NORDUGRID

The Hopi manual

Zsombor Nagy[*]

[*]zsombor@niif.hu

Contents

1   Introduction

The Hopi service is a HTTP(S) server supporting GET and PUT methods and capable of generating a HTML listing of files. It's main configuration parameter is the document root directory.

It also has a slave mode which is used in the Chelonia storage system[2] tied to the Shepherd service. In the slave mode: 1. there is no HTML listing generated; 2. the files are removed after download; 3. upload is only possible to an existing file, which will be removed after the upload. This does not make much sense unless some other service keeps a second hardlink of the uploaded files somewhere else, which is exactly what the Shepherd service of Chelonia does, thus implementing a ,,one-time transfer url" behaviour, where you can only upload files to prepared URLs and you can only do it once.

The Hopi service is running within the ARC's HED daemon, which creates a chain of message chain components (MCCs) which are used the communicate with the outer world. The Hopi service can be the only service in a message chain, in that case it does not need to have a Plexer (which is a component which routes the incoming messages to different services based on the path of the URL), and the URLs will not contain any extra path; or the Hopi could be after the Plexer, in that case all the URLs contains a fixed path (e.g. `https://localhost:60000/hopi/`). See the HED documentation for further reference.[3]

2   Clients

Because Hopi is a HTTP(S) server, any HTTP client can be used (for HTTPS maybe needs more configuration)

- web browser (needs importing the client credentials into the browser)
- curl/wget (needs specifying the client credentials as command line argument)
- arccp (the client credentials comes from the userconfig)

3   Installation

The Hopi service is part of the ARC middleware. Please refer to the ARC Installation instructions[4]. From the package repositories, you will need the `nordugrid-arc-hopi` package.

4   Configuration

Here is an example configuration of a secure Hopi with Plexer:

```
<?xml version="1.0"?>
<cfg:ArcConfig xmlns="http://www.nordugrid.org/schemas/loader/2009/08"
xmlns:cfg="http://www.nordugrid.org/schemas/arcconfig/2009/08"
xmlns:tcp="http://www.nordugrid.org/schemas/tcp/2009/08"
xmlns:tls="http://www.nordugrid.org/schemas/tls/2009/08">
    <cfg:Server>
        <cfg:PidFile>/tmp/arched.pid</cfg:PidFile>
        <cfg:Logger>
            <cfg:File>/tmp/arched.log</cfg:File>
            <cfg:Level>ERROR</cfg:Level>
        </cfg:Logger>
    </cfg:Server>
    <ModuleManager>
```

---

[2]http://www.nordugrid.org/chelonia/
[3]http://www.nordugrid.org/documents/ARCHED_article.pdf
[4]http://www.nordugrid.org/documents/arc-server-install.html#soft

```
                <Path>/usr/lib64/arc</Path>
        </ModuleManager>
        <Plugins>
                <Name>mcctls</Name>
                <Name>mcchttp</Name>
                <Name>mccsoap</Name>
                <Name>mcctcp</Name>
        </Plugins>
        <Chain>
                <Component name="tcp.service" id="tcp">
                        <next id="tls"/>
                        <tcp:Listen>
                                <tcp:Interface>0.0.0.0</tcp:Interface>
                                <tcp:Port>50000</tcp:Port>
                                <tcp:Version>4</tcp:Version>
                        </tcp:Listen>
                </Component>
                <Component name="tls.service" id="tls">
                        <next id="http"/>
                        <tls:KeyPath>/etc/grid-security/hostkey.pem</tls:KeyPath>
                        <tls:CertificatePath>/etc/grid-security/hostcert.pem</tls:CertificatePath>
                        <tls:CACertificatesDir>/etc/grid-security/certificates</tls:CACertificatesDir>
                </Component>
                <Component name="http.service" id="http">
                        <next id="soap">POST</next>
                        <next id="plexer">GET</next>
                        <next id="plexer">PUT</next>
                </Component>
                <Component name="soap.service" id="soap">
                        <next id="plexer"/>
                </Component>
                <Plexer name="plexer.service" id="plexer">
                        <next id="hopi">^/hopi/</next>
                </Plexer>
                <Service name="hopi" id="hopi">
                        <DocumentRoot>/tmp/hopi</DocumentRoot>
                        <SlaveMode>0</SlaveMode>
                </Service>
        </Chain>
</cfg:ArcConfig>
```

There is no rule where to put the configuration XML, let's name it now: `/etc/arched.xml`. Then, on red-hat based distributions, we have to specify this path in the `/etc/sysconfig/arched` file, like this:

```
ARC_CONFIG=/etc/arched.xml
```

On debian-based distributions, you should put this line into `/etc/default/arched`.

We also need to have host certificates installed into e.g. `/etc/grid-security`. If the certificates or the installed libraries are at different paths, we should modify the config file accordingly.

We also have to create a directory from which Hopi should serve the files, e.g. `/tmp/hopi`.

5 Running the service

Now we can start the service:

```
$ sudo /etc/init.d/arched start
```

And now we can use any http client (which has access to our X.509 certificate) to fetch the root URL of our Hopi service (`https://localhost:50000/hopi/` Note the trailing „/" and compare it to the path in the Plexer part of the config.)