Structured Peer-to-Peer Overlay Networks: Ideal Botnets Command and Control Infrastructures?

Carlton R. Davis[1], Stephen Neville[2], José M. Fernandez[1], Jean-Marc Robert[3], and John McHugh[4]

[1] École Polytechnique de Montréal
{carlton.davis,jose.fernandez}@polymtl.ca
[2] University of Victoria
sneville@ece.uvic.ca
[3] École de technologie supérieure
jean-marc.robert@etsmtl.ca
[4] Dalhousie University
mchugh@cs.dal.ca

Abstract. Botnets, in particular the Storm botnet, have been garnering much attention as vehicles for Internet crime. Storm uses a modified version of Overnet, a structured peer-to-peer (P2P) overlay network protocol, to build its command and control (C&C) infrastructure. In this study, we use simulation to determine whether there are any significant advantages or disadvantages to employing structured P2P overlay networks for botnet C&C, in comparison to using unstructured P2P networks or other complex network models. First, we identify some key measures to assess the C&C performance of such infrastructures, and employ these measures to evaluate Overnet, Gnutella (a popular, unstructured P2P overlay network), the Erdős-Rényi random graph model and the Barabási-Albert scale-free network model. Further, we consider the three following disinfection strategies: a) a *random* strategy that, with effort, can remove randomly selected bots and uses no knowledge of the C&C infrastructure, b) a *tree-like* strategy where local information obtained from a disinfected bot (e.g. its peer list) is used to more precisely disinfect new machines, and c) a *global* strategy, where global information such as the degree of connectivity of bots within the C&C infrastructure, is used to target bots whose disinfection will have maximum impact. Our study reveals that while Overnet is less robust to random node failures or disinfections than the other infrastructures modelled, it outperforms them in terms of resilience against the targeted disinfection strategies introduced above. In that sense, Storm designers seem to have made a prudent choice! This work underlines the need to better understand how P2P networks are used, and can be used, within the botnet context, with this domain being quite distinct from their more commonplace usages.

1 Introduction

Botnets have emerged as one of the most pressing security issues facing Internet users [1,2,3]. In early 2007, researchers estimated that 11 percent of the more

S. Jajodia, and J. Lopez (Eds.): ESORICS 2008, LNCS 5283, pp. 461–480, 2008.

than 650 million computers attached to the Internet were conscripted as bots [3]. Members of the security research community have tracked botnets with sizes ranging from several hundred to 350 thousand federated hosts [1,2,4,5].

Botnets are big business; whether they be used for sending spam [6], or as tools for profit-motivated on-line crime [7]. As computer users become more aware of security issues, and vulnerabilities are more quickly fixed via automatic updates, more sophisticated social engineering techniques are being used to install malicious codes on victims' machines. One of the commonly used techniques for planting bot codes on machines, involves spam emails with enticing subjects (such as "Britney Did it Again") with links to Web sites containing malicious codes. Electronic greeting cards and "free" downloads have also been used to trick users into clicking on links containing exploit codes which are subsequently installed on the unsuspecting victims' machines, thus transforming them into bots [8].

Once infected, the bots must be controlled by the external malicious agents. This can be achieved by a command and control (C&C) infrastructure, ideally allowing the distribution of any command to any bot. This infrastructure has three competing goals: a) to be as efficient as possible, by ensuring the rapid propagation of commands, b) to be as stealthy as possible, by minimising the risk that the botnet's activities will be observed, and c) to be as resilient as possible, *i.e.* to minimise the impact of node disinfection or node failure. In this work, we refer to *robustness* as the network's capacity to retain its capabilities in light of random failures or uninformed disinfection strategies, while we use the term *resilience* to refer to a network's capacity to retain its capabilities when subject to targeted and informed disinfection strategies.

Prior to late 2006, most observed botnets used Internet Relay Chat (IRC) [9] as a communication protocol for C&C [10]. Awareness of this fact spurred researchers to develop botnet detection schemes which are based on analysis of IRC traffic [11,12,13,14,15]. This, in turn, likely pushed the development of more sophisticated botnets, such as Storm and Nugache [16] and Peacomm [17], towards the utilisation of P2P networks for their C&C infrastructures. In response to this trend, researchers [4,18] have proposed various models of botnets that are based on self-organised complex networks or P2P infrastructures, as possibilities for advanced botnets C&C infrastructures.

The Storm botnet is one of the largest and better known recent botnets. It adapted the Overnet P2P file-sharing application [19] —itself based on the Kademlia distributed hash table algorithm [20]— and utilises it for its C&C infrastructure [21]. Storm has received much scrutiny in the electronic media [1,2,3], and in the anti-virus research community [8,16,21]. Such attention has spurred the Storm operators to episodically evolve the details of how Storm operates, for example, by encrypting the C&C traffic [22]. The level of sophistication Storm exhibits —for instance, by using Fast Flux service networks [23] for DNS services, and launching distributed denial-of-service attacks on computers that are used to investigate its bots [24]— indicates that its operators are quite savvy. Consequently, it is conceivable that they are likely to continue to enhance their botnets to make them less detectable and more resilient to disinfection, whether this be through their own discoveries or through leveraging relevant research results available within the literature.

A botnet can be seen as a complex network, with hundred of thousands of nodes, each representing a bot. While direct communications between any two bots are possible using the Internet Protocol (IP), in practice meaningful communications between bots can only happen if one of them knows about the fact that the other computer is indeed a bot and what parameters (e.g. open listening sockets, cryptographic keys) are needed to contact it. Thus, edges of this (directed) graph correspond to communication links where the source node knows of and how to contact the destination node.

These freely self-organised networks can be described by different theoretical models: Erdős-Rényi random graphs, Barabási-Albert scale-free graphs, or Watts-Strogatz small-world network models. The efficiency of their underlying C&C infrastructures depends, at least in part, on the intrinsic properties of the underlying graphs. It is well established in the research literature that the Erdős-Rényi random graph model [25] shows more resilience to targeted removal of nodes than the other well-known, theoretical network models, *i.e.*, the Barabási-Albert scale-free [26] and the Watts-Strogatz [27] small-world networks, whilst keeping the same underlying properties of the graph (*i.e.* size and connectivity). It is intuitively clear that removing the highly connected nodes from scale-free graphs may easily impact the connectivity of those graphs. In light of these results, it is natural to ask what advantage, if any, a botnet which employs the theoretical Erdős-Rényi random graph or Barabási-Albert scale-free network model would have, compared to botnets utilising structured or unstructured P2P networks, such as Gnutella or Overnet. This question is doubly relevant. First, because in the research on botnet C&C performance to date, little attention has been paid to the actual methods employed by current botnets to build these C&C infrastructures. Second, because if the real-world use of these theoretical models could yield better C&C performance, it would provide us with an indication of likely future evolution in the botnet arms race.

Our findings and the main contributions of our work can be summarised as follows:

1. We introduce and discuss three key measures for assessing the performance of botnets command and control; two of these measures, to the best of our knowledge, have not been previously explored in the context of botnets.
2. We introduce and consider the effects of three distinct disinfection strategies, on a structured (Overnet) and an unstructured (Gnutella) P2P overlay networks, and on the Erdős-Rényi random graph and Barabási-Albert scale-free network models.
3. Most significantly, we show how botnets using a structured P2P networks (Overnet) as their C&C infrastructures can achieve even more resistance to targeted attacks than that achievable through the Erdős-Rényi random graph model, already known to show good resilience.

4. Finally, our results indicate that there is an apparent general trade-off between the efficiency of the C&C infrastructure to distribute commands, and its resilience to disinfection.

The rest of the paper is organised as follows. Section 2 lists the related works and provides an outline of how our work differs from previous works. Section 3 contains background information about four network architectures we investigated as possible infrastructures for botnets C & C infrastructures. Section 4 contains information relating to the simulation setup, a discussion of the developed measures, and some of the initial assessment results. In Section 5, we describe the disinfection strategies we considered, and present the disinfection analysis results. In the final section, we discuss our findings, summarise our contributions and suggest some directions for future work.

2 Related Work

Theoretical models of complex networks have received significant attention in the Physics literature. This research has looked carefully at the properties of these graphs, as nodes are removed randomly or in a deliberate and targeted fashion.

Albert, Jeong and Barabási [28] investigated the error and attack tolerance of complex network using simulation. They studied the change in diameter of Erdős-Rényi (ER) random graph [25] and Barabási-Albert (BA) scale-free network models [26] when small fraction of nodes were removed. Their results indicated that BA model shows high degree of tolerance against random error (high robustness), but that it is more susceptible to be disconnected than ER model when the most connected nodes are targeted (low resilience).

Crucitti, Lattora, Marchiori and Rapisarda [29] conducted similar studies which compared the resilience of ER and BA networks against targeted attacks. Instead of using changes in diameter as a measurement of robustness and resilience, the authors used the global efficiency, which is defined as the average of the efficiency $\varepsilon_{ij} = 1/t_{ij}$ over all couple of nodes; where t_{ij} is the time it takes to send a unit packet of information through the fastest path. Their studies showed that ER random graphs exhibit similar tolerance with respect to error and targeted attacks, while the BA scale-free network model is robust to random errors, but vulnerable to targeted attacks.

Holme, Kim, Yoon and Han [30] studied the response of complex networks subjected to attacks on nodes and edges. They investigated the changes in average shortest path length and the size of the giant component of ER, BA and Watts-Strogatz (WS) [27] graphs when a fraction of the nodes are removed. In the simulation experiments, nodes of the graphs were selected and removed in decreasing order of their incidence degree and their betweenness centrality measure. This latter value captures the notion of whether a given node is on most of the shortest paths between any pairs of nodes in the graph. The authors concluded from their study that the ER model, because of its lack of structural bias, is the most resilient network of the set they tested.

The theoretical models of complex networks have also been considered in the botnet literature. Cooke, Jahanian and McPherson [10] investigated possible advanced botnet communication topologies. They outlined three topologies (a centralised structure, a generic P2P model and a simplistic random model) without comparing their effectiveness, and suggested possible detection methods based on the correlation of events gathered by distributed sensors. To their credit, the authors forecasted the appearance of botnets like Storm using P2P networks.

Wang, Sparks and Zou [18] presented the design of an advanced hybrid P2P botnet and provided analysis and simulation results which attest to the resilience of their botnet architecture. Their theoretical P2P protocol is very simple compared to Kademlia, used by Overnet, and gives graphs with weak structures. The authors look essentially at only one measure to evaluate the performance of their approach: the connectivity of the resulting graph after targeted disinfection. Furthermore, they did not compare their protocol with any other complex network model.

Dagon, Gu, Lee and Lee [4] identified three measures to measure the performance of the C&C infrastructure. First is the size of the giant component of the graph, which represents the size of the reachable (and thus usable) portion of the botnet. Then they consider the graph diameter, which measures the efficiency of the botnet in terms of rapidity to reach all nodes in the connected component. The last measure is the graph redundancy, measuring the probability that, if two edges of the graph share a node, they are part of a triangle, and is related to the robustness of the botnet. The authors considered the following four network models: Erdös-Rényi random graphs, Barabási-Albert scale-free networks, Watts-Strogatz small world networks. They also consider P2P models, but approximate them with the theoretical models: structured P2P models approximated as ER graphs, and unstructured P2P models approximated as BA networks (we describe more precisely this distinction in Section 3).

Our work can be differentiated from the works listed above, as follows:

- None of this previous work investigated the performance differences between structured and unstructured P2P networks, and that between P2P networks and theoretical complex network models.
- Two of the three measures that we identified for assessing the performance of botnets (*i.e.* reachability from a given node and the distribution of the shortest paths) have not been explored in any of the previous works.
- We describe and analyse a disinfection strategy (tree-like disinfection), which has not been considered in previous work.

3 Background

In this section, we give a brief overview of the four network models we studied as C&C infrastructures for botnets. We commence with P2P overlay networks.

P2P overlay networks are generally classified into two categories: structured and unstructured networks. The nodes in a *structured P2P network* connect to at most k peers, where k is a fixed parameter; and there are stipulations regarding the identities of nodes to which a given node can connect. For the case of Overnet, a node can only connect to nodes which have IDs that are less than a certain distance (see Section 3.1 below). Whereas, for *unstructured P2P networks*, there is no fixed limit to the number of peers that a node may connect to and, more importantly, there is no stipulation regarding the identity of which nodes a given node is allowed connections with. Examples of structured P2P networks are Overnet [19] and Chord [31]. Gnutella [32] and Freenet [33] are examples of unstructured P2P networks. We choose Overnet and Gnutella for our simulation studies because they are the more real-world popular examples of their respective network types. Brief overviews of both are provided below along with brief descriptions of Erdős-Rényi random graphs and Barabási-Albert scale-free models of complex networks.

3.1 Brief Overview of Overnet

Overnet is a popular file sharing overlay network which implements a distributed hash table (DHT) algorithm called Kademlia [20]. Each node participating in an Overnet network generates a 128-bit ID when it first joins the network. The ID is transmitted with every message the node sends. This permits recipients of messages to identify the sender's existence as necessary. Each node in an Overnet network stores contact information about each other in order to route query messages. Every node keeps a separate list of $\langle \text{IP address}, \text{UDP port}, \text{ID} \rangle$ triplets for nodes of distance 2^i and 2^{i+1} from itself, for each $0 < i < 128$. The distance $d(x, y)$ between two IDs x and y is defined as the bitwise exclusive or (XOR) of x and y interpreted as an integer, *i.e.*, $d(x, y) = x \oplus y$. These peer lists are referred to as k-buckets and they are kept sorted by time last seen, ordered by least-recently seen at the head and the most recently-seen at the tail.

A node n wishing to join an Overnet network must have contact with some node m already participating in the network. The new node n inserts its contact m into the appropriate k-bucket then broadcasts node lookup query messages to search for the k closest nodes to its ID through the node m. The new node n can then populate its k-buckets based on messages it receives. In the process, seeing the broadcast messages from n, other nodes can also refresh their k-buckets and insert n in their k-buckets as necessary.

3.2 Brief Overview of Gnutella

Gnutella is a popular unstructured file sharing overlay network. In order to join a Gnutella network, a node n connects to a node m that is already connected to the network. Once attached to the network, n broadcasts a PING message through m to announce its presence. When a node receives a PING message, it forwards it to its neighbours and sends a PONG message to the sender of the PING message along the reverse path of the PING message. The transmission of these messages allows nodes to learn about each other. A new node n typically connects to the first k nodes it hears from, where k is a configurable parameter.

3.3 Brief Description of Erdős-Rényi and Barabási-Albert Models

Erdős-Rényi (ER) random graph model: An ER graph [25] (also described at length in [34]) is a random graph consisting of N nodes connected by edges. Each of the $\binom{N}{2}$ edges is chosen independently with probability p. The ER model depicts a random network with no particular structural bias.

Barabási-Albert (BA) scale-free model: The BA scale-free model [26] more closely approximates real-world complex networks, for example, the World Wide Web, biological networks and social networks. In these networks, the probability that a node connects with k other nodes is roughly proportional to $k^{-\gamma}$, for some constant γ (thence, they are also referred to as *power-lay graphs*). Therefore, it is more likely to observe few highly connected hubs, although most nodes are connected to few other nodes. Barabási and Albert provided a simple methodology for constructing such graphs based on a growth process which uses preferential attachment. Starting with a small number nodes, at every time step add a new node that is more likely to connect to nodes with higher incidence degree. The resulted graph (or network) shows a power-law degree distribution $P(k) \backsim k^{-\gamma}$, where $\gamma = 2.9 \pm 0.1$.

4 Simulation Setup and Results

For our simulation analysis, we constructed sets of random graphs using the four models described in the previous section. Each graph $G = (V, E)$ —where V is the set of nodes and E is the set of edges— has $|V| = 25{,}000$ nodes. Relevant details regarding each graph types are outlined below. The tested networks were implemented in the C programming language with the igraph C library [35], used to support the implementation of the simulations. For our analyses, we performed 20 simulation runs, each with a different set of graphs, and the presented results are the averages obtained across the composite of these runs.

Overnet graphs: We simulated an Overnet network which grows from an initial set of 2 nodes to 25,000 nodes. Each node in the network has k-buckets with a total of at most 20 peers, *i.e.* $k = 20$. We modelled this network as sets of random undirected graphs. Each graph having 25,000 nodes and maximum degree of 20, the maximum number of edges is $|E| = 25,000 * 20/2 = 250,000$. In fact, for the Overnet graphs we generated for the simulation analysis, the average number of edges is 221,137, corresponding to an average degree of 17.69.

Gnutella graphs: We simulated a Gnutella network which starts with an initial node set of 2 nodes and grows to 25,000 nodes. In the simulation implementation, we placed no limits on the number of peers that a node may connect to; however, the number of connections that any given node can initiate was limited to 9. This restriction allows the number of edges in the Gnutella graph to approximate that of the Overnet graph, since the expected average overall degree should be $18 = 2 * 9$. We modelled the simulated Gnutella network as sets of random undirected graphs; each graph has 25,000 nodes and the set of 20 Gnutella graphs has an average of 224,427 edges, with an average degree of 17.95.

Erdős-Rényi (ER) random graphs: An ER random graph can be represented as $\mathcal{G}(n, p)$ where n is the number of nodes and p is the probability that an edge—drawn from the edge set with $\binom{n}{2}$ edges—is present. We utilised the igraph C library to generate 20 undirected ER graphs with $n = 25,000$ and $p = 0.000708$. The average number of edges for the set of 20 ER graphs is $\binom{25000}{2} * 0.000708$, i.e., $\frac{25000*24999}{2} * 0.000708 = 221,241$, *i.e.* an average degree of 17.71, where this value for p was intentionally selected to approximate the connectivity of the tested Gnutella and Overnet networks.

Barabási-Albert (BA) scale-free graphs: We utilised the igraph C library to generate 20 undirected BA graphs for our simulation. Each graph has 25,000 nodes, and each node has a maximum of 9 outward connections, which for similar reasons as for Gnutella networks should yield a similar number of edges. In fact, the average number of edges for the set of 20 BA graphs is 224,991, corresponding to an average degree of 17.99.

4.1 Degree Distribution of the Graphs

Figure 1 shows the degree distribution of the four graphs we discussed above. The standard deviation for the histogram values (number of nodes having a given degree) ranges from 0 to 14.5% of the calculated mean values.

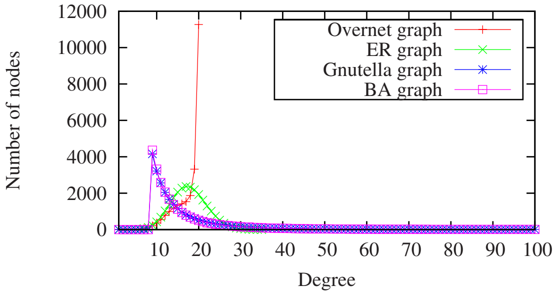


Fig. 1. Degree distribution of Overnet, ER, Gnutella and BA graphs

It is readily apparent from this figure that the Gnutella graph is very similar to the BA graph. This supports the findings of previous works [36,37] which indicate that Gnutella networks exhibit similar power-law properties as BA scale-free networks. The degree distribution for the ER graph is a binomial distribution, as expected. The use of the DHT algorithm in Overnet has the effect or randomly selecting nodes in the network, which is almost equivalent to the construction of the ER graph, and hence the head of their respective degree distributions is somewhat similar. The key difference between these two models is that, since in Overnet there is a maximum degree limit of 20, the tail of what would be otherwise a binomial distribution is "bunched up" at degree values 19 and 20.

4.2 Performance Measures

We identified three key performance measures for assessing the effectiveness of a botnet. Only the diameter of the graph has been previously used in this context. We present them below:

Reachability from a given node: With a decentralised C&C infrastructure, a botnet operator can issue commands to the botnet from any node within the botnet. A key measure, therefore, of the effectiveness of the botnet, is the number of nodes that can be reached within a given distance from a node x. Let $\mathit{\Gamma}_k(x)$ denotes; where the set of nodes at distance k from a node x in a graph $G = (V, E)$.

$$\mathit{\Gamma}_k(x) = \{y \in V : d(x, y) = k\},$$

where $d(x, y)$ represents the length, *i.e.*, number of hops, of the shortest path between node x and y. Let $N_k(x)$ represents the set of nodes at distance *at most* k from x.

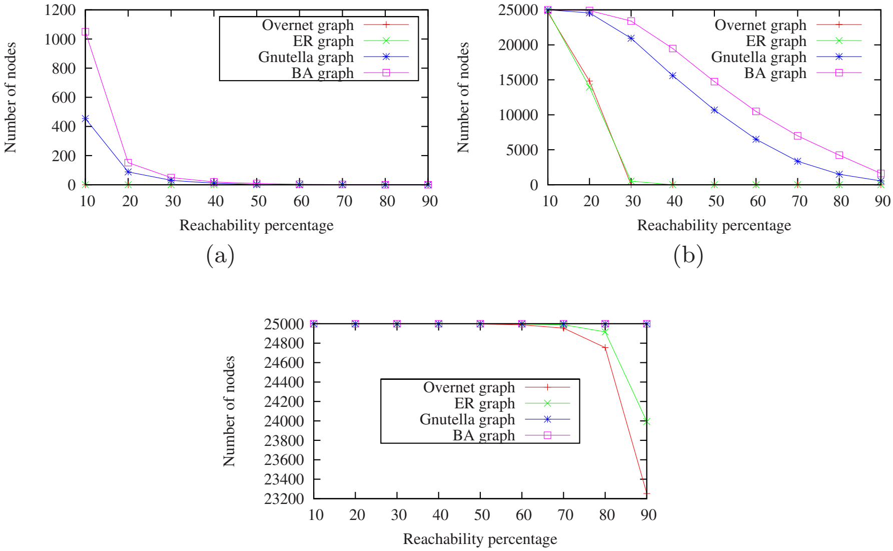
$$N_k(x) = \bigcup_{i=0}^{k} \mathit{\Gamma}_i(x)$$

$N_k(x)$ with high cardinality for small k's is more advantageous for botnet operators. The higher the cardinality of $N_k(x)$, the better the botnet will perform, since, a larger percentage of nodes will be reachable within k hops from any given node.

Figure 2 shows the histogram for reachability percentages, rounded up to nearest 10%, *i.e.* $\lceil N_k(x)/25{,}000 * 100\rceil$) for $k = 1, 2, 3$, respectively, for the four models considered. The standard deviation for these histogram values ranged from 0 to 16.4% of the calculated mean values over the 20 graphs generated. For example, Figure 2(a) in particular, indicates that none of the 25,000 nodes in either the Overnet or ER networks we simulated, are able to reach even 10 percent of the nodes in the botnet within 1 hop. On the other hand, Figures 2(b) and 2(c) indicate that of the four graph types, BA graphs have the highest reachability within 2 and 3 hops, respectively, followed by Gnutella, ER and Overnet graphs. This is likely due to the fact that the BA graphs have the largest number of highly connected nodes, followed by Gnutella, ER and Overnet graphs. The presence of highly connected nodes creates the opportunity for shorter paths between the origin x and its target nodes, and hence increase the size of $N_k(x)$. The difference in the number of such nodes for the four graph types is readily observable in Figure 1, except for the case of BA and Gnutella which appear very similar from the plot. A more detailed analysis of the raw data used to generate Figure 1 indicates, however, that the BA graphs achieve a slightly larger number of highly connected nodes.

Shortest path length sets: Let $d(u, v)$ represents the length of the shortest path between u and v, where $u, v \in V$, for a graph $G = (V, E)$. Let $\mathcal{L}_l(u, v)$ denote the set of all node pairs (u, v), such that, $d(u, v) = l$, *i.e.*,

$$\mathcal{L}_l(u, v) = \{(u, v) : u, v \in V \wedge d(u, v) = l\}$$



(a)

(b)

(c)

Fig. 2. Reachability histogram for k hops, with (a) $k = 1$, (b) $k = 2$, and (c) $k = 3$

A network with sets $\mathcal{L}_l(u, v)$ of high cardinality for small values of l is more advantageous for botnet operators, since this allows messages to reach intended recipients in fewer hops. One may ask, why would a botnet operator care about the number hops a message must traverse in order to reach its recipient? Since in today's Internet, each hop involves no more than milliseconds or at worst a few seconds, a few more hops probably do not significantly affect the speed of propagation of botnet commands. However, each extra hop required to reach a given fraction of the network, will result in approximately a 9- or 18-fold increase in the number of messages (since in our case, the average outdegree is either 9 or 18, depending on the network model). Thus, since the overall network "footprint" of the C&C infrastructure increases exponentially with the number of hops, reachability within a given number of hops or equivalently the number of hops required to achieve a given portion of the network are very significant measures in terms of stealth. Botnets with $\mathcal{L}_l(u, v)$ with higher cardinality for small l, will likely operate with greater degree of stealth than those with $\mathcal{L}_l(u, v)$ with lower cardinality for small l.

Figure 3 shows the simulation results for the $\mathcal{L}_l(u, v)$ cardinalities for the four graph types we tested. The standard deviation for these cardinalities was within 0.8% and 24% of the calculated mean values over the 20 graphs generated. The results indicate that for $l < 4$, $\mathcal{L}_l(u, v)$ has higher cardinality for BA, followed by Gnutella, ER and Overnet graphs; $|\mathcal{L}_l(u, v)|$ for ER is only slightly higher than that of Overnet graph for $l < 3$. Whereas for $l \geq 4$, the order for $|\mathcal{L}_l(u, v)|$ is reversed; being Overnet, followed by ER, Gnutella and BA. These results, again can be attributed to the fact that BA graphs have higher number of highly

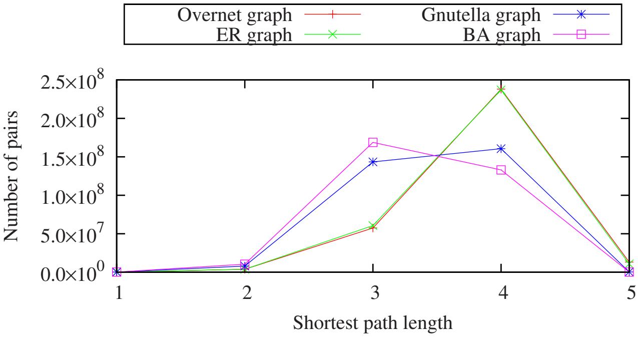


Fig. 3. Shortest path lengths results, indicating cardinalities of $|L_l(u, v)|$, on the y-axis, for various path lengths l, on the x-axis

Table 1. Diameter of the network graphs

Graph	*Diameter*
Overnet	6
ER	6
Gnutella	5
BA	5

connected nodes than Gnutella, ER and Overnet graphs; similarly, Gnutella graphs have higher number of highly connected nodes than ER and Overnet, and so on.

Diameter of the network graph. The diameter, $\mathrm{diam}(G)$, of a graph $G = (V, E)$ is the length of the longest shortest path separating any two nodes. Thus, it can be defined as $\mathrm{diam}(G) = \max_{u,v} d(u, v)$, where $d(u, v)$ is the length of the shortest path between u and v. Botnets with smaller diameter are desirable for botnet operators, since this allows messages to traverse fewer nodes before reaching their intended recipients, and this has non-negligible impact in terms of stealth, as previously discussed. This measure has been used previously by Dagon, Gu, Lee and Lee [4]. Table 1 shows the diameter of the four network we simulated. Once again, the diameters of the four network graphs are very similar, with ER and Overnet being only slightly worse.

5 Disinfection Analysis

The disinfection of bot code from infected machines can be modelled as the removal of nodes (and incident edges) from the graph $G = (V, E)$ representing the underlying C&C infrastructure. Let $A = \{n_1, n_2, .., n_j\}$ be the nodes corresponding to the disinfected bots (removed from the botnet C&C infrastructure) and let $\bar{G} = (\bar{V}, \bar{E})$, with $\bar{V} = V - A$, denote the new underlying graph of the C&C infrastructure. The effectiveness of the disinfection strategy can be characterised by the decrease of $|\bar{V}|$ and $|\bar{E}|$.

5.1 Disinfection Strategies

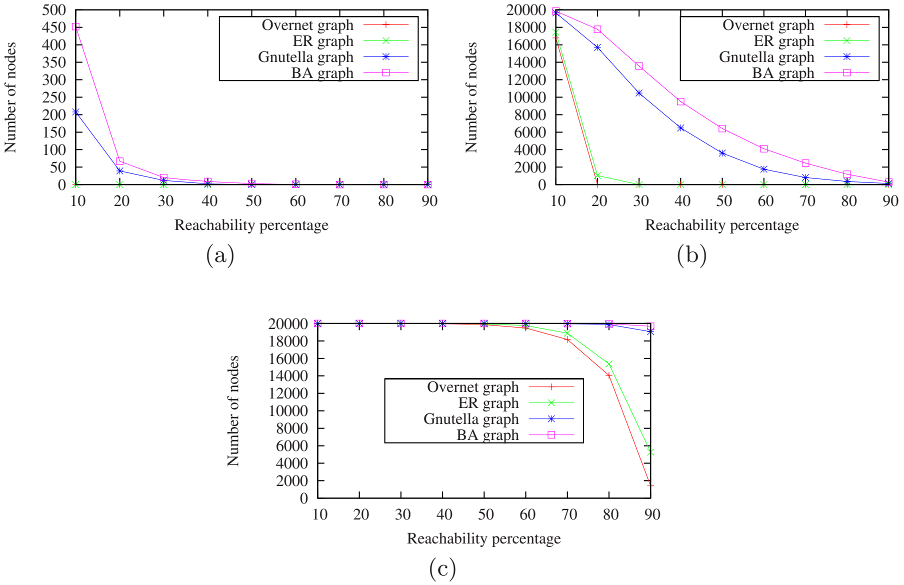
For our simulation analysis, we consider three disinfection strategies, as described below.

Random disinfection. The focus here is just to disinfect bots as they are discovered, without attempt to gain insight in the overall C&C infrastructure of the botnet. This strategy is equivalent to the occurrence of random errors in the botnet, *i.e.* random removal of nodes from G. This disinfection approach models a user or system administrator discovering and successfully removing the bot code from the machine, while making no attempt to acquire or use any information gleaned from the bot to aid in the rolling-up the overall botnet.

Tree-like disinfection. When bots are discovered, information about their peer lists (peers they are connected to) can be gleaned from analysing their communication traffic or by reverse engineering the bot code. A peer list can then be used to identify other bots, and the other bots peer lists, in turn can be used to discover other bots, and so on.

Global information-based disinfection. The aim of this approach is to acquire information about a botnet C & C infrastructure within an allowed time period, then use the information to prioritise the bots in terms of the order with which they should be disinfected. This approach divides time into discrete time windows Δt_i's. All bots discovered within a given time slot Δt_i are considered as an ordered set A_i whose elements are ordered according to their assessed disinfection priorities. At the end of Δt_i, the elements of A_i are disinfected according to their order in the set. The bots in the sets A_i's can be ordered in decreasing order of the degrees of the given bots within the botnet C&C infrastructure. Bots with the same degree are ordered according to the order they were discovered. This approach models, for example, a large-scale ISP or large private- or public-sector organisation observing a given botnet, active within its confines, and then using the gained information to inflict maximal damage on the botnet, as facilitated by having local bot discovery processes forward what they learn to a centralised analysis process, which then selects the most appropriate disinfection approach.

This latter disinfection mechanism requires a lot of global information which may be hard to gather across different administrative domains. However, as mentioned in the review of the literature, Cooke, Jahania, McPherson [10] already suggested possible detection methods based on the correlation of events gathered by distributed sensors. In any case, this global information approach is useful since it should represent the optimal strategy against which any disinfection strategy should be compared.



(a) (b)

(c)

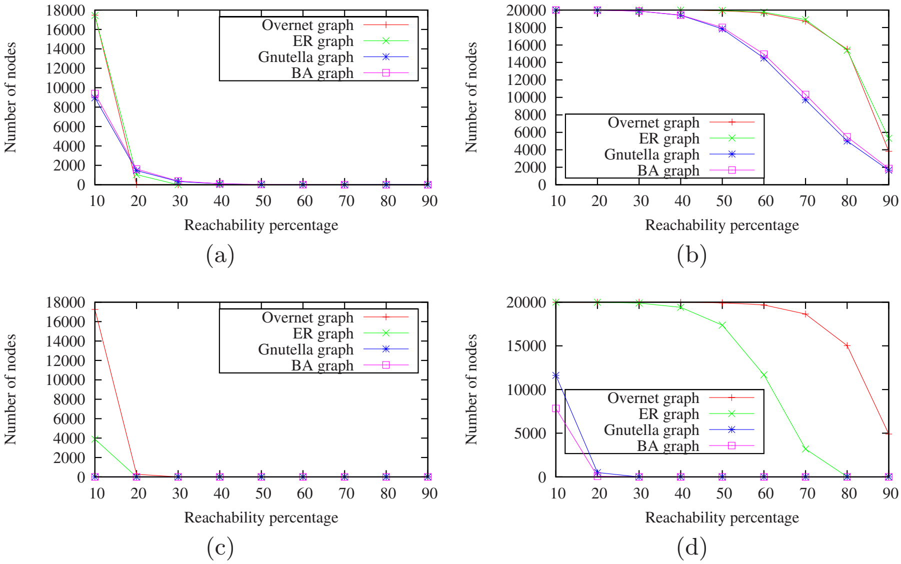
Fig. 4. Random disinfection reachability histograms for k hops after 20% of the nodes are removed, for (a) $k = 1$, (b) $k = 2$, and (c) $k = 3$

5.2 Disinfection Analysis Results

Figure 4 shows the reachability results for $k = 1, 2, 3$, for the four graph types after 20% of the nodes were removed randomly. The standard deviation for the histogram values range from 0 to 4.8% of the calculated mean values over the 20 graphs generated for each network type.

The random disinfection results of Figure 4 show the same trends as those of Figure 2. Additionally, comparison of Figures 2(b) with 4(b) and 2(c) with 4(c) indicates that the removal of a fixed percentage of nodes have greater effect on ER and Overnet graphs. For example, Figures 2(b) and 4(b) show that when 20% of the nodes are randomly removed from ER and Overnet graphs, the number of nodes with 20% reachability fell from 15,000 to approximately 1,500, *i.e.*, a decrease of 90%. Whereas for ER and Overnet graphs, the number of nodes with 20% reachability fell from approximately 25,000 to 18,000 for BA and 16,000 for Gnutella, *i.e.*, a decrease of 28% and 36%, respectively. This supports previous results [28,29,30] indicating that BA graphs are more resilient to random errors (random removal of nodes) than ER graphs. In essence, since both Gnutella and BA graphs exhibit only a few nodes of very high degree, there is a low probability that a given random removal will remove such a node. Hence, reachability is preserved since it is highly probable that all other nodes have a short path to one of these highly connected nodes.

Similarly to the scenario for the random removal strategy, Figures 5 and 6 provide, respectively, the results for reachability and shortest path length sets



(a)

(b)

(c)

(d)

Fig. 5. Reachability histograms for $k = 2, 3$ hops after tree-like disinfection, (a) and (b), and global information-based disinfection, (c) and (d), respectively

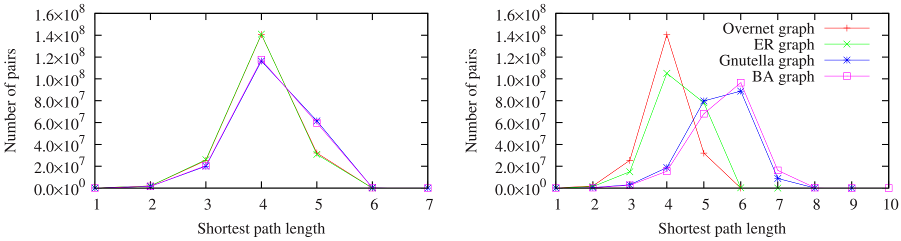


Fig. 6. Shortest path lengths results after 20% of the nodes removed via (a) tree-like disinfection, and (b) global information-based disinfection

cardinalities for the other two directed disinfection strategies, after the same 20% portion of the nodes have been disinfected.

In the case of the tree-like disinfection, comparison of Figures 5(a) with 2(b), 5(b) with 2(c), and 6(a) with 3, shows that Overnet and ER graphs exhibit greater degree of resilience to disinfection than Gnutella and BA graphs. For example, Figures 6(a) and 3 reveal that when 20% of the nodes are removed from the graphs via tree-like disinfection, the number of pairs with the length of the shortest path equal to 3, decreases from approximately 1.75×10^8 for BA and 1.25×10^8 for Gnutella to approximately 2.0×10^7 for both; a decrease of over 88% for BA and 84% for Gnutella graphs. Whereas for Overnet and ER graphs, the decrease is from approximately 5.0×10^7 to 2.0×10^7, *i.e.*, a decrease of 60%.

Table 2. Disinfection data: fraction of nodes removed vs. diameter

	Random				**Tree-like**				**Global info.**			
f	*ON*	*ER*	*GN*	*BA*	*ON*	*ER*	*GN*	*BA*	*ON*	*ER*	*GN*	*BA*
0	6	6	5	5	6	6	5	5	6	6	5	5
0.1	6	6	5	5	6	6	6	6	6	6	8	∞
0.2	6	6	6	6	6	6	7	7	6	7	∞	∞
0.3	7	7	6	6	6	7	∞	∞	7	∞	∞	∞
0.4	7	7	∞	∞	7	7	∞	∞	∞	∞	∞	∞

Of course, from the perspective of a graph intended to malicious use, tree-like disinfection can be viewed as a measure of the ease with which the network could be rolled-up based on iteratively exploiting local connectivity knowledge.

For global information-based disinfection, comparison of Figures 2(b) with 5(c), 2(c) with 5(d), and 3 with 6(b) reveal the most interesting results: Overnet graphs exhibit much greater resilience to global information-based disinfection than ER graphs. For example, a look at Figures 2(b) and 5(c) shows that when 20% of the nodes of the graphs are removed via global information-based disinfection, the number of nodes that have 10% reachability for $k = 2$, decreases from all 25,000 nodes for both Overnet and ER graphs, to approximately 17,500 for Overnet and 4,000 for ER. Obviously, this is a key design consideration if one is seeking to construct P2P networks to support malicious activities under the expectation that the defensive community will be activity engaged in cooperatively trying to disable the network.

The results from diameter analysis also confirm this trend. Table 2 indicates that for global information-based disinfection, the ER graphs became disconnected when 20% of the nodes were removed; whereas, for Overnet graphs, 30% of the nodes had to be removed for the graphs to become disconnected. It is important to notice that diameter changes are not gradual, but instead occur at sharp thresholds (see Table 2). This is much akin to the previously known [34,38] sharp transitions in connectivity in ER random graph processes, where edges are added one at a time with the given probability p. It should be noted that for the disinfection analysis via the diameter measure, the mode of data sets for the 20 simulation runs, instead of their mean, was computed to support the requirement to include graph disconnection, as represented by ∞ in Table 2.

Finally, Figure 7 tells the most compelling story of all. Even though, all results so far indicate that Overnet is the most resilient botnet C&C structure, the comparison of the effect of the various disinfection strategies highlights the need for further research efforts to develop effective mitigation schemes. Whereas global information-disinfection strategy has much more dramatic effects on BA, Gnutella and ER graphs, than on the Overnet Graph,there is essentially very little difference between the three disinfection strategies for Overnet. In other words, the significant extra effort necessary to implement the most complex disinfection strategies only pays off against the less resilient types of network, but not against Overnet. Against Overnet, for the same percentage of nodes

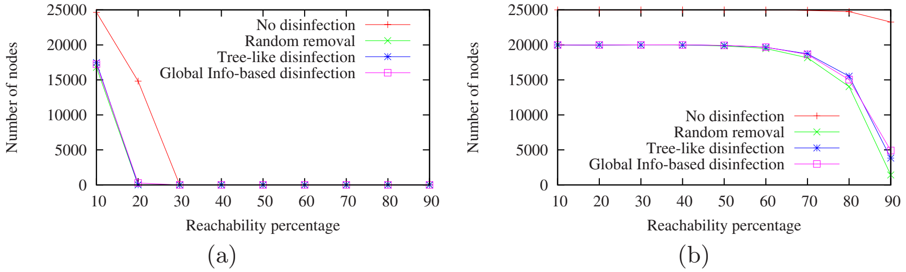


Fig. 7. The effect of the three disinfection strategies on Overnet, for (a) $k = 2$ and (b) $k = 3$, after removal of 20% of the nodes

removed, the simpler random removal strategy is equally effective (or inneffective) as the more complex tree-like or global information-based strategies. This suggests the need for further research geared to develop more efficient botnet mitigation schemes against Overnet-type C&C infrastructures.

6 Discussion

This work began from the general research supposition that Storm was unlikely to have arrived at its use of Overnet by happenstance. Instead, it was more likely that Overnet provided an available solution that well-served the intrinsic needs created when one tries to run large-scale botnets to service malicious activities. Through the analysis above, it has been shown Overnet indeed provides a solution which allows stealthiness and resilience to be traded-off against efficiency. In effect, of the networks tested, Overnet provides the best solution for a P2P network designed to support malicious activities within an environment within which the P2P network itself will be under attack at the cost of only relatively mild losses in efficiency. No claim is made that Overnet represents the ultimate solution for malicious botnet design, merely that as the current step along the evolutionary path it appears to be a fairly good solution from the context of engineering design, assuming one of the key design criteria is botnet longevity.

In parallel, the question was explored as to whether the available formal graph-theoretic models, *i.e.*, Erdős-Rényi random graphs and Barabási-Albert scale-free networks, would better serve the botnet operators' needs. From the research perspective, the applicability of such models would have the distinct advantage that at-scale network behaviours would, in the worst-case, depending on the parameter of interest, be asymptotically computable; hence, side-stepping the need for at-scale simulation studies. Two interesting results were observed via this comparison. The non-maliciously used P2P solutions, namely Gnutella, did follow relatively closely the Barabási-Albert scale-free network model, at least with respect to the tested measures. Hence, it would not be unreasonable to model such networks as Barabási-Albert networks.

The behaviour of Overnet, on the other hand, although closest to Erdős-Rényi random graphs, was not well modelled as an Erdős-Rényi graph and, in fact, significantly surpassed their performance with respect to tree-like and global information-based disinfection. These disinfection approaches, in particular, model the defender iteratively attempting to roll-up the botnet; hence, Overnet's success may help to explain why, in part, its real-world disinfection has presented a challenge. In essence, Overnet is the most diffuse and least-tree like of all of the tested networks, where each node contains (or exposes once discovered) the least information about the botnet's overall structure. Whereas, efficiency pushes the network solution toward a much more tree-like structure, ideally with the trunk of the tree being the high capacity nodes, but this entails creating a network which is easily rolled-up or disconnected.

The above questions were explored through three newly introduced measures in this context, namely: reachability, shortest path sets, and diameter. It was shown that together these measures provided a quantitative mechanism to explore what appears to be an innate trade-off of network efficiency versus its stealth and resilience. In particular, these measures allow some insight to the design of concern when constructing P2P networks to service malicious activities and, hence, expected to exist and operate while themselves under direct and continual threat. No claim is made that the proposed measures are in and of themselves either complete or sufficient. It is fully expected that other measures exist which are equally important in exploring and understanding the design considerations of botnet C&C. The proposed measures do, however, expose issues which have not been previously addressed.

6.1 Conclusions

The conclusions of this work can be succinctly stated as follows:

1. A general trade-off of network efficiency versus stealthiness and resilience exists and allows the operators of malicious botnets to sacrifice a modicum of efficiency to achieve significant gains in likely botnet longevity.
2. The developed measures of reachability, shortest path sets, and diameter when combined provide an effective mechanism to explore the nature of such trade-offs.
3. It appears that non-maliciously used P2P networks, *i.e.*, Gnutella, can likely be well modelled via existing graph-theoretic models, *i.e.*, Barabási-Albert networks, whereas malicious botnets, *i.e.*,Overnet, cannot; this implies a need to either augment the theory models to include Overnet-like behaviours, a seemingly difficult task due to the hard peer-list thresholding done within individual nodes, or the need to turn to simulation-based studies to explore the at-scale behaviours of such botnets.
4. If one was building a botnet to service malicious activities then Overnet would appear to provide a strong solution to a number of the engineering challenges faced when the deployment environment is assumed to be hostile, where this is irrespective of the mechanisms by which Storm's actual operators may have arrived at this solution.
5. Overnet, due to its quite diffuse structure, shows the particular troubling behaviour of a very slow degradation in its capabilities, as nodes are removed

in a tree-like fashion using the local peer list information, with disconnection only occurring suddenly once one has already removed more than 40% of the network's nodes.

6.2 Future Work

Obviously, this work, by the nature of the approach applied, has focused solely on the issues and measures which can be assessed through static graph analysis. A number of interesting and important issues exists with respect to how the proposed network models actually behave within real networks. For example, as discussed above, stealthiness is a critical issue if the botnet is to achieve longevity. Achieving stealthiness is, at least in part, related to a) ensuring that network hot spots do not arise due to intra-botnet communications, and b) reducing the message footprint by keeping short intra-botnet path lengths. Additionally, a key concern is gaining an understanding of just how quickly a given command can be propagate through the actual botnet, or more generally, the time frame require to ensure that M machines of the botnet's available N machines have been recruited to serve a particular need, *i.e.*, spam generation, a DDoS attack, network probing activities, *etc.*. Exploring such issue requires simulating such botnets at-scale, given the likelihood of emergent behaviours, inclusive of the actual network traffic they generate. We are moving forward with developing such simulations. Within this context, we are also beginning to look at whether more effective and practical approaches to counter a Storm-like botnet may exist and what these may entail. Obviously, it is unlikely that Storm-like botnets represent an evolutionary end-point of malicious botnets; gaining an understanding of how such networks can be tuned and designed to survive disinfection approaches is important to improving our ability to effectively counter such networks. It is unclear whether disinfection and mitigation approaches developed under small-scale system analysis will translate effectively to large-scales systems, *i.e.*, into the botnet-scales already seen in real-world. Hence, an area we are exploring is the analysis and characterisation of the emergent behaviours which are exhibited by P2P networks and, more generally, botnets as they scale, as well as the development of effective at-scale disinfection strategies. Finally, there is of course the need to explore how on-going birth and death processes effect measured network behaviours and capabilities.

References

1. CNN Technology News: Expert: Botnets no. 1 emerging Internet threat (January 2006), `www.cnn.com/2006/TECH/internet/01/31/furst/`
2. Washington Post Technology news: The botnet trackers (February 2006), `www.washingtonpost.com/wp-dyn/content/article/2006/02/16/AR2006021601388.html`
3. New York Times Technology news: Attack of the zombie computers is growing threat (January 2007), `www.nytimes.com/2007/01/07/technology/07net.html`
4. Dagon, D., Gu, G., Lee, C., Lee, W.: A taxonomy of botnets. In: Proc. Computer Security Applications Conference (ACSAC), December 2007, pp. 325–339 (2007)

5. Vogt, R., Aycock, J., Jacobson Jr., M.J.: Army of botnets. In: Proc. 14^{th} Annual Network and Distributed System Security Symposium (NDSS) (March 2007)
6. Ramachandran, A., Feamster, N.: Understanding the network-level behavior of spammers. In: Proc. Conference on Applications, technologies, architectures, and protocols for computer communications (October 2006)
7. Lanelli, N., Hackworth, A.: Botnets as a vehicle for online crime (December 2005), www.cert.org/archive/pdf/Botnets.pdf
8. Bureau, P.M., Lee, A.: Malware storms: a global climate change. Virus Bulletin (November 2007), http://www.virusbtn.com
9. Oikarinen, J., Reed, D.: Internet relay chat protocol. Request for Comments (RFC 1459) (May 1993)
10. Cooke, E., Jahanian, F., McPherson, D.: The zombie roundup: Understanding, detecting, and disrupting botnets. In: Proc. 1^{st} Conference on Steps to Reducing Unwanted Traffic on the Internet (SRUTI) (July 2005)
11. Barford, P., Yegneswaran, V.: An inside look at botnets. Advances in Information Security 27, 171–191 (2007)
12. Binkley, J.R., Singh, S.: An algorithm for anomaly-based botnet detection. In: Proc. 2^{nd} Conference on Steps to Reducing Unwanted Traffic on the Internet (SRUTI) (July 2006)
13. Strayer, W.T., Walsh, R., Livadas, C., Lapsley, D.: Detecting botnets with tight command and control. In: Proc. 31^{st} IEEE Conference on Local Computer Networks (November 2006)
14. Rajab, M.A., Zarfoss, J., Monrose, F., Terzis, A.: A multifaceted approach to understanding the botnet phenomenon. In: Proc. 6^{th} ACM SIGCOMM Conference on Internet measurement (October 2006)
15. Gu, G., Zhang, J., Lee, W.: BotSniffer: Detecting botnet command and control channels in network traffic. In: Proc. 15^{th} Annual Network and Distributed System Security Symposium (NDSS) (February 2008)
16. Fisher, D.: Storm, nugache lead dangerous new botnet barrage (December 2007), http://SearchSecurity.com
17. Grizzard, J., Sharma, V., Nunnery, C., Kang, B., Dagon, D.: Peer-to-peer botnets: overview and case study. In: Proc. 1^{st} Workshop on Hot Topics in Understanding Botnets (HotBots 2007) (April 2007)
18. Wang, P., Sparks, S., Zou, C.C.: An advanced hybrid peer-to-peer botnet. In: Proc. 1^{st} Workshop on Hot Topics in Understanding Botnets (HotBots 2007) (April 2007)
19. Kutznet, K., Fuhrmann, T.: Measuring large overlay networks - the overnet example. In: Proc. Kommunikation in Verteilten Systemen (KiVS) (March 2005)
20. Maymounkov, P., Mazières, D.: Kademlia: A peer-to-peer information system based on the XOR metric. In: Revised Papers from the 1^{st} International Workshop on Peer-to-Peer Systems (IPTPS) (March 2002)
21. Stewart, J.: Storm worm DDoS attack (February 2007), http://www.secureworks.com/research/threats/storm-worm
22. Utter, D.: Storm botnets using encrypted traffic (October 2007), http://www.securitypronews.com
23. Honeynet Project: Know your enemy: Fast-flux service networks (July 2007), http://www.honeynet.org/papers/honeynet
24. Gaudin, S.: Storm botnet puts up defenses and starts attacking back. InformationWeek (August 2007), http://www.www.informationweek.com
25. Erdös, P., Rényi, A.: On random graphs I. Publ. Math. 15, 290–297 (1959)
26. Barabási, A.L., Albert, R.: Emergence of scaling in random networks. Science 286, 509–512 (1999)

27. Watts, D.J., Strogatz, S.H.: Collective dynamics of ‘small-world’ networks. Nature 393, 440–442 (1998)
28. Albert, R., Jeong, H., Barabási, A.L.: Error and attack tolerance of complex networks. Nature 406, 378–382 (2000)
29. Crucitti, P., Latora, V., Marchiori, M., Rapisarda, A.: Error and attack tolerance of complex network. Physica A 340, 388–394 (2004)
30. Holme, P., Kim, B.J., Yoon, C.N., Han, S.K.: Attack vulnerability of complex networks. Physical Review E 65, 56109 (2002)
31. Stoica, I., Morris, R., Karger, D., Kaashoek, M., Balakrishnan, H.: Chord: a scalable peer-to-peer lookup service for internet applications. In: Proc. Annual ACM Conference of the Special Interest Group on Data Communication (SIGCOMM) (August 2001)
32. Gnutella forum: Gnutella (March 2001), http://www.gnutella.com
33. Clarke, I., Sandberg, O., Wiley, B., Hong, T.: Freenet: a distributed anonymous information storage and retrieval system. In: Proc. ICSI Workshop on Design Issues in Anonymity and Unobservability (July 2000)
34. Bollobás, B.: Random Graphs. Academic Press, London (1985)
35. Csárdi, G.: The igraph library (2005), http://cneurocvs.rmki.kfki.hu/igraph
36. Ripeanu, M., Foster, I., Iamnitchi, A.: Mapping the gnutella network: Properties of large-scale peer-to-peer systems and implications for system design. IEEE Internet Computing Journal 6, 50 (2002)
37. Jovanovic, M., Annexstein, F., Berman, K.: Scalability issues in large peer-to-peer networks - a case study of gnutella. Technical report, University of Cincinnati (January 2001)
38. Newman, M., Strogatz, S., Watts, D.: Random graphs with arbitrary degree distributions and their applications. Physical Review E 64(026118) (2001)