Samaritan Security

Protecting a variety of devices for all potential users within a system, and detecting anomalous activity.

Design Document

Team Number: sdmay20-45

Client: Self (Devin Üner)

Advisor: Dr. Goce Trajcevski

Team Members:

Kate Brune
Ryan Goluch
Thomas Paris
Devin Uner
Saba Shaarbaf-Toosi
Ann Gould

Team Email: sdmay20-45@iastate.edu

Team Website: http://sdmay20-45.sd.ece.iastate.edu/

Revised: October 8th 2019 / v1.0

Table of Contents

1 Introduction

1.1 Acknowledgement

Samaritan Security is a self proposed project, and will continue to be developed in a closed source fashion.

1.2 Problem and Project Statement

When it comes to monitoring employees, companies have the option to monitor employee actions typically through different tooling systems (computer monitoring/video camera feeds), but there isn't really a product on the market to combine these security monitoring systems. Using machine vision and machine learning, Samaritan Security will be able to achieve integration of monitoring systems with the additional use of these technologies. Samaritan Security aims to achieve add additional security and threat detection by utilizing machine learning and machine vision and integration of security systems.

1.3 Operational Environment

Samaritan Security is security software designed to be hosted inside the workplace. As a piece of software, Samaritan does not encounter many operational environment hazards. In any scenario where there exists hazards that need to be overcome, the responsibility falls on the hardware to overcome the hazard. This could include the computer or server to have components to keep heat management under control or for out-door cameras to work in rainy conditions. The software itself does not have environmental hazards.

1.4 Requirements

Samaritan can be broken up into four main functional requirements: Compiling information about actors using past and current records, identify people and recognize actions that deviate from learned norms, Automatically identify and use new data sources, and using data to make future predictions about the threat level of actors. Compiling information comes from records from the company such as employee list, clearance levels assigned to employees, misconduct of employees, watch lists, etc.Samaritan uses web scraping to collect information about actors it detects to collect information about potential security risks. Using this data along with facial recognition software, the system will make decisions about the potential threat level of actors it detects.

1.5 Intended Users and Uses

This product is intended for companies to use to monitor employees and employee actions throughout the work day/within a specific area. Companies can monitor employee usage of proprietary equipment/data to ensure employees are compliant with industry/company standards for data usage or technology usage.

1.6 Assumptions and Limitations

Assumptions:

- Pre-existing video feeds are integrated and connected to server
- Servers are up and running and able to communicate between video and website
- Video feeds show live and accurate video
- Clients have the ability to capture audio or will be adding that capability
- A willingness to use this product

Limitations:

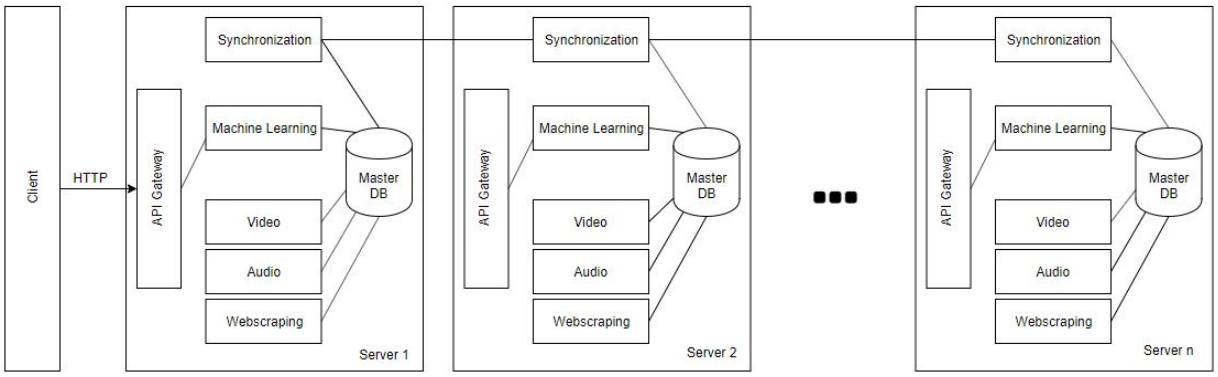
- A limitation we have is keeping users' privacy in mind while also working to build a security system that keeps the user safe at all times.
- A second limitation is the accuracy of our video feeds, as we are using preexisting video sources, so we will not be able to customize the cameras or video feeds and their capabilities.
- Clients having a high enough bandwidth network that allows our services to run properly

1.7 Expected End Product and Deliverables

By the end of our senior design project, Samaritan should be capable of showing all places a specified person has been, along with all known information about them (known associates, average heart rate, height, voice recordings, etc). Samaritan should also be capable of automatically identifying potential threats to overall security and automatically alert security resources of all threats.

2 Specifications and Analysis

2.1 Proposed Design



2.2 Design Analysis

The current design uses multiple data sources such as a video and audio feeds, inputted files, and the web to collect information which is then piped to a machine learning algorithm to determine the threat of an actor. The design choice of having multiple data feeds for the algorithm to take in allows for more sources to be added in the future. This allows Samaritan to be able to adapt in the presence of new information to make the best possible choices. A potential risk of Samaritan is that is purely software. This means that it the quality of the video feed is partially dependant on the cameras chosen by the client.

2.3 Development Process

Samaritan Security is developed using an agile development process. The development of the project is broken down into smaller parts to ensure continuous progress. Working in iterative sprints, the team can easily evaluate the project and adapt to needed changes quickly and efficiently. At the end of each sprint, the reviews and comes up with a plan to improve the project and personal work so that the team continues to strive for excellence.

2.4 Design Plan

The current design is based around several use cases. We want users to be able to view camera and audio data related to detected threats, so our design has modules to handle video and audio feeds, as well as a machine learning module to identify threats in these feeds. Additionally, users can view information about an entity, so we will have a module to perform web scraping. Lastly, users are able to track entities, so we will use our different modules and a master database to store and query different entities in the environment.