**Requirements**

ID Number: 01

Type: Functional

Description: The Kali virtual machine should have access to the internet

Rationale: Without internet access, the Kali machine could not act as a router as it cannot forward and log traffic.

Fit Criterion: The system should be able to connect to the internet.

Priority: High

Dependencies: None

ID Number: 02

Type: Functional

Description: The Airgeddon Evil Twin should be able to launch and make a fake access point.

Rationale: To have users connect to the AP it needs to be launched.

Fit Criterion: Airgeddon launches the twin succesfully.

Priority: High

Dependencies: None

ID Number: 03

Type: Functional

Description: The system should be able to intercept network traffic passing through the evil twin.

Rationale: Intercepting network traffic is essential for analyzing and logging packets.

Fit Criterion: The system should capture and forward packets to the analysis module without dropping more than 1% of packets.

Priority: High

Dependencies: 01, 02

ID Number: 04

Type: Functional

Description: The Python analysis module should be able to parse packet headers and payloads for encryption, only HTTP and HTTP packets.

Rationale: Parsing packet data allows for identifying potential unencrypted packets.

Fit Criterion: Packets should be checked for encryption with 90% accuracy.

Priority: High

Dependencies: 03

ID Number: 05

Type: Functional

Description: The system should be able to log unencrypted packets and where the packets were sent.

Rationale: Logging unencrypted packets for counting and analysis.

Fit Criterion: Packets should be checked and logged within 100 milliseconds.

Priority: Medium

Dependencies: 01, 02, 03, 04

ID Number: 06

Type: Functional

Description: The system should be able to count all unencrypted or encrypted packets.

Rationale: Keeping count of all packets so that analysis will be more straightforward.

Fit Criterion: Counts should be made legibly and concisely.

Priority: Medium

Dependencies: 04, 05

ID Number: 07

Type: Security

Description: After checking for encryption, the system should log the website URL and hash but delete all data afterward.

Rationale: Deletion of data to ensure no information is kept after analysis.

Fit Criterion: Data should be deleted after checking for encryption.

Priority: High

Dependencies: None

ID Number: 08

Type: Usability

Description: The Python analysis module should be able to provide a user-friendly command-line interface for configuration and monitoring.

Rationale: A user-friendly interface simplifies system management and usage.

Fit Criterion: Users should be able to configure and monitor the analysis module with minimal knowledge.

Priority: Medium

Dependencies: None

ID Number: 09

Type: Performance

Description: The system will be able to process at least 500 packets per second.

Rationale: The system must handle network traffic efficiently to detect encryption and allow packets to flow freely without bottlenecking them.

Fit Criterion: The system should maintain a packet processing rate of at least 500 packets per second under normal load conditions.

Priority: High

Dependencies: 01, 02

ID Number: 10

Type: Functional

Description: The system should be able to provide the capability to blacklist certain websites.

Rationale: Customizable rules allow for blacklisting specific sites so that there is no chance for personal information to be gotten.

Fit Criterion: The system should be able to deny access to websites such as online banks, college portals, hospital databases, etc.

Priority: High

Dependencies: 07

ID Number: 11

Type: Usability

Description: The system should be able to generate reports on all internet traffic for further analysis.

Rationale: Detailed reports aid in the analysis after the fact.

Fit Criterion: Reports should include website URLs, encryption, and counts.

Priority: Medium

Dependencies: 05, 06, 07

ID Number: 12

Type: Performance

Description: The system should be able to use low resource utilization to minimize impact on network performance.

Rationale: High resource utilization can degrade network performance and user experience.

Fit Criterion: Resource utilization (CPU, memory, disk I/O) should remain below 50% under normal operating conditions.

Priority: Medium

Dependencies: 01, 02, 03

ID Number: 13

Type: Functional

Description: The system should be able to support automatic software patch updates.

Rationale: Regular updates are essential for software to work efficiently and as expected.

Fit Criterion: The system should check for updates periodically and apply them automatically with user approval.

Priority: Low

Dependencies: 08

ID Number: 14

Type: Security

Description: The system should maintain an audit trail of configuration changes and system activities.

Rationale: Audit trails are essential for compliance, forensic analysis, and accountability.

Fit Criterion: There should be a log of all configuration changes, data access, and system events with timestamps.

Priority: High

Dependencies: 09

ID Number: 15

Type: Performance

Description: The system should be able to maintain high availability to ensure continuous operation.

Rationale: High availability minimizes downtime.

Fit Criterion: The system should have a minimum uptime of 99.9% over a one-day period.

Priority: High

Dependencies: None

ID Number: 16

Type: Usability

Description: The system should be able to provide intuitive error messages for configuration errors or system failures.

Rationale: Clear error messages help Users troubleshoot issues efficiently.

Fit Criterion: Error messages should accurately describe the problem and suggest possible solutions.

Priority: Medium

Dependencies: None

ID Number: 17

Type: Performance

Description: The system should provide real-time monitoring and alerting capabilities with minimal latency.

Rationale: Real-time monitoring and alerting enable timely detection and response to security incidents.

Fit Criterion: Alerts should be generated and delivered to users within 1 second of detecting an error.

Priority: Medium

Dependencies: None

ID Number: 18

Type: Functional

Description: The system can parse both UDP and TCP packets.

Rationale: Being able to parse and analyze both kinds of packets will allow the system to be more accurate with its logging.

Fit Criterion: The system should be able to parse and analyze TCP and UDP packets.

Priority: High

Dependencies: 02, 03

ID Number: 19

Type: Usability

Description: The Python script will be written in a way that is easy to read and has comments.

Rationale: For both the user and others, the code should be easy to read so everyone can understand what each part does.

Fit Criterion: The Python code should be easily understood even by those with little to no Python experience.

Priority: Low

Dependencies: 03

ID Number: 20

Type: Usability

Description: The Python script must be able to run.

Rationale: The code not running would defeat the whole purpose.

Fit Criterion: The code should be able to run.

Priority: High

Dependencies: 03