14 JANVIER 2026

NETWORK TRAFFIC ANALYSIS TOOL - USER MANUAL

SEE SUMMARY

ABIDERRAHMANE ADAM
STUDENT - BUT NETWORKS & TELECOMMUNICATIONSIUT ROANNE

SUMMARY

I. WHAT'S THIS ABOUT? :

So basically, we had this network issue at the India site - super slow connection, packets dropping everywhere. Turned out we needed to analyze tcpdump captures to find what's causing the problem.

I built this toolkit with 2 Python scripts and an Excel macro. It's pretty straightforward once you get it, trust me.

Setup (Do This First) :

- Script 1 converts messy tcpdump files into clean CSV
- Script 2 analyzes the CSV and makes nice charts
- Excel macro does extra analysis if you need it

Let's go through it step by step.

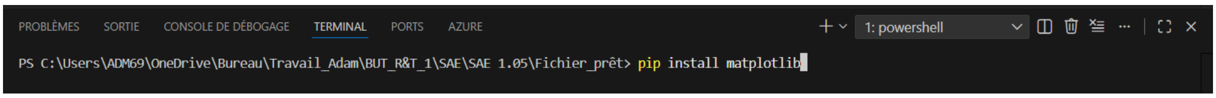
II. SETUP (DO THIS FIRST) :

You need Python 3 installed. If you don't have it:

- Go to python.org and download it
- When installing, CHECK that box that says "Add to PATH" (important!)

Then open cmd/terminal and run:

pip install matplotlib



That's the graphing library. Takes like 30 seconds to install.

For the Excel part, you obviously need Excel (2016 or newer works fine).

III. PART 1: CONVERTING TCPDUMP TO CSV :

What You're Doing :

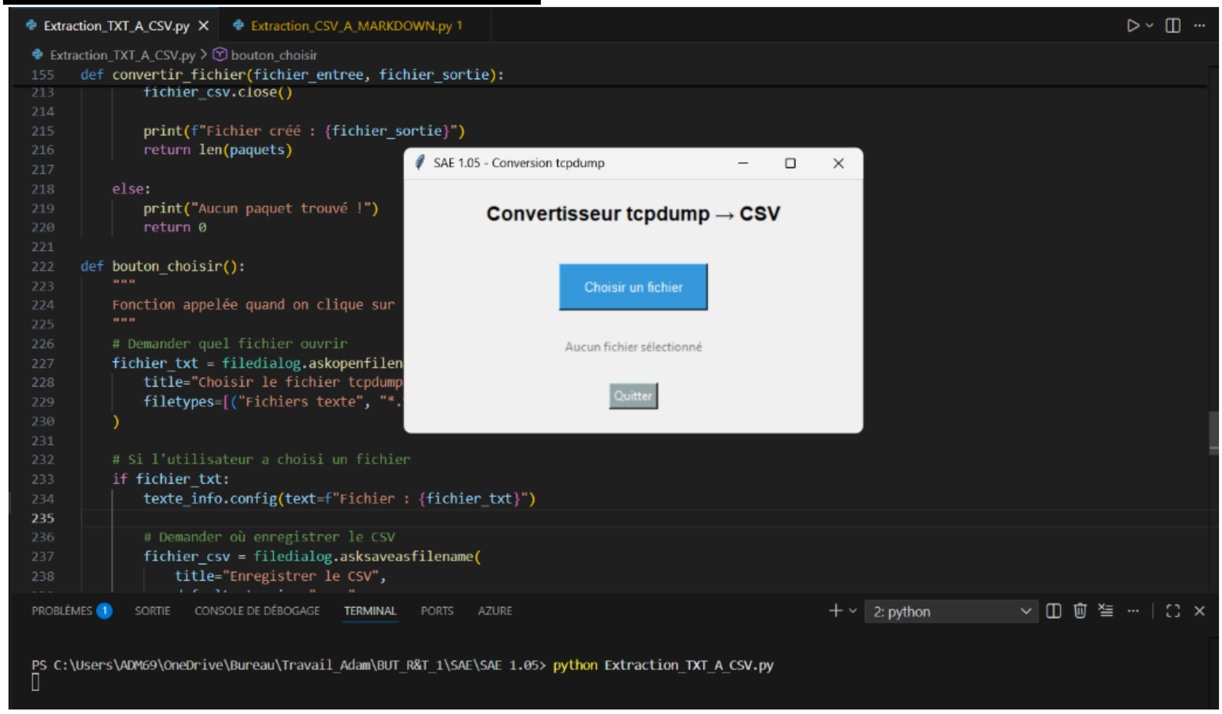
Tcpdump files are unreadable. This script parses them into a proper spreadsheet format.

Steps :

1. Put `Extraction_TXT_A_CSV.py` and your tcpdump file in the same folder (makes life easier)

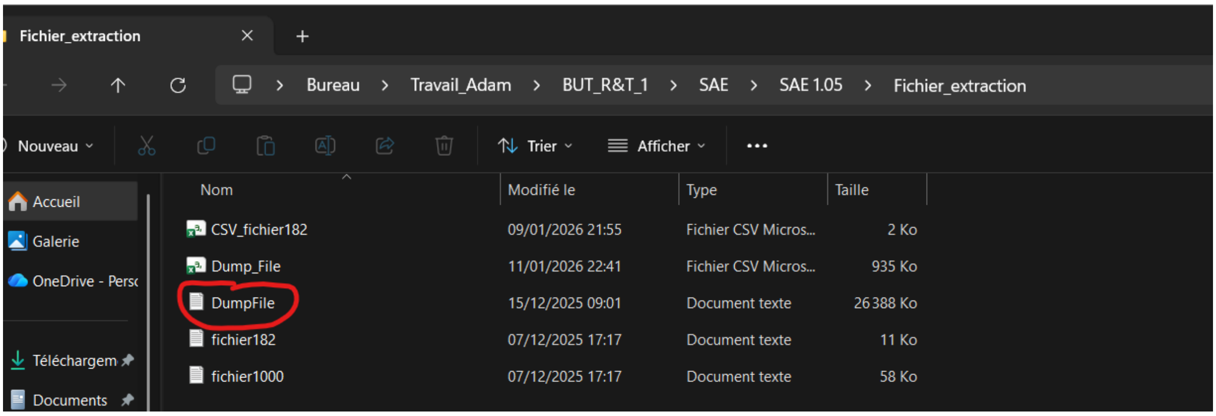
2. Open terminal/cmd in that folder and run:

```
python3 Extraction_TXT_A_CSV.py
```

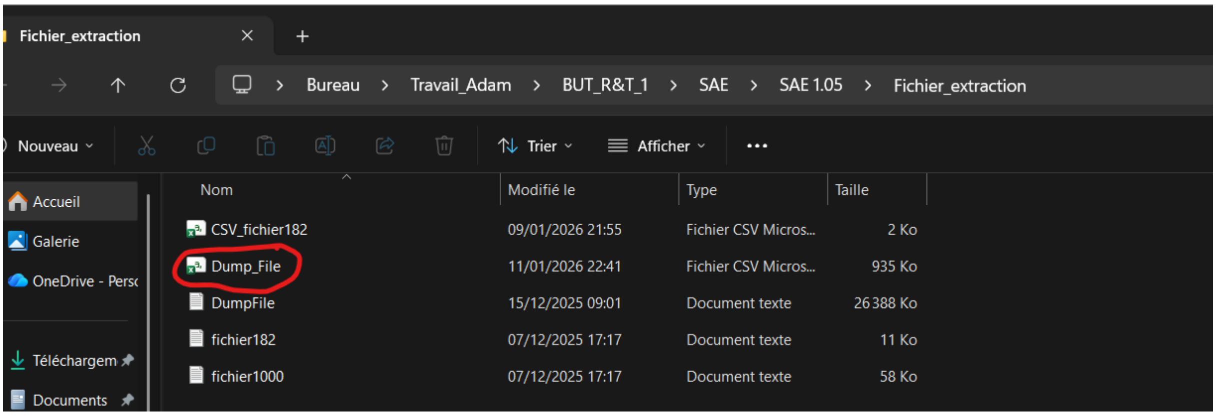


3. A window pops up. Click "Choisir un fichier" that's choose file

4. Select your .txt tcpdump file



5. Save it as DumpFile.csv



6. Wait a bit. The console shows you progress:

- Lines read
- Packets extracted
- Lines ignored

7. Done! Your CSV is ready.

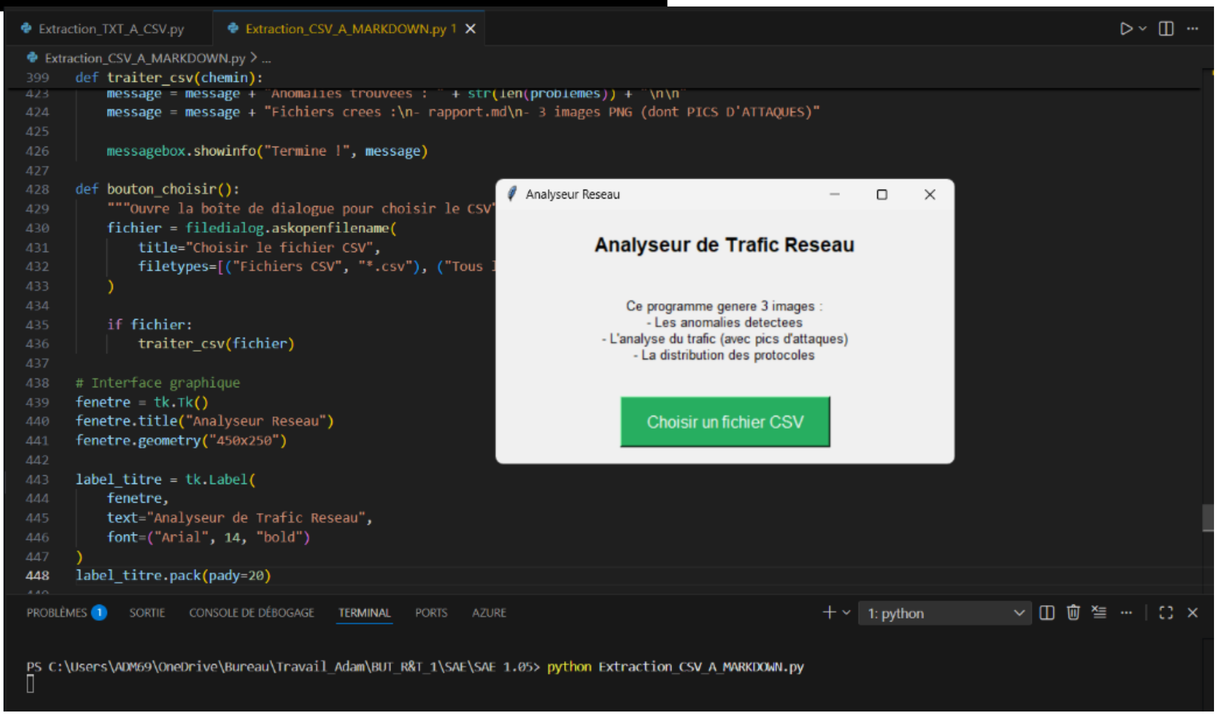
IV. PART 2: ANALYSIS & CHARTS :

This is where it gets interesting. The script analyzes everything and detects anomalies automatically.

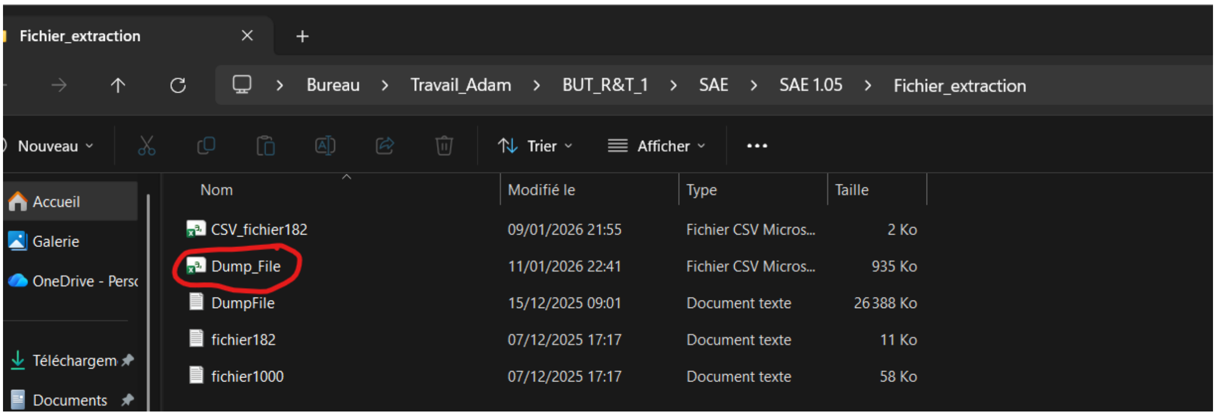
Steps :

1. Run the second script:

python3 Extraction_CSV_A_MARKDOWN.py



2. Window opens, choose your CSV file from Part 1



3. It crunches the numbers (takes 1-2 mins for big files)

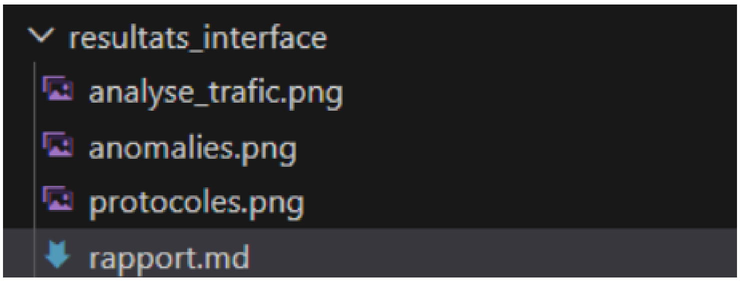
4. Check the `resultats_interface/` folder that gets created:

- **anomalies.png** - Shows if something's wrong (red = bad, green = all good)

- **analyse_trafic.png** - 3 graphs showing traffic patterns and top IPs

- **protocoles.png** - Pie chart of protocols (TCP, UDP, ICMP, etc.)

- **rapport.md** - Full text report with all the stats



What to Look For :

If anomalies.png shows red text, you've got issues:

1. **SYN FLOOD** (serious)
   - One IP sending 100+ SYN packets
   - Usually means DDoS attack or infected machine
   - Action: Block that IP asap

2. **PORT SCAN** (medium)
   - One IP trying 50+ different ports
   - Someone's looking for open services
   - Action: Investigate, might be recon

3. **TRAFFIC FLOOD** (serious)
   - One IP = 40%+ of all traffic
   - Could be legit (backup server) or problem (compromised host)
   - Action: Check what that IP is doing

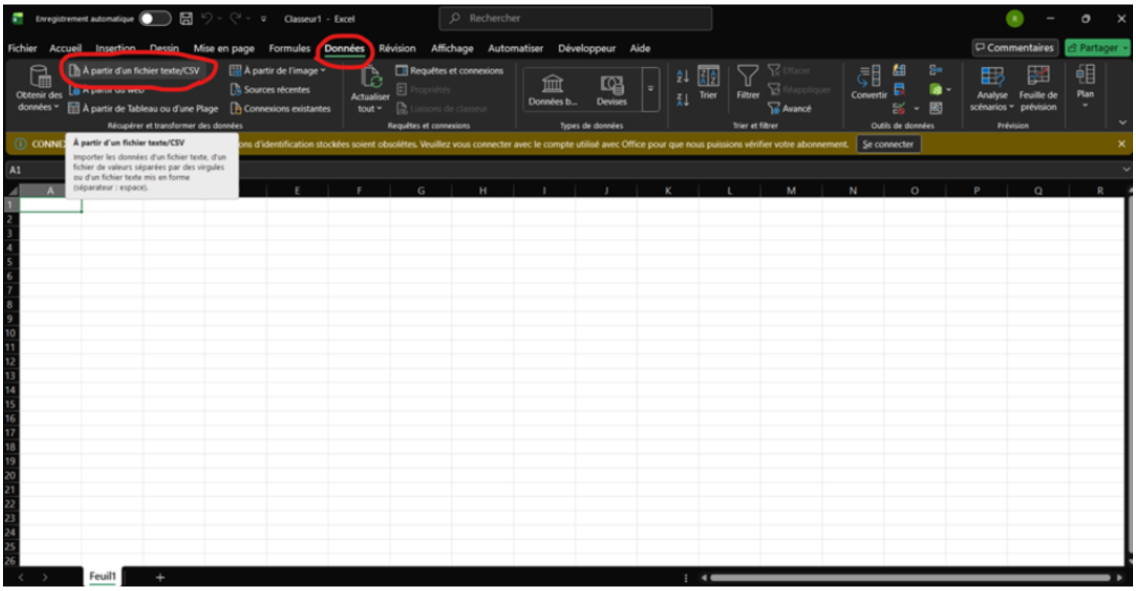
The thresholds (100, 50, 40%) are in the code if you want to adjust them.

This tool was specifically designed to identify the two suspicious activities reported on the India production site: SYN Flood and Port Scan

V. PART 3: EXCEL ANALYSIS (OPTIONAL) :

If you want more details or prefer working in Excel:

1. Open Excel



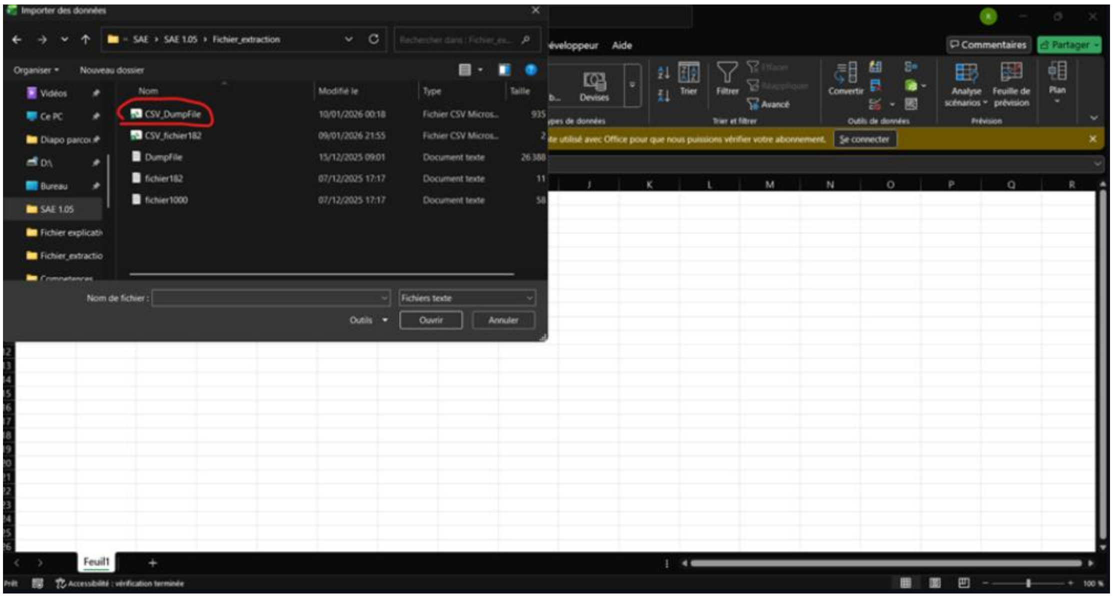
2. Import your CSV:

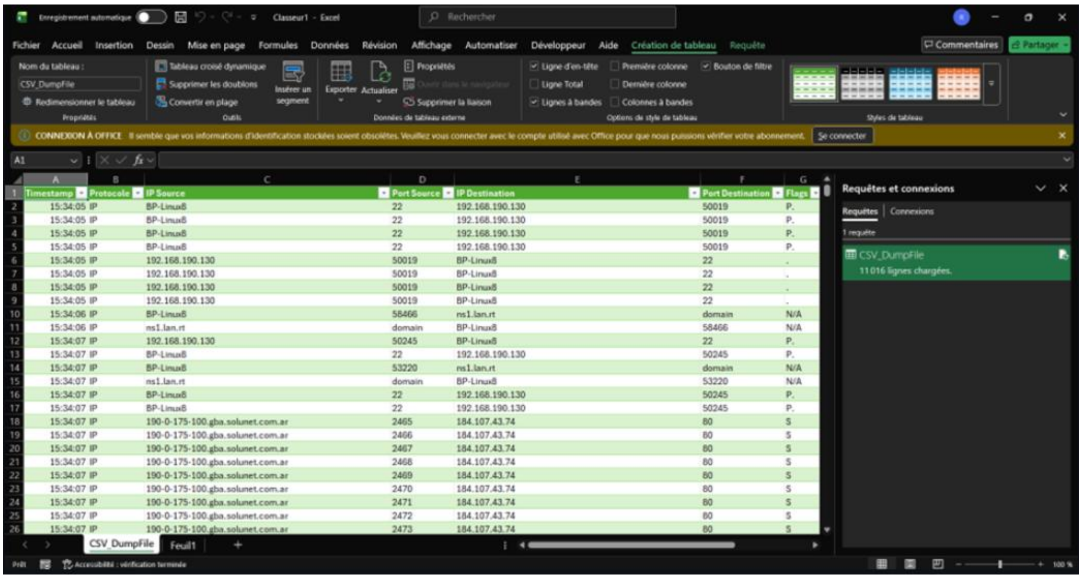
- Data tab → From Text/CSV

- Pick your file

- Make sure delimiter is set to **semicolon** (not comma!)

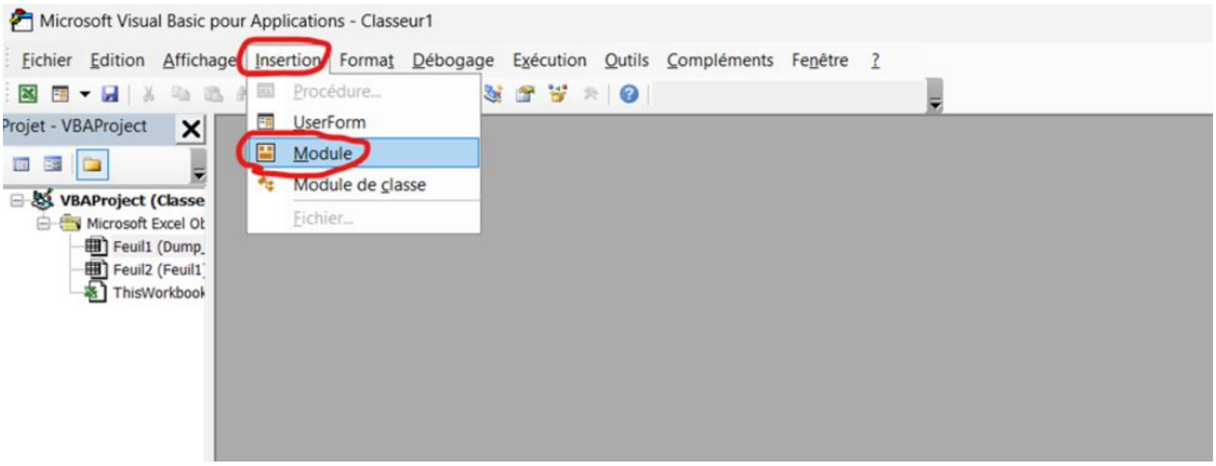
- Load it

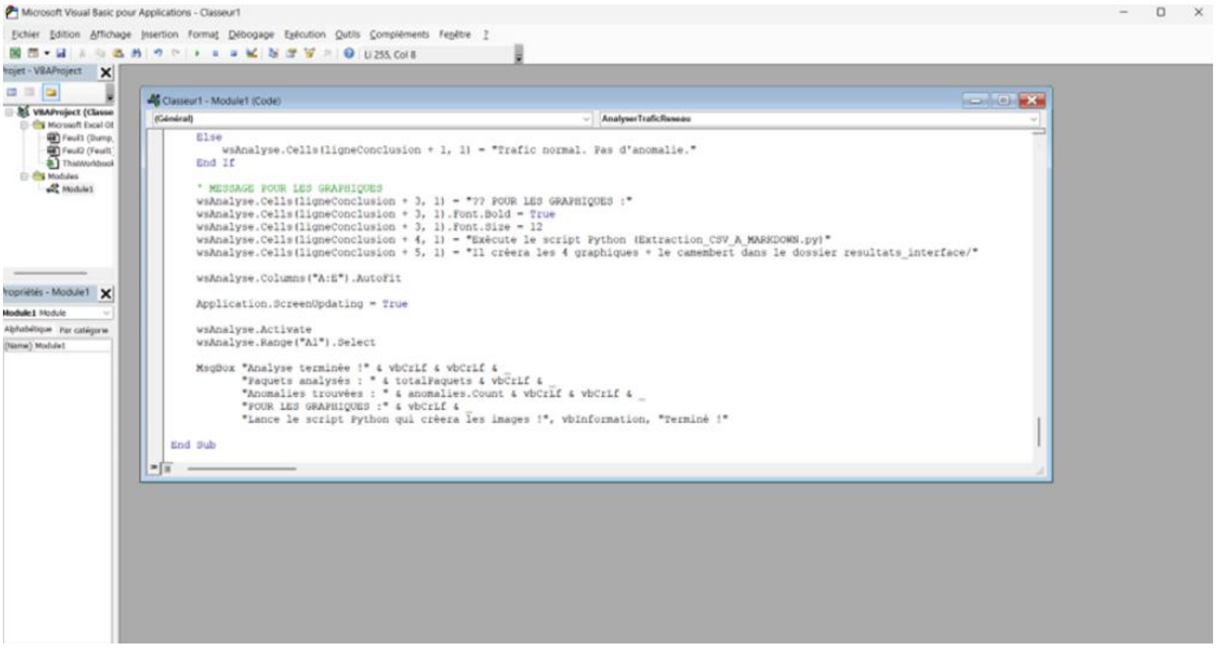




3. Install the macro:

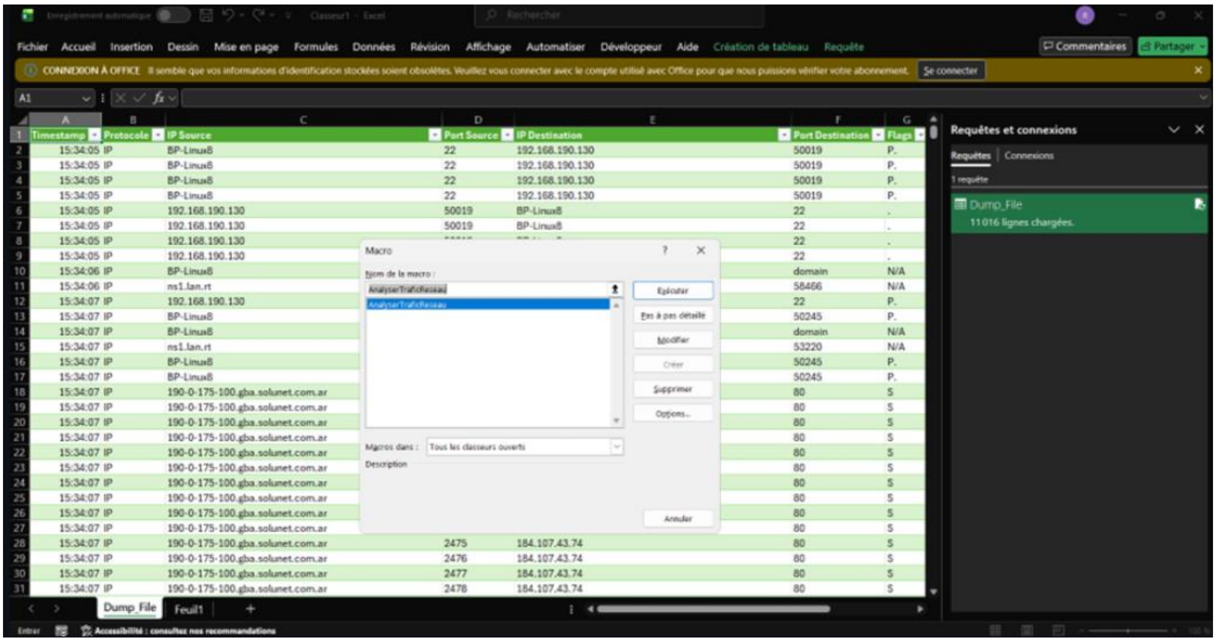
- Alt+F11 to open VBA editor
- Right click → Insert Module
- Copy everything from my github **vba.txt** and paste it
- Alt+F11 to close





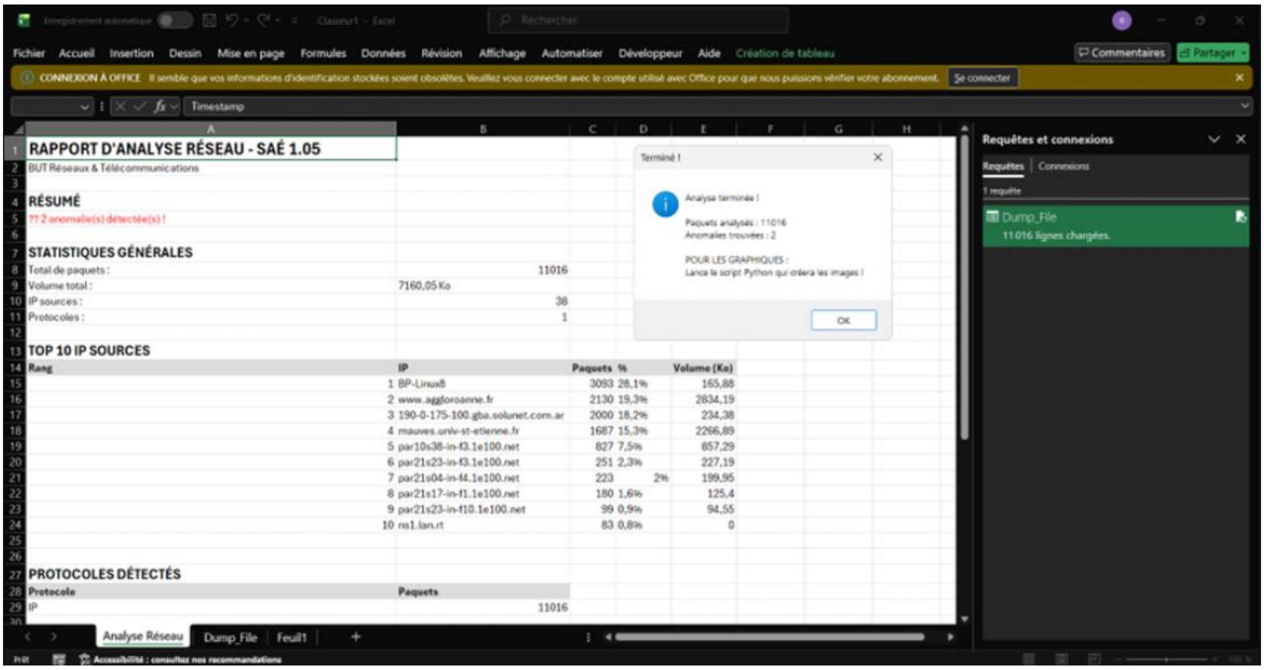
4. Run it with the macro:

- Alt+F8



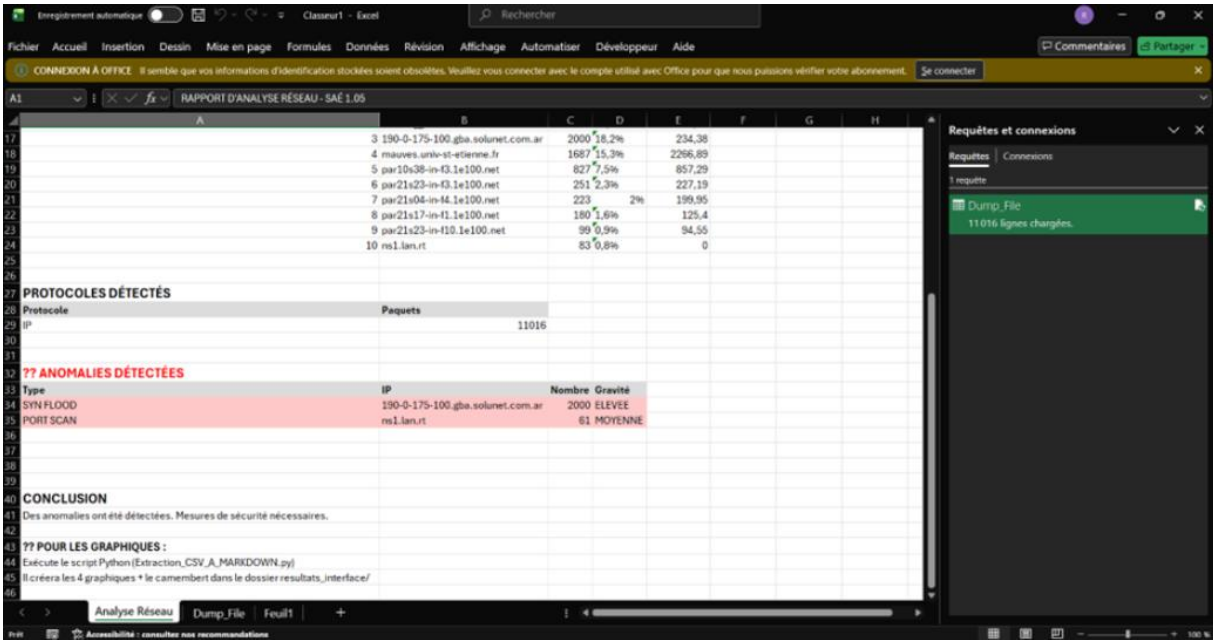
- Select **AnalyserTraficReseau**

- Click Run



5. New sheet appears: **"Analyse Réseau"**

- Top 10 IPs with stats
- Protocol breakdown
- Anomalies table (if any)
- Colored cells for problems (red = issue)



Pretty useful if your boss wants an Excel report.

VI. UNDERSTANDING THE CSV :

The CSV has 11 columns. Here's what matters:

TIMESTAMP	**WHEN THE PACKET WAS SENT (HH:MM:SS)**
PROTOCOL	**TCP, UDP, ICMP,**
IP SOURCE	**Who sent it**
PORT SOURCE	**From which port**
IP DESTINATION	**Where it's going**
PORT DESTINATION	**To which port (80=web, 22=ssh, 443=https)**
FLAGS	**S=new connection, .=data, F=closing**
LENGTH	**Packet size in bytes**
SEQ	TCP sequence number
ACK	TCP acknowledgment number
WINDOWS	TCP flow control value

These three columns (Seq, Ack, Window) are used for advanced TCP debugging and connection troubleshooting. They are not needed for basic anomaly detection (SYN floods, port scans, traffic floods), which only require IP addresses, ports, flags, and packet length.

VII. COMMON PORTS (QUICK REFERENCE) :

When you see these ports in the CSV :

22	SSH	
80	HTTP	
443	HTTPS	

VIII. TROUBLESHOOTING :

- **Problem: "ModuleNotFoundError: matplotlib"**

Fix: ` pip install matplotlib ` (you probably forgot this)

- **Problem: CSV shows everything in column A**

Fix: In Excel, select column A → Data → Text to Columns → Delimited → Check "Semicolon"

- **Problem: Python script won't open**

Fix: Make sure Python is in your PATH. Reinstall Python and CHECK that box during setup.

- **Problem: Takes forever to run**

Fix: Your file is probably huge. Try analyzing just 1 hour of traffic first to test.

- **Problem: VBA macro gives error**

Fix: Make sure the CSV is actually imported (should have 11 columns). If it's all in one column, see solution above.

IX. BEST PRACTICES AND RECOMMENDATIONS :

Good practices :

- Capture traffic during normal hours (get baseline)

- Always keep the original .txt file (in case you need to reprocess)

Things to avoid :

- Don't capture passwords/sensitive data (use filters)

- Don't panic if you see unknown IPs (check first)

- Don't edit the CSV manually (breaks the analysis)

For deployment :

- Test on small capture first
- Adjust thresholds if you get too many false positives
- Document any modifications you make

X. WHAT TO DO IF YOU FIND ISSUES :

Serious (red) anomalies :

1. Note the IP address
2. Check what that IP is (user PC, server, external?)
3. Email IT/security team immediately with:
   - The IP address
   - Type of anomaly (SYN flood, port scan, etc.)
   - The charts (attach anomalies.png)
   - The timestamp range

Medium (orange) anomalies :

1. Note the IP
2. Check if it's a known scanner (security team doing their job?)
3. Monitor it for 24h
4. If it continues, escalate

No anomalies (green) :

- Great ! But still review the traffic patterns
- Sometimes issues are subtle (not caught by thresholds)
- Check the Top 10 IPs manually

XI. FILE STRUCTURE :

```
project_folder/
├── Extraction_TXT_A_CSV.py
├── Extraction_CSV_A_MARKDOWN.py
├── vba.txt
├── DumpFile.txt (your input)
├── DumpFile.csv (converted)
└── resultats_interface/
    ├── anomalies.png
    ├── analyse_trafic.png
    ├── protocoles.png
    └── rapport.md
```

Keep this organized. You might need to go back to old captures later.

XII. MODIFYING DETECTION THRESHOLDS :

If you want to adjust when anomalies trigger (too sensitive/not sensitive enough):

1. Open `Extraction_CSV_A_MARKDOWN.py` in any text editor

2. Find the function `detecter_problemes()` (around line 115)

3. Change these values:

```
if nb_syn > 100:        # SYN flood threshold (default: 100)
if nb_ports > 50:       # Port scan threshold (default: 50)
if pourcentage > 40:    # Traffic flood threshold (default: 40%)
```

Lower values = more sensitive (catches more but more false positives)

Higher values = less sensitive (only catches serious issues)

XIII. DEPLOYMENT NOTES (FOR INDIA TEAM) :

When you deploy this in India:

1. Install Python 3.8 or higher + matplotlib on the analysis machine

- Operating Systems: Windows 10+, Linux, macOS 10.14+

- Administrator privileges for installation

- Minimum 4GB RAM for large capture files

Installation Steps :

Update package manager

sudo apt update

Install Python 3 and pip

sudo apt install python3 python3-pip -y

Install matplotlib pip3

install matplotlib

Verify installation

python3 --version pip3 list | grep matplotlib

Transfer Files :

Clone from GitHub

git clone https://github.com/AdamAbiderrahmane/SAE1.05 cd SAE1.05

Test the scripts :

1.python Extraction_TXT_A_CSV.py

2.python Extraction_CSV_MARKDOWN.py

2. Test with a small capture first

3. The tool expects tcpdump format .txt

4. Results go to **resultats_interface/** folder (created automatically)

5. If you get false positives, adjust thresholds (see above)

Contact :

For technical support : adam.abiderrahmane@etu.univ-st-etienne.fr

My github : https://github.com/AdamAbiderrahmane/SAE1.05

I'll help you get it working.

XIV. FINAL NOTES :

This tool isn't perfect. It's meant to give you a starting point for investigating network issues. Always:

- Cross-reference with your IDS/firewall logs
- Verify suspicious IPs before blocking
- Document what you find
- Keep captures if needed

The thresholds work for our setup but might need adjustment for yours. Network profiles differ.

Good luck! Hope this helps solve the saturation issue.

v1.0 - Jan 2026

Created for SAE 1.05 project - BUT R&T