Yocto Project® Virtualization (and Security)

Christopher Clark, OpenXT Project
Tim Orling, Intel Corporation
François Dugast, Intel Corporation

Yocto Project DevDay *Virtual*, North America, 2020

Content and Continuous Integration

https://gitlab.com/moto-timo/yp-dev-day_virtualization.git

KVM Hypervisor

The Linux Kernel Hypervisor

https://www.linux-kvm.org/page/Main_Page

Kernel-based Virtual Machine (KVM)

”The Kernel-based virtual Machine (KVM) is a full virtualization hypervisor for Linux. The work of the **KVM hypervisor** is handled by the **Linux kernel**. Each guest in KVM runs as a process and can be managed by Linux tools such as **top** and **kill**.

KVM isn’t a complete virtualization solution. It depends on both the **libvirt** tools for management and the open source processor emulator **QEMU** for hardware emulation. Therefore, you will need those **installed** as well.”

-- Stephen Figgins, Robert Love, Arnold Robbins, Ellen Siever, Linux in a Nutshell, 6th ed., O’Reilly Media, Inc, 2009.

Let’s do this!

packagegroup-kvm-host.bb

```
SUMMARY = "Provides a set of tools for hosting KVM guests."

inherit packagegroup

RDEPENDS_${PN} = "\
       packagegroup-core-boot \
       qemu \
       libvirt \
       libvirt-libvirtd \
       libvirt-virsh \
       "
```

kvm-binary-image-vessel-package.bbclass
(Avoiding the overloaded meaning of “container”)

```
SUMMARY = "Package for ${IMAGE_NAME}"
# This license statement is a lie. Ideally set it to something more appropriate.
LICENSE = "CLOSED"

INHIBIT_DEFAULT_DEPS = "1"

inherit bin_package

# Where to install the image
vesseldir ?= "${localstatedir}/lib/libvirt/images"

do_install[depends] += "libvirt:do_install"

do_install () {
        install -d ${D}${vesseldir}
        install ${S}/../${VESSEL_PAYLOAD_NAME} ${D}${vesseldir}/${VESSEL_PAYLOAD_NAME}
}
```

Based on https://github.com/intel/meta-acrn/blob/master/classes/container-package.bbclass by Ross Burton

ubuntu-kvm-image-package.bb

```
SUMMARY = "Ubuntu cloud kvm image"
# Probably this should be Canonical IPRights?
LICENSE="CLOSED"

inherit kvm-binary-guest-package

# precise, xenial and bionic do not have kvm images
# eoan, focal and groovy do
UBUNTU_BASE_URL ??= "https://cloud-images.ubuntu.com"
UBUNTU_RELEASE ??="focal"
UBUNTU_IMAGE_ARCH ??="amd64"
UBUNTU_IMAGE_NAME ?= "${UBUNTU_RELEASE}-server-cloudimg-${UBUNTU_IMAGE_ARCH}-disk-
kvm.img"
UBUNTU_IMAGE_DATE ?= "current"
VESSEL_PAYLOAD_NAME = "${UBUNTU_IMAGE_NAME}”

[…]
```

ubuntu-kvm-image-package.bb (cont'd)

```
SRC_URI =
"${UBUNTU_BASE_URL}/${UBUNTU_RELEASE}/${UBUNTU_IMAGE_DATE}/${UBUNTU_IMAGE_NAME}"
SHA256SUMS_URI = "${UBUNTU_BASE_URL}/${UBUNTU_RELEASE}/${UBUNTU_IMAGE_DATE}/SHA256SUMS"

# See  http://www.burtonini.com/blog/2017/06/13/dynamic-source-checksums
do_fetch[prefuncs]  += "fetch_checksums"

python fetch_checksums() {
    import re
    import urllib

    match = "*{}".format(d.getVar("UBUNTU_IMAGE_NAME"))
    for line in urllib.request.urlopen(d.getVar("SHA256SUMS_URI")):
        (sha256, filename) = line.decode("ascii").strip().split()
        if filename == match:
            d.setVarFlag("SRC_URI", "sha256sum", sha256)
            return
    bb.error("Could not find remote checksum for %s" %  filename)
}
```

Future work

- Secure Boot
- virsh (use a template to create XML)
- Launch script to automatically boot the guest
- Insert ssh keys into guest