Virtual Machine Networking and Microservice Deployment using VirtualBox

Name: Megha Pandey

Roll. No: G25AI1028

Course: Virtualization and Cloud Computing

Professor: Sumit Kalra

Tools Used: Oracle VirtualBox, Ubuntu Linux, Node.js, Express.js

---

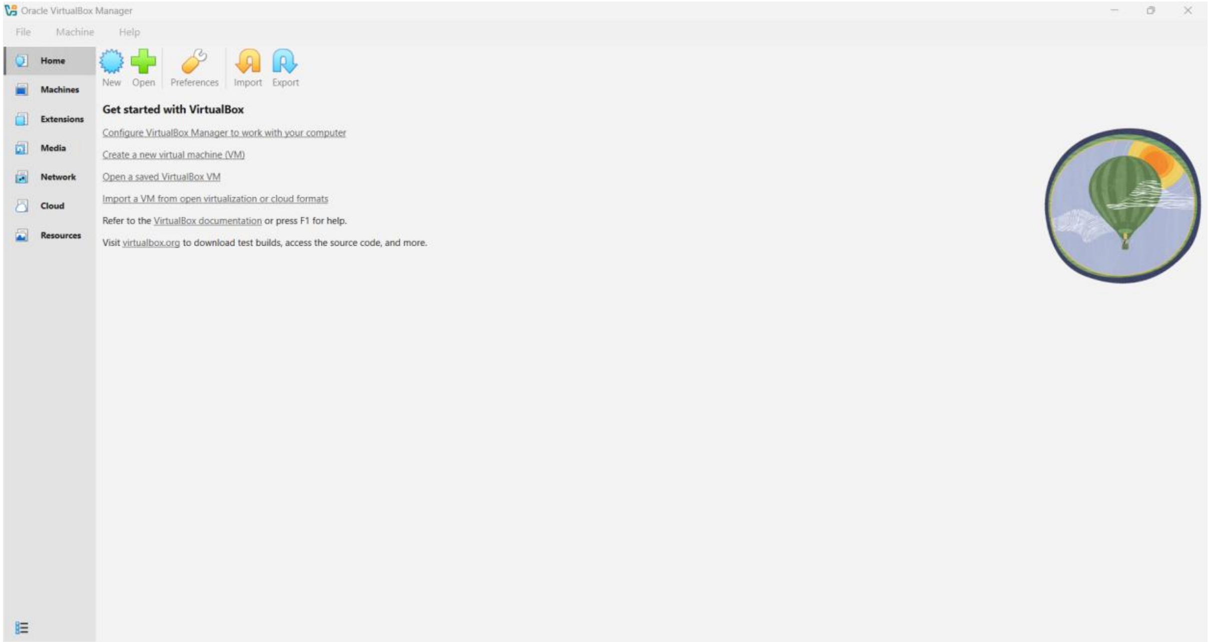
1. Objective

The objective of this assignment is to create and configure multiple Virtual Machines (VMs) using Oracle VirtualBox, establish a private network between them, and deploy a simple microservice-based application. One VM hosts a Node.js REST API, while the second VM acts as a client that communicates with the API over the configured network.

2. Environment Setup

2.1 VirtualBox Installation

Oracle VirtualBox is installed on the host machine to create and manage virtual machines. Ubuntu (64-bit) ISO was used as the guest operating system.



Get started with VirtualBox

Configure VirtualBox Manager to work with your computer

Create a new virtual machine (VM)

Open a saved VirtualBox VM

Import a VM from open virtualization or cloud formats

Refer to the VirtualBox documentation or press F1 for help.

Visit virtualbox.org to download test builds, access the source code, and more.

3. Virtual Machine Configuration

Two Ubuntu-based virtual machines are created.

3.1 VM 1 – API Server

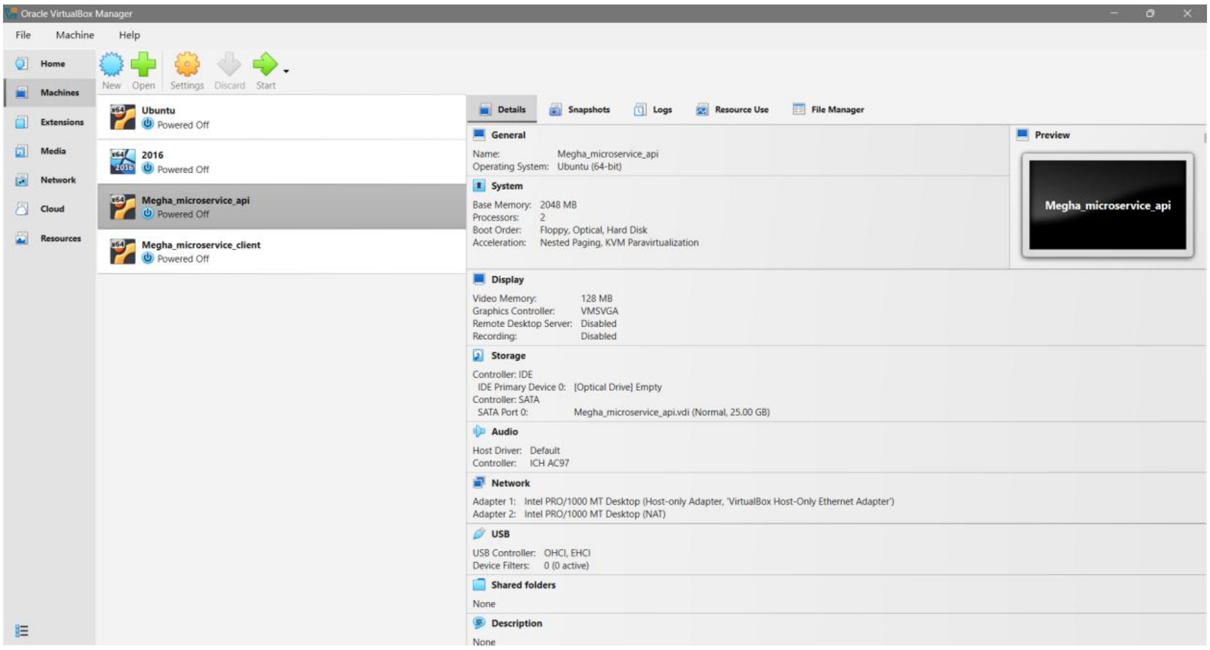
VM Name: Megha_microservice_api
Role: Hosts the microservice (Node.js REST API)

System Configuration

- Operating System: Ubuntu (64-bit)
- Base Memory: 2048 MB
- CPUs: 2
- Storage: Virtual Hard Disk (VDI)

Network Configuration

- Adapter 1: **Host-only Adapter**
- Network Name: VirtualBox Host-Only Ethernet Adapter
- Adapter Type: Intel PRO/1000 MT Desktop
- IP Address (assigned via DHCP): 192.168.56.101
- Purpose: Enables private communication with the client VM



```
General
Name:              Megha_microservice_api
Operating System:  Ubuntu (64-bit)

System
Base Memory:   2048 MB
Processors:    2
Boot Order:    Floppy, Optical, Hard Disk
Acceleration:  Nested Paging, KVM Paravirtualization

Display
Video Memory:            128 MB
Graphics Controller:     VMSVGA
Remote Desktop Server:   Disabled
Recording:               Disabled

Storage
Controller: IDE
 IDE Primary Device 0:  [Optical Drive] Empty
Controller: SATA
 SATA Port 0:           Megha_microservice_api.vdi (Normal, 25.00 GB)

Audio
Host Driver:  Default
Controller:   ICH AC97

Network
Adapter 1:  Intel PRO/1000 MT Desktop (Host-only Adapter, 'VirtualBox Host-Only Ethernet Adapter')
Adapter 2:  Intel PRO/1000 MT Desktop (NAT)

USB
USB Controller:  OHCI, EHCI
Device Filters:  0 (0 active)

Shared folders
None

Description
None
```

3.2 VM 2 – Client Machine

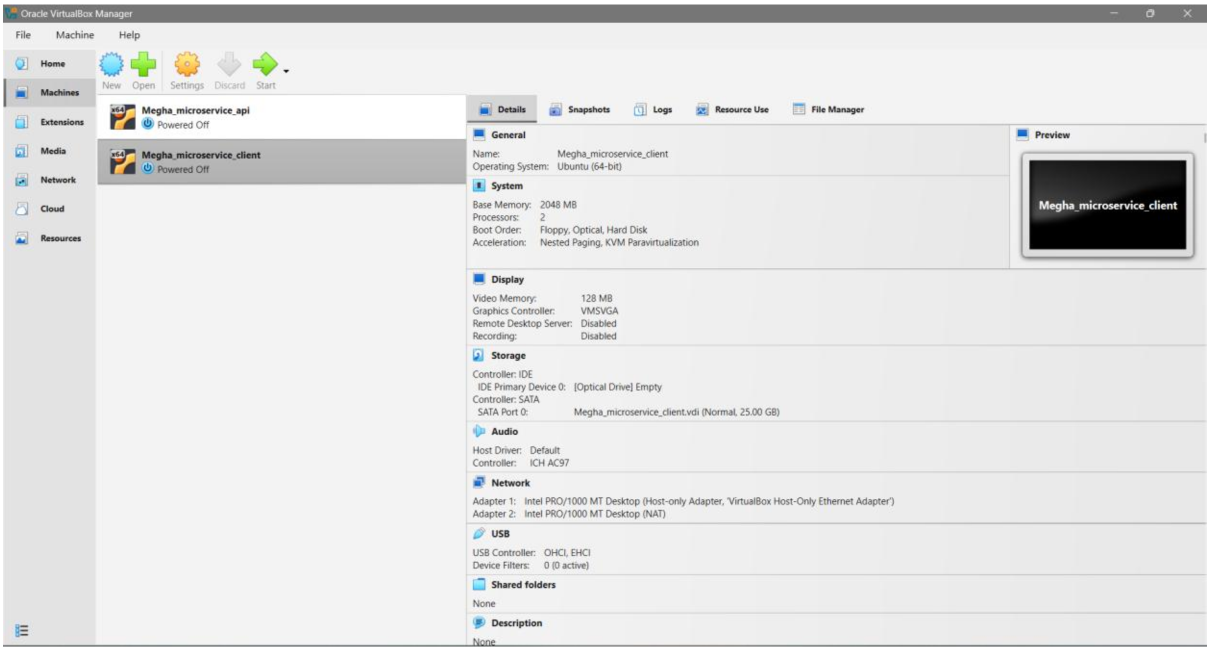
VM Name: Megha_microservice_client
Role: Acts as the consumer of the microservice

System Configuration

- Operating System: Ubuntu (64-bit)
- Base Memory: 2048 MB
- CPUs: 2
- Storage: Virtual Hard Disk (VDI)

Network Configuration

- Adapter 1: **Host-only Adapter**
- Network Name: VirtualBox Host-Only Ethernet Adapter
- Adapter Type: Intel PRO/1000 MT Desktop
- IP Address (assigned via DHCP): 192.168.56.102



Megha_microservice_api
Powered Off

Megha_microservice_client
Powered Off

General

Name: Megha_microservice_client
Operating System: Ubuntu (64-bit)

System

Base Memory: 2048 MB
Processors: 2
Boot Order: Floppy, Optical, Hard Disk
Acceleration: Nested Paging, KVM Paravirtualization

Display

Video Memory: 128 MB
Graphics Controller: VMSVGA
Remote Desktop Server: Disabled
Recording: Disabled

Storage

Controller: IDE
IDE Primary Device 0: [Optical Drive] Empty
Controller: SATA
SATA Port 0: Megha_microservice_client.vdi (Normal, 25.00 GB)

Audio

Host Driver: Default
Controller: ICH AC97

Network

Adapter 1: Intel PRO/1000 MT Desktop (Host-only Adapter, 'VirtualBox Host-Only Ethernet Adapter')
Adapter 2: Intel PRO/1000 MT Desktop (NAT)

USB

USB Controller: OHCI, EHCI
Device Filters: 0 (0 active)

Shared folders

None

Description

None

4. Network Connectivity Verification

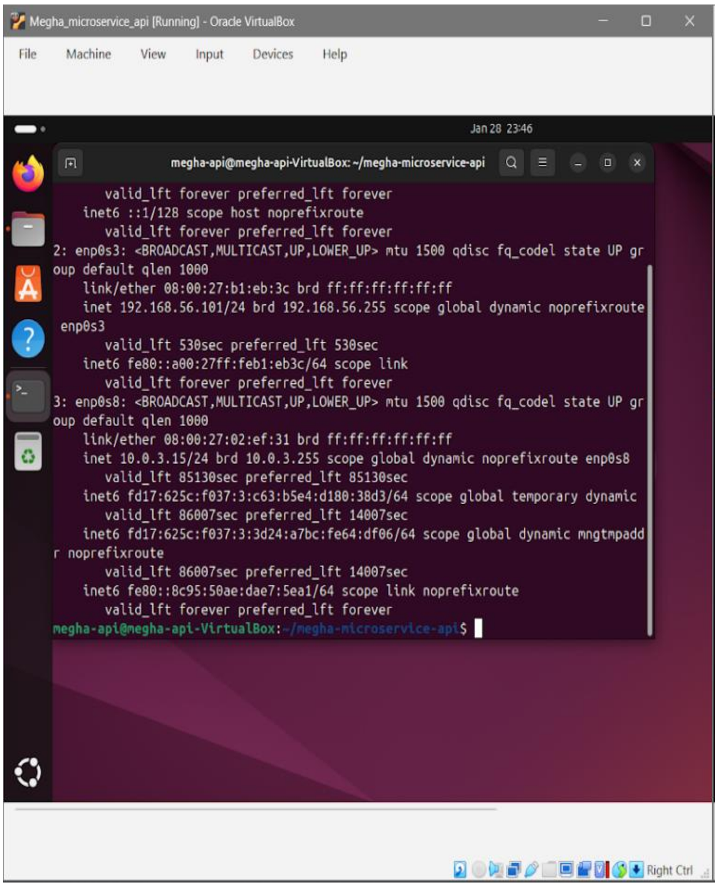
To verify connectivity between the two VMs, ICMP ping tests were performed.

From Client VM to API VM

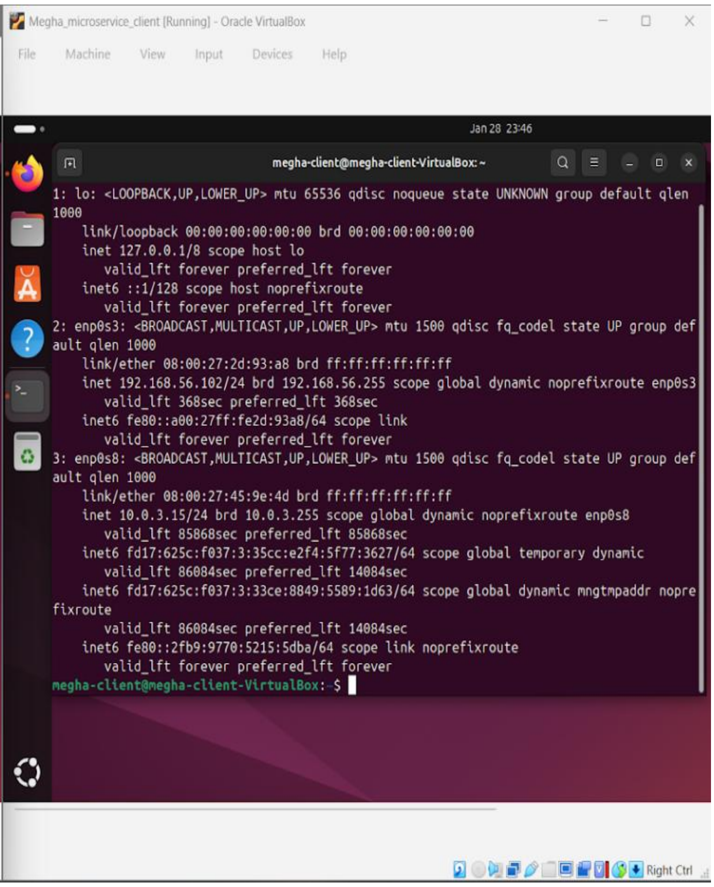
ping 192.168.56.101

From API VM to Client VM

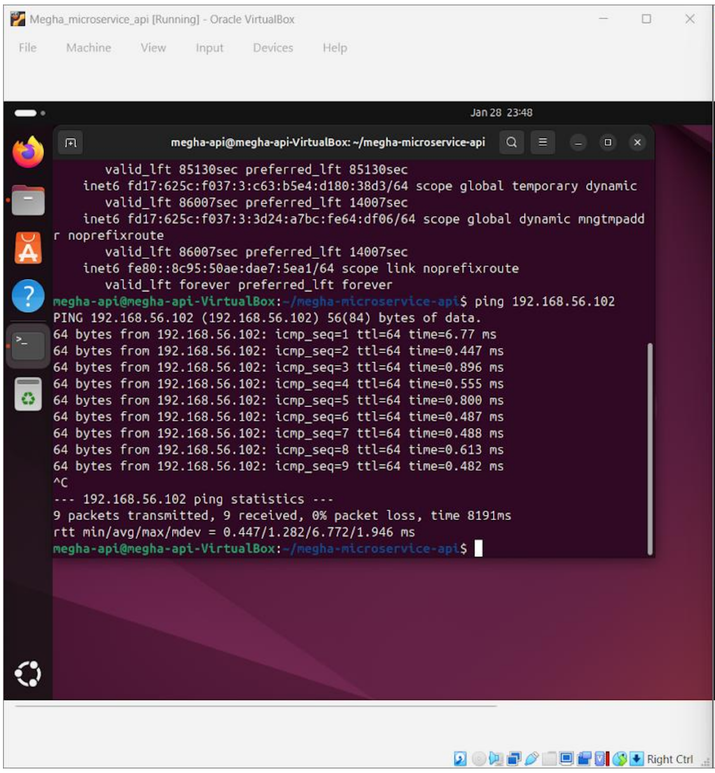
ping 192.168.56.102


Megha_microservice_api [Running] - Oracle VirtualBox
File
Machine
View
Input
Devices
Help
Jan 28 23:46
megha-api@megha-api-VirtualBox: ~/megha-microservice-api
Right Ctrl

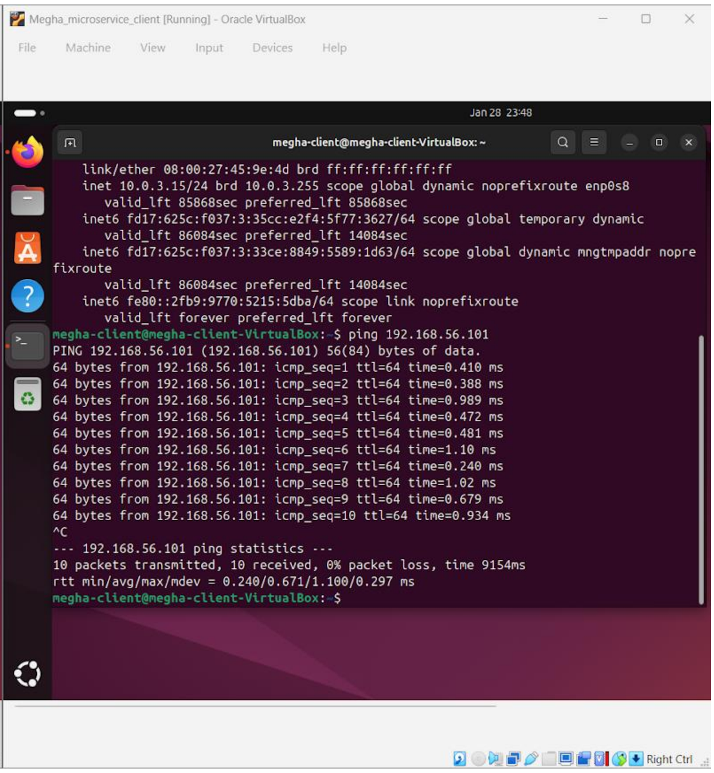
```
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
    link/ether 08:00:27:b1:eb:3c brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.101/24 brd 192.168.56.255 scope global dynamic noprefixroute
 enp0s3
       valid_lft 530sec preferred_lft 530sec
    inet6 fe80::a00:27ff:feb1:eb3c/64 scope link
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP gr
oup default qlen 1000
    link/ether 08:00:27:02:ef:31 brd ff:ff:ff:ff:ff:ff
    inet 10.0.3.15/24 brd 10.0.3.255 scope global dynamic noprefixroute enp0s8
       valid_lft 85130sec preferred_lft 85130sec
    inet6 fd17:625c:f037:3:c63:b5e4:d180:38d3/64 scope global temporary dynamic
       valid_lft 86007sec preferred_lft 14007sec
    inet6 fd17:625c:f037:3:3d24:a7bc:fe64:df06/64 scope global dynamic mngtmpadd
r noprefixroute
       valid_lft 86007sec preferred_lft 14007sec
    inet6 fe80::8c95:50ae:dae7:5ea1/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
megha-api@megha-api-VirtualBox:~/megha-microservice-api$
```


Megha_microservice_client [Running] - Oracle VirtualBox
File
Machine
View
Input
Devices
Help
Jan 28 23:46
megha-client@megha-client-VirtualBox: ~
Right Ctrl

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen
1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group def
ault qlen 1000
    link/ether 08:00:27:2d:93:a8 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.102/24 brd 192.168.56.255 scope global dynamic noprefixroute enp0s3
       valid_lft 368sec preferred_lft 368sec
    inet6 fe80::a00:27ff:fe2d:93a8/64 scope link
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group def
ault qlen 1000
    link/ether 08:00:27:45:9e:4d brd ff:ff:ff:ff:ff:ff
    inet 10.0.3.15/24 brd 10.0.3.255 scope global dynamic noprefixroute enp0s8
       valid_lft 85868sec preferred_lft 85868sec
    inet6 fd17:625c:f037:3:35cc:e2f4:5f77:3627/64 scope global temporary dynamic
       valid_lft 86084sec preferred_lft 14084sec
    inet6 fd17:625c:f037:3:33ce:8849:5589:1d63/64 scope global dynamic mngtmpaddr nopre
fixroute
       valid_lft 86084sec preferred_lft 14084sec
    inet6 fe80::2fb9:9770:5215:5dba/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
megha-client@megha-client-VirtualBox:~$
```


Megha_microservice_api [Running] - Oracle VirtualBox
File
Machine
View
Input
Devices
Help
Jan 28 23:48
megha-api@megha-api-VirtualBox: ~/megha-microservice-api
Right Ctrl

```
       valid_lft 85130sec preferred_lft 85130sec
    inet6 fd17:625c:f037:3:c63:b5e4:d180:38d3/64 scope global temporary dynamic
       valid_lft 86007sec preferred_lft 14007sec
    inet6 fd17:625c:f037:3:3d24:a7bc:fe64:df06/64 scope global dynamic mngtmpadd
r noprefixroute
       valid_lft 86007sec preferred_lft 14007sec
    inet6 fe80::8c95:50ae:dae7:5ea1/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
megha-api@megha-api-VirtualBox:~/megha-microservice-api$ ping 192.168.56.102
PING 192.168.56.102 (192.168.56.102) 56(84) bytes of data.
64 bytes from 192.168.56.102: icmp_seq=1 ttl=64 time=6.77 ms
64 bytes from 192.168.56.102: icmp_seq=2 ttl=64 time=0.447 ms
64 bytes from 192.168.56.102: icmp_seq=3 ttl=64 time=0.896 ms
64 bytes from 192.168.56.102: icmp_seq=4 ttl=64 time=0.555 ms
64 bytes from 192.168.56.102: icmp_seq=5 ttl=64 time=0.800 ms
64 bytes from 192.168.56.102: icmp_seq=6 ttl=64 time=0.487 ms
64 bytes from 192.168.56.102: icmp_seq=7 ttl=64 time=0.488 ms
64 bytes from 192.168.56.102: icmp_seq=8 ttl=64 time=0.613 ms
64 bytes from 192.168.56.102: icmp_seq=9 ttl=64 time=0.482 ms
^C
--- 192.168.56.102 ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8191ms
rtt min/avg/max/mdev = 0.447/1.282/6.772/1.946 ms
megha-api@megha-api-VirtualBox:~/megha-microservice-api$
```


Megha_microservice_client [Running] - Oracle VirtualBox
File
Machine
View
Input
Devices
Help
Jan 28 23:48
megha-client@megha-client-VirtualBox: ~
Right Ctrl

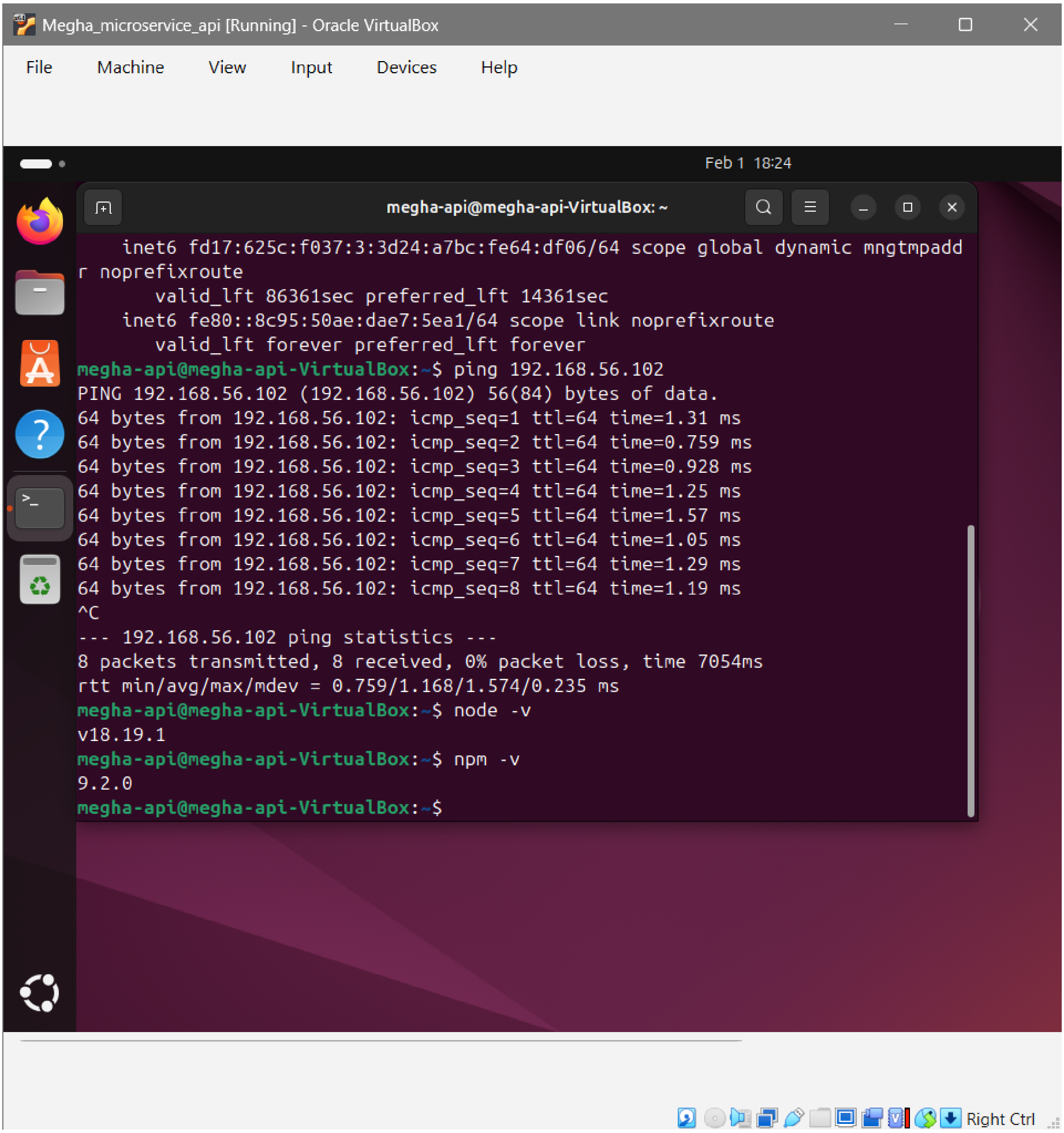
```
    link/ether 08:00:27:45:9e:4d brd ff:ff:ff:ff:ff:ff
    inet 10.0.3.15/24 brd 10.0.3.255 scope global dynamic noprefixroute enp0s8
       valid_lft 85868sec preferred_lft 85868sec
    inet6 fd17:625c:f037:3:35cc:e2f4:5f77:3627/64 scope global temporary dynamic
       valid_lft 86084sec preferred_lft 14084sec
    inet6 fd17:625c:f037:3:33ce:8849:5589:1d63/64 scope global dynamic mngtmpaddr nopre
fixroute
       valid_lft 86084sec preferred_lft 14084sec
    inet6 fe80::2fb9:9770:5215:5dba/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
megha-client@megha-client-VirtualBox:~$ ping 192.168.56.101
PING 192.168.56.101 (192.168.56.101) 56(84) bytes of data.
64 bytes from 192.168.56.101: icmp_seq=1 ttl=64 time=0.410 ms
64 bytes from 192.168.56.101: icmp_seq=2 ttl=64 time=0.388 ms
64 bytes from 192.168.56.101: icmp_seq=3 ttl=64 time=0.989 ms
64 bytes from 192.168.56.101: icmp_seq=4 ttl=64 time=0.472 ms
64 bytes from 192.168.56.101: icmp_seq=5 ttl=64 time=0.481 ms
64 bytes from 192.168.56.101: icmp_seq=6 ttl=64 time=1.10 ms
64 bytes from 192.168.56.101: icmp_seq=7 ttl=64 time=0.240 ms
64 bytes from 192.168.56.101: icmp_seq=8 ttl=64 time=1.02 ms
64 bytes from 192.168.56.101: icmp_seq=9 ttl=64 time=0.679 ms
64 bytes from 192.168.56.101: icmp_seq=10 ttl=64 time=0.934 ms
^C
--- 192.168.56.101 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9154ms
rtt min/avg/max/mdev = 0.240/0.671/1.100/0.297 ms
megha-client@megha-client-VirtualBox:~$
```

Both VMs successfully communicated with each other with **0% packet loss**, confirming correct network configuration.

5. Microservice Deployment

5.1 Technology Stack

- Node.js (v18.19.1)
- npm (9.2.0)
- Express.js

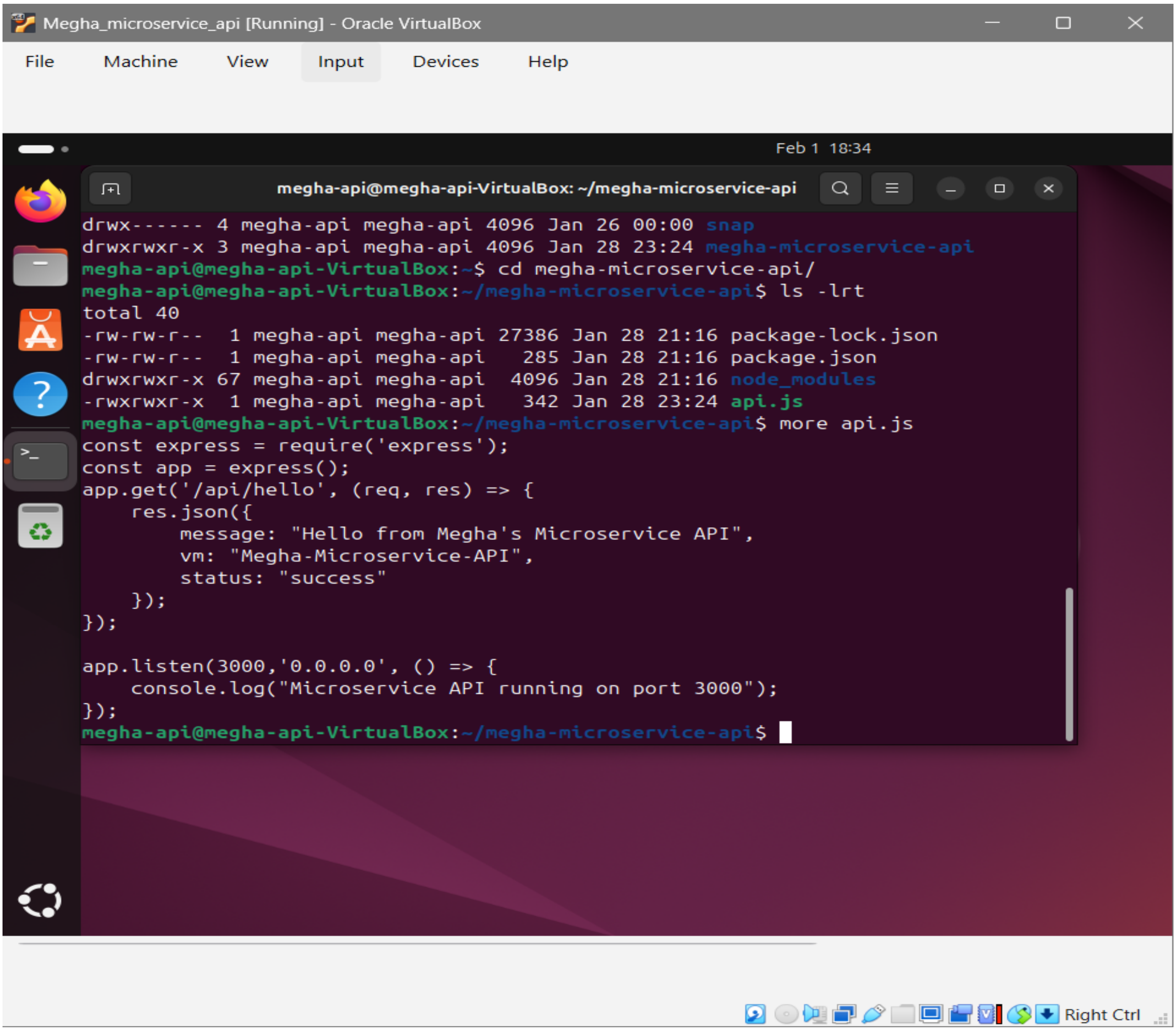


Megha_microservice_api [Running] - Oracle VirtualBox
File
Machine
View
Input
Devices
Help
Feb 1 18:24
megha-api@megha-api-VirtualBox: ~
Right Ctrl

```
    inet6 fd17:625c:f037:3:3d24:a7bc:fe64:df06/64 scope global dynamic mngtmpadd
r noprefixroute
       valid_lft 86361sec preferred_lft 14361sec
    inet6 fe80::8c95:50ae:dae7:5ea1/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
megha-api@megha-api-VirtualBox:~$ ping 192.168.56.102
PING 192.168.56.102 (192.168.56.102) 56(84) bytes of data.
64 bytes from 192.168.56.102: icmp_seq=1 ttl=64 time=1.31 ms
64 bytes from 192.168.56.102: icmp_seq=2 ttl=64 time=0.759 ms
64 bytes from 192.168.56.102: icmp_seq=3 ttl=64 time=0.928 ms
64 bytes from 192.168.56.102: icmp_seq=4 ttl=64 time=1.25 ms
64 bytes from 192.168.56.102: icmp_seq=5 ttl=64 time=1.57 ms
64 bytes from 192.168.56.102: icmp_seq=6 ttl=64 time=1.05 ms
64 bytes from 192.168.56.102: icmp_seq=7 ttl=64 time=1.29 ms
64 bytes from 192.168.56.102: icmp_seq=8 ttl=64 time=1.19 ms
^C
--- 192.168.56.102 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 7054ms
rtt min/avg/max/mdev = 0.759/1.168/1.574/0.235 ms
megha-api@megha-api-VirtualBox:~$ node -v
v18.19.1
megha-api@megha-api-VirtualBox:~$ npm -v
9.2.0
megha-api@megha-api-VirtualBox:~$
```

5.2 Microservice Source Code

```
const express = require('express');

const app = express();

app.get('/api/hello', (req, res) => {

  res.json({

    message: "Hello from Megha's Microservice API",

    vm: "Megha_microservice_api",

    status: "success"

  });

});

app.listen(3000, '0.0.0.0', () => {

  console.log("Microservice API running on port 3000");

});
```

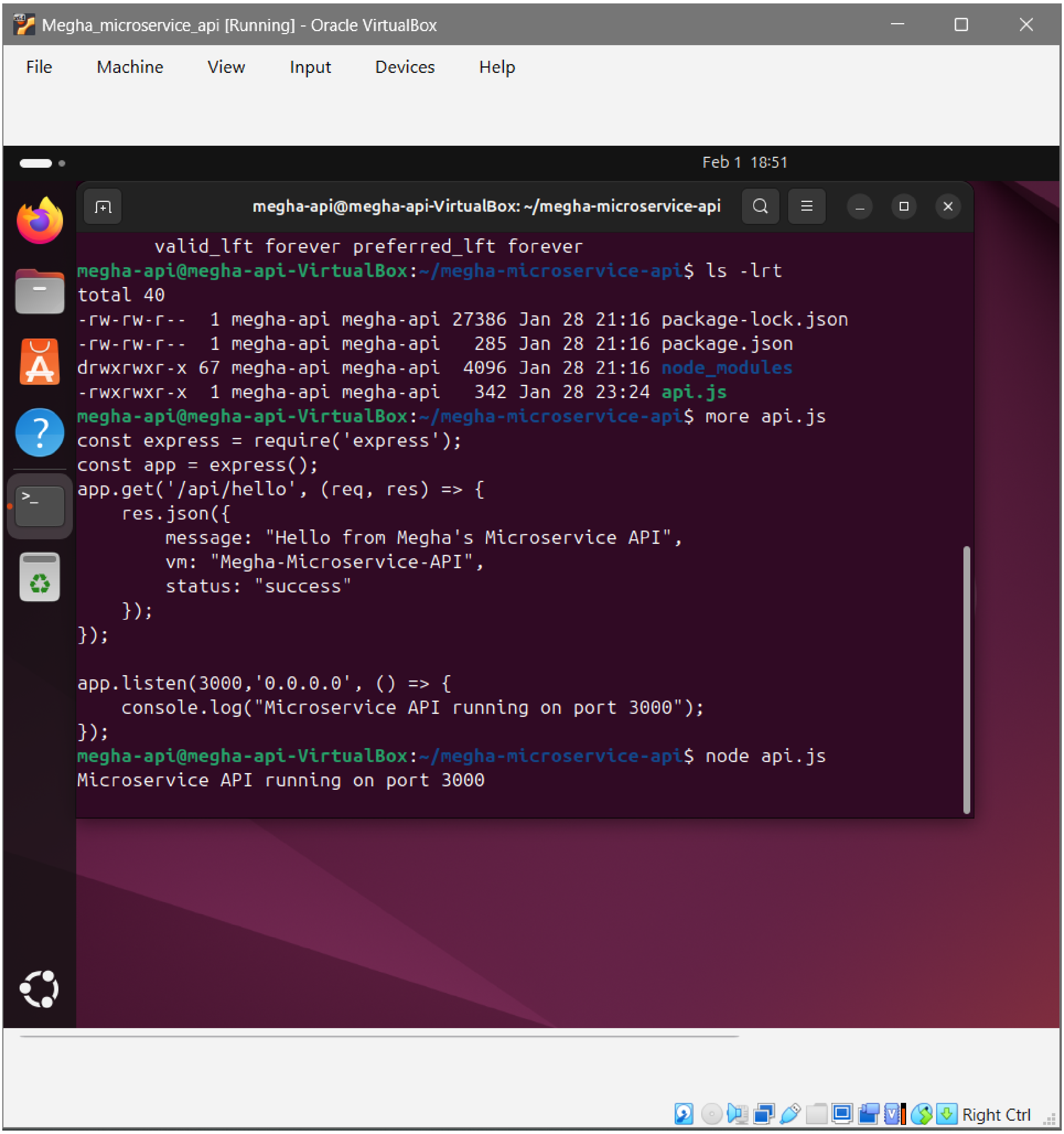

Megha_microservice_api [Running] - Oracle VirtualBox
File
Machine
View
Input
Devices
Help
Feb 1 18:34
megha-api@megha-api-VirtualBox: ~/megha-microservice-api
Right Ctrl

```
drwx------ 4 megha-api megha-api 4096 Jan 26 00:00 snap
drwxrwxr-x 3 megha-api megha-api 4096 Jan 28 23:24 megha-microservice-api
megha-api@megha-api-VirtualBox:~$ cd megha-microservice-api/
megha-api@megha-api-VirtualBox:~/megha-microservice-api$ ls -lrt
total 40
-rw-rw-r--  1 megha-api megha-api 27386 Jan 28 21:16 package-lock.json
-rw-rw-r--  1 megha-api megha-api   285 Jan 28 21:16 package.json
drwxrwxr-x 67 megha-api megha-api  4096 Jan 28 21:16 node_modules
-rwxrwxr-x  1 megha-api megha-api   342 Jan 28 23:24 api.js
megha-api@megha-api-VirtualBox:~/megha-microservice-api$ more api.js
const express = require('express');
const app = express();
app.get('/api/hello', (req, res) => {
    res.json({
        message: "Hello from Megha's Microservice API",
        vm: "Megha-Microservice-API",
        status: "success"
    });
});

app.listen(3000,'0.0.0.0', () => {
    console.log("Microservice API running on port 3000");
});
megha-api@megha-api-VirtualBox:~/megha-microservice-api$
```

5.3 Running the Microservice

On the API VM:

node api.js



Megha_microservice_api [Running] - Oracle VirtualBox
File
Machine
View
Input
Devices
Help
Feb 1 18:51
megha-api@megha-api-VirtualBox: ~/megha-microservice-api
Right Ctrl

```
       valid_lft forever preferred_lft forever
megha-api@megha-api-VirtualBox:~/megha-microservice-api$ ls -lrt
total 40
-rw-rw-r--  1 megha-api megha-api 27386 Jan 28 21:16 package-lock.json
-rw-rw-r--  1 megha-api megha-api   285 Jan 28 21:16 package.json
drwxrwxr-x 67 megha-api megha-api  4096 Jan 28 21:16 node_modules
-rwxrwxr-x  1 megha-api megha-api   342 Jan 28 23:24 api.js
megha-api@megha-api-VirtualBox:~/megha-microservice-api$ more api.js
const express = require('express');
const app = express();
app.get('/api/hello', (req, res) => {
    res.json({
        message: "Hello from Megha's Microservice API",
        vm: "Megha-Microservice-API",
        status: "success"
    });
});

app.listen(3000,'0.0.0.0', () => {
    console.log("Microservice API running on port 3000");
});
megha-api@megha-api-VirtualBox:~/megha-microservice-api$ node api.js
Microservice API running on port 3000
```

6. Accessing the Microservice from Client VM

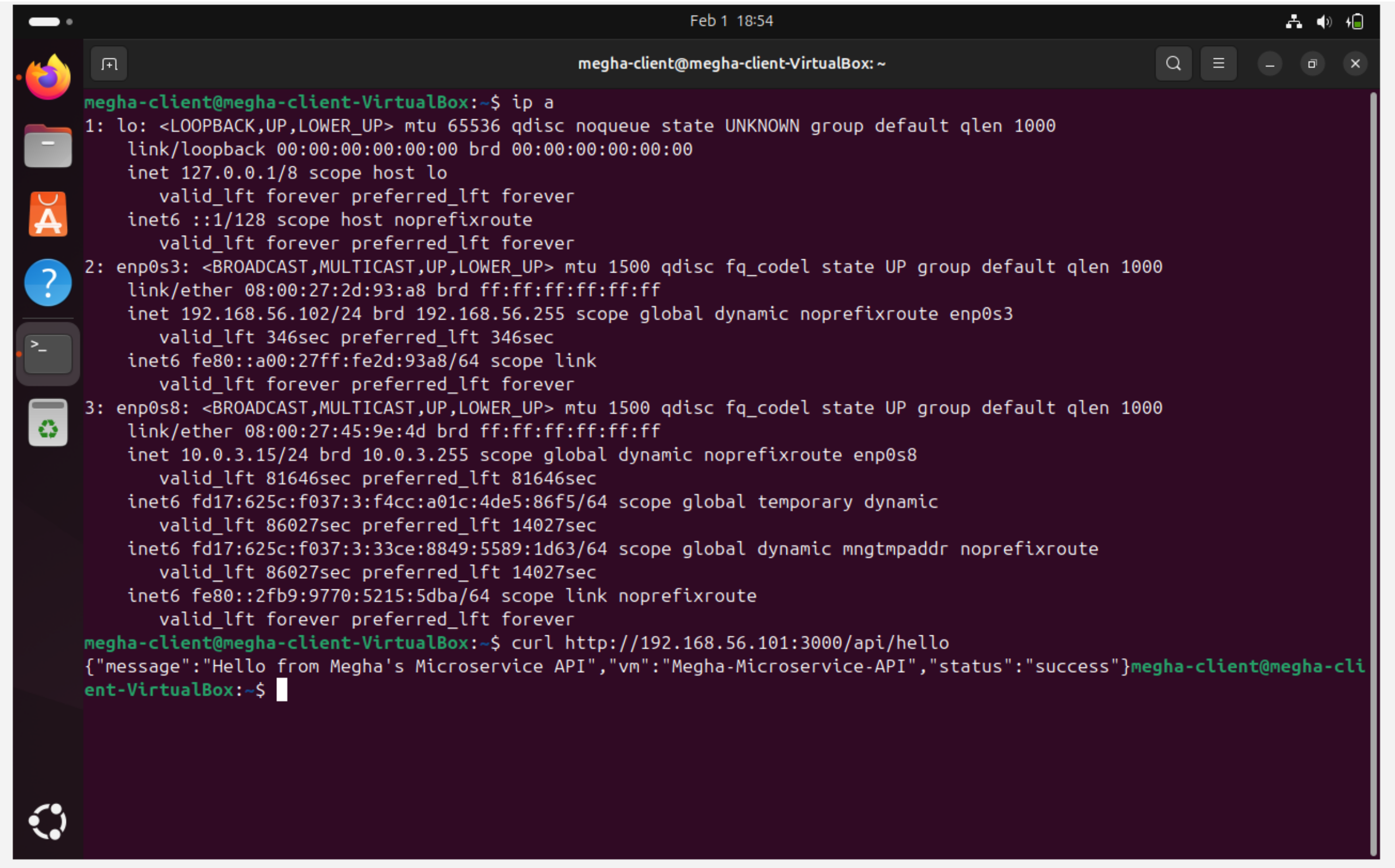
Since both VMs are on the same Host-only network, the client accesses the API using the API VM’s IP address.

Request from Client VM

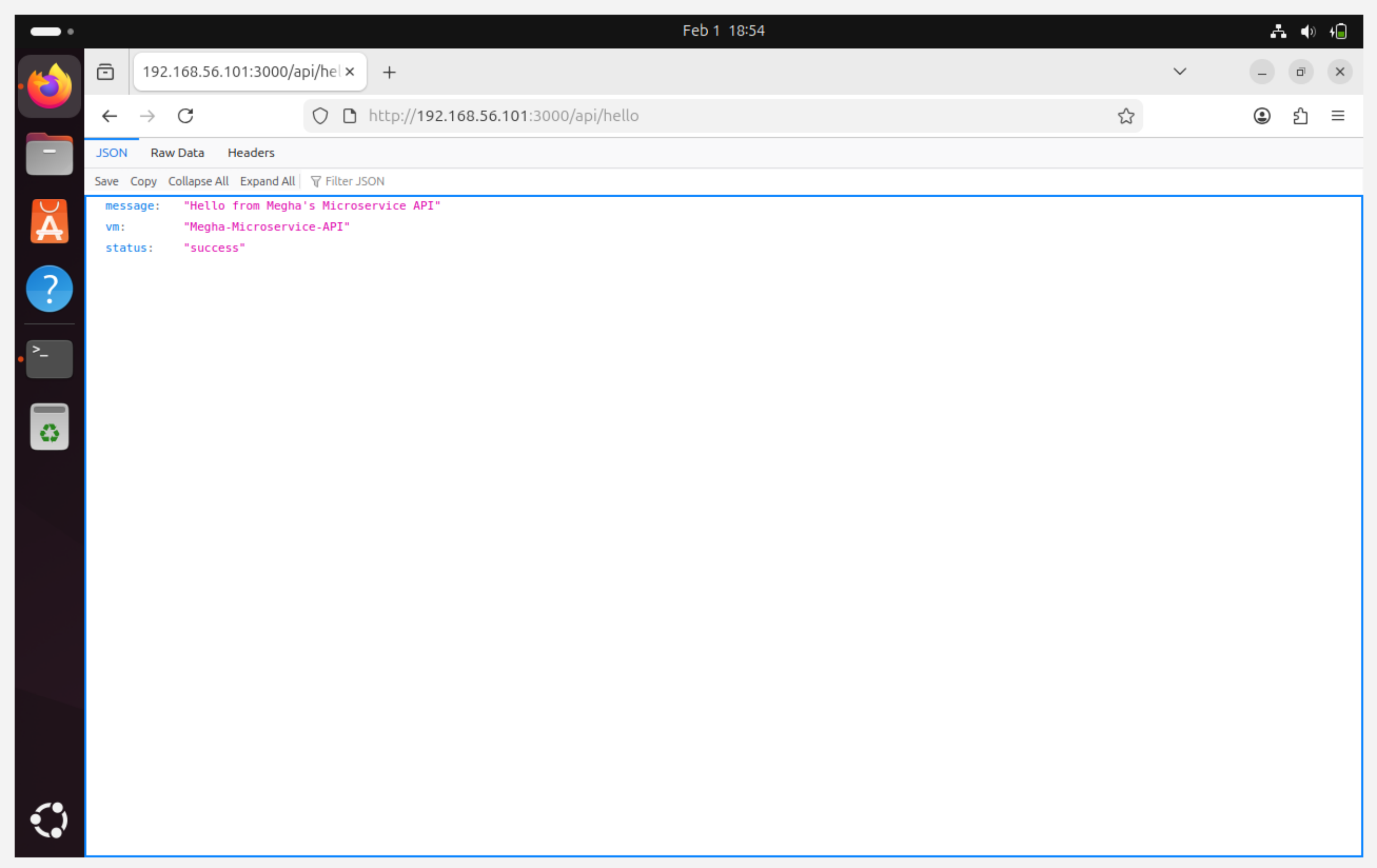
```
curl http://192.168.56.101:3000/api/hello
```

Response

```
{
  "message": "Hello from Megha's Microservice API",
  "vm": "Megha_microservice_api",
  "status": "success"
}
```



```
megha-client@megha-client-VirtualBox:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:2d:93:a8 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.102/24 brd 192.168.56.255 scope global dynamic noprefixroute enp0s3
       valid_lft 346sec preferred_lft 346sec
    inet6 fe80::a00:27ff:fe2d:93a8/64 scope link
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:45:9e:4d brd ff:ff:ff:ff:ff:ff
    inet 10.0.3.15/24 brd 10.0.3.255 scope global dynamic noprefixroute enp0s8
       valid_lft 81646sec preferred_lft 81646sec
    inet6 fd17:625c:f037:3:f4cc:a01c:4de5:86f5/64 scope global temporary dynamic
       valid_lft 86027sec preferred_lft 14027sec
    inet6 fd17:625c:f037:3:33ce:8849:5589:1d63/64 scope global dynamic mngtmpaddr noprefixroute
       valid_lft 86027sec preferred_lft 14027sec
    inet6 fe80::2fb9:9770:5215:5dba/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
megha-client@megha-client-VirtualBox:~$ curl http://192.168.56.101:3000/api/hello
{"message":"Hello from Megha's Microservice API","vm":"Megha-Microservice-API","status":"success"}megha-client@megha-cli
ent-VirtualBox:~$
```

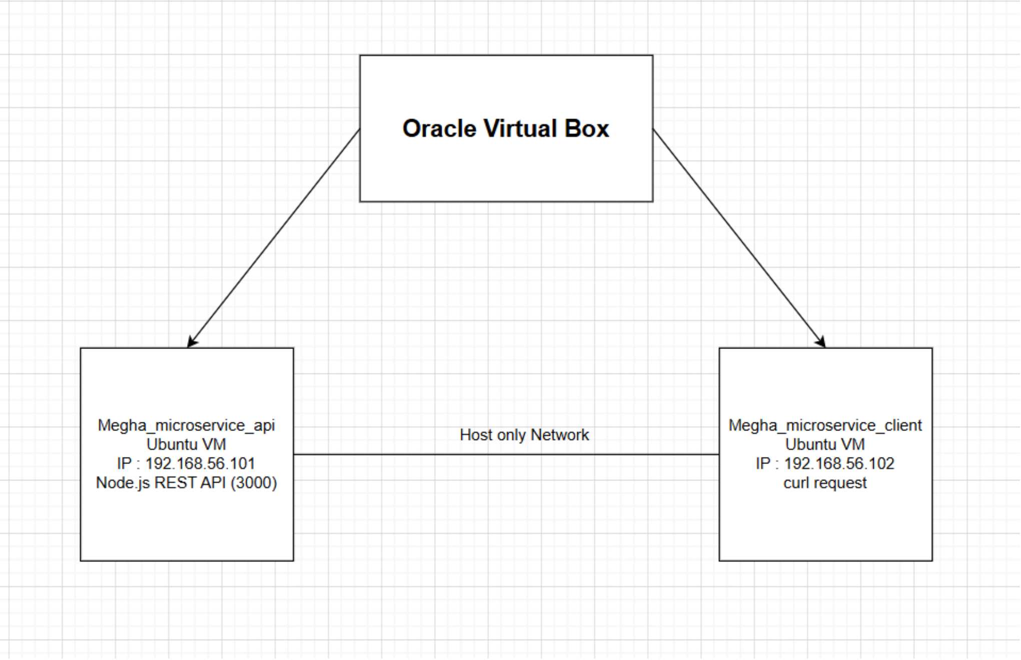


```
message:   "Hello from Megha's Microservice API"
vm:        "Megha-Microservice-API"
status:    "success"
```

7. Architecture Design

System Architecture Description

- Two Ubuntu VMs running on Oracle VirtualBox
- Connected via a Host-only private network
- API VM hosts the Node.js microservice
- Client VM consumes the REST API using HTTP



8. Source Code Repository

GitHub Repository:

https://github.com/Codewith-Megha/Multi-VM-Network-Configuration-and-Microservice-Deployment

9. Video Demonstration

Recorded Video Link:

https://drive.google.com/drive/folders/1sVFxNfgBRCop5wJU0kQuNBONS4jkpqj9

The video demonstrates:

- VM creation
- Network configuration
- Ping test
- Node.js API execution
- Client accessing the API