FTP: Cleartext Protocol Analysis Write Up

Cleartext Protocol Analysis

- It is important to create statistics and key results from the investigation process.

FTP (File Transfer Protocol)

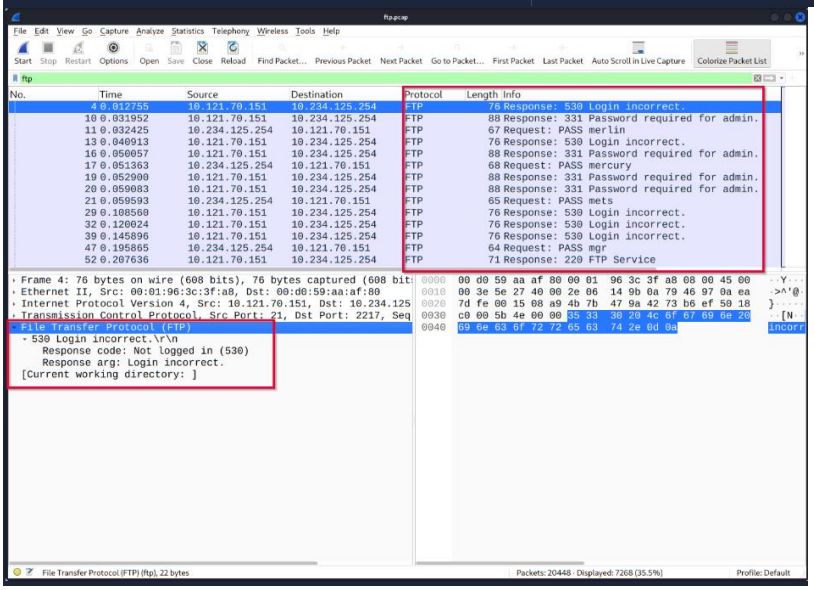
- Designed to transfer files
- This is a simplicity protocol and using this in an unsecure environment can cause security threats such as:
  a. MitM attacks
  b. Credential stealing and unauthorized access
  c. Phishing
  d. Malware planting
  e. Data exfiltration

Notes	Wireshark Filter
Global search	• `ftp`
"FTP" options for grabbing the low-hanging fruits: • **x1x series:** Information request responses. • **x2x series:** Connection messages. • **x3x series:** Authentication messages. **Note:** "200" means command successful.	---

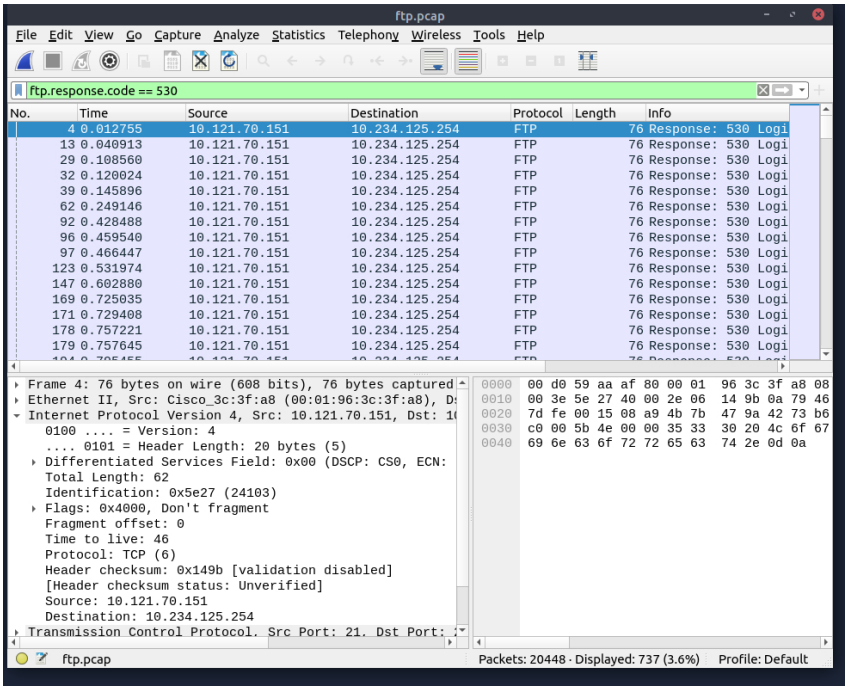
"x1x" series options for grabbing the low-hanging fruits: • **211:** System status. • **212:** Directory status. • **213:** File status	• `ftp.response.code == 211`
"x2x" series options for grabbing the low-hanging fruits: • **220:** Service ready. • **227:** Entering passive mode. • **228:** Long passive mode. • **229:** Extended passive mode.	• `ftp.response.code == 227`
"x3x" series options for grabbing the low-hanging fruits: • **230:** User login. • **231:** User logout. • **331:** Valid username. • **430:** Invalid username or password • **530:** No login, invalid password.	• `ftp.response.code == 230`
"FTP" commands for grabbing the low-hanging fruits: • **USER:** Username. • **PASS:** Password. • **CWD:** Current work directory. • **LIST:** List.	• `ftp.request.command == "USER"` • `ftp.request.command == "PASS"` • `ftp.request.arg == "password"`
Advanced usages examples for grabbing low-hanging fruits: • **Bruteforce signal:** List failed login attempts.	• `ftp.response.code == 530` • `(ftp.response.code == 530) and (ftp.response.arg contains "username")`

- **Bruteforce signal:** List target username.
- **Password spray signal:** List targets for a static password.

- `(ftp.request.command == "PASS" ) and (ftp.request.arg == "password")`

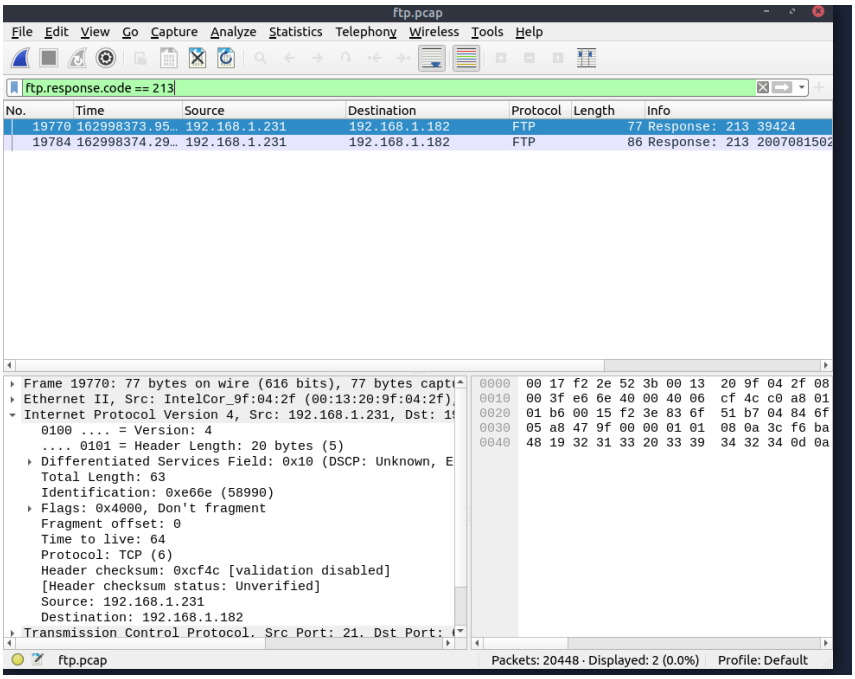


1. **How many incorrect login attempts are there?**
   - To answer this question, let's use the filter ftp.response.code == 530 to determine the number of failed attempts



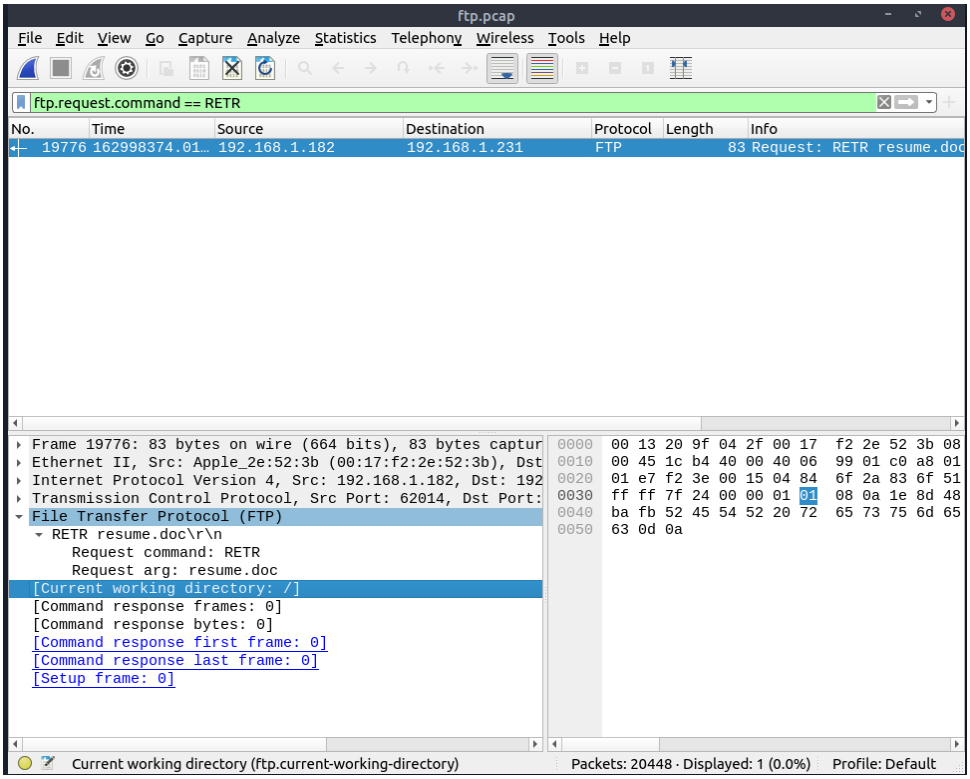
Answer: 737

2. **What is the size of the file accessed by the “ftp” account?**
   - We can use the filter ftp.response.code == 213 to filter out the size of the file access



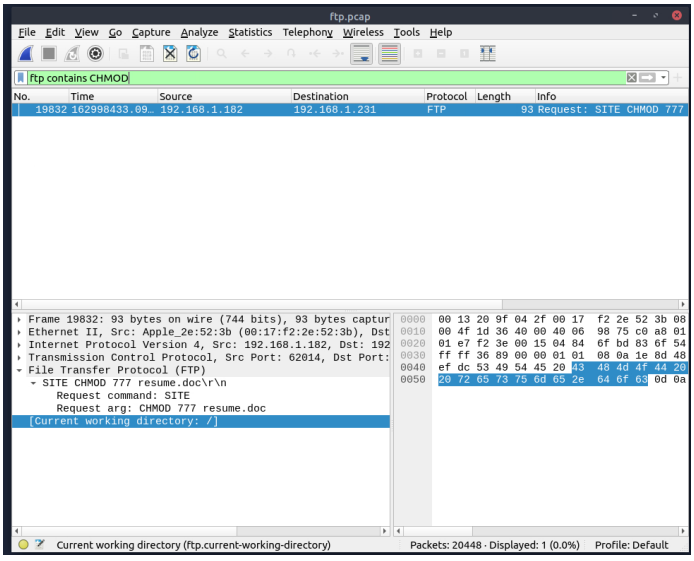
Answer: 39424

3. The adversary uploaded a document to the FTP server. What is the filename
   - Usually when a file is uploaded, we can filter using the “RETR”, meaning that a file or a document of some sort was retrieved from the user (in this case the adversary) and uploaded.
   - ftp.request.command == RETR



Answer: resume.doc

4. **The adversary tried to assign special flags to change the executing permissions of the uploaded file. What is the command used by the adversary?**
   - We can filter out a command that is used in a linux machine that changes permissions to a file; which is chmod
   - ftp contains CHMOD



Answer: CHMOD 777