

DI-192 nginx + сети (Шенгелай Всеволод)

Основное задание

ссылка для проверки работы

<https://109.71.246.234/>

Скачиваем архив с веб-сайтом (версию с уже созданной страницей 404)

```
curl -u my_own_username:my_own_password -o simplestyle_horizon_with_404_page.zip https://jira.web-bee.ru/secure/attachment/33620/simplestyle_horizon_with_404_page.zip
```

Директории, в которой обычно лежат статические веб-сайты виртуальной машине не оказалось, создадим её

```
sudo mkdir -p /var/www/html
```

Распакуем файл в эту директорию

```
sudo unzip simplestyle_horizon_with_404_page.zip -d /var/www/html/
```

Теперь установим nginx

```
sudo apt update  
sudo apt install nginx
```

Посмотрим, что лежит в конфигурационном файле nginx

```
cat /etc/nginx/nginx.conf
```

Создадим самоподписанный SSL сертификат

```
sudo mkdir /etc/nginx/ssl  
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/nginx/ssl/nginx-selfsigned.key -out /etc/nginx/ssl/nginx-selfsigned.crt
```

Создадим файл конфигурации /etc/nginx/sites-available/my_site

И запишем в него саму конфигурацию

```
server {
    listen 80;
    server_name 109.71.246.234;

    root /var/www/html/simplestyle_horizon_with_404_page;
    index index.html;

    location / {
        try_files $uri $uri/ =404;
    }

    error_page 404 /404.html;
    location = /404.html {
        internal;
    }

    listen 443 ssl; # managed by Certbot
    ssl_certificate /etc/nginx/ssl/nginx-selfsigned.crt;
    ssl_certificate_key /etc/nginx/ssl/nginx-selfsigned.key;
}

server {
    listen 80;
    server_name 109.71.246.234;
    return 301 https://$host$request_uri;
}
```

Создадим символическую ссылку на конфигурацию в sites-enabled

```
sudo ln -s /etc/nginx/sites-available/your_site /etc/nginx/sites-enabled/
```

Перезагрузим nginx, чтобы применить изменения

```
sudo systemctl reload nginx
```

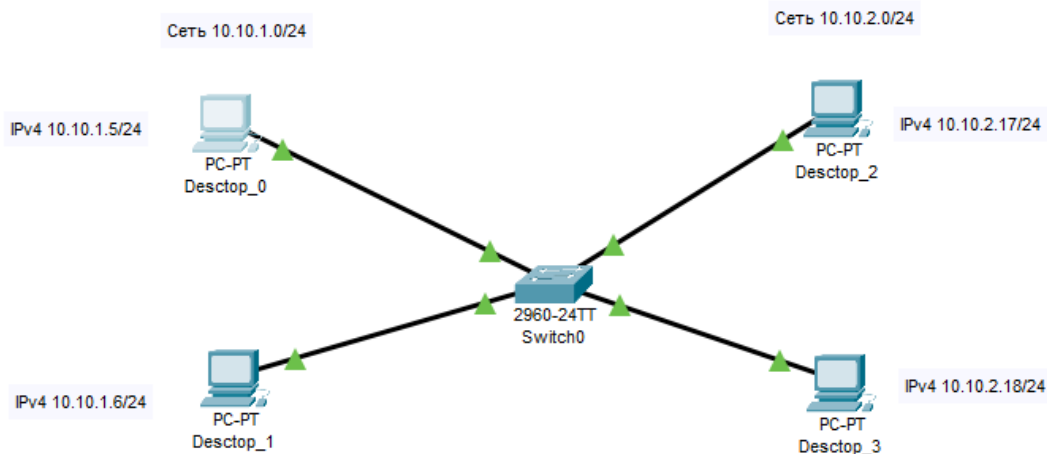
Дополнительные задания

Задание 1

- IANA выделила IP-адрес 0.0.0.0 для RFC 1122, который описывает этот адрес как "этот хост в этой сети". Этот адрес не может быть использован в качестве адреса назначения в IPv4, но может быть использован в качестве адреса источника в определенных обстоятельствах, таких как отправка начального пакета DHCPDISCOVER при использовании DHCP, когда хост еще не имеет назначенного IP-адреса.
- В таблицах маршрутизации: сетевой адрес 0.0.0.0 имеет маршрут по умолчанию в таблицах маршрутизации. Запись 0.0.0.0 с маской сети 0.0.0.0 в таблице маршрутизации обозначает, что этот маршрут подходит для всех IP-адресов.
- В Linux может быть ассоциирован с localhost

Задание 2.1 Вариант#1

Имеем сеть со следующей конфигурацией



Конфигурация сети

Каждый компьютер может пинговать каждый в пределах своей подсети, и не может пропинговать компьютер в другой подсети. Для осуществления пинга узлов в другой подсети нам обязательно будет нужен маршрутизатор (или коммутатор L3). Чтобы обойтись без маршрутизатора, мы можем расширить маску подсети на всех хостах, объединив их в более крупную подсеть

На всех компьютерах(хостах) "расширяем" маску подсети, убрав из неё два разряда (именно 2, т.к. если расширим на 1 разряд, мы не покроем все адреса обеих подсетей), 255.255.255.0 заменяем на 255.255.252.0.

Если бы нам не было важно статическое закрепление IP за хостами, можно было бы подключить DHCP-сервер, и там установить новую настройку на маску

```

C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>
C:\>ping 10.10.1.6 -n 2

Pinging 10.10.1.6 with 32 bytes of data:

Reply from 10.10.1.6: bytes=32 time<1ms TTL=128
Reply from 10.10.1.6: bytes=32 time<1ms TTL=128

Ping statistics for 10.10.1.6:
    Packets: Sent = 2, Received = 2, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>ping 10.10.2.17 -n 2

Pinging 10.10.2.17 with 32 bytes of data:

Reply from 10.10.2.17: bytes=32 time<1ms TTL=128
Reply from 10.10.2.17: bytes=32 time<1ms TTL=128

Ping statistics for 10.10.2.17:
    Packets: Sent = 2, Received = 2, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\>
  
```

Узел назначения пингуется

Задание 2.2 Вариант#2

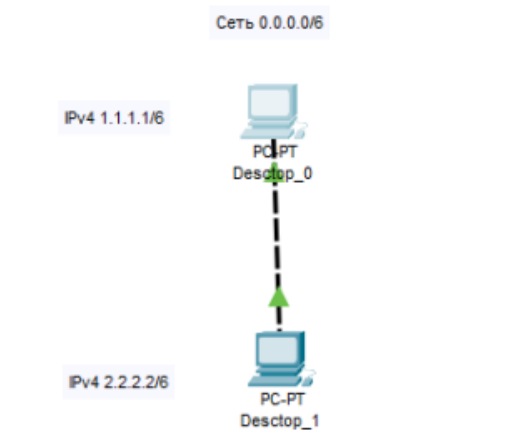
Мы можем организовать сетевую видимость для хостов из разных подсетей с помощью маршрутизатора

Задание 3

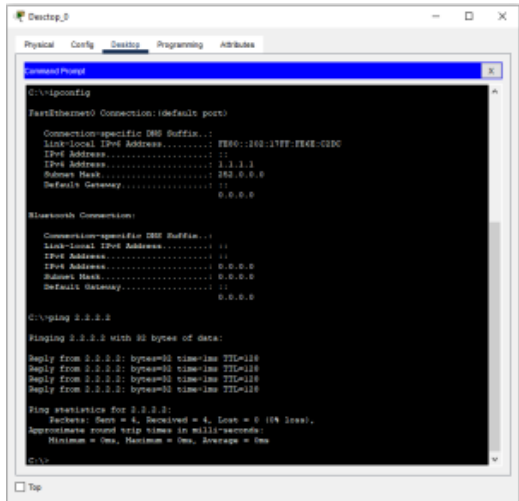
Смогут ли хосты с IP 1.1.1.1 и 2.2.2.2 пропинговать друг друга зависит от того, находятся ли они в одной подсети.

Для того, чтобы ping "прошёл", нужно выбрать правильную маску подсети. Выберем минимально достаточную маску подсети, чтобы объединить эти IP адреса в одну подсеть - 252.0.0.0.

Если при указании IP-адресов на сетевых адаптерах не указывать маску подсети, по крайней мере на Windows будет выставлена by default маска - 255.255.255.0, которая не сможет объединить наши IP-адреса в подсеть. И, так как ARP работает только в пределах одной подсети, устройство с IP 1.1.1.1 не сможет узнать MAC-адрес устройства с IP 2.2.2.2, и, соответственно, ICMP пакет не достигнет хоста назначения.



Конфигурация сети



Узел назначения пингуется

Задание 4.

```

C:\Users\Vsevolod>ping web-bee.ru

Обмен пакетами с web-bee.ru [158.160.138.74] с 32 байтами данных:
Ответ от 158.160.138.74: число байт=32 время=28мс TTL=50
Ответ от 158.160.138.74: число байт=32 время=28мс TTL=50
Ответ от 158.160.138.74: число байт=32 время=28мс TTL=50
Ответ от 158.160.138.74: число байт=32 время=28мс TTL=50

Статистика Ping для 158.160.138.74:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 28мсек, Максимальное = 28 мсек, Среднее = 28 мсек

C:\Users\Vsevolod>ping www.w3.org

Обмен пакетами с www.w3.org [188.114.99.224] с 32 байтами данных:
Ответ от 188.114.99.224: число байт=32 время=85мс TTL=51
Ответ от 188.114.99.224: число байт=32 время=86мс TTL=51
Ответ от 188.114.99.224: число байт=32 время=85мс TTL=51
Ответ от 188.114.99.224: число байт=32 время=85мс TTL=51

Статистика Ping для 188.114.99.224:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 85мсек, Максимальное = 86 мсек, Среднее = 85 мсек

```

Отправка 4-х ICMP пакетов на www.w3.org и на web-bee.ru

Что мы видим в выводе команды ping?

- Значение time показывает, за сколько времени данные доходят до www.w3.org и возвращаются обратно, в миллисекундах. В нашем случае среднее время ~ 85ms
- Значение ttl - это time to life вернувшегося пакета, по нему мы сможем узнать, через сколько узлов прошёл пакет
- Статистика по отправленным ICMP-запросам

Посмотрим на список узлов(хопов), через которые пройдут ICMP-пакеты, отправленные командой tracert.

```

C:\Users\Vsevolod>tracert www.w3.org

Трассировка маршрута к www.w3.org [188.114.98.233]
с максимальным числом прыжков 30:

 1  <1 мс    <1 мс    <1 мс    192.168.0.1
 2  <1 мс     5 мс     <1 мс    172.16.50.254
 3   1 мс     <1 мс     <1 мс    10.0.0.62
 4   8 мс     1 мс      1 мс    212.110.156.222
 5   9 мс     9 мс      9 мс    ae-3332-smfl-crumcom.rt.ru [95.167.18.57]
 6   *        *        *        Превышен интервал ожидания для запроса.
 7  40 мс    39 мс     39 мс    95.71.2.226
 8  42 мс    39 мс     39 мс    172.68.8.49
 9  35 мс    35 мс     35 мс    188.114.98.233

Трассировка завершена.

```

Трассировка маршрута к www.w3.org

```
C:\Users\Vsevolod>tracert web-bee.ru
```

```
Трассировка маршрута к web-bee.ru [158.160.138.74]  
с максимальным числом прыжков 30:
```

1	<1 мс	<1 мс	<1 мс	192.168.0.1
2	<1 мс	<1 мс	<1 мс	172.16.50.254
3	<1 мс	<1 мс	<1 мс	10.0.0.62
4	1 ms	1 ms	1 ms	212.110.156.222
5	24 ms	24 ms	24 ms	10.0.11.2
6	24 ms	24 ms	24 ms	109.239.136.69
7	24 ms	24 ms	24 ms	169.254.0.1
8	*	*	*	Превышен интервал ожидания для запроса.
9	*	*	*	Превышен интервал ожидания для запроса.
10	*	*	*	Превышен интервал ожидания для запроса.
11	*	*	*	Превышен интервал ожидания для запроса.
12	*	*	*	Превышен интервал ожидания для запроса.
13	*	*	*	Превышен интервал ожидания для запроса.
14	28 ms	28 ms	28 ms	158.160.138.74

```
Трассировка завершена.
```

Трассировка маршрута к web-bee.ru