Adspect Documentation

Adspect

Apr 11, 2021

Contents:

CHAPTER 1

Overview

1.1 What is Adspect

Adspect is an easy-to-use cloud-based service for protecting affiliate campaigns (CPA offers, landing pages) from "bad" traffic. By bad traffic we mean:

- **moderators and policy teams** of ad networks
- click fraud, ubiquitous in display ads and popunder
- spy services used by competitors to steal your creatives and landing pages
- content scrapers
- credential stuffing bots
- bots of antivirus companies
- and other flavors of unwanted or outright hostile visitors

You may find additional information in our FAQ.

1.2 Traffic Sources

We work with all traffic sources, both existing and those that will appear in future–our filtering algorithms are perfectly universal and equally efficient across all possible origins of traffic. We support all the largest advertising networks, including:

- **Facebook and Instagram**
- **Google Ads**
- **TikTok**
- **Microsoft Advertising (Bing Ads)**
- Yandex.Direct

- myTarget
- VK
- ZeroPark
- ExoClick
- Taboola
- MGID
- TrafficStars
- **and hundreds of others**

We also protect your landing pages and offers from various antivirus, security, and ad scoring companies, including:

- **Google Safe Browsing**
- **GeoEdge**
- Integral Ad Science
- Kaspersky Labs
- Avast
- Forcepoint
- residential and mobile proxies, **including Luminati and GeoSurf**
- and many others

1.3 Integration

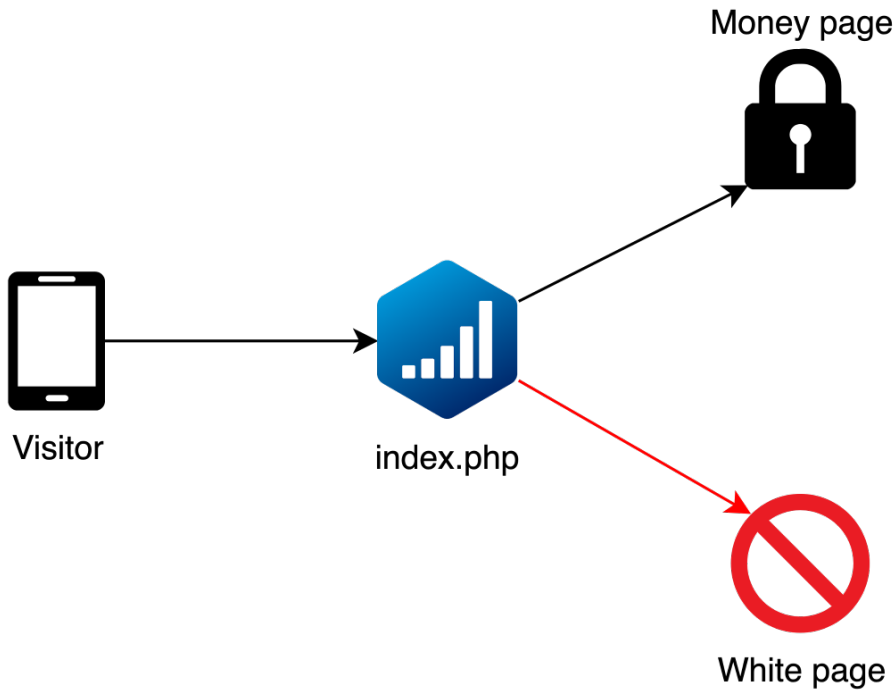
We support several types of integration that differ in technical details and use cases:

- Forward PHP integration via a standalone `index.php` file
- Reverse PHP integration via including a `filter.php` file
- JavaScript integration via `<script>` HTML tag embedding using a remote `ajax.php` file

More details will be given later in the *Integration* chapter.

1.3.1 Forward PHP Integration

In forward PHP integration filtering is done by a special `index.php` file that you place in your landing page directory or elsewhere accessible via HTTP. This file acts as an entry point for web traffic and is wired to our servers that process clicks and make decisions. Depending on filtering decision a visitor may be directed to your actual page or to a "white page", that is, a page that contains no sensitive content. In other words, Adspect acts as an intermediary stage in your traffic flow, actively filtering unwanted traffic from legitimate visitors.

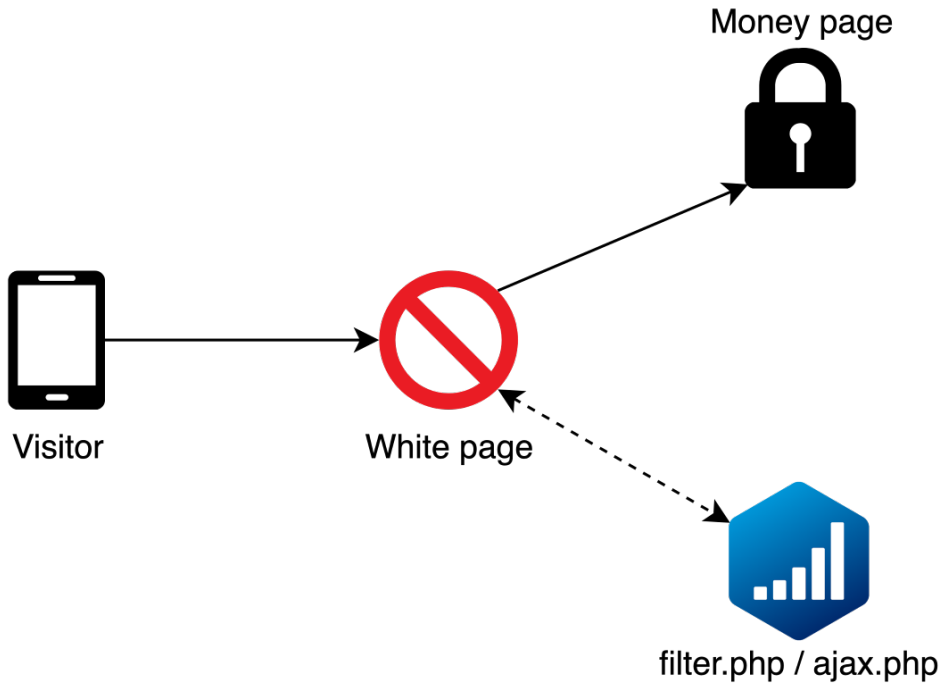


Traffic flow chart

Forward PHP integration is the most common type of integration.

1.3.2 Reverse PHP Integration

There's also a slightly different reverse PHP integration that uses a `filter.php` file which is included into your PHP page file (normally your white page) via a single line of PHP code. Traffic lands directly on this page, our code in the `filter.php` file inspects it and chooses either to keep the visitor on the page or display a different one.



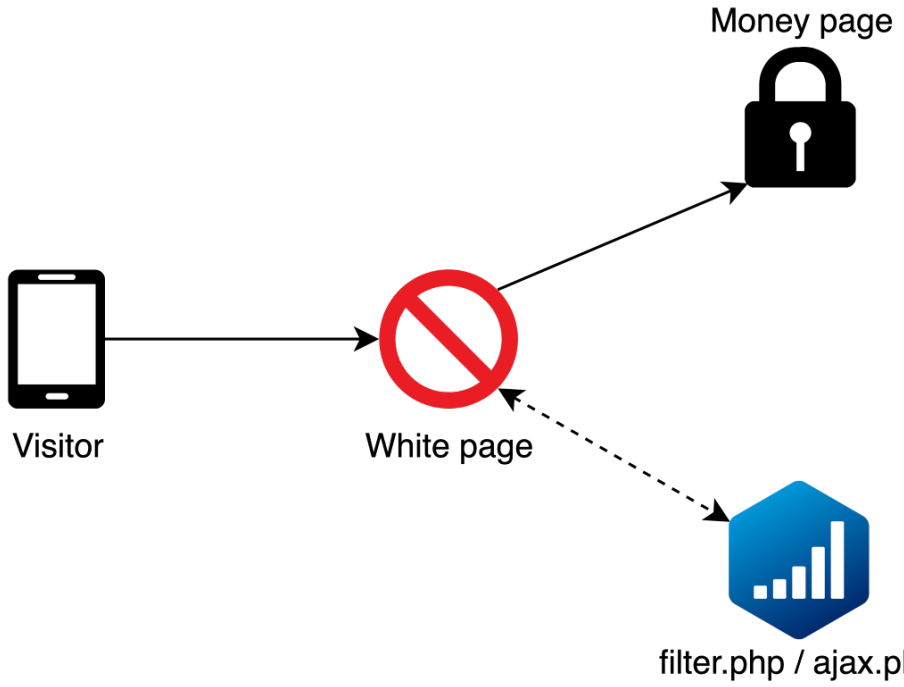
Traffic

flow chart

Reverse PHP integration is useful for integrating Adspect into sites based on WordPress or similar CMS (content management systems.)

1.3.3 JavaScript integration

JavaScript integration is meant to be used with third party services like Shopify, Blogspot, or Tilda, where you cannot upload custom PHP files to do PHP integration. Traffic flow is much like in reverse PHP integration: visitors come to the white page first, then legitimate ones are displayed the money page whereas moderators and bots are left where they are.



Traffic flow chart

You will also need to download a PHP file called `ajax.php` and host it somewhere, but its final location does not matter as it will be linked into the white page using a `<script>` HTML tag.

1.4 Adspect PHP Files

The fact that we use PHP scripts to filter traffic naturally implies that you need a PHP-enabled web hosting provider or a tracker with support for landing pages written in PHP. Our files are referred to throughout the documentation as `index.php`, `filter.php`, and `ajax.php`, but you may rename them however you like.

The code is carefully written to be compatible with a wide variety of web hosting environments, ranging from virtual hosting and VPS to dedicated servers and Amazon AWS. Both Windows and Unix-like operating systems are supported, to the extent supported by PHP. **PHP 5.4 or higher is required!**

The only requirement is that PHP needs to have cURL support. You may check if cURL is supported by examining phpinfo. Usually cURL support may be enabled by installing the `php-curl` package.

Do not use Namecheap shared hosting! Please see the *Best Practices* chapter for details.

1.5 Workflow

Common Adspect workflow for affiliate marketing campaigns consists of the following steps:

1. *Create an Adspect stream* for your campaign;
2. Choose an appropriate integration method and follow instructions on the integration page;
3. Set the stream to the All Money mode and test it to make sure that money page is displayed correctly;
4. Set the stream to the All White mode and test it to make sure that white page is displayed correctly;
5. Set the stream to the Filtering mode and test it to make sure that **you get forwarded to money page** without any errors (temporarily disable any manual filters that may block you);
6. Set the stream to the On Review mode;
7. Create an ad campaign and submit it for review;
8. Wait for campaign approval and switch the stream into Filtering mode;
9. Run traffic and explore statistics in the *Reporting section*.

CHAPTER 2

Traffic Filtering

There are several approaches to detecting and filtering unwanted visitors in web traffic. In this chapter we will touch upon the basics of all of the major techniques of automatic filtering and explain what makes Adspect innovative and unique on the market.

2.1 Blacklisting

This is the most primitive, naïve, and widespread approach. It normally involves comparing a narrow set of features of a visitor (IP address, HTTP request headers, etc.) against a pre-collected blacklist. A match signals that the visitor should not be allowed further. While popular, this approach suffers from two major flaws:

1. Blacklists are never exhaustive and thus are trivial to circumvent, e.g. by cycling through a very long list of available IP addresses during each campaign review, as often facilitated by specialized proxy services. One cannot blacklist everything, there will always be wide gaps that allow malicious parties to get through. There are entire companies that do their business by maintaining vast pools of clean residential IP addresses ready for use for a fee. Maintaining up to date blacklists of these proxy IP addresses is infeasible.
2. Blacklists may be too broad, yielding false positives. This is especially bad with IPv4 address blacklists. The rather narrow 32-bit IPv4 address space has been exhausted, prompting Internet service providers and carriers to employ NAT (network address translation) to aggregate entire networks of subscribers behind a single shared IP address. This means that blacklisting, say, a single shared residential IP address under suspicion of proxy (yes, there are ways to maintain proxies behind NATs) in a large metropolitan area leads to blacklisting thousands of legitimate potential visitors and high bounce rate.

Blacklisting is the most common–and often the only–approach used by cloaking services in the affiliate marketing domain. While a viable solution in many cases, it is rough and unreliable and cannot be used on its own. Blacklist false negatives are the most common reason of cloaking faults. Adspect maintains massive built-in IP address blacklists of positively bad traffic sources that count up to one billion addresses.

2.2 Fingerprinting

Fingerprinting is, as the name suggests, the process of collecting a fingerprint of a visitor that identifies them. However, unlike human fingerprints that are universally unique, machine fingerprints aren't unique. Depending on implementation, they are composed of varying numbers of features, some of which are very common, like user agent strings of popular browsers. But some of the less common features happen to indicate with high accuracy the exact "bad" traffic that we protect against. And we know which.

Fingerprinting is a much more advanced technique normally used by business-oriented fraud protection companies. You may see their services employed, in particular, by value-added services (VAS) providers, protecting mobile "wap-click" offers from click fraud. Adspect is proud to call itself the pioneer of fingerprint scanning in the adtech industry.

Adspect has great expertise in JavaScript fingerprinting, that is, analyzing fingerprints composed of features of the visitor's JavaScript execution environment. Our fingerprints usually consist of 1600 to 2200 different facts, giving us a very detailed view into every visitor's internal works. We run collected fingerprints against dozens of high-precision tests that allow us to detect malicious visitors with unmatched accuracy. Adspect aims to bring high-end fraud protection into the realm of affiliate marketing.

2.3 Machine Learning

Machine learning (ML) is a broad term colloquially referring to making computers learn and then use what they have learned to do their task. With respect to traffic protection, machine learning is used to analyze the features of each visitor to classify them as either legitimate or malicious. This can be done with great precision, given enough information to teach the learning model.

Machine learning makes a perfect solution for inspecting fingerprints. Adspect is powered by a proprietary machine learning technology called VLA™, constantly trained to detect features of bad traffic well beyond the criteria initially built into it. Please refer to the *VLA chapter* for a detailed discussion.

Machine learning is the rocket science used by very few companies on the market, all of them being the big whales on the anti-fraud market. Adspect is the first vendor to bring machine learning into affiliate marketing.

2.4 Our Approach

Adspect employs all three of these techniques together without relying wholly on any single one. This allows us to make accurate decisions with the lowest rates of false positives and false negatives. We firmly believe that extensive fingerprinting coupled with machine learning appliances will play the leading role in defensive adtech because of the immense potential of both technologies, especially if combined.

CHAPTER 3

VLA™

VLA™ stands for "Virtual Learning Appliance." It is the trademark of our machine learning technology that powers the most advanced filtering capabilities of Adspect. In simple terms, it is a self-adapting mathematical machine that observes incoming traffic and finds suspicious recurring patterns in its fingerprints (thousands of features in every fingerprint) that indicate moderators, fraud, and other malicious activity. VLA constantly teaches itself, evolving and adapting to new types of threats as they emerge. We believe that VLA is our strongest weapon in the race of arms of affiliate marketing as it is able to see well beyond what we initially put into it. What a human analyst may overlook will never escape the mathematically strict scrutiny of a carefully programmed machine.

The concept behind machine learning is best described by analogy. Suppose a policeman at an airport is instructed to detain all passengers with a specific tattoo as they are known to be part of a dangerous gang. The policeman detained ten such persons during the last month, each time noticing that they all were also wearing T-shirts with the same symbol as on their tattoo. Now, the policeman will also stop people wearing those T-shirts under the same suspicion, regardless of whether they have the tattoo.

Whereas fingerprint checks yield a close to 100% confidence in that a given fingerprint belongs to a bot (moderator, spy service, etc.), VLA is inherently probabilistic in nature. The real deal here is that fingerprint checks encompass only those threats that we already know of while VLA detects previously unknown dangers. It takes a fingerprint, inspects every feature encoded in it, and yields a confidence percentage, as if saying, e.g., "I am 97% sure that this fingerprint belongs to someone you better filter out!"

Now, it only remains to determine what confidence is high enough to trigger the filter, and the choice is yours where to draw that line. The VLA section of every stream has a "VLA precision" setting that serves that very purpose: you specify the minimum confidence that you require VLA to have in order to filter out a visitor. For example, if you set VLA precision to 95%, then VLA will filter out all visitors for which it yields certainty of 95% and above, but will let through those that it is less confident about. This single precision parameter lets you fine-tune the system in accordance to your own idea of what is "confident enough". Our tests have shown that 95% is a good value to begin with.

CHAPTER 4

Use Cases

Adspect has a few well-defined use cases that have proven to be consistent and useful. Remember that Adspect has two intertwined but still distinct functions: cloaking and bot filtering. The latter helps a lot in achieving the former. We will describe the benefits of both just below.

4.1 Cloaking

Cloaking is the practice of hiding the real web page, be it a landing page or a CPA offer, from those who should not see it, at the discretion of the one who is in control of that page. We at Adspect firmly believe that if you do not wish to expose your content to a certain party, then you should be able to limit access to it, regardless of your reasons. We give you the power to do so. In particular, this means hiding your landing pages from ad network moderators, spy services, and antivirus robots. Those visitors will never make you any money.

4.2 Detecting Bot Zones

Popular purely web-based ad formats like banners, teasers, native ads, and popunders all suffer from click fraud. Technology they are based on–HTTP, HTML, and JavaScript–allows for relatively easy and cheap automated clicking, just pick any of the programmable headless browsers out there. No wonder these browsers, initially meant for website testing automation, are now widely used to forge clicks in ad networks and make advertisers pay for what will never bring them any revenue.

Adspect can detect all of them with ease. All you have to do is configure a stream to parse the subaccount ID out of the tracking URL as described in the chapter on streams. If you pass a publisher, site, or zone ID (let's call it a *source* from now on) to Adspect via a link parameter, then you will be able to pull per-source reports with exact figures of human ratio in the traffic. The rightmost "Quality" column in reports will let you evaluate and compare different sources, showing the percentage of legitimate traffic among the whole. Just select "Sub ID" in the grouping list to the left of the timezone picker.

Drawing a line at, say, 80% of humans in traffic, you can easily find sources that meet the requirement–just click the "Quality" column header to sort the table by that column. Sources with quality above 80% will give you a whitelist; conversely, sources with quality below 80% will be your blacklist. You will find this simple method invaluable for

determining converting traffic sources in display ads and popunder without spending fortunes on filtering sources by sheer CR (conversion rate.) Filter out bot-ridden sources first, then filter the rest by CR as you would normally do.

4.3 Hiding Traffic Sources

Many affiliate networks have internal media buying teams that would be all too happy to discover your traffic sources and use this information to steal your campaigns. Therefore you would normally want to hide your sources from affiliate networks, or any other parties, for that matter. Adspect does this for you by removing the Referer HTTP request header from all clicks, making sure that observers on the traffic redirect chain will not see the source of your traffic in web server logs.

CHAPTER 5

Configuring Streams

Traffic management in Adspect is organized in terms of streams. A stream is a traffic channel that is managed as a whole, much like a campaign in an ad network or a scheme in TDS. Streams are managed in the Streams section of the clients area. Use the New Stream button to create new streams. Below we will visit each stream setting in detail.

Please note that default settings are generally adequate for most traffic sources and use cases. You are by no means required to fill in all the available fields; normally, it is enough to configure just money and white pages and leave the rest to Adspect.

5.1 Name

Stream name is just a human-readable identifier that lets you distinguish between different streams. It is a good idea to match stream names with ad campaigns on one-to-one basis to maintain consistency and clarity across your traffic sources and Adspect. We also recommend that you create one stream per GEO (country) to make obtaining per-GEO statistics easier.

5.2 Mode

This is the mode that streams currently operates in. There are four modes:

- **Filtering** – this is primary working mode in which we actively inspect every click coming in the stream and filter legitimate visitors from moderators, click fraud, and other unwanted types of traffic. All filtering technologies of Adspect, including *VLA™*, work only in this mode.
- **On review** – this mode is meant to be used when ad campaign that points to the stream is on review by ad network moderators. Every visitor in this mode will be directed to the white page. There are additional settings that apply in this mode, they will be described below.
- **All money** – auxiliary mode in which all visitors are directed to the money page. Useful for testing accessibility of the money page.

- **All white** – auxiliary mode in which all visitors are directed to the white page. Useful for testing accessibility of the white page. It is also a good idea to put streams into this mode whenever campaigns are paused in ad networks, to prevent unauthorized access to your landing pages or offers during inactivity periods.

On review is the default mode when creating a new stream. You *should* always use it when sending campaigns to moderation. After a campaign is approved, you should change its stream mode to Filtering before actual traffic starts coming.

5.3 Money Page

This is your actual landing page or offer that you are going to advertise. The "money" word is intended to indicate that this is the page that makes you money. You may specify up to 254 money pages for A/B testing. Traffic will be distributed across them according to rules of a particular rotator; see the *Rotator paragraph* below.

There are two types of values that may be specified: page file name or an URL. Page file name is the advised way of specifying money page–it is the name of an HTML or PHP file of your real landing page that *must* be located in the same directory as where you put `index.php` after stream creation, i.e. in the root directory of your landing page.

The file name should not be easily guessable because it lets determined moderators or competitors figure out the URL of your real landing page. Pick a random long file name.

Do not name your money page `index.html` or `index.htm`! Apart from being easily guessable (i.e. trivially uncloakable), those file names may be in conflict with your existing web server configuration, leading to unforeseen problems.

To put it all together: if you have a landing page directory and the actual landing page file inside named `index.php` (as is most often the case), then you should first rename that `index.php` file to something hard-to-guess like `re3NBX1XtH.php`, then put our special `index.php` file next to `re3NBX1XtH.php` in the same directory after stream creation. Our `index.php` will then display the real file `re3NBX1XtH.php` to approved visitors.

Alternatively, you may use an URL instead, for instance a direct offer URL taken from an affiliate network. This may be optimal for some campaigns, however, external URL implies an additional HTTP redirect, with associated latency and traffic loss considerations, especially on low-quality ad formats like popunder.

You may also use various non-HTTP URLs to achieve specialized tasks on your visitors' devices. Some of the more common examples:

- `mailto:user@example.com` will open up a default e-mail program in compose mode;
- `tel:+08001234567` will dial the number on mobile devices and some desktops with installed telephony software;
- `market://details?id=app` will bring the visitor to a particular app's page in Google Play.

This is particularly useful with the so called "deep links" that link to mobile in-app content.

5.3.1 Param Setting

Param stands for "URL parameters passthrough." When enabled, parameters passed in the incoming URL will be appended to money page URL or file.

For example, consider the stream's money page is configured as follows:

```
https://example.com/?utm_campaign=sweeps
```

A visitor accesses `index.php` of the stream using the following URL:

```
https://tracker.test/lander/index.php?utm_medium=ppc&utm_source=search
```

If the visitor is considered legitimate, they will be redirected to the money URL with URL parameters combined from both of the above:

```
https://example.com/?utm_campaign=sweeps&utm_medium=ppc&utm_content=search
```

5.3.2 Weight Setting

Each money page has associated abstract "weight" that defaults to 10. This setting is taken into account when you have more than one money page in A/B testing. Exact meaning of this parameter depends on the rotator used by the stream; see the *Rotator paragraph* below.

5.3.3 On Setting

The On checkbox allows you to turn individual money pages on or off.

5.3.4 URL Macros

Adspect supports several URL macros that you can use in money and white page fields:

- `{ip}` – IP address of the visitor;
- `{asn}` – autonomous system number of the visitor;
- `{agent}` – user agent string of the visitor;
- `{referrer}` – referrer of the visitor;
- `{clickid}` – unique click ID (external ID from URL parameter or generated by Adspect);
- `{country}` – ISO 3166-1 alpha-2 country code of the visitor;
- `{os}` – operating system of the visitor with version for Windows and Android;
- `{browser}` – name of the browser of the visitor;
- `{engine}` – name of the browser engine of the visitor;
- `{epoch}` – Unix timestamp of the visit;
- `{tags}` – click processing tags, if any;
- `{p:parameter}` – value of the named parameter in the request URL.

In zero redirect (file-based) display mechanism, you may still add parameters with macros after the file name of your money/white page, and they will be parsed and made available in PHP in the `$_GET` superglobal variable.

Redirect example:

```
https://example.com/offer?clickid={clickid}&geo={country}&os={os}
```

Zero redirect example:

```
page.php?clickid={clickid}&geo={country}&os={os}
```

```
<!-- Inside page.php file -->
<a href="https://example.com/offer?clickid=<?= $_GET['clickid'] ?>">Offer</a>
```

5.4 Rotator

Rotator controls how several money pages are rotated, i.e. how the system decides which money page to display to each particular visitor. If there is only a single money page specified, then rotator choice does not affect anything. Currently Adspect supports two rotators: Split and Timer.

5.4.1 Split Rotator

This is the default rotator that splits traffic across all enabled money pages according to their weights (A/B testing.) The more the weight is, the more traffic that money page will receive, proportionally.

For example, if you have three money pages with weights 10, 15, and 25, then the first page will receieve about 20% of all human traffic, the second page will receieve 30%, and the third page will get around 50%.

Since this rotator is based on a pseudorandom number generator (PRNG), there may be distribution bias on small scale. However, mathematical properties of the PRNG guarantee that distribution will reach target weights on distance.

5.4.2 Timer Rotator

Timer rotator cycles through enabled money pages, using weight as a number of seconds that a money page is active for.

For instance, if you have three money pages with weights 60, 120, and 180, then the first page will be shown to visitors for 1 minute, then the rotator will cycle to the second page and display it to incoming clicks for 2 minutes, then go to the third page and use it for 3 minutes, then cycle back to the first one, and so on.

This rotator is useful for automatic time-based switching of domains.

5.5 White Page

This is the safe page to show to moderators, robots, scrapers, etc. It should not contain any sensitive content that may put your affiliate campaign in danger or in violation of any rules. Everything described above for the money page also applies to the white page–you may use an URL or a landing page file. In the latter case, if your money page is also a self-hosted landing page, you will need to effectively merge two landing page directories together to meet the requirement that both page files are located in the root of the common directory.

We strongly advise using a landing page file over external URL for white page. This has to do with suspicion and increased scrutiny from compliance teams of certain ad networks that discourage or outright prohibit the use of redirects in traffic flow.

5.6 Pass URL Parameters to White URL

When enabled, parameters passed in the incoming URL will be appended to white page URL or file. This setting works the same way like the Param setting of money pages.

5.7 VLA™

VLA™ stands for Virtual Learning Appliance, the trademark of the machine learning system at the heart of Adspect. It is discussed in detail in the *VLA chapter*. 95% is a good VLA precision value to begin with.

5.8 HyperLogLog

HyperLogLog is an advanced algorithm used to estimate cardinality of large sets. It is used in the eponymous state of the art filter that we invented to perform pattern-based filtering in real time based on our entire operation history. Lower values result in more aggressive filtering and thereby better protection at the expense of higher chances of false positives.

We recommend to set it to 1 when working with Google Ads and TikTok!

5.9 Sub ID

Sub ID refers to an URL parameter that you want to use for per-subaccount reports, available in the Reporting section by selecting the Sub ID grouping. Please refer to the *Reporting* chapter for details.

The concept is best described by example. Suppose your ad network has a notion of zones for dividing different publishers or ad placements into numbered groups. You would use some form of macro, e.g. `{zoneid}`, to put zone identifiers into your click URL. Your campaign tracking link might look like this:

```
https://tracker.test/lander/index.php?subid={zoneid}
```

For each click, the ad network will replace the `{zoneid}` macro with an actual identifier which can then be taken out of the click link and tracked individually. In this example, `subid` is name of the parameter used to track zone IDs. If you specify `subid` for the Sub ID stream setting, then you will be able to pull per-zone reports in the Reporting section of the clients area as mentioned above. This may come in very handy for building blacklists of bot-ridden publishers, zones, placements, etc.

Sub ID may also be anything else: GEO, hardware platform, OS version, any URL-trackable parameter. You can also combine several parameters into a compound subaccount by using more than one macro in the same parameter:

```
https://tracker.test/lander/index.php?subid={zoneid}-{platform}
```

In the example above each subaccount will be a combination of a zone and a device platform seen within that zone.

5.10 Click ID

Click ID works the same way as Sub ID, but for tracking unique click identifiers often supplied by ad networks or affiliate trackers. If the Click ID setting is specified, click IDs are taken out of that link parameter and recorded in statistics along with other click data. This allows you to find and examine individual clicks in raw CSV reports. One use case would be compiling lists of bot clicks as a proof of click fraud.

If the Click ID setting is omitted, then Adspect will generate its own click IDs for use with its *tracker*. Then it may be communicated to the money or white page via the `{clickid}` URL macro.

5.11 Paranoid Mode

The paranoid mode enables additional strict fingerprint checks and vast IP address blacklists (counting up to 2 billion IPv4 addresses) that are considered "paranoid", that is, with higher false positives chance, but at the same time providing equally higher chance of busting moderators.

We recommend to enable this mode when working with Google Ads and TikTok!

5.12 Allow Traffic From Mobile Apps

This setting tells us to allow traffic that originates from inside mobile applications, e.g. from WebView Android browser. While natural for certain niche ad formats, such traffic is widely seen as click fraud in other formats (automated clicks generated by mobile malware) and should normally be disabled unless your ad format is somehow based on mobile applications.

5.13 Allow Traffic From Frames, Iframes, and Embedded Objects

This setting tells us to allow traffic that originates from inside embedded environments such as `<iframe>`, `<embed>`, `<object>`, etc. Similarly to traffic from mobile apps, this setting should be set or cleared depending on your particular traffic format and source. Keep it enabled if unsure.

5.14 Countries, Operating Systems, Browsers, Engines, Languages, and Time Zones

These manual targeting options allow you to further restrict your stream to only allow visitors from specified countries, using specified operating systems, browsers, browser engines, browser language preferences, or time zones. You would normally set them to match your campaign targeting settings. If any of these settings is omitted (the list is empty), then no check will be made for that setting.

Note: time zone settings are restricted to full hour offsets from UTC. If a visitor's time zone is not offset by full hours, then the offset will be rounded.

5.15 Match Browser Time Zone to Location Time Zone

If this setting is enabled, then Adspect will check whether the time zone reported by visitor's browser matches the time zone of the visitor as determined by our geolocation. This check may slightly increase the rate of false positives, but it significantly boosts protection against moderators and bots that use VPN or proxy services. If enabled, the manual time zone list described above is ignored. It is recommended to enable this setting.

5.16 Schedule

Schedule allows you to specify dayparts and optionally days of week during which traffic filtering is on. All visits on time and days not explicitly listed will be blocked. Schedule is active if at least one daypart is specified. If a daypart does not specify days of week, then it is applied to all days.

5.17 Click Limit

This is the maximum number of clicks per each IP address that will be allowed. 0 means no limit. Visitors from IP addresses that have exceeded the limit will be filtered out. The "Reset" button may be used to reset all click counters for the stream.

5.18 Blacklist IP Addresses upon Hitting the Limit

If this setting is enabled, then all IP addresses that exceed the click limit will be added to the IP/ASN blacklist (see below.)

5.19 URL Rules

This section allows you to create up to 30 custom rules for checking and manipulating URL parameters. Each rule consists of:

- Parameter name – this is the name of the URL parameter that will be checked or altered;
- Operator – specific check or operation that will be executed;
- Argument – argument of the operator, if applicable (string interpolation macros supported);
- "On" checkbox – toggle that allows you to turn rules on or off.

The following operators are supported:

- `EXISTS` – checks if parameter exists (rule argument ignored);
- `! EXISTS` – checks if parameter does not exist (rule argument ignored);
- `REGEX` – checks if parameter value matches a Perl-compatible regular expression (PCRE) in rule argument (case-sensitive);
- `REGEX (no case)` – checks if parameter value matches a regular expression in rule argument (case-insensitive);
- `! REGEX` – checks if parameter value does not match a regular expression in rule argument (case-sensitive);
- `! REGEX (no case)` – checks if parameter value does not match a regular expression in rule argument (case-insensitive);
- =, , >, , <, – compare parameter value with rule argument; integers and real values are compared as numbers, strings are compared according to lexicographical order;
- `ASSIGN` – assigns rule argument as parameter value;
- `RENAME` – renames parameter to rule argument;
- `DELETE` – deletes parameter (rule argument ignored).

The order of rule execution is as follows:

1. Checks: `EXISTS` and `REGEX` rules, =, , >, , <, – failed check sends to white page;
2. `ASSIGN` rules;
3. `RENAME` rules;
4. `DELETE` rules.

Rule argument supports all the same string interpolation macros available for money/white page settings.

5.20 User Agent Filter

This setting allows you to specify a custom Perl-compatible regular expression (PCRE) for filtering visitors by their user agent string. Regular expression matching is case-sensitive. By default, the search is done in any part of the user agent string; you may use anchors to bind matching to the start or the end of the string (see examples below.)

PCRE syntax is very rich and powerful and is well out of scope of this document. Regular expressions can be combined using various syntax constructs to create arbitrarily complex patterns, but please note that the current implementation limits regular expression length by 1023 characters.

Some examples:

```
Firefox|Nexus|Miui
```

This regex will match any user agent that contains words “Firefox”, “Nexus”, or “Miui”, and can be used to filter out visitors that use Mozilla Firefox, Google Nexus, or Xiaomi built-in browser.

```
^Mozilla/4[.]0
```

This regex will match any user agent that begins with “Mozilla/4.0”, banning shady visitors that report themselves to be very old browsers yet support contemporary JavaScript features (implied by being able to run our fingerprint collecting code.)

```
^Mozilla/5[.]0$
```

This regex will match user agents that are exactly “Mozilla/5.0”, blocking visitors without concrete browser, HTML engine, and platform information, which is very uncommon and suspicious.

All of the expressions above can be combined using logical “or” (i.e. to match the first expression *or* the second *or* the third) this way:

```
Firefox|Nexus|Miui|^Mozilla/4[.]0|^Mozilla/5[.]0$
```

Please be careful! Improperly formed regular expression can lead to erroneous matching and filtering of vast amounts of legitimate traffic. Use this setting only if you know what you are doing.

5.21 Referer Filter

This setting works similarly to the user agent filter described above, but deals with HTTP referer instead. It also takes a Perl-compatible regular expression and filters out all visitors whose referers match it. Regular expression matching is case-sensitive.

One common use case is filtering empty or non-existent referers. This can be achieved with the following regex:

```
^$
```

Please be careful! Improperly formed regular expression can lead to erroneous matching and filtering of vast amounts of legitimate traffic. Use this setting only if you know what you are doing.

5.22 IP Extrapolation

IP extrapolation allows you to control fuzzy matching of IP addresses against our internal blocklists. Larger values result in banning of more addresses adjacent to already blacklisted addresses, which results in better protection at the expense of higher chances of false positives.

It is recommended to set higher values when working with Google Ads and TikTok! Start in the 128–255 range and lower gradually if you suspect high false positives.

5.23 IP/ASN List Mode

Adspect supports traffic filtering using lists of IP addresses, IP address ranges, and/or autonomous system numbers (ASN). Two such lists exist: blacklist and whitelist.

The IP/ASN list mode controls how blacklist and whitelist combine to determine whether a particular visitor should be blocked. Two modes are supported:

- Black: a visitor will be blocked only if their IP address or ASN is in the blacklist and is not in the whitelist. As such, the whitelist may be used to override the blacklist. This is the default mode.
- White: a visitor will be blocked unless their IP address or ASN is in the whitelist and is not in the blacklist. As such, the blacklist may be used to override the whitelist.

Please note that whitelists do not override IP/ASN blocklists built into the system.

5.24 IP/ASN Blacklist

This is the blacklist. Both IPv4 and IPv6 addresses are supported, as well as CIDR and range notations. Examples:

- 192.0.2.1
- 192.0.2.0/24
- 192.0.2.0–192.0.2.255
- 2001:db8::1
- 2001:db8::/112
- 2001:db8::-2001:db8::ffff
- 721
- AS721

Individual entries should be delimited by newlines or whitespaces. Please note that the system will automatically merge adjacent or overlapping ranges in order to optimize storage space and lookup speed.

5.25 Blacklist All IP Addresses in Review Mode

If enabled, this setting instructs Adspect to add IP addresses of all incoming visitors to the IP blacklist if the stream is in Review mode. Since the Review mode is meant to be used only when your ad campaigns are under review by moderators, it is safe to assume that every visitor in this mode is a moderator and should be barred. We recommend you to always enable this setting, but pay attention to the moment your campaign is approved, to switch the stream mode to Filtering in time lest you blacklist IP addresses of legitimate visitors when your campaign goes live.

5.26 IP/ASN Whitelist

This is the whitelist. It has exactly the same input format as the blacklist.

CHAPTER 6

Integration

After creating a stream you will be immediately brought to the integration page. Adspect supports several types of integration that differ in technical details and use cases:

- Forward PHP integration via a standalone `index.php` file
- Reverse PHP integration via including a `filter.php` file
- JavaScript integration via `<script>` HTML tag embedding using a remote `ajax.php` file

6.1 __sid

Each stream has its own `index.php`, `filter.php`, and `ajax.php` files wired to it that have the stream ID encoded inside. However, you may override that encoded stream ID and send a click to a different stream by putting the destination full stream ID into the `__sid` URL parameter, e.g:

```
https://example.com/index.php?__sid=1ea85c7c-b977-6804-8e69-00162501c2b4
```

You may find stream ID next to its name in the streams list.

If you need use a different parameter name instead of `__sid`, then open Adspect PHP file in a text editor and replace the `__sid` string with the desired name (e.g. `utm_campaign`.)

6.2 PHP Integration

PHP integration comes in two flavors: forward and reverse. They differ only in how our PHP files are wired with your *locally hosted* landing pages, i.e. which file receives incoming traffic.

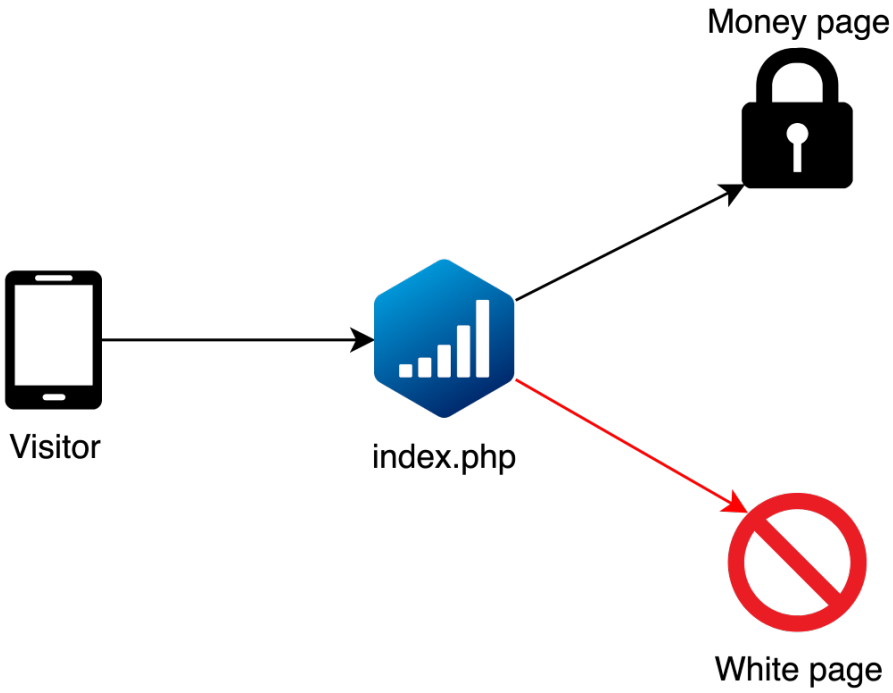
Both integration types support several methods for displaying links to external sites, that is, money and white pages specified as URLs in stream settings:

- HTTP redirect – regular redirection to the remote URL via HTTP 302 status code. This is the usual choice in most cases. **If you don't know which display method to choose, then go with HTTP redirect.**

- HTML iframe – display the remote URL on your domain inside an `<iframe>` tag. Please note that websites may forbid displaying their content inside an iframe by using the X-Frame-Options response header.
- Reverse proxy – display the remote URL on your domain by HTTP request proxying. This method also suffers from several technical complications and should be used only if no other method fits.

6.3 Forward PHP Integration

In forward PHP integration filtering is done by a special `index.php` file that you place in your landing page directory or elsewhere accessible via HTTP. This file acts as an entry point for web traffic and is wired to our servers that process clicks and make decisions.



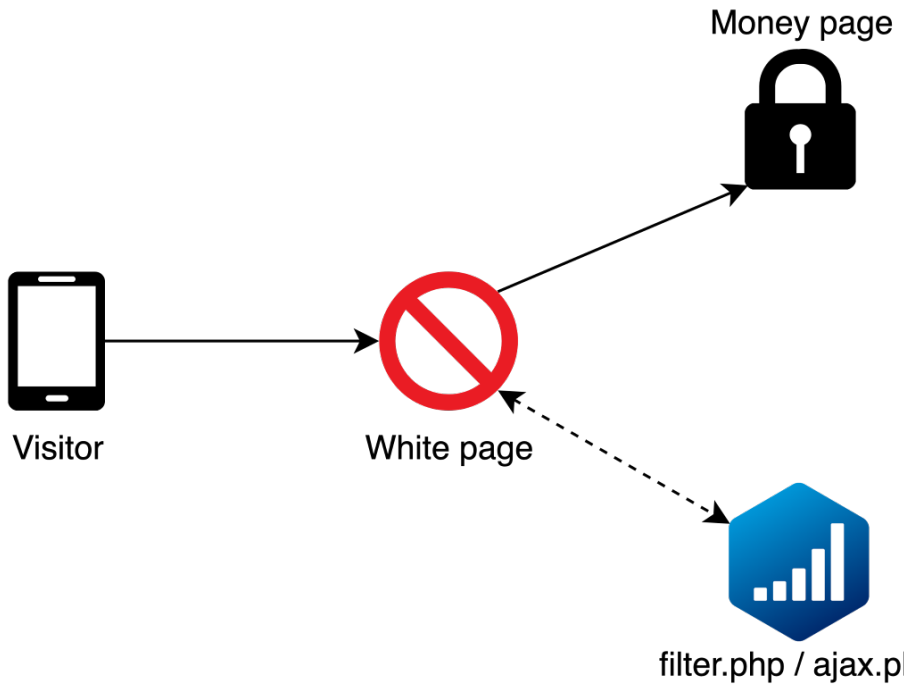
Traffic flow chart

After uploading the `index.php` file to your hosting its URL will be the cloaked URL suitable for use in advertising campaigns. Several copies of the same `index.php` file may be used for protecting several offers or landing pages without interfering with each other except for shared statistics.

Forward PHP integration is the most common type of integration. **If you don't know which integration type to choose, then go with forward PHP integration.**

6.4 Reverse PHP Integration

There's also a slightly different reverse PHP integration that uses a `filter.php` file which is included into your PHP page file (normally your white page) via a single line of PHP code. Traffic lands directly on this page, our code in the `filter.php` file inspects it and chooses either to keep the visitor on the page or display a different one.



Traffic flow chart

In order to perform reverse PHP integration you first need to download the `filter.php` file on the Reverse PHP Integration tab and put it into the folder of your site or landing page. Several copies of the same `filter.php` file may be used for protecting several sites or landing pages without interfering with each other except for shared statistics.

Then add the following code as **the first line** of your site or landing page index file (usually named `index.php`) above all other code:

```
<?php require __DIR__ . '/filter.php' ?>
```

If your site is written in pure HTML, then you may safely rename your `index.html` or any other HTML file to `index.php` or any other name ending in `.php` before adding the code to it.

If you added the code into your white page, then leave the White Page field empty in the stream settings. Empty field means "no action", i.e. Adspect will not take any action, leaving the visitor on their current page which is the white page. Likewise, if you added the code into your money page, then leave the Money Page field of your stream empty.

Once set up, you then simply direct traffic to the page you added the code into.

6.4.1 WordPress and Other CMS

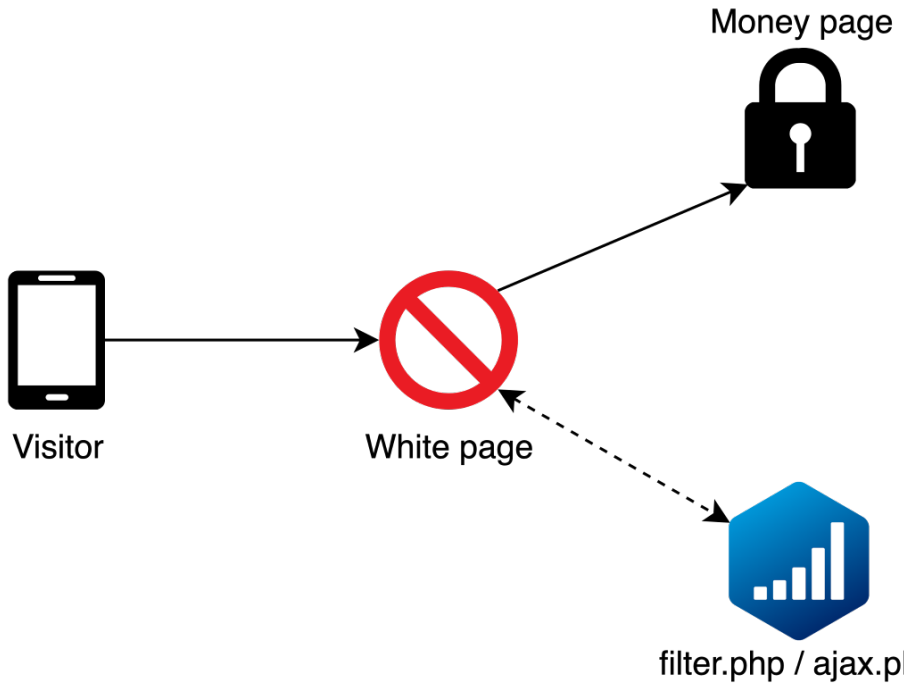
Reverse PHP integration is useful for integrating Adspect into sites based on WordPress or similar CMS (content management systems.)

WordPress has a file named `index.php` in its root folder. That is the file where you should add that single line of PHP code discussed above. Place the `filter.php` file into the same folder.

Most other PHP-based CMS software also has such `index.php` entryway files in their root folders.

6.5 JavaScript integration

JavaScript integration is meant to be used with third party services like Shopify, Blogspot, or Tilda, where you cannot upload custom PHP files to do PHP integration. Traffic flow is much like in reverse PHP integration: visitors come to the white page first, then legitimate ones are displayed the money page whereas moderators and bots are left where they are.



Traffic flow chart

You will also need to download a PHP file called `ajax.php` and host it somewhere, but its final location does not matter as it will be linked into the white page using a `<script>` HTML tag.

It is important to link `ajax.php` via HTTPS if the website you are integrating Adspect into also uses HTTPS (which is almost always the case.) Trying to link `ajax.php` to an HTTPS site via plain HTTP will result in mixed content error in most modern browsers, and cloaking will not work.

Like PHP integration, JavaScript integration also supports several modes of operation:

- In JavaScript redirect mode, legitimate visitors as determined by our filters will be directed to the money page via JavaScript redirect using the `location.replace()` method, i.e. the URL in the address bar will change. This is the usual mode. **If you don't know what to choose, then go with JavaScript redirect.**
- In iframe overlay mode, legitimate visitors will be shown the money page via an iframe overlay without redirecting them anywhere, i.e. the money page iframe will be placed over the white page. Please note that websites may forbid displaying their content inside an iframe by using the X-Frame-Options response header.
- In passive mode our statistics will be updated, but no further action will be taken–the visitor will remain on the page. This mode is like Google Analytics–perfect for collecting passive insights and blacklists of bot-ridden sources in cases that do not require cloaking.

Several copies of the same `ajax.php` file may be used for protecting several pages without interfering with each other except for shared statistics.

Please note that white page setting is ignored in JavaScript integration because visitors initially land on the white page, which is the page that our `<script>` tag is placed on.

6.6 Debugging

The most often observed error is 500 Internal Server Error, which is usually caused by either of the following misconfigurations:

1. File specified for zero redirect displaying of money/white page could not be found;
2. Your PHP installation does not have cURL support–you need to install the `php-curl` package.

All Adspect PHP files support debug mode. If enabled, any server-side errors will be displayed directly in browser window. In order to enable debug mode you need to open our PHP file in a text editor, find the following piece of code in the beginning of the file:

```
<?php
 define('ADSPECT_DEBUG', 0);
```

and replace `0` with `1` in it:

```
<?php
 define('ADSPECT_DEBUG', 1);
```

If your PHP integration does not work, e.g. you observe an HTTP 500 error, then you will see error details in debug mode. When debugging JavaScript integration, you should navigate directly to the URL of the `ajax.php` file. If you don't know how to fix a particular error, then please contact us in Telegram about it.

Do not forget to disable debug mode after all errors are fixed.

CHAPTER 7

Tracker

Tracker is an indispensable tool in digital marketing as a whole and affiliate marketing in particular. Its primary function is to register conversion events, that is, orders or sales, and track them back to specific visitors. This in turn allows marketers to gather conversion statistics and analyze it from different angles using various criteria, building what is known as conversion funnels.

Adspect is equipped with a lightweight yet efficient tracker built right into the core of the system. The *Reporting* section of the clients area allows you to explore funnels built with different groupings and filters. Among others, it calculates and displays such important metrics as conversions, cost, revenue, CR (conversion rate), ROI (return of investment), CPA (cost per action), and EPC (earn per click) / EPM (earn per thousand clicks.) These are especially useful combined with subaccounting by source identifiers as described in the previous chapter in the paragraph on Sub ID.

7.1 Postback

In order to use tracking you need to configure postback of conversion events to our postback URL located in your profile. We accept postback via any HTTP method: `GET`, `POST`, `PUT`, etc. The postback URL takes three parameters:

1. `aid` – Adspect account ID, which is pre-filled and normally will not change;
2. `cid` – unique click ID that identifies particular click that made a conversion;
3. `sum` (optional) – payout sum of the conversion in case of CPA or revenue share tracking.

Most affiliate programs and networks support postback and provide various macros that can be used to fill variable portions of the URL (`cid` and `sum` parameters.) If you need to fire postback manually, then you could place a conversion pixel somewhere, e.g. on a "Thank you for your order" page. For example, assuming that click ID is contained in the `clickid` link parameter:

```
<script>
(function () {
  const cid = new URLSearchParams(location.search).get("clickid");
  const url = "https://rpc.adspect.net/v1/postback?aid=1ea704aa-d0d3-6262-bf65-
↪ac1f6b95a853&cid=" + cid;
```

(continues on next page)

(continued from previous page)

```
  fetch(url, {mode: "no-cors"});
})();
</script>
```

This code will send a postback request immediately when the page is loaded. For a successfully registered conversion, the postback URL will return the HTTP status code 200 and `OK` in plain text.

7.2 Click IDs

For a conversion to be registered and processed it is required that Adspect has a record of the corresponding click in its statistics database as determined by its click ID. You may either use externally generated click IDs passed to Adspect from outside in a link parameter, in which case you should put the name of that parameter into the Click ID stream setting, or omit the Click ID stream setting and let Adspect generate click IDs automatically. Conversions with click IDs not previously registered by Adspect will be discarded.

CHAPTER 8

Reporting

Our reports are a comprehensive and valuable source of analytical information on your affiliate campaigns performance and traffic quality. You can use reports to evaluate traffic quality of different sources, publishers, ad spots, etc. Reports come in two flavors: raw and aggregate.

Please note that statistical data is not real-time and is updated once every minute.

8.1 Aggregate Reports

Aggregate reports are produced by splitting raw reports into groups and summing metrics on per-group basis. Grouping may be set set in the field to the left of the timezone selector and defaults to Stream, meaning that values will be computed on per-stream basis. Other grouping options include Date for per-date grouping and Sub ID for per-subaccount grouping (refer to this paragraph for an idea of what may constitute a subaccount.)

Grouping may be nested, e.g. Date followed by Stream–this will result in first splitting reports by date, and then by stream on each date. You may combine grouping criteria in any way you need to inspect different funnels.

Press the Get Report button to produce an aggregate report in the form of a table that will be displayed to you right below the report settings panel.

8.2 Aggregate Report Columns

Each aggregate report consists of the grouping columns on the left site followed by a number of statistical columns. Some of the statistical columns have a gray percentage value following a slash–that is percentage of the total clicks, displayed for more comprehensivess.

The list of statistical columns, explained:

- Clicks – the total number of clicks that accessed the `index.php` file.
- Uniques – approximate number of unique visitors are per uniqueness of their IP addresses.

- FP – the number of visitors that successfully executed our JavaScript fingerprint collector code and submitted their fingerprints for analysis. This figure may be lower than the total number of clicks for various reasons, most often it being the inability of dumb click bots to run JavaScript.
- Money hits – how many visitors have been shown the money page. This is the best metric for accounting legitimate traffic. Please note that this also includes all visitors when the stream works in the All Money mode.
- White hits – how many visitors have or would have been shown the white page. This metric is calculated as clicks minus money hits and includes those dumb bots that would have been show the white page if they were able to execute JavaScript (there's a fallback "meta refresh" mechanism to deal with them.)
- GIVT – general invalid traffic, which is computed as the number of visitors that failed to produce a fingerprint. As mentioned above, these are often dumb bots with limited JavaScript support. Another common reason is network latency, especially evident in traffic with slow connection rates when visitors manage to close the tab or window before their fingerprint is submitted. Currently, this column also accounts all clicks received when a stream was in All money, All white, or On review mode with disabled fingerprint collection because fingerprint scanning is not performed in these modes. We plan to change this logic in future to make this column reflect real GIVT more precisely and transparently.
- SIVT – sophisticated invalid traffic, that is, the number of fingerprints that Adspect consciously filtered out as bad traffic. This may serve as a rough traffic quality metric with respect to the more advanced types of click fraud that get more spread today. This metric also includes visitors blocked by manual stream filters (country, OS, browser, regular expressions, IP address blacklist.)
- Cost – total traffic cost computed as a sum of costs of each click, if passed via URL parameter.
- Bots cost – cost of the traffic that was directed to the white page, which is the precise metric of your budget loss.
- Quality – percentage of money hits in the whole click volume. This is the the best metric for evaluating traffic quality as a whole and may be used to compare different traffic sources, publishers, ad spots, etc. Especially useful with grouping by sub ID for compiling blacklists of bot-ridden zones, as described in a dedicated paragraph.
- Conversions – total number of conversions as accounted via the postback mechanism.
- CR – conversion rate computed as conversions / clicks.
- Revenue – gross revenue of the funnel as accounted via the postback mechanism.
- Profit – net profit computed as revenue cost.
- ROI – return of investment computed as profit / cost.
- CPC / CPM – cost per click computed as cost / clicks, and cost per mille computed as CPC × 1000.
- CPA – cost per action computed as cost / conversions.
- EPL – earn per lead computed as revenue / conversions.
- eCPM – effective cost per mille computed as revenue / clicks × 1000.

8.3 Raw Reports

Raw reports are per-click, that is, they contain information on every click that was processed by Adspect. They are available for download in the CSV format. You may find the Download .CSV button in the bottom left corner of each report table.

Report will be limited to the selected date range and filters. Downloaded CSV files may then be imported into Microsoft Excel or similar spreadsheet software.

Please do not select too broad date ranges as it will lead to the formation of huge CSV files and additional strain on our servers. We limit the total number of rows that will be included in report, and this limit is subject to change at the sole discretion of our systems administrators.

8.4 Raw Report Columns

Raw reports can have either one or two rows per each click. The first row corresponds to serving of our fingerprint collector script for the client browser to execute. The second row, if present, corresponds to fingerprint scanning and making the decision: allow or block. The second row may be missing if the visitor failed to produce or submit a fingerprint.

Raw reports consist of the following columns:

- `timestamp` – date and time of the event;
- `ip_address` – IP address of the visitor in IPv6 format (IPv4 addresses are represented via standard IPv4-to-IPv6 mapping);
- `stream_id` – ID of the stream that the event happened in;
- `country_code` – ISO 3166-1 alpha-2 country code of the visitor;
- `os` – name and release of the visitor's operating system;
- `browser` – name of the visitor's browser;
- `cost` – cost of the click, if passed via URL parameter;
- `sub_id` – sub ID of the click, if passed via URL parameter;
- `click_id` – unique ID of the click, if passed via URL parameter;
- `mode` – stream mode at the moment of the event;
- `sequence` – click processing stage;
- `target` – target page shown to the visitor: 0 for white page, 1 and above for money pages;
- `tags` – list of mnemonic tags, mostly for internal use, that represent particular filtering reasons.

8.4.1 Tags

The exact nature of click tags is a trade secret–we do not disclose our filtering techniques. However, we do give out information about some of them that can be used as proofs of bot traffic (e.g. for demanding refunds from ad networks) or for debug purposes:

- `REVIEW`, `MONEY`, `WHITE` – decision made by customer via stream mode;
- `IP`, `EX` – IP address blacklisted by us: proxies, VPN and hosting providers, antivirus companies, ad scoring companies, security companies, known moderator origins, etc;
- `BL` – IP address blacklisted by the stream IP/ASN blacklist;
- `GBL` – IP address blacklisted by the global IP/ASN blacklist;
- `WL` – IP address not whitelisted by the stream or global IP/ASN whitelist;
- `LIMIT` – IP address blocked due to exceeding its click limit;
- `BOT` – visitors that identify themselves as bots, including known device emulators and virtualized environments;
- `PARANOID` – visitors blocked by paranoid mode;

- NOGEO – IP address has no officially assigned country code;
- GEO, OS, BROWSER, LANG, TZ, IPTZ – visitors blocked by manual stream filters;
- RULE – visitors blocked by a user-defined URL rule;
- UARE – visitors whose user agent matched customer-supplied regular expression;
- REF – visitors whose referer matched customer-supplied regular expression;
- NOPAGE – no money page specified in stream settings, or all money pages are turned off.

CHAPTER 9

Best Practices

Below you may find a compiled list of general recommendations that we give to all of our customers. We highly encourage you to follow them in order to achieve the best results with Adspect.

WARNING: failure to abide by these recommendations may result in bans and/or dramatic decrease in cloaking quality! Adspect may not be held responsible for any such negative consequences.

9.1 Domains and hosting

1. **Do not use** domain names in cheap zones like `.site`, `.club`, `.world`, etc. as they attract more thorough and frequent scrutiny from antivirus and ad scoring companies, effectively being tainted from the beginning. **Only use** domains in `.org`, `.net`, and `.com` zones, in the order of priority.

   Tip: you may check domain ban status in Facebook using Developer Tools.

2. **Do not use** domain names that contain questionable or stop words: "sex", "xxx", "win", "diet", "health", "meet", "date", and so forth. Be creative and choose domain names that look and sound like brands.

3. **Always use** Cloudflare to hide IP addresses of your servers. Many ad networks flag IP addresses behind domains in banned accounts, however, they will never flag addresses that belong to a huge CDN company serving 10% of the World Wide Web. Adspect fully supports Cloudflare in proxy mode.

4. **Do not use** technical domains of hosting companies. They both raise suspicion and uniquely identify particular servers of the hoster.

5. **Do not use** Namecheap virtual hosting because they employ WAF (web application firewall) that by default blocks POST requests that we rely upon, causing 403 Forbidden errors.

6. **Do not name** any of your money or white page files `index.html` because it is likely to take priority over our `index.php` if the file name is omitted in the URL after `/`, thus exposing your real pages. Always use distinct and hard to guess names for money and white pages.

7. **Make sure** that your white pages do not have broken links, non-loading images, or scripts with errors. Check the Console and Network tabs of developer tools in your browser, it will highlight potential problems red.

8. **Do not use** web resources (images, styles, scripts) from third party domains unless they are well-known CDNs (content delivery networks) like those of jQuery, Bootstrap, Font Awesome, Google Fonts, etc. Better download them and host locally.
9. **Always set up** your pages for HTTPS. Cloudflare provides free SSL/TLS certificates for domains in proxy mode. Let's Encrypt also provides an infrastructure for obtaining free SSL/TLS certificates.
10. **It is recommended** to use hosting providers in proximity to your target audience, ideally in the same country. This is especially important when working with popunder ad format.

9.2 Cloaking

1. **Never reuse** domain names, creatives, or white pages in the same traffic source without modifications. Register new domain names for new accounts and modify your creatives and white pages to be unique.
2. **Always use** the most strict manual filters by country, OS, browser, and languages. Match them to targeting settings of your campaigns.
3. **Always use** zero redirect (file-based) display mechanism for white pages, if possible. This rule is **mandatory in Facebook, Google Ads, TikTok, Bing, and Gemini** (among others) with PHP integration!
4. **Always make sure** that your white pages are convincing and relevant to your ad campaigns (creatives, languages, targeting options.) **Never use** obviously bogus white pages like redirections to Google.
5. **Use self-hosted white pages** instead of site constructors (Shopify, Wix, Tilda, etc.), especially when working with Facebook or Google Ads because both ad networks seem to ban campaigns with site constructors preemptively now due to their widespread use for cloaking purposes.
6. When using site constructors, **check availability** of your site periodically–it is not uncommon for such services to ban sites for suspected use in various cloaking schemes.
7. **Put links** to legitimate-looking terms of use, privacy policy, and cookies policy on your white pages when working with the more strict ad networks like Facebook, Google Ads, Microsoft Advertising, etc. **The EU Cookie Law also requires** websites to obtain informed consent from visitors if using cookies.
8. **Put** `robots.txt` and `sitemap.xml` files into the root of your domain, especially when working with search engine-based ad networks (Google Ads, Microsoft Advertising, Verizon Media Native, etc.)
9. **Do not use** copied landing pages as is, always alter them in one way or another to make them overall unique. This practice helps with disarming signature-based flagging.
10. **Do not use** copyrighted materials on your white pages, they may be detected and rejected.
11. **Do not use** overly simple, single page, skewed/broken, non-mobile-optimized, or low quality white pages. Always remember that a white page must look like a legitimate, useful website with authentic content.
12. **Do not use** landing pages of supposedly "white" offers as white pages–advertising networks tend to have a wildly different idea of "white". **Never use** direct affiliate links as white pages.
13. **Do not alter** white pages of running campaigns in any way. The more strict ad networks may detect even small changes like additions of extra `<script>` tags and initiate campaign or account review.
14. **Use one stream** per ad campaign. This rule lets us detect suspicious visitors better across many streams by statistical means. This also helps us to analyze traffic upon request in case of bans because clicks from different campaigns do not commingle in the same stream. **We will not be able** to inspect your traffic if you mix several campaigns in a single stream!
15. **Always** submit campaigns for review with corresponding streams set to **On Review mode** and with **Blacklist All IP Addresses in Review Mode** setting enabled. **Do not** start campaigns in Filtering mode!

16. **Always** download new PHP files (`index.php`, `filter.php`, or `ajax.php`, depending on integration type) after bans–each time a file is downloaded a new unique JavaScript fingerprinting code is generated for it, ensuring that the code will not be detected as malicious or suspicious.

9.3 Facebook Pixel

If you must use Facebook Pixel to signal conversion events from your money page, then **do not** do so using their usual script as it will expose the URL of your money page in its Referer header. However, there are relatively safe workarounds.

The same techniques may be used to secure pixels of other ad networks.

9.3.1 Disable Referrer Globally

One way to prevent referrer information from leaking is to disable referrer globally. In order to do so, add the following code into the `<head>` tag of the page that contains your Facebook Pixel:

```
<meta name="referrer" content="no-referrer">
```

9.3.2 Custom Pixel

Another way to make your Facebook Pixel safe is to use a custom version of it in order to hide referrer information. Facebook provides a short version of their pixel for visitors with disabled JavaScript, e.g:

```
<img height="1" width="1" style="display:none" src="https://www.facebook.com/tr?
↪id=111111111111111&ev=Lead&noscript=1">
```

Take the pixel URL out of the `src` attribute and use it in either of the two safe variants below, whichever you find more appropriate for your particular use case:

1. Static HTML iframe method, suitable for firing the pixel immediately when the page is loaded:

   ```
   <iframe height="1" width="1" style="display:none" src="https://www.facebook.com/
   ↪tr?id=111111111111111&ev=Lead&noscript=1" referrerpolicy="no-referrer">
   ```

2. Dynamic JavaScript method, suitable for firing the pixel from a script:

   ```
   <script>
   fetch("https://www.facebook.com/tr?id=111111111111111&ev=Lead&noscript=1", {mode:
   ↪"no-cors", referrerPolicy: "no-referrer"});
   </script>
   ```

CHAPTER 10

Tips and Tricks

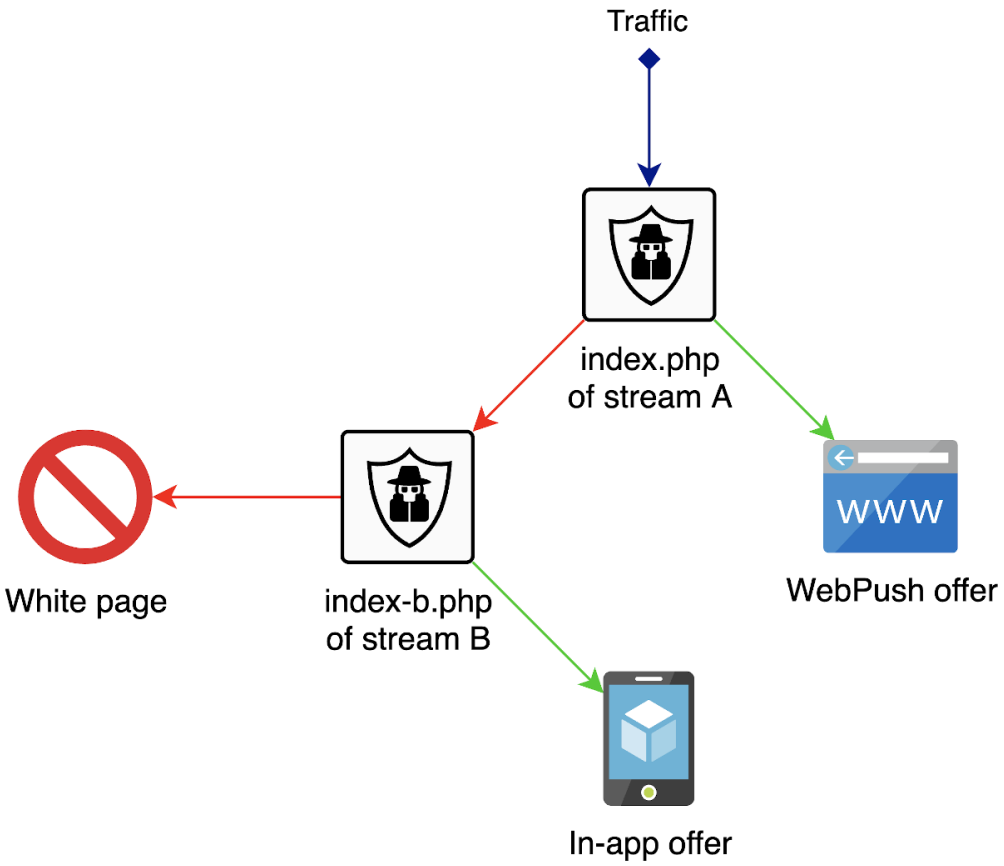
The flexible nature of `index.php` files employed by Adspect combined with file-based page display mechanism (which uses the `require()` language construct of PHP) allows for more sophisticated setups of streams. In this chapter we will describe several advanced scenarios that may be very useful to you.

10.1 Stream Chaining

Since `index.php` is a regular PHP script, it may be used as a money page or a white page of a stream, that is, one stream may redirect visitors to another stream, letting you chain them in different ways. Typical setup of stream chaining is best described by example of a real-world application.

Suppose you have an ad campaign in a source that supplies both browser-based and mobile app-based visitors in a mix without providing any option to split them. Such networks exist–push notifications networks that have both WebPush subscribers and subscribers for in-app or PWA notifications. You would like to split these traffic types and send them to different affiliate offers: `https://example.com/webpush-offer` for WebPush-based visitors and `https://example.com/inapp-offer` for mobile app-based visitors.

You can accomplish this by chaining two streams that have different settings for mobile app-based visitors. The first stream A accepts incoming clicks and filters out app-based ones to its white page. The second stream B is attached to the first one as its white page and filters out bots and moderators, letting legitimate app-based visitors through to the separate offer.



Traffic flow chart

Here are the relevant settings of the entryway stream A:

- Money page: `https://example.com/webpush-offer`
- White page: `index-b.php`
- Allow traffic from mobile apps: *disabled*

The relevant settings of the rear stream B:

- Money page: `https://example.com/inapp-offer`
- White page: `https://google.com/`
- Allow traffic from mobile apps: *enabled*

The mobile apps setting of stream A is what makes it consider app-based visitors as malicious and send them all to stream B along with actual bots and moderators. Stream B will in turn re-analyze the traffic and send good app-based clicks to the in-app offer while still filtering out unwanted visitors to Google.

Finally, you would place `index.php` files of both streams into the same directory: leave the `index.php` file of stream A named as is and rename the file of stream B to `index-b.php`, which will act as a surrogate white page of stream A.

10.2 Dedicated IP Blacklist Stream

Streams have a useful "Blacklist all IP addresses in "Review" mode" setting for collecting IP blacklists of moderators during the "Review" phase of stream lifecycle. This setting may be used to create a dedicated stream for the purpose of collecting a single blacklist of all moderators, bots, etc.

The process is as follows:

1. Create a separate stream that will be used for accumulating IP addresses in its blacklist. Set and leave it in the "Review" mode and enable the "Blacklist all IP addresses in "Review" mode" setting. Effectively, this means that the stream will blacklist IP addresses of every visitor.
2. Use the stream's `index.php` file as a white page for other streams as described in Stream Chaining above. This will direct all bad visitors to the blacklisting stream, making it collect their IP addresses. Alternatively, you may use the stream as a white page only during the "Review" phase of other streams to exclude IP addresses of regular bots from the blacklist as it may possibly lead to false positives (please read this chapter for an explanation of why this may happen.)
3. Watch the IP blacklist of your dedicated stream being collected and copy-paste it into other streams every once in a while (yes, this isn't very convenient, we are working on a solution for blacklist sharing.)

10.3 Combining Cloakers

In case you have access to other cloaking and traffic protection solutions, you may use them together with Adspect to attain potentially higher levels of protection at the expense of additional processing latency. Since most of our competing solutions have a notion of money and white (safe) pages, you should always put Adspect at the rear side of the cloaker chain and create a special setup with two separate streams:

- One stream will be used as a money page of your front-side cloaker, taking the baton from it and inspecting supposedly good visitors that the cloaker has allowed through. The money page of this stream will be your final traffic destination whereas the white page will be the real white page that your intend to cloak with. This stream should be set to the "Filtering" mode.
- Another stream will be used as a white (safe) page of your front-side cloaker, accepting traffic from it in order to collect an IP address blacklist of visitors that the front-side cloaker deems dangerous and additionally train our *VLA* machine learning system on their results. This data lets us absorb their filtering techniques and make Adspect more comprehensive and precise. You should always enable the "Blacklist all IP addresses in "Review" mode" setting as described in the section above in order to populate the IP address blacklist automatically. Set both money and white pages of the stream to the real white page that you intend to cloak with. Leave the stream in "Review" mode.

CHAPTER 11

Drawbacks and Pitfalls

As with every sophisticated system, Adspect has its drawbacks and pitfalls. You should be aware of them. This chapter will educate you how to avoid common mistakes lest you put your affiliate success at stake.

11.1 Do Not Stand Out!

Most ad networks have a routine practice of reviewing all advertising campaigns every once in a while. Apart from helping you pass the initial review after launching a new campaign, the first and foremost task of any cloaking service is to protect your running campaigns from these recurring reviews. Adspect does this very efficiently, proven by many successful campaigns in different ad networks.

But there's a catch: if your advertising activity stands out compared to an average advertiser as perceived by a particular network, it will eventually attract attention of their policy-enforcing team, invoke manual and detailed scrutiny, and will inevitably lead to "piercing the cloak" and suspension. We can guarantee solid protection from routine reviews, but no service can protect you from determined investigators driven by suspicion.

Remember: *do not stand out!* If you slip somehow and put network staff on alert, then you're done. There's no coming back from that one. Here are some common preventive guidelines:

- Do not run ridiculous amounts of traffic from a single account. High volume indicates consistent profitability and thereby raises interest in the nature of your campaigns.
- Keep the number of active campaigns per account low. More campaigns = more things to inspect and uncover.
- Always use a tracker. Continuous affiliate campaigns run without any tracking prompts an observer to wonder how it is possible to sustain them on profitable level.
- Always use tracking parameters with macros supported by the network. This tip is related to the one above and is especially crucial for campaigns with wide targeting settings.
- Use postback feature, if supported by the network.

These considerations outline the principle of division of responsibility: Adspect is responsible for protecting you from regular campaign reviews, and you are responsible for not attracting attention.

11.2 Long Redirect Chains

One notable drawback of cloud-based services like Adspect is that they contribute to overall latency of click processing because of round-trip network lags between your tracker and the service's backend servers. If you observe high technical loss in the "Reporting" section, then it may indicate a network latency problem.

We highly recommend to keep your redirect chains as short as possible. Host your own landing pages with file-based display mechanism instead of redirects to external URLs (detailed here.) Put tracker either before or after our `index.php` file in the traffic flow, but not at both sides. Place your tracker geographically close to your target country or region, if possible.

On a side note, we are developing a self-hosted filtering solution that will allow our clients to cut network latency by performing real-time filtering right at their tracker side, i.e. without having to contact our backend servers on every incoming click. Additional information will be released later.

11.3 False Positives

False positives occur when a filter bars a legitimate visitor, falsely classifying them as malicious. No solution can guarantee 100% precise results, but we are confident enough to state that our rate of false positives is the lowest on the market, sometimes dramatically lower compared to certain competitors. Some solutions on the market yield lower amounts of filtered traffic than Adspect, but that is in fact their fault and not an achievement–they let malicious visitors slip through. Such misclassifications are called false negatives and can be easily proved on click-by-click analysis of decisions. Adspect provides decision logs in the form of raw CSV reports.

11.4 False Negatives

False negatives occur when a filter fails to detect a malicious visitor, letting them through. This is the very reason why affiliate campaign cloaking fails every once in a while. False positives stem from the practical impossibility to detect and filter out every malicious visitor. Any and all measures taken can be circumvented, given enough determination and proper technical expertise. What we described in the Do Not Stand Out! paragraph above is how such determination occurs in the heads of those who you are protecting from; and be sure that they do have all the technical skills needed.

Remember: *false negatives and their consequences result from drawing attention!*

CHAPTER 12

Referral Program

Adspect has a referral program that allows our customers to earn money for bringing new clients to us. The referral program works on a revenue share basis, that is, the referral fee is credited to your balance each time a customer that you brought purchases a subscription.

The referral fee is 10%. However, it may be increased individually if you manage to bring new clients consistently. Please contact us if you have considerable presence on affiliate marketing or SMM forums, blogs, or in Telegram groups, and we will find a mutually profitable agreement.

The referral fee is credited to your account balance. Each time an invoice is created, funds on your balance are used first and deducted from the total amount that remains to be paid. This means that if you have enough funds on your balance, then you may use them to purchase a subscription wholly. For each credited fee, you will receive a referral fee receipt in the Invoices section of the members area.

You may find your referral link in the Ref. Program section, along with a list of your referral clients. Use that link to promote Adspect, and each new member registered by it will be accounted as your referral.

CHAPTER 13

REST API

Adspect provides a REST API that may be used to manage streams programmatically. The API uses JSON data encoding and supports several methods for basic stream operations. It employs HTTP Basic authentication scheme in which API key is supplied as username and password is left blank. You may find your API key in profile.

Each API request must contain two mandatory headers:

1. `Content-Type: application/json` to indicate the use of JSON data encoding;
2. `Authorization: Basic ###` to authorize access, where `###` is `base64(API key + ":")`.

The base URL for all API methods is `https://api.adspect.net/v1/`. Descriptions below specify paths relative to this base URL. Stream IDs used in API methods should be speficied as full UUIDs, e.g. `cbb360ff-5a28-41d0-9ac8-9889a01149fa`.

Currently, only stream management methods have been implemented.

13.1 Stream Fields

Each stream is represented as a JSON object that contains the following properties:

- `stream_id` – full stream ID in UUID format;
- `account_id` – full account ID in UUID format, read-only;
- `name` – stream name, string;
- `mode` – stream mode, string, either `Filter`, `Review`, `Money`, or `White`;
- `money_pages` – array of one or more (up to 254) money page objects, each having the following format:
  - `page` – target URL or page file name, string;
  - `arg_passthru` – whether to perform URL parameters passthru, boolean;
  - `weight` – relative weight for A/B traffic distribution, integer;
  - `enabled` – whether this money page is enabled, boolean;

- `white_page` – white page URL or file name, string;
- `white_arg_passthru` – whether to perform URL parameters passthru to white page, boolean or integer;
- `ml_precision` – VLA precision in percents, integer;
- `cost_parameter` – parameter name for click cost accounting, string;
- `sid_parameter` – sub ID parameter name, string;
- `cid_parameter` – click ID parameter name, string;
- `enable_fp` – JavaScript fingerprints enabled flag, boolean or integer;
- `paranoid` – paranoid mode on flag, boolean or integer;
- `allow_apps` – mobile apps allowed flag, boolean or integer;
- `countries` – array of allowed country strings in ISO 3166-1 alpha-2 format;
- `os` – array of allowed operating system strings:
  - `Android 1`
  - `Android 2`
  - `Android 3`
  - `Android 4`
  - `Android 5`
  - `Android 6`
  - `Android 7`
  - `Android 8`
  - `Android 9`
  - `Android 10`
  - `Android 11`
  - `iOS`
  - `macOS`
  - `Linux`
  - `Other`
  - `Windows XP`
  - `Windows Vista`
  - `Windows 7`
  - `Windows 8`
  - `Windows 8.1`
  - `Windows 10`
  - `Windows Other`
- `browsers` – array of allowed browser strings:
  - `Apple Safari`
  - `Google Chrome`

    - Internet Explorer
    - Microsoft Edge
    - Mozilla Firefox
    - Opera
    - Other
    - Samsung Internet
    - UC Browser
    - WebView
    - Yandex Browser
- engines – array of allowed browser engine strings:
    - Blink
    - EdgeHTML
    - Gecko
    - Other
    - Presto
    - Trident
    - WebKit
- languages – array of allowed browser language codes;
- timezones – array of allowed time zones as integer hour offsets from UTC;
- tz_match_ip – match browser time zone to IP time zone flag, boolean or integer;
- url_rules – array of zero or more (up to 64) URL rule objects, each having the following format:
    - param – URL parameter name, string;
    - op – rule operator, one of:
        * EXISTS – parameter exists;
        * NEXISTS – parameter does not exist;
        * REGEX – value matches regular expression;
        * IREGEX – value matches regular expression (case-sensitive);
        * NREGEX – value does not match regular expression;
        * NIREGEX – value does not match regular expression (case-insensitive);
        * EQ – value equals argument;
        * NEQ – value does not equal argument;
        * GT – value is greater than argument;
        * GE – value is greater than or equal to argument;
        * LT – value is less than argument;
        * LE – value is less than or equal to argument;
        * ASSIGN – assign new value to parameter;

        * RENAME – rename parameter;
        * DELETE – delete parameter;
      - arg – rule argument, string;
      - enabled – rule enabled flag, boolean;
  - ua_regex **(obsolete, will be removed)** – regular expression for the user agent filter, string;
  - referer_regex **(obsolete, will be removed)** – regular expression for the referer filter, string;
  - ip_on_review – blacklist IP addresses in "Review" mode flag, boolean or integer.

Example:


```
{
   "stream_id": "1eacc6d0-875f-6f5c-bff8-00162501c2b4",
   "account_id": "1eaa2ce5-d4dd-63ec-b8a4-00162501c2b4",
   "name": "Example stream",
   "mode": "Filter",
   "money_pages": [
      {
         "page": "https://example.com/offer1?clid={clickid}",
         "arg_passthru": true,
         "weight": 10,
         "enabled": true
      },
      {
         "page": "https://example.com/offer2?clid={clickid}",
         "arg_passthru": true,
         "weight": 20,
         "enabled": true
      }
   ],
   "white_page": "white.html",
   "white_arg_passthru": 0,
   "ml_precision": 95,
   "cost_parameter": "cost",
   "sid_parameter": "sourceid",
   "cid_parameter": "",
   "enable_fp": 1,
   "paranoid": 0,
   "allow_apps": 1,
   "countries": [
      "CA",
      "US"
   ],
   "os": [
      "iOS",
      "macOS"
   ],
   "browsers": [
      "Google Chrome"
   ],
   "engines": [
      "Blink"
   ],
   "languages": [
      "en",
      "fr",
```


(continues on next page)

(continued from previous page)


```
        "es",
    ],
    "timezones": [
        -5,
        -6,
        -7,
    ],
    "tz_match_ip": 1,
    "url_rules": [
        {
            "param": "secretkey",
            "op": "EQ",
            "arg": "4gHzQvF2IoqeQ",
            "enabled": true
        }
    ],
    "ua_regex": "",
    "referer_regex": "",
    "ip_on_review": 1
}
```


13.2 GET /streams

Returns an array of all streams in the account.

13.3 GET /streams/<id>

Returns a specified stream.

13.4 POST /streams

Creates and returns a new stream. Send stream object in JSON format in request body.

13.5 PATCH /streams/<id>

Updates a stream. Send stream object in JSON format in request body.

13.6 DELETE /streams/<id>

Deletes a stream.

13.7 index.php, filter.php, and ajax.php

You may obtain `index.php`, `filter.php`, and `ajax.php` files for any stream via the following requests:

- `index.php / filter.php – GET https://clients.adspect.ai/getindex.php?sid=<id>&mode=<mode>`
- `ajax.php – GET https://clients.adspect.ai/getindex.php?sid=<id>&mode=ajax`

Where `<mode>` is one of:

- `redirect` – redirect to remote URL via HTTP 302 status code;
- `iframe` – display remote URL on your domain inside an `<iframe>` tag;
- `proxy` – display remote URL on your domain by HTTP request proxying.

`index.php` and `filter.php` files are identical.