AKAMAI WHITE PAPER

Scrapers and Bot Series
Managing Professional Bots

Larry Cashdollar, Akamai SIRT

Akamai
FASTER FORWARD

TABLE OF CONTENTS

Editor's Note: In this installment of the Akamai SIRT white paper series, we continue our examination of scrapers and bots. We focus on aggressive bot behavior, methods they use, and how to mitigate them. This paper focuses on those "bad" bots and the unwanted, inconsiderate, and malicious traffic they create.

OVERVIEW

In our previous white paper, "When Good Bots Go Bad", we discussed various bot types, motives, and behavior. In this issue, we highlight aggressive bots that ruthlessly scrape site content, create DDoS-like conditions with hundreds of requests per second, spoof legitimate good bot user-agent strings to bypass WAFs, and in extreme cases, attempt to compromise the host in order to infect its victim with malware.

These aggressive bots can make so many requests that they put unwanted stress on the origin server and, if a customer's rate controls aren't configured correctly, can cause a Denial-of-Service condition as a side effect. We say side effect because both the site owner and the Bot operator lose if the site is taken down; the scraper isn't getting the information it wants and the site isn't available to end users.

In the next section, we examine the worst of these bots and ways to mitigate their unwanted traffic, starting with product scrapers.

METHODOLOGY

Scrapers

Scrapers typically mine a site in order to steal product data, harvest email, or duplicate a site in order to stand up a fraudulent phishing site. Bot traffic might last anywhere from a few minutes to a couple of days. However, some scraper bots are relentless — pulling data from a site for weeks or months. The hardest-hit sites? Usually travel and hospitality industry websites, since bots will be looking for cheap rates.

The graph below shows an example of a scraper bot targeting a retail customer by systematically indexing all of the site's product pages. The connections attempt to imitate what a typical Chrome browser looks like in order to escape detection. The bot maintainer has falsified the user-agent string on the botnet; however, the request headers don't match and some are missing for a normal Chrome browser session. This is a common feature of many bots, where the attempt to create a false user-agent string is close to, but not quite the same as, the real thing.

The bot is geographically distributed over 250 networks with just over 1,500 unique IP addresses. This can happen if the botnet is sharing the workload among nodes or if it's a single system utilizing round-robin connections over many proxies and VPNs. This Bot scrapes a product list off the main page then scrapes the detailed information of each product from their individual pages.

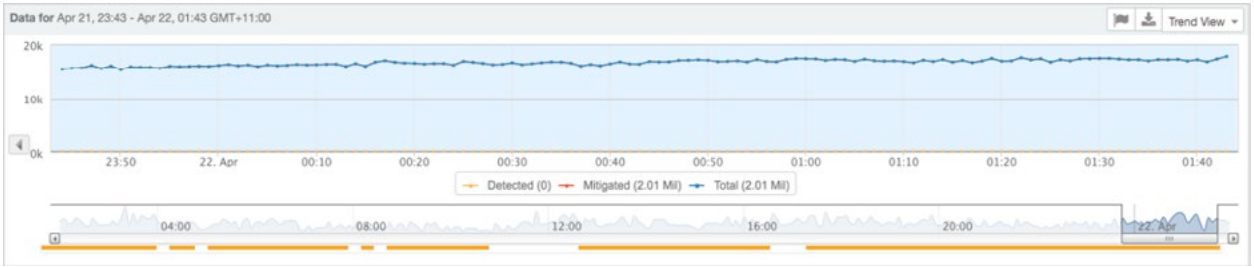
PATH	QUERY	STATUS	USER AGENT	SOURCE IP	COUNTRY / CITY
/us/39511361ST/item		200	Mozilla/5.0 (Windows NT 6.2; WOW64) Appl...	5.178.87.247	RU / SAINTPETERSBU..
/us/39579500AP/item		200	Mozilla/5.0 (Windows NT 6.2; WOW64) Appl...	5.178.87.247	RU / SAINTPETERSBU..
/us/37715466ST/item		200	Mozilla/5.0 (Windows NT 6.2; WOW64) Appl...	5.178.87.247	RU / SAINTPETERSBU..
/US/34529565/item		200	Mozilla/5.0 (Windows NT 6.1; WOW64) Appl...	23.247.82.94	JP / TOKYO
/us/37712724KE/item		200	Mozilla/5.0 (Windows NT 6.2; WOW64) Appl...	5.178.87.247	RU / SAINTPETERSBU..
/us/37703201AN/item		200	Mozilla/5.0 (Windows NT 6.2; WOW64) Appl...	5.178.87.247	RU / SAINTPETERSBU..
/US/36558961/item		200	Mozilla/5.0 (Windows NT 6.1; WOW64) Appl...	107.179.86.119	US / LOSANGELES
/US/44871878/item		200	Mozilla/5.0 (Windows NT 6.1; WOW64) Appl...	192.200.218.83	US / WALNUT
/us/39572570TQ/item		200	Mozilla/5.0 (Windows NT 6.2; WOW64) Appl...	5.178.87.247	RU / SAINTPETERSBU..
/US/34489406/item		200	Mozilla/5.0 (Windows NT 6.1; WOW64) Appl...	23.247.63.148	US / LOSANGELES

Request Headers

Client IP	52.8.102.235
Source Port	55114
User-Agent	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.3 (KHTML, like Gecko) Chrome/19.0.1061.1 Safari/536.3
Method	GET
Scheme	http
Host	<redacted>
Port	80
Path	/us/44978316IA/item
Accept-Language	en
Accept-Encoding	gzip,deflate
Accept	text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
User-Agent	Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/536.3 (KHTML, like Gecko) Chrome/19.0.1061.1 Safari/536.3
Host	<redacted>
Referer	<redacted>
HTTP Version	1.1

DDoS

Some bot networks aren't built to steal information. Some are created to be used as a digital weapon. The bot in the example below is being used to DDoS attack a large gaming and entertainment company website; it uses close to 4,000 unique IP addresses:



The user-agent string and request headers are unique to this botnet, allowing us to block it not only using rate controls, but its distinct request header signature as well. The bot is targeting the main page of the website — the root directory, with no file path or queries — with a simple GET request. This attack generated traffic rates of 16,000 requests per minute. Unlike the previous example, this bot didn't make any effort to hide its nature. It's a blunt trauma weapon and the owner didn't care if the target knew it was there.

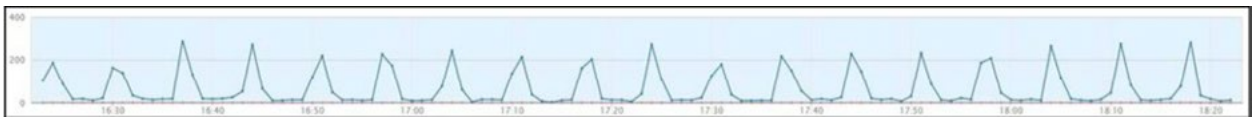
Request Headers

HTTP Request:

```
GET / HTTP/1.1
Request Headers
User-Agent: XXXX
Host: <redacted>
Cache-Control: no-cache
```

Account Checkers

Recently, one of Akamai's large Media & Entertainment customers experienced an account checker attack against the main login page of its site. The bot consisted of over 1,200 unique IP addresses distributed geographically throughout multiple countries, with concentrations in Italy, Spain, Finland, and Germany.



The IP addresses consisted mostly of Virtual Private Server (VPS) hosting sites. The bot utilized more than 90 different user-agent strings in order to evade WAF rules. Account checkers take great care to hide their true nature, even more than scrapers. They are used to checklists of usernames and passwords harvested from other sites and lose all value if their activities can be detected and blocked.

Top user-agent headers

```
Mozilla/5.0 (Windows NT 6.3; WOW64; rv:42.0) Gecko/20100101 Firefox/42.0 Mozilla/5.0
(Windows NT 6.1; rv:38.0) Gecko/20100101 Firefox/38.0

Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_1) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/46.0.2490.86 Safari/537.36

Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/46.0.2490.86 Safari/537.36

Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/42.0.2311.135 Safari/537.36 Edge/12.10240

Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_0) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/45.0.2454.101 Safari/537.36 Mozilla/5.0(compatible;MSIE9.0;WindowsNT6.0;
Trident/5.0; Trident/5.0)
```

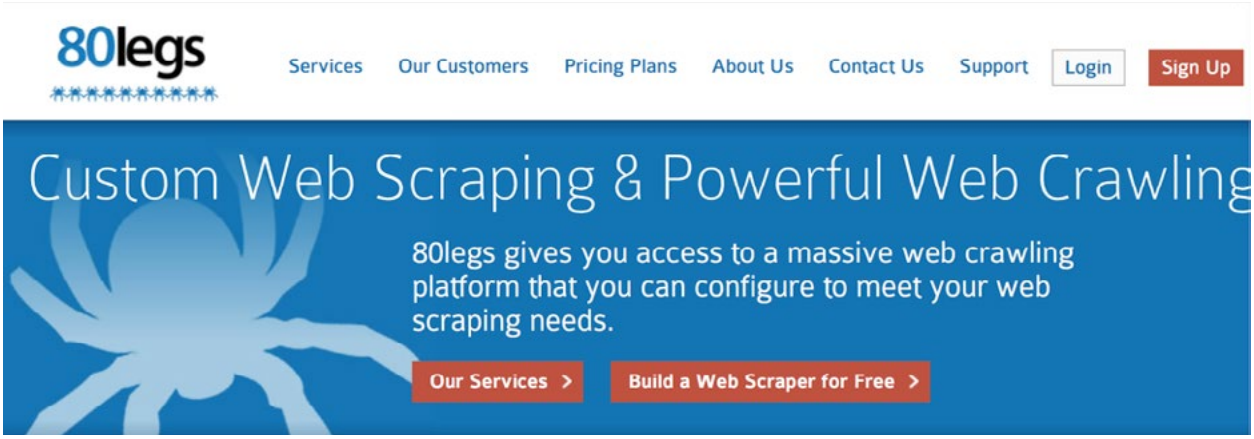
Request Header

```
POST             /index.php?id=5791 HTTP/1.1
Content-Type:   application/json; charset=UTF¬8
Accept:          application/json, text/javascript, */*; q=0.01
Content-Encoding: identity
Connection:      keepalive
User-Agent:      Opera/9.80 (Macintosh; Intel Mac OS X 10.6.8; U; fr)
                 Presto/2.9.168 Version/11.52
Accept-Language: enUS,en;q=0.5
Referer:         <redacted>
X-Requested-With: XMLHttpRequest
Accept-Encoding: gzip, deflate
Cookie:          <redacted>
Host:            <redacted>
Content-Length: 58
```

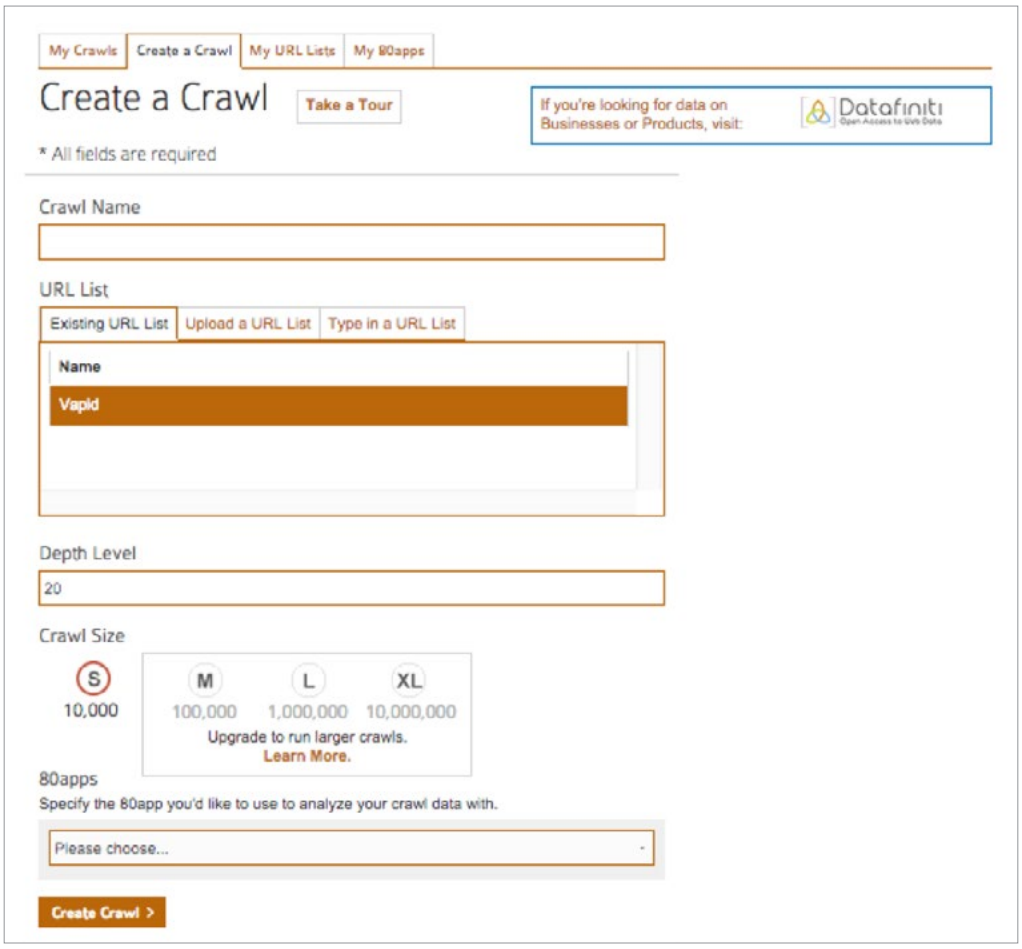
With the above two volumetric Bots DDoS and account checkers, mitigation will rely on controlling the requests per second to a site from a specific IP address. The more you slow them down, the less efficient they become.

Professional Scrapers (Bots for Hire)

Professional web scraping has become a big business because of the value created by websites across the Internet today. Legitimate websites create value in the form of content, which can be spoofed on fake sites or used by competitors in competitive pricing. The website http://www.80legs.com will index and scrape websites of the attacker's choosing, for a fee.



After logging in, we can create a custom job with the option to upgrade the 'Crawl Size'.



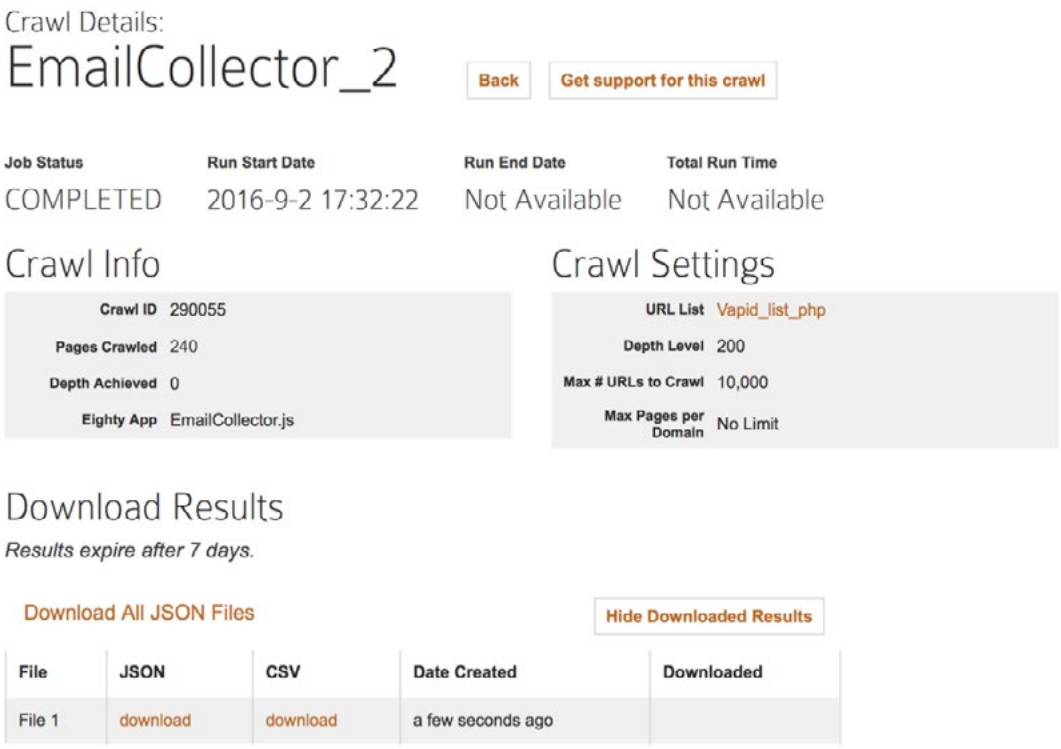
Depending on the type of content you're after, you can specify 'Crawl Methods' to target specific data types like images, links, emails, and files.

Crawl Methods

```
CheckStatusOnExternalDomain.js
CrawlImages.js
CrawlInternalLinks.js
DocumentData.js
DomainCollector.js
EmailCollector.js
EmailsAndPageContent.js
ExternalLinkCollector.js
FileFinder.js
FullPageContent.js
HeaderData.js
KeywordCount.js
LinkCollector.js
LinksAndKeywords.js
LinkTracer.js
LossyDocumentData.js
LossyPageContent.js
LossyPageContentInternalLinks.js
TextFromURLListOnly.js
```

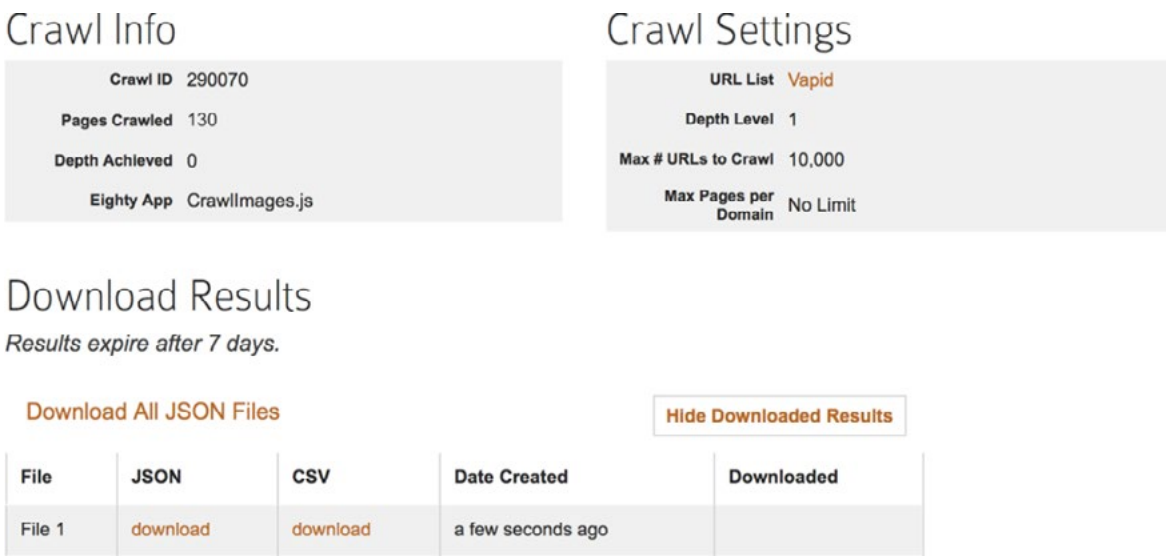
The 80legs scraper bot contains the user-agent "voltron". This should make it easy to block based on this unique string alone; however, with this type of professional bot, the site has instructions on limiting what directories and files are crawled, therefore limiting the bot's access to the target site.

The 80legs platform has the capability to create two crawler types, both of which were pointed at a sacrificial domain for testing. The first crawler uses the `EmailCollector.js script`. This script collects email from the site by scraping pages and looking for text denoting a standard email address. This can be used to gather a list of email addresses for use in creating spam or a phishing campaign against a target company.



The script creates a nice .csv file, populated with email addresses pulled from the target website and the corresponding webpage they are found on. The 80legs crawler connections all had the string "voltron" as the user agent. A common tactic used by adversaries is to use a script like this to discover emails for a specific company or domain.

The second script used is `CrawlImages.js`, invoked to extract image files from a website. A product counterfeiting outfit might use this to get a look at a target's product offerings or attempt to find pictures of a soon-to-be-released product. These images can also be used to create a fake version of the target site for fraud or phishing attempts.



The web interface displays results similar to the above crawl. This time the downloaded .json file contains a list of images, base64 encoded images, and the web page urls they were collected from. Due to the nature of the data, a request to download in .csv format returns an error, as 80legs can't format the processed data into a .csv file.

The .json file could easily be processed and the image files decoded and stored for whatever purpose the user has intended for them.

BOT TYPES

Auto Add-to-Cart bots

Auto Add-to-Cart (ATC) bots are bot networks built specifically for making purchases on limited high-demand items — ranging from designer clothing lines to airfare costs during peak travel season. There are several categories of ATC bots, almost all of which target certain high-demand items. These range from specific clothing lines to minimizing cost on airfare prices during high travel seasons. The bot allows the purchaser to set up an account that will seek out a certain item during its release, or when there is a price drop, and then add it automatically to a shopping cart. Many of these bots have the ability to check social media for the announcement of a new item or the release of a highly anticipated item, giving the user the luxury of not having to wait around for an item to finally be available. These bots also allow multiple accounts, in some cases up to 100, to circumvent item limits per purchaser. This is handy for resellers looking to cash in on popular items. Auto Checkout bots do what the ATC bot does, but with the additional final step of completing the purchase. The user sets up name, address, credit card account information, and contact information, allowing the bot to fully order the desired item.

Some of these bots promise to:

- Add the item the user wants to the cart
- Keep trying to add items to the cart, even if the website crashes
- Keep trying to check out until it succeeds, even if the website crashes
- Notify the user by email or SMS that the purchase has been made

Bots aren't only built to automate retail purchases, as we will see later in the paper. We'll test AIO and the Gold Phantom bots to highlight these capabilities, then discuss bots that can be used to sway voting process outcome.

Product Specific Bots - Sneaker Bots

All in One Bot (AIO)

This bot supports automated sneaker purchases from over 25 retailers. The bot allows users to retry purchasing a high-fashion sneaker over and over again using multiple logins and a multi-threaded software.

There are other product-specific bots, like Nike bot and Adidas bot, tailored to monitor social media for sales and product availability. These bots can begin their purchasing cycle after being triggered by an alert on a sale or product availability from social media. Instead of the user constantly monitoring Twitter feeds for the opportunity to purchase a highly sought-after sneaker, these bots can begin the user's purchasing cycle after being triggered by an alert on a sale or product availability from social media.

General Retail Bots

Gold Phantom

The Gold Phantom bot website describes itself as, "..the best programs available to help you obtain the latest product releases and more. Our Add to Cart bots have been and will continue to dominate the market, since they are regularly tested and updated. We sell our extremely successful online products at the cheapest prices and back it with our high success rates over all other sites." The site offers multiple bots that are custom engineered to target specific retailers:

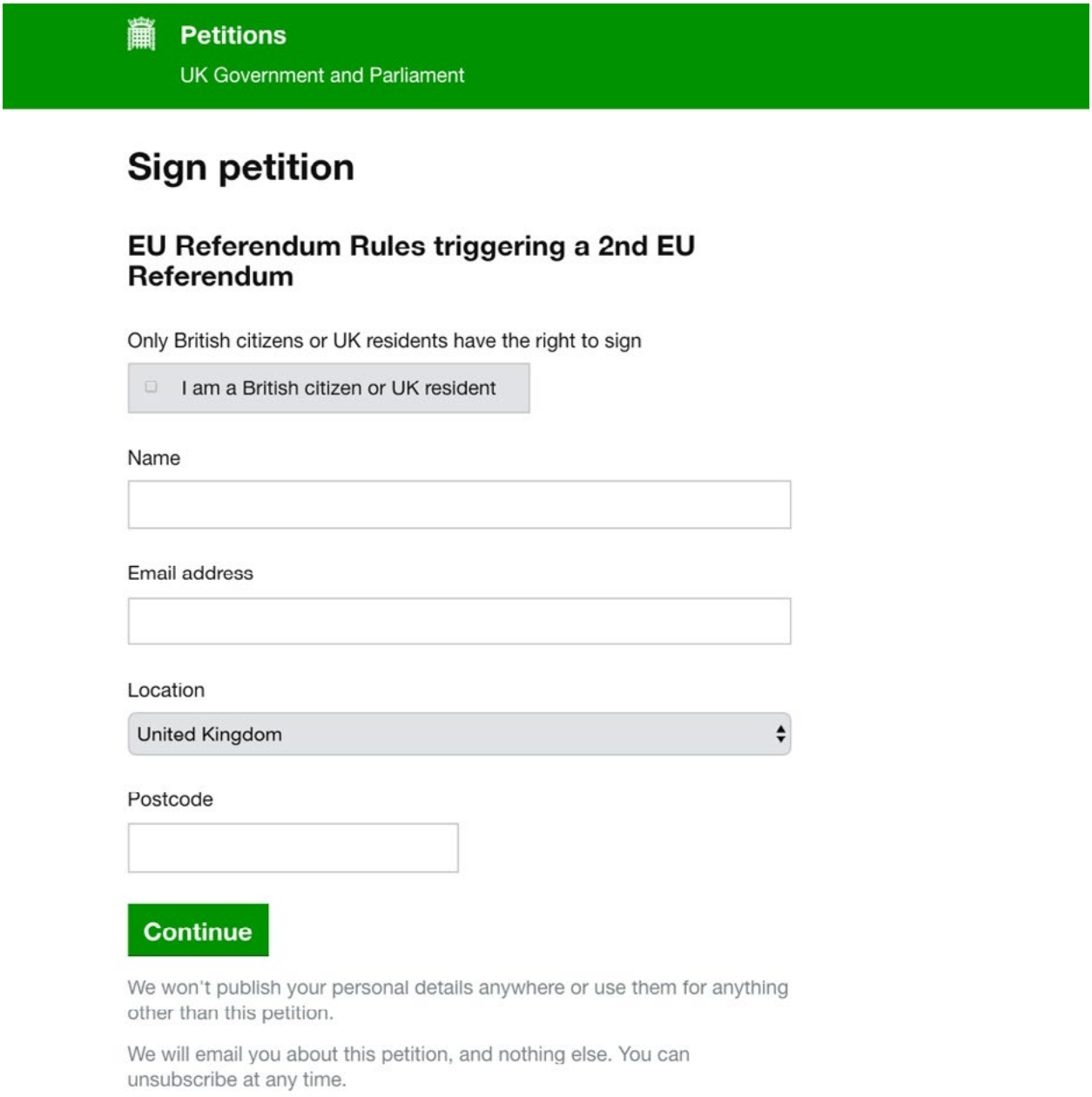
- SweetHoney bot
- Tula bot
- Sassy Chic Boutique bot
- Well Dressed Wolf bot
- Sibling Rivalry Designs bot
- Bourgeois Bebe bot
- Petite Bourgeois bot
- Rosa Flamenco bot

Each bot is matched to that specific clothing line, giving a user the ability to target a specific item for purchase from high-end boutique retailers. The items that this bot targets are in high demand, but not necessarily pricey, which makes them hard to acquire. For $19.95 a month, this bot will help you automate those high-demand, low-cost purchases. Each bot has a demonstration video linked on its specification page.

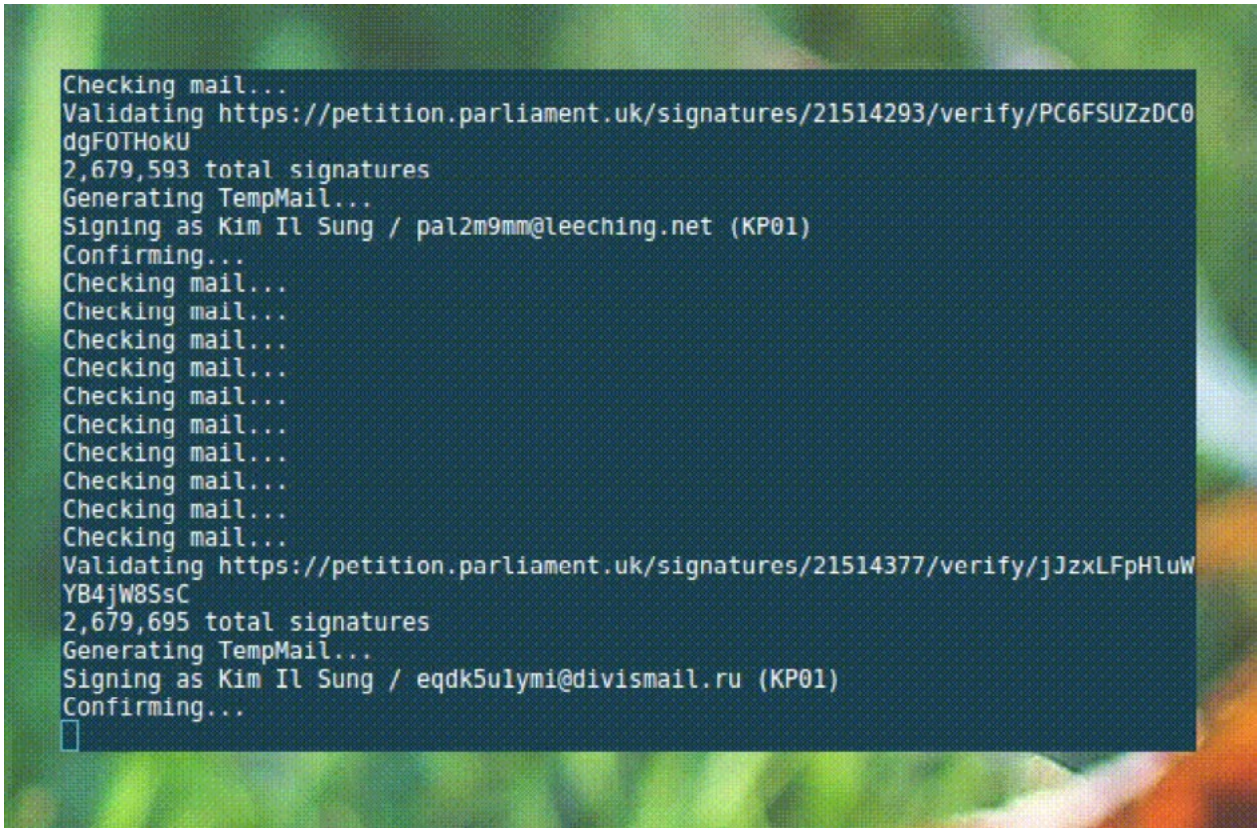
Voting Bots

Vote bots do exactly as the name indicates — they automate the process of submitting an online vote. In some benign cases, the vote may be a cutest baby contest, or a vote for the best gaming website; in contrast, an EU referendum voting website was recently hit with a deluge of automated votes adding fake signatures as a prank from the users of 4chan. The users created scripts and used botnets to automate vote submissions to the website, with IP addresses originating from places like North Korea, Antarctica, and Vatican City.

The EU voting website's submission form page doesn't appear to do any basic user verification or validation, such as a "captcha" or rate controls, which would have made voter fraud much more difficult. The lack of basic controls to detect if a visitor to the website is actually human left the voting site open to abuse. Users could submit multiple votes, votes on other people's behalf, or votes from people who don't even exist. In this case, user verification is key; the site should have been built in a way to validate that the user is eligible to vote and ensure they can do so only once.



See screen captures below of users posting to 4chan of their successful voting fraud.



We can see the above script generating fake email addresses with what appears to be valid .ru mail domains, but using the same name Kim Il Sung.

REMEDIATION

Typically, with bot traffic, you'll see an increased load on your network, your webserver, and your backend database. In order to combat this, you could purchase extra resources to beef up your infrastructure's ability to handle the traffic load or buy a black box appliance to monitor your network and thwart unwanted traffic. Unfortunately, the bot traffic will still be consuming your network resources in order for you to analyze it. Extra hardware could be purchased either specific to bot mitigation or to handle the additional load of bots scraping your site. Fortunately, Akamai already has a solution that moves the traffic off of your network.

With Akamai's large deployed network, we are able to see a large portion of Internet traffic. This is then leveraged to categorize different types of bots, which customers can use to define their own bots based on traffic they are seeing on their website. Akamai analyzes the known bot traffic on a weekly basis and we update our categories often, correlating botnet activity across industry verticals. Botnets that have gone defunct are retired after a cooling period, while new bots are added and categorized.

Rate Controls

The rate at which bot machines are making connections and requesting pages is often faster than any human could feasibly match. Real users have to type or click on a link, which is comparatively slow, not hundreds of requests per minute. Triggering based on the number of requests per minute is the first line of defense when determining if a request is automated or human.

Reputation

IP addresses can be flagged as malicious or originating from known botnet networks, either good or bad, and can be used to determine how traffic should be handled. Malicious traffic can be blocked, while well-behaved bots can be given higher priority.

Cross-time/space analysis

Akamai's Bot Manager evaluates connection request fingerprints and categorizes bots based on their behavior and active time of day. It then clusters them into a single entity. With that, a customer can evaluate the cluster's actions as a whole. In other words, if certain connections are made at specific times of day and they only visit the same web pages each time, those IP addresses can be categorized as a bot.

Signature based (User Agent)

Many botnets use known User-Agent headers. Comparing reported User-Agent request header strings against a known valid list of User-Agent strings and other anomalies can be used to cluster traffic together.

Anomaly based

Request headers can be examined for order, typos, format, double spaces, and other quirks that might be unique to a certain bot network.

Bot vs. Browser

JavaScript libraries can force the system to perform math problems to determine if the requestor on the other end has the capabilities of a normal web browser. Many bots do not have the full capabilities required in order to return the answer to the problem, having been pared down to the minimum capabilities needed to make the bot work.

Behavioral analysis

Work flows on how a website should be utilized can be compared with what is observed when a connection is made to a site. A normal user will likely browse the site before logging in and making a purchase, whereas a bot won't.

Conclusion

We see all types of bot traffic across our network. Getting your site indexed by a major search engine is welcome. However, the vast majority of this type of automated traffic is not as welcome, regardless of whether it is actively malicious or merely annoying. This traffic may cause unwanted load on your website's backend infrastructure and network, utilizing resources you'd rather have allocated to legitimate site visitors. Through traffic analysis and screening, you can mitigate the impact of bots on your web application and database operations. Akamai's Bot Manager product utilizes all of the identification methods covered above and more to mitigate automated traffic to websites. In our next installment, we will discuss stealthy bots — bots that do their best to fly under the radar past rate controls and look like legitimate user requests. For more information, please visit www.akamai.com/bot-manager