Thinclient

Nachbau des „Raspberry Pi Thin Client Project" (RPITC)

In Gang kam dieses Thema, weil ich für jemanden ein „Angebot" erstellt hatte, wo es um ein billiges (nicht preiswertes) System aus einem „Server" und bis zu fünf Clients ging. Da ist der Raspi natürlich mit dabei! Im folgenden Fall habe ich dann noch meinen Beruf einfließen lassen, bei dem ich viel mit Citrix zu tun habe. Das ist eine „Weiterentwicklung" des Microsoft Terminalservers, die ermöglichen soll, mit einfachen Clients (hier also der Raspi) hochwertige Programme auf einem Server bedienen zu können. Die Gründe dazu sind vielfältig und hier unwichtig.

Der Nachbau ist natürlich nur softwaretechnisch gemeint. Am Raspberry Pi (hier Verrsion B mit 512MB und ohne +) sind keine Änderungen nötig und man kann die Hardware inklusive Gehäuse nutzen. Nötig sind noch ein Monitor mit HDMI-Anschluss, eine flotte SD-Karte mit mindestens 2GB, sowie Tastatur und Maus. Wer damit nicht beide USB-Slots belegen möchte, weil auch noch ein USB-Stick genutzt werden soll, kann z. B. die Wireless-Logitech-Kombi K400r nutzen. Dass der Raspi ein Netzteil benötigt, ist vermutlich klar. Bei zusätzlicher Nutzung von USB sollten es möglichst mehr, als nur ein Ampere sein.

Als Voraussetzung nenn ich hier wieder Kenntnisse beim Umgang mit dem Raspi. Wer die nicht hat, sollte bitte vorher meine diversen Grundlagenseiten beim Streamingclient ansehen! – Los geht's!

1. Download des Image von der RPITC-Seite. Ich nutze hier ausschließlich die Version „RPi-TC ARMv6j Release 5"! Die darauffolgende Version enthält keinen Citrix-Receiver, den ich aber benötige! Und dieses Image ist ein sehr seltenes, weil es den Receiver in der Version 13 enthält, wenn auch 13.0, die bekanntermaßen fehlerhaft war.

2. Mittels Win32DiskImager wird **das Image auf die SD-Karte gebracht**. Anschließend kann man mittels Notepad++ die **config.txt ändern**, indem man bei „#disable_overscan=1" das erste Zeichen entfernt und ganz zum Schluss die Zeile „kernel=kernel.img" in „kernel=kernel-std.img"

ändert. Allein das Letztere herauszubekommen hat mich Stunden gekostet, war aber für die Nutzung von WLAN unerlässlich! (Es lässt sich später auch im Raspi mittels nano die Datei /boot/config.txt entsprechend ändern.)

3. Mit dieser geänderten Datei fährt man den Raspi hoch und kann sich schon mal an den Grafiken erfreuen! 🙂 Die erscheinende Readme-Datei sollte man einmal lesen und dann kann man das Fenster schließen. Zuerst **ändere ich das Keyboard**, indem ich mit rechts auf einen freien Platz im Desktop klicke und „Configurations" „Change Keyboard Configuration" auswähle. Nach ein paar Sekunden Wartezeit (der Raspi ist leider nicht der Schnellste) erscheint der Dialog, den man als „raspi-config" kennt und der genauso bedient wird. Ich stelle „Generic 105-key (Intl) PC" ein. Danach wählt man nach ein paar Navigationen durch die Menüs „German" und „German" aus. Weiter geht es mit „The default for keyboard layout" „"No compose key" und „No" für den XServer.

4. Die **Ländereinstellungen** ändert man wieder mittels Rechtsklick im Desktop, „Configurations", „Change Locales". Dort wählt man „de_DE.UTF-8 UTF-8" „None" und nach etwas Wartezeit ist auch das erledigt. Damit wir dieses Tastaturlayout nutzen können, machen wir einen reboot.

5. Leider ist die **Uhrzeit** damit noch nicht geändert. Das ist anscheinend nicht vorgesehen und muss manuell erfolgen. Damit man das auch später ändern kann und weil es logisch ist, baut man sich dafür einen weiteren Eintrag in das Configuration-Menü. Ein Rechtsklick auf den leeren Desktop, „Configuration" „Change This Menu". Im Abschnitt des „label=Configurations" fügt man hinter „…reconfigure locales…" und der darunter stehende Zeile mit „</item>" weitere Zeilen ein.

```
<item label="Change Timezone">
<action name="Execute"><command>lxterminal -e „sudo dpkg -reconfigure
tzdata"</command></action>
</item>
```

Datei speichern und schließen. Dann wieder „Configuration" und „Reload Menu Files". Wenn man alles richtig gemacht hat, kann man jetzt in „Configuration" „Change Timezone" auswählen. Dort markiert man „Europe" „Berlin" und schon stimmt auch die Uhrzeit.

6. Jetzt kann man auch noch die nervige **ReadMe ausschalten**. Dazu wieder der Desktop-Rechtsklick, „Configurations" „Change Autostart Applications". Unter „Default Autostart Appz" entfernt man die beiden Zeilen, wo es um die Readme geht und speichert die Datei.

7. Diesen Punkt muss man nur machen, wenn man **WLAN** nutzen will! Rechtsklick, „Applications" „LXTerminal". Im Folgenden wird vor jeden Befehl ein „$" vorgestellt, welches aber nicht eingegeben werden darf.

```
$sudo bash
raspberry
$cd /etc/network
$nano interfaces
```

So sollte abschließend der Inhalt aussehen:

```
auto lo
iface lo inet loopback
iface eth0 inet dhcp
allow-hotplug wlan0
iface wlan0 inet dhcp
wpa-ap-scan 1
```

wpa-scan-ssid 1
wpa-ssid „MEIN WLAN-NAME"
wpa-psk „MEIN GEHEIMER WLAN-SCHLÜSSEL"
Speichern nicht vergessen. Wer Citrix machen will, springt jetzt zuerst zu Punkt 8, ansonsten macht man
$reboot

8. Für **Citrix-Receiver**-User gibt es noch etwas zu tun – Stichwort **Zertifikat**. Man ist also mit Root-Rechten im LXTerminalfenster und gibt den folgenden Befehl genauso ein, wie er hier steht:
$ln -s /usr/share/ca-certificates/mozilla/* /opt/Citrix/ICAClient/keystore/cacerts/
Damit sollte ein auftretender SSL-Error verschwinden. Wer in Punkt 7 noch keinen reboot gemacht hatte, macht ihn jetzt:
$reboot

9. Wer häufiger die **WLAN-Settings** bearbeiten möchte und kein Sicherheitsfreak ist, für den gibt es diese Variante:
$sudo bash
raspberry
$cd /etc/network
$chmod 666 interfaces
Nun kann absolut jeder die Netzwerkeinstellungen ändern, aber das war in der alten Variante mit WPA-Supplicant auch nicht besser! Dann baue ich mir einen weiteren Eintrag in das Configurtion-Menü, ähnlich wie unter Punkt 5, aber unterhalb von Autostart-Applications:
<item label="Change Network Settings">
<action name="Execute"><command>leafpad /etc/network/interfaces</command></action>
</item>

10. Für WLAN-User, die gerne ihren **WPA-PSK-Key verstecken** wollen, gibt es noch diese Befehle, die man als Root in einer Terminalsession ausführen kann:
$cd /etc/network
$ wpa_passphrase Name_meines_WLAN Meine_geheime_ID >> interfaces
Dadurch wird an das Ende von der Datei „interfaces" der verschlüsselte Key (und noch mehr) angehangen. Alles, was hinter „psk=" steht, aber ohne die abschließende Klammer, ist der verschlüsselte Key. Dieses wird mittels des Editors „nano" komplett, aber ohne die bisherigen Anführungszeichen, hinter „wpa-psk " geschrieben.

11. Wer alles funktionsfähig hat, sollte jetzt unbedingt die **SD-Karte kopieren!**

Weitere Anmerkungen:

Auf meinem alten 15-Zoll-Monitor werden zwei Icons nicht angezeigt, weil sie außerhalb liegen. (Dafür sind die Auflösung und die Farben schöner anzusehen, als auf einem modernen 22-Zöller.) Den Browser kann man aber auch mittels Rechtsklick, „Applications" „Iceweasel" starten. Das WPA-Icon funktioniert nicht und könnte gelöscht werden. Wer Änderungen im WLAN machen möchte, muss immer über die Datei /etc/network/interfaces gehen!

Ich habe nur die Dinge getestet, die mich interessieren. Weitere Probleme sind also nicht ausgeschlossen! Trotzdem:

Viel Spaß mit diesem Produkt!