**Task 1:**

Assuming there are "N" number of devices(having Linux on them) of type "X" where N > 0, out there in the field(distributed devices all over the world with limited internet connectivity, so they are not in same network. Example - a mobile phone, a raspberry pi ) .Each device has two types of network interfaces - One is Ethernet port and second is wireless interface(wifi adapter) . Each interface would have a MAC address .

Case 1 :

We have to build a system to collect information from the device at regular interval . How would you model the system to make this happen . Following can be the type of information we are collecting from the devices -

* Average cpu usage per/day
* Average memory used per/day
* disk space of the device
* ethernet mac id
* wifi interface mac id

Task : How would you design the overall architecture to solve this problem .

**Solution:**

To gather the information from all the servers deployed in different regions with limited connectivity. Following requirements should be completed to build the system:

1. First of all, there should be a public server configured properly with all other servers, so that it should be able to connect with all other servers and retrieve data from them. Also it should have good Internet connectivity. It should be referred as a host server.
2. Install Prometheus on the host server which is a monitoring platform that collects metrics from monitored targets by scraping metrics HTTP endpoints on these targets.
3. Next Step is to install ‘node exporter’,Node Exporter is a single binary and has a configurable set of collectors for gathering various types of host-based metrics. By default, collectors gather CPU, memory, disk, and other metrics and expose them for scraping.
4. After configuration, execute ‘node exporter’ on all the target servers required to monitor.
5. Create a ‘prometheus.yml’ file on the host server which includes entries of all the target servers to enable communication for data flow.
6. Run prometheus and it will retrieve all the data from the servers. Different parameters are available to collect desired information.

**Architecture:**



**Output:**

