Изменена конфигурация канала между маршрутизаторами главного и дополнительного офиса. Команды, которые были использованы:

DCERouter> enable

DCERouter # conf t

DCERouter (config)# username MainOffice password 123

DCERouter (config)# interface serial 0/1/0

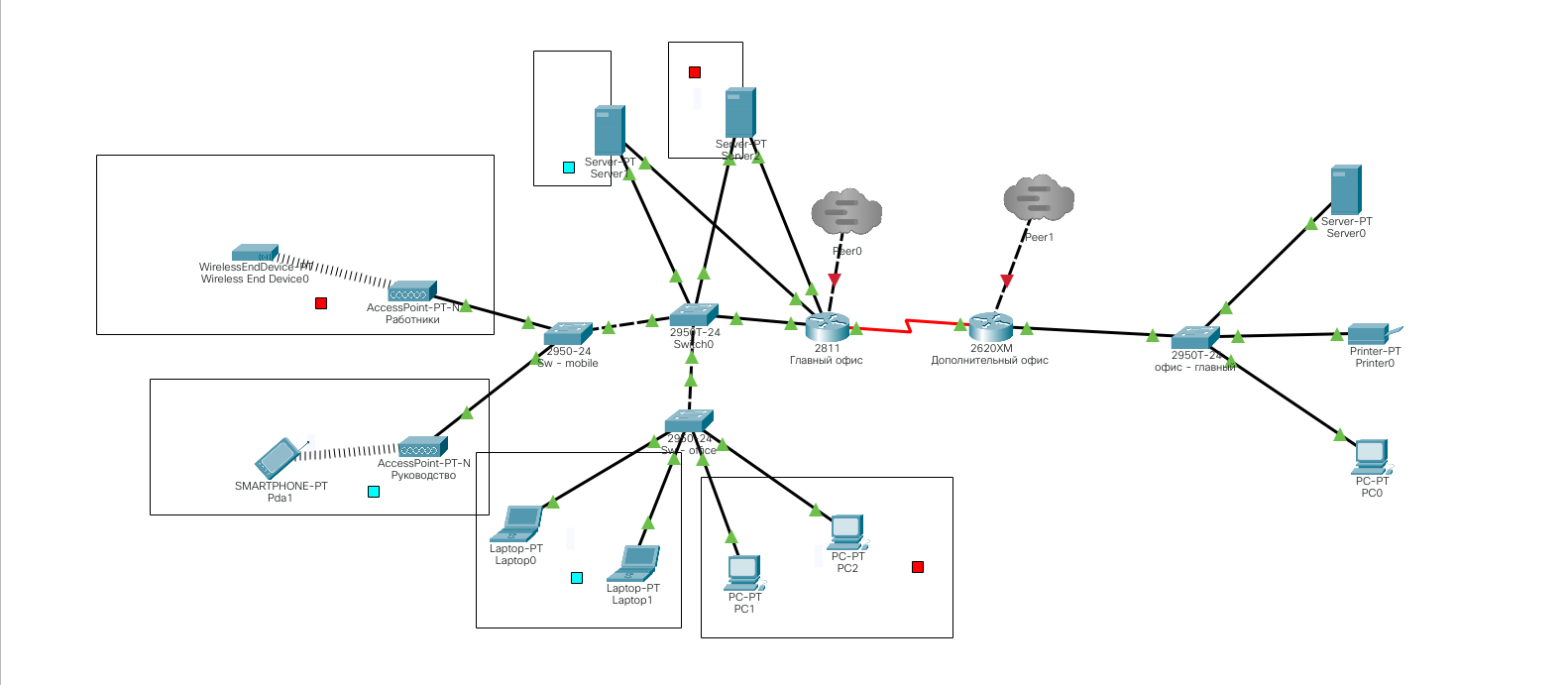
DCERouter (config-if)# encapsulation ppp

DCERouter (config-if)# ppp authentication chap

DCERouter (config-if)# clock rate 64000

Для MainOffice аналогично

1. Сеть главного офиса разделена на две виртуальные подсети



1. В рамках выделенного диапазона главного офиса были сформированы две подсети:

Главный офис: 10.15.30.1/19, 10.15.40.1/19, 10.15.192.2/19, 10.15.224.2

PC0: 10.15.40.2

PC1: 10.15.40.3

Laptop0: 10.15.30.2

Laptop1: 10.15.30.3

Smartphone0: 10.15.30.5

WED: 10.15.40.5

Server0: 10.15.30.4

Server1: 10.15.40.4

Адрес сети Vlan30: 10.15.30.0

Адрес сети Vlan40: 10.15.40.0

Для создания подсетей была произведена настройка коммуникаторов главного офиса. Для этого потребовались следующие команды:

SWD> enable

SWD# configure t

SWD(config)# int fa 0/1

Для типа доступа access (связь коммуникатора с конечным устройством):

SWD (config-if)# switchport mode access

SWD (config-if)# switchport access vlan 40

Для типа доступа trunk (связь коммуникатора с другими коммуникаторами и маршрутизаторами):

Switch(config-if)# switchport mode trunk

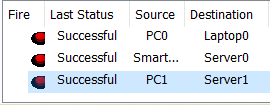
Switch(config-if)# switchport trunk allowed vlan 30-40

1. Созданы локальные интерфейсы (FastEthernet 0/0.30 - FastEthernet 0/0.30 и FastEthernet 0/0.40 - 10.15.40.1/19) в главном офисе на базе интерфейса FastEthernet 0/0

Для создания виртуального интерфейса потребовались следующие команды:

MainRouter (config)# int fa 0/0.30

MainRouter (config)# encapsulation dot1Q 30

Для второго интерфейса аналогично. Связь между устройствами виртуальных сетей настроена: 

1. Была настроена связь между серверами, подключенными в коммутатор маршрутизатора. Были выделены следующие адреса:  
   Server1 – 10.15.192.2/19

Server2– 10.17.224.2/19

Vlan30 – 10.17.192.1/19

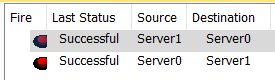
Vlan40 – 10.17.224.1/19

Для этой настройки потребовались следующие команды:

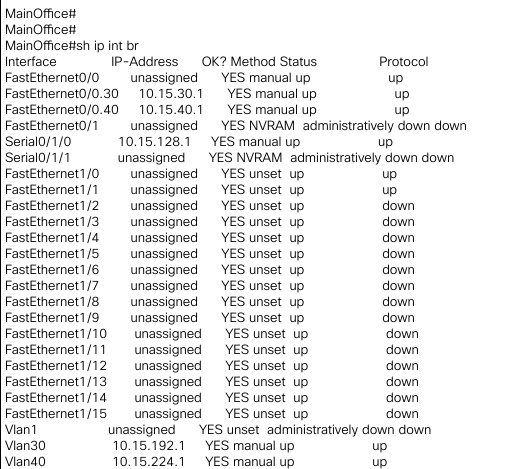
MainOffice (config)# int vlan30

MainOffice (config)# ip address 10.15.192.2 255.255.224.0

Для второго vlan интерфейса аналогично.  
MainOffice (config)# ip routing

Связь между серверами:  


Настройки роутера главного офиса:



1. Настроен размер кадра. Команда:

Router(config)# mtu 1290

1. Ответы на вопросы
2. Мы не изменяли настройки маршрутизатора доп. офиса, т.к. подсети остались прежними. Новые адреса устройств Главного офиса находятся в подсети 10.15.0.0/19, поэтому добавлять новые записи в статическую маршрутизацию не потребовалось.
3. Да, интерфейсы могут находится в одном vlan, т.к. они подключены в коммутатор, который может создавать VLAN, а сами сервера находятся в разных подсетях.
4. При передаче пакета в главный офис или другую подсеть сетевые узлы переупаковывают данные под необходимую длину кадра из-за того, что канальный уровень отбрасывается при получении пакетов. В случае передаче кадров по сети доп. офиса размер кадров одинаков.
5. Код программы:

#include <iostream>

#include <fstream>

using namespace std;

int check(unsigned int i);

int shift(unsigned int i, unsigned int bit);

int CRC32(unsigned int polinom, unsigned int registr, string file);

int main() {

setlocale(LC\_ALL, "Russian");

unsigned int init = 0xFFFFFFFF, polynom = 0x04C11DB7;

unsigned int result = CRC32(polynom, init, "D:/study/СетиЭВМ/lab4/avidreaders.ru\_\_voyna-i-mir-tom-1.txt");

cout << std::hex;

cout << "CRC32" << endl;

cout << "Полином = " << polynom << endl;

cout << "Чек сумма = " << result << endl;

return 0;

}

int check(unsigned int i) {

return ((i & (1 << 31)) ? 1 : 0);

}

int shift(unsigned int i, unsigned int bit) {

return i = i << 1 | bit;

}

int CRC32(unsigned int polinom, unsigned int registr, string file) {

char ch;

ifstream stream;

stream.open(file, ifstream::in);

if (!stream.is\_open()) {

return -1;

}

while (stream.get(ch)) {

if (stream.eof()) exit;

for (int i = 7; i >= 0; i--) {

if (check(registr)) {

registr = shift(registr, ch & (1 << i) ? 1 : 0);

registr ^= polinom;

}

else {

registr = shift(registr, ch & (1 << i) ? 1 : 0);

}

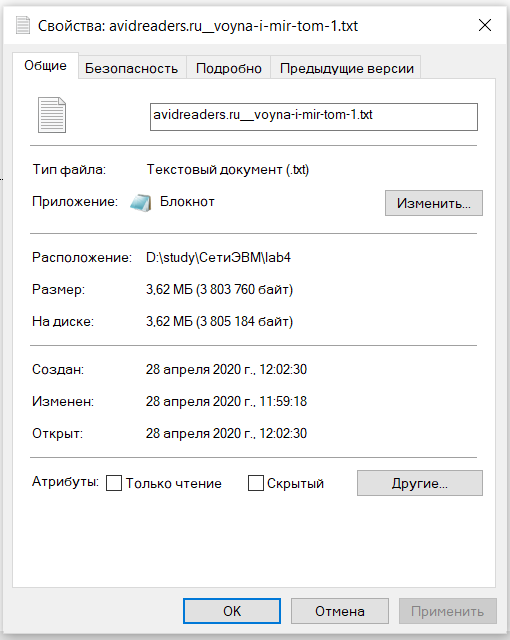
}

}

return registr;

}

Исходный файл:



Результат работы программы:

