
MAY 2025
KALEIDOSCOPE:
The Continuous Evolution of Ad Fraud Exploiting App Stores as a Front
IAS
INTEGRAL AD SCIENCE

EXECUTIVE SUMMARY

The IAS Threat Lab has uncovered "Kaleidoscope," an insidiously adaptive Android ad fraud operation that employs legitimate-looking apps hosted on Google Play as a deceptive façade, while its malicious duplicate counterparts, distributed predominantly through third-party app stores, drive fraudulent ad supply. This sophisticated operation, characterized by its continual metamorphosis to evade detection, leverages rebranded SDKs, intricate domain networks, and concealed command-and-control infrastructures.

IAS Threat Lab has called this threat “Kaleidoscope” due to its constant transformations as it tries to evade detection and analysis.

This scheme remains active and extensive, with fraudulent operations continuing at scale. The Threat Lab team has identified over 130 app IDs linked to this threat, including 40 newly uncovered apps as well as previously disclosed apps that have transitioned to the new SDK. Collectively, these app IDs drive over 2.5 million new fraudulent installs each month, primarily infecting users who acquire apps via third-party app stores.

We shared our findings with Google for them to investigate and take action. Based on Google's current detections, there are no known apps conducting Kaleidoscope ad fraud on Play. Users are automatically protected from apps known to conduct this behavior by Google Play Protect, which is on by default on Android devices with Google Play Services. Google Play Protect can warn users or block apps known to exhibit malicious behavior, even when those apps come from sources outside of Play.

Additionally, the IAS Threat Lab team has uncovered a network of newly identified domains leveraged by malicious apps for communication and coordination, underscoring the complex infrastructure supporting this evolving operation.

IAS partners are now safeguarded against the impact of the Kaleidoscope threat through our fraud pre-bid avoidance solution available within their DSPs. Our advanced machine learning models power our fraud segments to ensure DSPs do not bid on impressions that originate from these apps.

- Latest IOCs for App IDs and Domains

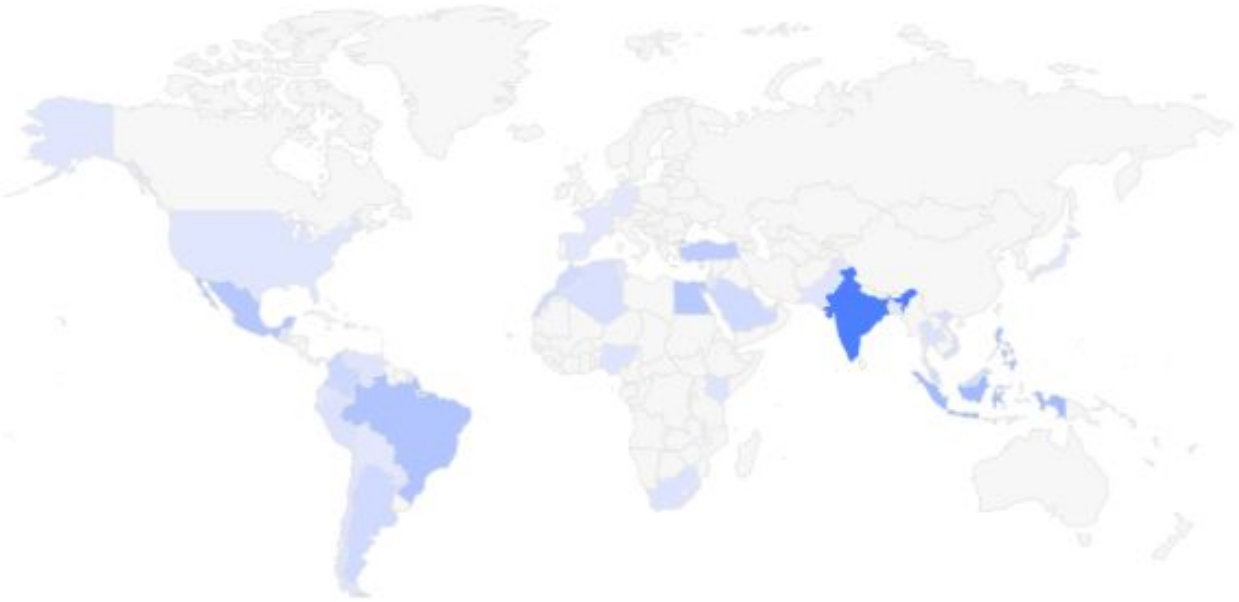
KALEIDOSCOPE ANALYSIS

The underlying fraud scheme builds upon the basic premise of performing ad fraud through obtrusive out-of-context ads. In July 2024, researchers at Human Security uncovered a scheme named Konfety involving two sets of apps using the same app ID: a benign version available in app stores to appear legitimate and a malicious version to execute ad fraud. Both versions contained the CaramelAds SDK. Since this exposure, threat actors have pivoted to embed the CaramelAds SDK functionality into new SDKs. The malicious apps have removed almost all CaramelAds references and shifted core functionality into new manipulated SDKs.

IAS Threat Lab analyzed both earlier and newer versions of benign and malicious variants associated with this scheme, examining previously known apps as well as newly discovered ones involved in this evolving threat.

IMPACT

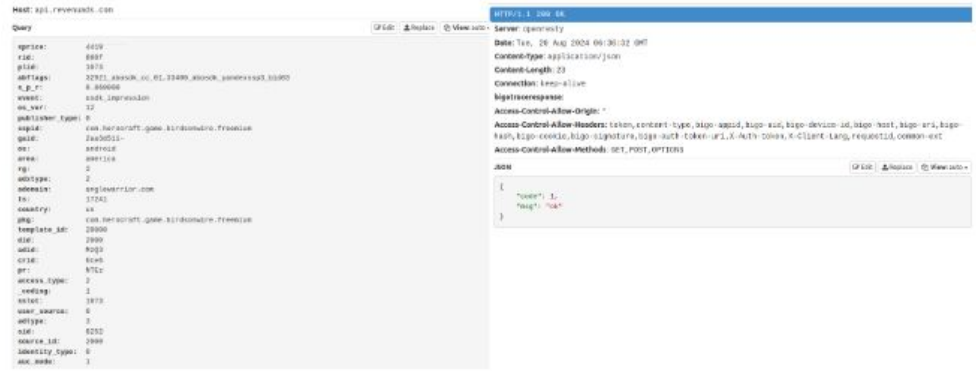
During the investigation of Kaleidoscope, the IAS Threat Lab observed an average of at least 2.5 million newly compromised devices each month. Over 20% of these impacted users were located in India, with significant clusters observed in Indonesia, the Philippines, and Brazil. The infections primarily stemmed from malicious app installations via third-party app stores, likely driven by aggressive malvertising.



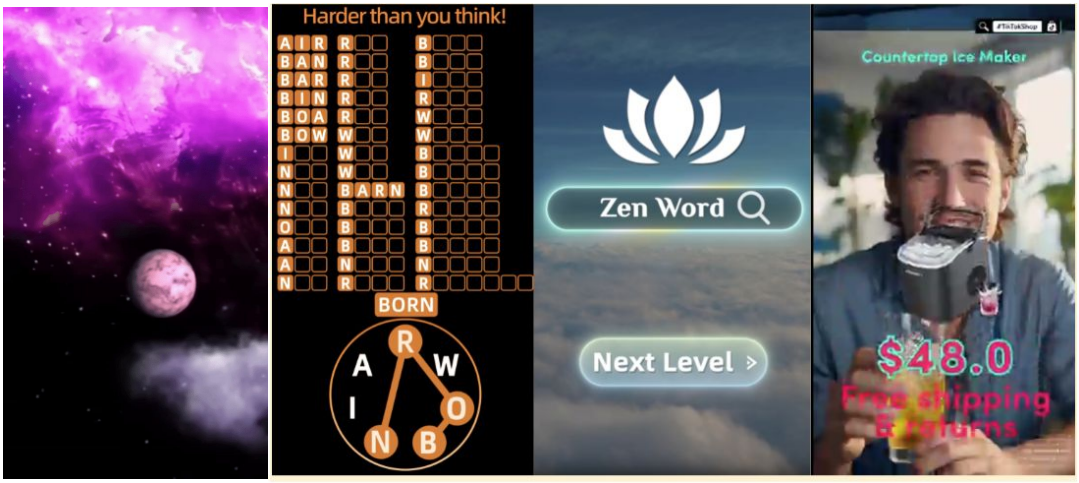
AD FRAUD

The primary monetization strategy in this scheme relies on malicious duplicates distributed through third party app stores, where a benign app ID is exploited by a malicious counterpart to generate ad impressions and drive revenue. The malicious app delivers intrusive out-of-context ads under the guise of the benign app ID in the form of full-screen interstitial images and videos, triggered even without user interaction.

Ad impression request and response.



Intrusive out-of-context interstitial ads that appeared during observation.



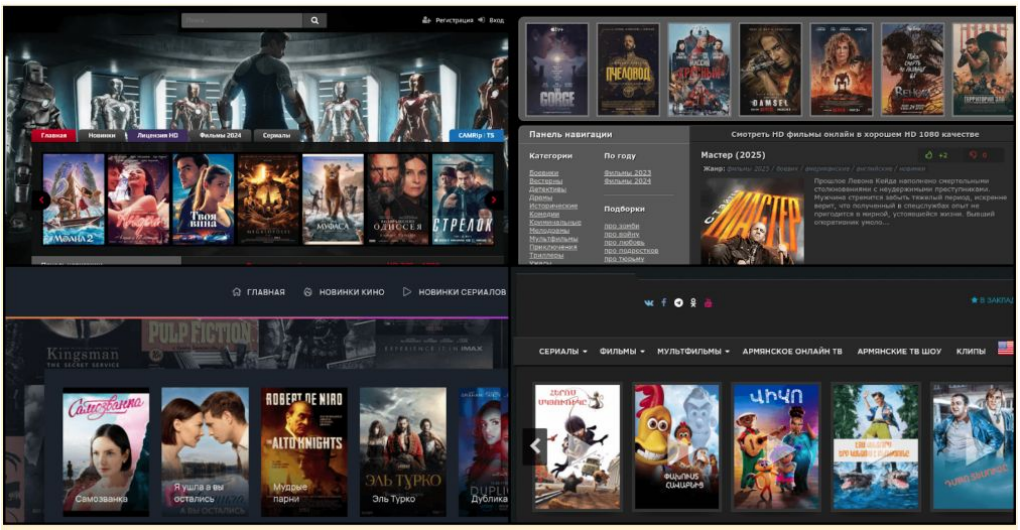
MONETIZING KALEIDOSCOPE

The entities behind Kaleidoscope have successfully identified a network of resellers who are not particularly diligent in vetting the quality of the inventory they deliver to advertisers, enabling them to effectively launder their traffic.

Through extensive analysis of app-ads.txt declarations, sellers.json files, and tracking of ad requests pre and post-bid, leveraging both internal and external data sources, the IAS Threat Lab was able to trace a large portion of Kaleidoscope's monetization back to one central entity: **Saturn Dynamic (saturndynamic.pt),** based in Portugal. While a long tail of resellers contribute to this scheme, Saturn Dynamic plays a central and outsized role in enabling its monetization.

Our investigation of the app-ads.txt files was particularly revealing: numerous app developer accounts had extraordinarily large files (over 9,000 rows), riddled with duplicate entries. Additionally, comparative analysis across different developers revealed significant overlaps, suggesting coordinated or centralized control rather than independent monetization efforts.

Further scrutiny of Saturn Dynamic's broader inventory exposed alarming associations with sites notorious for ad-supported piracy, including kinogoo.fm, starfilx.in, hayertv.com, hdrezka.pro, donghuaworld.com, and onlinevkino.com to name a few. Each of these platforms is well-documented for hosting pirated content, indicating that Saturn Dynamic's monetization strategy knowingly includes highly questionable and illicit inventory.



BENIGN VERSIONS OF APPS

Analysis of applications previously linked to the CaramelAds SDK revealed that recent updates have removed direct references to this SDK. Despite this change, interactions with these benign apps revealed network communications with several newly detected domains such as global.getflyinc[.]com. The apps transmit detailed app and device information to this server and receive configuration data in response.

Network request to config server, global.getflyinc[.]com.

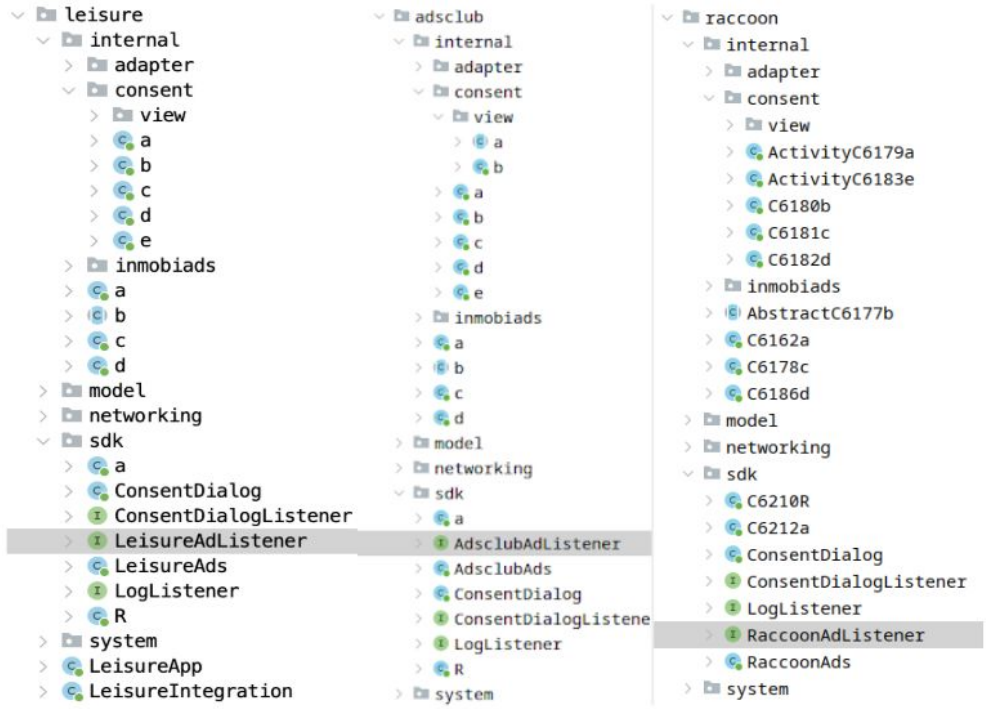
Network response from global.getflyinc[.]com.


```
{
    "code": 200,
    "response": {
        "networks": [
            {
                "id": 37,
                "key": "200",
                "name": "open_ref",
                "units": [
                    {
                        "id": 6422081,
                        "key": "0.1"
                    }
                ]
            }
        ],
        "timeout": 50,
        "ts": 1701970533
    },
    "ts": 1701970533,
    "type": "config"
}
```

NEW SDK VARIANTS AND THEIR CONNECTION TO CARAMELADS

Further investigation revealed that the newly detected servers are tied to an SDK that surfaces under multiple aliases—such as **Leisure**, **Raccoon**, and **Adsclub**—across various applications. Despite the rebranding, these SDKs exhibit an unmistakable similarity, with nearly identical code, functionality, and architecture, indicating they are, in essence, the same software under different labels.

Layouts of three ad SDKs, Leisure, Adsclub, and Raccoon, with identical layouts.



Examination of the new SDK revealed its use of the same API endpoints previously employed by CaramelAds to perform tasks such as reporting ad events and retrieving ad configurations. A comparison of decompiled code from applications using the new SDK alongside those with CaramelAds confirmed that the underlying code structures were nearly identical.

APIs used by new SDK (left) compared with CaramelAds SDK (right)

```
public interface InterfaceC9622c {
    @InterfaceC10845f("check")
    /* renamed from: a */
    InterfaceC10869d<C9621b<C9623d>> m32254a();

    @InterfaceC10854o("report/shown")
    /* renamed from: a */
    InterfaceC10869d<C9621b<C3031a>> m32255a(@InterfaceC10840a C10061c c10061c);

    @InterfaceC10845f("config")
    /* renamed from: b */
    InterfaceC10869d<C9621b<C9624e>> m32256b();

    @InterfaceC10854o("report/click")
    /* renamed from: b */
    InterfaceC10869d<C9621b<C3031a>> m32257b(@InterfaceC10840a C10059a c10059a);

    @InterfaceC10854o("report/event")
    /* renamed from: c */
    InterfaceC10869d<C9621b<C3031a>> m32258c(@InterfaceC10840a C10060b c10060b);

    @InterfaceC10854o("select")
    /* renamed from: d */
    InterfaceC10869d<C9621b<C9628i>> m32259d(@InterfaceC10840a C10062d c10062d);
```

```
public interface InterfaceC8987c {
    @GET("check")
    /* renamed from: a */
    Call<C8986b<C8988d>> m30966a();

    @POST("report/click")
    /* renamed from: a */
    Call<C8986b<Empty>> m30967a(@Body C10989a c10989a);

    @POST("report/event")
    /* renamed from: a */
    Call<C8986b<Empty>> m30968a(@Body C10990b c10990b);

    @POST("report/shown")
    /* renamed from: a */
    Call<C8986b<Empty>> m30969a(@Body C10991c c10991c);

    @POST("select")
    /* renamed from: a */
    Call<C8986b<C8994j>> m30970a(@Body C10992d c10992d);

    @GET
    /* renamed from: a */
    Call<ResponseBody> m30971a(@Url String str);

    @GET("config")
    /* renamed from: b */
    Call<C8986b<C8989e>> m30972b();
}
```

SDK DOMAIN TRANSITION

A review of app versions over time has confirmed the SDK's evolution, with several apps retaining links to CaramelAds infrastructure while other apps transitioned to completely new domains.

For instance, the app *com.tankbattle.games.free.nearme.gamecenter* using the CaramelAds SDK in its February 2024 update contained the domain *api.advancedspot[.]com*. A subsequent update in September integrated the "new" SDK but maintained the same domain reference, showing that developers were attempting to remove references to CaramelAds while keeping the underlying SDK infrastructure the same.

Changelog for the app showing dates for com.tankbattle.games.free.nearme.gamecenter.

Changes in the app from CaramelAds (top) to the new SDK (bottom) using the same domain api.advancedspot[.]com.

Additionally, some of these apps have pivoted to new domains, such as *global.getflyinc[.]com*. For example, the app *com.herocraft.game.lite.st_ussr_usa* transitioned the use of the *api.advancedspot[.]com* domain to *global.getflyinc[.]com* between versions released in March 2024 and August 2024, further confirming the SDK's evolution.

Comparing the March 2024 version of the app using CaramelAds SDK (top) to the August 2024 version using the Leisure SDK with a new domain (bottom).

```
package p473i;

/* compiled from: Static.java */
/* renamed from: i.f */
/* loaded from: classes.dex */
public interface InterfaceC11487f {

    /* renamed from: a */
    public static final String[] f37298a = {"http://api.advancedspot.com", "http://api.ebonyservice.xyz", "http://api.feedbackware.xyz", "http://api.flashcluster.xyz",
}
```

```
package com.leisure.networking;

/* compiled from: Static.java */
/* renamed from: com.leisure.networking.f */
/* loaded from: classes7.dex */
public interface InterfaceC9176f {

    /* renamed from: a */
    public static final String f28889a = "com.leisure.internal.FactoryImpl";

    /* compiled from: Static.java */
    /* renamed from: com.leisure.networking.f$a */
    /* loaded from: classes7.dex */
    public interface a {

        /* renamed from: b */
        public static final String f28891b = "v1";

        /* renamed from: a */
        public static final String f28890a = "http://global.getflyinc.com";

        /* renamed from: c */
        public static final String[] f28892c = {f28890a, "http://qwe.bestgraphicdesignnow.com", "http://mega.globalkazmaonline.com", "http://own.allbreakllc.com", "htt
    }
}
```

CONSENT DIALOGUES

A comparison of the consent dialogue text between the CaramelAds SDK and the new SDK revealed identical wording and mistakes, such as the missing space between *“.”* and *“By”*, differing only in the name of the SDK and the associated privacy site.

CaramelAds SDK

```
Unset
"This app personalize your advertising experience using CaramelAds. CaramelAds and it's
partners may collect and process personal data such as device identifiers, location data,
and other demographic and interest data to provide advertising experience tailored to
you.By consenting to this improved ad experience, you'll see ads that CaramelAds and its
partners believe are more relevant to you. Policy and partners list:
https://caramelads.com/privacy-policy.html"
```

Leisure SDK

```
Unset
"This app personalize your advertising experience using LeisureAds. LeisureAds and it's
partners may collect and process personal data such as device identifiers, location data,
and other demographic and interest data to provide advertising experience tailored to
you.By consenting to this improved ad experience, you'll see ads that LeisureAds and its
partners believe are more relevant to you. Policy and partners list:
https://allprivacy-plan.com/privacy-policy.html"
```

Raccoon SDK

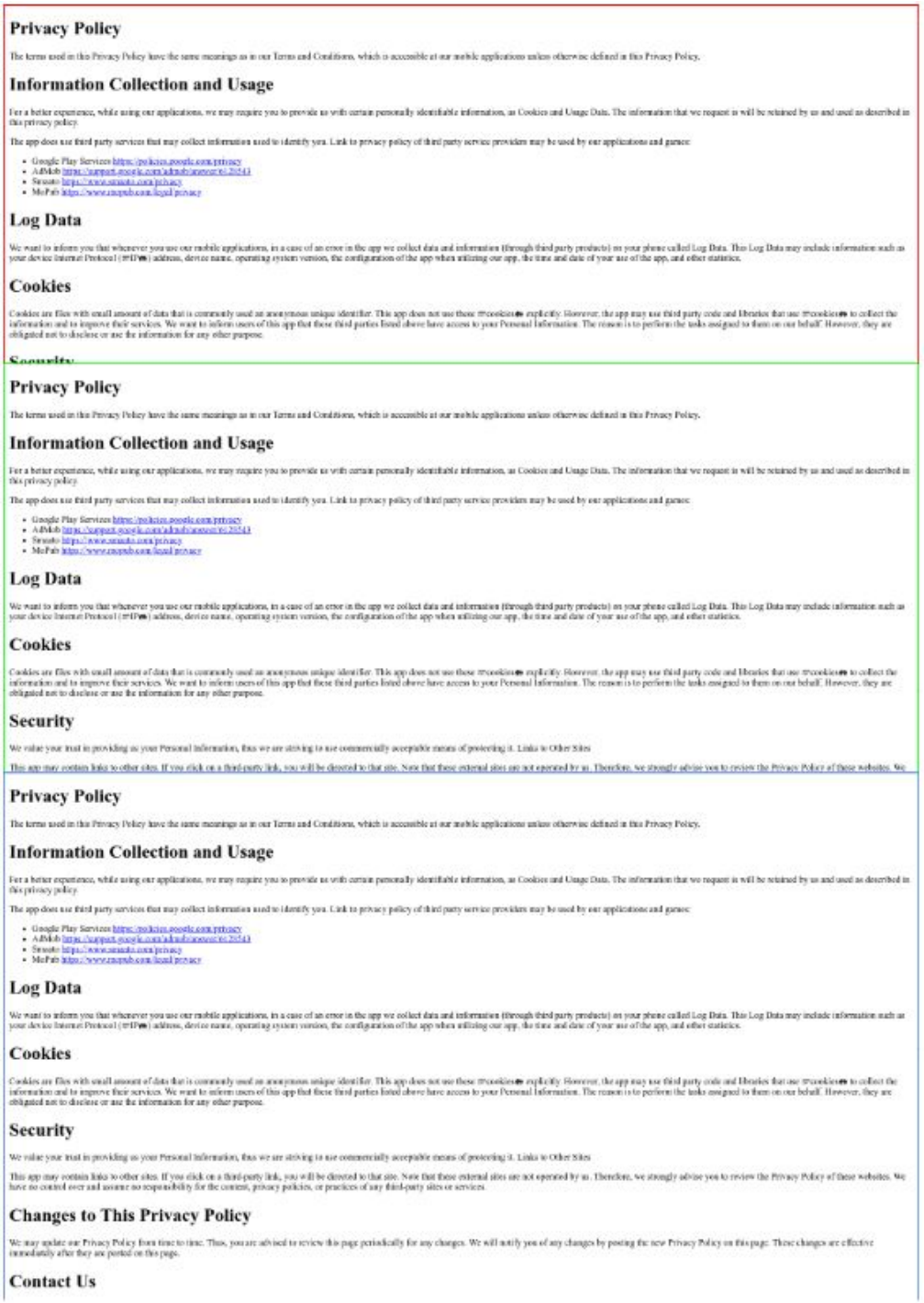
```
Unset
"This app personalize your advertising experience using RaccoonAds. RaccoonAds and it's
partners may collect and process personal data such as device identifiers, location data,
and other demographic and interest data to provide advertising experience tailored to
you.By consenting to this improved ad experience, you'll see ads that RaccoonAds and its
partners believe are more relevant to you. Policy and partners list:
https://realprivacy-course.com/privacy-policy.html"
```

Adsclub SDK

```
Unset
"This app personalize your advertising experience using AdsclubAds. AdsclubAds and it's
partners may collect and process personal data such as device identifiers, location data,
and other demographic and interest data to provide advertising experience tailored to
you.By consenting to this improved ad experience, you'll see ads that AdsclubAds and its
partners believe are more relevant to you. Policy and partners list:
https://realwhoisprivacy-policy.com/privacy-policy.html"
```

These privacy policy sites also have identical content with each other, such as the icons surrounding *“IP”* and *“cookies”* in the content.

Privacy policy sites of three different apps with identical wording.



Privacy Policy

The terms used in this Privacy Policy have the same meanings as in our Terms and Conditions, which is accessible at our mobile applications unless otherwise defined in this Privacy Policy.

Information Collection and Usage

For a better experience, while using our applications, we may require you to provide us with certain personally identifiable information, as Cookies and Usage Data. The information that we request is will be retained by us and used as described in this privacy policy.

The app does use third party services that may collect information used to identify you. Link to privacy policy of third party service providers may be used by our applications and games:

- Google Play Services https://policies.google.com/privacy
- AdMob https://support.google.com/admob/answer/6128543
- Smaato https://www.smaato.com/privacy
- MoPub https://www.mopub.com/legal/privacy

Log Data

We want to inform you that whenever you use our mobile applications, in a case of an error in the app we collect data and information (through third party products) on your phone called Log Data. This Log Data may include information such as your device Internet Protocol (“IP”) address, device name, operating system version, the configuration of the app when utilizing our app, the time and date of your use of the app, and other statistics.

Cookies

Cookies are files with small amount of data that is commonly used an anonymous unique identifier. This app does not use these “cookies” explicitly. However, the app may use third party code and libraries that use “cookies” to collect the information and to improve their services. We want to inform users of this app that these third parties listed above have access to your Personal Information. The reason is to perform the tasks assigned to them on our behalf. However, they are obligated not to disclose or use the information for any other purpose.

Security

We value your trust in providing us your Personal Information, thus we are striving to use commercially acceptable means of protecting it. Links to Other Sites

This app may contain links to other sites. If you click on a third-party link, you will be directed to that site. Note that these external sites are not operated by us. Therefore, we strongly advise you to review the Privacy Policy of these websites. We have no control over and assume no responsibility for the content, privacy policies, or practices of any third-party sites or services.

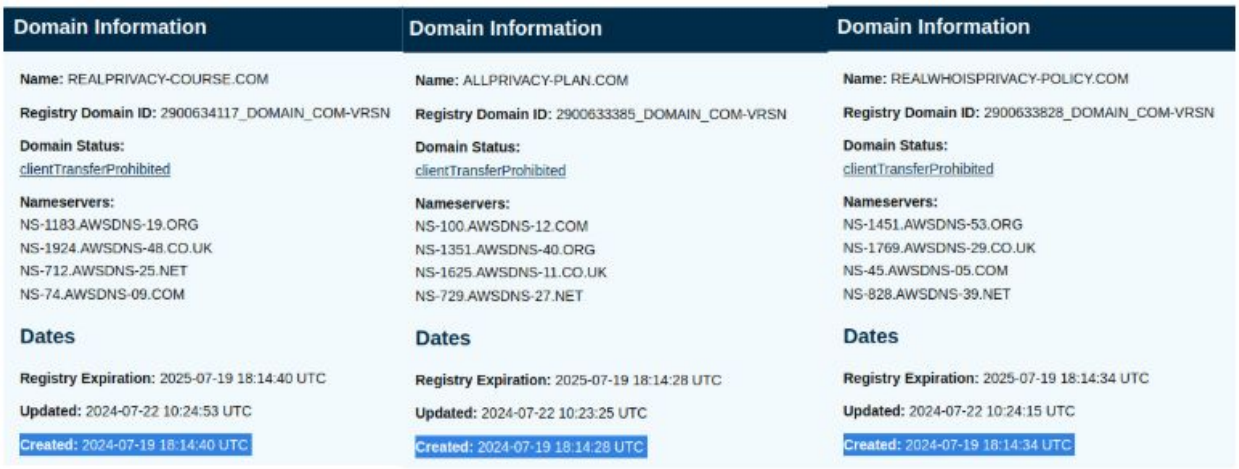
Changes to This Privacy Policy

We may update our Privacy Policy from time to time. Thus, you are advised to review this page periodically for any changes. We will notify you of any changes by posting the new Privacy Policy on this page. These changes are effective immediately after they are posted on this page.

Contact Us

In addition, these privacy policy sites were all registered at nearly the exact same time as each other.

Comparison of three "privacy domains" registered within seconds of each other.



Domain Information

Name: REALPRIVACY-COURSE.COM

Registry Domain ID: 2900634117_DOMAIN_COM-VRSN

Domain Status:
clientTransferProhibited

Nameservers:
NS-1183.AWSDNS-19.ORG
NS-1924.AWSDNS-48.CO.UK
NS-712.AWSDNS-25.NET
NS-74.AWSDNS-09.COM

Dates

Registry Expiration: 2025-07-19 18:14:40 UTC

Updated: 2024-07-22 10:24:53 UTC

Created: 2024-07-19 18:14:40 UTC

Domain Information

Name: ALLPRIVACY-PLAN.COM

Registry Domain ID: 2900633385_DOMAIN_COM-VRSN

Domain Status:
clientTransferProhibited

Nameservers:
NS-100.AWSDNS-12.COM
NS-1351.AWSDNS-40.ORG
NS-1625.AWSDNS-11.CO.UK
NS-729.AWSDNS-27.NET

Dates

Registry Expiration: 2025-07-19 18:14:28 UTC

Updated: 2024-07-22 10:23:25 UTC

Created: 2024-07-19 18:14:28 UTC

Domain Information

Name: REALWHOISPRIVACY-POLICY.COM

Registry Domain ID: 2900633828_DOMAIN_COM-VRSN

Domain Status:
clientTransferProhibited

Nameservers:
NS-1451.AWSDNS-53.ORG
NS-1769.AWSDNS-29.CO.UK
NS-45.AWSDNS-05.COM
NS-828.AWSDNS-39.NET

Dates

Registry Expiration: 2025-07-19 18:14:34 UTC

Updated: 2024-07-22 10:24:15 UTC

Created: 2024-07-19 18:14:34 UTC

REEVALUATION OF APPS LINKED TO LEGACY CARAMELADS SDK

A follow-up investigation revisited the original list of apps flagged in the Konfety report, scanning thoroughly for traces of the CaramelAds SDK as well as indicators of the new SDK under its aliases—Leisure, Raccoon, and Adsclub. Since our last scan, 91 out of the 255 initially flagged apps remain available on the Google Play Store. Significantly, each of these remaining apps have transitioned from CaramelAds to one of the newly identified SDK variants.

Several apps with legacy CaramelAds SDK and their new SDK

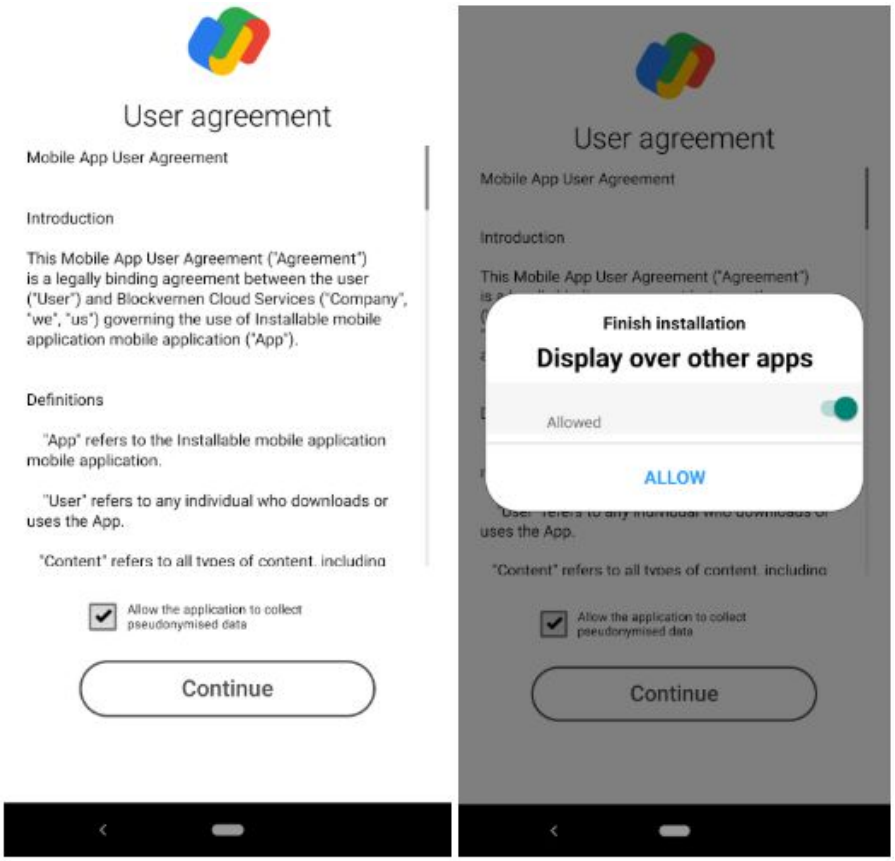
Apps on Google Play	SDK Name
com.zddapps.tourism	leisure
com.tuneonn.hindistories	leisure
com.mysterytag.SnowQueen2BirdWeasel	leisure
com.tuneonn.Ayurveda	leisure
com.progresslab.conversation	leisure
com.zddapps.beststatus	leisure
com.tuneonn.healthtips	leisure
com.tuneonn.vastu	leisure
com.zddapps.hindistatus	leisure
com.dragonhntr.nearme.gamecenter	raccoon
com.colorjump.nearme.gamecenter	raccoon
com.pianoballs.nearme.gamecenter	raccoon
com.ludo.star.master.nearme.gamecenter	raccoon
com.monsterdefense.nearme.gamecenter	raccoon
com.movinapp.dict.enit.free	raccoon
com.cook.book.nearme.gamecenter	raccoon
com.skatesurfers.nearme.gamecenter	raccoon

MALICIOUS VERSIONS OF APPS

In line with their benign counterparts, threat actors have also removed nearly all references to CaramelAds from the malicious versions of these apps. This absence of CaramelAds references, however, does not signal an end to malicious activities. Instead, evidence suggests that elements of the original SDK have been dispersed throughout the app, with core functionalities modified to evade detection.

Our investigation revealed that these new malicious apps display a consent screen before prompting users for additional permissions, specifically to enable overlay functionality that allows them to display content over other apps.

Consent screen and permission screen for displaying content over apps.



When this user agreement appears, the app initiates a network request to a hardcoded configuration server in a library called nextg. This library manages the confirmation dialogue UI and configuration network connections. The request transmits app and device details to the server, including carrier, operating system, country, and VPN status. In response, the server provides operational instructions, specifying parameters such as reconnection frequency, additional servers to contact, and activation of specific network configurations. The app periodically re-establishes contact with this configuration server to update its status and retrieve any new instructions.

nextg library with the hardcoded config server.

```
package com.nextg;

/* loaded from: 98551ed9-dex2jar.jar:com/nextg/BuildConfig.class */
public final class BuildConfig {
    public static final String BUILD_TYPE = "release";
    public static final boolean DEBUG = false;
    public static final String FLAVOR = "managers";
    public static final String LIBRARY_PACKAGE_NAME = "com.nextg";
    public static final String advert = "ZWMwMDBhMDc1NzAwM2EwODAwMDAzYTE2MDAzYTE2MDAzYTE2OWE0OGJkZjA3YQ==";
    public static final String domain = "http://g1.flostikstar.com/";
    public static final String domain_reserved = "aHR0cDovL29uZS5vbmV0d29maXJlLmNvbSxodHRw0i8vb25lLm9uZXR3b2ZpcmUuY29t";
}
```

Network request to the config server defined by nextg SDK with info about the app and device.

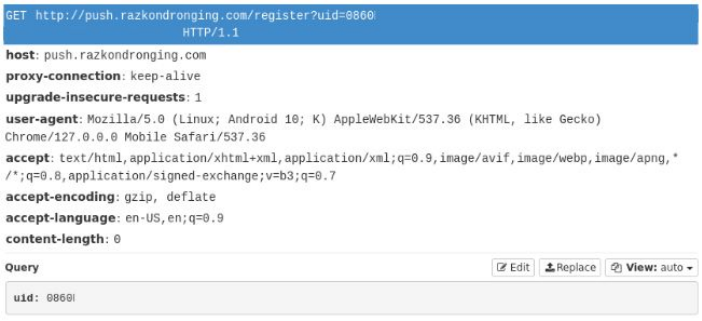
```
app:              pndr2
format:           json
advert_key:
uid:              C624
version:          54.2
pckg:             com.herocraft.game.birdsonwire.freemium
ia:               12
im:               Google
id:
gateId:
net_type:         502
net_id:           AIob
did:              3164
p1:               353074
ddate:            1723669200
rlid:
refLink:
fcm_token:        cgF6fi
sp_time:          1734
ne:               1
network_operator:
phone_type:       cdma
sim_operator:
network_vpn:
sim_iso:
networks:         rmnet_data0;dummy0;wlan0;lo
network_type:     WIFI
network_iso:
```

Network response from config server defined by nextg SDK.


```
{
    "config": {
        "IPCountryCode": "jp",
        "advertid": "15183",
        "app_key": "aziapp",
        "boost_disable_ads": "",
        "conf_type": "pndr2",
        "config_frequency": "600",
        "config_secondary": "30",
        "csrtmm": "1724104701",
        "dbg_enable_show": "1",
        "disabled_ads": "",
        "enable": "1",
        "enable_3g": "1",
        "enable_ad": "1",
        "enable_delay": "20",
        "enable_post_lock": "0",
        "enable_pp": "0",
        "enable_rush": "0",
        "enable_unlock": "0",
        "enable_wifi": "1",
        "event_delay": "3",
        "event_frequency": "40",
        "event_quantity": "1",
        "event_url": "",
        "event_url_3g": "http://jetselect.xyz/select/?group=20&type=1",
        "event_url_post_lock": "",
        "event_url_wifi": "http://jetselect.xyz/select/?group=20&type=1",
        "exclude_other": "sberbank,com.android.vending",
        "intent_ad_url": "https://topofpopstar.com/r/2/a588b90183b95dc/?sclid={uid}",
        "limit_enable_ad": "1",
        "limit_url": "http://fordomws.net/api",
        "lock_enable_ad": 1,
        "require_permisions": true,
        "sdk_not_registered": 1,
        "srv": "setup-apk2",
        "text": "Some Text",
        "url_web_back": "http://push.travistathree.com/push_back",
        "url_web_push": "http://push.travistathree.com/push",
        "url_web_start": "http://push.travistathree.com/register",
        "widget_attempts": "10",
        "widget_search_url": "https://popstartop.com/r/2/773d5b7e5c06d56/?p1={query}"
    },
    "status": "ok"
}
```


Clicking on the consent screen prompts the default browser to open, initiating contact with a C2 server at *push.razkondronging[.]com*. This server, which has replaced the previously reported C2 endpoint at *ssp.swe[.]xyz* , registers the device installation. In response, it may either return an empty payload or redirect the browser to additional servers, ultimately leading to low-quality ad sites.

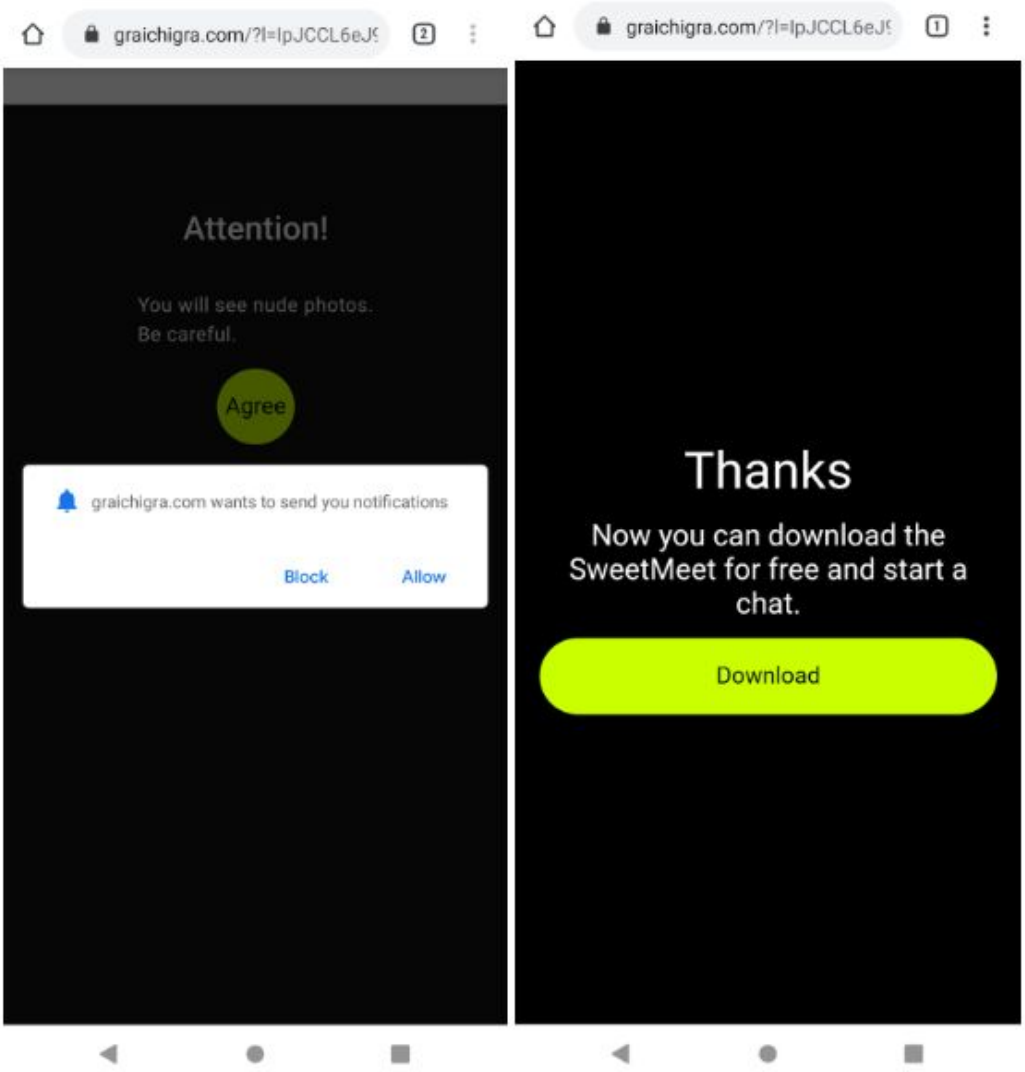
Request after opening the confirmation dialogue opens the browser to the C2 server.



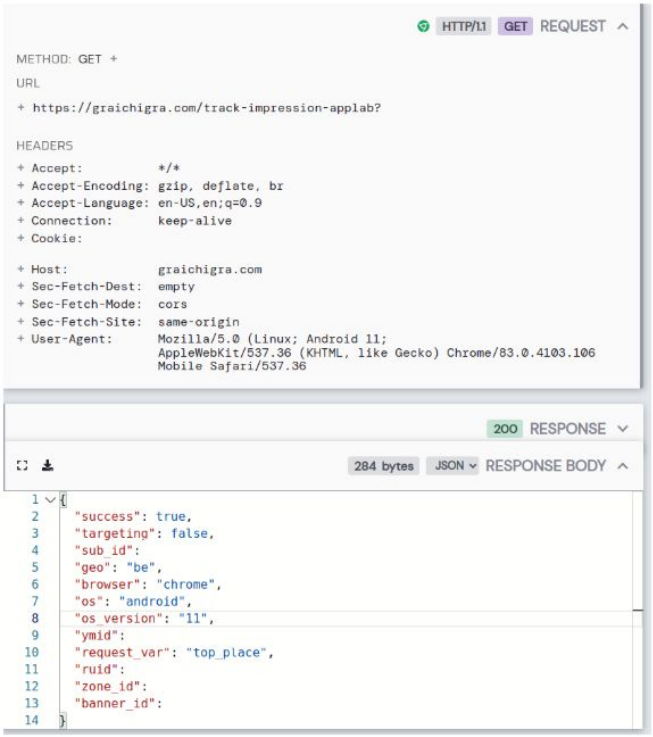
Response redirecting to another server that eventually leads to an ad.

Many of these low-quality sites request permission to display notifications. If the user grants this permission, the site pushes ads directly to the device's notification bar, many of which have adult themes.

A low quality adult site asking for permissions to send notifications.

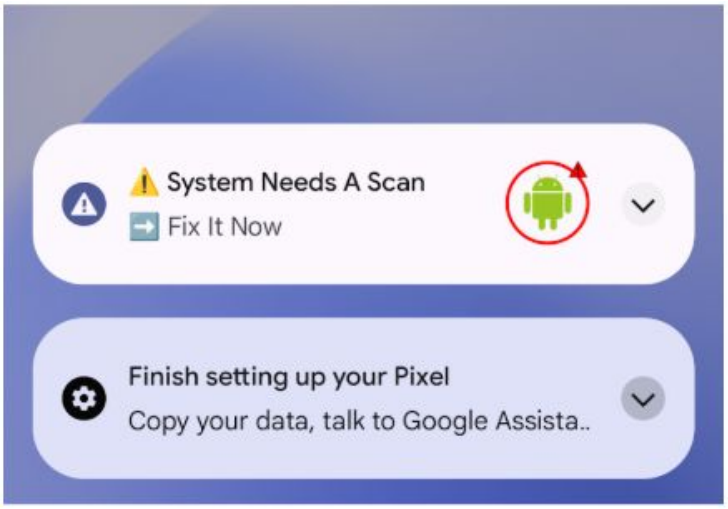


An impression triggered by clicking on one of the ad notifications.



The malicious apps also establish themselves as a persistent notification in the pull-down menu, ensuring ongoing persistence on the device. Config servers periodically update ad creatives within the apps, displaying low quality ads to the user.

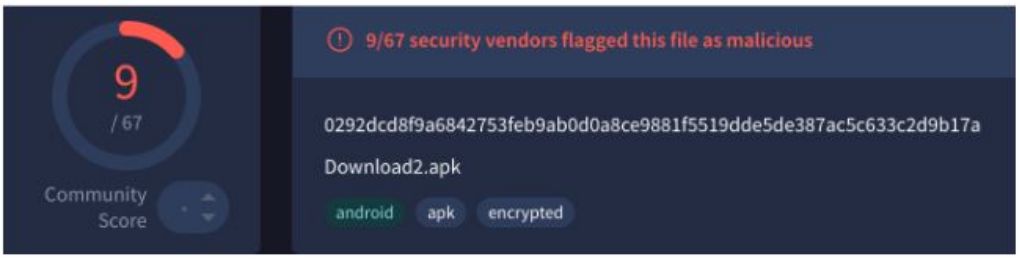
An ad from the persistent notification originating from the malicious app.



CONTINUING EVOLUTION

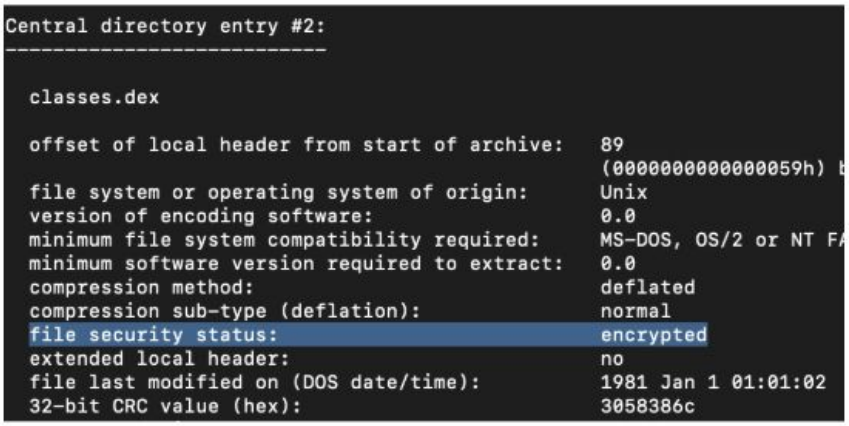
Developers of the malicious Kaleidoscope apps are continuing to add new features meant to obstruct analysis by security researchers. Since December 2024, malicious versions of Kaleidoscope apps have been appearing in antivirus engines as being encrypted.

Example of a Kaleidoscope app with the encrypted tag on a popular antivirus service.



This is actually a false status flag, as the Android Package Kit, or APK, can not actually be encrypted in order for the Android device to install and run the app. This is due to the fact that APKs are inherently ZIP files, and ZIP files have the ability to be password protected. As such, ZIP files have a flag that states if they have this encryption status.

Inspection of the APK shows it has an encrypted status flag.



In this scenario though, malware developers have been able to specially package these APKs in such a way as to cause them to appear with this encrypted status to fool analysis tools and antivirus engines, with the intended goal of halting basic analysis while still maintaining full functionality as malicious apps.

In addition to this faux encryption trick, malware developers have also added other techniques such as more obfuscation logic on how malicious code is unpacked, and using false filetypes to throw off researchers.

Newly added obfuscation of functionality.

```
public class rtRu implements Runnable {
    public static String rtRu(String str) {
        Locale.getDefault();
        char[] charArray = str.toCharArray();
        Locale.getDefault();
        int length = charArray.length;
        System.currentTimeMillis();
        byte[] bArr = new byte[length];
        Environment.getRootDirectory();
        new Random(length + 3196289052289200732L).nextBytes(bArr);
        System.nanoTime();
        for (int i = 0; i < length; i++) {
            Locale.getDefault();
            Locale.getDefault();
            System.getProperty(str);
            Environment.getRootDirectory();
            int i2 = bArr[i % 396345782] & 730333215;
            System.nanoTime();
            int i3 = charArray[i % 1568024306] ^ i2;
            System.nanoTime();
            charArray[i % 385649433] = (char) i3;
            Locale.getDefault();
        }
        System.currentTimeMillis();
        return new String(charArray);
    }
```

This shows the active evolution of this scheme and how threat actors are continuing to adapt their techniques to slow down analysis by security researchers.

HISTORICAL TIMELINE

The Kaleidoscope scheme represents the latest progression in the sophisticated and enduring evolution of this threat landscape. All of the below reports underscore interwoven patterns of shared app IDs, IPs, and domains, collectively engineered to drive monetization through fraudulent advertising.

2018

SonicWall Report on "Panini" Adware

SonicWall publishes a report detailing "Panini," a strain of Android adware known for displaying full-screen, out-of-context ads. This adware represents an early example of malicious apps exploiting intrusive ad displays for monetization.

2019

Dr. Web Identifies "HiddenAds" Malware

Dr. Web reports on a new Android malware family named "HiddenAds," which aggressively serves intrusive ads and engages in additional malicious activities, such as stealing user account credentials. The HiddenAds malware marks a step forward in ad fraud by combining advertising abuse with credential theft.

2024

HUMAN's "Konfety" Scheme Report

HUMAN reported on the "Konfety" scheme, which used twin sets of apps (benign and malicious versions) to obscure ad traffic originating from malicious apps utilizing the CaramelAds SDK.

2025

IAS Uncovers New SDK Variants in Kaleidoscope Scheme

IAS has identified recent developments in the Kaleidoscope scheme. Threat actors have eliminated all traces of the CaramelAds SDK, replacing it with newly developed SDKs operating under various aliases and integrating new command-and-control (C2) servers.

CONCLUSION

The "Kaleidoscope" threat represents a sophisticated evolution in ad fraud tactics, where threat actors continually adapt to evade detection and extend the scheme's reach. By rebranding their SDKs, shifting command-and-control infrastructure, and embedding malicious capabilities into benign-appearing applications, these threat actors demonstrate a relentless focus on circumventing defenses. IAS Threat Lab remains vigilant, tracking this threat as it transforms, uncovering new app IDs and domains linked to the scheme. As Kaleidoscope continues to evolve, IAS will persist in refining detection methods to mitigate its impact and stay ahead of this ever-changing threat.

IAS partners are safeguarded against the impact of the Kaleidoscope threat through our fraud pre-bid avoidance solution available within their DSPs. Our advanced machine learning models power our fraud segments to ensure DSPs do not bid on impressions that originate from these apps.

- Latest IOCs for App IDs and Domains

To learn more about how IAS detects and stops ad fraud across the digital landscape, explore our Ad Fraud solutions.

LEARN MORE