

So verwenden Sie einen Raspberry Pi im Kiosk-Modus

[← Alle Tutorials](#)



Kioske sind so konzipiert, dass sie den Benutzern bestimmte Informationen oder ein bestimmtes Erlebnis bieten und gleichzeitig den Zugriff auf andere Aktivitäten auf dem Gerät verhindern. Sie sind häufig in Flughäfen, Geschäften, Krankenhäusern, Cafés und Museen zu finden – überall dort, wo Menschen einfachen Zugang zu Informationen oder Dienstleistungen wie Fahrplänen, Wartezeiten, Produktinformationen, Wegbeschreibungen, Self-Check-in-Automaten usw. benötigen. Der Kiosk-Modus auf Ihrem Raspberry Pi ermöglicht es Ihnen, direkt in eine Vollbild-Webseite oder eine Anwendung zu booten, ohne die Desktop-Umgebung zu verwenden. Es ist die Grundlage

für viele verschiedene Projekte, in denen Sie Informationen für eine dedizierte Interaktion mit einem Benutzer anzeigen möchten. Um den Kiosk-Modus zu demonstrieren, werden wir unseren Raspberry Pi so einrichten, dass er automatisch zu einer [Vollbild-raspberrypi.com-Webseite](https://vollbild-raspberrypi.com) bootet, die sich mit der time.is/London Webseite dreht.

Vorräte

- Raspberry Pi
- Raspberry Pi Netzteil (Details dazu finden Sie [in der Netzteildokumentation](#))
- microSD-Karte ([weitere Informationen finden Sie in der Dokumentation zur SD-Karte](#))
- Adapter zum Verbinden Ihrer microSD-Karte mit *Ihrem gewohnten Computer*
- Display-Kabel, um Ihren Raspberry Pi mit Ihrem Monitor zu verbinden ([weitere Informationen finden Sie in der Display-Dokumentation](#))
- Monitor

Für die Ersteinrichtung der SD-Karte benötigen Sie:

- Ein anderer Computer, der mit Ihrem Netzwerk verbunden ist. Wir bezeichnen diesen *Computer als Ihren üblichen Computer*, um ihn von dem Raspberry Pi-Computer zu unterscheiden, den Sie als Umgebungsbeleuchtungscontroller einrichten.

Wählen Sie den richtigen Raspberry Pi

Der Kiosk-Modus funktioniert auf allen Raspberry Pi-Modellen. Da in diesem Tutorial der Wayland-Fensterserver verwendet wird, empfehlen wir ein Gerät, das standardmäßig Wayland verwendet, z. B. [einen Raspberry Pi 4 Model B](#) oder [Raspberry Pi 5](#). Du kannst jedoch ältere Raspberry Pi Modelle verwenden, wenn du [zu Wayland wechselst](#).

Konfigurieren Sie Ihren Raspberry Pi

Befolgen Sie zunächst die Dokumentation ["Erste Schritte"](#), um Ihren Raspberry Pi einzurichten. Wählen Sie **als Betriebssystem Raspberry Pi OS (32-Bit)**, um Headless (ohne Maus und Tastatur) auszuführen.

Bearbeiten Sie die Einstellungen während der Phase der Betriebssystemanpassung wie folgt:

- Geben Sie einen **Hostnamen** Ihrer Wahl ein (wir empfehlen für dieses Tutorial)pi-kiosk
- Geben Sie einen **Benutzernamen** und ein **Passwort ein**. Sie benötigen diese später zur Authentifizierung
- Aktivieren Sie das Kontrollkästchen neben **WLAN konfigurieren**, damit Ihr Pi automatisch eine WLAN-Verbindung herstellen kann
 - Geben Sie Ihre **Netzwerk-SSID** (Name) und **Ihr Passwort ein**. Diese finden Sie in Ihren WLAN-Einstellungen oder auf einem Aufkleber auf Ihrem Router
- Aktivieren Sie das Kontrollkästchen neben **SSH aktivieren**, damit wir ohne Maus und Tastatur eine Verbindung zum Pi herstellen können

Remote-Verbindung zu Ihrem Raspberry Pi

SSH ermöglicht Ihnen eine drahtlose Verbindung zu Ihrem Raspberry Pi, sodass keine Tastatur und Maus erforderlich sind. Es ist perfekt, wenn sich Ihr Raspberry Pi an einer schwer zugänglichen Stelle befindet, z. B. auf der Rückseite Ihres Fernsehers.

ANMERKUNG

Um eine SSH-Verbindung zum Raspberry Pi herzustellen, verwenden Sie den Hostnamen, den Sie in Imager festgelegt haben. Wenn Sie mit dieser Methode Probleme beim Herstellen einer Verbindung haben, können Sie stattdessen die IP-Adresse des Raspberry Pi verwenden.

Weitere Informationen zum Ermitteln Ihrer IP-Adresse und zum Fernzugriff auf Ihren Raspberry Pi finden Sie [in der Dokumentation zum Fernzugriff](#).

Verbinden Sie sich über SSH

Öffnen Sie eine Terminalsitzung auf *Ihrem gewohnten Computer*. Um über SSH auf Ihren Raspberry Pi zuzugreifen, führen Sie den folgenden Befehl aus und ersetzen Sie ihn durch den Benutzernamen, den Sie im Imager ausgewählt haben:<username>

```
$ ssh <username>@pi-kiosk.local
```

```
$ ssh <username>@pi-kiosk.local
```

```
The authenticity of host 'pi-kiosk.local (fd81:b8a1:261d:1:acd4:610c
ED25519 key fingerprint is SHA256:s6aWAEe8xrbPmJzhctei7/gEQit09mj2i]
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/
[fingerprint])? yes
Warning: Permanently added 'pi-kiosk.local' (ED25519) to the list of
```

```
<username>@pi-kiosk.local's password:
```

```
Linux pi-kiosk 6.1.21-v8+ #1642 SMP PREEMPT Mon Apr  3 17:24:16 BST
```

```
The programs included with the Debian GNU/Linux system are free soft
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.
```

```
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
```

```
Last login: Tue Oct 24 09:41:00 2023
```

```
<username>@pi-kiosk:~ $
```

Wenn Sie nach Ihrem Passwort gefragt werden, verwenden Sie das Passwort, das Sie im Raspberry Pi Imager erstellt haben.

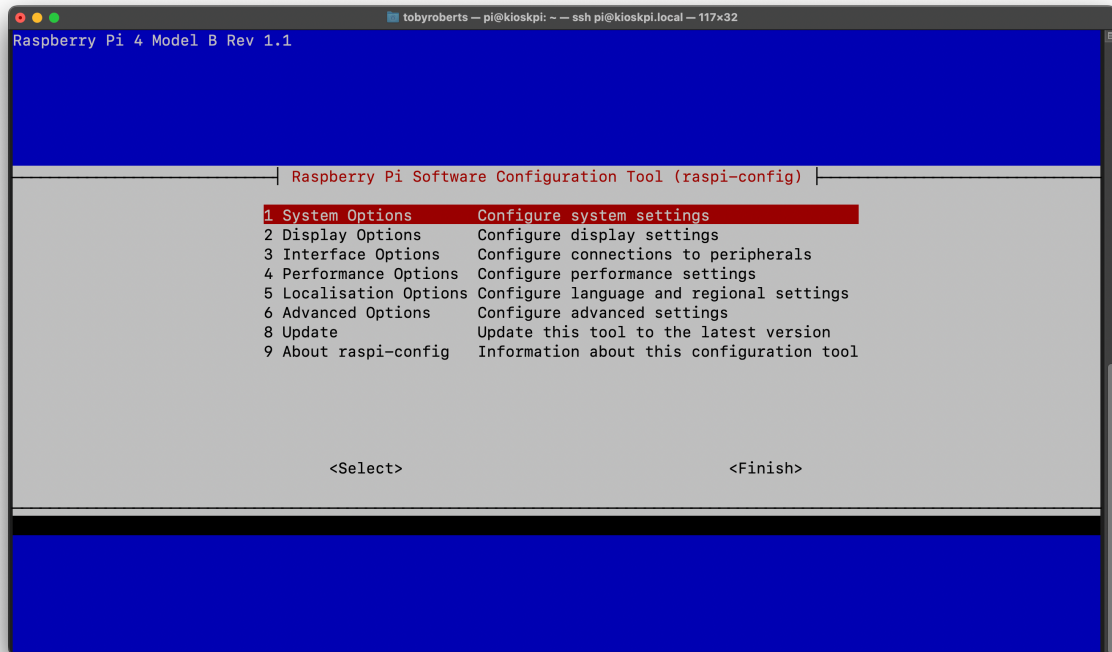
Einrichten des Kioskmodus

Für dieses Tutorial ist eine zusätzliche Software erforderlich, die die Tastaturaktivität simuliert. Um es zu installieren, führen Sie den folgenden Befehl aus:wtype

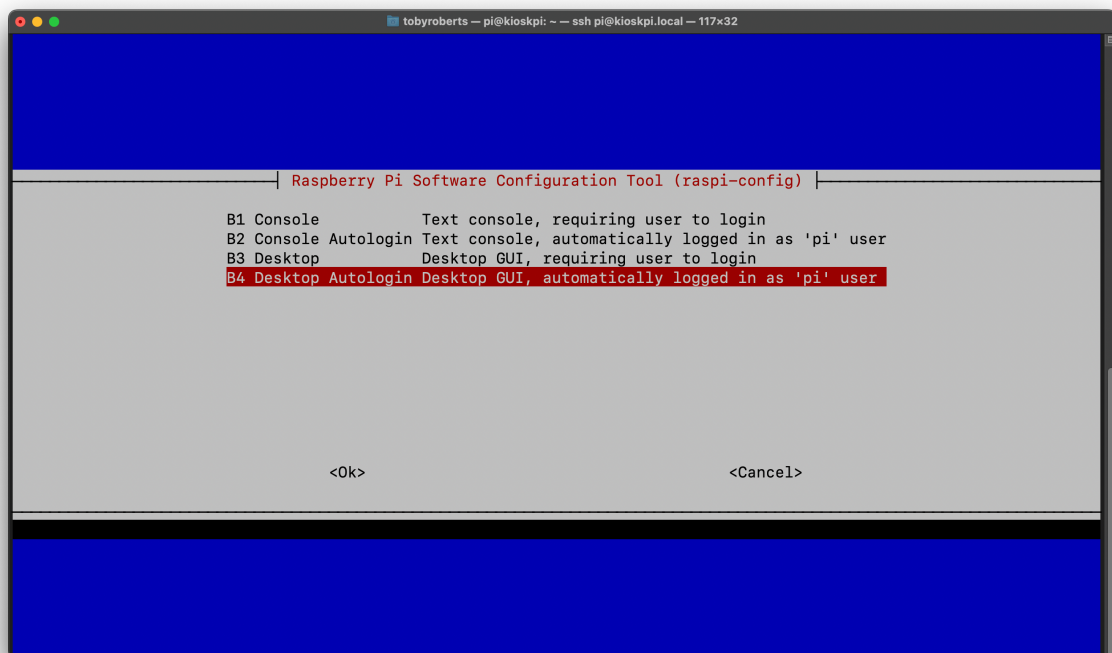
```
$ sudo apt install wtype
```

Um sicherzustellen, dass Ihr Raspberry Pi nach dem Einschalten direkt in die Desktop-Umgebung startet, verwenden Sie `:raspi-config`

```
$ sudo raspi-config
```



Wählen Sie >> aus System Options Boot / Auto Login Desktop
Autologin: Desktop GUI, automatically logged in as 'pi' user



Drücken Sie **die Eingabetaste** und **fertig stellen**, um diese Einstellung zu speichern. Wenn Sie dazu aufgefordert werden, starten Sie erneut. Nachdem Sie Ihren Raspberry Pi neu gestartet haben, sollten Sie die Desktop-Umgebung auf Ihrem Monitor sehen.

Als Nächstes weisen wir Ihren Raspberry Pi an, was er im Kiosk-Modus präsentieren soll und wie er es präsentieren soll. In diesem Tutorial zeigen wir die Raspberry Pi-Startseite und eine Seite mit der Uhrzeit in London an und wechseln alle paar Sekunden zwischen diesen beiden Seiten.

Um dies zu erreichen, werden wir bearbeiten `.config/wayfire.ini`, eine Konfigurationsdatei, die verwendet wird, um das Verhalten und das Erscheinungsbild von Wayfire zu ändern, das zum Rendern des Desktops in Raspberry Pi OS verwendet wird..`.config/wayfire.ini.config/wayfire.ini`

Führen Sie den folgenden Befehl aus, um die Datei zur Bearbeitung in nano, einem Texteditor, zu öffnen:

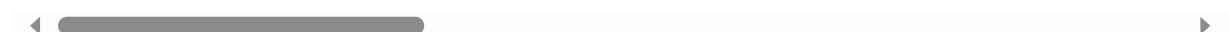
```
$ sudo nano .config/wayfire.ini
```

Werfen Sie einen Blick auf den Abschnitt mit dem Titel `[autostart]`. Im Moment liest es sich so:[autostart]

```
[autostart]
panel = wfrespawn wf-panel-pi
background = wfrespawn pcmanfm --desktop --profile LXDE-pi
xdg-autostart = lxsession-xdg-autostart
```

Fügen wir diesem Abschnitt ein paar Zeilen hinzu, um den Kiosk-Modus automatisch zu starten, wenn sich Ihr Raspberry Pi einschaltet:

```
chromium = chromium-browser https://raspberrypi.com https://time.is,
switchtab = bash ~/switchtab.sh
screensaver = false
dpms = false
```



Fügen Sie der Datei direkt unter dem vorhandenen Inhalt des Abschnitts hinzu. Wenn Sie fertig sind, sollten Sie am Ende Folgendes haben:
[autostart]

```
[autostart]
panel = wfrespawn wf-panel-pi
background = wfrespawn pcmanfm --desktop --profile LXDE-pi
xdg-autostart = lxsession-xdg-autostart
chromium = chromium-browser https://raspberrypi.com https://time.is,
switchtab = bash ~/switchtab.sh
screensaver = false
dpms = false
```

Diese Zeile öffnet den Chromium-Browser im Kioskmodus, wobei zwei Registerkarten geöffnet sind: und . Die zusätzlichen Optionen ändern den Kioskmodus auf folgende Weise: `raspberrypi.comtime.is`

--noerrdialogs

unterdrückt Fehlermeldungen

--disable-infobars

Deaktiviert die Infoleisten für Benachrichtigungen

--no-first-run

Überspringt die Erstausführung, die normalerweise beim ersten Start angezeigt wird

--ozone-platform=wayland

verwendet eine Wayland-kompatible [Ozone-Plattform](#)

--enable-features=OverlayScrollbar

Bildlaufleisten werden nur bei Bedarf angezeigt und überlagern Inhalte, anstatt einen dedizierten Bildlaufsteg zu verwenden

--start-maximized

Startet den Browser im maximierten Vollbildmodus

Schließlich führt die Zeile ein Bash-Skript aus, das alle zehn Sekunden automatisch zwischen den beiden Registerkarten wechselt. `switchtab = bash /home/kiosk/switchtab.sh`

Drücken Sie **Strg+X**, dann **Y** und schließlich **die Eingabetaste**, um die bearbeitete Datei mit zu speichern. Als Nächstes schreiben wir das Bash-Skript, das die Anzeige zwischen den beiden Registerkarten umschaltet. Normalerweise wechselt die Tastenkombination **Strg+Tab** durch die geöffneten Browser-Tabs. Unser Skript verwendet das von uns installierte

Programm, um Tastenanschläge zu simulieren und zu automatisieren.
Geben Sie Folgendes ein, um das Skript zu erstellen und es in zu
öffnen:nanowtypenano

```
$ nano ~/switchtab.sh
```

Fügen Sie der Datei Folgendes hinzu:

```
#!/bin/bash

# Find Chromium browser process ID
chromium_pid=$(pgrep chromium | head -1)

# Check if Chromium is running
while
[
[ -z $chromium_pid ]]; do
    echo "Chromium browser is not running yet."
    sleep 5
    chromium_pid=$(pgrep chromium | head -1)
done

echo "Chromium browser process ID: $chromium_pid"

export XDG_RUNTIME_DIR=/run/user/1000

# Loop to send keyboard events
while true; do
    # Send Ctrl+Tab using `wtype` command
    wtype -M ctrl -P Tab

    # Send Ctrl+Tab using `wtype` command
    wtype -m ctrl -p Tab

    sleep 10
done
```

Dieses Skript überprüft zunächst, ob der Chromium-Browser ausgeführt wird. Wenn nicht, wartet es fünf Sekunden, bevor es erneut versucht wird (dies gibt Chromium genügend Zeit zum Starten, bevor es weitergeht). Um

zwischen den beiden Registerkarten zu wechseln, simuliert das Skript alle zehn Sekunden **Strg+Tab**.wtype

Drücken Sie **Strg+X**, dann **Y** und schließlich **die Eingabetaste**, um die neue Datei mit zu speichern. Starten Sie abschließend Ihren Raspberry Pi neu:nano

```
$ sudo reboot
```

Auf Ihrem Display sollte jetzt Chrome im Kioskmodus angezeigt werden, wobei alle zehn Sekunden zwischen und gewechselt werden sollte.raspberrypi.comtime.is

Fügen Sie Sicherheit und ausfallsichere Funktionen hinzu

Geräte im Kioskmodus werden über lange Zeiträume unbeaufsichtigt ausgeführt. Sie starten häufig neu. Sie werden unbeaufsichtigt an öffentlichen Orten gelassen, was sie anfällig für unerwünschte Aufmerksamkeit oder Störungen macht. Nichts davon ist für eine zuverlässige, langfristige Installation des Kiosk-Modus förderlich, und Kioske können auch ein potenzielles Ziel für Hacker sein. Obwohl es nicht realistisch ist, sie zu 100 % sicher zu machen, gibt es eine Reihe von Dingen, die wir tun können, um Geräte zu schützen, die im Kioskmodus ausgeführt werden.

Deaktivieren von Ports

So wie dieses Projekt derzeit aussieht, würde das einfache Einstecken von Tastatur und Maus in die USB-Anschlüsse unseres Raspberry Pi einem Angreifer die volle Kontrolle geben. Eine weitere Schwachstelle stellt der Zugriff auf den Ethernet-Port dar. Um sich dagegen zu wehren, deaktivieren Sie alle ungenutzten Ports. Mit dem folgenden Befehl werden alle USB-Anschlüsse deaktiviert:

```
$ echo '1-1' | sudo tee /sys/bus/usb/drivers/usb/unbind
```

ANMERKUNG

Um die USB-Anschlüsse wieder zu aktivieren, führen Sie den folgenden Befehl aus:

```
$ echo '1-1' | sudo tee /sys/bus/usb/drivers/usb/bind
```

Um den Ethernet-Port zu deaktivieren, geben Sie den folgenden Befehl ein:

```
$ sudo ifconfig eth0 down
```

ANMERKUNG

Um den Ethernet-Port wieder zu aktivieren, führen Sie den folgenden Befehl aus:

```
$ sudo ifconfig eth0 up
```

Diese Deaktivierungsbefehle sind nicht persistent, so dass die Ports beim Aus- und Einschalten wieder funktionieren. Um sie jedes Mal zu deaktivieren, wenn das Gerät eingeschaltet wird, nehmen wir Änderungen an Ihrer Datei vor. Diese Datei wird am Ende jedes Bootvorgangs ausgeführt. Öffnen Sie die Datei zur Bearbeitung wie folgt: `/etc/rc.local`

```
$ sudo nano /etc/rc.local
```

Fügen Sie die beiden Deaktivierungsbefehle oberhalb der Zeile hinzu: `exit 0`

```
echo '1-1' | sudo tee /sys/bus/usb/drivers/usb/unbind  
sudo ifconfig eth0 down
```

Drücken Sie **Strg+X**, dann **Y** und schließlich **die Eingabetaste**, um die neue Datei mit zu speichern. Die Datei sollte nun wie folgt aussehen: `nano`

```
#!/bin/sh -e  
#  
*## rc.local  
#
```

```
*** This script is executed at the end of each multiuser runlevel.
*** Make sure that the script will "exit 0" on success or any other
*** value on error.
#
*** In order to enable or disable this script just change the execut
*** bits.
#
*** By default this script does nothing.

echo '1-1' | sudo tee /sys/bus/usb/drivers/usb/unbind
sudo ifconfig eth0 down

exit 0
```

Jedes Mal, wenn Ihr Raspberry Pi hochfährt, deaktiviert dieses Skript die Ethernet- und USB-Anschlüsse.

Bluetooth ausschalten

Um Bluetooth auszuschalten, bearbeiten Sie die Datei: /boot/firmware/config.txt

```
$ sudo nano /boot/firmware/config.txt
```

Fügen Sie am Ende der Datei die folgende Zeile hinzu:

```
dtoverlay=disable-bt
```

Drücken Sie **Strg+X**, dann **Y** und schließlich **die Eingabetaste**, um die Datei mit zu speichern. Starten Sie abschließend Ihren Raspberry Pi neu: nano

```
$ sudo reboot
```

Solange die von Ihnen hinzugefügte Leitung dort verbleibt, wird Bluetooth deaktiviert.

ANMERKUNG

Entfernen von , um Bluetooth wieder zu aktivieren. dtoverlay=disable-bt/boot/firmware/config.txt

SSH mit Schlüsseln statt mit einem Passwort einrichten

Bisher haben Sie sich mit einem Passwort über SSH mit Ihrem Raspberry Pi verbunden. Wir können eine zusätzliche Sicherheitsebene bieten, indem wir eine Verbindung mit vorgenerierten privat-öffentlichen RSA-Schlüsseln herstellen und die Möglichkeit, sich mit einem Benutzernamen und Passwort anzumelden, vollständig entfernen. Bei der schlüsselbasierten Verschlüsselung behalten Sie einen privaten Schlüssel auf *Ihrem gewohnten Computer* und einen öffentlichen Schlüssel auf Ihrem Raspberry Pi. Diese Variante der Authentifizierung ist viel schwieriger zu knacken als ein Benutzername und ein Passwort. Um Schlüssel zu erstellen, öffnen Sie ein Terminal auf *Ihrem üblichen Computer* und führen Sie Folgendes aus:

```
$ ssh-keygen
```

Wenn Sie die Standardeinstellungen akzeptieren, wenn Sie dazu aufgefordert werden, werden ein privater und ein öffentlicher Schlüssel erstellt. Der Speicherort dieser gespeicherten Dateien wird ebenfalls angegeben:

```
$ ssh-keygen
```

```
Generating public/private rsa key pair.
```

```
Enter file in which to save the key (/Users/<username>/.ssh/id_rsa)
```

```
Enter same passphrase again:
```

```
Your identification has been saved in /Users/<username>/.ssh/id_rsa
```

```
Your public key has been saved in /Users/<username>/.ssh/id_rsa.pub
```

```
The key fingerprint is:
```

```
SHA256:GYBHXNZrjEcDqqG0AV83V/yy8hI35F+8MDa+yWcN/Yo
```

```
<username>@<hostname>.local
```

```
The key's randomart image is:
```

```
+---
```

```
[RSA 3072]-----+
```

```
| . . . + = . = . |
| o . . . o * = |
| + . . . . + + |
| . + o . o B . |
| o . S = o . . |
| o = = + . |
| = = = + . |
| . . . oo + o |
| . E + . . |
```

```
+-----  
[SHA256]-----+  
$
```

Die Datei ist Ihr privater Schlüssel. Teilen Sie es nicht. Die Datei ist Ihr öffentlicher Schlüssel. Um die schlüsselbasierte Authentifizierung zu verwenden, übertragen Sie den öffentlichen Schlüssel auf Ihren Raspberry Pi.
`id_rsa.pub`

Führen Sie den folgenden Befehl aus, um Ihren öffentlichen Schlüssel auf Ihrem Terminal auszugeben:

```
$ cat ~/.ssh/id_rsa.pub
```

Kopieren Sie die Ausgabe des Befehls in die Zwischenablage. Verschieben wir nun den öffentlichen Schlüssel auf Ihren Raspberry Pi.

Verbinden Sie sich über SSH mit Ihrem Raspberry Pi:

```
$ ssh <username>@pi-kiosk.local
```

Erstellen Sie dann einen Ordner mit dem Namen in Ihrem Home-Verzeichnis: `.ssh`

```
$ mkdir ~/.ssh
```

Öffnen Sie als Nächstes einen Texteditor für eine neue Datei mit dem Namen im Ordner: `authorized_keys`

```
$ nano ~/.ssh/authorized_keys
```

Fügen Sie den Inhalt von von Ihrem gewohnten Computer in diese Datei ein. Drücken Sie **Strg+X**, dann **Y** und schließlich **die Eingabetaste**, um die Datei mit zu speichern. Starten Sie abschließend Ihren Raspberry Pi neu.
`id_rsa.pub`

```
$ sudo reboot
```

Warten Sie einige Sekunden, bis Ihr Raspberry Pi hochgefahren ist, und stellen Sie dann eine Verbindung über SSH her:

```
$ ssh <username>@pi-kiosk.local
```

Wenn alles wie erwartet funktioniert, werden Sie nicht mehr aufgefordert, ein Passwort einzugeben. Wir können jetzt den Passwortzugriff vollständig deaktivieren, indem wir die Datei bearbeiten:/etc/ssh/sshd_config

```
$ sudo nano /etc/ssh/sshd_config
```

Suchen Sie die Zeile mit dem Text . Heben Sie die Auskommentierung auf, indem Sie das entfernen, um die Zeile zu aktivieren, und ändern Sie es in , so dass es so aussieht:#PasswordAuthentication yes#yesno

```
# To disable tunneled clear text passwords, change to no here!  
PasswordAuthentication no  
#PermitEmptyPasswords no
```

Drücken Sie **Strg+X**, dann **Y** und schließlich **die Eingabetaste**, um die Datei mit zu speichern. Starten Sie Ihren Raspberry Pi noch einmal neu:nano

```
$ sudo reboot
```

Nun ist Ihr Raspberry Pi über SSH gegen Angreifer abgesichert.

Nur-Lesemodus der SD-Karte

Wenn Sie Ihren Raspberry Pi vom Stromnetz trennen, um ihn herunterzufahren, kann dies schließlich zu einer Beschädigung der Systemdatei führen. Jedes Betriebssystem, das auf einer SD-Karte gespeichert ist, kann bei längerer Verwendung möglicherweise fehlschlagen. Um diese Risiken zu verringern, können Sie Ihre SD-Karte schreibgeschützt machen. Öffnen Sie das Werkzeug mit dem folgenden Befehl:raspi-config

```
$ sudo raspi-config
```

Navigieren Sie wie folgt durch das Menüsystem:

1. Auswählen Performance options
2. Auswählen Overlay File System

3. Bestätigen Sie, dass Sie das Overlay-Dateisystem aktivieren möchten
4. Bestätigen Sie, dass Sie die Boot-Partition mit einem Schreibschutz versehen möchten
5. Neustart bestätigen

Dieser gesamte Vorgang kann einige Minuten dauern.

ANMERKUNG

Wenn Sie den Zugriff wieder aktivieren müssen, um ein Update durchzuführen oder Ihre Skripte anzupassen, verbinden Sie sich mit SSH wieder mit Ihrem Raspberry Pi und führen Sie diesen Befehl aus, um den Schreibzugriff vorübergehend zu erlauben:

```
$ sudo mount -o remount,rw /boot
```

Sie können nun alle erforderlichen Arbeiten ausführen. Ein weiterer Neustart versetzt Ihre SD-Karte in einen schreibgeschützten Zustand.

Erweitern Sie den Kioskmodus

In diesem Leitfaden werden die Grundlagen zum Einrichten eines einfachen Webseitenbetrachters im Kioskmodus erläutert. Darauf aufbauend können Sie sich auf eine Reise zu komplexeren Projekten begeben, wie z. B. einer CCTV-Anzeigestation, einem Hausautomationssystem oder vielleicht dem ultimativen Projekt im Kiosk-Modus: einem [magischen Spiegel](#). Unser Tutorial basiert auf [Magischer Spiegel²](#), dem Gewinner der [50. Ausgabe](#) des Magazins [The MagPi](#), der von der Raspberry Pi-Community gewählt wurde.

Folgen Sie uns

✉ Melden Sie sich für den Newsletter an

X x

🌀 Fäden

🎵 TikTok (Englisch)

Über Raspberry Pi

Nachrichten

Investorenbeziehungen

Kontaktieren Sie uns

Warenzeichen

 [Auf YouTube \(Englisch\)](#)

 [Auf Instagram](#)

 [LinkedIn \(Englisch\)](#)

 [Auf Facebook \(Englisch\)](#)

[Über uns](#)

[Unsere zugelassenen Wiederverkäufer](#)

[Arbeitsplätze](#)

[Zugänglichkeit](#)

[Erklärung zur modernen Sklaverei](#)

[Nutzungsbedingungen für die Website](#)

[Akzeptable Nutzung](#)

[Cookies](#)

[Zulassung](#)

[Verkaufsbedingungen](#)

[Privatsphäre](#)

[Sicherheit](#)

[Überprüfen Sie unsere Bankverbindung](#)

Für zu Hause

[Raspberry Pi für zu Hause](#)

[Tutorials](#)

Für die Industrie

[Raspberry Pi für die Industrie](#)

[Thin-Clients](#)

[Raspberry Pi im All](#)

[Angetrieben von Raspberry Pi](#)

[Design-Partner](#)

[Erfolgsgeschichten](#)

Hardware

[Computer und Mikrocontroller](#)

[Kameras und Displays](#)

[Add-on-Platinen](#)

[Netzteile und Kabel](#)

[Etuis](#)

[Peripheriegeräte](#)

Software

[Raspberry Pi Verbinden](#)

[Raspberry Pi Desktop](#)

[Raspberry Pi Imager](#)

[Raspberry Pi Betriebssystem](#)

Dokumentation

[Alle Kategorien](#)

[Produktinformationsportal](#)

[Datenblätter](#)

Gemeinschaft

[Foren](#)

[Ereignisse](#)

Raspberry Pi Store

[Informationen zum Speichern](#)

Raspberry Pi Presse

[Über Raspberry Pi Press](#)

Raspberry Pi Offizielles Magazin

Bücher