



Versuch: Home-Control mit Raspberry Pi



Studiengänge:

- Informationssystemtechnik/
Multimediatechnik/
Elektrotechnik

Ausbildungsziel:

- Kennen lernen von allgemeinen administrativen Aufgaben zur Verwaltung von Webserver, FTP-Server, VNC-Server
- Realisierung einfacher Hardwareansteuerung (433MHz-Sender, Temperatur-Sensor) über eine HTTPS-Schnittstelle

Ausbildungsinhalte:

- Installation von WWW-Server, FTP-Server, VNC-Server
- Installation von PHP als Basis zur serverseitigen Scriptverarbeitung
- Realisierung einer HTTPS-Verbindung durch Erzeugen eines privaten SSL-Zertifikates
- Erzeugen eines einfachen PHP-Scriptes zum Ein- und Ausschalten von 433MHz-Funksteckdosen
- Erzeugen eines einfachen PHP-Scriptes zum Einlesen einer Raumtemperatur und dessen Bereitstellung über HTTPS

Gerätetechnik:

- 1 Raspberry Pi
- 1 433MHz-Sender
- 1 PC für die Administration des Raspberry Pi per SSH und VNC

Vorkenntnisse:

- Kenntnisse im Bereich Grundlagen der KT

Versuchsumfeld

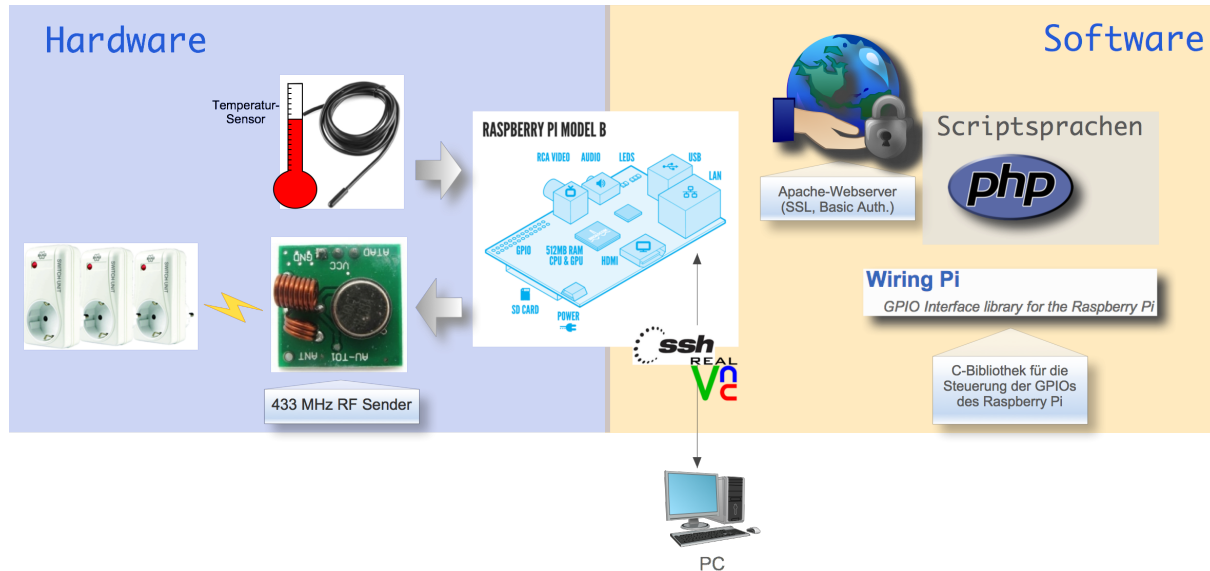


Abbildung 1: Versuchsumfeld

Einführung

Viele Geräte beinhalten zunehmend einen Webserver, welcher aufgrund des standardisierten Anwendungsprotokolls HTTP eine problemlose Administration mittels PC, Tablet oder auch Smartphone ermöglicht. Durch das weltweite routen von HTTP-Paketen ist darüberhinaus ein Zugriff auf die Geräte auch von außerhalb des eigenen Netzwerkes möglich. Desweiteren bieten standardisierte Lösungen zur Datenverschlüsselung (z.B. HTTPS) und Authentifizierung (z.B. Basic Authentication) die Möglichkeit, den Zugriff auf die Daten zu beschränken.

Ziel dieses Versuches soll es sein, auf Basis eines Raspberry Pi eine einfache Haussteuerung per Webinterface zu realisieren. Der Raspberry Pi ist ein von der gleichnamigen Raspberry Pi Foundation entwickelter Mini-Computer in der Größe einer Kreditkarte. Aufgrund des geringen Stromverbrauchs und der überschaubaren Anschaffungskosten von ca. 40,- € eignet er sich gut zum Einsatz als Steuereinheit für Hausautomationsaufgaben. Wie in Abbildung 1: Versuchsumfeld ersichtlich, verfügt der Mini-Computer über eine RCA-Video-, Audio-, 2xUSB-, LAN- und HDMI-Schnittstelle. Als Betriebssystem kommt eine ARM-optimierte Linux-Version zum Einsatz, was die problemlose Installation von Standardkomponenten, wie z.B. Webserver, Datenbankserver, FTP-Server oder VNC-Server ermöglicht. Darüber hinaus können Applikationen mittels verschiedener Programmiersprachen umgesetzt werden, wie z.B. C++, Java, Python oder PHP.

Aufbauend auf diesen Versuch, soll in einem zweiten Teil eine Smartphone-App zum Ein- und Ausschalten von Haushaltsgeräten entworfen und programmiert werden.

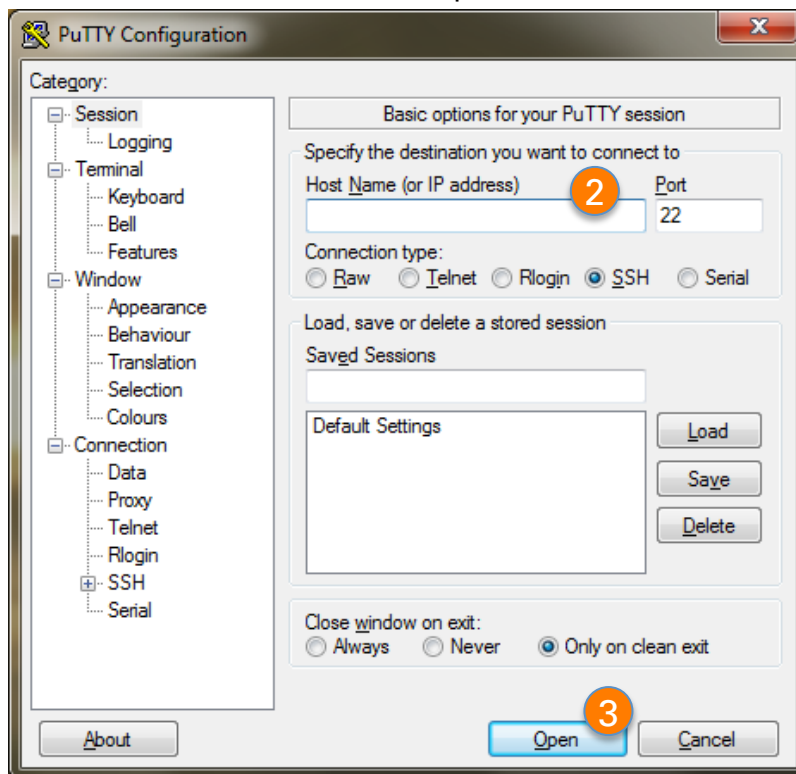
Versuchsdurchführung

Raspberry Pi starten

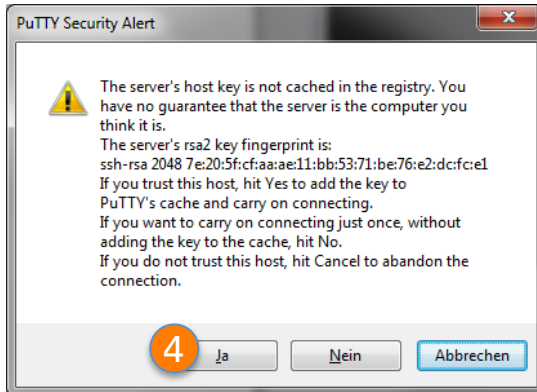
1. Schließen Sie den Raspberry Pi mittels des Cat.5-Kabels an den Switch an!
2. Vergewissern Sie sich, dass die SD-Karte fest im Raspberry Pi steckt!
3. Schließen Sie den Netzstecker des Raspberry Pi an eine Netzsteckdose an!

Verbindung mit Raspberry Pi herstellen

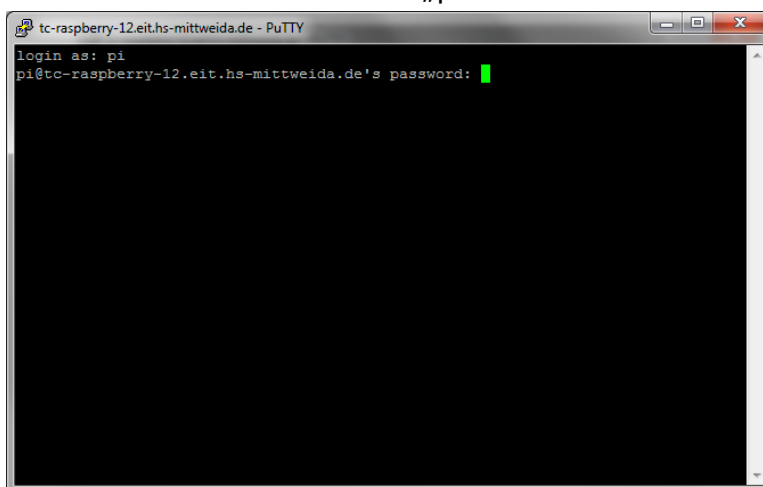
1. Starten Sie auf Ihren PC das Programm „PuTTY“!
2. Geben Sie im Feld „Host Name (or IP address)“ den aufgedruckten Hostname Ihres Raspberry Pi’s an, z.B. tc-raspberry-01.eit.hs-mittweida.de!
3. Klicken Sie anschließend auf „Open“!



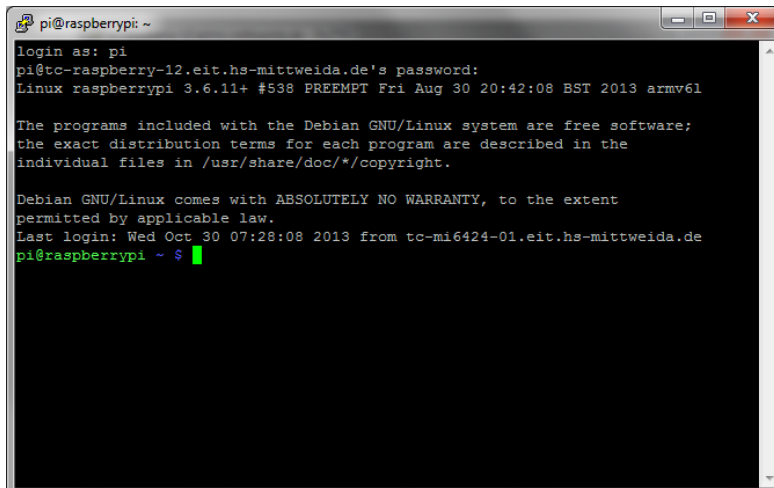
- Die anschließende Warnung können Sie mit „Ja“ bestätigen.



- Geben Sie bitte den Nutzernamen „pi“ ein und drücken Sie Enter!
- Geben Sie bitte das Passwort „praktikumkt“ ein und drücken Sie Enter!



- Sie sind nun mit der Kommandozeile des Raspberry Pi per SSH verbunden.



VNC-Server installieren

Um die grafische Oberfläche des Raspberry Pi auch ohne angeschlossenen Monitor nutzen zu können, soll als nächstes ein VNC-Server (Virtual Network Computing) auf dem Raspberry Pi eingerichtet werden.

VNC ist eine Software, die den Bildschirminhalt eines entfernten Rechners auf einem lokalen Rechner anzeigt und die lokalen Tastatur- und Mausbewegungen an den entfernten Rechner sendet.

1. Führen Sie vor der Installation ein Betriebssystemupdate durch, mittels:

```
$ sudo apt-get update
```

```
$ sudo apt-get upgrade
```

2. Installieren Sie die Software „tightvncserver“ mittels des Befehls:

```
$ sudo apt-get install tightvncserver
```

3. Konfigurieren Sie die Software durch den Befehl:

```
$ tightvncserver
```

Daraufhin erscheint eine Aufforderung zur Passworteingabe. Geben Sie hier bitte das Passwort „praktikumkt“ an! Dieses Passwort dient dem späteren Remotezugriff.

Geben Sie zur Verifizierung das Passwort „praktikumkt“ noch einmal ein.

Ein view-only Passwort wird für unsere Zwecke nicht benötigt. Bestätigen Sie diese Abfrage somit mit „n“!

4. Beenden Sie für die weitere Administration vorerst den VNC-Server mittels:

```
$ vncserver -kill :1
```

5. Standardmäßig wird der VNC-Server mit dem Systemstart nicht automatisch gestartet. Dies hat den Vorteil, dass die begrenzten Ressourcen des Raspberry Pi geschont werden.

Falls man dennoch einmal die grafische Oberfläche benötigt, genügt es daher den VNC-Server manuell zu starten. Hierfür muss ein Startscript im Heimverzeichnis des Nutzers „pi“ mit folgendem Befehl angelegt werden:

```
$ nano ~/startVNC.sh
```

Tragen Sie im nano-Editor folgende Zeilen ein:

```
#!/bin/sh
```

```
vncserver :1 -geometry 1280x720 -depth 24
```

6. Speichern Sie das Script durch drücken von Strg+x!
Bestätigen Sie die Abfrage mit „J“ und drücken Sie Enter!
7. Zum Abschluss muss das Startscript noch ausführbar gemacht werden. Dies erfolgt mittels des Befehls:

```
$ chmod +x ~/startVNC.sh
```

8. Starten Sie nun den VNC-Server mittels der Befehle:

```
$ cd ~
```

```
$ ./startVNC.sh
```

9. Starten Sie auf Ihrem PC das Programm „VNCViewer“!
10. Geben Sie den aufgedruckten Hostname Ihres Raspberry Pi's an und fügen Sie zusätzlich die Portnummer 5901 hinzu, z.B. tc-raspberry-01.eit.hs-mittweida.de:5901!

11. Drücken Sie Connect und geben Sie das Passwort „praktikumkt“ an!

Einrichten des Webservers

Als Webserver wird lighttpd eingesetzt, welcher eine schlanke Alternative zum, in Linux üblichen, Apache-Webserver darstellt. Lighttpd unterstützt dabei alle relevanten Funktionen, wie z.B. Auslieferung von Ressourcen per HTTP und HTTPS. Auch die Erweiterung um serverseitige Scriptsprachen wie z.B. PHP oder Python wird durch lighttpd unterstützt. Die Konfiguration erfolgt dabei über eine Konfig-Datei, welche im Verzeichnis `/etc/lighttpd/lighttpd.conf` zu finden ist.

1. Installieren Sie lighttpd mittels des Befehls:

```
$ sudo apt-get install lighttpd
```

2. Installieren Sie weiterhin die PHP-Erweiterung mittels der Befehle:

```
$ sudo apt-get install php5-common php5-cgi php5
```

```
$ sudo lighty-enable-mod fastcgi-php
```

```
$ sudo service lighttpd force-reload
```

3. Das Wurzelverzeichnis des Webservers besitzt nach der Installation nur Schreibrechte für den root-Nutzer. Damit auch der Webserver selbst, welcher mit dem Gruppen- und Nutzerbezeichner „www-data“ auftritt, Änderungsrechte besitzt, sind folgende Befehle auszuführen:

```
$ sudo chown -R www-data:www-data /var/www/html
```

```
$ sudo chmod 775 /var/www/html
```

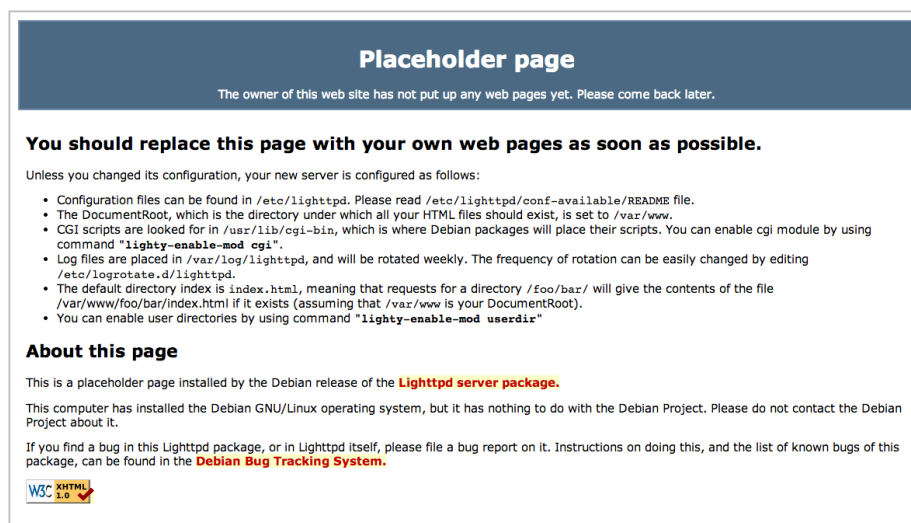
```
$ sudo usermod -a -G www-data www-data
```

```
$ sudo usermod -G www-data -a pi
```

4. Starten Sie nun den Webserver neu, um die Änderungen zu übernehmen!

```
$ sudo /etc/init.d/lighttpd restart
```

5. Starten Sie nun auf Ihrem PC einen Webbrowser und geben Sie als URL die auf dem Raspberry Pi aufgedruckte Adresse ein! Daraufhin müsste eine einfache Webseite erscheinen.



Placeholder page

The owner of this web site has not put up any web pages yet. Please come back later.

You should replace this page with your own web pages as soon as possible.

Unless you changed its configuration, your new server is configured as follows:


- Configuration files can be found in `/etc/lighttpd`. Please read `/etc/lighttpd/conf-available/README` file.
- The DocumentRoot, which is the directory under which all your HTML files should exist, is set to `/var/www`.
- CGI scripts are looked for in `/usr/lib/cgi-bin`, which is where Debian packages will place their scripts. You can enable cgi module by using command `"lighty-enable-mod cgi"`.
- Log files are placed in `/var/log/lighttpd`, and will be rotated weekly. The frequency of rotation can be easily changed by editing `/etc/logrotate.d/lighttpd`.
- The default directory index is `index.html`, meaning that requests for a directory `/foo/bar/` will give the contents of the file `/var/www/foo/bar/index.html` if it exists (assuming that `/var/www` is your DocumentRoot).
- You can enable user directories by using command `"lighty-enable-mod userdir"`

About this page

This is a placeholder page installed by the Debian release of the **Lighttpd server package**.

This computer has installed the Debian GNU/Linux operating system, but it has nothing to do with the Debian Project. Please do not contact the Debian Project about it.

If you find a bug in this Lighttpd package, or in Lighttpd itself, please file a bug report on it. Instructions on doing this, and the list of known bugs of this package, can be found in the **Debian Bug Tracking System**.



6. Erzeugen Sie im Verzeichnis `/var/www/html` eine PHP-Datei mit der Bezeichnung „phpinfo.php“!

```
$ sudo nano /var/www/html/phpinfo.php
```

7. Fügen Sie in diese Datei folgenden Inhalt ein:

```
<?php
    phpinfo();
?>
```

8. Speichern Sie die Datei mittels „Strg+x“ und bestätigen Sie die Abfrage mit „J“!

9. Geben Sie in Ihrem Webbrowser als URL die Adresse Ihres Raspberry Pi ein und fügen Sie anschließend den Bezeichner „/phpinfo.php“ hinzu:


z.B.: <http://tc-raspberry-01.hs-mittweida.de/phpinfo.php>

Als Ergebnis sehen Sie eine Übersicht der Installierten PHP-Version. Würde PHP auf dem Webserver nicht richtig funktionieren, würde an dieser Stelle nur der Quellcode der Datei „phpinfo.php“ angezeigt werden.



System	Linux raspberrypi 3.6.11+ #538 PREEMPT Fri Aug 30 20:42:08 BST 2013 armv6l
Build Date	Oct 13 2013 04:09:29
Server API	CGI/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php5/cgi
Loaded Configuration File	/etc/php5/cgi/php.ini
Scan this dir for additional .ini files	/etc/php5/cgi/conf.d
Additional .ini files parsed	/etc/php5/cgi/conf.d/10-pdo.ini
PHP API	20100412
PHP Extension	20100525
Zend Extension	220100525
Zend Extension Build	API220100525,NTS
PHP Extension Build	API20100525,NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	disabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	disabled
Registered PHP Streams	https, ftps, compress.zlib, compress.bzip2, php, file, glob, data, http, ftp, phar, zip
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, tls
Registered Stream Filters	zlib.*, bzip2.*, convert.iconv.*, string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, dechunk

This program makes use of the Zend Scripting Language Engine:
Zend Engine v2.4.0, Copyright (c) 1998-2012 Zend Technologies

Powered By 

Passwort geschützter Zugang für Webserver einrichten

Um einen unberechtigten Zugang auf den Webserver zu unterbinden, soll das Wurzelverzeichnis des Webserver (`/var/www/html`) mit einer Basic-Authentikation gesichert werden. Hierfür ist die Konfiguration des `lighttpd`-Webserver notwendig, was über die Konfigurationsdatei `/etc/lighttpd/lighttpd.conf` erfolgt.

1. Öffnen Sie die Konfigurationsdatei im Editor

```
$ sudo nano /etc/lighttpd/lighttpd.conf
```

2. Fügen Sie das Modul „`mod_auth`“ der „`server.modules`“-Liste hinzu!

```
server.modules = ( "mod_access",  
                  "mod_alias",  
                  "mod_compress",  
                  "mod_redirect",  
                  "mod_rewrite",  
                  "mod_auth", )
```

3. Definieren Sie die zu verwendende Authentifizierungsmethode. Fügen Sie hierfür in der letzten Zeile folgenden Inhalt hinzu:

```
auth.debug = 2  
auth.backend = "plain"  
auth.backend.plain.userfile = "/home/pi/www/password"
```

4. Zusätzlich muss noch das Verzeichnis festgelegt werden, welches per Passwort zu schützen ist. Hierfür soll das Wurzelverzeichnis des Webserver angegeben werden. Fügen Sie daher in der letzten Zeile folgenden Inhalt hinzu:

```
auth.require = ( "" =>  
                ("method"=>"basic",  
                 "realm"=>"Password protected area",  
                 "require"=>"user=praktikum"  
                ))
```

5. Speichern Sie die Datei mittels „`Strg+x`“ und bestätigen Sie die Abfrage mit „`J`“!
6. Legen Sie im Heimverzeichnis des Nutzers „`pi`“ einen neuen Ordner „`www`“ an! (Generell wäre aber auch jeder andere Pfad möglich.)

```
$ mkdir ~/www
```

7. Legen Sie nun im Verzeichnis `/home/pi/www/password` die Passwort-Datei an! Diese Datei beinhaltet den Nutzernamen und das zugehörige Passwort, welche durch ein „`:`“ voneinander getrennt werden.

```
$ sudo nano /home/pi/www/password
```

8. Geben Sie folgenden Inhalt ein und speichern Sie anschließend die Datei mittels „`Strg+x`“ und bestätigen Sie die Abfrage mit „`J`“!

```
praktikum:praktikumkt
```

9. Damit die Datei vom Webserver gelesen werden kann, müssen noch die erforderlichen Rechte für die Gruppe und den Nutzer `www-data` eingestellt

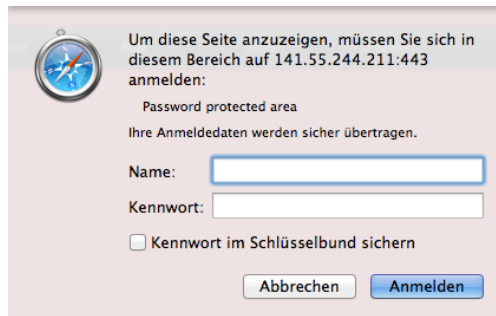
werden:

```
$ sudo chown www-data:www-data /home/pi/www/password
```

10. Starten Sie zur Übernahme der durchgeführten Änderungen den Webserver neu!

```
$ sudo /etc/init.d/lighttpd restart
```

11. Rufen Sie, zum Test, in Ihrem Webserver noch einmal die Ressource „phpinfo.php“ auf! Als Ergebnis müssten Sie nun eine Eingabeaufforderung für Nutzernamen und Passwort erhalten. Geben Sie als Nutzernamen „praktikum“ und als Passwort „praktikumkt“ ein!



Um diese Seite anzuzeigen, müssen Sie sich in diesem Bereich auf 141.55.244.211:443 anmelden:

Password protected area
Ihre Anmeldedaten werden sicher übertragen.

Name:

Kennwort:

Kennwort im Schlüsselbund sichern

SSL-Verschlüsselung für die HTTP-Verbindung

Im vorherigen Abschnitt haben Sie den Zugang auf den Webserver mit einem Passwort gesichert. Die mit HTTP übertragenen Daten werden allerdings immer noch unverschlüsselt transportiert. Dies gilt auch für die Anmeldesequenz, wodurch Nutzernamen und Kennwörter ebenfalls im Klartext übertragen werden. Um das Mitlesen der gesendeten Daten zu unterbinden ist es sinnvoll, auf eine HTTPS-Verbindung umzustellen. Normalerweise sollten die für HTTPS notwendigen Zertifikate durch eine registrierte Zertifizierungsinstanz bestätigt werden. Hierfür sind allerdings u.a. Angaben zum Domain-Bezeichner notwendig. Darüber hinaus benötigt der Host einen Fully Qualified Domain Name (FQDN), wie z.B. www.homecontrol-raspberry.de. Dies würde demzufolge die Beantragung eines Domainbezeichners erfordern, welcher mit Kosten verbunden wäre.

Für den privaten Einsatz von HTTPS reicht es allerdings aus, ein selbst-signiertes Zertifikat zu verwenden, was in den folgenden Schritten durchgeführt werden soll. Dies führt allerdings zu einer Warnung im Webbrowser, dass die Identität des Webserver nicht bestätigt ist. Dennoch wird aber die gesamte Kommunikation erfolgreich verschlüsselt.

1. Wechseln Sie in das Verzeichnis des Webserver `/etc/lighttpd`:

```
$ cd /etc/lighttpd
```

2. Erzeugen Sie ein neues Verzeichnis „ssl“!

```
$ sudo mkdir ssl
```

3. Um ein SSL-Zertifikat zu erstellen, wird die Software OpenSSL benötigt. Installieren Sie daher vorerst OpenSSL mittels folgenden Befehls:

```
$ sudo apt-get install openssl
```

4. Wechseln Sie in das Verzeichnis „ssl“!

```
$ cd ssl
```

5. Erstellen Sie ein Zertifikat mittels folgenden Befehls und folgen Sie den Eingabeaufforderungen!

Achtung: Das Symbol „↵“ kennzeichnet nur einen Zeilenumbruch in dieser Anleitung. Es handelt sich demzufolge um einen Befehl, der in einer Zeile, ohne das Symbol „↵“, anzugeben ist!

```
$ sudo openssl req -new -x509 -keyout server.pem -out ↵  
server.pem -days 365 -nodes
```

```
Country Name (2 letter code) [AU]:DE  
State or Province Name (full name) [Some-State]:Sachsen  
Locality Name (eg, city) []:Mittweida  
Organization Name (eg, company) [Internet Widgits Pty Ltd]:homecontrol-raspberry.de  
Organizational Unit Name (eg, section) []:  
Common Name (e.g. server FQDN or YOUR name) []:www.homecontrol-raspberry.de  
Email Address []:kontakt@homecontrol-raspberry.de
```

6. Im Verzeichnis „ssl“ befindet sich nun eine Datei server.pem, welche dem Webserver nun noch als Server-Zertifikat angegeben werden muss. Öffnen Sie daher wieder die Konfigurationsdatei des Webserver!

```
$ sudo nano /etc/lighttpd/lighttpd.conf
```

7. Ändern Sie die Port-Nummer für den WWW-Server von 80 auf 443!

```
server.port = 443
```

8. Aktivieren Sie für den Webserver SSL und geben Sie den Pfad zum SSL-Zertifikat an!

Fügen Sie hierfür unter die „server.port“-Zeile folgenden Inhalt ein:

```
ssl.engine = "enable"  
ssl.pemfile = "/etc/lighttpd/ssl/server.pem"
```

9. Speichern Sie die Datei mittels „Strg+x“ und bestätigen Sie die Abfrage mit „J“!

10. Starten Sie zur Übernahme der Änderungen den Webserver neu!

```
$ sudo service lighttpd restart
```

11. Testen Sie die Funktionalität der HTTPS-Verbindung durch Aufruf der URL <https://<raspberrypi-adresse>/phpinfo.php>! Ersetzen Sie den Bezeichner <raspberrypi-adresse> durch die aufgedruckte Adresse Ihres Raspberry Pi's. Der Webserver zeigt Ihnen daraufhin die bereits besagte Warnung an. Bestätigen Sie daher die Warnung!

z.B.



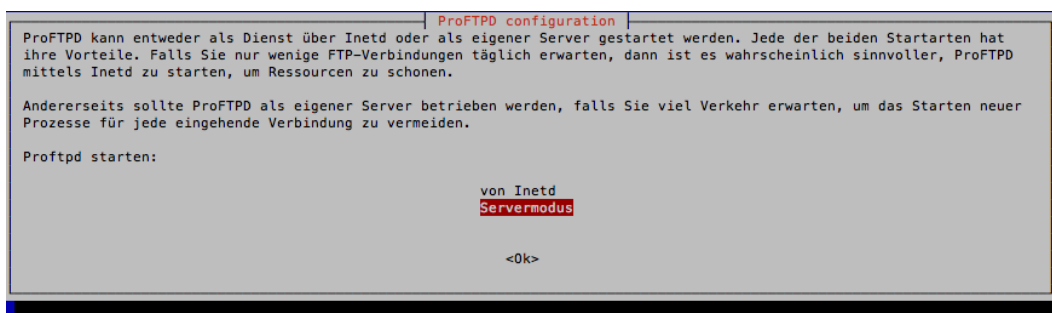
Einrichten eines FTP-Servers

Um einen komfortablen Dateitransfer zwischen Webserver und PC zu realisieren, soll der FTP-Server „Proftpd“ installiert werden. Hierüber können dann beispielsweise PHP-Dateien am PC programmiert und anschließend auf den Raspberry Pi kopiert werden.

1. Installieren Sie Proftpd mittels folgenden Befehl!

```
$ sudo apt-get install proftpd
```

2. Wählen Sie im erscheinenden Dialog „Servermodus“ und bestätigen Sie mit Enter!



3. Die Konfiguration des FTP-Servers erfolgt über die Datei `/etc/proftpd/proftpd.conf`. Hier müssen die freizugebenden Verzeichnisse und die Nutzerverwaltung definiert werden.

Öffnen Sie daher die Datei `/etc/proftpd/proftpd.conf` mittels:

```
$ sudo nano /etc/proftpd/proftpd.conf
```

4. In der geöffneten Datei muss folgender Inhalt vor der Zeile `„# RequireValidShell off“` eingefügt werden:

```
DefaultRoot ~
AuthOrder      mod_auth_file.c mod_auth_unix.c
AuthUserFile   /etc/proftpd/ftpd.passwd
AuthPAM off
RequireValidShell off
```

5. Speichern Sie die Datei mittels „Strg+x“ und bestätigen Sie die Abfrage mit „J“!
6. Legen Sie nun einen virtuellen Nutzer an, welcher für die Anmeldung am FTP-Server genutzt werden soll!

Wechseln Sie hierfür in das Proftpd-Verzeichnis:

```
$ cd /etc/proftpd
```

7. Damit die durch den virtuellen Nutzer erzeugten Dateien auch durch den Webserver aufgerufen werden können, soll dieser als „www-data“-Nutzer auftreten. Hierfür muss vorerst die uid des „www-data“-Nutzers ermittelt werden.

```
$ id www-data
```

```
→ pi@raspberrypi /etc/proftpd $ id www-data  
uid=33(www-data) gid=33(www-data) Gruppen=33(www-data)
```

8. Jetzt soll der virtuelle Nutzer „praktikum“ mittels folgenden Befehls angelegt werden. Achten Sie dabei darauf, dass Sie den im vorherigen Befehl ausgegebenen uid- und gid-Wert nutzen. Im hier vorliegenden Beispiel entspricht dies dem Wert „33“. Geben Sie bei der Aufforderung zur Passworteingabe das Kennwort „praktikumk“ an!

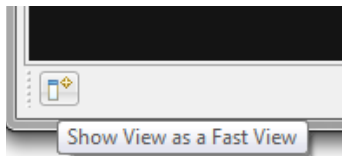
Achtung: Das Symbol „↵“ kennzeichnet nur einen Zeilenumbruch in dieser Anleitung. Es handelt sich demzufolge um einen Befehl, der in einer Zeile, ohne das Symbol „↵“, anzugeben ist!

```
$ sudo ftpasswd --passwd --name praktikum --uid 33 --gid 33↵  
--home /var/www/html --shell /bin/false
```

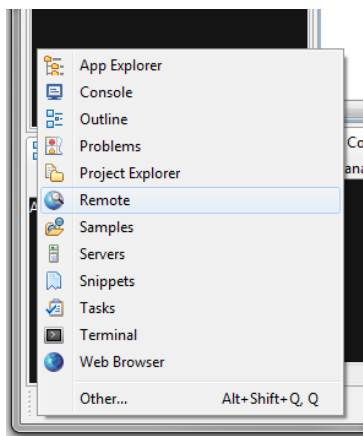
9. Starten Sie nun den FTP-Server neu!

```
$ sudo /etc/init.d/proftpd restart
```

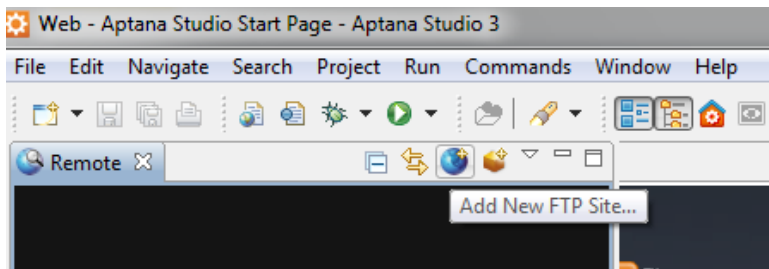
10. Öffnen Sie die Software „Aptana“ auf Ihrem Rechner!
Drücken Sie den links unten befindlichen Button.



Wählen Sie „Remote“ aus!



Wählen Sie „Add New FTP Site...“!



Tragen Sie im folgenden Dialog Ihre Verbindungsdaten ein:

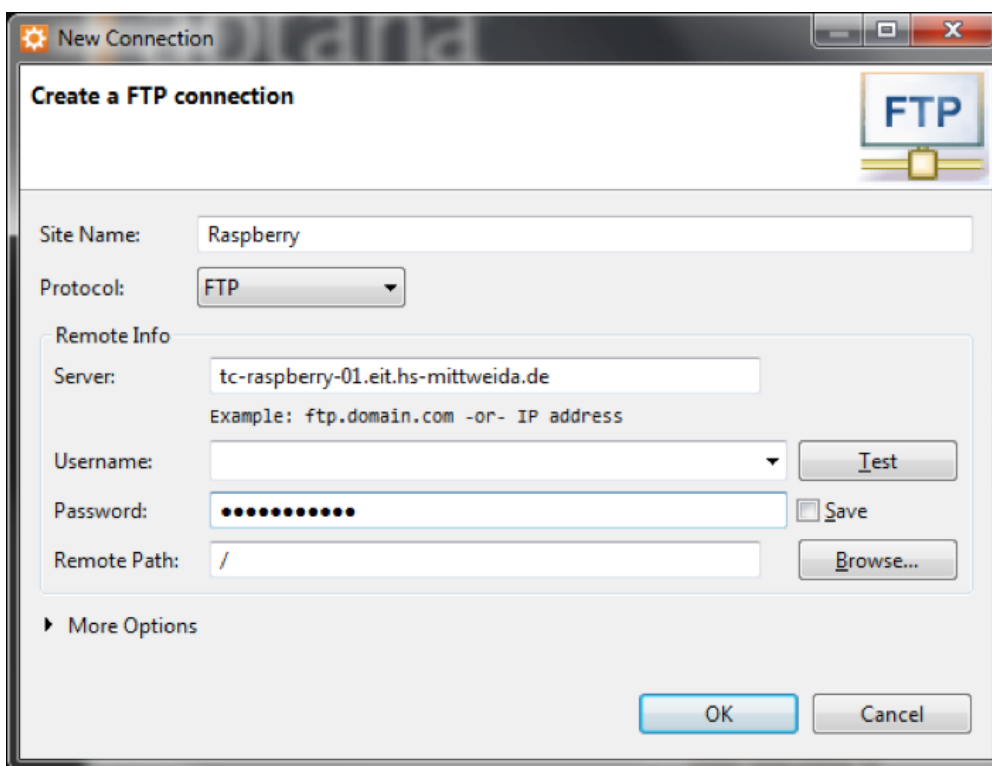
Site Name: *Raspberry*

Server: „aufgedruckte Adresse Ihres Raspberry Pi's“

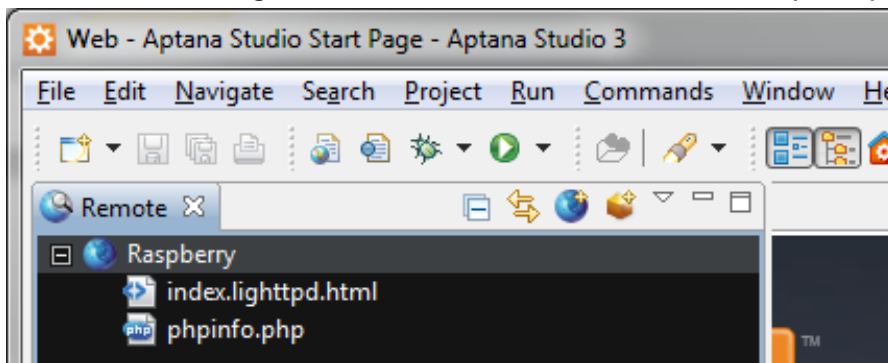
Username: *praktikum*

Password: *praktikumkt*

Wählen Sie anschließend „OK“



Sie haben nun Zugriff auf das Wurzelverzeichnis des Raspberry-Webservers.



Schalten einer Funksteckdose mittels Raspberry-Pi

Über die GPIO-Ausgänge des Raspberry Pi besteht die Möglichkeit verschiedenste Geräte anzuschalten. Für diesen Versuch wurde am Raspberry Pi ein 433MHz-Sender angeschlossen. Über diesen Sender besteht die Möglichkeit handelsübliche Funksteckdosen ein- bzw. auszuschalten. Um dies zu realisieren, ist eine Software notwendig, welche wie folgt zu installieren ist.

1. Führen Sie eine Aktualisierung des Raspberry Pi Betriebssystems durch! Rufen Sie hierfür folgende Befehle auf (kann ca. 10-15min dauern):

```
$ sudo apt-get update && sudo apt-get upgrade -y
```

2. Installieren Sie die benötigte Software „Wiring Pi“ mittels folgender Befehle:
Achtung: Das Symbol „↵“ kennzeichnet nur einen Zeilenumbruch in dieser Anleitung. Es handelt sich demzufolge um einen Befehl, der in einer Zeile, ohne das Symbol „↵“, anzugeben ist!

```
$ cd ~
```

```
$ wget http://www.staff.↵
```

```
hs-mittweida.de/~rthomane/praktikumkt/raspberry/wiringPi.zip
```

```
$ tar xvf wiringPi.zip
```

```
$ cd wiringPi
```

```
$ ./build
```

3. Versuchen Sie nun eine Funksteckdose mit dem Raspberry Pi ein- bzw. auszuschalten!

Hierfür wird das Programm „send“ verwendet, welches im Verzeichnis „/home/pi/wiringPi/433Utils/RPi_utils/“ zu finden ist. Das Programm „send“ erfordert hierbei folgende Parameter:

- die GPIO (**G**eneral **P**urpose **I**nput/**O**utput) Nummer, wo der 433MHz-Sender angeschlossen ist:

→ Nutzen Sie hierfür den Wert **14**

- der Systemcode, der in der Steckdose eingestellt ist:

→ Nutzen Sie hierfür die Nummer Ihres Raspberry Pi in binärer Darstellung (5 Bit), z.B.

→ z.B. tc-raspberry-12.eit.hs-mittweida → 12d = **01100**

- der Unitcode, der die Funksteckdose spezifiziert:

→ Nutzen Sie hierfür den Wert **1**

Für den Raspberry Pi mit der Nummer 12 sieht der Programmaufruf dann beispielsweise wie folgt aus:

Einschalten:

```
$ sudo ~/wiringPi/433Utils/RPi_utils/send 14 01100 1 1
```

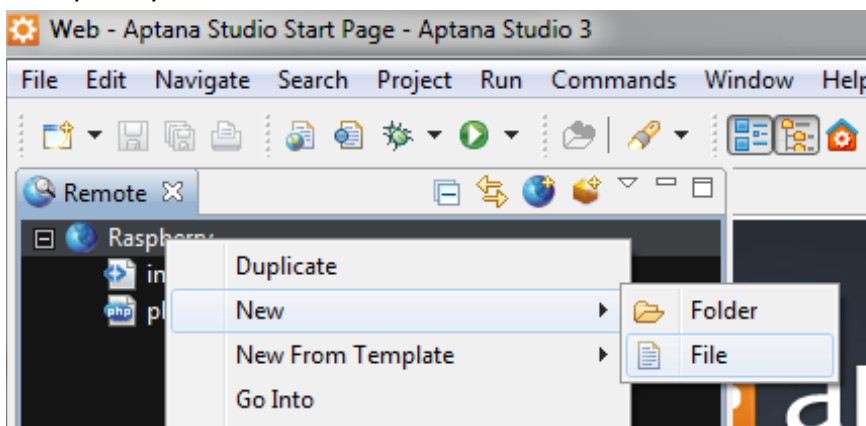
Ausshalten:

```
$ sudo ~/wiringPi/433Utils/RPi_utils/send 14 01100 1 0
```

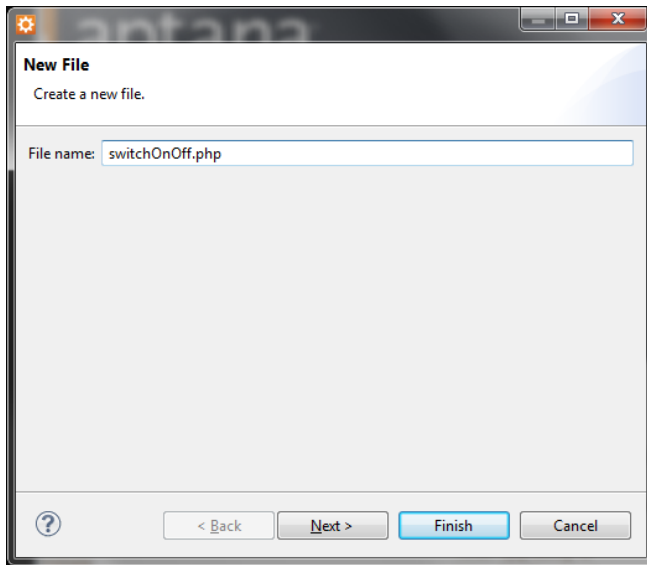
Funksteckdose mittels HTTP ein- bzw. ausschalten

Das Ein- bzw. Ausschalten der Funksteckdose funktioniert bis jetzt nur direkt vom Raspberry Pi aus. Wünschenswert wäre allerdings auch eine Ansteuerung von jedem , am Netzwerk angeschalteten, Gerät. Um dies zu ermöglichen, muss der Raspberry Pi Steuerbefehle über das Netzwerk empfangen und daraufhin die Funksteckdose schalten. Um möglichst viele verschiedene Geräte zu unterstützen, sollte man zur Kommunikation zwischen Raspberry Pi und Steuergerät ein standardisiertes Protokoll verwenden, welches auf einer Server – Client – Architektur basiert. Ein weltweit standardisiertes Protokoll stellt dabei das HTTP (Hypertext Transfer Protocol) dar. Der Raspberry Pi müsste dabei die Rolle des Servers übernehmen, was durch den bereits installierten Webserver gegeben ist. Der auf einem entfernten Rechner installierte Webbrowser würde dann die Rolle des Clients übernehmen. Das Schalten der Funksteckdose kann dann ganz bequem über den Aufruf einer URL erfolgen. Um den Webserver zum Ausführen des „send“-Programms zu bewegen, kann dieser Befehl in einer PHP-Datei untergebracht werden.

1. Öffnen Sie das Programm „Aptana“!
2. Öffnen Sie die vorhin eingerichtete FTP-Verbindung „Raspberry“!
3. Erzeugen Sie eine neue Datei, indem Sie mit der rechten Maustaste auf „Raspberry“ klicken und dann New→File auswählen!



4. Benennen Sie die Datei als „switchOnOff.php“ und Drücken Sie auf „Finish“!



5. Fügen Sie in die Datei „switchOnOff.php“ folgenden Inhalt ein und speichern Sie die Datei!

```
<?
$gpio=$_GET['gpio'];
$systemcode = $_GET['systemcode'];
$unitcode = $_GET['unitcode'];
$onoff = $_GET['onoff'];
echo @shell_exec("sudo
/home/pi/wiringPi/433Utils/RPi_utils/send $gpio $systemcode
$unitcode $onoff -r 3");
?>
```

6. Nun muss für den Webserver noch die Berechtigung zum Ausführen von Shell-Befehlen festgelegt werden. Öffnen Sie hierfür bitte die Datei `/etc/sudoers` im Editor mittels:

```
$ sudo visudo
```

7. Fügen Sie folgenden Inhalt als letzte Zeile ein:

```
www-data ALL=(ALL) NOPASSWD: ALL
```

8. Speichern Sie die Datei mittels „Strg+x“ und Bestätigen Sie die Abfrage mit „J“!

9. Starten Sie den Webserver neu!

```
$ sudo /etc/init.d/lighttpd restart
```

10. Öffnen Sie einen Webbrowser auf Ihrem PC und geben Sie die URL zur switchOnOff.php-Datei an! Achten Sie darauf, den für Ihre Steckdose eingestellten Systemcode und Unitcode zu verwenden.

z.B. <https://tc-raspberry-01.eit.hs-mittweida.de/switchOnOff.php?gpio=14&systemcode=11000&unitcode=1&onoff=0>

Sie sollten nun in der Lage sein, die Funksteckdose mittels des Webbrowsers ein- bzw. auszuschalten. Hierfür müssen Sie für den Parameter „onoff“ den Wert „0“ oder „1“ angeben!

Temperatur auslesen

Am Ihrem Raspberry Pi ist weiterhin ein Temperatur-Sensor angeschlossen. Die Sensoren werden im Betriebssystem des Raspberry Pi wie eine Datei behandelt. Der Temperatur-Sensor ist hierbei im Verzeichnis „/sys/bus/w1/devices/“ zu finden. Für jeden am Raspberry Pi angeschlossenen 1-wire-Sensor, existiert unter „/sys/bus/w1/devices/“ ein eigenes Verzeichnis. Dieses Verzeichnis entspricht der digitalen Kennung des Sensors, z.B. 28-000004f19477. Innerhalb des Sensor-Verzeichnisses existiert dann eine Datei „w1_slave“, welche die aktuelle Temperatur beinhaltet.

1. Finden Sie die digitale Kennung Ihres Temperatur-Sensors heraus! Rufen Sie hierfür folgenden Befehl auf!

```
$ ls -l /sys/bus/w1/devices/
```

Als Ergebnis erscheint:

```
insgesamt 0
lrwxrwxrwx 1 root root 0 Nov 13 14:03 28-000004f19477 -> ../../../../devices/w1_bus_master1/28-000004f19477
lrwxrwxrwx 1 root root 0 Nov 13 13:48 w1_bus_master1 -> ../../../../devices/w1_bus_master1
```

Sie sehen als Ergebnis eine Auflistung aller Verzeichnisse innerhalb von „/sys/bus/w1/devices“. Der Verzeichnisname des Temperatur-Sensors beginnt dabei immer mit einer Ziffer. In vorliegendem Beispiel ist der angeschlossene Temperatur-Sensor demzufolge das Verzeichnis 28-000004f19477.

2. Zeigen Sie sich die aktuelle Temperatur an! Rufen Sie hierfür die Datei „w1_slave“ im Verzeichnis des Temperatur-Sensors auf! Achten Sie darauf, den Verzeichnisname Ihres Temperatur-Sensors zu verwenden!

```
$ more /sys/bus/w1/devices/28-000004f19477/w1_slave
```

Als Ergebnis erscheint:

```
pi@raspberrypi /sys/bus/w1/devices/28-000004f19477 $ more w1_slave
63 01 4b 46 7f ff 0d 10 15 : crc=15 YES
63 01 4b 46 7f ff 0d 10 15 t=22187
```

Datenübertragung erfolgreich

Temperatur/1000

Wie in der Grafik ersichtlich, wird die Temperatur mit $t=22187$ angezeigt. Dieser Wert muss noch durch 1000 geteilt werden. Daher würde sich im vorliegenden Beispiel die Temperatur 22.187°C ergeben.

3. Schreiben Sie sich den Verzeichnisname Ihres Temperatur-Sensors auf!

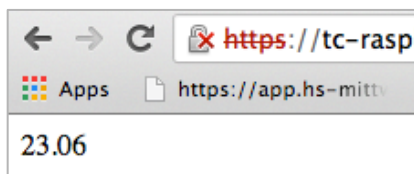
Temperatur per HTTPS zur Verfügung stellen

Ziel soll es sein, die Raumtemperatur auch über den Webserver anzubieten.

1. Erzeugen Sie daher in Aptana eine neue PHP-Datei „getTemperature.php“!
2. Fügen Sie folgenden Inhalt in die Datei „getTemperature.php“ ein! Achten Sie darauf den Verzeichnisname Ihres Temperatur-Sensors zu nutzen!

```
<?
//Sensor-File öffnen
$fp = fopen('/sys/bus/w1/devices/28-000004f19477/w1_slave', 'r');
//File auslesen
while(!feof($fp)){
    $tempText .= fread($fp, 1024);
}
//Text anhand des Textes "t=" in ein Array wandeln
$tempArray = split("t=", $tempText);
//Temperatur-Wert durch 1000 teilen und runden
$temperatur = round(floatval($tempArray[1])/1000,2);
//Temperatur ausgeben
echo $temperatur;
?>
```

3. Speichern Sie die Datei!
4. Öffnen Sie einen Webbrowser und rufen Sie die Datei „getTemperature.php“ mittels HTTPS-Request auf!



Als Ergebnis wird die aktuelle Temperatur angezeigt.

Raspberry Pi über HTTP neu starten bzw. ausschalten

Nach Durcharbeitung der vorhergehenden Schritte steht uns nun eine einfache Home Control-Lösung zur Verfügung. Weitere Sensoren oder Steuergeräte können nach den gleichen oder ähnlichen Konzepten abgefragt oder angesteuert werden. Aufbauen auf diesen Versuch, soll in einem zweiten Teil eine einfache Smartphone-App programmiert werden, welche dann zum Schalten der Steckdose(n) bzw. zur Anzeige der Raumtemperatur genutzt werden kann.

Da der Raspberry Pi keinen Reset- bzw. Ein/Ausschaltknopf besitzt, soll auch ein neu starten bzw. ausschalten über einen HTTPS-Request realisiert werden.

1. Öffnen Sie Aptana und legen Sie 2 neue Dateien „restart.php“ und „shutdown.php“ an!

2. Fügen Sie in die Datei „restart.php“ folgenden Inhalt ein und speichern Sie anschließend die Datei!

```
<?
echo @shell_exec("sudo shutdown -r now");
echo "Raspberry Pi wird neu gestartet";
?>
```

3. Fügen Sie in die Datei „shutdown.php“ folgenden Inhalt ein und speichern Sie anschließend die Datei!

```
<?
echo @shell_exec("sudo shutdown -h now");
echo "Raspberry Pi wird ausgeschaltet";
?>
```

4. Öffnen Sie in Ihrem Webbrowser und rufen Sie die Datei „restart.php“ auf!
→ Ihr Raspberry Pi müsste sich nun neu starten.
5. Rufen Sie nach dem Neustart in Ihrem Webbrowser die Datei „shutdown.php“ auf!
→ Ihr Raspberry Pi müsste sich nun ausschalten. Wenn nur noch die rote LED leuchtet, können Sie den Raspberry Pi vom Stromnetz nehmen.

Durch diese beiden PHP-Dateien kann nun der Raspberry Pi komfortabel ausgeschaltet bzw. neu gestartet werden.