Security Issues in Structured P2P Overlay Networks

Mikko Vestola
Helsinki University of Technology
mikko.vestola@tkk.fi

Abstract

Nowadays, P2P networks are used for many purposes, such as files sharing, instant message communication and distributed computing. Popular services such as Skype, BitTorrent and eMule rely on P2P networks. This makes the networks an attractive target for attackers. Over time, researchers have discovered some major security problems with P2P networks, which most of them have been now well-known for a long time. This study describes the most important security issues in the overlay level of structured P2P networks. The following attacks are included: Sybil attack, ID mapping attack, Eclipse attack, identity theft and churn attack. These attacks are not just theoretical, but, some of them are surprisingly easy to perform in real-life P2P networks. Several countermeasures exist, which are analyzed in this paper, as well as how the attacks are related to each other. This study shows that structured P2P networks can be seriously compromised if they are not effectively protected against these attacks. For example, in an unprotected distributed file sharing network, a malicious user can intercept file requests and return data of its own choosing. In the worst case, an adversary might eventually be able to gain full control over the whole network and cause a denial-of-service attack.

KEYWORDS: P2P, security, identity assignment attacks, routing level attacks, Eclipse attack, Sybil attack

1 Introduction

Peer-to-Peer (P2P) networks have turned out to be a popular network technology as they allow the design of low cost and high availability content distribution systems. These networks are based on a distributed architecture where the clients of the network also act as servers. The content is distributed directly between the participants of the network (peers), which also results in distributing the load of the underlying physical network. In contrast, the traditional client-server networks are based on a centralized architecture where the content is stored and provided only via a central server(s). When the number of users in the network is large, the traditional centralized architecture becomes very expensive because the network needs more servers and bandwidth to be able to serve the connecting clients.

Compared to the traditional client-server networks, P2P networks are very dynamic and inexpensive since there is no need for centralized servers. On the other hand, such networks are typically much harder to design. Moreover, the security of the networks is a great concern as peers join and leave the network without any central control. Nowadays, P2P networks are used for many purposes, such as files sharing, instant message communication and distributed computing. Popular services, such as Skype, BitTorrent and eMule rely on P2P networks. However, the large number of users using these services has also attracted attackers to exploit the security problems in P2P networks.

This paper answers the following research questions based on the current research presented in the literature:

1. What types of attacks are there against structured Peer-to-Peer networks in the overlay network level?

2. Are there effective countermeasures to all of these attacks?

Some similar types of studies exist [22, 20], however, they focus on different types of attacks than this paper. The focus in this survey is on overlay network level attacks, such as overlay routing level attacks. Neither application level attacks (such as index poisoning or storage and retrieval attacks) nor the attacks against the underlying network (such as attacks against TCP protocol) are in the scope of this study. This paper gives a fresh view on the security issues in P2P networks and, hopefully, helps to build safer Peer-to-Peer overlay networks.

The rest of this paper is organized as follows. In section 2, we [1] introduce the general background information about the different types of P2P networks, which are susceptible to different types of attacks. These attacks and their countermeasures are more deeply studied in section 3. Finally, section 5 analyzes the relationship between the presented attacks and also gives conclusions about the current level of security in P2P networks.

2 P2P technology

2.1 P2P networks

In this section, we briefly describe the characteristics of the different types of P2P networks, which all have their strengths and weaknesses. Peer-to-Peer networks are so called overlay networks. This means that the network is a virtual network build on top of another network, that is on top of the Internet Protocol (IP) network. As described in [6], P2P networks can be classified into two categories: unstructured and structured networks. The following subsec-

[1] Use of the plural pronoun is customary even in solely authored research papers and thus is also used in this paper.

tions will describe the differences between these two types of network structures.

2.1.1 Unstructured P2P networks

The first Peer-to-Peer networks were based on the concept of unstructured networks [6]. In an unstructured P2P network, the links between nodes are established arbitrarily. There is no correlation between a peer and the content managed by it. In other words, the content might be stored anywhere in the network and it must be searched using flooding. If a node wants to find a piece of data from the network, the node has to flood the query through the network to find as many peers as possible which share the data. When a peer receives the flood query, it sends a list of all content matching the query to the originating peer. The main disadvantage of flooding is that it generates a huge amount of signaling traffic to the network and hence such networks typically have very poor search efficiency. In addition, as the routing mechanism is based on best effort, a peer looking for rare data shared by only a few other peers, might not get a reply even though the data is available in the network.

Early unstructured networks also used centralized servers to store the IP addresses of peers sharing content. This type of P2P network structure is known as centralized P2P network, whereof Napster is a well-know example. Although these networks still needed a centralized server to index peers, the approach greatly reduced the load of the centralized server because it didn't have to distribute the actual files like in pure client-server architecture. The problem with centralized P2P networks is, however, that the centralized server can be a single point of failure. In addition to Napster, other examples of unstructured P2P networks are Freenet, Gnutella, FastTrack/KaZaA, BitTorrent and Overnet/eDonkey 2000 [12].

2.1.2 Structured P2P networks

In contrast to the loosely organized unstructured networks, the topology of structured P2P networks is tightly controlled. The content in structured networks is placed not at random peers but at specified locations. The overlay network assigns keys to data items and organizes its peers into a graph that maps each data key to a peer. This enables efficient discovery of data items using the key of a data element.

Structured P2P networks are usually based on distributed hash tables (DHT), which are decentralized and distributed systems providing a lookup service similar to a hash table. The most fundamental aspect of DHT-based P2P networks is the existence of identifiers for both nodes and keys. In DHT-based P2P networks, each node has a unique identifier. Likewise, each data item also has an identifier. The DHT is used to store the [key, value] pairs where the key is the identifier of the data item and the value is the identifier of the node responsible for the data item. Participants of the network can then perform effective searches for files based on the data item identifier. This allows DHTs to scale to extremely large numbers of nodes.

DHTs have been used in numerous popular Peer-to-Peer systems in the real world. One of the most popular DHT-based structured network is the Kademlia overlay protocol. Real-life implementations which use the Kadmelia network include popular services such as the KAD network, eMule and BitTorrent's distributed tracker [20]. Other examples of structured DHT-based networks are Content Addressable Network (CAN), Tapestry, Chord, Pastry, and Viceroy [12].

3 Attacks and protection

In this section, we focus on attacks and their countermeasures in structured P2P networks. The attacks introduced in this section are categorized based on the classification presented in [25] with some modifications. Since the focus of this paper is on the overlay network level, the attacks on the application level are out of the scope of this study.

The reader should be noted that the attacks presented in this study are not just theoretical, but, they can be quite easily applied in real-life implementations. For example, two studies [7, 18] have analyzed the effects of the Sybil attack (see section 3.1.1) and the Eclipse attack (see section 3.2.1) in the popular P2P network Kadmelia. Both studies concluded that the attacks are surprisingly easy to perform in the Kadmelia network and can seriously compromise the whole network. No special hardware was needed but the attacks can be launched from a single PC connected to the Internet via a broadband connection.

3.1 Identity assignment attacks

Identity assignment attacks in P2P networks are based on the weaknesses of assigning identities to the participants of P2P networks. Before joining a P2P network, every peer must usually generate a user identifier (ID). These user identifiers, or identities, uniquely identify participants (nodes) in a P2P network much like IP addresses uniquely identify participants of the Internet. The IDs in P2P networks are used, for example, as the basis of routing and mapping content directly onto nodes. For this reason, the proper assignment and use of IDs are essential to the correct operation of the network. One physical entity of the network is assumed to own one random identity to participate the network. However, the assignment of IDs is usually not controlled enough in P2P networks. This allows malicious users to perform different types of attacks against the network. The following subsections will describe the two most important identity assignment attacks: the Sybil attack and the ID mapping attack, which are both closely related to each other.

3.1.1 Sybil attack

The Sybil attack is one of the most challenging and difficult problems to solve in decentralized Peer-to-peer networks. The attack was first described by Douceur in the year 2002 [5]. In a Sybil attack, a single malicious user creates multiple fake peer identities and pretends to be multiple, distinct physical nodes in the system. These fake identities are called *Sybils*.

If an adversary is able to create a large number of identifiers, it can control the network substantially. For example, if a malicious user can choose its identifier arbitrary, it can allocate itself a collection of identifiers closer to some resource's key than any existing node in the system [21]. This would allow the malicious user to censor the resource from the network. Moreover, an adversary can maximize its chances of appearing in a victim node's routing tables by generating a huge number of shadow identifiers. The malicious user could then mediate or censor the victim's communication on the overlay network [21].

The designers of the original structured P2P overlays paid little attention to the severity of Sybil attacks. Most protocols either ignore it or include limited defenses. For example, the CAN protocol assumes that nodes pick random IDs when they enter the network. However, CAN does not monitor the ID assignment, which allows an adversary to easily create many IDs and compromise the network. In contrast, Chord and Pastry limit the number of user identifiers per single physical participant, at least in theory. The designers of Chord and Pastry specified that the user identifier is the hash of user's IP address. In principle, this should prevent users for having multiple identifiers. However, a malicious user can simultaneously spoof many IP addresses to quickly obtain a multitude of identities. The adoption of IPv6 addresses also makes this defense ineffective since acquiring a large number of IPv6 addresses is much more easier than obtaining IPv4 addresses. [15]

Many studies have been conducted to solve how to prevent Sybil attacks [4, 10, 11, 13, 15]. Most of them approach the problem by increasing the cost of creating a new identity, which allows limiting the number of identities a single user can have. The cost is usually material, computational or social [11].

An example of material cost is to link the identities to smartcards which are provided by some trusted third party. This solution is not, however, so practical as the entrance barrier of the network is quite high even for a legitimate user and the trusted third party has a major control over the network.

A more practical solution utilizing material cost is the *Self-Registeration (SR)* method presented in [4]. The main idea of SR is to use the P2P network itself as a registration entity and to bind user identifiers to IP addresses. The material cost is thus the cost of acquiring an IP address. In SR, when a node wants to join a P2P network, the node calculates its identifier based on the used IP address and port number. The node then sends its ID to the nodes already successfully registered to the network. These nodes will then verify the registration of the new participant (e.g. check that the node has not reached the maximum limit of identifiers per single user). This method allows only a limited number of identifiers per IPv4 address and a limited number of identifiers per IPv6 address prefix. The new node can only join the network if the majority of the existing registered nodes allow it to join the network.

One problem with SR is that the cost of acquiring an IP address is nowadays decreasing. As criminals has shown, it is not so difficult to acquire a large number of IP addresses by using a collection of compromised computers from criminals' botnets. Mashimo et al. [13] pointed out another problem with SR: It is effective only when the fraction of malicious nodes in the network is low. The probability of false registrations (accepting malicious nodes or rejecting legitimate nodes) increases when using SR. Therefore, the method does not account for an increase in the number of malicious users. Mashimo et al. proposed an enhanced decentralized authentication scheme called *Self-Registeration with Judgement evaluation (SRJE)*. SRJE adds a survillance menchanism to SR where the evaluated values of nodes which send faulty judgements are lowered and subsequent judgements from these nodes are depreciated. SRJE was indeed shown to be more effective than SR but still did not completely prevent Sybil attacks.

The second method to control the cost of joining a P2P network is applying computational cost. For example, Rowaihy et al. propose a challenge-based admission control system (ACS) [15] to mitigate Sybil attacks. The ACS limits the rate at which a node can obtain IDs by controlling the amount of effort needed to acquire identifiers. In ACS, when a node wishes to join a network, it is challenged by the other nodes of the network with a cryptographic puzzle, which the node must solve in order to join the network. Although the effort is not overly burdensome to a single node, it makes it difficult for a malicious user to acquire a large fraction of IDs. Rowaihy et al. showed that an adversary must perform days or weeks of effort to obtain a small percentage of nodes in small P2P networks [15]. It takes a malicious user just over 3 days to obtain 10% of the IDs in a network of only 8,000 nodes. Although this approach clearly limits the number of Sybils, an attacker with enough computing power is still able to generate a large number of identities.

The third method to increase the cost of creating a new identity is to obtain identities through social relationships. SybilGuard [24] is an example of a method utilizing social relationships. The SybilGuard protocol is based on the social network among user identities. The network can be seen as a graph where a node represents an identity and an edge between two nodes indicates a human-established trust relationship (like friend relations). Although a malicious user can create many nodes, these nodes have only a few trust relationships [24]. Therefore, Sybil nodes can be detected from the graph because there is a "cut" in the graph between the Sybil nodes and the honest nodes (see Figure 1). SybilGuard uses a special kind of verifiable random walk in the graph and examines the intersections between such walks to discover Sybil nodes. According to the authors, SybilGuard

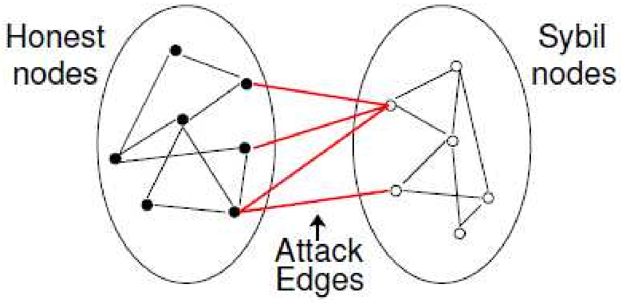

Figure 1: The principle behind SybilGuard and SybilLimit: Sybils can be detected based on the social networks with honest nodes and Sybil nodes. [24]

guarantees that with high probability, an honest node only accepts a bounded number of Sybil nodes. The authors further developed the protocol and published an improved version of it called SybilLimit [23]. The enhanced protocol is based on SybilGuard but is said to offer 200 times improvement over SybilGuard being a near-optimal defense against Sybil attacks using social networks [23].

Recently, also other types of countermeasures based on social relationships were proposed to cope with Sybil attacks. For example, [11] describes an identification scheme based on invitations and on the moderation of their delivery. To obtain a valid identifier on the network, a user has to be invited by an existing member. According to the authors, the proposed method should prevent a member from controlling a large fraction of identifiers and from choosing his identifier. Another novel solution, named *SyMon* has been described in [10]. In SyMon, every peer is associated with another non-Sybil peer, referred to as SyMon (Sybil Monitor). The chosen SyMon prevents Sybils from targeting honest peers by monitoring the transactions involving the given peer. The authors say that the approach is the first attempt to defend against Sybil attack through transaction monitoring process.

3.1.2 ID mapping attack

Another important identity assignment attack is the one called *ID mapping attack*, which is closely related to the Sybil attack described above. The difference between the Sybil attack and the ID mapping attack is that the first one is used to generate a large number of random identifiers, whereas, the latter is utilized to obtain some particular identifiers. ID mapping attacks are possible because some networks allow a participant to choose its identifier. If a user can choose its own identifier, the user can obtain a particular position on the overlay network. This will eventually allow a malicious user to gain control over certain resources. For example, a malicious user could take control over a target resource by obtaining valid identifiers which are all closer to the target resource than any of the nodes responsible for it.

Cerri et al. proved in [2] that even requiring nodes to have random identifiers is not enough to prevent the ID mapping attack. Although a node could not choose its identifier directly but is forced to create a random identifier (e.g. by applying a hash function to the users public key), the attacker could still choose its identifier indirectly by repeatedly generating a new identifier until an ID that is sufficiently close to the target one is acquired.

Because a random identifier is not enough to prevent a user from choosing its identifier, the ID mapping attack can be protected only if the identifier depends on some piece of information outside of the control of a node [2]. A possible solution could be to use some sort of centralized authority which distributes the identifiers. However, a centralized authority is not a feasible solution in completely distributed structured networks. The centralized authority would be a potential single point of failure, which is not really acceptable in P2P networks.

A better solution named *constrained ID selection mechanism* was presented in [2]. This forces a node to derive its identifier from its IP address and port number and hashing the outcome. This prevents a user from choosing a particular identifier or indefinitely ask for a new one until it acquires a desired value.

3.2 Routing level attacks

The identity assignment attacks, described at the previous section, were based on the problems in assigning valid identities to the participants of P2P networks. Instead, routing level attacks are performed by exploiting the weaknesses in the routing mechanisms of the overlay network. Because routing in Peer-to-Peer networks relies heavily on assigned identifiers, there is a close relationship between identity assignment attacks and routing level attacks. Many of the routing level attacks can be initialized or amplified by performing an identity assignment attack. The following subsections describe the three most important routing level attacks faced in P2P networks: Eclipse attack, identity theft attack and churn attack.

3.2.1 Eclipse attack

The routing mechanisms in DHT-based P2P networks are based on the principle that each node in the network maintains its own local, relatively small routing table which contains links to a set of neighbor nodes. When a node wants to send a message to some other node, it needs to perform a lookup which tries to resolve the IP address of the receiving node. If the destination node is not in the local routing table of the source node, the source node will perform a lookup from its neighbors. The neighbors will then return a set of identifiers closest to the target node they know. This continues iteratively when, at some point, the correct IP address for the destination node is received. Additionally, a node may give the message directly to some of its neighbors to be routed to the destination node. In both cases, the source node must trust that the neighbor nodes behave correctly.

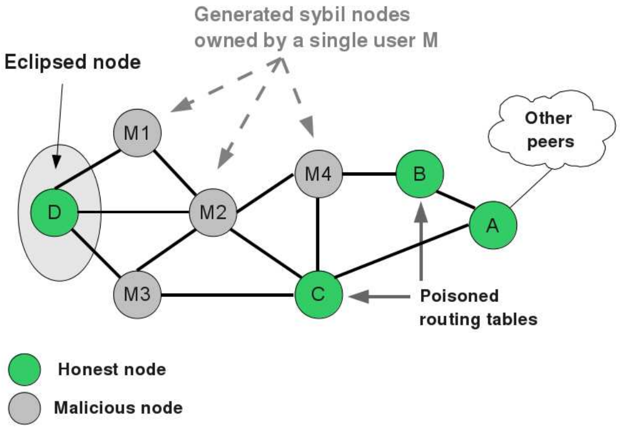

Figure 2: An example P2P network where a malicious user M has generated fake identities M1-4. Node D is eclipsed, i.e. malicious nodes take full control over its traffic. Nodes B and C have malicious entries in their routing tables.

Proper network operation requires that the nodes are able to send messages by forwarding them through their neighbor nodes and that the neighbor nodes answer correctly to lookup requests. If an attacker controls a sufficient fraction of the neighbors of correct nodes, the malicious nodes can "eclipse" some correct nodes so that all requests will be routed across the attacker (see Figure 2). The attacker can then, for example, drop the messages or provide fake answers for lookup requests. This attack is known as *Eclipse attack* or *routing table poisoning attack* [20].

The Eclipse attack is closely related to the Sybil attack described previously in this study. A malicious user can exploit a Sybil attack to launch an Eclipse attack by generating a large number of fake identities. Therefore, preventing Sybil attacks also helps to mitigate Eclipse attacks. However, although a Sybil attack is usually used as the base of an Eclipse attack, even the most effective defenses against Sybil attacks (e.g. certified node identities) do not completely prevent Eclipse attacks because attackers may manipulate the overlay maintenance algorithm to mount an Eclipse attack [17]. This is possible since the nodes in P2P networks periodically discover new neighbors by consulting the neighbor sets of existing neighbors. A malicious user can exploit this by advertising neighbor sets which consist of only other malicious nodes. For that reason, a small number of malicious nodes with legitimate identities is sufficient to carry out an Eclipse attack. Furthermore, an Eclipse attack can be used to facilitate other attacks, such as denial-of-service or censorship attacks. In the worst case, an adversary might gain full control over all overlay traffic.

Several countermeasures for Eclipse attack has been described [7, 9, 16, 17, 20]. Singh et al. [17] presented a defense against Eclipse attacks based on anonymous auditing of nodes' neighbor sets. In other words, if a node has significantly more links than the average, it might be mounting an Eclipse attack. When all nodes in the network perform this auditing routinely, malicious users are discovered and can be removed from the neighbor sets of correct nodes.

Another study [7] analyzed the effects of Eclipse attacks in the popular DHT-based P2P network *Kadmelia*, which is used primarily for file sharing. The authors found out that the Kadmelia protocol is clearly susceptible to Eclipse attacks. In Kadmelia, each participant of the network has a randomly generated identifier (160 bit hash of a random value). Each file is distributed over the same identifier space. If a node A wants to share a file, it calculates the hash H of the file. The node A then finds the closest node to the key H, say B, which will become the node responsible for this key. When some other node, say C, wants to download the file, the node C must know the ID of the file and perform a lookup for this key. The node C should eventually find out that the node B is responsible for the file and receive the IP address of the node A from the node B. The node C can then open a TCP connection to the node A and start downloading the file. However, if either node B or C is eclipsed by a malicious user, the attacker can respond with a fake IP address and thus being able to hide the file from the network or to provide some bogus file. The simulations performed in [7] confirmed that if an attacker is able to choose its identifier arbitrarily and place those identifiers in the network before the file is published, the attack reports almost 100% of success. This means that almost all the requests for the file are captured by the malicious user.

According to [20], the most basic defense against the Eclipse attack is to constrain the identifiers of nodes that can be used in routing tables. This can be achieved by using node identifiers issued by a trusted central authority. However, as stated earlier, these central authorities have always the risk of being a single point of failure.

In contrast to centralized solutions, Castro et al. proposed a decentralized approach [1] where they use two routing tables: an optimized routing table and a verified routing table. The first one is used in the normal operation, whereas, the second one is used in the case of routing failures and contains only entries which can be verified. The approach was, however, criticized by Condie et al. [3]. They pointed out that the poisoning in the optimized routing table tends to increase over time. Therefore, they proposed an improvement called *induced churn* [3]. The method forces each node in the network to periodically leave the overlay and rejoin with a new identifier while resetting their optimized routing table to the contents of the verified routing table. This makes the optimized routing table less efficient but more attack-resistant. Also, the forced unpredictable identifier changes will impair the opportunities for an adversary to perform targeted Eclipse attacks. The induced churn has been said [20] to provide an adequate defense against the Eclipse attack. However, it will generate some overhead for networks when nodes periodically join and leave with new identifiers.

3.2.2 Identity theft attack

In DHT-based P2P networks, each content item (e.g. a single file in a distributed file system) is assigned a key, which is mapped to a unique live node, called the key's root node. This root node is usually defined as the peer with user identifier closest to the key. If some other node wants to deliver a message to this root node (e.g. a node requesting the contents of a file), it uses so called key-based routing where the message is routed through the other nodes of the overlay network. Because of scalability, each node of the network only knows a small fraction of other nodes. In other words, the nodes have very small local routing tables, which contain just a limited number of neighbor nodes. A node wanting to deliver a message to the root node of some key just has to trust that the other nodes will route the message to the correct root node. [8]

This trust, however, allows an adversary to perform an *identity theft attack* by exploiting the fact that each node only sees a small subset of the overlay members. If there is a malicious node on the route of the message, the node can intercept the message and respond to the source claiming to be the root node of the key. By claiming to be the root node, the attacker can intercept application requests and return data of its own choosing. For example, the attacker can hijack a request for a block of file in a distributed file sharing system and respond with fake data. The attack can be amplified by performing a Sybil or an Eclipse attack, which were described in more detail earlier in this paper.

Puttaswamy et al. have proposed [14] a method for securing P2P networks against identity theft attacks. The method uses existence proofs, blacklists and malice-aware routing and it was shown to effectively detect, mark and redirect traffic away from attackers. The proposed method is based on the principle where nodes detect identity thefts through the generation and timely dissemination of self-verifying "existence proofs". These proofs are digitally signed certificates, which include the signer's user identifier and a timestamp signed by the sender. Overlay nodes periodically construct and distribute these proofs to randomly selected "proof managers", which store these proofs and provide them on request. Based on the existence proofs, nodes can detect identity thefts by verifying existence of nodes matching closer node identifier to the key.

3.2.3 Churn attack

The third important routing level attack in structured P2P networks is based on the inherent property of P2P systems: peers are constantly joining and leaving the network. A peer joins the network when a user starts an application (e.g. a file sharing application) and leaves the network when a user exits the application. The independent arrival and departure of thousands or millions of peers creates a collective effect called *churn* [19]. This effect needs to be taken into account in the design of any P2P system. As churn (the rate at which the peers join and leave) increases, both the latency and the probability of DHT queries failing increases. A malicious user could exploit the churn effect by generating peers joining and leaving the network fast enough to destabilize the routing infrastructure. Again, the Sybil attack can be used to amplify the impact of churn. When churn is high, the network has to transfer much extra data to maintain the network's stabilization, which impairs efficiency.

Surprisingly, unlike Sybil and Eclipse attacks, the attacks utilizing churn are not widely studied. Clearly, churn might be an attractive tool for an adversary to perform attacks against P2P networks. However, because of little research on churn, it is hard to say how easy an attacker can exploit churn in practice and how high churn affects the different implementations of structured P2P networks. Stutzbach and Rejaie studied churn in [19]. They criticized that the characteristics of churn in large-scale P2P systems are not currently well understood. The reason for this is mainly that it is hard to monitor and measure churn. In other words, because of the large size and highly dynamic nature of P2P networks, it is challenging to acquire information about the arrival and departure of peers.

To cope with churn, Stutzbach and Rejaie pointed out that P2P networks should be designed to be able to efficiently handle the large number of peers joining the system for just a few minutes. In practice, this means that each peer should prefer selecting long-lived peers as neighbors to ensure better connectivity and resiliency against churn. Otherwise, churn could significantly affect the connectivity of P2P overlays.

4 Attack relationship analysis

Although each of the attacks presented at the previous sections has its own characteristics, they are closely related to each other and do not usually exist separately [25]. One attack can be just used to create convenient conditions to perform another. For the designers of P2P networks, it is important to understand the relationship between the attacks in order to focus on preventing the most harmful ones.

Clearly, the Sybil attack is the most harmful. Together with the ID mapping attack, it can be used to significantly amplify the effects of other attacks, such as Eclipse attack. By generating a lot of shadow identities (Sybil attack) using targeted identifiers (ID mapping attack), a malicious user can easily create an effective Eclipse attack [7, 18]. Furthermore, Eclipse attack can be used to generate identity theft attack, which can be further used to perform a denial-of-service attack. For example, if an adversary has eclipsed some node, the attacker can basically steal the node's identity and claim to own any content associated with the node.

As we can see, eliminating the possibility for Sybil and ID mapping attacks would clearly make other attacks more difficult. Therefore, the safe management of identity assignment is particularly important in designing secure P2P systems. Using a trusted centralized authority to distribute the user identifiers would be a solution to many of the problems. However, it is not really suitable for decentralized networks.

5 Conclusions

In this paper, we introduced the most important attacks and their countermeasures in structured P2P networks. Two types of attacks were presented: identity assignment attacks and routing level attacks. The Sybil attack and the ID mapping attack are examples of identity assignment attacks. Both attacks are major threats to P2P networks, especially the Sybil attack. There are several countermeasures to mitigate the effects of identity assignment attacks, however, none of these provide full protection. Routing level attacks, such as the Eclipse attack and the identity theft attack are usually initialized or amplified by performing an identity assignment attack. Therefore, preventing identity assignment attacks should be the top priority.

To sum up, effective and secure P2P networks are hard to design. The fact is that there is and there will always be adversarial users in the network. A fraction of peers will always act maliciously. Unfortunately, the security in current P2P systems is still weak. Most of the attacks presented in this paper are relatively easy to perform in real-life P2P networks. Much more work is needed to make these networks safe. Although many promising countermeasures have been proposed against the most critical attacks, it has been criticized [18] that the solutions are not sufficiently practical because they impose heavy constraints on the networks and require procedures that are difficult to implement. For example, the SybilLimit protocol can clearly mitigate the effects of Sybil attacks. However, implementing it to existing P2P networks might not be easy since the protocol is quite complicated. Thus, there is a need for solutions that are technically feasible and easy to implement, in other words, simple but effective solutions.

References

[1] M. Castro, P. Druschel, A. Ganesh, A. Rowstron, and D. S. Wallach. Secure routing for structured peer-to-peer overlay networks. *SIGOPS Oper. Syst. Rev.*, 36(SI):299–314, 2002.

[2] D. Cerri, A. Ghioni, S. Paraboschi, and S. Tiraboschi. ID mapping attacks in P2P networks. In *Global Telecommunications Conference, 2005. GLOBECOM '05. IEEE*, volume 3, Dec. 2005.

[3] T. Condie, V. Kacholia, S. Sankararaman, J. M. Hellerstein, and P. Maniatis. Induced churn as shelter from routing-table poisoning. In *In Proc. 13th Annual Network and Distributed System Security Symposium (NDSS)*, 2006.

[4] J. Dinger and H. Hartenstein. Defending the Sybil attack in P2P networks: taxonomy, challenges, and a proposal for self-registration. Apr. 2006.

[5] J. R. Douceur. The Sybil attack. In *IPTPS '01: Revised Papers from the First International Workshop on Peer-to-Peer Systems*, pages 251–260. Springer-Verlag, 2002.

[6] J. Eberspächer and R. Schollmeier. First and second generation of peer-to-peer systems. pages 35–56, 2005.

[7] R. Fantacci, L. Maccari, M. Rosi, L. Chisci, L. Aiello, and M. Milanesio. Avoiding Eclipse attacks on Kad/Kademlia: An identity based approach. In *Communications, 2009. ICC '09. IEEE International Conference on*, pages 1–5, Jun. 2009.

[8] L. Ganesh and B. Zhao. Identity theft protection in structured overlays. pages 49 – 54, Nov. 2005.

[9] K. Hildrum and J. Kubiatowicz. Asymptotically efficient approaches to fault-tolerance in peer-to-peer. In *In Proc. of DISC*, pages 321–336, 2003.

[10] B. Jyothi and J. Dharanipragada. SyMon: Defending large structured P2P systems against Sybil attack. In *Peer-to-Peer Computing, 2009. P2P '09. IEEE Ninth International Conference on*, pages 21–30, Sept. 2009.

[11] F. Lesueur, L. Me, and V. Tong. A sybilproof distributed identity management for P2P networks. In *Computers and Communications, 2008. ISCC 2008. IEEE Symposium on*, pages 246–253, Jul. 2008.

[12] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim. A survey and comparison of peer-to-peer overlay network schemes. *IEEE Communications Surveys and Tutorials*, 7:72–93, 2005.

[13] Y. Mashimo, M. Yasutomi, and H. Shigeno. SRJE: Decentralized authentication scheme against sybil attacks. In *Network-Based Information Systems, 2009. NBIS '09. International Conference on*, pages 220–225, Aug. 2009.

[14] K. Puttaswamy, H. Zheng, and B. Zhao. Securing structured overlays against identity attacks. *Parallel and Distributed Systems, IEEE Transactions on*, 20(10):1487–1498, Oct. 2009.

[15] H. Rowaihy, W. Enck, P. McDaniel, and T. La Porta. Limiting Sybil attacks in structured P2P networks. pages 2596 –2600, May 2007.

[16] A. Singh, M. Castro, P. Druschel, and A. Rowstron. Defending against Eclipse attacks on overlay networks. In *EW11: Proceedings of the 11th workshop on ACM SIGOPS European workshop*, page 21. ACM, 2004.

[17] A. Singh, T.-W. Ngan, P. Druschel, and D. S. Wallach. Eclipse attacks on overlay networks: Threats and defenses. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1–12, Apr. 2006.

[18] M. Steiner, T. En-Najjary, and E. W. Biersack. Exploiting KAD: possible uses and misuses. *SIGCOMM Comput. Commun. Rev.*, 37(5):65–70, 2007.

[19] D. Stutzbach and R. Rejaie. Understanding churn in peer-to-peer networks. In *IMC '06: Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, pages 189–202. ACM, 2006.

[20] G. Urdaneta, G. Pierre, and M. van Steen. A survey of DHT security techniques. *ACM Computing Surveys*, 2009. available at `http://www.globule.org/publi/SDST_acmcs2009.html`.

[21] D. S. Wallach. A survey of peer-to-peer security issues. In *In International Symposium on Software Security*, pages 42–57, 2002.

[22] L. Wang. Attacks against peer-to-peer networks and countermeasures. *Paper on the course T-II0.5290 Seminar on Network Security at TKK*, 2006. available at `http://www.tml.tkk.fi/Publications/C/22/papers/Wang_final.pdf`.

[23] H. Yu, P. B. Gibbons, M. Kaminsky, and F. Xiao. SybilLimit: A near-optimal social network defense against Sybil attacks. *Networking, IEEE/ACM Transactions on*, PP(99):1 –14, 2009.

[24] H. Yu, M. Kaminsky, P. B. Gibbons, and A. Flaxman. SybilGuard: defending against sybil attacks via social networks. In *SIGCOMM '06: Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 267–278. ACM, 2006.

[25] X. Yue, X. Qiu, Y. Ji, and C. Zhang. P2P attack taxonomy and relationship analysis. In *ICACT'09: Proceedings of the 11th international conference on Advanced Communication Technology*, pages 1207–1210. IEEE Press, 2009.