Network Monitoring Tool

Table of Contents

1.0 Overview

Network monitoring tools and systems constantly monitor a network's health and reliability by tracking and logging network parameters and searching for trends. A network monitoring system will watch and compare data transmission rates (throughput), uptime/downtime rates, error rates, response times to inputs and requests (both user-generated and automated), and use-time percentages to parameter thresholds that users set in advance. When levels reach those thresholds, the network monitoring system triggers an alarm and initiates network fault management processes.

There is more than one way network traffic monitoring tools as part of a network monitoring system might alert administrators to performance and security problems that can harm the network. Triggers are events that will generate alarms in the system. An event might refer to a deviation from mean value of a parameter, a crossed threshold parameter value, a change in the state of a node.

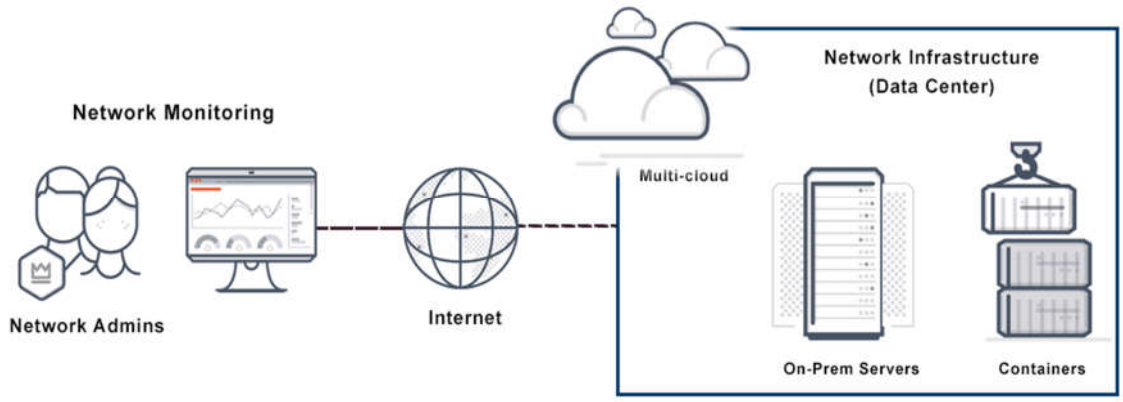
1.1 What is Computer Network?

Facility makes possible communication between multiple computers or network devices connected together to Exchange / transfer data and share resources via the same or different medium.

1.2 What Is Network Monitoring tool?

Network Monitoring, a subset of network management, is a systematic attempt by a computer network to identify slow or failing components before they cause problems. For example, crashed, frozen, or overloaded servers; failed switches; failing routers; and other troublesome components can all potentially cause outages or network failures. Should some problem arise and trigger an outage, it is the role of the network monitoring system to alert the network administrator in a timely way.

Typically, administrators monitor and manage a network using network monitoring tools and software applications. These network monitoring services help users monitor performance, and detect whether a web server is properly connected to worldwide networks and functioning as expected. In fact, many network performance monitoring tools also offer an end-to-end visualization of the networks and the applications.



1.3 What Is purpose of using Network Monitoring tool?

- *Evaluating network components like switches, routers, firewalls, and servers for fault and performance*

- *Network monitoring tool gives you the visibility you need to stay one step ahead of potential issues.*
- *Network monitoring tools focused on improving network performance.*

2.0 Scope of project

- *network monitoring tools is one of the best ways to prevent these network outages. Monitoring tools not only provide visibility into the network issues but also help businesses to stay one step ahead.*

- *Open source java script network monitoring tool and monitor network and all its components.*

- *Develop network monitoring tool to monitor network and its components and identify sources of network performance problems.*

- *Tool will monitor network ,all its component and measure network performance. Tool will not intercept information and privacy hack.*

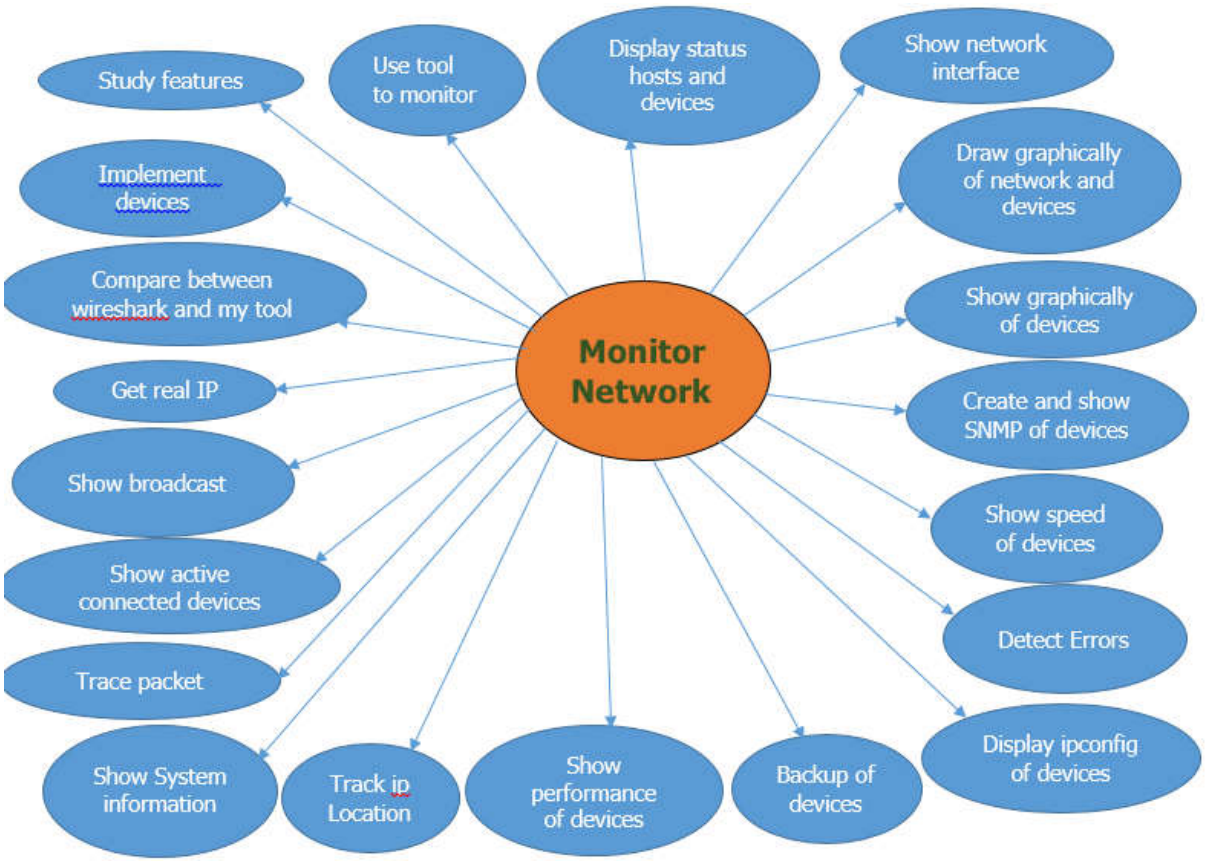
- *Network monitoring tool must be deliver at certain time ,within certain budget and with feature agreement .*

- *back up of devices using ip config and interface of devices and speed of devices.*

- *Detect errors of devices and problems in netwoks.*

- *Design network graphically and show graphically.*

2.1 Use case diagram



2.2 business case

- *Reduce network outages by quickly identifying exactly where network faults occur.*
- *Provide end-to-end visibility across the network.*
- *Increased network uptime and reducing downtime*
- *Optimise network performance*

2.3 Work plan

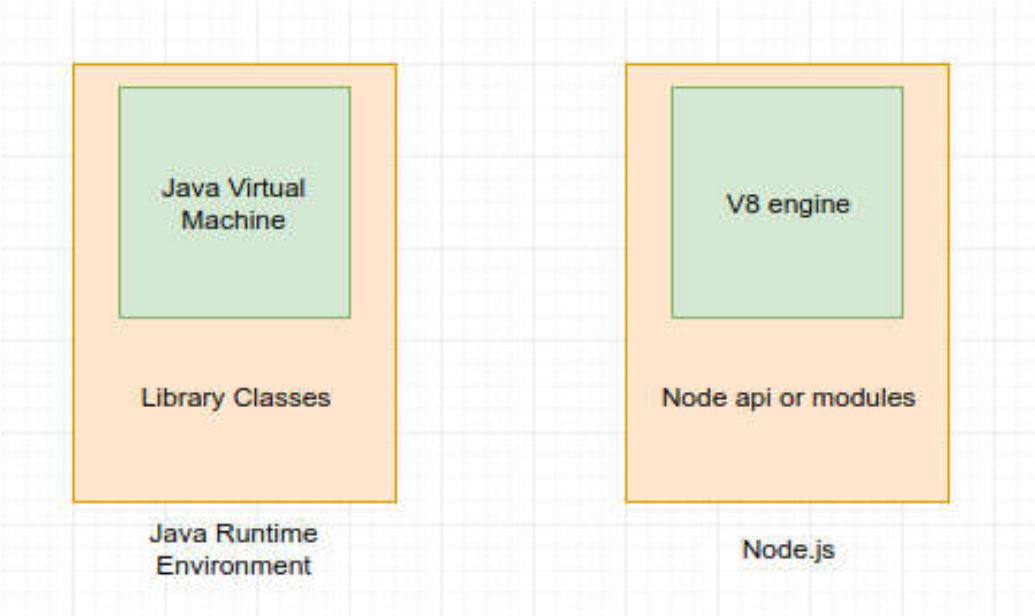
Task Mode	Task Name	Duration	Start	Finish	Predecessors	Resource Names
Auto Scheduled	**Main features for network**	**63 days**	**إ ٢١/١١/٠١**	**أ ٢٢/٠١/٢٦**		
Manually Scheduled	Overall search and study to make required features	14 days	إ ٢١/١١/٠١	خ ٢١/١١/١٨		all team
Manually Scheduled	tool is used monitor network and its components	13 days	ج ٢١/١١/١٩	ث ٢١/١٢/٠٧	2	Ramy and Abdelrahman
Manually Scheduled	Know the status of devices and amount of traffic and statistics	16 days	أ ٢١/١٢/٠٨	أ ٢١/١٢/٢٩	3	Ramy and Omar
Manually Scheduled	Fronted Ui desgin for our tool	15 days	خ ٢١/١٢/٣٠	أ ٢٢/٠١/١٩	2;3;4	Abdelrahman
Manually Scheduled	Presentation,Documention,and Ghantt chart	5 days	خ ٢٢/٠١/٢٠	أ ٢٢/٠١/٢٦	2;3;4;5	Abdrahman,Mustafa , Mahmoud and Mohanad
Manually Scheduled	Identify sources of network performance problems	28 days	أ ٢١/١٢/٠٨	ج ٢٢/٠١/١٤	3	Hossam and Mohanad and Michael
Auto Scheduled	**Protocol**	**100 days**	**خ ٢١/١٢/٣٠**	**أ ٢٢/٠٥/١٨**		
Manually Scheduled	Use necessary protocols on various network	13 days	خ ٢١/١٢/٣٠	إ ٢٢/٠١/١٧	4	Mostafa and Mahmoud
Manually Scheduled	Use protocols analysis software to monitor traffic and discover network problems	20 days	ث ٢٢/٠١/١٨	إ ٢٢/٠٢/١٤	9	Mostafa , Mahmoud and Mohanad
Manually Scheduled	Backup the status of every device	15 days	ث ٢٢/٠١/١٨	إ ٢٢/٠٢/٠٧	10SS	Mostafa and Mahmoud
Manually Scheduled	Final graphically show connections of all devices	87 days	ث ٢٢/٠١/١٨	أ ٢٢/٠٥/١٨	10SS	Ramy and Abdelrahman
Auto Scheduled	**testing and applying**	**3 days**	**خ ٢٢/٠٥/١٩**	**إ ٢٢/٠٥/٢٣**		
Manually Scheduled	The tool will be implemented on the devices of the labs at the information technology department as a proof of concept .	1 day	خ ٢٢/٠٥/١٩	خ ٢٢/٠٥/١٩	12	all team
Manually Scheduled	The tool is implemented at the devices of Aussit university.	1 day	ج ٢٢/٠٥/٢٠	ج ٢٢/٠٥/٢٠	14	all team
Manually Schedule	A performance evaluation between our tool and	2 days	س ٢٢/٠٥/٢١	إ ٢٢/٠٥/٢٣	14	all team

d	Wireshark will be made					

3.0 What exactly is Node.js?

Node.js is a JavaScript runtime environment. Sounds great, but what does that mean? How does that work?

The Node.js run-time environment includes everything you need to execute a program written in JavaScript.



Node.js came into existence when the original developers of JavaScript extended it from something you could only run in the browser to something you could run on your machine as a standalone application.

Now you can do much more with JavaScript than just making websites interactive. Java script now has the capability to do things that other scripting languages like Python can do.

Both your browser JavaScript and Node.js run on the V8 JavaScript runtime engine. This engine takes your JavaScript code and converts

it into a faster machine code. Machine code is low-level code which the computer can run without needing to first interpret it.

Why Node.js?

Here's a formal definition as given on the official Node.js website:

Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine.

Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient.

Node.js' package ecosystem, npm, is the largest ecosystem of open source libraries in the world.

Fundamentals of Node.js

Now that we know what Node.js is, let's explore the fundamentals of this tool.

Console

The console is a module provided by Node.js that is akin to the JavaScript console in the browser when you inspect a webpage. The console has methods that are available for us to use for debugging purposes.

console.log(): *Frequently used to log some sort of output.*

console.warn(): *Explicitly delivers a warning to the console.*

console.error(): *Explicitly delivers an error message to the console.*

You can log an error as a string or as an object. If logged as a new Error(), a traceback will be included as part of the message.

console.trace(): *Logs a traceback when an error occurs in your code. Gives line number and column number of the file that the error probably occurred.*

Buffer:

At its core, the Buffer class in Node.js is a temporary storage solution for file systems. Due to its low-level nature, as web developers we will rarely actually use the Buffer class directly. The main purpose of this class is to allocate memory.

File System:

The file system (fs) module allows us to interact with files in Node.js. There are synchronous and asynchronous methods that can be used to read or write to a file using the fs module. In contrast to using console or the Buffer class, we need to import the fs module into the file that we would like to use in order to get it to work.

The code example below shows how the readFile, an asynchronous method, works. Notice that the last argument in the method is a callback function whose first argument is an error. By definition this callback will always pass in the error first before the data.

Event Loop:

Much of Node.js is built to be event-driven. When a user clicks on an interface or types in a form, an event is triggered to happen and then something occurs as a result. To attach a function or set of functions to a specific event is emitting an event.

These functions, called event listeners, are one part of an overall journey called the Event Loop.

Globals:

Global objects are available in every module, so they can be used without importing a specific module. The Buffer class, for example, class is defined as a global in Node.js. Some other common global objects are.

The console object is used to print to stdout and stderr.

Timers, such as setImmediate, setInterval, and setTimeout, are also globals.

The process object is also global.

In a browser, the top-level scope is the global scope. But in Node.js, the top-level scope is not the global scope.

(3.1) NPM - Node Package manager

npm is two things: first and foremost, it is an online repository for the publishing of open-source Node.js projects; second, it is a command-line utility for interacting with said repository that aids in package installation, version management, and dependency management. A plethora of Node.js libraries and applications are published on npm, and many more are added every day. These applications can be searched for on https://www.npmjs.com/. Once you have a package you want to install, it can be installed with a single command-line command.

Let's say you're hard at work one day, developing the Next Great Application. You come across a problem, and you decide that it's time to use that cool library you keep hearing about - let's use Caolan McMahon's async as an example. Thankfully, npm is very simple to use: you only have to run npm install async, and the specified module will be installed in the current directory under ./node_modules/. Once installed to your node_modules folder, you'll be able to use require()

on them just like they were built-ins.

Let's look at an example of a global install - let's say coffee-script. The npm command is simple: npm install coffee-script -g. This will typically install the program and put a symlink to it in /usr/local/bin/. This will then allow you to run the program from the console just like any other CLI tool. In this case, running coffee will now allow you to use the coffee-script REPL.

Another important use for npm is dependency management. When you have a node project with a package.json file, you can run npm install from the project root and npm will install all the dependencies listed in the package.json. This makes installing a Node.js project from a git repo much easier! For example, vows, a Node.js testing framework, can be installed from git, and its single dependency, eyes, can be automatically handled:

Example:

```
ð git clone https://github.com/cloudhead/vows.git
ð cd vows
ð npm install
```

After running those commands, you will see a node_modules folder containing all of the project dependencies specified in the package.json

(3.2) node-modules to Network monitor

Cap

Cap was my final year project for my BSc. It's a language that compiles to JavaScript, for use in the browser and node. It takes inspiration from JavaScript, Coffee Script, Jade, Stylus, Python and

ML. The reason I'm open sourcing it is just for show. I wouldn't recommend using it in production, in fact, I'd definitely recommend not using it. Feel free to try it out though, and I'm happy to help out if you get stuck with that.

Intro

Cap is a language built on top of Javascript. It aims to unify the underlying concepts of functional programming with an elegant and succint syntax.

Quickstart

The Cap compiler is implemented in node, get that first (pre-compiled binaries available for Win/Mac, Linux you have to build yourself). Node comes with the package manager npm which you can use to install the Cap compiler.

Net-SNMP : ”Simple Network Management Protocol” (SNMP)

Is an application–layer protocol defined by the Internet Architecture Board (IAB) in RFC1157 for exchanging management information between network devices. It is a part of Transmission Control Protocol/Internet Protocol (TCP/IP) protocol suite.

SNMP is one of the widely accepted network protocols to manage and monitor network elements. Most of the professional–grade network elements come with bundled SNMP agent. These agents have to be enabled and configured to communicate with the network monitoring tools or network management system (NMS).

SNMP consists of

- *SNMP Manager*
- *Managed devices*
- *SNMP agent*
- *Management Information Database Otherwise called as Management Information Base (MIB)*

SNMP Manager:

A manager or management system is a separate entity that is responsible to communicate with the SNMP agent implemented network devices. This is typically a computer that is used to run one or more network management systems.

SNMP Manager's key functions

- *Queries agents*
- *Gets responses from agents*
- *Sets variables in agents*
- *Acknowledges asynchronous events from agents*

Managed Devices:

A managed device or the network element is a part of the network that requires some form of monitoring and management e.g. routers, switches, servers, workstations, printers, UPSs, etc...

SNMP Agent:

The agent is a program that is packaged within the network element. Enabling the agent allows it to collect the management information database from the device locally and makes it available to the SNMP manager, when it is queried for. These agents could be standard (e.g.

Net-SNMP) or specific to a vendor (e.g. HP insight agent)

SNMP agent's key functions

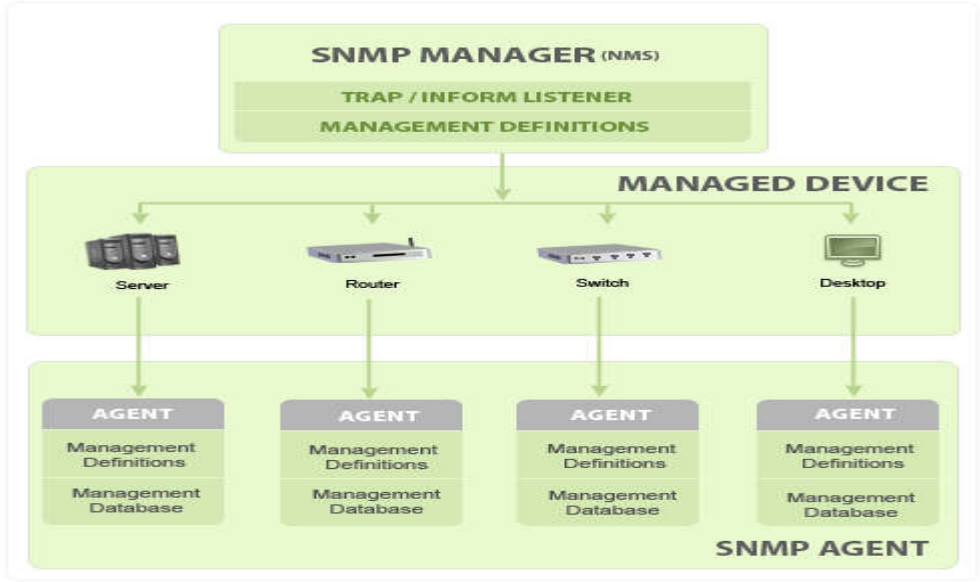
Collects management information about its local environment

Stores and retrieves management information as defined in the MIB.

Signals an event to the manager.

Acts as a proxy for some non–SNMP manageable network node.

Basic SNMP Communication Diagram



Management Information database or Management Information Base (MIB)

Every SNMP agent maintains an information database describing the managed device parameters. The SNMP manager uses this database to request the agent for specific information and further translates the information as needed for the Network Management System (NMS). This commonly shared database between the Agent and the Manager is called Management Information Base (MIB).

Typically these MIB contains standard set of statistical and control values defined for hardware nodes on a network. SNMP also allows

the extension of these standard values with values specific to a particular agent through the use of private MIBs.

In short, MIB files are the set of questions that a SNMP Manager can ask the agent. Agent collects these data locally and stores it, as defined in the MIB. So, the SNMP Manager should be aware of these standard and private questions for every type of agent.

SNMP Object Identifier:

OID (Object Identifier) is the name for a numeric identifier that is unique for every value in SNMP. This identifier consists of a numbers separated by points. The shape of each OID is determined by the identifier value of the parent element and then this value is complemented by a point and current number

OID	Name	Description
1.3.6.1.2.1.31.1.1.1.3	ifInBroadcastPkts	The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to a broadcast address at this s…
1.3.6.1.2.1.31.1.1.1.5	ifOutBroadcastPkts	The total number of packets that higher-level protocols requested be transmitted, and which were addressed to a broadcast address…
1.3.6.1.2.1.31.1.1.1.2	ifInMulticastPkts	The number of packets, delivered by this sub-layer to a higher (sub-)layer, which were addressed to a multicast address at this s…
1.3.6.1.2.1.31.1.1.1.4	ifOutMulticastPkts	The total number of packets that higher-level protocols requested be transmitted, and which were addressed to a multicast address

second net-snmp module:

```
var snmp = require("net-snmp");
var oids = ["1.3.6.1.2.1.31.1.1.1.3.1","1.3.6.1.2.1.31.1.1.1.5.1","1.3.6.1.2.1.31.1.1.
var discriptions=["In Broadcast packet counter: ","Out Broadcast packet counter: ","In

function theOutput(res){
    var session = snmp.createSession("127.0.0.1", "CCNP");
    var output= [];
    session.get(oids, function (error, varbinds) {
        if (error) {
            console.error(error);
        } else {
            for (var i = 0; i < varbinds.length; i++) {
                if (snmp.isVarbindError(varbinds[i])) {
                    console.error(snmp.varbindError(varbinds[i]));
                } else {
                    output.push(discriptions[i]+varbinds[i].oid + " = " + varbinds[i].
                }
            }
        }
        session.close();
        return res.json(output);
    });
```

The output of this code

the features of net-snmp:-

- *Support for all SNMP versions: SNMPv1, SNMPv2c and SNMPv3*
- *SNMPv3 message authentication using MD5 or SHA, and privacy using DES or AES encryption*
- *Community-based and user-based authorization*
- *SNMP initiator for all relevant protocol operations:*

- *Get, GetNext, GetBulk, Set, Trap, Inform*
- *Convenience methods for MIB "walking", subtree collection, table and table column collection*
- *SNMPv3 context support*
- *Notification receiver for traps and informs*
- *MIB parsing and MIB module store*
- *SNMP agent with MIB management for both scalar and tabular data*
- *Agent table index support for non-integer keys, foreign keys, composite keys and table augmentation*
- *Agent support for "Row Status" protocol-based creation and deletion of table rows*
- *Agent support for these MIB constraints: MAX-ACCESS, integer ranges, string sizes, integer enumerations*
- *SNMP proxy forwarder for agent*
- *AgentX subagent*
- *IPv4 and IPv6*

4.0 Classification

Node files descriptions:

1- Network All Active Connected Devices node file:

Used to show only active connected devices to my network in some detail IP Address, MAC address.

2- List Interfaces node file:

Used to display all interfaces and components in my device(server) and their details.

3- Devices & Hosts Status node file:

It used to check for a particular device (IP address) or host (ex www.google.com) if it active or not.

4- Ipconfig node file:

This used to display my device local IP address and MAC address and Gitway.

5- Trace Packets node file:

Used to trace TCP or UDP packet inert to my device in details number of packets source IP address and port number and destination IP address and port number and for some of them some details.

6- System Information node file:

Display information about my system such that (manufacturer, brand, vendor, family, speed, cores, ……)

7- Upload and Download Host Speed node file:

Display the network upload and download for particular host.

8- Draw graphically:

If you want to make primary design or draw something or save a note.

9- SNMP:

Show network information by their OIDS using simple network management protocol.

10- Speed Tester:

Measure server graphically Download and Upload speed.

11- Error Detect:

Detect some network problems using SNMP.

12- Backup:

Show latest used data.

13- Performance:

Show graphically system performance.

14- Track IP Location:

Information about my IP address

15- Broadcast:

Count the number of different broadcast packets using SNMP.

16- Network datagram:

Tool to design your network.

17- Get real IP:

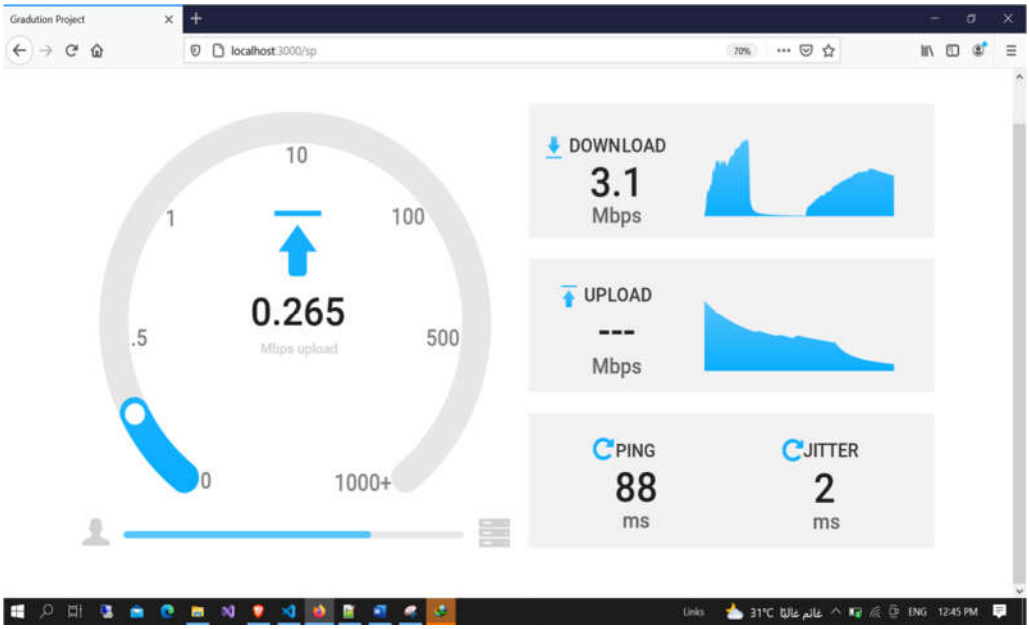
Git your real IP not one taken by DNS.

5.0 Tools identification

- *Visual studio code: A standalone source code editor that runs on Windows and Linux. The top pick for JavaScript and web developers, with tons of extensions to support just about any programming language.*
- *Node.js®: is a JavaScript runtime built on Chrome's V8 JavaScript engine.*
- *Adobe XD: is a vector-based user experience design tool for web apps and mobile apps, developed and published by Adobe Inc.*
- *CSS2: for describing the presentation of a document written in a markup language such as HTML.*
- *HTML5: for structuring and presenting content on the World Wide Web. It is the fifth and last major HTML version that is a World Wide Web Consortium (W3C).*
- *GNS3: GNS3 is open source, free software that you can download from http://gns3.com, used to emulate, configure, test and troubleshoot virtual and real networks. GNS3 allows you to run a small topology consisting of only a few devices on your laptop, to those that have many devices hosted on multiple servers or even hosted in the cloud.*
- *Microsoft Project: It is designed to assist a project manager in developing a schedule, assigning resources to tasks, tracking progress, managing the budget, and analyzing workloads.*

5.1 Graphics

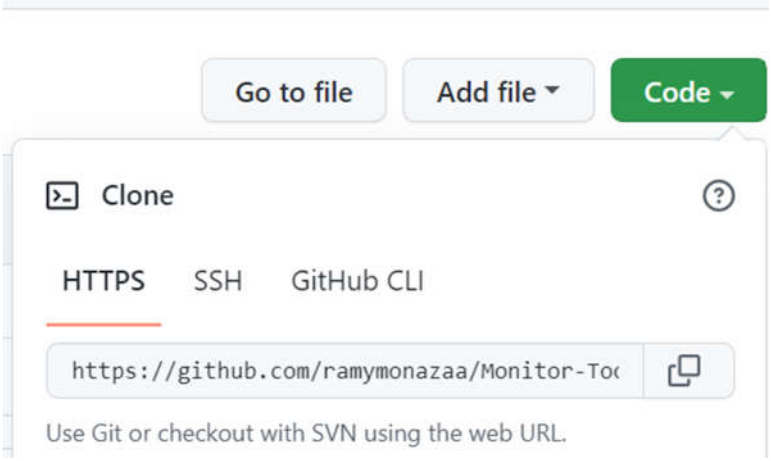
“CPU Performance”



“Download speed”

5.2 How to run tool

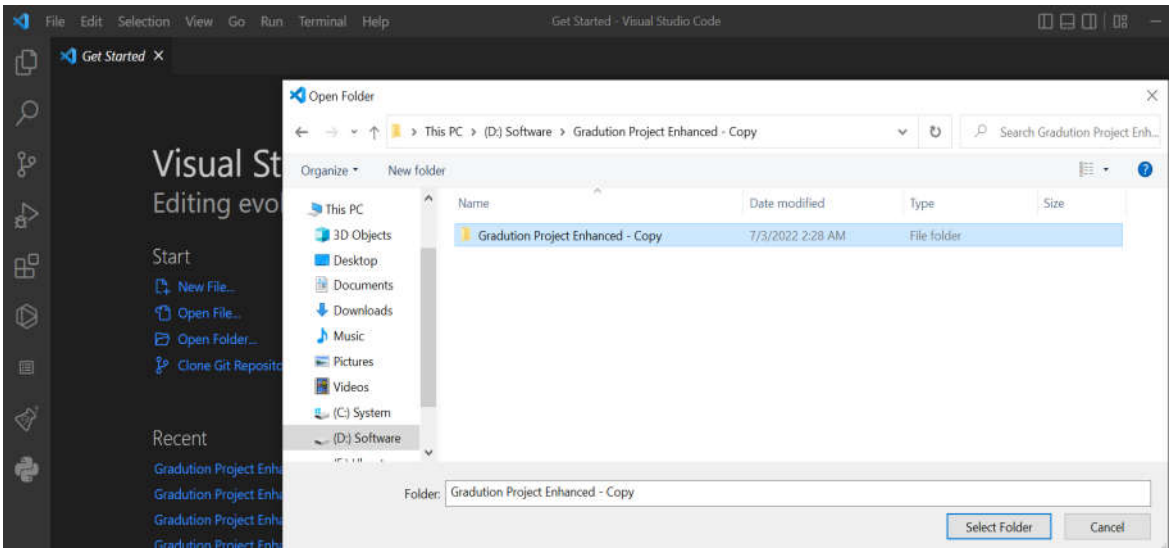
1- You have to download the project from github "https://github.com/ramymonazaa/Monitor-Tool/tree/main”



By Git clone this URL from your terminal.

2- *Open your Visual Studio Code app (if you haven't download it)*

3- *From file select "Open Folder" and then the choose the project folder.*



4- *In file network_active_connected_devices change the ip of getLocalDeviceList in line 22 to your ip*

5- *You can git your IP by open cmd write ipconfig*

6- *In file TracePackets also change Cap.findDevice in line 22 to your IP but replace the last part with 1, like that 192.168.43.93 ➔192.168.43.1*

7- *Download node js node-v16.13.0-x64 or any higher versions https://nodejs.org/en/download/*

8- *Install it.*

9- *In VS code from terminal choose new terminal.*

10- *Use command node --inspect start.js*

11- *Open Firefox browser and take the link copy past in it.*

6.0 References:

[1] http://www.java2s.com/Open-Source/Javascript_Free_Code/Node.JS/Download_node_pcap_parser_Free_Java_Code.htm.

[2] https://www.npmjs.com/package/net-snmp#using-this-module-command--notification-generator.
[3] https://nodejs.org/en.
[4] -https://stackoverflow.com/questions/53362082/how-to-execute-adb-backup-command-in-jscan-be-node-js-using-adbkit?fbclid=IwAR1SXqLVpWReC1KLrVkYxiXn-asXC9KAMt3SLGnN2VZtj7QL2y349OTFUGM.

7.0 Conclusion statement

Your network is your business. Network monitoring helps you stay in business. The more you know about your network, the more you can reassure internal and external users that your network will be able to achieve performance, availability, and access goals for users to access services safely and effectively.

Finally, we would like to thank **Dr. *Islam Taj-Eddin* and *Eng Mustafa Farouk*** to assist us in this project.