**1. VM installation**

* Скачать образ отсюда [Debian](https://cdimage.debian.org/debian-cd/current/i386/iso-cd/) (debian-9.9.0-i386-netinst.iso)
* Разбить на три диска-1) 4.5 первичный (/),2) 1.2 логический(раздел подкачки), 3) 2.9 логический (домашняя).
* При установке выбрать только последний и предпоследний пункт при установке ПО
* Посмотреть итог раздела диска lsblk

**2. Check for updates and configure sudo**

Заходим под root:

$ su

$ apt install sudo vim -y

* редактируешь файл vim /etc/sudoers

# User privilege specification

root ALL=(ALL:ALL) ALL

user ALL=(ALL:ALL) ALL

Добаляем пользователя в группу root:

$ adduser user sudo либо su – user и потом перезагрузить sudo reboot

Обновляем пакеты:

$ sudo apt update(apt-get update)

$ sudo apt upgrade(apt-get upgrade)

* устанавливаем необходимые программы:

$ sudo apt install ufw portsentry fail2ban apache2 net-tools mailutils -y

* проверяем нет ли запрещенных пакетов:

$ dpkg --get-selections | grep -iE "Docker|Vagrant|Traefik" || echo "I have not found any restricted packages"

**3. Set static IP**

* Отредактируй файл sudo vim /etc/network/interfaces:

# This file describes the network interfaces available on your system

# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/\*

# The loopback network interface

auto lo

iface lo inet loopback

# The primary network interface

allow-hotplug enp0s3

iface enp0s3 inet static – интерефейс enp0s3 находиться в диапазоне адресов IPv4(inet) со статическим IP

address 10.0.2.15 – IP адрес сетевой карты

netmask 30 – маска подсети

gateway 10.0.2.2 – адрес шлюза по умолчанию

* Теперь перезапускаем одной из следующих команд:

$ sudo /etc/init.d/networking restart

$ sudo service networking restart

* Поднимаем командой:

sudo ifup enp0s3

sudo ifconfig enp0s3 up

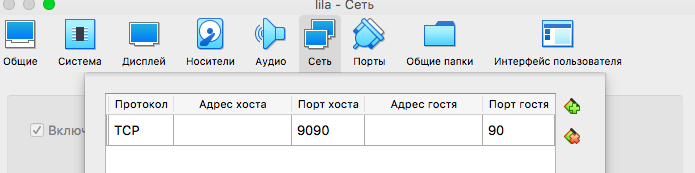
* Проверяем:

ping 8.8.8.8

выходим ctrl + c

**4. SSH setup**

* В настройках виртуалки делаешь следующее

****

* Редактируем файл на VM vim /etc/ssh/sshd\_config следующим образом

port 90

PasswordAuthentication yes

* Теперь перезапускаем командой:

$ sudo /etc/init.d/ssh restart

$ sudo service restart

* Подключаемся:

$ ssh user@127.0.0.1 -p 9090

* В терминале на Mac генирируем ключи:

ssh-keygen -t rsa

* Отправляем их на машину

ssh-copy-id -i ~/.ssh/id\_rsa.pub user@127.0.0.1 -p 9090

* Проверяем следующими командами:

cd /home/lfarlen/.ssh

cat authorized\_keys

* Опять редактируем файл на VM vim /etc/ssh/sshd\_config:

PermitRootLogin no

PubkeyAuthentication yes

#PasswordAuthentification no

* Перезапускаем командой:

$ sudo /etc/init.d/ssh restart

**5. Firewall**

* Настраиваем ufw:

$ sudo ufw status – проверяем статус(по умолчанию отключен)

$ sudo ufw default deny incoming – разрешает все исходящие соединения

$ sudo ufw default allow outgoing – запрещает все входящие

$ sudo ufw allow 90/tcp – ssh || sudo ufw allow ssh

$ sudo ufw allow 80/tcp – http || – незашифрованные соединения sudo ufw allow http

$ sudo ufw allow 443 – https || зашифрованные соединения sudo ufw allow https

$ sudo ufw enable - включаешь

$ sudo ufw reload – перезагружаешь и проверяешь статус

**6. DOS (Denial Of Service Attack) protection**

* Создаем локальную копию файла и работаем уже только с ним:

$ sudo cp /etc/fail2ban/jail.conf /etc/fail2ban/jail.local

* Редактируем файл следующим образом sudo vim /etc/fail2ban/jail.local:

[sshd]

enabled = true

port = ssh

logpath = /var/log/auth.log

backend = %(sshd\_backend)s

maxretry = 3

findtime = 21

bantime = 120

ignoreip = 10.0.2.2 –игнорирует этот адрес при защите

Дополнительно для защиты SSH активируйте следующую секцию:

[sshd-ddos]

enabled = true

port = ssh

logpath = /var/log/auth.log

backend = %(sshd\_backend)s

maxretry = 3

findtime = 21

bantime = 120

Для защиты веб-сервера Apache можно использовать следующие настройки Fail2ban:

[apache-auth]

enabled = true

port = http,https

logpath = %(apache\_error\_log)s

[apache-badbots]

enabled = true

port = http,https

logpath = %(apache\_access\_log)s

bantime = 172800

maxretry = 1

[apache-noscript]

enabled = true

port = http,https

logpath = %(apache\_error\_log)s

[apache-overflows]

enabled = true

port = http,https

logpath = %(apache\_error\_log)s

maxretry = 2

Значения: **logpath** — полный путь к файлу, в который будет записываться информация о попытках получения доступа. **findtime** — время в секундах, в течение которого наблюдается подозрительная активность; **maxretry** — разрешенное количество повторных попыток подключения к серверу; **bantime** — промежуток времени, в течение которого попавший в черный список IP будет оставаться заблокированным.

* Теперь перезапускаем командой: $ sudo /etc/init.d/fail2ban restart
* \**первое* смотрит состояние, \**второе* для разблокировки sshd, \**третье* для разблокировки apache

sudo fail2ban-client status

sudo fail2ban-client set sshd unbanip 10.0.2.2

sudo fail2ban-client set apache-auth unbanip 10.0.2.2

sudo fail2ban-client set apache-badbots unbanip 10.0.2.2

sudo fail2ban-client set apache-noscript unbanip 10.0.2.2

sudo fail2ban-client set apache-overflows unbanip 10.0.2.2

Почитать - <https://www.dmosk.ru/instruktions.php?object=fail2ban>

<https://vps.ua/wiki/configuring-fail2ban/>

**7. Protection port scans**

* Редактируем файл следующим образом sudo vim /etc/default/portsentry

TCP\_MODE="atcp"

UDP\_MODE="audp"

* Редактируем файл следующим образом sudo vim /etc/portsentry/portsentry.conf

BLOCK\_UDP="1"

BLOCK\_TCP="1"

# iptables support for Linux

KILL\_ROUTE="/sbin/iptables -I INPUT -s $TARGET$ -j DROP"

#KILL\_HOSTS\_DENY="ALL: $TARGET$ : DENY"

* Перезагружаем командой: sudo /etc/init.d/portsentry restart

**8. Stop the services**

* Смотрим наши сервисы: ls -l /etc/init.d
* Активные сервисы: sudo service --status-all
* Более подробно:sudo systemctl --full --type service

Отключаешь командой - sudo service service\_name stop

+ отключаешь автохагрузку

sudo systemctl disable service\_name

у себя я отключил следующее:

sudo service exim4 stop - агента пересылки почтовых сообщений Exim4

для полного удаления можно юзать sudo apt-get --purge autoremove *package\_name* autoremove для удаления зависимых файлов --purge для удаления конфигурационных файлов

запустить службу sudo systemctl start *service\_name*

**9. Update Packages**

* Создаем файл для логов: sudo vim /var/log/update\_script.log
* Создаем скрипт: sudo vim /root/update\_script.sh:

#! / bin / bash

apt update -y >> /var/log/update\_script.log

apt upgrade -y >> /var/log/update\_script.log

* двем права sudo chmod 755 /root/update\_script.sh
* Делаем root пользователя владельцем, чтобы все совсем автоматически делалось (это для подстраховки, по идеи владельцем так и так является root) sudo chown root /root/update\_script.sh
* Автоматизируем через sudo crontab -e:

0 4 \* \* 1 root /root/update\_script.sh

@reboot root /root/update\_script.sh

**10. Monitor Crontab Changes**

* Создаем скрипт sudo vim /root/crontab\_monitor.sh:

#!/bin/bash

DIFF=$(diff /etc/crontab\_save /etc/crontab)

cat /etc/crontab > /etc/crontab\_save

if [ "$DIFF" != ""] then

echo "crontab check: changed, notifying admin."

sendmail root@127.0.0.1 < ~/mail.txt

else

echo "crontab check: unchanged."

fi

* Меняем права sudo chmod 755 /root/crontab\_monitor.sh
* Делаем root пользователя владельцем (если он не владелец)
* Автоматизируем через sudo crontab -e:

0 0 \* \* \* root /root/scripts/crontab\_monitor.sh

Создаем свой sudo vim mail.txt, например Fail /etc/crontab changed. Go to u VM and see.

* Включаешь крон sudo systemctl enable cron
* \*Изменить время date --set hh:mm \*\*Если начнет выебываться, ответ тут https://mycyberuniverse.com/ru/error/how-to-fix-setting-locale-failed.html или тут https://askubuntu.com/questions/162391/how-do-i-fix-my-locale-issue

**11. Web Part**

а) Написать сайт index.html б) архивировать файлы сайта t ar -cf site.tar \*

в) скопировать файл с мака на ВМ

scp -P 9090 /Users/lfarlen/Desktop/mysite/site.tar lfarlen@127.0.0.1:~/

г) переместить из домашнего каталога в /var/www/html

sudo mv site.tar /var/www/html д) разархивировать sudo tar -xvf site.tar е) дописываем 80 порт для http (Host Port -8080, Guest Port -80)

ж\*) подключаемся проверить. В браузере на маке можно прописать один из следующих вариантов: https://oa-n6.21-school.ru:8080/ *(* *зависит от места где сидишь)*http:// 127.0.0.1:8080/ \* http:// 0.0.0.0:8080/

**12. Configure SSL**

* Генерируем самоподписанный ключ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/apache-selfsigned.key -out /etc/ssl/certs/apache-selfsigned.crt
* Cоздаем файл sudo vim /etc/apache2/ssl-params.conf:

SSLCipherSuite EECDH+AESGCM:EDH+AESGCM:AES256+EECDH:AES256+EDH

SSLProtocol All -SSLv2 -SSLv3

SSLHonorCipherOrder On

Header always set Strict-Transport-Security "max-age=63072000; includeSubdomains; preload"

Header always set X-Frame-Options DENY

Header always set X-Content-Type-Options nosniff

# Requires Apache >= 2.4

SSLCompression off

SSLSessionTickets Off

SSLUseStapling on

SSLStaplingCache "shmcb:logs/stapling-cache(150000)"

SSLOpenSSLConfCmd DHParameters "/etc/ssl/certs/dhparam.pem"

* В файле sudo vim /etc/apache2/sites-available/default-ssl.conf редактируем:

ServerAdmin cstill@student.21-school.ru

ServerName 127.0.0.1

SSLCertificateFile /etc/ssl/certs/apache-selfsigned.crt

SSLCertificateKeyFile /etc/ssl/private/apache-selfsigned.key

BrowserMatch "MSIE [2-6]" \

nokeepalive ssl-unclean-shutdown \

downgrade-1.0 force-response-1.0

* В файл sudo vim /etc/apache2/sites-available/000-default.conf дописываем:

Redirect "/" "https://127.0.0.1/"

* Добавляем в фаерволл https и перезапускаем:

sudo ufw allow 4443

sudo ufw reload

вывод команды sudo ufw status

vim

* Настраиваем ssl в apache:

sudo apache2ctl configtest

sudo a2enmod ssl

sudo a2enmod headers

sudo a2ensite default-ssl

sudo a2enconf ssl-params

sudo systemctl reload apache2

если выходит это- AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'ServerName' directive globally to suppress this message

Syntax OK

То вот держи - <https://askubuntu.com/questions/256013/apache-error-could-not-reliably-determine-the-servers-fully-qualified-domain-n>

Read - <https://www.digitalocean.com/community/tutorials/how-to-create-a-self-signed-ssl-certificate-for-apache-in-ubuntu-16-04>

**13. Part Deployment**

* Копируем репозиторий на Mac и VM
* Пушим сайт на Git
* На VM в файле sudo vim /etc/apache2/sites-available/default-ssl.conf меняешь var/www/html/ на /home/prawney/code21/web\_roger аналогично предыдущему пункту меняешь в файле sudo vim /etc/apache2/sites-available/000-default.conf
* В файле sudo vim /etc/apache2/apache2.conf меняешь следующие строки

<Directory />

Options FollowSymLinks

AllowOverride None

Require all granted

</Directory>

<Directory /usr/share>

AllowOverride None

Require all granted

</Directory>

<Directory /home/prawney/code21/web\_roger>

Options Indexes FollowSymLinks

AllowOverride None

Require all granted

</Directory>

\*/home/prawney/code21/web\_roger - это папка с сайтом склоненная с Git'a

* Теперь любые изменения которые мы запушили с Мака, можно сразу получить на VM и увидеть на сайте командой git pull.