Software and Computer Security

SOFE 4848U

Phase 2

Date: March 15, 2023

Project Group: Group 17

Student Name	**Student Number**
Carson McClelland	100725653
Shwan Majeed	100749077
Walid Ayub	100695612
Kalid Ajibade	100660188

Tools: Linux, Python, Clamav- daemon, Pyclamd and EICAR test

Linux - The linux operating system has many capabilities such as it being open source, this offers us much more flexibility when implementing different tools and functionalities because it's modular in design. It is known for high security which relates back to our main objective of the project. This is due to the nature of it needing authorization from the administrator to execute applications, it's not perfect but its security is better than many other operating systems.

Python - We choose python as our programming language because it's open source and is known to be useful in cybersecurity due to its data analysis abilities. It also has a vast library that has some tools for security. It is also easier to use and we won't have to deal with memory management or garbage collection allowing us to focus solely on the functionality of our application.

Clamav-daemon - Is a tool that is essential in ensuring the security of our usb, its an open-source antivirus engine that is capable of detecting viruses, malware, trojans and many other security threats. It is said to have high performance, offering services such as a multi-threaded scanner, automatic signature updates and command-line utilities for file scanning. It is also versatile, being able to scan a multitude of file formats and signature languages. So this tool will be critical in being able to identify threats within each usb device. It is also compatible with most operating systems. ClamAV uses a combination of signature-based scanning and heuristic analysis to detect malware in files. Signature-based scanning involves comparing the contents of a file to a database of known virus signatures. Heuristic analysis involves looking for suspicious behavior or characteristics that may indicate the presence of malware.

Pyclamd - is a Python interface for ClamAV's scanning daemon, it allows Python applications to interact with ClamAV and perform virus scanning on files.

EICAR test - The EICAR test file is a commonly used standard file for verifying the proper functioning of antivirus software. It comprises a single line of text that is intentionally created to be detected as a virus signature by antivirus scanners.

The string of characters used is:

```
X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*
```

How they work in conjunction:

Diagram 1: UML Sequence diagram

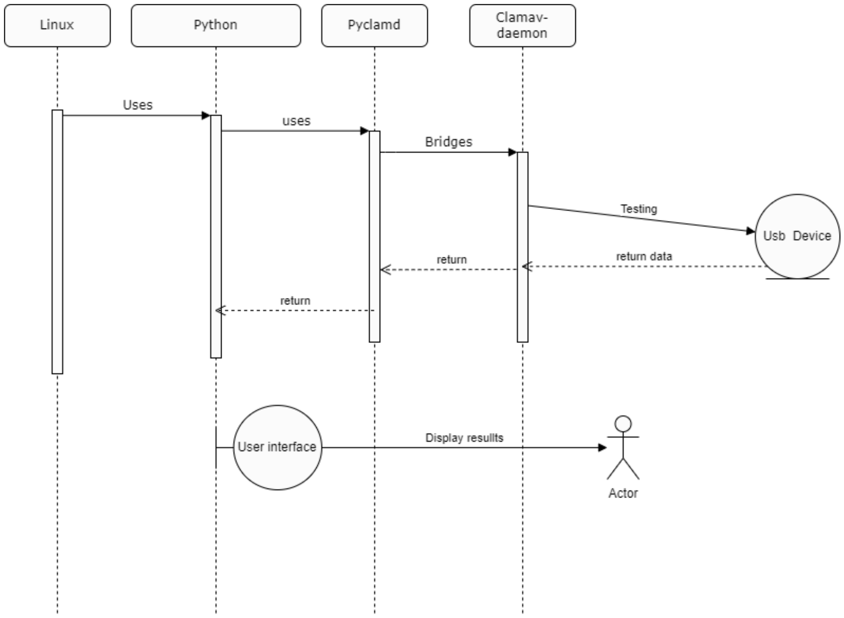
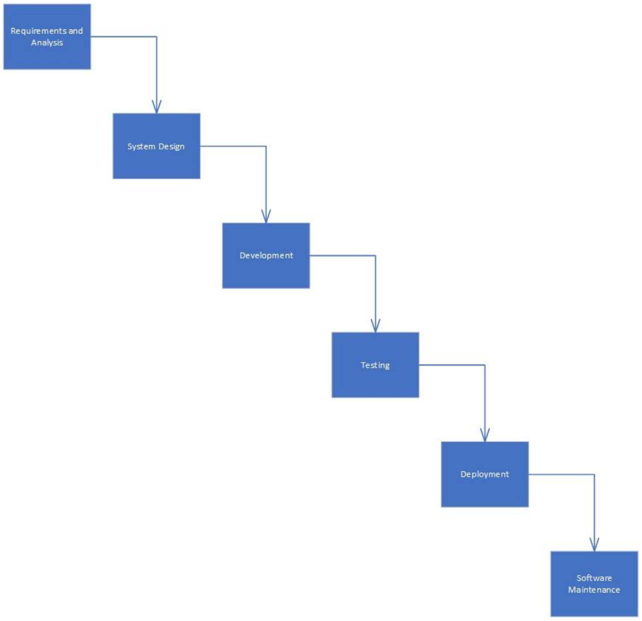


Diagram 2: Waterfall design process



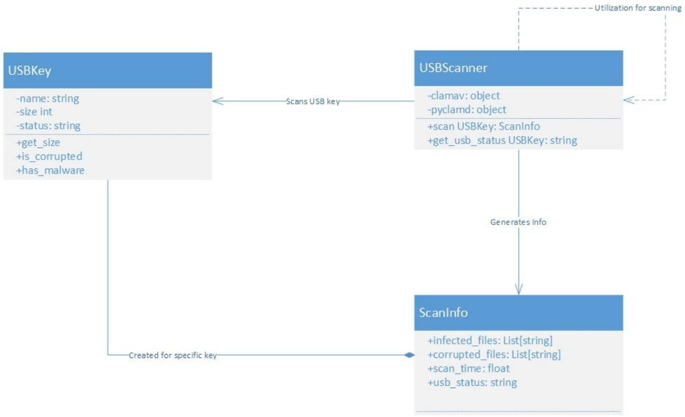
6 Steps to this design process:

1. Outline the requirements - which are, to be able to scan the USB keys for any potential risks and provide all the details of what was found.
2. Design of the architecture - the web app needs to be able to know whether a USB key has been connected, so that the ClamAV can scan the data for any corrupted files and provide all its findings
3. Developing the application - this involves all the code to be completed within the system so that the ClamAV can run its scans on the USB

4. Testing the application - will require multiple tests under certain conditions to determine if the required application features are running successfully
5. Deploying the application - if the application is running properly with all requirements met, the application can be deployed for users to try out
6. Maintenance (optional) - to fix any small bugs that come over time and to make sure it is always running smoothly along with extra features to be added if required.

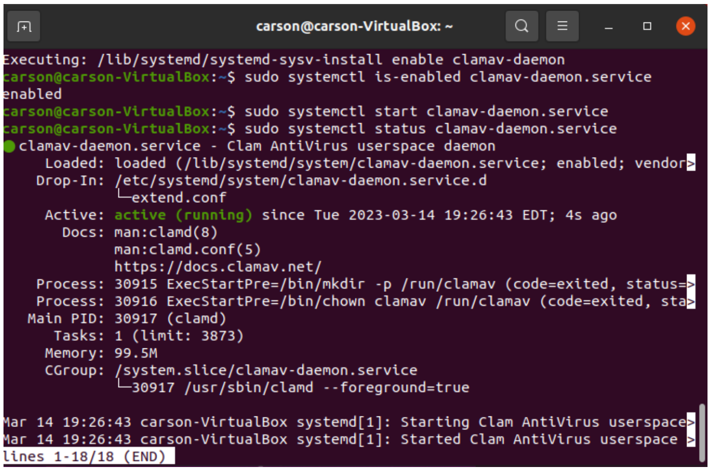
Diagram 3: UML Class diagram

This class diagram shows the design for a web app that scans USB keys and notifies users with all required information determining whether it has any potential risks or not along with other information given.



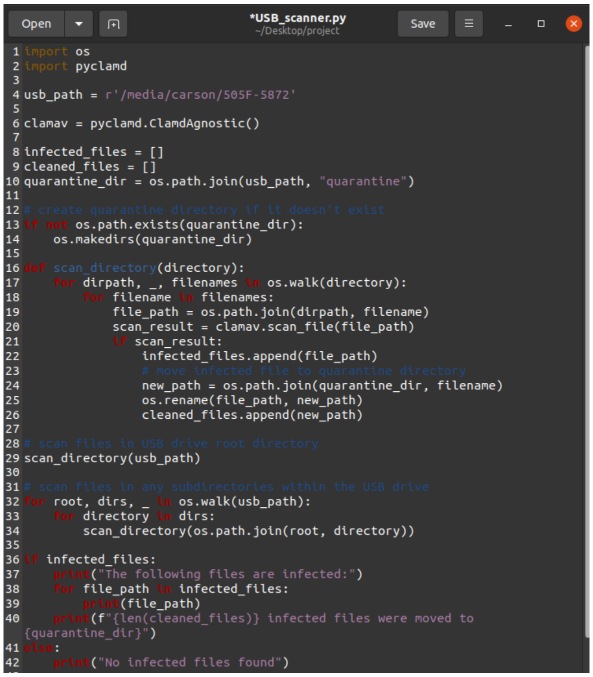
3 Classes:

1. USBKey - which represents the USB key that is used to check for properties such as name, size and status for its health whether it has viruses or not
2. USBScanner - which represents the web app that scans the USB keys using ClamAV and Pyclamd for scanning and determining its status
3. ScanInfo - is the information given to the user of what the USB scanner was able to retrieve such as a list of infected or corrupted files, along with how long it took and the current status of the USB's health.



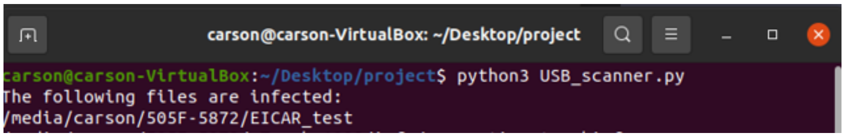
```
Executing: /lib/systemd/systemd-sysv-install enable clamav-daemon
carson@carson-VirtualBox:~$ sudo systemctl is-enabled clamav-daemon.service
enabled
carson@carson-VirtualBox:~$ sudo systemctl start clamav-daemon.service
carson@carson-VirtualBox:~$ sudo systemctl status clamav-daemon.service
● clamav-daemon.service - Clam AntiVirus userspace daemon
     Loaded: loaded (/lib/systemd/system/clamav-daemon.service; enabled; vendor>
    Drop-In: /etc/systemd/system/clamav-daemon.service.d
             └─extend.conf
     Active: active (running) since Tue 2023-03-14 19:26:43 EDT; 4s ago
       Docs: man:clamd(8)
             man:clamd.conf(5)
             https://docs.clamav.net/
    Process: 30915 ExecStartPre=/bin/mkdir -p /run/clamav (code=exited, status=>
    Process: 30916 ExecStartPre=/bin/chown clamav /run/clamav (code=exited, sta>
   Main PID: 30917 (clamd)
      Tasks: 1 (limit: 3873)
     Memory: 99.5M
     CGroup: /system.slice/clamav-daemon.service
             └─30917 /usr/sbin/clamd --foreground=true

Mar 14 19:26:43 carson-VirtualBox systemd[1]: Starting Clam AntiVirus userspace>
Mar 14 19:26:43 carson-VirtualBox systemd[1]: Started Clam AntiVirus userspace >
lines 1-18/18 (END)
```

In the image above, we installed and ran the clamav-daemon.service, which is the antivirus software in which we are using. By running this program, it will be able to detect viruses within the USB. We then ran a status command to ensure that the program being used is up and running. As we can see in the screenshot above, the status of the program is active and running, with one task in use.



```python
import os
import pyclamd

usb_path = r'/media/carson/505F-5872'

clamav = pyclamd.ClamdAgnostic()

infected_files = []
cleaned_files = []
quarantine_dir = os.path.join(usb_path, "quarantine")

# create quarantine directory if it doesn't exist
if not os.path.exists(quarantine_dir):
    os.makedirs(quarantine_dir)

def scan_directory(directory):
    for dirpath, _, filenames in os.walk(directory):
        for filename in filenames:
            file_path = os.path.join(dirpath, filename)
            scan_result = clamav.scan_file(file_path)
            if scan_result:
                infected_files.append(file_path)
                # move infected file to quarantine directory
                new_path = os.path.join(quarantine_dir, filename)
                os.rename(file_path, new_path)
                cleaned_files.append(new_path)

# scan files in USB drive root directory
scan_directory(usb_path)

# scan files in any subdirectories within the USB drive
for root, dirs, _ in os.walk(usb_path):
    for directory in dirs:
        scan_directory(os.path.join(root, directory))

if infected_files:
    print("The following files are infected:")
    for file_path in infected_files:
        print(file_path)
    print(f"{len(cleaned_files)} infected files were moved to {quarantine_dir}")
else:
    print("No infected files found")
```

This image above shows the python program that will run to detect infected files within a USB. There are two types of files that the program is looking for, which are infected files and cleaned files. The program scans all the files within the directory and if it's determined that the file is infected, the program will list all the files that are in security risk to inform the user and move the files to a quarantine directory to isolate the files from additional harm.



```
carson@carson-VirtualBox:~/Desktop/project$ python3 USB_scanner.py
The following files are infected:
/media/carson/505F-5872/EICAR_test
```

In the image above, the program to scan a USB was run. As we can see, the program went through all of the files within the directory, and using the ClamAV service, the scanner detected the EICAR test file and moved it to a created quarantine directory.

In conclusion, the program provides a simple and effective way to scan for viruses on a USB drive using the ClamAV antivirus engine. Its straightforward design allows users to scan for infected files quickly and easily, and the ability to move any infected files to a quarantine folder provides added security. Additionally, the use of an open-source antivirus engine such as ClamAV ensures that the virus scanning solution is reliable and up to date. One potential improvement for this code could be to add error handling for cases where the USB drive or quarantine folder is not accessible. Overall, this program can be a useful tool for individuals or organizations that need to scan USB drives for viruses. This course has provided a comprehensive overview of key security concepts and strategies that we were able to use and implement in our project.