



# FTPCHAT PROJECT

## FILE TRANSFER PROTOCOL CHAT PROJECT



### Features

Local and global mode . / Uses a low amount of electricity which is also ECO-Friendly. / Works with out internet in local mode which make it also more secure . Cheap to deploy (1900 EGP for around 50-100 users) 24 Layer of mono-alphabetic encryption. / Small and compact size.

### Data Analysis

FTPChat successfully relayed encrypted messages without using WebSockets. Relay speed remained consistent across both local routers and cloud-hosted FTP servers. The 24-layer encryption added security without increasing file size or causing latency. Routers like the ZTE ZXHN H188A can hold 50-100 users (According to the calculations). AI-based inspection tools were unable to decrypt the messages, confirming encryption strength. FTPChat is eco-friendly, since it runs on low-power devices such as routers.

### Applications

- Secure chat in schools and labs. Easy setup for private messaging without WebSockets.
- Works on old devices. Runs on routers, old PCs and low-power systems.
  - Company communication. Safe internal messaging between employees using FTP.
- Global and local use. Works with cloud servers or home routers, no special tools needed.

### Results

Messages sent and received successfully. Encryption verified across all tests. Protocol hosted on ZTE router and SFTPCloud.io. No tunneling tools or WebSockets libraries used. Stable performance across devices.

### Conclusion

FTP can be used as a secure relay for encrypted messaging. FTPChat proves that legacy protocols can be repurposed for modern secure communication. The system is WebSockets-free, stable, and deployable across various platforms.

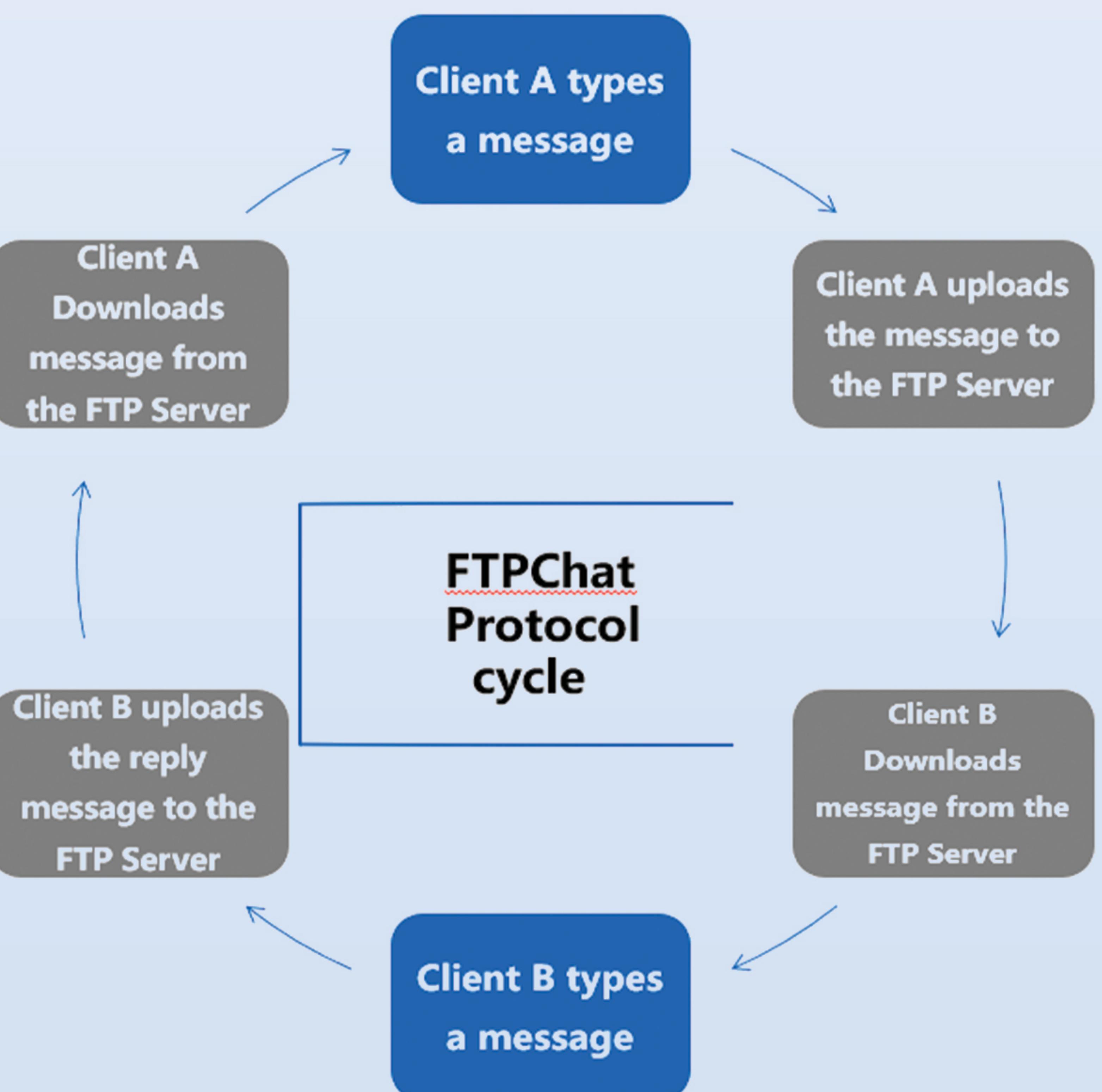
### Purpose

To build a secure messaging system without using WebSockets. FTPChat uses FTP as a relay layer and applies multi layer encryption to protect messages. This project's goal is to prove that legacy protocols can be repurposed for modern secure communication and decrease the reliance on WebSockets and repurpose old devices for another uses such as hosting FTP servers which reduces E-waste.

### Background Research

WebSockets often introduce instability and security risks. FTP is a legacy protocol designed for file transfer, but it can be repurposed as a message relay. Routers like ZTE ZXHN H188A and platforms like SFTPCloud.io support FTP hosting. Python provides libraries for encryption and file handling.

### Visualizing how it works



### Hypothesis

It's possible to build a stable, encrypted messaging protocol that works on legacy hardware and modern networks, without tunneling tools or WebSockets dependencies. Making developing chat apps easier for software developers to develop.

### Materials Used

Router with FTP support (ZTE ZXHN H188A)  
SFTPCloud.io website  
Python 3.x  
Inno Setup  
Code editor (VS Code)  
Internet connection for global testing  
32GB pen drive  
Total Price: 1900 EGP

### Procedure

Define the problem: decrease WebSockets dependency.  
Learn Python and FTP file handling.  
Write relay logic using FTP upload/download.  
Implement 24-layer mono-alphabetic encryption.  
Host protocol on router and SFTPCloud.io.  
Test message delivery and encryption integrity.