OpenWRT - mehr als ein Router

Jan Fischer

Jan.Fischer@E-Technik.TU-Chemnitz.de

Linuxstammtisch der CLUG am 13. Dezember 2013

Motivation

- Gerät in der Regel bereits vorhanden
- Beim Dauerbetrieb keine zusätzlichen Stromverbraucher
- Neue Konfigurations- und damit Einsatzmöglichkeiten
- Erweiterbarkeit durch zusätzliche Programmpakete

Hardwareauswahl 1/3

- WRT54xx gehen alle sind aber auch schon etwas veraltet, WRT54GS ist noch i.O. als Neuanschaffung
- 4MB Flash-Speicher ist knapp – besser 8 oder 16MB
- Ältere Geräte mit 2 MB Flashspeicher noch mit älteren OpenWRT-Versionen lauffähig (siehe dazu auch Imagegenerator)

Hardwareauswahl 2/3

- Im OpenWRT Einkaufsratgeber (http://wiki.openwrt.org/de/toh/buyerguide) Hauptspeichergröße als Problem genannt, aber Flashspeichergröße wirkt eher limitierend
- CPU ab 400 MHz zu empfehlen
- Vorsicht bei Gerätebezeichnungen
  - Gag der Hersteller: Ein Name verschiedene Geräte, unterschiedliche Plattformen, CPUs, Chipsätze, Speicherausstattungen

Hardwareauswahl 3/3

  - Beispiel Netgear WNDR3700 (siehe http://wiki.openwrt.org/toh/netgear/wndr3700), (Gerät soll recht gut sein, falls man das richtige erwischt)
- OpenWRT-Hardwaretabelle unter http://wiki.openwrt.org/toh/start
- Für günstigen Einkauf unter http://geizhals.de → WLAN-Router und unter Besonderheiten OpenWRT auswählen bzw. http://geizhals.de/?cat=wlanrout&xf=758_OpenWrt#xf_top

Hardwareerweiterung 1/2

- Viele Geräte mit intern vorhandenen, jedoch nicht herausgeführten diversen Ports
- Beispiel WRT54 (genaueres unter http://wiki.openwrt.org/toh/linksys/wrt54g#hardware)
  - 2 * serieller Port
  - USB-Port (2 mal vorhanden?)
  - JTAG-Port (Bietet Hilfe falls man wirklich den Bootloader komplett abgeschossen hat.)

Hardwareerweiterung 2/2

- SD-Card (tiefer Eingriff, Verlust der Funktionalität einiger LED's und des Front-Buttons siehe dazu http://wiki.openwrt.org/oldwiki/port.gpio), Kernelneucompilation notwendig
- Theoretisch weitere 8 (frei verwendbare?) IO-Pins vorhanden, jedoch wahrscheinlich nicht herausgeführt

Memory Technology Device – (MTD) 1/2

- Ansicht der Partitionierung

```
cat /proc/mtd
dev:    size   erasesize  name
mtd0: 00040000 00010000 “cfe”
mtd1: 003b0000 00010000 “linux”
mtd2: 002e5000 00010000 “rootfs”
mtd3: 000f0000 00010000 “rootfs_data”
mtd4: 00010000 00010000 “nvram”
```

Memory Technology Device – (MTD) 2/2

- Layout platformspezifisch, daher keine Befehle einfach abschreiben!
- Bearbeitung mit Befehl mtd (siehe http://wiki.openwrt.org/doc/techref/mtd)
- hauptsächlich zum Update/Downgrade des OS benötigt
- Befehl mtd mit Vorsicht und Respekt benutzen!

Installation 1/2

- Hardwaretabelle konsultieren
- Plattform bestimmen
- Informationen über den Weg zurück zur OEM-Software einholen! (plattformspezifisch)
- Unter http://downloads.openwrt.org/*Codename/Versionsnummer/Plattformname/* generic/ passendes Binary heraussuchen und herunter laden (MD5-Summe prüfen!)

Installation 2/2

- Zugang zum Weserver der OEM-Firmware verschaffen
- Binary im Webserver als Softwareupdate einspielen

Erste Schritte 1/2

- eth0 am eigenen Rechner auf 192.168.1.n (1<n<255) konfigurieren, Netzmaske entsprechend setzen.
- Verbindung per Telnet zu 192.168.1.1 herstellen
- Passwort setzen! (passwd)
- Netzwerk für eth0 einrichten (Alternative Neustart und Netzwerk per Webbrowser einrichten)

Erste Schritte 2/2

- Sobald Passwort gesetzt ist und Neustart ausgeführt wurde Zugang nur noch per ssh oder Webbrowser
- Hinweis zu WRT54 unter http://wiki.openwrt.org/toh/linksys/wrt54g#using.the.tftp.method beachten. (Sicherung des Rückweges)

Failsave-Mode

- Selbstaussperrung durch blödes Script, Firewall, …
- Ausschalten, Einschalten und schnell in Abständen von etwa 1sec Reset-Knopf drücken
- Bei Erfolg blinken Power-LED und DMZ-LED
- Zugang per Telnet über 192.168.1.1 möglich
- Methode etwas Hardware abhängig (gegebenenfalls im OpenWRT-Wiki nachsehen)

Unified Configuration Interface (UCI)

Unified Configuration Interface (UCI)

- Befehl uci *action* <*config*><*.section*><*.optionname*><=*value*>
- Befehlsinterface eher für Einzelwerte geeignet
- Syntaxbeschreibung unter http://wiki.openwrt.org/doc/uci
- Konfigurationsfiles unter /etc/config/*name*

Netzwerkkonfiguration und uci Beispiel

```
uci set network.lan.proto=static
uci set network.lan.ipaddr=192.168.111.254
uci set network.lan.netmask=255.255.255.0
uci set network.lan.gateway=192.168.110.240
uci set network.lan.dns='8.8.8.8 8.8.4.4'
uci commit network
```

Dann theoretisch

```
ifdown lan
ifup lan
```

Besser `reboot -f`

Aufbau der uci-Konfigfiles

- config 'section2'

  option 'optionname1' 'wert_a'

  option 'optionname2' 'wert_b'

  …

- config 'section2'

  #z.B. Angabe diverser Timeserver

  list 'optionname' 'wert_a'

  list 'optionname' 'wert_b'

  …

uci-Konfigfile Beispiel /etc/config/system 1/2

```
config 'system'
option 'hostname' 'loki'
option 'zonename' 'Europe/Berlin'
option 'timezone' 'CET-
1CEST,M3.5.0,M10.5.0/3'
option 'conloglevel' '8'
option 'cronloglevel' '8'
```

uci-Konfigfile Beispiel /etc/config/system 2/2

```
config 'timeserver' 'ntp'
list 'server' 'ptbtime1.ptb.de'
list 'server' 'ptbtime2.ptb.de'
list 'server' 'ptbtime3.ptb.de'
...
```

opkg der Paketmanager

- Vor Installation möglichst `opkg update`
- `opkg list` – alle verfügbaren Pakete
- `opkg list-installed` derzeit installierte Pakete
- `opkg install` *paketname* installiert Paket (Abhängigkeiten werden aufgelöst [nur bei der Installation nicht bei der Deinstallation])
- `opkg remove` paketname deinstalliert Paket

Absicherung 1/3

- Zugriff auf ssh-Server dropbear von außen und per WLAN unterbinden

```
uci set dropbear.@dropbear[0
].Interface=lan
```

- uhttp-server auf tls (https) umstellen
  - Installation von luci-ssl uhttpd-mod-tls

    ```
    opkg install luci-ssl uhttpd-mod-tls
    ```
  - `eventl. uci set uhttpd.px5g.bits=4096`
  - `Neustart /etc/init.d/uhttpd restart`

Absicherung 2/3

- uhttpd mit TLS (https) Testen und unverschüsselten http-Zugang beenden
  - Zeile list listen_http ip-addr in /etc/init.d/uhttpd auskommentieren
- uhttpd auf Lan-Interface beschränken
  - uci set uhttpd.main.listen_https=ip-addr
- WLAN vom LAN abtrennen
  - Siehe Firewall und WLAN

Absicherung 3/3

- User anlegen (dazu eventuell shadow-useradd installieren [siehe dazu “opkg der Paketmanager”])
- Files .ssh/id_rsa.pub bzw. .ssh/id_dsa.pub und .ssh/authorized_keys nach /home/username kopieren
- /etc/passwd editieren und /bin/ash als Shell eintragen

WLAN 1/5

- Am besten im Webbrowser unter Network → Wifi konfigurieren
- Zwecks WLAN-Abtrennung vom LAN unter “General” Punkt Network neues Netzwerk “wlan” erzeugen (unspecified or create)
- Andere Einstellungen wie üblich
- WPA2 Einstellungen unter “Wireless Security”

WLAN 2/5

- Alternative Einstellung unter /etc/config/wireless möglich (siehe dazu http://wiki.openwrt.org/doc/uci/wireless)
- wlan zone einrichten siehe Firewall
- Paket wifitoggle zum Ein-/Ausschalten des WLAN per Frontknopf hat beim Author nicht funktioniert

WLAN 3/5

- Alternative

```
Cat /etc/hotplug.d/button/01-radio-toggle

#!/bin/sh

if [ "$BUTTON" = "ses" ] && \ [ "$ACTION" =
"pressed" ] ; then

( sleep 1; /sbin/wlan-toggle ) &

fi
```

WLAN 4/5

```
cat /sbin/wlan-toggle

#!/bin/sh

case "$(uci get wireless.@wifi-device[0
].disabled)" in

1)

uci set wireless.@wifi-device[0].disabled=0

wifi

echo 1 > /proc/diag/led/ses_white

;;
```

WLAN 5/5

```
*)
uci set wireless.@wifi-device[0].disabled=1
wifi
echo 0 > /proc/diag/led/ses_white
echo 2 > /proc/diag/led/wlan
;;
esac
```

Firewall

- Konfiguration durch Webinterface oder per Konfigscript (/etc/config/firewall) möglich
  - Groß-/Kleinschreibung beachten!
- Für spezielle und sehr detailreiche Regeln besser Konfigscript benutzen (siehe dazu http://wiki.openwrt.org/doc/uci/firewall)
- wlan Zone einrichten unter Network → Firewall → General Settings → Zones Punkt “add”

Backup

- Durch /overlay Verzeichnis/Filesystem sehr einfach
- Gute Idee vorher nvram mit sichern
  - `nvram show >/tmp/nv.bak`
- Dann wie üblich weiter mit tar
  - `tar -zcvf owrt-back.tgz /home \`
    `/overlay /tmp/nv.bak`

Sysupgrade 1/2

- 3 Möglichkeiten: sysupgrade-script Weboberfläche oder mtd
- a) sysupgrade-script nur für Upgrade, nicht für Update verwendbar
- Unter http://downloads.openwrt.org/ *Codename/Versionsnummer/Plattformname/* generic/ passendes Binary heraussuchen und herunter laden (MD5-Summe prüfen!)
- Per ssh nach /tmp befördern

Sysupgrade 2/2

- sysupgrade -v /tmp/file
- Script prüft Version und Kompatibilität, falls nicht gegeben Fehlermeldung und Ausstieg
- b) Weboberfläche nutzt das sysupgrade-script
- c) mtd, geht immer
- Die ersten beiden Schritte wie Variante a
- mtd -r write /tmp/file firmware

Sysupdate

- Unter http://downloads.openwrt.org/*Codename/Versionsnummer/Plattformname/* generic/ trx-File heraussuchen und herunter laden (MD5-Summe prüfen!)
- Per ssh nach /tmp befördern
- mtd -r write /tmp/file.trx linux (Achtung Befehl plattformabhängig)

Imagegenerator 1/3

- Zum Bau angepasster Binary-Images ohne Entwicklungsumgebung (z.B. Zu wenig Flash-Speicher auf dem Zielsystem)
- Nur für i486 verfügbar, sonst müsste erst der Imagegenerator neu übersetzt werden
- Funktioniert vernünftig erst ab Version 12.09

Imagegenerator 2/3

- Unter http://downloads.openwrt.org/*Codename/Versionsnummer/Plattformname/*generic/OpenWrt-ImageBuilder-*plattformname*_generic-for-linux-i486.tar.bz2 herunter laden und auspacken
- $ make image PROFILE=XXX \ PACKAGES="pkg1 pkg2 pkg3" FILES=files/
- Vorgefertigte Profile für verschiedene Geräte
- Packages im gleichnamigen Unterverzeichnis

Imagegenerator 3/3

- `make image PROFILE=Broadcom-b43\`
  `PACKAGES="bc dc"`