

Importing of GPG-Keys

Why?

On 2008-01-23 openSUSE-Project decided to use separate keys for each project instead of one common key that was used until then. The respective key is delivered with the first update of a project, so that every user must import the keys by and by. Depending on the number of keys, this

may be a tedious work



Bernhard Walle, member of openSUSE project team, has written an import script (in Python), that is available via http://www.bwalle.de/programme/scripts/smart_fetch_keys_buildservice. This script searches the repositories you have defined in smart, and if they are on opensuse.org, the respective key is fetched.

First all repositories are collected, then the respective keys are searched, and in a third step the keys are imported. I had the problem that one repo had no key, so the script hit the wall. Since I did not succeed in modifying Bernhard's script, I wrote a shellscript to do this task:

The script

```
#!/bin/bash

TEMPREPO="/tmp/search.repo"
TEMPKEY="/tmp/keyfile"

# the base URL we search on
# new: use more than one base URL for your repositories:
URLLIST="http://download.opensuse.org/repositories/
ftp://ftp5.gwdg.de/pub/opensuse/repositories/"
URLLIST="$URLLIST http://software.opensuse.org/download/"

for SOS_URL in $URLLIST; do
    SOS_LEN=$(expr length "$SOS_URL")
    # only URLs containing $SOS_URL please:
    URLLIST=$(smart channel --show | grep ^baseurl | cut -d' ' -f 3 | grep
"$SOS_URL" | sort)

    for URL in $URLLIST; do
        # make sure we have a trailing slash
        echo "$URL" | grep \/ $ >/dev/null 2>&1 || URL="$URL/"

        # inside the directory should be a .repo file
        # so we try to find its name
        # substring handling is somewhat #@%$&# in bash...
        URLAST=${URL#"$SOS_URL"}
```

```
URLAST=$(echo "$URLAST" | rev | cut -d'/' -f 3- | rev | tr -d '/')

# ...finally...
rm -f "$TEMPREPO"
wget -q "${URL}${URLAST}.repo" -O "$TEMPREPO" 2>&1 >/dev/null
# REPO file exists and is not zero sized?
if [ ! -f "$TEMPREPO" -o ! -s "$TEMPREPO" ]; then
    echo "Error getting REPO file for $URLAST from $URL"
    continue
fi

# now we read the URL of the keyfile from the repo file
KEYURL=$(grep ^gpgkey "$TEMPREPO" | cut -d'=' -f 2)
if [ -z "$KEYURL" ]; then
    echo "No key for $URLAST detected"
    continue
fi

# download it...
rm -f "$TEMPKEY"
wget -q "$KEYURL" -O "$TEMPKEY" 2>&1 >/dev/null
if [ ! -f "$TEMPKEY" ]; then
    echo "Error getting keyfile $KEYURL for $URLAST"
    continue
fi

# identify it, maybe it is already there
KEYID=$(gpg "$TEMPKEY" | cut -d'/' -f 2 | cut -d' ' -f 1 | tr 'A-Z' 'a-
z')
INSTALLEDKEYS=$(LANG=C rpm -q "gpg-pubkey-$KEYID" 2>/dev/null)
DOINSTALL=0
echo $INSTALLEDKEYS | grep 'is not installed' >/dev/null 2>&1 &&
DOINSTALL=1

# so, at the very end, import it - or not :-)
if [ $DOINSTALL -eq 1 ]; then
    echo "Importing key $KEYID for $URLAST"
    rpm --import "$TEMPKEY"
    # if you use apt in parallel, you will love the next line ;-)
    gpg --import "$TEMPKEY"
else
    echo "Key $KEYID for $URLAST already imported"
fi
done
done
```

Remarks

The script asks smart for all repositories (whether they are disabled or not), then checks them for

opensuse.org (one may change the SOS_URL to <http://software.opensuse.org/repositories/>, or to <ftp://ftp5.gwdg.de/pub/opensuse/repositories/>, it will also work). Other than Bernhard's script, I check the URLs sequentially for a key and import it (if it is not already there); I do not import them all

together. But of course, all the tricks are shamelessly stolen from his script



```
# smart channel --show | grep ^baseurl | grep  
download.opensuse.org/repositories | wc -l  
392
```

It works for me with 392 repositories



This page is available [in deutsch](#) also.

From:

<http://www.wernerflamme.name/> - **Werners Wiki**

Permanent link:

http://www.wernerflamme.name/doku.php?id=users:werner:getrepokeys_en

Last update: **2008-04-18 07:33**

