Tecnológico de Monterrey

**Programación de estructuras de datos y algoritmos fundamentales**

# Act/Reto 3 - Actividad Integral sobre el uso de conjuntos y diccionarios (Evidencia Competencia)

**Javier Corona Del Río A01023063**

**Kevin López Cano A01028138**

**Se respondieron las preguntas de la siguiente manera:**

1. **Hay algún nombre de dominio que sea anómalo (Esto puede ser con inspección visual).**

Se realizó una inspección visual y se determinó que los siguientes dos dominios son anómalos.

R//Las conexiones anómalas son:

- **"3hdppu5kkb36hua85azm.org"**
- **"rn2wzlj4fwo5rioc8egp.xxx" .**

2. **De los nombres de dominio encontrados en el paso anterior, ¿cuál es su ip? ¿Cómo determinarías esta información de la manera más eficiente en complejidad temporal?**

Para responder correctamente la dos y encontrar las ips de las conexiones anómalas en la complejidad temporal más eficiente recorrimos nuestro vector "DataBase" y cuando las encontramos hicimos un break que finaliza la búsqueda.

R//Las ips de las conexiones anómalas son:

- **“3hdppu5kkb36hua85azm.org”: 171.172.170.23**
- **“rn2wzlj4fwo5rioc8egp.xxx”: 44.167.41.91**

La parte de nuestro código que se utilizó para ello fue:

```
cout << "2) Las ip de las conexiones anomalas son:" << endl;
cout << endl;
vector <string> Anomalos;
int p = 0;
while(p < DataBase.size())
{
    if(DataBase[p].getHostname_destine() == "3hdppu5kkb36hua85azm.org")
    {
        Anomalos.push_back(DataBase[p].getIPDestine());
        cout << "3hdppu5kkb36hua85azm.org: " << DataBase[p].getIPDestine() << endl;
    }
    if(DataBase[p].getHostname_destine() == "rn2wzlj4fwo5rioc8egp.xxx")
    {
        Anomalos.push_back(DataBase[p].getIPDestine());
        cout << "rn2wzlj4fwo5rioc8egp.xxx: " << DataBase[p].getIPDestine() << endl;
    }
    if(Anomalos.size() == 2)
        break;
    p++;
}
cout << endl;
```

3. **De las computadoras pertenecientes al dominio reto.com determina la cantidad de ips que tienen al menos una conexión entrante. (Recuerda que ya tienes la dirección de la red y el último octeto puede tener computadoras del .1 al .254). Imprime la cantidad de computadoras.**

Para responder la pregunta 3, cada vez que el size de un stack es mayor que cero nos indica que tiene conexiones entrantes por lo que aumentamos un contador.

**R//La cantidad de ips que tienen al menos una conexión entrante es: 32**

Con el fin de obtener el resultado de la pregunta anterior se hizo esta parte de código:

```
ik = Diccionario.begin();
int contador = 0;
while(ik != Diccionario.end())
{
    string cadena = ik->first;
    int l = cadena.find_first_of(".\\");
    string result = cadena.substr(l, cadena.size()-1);
    if(result == ".reto.com")
    {
        if(ik->second.getStack().size() > 0)
        {
            contador++;
        }
    }
    ik++;
}
cout << "3) La cantidad de ips que tienen al menos una conexión entrante es: " << contador  << endl;
cout << endl;
```

4. **Toma algunas computadoras que no sean server.reto.com o el servidor dhcp. Pueden ser entre 5 y 150. Obtén las ip únicas de las conexiones entrantes.**

Para esta pregunta almacenamos las conexiones entrantes de las computadoras seleccionadas en un set llamado ips.

**R//Las ip únicas de las conexiones entrantes están en un set de tamaño: 33.**

Se obtuvieron las ip únicas sin contar server.reto.com o dhcp y se determinó lo siguiente.

```
ik = Diccionario.begin();
set<string> ips;
while(ik != Diccionario.end())
{
    stack<string> stack2 = ik->second.getStack();
    while(stack2.empty() == false)
    {
        if(stack2.top() != "-")
       {
            ips.insert(stack2.top());
       }
       stack2.pop();
    }
    ik++;
}

cout << "4) Las ip únicas de las conexiones entrantes estan en un set de tamaño: "<< ips.size() << endl;
cout << endl;
```

5. **Considerando el resultado de las preguntas 3 y 4, ¿Qué crees que esté ocurriendo en esta red? (Pregunta sin código)**

**R//Podemos decir entonces que posiblemente algunos servidores están navegando en páginas desconocidas, lo cual puede significar ataques de bots.**

6. **Para las ips encontradas en el paso anterior, determina si se han comunicado con los datos encontrados en la pregunta 1.**

Finalmente si alguna de las ips guardadas en "ips" coincide con las ips de las conexiones anómalas podemos saber si estas tuvieron comunicación.

**R//No, no se han comunicado los datos encontrados en la pregunta 1 con las ips encontradas en el paso anterior.**

En la pregunta 6 no se han comunicado los datos encontrados en la pregunta 1 con las ips encontradas en el paso anterior. Por eso la respuesta es no.

```
it = ips.begin();
int p6 = 0;
while(it != ips.end())
{
    if((*it) == "171.172.170.23")
        p6++;
    if((*it) == "44.167.41.91")
        p6++;
    it++;
}
if(p6 == 0)
    cout << "6) No, no se han comunicado los datos encontrados en la pregunta 1 con las ips encontradas en el paso anterior." << endl;
if(p6 > 0)
    cout << "6) Si, se han comunicado los datos encontrados en la pregunta 1 con las ips encontradas en el paso anterior y han tenido " << p6 << " conexiones." << endl;

return 0;
```

7. **(Extra): En caso de que hayas encontrado que las computadoras del paso 1 y 4 se comunican, determina en qué fecha ocurre la primera comunicación entre estas 2 y qué protocolo se usó.**

**R//Con las computadoras que elegimos en el paso anterior, se puede decir que no tuvieron comunicación con las del paso 1 por lo que esta pregunta no aplica.**

**Reflexión individual:**

En definitiva, el uso de conjuntos y diccionarios facilita mucho el manejo de datos porque asocia sus valores a una palabra clave lo que permite no repetir valores en el caso de los conjuntos o contabilizar la cantidad de elementos para el caso de los diccionarios. Este reto, me sirvió mucho para entender la manipulación de datos y sin duda será algo que aplicaré en el ámbito profesional.