Infrastructure Protection on Cisco IOS Software-Based Platforms

This document describes currently available tools that you can use to protect Cisco IOS software-based infrastructure elements, such as routers and switches, from direct attacks. The same tools can help prevent accidental misconfiguration that may present a risk to the infrastructure. This document also provides deployment guidelines to help facilitate the implementation of these technologies as an integrated security solution, rather than as isolated elements.

The first section provides an overview for the set of basic tools that help mitigate attacks designed to overwhelm the resources available on a device. The next three sections provide a closer look at more advanced features that require additional explanation. The last section provides deployment guidelines explaining how to implement these features in an integrated way. The appendices provide additional useful reference information.

Contents

Corporate Headquarters:

Basic Tools and Techniques for Infrastructure Protection

This section describes the following basic tools and techniques, which provide infrastructure protection for Cisco IOS software-based platforms by helping to control the utilization of the limited resources on a device:

- Tuning input hold queues
- Protecting against ICMP unreachable overload
- Configuring Scheduler allocation

In addition to implementing these basic tools and techniques, Cisco IOS software-based devices should be configured according to the device hardening best practices, which help ensure the security of the device by disabling unnecessary services, and by controlling access to the device. Refer to Appendix A for best practices regarding disabling unnecessary services. Refer to Appendix B for best practices regarding device access control.

Tuning Input Hold Queues

The input hold queues hold packets destined to the router or that need to be processed by the route processor (RP). These queues are maintained for each physical interface, and are shared among subinterfaces. With the exception of asynchronous interfaces, the default input queue size is 75 packets, and when that input queue limit is reached the router starts dropping packets.

Denial of service (DoS) attacks against a router can fill the input queue, knocking out legitimate packets. This is especially dangerous for routing and other control plane traffic, such as Border Gateway Protocol (BGP).

Fortunately, the size of the input queue is configurable per interface using the **hold-queue [size]** command from interface configuration mode. Generally speaking, it is recommended to increase the queue to 1500 packets. However, before doing that, it is a good practice to first check the memory available. The number of packets currently set in the input queue can be seen in the "input queue" field in the output from the **show interface** command.

For more information about the **hold-queue** command, see the following website:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123cgcr/inter_r/int_d1g.htm#wp1142192

Protecting Against ICMP Unreachable Overload

According to Internet standards (RFC 1812), whenever a router drops a packet, it should return an ICMP unreachable packet to the packet source. Routers typically drop incoming packets either because they cannot find a valid route or because the packet should be routed to the Null interface. The latter is typically the case with "black hole" filtering. In the past, the Cisco 12000 series routers processed ICMP unreachable packets with the RP, which left an an opening for DoS attacks against the router. It was possible to overwhelm the RP by generating a large amount of packets that required the creation of ICMP unreachables. At this time, ICMP unreachables are handled by the line cards themselves, which protects the RP.

There are two workarounds to solve the issue that affects these older Cisco routers:

- Disable ICMP unreachable messages
- Rate limit ICMP unreachable traffic

The first workaround is to prevent the router from sending ICMP unreachables by entering the **no ip unreachables** command from interface configuration mode as in the following example:

```
router(config)# interface ethernet 0
router(config-if)# no ip unreachables
```

However, in some cases ICMP unreachables are necessary, so preventing the router from sending them is not always appropriate.

The second workaround is to rate limit the number of ICMP unreachables packets that are sent. In Cisco IOS software-based routers this is possible with the **ip icmp rate-limit** command. The Supervisor 720 (Catalyst 6500 Series Switches and the Cisco 7600 Series Routers) provides a hardware-based rate limiter that is configurable with the **mls rate-limit unicast ip icmp unreachable** command.

The following is an example of the **ip icmp rate-limit** command for Cisco IOS software-based routers:

```
router(config)# ip icmp rate-limit unreachable [df] milliseconds
```

Replace *milliseconds* with the number of milliseconds between two consecutive ICMP unreachable packets. The default is 500 ms, which means that no more than one ICMP unreachable packet is sent every 500 ms. The optional **df** flag rate limits ICMP unreachable packets with code 4 (fragmentation required and DF set). The best practice is to set *milliseconds* to 2000 and use the **df** option.

For more information about the **ip icmp rate-limit unreachable** command, see the following website:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123cgcr/ipras_r/ip1_i1g.htm#wp1081902

For more information about the ICMP unreachable rate limiter and other DoS protection controls available on the Supervisor 720, see the following website:

http://www.cisco.com/en/US/partner/products/hw/switches/ps708/products_configuration_guide_chapter09186a0080435872.html

Configuring Scheduler Allocation

Using the **scheduler allocate** command, which schedules CPU time spent on processes versus interrupts, is another good practice to mitigate an ICMP Unreachable overload condition.

When a Cisco router is fast-switching a large number of packets, it can spend so much time responding to interrupts from the network interfaces that no other processing is performed. Some very fast packet floods can cause this condition. The effect can be reduced by using the **scheduler interval** command, which instructs the router to stop handling interrupts and attend to other business at regular intervals. The following is a typical configuration:

```
router(config)# scheduler interval 500
```

This command specifies that process-level tasks will be handled no less frequently than every 500 milliseconds. This command very rarely has any negative effects, and should be a part of your standard router configuration unless you know of a specific reason to leave it out.

Many newer Cisco platforms use the **scheduler allocate** command instead of **scheduler interval**. You use the **scheduler allocate** command to configure two intervals (in microseconds): an interval for the system to run with interrupts enabled, and an interval for the system to run with interrupts masked. If your system does not recognize the **scheduler interval 500** command, try the **scheduler allocate** command, as shown in the following example:

```
router(config)# scheduler allocate 4000 1000
```

The values in this example are those used by the AutoSecure feature, but you should tune these parameters for your specific platform. For more information about the **scheduler allocate** command, see the following website:

http://www.cisco.com/univercd/cc/td/doc/product/software/ios123/123cgcr/fun_r/cfr_1g06.htm#wp1033741

Infrastructure Protection Access Control Lists

This section describes infrastructure protection access control lists (iACLs), which help prevent or mitigate direct infrastructure attacks by explicitly permitting only authorized traffic to the infrastructure equipment, while allowing transit traffic. Although designed for Internet Service Providers (ISPs), iACLs can also be used to protect the enterprise infrastructure with a few alterations. This section includes the following topics:

- iACL Technology Overview

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.