Neurodefender XDR Platform

An advanced Extended Detection and Response (XDR) platform integrating SIEM, NGFW, and Phishing Protection with deep learning capabilities.

Overview

Neurodefender is an enterprise-grade security platform that combines:

- Next-Generation Firewall (NGFW) with deep learning-based threat detection
- Security Information and Event Management (SIEM) with advanced correlation
- AI-powered Phishing Protection system

Key Features

NGFW Component

- Real-time deep learning-based threat detection
- Advanced protocol analysis
- Zero-day attack prevention
- Custom rule engine with ML augmentation

SIEM Component

- Intelligent log aggregation and correlation
- Machine learning-based alert prioritization
- Automated incident response
- Advanced threat hunting capabilities

Phishing Protection

- Real-time email analysis
- URL reputation checking
- Deep learning-based content analysis
- Behavioral pattern detection

System Requirements

Production Environment

- CPU: 16+ cores
- RAM: 64GB minimum
- Storage: 1TB+ NVMe SSD
- Network: 10Gbps interface
- OS: Linux (Ubuntu 22.04 LTS or RedHat 8.x)

Development Environment

- CPU: 8+ cores
- RAM: 32GB minimum
- Storage: 512GB SSD
- OS: Linux/macOS

Quick Start

1. Clone the repository:

```
git clone [repository-url]
cd neurodefender
```

2. Set up environment variables:

```
cp .env.example .env
# Edit .env with your configuration
```

3. Start development environment:

```
make dev-setup
docker-compose -f docker-compose.dev.yml up
```

4. Run tests:

```
make test
```

Development Setup

Prerequisites

- Docker Engine 24.0+
- Docker Compose 2.20+
- Rust 1.75+
- Python 3.11+
- Go 1.21+
- NVIDIA GPU drivers (for ML components)

Building from Source

1. Install dependencies:

```
make install-deps
```

2. Build components:

```
make build-all
```

3. Start services:

```
make run
```

Project Structure

```
neurodefender/
├── security-core/    # Core security services
├── ngfw-core/       # NGFW implementation
├── siem-processor/  # SIEM processing engine
├── ml-platform/     # Machine learning components
└── phishing-protection/ # Anti-phishing system
```

Configuration

Configuration is managed through:

- Environment variables
- Configuration files in `config/`
- Runtime settings via admin API

See `docs/configuration/` for detailed configuration options.

Documentation

- Architecture Overview: `docs/architecture/`
- API Documentation: `docs/api/`
- Deployment Guide: `docs/deployment/`
- User Guides: `docs/user_guides/`

Security

- All security vulnerabilities should be reported to [security contact]
- See SECURITY.md for our security policy
- Regular security audits are conducted

Contributing

1. Read CONTRIBUTING.md for guidelines
2. Set up development environment
3. Create feature branch
4. Submit pull request

License

Proprietary software. See LICENSE file for details.

Support

- Enterprise Support: [support contact]
- Documentation: [docs link]
- Training: [training contact]

Acknowledgments

Built with:

- Rust for performance-critical components
- Python for ML/AI components
- Go for service coordination