Browser-Based SIEM Telemetry & Detection Engine

Author: Rimsha Razi | **Target Domain:** rimrazcyber.com | **MITRE ATT&CK:** T1110.003

1. Executive Summary

This interactive tool executes Python 3 runtime logic natively inside the browser using WebAssembly (Wasm via PyScript). It parses SSH authentication logs, extracts structured indicators of compromise (IOCs), aggregates failure velocity per source IP, and triggers real-time alerts for brute-force attacks.

2. Technical Architecture & Component Stack

- **Runtime Engine:** PyScript (2024.1.1 Core) / Pyodide WebAssembly
- **Parsing & Extraction:** Python standard library regular expressions (`re` module with named capture groups)
- **State Aggregation:** `collections.defaultdict` tracking failed login velocity
- **DOM Interface:** `pyscript.document` API for real-time client-side rendering

3. Detection Algorithm

```
import re
from collections import defaultdict
from pyscript import document

def run_detection(event=None):
    log_data = document.getElementById("log-input").value
    pattern = r"status=(?P<status>\w+) user=(?P<user>\S+) src_ip=(?P<src_ip>[\d\.]+)"
    failed_attempts = defaultdict(list)

    for line in log_data.strip().split("
"):
        match = re.search(pattern, line)
        if match and match.group("status") == "FAILED":
            failed_attempts[match.group("src_ip")].append(match.group("user"))
```