Contents

Embedded Development Environement 3.x.x - Devel

EDE Team

July 27, 2022

Chapter 1

EDE Server

1.1 EDE Server Configuration

The *EDE Server Configuration* is stored in *aos-ne-os* repository and is described in file:
aos-ne-os/ede-server-scripts/README.txt.
The document consist of:

- hardware configuration
- EDE server install procedure
- Use server configuration scripts
- Manual steps
- Setup EDE development environment

1.1.1 Metaport - auto apply patches

This mechanism is available in repository in *aos-ne-os/metaport/* directory.
Example for targets, package *net-libs/libssh*

```
ls aos-ne-os/metaport/targ/net-libs/libssh/
    libssh-0.8.5.patch
    libssh-0.8.6.patch
    libssh-0.9.0.nop
```

Description:
The files *libssh-0.8.5.patch, libssh-0.8.6.patch* consist of patches for *ebuilds*, but file *libssh-0.9.0.nop* is ***empty*** and extension ****.nop*** tell for patching mechanism: ***stop try apply patches*** from previous versions since *libssh-0.9.0* version.
Patching mechanism works on ***EDE servers*** and it's managed by *Jenkins Project* **ede-portage-update**.
Patch example:

```
--- libssh-0.8.6.ebuild 2019-02-07 14:39:35.000000000 +0100
+++ libssh-0.8.6-1.ebuild       2019-03-06 11:41:13.860830293 +0100
@@ -70,6 +70,7 @@
                -DWITH_GCRYPT="$(usex gcrypt)"
                -DWITH_GSSAPI="$(usex gssapi)"
                -DWITH_MBEDTLS="$(usex mbedtls)"
+               -DTHREADS_PTHREAD_ARG=yes
                -DWITH_NACL=no
                -DWITH_PCAP="$(usex pcap)"
                -DWITH_SERVER="$(usex server)"
```

Chapter 2

EDE build components

2.1 EDE Jenkins Projects

List of the Jenkins Projects

- ede-base_layout_update_or_create-new
- ede-crossdev_update_or_create-new
- ede-embedded_targets_update_or_create

  small changes is possible to apply by one *ede-targ-selected-arch* Jenkins job on all architectures.

  included tests on *EDE* server (ede-base_targets_checker, ede-base_targets_listcheck, ...)
- ede-native_targets_update_or_create-new [1]
- ede-products
- ede-3-generate_documentation
- ede-portage-update *(metaport)*
- ede-do-release

Jenkins: (global link to EDE projects)
https://muc-jenkins-ne.rd.advaoptical.com/job/ede-release-automatization
Repository:
aos-ne-os
git link:
ssh://user@muc-gerrit.rd.advaoptical.com:29418/aos-ne-os

2.1.1 Jenkins Job flow

All required steps:

- create snapshot and working branch
- run build scripts on working branch
- merge changes/results
- remove working branch

Flow in details: *(example: ede-base_layout_update_or_create-new)*

- create snapshot and working branch

  In Jenkins project:

[1] (disabled since *ede-3.5.0*)

```
#!/bin/bash -xe

if ${REBUILD_SPECIFIED_BUILD}; then exit 0; fi

if ${EDE_SNAPSHOT_BRANCH_PREPARE}; then
  echo "JOB_NAME    : ${JOB_NAME}"
  echo "Build number: ${BUILD_NUMBER}"
  echo "EDE Maj.Min : ${EDE_MAJ_MIN}"
  ssh bld@${IP_EDE_SERVER} "sudo /home/bld/gitRepos/aos-ne-os/
      other-tools-and-scripts/proxy_script.sh
      \"${WORKSPACE}/aos-ne-os/other-tools-and-scripts/
      jenkins_scripts/ede-create-new--ede-base_layout_create_init.sh
      ${JOB_NAME}  ${EDE_MAJ_MIN}  ${BUILD_NUMBER}\" "
fi
```

[2]

- run build scripts on working branch

  In Jenkins project:

  ```
  #!/bin/bash -xe
  if ${EDE_BASE_CREATE_REBUILD}; then
  edeRebuildAll="yes"
  else
  edeRebuildAll="no"
  fi

  if ${REBUILD_SPECIFIED_BUILD}; then
  BUILD_NUMBER=${BUILD_NUMBER_REBUILD}
  fi

  if ${EDE_BASE_CREATE}; then
    echo "JOB_NAME    : ${JOB_NAME}"
    echo "Build number: ${BUILD_NUMBER}"
    echo "EDE Maj.Min : ${EDE_MAJ_MIN}"
    ssh bld@${IP_EDE_SERVER} "sudo /home/bld/gitRepos/aos-ne-os/
        other-tools-and-scripts/proxy_script.sh  \"${WORKSPACE}/
        aos-ne-os/other-tools-and-scripts/jenkins_scripts/
        ede-create-new--ede-base_layout_create.sh
        ${JOB_NAME}  ${EDE_MAJ_MIN}  ${BUILD_NUMBER} ${edeRebuildAll}\" "
  fi
  ```

- merge changes/results

  In Jenkins project:

  ```
  #!/bin/bash -xe

  if ${REBUILD_SPECIFIED_BUILD}; then
  BUILD_NUMBER=${BUILD_NUMBER_REBUILD}
  fi

  if ${EDE_MERGE_CHANGES}; then
    echo "JOB_NAME    : ${JOB_NAME}"
    echo "Build number: ${BUILD_NUMBER}"
    echo "EDE Maj.Min : ${EDE_MAJ_MIN}"
    ssh bld@${IP_EDE_SERVER} "sudo /home/bld/gitRepos/aos-ne-os/
        other-tools-and-scripts/proxy_script.sh  \"${WORKSPACE}/
        aos-ne-os/other-tools-and-scripts/jenkins_scripts/
  ```

[2] All build scripts are called by proxy script: aos-ne-os/other-tools-and-scripts/proxy_script.sh

```
      ede-create-new--ede-base_layout_create_merge.sh
      ${JOB_NAME}  ${EDE_MAJ_MIN}  ${BUILD_NUMBER}\" "
fi
```

- remove working branch

  In Jenkins project:

  ```
  #!/bin/bash -xe

  if ${REBUILD_SPECIFIED_BUILD}; then
  BUILD_NUMBER=${BUILD_NUMBER_REBUILD}
  fi

  if ${EDE_REMOVE_BRANCH}; then
    echo "JOB_NAME    : ${JOB_NAME}"
    echo "Build number: ${BUILD_NUMBER}"
    echo "EDE Maj.Min : ${EDE_MAJ_MIN}"
    ssh bld@${IP_EDE_SERVER} "sudo /home/bld/gitRepos/aos-ne-os/
        other-tools-and-scripts/proxy_script.sh  \"${WORKSPACE}/aos-ne-os/
        other-tools-and-scripts/jenkins_scripts/
        ede-create-new--ede-base_layout_create_cleanbr.sh
        ${JOB_NAME}  ${EDE_MAJ_MIN}  ${BUILD_NUMBER}\" "
  fi
  ```

2.1.2 Jenkins project: ede-base_layout_update_or_create-new

Jenkins Projects configurations are available in *aos-ne-os* repository in path:
aos-ne-os/other-tools-and-scripts/JenkinsConfigXml/. The naming convention is,
(for example project ***ede-base_layout_update_or_create-new_config***),
ede-base_layout_update_or_create-new_config.xml.

Jenkins Projects scripts are stored in *aos-ne-os/other-tools-and-scripts/jenkins_scripts/.*
For project *ede-base_layout_update_or_create-new* (build) is
aos-ne-os/other-tools-and-scripts/jenkins_scripts/ede-create-new–ede-base_layout_create.sh

2.1.3 Jenkins project: ede-crossdev_update_or_create-new

Project configuration:

- *aos-ne-os/other-tools-and-scripts/JenkinsConfigXml/ede-crossdev_update_or_create-new_config.xml*

Project scripts:

- *aos-ne-os/other-tools-and-scripts/jenkins_scripts/ede-create-new–ede-base_layout_create_crossdev.sh*

2.1.4 Jenkins project: ede-embedded_targets_update_or_create

Project configuration:

- *aos-ne-os/other-tools-and-scripts/JenkinsConfigXml/ede-embedded_targets_update_or_create_config.xml*

  embedded targets*_config.xml

  ede-base_targets_checker

  ede-base_targets_listcheck

Project scripts:

- *aos-ne-os/other-tools-and-scripts/jenkins_scripts/ede-embedded_targets_update_or_create–create_target.sh*

Example: in Jenkins project*, parameters to target architecture ede-targ-arch-x86_64-vm-linux-gnu-new:*

```
#!/bin/bash -xe

if ${EDE_TARG_CREATE_REBUILD}; then
edeRebuildAll="yes"
```

```
else
edeRebuildAll="no"
fi

if ${REBUILD_SPECIFIED_BUILD}; then
BUILD_NUMBER=${BUILD_NUMBER_REBUILD}
fi

if ${EDE_TARG_CREATE_VM64}; then
  echo "JOB_NAME    : ${JOB_NAME}"
  echo "Build number: ${BUILD_NUMBER}"
  echo "EDE Maj.Min : ${EDE_MAJ_MIN}"
  TARG_ARCH="x86_64-vm-linux-gnu"
  echo "TARG_ARCH   : ${TARG_ARCH}"
  ssh bld@${IP_EDE_SERVER}
      "sudo /home/bld/gitRepos/aos-ne-os/other-tools-and-scripts/proxy_script.sh
      "${WORKSPACE}/aos-ne-os/other-tools-and-scripts/jenkins_scripts/
      ede-embedded_targets_update_or_create--create_target.sh
      ${JOB_NAME}  ${EDE_MAJ_MIN}  ${BUILD_NUMBER} ${TARG_ARCH} ${edeRebuildAll}\" "
fi
```

2.1.5 Jenkins project: ede-native_targets_update_or_create-new

Jenkins project: **ede-native_targets_update_or_create-new**
Project configuration:

- *aos-ne-os/other-tools-and-scripts/JenkinsConfigXml/ede-native_targets_update_or_create-new_config.xml*
  ede-targ-arch-x86_64-nfv-linux-gnu-new_config.xml

Project scripts:

- *aos-ne-os/other-tools-and-scripts/jenkins_scripts/ede-native_targets_create–ede-targ-arch-x86_64-nfv-linux-gnu-new.sh*

2.1.6 Jenkins project: ede-products

Jenkins project: *ede-products*
Project configuration:

- *aos-ne-os/other-tools-and-scripts/JenkinsConfigXml/ede-products_config.xml*
  ede-prod-vm64_config.xml

Project scripts:

- *aos-ne-os/other-tools-and-scripts/jenkins_scripts/ede-products–ede-prod-vm64.sh*

2.1.7 Jenkins project: ede-3-generate_documentation

Jenkins project: *ede-3-generate_documentation*
Project configuration:

- *aos-ne-os/other-tools-and-scripts/JenkinsConfigXml/ede-3-generate_documentation_config.xml*

Project scripts:

- *aos-ne-os/other-tools-and-scripts/jenkins_scripts/ede-3-generate_documentation.sh*

2.1.8 Jenkins project: ede-portage-update

Jenkins project: *ede-portage-update*
Project configuration:

- *aos-ne-os/other-tools-and-scripts/JenkinsConfigXml/ede-portage-update_config.xml*

Project scripts:

- *aos-ne-os/other-tools-and-scripts/jenkins_scripts/ede-update-current–ede-portage-update.sh*

2.1.9 Jenkins project: ede-do-release

Jenkins project: *ede-do-release*
Project configuration:

- *aos-ne-os/other-tools-and-scripts/JenkinsConfigXml/ede-do-release_config.xml*

Project scripts:

- *aos-ne-os/other-tools-and-scripts/jenkins_scripts/ede-release.py*

Chapter 3

EDE Development (working on EDE server)

3.1 Branches and snapshots - dede command

Branches and snapshots are stored in */mnt/dede/devel* directory.

dede commands: [1]

- create a new snapshot (from 3.0 branch)

  ```
  dede sn 3.0
  ```

- create a new branch (from 3.0 branch)

  ```
  dede br 3.0a 3.0
  ```

- remove branch

  ```
  dede rmbr 3.0a
  ```

- pull branch

  ```
  dede brup! 3.0a
  ```

- merge branch (to 3.0 in this case)

  ```
  dede brmrg! 3.0a
  ```

- mount/unmount branch (3.0a in this case)

  ```
  dede mount  3.0a/head
  dede umount 3.0a/head
  ```

- mount all branches and snapshots

  ```
    dede mount
  or
    /etc/rc.local
  ```

- show all branches and snapshots

  ```
  dede ls
  ```

[1] initial branch (in this case 3.0) must be created earlier. Information about it is available in aos-ne-os repository in /aos-ne-os/ede-server-scripts/README.txt in section Setup EDE development environment.

3.1.1 rede command

The *rede* command is equivalent as *reho* command, but it is dedicated to use on *EDE server*.
Examples:

```
  rede 3.0a
or
  rede 3.0a <command>
  ex.:
  rede 3.0a emerge --info

with crossdev layer:
  rede 3.0a:crossdev
```

3.1.2 nede commands

nede build commands examples:

```
nede 3.0a <job1>,<job2>,...

ex.:
nede 3.0a sync,reho,native
```

nede commands:

```
clear
dirs     - creates dirs. It's prefer to use after clear
sync     - syncing files from repository: $HOME/gitRepos/aos-ne-os/lrs
reho     - compile reho tool
stage    - extract  /mnt/dede/stages/latest.tar.xz
native   - compile EDE base (tools)
crossdev - compile crossdevs for all architectures
```

[2]

3.2 EDE important build configuration files

3.2.1 EDE base tools

- base packages list: *aos-ne-os/dede/bin/int/natives*
- packages installed by *PIP*: *aos-ne-os/dede/bin/int/natives_pip*
- packages in the same version in *EDE Tools* and embedded targets: *aos-ne-os/dede/bin/int/pinned*

3.2.2 Embedded targets

[3]

- architectures table: *aos-ne-os/dede/bin/archtab*
- canon packages list: *aos-ne-os/dede/bin/int/crosspacks*
- packages for specified architectures: *aos-ne-os/dede/bin/int/crosspacks.spec*
- remove packages (if exists): *aos-ne-os/dede/bin/int/crosspacks.remove*
- packages in the same version in *EDE Tools* and embedded targets: *aos-ne-os/dede/bin/int/pinned*

[2] Typical usage is available in EDE Build scripts. (stored in aos-ne-os repository in: aos-ne-os/other-tools-and-scripts/jenkins_scripts/

[3] The native target is disabled since ede-3.5.0

3.2.3 Embedded product

vm64

- kernel configuration files: *aos-ne-os/lrs/prod-devel/targ/prod-devel/vm64/kernel_configs*
  Current configuration:
  aos-ne-os/lrs/prod-devel/targ/prod-devel/vm64/kernel_configs/v4.18-aufs/kernel_configuration-4.18
- product configuration examples:

  disk image: *aos-ne-os/lrs/prod-devel/targ/prod-devel/vm64/conf-vm64*

  rootfs: *aos-ne-os/lrs/prod-devel/targ/prod-devel/vm64/conf-vm64-rootfs*
- HowTo: *aos-ne-os/lrs/prod-devel/targ/prod-devel/vm64/README.txt*

3.2.4 Native target

[4]

- configuration files:

  for creating target: *aos-ne-os/other-tools-and-scripts/ede-update_native_targets/create_new–x86_64-nfv-linux-gnu_step1.txt*

  for updates: *aos-ne-os/other-tools-and-scripts/ede-update_native_targets/create_new–x86_64-nfv-linux-gnu.txt*
- additional packages list: *aos-ne-os/other-tools-and-scripts/ede-update_native_targets/create_new–x86_64-nfv-linux-gnu_productSpecificPackagesList.txt*
- to remove packages list (before install) which are available in embedded canon packages list, but are not compatible with native target: *aos-ne-os/other-tools-and-scripts/ede-update_native_targets/create_new–x86_64-nfv-linux-gnu_removePackagesList.txt*

3.2.5 Documentation

Documentation compilation files

- ***ede-2.x.x*** - aos-ne-os/Documentation/ede-2.x.x/user_manual/user_manual.tex
  compilation:

  ```
  cd aos-ne-os/Documentation/ede-2.x.x/user_manual/
  pdflatex user_manual.tex (two times)
  ```
- ***ede-3.x.x*** - aos-ne-os/Documentation/ede-3.x.x/manual.tex
  compilation:

  ```
  cd aos-ne-os/Documentation/ede-3.x.x/
  pdflatex manual.tex
  ```
- ***ede-3.x.x devel*** - aos-ne-os/Documentation/ede-3.x.x/ede_devel/manual_dev.tex
  compilation:

  ```
  cd aos-ne-os/Documentation/ede-3.x.x/ede_devel/
  pdflatex manual_dev.tex
  ```

[4] (disabled since *ede-3.5.0*)