Un mécanisme de révocation orienté services pour les réseaux P2P

Thibault Cholez, Isabelle Chrisment, Olivier Festor

Un mécanisme de révocation orienté services pour les réseaux P2P

Thibault Cholez, Isabelle Chrisment and Olivier Festor
MADYNES - INRIA Nancy-Grand Est, France
{thibault.cholez, isabelle.chrisment, olivier.festor}@loria.fr

***Résumé*—Avec le déploiement sans cesse croissant des réseaux pair-à-pair, superviser les comportements malveillants des utilisateurs qui dégradent la qualité et les performances de ces réseaux devient un enjeu majeur. Dans ce papier, nous proposons un mécanisme de révocation complètement distribué et adaptatif basé sur la réputation de chaque pair accessible grâce à des comptes distants stockés au sein de la DHT. L'originalité de notre approche est d'intégrer la révocation au cœur des services proposés par le protocole P2P ce qui permet d'éviter les mécanismes complexes de consensus et de chiffrement passant difficilement à l'échelle. Le premier critère utilisé pour construire la réputation d'un pair est sa contribution au réseau afin de combattre les comportements égoïstes. Les délais induits par le mécanisme sont transparents pour l'utilisateur. Le mécanisme est robuste aux attaques si l'identité des pairs est maitrisée.**

***Index Terms*—réseaux P2P, mécanisme de révocation, mécanisme de réputation, comptes distribués, KAD**

I. Introduction

Les réseaux pair-à-pair (P2P) ont prouvé leur capacité à rassembler et partager de grandes quantités de ressources grâce à la collaboration individuelle de nombreux pairs. Ils proposent de nombreux avantages par rapport au schéma client-serveur classique : ils passent mieux à l'échelle, tolèrent les pannes et distribuent le coût de l'infrastructure.

Cependant, les réseaux P2P rencontrent plusieurs difficultés dues au nombre important d'utilisateurs malveillants qui dégradent la qualité de service. L'absence d'autorité centrale et le comportement individuel de chaque pair rendent leur gestion difficile. Les pairs malveillants peuvent exercer trois sortes d'actions néfastes : l'attaque ou le non respect du protocole P2P, la diffusion de contenu néfaste (malware, pollution, contenu illégal) et les comportements égoïstes. Plusieurs études ont été menées afin d'estimer l'impact de ces comportements. Ainsi dans Gnutella [1], 70% des utilisateurs ne partagent rien et 50% des ressources sont partagées par seulement 1% des utilisateurs. Le phénomène de pollution a été étudié dans Kazaa [2] où en moyenne 50% des fichiers audio partagés sont pollués et même davantage pour les fichiers récents. Les auteurs mettent ainsi en évidence les limites d'un système basé sur le volontarisme dans un réseau où chacun est anonyme et sont pessimistes quant à l'avenir du réseau si ces phénomènes ne sont pas résolus.

Dans ce contexte, les réseaux P2P ont besoin de contrôler les comportements de leurs utilisateurs. Pour cela, nous avons conçu un mécanisme de révocation distribué et adaptatif. Ce résumé est organisé comme suit. La section II présente d'autres travaux concernant la réputation et la révocation dans les réseaux P2P. Le fonctionnement de notre architecture est décrit dans la section III en prenant pour exemple le réseau KAD. La section IV concerne l'évaluation des performances et les questions de sécurité avant de conclure en section V.

II. Travaux relatifs

A. Réputation

Gérer la réputation dans un environnement distribué est un réel défi. La majorité des systèmes de réputation existants n'ont qu'un point de vue local : chaque pair stocke la réputation de ceux avec lesquels il est entré en relation [3]. Un tel système présente plusieurs inconvénients : il est impossible de savoir si un pair est malveillant avant d'être entré en contact avec lui car les avis ne sont pas partagés entre les pairs et une connaissance locale est insuffisante pour détecter les mauvais comportements (trop peu de pairs connus, trop peu de transactions avec chacun). C'est pourquoi ces systèmes sont peu efficaces pour lutter contre les utilisateurs égoïstes.

B. Révocation

Dans [4], les auteurs présentent et analysent différentes approches pour réaliser un contrôle d'accès sans autorité centrale. Les performances de plusieurs politiques possibles sont évaluées, impliquant différents seuils et mécanismes de chiffrement pour implanter le contrôle d'accès. Il apparaît qu'ils passent difficilement à l'échelle ce qui les rend plus adaptés à de petits réseaux avec d'importantes contraintes de sécurité.

Dans [5] est présentée une manière originale de réaliser une révocation dynamique dans un réseau P2P. Lorsqu'un pair détecte qu'un autre est malveillant, il envoie une note de révocation incluant son identité et celle du pair malveillant à l'ensemble du réseau, considérant sa propre existence comme moins importante que la bonne santé du réseau. Le coût du suicide limite les détournements possibles du mécanisme. Cet avantage est aussi sa principale limite car il ne peut être utilisé dans un réseau public où chaque pair a un intérêt individuel.

III. Architecture

A. Comptes distribués

Dans notre système, nous reprenons le principe des comptes distribués présenté dans PeerMint [6] car il s'avère être un moyen pertinent pour introduire la réputation dans un réseau P2P. Le stockage d'un compte se fait en le distribuant sur plusieurs pairs grâce à une DHT (ex : Kademlia). Pour cela, chaque compte d'utilisateur a une adresse logique (userID) de 128 bits qui doit persister entre chaque connexion en plus de l'adresse du pair lui-même (KadID). Ce groupe de pairs est périodiquement mis à jour afin de garder l'information au sein du réseau malgré le *churn* ; de plus, la réplication rend le mécanisme robuste. Chaque pair dispose donc d'un compte public (i.e. un ensemble d'informations) stocké sur le réseau. La réputation d'un pair est ainsi accessible sur son compte et évolue avec les retours des autres membres, de manière à ce que chaque connaissance soit utile à la communauté.

Services révoqués	Partage	Sécurité
bootstrap	Non	Oui
publication et envoi	Non	Oui
téléchargement	Oui	Oui
routage et recherche	Non	Non

TAB. I
SERVICES RÉVOQUÉS EN FONCTION DU CRITÈRE DE RÉPUTATION

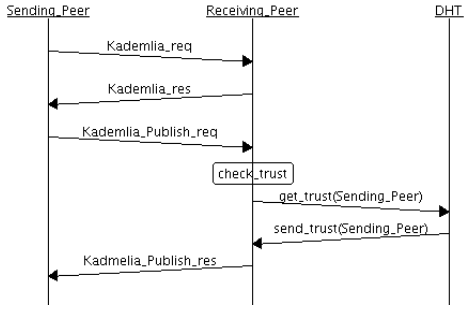

FIG. 1. Vérification de la réputation pendant le processus de publication

B. Évolution de la réputation

Pour illustrer comment les comptes distribués peuvent être utilisés pour faire évoluer la réputation d'un utilisateur, nous avons décrit un mécanisme de réputation basé sur la contribution d'un pair aux ressources du réseau. L'idée est d'obliger les utilisateurs à proposer des données intéressant la communauté grâce au mécanisme de révocation, leur réputation augmentant par leur contribution au réseau et inversement diminuant avec leur consommation. Ce mécanisme est décrit dans le papier [7].

C. Le mécanisme de révocation

Le mécanisme de révocation utilise la réputation affichée sur le compte de chaque pair pour décider s'il doit être révoqué d'après un seuil critique. Comme les réseaux P2P reposent sur les services mutuels fournis entre pairs, un moyen de révoquer de manière distribuée consiste à vérifier la réputation d'un pair avant de lui rendre service. Ainsi, si chaque pair du réseau agit de cette manière, un pair présentant une mauvaise réputation sera automatiquement révoqué, ses requêtes étant rejetées par le réseau. Par ailleurs, ce mécanisme est adaptatif car les services refusés peuvent changer en fonction du critère de réputation utilisé (tab I).

Nous avons appliqué la révocation de services sur le réseau P2P KAD, implanté par l'application eMule. Une fois connecté au réseau, un pair demande différents services en envoyant des requêtes aux autres pairs. Dans KAD, ceci est réalisé en deux étapes. Dans un premier temps des requêtes génériques *Kademlia_REQ* sont envoyées pour trouver les nœuds potentiellement capables de fournir le service (selon leur place dans la DHT). Ensuite, quand un pair est finalement trouvé, une autre requête spécifique au service demandé est alors envoyée (publication, recherche, transfert). Vérifier la réputation juste avant cette seconde étape permet de distinguer les services à révoquer. La figure 1 présente le déroulement d'une demande de publication en y incluant la vérification de la réputation. Les autres services suivent le même principe.

IV. ANALYSE ET DISCUSSION

A. Évaluation des performances

Nous avons évalué le temps nécessaire pour consulter la réputation d'un pair dans le réseau. Nous avons pour cela déployé sur EmanicsLab un client modifié implantant le mécanisme de révocation. Le nombre de réponses retournées et les délais varient en fonction du facteur de réplication du compte. En moyenne 2/3 des comptes répliqués sont trouvés avant l'expiration de la recherche. Celle-ci expire après un timeout (300 sec) mais plus généralement après avoir contacté 50 pairs dans le réseau (170 sec). Dans tous les cas, ce délai n'est pas ressenti par les utilisateurs car les services surveillés ne sont pas temps-réel.

B. Analyse de la sécurité

La sécurité des mécanismes fut une contrainte majeure lors de leur conception. Nous avons anticipé les différents comportements malveillants possibles de chacun des acteurs afin de rendre les mécanismes plus sûrs [7]. Le principal problème encore à l'étude vient de l'identité des pairs qui d'ordinaire n'est pas contrainte dans les grands réseaux P2P publics. Si l'identité n'est pas maîtrisée, un pair malveillant peut revenir dans le réseau avec une nouvelle identité après avoir été révoqué (whitewhash). Le mécanisme est également vulnérable à une Sybil attaque [8] car un pair malveillant peut alors prendre le contrôle de son propre compte ou de celui d'un autre, maîtrisant ainsi le mécanisme de révocation.

V. CONCLUSION

Superviser les comportements des utilisateurs malveillants est nécessaire pour le bon développement des réseaux P2P publics. Afin de répondre à ce problème, nous proposons un mécanisme de révocation basé sur les comptes distribués et le contrôle des services, qui est complètement distribué, adaptatif, et qui ne nécessite pas de consensus ni de moyens de chiffrement complexes.

Les travaux futurs consistent à limiter le problème de l'identité des pairs pouvant conduire à des attaques. Nous développerons ensuite le mécanisme de réputation afin de prendre en considération d'autres critères comme la qualité du contenu partagé pour lutter contre la pollution et la diffusion de malware.

RÉFÉRENCES

[1] Daniel Hughes, Geoff Coulson, and James Walkerdine. Free riding on gnutella revisited : The bell tolls ? *IEEE Distributed Systems Online*, 6(6) :1, 2005.

[2] J. Liang, R. Kumar, Y. Xi, and K. Ross. Pollution in peer-to-peer file sharing systems. In *IEEE Infocom*, pages 1174–1185, march 2005.

[3] Elizabeth Chang Farookh Khadeer Hussain and Omar Khadeer Hussain. State of the art review of the existing bayesian-network based approaches to trust and reputation computation. In *ICIMP 2007 : The 2d International Conference on Internet Monitoring and Protection*, July 2007.

[4] Nitesh Saxena, Gene Tsudik, and Jeong H. Yi. Admission control in peer-to-peer : design and performance evaluation. In *SASN '03 : Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks*, pages 104–113, New York, NY, USA, 2003. ACM Press.

[5] Jolyon Clulow and Tyler Moore. Suicide for the common good : a new strategy for credential revocation in self-organizing systems. *SIGOPS Oper. Syst. Rev.*, 40(3) :18–21, 2006.

[6] David Hausheer and Burkhard Stiller. Peermint : Decentralized and secure accounting for peer-to-peer applications. In *NETWORKING 2005 : 4th International Networking Conference*, pages 40–52, May 2005.

[7] Thibault Cholez, Isabelle Chrisment, and Olivier Festor. A distributed and adaptive revocation mechanism for p2p networks. In *ICN 2008 : The Seventh International Conference on Networking*, April 2008.

[8] Moritz Steiner, Taoufik En Najjary, and Ernst W Biersack. Exploiting KAD : possible uses and misuses. *Computer communications review, Volume 37 N5, October 2007*, 2007.