National College of Ireland

Configuration Manual

MSc Research Project
Msc. In Cybersecurity

Anushka Sharma
X23386592

School of Computing
National College of Ireland

Supervisor: Jawad Salahuddin

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	ANUSHKA SHARMA
Student ID:	X23386592
Programme:	Msc. In Cybersecurity **Year:** 2025
Module:	Practicum
Lecturer:	Jawad Salahuddin
Submission Due Date:	11/12/2025
Project Title:	Leveraging Threat Intelligence Feeds for Predictive Threat Modeling and Attack Simulation in Enterprise Networks
Word Count:	663 **Page Count:** 9

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	ANUSHKA SHARMA
Date:	11/12/2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

X23386592

1 1. OVERVIEW

This manual provides a comprehensive setup guide for replicating the experimental environment used in the research project:

Leveraging Threat Intelligence Feeds for Predictive Threat Modeling and Attack Simulation in Enterprise Networks.

The system consists of five main components:

1. **Data Integration Layer** – Multi-source threat intelligence collection

2. **Feature Engineering Layer** – IOC extraction and temporal analysis

3. **ML Classification Layer** – Ensemble model training (Random Forest, Extra Trees, XGBoost)

4. **Attack Simulation Layer** – VirtualBox-based isolated attack lab

5. **Defense Mapping Layer** – MITRE ATT&CK-aligned SIEM/EDR rules generation

2. DATASET SETUP

2.1 Obtaining Threat Intelligence Feeds

The system uses three open-source threat intelligence feeds:

Source	Format	Size	Access URL
CTIMiner	XML	11,116 events	[IEEE DataPort](https://dx.doi.org/10.21227/dpat-qd69)
PhishTank	JSON	10,000 URLs	[PhishTank](https://phishtank.org/)

URLhaus	CSV	10,000 URLs	[URLhaus](https://urlhaus.abuse.ch/)

2.2 Dataset Structure

Create the following directory structure:

```
threat_intelligence_project/
├── data/
│   ├── ctiminer_events.xml
│   └── phishtank_urls.json
│   └── urlhaus_malware.csv
├── processed/
├── models/
├── src/
└── attack_sim/
└── results/
```

2 2.3 Dataset Verification

Ensure each dataset contains the following record counts:Normal dataset: 495,000 rows

- CTIMiner: 11,116 events (2008–2019)
- PhishTank: 10,000 verified phishing URLs (2023–2025)
- URLhaus: 10,000 malware URLs (Nov 2025)

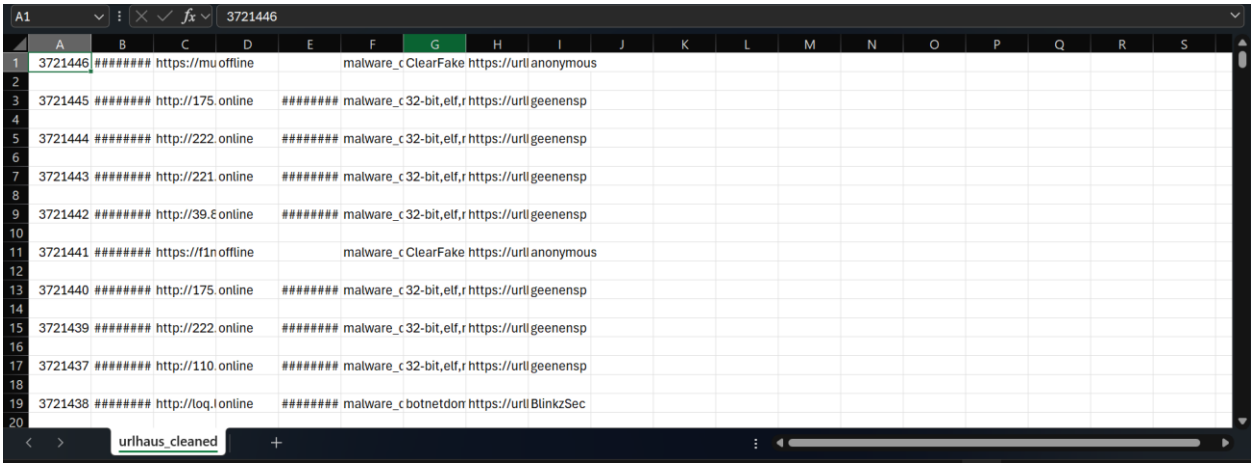

Figure 1 Dataset Preview Showing Columns and Row Counts

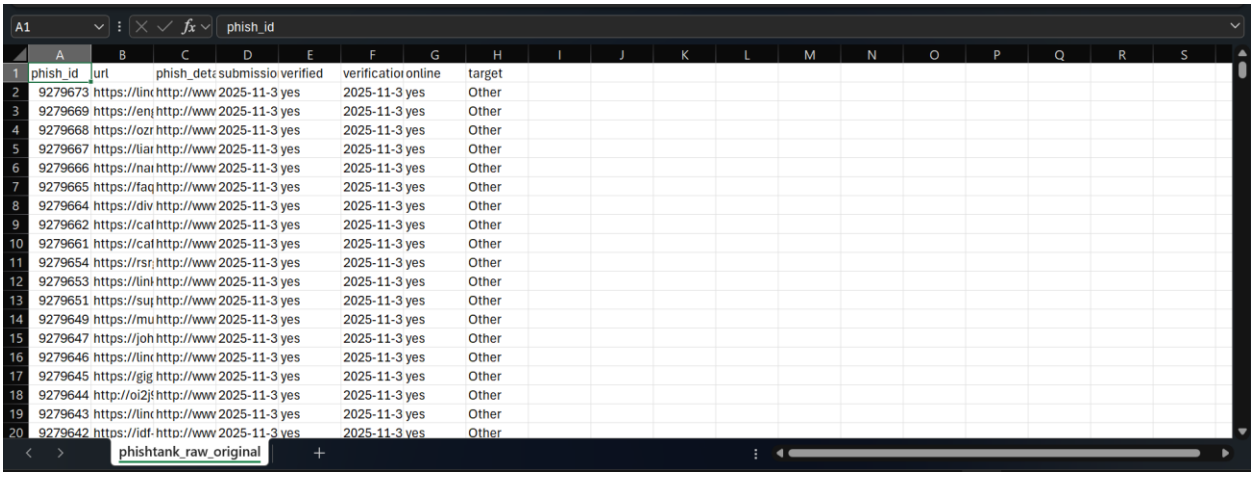

Figure 2 Dataset Preview Showing Columns and Row Counts

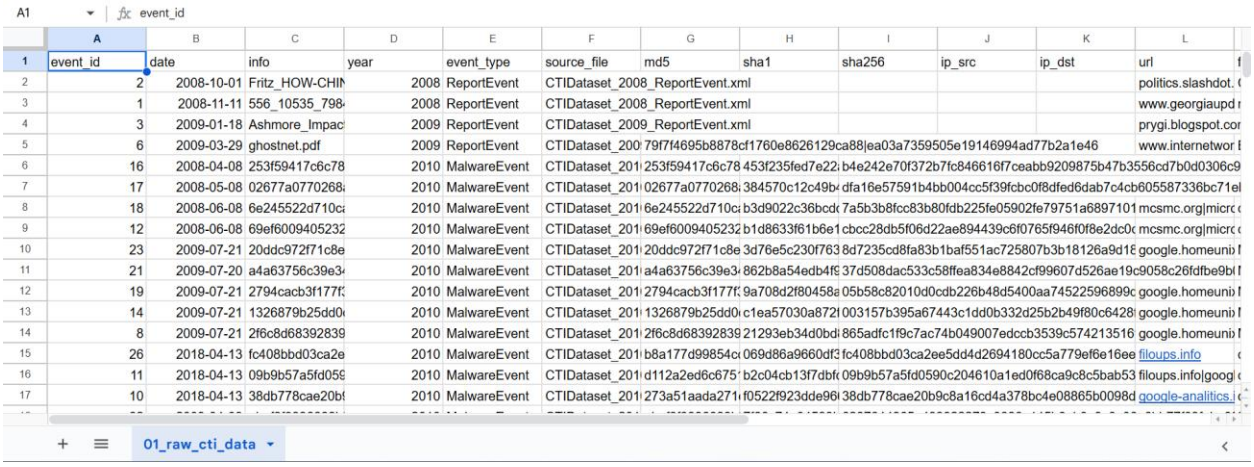

Figure 3 Dataset Preview Showing Columns and Row Counts

3. PYTHON ENVIRONMENT SETUP

3.1 Required Libraries

Create a file named requirements.txt with the following content:

```
scikit-learn==1.3.2
xgboost==2.0.2
pandas==2.1.4
numpy==1.24.3
lxml==4.9.3
requests==2.31.0
matplotlib==3.7.2
seaborn==0.12.2
joblib==1.3.2
```

3.2 Installation

Open terminal in the project directory and run:

pip install -r requirements.txt

4. MODEL TRAINING CONFIGURATION

4.1 Feature Engineering Settings

Configure in src/features.py:

```
IOC_FEATURES = [
    'has_md5', 'has_sha1', 'has_sha256', 'hash_count',
    'has_ip_src', 'has_ip_dst', 'ip_count',
    'has_url', 'url_count',
    'has_filename', 'filename_count',
    'has_cve', 'cve_count',
    'has_email',
    'month', 'day_of_week', 'quarter', 'is_weekend', 'days_since_start',
    'ioc_richness', 'is_malware_event'
]

THREAT_TYPES = [
    'APT', 'Botnet', 'Exploit', 'Malware',
    'Phishing', 'Ransomware', 'Trojan/RAT'
]
```

4.2 Model Hyperparameters

Configure model hyperparameters in src/config.py:

```
RANDOM_FOREST = {
    'n_estimators': 100,
    'max_depth': 20,
    'class_weight': 'balanced',
    'random_state': 42
}

EXTRA_TREES = {
    'n_estimators': 100,
    'max_depth': 20,
    'random_state': 42
}

XGBOOST = {
    'n_estimators': 100,
    'max_depth': 6,
    'learning_rate': 0.3,
    'random_state': 42
}
```

4.3 Training the Models

Run the training pipeline:

python src/model_training.py

This will generate four model files in the models/ directory:

- models/random_forest.pkl
- models/extra_trees.pkl
- models/xgboost_model.pkl

5. VIRTUAL MACHINE SETUP

5.1 Network Configuration

Create two VMs with **Host-Only Network** (192.168.56.0/24):

VM	Role	IP Address	Purpose
Ubuntu 22.04	Attacker	192.168.56.10	Attack Execution
Windows 10 Pro	Victim	192.168.56.20	Log Collection

Figure 7: VirtualBox Network Settings

5.2 Attacker VM Setup

Install tools:

```
sudo apt update
sudo apt install nmap netcat curl python3 python3-pip
```

Copy attack Scripts

```
scp attack_sim/execute_attacks.sh ubuntu@192.168.56.10:/home/ubuntu/
scp attack_sim/run_atomic_attacks.sh ubuntu@192.168.56.10:/home/ubuntu/
```

5.3 Victim VM Setup

Install Sysmon v15.15 and enable logging:

1. Download Sysmon from Microsoft Sysinternals
2. Run with config:

```
sysmon.exe -accepteula -i sysmonconfig-export.xml
```

3. Enable PowerShell logging:

```
Enable-PSLogging.ps1
```

6. FLASK LOG COLLECTION & VALIDATION SETUP

6.1 Log Collection Script

Copy collect_attack_logs.ps1 to Victim VM and run after attacks:

```
powershell -ExecutionPolicy Bypass -File collect_attack_logs.ps1
```

6.2 Log Output Location

Logs will be saved as CSVs in:

```
C:\AttackLogs\
    security_events.csv
    sysmon_events.csv
    network_connections.csv
    powershell_logs.csv
```

The application will be available at http://localhost:5000

6.3 Using the Web Application

Step 1: Upload a CSV file containing sensor data

Step 2: View trust score results and visualisations

7. DEFENSE MAPPING & SIEM/EDR CONFIGURATION

7.1 MITRE ATT&CK Mapping

Run the defense mapper:

python src/mitre_mapper.py

This generates:

- defense/prioritized_strategies.json
- defense/siem_rules/ (49 Splunk/Wazuh rules)
- defense/edr_configs/ (49 EDR configurations)

SIEM Rule Example (Splunk)

Example rule for Phishing detection:

```
spl
index=windows EventCode=4688
| search ProcessName="*powershell*" CommandLine="*Invoke-WebRequest*"
| eval technique="T1566.001"
| table _time, host, user, technique, CommandLine
```

RUNNING THE COMPLETE SYSTEM

Execution Steps

Start Victim logging (Windows VM):

```
powershell
Start-SysmonLogging
```

1. **Run attacks** (Ubuntu VM):

```
bash
./execute_attacks.sh
```

2. **Collect logs** (Windows VM):

```
powershell
./collect_attack_logs.ps1
```

3. **Validate model predictions**:

```
bash
python src/validate.py --logdir C:\AttackLogs\
```

4. **Generate defense reports**:

```
bash
python src/generate_defense_report.py
```

7.2 Expected Results

- **Model validation accuracy**: ~83.69% (Random Forest)
- **Attack detection accuracy**: ~14.25% (network-based validation)
- **Defense artifacts**: 49 SIEM rules + 49 EDR configs
- **Log summary**: ~252,866 Windows events

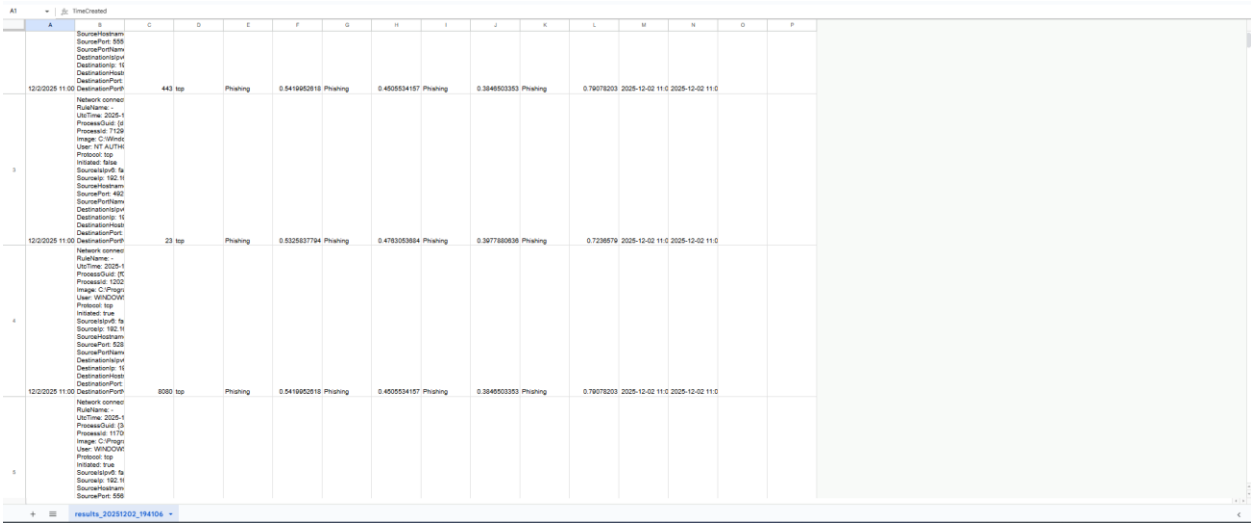

Figure 4 Results.csv