ELEC-H404

# Advanced Security Evasion in Windows

**Andranik Voskanyan**
**Cédric Sipakam**
**Zinar Mutu**

Professor
.........

Academic Year
2024 - 2025

Faculty
Electrical Engineering

# Contents

# List of Figures

## Abstract

This project investigates the efficacy of Windows security solutions, primarily focusing on Windows Defender, in detecting and responding to advanced stealthy attack techniques. The study evaluates a range of common attacker methodologies including the deployment of keyloggers, creation of backdoors for remote access, non-visual command execution leveraging built-in system tools, process injection for memory manipulation, and various persistence mechanisms designed to maintain unauthorized access. The evaluation involved executing these attack scenarios in a controlled environment while meticulously logging system behavior, network activity, and Windows event logs to analyze the detection capabilities and response of the security software.

# 1 Introduction

## 1.1 Problem Statement and Motivation

The Landscape of cyber threats is constantly evolving, with attackers developing increasingly sophisticated techniques to evade detection by security systems. Detecting these stealthy attacks is a significant challenge for individuals and organization alike. Understanding the methods attackers use to bypass security measures is crucial for defenders to improve their strategies, tools, and overall security posture. This project aims to shed light on these evasion techniques within the Windows operating System, a prevalent target for cyber-attacks.

## 1.2 Project Aims and Objectives

The Primary aims of this project are:

- To evaluate the detection capabilities of Windows Defender, [...], and [....] against a set of specific attacker techniques
- The attacker techniques investigated include:
  - Keylogger deployment
  - Backdoor creation and remote access
  - Non-visual command execution and process hiding
  - Process injection and memory manipulation
  - Persistence techniques
- To analyze system behavior, network activity, and event logs during these simulated attacks to understand how security solutions respond and what artifacts are generated
- To asses the effectiveness of various evasion methods employed by attackers

## 1.3 Scope of the project

This project focuses on:

- **Operating System:** Windows 11 Home
- **Primary Security Solution:** Windows Defender, [...], [....]
- **Attacker Tools:** A combination of publicly available tools: Microsoft Windows Command Prompt, Netcat/Nmap, Python.
- **Exclusions:** This study does not cover all possible evasion techniques or every security product available on the market. The focus is on the selected methods.

# 2 Background

## 2.1 Fundamentals of Security Evasion

Security Evasion refers to the set of techniques and strategies employed by attacker to avoid detection by security mechanism such as antivirus software, Endpoint Detection and Response Solutions (EDR), Intrusion Detection/Prevention Systems (IDS/IPS), and firewalls. The primary goal of evasion is to allow malicious activities to proceed unnoticed, enabling attackers to achieve their objectives, which could range from data heft and espionage to system disruption or financial gain. Attacker motivations are diverse but often include maintaining stealth to ensure long-term access a.k.a persistence, escalating privileges to gain deeper system control, and exfiltrating sensitive information without triggering alarms.

## 2.2 Overview of Windows Security Architecture

The Windows Operating System incorporates a multi-layered security architecture designed to protect against a wide array of threats.

- **Windows Defender Antivirus:** This is the built-in anti-malware solution in Windows. Its features include:
  - Real-time scanning
  - Behavior monitoring
  - Anti-malware Scan Interface (AMSI)
  - Cloud-delivered protection
  - Network Inspection System (NIS)
  - Controlled Folder Access
- **Windows Event Logging:** The OS records a wide variety of events related to system, security, application, PowerShell, and other application. Specific event IDs can indicate suspicious activities, login attempts, process creation, and security policy changes.
- **User Account Control (UAC):** This helps prevent unauthorized changes to the system by prompting for permission or an administrator password before performing actions that could potentially affect the computer's operation or security.
- **Windows Firewall:** Controls network traffic flowing in and out of the system, based on configured rules.
- **BitLocker Drive Encryption:** Provides full-disk encryption to protect data at rest
- **AppLocker/Windows Defender Application Control (WDAC):** Allows administrators to control which applications and files users can run.

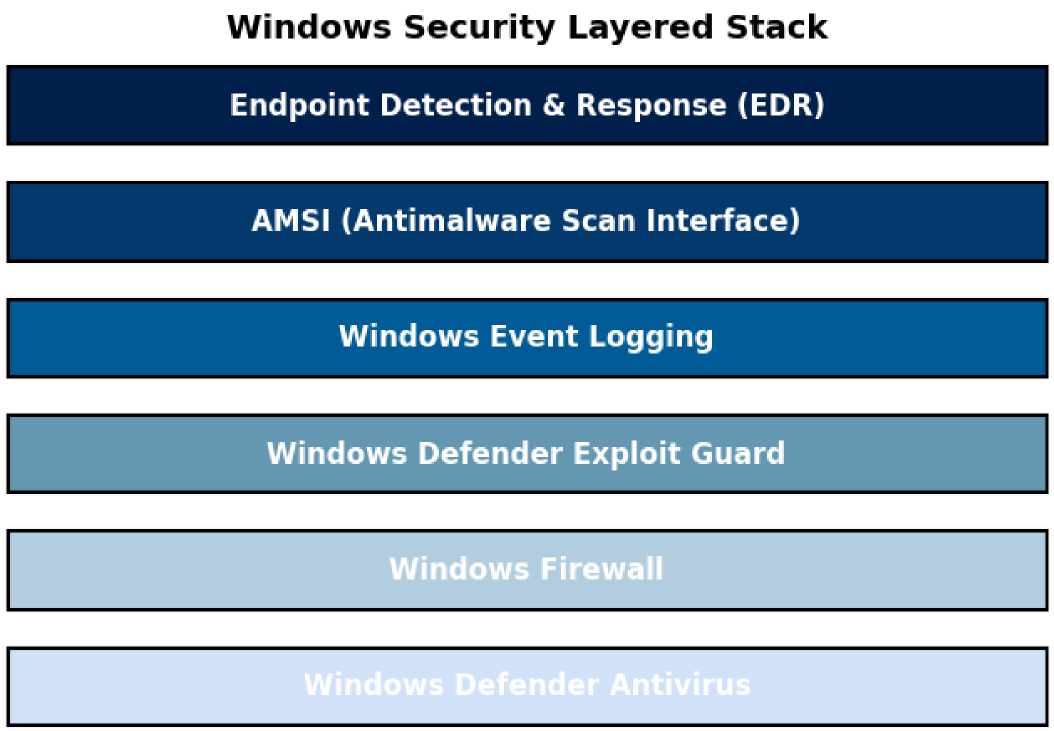
Windows Security Layered Stack
Endpoint Detection & Response (EDR)
AMSI (Antimalware Scan Interface)
Windows Event Logging
Windows Defender Exploit Guard
Windows Firewall
Windows Defender Antivirus

Figure 1: Components of the Windows Security Architecture

## 2.3 Theoretical Overview of Attacker Techniques Investigated

### 2.3.1 Keyloggers

### 2.3.2 Backdoors and Remote Access

### 2.3.3 Non-Visual Command Execution and Process Hiding

### 2.3.4 Process Injection and Memory Manipulation

### 2.3.5 Persistence Techniques

## 2.4 Relevant Frameworks and Tools

# 3 Description

## 3.1 Test Environment Setup

### 3.1.1 Victim Machine

### 3.1.2 Attacker Machine

### 3.1.3 Network Configuration

## 3.2 Security Solution Configuration