Anhang 1: Technischer Appendix

Inhaltsverzeichnis

A. KONFIGURATIONSOPTIONEN

Forensische Vergleiche einer im Jahr 2014 öffentlich gewordenen Version von FinSpy mit der A-Malware zeigen, dass die Quellcodes beider Malwares praktisch identisch sind, sodass es sich definitiv um verschiedene Versionen derselben Malware handelt, vgl. Malware A in Anhang 1 und FinSpy 2014 in Anhang 2. So stimmen die Konfigurationsoptionen der A-Malware sowie der 2014 öffentlich gewordenen Version von FinSpy nahezu überein,

> Anhang 2: FinSpy Malware von August 2014; für eine detailliertere Übersicht der Funktionen von FinSpy siehe https://citizenlab.ca/2012/08/the-smartphone-who-loved-me-finfisher-goes-mobile/.; https://www.symantec.com/security-center/writeup/2012-072615-4146-99?tabid=2; beide zuletzt abgerufen am 4. Juli 2019.

Dabei handelt es sich um jene Teile des Quellcodes, die maßgeblich bestimmen, wie genau die Malware arbeitet, beispielsweise welche Informationen des Benutzers des Endgeräts abgegriffen werden.

Dass es sich bei der A-Malware aus Anhang 1 um exakt jene Malware handelt, mittels derer im Jahr 2017 auf der türkischen Adalet-Website Oppositionspolitiker angegriffen wurden, kann leicht nachvollzogen werden. Dazu muss lediglich die Datei, die dieser Anzeige beigefügt ist, mit der Datei verglichen werden, die von der auf archive.org archivierten Adalet-Website abrufbar ist. Hierbei wird deutlich, dass beide Dateien dieselbe kryptographische Prüfsumme (sog. Hash) aufweisen. Eine solche Prüfsumme ist ein eindeutiger digitaler Fingerabdruck einer Datei, sodass es sich bei Übereinstimmung zweier Prüfsummen wie hier um dieselbe Datei handelt.

Doch die A-Malware ist nicht nur mit der von Forschern im August 2014 veröffentlichten FinSpy-Stichprobe praktisch identisch. Auch zwischen ihr und einer neueren Version aus dem Juli 2015 teilt sie mehr als 90% ihres Quellcodes. Abgesehen von eher kosmetischen Unterschieden – nämlich Änderungen, die zur Verschleierung des Herstellers führen sollten –, verwendet die A-Malware den gleichen Code wie frühere FinSpy-Samples. So ist beispielsweise der Code, der für die Aufzeichnung von Telefonaten verwendet wird, praktisch identisch, bis hin zur Verwendung des gleichen Musters für die Dateinamen der aufgezeichneten Daten („tmp460“ + Zeitstempel in Millisekunden + „.dat“). Dass zwei unabhängig voneinander entwickelte Überwachungsprogramme rein zufällig exakt dieselbe Namenskonvention verwenden ist eine rein theoretische Möglichkeit und kann ausgeschlossen werden.

```
new File(this.getContext().getFilesDir(), "cLogFile").createNewFile();
v29 = Long.toHexString(System.currentTimeMillis());
v17 = new File(this.getContext().getFilesDir(), "tmp460" + v29 + ".dat");
v30 = new FileOutputStream(v17).getChannel();
v30.write(v10.toArray(new ByteBuffer[v10.size()]));
v30.close();
v17.renameTo(new File(this.getContext().getFilesDir(), "460" + v29 + ".rd"));
```

```
v3_2 = Long.toHexString(System.currentTimeMillis());
v4_2 = new File(org.customer.fu.a.d.getFilesDir(), "tmp460" + v3_2 + ".dat");
v5_1 = new FileOutputStream(v4_2).getChannel();
v5_1.write(((ByteBuffer[])v2_4));
v5_1.close();
v4_2.renameTo(new File(org.customer.fu.a.d.getFilesDir(), "460" + v3_2 +
".r1d"));
org.customer.fu.a.l = v13;
org.customer.fu.a.f();
this.getClass().getSimpleName();
new StringBuilder("id ").append(Thread.currentThread().getId()).append("
RecordedFilesCallLogs 460").append(v3_2).append(".rd Size ").append(new
File(org.customer.fu.a.d.getFilesDir(), "460" + v3_2 +
".r1d").length()).toString();
```

Links: Code der FinSpy Malware aus dem Jahre 2014; Rechts: Code der A-Malware

B. TEXTLICHE HINWEISE

Im Code der A-Malware sind diverse Wörter in deutscher Sprache auffindbar. Diese treten vermehrt in Einstellungs-Dateien mit dem Namen „einstellung.xml“ auf.

Anlage 1: Sample der A-Malware; Access-Now-Bericht, S. 9.

Das spricht für eine Entwicklung durch einen deutschen Hersteller und jedenfalls gegen eine Eigenentwicklung durch türkische Stellen.

Darüber hinaus gibt es tief in den Code eingebettete Hinweise, die auf den ursprünglichen Titel der A-Malware hinweisen, wie z.B. der Text „FIN_GIFT“.

```
new StringBuilder(“id_”).append(Thread.currentThread().getId()).append(“
FIN_GIFT CheckRootFunctionality_Root_fG”).toString();
```

Auszug aus dem Code der A-Malware, der “FIN_GIFT” enthält.

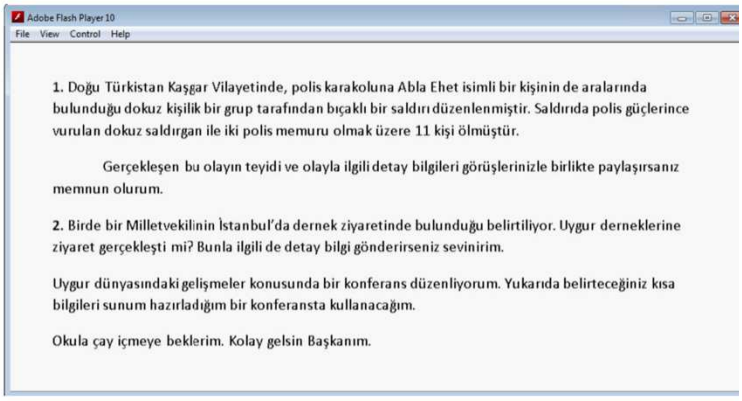
Beide Indizien zusammengenommen verweisen eindeutig auf den in Deutschland ansässigen Hersteller FinFisher.

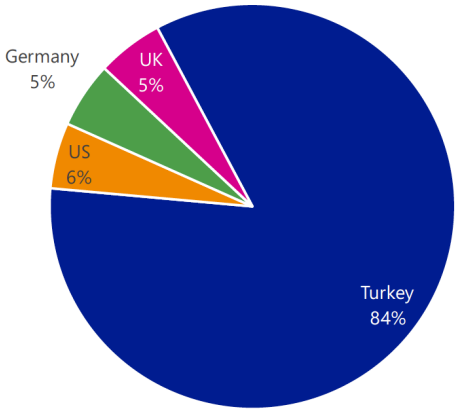
C. MICROSOFT SECURITY REPORT

Aus dem Microsoft Security Intelligence Report für Januar bis Juni 2016 (Band 21) ergeben sich weitere Indizien für einen türkischen Kauf von FinSpy.

Im Dezember 2016 vermeldete Microsoft das Auftauchen eines Zero-Day-Exploits, also einer bisher unbekannten herstellerseitigen Sicherheitslücke im Windows-Betriebssystem. Angreifer nutzten hierbei den Adobe Flash Player, um die Windows-Sicherheitsarchitektur zu kompromittieren. Diese Sicherheitslücke wurde ausgenutzt, um eine Malware zu installieren, die Microsoft als FinSpy identifizierte. Microsoft benutzte hierbei das der Firma eigene Namensschema und nutzte „Neomydium“ als Bezeichnung für FinFisher und „Wingbird“ für Finspy.

Darüber hinaus erklärte Microsoft, dass von der Sicherheitslücke Dutzende Opfer betroffen waren; die weit überwiegende Mehrheit hiervon befand sich in der Türkei. Dass die Türkei als vorrangiges Ziel des Angriffs auserkoren war, schlussfolgerte Microsoft außerdem daraus, dass die Malware über Websites und Tweets in türkischer Sprache als Köder verbreitet wurde.

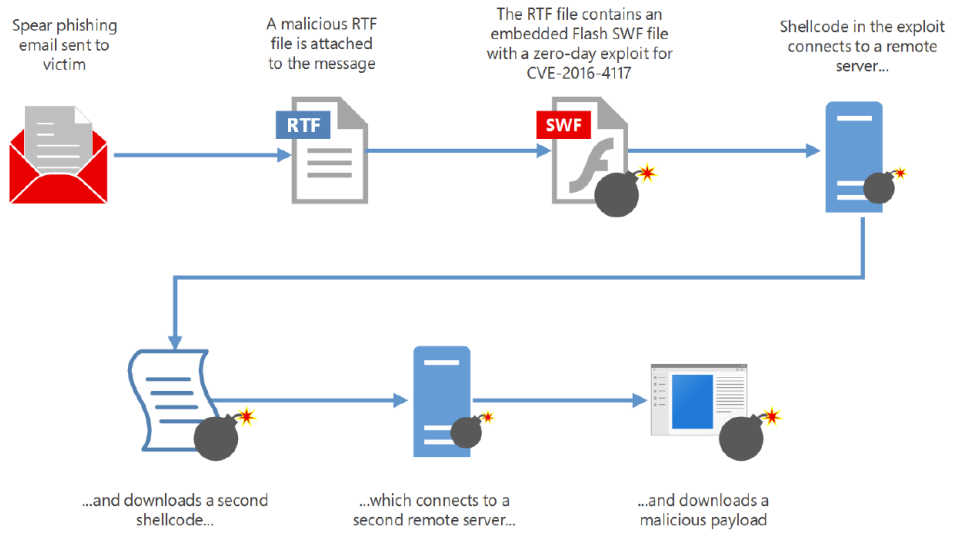




Links: Die türkischsprachige Ködernachricht, mit der FinSpy laut Microsoft in der Türkei verbreitet wurde. Rechts: FinSpy-Opfer, durch Microsoft nach Ländern eingeteilt

Darüber hinaus bestätigen die Ergebnisse von Microsoft die unter 1. und 2. dargestellten forensischen Software-Analyse. Die von Microsoft identifizierte Version von FinSpy zeigt im Vergleich zur A-Malware ein zum Verwechseln ähnliches Verhalten, einschließlich der Verwendung desselben Domain-Dienstleisters.

Microsoft Security Intelligence Bericht, Volume 21, January through June, 2016, S. 22 ff.; Access-Now-Bericht, S.10



Darstellung der Funktionalität der von Microsoft identifizierten FinSpy-Version.

D. WEITERE FINSPY-MALWARE IN DER TÜRKEI

Die hier als A-Malware bezeichnete FinSpy-Version war jedoch nicht die einzige, die in der Türkei eingesetzt wurde. Vielmehr wurde am 21. Juli 2017 auf die Website „VirusTotal“, einen Internet-Dienst zur Identifikation und Archivierung von Schadsoftware, eine Datei hochgeladen, die im Folgenden als B-Malware bezeichnet wird. Sie kann bis heute unter dem folgenden Link abgerufen werden:

> https://www.virustotal.com/gui/file/23f154723213452634abe6063fd07bd3a38700a6b0ba4117db3224ae1411dada/detection; zuletzt abgerufen am 4. Juli 2019.

Die Datei ist mittels des SHA-256-Hashes „23f154723213452634abe6063fd07bd3a38700a6b0ba4117db3224ae1411dada“ eindeutig zu identifizieren.

Sie wurde von VirusTotal als „FinSpy“ bzw. als „Belesak“ erkannt. Letzteres ist eine alternative, unter Antivirus-Experten gängige Bezeichnung für FinSpy. Da die B-Malware intern dieselben Zeichenketten verwendet wie die A-Malware muss sie zur gleichen Malware-Familie gehören. So bezeichnen beide eine Programmkomponente als org.customer.fu.S5tartVers10n und verwenden außerdem den Paketnamen org.tech.fu. Mithin erscheint es höchstwahrscheinlich, dass auch die B-Malware von FinFisher hergestellt wurde.

Die Virus-Identifikationsmechanismen von Virus Total sind höchst zuverlässig. Das von Google angebotene VirusTotal nutzt die sogenannte „Yara binary identification“, einen anerkannten Industriestandard. Dabei werden in der ausführbaren Datei bestimmte charakteristische Merkmale – beispielsweise Zeichenketten – gesucht und verglichen. Wenn also VirusTotal die hochgeladene Datei als FinSpy identifiziert, so muss gemäß Industriestandard davon ausgegangen werden. An der B-Malware kann darüber hinaus abgelesen werden, dass die türkischen FinFisher-Kunden bis in den Juli 2017 hinein Zugriff auf FinSpy hatten, FinFisher die hauseigene Malware also bis dahin ausführte.

Denn die digitale Signatur der B-Malware belegt, dass die Datei erst am 18. Juli 2017 unterschrieben wurde. Da die Signatur seitens des in München ansässigen Herstellers angebracht wird, wie in den nächsten Abschnitten zu zeigen sein wird, kann die Software unmöglich vor diesem Datum exportiert worden sein.

```
Issuer: CN=e, OU=e, O=e, L=e, ST=e, C=e
Serial number: 5257eb4f
Valid from: Tue Jul 18 14:01:19 CEST 2017 until: Sat Dec 03 13:01:19 CET
2044
Certificate fingerprints:
       SHA1: 35:D6:63:83:05:EB:5E:46:FB:FF:BE:17:AA:6A:27:3B:E9:9B:A6:3F
       SHA256:
EE:7B:3C:44:DB:67:5C:03:B3:FA:A2:18:93:27:69:63:FD:02:F9:9C:BA:D7:97:2A:FD:
BE:0C:FA:1A:50:27:3D
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 2048-bit RSA key
Version: 3
```

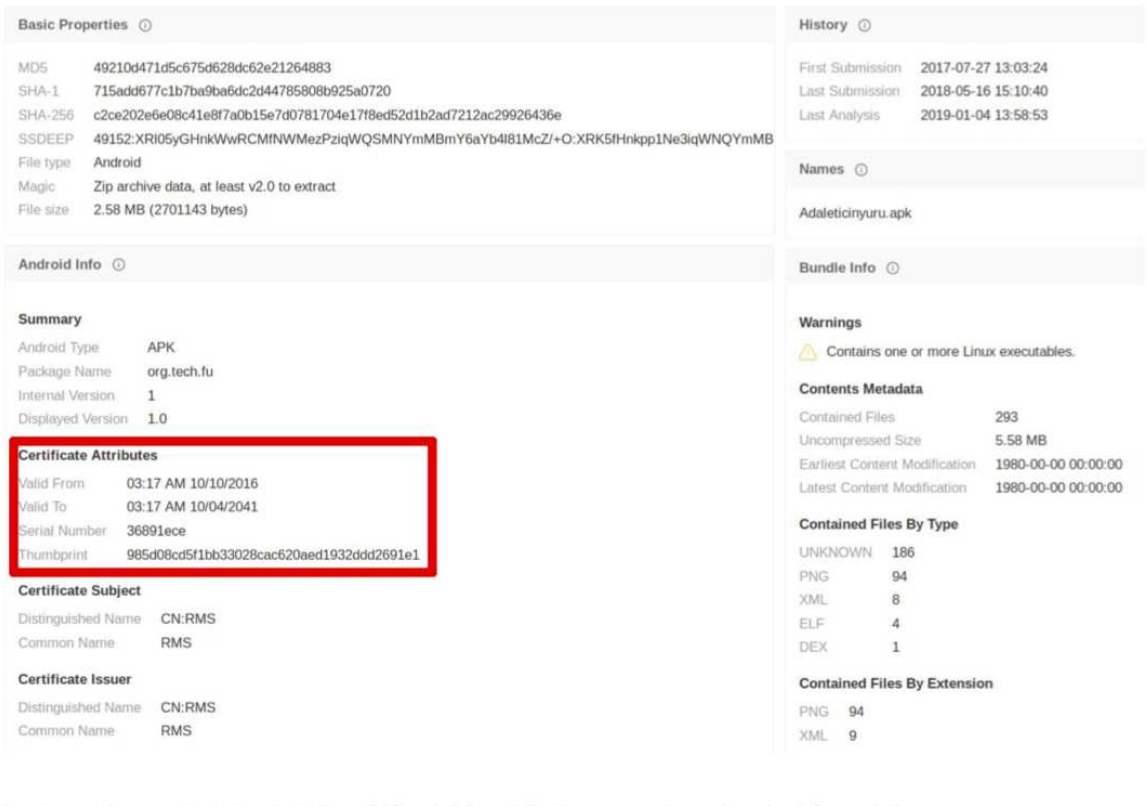
In dieser digitalen Signatur der B-Malware wird das Erstellungsdatum der Unterschrift in der dritten Zeile als 18. Juli 2017 angegeben.Alle analytischen Schritte können anhand einer Analyse der B-Malware, die dieser Anzeige angehängt ist, nachvollzogen werden,

Anhang 3: B-Malware vom 21. Juli 2017.

E. FINSPY-SAMPLE IN LYBIEN

Der Vergleich zweier FinSpy-Versionen, die mit demselben Zertifikat digital signiert wurden, beweist, dass diese digitalen Unterschriften tatsächlich bereits vom Hersteller angebracht werden.

Zunächst belegt eine Analyse der Metadaten und Software-Eigenschaften jener Malware, die aus Libyen auf die VirusTotal-Website hochgeladen wurde, dass es sich auch hierbei um FinSpy handeln muss. Denn die libysche Malware wurde mit demselben kryptographischen Schlüssel und demselben Zertifikat wie die A-Malware unterschrieben.



Basic Properties

MD5	49210d471d5c675d628dc62e21264883
SHA-1	715add677c1b7ba9ba6dc2d44785808b925a0720
SHA-256	c2ce202e6e08c41e8f7a0b15e7d0781704e17f8ed52d1b2ad7212ac29926436e
SSDEEP	49152:XRI05yGHnkWwRCMfNWMezPziqWQSMNYmMBmY6aYb4l81McZ/+O:XRK5fHnkpp1Ne3iqWNQYmMB
File type	Android
Magic	Zip archive data, at least v2.0 to extract
File size	2.58 MB (2701143 bytes)

History

First Submission	2017-07-27 13:03:24
Last Submission	2018-05-16 15:10:40
Last Analysis	2019-01-04 13:58:53

Names

Adaleticinyuru.apk

Android Info

Summary

Android Type	APK
Package Name	org.tech.fu
Internal Version	1
Displayed Version	1.0

Certificate Attributes

Valid From	03:17 AM 10/10/2016
Valid To	03:17 AM 10/04/2041
Serial Number	36891ece
Thumbprint	985d08cd5f1bb33028cac620aed1932ddd2691e1

Certificate Subject

Distinguished Name	CN:RMS
Common Name	RMS

Certificate Issuer

Distinguished Name	CN:RMS
Common Name	RMS

Bundle Info

Warnings

Contains one or more Linux executables.

Contents Metadata

Contained Files	293
Uncompressed Size	5.58 MB
Earliest Content Modification	1980-00-00 00:00:00
Latest Content Modification	1980-00-00 00:00:00

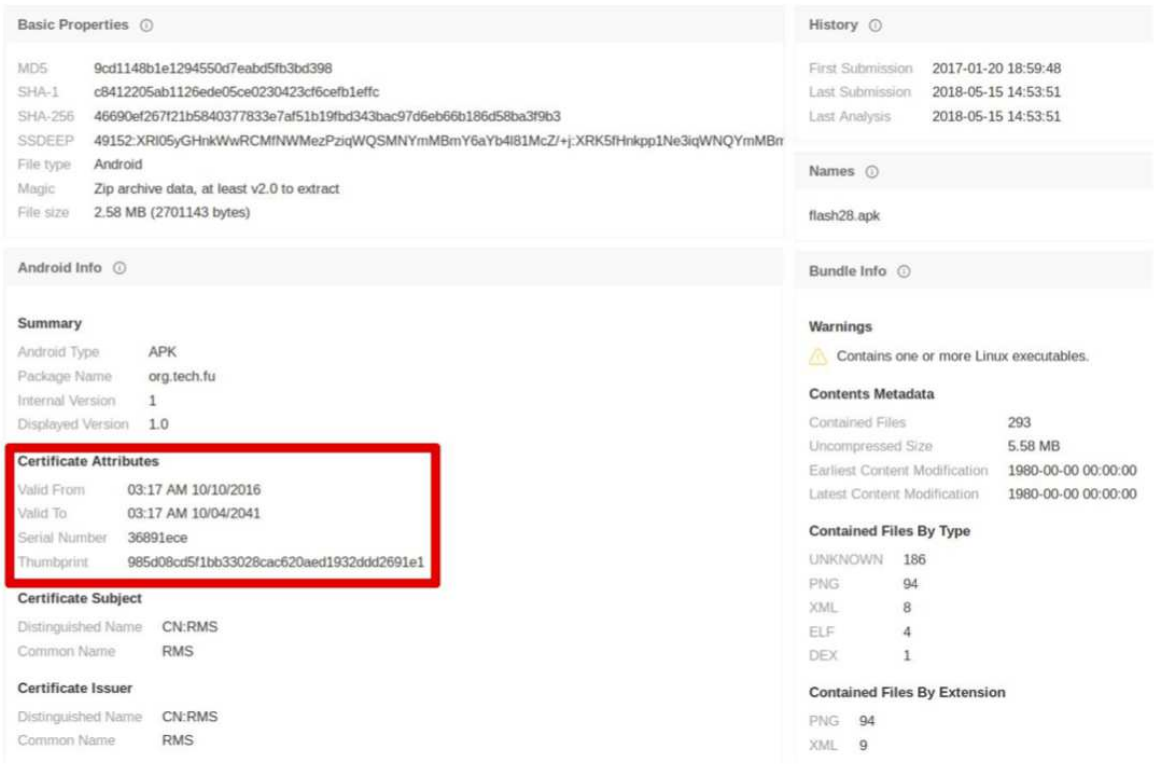
Contained Files By Type

UNKNOWN	186
PNG	94
XML	8
ELF	4
DEX	1

Contained Files By Extension

PNG	94
XML	9

Metadaten der A-Malware



Basic Properties

MD5	9cd1148b1e1294550d7eabd5fb3bd398
SHA-1	c8412205ab1126ede05ce0230423cf6cefb1effc
SHA-256	46690ef267f21b5840377833e7af51b19fbd343bac97d6eb66b186d58ba3f9b3
SSDEEP	49152:XRl05yGHnkWwRCMfNWMezPziqWQSMNYmMBmY6aYb4l81McZ/+j:XRK5fHnkpp1Ne3iqWNQYmMBn
File type	Android
Magic	Zip archive data, at least v2.0 to extract
File size	2.58 MB (2701143 bytes)

History

First Submission	2017-01-20 18:59:48
Last Submission	2018-05-15 14:53:51
Last Analysis	2018-05-15 14:53:51

Names

flash28.apk

Android Info

Summary

Android Type	APK
Package Name	org.tech.fu
Internal Version	1
Displayed Version	1.0

Certificate Attributes

Valid From	03:17 AM 10/10/2016
Valid To	03:17 AM 10/04/2041
Serial Number	36891ece
Thumbprint	985d08cd5f1bb33028cac620aed1932ddd2691e1

Certificate Subject

Distinguished Name	CN:RMS
Common Name	RMS

Certificate Issuer

Distinguished Name	CN:RMS
Common Name	RMS

Bundle Info

Warnings

⚠ Contains one or more Linux executables.

Contents Metadata

Contained Files	293
Uncompressed Size	5.58 MB
Earliest Content Modification	1980-00-00 00:00:00
Latest Content Modification	1980-00-00 00:00:00

Contained Files By Type

UNKNOWN	186
PNG	94
XML	8
ELF	4
DEX	1

Contained Files By Extension

PNG	94
XML	9

Metadaten der libyschen Malware. Es wird deutlich: Die Metadaten sind identisch.

Beide Dateien teilen das gleiche Zertifikat, das gleiche Erstellungsdatum und den gleichen Seriennamen. Die Verwendung desselben Zertifikats zum Signieren von Software, die mit zwei verschiedenen Kommando-Servern kommunizieren sollen und in zwei verschiedenen Ländern verwendet wurden, belegt, dass diese Schlüssel von den ursprünglichen Entwicklern – also FinFisher – und nicht von den Endkunden oder Betreibern digital signiert wurden.

F. ZEITPUNKT DER ÜBERMITTLUNG

Die forensische Analyse der A-Malware zeigt außerdem, dass sie nach dem 1. Januar 2015, als die entsprechende Ergänzung der Dual Use Verordnung für Intrusionssoftware in Kraft trat, ausgeführt sein worden muss. So weisen diverse Eigenschaften der A-Malware "Adaleticinyuru.apk" auf September und Oktober 2016 als Erstellungszeitpunkte hin.

Der erste Beleg ist in der Datei „build-data.properties“ enthalten, die durch einfaches Entpacken der ursprünglichen APK-Datei (die im Wesentlichen nur ein Zip-Archiv ist) überprüft werden kann. Diese Datei enthält Metadaten zur Erstellung einer Bibliothek namens „GMSCore“, die in der A-Malware enthalten ist. Aus diesen Metadaten ergibt sich, dass zur Herstellung von „GSMCore“ das System „Blaze“ eingesetzt wurde. Die verwendete Version des „Blaze“-Systems wiederum wurde erst am 9. Juli 2016 veröffentlicht:

```
build.tool=Blaze, release blaze-2016.07.09-3 (mainline @126938038)
```

Daher kann die Komponente „GSMCore“ in der A-Malware niemals vor diesem Tag erstellt worden sein, mithin auch nicht die A-Malware selbst.

Zweitens ist in diesen Daten das Datum enthalten, an dem die Version von „GSMCore“ in der A-Malware kreiert wurde, nämlich der 23. September 2016:

```
build.time=Fri Sep 23 14\:39\:54 2016 (1474666794)
```

Da „GMSCore“ einen integralen Funktionsbestandteil der Malware darstellt, folgt daraus, dass die FinSpy-Version der A-Malware nicht vor dem 23. September 2016 erstellt und mithin auch nicht ausgeführt worden sein kann.

Dasselbe ergibt sich auch aus einer anderen Datei. Denn in dem Dateibestandteil “META-

```
Manifest-Version: 1.0
Built-By: Generated-by-ADT
Created-By: Android Gradle 2.2.1
```

INF/MANIFEST.MF” wird auf die Software „Android Gradle Version 2.2.1.“ verwiesen. Android Gradle ist eines der Software-Werkzeuge, die Programmierer beim Entwickeln von Android-Programmen verwenden. Die Android-Gradle-Version 2.2.1 wurde jedoch erst im September 2016 veröffentlicht.

> https://developer.android.com/studio/releases/gradle-plugin; zuletzt abgerufen am 5. Mai 2019.

Vor Veröffentlichung der Android-Gradle-Version 2.2.1 war eine Erstellung und Ausfuhr einer Software, die – wie die A-Malware – auf diese Version verweist, naturgemäß nicht möglich.

Außerdem beinhaltet der “AndroidManifest.xml” Dateibestandteil die folgenden Metadaten:

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
android:versionCode="1" android:versionName="1.0" package="org.tech.fu"
platformBuildVersionCode="24" platformBuildVersionName="7">
```

Hieraus geht hervor, dass die A-Malware mittels der Version 24 des Android Entwicklungssystems übersetzt wurde. Version 24 bezieht sich auf Android 7.0 „Nougat“, das auch erst im September 2016 veröffentlicht wurde.

Darüber hinaus wurde auch die digitale Signatur der A-Malware ausweislich der hierin enthaltenen Informationen erst am 10. Oktober 2016 erstellt:

```
Owner: CN=RMS
Issuer: CN=RMS
Serial number: 36891ece
Valid from: Mon Oct 10 05:17:01 CEST 2016 until: Fri Oct 04 05:17:01 CEST
2041
Certificate fingerprints:
       SHA1: 98:5D:08:CD:5F:1B:B3:30:28:CA:C6:20:AE:D1:93:2D:DD:26:91:E1
       SHA256:
1E:62:1A:88:3B:CD:9D:1B:D6:D5:61:11:C4:88:EE:10:D4:67:1D:2C:A6:64:F7:27:FE:
72:59:47:8A:68:79:67
Signature algorithm name: SHA256withRSA
Subject Public Key Algorithm: 2048-bit RSA key
Version: 3
```

All die oben genannten Elemente weisen also sämtlich darauf hin, dass die A-Malware nicht vor September oder Oktober 2016 erstellt und damit auch nicht vor diesem Zeitpunkt ausgeführt worden sein kann.