print out all modules and their flags. 'fw' is default module - create buffer inside kernel - used to debug VPN module, all flags - used to debug NAT (default is fw module) - used to debug cluster module, if,conf flags - VSX: debug specific VS member - read tail of buffer and output to file - turn debug OFF!!!!
fw ctl zdebug drop fw ctl zdebug all	Does all the above in one command from the 'fw'/firewall module only. This shows firewalls drops or ALL fw debug
vpn debug trunc vpn debug off ---- Turn off!!!	Debug the setting up of key exchanges and tunnel testing. Output is in $FWDIR/log/vpn.elg and ike.elg
vpn debug mon vpn deub moff —- Turn off!!!	Send vpn traffic to Wireshark readable file $FWDIR/log/ikemonitor.snoop.
vpn tu	list and kill tunnels

HA Cheats(see VSX for VSX HA cmds)

cphaprob ldstat	display sync serialization statistics
cphaprob stat cpstat ha	list the state of the high availability cluster members. Should show active and standby devices.
cphaprob syncstat	display sync transport layer statistics
cphastop/cphastart	Stop/Start the whole HA module for the local cluster member. Different than clusterXL_admin which brings only member down, not the entire HA module.
cphaprob -a if	Display state of interfaces
cphaprob -ia list	List all the monitored devices and their status to figure out why the firewall failed over.
clusterXL_admin up|down	Bring member down in the HA module. NOTE you have to failover other device to get this device back to active, does not automatically flip back to highest priority. Unless you set autorecovery in clusterXL menu for

VSX cheats

vsenv 7 cphastop;cphastart clusterXL_admin up/down	Specify VS: **NOTE!!! vsenv 0 will stop/start the WHOLE chassis!!!!** Stop/Start a specific VS HA module, not just put VS in down state Put the VS into a down state, but don't stop the HA module like above
Vsx GAIA: set vsx on/off	vsx command shell Turn VSX mode on/off and then can use WEBUI to administrate (**NOTE!!! make no routing/interface changes in WebUI!**
vsx set 7 or vsenv 7 vsenv	Set context to virtual system 7 to check connectivity Set context back to 0
vsx stat –[v|l] vsx stat 7	Print stats on all VSX instances or specific instance #7
vsx resctrl monitor enable/disable/show vsx resctrl reset vsx resctrl -[udq]	Enable/Disable cpu monitoring Reset the stats Show percentage of the total CPU resources per VS. –d 24 hours, -u per CPU
vsenv 0 vsx showncs [VS#] vsx_util show_interfaces vsx_util view_vs_conf [GAIA] show interfaces all	With VSswitches, can't see the interface IP addresses. This dump out the interface and routing information for the VS. On mgt station, show interface info On mgt station, show vs configuration info GAIA will show IPs on all addresses, Unix may not if there are Warp interfaces
vsenv 7 fw unloadlocal vsx unloadall	 Unload VS 7 policy Unload policies for VS and all VSs [danger, may impact cluster]

Dump Log Files

fw log –n $FWDIR/log/<filename>.log	Dump a specific SmartTracker log file to the screen, turn off DNS –n to increase speed.
fw log –f –d	Real time monitor the end of the current log file and watch new events come in. Do a 'fw logswitch' first to get to end.
fw logswitch	Archive current log file and start new log file for new events

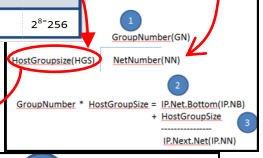
Filesystem/Log Locations

$MDSDIR/log	MDM log files
$MDS_TEMPLATE/log	More MDM log files, not sure why in separate place
$CPDIR/log	cpd logs, setup logs, general check point product logs
$FWDIR/log/	Gateway and Mgt log files. Use "ls -alt" to find recently modified log files
$FWDIR/log/*.elg or .log	component text log files
$FWDIR/log/fw.log	log file that shows up in Smart Tracker
/var/log/messages	Linux OS logs
$FWDIR/database && $FWDIR/state	Where policy is installed on gateway
$FWDIR/conf, $MDSDIR/conf/mdsdb	FW and MDS configuration files
/var/log/dmesg	All startup logs and some ongoing kernel messages. Same as 'dmesg' command

Subnet Masks

Mask	Mask Length	Bitmask	Net Bits	# Nets	Host Bits	Hosts
254	/7,/15,/23,/31	11111110	7	2^7=128	1	2^1=2
252	/6,/14,/22,/30	11111100	6	2^6=64	2	2^2=4
248	/5,/13,/21,/29	11111000	5	2^5=32	3	2^3=8
240	/4,/12,/20,/28	11110000	4	2^4=16	4	2^4=16
224	/3,/11,/19,/27	11100000	3	2^3=8	5	2^5=32
192	/2,/10,/18,/26	11000000	2	2^2=4	6	2^6=64
128	/1,/9,/17,/25	10000000	1	2^1=2	7	2^7=128
0	/0,/8,/16,/24	00000000	0	2^8=256	8	2^{8*}256

Subnet For- mula



4.56.21.123/27 = 123/32 = 3 GN

Subnet Example

3 * 32 = 96 = IP.NB = 4.56.21.96

96+32= 128 = IP.NN = 4.56.21.127

1 2 3