DRDoS Attack: a survey about how to defend against it, the right way.

MATTIA GIACOBBE – 1069075 – UNIVERSITÀ DEGLI STUDI DI PADOVA – DIPARTIMENTO DI MATEMATICA

This essay presents a survey about DRDoS Attacks and the possible ways to defend against it. Distributed Denial of Service (DDoS) attacks are a kind of disruptive technique that aims to deny a service provided by a network or server by sending an excessive load of traffic that exhaust the victim's resources. It is *Distributed* because an attacker takes control of a series of Internet hosts and makes them send malicious traffic to whatever destination he wants.

Distributed and *Reflective* DoS attacks are a new kind of DDoS attack that emerged only recently. In a DRDoS attack a series of legitimate hosts (called *reflectors*) are used to send huge amount of traffic to a victim using a technique known as IP spoofing. IP spoofing consist of altering, on the packet header, the IP address of the source changing it to the designated victim IP. This will make the reflector send the answer to the victim instead of back to the attacker. Often time, depending on the reflector, the answer will be way larger than the request leading to an amplification factor for the packets redirected to the victim. In this case we call the reflectors *amplifiers*.

In this paper I will analyze the concept of DRDoS attacks, the way they are designed and how they are performed. Then I will list some of the existing proposed solutions to point out a range of different design goals used to implement a countermeasure. I will then compare those proposals and point out strengths and weaknesses of each of them to get a general outline for future proposals.

I. Introduction

Distributed and Reflective Denial of Service attacks are a particular type of DDoS attack that emerged only recently and posed a serious threat to the integrity of the Internet. Massive DRDoS attacks can drisrupt major Internet applications and services making them unavailable for the public.

DRDoS attacks use so called reflectors to send large traffic packets to the victim using spoofed IP addresses. Attackers exploit internet hosts for which they know the IP address and then use them as reflectors to send response packets to the victim with the spoofed IP in the request header. These hosts are often used as amplifiers because the responses are way larger than the request causing overload and resource exhaustion on the victim's server or network.

On the other hand, a regular DDoS attack compromises a series of hosts and controls them to send traffic wherever the attacker wants. This requires a lot more effort from the attacker side because he has to possess deep knowledge of the network and the host he's trying to control to be able to enslave it. This makes it easier to trace back the attack source because there is no spoofing going on. Solutions that implemented trace back worked until this new kind of DDoS attack came out. They will not work anymore when the attack is reflected to the victim via IP spoofing because it is impossible to find the source of the request if the reflector "thinks" that sending it was the victim.

There were a lot of proposals to answer the problem and we will cover some of them later in this paper. This particular subset of papers is a good starting point to point out different kind of approaches to the problem, ranging from IP spoofing denial to per-packet basis detection. This is only an example of what I am going to cover later on.

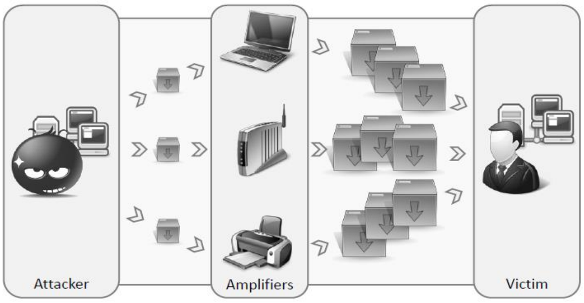
II. Conventional countermeasures

The basic thing to keep in mind while dealing with Distributed Reflective Denial of Service is that attacks are somewhat performed by innocent reflectors, making it a hard to identify the real attack source and to stop or mitigate the attack.

Since attack packets come from regular hosts rather that a malicious source, standard address filtering or black-listing is not effective at all because we cannot refuse communication from a legitimate hosts providing a service that might be useful to our network. Since these reflectors will be perceived as attackers it will take longer time to discover who really is performing the DRDoS, resulting in a huge effort from the defender point of view because he has to find a way to efficiently filter packets while the attacker, with much littler effort, will continue to send a huge amount of data from different places.

It is indeed difficult to distinguish an actual attack from a sudden rise in popularity for a given service due to a flash crowd ("Slashdot effect") [6]. This poses a high burden from the defender point of view: what are good requests and what are bad requests?

To have an idea of what goes on during a DRDoS attack, the image below [1], shows a typical scenario of a host that is victim of amplified traffic from reflectors:



We see that the attacker sends packets with spoofed IP address to a number of reflector or amplifiers and this points out why, as seen above, trace back techniques would lead us only to the reflector and hardly to the real attack source. It is also clear that packet filtering will not always be a good idea since the amplifiers provide a benign service to the public and completely blocking communication with them might result in a penalty from the victim's point of view.

It means that we have to move to other kinds of countermeasure design, considering the nature of a DRDoS attack and the way it is performed. Defense systems must be built on the source side, reflectors side (which means that every host on the Internet must install the same system since every host can be used as a reflector) or even on the routers that attack packets follow to reach the victims service or network.

The main problem to address is also what makes a DRDoS attack strong: IP spoofing. This leads to a countermeasure design that tries to negate the spoofing like TCP based networks do with the handshake. This won't always be acceptable and we will see that there are other sides to take into consideration.

We will see, in more details, examples of countermeasures designed one way or the other and compare them to find out what has to be kept in mind when proposing a countermeasure and how it's better to implement it and what limitation the Internet poses to it.

III. DRDoS countermeasure guideline

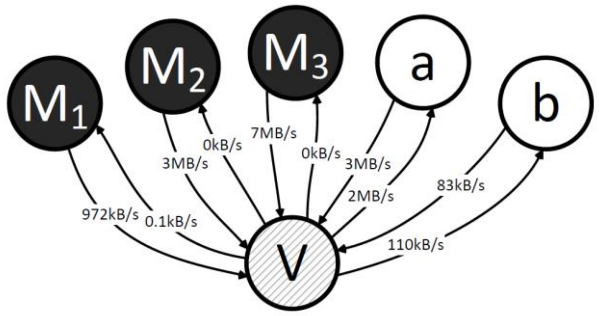
There is a general pattern leading to well-built DRDoS protection, both proactive and reactive:

I. Preventing IP address spoofing
II. Protocol hardening
III. Request/response symmetry
IV. Secure service configuration
V. Packet based filtering

The first point is the most obvious: if we can prevent IP address spoofing we have completely denied any kind of attack since its strength comes from this particular technique. Unfortunately, this is the hardest part and it is also what makes a DRDoS attack so strong. It is also not always applicable since a lot of services must run on UDP based network for the speed, thus enabling IP spoofing. For this very reason we need to move away from this concept.

Protocol hardening on the reflector side would reduce significantly the impact of an attack, since UDP is a way better candidate for DRDoS that TCP. Improving it will lead to better handling the attack when detected.

Request/response symmetry is a particular property that a host being attacked can monitor. As seen in the image below [1], the requested data from the victim to the reflector is way less that the amount of data the reflector sends to the victim.



We see that the three reflectors M1, M2 and M3 are sending huge amount of data to V while V never requested anything to them. This is a big indicator that the victim V is being attacked and it is possible to design a countermeasure based on this kind of indication.

From this point of view, we can monitor the in- and outgoing traffic and compare the amount of data transferred. If the imbalance is too high, it means that an attack is going on.

Secure service configuration is just a prerogative. A well configured service or network, with proper firewalls and all the precautions it's the fundamental in which to craft proper DRDoS protection.

Packet based filtering is another solution which is used by a lot of proposed countermeasures (like Pi – see below). We identify the attack monitoring the number of packets received and act accordingly. It basically takes into consideration that the system can recognize the source or the path that a packet takes to see if it is malicious or not.

IV. Relevant proposals

Now it is time to take a look at some proposals and see in details what they take into consideration and what aspect of the attack they are based on.

Yaar et al. proposed **SIFF** [2], a Stateless Internet Flow Filter that aims to mitigate the damage caused by a DRDoS attack. This system provides a server with the ability to establish a privileged connection with whatever client asking for it. Privileged channels are created through a particular handshake with the service provider and grants priority over unprivileged communications. This way, on a high traffic load scenario (which may indicate an attack is going on), only the privileged requests will be served and, since a spoofed request cannot satisfy the handshake, all of the attack packets will be ignored until the situation is less critical.

This kind of countermeasure tries to avoid the main threat of a DRDoS attack: IP spoofing (as we saw before). With the possibility of a handshake to establish a connection to the service we prevent the threat on its root. If an attacker tries to flood the victim with SYN packets requesting the privileged connection the damage will not be as big as it would otherwise so it is not a problem answered in this paper. The downside of SIFF is that in an excessive load scenario (be it malicious or benign) will be dropped. This includes requests sent by a legitimate user. And for some services the need for a handshake to handle an attack would not be acceptable.

Mirkovich et al. proposed **D-WARD** [3], which is a source-end defense against DRDoS. This technique monitors outgoing traffic directed to the victim and will stop any illegitimate traffic sent. This is achieved keeping statistics of regular traffic and, when it detects something unusual it will be blocked. Legitimate traffic is, on the other hand, granted and forwarded.

This is an example of a solution implemented at the source; D-WARD keeps track of all the traffic sent to the potential victim. If it reveals suspicious communication it compares it with regular ones, previously logged by the system, and drop all confirmed malicious requests. This is another kind of philosophy that doesn't answer IP-spoofing but chooses to "classify" what is benign and what is not and the kind of philosophy we are moving to.

The downside of this proposal is that the amount of data to keep tracking is huge and it will have performance issued in a network used by billions of people. For this reason, it may be a good solution for services aimed at a smaller user base, leaving the problem open for bigger networks.

Again Yaar et al. proposed another kind of countermeasure: **Pi** [4]. Pi is a path identification mechanism that aims to mark every packet routed to the victim with a unique marking based on the route it has taken. If a path is recognized as malicious the victim only needs to drop every packet with the Pi marking corresponding to that particular path. The only downfall of this idea is that in every router of the network must be installed the Pi marking mechanism and not always will be affordable. If this is granted tho, Pi would be one of the best way to deal with the massive amount of data that a DRDoS attack will send, since the work is alleviated through all the routers, thus making it easier for the host to react.

Pi poses another kind of mindset to challenge the problem: we detect malicious packet flows at the router/routing level. When we identify a route taken by a packet to be suspicious we know that all the packets with the same marking all come from the same source, meaning that they are either all benign or malicious. This solution takes for granted that there is a mechanism that can indicate if a route is malicious or not.

Another proposal comes from Tsunoda et al. [5] and, even if based on a simplified scenario, introduces a really solid and yet simple mechanism to block incoming DDoS traffic. The core concept is very simple as it focuses only on pairs of request and response packets without the need to manage the complicated transition of protocols such as TCP. For every response packet, the corresponding *valid* request packet must be monitored beforehand. In the basic case that there is a reflected response from an amplifier, no requests from the hosts would have been registered and thus it must be illegal traffic.

This mechanism is built on a dedicated detector that can monitor both request and response packets. The validity of a packet is granted by bidirectional symmetry between request and response. The detector can store the, supposedly, limited number of requests that can be sent to the potential victim with the ones being received. If there is a match the communication is ok and it can continue, otherwise it is considered malicious and dropped. The efficiency of the system relies on a limited amount of request types attributed to the host and its complexity will grow exponentially the more request a host can manage. This is not discussed in this paper because the focus is on the design and the core ideas behind the countermeasure.

This proposal shows a lot of potential, because malicious communications are easily detectable, once the monitoring system has been properly designed even for large number of requests. Since, nowadays, is too limited to be implemented in a real-world scenario it will remain just a theoretical countermeasure. Again we do not take in consideration the possibility to answer IP spoofing but, with an external entity as the detector, we analyze the traffic to detect incoming attacks.

Multiple factors are taken into consideration when building a countermeasure for DRDoS attacks. Some are more suitable for certain situation than others and also the complexity of the network plays an important role. The main DRDoS threat comes from IP spoofing but it is not necessary to deny it to protect against it. It sure is a serious problem exploited by many but it is also what the current Internet infrastructure allows and changing it is not an easily accomplishable task.

For these reason, proposals such as Pi and D-WARD are born. It makes it possible to limit denial of service damage by exploiting characteristics proper of the attack, referring on the list in chapter 3 of this paper.

V. Proposals at comparison

From the countermeasures listed above we can see that there is something that they all share: every solution is implemented and works in a way that exploits common DRDoS attacks behavior.

Following the guidelines provided and summarized in chapter 3 of this paper, exploiting the huge request/response asymmetry that an attack of this type causes and, of course, the IP spoofing prevention as in SIFF.

This trend poses the main thought to keep in mind while designing a new countermeasure or event when improving an existing one, to identify properly if an attack is being performed to be able to act accordingly.

All of the above considering, of course, correct and secure service configuration that is the core principle taken for granted in this paper.

In general, we see that Pi is the solution that differ mostly from the other solutions in the design, other than SIFF which is the only one presented here that tries to remove completely the concept of IP spoofing.

This is because Pi works at a packet basis, whilst every other defense mechanism implemented some kind of monitoring system that kept an eye on in and outgoing transmissions. All of which are implemented at the hosts side; Pi on the other hand is a way more scalable and distributed solution.

What all of the proposals have in common is that, one way or another, they implement a way to *detect* an ongoing attack and then take decision based on that. So, more that trying to be immune to attacks, a system must be able to detect when it is victim of a DRDoS. The focus then must be *detection* instead of complete *prevention*.

A DRDoS attack detected in time will not be able to cause enough traffic to disable, even temporarily a service or network if held correctly.

VI. Conclusions

In this paper we went through analyzing this relatively new kind of DoS attack, the DRDoS, and what are the main aspects and strength of its execution. We analyzed the way it interacts with the reflectors and compared it with the regular DDoS botnet.

I then summarized some of the most interesting proposals in terms of design goals. I chose these particular solutions because they ranged a lot on every aspect of a DRDoS and this led me to show what are some of the possible routes to take into consideration when designing a new proposal. I showed also how the listed proposals answered the particular aspect of the attack they were taking into consideration, with pros and cons. Finally compared the solutions and critically evaluated their properties such as scalability and real world deploying potential.

I also outlined that most of DRDoS countermeasures take into consideration a simplified version of the attack scenario or are built on top of existing technology to work best such as malicious routes detection for Pi. For this reason, they must be re-adapted to real-world scenarios to be effectively deployed.

VII. References

[1] Christian Rossow, *Amplification Hell: Revisiting Network Protocols for DDoS Abuse*, at: Horst Gortz Institute for IT-Security, Ruhr University Bochum, Germany.

[2] Abraham, Yaar Adrian, Perrig Dawn, Song, *SIFF: A Stateless Internet Flow Filter to Mitigate DDoS Flooding Attacks*, at: Carnegie Mellon University

[3] Jelena Mirkovic, Peter Reiher, *D-WARD: A Source-End Defense against Flooding Denial-of-Service Attacks*

[4] Abraham, Yaar Adrian, Perrig Dawn, Song, *Pi: A Path Identification Mechanism to Defend against DDoS Attacks*, at: Carnegie Mellon University

[5] Hiroshi Tsunoda, Kohei Ohta, Atsunori Yamamoto, Nirwan Ansari, Yuji Waizumi e, Yoshiaki Nemoto, *Detecting DRDoS attacks by a simple response packet confirmation mechanism*, at: Tohoku Institute of Technology

[6] Marc Kührer, Thomas Hupperich, Christian Rossow, Thorsten Holz, *Exit from Hell? Reducing the Impact of Amplification DDoS Attacks*, at: Ruhr-University Bochum