



(/)




Curriculum

SE Foundations 
Average: 123.8% 

0x0C. Web server

DevOps **SysAdmin**

 Weight: 1

 Project over - took place from Mar 25, 2024 4:00 AM to Mar 27, 2024 4:00 AM

☒ An auto review will be launched at the deadline

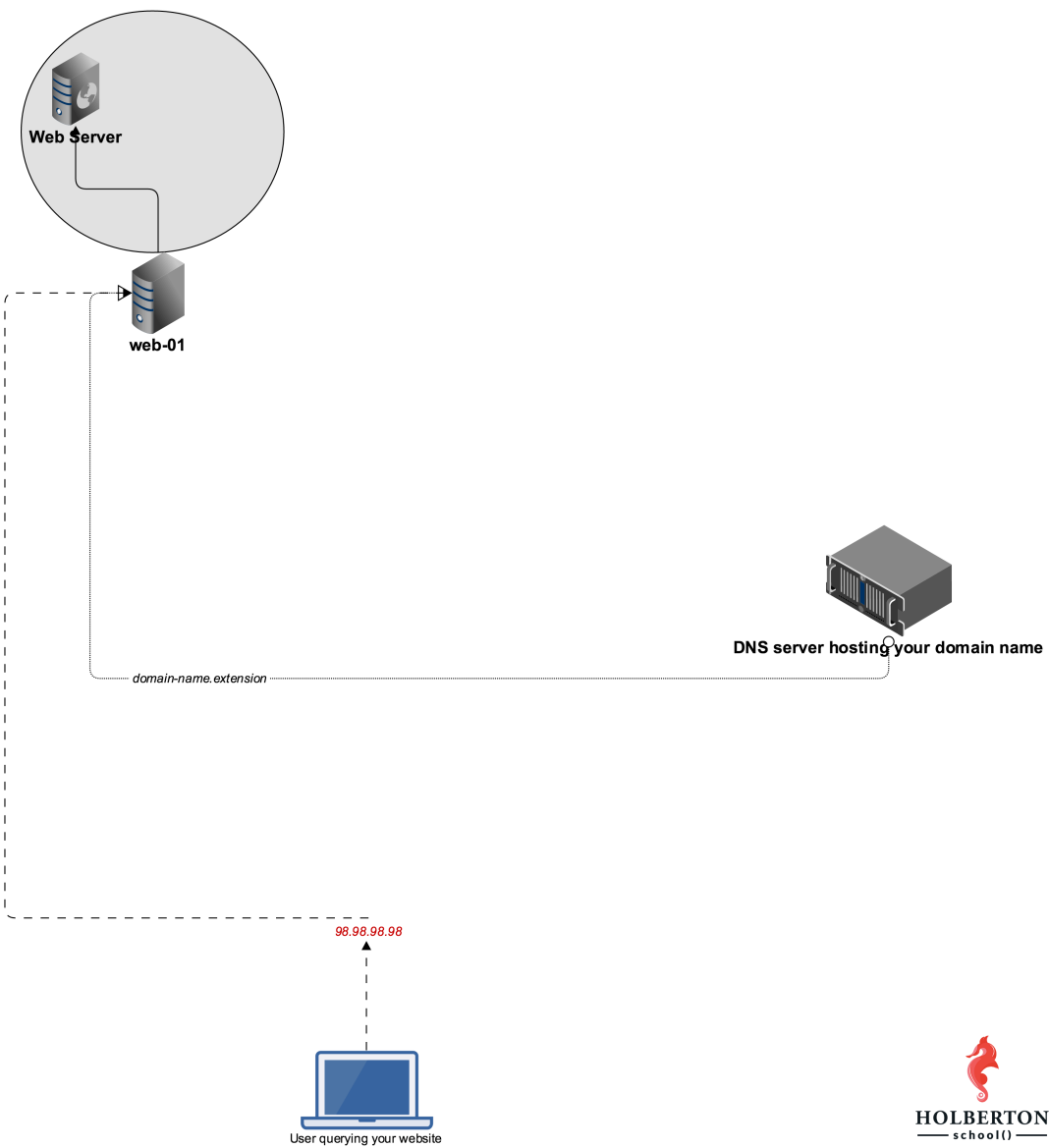
In a nutshell...

- **Auto QA review:** 0.0/13 mandatory & 0.0/4 optional
- **Altogether: 0.0%**
 - Mandatory: 0.0%
 - Optional: 0.0%
 - Calculation: $0.0\% + (0.0\% * 0.0\%) == \mathbf{0.0\%}$

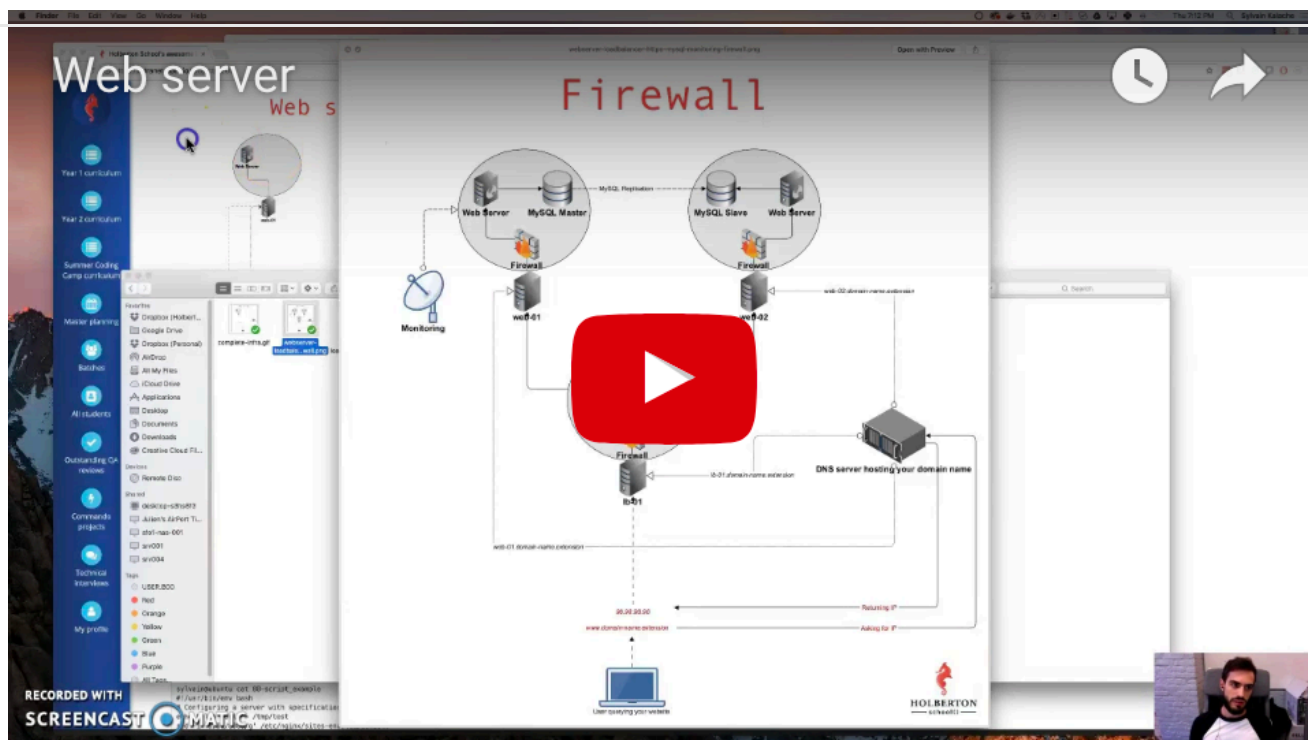


(/)

Web server



Background Context



(<https://www.youtube.com/watch?v=AZg4uJkEa-4&feature=youtu.be&hd=1>)

In this project, some of the tasks will be graded on 2 aspects:

1. Is your web-01 server configured according to requirements
2. Does your answer file contain a Bash script that automatically performs commands to configure an Ubuntu machine to fit requirements (meaning without any human intervention)

For example, if I need to create a file `/tmp/test` containing the string `hello world` and modify the configuration of Nginx to listen on port `8080` instead of `80`, I can use `emacs` on my server to create the file and to modify the Nginx configuration file `/etc/nginx/sites-enabled/default`.

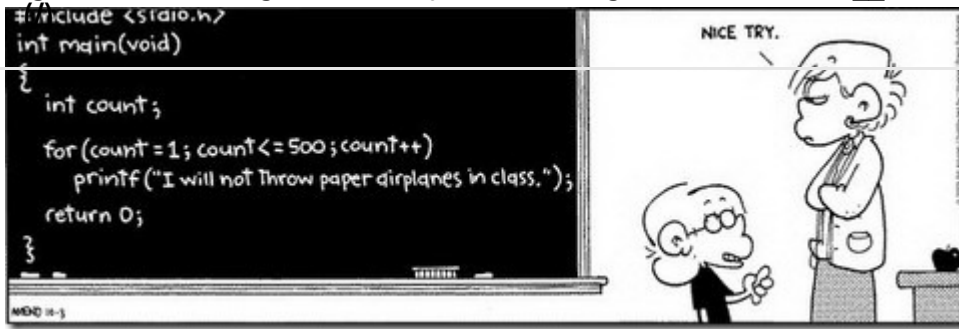
But my answer file would contain:

```
sylvain@ubuntu cat 88-script_example
#!/usr/bin/env bash
# Configuring a server with specification XYZ
echo hello world > /tmp/test
sed -i 's/80/8080/g' /etc/nginx/sites-enabled/default
sylvain@ubuntu
```

As you can tell, I am not using `emacs` to perform the task in my answer file. This exercise is aiming at training you on automating your work. If you can automate tasks that you do manually, you can then automate yourself out of repetitive tasks and focus your energy on something more interesting. For an SRE ([rltoken/9l0WufjKdW3TZA2EVrGnIQ](https://www.linkedin.com/in/rltoken/9l0WufjKdW3TZA2EVrGnIQ)), that comes very handy when there are hundreds or thousands of servers to manage, the work cannot be only done manually. Note that the checker will execute your script as the `root` user, you do not need to use the `sudo` command.



A good Software Engineer is a lazy Software Engineer (/rltoken/sRY__axKNHhNW0SVmsUC_A).



Tips: to test your answer Bash script, feel free to reproduce the checker environment:

- start a Ubuntu 16.04 sandbox
- run your script on it
- see how it behaves

Resources

Read or watch:

- How the web works (/rltoken/6TI3HiyFdwrBxWKVF24Gxw)
- Nginx (/rltoken/vkVMGlaf39j2DWAQWzo6EA)
- How to Configure Nginx (/rltoken/zKrpVxWuUHVdW4URAJdFbw)
- Child process concept page (/rltoken/Ar18u5sRis1fkVvVgzdcqg)
- Root and sub domain (/rltoken/xi3peVqYI02PfpHHHICtxQ)
- HTTP requests (/rltoken/sBrrP4EAml3NoYjlgZrUhw)
- HTTP redirection (/rltoken/Eaa4ZuKvye941hTkP8VIBQ)
- Not found HTTP response code (/rltoken/eJSp2QFTY6jqqNtz8OVDEw)
- Logs files on Linux (/rltoken/7WMNY5CWD-CBrxmQrdmfPg)

For reference:

- RFC 7231 (HTTP/1.1) (/rltoken/BGa6RrS0dnM6EdBGS_ZDUw)
- RFC 7540 (HTTP/2) (/rltoken/I22fyYn1qNZ9RXXsg5vG1g)

man or help:

- scp
- curl

Learning Objectives

At the end of this project, you are expected to be able to explain to anyone (/rltoken/EHyxclwPtD2SzEGRKOnT3g), **without the help of Google**:

General

- What is the main role of a web server
- What is a child process
- Why web servers usually have a parent process and child processes
- What are the main HTTP requests

DNS

- What DNS stands for



- What is DNS main role

(/)

DNS Record Types

- A
- CNAME
- TXT
- MX

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 16.04 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 `0.3.7`) without any error
- 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
- You can't use `systemctl` for restarting a process

Quiz questions

Great! You've completed the quiz successfully! Keep going! ([Show quiz](#))

Your servers

Name	Username	IP	State	
521277-web-01	ubuntu	100.27.13.16	running	Actions ▼



Tasks

0. Transfer a file to your server

mandatory

Score: 0.0% (Checks completed: 0.0%)

Write a Bash script that transfers a file from our client to a server:

Requirements:

- Accepts 4 parameters
 1. The path to the file to be transferred
 2. The IP of the server we want to transfer the file to
 3. The username `scp` connects with
 4. The path to the SSH private key that `scp` uses
- Display Usage: `0-transfer_file PATH_TO_FILE IP USERNAME PATH_TO_SSH_KEY` if less than 3 parameters passed
- `scp` must transfer the file to the user home directory `~/`
- Strict host key checking must be disabled when using `scp`

Example:

```
sylvain@ubuntu$ ./0-transfer_file
Usage: 0-transfer_file PATH_TO_FILE IP USERNAME PATH_TO_SSH_KEY
sylvain@ubuntu$
sylvain@ubuntu$ ssh ubuntu@8.8.8.8 -i /vagrant/sylvain 'ls ~/
afile
sylvain@ubuntu$
sylvain@ubuntu$ touch some_page.html
sylvain@ubuntu$ ./0-transfer_file some_page.html 8.8.8.8 sylvain /vagrant/private
_key
some_page.html                               100% 12      0.1KB/s   00:00
sylvain@ubuntu$ ssh ubuntu@8.8.8.8 -i /vagrant/private_key 'ls ~/
afile
some_page.html
sylvain@ubuntu$
```

In this example, I:

- remotely execute the `ls ~/` command via `ssh` to see what `~/` contains
- create a file named `some_page.html`
- execute my `0-transfer_file` script
- remotely execute the `ls ~/` command via `ssh` to see that the file `some_page.html` has been successfully transferred

That is one way of publishing your website pages to your server.

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x0C-web_server`
- File: `0-transfer_file`



(A) Done?

Check your code

Ask for a new correction

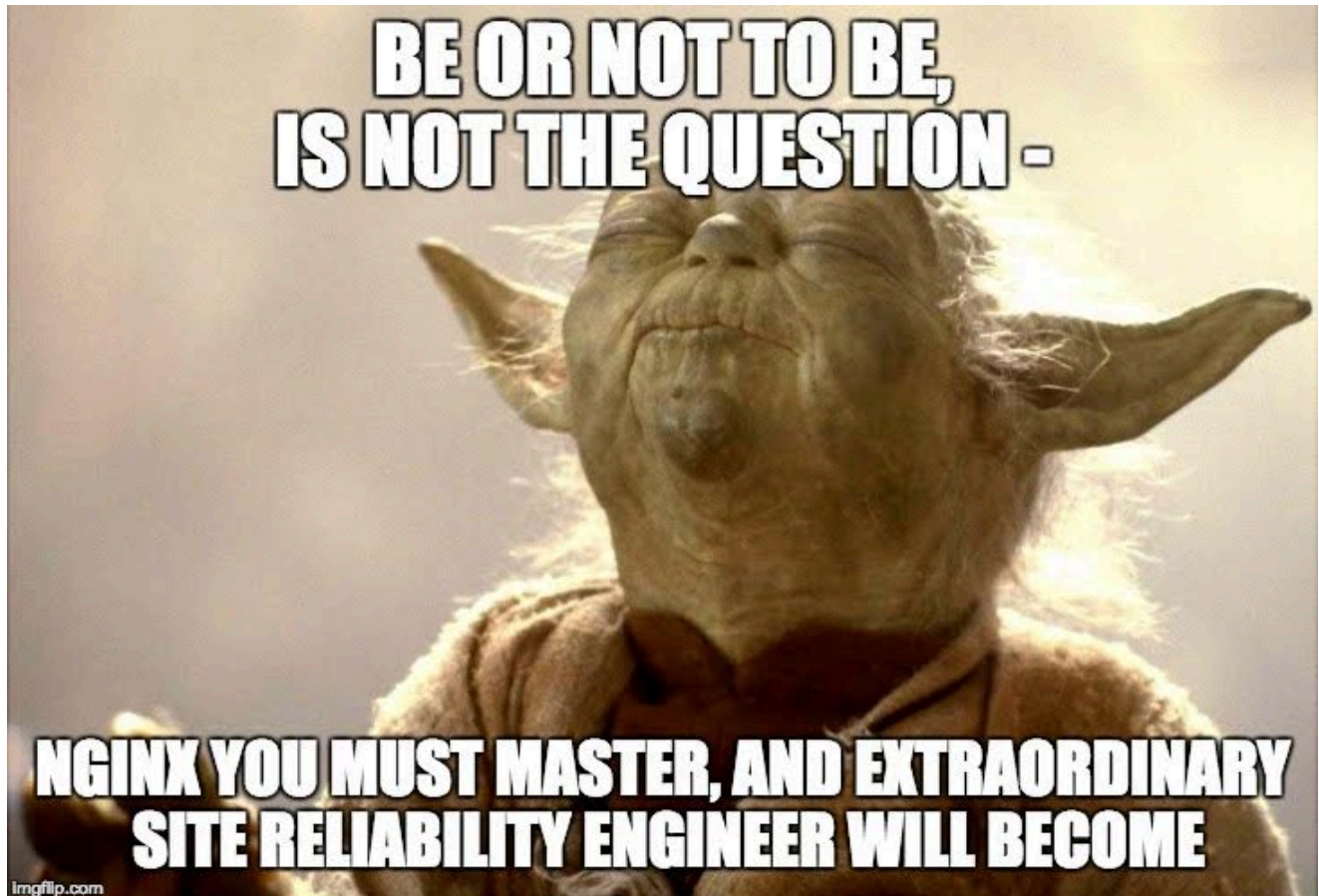
> Get a sandbox

QA Review

1. Install nginx web server

mandatory

Score: 0.0% (Checks completed: 0.0%)



Readme:

- -y on apt-get command (/rltoken/KJiFZ4yJyTGp_cv3DYQLaQ)

Web servers are the piece of software generating and serving HTML pages, let's install one!

Requirements:

- Install nginx on your web-01
- server
- Nginx should be listening on port 80
- When querying Nginx at its root / with a GET request (requesting a page) using curl, it must return a page that contains the string Hello world!
- As an answer file, write a Bash script that configures a new Ubuntu machine to respect above requirements (this script will be run on the server itself)
- You can't use systemctl for restarting nginx

Server terminal:



```
root@sy-web-01$ ./1-install_nginx_web_server > /dev/null 2>&1
root@sy-web-01$
root@sy-web-01$ curl localhost
Hello World!
root@sy-web-01$
```

Local terminal:

```
sylvain@ubuntu$ curl 34.198.248.145/
Hello World!
sylvain@ubuntu$ curl -sI 34.198.248.145/
HTTP/1.1 200 OK
Server: nginx/1.4.6 (Ubuntu)
Date: Tue, 21 Feb 2017 23:43:22 GMT
Content-Type: text/html
Content-Length: 30
Last-Modified: Tue, 21 Feb 2017 07:21:32 GMT
Connection: keep-alive
ETag: "58abea7c-1e"
Accept-Ranges: bytes

sylvain@ubuntu$
```

In this example 34.198.248.145 is the IP of my web-01 server. If you want to query the Nginx that is locally installed on your server, you can use `curl 127.0.0.1`.

If things are not going as expected, make sure to check out Nginx logs, they can be found in `/var/log/`.

Maarten's PRO-tip: When you use `sudo su` on your web-01 you can become root like this to test your file:

```
sylvain@ubuntu$ sudo su
root@ubuntu#
```

Repo:

- GitHub repository: `alx-system_engineering-devops`
- Directory: `0x0C-web_server`
- File: `1-install_nginx_web_server`

☐ Done?

2. Setup a domain name

mandatory

Score: 0.0% (Checks completed: 0.0%)



.TECH Domains (<https://get.tech/>) is one of the top domain providers. They are known for the stability and quality of their DNS hosting solution. We partnered with .TECH Domains so that you can learn about DNS.

YOU can have a **free .tech domain** for 1 year by following these steps:

(/)

- Access the tools space (/dashboards/my_tools)
- Unlock the **GitHub student pack**: WARNING - this invitation link is unique to you and can't be reclaimed! If you have any issue, please contact GitHub education support (<https://support.github.com/request/landing>)

GitHub Student Developer Pack

Learn to ship software like a pro

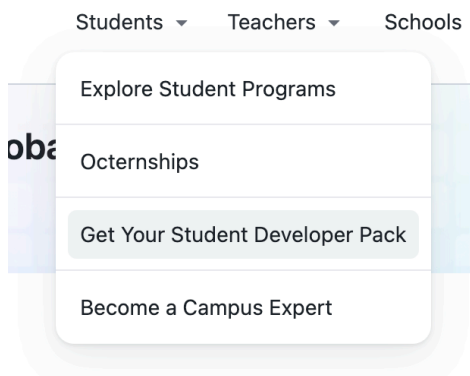
There's no substitute for hands-on experience. But for most learners, real world tools can be cost-prohibitive.

That's why we created the GitHub Student Developer Pack with some of our partners and friends: to give learners free access to the best developer tools in one place so they can learn by doing.


[More information](#)

Access to the pack [here](#)

- When registered, access your benefits (https://github.com/login?client_id=de7e3b6548f2ed9bbceb&return_to=%2Flogin%2Foauth%2Fauthorize%3Fclient_id%3Dc



- And scroll to **.Tech domain**:



About .TECH
A powerful domain extension to convey that you belong to the technology industry.

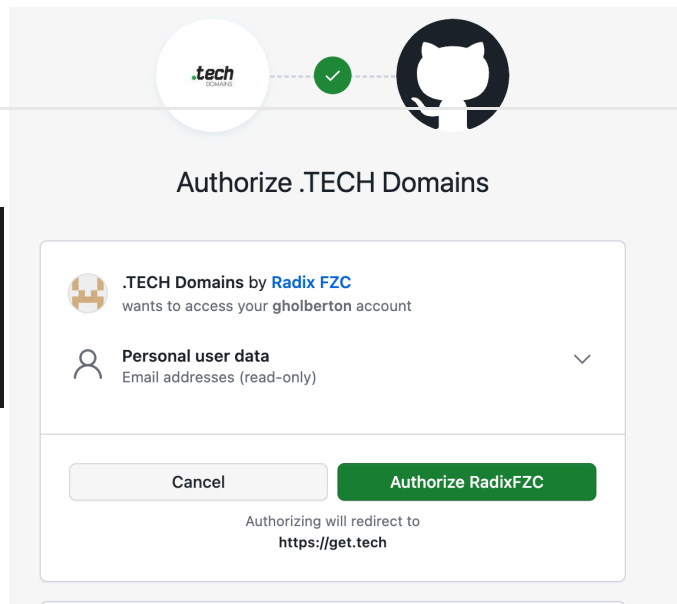
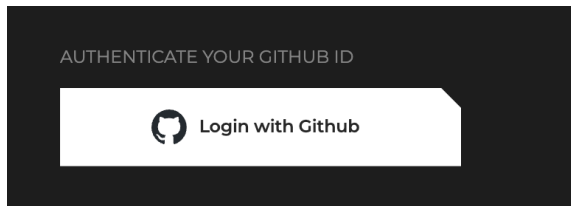
Offer
One standard .TECH domain free for 1 year and 2 free email accounts with 100 MB free storage
[Get access by connecting your GitHub account on .TECH >](#)

Get help at [.TECH support](#)

- Start to register your domain and checkout
- At the Checkout step, please click on "Login with GitHub":



(/)



- The cost of the domain should be now at **\$0**
- You can finalize it by creating an account in .Tech domains (<https://get.tech/>)
- And manage your domain there!

Provide the domain name in your answer file.

Requirement:

- provide the domain name only (example: `foobar.tech`), no subdomain (example: `www.foobar.tech`)
- configure your DNS records with an A entry so that your root domain points to your `web-01` IP address **Warning: the propagation of your records can take time (~1-2 hours)**
- go to your profile (`/users/my_profile`) and enter your domain in the `Project website url` field

Example:



```
sylvain@ubuntu$ cat 2-setup_a_domain_name
myschool.tech
sylvain@ubuntu$
sylvain@ubuntu$ dig myschool.tech

; <<>> DiG 9.10.6 <<>> myschool.tech
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 26785
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags;; udp: 512
;; QUESTION SECTION:
;myschool.tech.      IN  A

;; ANSWER SECTION:
myschool.tech. 7199      IN  A   184.72.193.201

;; Query time: 65 msec
;; SERVER: 8.8.8.8#53(8.8.8.8)
;; WHEN: Fri Aug 02 09:44:36 PDT 2019
;; MSG SIZE rcvd: 65

sylvain@ubuntu$
```

When your domain name is setup, please verify the Registrar here: <https://whois.whoisxmlapi.com/> (<https://whois.whoisxmlapi.com/>) and you must see in the JSON response: "registrarName": "Dotserve Inc"

Repo:

- GitHub repository: alx-system_engineering-devops
- Directory: 0x0C-web_server
- File: 2-setup_a_domain_name

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

3. Redirection

mandatory

Score: 0.0% (Checks completed: 0.0%)

Readme:

- Replace a line with multiple lines with sed (/rltoken/RRP9hX3MIQdABaKZD-Y_cA)

Configure your Nginx server so that /redirect_me is redirecting to another page.

Requirements:

- The redirection must be a "301 Moved Permanently"
- You answer file should be a Bash script containing commands to automatically configure a Ubuntu machine to respect above requirements



- Using what you did with 1-install_nginx_web_server , write 3-redirection so that it (/) configures a brand new Ubuntu machine to the requirements asked in this task

Example:

```
sylvain@ubuntu$ curl -sI 34.198.248.145/redirect_me/
HTTP/1.1 301 Moved Permanently
Server: nginx/1.4.6 (Ubuntu)
Date: Tue, 21 Feb 2017 21:36:04 GMT
Content-Type: text/html
Content-Length: 193
Connection: keep-alive
Location: https://www.youtube.com/watch?v=QH2-TGulwu4

sylvain@ubuntu$
```

Repo:

- GitHub repository: alx-system_engineering-devops
- Directory: 0x0C-web_server
- File: 3-redirection

☐ Done?

Check your code

Ask for a new correction

> Get a sandbox

QA Review

4. Not found page 404

mandatory

Score: 0.0% (Checks completed: 0.0%)

Configure your Nginx server to have a custom 404 page that contains the string Ceci n'est pas une page .

Requirements:

- The page must return an HTTP 404 error code
- The page must contain the string Ceci n'est pas une page
- Using what you did with 3-redirection , write 4-not_found_page_404 so that it configures a brand new Ubuntu machine to the requirements asked in this task

Example:



```
sylvain@ubuntu$ curl -sI 34.198.248.145/xyz
HTTP/1.1 404 Not Found

Server: nginx/1.4.6 (Ubuntu)
Date: Tue, 21 Feb 2017 21:46:43 GMT
Content-Type: text/html
Content-Length: 26
Connection: keep-alive
ETag: "58acb50e-1a"

sylvain@ubuntu$ curl 34.198.248.145/xyzfoo
Ceci n'est pas une page

sylvain@ubuntu$
```


Repo:

- GitHub repository: alx-system_engineering-devops
- Directory: 0x0C-web_server
- File: 4-not_found_page_404

☐ Done?

Check your code

Ask for a new correction

 Get a sandbox

QA Review

5. Install Nginx web server (w/ Puppet)

#advanced

Score: 0.0% (Checks completed: 0.0%)

Time to practice configuring your server with Puppet! Just as you did before, we'd like you to install and configure an Nginx server using Puppet instead of Bash. To save time and effort, you should also include resources in your manifest to perform a 301 redirect when querying /redirect_me.

Requirements:

- Nginx should be listening on port 80
- When querying Nginx at its root / with a GET request (requesting a page) using `curl`, it must return a page that contains the string `Hello World!`
- The redirection must be a "301 Moved Permanently"
- Your answer file should be a Puppet manifest containing commands to automatically configure an Ubuntu machine to respect above requirements

Repo:

- GitHub repository: alx-system_engineering-devops
- Directory: 0x0C-web_server
- File: 7-puppet_install_nginx_web_server.pp

☐ Done?

Check your code

Ask for a new correction

QA Review



(/)

Copyright © 2024 ALX, All rights reserved.

