

# Университет ИТМО

## Лабораторная работа №1 «WiFi»

*по дисциплине: Беспроводные сети*

Выполнили: Соболев Иван, Верещагин Егор, Р34312

Преподаватель: Оголюк Александр Александрович

Санкт-Петербург  
2024

# 1. Настройка ftp-сервера:

На компьютере с ОС Windows был настроен FTP-сервер FileZilla:

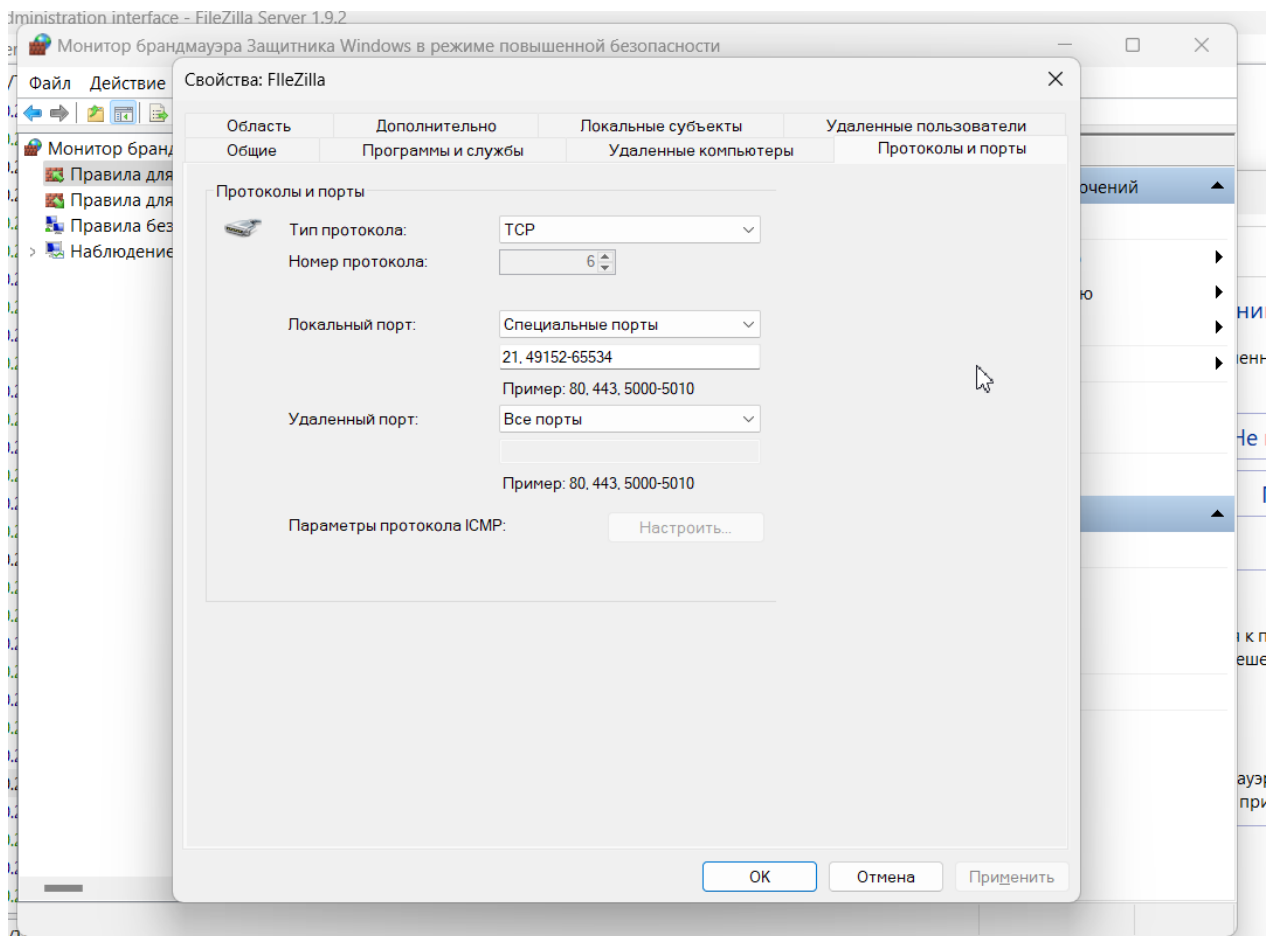
Administration interface - FileZilla Server 1.9.2

Server Window Help

Date/Time	Info	Type	Message
08.10.2024 22:51:36	FTP Session 27 192.168.0.13 isobolev	Command	QUIT
08.10.2024 22:51:36	FTP Session 27 192.168.0.13 isobolev	Response	221 Goodbye.
08.10.2024 22:51:36	FTP Server	Status	Session 27 ended gracefully.
08.10.2024 22:53:06	FTP Server	Status	Session 36 ended gracefully.
08.10.2024 22:56:33	FTP Session 37 192.168.0.13	Response	220-FileZilla Server 1.9.2
08.10.2024 22:56:33	FTP Session 37 192.168.0.13	Response	220 Please visit https://filezilla-project.org/
08.10.2024 22:56:33	FTP Session 37 192.168.0.13	Command	USER Asus
08.10.2024 22:56:33	FTP Session 37 192.168.0.13	Response	331 Please, specify the password.
08.10.2024 22:56:33	FTP Session 37 192.168.0.13	Command	PASS ****
08.10.2024 22:56:33	FTP Session 37 192.168.0.13	Response	230 Login successful.
08.10.2024 22:56:33	FTP Session 37 192.168.0.13 Asus	Command	TYPE I
08.10.2024 22:56:33	FTP Session 37 192.168.0.13 Asus	Response	200 Type set to I
08.10.2024 22:56:33	FTP Session 37 192.168.0.13 Asus	Command	PASV
08.10.2024 22:56:33	FTP Session 37 192.168.0.13 Asus	Response	227 Entering Passive Mode (192,168,0,8,209,145)
08.10.2024 22:56:33	FTP Session 37 192.168.0.13 Asus	Command	STOR large_file.bin
08.10.2024 22:56:33	FTP Session 37 192.168.0.13 Asus	Response	150 Starting data transfer.
08.10.2024 22:57:41	FTP Server	Status	Session 37 ended gracefully.
08.10.2024 22:57:56	FTP Session 38 192.168.0.13	Response	220-FileZilla Server 1.9.2
08.10.2024 22:57:56	FTP Session 38 192.168.0.13	Response	220 Please visit https://filezilla-project.org/
08.10.2024 22:57:56	FTP Session 38 192.168.0.13	Command	USER Asus
08.10.2024 22:57:56	FTP Session 38 192.168.0.13	Response	331 Please, specify the password.
08.10.2024 22:57:56	FTP Session 38 192.168.0.13	Command	PASS ****
08.10.2024 22:57:56	FTP Session 38 192.168.0.13	Response	230 Login successful.
08.10.2024 22:57:56	FTP Session 38 192.168.0.13 Asus	Command	TYPE I
08.10.2024 22:57:56	FTP Session 38 192.168.0.13 Asus	Response	200 Type set to I
08.10.2024 22:57:56	FTP Session 38 192.168.0.13 Asus	Command	PASV
08.10.2024 22:57:56	FTP Session 38 192.168.0.13 Asus	Response	227 Entering Passive Mode (192,168,0,8,217,69)
08.10.2024 22:57:56	FTP Session 38 192.168.0.13 Asus	Command	STOR large_file.bin
08.10.2024 22:57:56	FTP Session 38 192.168.0.13 Asus	Response	150 Starting data transfer.

Date/Time	Session ID	Protocol	Host	Username	Transfer
08.10.2024 22:57:56	38	FTP	192.168....	Asus	(1.08 GiB) 13.57 MiB/s (avg 7,61 MiB/s, 1.08 GiB) /large_file.bin

Также были настроены разрешения на брандмауэре:



## Создание программы для эмуляции ftp-клиента:

Для эмуляции ftp-клиента была написана программа:

```
import os
import time
from ftplib import FTP
import subprocess

address = "192.168.0.8"
command = ["ping", "-c", "10", address]

def ping():
    try:
        result = subprocess.run(command, stdout=subprocess.PIPE, stderr=subprocess.PIPE,
                                text=True)

        output = result.stdout
        errors = result.stderr

        print("Выходные данные команды ping:")
        print(output)

        if errors:
            print("Ошибки:")
            print(errors)

    except Exception as e:
        print(f"Произошла ошибка: {e}")

def create_large_file(filename, size_in_gb):
    """Создает файл заданного размера в Гб."""
    with open(filename, 'wb') as f:
        f.write(os.urandom(size_in_gb * 1024 * 1024))

def upload_file_ftp(ftp, filename):
    """Загружает файл на FTP-сервер и замеряет время передачи."""
    ftp.set_pasv(True)
    start_time = time.time()
    with open(filename, 'rb') as f:
        ftp.storbinary(f'STOR {os.path.basename(filename)}', f)
    end_time = time.time()
```

```

elapsed_time = end_time - start_time
file_size = os.path.getsize(filename)
speed_mbps = (file_size * 8) / (1024 * 1024 * elapsed_time) # Биты в Мегабиты/сек

return elapsed_time, speed_mbps

def main():
    filename = 'large_file_2.bin'
    size_in_mb = 300
    ftp_host = '192.168.0.8'
    ftp_user = 'Asus'
    ftp_pass = 'root'

    # print("Создание большого файла...")
    # create_large_file(filename, size_in_mb)
    # print(f"Файл {filename} успешно создан.")

    ftp = FTP(ftp_host)
    ftp.login(ftp_user, ftp_pass)

    print("Начало загрузки на FTP...")
    elapsed_time, speed_mbps = upload_file_ftp(ftp, filename)
    ftp.quit()

    print(f"Загрузка завершена.")
    print(f"Время передачи: {elapsed_time:.2f} секунд.")
    print(f"Скорость передачи: {speed_mbps:.2f} Мбит/с.")

    ping()

if __name__ == "__main__":
    main()

```

Она подключается к ftp-серверу и загружает на него файл размером 300Мб, при это измеряя скорость и время передачи, также выполняет команду ping 10 раз и собирает информацию.

Сравнение параметров:

На роутере удалось изменить стандарт WiFi на:

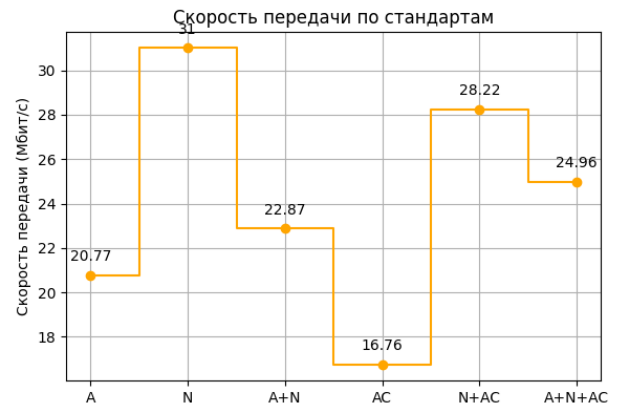
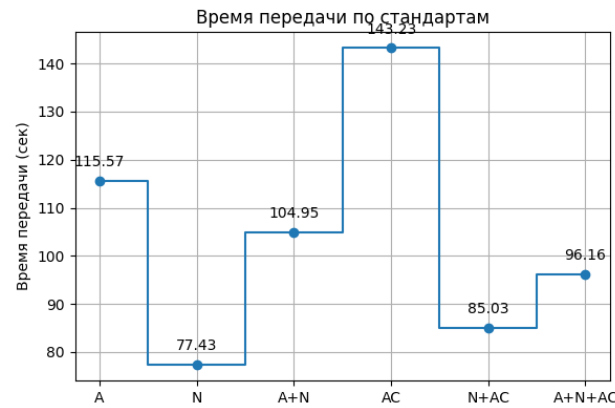
## WLAN Basic Settings

☐ Disable WLAN Interface

Band:

5 GHz (A)  
5 GHz (N)  
5 GHz (A+N)  
5 GHz (AC)  
5 GHz (N+AC)  
✓ 5 GHz (A+N+AC)

Стандарт	Время передачи(сек)	Скорость передачи (Мбит/с)	Среднее время отклика(мс)	Максимальное время отклика (мс)
A	115.57	20.77	17.170	29.644
N	77.43	31	9.542	17.783
A+N	104.95	22.87	23.407	35.724
AC	143.23	16.76	64.338	159.197
N+AC	85.03	28.22	18.020	30.948
A+N+AC	96.16	24.96	19.803	32.641



Можем видеть, что наилучшим по характеристикам стандартом оказался стандарт N.

Также на роутере можно изменить ширину канала:

Ширина канала:

Текущая ширина канала:

Auto  
☒ 20MHz  
 40MHz

Будем измерять различные характеристики от ширины канала на наиболее быстром стандарте – N.

Ширина канала (МГц)	Время передачи(сек)	Скорость передачи (Мбит/с)	Среднее время отклика(мс)	Максимальное время отклика (мс)
20	82.77	29	18.247	27.899
40	93.91	25.26	26.377	64.475



Можем видеть, что наилучшие параметры передачи достигаются при ширине канала 20 МГц. Чем больше ширина канала, тем больше пакетов данных может обрабатываться одновременно, из-за чего могут быть просадки скорости.

Также на роутере еще можно изменить мощность передатчика:

Мощность передатчика(mW):

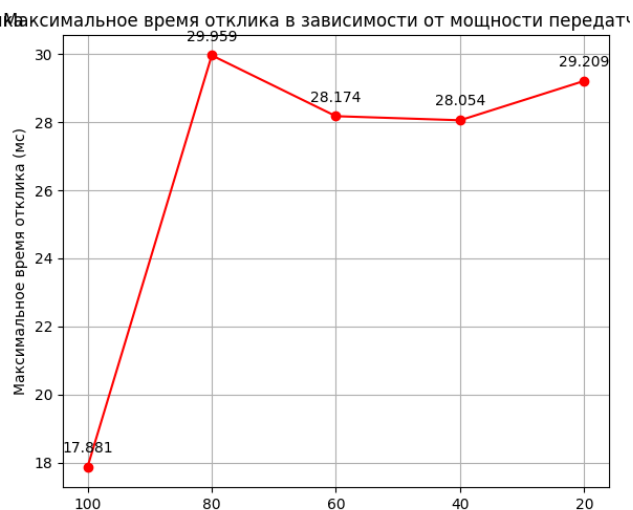
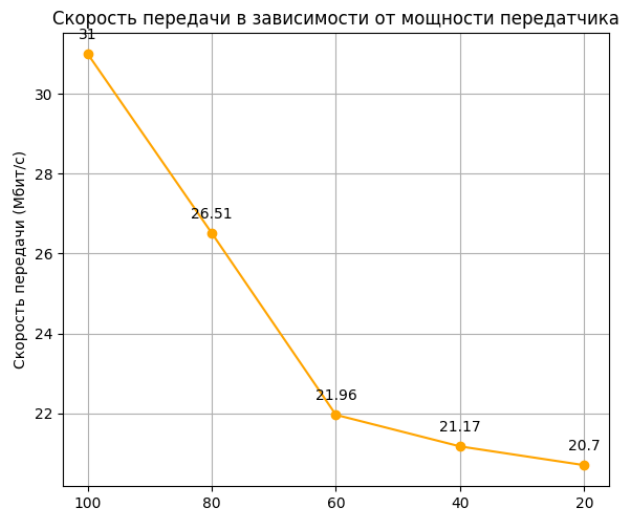
Максимальное количество клиентов:

☒ 100%  
 80%  
 60%  
 40%  
 20%

Мощность передатчика	Время передачи(сек)	Скорость передачи (Мбит/с)	Среднее время отклика(мс)	Максимальное время отклика (мс)
100	77.43	31	9.542	17.881
80	90.53	26.51	18.625	29.959

60	109.31	21.96	18.635	28.174
40	113.39	21.17	17.569	28.054
20	115.95	20.70	17.520	29.209

Чем больше мощность передатчика, тем быстрее скорость.



## Выводы

В рамках выполнения лабораторной работы мы подняли на ноутбуке FTP-сервер и подключились к нему с другого ноутбука по локальной сети. Далее с помощью скрипта измерили характеристики передачи данных при изменении параметров WI-FI. Для нашей локальной сети наиболее быстрым оказался стандарт N с шириной канала 20 МГц и максимальной мощностью передатчика.