GIAC Level Two Advanced Incident Handling and Hacker Exploits
Practical Assignment for Capitol SANS 2000
Option 2 - Document an exploit, vulnerability or malicious program

By: Aloysius Cheang
DSO National Laboratories, Singapore

Exploit Details:

Name: **Worm.Linux.Ramen**
Discovered on: January 17, 2001
Variants: none known officially but there are modified versions of it (with different file distributions) in the wild.
Operating System: Linux, specifically RedHat Linux versions 6.2 and 7.0
Protocols/Services:
rpc.statd (included in NFS -utils package)
Red Hat 6.2 - wu-ftp (port 21)
Red Hat 7.0 - First Edition for Intel not patched for LPRng
Brief Description: A self -propagating multi -component Linux worm known to infect Red Hat 6.2 and 7.0 machines by infecting the machines with vulnerabilities in wu -ftp, rpc.statd, and LPRng services. This worm has the ability to infect other Linux and Unix machines via a vulnerable wu -ftp version, rpc.statd and LPRng.

Protocols/Services Description

Ramen makes used of the following Red Hat pre-patched vulnerability as the mechanism for spreading.

NFS-Utils
Red Hat Advisory ID: RHSA -2000:043-04
Issue Date: 07-17-2000
Description:
This is an updated of RHSA -2000:043 that contains further upgrade instructions. The rpc.statd daemon in the nfs -utils package shipped in Red Hat Linux 6.0, 6.1, and 6.2 contains a flaw that could lead to a remote root break -in. Version 0.1.9.1 of the nfs-utils package corrects the problem.

In Red Hat Linux 6.0 and 6.1, the rpc.statd daemon was in the knfsd-clients package. The nfs -utils package replaces both the knfsd and knfsd -clients packages shipped in Red Hat Linux 6.0 and 6.1.

On systems running a kernel older than 2.2.16 -3, users should also take this opportunity to upgrade to the latest kernel release.

Patches (listed for i386 architecture only)
i386:
ftp://updates.redhat.com/6.2/i386/nfs -utils-0.1.9.1-1.i386.rpm
sources:
ftp://updates.redhat.com/6.2/SRPMS/nfs -utils-0.1.9.1-1.src.rpm

After installing the new nfs -utils package, the rpc.statd service must be restarted. To do this, run:
/etc/rc.d/init.d/nfslock restart

wu-FTP
Red Hat Advisory ID: RHSA -2000:039-02
Issue Date: 06-23-2000
Description:
A security bug in wu -ftpd can permit remote users, even without an account, to gain root access. An exploitable buffer overrun existed in wu -ftpd code's status update code. It is fixed by adding bounds checking by passing the status strings through %s in the new version.

Patches (listed for i386 architecture only)
Red Hat Linux 5.2:
i386:
ftp://updates.redhat.com/5.2/i386/wu -ftpd-2.6.0-2.5.x.i386.rpm
sources:
ftp://updates.redhat.com/5.2/SRPMS/wu -ftpd-2.6.0-2.5.x.src.rpm

Red Hat Linux 6.2:
i386:
ftp://updates.redhat.com/6.2/i386/wu -ftpd-2.6.0-14.6x.i386.rpm
sources:
ftp://updates.redhat.com/6.2/SRPMS/wu -ftpd-2.6.0-14.6x.src.rpm

LPRng
Red Hat Advisory ID: RHSA -2000:065-06

Issue Date: 09-26-2000
Description:
LPRng contains a critical string format bug in the use_syslog function, which could lead to root compromise. This function returns user input in a string that is passed to the syslog() function as the format string. It is possible to corrupt the print daemon's execution with unexpected format specifiers, thus gaining root access to the computer. The vulnerability is theoretically exploitable both locally and remotely.

Furthermore, syslog() is called for any line of network input which cannot be understood. For example,

```
telnet my.locallan.ip abc

Trying x.x.x.x...
Connected to my.locallan.ip abc (x.x.x.x).
Escape character is '^]'.
%s
Connection closed by foreign host.
<switching to another console>
...
[pid 643] --- SIGSEGV (Segmentation fault) ---
```

=> Leads to a segmentation fault.

Patches (listed for i386 architecture only)
i386:
ftp://updates.redhat.com/7.0/i386/LPRng -3.6.24-2.i386.rpm
sources:
ftp://updates.redhat.com/7.0/SRPMS/LPRng -3.6.24-2.src.rpm

How the Exploit Works

This is the first known worm infecting Red Hat Linux systems. The worm spreads itself from system to system by using well-known Red Hat security vulnerability (that normally will cause "buffer overrun") that allows for uploading to a remote system and running a short piece of code there that then downloads and activates the main worm component.

The worm uses three security vulnerabilities in R ed Hat versions 6.2 and 7.0, these breaches were discovered between June to September (see above) 2000, at least three months before the worm was discovered. In fact, according to the Ramen Crew, author of the worm, the purpose of the Ramen worm was to demonstrate that vigilance in observing the errata list and patching the OS is important. Few Linux-based virus or worm does not means that the Linux OS is secure.

The worm also contains routines that intend to attack FreeBSD and SuSE machines, but these routines are neither activated, nor used in worm code. However, since it is already there, there is nothing to stop any user with reasonable skill sets to activate them or replace some files in the worm components for distribution to cater for these Oses.

Description of the makeup of the worm

This is a multi-component worm that consists of 26 files about 300K in total length. These files are script programs and executable files. The script programs are ".sh" files that are run by a Linux command shell (like DOS BAT files and Windows CMD files). The executable files are standard Linux ELF executables. The main components of the worm are script ".sh" files that are run as hosts, and then run the rest of the files (additional ".sh" files and ELF executables) to perform necessary actions.
The list of components appears as follows:

Asp
asp62
asp7
bd62.sh
bd7.sh
getip.sh
hackl.sh
hackw.sh
index.html
l62
l7
lh.sh
randb62
randb7
s62
s7
scan.sh
start.sh
start62.sh
start7.sh
synscan62
synscan7
w62
w7
wh.sh
wu62

The "62" components are activated under Red Hat 6.2 systems, the "7" components are activated under Red Hat 7.0. The "wu62" file is not used at all.

Brief description of the spreading mechanism

Spreading (infecting a remote Linux machine) is done by a "buffer overrun" attack. This attack is performed as a special packet that is sent to a machine being attacked. The packet has a block of specially prepared data. That block of packet data is then executed as a code on that machine. This code opens a connection to an infected machine, obtains the rest of the worm's code, and activates it. At this moment, the machine is infected, and starts to spread the worm further. The worm is transferred from machine-to-machine as a "tgz" archive (standard UNIX archive) with a "ramen.tgz" name, with 26 worm components inside. While infecting a new machine, the worm unpacks the package there, and runs the main "start.sh" script that then activates other worm components.

The worm components then scan the global network for other Linux machines and upload the worm there if the "buffer overrun" attack is performed successfully.

The worm also appends a command to run its starting ".sh" shell script to a "/etc/rc.d/rc.sysinit" file, and as a result, the worm's components are activated upon each followed system start.
The worm also closes security breaches that have been used to infect the system. So, an infected machine cannot be attacked by the worm twice.

Details

To obtain IP addresses of remote machines in orde r to attack them, the worm scans the available global network for IP addresses using a tool called synscan that has been modified to fit its needs; i.e., operates similar to standard "sniffer" utilities. The initial attack starts with a scan for wuFTP stat d, and lpd. Let's see an attack via FTP (port 21). The worm will retrieve any FTP banners for any FTP services it encounters. The script uses this information to determine if it has contacted a system that may be vulnerable to one of its packaged exploits. Currently, Ramen uses the date encountered in the FTP banner of the system being scanned.

If a vulnerable system is detected, the worm starts a propagation script based on what vulnerability is likely to be present (in this case FTP). The propagation scr ipts and exploits run in parallel with the scanning process.

Using one of the exploitable services, Ramen executes an upload and activate its copy on a remote machine, the worm "buffer overrun" code contains instructions that switch to "root" privileges, runs a command shell, and follows the ensuing commands:

- creates a directory, "/usr/src/.poop", to download the worm, "ramen.tgz" file.
- exports a "TERM=vt100" variable that is necessary for the next step
- runs "lynx" (a WWW browser) that downloads the "ramen.tgz" file from a host machine (the machine from which the worm is spreading)
- unpacks all worm components from the "tgz" archive
- runs the worm startup component: the "start.sh" script

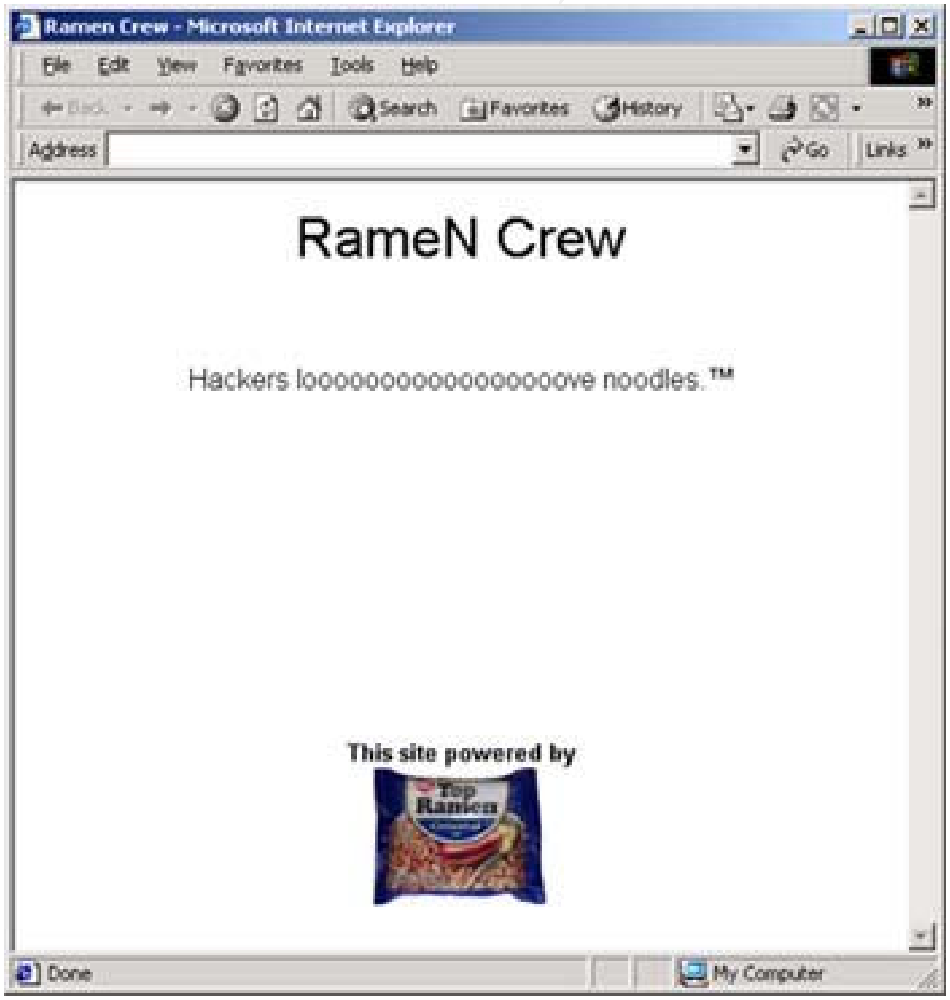
To send a "ramen.tgz" archive, the worm runs an additional server "asp" that sends the worm's "ramen.tgz" archive by request from a worm "buffer overrun" component.

When installed on the new system, Ramen attempts to set up very limited Web-like service on port 27374 to provide for further distribution of the Ramen pack age. The service uses port 27374 to provide a copy of the “ramen.tgz” file to any connection with any request on that port.

What Ramen do

The worm has several payload and other non-infectious routines.

First of all, it finds all "index.html" files (a Web server's starting pages) on a local machine starting from the root directory and replaces them with its own "index.html" file that contains the following text:


Ramen Crew - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites History
Address
Go Links
Done
My Computer

RameN Crew

Hackers looooooooooooooooove noodles. ™

This site powered by

Eat Your Ramen!

The worm deletes the "/etc/hosts.deny" file. This file contains a list of hosts (addresses and/or Internet names) that are denied access to this system (in case a so -called TCP wrapper is used). As a result, any of the restricted machines can acce ss an affected system.
When a new system is infected, the worm sends "notification" messages to three e -mail addresses:

1. the address of just the infected machine
2. gb31337@hotmail.com
3. gb31337@yahoo.com

Ramen then fixes the exploit on the machine, so that others cannot infect it again. On Red Hat 6.2 rpc.statd is removed; on Red Hat 7.0 lpd is removed. Existing FTP services (in inetd on Red Hat 6.2 or in xinetd on Red Hat 7.0) and rpc.statd are also disabled. This action may be to prevent any attempts to re - infect the systems with additional copies of the worm. In addition, the users "ftp" and "anonymous" are added to /etc/ftpusers to close the wu -ftpd hole.

Ramen continues to propagate by using the newly compromised system to scan Class B (/16) wide addres s spaces, searching for port 21 (FTP) and looking for new vulnerable hosts.

On networks and ISPs supporting multicasting, the SYN scanning performed by Ramen can disrupt network traffic when scanning the multicast network range.

Diagram

The diagram below gives the overview of how the worm works and propagates. What the attacker did was to scan the entire network for the availability of the exploits, for example ftp (port 21), not just the two PCs as shown in the diagram. For simplicity sake, not all the arrows are drawn. So if the PC can be and is compromised, it will run the worm component and do the same scanning and infecting process to the rest of the PCs in the network. However, if the PC happens to be infected with the worm, the infection will fail and the next PC will be tried. So the propagation will just carry on and on .

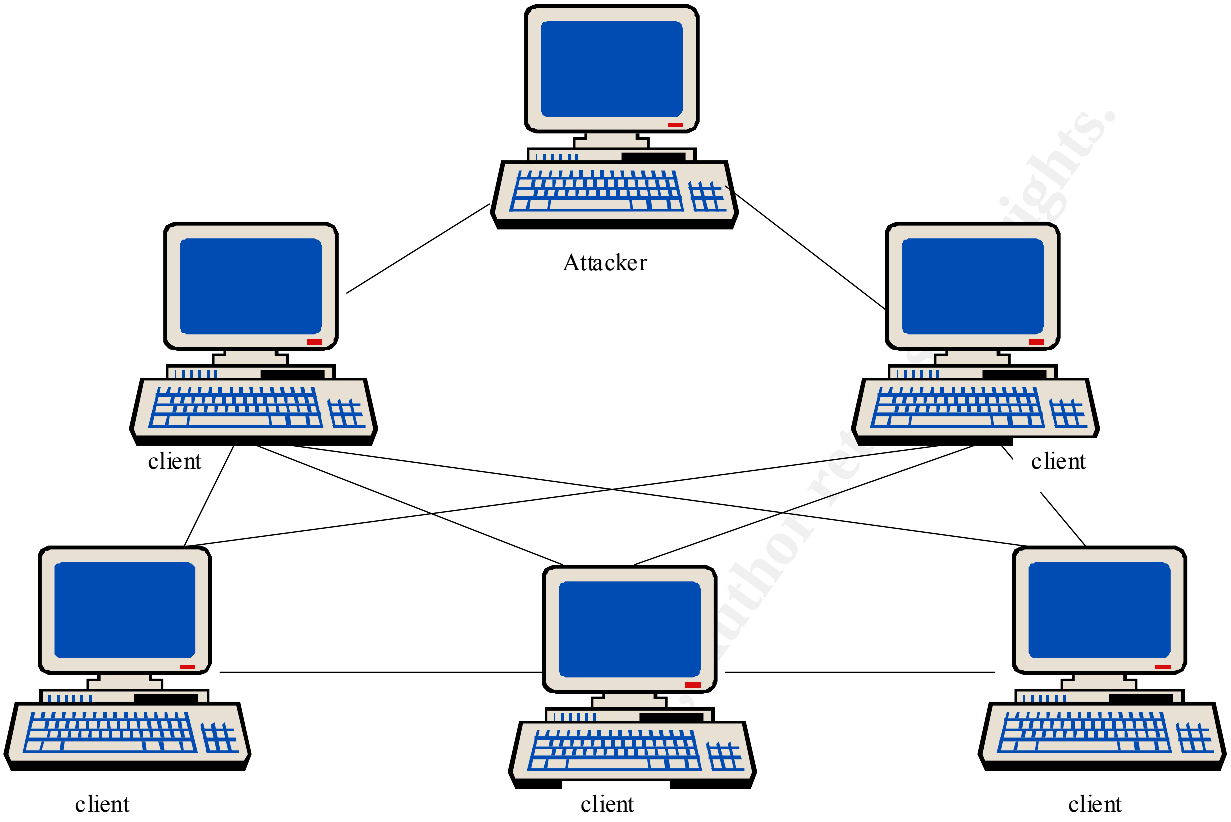


Diagram: The flow of the Ramen worm

How to Use the Exploit

Thus, based on the above, the worm can be successfully be infected by letting it loose to do all those as described in the "How the Worm Works" section or deliberately in the following way:

- do a scan, such as NMAP of your target network to find out their Linux version/distribution and their open ports
- If ftp, lpd or nfs (statd) services are available, then try Ramen worm on them.
- Done!

In the above simple 3 steps (which actually are redundant steps for the worm will do the scanning itself), Ramen will work on the ftp, lpd or nfs known vulnerabilities. However, beware of putting down your email contact, for reconnaissance purpose, use a common email in Hotmail etc to receive all the notifications.

Signature of the Attack

Ramen does not attempt to hide its presence or clean up after itself. It can be detected on a system by the presence of the directory /usr/src/.poop or by the presence of the file /sbin/asp.

How to Protect against It

The worm is targeted at the known security breaches for Red Hat Linux versions 6.2 and 7.0. Thus it is important to visit Red Hat's (or your distributions, for Ramen can be modified to attack other distributions of Linux) security errata page consistently for the price of security is eternal vigilance. Users who have kept their systems up to date with the patches are not impacted by the worm. Thus, due to the general -purpose exploits at the core of this worm, it is advisable to implement the following safeguards to prevent successful attacks from potential variations of this exploit:

1) Disable FTP if it is not a required service. FTP provides information that can be exploited to identify vulnerable systems, even when FTP is not vulnerable.
2) Do not permit outside network access to RPC services, including NFS.
3) Do not permit outside network access to LPR services.
4) Install and maintain all security fixes in a timely manner.

If you are infected by the worm, follow the steps below to remove it:
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. Stop ftp, rpc.statd and lpr services until updates/patches have been installed.

Source Code/ Pseudo Code

The Ramen distribution file, “Ramen.tgz” is included with the submission, together with “Ramen -clean.pl” which is a Perl disinfector that cleans out the ramen worm, obtained from HWA - Security.net under http://hwa-security.net/hot.html.

References:

Red Hat Security Alerts
http://www.redhat.com/support/alerts/ramen_worm.html

Red Hat Errata Page
http://www.redhat.com/support/errata/RHSA -2000-065-06.html

Security Focus
http://www.securityfocus.com

Antiviral Toolkit Pro
http://www.avp.ru

Viruslist.com
http://www.viruslist.com

Symantec Security updates
http://service1.symantec.com/sarc/sarc.nsf/html/Linux.Ramen.Worm.html

SecuritiTeam
http://www.securiteam.com/tools/Ramenfind__Ramen_Worm_detection_and_removal_tool.html

hwa-security.net
http://www.hwa-security.net/