REFLEXIÓN

Act 3.4 - Actividad Integral de BST (Evidencia Competencia)

Competencias

SICT0301B - Evalúa los componentes que integran una problemática de acuerdo a principios y procesos computacionales.
SICT0302B - Toma decisiones en la solución de problemas en condiciones de incertidumbre y diferentes niveles de complejidad con base metodologías de investigación y de cómputo.
SICT0303B - Implementa acciones científicas e ingenieriles o procesos computacionales que cumplen con el tipo de solución requerida.
SEG0702A - Tecnologías de Vanguardia. Evalúa diversas tecnologías con apertura a la búsqueda e implementación de alternativas relevantes en la transformación de la práctica profesional.

Estefania Perez Yeo
a01639270@itesm.mx

Como parte de la situación problema es importante identificar aquellas direcciones IP, por las que se han accedido en diferentes puertos; ya que de esta manera será más fácil encontrar algún ataque.

El ordenamiento de la bitácora con la información ser analizada es importante que se encuentre ordenada; aunque a pesar de las diferentes formas para llegar a un mismo punto; ahora es requerido basar el ordenamiento por su dirección IP, por ende se ha tomado la decisión de seguir con la metodología del Heap Sort.

Heap Sort es un método de ordenamiento comparativo basado es la estructura de datos en los Binary Heap; este suele ser dividido en dos partes: aquella identificada como heapify con una complejidad de tiempo de O(logn), y otra parte conocida como créate and Build Heap con una complejidad de tiempo de O(n); donde por consecuencia Heap Sort resulta con una complejidad de O(nlogn).

Si un Binary Heap, es un Complete Binary Tree, entonces se sabe que este puede llegar a tener dos tipos de orden en el acomodo de sus nodos. Por ejemplo, si se busca que el nodo padre sea mayor al de sus hijos, a este orden se le conoce como Max Heap, y en el caso contrario, donde se busca que el nodo padre sea menor que sus hijos, se le conoce como Min Heap.

Para la solución, en cuanto a saber qué IP’s son las más repetitivas, se ha decidido utilizar un Max Heap, el cual se aborda en tres partes, la primera siendo top(), el cual se encarga de obtener el nodo más grande, o en dicho caso la raíz, y maneja una complejidad de O(1), como segundo el método pop(), es el encargado de eliminar el nodo mayor o raíz, el cual maneja una complejidad de O(logn), y como tercero y último, aquel encargado de insertar el valor hasta el final y recorrerlo hacia arriba, maneja una complejidad de O(logn)

Aunque ¿cómo es que a partir de esto se puede detectar si una red está infectada? S ele conoce al Botnet como aquellas red de equipos infectados, los cuales es su mayoría mandan virus, spam, entre otros. Estos son detectados por un filtro de tráfico donde por lo general se le conoce como la detección del C&C (Command and Copy) puesto que los Botnets, al no utilizar servidores IRC, optan por

utilizar conexiones a puertos anormalmente altos, donde con altos se refiere a por encima de las 10,000 conexiones. He aquí la clave de identificar la repetición de las IP dentro de las bitácoras, y en especial aquellas 5 identificadas por el Max Heap con más accesos.

Liga de Replit: https://replit.com/join/ffrdlipucl-estefaniapy

Bibliografía

Fírvida, D. (22 de junio de 2014). *Localizando bots en una red local*. Obtenido de INCIBE: https://www.incibe-cert.es/blog/localizando-bots-en-una-lan

Geeks for Geeks. (14 de septiembre de 2021). *HeapSort*. Obtenido de Geeks for Geeks Organization: https://www.geeksforgeeks.org/heap-sort/