IDS.py - Simple Network Intrusion Detection System

ids.py is a Python-based script designed to detect common network intrusion attempts by analyzing live network traffic or captured .pcapng files. It leverages the Scapy library for packet sniffing and parsing.

Features

- **Logging:** All alerts are logged to a timestamped file in the logs/ directory.
- **Stealth Scan Detection:** Identifies NULL, FIN, XMAS, and SYN scans (with basic rate-limiting for SYN scans to prevent excessive alerts).
- **Nikto Scan Detection:** Detects the presence of the "Nikto" string in packet payloads, indicating a potential Nikto web scanner.
- **Shellshock Exploit Detection:** Identifies common patterns associated with Shellshock Bash vulnerability exploit attempts in packet payloads.

How it Works

The script operates by sniffing network packets and passing them to a `packhandler` function. This function then calls various detection modules:

- **`detect_stealth_scan(pkt)`**: Examines TCP flags to identify various stealth scan types (NULL, FIN, XMAS). For SYN scans, it tracks the frequency of SYN packets from a source IP within a defined time window (SYN_WINDOW_SECONDS) and triggers an alert if the count exceeds SYN_THRESHOLD.
- **`detect_nikto(pkt)`**: Checks the raw payload of packets for the presence of the byte string `b"Nikto"`.
- **`detect_shellshock(pkt)`**: Decodes the raw payload and looks for the characteristic Shellshock patterns `() {` and `; };`.

All detected events trigger an `alert()` function, which prints the alert to the console and logs it to a file with a unique alert ID.

Usage

This script can be run in two modes:

1. **Live Sniffing:** Sniff packets directly from a specified network interface.
2. **Pcap File Analysis:** Read and analyze packets from a .pcapng file.

Prerequisites

- Python 3.x
- Scapy library (`pip install scapy`)
- Administrative/root privileges may be required for live sniffing.

Running the Script

Navigate to the directory containing `IDS.py` in your terminal.

1. Live Sniffing

To sniff live traffic from a network interface (e.g., Ethernet, Wi-Fi, eth0, wlan0), use the `-L` argument:python3 ids.py -L [Network Interface Name]

Example (Windows):python3 IDS.py -L Ethernet

Example (Linux/macOS):sudo python3 ids.py -L eth0

(Note: Use `sudo` on Linux/macOS for network interface access)

2. Pcap File Analysis

To analyze a captured .pcapng file, use the `-r` argument:python3 ids.py -r [path/to/your/capture.pcapng]

Example:python3 ids.py -r /path/to/my_network_capture.pcapng

Output

Alerts will be printed to your console and simultaneously written to a log file within the `logs/` directory. The log file name will be timestamped (e.g., `logs/YYYY-MM-DD_HH-MM-SS.log`).