Why Does Your Data Leak?
Uncovering the Data Leakage in Cloud from Mobile Apps

Chaoshun Zuo, Zhiqiang Lin, and Yinqian Zhang

Department of Computer Science and Engineering
The Ohio State University

IEEE S&P 2019
Recent News Headlines about Cloud Data Leakage

Verizon data of 6 million users leaked online

by Selena Larson  @selenalarson
July 12, 2017 - 4:14 PM ET
Recent News Headlines about Cloud Data Leakage

Verizon data of 6 million users leaked online
by Selena Larson  @selenalarson
July 12, 2017, 4:14 PM ET

Cyber-Safe
Pentagon exposed some of its data on Amazon server
by Selena Larson  @selenalarson
November 17, 2017, 12:03 PM ET
Recent News Headlines about Cloud Data Leakage

Verizon data of 60 million customers at risk

Biggest Cloud Security Breaches in 2018

Cyber-Safe

Pentagon exposed some of its data on Amazon server

by Selena Larson  @selenalarson

November 17, 2017, 12:03 PM ET
Recent News Headlines about Cloud Data Leakage

- **Cloud Security Concerns in 2018: Data Breaches, Security Misconfigurations, AI and Botnets**
  - by Arun Balakrishnan
  - Amazon server

by Selena Larson  @selenalarson

November 17, 2017, 12:03 PM ET

---

**Introduction**

**Our Discovery**

**LeakScope**

**Evaluation**

**Related Work**

**Summary**

**References**
The Mobile Backend as a Service (mBaaS) Clouds

- Cloud App Engines
- Databases
- Push Notification
- Client Library
- Mobile Backend
- APIs running On the Cloud
- Client
- Server
The Mobile Backend as a Service (mBaaS) Clouds

Cloud App Engines

Mobile Backend

Databases

Push Notification

Client

Server

APIs running On the Cloud

Mobile Backend Client Library

AWS

Firebase

Azure

kinvey
Data Leakage is Essentially an Access Control Problem
Data Leakage is Essentially an Access Control Problem

- Access Control
  - Bob's Data
  - Alice's Data
Data Leakage is Essentially an Access Control Problem
Data Leakage is Essentially an Access Control Problem

Authentication

Authorization

Bob's Data

Alice's Data

Password Token
How Do Mobile Apps and mBaaS Cloud Communicate

Authentication

User A

Authorization

Cloud Resources
How Do Mobile Apps and mBaaS Cloud Communicate

Authentication

User A

Credential A (App Key)

Authorization

Cloud Resources
How Do Mobile Apps and mBaaS Cloud Communicate

Authentication

User A

Credential A (App Key)

Authorization

Cloud Resources
How Do Mobile Apps and mBaaS Cloud Communicate

**Authentication**

- User A
- Credential A (App Key)
- Cloud Resources

**Authorization**
How Do Mobile Apps and mBaaS Cloud Communicate

Authentication

User A
Credential A (App Key)

User B
Credential B (App Key)

Authorization

Cloud Resources

Related Work

References
How Do Mobile Apps and mBaaS Cloud Communicate

Authentication

Authorization

Developer/Administrator

Cloud Resources
How Do Mobile Apps and mBaaS Cloud Communicate

Authentication

Authorization

Developer/Administrator

Root Key

Cloud Resources
How Do Mobile Apps and mBaaS Cloud Communicate

Authentication

User A
Credential A (App Key)
Developer/Administrator
Root Key
User B
Credential B (App Key)

Authorization

Cloud Resources
Our Discovery

The Root Causes of the Cloud Data Leakage

1. Misuse of Various Keys in Authentication
   ▶ Microsoft Azure Storage
   ▶ Microsoft Azure Notification Hubs
   ▶ Amazon AWS

2. Misconfiguration of User Permissions in Authorization
   ▶ Google Firebase
   ▶ Amazon AWS
Our Discovery

The Root Causes of the Cloud Data Leakage

1. Misuse of Various Keys in Authentication
   - Microsoft Azure Storage
   - Microsoft Azure Notification Hubs
   - Amazon AWS

2. Misconfiguration of User Permissions in Authorization
   - Google Firebase
   - Amazon AWS
Our Discovery

The Root Causes of the Cloud Data Leakage

1. Misuse of Various Keys in Authentication
   - Microsoft Azure Storage
   - Microsoft Azure Notification Hubs
   - Amazon AWS

2. Misconfiguration of User Permissions in Authorization
   - Google Firebase
   - Amazon AWS
Misuse of Various Keys in Authentication

Authentication

User A
Credential A (App Key)
Developer/Administrator
Root Key
User B
Credential B (App Key)
Authorization

Cloud Resources
## Misuse of Root Keys in Microsoft Azure

<table>
<thead>
<tr>
<th>Service</th>
<th>Key Type</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Azure Storage</strong></td>
<td>Account Key</td>
<td>DefaultEndpointsProtocol=https; AccountName=<em>; AccountKey=</em></td>
</tr>
<tr>
<td></td>
<td>SAS</td>
<td>https://<em>.blob.core.windows.net/</em>?sv=<em>&amp;st=</em>&amp;se=<em>&amp;sr=b&amp; sp=rw&amp;sip=</em>&amp;spr=https&amp;sig=*</td>
</tr>
<tr>
<td><strong>Notification Hub</strong></td>
<td>Listening Key</td>
<td>Endpoint=sb://<em>.servicebus.windows.net/; SharedAccessKeyName= DefaultListenSharedAccessSignature; SharedAccessKey=</em></td>
</tr>
<tr>
<td></td>
<td>Full Access Key</td>
<td>Endpoint=sb://<em>.servicebus.windows.net/; SharedAccessKeyName= DefaultFullSharedAccessSignature; SharedAccessKey=</em></td>
</tr>
</tbody>
</table>
Misuse of Root Keys in Microsoft Azure

<table>
<thead>
<tr>
<th>Service</th>
<th>Key Type</th>
<th>Example</th>
</tr>
</thead>
</table>
| Azure Storage    | Account Key    | DefaultEndpointsProtocol=https; AccountName=*
AccountKey=* |
|                  | SAS            | https://*.blob.core.windows.net/*
?sv=*&st=*&se=*&sr=b&
sp=rw&sip=*&spr=https&sig=* |
| Notification Hub | Listening Key  | Endpoint=sb://*.servicebus.windows.net/; SharedAccessKeyName=*
DefaultListenSharedAccessSignature; SharedAccessKey=* |
|                  | Full Access Key| Endpoint=sb://*.servicebus.windows.net/; SharedAccessKeyName=*
DefaultFullSharedAccessSignature; SharedAccessKey=* |
Our Discovery

The Root Causes of the Cloud Data Leakage

1. Misuse of Various Keys in Authentication
   - Microsoft Azure Storage
   - Microsoft Azure Notification Hubs
   - Amazon AWS

2. Misconfiguration of User Permissions in Authorization
   - Google Firebase
   - Amazon AWS
Misconfiguration of User Permissions in **Authorization**

Authentication

- **User A**
  - Credential A (App Key)
  - Developer/Administrator
- **User B**
  - Credential B (App Key)

Authorization

- Cloud Resources
  - Root Key
  - Cloud Resources
Misconfiguration of User Permissions in **Authorization**

![Diagram of authentication and authorization process]

- **User A** (App Key: Credential A)
- **Developer/Administrator** (Root Key)
- **User B** (App Key: Credential B)
- **Cloud Resources**

Authentication process:
1. User A provides Credential A (App Key)
2. Developer/Administrator uses Root Key
3. User B provides Credential B (App Key)
4. Authorization decision made
Misconfiguration of User Permissions in Google Firebase

```
{
    "rules": {
        "users": {
            "$uid": {
                "read": "$uid === auth.uid",
                "write": "$uid === auth.uid"
            }
        }
    }
}
```

Figure: A Correct Firebase Authorization Rule
**Misconfiguration of User Permissions in Google Firebase**

```json
{
    "rules": {
        "users": {
            "$uid": {
                "read": "$uid === auth.uid",
                "write": "$uid === auth.uid"
            }
        }
    }
}
```

Figure: A Correct Firebase Authorization Rule

```json
{
    "rules": {
        "read": "auth != null",
        "write": "auth != null"
    }
}
{
    "rules": {
        "read": true,
        "write": true
    }
}
(a) (b)
```

Figure: Two Misconfigured Firebase Authorization Rules
Problem Statement

How to automatically detect the cloud leakage at scale

1. How to systematically identify various keys used by mobile apps (Cloud APIs)
2. How to identify the relevant key strings that are used by mobile apps (String Analysis)
3. How to design an obfuscation-resilient approach to identify cloud APIs and key strings of our interest (Obfuscation-Resilient)
4. How to determine the vulnerability without leaking sensitive information in the cloud (Vulnerability Confirmation)
Introducing LeakScope

Authentication
- Credential A (App Key)
- Root Key
- Developer/Administrator
- Credential B (App Key)
- User A
- User B

Authorization

Cloud Resources

APKs

Cloud API Identification

String Value Analysis

SDK APIs

Vulnerability Identification

Vulnerabilities
Cloud API Identification

1. APKs
2. Cloud API Identification
3. String Value Analysis
4. Vulnerability Identification
5. Vulnerabilities
6. SDK APIs
Cloud API Identification

<table>
<thead>
<tr>
<th>Cloud Service</th>
<th>APIs</th>
<th>Definition</th>
<th>Indexes of The String Parameters of Our Interest</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS</td>
<td>1*</td>
<td>TransferUtility: TransferObserver downloadUpload(String, String, File)</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>2*</td>
<td>AmazonS3Client: void S3objectAccess(String, String, ...)</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>3</td>
<td>CognitoCredentialsProvider: void &lt;init&gt;(String,String,String,String,...)</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>4</td>
<td>BasicAWSCredentials: void &lt;init&gt;(String,String)</td>
<td>0,1</td>
</tr>
<tr>
<td>Azure</td>
<td>5</td>
<td>MobileServiceClient: void &lt;init&gt;(String,Context)</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>6</td>
<td>MobileServiceClient: void &lt;init&gt;(String,Context)</td>
<td>0,1</td>
</tr>
<tr>
<td></td>
<td>7</td>
<td>NotificationHub: void &lt;init&gt;(String,Context)</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>CloudStorageAccount: CloudStorageAccount parse(String)</td>
<td>0</td>
</tr>
<tr>
<td>Firebase</td>
<td>9</td>
<td>FirebaseOptions: void &lt;init&gt;(String,String,String,String,String,String)</td>
<td>0,1,2,5</td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>FirebaseOptions: void &lt;init&gt;(String,String,String,String,String,String)</td>
<td>0,1,2,5</td>
</tr>
</tbody>
</table>
String Value Analysis

APKs

Cloud API Identification

String Value Analysis

Vulnerability Identification

SDK APIs

Vulnerabilities
String Value Analysis

1 package com.appname
2 public class ImagesHelper {
3    private final String storageAccountKey;
4    private final String storageAccountName;
5    
6    private ImagesHelper(Context arg3) {
7        int v0 = 2131099713;
8        int v1 = 2131099712;
9        this.storageAccountName = this.getResources().getString(v0);
10       this.storageAccountKey = this.getResources().getString(v1);
11    }
12
13    public void downloadImages(Callback arg5, OnDownloadImagesUpdateListener arg6) {
14        StringBuilder v0 = new StringBuilder();
15        v0.append("DefaultEndpointsProtocol=http;AccountName=");
16        v0.append(this.storageAccountName);
17        v0.append(";AccountKey=");
18        v0.append(this.storageAccountKey);
19        String v1 = v0.toString();
20        CloudStorageAccount v7 = CloudStorageAccount.parse(v1);
21        ...
22
23        ...
Vulnerability Identification

- APKs
- Cloud API Identification
- String Value Analysis
- Vulnerability Identification
- SDK APIs
- Vulnerabilities

Client Cloud Requiring Vulnerable Permission?
Non Vulnerable Unauthorized Operation Error
Access Data
Data Non Exist Error
Vulnerability Identification

APKs → Cloud API Identification → String Value Analysis → Vulnerability Identification

SDK APIs → Vulnærabilities

Client → Access Data

Cloud

- Requiring
- Non Vulnerable
- Vulnerable

Permission?

- Unauthorized Operation Error
- Data Non Exist Error

Non Vulnerable

Vulnerable

Data Non Exist Error

Unauthorized Operation Error
### Distributions of the Testing Apps

<table>
<thead>
<tr>
<th></th>
<th>Total #Apps</th>
<th>%</th>
<th>Non-Obfuscated #Apps</th>
<th>%</th>
<th>Obfuscated #Apps</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>w/ Cloud API</td>
<td>107,081</td>
<td>-</td>
<td>85,357</td>
<td>79.71</td>
<td>21,724</td>
<td>20.29</td>
</tr>
<tr>
<td>w/ AWS only</td>
<td>4,799</td>
<td>4.48</td>
<td>4,548</td>
<td>5.33</td>
<td>251</td>
<td>1.16</td>
</tr>
<tr>
<td>w/ Azure only</td>
<td>899</td>
<td>0.84</td>
<td>720</td>
<td>0.84</td>
<td>179</td>
<td>0.82</td>
</tr>
<tr>
<td>w/ Firebase only</td>
<td>99,186</td>
<td>92.63</td>
<td>78,475</td>
<td>91.94</td>
<td>20,711</td>
<td>95.34</td>
</tr>
<tr>
<td>w/ AWS &amp; Azure</td>
<td>3</td>
<td>0.00</td>
<td>2</td>
<td>0.00</td>
<td>1</td>
<td>0.00</td>
</tr>
<tr>
<td>w/ AWS &amp; Firebase</td>
<td>1,973</td>
<td>1.84</td>
<td>1,427</td>
<td>1.67</td>
<td>546</td>
<td>2.51</td>
</tr>
<tr>
<td>w/ Azure &amp; Firebase</td>
<td>210</td>
<td>0.20</td>
<td>174</td>
<td>0.20</td>
<td>36</td>
<td>0.17</td>
</tr>
<tr>
<td>w/ Three Services</td>
<td>11</td>
<td>0.01</td>
<td>11</td>
<td>0.01</td>
<td>0</td>
<td>0.00</td>
</tr>
</tbody>
</table>
# Distributions of the Testing Apps

<table>
<thead>
<tr>
<th></th>
<th>Total #Apps</th>
<th>%</th>
<th>Non-Obfuscated #Apps</th>
<th>%</th>
<th>Obfuscated #Apps</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>w/ Cloud API</td>
<td>107,081</td>
<td>-</td>
<td>85,357</td>
<td>79.71</td>
<td>21,724</td>
<td>20.29</td>
</tr>
<tr>
<td>w/ AWS only</td>
<td>4,799</td>
<td>4.48</td>
<td>4,548</td>
<td>5.33</td>
<td>251</td>
<td>1.16</td>
</tr>
<tr>
<td>w/ Azure only</td>
<td>899</td>
<td>0.84</td>
<td>720</td>
<td>0.84</td>
<td>179</td>
<td>0.82</td>
</tr>
<tr>
<td>w/ Firebase only</td>
<td>99,186</td>
<td>92.63</td>
<td>78,475</td>
<td>91.94</td>
<td>20,711</td>
<td>95.34</td>
</tr>
<tr>
<td>w/ AWS &amp; Azure</td>
<td>3</td>
<td>0.00</td>
<td>2</td>
<td>0.00</td>
<td>1</td>
<td>0.00</td>
</tr>
<tr>
<td>w/ AWS &amp; Firebase</td>
<td>1,973</td>
<td>1.84</td>
<td>1,427</td>
<td>1.67</td>
<td>546</td>
<td>2.51</td>
</tr>
<tr>
<td>w/ Azure &amp; Firebase</td>
<td>210</td>
<td>0.20</td>
<td>174</td>
<td>0.20</td>
<td>36</td>
<td>0.17</td>
</tr>
<tr>
<td>w/ Three Services</td>
<td>11</td>
<td>0.01</td>
<td>11</td>
<td>0.01</td>
<td>0</td>
<td>0.00</td>
</tr>
</tbody>
</table>
## Result of Cloud API Identification & String Value Analysis

<table>
<thead>
<tr>
<th>String Parameter Name</th>
<th>APIs</th>
<th>Non-Obfuscated</th>
<th>Obfuscated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>#API-Call</td>
<td>#APP</td>
</tr>
<tr>
<td><strong>AWS</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>bucketName</td>
<td>1*</td>
<td>2,460</td>
<td>1,229</td>
</tr>
<tr>
<td>bucketName</td>
<td>2*</td>
<td>2,069</td>
<td>1,703</td>
</tr>
<tr>
<td>identityPoolId</td>
<td>3</td>
<td>3,458</td>
<td>3,458</td>
</tr>
<tr>
<td>accessKey</td>
<td>4</td>
<td>3,280</td>
<td>1,769</td>
</tr>
<tr>
<td>secretKey</td>
<td>4</td>
<td>3,280</td>
<td>1,769</td>
</tr>
<tr>
<td><strong>Azure</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>appURL</td>
<td>5</td>
<td>185</td>
<td>39</td>
</tr>
<tr>
<td>appURL</td>
<td>6</td>
<td>824</td>
<td>316</td>
</tr>
<tr>
<td>appKey</td>
<td>6</td>
<td>824</td>
<td>316</td>
</tr>
<tr>
<td>connectionString</td>
<td>7</td>
<td>700</td>
<td>513</td>
</tr>
<tr>
<td>connectionString</td>
<td>8</td>
<td>345</td>
<td>97</td>
</tr>
<tr>
<td><strong>Firebase</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>google_app_id</td>
<td>9</td>
<td>2,378</td>
<td>1,228</td>
</tr>
<tr>
<td>google_api_key</td>
<td>9</td>
<td>2,378</td>
<td>1,228</td>
</tr>
<tr>
<td>firebase_database_url</td>
<td>9</td>
<td>2,378</td>
<td>1,228</td>
</tr>
<tr>
<td>google_storage_bucket</td>
<td>9</td>
<td>2,378</td>
<td>1,228</td>
</tr>
<tr>
<td>google_app_id</td>
<td>10</td>
<td>154,664</td>
<td>78,859</td>
</tr>
<tr>
<td>google_api_key</td>
<td>10</td>
<td>154,664</td>
<td>78,859</td>
</tr>
<tr>
<td>firebase_database_url</td>
<td>10</td>
<td>154,664</td>
<td>78,859</td>
</tr>
<tr>
<td>google_storage_bucket</td>
<td>10</td>
<td>154,664</td>
<td>78,859</td>
</tr>
</tbody>
</table>
## Statistics of The Detected Vulnerabilities

<table>
<thead>
<tr>
<th>The Root Cause</th>
<th>Non-Obfuscated</th>
<th>Obfuscated</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>#Apps</td>
<td>%</td>
</tr>
<tr>
<td>Azure</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Account Key Misuse</td>
<td>85</td>
<td>9.37</td>
</tr>
<tr>
<td>Full Access Key Misuse</td>
<td>101</td>
<td>11.14</td>
</tr>
<tr>
<td>AWS</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Root key Misuse</td>
<td>477</td>
<td>7.97</td>
</tr>
<tr>
<td>“Open” S3 Storage</td>
<td>916</td>
<td>15.30</td>
</tr>
<tr>
<td>Firebase</td>
<td></td>
<td></td>
</tr>
<tr>
<td>“Open” Database</td>
<td>5,166</td>
<td>6.45</td>
</tr>
<tr>
<td>No Permission Check</td>
<td>6,855</td>
<td>8.56</td>
</tr>
</tbody>
</table>
## Statistics of The Detected Vulnerabilities

<table>
<thead>
<tr>
<th>The Root Cause</th>
<th>Non-Obfuscated #Apps</th>
<th>Non-Obfuscated %</th>
<th>Obfuscated #Apps</th>
<th>Obfuscated %</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Azure</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Account Key Misuse</td>
<td>85</td>
<td>9.37</td>
<td>18</td>
<td>8.33</td>
</tr>
<tr>
<td>Full Access Key Misuse</td>
<td>101</td>
<td>11.14</td>
<td>12</td>
<td>5.56</td>
</tr>
<tr>
<td><strong>AWS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Root key Misuse</td>
<td>477</td>
<td>7.97</td>
<td>92</td>
<td>11.53</td>
</tr>
<tr>
<td>“Open” S3 Storage</td>
<td>916</td>
<td>15.30</td>
<td>195</td>
<td>24.44</td>
</tr>
<tr>
<td><strong>Firebase</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>“Open” Database</td>
<td>5,166</td>
<td>6.45</td>
<td>1,214</td>
<td>5.70</td>
</tr>
<tr>
<td>No Permission Check</td>
<td>6,855</td>
<td>8.56</td>
<td>2,168</td>
<td>10.18</td>
</tr>
</tbody>
</table>
## Statistics of The Detected Vulnerabilities

<table>
<thead>
<tr>
<th>#Downloads</th>
<th># Non-Vulnerable Apps</th>
<th># Vulnerable Apps</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Azure</td>
<td>AWS</td>
</tr>
<tr>
<td>1,000,000,000 − 5,000,000,000</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>500,000,000 − 1,000,000,000</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100,000,000 − 500,000,000</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>50,000,000 − 100,000,000</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>10,000,000 − 50,000,000</td>
<td>2</td>
<td>35</td>
</tr>
<tr>
<td>5,000,000 − 10,000,000</td>
<td>3</td>
<td>32</td>
</tr>
<tr>
<td>1,000,000 − 5,000,000</td>
<td>16</td>
<td>136</td>
</tr>
<tr>
<td>500,000 − 1,000,000</td>
<td>10</td>
<td>105</td>
</tr>
<tr>
<td>100,000 − 500,000</td>
<td>65</td>
<td>356</td>
</tr>
<tr>
<td>50,000 − 100,000</td>
<td>42</td>
<td>249</td>
</tr>
<tr>
<td>10,000 − 50,000</td>
<td>167</td>
<td>679</td>
</tr>
<tr>
<td>5,000 − 10,000</td>
<td>82</td>
<td>369</td>
</tr>
<tr>
<td>1,000 − 5,000</td>
<td>272</td>
<td>976</td>
</tr>
<tr>
<td>0 − 1,000</td>
<td>464</td>
<td>3,844</td>
</tr>
</tbody>
</table>

Table: The Number of Apps that Have Used the Cloud APIs in Each of The Accumulated Download Category.
## Statistics of The Detected Vulnerabilities

<table>
<thead>
<tr>
<th>#Downloads</th>
<th># Non-Vulnerable Apps</th>
<th># Vulnerable Apps</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Azure</td>
<td>AWS</td>
</tr>
<tr>
<td>1,000,000,000 – 5,000,000,000</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>500,000,000 – 1,000,000,000</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>100,000,000 – 500,000,000</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>50,000,000 – 100,000,000</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>10,000,000 – 50,000,000</td>
<td>2</td>
<td>35</td>
</tr>
<tr>
<td>5,000,000 – 10,000,000</td>
<td>3</td>
<td>32</td>
</tr>
<tr>
<td>1,000,000 – 5,000,000</td>
<td>16</td>
<td>136</td>
</tr>
<tr>
<td>500,000 – 1,000,000</td>
<td>10</td>
<td>105</td>
</tr>
<tr>
<td>100,000 – 500,000</td>
<td>65</td>
<td>356</td>
</tr>
<tr>
<td>50,000 – 100,000</td>
<td>42</td>
<td>249</td>
</tr>
<tr>
<td>10,000 – 50,000</td>
<td>167</td>
<td>679</td>
</tr>
<tr>
<td>5,000 – 10,000</td>
<td>82</td>
<td>369</td>
</tr>
<tr>
<td>1,000 – 5,000</td>
<td>272</td>
<td>976</td>
</tr>
<tr>
<td>0 – 1,000</td>
<td>464</td>
<td>3,844</td>
</tr>
</tbody>
</table>

Table: The Number of Apps that Have Used the Cloud APIs in Each of The Accumulated Download Category.
Engaging with the Cloud Providers

**Disclosed** all the vulnerabilities we have identified. Cloud providers further notified the app developers.

1. **Microsoft** immediately corrected the wrong documentation
2. **Google** plans to provide more user-friendly SDKs when configuring user permissions in authorization.
3. **Amazon** added new permission checks with its S3 storage in November 2017 (two weeks before we disclosed our details to them)
Disclaimer on the use of account key after we disclosed the vulnerability

```java
24  * MODIFY THIS!
25  *
26  * Stores the storage connection string.
27  +  * Only use Shared Key authentication (Account Key) for testing purposes!
28  +  * Your account name and account key, which give full read/write access to the associated Storage account,
29  +  * will be distributed to every person that downloads your app.
30  +  * This is not a good practice as you risk having your key compromised by untrusted clients.
31  +  * Please consult following documents to understand and use Shared Access Signatures instead.
32  +  * https://docs.microsoft.com/en-us/rest/api/storageservices/delegating-access-with-a-shared-access-signature
33  +  * https://docs.microsoft.com/en-us/azure/storage/common/storage-dotnet-shared-access-signature-part-1
27  */
28  public static final String storageConnectionString = "DefaultEndpointsProtocol=https;"
29  + "AccountName=[MY_ACCOUNT_NAME];"
```
Google’s Update

1. The big additions on Google’s side are tools for **local emulation** and **writing tests** against the database products including their security rules, which they expect to have a marked improvement on the ability of customers to test and validate security rules.

2. Additionally, they **have alerting for customers** (sent every few weeks) for anyone using the Realtime Database or Cloud Firestore with open rules.

3. They’re exploring more options, but those are a start.
Related Work

1. **Protocol Reverse Engineering.** A large body of research focusing on protocol reverse engineering [Bed, MLK+06, CKW07, CS07, WMKK08, LJXZ08, MWKK09, CPKS09]

Related Work

1. Mobile App Vulnerability Discovery.
   - Client Side: TaintDroid [EGC+10], PiOS [EKKV11], CHEX [LLW+12], SMV-Hunter [SSG+14].
   - Server Side: AutoForge [ZWWL16], SmartGen [ZL17], AuthScope [ZZL17].

2. Misconfiguration Vulnerability Identification: FIREMAN [YMS+06], ConfErr [KUC08], ConfAid [AF10], SPEX [XZH+13].
LeakScope

- A static analysis to identify server side data leakage vulnerabilities
- It performs cloud API identification, string value analysis to identify the vulnerabilities

Experimental Result w/ 100K apps
- 15,098 apps' cloud servers are vulnerable
- 200 Azure, 1,600 AWS, 13,200 Firebase
- Responsible disclosures were made to the cloud providers

Source code of LeakScope has been made available at https://github.com/OSUSecLab/LeakScope
Future Works

1. We only scratched the tip of the iceberg of the security of cloud based backend – mBaaS cloud backend.

2. What about their backend software stack (e.g., VMs, operating systems, network stacks)?

3. What about other vulnerabilities (e.g., SQL injection, XSS, XXE)?
Future Works

1. We only scratched the tip of the iceberg of the security of cloud based backend – mBaaS cloud backend.

2. What about their backend software stack (e.g., VMs, operating systems, network stacks)?

3. What about other vulnerabilities (e.g., SQL injection, XSS, XXE)?

Why Does Your Data Leak?

Uncovering the Data Leakage in Cloud from Mobile Apps

Chaoshun Zuo, Zhiqiang Lin, and Yinqian Zhang

Department of Computer Science and Engineering
The Ohio State University

IEEE S&P 2019
References I


References


## Backup

<table>
<thead>
<tr>
<th>App Name</th>
<th>App Description and Functionality</th>
<th>Obfuscated?</th>
<th>Data in Database/Storage</th>
<th>Privacy Sensitive?</th>
</tr>
</thead>
<tbody>
<tr>
<td>A1</td>
<td>Sending messages with multiple fancy features</td>
<td>✓</td>
<td>User Photos</td>
<td>✓</td>
</tr>
<tr>
<td>A2</td>
<td>Editing user photos with magical enhancements</td>
<td>✓</td>
<td>User Photos</td>
<td>✓</td>
</tr>
<tr>
<td>A3</td>
<td>Editing user photos with featured specialties</td>
<td>✓</td>
<td>User Photos; Posted Pictures</td>
<td>✓</td>
</tr>
<tr>
<td>A4</td>
<td>Allowing users to organize and upload photos</td>
<td>X</td>
<td>User Uploaded Pictures</td>
<td>✓</td>
</tr>
<tr>
<td>A5</td>
<td>Helping users in planning and booking trips</td>
<td>✓</td>
<td>User Photos</td>
<td>✓</td>
</tr>
<tr>
<td>A6</td>
<td>A game app to build and design attractive hotels</td>
<td>X</td>
<td>User Backups</td>
<td>✓</td>
</tr>
<tr>
<td>A7</td>
<td>A game app to express revenge on game NPCs</td>
<td>X</td>
<td>Premium Plug-ins</td>
<td>X</td>
</tr>
<tr>
<td>A10</td>
<td>Helping users to start a diet and control weight</td>
<td>✓</td>
<td>User Photos; Posted Pictures</td>
<td>✓</td>
</tr>
<tr>
<td>A11</td>
<td>Calculating and tracking calories for human health</td>
<td>X</td>
<td>User Photos</td>
<td>✓</td>
</tr>
<tr>
<td>A12</td>
<td>Showing fertility status from correspondent kits</td>
<td>X</td>
<td>User Uploaded Pictures</td>
<td>✓</td>
</tr>
<tr>
<td>A13</td>
<td>Helping users to easily play a popular game</td>
<td>X</td>
<td>Configurations about the Game</td>
<td>X</td>
</tr>
<tr>
<td>A14</td>
<td>A real time translation tool, for calls, chats, etc.</td>
<td>✓</td>
<td>User Photos; Chat History</td>
<td>✓</td>
</tr>
<tr>
<td>A15</td>
<td>Showing images of nations’ commemorative coins</td>
<td>✓</td>
<td>Coins Images</td>
<td>X</td>
</tr>
<tr>
<td>A16</td>
<td>A convenient tool to take notes with rich content</td>
<td>✓</td>
<td>User Uploaded Pictures</td>
<td>✓</td>
</tr>
<tr>
<td>A17</td>
<td>A convenient tool for users to schedule a taxi</td>
<td>X</td>
<td>Driver Photos</td>
<td>✓</td>
</tr>
<tr>
<td>A18</td>
<td>Allowing users to buy/renew general insurances</td>
<td>X</td>
<td>Inspection Videos</td>
<td>✓</td>
</tr>
<tr>
<td>A19</td>
<td>Providing accurate local weather forecast</td>
<td>✓</td>
<td>Device Info (IMEI, etc.)</td>
<td>✓</td>
</tr>
<tr>
<td>A20</td>
<td>Editing and enhancing users photos and selfies</td>
<td>X</td>
<td>User Info (①④); User Private Messages</td>
<td>✓</td>
</tr>
<tr>
<td>A21</td>
<td>Allowing users to guess information about music</td>
<td>✓</td>
<td>Music Details</td>
<td>X</td>
</tr>
<tr>
<td>A22</td>
<td>Allowing users to sell and buy multiple products</td>
<td>X</td>
<td>User Info (①②④); Transactions</td>
<td>✓</td>
</tr>
<tr>
<td>Photo Collage</td>
<td>Creating photo collage with personal photos</td>
<td>✓</td>
<td>User Info (②④)</td>
<td>✓</td>
</tr>
<tr>
<td>A23</td>
<td>Helping users to translate and learn languages</td>
<td>✓</td>
<td>User Info (①); Quiz Data</td>
<td>✓</td>
</tr>
<tr>
<td>A24</td>
<td>Editing user photos with effects for cartoon avatar</td>
<td>X</td>
<td>User Info (①); User Pictures</td>
<td>✓</td>
</tr>
<tr>
<td>A25</td>
<td>Help users to learn how to draw human bodies</td>
<td>✓</td>
<td>User Info (①②③); User Pictures</td>
<td>✓</td>
</tr>
<tr>
<td>A26</td>
<td>An offline bible learning app with texts and audios</td>
<td>X</td>
<td>User Info (①②④)</td>
<td>✓</td>
</tr>
<tr>
<td>A27</td>
<td>Music platform for hiphop mixtapes and musics</td>
<td>X</td>
<td>User Info (①②③); Play List</td>
<td>✓</td>
</tr>
<tr>
<td>A28</td>
<td>Helping users to learn drawing different things</td>
<td>✓</td>
<td>User Info (①②③); User Pictures</td>
<td>✓</td>
</tr>
</tbody>
</table>

Symbol ① denotes the user name, ② the user ID, ③ the user email, and ④ the user token.