Zentrum für sichere Informationstechnologie – Austria
Secure Information Technology Center – Austria

A-SIT

Seidlgasse 22 / 9, 1030 Wien
Tel.: (+43 1) 503 19 63–0
Fax: (+43 1) 503 19 63–66

Inffeldgasse 16a, 8010 Graz
Tel.: (+43 316) 873-5514
Fax: (+43 316) 873-5520

http://www.a-sit.at
E-Mail: office@a-sit.at

DVR: 1035461 ZVR: 948166612 UID: ATU60778947

ANDROID BROWSER – SICHERHEIT UND PRIVATSPHÄRE

Version 1.0 vom 24.11.2017
Alexander Marsalek – alexander.marsalek@a-sit.at
Dominik Ziegler – dominik.ziegler@a-sit.at

Zusammenfassung: Dieses Dokument beschreibt und vergleicht unterschiedliche Techniken, wie DNS-Sperren, IP-Sperren oder Xposed basierte Ansätze zur Blockierung von Werbung auf Android Geräten. Auf Basis des erstellten Vergleichs wurde die beste Technik ausgewählt und basierend darauf ein Werbe- und Malware-Blocker umgesetzt. Die praktische Evaluierung bestätigte dessen Funktionalität.

Inhaltsverzeichnis

1. Einleitung

Online-Werbung ist ein stetig wachsender Markt. Bereits 2007 wurden in Deutschland 976 Millionen Euro für Online-Werbung ausgegeben, dies entspricht einem Zuwachs von 103 Prozent im Vergleich zum Vorjahr [1]. Für 2017 wird in Deutschland bereits ein Umsatz von über 6,9 Milliarden Euro mittels Online-Werbung erwartet [2]. Aus Sicht von Anwendern und Anwenderinnen können sich einige Vorteile durch die Blockierung von Werbung ergeben. Zu den Vorteilen kann ein schnelleres Laden

der Seite gehören, geringerer Datenverbrauch, geringere Prozessorlast und dadurch ein geringerer Energieverbrauch sowie erhöhte Sicherheit durch den Wegfall von Malvertising. Unter Malvertising versteht man eine Methode, um schädliche Programme über Internet-Werbung zu verbreiten. Dies hat für den Angreifer den Vorteil, dass zielgerichtet bestimmte Seiten oder Personen attackiert werden können, ohne dass ein Server angegriffen oder übernommen werden muss. So können auch vertrauenswürdige Seiten wie die New York Times unbeabsichtigt Schadsoftware verbreiten [3] [4]. In der Desktopwelt gibt es viele Möglichkeiten sich gegen Schadsoftware zu schützen bzw. Werbung zu blockieren. Allerdings benötigt Software zum Schutz gegen Schadsoftware meist hohe Systemberechtigungen. Software zum Blockieren von Werbung wird außerdem meist in Form von Plugins oder Extensions für Browser angeboten. Diese Ansätze sind für mobile Plattformen wie Android, aufgrund unterschiedlicher Technologien, jedoch nur bedingt anwendbar. Ziel dieses Projektes ist es die Möglichkeiten zur Umsetzung solcher Lösungen auf Android vorzustellen und einen Demonstrator mit einer der vorgestellten Technologien umzusetzen. Im nächsten Kapitel werden verschiedene Ansätze kurz vorgestellt.

2. Grundlagen

In Android wird jede App in einer Sandbox ausgeführt, dadurch kann eine dritte App, beispielsweise ein Werbeblocker, nicht einfach auf die Daten von anderen Apps zugreifen oder diese verändern. Daher lassen sich beispielsweise Virenscanner, die alle Dateien scannen, nicht wie auf Desktopsystemen umsetzen. Auch Werbeblocker können nicht, wie auf Desktopsystemen, mittels Plugins oder Extensions umgesetzt werden, da gängige mobile Browser wie beispielsweise Chrome, zum Zeitpunkt dieser Studie, keine Unterstützung dafür anbieten. Zudem spielt auf mobilen Geräten neben der Werbung in Browsern zunehmend In-App Werbung eine große Rolle. Das restliche Kapitel beschreibt zuerst das notwendige Grundwissen und anschließend Methoden wie Werbeblocker dennoch auf Android Geräten umgesetzt werden können. Dabei werden sowohl Methoden für gerootete, also Systeme mit erhöhten Rechten, als auch für nicht gerootete Geräte berücksichtigt.

2.1. IP-Adressen

Jeder Rechner im Internet verfügt über eine oder mehrere IP-Adressen(n). Mittels dieser Adresse kann der Rechner im Netzwerk eindeutig erreicht werden. Beispielsweise kann der Rechner hinter „www.a-sit.at/“ unter der IPv4-Adresse 129.27.142.97 erreicht werden. Da IPv4-Adressen nur aus 32 Bit bestehen sind nur ca. 4,2 Milliarden unterschiedliche Adressen darstellbar. Da absehbar war, dass früher oder später alle IPv4-Adressen vergeben sein werden wurde IPv6 entwickelt. IPv6 Adressen bestehen aus 128 Bit, wodurch sich mehrere hundert Sextillionen Adressen[1] darstellen lassen. IPv6 Adressen werden meist hexadezimal dargestellt. Die Adresse „2a00:1450:4001:81d::200e“ gehört beispielsweise zu „ipv6.google.com“. Will man ganze Domains blockieren kann dies beispielsweise auf IP-Adress-Ebene durchgeführt werden, indem Daten an diese Adresse schlicht verworfen werden. Da IPv4- und IPv6-Adressen schwer zu merken sind wurde DNS entwickelt, welches die Zuordnung von Namen zu Adressen erlaubt.

2.2. DNS

Das Domain Name System (DNS) wird in vielen IP-basierten Netzwerken zur Namensauflösung verwendet. Wie in Abschnitt 2.1 beschrieben löst das Domain Name System für Menschen gut merkbare Domainnamen in die dazugehörigen IP-Adressen auf. Diese Funktion kann mit einem Telefonbuch verglichen werden. Ein DNS-Server kann bestimme Domains blockieren, indem er deren Adresse nicht auflöst, oder eine falsche Adresse zurückgibt. Diese Technik kann zur Blockierung von Werbung verwendet werden. In manchen Ländern wird diese Technik zudem eingesetzt, um bestimmte Internetportale zu blockieren [5]. Allerdings wird bei dieser Technik nicht der eigentliche Server blockiert, wodurch sich diese Sperren relativ leicht umgehen lassen. Ebenso wie über DNS unterstützen die meisten Betriebssysteme die Namensauflösung über einen sogenannte „hosts“-Datei. Diese wird im nächsten Abschnitt beschrieben.

[1] 2^128 entspricht in etwa 3,4*10^38 (Dreihundertvierzig Sextillionen)

2.3. Hosts-Datei

Bei der Hosts-Datei handelt es sich um eine lokale Textdatei, welche Hostnamen zu IP-Adressen zuordnet. Betriebssysteme versuchen die Namensauflösung zuerst über die Hosts-Datei. Wird dort kein Eintrag gefunden, kommen andere Verfahren wie DNS zum Einsatz. Dadurch kann die Hosts-Datei auch als Filter verwendet werden, indem zu blockierenden Domains eine andere, „falsche“ IP-Adresse zugeordnet wird, beispielsweise 127.0.0.1 (localhost). Dadurch ist der Abruf von Daten von dieser Domain systemweit gesperrt. Aufrufe führen nur zu einem Fehler. Diese Technik wird von Schadsoftware immer wieder missbraucht um Benutzer und Benutzerinnen auf gefälschte Onlinedienste zu leiten. Für diesen Angriff wird die IP-Adresse des echten Services durch die IP-Adresse eines Server ersetzt, welcher unter der Kontrolle des Angreifers steht. Auf diesem Server wird anschließend ein Service betrieben, welches sich optisch möglichst nicht vom Original unterscheidet.

2.4. Root

Die Sandbox und andere Schutzfunktionen von Android können mittels root-Rechten umgangen werden. Dadurch können auf gerooteten Geräten ähnliche Ansätze wie in der Desktopwelt verfolgt werden. Generell kann zwischen zwei Arten von Ad-Blockern unterschieden werden, welche root-Rechte benötigen: Xposed Module und sonstige Apps. Xposed ist ein Framework, welches im Grunde beliebige Anpassungen am Betriebssystem bzw. der installierten Applikationen erlaubt. Die genaue Funktionsweise wird im folgenden Abschnitt behandelt. Ansätze, die nicht auf Xposed basieren, funktionieren meist indem sie Einträge in der Hosts-Datei einfügen (siehe Abschnitt 2.3). Diese Methode ist sehr effizient, da keine zusätzliche App im Hintergrund betrieben werden muss. Nachteile dieser Methode sind die benötigten root-Rechte, sowie die Limitierung auf Host-Ebene. Beispielsweise ist es nicht möglich, bestimmte Kontexte oder bestimmte Dateitypen zu blockieren.
AdFree for Android [6], *AdAway* [7], *MoaAB* [8] und *AdBlocker* [9] verwenden zum Beispiel diesen Ansatz [10].

2.4.1. Xposed

Bei *Xposed* [11] handelt es sich um ein Framework, welches die Anpassung des ROMs ermöglicht, ohne das APKs[2] angepasst werden müssen oder ein alternatives modifiziertes Betriebssystem installiert werden muss. Mittels Xposed können beliebige Methoden von Programmen oder des Betriebssystems zur Laufzeit umgeschrieben werden. Prinzipiell kann über Xposed jede Form von Werbung blockiert werden, die meisten Module konzentrieren sich jedoch auf die Blockierung von In-App Werbung, da hierbei entsprechende Methodenaufrufe zu Werbebibliotheken einfach entfernt werden können. Die Blockierung von Browser-Werbung gestaltet sich schwieriger, da bei Browser-Werbung die einfache Blockierung von bestimmten Methoden nicht ausreichend bzw. zielführend ist. Daher eignen sich Xposed Module gut, um Werbung von bestimmten Bibliotheken zu blockieren, aber weniger gut, um Werbung im Allgemeinen zu blockieren. Ein Vorteil von Xposed Modulen ist, dass Werbung per App blockiert oder erlaubt werden kann. Die benötigten root-Rechte, sowie die Möglichkeit nur bekannte Werbebibliotheken zu blockieren gehören zu den Nachteilen dieses Ansatzes. Zwar kann mittels Xposed theoretisch jede bekannte Form von Werbung und jeglicher Netzwerkverkehr überprüft werden, allerdings ist dafür ein stark erhöhter Aufwand im Vergleich zu anderen Techniken nötig. Daher wird Xposed meist nur zur Blockierung von bekannten Werbebibliotheken in Apps verwendet.
Xad [12], *minMinGuard* [13] und *Youtube AdAway* [14] verwenden beispielsweise diesen Ansatz [10]. *Xad* blockiert beispielsweise Werbung der Google API, während sich *Youtube AdAway* auf Werbung in der *YouTube*, *YouTube Kids*, *YouTube TV* und *YouTube Gaming* App konzentriert.

2.5. HTTP Proxy

Proxies leiten die Anfragen von ihren Clients an die entsprechenden Server weiter. Da der gesamte HTTP-Datenverkehr über den Proxy geleitet wird, kann dieser den Verkehr analysieren und gegebenenfalls blockieren. Dadurch lassen sich beispielsweise Virenscanner oder URL-Filter umsetzen. Ein Nachteil dieser Methode ist, dass nur der HTTP-Datenverkehr analysiert werden kann.

[2] Android Apps werden als APKs ausgeliefert. Ähnlich wie ein Installationsprogramm enthält eine APK-Datei alle Daten die zur Ausführung der App notwendig sind.

2.6. VPN

VPN steht für „Virtual Private Network“ und beschreibt eine Technik, die es erlaubt, ein virtuelles privates Kommunikationsnetz zu bauen, welches auf bestehenden physischen Kommunikationsnetzwerken aufbaut. Mittels VPN kann beispielsweise ein Mitarbeiter von Zuhause aus auf sein Firmennetz zugreifen als würde er direkt vor Ort sein. Der gesamte Verkehr wird, sofern nicht anders konfiguriert, über das VPN geleitet. Daher lässt sich auch diese Technik zur Blockierung von bestimmten URLs bzw. Werbung nutzen. Prinzipiell kann man auf Android zwei unterschiedlichen Szenarios zur Blockierung von Werbung mittels VPN unterschieden: Der Verwendung eines lokalen VPN Servers und der Verwendung eines externen VPN Servers. Zur Blockierung von Werbung eignen sich beide Fälle, jedoch bietet der lokale Server deutlich höhere Privatsphäre, da keine externen Dienste benötigt werden. Ein weiterer Vorteil von VPN-basierten Filtertechniken ist der Umstand, dass unter Android keine root-Rechte benötigt werden um eine VPN-Verbindung aufzubauen. Daher kann eine App, die einen VPN-Server simuliert auf den gesamten Netzwerkverkehr des gesamten Betriebssystems sowie installierter Apps zugreifen. Beispielswiese verwendet *Block This* diese Technologie auf Android Geräten um an den Netzwerkverkehr zu kommen, die DNS Server zu verändern und damit Werbung zu blockieren [15].

2.7. Browser

Eine weitere Möglichkeit um Werbung zu blockieren ist die Verwendung eines speziellen Browsers. Spezielle Browser verwenden meist die gleichen Ansätze, wie sie von Werbeblock-Erweiterungen in der Desktopwelt verwendet werden. Der Unterschied ist im Wesentlichen, dass die Funktionalität der Erweiterung direkt in den Browser integriert wird. Dadurch ergibt sich der Vorteil, dass keine root-Rechte benötigt werden, sowie kein zusätzliches Programm permanent im Hintergrund mitlaufen muss. Der Nachteil dieses Ansatzes ist, dass die Werbung nur in dem speziellen Browser blockiert wird und nicht auch in anderen Browsern und Apps. Beispielsweise verwenden *NoChromo* [16] und *Adblock Browser for Android* [17] diesen Ansatz. Mit *Firefox: privat+sicher surfen* [18] gibt es auch einen aus der Desktopwelt bekannten Browser mit Plugin Unterstützung für Android.

2.8. Vergleich

Tabelle 1 vergleicht die verschiedenen technischen Möglichkeiten Werbung auf Android Geräten zu blockieren. Es wird unterschieden ob Werbung nur in Browsern oder auch in Apps blockiert werden kann, ob root-Rechte benötigt werden, ob jeglicher Datenverkehr analysiert werden kann oder nur http-Verkehr und ob die Blockierung von einzelnen URLs möglich ist.

		Blockierungstechnik						
		IP-Adresse	**DNS**	**Hosts-Datei**	**Xposed**	**http-Proxy**	**VPN**	**Spezieller Browser**
Wirksamkeit	**Browser**	+	+	+	+	+	+	+
	Apps	+	+	+	+	+	+	-
	Funktioniert auf nicht gerooteten Geräten	-	-	-	-	+	+	+
	Filtert HTTP-Verkehr	+	+	+	+	+	+	+
	Filtert sonstigen IP-basierten Verkehr	+	+/-	+/.	+	-	+	-
	Blockierung von spezifischen URLs	-	-	-	+	+	+	+

Tabelle 1: Vergleich verschiedener Werbeblockierungstechniken für Android Systeme.

3. Implementierung

Die Blockierung von Werbung mittels eines VPNs stellt eine einfache und geräteunabhängige Methode dar, Werbung effizient, sowohl im Browser als auch in Anwendungen von Drittanbietern zu blockieren ohne dabei auf invasive Methoden wir dem Rooten von Geräten zurückzugreifen. Im Vergleich zu HTTP Proxies funktioniert diese Methode nicht nur für HTTP-Datenverkehr, sondern erlaubt den gesamten Datenverkehr zu analysieren und stellt somit eine ideale Methode für das

Blockieren von Werbung und von Schadsoftware auf Android Geräten dar. Zu Demonstrationszwecken und um eine möglichst große Adaptation zu erreichen, wurde dafür NetGuard [19] erweitert. NetGuard ist eine Firewall für Android, welche sich auf das Blockieren des Internetzugriffs von spezifischen Anwendungen konzentriert. Die Anwendung ist Open Source unter der GNU General Public License Version 3 und kann auch direkt im Google Play Store geladen werden. Konkret hat der Benutzer oder die Benutzerin die Möglichkeit, mithilfe von NetGuard den gesamten Datenverkehr des Smartphones zu blockieren oder bei Bedarf den Internetzugriff von spezifischen Anwendungen individuell zu steuern. NetGuard konzentriert sich hauptsächlich auf das Blockieren des Internetzugriffs von Anwendungen, bietet aber auch die Möglichkeit, spezifische Adressen via Hosts-Datei (siehe Kapitel 2.3) zu blockieren. Im Gegensatz zum herkömmlichen Ansatz werden hierfür allerdings keine Root Rechte benötigt, da die Hosts-Datei nicht im Betriebssystem sondern direkt im VPN-Dienst installiert wird. Vor dem Auflösen von DNS Anfragen wird anschließend im VPN-Dienst überprüft ob die Anfrage erlaubt ist. Wird der VPN-Dienst heruntergefahren findet somit allerdings auch keine Blockierung via Hosts-Datei statt. Im Prinzip kann NetGuard also bis zu einem gewissen Maß Werbung blockieren. Nachteil dieser Methode ist allerdings, dass diese Hosts-Dateien entweder manuell erstellt werden müssen oder aus anderen Quellen bezogen werden müssen und das nur komplette Domains blockiert werden können. Hinzu kommt, dass diese Listen aktuell gehalten werden müssen, um eine korrekte Funktionsweise zu garantieren. Das Blockieren von Werbung via Hosts-Dateien ist also meist nur erfahreneren Benutzern und Benutzerinnen zuzumuten. Um diesen Prozess zu vereinfachen, wurde im Rahmen dieses Projektes die Funktionalität von NetGuard um die Unterstützung von sogenannten AdBlock Plus Filter Listen [20] zur automatisierten Blockierung von Werbung erweitert. Die Funktionsweise respektive die Implementierung wird im Folgenden behandelt.

3.1. AdBlock Plus Filterlisten

AdBlock Plus ist ein Programm, welches von Eyeo GmbH entwickelt wird [21]. Primär beschränkt sich dessen Funktionsweise auf das Blockieren von Werbung in Browsern. Dies geschieht über so genannte Filterlisten, welche Informationen über die zu blockierenden Inhalte beinhalten. Die Struktur von Filterlisten ähnelt im wesentlichen stark jener von Regular Expressions. Im einfachsten Fall können gesamte „Uniform Resource Locators“ (kurz URLs), welche blockiert werden sollen (z.B. `http://www.example.org/example.png)`, direkt in der Filterliste hinterlegt werden. Filterlisten bieten allerdings auch die Möglichkeit, über spezielle Attribute gezielt Inhalte wie Bilder oder Skripte zu blockieren bzw. in einigen Fällen sogar über HTML- oder CSS-Attribute Elementen direkt aus Dokumenten zu entfernen. Der Vorteil von Filterlisten, im Vergleich zu Hosts-Dateien, liegt in deren Struktur. Während bei der Verwendung von Hosts-Dateien nur der Aufruf, also der gesamte Inhalt von Webseiten, blockiert werden kann, erlauben Filterlisten spezifische Elemente (z.B. Bilder) oder URLs zu blockieren. Außerdem werden Filterlisten in der Regel über ein Abonnement-Modell eingebunden. Damit können Änderungen, wie zum Beispiel das Hinzufügen von zu blockierenden Elementen, einfach und schnell umgesetzt werden. Dies garantiert eine zuverlässige Methode bei häufig wechselnden Werbeinhalten. Im Vergleich zu Hosts-Dateien können auch mehrere Filterlisten gleichzeitig abonniert, bzw. bei Bedarf deaktiviert werden. Durch die Kombination von mehreren verschiedenen Filterlisten kann eine granulare Einstellung an zu blockierenden Inhalten (beispielsweise Social Media, Tracking) erreicht werden. AdBlock Plus existiert in mehreren Varianten, unter anderem als Erweiterungen für alle modernen Browser, sowie als eigenständige Applikation mit integriertem Webbrowser. Als solches kann AdBlock Plus jedoch keine Werbeinhalte in Anwendungen von Drittanbietern blockieren.

3.1.1. Aufbau

Filterlisten für AdBlock Plus folgen in der Regel einem einfachen Prinzip. Für den allgemeinen Fall, können direkt gesamte URLs in der Filterliste eingetragen werden. Mittels Wildcards wie z.B. „*“ können Filterregeln zusätzlich erweitert und modifiziert werden und erlauben somit ein allgemeineres Filtern. Via Ausnahmeregeln, die mittels „@@“ definiert werden, können außerdem bestimmte URLs von Filtern ausgenommen werden. Im Grunde können mehrere Konzepte für das Filtern von Webseiten angewandt werden. Einerseits via Teile aus der URL (z.B. `/banner/` oder `/ad/`), andererseits via Domainnamen (z.B. `||ads.example.org`). Zusätzliche Optionen, welche mit „$“ definiert werden (z.B. `$script,image`), erlauben außerdem eine feinere Abstimmung von Filteroptionen. Filterlisten unterstützen zusätzlich die Verwendung von Regular Expressions. Im

Vergleich zu Hosts-Dateien können über Filterlisten also genauere Regeln definiert werden. Abbildung 1 zeigt einen Ausschnitt einer Filterliste.

```
[Adblock Plus 2.0]
! Checksum: ARyUvHZRDVzRLpLJlcy9yg
! Version: 201707190940
! The contents of this filter list are fetched from the EasyList repository: https://github.com/easylist/easylist
! Title: EasyList
! Last modified: 19 Jul 2017 09:40 UTC
! Expires: 2 days (update frequency)
! Homepage: https://easylist.to/
! Licence: https://easylist.to/pages/licence.html
!
! Please report any unblocked adverts or problems
! in the forums (https://forums.lanik.us/)
! or via e-mail (easylist.subscription@gmail.com).
!
!-----------------------General advert blocking filters-----------------------!
! *** Fetched from: https://raw.githubusercontent.com/easylist/easylist/master/easylist/easylist_general_block.txt ***
&act=ads_
&ad.vid=$~xmlhttprequest
&ad_box_
&ad_channel=
&ad_classid=
&ad_height=
&ad_ids=
&ad_keyword=
&ad_network_
&ad_number=
&ad_revenue=
&ad_slot=
&ad_time=
&ad_type=
&ad_type_
&ad_url=
&ad_zones=
&adbannerid=
&adclient=
&adcount=
&adflag=
&adgroupid=
&admeld_
```

Abbildung 1: Filterliste für AdBlock Plus

3.2. Funktionsweise

Die in NetGuard implementierte Methodik zur Blockierung von Netzwerkverkehr beschränkt sich primär auf das Blockieren von DNS-Anfragen bzw. auf das Blockieren von Netzwerkanfragen. Damit werden blockierte Verbindungen erst gar nicht aufgebaut, was wiederrum den gesamten Energieverbrauch minimiert. Vor dem Aufbau einer Verbindung wird geprüft, ob die aufgerufene Domain in der Hosts-Datei eingetragen ist. Ist dies der Fall, wird der Verbindungsaufbau terminiert. Diese Vorgehensweise funktioniert aber nur zum Teil mit AdBlock Plus Filterlisten. Die spezielle Eigenschaft, gezielt Inhalte statt lediglich Domains zu blockieren, erfordert einige Änderungen in der Architektur von NetGuard. Die implementierten Änderungen werden in Folge diskutiert.

3.2.1. Abonnieren von Filterlisten

Das Aktivieren der AdBlock Funktion in NetGuard passiert in zwei Schritten: Im ersten Schritt wird die erweiterte Option „Datenverkehr filtern“ aktiviert (siehe Abbildung 2). Damit kann sichergestellt werden, dass die neu hinzugefügten Filterregeln angewandt werden können. Im zweiten Schritt wird das neu hinzugefügte AdBlock Feature, bzw. in Folge auch die benötigten Filterlisten aktiviert. Eine Vorauswahl häufiger Filterlisten wurde bereits zu Demonstrationszwecken hinzugefügt. Um eine Filterliste zu aktivieren bzw. zu deaktivieren muss demnach nur die jeweilige Liste ausgewählt werden und der jeweilige Schalter betätigt werden (siehe Abbildung 3).

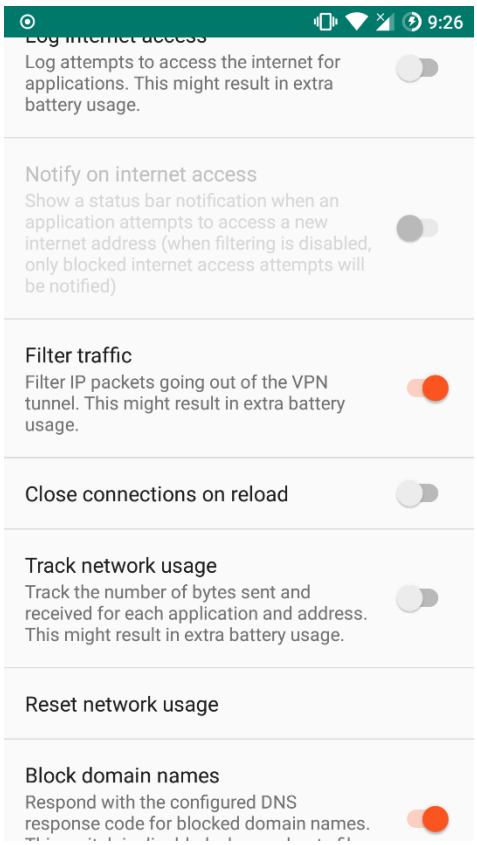

Abbildung 2: Aktivieren der erweiterten Option "Datenverkehr filtern"

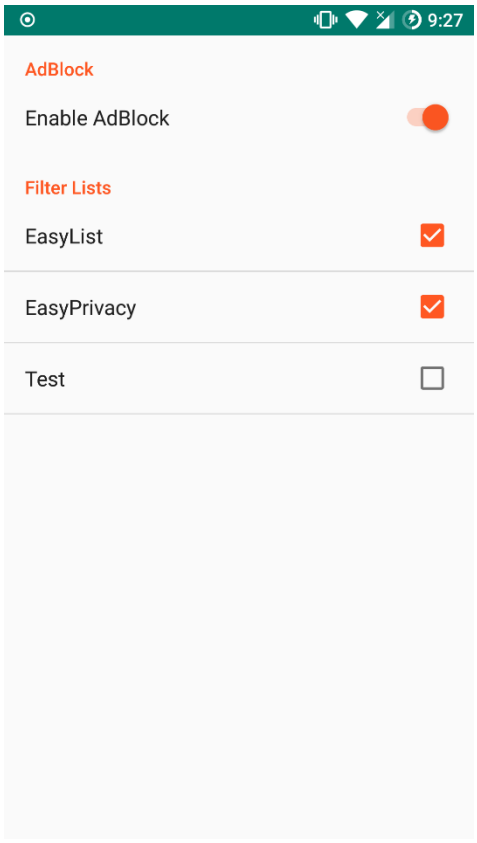

Abbildung 3: Aktivieren von AdBlock in NetGuard

3.3. Blockieren von spezifischen Inhalten

Ein großer Vorteil bei der Verwendung von AdBlock Plus Filterlisten liegt in der Möglichkeit, Inhalte gezielt zu blockieren. Dazu zählt beispielsweise das Blockieren von Bildern oder Skripten. Um diese

Funktionsweise in NetGuard zu implementieren, sind allerdings einige Anpassungen notwendig. Die vorhandene Infrastruktur zur Blockierung von Netzwerkverkehr kann nur zum Teil übernommen werden und bedarf einiger Adaptierungen. Dazu zählt beispielsweise die gezielte Analyse von Netzwerkpaketen, um die zuvor importieren Filterlisten respektive Filterregeln anwenden zu können. Das Blockieren von Werbung findet schlussendlich in zwei Schritten statt: Im ersten Schritt werden, analog zur vorhandenen Infrastruktur, Domainnamen bzw. DNS oder Netzwerkanfragen mit vorhandenen Filterregeln abgeglichen. Ist eine Regel anwendbar, so wird die Anfrage terminiert. Im zweiten Schritt werden ausgehende Netzwerkpakete im Detail analysiert und ebenfalls potentielle Filterregeln angewandt. Bei Übereinstimmung einer Filterregel mit einem ausgehenden Datenpaket wird diese Verbindung ebenfalls blockiert.

3.4. Limitierungen

Zwar bieten AdBlock Plus Filterlisten im Prinzip die Möglichkeit, nur gewisse Inhalte (z.B. Scripts oder Bilder von Webseiten) zu blockieren, allerdings wird diese Möglichkeit durch die Verwendung von VPN Diensten teilweise unterbunden. AdBlock Plus Filterliste sind speziell auf Werbung in Browsern ausgerichtet. Dieser Umstand wird vor allem deutlich, wenn man die Möglichkeit zum Blockieren von Elementen untersucht. Zum Beispiel können über Attribute Modifier alle Tabellen mit einem gewissen Attribut (z.B. „min width 80%“) blockiert bzw. ausgeblendet werden. Um ein ähnliches Verhalten über einen VPN Dienst zu verwirklichen ist es notwendig jede einzelne Verbindung zu inspizieren und diese Anfragen dementsprechend zu blockieren und gegebenenfalls sogar zu modifizieren. Hinzu kommt die Tatsache, dass bei der Verwendung von HTTPS, Verbindungen aufgebrochen werden müssen um dasselbe Verhalten zu bekommen. Vor allem bei Applikationen, die auf Certificate Pinning, also dem hinterlegen von Zertifikatsinformationen des Servers (z.B. Fingerabdruck) in der Applikation selbst setzen, kann dies dazu führen, dass Applikationen nicht mehr voll funktionsfähig sind.

3.5. Filter Matching

Zu Demonstrationszwecken wurde zuerst ein trivialer Algorithmus zum Filter Matching implementiert. Dieser kann einen passenden Filter zu einer Anfrage in *O(n)* Zeit finden. Es zeigte sich jedoch schnell, dass dies für größere Filterlisten viel zu langsam ist und die Performance des Gerätes stark beeinträchtigt wird. Wladimir Palant zeigt jedoch, dass die Performance, des Matching Algorithmus mittels modifiziertem Boyer-Moor Algorithmus auf *O(n/m)* im Best-Case bzw. *O(n)* im Worst-Case verbessert werden kann [22]. Diese Verbesserung wurde zu Demonstrationszwecken ebenfalls implementiert und hat eine deutliche Performanceverbesserung zur Folge. Diese ist gerade bei mobilen Endgeräten wichtig, da mobile Endgeräte meist über geringere Ressourcen als Desktopsysteme verfügen.

3.6. Ergebnis

Um die Wirksamkeit des Demonstrators zu verifizieren wurden verschiedene Webseiten mit aktiviertem und mit deaktivierten AdBlock Plus Filterlisten aufgerufen. Abbildung 1 und Abbildung 2 zeigen zwei dieser Webseiten. Auf den linken Bildern ist die Seite mit deaktivierten AdBlock Plus Filterlisten zu sehen, rechts mit aktivierten Filterlisten. Es ist gut erkennbar, dass die Werbung in beiden Fällen erfolgreich blockiert wurde. Die Webseite „derstandard.at“ ist ein gutes Beispiel, warum Werbeblockierung auf DNS-Ebene nicht ausreicht, da die Werbung von der „derstandard.at“-Domain geladen wird und somit nur eine komplette Blockierung der gesamten Seite aber nicht von einzelnen URLs (der Werbung) möglich wäre.

Abbildung 4: DerStandard.at, links mit deaktivierten AdBlock Filterlisten, rechts mit aktivierten.

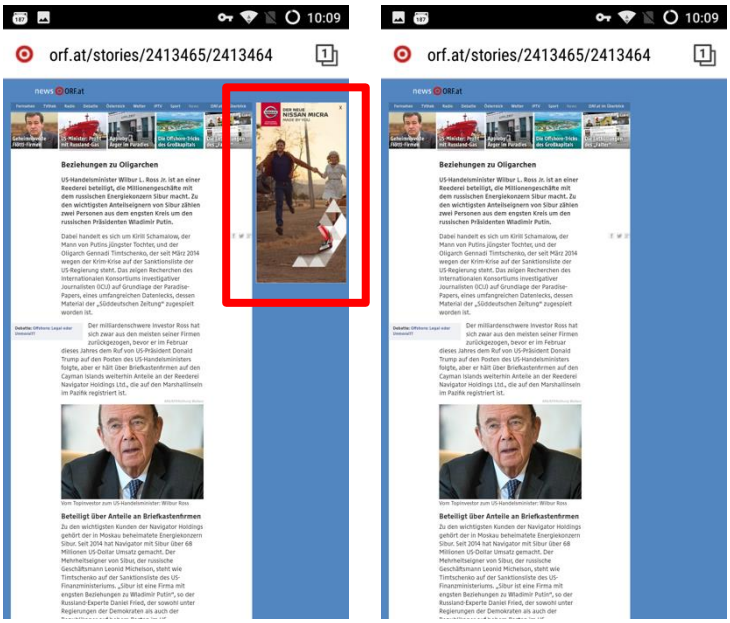

Abbildung 5: ORF.at, links mit deaktivierten AdBlock Filterlisten, rechts mit aktivierten.

4. Fazit

Es gibt mehrere Ansätze, die die Blockierung von Werbung unter Android ermöglichen. Je nachdem welche Anforderungen an den Werbeblocker gestellt werden eignen sich unterschiedliche Techniken bzw. Kombinationen dieser, um einen Blocker umzusetzen. In diesem Projekt wurde ein VPN-basierter Ansatz gewählt, da hierfür keine root-Rechte notwendig sind und dieser Ansatz die Umsetzung von DNS und IP-Sperren sowie die Blockierung von spezifischen URLs unterstützt. Konkret wurde eine bestehende Firewall App (NetGuard) erweitert um die Blockierung von Werbung und von Malware auf Basis von AdBlock Plus Filterlisten zu ermöglichen. Die praktische Evaluierung des Demonstrators zeigte dessen Wirksamkeit.

Referenzen

[1] J. Ihlenfeld, „Fast 1 Milliarde Euro für Online-Werbung in Deutschland," 7 1 2008. [Online]. Available: https://www.golem.de/0801/56835.html. [Zugriff am 11 07 2017].

[2] Statista 2017, „Umsätze mit Onlinewerbung in Deutschland in den Jahren 2005 bis 2020* (in Millionen Euro)," [Online]. Available: https://de.statista.com/statistik/daten/studie/165473/umfrage/umsatzentwicklung-von-onlinewerbung-seit-2005/. [Zugriff am 11 07 2017].

[3] Trend Micro, „New York Times pushes Fake AV malvertisement," 2009. [Online]. Available: http://countermeasures.trendmicro.eu/new-york-times-pushes-fake-av-malvertisement/. [Zugriff am 11 07 2017].

[4] https://twitter.com/nytimes/status/3958547840, „Attn: NYTimes.com readers: Do not click pop-up box warning about a virus -- it's an unauthorized ad we are working to eliminate.," 13 09 2009. [Online]. Available: https://twitter.com/nytimes/status/3958547840. [Zugriff am 11 07 2017].

[5] Futurezone, „Die beiden Internetportale kinox.tv und movie4k.tv werden seit einigen Tagen von österreichischen Providern blockiert.," 20 12 2016. [Online]. Available: https://futurezone.at/netzpolitik/oesterreichische-provider-setzen-neue-netzsperren-um/236.835.868. [Zugriff am 07 11 2017].

[6] delta_foxtrot2, „AdFree for Android," 2013. [Online]. Available: https://forum.xda-developers.com/showthread.php?t=2252747. [Zugriff am 11 07 2017].

[7] mrRobinson, „AdAway," 2013. [Online]. Available: https://forum.xda-developers.com/showthread.php?t=2190753. [Zugriff am 11 07 2017].

[8] BSDgeek_Jake, „MoaAB: Mother of All AD-BLOCKING > BLOCKS ADware Malware Spyware Bloatware Ransomware," 2012. [Online]. Available: https://forum.xda-developers.com/showthread.php?t=1916098. [Zugriff am 11 07 2017].

[9] Paget96, „AdBlocker," 2016. [Online]. Available: https://forum.xda-developers.com/android/software-hacking/ads-adblocker-v1-0-t3283855. [Zugriff am 11 07 2017].

[10] P. D. Ace, „[GUIDE][COLLECTION] All About Ad-Blocking," 2015. [Online]. Available: https://forum.xda-developers.com/android/general/guide-ad-blocking-t3218167. [Zugriff am 11 07 2017].

[11] rovo89, „Xposed - General info, versions & changelog," 2012. [Online]. Available: https://forum.xda-developers.com/xposed/xposed-installer-versions-changelog-t2714053. [Zugriff am 11 07 2017].

[12] DragonHunt3r, „[Xposed Module] Xad - Block the google Ad API," 2013. [Online]. Available: https://forum.xda-developers.com/xposed/modules/xposed-module-xad-block-google-ad-api-t2580344. [Zugriff am 11 07 2017].

[13] FatMinMin, „[Xposed] MinMinGuard," 2014. [Online]. Available: https://forum.xda-developers.com/xposed/modules/xposed-minminguard-v1-7-0-cancelled-t2597332. [Zugriff am 11 07 2017].

[14] wanam, „[APP][2.3+][Xposed][Youtube AdAway]," 2013. [Online]. Available: https://forum.xda-developers.com/xposed/modules/app-t2547316. [Zugriff am 11 07 2017].

[15] S. Georgiev, „Block This - a DNS based Ad Blocker for Android," [Online]. Available: https://github.com/ggsava/block-this. [Zugriff am 10 11 2017].

[16] nochromo, „#NoChromo - A wild ad-blocking browser appears," 2015. [Online]. Available: https://forum.xda-developers.com/android/apps-games/app-nochromo-wild-browser-appears-t3130776. [Zugriff am 12 07 2017].

[17] Eyeo GmbH, „Adblock Browser for Android," 2016. [Online]. Available: https://play.google.com/store/apps/details?id=org.adblockplus.browser&hl=en. [Zugriff am 12 07 2017].

[18] Mozilla, „Firefox: privat+sicher surfen," [Online]. Available: https://play.google.com/store/apps/details?id=org.mozilla.firefox. [Zugriff am 12 07 2017].

[19] M. Bokhorst, „NetGuard - A simple way to block access to the internet per application,“ 2017. [Online]. Available: https://www.netguard.me. [Zugriff am 18 07 2017].

[20] Eyeo GmbH, „Bekannte Filterlisten für Adblock Plus,“ 2017. [Online]. Available: https://adblockplus.org/de/subscriptions. [Zugriff am 18 07 2017].

[21] Eyeo GmbH, „Adblock Plus - Für ein Web ohne nervige Werbung!,“ 2017. [Online]. Available: https://adblockplus.org/de/. [Zugriff am 18 07 2017].

[22] W. Palant, „Investigating filter matching algorithms,“ 22 08 2006. [Online]. Available: https://adblockplus.org/blog/investigating-filter-matching-algorithms. [Zugriff am 15 07 2017].