# IDSdm — Información sobre las Herramientas Desarrolladas

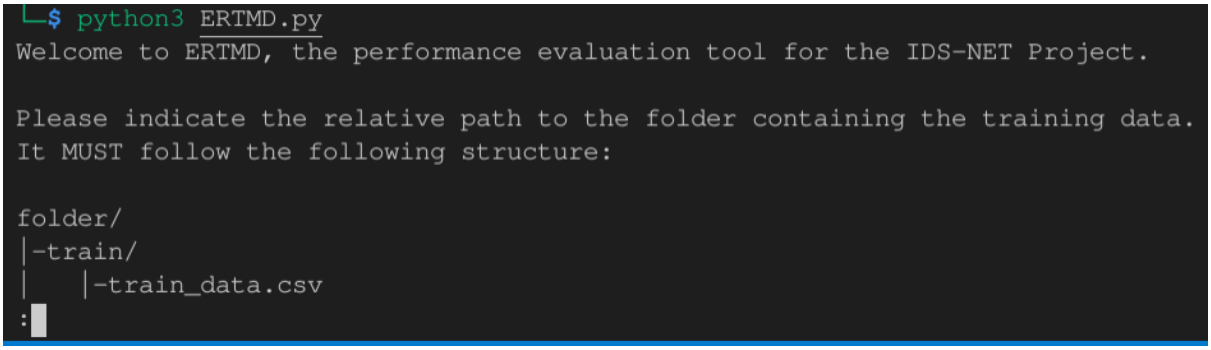
## Modelo de estructura de directorio para la carga de datos

Tal y como se ve en la siguiente figura, el directorio padre indicado debe contener dos subdirectorios llamados train y test. En el primero, estarán almacenados en un solo archivo .csv los datos de entrenamiento, es decir, datos etiquetados. En el segundo, estarán almacenados en un solo archivo .csv los datos de entrenamiento, es decir, datos no etiquetados sobre los cuales se desea realizar predicciones. Con la herramienta de apoyo para evaluación de rendimiento de técnicas de minería de datos, se puede omitir el subdirectorio con los datos de prueba.


Descripción de la estructura de directorio admitida por las aplicaciones.

## Resultados de la herramienta *ERTMD*

La siguiente figura muestra la pantalla inicial que la herramienta de apoyo para evaluación de rendimiento de técnicas de minería de datos presenta al usuario para cargar el directorio con los datos.




Pantalla inicial de la *herramienta ERTMD.*

Las siguientes figuras muestran cómo la herramienta de apoyo para evaluación de rendimiento de técnicas de minería de datos pide al usuario que indique el tipo de pruebas que quiere realizar y cómo la herramienta muestra el progreso de los test que se están realizando.

```
LOADING DATASET...DONE:
25192 records avaiable for training
22544 records avaiable for testing
Choose one of the following options:
Option 1: Compare algorithms amongst themselves with the same settings
Option 2: Compare the performance of one algorithm with different settings
Please introduce the desired option (1,2):
```

Menú inicial de la herramienta *ERTMD*.

```
Please introduce the desired option (1,2):1
Insert a list of the algorithms to compare (LOGREG,KNN,DTREE,GNB,MLPC): LOGREG KNN DTREE
Select an option for feature selection: (PCA,RFE,NONE)
(RFE is not avaiable with KNN, GNB & MLPC):None
Insert the number of splits to perform K-fold (int >=2):10
Please insert the number of neighbors to use in KNN: 5

Logistic_Regression :=======
```

Ejemplo de carga de parámetros de control en la funcionalidad 1 de la herramienta *ERTMD*.

Las siguientes figuras muestran cómo la herramienta de apoyo para evaluación de rendimiento de técnicas de minería de datos genera un archivo con los resultados de rendimiento y el contenido de dicho archivo.

```
Average metrics for Decision_Tree:
Accuracy:       0.9998809208743706
Precision:      0.9666666666666666
Recall:         0.9633333333333335
Fscore:         0.9597979797979797
The file ../data/dataset1/../data/dataset1/all_vs_all_LOGREG_KNN_DTREE_k10_TB.csv has been created.

BYE!
```

Ejemplo de muestra de resultados de la herramienta ERTMD.

TFG > app1 > data > dataset1 > all_vs_all_LOGREG_KNN_DTREE_k10_TB.csv

```
1  Algorithm,Accuracy,Precision,Recall,Fscore
2  Logistic_Regression,99.9960,100.0000,98.0000,98.8889
3  K_Nearest_Neighbour,99.9881,94.1667,98.7500,95.9048
4  Decision_Tree,99.9881,96.6667,96.3333,95.9798
5
```

Ejemplo de archivo de resultados .csv generado por la herramienta ERTMD.

# Resultados de la aplicación *IDS-NET*

Las siguientes figuras muestran un ejemplo de uso de la aplicación *IDS-NET* en el que se introducen una serie de parámetros de control y se genera un archivo con las conexiones consideradas intrusiones.

```
LOADING DATASET...DONE:
25192 records avaiable for training
22544 records avaiable for testing
116 features for each record
Insert a list separated by spaces of the algorithms you wish to use (LOGREG,KNN,DTREE,GNB,MLPC): L
LOGREG KNN
Do you want to use a specific confidence threshold? (Y/N):N
Do you want to perfom PCA or RFE to the DATA?(PCA,RFE,NO)(RFE not avaiable with KNN,GNB & MLPC):N
Please WAIT while training is performed
Do you want to agregate the results of the models?(Y/N)Y
Please WAIT for the results
A .csv file has been created in the folder ../data/dataset1 containing the detected intrusions
It contains the network conexions considered to be intrusions

BYE!
```

Ejemplo de uso de la aplicación *IDS-NET*.

```
1  duration,protocol_type,service,flag,src_bytes,dst_bytes,land,wrong_fragment,urgent,hot,
   num_failed_logins,logged_in,num_compromised,root_shell,su_attempted,num_root,num_file_creations,
   num_shells,num_access_files,num_outbound_cmds,is_host_login,is_guest_login,count,srv_count,
   serror_rate,srv_serror_rate,rerror_rate,srv_rerror_rate,same_srv_rate,diff_srv_rate,
   srv_diff_host_rate,dst_host_count,dst_host_srv_count,dst_host_same_srv_rate,dst_host_diff_srv_rate,
   dst_host_same_src_port_rate,dst_host_srv_diff_host_rate,dst_host_serror_rate,
   dst_host_srv_serror_rate,dst_host_rerror_rate,dst_host_srv_rerror_rate
2  0,tcp,private,SH,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1.0,1.0,0.0,0.0,1.0,0.0,0.0,30,1,0.03,1.0,1
   0,0.0,1.0,1.0,0.0,0.0
3  0,tcp,private,SH,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1.0,1.0,0.0,0.0,1.0,0.0,0.0,80,1,0.01,1.0,1
   0,0.0,1.0,1.0,0.0,0.0
```

Ejemplo de archivo de resultados .csv generado por la aplicación *IDS-NET*.

# Archivo de configuración de la aplicación *IDS-NET-DAEMON*

La siguiente figura 16 muestra el archivo de configuración de la aplicación *IDS-NET-DAEMON*, el cual es crítico y debe seguirse para poner en funcionamiento la aplicación.

```
INSTRUCTIONS TO USE IDS-NET-DAEMON IN DAEMON MODE:

1 - As the root user create the directory /etc/ids

2 - In such directory, always as the root user, copy the following files:
    TFG/app2/data/test/SSH_FTP_ISCX_test.csv
    TFG/app2/data/test/SSH_FTP_ISCX_train.csv
    TFG/app2/src/ids_1.0.py

3 - Edit the following global control variables in the file ids_1.0.py as needed:
    "INTERFACE" => Network interface to be used for analysis
    "CONFIDENCE_THRESHOLD" => Confidence threshold for the predictions

4 - As the root user copy the following file in the directory /etc/systemd/system:
    TFG/app2/src/ids-daemon.service

5 - Reload the systemd units and start the service with the following commands:
    sudo systemctl daemon-reload
    sudo systemctl enable ids-daemon
    sudo systemctl start ids-daemon
    sudo systemctl status ids-daemon

6 - The daemon will email the administrator when suspicius network traffic is identified.
    Such traffic will be logged in the file /etc/ids/ids_intrusions.csv

```

Archivo para la configuración de la aplicación *IDS-NET-DAEMON* (1/3).

```
INSTRUCTIONS TO USE IDS-NET-DAEMON IN DEBUG MODE:

1 - As the root user create the directory /etc/ids

2 - In such directory, always as the root user, copy the following files:
    TFG/app2/data/test/SSH_FTP_ISCX_test.csv
    TFG/app2/data/test/SSH_FTP_ISCX_train.csv
    TFG/app2/src/ids_1.0.py

3 - Edit the following global control variables in the file ids_1.0.py as needed:
    "INTERFACE" => Network interface to be used for analysis
    "CONFIDENCE_THRESHOLD" => Confidence threshold for the predictions

4- Execute the following command and some performance metrics will be displayed:
    sudo python3 /etc/ids/ids-1.0.py DEBUG

DEPENDENCIES:
- python 3.0 or higher
- sklearn library
- numpy library
- pandas library
- Sniffing command line application from the canadian institute of cibersecurity:
  https://gitlab.com/hieulw/cicflowmeter/-/tree/master

```

Archivo para la configuración de la aplicación *IDS-NET-DAEMON* (2/3).

```
DEPENDENCIES:
- python 3.0 or higher
- sklearn library
- numpy library
- pandas library
- Sniffing command line application from the canadian institute of cibersecurity:
    https://gitlab.com/hieulw/cicflowmeter/-/tree/master

CONFIGURATION FOR CICFLOWMETER as of 17th April 2022:
1 - Download the code from the provided link
2 - In the file cicflowmeter/src/cicflowmeter/flow_session.py change the line 86 to the following line:
        flow.add_packet(packet, direction)
3 - Replace the file cicflowmeter/src/cicflowmeter/sniffer.py with the one provided in this directory
4 - Install the dependencies listed in the file requirements.txt with pip3
5 - Build the code ececuting the following command in the cicflowmeter directory:
        python setup.py install
```

Figura 16.3 - Archivo de guía para la configuración de la aplicación *IDS-NET-DAEMON* (1/3).

## Archivo de unidad de servicio de la aplicación *IDS-NET-DAEMON*

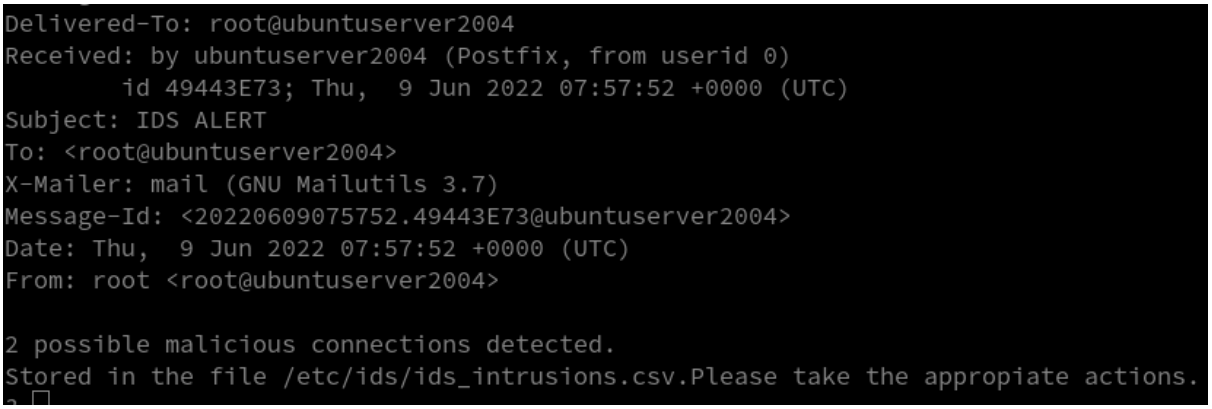
La siguiente figura muestra el archivo de unidad de servicio para la aplicación *IDS-NET-DAEMON* el cual es utilizado por systemd para controlar el demonio.

```
[Unit]
Description=Intrusion detection system
After=network-online.target
Wants=network-online.target

[Service]
ExecStart=/usr/bin/python3 /etc/ids/ids_1.0.py
User=root
Group=root

[Install]
Alias=ids-daemon
WantedBy=multi-user.target

```

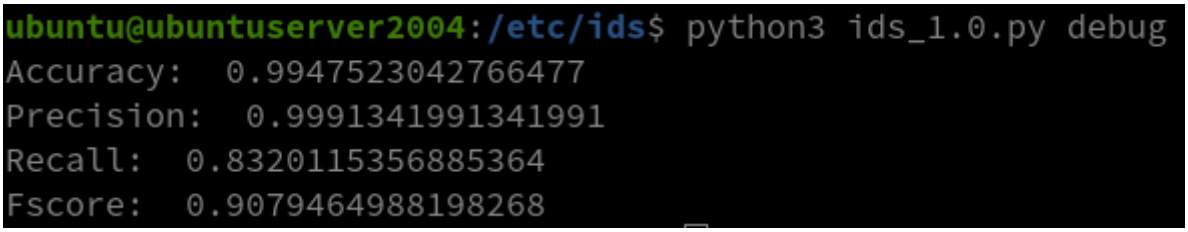
Archivo de unidad de servicio de la aplicación *IDS-NET-DAEMON*.

# Resultados de la aplicación *IDS-NET-DAEMON*

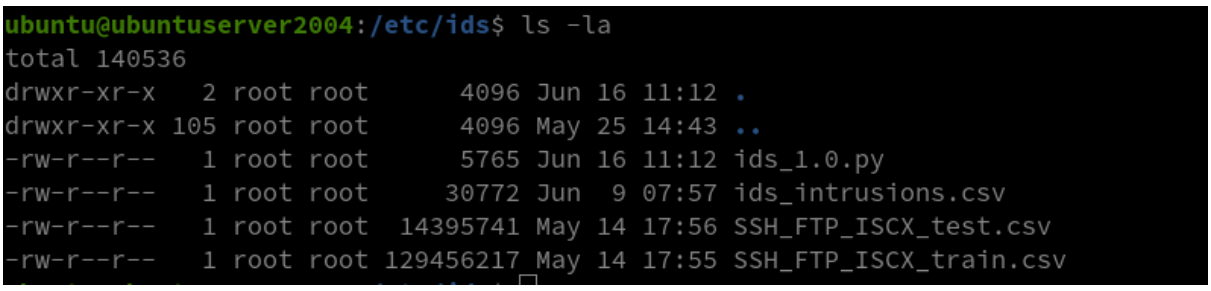
Las siguientes figuras muestran el resultado de la aplicación *IDS-NET-DAEMON*. Respectivamente muestran cómo la aplicación avisa por email al administrador del sistema de posibles intrusiones, cómo la aplicación puede realizar una prueba de rendimiento y el contenido del directorio desde el cual "opera" la aplicación.




Ejemplo de email enviado por la aplicación *IDS-NET-DAEMON*.



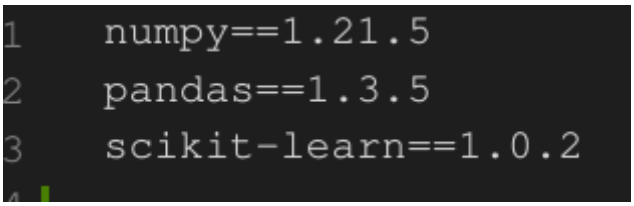

Ejemplo de modo debug de la aplicación *IDS-NET-DAEMON*.




Directorio de control /etc/ids de la aplicación *IDS-NET-DAEMON*.

## Dependencias de las aplicaciones

La siguiente figura muestra el contenido de un archivo requirements.txt el cual indica las librerías de las cuales depende el código y que deben ser instaladas utilizando pip.




Archivo “requirements.txt” para las aplicaciones.