Ruckus™ Simply Better Wireless. | BROCADE

WISPr integration

Karthikeyan Krish

Systems Engineer - South India, Sri Lanka and Maldives

Nov 2020

Wi-Fi Perfected

Introduction

- Background
  - DG Vaishnav, Chennai is an education institution was looking for a WLAN solution with self-service captive portal using their existing SQL server(MSSQL) and SMS authentication.
  - Aruba quoted Clearpass and Instant. We can't pitch our Cloudpath due to non-availability of SQL data source option.
  - I explored wispr-appnote but the data was outdated and basic in it.
  - Tested locally in homelab and presented to partner who expanded the SQL and SMS part.
  - Finally, solution was upsold to ZD instead of Unleashed.
- Usage
  - The content is available in ruckushtml.zip file and a presentation to explain how to use it.
  - If you have any partner who wants to run their own captive portal, the ruckushtml.zip file will help kickstart. It is PHP based. Partners are welcome to convert to any other web scripting language.

Requirements

- An external hosted captive portal to verify students via roll number and date of birth as input received from them.
- From received inputs, mobile number is identified from existing SQL database
- Send SMS OTP and validate mobile number
- User session will last only for 1 hour
- User can maximum use upto 2 hours of Wi-Fi only

WISPr

- WISPr – Wireless Internet Service Provider Roaming
- WISPr provides Authentication and accounting
- WISPr combines
  - Wireless AP,
  - Web browser,
  - HTTP server (Captive portal server),
  - RADIUS server to provide service.
- Resulting mechanism is called Universal Access Method (UAM) – UAM allows users to access hotspot services with general-purpose web browsers.
- No pre-configuration or pre-installed software required in user's device

DGV WISPr call flow

Client

Access Point

Captive Portal

Authenticator – Ruckus ZoneDirector

RADIUS Server – FreeRADIUS

1. Connect to Hotspot SSID

Accept Association

2. Open browser to URL (DNS/HTTP)

Restricted Mode

3. Redirect to captive portal login page

4. Display login page, Enter rollno, dob

5. Verify SMS OTP

6. Redirect to connect.php

7. Send NB credentials to ZoneDirector

8. Verify NB credentials with authentication server

9. Authentication Success

Unrestricted Mode

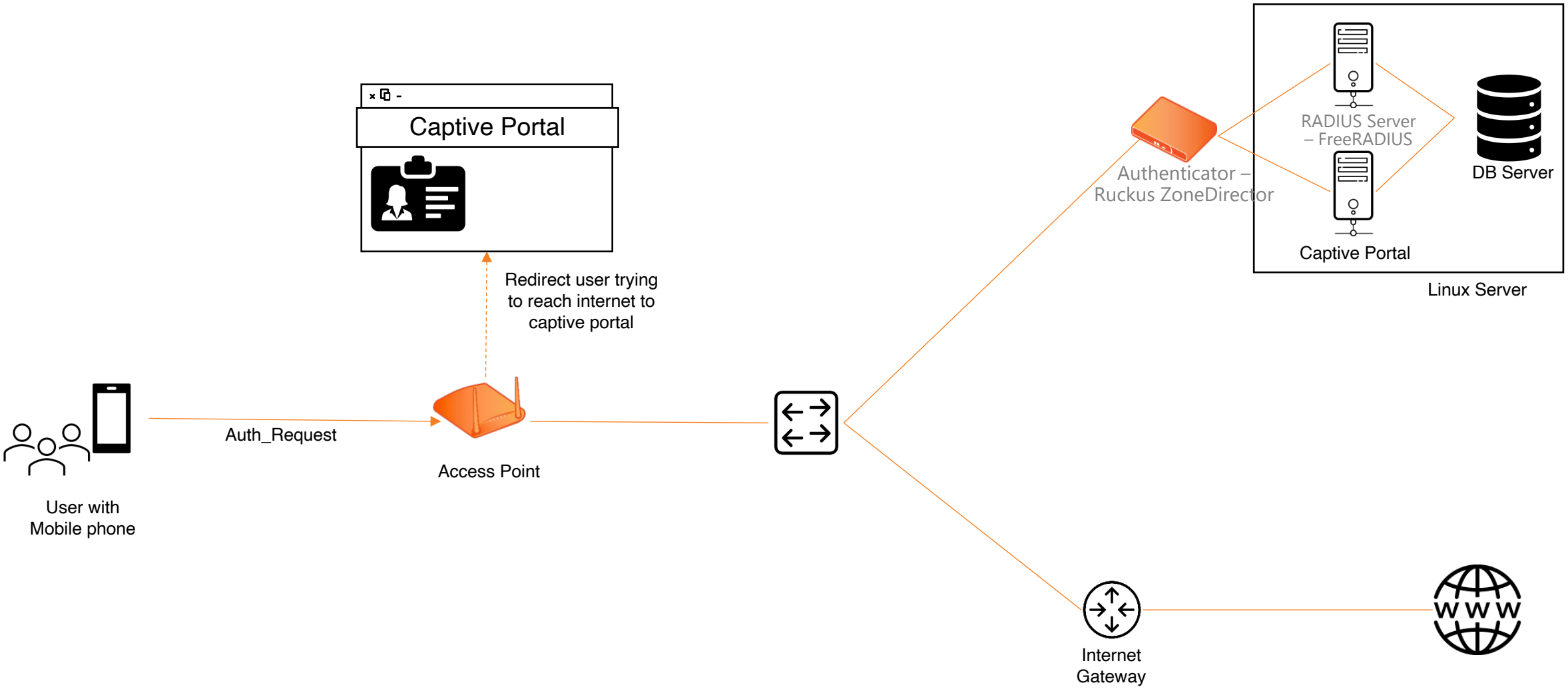
10. ZoneDirector approves user for 1 hour

11. RADIUS sends CoA to disconnect user after Session Disconnect

11. ZD disconnects user with Session Disconnect

Ruckus Simply Better Wireless. | BROCADE

Components



Demo Components

- On Server side - A Raspberry pi is used as HTTP Captive portal server and RADIUS server along with Database server for this demo
  - FreeRADIUS
  - Apache2
  - MySQL
- Captive portal is built with HTTP/PHP scripts

RUCKUS™ an ARRIS company

RaspberryPi

freeRADIUS

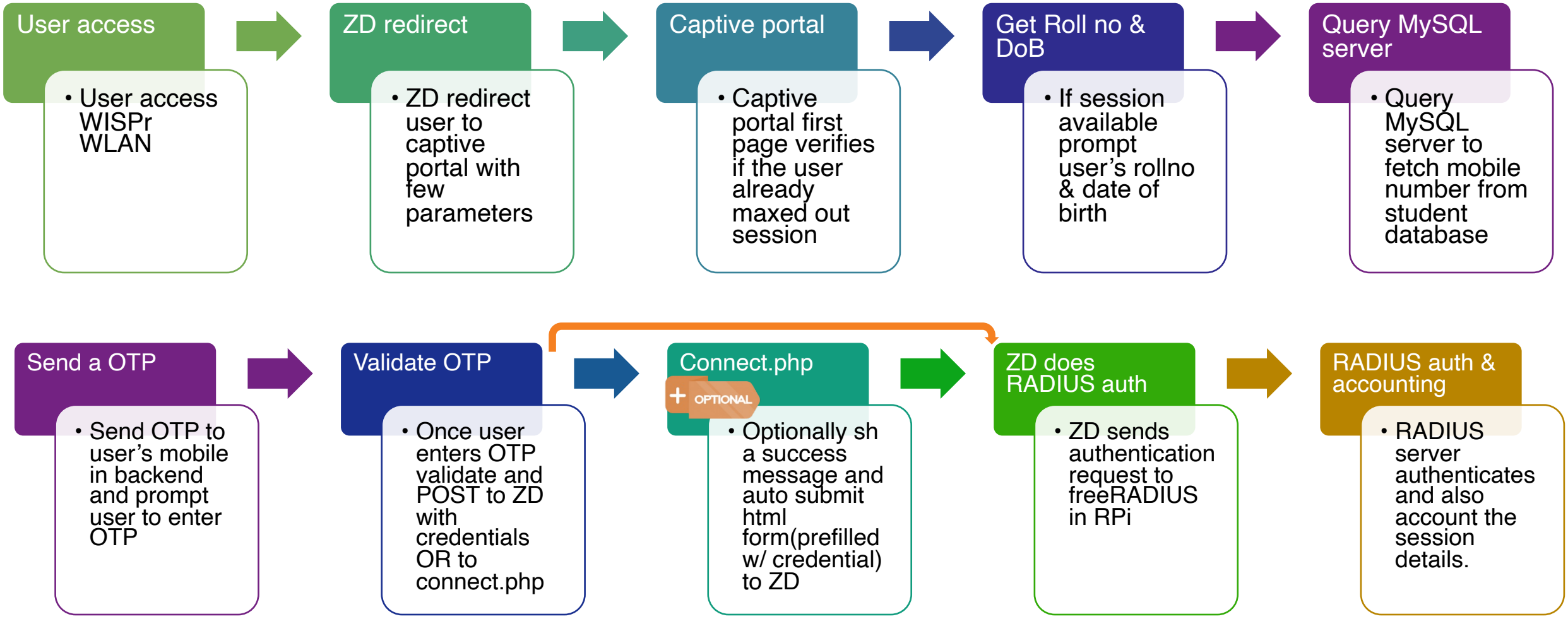
MySQL®

APACHE HTTP SERVER PROJECT

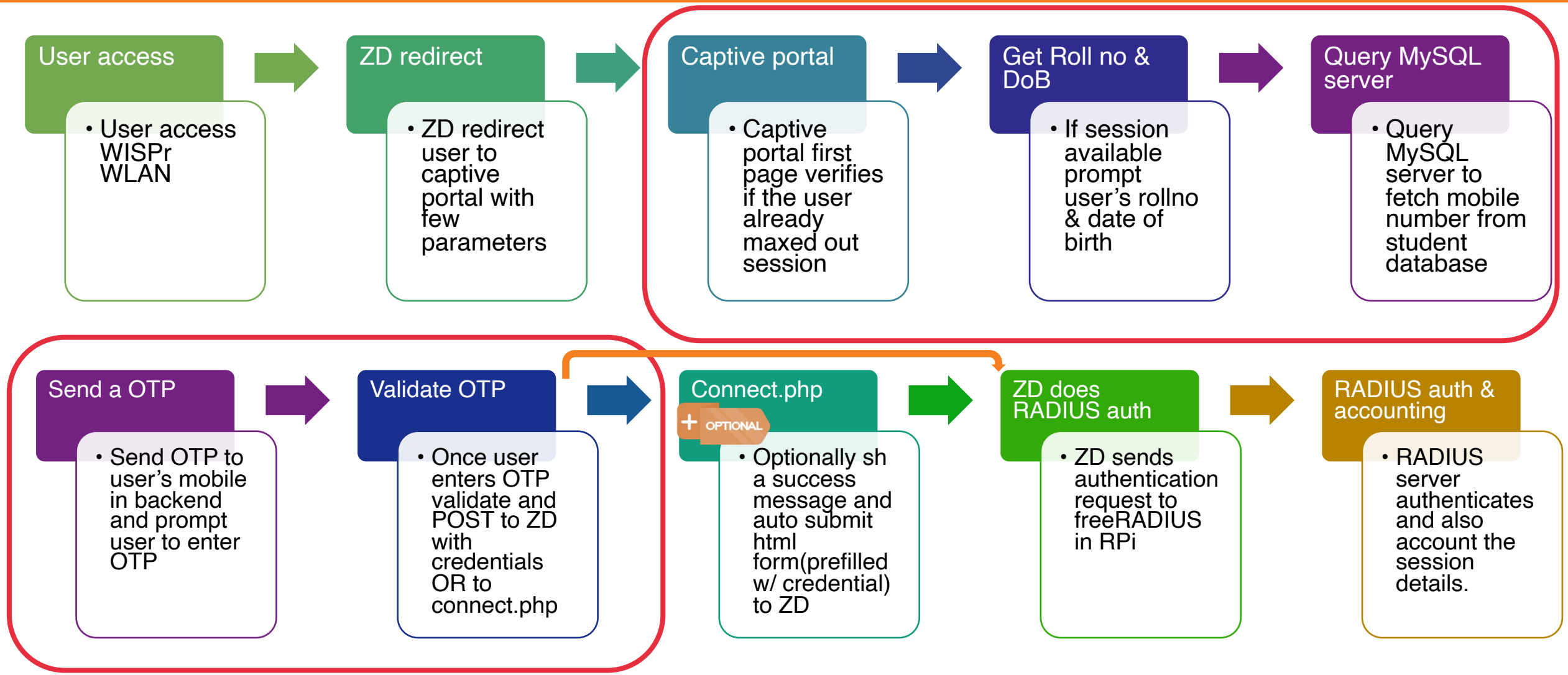
Demo MySQL Schema

- Databases and tables
  - Radius
    - radcheck – users list- In this case students database is going to be checked against by captive portal. So radius database will have only one user for northbound communication. Username : vaishnav password: 2346923hljdshgfydsjkgf#^$#^$shfldsahsdfds. This table also holds the session timeout value = 2 hours = 7200 seconds. *#Username and password are not hard set. Partner may change it as required.*
    - Radacct – Accounting details of each MAC address. If a user reached 2 hours session time limit of the day this table will be populated with Session-Timeout along with MAC addresss. This detail is checked in captive portal via a mysql_query before going to SMS OTP validation.
  - Students
    - Partner scope which contains the user details such as rollno, dob, mobileno.

HTTP Captive portal Server



Partner Dev team scope and coverage



WISPr attributes ZD pass to Captive portal

```
Eg initial redirect.
http://10.10.10.222/register.php?sip=10.10.10.254&mac=24682a1c1e31&client_mac=32e3abdc7f55&uip=10.10.10.218&lid=network%3dtest-wispr&dn=unleashed.ruckuswireless.com&url=http%3a%2f%2fwww.msftconnecttest.com%2fredirect&ssid=Vaishnav&loc=Home&vlan=1
```

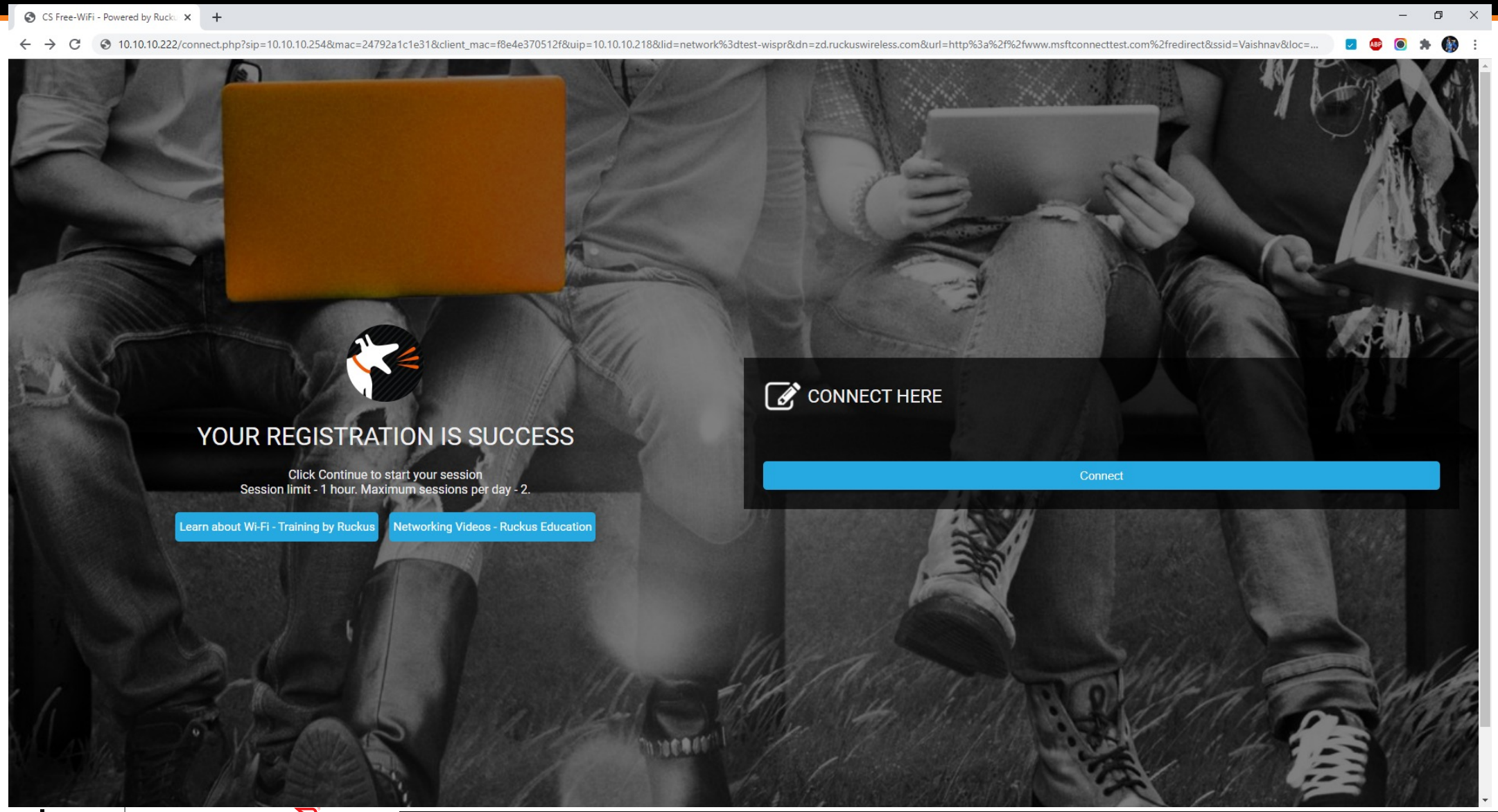
Abbreviation	Description
sip	The IP address of ZoneDirector.
mac	The MAC address of the Access Point (Ethernet).
lid	The Location ID of the Hotspot service.
uip	The client's real IP address. In a Layer 3 NAT environment, the client's IP address will be translated to the gateway's IP address when logging to the Hotspot service. In this case, the login request has to include the client's real IP address to be handled properly.
dn	The domain name of the ZoneDirector. The domain name is obtained from the SSL certificate when importing a certificate to ZoneDirector.
uid	The user's login ID (passed in the UAM login form's user name parameter).
client_mac	The client's MAC address.
SSID	The SSID to which the client is associated.
Loc	The location name defined in the AP settings.
vlan	The client's VLAN ID.
reason	The reason for redirection; can be empty for first redirect, failed for auth failure, or logout when client logs off.

Web page Requirements

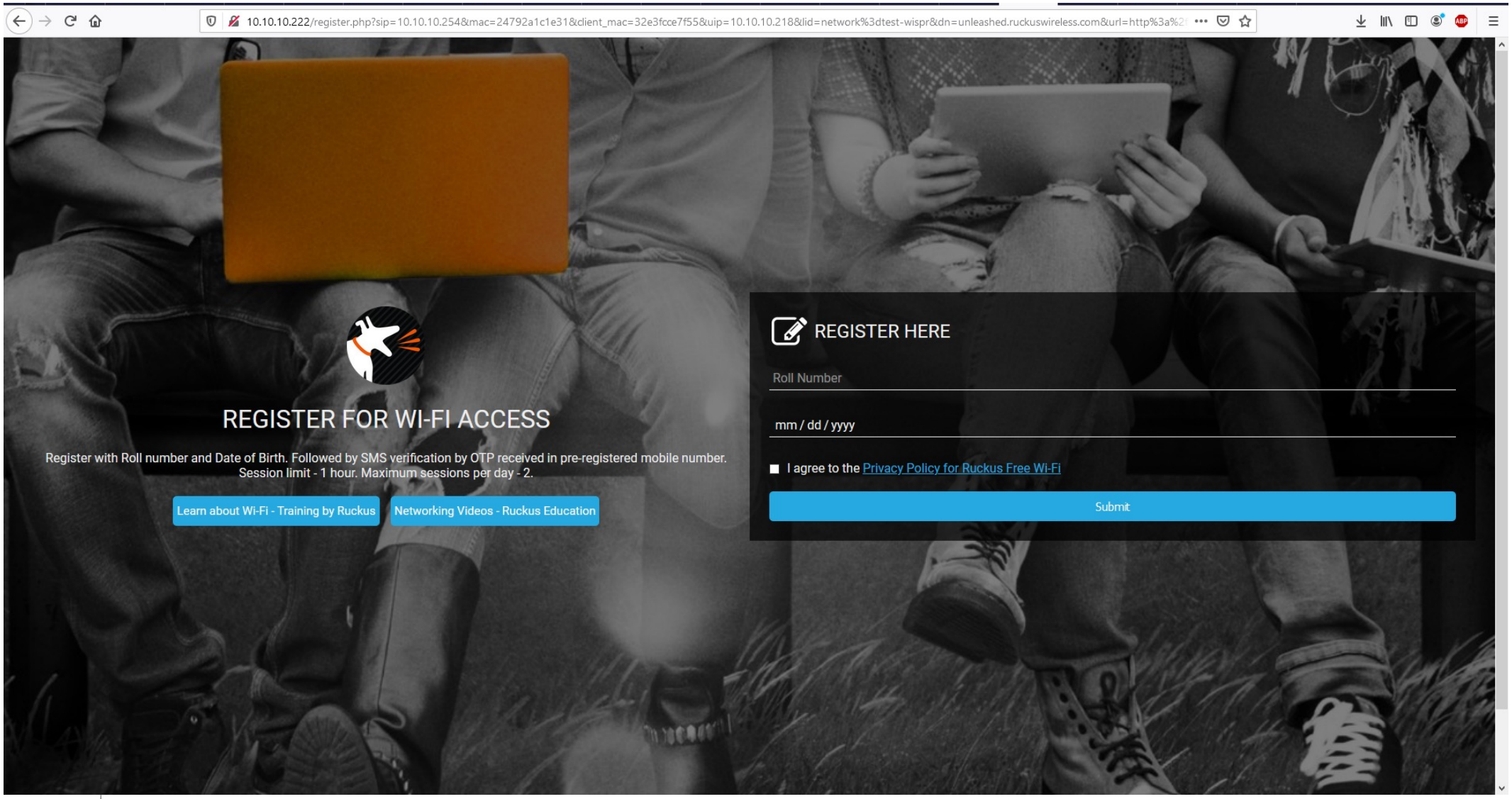
- Secure option : Can you avoid connect.php file completely?
  - Yes. You can. If you want to submit directly from SMS OTP Page directly to ZD then, radcheck table should have phone number and OTP.
  - Other suggestions
    - For SMS OTP use ajax to run timer and validate input type.
    - Use session variable to allow only for 3 times to input OTP.

- Less secure option: Once rollno and dob is verified before sending SMS OTP, check radcheck table to see if rollno/dob pair available or not. If not available, then INSERT rollno,dob to it.
- for connect.php
  - From start page through Mobile number verification, client_mac should be retained.
  - Finally client_mac should be posted from page where SMS OTP is verified as POST message.
  - Rollno and DoB should be maintained by session.
  - Connect.php file usually auto-submits after 30 seconds.

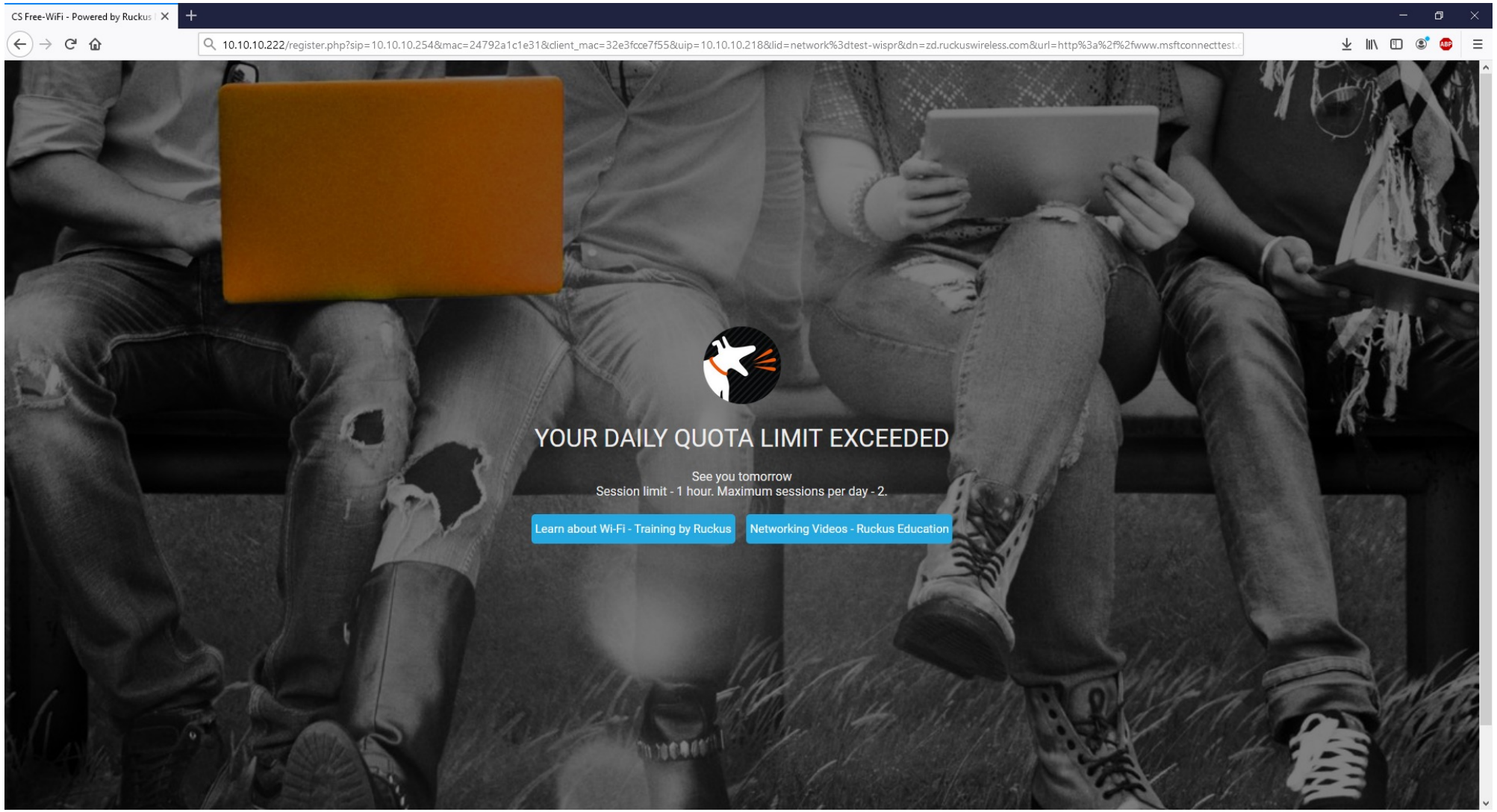
Connect.php



Sample First page where ZD redirects for user login



Sample First page where ZD redirects for users surpassed limit



Quota/Session limits

- User session will last only for 1 hour
  - Configured in ZD WISPr portal

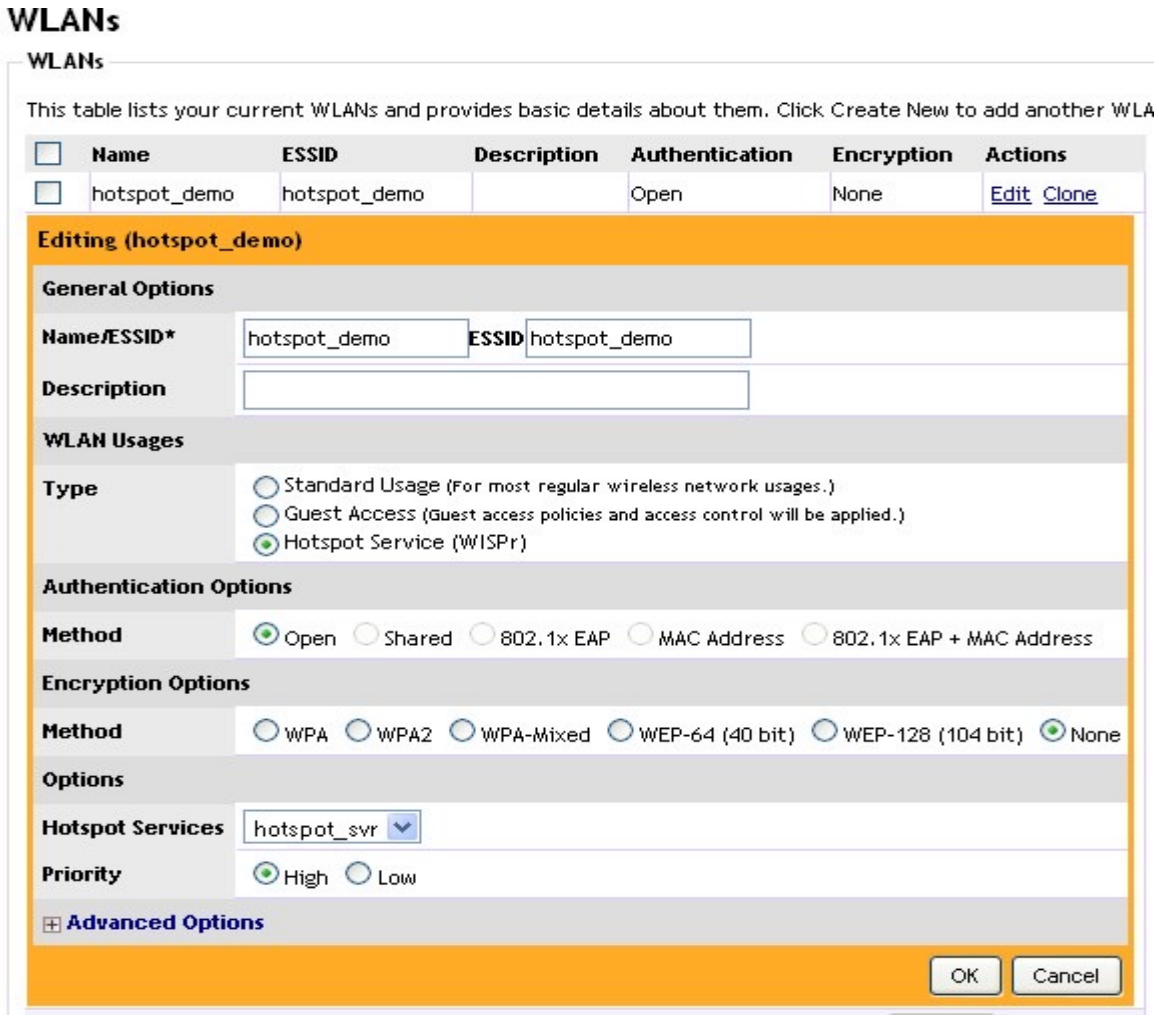
- User can maximum use upto 2 hours of Wi-Fi only
  - Configured in radcheck database. Update as follows
  - Select DB - `Use radius;`
  - If connect.php method used - `UPDATE radcheck SET value=7200 where UserName='vaishnav';`
  - If SMS OTP inserted into radcheck table - `UPDATE radcheck SET value=7200 where UserName='9876543210';`

References

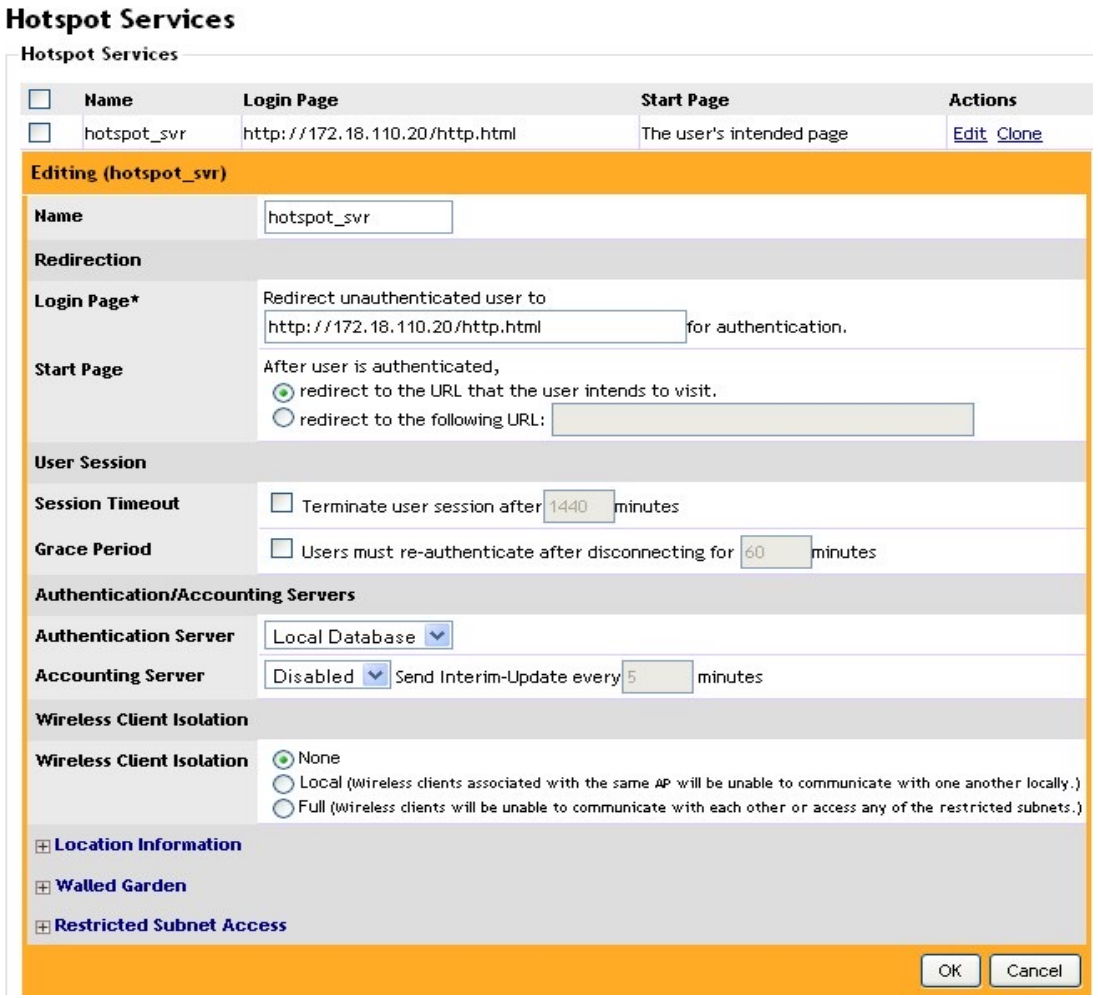
- Ruckus WISPr appnote - https://webresources.ruckuswireless.com/pdf/appnotes/appnote-wispr.pdf
- Common WISPr Attribute Abbreviations - https://docs.commscope.com/bundle/zd-10.0-userguide/page/GUID-6DEF7C24-4EBF-443B-B4DF-7F5700F0CE53.html
- WISPr callflow - https://www.youtube.com/watch?v=sqbvXkV3eUg

Backup slides

SSID configuration



WISPr configuration



FreeRADIUS installation

- raspi-config
- sudo apt-get install apache2 mysql-server libmysqlclient-dev
- sudo apt-get install php5 php-common php-gd php-curl php-mysql
- sudo apt-get install freeradius freeradius-mysql freeradius-utils

Dictionary File

```
#
#       The filename given here should be an absolute path.
#
$INCLUDE        /usr/share/freeradius/dictionary

#
#       Place additional attributes or $INCLUDEs here.  They will
#       over-ride the definitions in the pre-defined dictionaries.
#
#       See the 'man' page for 'dictionary' for information on
#       the format of the dictionary files.

#
#       If you want to add entries to the dictionary file,
#       which are NOT going to be placed in a RADIUS packet,
#       add them here.  The numbers you pick should be between
#       3000 and 4000.
#

#ATTRIBUTE      My-Local-String         3000    string
#ATTRIBUTE      My-Local-IPAddr         3001    ipaddr
#ATTRIBUTE      My-Local-Integer        3002    integer

ATTRIBUTE       Daily-Session-Time      3000    integer
ATTRIBUTE       Max-Daily-Session       3001    integer
```

Ruckus Simply Better Wireless. | BROCADE

Install mysql schema

```
root@raspberrypi:/etc/freeradius/sql/mysql# ls
admin.sql  counter.conf  cui.conf  cui.sql  dialup.conf  ippool.conf  ippool-dhcp.conf  ippool.sql  nas.sql
schema.sql  wimax.conf  wimax.sql

mysql -uroot -pPasswordForRootSql
CREATE DATABASE radius;
exit

root@raspberrypi:/etc/freeradius/sql/mysql# mysql -u root -p radius < /etc/freeradius/sql/mysql/schema.sql;
root@raspberrypi:/etc/freeradius/sql/mysql# mysql -u root -p radius < /etc/freeradius/sql/mysql/nas.sql;
```

Enabling SQL and counter

Uncomment the following lines in file radiusd.conf

```
root@raspberrypi:~ # vi /etc/freeradius/radiusd.conf

$INCLUDE sql.conf
$INCLUDE sql/mysql/counter.conf
```

Enabling SQL and counter

Uncomment the word ‘sql’ in file default and uncomment word ‘file’ for all categories This will instruct FreeRADIUS to relly on the database for user management.

```
root@raspberrypi:~ # vi /etc/freeradius/sites-available/default
authorize {
.....
sql
#file
....
}
accounting {
......
sql
....
}
post-auth {
......
sql
....
}
session{
......
sql
.....}
```

Enabling SQL and counter

Edit the radius SQL module's config

```
root@raspberrypi:/etc/freeradius/sql/mysql# vi /etc/freeradius/sql.conf
sql {
        #
        #  Set the database to one of:
        #
        #       mysql, mssql, oracle, postgresql
        #
        database = "mysql"

        #
        #  Which FreeRADIUS driver to use.
        #
        driver = "rlm_sql_${database}"

        # Connection info:
        server = "localhost"
        #port = 3306
        login = "radius"
        password = "YourPreferredPa$$word"

        # Database table configuration for everything except Oracle
        radius_db = "radius"
```

Enable Daily counter in SQL

```
root@raspberrypi:/etc/freeradius/sql/mysql# vi /etc/freeradius/sql/mysql/counter.conf

#  DEFAULT  Max-Daily-Session > 3600, Auth-Type = Reject
#      Reply-Message = "You've used up more than one hour today"
#
sqlcounter dailycounter {
        counter-name = Daily-Session-Time
        check-name = Max-Daily-Session
        reply-name = Session-Timeout
        sqlmod-inst = sql
        key = User-Name
        reset = daily

        # This query properly handles calls that span from the
        # previous reset period into the current period but
        # involves more work for the SQL server than those
        # below
        query = "SELECT SUM(acctsessiontime - \
                GREATEST((%b - UNIX_TIMESTAMP(acctstarttime)), 0)) \
                FROM radacct WHERE username = '%{%k}' AND \
                UNIX_TIMESTAMP(acctstarttime) + acctsessiontime > '%b'"

}
```

NB user

```
root@raspberrypi:/etc/freeradius/sql/mysql# mysql -uroot -p
Enter password:

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use radius
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from radcheck;
+----+----------+------------------+----+-----------------------------------------+
| id | username | attribute        | op | value                                   |
+----+----------+------------------+----+-----------------------------------------+
|  1 | karthik  | User-Password    | := | skldjkgftdfk873yjk4enw2876kebkjfsb682grb |
|  2 | karthik  | Max-Daily-Session | := | 7200                                    |
+----+----------+------------------+----+-----------------------------------------+
2 rows in set (0.01 sec)

mysql>
```