# **ARM Nodes**

#### (1) This is an experimental feature!

Some issues (e.g. jobs waiting for long periods of time in idle state, not being able to submit new jobs, etc...) may occur, please contact us at use r-support@lists.cnaf.infn.it to report this kind of problems.

### AArch64 is not Intel

Any executable compiled for the Intel CPU architecture (the typical CPU architecture supported at CNAF) won't work on ui-arm or ARM-based worker nodes.

In the HTCondor pool at CNAF are currently installed 4 ARM worker-nodes, they are configured to accept jobs both via local (on sn-02) or Grid submission (e.g. ce02-htc).

# **UI-ARM**

## Login to ui-arm

A new user-interface has been installed to let users test and submit jobs developed for ARM CPU architecture. Please find below all the instruction to log into **ui-arm.cr.cnaf.infn.it**.

 ui-arm should be accessed by users only from their usual user-interface to prevent any issues (for the sake of this tutorial ui-tier1 will be the one of choice). Do not access ui-arm from the bastion host even though it seems to be working. As a first step, a user should have generated a pair of ssh-keys on the usual user interface (e.g. ui-tier1.cr.cnaf.infn.it).

#### SSH-Key Generation

In case no ssh-keys are present into ~/.ssh/ directory, a new one can be generate executing the ssh-keygen command

```
[apascolini@ui-tier1 ~]$ ls -la ~/.ssh
total 17
drwx----- 2 apascolini cms 512 Feb 1 15:33 .
drwxr-x--- 19 apascolini cms 8192 Feb 1 15:33 ..
[apascolini@ui-tier1 ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/CMS/apascolini/.ssh/id_rsa): <press_enter>
Enter passphrase (empty for no passphrase): <insert_passphrase/press_enter>
Enter same passphrase again: <insert_passphrase/press_enter>
Your identification has been saved in /home/CMS/apascolini/.ssh/id_rsa.
Your public key has been saved in /home/CMS/apascolini/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:fWCSzY0p7WlafvPMqGNtQORJXRV/9Jb6pm8LAnqGNK4 apascolini@ui-tier1.cr.cnaf.infn.it
The key's randomart image is:
+---[RSA 2048]----+
           0. 0*
        = B ...+
        + 0 = .=
         * + ...
       oS.o *.
      0 + .* +.
       + 0..=.*0
      . o ..++B
     Е
            . 00=
+----[SHA256]----+
[apascolini@ui-tier1 ~]$ ls -la ~/.ssh
total 17
drwx----- 2 apascolini cms 512 Feb 1 15:34 .
drwxr-x--- 19 apascolini cms 8192 Feb 1 15:33 ..
-rw----- 1 apascolini cms 1679 Feb 1 15:34 id_rsa
-rw----- 1 apascolini cms 417 Feb 1 15:34 id_rsa.pub
```

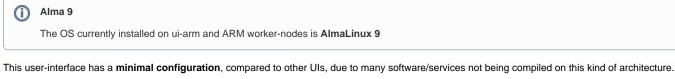
 Once both the public and private keys are present in the ~/.ssh directory, the user can enable ssh access to ui-arm via ssh-key, by appending the content of the ~/.ssh/id\_rsa.pub file into the ~/.ssh/authorized\_keys file

```
[apascolini@ui-tier1 ~]$ cd ~/.ssh
[apascolini@ui-tier1 .ssh]$ cat id_rsa.pub >> authorized_keys
[apascolini@ui-tier1 .ssh]$ ls -la
total 42
drwx----- 2 apascolini cms 512 Feb 1 15:42 .
drwxr-x--- 19 apascolini cms 8192 Feb 1 15:33 ..
-rw------ 1 apascolini cms 417 Feb 1 15:41 authorized_keys
-rw------ 1 apascolini cms 1679 Feb 1 15:34 id_rsa
-rw------ 1 apascolini cms 417 Feb 1 15:34 id_rsa.pub
```

3. After these steps the user is now able to access ui-arm via ssh with the same user used to log into the previous user-interface

```
[apascolini@ui-tier1 ~]$ ssh ui-arm
Warning: Permanently added 'ui-arm,131.154.192.208' (ECDSA) to the list of known hosts.
     \setminus | |
                \setminus |
                    | | / 
| || \| |_
                                 | | / _ 
                 ||
 _| | |\
                               | | \rangle | /
Welcome to ui-arm.cr.cnaf.infn.it - AlmaLinux 9.3
This node is under the control of Puppet 7.27.0
Kernel: Linux 5.14.0
_____
Last login: Wed Jan 31 16:50:35 2024 from 131.154.193.196
[apascolini@ui-arm ~]$
```

# Once on ui-arm



Please find below all the features supported on the UI.

Packages	Filesystem	Services
• condor	• /opt/exp_software	• cvmfs

In case a user needs other packages (compiled for aarch64) to be available on ui-arm, can contact the user-support team.



# Submitting on ARM worker-nodes

## Local Jobs

To submit local jobs to ARM wn no additional configuration is needed. A user can submit directly on **sn-02** as described in HTCondor jobs - Submit local job.

## **Grid Jobs**

In case of Grid jobs the cluster is configured to redirect to ARM nodes all jobs containing the attribute **WantARM = True**. Please find below few examples of grid jobs submitted both via VOMS proxies or SciTokens.

```
VOMS proxies submit
```

```
# Unix submit description file
# sleep.sub -- simple sleep job
use_x509userproxy = true
+owner = undefined
+WantARM = True
executable = sleep.sh
log = sleep.log
output = outfile.txt
error = errors.txt
should_transfer_files = Yes
when_to_transfer_output = ON_EXIT
queue
```

#### SciToken submit

```
# Unix submit description file
# sleep.sub -- simple sleep job
scitokens_file = $ENV(HOME)/token
+owner = undefined
+WantARM = True
executable = sleep.sh
log = sleep.log
output = outfile.txt
error = errors.txt
should_transfer_files = Yes
when_to_transfer_output = ON_EXIT
queue
```

Any further information on how to submit grid jobs can be found in HTCondor jobs - Submit grid job.