



**Tecnológico  
de Monterrey**

Campus Estado de México

Reflexión integradora 5

Samuel Rincón V. A01752573

Programacion de estructuras de datos y algoritmos fundamentales

06/05/2025

## 1. Importancia y eficiencia del uso de tablas hash

En esta actividad, el uso de una tabla hash con direccionamiento abierto y prueba cuadrática permite almacenar eficientemente un resumen de conexiones entre direcciones IP, evitando recorrer listas lineales para consultar información crítica como el número de accesos salientes y entrantes.

Las tablas hash son especialmente útiles en escenarios con gran cantidad de datos (más de 90,000 incidencias en este caso), ya que proporcionan acceso promedio en tiempo  $O(1)$  si se mantiene un buen factor de carga. En contraste, estructuras como listas enlazadas o mapas ordenados (`std::map`) requieren  $O(n)$  o  $O(\log n)$  respectivamente.

Gracias a la eficiencia de búsqueda, inserción y actualización que ofrecen las tablas hash, fue posible generar un resumen para cada IP en tiempo óptimo, incluso con decenas de miles de registros.

## 2. Complejidad computacional y factor de carga

La complejidad de las operaciones en la tabla hash depende fuertemente del número de colisiones, las cuales están directamente relacionadas con el factor de carga  $\alpha = n / m$  ( $n$  elementos,  $m$  tamaño de tabla).

- Si  $\alpha = 0.5$ , las colisiones son mínimas y la eficiencia es casi constante.
- Si  $\alpha = 1$  o más, las colisiones aumentan drásticamente, degradando el rendimiento hacia  $O(n)$ .

Esto se evidencia al comparar el número de colisiones con distintos tamaños de tabla:

Tamaño de Tabla Hash ( $m$ )	Elementos insertados ( $n$ )	Factor de Carga ( $\alpha$ )	Colisiones
10007	13470	1.34	8259
20011	13470	0.67	4523
40009	13470	0.34	1321

Como se observa, al duplicar el tamaño de la tabla, las colisiones disminuyen significativamente, validando que un buen diseño de tabla hash considera un tamaño primo suficientemente grande para mantener bajo el valor de  $\alpha$ .

### 3. Detección de IP sospechosa por posible DDoS

Una IP con muchos más accesos salientes que entrantes puede ser señal de actividad maliciosa, como intentos de ataque DDoS. Se buscó la IP con mayor relación de salientes/entrantes, utilizando el método `getIPSummary()`.

La IP 73.89.221.25 presentó el siguiente comportamiento:

- **Entrantes:** 7
- **Salientes:** 18
- **Relación:** 2.57143

Aunque otras IPs tenían más conexiones, esta mostró una relación anómala de salientes sobre entrantes, siendo significativamente más alta que la media. Además, sus conexiones fueron dirigidas a múltiples destinos únicos, lo cual refuerza la hipótesis de intento de saturación a distintas direcciones, un patrón común en ataques distribuidos.

Por tanto, 73.89.221.25 puede ser considerada sospechosa de participar en un ataque DDoS.

### 4. Conclusiones

El uso de una tabla hash con prueba cuadrática fue crucial para lograr eficiencia en esta actividad. Además, el análisis del comportamiento de las IPs no solo permitió evaluar la eficiencia algorítmica, sino también detectar posibles amenazas cibernéticas con un enfoque estructurado y computacional.

La correcta elección del tamaño de la tabla y la comprensión del impacto del factor de carga son elementos clave para mantener la eficiencia y confiabilidad de sistemas que procesan grandes volúmenes de datos.

## 5. Referencias

- Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to Algorithms* (3rd ed.). MIT Press.
- Weiss, M. A. (2014). *Data Structures and Algorithm Analysis in C++* (4th ed.). Pearson.
- GeeksforGeeks. (2024). Quadratic Probing in Hashing
- cppreference.com. (2025). `std::hash`
- LaFore, R. (2002). *Data Structures and Algorithms in C++*. SAMS Publishing.