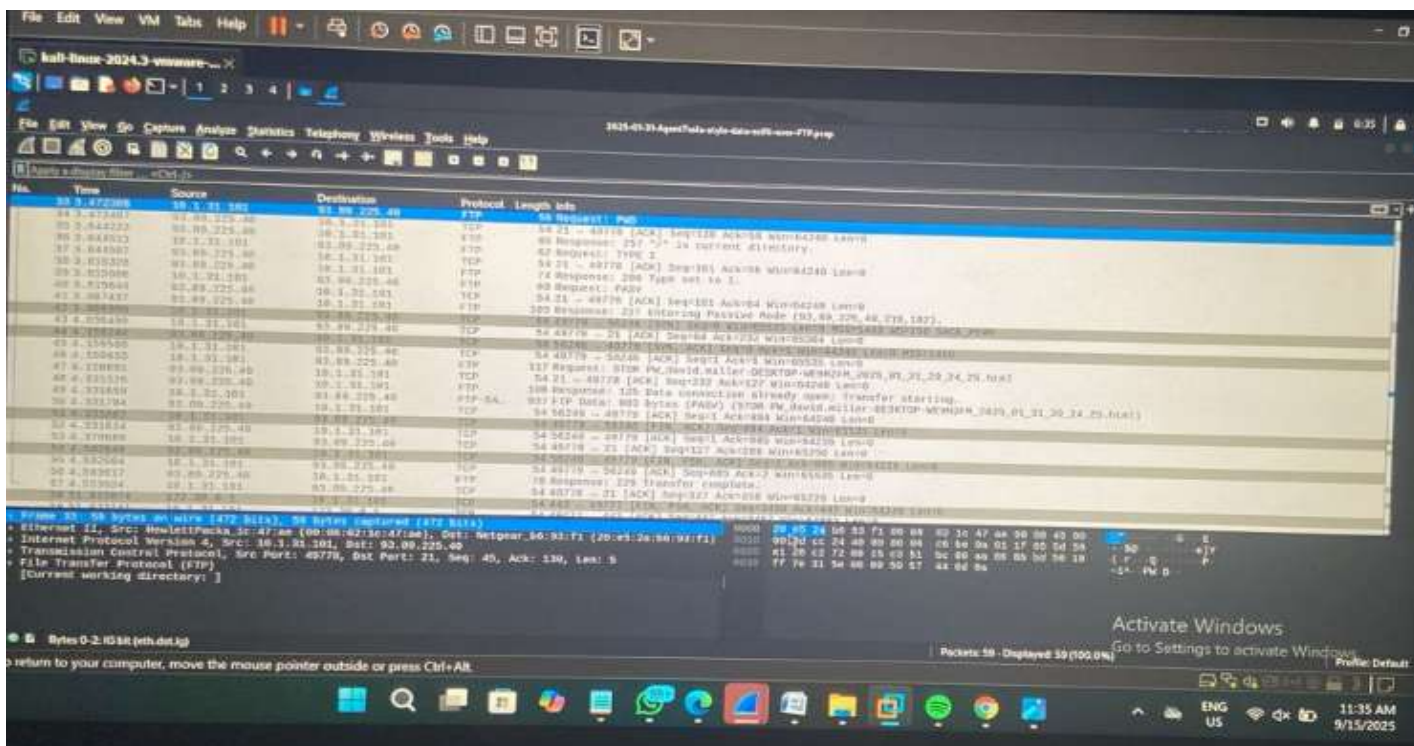


How I traced a malware infection from a client to an FTP exfiltration server using Wireshark.
An infected host checked in, authenticated over FTP with hardcoded credentials, uploaded a desktop snapshot, and the server returned 226 Transfer complete.

During the routine packet inspection I discovered an internal workstation (10.1.31.101) communicating with an external FTP server (93.89.225.40). The session contained plaintext credentials (USER/PASS), an OPTS UTF8 ON negotiation, and a STOR command that uploaded PW_daviv.miller-DESKTOP-WE9H2FM_2025-01_31_20_24_25.html. The server responded 226 Transfer complete, confirming successful upload. Banner text showed “Microsoft FTP Service” but the connection characteristics (random credentials, filename pattern, external IP reputation) indicate the endpoint functioned as the attacker’s exfiltration/C2 server (Agent Tesla-style). Evidence was preserved, and the incident was handled with containment, forensic collection, eradication, and recovery steps.

Here is how I uncovered the attack step by step

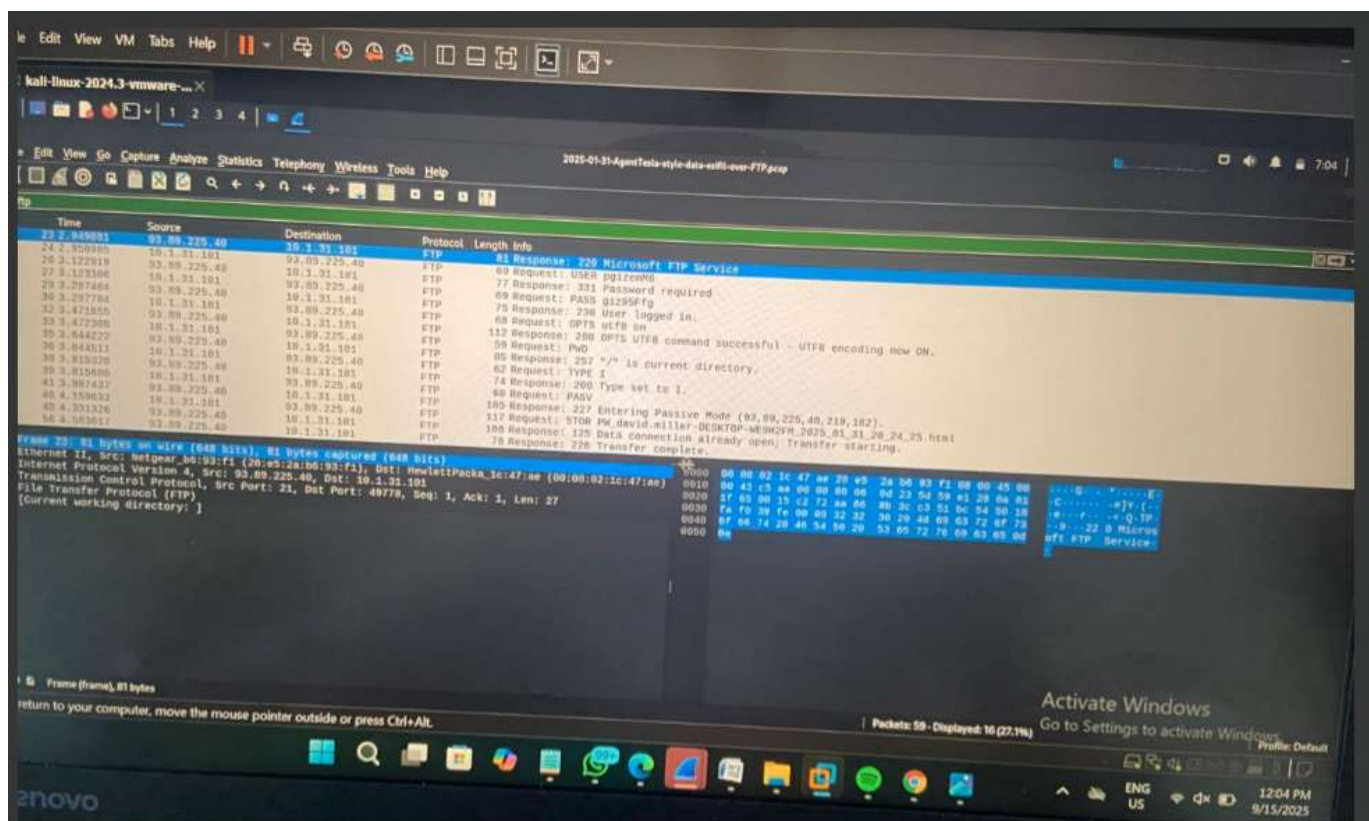




Step 1: Identifying suspicious FTP traffic

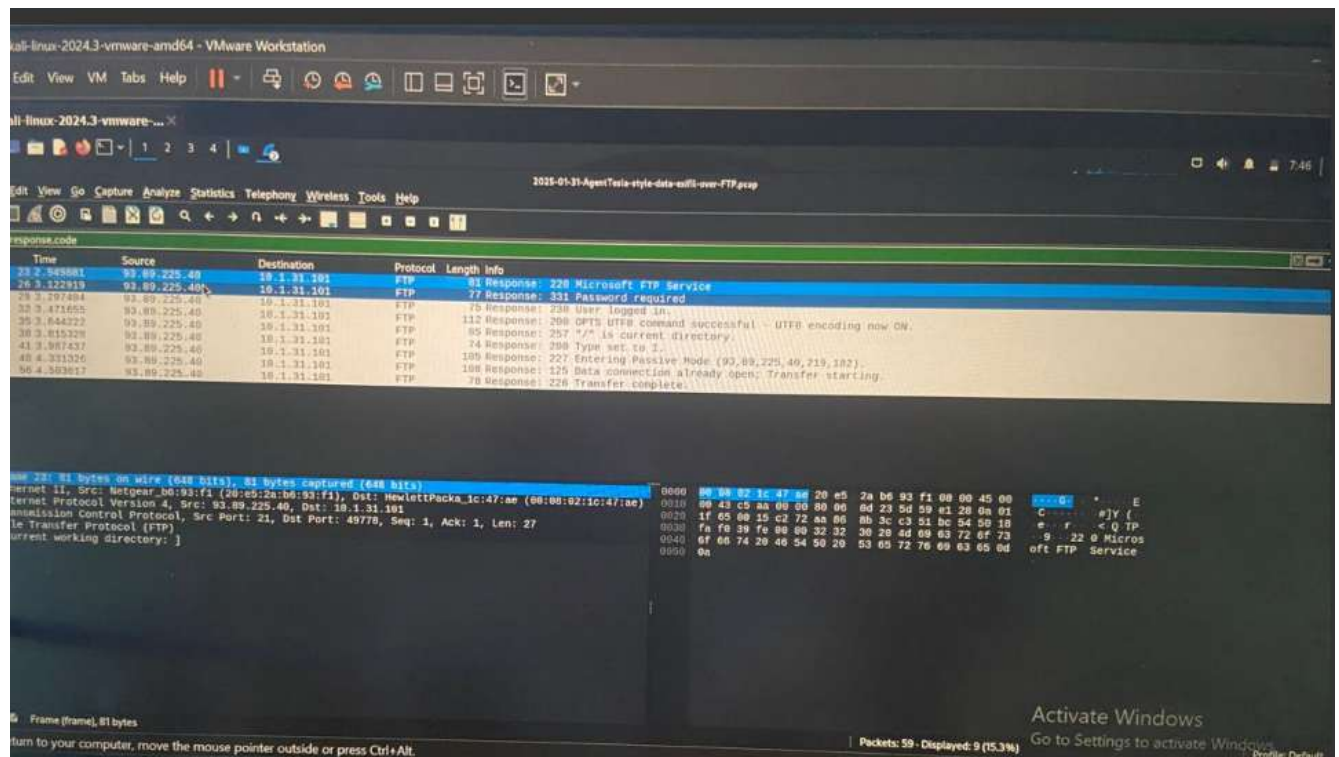
I started by filtering for FTP requests and found something odd:

- I found FTP control connection between 10.1.31.101 → 93.89.225.40
- This immediately narrowed the analysis to the control dialogue (USER/PASS/commands) and the subsequent data channel.



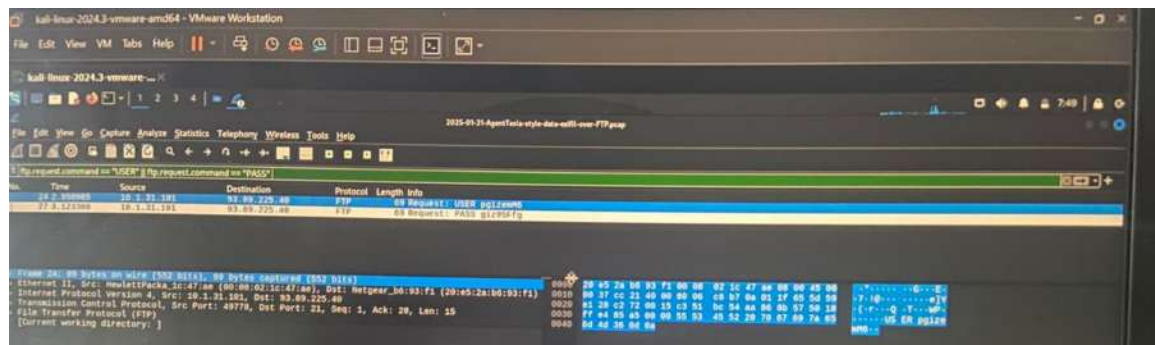
Step 2: I inspected the initial server reply and option negotiation statements inside the FTP control exchange

- I found out that The FTP control channel opened with a server banner: 220 Microsoft FTP Service
- The client 10.1.31.101 requested OPTS UTF8 ON and the server confirmed; OPTS UTF8 is a normal FTP negotiation to ensure correct filename encoding; it's context, not an indicator of compromise.



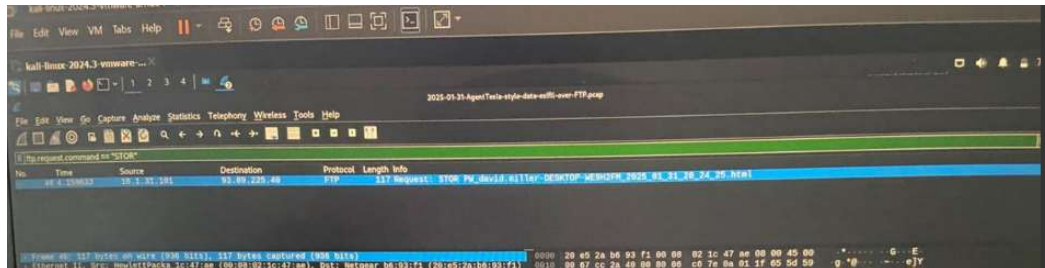
Step 3: Capture credentials

- I filtered the capture for authentication commands to see what credentials were used.
- I found out that the session contained plain text credentials sent over the control channel.



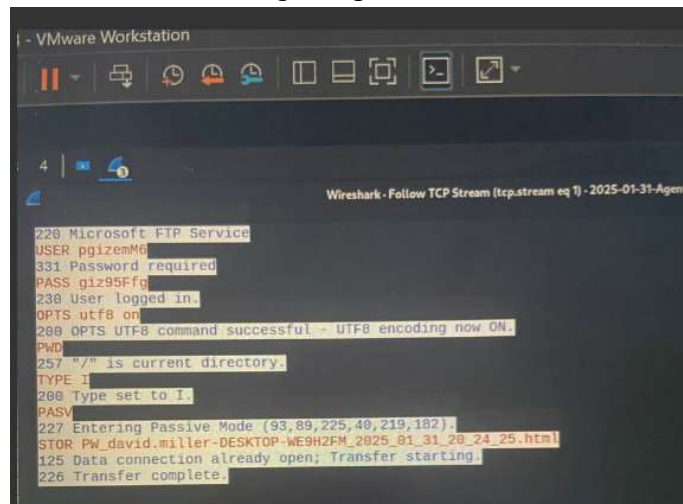
Step 4: Confirm exfiltration via STOR

- I searched for file transfer commands to confirm a file upload.
- I found out that the client issued a STOR command for a desktop-named file: **PW_daviv.miller-DESKTOP-WE9H2FM_2025-01_31_20_24_25.html**.
- Immediately following the STOR, FTP-DATA packets transferred the file bytes.
- The server returned 226 Transfer complete, confirming successful upload.

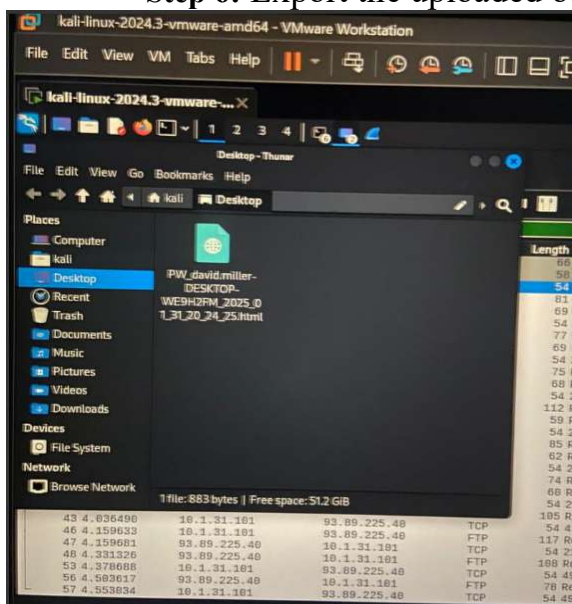


Step 5: Follow the TCP streams (control & data)

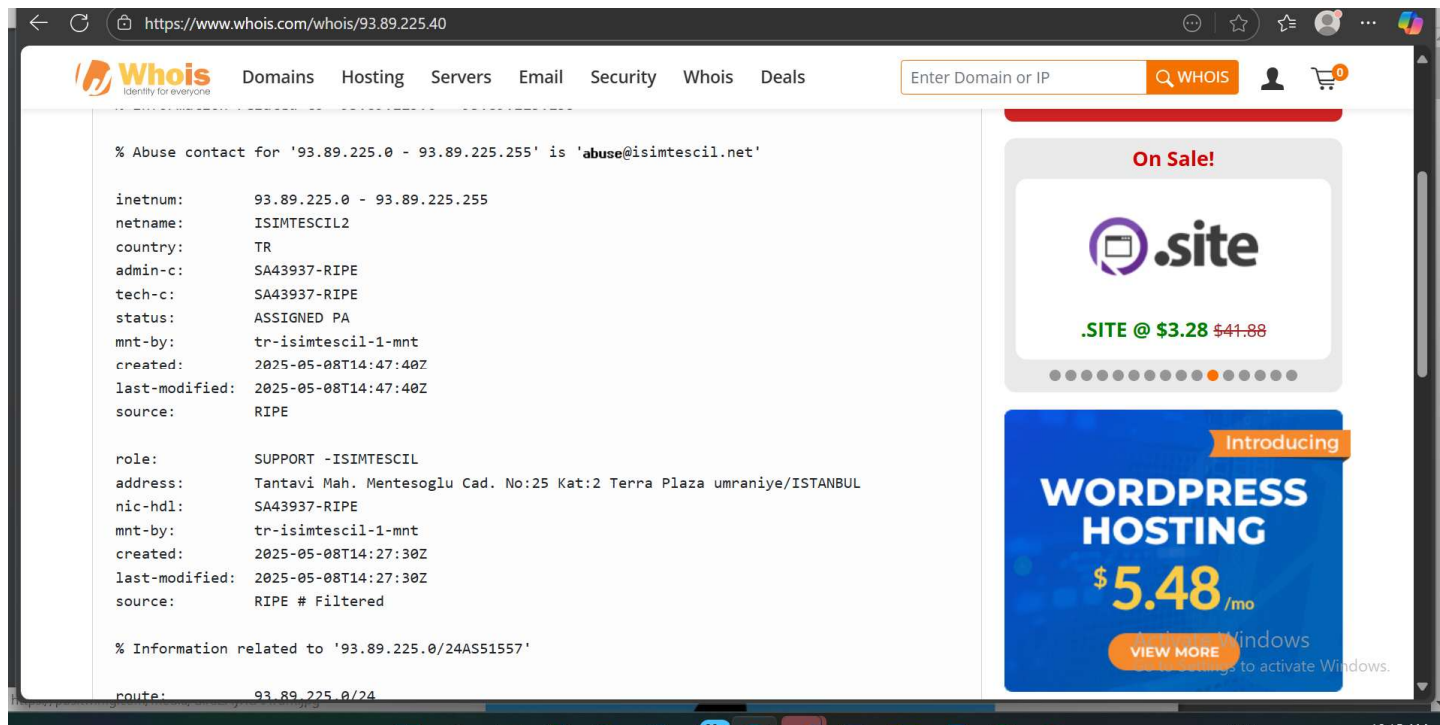
- I found the full control dialogue (login → OPTS → STOR → 226) in clear text (control) and the data transfer being completed.



Step 6: Export the uploaded object (evidence preservation)



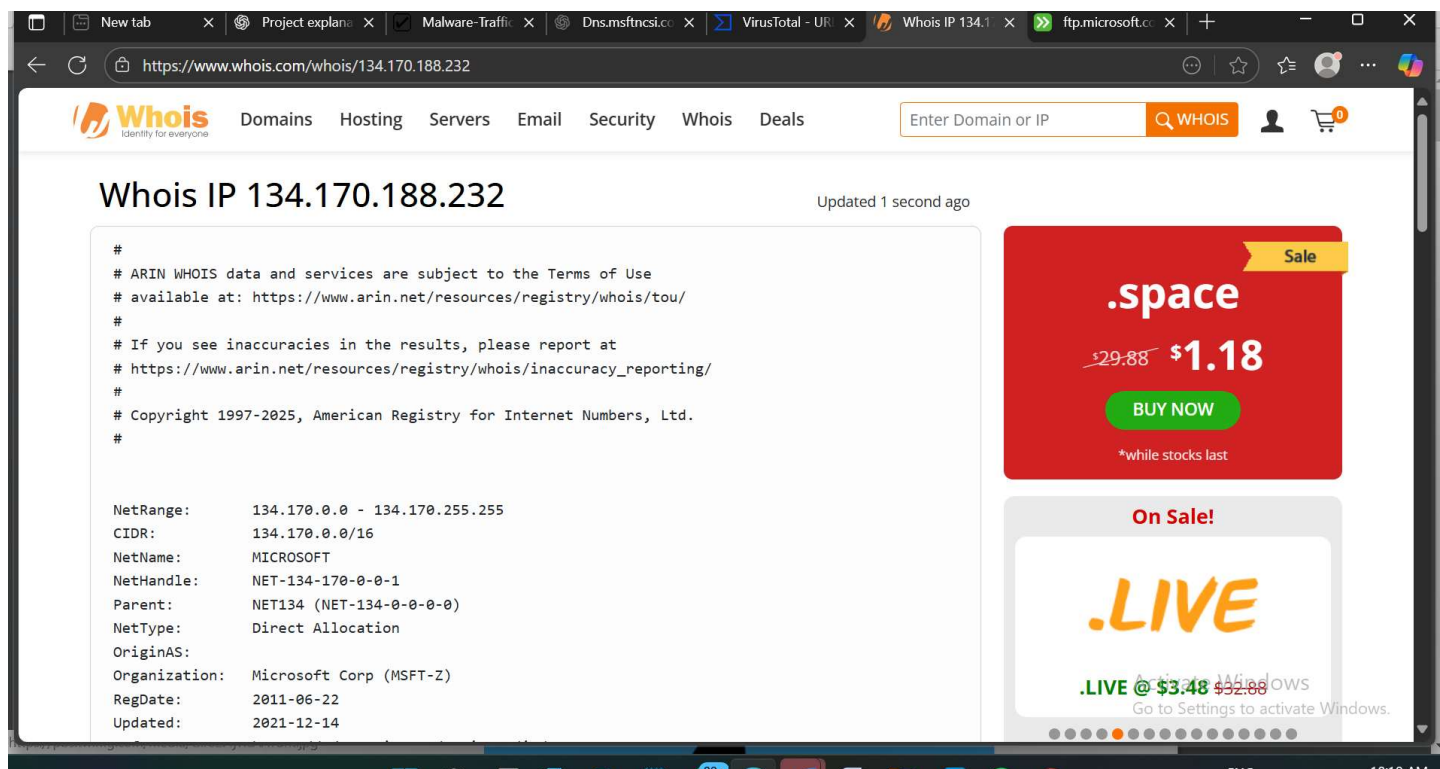
Step 7: Performed basic checks on the destination ip address and the domain name using WHOIS, I found out that Reverse lookup / WHOIS did not attribute the IP to internal infrastructure or to a corporate Microsoft service; it is hosted in outside provider space which means it's suspicious. It should be treated as attacker infrastructure. I went further into confirming the ip address of Microsoft FTP service..



The screenshot shows a web browser window with the URL <https://www.whois.com/whois/93.89.225.40>. The page displays WHOIS information for the IP address 93.89.225.40. The main content area shows the following details:

- % Abuse contact for '93.89.225.0 - 93.89.225.255' is 'abuse@isimtescil.net'**
- inetnum:** 93.89.225.0 - 93.89.225.255
- netname:** ISIMTESCIL2
- country:** TR
- admin-c:** SA43937-RIPE
- tech-c:** SA43937-RIPE
- status:** ASSIGNED PA
- mnt-by:** tr-isimtescil-1-mnt
- created:** 2025-05-08T14:47:40Z
- last-modified:** 2025-05-08T14:47:40Z
- source:** RIPE
- role:** SUPPORT -ISIMTESCIL
- address:** Tantavi Mah. Montesoglu Cad. No:25 Kat:2 Terra Plaza umraniye/ISTANBUL
- nic-hdl:** SA43937-RIPE
- mnt-by:** tr-isimtescil-1-mnt
- created:** 2025-05-08T14:27:30Z
- last-modified:** 2025-05-08T14:27:30Z
- source:** RIPE # Filtered
- % Information related to '93.89.225.0/24AS51557'**
- route:** 93.89.225.0/24

On the right side of the page, there are two promotional banners. The top banner is for ".site" domains, showing a price of \$3.28 (down from \$41.88). The bottom banner is for "WORDPRESS HOSTING" at \$5.48/month.



The screenshot shows a web browser window with the URL <https://www.whois.com/whois/134.170.188.232>. The page displays WHOIS information for the IP address 134.170.188.232. The main content area shows the following details:

- Whois IP 134.170.188.232** (Updated 1 second ago)
- # ARIN WHOIS data and services are subject to the Terms of Use**
- # available at: <https://www.arin.net/resources/registry/whois/tou/>**
- # If you see inaccuracies in the results, please report at https://www.arin.net/resources/registry/whois/inaccuracy_reporting/**
- # Copyright 1997-2025, American Registry for Internet Numbers, Ltd.**
- #**
- NetRange:** 134.170.0.0 - 134.170.255.255
- CIDR:** 134.170.0.0/16
- NetName:** MICROSOFT
- NetHandle:** NET-134-170-0-0-1
- Parent:** NET134 (NET-134-0-0-0-0)
- NetType:** Direct Allocation
- OriginAS:**
- Organization:** Microsoft Corp (MSFT-Z)
- RegDate:** 2011-06-22
- Updated:** 2021-12-14

On the right side of the page, there are two promotional banners. The top banner is for ".space" domains, showing a price of \$1.18 (down from \$29.88). The bottom banner is for ".LIVE" domains, showing a price of \$3.48 (down from \$32.88).

CONCLUSION: The infected system (10.1.31.101) was observed sending sensitive data to an external host (93.89.225.40). Network evidence including clear text FTP authentication (USER/PASS), the OPTS UTF8 ON negotiation, a STOR (PW_daviv.miller-DESKTOP-WE9H2FM_2025-01_31_20_24_25.html) command, and the subsequent FTP-DATA stream followed by a 226 Transfer complete response confirms that an upload to the remote endpoint occurred and was accepted. The payload naming pattern and session behavior are consistent with commodity information-stealing malware.

Key Takeaway:

Even when attackers abuse common services like **FTP**, their activity leaves **clear artifacts** — plaintext credentials, suspicious file uploads, and server responses confirming data transfers. By combining **protocol analysis (Wireshark)** with **threat intel checks**, defenders can spot exfiltration attempts that would otherwise blend into normal traffic.

What's your go-to method for analyzing suspicious network traffic? Let's discuss below

Incident Response Report: AgentTesla Malware Using FTP for Data Exfiltration

Incident Summary: An internal host (10.1.31.101) connected to an external FTP server (93.89.225.40) and used plaintext credentials to authenticate. It then issued a STOR command, uploading the file PW_daviv.miller-DESKTOP-WE9H2FM_2025-01_31_20_24_25.html. The server replied **"226 Transfer complete"**, confirming successful data exfiltration.

This report follows the four key steps of the incident response process: Identification, Containment, Eradication, and Recovery.

1. Identification

The incident was identified through suspicious FTP traffic originating from an internal (10.1.31.101). A detailed review in Wireshark revealed clear signs of malware-driven data exfiltration.

Evidence of Compromise:

FTP Session Analysis (ftp)

- The internal host connected to an external server 93.89.225.40 over FTP (port 21).
- The server banner replied 220 Microsoft FTP Service (may be spoofed).
- The client issued OPTS UTF8 ON → server confirmed with 200 OPTS UTF8 command successful.
- Clear text credentials observed: USER (pgizemM6) and PASS (giz95Ffg)

File Upload & Exfiltration (ftp-data)

- After login, the client executed a STOR command:
- STOR PW_daviv.miller-DESKTOP-WE9H2FM_2025-01_31_20_24_25.html
- This triggered an **FTP-DATA stream**, confirming file transfer.
- The server replied 226 Transfer complete, indicating successful upload.

Indicators of Exfiltration

- The exfiltrated file followed a naming convention typical of Agent Tesla payloads (desktop and timestamp).
- Use of random-looking credentials suggests malware-generated logins.

- External IP (93.89.225.40) is not part of organizational infrastructure; WHOIS/reputation checks flagged malicious associations.

Containment

Upon confirmation, immediate containment actions were taken to limit further data loss.

Short-Term Containment Actions

Blocked outbound FTP traffic to 93.89.225.40 at the firewall.

Disconnected the infected host (10.1.31.101) from the network.

Exported the malicious file and PCAP evidence for forensics.

Long-Term Containment Actions

Monitored outbound FTP traffic organization-wide to detect similar activity.

Implemented IDS signatures to flag suspicious FTP commands (`STOR`).

Segregated high-value assets to prevent lateral movement and exfiltration.

Eradication & Recovery

After containment, actions focused on removing the malware and restoring the system.

Eradication Actions

Conducted forensic imaging of the infected host.

Identified and removed persistence mechanisms (scheduled tasks, registry keys).

Scanned for additional malware artifacts (Agent Tesla variants).

Recovery Actions

Rebuilt the host from a clean, trusted backup.

Rotated all credentials exposed via plaintext FTP.

Strengthened endpoint detection and monitoring for abnormal FTP traffic.

Note: Even when malware uses a common service like FTP, it leaves clear trails: credentials, banners, and upload confirmations. With careful packet analysis and proper IR response, defenders can detect, contain, and eradicate such threats before widespread data theft occurs.

By applying these measures, similar attacks can be detected and mitigated faster in the future.