Strange Bedfellows: Community Identification in BitTorrent

David Choffnes[‡], Jordi Duch[◇], Dean Malmgren[◇], Roger Guimerà[◇], Fabián Bustamante[‡], Luís A. Nunes Amaral[◇]
[‡]EECS, [◇]NICO/Chem. & Bio. Eng., Northwestern University[*]

Abstract

While P2P systems benefit from large numbers of interconnected nodes, each of these connections provides an opportunity for eavesdropping. Using only the connection patterns gathered from 10,000 BitTorrent (BT) users during a one-month period, we determine whether randomized connection patterns give rise to communities of users. Even though connections in BT require not only shared interest in content, but also concurrent sessions, we find that strong communities naturally form – users inside a typical community are 5 to 25 times more likely to connect to each other than with users outside. These strong communities enable guilt by association, where the behavior of an entire community of users can be inferred by monitoring one of its members. Our study shows that through a single observation point, an attacker trying to identify such communities can uncover 50% of the network within a distance of two hops. Finally, we propose and evaluate a practical solution that mitigates this threat.

1. Introduction

P2P has enabled a wide range of Internet applications ranging from large-scale data distribution to video streaming and telephony. While much of the strength of the P2P model lies in large numbers of interconnected nodes, their connections offer multiple opportunities for eavesdropping. *In this paper, we show that these connections erode privacy in a way that is ignored by most distributed systems and invisible to end users.*

This work focuses on the BitTorrent (BT) file-sharing network where peers connect on the basis of common and concurrent interest in the same content, rather than on friendship, common language or geographic proximity. Using connection patterns gathered during a one-month period (comprising a stable population of 10,000 BT users), we investigate the existence of communities – collections of peers significantly more likely to connect to each other than to a randomly selected peer. We show that strong communities form naturally in BT, with users inside a typical community being 5 to 25 times more likely to connect to each other than with outside users.

Historically, this ability to classify users has been abused by third parties in ways that violate individual privacy. The existence of strong communities enables guilt by association, where the behavior of an entire community of users can be inferred by monitoring one of its members. We demonstrate that, through a single observation point, an attacker trying to identify such communities can reveal 50% of the measured network using only knowledge about a peer's neighbors and their neighbors (i.e., peers within two hops of the attacker). Further, an attacker monitoring only 1% of the network can correctly assign users to their communities of interest more than 86% of the time. Finally, we show how to mitigate this threat by adding between 25 and 50% additional random connections.

The remainder of the paper is structured as follows. Section 2 describes how to identify communities of users based on BT connection information. We show that these communities of shared interest can be exploited in a guilt-by-association attack in Sec. 3. To mitigate this threat, Sec. 4 discusses an approach that weakens and disrupts community analysis by adding random connections. We cover related work in Sec. 5, and conclude in Sec. 6.

2. Communities in BitTorrent

In this section, we describe our dataset, which contains connection information for 10,288 peers during a one-month period. We use this information to form a graph and analyze whether there are distinct communities in which users connect to each other more often than to users outside the community.

2.1 Dataset

The data used in this study was collected from BT users during the 31 days of March, 2008. Our dataset contains the endpoints and durations of P2P connections from each monitored host. In BT, a connection between users means they share interest in some content; however, we do *not* collect any information that identifies the particular content.

We restrict our analysis to a stable set of peers during the measurement period. Specifically, we filter our dataset to contain data only for hosts that have appeared in our records before March 1, 2008 and after March 31, 2008 (based on data from July 15, 2007 to December 11, 2008). We are left with 10,288 users, with 3,029 online users per day and 10,162 connections between them.

From this dataset, we create graphs of the connections between peers. Namely, for each day of the month we generate a graph, where each node is a peer that is online during that day and each edge indicates that

[*] Bustamante and Choffnes supported in part by NSF CAREER Award CNS-0644062. Duch currently with CS & Math, Universitat Rovira i Virgili (URV). Guimerà currently with ICREA & Chem. Eng., URV.

there was at least one connection established between the corresponding peers during that day. To avoid issues with users that connect at regular intervals (e.g., every Saturday), we aggregate the information into four weekly graphs. These graphs consist of weighted edges where the weight w_{ij} between nodes i and j indicates how many days this pair establishes a connection.

2.2 Extracting Communities

In social networks, individuals decide with whom they want to establish connections, so communities naturally appear. These communities are usually a reflection of past or present geographical colocation, shared interests, or co-membership in organizations, and manifest themselves in the network as groups of nodes that are more densely connected to each other than we would expect by random chance [1,21].

In contrast, nodes in many P2P networks, including BT, establish connections according to a protocol that selects peers at random from a pool of eligible hosts. Intuitively, this randomness might disrupt opportunities for community structure in P2P networks. However, much as in social networks, the existence of a connection between two users in a P2P network is a reflection of shared interest – in BT, a connection indicates concurrent shared interest in content. We now show this shared interest is sufficient to form strong communities of users in the BT network.

A successful approach for solving the community identification problem is based on the maximization of a quality function $\mathcal{M}$, usually called modularity [15]. For a given partition $\mathcal{P}$ of a weighted graph into communities, the modularity is defined as

$$\mathcal{M}(\mathcal{P}) = \frac{1}{2L} \sum_{ij} \left[w_{ij} - \frac{s_i s_j}{2L} \right] \delta_{m_i m_j} \qquad (1)$$

where the sum is over all pairs of nodes, w_{ij} is the weight of edge (i, j), s_i is the sum of the weights of all of node i's edges, $L = \sum_i s_i$, m_i is the community to which node i belongs (in partition $\mathcal{P}$), and δ_{ij} is the Kronecker symbol ($\delta_{ab} = 1$ if $a = b$ and $\delta_{ab} = 0$ otherwise).

The modularity function is a relative measure of how much edge weight falls *within* communities, as opposed to *between* communities. If there were no communities in the network, the total connection strength s_i of each node would be evenly distributed among all the other nodes, so that the weights w_{ij} would be proportional to s_i and s_j. Positive modularities indicate systematic deviations from the perfectly homogeneous null model. Otherwise, modularity is nearly zero for a random partition of the nodes into communities, when all nodes are in the same community, or when each node is in a different community.

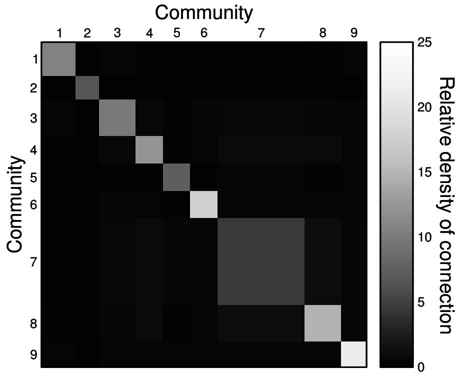


Figure 1: Density of connections within and between communities (relative to the average density of connections in the network) in a week-long network graph. Each row (column) corresponds to a community, and the height (width) indicates the size of the corresponding community. The density of connections within communities is 5 to 25 times higher than between communities.

The problem of optimizing modularity to detect communities in graphs is NP-hard, since the space of possible partitions of nodes into communities scales faster than any power of the system size [3]. We use the heuristic approach proposed by Duch et al. [9] to efficiently explore the space of possible partitions, as the technique provides a compromise between accuracy and speed [7]. We found its results were nearly identical to other randomized heuristic algorithms such as simulated annealing [10], and as we show in Sec. 4 it produces consistent communities [14] across multiple runs for the vast majority of nodes.

We use the extremal optimization method to investigate the community structure of the week-long graph that spans from March 22-28. Fig. 1 shows the density of peer connections that are inside and outside of their communities. The density within communities is 5 to 25 times higher than between communities. Although suggestive, these values do not necessarily mean that the communities we identify are statistically significant [12]. To address this issue, we compare our empirical results with an ensemble of randomized networks in which users connect with a uniform probability to each other (preserving the number of connections for each user). We find that the maximum modularity of the real graph is $\mathcal{M} = 0.439$, whereas the average maximum modularity of the randomized networks is $\mathcal{M} = 0.168$ with a standard deviation of 0.0012. With the real modularity more than 250 standard deviations larger than the random expectation ($z > 250$), we can safely conclude that the discovered communities are significant. For comparison, the modular structure of the worldwide air transportation network has $z \approx 430$, whereas the modularity of the Internet at the autonomous system level has $z \approx 80$ [11,13].

3. Community-Based Privacy Attacks

The BT network is already under privacy-intrusive attacks that entail using trackers and participating (rogue) clients to identify users that share particular content (e.g., to detect violation of copyrights) [19]. These attacks are limited by a number of factors, such as the number of trackers and torrents that must be monitored and rogue clients that must be run. In this section, we describe an attack that eliminates many of these restrictions by exploiting the BT community structure.

As we demonstrated in previous sections, nodes in the BT network form well-defined communities of shared interest. Given this, an attacker who identifies the content that a BT user is sharing can determine that all users in the same community are doing the same *without monitoring them directly*. We refer to this as a *guilt by association* attack – as first proposed by Cortes et al. [5] for identifying fraudulent callers in a phone network. We will show how this enables a small number of attackers to classify large numbers of peers.

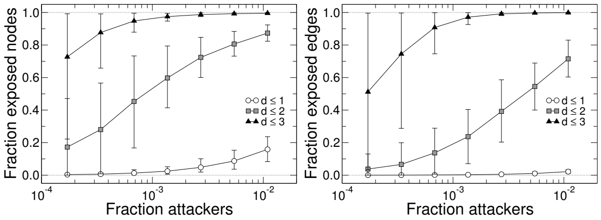
To realize this attack, we assume a threat model that comprises two phases. First, the attacker discovers as many connections as possible, then uses this information to identify communities of users that share interests. Because monitoring every P2P user is intractable, a viable alternative is to use a local discovery method to uncover the structure and the patterns of connections between users. For example, an attacker can deploy a number of participating clients to infiltrate the P2P network or sniff packets from a number of monitored hosts to observe connections.

In the second phase, the attacker extracts communities of shared interest for guilt by association. This can be accomplished using the same heuristic for finding modularity in the network discussed in Sec. 2.2. Because their analysis may be based on incomplete information, we study the effectiveness of classification in this context.

3.1 Discovering user connections

There are several methods that attackers might use to monitor the downloading activity of BT users. For instance, they can monitor information collected by trackers, acquire sets of peers connected to their neighbors via the Peer Exchange (PEX) protocol [6] or crawl the BT DHT for lists of peers connected to a torrent. In the case of monitoring trackers, an attacker could essentially reveal the entire network of connections, making it trivial to determine the community structure of users. However, recent developments, such as the popular tracker site The Pirate Bay moving entirely to DHT-based trackers [20], impact the effectiveness of this approach. To determine the limits of the guilt-by-association attack strategy, we analyze a worst-case scenario for attackers, where an incomplete view of the connectivity patterns in BT is revealed.

To this end, we model an attacker that crawls the BT network to obtain connectivity information. In particular, an attacker implements a breadth-first search (BFS) approach to find all users within a distance d of a rogue client, as acquired through the PEX protocol. By using multiple rogue clients, the attacker should be able to increase the coverage of peer connections.



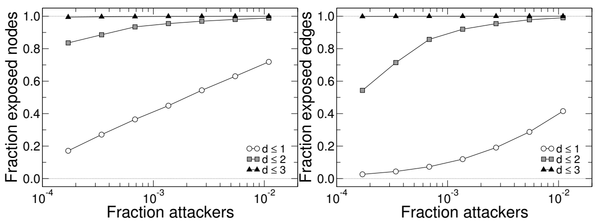
(a) Exposed nodes. (b) Exposed edges.

Figure 2: Fraction of exposed nodes and edges when a set of $N = 1, 2, 4, 8, 16, 32, 64$ attackers can monitor all nodes and edges within a distance d during one week. Attacking nodes selected uniformly at random. Symbols denote average over 100 Monte Carlo realizations and whiskers denote 95% confidence intervals.

To demonstrate the effectiveness of such an attack strategy, we select $N = 1, 2, 4, 8, 16, 32, 64$ nodes uniformly at random from each of the weekly BT networks and determine the fraction of nodes that they could collectively monitor within a distance d of all attackers. We repeat this Monte Carlo sampling 100 times to obtain reliable estimates of how much information a small set of attacking nodes could gather with such an approach. Figure 2(a) shows the fraction of exposed nodes for a different number of attackers within a monitoring distance d. In this case, a single attack node observes, on average, over 70% of all nodes within a distance $d \leq 3$ and a coordinated attack mounted by a small 1% of nodes observes, on average, over 80% of all nodes within a distance $d \leq 2$.

Of course, to optimize their effectiveness, attackers exploiting a BFS strategy would try to connect to as many users as possible to be able to monitor, first-hand, as much of the network as possible. We demonstrate the effectiveness of highly connected attackers by selecting the $N = 1, 2, 4, 8, 16, 32, 64$ most connected nodes from each of the weekly BT networks to determine the fraction of nodes that they could collectively monitor within a distance d of all of the attackers. Figure 3(a) shows the fraction of exposed nodes for a different number of attackers within a monitoring distance d, in this case for highly connected attackers. Here, a single attack node can observe over 80% of the network within a distance $d \leq 2$ and almost all of the network within a distance $d \leq 3$. As the figure shows, monitoring coverage grows with the fraction of attack nodes.

As these strategies illustrate, an attacker can reveal a large portion of the BT network's connectivity patterns without centralized information. Now we examine how



(a) Exposed nodes.

(b) Exposed edges.

Figure 3: Fraction of exposed nodes and edges when a small set of $N = 1, 2, 4, 8, 16, 32, 64$ attackers can monitor all nodes and edges within a distance d during one week. Attacking nodes are the set of N nodes with the largest degree.

		d		
		1	2	3
	0.0001	0.131	0.485	0.859
f	0.001	0.214	0.703	0.855
	0.01	0.343	0.864	0.902

Table 1: Similarity between the community structure of the real network and a partial reconstruction of the network discovered using a fraction f of attackers that observe to distance d (using $\tau = 8$ and $R = 10$).

this incomplete information can be used to determine community structures.

3.2 Detecting community structure

In the next step of the guilt-by-association attack, the attackers attempt to identify communities of shared interests. With access to information about the whole network, the community detection algorithm described in Sec. 2.2 accurately identifies communities. We now address how accurate community detection can be when using incomplete information. Specifically, we measure the probability p that two nodes are classified in the same community in the full network given that they are classified in the same community in the partial network.

First, we address how to confidently assign users i and j to the same community given that our community detection algorithm is nondeterministic. Our approach is to run extremal optimization R times to obtain high-modularity partitions $\{P_1, P_2, \ldots, P_R\}$. We then assume that users i and j can be confidently associated with each other if they are assigned to the same community at least τ times. We choose $R = 10$ and explore two different thresholds, $\tau = 5$ and $\tau = 8$.

Based on the analysis in Sec. 3, Table 1 shows p calculated for different values of the fraction f of monitoring nodes and the distance d for exposed edges in the network from March 22–28. For $\tau = 8$, we find that if 0.01% of the nodes in the graph are attackers, they can correctly classify users into communities more than 85% of the time for $d \leq 3$. If 1% of the nodes are attackers, they can achieve the same accuracy by only monitoring users that are within a distance $d \leq 2$. We find similar results for attackers that use a more relaxed threshold for assigning users to the same community ($\tau = 5$): $p = 0.819$ for $f = 0.0001$ and $d \leq 3$, and $p = 0.805$ for $f = 0.01$ and $d \leq 2$.

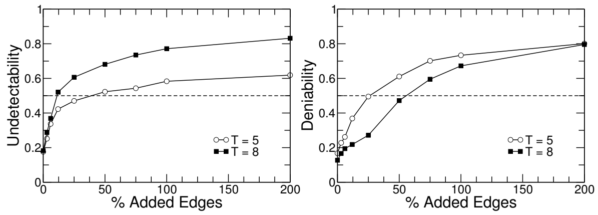
4. Hiding in the Crowd

The previous section showed that a small fraction of attackers can accurately infer communities. The success of this attack strongly depends on the assumption that attackers can reliably infer user interests based on the connections that they have with other peers. We posit that the best defense against this attack is simply to introduce noise such that this assumption no longer holds. Specifically, our approach is to add random edges to disrupt an attacker's ability to (*i*) correctly infer real connections and thus (*ii*) infer community membership. These edges come from connections to swarms for randomly selected torrents (e.g., free software and media).

To determine the effectiveness of our defense strategy, we simulate adding a varying number of random edges between nodes. Since we expect that exceptionally active users will have more incentive to hide their connectivity patterns than infrequent users, we add random edges proportional to the number of edges of each user. We now quantify the effectiveness of this method.

First, we measure the *undetectability* of users: the probability that any two users are not detected in the same community after adding random edges, given they are in the same community before adding random edges. That is, if an attacker found two users i and j classified in the same community without adding random edges, undetectability quantifies the likelihood that an attacker would correctly identify i and j in the same community after adding random edges. Fig. 4(a) shows how random edges increases undetectability. We again present the results for two different thresholds for community detection, a more restrictive one with $\tau = 8$, and a less restrictive $\tau = 5$. For $\tau = 8$ and only 10% additional random edges, an attacker would incorrectly infer that two users are in the same community more than 50% of the time. For $\tau = 5$, the same result is achieved with as few as 50% random edges.

Second, we measure the *deniability* for users, i.e., the probability that any two users are not detected in the same community before adding edges, given that they are in the same community after adding random edges. If an attacker found two users i and j in the same community after adding random edges, deniability quantifies the likelihood that an attacker would incorrectly determine that i and j in the same real community. Fig. 4(b) shows how deniability increases for the same two thresholds $\tau = 5$ and $\tau = 8$ as more random edges are added. For both thresholds, adding 50% additional random edges increases the deniability to 50%, meaning that attackers'



(a) Undetectability.

(b) Deniability.

Figure 4: Plots showing effectiveness of the attack mitigation strategy. Undetectability (left) measures how well our approach disrupts community identification while deniability (right) measures the inaccuracy of communities that are identified.

classifications are wrong the majority of the time.

We also find that the modularity of the network formed when adding two random links for each real link ($\mathcal{M} = 0.17$) is statistically indistinguishable from that of a completely randomized network. Thus, while there may exist an algorithm that can extract communities despite our proposed countermeasure, our approach is robust to any technique relying on *modularity*.

These results demonstrate that by adding only a few random edges (as few as $10 - 20\%$) we substantially increase the privacy of the user by making it difficult to correctly associate users that share the same type of content and by reducing the credibility of guilt-by-association attacks. We have implemented and deployed software that implements this approach for the Vuze BT client, which has been downloaded thousands of times and is described in the associated technical report [4].

5. Related Work

There is a large body of previous work that addresses privacy in distributed systems. For example, encryption hides plaintext data in connections; however, Saponas et al. [18] demonstrated several classes of devices and attacks that allow an attacker to obtain information about users. Another privacy layer is anonymization, which entails disassociating user-identifiable information from network flows [2, 8]. Unlike these solutions, which hide senders and receivers, our approach hides a user's communities of interest, which does not itself require hiding endpoints. An alternative to anonymity in open networks is trusted identities in private (i.e., closed) networks [16]. Such "networks of friends" are susceptible to community-based classification as described in this paper (i.e., the network is the community) and the guilt-by-association attack.

Our work is inspired by projects that share its spirit but are applied in different contexts. Cortes et al. [5] describe an approach for identifying communities of interest to identify fraudulent callers in a phone network. Crowds [17] hides a user's Web request in a crowd of other requests for the same content; similarly, our work provides privacy by hiding user-generated BT traffic in a crowd of random connections.

6. Conclusion

As P2P systems grow in size and popularity, privacy becomes an increasingly important and challenging goal. In this paper, we analyzed connection information from real users in the BT network and revealed strong communities of shared interest. We showed that this information can be exploited by an attacker to classify large numbers of users with relatively little monitoring. To address this threat, we designed and implemented a strategy to disrupt attempts to classify users.

7. References

[1] Arenas, A., Danon, L., Díaz-Guilera, A., Gleiser, P. M., and Guimerà, R. Community analysis in social networks. *Eur. Phys. J. B 38* (2004), 373–380.

[2] Bauer, K., McCoy, D., Grunwald, D., and Sicker, D. BitBlender: Light-weight anonymity for bittorrent. In *Proc. AIPACa* (Sept. 2008).

[3] Brandes, U., Delling, D., Höfer, M., Gaertler, M., Görke, R., Nikoloski, Z., and Wagner, D. On finding graph clusterings with maximum modularity. In *Proc. of Graph-Theoretic Concepts in C* (2007), LNCS.

[4] Choffnes, D., Duch, J., Malmgren, D., Guierma, R., Bustamante, F., and Amaral, L. SwarmScreen: Privacy through plausible deniability in P2P systems. Tech. Rep. NWU-EECS-09-04, Northwestern University, March 2009.

[5] Cortes, C., Pregibon, D., and Volinsky, C. Communities of interest. In *IDA* (2001), pp. 105–114.

[6] Crooks, A. BitTorrent peer exchange conventions. `http://wiki.theory.org/BitTorrentPeerExchangeConventions`.

[7] Danon, L., Díaz-Guilera, A., Duch, J., and Arenas, A. Comparing community structure identification. *J. Stat. Mech.: Theor. Exp.* (2005), art. no. P09008.

[8] Dingledine, R., Mathewson, N., and Syverson, P. Tor: The second-generation onion router. In *Proc. of USENIX Security Symposium* (2004), pp. 303–320.

[9] Duch, J., and Arenas, A. Community detection in complex networks using extremal optimization. *Phys. Rev. E 72* (2005).

[10] Guimerà, R., and Amaral, L. A. N. Functional cartography of complex metabolic networks. *Nature 433* (2005), 895–900.

[11] Guimerà, R., Mossa, S., Turtschi, A., and Amaral, L. A. N. The worldwide air transportation network: Anomalous centrality, community structure, and cities' global roles. *Proc. Natl. Acad. Sci. USA 102*, 22 (May 2005), 7794–7799.

[12] Guimerà, R., Sales-Pardo, M., and Amaral, L. A. N. Modularity from fluctuations in random graphs and complex networks. *Phys. Rev. E 70* (2004), art. no. 025101.

[13] Guimerà, R., Sales-Pardo, M., and Amaral, L. A. N. Classes of complex networks defined by role-to-role connectivity profiles. *Nature Phys. 3* (2007), 63–69.

[14] Kwak, H., Choi, Y., Eom, Y.-H., Jeong, H., and Moon, S. Mining communities in networks: a solution for consistency and its evaluation. In *Proc. of IMC* (2009).

[15] Newman, M. E. J., and Girvan, M. Finding and evaluating community structure in networks. *Phys. Rev. E 69*, 2 (2004).

[16] Popescu, B. C., Crispo, B., and Tanenbaum, A. S. Safe and private data sharing with Turtle: Friends team-up and beat the system. In *Proc. Cambridge Intl. Workshop on Security Protocols* (2004).

[17] Reiter, M. K., and Rubin, A. D. Crowds: Anonymity for web transactions. *ACM Transactions on Information and System Security 1* (1998), 66–92.

[18] Saponas, T. S., Lester, J., Hartung, C., Agarwal, S., and Kohno, T. Devices that tell on you: privacy trends in consumer ubiquitous computing. In *Proc. of USENIX Security Symposium* (2007).

[19] Siganos, G., Pujol, J. M., and Rodriguez, P. Monitoring the BitTorrent monitors: A bird's eye view. In *Proc. PAM* (April 2009).

[20] The Pirate Bay. http://thepiratebay.org/blog/175.

[21] Wasserman, S., and Faust, K. *Social Network Analysis*. Cambridge University Press, Cambridge, UK, 1994.