1 Objetivos

- Aplicar los conocimientos vistos en clase sobre las técnicas de análisis estático de malware
- Implementar el algoritmo K-means para la clasificación de familias de malware
- Determinar la similitud entre las familias de malware encontradas

2 Preámbulo

FAMILIAS DE MALWARE

Para que un malware se considere parte de una familia no es necesario que el código sea 100% idéntico. Otras diferencias comunes entre el los miembros de una familia están relacionadas con la configuración, las direcciones para el C&C, y las características con que evolucionan, de ello se pueden desprender nuevas sub-familias.

Por ejemplo, en el 2022 los investigadores de Mandiant identificaron 49 **familias nuevas** de malware mensualmente, sin embargo, solamente 321 familias fueron detectadas en los incidentes. Esto es importante porque permite implementar controles similares en base a riesgos ya conocidos, y una respuesta a incidentes más rápida en base a las tácticas y técnicas conocidas de una familia de malware.

3 Desarrollo

El laboratorio consiste en la **creación** de un dataset de características a partir de ejemplos de malware proporcionados. Existen familias entre los ejemplos y se deben determinar que familias existen entre ellos.

Se debe entregar un enlace a un repositorio de GitHub con un notebook donde se detalle: la creación del dataset, implementación del algoritmo K-means, cálculo de número óptimo de clústeres, evaluación de clústeres, y la similitud del malware que pertenezca a cada grupo. Se debe explicar cómo se obtuvo el número óptimo de familias posibles.

NOTA: se proporcionan ejemplos reales de malware, para efectos de aplicar los conocimientos académicos de análisis estático de malware, y es responsabilidad del alumno(a) cualquier uso adicional que no sea el indicado en este laboratorio. Luego de finalizar el laboratorio se deben eliminar todos los ejemplares.

Se proporciona una carpeta con el nombre MALWR_lab4.zip en CANVAS, la cual posee la contraseña *infected*

Para los usuarios de Windows se debe utilizar una VM con Linux para trabajar. Se debe descargar el archivo y descomprimirlo en la ubicación deseada. Luego se debe descomprimirlo y NO se debe manipular manualmente ningún archivo, pues existe el riesgo de ejecutarlo e infectarse.

Parte 1

Creación del dataset

Se debe realizar un análisis estático utilizando la herramienta pefile sobre los archivos de malware proporcionados. Con la información que se obtenga del análisis se construirá el dataset inicial. Recuerde lo aprendido sobre el PE header, el nombre de las secciones, las llamadas a las funciones que realiza, etc. Usted define la información a incluir.

Exploración y pre-procesamiento de datos

Analice el dataset y determine qué técnicas de pre-procesamiento son necesarias aplicar a las columnas para que los algoritmos de aprendizaje no-supervisado puedan manipular la data.

Parte 2

Implementación del modelo

Utilice dos algoritmos de partición para crear los clústeres a partir del dataset. Para cada algoritmo, utilice el método del codo, genere la gráfica del error contra K (número de clústeres) para determinar de forma empírica el número óptimo de clústeres que hay (explique su razonamiento).

Luego calcule el coeficiente de Silhouette, analice la gráfica del coeficiente contra K y utilícela para determinar de mejor manera el número de clústeres o familias de malware que hay.

Etiquete cada observación según el clúster indicado por los algoritmos.

Utilizando Gemini, genere los embeddings de la información de su dataset, utilice un modelo de análisis de texto con una tarea de tipo clustering. Aplique reducción de dimensionalidades y trate de gráficar en 2 dimensiones la data para tratar de agrupar la data.

Análisis de similitud

Utilice el índice de Jaccard para encontrar la similitud del malware que pertenece a cada familia (realice el análisis con al menos tres umbrales distintos de similitud), utilizado dos características distintas: strings y llamadas a las funciones. Para cada una de las características

- Genere un grafo para cada familia.
- Genere un grafo para todo el conjunto de los ejemplos de malware.

Conclusiones

1. Para ambos algoritmos, ¿para qué número de clústeres se obtiene el coeficiente de Silhouette más alto?
2. Para ambos algoritmos, ¿En que medida coincide el coeficiente de Silhouette con el método del codo?
3. Según los resultados obtenidos de ambos algoritmos ¿Cuántas familias cree que existen entre los ejemplares de malware proporcionados?
4. ¿En qué medida coincide el análisis de similitud con las familias encontradas utilizando los algoritmos de partición, para ambas características (strings, llamadas a las funciones)?

Entregable

1. Enlace de GitHub con el notebook donde se realizó el laboratorio
2. Grafos en formato .png
3. Gráfica con la comparación de embeddings
4. Reporte escrito (puede ser en el mismo notebook), donde se debe describir el proceso realizado y la respuesta a las preguntas, **O** un video donde se explique el trabajo realizado y la respuestas a las preguntas con diapositivas u otra forma de presentación. Puede ser publicado en YouTube u otros sitios afines.

Rúbrica

Aspecto	**Punteo (sobre 100 pts)**
Creación del dataset	15
Pre-procesamiento	10
Gemini embeddings	15
Implementación de los modelos de participación (7.5 pts c/u)	15
Análisis de similitud	15
Comparación de modelos	30