Private Peer-to-Peer Networks

Michael Rogers and Saleem Bhatti

Abstract This chapter offers a survey of the emerging field of private peer-to-peer networks, which can be defined as internet overlays in which the resources and infrastructure are provided by the users, and which new users may only join by personal invitation. The last few years have seen rapid developments in this field. We describe deployed systems, classify them architecturally, and identify some technical and social tradeoffs in the design of private peer-to-peer networks.

1 Introduction

Most peer-to-peer networks are designed to be open to the public: anyone can join a Kademlia overlay or share files in Gnutella simply by obtaining the addresses of other participants, which are often available from public servers [46, 61]. Even systems designed to protect the privacy of their users often have open membership policies [14, 29, 40, 58]. This openness enables wide participation, ensuring that a large amount of content is available in file sharing networks, for example, but it also allows attackers to monitor, join, and disrupt peer-to-peer networks [5, 11, 44, 71, 73].

In recent years, pervasive surveillance and censorship of the internet and highly publicised lawsuits against users of file sharing networks have led to increasing interest in private, authenticated communication between friends [19, 24, 25]. In a parallel development, creators of collaborative software have sought to combine the flexibility and autonomy of peer-to-peer networks with the confidentiality and authentication provided by traditional groupware.

Michael Rogers
University College London, London WC1E 6BT, UK, e-mail: m.rogers@cs.ucl.ac.uk

Saleem Bhatti
University of St Andrews, Fife KY16 9SS, UK, e-mail: saleem@cs.st-andrews.ac.uk

X. Shen et al. (eds.), *Handbook of Peer-to-Peer Networking,*
DOI 10.1007/978-0-387-09751-0_28, © Springer Science+Business Media, LLC 2010

This area has so far received relatively little attention from researchers; as with the first wave of peer-to-peer networks in 1999–2001, the developer community has often moved ahead of the academic community, opening up new areas of potential research. Consequently, many of the references in this chapter are to websites rather than to peer-reviewed papers.

The next section defines the scope of this chapter and describes some of the technical challenges faced by private peer-to-peer networks. Section 3 provides a survey of deployed systems, and in Section 4 we classify the systems architecturally and discuss design tradeoffs. Related research is discussed in Sections 5 and 6 concludes the chapter.

2 Background

2.1 Definitions

We define a *private peer-to-peer network* as an internet overlay in which the resources and infrastructure are provided by the users, and new users may only join the network by personal invitation. This definition excludes systems that rely on public servers, such as many online social networks and media sharing websites, but it does not necessarily imply decentralisation – some private peer-to-peer networks use central servers, but access to those servers is restricted to invited users, and the servers are owned and operated by users of the network.

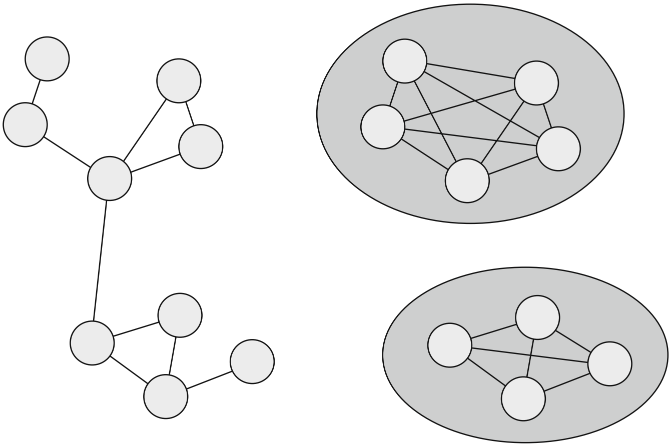

Fig. 1 Friend-to-friend and group-based networks. In a friend-to-friend network (*left*), users only connect directly to people they know. In a group-based network (*right*), any member of a group may connect directly to any other

Some private peer-to-peer networks allow direct connections between any pair of users, while others only allow direct connections between users who know one another. We will refer to the former as *group-based networks* and the latter as *friend-to-friend networks* [8, 34]. The distinction is illustrated in Fig. 1.

2.2 Technical Challenges

Firewalls and network address translators create significant problems for peer-to-peer networks, whether public or private. As the number of internet-connected devices increases, a growing proportion of users are behind "middleboxes" of one kind or another, and many also have dynamic network addresses that change daily or with every session.

Studies of peer-to-peer networks have shown rapid turnover or "churn" in the node population: many nodes are offline at any given time, there are significant daily and weekly cycles in the number of online nodes, and session durations are highly variable, with a few nodes staying online for long periods but most spending only a few hours online in each session [9, 37, 65, 70].

Middleboxes, dynamic network addresses and churn are especially problematic for private networks because of their restricted topologies. In a group-based network, at least one member of the group must run a stable node that is reachable from the public internet; the current addresses of other members can then be learned from that member. In a friend-to-friend network, on the other hand, every user needs at least one friend with a stable, reachable node.

Some systems cope with this problem by using internal or external lookup services for middlebox traversal and address discovery; others require users to exchange updated addresses out-of-band if they cannot reconnect to the network. By monitoring or participating in a lookup service, an attacker may be able to discover who connects to whom, so the decision to use such services involves a tradeoff between privacy and ease of use.

Data integrity, confidentiality and authentication are areas where private peer-to-peer networks may have an advantage over public networks. Because the users know one another, it is feasible for them to exchange cryptographic keys out-of-band; private peer-to-peer networks could even be bootstrapped using existing keys and trust relationships, such as those recorded in the PGP web of trust [10].

"Free riding" is a well-known problem in public peer-to-peer networks, where users often consume the resources provided by others without contributing anything in return [1]. Li and Dabek [43] argue that users of private networks will be more willing to contribute resources than users of public networks, since their contributions will benefit people they know. However, in some private peer-to-peer systems, such as those that support anonymous communication, resources can be consumed by people the provider does not know; free riding could be a problem for such systems.

3 Survey of Deployed Systems

In this section we review a number of private peer-to-peer systems that have been deployed in recent years.

3.1 Group-Based Networks

Groove [35, 72] is a groupware application for creating "shared spaces" that can span organisational boundaries. Each member of the group maintains a copy of the shared space's state, and encrypted updates are transmitted to other members when the state changes. It is not necessary to maintain connections between every pair of members, and indeed firewalls may make this impossible; members who are unable to communicate directly can exchange messages through dedicated relays. Changes to the shared space can be made while members are offline and synchronised when they reconnect.

Two kinds of shared space can be created. In a mutually trusting space, all changes to the state are authenticated using a single key. This makes it possible for members to spoof updates from other members. In a mutually suspicious space, an authentication key is generated for each pair of members, preventing spoofing but increasing the size of update messages.

Members can only join groups by invitation. The inviter is responsible for communicating the new member's encryption and authentication keys to the group, so messages from new members can be spoofed by their inviters, even in mutually suspicious spaces. Any member can evict any other member from a group, which is done by creating a new group key and transmitting it to all members except the evicted member.

N2N [20] is a group-based virtual private network (VPN) that uses relay nodes for address discovery and middlebox traversal.

Shinkuro [66] and PowerFolder [56] support group-based file sharing on local area networks; wide area connections are also possible if at least one member of the group acts as a relay. Each group is associated with a shared directory, and changes are synchronised automatically.

Direct Connect [18, 50] requires one member of each group to run a server, which is used for address discovery, keyword searches and chat.

Octopod [51] avoids the need for central servers by using OpenDHT [60], a public distributed hash table, for address discovery. Each group is associated with a shared directory, and the group owner can grant other users read-only or read-write access by sending them the appropriate keys.

WASTE [23, 74] is a group-based network created by Justin Frankel, the author of Gnutella. Like Gnutella it supports flooded queries and reverse path forwarded replies; these are used to implement keyword searches, file sharing and chat. Nodes can relay one another's messages if all-to-all connectivity is not possible. Links are encrypted and optionally padded to a constant traffic level, but there is no end-to-end encryption or authentication, so users can eavesdrop on one another and spoof messages. There are no group keys, so it is hard to evict users from groups.

Phex [52, 53] is a Gnutella implementation that supports the creation of private networks.

3.2 Friend-to-Friend Networks

Turtle [48, 54] is a friend-to-friend file sharing network designed for censorship resistance. Searches are flooded through the network, search results are forwarded back along the reverse path, and virtual circuits can be established for anonymous file transfer. The virtual circuit architecture is also capable of supporting other applications, including real-time communication.

Turtle uses a novel key agreement protocol in which friends exchange personal questions, the answers to which are assumed to be known to both users but not to eavesdroppers. This avoids the need for out-of-band key exchange, but the strength of the resulting keys will depend on the extent of the eavesdropper's knowledge about the users.

Freenet is a distributed cache where files can be published and retrieved anonymously. Early versions of the software depended on nodes learning addresses from successful queries [14], but Freenet 0.7 uses a new routing algorithm that does not require connections between untrusted nodes [13].

The new algorithm implements a distributed hash table where efficient routing is achieved by changing the nodes' locations in the key space rather than by changing the topology: each node starts at a random location in the key space and uses a stochastic algorithm to swap locations with other nodes until the network converges on a suitable arrangement for efficient routing [63]. Files are stored in the distributed hash table, allowing publishers and readers to remain anonymous.

Freenet stores two kinds of data: content hash keys (CHKs), which are blocks of data identified by their cryptographic hashes, and signed subspace keys (SSKs), which are blocks of data signed with a private key and identified by the hash of the corresponding public key. SSKs can be updated by anyone who knows the private key and retrieved by anyone who knows the public key, which makes it possible to implement a wide range of services over Freenet, including web browsing, message boards and email. An SSK keypair can be derived from a keyword, in which case anyone who knows the keyword can retrieve and update the SSK.

Like Freenet, GNUnet [6] can be configured to connect only to trusted nodes. GNUnet provides content-based and updatable keys and supports keyword searches [36]. Queries are routed using randomised flooding, which might seem to reveal less information about users than Freenet's social network-based routing. However, Kugler [41] describes a statistical attack that makes it possible, given a long series of related requests, to determine whether the requests are likely to have originated at a neighbouring node or to have been forwarded on behalf of another node. Similar attacks might be possible against Freenet even though its routing algorithm is deterministic, because related requests are routed independently [75].

SockeToome [68] enables friend-to-friend file transfers between users with dynamic network addresses, but it is arguably not a peer-to-peer network since no overlay is constructed. Gazzera [32] and Hybrid Share [39] use manually configured connections for friend-to-friend file sharing.

In anoNet [4], virtual private network (VPN) tunnels between friends are connected to form an encrypted overlay. The overlay uses standard internet protocols such as BGP, and even has an internal DNS hierarchy. A reserved network prefix is used to avoid accidentally routing packets between the overlay and the public internet.

Easter [22] uses email as a substrate for friend-to-friend file sharing. This makes it possible to circumvent many firewalls, but the protocol requires frequent polling of email accounts, which might attract the attention of system administrators.

CSpace [16] is a general-purpose friend-to-friend connection service based on a distributed hash table. The connections established by CSpace can be used for any application: file sharing, screen sharing and chat have been implemented so far. Participants in the distributed hash table can observe who connects to whom, which may have implications for privacy.

The Retroshare [59] instant messaging and file sharing network also uses a DHT for address discovery. Users can communicate indirectly through mutual friends, which may allow them to build up trust in one another before requesting direct connections. Galet [31] and Cryptic6 [15] allow friends-of-friends to communicate in a similar way, but they do not use distributed hash tables or relay servers for address discovery, so addresses and encryption keys must be exchanged out-of-band or through mutually trusted friends. Alliance [3] optionally creates direct connections between friends-of-friends, which may help with address discovery, churn and middlebox traversal. Users lose some privacy by revealing their network addresses to friends-of-friends, but this may be preferable to using a third-party lookup service operated by strangers.

Tsnecv [28] is a file sharing system for local area networks in which users may assign different access levels to friends, friends-of-friends and strangers.

3.3 Other Networks

Aimster [2] was the first peer-to-peer system to enable private file sharing between friends. Because of its reliance on centralised public servers, it was quickly shut down [24].

GigaTribe [33] uses a public server to coordinate group-based file sharing and chat, with an optional commercial relay service for users who are unable to connect directly.

Sneakernet [67] uses small data-carrying devices such as mobile phones and memory sticks to pass information between friends. A gossip-based protocol allows encrypted messages to travel over multiple hops between a trusted public server and anonymous users.

Many other systems use websites or other public servers to coordinate peer-to-peer communication between friends or in private groups, and recently some centralised instant messaging networks have also begun to support peer-to-peer voice and video connections. We consider such systems to be outside the scope of this chapter, however, because of their reliance on public servers.

4 Architecture

The private peer-to-peer networks described in the previous section can be classified architecturally along four axes: scale, visibility, centralisation, and application support. Table 1 summarizes existing research works based on these axes.

1. Scale – does the system consist of isolated local networks or a single global network?

The issue of scale raises difficult technical and social tradeoffs. Large private networks are likely to face many of the same connectivity challenges as public peer-to-peer networks, including heterogeneity and churn. Because of their size, they are also more likely to attract the attention of eavesdroppers and attackers. Users may feel less of an obligation to contribute to strangers than friends, so free riding may also be an issue for large networks. On the other hand, the wider range of people and resources can make large networks more attractive to potential users [45].

Deployed systems deal with these tradeoffs in a variety of ways. At one end of the axis is WASTE, which is designed for small groups of mutually trusting users; users can belong to more than one network, but traffic does not pass between networks. The group size is limited in practice by the protocol's use of flooding and the difficulty of evicting misbehaving users, which encourages users to be cautious about giving invitations.

At the other end of the axis is Freenet 0.7, which is designed to be a "globally scalable darknet" [12]. Freenet's routing algorithm is based on the assumption that all users belong to a single small-world social network, and it may not be possible to merge mature networks without seriously disrupting routing. It is difficult to grow a global network from a single seed, however: not all potential users know a user of the main network, so recent versions of the software optionally create connections between strangers in addition to manually configured friend-to-friend connections [64].

GNUnet and Turtle take an intermediate approach: messages can be forwarded anonymously across the friend-to-friend overlay, and local networks created by small groups of users can be merged by establishing friend-to-friend connections between them. However, GNUnet and Turtle both rely on flooding, which does not perform well in large networks; thus even in a merged network, communication may effectively be confined to local regions of the overlay. Indirect communication in Galet, Cryptic6 and Alliance is explicitly local: friends-of-friends can communicate pseudonymously, which may allow users to build up trust in new friends before connecting to them directly.

The ability to merge mature networks could be important for the growth of private peer-to-peer systems, because it may be easier for a potential user to find friends who are interested in setting up a local network than to contact and befriend a member of an existing network.

2. Visibility – can users connect to everyone in the network, or only to their friends? Who else can see that they are participating?

The issue of visibility separates group-based networks from friend-to-friend networks. This distinction becomes more important as networks grow, because any user may invite a friend who does not know all the other users. In a group-based network, the newly invited user will be able to connect to any existing user; thus group-based networks become less private as they grow, whereas friend-to-friend networks can (at least in theory) remain private at any scale. Indirect pseudonymous communication through mutual friends represents an intermediate position between group-based and strictly friend-to-friend visibility.

Group-based networks could be vulnerable to Sybil attacks [21], where an attacker uses multiple identities simultaneously, and whitewashing [30], where an attacker changes identities to escape the consequences of past misbehaviour. For example, it is easy to imagine an attacker automatically “inviting” new identities into a group more quickly than the other users can manually evict them. Such attacks can be prevented by requiring a certain fraction of existing members to approve each invitation [57], but this approach does not scale to large groups where not all users know one another.

Friend-to-friend connections are not a panacea for identity-related attacks, but they guarantee that every identity within a local scope belongs to a different individual, which prevents whitewashing; it might also be possible to use the structure of social networks to limit the impact of Sybil attacks (see Section 5).

Regardless of whether the network is group-based or friend-to-friend, users may need to make additional connections to discover one another’s addresses. Some networks use public distributed hash tables for address discovery, while others use external servers for NAT and firewall traversal [27, 38, 62]. Unfortunately, involving third parties in communication can have implications for autonomy and privacy: if participants in a public DHT or the operators of a public server can identify the users of a private network, and perhaps even observe which users connect to which others, many of the benefits of using a private network will have been lost.

It may be possible to avoid relying on external lookup services if some members of the network have reliable nodes with stable network addresses. In a group-based network, only one member of the group needs a stable address, but in friend-to-friend network, every user needs at least one friend with a stable address.

Table 1 The architecture of deployed private peer-to-peer networks

Name	Scale	Visibility	Centralisation	Application support
Direct Connect	Local	Group	Central server	File sharing, chat
N2N	Local	Group	Dedicated relays	VPN
Groove	Local	Group	Dedicated relays	Shared workspace
PowerFolder	Local	Group	Members may run relays	Shared workspace
Shinkuro	Local	Group	Members may run relays	Shared workspace
WASTE	Local	Group	Members may run relays	File sharing, chat
Phex	Local	Group	Members may run relays	File sharing
Easter	Local	Friends	Email servers	File sharing
Gazzera	Local	Friends	Decentralised	File sharing
Hybrid Share	Local	Friends	Decentralised	File sharing
Tsnecv	Local	Configurable	Decentralised	File sharing
Alliance	Flexible	Friends-of-friends	Decentralised	File sharing, chat
Cryptic6	Flexible	Friends-of-friends	Decentralised	File sharing, chat
Galet	Flexible	Friends-of-friends	Decentralised	File sharing, chat
Turtle	Flexible	Friends	Decentralised	Virtual circuits
GNUnet	Flexible	Friends	Decentralised	Distributed cache
Freenet 0.7	Global	Friends	Decentralised	Distributed cache
anoNet	Global	Friends	Decentralised	VPN
CSpace	Global	Friends, DHT	Decentralised (DHT)	Virtual circuits
Retroshare	Global	Friends-of-friends, DHT	Decentralised (DHT)	File sharing, chat
Octopod	Global	Group, DHT	Decentralised (DHT)	File sharing

Connections between friends-of-friends may help to mitigate this problem by giving each node a larger set of neighbours.

If a user cannot contact any previously known nodes when reconnecting to the network, it may be necessary to exchange updated addresses with a friend out-of-band, so the use of external address discovery services involves a tradeoff between privacy and ease of use.

3. Centralisation – Does the Network Rely on a Central Server?

A number of public peer-to-peer networks have been shut down by attacking their central servers [24], leading to the perception that centralised designs are fragile and should be avoided. However, the risks may be different in private networks, where servers can be more or less hidden from untrusted parties. Centralisation can make it easier to manage identities, exchange cryptographic keys, and learn the current addresses of other users, provided all users trust the operator of the server.

Direct Connect requires a central server or "hub" for every group. Many other group-based networks can operate without servers if all users are on the same local area network, but require at least one member to act as a relay for wide area communication. Groove and N2N use dedicated relays that can see who is communicating but cannot decrypt the messages they forward. Easter relies on email servers, which are decentralised but may not be controlled by users of the network. Most other friend-to-friend systems rely on manual port forwarding or hole punching to traverse NATs and firewalls [27, 38, 62]. Friend-to-friend connections can be lost if both friends change their addresses at the same time, and it may be necessary to exchange updated addresses through an alternative channel such as email.

Octopod, Retroshare and CSpace use distributed hash tables for address discovery, which may allow untrusted parties to observe which users connect to which others. Freenet avoids this problem because its DHT implementation only uses existing friend-to-friend connections; users can publish their encrypted contact details under updatable keys for their friends to retrieve anonymously.

4. Application support – what functionality does the network provide?

The systems described in Section 3 are designed for a wide range of purposes, from business collaboration to resisting censorship. Some focus on supporting a specific use case, while others aim to provide a general-purpose communication layer.

N2N and anoNet are the most flexible systems, creating virtual private networks that can be used by a wide range of existing software. Turtle and CSpace provide general-purpose virtual circuits, but existing applications cannot use them without modification. Similarly, Freenet and GNUnet provide general-purpose anonymous storage layers, but they cannot be used transparently by standard software.

Groove, PowerFolder and Shinkuro create shared workspaces with automatic synchronisation, which is useful for collaborative projects but may not be ideal for sharing large collections of files. Groove is also integrated with the Microsoft Office suite, enabling other applications to make use of its private connections.

Most private peer-to-peer networks simply provide a graphical user interface for file sharing and chat and do not attempt to integrate with other applications.

The disadvantage of restricting a private network to a single application is that users must go to the trouble of constructing a new overlay whenever they wish to use a new application; general-purpose networks avoid this duplication of effort. On the other hand, by blurring the line between local and wide area networks, and by supporting existing applications that may not have been designed with security in mind, general-purpose networks may inadvertently undermine their users' security or privacy.

5 Related Research

In a seminal paper that predates most of the systems described in this chapter, Biddle et al. [7] predict that as public peer-to-peer networks come under attack from copyright holders, users will turn to sharing content with their friends through “the darknet” – not a single global network, but rather a patchwork of local networks, technologically isolated but interconnected through their users. “In light of strong cryptography,” Biddle *et al.* argue, “it is hard to imagine how sharing could be observed and prosecuted as long as users do not share with strangers.”

Pouwelse et al. [55] identify five research challenges for peer-to-peer systems: decentralising functionality; maintaining availability in the face of churn; ensuring the integrity of data and metadata; creating incentives to contribute resources; and achieving network transparency across middleboxes. They suggest that all five challenges can be addressed by encouraging users to form cooperative social groups.

A number of researchers have proposed using explicit or implicit information about the social connections between users to improve the robustness of peer-to-peer networks [17, 42, 47]. SybilGuard [77] and SybilLimit [76] use the structure of social networks to limit the impact of Sybil attacks on open systems.

Figueiredo et al. [26] describe how to establish VPN connections between users of social networking websites to create “social VPNs”; the websites act as trusted third parties for key exchange. Nagaraja [49] proposes an anonymous communication system that uses a social networking website for key distribution.

Strufe and Reschke [69] describe a peer-to-peer overlay that stores group information as well as file information, so users can create authenticated groups for file sharing and instant messaging. Each group has a single owner who is responsible for adding and removing members.

These systems are not private peer-to-peer networks according to the definition used in this chapter, but they demonstrate some of the advantages of incorporating the social relationships between users into the design of communication networks.

6 Conclusions

This chapter has provided a brief survey of the emerging field of private peer-to-peer networks, which attempt to combine the flexibility and autonomy of peer-to-peer architectures with the confidentiality and authentication of traditional groupware. Deployed systems can be classified along four architectural axes: scale, visibility, centralisation, and application support. Each of these axes involves tradeoffs affecting the robustness, scalability, privacy, and ease of use that the resulting systems can provide.

Private peer-to-peer networks are already being used in fields as diverse as business collaboration, file sharing, social networking, grassroots political activity and censorship-resistant communication. Considering the varied and sometimes conflicting requirements raised by these applications, we do not expect that any single network will be able to meet the needs of all users; instead we will continue to see a range of architectures that are adapted to particular uses.

References

1. Adar, E., Huberman, B.: Free riding on Gnutella. First Monday **5**(10) (2000). URL http://firstmonday.org/issues/issue5_10/adar/
2. URL http://web.archive.org/web/20010801151157/aimster.com/. Aimster website, archived August 2001, available from http://web.archive.org/web/20010801151157/aimster.com/
3. URL http://www.alliancep2p.com/. Alliance website, http://www.alliancep2p.com/
4. URL http://anonet.org/. AnoNet website, http://anonet.org/
5. Banerjee, A., Faloutsos, M., Bhuyan, L.: P2P: Is Big Brother watching you? Tech. Rep. UCR-CS-2006-06201, Department of Computer Science and Engineering, University of California, Riverside (2006). URL http://www1.cs.ucr.edu/store/techreports/UCR-CS-2006-06201.pdf
6. Bennett, K., Grothoff, C.: GAP - practical anonymous networking. In: Proceedings of the 3rd International Workshop on Privacy Enhancing Technologies (PET 2003), Dresden, Germany, *Lecture Notes in Computer Science*, vol. 2760, pp. 141–160 (2003). URL http://gnunet.org/download/aff.ps
7. Biddle, P., England, P., Peinado, M., Willman, B.: The darknet and the future of content protection. In: Proceedings of the 2nd International Workshop on Digital Rights Management (DRM 2002), Washington, DC, USA, *Lecture Notes in Computer Science*, vol. 2696, pp. 155–176 (2003). URL http://msl1.mit.edu/ESD10/docs/darknet5.pdf
8. Bricklin, D.: Friend-to-friend networks (2000). URL http://www.bricklin.com/f2f.htm. Available from http://www.bricklin.com/f2f.htm
9. Bustamante, F., Qiao, Y.: Friendships that last: Peer lifespan and its role in P2P protocols. In: 8th International Workshop on Web Content Caching and Distribution, Hawthorne, NY, USA (2003). URL http://2003.iwcw.org/papers/bustamante.pdf
10. Cederlöf, J.: Web of trust statistics and pathfinder. URL http://www.lysator.liu.se/~jc/wotsap/. Available from http://www.lysator.liu.se/~jc/wotsap/
11. Christin, N., Weigend, A., Chuang, J.: Content availability, pollution and poisoning in file sharing peer-to-peer networks. In: ACM Conference on Electronic Commerce, Vancouver, Canada (2005). URL http://www.weigend.com/ChristinWeigendChuang2005.pdf
12. Clarke, I.: Project status update, and request for your help (2005). URL http://archives.freenetproject.org/message/20050914.103042.4b8aac35.en.html. Available from http://archives.freenetproject.org/message/20050914.103042.4b8aac35.en.html
13. Clarke, I., Sandberg, O.: Routing in the dark: Scalable searches in dark P2P networks. In: DefCon 13, Las Vegas, NV, USA (2005). URL http://www.math.chalmers.se/~ossa/defcon13/vegas1_print.pdf
14. Clarke, I., Sandberg, O., Wiley, B., Hong, T.: Freenet: A distributed anonymous information storage and retrieval system. In: Proceedings of the International Workshop on Design Issues in Anonymity and Unobservability, Berkeley, CA, USA, *Lecture Notes in Computer Science*, vol. 2009, pp. 46–66 (2001). URL http://www.cl.cam.ac.uk/~twh25/academic/papers/icsi-revised.pdf
15. URL http://cryptic6.sourceforge.net/. Cryptic6 website, http://cryptic6.sourceforge.net/
16. URL http://www.cspace.in/. CSpace website, http://www.cspace.in/
17. Danezis, G., Lesniewski-Laas, C., Kaashoek, M., Anderson, R.: Sybil-resistant DHT routing. In: 10th European Symposium on Research in Computer Security (ESORICS 2005), Milan, Italy (2005). URL http://www.cl.cam.ac.uk/users/gd216/sybildht.pdf
18. URL http://dcplusplus.sourceforge.net/. DC++ website, http://dcplusplus.sourceforge.net/

19. Deibert, R., Palfrey, J., Rohozinski, R., Zittrain, J. (eds.): Access Denied: The Practice and Policy of Global Internet Filtering. MIT Press (2008)
20. Deri, L., Andrews, R.: N2N: A layer two peer-to-peer VPN. In: Proceedings of the 2nd International Conference on Autonomous Infrastructure, Management and Security (AIMS 2008), Bremen, Germany, *Lecture Notes in Computer Science*, vol. 5127, pp. 53–64 (2008). URL http://luca.ntop.org/n2n.pdf
21. Douceur, J.: The Sybil attack. In: Proceedings of the 1st International Workshop on Peer-to-Peer Systems (IPTPS '02), Cambridge, MA, USA, *Lecture Notes in Computer Science*, vol. 2429, pp. 251–260 (2002). URL http://www.cs.rice.edu/Conferences/IPTPS02/101.pdf
22. URL http://easta.sourceforge.net/. Easter website, http://easta.sourceforge.net/
23. Ek, M., Hultin, F., Lindblom, J.: WASTE peer-to-peer protocol (2005). URL http://prdownloads.sourceforge.net/j-waste/waste_documentation-1.1.pdf?download. Reverse-engineered protocol documentation, available from http://prdownloads.sourceforge.net/j-waste/waste_documentation-1.1.pdf?download
24. Electronic Frontier Foundation: RIAA v. the people: Four years later (2007). URL http://w2.eff.org/IP/P2P/riaa_at_four.pdf. Available from http://w2.eff.org/IP/P2P/riaa_at_four.pdf
25. Electronic Privacy Information Center, Privacy International: Privacy and Human Rights 2006: An International Survey of Privacy Laws and Developments. Washington, DC: Electronic Privacy Information Center (2007). URL http://www.privacyinternational.org/article.shtml?cmd[347]=x-347-559458
26. Figueiredo, R., Boykin, O., St. Juste, P., Wolinsky, D.: Social VPNs: Integrating overlay and social networks for seamless P2P networking. In: Workshop on Collaborative Peer-to-Peer Systems (COPS), Rome, Italy (2008). URL http://ast-deim.urv.cat/wwiki/images/1/18/Socialvpn-cops08.pdf
27. Ford, B., Srisuresh, P., Kegel, D.: Peer-to-peer communication across network address translators. In: USENIX Annual Technical Conference, Anaheim, CA, USA (2005). URL http://www.usenix.org/events/usenix05/tech/general/full_papers/ford/ford.pdf
28. Fredriksen, L.: Securing private peer-to-peer networks. Master's thesis, University of Tromsø(2007). URL http://www.ub.uit.no/munin/handle/10037/1197
29. Freedman, M., Morris, R.: Tarzan: A peer-to-peer anonymizing network layer. In: 9th ACM Conference on Computer and Communications Security (CCS 2002), Washington, DC, USA (2002). URL http://pdos.csail.mit.edu/tarzan/docs/tarzan-ccs02.pdf
30. Friedman, E., Resnick, P.: The social cost of cheap pseudonyms. Journal of Economics and Management Strategy **10**(2), 173–199 (2001). URL http://www.si.umich.edu/~presnick/papers/identifiers/081199.pdf
31. URL http://galet.sourceforge.net/. Galet website, http://galet.sourceforge.net/
32. URL http://www.lugato.net/gazzera/. Gazzera website, http://www.lugato.net/gazzera/
33. URL http://www.gigatribe.com/. GigaTribe website, http://www.gigatribe.com/
34. Gonze, L.: Friendnet (2002). URL http://www.oreillynet.com/pub/wlg/2428. Available from http://www.oreillynet.com/pub/wlg/2428
35. URL http://www.groove.net/. Groove Networks website, http://www.groove.net/
36. Grothoff, C., Grothoff, K., Horozov, T., Lindgren, J.: An encoding for censorship-resistant sharing (2005). URL http://gnunet.org/download/ecrs.ps. GNUnet white paper, available from http://gnunet.org/download/ecrs.ps

37. Guha, S., Daswani, N., Jain, R.: An experimental study of the Skype peer-to-peer VoIP system. In: Proceedings of the 5th International Workshop on Peer-to-Peer Systems (IPTPS '06), Santa Barbara, CA, USA, pp. 1–6 (2006). URL http://saikat.guha.cc/pub/iptps06-skype.pdf
38. Guha, S., Francis, P.: Characterization and measurement of TCP traversal through NATs and firewalls. In: Internet Measurement Conference (IMC 2005), Berkeley, CA, USA (2005)
39. URL http://hybrid-share.sourceforge.net/. Hybrid Share website, http://hybrid-share.sourceforge.net/
40. URL http://www.i2p2.de/. I2P website, http://www.i2p2.de/
41. Kugler, D.: An analysis of GNUnet and the implications for anonymous, censorship-resistant networks. In: Proceedings of the 3rd International Workshop on Privacy Enhancing Technologies (PET 2003), Dresden, Germany, *Lecture Notes in Computer Science*, vol. 2760, pp. 161–176 (2003). URL http://gnunet.org/papers/GNUnet_pet.pdf
42. Lesniewski-Laas, C.: A Sybil-proof one-hop DHT. In: 1st International Workshop on Social Network Systems (SocialNets 2008), Glasgow, Scotland (2008). URL http://pdos.csail.mit.edu/papers/sybil-dht-socialnets08.pdf
43. Li, J., Dabek, F.: F2F: Reliable storage in open networks. In: 5th International Workshop on Peer-to-Peer Systems (IPTPS '06), Santa Barbara, CA, USA (2006). URL http://pdos.csail.mit.edu/~jinyang/pub/iptps-f2f.pdf
44. Liang, J., Kumar, R., Xi, Y., Ross, K.: Pollution in P2P file sharing systems. In: IEEE Infocom, Miami, FL, USA (2005). URL http://cis.poly.edu/~ross/papers/pollution.pdf
45. Liebowitz, S., Margolis, S.: Network externalities (effects). In: New Palgrave Dictionary of Economics and the Law, MacMillan (1998). URL http://wwwpub.utdallas.edu/~liebowit/palgrave/network.html
46. Lua, E., Crowcroft, J., Pias, M., Sharma, R., Lim, S.: A survey and comparison of peer-to-peer overlay network schemes. IEEE Communications Surveys and Tutorials **7**(2) (2005). URL http://www.cl.cam.ac.uk/users/ekl25/lua.pdf
47. Marti, S., Ganesan, P., Garcia-Molina, H.: SPROUT: P2P routing with social networks. In: Proceedings of the 9th International Conference on Extending Database Technology (EDBT 2004), Heraklion, Crete, Greece, *Lecture Notes in Computer Science*, vol. 3268, pp. 425–435 (2004). URL http://home.comcast.net/~sergiomarti/papers/2004-5.pdf
48. Matějka, P.: Security in peer-to-peer networks. Master's thesis, Department of Software Engineering, Charles University, Prague (2004). URL http://www.turtle4privacy.org/documents/masterThesis.pdf
49. Nagaraja, S.: Anonymity in the wild: Mixes on unstructured networks. In: Proceedings of the 7th Workshop on Privacy Enhancing Technologies (PET 2007), Ottawa, Canada, pp. 254–272 (2007). URL http://www.cl.cam.ac.uk/~sn275/papers/unstructured-mixes.pdf
50. URL http://web.archive.org/web/20050627012020/www.neo-modus.com/. NeoModus Direct Connect website, archived June 2005, available from http://web.archive.org/web/20050627012020/www.neo-modus.com/
51. URL http://sysnet.ucsd.edu/octopod/. Octopod website, http://sysnet.ucsd.edu/octopod/
52. URL http://www.phex.org/. Phex website, http://www.phex.org/
53. Phex wiki: Creating a private network. URL http://www.phex.org/wiki/index.php/Creating_a_private_Network. Available from http://www.phex.org/wiki/index.php/Creating_a_private_Network
54. Popescu, B., Crispo, B., Tanenbaum, A.: Safe and private data sharing with Turtle: Friends team-up and beat the system. In: 12th International Workshop on Security Protocols, Cambridge, UK (2004). URL http://www.cs.vu.nl/~bpopescu/papers/sec_prot04/sec_prot04.pdf

55. Pouwelse, J., Garbacki, P., Wang, J., Bakker, A., Yang, J., Iosup, A., Epema, D., Reinders, M., van Steen, M., Sips, H.: Tribler: A social-based peer-to-peer system. In: 5th International Workshop on Peer-to-Peer Systems (IPTPS '06), Santa Barbara, CA, USA (2006). URL http://iptps06.cs.ucsb.edu/papers/Pouw-Tribler06.pdf
56. URL http://www.powerfolder.com/. PowerFolder website, http://www.powerfolder.com/
57. Quercia, D., Hailes, S., Capra, L.: TATA: Towards anonymous trusted authentication. In: Proceedings of the 4th International Conference on Trust Management (iTrust 2006), Pisa, Italy, pp. 313–323 (2006). URL http://www.cs.ucl.ac.uk/staff/D.Quercia/publications/querciaTATA06.pdf
58. Rennhard, M., Plattner, B.: Practical anonymity for the masses with MorphMix. In: Proceedings of the 8th International Financial Cryptography Conference (FC 2004), Key West, FL, USA, *Lecture Notes in Computer Science*, vol. 3110, pp. 233–250 (2004). URL http://www.tik.ee.ethz.ch/~rennhard/publications/FC2004.pdf
59. URL http://retroshare.sourceforge.net/. Retroshare website, http://retroshare.sourceforge.net/
60. Rhea, S., Godfrey, B., Karp, B., Kubiatowicz, J., Ratnasamy, S., Shenker, S., Stoica, I., Yu, H.: OpenDHT: A public DHT service and its uses. In: SIGCOMM 2005, Philadelphia, PA, USA (2005). URL http://berkeley.intel-research.net/sylvia/f230-rhea.pdf
61. Risson, J., Moors, T.: Survey of research towards robust peer-to-peer networks: Search methods. Tech. Rep. UNSW-EE-P2P-1-1, University of New South Wales (2004). URL http://uluru.ee.unsw.edu.au/~john/tr-unsw-ee-p2p-1-1.pdf
62. Rosenberg, J., Weinberger, J., Huitema, C., Mahy, R.: RFC 3489: STUN - simple traversal of user datagram protocol (UDP) through network address translators (NATs) (2003). URL http://www.ietf.org/rfc/rfc3489.txt
63. Sandberg, O.: Distributed routing in small-world networks. In: 8th Workshop on Algorithm Engineering and Experiments (ALENEX06), Miami, FL, USA (2006). URL http://www.math.chalmers.se/~ossa/swroute.pdf
64. Sandberg, O., Clarke, I.: The evolution of navigable small-world networks. Tech. Rep. 2007:14, Department of Computer Science and Engineering, Chalmers University of Technology (2007). URL http://www.math.chalmers.se/~ossa/evolution.pdf
65. Saroiu, S., Gummadi, P.K., Gribble, S.: A measurement study of peer-to-peer file sharing systems. In: Multimedia Computing and Networking (MMCN '02) (2002). URL http://www.cs.washington.edu/homes/gribble/papers/mmcn.pdf
66. URL http://shinkuro.com/. Shinkuro website, http://shinkuro.com/
67. URL http://code.google.com/p/sneakernet/. Sneakernet website, http://code.google.com/p/sneakernet/
68. URL http://www.ziggy.speedhost.com/bdsock.html. SockeToome website, http://www.ziggy.speedhost.com/bdsock.html
69. Strufe, T., Reschke, D.: Efficient content distribution in semi-decentralized peer-to-peer networks. In: Proceedings of the 8th International Netties Conference, Ilmenau, Germany, pp. 33–38 (2002). URL http://eris.prakinf.tu-ilmenau.de/pub/papers/strufe02efficient.pdf
70. Stutzbach, D., Rejaie, R.: Towards a better understanding of churn in peer-to-peer networks. Tech. Rep. UO-CIS-TR-04-06, Department of Computer Science, University of Oregon (2004). URL http://www.barsoom.org/~agthorr/papers/tr04-06.pdf
71. Svensson, P.: Comcast blocks some internet traffic. Associated Press (2007). URL http://www.msnbc.msn.com/id/21376597/
72. Udell, J., Asthagiri, N., Tuvell, W.: Security. In: A. Oram (ed.) Peer-to-Peer: Harnessing the Power of Disruptive Technologies, chap. 18. O'Reilly (2001). This chapter describes Groove.
73. Veiga, A.: Music labels tap downloading networks. Associated Press (2003). URL http://www.usatoday.com/tech/webguide/music/2003-11-14-sharestats_x.htm

74. URL http://waste.sourceforge.net/. WASTE website, http://waste.sourceforge.net/
75. Wright, M., Adler, M., Levine, B., Shields, C.: An analysis of the degradation of anonymous protocols. In: ISOC Symposium on Network and Distributed System Security, San Diego, CA, USA (2002). URL http://www.freehaven.net/anonbib/cache/wright02.pdf
76. Yu, H., Gibbons, P., Kaminsky, M., Xiao, F.: SybilLimit: A near-optimal social network defense against Sybil attacks. In: IEEE Symposium on Security and Privacy, Oakland, CA, USA (2008). URL http://www.comp.nus.edu.sg/~yuhf/yuh-sybillimit.pdf
77. Yu, H., Kaminsky, M., Gibbons, P., Flaxman, A.: SybilGuard: Defending against Sybil attacks via social networks. In: SIGCOMM 2006, Pisa, Italy (2006). URL http://sigcomm06.stanford.edu/discussion/getpaper.php?paper_id=26