Automation of IP-PBX Maintenance

Prem Kumar A, Jayavignesh T
M.Tech Communication Engineering, School of Electronics Engineering, VIT University Chennai Campus, Chennai, India
Assistant Professor, School of Electronics Engineering, VIT University Chennai Campus, Chennai, India

***Abstract*— Arrangement of self-administration is called as Automation. This research focused on giving automated telecommunication benefit utilizing pre-characterized techniques which are done electronically without human mediation. Four primary regions that can be automated in Telecommunication are Administration, Reporting, Policy Enforcement and Auditing. This work is about making a Periodic Administrative Maintenance instrument for IP-PBX (Internet Protocol – Private Branch Exchange) Communication Server. An IP-PBX is only a Private Branch Exchange with Internet Protocol which switches calls between ordinary phones and also VOIP clients on nearby lines. This automated tool is made utilizing an effective scripting dialect Python with MySQL db, Django, HTML and JavaScript. Prepare incorporates - Scheduling the day and time, when it coordinates the present day and time, python center script keeps running at foundation. Every node is Telnet-ted using IP address, login and password which then flows down with a number of commands through every node and results will extracted in an excel sheet". Followed by a mail to the user for check. This process will happens periodically when the day and time coordinates the present day and time and each time a mail with excel sheet report of an as of now executed undertaking will be sent.**

***Index Terms*— IP-PBX, VOIP, Communication Server, Telnet, Python, MySQL db, HTML and JavaScript.**

I. Introduction

Automation is a process of reducing human intervention and increasing accuracy. Through automation, procedures are carried out electronically and not by manually. A PBX (Private Branch Exchange) is called as private telephone network used within an enterprise. It is used to switch calls between company users through the local lines and also allow users to make external calls through a certain number of external phone lines. A PBX with the ability to switch calls between VoIP (Voice over Internet Protocol) users is IP-PBX. It is also called as VoIP phone system. SIP (Session Initiation Protocol) phones are IP phones which uses VoIP to establish a connection. In telecommunication, automation plays an important role. Reporting, Policy Enforcement, Auditing, Administration and Maintenance are the main areas in telecommunication that can be automated. An IP-PBX is effectively known as Enterprise Communication Server. This project is purely focused on maintenance of above mentioned server. For this, a web based administrative tool is created which will run periodically. This tool is created using a powerful scripting language Python along with Django, HTML and JavaScript. This tool will run once in 7 days and a report will be created. Companies like

Companies research and development wing be made up of number of Teams. Each team maintains several number of mockups and there will be more than 6 nodes in each mockup. To maintain the quality of these nodes, they need to check several commands or test cases in each node (approx. 200 nodes) and it is a time consuming task. To overcome that, this automation tool is created, which will automatically logged into each and every node through telnet. In each node, several number of commands will be executed and the result will be stored in excel sheet. This excel sheet will get mailed to the concerned person automatically.

II. Background Study

A. IP-PBX

A Communication Server used by an enterprise is commonly called as an IP-PBX. Communication Server consists of Physical boards with several number of ports. Analog port, Digital port, Gateway port, Server port, Ethernet port, Power port, Trunk port, D2A port, A2D port. Each Communication Server will have a separate IP address and MAC address. IP phones will be connected to the digital port and analog phones to the analog port. A trunk port will be used to connect between two communication servers. The IP phones and analog phones which are connected to communication server, will have the IP address in the same network of communication server IP address. Some uses of a communication servers are,

- To configure analog phones, digital phones such as IP phones, SIP phones and cordless phones.
- To create trunks, links between two server in same network and also in different networks.
- To configure features such as voice mail, call handling, call flow, call forwarding etc.
- To configure VoIP (Voice over IP) connection, PSTN (Public Switched Telephone Network).
- To install operating system, to configure Routing paths, to maintain routing list etc.
- To configure ports such as Ethernet port, D2A port, A2D port, TCP/UDP port, server port, gateway port, Audio/Video port etc.

B. Commands

There are commands for Addressing modes, Hardware maintenance, Resource maintenance, Network maintenance, incidents, duplication, Data base, Accounting, Voice guides, Processing, Application, and Trunk etc. Some of them are

- CPU status: To check the CPU is up or down. Like every other system, an IP-PBX has CPU which contains all the boards.
- Trunk status: To configure and check the local trunk and group trunk status. Trunk is the link between two servers.
- Encryption: To check whether the system is secured or the system isn't secured.
- Database: To check whether the database is ok or is there any error:
- Configure: To configure the Physical boards such as analog board, digital board etc. with CPU.
- LinkVisual: To see the links between different server.
- OSversion: To check the Operating system version.
- Trunk Visual: To see the channels for making call is free or busy or down. These channel are grouped into different group of trunks.
- Cordless sets: To configure the cordless phones.
- SIPphones: To configure SIP phone. Session initiation protocol is for signaling to control multimedia sessions. It is a communication protocol.
- IPsets: To configure IP phones. IP phone works on VoIP system. The major difference between SIP and VoIP is for data and voice through internet and SIP is used to carry all forms of data such as voice, data and video. SIP is an application layer protocol used to establish,

modify and terminate multimedia network sessions like VoIP calls.

- Regsets: What are all the registered sets currently inservice.
- Routelist: To configure and view the Automation Route list that phones are using to pass data.

III. LITERATURE SURVEY

Sandeep Singh, R.A. Khan, Alkha Agarwal [1] (2015) proposed infrastructure based Software Define Network (SDN). It can isolate information plane from the control plane in a system. Visual Network Description (VND) is a GUI instrument which builds up this system physically while reproduction. With this instrument, one can physically setup the required segments in SDN and can characterize the properties of various connections, switches and controllers joined with it for recreating this system through Mininet. VND consequently produces python script perfect for recreation on Mininet for this physically characterized organize. The work in this paper examines about the part of VND while recreation of foundation based Software Defined Network utilizing mininet.

K. Jambunantha (2015) proposed an Automated Procedure to upgrade remote network devices using Python. In this paper, a one of a kind technique for automation procedure is characterized which is chiefly utilized for updating the remote network devices software images, called as software builds. AP is produced utilizing Python. The AP portrayed gives an effective technique to Automation testing. As AP requires insignificant information and less handling time, it diminishes the time expended and human endeavors required over manual testing.

Di Liu, Pin Wang (2015) proposed python automated script based on Open Stack platform for the deployment of service cloud. As an Open Stack client or head, you regularly need to compose scripts to automate common tasks. Notwithstanding the REST and summon line interfaces, Open Stack is likewise unveiled a local Python API bindings. Open Stack accompanies a dashboard Web application which is perfect for performing manual undertakings, for example, beginning a solitary virtual machine (VM) occurrences, be that as it may, in the event that you need to automate cloud-based tasks, you have to compose a script which can work Open Stack.

IV. RELATED WORK

An IP-PBX is a PBX with the ability to switch calls between enterprise users through IP network. This project is focused on maintaining IP-PBX. A Communication Server used by an enterprise is commonly called as an IP-PBX. It is basically arranged in racks, each rack consists of shelves and a communication server placed in each shelves. This Communication Server consists of Physical boards with several number of ports. Analog port, Digital port, Gateway port, Server port, Ethernet port, Power port, Trunk port, D2A port, A2D port. To maintain this, we have to login each node using the IP address through Telnet or SSH and a series of steps to be followed. It is a time consuming task. This automation tool is created to reduce the time consumption and to maintain the quality of the node. User need to schedule the day and time to run the tool. Schedule input will get stored in MySQL db. When the current time equals with input, python core script starts to run. It will get the IP address, Login and Password from MySQL db for each node, which should be stored in db in prior. The result for the current run is stored in excel sheet. A mail with result attachment will be sent to the concerned person for further doings. Same way the process runs periodically when the input day and time equals with the current day and time. We can also stop the schedule and reschedule the script with different day and time. The web user interface is done by using Django, HTML and JavaScript. Core part is done by using Python. Python script login each node using IP address through Telnet and number of commands will be executed. These commands are considered as test cases and these test cases are considered as Pass or Fail based on some defined criteria. Using the number of pass/fail test cases, Pass and Fail percentage is calculated and reasons for pass/fail will get stored in excel sheets Team wise, Mockup Wise and Node wise. This is the report that get mailed to the concerned person. Using this report necessary actions will be taken to maintain the quality of the mockups.

V. Overall View

Date and time input will be saved in the MySql db and when the local time equals the input given, python script runs. Data will be taken from the MySql db for the process and the output will be extracted in as Excel sheet.

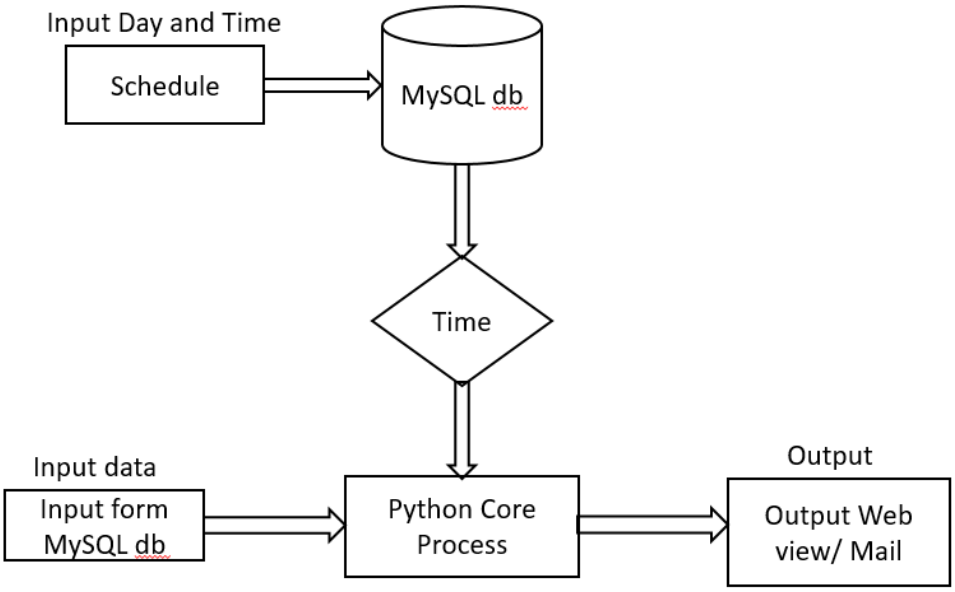


Fig. 1. Input to Output Flow

VI. Web view

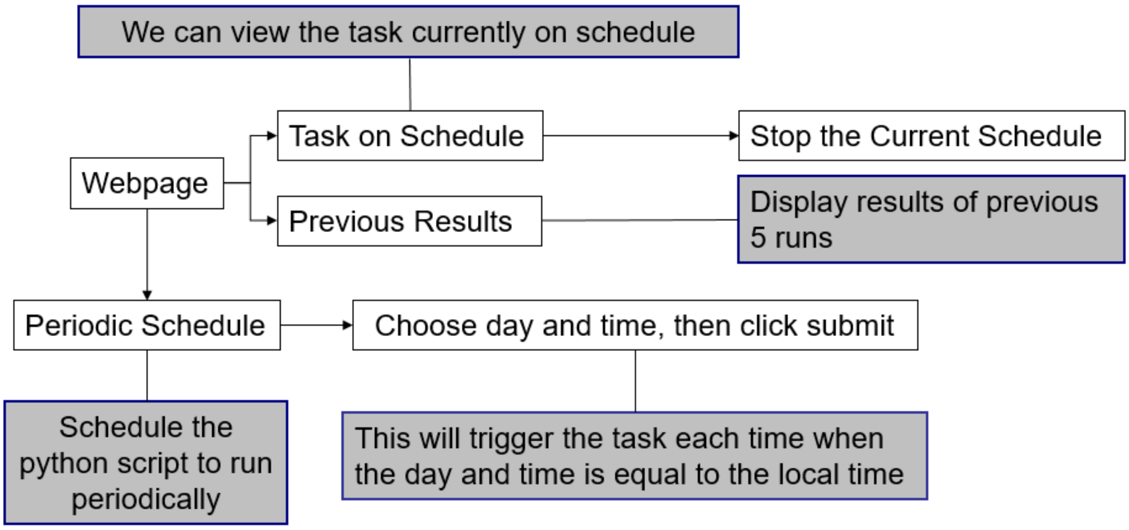

We can view the task currently on schedule
Webpage
Task on Schedule
Stop the Current Schedule
Previous Results
Display results of previous 5 runs
Periodic Schedule
Choose day and time, then click submit
Schedule the python script to run periodically
This will trigger the task each time when the day and time is equal to the local time

Fig. 2. Web view

VII. Results

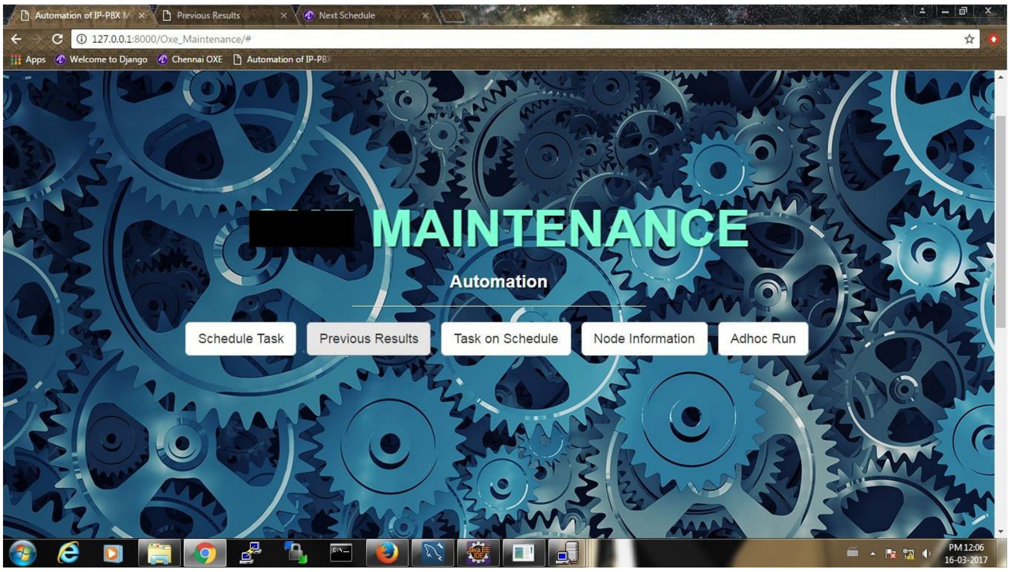
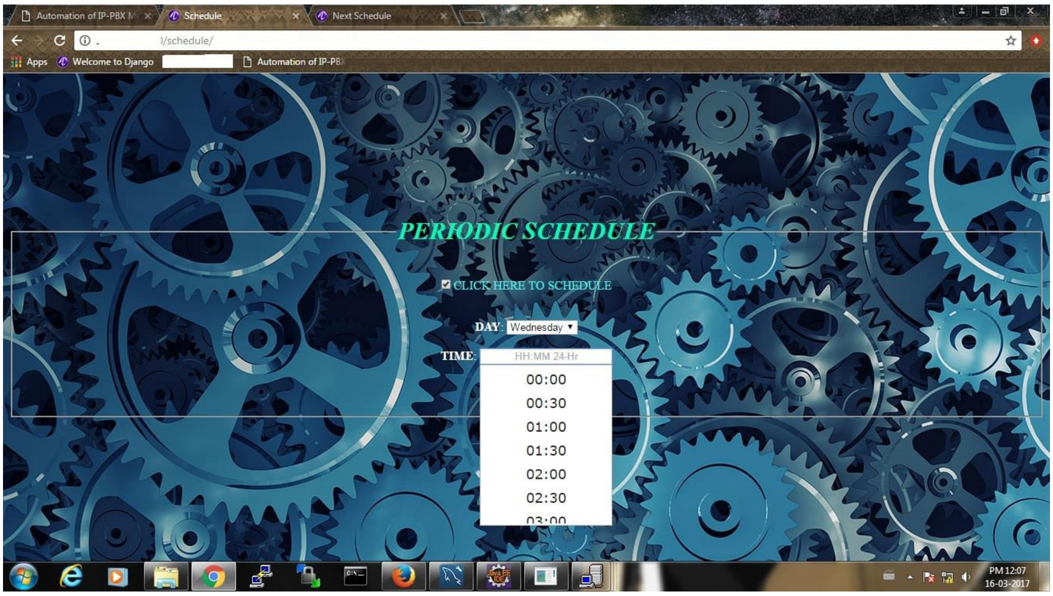


Fig. 3. Main Page

Fig. 3 shows the main pain page where we have options like Schedule (Fig.4) – to give day and time as input, Task on Schedule – to know the currently scheduled day and time, Previous Results (Fig.5) – to see the results for previous runs, Ad-hoc Run (Fig.6) – to run the script immediately and Node information – to give details above the each node.



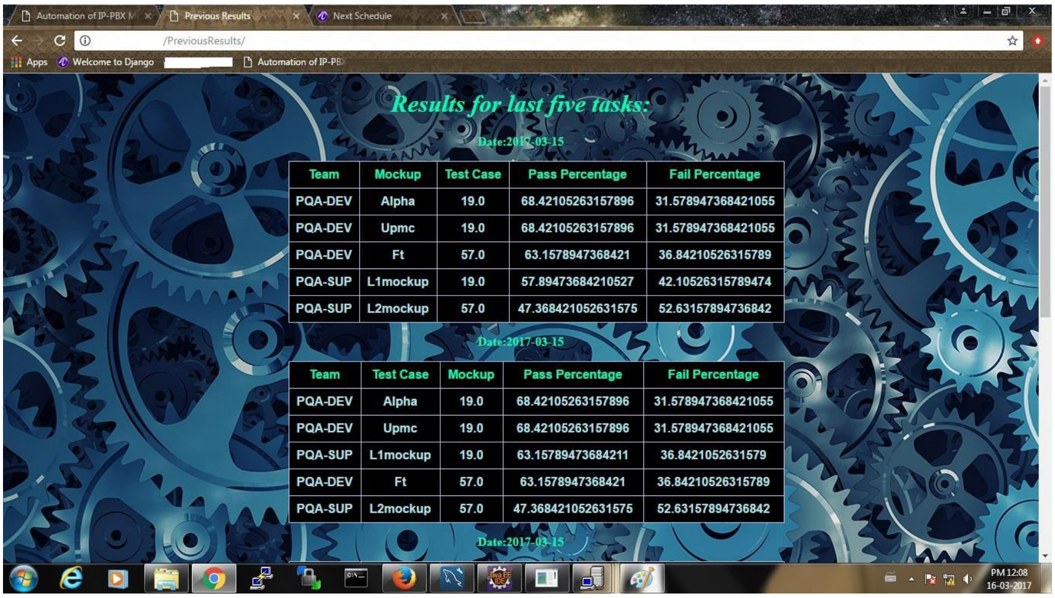
PERIODIC SCHEDULE

☑ CLICK HERE TO SCHEDULE

DAY: Wednesday

TIME: HH:MM 24-Hr

Fig. 4. Periodic Schedule



Results for last five tasks:

Date:2017-03-15

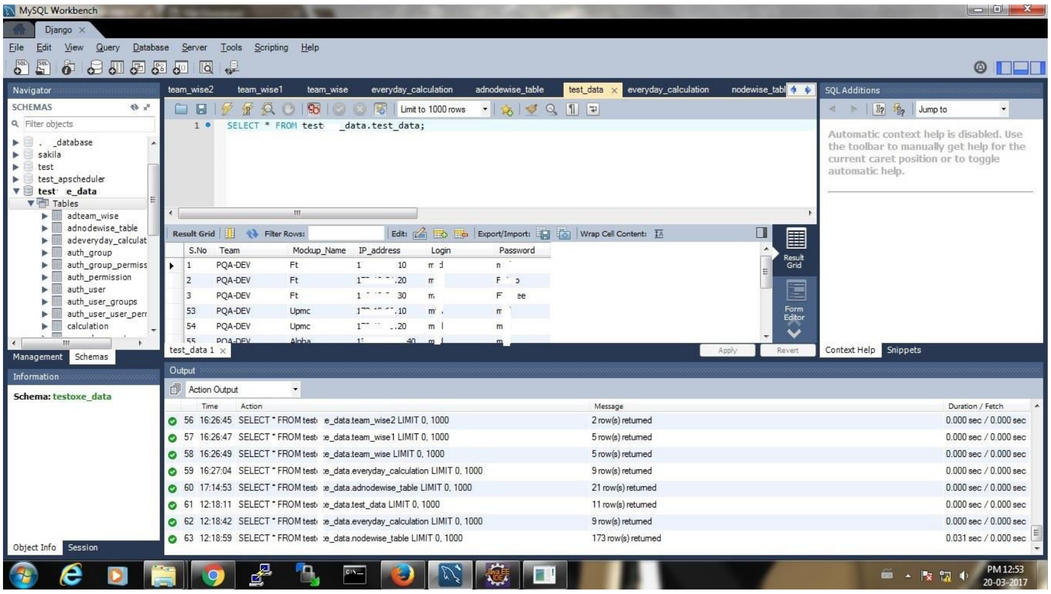
Team	Mockup	Test Case	Pass Percentage	Fail Percentage
PQA-DEV	Alpha	19.0	68.42105263157896	31.578947368421055
PQA-DEV	Upmc	19.0	68.42105263157896	31.578947368421055
PQA-DEV	Ft	57.0	63.1578947368421	36.84210526315789
PQA-SUP	L1mockup	19.0	57.89473684210527	42.10526315789474
PQA-SUP	L2mockup	57.0	47.368421052631575	52.63157894736842

Date:2017-03-15

Team	Test Case	Mockup	Pass Percentage	Fail Percentage
PQA-DEV	Alpha	19.0	68.42105263157896	31.578947368421055
PQA-DEV	Upmc	19.0	68.42105263157896	31.578947368421055
PQA-SUP	L1mockup	19.0	63.15789473684211	36.8421052631579
PQA-DEV	Ft	57.0	63.1578947368421	36.84210526315789
PQA-SUP	L2mockup	57.0	47.368421052631575	52.63157894736842

Date:2017-03-15

Fig. 5. Previous Result



Schema: testoxe_data

```
1  SELECT * FROM test[redacted]_data.test_data;
```

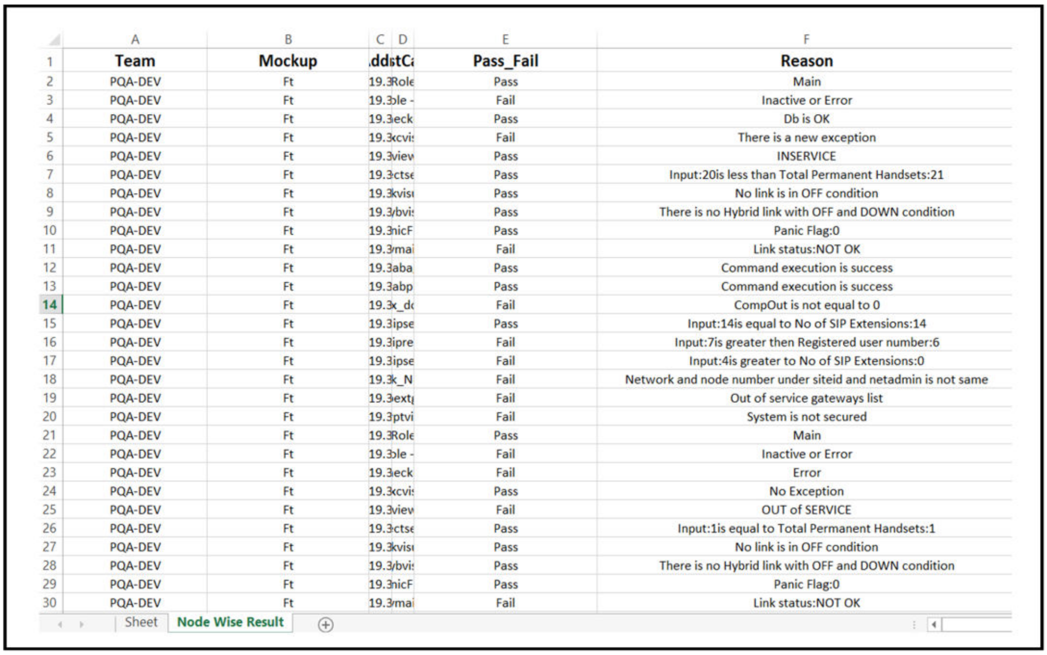
S.No	Team	Mockup_Name	IP_address	Login	Password
1	PQA-DEV	Ft	1[redacted]10	m[redacted]d	n[redacted]
2	PQA-DEV	Ft	1[redacted].20	m[redacted]	F[redacted]o
3	PQA-DEV	Ft	1[redacted]30	m[redacted]	F[redacted]ee
53	PQA-DEV	Upmc	1[redacted].10	m[redacted]	m[redacted]
54	PQA-DEV	Upmc	1[redacted].20	m[redacted]	m
55	PQA-DEV	Alpha	1[redacted]40	m[redacted]	m[redacted]

	Time	Action	Message	Duration / Fetch
56	16:26:45	SELECT * FROM test[redacted]e_data.team_wise2 LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
57	16:26:47	SELECT * FROM test[redacted]e_data.team_wise1 LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
58	16:26:49	SELECT * FROM test[redacted]e_data.team_wise LIMIT 0, 1000	5 row(s) returned	0.000 sec / 0.000 sec
59	16:27:04	SELECT * FROM test[redacted]e_data.everyday_calculation LIMIT 0, 1000	9 row(s) returned	0.000 sec / 0.000 sec
60	17:14:53	SELECT * FROM test[redacted]e_data.adnodewise_table LIMIT 0, 1000	21 row(s) returned	0.000 sec / 0.000 sec
61	12:18:11	SELECT * FROM test[redacted]e_data.test_data LIMIT 0, 1000	11 row(s) returned	0.000 sec / 0.000 sec
62	12:18:42	SELECT * FROM test[redacted]e_data.everyday_calculation LIMIT 0, 1000	9 row(s) returned	0.000 sec / 0.000 sec
63	12:18:59	SELECT * FROM test[redacted]e_data.nodewise_table LIMIT 0, 1000	173 row(s) returned	0.031 sec / 0.000 sec

Fig. 6. Input from MySql db

Fig. 7 shows the input table from MySQL db. These inputs are used in python script to login the each node through Telnet.

Fig. 8 shows the excel report. This report contains Team name, Mockup name, IP address, and Test case, Pass or Fail and Reason for the Pass or Fail. This excel report will get mailed to the concerned person to check the quality of the nodes. We can also view the short version of the results in webpage by clicking previous results from the main page.



	A	B	C	D	E	F
1	Team	Mockup	.dd	stCa	Pass_Fail	Reason
2	PQA-DEV	Ft	19.3	Role	Pass	Main
3	PQA-DEV	Ft	19.3	ble -	Fail	Inactive or Error
4	PQA-DEV	Ft	19.3	eck	Pass	Db is OK
5	PQA-DEV	Ft	19.3	kcvis	Fail	There is a new exception
6	PQA-DEV	Ft	19.3	view	Pass	INSERVICE
7	PQA-DEV	Ft	19.3	ctse	Pass	Input:20is less than Total Permanent Handsets:21
8	PQA-DEV	Ft	19.3	kvisı	Pass	No link is in OFF condition
9	PQA-DEV	Ft	19.3	ybvis	Pass	There is no Hybrid link with OFF and DOWN condition
10	PQA-DEV	Ft	19.3	nicF	Pass	Panic Flag:0
11	PQA-DEV	Ft	19.3	ymai	Fail	Link status:NOT OK
12	PQA-DEV	Ft	19.3	aba	Pass	Command execution is success
13	PQA-DEV	Ft	19.3	abp	Pass	Command execution is success
14	PQA-DEV	Ft	19.3	x_dc	Fail	CompOut is not equal to 0
15	PQA-DEV	Ft	19.3	ipse	Pass	Input:14is equal to No of SIP Extensions:14
16	PQA-DEV	Ft	19.3	ipre	Fail	Input:7is greater then Registered user number:6
17	PQA-DEV	Ft	19.3	ipse	Fail	Input:4is greater to No of SIP Extensions:0
18	PQA-DEV	Ft	19.3	k_N	Fail	Network and node number under siteid and netadmin is not same
19	PQA-DEV	Ft	19.3	extg	Fail	Out of service gateways list
20	PQA-DEV	Ft	19.3	ptvi	Fail	System is not secured
21	PQA-DEV	Ft	19.3	Role	Pass	Main
22	PQA-DEV	Ft	19.3	ble -	Fail	Inactive or Error
23	PQA-DEV	Ft	19.3	eck	Fail	Error
24	PQA-DEV	Ft	19.3	kcvis	Pass	No Exception
25	PQA-DEV	Ft	19.3	view	Fail	OUT of SERVICE
26	PQA-DEV	Ft	19.3	ctse	Pass	Input:1is equal to Total Permanent Handsets:1
27	PQA-DEV	Ft	19.3	kvisı	Pass	No link is in OFF condition
28	PQA-DEV	Ft	19.3	ybvis	Pass	There is no Hybrid link with OFF and DOWN condition
29	PQA-DEV	Ft	19.3	nicF	Pass	Panic Flag:0
30	PQA-DEV	Ft	19.3	ymai	Fail	Link status:NOT OK

Fig. 7. Output in Excel Sheet

VIII. Conclusion

The contribution of this paper is, understanding the time consumption of manual maintenance in which human intervention is needed most. By using automated tool, this could reduce the lot of time consumption for logging in individually all nodes through Telnet and for the execution of various commands one by one in each node.

IX. Future Scope

This project is all about maintaining the quality of the node by checking whether the test cases (commands) are passed or failed. In future, this can be taken forward by adding automated correction of failed test cases. This could reduce more human involvement and time consumption.

Appendix

```
from django.shortcuts import render, render_to_response
from django.http import HttpResponse
from django.template.loader import get_template
from django.views.decorators.csrf import csrf_exempt
from django.template import Context
import pymysql
from Automation import Periodic, Adhoc
@csrf_exempt
def schedule(request):
tp=get_template('schedule.html')
html=tp.render()
a=0
if request.method =="POST":
if "form1" in request.POST:
day=request.POST.get("day")
tym=request.POST.get("time")
print(day)
print(tym)
if day=="sunday":
a=6
b="sunday"
elif day=="monday":
a=0
b="monday"
elif day=="tuesday":
a=1
b="tuesday"
elif day=="wednesday":
a=2
b="wednesday"
elif day=="thursday":
a=3
b="thursday"
```

```
elif day=="friday":
a=4
b="friday"
elif day=="saturday":
a=5
b="saturday"
Periodic.success(a,b,tym)
temp=str(tym).split(":")
hr=str(temp[0]).strip()
mint=str(temp[1]).strip()
tp1 = get_template("runningtask.html")
html1= tp1.render(Context({'runningday': b,'hr': hr,'mint': mint}))
return HttpResponse(html1)
#return render_to_response("EverydaySchedule.html")
return HttpResponse(html)
global scheduler
scheduler = BackgroundScheduler()
scheduler.start()
@csrf_exempt
def success(a,b,tym):
#schedule1(a,b,tym):
global hr, mint, scheduled_day, runningday
scheduled_day=a
runningday=b
temp=str(tym).split(":")
hr=str(temp[0]).strip()
mint=str(temp[1]).strip()
print(hr,mint)
global my_job
global x
x=0
my_job=scheduler.add_job(job,'cron',)
print(my_job)
x=1
print(x)
#'cron', day_of_week=str(scheduled_day), hour=str(hr), minute=str(mint)
#'interval', seconds=15
print("Successfully Scheduled")
print(scheduler.get_jobs())
@csrf_exempt
def running_task(request):
tp=get_template("runningtask.html")
html= tp.render(Context({'runningday': runningday,'hr': hr,'mint': mint}))
return HttpResponse(html)
```

REFERENCES

[1] Sandep Singh, R.A, Khan, Alka Agarwal, Department of Computer Science, KMCUAF University, Lucknow, “Role of VND for designing of an infrastructure based software defined network”, International

Conference on Computing, Communication and Automation (ICCCA), 15-16 May 2015, IEEE DOI 10.1109/CCAA.2015.7148414.

[2] K Jambunatha, Computer Networking Engineering, “Design and implement Automated Procedure to upgrade remote network devices using Python”, IEEE International Advanced Computing Conference (IACC), 2015, IEEE DOI 10.1109/IADCC.2015.7154701.

[3] Di Liu, Pin Wang, University of Electronic Science and Technology of China, “Use Python API to automate script based on Open Stack platform”, Interational Computer Conference on Wavelet Actve Media Technology and Information Processing (ICCWAMTIP), 18 – 20 Dec 2015, IEEE DOI 10.1109/ICCWAMTIP.2015.7494032.

AUTHOR(S) BIOGRAPHY

Prem Kumar A, pursuing M.tech Communication Engineering in Vellore Institute of Technology (VIT), Chennai, India. Previous research works includes An Efficient Routing Mechanism for VANETS, Traffic Light Indication in Vehicles at a Four way Junction and An impressive Greedy Routing Mechanism. Acquired Bachelors of Engineering in Electronics and Communication from G.K.M College of Engineering and Technology, Anna University, Chennai.

Jayavignesh Thyagarajan Is a faculty in Vellore Institute of Technology (VIT), Chennai, India. His research interests include Routing Protocols Design, Analysis of Medium Access Protocols and Congestion Control mechanism in the area of Wireless Adhoc, Sensor, Mesh, Vehicular and Opportunistic networks. He acquired his Masters in Engineering from Madras Institute of Technology, Anna University and secured First Rank with a Gold medal in Communication and Networking program. He obtained his Bachelors of Engineering degree in Electronics and Communication from College of Engineering, Guindy, Chennai.