On Tracking Freeriders in Gossip Protocols

Rachid Guerraoui
EPFL
Lausanne, Switzerland

Kévin Huguenin
IRISA
Rennes, France

Anne-Marie Kermarrec
INRIA
Rennes, France

Maxime Monod
EPFL
Lausanne, Switzerland

Abstract

Peer-to-peer content dissemination applications suffer immensely from freeriders, i.e., nodes that do not provide their fair share. The Tit-for-Tat (TfT) incentives have received much attention as they help make such systems more robust against freeriding. However, these rely on an asymmetric component, namely opportunistic pushes, that let peers receive content without sending anything in return. Opportunistic push constitutes the Achilles' heel of TfT-based protocols as illustrated by the fact that all known attacks against them exploit it. This problem becomes even more serious when used by colluding freeriders.

In this paper, we discuss the possibility of using accountability to secure gossip-based dissemination protocols based on asymmetric exchanges. The fact that gossip protocols are dynamic and randomized makes our approach robust against collusion and alleviates the need for cryptography. We present the challenges raised by an auditing approach and give insights into how to build a freerider-tracking protocol for gossip-based content dissemination.

1. Introduction

High-bandwidth content dissemination applications are becoming increasingly popular over the Internet. For obvious scalability reasons, the tendency has been shifting toward the use of the peer-to-peer paradigm with its efficient distributed protocols. These include multi tree-based protocols such as [3], mesh-based protocols such as [4], and gossip-based protocols such as [5,6,12]. Such dissemination protocols highly rely on the willingness of peers to collaborate, i.e., to devote a fraction of their resources – specifically their upload bandwidth – to the system.

Not surprisingly, such systems are vulnerable to *freeriders*, i.e., peers that do not contribute their fair share [10]. Freeriding is a common behavior in large-scale systems deployed in the public domain [1]. The impact of freeriders is even higher when they collude, i.e., when they collaborate to favor both their individual and common interests.

Two orthogonal approaches have been considered to cope with freeriding: *incentive* and *coercive* mechanisms. While the first incites peers to share their resources by making their profits proportional to their contribution, the latter punishes peers that do not collaborate. The Tit-for-Tat (TfT) incentive mechanism, where content is exchanged only between peers with mutual interest, is a simple and efficient way to discourage freeriders in a system where dissemination is symmetric, i.e., the exchanges between pairs of peers are balanced and bidirectional [4, 12]. Yet, this requires a complementary optimistic component, i.e., nodes altruistically pushing content to other nodes without requiring anything in return. The optimistic component, namely the asymmetric communication, is vulnerable to specific attacks as we will discuss later. On the other hand, accountability, i.e., logging peers' actions for further verifications, provides a powerful coercive mechanism. However, none of the proposed solutions [12] is applicable to randomized algorithms such as epidemic protocols without using asymmetric cryptography (which is both expensive and non-scalable).

We believe that a verification mechanism, inspired by accountability and acting as a coercive mechanism, can *(i)* lead to a fair system on its own and *(ii)* be a relevant complement to collaboration incentives based on bidirectional exchanges such as TfT. In fact, the fear of being detected can itself act as an incentive not to misbehave: consider a non-zero probability of being detected when misbehaving and that detection leads to some punishment (e.g., a penalty or exclusion). In this case, nodes might want to cooperate by providing their fair share. However, designing a verification mechanism to detect possibly colluding freeriders in a content dissemination system based on random peer selection without cryptography raises the following challenges: *(i)* how to verify that a peer contributes its fair share when the set of peers to which it is supposed to upload data may not be predictable (randomness of peer selection)? ; *(ii)* how to use untrusted logs (no cryptography) in the presence of colluding peers that can mutually cover each other up? ; *(iii)* how to distinguish between a message lost by the network and a message dropped on purpose by a freerider?

978-1-4244-5067-1/09/$26.00 ©2009 IEEE

This paper proposes a new fully decentralized approach to address these issues. In addition to securing the wide class of epidemic protocols, our approach can be used as a complement to existing solutions designed for dissemination protocols based on bidirectional exchanges. Currently, such protocols, e.g., BitTorrent [4] or BAR Gossip [12], include an opportunistic unchoking mechanism, necessary in TfT-based dissemination protocols, which ends up being their Achilles' heel in the presence of freeriders [13,14], as nodes can be served without contributing. We present our approach in the context of a generic gossip-based protocol, where data is disseminated in a random manner following an *asymmetric* push scheme, as opposed to *symmetric* exchanges where a balance between the interests of a pair of nodes can be achieved by means of TfT. Moreover, gossip-based protocols are increasingly popular and have been successfully applied to streaming [6] and file sharing [5], two of the most attractive Internet applications.

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 clearly positions our approach with respect to previous work and motivates our work. Section 4 describes the general context and exposes the proposed approach. Section 5 concludes.

2. Related Work

The novel approach proposed in this paper aims at preventing freeriding in epidemic content dissemination protocols via coercion. It is related to the following recent work.

Incentives have been broadly used to deal with freeriders in file sharing systems such as BitTorrent [4] by making the benefits of the peers proportional to their contribution to the system. However, there exists several attacks to the popular Tit-for-Tat incentive which allows freeriders to download content without any contribution to the system [13,14], mainly by taking advantage of peers' generosity, engendered by the opportunistic unchoking mechanism.

PeerReview [7] deals with malicious nodes via an accountability approach. Peers maintain verifiable signed logs of their actions that can be audited and checked using a simulator of the application, i.e., a reference implementation, run at each peer (in addition to the application). While powerful and highly generic, PeerReview is not applicable to non-deterministic protocols whose execution relies on randomized algorithms. Recent work proposed CSAR [2], based on verifiable random functions that allow accountability of randomized algorithms, but it is implemented by means of asymmetric cryptography primitives.

Indeed, the inherent randomness of epidemic protocols makes it difficult to make these protocols robust. Several solutions have been proposed, mainly in the context of streaming applications. Li *et al*. proposed BAR Gossip [12] and its improvement FlightPath [11], a gossip-based streaming application resilient to rational nodes, i.e., nodes that contribute only if they perceive a benefit in return. Randomness is handled as in CSAR. BAR Gossip is composed of opportunistic pushes performed by altruistic nodes (essential for the efficiency of the protocol) and balanced pairwise exchanges based on a TfT mechanism made robust using cryptographic primitives. Therefore, it cannot be used to make protocols based only on optimistic pushes, such as the ones considered in this paper, robust against freeriders.

Name	Exchanges	Overlay	Crypto.
PeerReview [7]	any	any	yes
Bar Gossip [12]	symmetric	dynamic	yes
Distributed auditing [8]	asymmetric	static	no

Table 1. Summary of related work

The two approaches that are most closely related to ours are the distributed auditing protocol proposed by Haridasan *et al*. in [8] and the passive monitoring protocol proposed by Karakya *et al*. in [9]. The first protocol targets live streaming applications. Freeriders are detected by cross-checking their neighbors reports. The latter focuses on gossip-based search in the Gnutella network. The peers monitor the way their neighbors forward/answer queries in order to detect freeriders and peers who drop queries. Yet, contrarily to gossip-based content dissemination – which is based on random peer selection – in both protocols, the peers' neighborhoods are static, i.e., they form a fixed mesh overlay. Therefore, the situation where colluding peers mutually cover up for each other (not addressed by the authors) makes such monitoring protocols unusable.

Table 1 summarizes the related work and highlights its main characteristic.

3. Motivation

Since its successful application in BitTorrent the Tit-for-Tat (TfT) incentives have become a *de facto* standard for dealing with freeriders in large scale content distribution systems. BAR Gossip – and by extension FlightPath – is the most successful and complete application including TfT to ensure fair collaboration of participants in the context of gossip protocols. However, it suffers from several issues that drastically limit both its efficiency and its practicality in a large-scale content dissemination system.

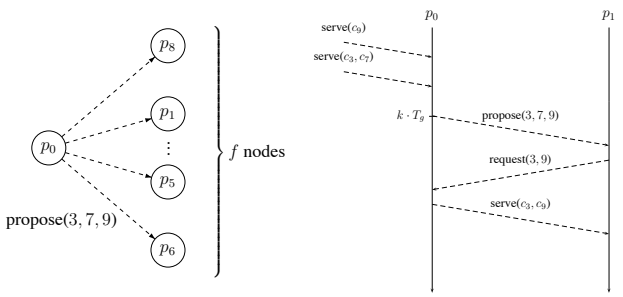
First, BAR Gossip makes intensive use of both symmetric and asymmetric cryptography. Beyond the fact that this adds a non negligible overhead to the protocol, both in terms of message exchanges and computation, it requires a trusted third party to issue identification certificates, namely a public key infrastructure. In addition to requiring prior registration, asymmetric cryptography techniques generate a high load, proportional to the size of the network, on the cen-

tralized server. Further, it is shown in [8] that BAR Gossip collapses when the system grows beyond a given size.

Second, similarly to TfT-based protocols, BAR Gossip relies on altruistic nodes and opportunistic pushes where nodes upload pieces of data without receiving anything in exchange. This component is essential for TfT to work in a real life setting mainly to ensure that nodes joining the system can gain bargaining power (i.e., pieces to exchange) to initiate symmetric exchanges but also to ensure that nodes with low upload capacities are unchoked (which is unlikely in bidirectional exchanges, as nodes with high upload bandwidth are preferred). In general, this component is used without any protection against freeriders despite the fact that most attacks to TfT actually exploit it. This shows that protocols using TfT are still sensitive to freeriders and that the problem of designing incentives for the wide class of epidemic dissemination protocols (to which the opportunistic pushes belong) is of the utmost importance in fighting against freeriders. In BAR Gossip this issue is addressed by forcing the peers that are opportunistically unchoked to send the exact same amount of data they received, should they send junk data if they have nothing of interest. This results in a waste of bandwidth. Although it is acceptable in BAR Gossip since nodes can often initiate balanced exchanges, the proposed solution cannot be applied in a system based exclusively on asymmetric exchanges.

Third, protocols such as BAR Gossip are extremely vulnerable to the presence of colluding nodes. The first reason for this is that the game theory at the core of BAR Gossip does not handle teams of players, but an even larger problem is that the opportunistic component is alone sufficient for achieving very good performance when exploited by a coalition. Assuming a set of colluding freeriders (potentially hosted on a high speed network or even on the same machine) that take advantage of the opportunistic component to obtain pieces for free (or against a small amount of junk data) and then share these pieces among the coalition, the stream can be downloaded with a small contribution in return. This situation where nodes collude is not captured by the BAR model which considers rational nodes to be nodes willing to contribute as long as this generates profit in return without attention to the nodes with which they collaborate.

These reasons motivate the need for a lightweight tracking system for gossip-based content dissemination protocols. Accountability appears as a promising solution in this context. Yet, introducing accountability in gossip protocols is very challenging for several reasons: such protocols are inherently random, dynamic and rely on asymmetric exchanges that prevent the use of game theory. These intrinsic properties of gossip raise the following challenges: *(i)* to deal with randomness in partner selection, *(ii)* to rely on untrusted logs (no cryptography) in the presence of colluding nodes that can mutually cover each other up, and *(iii)* to distinguish between a message lost by the network and a message dropped on purpose by a freerider.



(a) Gossip protocol (propose phase) (b) Three phases for a given proposal

Figure 1. Three-phase gossip protocol

From the previous discussion, we conclude that the underlying dynamic overlay, randomness, collusion, and trust are the main issues faced when dealing with freeriders in gossip-based systems. Interestingly enough, the dynamic overlay and the randomness of gossip can help in addressing the other two issues. In static mesh networks, nodes pick a set of neighbors and keep them for the duration of the system lifetime or until a problem arises, in which case the node might add nodes to its neighborhood or replace the malfunctioning ones by possibly bootstrapping again. Therefore, as explained above, colluding nodes can build a sub-network within the network as a whole and exploit honest nodes. Such an attack is difficult to detect as colluding nodes would cover each other up, making any verification system useless. However, this attack cannot be performed in a system where nodes pick their neighbors at random and renew their neighborhood at a high frequency. Based on this line of reasoning, we propose an approach based on distributed verification coupled with a statistical auditing protocol ensuring that the underlying dynamic overlay and randomness of the gossip protocol are respected.

4. Lightweight freerider-tracking in gossip

In this section, we briefly present the illustrative and widely known example of the three-phase gossip protocol for content dissemination and list the freeriding opportunities in such a protocol. We then explore the design space of distributed freerider-tracking protocols based on the rationale presented in the previous section.

We consider a system where some content, divided into chunks, is broadcast from a source to all nodes using a gossip-based protocol. In short, each node periodically proposes the set of chunks it received during the last gossip period to a fixed-size set of f random nodes. Upon receipt of a proposal, a node tells the sender which chunk it needs and

the sender then pushes the requested chunks. These three phases are illustrated in Figures 1(a)-1(b).

Looking at each of the three phases, it can be seen that a freerider can decrease its contribution in only four ways: *(i)* decreasing its gossip period in order to increase the probability that the nodes to which it proposes the chunks already have them; *(ii)* decreasing the number of nodes to which it sends proposals; *(iii)* not sending the chunks it proposed; *(iv)* biasing the selection of partners. The aim of the last attack is twofold: first, it increases the benefit of the coalition by uploading content only to colluding nodes; and second, it decreases the chance of colluding freeriders being caught by covering each other up in case of verification.

It can be concluded from this list of attacks that a distributed verification mechanism must ensure that the nodes *propagate* the chunks they received to *at least f randomly chosen* nodes, *on time*. Note that this is an orthogonal approach to verification for symmetric exchange protocols which ensure that transfers are bidirectional. To ensure correct propagation of the chunks, the sender checks that the chunks which it sent are proposed in turn to other nodes. When a node p_1 receives chunks from a node p_0, p_1 must provide p_0 with the list of the f nodes to which it proposed these chunks. Node p_0 assesses the validity of the list by polling the f alleged recipients. In addition, the nodes that receive a proposal from p_1 and requested some chunks in return check whether p_1 indeed sends the requested chunks. Such a scheme is efficient only if the verifiers are honest. Based on the fact that not all nodes will cover up for a given node, verifying that a node interacts with a large subset of the network ensures the efficiency of the distributed verification. This comes down to assessing the randomness of a node's past collaborators. To this end, each node maintains a log of the nodes with which it collaborated in the last h seconds. The randomness (with respect to a uniform distribution) is assessed using statistical goodness-of-fit tests such as entropy-based tests, the χ^2 test or the Kolmogorov-Smirnov test. The validity of the log is assessed by polling all or a subset of the nodes that appear in the log. Note that statistical verification ensures that the contribution of a node is evenly spread in the network and not only in a small subset of colluding nodes.

Nodes' behavior is evaluated by the nodes themselves, and since they are changing at each gossip period, their evaluations must be compiled into a consistent score. When nodes detect a misbehavior, they emit blames containing a negative score proportional to the severity of the detected misbehavior. Two issues arise from this. First, a node might be wrongfully blamed due to message losses. In order to avoid false positives, nodes must thus dynamically adjust the scores of their peers so that the average score of honest nodes is zero on average. Second, the decision mechanism which determines if a node should be punished or not, based on its score, should only consider absolute scores, i.e., not comparing the scores of the nodes, otherwise freeriders would have interest in wrongfully blaming honest nodes. With absolute scores and a given threshold, freeriders have no interest in blaming honest nodes, as it would result in worse overall performance. This phenomenon is known as the *tragedy of the commons*.

5. Conclusion

In this paper we highlighted the limitations of Tit-for-Tat as an incentive for peer collaboration in the presence of freeriders, especially when they collude. We motivated a new approach for securing asymmetric exchanges (including optimistic pushes of TfT-based protocols) and proposed a mechanism based on gossip, alleviating the need for cryptography. Preliminary results are very encouraging, and we plan to pursue our work in the direction of a fully-fledged lightweight freerider-tracking system.

References

[1] E. Adar and B. Huberman. Free riding on Gnutella. *First Monday*, 5(10), October 2000.

[2] M. Backes, P. Druschel, A. Haeberlen, and D. Unruh. CSAR: A Practical and Provable Technique to Make Randomized Systems Accountable. In *NDSS*, 2009.

[3] M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh. SplitStream: High-bandwidth Multicast in Cooperative Environments. In *SOSP*, 2003.

[4] B. Cohen. Incentives Build Robustness in BitTorrent. In *P2P Econ*, 2003.

[5] M. Deshpande, B. Xing, I. Lazardis, B. Hore, N. Venkatasubramanian, and S. Mehrotra. CREW: A Gossip-based Flash-Dissemination System. In *ICDCS*, 2006.

[6] D. Frey, R. Guerraoui, A.-M. Kermarrec, M. Monod, and V. Quéma. Stretching Gossip with Live Streaming. In *DSN*, 2009.

[7] A. Haeberlen, P. Kouznetsov, and P. Druschel. PeerReview: Practical Accountability for Distributed Systems. In *SOSP*, 2007.

[8] M. Haridasan, I. Jansch-Porto, and R. Van Renesse. Enforcing Fairness in a Live-Streaming System. In *MMCN*, 2008.

[9] M. Karakaya, I. Körpeoğlu, and O. Ulusoy. Counteracting Free-riding in Peer-to-Peer Networks. *Computer Networks*, 52(3):675–694, 2008.

[10] R. Krishnan, M. Smith, Z. Tang, and R. Telang. The Impact of Free-Riding on Peer-to-Peer Networks. In *HICSS*, 2004.

[11] H. Li, A. Clement, M. Marchetti, M. Kapritsos, L. Robinson, L. Alvisi, and M. Dahlin. FlightPath: Obedience v.s. Choice in Cooperative Services. In *OSDI*, 2008.

[12] H. Li, A. Clement, E. Wong, J. Napper, I. Roy, L. Alvisi, and M. Dahlin. BAR Gossip. In *OSDI*, 2006.

[13] T. Locher, P. Moor, S. Schmid, and R. Wattenhofer. Free Riding in BitTorrent is Cheap. In *HotNets*, 2006.

[14] M. Sirivianos, J. Park, R. Chen, and X. Yang. Free-riding in BitTorrent with the Large View Exploit. In *IPTPS*, 2007.