File Copy Protocol Design Specification
By Dylan Hoffmann & Lucas Campbell

- ❖ Network Protocol:
  - ➢ The network is composed of 3 types of packets:
    - ■ Directory Pilot Packets (Type D)
    - ■ File Pilot Packets (Type P)
    - ■ File Data Packets (Type F)
  - ➢ Packets are generally sent in the above order.
  - ➢ Pilot packets communicate metadata from the client to the server
  - ➢ Data packets transfer file data from the client to the server with minimal basic identifying information
- ❖ Packet Specifications:
  - ➢ Directory Pilot Packet (Type D)
    - ■ In the form: "D ####### T..."
    - ■ Where:
      - ● D is the packet type indicator for a Directory pilot packet (Always 1 byte)
      - ● # is the number of files in the directory (Up to 7 bytes = 9,999,999 files)
      - ● T... is a variable length field for the Target (up to 500 bytes long)
  - ➢ File Pilot Packet (Type P)
    - ■ In the form : "P ####### IIIIIII HHHHHHHHHHHHHHHHHHHH F....."
    - ■ Where:
      - ● F is the packet type indicator for a file Pilot packet (Always 1 byte)
      - ● # is the number of packets for the file == (file-size // 420) +1 (up to 7 bytes = 9,999,999 packets)
      - ● I is the file ID (up to 7 bytes = 9,999,999 files)
      - ● H is the SHA1 hash of the file (Always 20 bytes)
      - ● F... is a variable length field for the file name (up to 450 bytes long)
  - ➢ File Data Packet (Type F)
    - ■ In the form: "F ####### PPPPPPP D....."
    - ■ Where:
      - ● F is the packet type indicator for a File data packet (Always 1 byte)
      - ● # is the packets number of this packet (Up to 7 bytes = 9,999,999 packets)
      - ● P is the file ID (Up to 7 bytes = 9,999,999 files)
      - ● D... is a variable length field for the file data (up to 480 bytes long)
- ❖ Given the above specifications the constraints of this system become apparent:
  - ➢ Overflow can occur in directories containing more than 9,999,999 files, or files larger than 4.8 Gb.
    - ■ However, both of those overflows seem unlikely (though certainly not impossible) and this system can currently handle, assuming roughly uniform file sizes, almost 48 Petabytes of data within the existing confines of the protocol. (9,999,999 files x 9,999,999 packets x 480b)
  - ➢ Each File copy attempt requires at least 2 + ((file_size // 480) + 1) which, while not egregious, under sufficiently terrible network conditions the requirement for pilot packets may become an apparent drawback.

❖ The benefits of this system are that because the metadata is sent separately from the data there is more room in the packets for data, which will hopefully allow for fewer packets to need to be sent in total.