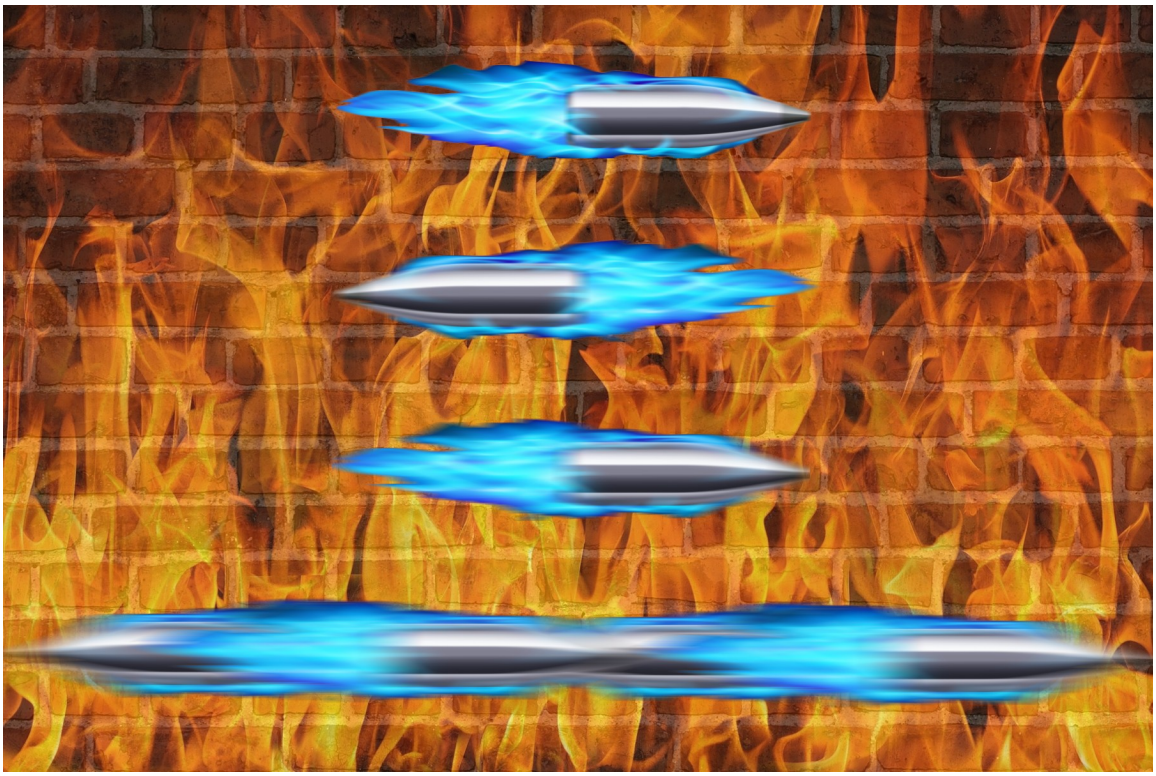


Max Power: Check Point Firewall Performance Optimization Addendum - 7/29/2015

Additional Tips, Tricks, and R77.30 Supplement

Timothy C. Hall

Content Review by Dameon D. Welch-Abernathy and Eric Anderson

Introduction

It has been approximately four months since the release of my book *Max Power: Check Point Firewall Performance Optimization*, and the positive response from readers has been overwhelming. Readers have also contacted me from all over the world with additional firewall performance tips and tricks that I either was not aware of or was not able to include in the original book. This addendum will share with the Check Point community those reader-submitted tips, as well as other useful techniques and utilities I've discovered in the meantime. This addendum document, while free of charge but copyrighted, may be freely copied and distributed provided its content and authorship remains intact.

At the time the book was written, R77.20 was the latest release of code available from Check Point. I'm pleased to report that the R77.30 release issued on May 27, 2015 rectified some of the firewall performance limitations detailed in the book, and added a number of new performance-related features. The next section will list these tips and R77.30 enhancements in page-number order, and serve to supplement the original content of the book. If you own a physical hardcopy, I would recommend that you mark or otherwise indicate on the page numbers listed below that additional content relevant to those page(s) is present in this addendum for future reference.

For those that own a PDF copy of the book purchased from maxpowerfirewalls.com, the embedded permissions of your purchased PDF permit the use of Adobe's Commenting/Annotation tools directly from the free Adobe Reader program; a paid copy of Adobe Acrobat Pro is not necessary. Copy/pasting the following material into annotations on the relevant page numbers will make them readily available right in your original purchased book PDF.

One common theme you will notice in the upcoming section is quite a few references to the relatively new **cpview** tool. My initial impression of this tool, introduced in R77, was that it was simply a fullscreen Character-Based User Interface (CHUI) to various system counters that were already available via CLI commands

previously covered in the book (such as **fw ctl pstat**). As such, it got pretty short shrift in the book, with only a passing mention on page 61 involving CPU contexts.

As many readers have pointed out, however, there is a wealth of performance-related information available via this tool that does not appear to be readily available anywhere else. As an example, for the new R77.30 Priority Queues feature, statistical data appears to only be accessible through **cpview**. If there is one major takeaway from this addendum, it is that getting familiar with the **cpview** tool and all it has to offer is well worth your time and effort.

The all-new Firewall Priority Queues feature introduced in R77.30 (sk105762: Firewall Priority Queues in R77.30) certainly looks interesting from a firewall performance perspective, but the tuning recommendations in the book should still be followed first, prior to potentially enabling this new QoS capability. Enabling any form of QoS always introduces some processing overhead; after the firewall is properly tuned you may find you don't need QoS anyway! I'm reserving judgment concerning this new feature until I can get some real-world experience with it under my belt. It can prioritize firewall control traffic such as SSH and OSPF over regular production traffic, and only becomes active when a Firewall Worker Core reaches 100% utilization.

In general, when it comes to QoS, I have absolutely no problem with bandwidth limits, whether they are enforced by the Advanced Networking Blade's QoS feature or by a Limit action in the APCL/URLF policy. The limit mechanism explicitly defines a "loser" when it comes to bandwidth allocation, and the effects against the traffic in question are well understood.

However, things get a bit messy when priorities and guarantees start to be enforced by QoS. Explicitly defining "winners" implicitly creates "losers" that can sometimes be difficult to predict in real-world network conditions. Don't get me wrong: prioritization and guarantees are absolutely the right thing to do for delay-sensitive applications such as voice and video. But anything beyond that needs to be undertaken with a great degree of caution. By definition, networks are dynamic and rapidly-changing; important, production-affecting "losers" that seem to randomly suffer degraded performance can present a very difficult situation for troubleshooting.

Supplementary Material by Page Number

Page 16: If your site does not have the Monitoring Blade present, be sure to check out the **nmon** tool discussed in the Page 58 entry below.

Page 21: If you are unlucky enough to be forced to utilize Emulex NICs (driver name be2net) on your firewall, be aware that a nasty firewall stability issue involving these NICs was fixed in R77.30 and R77.20 jumbo hotfix Take 94 and later. You'll definitely want to install this fix if using Emulex NICs on your firewall.

Page 26: The book recommended always using an even number of physical interfaces in a bonded aggregate Ethernet interface. After some reader questions, I dug into it a little further, as this has been an unofficial recommendation floating around for quite some time. While I was not able to learn the exact nature of the issue, I was assured that it was an Intel driver issue and that it was fixed in R77.30. However, of the four main Intel drivers shipped with Gaia R77.20 (e1000, e1000e, igb, ixgbe), only the e1000e driver was updated (from version 1.2.20 to 2.1.4) in the R77.30 release. So unless your firewall is using the e1000e driver (igb and ixgbe are by FAR the most common though), this recommendation does not appear to be valid. It is also possible that this recommendation is a bit of a myth, created by the fact that some networking vendors do not support using an odd number of physical interfaces when aggregating them using the older EtherChannel technique. If you have further insights, or would like to keep abreast of the evolving knowledge on this topic, stay tuned to this thread at CPUG: https://www.cpug.org/forums/showthread.php/20588-Amalgamating-Joining-Bonds

Page 34: One other potential STP-related issue pointed out by a student of mine, is that different variants of the spanning tree algorithms don't mix well. As an example, if two switches are connected together and one of them is using the original 802.1D standard

STP and the other is using Rapid STP, the various timers will be radically different between the two and cause network stability issues.

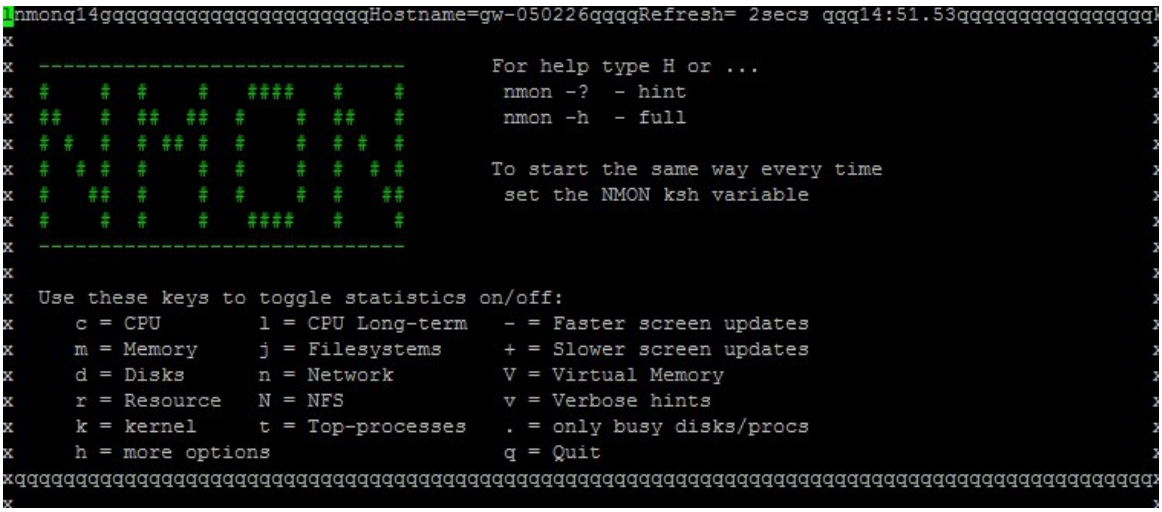
Page 49: ICMP isn't just all about **ping** and **traceroute**; the various types and codes of ICMP datagrams can sometimes indicate that performance-impacting conditions are occurring within the network. Running a **netstat -s** on the firewall shows counters for how many different types of ICMP messages have been received by the firewall. Particular ones that can impact performance and be helpful to investigate further are:

- Fragmentation required but DF set (Type 1, Code 4)
- Precedence cutoff in effect (Type 1, Code 15)
- Source Quench (Type 4, Code 0) – very rare
- Redirect (Type 5)
- Time Exceeded (Type 11)

If nonzero values are noted for any of these in the **netstat -s** output, it is entirely possible they came from the Internet and you have no control over their generation. However, seeing these types of ICMP datagrams arriving on the firewall's internal interfaces via tcpdump should be checked out. To display all ICMP traffic on an internal interface that is not associated with ping testing traffic, use this command:

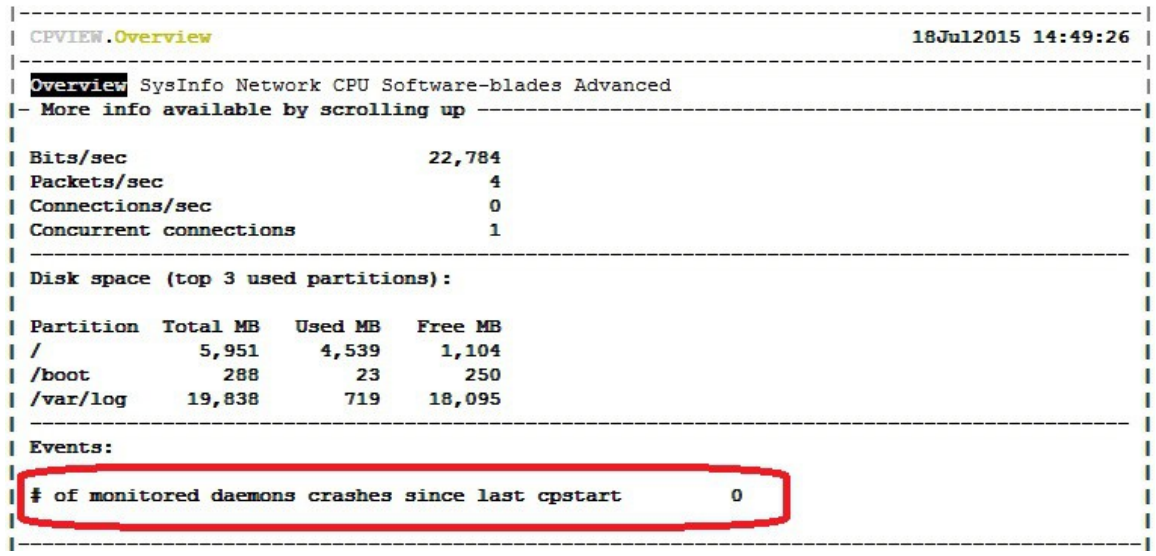
tcpdump -eni (interface name) icmp and not icmp[0]=0 and not icmp[0]=8

Page 58: One additional built-in CPU profiling tool brought to my attention is **nmon:**



Added to Gaia in R76, it serves many of the same functions as the **top** command for monitoring CPU usage on the firewall, but in a more graphical format. As an example, typing lowercase "L" provides a CPU usage graph; hitting "c" will show a graph for a for multi-core systems. While most of these CPU monitoring statistics are also available in **top**, a significant value-add of **nmon** is the ability to monitor and graph disk usage, which is very handy if the **wa** percentage shown by **top** is excessive. **Nmon** can also graph and monitor network interface activity, and serve as a useful stand-in for the **Traffic** and **System Counters** reports that are available via the SmartView Monitor, but only when a Monitoring Blade license is present. Thanks to Yasushi Kono of Arrow ECS for submitting this tip.

Page 59: An easier way to see if cpwd has restarted any firewall processes since the last cpstart or firewall boot is to run **cpview** then select **Overview.** Down-arrow to the bottom of the page, and you will see the counter "# of monitored daemons crashes since last cpstart":



If this value is nonzero, run **cpwd_admin list** to determine which daemon(s) are having a problem.

Pages 59-60: If while running **top** you notice a process called **kipmi0** consuming an excessive amount of CPU on an open hardware firewall, this is a known issue and you should consult sk104316: kipmi0 daemon consumes CPU at 100% on Open Servers running Gaia OS.

Page 76: In addition to hitting "1" while running **top** to see individual core utilizations, the command **cpstat os -f multi_cpu** can also be used to obtain this information. Thanks to Yasushi Kono of Arrow ECS for submitting this tip.

Pages 84-87: Check Point has created an all-new SK documenting Security Policy best practices here: sk106597: Best Practices - Rulebase Construction and Optimization.

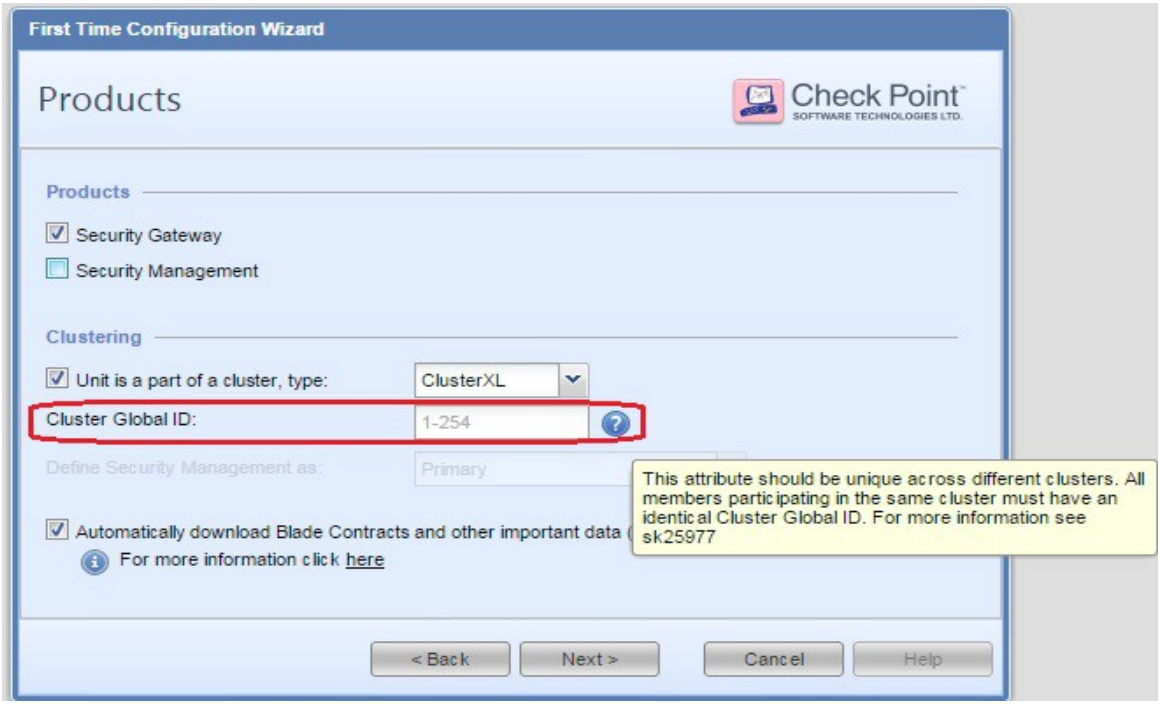
Page 89: As stated in the book, setting **fw_rst_expired_conn** to 1 should always be tried first to gracefully terminate application-based connections that aren't closing properly and impacting perceived application performance. In some cases, however, this will not fully remediate the situation, and you will be forced to go one step further with

this: **fw ctl set int fw_reject_non_syn 1.** A classic example of an application that requires this firewall setting is SAP HANA traffic. This setting also handles client port reuse out of state errors when RST packets from the server to the clients get lost (e.g. due to policy install or packet loss).

Bear in mind, however, that this setting is quite likely to make your "Allsafe Cybersecurity" auditor/penetration tester upset with you, since the firewall will now issue a TCP RST for ***all*** received packets that are out of state and have the ACK flag set. An auditor running a TCP ACK nmap scan will have it light up like a Christmas tree, with tens of thousands of ports showing up as filtered instead of closed. For this reason, setting **fw_reject_non_syn** to 1 is generally not recommended on an Internet perimeter firewall, but may be acceptable on internal firewalls. Thanks to Andrew Craick of Dimension Data for submitting this tip.

Page 90: The TCP State Logging function was introduced in R77, and is not available on older firewalls. An alternative to this feature on pre-R77 firewalls is using the **Account** option in the **Track** column of a rule. When this option is set for a rule, an Accept entry is created at the start of the connection, just as it is when the **Track** is set to **Log**. However, once the connection finishes (FIN, RST, idle time out etc.), the existing log entry is converted from a **Log** type to an **Account** type. Additional statistics are then provided for the connection, including the connection duration and number of payload/data bytes sent and received by the connection. These statistics can be used to infer the connection's behavior and assist in troubleshooting.

Page 97: R77.30 has added the ability to set the "Magic MAC" value via the Gaia web interface, instead of by hand-editing the fwkern.conf file. During the firewall's post-installation dialog in the Gaia web interface, if "Unit is part of a cluster" is checked, the new field "Cluster Global ID" will become editable:



The Cluster Global ID should be set identically on all members of the same cluster, but be a unique value for different clusters. The Cluster Global ID can also be configured and verified from the CLI in R77.30 and later. The command **cphaconf cluster_id get** will display the current setting, and **cphaconf cluster_id set <Cluster ID Value>** can be used to modify it. See sk25977: Connecting multiple clusters to the same network segment (same VLAN, same switch) for more information. Thanks to Eric Anderson of Netanium for submitting this tip.

Page 139: Some additional commands to check CoreXL licensing status are:

[Expert]# **fw ctl get int fwlic_num_of_allowed_cores**

fwlic_num_of_allowed_cores = 8

[Expert]# **fw ctl get int fwlic_num_of_allowed_cpus**

fwlic_num_of_allowed_cpus = 8

Thanks to Yasushi Kono of Arrow ECS for submitting this tip.

Pages 141 & 146: On these pages it was mentioned that SecureXL can accelerate some IPSec VPN encryption/decryption operations. If SecureXL is enabled on your firewall and you'd like to check if this is occurring, run **fwaccel stats**. Nonzero or rapidly incrementing values in the **Accelerated VPN Path** section of the output indicate that SecureXL acceleration of IPSec traffic is occurring.

Pages 149-151: I'm pleased to report that R77.30 has added the option to substantially improve Firewall Worker Core load distribution via the new Dynamic Dispatcher Feature (sk105261: CoreXL Dynamic Dispatcher in R77.30). This new Firewall Worker Core load-balancing feature is disabled by default in R77.30; as a general rule of thumb you should consider enabling this feature when the following conditions are present*:

- Firewall has 6 or more total cores
- Firewall Worker CPU loads consistently vary from each other by >10% **
- Firewall is NOT using a SAM card (i.e. 21000 series)

  * Enabling this feature may break VoIP traffic being processed by the firewall without a special hotfix, see sk106665: VoIP traffic, or traffic that uses reserved VoIP ports is dropped after enabling CoreXL Dynamic Dispatcher.

  ** Keep in mind that all IPSec VPN and VoIP traffic can only be processed on the lead (lowest-numbered) Firewall Worker Core as specified on page 141 (this limitation has still not been lifted in R77.30). If there is substantial IPSec and/or VoIP traffic traversing the firewall, exclude the lead Firewall Worker Core from consideration when applying the 10% rule of thumb above.

Page 162: When attempting to re-enable SecureXL with IPSec VPNs present, watch for this issue: sk102742: When SecureXL is enabled, traffic through the VPN trusted interface is sent encrypted instead of clear. A separate hotfix must be obtained (this fix does not appear to be included in the current R77.20 jumbo hotfix) or you can upgrade to R77.30.

Page 169: While **fwaccel stats -s** provides useful acceleration packet counters showing total number of packets processed by the SXL/PXL/F2F processing paths, you can also view live throughput numbers for each of the three paths in expressed in pps and Mbps. Run **cpview** then select Advanced...Network...Path:

```
CPVIEW.Advanced.Network.Path                                    18Jul2015 14:46:50
Overview SysInfo Network CPU Software-blades Advanced
CPU-Profiler Memory Network SecureXL CoreXL PrioQ Streaming RAD
Path Direction Size
Path distribution summary (available when SecureXL is on):

Totals              SXL Mbps     SXL pps     PXL Mbps     PXL pps     FW Mbps      FW pps
TCP                        0           0            0           0           0           0
UDP                        0           0            0           0           0           0
Other                      0           0            0           0           0           0

Protocol            SXL Mbps     SXL pps     PXL Mbps     PXL pps     FW Mbps      FW pps
-                          -           -            -           -           -           -
```

Page 173: There are a plethora of stability fixes for 21000-series firewall models that utilize a SAM card in R77.30. If using a SAM card, upgrading to R77.30 (or at least loading the latest R77.20 jumbo hotfix) is highly recommended.

Page 176-178: ***Correction:*** Changing the IPS Scope setting from "Perform Inspection on all Traffic" to "Protect internal hosts only" does NOT potentially make more traffic eligible for the Accelerated Path. Setting "Protect internal hosts only" has a similar effect to creating an IPS Exception, in that it can save CPU time in the Medium Path (PXL). So while changing this setting does have a positive impact on performance (by potentially saving CPU time in the Medium Path), it is not for the reason originally stated in the book (that more traffic is made eligible for Accelerated Path).

Page 194: This section of the book spends a great deal of time trying to reduce firewall CPU load on the Firewall Worker Cores, most of which occurs in the Medium Path

(PXL) on the vast majority of real-world firewalls. R77.30 has introduced an exciting ability to view the *top connections by CPU usage*. This capability is a subset of the new R77.30 Firewall Priority Queues feature (sk105762: Firewall Priority Queues in R77.30), and the good news is that this helpful information can be obtained without having to fully enable this feature. To obtain this ability, run the following command: **fw ctl multik set_mode 1** and reboot the firewall. Now, when running **cpview**, select CPU...Top Connections to see the top individual connections by CPU consumption.

Page 208: The book indicates that **fw ctl zdebug drop** can be used to determine what non-logged IPS signatures are inappropriately dropping traffic. This statement is not completely accurate, because the default reason for the drop shown by zdebug will be very generic, and simply indicate it had something to do with IPS enforcement.

Warning: The following procedure will substantially increase the size and memory requirements of enforcing the compiled policy on the firewall. Use with caution on production systems.

To obtain the actual IPS signature name in the zdebug output, launch the SmartConsole tool GUIdbedit and under Table...Global Properties...Properties change variable **enable_inspect_debug_compilation** from **false** to **true**, and reinstall policy to the firewall. This setting will cause additional debug information to be compiled into the firewall's policy, such that the actual offending IPS signature name will be displayed in the zdebug output.

Page 213: If the **Website Categorization Mode** has been set to **Hold** as recommended in the book, and an unacceptable level of latency is encountered categorizing websites for the URL Filtering function, additional statistics can be enabled in the Resource Advisor Daemon (RAD). The RAD process handles interaction between the firewall and the Check Point cloud for dynamic lookups of content such as URLs. Note that this daemon is also used to update signatures and verify content for the Application Control, Anti-Malware, and Anti-Virus software blades; therefore statistics are available for these other

three functions as well. To enable statistics for the URL filtering function specifically, execute the command **rad_admin stats on urlf**. To view URL caching and cloud interaction statistics, run **cpview** and select Advanced...RAD:

```
|-------------------------------------------------------------------------------|
| CPVIEW.Advanced.RAD                                    19Jul2015 14:04:23     |
|-------------------------------------------------------------------------------|
| Overview SysInfo Network CPU Software-blades Advanced                         |
|-------------------------------------------------------------------------------|
| CPU-Profiler Memory Network SecureXL CoreXL PrioQ Streaming RAD               |
|- More info available by scrolling up ---------------------------------------|
| Name                              APPI       AB       AV       URLF         |
| Found in LDB                       N/A      N/A      N/A      N/A          |
| Sent to Site                       N/A      N/A      N/A      N/A          |
| Round Trip (ms)                    N/A      N/A      N/A      N/A          |
| Hit Count                          N/A      N/A      N/A      N/A          |
| Miss Count                         N/A      N/A      N/A      N/A          |
| Error Count                        N/A      N/A      N/A      N/A          |
| Cache Size (bytes)                 N/A      N/A      N/A      N/A          |
| Max Cache Size (bytes)             N/A      N/A      N/A      N/A          |
| Cache Total Host Records           N/A      N/A      N/A      N/A          |
| Max Cache Total Host Records       N/A      N/A      N/A      N/A          |
| Avg Family Size                    N/A      N/A      N/A      N/A          |
| Max Family Size                    N/A      N/A      N/A      N/A          |
| Expired Requests                   N/A      N/A      N/A      N/A          |
|                                                                               |
|-------------------------------------------------------------------------------|
```

Don't forget to turn off the statistics gathering with the **rad_admin stats off urlf** command when finished!

Pages 220-221: The HTTPS Inspection feature was significantly enhanced in R77.30. While many of the relevant fixes are included in the R77.20 jumbo hotfix, it appears that there are many enhancements exclusive to R77.30 that can improve the functionality and performance of the HTTPS Inspection feature. While the bulk of R77.30 HTTPS Inspection operations appear to still occur in the Firewall Path, the firewall performance impact of Bypass actions and SSL negotiation have been substantially improved.

Page 234: Alternatively, to view the firewall's New Connection Rate (Connections/sec) from the CLI, run the **cpview** command and select **Network**.

Page 275-276: I'm pleased to report that R77.30 has an available built-in fix for the Hide NAT port allocation failures that are much more likely to occur when Hyperspect is enabled, as discussed in #8. Ports used for Hide NAT source port reallocation can be dynamically pooled among the Firewall Worker Cores, instead of being statically assigned. This new feature is not enabled by default. It involves setting the **fwx_nat_dynamic_port_allocation** variable from 0 to 1. There is a separate hotfix available for R77.20 to add this functionality, however it does not appear to be a part of the R77.20 jumbo hotfix yet. See sk103656: Dynamic NAT port allocation feature for more details.

Page 282: If performing lab benchmarking of Check Point firewalls, be sure to enable the following feature: sk105261: CoreXL Dynamic Dispatcher in R77.30. Network load-testing traffic is infamous for its non-uniqueness, which can cause an imbalance of Firewall Worker Core loading and severely crimp firewall throughput results. Also, if performing benchmarking of HTTPS Inspection on a Check Point firewall, be sure to enable HTTPS Inspection in "Test Mode" as detailed here: sk104717: HTTPS Inspection Enhancements in R77.30. HTTPS Inspection Test Mode compensates for similar quirks in HTTPS load-testing traffic and ensures accurate performance results.

Page 283: If you've reached this section of the book and can't obtain acceptable performance from your firewall despite following all the tuning recommendations, and no immediate relief is in sight in the form of newer, faster hardware, consider employing this new R77.30 feature discussed in the Introduction to help make the most of what you do have: sk105762: Firewall Priority Queues in R77.30.