

Internet of Things Certified Hardware Coverage for Ubuntu Core 24 / Ubuntu 24.04

Contents

| | | |
|----------|--------------------------------------|-----------|
| 1 | Introduction | 2 |
| 2 | client-cert-iot-ubuntucore-24 | 3 |
| 2.1 | Blocking | 3 |
| 2.2 | Non-blocking | 17 |
| 2.3 | Manifest Entries | 17 |
| 3 | client-cert-iot-server-24-04 | 19 |
| 3.1 | Blocking | 19 |
| 3.2 | Non-blocking | 35 |
| 3.3 | Manifest Entries | 35 |
| 4 | client-cert-iot-desktop-24-04 | 37 |
| 4.1 | Blocking | 37 |
| 4.2 | Non-blocking | 54 |
| 4.3 | Manifest Entries | 54 |
| 5 | Appendix A. FWTS tests | 56 |

1. Introduction

This document lists the coverage for certification of Internet of Things (IoT) devices with Ubuntu images. IoT devices can be certified with the following image types:

- Ubuntu Core 24
- Ubuntu Server 24.04 LTS
- Ubuntu Desktop 24.04 LTS

The guide applies to devices submitted to Canonical through one of the following programmes:

- IoT Devices Enablement Programme with Certification
- IoT ODM Partner Programme

For each test job, one of the following certification statuses is specified:

Blocking

Features that are required for certification. If any of the blocking tests fails, the certification will fail.

Non-blocking

Features that are tested but not mandatory for certification. Failure in non-blocking tests will not prevent certification. However, a note will be added to the certificate to inform potential customers or users.

Note

Only categories of hardware are tested and not specific types of hardware. For example, tests are run to verify USB controllers work, but the type of peripheral(s) used during those tests are not specified.

Coverage is flexible based on customer requirements (for example, if a device's use cases don't require LEDs, then LEDs can be non-blocking)

Certain test jobs are designed to validate specific hardware capabilities, such as camera and audio playback functionality. To ensure that the required hardware capabilities are present and properly recognised on the machine under test, these features are explicitly defined in *manifest entries* and linked to the relevant test jobs. This prevents test jobs from being skipped due to system deficiencies in automated detection.

Full test descriptions can be found in Canonical certification site for partners:

<https://certification.canonical.com>

2. client-cert-iot-ubuntu-core-24

Note

The certification tests presented in this document are validated by [Checkbox¹](#) version 4.3.0.dev71.

2.1. Blocking

2.1.1. Advanced Configuration and Power Interface

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------|----------------------------|
| acpi/oem_osi | Test ACPI OEM _OSI strings |

2.1.2. Audio tests

The following test units are covered in this category:

| Test unit ID | Summary |
|-------------------------------|--|
| audio/alsa-loopback-automated | Captured sound matches played one (automated) |
| audio/alsa-playback | Playback works |
| audio/detect-capture-devices | Check that at least one audio capture device exists |
| audio/detect-playback-devices | Check that at least one audio playback device exists |

2.1.3. Bluetooth - BlueZ Self Tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--|------------------------------------|
| bluetooth/bluez-internal-bnep-tests_bluez-internal-bnep-test | BlueZ-{bluez-internal-bnep-test} |
| bluetooth/bluez-internal-hci-tests_bluez-internal-hci-test | BlueZ-{bluez-internal-hci-test} |
| bluetooth/bluez-internal-rfcomm-tests_bluez-internal-rfcomm-test | BlueZ-{bluez-internal-rfcomm-test} |
| bluetooth/bluez-internal-uc-tests_bluez-internal-uc-test | BlueZ-{bluez-internal-uc-test} |

¹ <https://github.com/canonical/checkbox/tree/beta>

2.1.4. Bluetooth tests

The following test units are covered in this category:

| Test unit ID | Summary |
|---|--|
| bluetooth/audio-a2dp | Verify Bluetooth device's High Fidelity Playback (A2DP) capability. |
| bluetooth/audio_record_playback | Verify Bluetooth HSP/HFP profile capability for recording and playback. |
| bluetooth/bluetooth_obex_send | Bluetooth OBEX send |
| bluetooth/bluez-controller-detect | Check bluez lists a controller if rkill detects one |
| bluetooth/detect | Make sure at least one bluetooth device is detected |
| bluetooth/detect-output | Store Bluetooth device information for reports. |
| bluetooth4/HOGP-keyboard | Verify HOGP keyboard functionality with Bluetooth Smart. |
| bluetooth4/beacon_eddystone_url_interface | Test system can get beacon EddyStone URL advertisements on the {interface} adapter |

2.1.5. Camera tests

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| camera/multiple-resolution-images-rpi-attachment_name | Attach an image from the multiple resolution images test on rpi |
| camera/multiple-resolution-images-rpi_name | Webcam multiple resolution capture test for Pi Camera |
| camera/multiple-resolution-images_name | Webcam multiple resolution capture test for {product_slug} |
| camera/roundtrip-qr-code_name | Test video output and camera {{ name }} by displaying and reading a QR code |

2.1.6. CPU tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------------|--|
| cpu/arm64_vfp_support_platform | Validate that the Floating Point Unit is running on {platform} device |
| cpu/armhf_vfp_support_platform | Validate that the Vector Floating Point Unit is running on {platform} device |
| cpu/clocktest | Tests the CPU for clock jitter |
| cpu/cstates | Run C-States tests |
| cpu/cstates_results.log | Attach C-States test log |
| cpu/maxfreq_test | Test that the CPU can run at its max frequency |
| cpu/maxfreq_test-log-attach | Attach CPU max frequency log |
| cpu/offlining_test | Test offlining of each CPU core |
| cpu/scaling_test | Test the CPU scaling capabilities |
| cpu/scaling_test-log-attach | Attach CPU scaling capabilities log |
| cpu/topology | Check CPU topology for accuracy between proc and sysfs |

2.1.7. Disk tests

The following test units are covered in this category:

| Test unit ID | Summary |
|-----------------------------|--|
| disk/check-software-raid | Validate the configuration of software RAID devices are expected |
| disk/detect | Gathers information about each disk detected |
| disk/encryption/detect | Test that Full Disk Encryption is in use |
| disk/read_performance_name | Disk performance test for {product_slug} |
| disk/stats_name | Disk statistics for {product_slug} |
| disk/storage_device_name | Disk I/O stress test for {product_slug} |
| thunderbolt3/storage-manual | Thunderbolt 3 HDD storage insertion + read/write + removal |

2.1.8. Docker containers

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------------|---|
| docker/build-single_arch | Test docker build with a single container |
| docker/commit_arch | Test docker commit a change to a single container |
| docker/compose-and-basic_arch | Test docker compose and basic command |
| docker/compose-restart_arch | Test compose a container with restart policy applied |
| docker/compose-single_arch | Test docker compose with a single container |
| docker/copy_arch | Test copy a file bwtween a container and local filesystem |
| docker/deploy-registry_arch | Deploy a registry server and run it on localhost |
| docker/diff_arch | Test changes to files in Ubuntu container |
| docker/export-and-import_arch | Test docker import and export a docker container |
| docker/info | Display system-wide information about docker |
| docker/inspect_arch | Test query low-level information on a docker object |
| docker/interactive_arch | Test an interactive shell in Ubuntu container |
| docker/kill_arch | Test killing containers |
| docker/restart-always_arch | Test container restart policy with always applied |
| docker/restart-on-failure_arch | Test container restart policy with on_failure applied |
| docker/run_arch | Download and run ubuntu container |
| docker/save-and-load_arch | Test docker save and load a docker image |
| docker/start-stop_arch | Start and stop a single container |
| docker/update_arch | Test update configuration of one container |
| docker/version | Display docker version information |

2.1.9. Eeprom

The following test units are covered in this category:

| Test unit ID | Summary |
|-------------------|--|
| eeprom/read-write | Test EEPROM read and write functions for the system. |

2.1.10. Error Detection And Correction (EDAC) Memory Controllers

The following test units are covered in this category:

| Test unit ID | Summary |
|---------------------|--|
| edac/default-report | Attach the default report from EDAC util |
| edac/detect | Detect if the EDAC drivers are loaded and Memory Controllers are found |

2.1.11. Ethernet Device tests

The following test units are covered in this category:

| Test unit ID | Summary |
|----------------------------|---|
| ethernet/detect | Detect if at least one ethernet device is detected |
| ethernet/hotplug-interface | Ensure hotplugging works on port {{ interface }} |
| ethernet/ping_interface | Can ping another machine over Ethernet port {interface} |
| ethernet/wol_S3_interface | Wake on LAN (WOL) test from S3 - {interface} |
| ethernet/wol_S4_interface | Wake on LAN (WOL) test from S4 - {interface} |
| ethernet/wol_S5_interface | Wake on LAN (WOL) test from S5 - {interface} |

2.1.12. Firmware tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--|---|
| firmware/fwts_desktop_diagnosis | Run FWTS QA-concerned desktop-specific diagnosis tests. |
| firmware/fwts_desktop_diagnosis_results.log.gz | Attach FWTS desktop diagnosis log to submission |

2.1.13. Gathers information about the DUT

The following test units are covered in this category:

| Test unit ID | Summary |
|------------------|---|
| connections | Collect information about connections |
| rtc | Creates resource info for RTC |
| serial_assertion | Collect serial assertions on the device |
| sleep | Create resource info for supported sleep states |
| snap | Collect information about installed snap packages |

2.1.14. General Purpose I/O

The following test units are covered in this category:

| Test unit ID | Summary |
|--|---|
| gpio/gpiomem_loopback_pairs_model | Test GPIO lines exposed on headers can be controlled via /dev/gpiomem |
| gpio/sysfs_loopback_pairs_model | Test GPIO lines exposed on headers can be controlled via sysfs |
| gpio/sysfs_loopback_pairs_vendor_product | Test GPIO lines exposed on headers can be controlled via sysfs |

2.1.15. Image verification tests

The following test units are covered in this category:

| Test unit ID | Summary |
|-------------------|---|
| image/model-grade | Check that the model grade is correctly set |

2.1.16. Informational tests

The following test units are covered in this category:

| Test unit ID | Summary |
|-------------------------------------|--|
| info/systemd-analyze | System boot-up performance statistics |
| info/systemd-analyze-critical-chain | Print the tree of the time-critical chain of SystemD |
| lspci_attachment | Attach a list of PCI devices |
| lsusb_attachment | Attach output of lsusb |
| net_if_management_attachment | Collect logging from the net_if_management job |
| parts_meta_info_attachment | Attaches an information about all parts that constituted this snap |

2.1.17. Intel Integrated Sensor Hub Transport Protocol (ISHTP)

The following test units are covered in this category:

| Test unit ID | Summary |
|--|--|
| eclite/module-detect | Verifies that the ishtp_eclite module is loaded |
| eclite/temperature-reading-from-thermal-acpitz | Read the temperature of acpitz to ensure Eclite is functional |
| ishtp/device-detect | Verify that at least 1 device entry exists in /sys/bus/ishtp/devices |
| ishtp/module-detect | Verifies that the intel_ish_ipc module for ISHTP is loaded |

2.1.18. Intel Management Engine Interface (MEI)

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------|---|
| mei/check-device | Detect if the Intel MEI device is available |
| mei/check-module | Detect if the Intel MEI kernel module is loaded |
| mei/get-firmware-version | Retrieve MEI firmware version by MEI interface |

2.1.19. I²C (Inter-Integrated Circuit)

The following test units are covered in this category:

| Test unit ID | Summary |
|-----------------------|---|
| i2c/i2c-bus-detect | Check presence of an I ² C bus |
| i2c/i2c-device-detect | Check if any I ² C devices can be detected |

2.1.20. Kernel snap tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--|--|
| kernel-snap/booted-kernel-matches-current-name | The booted kernel image matches the image in the current kernel snap |

2.1.21. LED tests

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| led-indicator/gpio-controller-leds-name | Check control of {name} LED indicator. |
| led-indicator/gpio-leds-name | Check control of {name} LED indicator. |
| led-indicator/sysfs-leds-manual | Check control of {name} LED. |
| led/fn | Test the Fn key LED functionality by activating/deactivating the Fn keys locking. |
| led/power | Power LED behavior when powered |
| led/power-blink-suspend | Power LED behavior when suspended |
| led/serial | Serial ports LED behavior |
| led/sysfs_led_brightness_off_vendor_product | Ensure the leds_adeon driver properly sets all LEDs to off or minimum brightness by running a test. |
| led/sysfs_led_brightness_on_vendor_product | Verify the functionality of the leds_adeon driver by ensuring all external LEDs achieve maximum brightness. |
| led/wireless | Verify the WLAN/Bluetooth LED functionality by toggling wireless connections. |

2.1.22. Media Card tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--|--|
| mediacard/cf-storage-manual | Test Compact Flash (CF) card insertion + read/write + removal. |
| mediacard/mmc-storage-manual | Test Multimedia Card (MMC) insertion + read/write + removal. |
| mediacard/ms-storage-manual | Test Memory Stick (MS) card insertion + read/write + removal. |
| mediacard/msp-storage-manual | Test Memory Stick Pro (MSP) card insertion + read/write + removal. |
| mediacard/sdhc-storage-manual | Test SDHC card insertion + read/write + removal. |
| mediacard/sdxc-storage-manual | Test SDXC card insertion + read/write + removal. |
| mediacard/storage-preinserted-symlink_uuid | Automated test of SD Card reading & writing ({symlink_uuid}) |
| mediacard/xd-storage-manual | Test Extreme Digital (xD) card insertion + read/write + removal. |

2.1.23. Memory tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------|--|
| memory/info | Check the amount of memory reported by meminfo against DMI |

2.1.24. Miscellaneous tests

The following test units are covered in this category:

| Test unit ID | Summary |
|-------------------------------------|--|
| miscellanea/device_check | Device Check |
| miscellanea/secure_boot_mode_gadget | Test that {gadget} Ubuntu Core system booted with Secure Boot active |
| miscellanea/submission-resources | Check that data for a complete result are present |

2.1.25. Monitor tests

The following test units are covered in this category:

| Test unit ID | Summary |
|-----------------------------|-----------------------------------|
| monitor/displayport_hotplug | Can hotplug monitor (DisplayPort) |
| monitor/dvi | Monitor works (DVI) |
| monitor/dvi-to-vga | Monitor works (DVI-to-VGA) |
| monitor/hdmi | Monitor works (HDMI) |
| monitor/hdmi-to-vga | Monitor works (HDMI-to-VGA) |
| monitor/vga | Monitor works (VGA) |

2.1.26. Non-device specific networking tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--|--|
| ipv6_detect | Test if the kernel is IPv6 ready |
| ipv6_link_local_address_interface | Test that {interface} has an IPv6 link local address |
| networking/info_device__index__ interface | Network Information of device {__index__} ({interface}) |
| networking/predictable_names | Verify that all network interfaces have predictable names. |

2.1.27. Power Management tests

The following test units are covered in this category:

| Test unit ID | Summary |
|---|--|
| power-management/cold-reboot | Cold reboot |
| power-management/post-cold-reboot | Post cold reboot service check |
| power-management/post-warm-reboot | Post warm reboot service check |
| power-management/warm-reboot | Warm reboot |
| watchdog/detect | Detect the presence of a Hardware Watchdog |
| watchdog/post-trigger-system- reset-auto | Post watchdog reset service check |
| watchdog/systemd-config | Check if the hardware watchdog is properly configured |
| watchdog/trigger-system-reset- auto | Test that the watchdog module can trigger a system reset |

2.1.28. Quadrature Encoder Peripherals

The following test units are covered in this category:

| Test unit ID | Summary |
|--|--|
| qep/qep-device-driver-for-qep- device | Verify PCI Device {qep-device} is using the correct driver |
| qep/qep-device-node-for-qep- device | Verify device directory exists for {qep-device} |

2.1.29. Real Time Clock (RTC)

The following test units are covered in this category:

| Test unit ID | Summary |
|-----------------------|--|
| rtc/battery | RTC battery tracks the time and ensures the system can wake up from power off state. |
| rtc/rtc_alarm_ rtc | Check that RTC alarm of {rtc} works |
| rtc/rtc_clock_ rtc | Check that {rtc} clock is synchronized with system clock |
| rtc/rtc_number | Check the number of RTC as expected. |

2.1.30. Serial Port

The following test units are covered in this category:

| Test unit ID | Summary |
|----------------------|---|
| serial/loopback-dev | Serial loopback test of {dev} |
| serial/rs232-console | Serial debugging console is enabled and operational |

2.1.31. Snapd

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------------|--|
| snappy/os-refresh | Refresh the system using the snap tool |
| snappy/os-revert | Rollback system update using the snap tool |
| snappy/snap-install | Test the snap install command is working |
| snappy/snap-list | Test that the snap list command is working. |
| snappy/snap-refresh-automated | Test whether the snap refresh command is working. |
| snappy/snap-remove | Test the snap remove command is working. |
| snappy/snap-reupdate-automated | Test the snap refresh command works after blacklisting. |
| snappy/snap-revert-automated | Test the snap revert command is working. |
| snappy/snap-search | Test that the snap find command is working. |
| snappy/test-snaps-confinement | Test all the snaps' confinement |
| snappy/test-store-config-store | Test that image is using the correct snappy store configuration. |
| snappy/test-store-install-beta | Snappy install command - beta channel store |
| snappy/test-store-install-edge | Snappy install command - edge channel store |
| snappy/test-system-confinement | Test if the system confinement is strict |

2.1.32. SocketCAN interface tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--|--|
| socketcan/send_packet_local_eff_virtual | Virtual CAN device support test (Local test with raw socket and EFF) |
| socketcan/send_packet_local_eff_interface | CAN device support test for {interface} (Raw, Local, EFF) |
| socketcan/send_packet_local_fd_virtual | Virtual CAN device support test (Raw, Local, FD) |
| socketcan/send_packet_local_fd_interface | CAN device support test for {interface} (Raw, Local, FD) |
| socketcan/send_packet_local_sff_virtual | Virtual CAN device support test (Raw, Local) |
| socketcan/send_packet_local_sff_interface | CAN device support test for {interface} (Raw, Local) |
| socketcan/send_packet_remote_eff_interface | CAN device support test {interface} (Raw, Remote, EFF) |
| socketcan/send_packet_remote_fd_interface | CAN device support test {interface} (Raw, Remote, FD) |
| socketcan/send_packet_remote_sff_interface | CAN device support test {interface} (Raw, Remote) |

2.1.33. TPM 2.0 (Trusted Platform Module)

The following test units are covered in this category:

| Test unit ID | Summary |
|---|--|
| clevis-encrypt-tpm2/detect-ecc-capabilities | Ensure the TPM has required capabilities for clevis ECC test |
| clevis-encrypt-tpm2/detect-rsa-capabilities | Ensure the TPM has required capabilities for clevis RSA test |
| clevis-encrypt-tpm2/ecc | clevis encrypt/decrypt key ecc |
| clevis-encrypt-tpm2/precheck | clevis encrypt/decrypt precheck |
| clevis-encrypt-tpm2/rsa | clevis encrypt/decrypt key rsa |
| tpm2/fwts-event-log-dump | Dump the contents of the TPM Event Log |

2.1.34. Ubuntu Core OS feature tests

The following test units are covered in this category:

| Test unit ID | Summary |
|-------------------------------------|--|
| ubuntucore/os-fail-boot-description | Automatically rollback after failed boot after upgrade |
| ubuntucore/os-recovery-mode | Reboot into recovery mode and log into the system using prior credentials. |
| ubuntucore/os-reinstall-mode | Reboot into reinstall mode and trigger a factory reset on the device. |
| ubuntucore/sshd | SSH is enabled and operational |

2.1.35. USB tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------------------|--|
| usb-c-otg/g_ether | Check DUT can be detected as USB ethernet device |
| usb-c-otg/g_ether-cleanup | Cleanup USB OTG ethernet interface setup after ethernet device test |
| usb-c-otg/g_mass_storage | Check DUT can be detected as a mass storage device |
| usb-c-otg/g_mass_storage-cleanup | Cleanup mass storage setup after mass storage device test |
| usb-c-otg/g_serial | Check if USB OTG can work as a serial port. |
| usb-c-otg/g_serial-cleanup | Cleanup USB OTG serial interface setup after serial device test |
| usb-c/c-to-a-adapter/hid | USB HID work on USB Type-C port using a "USB Type-C to Type-A" adapter |
| usb-c/c-to-a-adapter/storage-manual | Test USB 3 storage device insertion + read/write + removal using a "Type-C to Type-A" adapter. |
| usb-c/storage-manual | USB 3.0 storage device insertion + read/write + removal on USB Type-C port |
| usb/hid | USB keyboard works |
| usb/storage-detect | Detect storage partitions on a device on the USB bus |
| usb/storage-manual | Test USB 2.0 storage device insertion + read/write + removal. |
| usb/storage-preinserted-symlink-uuid | Test USB storage on 2.0 or 1.1 ports detected by udev ({symlink_uuid}) |
| usb3/storage-manual | Test USB 3.0 storage device insertion + read/write + removal. |

2.1.36. Wi-Fi access point

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| wireless/nmcli_wifi_ap_a_interface | Create 802.11a Wi-Fi AP on {{ interface }} using NetworkManager |
| wireless/nmcli_wifi_ap_bg_interface | Create 802.11b/g Wi-Fi AP on {{ interface }} using NetworkManager |
| wireless/wifi_ap_open_b_no_sta_interface_auto | Create open 802.11b Wi-Fi AP on {interface} with no STA |
| wireless/wifi_ap_open_b_no_sta_interface_manual | Create open 802.11b Wi-Fi AP on {interface} with no STA (Manual) |
| wireless/wifi_ap_open_g_no_sta_interface_auto | Create an open 802.11g Wi-Fi AP on {interface} with no STA connected. |
| wireless/wifi_ap_open_g_no_sta_interface_manual | Create open 802.11g Wi-Fi AP on {interface} with no STA (Manual) |
| wireless/wifi_ap_setup_wizard_interface_auto | Create Access Point on {interface} using wifi-ap.setup-wizard |
| wireless/wifi_ap_wpa_b_no_sta_interface_auto | Create WPA2 802.11b Wi-Fi AP on {interface} with no STA |
| wireless/wifi_ap_wpa_b_no_sta_interface_manual | Create WPA2 802.11b Wi-Fi AP on {interface} with no STA (Manual) |
| wireless/wifi_ap_wpa_b_with_sta_interface_auto | Create a WPA2 802.11b Wi-Fi Access Point on {interface} with active STA |
| wireless/wifi_ap_wpa_g_no_sta_interface_auto | Create WPA2 802.11g Wi-Fi AP on {interface} with no STA |
| wireless/wifi_ap_wpa_g_no_sta_interface_manual | Create WPA2 802.11g Wi-Fi AP on {interface} with no STA (Manual) |
| wireless/wifi_ap_wpa_g_with_sta_interface_auto | Create WPA2 802.11g Wi-Fi Access Point on {interface} with active STA |

2.1.37. Wireless networking tests

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| wireless/check_iwlwifi_microcode_crash_interface | Check there have been no iwlwifi crashes |
| wireless/detect | Detect if at least one Wireless LAN device is detected |
| wireless/wireless_connection_open_ac_nm_interface | Connect to unencrypted 802.11ac Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_ac_np_interface | Connect to unencrypted 802.11ac Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_open_ax_nm_interface | Connect to unencrypted 802.11ax Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_ax_np_interface | Connect to unencrypted 802.11ax Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_open_be_nm_interface | Connect to unencrypted 802.11be Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_be_np_interface | Connect to unencrypted 802.11be Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_open_bg_nm_interface | Connect to an unencrypted 802.11b/g Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_bg_np_interface | Connect to unencrypted 802.11b/g Wi-Fi network on {{ interface }} using netplan |
| wireless/wireless_connection_open_n_nm_interface | Connect to an unencrypted 802.11n Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_n_np_interface | Connect to unencrypted 802.11n Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa3_ax_nm_interface | Connect to WPA3-encrypted 802.11ax Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa3_ax_np_interface | Connect to WPA3-encrypted 802.11ax Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa3_be_nm_interface | Connect to WPA3-encrypted 802.11be Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa3_be_np_interface | Connect to WPA3-encrypted 802.11be Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_ac_nm_interface | Connect to WPA-encrypted 802.11ac Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_ac_np_interface | Connect to WPA-encrypted 802.11ac Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_ax_nm_interface | Connect to WPA-encrypted 802.11ax Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_ax_np_interface | Connect to WPA-encrypted 802.11ax Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_be_nm_interface | Connect to WPA-encrypted 802.11be Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_be_np_interface | Connect to WPA-encrypted 802.11be Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_bg_nm_interface | Connect to WPA-encrypted 802.11b/g Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_bg_np_interface | Connect to WPA-encrypted 802.11b/g Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_n_nm_interface | Connect to a WPA-encrypted 802.11n Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_n_np_interface | Connect to a WPA-encrypted 802.11n Wi-Fi network on {{ interface }} using netplan |
| wireless/wireless_scanning_interface | Test system can discover Wi-Fi networks on {{ interface }} |

2.1.38. Wireless Wide Area Network

The following test units are covered in this category:

| Test unit ID | Summary |
|--|---|
| wwan/3gpp-scan-manufacturer-model-hw_id-auto | Scan for available 3GPP networks with the {model} modem |
| wwan/check-sim-present-manufacturer-model-hw_id-auto | Check if a SIM card is present in a slot connected to the modem |
| wwan/detect | Identify if WWAN module is missing |
| wwan/gsm-connection-manufacturer-model-hw_id-auto | Verify a GSM broadband modem can create a data connection |

2.2. Non-blocking

2.2.1. Informational tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------|-------------------|
| manifest | Hardware Manifest |

2.3. Manifest Entries

The following manifest entries are required for certification:

| Manifest entry | Summary |
|----------------------------------|---|
| dangerous_grade_core_image | Image is using 'dangerous' grade (should be set to 'No' unless you're...) |
| gpio_loopback | GPIO Loopback Connector |
| has_audio_capture | Audio capture: Machine can record sound. (For example, a desktop...) |
| has_audio_loopback_connector | Audio Loopback Connector |
| has_audio_playback | Audio playback: Machine can emit sound. (For example, a desktop...) |
| has_bt_adapter | A Bluetooth Module |
| has_bt_obex_support | A Bluetooth Module with OBJECT EXchange (OBEX) Support |
| has_bt_smart | A Bluetooth Module with Smart (4.0 or later) Support |
| has_card_reader | Media Card Reader |
| has_dp | DisplayPort |
| has_dvi | DVI |
| has_eclite | Has support for Intel ISHTP eclite controller Driver (ECLITE) |
| has_edac_module | An Error Detection And Correction (EDAC) Module |
| has_eeprom | Does device supported EEPROM? |
| has_ethernet_adapter | An Ethernet Port |
| has_ethernet_wake_on_lan_support | Wake-on-LAN support through Ethernet port |
| has_hardware_watchdog | A Hardware Watchdog Timer |
| has_hdmi | HDMI |
| has_i2c | An I ² C bus |
| has_ishtp | Has support for Intel Integrated Sensor Hub Transport Protocol (ISH...) |
| has_led_fn_lock | Function key lock (Fn lock) |
| has_led_gpio_sysfs | LEDs controlled via GPIO or sysfs |

Table 1 – continued from previous page

| Manifest entry | Summary |
|--------------------------------|---|
| has_led_power | Power |
| has_led_serial | Serial transfer |
| has_led_wireless | Wireless |
| has_md_raid | Software RAID |
| has_mei | Has support for Intel Management Engine Interface (MEI) |
| has_qep | Has support for Quadrature Encoder Peripherals |
| has_sim_card | A working SIM card inserted |
| has_thunderbolt | Thunderbolt Support |
| has_thunderbolt3 | Thunderbolt 3 Support |
| has_tpm2_chip | TPM 2.0 Support |
| has_usb_storage | USB Storage Device Connected |
| has_usbc_data | USB Type-C Data (e.g. HID, Drives, Ethernet) |
| has_usbc_otg | Does the platform support USB-C OTG? |
| has_vga | VGA |
| has_wlan_adapter | A Wi-Fi Module |
| has_wwan_module | A WWAN Module |
| socket_can_echo_server_running | A SocketCAN Echo Server |

3. client-cert-iot-server-24-04

Note

The certification tests presented in this document are validated by [Checkbox²](#) version *4.3.0.dev71*.

3.1. Blocking

3.1.1. Advanced Configuration and Power Interface

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------|----------------------------|
| acpi/oem_osi | Test ACPI OEM _OSI strings |

3.1.2. Audio tests

Output needs to be undistorted between 0%-100%. Output lines tested:

- Internal speakers
- 3.5mm headphones
- HDMI audio output
- DisplayPort audio output

Input needs to be recorded undistorted between 0%-100%. Input lines tested:

- Internal microphone
- 3.5mm microphone

Plug detection: when a new audio line input or output is plugged in the system, it needs to be recognized.

The following test units are covered in this category:

| Test unit ID | Summary |
|-------------------------------|--|
| audio/alsa-loopback-automated | Captured sound matches played one (automated) |
| audio/alsa-playback | Playback works |
| audio/detect-capture-devices | Check that at least one audio capture device exists |
| audio/detect-playback-devices | Check that at least one audio playback device exists |

² <https://github.com/canonical/checkbox/tree/beta>

3.1.3. Bluetooth - BlueZ Self Tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--|------------------------------------|
| bluetooth/bluez-internal-bnep-tests_bluez-internal-bnep-test | BlueZ-{bluez-internal-bnep-test} |
| bluetooth/bluez-internal-hci-tests_bluez-internal-hci-test | BlueZ-{bluez-internal-hci-test} |
| bluetooth/bluez-internal-rfcomm-tests_bluez-internal-rfcomm-test | BlueZ-{bluez-internal-rfcomm-test} |
| bluetooth/bluez-internal-uc-tests_bluez-internal-uc-test | BlueZ-{bluez-internal-uc-test} |

3.1.4. Bluetooth tests

Bluetooth LE (Smart and Smart Ready) is tested for device scanning and pairing. Apart from pairing, several profiles are specifically tested and required:

- Eddystone Beacon
- HID Over GATT Profile (HOGP), Low-Energy keyboard or mouse with basic functionality

The following test units are covered in this category:

| Test unit ID | Summary |
|---|--|
| bluetooth/audio-a2dp | Verify Bluetooth device's High Fidelity Playback (A2DP) capability. |
| bluetooth/audio_record_playback | Verify Bluetooth HSP/HFP profile capability for recording and playback. |
| bluetooth/bluetooth_obex_send | Bluetooth OBEX send |
| bluetooth/bluez-controller-detect | Check bluez lists a controller if rfkill detects one |
| bluetooth/detect | Make sure at least one bluetooth device is detected |
| bluetooth/detect-output | Store Bluetooth device information for reports. |
| bluetooth4/HOGP-keyboard | Verify HOGP keyboard functionality with Bluetooth Smart. |
| bluetooth4/beacon_eddystone_url_interface | Test system can get beacon EddyStone URL advertisements on the {interface} adapter |

3.1.5. Camera tests

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| camera/multiple-resolution-images-rpi-attachment_name | Attach an image from the multiple resolution images test on rpi |
| camera/multiple-resolution-images-rpi_name | Webcam multiple resolution capture test for Pi Camera |
| camera/multiple-resolution-images_name | Webcam multiple resolution capture test for {product_slug} |
| camera/roundtrip-qr-code_name | Test video output and camera {{ name }} by displaying and reading a QR code |

3.1.6. CPU tests

x86_64 and ARM processors are tested to ensure proper functionality. We will test specific features as:

- CPU's performance states (frequency up and down in runtime)
- CPU's sleep states (cpu on and off in runtime)
- Running CPU at its maximum frequency

We will also include a general stress test performed for 120 minutes to verify that the system can handle a sustained high load for a period of time. This test uses the tool "stress-ng" available in the Universe repositories.

For Intel CPU's, the IPDT (Intel Processor Diagnostic Tool) test suite will be run. The diagnostic checks for brand identification, verifies the processor operating frequency, tests specific processor features, and performs a stress test on the processor.

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------------|--|
| cpu/arm64_vfp_support_platform | Validate that the Floating Point Unit is running on {platform} device |
| cpu/armhf_vfp_support_platform | Validate that the Vector Floating Point Unit is running on {platform} device |
| cpu/clocktest | Tests the CPU for clock jitter |
| cpu/cstates | Run C-States tests |
| cpu/cstates_results.log | Attach C-States test log |
| cpu/maxfreq_test | Test that the CPU can run at its max frequency |
| cpu/maxfreq_test-log-attach | Attach CPU max frequency log |
| cpu/offlining_test | Test offlining of each CPU core |
| cpu/scaling_test | Test the CPU scaling capabilities |
| cpu/scaling_test-log-attach | Attach CPU scaling capabilities log |
| cpu/topology | Check CPU topology for accuracy between proc and sysfs |

3.1.7. Disk tests

The following test units are covered in this category:

| Test unit ID | Summary |
|-------------------------------|--|
| disk/check-software-raid | Validate the configuration of software RAID devices are expected |
| disk/detect | Gathers information about each disk detected |
| disk/encryption/check-fde-tpm | Disk decryption after TPM change |
| disk/encryption/detect | Test that Full Disk Encryption is in use |
| disk/read_performance_name | Disk performance test for {product_slug} |
| disk/stats_name | Disk statistics for {product_slug} |
| disk/storage_device_name | Disk I/O stress test for {product_slug} |
| thunderbolt3/storage-manual | Thunderbolt 3 HDD storage insertion + read/write + removal |

3.1.8. Docker containers

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------------|---|
| docker/build-single_arch | Test docker build with a single container |
| docker/commit_arch | Test docker commit a change to a single container |
| docker/compose-and-basic_arch | Test docker compose and basic command |
| docker/compose-restart_arch | Test compose a container with restart policy applied |
| docker/compose-single_arch | Test docker compose with a single container |
| docker/copy_arch | Test copy a file bwtween a container and local filesystem |
| docker/deploy-registry_arch | Deploy a registry server and run it on localhost |
| docker/diff_arch | Test changes to files in Ubuntu container |
| docker/export-and-import_arch | Test docker import and export a docker container |
| docker/info | Display system-wide information about docker |
| docker/inspect_arch | Test query low-level information on a docker object |
| docker/interactive_arch | Test an interactive shell in Ubuntu container |
| docker/kill_arch | Test killing containers |
| docker/restart-always_arch | Test container restart policy with always applied |
| docker/restart-on-failure_arch | Test container restart policy with on_failure applied |
| docker/run_arch | Download and run ubuntu container |
| docker/save-and-load_arch | Test docker save and load a docker image |
| docker/start-stop_arch | Start and stop a single container |
| docker/update_arch | Test update configuration of one container |
| docker/version | Display docker version information |

3.1.9. Eeprom

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------|--|
| eeeprom/read-write | Test EEPROM read and write functions for the system. |

3.1.10. Error Detection And Correction (EDAC) Memory Controllers

The following test units are covered in this category:

| Test unit ID | Summary |
|---------------------|--|
| edac/default-report | Attach the default report from EDAC util |
| edac/detect | Detect if the EDAC drivers are loaded and Memory Controllers are found |

3.1.11. Ethernet Device tests

Connections are tested for functionality, but not for performance.

The following test units are covered in this category:

| Test unit ID | Summary |
|----------------------------|---|
| ethernet/detect | Detect if at least one ethernet device is detected |
| ethernet/hotplug-interface | Ensure hotplugging works on port {{ interface }} |
| ethernet/ping_interface | Can ping another machine over Ethernet port {interface} |
| ethernet/wol_S3_interface | Wake on LAN (WOL) test from S3 - {interface} |
| ethernet/wol_S4_interface | Wake on LAN (WOL) test from S4 - {interface} |
| ethernet/wol_S5_interface | Wake on LAN (WOL) test from S5 - {interface} |

3.1.12. Firmware tests

The Ubuntu image must be installed using the factory default bootloader firmware (for example BIOS, UEFI or uboot as applicable) and with the default options (including SecureBoot, if that's the default setting). Firmware needs to be compliant with Canonical Firmware Test Suite (FWTS).

It is recommended that after running Canonical fwts with the list of tests defined in the *Appendix A*, ideally, no CRITICAL or HIGH failures should be reported, but those are not automatically certification blockers.

The following test units are covered in this category:

| Test unit ID | Summary |
|--|---|
| firmware/fwts_desktop_diagnosis | Run FWTS QA-concerned desktop-specific diagnosis tests. |
| firmware/fwts_desktop_diagnosis_results.log.gz | Attach FWTS desktop diagnosis log to submission |

3.1.13. Gathers information about the DUT

The following test units are covered in this category:

| Test unit ID | Summary |
|------------------|---|
| connections | Collect information about connections |
| rtc | Creates resource info for RTC |
| serial_assertion | Collect serial assertions on the device |
| sleep | Create resource info for supported sleep states |
| snap | Collect information about installed snap packages |

3.1.14. General Purpose I/O

We test the functionality of individual GPIO lines when the associated controller driver in the kernel implements a GPIO Sysfs Interface via the gpiolib implementers framework. In such cases, the GPIO system may be tested in two ways:

- Direct:
 - GPIO controllers are exposed through sysfs
 - GPIO lines are accessible by the user
- Indirect:
 - Communication with device connected via GPIO

The following test units are covered in this category:

| Test unit ID | Summary |
|--|---|
| gpio/gpiomem_loopback_pairs_model | Test GPIO lines exposed on headers can be controlled via /dev/gpiomem |
| gpio/sysfs_loopback_pairs_model | Test GPIO lines exposed on headers can be controlled via sysfs |
| gpio/sysfs_loopback_pairs_vendor_product | Test GPIO lines exposed on headers can be controlled via sysfs |

3.1.15. Informational tests

The following test units are covered in this category:

| Test unit ID | Summary |
|-------------------------------------|--|
| info/systemd-analyze | System boot-up performance statistics |
| info/systemd-analyze-critical-chain | Print the tree of the time-critical chain of SystemD |
| lspci_attachment | Attach a list of PCI devices |
| lsusb_attachment | Attach output of lsusb |
| net_if_management_attachment | Collect logging from the net_if_management job |
| parts_meta_info_attachment | Attaches an information about all parts that constituted this snap |

3.1.16. Intel Integrated Sensor Hub Transport Protocol (ISHTP)

The following test units are covered in this category:

| Test unit ID | Summary |
|--|--|
| eclite/module-detect | Verifies that the ishtp_eclite module is loaded |
| eclite/temperature-reading-from-thermal-acpitz | Read the temperature of acpitz to ensure Eclite is functional |
| ishtp/device-detect | Verify that at least 1 device entry exists in /sys/bus/ishtp/devices |
| ishtp/module-detect | Verifies that the intel_ish_ipc module for ISHTP is loaded |

3.1.17. Intel Management Engine Interface (MEI)

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------|---|
| mei/check-device | Detect if the Intel MEI device is available |
| mei/check-module | Detect if the Intel MEI kernel module is loaded |
| mei/get-firmware-version | Retrieve MEI firmware version by MEI interface |

3.1.18. I²C (Inter-Integrated Circuit)

All devices attached to the I2C bus must be detectable. This includes:

- Temperature sensors
- Humidity sensors
- Accelerometers

The following test units are covered in this category:

| Test unit ID | Summary |
|-----------------------|---|
| i2c/i2c-bus-detect | Check presence of an I ² C bus |
| i2c/i2c-device-detect | Check if any I ² C devices can be detected |

3.1.19. LED tests

When LEDs exist, they will be tested by following some basic expectations here. The actual behavior may vary depending on the hardware design. To ensure that the behavior is working as expected, please be sure to test against specifications obtained from OEM, as each OEM may have different defined behavior for LEDs. The following LEDs are tested:

- Power
- Serial Port LEDs (indicating activity)

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| led-indicator/gpio-controller-leds-name | Check control of {name} LED indicator. |
| led-indicator/gpio-leds-name | Check control of {name} LED indicator. |
| led-indicator/sysfs-leds-manual | Check control of {name} LED. |
| led/fn | Test the Fn key LED functionality by activating/deactivating the Fn keys locking. |
| led/power | Power LED behavior when powered |
| led/power-blink-suspend | Power LED behavior when suspended |
| led/serial | Serial ports LED behavior |
| led/sysfs_led_brightness_off_vendor_product | Ensure the leds_aaeon driver properly sets all LEDs to off or minimum brightness by running a test. |
| led/sysfs_led_brightness_on_vendor_product | Verify the functionality of the leds_aaeon driver by ensuring all external LEDs achieve maximum brightness. |
| led/wireless | Verify the WLAN/Bluetooth LED functionality by toggling wireless connections. |

3.1.20. Media Card tests

Media Card readers are tested for read and write for the following type of cards:

- CF
- MMC
- MS
- MSP
- SD
- SDHC
- SDXC
- XD

The following test units are covered in this category:

| Test unit ID | Summary |
|--|--|
| mediacard/cf-storage-manual | Test Compact Flash (CF) card insertion + read/write + removal. |
| mediacard/mmc-storage-manual | Test Multimedia Card (MMC) insertion + read/write + removal. |
| mediacard/ms-storage-manual | Test Memory Stick (MS) card insertion + read/write + removal. |
| mediacard/msp-storage-manual | Test Memory Stick Pro (MSP) card insertion + read/write + removal. |
| mediacard/sdhc-storage-manual | Test SDHC card insertion + read/write + removal. |
| mediacard/sdxc-storage-manual | Test SDXC card insertion + read/write + removal. |
| mediacard/storage-preinserted-symlink_uuid | Automated test of SD Card reading & writing ({symlink_uuid}) |
| mediacard/xd-storage-manual | Test Extreme Digital (xD) card insertion + read/write + removal. |

3.1.21. Memory tests

Proper detection of the amount of memory installed is required (the amount of memory installed is the memory seen by the OS).

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------|--|
| memory/info | Check the amount of memory reported by meminfo against DMI |

3.1.22. Miscellaneous tests

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| install/apt-get-gets-updates | Ensure apt can access repositories and get updates without installing them, to aid in recovery from broken updates. |
| miscellanea/check_prerelease | Test that the system is not a pre-release version |
| miscellanea/debsums | Check the MD5 sums of installed Debian packages |
| miscellanea/device_check | Device Check |
| miscellanea/grub_file_check | Check if the file core.efi exists to make sure shim and grub can be upgraded |
| miscellanea/oops | Run FWTS OOPS check |
| miscellanea/oops_results.log | Attach the FWTS oops results for submission. |
| miscellanea/secure_boot_mode_gadget | Test that {gadget} Ubuntu Core system booted with Secure Boot active |
| miscellanea/submission-resources | Check that data for a complete result are present |
| miscellanea/ubuntu-desktop-minimal-recommends | Check that all the recommended packages for ubuntu-desktop-minimal are installed |
| miscellanea/ubuntu-desktop-recommends | Check that all the recommended packages for ubuntu-desktop are installed |

3.1.23. Monitor tests

Each of the available external video ports (supported ports are HDMI, DisplayPort, DVI) are tested one by one. Output to the display must work i.e. a console is presented.

The following test units are covered in this category:

| Test unit ID | Summary |
|-----------------------------|-----------------------------------|
| monitor/displayport_hotplug | Can hotplug monitor (DisplayPort) |
| monitor/dvi | Monitor works (DVI) |
| monitor/dvi-to-vga | Monitor works (DVI-to-VGA) |
| monitor/hdmi | Monitor works (HDMI) |
| monitor/hdmi-to-vga | Monitor works (HDMI-to-VGA) |
| monitor/vga | Monitor works (VGA) |

3.1.24. Non-device specific networking tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--|--|
| ipv6_detect | Test if the kernel is IPv6 ready |
| ipv6_link_local_address_interface | Test that {interface} has an IPv6 link local address |
| networking/info_device_index_interface | Network Information of device {__index__} ({interface}) |
| networking/predictable_names | Verify that all network interfaces have predictable names. |

3.1.25. Power Management tests

Warm reboot is tested such that the system must be able to perform the reboot command and services must be restarted such that systemctl does not identify a failed state.

Cold reboot is performed where an RTC is available (see next section). The wakealarm is used to reboot the system after a period of rest and services must be restarted such that systemctl does not identify a failed state.

The following test units are covered in this category:

| Test unit ID | Summary |
|---|--|
| power-management/cold-reboot | Cold reboot |
| power-management/post-cold-reboot | Post cold reboot service check |
| power-management/post-warm-reboot | Post warm reboot service check |
| power-management/warm-reboot | Warm reboot |
| watchdog/detect | Detect the presence of a Hardware Watchdog |
| watchdog/post-trigger-system-reset-auto | Post watchdog reset service check |
| watchdog/systemd-config | Check if the hardware watchdog is properly configured |
| watchdog/trigger-system-reset-auto | Test that the watchdog module can trigger a system reset |

3.1.26. Quadrature Encoder Peripherals

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------------------|--|
| qep/qep-device-driver-for-qep-device | Verify PCI Device {qep-device} is using the correct driver |
| qep/qep-device-node-for-qep-device | Verify device directory exists for {qep-device} |

3.1.27. Real Time Clock (RTC)

The following test units are covered in this category:

| Test unit ID | Summary |
|-----------------------|--|
| rtc/battery | RTC battery tracks the time and ensures the system can wake up from power off state. |
| rtc/rtc_alarm_ rtc | Check that RTC alarm of {rtc} works |
| rtc/rtc_clock_ rtc | Check that {rtc} clock is synchronized with system clock |
| rtc/rtc_number | Check the number of RTC as expected. |

3.1.28. Serial Port

Tests are carried out on ports that provide access via the Linux tty layer. The exact tests performed depend on the physical characteristics of the driver/receiver hardware. The possible tests include:

- Ensure expected number of devices are available
- Looped tests:
 - RS232 Ports: perform loopback test to ensure RX/TX
 - RS422/485 Ports: connect together to ensure RX/TX
- Machine to Machine tests: confirm that a connection can be made to another PC device and RX/TX is operational

The following test units are covered in this category:

| Test unit ID | Summary |
|----------------------|---|
| serial/loopback-dev | Serial loopback test of {dev} |
| serial/rs232-console | Serial debugging console is enabled and operational |

3.1.29. Snappd

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------------|--|
| snappy/snap-install | Test the snap install command is working |
| snappy/snap-list | Test that the snap list command is working. |
| snappy/snap-refresh-automated | Test whether the snap refresh command is working. |
| snappy/snap-remove | Test the snap remove command is working. |
| snappy/snap-reupdate-automated | Test the snap refresh command works after blacklisting. |
| snappy/snap-revert-automated | Test the snap revert command is working. |
| snappy/snap-search | Test that the snap find command is working. |
| snappy/test-snaps-confinement | Test all the snaps' confinement |
| snappy/test-store-config-store | Test that image is using the correct snappy store configuration. |
| snappy/test-store-install-beta | Snappy install command - beta channel store |
| snappy/test-store-install-edge | Snappy install command - edge channel store |
| snappy/test-system-confinement | Test if the system confinement is strict |

3.1.30. SocketCAN interface tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--|--|
| socketcan/send_packet_local_eff_virtual | Virtual CAN device support test (Local test with raw socket and EFF) |
| socketcan/send_packet_local_eff_interface | CAN device support test for {interface} (Raw, Local, EFF) |
| socketcan/send_packet_local_fd_virtual | Virtual CAN device support test (Raw, Local, FD) |
| socketcan/send_packet_local_fd_interface | CAN device support test for {interface} (Raw, Local, FD) |
| socketcan/send_packet_local_sff_virtual | Virtual CAN device support test (Raw, Local) |
| socketcan/send_packet_local_sff_interface | CAN device support test for {interface} (Raw, Local) |
| socketcan/send_packet_remote_eff_interface | CAN device support test {interface} (Raw, Remote, EFF) |
| socketcan/send_packet_remote_fd_interface | CAN device support test {interface} (Raw, Remote, FD) |
| socketcan/send_packet_remote_sff_interface | CAN device support test {interface} (Raw, Remote) |

3.1.31. TPM 2.0 (Trusted Platform Module)

On Intel and AMD x86 platforms that include TPM 2.0 compliant modules, it is required that all commands necessary to support Ubuntu's Full Disk Encryption functionality are supported.

The following test units are covered in this category:

| Test unit ID | Summary |
|---|--|
| clevis-encrypt-tpm2/detect-ecc-capabilities | Ensure the TPM has required capabilities for clevis ECC test |
| clevis-encrypt-tpm2/detect-rsa-capabilities | Ensure the TPM has required capabilities for clevis RSA test |
| clevis-encrypt-tpm2/ecc | clevis encrypt/decrypt key ecc |
| clevis-encrypt-tpm2/precheck | clevis encrypt/decrypt precheck |
| clevis-encrypt-tpm2/rsa | clevis encrypt/decrypt key rsa |
| tpm2/fwts-event-log-dump | Dump the contents of the TPM Event Log |

3.1.32. USB tests

USB 2.0

USB storage devices must work on all available USB ports. USB Human Interface Devices (HID), specifically keyboard or mouse, should be working properly on any USB port.

USB 3.0

USB storage devices must work on all available USB ports. USB Human Interface Devices (HID), specifically keyboard or mouse, should be working properly on any USB port.

USB Type C (USB 3.1)

USB Type C (USB 3.1) supports various types of devices (e.g. Video, Power) through the use of adapters or peripherals. The following adapters/peripherals should work:

- Storage devices
- Keyboard or mouse (basic functionality)
- When DisplayPort over USB Type-C is advertised:
 - **Display hot plugging and the following display are required to work:**
mirrored, extended, internal only, external only.
- Audio output needs to be undistorted over this port.

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------------------|--|
| usb-c-otg/g_ether | Check DUT can be detected as USB ethernet device |
| usb-c-otg/g_ether-cleanup | Cleanup USB OTG ethernet interface setup after ethernet device test |
| usb-c-otg/g_mass_storage | Check DUT can be detected as a mass storage device |
| usb-c-otg/g_mass_storage-cleanup | Cleanup mass storage setup after mass storage device test |
| usb-c-otg/g_serial | Check if USB OTG can work as a serial port. |
| usb-c-otg/g_serial-cleanup | Cleanup USB OTG serial interface setup after serial device test |
| usb-c/c-to-a-adapter/hid | USB HID work on USB Type-C port using a "USB Type-C to Type-A" adapter |
| usb-c/c-to-a-adapter/storage-manual | Test USB 3 storage device insertion + read/write + removal using a "Type-C to Type-A" adapter. |
| usb-c/storage-manual | USB 3.0 storage device insertion + read/write + removal on USB Type-C port |
| usb/hid | USB keyboard works |
| usb/storage-detect | Detect storage partitions on a device on the USB bus |
| usb/storage-manual | Test USB 2.0 storage device insertion + read/write + removal. |
| usb/storage-preinserted-symlink-uuid | Test USB storage on 2.0 or 1.1 ports detected by udev ({symlink_uuid}) |
| usb3/storage-manual | Test USB 3.0 storage device insertion + read/write + removal. |

3.1.33. Wi-Fi access point

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| wireless/nmcli_wifi_ap_a_interface | Create 802.11a Wi-Fi AP on {{ interface }} using NetworkManager |
| wireless/nmcli_wifi_ap_bg_interface | Create 802.11b/g Wi-Fi AP on {{ interface }} using NetworkManager |
| wireless/wifi_ap_open_b_no_sta_interface_auto | Create open 802.11b Wi-Fi AP on {interface} with no STA |
| wireless/wifi_ap_open_b_no_sta_interface_manual | Create open 802.11b Wi-Fi AP on {interface} with no STA (Manual) |
| wireless/wifi_ap_open_g_no_sta_interface_auto | Create an open 802.11g Wi-Fi AP on {interface} with no STA connected. |
| wireless/wifi_ap_open_g_no_sta_interface_manual | Create open 802.11g Wi-Fi AP on {interface} with no STA (Manual) |
| wireless/wifi_ap_setup_wizard_interface_auto | Create Access Point on {interface} using wifi-ap.setup-wizard |
| wireless/wifi_ap_wpa_b_no_sta_interface_auto | Create WPA2 802.11b Wi-Fi AP on {interface} with no STA |
| wireless/wifi_ap_wpa_b_no_sta_interface_manual | Create WPA2 802.11b Wi-Fi AP on {interface} with no STA (Manual) |
| wireless/wifi_ap_wpa_b_with_sta_interface_auto | Create a WPA2 802.11b Wi-Fi Access Point on {interface} with active STA |
| wireless/wifi_ap_wpa_g_no_sta_interface_auto | Create WPA2 802.11g Wi-Fi AP on {interface} with no STA |
| wireless/wifi_ap_wpa_g_no_sta_interface_manual | Create WPA2 802.11g Wi-Fi AP on {interface} with no STA (Manual) |
| wireless/wifi_ap_wpa_g_with_sta_interface_auto | Create WPA2 802.11g Wi-Fi Access Point on {interface} with active STA |

3.1.34. Wireless networking tests

Wi-Fi interfaces are tested for connection to access points configured for 802.11 b/g/n/ac/ax protocols.

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| wireless/check_iwlwifi_microcode_crash_interface | Check there have been no iwlwifi crashes |
| wireless/detect | Detect if at least one Wireless LAN device is detected |
| wireless/wireless_connection_open_ac_nm_interface | Connect to unencrypted 802.11ac Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_ac_np_interface | Connect to unencrypted 802.11ac Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_open_ax_nm_interface | Connect to unencrypted 802.11ax Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_ax_np_interface | Connect to unencrypted 802.11ax Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_open_be_nm_interface | Connect to unencrypted 802.11be Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_be_np_interface | Connect to unencrypted 802.11be Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_open_bg_nm_interface | Connect to an unencrypted 802.11b/g Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_bg_np_interface | Connect to unencrypted 802.11b/g Wi-Fi network on {{ interface }} using netplan |
| wireless/wireless_connection_open_n_nm_interface | Connect to an unencrypted 802.11n Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_n_np_interface | Connect to unencrypted 802.11n Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa3_ax_nm_interface | Connect to WPA3-encrypted 802.11ax Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa3_ax_np_interface | Connect to WPA3-encrypted 802.11ax Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa3_be_nm_interface | Connect to WPA3-encrypted 802.11be Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa3_be_np_interface | Connect to WPA3-encrypted 802.11be Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_ac_nm_interface | Connect to WPA-encrypted 802.11ac Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_ac_np_interface | Connect to WPA-encrypted 802.11ac Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_ax_nm_interface | Connect to WPA-encrypted 802.11ax Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_ax_np_interface | Connect to WPA-encrypted 802.11ax Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_be_nm_interface | Connect to WPA-encrypted 802.11be Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_be_np_interface | Connect to WPA-encrypted 802.11be Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_bg_nm_interface | Connect to WPA-encrypted 802.11b/g Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_bg_np_interface | Connect to WPA-encrypted 802.11b/g Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_n_nm_interface | Connect to a WPA-encrypted 802.11n Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_n_np_interface | Connect to a WPA-encrypted 802.11n Wi-Fi network on {{ interface }} using netplan |
| wireless/wireless_scanning_interface | Test system can discover Wi-Fi networks on {{ interface }} |

3.1.35. Wireless Wide Area Network

WWAN interfaces are tested for connection to 3G/4G/LTE services.

The following test units are covered in this category:

| Test unit ID | Summary |
|--|---|
| wwan/3gpp-scan-manufacturer-model-hw_id-auto | Scan for available 3GPP networks with the {model} modem |
| wwan/check-sim-present-manufacturer-model-hw_id-auto | Check if a SIM card is present in a slot connected to the modem |
| wwan/detect | Identify if WWAN module is missing |
| wwan/gsm-connection-manufacturer-model-hw_id-auto | Verify a GSM broadband modem can create a data connection |

3.2. Non-blocking

3.2.1. Informational tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------|-------------------|
| manifest | Hardware Manifest |

3.3. Manifest Entries

The following manifest entries are required for certification:

| Manifest entry | Summary |
|----------------------------------|--|
| gpio_loopback | GPIO Loopback Connector |
| has_audio_capture | Audio capture: Machine can record sound. (For example, a desktop) |
| has_audio_loopback_connector | Audio Loopback Connector |
| has_audio_playback | Audio playback: Machine can emit sound. (For example, a desktop) |
| has_bt_adapter | A Bluetooth Module |
| has_bt_obex_support | A Bluetooth Module with Object EXchange (OBEX) Support |
| has_bt_smart | A Bluetooth Module with Smart (4.0 or later) Support |
| has_card_reader | Media Card Reader |
| has_dp | DisplayPort |
| has_dvi | DVI |
| has_eclite | Has support for Intel ISHTP eclite controller Driver (ECLITE) |
| has_edac_module | An Error Detection And Correction (EDAC) Module |
| has_eeprom | Does device supported EEPROM? |
| has_ethernet_adapter | An Ethernet Port |
| has_ethernet_wake_on_lan_support | Wake-on-LAN support through Ethernet port |
| has_hardware_watchdog | A Hardware Watchdog Timer |
| has_hdmi | HDMI |
| has_i2c | An I ² C bus |
| has_ishtp | Has support for Intel Integrated Sensor Hub Transport Protocol (ISH) |
| has_led_fn_lock | Function key lock (Fn lock) |

Table 1 – continued from previous page

| Manifest entry | Summary |
|--------------------------------|---|
| has_led_gpio_sysfs | LEDs controlled via GPIO or sysfs |
| has_led_power | Power |
| has_led_serial | Serial transfer |
| has_led_wireless | Wireless |
| has_md_raid | Software RAID |
| has_mei | Has support for Intel Management Engine Interface (MEI) |
| has_qep | Has support for Quadrature Encoder Peripherals |
| has_sim_card | A working SIM card inserted |
| has_thunderbolt | Thunderbolt Support |
| has_thunderbolt3 | Thunderbolt 3 Support |
| has_tpm2_chip | TPM 2.0 Support |
| has_usb_storage | USB Storage Device Connected |
| has_usbc_data | USB Type-C Data (e.g. HID, Drives, Ethernet) |
| has_usbc_otg | Does the platform support USB-C OTG? |
| has_vga | VGA |
| has_wlan_adapter | A Wi-Fi Module |
| has_wwan_module | A WWAN Module |
| socket_can_echo_server_running | A SocketCAN Echo Server |

4. client-cert-iot-desktop-24-04

Note

The certification tests presented in this document are validated by [Checkbox³](#) version *4.3.0.dev71*.

4.1. Blocking

4.1.1. Advanced Configuration and Power Interface

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------|----------------------------|
| acpi/oem_osi | Test ACPI OEM _OSI strings |

4.1.2. Audio tests

Output needs to be undistorted between 0%-100%. Output lines tested:

- Internal speakers
- 3.5mm headphones
- HDMI audio output
- DisplayPort audio output

Input needs to be recorded undistorted between 0%-100%. Input lines tested:

- Internal microphone
- 3.5mm microphone

Plug detection: when a new audio line input or output is plugged in the system, it needs to be recognized.

The following test units are covered in this category:

| Test unit ID | Summary |
|-------------------------------|--|
| audio/alsa-loopback-automated | Captured sound matches played one (automated) |
| audio/alsa-playback | Playback works |
| audio/detect-capture-devices | Check that at least one audio capture device exists |
| audio/detect-playback-devices | Check that at least one audio playback device exists |

³ <https://github.com/canonical/checkbox/tree/beta>

4.1.3. Bluetooth - BlueZ Self Tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--|------------------------------------|
| bluetooth/bluez-internal-bnep-tests_bluez-internal-bnep-test | BlueZ-{bluez-internal-bnep-test} |
| bluetooth/bluez-internal-hci-tests_bluez-internal-hci-test | BlueZ-{bluez-internal-hci-test} |
| bluetooth/bluez-internal-rfcomm-tests_bluez-internal-rfcomm-test | BlueZ-{bluez-internal-rfcomm-test} |
| bluetooth/bluez-internal-uc-tests_bluez-internal-uc-test | BlueZ-{bluez-internal-uc-test} |

4.1.4. Bluetooth tests

Bluetooth LE (Smart and Smart Ready) is tested for device scanning and pairing. Apart from pairing, several profiles are specifically tested and required:

- Eddystone Beacon
- HID Over GATT Profile (HOGP), Low-Energy keyboard or mouse with basic functionality

The following test units are covered in this category:

| Test unit ID | Summary |
|---|--|
| bluetooth/audio-a2dp | Verify Bluetooth device's High Fidelity Playback (A2DP) capability. |
| bluetooth/audio_record_playback | Verify Bluetooth HSP/HFP profile capability for recording and playback. |
| bluetooth/bluetooth_obex_send | Bluetooth OBEX send |
| bluetooth/bluez-controller-detect | Check bluez lists a controller if rfkill detects one |
| bluetooth/detect | Make sure at least one bluetooth device is detected |
| bluetooth/detect-output | Store Bluetooth device information for reports. |
| bluetooth4/HOGP-keyboard | Verify HOGP keyboard functionality with Bluetooth Smart. |
| bluetooth4/HOGP-mouse | Test the functionality of a Bluetooth Smart mouse using HID Over GATT Profile. |
| bluetooth4/beacon_eddystone_url_interface | Test system can get beacon EddyStone URL advertisements on the {interface} adapter |

4.1.5. Camera tests

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| camera/multiple-resolution-images-rpi-attachment_name | Attach an image from the multiple resolution images test on rpi |
| camera/multiple-resolution-images-rpi_name | Webcam multiple resolution capture test for Pi Camera |
| camera/multiple-resolution-images_name | Webcam multiple resolution capture test for {product_slug} |
| camera/roundtrip-qr-code_name | Test video output and camera {{ name }} by displaying and reading a QR code |

4.1.6. CPU tests

x86_64 and ARM processors are tested to ensure proper functionality. We will test specific features as:

- CPU's performance states (frequency up and down in runtime)
- CPU's sleep states (cpu on and off in runtime)
- Running CPU at its maximum frequency

We will also include a general stress test performed for 120 minutes to verify that the system can handle a sustained high load for a period of time. This test uses the tool "stress-ng" available in the Universe repositories.

For Intel CPU's, the IPDT (Intel Processor Diagnostic Tool) test suite will be run. The diagnostic checks for brand identification, verifies the processor operating frequency, tests specific processor features, and performs a stress test on the processor.

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------------|--|
| cpu/arm64_vfp_support_platform | Validate that the Floating Point Unit is running on {platform} device |
| cpu/armhf_vfp_support_platform | Validate that the Vector Floating Point Unit is running on {platform} device |
| cpu/clocktest | Tests the CPU for clock jitter |
| cpu/cstates | Run C-States tests |
| cpu/cstates_results.log | Attach C-States test log |
| cpu/maxfreq_test | Test that the CPU can run at its max frequency |
| cpu/maxfreq_test-log-attach | Attach CPU max frequency log |
| cpu/offlining_test | Test offlining of each CPU core |
| cpu/scaling_test | Test the CPU scaling capabilities |
| cpu/scaling_test-log-attach | Attach CPU scaling capabilities log |
| cpu/topology | Check CPU topology for accuracy between proc and sysfs |

4.1.7. Disk tests

The following test units are covered in this category:

| Test unit ID | Summary |
|-------------------------------|--|
| disk/check-software-raid | Validate the configuration of software RAID devices are expected |
| disk/detect | Gathers information about each disk detected |
| disk/encryption/check-fde-tpm | Disk decryption after TPM change |
| disk/encryption/detect | Test that Full Disk Encryption is in use |
| disk/read_performance_name | Disk performance test for {product_slug} |
| disk/stats_name | Disk statistics for {product_slug} |
| disk/storage_device_name | Disk I/O stress test for {product_slug} |
| thunderbolt3/daisy-chain | Daisy-chain testing for Thunderbolt 3 storage and display device |
| thunderbolt3/storage-manual | Thunderbolt 3 HDD storage insertion + read/write + removal |

4.1.8. Docker containers

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------------|---|
| docker/build-single_arch | Test docker build with a single container |
| docker/commit_arch | Test docker commit a change to a single container |
| docker/compose-and-basic_arch | Test docker compose and basic command |
| docker/compose-restart_arch | Test compose a container with restart policy applied |
| docker/compose-single_arch | Test docker compose with a single container |
| docker/copy_arch | Test copy a file bwtween a container and local filesystem |
| docker/deploy-registry_arch | Deploy a registry server and run it on localhost |
| docker/diff_arch | Test changes to files in Ubuntu container |
| docker/export-and-import_arch | Test docker import and export a docker container |
| docker/info | Display system-wide information about docker |
| docker/inspect_arch | Test query low-level information on a docker object |
| docker/interactive_arch | Test an interactive shell in Ubuntu container |
| docker/kill_arch | Test killing containers |
| docker/restart-always_arch | Test container restart policy with always applied |
| docker/restart-on-failure_arch | Test container restart policy with on_failure applied |
| docker/run_arch | Download and run ubuntu container |
| docker/save-and-load_arch | Test docker save and load a docker image |
| docker/start-stop_arch | Start and stop a single container |
| docker/update_arch | Test update configuration of one container |
| docker/version | Display docker version information |

4.1.9. Eeprom

The following test units are covered in this category:

| Test unit ID | Summary |
|-------------------|--|
| eeprom/read-write | Test EEPROM read and write functions for the system. |

4.1.10. Error Detection And Correction (EDAC) Memory Controllers

The following test units are covered in this category:

| Test unit ID | Summary |
|---------------------|--|
| edac/default-report | Attach the default report from EDAC util |
| edac/detect | Detect if the EDAC drivers are loaded and Memory Controllers are found |

4.1.11. Ethernet Device tests

Connections are tested for functionality, but not for performance.

The following test units are covered in this category:

| Test unit ID | Summary |
|----------------------------|---|
| ethernet/detect | Detect if at least one ethernet device is detected |
| ethernet/hotplug-interface | Ensure hotplugging works on port {{ interface }} |
| ethernet/ping_interface | Can ping another machine over Ethernet port {interface} |
| ethernet/wol_S3_interface | Wake on LAN (WOL) test from S3 - {interface} |
| ethernet/wol_S4_interface | Wake on LAN (WOL) test from S4 - {interface} |
| ethernet/wol_S5_interface | Wake on LAN (WOL) test from S5 - {interface} |

4.1.12. Firmware tests

The Ubuntu image must be installed using the factory default bootloader firmware (for example BIOS, UEFI or uboot as applicable) and with the default options (including SecureBoot, if that's the default setting). Firmware needs to be compliant with Canonical Firmware Test Suite (FWTS).

It is recommended that after running Canonical fwts with the list of tests defined in the *Appendix A*, ideally, no CRITICAL or HIGH failures should be reported, but those are not automatically certification blockers.

The following test units are covered in this category:

| Test unit ID | Summary |
|--|---|
| firmware/fwts_desktop_diagnosis | Run FWTS QA-concerned desktop-specific diagnosis tests. |
| firmware/fwts_desktop_diagnosis_results.log.gz | Attach FWTS desktop diagnosis log to submission |

4.1.13. Gathers information about the DUT

The following test units are covered in this category:

| Test unit ID | Summary |
|------------------|---|
| connections | Collect information about connections |
| rtc | Creates resource info for RTC |
| serial_assertion | Collect serial assertions on the device |
| sleep | Create resource info for supported sleep states |
| snap | Collect information about installed snap packages |

4.1.14. General Purpose I/O

We test the functionality of individual GPIO lines when the associated controller driver in the kernel implements a GPIO Sysfs Interface via the gpiolib implementers framework. In such cases, the GPIO system may be tested in two ways:

- Direct:
 - GPIO controllers are exposed through sysfs
 - GPIO lines are accessible by the user
- Indirect:
 - Communication with device connected via GPIO

The following test units are covered in this category:

| Test unit ID | Summary |
|--|---|
| gpio/gpiomem_loopback_pairs_model | Test GPIO lines exposed on headers can be controlled via /dev/gpiomem |
| gpio/sysfs_loopback_pairs_model | Test GPIO lines exposed on headers can be controlled via sysfs |
| gpio/sysfs_loopback_pairs_vendor_product | Test GPIO lines exposed on headers can be controlled via sysfs |

4.1.15. Graphics tests

The following test units are covered in this category:

| Test unit ID | Summary |
|----------------------------------|---------------------------------------|
| graphics/VESA_drivers_not_in_use | Test that VESA drivers are not in use |

4.1.16. Hardware video acceleration

The following test units are covered in this category:

| Test unit ID | Summary |
|----------------------|--------------------------------------|
| va-api/va-initialize | Detect if the VA API could be loaded |

4.1.17. Informational tests

The following test units are covered in this category:

| Test unit ID | Summary |
|-------------------------------------|--|
| info/systemd-analyze | System boot-up performance statistics |
| info/systemd-analyze-critical-chain | Print the tree of the time-critical chain of SystemD |
| lspci_attachment | Attach a list of PCI devices |
| lsusb_attachment | Attach output of lsusb |
| net_if_management_attachment | Collect logging from the net_if_management job |
| parts_meta_info_attachment | Attaches an information about all parts that constituted this snap |

4.1.18. Intel Integrated Sensor Hub Transport Protocol (ISHTP)

The following test units are covered in this category:

| Test unit ID | Summary |
|--|--|
| eclite/module-detect | Verifies that the ishtp_eclite module is loaded |
| eclite/temperature-reading-from-thermal-acpitz | Read the temperature of acpitz to ensure Eclite is functional |
| ishtp/device-detect | Verify that at least 1 device entry exists in /sys/bus/ishtp/devices |
| ishtp/module-detect | Verifies that the intel_ish_ipc module for ISHTP is loaded |

4.1.19. Intel Management Engine Interface (MEI)

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------|---|
| mei/check-device | Detect if the Intel MEI device is available |
| mei/check-module | Detect if the Intel MEI kernel module is loaded |
| mei/get-firmware-version | Retrieve MEI firmware version by MEI interface |

4.1.20. I²C (Inter-Integrated Circuit)

All devices attached to the I²C bus must be detectable. This includes:

- Temperature sensors
- Humidity sensors
- Accelerometers

The following test units are covered in this category:

| Test unit ID | Summary |
|-----------------------|---|
| i2c/i2c-bus-detect | Check presence of an I ² C bus |
| i2c/i2c-device-detect | Check if any I ² C devices can be detected |

4.1.21. LED tests

When LEDs exist, they will be tested by following some basic expectations here. The actual behavior may vary depending on the hardware design. To ensure that the behavior is working as expected, please be sure to test against specifications obtained from OEM, as each OEM may have different defined behavior for LEDs. The following LEDs are tested:

- Power
- Serial Port LEDs (indicating activity)

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| led-indicator/gpio-controller-leds-name | Check control of {name} LED indicator. |
| led-indicator/gpio-leds-name | Check control of {name} LED indicator. |
| led-indicator/sysfs-leds-manual | Check control of {name} LED. |
| led/fn | Test the Fn key LED functionality by activating/deactivating the Fn keys locking. |
| led/power | Power LED behavior when powered |
| led/power-blink-suspend | Power LED behavior when suspended |
| led/serial | Serial ports LED behavior |
| led/sysfs_led_brightness_off_vendor_product | Ensure the leds_adeon driver properly sets all LEDs to off or minimum brightness by running a test. |
| led/sysfs_led_brightness_on_vendor_product | Verify the functionality of the leds_adeon driver by ensuring all external LEDs achieve maximum brightness. |
| led/wireless | Verify the WLAN/Bluetooth LED functionality by toggling wireless connections. |

4.1.22. Media Card tests

Media Card readers are tested for read and write for the following type of cards:

- CF
- MMC
- MS
- MSP
- SD
- SDHC
- SDXC
- XD

The following test units are covered in this category:

| Test unit ID | Summary |
|--|--|
| mediacard/cf-storage-manual | Test Compact Flash (CF) card insertion + read/write + removal. |
| mediacard/mmc-storage-manual | Test Multimedia Card (MMC) insertion + read/write + removal. |
| mediacard/ms-storage-manual | Test Memory Stick (MS) card insertion + read/write + removal. |
| mediacard/msp-storage-manual | Test Memory Stick Pro (MSP) card insertion + read/write + removal. |
| mediacard/sdhc-storage-manual | Test SDHC card insertion + read/write + removal. |
| mediacard/sdxc-storage-manual | Test SDXC card insertion + read/write + removal. |
| mediacard/storage-preinserted-symlink_uuid | Automated test of SD Card reading & writing ({symlink_uuid}) |
| mediacard/xd-storage-manual | Test Extreme Digital (xD) card insertion + read/write + removal. |

4.1.23. Memory tests

Proper detection of the amount of memory installed is required (the amount of memory installed is the memory seen by the OS).

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------|--|
| memory/info | Check the amount of memory reported by meminfo against DMI |

4.1.24. Miscellaneous tests

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| install/apt-get-gets-updates | Ensure apt can access repositories and get updates without installing them, to aid in recovery from broken updates. |
| miscellanea/check_prerelease | Test that the system is not a pre-release version |
| miscellanea/chvt | Verify the system's ability to switch between a virtual terminal and the X session. |
| miscellanea/debsums | Check the MD5 sums of installed Debian packages |
| miscellanea/device_check | Device Check |
| miscellanea/grub_file_check | Check if the file core.efi exists to make sure shim and grub can be upgraded |
| miscellanea/oops | Run FWTS OOPS check |
| miscellanea/oops_results.log | Attach the FWTS oops results for submission. |
| miscellanea/secure_boot_mode_gadget | Test that {gadget} Ubuntu Core system booted with Secure Boot active |
| miscellanea/submission-resources | Check that data for a complete result are present |
| miscellanea/ubuntu-desktop-minimal-recommends | Check that all the recommended packages for ubuntu-desktop-minimal are installed |
| miscellanea/ubuntu-desktop-recommends | Check that all the recommended packages for ubuntu-desktop are installed |

4.1.25. Non-device specific networking tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--|--|
| ipv6_detect | Test if the kernel is IPv6 ready |
| ipv6_link_local_address_interface | Test that {interface} has an IPv6 link local address |
| networking/info_device__index__interface | Network Information of device {__index__} ({interface}) |
| networking/predictable_names | Verify that all network interfaces have predictable names. |

4.1.26. Power Management tests

Warm reboot is tested such that the system must be able to perform the reboot command and services must be restarted such that systemctl does not identify a failed state.

Cold reboot is performed where an RTC is available (see next section). The wakealarm is used to reboot the system after a period of rest and services must be restarted such that systemctl does not identify a failed state.

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| power-management/cold-reboot | Cold reboot |
| power-management/lid | Check if the laptop lid sensors cause the system to suspend when the lid is closed. |
| power-management/lid_close_suspend_open | Test the functionality of the laptop's lid sensor for suspend/resume actions. |
| power-management/light_sensor | Test the functionality of the Ambient Light Sensor by checking if sensor values and screen backlight change when covered. |
| power-management/post-cold-reboot | Post cold reboot service check |
| power-management/post-warm-reboot | Post warm reboot service check |
| power-management/warm-reboot | Warm reboot |
| watchdog/detect | Detect the presence of a Hardware Watchdog |
| watchdog/post-trigger-system-reset-auto | Post watchdog reset service check |
| watchdog/systemd-config | Check if the hardware watchdog is properly configured |
| watchdog/trigger-system-reset-auto | Test that the watchdog module can trigger a system reset |

4.1.27. Quadrature Encoder Peripherals

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------------------|--|
| qep/qep-device-driver-for-qep-device | Verify PCI Device {qep-device} is using the correct driver |
| qep/qep-device-node-for-qep-device | Verify device directory exists for {qep-device} |

4.1.28. Real Time Clock (RTC)

The following test units are covered in this category:

| Test unit ID | Summary |
|-------------------|--|
| rtc/battery | RTC battery tracks the time and ensures the system can wake up from power off state. |
| rtc/rtc_alarm_rtc | Check that RTC alarm of {rtc} works |
| rtc/rtc_clock_rtc | Check that {rtc} clock is synchronized with system clock |
| rtc/rtc_number | Check the number of RTC as expected. |

4.1.29. Serial Port

Tests are carried out on ports that provide access via the Linux tty layer. The exact tests performed depend on the physical characteristics of the driver/receiver hardware. The possible tests include:

- Ensure expected number of devices are available
- Looped tests:
 - RS232 Ports: perform loopback test to ensure RX/TX
 - RS422/485 Ports: connect together to ensure RX/TX
- Machine to Machine tests: confirm that a connection can be made to another PC device and RX/TX is operational

The following test units are covered in this category:

| Test unit ID | Summary |
|----------------------|---|
| serial/loopback-dev | Serial loopback test of {dev} |
| serial/rs232-console | Serial debugging console is enabled and operational |

4.1.30. Snappd

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------------|--|
| snappy/snap-install | Test the snap install command is working |
| snappy/snap-list | Test that the snap list command is working. |
| snappy/snap-refresh-automated | Test whether the snap refresh command is working. |
| snappy/snap-remove | Test the snap remove command is working. |
| snappy/snap-reupdate-automated | Test the snap refresh command works after blacklisting. |
| snappy/snap-revert-automated | Test the snap revert command is working. |
| snappy/snap-search | Test that the snap find command is working. |
| snappy/test-snaps-confinement | Test all the snaps' confinement |
| snappy/test-store-config-store | Test that image is using the correct snappy store configuration. |
| snappy/test-store-install-beta | Snappy install command - beta channel store |
| snappy/test-store-install-edge | Snappy install command - edge channel store |
| snappy/test-system-confinement | Test if the system confinement is strict |

4.1.31. SocketCAN interface tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--|--|
| socketcan/send_packet_local_eff_virtual | Virtual CAN device support test (Local test with raw socket and EFF) |
| socketcan/send_packet_local_eff_interface | CAN device support test for {interface} (Raw, Local, EFF) |
| socketcan/send_packet_local_fd_virtual | Virtual CAN device support test (Raw, Local, FD) |
| socketcan/send_packet_local_fd_interface | CAN device support test for {interface} (Raw, Local, FD) |
| socketcan/send_packet_local_sff_virtual | Virtual CAN device support test (Raw, Local) |
| socketcan/send_packet_local_sff_interface | CAN device support test for {interface} (Raw, Local) |
| socketcan/send_packet_remote_eff_interface | CAN device support test {interface} (Raw, Remote, EFF) |
| socketcan/send_packet_remote_fd_interface | CAN device support test {interface} (Raw, Remote, FD) |
| socketcan/send_packet_remote_sff_interface | CAN device support test {interface} (Raw, Remote) |

4.1.32. Touchscreen tests

The following test units are covered in this category:

| Test unit ID | Summary |
|---|--|
| touchscreen/drag-n-drop | Assess touchscreen functionality for drag & drop tasks. |
| touchscreen/evdev/2-touch-tap-product_slug | Validate proper detection of a 2-touch tap on touchscreen devices. |
| touchscreen/evdev/3-touch-tap-product_slug | Validate proper detection of a 3-touch tap on touchscreen devices. |
| touchscreen/evdev/4-touch-tap-product_slug | Validate the detection of a 4-touch tap on touchscreen devices. |
| touchscreen/evdev/single-touch-tap-product_slug | Validate the detection of a single-touch tap on touchscreen devices. |
| touchscreen/multitouch-rotate | Check touchscreen pinch gesture for rotate |
| touchscreen/multitouch-zoom | Check touchscreen pinch gesture for zoom |

4.1.33. TPM 2.0 (Trusted Platform Module)

On Intel and AMD x86 platforms that include TPM 2.0 compliant modules, it is required that all commands necessary to support Ubuntu's Full Disk Encryption functionality are supported.

The following test units are covered in this category:

| Test unit ID | Summary |
|---|--|
| clevis-encrypt-tpm2/detect-ecc-capabilities | Ensure the TPM has required capabilities for clevis ECC test |
| clevis-encrypt-tpm2/detect-rsa-capabilities | Ensure the TPM has required capabilities for clevis RSA test |
| clevis-encrypt-tpm2/ecc | clevis encrypt/decrypt key ecc |
| clevis-encrypt-tpm2/precheck | clevis encrypt/decrypt precheck |
| clevis-encrypt-tpm2/rsa | clevis encrypt/decrypt key rsa |
| tpm2/fwts-event-log-dump | Dump the contents of the TPM Event Log |

4.1.34. USB tests

USB 2.0

USB storage devices must work on all available USB ports. USB Human Interface Devices (HID), specifically keyboard or mouse, should be working properly on any USB port.

USB 3.0

USB storage devices must work on all available USB ports. USB Human Interface Devices (HID), specifically keyboard or mouse, should be working properly on any USB port.

USB Type C (USB 3.1)

USB Type C (USB 3.1) supports various types of devices (e.g. Video, Power) through the use of adapters or peripherals. The following adapters/peripherals should work:

- Storage devices
- Keyboard or mouse (basic functionality)
- When DisplayPort over USB Type-C is advertised:
- **Display hot plugging and the following display are required to work:**
mirrored, extended, internal only, external only.
- Audio output needs to be undistorted over this port.

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------------------------------|--|
| usb-c-otg/g_ether | Check DUT can be detected as USB ethernet device |
| usb-c-otg/g_ether-cleanup | Cleanup USB OTG ethernet interface setup after ethernet device test |
| usb-c-otg/g_mass_storage | Check DUT can be detected as a mass storage device |
| usb-c-otg/g_mass_storage-cleanup | Cleanup mass storage setup after mass storage device test |
| usb-c-otg/g_serial | Check if USB OTG can work as a serial port. |
| usb-c-otg/g_serial-cleanup | Cleanup USB OTG serial interface setup after serial device test |
| usb-c/c-to-a-adapter/hid | USB HID work on USB Type-C port using a "USB Type-C to Type-A" adapter |
| usb-c/c-to-a-adapter/storage-manual | Test USB 3 storage device insertion + read/write + removal using a "Type-C to Type-A" adapter. |
| usb-c/storage-manual | USB 3.0 storage device insertion + read/write + removal on USB Type-C port |
| usb/hid | USB keyboard works |
| usb/storage-detect | Detect storage partitions on a device on the USB bus |
| usb/storage-manual | Test USB 2.0 storage device insertion + read/write + removal. |
| usb/storage-preinserted-symlink-uuid | Test USB storage on 2.0 or 1.1 ports detected by udev ({symlink_uuid}) |
| usb3/storage-manual | Test USB 3.0 storage device insertion + read/write + removal. |

4.1.35. Wi-Fi access point

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| wireless/nmcli_wifi_ap_a_interface | Create 802.11a Wi-Fi AP on {{ interface }} using NetworkManager |
| wireless/nmcli_wifi_ap_bg_interface | Create 802.11b/g Wi-Fi AP on {{ interface }} using NetworkManager |
| wireless/wifi_ap_open_b_no_sta_interface_auto | Create open 802.11b Wi-Fi AP on {interface} with no STA |
| wireless/wifi_ap_open_b_no_sta_interface_manual | Create open 802.11b Wi-Fi AP on {interface} with no STA (Manual) |
| wireless/wifi_ap_open_g_no_sta_interface_auto | Create an open 802.11g Wi-Fi AP on {interface} with no STA connected. |
| wireless/wifi_ap_open_g_no_sta_interface_manual | Create open 802.11g Wi-Fi AP on {interface} with no STA (Manual) |
| wireless/wifi_ap_setup_wizard_interface_auto | Create Access Point on {interface} using wifi-ap.setup-wizard |
| wireless/wifi_ap_wpa_b_no_sta_interface_auto | Create WPA2 802.11b Wi-Fi AP on {interface} with no STA |
| wireless/wifi_ap_wpa_b_no_sta_interface_manual | Create WPA2 802.11b Wi-Fi AP on {interface} with no STA (Manual) |
| wireless/wifi_ap_wpa_b_with_sta_interface_auto | Create a WPA2 802.11b Wi-Fi Access Point on {interface} with active STA |
| wireless/wifi_ap_wpa_g_no_sta_interface_auto | Create WPA2 802.11g Wi-Fi AP on {interface} with no STA |
| wireless/wifi_ap_wpa_g_no_sta_interface_manual | Create WPA2 802.11g Wi-Fi AP on {interface} with no STA (Manual) |
| wireless/wifi_ap_wpa_g_with_sta_interface_auto | Create WPA2 802.11g Wi-Fi Access Point on {interface} with active STA |

4.1.36. Wireless networking tests

Wi-Fi interfaces are tested for connection to access points configured for 802.11 b/g/n/ac/ax protocols.

The following test units are covered in this category:

| Test unit ID | Summary |
|---|---|
| wireless/check_iwlwifi_microcode_crash_interface | Check there have been no iwlwifi crashes |
| wireless/detect | Detect if at least one Wireless LAN device is detected |
| wireless/wireless_connection_open_ac_nm_interface | Connect to unencrypted 802.11ac Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_ac_np_interface | Connect to unencrypted 802.11ac Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_open_ax_nm_interface | Connect to unencrypted 802.11ax Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_ax_np_interface | Connect to unencrypted 802.11ax Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_open_be_nm_interface | Connect to unencrypted 802.11be Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_be_np_interface | Connect to unencrypted 802.11be Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_open_bg_nm_interface | Connect to an unencrypted 802.11b/g Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_bg_np_interface | Connect to unencrypted 802.11b/g Wi-Fi network on {{ interface }} using netplan |
| wireless/wireless_connection_open_n_nm_interface | Connect to an unencrypted 802.11n Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_open_n_np_interface | Connect to unencrypted 802.11n Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa3_ax_nm_interface | Connect to WPA3-encrypted 802.11ax Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa3_ax_np_interface | Connect to WPA3-encrypted 802.11ax Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa3_be_nm_interface | Connect to WPA3-encrypted 802.11be Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa3_be_np_interface | Connect to WPA3-encrypted 802.11be Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_ac_nm_interface | Connect to WPA-encrypted 802.11ac Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_ac_np_interface | Connect to WPA-encrypted 802.11ac Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_ax_nm_interface | Connect to WPA-encrypted 802.11ax Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_ax_np_interface | Connect to WPA-encrypted 802.11ax Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_be_nm_interface | Connect to WPA-encrypted 802.11be Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_be_np_interface | Connect to WPA-encrypted 802.11be Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_bg_nm_interface | Connect to WPA-encrypted 802.11b/g Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_bg_np_interface | Connect to WPA-encrypted 802.11b/g Wi-Fi network on {{ interface }} - netplan |
| wireless/wireless_connection_wpa_n_nm_interface | Connect to a WPA-encrypted 802.11n Wi-Fi network on {{ interface }} |
| wireless/wireless_connection_wpa_n_np_interface | Connect to a WPA-encrypted 802.11n Wi-Fi network on {{ interface }} using netplan |
| wireless/wireless_scanning_interface | Test system can discover Wi-Fi networks on {{ interface }} |

4.1.37. Wireless Wide Area Network

WWAN interfaces are tested for connection to 3G/4G/LTE services.

The following test units are covered in this category:

| Test unit ID | Summary |
|--|---|
| wwan/3gpp-scan-manufacturer-model-hw_id-auto | Scan for available 3GPP networks with the {model} modem |
| wwan/check-sim-present-manufacturer-model-hw_id-auto | Check if a SIM card is present in a slot connected to the modem |
| wwan/detect | Identify if WWAN module is missing |
| wwan/gsm-connection-manufacturer-model-hw_id-auto | Verify a GSM broadband modem can create a data connection |

4.2. Non-blocking

4.2.1. Informational tests

The following test units are covered in this category:

| Test unit ID | Summary |
|--------------|-------------------|
| manifest | Hardware Manifest |

4.3. Manifest Entries

The following manifest entries are required for certification:

| Manifest entry | Summary |
|----------------------------------|---|
| gpio_loopback | GPIO Loopback Connector |
| has_audio_capture | Audio capture: Machine can record sound. (For example, a desktop P |
| has_audio_loopback_connector | Audio Loopback Connector |
| has_audio_playback | Audio playback: Machine can emit sound. (For example, a desktop P |
| has_bt_adapter | A Bluetooth Module |
| has_bt_obex_support | A Bluetooth Module with OBject EXchange (OBEX) Support |
| has_bt_smart | A Bluetooth Module with Smart (4.0 or later) Support |
| has_card_reader | Media Card Reader |
| has_eclite | Has support for Intel ISHTP eclite controller Driver (ECLITE) |
| has_edac_module | An Error Detection And Correction (EDAC) Module |
| has_eeprom | Does device supported EEPROM? |
| has_ethernet_adapter | An Ethernet Port |
| has_ethernet_wake_on_lan_support | Wake-on-LAN support through Ethernet port |
| has_hardware_watchdog | A Hardware Watchdog Timer |
| has_i2c | An I ² C bus |
| has_ishtp | Has support for Intel Integrated Sensor Hub Transport Protocol (ISH |
| has_led_fn_lock | Function key lock (Fn lock) |
| has_led_gpio_sysfs | LEDs controlled via GPIO or sysfs |
| has_led_power | Power |
| has_led_serial | Serial transfer |

Table 1 – continued from previous page

| Manifest entry | Summary |
|--------------------------------|---|
| has_led_wireless | Wireless |
| has_md_raid | Software RAID |
| has_mei | Has support for Intel Management Engine Interface (MEI) |
| has_qep | Has support for Quadrature Encoder Peripherals |
| has_sim_card | A working SIM card inserted |
| has_thunderbolt | Thunderbolt Support |
| has_thunderbolt3 | Thunderbolt 3 Support |
| has_touchscreen | Touchscreen |
| has_tpm2_chip | TPM 2.0 Support |
| has_usb_storage | USB Storage Device Connected |
| has_usbc_data | USB Type-C Data (e.g. HID, Drives, Ethernet) |
| has_usbc_otg | Does the platform support USB-C OTG? |
| has_va_api | Has support for hardware video acceleration (VA API) |
| has_wlan_adapter | A Wi-Fi Module |
| has_wwan_module | A WWAN Module |
| socket_can_echo_server_running | A SocketCAN Echo Server |

5. Appendix A. FWTS tests

As part of the certification process, we run a series of firmware tests that are part of the Canonical Firmware Test Suite. In general, any HIGH or CRITICAL error found in the fwts log can cause potential errors in the system and should be looked at by OEMs/ODMs.

| Category | Test Item | Description |
|-------------|--------------|--|
| Information | acpidump | Check ACPI table acpidump. |
| Information | version | Gather kernel system information. |
| ACPI | acpitables | ACPI table settings confidence checks. |
| ACPI | apicinstance | Check for single instance of APIC/MADT table. |
| ACPI | hpet_check | High Precision Event Timer configuration test. |
| ACPI | mcfg | MCFG PCI Express* memory mapped config space. |
| ACPI | method | ACPI DSDT Method Semantic Tests. |
| CPU | mpcheck | Check Multi Processor tables. |
| CPU | msr | CPU MSR consistency check. |
| CPU | mtrr | MTRR validation. |
| System | apicedge | APIC Edge/Level Check. |
| System | klog | Scan kernel log for errors and warnings. |