A Concurrent Real-Time White Paper

2881 Gateway Drive
Pompano Beach, FL 33069
(954) 974-1700
real-time.ccur.com

Real-Time Performance of a Security-Hardened RedHawk™ Linux® System During Denial-of-Service Attacks

By: Rajiv Vaidyanath
Concurrent Consulting Engineer

August 2013

Abstract

This paper is intended for those who wish to understand the real-time performance of a security-hardened RedHawk Linux system when it is under Denial-of-Service (DoS) attacks.

NSA hardening guidelines and STIG rules are applied to harden the RedHawk Linux system. Real-Time performance of the RedHawk Linux system is measured while the attacks are in progress.

A familiarity with RedHawk Linux system administration is assumed in this discussion.

Introduction

A full installation of Kali Linux (http://kali.org) is deployed to launch DoS attacks on a RedHawk Linux system. Kali Linux is a Debian-based Linux distribution that specializes in penetration testing and vulnerability assessment. This distribution is supported and maintained by Offensive Security® (http://offensive-security.com).

The RedHawk Linux system is subjected to two types of DoS attacks.

- Apache Web Service denial
- DHCP Starvation

Web services are made unavailable by saturating the web server's tcp ports with connection requests thereby denying legitimate requests for web pages. This type of DoS attack can be mitigated by appropriate Netfilter/Iptables rules to drop traffic from the offending system.

DHCP starvation denies clients of dhcp leases by overwhelming the dhcp server with bogus lease requests. Linux implements ISC DHCP which makes use of raw sockets. Packets from raw sockets are processed before they can be intercepted and filtered by Netfilter/Iptables rules. Mitigation for this attack has to be configured as a rate limited option in the network switch. We let the RedHawk Linux system endure this incursion for the purpose of real-time performance measurement.

Security-Hardening of RedHawk Linux

Procedures stipulated by NSA's security-hardening guidelines are followed to enhance the security of the RedHawk Linux system, and it is configured as follows:

- SELinux is enabled and targeted policy is enforced.
- Netfilter/Iptables is enabled and configured only to allow necessary network communication. Unwanted ports are closed and undesirable communication is set to be dropped.
- System services that are unnecessary are disabled.
- IPv6 is disabled.
- SUID/SGID bits for binaries are disabled.
- Auditing with STIG audit rules is enabled.

Setup

The following RedHawk Linux system (hostname: *ihawk*) is selected for this benchmark:

Hardware

- TYAN S2880 Motherboard
- Two CPU AMD Opteron 244
- 4G RAM (2G per NUMA node)

Software

- RedHawk Linux 6.3.5 (64 bit)
- ccur-rtbench package
- Apache Web Server package
- DHCP Server package

The ccur-rtbench package provides *cyclictest* to measure real-time response time and *stress* to produce I/O, memory, CPU and disk loads. The system load is monitored via /proc/loadavg and *stress* is adjusted to provide a constant load of 50% of system capacity.

Apache Web Server and DHCP Server are configured and started on *ihawk*.

The attack host (hostname: *kali*) is a Dell Optiplex 790 running 64-bit Kali Linux 1.0 distribution. Two configurable applications - slowHTTPTest and yersinia - are launched from *kali* targeting *ihawk*'s Apache Web Server and DHCP Server.

Netfilter/Iptables rules are set on *ihawk* to drop packets originating from *kali* to its http ports 80 and 443.

```
ihawk:~# iptables -A INPUT -s 10.134.30.151/32 -p tcp -m tcp --dport 80 -j DROP
ihawk:~# iptables -A INPUT -s 10.134.30.151/32 -p tcp -m tcp --dport 443 -j DROP
```

After completing the previous step, a portscan of *ihawk* reveals that its Apache Web Server ports are being packet filtered.

```
kali:~#  nmap -O  ihawk

Starting Nmap 6.25 ( http://nmap.org ) at 2013-05-01 16:31 EDT
Nmap scan report for ihawk (10.134.30.57)
Host is up (0.00029s latency).
PORT    STATE    SERVICE
21/tcp   closed   ftp
22/tcp   open     ssh
23/tcp   closed   telnet
25/tcp   closed   smtp
80/tcp   filtered http
110/tcp  closed   pop3
139/tcp  closed   netbios-ssn
443/tcp  filtered https
445/tcp  closed   microsoft-ds
3389/tcp closed   ms-wbt-server
MAC Address: 00:E0:81:52:9D:37 (Tyan Computer)
Device type: general purpose
Running: Linux 2.6.X|3.X
OS CPE: cpe:/o:linux:linux_kernel:2.6 cpe:/o:linux:linux_kernel:3
OS details: Linux 2.6.32 - 3.6
Network Distance: 1 hop

OS detection performed. Please report any incorrect results at http://nmap.org/submit/
Nmap done: 1 IP address (1 host up) scanned in 2.96 seconds
```

A *dhcping* of *ihawk* from *kali* reveals that an active DHCP server is up and running on *ihawk*.

```
kali:~# dhcping -s ihawk
Got answer from: 10.134.30.57
```

Enabling DoS Attacks

The application layer DoS tool *slowhttptest* is invoked from *kali* targeting *ihawk*'s Apache Web Server.

```
kali:~# slowhttptest -c 1000 -H -g -o stat -i 10 -r 200 -t GET -u http://ihawk.ccur.com -x 24 -p 3 -l 86400

Wed May  1 16:43:36 2013:
Using:
test type:                        SLOW HEADERS
number of connections:  1000
URL:                              http://ihawk.ccur.com/
verb:                             GET
Content-Length header value:      4096
follow up data max size:          52
interval between follow up data:   10 seconds
connections per seconds:          200
probe connection timeout:         3 seconds
test duration:                    86400 seconds
using proxy:                      no proxy
Wed May  1 16:43:36 2013:slow HTTP test status on 0th second:
initializing:       0
pending:            1
connected:      0
error:              0
closed:             0
service available:   YES
Wed May  1 16:43:41 2013:slow HTTP test status on 5th second:
initializing:       0
pending:          947
connected:      0
error:              0
closed:             0
service available:   NO
...
...
```

Note that since packet filtering is active on *ihawk* all packets from *kali* to *ihawk*'s http port are dropped. At this point http://ihawk.ccur.com will still be accessible from any system other than *kali* within the same network.

Next, the layer 2 DoS tool *yersinia* is invoked from *kali* targeting *ihawk*'s DHCP Server.

```
kali:~#  yersinia dhcp -attack 1 -dest  00:E0:81:52:9D:37
<*> Starting DOS attack sending DISCOVER packet...
<*> Press any key to stop the attack <*>
```

Within a few seconds of invoking *yersinia* on *kali*, the DHCP Server on *ihawk* is saturated with bogus lease requests and the following syslog messages appear:

```
May 1 16:46:03 ihawk dhcpd: DHCPDISCOVER from 52:4b:b3:39:4d:10 via eth1: network 10.134.30.0/24: no free leases
May 1 16:4603 ihawk dhcpd: DHCPDISCOVER from 82:03:75:19:f6:20 via eth1: network 10.134.30.0/24: no free leases
May 1 16:46:03 ihawk dhcpd: DHCPDISCOVER from 54:e8:92:0d:d6:fb via eth1: network 10.134.30.0/24: no free leases
May 1 16:46:03 ihawk dhcpd: DHCPDISCOVER from 52:f7:04:25:ba:48 via eth1: network 10.134.30.0/24: no free leases
May 1 16:46:03 ihawk dhcpd: DHCPDISCOVER from d6:49:38:07:bf:cf via eth1: network 10.134.30.0/24: no free leases
May 1 16:46:03 ihawk dhcpd: DHCPDISCOVER from d4:ef:8c:08:a0:67 via eth1: network 10.134.30.0/24: no free leases
May 1 16:46:03 ihawk dhcpd: DHCPDISCOVER from 06:d7:11:06:c4:38 via eth1: network 10.134.30.0/24: no free leases
May 1 16:46:03 ihawk dhcpd: DHCPDISCOVER from d2:ca:1b:42:d4:82 via eth1: network 10.134.30.0/24: no free leases
```

Real-Time Performance Measurement

On *ihawk*, a NUMA node is shielded for interrupts, processes, local timer interrupts and cross node memory activities and *cyclictest* is started.

```
ihawk:~# shield -a n1 -m1
ihawk:~# cyclictest -a 1 -m -p 95
```

System load generator *stress* is started from a different xterm window.

```
ihawk:~# stress --cpu 20 --io 10 --vm 10
```

After 24 hours, *cyclictest* measures a worst-case response time of **16 microseconds**.

Conclusions

Security-hardening of RedHawk Linux diminishes the attack surface and restricts a system from vulnerabilities and exploits. As evidenced by this benchmark, imposing strict security policies on a RedHawk Linux system does not affect its real-time performance.

About Concurrent Real-Time

Concurrent Real-Time is a global leader in innovative solutions serving the aerospace and defense, automotive, and financial industries. As the industries' foremost provider of high-performance real-time computer systems, solutions, and software for commercial and government markets, Concurrent Real-Time focuses on hardware-in-the-loop and man-in-the-loop simulation, data acquisition, and industrial systems. Concurrent's Real-Time product group is located in Pompano Beach, Florida with additional offices in North America, Europe, Asia and Australia. For more information, please visit Concurrent Real-Time at www.real-time.ccur.com.