EK

Totally not a hacker

- Email
- Twitter
- Github

Linux-Unix-IT Tips and Tricks #4

Different Linux / Unix / IT tips, notes, howto part 4

Other Parts

- Part 1
- Part 2
- Part 3

Who is writing to a mysql replica

```
# if you are using 5.6, you can do it!
# http://dev.mysql.com/doc/en/connection-summary-tables.html
mysql> desc performance_schema.events_statements_summary_by_user_by_eve
mysql> select user, event_name, count_star, sum_timer_wait, sum_rows_af
    ->   from performance_schema.events_statements_summary_by_user_by_e
    ->  where sum_timer_wait > 0
    ->  order by user, sum_timer_wait desc;
mysql> select * from sys.user_summary_by_statement_type;
mysql> slave1 [localhost] {msandbox} (sys) > select * from user_summary
mysql> slave1 [localhost] {msandbox} (sys) > SELECT * FROM sys.user_sum
mysql> slave1 [localhost] {msandbox} (sys) > SELECT * FROM performance_
```

Display Mount FS in Nice Layout

```
$ mount | column -t
```

FreeBSD version

```
# to see the version and patch level of the installed kernel:
$ freebsd-verion -k
# to see the version and patch level of the installed userland:
$ freebsd-verion -u
# to find out if FreeBSD kernel is running in 32bit or 64bit mode:
$ getconf LONG_BIT
```

Linux namespaces implementations

```
# Linux namespaces implementations:
- mnt: mount points and filesystems isolation
- pid: process isolation
- net: network stack isolation (contains its own routes, network device
- ipc: System V IPC isolation
- uts: hostname isolation
- user: user isolation by means of UIDs

# network namespace
$ ip netns add ns1
$ ip netns
$ ip netns exec ns1 ip link
```

Firewalld in CentOS7\RHEL

Simple Systemd

Manage logging in Systemd

Systemd cgroups

Systemd Targets/Run levels

DebTree – pkg dependency graphs

```
$ systemctl status firewalld
$ firewall-cmd --state
$ firewall-cmd --get-default-zone
$ firewall-cmd --get-active-zones
$ firewall-cmd --zone=public --list-all
$ firewall-cmd --reload
$ firewall-cmd --list-services
$ firewall-cmd --permanent --add-service=ssh
$ firewall-cmd --zone=public --add-port=514/udp
$ firewall-cmd --zone=public --add-forward-port=port=22:proto=tcp:topor
$ firewall-cmd --direct --add-rule ipv4 filter INPUT 0 -p tcp --dport 9
$ firewall-offline-cmd --direct --add-rule ipv4 filter INPUT 0 -p tcp -
$ systemctl disable firewalld
$ systemctl stop firewalld
```

Encrypt Linux Home Dir

```
$ apt-get install ecryptfs-utils
$ sudo ecryptfs-migrate-home -u <login>
# after login
$ ecryptfs-unwrap-passphrase
$ ecryptfs-add-passphrase
# change passpharase
$ ecryptfs-rewrap-passphrase /home/$USER/.ecryptfs/wrapped-passphrase
```

Moreutils

```
chronic - runs a command quietly unless it fails
$ 0 1 * * * chronic backup # instead of backup >/dev/null 2>&1

combine - combine sets of lines from two files using boolean operations
```

```
errno - look up errno names and descriptions
ifdata - get network interface info without parsing ifconfig output
ifne - Run command if the standard input is not empty
$ find . -name core | ifne mail -s "Core files found" root
isutf8 - check whether files are valid UTF-8
```

Send Squid access_log to SIEM

```
# http://wiki.squid-cache.org/Features/LogModules
# /etc/squid/squid.conf
access_log syslog:local6.info

# /etc/rsyslog.conf
local6.info @SIEMIP:514
$ systemctl restart squid.service
```

Send HP-UX syslogd to SIEM

```
# /etc/syslog.conf
*.* @<SIEM_IP>

$ kill -HUP `cat /etc/syslog.pid`
or
$ /sbin/init.d/syslog stop
$ /sbin/init.d/syslog start
```

FreeBSD Vagrant

```
# https://atlas.hashicorp.com/FreeBSD/

# Vagrantfile
Vagrant.configure("2") do |config|
  config.vm.synced_folder ".", "/vagrant", id: "vagrant-root", disabled
  config.vm.box = "freebsd/FreeBSD-11.0-CURRENT"
  config.ssh.shell = "sh"
  config.vm.base_mac = "080027D14C66"
end
$ vagrant up

or
$ vagrant init freebsd/FreeBSD-10.2-RELEASE; vagrant up --provider virt
```

LXC CentOS7 Container

```
$ yum -y install libvirt virt-install
$ systemctl start libvirtd

$ yum -y --installroot=/var/lib/libvirt/filesystems/centos7 \
--releasever=7 install systemd passwd yum \
centos-release vim-minimal procps-ng iproute \
net-tools dhclient policycoreutils

$ chroot /var/lib/libvirt/filesystems/centos7/
$ passwd root
$ echo "pts/0" >> /var/lib/libvirt/filesystems/centos7/etc/securetty

$ virt-install --connect lxc:// --name centos7 --ram 256 --filesystem /
Escape character is ^]
$ virsh --connect lxc:// console centos7
```

Systemd-nspawn CentOS7 Container

```
$ yum -y --installroot=/var/lib/libvirt/filesystems/centos7 \
--releasever=7 install systemd passwd yum \
centos-release vim-minimal procps-ng iproute \
net-tools dhclient policycoreutils

$ systemd-nspawn -D /var/lib/libvirt/filesystems/centos7 --machine cent
$ machinectl login centos_container
$ systemctl stop machine-centos_container.scope
```

Simple Systemd

```
# http://rus-linux.net/MyLDP/boot/img/systemd/systemd4_1.png
# get service states
$ systemctl list-unit-files -t service
$ systemctl list-units -t service --all
$ systemctl is-enabled sshd.service; echo $?

# enable service
systemctl enable clamd@scan.service

# systemd analyze speed
$ systemd-analyze blame

# check service depends
systemctl list-dependencies firewalld.service

# reload all daemons
systemctl daemon-reload
```

Manage logging in Systemd

```
# set timedate
$ timedatectl list-timezones
$ timedatectl set-timezone Europe/Moscow
$ timedatectl status

# to watch all logs
$ journalctl

# logs from last boot
$ journalctl -b

# filter logs with datetime
$ journalctl --since "2015-07-20 17:15:00"
$ journalctl --since yesterday
$ journalctl --since 09:00 --until now
$ journalctl --since 10:00 --until "1 hour ago"

# filter logs with service name
$ journalctl -u sshd.service
$ journalctl -u nginx.service --since yesterday
$ journalctl -u nginx.service -u php-fpm.service —since today

# filter logs with proccess, users or groups name
$ journalctl _PID=1229 (sshd)
$ journalctl _UID=99 (uid=99(nobody) gid=99(nobody) groups=99(nobody))
$ journalctl -F _UID
$ journalctl -F _GUID
$ man systemd.journal-fields

# filter logs with path
$ journalctl /usr/sbin/sshd

# see kernel logs
$ journalctl -k
```

```
# filter logs with error level
$ journalctl -p 3 -b
0 — EMERG
1 — ALERT
2 — CRIT
3 — ERR
4 — WARNING
5 — NOTICE
6 — INFO
7 —DEBUG

# write logs to stdout (default use less)
$ journalctl --no-pager

# write logs with formating
$ journalctl  -u sshd.service -o json
$ journalctl  -u sshd.service -o json-pretty

# formats:
cat
export
short
short-iso
short-monotonic
short-precise
verbose

# see new logs
$ journalctl -n
$ journalctl -n 20

# see logs in tail mode
$ journalctl -f

# count logs space usage
$ journalctl --disk-usage
```

```
# rotate logs
$ journalctl --vacuum-size=1G
$ journalctl --vacuum-time=1years

# write logs to syslog
# /etc/systemd/journald.conf
ForwardToSyslog=yes
```

Systemd cgroups

```
# To get the full hierarchy of control groups, type:
$ systemd-cgls

# To get the list of control group ordered by CPU, memory and disk I/O
$ systemd-cgtop

# To put resource limits on a service (here 500 CPUShares), type:
$ systemctl set-property httpd.service CPUShares=500
$ systemctl show -p CPUShares httpd.service
$ systemctl show httpd.service | grep CPUShares

# cgroup slices
systemctl status user-${UID}.slice

# For example, if we want to limit cpu.shares of all processes of user
$ systemctl set-property user-1000.slice CPUShares=100
$ systemctl daemon-reload
```

Systemd Targets/Run levels

```
# To get the current default run level, type:
$ systemctl get-default

# To move to maintenance mode, type:
$ systemctl rescue

# To set the default run level to non-graphical mode, type:
$ systemctl set-default multi-user.target

# To set the default run level to graphical mode, type:
$ systemctl set-default graphical.target

# To stop, reboot a server, suspend it or put it into hibernation, type
$ systemctl poweroff
$ systemctl reboot
$ systemctl suspend
$ systemctl hibernate
```

DebTree – pkg dependency graphs

```
$ apt-get install debtree
$ debtree nginx
# Generate the dependency graph
$ debtree dpkg >dpkg.dot
#  generate an SVG image from the `.dot'
$ dot -Tsvg -o dpkg.svg dpkg.dot
# Generate  graph for package dpkg as PNG image
$ debtree dpkg | dot -Tpng >dpkg.png
```

Linux-Unix-IT Tips and Tricks #4 was published on August 17, 2015 and last modified on August 17, 2015.

0 Comments EK BLOG 1 Login

♡ Recommend Tweet Share Sort by Best

Start the discussion…

LOG IN WITH

OR SIGN UP WITH DISQUS ?

Name

Be the first to comment.

✉ Subscribe Add Disqus Disqus' Privacy Policy DISQUS

YOU MIGHT ALSO ENJOY

(VIEW ALL POSTS)

- Move from HDD to SSD with ArchLinux
- Linux SysAdm/DevOps Interview Questions
- ArchLinux Installation Guide