**Step 6**

**Team Members:**

**Mitigation Strategy**

Upon detecting the ransomware-like activity and the system that immediately quarantines the infected files/folders.

Suspicious files that are moved from the critical working directory to the secure quarantine location **(/home/nikitha/quarantine)** to prevent the further spread or encryption

**Implementation Details**

Detection Trigger: Detection is based on the spike in the file system events (that are creation, modification, deletion) within the short time window (10 seconds).

Threshold Rule: If the number of file events that exceeds 100 in 10 seconds and it is flagged as ransomware activity.

**Action Taken:**

Move the entire monitored folder (i.e **/home/nikitha/critical**) to the quarantine directory as well. And then Ensure the quarantine folder that is created if it doesn’t already exist.

**Code logic for Mitigation**

Quarantine logic is to be implemented inside the detect.py python file then shutil.move() is used to relocate the suspicious files safely.

If an error that occurs during moving and it is logged for the further manual investigation.

**Backup Strategy:**

We Maintain the regular offsite and offline backups of the critical files.

Backups should not be accessible from the infected machine that we run

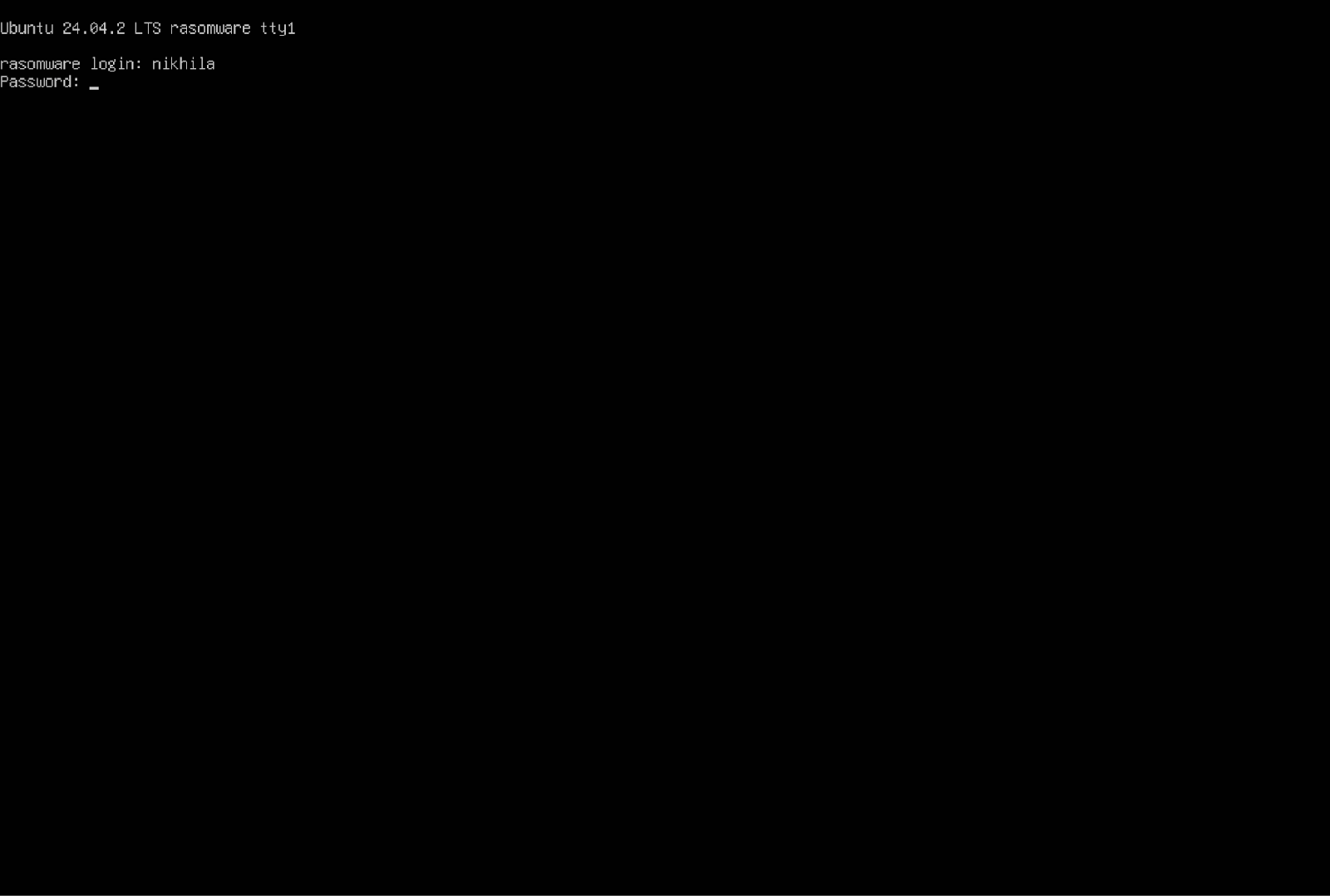
**Recovery Plan:**

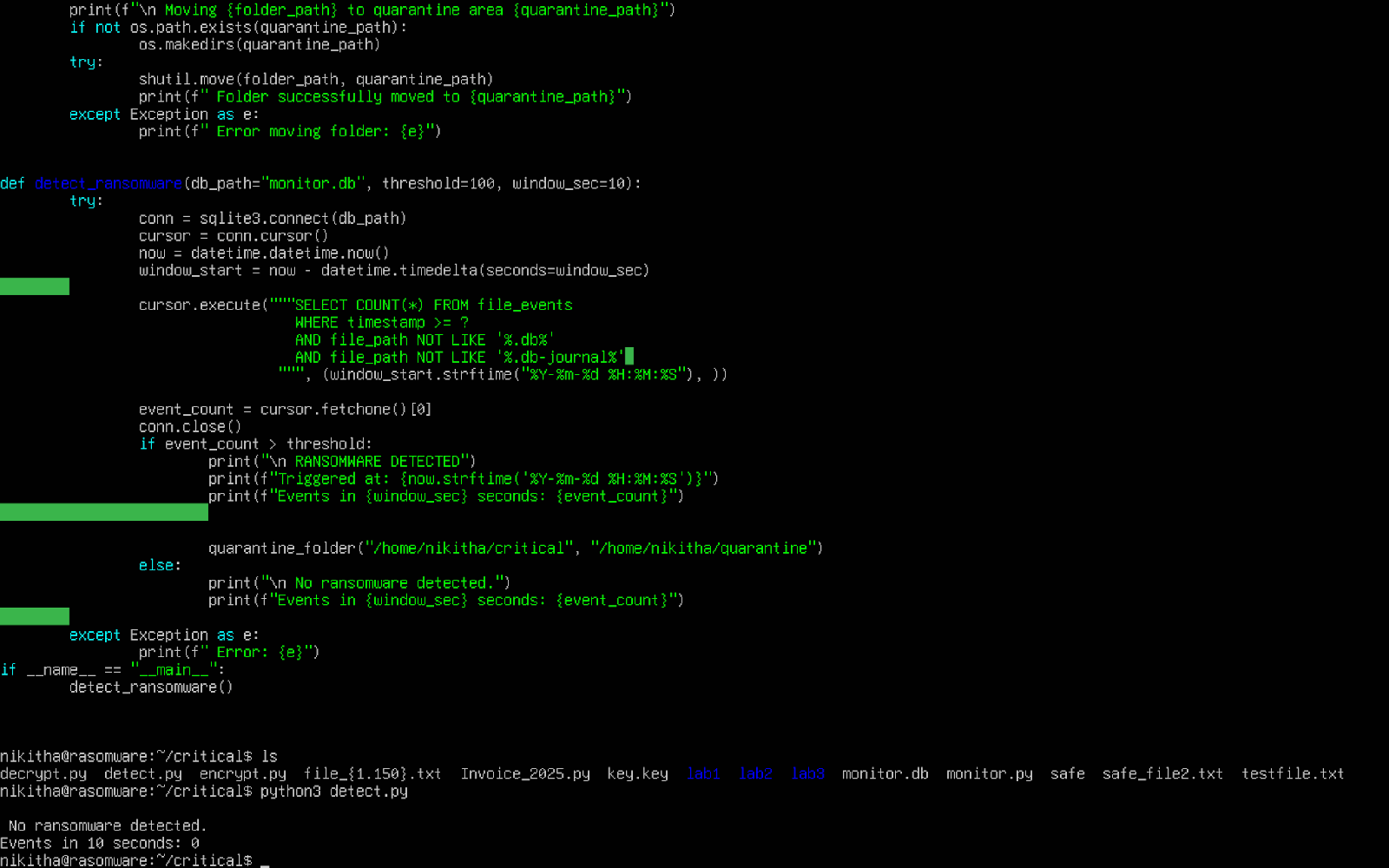
After ransomware detection and the system administrators can review the quarantined files.

If files are clean and they can be safely restored. And we also look If files are confirmed encrypted and then they are deleted after backups are verified.

**Open the UTM virtual Machine to execute the Mitigation code**

.



Here is the modified **detect.py** code with quarantine logic

And we noticed there is No ransomware detected in last 10 seconds because we don’t pass any data into the **monitor.db** database.

