G. Pape

runit - a UNIX init scheme with service supervision

runit is a cross-platform Unix init scheme with service supervision, a replacement for sysvinit, and other init schemes. It runs on **GNU/Linux**, ***BSD**, **MacOSX**, **Solaris**, and can easily be adapted to other Unix operating systems. If *runit* runs for you on any other operating system, please let me know.

runit is discussed on the <supervision@list.skarnet.org> mailing list. Please contact this list and not me privately.

To subscribe send an empty email to <supervision-subscribe@list.skarnet.org>.

Mailing list archives are available at skarnet.org, and gmane.org.

The program runit is intended to run as Unix process no 1, it is automatically started by the runit-init `/sbin/init`-replacement if this is started by the kernel.

runit performs the system's *booting*, *running* and *shutting down* in **three stages**:

- **Stage 1:**
  runit starts `/etc/runit/1` and waits for it to terminate. The system's one time initialization tasks are done here. `/etc/runit/1` has full control over `/dev/console` to be able to start an emergency shell in case the one time initialization tasks fail.
- **Stage 2:**
  runit starts `/etc/runit/2` which should not return until the system is going to halt or reboot; if it crashes, it will be restarted. Normally, `/etc/runit/2` runs runsvdir. In Stage 2 *runit* optionally handles the INT signal (ctrl-alt-del keyboard request on Linux/i386).
- **Stage 3:**
  If *runit* is told to halt or reboot the system, or Stage 2 returns without errors, it terminates Stage 2 if it is running, and runs `/etc/runit/3`. The systems tasks to shutdown and halt or reboot are done here.

These are working examples for Debian sarge: /etc/runit/1, /etc/runit/2, /etc/runit/3.

The program runit-init is intended to replace `/sbin/init`. The command **`init 0`** tells *runit* to halt the system, and **`init 6`** to reboot. Runlevels are handled through the runsvdir and runsvchdir programs. Service dependencies are resolved automatically.

runit is optimized for reliability and small size. The amount of code in process no 1 should be minimal.

See How to install runit for installing *runit*, and How to replace init for configuring *runit* to run as process no 1. See How to use with current init if you want to use *runit* without replacing the current init scheme. Please read the list of Frequently asked questions with answers.

If *runit* on Linux is compiled and linked with the dietlibc, it yields in a statically linked `runit` binary of 8.5k size and this `ps axuw` output on my system:

```
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0     20    16 ?        S     2002   0:02 runit
```

I recommend doing this; for instructions, see How to use dietlibc.

The following distributions are known to include or package *runit*:

- Debian GNU/Linux (as alternative init scheme)
- FreeBSD
- OpenBSD
- NetBSD
- Ubuntu (as alternative init scheme)
- Gentoo
- Linux from Scratch
- Finnix
- SME server
- Linux-VServer
- T2
- GoboLinux
- Dragora GNU/Linux (as default init scheme)

- ArchLinux
- OpenSDE
- Zinux Linux (as default init scheme)
- deepOfix Mail Server (as default init scheme)
- Void Linux (as default init scheme)
- Artix Linux (as default init scheme)

If you know of more distributions, please let me know.

***runit* in use**: I replaced *sysvinit* successfully with *runit* on several server systems and a laptop running Debian/GNU Linux sarge, woody, and potato. Here is an example:

```
# strings /proc/1/exe |grep Id
$Id: runit.c,v 1.7 2002/02/13 09:59:52 pape Exp $
# uptime
 11:59:13 up 365 days, 23:22,  3 users,  load average: 0.01, 0.02, 0.00
# ps axuw |head -n20
USER       PID %CPU %MEM    VSZ  RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0     20   16 ?        S      2002   0:07 runit
root         2  0.0  0.0      0    0 ?        SW     2002   0:00 [keventd]
root         3  0.0  0.0      0    0 ?        SWN    2002   0:51 [ksoftirqd_CPU0]
root         4  0.0  0.0      0    0 ?        SW     2002 144:38 [kswapd]
root         5  0.0  0.0      0    0 ?        SW     2002   0:08 [bdflush]
root         6  0.0  0.0      0    0 ?        SW     2002   7:24 [kupdated]
root       168  0.0  0.0   1652  168 ?        S      2002   0:27 /usr/sbin/cron
root       174  0.0  0.0     36   24 ?        S      2002   1:06 runsvdir /var/service lo
root       176  0.0  0.0     20   20 ?        S      2002   0:00 runsv qmail-send
root       177  0.0  0.0     20   20 ?        S      2002   0:00 runsv getty-5
root       178  0.0  0.0     20   20 ?        S      2002   0:00 runsv getty-4
root       179  0.0  0.0     20   20 ?        S      2002   0:00 runsv getty-3
root       180  0.0  0.0     20   20 ?        S      2002   0:00 runsv getty-2
root       182  0.0  0.0     20   20 ?        S      2002   0:00 runsv socklog-unix
root       183  0.0  0.0   1256    4 tty5     S      2002   0:00 /sbin/getty 38400 tty5 l
root       184  0.0  0.0   1256    4 tty3     S      2002   0:00 getty 38400 tty3 linux
root       185  0.0  0.0     20   20 ?        S      2002   0:00 runsv socklog-klog
root       186  0.0  0.0     20   20 ?        S      2002   0:00 runsv ssh
root       187  0.0  0.0   1256    4 tty4     S      2002   0:00 getty 38400 tty4 linux
# pstree
runit-+-bdflush
      |-cron
      |-gcache
      |-keventd
      |-ksoftirqd_CPU0
      |-kswapd
      |-kupdated
      `-runsvdir-+-runsv-+-multilog
                 |       `-qmail-send-+-qmail-clean
                 |                    |-qmail-lspawn
                 |                    `-qmail-rspawn---qmail-remote
                 |-4*[runsv---getty]
                 |-2*[runsv-+-multilog]
                 |          `-socklog]
                 |-runsv-+-multilog
                 |       `-sshd-+-sshd---sshd---bash---bash---pstree
                 |              `-sshd---sshd---rsync
                 |-runsv---clockspeed
                 |-runsv-+-dnscache
                 |       `-multilog
                 |-runsv---apache-ssl-+-9*[apache-ssl]
                 |                    |-gcache
```

```
|                    `-4*[multilog]
|-7*[runsv-+-multilog]
|          `-tcpserver]
|-4*[runsv-+-multilog]
|          `-tinydns]
|-runsv---uncat
|-2*[runsv-+-multilog]
|          `-tcpsvd]
|-runsv-+-svlogd
|       `-tcpsvd-+-smtpfront-qmail
|                `-smtpfront-qmail---qmail-queue
`-runsv-+-svlogd
        `-tcpsvd---bincimap-up---bincimapd
```

See http://smarden.org/runit/ for recent informations.

Related links:

- minit - a small yet feature-complete init
- svscan as process 1 - by Paul Jarc
- sysvinit - source code
- FreeBSD's init - CVS repository
- NetBSD's init - CVS repository
- OpenBSD's init - CVS repository
- Linux Boot Scripts - by Richard Gooch

Gerrit Pape <pape@smarden.org>