IJARIIT INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X

Impact factor: 4.295

(Volume 4, Issue 3)

Available online at: www.ijariit.com

Wi-Fi hotspot with captive portal on Raspberry Pi

Annapurna B P
annapurnabp03@gmail.com
Bangalore Institute of Technology,
Bangalore, Karnataka

Vani V
vanisrin@gmail.com
Bangalore Institute of Technology,
Bangalore, Karnataka

ABSTRACT

The Internet is the biggest network of computers world-wide for communication. Internet or inter-network of these devices help make rapid progress in the technology making the world a better place to live in. Every day there is an innovation which leads to new methods of communication and thus, networking. In the proposed work, Wi-Fi hotspot is enabled on Raspberry Pi with a Captive portal technique implemented, where the user can connect to the hotspot with the help of a captive portal page. The required entries like username and password need to be entered in the portal page which authenticates the user. The entries are verified in the database, if the entries are valid the user gets the internet access for certain period of time. If the entries are invalid access gets denied. The message status is maintained in the database to check whether the message is sent to the user or not.

Keywords: *Captive portal, Hotspot, SSID, Wi-Fi.*

1. INTRODUCTION

Wireless Fidelity (Wi-Fi), is the commercial name for 802.11 standards. In both business and home environments it has become the preferred technology for wireless local area networking (WLAN). People can obtain Internet access through hotspot which typically uses Wi-Fi technology via WLAN using router connected to an internet service provider (ISP). Public hotspots are created from wireless access points configured to provide internet access. These public hotspots may be available for free of cost and some may have terms and conditions with payments. In the proposed system, Wi-Fi hotspot is enabled with a captive portal on the raspberry pi. Here operating system, database, and webserver are installed and configured on the raspberry pi board. The random is generated and saved along with the user's mobile number in the database. The user gets the message which contains randomly generated alphanumeric string as password and user mobile number as user name. When user device tries to access Wi-Fi hotspot it gets connected to the device and redirects user to the captive portal page. Here user needs to enter the credentials required. These credentials are validated with the database and if it matches with the database the device gets internet access for certain period of time. If the entries in the captive portal page do not match with the database entries there will be no access to the internet.

2. MOTIVATION

Currently, routers are used to enable Wi-Fi with a captive portal. This is very expensive, and the maintenance of the system is very difficult. Using Raspberry Pi for this reduces the cost, maintenance is easy and consumes less power.

3. PROBLEM STATEMENT

In the proposed system, Wi-Fi hotspot with a captive portal on Raspberry Pi is configured. The overall mechanism is as follows. Firstly, enable Wi-Fi hotspot with a captive portal by configuring the network and generate a random string. The user needs to enter the required details. If it is matched the internet can be accessed by the user. Access gets disconnected after a certain period. If it does not match internet access is denied.

4. LITERATURE SURVEY

In NoCat the access is given only to the list usernames and passwords. When a user tries to access, the captive portal page is displayed. Access is granted to only those listed usernames with password need to be entered in the captive portal page. If the user name and password is correct then the user gets internet access. The Wi-Fi Dog authentication server is a PHP and Postgre SQL server-based solution written to authenticate clients in a captive portal environment. Wi-Fi Dog Auth provides portal specific content management, allows users to create wireless internet access accounts using email access, and provides gateway uptime statistics and connection specific and user log statistics [1].

Captive portals can also be built using Open BSD and routers. For this, first the webserver is set up and then the portal application is done. Later, DHCPD, BIND, and firewall are configured [2].

The captive portal feature in the router allows blocking wireless clients from accessing the network until user verification has been established. A captive portal is configured in such a way that allows access for authenticated users after verification. Authorized users provide a valid user name and password that is verified with the local database or server [3].

Portal authentication is known as web authentication. With portal function configured on EAP Controller, when the unauthorized wireless client is connected to EAP managed by EAP Controller and tries to access to the internet, it will be directed to a pre-set web page which requires additional authentication information for accessing to the internet. The only authorized wireless client can access the internet by passing the authentication page [4].

5. METHODOLOGY

The architecture of the proposed system is as shown in the Fig.1. The operating system is installed first, then the database and web server will be installed and configured on Raspberry Pi. When client device tries to access the internet through Wi-Fi then login portal is opened when clicked on the SSID. The client needs to enter his username and password. Once the required details are entered it is validated against the database. If the username and password matches, then the client device gets access to the internet for a certain duration of time. After the specified time, the internet gets disconnected. The user name, password and message status are saved in a database for future reference. The username and password are messaged to the user which need to be entered in the captive portal to access the internet.

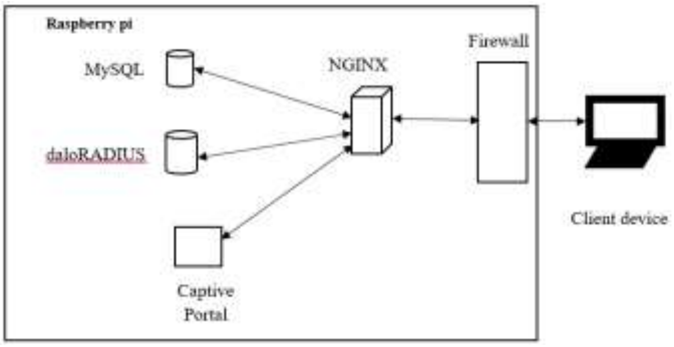


Fig.1. The architecture of the proposed system

Step 1: Installation of OS and required software
The Raspbian OS needs to be installed on the Raspberry Pi board. Along with OS installation, some software is installed. The software that exists with standard installation needs to be updated. The required packages like debhelper, libssl-dev, libcurl4-gnutls-dev, hostapd, dnsmasq, nginx are installed.

Step 2: Set static IP address
Raspberry Pi will be the DHCP server on the wireless network, a static IP address needs to be assigned to wireless adapter because Raspberry Pi will be the DHCP server on the wireless network. In the interfaces configuration file, the static IP address is assigned. To verify enter ifconfig command at command prompt.

Step 3: Configuring hostapd
Hostapd is software allows the Raspberry Pi to accept wireless connections from clients. So, its configuration file for network needs to be built. Depending on wireless adapter the driver parameters need to be set.

Step 4: Configuring Dnsmasq
Dnsmasq is software that assigns IP addresses to devices when they connect to the network and resolves host names to IP addresses. The DNS server address, route, the range of IP address are to be edited in its configuration file.

Step 5: Configuring MySQL and FreeRadius
MySQL database is to be installed and the database radius is to be created. Install FreeRadius. Edit the radius configuration file and default file.

Step 6: Connect to captive portal page

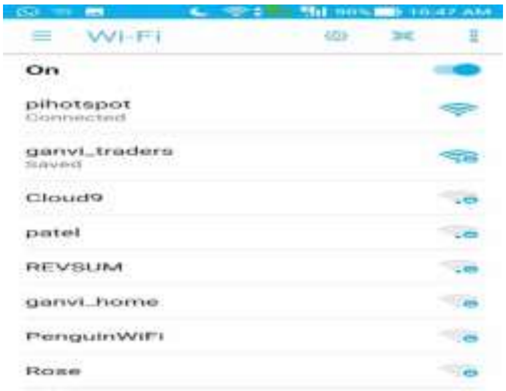
Create the directory that will contain captive portal page. Setting the permissions of the directory to setgid bit (g+s) makes it so that any new files created within the directory will inherit the directory's group owner. Nginx uses the group www-data, so the owner of all files within the HTML directory to pi and the group owner www-data are set.

Step 7: Configure Nginx

Nginx host multiple sites, and each one requires its own configuration file. The configuration file tells nginx where the files for the site are located and how to handle client requests. Create a configuration file and activate the new site by creating a link to it. Deactivate the default nginx site to prevent conflicts.

6. RESULTS AND SNAPSHOTS

In this section, the output of the proposed system is discussed along with the snapshots. The below Fig. 2 shows that Wi-Fi hotspot is enabled on the raspberry pi. The SSID is pihotspot. To enable the SSID set the static IP address and configure hostapd configuration. The device is connected to pi hotspot but there is no internet access for the device. To enable the wi-fi hotspot on raspberry pi dnsmasq is configured. Apache server and Mysql need to be installed. Freeradius need to be configured with Mysql.



On

pihotspot
Connected

ganvi_traders
Saved

Cloud9

patel

REVSUM

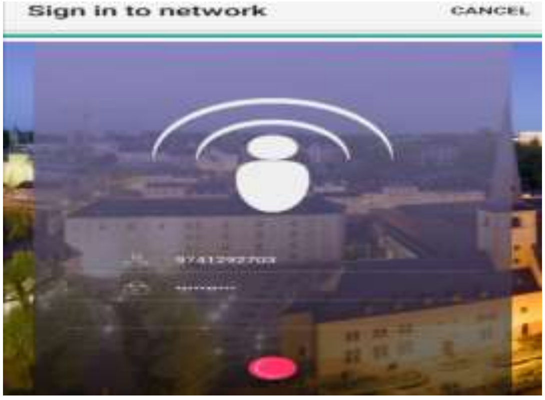
ganvi_home

PenguinWiFi

Rose

Fig. 2. Connected to Wi-Fi hotspot but no internet

The below Fig. 3 shows the captive portal page when the user clicks on SSID device gets redirected to the captive portal page. To create this page css script is used. Many functions are used to build this page. SVG, keyframes, media etc., functions are used. The background image, width, margin, height everything is specified. In sign-in username and password are entered to get an internet access. This takes user phone number as username and the randomly generated string as a password. When the user enters this, it is checked with the database whether it's valid or not.



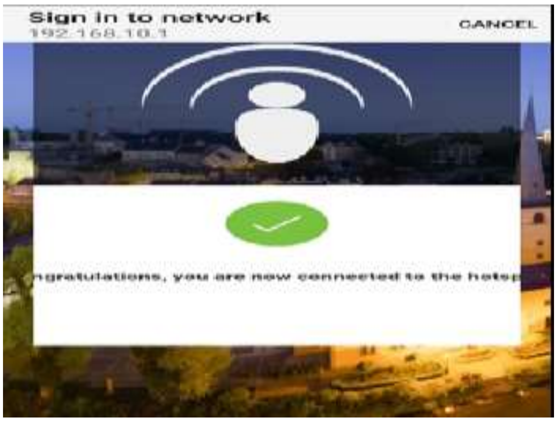
Sign in to network

9741292703

••••••••

Fig. 3. Sign in

The Fig. 4 below shows device got access to the internet. The entered username and password are checked with the database and it is correct so the device has got an access to the internet.

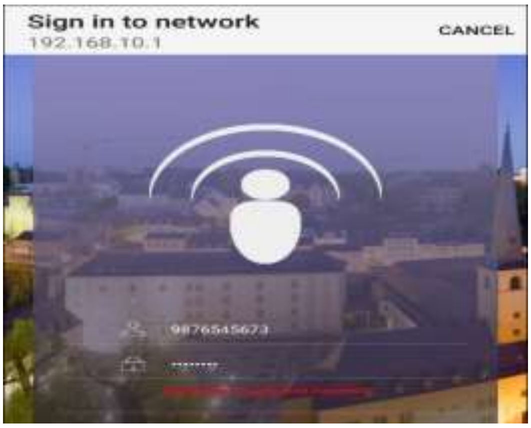


Sign in to network
192.168.10.1
CANCEL

ngratulations, you are now connected to the hotsp

Fig. 4. Device obtained internet access

The below Fig. 5 shows unsuccessful login. When the user enters the user name and password and clicks on the sign in it checks the database whether the given details are valid or not. If the username or password is not matching with the details in the database internet access will be denied.

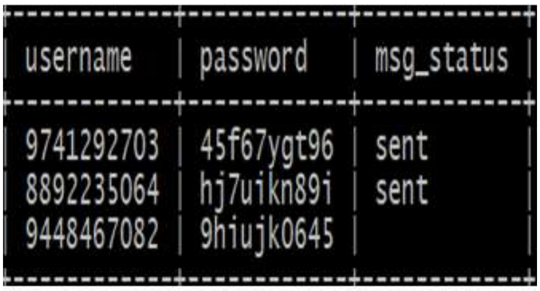


Sign in to network
192.168.10.1
CANCEL

9876545673
....

Fig. 5. Access denied

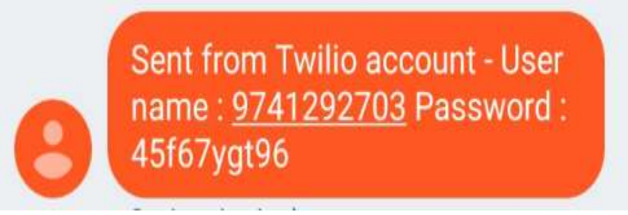
The below Fig. 6 shows the details that are saved in the database table. The password is generated randomly, message status is maintained whether the message has been sent to the user or not is checked.



```
+------------+------------+------------+
| username   | password   | msg_status |
+------------+------------+------------+
| 9741292703 | 45f67ygt96 | sent       |
| 8892235064 | hj7uikn89i | sent       |
| 9448467082 | 9hiujk0645 |            |
+------------+------------+------------+
```

Fig. 6. MySQL Database

The Fig. 7 shows the message sent from the Twilio account to the client. This uses the number generated by the Twilio account during registration to send a message. The message contains username and password.



Sent from Twilio account - User name : 9741292703 Password : 45f67ygt96

Fig. 7. Twilio message

7. CONCLUSION

The wi-fi hotspot is enabled on the Raspberry Pi which acts as a router. The captive portal page is displayed for the users. The random generation of the string is done, and the message status is maintained. The login credentials are sent to users. The login credentials are entered in the portal page. These entries are checked with the database. The valid entries get the access to the internet for 900 seconds and after time out automatically logs out. If the entries are invalid access gets denied.

8. REFERENCES

[1] Michael Lenczer, "Wireless portal with Wi-Fi Dog", Linux Journal, Oct 2005.
[2] The concept of building a captive portal on Open BSD is referred from below website. http://www.bsdguides.org/2012/building-a-captive-portal-with-openbsd/
[3] The concept of building a captive portal on the router is referred from below website. https://www.linksys.com/ca/support-article?articleNum=159676
[4] The portal function configured on EAP Controller is referred from below website.
https://www.tp-link.com/us/faq/896.html