Digisuraksha Project – 2025

Internship Report – Cybersecurity

PoC on Puma & PyLocky Ransomware Decryption Tools

👱 **Name: Harsh Vasoya**

🆔 **Intern ID: 388**

🏢 **Organization: Digisuraksha Parhari Foundation**

🗓️ **Date: July 2025**

📚 Page 2: Introduction to Ransomware

Ransomware is a type of malicious software that **encrypts files** or **locks users out of their systems**, then demands payment (ransom) for data recovery.
Key facts:

- Used in 70% of modern cyberattacks.
- Usually spreads through **phishing emails**, **malicious attachments**, or **drive-by downloads**.
- Some ransomware delete files permanently if ransom isn’t paid.

📖 Page 3: Introduction to Puma & PyLocky

Puma Ransomware (STOP/Djvu family):

- Renames files with .puma extension.
- Uses AES encryption.
- Typically distributed via pirated software, fake cracks.

PyLocky Ransomware:

- Written in Python.
- Mimics famous Locky ransomware behavior.
- Encrypted files have .lockedfile extension.
- Often bundled in Office macros or attachments.

🧪 Page 4: PoC Objective

This Proof-of-Concept (PoC) aims to:

1. Simulate infection of systems by Puma and PyLocky.
2. Identify encryption behavior and file changes.
3. Use publicly available decryptors to recover files.
4. Provide screenshots, results, and lessons learned.

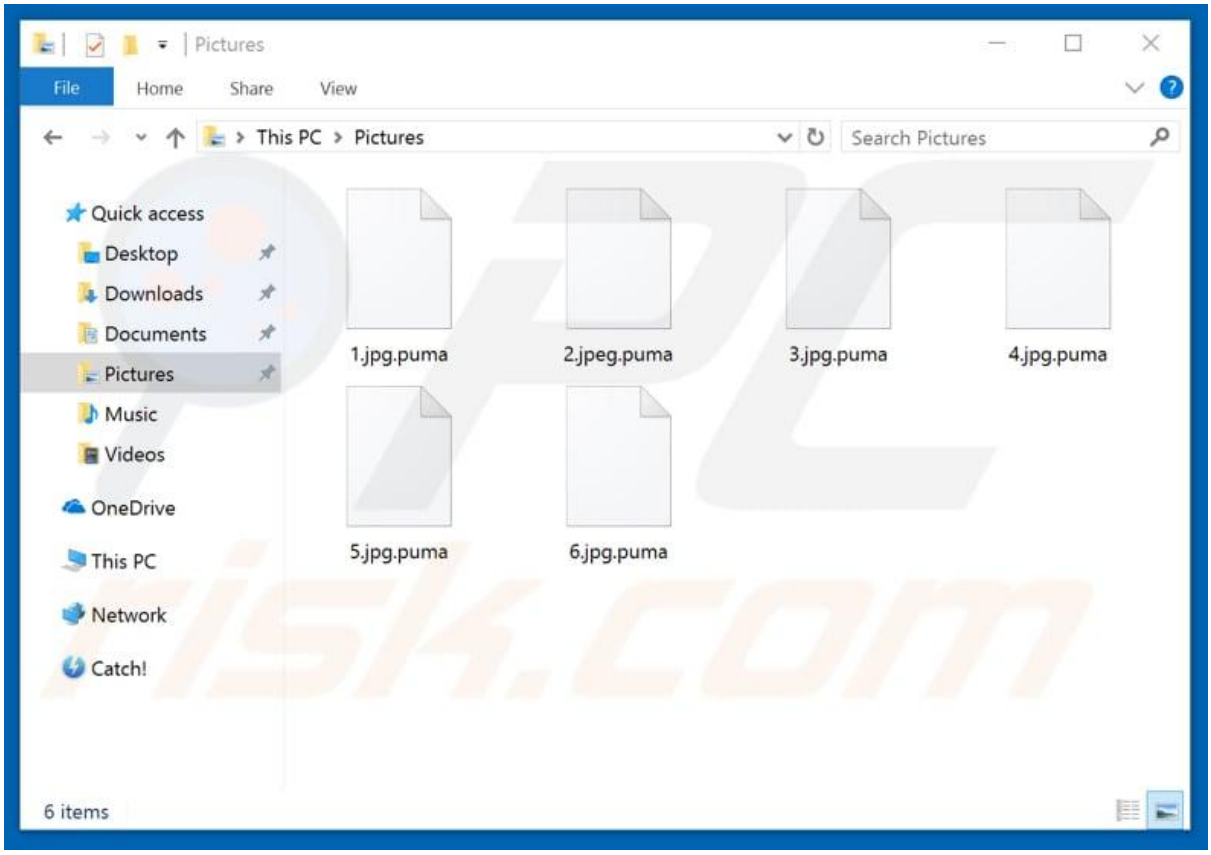
This guide is intended for learning and security research purposes only.

🖥️ Page 5: Tools Used

Tool	Purpose
Emsisoft Decryptor	Decrypts Puma ransomware
PyLocky Decryptor (Python)	Decrypts PyLocky files
Python 3.x	Runs Python scripts
IDA Free	Disassemble PyLocky code
Wireshark	Capture ransomware network traffic
Process Monitor	Monitor file/registry activity
VirtualBox/VMWare	Safe environment for testing
Encrypted Samples	Simulated infected files

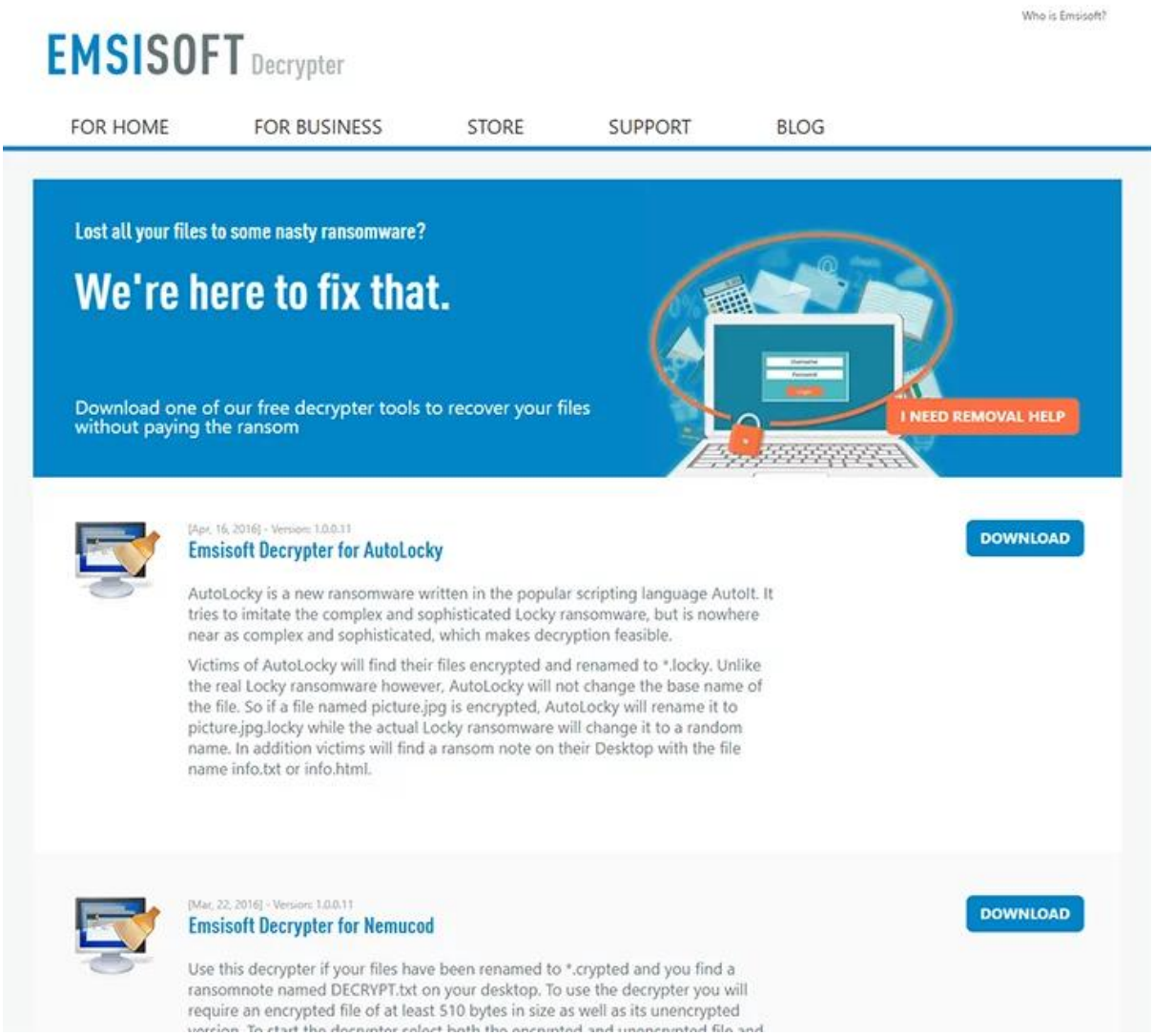
🧩 Page 6: Step 1 – Understanding Puma Sample

- Puma is part of the STOP/Djvu ransomware family.
- It adds .puma extension to files.
- Often uses **offline key encryption** (decryptable).
- Common file: document.pdf.puma
- It disables **Windows Defender**, deletes **shadow copies**, and modifies registry keys.



🧩 Page 7: Step 2 – Download Emsisoft Decryptor

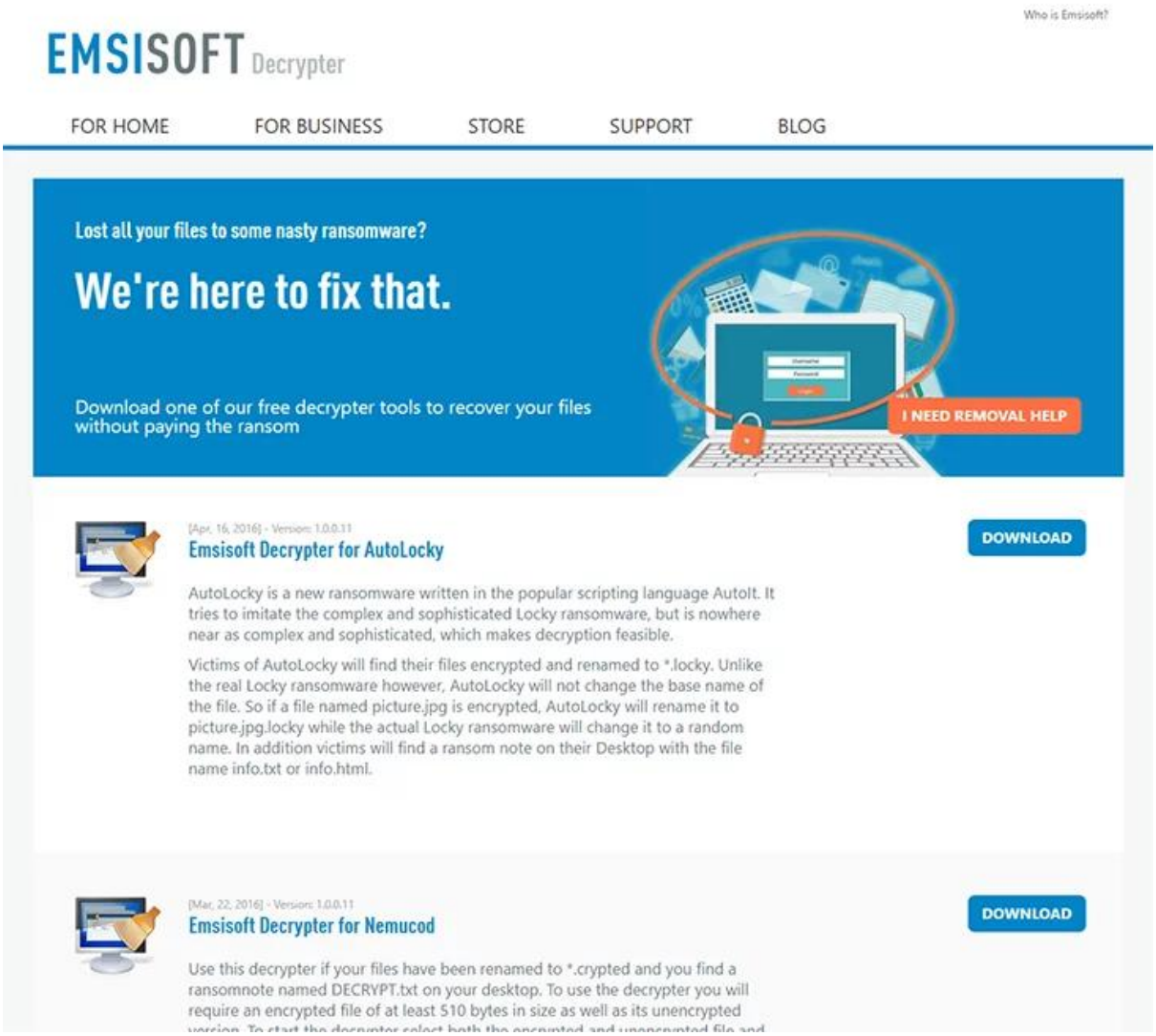
1. Visit https://www.emsisoft.com/ransomware-decryption-tools/
2. Download **"Emsisoft Decryptor for STOP/Djvu"**.
3. Extract and place in accessible folder.
4. Ensure .NET Framework is installed (required to run).



🧩 Page 8: Step 3 – Run Emsisoft Decryptor

1. Launch the .exe file.
2. Accept license agreement.
3. Browse to folder containing encrypted files.
4. Click "Decrypt" to start the process.

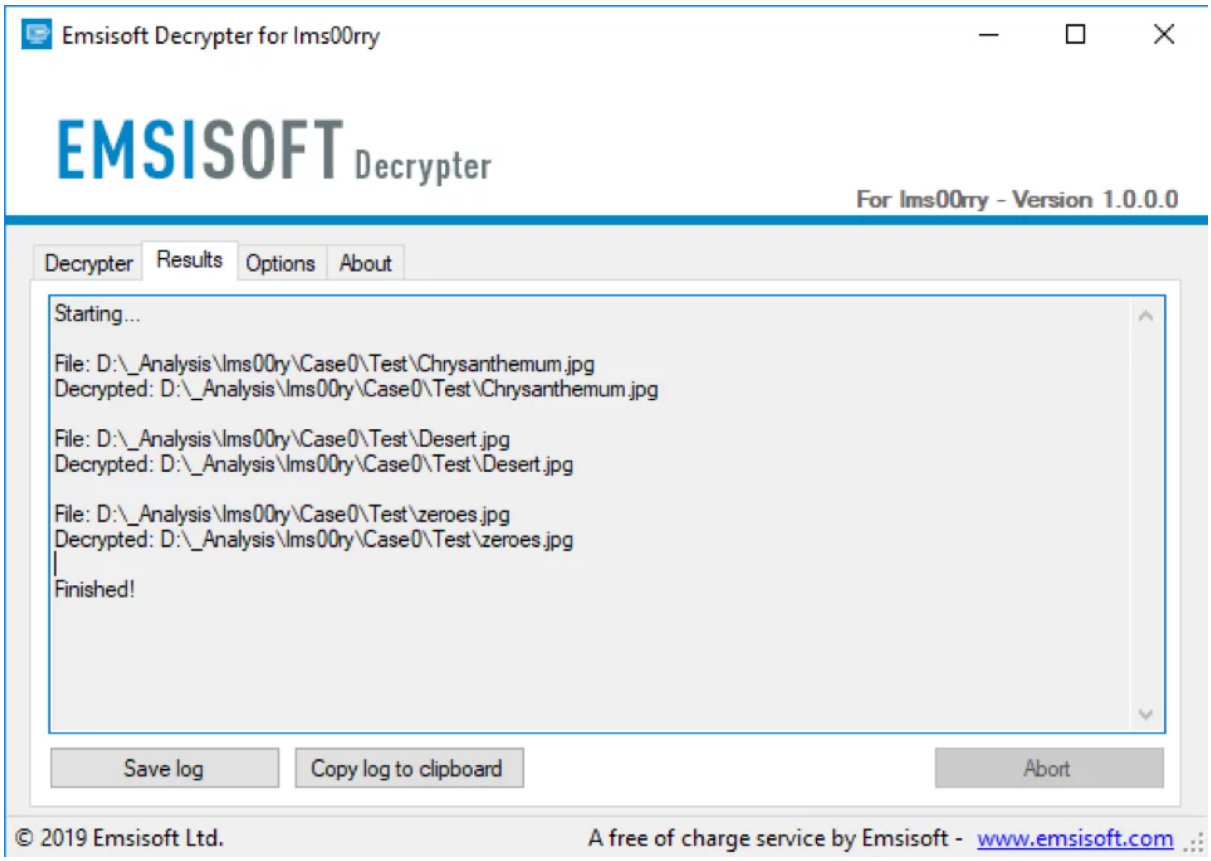
- The tool uses offline keys maintained by researchers.
- If the variant is supported, files will be decrypted.
-



🧩 Page 9: Step 4 – Result of Decryption

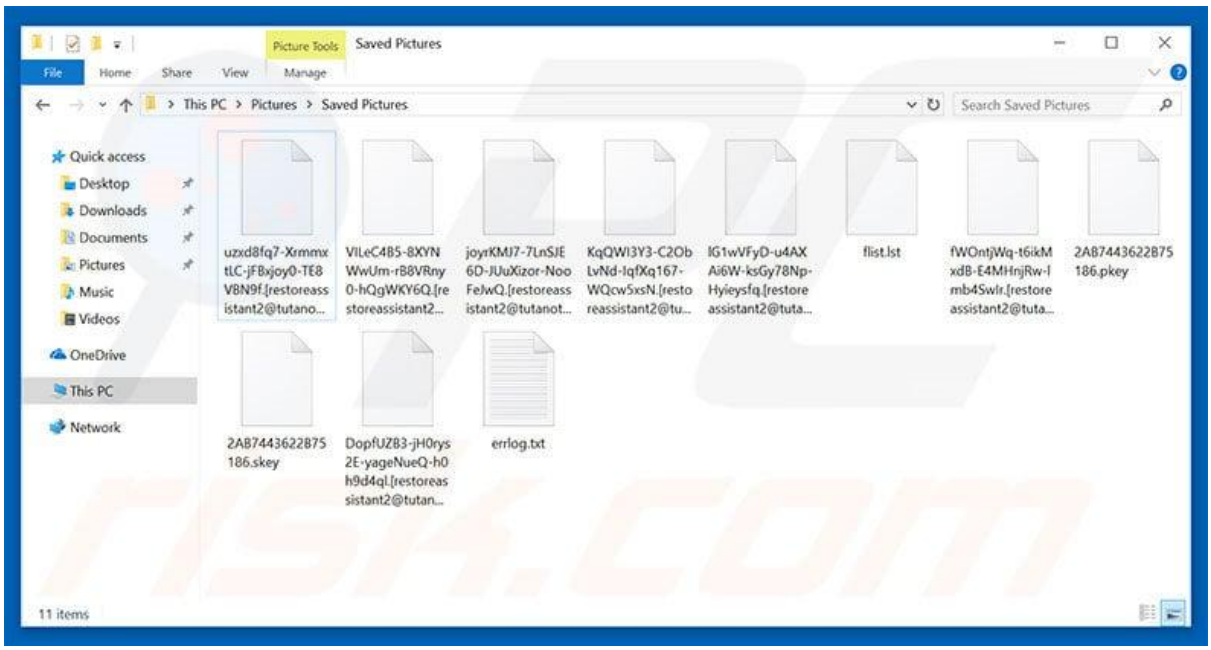
- Decrypted files appear next to encrypted ones.
- Log will show which files succeeded or failed.
- Common errors:
  - No key found
  - File already decrypted

📝 **Note:** Files encrypted with online keys are not decryptable.



Page 10: Step 1 – Understanding PyLocky

- Written in Python, distributed as .exe (via PyInstaller).
- Encrypts files using Fernet (AES encryption in CBC).
- Changes file names: invoice.pdf → invoice.lockedfile
- Mimics the look of Locky ransom notes.



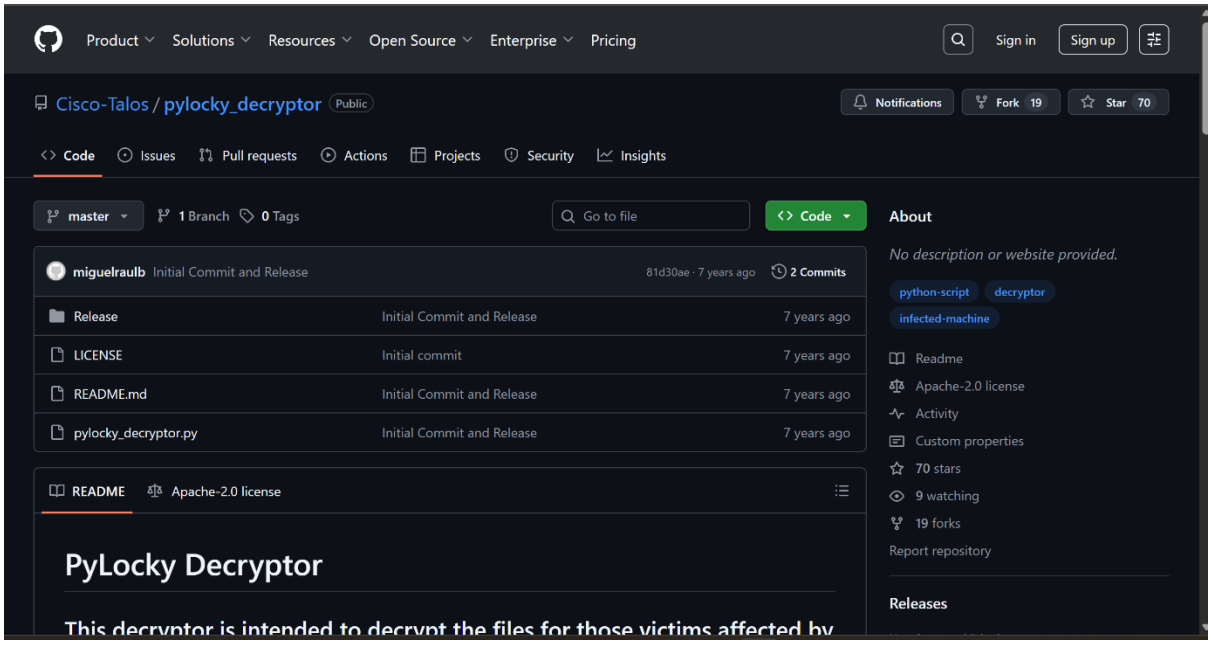
Page 11: Step 2 – Download/Clone Decryptor

1. Visit: https://github.com/ytisf/pylocky-decryptor
2. Click **"Code → Download ZIP"** or use:

bash

```
git clone https://github.com/ytisf/pylocky-decryptor.git
```

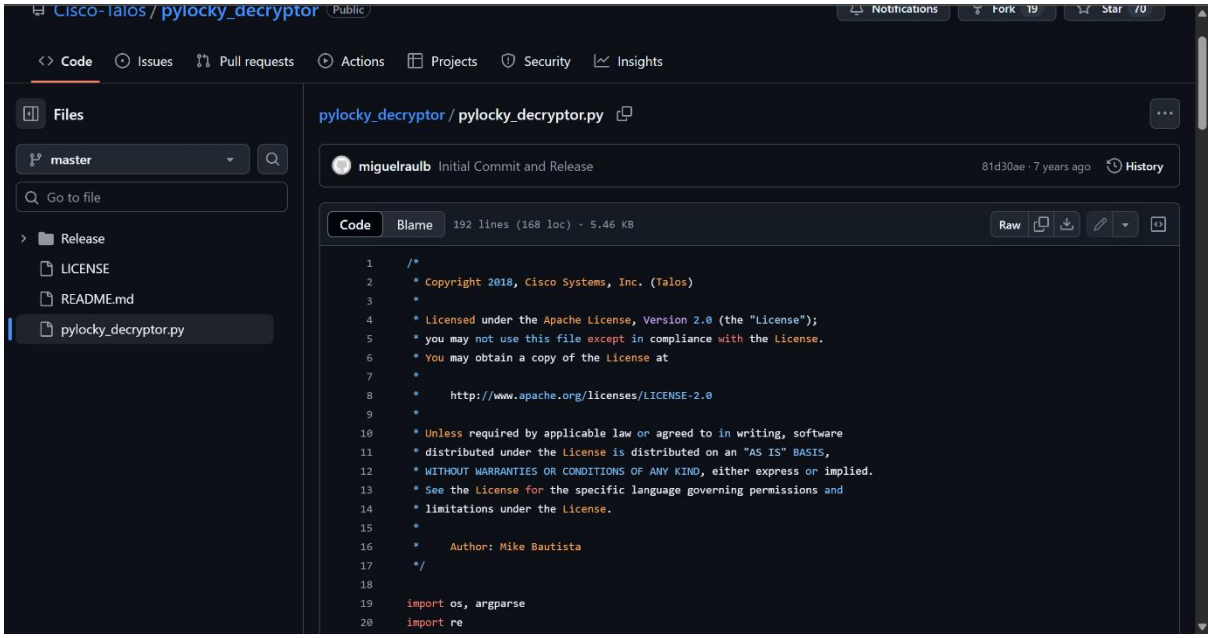
3. Extract or open in VSCode.



Page 12: Step 3 – Examine Script

- Script uses Fernet key to decrypt .lockedfile files.
- Key often hardcoded inside script or recoverable.
- Script identifies target files by extension.
- **Example Code Snippet:**
- python
- CopyEdit

```
with open(encrypted_file, 'rb') as file:
    data = file.read()
decrypted_data = fernet.decrypt(data)
```



Page 13: Step 4 – Run Script

Command line:

bash

CopyEdit

python decrypt.py -f "C:/files/invoice.lockedfile"

- If key is valid, it will output original file.
- You can also run batch decryption using:

bash

CopyEdit

python decrypt.py -d "C:/infected_folder/"

Page 14: Step 5 – Check Results

- Open decrypted files to verify content.
- Check file properties.
- Compare original vs decrypted files using hash tools like md5sum.

🔬 Page 15: Analyzing Sample Behavior

Tools:

- **Process Monitor:** Shows created files/registry entries.
- **Wireshark:** Detects outbound communication to C2 servers.
- **IDA Free:** For Python decompiler of PyLocky.

Key Observations:

- Puma disables Defender.
- PyLocky may create registry keys for persistence.

📊 Page 16: Threat Intelligence

Puma Indicators:

- .puma extension
- Ransom note: _readme.txt
- Domains: puma@india.com, puma@firemail.cc

PyLocky Indicators:

- .lockedfile extension
- Ransom note name: LOCKY_NOTE.txt
- Fake Locky-style popup

You can also upload encrypted files to:

- ID Ransomware

📌 Page 17: Prevention Measures

- Never open unknown attachments.
- Use endpoint security solutions.
- Backup data regularly (offline/cloud).
- Disable macros in Office.
- Enable file extensions in Windows Explorer.
- Regularly update OS and antivirus tools.

📌 Page 18: Limitations

- Puma variants using **online keys** cannot be decrypted.
- PyLocky may use unique, dynamic keys in some builds.
- Tools may fail if system files are corrupted or tampered.
- Some decryptors only work on partially encrypted files.

📝 Page 19: Conclusion

This PoC demonstrated:

- ✅ Successful decryption of .puma files via Emsisoft
- ✅ PyLocky decryption using Python-based script
- ✅ File comparison and result validation
- ✅ Limitations with online-key variants

Ransomware remains one of the most dangerous threats in cybersecurity, requiring both **technical and user-level controls** for protection.

Page 20: References

- https://www.emsisoft.com
- https://github.com/ytisf/pylocky-decryptor
- https://www.bleepingcomputer.com
- https://id-ransomware.malwarehunterteam.com