Sunday, December 01, 2024

Week 3 Journal

Project Description:

The project meets original expectations by turning a Raspberry Pi into a security camera that's both reliable and secure. The camera does use open-source software for motion detection and visual storage. However, our deliverables diverge from the first solution in our proposal that is MotionEyeOs from a GitHub. Instead, we use a similar interface with MotionEye software without the OS itself on our Raspberry Pi operating system. Also, we do not install Debian OS for the second method on a SD card. As a substitute, we have installed the Raspberry Pi OS on another SD card to not encounter storage problems between the two software.

Requirements:

1- We do not use SSH authentication. The interface offers user authentication for MotionEye, by changing default settings. Motion does it with specific sudo commands such as giving permission to access the log files for only the user of the Raspberry Pi. For instance, in journal 2 we used sudo chown. Moreover, access is restricted to the local machine user running the Motion software.

2- Indeed, we used Systemd mentioned in the proposal to create a service file to manage and configure the motion service, managing the recorded videos, etc.

3- Using Shell(bash) is used to boot the system with simple commands such as to find the IP and port numbers to run the solutions on web. Storage cleaning is used with options to delete in settings for Motion Eye and pictures preserve. Nevertheless, it isn't available for Motion, unless used with bash commands to delete from the directory.

4- The detailed set-up of both solutions in the proposal is done, but with the above-mentioned software.

Timeline of deliverables:

Although we did respect the content to do in our timeline, the first week we couldn't install both methods completely with testing the new features and the usage of Raspbian. Since we tried to use ChatGPT to install Motion software, it listed us a detailed and not a simple installation, thus we encountered problems with a kind of FTP (file transfer between computers) setup that we let go for our project. MotionEyeOS recent release was of 2020 in GitHub, which made it inefficient. On the other hand, we were able to do the basic setup by the second journal.

Enhanced testing features manually for MotionEye camera

- In Video Device, changed the video resolution to 1280x720 for better quality and frame rate to 25 for better speed.
- In Text Overlay, set text scale to 2 for the size of dates on the camera captures.
- In Still images, we change data clearage for one week in preserve pictures and capture mode to motion triggered in directory.
- In Movies, the preserve movies are set for one week and movie format to H.264/V4L2M2M (.mp4)
- In Motion Detection, the set frame change threshold to 10% for motion to be detected, set Mask to on, and show frame changes to on.
- To force remove all files use command sudo rm –rf *, after being in the right directory of the files.

Websites

https://pimylifeup.com/raspberry-pi-security-camera/ for MotionEye setup.

https://shorturl.at/HygLE YouTube, for basic camera setup and Raspberry Pi, also tryouts for MotionEyeOS. The repository we referred to install, hence didn't work https://github.com/motioneye-project/motioneyeos .

https://www.youtube.com/watch?v=e27jyEcE5lU for MotionEye basic settings.

Our GitHub Repository:

https://github.com/UAlaina/EyeOnPi.git