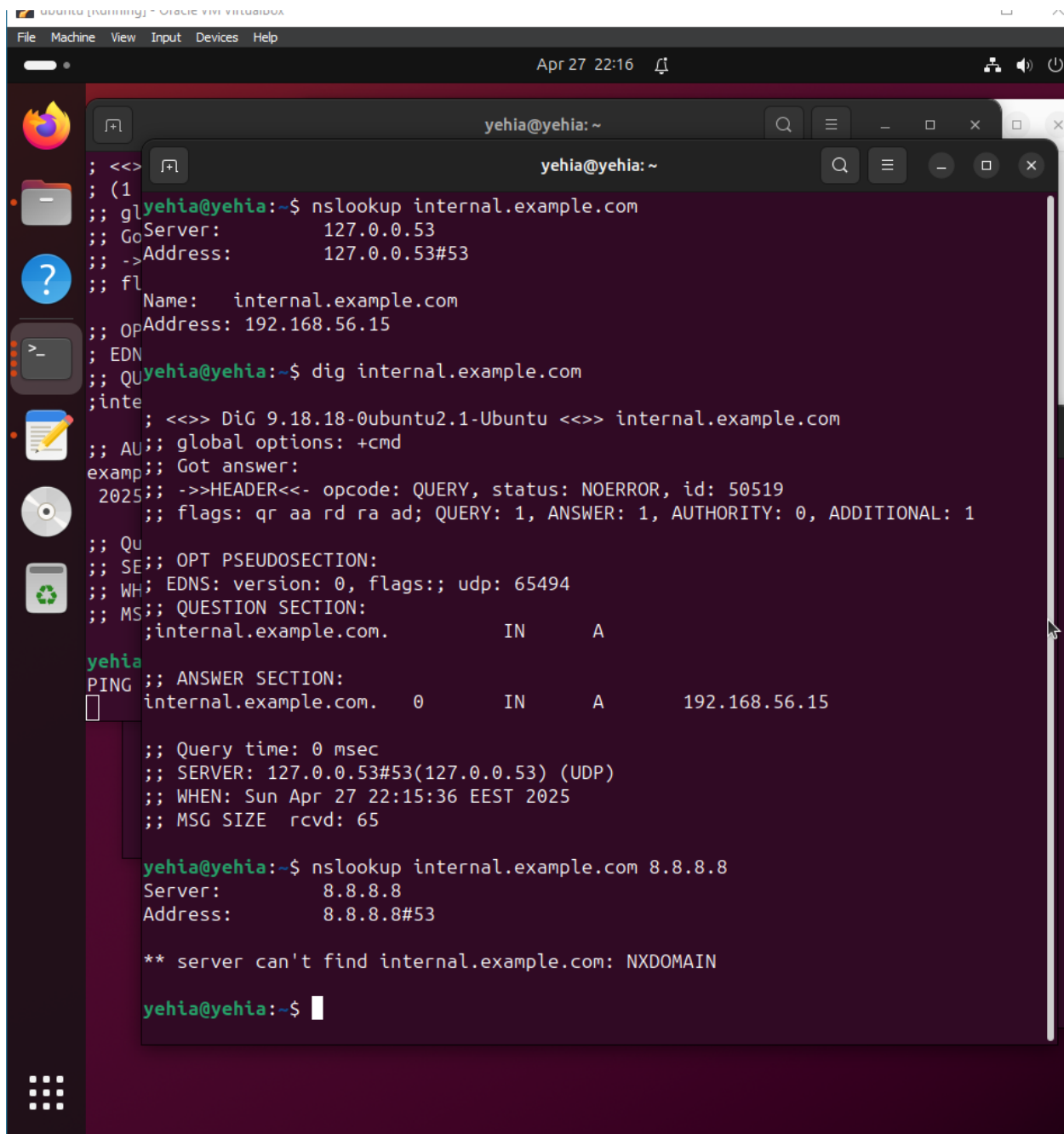


## 1. Verify DNS Resolution:

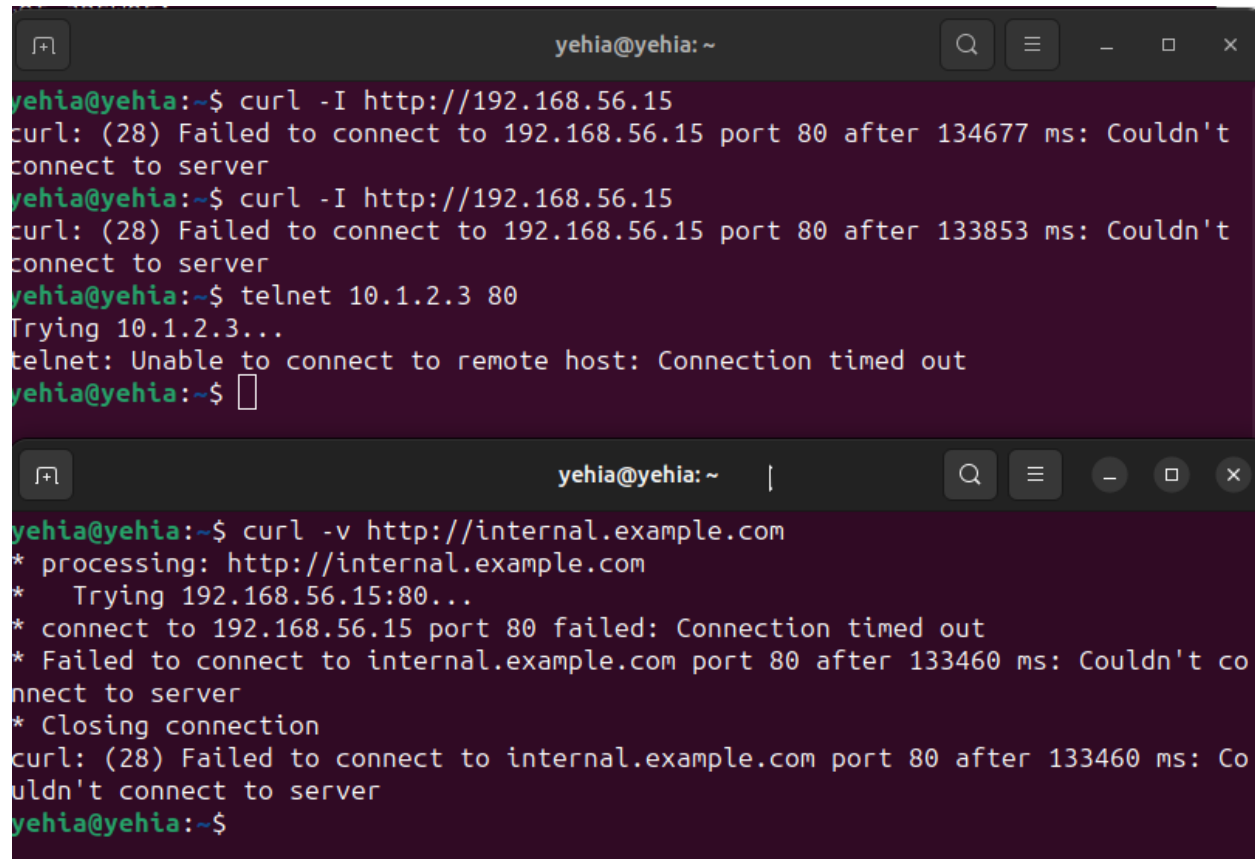
Check what IP internal.example.com resolves.



The screenshot shows a terminal window within a virtual machine environment. The terminal displays the results of two DNS lookup commands: `nslookup` and `dig`. The `nslookup` command is used twice: first with the default server (127.0.0.53) and then with the external server 8.8.8.8. The `dig` command is also used to query the default server. The output shows that the domain `internal.example.com` resolves to the IP address `192.168.56.15` when using the default server, but fails to resolve (NXDOMAIN) when using the external server 8.8.8.8.

```
yehia@yehia: ~  
; <<> DiG 9.18.18-0ubuntu2.1-Ubuntu <<> internal.example.com  
;; global options: +cmd  
;; Got answer:  
2025;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 50519  
;; flags: qr aa rd ra ad; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1  
;; QUESTION SECTION:  
internal.example.com. IN A  
;; ANSWER SECTION:  
internal.example.com. 0 IN A 192.168.56.15  
;; Query time: 0 msec  
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)  
;; WHEN: Sun Apr 27 22:15:36 EEST 2025  
;; MSG SIZE rcvd: 65  
  
yehia@yehia:~$ nslookup internal.example.com 8.8.8.8  
Server: 8.8.8.8  
Address: 8.8.8.8#53  
  
** server can't find internal.example.com: NXDOMAIN  
  
yehia@yehia:~$
```

## 2. Diagnose Service Reachability:



The image shows two terminal windows from a user named 'yehia' on a machine named 'yehia'. The top terminal shows two failed curl commands to 'http://192.168.56.15' and one failed telnet command to '10.1.2.3 80'. The bottom terminal shows a detailed curl command to 'http://internal.example.com' which also fails, providing more context about the connection attempt and timeout.

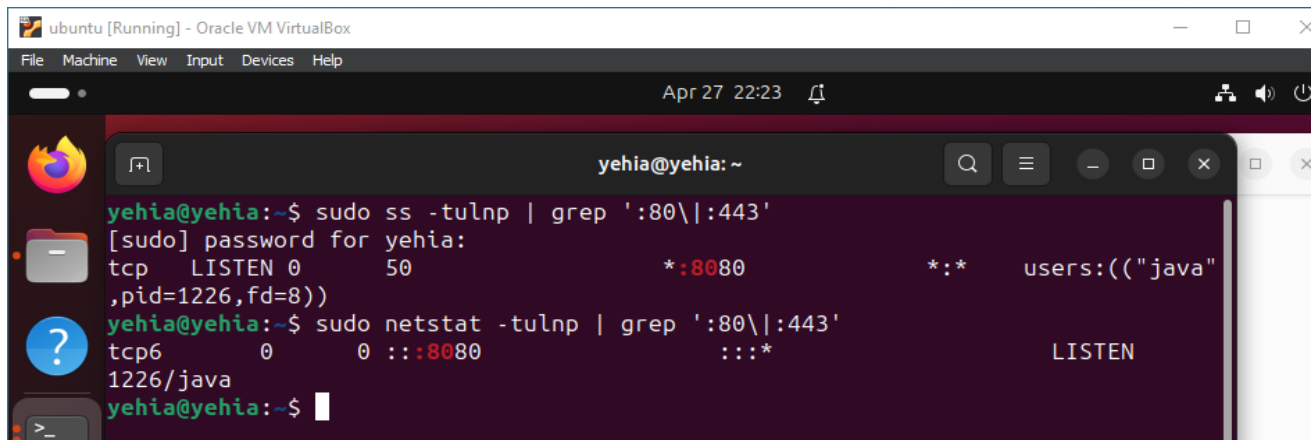
```
yehia@yehia:~$ curl -I http://192.168.56.15
curl: (28) Failed to connect to 192.168.56.15 port 80 after 134677 ms: Couldn't
connect to server
yehia@yehia:~$ curl -I http://192.168.56.15
curl: (28) Failed to connect to 192.168.56.15 port 80 after 133853 ms: Couldn't
connect to server
yehia@yehia:~$ telnet 10.1.2.3 80
Trying 10.1.2.3...
telnet: Unable to connect to remote host: Connection timed out
yehia@yehia:~$

yehia@yehia:~$ curl -v http://internal.example.com
* processing: http://internal.example.com
*   Trying 192.168.56.15:80...
* connect to 192.168.56.15 port 80 failed: Connection timed out
* Failed to connect to internal.example.com port 80 after 133460 ms: Couldn't co
nnect to server
* Closing connection
curl: (28) Failed to connect to internal.example.com port 80 after 133460 ms: Co
uldn't connect to server
yehia@yehia:~$
```

Test if the service port is open for port 80 (HTTP) using

- telnet 10.1.2.3 80
- and curl to actually hit the service:
- curl -v <http://internal.example.com>

telnet fails, so it's a network or firewall problem. Also curl fails, so it can be web server misconfiguration

A terminal window titled 'yehia@yehia: ~' showing the output of two commands. The first command is 'sudo ss -tulnp | grep ':80\|:443'', which returns 'tcp LISTEN 0 50 \*:8080 \*: \* users:(("java",pid=1226,fd=8))'. The second command is 'sudo netstat -tulnp | grep ':80\|:443'', which returns 'tcp6 0 0 :::8080 :::\* LISTEN 1226/java'. The terminal background is dark purple with green and white text.

```
yehia@yehia:~$ sudo ss -tulnp | grep ':80\|:443'
[sudo] password for yehia:
tcp LISTEN 0 50 *:8080 *: * users:(("java",pid=1226,fd=8))
yehia@yehia:~$ sudo netstat -tulnp | grep ':80\|:443'
tcp6 0 0 :::8080 :::* LISTEN
1226/java
yehia@yehia:~$
```

There is NO service listening on port 80 (HTTP).

## A. DNS Misconfiguration

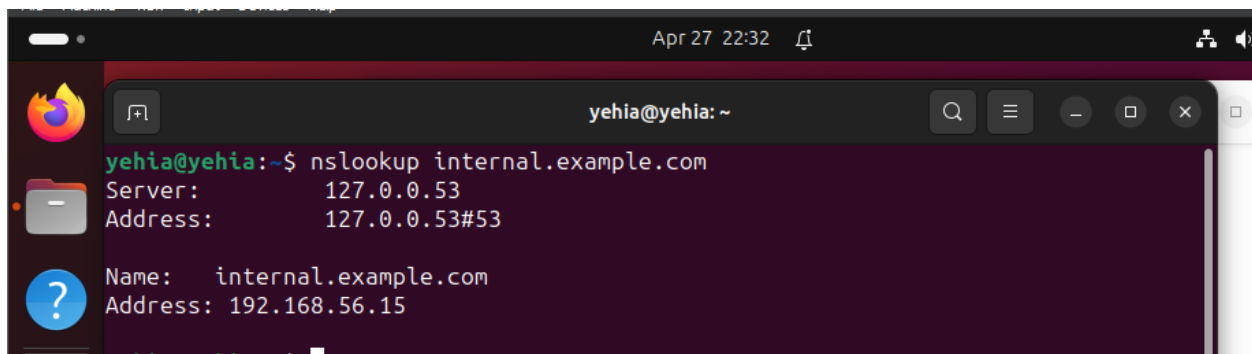
nslookup internal.example.com

If I get

- no answer
- Wrong IP

Then DNS misconfiguration is the cause.

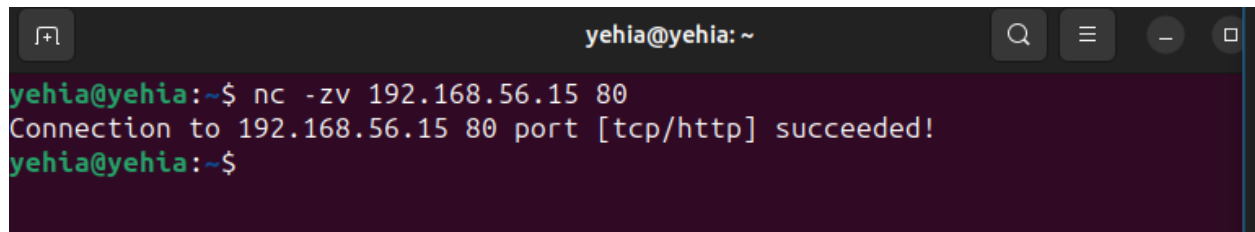
But from what I get, DNS is working correctly so this means DNS misconfiguration is not the root cause.

A terminal window titled 'yehia@yehia: ~' showing the output of the 'nslookup internal.example.com' command. The output shows the server as 127.0.0.53, the address as 127.0.0.53#53, and the name and address for internal.example.com as 192.168.56.15. The terminal background is dark purple with green and white text.

```
yehia@yehia:~$ nslookup internal.example.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Name:   internal.example.com
Address: 192.168.56.15
yehia@yehia:~$
```

## B. Firewall Blocking Ports

A terminal window with a dark background. The title bar shows 'yehia@yehia: ~'. The prompt is 'yehia@yehia:~\$'. The user enters 'nc -zv 192.168.56.15 80'. The output is 'Connection to 192.168.56.15 80 port [tcp/http] succeeded!'. The prompt returns to 'yehia@yehia:~\$'.

```
yehia@yehia:~$ nc -zv 192.168.56.15 80
Connection to 192.168.56.15 80 port [tcp/http] succeeded!
yehia@yehia:~$
```

If I get:

- Connection refused
- Timed out

Then the firewall is likely blocking the ports. But in my case, firewall is not the root problem.

If firewall was the problem, I would fix it by allowing incoming connections. By using

- Sudo ufw allow <port number>
- Sufo ufw reload