Setup Instructions for Raspberry Pi Kiosk

Contents

[Prerequisite: 2](#_Toc532558696)

[Initial Setup: 2](#_Toc532558697)

[Enable SSH: 2](#_Toc532558698)

[Change default user and password: 2](#_Toc532558699)

[Change to RSA SSH Certificates: 3](#_Toc532558700)

[Install Required Packages: 3](#_Toc532558701)

[Set Auto Login: 3](#_Toc532558702)

[Setup Wifi: 4](#_Toc532558703)

[Update to Boot Chrome: 5](#_Toc532558704)

[Update Network Priority 6](#_Toc532558705)

[Create file with a list of all Pi’s 6](#_Toc532558706)

[Update Look & Feel: 7](#_Toc532558707)

[Disable Unused Hardware Features: 7](#_Toc532558708)

[Setup Firewall and block everything but 443: 7](#_Toc532558709)

[Create Read-only Image: 8](#_Toc532558710)

[Appendix: Current digital signage displays 10](#_Toc532558711)

[Digital Signage with Ticket Numbers 10](#_Toc532558712)

[Digital Signage Calling by Name 11](#_Toc532558713)

[Digital Signage with no reception 11](#_Toc532558714)

# Prerequisite:

Download latest Raspbian Lite distribution (Jessie in my case) from the official [Raspberry website](https://downloads.raspberrypi.org/raspbian_lite_latest).\*

Then Install image on USB card and startup Raspberry Pi

# Initial Setup:

Login into the Raspberry pi using default User Id & Password

## Enable SSH:

 Enter command: sudo raspi-config

Go to: interface Options --> Enable SSH`

Exit and Reboot

From your workstation, log into the Raspberry Pi using SSH

## Change default user and password:

Enter Command: sudo passwd

Change password to something else.

Add another base id:

Enter Command: sudo useradd -m <newuser>l -G sudo

Enter Command: sudo passwd <newuser>

Enter Command: sudo shutdown -r now

Re-login with <newuser>

Enter Command: sudo deluser -remove-home pi

Enter Command: sudo nano /etc/sudoers.d/010\_pi-nopasswd

Update this line:

pi ALL=(ALL) PASSWD: ALL

Enter Command: sudo chsh -s /bin/bash <newuser>

sbcspecial sudo usermod -a -G adm,dialout,cdrom,audio,plugdev,games,users,input,netdev,gpio,i2c,spi <newuser>

# Change to RSA SSH Certificates:

Enter Command: cd ~

Enter Command: install -d -m 700 ~/.ssh

Enter Command: sudo nano .ssh/authorized\_keys

Paste public SSH Key in file, then save and exit.

Enter Command: sudo chmod 644 ~/.ssh/authorized\_keys

Enter Command: sudo chown <newuser>:<newuser> ~/.ssh/authorized\_keys

Enter Command: sudo chmod 700 ~/.ssh

Enter Command: sudo nano /etc/ssh/sshd\_config and add the following to end of the file:

Port <newport>

UsePAM no

PermitRootLogin no

AllowUsers <newuser>

PubkeyAuthentication yes

AuthorizedKeysFile .ssh/authorized\_keys

PasswordAuthentication no

 Save, exit and reboot

# Install Required Packages:

Enter Command: sudo apt-get update

Enter Command: sudo apt-get dist-upgrade

Enter Command: sudo apt-get install --no-install-recommends openbox xserver-xorg xserver-xorg-legacy xinit rpi-chromium-mods python-sense-emu python3-sense-emu unclutter matchbox-window-manager x11-xserver-utils libgl1-mesa-dri mesa-vdpau-drivers ufw libgtk-3-0 xcompmgr mesa-utils ifplugd

# Set Auto Login:

Enter Command: sudo raspi-config

Set localization (keyboard, TimeZone and system locale)

Set boot options --> b1 --> b2

Set Network options -->n3 (disasable predictive network interface)

Set Advanced Option -> Memory Split to 256MB for GPU

Set Advanced Option -> GL Driver -> G1

Reboot

# Setup Wifi:

Enter Command: sudo nano /etc/network/interfaces

Copy the following at the end of the file:

|  |
| --- |
| auto lo  iface lo inet loopback    auto eth0  allow-hotplug eth0  iface eth0 inet dhcp    auto wlan0  iface wlan0 inet dhcp  pre-up wpa\_supplicant -B -Dwext -i wlan0 -c/boot/etc/config.conf  post-down killall -q wpa\_supplicant    iface default inet dhcp |

Save & exit.

Enter Command: sudo mkdir /boot/etc

Enter Command: sudo nano /boot/etc/config.conf and paste the following:

|  |
| --- |
| ctrl\_interface=DIR=/var/run/wpa\_supplicant GROUP=netdev  #update\_config=1    network={  ssid="<SSID OF Wireless Access Point>"  scan\_ssid=1  key\_mgmt=WPA-EAP  pairwise=CCMP TKIP  group=CCMP TKIP  eap=PEAP  identity="<Active Directory User ID>"  password=hash:<hashed Password>  } |

Save and exit

# Update to boot automatically to chromium browser:

Enter Command: **sudo** nano /etc/X11/Xwrapper.config

Add these lines:

allowed\_users=anybody  
needs\_root\_rights=yes

Save & Exit

Enter Command: **sudo** gpasswd -a <newuser> tty

Enter Command: **sudo** sed -i '/^exit 0/c\chmod g+rw /dev/tty?\nexit 0' /etc/rc.local

Enter Command: **sudo** nano /home/<newuser>/startkiosk.sh

Paste the following:

|  |  |
| --- | --- |
| #!/bin/bash  # disable DPMS (Energy Star) features.  xset -dpms  # disable screen saver  xset s off  # don't blank the video device  xset s noblank  # disable mouse pointer  unclutter &  # run window manager  matchbox-window-manager -use\_cursor no -use\_titlebar no &  # run browser  echo "Starting chromium in 5 seconds..."  sleep 5    #Get MAC ADDRESSS#  IFACE=e\*  read LOCALMAC </sys/class/net/$IFACE/address    #Set address correctly based on MAC Address    while read mac tzpi url remainder  do  if [[ $LOCALMAC == $mac ]]; then  TZPI=$tzpi;  URL=$url;  fi  done < /boot/sites.txt  #Run Browser in Kiosk Mode  while :  do  chromium-browser %U --kiosk ${URL}?tz=${TZPI}  done |  |

Save and Exit

Enter Command: **sudo** nano /etc/chromium-browser/customizations/00-rpi-vars

Add this line:

CHROMIUM\_FLAGS="--ignore-gpu-blacklist --force-gpu-rasterization --enable-zero-copy --enable-native-gpu-memory-buffers --disable-infobars --disable-session-crashed-bubble --noerrdialogs --incognito --kiosk"

Save & Exit

Enter Command: sudo chmod +x /home/<newuser>/startkiosk.sh

Enter Command: sudo nano */home/*<newuser>*/.bashrc*

Add the following to the end of the file:

if [ -z "${SSH\_TTY}" ]; then

xinit ~/startkiosk.sh

fi

Save & Exit

# Update Network Priority

Enter Command: sudo nano /etc/ifplugd/action.d/ifupdown

Add the two lines in the following file that are highlighted yellow:

|  |
| --- |
| #!/bin/sh  set -e    case "$2" in  up)    /sbin/ifup $1    if [ "$1" == eth0 ]; then /sbin/ifdown wlan0 ; fi # This is a new bit    ;;  down)    /sbin/ifdown $1    if [ "$1" == eth0 ]; then /sbin/ifup wlan0 ; fi # Another new bit    ;;  esac |

Save & Exit & Reboot

# Create file with a list of all Pi’s

Enter Command: **sudo** nano /boot/sites.txt

Use this template:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Mac\_Address | Time\_Zone | URL | Location | WIFI\_MAC | Static\_IP |
| <ethernet\_mac> | America/Vancouver | <URL>/smartboard/<Office\_id> | TestMachine | <wifi mac> | <Reserved IP> |
| <ethernet\_mac> | America/Vancouver | <URL>/smartboard/<Office\_id> | TestMachine | <wifi mac> | <Reserved IP> |

Save & Exit

# Update Look & Feel:

Enter Command: sudo nano /etc/systemd/timesyncd.conf

Update with your internal time servers:

NTP=<time server 1> <time server 2>

Enter Command: sudo nano /etc/rc.local & add this before 'exit 0':

#Suppress Kernel Messages

dmesg --console-off

Enter Command: sudo nano /etc/systemd/system/autologin\@.service

Change your auto login ExecStart

FROM: ExecStart=-/sbin/agetty --autologin pi --noclear %I $TERM

TO: ExecStart=-/sbin/agetty --skip-login --noclear --noissue --login-options "-f <newuser>" %I $TERM

# Disable Unused Hardware Features:

Enter Command: sudo nano /boot/config.txt

Add this line:

dtoverlay=pi3-disable-bt

Save & Exit

Enter Command: sudo nano /etc/modprobe.d/raspi-blacklist.conf

Add these lines:

blacklist btbcm

blacklist hci\_uart

Save & Exit

Enter Command: sudo systemctl disable hciuart

# Setup Firewall and block everything but 443:

Enter Command: sudo ufw default deny incoming

Enter Command: sudo ufw default allow outgoing

Enter Command: sudo ufw allow 443

Enter Command: sudo ufw allow <newport>

Enter Command: sudo ufw enable

Enter Command: sudo ufw -- force enable

# Create Read-only Image:

Enter Command: sudo bash

Enter Command: rm -r /var/log

Enter Command: sudo nano /etc/rsyslog.conf

Comment out all items in Rules Section and add this line:

\*.\* ~

Enter Command: echo overlay >>/etc/initramfs-tools/modules

Enter Command: nano /etc/initramfs-tools/scripts/overlay

Paste the following:

|  |
| --- |
| # Local filesystem mounting -\*- shell-script -\*-  #  # This script overrides local\_mount\_root() in /scripts/local  # and mounts root as a read-only filesystem with a temporary (rw)  # overlay filesystem.  #    . /scripts/local    local\_mount\_root()  {  local\_top  local\_device\_setup "${ROOT}" "root file system"  ROOT="${DEV}"    # Get the root filesystem type if not set  if [ -z "${ROOTFSTYPE}" ]; then  FSTYPE=$(get\_fstype "${ROOT}")  else  FSTYPE=${ROOTFSTYPE}  fi    local\_premount    # CHANGES TO THE ORIGINAL FUNCTION BEGIN HERE  # N.B. this code still lacks error checking    modprobe ${FSTYPE}  checkfs ${ROOT} root "${FSTYPE}"    # Create directories for root and the overlay  mkdir /lower /upper    # Mount read-only root to /lower  if [ "${FSTYPE}" != "unknown" ]; then  mount -r -t ${FSTYPE} ${ROOTFLAGS} ${ROOT} /lower  else  mount -r ${ROOTFLAGS} ${ROOT} /lower  fi    modprobe overlay    # Mount a tmpfs for the overlay in /upper  mount -t tmpfs tmpfs /upper  mkdir /upper/data /upper/work    # Mount the final overlay-root in $rootmnt  mount -t overlay \  -olowerdir=/lower,upperdir=/upper/data,workdir=/upper/work \  overlay ${rootmnt}  } |

 Save and exit.

Enter Command: update-initramfs -c -k $(uname -r) \* Will create initrd.img<kernelversion>

Enter Command: cd /boot

Enter Command: mv initrd.img-<kernelversion> initrd7.img

Enter Command: nano /boot/config.txt

At the end, add the following.

disable\_splash=1

kernel=kernel7.img

initramfs initrd7.img

Save & Exit

Enter Command: sudo nano /boot/cmdline.txt

Add the following at the beginning of the same line:

boot=overlay console=tty3 quiet splash loglevel=0 logo.nologo vt.global\_cursor\_default=0 net.ifnames=0

Save & Exit

Reboot

***You should now have a read-only Image that will connect to Wifi & Display Chromium with the assigned URL.***

# Appendix: Current digital signage displays

# Digital Signage with Ticket Numbers



# Digital Signage Calling by Name



# Digital Signage with no reception

