Why are they hiding ? Study of an Anonymous File Sharing System

Mathieu Cunche, Mohamed Ali Kaafar, Terence Chen, Roksana Boreli, Anirban Mahanti

Why are they hiding ? Study of an Anonymous File Sharing System

Mathieu Cunche*†, Mohamed Ali Kaafar†*, Jiefeng (Terence) Chen*,
Roksana Boreli*, Anirban Mahanti*

*National ICT Australia †INRIA Rhône-Alpes Grenoble France
firstname.lastname@nicta.com.au *firstname.lastname*@inria.fr

***Abstract*—This paper characterizes a recently proposed anonymous file sharing system, OneSwarm. This characterisation is based on measurement of several aspects of the OneSwarm system such as the nature of the shared and searched content and the geolocation and number of users. Our findings indicate that, as opposed to common belief, there is no significant difference in downloaded content between this system and the classical BitTorrent ecosystem. We also found that a majority of users appears to be located in countries where anti-piracy laws have been recently adopted and enforced (France, Sweden and U.S). Finally, we evaluate the level of privacy provided by OneSwarm, and show that, although the system has strong overall privacy, a collusion attack could potentially identify content providers.**

I. Introduction

Peer-to-peer (P2P) file sharing continues to be a popular service for many reasons including the large variety of content available on the P2P ecosystem, the ease of accessibility with no sign-up required for downloading content, and good system performance, particularly when downloading popular content. P2P applications such as BitTorrent, however, do little to preserve a user's privacy. The open nature of these applications enables easy monitoring and identification of both content providers and downloaders [5], [9]–[11].

In recent years, a number of file sharing solutions offering differing degrees of privacy have been adopted, arguably coinciding with the increased privacy-awareness of Internet users and the tightening, in some countries, of anti-piracy laws [3], [6]. Centralized One-Click Hosting services have emerged as an alternative to P2P file sharing [2] and potentially offer greater degree of privacy. Another approach is to augment existing P2P systems with solutions that obfuscate IP addresses, for example, by using standard BitTorrent over an anonymization network such as Tor [4] or I2P[1]. BitTorrent over Tor has been adopted by a large population of users [11], but anonymity comes at the cost of system performance [5], [11]. More recently, a promising privacy-aware P2P system, OneSwarm, was proposed [5]. OneSwarm includes a number of privacy preserving techniques, and this system is reportedly used by thousands of users [5].

In this paper, we study the usage and the characteristics of the OneSwarm system, including the types of content shared, the geographical location of its users, the size of the system and the anonymity provided by the system. We discuss the possible reasons motivating adoption of OneSwarm, compared to those for using other common P2P alternatives (e.g., BitTorrent over Tor and the standard BitTorrent) which provide varying levels of privacy. We have also developed a mechanism to exploit the OneSwarm client compatibility and the apparent use of BitTorrent for downloading content not currently available in OneSwarm.

Our key observations include the following. We characterize the use of OneSwarm, including the likely geographical locations of users and the type of content shared in the system. There are strong indications that the vast majority of users are located in France (around 50%), Sweden (close to 38%), and the US (under 10%). Our results additionally indicate that the content shared on OneSwarm is similar to that shared using other P2P systems. We argue that the motivations for using file sharing systems can be broadly categorized as sociological (e.g., the potential implications to the person downloading or searching for content, in regards to interest in specific content), legal (e.g., evading infringement notices for downloading copyrighted material), technological (e.g., enabling content download in countries where P2P traffic is filtered out by ISPs), enhanced awareness (e.g., genuine interest in preserving privacy), or a combination thereof. We find strong indicators for the prevalence of the legal/regulatory policy driven adoption, based on the user locations and the timing of OneSwarm introduction compared to regulatory developments in specific countries [3], [6].

The OneSwarm system provides strong overall privacy; however, we identify a potential collusion attack that can compromise privacy. We also discuss a privacy breach introduced by the default behaviour of the OneSwarm client, wherein the client automatically searches for content on the public BitTorrent system if the content is not found within OneSwarm.

This paper is organized as follows. Section 2 provides an overview of OneSwarm. Section 3 describes the measurement methodology. Our results are presented in Section 4. Section 5 identifies a couple of limitations in the OneSwarm design. Section 6 concludes the paper.

II. Overview of OneSwarm

OneSwarm uses an overlay network to propagate search messages and establish connections between peers for down-

[1]I2P Anonymous Network http://www.i2p2.de

loading content. The connections relate to trust relationships between nodes (e.g., established via manual or automatic exchange of public keys), where trust represents the certainty that the trusted party will not reveal information about activity of nodes (e.g., the origin of the forwarded traffic). To enable new user connections, when trusted connections cannot be established, OneSwarm allows untrusted connections, with limited interaction between the peers.

OneSwarm uses community servers to automatically establish both trusted and untrusted connections. These servers store the identities of the users and allocate connections to new users. Public community servers facilitate introduction of new users; new users are allocated a set of untrusted connections with other subscribers of this server. Currently, there are four public community servers: the default community server hosted by the University of Washington (UW CSE), two French servers, oneswarm-fr (OS-fr) and lavilette, and a Swedish server subcult[2].

OneSwarm supports two content identification methods: Unique Resource Identification (URI) based, which includes support for the infohash convention used by BitTorrent, and a content name based method. Correspondingly, two types of messages are used to search for content: *Hash search* and *Text search*. A Hash search message contains a truncated version of the infohash of the content. A Text search message contains a URI or a human readable string representing a set of keywords. Note that Hash search and Text search containing a URI target unique content, while a Text search containing keywords can target any matching content.

The OneSwarm system requires a connection to be established between the requester and the provider to facilitate content download. OneSwarm uses a flooding algorithm, wherein a search message targeting a specific content is sent by the requester and forwarded by the overlay nodes. Once the message reaches a node that has the targeted content, a response message is sent back following the same path, thus completing the connection establishment.

OneSwarm is backward compatible with the standard BitTorrent protocol. The OneSwarm client can handle *.torrent* files, and the content associated with a torrent can be downloaded either from the OneSwarm network, or from the public BitTorrent ecosystem. When connected to public BitTorrent ecosystem, the OneSwarm node concurrently becomes a content provider on OneSwarm for this content.

III. Measurement Methodology

Our measurements consider a combination of OneSwarm and BitTorrent as shown in Figure 1. The OneSwarm part comprises of public and private communities, the former having a number of community servers (CS). For the sake of simplicity, only one CS is shown in Figure 1. Similarly, BitTorrent includes public and private components, and a (large) number of trackers. The nodes which simultaneously reside in both OneSwarm communities, or in both OneSwarm and BitTorrent, are denoted as bridge nodes (B). We monitor OneSwarm messages and three major BitTorrent trackers[3] over a period of one month.

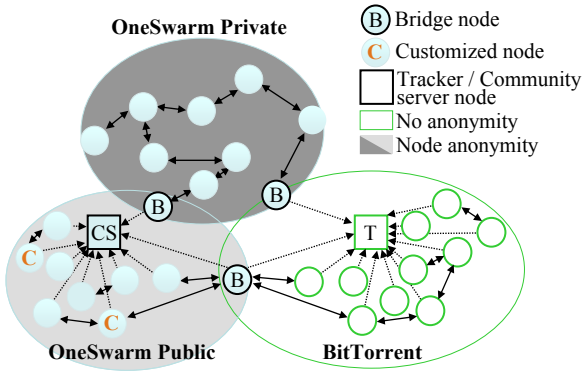

Fig. 1. Overview of the measurement environment, including the public and private OneSwarm, and BitTorrent ecosystems.

For this study, we developed a customized client derived from the OneSwarm client version 0.6. Our customized client can generate and monitor all OneSwarm message types. Our measurement setup includes connecting two customized clients to all four OneSwarm community servers. The captured OneSwarm messages are used for analysis and characterization of the system. We additionally determined the IP addresses of OneSwarm clients, first from the direct connections allocated to our nodes by the community servers, and second using a method which exploits OneSwarm's backward compatibility with BitTorrent. As a representative data set for public BitTorrent use, we collected content hashes from the Pirate Bay site. Most of our data collection is restricted to nodes that are directly interacting with the public communities. We argue that, in practice, due to bridge nodes (cf. Figure 1) we will observe a mix of traffic from both private and public communities.

A. OneSwarm Monitoring

We monitor all OneSwarm Hash search and Text search request messages and use this data to characterize the activity of the OneSwarm users. We combine this monitoring with additional BitTorrent monitoring to determine the IP addresses of bridge nodes and their geolocation. Cautionary measures were taken to prevent the monitoring nodes from generating artificial traffic or forwarding messages that could modify the overlay topology (e.g., monitoring nodes do not act as bridge nodes between different parts of the network).

During our measurement period, our nodes connected to 190 peers within the 4 public communities. A total of 630 millions[4] OneSwarm search messages have been collected, including 520 million Hash search messages and 110 million Text messages. Of those, $25,853$ were Text search messages in plain text, containing keywords which reflect user interest in specific types of content.

[2]Respectivly https://community.oneswarm.org, https://forum.oneswarm-fr.net, https://lavilette.dyndns.org:8081, https://kf.subcult.org:8081

[3]The BitTorrent trackers monitored are udp://tracker.openbittorrent.com, udp://tracker.publicbt.com and udp://tracker.istole.it

[4]This high number is explained by the fact that OS clients are periodically generating messages while downloading.

TABLE I
CONNECTIVITY BETWEEN THE DIFFERENT PUBLIC COMMUNITIES IN ONESWARM NETWORK.

	UW CSE	oneswarm-fr	lavilette
UW CSE	91.9%	83.5%	94.9%
oneswarm-fr	99.3%	99.4%	98.3%
lavilette	97.3%	80.8%	96.2%

To verify the efficiency of our passive traffic monitoring, we perform an experiment to estimate the fraction of messages captured by our monitoring nodes, compared to the total number of messages generated in the system. For this, we introduce six customized nodes and place two nodes in each of the three public communities (excluding the Swedish community server, which was intermittently available). Every 10 seconds, the controlled nodes send a search message and record all incoming traffic. From the captured traffic, we then compute the cross-community connectivity between each pair of our controlled clients, as represented by the fraction of messages received by the node in one community, from among those sent by the node in the second community. The results are reported in Table I. The observed high connectivity values, i.e. 80.8% to 99.4%, enable us to be confident that the placement of monitoring nodes enables capture the majority of the traffic originating from the public part of the Oneswarm system. It is however difficult to estimate the fraction of captured traffic originating from the private part of the network. If the structure of the overlay network is homogeneous over the overall network (public and private part), the fraction of captured traffic from the private parts should be similar to the one of the public part.

B. Identifying BitTorrent Bridge Nodes

Our method is based on the hypothesis that a number of OneSwarm clients are acting as BitTorrent bridges. This is supported by the observed behaviour of OneSwarm clients which are connecting to the public BitTorrent system and acting as content importers into OneSwarm.

We use the ability of OneSwarm clients to utilize *.torrent* files to identify the IP addresses of bridge nodes. We note that OneSwarm clients searching for a content source on OneSwarm in this way will, after an unsuccessful search period (of 90 seconds), automatically connect to the corresponding BitTorrent tracker[5]. The passive monitoring of OneSwarm captures all content search messages. We additionally use the torrent data set from the Pirate Bay site. For each new content instance observed in the Hash search message set, we correlate the truncated infohash (cf. Section II) to the full BitTorrent infohash. We then start monitoring the corresponding BitTorrent tracker for newly connected peers. The tracker is monitored for a period of 5 minutes (heuristic value). Finally, from the set of newly detected BitTorrent peers, we select those using the OneSwarm client (determined by verifying that their *BitTorrent client identifier* starts with *-OS*).

To capture the newly connected peers, we first have to capture the full set of peers currently connected to this tracker and then attempt to identify the new peers close to the time of their initial connection. Obtaining both sets of peers in the short time frame available necessitated the use of a number of proxies, as the tracker limits the rate of queries from a specific IP address. To minimize the amount of resources required for monitoring and still ensure the capture of the majority of peers, we limited our measurements to swarms with an advertised size of less than 1000 peers. We additionally note that the generous time frame allocation (5 minutes) enables a high likelihood of capturing bridge nodes, regardless of the delays in the OneSwarm overlay. This time is also required to ensure that the majority of peers, including the newly connected peers, have been collected from the tracker site.

We identified 778 unique OneSwarm users downloading content through BitTorrent. We argue with a high likelihood that these OneSwarm clients used on BitTorrent are also part of OneSwarm, due to the fact that the search for the same content is being detected within a short time frame of the captured BitTorrent activity of these clients.

detect new peers. Indeed the number of random subset required to obtain a particular element increases quickly with the size of the set (interested reader may refer to the coupon collector problem [1]).

[5]This is hard coded into the OneSwarm client version 0.6.

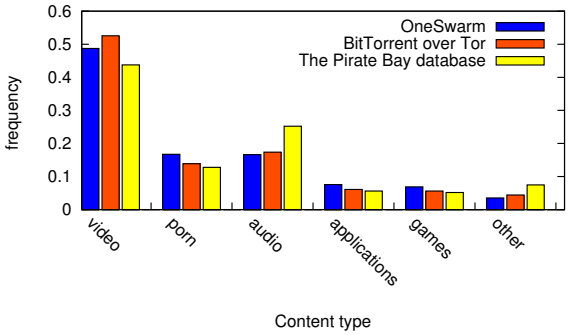

Fig. 2. Content 'type' frequency comparison between the torrents found on OneSwarm, BitTorrent over Tor, and the Pirate Bay.

IV. RESULTS

Our characterization of the OneSwarm system attempts to answer a number of questions: What is the nature of the content that is searched for and downloaded? Where are the users located? What is the size of the system? We then discuss the motivation to use this anonymous file sharing system.

A. Characterizing the Content

We compared content downloaded on OneSwarm with that found on a typical BitTorrent ecosystem and BitTorrent over Tor. For OneSwarm, we consider the torrents that have been downloaded at least once during our passive monitoring (77000 content in total). In the case of the Pirate Bay, we consider the approximately 1.3 million torrents available at the time of data collection. The BitTorrent over Tor data set is composed of torrents collected at the output of six Tor exit nodes as reported in [10].

Our analysis is based on the meta-information associated to torrents hosted by The Pirate Bay website. More particularly we consider the two level content classification, by 'type' and

by 'subtype'. Figure 2 shows the results from our analysis. We observe that the available content in either systems is similar, while noting only a minimal difference for the *video*, *porn*, and *audio* types.

B. Characterizing Keyword Based Search

The OneSwarm system allows users to search for content using keywords in a privacy preserving manner. We have collected a total of 25853 search strings issued on the OneSwarm system. We compare the most popular search strings to similar statistics available on the Pirate Bay web pages[6].

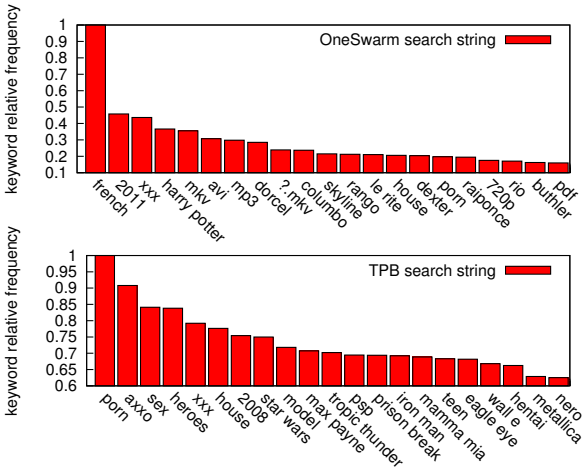

Fig. 3. Relative frequency of string searches.

Figure 3 shows the relative frequency of the most popular strings for both OneSwarm and Pirate Bay content. Notice that the most popular search string on OneSwarm are different than the corresponding ones of the Pirate Bay. The search string *french* is the most popular on OneSwarm. In general, in the OneSwarm data set we have observed very few strings related to languages other than French.

Pornography related strings are highly represented on the Pirate Bay. Although these search strings can also be found on OneSwarm, they are in smaller numbers and have smaller relative frequency. It seems that people using the anonymous search engine are not primarily interested in pornographic content, and therefore, not motivated by sociological reasons such as hiding interest in such content.

C. Geolocalisation of Users

We determined the geographic location of OneSwarm users using an IP-to-geolocation database[7]. As mentioned in Section 3, we identified users that were assigned to our monitoring nodes by the community servers (*Community server Set*) as well as users that were identified by monitoring BitTorrent bridge nodes (*BitTorrent Set*). There are 190 and 778 unique users (IP addresses) in the *Community server Set* and the *BitTorrent Set*, respectively.[8]

[6]The fonts of tag cloud, available at http://thepiratebay.org/searchcloud, show the popularity of a string relative to the popularity of the most popular string.

[7]http://www.maxmind.com/app/ip-location

[8]Our analysis was performed on-the-fly. We do not store any information such as IP addresses, user identities, or public keys.

TABLE II
COUNTRY REPRESENTATION IN ONESWARM.

	BitTorrent Set	Community Server Set
FR	52.94%	69.86%
SWE	28.12%	5.80%
US	9.03%	11.16%
AUS	1.86%	0.44%
CAN	1.86%	1.11%

TABLE III
COUNTRY REPRESENTATION IN PUBLIC COMMUNITY SERVERS.

	UW CSE	oneswarm-fr	subcult.org
FR	43.27%	90.70 %	71.42%
SWE	7.01%	1.76%	19.0%
US	25.14%	2.21%	4.76%

Table II presents results of the country-wise breakdown of OneSwarm user locations; Table III presents the same information for users found in the *Community Server Set* classified by the identity of the server[9]. Both tables show that an overwhelming majority of the users (among those discovered) are based in France. In the BitTorrent set, Sweden and the US account for the next biggest group of users. We notice that only few Swedish IPs were found on the default community server, and even on the Swedish community server (subcult.org) they represent only 19% of the users. There are two possible explanations for what we observe. Either the BitTorrent backward compatibility feature is prevalently used among OneSwarm's Swedish users, or these users do not rely on community servers to establish new connections in OneSwarm and instead manually exchange public keys. In fact, the OneSwarm official forum[10] hosts a thread, dedicated to manual exchange of public keys of Sweden-based users, that includes more than 1200 public keys. This possibly explains the low presence of users from Sweden in our Community server Set, although Sweden is well-represented (28.12% of the collected IPs) in the BitTorrent Set.

The distribution of user per country in Oneswarm largely differs of the one of BitTorrent [12]. Some countries like Netherlands, Luxembourg, China and Russia are almost absent from Oneswarm, while they are in the most represented countries on BitTorrent. A large fraction of Oneswarm's users are based in France and Sweden, countries that have enacted strong Intellectual Property Rights Protection and Enforcement laws [3], [6] (IPRPE laws). In other words, amongst the highly represented countries in the BitTorrent ecosystem, those with weak or no IPRPE laws have not adopted Oneswarm, while a subset of those with strong IPRPE laws have largely adopted in Oneswarm. The existence of IPRPE laws in a country appears to be a possible reason for Oneswarm adoption, we call this the *Copyright law effect*. The wide adoption of OneSwarm in those countries can also be explained by social propagation and the existence of community of users. Indeed regional community servers has been deployed in those countries along with forums enabling the creation of a strong community of users. In order

[9]Lavilette and subcult community servers were not included in this experiment because of their intermitent availability.

[10]http://forum.oneswarm.org

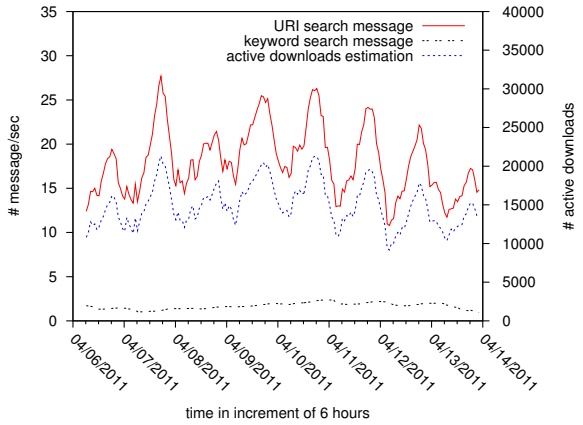

Fig. 4. System activity and number of active downloads as a function of time (GMT).

to assess the importance of the *copyright law effect*, it will be interesting to study how OneSwarm use evolves as other countries introduce stringent copyright laws.

D. Characterizing the System Activity

The search messages in OneSwarm can be divided into two categories: keyword-based and URI-based messages. To locate or download a content, either the OneSwarm user sends a Text search message, containing keywords that may target several contents, or the client sends a URI-based search message (Text or Hash search messages) targeting a unique content. Further, once the download is initiated, for performance purposes, the OneSwarm client keeps periodically[11] sending the URI-based search messages in the quest of potential new resources, or to update the list of connected known locations.

To analyze the network activity of OneSwarm, we consider the first category, i.e., the keywords-based search traffic, as an indicator of human activity as very often these messages require human-machine interactions. We then consider the second category of search messages, as an indication of active downloads. Since an active download is generating a search message for the content every 10 minutes, an estimate of the number of active downloads can be computed from the number of search messages observed over a given period of time.

Figure 4 shows our results for system activity. Time-of-day non-stationarity, similar to those observed for in many other network measurements, is observed. The observed traffic is the sum of the traffic of users in different time zones, a fraction of them being continuously connected while others may only be periodically online. According to Akamai statistics[12], the highest Internet activity in the European regions is observed between 2 PM and 6 PM GMT, which coincides with the time frame of highest activity on OneSwarm. This observation lends further credence to our hypothesis that a large fraction of OneSwarm users are located in Europe.

E. Estimation of the Population Size

By design, OneSwarm prevents observers from either identifying or crawling its users. We estimate the number of OneSwarm users by leveraging the interaction with each of the public community servers. Specifically, when a user subscribes to a community server, the latter returns a list of 20 peers that are the closest to the subscriber[13]. By subscribing multiple times to a community server, using a different randomly generated public key, we can collect different sets of peers. By estimating the relative size of redundancy occurring during multiple subscriptions, a rough estimate of the population size can be obtained.

TABLE IV
ESTIMATION OF THE COMMUNITY SERVER POPULATION SIZE.

	UW CSE	**oneswarm-fr**	**lavilette**
estimated size	2518	666	51
95 % conf.	(2002;3394)	(553; 780)	(45;59)

We estimated each of the communities' population size using a Mark-and-Recapture method and the Schumacher-Eschmeyer estimator [8] . A total of n samples are collected, each sample i containing C_i peers. We note R_i the number of peers in capture i that have already been collected in previous samples, and M_i the number of distinct peers collected before sample i. The Schumacher-Eschmeyer estimator of the server population is : $N = \frac{\sum_{i=1}^{n} C_i M_i^2}{\sum_{i=1}^{n} R_i M_i}$.

For each community server, we subscribed 20 times, each time with a different identity. The results from our estimation are shown in Table IV. The UW CSE community has the largest population, possibly because it was the first, and the default community server, for the system. Similarly, we estimated the overlap between those communities, and our results show that there were in the order of 3000 users on those community servers.

V. PRIVACY IN ONESWARM

OneSwarm provides a high level of privacy to its users. In this section, we discuss two limitations that can result in a privacy breach.

A. The case of peer identification through BitTorrent

The backward compatibility with BitTorrent can enable identification of OneSwarm users that download content from the public BitTorrent ecosystem (see Section III-B). We notice that the default OneSwarm client behaviour may not be apparent to the users, as there is no visible indication that the client switches from the anonymous to the public mode. Adding a user warning or, preferably, user control of this feature, is a potential for further improvement in OneSwarm.

B. Indirect collusion attacks

In OneSwarm, a Hash search message is forwarded only if a node does not posses the requested content. This feature can be exploited by a set of colluding nodes, to infer the possession of a content by a targeted node. A direct collusion

[11]Every 10 minutes in the OneSwarm client version 0.6.

[12]http://www.akamai.com/html/technology/nui/retail/charts.html

[13]Distance between subscribers is defined in terms of distances between users' public keys.

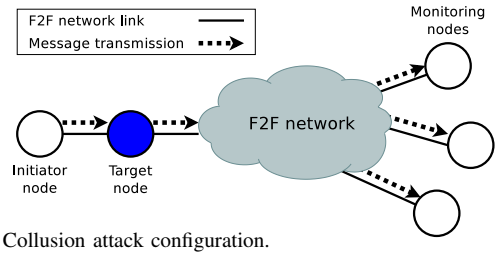

Fig. 5. Collusion attack configuration.

attack consists of an initiator that sends a Hash search message to the directly connected target node, while a set of observers monitor the traffic coming from the directly connected target. Depending on whether or not the Hash search message has been forwarded to the observers, the colluding nodes are able to infer if the content is shared by the target or not. Introduced by OneSwarm's designers, probabilistic forwarding [5] was supposed to solve this issue by making the attack ineffective unless the attacker controls a large number of nodes directly connected to the target. Prusty et. al. [7] recently shown that this countermeasure was not sufficient and that direct collusion attack was practical.

We present an even more powerful variant of the collusion attack, wherein the observers are not directly connected. Note that messages forwarded by a targeted node will propagate through the overlay network and may be detected by monitoring nodes placed anywhere in the overlay network. As shown in Section III-A, a small set of monitoring nodes is sufficient for collection of a significant portion of the traffic generated by a node in the public community (e.g., one monitoring node capture 80% or more of the traffic). By replaying the attack a number of times, and by increasing the number of monitoring nodes, the confidence in the inference can be made arbitrarily close to one. Figure 5 shows the configuration of the attack, where one *initiator node* is directly connected to the *targetet node* replay search messages, while *monitoring nodes* are connected to the F2F network, but not necessarily to the *targeted node*. The *initiator node* send search messages to the *target node*, which in turn will forward them to the rest of the F2F network. Those messages will propagate through the F2F network and eventually reach the *monitoring modes*.

By design a request message for a content shared by the target will not be forwarded, the false negative probability is therefore equal to zero. A false positive is when the message is forwarded but not received by the monitoring nodes. Meaning that the message has not been propagated to the monitoring nodes. Let M_i be the event where the $i-th$ search message sent to the target is detected by at least one monitoring node. Then, the false positive probability of a test with n replay is:

$$P_{fp} = \prod_{i=1}^{n}(1 - P(M_i|\text{content is not shared})) = (1-\alpha)^n$$

Where α is the probability that a message forwarded by the target is detected by at least one monitoring node. The value of α can be evaluated in the same way the connectivity between community was evaluated in section III-A . Its value depends on the number of monitoring nodes and the topology of the network between the target and the monitoring nodes. In the case of a target node and only one monitoring node, both connected to public communities, we can infer from Table I that $\alpha \geq 80.8\%$.

We performed an experiment to test the effectiveness of an indirect collusion attack, with one directly connected node, a target node with known content, and only one monitoring node connected to the four public community servers and 100 replays of the search message. We have been able to capture the availability of specific content with a false negative probability of 0% and false positive probability of 2.02%.

VI. Conclusion

This paper presents a measurement-driven study of the OneSwarm system, with the goal of understanding the usage of this privacy-aware file sharing systems. We observe that content shared in the system is similar to that shared in other P2P systems and that a vast majority of users are located in few countries (France, Sweden, and US). Even if propagation in social media can explain this adoption, we observe that the highly represented countries have recently adopted anti copyright infringement laws, suggesting that the adoption of OneSwarm has been motivated by legal aspect. We conclude by identifying two limitations that may compromise the privacy of OneSwarm users.

References

[1] Ilan Adler and Sheldon M. Ross. The coupon subset collection problem. *J. Applied Probabability 38 (2001), no. 3, 737–746*.

[2] Demetris Antoniades, Evangelos P. Markatos, and Constantine Dovrolis. One-click hosting services: a file-sharing hideout. In *Proceedings of the 9th ACM SIGCOMM conference on Internet Measurement Conference*, pages 223–234, New York, NY, USA, 2009. ACM.

[3] Max Colchester. All eyes on france as officials enforce new antipiracy law. The Wall Street Journal , 29 October 2010. http://yaleglobal.yale.edu/content/france-antipiracy-law.

[4] Roger Dingledine, Nick Mathewson, and Paul Syverson. Tor: The second-generation onion router. In *Proceedings of the 13 th Usenix Security Symposium*, 2004.

[5] Tomas Isdal, Michael Piatek, Arvind Krishnamurthy, and Thomas Anderson. Privacy-preserving P2P data sharing with OneSwarm. In *Proceedings of the ACM SIGCOMM 2010*, volume 40, pages 111–122, New York, NY, USA, August 2010. ACM.

[6] Office of the United States Trade Representative Ambassador Ron Kirk. 2010 Special 301 Report.

[7] Swagatika Prusty, Brian Neil Levine, and Marc Liberatore. Forensic Investigation of the OneSwarm Anonymous Filesharing System. In *Proc. ACM Conference on Computer & Communications Security (CCS)*, October 2011.

[8] F. X Schumacher and R. W. Eschmeyer. The estimation of fish populations in lakes and ponds. *Journal of the Tennessee Academy of Sciences*, 18:228–249., 1943.

[9] Georgos Siganos, Josep M. Pujol, and Pablo Rodriguez. Monitoring the bittorrent monitors: A bird's eye view. In *Proceedings of the 10th International Conference on Passive and Active Network Measurement*, PAM '09, pages 175–184, 2009.

[10] Stevens Le Blond, Arnaud Legout, Fabrice Le Fessant, Walid Dabbous, and Mohamed Ali Kaafar. Spying the World from your Laptop – Identifying and Profiling Content Providers and Big Downloaders in BitTorrent. In *3rd USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET'10)*.

[11] Stevens Le Blond, Pere Manils, Chaabane Abdelberi, Mohamed Ali Kaafar, Claude Castelluccia, Arnaud Legout, and Walid Dabbous. One Bad Apple Spoils the Bunch: Exploiting P2P Applications to Trace and Profile Tor Users. In *4th USENIX Workshop on Large-Scale Exploits and Emergent Threats (LEET '11)*.

[12] Chao Zhang, Prithula Dhungel, Di Wu, and Keith W. Ross. Unrevealing the bittorrent ecosystem. *IEEE Transactions on Parallel and Distributed Systems*, (PP Issue:99).