Datos, ciberseguridad e inteligencia artificial
Fecha: 23-06-2025.
Alumno: Taisen Romero Bañuelos.

Detección y análisis de malware e intrusiones en la red

Creo que los aprendizajes de esta práctica que más tengo en cuenta son sobre los modelos de SVM. Por ejemplo, la selección del kernel me llamó la atención porque pese a que se usó el kernel que mejores resultados daba eso no impidió que el modelo no generalizó bien al probarse con datos reales, mostrando signos de sobreajuste. Me pregunto si no hubiéramos notado eso si no usábamos los datos de validación (el último dataset, el de los datos reales). Adicionalmente, también me pregunto si las técnicas para evaluar un modelo de ML también serían aplicables a este contexto (para no depender de un segundo dataset de prueba).
También me pareció interesante como con RF se alcanzó una precisión del 99% incluso con un dataset pequeño, de hecho, me pareció bastante conveniente. Es decir, ¿Qué es más probable, que el modelo de una buena precisión pero que su rendimiento no sea realmente bueno, o que haya sobreajuste o que realmente el modelo sea tan bueno como sugieren sus métricas?, creo que sería interesante implementar la métrica kappa para estos problemas.
Y hablando de cosas un poco más generales, también fue interesante la recopilación de información sobre datasets públicos. De hecho, se me ocurrió que de cara a los fines educativos uno mismo podría generar datasets interesantes con wireshark. Hay un sitios que son una especie de wikipedia de malware (exploitDB, vulnhub, por ejemplo), que incluyen el ID de las vulnerabilidades conocidas hasta el momento y bastante información relativa a esas vulnerabilidades, a veces hasta incluyen cosas del propio malware, entonces, con los recursos disponibles en Exploit Database, por ejemplo, se podrían extraer cosas de malware para que en un entorno controlado se simule un tráfico de red y con eso generamos datos. Pienso que no sería tan difícil ya que la cantidad de tráfico que se puede capturar con WireShark es absurda, por lo que es más probable que tengamos una maldición de la dimensionalidad a que nos quedemos con un dataset pequeño. También, creo que sería interesante una práctica donde tengamos nuestro modelo de ML de clasificación y detección de malware y analicemos la efectividad de ambos ante un malware. Y luego observar cómo se podría modificar dicho malware para burlar el modelo de clasificación (que según entendí, es el que tiene mejor rendimiento si lo comparamos con un enfoque de detección). Quizás la manera de poder burlar el sistema sea haciendo uso de la "capa 8" o haciendo alguna especie de enmascaramiento del tráfico de red.
Y por último, creo que lo más importante que me llevo es la lección de que la selección de características (o ingeniería de características) es casi tan importante como la propia calidad de los datos.

Dato curioso: Lo que hacía antes era hacer un escaneo de vulnerabilidades con Nmap o Nessus y en el propio escaneo se mostraba el "ID" de las vulnerabilidades encontradas para que luego se busque en ExploitDB información al respecto. Entonces, si quisiéramos generar datos a partir de un entorno simulado creo que se podría hacer de forma realista aprovechando las vulnerabilidades que se detecten en ese momento.