**Problem Statement- Networking Bot**

**Introduction-**

There is a company X that supplies internet connection to the households. They want us to build a system where they can differentiate between the snag happening at their end in internet supply or at the receiver’s end. They want to figure out whether their switches installed are working properly or not.

**Explanation-**

In a household, there is a modem supplied with the connection to a particular internet Service Provider (ISP). It has 4 lights on it. Two for the DNS Server (DSL) and two for the Internet. If the light is on in both the sections, we are able to use the internet on the computer. However, if the DNS light is off, we say that ‘internet connectivity is down from the back end’ and no internet is supplied to our homes. If the Internet light is off, we say that ‘machine/modem particularly has some technical issues’.

The similar is the situation here.

Just that we play at the up-scaled level of a switch and a port and try to find the problem at the end port of the network system.

Switch: A switch is a multiport bridge with a buffer and a design that can boost its efficiency (a large number of ports imply less traffic) and performance. It’s a Data Link Layer (DLL) device.

Port: A placeholder where the connection to the ether-net cable is made.

The problem is to figure out which port is working and which is not only when a complaint is triggered.

**Proposed Solution-**

You receive a complaint from the consumer end (end-user of X) that the internet at their end is not working. So, this complaint triggers a bot engine placed on the server to check the health condition of the ports associated and controlled by the particular Service Provider X. This bot engine runs a ping request to that particular switch where those ports are located. It verifies the condition of all ports associated with that particular switch depending on the level the X is in the network architecture. If all is OK, it reports back that the complaint is not at its end but the end-user. However, if it is reported FAULTY, it simply sends the request to the Network Administrator.

**USP:** Basically, we are eliminating the presence of call centres and establishing a direct connectivity from the X server to the point of complaint, thus, registering only those complaints further that are genuine and to the concern of the company X.

**Proposed Simulation-**

The situation is to simulate an internet connection over two-layer switched network architecture as shown in the diagram. The wires used will be ether-net cables and the test system will be a laptop. The complaint system will be run on the server from where it will trigger a health status check-up of the concerned port locations. Returning a result: SUCCESS or FAILURE depending on the result and marking the complaint as genuine to proceed further to the Network Administrator for the concerned steps.

NETWORK ARCHITECTURE:

X Server

(BOT ENGINE)

How to read the network architecture location address?

For Server of X, there are two ports namely: X-L01 and X-L02

For Switch-S1, there are two ports namely: S1-L01 and S1-L02

For Switch-S2, there are two ports namely: S2-L01 and S2-L02

For Ethernet cables, the hierarchy is from top to bottom starting from C1, C2 and C3.

So, to reach to the Laptop-01, the internet route will provide an address as: X-L01:S1-L02:S2-L02:LAPTOP-01.

Also, the address to the Laptop-01 will be: X-L02:LAPTOP-01

LAPTOP-02

Switch-S1

Switch-S2

LAPTOP-01

So, there are complaints received by the company X, that a particular connection in the area (LAPTOP-01) is not working. Laptop-02 has the application deployed to track route connectivity issues at the switches. That complaints get registered via LAPTOP-02 with complaints database residing on the X Server for the time being. A registry of network routes to be maintained on x-Server based on the location co-ordinates of LAPTOP-01. LAPTOP-02 initiates a trigger to the NetworkBot Engine placed on the X-Server to

* Determine the route to track
* ping requests till the address:

X-L01:S1-L02:S2-L02. This is a single path for we know the “exact route” from the route registry. Similar scenario is proposed in production environment. Exact routes shall be known to ping to.

**Flowchart of the process:**

On Clicking the ‘Fault Detection’ button on the App, a request is sent to the BOT ENGINE to initiate the BOT to find the FAULTY PORT index from the registry based on the Location co-ordinates.

Complaint on saving gets registered at the Database of the X with the details of the nature of complaint and complainant.

Complaint registered on the X service website (just considering it as an option)

**Email Alert**

BOT on X-server basis route identified, sends a ping request to the Faulty reported port as much further layer the registry supplies them with the route.

“No complaint on our part.”

PING=SUCCESSFUL

Complaint goes to the Network Admin to be resolved.

PING=FAILURE

**Email Alert(if desired by the customer from the ISP end.)**

1. **Software level:**
2. Design of the software: it’s a web app that works with the internet supply.
3. Aim of the software: To figure out whether port is working properly (during complaints being registered.)
4. Pre-requisite – The device through which X-Server/Bot Engine is connected should have secure connection to X-Server/Bot Engine via VPN.
5. Layout of the software:
6. UI/UX design of the software:
7. Screen-01: This is for customer to lodge the complaint

Logo of company Name of company

Complaint Redressal System

Customer ID

Register🡺

Powered by Parity InfoTech Solutions private Limited

On clicking on Register, the complaint gets saved in a database whose Concern status is False along with the Verification status also put to False. CustomerID links us to the IP address of the device also. So, our problem is solved for getting an IP.

1. Screen-02: This is for the authenticated employee who have open permitted access to the server.

Logo of company Name of company

Employee Login

Password

Powered by Parity InfoTech Solutions private Limited

1. Screen-03: The Network engineer sees into his screen to find a record of complaints.

Logo of company Name of company

S. No. Customer\_ID Date of complaint Actions Status Of Concern

1 ID\_1234 2020.08.10 Connect Success No

2 ID\_5678 2020.08.10 Connect Failure Yes

Powered by Parity InfoTech Solutions private Limited

1. When he clicks on the button ‘connect’, a bash file is created that connects the server to the point of problem or the end-user. A ping request is made automatically to that area of problem.
2. His actions can also be recorded this way as to when he responded to the complaints. (if you want to)
3. He logs out after seeing the success msg and forwards the request to the Network administrator if found the failure msg and thus marking the request as of concern.