# Project Proposal: AI-Based Threat Detection in Zero Trust Architecture

**Srushti Waichal (A20554818)**

## 1. Introduction

### 1.1 Background

This project builds upon a previous theoretical study on Zero Trust Architecture (ZTA), which outlined a framework for organizations to transition from traditional perimeter-based security models to a more robust, granular security approach. The original study emphasized six core processes essential for ZTA implementation, including planning, designing, migration, and continuous monitoring

### 1.2 Project Objective

The primary objective of this project is to develop an AI-based threat detection module as a critical component of a Zero Trust Architecture implementation. This focused approach aligns with the professor's recommendation to concentrate on one subtopic this semester, using it as a foundation for potential future expansion.

Key objectives include:

1. Developing and training AI models for accurate threat detection in network traffic
2. Integrating the AI module with core ZTA principles
3. Implementing real-time threat analysis capabilities
4. Optimizing system performance for high accuracy and low latency
5. Assessing the scalability and adaptability of the solution

By concentrating on AI-based threat detection within the ZTA framework, this project seeks to create an innovative solution that addresses modern cybersecurity challenges while providing a foundation for potential future expansions into comprehensive ZTA implementation.

---

## 2. Scope and Deliverables

### 2.1 Project Scope

The project will focus on the following key aspects:

1. Development of an AI-based Threat Detection Module
2. Integration of the module with core ZTA principles
3. Performance evaluation of the threat detection system

### 2.2 Deliverables

1. AI models for network traffic anomaly detection
2. Prototype system integrating AI models with basic ZTA principles
3. Performance evaluation report
4. Project documentation and final report

---

## 3. Implementation Strategy

### 3.1 Data Collection and Preprocessing

- Dataset Acquisition: Gather diverse network traffic datasets, including both normal and malicious traffic patterns. Potential sources include public cybersecurity datasets (e.g., UNSW-NB15, CICIDS2017) and simulated network traffic.
- Data Cleaning: Remove inconsistencies, handle missing values, and normalize data formats.
- Feature Engineering: Extract relevant features from raw network traffic data, such as packet sizes, inter-arrival times, and protocol-specific attributes.
- Data Labeling: Ensure accurate labeling of normal and malicious traffic samples for supervised learning.

### 3.2 Model Development

- **Algorithm Selection**: Implement and train multiple machine learning models, including:
    - Random Forests for their ability to handle high-dimensional data
    - Neural Networks for capturing complex patterns in network traffic
    - Support Vector Machines for their effectiveness in binary classification tasks
- **Model Architecture**: Design appropriate model architectures, considering factors like input dimensionality and desired output format.
- **Training Process**: Utilize Python libraries such as TensorFlow and Scikit-learn for model implementation and training.
- **Hyperparameter Tuning**: Employ techniques like grid search or random search to optimize model hyperparameters.

### 3.3 Model Evaluation and Optimization

- **Performance Metrics**: Evaluate models using metrics such as accuracy, precision, recall, F1-score, and area under the ROC curve (AUC-ROC).
- **Cross-Validation**: Implement k-fold cross-validation to ensure robust performance estimates.
- **Ensemble Methods**: Explore ensemble techniques to combine predictions from multiple models for improved accuracy.
- **Feature Importance Analysis**: Identify the most influential features for threat detection to refine the model and improve interpretability.

### 3.4 Integration and Testing

- **ZTA Integration**: Develop a prototype system that integrates the AI models with basic ZTA principles, focusing on the "never trust, always verify" approach.
- **Real-Time Processing**: Implement efficient data processing pipelines to enable real-time threat detection.
- **Testing Framework**: Create a comprehensive testing framework using tools like pytest for unit testing and network traffic simulators for system-level testing.
- **Performance Benchmarking**: Conduct thorough performance testing, including latency measurements and scalability assessments.

---

### 4. Alignment with Course Curriculum

This project aligns with several key areas of the course:

- **Network Security**: Addresses threat detection in network environments, including identification of potential DoS attacks and unauthorized access attempts.
- **Machine Learning in Security**: Directly applies ML techniques to cybersecurity challenges, demonstrating the practical use of AI in threat detection.
- **System Security**: Aims to detect various system-level threats like malware, botnets, and intrusion attempts.

---

### 5. Tools and Technologies

- **Programming Language**: Python
- **ML Libraries**: TensorFlow, Scikit-learn, PyTorch
- **Data Processing**: Pandas, NumPy
- **Testing**: Pytest, network traffic simulators
- **Version Control**: Git

---

### 6. Timeline and Milestones

| Week | Milestone | Details |
|---|---|---|
| 1-2 | Data collection and preprocessing | Acquire datasets, clean data, engineer features |
| 3-4 | Initial model development and training | Implement and train multiple ML models |
| 5-6 | Model evaluation and optimization | Evaluate performance, tune hyperparameters |
| 7-8 | Integration with basic ZTA principles | Develop prototype integrating AI models with ZTA concepts |
| 9-10 | Testing and performance analysis | Conduct comprehensive testing and benchmarking |
| 11-12 | Documentation and final report preparation | Prepare detailed documentation and final project report |

---

### 7. Future Expansion

While this semester's focus is on AI-based threat detection, the project lays the groundwork for future expansion into other ZTA components, such as:

- Network Access Control System
- Real-Time Monitoring Dashboard
- Comprehensive ZTA Implementation

## 8. Conclusion

This project offers a focused and manageable scope for the semester while providing valuable insights into advanced cybersecurity techniques. By concentrating on AI-based threat detection within the ZTA framework, we aim to develop a practical, innovative solution that addresses real-world security challenges. The knowledge and experience gained will serve as a solid foundation for potential expansion into a more comprehensive ZTA implementation in future work or graduate thesis research.

## 9. References


https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=10052642

https://arxiv.org/pdf/2309.03582

https://ieeexplore.ieee.org/abstract/document/9773102

https://link.springer.com/article/10.1186/s43067-024-00155-z

https://www.sciencedirect.com/science/article/abs/pii/S1389128622003929

https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4331547