VARENA-TRUST: UNA APLICACIÓN PARA GARANTIZAR PRIVACIDAD EN SISTEMAS P2P DHT

Resumen

Este artículo presenta el diseño y evaluación de Varena-Trust, una aplicación implementada bajo el API de freePastry y diseñada para reducir los problemas asociados a la privacidad en sistemas Peer-to-Peer DHT. Varena-Trust maneja un concepto de reputación en cada uno de los nodos, el cual aumenta o disminuye de acuerdo a las operaciones que un nodo realice o la opinión que sus vecinos tengan acerca de él, con el cual se busca garantizar un control de acceso para los contenidos que se encuentran distribuidos en el DHT.

Palabras Clave: P2P, DHT

1. Introducción

Los sistemas Peer-to-Peer y sus aplicaciones son sistemas distribuidos en los cuales no existen controles de seguridad y todos los participantes pueden acceder a los contenidos que se encuentran distribuidos dentro de estos.

Es por esto que surge la necesidad de crear un mecanismo de control de acceso el cual garantice que los contenidos solamente sean consultados por nodos autorizados y de igual forma, sólo puedan ser modificados por nodos autorizados a hacerlo. Lo anterior busca cumplir con dos propiedades de la privacidad de la información: la confidencialidad y la integridad de los datos.

Existen varias implementaciones que buscan dar soporte de privacidad a los sistemas actuales. Estas implementaciones se conocen como "Sistemas de Gestión de Reputación" en el que se definen algoritmos que permitan a los usuarios (peers) conocer a otros en función de sopesar la opinión de aquellos con quienes interactuaron en el pasado. Con esto, se persigue un objetivo: Minimizar el número de contactos (sesiones de trabajo o colaboración) con usuarios malintencionados.

Varena-Trust es una aplicación que busca garantizar las dos propiedades mencionadas anteriormente, agregándole un atributo de reputación a cada nodo y colocándole controles de consulta y modificación a los contenidos almacenados.

El resto de este artículo tiene la estructura descrita a continuación: La sección 2 compara Varena-Trust con otras aplicaciones ya existentes, la sección 3 presenta el modelo bajo el cual se implementó Varena-Trust, la sección 4 muestra resultados experimentales para medir el desempeño de Varena-Trust y finalmente, la sección 5 presenta trabajo a futuro relacionado con la temática de privacidad en los sistemas Peer-to-Peer.

2. Trabajo relacionado

Los sistemas de gestión de la reputación, desarrollan algoritmos para ayudar a los usuarios a obtener con la mayor precisión el grado de fiabilidad de otros usuarios. El nivel de privacidad del sistema es inversamente proporcional al nivel de reputación de un nodo, es decir, entre mayor sea el nivel de reputación, menor es el nivel de privacidad asignado a un nodo particular. El objetivo final es el aislamiento paulatino de usuarios malintencionados (poco fiables) para aumentar la confianza en las aplicaciones. Con el propósito de encontrar el mejor enfoque para definir el modelo de reputación que se utilizaría, se analizaron las siguientes alternativas que dieron el marco conceptual y teórico para el desarrollo de Varena-Trust.

- EigenTrust [1]: Aplica un concepto de "Confianza Transitiva", según el cual, un usuario i tendrá una buena opinión de aquellos otros usuarios que le hayan proporcionado archivos no corrompidos. Incluso, dicho peer tenderá a confiar en las opiniones de esos mismos usuarios, debido a que si han sido honestos con los archivos que ellos mismos proveen deberán serlo en cuanto a la información que reporten sobre sus valores de confianza local. La reputación total de cada peer i viene dada por los valores de confianza local asignados a i por otros peers, y ponderados por las reputaciones globales de quienes los asignan. La forma natural es que el usuario i pregunte a sus "conocidos" acerca de sus opiniones sobre el peer en cuestión, y en función de su confianza en éstos dar un mayor o menor peso a esas opiniones.

Valor de Confianza Local (local trust value): sij

El valor de la confianza local está dado por la suma de las tasas de cada una de las descargas que el peer i ha recibido del peer j.

Tasa de descarga positiva: tr(i,j)=1

Tasa de descarga negativa: tr(i,j)=-1

$$s_{ij} = \sum tr_{ij} = sat(i,j) - unsat(i,j)$$

Este planteamiento da origen a un problema en relación con la agregación de valores de confianza local s_{ij} sin almacenarlos ni gestionarlos de manera centralizada. Este problema podría ser resuelto mediante la agregación de los valores de confianza local de todos los usuarios de manera natural, con el mínimo coste en términos de complejidad de los mensajes.

Para determinar la "Suma de los Valores de Confianza Local" de una manera natural es que el usuario i pregunte a sus "conocidos" acerca de sus opiniones sobre el peer en cuestión, y en función de su confianza en éstos dar un mayor o menor peso a esas opiniones. Está determinado por la siguiente función:

$$t_{ik} = \sum_j c_{ij} c_{jk} \qquad \vec{t} = \left(C^T\right) \vec{c}_i$$

Un parámetro adicional en el modelo propuesto por EigenTrust, es la tasa de confianza del peer i en el k en base a las opiniones de los conocidos. Se puede profundizar aún más y preguntar a los conocidos de nuestros conocidos (Obtendremos una visión más completa de la red después de muchas iteraciones "n").

$$\vec{t} = \left(C^T\right)^n \vec{c}_i$$

Algoritmo Eigen Trust Distribuido:

Para cada peer i {

-Primero, preguntar a los usuarios que han descargado archivos de i sus opiniones. (tj(0)=pj)

-Mientras no converja repetir {

-Cálculo del valor de confianza global actual del peer i: $ti^{(k+1)}=(1-a)(c_{(1i)}t1^{(k)}+..+c_{(qi)}tq^{(k)})+ap_i$

-Envía tu opinion cij y el valor de confianza global ti(k+1) a los peers de quienes has descargado archivos.

-Espera a que tus conocidos te envíen sus valores de confianza y opiniones actualizados.

}

}

- HPRS [2]: El uso generalizado de las redes Peer-to-peer produce inevitablemente Peers maliciosos que tratan de alterar la red con inundaciones de archivos no auténticos. Ello lleva a la necesidad de un sistema de reputación sólido para identificar Peers y archivos maliciosos. Se presenta un novedoso sistema híbrido de reputación de sistemas P2P (HPRS) con los peers y calificación de archivos. Esta calificación combinada permite a los Peers catalogados como “buenos” juzgar los archivos de manera individual y así eliminar las descargas no auténticas. También ayuda a un peer sacar el máximo partido al sistema que le permita obtener los archivos buenos de sus Peers catalogados como “malos” y rechazar solamente los archivos calificados negativamente. Resultados de la simulación muestran que HPRS mejora significativamente el rendimiento en una red.

- Gossip based reputation [3]: Los sistemas de reputación Peer-to-Peer (P2P) son necesarios para evaluar la confiabilidad de los Peers participantes y para combatir los comportamientos “egoístas y malintencionados” de los Peers. El sistema de reputación recolecta datos de retroalimentación generados localmente en los

Peers y los agrega para dar las puntuaciones de reputación a nivel global. El desarrollo de un sistema de reputación descentralizado tiene una gran demanda para las redes P2P no estructuradas ya que la mayoría de aplicaciones P2P en Internet son no estructuradas. A falta de una rápida dispersión (Hashing) y mecanismos de búsqueda, la forma de realizar la agregación de la reputación de eficiencia es un reto importante en la computación P2P no estructurada. GossipTrust calcula la puntuación de reputación global de todos los nodos al mismo tiempo. Al recurrir a un protocolo en modo promiscuo y aprovechando los super-nodos, GossipTrust se adapta a la dinámica de los Peers y es robusto a las perturbaciones de los Peers malintencionados. Experimentos de simulación muestran un sistema escalable, preciso, robusto y tolerante a fallos. Estos resultados demuestran las ventajas reivindicadas de agregación, la eficiencia de almacenamiento, y es de anotar la exactitud en las redes P2P no estructuradas. Con pequeñas modificaciones, el sistema también es aplicable a sistemas estructurados de P2P con el mejor desempeño proyectado.

3. Modelo

Por ser Varena-Trust una aplicación de manejo de contenido que implementa consultas de igual dad de metadatos (del tipo Metadatos=Valor), en el algoritmo propuesto para determinar la reputación se definió una variable para determinar el nivel de participación de los nodos en relación con el almacenamiento de objetos y metadatos en el sistema. En esta propuesta, los nodos pueden evaluar el grado de satisfacción alcanzado tras su interacción con otros nodos, tanto en el almacenamiento de objetos y metadatos como en la resolución satisfactoria de consultas de igualdad de metadatos asociados a documentos.

Otro parámetro a tener en cuenta para determinar la reputación, está relacionado con la votación global del sistema que se obtiene por la sumatoria de las votaciones de los nodos en función de las operaciones realizadas entre los nodos que interactúan en el almacenamiento y consulta.

Como un criterio a tener en cuenta en nuestra propuesta de implementación de un sistema de gestión de reputación, está relacionada con la estrategia de EigenTrust en cuanto al manejo de la reputación local de manera descentralizada para el cálculo de confianza transitiva.

3.1. Implementación de Confianza Transitiva

Como estrategia para determinar la confianza transitiva de manera distribuida, nos apoyamos en la tabla de enrutamiento “RoutingTable” de cada nodo. Mediante esta tabla de enrutamiento, implementamos la siguiente política de votación:

- Dado un nodo A que contiene un objeto, el nodo B recibe una votación de A, cuando B almacena un metadato del objeto.

- Si el nodo B, resuelve una consulta en relación con el nodo A, el nodo B recibe una votación de A.

- La votación del nodo A al nodo B no se hace directa de nodo a nodo sino distribuida mediante una tabla de enrutamiento.

- La votación del nodo A se hace a través de la tabla de enrutamiento "RoutingTable" para lograr hacer una votación al nodo B.

- Para cada Nodo en la tabla de enrutamiento del nodo A se determina su reputación.

- Si en el enrutamiento de A hacia B, encuentra un nodo con reputación negativa, no tiene lugar la votación de A al nodo B.

- Si en el enrutamiento de A hacia B, un nodo X tiene una reputación positiva, se determina si existe una votación del nodo X al nodo B.

- Si existe una votación de X al nodo B, la votación de A hacia B es el resultado de la suma de la votación de A más la votación de X. Por Ejemplo:

Votación (Caso 1):

A -> B = Positiva (1)

X -> B = Positiva (1)

Votación recibida por B: 2

Votación (Caso 2):

A -> B = Positiva (1)

X -> B = Negativa (-1)

Votación recibida por B: 0

Votación (Caso 3):

A -> B = Negativa (-1)

X -> B = Negativa (-1)

Votación recibida por B: -2

3.2. Consultas y Modificaciones por Reputación Distribuida

Varena-Trust está en capacidad de realizar consultas y modificaciones con restricción de reputación, esto con el fin de garantizar las propiedades de confidencialidad e integridad de la información.

Para esto Varena-Trust implementa un método que revisa si cada uno de los nodos involucrados en la resolución de una consulta cumple con la restricción de reputación establecida por el objeto que se va a consultar o modificar. Para clarificar la idea, miremos el siguiente ejemplo:

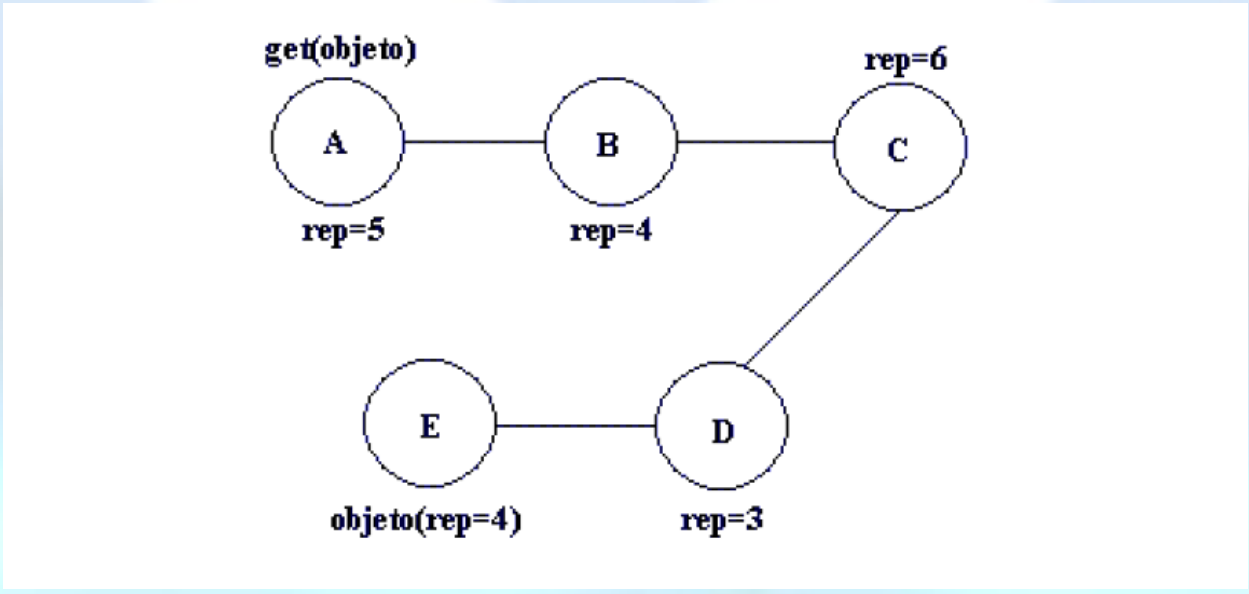

FIGURA 1. CONSULTA DE REPUTACION

Para obtener el objeto del nodo E, el nodo A tiene que pasar primero por los nodos B,C y D. Como se puede observar en la figura 1, el objeto consultado exige una reputación de 4, la cual es cumplida por todos los nodos que se encuentran en el camino menos por el nodo D, que tiene una reputación de 3, por esta razón Varena-Trust no permite la ejecución de la consulta, lo cual nos garantiza la propiedad de la confidencialidad para los contenidos almacenados.

Del mismo modo, para que un nodo pueda modificar los metadatos de un objeto, éste debe cumplir la restricción de reputación establecida para modificar ese objeto.

3.3. Arquitectura de la Aplicación

La figura 2, muestra la arquitectura de nuestro sistema. Con el propósito de modelar un diseño arquitectónico que nos garantice un aislamiento de la lógica de FreePastry se acogieron las mejoras prácticas de diseño y se incorporaron patrones de creación y de estructura en un entorno multicapa donde el procesamiento se distribuye entre las diferentes capas a través de componentes portables, escalables y eficientes en los que se fundamenta nuestra arquitectura.

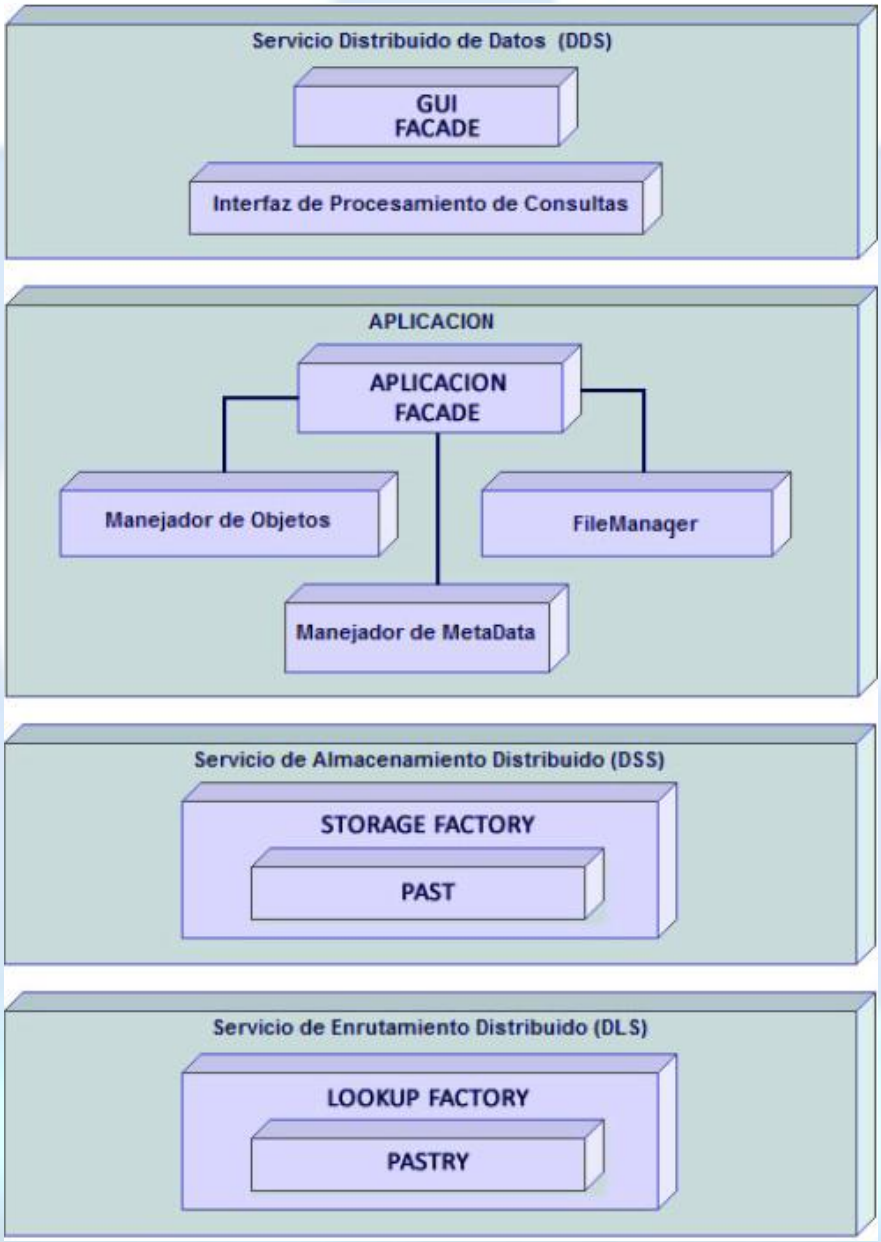

FIGURA 2. ARQUITECTURA DE LA APLICACION

4. Desempeño

Para medir el desempeño del sistema, se realizaron 50 consultas con reputación simple (de un solo término, de la forma Metadato=Valor), se midieron los tiempos de resolución de esas consultas. Del mismo modo para el escenario descrito anteriormente, se realizaron 50 consultas normales simples sin restricción de reputación y se midieron los tiempos de resolución de esas consultas, lo anterior para observar el costo de tener implementada una solución con reputación en un sistema P2P. La unidad de medida es tiempo en milisegundos:

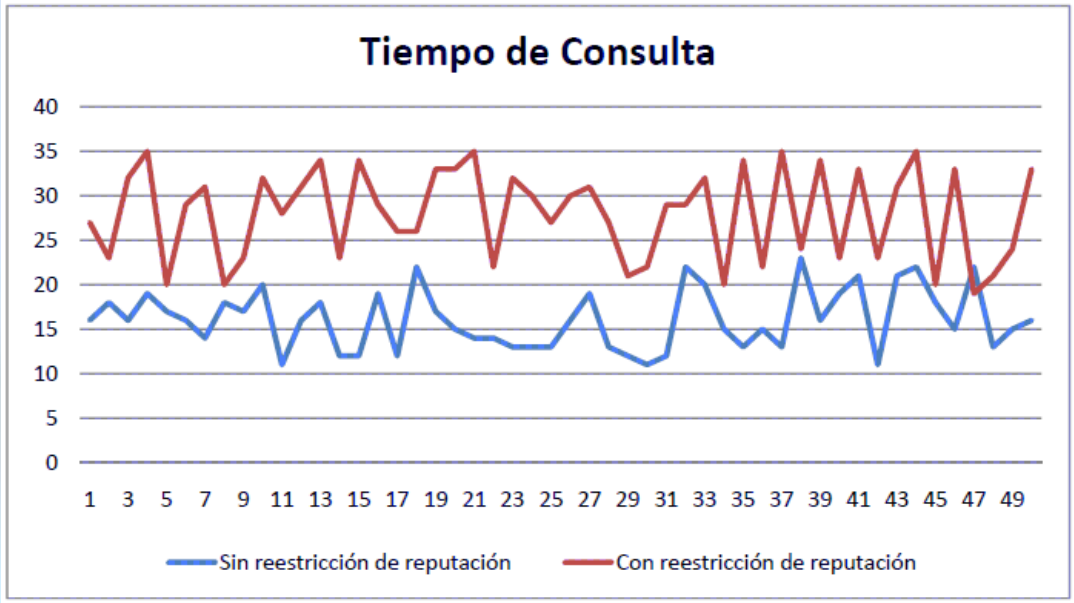

FIGURA 2. DESEMPEÑO DE LA APLICACION

Como se observa en la Figura 3, el punto mínimo del tiempo de consulta sin restricción de reputación es de 11 ms mientras que el máximo es de 23 ms. Por otro lado, el punto mínimo del tiempo de consulta con restricción de reputación es de 19 ms mientras que el punto máximo es de 35 ms. En base a esta información se puede deducir que el tiempo de consulta aumenta en promedio en un factor de 11 ms, lo cual sería el costo que afectaría el desempeño de la aplicación implementando el sistema de reputación.

5. Trabajo futuro

Como líneas futuras de trabajo, es pertinente incorporar una estrategia para lograr un aislamiento progresivo de los usuarios con baja reputación, mediante la agregación de sistemas de administración de Churn basados en conceptos de privacidad y confianza.

6. Conclusiones

En el presente artículo se presentó Varena-Trust, una aplicación implementada bajo el API de freePastry, la cual ayuda a resolver la problemática de privacidad en sistemas Peer-to- Peer. Se mostraron las dos fortalezas principales de la aplicación: la confianza transitiva y las consultas y modificaciones con restricción de reputación. También se mostró que el desempeño del sistema Peer-to-Peer es afectado al implementar una solución de este tipo.

7. Referencias

[1] Sepandar D. Kamvar, Mario T. Schlosser
Stanford University, Hector Garcia-Molina
The EigenTrust Algorithm for Reputation Management in P2P Networks.

[2] Sepandar D. Kamvar, Mario T. Schlosser
Stanford University, Hector Garcia-Molina
The EigenTrust Algorithm for Reputation Management in P2P Networks.

[3] Runfang Zhou and Kai Hwang, Gossip-based Reputation Aggregation for Unstructured Peer-to-Peer Networks

Autores:

Oswaldo Vargas H.
o.vargas45@uniandes.edu.co

Sebastián Arenas V.
s.arenas62@uniandes.edu.co

Plan Amigos 2013 A

El plan Amigos es un programa por medio del cual la CUN pretende ampliar su comunidad Estudiantil, permitiendo así a la institución cumplir con los planes de cobertura establecidos por el gobierno en Plan Decenal de Educación. El objetivo es incentivar a la población actual, estudiantes, administrativos, docentes y egresados, para lograr el objetivo de ampliar la cobertura a través de nuevas vinculaciones en las ciudades y municipios donde hacemos presencia.

A quien va Dirigido?

Todos los que conforman la comunidad Cunista a nivel nacional:

Estudiantes
Personal Administrativo
Docentes
Egresados

¿QUIÉN ES ESTUDIANTE REFERIDO EFECTIVO?
Será toda aquella persona que se matricule por primera vez en la CUN, bien sea para primer semestre o a través de homologación externa. Para que este referido sea efectivo es necesario que el mismo se encuentre registrado debidamente en la página web a través del siguiente link. http://www.cun.edu.co/referidos.html

Ingeniería de sistemas