HACK KRMU 5.0

- Problem Statement ID : PS 1
- Team Name : TEAM ROCKET
- TEAM ID : HK-105
- TEAM MEMBERS : K.PRAJVAL
  D.V.S.SAAKETH
  G.SRI VYSHNAVI

PROBLEM & SOLUTION

TOOL FOR SCANNING WEB APPLICATIONS & API'S

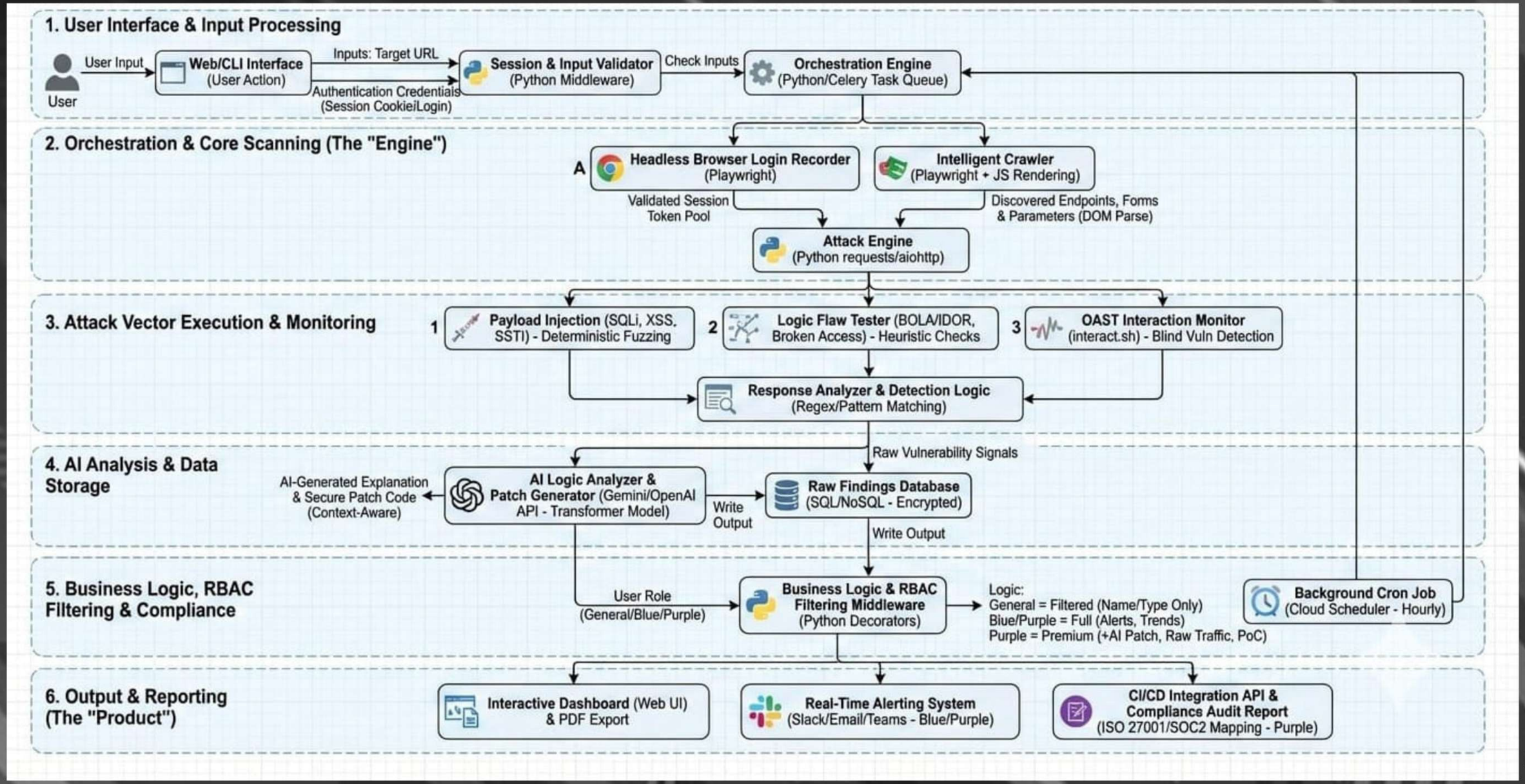
- Legacy scanners are not able to scan JavaScript heavy Single Page Applications (SPA's) and therefore are not able to find important URLs to scan.
- Most tools lose "session context" during an authenticated user session scan which limits their ability to find "authenticated-only" vulnerabilities like BOLA/IDOR.
- Standard tools are able to find syntax errors (SQLi) but do not have a way to detect logical errors (business logic abuse).

THE SOLUTION: V.I.P.E.R.

- V.I.P.E.R. is a tool that uses a headless browser (Playwright) to completely render the entire DOM of React/Vue Apps while ensuring 100% crawling coverage.
- V.I.P.E.R. uses Dual Session Logic which allows it to have two active user sessions open at the same time in order to detect BOLA by passing requests between users.
- V.I.P.E.R. is able to not only identify bugs but will utilize GenAI to provide developers with recommended secure coding fixes in real-time.

HACK KRMU 5.0

FLOW OF SOLUTION



TECH STACK & APPROACH

FRONT END:

BACK END:

Our Approach:

- Hybrid Analysis: Combining Static Analysis (response parsing) with Dynamic Analysis (headless browser interaction) for maximum accuracy.
- Microservices Architecture: Separating the "Crawler," "Scanner," and "Dashboard" allows the system to scale horizontally.

UNIQUENESS & INNOVATION FACTOR

- AI-Powered "Fix It" Button: We make connections between development and security by allowing users to easily submit a bug or threat to a LLM (Gemini/OpenAI) for patches via one simple click
- Shadow API Discovery: Our system actively hunts for "Zombie APIs" (e.g., /api/v1 when v2 is live) and exposed dev files (.env, .git).
- Real-Time "Attack Replay": A live terminal on the dashboard allows users to re-send a malicious request, proving the existence of vulnerabilities and eliminating false positives
- The Freemium Pipeline: Hook, Alert, and Resolve

1. General User (Free Hook): Lead generation tier offering Manual One-Off Scans and basic Vulnerability Name & Severity reporting to establish risk awareness.
2. Blue Team: Compliance Core (₹200 / 3 Months): Built for defenders. Unlocks Automated Background Scans, real-time Email/Slack Alerts, and Audit-Ready Compliance Checklists (ISO 27001/SOC2).
3. Purple Team: Ultimate Arsenal (₹800 / 1 Year): Built for elite security teams. Grants full access to Raw HTTP Traffic & Payload Data, AI-Generated Code Patches, automated Proof of Concept (PoC) Generation, and API Access for CI/CD pipeline integration.

FEASIBILITY & CHALLENGES

Feasibility:

- Modular Design: open-source platforms allowed our developers to spend their time coding logic (BOLA detection) instead of trying to reinventing the wheel.
- Containerization: The entire stack is Dockerized, ensuring it runs on any judge's machine immediately
- Microservices Architecture & RBAC Security

1. Scalable Microservices: Separating the "Crawler," "Scanner," and "Dashboard" allows the system to scale horizontally. Heavy tasks run asynchronously via message queues to prevent UI freezing.
2. Strict Role-Based Access Control (RBAC): Premium features are not just hidden via frontend CSS. Our backend API physically filters and strips out sensitive data (like AI-generated patches and raw traffic payloads) before the response ever reaches lower-tier users, completely preventing network-tab data theft..

Challenges Faced:

- Scan Duration: Headless browsing is slower than simple HTTP requests. Mitigation: We implemented a Redis Task Queue to run scans asynchronously without freezing the UI.
- Authentication Complexity: Maintaining session states (cookies/tokens) during aggressive scans. Mitigation: A dedicated "Session Keep-Alive" loop .

RESEARCH & REFERENCE

- OWASP Top 10 (2021): The core framework guiding our vulnerability detection categories (SQLi, BOLA, Broken Access Control).
- ProjectDiscovery.io: Research on OAST (Out-of-band Application Security Testing) and Nuclei templates.
- Playwright Documentation: Best practices for handling modern web capabilities and Shadow DOM.
- CVSS v3.1 is used to help establish the logic of automated severity scoring.

THANK YOU