Network DDoS Layer 3/4/7 Mitigation via Dynamic Web Redirection

Todd Booth and Karl Andersson(✉)

Division of Computer Science, Luleå University of Technology,
97187 Luleå, Sweden
PhD@ToddBooth.Com, Karl.Andersson@Ltu.Se
http://OrcId.Org/0000-0003-0593-1253,
http://OrcId.Org/0000-0003-0244-3561

Abstract. Layer 3, 4 and 7 DDoS attacks are common and very difficult to defend against. The academic community has published hundreds of well thought out algorithms, which require changes in computer networking equipment, to better detect and mitigate these attacks. The problem with these solutions, is that they require computer networking manufacturers to make changes to their hardware and/or software. On the other hand, with our solution, absolutely no hardware or software changes are required. We only require the use of BGP4 Flow-Spec, which has already been widely deployed many years ago. Further the customers' own ISP does not require Flow-Spec. Our algorithm protects groups of over sixty-five thousand different customers, via the aggregation into one very small Flow-Spec rule. In this paper, we propose our novel, low cost and efficient solution, to both detect and greatly mitigate any and all types of L347 DDoS Web attacks.

Keywords: DDoS · DRDoS · Bandwidth · Reflector · BotNet · BGP4 · Flow-Spec

1 Introduction

Various acronyms and terms used in this paper, are defined in the Table 1. There are numerous academic papers, which provide the background, present case studies, and/or perform a literature survey concerning detecting and/or mitigating network based distributed denial of service (DDoS) attacks [2,6,9–11,17–21]. Therefore, this paper will limit the background and will not repeat the same numerous figures. As an illustration, in this paper we will refer to a Bank, as the on-line web service under attack. However, there is nothing bank specific in the solution, so it is applicable to any public Web service. Note that this paper is only a conceptual design and the experiment has been left as recommended future work.

R. Doss et al. (Eds.): FNSS 2016, CCIS 670, pp. 111–125, 2016.
DOI: 10.1007/978-3-319-48021-3_8

Table 1. Acronym and term definition table

Term	Definition
BGP4	Border Gateway Protocol version 4
BotNet	A network collection of zombies (PCs infected with Malware)
CAPTCHA	Completely Automated Public Turing Test
CDN	Content Delivery Network
DoS	Denial of Service attack
DDoS	Distributed Denial of Service attack
DSR	Design Science Research methodology
Booters	DDoS attacks as a service (for rent)
DRDoS	Distributed Reflection DoS
IETF	Internet Engineering Task Force
IP	Focus in this paper is IPv4
ISP	Internet Service Provider
L4	Layer 4 (transport)
L7	Layer 7 (application)
L347	IP layer 3, 4 and/or 7 attacks
MPLS	Multi-protocol Label Switching
NATO	North Atlantic Treaty Organization
Null-Route	ISP basically discards all traffic, concerning the DDoS
NTP	Network Time Protocol
RFC	IETF request for comments document
SDN	Software Defined Networks
WAF	Web Application Firewall
Zombies	A collection of malware infected, remote controlled hosts

1.1 Research Problem

Information Systems, often include Web servers, which are accessible via the Internet (publicly facing). These Information Systems are being constantly being successfully attacked via network based DDoS attacks. There are a wide variety of network based DDoS attacks, such as network layer 3 (L3), transport layer 4 (L4) and application layer 7 (L7) attacks. Collectively, we will refer to these as L347 network attacks. A recent 2016 DDoS Internet attack measured over 400 Gbps [8]. Very close to 0 % of organizations have 400 Gbps, of ISP bandwidth, so these attacks cannot be prevented, by trying to only stop the attack, at the organizations' premises. So organizations sometimes try to have their ISP or Web server provider mitigate these attacks. However, many ISPs and cloud provides will not have enough free bandwidth to handle an attack of 400 Gbps. Even if they did, the solution to process the 400 Gbps stream is often very expensive. What many ISPs and cloud providers will do, is during a DDoS attack, they will null-route the organizations incoming traffic, until the DDoS is over, which means that the organization will be completely down. There are many other types of L347 network attacks. There are some really great solutions, however these are often too expensive, too complex, or require computer network manufacturer hardware and/or software changes. For example, CloudFlare's anti-DDoS Enterprise solution starts at 5,000 USD/month.

Our research problem context is limited to DDoS L347 network attacks which traverse the Internet and attack on-line Web servers. We focus on protecting online services which require authentication (logging in). There is a great deal of general literature, as how to detect and/or mitigate L347 DDoS attacks. However, the research literature is very weak, concerning how to do this, in our specific research context. Also, much of the literature only answers specific practical questions, but there is a lack of literature concerning the related conceptual and applied research questions. To mitigate all of the L347 network attacks, one must also come also up with the algorithm, as how to mitigate all of these, in a very efficient manner. This requires putting together best in class L3, L4, and L7 specific solutions, into a comprehensive high level system algorithm. Our research question is to design a best in class anti-DDoS L347 solution which costs almost nothing, which is simple to implement, easy to understand and does not require any network equipment hardware or software changes.

1.2 Contributions

As related to L347 attacks against the Bank, we answer various conceptual, applied and practical questions in this paper. In security, there is a defense in which the risk is transferred. As a conceptual design principle, we propose that (1) whenever possible, the Bank transfers DDoS risks to service providers at a very low cost, (2) the Bank uses different IP addresses for Web services, one for pre-authentication and one for post-authentication, and (3) the Bank gives each customer their very own unique sub-domain, which is used by the customer after authentication, to access the bank's Web services.

The above design contributions and reasons for them are explained later in this paper. In addition to the previous design contribution, we have other design contributions which are found in our design cycle discussions.

1.3 Research Methodology

We followed the design science research (DSR) methodology [14]. A DSR IT artifact can also be the design guidelines for an IT artifact, as opposed to a physical IT artifact itself. Our high level IT artifact is our proposed design guidelines and algorithms, which greatly mitigate any and all L347 network based DDoS attacks. Via DSR, an IT artifact should be created, then evaluated, and then re-designed with improvements (based on the feedback from the evaluation). This cycle is then repeated several times. These cycles then continue, until an adequate level of new knowledge is acquired and/or a practical solution emerges.

It turns out that this approach will make it easier for the reader to understand our final and total solution.

1.4 Summary of Network DDoS Attacks

We will provide a very brief overview, of DDoS attacks. Direct network attacks and indirection Reflection Attacks are shown in Fig. 1.

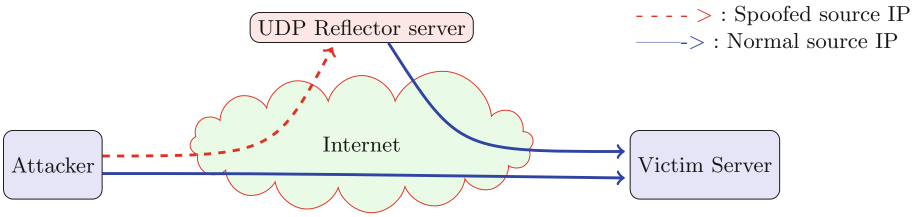

Fig. 1. Direct and reflection attack

An example of how IP source address spoofing works, is shown in Fig. 2.

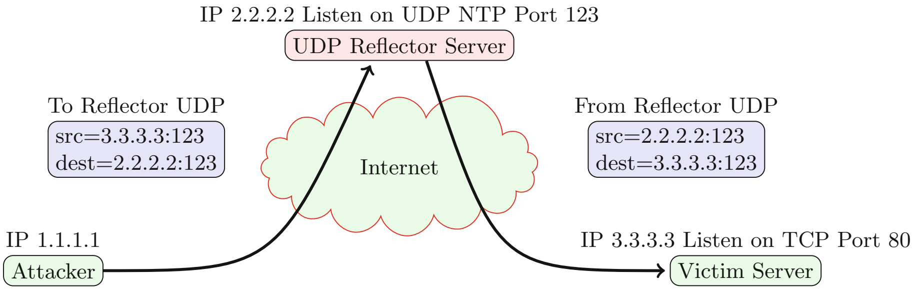

Fig. 2. Packet transitions during UDP reflection attack

With a DDoS attack, there is the attacker controller, masters, Zombies, Reflectors and Victims, as shown in Fig. 3.

1.5 Outline of This Paper

The rest of this paper is organized as follows: In Sect. 2, we perform DSR methodology and go through several design cycles. This is where we explain the specific research problem issues and our proposed solutions. In Sect. 3, we analyze related works and include a synthesis of those works. In Sect. 4, we provide our conclusions and future work suggestions.

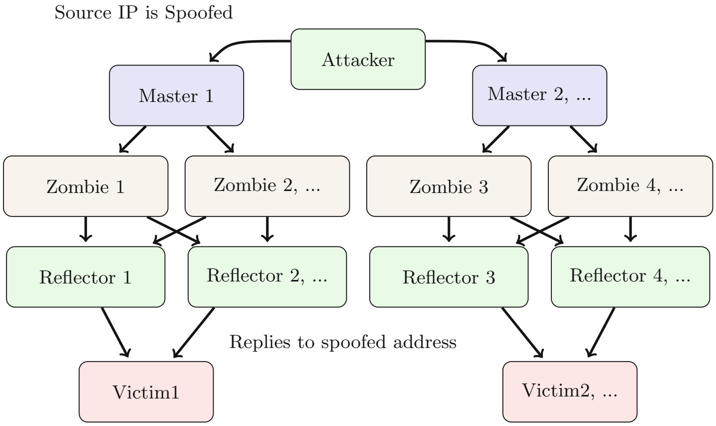

Fig. 3. Detailed and complex reflection attack

2 DSR Methodology - Design Cycles

We now present our DSR design cycles and related design contributions.

1. On Premise Solution: We will first provide our design based on a very simple case. Assume that the bank has an on premise only solution, that the bank's ISP link is 10 Gbps, and that there is an attack of 15 Gbps. The only way to defend against this is for the bank to increase their ISP like to, for example 20 Gbps, which may take several days. However, the attacker will perhaps just immediately increase the attack to 30 Gbps. So the bank can not provide a defense, which is entirely on premise.

2. Local ISP Solution: The bank decides to try and mitigate the attack, at their ISP. Some ISPs scrub out malicious DDoS traffic in-house. However, most ISPs do not offer an in-house solution. So most ISPs outsource the scrubbing dynamically after detection, by rerouting customer traffic, via BGP4. The outsource solution might take an hour, after the attack is detected, to start diverting all traffic (via BGP4), to the outsource scrubbing service. The attackers could repeated stop the attack after the outsource is operational. Then they could wait for the traffic to be sent directly to the customer at which time they start the attack again. Both types suffer from an inability to detect all malicious and all valid traffic. So most of the time, some malicious traffic is let through and some valid customer traffic is dropped.

3. Upstream of ISP Solution: In the previous cycle, the bank had the ISP try to filter the attack, before the malicious traffic reached the bank. The ISP can use a similar strategy. The ISP can try to have it's upstream neighbors filter the malicious traffic, before it reaches the ISP. There is a standardized way for the ISP to state which malicious traffic should be filtered upstream, which is called border gateway protocol version 4 (BGP4) Flow-Spec. However, most ISPs don't support BGP4 Flow-Spec. Let's assume the bank is sharing their ISP connection for (1) incoming Web service traffic and (2) outgoing general employee traffic. In this scenario, it is not possible to create BGP4 Flow-Spec filters, which would filter out most of the DDoS attack traffic.

4. Solution Located Many Hops from Customer: To solve all of the previous cycle issues, let's first create a theoretical and virtual solution. We now assume that there is one really great low cost virtual ISP in the world, that can be used by each and every end customer in the world (including the bank). We assume that every worldwide customer, such as the bank, can connect directly to this virtual ISP. So the bank can simply connect directly to this virtual ISP via a 10 Gbps and the virtual ISP can filter out most of the malicious traffic, at a low cost. We have found an actual current solution, which has many similar features, to this virtual ISP solution. However, instead of this virtual ISP connecting to the bank in-line, we need to use a variety of different technologies and completely change the design.

To understand the rest of this paper, one must have a good understanding of how content delivery networks (CDN) and reverse Web proxies work, a great tutorial which is found here [16]. As an example, to illustrate our solution, we will discuss the CloudFlare CDN solution, which has 86 data centers. To use this CDN, in order to gain the above virtual ISP functionality, a summary of the changes follow (which is simplified).

Let's assume that the bank's Web DNS entry currently points to the bank's public and well known IP address, of 11.1.1.1. The bank's Web DNS entry IP address needs to be changed to CloudFlare's IP address of 22.2.2.2. The bank needs to change their previously public IP address of 111.1.1.1 to some secret public IP address, such as 33.3.3.3. This CloudFlare service will setup a reverse proxy in each of their 86 data centers. The bank's customers and attackers will now receive the CloudFlare IP address of 22.2.2.2, when trying to connect to the http//bank.com DNS name. It is important to understand that all 84 CloudFlare data centers will have the same IP address of 22.2.2.2, which is considered anycast routing.

When the bank's customers try to connect to bank.com/22.2.2.2, they will be sent to the closest CloudFlare data center's reverse proxy. Then the CloudFlare reverse proxy will connect to the bank's secret IP address, which is 33.3.3.3. Customers and attackers will never know the bank's secret IP address, so they will never communicate directly with this address of 33.3.3.3. Let's assume that the attackers are sending a 400 Gbps DDoS attack to the bank. Again, via the DNS name of bank.com, they will only learn the CloudFlare IP address of 22.2.2.2. The attackers may use hundreds or thousands of zombies, which are part of a BotNet. So these zombies may be scattered around the world. For each zombie, their attack traffic will be directed to the CloudFlare data center, which is located the closest to this zombie. In summary, the 400 Gbps attack traffic is distributed among the 86 CloudFlare data centers, so the attack at each CloudFlare data center is normally much smaller than the aggregate 400 Gbps.

The majority of high volume bandwidth DDoS attacks are of the type, reflection attacks, which we fully described in our previous International Conference on Future Network Systems and Security (FNSS) conference contribution [4] and a related journal article [5]. This paper picks up where those papers left off. A reflection summary is simply that the attacker can send an aggregate of, for example 10 Gbps, and the reflectors will amplify that by, for example 40x. So the reflectors will receive 10 Gbps from the attackers but they will send 400 Gbps to the victim servers. This 400 Gbps of reflection attack is not valid HTTP traffic, so no possible reflection attack traffic would be forwarded to the bank's secret IP address. So none of the DDoS reflection attack traffic, with a bandwidth of 400 Gbps would ever reach the bank's local premise!

The bank, other organizations and individuals can get this functionality from CloudFlare for free. Now CloudFlare would not be very happy to keep handling 400 Gbps bandwidth attacks without any service revenue. However, CloudFlare can (and does) use BGP4 Flow-Spec. This can be used by CloudFlare, to require their upstream neighbors to filter all non-Web traffic, which is destined to reach CloudFlare's Web reverse proxy servers. So CloudFlare does not need to receive any of this reflection DDoS attack 400 Gbps traffic. Having said that, CloudFlare can instead accept the DDoS attack for a short while, before sending the BGP4 Flow-Spec request, which will allow CloudFlare to better analyze the attack, and to properly rate the agreement Gbps volume.

Of course, the attackers can try other ways to send very high bandwidth DDoS attacks, its just that the reflection DDoS attacks will not be received by the bank's Web server (33.3.3.3). From the perspective of the bank and CloudFlare, all reflection attacks have been eliminated. However, in security terminology, we say that this attack risk has been transferred to CloudFlare's upstream neighbors, who will perform the filtering, based on receiving BGP4 Flow-Spec filter requests.

5. Solution for L3 Non-reflection Attacks: L3 network attacks can be either reflection attacks, or non-reflection attacks. In the previous cycle, we basically eliminated DDoS reflection L3 network attacks (from the point of view from CloudFlare and the bank). Without these attacks, we only have L3 non-reflection attacks left. L3 non-reflection attacks can be divided into two types, with IP source address spoofing and without spoofing. The only type of attack traffic that would reach the bank's Web site (33.3.3.3) are valid L4 and/or L7 traffic (which we do not consider as also an L3 attack). So for either type of L3 attack, none of this L3 network attack traffic would reach the bank's Web site (33.3.3.3). All of this L3 attack traffic, would be terminated at or before the CloudFlare reverse proxy. So L3 attacks have no direct effect, on the bank's Web servers. This L3 attack risk has been transferred from the bank to CloudFlare.

6. Solution for L4/7 Attacks: The only remaining attack traffic to consider are L47 attacks. We'll now consider L4 attacks, which we will consider as follows. L4 attacks are based on L4 TCP connection/termination requests, setting TCP flags, and performing strange or unusual TCP transport activities, often at a very rapid rate. For any attack that actually sends traffic to the Web server process, we will consider that later in this paper, as an L7 attack. There are of course attacks, which are both an L4 and L7 attack. We'll now address pure L4 attacks and the L4 portion of any L4+7 attacks. For the attacks that don't end up opening an L4 TCP transport connection, the CloudFlare reverse proxies will not open a TCP connection to the bank's Web server.

Many of the other L4 attacks can be stopped with a standard stateful firewall. CloudFlare's professional plan, which is 20 USD/month, includes a Web application firewall (WAF), which includes support to stop most L4 attacks. An alternate solution is the following, where the bank continues to use the CloudFlare free plan. Then, instead of having the bank Web server on premises, at 33.3.3.3, the bank runs their own virtual machine (VM) guest Web server, in the Microsoft Azure cloud. Then CloudFlare is the front-end, for this Microsoft Azure cloud based bank Web server. The low end Azure cloud cost is about 20 Euro/month for a VM guest. This Azure service includes a free L4 stateful firewall, which will stop most of the L4 attacks. Also, the VM guests include a 10 Gbps link, which will handle DDoS L47 bursts in traffic, at a very low cost. The attacks can still spoof their source IP address, to that of a valid session. However, they would need to know the state of the L4 connection, since otherwise the stateful firewalls would block the malicious traffic. If the attack knows the L4 connection state, they can for example, keep sending the most recent TCP response, which could be forwarded to the bank's Web server. However, we'll consider this as an L7 attack, which will be addressed next. So the bank's Web server does not directly receive these L4 attacks. All pure L4 attacks risks has been transferred from the bank to CloudFlare, or in the alternate design to Microsoft.

Solution for Remaining L7 Attacks: We now consider the remaining L7 attacks, which require a TCP connection to be opened with the reverse proxy. To complete the TCP three-way handshake, the client would not be able to use IP source address spoofing. However, once the TCP connection is established, IP source address spoofed traffic can be sent. Here is an example of an L7 attack. Numerous attack clients could collectively open millions of TCP connections and slowly request web pages, in order to deplete the bank's Web server memory, and processing power. This would also be an attack against the network bandwidth. However, since the attack is based on L7 requests, we consider this as an L7 attack, instead of an L3 attack.

We will call the Web clients who are accessible the bank's Web server, but have not yet authentication, as pre-authenticated clients. After they login, we will call those clients as authenticated clients. Many organizations use the same URL for both pre-authenticated and authenticated clients. In this case, as these

L4 attacks are performed, even from pre-authenticated clients, it might have an effect on the authenticated clients. Our design guideline, is to use different Web servers, for pre-authenticated and authenticated clients.

For our example, let's assume that the bank also uses the Microsoft Azure Web hosting service, for handling just the authenticated clients. With this hosting service, it is Microsoft who owns and operates the Web server and the bank only receives Web requests which include the bank's URL. With this design, any L347 attacks towards the Microsoft Web server, which don't include the bank's URLs are handled by Microsoft and have limited effect on the bank's cloud based processes. We recommend that the CloudFlare solution (and bank Web server 33.3.3.3) is now only used for the pre-authenticated traffic. Upon authentication, the specific customer should be sent a Web redirect (or via click URL) to move from the CloudFlare IP address to the Azure Web service.

Dynamic Web Redirection Solution: During normal operation, where there is not an attack, we will have all customers surf to the same URL. However, during a DDoS attack, we will redirect all customers (after authentication) to their own unique URLs. We will now design the architecture, so that during a DDoS attack, we can very easily move almost all of remaining possible Azure related attack traffic, from the bank's Azure Web service process to the Microsoft Web server.

Let's assume that the bank has 1,000 customers. We propose that the bank assigns each of these customers a unique 40-character sub-domain name. Let's suppose account ID number 74 is assigned the DNS sub-domain of "0745X4...BE6". In the Azure web server, you could then create a DNS CNAME entry of `http://0745X4...BE6.Bank2.Com`, which points to this Azure site. If the customer tries to access their account information, the URL might be something like: `http://0745X4...BE6.Bank2.Com/account-info`, instead of http://Bank2.Com/account-info (which would only be used when there is no DDoS attack). The bank should configure the DNS server to prevent any unauthorized zone transfers, since we need to keep these customer sub-domain names secret. The bank then configures the Azure Web server to accept traffic for these 1,000 sub-domains, but not to accept any other Web requests. When a customer authenticates, via the CloudFlare service, they are redirected to the Azure Web server, with their very own secret sub-domain name.

Now let's talk about how to detect DDoS attacks which reach the Bank's Azure Web server process. We create a list for each specific customer ID/sub-domain. If there are 1,000 active customers, we have 1,000 active lists. For a given customer sub-domain list, we keep track of all source IP addresses, that are actively sending traffic to this customer's sub-domain. A customer would normally not login from more than a couple of IP addresses simultaneously. However, a DDoS attack, by definition, would be when a large number of attackers, would be sending traffic. So if, for example, there is incoming traffic from more than ten source IP addresses, to the same customer sub-domain, we have detected an L7 attack. Put another way, we analyze all traffic, to a given customer sub-domain, and looking at that traffic only, we try to figure out if there is a DDoS attack. It is perhaps 1,000 times easier to detect a DDoS attack, since we only consider traffic towards each customer sub-domain, on its own, and then decide if it looks like a normal bank customer's traffic pattern.

Once an attack is detected, for a specific customer, we have a variety of options. Here is one option. As long as possible, do the following (and only until there is a huge amount of attack traffic). Continue to service requests, to this customer ID sub-domain. However, add a random 1–3 s delay per request, before serving the web pages. Hopefully, the attack would come from a large number of IP addresses, which can be retained. It is public information, as to which ISP/AS owns every public IP address.

When the bank decides to stop an attack, they can (1) terminate all of only this customer's sessions, (2) delete this customer's sub-domain, (3) assign the customer a new sub-domain, and (4) register this new sub-domain on the Azure service. Only after a new successful login, would the customer be redirected to their new customer ID sub-domain, on the Azure service. By deleting the old domain, the following will occur. For all future attack traffic, to this customer's old sub-domain, it would no longer reach the bank's Azure Web process. Instead, it would be the Microsoft Web server, which would be forced to process and filter/drop this attack traffic. So we have also transferred this risk/issue from the bank to Microsoft.

With the above in place, we can now optimize our solution. Microsoft will charge by the minute, for each of the 1,000 Web sites. For 1,000 Web sites, it costs 1,000 times as it would cost for one Web site. Most banks would only be under attack, less than 1 % of the time. So we recommend that when the bank is not under attack, they have just one Azure Web site. When the attack is active, they can have 1,000 Web sites. If the bank wants to save money, where there is an attack, they can instead of ten groups of 100 customers, on ten Web sites. Or they can have one hundred groups of ten customers. For the groups that have an attack, they can then divide the group into ten new groups and redirect the customers to these new groups. For the groups that don't have any attack, they can put these groups back together, in bigger groups.

3 Related Work and Synthesis

We will first present a few comments, concerning the most relevant works and then provide a synthesis, in a table. For the following papers, any of our comments will begin with "**comments:** ".

In [4], we (Booth/Andersson) found a way to mitigate some UDP DDoS reflection attacks. **Comments:** However, if the attackers directly attacked our TCP ports, for the services we were running on each server, we offered no defense.

In [5], we (Booth/Andersson) extended our solution to stop some UDP and some TCP reflection DDoS attacks. **Comments:** However, again, if the attackers directly attacked our TCP ports or directly attacked our UDP ports, for the services we were running on each server, we offered no defense. This paper you are reading now, has continued building knowledge, I.E. improving the mitigation of all DDoS attacks, where our previous papers left off.

In [6], Chonka et al. present that one of the most serious threats to cloud computing itself comes from HTTP Denial of Service or XML-Based Denial of Service attacks. They present their Cloud TraceBack (CTB) solution to find the source of these attacks. **Comments:** Our traceback solution is so much better, since we know the specific customer sub-domain compromised and we have the list of all the non-spoofed source IP addresses, against this specific customer.

In [7], Chung et al. present a way to detect the vulnerable servers, which are used in the DDoS reflection attacks. **Comments:** Our solution simply transfers all reflection attack risks from the Bank to CloudFlare, at no cost.

In [19], Rai and Selvakumar have some up with an algorithm to detect DDoS attacks using the existing machine learning techniques such as neural classifiers. **Comments:** Their problem is that they are analyzing all incoming DDoS attack traffic, together, in one huge messy context. Our solution is much better, since we created an architecture, so that we can analyze incoming attack traffic, against a given customer, in its own customer context. With our approach, it becomes perhaps 1,000 times easier to identify any DDoS attack. In summary, with our approach, we basically have eliminated the usefulness of any, let's analyze all L347 attack traffic, in the global context approaches.

Here are some more of those, let's analyze all incoming DDoS attack traffic, in one huge messy context: [13,24,26,30].

In [29], Yang and Yang propose a new hybrid IP traceback scheme with efficient packet logging to help locate attack hosts which are spoofing their IP addresses. **Comments:** With our contribution, it becomes extremely simple to perform traceback, concerning any attack traffic which reaches the banks' Azure Web process, since the IP address can't be spoofed. However, their solution is perhaps interesting to CloudFlare, since they must defeat the spoofing DDoS attacks (not the bank).

A variety of surveys are available, to help understand the DDoS research topic, such as [2,3,12,15,17,21,23,25,27,31].

In [11], Furfaro et al., propose a DDoS simulator, which can be used to analyze various proposed anti-DDoS algorithms. **Comments:** This should be very useful to WAF vendors to test different anti-DDoS proposed algorithms, before they are put into production.

In [10], Fachkha et al., proposes to characterize Internet-scale DNS Distributed Reflection Denial of Service (DRDoS) attacks by leveraging the darknet space. They empirically evaluate the proposed approach using 1.44 TB of real darknet data collected from a/13 address space during a recent several month period. Their analysis reveals that the approach was successful in inferring significant DNS amplification DRDoS activities including the recent prominent attack that targeted one of the largest anti-spam organizations. **Comments:** It would be interesting for us to implement our proposed solution in the darknet, in addition to using actual beta customers.

In [9], Dietzel et al., study the use of Internet Exchange Points (IXPs)to black-hole DDoS traffic at upstream providers. They find that the research community has been unaware that IXPs have deployed black-holing as a service for their members. Within a 12-week period they found that traffic to more than 7, 864 distinct IP prefixes were black-holed by 75 ASes. **Comments:** Black-holing will also block all valid traffic. In our solution, we have found a way to greatly mitigate any and all L347 attacks, without any required black-holing of valid traffic.

In [28], Yan et al., explore how to defend against DDoS via recent advances in software-defined networking (SDN). They provide a comprehensive survey of defense mechanisms against DDoS attacks using SDN.

In [20], Santanna et al., study Booters, which are DDoS attack platforms as a service, which can be rented, starting at one USD. As a consequence, any user on the Internet is able to launch attacks at any time. In this paper they extend the existing work by providing an extensive analysis on 15 distinct Booters. **Comments:** This is promising since they have an enormous about of actual attack traffic. Once this paper is accepted, we plan to immediately contact them, so that we can analyze how to design will perform against their collected actual DDoS attack traffic.

We'll now analyze the above and other references, via the following specific criteria:

1. Provides strong background, case study and/or survey about DDoS issues?
2. Anti-DDoS Solution?
3. If DDoS solution, can it utilize upstream assistance?

Table 2. Analysis of research categorized by our research criteria categories

Item	Cite	1	2	3	4	Item	Cite	1	2	3	4
0	This paper	✓	✓	✓	✓	14	[18]	✓	✓	✓	
1	[1]		✓	✓		15	[19]		✓	✓	
2	[2]	✓				16	[20]	✓			
3	[3]		✓			17	[21]	✓			
4	[4]		✓			18	[22]		✓	✓	
5	[5]		✓			19	[23]	✓	✓	✓	
6	[6]	✓	✓	✓		20	[24]	✓	✓		
7	[7]		✓	✓	✓	21	[25]		✓		
8	[9]	✓	✓	✓		22	[26]		✓	✓	
9	[10]	✓				23	[27]	✓	✓	✓	
10	[11]	✓	✓			24	[28]	✓	✓	✓	
11	[12]		✓	✓		25	[29]	✓	✓	✓	
12	[13]		✓	✓		26	[30]		✓		
13	[17]		✓			27	[31]	✓			

4. If DDoS solution, does it attempt to remove just the attack traffic, out of line, from authenticated sessions?

We created Table 2, based on our criteria. The citation column (as always) has click-able links to the bibliography. The first item, item 0, is referring to this contribution.

4 Conclusion and Future Work

We have described the research problem as that there are numerous successful DDoS L347 attacks, and that almost all Information Systems are vulnerable. There is an abundance of academic papers, which can detect one type of DDoS or provide mitigation for one type of DDoS. We were unable to find any academic papers or practical solutions, which described a complete, easy to implement, and low cost solution, for organizations who wish to greatly mitigate any and all L347 DDoS attacks, against Web services.

Our hybrid research contribution design filters most of the general attacks, via the free CloudFlare solution. The Microsoft cloud and Microsoft Web server then filters out all of the remaining general attacks. Then within our cloud Web process, we can very easily detect any DDoS and eliminate the DDoS by deleting the attacked sub-domain. We even know which customer is associated with each and every DDoS attack on the Azure Web process.

Our solution will significantly reduce the false positives, as compared to the major anti-DDoS solutions, which are extremely expensive. We can also create lists of known malicious source IP addresses, and share that information with whoever is interested. Our design is extremely low cost and easy to implement solution, which greatly mitigates all of these L347 threats.

Note that this paper is only a conceptual design and the experiment has been left as recommended future work. As future work, we are planning to implement our solution, put it into production, and publish the related case studies. We are actively searching for volunteers, who wish to participate in our experiments. Other future work is to also come up with other similar solutions, for protocols other than HTTP and HTTPS.

References

1. Alwabel, A., Yu, M., Zhang, Y., Mirkovic, J.: SENSS: observe and control your own traffic in the internet. In: Proceedings of the 2014 ACM Conference on SIGCOMM, SIGCOMM 2014, pp. 349–350. ACM, New York (2014)
2. Arukonda, S., Sinha, S.: The innocent perpetrators: reflectors and reflection attacks. Adv. Comput. Sci. **4**, 94–98 (2015)
3. Bhuyan, M.H., Bhattacharyya, D.K., Kalita, J.K.: An empirical evaluation of information metrics for low-rate and high-rate DDoS attack detection. Pattern Recogn. Lett. **51**, 1–7 (2015)
4. Booth, T.G., Andersson, K.: Elimination of DoS UDP reflection amplification bandwidth attacks, protecting TCP services. In: Doss, R., Piramuthu, S., ZHOU, W. (eds.) FNSS 2015. CCIS, vol. 523, pp. 1–15. Springer, Heidelberg (2015)

5. Booth, T., Andersson, K.: Network security of internet services: eliminate DDoS reflection amplification attacks. J. Internet Serv. Inf. Secur. (JISIS) **5**(3), 58–79 (2015)
6. Chonka, A., Xiang, Y., Zhou, W., Bonti, A.: Cloud security defence to protect cloud computing against HTTP-DoS and XML-DoS attacks. J. Netw. Comput. Appl. **34**(4), 1097–1107 (2011)
7. Chung, C.-J., Khatkar, P., Xing, T., Lee, J., Huang, D.: NICE: network intrusion detection and countermeasure selection in virtual network systems. IEEE Trans. Dependable Secur. Comput. **10**(4), 198–211 (2013)
8. CloudFlare. 400gbps: Winter of Whopping Weekend DDoS Attacks. https://blog.cloudflare.com/a-winter-of-400gbps-weekend-ddos-attacks. Accessed 2 May 2016
9. Dietzel, C., Feldmann, A., King, T.: Blackholing at IXPs: on the effectiveness of DDoS mitigation in the wild. In: Karagiannis, T., et al. (eds.) PAM 2016. LNCS, vol. 9631, pp. 319–332. Springer, Heidelberg (2016). doi:10.1007/978-3-319-30505-9_24
10. Fachkha, C., Bou-Harb, E., Debbabi, M.: Inferring distributed reflection denial of service attacks from darknet. Comput. Commun. **62**, 59–71 (2015)
11. Furfaro, A., Malena, G., Molina, L., Parise, A.: A simulation model for the analysis of DDOS amplification attacks. In: 17th USKSIM-AMSS International Conference on Modelling and Simulation, pp. 267–272 (2015)
12. Gillman, D., Lin, Y., Maggs, B., Sitaraman, R.K.: Protecting websites from attack with secure delivery networks. Computer **48**(4), 26–34 (2015)
13. Giotis, K., Androulidakis, G., Maglaris, V.: A scalable anomaly detection and mitigation architecture for legacy networks via an OpenFlow middlebox. Secur. Commun. Netw. **9**, 1958–1970 (2016)
14. Hevner, A.R., March, S.T., Park, J., Ram, S.: Design science in information systems research. MIS Q. Manag. Inf. Syst. **28**(1), 75–105 (2004)
15. Nexusguard: Whitepapers on DDoS Mitigation, Cyber Attack. https://www.nexusguard.com/genius/whitepapers. Accessed 20 Apr 2016
16. Nygren, E., Sitaraman, R., Sun, J.: The Akamai network: a platform for high-performance internet applications. SIGOPS Oper. Syst. Rev. **44**(3), 2–19 (2010)
17. Osanaiye, O.A.: Short Paper: IP spoofing detection for preventing DDoS attack in Cloud Computing. In: 2015 18th International Conference on Intelligence in Next Generation Networks (ICIN), pp. 139–141, February 2015
18. Poulopoulos, L., Mamalis, M., Polyrakis, A.: FireCircle: GRNET's approach to advanced network security services' management via BGP flow-spec and NETCONF. In: 2012 Proceedings of the 28th TERENA Networking Conference (2012)
19. Raj, K., Selvakumar, S.: Distributed denial of service attack detection using an ensemble of neural classifier. Comput. Commun. **34**(11), 1328–1341 (2011)
20. Santanna, J.J., Durban, R., Sperotto, A., Pras, A.: Inside booters: An analysis on operational databases. In: 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 432–440, May 2015
21. Santanna, J.J., van Rijswijk-Deij, R., Hofstede, R., Sperotto, A., Wierbosch, M., Granville, L.Z., Pras, A., Booters; An analysis of DDoS-as-a-service attacks. In: 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 243–251, May 2015
22. van der Steeg, D., Hofstede, R., Sperotto, A., Pras, A.: Real-time DDoS attack detection for Cisco IOS using NetFlow. In: 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 972–977, May 2015

23. Steinberger, J., Sperotto, A., Baier, H., Pras, A.: Collaborative attack mitigation and response: a survey. In: 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM), pp. 910–913. IEEE (2015)
24. Thatte, G., Mitra, U., Heidemann, J.: Parametric methods for anomaly detection in aggregate traffic. IEEE/ACM Trans. Netw. **19**(2), 512–525 (2011)
25. Usha Devi, G., Priyan, M.K., Vishnu Balan, E., Gokul Nath, C., Chandrasekhar, M.: Detection of DDoS attack using optimized hop count filtering technique. Indian J. Sci. Technol. itextbf8(26) (2015)
26. Xiang, Y., Li, K., Zhou, W.: Low-rate DDoS attacks detection and traceback by using new information metrics. IEEE Trans. Inf. Forensics Secur. **6**(2), 426–437 (2011)
27. Yan, Q., Yu, F.R.: Distributed denial of service attacks in software-defined networking with cloud computing. IEEE Commun. Mag. **53**(4), 52–59 (2015)
28. Yan, Q., Yu, F.R., Gong, Q., Li, J.: Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: a survey, some research issues, and challenges. IEEE Commun. Surv. Tutor. **18**(1), 602–622 (2016)
29. Yang, M.-H., Yang, M.-C.: RIHT: a novel hybrid IP traceback scheme. IEEE Trans. Inf. Forensics Secur. **7**(2), 789–797 (2012)
30. Yu, S., Zhou, W., Jia, W., Guo, S., Xiang, Y., Tang, F.: Discriminating DDoS attacks from flash crowds using flow correlation coefficient. IEEE Trans. Parallel Distrib. Syst. **23**(6), 1073–1080 (2012)
31. Zargar, S.T., Joshi, J., Tipper, D.: A survey of defense mechanisms against distributed denial of service (DDoS) flooding attacks. IEEE Commun. Surv. Tutor. **15**(4), 2046–2069 (2013)