

ADAGUC

Intro

ADAGUC is a WMS-based implementation, made by KNMI. Its distinct feature is to serve time-enabled raster data, from NetCDF file or OPeNDAP server, to (web) clients using the WMS protocol.

For more information on ADAGUC and its features please visit KNMI's ADAGUC pages <http://dev.knmi.nl/projects/adaguc>

ADAGUC consists of

- a server part
- a client/viewer part

This Tech Note describes how to setup ADAGUC on CentOS 6.4 Linux. This was performed on a local Virtual Machine as an exercise before setting up a production machine for ADAGUC.

ADAGUC Server setup

CentOS (<http://www.centos.org>) is based on the same sources as Red Hat Enterprise Linux.

KNMI describes installation on Red Hat Linux here: <http://dev.knmi.nl/projects/adagucserver/wiki/Installation>

This recipe has been followed. In short it is repeated here.

Note: unless otherwise stated, the commands are (Bash) shell commands.

Setup prerequisites

- Verify yum configuration for EPEL repository:

```
[mylocalhost]# yum repolist
```

- Install from EPEL repository (compiling netcdf and hdf5 is not needed in this case):

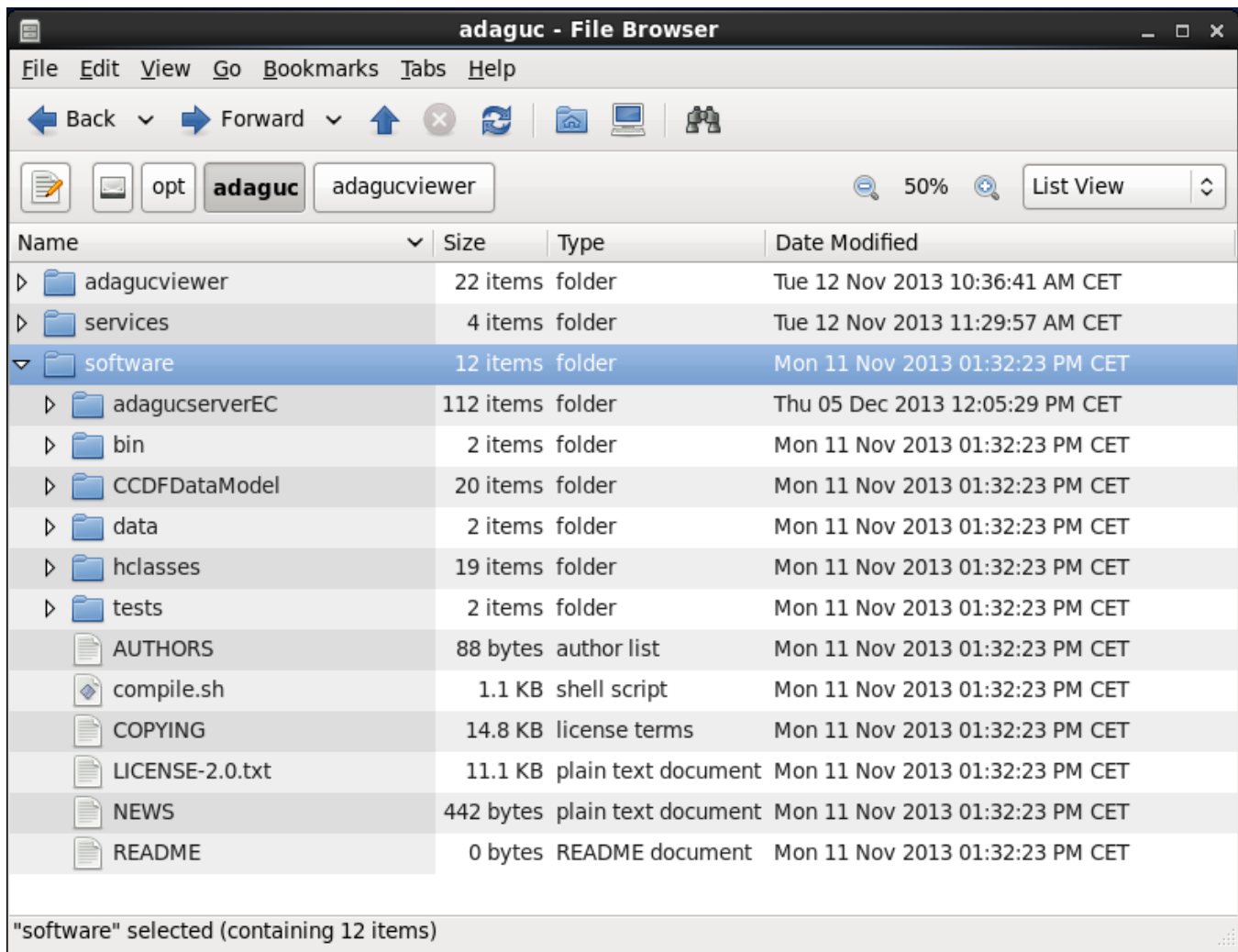
```
[mylocalhost]# yum install gcc gcc-c++ mercurial libpng-devel zlib-devel libxml2-devel gd-devel netcdf-devel hdf5-devel proj-devel  
postgresql-devel udunits2-devel gdal-devel cairo-devel httpd postgresql-server sqlite3-devel
```

Note: somehow this install was not completed successfully because later on the compiler complained about missing dependencies. I just re-executed this yum install (with all packages listed) and after that all worked fine.

Compile ADAGUC server

- Setup a directory structure for ADAGUC in /opt:

/opt/adaguc/adagucviewer	stores ADAGUC client/viewer related files and components
/opt/adaguc/services	stores ADAGUC server runtime related configuration, logs and optionally data
/opt/adaguc/software	tree with ADAGUC server source files; should not be present on a production environment



CentOS has a default Apache installation and this will be used for ADAGUC. Hence ownership of all files and directories involved is assigned explicitly to OS-user *apache* and OS-group *apache* (*apache:apache*).

- get ADAGUC server source components from KNMI's Mercurial repository:

```
[mylocalhost]# hg clone http://dev.knmi.nl/hg/adagucserver /opt/adaguc/software
```

- start compile by executing *compile.sh* from */opt/adaguc/software* directory. It could be necessary to set the executable flag first:

```
[mylocalhost]# chmod +x compile.sh
```

```
[mylocalhost]# ./compile.sh
```

The compilation results in a binary in the */opt/ADAGUC/software/adagucserverEC* directory.

Apache setup

ADAGUC server requires a Web Server. The ADAGUC server operates as a CGI-bin executable, triggered by the Web Server. CentOS has the Apache Web Server installed by default.

mod_php installation

The ADAGUC viewer uses some PHP scripts. For this the Apache *mod_php* is needed. This is not installed by default on CentOS. To install:

```
[mylocalhost]# yum install mod_php
```

This yum installation suffices; Apache on CentOS is set up in such a way that there is no need for explicit changes to configuration in order to enable this new module. An Apache restart is required afterwards.

Apache configuration

The Apache configuration needs some adaption. On CentOS the Apache configuration definition can be found in `/etc/httpd/conf/httpd.conf`. Adaptions /changes were made in the `VirtualHost`-section:

```
<VirtualHost *:80>
    ServerAdmin webmaster@localhost

    DocumentRoot /opt/adaguc/adagucviewer/
    <Directory />
        Options FollowSymLinks
        AllowOverride None
    </Directory>
    <Directory /opt/adaguc/adagucviewer/>
        Options Indexes FollowSymLinks MultiViews
        AllowOverride None
        Order allow,deny
        allow from all
    </Directory>

    ScriptAlias /cgi-bin/ /var/www/cgi-bin/
    <Directory "/var/www/cgi-bin">
        AllowOverride None
        Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
        Order allow,deny
        Allow from all
    </Directory>

    AddHandler cgi-script .cgi

    ErrorLog /var/log/httpd/adaguc_error.log

    # Possible values include: debug, info, notice, warn, error, crit,
    # alert, emerg.
    LogLevel warn

    CustomLog /var/log/httpd/adaguc_access.log combined
</VirtualHost>
```

Some explanation: ADAGUC server is a CGI-bin executable. The executable is called from a bash-script. SELinux constraints demand that CGI-scripts are executed from `/var/www/cgi-bin`.

For now the scripts are to be located in `/opt/adaguc/services/cgi-bin`; however this could be revised.

User and Group ownership of scripts is set to `apache:apache`

PostgreSQL setup

ADAGUC server requires a PostgreSQL RDBMS. PostgreSQL is not installed by default on CentOS. It should have been installed by the **Setup prerequisites** actions. I'm not sure why I (re)installed PostgreSQL here.

- Install PostgreSQL

```
[mylocalhost]# yum install postgresql.x86_64
```

- Initialise database
(PostgreSQL data directory : `/var/lib/pgsql/data`)

```
[mylocalhost]# service postgresql initdb
```

- If necessary change connection constraints in `/var/lib/pgsql/data/pg_hba.conf`.

```
# TYPE      DATABASE      USER      CIDR-ADDRESS      METHOD

# "local" is for Unix domain socket connections only
local      all          all                                trust
# IPv4 local connections:
host      all          all      127.0.0.1/32      trust
```

After making changes, perform:

```
[mylocalhost]# service postgresql reload
```

- Enable PostgreSQL service startup at boot time

```
[mylocalhost]# chkconfig postgresql on
```

- Create ADAGUC database

```
[mylocalhost]# su - postgres
```

Start psql client:

```
[mylocalhost]# psql
```

and execute these SQL statements:

```
create user adaguc password 'adaguc';
create database mydemo with owner=adaguc;
grant connect on database mydemo to adaguc;
grant all on database mydemo to adaguc;
```

In essence the ADAGUC server is now ready to use. <http://dev.knmi.nl/projects/adagucserver/wiki/Tutorials> shows by examples how to setup and configure WMS with ADAGUC.

SELinux

It was the initial idea to setup ADAGUC in a directory structure under /opt. Then here you will have the WMS configurations, the related CGI-script and optionally local data files close together. By means of file system privileges it is possible to allow non-root users to add, configure ADAGUC WMS. However SELinux does not allow for Apache to execute CGI-bin scripts that are outside of /var/www/cgi-bin. The use of symbolic links in /var/www/cgi-bin does not change this.

On top of that it is also not allowed to call executables from CGI-bin scripts that are located outside of this path.

These SELinux constraints are configurable. However this has not been changes yet. A shortcut was followed in slacking the SELinux constraints from *enforcing* to *permissive* in /etc/sysconfig/selinux:

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
#   enforcing - SELinux security policy is enforced.
#   permissive - SELinux prints warnings instead of enforcing.
#   disabled - No SELinux policy is loaded.
SELINUX=permissive
# SELINUXTYPE= can take one of these two values:
#   targeted - Targeted processes are protected,
#   mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

So for the ADAGUC setup a neat solution should be thought of which:

- fully enables SELinux security
- allow non-root users to setup ADAGUC WMS services

This is still to be done before a production environment for ADAGUC could be deployed.

ADAGUC Viewer

The ADAGUC Viewer is a mix of JavaScript and PHP coding. Setup of the viewer on a web server is done as follows:

- get ADAGUC viewer components from KNMI's Mercurial repository:

```
[mylocalhost]# hg clone http://dev.knmi.nl/hg/adagucviewer /opt/adaguc
```

- Apache's mod_php should be installed and functional. This is already described in de ADAGUC Server setup previously.
- Apache configuration
The DocumentRoot setting in de VirtualHost section in /etc/httpd/conf/httpd.conf is directed to /opt/adaguc/adagucviewer/

If this is setup then the ADAGUC viewer will load in the web browser when it is pointed to the web server.