

The systraq Manual

December 2001

Joost van Baal

This document is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version. This is distributed in the hope that it will be useful, but *without any warranty*; without even the implied warranty of *merchantability* or *fitness for a particular purpose*. See the GNU General Public License for more details. You should have received a copy of the GNU General Public License along with this manual (see COPYING); if not, check with <http://www.gnu.org/copyleft/gpl.html> (<http://www.gnu.org/copyleft/gpl.html>) or write to the Free Software Foundation, Inc., 59 Temple Place - Suite 330, Boston, MA 02111, USA.

Table of Contents

| | |
|--|----------|
| 1. Introduction..... | 1 |
| 1.1. What is it? | 1 |
| 1.2. Why? | 2 |
| 2. Installation..... | 3 |
| 2.1. Install scripts and documentation | 3 |
| 2.2. User account..... | 3 |
| 2.3. Set up configuration files..... | 3 |
| 2.4. Inspecting current state of your system, making the first snapshot..... | 4 |
| 2.5. Setting up cronjob..... | 5 |
| 3. Daily Maintenance..... | 5 |
| 4. Internals..... | 6 |
| 4.1. Files used | 6 |
| 4.2. Dependencies | 6 |
| 4.3. The systraq command..... | 7 |
| 4.4. The st_snapshot command..... | 7 |
| 4.5. The filetraq command..... | 7 |
| 5. More information..... | 7 |

1. Introduction

1.1. What is it?

systraq Is a small set of simple scripts monitoring your system, and warning you when system files change. systraq Daily sends you an email listing the state of your system. Furthermore, if critical files have changed, you'll get an email within a shorter notice. It consists of few very small shell scripts. It's written for Debian

GNU/Linux, but very likely runs on any Unix like operating system. (Examples and default configuration will likely be somewhat Debian centered, 'though.)

1.2. Why?

I have some *BSD boxes, which regularly mail me stuff like:

```
checking setuid files and devices:

checking for uids of 0:
root 0
toor 0

checking for passwordless accounts:

hille.mdcc.cx login failures:
Oct 11 11:31:52 hille login: 1 LOGIN FAILURE ON ttyv0
Oct 11 11:31:52 hille login: 1 LOGIN FAILURE ON ttyv0, .^[^[^[MS

hille.mdcc.cx refused connections:

Disk status:
Filesystem 1K-blocks    Used    Avail Capacity  Mounted on
/dev/ad0s1a    39647    27927    8549    77%    /
/dev/ad0s1f   1016303    857921    77078    92%    /usr
/dev/ad0s1g   7026508    6219148    245240    96%    /usr/home
/dev/ad0s1e    19815     6712    11518    37%    /var
procfs         4         4         0    100%    /proc

Network interface status:
Name Mtu Network Address Ipkts Ierrs Opkts Oerrs Coll
lp0* 1500 <Link#1> 0 0 0 0 0
ed0 1500 <Link#2> 00:00:e8:6b:a9:3b 651828 1177 4029190 4 36522

=====
/etc/sshd_config diffs (OLD < > NEW)
=====
lc1,11
< # $OpenBSD: mailer.conf,v 1.3 2000/04/06 18:24:19 millert Exp $
---
> # This is ssh server systemwide configuration file.
>
> Port 22
> #Protocol 2,1
```

On OpenBSD boxes, the shellscripts `/etc/daily`, `/etc/weekly` and `/etc/monthly` kick off the process to generate these status mails. The shellscript `/etc/security` is called, as well as the `mtree(8)` (<http://www.tac.eu.org/cgi-bin/man-cgi?mtree+8+NetBSD-1.5.1>) command.

I very much like this system, taking care of the automatic monitoring of my system. I run GNU/Linux also, these boxes lacked such a system (tripwire is too heavyweigh for my demands.) This system seemed not very portable to GNU/Linux, unfortunately. (Which is another way of stating: I'm too lazy to port the complete system.)

Jeremy Weatherford (<http://xidus.xidus.net/>) wrote FileTraq (<http://filetraq.xidus.net/>) for his Red Hat Linux box. This small tool could be regarded as a first estimate to what I wanted. Jeremy runs it as root, I believe. I want to avoid that as much as possible. I do want to monitor files like `/etc/shadow`, but I do not want to get the diff emailed when these change. I *do* want to get a notice if such a file changes. Christoph Lameter (<http://lameter.com/>)'s `debsums` (<http://packages.debian.org/stable/admin/debsums.html>) is a tool, for monitoring files installed from Debian packages, which has functionality like this. On Debian systems, there's `checksecurity(8)` in the `cron` package, which monitors permissions on device files.

So, I mixed ideas of the BSD 'daily run output' style emails with FileTraq and some other tools. That's systraq.

2. Installation

2.1. Install scripts and documentation

Run

```
$ ./configure
$ make
# make install
```

This will install `st_snapshot`, `systraq` and a patched `filetraq`, which comes with this package, in `/usr/local/bin/`. Furthermore, it will install documentation in `/usr/local/share/doc/systraq/`. Configuration files will get installed in `/usr/local/etc/systraq/`.

2.2. User account

Create a dedicated systraq user account. E.g.

```
# adduser --system --home /usr/local/var/lib/systraq --disabled-password systraq
```

This user will read worldreadable files, and write files under `/usr/local/var/state/systraq/`.

2.3. Set up configuration files

2.3.1. Introduction

Example configuration files are distributed with this manual. (On Debian system's, the examples could be

used as reasonable defaults, except for the `filetraq.conf` file, which needs to be generated for your particular system.) All configuration files are line oriented, lines with a leading `#` are ignored. We give some descriptions here.

2.3.2. `filetraq.conf`

The files listed in `/usr/local/etc/systraq/filetraq.conf` will be checked by **filetraq** for changes in content every half hour. Diff's will be emailed to the administrator. The files `snapshot_pub.stat` and `snapshot_root.stat` should be listed here, as well as `systraq.md5sums` (all these files reside in `/usr/local/var/state/systraq/`). It is advisable to also list every worldreadable file under `/etc/` (and possibly `/usr/local/etc/`) here. You also might like to list each user's `~/.ssh/authorized_keys` and `~/.ssh/authorized_keys2` file here.

All files listed in `filetraq.conf` should exist on your system, and should be worldreadable. (You can monitor non-world readable files in `/etc/` by adding them to `snapshot_root.list`).

2.3.3. `st_snapshots`'s file lists

Daily, **st_snapshot** will check all files as listed in its configuration files, aka listfiles. These listfiles are `/usr/local/etc/systraq/snapshot_{pub,root}.list` and `/usr/local/etc/systraq/snapshot_{pub,root}.home`. These files are installed when running **make install**.

`snapshot_pub.list` Should contain all world readable files for which we want to monitor existence, ownership, permissions and changes in content. It should contain `/usr/local/var/state/systraq/systraq.md5sums` too. `snapshot_root.list` should contain all files which are not world readable, we wanna monitor. `snapshot_{pub,root}.home` should contain files we expect to find in homedirectories of users. All users homedirectories are scanned for files listed in these two listfiles. Think of files like shell startup scripts and stuff in `~/.ssh/` and `~/.rhosts`.

If a file listed in a listfile is a directory, all files residing in this directory, or any subdirectory thereof, gets counted in. Shell wildcards are allowed in the listfiles.

2.4. Inspecting current state of your system, making the first snapshot

Inspect all files listed in the listfiles, and decide whether their content is ok for your security policy. Especially, the `authorized_keys` files need inspection. Once you're happy with their contents, run **st_snapshot** manually: As user `systraq`, run:

```
$ st_snapshot pub > /usr/local/var/state/systraq/snapshot_pub.stat
```

and, as root:

```
# st_snapshot root > /usr/local/var/state/systraq/snapshot_root.stat
```

Inspect the permissions as listed in the output files, and decide whether you're happy with them. Check if all files listed should really be on your system. (One could argue about whether one should have `~/.netrc`,

`~/.rhosts`, `~/.ssh/identity`, `~/.shosts`, `/etc/exports`, `/etc/*hosts.equiv`. Of course, this depends on your planned use of the system.) If you're not happy, fix the permissions and ownerships. You might like to take a look at the OpenBSD `/etc/security` script (<http://www.openbsd.org/cgi-bin/cvsweb/src/etc/security?rev=1.49&type=text/x-cvsweb-markup>) to get inspiration.

Make sure you trust all binary files, which are not in Debian packages (e.g. stuff in `/usr/local/bin/`), as they are on your system now. (You could e.g. reinstall them from trusted sources.) Once you feel safe, generate a file containing md5sums of these files. You can generate this by running e.g., as user `systraq`,

```
$ md5sum `find /usr/local/sbin/ /usr/local/bin/ /usr/local/lib/ \
    /usr/local/share/ -type f | sort` > \
    /usr/local/var/state/systraq/systraq.md5sums
```

Make sure you trust all files in your `filetraq.conf` file, and manually run, as user `systraq`,

```
$ filetraq /usr/local/etc/systraq/filetraq.conf \
    /usr/local/var/state/systraq/filetraq
```

to create the first `filetraq` backup.

2.5. Setting up cronjob

The `systraq-version/etc/systraq` file is installed as `/usr/local/etc/cron.d/systraq`. Activate it, by doing

```
# ln -s /usr/local/etc/cron.d/systraq /etc/cron.d/systraq
```

This makes sure **filetraq** gets run every half hour, **systraq** gets run daily, and the `systraq` status files get updated by running **st_snapshot** each hour.

3. Daily Maintenance

When a user is added to the system: update `filetraq.conf` with this user's `authorized_keys` (or `~/.ssh/authorized_keys`). `filetraq.conf` needs maintenance also once files listed there have been removed by system upgrades, or once files have been added to e.g. `/etc/`.

In case Debian packages are installed with missing `/var/lib/dpkg/info/*.md5sums` file, things break. Consult the examples section in the `debsums` manpage, for a hint on how to deal with these broken packages. Alternatively, one can do:

```
debsums -s > /tmp/sums 2>&1
awk '{print $5}' /tmp/sums > /tmp/pkggs
```

check the contents of `/tmp/pkgs`

```
apt-get --download-only --reinstall install `cat /tmp/pkgs`
cd /var/cache/apt/archives
dpkg -i *.deb
debsums --silent --generate=missing,keep `cat /tmp/pkgs`
apt-get clean
```

4. Internals

4.1. Files used

We list all files used by the *systraq* system, along with a short description of their role.

Files used

`/usr/local/var/lib/systraq/`

homedir of *systraq* user.

`/usr/local/var/state/systraq/`

stores *systraq* status files, should be writable by *systraq* user.

`/usr/local/var/state/systraq/snapshot_pub.stat`

`/usr/local/var/state/systraq/snapshot_root.stat`

stdout of **st_snapshot**, listing permissions, ownership and md5sums of some files, both publicly readable, as well as non-world readable.

`/usr/local/etc/systraq/filetraq.conf`

configuration file for **filetraq**, listing files to get monitored.

`/usr/local/etc/systraq/snapshot_pub.list`

`/usr/local/etc/systraq/snapshot_root.list`

`/usr/local/etc/systraq/snapshot_pub.homelist`

`/usr/local/etc/systraq/snapshot_root.homelist`

configuration files for **st_snapshot**, listing both publicly readable, as well as non-world readable files to get monitored, as well as files to be found in homedirectories.

`/usr/local/var/state/systraq/systraq.md5sums`

md5sums of binary files not in Debian packages, verified by running **systraq**.

4.2. Dependencies

FIXME: diagram listing dependencies: what calls what, what reads and writes what.

4.3. The systraq command

systraq runs various system commands, to inspect the state of the system: what is it doing now, what has it been doing recently, are we running to hardware limitations. It inspect some files in users' homedirectories, as well as some system files, for frequently seen flaws. E.g. unsafe umask setting in shell startup scripts, or unsafe PATH in these scripts. It runs **debsums**, to check md5sums as stated in packaging files with the sums of the actual files running the system. It runs **md5sum** to check md5sums as locally maintained in `/usr/local/var/state/systraq/systraq.md5sums`.

4.4. The st_snapshot command

The **st_snapshot** command comes with its own manpage, distributed with this manual.

4.5. The filetraq command

The **filetraq** command comes with its own documentation, distributed in the FileTraq (<http://filetraq.xidus.net/>) package. Sergio Talens-Oliag wrote a **filetraq** manpage for the Debian package. This manpage is available online (<http://doc.mdcc.cx/cgi-bin/dwww?type=man&location=/usr/share/man/man1/filetraq.1.gz>). I've written a small patch to be able to tweak filetraq's diff style.

5. More information

There is a **st_snapshot** manpage. You can mail me at `<joostvb-systraq-manual-20011207@mdcc.cx>`.