Kann Malware in Android-Apps automatisch gefunden werden?

Cöllen, Markus
Hochschule Mannheim
Fakultät für Informatik
Paul-Wittsack-Str. 10, 68163 Mannheim

***Zusammenfassung*—An dieser Stelle steht eine kurze Zusammenfassung des Inhaltes des Dokuments.**

Inhaltsverzeichnis

1. Einleitung

2. Grundlagen

2.1. Android als Ziel

Smartphones bieten aufgrund ihrer vielen Schittstellen zur Außenwelt, wie z.B. WLAN, Bluetooth, NFC, USB usw. viele Angriffspunkte. Ein Hauptgrund, warum Smartphones mit dem Android Betriebssystem so beliebt für Angreifer sind, ist der hohe Verbreitungsgrad dieser Geräte. Dieser liegt im Jahr 2018 bei ca. 86 Prozent und ist somit Sechs mal so hoch wie der Konkurrent IOS mit knapp 16 Prozent Marktanteil [5]. Weil Android unter der Apache Open-Source Lizenz freigegeben ist und das Betriebssystem auf dem Linux-Kernel basiert, können Hersteller das Andoid Betriebssystem beliebig verwenden und für ihre Produkte anpassen. Zudem dient der Linux-Kernel als eine Abstraktion zwischen Software und Hardware und somit kann Android auf verschiedenen Geräten eingesetzt werden. Aufgrund der Tatsache, dass Linux über ein etabliertes Sicherheitskonzept verfügt [9], greift Android auf viele Schutzmechanismen des Linux Kernels zurück oder benutzt diese als ihre Basis [8].

2.2. Android Update Problematik

Wie man in Abbildung 1 sehen kann verdrängen die neueren Android Version die älteren, allerdings ist Android Oreo, mit gerade mal 4,6 Prozent, eine der am wenigsten vertretenden Version, obwohl sie schon seit dem 27. August 2017 auf dem Markt ist.

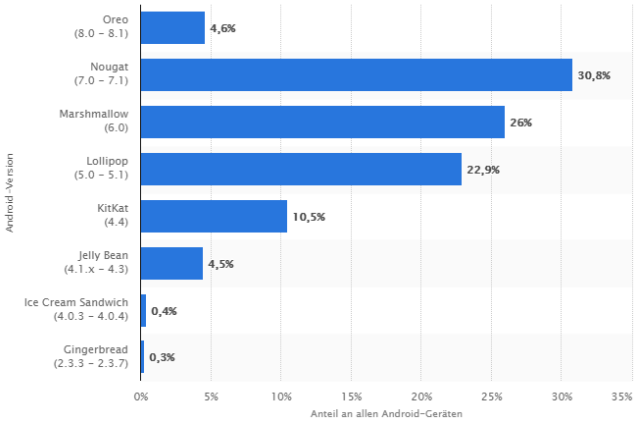

Abbildung 1. Android-Versionen [1]

Dies liegt daran, dass nicht jede Andoid Version mit jedem Handy kompatible ist. Die Smartphone Hersteller müssen die Versionen an ihre Geräte und die jeweilige Hardware anpassen. Diese umfassen Anpassungen an WLAN, Kamera, Bluetooth, GPS usw. Anschließend müssen die erstellten Anpassungen noch umfangreich getestet werden, was auchnoch viel Zeit in anspruch nimmt. Weil die Hersteller oft neue Produkte verkaufen wollen und das Updaten älterer Smartphones zu teuer und Zeitaufwendig ist, bleibt den Nutzern oft nichts anderes übrig, als neue Smartphones zu kaufen. Anderndfalls behält man ein Smartphone mit einer veralteten Version und bekannten Sicherheitslücken. [8].

2.3. Application Sandbox

Linux arbeitet als ein Mehrbenutzer-Betriebssystem, das bedeutet es kann mehrere Nutzer geben und das Betriebssystem verhindert, dass Daten eines Nutzers von einem anderen Nutzer eingesehen, geändert oder gelöscht

werden können. Diese Nutzermanagement wird beim Application Sanboxing verwendet. Hierbei erhält jede Applikation eine eigene Nutzer-ID und führt diese in einem separaten Prozess aus. Somit gewährleistet Android, dass eine Applikation, anderen Applikationen oder dem Betriebssystem keinen Schaden zufügen kann. Sollte eine Applikation aber durch das Ausnutzen einer Schwachstelle oder einer Sicherheitslücke an Root-Recht kommen, kann die Applikation dieses Application Sanboxing einfach umgehen. Wie schon in Abschnitt **??** beschrieben, gibt es wegen den stark verzögerten Android Updates oft Sicherheitslücken, welche über längere Zeit bekannt sind und leicht ausgenutzt werden können um Root-Rechte zu erlangen [8].

2.4. Permission Model

Standardmäßig hat eine Applikation keinen Zugriff auf Daten außerhalb der Sandbox. Sollte eine Applikation Systemressourcen außerhalb dieser verwenden wollen, muss erst eine Zugriffsanfrage gestellt werden. Geschützte Ressourcen sind z.B. Kamera, SMS, Bluetooth usw. [6].

2.5. Google Play Store

Anders als bei IOS, bei welchem die Nutzer an den Apple Store gebunden sind, können die Android Benutzer selbst entscheiden welchen Store sie verwenden. Der Google Play Store ist mit 82 Milliarden Downloads pro [2] mit Abstand die größte Plattform. Er wurde am 28. August 2008 unter dem Namen Android Market veröffentlicht. Dieser bietet den Nutzern einfaches herunterladen und installieren von mobilen Anwendungen, sogenannten Applikation. Im Google Play Store sind bereits über 3,7 Millionen Anwendungen bereit zum Herunterladen [4] und jeden Monat kommen ca. 30.000 neue Applikation hinzu [3]. Durch den rasanten Wachstum steigt auch die Anzahl von schädlicher Software, sogenannter Malware. Der Anteil von bösartigen Applikation ist von 2011 bis 2013 um 388% gewachsen [7]. Da nicht alle Anwendungen von Mitarbeitern geprüft werden können hat Google das Programm Bouncer ins Leben gerufen (siehe Kapitel 4). Android bietet zudem die Möglichkeit per Remote-Verbindung Applikationen zu installieren und zu löschen. Hierfür braucht man lediglich Zugriff auf den Google Play Account. Falls ein Angreifer an diese Daten kommen sollte, könnte er ohne Erlaubnis des Nutzers, Applikationen aus dem Google Play Store installieren.

2.6. Malware

Der Begriff Malware steht für malicous software und bezeichnet Programme, welche unerwünschte oder auch schädliche Funktionen ausführen. Sie stellen eine immer größer werdende Bedrohung dar und durch den rasanten Wachstum ist eine manuelle Auswertung mittlerweile unmöglich geworden. Obwohl es sich bei vielen neuen Arten um verschiedene Varianten bereits bekannter Malware handelt, müssen Analysten erst jedes Sample erneut analysieren um dies feststellen zu können [10]. Bei der Analyse kann Grundsätzlich in zwei Arten unterschieden werden, Statische und Dynamische Analyse (siehe Kapitel 3.1 und 3.2).

3. Analysemethoden

3.1. Statische Analyse

3.1.1. Data Flow.

3.1.2. Control Flow.

3.2. Dynamische Analyse

4. Bouncer

5. Fazit

Eine Abkürzung = Application-to-Application (A2A)

Abkürzungen

A2A Application-to-Application

Literatur

[1] *Anteile der verschiedenen Android-Versionen an allen Geräten mit Android OS weltweit im Zeitraum 10. bis 16. April 2018*. https://de.statista.com/statistik/daten/studie/180113/umfrage/anteil-der-verschiedenen-android-versionen-auf-geraeten-mit-android-os/. Accessed: 2018-04-25.

[2] *Anzahl der Apps, die im Google Play Store heruntergeladen wurden*. https://de.statista.com/statistik/daten/studie/243412/umfrage/anzahl-von-downloads-im-google-play-store/. Accessed: 2018-04-25.

[3] Steffen Bartsch u.a. „Zertifizierte Datensicherheit für Android-Anwendungen auf Basis statischer Programmanalysen." In: *Sicherheit*. 2014, S. 283–291.

[4] *Google Play Apps Statistik*. https://de.statista.com/statistik/daten/studie/74368/umfrage/anzahl-der-verfuegbaren-apps-im-google-play-store/. Accessed: 2018-04-22.

[5] *Marktanteil von Android*. https://www.netzwelt.de/news/164146-android-vs-ios-plattform-treue-android-91-prozent-deutlich-hoeher.html. Accessed: 2018-04-25.

[6] *Request App Permissions*. https://developer.android.com/training/permissions/requesting.html. Accessed: 2018-04-25.

[7] *RiskIQ report about Malicious Mobile Apps in Google Play*. https://www.riskiq.com/press-release/riskiq-reports-malicious-mobile-apps-google-play-have-spiked-nearly-400/. Accessed: 2018-04-22.

[8] Julian Scheid. „Kapitel 1 Sicherheit mobiler Geräte-Schutzmaßnahmen, Angriffsarten & Angriffserkennung auf Android". In: *Ausgewählte Themen der IT-Sicherheit* (2012), S. 7.

[9] *Sicherheitskonzepte von Linux*. https://wiki.ubuntuusers.de/Archiv/Sicherheitskonzepte/. Accessed: 2018-04-25.

[10] Philipp Trinius. *Visualisierung von Malware-Verhalten*. Universitätsbibliothek Dortmund, 2010.