INTERNATIONAL RESEARCH JOURNAL ON ADVANCED SCIENCE HUB
e-ISSN : 2582 - 4376
Open Access
RSP SCIENCE HUB
(The Hub of Research Ideas)
Available online at www.rspsciencehub.com

Special Issue of Second International Conference on Advances in Science Hub (ICASH 2021)

Implementation of End to End Automation for BTS commissioning using Python

Rani RN[1], HV Kumaraswamy[2], Jeyaraj Chellapandi[3]
[1]PG Student, Department of ETE, RVCE, Bengaluru, Karnataka
[2]Professor and Associate Dean, Department of ETE, RVCE, Bengaluru, Karnataka
[3]Squad Lead, Nokia Solutions and Networks India Pvt. Ltd, Bengaluru, India
ranijsb@gmail.com[1], kumaraswamyhv@rvce.edu.in[2], jeyaraj.chellapandi@nokia.com[3]

Abstract

Network Management Systems (NMSs) are the innovative products used by telecommunication companies to monitor, control, analyze and manage telecommunication networks. Base Transceiver Station (BTS) is a network element which facilitate wireless communication between user equipment and network. BTSs can be integrated with NMS and all related applications available in NMS can be used over BTS. One such application is software upgrade or downgrade in BTS which is termed here as BTS commissioning. This BTS commissioning process involves multiple steps to be performed for successful software upgrade or downgrade in BTS manually through NMS. In this paper methodology to implement end to end automation of BTS commissioning using python is discussed. GUI is developed using JavaScript, jQuery, Flask, Python, HTML and CSS. The reason for using Python for this website is because it is powerful, simple and has many libraries. Developed feature is tested by integrating a test BTS with Nokia's NMS named NetAct.

Keywords: Automation, Python, Selenium, BTS, NMS, Flask, Web technologies, Paramiko

1. Introduction

A Automation can be defined as the usage of technology for executing recurring process in business where manual effort can be greatly reduced. Automation is done to increase efficiency, to streamline process and to attenuate costs. By automating the recurring manual process great number of advantages can be achieved such as completing tasks in faster and cheaper way, enhancing accuracy, reducing errors, increasing consistency, reducing risks, enforces data driven decisions, dependency on employees will be reduced and streamlining of process will be achieved. Network Management System is an innovative product for multi-vendor and multi-technology networks. NMSs are used to monitor, control, analyze and manage telecommunication networks. Most of the NMS uses virtualized hardware platform. With NMS both the network and services within the network are managed centrally so that the operator can view network element failure, service quality indicators and traffic from one screen. Most of the NMS supports Network Element (NEs) in mobile radio and core, Wi-Fi, IoT, public safety and telco cloud. A BTS is a device that enables wireless communication between a network and user equipment (UE). UEs are wirelessly connected devices like as laptops, phones, and other electronic gadgets. Any wireless communication technology, such as CDMA, GSM, Wi-Fi, or other wide area technology, can be used to create the network. BTS requires regular software upgrades or downgrades in telecommunication companies for testing or customer demo purpose with the software package

released by R&D teams. This process of software upgrade or downgrade in BTS is performed by integrating BTS with NMS manually. This complete process can be End to End (E2E) automated using python. Motivations for implementing this use case are to avoid human resource efforts, to prevent human errors, to reduce human intervention, to reduce time taken for each BTS software upgrade/downgrade, to achieve streaming of process and to avoid dependency on human resource.[1-4].

2. Background

In this section manual work and series of steps involved in BTS commissioning is discussed. Initially the BTS which needs to be upgraded or downgraded with software package needs to be integrated with the NMS. Once integration is done next step is to perform precheck, to make sure BTS is in proper working condition or not. If BTS is in proper working condition, then only software upgrade or downgrade can be performed on it. Next step is to download the software package to local machine and this software package needs to be copied to the NMS's dedicated directory and unzipping the software package needs to be performed. Once importing of software package is completed in NMS, it will be followed by downloading the same package to BTS from NMS and performing provisioning. Provisioning completes the BTS commissioning process and post check will be done to verify the proper functionality of BTS after software upgrade or downgrade. Glitches are involved in all the intermediate steps while performing BTS commissioning manually.[5-8]. NMS uses virtual hardware platforms; in telecommunication companies the installation of NMS application is done in number of virtual machines (VMs) and each VM is dedicated to specific products and these VMs are termed as labs. Integration of BTS with labs will be done based on project requirements for testing software packages released by R&D for customer demo or trail purposes. Human errors are possible in the logging in to specific lab and BTS and also in identifying particular integrated BTS and while downloading the proper software versions from repository. Human efforts are needed to manually login to labs and BTSs and in various process like checking for compatibility, identifying integrated BTSs, downloading the required package versions from repository and verifying proper functionality of BTS after upgrade or downgrade of software. Since the software packages will be several GBs download time required will be more and couple of Linux commands involved in the BTS commissioning process takes several minutes to give response. Since automation VM will be directly connected with telecommunication company network, time required will be less. So, the time required by an employee for BTS software upgrade or downgrade will be more when done manually. Further if multiple requests are there for BTS commissioning through different labs and for different BTS, then streaming this process will not be possible if it is done manually.[9-13].

3. Technology Introduction

The technologies used in the development of this use case are Python, Flask, HTML, CSS, jQuery, JavaScript, Selenium, SCP and Paramiko.

3.1 Web Technologies

Graphical User Interface required for user access of this application is developed using HTML, CSS and jQuery. HTML is Hyper Text Markup Language which is used to develop the template of the GUI and the same template is styled using Cascading Style Sheets. The static GUI developed using HTML and CSS is made dynamic as per application requirements by using jQuery. jQuery is a JavaScript library which has ability to keep the code simple, reusable, clear and readable.

3.2 Python and Flask

Python is powerful, simple and provides wide set of libraries which helps faster development and easy implementation of user requirements. In this application Python is used for backend computing and Python version used is Python 3.8.

Flask is a Python web framework that is lightweight. It facilitates the creation of Web-applications such as blog or wiki by providing modules, libraries, and tools to the users. Flask version used is 1.1.2.

3.3 Selenium

Selenium is a free open-source automated testing tool for validating web applications across several browsers and platforms. Selenium Test Scripts can be written in a variety of programming languages, including Python, C#, Java, and others. Selenium is used in this application to interface with the GUI

and download the software package mentioned by the user from Nokia's Work Flow Tool (WFT). Selenium Version used is 3.141.0

3.4 Paramiko

Paramiko is a Python package that allows you to interact with remote servers. It's a Python implementation of the SSHv2 protocol that allows for secure client and server communications. The protocol includes the ability to use the encrypted tunnel to open arbitrary channels to remote services. Paramiko version used is 2.7.1.[14-20].

3.5 Rest API

In this application a Rest API is developed and providedto the users to trigger BTS commissioning process. Rest API means Representational State Transfer is an architectural style for an application program interface (API) which is based on stateless HTTP protocol. It uses this HTTP protocol requests to GET, PUT, POST and DELETE data. In this use case user has to enter details regarding NMS, software version and BTS id in Rest API URL in order to send request to the script.

4. Methodology

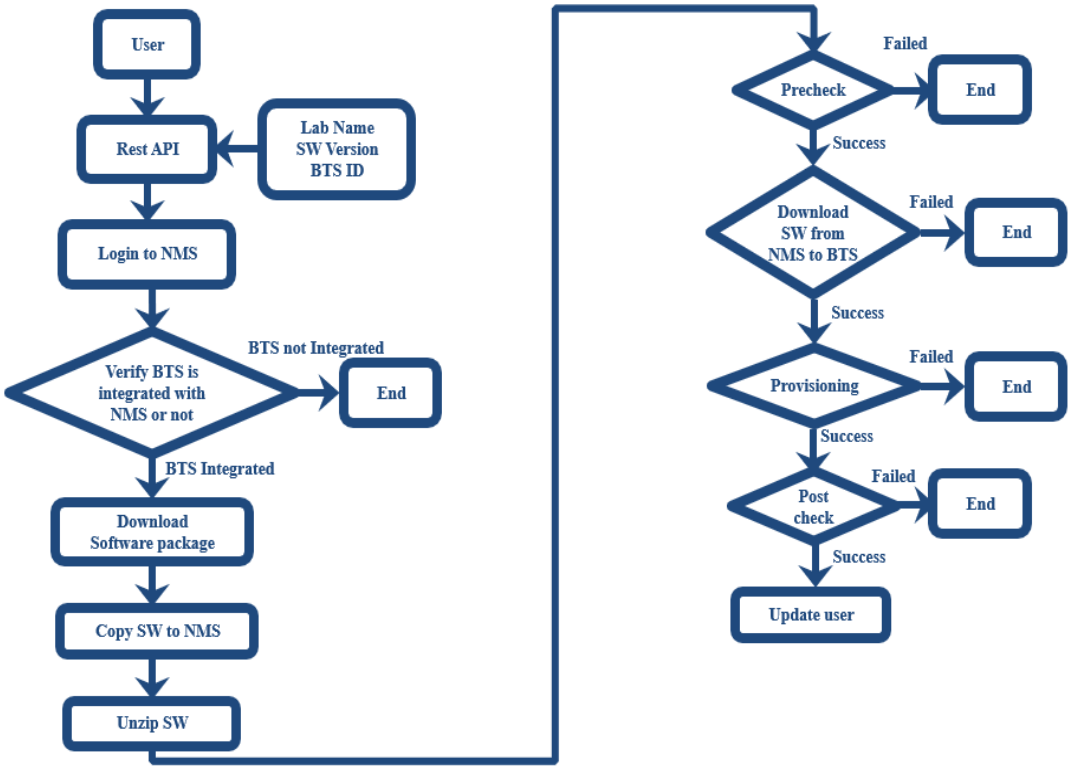

Fig.1. E2E Methodology of BTS Commissioning

Fig.1.shows the complete E2E methodology of BTS commissioning. As shown in this figure user will be provided with developed Rest API URL which will be mapped with the server address where this complete script resides. User has to fill the details of NMS, software package version and BTS ID. Once user fills these details in URL and hits the same the request will be sent to the web application to start the BTS commissioning process. As soon as the request comes next step is to login to the NMS and to check whether the requested BTS to which software package needs to be downgraded or upgraded is integrated with the NMS or not. If BTS is not integrated, then BTS commissioning process cannot be performed, and the script will be terminated. If BTS is integrated with the NMS, then the requested software version by the user will be downloaded from its repository to the local machine using Selenium Python library. After downloading the software package, same will be copied to the NMS using SCP Python library. Copied software package will be in zip format. So, this package will be unzipped to proceed with further processing. Before performing any operation on BTS, first we need to make sure that the BTS is up or not. If BTS is not in proper working condition the software upgrade or downgrade cannot be performed in the BTS. If

BTS is up the next step is to copy the unzipped software package from NMS to the BTS and to perform Provisioning. Provisioning is same as activating the new software package. For example, whenever we have a software upgrade request in mobile, initially software version will be downloaded then mobile will get restarted and the downloaded software version will bet activated. This same will happen in provisioning step. Here the BTS will get activated with newer software version. Once provisioning step is successful, we need to perform post check on the BTS to make sure that the BTS continues to be in proper working condition after the activation of newer software version. In order to avoid BTS service down scenarios. In case of any glitch in working condition of BTS. Software can be rolled back to the older software version. Success in each of these intermediate steps results in proceeding with the next step and completing the BTS commissioning process successful.[21-23].

5. Results

Developed functionality is tested using Nokia's innovative Network Management System called NetAct. A test BTS is considered and integrated with a test VM where NetAct is installed. By using Rest API developed for this application a request is triggered for software upgrade for the test BTS and results are monitored through both NMS and GUI of the use case. All the intermediate results in the steps mentioned in methodology are captured and displayed on GUI for user access.

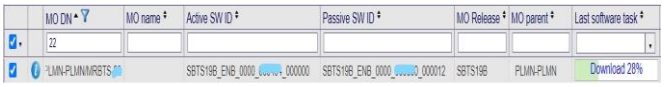

Fig. 2. Download progress status in NMS

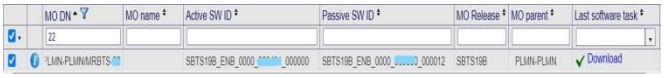

Fig. 3. Download completion status in NMS

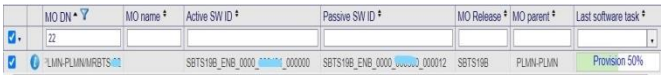

Fig. 4. Provision progress status in NMS

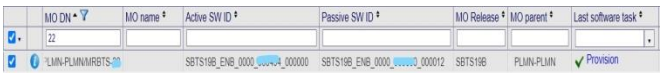

Fig. 5. Provision completion status in NMS

Fig.2 and Fig.3 shows the progress and completion status of downloading the software package from NMS to BTS respectively. Fig.4 and Fig.5 shows the progress and completion status of provisioning the software package in BTS respectively. Once sftware provisioning is completed the newer software version will be activated in the BTS and this can be viewed by checking the software version by loging in to BTS GUI.[24-26].

Conclusions

In this paper automation for BTS software upgrade/downgrade is discussed. Automating a task which involves constant steps by human resource results in major advantages in terms of avoiding human errors, reducing human efforts, increasing time efficiency, achieving expected outcomes and also can achieve streamlining the process. All the steps involved in implementing automation for BTS commissioning are explained in detail and also this use case involves individual components like NMS, BTS and various coding technologies, maintaining interaction between these components and also objective to achieve BTS commissioning is shown and corresponding results are depicted.

References

[1]. Maksim Vorontsov, Radmir Sadikov, 'Automation of Message Sending using Specialized Software', 2021 IEEE Conference of Russian Young Researches in Electrical and Electronics Engineering.

[2]. Md Shahedul Alam, Udayanto Dwi Atmojo, Jan Olaf Blech, Jose L. Martinez Lastra , 'A REST and HTTP-based Service Architecture for Industrial Facilities' ,2020 IEEE.

[3]. Bob Maulana Adam, Adnan Rachmat Anom Besari, Mochamad Mobed Bachtiar, 'Backend Server System Design Based on REST API for Cashless Payment System on Retail Community', 2019 IEEE International Electronics Symposium (IES).

[4]. Prasad Narode, Samruddhi Kalekar, Sanket Sanap, Harmeet Khanuja, 'Smart Parking Management System', 2019 IEEE.

[5]. Yunwei Zhao ,Xin Wan, 'The Design of Embedded Web System based on REST Architecture' , 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC 2019).

[6]. Usha Tiwari, Shabana Mehfuz, Satyam Sharma, Vinayak Triguna Pandey, 'Design of Python Based Lost and Found Website for College Campus', 2019 IEEE.
[7]. Mohammad Robihul Mufid, Arif Basofi, M. Udin Harun Al Rasyid, Indhi Farhandika Rochimansyah, Abdul rokhim, 'Design an MVC Model using Python for Flask Framework Development', 2019 IEEE International Electronics Symposium (IES).
[8]. Yingchun Xia, Lanxiang Jiang, Ruien Zhao, Quan Liu, 'Design and Implementation of Coverage Testing System Based on Python', 2018 IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference(IAEAC 2018).
[9]. Kesinee Boonchuay, Youppadee Intasorn, Kritwara Rattanaopas, 'Design and Implementation a REST API for Association Rule Mining', 2017 IEEE 14th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON).
[10].S M Sohan, Frank Maurer, Craig Anslow, Martin P. Robillard, 'A Study of the Effectiveness of Usage Examples in REST API Documentation', 2017 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC).
[11].Faisal Bin Al Abid, A.N.M Rezaul Karim, 'Cross-Platform Development for an online Food Delivery Application', 2017 IEEE.
[12].Paul MIHAILA, Titus BALAN, Radu CURPEN, Florin SANDU, 'Network Abstraction and Automation using Python programming Methods,' MACRo 2017 – 6th International Conference on Recent Achievements in Mechatronics, Automation, Computer Science and Robotics.
[13].Baphumelele MASIKISIKI, Siyabulela DYAKALASHE and Mfundo Shakes SCOTT, 'Network Monitoring System for Network Equipment Availability and Performance Reporting', 2017 IEEE.
[14].Patrick Vogel, Thijs Klooster, Vasilios Andrikopoulos, Mircea Lungu, 'A Low-Effort Analytics Platform for Visualizing Evolving Flask-Based Python Web Services', 2017 IEEE Working Conference on Software Visualization.
[15].Andrew John Poulter, Steven J. Johnston, Simon J. Cox, 'Using the MEAN Stack to Implement a RESTful Service for an Internet of Things Application', 2016 IEEE.
[16].Miroslav Bures, Martin Filipsky 'SmartDriver: Extension of Selenium WebDriver to Create More Efficient Automated Tests', 2016 IEEE.
[17].K Jambunatha, 'Design and Implement Automated Procedure to Upgrade Remote Network Devices Using Python', 2015 IEEE
[18].Zhixiang Niu, Cheng Yang, Yingya Zhang, 'A Design of Cross-terminalWeb System Based on JSON and REST', 2014 IEEE.
[19].Xiaoyue Wang, Bin Xu, Rui Gu, 'The Application of Code Reuse Technology Based on the MVC Framework', 2013 IEEE International Conference on Computer Sciences and Applications.
[20].Dragos-Paul Pop, Adam Altar, 'Designing an MVC Model for Rapid Web Application Development', 24th DAAAM International Symposium on Intelligent Manufacturing and Automation, 2013.
[21].Dandan Zhang, Zhiqiang Wei,Yongquan Yang, 'Research on Lightweight MVC Framework Based on Spring MVC and Mybatis', 2013 IEEE Sixth International Symposium on Computational Intelligence and Design.
[22].D. Radosevic , I. Magdalenic , 'Python Implementation of Source Code Generator Based on Dynamic Frames', MIPRO 2011, May 23-27, 2011, Opatija, Croatia.
[23].Chakrit Wannachakrit, Toni Anwar, 'Development of Network Management System for Network Services Provider', 2011 5th Malaysian Conference in Software Engineering (MySEC).
[24].Mi Xifeng,Fan Yuanyuan. 'Network Management System Design and Implementation', 2011 IEEE.
[25].Andreas Bruns, Andreas Kornstadt, Dennis Wichmann, 'Web Application Tests with Selenium ', 2009 IEEE.
[26].G. Miguel, "Flask Web Development," O'Reilly Media, Inc. 2014.