Enhancing Resilience of P2P Systems to DDoS Reflector Attacks

Xin Sun, Ruben Torres and Sanjay Rao
School of Electrical and Computer Engineering, Purdue University
{sun19,rtorresg,sanjay}@purdue.edu

Categories and Subject Descriptors

C.2.4 [**COMPUTER COMMUNICATION NETWORKS**]: Distributed Systems

General Terms

Design, Experimentation, Security

Introduction: Recently, we and other researchers have identified that P2P systems can be used as reflectors to launch large-scale DDoS attacks on nodes not even part of the overlay. More specifically, we focus on attacks where malicious nodes in a P2P system may subvert membership management mechanisms, and force a large fraction of nodes in the system to believe in the existence of, and communicate with a potentially arbitrary node in the Internet.

As an example, consider the search mechanism of Kad, a large kademlia-based DHT, as shown in Figure 1.a). Node A wishes to download a file F. A first queries node B which is known to it and whose ID is closer to the ID of F. B responds with C whose ID is even closer. Next A queries C who will respond with I, the index node for F. Then A will query I and obtain a set of sources for F. This mechanism can be exploited to cause a DDoS attack, as shown in Figure 1.b). A malicious node M may join the DHT as normal, but will respond with the victim V whenever being queried. So traffic will be redirected to V which is not part of the DHT. Such attacks may be hard to detect as the packets arriving at a victim are not distinguishable from normal protocol packets.
•*Contributions/Our Research:* This is the first effort at designing P2P systems resistant to DDoS attacks, and we are contributing to a systematic understanding of the interplay between P2P system design and vulnerability to DDoS attacks.

Achieving High Magnitude of Attacks: High attack magnitude can be achieved by the following heuristics: a) A malicious node may proactively push information about himself to a large number of nodes in the system, forcing them to add him to their routing tables, to increase the frequency of being contacted; b) he may also include the victim's address several times with different IDs in response to a query, to redirect the innocent clients to the victim multiple times and c) larger number of malicious nodes can be employed. We implemented the above heuristics, and easily achieved an attack of over 700 Mbps on a victim, by exploiting the live Kad system, as shown in Figure 2.

SIGCOMM'08, August 17–22, 2008, Seattle, Washington, USA.
ACM 978-1-60558-175-0/08/08.

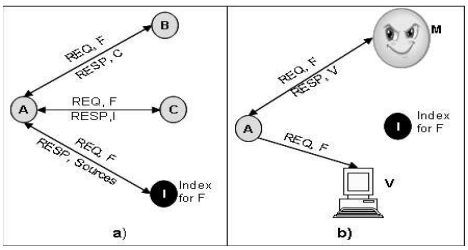

Figure 1: a) Kad Search mechanism. b) Redirection attack

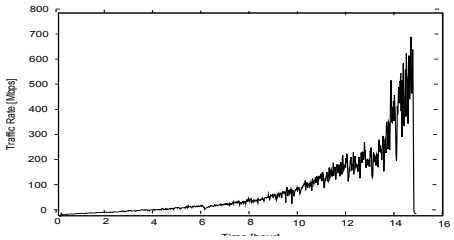

Figure 2: Traffic seen at victim, with 200 attackers.

Lessons for P2P System Designers: we extract more broadly applicable lessons regarding how the choices of system designers can influence the significance of DDoS reflector attacks possible.
•*Push VS. Pull:* The protocol must limit the ability of the attacker to redirect or infect a large number of innocent clients. Thus pull-based designs, where any information conveyed by a member is always in response to a prior solicitation, should be favored over push-based designs, where members may disseminate membership information to others in an unsolicited fashion.
•*Minimize skewness in node popularity:* The protocol must also limit the ability of an attacker to attract queries from innocent nodes toward itself.
•*Validate membership:* It's important to validate membership information that a node gets. Direct validation, where a node probes the new member it learns about to verify its existence, may itself become a source of spurious packets to the victim. Thus there must be strategies added to limit repeated failures of probes to the same network address.

Design Considerations: Our primary focus is on mechanisms that may be easily integrated in a wide range of existing peer-to-peer deployments.

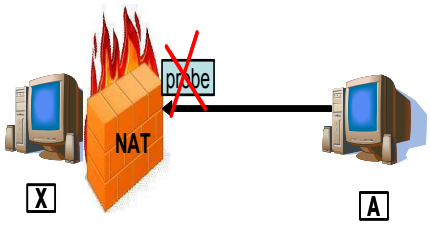

Figure 3: Benign validation failures due to NATs.

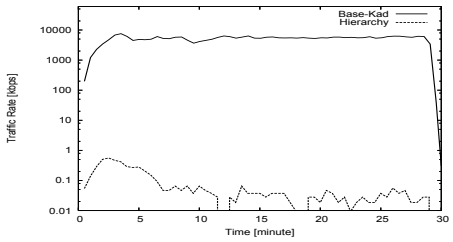

Figure 4: Traffic seen by the victim with Kad as a function of time with 50% of the nodes behind NATs and 5 attackers.

We believe that such mechanisms should not rely on centralized authorities. Centralized authorities can simplify validation of membership information by providing signed certificates that indicate that a member belongs to a group.

However, in many existing deployments, such as Kad and the DHT versions of BitTorrent, there are no such centralized authorities. Thus their existence cannot be assumed. Also, such mechanisms should be applicable to both structured and unstructured approaches overlays.

Framework Overview: We present our framework for enhancing the DDoS resilience of peer-to-peer systems. This is to be run locally on each client.

•*Self-monitoring validation failures:* Our scheme uses an active-probing based validation approach that monitors validation failures to network destinations. Repeated validation failures to a destination are an indication that the destination is under attack, and future validations are not sent in such a case. Tracking validation failures to individual IP addresses alone is not sufficient to tackle attacks against networks as a whole. Instead, our scheme monitors failures to network destinations at multiple levels of granularity, such as <IP, port>, IPs and prefixes.

Further, to tolerate benign failures due to churn, NATs and network congestions, our scheme considers not only the total number of failures, but also the *diversity* in the failures. For example, we will not send further validation packets to a prefix only when we have seen failures occurred to more than m IPs belonging to that prefix.

Finally, each validation failure is associated with a timestamp indicating when the failure occurred, and only those that occurred in a recent time window T are considered in our scheme.

•*NAT-Aware Validation:* Many systems including ESM, and the control/search phase of Kad employ *NAT-Agnostic* membership management operations. In these systems, when member B propagates information about member X to member A, there is no indication as to whether X is behind a NAT or not. As shown in Figure 3, if X were behind a NAT, a validation packet sent by A to X will not successfully reach X unless X had previously contacted A. To handle this, we require that membership management operations are *NAT-Aware*. In particular, membership information about nodes behind NATs are propagated with a flag indicating they are behind NATs. When a node A learns about X, it probes X only if it is not behind a NAT, thereby avoiding benign validation failures.

Resisting Disconnection Attacks: While our validation scheme has the potential to limit DDoS attacks on victims not in the group, it has a different vulnerability that could be exploited by a malicious node M. In particular, M could disconnect a genuine member (victim V) who is really part of the group, by flooding another node (say A) with several fake membership entries corresponding to the victim prefix. This causes A to exceed the bound for the victim prefix, and not communicate with genuine participants in that prefix, including V. These disconnection attacks are internal to the system and impact its performance.

We propose a defense scheme targeted at regimes where the total number of prefixes with attacker machines is small compared to the total number of prefixes spanned by all participants in the system. In the scheme once a prefix has experienced several validation failures and is suspected of being under a DDoS attack, i.e. once the diversity threshold is reached, instead of bounding the prefix entirely, future validations are sent to the prefix selectively based on the source IP address from which membership information is learnt. In particular, a random subset of IP prefixes is decided on, and validations are conducted only if (i) the source IP prefix belongs to the random subset; and (ii) other IP addresses with that prefix have not previously triggered a validation failure. Our scheme dynamically adjusts the size of the random subset, starting with the entire IP space, and reducing it by half each time that a failed validation occurs. The intuition is that if the attacker machines are localized to a small number of prefixes, then, the system would stabilize at larger subnet size. However, continued validation failures indicate that the number of prefixes spanned by attacker machines increases is larger, and the system moves to the smaller size.

Experiment Results: We implemented our defense framework in Kad. We have evaluated the framework on planetlab with real and synthetic traces of join/leave patterns. We use 680 nodes in total with certain fraction of them behind NATs.

Figure 4 shows the attack traffic generated at the victim as a function of time. With the original Kad, the traffic was as high as 10 Mbps throughout the run. However, with our scheme, the attack magnitude was effectively reduced by a factor of 100,000 from 10 Mbps to 0.1 Kbps.

Other results show that our scheme does not impact the application performance, can keep the false positives due to benign validation failures low and works well even with a large fraction of nodes behind NATs.

Ongoing work: We are exploring designs where nodes can share with peers the information about validation failures. So the DDoS traffic can be further reduced by requiring only a subset of nodes to perform validations. We are investigating how to enable this by leveraging reputation mechanisms.

Additional information: Please visit the project homepage at http://cobweb.ecn.purdue.edu/ isl/secp2p.htm