Software Requirements Specification (SRS)

1. Introduction

1.1 Purpose

Hyde is a type of **Evil Twin attack**. It pretends to be a trusted network to fool unsuspecting users to connect to the fake network in order to attack and steal sensitive information such as passwords, email etc. and even gain access to settings and systems.

The purpose of this device is to educate and raise awareness to the people who use public networks unsuspectingly in public places. This device has very low limitations as to what damage it can do both to your devices and to you yourself once you are connected.

This study aims to show you as to how easy it is to attack your devices using Evil Twin and the damage it could possibly deal to both your device and being. This project will raise awareness and share tips on how to avoid being a victim of this kind of attack.

Note that this is for educational purposes only and does not intend to break any forms of law corresponding to cybersecurity nor to deal damage in any form, in any kind of device and persons unless agreed upon and permitted.

1.2 Scope

Hyde can attack any unsuspecting user that connects to the spoofed network created. One of its distinguished features is its ability to be able to attack remotely by using a laptop as a server and can also be piloted through the use of a phone application that the researchers plan to develop..

1.3 Definitions, Acronyms, and Abbreviations

Hyde - is a reference to a novel written by Robert Louis Stevenson titled "The Strange Case of Dr. Jekyll and Mr. Hyde

Evil Twin Attack - is a Wi-Fi hacking technique that tricks the user into connecting to a spoofed targeted network, making it nearly impossible to determine whether the network is real or fake, resulting in the user entering their password in the fake network hosted by the Hacker.

GUI - stands for Graphical User Interface. GUI is the interface that uses graphical elements to let people interact as per requirement with electronic devices including computers, laptops, tablets, and smartphones.

SSID - Service Set Identifier; the name of a Wi-Fi network.

DNS - Domain Name System; translates domain names into IP addresses.

MitM - Man-in-the-Middle; an attack where the attacker intercepts communication between two parties.

1.4 References

Evil Twin in Kali Linux - GeeksforGeeks

The Strange Case of Dr. Jekyll and Mr. Hyde | Summary, Characters, & Facts | Britannica

1.5 Overview

This document outlines the software requirements for "Hyde," an educational Evil Twin attack simulation tool. It covers the overall description, specific requirements, external interface requirements, system features, and other non-functional requirements.

2. Overall Description

2.1 Product Perspective

This form of attack has been present for a long time now ever since we went online. Hackers build their own hardware to automate or simulate an evil twin attack. However, there is no correct way of building one since the limits are not set on stone. Meaning you can build a device that uses evil twin depending on your goals as the limit.

The researchers aim to build their own hardware device through the use of pre-existing microcontrollers and other hardware pieces and modules. The main features this project offers is that it has the ability to target victims all while being discrete as the hacker uses only a phone that has remote control to the server. This can be achieved through application development with GUI.

2.2 Product Features

Mimicry:

- SSID Spoofing: The attacker uses the same network name (SSID) as a legitimate Wi-Fi network, often a popular one in the area (e.g., "CoffeeShop Free Wi-Fi"). This tricks users into thinking they're connecting to the real network.
- Stronger Signal: The attacker might boost the signal strength of their rogue access point, making it more enticing for devices to connect automatically.

Deception:

- Captive Portals: Attackers often set up fake captive portals that mimic those used by legitimate hotspots. These portals might ask for login credentials, personal information, or even payment details.
- DNS Spoofing: The attacker can manipulate DNS settings to redirect users to fake websites that look like legitimate ones (e.g., a fake online banking site).

Man-in-the-Middle (MitM) Positioning:

- Traffic Interception: By sitting between the user and the real network, the attacker can intercept all data transmitted, including sensitive information like passwords and credit card numbers.

- Data Manipulation: The attacker can not only capture data but also modify it in transit. This could be used to inject malware into downloads or alter online transactions.

Accessibility and Portability:

- Easy Setup: With readily available software and affordable hardware like Wi-Fi pineapples or even smartphones, creating an evil twin is relatively easy for attackers.
- Mobile Hotspots: Attackers can create evil twins using portable devices, making the attack possible in various locations and increasing its reach.

Difficulty of Detection:

- Hidden in Plain Sight: Evil twins often blend in with legitimate networks, making it challenging for users to identify them without careful inspection.
- No Obvious Warnings: Unlike some malware or phishing attacks, evil twins often don't trigger immediate red flags, making users less likely to suspect anything is amiss.

2.3 User Classes and Characteristics

- **Primary User:** Individuals with a basic understanding of networking concepts who are interested in learning about cybersecurity threats. This includes students, educators, and technology enthusiasts. These users will primarily interact with the device to observe the attack process and understand its implications.
- **Secondary User:** More technically proficient individuals, such as security professionals or penetration testers, who may use the device for training or demonstration purposes. These users may have a deeper understanding of network security and may be interested in exploring the device's capabilities in more detail.
- **Security Professionals:** Utilize "Hyde" for training and awareness purposes within organizations.

2.4 Operating Environment

Hardware: ESP 8266 microcontroller (specific model to be determined based on processing and storage needs), Wireless Network Adapter with monitor mode capability, Battery Pack (for portability)

Operating System: Kali linux (chosen for its pre-installed penetration testing tools)

Third-party software: Hostapd (for access point creation), dnsmasq (for DNS spoofing), Aircrack-ng suite (for network monitoring and attack execution), Python (for application development).

2.5 Design and Implementation Constraints

- The device should be low-cost and utilize readily available components.
- The mobile application must be user-friendly and intuitive.
- The device must not be used for illegal activities or to cause harm.

2.6 Assumptions and Dependencies

- Users have basic knowledge of networking concepts.
- Users have access to a Wi-Fi-enabled device for testing.
- The device will be used in a controlled environment for educational purposes.

3. Specific Requirements

3.1 Functional Requirements

- **Network Scanning:**
  - Scan for available Wi-Fi networks and display their SSIDs and signal strengths.
  - Allow the user to select a target network to mimic.
- **Access Point Creation:**
  - Create a fake access point with the same SSID as the target network.
  - Optionally, broadcast a stronger signal than the target network.
- **Captive Portal:**
  - Present a fake captive portal to users who connect to the fake access point.
  - Capture user credentials entered into the captive portal.
  - Optionally, redirect users to a legitimate website after capturing credentials.
- **DNS Spoofing:**
  - Configure a DNS server to redirect users to fake websites.
  - Allow the user to specify which domains to spoof and the corresponding fake websites.
- **Traffic Interception:**
  - Capture all data transmitted between the user and the internet.
  - Display intercepted data in a readable format.
- **Data Manipulation (Optional):**
  - Allow the user to modify data in transit.
  - Implement safeguards to prevent malicious use of this feature.
- **Mobile Application Control:**
  - Start and stop the Evil Twin attack remotely from the mobile application.
  - Monitor the status of the attack and connected users.

    - View captured data and network traffic.
    - Configure attack settings (e.g., target network, captive portal options, DNS spoofing).

3.2 User Interfaces

- **Mobile Application GUI:**
    - A clean and intuitive interface with clear instructions.
    - Display of available Wi-Fi networks with relevant information (SSID, signal strength, security).
    - Easy selection of the target network.
    - Configuration options for captive portal and DNS spoofing.
    - Real-time display of connected users and captured data.
    - Secure authentication to prevent unauthorized access to the device.

3.3 Hardware Interfaces

- The software will interact with the Wi-Fi adapters to scan for networks, create the fake access point, and capture network traffic.

3.4 Software Interfaces

- The software will utilize hostapd to create the fake access point.
- The software will utilize dnsmasq for DNS spoofing.
- The software may integrate with other open-source tools for network analysis and manipulation.

3.5 Communication Interfaces

- The mobile application will communicate with the server application over Wi-Fi.
- The communication protocol will be HTTPS to ensure secure data transmission.

3.6 Performance Requirements

- The device should be able to handle a reasonable number of connected users simultaneously (e.g., 10-20).
- The mobile application should have a responsive and lag-free interface.

3.7 Security Requirements

- The device should not be accessible to unauthorized users.
- Captured data should be stored securely and only accessible to authorized personnel.
- The device should not be used to perform illegal activities or cause harm.

3.8 Reliability Requirements

- The device should be stable and reliable during operation.
- The software should handle errors gracefully and prevent crashes.

3.9 Maintainability Requirements

- The software should be modular and well-documented to facilitate maintenance and updates.
- The code should adhere to coding standards for readability and consistency.

3.10 Portability Requirements

- The software should be portable to different Raspberry Pi models or similar microcontrollers.