IDS Project - Clean Structure

Current Active Components

🚀 **Main Pipeline** (Primary System)

Location: `dpdk_suricata_ml_pipeline/`

This is the **production-ready, flow-based ML inference pipeline**:

- ✅ Processes ALL network flows (100% coverage)
- ✅ CICIDS2017 65-feature extraction
- ✅ Dual detection: Suricata signatures + ML anomaly detection
- ✅ Combined threat scoring
- ✅ Real-time Kafka streaming

Quick Start:

```
cd dpdk_suricata_ml_pipeline
# Follow README.md for setup
```

📊 **Jupyter Notebooks** (Analysis & Research)

- `CICIDS2017.ipynb` - CICIDS2017 dataset analysis and model training
- `CICIDS2018.ipynb` - CICIDS2018 dataset analysis and model training
- `PerformanceEvaluation_AdaptiveEnsembles.ipynb` - Ensemble model evaluation
- `Suricata_DPDK_Feature_Extraction.ipynb` - Feature engineering experiments

Purpose: Research, model training, dataset analysis

PROF

🤖 **ML Models**

Location: `ML Models/`

- `random_forest_model_2017.joblib` - Random Forest trained on CICIDS2017
- `lgb_model_2018.joblib` - LightGBM trained on CICIDS2018

Used by: `dpdk_suricata_ml_pipeline/src/model_loader.py`

🧪 **Testing & Development Scripts**

Keep these for testing specific functionality:

- `test_adaptive_ensemble.py` - Ensemble model testing
- `test_attack_generator.py` - Attack pattern generation
- `test_benign_traffic.py` - Benign traffic testing
- `test_dpdk_scapy_integration.py` - DPDK integration tests

- `test_ensemble_model.py` - Model validation
- `test_ml_attack_patterns.py` - Attack pattern validation
- `test_ml_classifications.py` - Classification testing
- `quick_attack_demo.py` - Quick attack demo
- `quick_dpdk_test.py` - Quick DPDK testing

🔧 Utility Scripts

- `activate_venv.sh` - Activate Python virtual environment
- `install_dpdk_suricata.sh` - Install DPDK and Suricata
- `create_test_models.py` - Generate test models
- `ml_enhanced_ids_pipeline.py` - Standalone ML-enhanced IDS (development version)
- `ml_enhanced_pipeline.sh` - Helper script for ML pipeline

🧬 Advanced Research Components

- `adaptive_ensemble_predictor.py` - Adaptive ensemble implementation
- `advanced_attack_generator.py` - Advanced attack generation
- `attack_simulator.py` - Network attack simulation

🌐 Python Virtual Environment

Location: `venv/`

- Python 3.12.3 with all dependencies installed
- Activate with: `source venv/bin/activate`

📁 Configuration

- `config/ids_config.yaml` - IDS system configuration
- `requirements.txt` - Python dependencies

📖 Documentation

PROF

- `README.md` - This file (project overview)
- `DPDK_SURICATA_INSTALLATION.md` - DPDK/Suricata installation guide
- `VENV_SETUP.md` - Virtual environment setup guide
- `VENV_SETUP_COMPLETED.md` - Setup completion notes

🗃 Legacy/Reference Directories

Keep for reference or specific use cases:

- `Building DPDK Pipeline for Packet Generation/` - DPDK packet generation research
- `Suricata Integration/` - Suricata integration experiments
- `Suricata_Integration/` - Additional integration work
- `Suricata_Setup/` - Suricata setup files
- `src/` - Source code utilities and experiments

🗑️ Files Removed (Redundant/Obsolete)

The following files were removed as they're superseded by `dpdk_suricata_ml_pipeline/`:

Removed Python Files:

- ~~`ml_enhanced_ids_pipeline.py.broken`~~ - Broken version
- ~~`ml_enhanced_ids_pipeline_original.py`~~ - Old version
- ~~`ml_alert_consumer.py`~~ - Superseded by `dpdk_suricata_ml_pipeline/src/ml_kafka_consumer.py`
- ~~`realtime_dpdk_pipeline.py`~~ - Superseded by new pipeline
- ~~`realtime_ids_monitor.py`~~ - Superseded by `status_check.sh`

Removed Shell Scripts:

- ~~`ml_enhanced_pipeline.sh.debug_backup`~~ - Debug backup
- ~~`fix_ensemble_integration.sh`~~ - One-time fix script
- ~~`ensemble_demo.sh`~~ - Demo script
- ~~`ml_testing_demo.sh`~~ - Demo script
- ~~`test_args.sh`~~ - Test script
- ~~`quick_pipeline_check.sh`~~ - Replaced by `status_check.sh`
- ~~`setup_dpdk_pktgen.sh`~~ - Old setup
- ~~`setup_real_dpdk_suricata.sh`~~ - Old setup
- ~~`setup_realtime_dpdk.sh`~~ - Old setup
- ~~`start_pipeline.sh`~~ - Replaced by numbered scripts
- ~~`validate_complete_pipeline.sh`~~ - Validation script
- ~~`system_overview.sh`~~ - Old monitoring
- ~~`system_status.sh`~~ - Replaced by `status_check.sh`

Removed Documentation:

- ~~`ENSEMBLE_IMPLEMENTATION_SUCCESS.md`~~ - Historical doc
- ~~`BENIGN_TRAFFIC_FIXES.md`~~ - Historical doc
- ~~`REPOSITORY_STATUS.md`~~ - Outdated status
- ~~`ML_ENHANCED_README.md`~~ - Superseded by pipeline README
- ~~`REALTIME_DPDK_README.md`~~ - Superseded by pipeline docs
- ~~`ENSEMBLE_README.md`~~ - Consolidated
- ~~`ATTACK_GENERATOR_README.md`~~ - Consolidated
- ~~`ADAPTIVE_ENSEMBLE_INTEGRATION.md`~~ - Historical doc
- ~~`TESTING_README.md`~~ - Consolidated
- ~~`config.yaml`~~ - Old config format

Removed Test Files:

- ~~`quick_attack_test.py`~~ - Empty file
- ~~`test_ml_attacks.py`~~ - Empty file
- ~~`test_enhanced_pipeline.py`~~ - Redundant
- ~~`test_ensemble_complete.py`~~ - Redundant

- ~~test_ml_integration.py~~ - Redundant

Project Structure (Clean)

```
IDS/
├── dpdk_suricata_ml_pipeline/     ← MAIN PRODUCTION PIPELINE
│   ├── src/                       (4 Python modules)
│   ├── scripts/                   (Pipeline management)
│   ├── config/                    (Configuration files)
│   ├── logs/                      (Runtime logs)
│   └── pcap_samples/              (Test PCAP files)
├── ML Models/                     (Trained ML models)
├── venv/                          (Python virtual environment)
├── config/                        (System configuration)
├── src/                           (Utility source code)
├── *.ipynb                        (Jupyter notebooks for research)
├── test_*.py                      (Unit tests - keep for validation)
├── quick_*.py                     (Quick test scripts)
├── *_ensemble*.py                 (Ensemble research code)
├── advanced_attack_generator.py   (Advanced testing)
├── attack_simulator.py            (Attack simulation)
├── install_dpdk_suricata.sh       (Installation script)
├── activate_venv.sh               (Venv activation)
├── requirements.txt               (Dependencies)
└── README.md                      (This file)
```

Usage Priority

For Production IDS:

1. **Use**: `dpdk_suricata_ml_pipeline/`
2. **Documentation**: See `dpdk_suricata_ml_pipeline/README.md`
3. **Setup**: Follow `dpdk_suricata_ml_pipeline/SETUP_GUIDE.md`

PROF

For Research/Development:

1. **Notebooks**: CICIDS*.ipynb, PerformanceEvaluation*.ipynb
2. **Test Scripts**: test_*.py, quick_*.py
3. **Experimental Code**: adaptive_ensemble_predictor.py, etc.

For Testing:

1. **Unit Tests**: Run `test_*.py` scripts
2. **Attack Simulation**: Use `attack_simulator.py`, `advanced_attack_generator.py`
3. **Integration Tests**: Use `test_dpdk_scapy_integration.py`

Quick Commands

Start Production Pipeline:

```
cd dpdk_suricata_ml_pipeline
sudo ./scripts/01_bind_interface.sh
./scripts/02_setup_kafka.sh
sudo ./scripts/03_start_suricata.sh
./scripts/04_start_ml_consumer.sh
```

Run Tests:

```
source venv/bin/activate
python test_adaptive_ensemble.py
python test_ensemble_model.py
```

View Results:

```
# ML Consumer logs
tail -f dpdk_suricata_ml_pipeline/logs/ml/ml_consumer.log

# Enhanced alerts
kafka-console-consumer.sh --bootstrap-server localhost:9092 \
    --topic ml-predictions --from-beginning
```

Notes

- **Main Pipeline**: `dpdk_suricata_ml_pipeline/` is the complete, production-ready system
- **Legacy Code**: Kept for reference and specific research purposes
- **Jupyter Notebooks**: Useful for dataset analysis and model training
- **Test Scripts**: Keep for validation and development

PROF

Summary

✅ **Cleaned**: Removed 30+ redundant/obsolete files
✅ **Organized**: Clear separation between production, research, and testing
✅ **Documented**: Comprehensive README with usage instructions
✅ **Functional**: All essential components preserved and working

For detailed architecture and flow-based ML inference information, see:

- `dpdk_suricata_ml_pipeline/FLOW_BASED_ML_ARCHITECTURE.md`
- `dpdk_suricata_ml_pipeline/IMPLEMENTATION_SUMMARY.md`