Effiziente und erklärbare Erkennung mobiler Schadsoftware mittels maschineller Lernmethoden[1]

Daniel Arp[2]

Abstract: Die Verbreitung von Smartphones und Tablets hat in den vergangenen Jahren stark zugenommen. Aufgrund ihrer hohen Popularität haben sich diese Geräte jedoch zugleich auch zu einem lukrativen Ziel für Autoren von Schadsoftware entwickelt, weshalb mittlerweile täglich neue Schadprogramme gefunden werden, insbesondere für das Android-Betriebssystem. Durch das hohe Aufkommen bösartiger Applikationen bieten aktuelle Erkennungsmethoden wie Antivirenprogramme jedoch oft keinen ausreichenden Schutz, da sie meist signaturbasiert arbeiten und somit auf die Bereitstellung zeitnaher Updates für die Erkennung neuer Schadsoftware angewiesen sind.

In dieser Dissertation wird eine richtungsweisende Methode zur Erkennung von mobiler Schadsoftware vorgestellt, die eine effektive Erkennung direkt auf dem Mobilgerät ermöglicht. Hierfür werden Techniken des maschinellen Lernens und der statischen Code-Analyse derart kombiniert, dass selbst auf Mobilgeräten mit geringer Rechenleistung eine zuverlässige Erkennung von Schadsoftware innerhalb weniger Sekunden möglich wird. Im Vergleich zu anderen Methoden kann eine deutliche Steigerung der Erkennungsleistung erzielt werden und das Verfahren liefert darüber hinaus als erstes seiner Art interpretierbare Entscheidungen zurück.

1 Problembeschreibung

Smartphones und Tablets erfreuen sich nach wie vor steigender Beliebtheit, sodass diese Geräte mittlerweile von mehr als 2,5 Milliarden Menschen weltweit verwendet werden [Ne19]. Zu dieser rasanten Verbreitung hat nicht zuletzt Google's Betriebssystem *Android* erheblich beigetragen, welches mittlerweile auf mehr als 85% der Mobilgeräte eingesetzt wird. Durch die hohe Popularität dieses Betriebssystems haben sich diese Geräte allerdings zugleich auch zu einem lukrativen Angriffsziel für Autoren von Schadsoftware entwickelt. So konnte beispielsweise der Antivirenhersteller *GData* im Jahr 2018 mehr als 4 Millionen neuer Instanzen mobiler Schadsoftware identifizieren und bislang ist kein Rückgang dieses Trends zu beobachten [Bu19].

Obwohl für die Erkennung von Schadsoftware (auch *Malware* genannt) bereits verschiedene Lösungsansätze existieren, bieten diese häufig keinen ausreichenden Schutz. Ein Hauptnachteil aktueller Erkennungssoftware liegt etwa darin, dass sie Schadapplikationen meist nur anhand bestimmter Muster, sogenannter Signaturen, identifizieren, die zuvor manuell erstellt werden müssen und somit für neue Schadapplikationen nicht vorhanden sind [Wr17, MKK07]. Durch das hohe Aufkommen neuer Malware können passende Er-

[1] Englischer Titel der Dissertation: "Efficient and Explainable Detection of Mobile Malware with Machine Learning"

[2] Technische Universität Braunschweig, d.arp@tu-braunschweig.de

kennungssignaturen jedoch häufig nicht rechtzeitig bereitgestellt werden, sodass Mobilgeräte trotz installiertem Antivirenprogramm infiziert werden können.

Eine vielversprechende Forschungsrichtung, um auf die Bedrohung durch mobile Malware zu reagieren, stellt der Einsatz maschineller Lernmethoden dar. Diese Algorithmen erlauben die automatische Extraktion effektiver Erkennungsmuster aus großen Datenbeständen, bringen jedoch auch mehrere Herausforderungen mit sich, die den Einsatz dieser Techniken auf Mobilgeräten bislang verhindert haben. Zum einen benötigen diese Techniken erhebliche Rechenressourcen und sind damit normalerweise für den Einsatz direkt auf dem Mobilgerät nicht geeignet. Weiterhin sind die gelernten Modelle meist nicht interpretierbar, sodass Malware-Analysten und Smartphone-Benutzer die vom System getroffenen Entscheidungen nicht nachvollziehen können. Ferner sind zudem Angriffe auf diese Systeme denkbar, die ihre Erkennungsrate in der Praxis deutlich reduzieren können.

1.1 Beiträge der Dissertation

In dieser Dissertation wird ein richtungsweisendes Verfahren zur Erkennung von mobiler Schadsoftware vorgestellt, das die zuvor diskutierten Herausforderungen beim Einsatz maschineller Lernmethoden auf Mobilgeräten adressiert und somit als erste Methode eine effiziente, effektive und zugleich interpretierbare Erkennung von Schadsoftware direkt auf dem Mobilgerät ermöglicht. Die Dissertation kann dabei grob in drei unterschiedliche Schwerpunkte unterteilt werden [Ar19]:

1. *Ultraschallbasierte Schadsoftware.* Um die Notwendigkeit zur Entwicklung neuer Erkennungsansätze zu motivieren, werden zunächst die Ergebnisse einer Studie über eine fortgeschrittene Schadsoftware präsentiert, die mittels eines ultraschallbasierten Seitenkanals persönliche Daten von Smartphone-Nutzern ausspioniert. In dieser Studie konnten insgesamt 234 Applikationen mit der entsprechenden Funktionalität identifiziert werden [Ar17]. In Zusammenarbeit mit Google konnten allerdings alle betroffenen Applikationen aus dem *GooglePlay Store* entfernt werden, sodass diese kein Sicherheitsrisiko mehr darstellen. Im Rahmen der Studie werden die Schwächen signaturbasierter Ansätze anhand konkreter Beispielen erläutert und die Notwendigkeit zur Entwicklung lernbasierter Erkennungsansätze ersichtlich.
2. *Lernbasierte Erkennung.* Anschließend wird eine lernbasierte Methode names DREBIN vorgestellt, die eine zuverlässige Erkennung verschiedener Typen von Schadsoftware direkt auf dem Mobilgerät erlaubt. Hierfür analysiert DREBIN einen großen Datenbestand aus gutartigen und bösartigen Applikationen zunächst statisch und leitet die notwendigen Muster zur Erkennung von Schadsoftware mittels maschineller Lernmethoden automatisch ab. Das daraus resultierende Erkennungsmodell kann anschließend auf dem Mobilgerät verwendet werden. In einer Vielzahl von Experimenten wird gezeigt, dass DREBIN bei geringer Laufzeit meist deutlich bessere Ergebnisse erzielt als vergleichbare Methoden [Ar14].
3. *Analyse des Erkennungsmodells.* Das Erkennungsmodell wird im Detail analysiert, wobei unter anderem die Erklärbarkeit sowie die Robustheit der Methode gegenüber

gezielten Angriffen untersucht wird. Im Gegensatz zu anderen Verfahren liefert DREBIN aussagekräftige Erklärungen für seine Entscheidungen, was durch mehrere Experimente untermauert wird. Weiterhin ist es eines der ersten Erkennungssysteme, dessen Robustheit gegenüber realistischen Angriffen systematisch untersucht und verbessert wird [Ar14, De19].

Im Folgenden wird die Studie über ultraschallbasierte Malware (Abschnitt 2), sowie die Methode zur Erkennung von Schadsoftware kurz vorgestellt (Abschnitt 3). Abschließend werden in Abschnitt 4 die wichtigsten Ergebnisse der Dissertation kurz zusammengefasst. Für eine detaillierte Diskussion der Ergebnisse sei auf die Dissertation verwiesen [Ar19].

2 Ultraschallbasierte Schadapplikationen

Um die Notwendigkeit zur Entwicklung neuer Erkennungsmethoden zu motivieren und an die Thematik der mobilen Schadsoftware heranzuführen, werden zunächst die Ergebnisse einer Studie über die Verwendung eines ultraschallbasierten Seitenkanals in mobilen Applikationen (kurz *Apps*) vorgestellt. Hierbei kann in einem Fall gezeigt werden, dass diese Technologie von einem der Anbieter missbraucht wird, um das Konsumverhalten von Mobilnutzern heimlich auszuspionieren.

In der Studie werden Software-Lösungen von drei verschiedenen Anbieter analysiert, die im Frequenzbereich zwischen 18 kHz und 20 kHz Informationen übertragen. Die Übertragung von Informationen in diesem ultraschallnahen Frequenzbereich hat den Vorteil, dass sie für die meisten Menschen nicht hörbar ist und dadurch für verschiedene Zwecke verwendet werden kann. Abbildung 1 zeigt etwa ein Beispiel, in dem zusätzliche Informationen zusammen mit einem Musikstück übertragen werden. Während der Einsatz dieser Technologie bei zwei der Anbieter mit dem Einverständnis der Mobilnutzer geschieht, kann bei der Analyse des *Software Development Kits (SDK)* des Anbieters *SilverPush* festgestellt werden, dass personenbezogene Daten der Nutzer ohne deren Wissen gesammelt und an den Anbieter gesendet werden.

Das SDK dieses Anbieters kann in verschiedene mobile Apps integriert werden und erlaubt dem Anbieter durch die Verwendung des Ultraschall-Seitenkanals ein geräteübergreifendes Profil des Konsumverhaltens einer Nutzerin zu erstellen. Dies ermöglicht wiederum das Anzeigen personalisierter Werbung. Hierfür wird von Apps mit diesem SDK ein Hintergrundprozess gestartet, der kontinuierlich die Umgebungsgeräusche nach speziellen, hochfrequenten Signalen absucht. Diese sogenannten *Audio Beacons* werden im Audiosignal verschiedener Medien, wie etwa Fernsehwerbung, eingebettet und erlauben SilverPush das Konsumverhalten einer Benutzerin geräteübergreifend zu verfolgen. Weiterhin sammelt die Spionagesoftware eine Vielzahl von sensiblen Daten der Geräte, einschließlich der Telefonnummern von Nutzern, und sendet diese zusammen mit anderen personenbezogenen Informationen an die Entwickler von SilverPush.

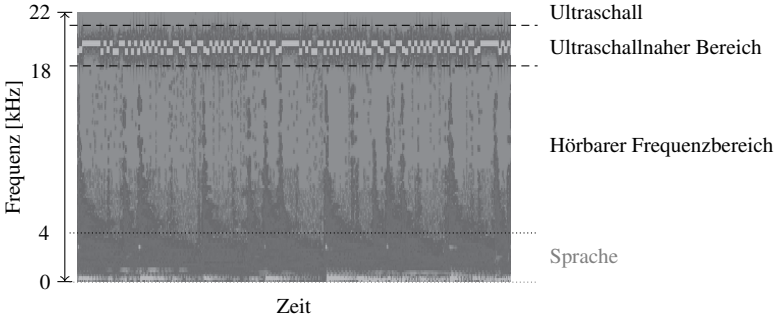


Abb. 1: Beispiel eines im Frequenzbereich zwischen 18 und 20 kHz übertragenen Signals.

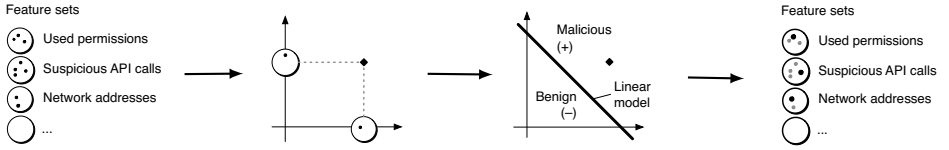
Ergebnisse. Neben einer detaillierten Analyse der Schadsoftware wird auch ihre Verbreitung untersucht, da der Anbieter nicht bekanntgegeben hat mit welchen Firmen er kollaboriert. Hierzu wird ein Datenbestand mit mehr als 1,3 Millionen mobilen Applikationen analysiert. Mit Hilfe speziell auf die Erkennung dieser Schadsoftware zugeschnittenen Techniken können insgesamt 234 Android-Applikationen identifiziert werden, die das SDK enthalten, wobei einige Applikationen millionenfach aus dem GooglePlay Store heruntergeladen wurden. Durch eine Zusammenarbeit mit Google konnten alle Applikationen mit dem Silverpush-SDK aus dem Store entfernt werden, sodass von dieser speziellen Spionagesoftware keine Gefahr für die Privatsphäre von Mobilnutzern mehr ausgehen sollte. Durch die hohe Konzentration sensibler Nutzerdaten bleiben Smartphones allerdings weiterhin ein beliebtes Ziel von Malware-Autoren und dubiosen Werbefirmen. Nicht zuletzt zeigt der Fall auch die Notwendigkeit für die Entwicklung von neuen Erkennungstechniken für mobile Schadsoftware.

3 Lernbasierte Erkennung mobiler Schadsoftware

Die im vorangegangenen Abschnitt beschriebene Spionagesoftware zeigt eindrucksvoll die rasante Weiterentwicklung im Bereich der mobilen Malware. Eine aussichtsreiche Richtung dieser Entwicklung zu begegnen stellt der Einsatz maschineller Lernmethoden dar, da diese Techniken die automatische Extraktion effektiver Erkennungsmuster aus großen Datenbeständen ermöglichen. Die in der Dissertation vorgestellte Methode namens DREBIN nutzt diese Techniken, um eine effektive, effiziente und zugleich interpretierbare Erkennung von mobiler Schadsoftware zu bewerkstelligen. Die Details der Methode werden in den folgenden Abschnitten erläutert.

3.1 Überblick der Methode

Zur Erkennung von Schadsoftware auf einem Mobilgerät muss DREBIN Applikationen in eine Repräsentation überführen, die für die Anwendung von maschinellen Lernalgorithmen geeignet ist. Hierzu wird eine Applikation zunächst statisch analysiert und relevante



(a) Statische Analyse (b) Vektorraum-Einbettung (c) Lernbasierte Erkennung (d) Erklärung

Abb. 2: Schematische Darstellung der verschiedenen Schritte des Erkennungssystems

Merkmale aus der Applikation extrahiert, die es anschließend erlauben, Android-Apps in eine neuartige Vektorrepräsentation zu überführen. In dem daraus resultierenden hochdimensionalen Vektorraum können Applikationen schließlich weiter analysiert und klassifiziert werden. Ein Überblick über diesen Prozess ist schematisch in Abbildung 2 dargestellt. Dabei werden im Detail folgende Schritte durchlaufen:

a) *Statische Analyse.* Im ersten Schritt führt DREBIN eine effiziente Code-Analyse der Android-Applikation durch und extrahiert dabei unterschiedliche Merkmalstypen aus verschiedenen Bestandteilen einer Anwendung.

b) *Vektorraum-Einbettung.* Anschließend werden die Merkmale in einen hochdimensionalen Vektorraum abgebildet, der es ermöglicht, nach relevanten Mustern und markanten Kombinationen von Merkmalen zu suchen.

c) *Lernbasierte Erkennung.* Durch die Überführung in den Vektorraum wird die Anwendung von Lernalgorithmen möglich, um Schadsoftware effektiv zu erkennen.

d) *Interpretation.* Schließlich werden Merkmale identifiziert, die auf die Entscheidung einen hohen Einfluss hatten und der Benutzerin oder Malware-Analystin präsentiert.

Die folgenden Abschnitte legen die einzelnen Schritte der Methode etwas ausführlicher dar. Abschließend wird die Robustheit der Methode diskutiert. Für Details sei auf die Ausführungen in der Dissertation [Ar19] beziehungsweise der entsprechenden Publikationen [Ar14, De19] verwiesen.

3.1.1 Statische Analyse mobiler Applikationen

Zunächst wird die mobile Applikation direkt auf dem Mobilgerät statisch analysiert. Aufgrund der Hardwarelimitierungen von Mobilgeräten muss dies besonders effizient und ressourcensparend erfolgen, um der Benutzerin in annehmbarer Zeit ein Ergebnis präsentieren zu können. Dafür analysiert DREBIN die sogenannte *Manifest*-Datei und den disassemblierten *Dalvik Bytecode* einer App. Hierbei werden insgesamt 8 verschiedene Merkmalskategorien aus der Anwendung extrahiert.

- *Android-Manifest.* Die Manifest-Datei ist fester Bestandteil einer jeden Android-Applikation und enthält verschiedene Meta-Informationen, die während der Instal-

lation und zur Laufzeit einer Anwendung benötigt werden. Aus dieser Datei extrahiert DREBIN verschiedene Informationen, wie die angeforderten Berechtigungen einer Applikation und die Namen der Komponenten, aus der sich die Anwendung zusammensetzt.

- *Dalvik Bytecode.* Android-Applikationen werden meist in Java oder Kotlin geschrieben und dann in sogenannten Dalvik Bytecode übersetzt. DREBIN disassembliert zunächst den Bytecode einer Applikation und extrahiert anschließend verschiedene Merkmale, die zur Erkennung von Schadsoftware relevant sein können. Dies schließt beispielweise bestimmte Aufrufe der Android-API mit ein, die häufig in mobiler Schadsoftware zu finden sind.

Im Gegensatz zu vorherigen Methoden führt DREBIN eine sehr breite Analyse der Applikationen durch und kann so erstmals verschiedene Varianten von Schadsoftware modellieren und erkennen.

3.1.2 Einbettung in den Vektorraum

Jede der von DREBIN extrahierten Merkmalskategorien bildet eine Menge von Zeichenketten. Da die meisten maschinellen Lernmethoden allerdings auf Vektorräumen agieren, müssen Android-Applikationen entsprechend erst als Vektoren repräsentiert werden können. Um zu einer passenden Vektorrepräsentation zu gelangen, werden die einzelnen Merkmalskategorien zunächst zu einer Gesamtmenge S vereint, die die Menge aller zur Trainingszeit extrahierten Merkmale beschreibt.

Die vereinte Menge S erlaubt anschließend die Definition eines $|S|$-dimensionalen Vektorraums, in dem jede Dimension entweder den Wert 0 oder 1 annimmt. Um eine Android-Applikation $z \in \mathscr{Z}$ in diesen Vektorraum zu überführen, wird eine Mappingfunktion $\varphi : \mathscr{Z} \to \{0,1\}^{|S|}$ verwendet. Diese bildet die Applikation z aus der Menge aller Android-Applikationen $\mathscr{Z}$ auf einen Vektor $\varphi(z) \in \{0,1\}^{|S|}$ ab, in welchem die entsprechenden Dimensionen aller aus der App extrahierten Merkmale auf 1 und aller nicht-vorhandenen Merkmale auf 0 gesetzt werden. Die Vektorrepräsentation $\varphi(z)$ einer Applikation z könnte beispielweise wie folgt aussehen:

$$\varphi(z) \mapsto \begin{pmatrix} \cdots \\ 0 \\ 1 \\ \cdots \\ 1 \\ 0 \\ \cdots \end{pmatrix} \begin{matrix} \cdots \\ \texttt{HARDWARE::android.hardware.wifi} \\ \texttt{HARDWARE::android.hardware.telephony} \\ \cdots \\ \texttt{REQ\_PERMISSION::SEND\_SMS} \\ \texttt{REQ\_PERMISSION::DELETE\_PACKAGES} \\ \cdots \end{matrix}$$

Durch diese neuartige Einbettung von Android-Applikationen wird es möglich, diese effizient zu analysieren und dadurch häufige Muster mobiler Schadsoftware automatisch zu

extrahieren. Der Ansatz baut dabei auf Konzepten aus der Sprachverarbeitung [SWY75] und Bioinformatik [LEN02] auf und verbindet so moderne Konzepte der Datenanalyse mit aktuellen Problemstellungen aus der IT-Sicherheit.

3.1.3 Erkennung von Schadsoftware

In dem resultierenden Vektorraum liegen Applikationen geometrisch dichter beieinander, die ähnliche Merkmale aufweisen, als Applikationen, die sehr verschieden sind. Im nächsten Schritt muss daher eine Funktion gefunden werden, die in diesem Raum gutartige und schädliche Applikationen geometrisch voneinander trennen kann.

Für diese Aufgabe existiert eine Vielzahl von Lernalgorithmen, jedoch erfordern viele einen hohen Rechenaufwand und nur wenige erlauben eine Interpretation der zurückgelieferten Ergebnisse. DREBIN verwendet daher einen bereits gut verstandenen Lernalgorithmus — die lineare *Support Vector Machine* (SVM) [Va79]. Diese ist für die Anwendung besonders gut geeignet, da durch die gezielte Ausnutzung charakteristischer Eigenschaften des Algorithmus eine sehr effiziente und zugleich erklärbare Erkennung von Schadsoftware möglich wird.

Die Verwendung der linearen SVM erlaubt schließlich die Klassifizierung von Apps direkt auf dem Mobilgerät. Hierfür kann der Umstand ausgenutzt werden, dass nur die in einer Applikation vorhandenen Merkmale und auch diese lediglich bei ihrem ersten Auftreten im Bytecode in die Entscheidung einfließen. Somit steigt die Laufzeit zur Erkennung von Apps nicht einmal linear mit der Größe des Bytecodes an.

3.1.4 Interpretation der Ergebnisse

Ein großer Nachteil vieler maschineller Lernmethoden besteht darin, dass die von ihnen zurückgelieferten Modelle meist nicht interpretierbar sind [SP10]. DREBIN ist hingegen in der Lage, Erklärungen für seine Entscheidungen zurückzuliefern.

Das Erkennungsmodell wurde bei DREBIN so gewählt, dass es sich um eine lineare Funktion handelt und somit für jedes extrahierte Merkmal der Einfluss an der Gesamtentscheidung exakt bestimmt werden kann. Während der Berechnung der Erkennungsfunktion werden dazu die k Merkmale gespeichert, die den größten Einfluss auf die Entscheidung hatten. Diese können anschließend in verständlich aufbereiteter Form der Smartphone-Nutzerin oder einer Analystin präsentiert werden. Somit wird es möglich nachzuvollziehen, weshalb eine App von DREBIN als gutartig oder bösartig eingestuft wurde.

3.1.5 Angriffe auf das Lernmodell

Die Sicherheit lernbasierter Systeme gegen gezielte Angriffe ist ein wichtiger Bestandteil aktueller Forschung. Auch in der Dissertation wird die Durchführbarkeit von *Evasion-*

angriffen [De19] auf das Erkennungsmodell untersucht, da diese Art von Angriffen eine realistische Bedrohung für lernbasierte Erkennungssysteme in der Praxis darstellt.

Bei einem Evasionangriff manipuliert die Angreiferin gezielt die Merkmale einer Applikation, um eine Falsch-Klassifikation des Systems herbeizuführen. Der Angriff ist hierbei allerdings gewissen Limitierungen unterworfen, da die Angreiferin in der Praxis nicht beliebige Merkmale manipulieren kann, ohne die Funktionalität einer App zu beeinflussen. In der Dissertation wird gezeigt, dass diese Angriffe unter bestimmten Umständen tatsächlich die Erkennungsleistung des Systems reduzieren können, da einzelne Merkmale auf die Entscheidungsfunktion der SVM einen hohen Einfluss haben können. Gelingt es der Angreiferin eben diese Merkmale zu ermitteln und zu manipulieren, sinkt die Erkennungsleistung des Systems signifikant.

Um hier entgegenzuwirken, kann etwa das Optimierungsproblem der SVM derart modifiziert werden, sodass sichergestellt wird, dass eine Entscheidung nicht nur von wenigen Merkmalen abhängt. Der resultierende Klassifikator, genannt Sec-SVM, weist dadurch eine deutlich höhere Robustheit gegenüber gezielten Angriffen auf.

4 Zusammenfassung der wichtigsten Ergebnisse

In der Dissertation wird eine neuartige Methode namens DREBIN vorgestellt, die eine effiziente, effektive und interpretierbare Erkennung von mobiler Schadsoftware direkt auf dem Mobilgerät ermöglicht. Dies wird durch zahlreiche Experimente untermauert, in denen unter anderem die Erkennungsleistung, Interpretierbarkeit, sowie Robustheit gegen gezielte Angriffe untersucht wird. Zur Evaluation werden mehrere Datensätze verwendet, die sich aus insgesamt mehr als 450.000 Android-Applikationen aus dem Zeitraum zwischen 2010 und 2017 zusammensetzen.

Erkennungsleistung. Beim Vergleich mit anderen Methoden kann gezeigt werden, dass DREBIN meist deutlich bessere Ergebnisse erzielt und eine zuverlässige Erkennung von Schadsoftware ermöglicht, einschließlich der in Abschnitt 2 diskutierten ultraschallbasierten Spionage-Applikationen. In den meisten Experimenten kann eine Erkennungsrate von über 90% bei einer geringen Falsch-Positiv-Rate von gerade einmal 1% erreicht werden. Dies entspricht einem Falsch-Alarm pro 100 installierten Applikationen, was auch im praktischen Einsatz akzeptabel zu sein scheint, insbesondere wenn die Detektion direkt auf dem Gerät erfolgt.

Auch bei einem Vergleich mit populären Antivirenscannern erzielt DREBIN vergleichbare und in vielen Fällen sogar bessere Ergebnisse. Während dieser Untersuchung zeigt sich auch, dass Antivirenprodukte häufig eine gewisse Vorlaufzeit für die Bereitstellung effektiver Erkennungssignaturen benötigen. Dies kann im schlimmsten Fall zu einer unerkannten Infektion des Mobilgeräts führen.

Laufzeit. Zur Evaluation des Laufzeitverhaltens wird ein Prototyp der Applikation implementiert und die Dauer zur Ausgabe eines Ergebnisses auf fünf Mobilgeräten mit unterschiedlicher Rechenleistung untersucht. Die Laufzeit wird hierbei mit 100 populären Applikationen aus dem GooglePlay Store durchgeführt. Im Durchschnitt dauert es auf den Geräten weniger als 15 Sekunden bis eine Entscheidung zurückgeliefert werden kann und selbst auf sehr rechenschwachen Geräten nie länger als eine Minute.

Erklärbarkeit. Im Gegensatz zu allen zuvor vorgestellten lernbasierten Erkennungsmethoden für mobile Malware stellt die Interpretierbarkeit einen essentiellen Bestandteil von DREBIN dar. In einer Evaluation mit populären Malware-Familien kann gezeigt werden, dass DREBIN in der Lage ist, automatisch charakteristische Merkmale dieser Familien zu identifizieren und zur Erkennung zu nutzen. Dieses Ergebnis wird sowohl von den präsentierten Erklärungen als auch einer detaillierten Analyse der Generalisierungseigenschaften des gelernten Modells gestützt.

Robustheit. Zwar kann gezeigt werden, dass Evasionangriffe unter bestimmten Voraussetzungen die Erkennungsleistung von DREBIN beeinflussen können, allerdings existieren unterschiedliche Möglichkeiten den Einfluss solcher Angriffe stark zu reduzieren. So verhindert etwa der Einsatz der Sec-SVM, dass eine Entscheidung nur von einer kleinen Zahl von Merkmalen abhängt, die der Angreifer unter Umständen manipulieren kann. Infolgedessen erkennt die Sec-SVM beispielsweise noch weit über 80% der Schadsoftware, selbst wenn der Angreifer mehrere Merkmale manipulieren kann.

5 Fazit

Zusammenfassend zeigen die in dieser Dissertation vorgestellten Ergebnisse, dass die Verwendung maschineller Lernverfahren einen vielversprechenden Ansatz darstellt, um die Sicherheit mobiler Geräte zu verbessern. Zwar können auch diese Techniken die Bedrohung durch mobile Schadanwendungen nicht vollends beseitigen, aber zumindest die Infektion von Mobilgeräten deutlich zu erschweren. Die vorgestellte Methode leistet hier einen wichtigen Beitrag und dient Wissenschaftlern weltweit als Grundlage für weiterführende Forschung in diesem Bereich.

Literaturverzeichnis

[Ar14] Arp, Daniel; Spreitzenbarth, Michael; Hübner, Malte; Gascon, Hugo; Rieck, Konrad: Drebin: Efficient and Explainable Detection of Android Malware in Your Pocket. In: Proc. of Network and Distributed System Security Symposium (NDSS). 2014.

[Ar17] Arp, Daniel; Quiring, Erwin; Wressnegger, Christian; Rieck, Konrad: Privacy Threats through Ultrasonic Side Channels on Mobile Devices. In: Proc. of IEEE European Symposium on Security and Privacy (EuroS&P). 2017.

[Ar19] Arp, Daniel: Efficient and Explainable Detection of Mobile Malware with Machine Learning. Dissertation, Technische Universität Braunschweig, 2019.

[Bu19] Bundeskriminalamt: Bundeslagebild Cybercrime 2018. Bericht, 2019.

[De19] Demontis, A.; Melis, M.; Biggio, B.; Maiorca, D.; Arp, D.; Rieck, K.; Corona, I.; Giacinto, G.; Roli, F.: Yes, Machine Learning Can Be More Secure! A Case Study on Android Malware Detection. IEEE Transactions on Dependable and Secure Computing (TDSC), 2019.

[LEN02] Leslie, Christina S.; Eskin, Eleazar; Noble, William Stafford: The Spectrum Kernel: A String Kernel for SVM Protein Classification. In: Proc. of the Pacific Symposium on Biocomputing (PSB). S. 566–575, 2002.

[MKK07] Moser, Andreas; Kruegel, Christopher; Kirda, Engin: Limits of Static Analysis for Malware Detection. In: Proc. of Annual Computer Security Applications Conference (ACSAC). 2007.

[Ne19] Newzoo: Global Mobile Market Report. Bericht, 2019.

[SP10] Sommer, R.; Paxson, V.: Outside the Closed World: On Using Machine Learning for Network Intrusion Detection. In: Proc. of IEEE Symposium on Security and Privacy (S&P). 2010.

[SWY75] Salton, G.; Wang, A.; Yang, C. S.: A Vector Space Model for Automatic Indexing. Communication of the ACM (CACM), 18(11):613–620, 1975.

[Va79] Vapnik, V.N.: Estimation of Dependences Based on Empirical Data [in Russian]. Nauka, 1979.

[Wr17] Wressnegger, Christian; Freeman, Kevin; Yamaguchi, Fabian; Rieck, Konrad: Automatically Inferring Malware Signatures for Anti-Virus Assisted Attacks. In: Proc. of ACM Asia Conference on Computer Computer and Communications Security (ASIA CCS). 2017.

Daniel Arp, geboren am 19. März 1986 in Berlin, ist derzeit Post-Doktorand am Institut für Systemsicherheit der Technischen Universität Braunschweig. Er hat Technische Informatik an der Technischen Universität Berlin studiert und anschließend an der Georg-August-Universität Göttingen und der Technische Universität Braunschweig im Fachbereich IT-Sicherheit promoviert. In seiner Dissertation stellt er ein neuartiges Verfahren zur Erkennung mobiler Schadsoftware vor und wurde für seinen Beitrag im Bereich der Malwareerkennung dafür bereits vom Wirtschaftsministerium Niedersachsen als “KI Talent 2019” ausgezeichnet. Während seiner Promotion veröffentlichte er zahlreiche wissenschaftliche Beiträge, die insgesamt mehr als 2.000 Mal zitiert wurden.