LOCKBIT 3.0 RANSOMWARE ATTACK & AI DETECTION SYSTEM- Complete SetupGuide

Harsha Vardhan Govada	Prasanna Jonnadula	Sathvika Veluru	Akarsh Reddy Sattu
hgovada@gmu.edu	pjonnadu@gmu.edu	sveluru2@gmu.edu	asattu@gmu.edu
G01482708	G01482709	G01476762	G01548211

NOTE: BEFORE YOU START - EXTREMELY IMPORTANT
1. This project uses REAL encryption (AES-256)
2. Only run in ISOLATED virtual machines
3. NEVER connect VMs to your real network
4. ALWAYS take VM snapshots before running
5. Files WILL be encrypted (but recoverable)
6.This is for EDUCATION ONLY - illegal if misused YOU HAVE BEEN WARNED - PROCEED WITH CAUTION!

1. What is This Project?

This cybersecurity instructional project mimics how ransomware functions and how to identify it. You'll discover:

How your files are encrypted by hackers
How assaults are detected by security systems
How to handle cyber incidents Actual cryptography (AES-256)
AI-driven threat identification

Consider it a secure cybersecurity "flight simulator" where you may practice without actual risk!

2. What You Need?

Required Software (Free)

1. VMware Workstation or VirtualBox (virtualization software)
    - VMware: https://www.vmware.com/products/workstation-player/workstation-player-evaluation.html
    - VirtualBox: https://www.virtualbox.org/wiki/Downloads
2. Operating System Images:
    - Ubuntu 20.04 Desktop ISO (for Controller and Attacker VMs)
        - Download: https://ubuntu.com/download/desktop
    - Windows 10/11 ISO (for Victim VM)
        - Download: https://www.microsoft.com/software-download/windows11

Hardware Requirements

1. Minimum:
    - CPU: 4 cores
    - RAM: 8 GB
    - Disk: 60 GB free space
    - Internet: For initial downloads only
2. Recommended:

- CPU: 6+ cores
- RAM: 16 GB
- Disk: 100 GB free space

Time Required

- First-time setup: 2-3 days
- Running simulation: 10 minutes
- Total project: One day

3. Step 1: Setup Virtual Machines

3.1 Create Three Virtual Machines

You need to create 3 separate VMs:

VM Name	OS	IP Address	Purpose
Controller	Ubuntu 20.04	192.168.64.30	C2 Server + Dashboard
Attacker	Ubuntu 20.04	192.168.64.20	Payload Generator
Victim	Windows 10/11	192.168.64.10	Target System

3.2 Network Configuration

IMPORTANT: All VMs must be on the same **isolated** network!

For VirtualBox:

1. File → Host Network Manager
2. Click Create
3. Configure Adapter:
    - IPv4 Address: 192.168.64.1
    - Subnet Mask: 255.255.255.0
4. Disable DHCP

For each VM:

For all 3 VMs use intent

1. Settings → Network
2. On Adapter 1: choose internal network
3. Name : intent
4. Adapter type: Intel PRO/1000 MT Desktop

3.3 Create Controller VM (Ubuntu)

1. Create New VM
    - Select "Linux" → "Ubuntu 64-bit"
    - Name: "Controller"
2. Allocate Resources:
    - CPU: 2 cores

- RAM: 4 GB
- Disk: 25 GB

3. Install Ubuntu:
   - Boot from Ubuntu ISO
   - Click "Install Ubuntu"
   - Username: student (or your choice)
   - Password: (choose a simple one like password)
   - Hostname: controller
4. Set Static IP:
   - sudo ip addr flush dev enp0s3
   - sudo ip addr add 192.168.64.30/24 dev enp0s3
   - sudo ip link set enp0s3 up

3.4 Create Attacker VM (Ubuntu)

Same process as Controller VM, but:

- **Set static ip as follows:**
  - sudo ip addr flush dev enp0s3
  - sudo ip addr add 192.168.64.20/24 dev enp0s3
  - sudo ip link set enp0s3 up

3.5 Create Victim VM (Windows)

1. Create New VM:
   - Select "Windows 10/11 64-bit"
   - Name: "Victim"
2. Allocate Resources:
   - CPU: 2 cores
   - RAM: 4 GB
   - Disk: 40 GB
3. Install Windows:
   - Boot from Windows ISO
   - Follow installation wizard
   - Create user account
4. Set Static IP:
   - Open PowerShell **as Administrator**:
   - powershell

 # Find your network adapter name

Get-NetAdapter

 # Set static IP (replace "Ethernet0" with your adapter name)

New-NetIPAddress -InterfaceAlias "Ethernet0" -IPAddress 192.168.64.10 -PrefixLength 24

5. **Verify IP:**
   - powershell
   - ipconfig
   - # Should show 192.168.64.10
6. Set-MpPreference -DisableRealtimeMonitoring $true

3.6 Test Network Connectivity

From Controller VM:

1. Terminal
   - ping 192.168.64.20 *# Should reach Attacker*
   - ping 192.168.64.10 *# Should reach Victim*

From Victim VM:

2. powershell
   - Test-NetConnection 192.168.64.30
   - Test-NetConnection 192.168.64.20

 # Both should succeed

If all pings work, network is configured correctly!

4. Step 2: Controller VM Setup

4.1 Install Required Software

Open Terminal on Controller VM:

1. # Update system
   - sudo apt update && sudo apt upgrade -y
2. # Install Python and tools
   - sudo apt install -y python3 python3-pip python3-venv git nano
3. # Install system dependencies

- sudo apt install -y build-essential libssl-dev libffi-dev python3-dev

4.2 Create Project Directory

1. mkdir -p ~/ransom_lab
2. cd ~/ransom_lab
3. *# Create virtual environment*
    - python3 -m venv venv
4. *# Activate virtual environment*
    - source venv/bin/activate

4.3 Install Python Packages

Terminal

- # Install required packages (this takes 2-3 minutes)
- pip install flask flask-cors streamlit pandas plotly cryptography requests psutil

4.4 Create C2 Server File

Terminal:

1. *# Create the C2 server*
    - nano c2_server.py

Copy and paste this code: (provided in artifacts - c2_server.py)

4.5 Create Detection Dashboard File

1. # Create the dashboard
    - nano detection_dashboard.py

Copy and paste this code: (provided in artifacts - detection_dashboard.py)

4.6 Make Files Executable

- chmod +x c2_server.py detection_dashboard.py

4.7 Test C2 Server

1. *# Start C2 server*
    - python3 c2_server.py

Success! Press Ctrl+C to stop (we'll start it properly later).

4.8 Configure Firewall

Terminal:

1. # *Allow C2 and Dashboard ports*
    - sudo ufw allow 5000/tcp
    - sudo ufw allow 8501/tcp
    - sudo ufw enable

5. Step 3: Attacker VM Setup

5.1 Install Python and Dependencies

Open Terminal on Attacker VM:

1. # *Update system*
    - sudo apt update && sudo apt upgrade -y
2. # *Install Python*
    - sudo apt install -y python3 python3-pip
3. # *Install required libraries*
    - pip3 install requests cryptography Pillow

5.2 Create Project Directory

- mkdir -p ~/ransom_lab/attacker
- cd ~/ransom_lab/attacker

5.3 Create Ransomware Payload

- nano lockbit3_realistic.py
- **Copy and paste this code:** (provided in artifacts - lockbit3_realistic.py)
- Save and exit (Ctrl+O, Enter, Ctrl+X).

5.4 Create Decryption Tool

- nano decrypt_lockbit.py
- **Copy and paste this code:** (provided in artifacts - lockbit3_realistic.py)
- Save and exit (Ctrl+O, Enter, Ctrl+X).

5.5 Make Files Executable

- chmod +x lockbit3_realistic.py decrypt_lockbit.py

6. Step 4: Victim VM Setup

6.1 Install Python on Windows

6.2 python –version to verify installation

6.3 Create Lab Directory

New-Item -Path "C:\lab" -ItemType Directory -Force
cd C:\lab # *Create test data folder*
New-Item -Path "C:\lab\test_data" -ItemType Directory -Force

6.4 Create Test Files

6.5 Disable Windows Defender (Temporary)

Open PowerShell as Administrator
Set-MpPreference -DisableRealtimeMonitoring $true

6.6 TAKE VM SNAPSHOT NOW!

VirtualBox:

1. Machine → Tools → Snapshots
2. Click "Take" icon
3. Name: "Pre-Attack Clean State"
4. Click "OK"

You can now restore to this point anytime!

7. Step 5: Running The Simulation

7.1 Start Infrastructure (Controller VM)

Open **TWO terminals** on Controller VM.

Terminal 1 - C2 Server:

- cd ~/ransom_lab
- source venv/bin/activate
- python3 c2_server.py

``` Keep this running! We should see: ```

[*] Starting on 0.0.0.0:5000

Terminal 2 - Dashboard:

- cd ~/ransom_lab
- source venv/bin/activate
- streamlit run detection_dashboard.py --server.port 8501 --server.address 0.0.0.0

``` Keep this running! You should see:

``` You can now view your Streamlit app in your browser.

Network URL: http://192.168.64.30:8501 ```

You should see the detection dashboard with:

- 0 Infected Systems
- 0 Files Encrypted
- Empty activity timeline

Dashboard is ready!

7.2 Prepare Payload (Attacker VM)

After creating the executable from lockbit_realistic.py in attacker, it will automatically create a dist folder which contains lockbit_realistic.exe file

- cd ~/ransom_lab/attacker/dist
- *# Start HTTP server to host*
- python3 -m http.server 8000

**7.3** Download Payload (Victim VM)

On victim go to the server where attacker is serving.

We are assuming the scenario that attacker sent an email to the victim saying that its realistic game and the attacker will send the link to the victim.
```

Victim believes that it’s a game and will open the link. As soon as he opens the link it will download the lockbit_realistic.exe file then victim will double click the file assuming to play the game and then the attack will happen to the laptop

We'll see warnings:

```
⚠⚠⚠ CRITICAL WARNING ⚠⚠⚠
This uses REAL AES-256 encryption!
Original files will be DELETED!
...

Continue? Type 'YES ':
```

Type exactly: `YES`

5.5 Watch What Happens! 👀

On Victim Console (15-20 seconds):

```
[Phase 1] System Reconnaissance
[*] Target System: vmuser (192.168.64.10)

[Phase 2] C2 Communication
[+] C2 Registration successful
[+] Victim ID: 85521745e04d5e8e

[Phase 3] Multi-Threaded File Encryption
[Thread-0] Encrypted: C:\lab\test_data\document1.txt
[Thread-1] Encrypted: C:\lab\test_data\document2.txt
... (continues)

[+] Encryption complete: 21 files encrypted

[Phase 4] Ransom Note Deployment
[+] Ransom note deployed

[Phase 5] Wallpaper Hijacking
[+] Wallpaper changed

[Phase 6] Persistence & C2 Monitoring
Press Ctrl+C to stop C2 monitoring
```

7.4 On Dashboard (Real-time updates**):**

- **Infected Systems:** 0 → 1 ⚡
- **Files Encrypted:** 0 → 21 ⚡
- **Threat Score:** 0 → 72.1/100 ⚡
- **Red Alert:** "🚨 CRITICAL THREAT DETECTED"

7.5 **On Victim Desktop:**

- Wallpaper changes to ransom message!
- Files now have .lockbit extension
- Ransom notes everywhere!

8. Manual Recovery: Assuming victim pays ransom and attacker provides decrypt file

On Victim VM:

- powershell
- cd C:\lab
- python decrypt_lockbit.py

Clean Up:

powershell

 # Ransom notes should be auto-removed
 # Wallpaper should be reset
 # Desktop clean

 # If anything remains:
Remove-Item C:\lab\test_data\README_LOCKBIT.txt -ErrorAction SilentlyContinue
Remove-Item $env:USERPROFILE\Desktop\README_LOCKBIT.txt -ErrorAction SilentlyContinue
Remove-Item $env:USERPROFILE\Desktop\LOCKBIT_RANSOM.bmp -ErrorAction SilentlyContinue
```

**System fully restored!**

**9.** Files and loction:
In this zip file:
Victim folder: Contains files that victim vm needs to run the attack and contains VM
Attacker folder: contains files related to the attacker and Contains VM
Controller: Contains c2_server.py and detection dashboard and Contains VM
Lockbit3.mp4: Contains demo execution steps
Report: Clear Details about the Ransomware
Readme file: Step by Step Execution of the attack
Powerpoint presentation: Contains details about the project
```