adform

Adform Fraud Protection

How Adform Discovered HyphBot

One of the Largest Botnets to Ever Hit Digital Advertising

22/11/2017

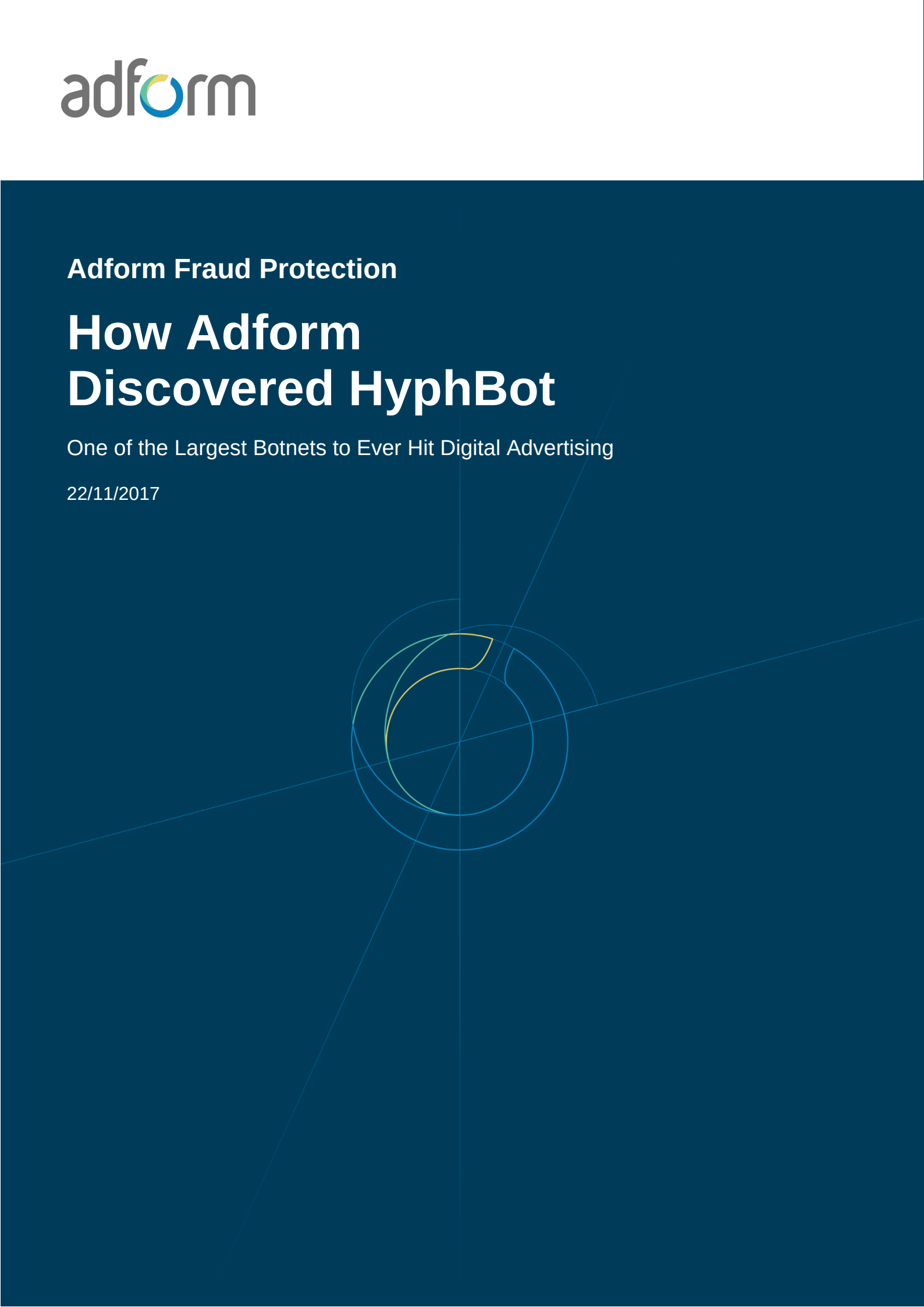


Table of Contents

Introduction

At Adform, our main objective is to build a secure and effective full stack of Advertising Technology products (DSP, Ad Server, DMP). Our suite of products are not only transparent but continue to deliver impactful advertising. As such, keeping our platform - and the ecosystem - free from fraud is one of our most important objectives and we invest the time and resources of some of our smartest people to achieve this.

Our fraud team have made a major discovery and have identified what is potentially the biggest bot network ever to hit our industry. It seems to be between 3-4 times the size of the famous Methbot network that was discovered by WhiteOps (see here) around 11 months ago and targets premium inventory. Adform is happy to inform our clients that the full HyphBot impact on all Adform's platforms was extremely limited, costing less than $1,000 USD per month.

Over the past eight weeks we've worked hard to fully investigate HyphBot and its structure and are now able to share our findings. All the details have been cross-checked by external experts (Shailin Dhar, Director Research at Method Media and Dr. Neal Richter, Co-Chair of the IAB Tech Lab) who have verified our discovery and confirmed that it is in fact fraud by reviewing our methodology and results. As part of this external verification, Neal and Shailin received full disclosure of our data and analysis. The granularity of data went far beyond what we are publishing in this white paper.

With this paper we are now handing out all necessary information that partner SSPs / DSPs etc. need in order to learn and improve their own approaches and eliminate not only HyphBot but also safeguard against similar networks. We are releasing these details with the clear goal of further enhancing trust and transparency in Digital Advertising and setting the bar at the highest possible level across all connected platforms. At the same time we are limiting the information in this white paper to ensure fraudsters cannot reverse engineer our blocking filters.

Summary

HyphBot, which was identified using our in-house algorithms and expert analytics, appears to be one of the largest bot networks to be discovered in Digital Advertising. Significantly, it has also included a large amount of premium inventory. It was also active through at least 14 different exchanges and SSPs. Within each SSP, multiple networks were infected. HyphBot was generating up to 1.5 billion requests per day and it generated fake traffic on more than 34,000 different domains, including premium publishers, and more than a million different URLs. Our analysis suggests that infected devices – a network of bots – accessing the Internet from more than half a million IP addresses (mostly from the US) are responsible for this wave of non-human traffic. Most notable, the problem is not in the long tail, it is affecting premium publishers. The details indicate a rather advanced methodology. This meant that detection needed deep analysis and rich data drawn from various sources across our product suite.

As of today, Adform has managed to not only identify HyphBot but to gain access to many of the details on how it was run. We have access to parts of the actual source code, and therefore know how the pages of premium sites were faked and where ads were shown and setup. We have access to the IPs of data centers where those servers are hosted. We also have direct access to advertising tags that are placed directly in the source code. This means, we have information on where exactly the bid requests originate from.

Detailed Analysis

In the following white paper, we'll present the full case in chronological order. First, we'll showcase how we performed our high-level data analysis which identified the patterns that define traffic originating from the botnet. Later, we'll deep dive into specifics on how - through harnessing data - we were able to directly measure and track using advertising impressions deliberately bought and

subsequently delivered via our Ad Server. This enabled very deep analysis which is not possible with pure-play DSPs that do not have access to this level of direct measurement via Ad Serving tags.

Bid Request Patterns

This whole story starts with ads.txt. When observing data that is sold by non-legitimate sellers we follow a simple yet powerful routine. This approach is based on ads.txt data which is publicly available. As such, all impressions we observe that are non-compliant according to ads.txt are analyzed in-depth to answer the following question: Is the ads.txt signal we receive raised due to the reselling of legitimate traffic via unregistered networks? Is the problem due to ads.txt files that are not properly setup on publisher sites? Or, is it non-legitimate traffic that is generated using domain spoofing? If the answer to the last question is yes then we have found what we were looking for.

During the analysis of this case, which we later started calling the HyphBot network, we identified patterns within the full URLs of the suspicious traffic. We noticed that the following two patterns appear relatively frequently:

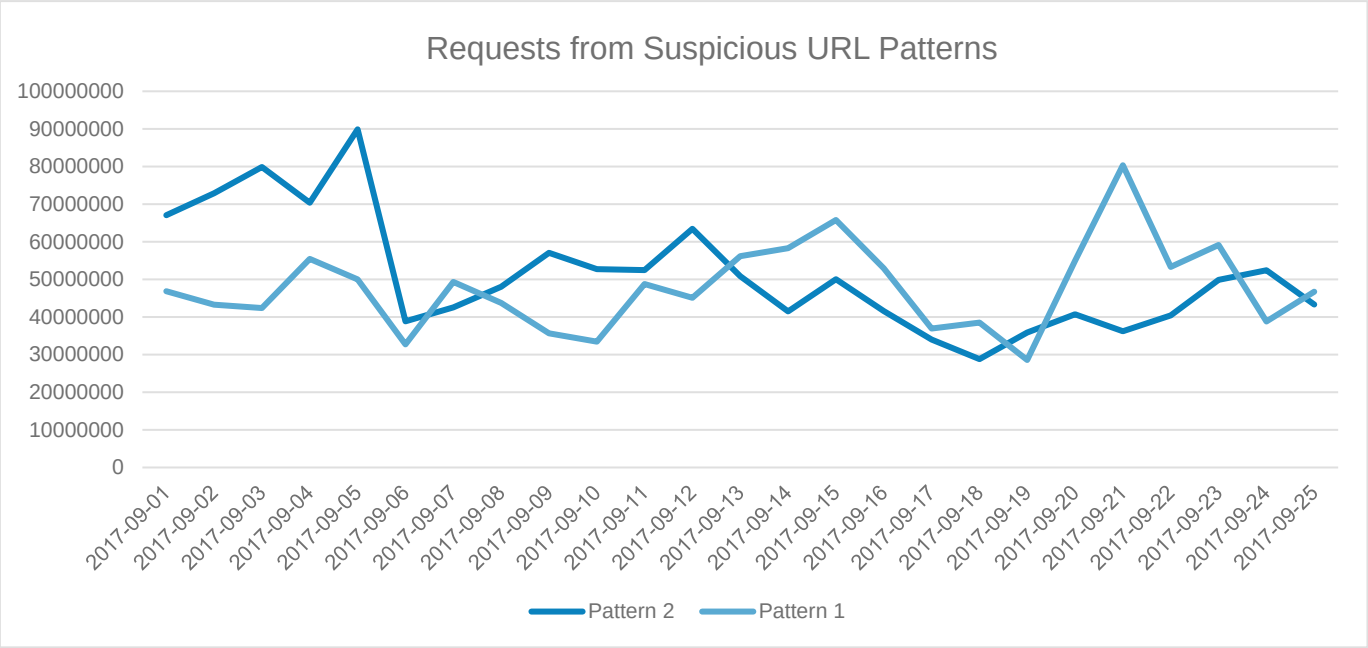
- **Example of Pattern 1**
  - http://economist.com/bacteriostatic_Lema_swanned_nonspirituness/nonteachably_auto-audible/uncanonization/Mendelson
  - http://forbes.com/red-throated_Mid-atlantic/auctorizate_cowperitis_jicara/aschistic_Reduviidae_chromophobia/someonell/patencies_PO_fraudproof/Osterhus_lassiehood_cowthwort_augen-gabbro/periclitation_Porifera
- **Examples of Pattern 2**
  - http://economist.com/z556
  - http://forbes.com/q454

To wrap up, we observed spoofed traffic that is supposedly run on legit domains followed by:

- Either some concatenated, meaningless, but real words as can be seen in the first example.
- Or, a character followed by three numbers as presented in the second.

Initially, we considered the hypothesis that these findings could be the result of a technical issues from an SSP and that this represented some sort of “encryption”. Of course, this was rather unlikely but we needed to make sure that it was not just some error where some variables were incorrectly “injected” into the URL of the bid request. On closer inspection we confirmed that all of these URLs simply did not exist.

At this point it was clear we were on to something. The next step was to establish size and scale. Using data within our data warehouse, we were able to run through our log-level data via regular expression queries and come up with the number of affected bid requests. The results can be seen in the Table below.



Please note, for this scale impact analysis, we only used known and confirmed invalid and fraudulent URL patterns. The botnet also produced legitimate URLs that did not follow the mentioned pattern. As such, the above numbers represent a conservative analysis of the true scale of HyphBot.

As those regular expression patterns can be too broad, we manually checked the most common domains that popped up. We found very few entries in our result set (mostly forum sites) where the URLs which pattern matching identified are valid. For downstream analysis these were excluded. A set of most common URLs with the affected patterns can be found here.

In case you are interested in how these URLs are created, we later found out that they are constructed in the case of Pattern 1 using concatenations of random words that were generated using sortedUnixWords.txt.

Ad Server Data

One of the major benefits of a full stack advertising platform is that we can compare patterns from Bid Request data taken from our DSP with actual delivery data from our Ad Server. As such, we deliberately decided to buy traffic from HyphBot in order to perform deeper analysis by injecting our java scripts via Ad Server tags.

As a result, we extracted full URLs by querying the browser directly via JavaScript. On a DSP you have to trust what SSPs are offering you as part of the bid request. The “real” URL that is measured via JavaScript is way harder to fake. However, HyphBot was built by smart people. Our analysis showed that full URLs tracked via our Ad Server match 99% of the time the URL that was offered in the bid request. This means that the browser actually “believes” that it’s visiting these URLs.

At this point we started testing further hypothesis to get a full understanding of what was happening. This includes, of course, considering the small chance that the reason for this traffic can be attached to the usage of some sort of “anonymization add-on” within certain browsers. But, chances of this are slim, especially considering the scale.

Aggregated Analysis

In the following we present further details that demonstrate that we are in fact not dealing with human traffic but, are step-by-step seeing more clearly that this is actually a major fraud network.

User-Agent and Operating System

Requests with suspicious URL patterns were coming from Chrome for Desktop on Windows 7 or 8 in almost all cases. For later analysis we opt to focus on desktop only. The reason for that is that Mobile traffic had slightly higher rates of false positive URLs. Meaning, in still a low fraction, but in more cases, the URLs actually existed.

Please, see the results of the analysis regarding the most commonly used user-agents below:

- Mozilla/5.0 (Windows NT 6.2; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.90 Safari/537.36
- Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.90 Safari/537.36
- Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.90 Safari/537.36
- Mozilla/5.0 (Windows NT 6.2; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.78 Safari/537.36
- Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.72 Safari/537.36
- Mozilla/5.0 (Windows NT 6.3; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/60.0.3112.78 Safari/537.36

Based on this data, derived from DSP bid requests, we were able to do further analysis using data from our Ad Server. We introduced custom logging via scripts delivered into those suspicious impressions – bought for explorative purposes. Browsers that are visiting the spoofed sites supported all of the features and function calls that a real Chrome browser would. This strongly indicates that user-agents are not spoofed and it is a real Chrome browser.

IPs / Countries

The following analysis is done on IP addresses. Please note, sometimes SSPs send us anonymized IPs (randomized last octet). Our analysis is of course run on a non-randomized subset. Below, you can see the top user countries (out of a total of 122 where the traffic was origination from):

Country Name	Distinct IPs	Distinct Subnets	Requests
United States	590,382	277,335	1,787,729,200
United Kingdom	4,611	4,334	6,830,100
Canada	1,469	1,407	2,432,800
Netherlands	89	65	140,300

If we also include anonymized IPs one additional country appears:

Country Name	Distinct Subnets	Requests
India	4,002	1,018,100

In order to confirm that the computers are actually run and owned by “real people”, we checked registration data from a very large random sample of those IPs. The results show that virtually all IPs are residential IPs from wide range of Internet Service Providers.

Video / Non-Video

Of course, the type of media that is being offered and served is of major interest as this clearly has an impact on the potential fraud damage. The results speak for themselves, we are talking about predominantly video inventory.

Is Video?	Requests
true	1,578,765,000
false	231,622,400

Methbot Comparison

After seeing summary statistics our first idea was that it looks very similar to Methbot. In particular:

- Primarily US.
- Primarily video.
- Primarily desktop.
- Non-existent URLs, but spoofing not easily detectable by Ad Servers.
- Large scale - already our initial aggregate analysis indicated a size similar to that of Methbot. Later in the white paper we will explain our methodology and that we are seeing a volume of fraudulent impressions between 400 million to 1.5 billion requests per day. This is in comparison to Methbot's 300 million per day.

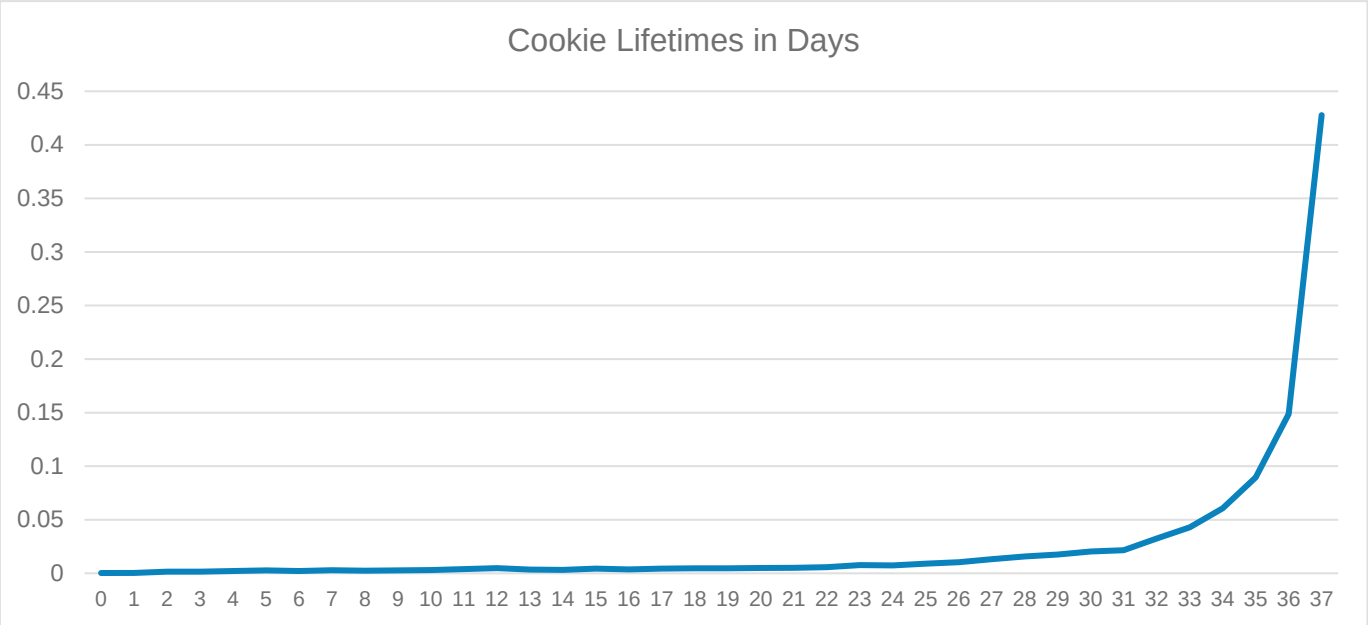
However, there were also important differences:

- Methbot was running on a custom browser, not Chrome.
- Methbot had a very limited set of IP ranges (number of IPs were high, but they could be stacked into 161 IP ranges), while in this case IPs are much more scattered and do not fall into ranges.
- IPs in our case are generating 30 times more traffic than just the patterns (including direct campaign transactions) so either the scale of this scheme is 30 times larger than initial analysis suggest or it’s running on infected machines and some part of IP traffic is valid human traffic.

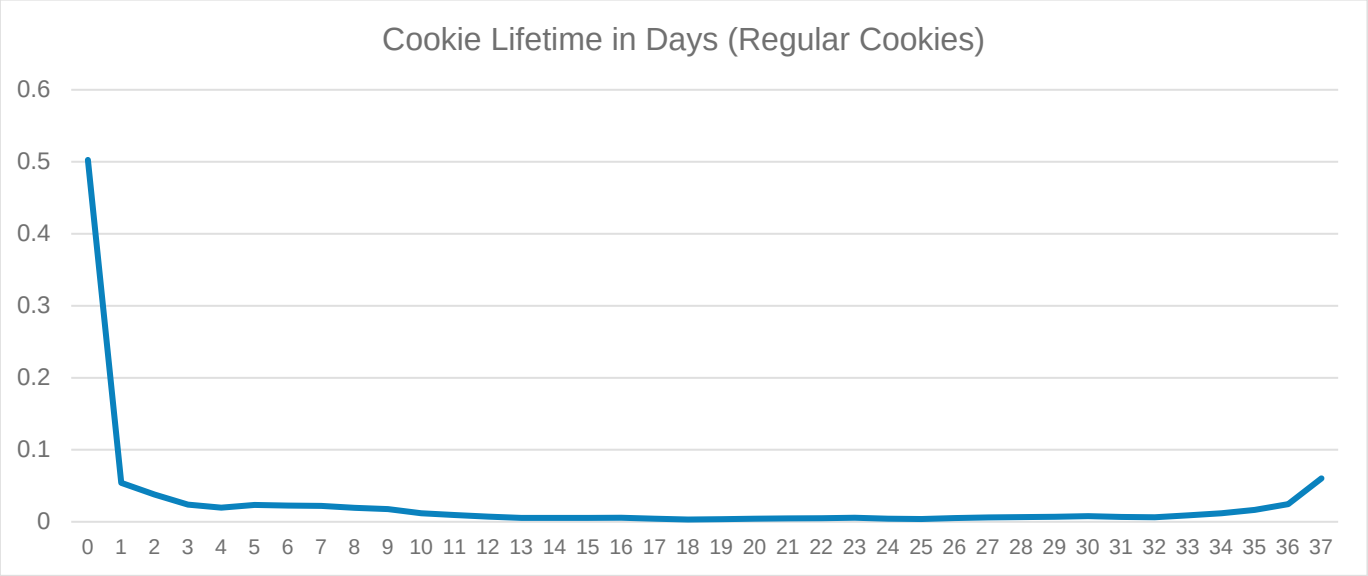
Because of this we believe it to be infected machines instead of data centers with forged registration data.

Cookie-Level Analysis

In cases like this, cookies are a better indicator of suspicious behavior than IPs. Therefore, we analyzed cookies that visited a URL matching our pattern and found the following. The selected cookies visited almost only sites that we identified to be spoofed and to be part of this fraud scheme (95% of all visited URLs). It is also interesting that almost all of those cookies have very high lifetimes. For details refer to the graph which is calculated using 38 days of data:



In comparison, for random cookies which we picked for comparison, the curve looks like this:



In a real-world set of cookies, there are many one-day-only cookies. Reasons for that are numerous, e.g. if you open an incognito tab your cookie changes every time or we see similar behaviors when you use Safari in many scenarios. Also, the daily activity rate of suspicious to random cookies varies significantly.

Our assumption here: In case of infected machines, a separate Chrome browser is launched which has its own very stable cookie that is active only when the user is active. Worth noting, HyphBot is spoofing 34,000 distinct domains that are being rotated. So, these cookies collect very rich browsing histories.

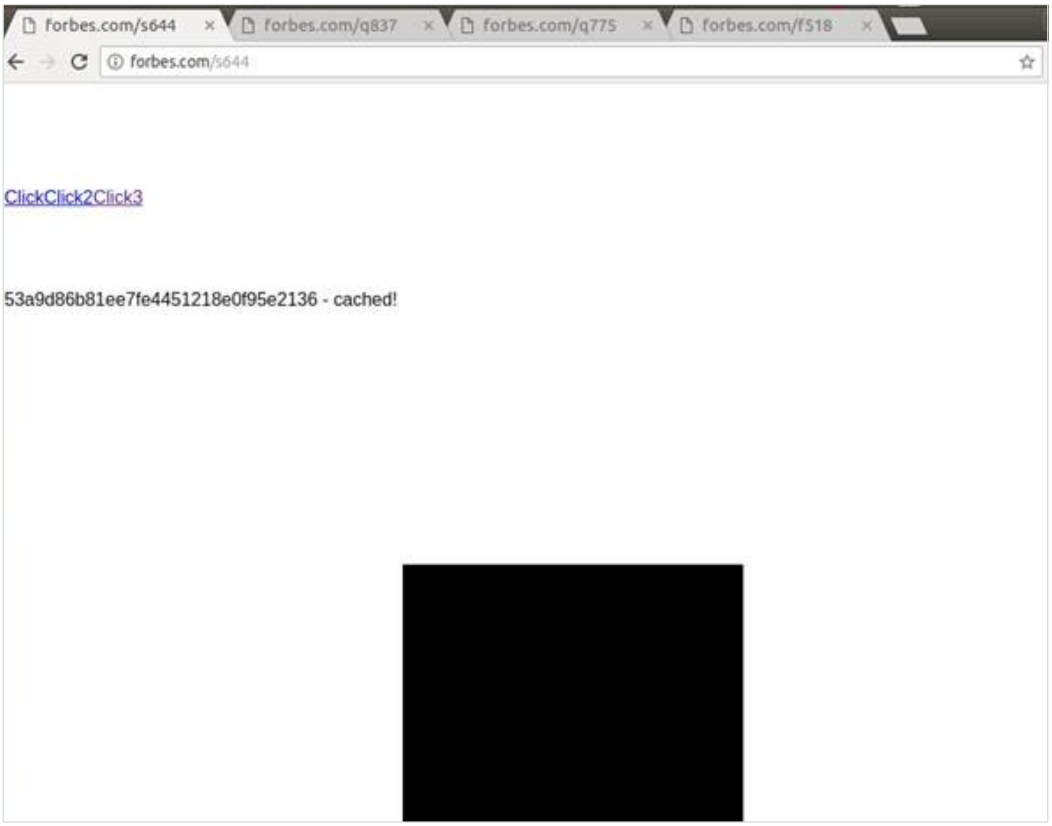
Additional Logging Through Ad Server

In order to derive even more proof, we logged additional information by deliberately buying traffic from suspicious sources using our DSP and then injecting the analytics code via our Ad Server. Again, the results were very telling. We identified at least four distinct and very clear signals that allow us to separate HyphBot traffic from real-people traffic. The resulting signals are so strong, that we can use them as a fingerprint to classify fraudulent impressions that originate from HyphBot.

This shows the strength of Adform's ability to tap into bid stream patterns (DSP) as well as to analyze in-depth using our own measurement and tracking modules (Ad Server). The ability to peak into multiple steps of the delivery chain give us a rather unique opportunity to identify fraud. During this analysis, we built and identified these fraud fingerprints by injecting the necessary scripts not only into the suspicious traffic but into all our traffic. This had – of course – no operational impact on our clients but enabled us to really see what differentiates HyphBot traffic from real traffic and, as a result, improve our fraud filters.

IPs Where Sites are Hosted

Through our custom ad-server logging we learned that all infected browsers call one of seven data centers in Germany. We simulated their behavior by changing hosts on our browsers for these domains to resolve to the data center IPs and we were able to load following page:



As a result, our browser now "thinks" that it is visiting Forbes.com, as you can see in the address bar. However, the site contains just the video player to show ads and a couple of links. After clicking on those links you get redirected to another fake Forbes site with a different URL.

Full Set of Domains

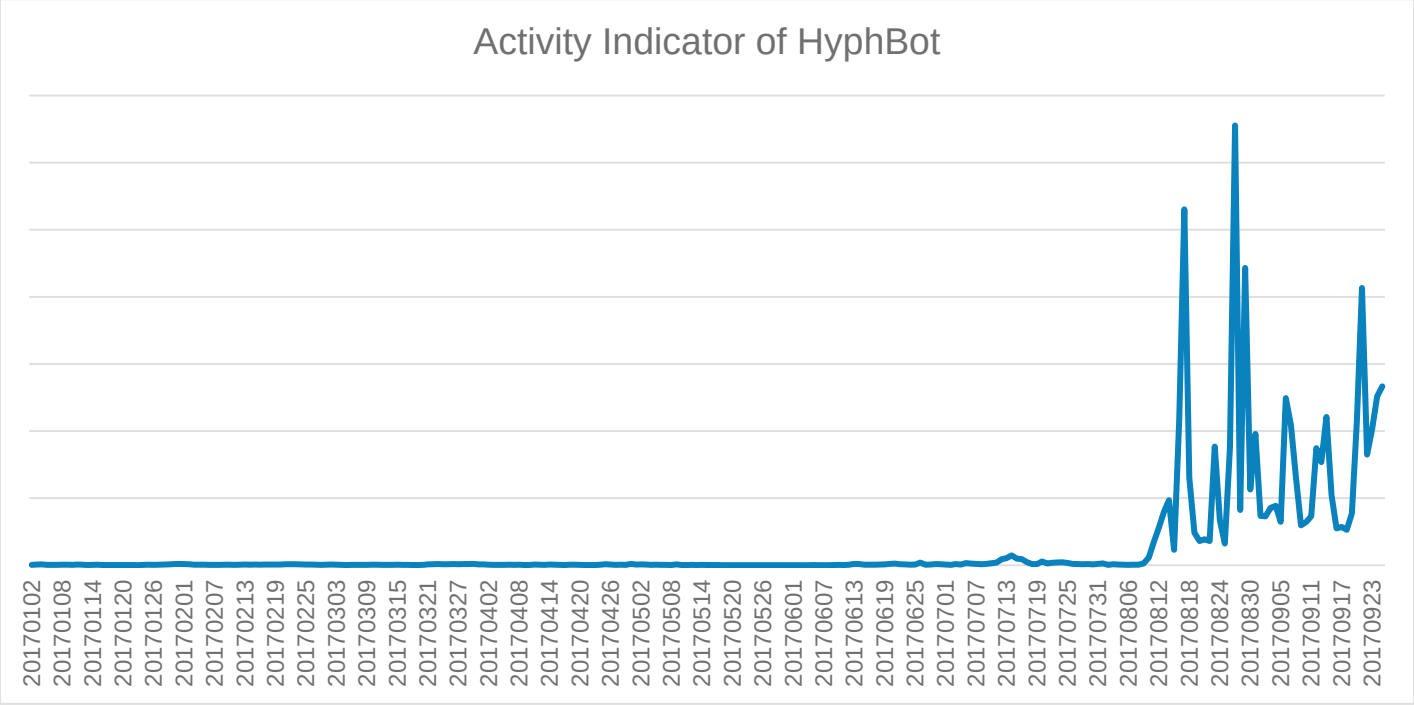
Then, we simulated the same logic for how infected cookies are redirected between the spoofed sites. As a as result, we extracted a set of 34,000 domains that are being hosted on these machines. The full set can be found here.

When Did it Start?

We have found forum posts where site owners complain about strange calls by ad / site-categorization robots to non-existent URLs on their sites that date back to November, 2016. However, they mention that it has become particularly bad since August.

Due to our data governance and retention strategies, we store full URLs for a limited period of time. Hence, we cannot tell precisely when patterns started to appear. However, we have aggregate info by domain, which is stored for very long periods.

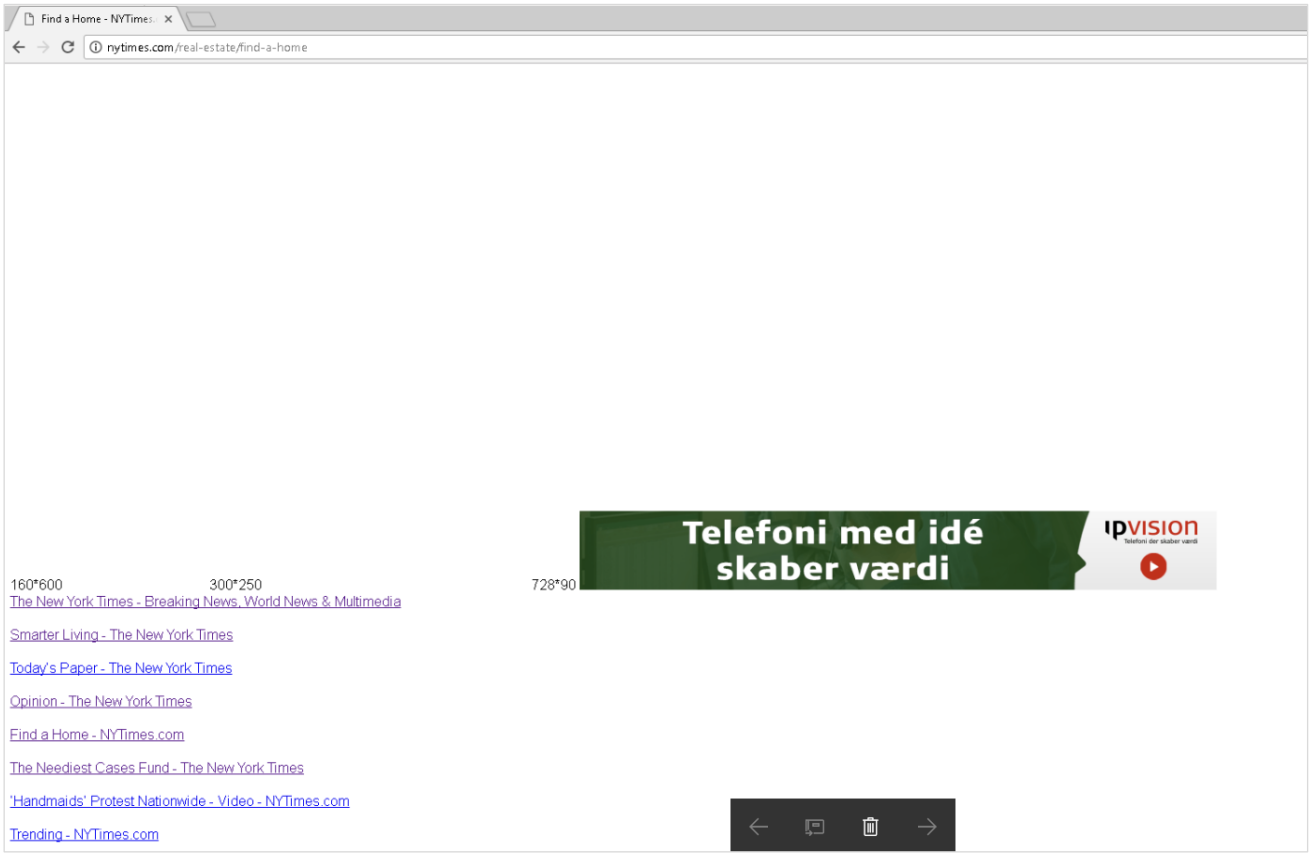
In the case of HyphBot, legitimate publisher domains together with very obscure sites were spoofed. We hand-picked domain examples that we identified to be specific to HyphBot and checked their activity over time using aggregate information from our data warehouse:



It seems clear from this analysis that HyphBot was in a testing phase in July 2017 and became active from mid-August on.

What's the True Scale

As already mentioned, the "Patterns" of the bid requests (as depicted by the first graph in this white paper) represents only a lower bound for the overall scale. We found many examples where other, also legitimate, URLs are formed on the host machines, not only URLs with the suspicious and very structured pattern. For example:



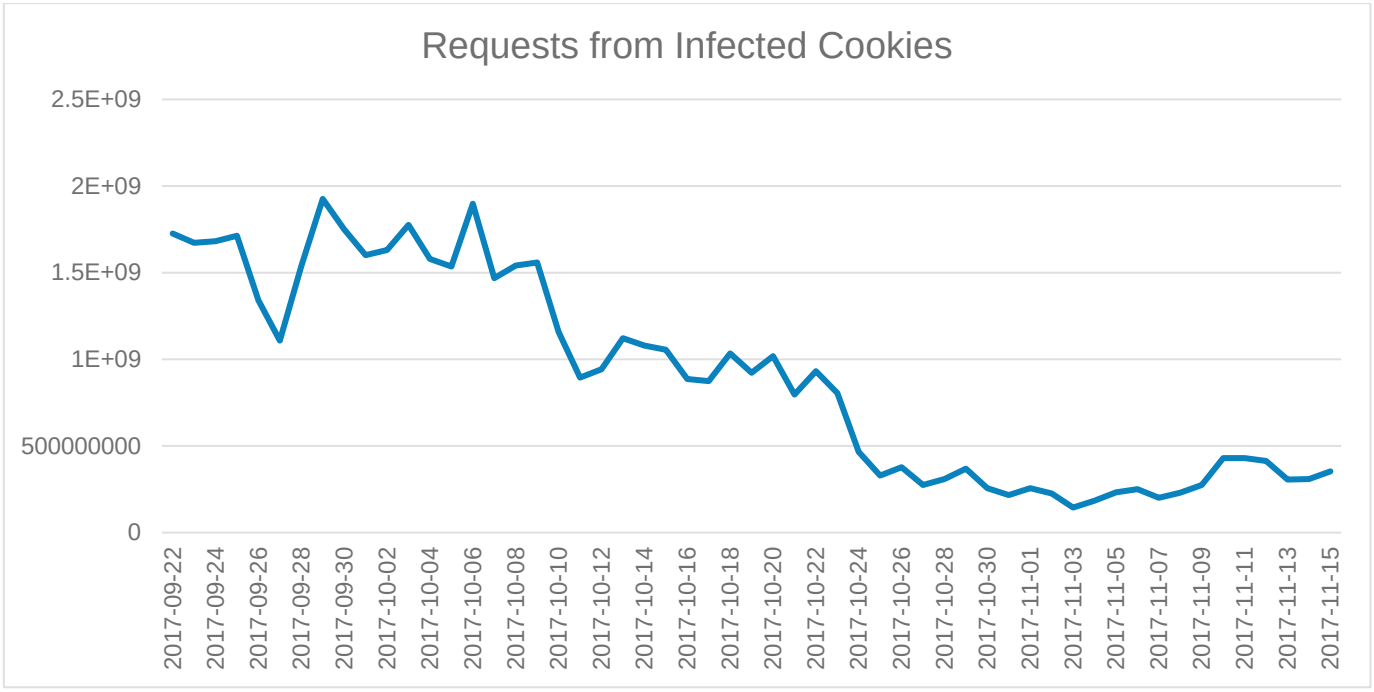
We tried to estimate the full scale of HyphBot using the following methods:

- **Method 1**
  We hand-pick sites from the spoofed domain list that are HTTPS-everywhere. We then select the ratio of HTTP requests, which we believe are invalid and part of same spoofing scheme, to the ratio of requests that had the original suspicious pattern in them. The result: There were around four times more invalid requests than the original pattern analysis suggested.
- **Method 2**
  We check the ratio of all requests from infected cookies to cookies that exhibit the “infected pattern”. The result: There were around 15 times more requests from infected cookies.
- **Method 3**
  We checked the ratio of impressions – bought purely for research purposes – which we discovered were fraudulent based on the fingerprint signals we derived via Ad Server scripts. The ratio was also around 15. Please note, that these were of course bought on a strongly biased sample.

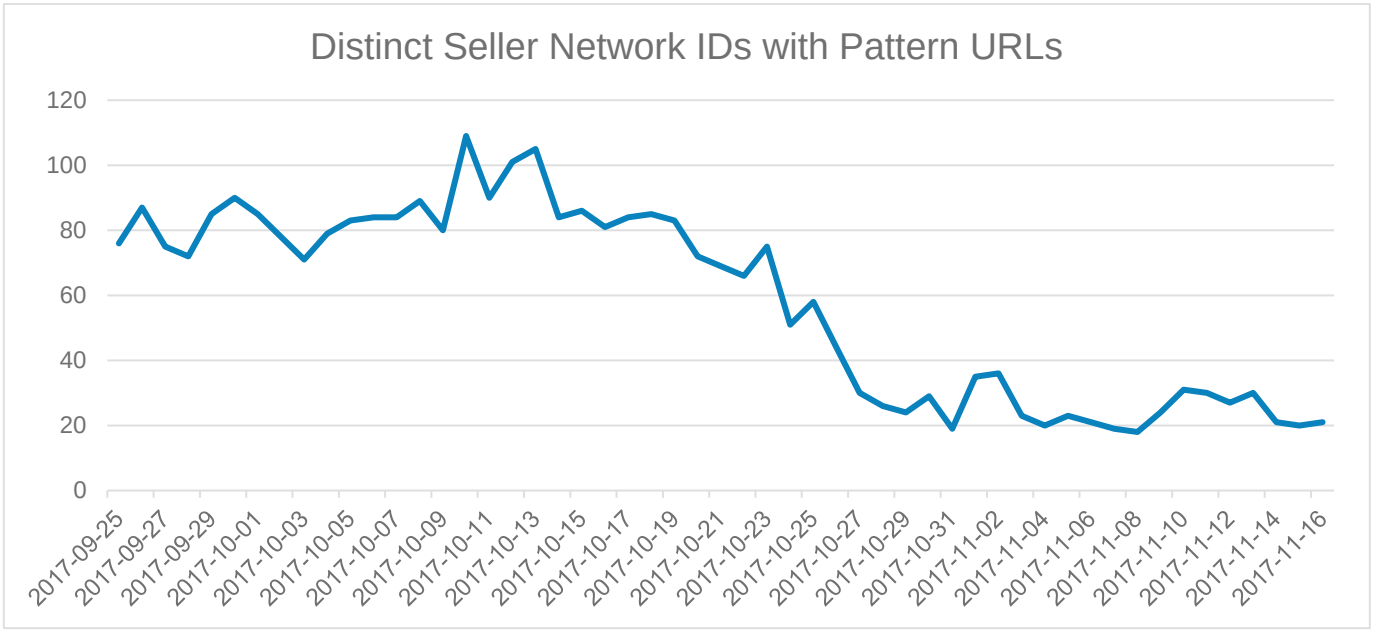
We believe Method 2 and Method 3 to be most robust. As both produce a similar factor of 15, we therefore can derive that the 100 million requests per day with the identified clearly suspicious URL pattern translated into 1,500 million requests per day as we can expect at this multiple of 15. The most (and very likely) conservative estimation is based on method one and yields only 400 million requests per day based on the multiple of 4 derived via Method 1.

Dynamics Over Time

We contacted the majority of exchanges 2 days after the analysis started when we had hard proof that the traffic was fraudulent. However, it took more than a week until we started to see a significant reduction in fraudulent traffic being sent.



Number of networks that send this traffic (same network can be counted multiple times if it is visible through multiple exchanges):



What We Do Not Know

Unfortunately, we still do not know how people and their respective computers and browsers are infected.

Financial Impact

Estimating the impact of HyphBot on financials is a tricky effort. Only a few players in the industry have the actual numbers which serve as hard facts and can be used for a strong parametric estimation. To estimate how bid requests translate into real Dollars, we need to take into account several parameters:

- **Header Bidding Factor**

  How many requests are generated for one impression? The strong move towards header biding potentially can inintiate multiple auctions for the same impressions via different supply partners.
- **Fill Rate**

  How often are actual ads / videos being served? Just because there is an opportunity to buy an impression does not mean that each impression is actually “filled” as some will at the end be remnant and not sold.
- **Avarage CPM**

  The average CPM price that is being paid for the fraudulent inventory.

Unfortunatelly, at Adform we cannot know the first two parameters. As these can vary greatly, we try to use estimations based on best practices. Regarding the last parameter, the average CPM, we estimate it to be between $7 and $12 CPM. This is based on parametric estimation using analogue values of similar requests we do see on our platform.

In conclusion, the total financial impact, or, in other words, the money spend on these requests is calculated as:

$$Spend = (CPM * Requests * Fill\ rate)/(Header\ bidding\ factor * 1000)$$

At its peak, the botnet generates 1,500 million requests per day. The table below provides a range of estimates on how this impression volume can be translated into actual Dollars spent.

	Header Bidding Factor	Fill Rate	Average CPM	Spend per Day
Low	12	0.3	$7	$262,500
High	7	0.5	$12	$1,285,714

What Industry Players Can Do

Impactful Solution

DSPs and SSPs should check their data warehouses for the patterns highlighted above and they should also investigate requests from cookies that visited those patterns. In addition to this, they should contact all networks that are sending traffic from these patterns.

If networks are not transparent about their source, we suggest you shut them down. There needs to be a real cost of doing business with fraudulent players.

Partial Solution

Short-term, the following measures are potentially easier for affected players to act upon:

- Block infected IPs. The downside is you would also filter a large fraction of legitimate traffic as well and will not be able to filter everything as new IPs will continue to appear.
- Block infected cookies. This will only work in the short term as cookies are easy to change.
- Some of the domains in the spoofed domain list generate a lot of transactions, but don't contain ads – you can block traffic from these domains from all sellers. However, manual work is necessary to hand-pick the domains.

Enforce and Use Ads.txt

A lot of this traffic could have been avoided if most industry players were using Ads.txt (for analysis or as filters). However, the key points are:

- Ads.txt does not guarantee "fraud-free". You need to have faith that the publisher or authorized seller / reseller does not inflate the traffic by mixing legitimate traffic with traffic from bot networks. We already have examples where premium domains have now added networks that were sending traffic from this botnet to their ads.txt files.
- It's relatively easy for botnets like HyphBot to change their spoofing scheme to generate traffic from domains that don't have Ads.txt so it's also good to use domain whitelists in general – at least for Branding campaigns.