

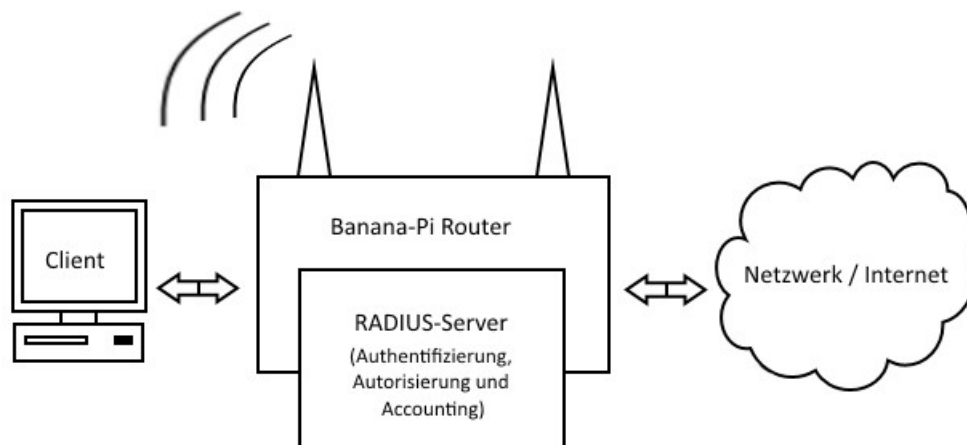
## 3.2 Radius

### 3.2.1 Allgemeines

Der Remote Authentication Dial-In User Service (RADIUS, deutsch Authentifizierungsdienst für sich einwählende Benutzer) ist ein Protokoll zwischen Benutzer und Server, das für die 3 A's (dem sogenannten Tripple-A-System), also der Authentifizierung, Autorisierung und das Accounting zuständig ist.

Laut Internetquellen ist es der „De-facto Standard bei der zentralen Authentifizierung von Einwahlverbindungen über Modem, ISDN, VPN, WLAN (IEEE 802.1X) und DSL“.<sup>[1]</sup>

### 3.2.2 Funktionsweise



Anhand der obenstehenden Abbildung lässt sich gut erkennen wie der Aufbau ist. Der Client (Smartphone, PC, etc.) kann sich wahlweise per WLAN oder LAN mit dem Banana-Pi Router verbinden, dabei sendet er eine Authentifizierungsanfrage an diesen, welche der Router an den Radius-Server, welcher auf dem Banana-Pi läuft, weiterleitet. Dieser verarbeitet nun die Anfrage indem er, je nach Konfiguration, in unserem Fall über eine SQL-Datenbank überprüft ob der Client berechtigt ist dem Netzwerk beizutreten. Zusätzlich kann diese Verbindung dann auch limitiert werden (Volumen-Limit, Bandbreiten-Drosselung, beschränkter Zugriff auf Subnetze/VLANs).

### 3.2.3 FreeRADIUS

FreeRADIUS ist wie der Name schon vermuten lässt eine freie und kostenlose Implementierung des RADIUS-Protokolls und unter der GNU General Public License, version 2 lizenziert. Es ist laut eigenen Angaben der weltweit am meisten eingesetzte RADIUS-Server und wird von den meisten Internetdiensteanbietern (Providern) sowie den 500 umsatzstärksten Unternehmen der Welt benutzt.<sup>[2]</sup>

Es findet außerdem auch in der Hochschule sowie im akademischen Forschungsnetzwerk Eduroam Einsatz. Es unterstützt den meistverbreiteten Authentifizierungsstandard EAP, auf deutsch etwa

„Erweiterbares Authentifizierungsprotokoll“<sup>[3]</sup>, der ca. 40 verschiedene Verfahren anbietet, welche heutzutage unter anderem in den Sicherheitsimplementationen WPA und WPA2 Verwendung finden.

### 3.2.3.1 Installation

FreeRADIUS ist auch in den offiziellen Paket-Quellen von Debian enthalten und kann somit ganz einfach über den Debian-Paketmanager apt-get installiert werden. Als erstes stellen wir sicher dass das System up-to-date ist und die aktuellen Paketquellen besitzt.<sup>[4]</sup>

```
sudo apt-get update
sudo apt-get upgrade
```

```
sudo apt-get freeradius freeradius-utils freeradius-mysql mysql-
server
```

Bei der Installation des MySQL-Servers wird man gebeten das Passwort für den Root-User zu setzen und zu bestätigen (in unserem Fall ist dies *bananapi*). Im nächsten Schritt wird eine Datenbank für RADIUS angelegt.

```
mysql -uroot -p

CREATE DATABASE radius;
GRANT ALL PRIVILEGES ON radius.* TO root@localhost IDENTIFIED BY
"bananapi";
flush privileges;
exit
```

Danach noch die SQL Schemas in die Datenbank laden:<sup>[5]</sup>

```
mysql -uroot -p radius < /etc/freeradius/sql/main/mysql/schema.sql
mysql -uroot -p radius < /etc/freeradius/sql/main/mysql/setup.sql
```

Zukünftig können nun Benutzer mit folgendem MySQL-Befehl angelegt werden:

```
insert      into      radcheck      (username,attribute,op,value)
values("USERNAME", "Cleartext-Password", ":", "PASSWORD");
```

Nun muss RADIUS dafür konfiguriert werden, das SQL-Modul zu benutzen, dafür editiert man die Datei */etc/freeradius/sql.conf* und trägt im Bereich *#Connection info* die korrekten Login-Daten ein. Desweiteren muss man in der Datei *radius.conf* die Zeile *\$INCLUDE sql.conf*, sowie in der Datei */etc/freeradius/sites-available/default* zwei mal *sql* in den Sektionen *authorize {* und *accounting {* ein kommentieren.

### 3.2.3.2 Testen

Um die Konfiguration zu überprüfen kann der RADIUS-Server im Debugging-Modus gestartet werden. Dazu sollte der RADIUS-Dienst erst beendet werden:

```
service freeradius stop
freeradius -X
```

Danach kann in einem zweiten Terminal (oder ggf. mit Terminal-Multiplexer) das Programm radtest verwendet werden um Authentifizierungsanfragen an den RADIUS-Server zu senden:<sup>[6]</sup>

```
radtest USERNAME PASSWORD 127.0.0.1 0 mysecret
```

Eine typische Ausgabe einer erfolgreichen Anfrage sollte wie folgt aussehen:

```
root@bananapi ~ # radtest mysqluser1 "testpass" localhost 1812 testing123
Sending Access-Request of id 140 to 127.0.0.1 port 1812
    User-Name = "mysqluser1"
    User-Password = "testpass"
    NAS-IP-Address = 141.28.60.191
    NAS-Port = 1812
    Message-Authenticator = 0x00000000000000000000000000000000
rad_recv: Access-Accept packet from host 127.0.0.1 port 1812, id=140, length=20
```

## Quellen

- [1] [http://i.techrepublic.com/downloads/PDF/SolutionBase\\_RADIUS\\_deployment\\_scenarios.pdf](http://i.techrepublic.com/downloads/PDF/SolutionBase_RADIUS_deployment_scenarios.pdf)
- [2] <http://freeradius.org/>
- [3] <https://www.heise.de/glossar/entry/Extensible-Authentication-Protocol-397255.html>
- [4] <https://www.vultr.com/docs/install-freeradius-on-debian-7>
- [5] <https://wiki.freeradius.org/guide/SQL-HOWTO-for-freeradius-3.x-on-Debian-Ubuntu>
- [6] <https://extremeshok.com/5486/debian-7-freeradius-server-mysql-authentication/>