[common]

welcome\_text = Welcome to System Certification!

This application will gather information from your system. Then you will be asked manual tests to confirm that the system is working properly. Finally, you will be asked for the Secure ID of the computer to submit the information to the certification.canonical.com database.

Please be sure you have edited /etc/xdg/canonical-certification.conf as necessary to set your Secure ID, and network and virtualization test parameters.

To learn how to create or locate the Secure ID, please see here:

https://certification.canonical.com/home/ubuntu-certified

# If you have a secure\_id, uncomment the next two lines and set your

# 15- or 18-character secure\_id here for automatic submission. If

# unsure, it's safe to leave this commented out.

[transport:c3]

secure\_id =

[environment]

# For disk testing:

#

# Set override minimum read performance values for NVME and MDADM devices in

# the disk\_read\_performance\_test.sh test script.

#

DISK\_NVME\_READ\_PERF = 400

DISK\_MDADM\_READ\_PERF = 150

# For virtualization testing:

#

# The KVM\_IMAGE and LXD\_TEMPLATES variables are now used in the lxd

# virtual machine test.

#

# The LXD\_TEMPLATES is the tarball that contains the LXD templates and YAML

# necessary for defining containers and virtual machines. This variable

# should be set to the same value that is used for lxd container testing.

#

# KVM\_IMAGE the location to find the cloud image to use for testing KVM. This

# will allow you to specify a variety of location options depending on your

# needs.

# Options available:

# - http://someotherserver.net/imagedir/ubuntu/mycloudimg.img

# - This will tell the script to download the file mycloudimg.img from the

# remote server.

# - file:///local/path/to/cloud.img

# - This will tell the script to look at the path specified on the local

# filesystem for a copy of the cloud image to use.

# Example:

# KVM\_IMAGE = http://cloud-images.ubuntu.com/daily/server/daily/server/{release}

#

# Example:

# KVM\_IMAGE = file:///home/ubuntu/jammy-server-cloudimg-amd64.img

#

# Note that a MAAS server set up via the MANIACS guide may optionally house

# copies of the KVM images under http://{ip-address}/cloud/.

#

# UNCOMMENT THE FOLLOWING LINE and set the proper

# parameters as necessary:

# KVM\_IMAGE = /path/to/cloudimage-filename.img

# For LXD testing:

#

# LXD\_ROOTFS is the tarall that contains the pre-created root filesystem for

# LXD containers.

# LXD\_TEMPLATES is the tarball that contains the LXD templates and YAML

# necessary for defining containers

KVM\_IMAGE = file:///home/ubuntu/jammy-server-cloudimg-amd64.img

UVT\_IMAGE\_OR\_SOURCE = file:///home/ubuntu/jammy-server-cloudimg-amd64.img

LXD\_ROOTFS = file:///home/ubuntu/jammy-server-cloudimg-amd64.squashfs

LXD\_TEMPLATE = <file:///home/ubuntu/jammy-server-cloudimg-amd64-lxd.tar.xz>

KVM\_IMAGE = http://192.168.1.1/cloud/jammy-server-cloudimg-amd64.img

UVT\_IMAGE\_OR\_SOURCE = http://192.168.1.1/cloud/jammy-server-cloudimg-amd64.img

LXD\_ROOTFS = http://192.168.1.1/cloud/jammy-server-cloudimg-amd64.squashfs

LXD\_TEMPLATE = http://192.168.1.1/cloud/jammy-server-cloudimg-amd64-lxd.tar.xz

# Note: You must provide a full URL to the files for LXD even if they're stored

# locally. The following examples demonstrate proper URLs for this section:

# - http://someotherserver.net/imagedir/ubuntu/server-cloudimg-amd64.squashfs

# - This will tell the script to download the file server-cloudimg-amd64.squashfs

# from the remote server.

# - file:///local/path/to/server-cloudimg-amd64-lxd.tar.xz

# - This will tell the script to look at the path specified on the local

# filesystem for server-cloudimg-amd64-lxd.tar.xz to use.

#

# Note that a MAAS server set up via the MANIACS guide may optionally house

# copies of the LXD images under http://{ip-address}/cloud/.

#

# UNCOMMENT BOTH OF THE FOLLOWING 2 LINES nad set the proper URLs as necessary

# OR leave them commented out and the LXD test will obtain the needed images

# from cloud-images.ubuntu.com.

# The STRESS\_NG\_MIN\_SWAP\_SIZE environment variable sets the minimum swap

# space, in gibibytes (GiB), that the memory/memory\_stress\_ng test requires.

# The STRESS\_NG\_MIN\_SWAP\_SIZE environment variable sets the minimum swap

# space, in gibibytes (GiB), that the memory/memory\_stress\_ng test requires.

# If less than the specified swap space is available, the script will

# attempt to create a swap file of the required size. If that attempt fails,

# the memory stress test will fail.

STRESS\_NG\_MIN\_SWAP\_SIZE = 16

# For network testing:

#

# TEST\_TARGET\_IPERF is the IP/Hostname of the iperf server for network testing.

#

# UNCOMMENT THE FOLLOWING LINE and set the IP/Hostname of your iperf target

# system as necessary

TEST\_TARGET\_IPERF = 192.168.1.1 192.168.1.101 192.168.1.102 192.168.1.103