0x08. Networking basics #1

**DevOpsNetworkSysAdmin**

* By: Sylvain Kalache
* Weight: 1
* Project over - took place from Oct 4, 2023 5:00 AM to Oct 6, 2023 5:00 AM
* An auto review will be launched at the deadline

In a nutshell…

* **Auto QA review:** 0.0/3 mandatory & 0.0/1 optional
* **Altogether:**  **0.0%**
  + Mandatory: 0.0%
  + Optional: 0.0%
  + Calculation:  0.0% + (0.0% \* 0.0%)  == **0.0%**



Resources

**Read or watch**:

* [What is localhost](https://intranet.alxswe.com/rltoken/Odcc_tyAQlcANCCrtmxo6A)
* [What is 0.0.0.0](https://intranet.alxswe.com/rltoken/fUb9IpnxrNaddMljzwbhJQ)
* [What is the hosts file](https://intranet.alxswe.com/rltoken/4_MBpFTulKliFM69jCPzOQ)
* [Netcat examples](https://intranet.alxswe.com/rltoken/OR0lOEwAw9I1Rj4aGp1Ljg)

**man or help**:

* ifconfig
* telnet
* nc
* cut

Learning Objectives

At the end of this project, you are expected to be able to [explain to anyone](https://intranet.alxswe.com/rltoken/IpTKeVwlKHT4ZVva_T891w), **without the help of Google**:

General

* What is localhost/127.0.0.1
* What is 0.0.0.0
* What is /etc/hosts
* How to display your machine’s active network interfaces

Copyright - Plagiarism

* You are tasked to come up with solutions for the tasks below yourself to meet with the above learning objectives.
* You will not be able to meet the objectives of this or any following project by copying and pasting someone else’s work.
* You are not allowed to publish any content of this project.
* Any form of plagiarism is strictly forbidden and will result in removal from the program.

Requirements

General

* Allowed editors: vi, vim, emacs
* All your files will be interpreted on Ubuntu 20.04 LTS
* All your files should end with a new line
* A README.md file, at the root of the folder of the project, is mandatory
* All your Bash script files must be executable
* Your Bash script must pass Shellcheck (version 0.7.0 via apt-get) without any errors
* The first line of all your Bash scripts should be exactly #!/usr/bin/env bash
* The second line of all your Bash scripts should be a comment explaining what is the script doing

Quiz questions

**Great!** You've completed the quiz successfully! Keep going! (Show quiz)

Tasks

0. Change your home IP

**mandatory**

Score: 0.0% (*Checks completed: 0.0%*)

Write a Bash script that configures an Ubuntu server with the below requirements.

Requirements:

* localhost resolves to 127.0.0.2
* facebook.com resolves to 8.8.8.8.
* The checker is running on Docker, so make sure to read [this](https://intranet.alxswe.com/rltoken/XSXhQPoDu3QecXs3j9XgPQ)

Example:

sylvain@ubuntu$ ping localhost

PING localhost (127.0.0.1) 56(84) bytes of data.

64 bytes from localhost (127.0.0.1): icmp\_seq=1 ttl=64 time=0.012 ms

^C

--- localhost ping statistics ---

1 packets transmitted, 1 received, 0% packet loss, time 0ms

rtt min/avg/max/mdev = 0.012/0.012/0.012/0.000 ms

sylvain@ubuntu$

sylvain@ubuntu$ ping facebook.com

PING facebook.com (157.240.11.35) 56(84) bytes of data.

64 bytes from edge-star-mini-shv-02-lax3.facebook.com (157.240.11.35): icmp\_seq=1 ttl=63 time=15.4 ms

^C

--- facebook.com ping statistics ---

1 packets transmitted, 1 received, 0% packet loss, time 0ms

rtt min/avg/max/mdev = 15.432/15.432/15.432/0.000 ms

sylvain@ubuntu$

sylvain@ubuntu$ sudo ./0-change\_your\_home\_IP

sylvain@ubuntu$

sylvain@ubuntu$ ping localhost

PING localhost (127.0.0.2) 56(84) bytes of data.

64 bytes from localhost (127.0.0.2): icmp\_seq=1 ttl=64 time=0.012 ms

64 bytes from localhost (127.0.0.2): icmp\_seq=2 ttl=64 time=0.036 ms

^C

--- localhost ping statistics ---

2 packets transmitted, 2 received, 0% packet loss, time 1000ms

rtt min/avg/max/mdev = 0.012/0.024/0.036/0.012 ms

sylvain@ubuntu$

sylvain@ubuntu$ ping facebook.com

PING facebook.com (8.8.8.8) 56(84) bytes of data.

64 bytes from facebook.com (8.8.8.8): icmp\_seq=1 ttl=63 time=8.06 ms

^C

--- facebook.com ping statistics ---

1 packets transmitted, 1 received, 0% packet loss, time 0ms

rtt min/avg/max/mdev = 8.065/8.065/8.065/0.000 ms

In this example we can see that:

* before running the script, localhost resolves to 127.0.0.1 and facebook.com resolves to 157.240.11.35
* after running the script, localhost resolves to 127.0.0.2 and facebook.com resolves to 8.8.8.8

If you’re running this script on a machine that you’ll continue to use, be sure to revert localhost to 127.0.0.1. Otherwise, a lot of things will stop working!

**Repo:**

* GitHub repository: alx-system\_engineering-devops
* Directory: 0x08-networking\_basics\_2
* File: 0-change\_your\_home\_IP

 Done! Help Check your code Ask for a new correction Get a sandbox QA Review

1. Show attached IPs

**mandatory**

Score: 0.0% (*Checks completed: 0.0%*)

Write a Bash script that displays all active IPv4 IPs on the machine it’s executed on.

Example:

sylvain@ubuntu$ ./1-show\_attached\_IPs | cat -e

10.0.2.15$

127.0.0.1$

sylvain@ubuntu$

Obviously, the IPs displayed may be different depending on which machine you are running the script on.

Note that we can see our localhost IP :)

**Repo:**

* GitHub repository: alx-system\_engineering-devops
* Directory: 0x08-networking\_basics\_2
* File: 1-show\_attached\_IPs

 Done? Help Check your code Ask for a new correction Get a sandbox QA Review

2. Port listening on localhost

**#advanced**

Score: 0.0% (*Checks completed: 0.0%*)

Write a Bash script that listens on port 98 on localhost.

**Terminal 0**

Starting my script.

sylvain@ubuntu$ sudo ./100-port\_listening\_on\_localhost

**Terminal 1**

Connecting to localhost on port 98 using telnet and typing some text.

sylvain@ubuntu$ telnet localhost 98

Trying 127.0.0.2...

Connected to localhost.

Escape character is '^]'.

Hello world

test

**Terminal 0**

Receiving the text on the other side.

sylvain@ubuntu$ sudo ./100-port\_listening\_on\_localhost

Hello world

test

For the sake of the exercise, this connection is made entirely within localhost. This isn’t really exciting as is, but we can use this script across networks as well. Try running it between your local PC and your remote server for fun!

As you can see, this can come in very handy in a multitude of situations. Maybe you’re debugging socket connection issues, or you’re trying to connect to a software and you are unsure if the issue is the software or the network, or you’re working on firewall rules… Another tool to add to your debugging toolbox!

**Repo:**

* GitHub repository: alx-system\_engineering-devops
* Directory: 0x08-networking\_basics\_2
* File: 100-port\_listening\_on\_localhost

 Done! Help Check your code Ask for a new correction Get a sandbox QA Review

Copyright © 2023 ALX, All rights reserved.