independIT Integrative Technologies GmbH

schedulix repository

11. Mai 2020

Legal notice

The Repositories

If the repository file is installed, e.g. by using the schedulix-repo rpm package from the schedulix web site, a file called `schedulix.repo` is added to the `/etc/yum.repos.d` directory.
This file consists of several sections, each defining a repository. Obviously it doesn't make sense to activate them all, but first things first.
So first of all we'd like to see what repositories are offered by the newly installed repository definition file. A simple `yum repolist all` does the trick. Like in

```
[ronald@ocelot ~]$ yum repolist all | egrep "schedulix|repo"
repo id                             repo name                    status
schedulix/7/x86_64                  schedulix                    enabled:     88
schedulix-2.7/7/x86_64              schedulix-2.7                disabled
schedulix-2.8/7/x86_64              schedulix-2.8                disabled
schedulix-2.8-noarch/7              schedulix-2.8-noarch         disabled
schedulix-2.9/7/x86_64              schedulix-2.9                disabled
schedulix-2.9-noarch/7              schedulix-2.9                disabled
schedulix-noarch/7                  schedulix-noarch             enabled:     43
repolist: 10,838
```

And this might look a little complexer than expected. Why a number of repositories and not just a single one?
The first repository, called schedulix, is a repository where you'll find the newest stable build, regardless the release. Currently the latest stable release is version 2.9. If the schedulix repository is enabled, a yum update will upgrade a 2.8 installation to the 2.9 release. This is perfect for a test system, but is somewhat optimistic for a production system.
In case of a production system it is nice to get the bugfixes, but nobody is asking for out-of-the-blue upgrades to the next release. And this is where the other repositories come into play.
If one of the numbered repositories, e.g. schedulix-2.9, is enabled, it is guaranteed that only bug fixes for that release (2.9) will be installed. In order to upgrade it will be necessary to activate the repository with the desired release number and to disable the current repository.
In order to enable a repository either the schedulix.repo file can be edited manually, or it is done with the appropriate `yum-config-manager` command (you might need to install the `yum-utils` package). As an example the following two commands were issued:

```
yum-config-manager --disable schedulix
yum-config-manager --enable schedulix-2.9
```

And the result meets the expectations:

```
[root@ocelot ~]# yum repolist all | egrep "schedulix|repo"
repo id                             repo name                    status
schedulix/7/x86_64                  schedulix                    disabled
schedulix-2.7/7/x86_64              schedulix-2.7                disabled
```

```
schedulix-2.8/7/x86_64                schedulix-2.8              disabled
schedulix-2.8-noarch/7                schedulix-2.8-noarch       disabled
schedulix-2.9/7/x86_64                schedulix-2.9              enabled:     58
schedulix-2.9-noarch/7                schedulix-2.9              disabled
schedulix-noarch/7                    schedulix-noarch           enabled:     43
repolist: 10,808
```

The schedulix repository which was originally enabled is now disabled. Instead the schedulix-2.9 repository has been enabled.
After setting up the repositories to use, it will make sense to delete the cached information. A command like

```
[root@ocelot ~]# yum clean all
Loaded plugins: fastestmirror
Cleaning repos: base extras schedulix-2.9 schedulix-noarch updates
Cleaning up list of fastest mirrors
Other repos take up 392 k of disk space (use --verbose for details)
```

will do the job.

The Packages

After choosing the appropriate repository to install from, a number of packages are ready to be installed.

```
[root@ocelot ~]# yum list | grep schedulix
schedulix-base.x86_64                   2.9-19.el7              @schedulix
schedulix-client.x86_64                 2.9-19.el7              @schedulix
schedulix-repo.noarch                   2.9-15.el7              @schedulix-noarch
schedulix-server-pg.x86_64              2.9-19.el7              @schedulix
schedulix-zope.x86_64                   2.9-19.el7              @schedulix
schedulix-doc.noarch                    2.9-19.el7              schedulix-noarch
schedulix-examples.noarch               2.9-19.el7              schedulix-noarch
schedulix-server-mariadb.x86_64         2.9-19.el7              schedulix-2.9
schedulix-server-rmt.x86_64             2.9-19.el7              schedulix-2.9
```

As visible above the system has schedulix-2.9 installed and yum offers 9 schedulix-2.9 packages. Three of them are independent of the architecture:

- schedulix-doc — The documentation
- schedulix-repo — The repository package
- schedulix-examples — some examples and example jobservers

The schedulix-example package is something special and requires a eclipse-swt to be installed. Unfortunately this package isn't available in the current CentOS/RedHat distribution. But it was part of the el6 distribution and from our experience this installs without issues. It can be easily found in the Internet.
For production installations there's no need to install the example package.
The schedulix-doc package will install the documentation in `/usr/share/doc`.

The other packages are more interesting:

schedulix-base — The base package, required by all other packages

schedulix-client — The package that is required for the installation of jobservers (agents)

schedulix-server-mariadb — The package that'll install the server with a MariaDB Database underneath

schedulix-server-pg — The package that'll install the server with a PostgreSQL Database underneath

schedulix-server-rmt — The package that'll install the server without initializing a database.
This package is used if you want to set up the repository on a remote database server. The required steps are documented in the installation guide

schedulix-zope — The package that'll install the Zope Application Server for the Web-GUI

It is important to note that the three server packages are mutually exclusive. Using rpms doesn't allow multiple installations on a single machine.

It is also important to note that the Zope installation requires Internet access. After the installation the Internet access can be removed.

It is possible to query the packages for more information. E.g.

```
[root@ocelot ~]# yum info schedulix-server-pg
Loaded plugins: fastestmirror, langpacks
Loading mirror speeds from cached hostfile
 * base: ftp.fau.de
 * epel: mirror.infonline.de
 * extras: centos.mirror.net-d-sign.de
 * updates: mirror.checkdomain.de
Available Packages
Name        : schedulix-server-pg
Arch        : x86_64
Version     : 2.8
Release     : 18.el7
Size        : 49 k
Repo        : schedulix-2.8/7/x86_64
Summary     : The schedulix server pg package installs a schedulix server based on \
              an underlying Postgres RDBMS
URL         : http://www.schedulix.org
Licence     : AGPL
Description :
            : schedulix is a production proof open source enterprise job scheduling
            : system.
            : It consists of a central scheduling server and several agents called
            : jobservers, as well as a Zope application server which provides the
            : access to the system by a standard web browser.
            : For operation schedulix requires an installed RDBMS and a suitable
            : JDBC driver.
            : The server package to install should reflect the RDBMS that is installed.
            :
            : On installation a user called 'schedulix' is created. The password is set to
: 'schedulix' (same as user name) and should be changed after installation.
: If one of the server packages is installed, it will restart the required DBMS.
: This might cause side effects if some other software that uses the database
: system is running.
:
: The schedulix server pg package installs a schedulix server based on
: an underlying Postgres RDBMS.
: It loads the convenience package, but does not load the examples.
:
:
: Within the scheduling system there will be a user called 'SYSTEM' with
: password 'G0H0ME' installed. It is advisable to change the password as
: soon as possible.  This is done by changing the SysPasswd property in
: the /opt/schedulix/etc/server.conf file.
: Afterwards it will be a good idea to change the corresponding entry in
: the ~/.sdmshrc file too.
:
: Another issue can be firewall related. In order to be able to access
: the scheduling server you might need to add a rule to the iptables like:
:
: -A INPUT -p tcp -m state --state NEW -m tcp --dport 2506 -j ACCEPT
:
: This will make the port 2506 accessible from other computers.
: (We don't do this, because we don't want to automagically introduce holes
: into your security concept).
:
: The configuration file will be changed. Instead of the 'ident' method we
: need the 'md5' method in order to be able to connect by jdbc.
```

As visible from the above, the software is installed in the `/opt/schedulix` directory. The newly created user called schedulix will be the owner of the installation.