Verziókezelt konfigurációmanagement++

Pásztor György, SZTE Klebelsberg Könyvtár

pasztor@ek.szte.hu

Áttekintés

- **Probléma felvetése: konfiguráció verziókezelten**
- **Igények áttekintése**
- **Saját megoldás nyújtotta lehetőségek**
- **Saját megoldás bemutatása, működése**

Lehetséges megoldások a konfiguráció terítésére

- **puppet**
- **cfengine**
- **...**
- **Saját megoldás**

Miért legyen saját fejlesztés

- **Olyan dolgokhoz kellett, ami nem „tömeges”**
  (dns zóna+cfg, dhcp, munin, icinga, … akár puppet is!)
- **Szabályozás kell, hogy ki és mit módosíthat**
- **Az alap konfighoz képest valami ++ (pl. m4 preprocess)**
- **Szintaxis ellenőrzés, ha van**
- **Commit-re automatikus terítés**
- **Egyszerű legyen új elemeket hozzáadni**
- **Bővíthető legyen a pluginkészlete**

A múlt

- **CVS, cvspserver**
- **Shell scriptekkel pre-commit ellenőrzések**
- **Modulonként (cvs) megadható check scriptek**
- **Shell scriptekkel post-commit terítés**

Problémák, tanulságok a régi megoldásnál

- **„rugalmatlan” keretrendszer**
- **Nincs tranzakció a cvs-nél (lényegében egy felturbózott rcs)**
- **Post-commit minden funkcióhoz, szerverenként külön kulcs**
- **Kulcsok elérési jogosultságait szabályozni local userek felé**
- **A szerver jön, saját munkapéldányt update-lni a repóból, és utána reload/restart**
- **Emiatt minden szerveren, a megfelelő init.d scriptbe bele kell nyúlni**

Terv & Megoldás

- **SVN, ssh-key alapú user azonosítás**
- **Így minden az svn user jogaival fut**
- **Egyszerű, áttekinthető pre-commit script pluginekkel**
- **Egyszerű, áttekinthető post-commit script pluginekkel**
- **Általános funkciógyűjtemény:**
  - **m4 előfeldolgozásra, XY szintaxisellenőrzésre, XY konfig terítésére, ...**
- **Egykulcsos (ssh) belépési pont post-commit terítéshez**
- **„Önkonfigurálhatóság”, mint a CVS-nél**

A „net” SVN repó I.

- **dhcp**
  - **dhcp**
  - **kirk**
  - **spock**
- **dns**
  - **cfg-kirk**
  - **cfg-spock**
  - **zones**

A „net” SVN repó II.

- **icinga**
  - **modules**
  - **objects**
  - **stylsheets**
- **munin**
  - **munin-conf.d**
  - **plugin-conf.d**
  - **plugins**
  - **templates**

A „net” SVN repó III.

- **scripts**
  - **post-commit.d**
  - **pre-commit.d**
- **svnag**
  - **bin**
  - **cfg**
- **syslog-ng**

„Önjáró”

- **Az svn user home-jában a repo mellett egy working copy**

```
root@kirk:/var/lib/svn# ls -lG net/hooks/ |grep ^l
lrwxrwxrwx 1 svn   52 Apr 21  2012 commit-access-control.cfg
          -> ../../working_copy/scripts/commit-access-control.cfg
lrwxrwxrwx 1 svn   38 Apr 21  2012 post-commit
          -> ../../working_copy/scripts/post-commit
lrwxrwxrwx 1 svn   37 Apr 21  2012 pre-commit
          -> ../../working_copy/scripts/pre-commit
```

A munkapéldány oldaláról

```
pasztor@intrepid:~/net$ ls -lG scripts/
total 28
-rw-rw-r-- 1 pasztor  740 Nov  7 02:09 commit-access-control.cfg
-rw-rw-r-- 1 pasztor 6161 Nov  7 02:09 common-functions.sh
-rwxrwxr-x 1 pasztor  986 Nov  7 02:09 post-commit
drwxrwxr-x 3 pasztor 4096 Nov  7 02:09 post-commit.d
-rwxrwxr-x 1 pasztor 1182 Nov  7 02:09 pre-commit
drwxrwxr-x 3 pasztor 4096 Nov  7 02:09 pre-commit.d
```

pre-commit I.

```
REPOS="$1"
TXN="$2"
MAILTO="net-admin@bibl.u-szeged.hu"
WC=/var/lib/svn/working_copy
SVNLOOK=/usr/bin/svnlook
SVNSPEC=( "-t" "$TXN" "$REPOS" )
wd=$(mktemp -d)
LOGFILE=$wd/log.pre-commit.txt
# stdout, stderr megy a logba, fd3 megy orig stderr-re
exec 3>&2 >>$LOGFILE 2>>$LOGFILE
. $WC/scripts/common-functions.sh
trap 'echo "Error happened..."; final_mail "SVN pre-commit faliure"; rm -r $wd '
ERR
set -e
```

pre-commit II.

```
svnlook log "${SVNSPEC[@]}" >$wd/svn.log
svnlook dirs-changed "${SVNSPEC[@]}" >$wd/svn.dirs-changed
svnlook changed "${SVNSPEC[@]}" >$wd/svn.changed
do_post_report=false

for i in ${WC}/scripts/pre-commit.d/* ; do
        echo Test: ${i##*/}
        . $i
done
# Finish up
$do_post_report && final_mail "SVN Pre-commit report" || true
rm -r $wd
# All checks passed, so allow the commit.
exit 0
```

pre & post -commit diff I.

- **TXN vs. REV:**

```
-TXN="$2"
+REV="$2"
```

- **SVNSPEC:**

```
-SVNSPEC=( "-t" "$TXN" "$REPOS" )
+SVNSPEC=( "-r" "$REV" "$REPOS" )
```

- **Logfile:**

```
-LOGFILE=$wd/log.pre-commit.txt
+LOGFILE=$wd/log.post-commit.txt
```

pre & post -commit diff II.

- **Trap msg:**

```
-trap 'echo "Error happened..."; final_mail "SVN pre-commit faliure" ;
rm -r $wd ' ERR
+trap 'final_mail "SVN Post-commit faliure" ; rm -r $wd' ERR
```

- **Test vs. Doing:**

```
-for i in ${WC}/scripts/pre-commit.d/* ; do
-       echo Test: ${i##*/}
+for i in ${WC}/scripts/post-commit.d/* ; do
+       echo Doing: ${i##*/}
```

- **Mail Subject:**

```
-$do_post_report && final_mail "SVN Pre-commit report" || true
+$do_post_report && final_mail "SVN Post-commit report" || true
```

pre-commit.d

```
pasztor@intrepid:~/net$ ls -lG scripts/pre-commit.d/
total 32
-rw-rw-r-- 1 pasztor  95 Nov  7 02:09 00_check_log_message
-rw-rw-r-- 1 pasztor 299 Nov  7 02:09 01_access_control
-rw-rw-r-- 1 pasztor  98 Nov  7 02:09 20_dns_cfg_kirk
-rw-rw-r-- 1 pasztor 100 Nov  7 02:09 20_dns_cfg_spock
-rw-rw-r-- 1 pasztor  91 Nov  7 02:09 20_dns_zones
-rw-rw-r-- 1 pasztor  93 Nov  7 02:09 30_dhcp_cfg_dhcp
-rw-rw-r-- 1 pasztor  93 Nov  7 02:09 30_dhcp_cfg_kirk
-rw-rw-r-- 1 pasztor  95 Nov  7 02:09 30_dhcp_cfg_spock
```

post-commit.d

```
pasztor@intrepid:~/net$ ls -lG scripts/post-commit.d/
total 44
-rw-rw-r-- 1 pasztor  98 Nov  7 02:09 00_update_working_copy
-rw-rw-r-- 1 pasztor 270 Nov  7 02:09 20_icinga
-rw-rw-r-- 1 pasztor 198 Nov  7 02:09 20_munin
-rw-rw-r-- 1 pasztor 158 Nov  7 02:09 25_munin-node
-rw-rw-r-- 1 pasztor 158 Nov  7 02:09 30_dns_cfg_kirk
-rw-rw-r-- 1 pasztor 162 Nov  7 02:09 30_dns_cfg_spock
-rw-rw-r-- 1 pasztor 157 Nov  7 02:09 40_dns_zones
-rw-rw-r-- 1 pasztor 156 Nov  7 02:09 50_dhcp_cfg_dhcp
-rw-rw-r-- 1 pasztor 153 Nov  7 02:09 50_dhcp_cfg_kirk
-rw-rw-r-- 1 pasztor 156 Nov  7 02:09 50_dhcp_cfg_spock
-rw-rw-r-- 1 pasztor 123 Nov  7 02:09 99_commit_email
```

post-commit példák

```
pasztor@intrepid:~/net$ cat scripts/post-commit.d/20_munin
if egrep -q '^munin/' <$wd/svn.dirs-changed >/dev/null ;
then
        do_post_report=true
        rsync -aCH --delete munin/ root@monitor.bibladm.intra:/etc/munin/
        ssh root@monitor.bibladm.intra muninpermfix
fi
pasztor@intrepid:~/net$ cat scripts/post-commit.d/25_munin-node
if egrep -q '^munin(/|plugins/|plugin-conf.d/)$' <$wd/svn.dirs-changed
>/dev/null ;
then
        do_post_report=true
        ssh root@monitor.bibladm.intra muninrestart
fi
```

common-functions.sh

- **final_mail () {}**
- **debug_and_run () {}**
- **check_dns_config () {}**
- **affected_zonefiles () {}**
- **check_dns_zones () {}**
- **do_dns_config () {}**
- **get_new_serial () {}**
- **store_new_serial () {}**
- **check_dhcp_config () {}**
- **do_dhcp_config () {}**

common-functions.sh mélyvíz

```
prepare_m4_dir () {}
# function prepare_m4_dir ( [ options ] dirname [M4 macro definitions] ) : generated
dir's name
# Options:
#  -f filelist: give a manualy precreated list of files, to do
#       macroprocessing on
#  -F listfunc: function get list of changed files, to generate the list
#       of files to do macroprocessing on
#  -g MACRONAME: define an m4 macro, which contain the name of the
#       generated dir
#  -G rsyncdir: Instead of mkdir, do an rsync copy of rsyncdir as the
#       initial gendir
#  -c : ??? regenerate only the changed files ???
```

DNS++

```
pasztor@intrepid:~/net$ cat scripts/pre-commit.d/20_dns_cfg_kirk
if grep "^dns/cfg-kirk/" $wd/svn.dirs-changed >/dev/null ; then
        check_dns_config dns/cfg-kirk
fi
pasztor@intrepid:~/net$ cat scripts/pre-commit.d/20_dns_zones
if grep "^dns/zones/" $wd/svn.dirs-changed >/dev/null ; then
        check_dns_zones dns/zones
fi
pasztor@intrepid:~/net$ cat scripts/post-commit.d/30_dns_cfg_kirk
if egrep -q '^dns/cfg-kirk/' <$wd/svn.dirs-changed >/dev/null ; then
        do_post_report=true
        do_dns_config dns/cfg-kirk 'root@kirk.bibladm.intra:/etc/bind/'
fi
pasztor@intrepid:~/net$ cat scripts/post-commit.d/40_dns_zones
if egrep -q '^dns/zones/' <$wd/svn.dirs-changed >/dev/null ; then
        do_post_report=true
        do_dns_zones dns/zones 'root@kirk.bibladm.intra:/var/cache/bind/'
fi
```

svnag: Deploy

```
pasztor@intrepid:~/net$ cat svnag/Makefile
HOST=$(shell uname -n)

install:
	[ -d /root/.ssh ] || install -d -m 700 -o root -g root /root/.ssh
	grep svnag /root/.ssh/authorized_keys || echo \
		'command="/usr/local/sbin/svnag" ssh-rsa AAA... svn@kirk' \
		>>/root/.ssh/authorized_keys
	install -m 755 -o root -g root bin/svnag /usr/local/sbin/svnag
	[ -e cfg/$(HOST).cfg ] && install -m 644 -o root -g root \
		cfg/$(HOST).cfg /etc/svnag.cfg \
	|| install -m 644 -o root -g root cfg/default.cfg /etc/svnag.cfg
```

svnag: Config

monitor:

```
rsync_allowed=( "/etc/munin" "/etc/icinga" )
nagiosdaemon="icinga"
modules_allowed=( muninrestart muninpermfix nagiosrestart nagiospermfix )
```

kirk, spock:

```
rsync_allowed=( "/etc/bind" "/var/cache/bind" "/etc/dhcp" )
modules_allowed=( bindpermfix bindreload dhcppermfix dhcprestart )
```

svnag: Security „modules”

```
rsync_server="rsync --server "
...
check_modules_allowed () {
        [ "${SSH_ORIGINAL_COMMAND:0:${#rsync_server}}" =
                "${rsync_server}" ] && return 0
        for i in ${!modules_allowed[@]} ; do
                [ "${SSH_ORIGINAL_COMMAND}" = "${modules_allowed[$i]}" ] &&
                        return 0
        done
        do_error
}

check_modules_allowed
```

svnag: „modules”

```
case "$SSH_ORIGINAL_COMMAND" in
        rsync*)
                rok=0
                for i in ${!rsync_allowed[@]} ;do
                        p=" . ${rsync_allowed[$i]}/"
                        [ "${SSH_ORIGINAL_COMMAND: -${#p}:${#p}}" = "$p" ] \
                                        && rok=1
                done
                [ $rok -eq 1 ] && exec $SSH_ORIGINAL_COMMAND
                do_error
        ;;
        muninrestart)
                /etc/init.d/munin-node restart
        ;;
            ...
```

Köszönöm a figyelmet!