Global Information Assurance Certification Paper

Ramen Noodles
Micah Arthur
February 6, 2001

Ramen Noodles, Food for thought....

Its 7:29 AM, and your asleep safe in your bed. You knew about the troubles associated with WU-ftpd, but you figured they were not a serious threat. Rising late in the morning, there's no time for breakfast as you dash out the door. There's no need; you have Ramen Noodles waiting for you at the office.

The Ramen worm is a package of well-known exploits for WU-ftpd, rpc.statd, and lprng. The purpose of this article is three-fold:

- To detail the usage and spread of the Ramen worm.

- To take a detailed look at the exploitable vulnerabilities.

- Provide an in-depth 'HOW-TO' on vulnerability removal.

Ramen Toolkit/Worm

The Ramen toolkit/worm was added to the CERT website on Thursday, Jan. 18th, 2001. A 'rootkit' in reference to its designed purpose of establishing root access, and a worm in its ability to replicate and seek out new victims. As reported by Robert Lemos on zdnet.com- "Depending on the version of the operating system it's infecting, Ramen can use well-known flaws in Washington University's FTP server software, a component of the Remote Procedure Call services or the printing software LPrng. These programs are normally placed on servers during the default installation of Red Hat 6.2 and 7.0." The most striking piece of information is, "Patches are available for all the flaws used by the worm."

The basic pattern of the exploit runs like this:

- The initial scan is made on port 21 (FTP), and any FTP banners are retrieved. Ramen then uses this banner information to determine if the server is a vulnerable target.
- If the target is vulnerable, a replication script is started and the exploits are launched.
- Using one of the vulnerable services, Ramen creates the /usr/src/.poop directory on the target, and then requests a copy of itself.
- After replicating itself Ramen opens port 27374, and searches out and replaces the target's 'index.html' file.

- It then disables the vulnerable services by which it gained entry, and the target now becomes attacker and seeks out new hosts.

It is worthy of note that this worm currently only targets RedHat installations. Other Linux distributions could be vulnerable as well, but at this time Ramen only targets RedHat systems running 6.2 and 7.0. Even though only RedHat is a target at this time, a few points need to be considered.

1. Ramen is openly available. Variants of the original have and will continue to pop up. If you're running a server with a distribution other than RedHat, you may yet be a target. The same vulnerabilities that Ramen exploits are also native to SuSe, Mandrake, Caldera, and possibly others. As Michael Warfield of ISS explains, "Ramen is currently known to attack Red Hat systems running vulnerable versions of wu-ftp, rpc.statd, and LPRng. New exploits can be added to the existing worm to expand its capabilities."

2. Ramen is a toolkit. It is a package of exploits that have been automated. Each of these exploits can still be run individually without the automation of Ramen.

3. The vulnerabilities that Ramen is searching for were published up to over 4 months ago. The case in point is a clear example of why administrators need to keep abreast of published vulnerabilities. The "script kiddies" are paying attention, even if we are not.

THE VULNERABILITIES

Let us now take a look at the very heart of the Ramen problem, the vulnerabilities that WU-ftpd, rpc.statd, and lprng contain.

WU-ftpd

The Washington University-ftpd software package is one of the more common FTP programs packaged into Linux and BSD distributions. There are two recently reported vulnerabilities in WU-ftpd; the "Site exec" vulnerability, and the "setproctitle()" vulnerability. Of the two, "Site exec" is the one that the Ramen worm is concerned with. The exploit takes advantage of a character-formatting error in the site exec command in versions 2.60 and earlier. To quote CERT, "The wu-ftpd "site exec" vulnerability is the result of missing character-formatting argument in several function calls that implement the "site exec" command functionality. Normally if "site exec" is enabled, a user logged into an ftp server (including the 'ftp' or 'anonymous' user) may execute a restricted subset of quoted commands on the server itself. However, if a malicious user can pass character format strings consisting of carefully constructed *printf() conversion characters (%f, %p, %n, etc) while executing a "site exec" command, the ftp daemon may be tricked into executing arbitrary code as root."

To identify what FTP service is running on your system you can do the following:

- % ftp 'hostname' This should output a banner message that identifies your FTP program.

- If the version number is not displayed, then login to your ftp service and enter, 'quote stat' at the ftp> prompt.

- The vulnerability in discussion resides in WU-ftpd versions 2.60 and earlier.

According to various websites, this vulnerability may have been present for as far back as 1993 and any distributions based off of WU-ftpd are likely to also have this vulnerability. Fortunately, future releases contain the necessary fixes, and for those currently running vulnerable version of WU-ftpd patches are available. RedHat recommends, "If you have not updated your system, we recommend you update these packages immediately." and goes on to offer links to download the appropriate RPMs.

rpc.statd

Ramen also searches for the statd Remote Procedure Call. SecurityFocus.com states that,

"A vulnerability exists in the rpc.statd program which is part of the nfs-utils packages, distributed with a number of popular Linux distributions. Because of a format string vulnerability when calling the syslog() function a malicious remote user can execute code as root. The rpc.statd server is an RPC server that implements the Network Status and Monitor RPC protocol. It's a component of the Network File System (NFS) architecture. The logging code in rpc.statd uses the syslog() function passing it as the format string user supplied data. A malicious user can construct a format string that injects executable code into the process address space and overwrites a function's return address, thus forcing the program to execute the code. rpc.statd requires root privileges for opening its network socket, but fails to drop these privileges later on. Thus code executed by the malicious user will execute with root privileges. Debian, Red Hat and Connectiva have all released advisories on this matter. Presumably, any Linux distribution which runs the statd process is vulnerable, unless patched for the problem."

The basic idea is that rpc.statd is passing user supplied data to syslog as a format string. If there isn't any validation of this string, machine code can be supplied and executed with rpc.stad's privileges- which are usually root. Again, patches pertaining to this vulnerability have been available for some time. By this time, it should becoming painfully obvious to administrators that regular checks need to be performed to see if vulnerabilities have been reported for their running software. Unlike WU-ftpd, the rpc.ststd vulnerability leaves behind some excellent clues. CERT placed a detailed description of intruder activity along with their advisory. Consider the following example log message,

```
Aug XX 17:13:08 victim rpc.statd[410]: SM_MON request for hostname
containing '/': ^D^D^E^E^F
^F^G^G08049f10
bffff754 000028f8 4d5f4d53 72204e4f 65757165 66207473 6820726f 6e74736f
20656d61 746e6f63 696e6961 2720676e 203a272f
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000000000000000000000000000000000000000000000000000000000000000000000bffff7
040000000000000000000000000000000000000000000000bffff7050000bffff7060000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000bffff707K^v

^[^^. #^
1
F'F* FF+,
NV1@/bin
/sh -c echo 9704 stream tcp
nowait root /bin/sh sh -i >> /etc/inetd.conf;killall -HUP inetd
```

If you see entries similar to the log above, you should take steps to examine if your system has been possibly compromised. Unfortunately, security information tends to be a two edged sword. A successful attack may have cleared such entries. There seems to be no evidence of Ramen clearing files related to the attack, but variations have begun to surface and log scrubbing will most likely be included in future versions. Steps should be taken to apply the vendor updates to your system immediately.

LPrng

The LPrng vulnerability that Ramen executes is very similar in nature to the rpc.statd format string vulnerability. This quote is also from Security Focus, "LPRng is an implementation of the Berkeley lpr print spooling utility. LPRng contains a function, use_syslog(), that returns user input to a string in LPRng that is passed to syslog() as the format string. As a result, it is possible to corrupt the program's flow of execution by entering malicious format specifiers. In testing this has been exploited to remotely elevate privileges. This vulnerability was tested on RedHat 7.0. Earlier versions are likely also to be vulnerable, as well as other operating systems that ship with LPRng. The following is a sample logfile from an attempted LPrng exploit,

```
Nov 26 10:01:00 foo SERVER[12345]: Dispatch_input: bad request line
'BB{E8}{F3}{FF}{BF}{E9}{F3}{FF}{BF}{EA}{F3}{FF}{BF}{EB}{F3}{FF}{BF}
XXXXXXXXXXXXXXXX%.168u%300$nsecurity.%301 $nsecurity%302$n%.192u%303$n
{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}{90}
{90}{90}
1{DB}1{C9}1{C0}{B0}F{CD}{80}{89}{E5}1{D2}{B2}f{89}{D0}1{C9}{89}{CB}C{89}
]{F8}C{89}]{F4}K{89}M{FC}{8D}M{F4}{CD}{80}1{C9}{89}E{F4}Cf{89}]{EC}f{C7}
E{EE}{F}'{89}M{F0}{8D}E{EC}{89}E{F8}{C6}E{FC}{10}{89}{D0}{8D}
M{F4}{CD}{80}{89}{D0}CC{CD}{80}{89}{D0}C{CD}{80}{89}{C3}1{C9}{B2}
?{89}{D0}{CD}{80}{89}{D0}A{CD}{80}{EB}{18}^{89}u{8}1{C0}{88}F{7}{89}
E{C}{B0}{B}{89}{F3}{8D}M{8}{8D}U{C}{CD}{80}{E8}{E3}{FF}{FF}{FF}/bin/sh{A}'
```

If you see entries similar to the above example log, you need to steps to discover is your system has possibly been compromised. Vendor specific patches for this vulnerability are available from their main websites.

<u>CONCLUSION</u>

If a Ramen attack is successful in compromising root, then a number of system modifications will take place. Consider the following CERT information,

- The 'index.html' files on the victim are replaced with an attacker-supplied 'index.html' file

- The '/etc/hosts.deny' file is deleted

- The file '/usr/src/.poop/myip' is created and contains an IP address for the local system

- A script is added to the end of '/etc/rc.d/rc.sysinit' to initiate scanning and exploitation during system startup

<u>For systems with '/etc/inetd.conf'</u>

- An intruder supplied program is added as '/sbin/asp'. A service named 'asp' is added to '/etc/inetd.conf' and inetd is sent a signal to reload the configuration file. This causes inetd to listen on TCP socket number 27374 for incoming connections.

- Usernames 'ftp' and 'anonymous' are added to '/etc/ftpusers'

- Services 'rpc.statd' and 'rpc.rstatd' are terminated

- The system files '/sbin/rpc.statd' and '/usr/sbin/rpc.statd' are deleted

For systems without '/etc/inetd.conf'

- An intruder-supplied program is added as '/usr/sbin/asp'. A service named 'asp' is added to '/etc/xinetd.d' and xinetd is sent a signal to reload it's configuration. This causes xinetd to listen on TCP socket number 27374 for incoming connections.

- The 'lpd' service is terminated

- The system file '/usr/sbin/lpd' is deleted and replaced with an empty file

- Usernames 'ftp' and 'anonymous' are added to '/etc/ftpusers'

To detect if your system has been infected, a Ramen detection script has been written by William Stearns and is available at http://www.sans.org/y2k/ramen.htm . If you have found that your system has indeed been compromised, there is a very informative section about Ramen removal that can also be found at the above listed Sans.org website.

Steams, William "Ramen Worm"
Version 0.3 UPDATED 02/05/2001.
http://www.sans.org/y2k/ramen.htm (02/06/01)

Houle, Kevin "Widespread Compromises via "ramen" Toolkit"
CERT® Incident Note IN-2001-01 01/18/01
http://www.cert.org/incident_notes/IN-2001-01.html (02/06/01)

Evans, Chris "Multiple Vendor LPRng User-Supplied Format String Vulnerability"
Security Focus article on LPrng (from bugtraq) 09/25/01
http://www.securityfocus.com/bid/1712 (02/06/01)

Lemos, Robert "Net worm hobbles Linux servers"
Ziff Davis online UPDATED 01/23/01
http://www.zdnet.com/zdnn/stories/news/0,4586,2675147,00.html (02/06/01)

Lemos, Robert "Vandals mutate Ramen Linux worm"
Ziff Davis online 01/22/01
http://www.zdnet.com/zdnn/stories/news/0,4586,2677152,00.html (02/06/01)

Warfield, Michael "Ramen Linux Worm Propagation"
Internet Security Systems Security Alert 01/18/01
http://xforce.iss.net/alerts/advise71.php (02/06/01)