**Detailed Project Report (DPR): AI-Powered Cybersecurity Monitoring System**

**1. Executive Summary**

**1.1 Project Title**

AI-Powered Security Monitoring System for PCAP-Based Threat Intelligence

**1.2 Project Overview**

This project aims to develop an **AI-driven cybersecurity monitoring platform** that processes **PCAP files** to detect, analyze, and predict cyber threats. By integrating **machine learning, predictive analytics, and NLP-based threat intelligence**, this system enhances traditional **Intrusion Detection and Security Information and Event Management (SIEM) solutions**.

**1.3 Objectives**

* Develop a **virtual machine (VM)-based** deployment for easy setup.
* Implement **PCAP-based analysis** to identify cyber threats.
* Integrate **AI models** for **threat intelligence extraction, predictive analytics, and automated risk scoring**.
* Build an interactive **SOC dashboard** for real-time security insights.

**2. Project Justification**

**2.1 Problem Statement**

Traditional cybersecurity monitoring tools rely on **signature-based detection**, which struggles against **zero-day attacks** and evolving adversarial tactics. Current **Security Onion implementations** provide strong network monitoring capabilities, but lack **AI-enhanced threat intelligence**.

**2.2 Proposed Solution**

By integrating **AI-powered analysis** into a Security Onion-like architecture, this project aims to **process historical network traffic (PCAP files) and detect cyber threats proactively**.

**3. Project Scope**

**3.1 Key Features**

* **PCAP Import & Analysis**: Zeek and Suricata for metadata extraction.
* **AI-Driven Threat Intelligence**:
  + NLP-based cybersecurity OSINT extraction (BERT, RoBERTa models).
  + LSTM/ARIMA predictive analytics for forecasting attack trends.
  + Risk scoring using Random Forest, XGBoost, and deep learning.
* **SOC Dashboard**:
  + React.js and Kibana-based real-time visualization.
  + Automated risk assessment and alerting system.

**3.2 Limitations**

* The system **does not monitor live traffic** (PCAP-based only).
* Requires **manual import of PCAP files**.
* Initial AI model training may require high computational resources.

**4. System Architecture**

**4.1 High-Level Design**

1. **PCAP Processing Pipeline** → Parses network traffic using Zeek & Suricata.
2. **AI-Powered Analysis** → Extracts intelligence & predicts future attacks.
3. **Data Storage** → Logs stored in Elasticsearch & PostgreSQL.
4. **SOC Dashboard** → Displays security insights & automated alerts.

**4.2 Technology Stack**

| **Component** | **Technology Used** |
| --- | --- |
| PCAP Processing | Zeek, Suricata |
| AI Models | TensorFlow, PyTorch, Scikit-Learn |
| Data Storage | Elasticsearch, PostgreSQL, Kafka |
| Dashboard | React.js, Kibana |
| Backend API | FastAPI, Flask |

**5. Implementation Plan**

**5.1 Development Phases**

| **Phase** | **Duration** | **Activities** |
| --- | --- | --- |
| Research & Planning | 2 weeks | Define scope, study cybersecurity tools. |
| Environment Setup | 1 week | Install OS, configure VM, install software. |
| PCAP Processing | 2 weeks | Implement Zeek & Suricata pipeline. |
| AI Model Integration | 3 weeks | Train NLP, LSTM models for threat intelligence. |
| Backend & API | 2 weeks | Develop FastAPI for system communication. |
| Dashboard Development | 3 weeks | Build React.js-based SOC UI. |
| Testing & Optimization | 2 weeks | Evaluate AI accuracy, system performance. |
| Deployment & Documentation | 2 weeks | Package system, write user manual. |

**5.2 Resource Requirements**

* **Hardware**: 4-core CPU, 8GB RAM, 100GB storage.
* **Software**: Ubuntu 20.04, Python, Zeek, Suricata, Elasticsearch, React.js.

**6. Cost Estimation**

**6.1 Budget Breakdown**

| **Item** | **Estimated Cost (USD)** |
| --- | --- |
| Virtual Machine Setup | $100 |
| AI Model Training | $300 |
| Software Licenses (if any) | $50 |
| Development & Testing | $500 |
| Documentation & Deployment | $200 |
| **Total** | **$1150** |

**7. Risk Assessment**

**7.1 Key Risks & Mitigation Strategies**

| **Risk** | **Impact** | **Mitigation Strategy** |
| --- | --- | --- |
| High computational requirements | Medium | Use pre-trained AI models to reduce training time. |
| Accuracy of AI threat prediction | High | Continuously refine models with updated datasets. |
| Security vulnerabilities | High | Apply security best practices (RBAC, encryption). |
| Data storage scalability | Medium | Optimize Elasticsearch and use database indexing. |

**8. Expected Outcomes**

* A **fully functional VM-based AI-powered cybersecurity platform**.
* Improved **threat detection accuracy** compared to traditional IDS systems.
* An **interactive SOC dashboard** with real-time AI-driven insights.
* Demonstrated ability to **predict cyber threats using historical data**.

**9. Conclusion**

This project provides an **affordable, research-friendly cybersecurity solution** that integrates **AI into traditional security monitoring workflows**. By using **PCAP imports and machine learning models**, it **enhances threat detection and proactive security analysis**.

**10. Next Steps**

1. **Deploy prototype on a test environment**.
2. **Train AI models on diverse attack datasets**.
3. **Optimize performance for real-time analysis**.
4. **Expand dashboard capabilities for better visual insights**.