



Hacker Intelligence Initiative, Monthly Trend Report #12

Denial of Service Attacks:

A Comprehensive Guide to Trends, Techniques, and Technologies

ADC Monthly Web Attacks Analysis, September 2012

1. Executive Summary

On hacker forums, denial of service remains the most discussed topic. Hackers continue to develop tools to optimize this attack method. Why? Distributed denial of service (DDoS) attacks does not seek to breach data integrity or privacy; they can be conducted without the requirement of identifying vulnerabilities to exploit the application. This report catalogs the latest trends, techniques and technologies deployed by hackers and gives security professionals specific steps to mitigate the threat.

- › *An Overview of Application DoS: Though not new, application DoS does not address the network resources, or the bandwidth to the service, but instead focuses on the Web application and the server itself. Application DoS may be directed at specific flavors of Web servers (e.g., IIS or Apache) or to specific applications (such as SharePoint) by understanding backend architecture.*
- › *Analysis of Hactivist Operations: Examining real-world incidents from OpColumbia, OpBahrain and OpRuskin, Imperva discusses how and why hactivists utilize DDoS to support its goals and promote a message.*
- › *Technical Tool Group Review: Imperva research highlights the most popular DDoS tools, including LOIC, SlowHTTPTest, and railgun to demonstrate how black-hat hackers conduct attacks by utilizing white-hat testing tools.*
- › *Detection and Mitigation Techniques: Imperva discusses the characteristics at the essence of DDoS attacks and attack-specific characteristics, so that information security professionals can arm themselves with the knowledge required to identify and protect against DDoS attacks.*

2. Overview: The Evolution of DoS

A Denial-of-Service (DoS) attack is a general name for any kind of attack against data availability. In the Web application world, a DoS attack aims to “take down” the site in order to make it inaccessible to its users. This may cause some serious financial damage to the site, both directly and indirectly by damaging its reputation.

DoS attack is mostly conveyed by depleting some of the system’s limited resources. As a Web server owns many different resources, and there are numerous ways to consume each of them, there are many different ways to create such attack, and the term DoS attack does not define the technical characteristics of the attack. Some prominent examples of DoS attacks are:

- › IP (Layer 3) attacks on the network bandwidth
- › TCP (Layer 4) attacks on the server sockets
- › HTTP (layer 7) attacks on the Web server threads
- › Web application (layer 7+) attacks on CPU resources

Over the last years, attackers move their DoS attacks up the stack and into the Web application layer in order to decrease costs, as Web app DoS is more efficient and avoids detection as many anti-DoS solutions are traditionally focused on lower layers.

However, DoS attack have some very different characteristics from other popular Web application attacks types, such as SQLi, RFI, or XSS, that target data integrity and privacy:

- › Data integrity and privacy attacks require a vulnerability in the application that can be avoided or patched by secure coding, while DoS attacks leverage the inherent limitations of the application as system resources are always finite.
- › Data integrity and privacy attacks try to remain as covert as possible, in order to maximize the monetary gain while remaining undetected, while DoS attacks are usually self evident by nature.

A way to increase the efficiency of a DoS attack, while evading detection and blocking, is to split the attack load among numerous machines simultaneously. Such coordinated attacks are called Distributed Denial of Service attack, or DDoS, and we believe are the most effective form of DoS today. DDoS attacks are often executed using massive botnets and compromised servers. As we describe in detail later, lately DDoS attacks are also carried out by convincing volunteers from around the world to contribute their own machines (PCs or mobile) to the cause.

And hackers like DoS. In our hacker forum study from October 2011, we observed that 22% of discussions focused on DoS, slightly higher than SQL injection at 19% of all discussions.

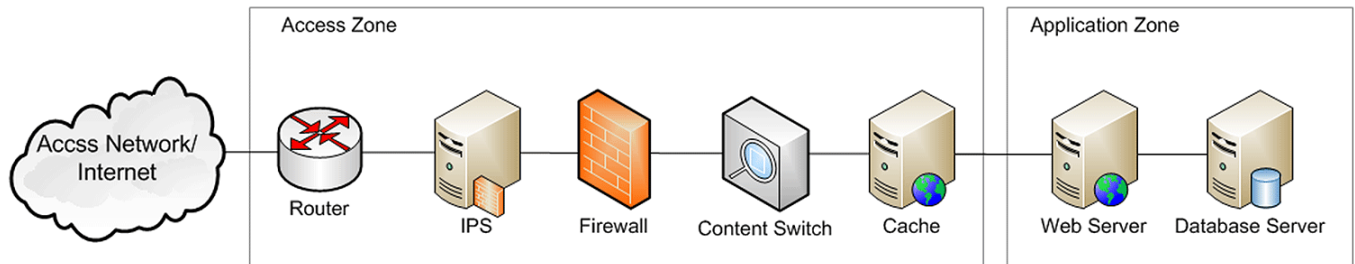
The DDoS attacks that gain most of the public attention are usually those carried out by hacktivist groups for ideological reasons. These operations are intended to get the crowd’s attention, so they are usually accompanied by public announcements that state the cause, the targets, the tools to use, and the date of initiation. We describe such operations later on in this document.

DDoS attacks are not limited to hacktivists attacks and might also be used for business reasons, for example, to take down the site of a competitor. Another related phenomenon is the use of “blackmail” DDoS, as hackers can threaten a Website owner to take down the site unless paid.

It is important to note that many DoS and DDoS tools are freely available online and, as with any technology, are not confined to their creator’s original intention. Once a tool is published, it can be used by anyone against any target. We have encountered such cases where tools originally created for a hacktivist causes, or even for security testing, were later used for different reasons by other attackers.

Recently, hackers have focused much more on application DoS. Though not new, application DoS does not address the network resources, or the bandwidth to the service, but on the Web application and the server itself. Application DoS will be directed at specific flavors of Web servers (e.g., IIS or Apache) or to specific applications (such as SharePoint) by understanding backend architecture.

The following diagram demonstrates a logical schematic of a Web service with most of the common components that are usually present in an enterprise application datacenter:



Network DoS resides in the logical “Access Zone” whereas application DoS resides mostly in the “Application Zone,” that includes the Web front end and the data store for it. The building blocks of a successful application DoS attack are first to bypass ALL of the Access Zone controls in place, address a weakness on the Application Zone, and then execute a payload that communicates directly with the Web server to hit the application or server. All building blocks are usually wrapped inside a tool, for the Hacktivist’s ease of use.

3. Detailed Analysis

In this section, we describe some of the DDoS tools observed in our data. We begin with a technical description of the tool and its characteristics and then go on to the observed attacks and how were they identified in the wild. Our focus is DoS attacks abusing network resources (bandwidth, connections, etc.). When possible, we try to support our data with related publications and give a bigger picture of the sequence of events behind an attack.

3.1 Mobile LOIC

LOIC stands for Low Orbit Ion Canon, an open source denial-of-service attack application, originally written in C#. LOIC was used in many hacktivist operations, like Project Chanology against the Church of Scientology, Operation Payback against credit card companies, and Operation Megaupload. The use of Mobile LOIC as an integrated part of an attack was described in detail in Imperva’s analysis of the anatomy of an Anonymous attack.¹ In general, using LOIC attack tool enables anyone to participate in the attack; no special resources or technical knowledge needed.

Originally, LOIC tools were applications that users downloaded and installed on their computer. Such programs send bogus requests to the target, thus contributing to the attack. When a large number of coordinated participants direct the application at the same target, it can be overwhelmed.

The introduction of Mobile LOIC has made participation in a DDoS attack even easier. The DoS is initiated by the user browsing to an attack page. The page contains the attacking code, which is written in JavaScript and is automatically downloaded to the user’s browser and executed. The script iterates endlessly and generates a new image attribute. The source of the image is the victim’s Web page. Doing this creates multiple requests to the victims’ Website as the page is rendered by the browser. As long as the page is open on the user’s browser, the browser continues to send the requests. This makes the attack tool more accessible, as it can operate from any platform. Furthermore, traditional LOIC required the user to download and install a program, something not all users would be willing to do. In mobile LOIC, no download is needed, and this may contribute to non-hacker-volunteers’ feeling that by participating they don’t put themselves at risk. The feeling that participation is safe and the simplicity of the process may both facilitate willingness to participate. The tool’s popularity can be verified by a Web search that reveals hundreds of mobile LOIC pages.

¹ http://www.imperva.com/docs/HII_The_Anatomy_of_an_Anonymous_Attack.pdf

The effect of a DoS attack depends on the coordination between its participants, since only together can they expend more resources than the target site has. Anonymous became famous for its ability to harness large groups of volunteers to take part in such attacks based on ideological reasons. However, according to the recently published book *We Are Anonymous*,² at least in the first successful Anonymous DDoS attacks, it is estimated that 80-90% of the malicious traffic was generated by 1-2 big botnets, and just 10-20% of the traffic was generated by Anonymous volunteers using LOIC.

3.1.1 Description/ Technical Analysis

We have observed traffic generated by several mobile LOIC DoS. In all these cases, the malicious HTTP requests had a URL in the HTTP Referer header, directing to the attacking page. The pages are very similar in content and functionality, and vary mostly in terms of graphic design and formatting. Normally, the user can select three things on the page:

1. Target URL
2. Number of requests per second
3. "Propaganda" message. This message will be inserted as a HTTP parameter in the requests, and will eventually show up in the attacked site's log of the traffic. The default messages we saw were the infamous Anonymous slogans "We hope this has your attention" and "We Are Legion."

The page usually includes a counter of sent requests and their status, giving the user an immediate feedback on their contribution to the attack.

3.1.2 Case Studies

3.1.2.1 OpColombia

As described in <http://www.cyberwarzone.com/cyberwarfare/opcolombia-target-list-released-anonymous-europe>, Anonymous announced "OpColombia" last April, claiming: "Colombia's government passed a law that violates freedom on the Internet without the people's opinion."

As we have described in our Anonymous report,³ Anonymous uses crowd-sourcing to accumulate bandwidth resources and to enhance the impact of its DoS attacks. For Operation Colombia, the group followed its practice of preparing a set of Web pages, each one with a specific target site.

A participant in the attack has merely to surf to one of these pages. The script embedded in each page automatically starts flooding the target site with bogus requests, originating from the participant's browser. At the same time, the page gives the participant visual feedback on his contribution to the global attack.

These DoS attacks were identified in the network traffic Imperva monitors, so we can confirm that the attack generated a large volume of traffic against multiple sites. It seems that the attack has some success in taking down its targets.

Furthermore, a list of all Anonymous pages used in the attack, mapped to their targets, was published online. Among the specified targets were many government sites. Some examples are the Colombian army, district attorney's office, president's office, senate, ministry of justice, Summit of the Americas, etc. (Summit of the Americas is a conference of the heads of states of the Americas, the last one was held in Colombia last April.) As Figure 1 demonstrated, the pages used in OpColombia were hosted by <http://pastehtml.com>, a free anonymous Web hosting service. We used this URL as a fingerprint to find other mobile LOIC attacks in our data. Pastehtml is a free service and can be used by anyone for any purpose, so having it in the referrer header is not necessarily a sign for malicious activity. Hackers can use many other services to host mobile LOIC pages, so attention to the referrer header is important.

² *We Are Anonymous: Inside the Hacker World of LulzSec, Anonymous, and the Global Cyber Insurgency*, by Parry Olson, Little, Brown & Company, 2012.

³ <http://www.imperva.com/download.asp?id=312>

```
Fiscalia: http://pastehtml.com/view/[REDACTED].html
Ejercito: http://pastehtml.com/view/[REDACTED].html
Presidencia: http://pastehtml.com/view/[REDACTED].html
Presidencia2: http://pastehtml.com/view/[REDACTED].html
Senado: http://pastehtml.com/view/[REDACTED].html
Ministerio de Justicia: http://pastehtml.com/view/[REDACTED].html
Camara: http://pastehtml.com/view/[REDACTED].html
Visepresidencia: http://pastehtml.com/view/[REDACTED].html
MinTransporte: http://pastehtml.com/view/[REDACTED].html
Santos Presidente: http://pastehtml.com/view/[REDACTED].html
Cumbre de las America: http://pastehtml.com/view/[REDACTED].html
summit-americas: http://pastehtml.com/view/[REDACTED].html
```

Figure 1: The published list of target organizations with their mobile LOIC attack page.

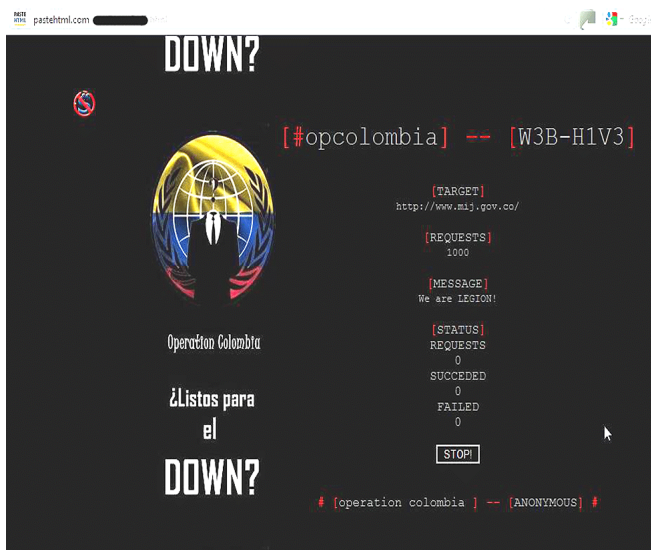


Figure 2: A screen shot of an attack site in action ("ready for the DOWN?").

The attack seemed to be successful, at least in part, as some Colombian government sites (like <http://www.mij.gov.co/>, Ministry of the Interior) were actually taken down:

It's not just you! <http://www.mij.gov.co> looks down from here.

3.1.2.2 OpBahrain

The Anonymous announcement about the operation was on April 22, about a week after the announcement about OpColombia. Here the accusations are graver:

Anonymous has watched with growing concern the violations of human rights in the Kingdom of Bahrain. We have seen thousands of innocent demonstrators brutally repressed by the regime...And we suffered from the mainstream media blackout about what is really happening in Bahrain...Anonymous demands the Government of Bahrain to put an immediate end to torture and stop the use of force and violence against peaceful demonstrators.



Figure 3: A screenshot from the YouTube announcement⁴ about OpBahrain.

A day after the announcement, we observed the attack in our monitored network. The attack characteristics were similar, but not identical to what we saw in OpColombia.

Several IPs were observed as targets, all belonging to central government and financial sites. Once again, the requests carry the Anonymous message, which leaves no room for doubt regarding the identification of the attack: "We hope this has your attention..."

The referrer header contains the URL of the attacking page.

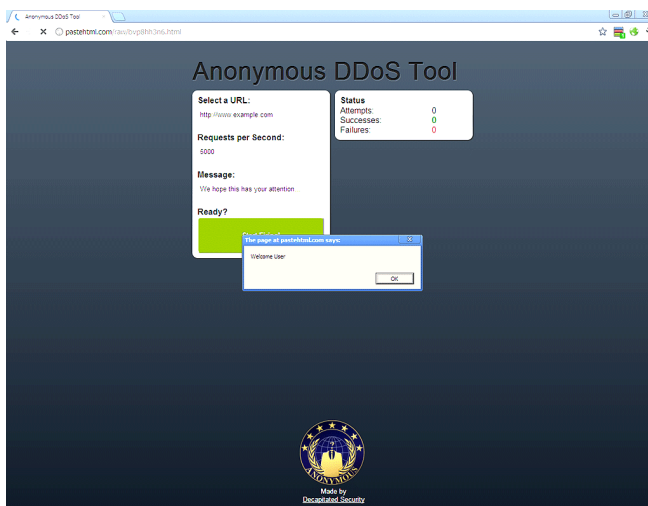


Figure 4: The attacking page for OpBahrain

⁴ <http://www.youtube.com/watch?v=PtZLyZztVGg>

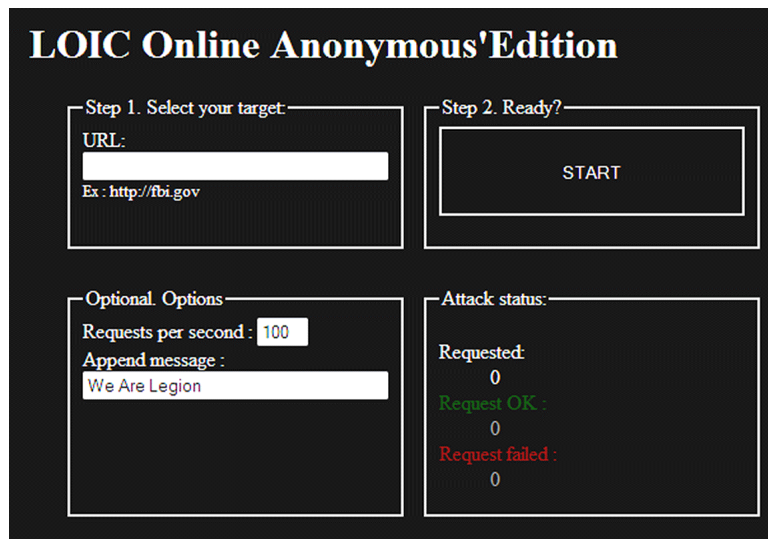
3.1.2.3 OpRussia

Another attack was observed on May 6, 2012 that targeted one of the main Web sites of the Russian government. This was the day of the elections for the Russian president, which were accompanied by extensive public debate and opposition. The official announcement about OpRussian, also called OpDefiance, specifically stated this site as the main target, including the exact start time of the attack. Other Russian sites were attacked as well during the following week, and are discussed below.

RUSSIAN SITES #opRussia [http://\[REDACTED\]/opRussia](http://[REDACTED]/opRussia)
<http://www.government.ru/> DDOS attack May 6, 2012 in 15-00! IMPORTANT!! PRIORITY GOAL!

The requests had similar but not identical characteristics to what we saw in OpColombia and OpBahrain. The attack page's user interface is different in design than the ones seen previously, but its functionality remains the same.

What's unique here in comparison to the previously discussed operations is that the used online Loic is a relatively old one (from January 2012).



LOIC Online Anonymous' Edition

Step 1. Select your target:
 URL:

 Ex : <http://fbi.gov>

Step 2. Ready?

Optional. Options
 Requests per second :
 Append message :

Attack status:
 Requested:
 0
 Request OK :
 0
 Request failed :
 0

Figure 5: The attacking page for OpRussia

About a month later, we saw the URL of this page in the referrer header of requests directed at completely other targets. We also saw a DDoS attack on the Website of a boarding school, which was generated by one of the mobile LOIC pages of OpBahrain. As can be seen from the pages above, both the target URL and the message can be easily changed by the user. This means that mobile LOIC pages, once published, can be used by anyone against any target, and do not necessarily serve the original purpose of their creator. Such pages are like abandoned weapons lying in the open, waiting to be picked up.

3.2 Slowhttp

3.2.1 Description/ Technical Analysis

Slowhttptest is an open-source tool that implements several kinds of DoS attacks.⁵ According to the tool's development site: SlowHTTPTest is a highly configurable tool that simulates some Application Layer Denial of Service attacks. It implements most common low-bandwidth Application Layer DoS attacks, such as **slowloris**, **Slow HTTP POST**, **Slow Read attack** by draining concurrent connections pool, as well as **Apache Range Header** attack by causing very significant memory and CPU usage on the server.

⁵ <http://code.google.com/p/slowhttptest/>

Although its official intention is legitimate, testing the tool's availability online makes it useful for black-hat hackers as well. Without modifications, the tool is easy to identify because all the requests' Referer headers direct to the tool's development site.

As described on its site, slowhttptest implements three DoS methods:

Slowloris is a DoS tool based on the concept of keeping the server busy with very few resources. Instead of flooding the server with requests, it holds the connections open for a very long time. To do so, a Slowloris client sends partial HTTP requests, and continues to send subsequent headers at regular intervals to keep the sockets from closing. Slowloris must wait for all the application's sockets to become available, so other users of the system must finish their sessions before the sockets become available for Slowloris to consume.⁶ Eventually, the server's connection pool will be entirely busy processing Slowloris requests, and it will start denying new connection attempts from legitimate clients.

Slow HTTP POST method is very similar to Slowloris. Here, the requests are HTTP POST requests, in which all the headers are sent correctly, including the Content-length. After the headers are sent and received, the POST message body is sent at a very low rate, thus keeping the connection open for a prolonged time. The server has to wait until all content arrives according to the declared Content-length. Unlike Slowloris, Slow HTTP POST can't be mitigated by load balancers.⁷

Slow Read works the other way around. Instead of pushing data slowly to the server, the client forces the server to send a large amount of data, which it accepts at a very slow rate. Upon establishing the connection, the attacking client declares a very small receive-window-size. This makes the server split the response to many small pieces that would fit the buffer size, resulting in very slow ongoing responses. A possible way to mitigate such attacks is to configure the server to ignore connection requests with abnormally small window sizes.

3.2.2 Case Studies

3.2.2.1 Patriarchia

During the last week of March 2012 and the beginning of April 2012, we observed a DoS attack on Patriarchia, the official site of the Moscow Patriarchate – the Russian Orthodox Church. This attack used the Slowhttp tool.

There were two related destination IPs, one of which seems to be a version for mobile browsers. On each attacked IP, there were two attacked URLs: '/' and '/search/'.

The 15,000 accesses to '/' were made using POST method, with content length of 4096. There was a referrer header with this value: <http://code.google.com/p/slowhttptest/>, which directs to the tool's Website and serves as an identifier.

The 13,000 accesses to '/search/' were made using the GET method. These requests didn't have a referrer header, but had a parameter (probably for the search) with a random string (e.g. oNsNDTdC). The random parameter ensured that the client's requests reached the server, bypassing any caching between them.

⁶ <http://hackers.org/slowloris/>

⁷ <http://www.darkreading.com/vulnerability-management/167901026/security/attacks-breaches/228000532/index.html>

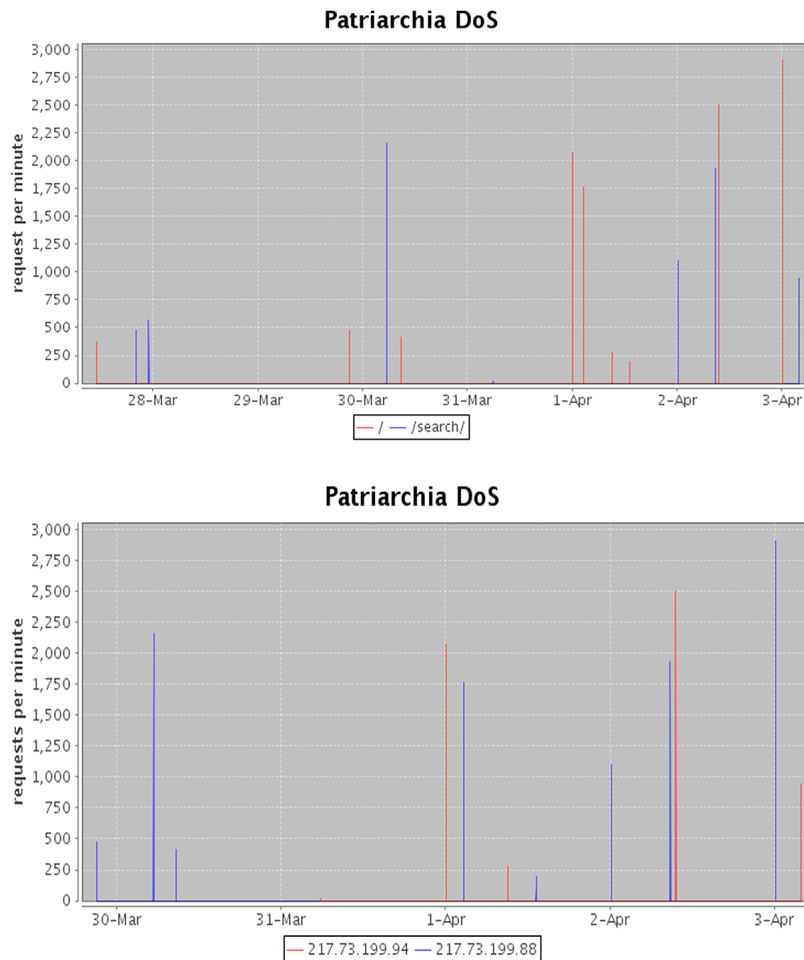


Figure 6: Number of requests per minute for each target IP and each URL

3.2.2.2 OpRussia

The week of May 7, 2012 was the week of the Russian elections. During this time, the LOIC attack described above and we observed two more attacks against Russian government sites. Both were DoS attempts using the slowhttp tool. This was also a part of Anonymous' struggle against the re-election of Putin for president.⁸ The attack on one of the targets continued until the end of May.

3.3 railgun

Like slowhttptest, this is an open-source tool.⁹ Its availability online makes it useful for black-hat hackers.

3.3.1 Technical Analysis

During last May, we have encountered four different attacks, each against a different target, which had similar characteristics. All the attacks' requests had similar features: for one, the Content-Length was set to 10,000, which is very distinctive – both because it is large (for a non-multipart post) but mainly because it's a very "human" number. Other than that, the requests had no Accept Headers, and the User Agent was changed between requests, probably randomly from a list. The parameters from the URL appear to be random to prevent caching of the responses.

Using these characteristics, we were able to locate the tool probably used to generate these attacks. The attack tool appears to be Railgun: <http://code.google.com/p/railgun/> (open source)

⁸ http://www.theregister.co.uk/2012/05/10/anonymous_kremlin_ddos_putin/

⁹ <http://code.google.com/p/railgun/>

From this tools description:

This tool is to demonstrate the “Slow HTTP POST” and “Slowloris” DoS vulnerabilities in various Webservers. It is hoped that our tool will allow administrators to stress test their servers to see how easy it is for a remote attacker to launch Denial of Service attacks against them.

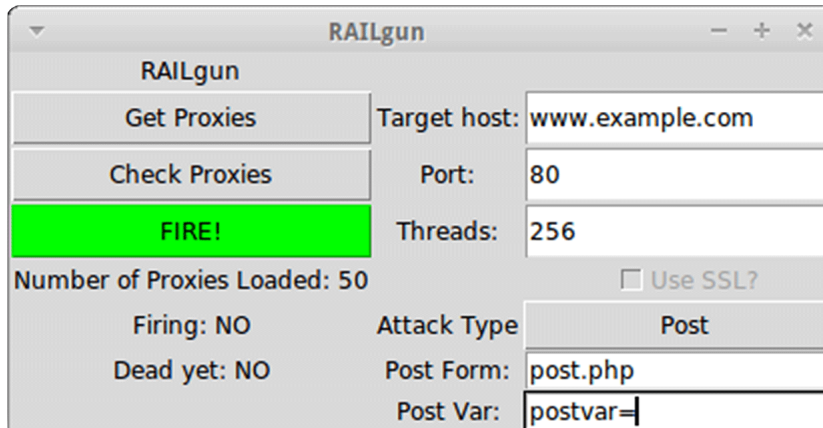


Figure 7: A screenshot of Railgun DoS tool.

We have analyzed some of the changes the code underwent in its latest versions. As seen in our data, the tool changes user-agent in order to escape detection. To do so, the earlier version used a list of 20 user agents, from which one was chosen randomly for each request. In the newer versions, this list was replaced by a far larger one – a 10MB text file containing more than 70,000 different User-Agents. This of course improves the tool’s ability to go undetected. Another important change is the addition of Accept headers to the requests. Accept headers are a part of a normal browser’s behavior and their absence can often be used as a way to identify automated attacks.¹⁰ Previously, requests produced by Railgun had no such headers, and could therefore be marked as suspicious automation. With the new version, the requests have browser-like headers with normal appearing values:

- › **Accept:** text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
- › **Accept-Language:** en-us,en;q=0.5
- › **Accept-Encoding:** gzip,deflate
- › **Accept-Charset:** ISO-8859-1,utf-8;q=0.7,*;q=0.7

Another addition to the tool is its ability to switch easily between DoS methods, and change request content according to the selected method – Slowloris or Slow HTTP POST.

Lastly, the content length was changed to 6,000 instead of 10,000. For now this seems to be a constant value and may therefore be used as a fingerprint to identify this tool.

3.3.2 Case Studies

We have seen attacks with these identifiers to four different targets. The targets don’t seem to be connected to each other, although two of the attacks occurred simultaneously.

On March 27, 2012, we saw an attack on a site offering religious Christian support. A month later, another attack with similar characteristics was seen on a financial services company.

Later on, on May 23, 2012, a simultaneous attack took place on two targets, of which, one is a Chinese microblogging service.

¹⁰ <http://www.imperva.com/download.asp?id=360>

The attack on both targets lasted less than two minutes, during which each target suffered a couple of hundred requests. The graph below shows the number of requests per second for both targets. The attack process seems to be almost identical.

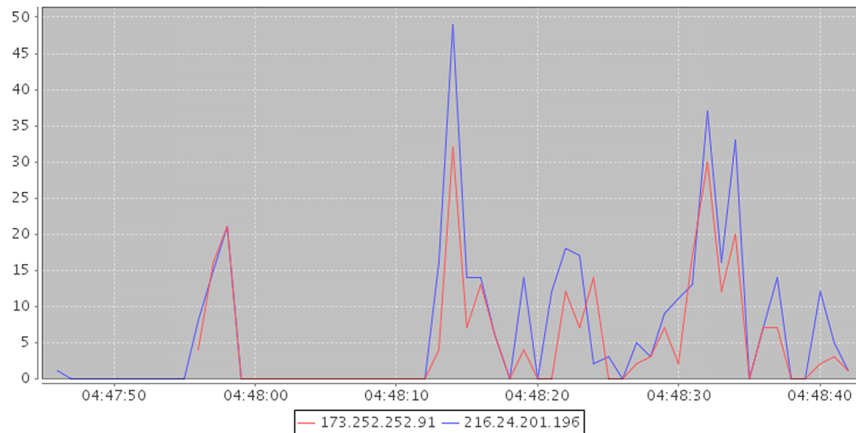


Figure 8: Requests per second for the two targets.

At least two of the targets responded with Service Unavailable, i.e., the attacks succeeded at least partially. They all have since recovered.

3.4 More tools

Like in any other field, the hacking arena is a competitive one, in which ego often plays an important role. It is not uncommon to come across security researchers and enthusiasts who not only develop and publish open-source hacking tools, but also take the effort to decompile and publish the weaknesses of their competitors' tools. A war is waging between those who consider themselves "true" security experts, and "skiddies" (also referred to as script kiddies, skidiots, etc.). Executing a DoS attack is relatively easy and requires very little technical knowledge, relative to other attacks, aimed at obtaining specific data or changing the server's behavior. A DoS attack has very little requirements – its only aim is to occupy the server's resources by any means and thus make it unresponsive to users. Therefore, there is a relatively high amount of simple, brute force, DoS tools available freely online. The more sophisticated ones are those that also take precautionary measures against detection, distribute the attack among proxies, and use wisely and efficiently the server's resources. Table 1 summarizes the features of some freely available DDoS tools.

Tool name	DDoS method	Accept Headers	Fingerprint	Evasion Techniques
Are-You-Dead-Yet ¹¹	Slow POST	None		
Tor's Hammer	Slow POST	None	Content Length= 10,000	Random User-Agent
Simple Slowloris Flooder	Slowloris	None	Content Length= 5,235	
Nuclear DDoSer	Slowloris Slow POST	Accept Language, Accept Charset	data= nuclear ddoser	Changing Proxies
Dirt Jumper	HTTP flood SYN flood POST flood and more	None		Random User-Agent

Table 1: A sample of DDoS tools and their characteristics

Dirt Jumper - Dirt Jumper is a family of DoS tools. Dirt Jumper itself has more than 5 versions, all freely available online. Its availability lead to the development of many similar tool with minor modifications that go by other names, like RussKill, September, Simple Di Botnet and Pandora DDoS. The different versions vary in the DoS methods they implement and in their stealthiness. Most versions implement at least three DoS methods, including HTTP flood, SYN flood and POST flood.

¹¹ <http://code.google.com/p/r-u-dead-yet/>

3.5 Detection

As we described earlier, DDoS is a general name for many types of attacks whose goal is to deplete the victim's resources. Therefore, there's not a single mandatory string or command that has to appear in all the requests. On the contrary, the requests in themselves might look like ordinary HTTP. We list several characteristics of DoS requests we identified in our data, but it is important to keep in mind that some are not a mandatory part of the request and can be changed at any time without influencing the effectiveness of the attack. Therefore, we suggest distinguishing between two types of DoS characteristics:

- › **Characteristics at the essence of the attack:** For example, detecting Slowloris attack by its low rate. The low rate and long period between headers are an integral part of the attack and can therefore be used for detection. Another example is the referrer header in Mobile LOIC attacks that contains the URL of the attacking Webpage. As demonstrated above, mobile LOIC pages can be reused against more than one target, so having a list of malicious referrers might also be beneficial. The referrer can also be used to identify tools like slowhttptest.
- › **Attack-specific characteristics:** This category includes typical values and parameters used by DoS tools. These are not mandatory and can be changed with time. It is important to keep track of these changes, as they can be useful in detecting an attack and preparing mitigation for future attacks. The "msg" parameter observed in many mobile LOIC attacks is a good example. This parameter delivers the propaganda message of the campaign, and often includes words like "Legion," "Anonymous" and "Expect us." The name of the parameter itself can change, so looking for such code words in any HTTP parameter can serve as simple fingerprints. Another example is constant values, hard-coded into the DoS tool. For Railgun, we used the constant Content-Length as a fingerprint, but, as we've seen later, this value had already changed. Like slowhttptest, other open source tools developed by white hats might be designed to include identifiers with links to the code. When such tools are abused for malicious purposes, it depends on the hacker's skill and awareness to change the incriminating details.

As can be seen in Table 1, the requests generated by many freely-available DDoS tools have very little content. Some don't have any Accept headers, so they can be detected using anti-automation methods.¹² Anti-automation includes the detection of traffic without browser-like headers, with abnormal User-Agents and in high rate, and can catch a considerable amount of DoS traffic. The more sophisticated DDoS tools use various techniques to avoid such detection, like randomly changing User-Agents and sending normal-appearing headers. Another way to identify automated tools is by analyzing the headers order. While the headers and their payloads are valid, their order may be abnormal and not resemble the behavior of a normal browser.

3.6 The Economy of DDoS

Taking down a site can be monetized by malicious attackers in several ways, such as by extortion ("pay me or I'll take down your site"). In the same vein, DoS attacks have also become industrialized, and can be purchased as a service from professionals. You don't need to be a hacker or part of an ideological movement to want to take down a site. Today, all you have to do is pay someone who'd take it down for you, for as much time as you like (and are willing to pay for). This service can be purchased online from several groups.

One such site we had observed offers various services for hackers, of which DDoS is only one.

The sales pitch for the DDoS service is convincing: take down the site of your business competitors. They offer really good prices with several ways to pay the bill. Moreover, the price is negotiable and depends on the target and the difficulty to take it down. Lastly, like any honest worker, they complain that their services and hard work are being abused by scammers and defrauders.

¹² http://www.imperva.com/docs/HII_Automation_of_Attacks.pdf



Figure 9: "Any DDOS services 24/7."

Attention! In view of the widespread fraud Ddos-orders, ddos service, do the test only after the transfer of money under the protection code, in order to combat resellers and scammers.

The minimum cost of services:

- 12:00 - \$ 35
- 24 - \$ 50
- Sunday - from \$ 320
- 2 weeks - from \$ 600
- Month - \$ 1000
- Warning 3.06.2012 updated botnet attacks now are even more powerful, more versatile, more efficient!

Note: The course for each project - individual price. Above are the minimum prices of **DDoS attacks**. If the administration site attacked supply protection - price increase due to the resources required to attack that, in general it is logical.

We accept:

Web-mani.

 WebMoney

Yandex-money.

 Яндекс
деньги

Liberty-Reserve.

 Liberty Reserve

Also, there may be other ways to pay as agreed.

Figure 10: Terms and prices for purchasing a DDoS attack

Another DDoS service is Gwapo. Their prices range from \$5 per hour for small business sites to \$50 per hour for larger and better-protected sites. They only accept anonymous payment via Liberty Reserve/ Bitcoins and MoneyBookers. Strangely enough, they use some aggressive advertising method, publishing themselves boldly on YouTube and not, like most services, in the hidden pages of hacker closed communities. Forums seem to be the most common way to spread the word of a newly available DDoS service.

[CHEAP] DDOS Service [2\$ /Per Hour]	Thread Options
<p>12-01-2011, 02:54 PM (this post was last modified: 12-23-2011 06:57 PM by Wajid)</p> <p>DDoS SERVICE PROVIDER</p> <p>ddosdoesnotexist... ★★★★★</p> <p>Posts: 280 Joined: Sep 2011 Vouch:</p>	<p>Post: #1</p> <p>Cheap DDoS service</p> <p>supported methods of attack:</p> <ul style="list-style-type: none"> - UDP Flood - SYN Flood - HTTP GET Flood - HTTP POST Flood <p>prices for attack 1 target:</p> <ul style="list-style-type: none"> - 4\$ / hour - 35\$ / day - 200\$ / week <p>* prices may be change, if target have Anti-DDoS protection!</p> <p>payment:</p> <ul style="list-style-type: none"> - MIZ - Liberty reserve <p>contact me:</p> <p>Skype : [REDACTED] 123345</p> <p>AIM : [REDACTED]@email.com</p> <p>ICQ : [REDACTED]</p> <p>MSN : [REDACTED]@hotmail.fr</p>
<h3 style="text-align: center;">CHEAP PROFESSIONAL DDOS SERVICE</h3> <p style="text-align: center;">Cheap Professional DDoS Service Trusted Strong/Fast Service Takes down Large Website/Forum/Game Servers etc. No time limit</p> <h3 style="text-align: center;">PRICE</h3> <p style="text-align: center;">1 - 4 hours / 2\$ per hour 12 - 24 hours / 4\$ per hour 24 - 72 hours / 5\$ per hour 1 month / 1000\$ fix price</p> <h3 style="text-align: center;">PAYMENT ACCEPTED</h3> <p style="text-align: center;">Paypal (Verified users only) Liberty Reserve Western Union</p>	

Figure 11: Ads publishing a DDoS services with detailed prices and payment methods

Another ad we found in a forum offers higher prices for DDoSing a target – \$5 for only 500 seconds of take down. Unlike the other services, this provider accepts payment by Paypal. And, as always when purchasing a service online, recommendations by former customers are everything:

Originally Posted by **Lunatic**
This guy took the site down for a long ass time for me... Hes legit.

As seen in all these examples, DDoS providers offer convenient prices and various payment methods, which keep both themselves and their customers anonymous.

4. Summary and Conclusions

- › DoS attacks usually abuse the inherent application limitations and do not require a vulnerability in the app code. Therefore:
 - We believe every site is a potential victim
 - A DoS attack cannot be mitigated only by secure coding of the application or having an SDLC process
- › DoS attacks are technically broadly available
 - Many attack tools are freely available
 - Some parties offer DoS attack as a service
- › DoS is moving up the stack and into the application layer. Application owners should verify that their DoS protection is relevant against application layer DoS attacks. Specifically check for SSL decryption and having elaborated HTTP parsing and rules creation ability.
- › Detecting and mitigating Web application attacks
 - Identify and block known threats – Most attack tools have some unique HTTP characteristics that can be extracted and provide a basis for detection.
 - Acquire reputation about attack sources – Many of the attacks are delivered via infected users and proxies. Acquiring data on such in advance is helpful in deflecting a large portion of the attack volume.
 - Stop automation – DoS attacks are highly automated. Identifying automation (e.g., by detecting missing headers) and blocking it would stop the DoS attack.
 - Have a stateful Anti-DoS rule engine which is able to define rules that take repetition into account. Most DoS attacks' HTTP requests may appear to be benign individually. Only by analyzing them on the context of the whole session may reveal the abnormal repetitions that constitute the attack. The repetition should be counted in the relevant context (IP/session/ user) and on any HTTP verb (e.g., URLs, headers, parameters.)

Hacker Intelligence Initiative Overview

The Imperva Hacker Intelligence Initiative goes inside the cyber-underground and provides analysis of the trending hacking techniques and interesting attack campaigns from the past month. A part of Imperva's Application Defense Center research arm, the Hacker Intelligence Initiative (HII), is focused on tracking the latest trends in attacks, Web application security and cyber-crime business models with the goal of improving security controls and risk management processes.