

Technical Security Analysis

Mobile App "TikTok"

Security Risk Assessment from Switzerland's Point of View

v1.0 / 3c4514bd
18.04.2023 07:05

Document ID	NTC-20230223-1-en
Subject	Technical Security Analysis Mobile App "TikTok"
Version	v1.0 / 3c4514bd
Date	18.04.2023 07:05
Classification	Public
Authors	Tobias Castagna, Sven Fassbender, Dilip Many, Raphael M. Reischuk, Fabio Zuber
Responsible	Dr. Raphael M. Reischuk

Management Summary

Initial Situation and Background

In recent years, several countries have drawn attention to potential security risks associated with the use of the "TikTok" app from the Chinese provider *ByteDance*. In recent months, there has been discussion in various countries about a possible ban on the app, which has been partially implemented. Current examples are the EU Commission and the UK government, which decided to ban the app in spring 2023 due to security concerns. Swiss authorities and companies are confronted with the same issue. To support the decision-making process with independent information, the National Test Institute for Cybersecurity NTC, at the suggestion and in consultation with the National Center for Cybersecurity NCSC, investigated the TikTok-App.

This report provides an assessment of the possible risks associated with the use of the TikTok-App on managed devices used by employees of Swiss companies and public authorities. The report focuses on technical aspects around the protection of individual privacy, the avoidance of surveillance and espionage when using TikTok on Android and iOS mobile devices. Other aspects such as protection against manipulation, censorship or political opinion-making were deliberately not taken into account.

The time and resources available were used primarily to analyze critical and practical risks. Further detailed analyses, for example through reverse engineering or long-term behavioral observations, have not been carried out yet. In addition, care was taken to ensure that the test conditions were as close to reality as possible, this means that no third-party protective measures, such as Mobile Device Management (MDM) solutions, were used.

Summary Assessment

The observed behavior of the TikTok-App is basically in line with the expectations of a social media app. However, the app requests extensive and potentially problematic system permissions that could be misused for user monitoring. Examples of this are the access to the microphone, camera, and location services. These permissions can mostly be explained by the typical functionalities of a social media app. For example, the microphone and camera are needed to record videos – one of the main functionalities of the TikTok-App. Nevertheless, location data is sent to ByteDance Backend servers: Provided permission has been granted, this occurs every time the app is launched on iOS. One positive aspect is that permissions are usually only requested when they are actually required for a feature used by the user. It is also possible to use the app even if permissions are not granted or revoked.

In addition to the basic risks associated with the permissions requested, there are other conceptual risks. For example, messages sent through the application are not encrypted end-to-end. Therefore, ByteDance, for example, as the operator of the TikTok infrastructure, can view and modify the messages. This risk is likely higher for individuals than for companies and government agencies, as other channels are generally used by the latter to exchange sensitive data. No evidence of user monitoring was found during the audit. Considering the fact that

only the presence of vulnerabilities can be proven, but not their absence, no blanket declaration of harmlessness can be given. For example, the monitoring of users by the application is technically feasible due to the far-reaching permissions. The application could already contain hidden monitoring functions that are only triggered under certain conditions (e.g. at certain locations or at certain times). In addition, due to frequent updates, hidden functions could be added almost unnoticed. This applies to any application, especially those with broad permissions. It was also discovered that a small portion of the communication with TikTok's backend servers is additionally encrypted. The exact content of this communication is unknown, so it is unclear what information may be flowing through this channel.

In summary, it is recommended to critically question the use of the TikTok-App, especially in a business or government context. This also applies to other apps with broad permissions that are of limited use in business and government contexts. If such apps are permitted, technical and organizational measures should be taken to limit their use to the minimum necessary and to ensure that only the necessary app permissions are granted.

Further details on the identified risks and the associated recommended measures are provided in [Section 3](#) on [Page 9](#).

Context

The study was carried out at the suggestion of and in consultation with the NCSC. As this is an initiative project with funding and realization by the NTC, the objectives, scope and framework were defined by the NTC.

The investigation took place in the period from February 23 to March 24, 2023, and was conducted by a core team of three test experts, who were supported selectively and as needed by other specialists from the NTC competence network. A total of approximately 300 working hours were invested in the investigation.

The analysis was conducted under the most realistic test conditions possible, without any special protections, such as those provided by restrictively configured mobile device management (MDM) solutions. For more details on scope and limitations, see [Section 1](#) on [Page 4](#).

Document History

Version	Date	Modifications	Internal ID
1.0	2023-04-18, 08:00	Initial document	3c4514bd

Table of Contents

1	Scope and Limitations	4
1.1	Scope at a Glance	4
1.2	Scope in Detail	5
2	List of Findings	7
3	Findings in Detail	9
3.1	Communication with the Backend	9
3.2	Mobile App	16
4	Testcases	33
4.1	Network Communication	33
4.2	Privacy and Data Protection	34

1 Scope and Limitations

This section describes the scope of the security analysis performed. The self-imposed, technical and resource constraints are addressed in this chapter. An overview of the most important points follows, after which a detailed explanation is given.

1.1 Scope at a Glance

The review focused on risks to individual privacy, surveillance, and espionage. More detailed analysis, such as reverse engineering or long-term behavioral tracking, was not conducted.

The test cases considered in the analysis are listed in [Section 4](#) on [Page 33](#).

The following list describes the rough scope of the analysis:

- Communication between the mobile apps and the ByteDance Backend
- Requested permissions of the apps and access to sensors such as camera, microphone and GPS

The following areas and aspects were deliberately **not** examined in this analysis:

- TikTok's public website and other platforms not listed
- The effectiveness of the protection features offered by the operating systems, in particular the restriction of the rights of an app
- Any disclosure of personal data to ByteDance partners and third parties
- Processes and algorithms for moderation and censorship of the shared and displayed content
- Psychological factors such as influences on self-expression, pressure to succeed, concentration span, etc.

The following diagram shows a schematic overview of all those components that are part of the present security analysis.

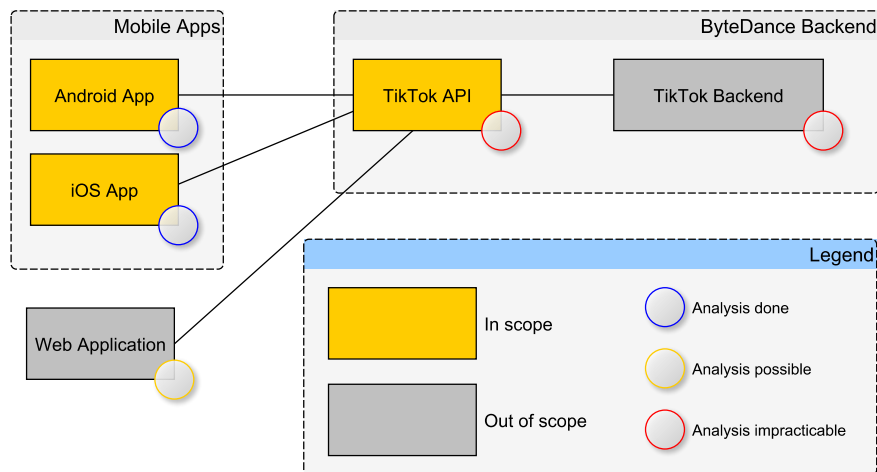


Figure 1: Component overview for the analysis

1.2 Scope in Detail

The time and resources available were used to analyze critical and practical risks. Further detailed analyses, for example by reverse engineering or long-term behavioral observations, were not possible in the time available and were not carried out. Such far-reaching investigations could possibly reveal any sophisticated monitoring techniques, if available. It is important to note, however, that only the presence of vulnerabilities can be proven, not their absence. This applies equally to more extensive testing. In addition, monitoring functions could be retrofitted due to the frequent updates of the app or could only be triggered under certain conditions (e.g., at certain locations or at certain times).

It is generally not known what is done with the sent data and metadata on ByteDance's side. It is impossible to check this without extensive cooperation from the provider. Therefore, no statement can be made about the use of the data and metadata.

The effectiveness of the protection features offered by the operating systems, especially the restriction of an app's rights, were not examined. The assessment assumes that these are fully effective.

The review focused on risks to individual privacy as well as surveillance and espionage when using the TikTok app on Android and iOS mobile devices in a business and government context. The TikTok website and other platforms such as Android TV were not considered. Likewise, social risks such as censorship, manipulation of opinion by algorithms, effects on the psyche of young people, or the like were not addressed.

The physical locations of the backend servers were not considered in more detail, as this is not a decisive criterion for making a statement about who the data is ultimately sent to or who has access to it. However, it seems relevant to mention that the data processor, ByteDance, is a company under Chinese legislation.

During the investigation, great care was taken to ensure that the test environment was as realistic as possible. Nevertheless, this cannot correspond one hundred percent to reality. Moreover, it cannot be ruled out that certain functionalities (problematic or unproblematic) could only be used or triggered under certain conditions. It is not possible to replicate all eventualities.

To enable better analysis of the network traffic, all communication was via Wi-Fi. The devices were not equipped with SIM cards and could not communicate via the mobile network.

Primarily, Android and iOS mobile devices with stock firmware were used for the study. In order to perform certain test cases that require in-depth system authorizations, iOS devices with jailbreaks were used in isolated cases.

All tests were performed on the most up-to-date version of the TikTok app at the start of the study:

- Android: 28.3.3
- iOS: 28.2.0 und 28.4.0 (When reinstalling during the test, only the latest version from the App Store could be installed. This is forced by Apple.)

It should be explicitly noted at this point that the review is a snapshot. Any adjustments to the app that are made before or after the fact cannot be recorded. The same applies to any app variants that are used in other countries or language regions.

As described in the management summary, the app's permissions play an important role in the risk assessment. In the analysis, the assumption is made that the permissions enforced by the Android and iOS operating systems take effect as expected and cannot be circumvented. This assumption can be made because the permission management on Android and iOS are generally considered robust and effective. However, this assumption was not verified in this study and is not valid without restrictions. For example, unaddressed vulnerabilities in the operating system or manipulated mobile devices ("rooting" on Android and "jailbreak" on iOS) could be used to bypass the permission management.

2 List of Findings

Below, all findings are listed and grouped into one of four categories: High Priority, Medium Priority, Low-Priority, and Information and Anomalies. All findings are discussed in detail in [Section 3](#).

High Priority (H)

Findings in this category correspond to severe vulnerabilities and should be analyzed and fixed immediately. Attackers may be able to exploit the vulnerabilities directly and cause severe damage.

NTCF-182 H	FB01	Transmission of contact hash values	9
NTCF-184 H	FB04	Lack of end-to-end encryption of direct messages	14
NTCF-186 H	FI02	Using location services every time the app is launched	20

Findings in this category may affect many or all users of the system. The vulnerabilities may be easily exploitable with sufficient privileges and are rather easy to detect. The vulnerabilities may be exploitable via the public Internet or by physically accessing a system. These vulnerabilities pose a realistic threat from amateurs and should be fixed before go-live.

Medium Priority (M)

Findings in this category should be analyzed and corrected in the medium term. Attackers may be able to exploit the vulnerabilities and cause moderate damage.

NTCF-188 M	FI04	Determining the device status	25
NTCF-190 M	FI06	Multi-factor authentication is not enforced	30

Findings in this category affect few to many users of the system. The vulnerabilities may be more difficult to exploit, and it may be more costly to detect them. The vulnerabilities may be exploitable via the Internet or by physically accessing a system. Thus, these vulnerabilities pose a realistic threat from advanced attackers and should be fixed within a short period of time.

Low Priority (L)

Findings in this category should be analyzed and reviewed for remediation in the medium term. Attackers may not be able to cause immediate damage, but they can at least gain an advantage.

NTCF-183 L	FB03	Encrypted content in HTTP headers	12
NTCF-185 L	FI01	Query installed apps	16
NTCF-187 L	FI03	Recurring background requests	23

NTCF-189 L FI05 Using an integrated browser 28

Findings in this category affect a small number of users or have no immediate impact on user data. The vulnerabilities tend to be complicated to exploit or require extensive privileges. Exploitation of these vulnerabilities may require knowledge of internal infrastructure or deep access to systems. These vulnerabilities can be understood as *defense-in-depth* controls that would improve the overall hardening of the system.


3 Findings in Detail

In this section, all findings are presented in detail.

This is how [Section 3.1 \(Page 9 ff.\)](#) describes the findings regarding communication with the backend. [Section 3.2 \(Page 16 ff.\)](#) describes the findings regarding mobile apps on iOS and Android.

3.1 Communication with the Backend

This chapter describes the findings regarding communication with the backend.

Finding NTCF-182  (Transmission of contact hash values): **The TikTok mobile app transmits hash values of existing contacts. ByteDance can use this information to establish connections between users.** FB01 [20230308]

Background

The TikTok-App transmits the hash values of all contacts from the address book, that contain a mobile number, to a ByteDance Backend. This information is used by ByteDance to suggest contacts known to users within the TikTok system as friends. While the contact information is not made available in plain text, it must be assumed that ByteDance is able to reconstruct the information from the hash values. Since there is a limited number of possible enumerable phone numbers, they can be reconstructed with high probability using the hash values. ¹.

This information is of great value to ByteDance because phone numbers can usually be assigned to a specific individual. If users grant the TikTok-App permission to access contacts, this information is transmitted directly to ByteDance. This takes place without the consent of the individuals concerned.

Evidence

The following HTTP request from an iOS device shows an example of submitting a contact to a ByteDance Backend:

¹ Theoretically possible, up to 250 billion numbers may exist worldwide; effectively, however, closer to ten billion probably exist. Source:https://en.wikipedia.org/wiki/List_of_countries_by_number_of_mobile_phones_in_use.

```
1 POST /aweme/v1/upload/hashcontacts/?version_code=[...] HTTP/2
2 Host: api16-normal-useast1a.tiktokv.com
3 [...]
4
5 need_unregistered_user=1&people_contact_list=%5B%7B%22contact%22%3A%2255282
  c18206b9beb9998f5eaa15b85c9388463965678af5209e2cc3a3ff5b947%22%2C%22phone_list%22%3A%5B%7B
  %22name%22%3A%2255282c18206b9beb9998f5eaa15b85c9388463965678af5209e2cc3a3ff5b947%22%2C%22
  hashed_phone%22%3A%221c1127746328b46401d4a1a8e296d09e2b1888e993e371a0a73cd21e5290675c%22%2
  C%22region_code%22%3A%223d914f9348c9cc0ff8a79716700b9fcd4d2f3e711608004eb8f138bcbaf7f14d9
  %22%7D%5D%7D%5D&scene=1&sync_only=1
```

The content of the above HTTP request is encoded. For better readability, the content is shown decoded below:

```
1 [
2   {
3     "contact": "55282c18206b9beb9998f5eaa15b85c9388463965678af5209e2cc3a3ff5b947",
4     "phone_list": [
5       {
6         "name": "55282c18206b9beb9998f5eaa15b85c9388463965678af5209e2cc3a3ff5b947",
7         "hashed_phone": "1c1127746328b46401d4a1a8e296d09e2b1888e993e371a0a73cd21e5290675c",
8         "region_code": "3d914f9348c9cc0ff8a79716700b9fcd4d2f3e711608004eb8f138bcbaf7f14d9"
9       }
10    ]
11  }
12 ]
```

Preconditions

In order for the TikTok-App to access the contacts, the users must grant this permission to the app. After logging-in on the TikTok-App, this permission is requested by the app. Once permission is granted, the local contacts are hashed and sent to the ByteDance Backend.

Impact

Knowing which contacts TikTok users have allows ByteDance to draw a wide variety of conclusions. Since phone numbers can usually be clearly assigned to an individual, this information is to be considered sensitive. Since neither the users nor ByteDance inform the data subjects about the sharing of the information, it must be assumed that the observed behavior violates the principle of transparency (Art. 4 para. 4 DPA) as well as the principle of good faith (Art. 4 para. 2 DPA). The technical measures taken – namely the derivation of hash values – does not provide sufficient protection for the confidentiality of this information.

Within the time frame of the investigation, it was not possible to determine the hash method used by means of reverse engineering. However, ByteDance must be able to interpret the hash values, otherwise an assignment to other TikTok users is not possible. Therefore, it can be assumed that a deterministic hash procedure is used, which generates the same hash value for each operation based on the same input. This could be confirmed experimentally by creating two different contacts with the same phone number. The subsequently transmitted hash values differed only in the name, but not in the phone number.

```
1  [
2  {
3    "contact": "55282c18206b9beb9998f5eaa15b85c9388463965678af5209e2cc3a3ff5b947",
4    "phone_list": [
5      {
6        "name": "55282c18206b9beb9998f5eaa15b85c9388463965678af5209e2cc3a3ff5b947",
7        "hashed_phone": "1c1127746328b46401d4a1a8e296d09e2b1888e993e371a0a73cd21e5290675c",
8        "region_code": "3d914f9348c9cc0ff8a79716700b9fcd4d2f3e711608004eb8f138bcba7f14d9"
9      }
10   ]
11 },
12 {
13   "contact": "8c41d3623e50d6a373040e51e1ce710eeb57b798676870cdf13ea3b1306c1da0",
14   "phone_list": [
15     {
16       "name": "8c41d3623e50d6a373040e51e1ce710eeb57b798676870cdf13ea3b1306c1da0",
17       "hashed_phone": "1c1127746328b46401d4a1a8e296d09e2b1888e993e371a0a73cd21e5290675c",
18       "region_code": "3d914f9348c9cc0ff8a79716700b9fcd4d2f3e711608004eb8f138bcba7f14d9"
19     }
20   ]
21 }
22 ]
```

Since ByteDance knows the hash algorithm used and any salt values, it is able to derive the associated phone number from the hash values, e.g. via precalculated tables. The same statement applies to the other hash values.

Recommendations

It is recommended not to allow the TikTok-App to access the contacts.

Finding NTCF-183 L (Encrypted content in HTTP headers): **The TikTok-App sends local content to the ByteDance Backend using encrypted HTTP headers.** FB03 [20230308]

Background

The TikTok-App sends content to a ByteDance Backend using non-standard HTTP headers. The content of these HTTP headers is apparently partly encrypted. However, it is unclear which content is encrypted here. In particular, the *X-Argus* header can have a length of up to 600 characters and is transmitted in every API call. Thus, there may be larger amounts of data contained in this header whose origin is unclear.

Examination of the data transmitted in these headers was not conclusively possible in the time available.

Evidence

The following excerpt shows an example of the *X-Argus* header:

```
1 POST /v3/conversation/get_read_index?aid=1233&device_platform=iphone&version_code=2840 HTTP/2
2 Host: api16-normal-useast1a.tiktokv.com
3 X-Argus: [...]
4 X-Gorgon: [...]
5 X-Khronos: 1678352403
6 X-Ladon: [...]
```

Research has shown that some information about the encryption, as well as presumed decrypted content, has already been publicly documented. For example, the following GitHub repository² records how the data is encrypted. It is also documented that the encrypted data is serialized in the `protobuf` format. The `protobuf` decrypted in this repository, does not contain any information that indicates sensitive content.

The NTC is currently considering further steps to verify the header in more detail and is in contact with the author of the repository.

Preconditions

Sending content via HTTP headers is possible without any further requirements as long as the app has access to the Internet. Since this is necessary for its correct functioning, Internet access can be assumed.

Impact

Although the encrypted data was not examined in more detail, some conjectures can be made about it. It is not assumed that the HTTP headers contain data collected via the camera, mi-

² <https://github.com/xteky/TikTok-X-Argus>

crophone or media library, for example. Access to these interfaces of the operating system is only possible with the permission of the user. In addition, a currently active access is recognizable via a *Sensor Notification* (green dot in the upper right corner under Android, colored dot in the upper right corner under iOS) and is additionally logged in the *App Privacy Report* under iOS and Android (if this is activated). Misuse of such interfaces could not be detected during the investigation.

However, it would basically be possible to transmit any data that the app has access to – e.g., a list of the apps currently installed on the operating system (see [Finding NTCF 185](#)) or the location determined at startup (see [Finding NTCF 186](#)).

Since the encrypted content could not be decrypted or made visible by means of reverse engineering in the time available, only a limited statement can be made about the possible effects. In principle, encryption is only used when the confidentiality of information is to be protected. At the same time, however, the use of encryption also renders the content provided with it unrecognizable. This makes it difficult for security researchers to determine what data is being transmitted.

Whether the encryption used here serves to legitimately protect sensitive information or to obscure the nature of the data being transmitted cannot be judged.

Recommendations

It is recommended that a more in-depth review be conducted. This should aim to make the encrypted content transmitted to the ByteDance Backend visible.

Finding NTCF-184 H (Lack of end-to-end encryption of direct messages): **The TikTok-App does not encrypt direct messages between users end-to-end. The operator of the infrastructure and the operator of the service can view the contents of direct messages.** iOS Android

FB04

[20230308]

Background

When sending a direct message, the TikTok-App must first transmit it to ByteDance's servers, which in turn deliver it to the respective recipient. To do this, ByteDance uses the infrastructure of the company *Akamai Technologies*. Since the direct messages sent are not protected by an additional layer of encryption (so-called end-to-end encryption), both *Akamai Technologies* and ByteDance, as well as any other infrastructure provider involved in the data transmission, could read and modify their content.

Evidence

The following snippet shows the direct message *Hey* sent in the HTTPS transport-only encrypted HTTP body of the request:

```
1 POST /v1/message/send?aid=1233&device_platform=iphone&version_code=2840 HTTP/2
2 Host: api16-normal-useast1a.tiktokv.com
3 [...]
4
5 [...]
6 {"aweType":0,"text":"Hey"}
7 [...]
```

Preconditions

Since the connection between the app and the servers of *Akamai Technologies* resp. ByteDance is encrypted by HTTPS, the lack of end-to-end encryption can only be exploited by these infrastructure providers to read the messages.

Impact

Both the operator of the infrastructure (*Akamai Technologies*) and the recipient of the information (ByteDance) can view and modify the contents of the direct messages. There is a possibility that the users of the app are not aware of this fact. Thus, users could use the TikTok-App to send potentially sensitive information via direct message to other users. Since the confidentiality of the messages is not guaranteed for the aforementioned parties, damage can occur depending on the content of the direct messages.

Recommendations

It is recommended to use TikTok's direct messaging only for non-sensitive content.

3.2 Mobile App

In this chapter, findings that affect the design of mobile apps are recorded.

Finding NTCF-185 L (Query installed apps): The TikTok-App for iOS checks whether certain other apps are installed on the smartphone during execution. The publisher of the app can use this information in many ways. There is currently no evidence that the collected data is sent to the backend servers. iOS FI01 [20230308]

Background

The TikTok-App for iOS checks whether certain apps are installed on the smartphone when it runs. The observed behavior may be related to the fact that the TikTok-App displays more or less buttons (e.g. for sharing content) depending on which other apps are installed. Since this information can, in the worst case, potentially reveal sensitive information about the user (e.g., affiliation to a certain ethnicity or religion), this querying should give cause for concern.

At this point, however, it is unclear whether the information is sent to the backend servers and analyzed by ByteDance. For this reason, the finding in this document is classified as low risk.

Evidence

The following excerpt from the `Info.plist` file of the TikTok-App shows the current list of third-party apps that can be detected by the TikTok-App at the time of the investigation:

```
1 <key>LSApplicationQueriesSchemes</key>
2 <array>
3 <string>akulaku</string>
4 <string>gojek</string>
5 <string>tngdwallet</string>
6 <string>tg</string>
7 <string>viber</string>
8 <string>fbapi</string>
9 <string>fb-messenger-api</string>
10 <string>fbauth2</string>
11 <string>fbshareextension</string>
12 <string>kakao61f447fe9723aa9c0b67a52eeb998e77</string>
13 <string>kakaokompassauth</string>
14 <string>storykompassauth</string>
15 <string>kakaolink</string>
16 <string>kakaotalk-5.9.7</string>
17 <string>storylink</string>
18 <string>line</string>
19 <string>instagram</string>
20 <string>instagram-stories</string>
21 <string>lineauth</string>
22 <string>line3rdp.com.zhiliaoapp.musically</string>
23 <string>whatsapp</string>
24 <string>fb-messenger-platform-20150714</string>
25 <string>zalo</string>
26 <string>twitter</string>
27 <string>twitterauth</string>
28 <string>bandapp</string>
29 <string>snapchat</string>
30 <string>kakaostory</string>
31 <string>navercafe</string>
32 <string>naverblog</string>
33 <string>vkauthorize</string>
34 <string>vk</string>
35 <string>vk-share</string>
36 <string>fb-messenger-share-api</string>
37 <string>fb-messenger</string>
38 <string>itms-beta</string>
39 <string>comgooglemaps</string>
40 <string>resso</string>
41 <string>ttmusic</string>
42 <string>mobilelegends</string>
43 <string>snssdk1233</string>
44 <string>ascendmoney</string>
45 <string>boostapp</string>
46 <string>momo</string>
47 <string>capcut</string>
48 <string>capcut840</string>
49 <string>reddit</string>
50 <string>scbeasy</string>
51 <string>lemon8opensdk</string>
52 <string>tiktoknow</string>
53 <string>lark</string>
54 <string>https</string>
55 <string>http</string>
56 </array>
```

In fact, while using the TikTok-App on the NTC analyst's device, the presence of the following third-party apps was detected:

```
1  canOpenURL: capcut://
2  canOpenURL: tiktoknow://
3  canOpenURL: kakaostory://
4  canOpenURL: zalo://
5  canOpenURL: whatsapp://
6  canOpenURL: navercafe://
7  canOpenURL: viber://
8  canOpenURL: bandapp://
9  canOpenURL: instagram://app
10 canOpenURL: twitter://
11 canOpenURL: naverblog://
12 canOpenURL: line://
13 canOpenURL: snapchat://
14 canOpenURL: tg://
15 canOpenURL: instagram-stories://share
16 canOpenURL: vkauthorize://authorize
17 canOpenURL: kakaolink://
18 canOpenURL: fb-messenger-share-api:/
19 canOpenURL: reddit://
20 canOpenURL: fbapi://
```

The following Frida script was used to dynamically analyze the behavior of the TikTok-App:

```
1  /*
2  $ frida -U -f com.zhiliaoapp.musically -l tiktok.js
3  */
4  Interceptor.attach(ObjC.classes.UIApplication["-_canOpenURL:"].implementation, {
5    onEnter: function (args) {
6      console.log('canOpenURL: ', ObjC.Object(args[2]).toString());
7    },
8    onLeave: function (retval) {
9    }
10 });
```

On Android, the `android.permission.get_tasks` permission is implicitly granted when the TikTok-App is installed. This can be used to read which processes are currently running. Due to time constraints, this was not examined in more detail on Android.

Preconditions

The entries in the `Info.plist` file (see Evidence section above) enable the querying of third-party apps. Among other things, this is checked as part of the Apple App Store review. Apparently, the App Store operator has no concerns that the permissions granted, could be abused by ByteDance.

Impact

Such a query may allow conclusions to be drawn about sensitive information concerning the users. For example, if a certain app is mainly used by people belonging to a certain ethnicity or religion, knowing whether the app is installed may allow the company ByteDance to draw corresponding conclusions.

A transmission of information about installed third-party apps to a ByteDance Backend was not detected during the analysis. However, it cannot be ruled out that this information is contained

in encoded or encrypted content of the HTTP requests (see [Finding NTCF 183](#)).

If this information is actually only used locally on the smartphone, the behavior of the TikTok-App is harmless. It only becomes problematic if the information is transmitted to a ByteDance Backend. However, this could not be ruled out in this study.

Recommendations

In practice, users have no way to prevent this behavior of the TikTok-App. A more thorough analysis whether the information is transmitted to a ByteDance Backend is recommended.

Finding NTCF-186 **H** (Using location services every time the app is launched): **The TikTok iOS app uses location services every time the app is launched and sends the exact location of the smartphone to a ByteDance Backend. The publisher of the app can use this information in many ways.** **iOS** **FI02** [20230308]

Background

On iOS, the TikTok-App transmits the user's latitude and longitude to a ByteDance Backend every time it is launched. This disclosure of sensitive information to ByteDance does not seem necessary for the intended use of the TikTok-App and thus exposes users to the risk of being geographically located.

On Android, no communication of the current location to a ByteDance Backend was detected when the TikTok-App was launched.

Evidence

The following HTTP request shows the transmission of the geographical longitude and latitude to ByteDance at the time of app launch:

```
1 POST /tiktok/location/submit-v2/?app_id=1233&sdk_version=2.2.6&version_code=28.4.0&language=en
  &app_name=musical_ly&app_version=28.4.0&op_region=CH&residence=CH&device_id
  =7207398165421950470&channel=App%20Store&mc_mnc=&tz_offset=3600&account_region=ch&
  sys_region=CH&aid=1233&locale=en&screen_width=640&uoo=1&openudid=
  ec168dab9cc9d038a6999641c617545c292650c0&cdid=A6F8EFE2-4C1A-412C-9DBB-8DBD131C1212&os_api
  =18&idfv=39656374-6B9F-4DA1-B426-15890ADE37C3&ac=WIFI&os_version=14.3&app_language=en&
  content_language=&tz_name=Europe/Zurich&current_region=CH&device_platform=iphone&
  build_number=284022&device_type=iPhone8,4&iid=7207399044682761990&idfa=9CFE9D17-82B9-42F0
  -8E68-67F83D99A1CF HTTP/2
2 Host: api16-normal-useast1a.tiktokv.com
3 [...]
4
5 {
6   "is_vpn" : false,
7   "status" : {
8     [...]
9   },
10  [...],
11 },
12 "is_proxy" : true,
13 "location" : {
14   "sys_location" : {
15     "locate_type" : 5,
16     "encrypted_lat" : "1|705383242|715618705|-1447592911|1|[...]|mV2IXi288=",
17     "provider" : "iOS",
18     "altitude_accuracy" : -1,
19     "province" : "ZH",
20     "timestamp" : 1678342297,
21     "address" : "ZH[...]",
22     "encrypted_lng" : "1|713618672|714862390|36977364|1|iy3fSH+Ivv\[...\]\NLR09Y=",
23     "city" : "[...]",
24     "coordinate_system" : "wgs84",
25     "accuracy" : 13373.875635070053,
26     "altitude" : 0,
27     "country" : "Switzerland",
28     "district" : "[...]",
29     "disable_location_shift" : 0
30   }
31 },
32 [...]
33 }
```

Preconditions

To access the exact location data of the users, the TikTok-App needs the permission to access location services. This is first requested when a video is uploaded for the first time and the **Location** field is selected. Users have the option on both iOS and Android to either grant the app one-time access to location services or to allow access permanently as long as the app is in use. If the user selects the option that allows access to location services while the app is in use, the iOS app will send the exact location to a ByteDance Backend every time the app is launched in the future. On Android, this behavior was not detected when the app was launched.

The geographical longitude and latitude transmitted by iOS appear to be encrypted. It is suspected that this encryption serves an exclusion of the infrastructure operator. Thus, the infrastructure operator – in this case *Akamai Technologies* – should not be able to read the exact geographic information. However, the determined country, region, city and address are transmitted unencrypted. This information can therefore also be read by the infrastructure operator.

Impact

Both the infrastructure operator (*Akamai Technologies*) and the recipient of the information (ByteDance) know the location of the TikTok-App user when the app is launched. This information can be used in conjunction with other datapoints to create a movement profile of a person.

Recommendations

It is recommended not to grant permission to access location services. Alternatively, when uploading a video, the location can be specified manually if required.

Finding NTCF-187 L (Recurring background requests): While the app is running in the background, it sends HTTP requests to the backend every hour. This allows a rough location based on the IP address. iOS Android FI03 [20230308]

Background

Apps can also send data while they are running in the background. However, the permission systems of Android and iOS prevent access to GPS data in such a case. However, since the TikTok-App sends background messages every hour, it is still possible for ByteDance to perform an imprecise location of the smartphone via the IP address.

Evidence

The following HTTP request was recorded on an Android smartphone (Samsung A13) while the TikTok-App was running in the background. The request is sent from the smartphone to the ByteDance Backend every hour.

```
1 POST /tiktok/location/info/?sdk_version=2.3.0-rc.7.2-bugfix&iid=7207781883278345989&device_id=7205879119363229190&ac=wifi&channel=googleplay&aid=1233&app_name=musical_ly&version_code=280303&version_name=28.3.3&device_platform=android&ab_version=28.3.3&ssmix=a&device_type=SM-A137F&device_brand=samsung&language=en&os_api=31&os_version=12&openudid=3feac993ea7b747b&manifest_version_code=2022803030&resolution=1080*2208&dpi=450&update_version_code=2022803030&rticket=1678280923393&current_region=GB&app_type=normal&sys_region=GB&timezone_name=Europe%2FZurich&residence=GB&app_language=en&ac2=wifi5g&uoo=0&op_region=GB&timezone_offset=3600&build_number=28.3.3&host_abi=armeabi-v7a&locale=en&region=GB&content_language=en%2C&ts=1678280924&cdid=68b2314c-a9e8-4070-a461-eaf413614ad8
2 Host: api16-normal-useast1a.tiktokv.com
3 [...]
4
5 {
6   "upload_source": "bdlocation_background_switch",
7   "status": {
8     "device_type": 2,
9     "is_strict_restricted_mode": false,
10    "system_language": "en",
11    "locale": "en_GB",
12    "location_mode": 1,
13    "mcc_mnc": "",
14    "permission": 1,
15    "system_region": "GB",
16    "restricted_mode": 2
17  },
18  "timestamp": 1678280923,
19  "is_vpn": false,
20  "is_proxy": true
21 }
```

Preconditions

The app must run in the background, which requires that users do not terminate it completely. However, to terminate the app completely, steps must be performed that go beyond exiting the

application. Therefore, it can be assumed that the TikTok-App is running in the background at least some of the time for most of its users.

Impact

Both the operator of the infrastructure (*Akamai Technologies*) and the receiver of the information (ByteDance) know the approximate location of the end device via the IP address. This information can be used in conjunction with other data to create a movement profile of a person.

Recommendations

It is recommended to terminate the TikTok-App completely when it is not in use.

Finding NTCF-188 M (Determining the device status): The TikTok-App for iOS detects information about the operating environment, which allows conclusions to be drawn whether the app is in a *test-bench* mode. The TikTok-App and backend services might behave differently than usual in such a case. iOS FI04 [20230309]

Background

The TikTok-App for iOS collects information about the operating environment. These can allow conclusions about whether the app is running on a test device. Such methods can be used by manufacturers to adapt the program flow to its environment, e.g. by suppressing behavior that should be kept secret.

Evidence

The TikTok-App checks for the presence of `cydia`, an unofficial app store. Cydia can only be run on jailbroken devices, which are often used by security researchers. The following excerpt shows the default app path checked by the app:

```
1 fileExistsAtPath: /Applications/Cydia.app
```

The above output was created using the following Frida script at app runtime:

```
1 /*
2 $ frida -U -f com.zhiliaoapp.musically -l tiktok.js
3 */
4 Interceptor.attach(ObjC.classes.NSFileManager["-fileExistsAtPath:"].implementation, {
5   onEnter: function (args) {
6     console.log('fileExistsAtPath: ', ObjC.Object(args[2]).toString());
7   },
8   onLeave: function (retval) {
9   }
10 });
```

If the TikTok-App detects a jailbreak, e.g. with a check for `cydia`, the app also sends this information encrypted to a ByteDance Backend. The following excerpt shows the transmitted content with the value pair `"JBDevice":true` before encryption:

```

1  {
2    "cell": {
3      "mcc": "",
4      "mnc": "",
5      "ra": ""
6    },
7    "shortbundleversion": "28.4.0",
8    "bundlename": "TikTok",
9    "timepassedsinclastlaunch": "195",
10   "timestamp": "1678365760.423975",
11   "uid": "1678262062308-3965637",
12   "platform": "iPhone8,4",
13   "fb_anon_id": "XZ64E1EB19-1CE7-4EAD-9DE3-DD2C1641BAA9",
14   "counter": "14",
15   "prev_session_dur": 191,
16   "reinstallCounter": "3",
17   "advertiserId": "9CFE9D17-82B9-42F0-8E68-67F83D99A1CF",
18   "systemversion": "14.3",
19   "iaecounter": "2",
20   "lang_code": "en",
21   "JBDevice": true,
22   "date3": "2023-03-08_085613+0100",
23   "deviceData": {
24     "cpu_type": "ARM64_V8",
25     "cpu_speed": "-1",
26     "cpu_64bits": "true",
27     "dim": {
28       "y_px": 1136,
29       "x_px": 640
30     },
31     "osVersion": "14.3_(Build_18C66)",
32     "ram_size": "2013",
33     "device_model": "iPhone8,4",
34     "cpu_count": "2"
35   },
36   "currentCountrycode": "CH",
37   "open_referrer": "",
38   "sc_o": "fu",
39   "date1": "2023-03-08_085422+0100",
40   "systemname": "iOS",
41   "ivc": false,
42   "localizedmodel": "iPhone",
43   "af_events_api": "1",
44   "bundleversion": "284022",
45   "eventName": "Launched",
46   "model": "iPhone",
47   "dev_key": "XY8Lpakui8g4kBcposRgxA",
48   "currentLanguage": "en-CH",
49   "wifi": true,
50   "advertiserIdEnabled": true,
51   [...],
52   "date1_2": "2023-03-08_085422+0100",
53   "disk": "5275/15238",
54   "sessioncounter": "17",
55   "date2": "2023-03-09_134239+0100",
56   "firstLaunchDate": "2023-03-08_085613+0100",
57   "originalAppsflyerId": "1677740633160-3499600",
58   "att_status": 0,
59   "platformextension": "ios_native",
60   "bundleIdentifier": "com.zhiliaoapp.musically"
61 }

```

The encryption of this content takes place using `kCCAlgorithmAES128` and can also be hooked and output using a Frida script. The encrypted content is sent to the web address

`log22-normal-useast1a.tiktokv.com`.

Other indicators of a test environment include the use of a proxy or a VPN, which allow network traffic to be examined. This information is also queried by the TikTok-App and transmitted to a ByteDance Backend:

```
1 POST /tiktok/location/info/?app_id=[...] HTTP/2
2 Host: api16-normal-useast1a.tiktokv.com
3 [...]
4 {
5   "status" : {
6     "locale" : "en-CH",
7     "restricted_mode" : 2,
8     "permission" : 60,
9     "carrier_region" : "",
10    "system_region" : "CH",
11    "sim_mccmnc" : {
12      "network" : "(null)(null)",
13      "primary" : "(null)(null)",
14      "secondary" : "(null)(null)"
15    },
16    "system_language" : "en-CH",
17    "network_sim_region" : "",
18    "location_mode" : 1,
19    "device_type" : 1
20  },
21  "is_vpn" : false,
22  "is_proxy" : true,
23  "timestamp" : 1678112554
24 }
```

In the time available, it was not possible to test whether the TikTok-App for Android detects information about root privileges or not. The above message to a ByteDance Backend about whether a VPN or proxy is used was also detected in the network communication of the Android app.

Preconditions

The query which detects Cydia or other files typical of a jailbreak is allowed to all apps.

Impact

It is unclear what the information above is used for by ByteDance. Therefore, the assumption that this is the detection of a test bench cannot be ruled out. If this is the case, such *test-bench* mode could hide certain behavior of the TikTok-App.

Recommendations

It is recommended to perform exhaustive investigations to be able to exclude different behavior of the TikTok-App during *test-bench* conditions.

Finding NTCF-189 L (Using an integrated browser): The TikTok-App for Android contains an integrated browser. This could be used to monitor and manipulate content and user inputs within that browser. As far as observed, the browser is only used in a limited scenario. The potential risk is therefore classified as low. Android FI05 [20230327]

Background

The TikTok-App for Android uses an integrated browser to display web content when a hyperlink shown on a users' profile is clicked ³. The hyperlink is followed via query to a ByteDance Backend. After loading the page, the option to open the page in an external browser is offered. Hyperlinks can only be opened from the user profile. Hyperlinks in messages to and from friends, in comments on videos, or in video descriptions are not clickable and can therefore not be opened in the integrated browser.

Evidence

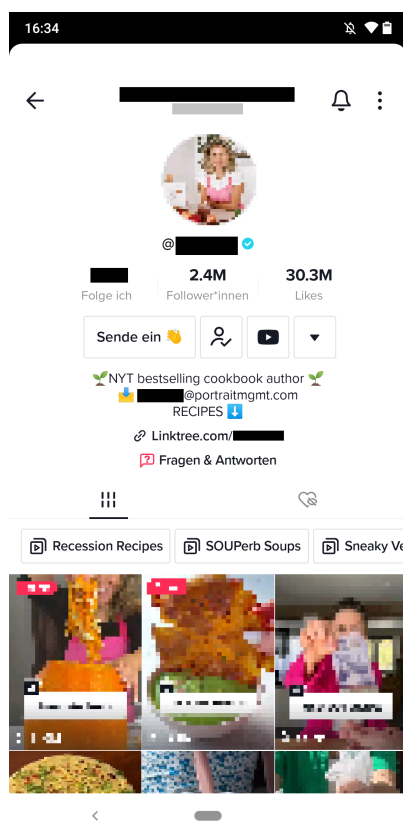


Figure 2: Link in Profile (Center part of figure)

The following HTTP request from an Android device shows the submission of a link followed to the ByteDance Backend:

³ On iOS this browser is not integrated

```
1 GET /link/?aid=1233&lang=de&scene=bio_url&target=Linktree.com\%2FXXXXXXXX&owner_suid=MS[...]7s
  HTTP/2
2 Host: web-va.tiktok.com
3 [...]
```

The server's response to this is:

```
1 HTTP/2 302 Found
2 Server: nginx
3 Content-Type: text/html; charset=utf-8
4 Content-Length: 52
5 Location: https://Linktree.com/XXXXXXXX
6 [...]
7
8 <a href="https://Linktree.com/XXXXXXXX">Found</a>.
```

Preconditions

A link in a user's profile is required.

Impact

It can be assumed that the profile page is primarily visited when a new profile is discovered and hardly ever after that. There is also only a single link per profile. Therefore, the damage potential is estimated to be low. For this reason, it was not investigated whether the browser is equipped with additional functions such as the transmission of user inputs to ByteDance.

Within the scope of the analysis, it was not possible to store a link in one's own test profile. This feature seems to be enabled only for selected profiles ⁴. Therefore, it was not possible to test whether arbitrary links can be shown in profiles.

It should be noted that the TikTok-App communicates with a ByteDance Backend when a link is followed, and ByteDance therefore has knowledge of a users visit of that page.

Recommendations

It is recommended not to enter any data on web pages that were opened from the TikTok-App. As a preventive measure, pages should also be opened in an external browser as early as possible.

⁴ <https://linktree.blog/how-to-add-a-linktree-to-your-tiktok-bio/>

Finding NTCF-190 M (Multi-factor authentication is not enforced): TikTok gives users the option to register with an email address and password or using a phone number. If one of these factors is compromised, control over the TikTok account can be taken. To protect against such attacks, TikTok optionally offers the option to enable multi-factor authentication. FI06

[20230327]

Background

ByteDance wants to make the registration and login process for users as simple as possible. For this reason, ByteDance does not use rigorous identity verification and, by default, more complex MFA factors. There is the possibility for attackers to use *SIM swapping* to hijack TikTok users' phone numbers and gain access to their accounts.

Enabling TikTok's optional "*2-step verification*" (wording TikTok) protects against this type of attack.

Evidence

TikTok only requires an email address and password or phone number when registering an account. Registration via other social media is also possible, but was not looked into in this analysis.

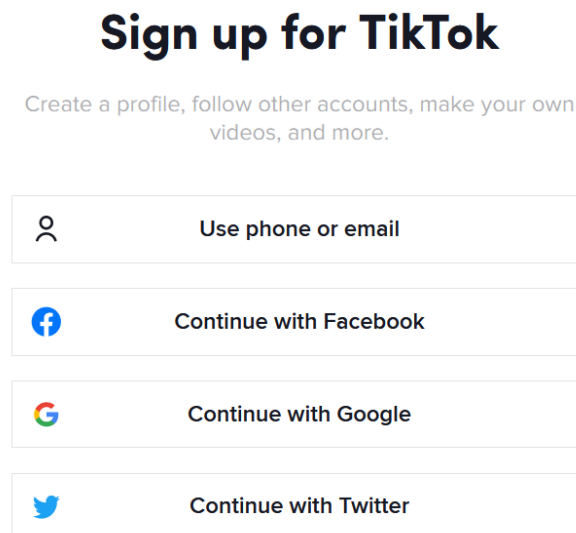


Figure 3: Registration options with TikTok

When a phone number is linked to an account, TikTok sends a text message with a six-digit code to the stored number upon login. This code is sufficient to log in with the linked account.

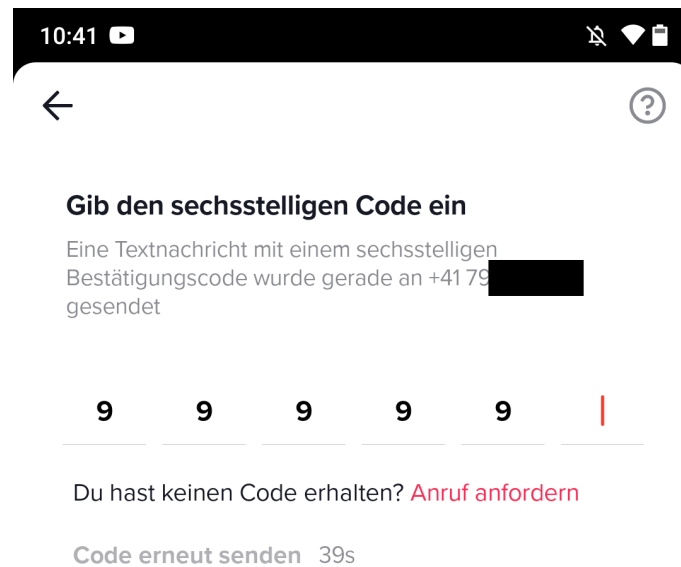


Figure 4: Login via SMS in der TikTok-App

Preconditions

If only one phone number is registered on the account and no 2-step verification is activated, it is possible to take over the account via SIM swapping. In this case, attackers take complete control over the phone number of the person. This is often achieved, via social engineering attacks on the mobile network provider of the person concerned.

Impact

If 2-step verification is not explicitly enabled, there is an increased risk of account takeover via SIM swapping. This risk applies not only to TikTok, but to all apps that only require the receipt of a text message.

One possible consequence of a hostile account takeover is the mass distribution of false messages within a short period of time. If accounts used by the authorities are affected, the effect is likely to be more severe. It is also conceivable that bots from other social media platforms pick up the false information and share it en masse in their own media, possibly making it accessible to a wider audience.

It must be mentioned that 2-step verification is enforced by TikTok, for all verified accounts⁵.

⁵ <https://support.tiktok.com/en/using-tiktok/growing-your-audience/how-to-tell-if-an-account-is-verified-on-tiktok#4>

Recommendations

It is recommended that all users enable the option for 2-step verification using an authenticator app in the account settings, see. [TikTok Help: Enable 2-Step Verification](#).

In addition, the use of strong and unique passwords is generally recommended.

4 Testcases

This section presents all testcases that were considered during the security analysis. Findings resulting from a specific testcase are linked under the short description of the testcase. If no finding is linked, no relevant vulnerability was found in the time frame of the analysis. If the testcases only apply to a subset of the components, the corresponding components are explicitly listed.

4.1 Network Communication

The communication testcases focus on the data exchanged between the TikTok-App and the ByteDance Backend, based on an expected use of the TikTok-App by legitimate users without malicious intent.

TC 1 Encrypted data transmission

Is the transmitted network traffic encrypted?

Risk: If the network traffic is not encrypted, it is easy for attackers to read or manipulate the content of the communication.

Findings: [183](#), [184](#)

TC 2 Encrypted data transmission with secure protocols

Is network traffic encrypted using secure protocols?

Risk: When network traffic is transmitted using insecure protocols, it is easier for attackers to read and manipulate the content of the communication.

TC 3 Create user account and initial identification

What account registration options are offered? How is the identity of the users verified?

Risk: If the identity of users is not sufficiently verified, it is possible for attackers to create accounts under someone else's name or take control over others' accounts. This allows sensitive data to be viewed or false information to be spread.

Findings: [190](#)

TC 4 Create and upload video

What data is collected and transmitted when a video is recorded and uploaded in the TikTok-App?

Risk: If more or different data is collected and transmitted to an ByteDance Backend than intended, it can be an indication of surveillance.

Findings: [186](#), [187](#)

TC 5 Exchange of private messages

Are the private messages end-to-end encrypted?

Risk: If private messages are not end-to-end encrypted, it is possible for third parties like the operators of the ByteDance Backend to read and modify the messages.

Findings: [184](#)

TC 6 TikTok-App activity in the foreground

While the TikTok-App is running in the foreground, it is possible to collect data with permissions already granted, such as for precise GPS location. Is this capability overused by sending data to a ByteDance Backend?

Risk: If data is unexpectedly sent to an ByteDance Backend without any apparent benefit to users, this can be an indication of surveillance.

Findings: [186](#)

TC 7 TikTok-App activity in the background

Is unexpected data sent to a ByteDance Backend while the TikTok-App is active in the background?

Risk: Unexpected data transmission in the background can be an indication of surveillance.

Findings: [187](#)

4.2 Privacy and Data Protection

In the privacy and data protection testcases listed below, particular focus was placed on how the TikTok-App handles the permissions of the mobile operating systems. Concretely, which permissions are requested and when, and whether the data received is transmitted to the ByteDance Backend.

TC 8 Geolocation tracking

Is the current geolocation of the device captured and transmitted to the ByteDance Backend? Is the GPS position required for the use of the TikTok-App?

Risk: Geolocation data can be used to create movement profiles. The more numerous and precise the data points are available, the more accurately users can be monitored. Movement profiles can enable conclusions to be drawn about place of residence, place of work, preferences, and habits, etc.

Findings: [186](#), [187](#)

TC 9 Contacts access

Is access to the contacts mandatory in order to use the TikTok-App? At what times are the contacts requested and transmitted?

Risk: Collecting contact information enables ByteDance to create user profiles and relationship patterns of uninvolved third parties - persons who do not use TikTok.

Findings: [182](#)

TC 10 Calendar access

Does the TikTok-App need access to the users' calendar?

Risk: Calendar information can contain sensitive information about the users' daily schedule and activities, and is therefore well suited for monitoring. Appointments can also contain valuable additional information, such as the persons participating, their contact details, locations, activities, etc.

TC 11 Camera access

At what times is the camera used? Is the data transferred directly to the ByteDance Backend?

Risk: The camera can be used to record images or videos unnoticed. This image information can provide information about the environment in which users are currently located, which persons are in the vicinity, or which activities are being carried out. Under certain circumstances, the images can also be used for blackmailing.

TC 12 Microphone access

At what times is the microphone used? Is the data transmitted directly to the ByteDance Backend?

Risk: The microphone can be used for unnoticed audio recordings. These recordings can provide information about the current environment in which the users are located and reveal the content of - possibly confidential - conversations. Under certain circumstances, the recordings can also be used for blackmailing.

TC 13 Access to external storage

When is the external storage accessed? Are only the selected files read or also other contents?

Component: Android

Risk: The TikTok-App can access and process locally stored data and transfer it to a ByteDance Backend without the knowledge and explicit consent of the user. Among other things, this data may include confidential and personal details.

TC 14 Local network access

Is access to the local network requested?

Component: iOS

Risk: The permission allows interaction with local network devices, such as smart home devices. Under certain circumstances, it could be possible to control such local network devices or read out sensitive information.

TC 15 Collect system information

Is information about the smartphone sent to a ByteDance Backend? Is information about installed or running apps collected and transmitted to the ByteDance Backend?

Risk: The transfer of system information enables the creation of a user profile and the surveillance of user activities. This is particularly important if users use multiple profiles (e.g. private

and professional) on the same device.

Findings: [185](#)

TC 16 Data extract according to Federal Act on Data Protection

If users exercise their right to information and request a complete data export, does the export contain user data that was collected without a legitimate purpose?

Risk: Data should only be collected and stored to the extent necessary for the purpose in question.

TC 17 Clipboard access

Does the TikTok-App access the clipboard without user interaction? Is the data from the clipboard automatically transferred to the ByteDance Backend?

Risk: The clipboard may contain confidential content such as passwords.

TC 18 Using an integrated browser

Does the app use an integrated browser? If so, what is its purpose? Does it have any unusual behavior?

Risk: An integrated browser can be extended by almost any functionality. This could, for example, enable the recording of displayed content or user input. This could capture potentially sensitive data and transmit it to a ByteDance Backend.

Findings: [189](#)