

1 Ann Marie Mortimer (State Bar No. 169077)

2 amortimer@HuntonAK.com

3 Jason J. Kim (State Bar No. 221476)

4 kimj@HuntonAK.com

5 Jeff R. R. Nelson (State Bar No. 301546)

6 jnelson@HuntonAK.com

7 **HUNTON ANDREWS KURTH LLP**

8 550 South Hope Street, Suite 2000

9 Los Angeles, California 90071-2627

10 Telephone: (213) 532-2000

11 Facsimile: (213) 532-2020

12 Attorneys for Plaintiff

13 FACEBOOK, INC.

14 **UNITED STATES DISTRICT COURT**  
15 **NORTHERN DISTRICT OF CALIFORNIA**  
16 **SAN FRANCISCO DIVISION**

17 FACEBOOK, INC., a Delaware  
18 corporation,

19 Plaintiff,

20 v.

21 ONEAUDIENCE LLC,

22 Defendant.

23 CASE NO.: 3:20-cv-01461

24 **COMPLAINT; DEMAND FOR**  
25 **JURY TRIAL**

Hunton Andrews Kurth LLP  
550 South Hope Street, Suite 2000  
Los Angeles, California 90071-2627

## INTRODUCTION

1  
2 1. Beginning no later than September 2019, Defendant OneAudience LLC  
3 (“OneAudience”) controlled a software development kit (“SDK”) designed to  
4 improperly obtain user data from Facebook, Google, and Twitter (“the malicious  
5 SDK”). OneAudience promoted the malicious SDK to third-party application (“app”)   
6 developers, who – in exchange for payment from OneAudience – bundled the malicious  
7 SDK with other software components within their apps. These apps were distributed  
8 online to app users on various app stores, including the Google Play Store, and included  
9 shopping, gaming, and utility-type apps. After a user installed one of these apps on  
10 their device, the malicious SDK enabled OneAudience to collect information about the  
11 user from their device and their Facebook, Google, or Twitter accounts, in instances  
12 where the user logged into the app using those accounts. With respect to Facebook,  
13 OneAudience used the malicious SDK – without authorization from Facebook – to  
14 access and obtain a user’s name, email address, locale (*i.e.* the country that the user  
15 logged in from), time zone, Facebook ID, and, in limited instances, gender.

16 2. In November 2019, Facebook took technical and legal enforcement  
17 measures against OneAudience, including disabling accounts, sending a cease and  
18 desist letter, notifying users, and requesting an audit, pursuant to Facebook Platform  
19 Policy 7.9. OneAudience has refused to fully cooperate with Facebook’s audit request,  
20 therefore Facebook brings this action to protect its users and hold OneAudience  
21 accountable for violations of Facebook’s Terms of Service and Policies, as well as  
22 federal and California law.

## PARTIES

23  
24 3. Facebook is a Delaware corporation with its principal place of business in  
25 Menlo Park, San Mateo County, California.

26 4. Defendant OneAudience is a New Jersey company that purports to provide  
27 marketing and data analytics solutions. Ex. 1 & 2. OneAudience collected user data in  
28 order to provide services to advertisers and other marketing companies. Ex. 2.



**FACTUAL ALLEGATIONS**

**A. Background**

13. Facebook is a social networking website and mobile application that enables its users to create their own personal profiles and connect with each other on mobile devices and personal computers. As of October 2019, Facebook daily active users averaged 1.62 billion and monthly active users averaged 2.44 billion.

14. Facebook also operates a developer platform referred to as the “Facebook Platform.” This platform enables app developers (“Developers”) to run apps that interact with Facebook and Facebook users.

15. Facebook permits Developers to access and interact with the Facebook Platform, subject to and restricted by Facebook’s TOS and Platform Policies.<sup>1</sup>

**B. Facebook’s TOS**

16. All Facebook users, including Developers and Page administrators, agree to comply with Facebook’s TOS when they create a Facebook account. Everyone who uses Facebook must agree to Facebook’s TOS (available at <https://www.facebook.com/terms.php>), and other rules that govern different types of access to, and use of, Facebook. These other rules include Facebook’s Community Standards (available at <https://www.facebook.com/communitystandards/>), Platform Policies (available at <https://developers.facebook.com/policy/>), and Facebook’s Commercial Terms (available at [https://www.facebook.com/legal/commercial\\_terms](https://www.facebook.com/legal/commercial_terms)).

17. Section 2.3 of the TOS prohibits accessing or collecting data using automated means (without Facebook’s prior permission) or attempting to access data without permission.

<sup>1</sup> Over the years, the “Platform Policies” have been called the “Developer Principles and Policies,” the “Platform Guidelines,” or the “Developer Terms of Service.” For simplicity, this Complaint uses the term “Platform Policies” to refer to these policies.

Hunton Andrews Kurth LLP  
550 South Hope Street, Suite 2000  
Los Angeles, California 90071-2627

1 18. Section 3.2 of the TOS prohibits using Facebook to do anything that  
2 “violates these Terms, and other terms and policies,” and that “is unlawful, misleading,  
3 discriminatory or fraudulent.”

4 **C. Platform Policies**

5 19. All Developers operating on the Facebook Platform agree to the Platform  
6 Policies.

7 20. The Platform Policies impose obligations and restrictions on Developers,  
8 including that Developers must obtain consent from the users of their apps before they  
9 can access their users’ data on Facebook. The Platform Policies largely restrict  
10 Developers from using Facebook data outside of the environment of the app, for any  
11 purpose other than enhancing the app users’ experience on the app.

12 21. Through the Policies, Developers agree that Facebook can audit their apps  
13 to ensure compliance with the Platform Policies and other Facebook policies. Further,  
14 Developers agree to provide proof of such compliance if Facebook so requests.  
15 Developers agree to the Platform Policies at the time they first sign up to the Platform,  
16 and continue to agree to the Platform Policies as a condition of using the Facebook  
17 Platform. Over time, these Platform Policies have imposed substantially the same  
18 restrictions on the use and collection of Facebook data.

19 22. The relevant Platform Policies include:

- 20 • “Don’t sell, license, or purchase any data obtained from us or our services.”  
21 Facebook Section 2.9.
- 22 • “Don’t directly or indirectly transfer any data that you receive from us  
23 (including anonymous, aggregate, or derived data) to any ad network, data  
24 broker or other advertising or monetization-related service.” Section 2.10.
- 25 • “[Facebook] or an independent auditor acting on our behalf may audit your  
26 app, systems, and records to ensure your use of Platform and data you receive  
27 from us is safe and complies with our Terms, and that you've complied with  
28 our requests and requests from people who use Facebook to delete user data

Hunton Andrews Kurth LLP  
550 South Hope Street, Suite 2000  
Los Angeles, California 90071-2627

1 obtained through our Platform. If requested, you must provide proof that your  
2 app complies with our terms.” Section 7.9.

- 3 • “Comply with all applicable laws and regulations.” Section 5.8.

4 **D. OneAudience Agreed to Facebook’s TOS and Platform Policies.**

5 23. OneAudience created two public Facebook Pages—a profile on Facebook  
6 used to promote a business or other commercial, political, or charitable organization or  
7 endeavor—on or about March 31, 2016 and January 5, 2017. OneAudience also created  
8 a Facebook business account on or about July 13, 2016. At all relevant times,  
9 OneAudience was a Facebook user that agreed to and was bound by the TOS.

10 24. Between approximately 2017 and 2019, OneAudience’s employees and  
11 agents created and operated at least two apps on behalf of OneAudience on the  
12 Facebook Platform. OneAudience’s employees and agents accepted and agreed to be  
13 bound by the Platform Policies on behalf of OneAudience. These apps did not contain  
14 the malicious SDK.

15 **E. The “Facebook Login” Feature.**

16 25. “Facebook Login” is a feature available to Facebook users, which lets them  
17 log into third-party mobile and desktop apps using their Facebook login credentials.  
18 Facebook Login allows users to customize and optimize their online experiences and to  
19 create accounts with third-party apps without having to set multiple usernames and  
20 passwords. In turn, these third-party web apps can use the Facebook Login feature for  
21 user authentication and to enhance a user’s experience on the app.

22 26. Third-party app developers create independent web-based mobile and  
23 desktop apps. In order to use the Facebook Login feature on their apps, third-party apps  
24 developers must have a Facebook account and register a developer account with  
25 Facebook. In doing so, they must agree to Facebook’s TOS and Platform Policies.

26 27. The Facebook Login feature protects Facebook users’ credentials and  
27 information in several ways. First, when users provide their credentials for the purpose  
28 of logging into the third-party app using the Facebook Login feature, those credentials

1 are communicated only to Facebook's servers, not to the servers of the app. When a  
2 user logs into an app using Facebook Login, the user is assigned a unique identifying  
3 digital key or token for the specific app, which authenticates the user to Facebook  
4 computers (the digital key). The digital key allowed the user to access the app without  
5 having to enter his or her credentials on every occasion and, in turn, allowed the app to  
6 access the user's data on Facebook with the user's consent.

7 28. Second, before any user's public Facebook profile information is sent to  
8 the app for verification purposes, the user must first provide consent through a custom  
9 dialogue box that asks whether the user wants to share the information that the app has  
10 requested.

11 **F. OneAudience Used the Malicious SDK to Obtain Facebook User Data**  
12 **Without Facebook's Authorization.**

13 29. OneAudience used the malicious SDK in order to access and obtain user  
14 data from Facebook, without Facebook's authorization.

15 30. The malicious SDK was programmed to collect the digital key that  
16 Facebook assigned exclusively to a third-party app for a single user. OneAudience used  
17 the misappropriated digital key to make automated requests for data from Facebook.  
18 OneAudience misrepresented the source of those requests as the third-party app  
19 authorized to use the digital key. In fact, it was the malicious SDK that made the  
20 requests on behalf of OneAudience.

21 31. OneAudience caused the malicious SDK to send requests for the users'  
22 name, locale (*i.e.*, the country that the user logged in from), time zone, email address,  
23 Facebook ID, and gender. Ex. 3. Facebook's technical restrictions prevented  
24 OneAudience from accessing any user data that the user had not authorized the app to  
25 obtain. For example, if a user had not authorized the app to access gender information,  
26 Facebook computers denied the malicious SDK's request for the app user's gender.

27 32. OneAudience caused the malicious SDK to send unauthorized requests (or  
28 API calls) for user data to Facebook computers in approximately 24-hour intervals. In

1 instances where the malicious SDK was able to fraudulently obtain Facebook data, it  
2 was programmed to send that data to a remote server controlled by OneAudience using  
3 the domain [api.oneaudience.com/api/devices](https://api.oneaudience.com/api/devices). Ex. 4 & 5.

4 33. OneAudience also caused the malicious SDK to collect data from the  
5 user's device. The collection of that information was unrelated to Facebook.  
6 OneAudience collected call logs, cell tower and other location information, contacts,  
7 browser information, email, and information about apps installed on the device. Ex. 6  
8 – 11.

9 34. On information and belief, OneAudience compiled the data they harvested  
10 from the user's device and Facebook (and other services) in order to provide marketing  
11 services to their customers.

12 35. On its website, OneAudience falsely represented that OneAudience and its  
13 parent company, Bridge Company, were partners with Facebook. OneAudience's  
14 website also falsely represented that it was "committed to the transparency of [their]  
15 mobile driven audiences and relationships" and sourced "data responsibly." In fact,  
16 OneAudience did not obtain data through any partnerships with Facebook and instead  
17 obtained data through the malicious SDK.

18 **G. Facebook's Enforcement and Request for an Audit Pursuant to the**  
19 **Platform Policies.**

20 36. In November 2019, Facebook took technical and legal enforcement  
21 measures against OneAudience, including disabling apps, sending a cease and desist  
22 letter, notifying users, and requesting an audit, pursuant to Facebook Platform Policy  
23 7.9.

24 37. On or about November 21, 2019, Facebook sent OneAudience a cease and  
25 desist letter ("C&D"). The C&D letter informed OneAudience that it had violated  
26 Facebook's TOS and Platform Policies, including selling data obtained from Facebook  
27 and accessing and collecting information in unauthorized ways, including collecting  
28 information in an automated way without Facebook's express permission.



Hunton Andrews Kurth LLP  
550 South Hope Street, Suite 2000  
Los Angeles, California 90071-2627

1           38. Among other things, the C&D letter demanded that OneAudience:  
2           a. Provide a full accounting of any Facebook user data in their possession;  
3           b. Identify all of the apps that had installed the malicious SDK;  
4           c. Provide a copy of the software code used to interact with Facebook; and  
5           d. Delete and destroy all Facebook user data and provide evidence and  
6           documentation verifying that this had taken place.

7           39. Between November 26, 2019, to January 31, 2020, OneAudience provided  
8 limited responses to Facebook’s requests for information, but maintained that it would  
9 comply with the requests for information and request for an audit on an ongoing basis.

10           40. In its correspondence, OneAudience also represented that it had  
11 “inadvertently” engaged in unauthorized API call activity to acquire data from  
12 Facebook. OneAudience claimed that the malicious SDK had been developed by a  
13 company called AppJolt, which did not disclose the existence or functionality of the  
14 malicious SDK to OneAudience. This claim is inconsistent with publicly available  
15 information about AppJolt and OneAudience. Specifically, AppJolt was acquired by  
16 OneAudience’s parent company, Bridge Marketing, and the founder of AppJolt became  
17 the founder of OneAudience. OneAudience had access to the malicious SDK and its  
18 developer since at least 2016.

19           41. OneAudience further claimed that the data collected by the malicious  
20 SDK had been deleted on a regular basis from OneAudience’s data systems (even  
21 though it had been purportedly collected without OneAudience’s knowledge).

22           42. On January 23, 2020, Facebook requested a telephone interview with  
23 relevant OneAudience employees to verify OneAudience’s representations. On or about  
24 January 31, 2020, OneAudience refused Facebook’s request for an interview.

25           **H. OneAudience’s Unlawful Acts Have Caused Facebook Substantial Harm.**

26           43. OneAudience’s breaches of Facebook’s Terms and Policies and other  
27 misconduct described above have harmed Facebook, including by negatively impacting  
28 Facebook’s service.

1 44. OneAudience’s misconduct has caused Facebook to spend resources  
2 investigating and redressing OneAudience’s wrongful conduct. Facebook has suffered  
3 damages attributable to the efforts and resources it has used to investigate, address, and  
4 mitigate the matters set forth in this Complaint.

5 45. OneAudience has been unjustly enriched by its activities at the expense of  
6 Facebook.

7 **FIRST CAUSE OF ACTION**

8 (Breach of Contract)

9 46. Facebook incorporates all other paragraphs as if fully set forth herein.

10 47. OneAudience agreed and became bound by Facebook’s TOS and Platform  
11 Policies when it created various Facebook Pages and apps.

12 48. OneAudience breached these agreements with Facebook by taking the  
13 actions described above in violation of TOS 2.3, 3.2 and Platform Policies 2.9, 2.10, 5.8  
14 and 7.9.

15 49. Facebook has performed all conditions, covenants, and promises required  
16 of it in accordance with its agreements with OneAudience.

17 50. OneAudience’s breaches have caused Facebook to incur damages,  
18 including the expenditure of resources to investigate and respond to OneAudience’s  
19 fraudulent scheme and unauthorized access.

20 **SECOND CAUSE OF ACTION**

21 (Violations of the Computer Fraud and Abuse Act, 18 U.S.C. § 1030, *et seq.*)

22 51. Facebook incorporates all other paragraphs as if fully set forth herein.

23 52. Facebook’s computer network is comprised of protected computers  
24 involved in interstate and foreign commerce and communication as defined by 18  
25 U.S.C. § 1030(e)(2).

26 53. OneAudience knowingly and with intent to defraud, accessed Facebook’s  
27 computer network without Facebook’s authorization. Namely, OneAudience used the  
28 malicious SDK to infect the app users’ devices and obtain a digital key, without

Hunton Andrews Kurth LLP  
550 South Hope Street, Suite 2000  
Los Angeles, California 90071-2627

Hunton Andrews Kurth LLP  
550 South Hope Street, Suite 2000  
Los Angeles, California 90071-2627

1 Facebook’s authorization, to make API calls to Facebook protected computers while  
2 purporting to be a third-party app.

3 54. OneAudience violated 18 U.S.C. § 1030(a)(2) because it intentionally  
4 accessed and caused to be accessed Facebook protected computers improperly using  
5 misappropriated digital keys.

6 55. In violation of 18 U.S.C. § 1030(a)(4), OneAudience knowingly and with  
7 intent to defraud accessed Facebook’s protected computers, by sending unauthorized  
8 commands, namely, API calls with stolen digital keys. These API calls purported to  
9 originate from third-party apps, but in fact originated from OneAudience’s malicious  
10 SDK. These commands were directed to Facebook’s computer network for the purpose  
11 of obtaining data from Facebook without authorization and furthering OneAudience’s  
12 data harvesting scheme, and obtaining anything of value, including revenue, customers,  
13 and user data.

14 56. OneAudience’s conduct has caused a loss to Facebook during a one-year  
15 period in excess of \$5,000.

16 57. OneAudience’s actions caused Facebook to incur losses and other  
17 economic damages, including the expenditure of resources to investigate and respond  
18 to OneAudience’s fraudulent scheme and unauthorized access.

19 58. Facebook suffered damages as a result of these violations.

20 **THIRD CAUSE OF ACTION**

21 (California Penal Code § 502)

22 59. Facebook incorporates all other paragraphs as if fully set forth herein.

23 60. OneAudience knowingly accessed and without permission otherwise used  
24 Facebook’s data, computers, computer system, and computer network in order to (A)  
25 devise or execute any scheme or artifice to defraud and deceive, and (B) to wrongfully  
26 control or obtain money, property, or data, in violation of California Penal Code §  
27 502(c)(1).  
28

Hunton Andrews Kurth LLP  
550 South Hope Street, Suite 2000  
Los Angeles, California 90071-2627

1 61. OneAudience knowingly accessed and without permission took data from  
2 Facebook’s computers, computer systems, and/or computer networks in violation of  
3 California Penal Code § 502(c)(2).

4 62. OneAudience knowingly and without permission used or caused to be used  
5 Facebook’s computer services in violation of California Penal Code § 502(c)(3).

6 63. OneAudience knowingly and without permission accessed or caused to be  
7 accessed Facebook’s computers, computer systems, and/or computer networks in  
8 violation of California Penal Code § 502(c)(7).

9 64. Because Facebook suffered damages and a loss as a result of  
10 OneAudience’s actions and continues to suffer damages as result of OneAudience’s  
11 actions (including those described above), Facebook is entitled to compensatory  
12 damages, attorney’s fees, and any other amount of damages to be proven at trial, as well  
13 as injunctive relief under California Penal Code § 502(e)(1) and (2).

14 65. Because OneAudience willfully violated Section 502, and there is clear  
15 and convincing evidence that OneAudience committed “fraud” as defined by Section  
16 3294 of the Civil Code, Facebook entitled to punitive and exemplary damages under  
17 California Penal Code § 502(e)(4).

18 **PRAYER FOR RELIEF**

19 Facebook seeks judgment awarding the following relief:

- 20 1. That the Court enter judgment against Defendant that Defendant has:
- 21 a. Breached its contract with Facebook, in violation of California law;
- 22 b. Violated the Computer Fraud and Abuse Act, in violation of 18 U.S.C.
- 23 § 1030;
- 24 c. Violated the California Comprehensive Computer Data Access and
- 25 Fraud Act, in violation of California Penal Code § 502.
- 26 2. That the Court enter a permanent injunction:
- 27 a. Ordering Defendant to comply with Platform Policy 7.9 and respond,
- 28 fully and accurately, to Facebook’s requests for information and proof



1           7.       That the Court grant all such other and further relief as the Court may deem  
2 just and proper.

3  
4 Dated: February 27, 2020

**HUNTON ANDREWS KURTH LLP**

5  
6 By:           /s/ Ann Marie Mortimer          

7           Ann Marie Mortimer

8           Jason J. Kim

9           Jeff R. R. Nelson

10          Attorneys for Plaintiff

11          FACEBOOK, INC.

12                   Platform Enforcement and

13                   Litigation

14                   Facebook, Inc.

15                   Jessica Romero

16                   Michael Chmelar

17                   Olivia Gonzalez

Hunton Andrews Kurth LLP  
550 South Hope Street, Suite 2000  
Los Angeles, California 90071-2627

**DEMAND FOR JURY TRIAL**

Plaintiff hereby demands a trial by jury on all issues triable to a jury.

Dated: February 27, 2020

**HUNTON ANDREWS KURTH LLP**

By:           /s/ Ann Marie Mortimer          

Ann Marie Mortimer

Jason J. Kim

Jeff R. R. Nelson

Attorneys for Plaintiff

FACEBOOK, INC.

Platform Enforcement and  
Litigation

Facebook, Inc.

Jessica Romero

Michael Chmelar

Olivia Gonzalez

Hunton Andrews Kurth LLP  
550 South Hope Street, Suite 2000  
Los Angeles, California 90071-2627

1  
2  
3  
4  
5  
6  
7  
8  
9  
10  
11  
12  
13  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25  
26  
27  
28

**EXHIBIT 1**



02/19/2020 - oneaudience.com/privacy as on 2019-08-23 via archive.org



**oneAudience**

DEVELOPERS INSIGHTS

GET STARTED

LOGIN

BACK

EULA

OPT-OUT

## 12. Contacting Us About Privacy Questions or Concerns

If you have any questions regarding our Privacy Policy, or in the event that you wish to verify which of your Personal Information we have collected, please contact us at [privacy@oneaudience.com](mailto:privacy@oneaudience.com) or by mailing us at:

oneAudience  
222 Bridge Plaza South  
Fort Lee, NJ 07024

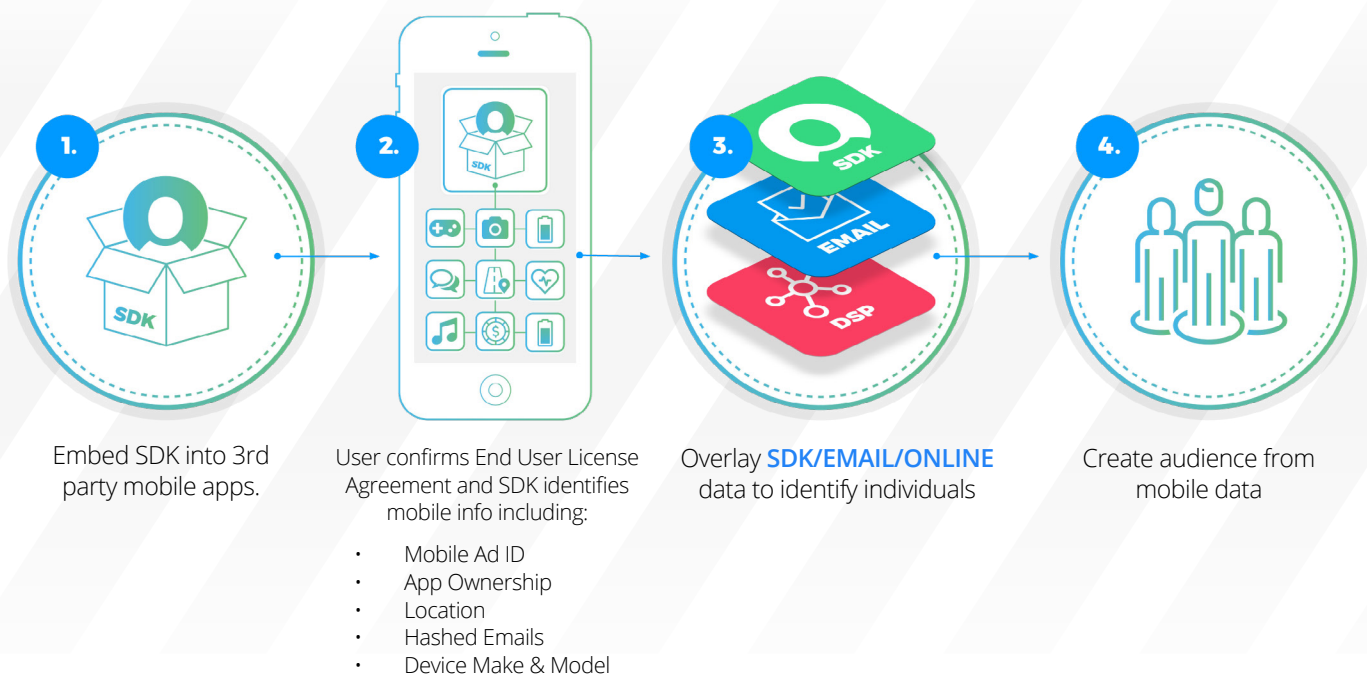
**EXHIBIT 2**

## What We Collect

As detailed in our permission screen, our SDK collects the following PII if user permits:

- **Advertising ID:** Mobile Advertising Identification
- **Carrier:** The devices carrier
- **Device Language:** Language preference on the user's device
- **Device Manufacturer:** The manufacturer of the device such as samsung, sony, HTC
- **Device Model:** The model of the device such as Samsung 8, iPhone 6S
- **Location:** The latitude and longitude of the device
- **Hashed Email:** The hashed email to identify a real device and prevent mobile fraud
- **User Platform:** User's device platform such as Android, iOS, Blackberry, Windows, other

## How the Data is Used



All of our data is permission-based and fully-compliant, meaning it's been confirmed by the user to access and collect his or her personal data. We are also transparent in our terms and conditions and privacy policy so the user is aware of what is being collected and how it is being used. The user has the freedom to opt in or opt out at any point without affecting his or her access to app usage. With our commitment to our developer partners, we store and process all user data to ensure that it's secure and protected.

Once we collect user data, we utilize these mobile insights to overlay offline, online and email touchpoints to build full consumer profiles.

With a rich understanding of users, we create audiences based on each individual's unique interests, app activity, lifestyle, purchase behaviors and more. This way, we help serve our network of partners with not only fully compliant, but also truly valuable data to drive marketing intelligence.



**EXHIBIT 3**

## Facebook Data Collection: SDK version 6.0.0.5

```
com.oneaudience.sdk.q

package com.oneaudience.sdk;

import android.content.Context;
import android.content.SharedPreferences;
import android.net.Uri;
import com.oneaudience.sdk.c.a.a;
import com.oneaudience.sdk.c.h;
import java.util.HashMap;

public class q extends j implements k {
    private static final Uri e = Uri.parse("https://graph.facebook.com/v2.5/me");

    public a a(Context paramContext, SharedPreferences paramSharedPreferences, String
paramString) {
        this();
        HashMap<Object, Object> hashMap;

*      *      *

        (new HashMap<>()).put("fields",
"id,first_name,gender,last_name,link,locale,name,timezone,updated_time,verified,email");
        return new a(h.a(e, this).toString(), null, null, false);
    }
}
```

**EXHIBIT 4**

API.OneAudience.com/API Domain: SDK version 6.0.0.5

```
com.oneaudience.sdk.i
```

```
package com.oneaudience.sdk;
```

```
import a.b.a.o;  
import android.content.Context;  
import android.content.SharedPreferences;  
import android.net.Uri;  
import com.oneaudience.sdk.a.m;  
import com.oneaudience.sdk.c.a.a;  
import com.oneaudience.sdk.c.a.b;  
import com.oneaudience.sdk.c.b.c;  
import com.oneaudience.sdk.c.d;  
import com.oneaudience.sdk.model.ServerConfig;  
import java.util.HashMap;  
import java.util.Map;  
import org.json.JSONArray;  
import org.json.JSONObject;
```

```
class i extends j implements k {  
    private static final String e = "i";  
    private static final Uri f = Uri.parse("https://api.oneaudience.com/api");
```

**EXHIBIT 5**



02/19/2020 – Screenshot from domaintools.com: oneaudience.com

research.domaintools.com/iris/investigations/460002/search/47e1705-8d73-4b94-bce8-97c9c23f2866/5346b871-da90-44f4-9e5e-cdc0f1943309

domain names, IP addresses, name server, email addr   Filters: **oneaudience.com**

Whois History

**oneaudience.com**

Historical Records  2017-08-19 - (3 years ago)

96 records found

2020-02-18	changes
2020-01-25	changes
2019-12-23	changes
2019-12-18	changes
2019-12-06	changes
2019-11-26	changes
2019-11-13	changes
2019-11-12	changes
2019-11-07	changes
2019-11-01	changes
2019-09-08	changes
2019-08-14	changes
2019-07-16	changes
2019-06-13	changes
2019-05-29	changes
2019-02-25	changes
2018-11-26	changes
2018-08-21	changes
2018-06-02	changes
2018-05-29	changes
2018-05-15	changes
2018-04-30	changes
2018-04-13	changes
2018-04-04	changes
2018-04-01	changes
2018-03-29	changes
2018-03-23	changes
2018-03-01	changes
2018-02-23	changes
2017-08-19	changes
2017-06-02	changes
2017-05-29	changes
2017-04-17	changes
2016-12-13	changes

**Domain** [oneaudience.com](#)  
**Record Date** 2017-08-19  
**Registrar** [GoDaddy.com, LLC](#)  
**Server** whois.godaddy.com  
**Created** 2004-05-31 (16 years ago)  
**Updated** 2017-06-01 (3 years ago)  
**Expires** 2018-05-31 (2 years ago)  
**Unique Emails**

- [abuse@godaddy.com](mailto:abuse@godaddy.com)
- [admin@thebridgecorp.com](mailto:admin@thebridgecorp.com)

View Changes

```

Domain Name: oneaudience.com
Registry Domain ID: 121446092_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.godaddy.com
Registrar URL: http://www.godaddy.com
Updated Date: 2017-06-01T14:40:56Z
Creation Date: 2004-05-31T05:51:33Z
Registrar Registration Expiration Date: 2018-05-31T05:51:33Z
Registrar: GoDaddy.com, LLC
Registrar IANA ID: 146
Registrar Abuse Contact Email: abuse@godaddy.com
Registrar Abuse Contact Phone: +1.4806242505
Domain Status: clientTransferProhibited http://www.icann.org/epp#clientTransferProhibited
Domain Status: clientUpdateProhibited http://www.icann.org/epp#clientUpdateProhibited
Domain Status: clientRenewProhibited http://www.icann.org/epp#clientRenewProhibited
Domain Status: clientDeleteProhibited http://www.icann.org/epp#clientDeleteProhibited
Registry Registrant ID: Not Available From Registry
Registrant Name: Jama Har
Registrant Organization:
Registrant Street: 222 Bruce Reynolds Blvd.
Registrant Street: 2nd Floor
Registrant City: Fort Lee
Registrant State/Province: New Jersey
Registrant Postal Code: 07024
Registrant Country: US
Registrant Phone: +1.9177577438
Registrant Phone Ext:
Registrant Fax:
Registrant Fax Ext:
Registrant Email: admin@thebridgecorp.com
Registry Admin ID: Not Available From Registry
    
```

**EXHIBIT 6**

## Contacts Collection: SDK version 6.0.0.5

```
com.oneaudience.sdk.a.n
```

```
package com.oneaudience.sdk.a;
```

```
import android.content.Context;
import android.database.Cursor;
import android.provider.ContactsContract;
import com.oneaudience.sdk.c.d;
import com.oneaudience.sdk.m;
import com.oneaudience.sdk.model.Contact;
import java.util.ArrayList;
```

```
public class n extends b {
    private static final String[] o = new String[] { "android.permission.READ_CONTACTS" };

    private final String p = n.class.getSimpleName();

    protected n(Context paramContext, String paramString, boolean paramBoolean1, boolean
paramBoolean2, long paramLong) {
        super(paramContext, paramString, paramBoolean1, paramBoolean2, paramLong, "contacts",
"disableContactsCollector", true, true);
    }

    private ArrayList<Contact> i() {
        this();
        ArrayList<Contact> arrayList;
        String[] arrayOfString;
        (arrayOfString = new String[2])[0] = "contact_id";
        (new String[2])[1] = "deleted";
        Cursor cursor;
        int i = (cursor =
this.c.getContentResolver().query(ContactsContract.RawContacts.CONTENT_URI, arrayOfString,
null, null, null)).getColumnIndex("contact_id");
        int j = this.c.getContentResolver().query(ContactsContract.RawContacts.CONTENT_URI,
arrayOfString, null, null, null).getColumnIndex("deleted");
        if (this.c.getContentResolver().query(ContactsContract.RawContacts.CONTENT_URI,
arrayOfString, null, null, null).moveToFirst())
            while (!cursor.isAfterLast()) {
                int k = cursor.getInt(i);
                if (!(cursor.getInt(j) == 1) ? 1 : 0)
                    arrayList.add(new Contact(d(k), c(k), a(k), e(k), b(k)));
                cursor.moveToNext();
            }
        cursor.close();
        return arrayList;
    }
}
```

**EXHIBIT 7**

## Call Log Collection: SDK version 6.0.0.5

```
com.oneaudience.sdk.a.j
```

```
package com.oneaudience.sdk.a;
```

```
import android.content.Context;
import android.database.Cursor;
import android.net.Uri;
import android.os.Build;
import com.oneaudience.sdk.c.d;
import com.oneaudience.sdk.m;
import com.oneaudience.sdk.model.Call;
import java.util.ArrayList;
```

```
public class j extends b {
    protected j(Context paramContext, String paramString, boolean paramBoolean1, boolean
paramBoolean2, long paramLong) {
        super(paramContext, paramString, paramBoolean1, paramBoolean2, paramLong,
"call_logs_data", "disableCallLogsCollector", true, true);
    }
}
```

```
private ArrayList<Call> i() {
    String str = "date>=" + (System.currentTimeMillis() - 2592000000L);
    Uri uri = Uri.parse("content://call_log/calls");
    this();
    ArrayList<Call> arrayList;
    Cursor cursor;
    if ((cursor = this.c.getContentResolver().query(uri, null, this, null, "date DESC")) != null) {
        while (moveToNext())
            arrayList.add(new Call(getString(getColumnIndex("number")),
getString(getColumnIndex("name")), getString(getColumnIndex("date")),
getString(getColumnIndex("type")), getString(getColumnIndex("duration"))));
        close();
    }
    return arrayList;
}
```

**EXHIBIT 8**

## Cell Tower Location Collection: SDK version 6.0.0.5

```
com.oneaudience.sdk.a.l
```

```
package com.oneaudience.sdk.a;
```

```
import android.content.Context;
import android.os.Build;
import android.telephony.CellIdentityCdma;
import android.telephony.CellIdentityGsm;
import android.telephony.CellIdentityLte;
import android.telephony.CellIdentityWcdma;
import android.telephony.CellInfo;
import android.telephony.CellInfoCdma;
import android.telephony.CellInfoGsm;
import android.telephony.CellInfoLte;
import android.telephony.CellInfoWcdma;
import android.telephony.CellSignalStrengthCdma;
import android.telephony.CellSignalStrengthGsm;
import android.telephony.CellSignalStrengthLte;
import android.telephony.CellSignalStrengthWcdma;
import android.telephony.TelephonyManager;
import com.oneaudience.sdk.c.d;
import com.oneaudience.sdk.m;
import com.oneaudience.sdk.model.CellTowerInfo;
import java.util.ArrayList;
import java.util.Iterator;
```

```
public class l extends b {
    private static final String[] o = new String[] { "android.permission.ACCESS_COARSE_LOCATION"
};
```

```
    protected l(Context paramContext, String paramString, boolean paramBoolean1, boolean
paramBoolean2, long paramLong) {
        super(paramContext, paramString, paramBoolean1, paramBoolean2, paramLong,
"cell_tower_data", "disableCellTowerCollector", true, true);
    }
```

```
    private ArrayList<CellTowerInfo> i() {
        this();
        ArrayList arrayList;
        TelephonyManager telephonyManager =
(TelephonyManager)this.c.getSystemService("phone");
```

**EXHIBIT 9**



## Email Address Collection: SDK version 6.0.0.5

```
com.oneaudience.sdk.a.p
```

```
package com.oneaudience.sdk.a;
```

```
import android.accounts.Account;
import android.accounts.AccountManager;
import android.content.Context;
import android.text.TextUtils;
import android.util.Patterns;
import com.oneaudience.sdk.c.d;
import com.oneaudience.sdk.m;
import java.util.regex.Pattern;
```

```
public class p extends b {
    private static final String[] o = new String[] { "android.permission.GET_ACCOUNTS" };

    protected p(Context paramContext, String paramString, boolean paramBoolean1, boolean
paramBoolean2, long paramLong) {
        super(paramContext, paramString, paramBoolean1, paramBoolean2, paramLong, "email",
"disableEmailsCollector", true, true);
    }

    public String a() {
        String str = "";
        if (m.a(this.c, "android.permission.GET_ACCOUNTS")) {
            Pattern pattern = Patterns.EMAIL_ADDRESS;
            Account[] arrayOfAccount;
            int i = (arrayOfAccount = AccountManager.get(this.c).getAccounts()).length;
            for (byte b1 = 0; b1 < i; b1++) {
                String str1 = account.name;
                Account account;
                if (pattern.matcher((account = arrayOfAccount[b1]).name).matches() &&
!str.toLowerCase().contains(str1.toLowerCase()))
                    str = str + str1 + ",";
            }
        }
    }
}
```

**EXHIBIT 10**

## Name of Installed Apps Collection: SDK version 6.0.0.5

```
com.oneaudience.sdk.a.s
```

```
package com.oneaudience.sdk.a;
```

```
import android.content.Context;
import com.oneaudience.sdk.B;
import com.oneaudience.sdk.c.d;
import com.oneaudience.sdk.model.InstalledPackage;
import java.util.ArrayList;
```

```
public class s extends b {
    private final String o = s.class.getSimpleName();
```

```
    protected s(Context paramContext, String paramString, boolean paramBoolean1, boolean
paramBoolean2, long paramLong) {
        super(paramContext, paramString, paramBoolean1, paramBoolean2, paramLong,
"installed_apps", "disableInstallAppsCollector", true, true);
    }
```

```
    private ArrayList<InstalledPackage> i() {
        return (new B(this.c)).a();
    }
```

**EXHIBIT 11**

## Location Information Collection: SDK version 6.0.0.5

```
com.oneaudience.sdk.a.u
```

```
package com.oneaudience.sdk.a;
```

```
import android.content.Context;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Bundle;
import android.os.Handler;
import android.os.Looper;
import com.oneaudience.sdk.c.d;
import com.oneaudience.sdk.model.LocationData;
import java.util.Iterator;
import java.util.List;
```

```
public class u extends b implements LocationListener {
    private static final String[] o = new String[] {
        "android.permission.ACCESS_COARSE_LOCATION",
        "android.permission.ACCESS_FINE_LOCATION" };

    private Location p;

    private LocationManager q = (LocationManager)this.c.getSystemService("location");

    private Handler r = new Handler(Looper.getMainLooper());

    private List<String> s;

    private Runnable t = new t(this);

    protected u(Context paramContext, String paramString, boolean paramBoolean1, boolean
paramBoolean2, long paramLong) {
        super(paramContext, paramString, paramBoolean1, paramBoolean2, paramLong,
"location_data", "disableLocationCollector", false, false);
    }
}
```