Ariel – CISCO Network Malicious Traffic Detection Challenge - 2021 (Based on the M.Sc. work of Ofek Bader and Adi Lichy)

Challenge Duration - From 17/11/21 to 9/12/21

Challenge Event Closing

Winners will be announced on 12/12/21 in a special event with Mr. Guy Keinan, Director of Engineering at Cisco. The event will also include a keynote lecture by cisco.

Prizes:

Cisco volunteered to sponsor this event and offer two prestige prizes for the first and second places: We will announce soon what are the exact prizes ☺.

The winners will be selected based on the highest average ranking score in all challenges, innovation, and Cisco judges' final verdict.

Description

In this challenge, your task is to predict and classify malicious and benign network traffic by utilizing your knowledge in data exploration and machine learning models.

In this challenge, you will have access to two **datasets:**

1. MTA – Malware samples from: https://www.malware-traffic-analysis.net/
2. USTC – Malware samples from: https://github.com/yungshenglu/USTC-TFC2016
3. Zero day dataset – will be provided in the last week of the competition.

For each dataset, you will have two types of labels, hence two types of **tasks**:

1. Binary classification – Detect if the file is benign/malware ('label' column in the data).
2. Multiclass classification – Detect the 'malware_family' – multiclass labeling including benign.

Note: You will have only labels for the Training part.

For each combination of **dataset** + **task,** you will have two types of **exercises/phases**:

1. Validation – without labels
2. Test – without labels

Resources

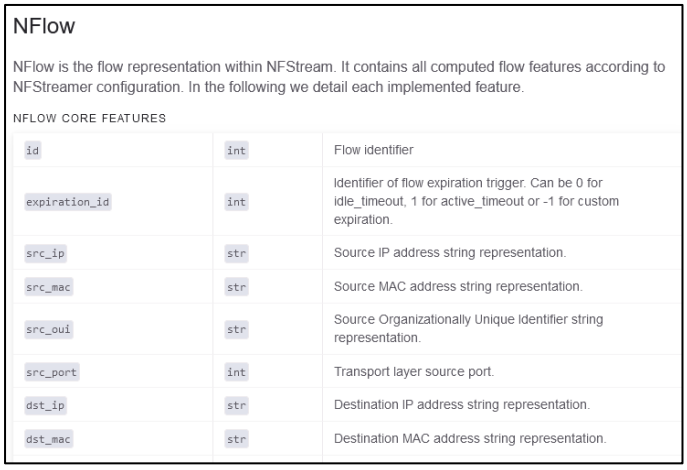
A. In Moodle, you will have a Jupyter notebook that shows how to generate the model output for the automatic validation using EvalAI. EvalAI is a cloud platform that you will be used to operate and check your model outputs.

Features in the dataset

The first 83 metadata features (columns) in the CSV files are identical to ones extracted by the NFStream utility. You can read about the features in the documentation of NFStream in the section of 'NFlow' here: https://www.nfstream.org/docs/api.

Note: there is no need to run the base features. We already did for you.

A partial screenshot of the documentation:



NFlow

NFlow is the flow representation within NFStream. It contains all computed flow features according to NFStreamer configuration. In the following we detail each implemented feature.

NFLOW CORE FEATURES

`id`	`int`	Flow identifier
`expiration_id`	`int`	Identifier of flow expiration trigger. Can be 0 for idle_timeout, 1 for active_timeout or -1 for custom expiration.
`src_ip`	`str`	Source IP address string representation.
`src_mac`	`str`	Source MAC address string representation.
`src_oui`	`str`	Source Organizationally Unique Identifier string representation.
`src_port`	`int`	Transport layer source port.
`dst_ip`	`str`	Destination IP address string representation.
`dst_mac`	`str`	Destination MAC address string representation.

Additional features

Feature names that start with 'udps', are additional features that we have provided to you from what NFStream has to offer.

1. **udps.n_bytes** - A 784 length list which contains the first 784 payload bytes of the flow/session**.**
2. **udps.n_bytes_per_packet** - A (100,2) matrix, which contain the 100 first payload bytes in the first 2 packets (non-empty payload packets) of the payload/flow.
3. **udps.protocol_header_fields** – A (32,4) matrix, [direction, payload size, delta_time, tcp win size] of the first 32 packets. If the session Is not TCP-based, then the TCP-win-size is set to 0.
4. **udps.stnn_image –** A (5,14) matrix, with statistical features about the flow/session.
   1st row: bidirectional related data.
   2nd row: src -> dst related data.
   3rd row: third row: dst -> src related data.
   4th row: handshake packets only related data.
   5th row: data packets only related data.

Each row consists of 14 statistical features:

[iat max, iat min, iat mean, iat stddev, iat skew,
size min, size max, size mean, size stddev, size skew,
#packets, #bytes, pkts/time, bytes/time]

Where IAT = inter arrival time or delta time, size = packet payload size.

Submission – EvalAi (will be available from the second week)

URL: https://eval.ai/web/challenges/challenge-page/1357/overview

How to submit:

1. Generate a **predictions.txt** file for an exercise with a specific **dataset**, **task**, **phase** combination.
2. Go to the Eval.ai URL – Submit page.
3. Select a phase.
4. Upload your submission file (predictions.txt).
5. Click submit.

You can see your results under "My submissions" – 'Result File' and on the Leaderboard page.

What to submit?

A. Mandatory to submit your last result to EvalAI website
B. Submit your final Jupyter code to Moodle with all the evidence of the training part.
C. Present 5 min slides about your approach:
    a. Features used
    b. AI/ML
    c. What is your proposed innovation for solving these problems?