VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ
DEPARTMENT OF TELECOMMUNICATIONS

ZABEZPEČENÍ MOBILNÍCH ZAŘÍZENÍ S OPERAČNÍM SYSTÉMEM ANDROID VE FIREMNÍ SFÉŘE
SECURITY OF MOBILE DEVICES RUNNING ANDROID IN A CORPORATE ENVIRONMENT

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE — **David Pecl**
AUTHOR

VEDOUCÍ PRÁCE — **Ing. Jakub Frolka**
SUPERVISOR

BRNO 2018

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Bakalářská práce

bakalářský studijní obor **Informační bezpečnost**
Ústav telekomunikací

Student: David Pecl ***ID:*** 185940
Ročník: 3 ***Akademický rok:*** 2017/18

NÁZEV TÉMATU:

Zabezpečení mobilních zařízení s operačním systémem Android ve firemní sféře

POKYNY PRO VYPRACOVÁNÍ:

Práce je zaměřena na bezpečnostní aspekty operačního systému Android. Věnuje se hrozbám a rizikům, které můžeme na zařízeních s OS Android najít a mechanismům ochrany před nimi. Zaměřuje se na zabezpečení mobilních zařízení a navrhuje možnosti opatření tak, aby splňovaly požadavky firemní sféry. Cílem práce je vytvořit dokument s nejčastějšími hrozbami na OS Android a možnostmi, jak tyto hrozby eliminovat, včetně popisu bezpečnostních mechanismů OS Android, a vytvořit laboratorní úlohu pro demonstraci dané problematiky.

DOPORUČENÁ LITERATURA:

[1] ELENKOV, Nikolay. Android Security Internals: An In-Depth Guide to Android's Security Architecture. San Francisco: No Starch Press, 2014. ISBN 9781593275815.

[2] STALLINGS, William. Cryptography and network security: principles and practices. 4th ed. Upper Saddle River: Pearson Prentice Hall, 2006, 680 s. ISBN 0-13-187316-4.

Termín zadání: 5.2.2018 ***Termín odevzdání:*** 29.5.2018

Vedoucí práce: Ing. Jakub Frolka
Konzultant: Ing. Maroš Barabas Ph.D., maros.barabas@aec.cz, AEC a. s.

prof. Ing. Jiří Mišurec, CSc.
předseda oborové rady

Abstrakt

Práce se zabývá problematikou bezpečnosti operačního systému Android. Nejdříve popisuje základní architekturu systému a bezpečnostní mechanizmy, které můžeme v tomto systému najít.

Ve druhé části se věnuje popisu hrozeb a útoků na zařízení s OS Android. Popisuje riziko, kterému je vystaven uživatel mobilního zařízení a dopad na bezpečnost zařízení, uživatele a dat, která jsou v zařízení obsažena. U každé hrozby i útoku je zmíněn způsob, jakým může být zařízení kompromitováno. Hrozby i útoky jsou ohodnoceny pomocí systému CVSS. Dále se věnuje problematice aktualizací na Androidu.

V poslední části jsou popsány aplikace, které mohou být použity především ve firemním prostředí pro správu a zabezpečení mobilních zařízení s Androidem. Jsou zde popsány funkce těchto aplikací a proti jakým hrozbám nebo útokům poskytují ochranu. Dále jsou zde uvedeny příklady takových aplikací.

V rámci práce je vytvořena laboratorní úloha pro demonstraci systémů pro správu mobilních zařízení, které se používají ve firemní sféře, a učební text formou prezentace jak pro přednášení, tak pro samostudium.

Klíčová slova

Android, bezpečnost, hrozba, mobilní malware, ochrana dat, správa mobilních zařízení, Google

Abstract

The thesis deals with the security of the Android operating system. Firstly, it describes the basic architecture of the system and the security mechanisms we can find in this system, namely Linux kernel, application sandboxing, and application permissions.

In the second part, it describes threats and attacks on the Android devices. Describes the risks to which the users are exposed and the impact on the device, user, and data security. For each threat and attack, the way the device can be compromised is mentioned. Threats and attacks are rated using CVSS. It also deals with Android updates.

The last section describes applications that can be used to manage and secure Android mobile devices primarily in the corporate environment. There are described features of these applications and against what threats or attacks provide protection. Furthermore, there are examples of such applications.

The thesis also provides a laboratory task that is created to demonstrate the mobile device management systems that are used in the corporate environment. Also, study material is created in the form of presentation for both lecture and self-study.

Keywords

Android, security, threat, mobile malware, data protection, mobile device management, Google

Bibliografická citace:

PECL, D. Zabezpečení mobilních zařízení s operačním systémem Android ve firemní sféře. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2018. 67 s. Vedoucí bakalářské práce Ing. Jakub Frolka.

Prohlášení

„Prohlašuji, že svou závěrečnou práci na téma Zabezpečení mobilních zařízení s operačním systémem Android ve firemní sféře jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a konzultanta bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.“

V Brně dne 29. května 2018 .....

podpis autora

Poděkování

Děkuji vedoucímu bakalářské práce, Ing. Jakubu Frolkovi, za cennou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce. Dále bych chtěl poděkovat firmě AEC, a. s., která mi poskytla možnost zpracovávat bakalářskou práci v jejich prostředí pod dohledem zkušených bezpečnostních konzultantů, kteří mi poskytli mnoho odborných rad. Také bych chtěl poděkovat konzultantovi bakalářské práce, Ing. Maroši Barabasovi, Ph.D., který mi poskytoval jak odborné, tak pedagogické rady.

V Brně dne 29. května 2018 .....

podpis autora

Obsah

Seznam obrázků

Seznam tabulek

Úvod

Bezpečnost mobilních zařízení je v dnešní době velmi podceňované téma. Můžeme to vidět například na rozdílu mezi zabezpečením počítačů a mobilních telefonů, téměř 88 % uživatelů chrání svůj počítač bezpečnostním softwarem, zatímco chytrý mobilní telefon chrání jen 53 % uživatelů [1]. Chytrý telefon je považován za nejslabší článek v celé bezpečnosti informačních technologií [2]. Rozšířenou hrozbou na mobilních zařízeních je malware zaměřující se na odcizení bankovních údajů, tzv. mobile banking trojan. Častý je také útok pomocí mobilního ransomware [3].

Pro uživatele je jednoduché zabezpečit svůj mobilní telefon, stačí dodržovat několik základních pravidel a mít nainstalovaný a aktualizovaný bezpečnostní software. Ve firemním prostředí je zabezpečení telefonu se systémem Android náročnější proces z několika důvodů. Prvním je, že administrátoři potřebují systém, který mohou spravovat centrálně, ideálně pomocí jedné konzole tak, aby nemuseli mít k zařízení fyzický přístup. Druhým důvodem je stále více se uplatňující politika BYOD[1], kdy si uživatelé do práce nosí vlastní zařízení, na kterém pracují. Netýká se to pouze notebooků, ale právě i chytrých telefonů. To přináší administrátorům nebo bezpečnostním správcům problémy, protože už nemohou zařízení spravovat tak, jak byli zvyklí. Nemohou zaměstnanci přikázat, jaké aplikace si do telefonu může instalovat a jaké ne, dále se potýkají s problémem ochrany soukromí, protože ve své centrální správě nemohou sledovat vše, co se na telefonu děje a často zaměstnanci také nesouhlasí s lokalizací telefonu z důvodu obavy ze sledování jejich pohybu [4].

V rámci této práce budou analyzovány hrozby a útoky na mobilní telefony se systémem Android tak, aby byla nalezena ideální forma zabezpečení těchto smartphonů, které používají zaměstnanci ve firmách, ať už se jedná o firemní nebo BYOD zařízení. Bude také kladen důraz na možnost centrální správy zabezpečení. Čtenář získá přehled o bezpečnostních funkcích, které poskytují různé aplikace určené pro zabezpečení mobilních zařízení.

V druhé kapitole je čtenáři představen OS Android, jeho architektura a verze. Třetí kapitola popisuje bezpečnostní mechanizmy jako například sandboxing aplikací, podepisování aplikací nebo zabezpečený booting. Následuje popis jednotlivých hrozeb a útoků na uživatele nebo na zařízení s Androidem a v páté kapitole jsou uvedeny bezpečnostní aplikace sloužící k ochraně mobilních zařízení s OS Android. Šestá kapitola pojednává o v současnosti hodně zmiňovaném tématu – problému zajištění aktuálního systému Android a vydávání aktualizací.

Cílem práce je poskytnout čtenáři úvod do problematiky bezpečnosti Androidu, popis útoků a hrozeb na zařízení s tímto systémem a popsat a doporučit ochranu takových zařízení. Praktickým výstupem je laboratorní práce pro použití v počítačových cvičeních a prezentace vytvářená jako studijní materiál pro použití ve výuce.

[1] BYOD = „bring your own device“, firemní zaměstnanci využívají svůj vlastní telefon ke správě firemního e-mailu a firemních dat. Za bezpečnost zařízení zodpovídá uživatel. Administrátoři nebo bezpečnostní správci nemají k zařízení přístup.

1 Mobilní operační systém Android

1.1 Úvod do Androidu

Android je operační systém, který byl primárně určený pro mobilní telefony. S pozdější dobou se z mobilního operačního systému stala platforma, kterou nyní můžeme najít také na tabletech, nositelných zařízeních jako jsou chytré hodinky nebo náramky, v televizích, v autech nebo třeba chytrých domácnostech[2]. Systém je uzpůsoben na ovládání dotykem, ale vzhledem k rozšíření do dalších typů zařízení je možné jej ovládat také hlasem, pohybem nebo různými ovladači. Jako většina ostatních operačních systémů i Android podporuje instalaci aplikací třetích stran. Ty je možné stahovat primárně z obchodu Google Play.

Android je open-source projekt, původně od firmy Android Inc. Tuto firmu v roce 2005 koupila společnost Google, která také založila konsorcium firem vyvíjejících otevřené standardy pro mobilní zařízení – Open Handset Alliance, zkráceně OHA. Tato aliance od svého vzniku zaštiťuje vývoj operačního systému Android.

Operační systém Android je rozdělen do několika verzí. Posledních sedm let je pravidelně na podzim představena nová verze tohoto operačního systému, která vždy opravuje chyby minulé verze, rozšiřuje funkčnost systému nebo zlepšuje zabezpečení. Prozatím nejnovější verze systému (k září 2017) byla uvedena 21. srpna 2017 jako Android 8.0 Oreo.

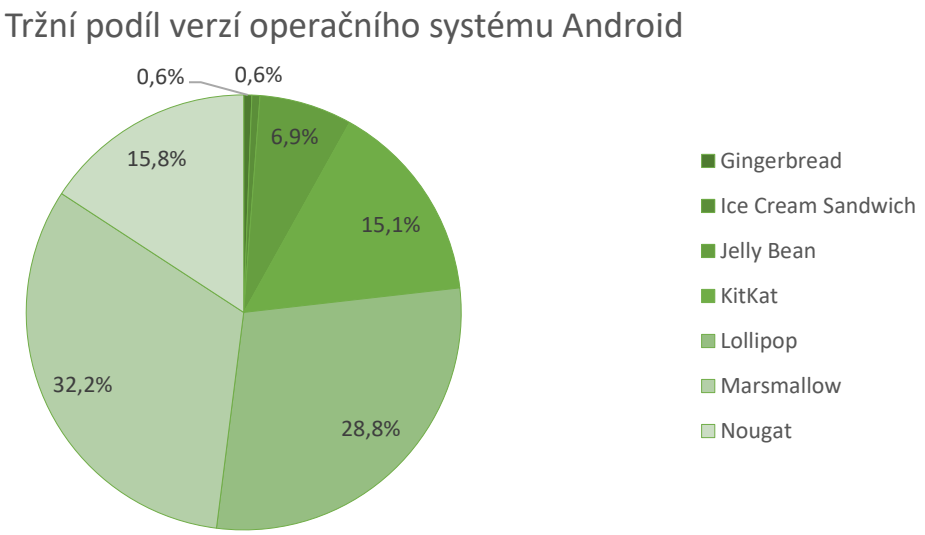
1.2 Tržní podíl OS Android

Pokud se podíváme na trh mobilních operačních systémů, posledních několik let je Android jasnou jedničkou. V srpnu 2017 měl Android 72,74% podíl a jasně trhu dominoval [5]. V dubnu 2017 se dokonce dostal na první místo na trhu všech operačních systému, tedy zahrnujícím jak mobilní telefony, tak počítače, a předběhl desktopový operační systém Windows [6].

Tabulka 1.1: Tržní podíl jednotlivých verzí OS Android [7]

Verze	Označení	API	Tržní podíl
2.3.3–2.3.7	Gingerbread	10	0,6 %
4.0.3–4.0.4	Ice Cream Sandwich	15	0,6 %
4.1.x	Jelly Bean	16	2,4 %
4.2.x		17	3,5 %
4.3		18	1,0 %
4.4	KitKat	19	15,1 %
5.0	Lollipop	21	7,1 %
5.1		22	21,7 %
6.0	Marsmallow	23	32,2 %
7.0	Nougat	24	14,2 %
7.1		25	1,6 %

[2] Seznam všech zařízení, která podporují Google Play, má 278 stránek.



Obrázek 1.1: Graf tržního podílu verzí OS Android [7]

1.3 Verze OS Android

První verze Androidu 1.0 označovaná jako Alpha byla uvedena na trh v roce 2008. První komerční telefon s Androidem byl HTC Dream. V roce 2009 vyšla verze 1.5, která byla poprvé označena podle sladkosti, dostala název Cupcake. Následovaly další verze, každá nesla pojmenování po sladkosti, která začínala na následující písmeno v abecedě[3].

V současnosti (září 2017) je stále nejpoužívanější verzí Android 6.0 Marshmallow [5]. Tato verze byla vydána 5. října 2015 a přesto, že má již dva roky, je nainstalována na téměř třetině ze všech mobilních telefonů používajících Android. Aktuálně nejstarší podporovaná verze je Android 4.4 KitKat, která je na 15,1 % zařízení. Zařízení, které obsahují starší verze (tedy již nepodporované) zastupují 8,1 % veškerých mobilních telefonů s Androidem. Tržní podíl jednotlivých verzí ukazuje Tabulka 1.1.

Aktuálnost operačního systému je velký problém. Pokud to shrneme, 8 % telefonů s Androidem používá nepodporovanou verzi operačního systému a dalších 75 % běží na dva roky nebo více staré verzi. V obchodech můžete stále narazit na mobilní telefony se systémem 5.0 Lollipop, který je již více než tři roky starý. Tohoto problému si je vědom také Google, který se rozhodl pozměnit architekturu systému (ve verzi 8.0 Oreo) tak, aby bylo pro výrobce telefonů jednodušší aktualizovat zařízení (více viz kapitola 5).

Téměř s každou novou verzí Androidu přišlo zlepšení zabezpečení. Níže jsou uvedeny verze, které přidávaly významné funkce v oblasti bezpečnosti:

- 2.2 Froyo: podpora číselného PINu a alfanumerického hesla k odemčení telefonu;
- 3.0 Honeycomb: možnost šifrovat všechna uživatelská data v telefonu[4];

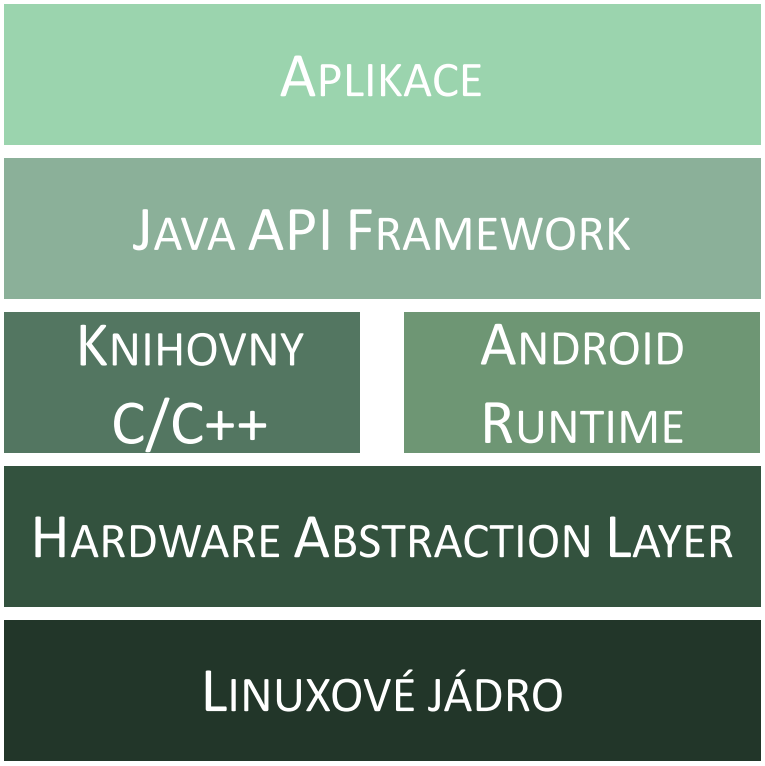
[3] Jediné dvě verze, 4.4 KitKat a 8.0 Oreo, nesou název podle komerčního produktu. Všechna ostatní pojmenování jsou obecné.

[4] Funkce přidána jako politika pro aplikace, pomocí kterých administrátoři spravují zařízení

- 4.0 Ice Cream Sandwich: možnost odemčení telefonu rozpoznáním obličeje (Face Unlock);
- 4.3 Jelly Bean: posílení sandboxování aplikací díky využití technologie SELinux[5];
- 4.4 KitKat: funkce zabezpečeného bootingu zařízení a kontrola integrity zařízení;
- 6.0 Marshmallow: přidáno API pro autentizaci pomocí otisku prstu, oprávnění aplikací lze měnit až po instalaci aplikace[6];
- 8.0 Oreo: Projekt Treble mění architekturu systému a umožňuje výrobcům zjednodušit aktualizaci zařízení, služba Google Play Protect zajišťující ochranu před malwarem a při ztrátě zařízení.

1.4 Architektura systému

Architekturu operačního systému Android můžeme rozdělit do pěti vrstev, viz Obrázek 1.2. Tou nejnižší je linuxové jádro, nad kterým můžeme najít hardwarovou abstraktní vrstvu (z anglického *hardware abstraction layer*, dále jen HAL). Linuxové jádro je srdcem systému a zajišťuje základní funkce systému. HAL je vrstva, která poskytuje přístup k hardwaru z vyšších vrstev. Nad vrstvou HAL se nachází knihovny jazyka C/C++ a prostředí Android Runtime (ART). Díky ART může každá aplikace běžet ve svém vlastním procesu. Druhou nejvyšší vrstvou je Java API Framework. Tato vrstva poskytuje rozhraní pro vývojáře a umožňuje jim využívat knihovny jazyku Java. Na vrcholu této pomyslné pyramidy jsou pak samotné aplikace.



Obrázek 1.2: Schéma architektury OS Android

[5] Původně představen již ve verzi 4.2, ale až ve verzi 4.3 ve výchozím nastavení zapnutý

[6] Do té doby byla oprávnění schvalována při instalaci aplikace a poté se již nedala měnit

1.4.1 Linuxové jádro

Nejnižší vrstvou operačního systému Android je linuxové jádro (Android používá standardně verzi 2.6), které zajišťuje základní funkčnost systému. Jeho hlavní součástí jsou ovladače, které provádí komunikaci mezi hardwarovou vrstvou a vrstvami vyššími. Dále jádro zajišťuje správu procesů, paměti, napájení, síťového spojení apod. K jádru samotnému je přidáno několik doplňků, které jsou potřebné pro mobilní platformy. Patří mezi ně například Power Manager, který umožňuje zvýšit životnost baterie, Low Memory Killer, který se stará o ukončování procesů a přidělování alokované paměti nebo Binder IPC driver (IPC = interprocess communication), který umožňuje komunikaci mezi jednotlivými procesy.

1.4.2 Hardware Abstraction Layer

Vrstva hardware abstraction layer (HAL) je rozhraní, které umožňuje vyšším vrstvám využívat hardwarové prostředky. HAL se skládá z několika modulů, kdy každý implementuje rozhraní pro konkrétní typ hardwarové komponenty. Pokud vyšší vrstva požádá o přístup k nějakému hardwarovému prostředku, HAL poskytne knihovny pro práci s touto komponentou.

Tím, že vrstva HAL poskytuje své vlastní knihovny pro práci s hardwarem, není zapotřebí zasahovat do vyšších vrstev. Výrobce zařízení si vytvoří vlastní knihovny pro práci se svým hardwarem a tyto implementuje do HAL vrstvy. Vyšší vrstvy jsou tedy nezávislé na hardwaru, na kterém běží.

1.4.3 Knihovny jazyka C/C++

Některé ze základních komponent platformy Android jsou psány takovým způsobem, že ke svému chodu vyžadují knihovny jazyka C/C++. Tyto knihovny tvoří část další vrstvy, společně s ART. Navíc pomocí Java API Framework mohou vývojáři některé z těchto knihoven využívat také ve svých aplikacích.

1.4.4 Android Runtime

Druhou částí vrstvy je Android Runtime (ART) prostředí. To umožňuje spouštět každou aplikaci zvlášť ve svém vlastním procesu. Každá aplikace je samostatnou instancí ART. Logika je velmi podobná s klasickou desktopovou Javou, kde programy běží v tzv. Java Virtual Machine (JVM). Dříve se na Androidu používal Dalvik Virtual Machine. Dalvik byl ale nahrazen modernějším ART. Pokud aplikace běží pod prostředím ART, pak by měla běžet i pod starší verzí Dalvik. Obrácená kompatibilita ale není zajištěna.

1.4.5 Java API Framework

Android poskytuje vývojářům přístup ke službám systému pomocí knihoven napsaných v jazyce Java. Tyto knihovny jsou shromážděny v balíčku, tzv. API. Vývojáři mohou tyto knihovny implementovat do svých aplikací a ty tak budou mít přístup například k ovládacím tlačítkům, notifikační liště, obsahu jiných aplikací a podobně. Každá verze operačního systému

pracuje s jiným API, například verze 1.0 pracovala s API 1, kdežto verze 7.0 Nougat pracuje s API 24[7].

1.4.6 Vrstva aplikací

Nejvyšší vrstvou je aplikační vrstva, která obsahuje všechny aplikace na zařízení nainstalované, ať už se jedná o předinstalované aplikace nebo stažené z Google Play. Aplikace na Androidu fungují dvěma způsoby. První je, že poskytují uživatelské rozhraní, pomocí kterého uživatel aplikaci ovládá a využívá. Druhý způsob je, že poskytují možnosti, jak mohou ostatní vývojáři pracovat s touto aplikací. Například pokud bude vývojář chtít, aby jeho aplikace doručovala SMS zprávy, nemusí tuto funkci implementovat znova, ale může využít jinou SMS aplikaci, která již je nainstalována v zařízení a nabízí svou funkci ostatním aplikacím.

[7] V dnešní době se při programování aplikací pro Android považuje za nejnižší API verze 19 pro Android 4.4 Kitkat. Pokud vývojář využije tuto verzi, má zajištěno, že jeho aplikace poběží na Androidu 4.4 a novější, což činí více než 90 % všech zařízení (září 2017).

2 Zabezpečení OS Android

Uživatelé si už zvykli, že na svých stolních počítačích nebo noteboocích by měli mít nainstalovaný nějaký software, který jim poskytuje zabezpečení. Ať už je to obyčejný antivirový program, pokročilé anti-malwarové řešení nebo kompletní balíček bezpečnostních funkcí, operační systém musí být chráněn. Běžní uživatelé si už ale neuvědomují, že zabezpečit potřebuje také jejich mobilní telefon, ať už běží na jakékoliv platformě. Přitom dost často se jedná o stejné riziko, jako by nechali nezabezpečený počítač. Mobilní telefon mohou využívat ke čtení e-mailů, k procházení webu, spravování bankovního účtu nebo v něm mohou mít uložena jiná cenná data, ať už osobní nebo firemní. Všechna tato data může útočník odcizit a následně využít, většinou k nelegálním činnostem.

Nejzákladnější prvek ochrany mobilního telefonu je zámek obrazovky. Pokud telefon někde ztratíte, zapomenete, necháte odložený nebo vám ho útočník odcizí, má první překážku, přes kterou se musí dostat. Zařízení bez zámku obrazovky je jako „naservírované na podnose“. I přesto, že tento prvek zabezpečení zvládne nastavit úplně každý a jeho používání je velice jednoduché a uživatelsky přívětivé, téměř třetina uživatelů tuto funkci nepoužívá [8].

Google bere bezpečnost svého operačního systému opravdu vážně. V několika posledních verzích můžeme vidět velký posun v oblasti zabezpečení zařízení, ať už se bavíme o nových funkcích, nebo o změně architektury systému. Základní prvky zabezpečení, které jsou zabudovány v systému, si v následujících podkapitolách přiblížíme.

2.1 Linuxové jádro jako základní prvek bezpečnosti

Jak už bylo zmíněno, Android běží na linuxovém jádře. Linuxové jádro je po mnoho let celosvětově používáno v mnoha systémech, důvěřují mu jak velké společnosti, tak odborníci na bezpečnost. Je prozkoumané do posledního bitu, a tak slouží jako základní bezpečnostní prvek na Androidu. Mimo běžné funkce, které linuxové jádro poskytuje, jsou z hlediska bezpečnosti důležité tyto dvě:

- oddělení dat jednoho uživatele od dat druhého uživatele;
- oddělení jednotlivých procesů.

2.1.1 Oddělení uživatelských dat

Android byl původně navrhován pouze pro chytré telefony a nebylo proto zapotřebí řešit podporu více uživatelů. Až s verzí 4.2 přišla možnost mít na zařízení více uživatelů, v té době ovšem jen na tabletech, které byly více sdíleny oproti mobilním telefonům. Každý uživatel zařízení může mít vlastní domovskou obrazovku, widgety, aplikace, nastavení a také soubory, které jsou ostatním uživatelům nepřístupné. Uživatelské účty jsou odlišeny uživatelským identifikátorem (user ID)[8].

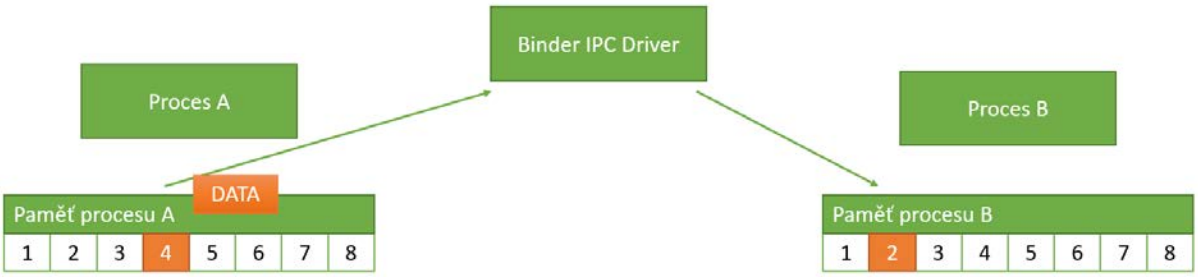
[8] Pozor na záměnu s UID, které je přiřazováno jednotlivým aplikacím.

2.1.2 Oddělení procesů

Oddělení procesů je dobré ze dvou důvodů – stability a bezpečnosti. Pokud více procesů bude přistupovat ke stejné části paměti, může nastat kolize, kdy se v jednom okamžiku snaží oba procesy zapisovat na stejné místo a může dojít k pádu systému. Z bezpečnostního hlediska je oddělení procesů výhodné, protože znemožňuje, aby neoprávněná aplikace přistupovala například k datům e-mailového klienta.

Tento způsob oddělení znemožňuje procesům poskytovat své služby navenek. Proto byl vytvořen mechanismus, který se nazývá Inter-Process Communication (IPC). Pomocí IPC mohou procesy přistupovat ke službám jiných procesů a využívat je. Standardní IPC mechanismus nebyl dostatečně flexibilní pro komunikaci na platformě Android, a tak bylo potřeba vyvinout jiný mechanismus – Binder.

Centrální jednotkou, přes kterou proudí všechna meziprocesová komunikace, je Binder driver. Princip komunikace zahrnuje tři kroky, viz Obrázek 2.1. Nejprve první proces vytvoří zprávu, ve které posílá data druhému procesu. Binder driver poté alokuje část paměti patřící druhému procesu a následně data obsažená ve zprávě zkopíruje do této paměti. Proces na příjímací straně má tak možnost přistoupit k datům prvního procesu, protože už jsou obsažená v jeho části paměti.



Obrázek 2.1: Princip komunikace mezi procesy

2.2 Sandboxing aplikací

Dříve, než se dostaneme k tomu, jak funguje sandboxing aplikací na Androidu, bychom si měli upřesnit, co to vlastně sandboxing je. Odborníci Bryan Ford a Russ Cox z Massachusettského Institutu Technologií definují sandboxing následovně:

„*A sandbox is a mechanism by which a host software system may execute arbitrary guest code in a confined environment, so that the guest code cannot compromise or affect the host other than according to a well-defined policy.*“ [9]

Jinak řečeno, sandbox spustí rizikový program v odděleném prostředí. Veškeré operace, které jsou programem vykonávány, mají dopad pouze na toto oddělené prostředí, ale nikoliv na hostující systém. Představit si to můžeme na virtuálním počítači (hostovaný systém = guest) spuštěném na našem klasickém počítači (hostující systém = host). V případě, že bude hostovaný systém dokonale oddělený od hostujícího systému, veškeré operace provedené

škodlivým kódem v hostovaném systému nebudou mít žádný dopad[9] na hostující systém, tedy na náš klasický počítač.

Aplikační sandboxing na Androidu není doslovný. Funguje tak, že aplikace jsou odděleny jak na úrovni procesů, tak na úrovni dat. Každé aplikaci je při její instalaci přiděleno různé identifikační číslo, tzv. UID[10]. Na základě tohoto UID se při spuštění aplikace definuje, zda bude mít svůj vlastní proces nebo bude přiřazena do již vytvořeného procesu jiné aplikace. Aplikace se stejným UID mohou běžet ve stejném procesu, zatímco pokud mají UID různé, každá běží jako jedinečný proces.

Bezpečnost je dále zajištěna tím, že jednotlivé procesy (aplikace) mezi sebou nemohou komunikovat jinak, než pomocí IPC. Dále nemohou přistupovat k samotnému operačnímu systému, bez dodatečných práv mohou využívat pouze základních služeb systému.

Každá aplikace má své vlastní úložiště. Práva číst a zapisovat má pouze aplikace, které patří ta část paměti a případně každá aplikace, která má stejné UID. Tím je zajištěno, že se k datům jiné aplikace nedostane žádná škodlivá aplikace. Pokud chce aplikace přistoupit k paměti druhé aplikace, nebo obecně k jakýmkoliv jejím zdrojům, musí si vyžádat dodatečná práva, která povoluje uživatel (například právo číst seznam kontaktů).

Na Androidu nejsou odděleny jen aplikace. Kromě nich běží v sandboxu také knihovny operačního systému, Java API framework a vše, co je v systému vrstev umístěno nad linuxovým jádrem. Navíc nejsou sandboxovány jen procesy aplikací, ale všechny jejich zdroje, tedy i paměť. Proto chyba v paměti jedné aplikace nenaruší bezpečnost celého systému. Bezpečnost sandboxingu aplikací je provázána s bezpečností kernelu. Aby byl prolomen sandboxing[11], musela by být prolomena bezpečnost samotného jádra systému.

2.2.1 UID

Pro každou aplikaci je vytvořený vlastní „virtuální“ uživatel, které má definováno UID a aplikace běží pod právy tohoto uživatele. Velmi malý počet systémových služeb běží pod root UID, které má hodnotu UID 0. Systémové služby a aplikace běží pod tzv. well-known UIDs, které začínají číslem 1000. Právě UID 1000 patří systémovému uživateli, který má speciální, přesto omezená práva. UID, která jsou automaticky generována a přiřazována aplikacím, začínají číslem 10000.

Aplikace mohou mít stejné UID (share user ID). V takovém případě spolu aplikace sdílejí svá data a běží ve stejném procesu. Časté použití je pro systémové aplikace, kdy pro

[9] Existují útoky na virtuální prostředí, kdy se může malware dostat ze sandboxu. Dalším příkladem nedokonalosti virtualizace je možnost zapisovat na dynamicky se zvětšující disk tak dlouho, až se systém přetíží a spadne.

[10] Převzato z Linuxových desktopových systémů, kde je různé UID (= user ID) přiřazováno různým uživatelům.

[11] Jak se tedy může stát, že malware má přístup k datům e-mailového klienta? Uživatel aplikaci automaticky potvrdí práva při instalaci, aniž by s nimi byl seznámen.

zajištění stability a modularity systému musí mít různé aplikace stejné UID. Příkladem je Android 4.4, kdy systémové UI a keyguard (implementace zámku obrazovky) sdílejí UID 10012 [10].

Sdílené UID je možné mít i u aplikací třetích stran, přestože to není doporučováno. V tom případě musí být podepsány stejným podpisovým klíčem. Často se sdílené UID používá u modulárních aplikací (například aplikace o počasí, do které můžete doinstalovávat předpovědi z různých radarových systémů).

2.3 Oprávnění

Každá aplikace má možnost požádat o přístup k jiným aplikacím nebo službám, než které má povoleny ve výchozím stavu. Systém poté přidělí aplikaci tzv. oprávnění. Dříve (do verze 6.0) se oprávnění přidělovala při instalaci aplikace, a pokud uživatel aplikaci nedal příslušná oprávnění, aplikace se nenainstalovala. Od verze 6.0 se oprávnění přidělují až ve chvíli, kdy je opravdu aplikace potřebuje. Aplikaci můžete nainstalovat, aniž byste jí přiřadili speciální oprávnění. Až v okamžiku, kdy aplikace požádá o přidělení oprávnění (například aplikace fotoaparátu požádá o přístup k jeho ovládání), uživatel se může rozhodnout, jestli oprávnění přidělí, nebo ne. Oprávnění jsou tedy přidělována až za běhu aplikace. Navíc uživatel může kdykoliv oprávnění aplikaci odebrat v nastavení (na rozdíl od dřívějších verzí, kdy odebrání práv nebylo možné a jediným způsobem bylo odinstalování aplikace).

2.3.1 Úroveň oprávnění

Oprávnění můžeme rozdělit do čtyř úrovní podle toho, jak velké je potenciální riziko, které dopadá na uživatele nebo systém, při udělení oprávnění. Jinak řečeno, oprávnění ACCESS_NETWORK_STATE (umožňuje aplikaci přístup k informacím o síti) se vyznačuje nízkým rizikem a spadá proto do nejnižší kategorie **normal**, zatímco oprávnění READ_SMS (povoluje aplikaci číst SMS zprávy) je rizikovější a aplikace s tímto oprávněním může narušit soukromí uživatele, spadá tedy do vyšší kategorie **dangerous**.

Normal

Nejméně rizikovou kategorií je úroveň normal. Oprávnění jsou vyhodnocena jako málo riziková pro systém, uživatele nebo další aplikace, a tak jsou přiřazována automaticky bez potvrzení uživatele, jakmile o ně aplikace zažádá. Příkladem může být USE_FINGERPRINT (povoluje aplikaci využít čtečku otisku prstů) nebo výše zmíněné ACCESS_NETWORK_STATE.

Dangerous

Oprávnění spadající do kategorie dangerous můžeme popsat jako oprávnění přístupu k datům nebo zdrojům, které by mohly zahrnovat uživatelova osobní data nebo by mohly mít škodlivý efekt na uživatelská data nebo operace ostatních aplikací. Pokud aplikace požádá o některé oprávnění dangerous, systém vyzve uživatele k přidělení oprávnění nebo zamítnutí žádosti. U žádosti je uveden popis oprávnění tak, aby se uživatel mohl rozhodnout, jestli daná aplikace

opravdu požadované oprávnění potřebuje. Příkladem dangerous oprávnění je READ_CALENDAR a WRITE_CALENDAR (umožňuje číst uživatelská data uložená v kalendáři nebo zapisovat) nebo oprávnění CAMERA (povoluje přístup k fotoaparátu a všem jeho funkcím).

Signature
Pokud aplikace požádá o signature oprávnění, musí být podepsána stejným klíčem jako aplikace, která oprávnění deklaruje. Všechny balíčky, které tvoří systém (Settings, System UI, Phone atd.) jsou podepsány stejným klíčem, tzv. platform key, a jsou definovány ve framework-res.apk. Aplikace, která požádá o systémová oprávnění, musí být podepsána stejným klíčem jako tento framework (platform key). Stejně to funguje i u aplikací třetích stran.

Jednoduše se tato problematika dá vysvětlit na příkladu aplikace Počasí. Ta chce přistupovat k poloze zařízení. Oprávnění ACCESS_FINE_LOCATION[12], které umožňuje aplikaci zjistit polohu zařízení, spadá do kategorie dangerous a uživatel musí potvrdit žádost o přidělení tohoto oprávnění. Poté si uživatel nainstaluje aplikaci Počasí2, která získává informace o počasí z jiných satelitů, ale je vydávána stejným autorem a podepsána stejným klíčem. Protože aplikace Počasí již má přidělené oprávnění ACCESS_FINE_LOCATION, žádost o oprávnění pro aplikaci Počasí2 již uživatel nemusí schvalovat a systém jí oprávnění přidělí automaticky.

SignatureOrSystem
Oprávnění spadající do této kategorie se přidělují aplikacím, které jsou buď součástí systému (Android System Image), nebo které mají stejný klíč jako aplikace, která oprávnění deklaruje. Výrobci zařízení, kteří mají v systému předinstalované své aplikace a chtějí využívat specifické funkce systému, již nemusí mít aplikaci podepsanou stejným klíčem jako systémové aplikace (platform key). Jejich aplikace jsou nainstalovány v adresáři /system/priv-app, a mají tak přidělen tento stupeň oprávnění.

2.3.2 Permission groups
Všechna dangerous systémová oprávnění jsou rozdělena do skupin, tzv. permission groups. Oprávnění se rozdělují podle typu požadavku, například všechna oprávnění pojící se se seznamem kontaktů patří do skupiny CONTACTS. Na základě toho, jestli aplikace již má nějaké oprávnění z dané skupiny přidělené, nebo ne, mohou nastat dva scénáře.

Představit se to dá na aplikaci Kontakty, která není předinstalovaná. V prvním případě, kdy aplikace ještě nemá přidělené žádné oprávnění, zažádá o přidělení oprávnění ke čtení seznamu kontaktů READ_CONTACTS. Toto je oprávnění typu dangerous, uživatel proto musí žádost potvrdit. V dialogovém okně je zobrazen popis celé skupiny oprávnění CONTACTS, žádost naopak neobsahuje popis konkrétního oprávnění READ_CONTACTS. Pokud

[12] Podobné oprávnění je ACCESS_COARSE_LOCATION, které má přístup k přibližné poloze definované internetovým připojením. ACCESS_FINE_LOCATION získává údaje o poloze z GPS.

uživatel potvrdí žádost, aplikaci se přidělí pouze to oprávnění, o které zažádala, tedy READ_CONTACTS.

Uživatel poté chce pomocí své aplikace Kontakty vytvořit nový kontakt. Protože aplikace nedisponuje oprávněním WRITE_CONTACTS, musí si o něj požádat. Jelikož ale už jedno oprávnění z této skupiny má přiděleno (READ_CONTACTS), automaticky jí systém oprávnění přidělí, aniž by musel uživatel žádost potvrdit.

2.3.3 Custom permissions

Vývojáři mohou funkce svých aplikací nabídnout k užívání ostatním aplikacím. To se děje pomocí služeb (services), které zpřístupní navenek. Aby aplikace mohla používat služby jiné aplikace, musí k tomu mít oprávnění. Toto oprávnění definuje vývojář ve své aplikaci, vzniká tzv. custom permission. Ostatní aplikace mohou zažádat o přidělení tohoto oprávnění za účelem využití služeb příslušné aplikace. Kromě názvu oprávnění musí vývojář uvést také popis, permission group a úroveň oprávnění.

2.4 Podepisování aplikací

Všechny aplikace na platformě Android musí být digitálně podepsány autorem. Nepodepsaná aplikace by se neměla dostat do Google Play – primárního zdroje pro instalaci aplikací. Pokud se tak přeci jen stane, je samotným Googlem smazána. V případě, že uživatel instaluje nepodepsanou aplikaci z jiného zdroje než z Google Play, operační systém zablokuje instalaci a nepovolí pokračování. Tímto je zaručeno, že každá aplikace bude mít identifikovatelného autora. Navíc podepsání aplikace umožňuje vývojářům snazší aktualizování aplikace. Podepisování se děje pomocí podpisového páru (veřejný a soukromý klíč). Soukromým klíčem je aplikace podepsána (ten musí být bezpečně uložený na straně autora aplikace), veřejný klíč je uložen do certifikátu a ten je přibalen k balíčku APK. Na základě podpisu je aplikaci přiřazeno UID.

2.4.1 Správa klíčů

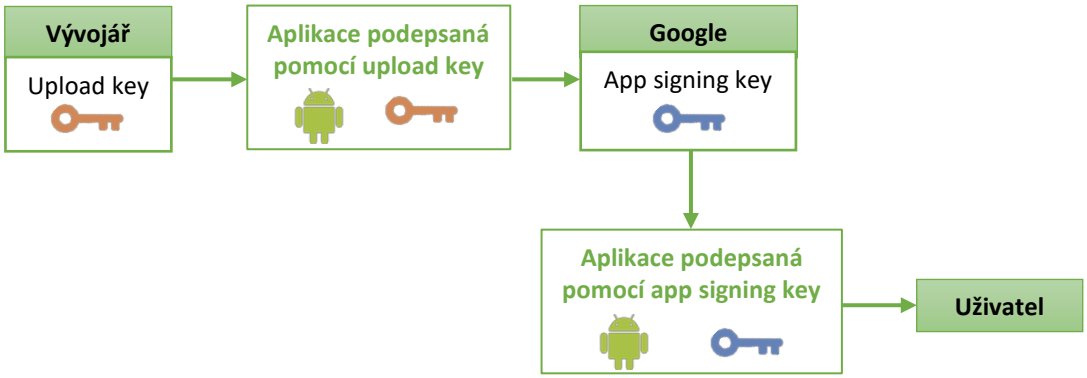
Jako autor aplikace musíte uchovávat svůj podpisový klíč v bezpečí. Pomocí něj lze podepsat novou verzi aplikace a distribuovat ji uživatelům. Ke správě klíčů lze využít dva způsoby uchovávání. Jeden je založen na uložení klíče v databázi společnosti Google, ten druhý na uložení klíče ve vlastnictví vývojáře.

Podepisování klíčem uloženým u Googlu

Tento způsob zahrnuje použití dvou klíčů, app signing key a upload key. Pomocí nástroje Play Encrypt Private Key lze zašifrovat a vyexportovat app signing key a následně ho nahrát do databáze Googlu. Následně si vývojář vytvoří upload key a zaregistruje si ho. Aplikaci poté podepíše svým upload key a nahraje ji do Google Play. Google na základě certifikátu ověří identitu vývojáře aplikace a přepíše podpis aplikace jeho app signing key. Aplikace je následně publikována v Google Play. Celý proces ukazuje Obrázek 2.2.

Pokud vývojář ztratí svůj upload key, jednoduše požádá Google o zablokování starého klíče a vygenerování nového. Protože app signing key je bezpečně uložen v databázi Googlu,

vývojář může bezpečně produkovat další aplikace, nové verze nebo aktualizace i se změněným upload key.

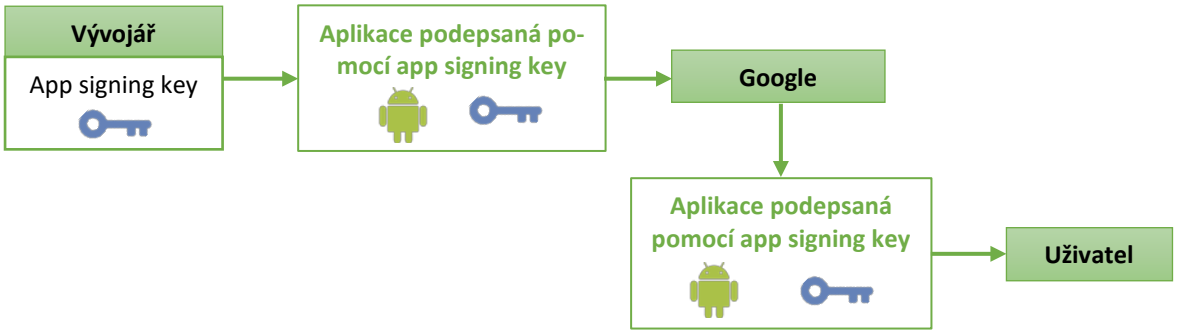


Obrázek 2.2: Podepisování aplikací upload key a app signing key [11]

Podepisování klíčem uloženým v klíčence

Namísto uložení podpisového klíče v databázi Googlu může vývojář uložit svůj klíč do své klíčenky (keystore). Přístup do klíčenky je chráněn heslem, její obsah je šifrován a může být použita pro uložení více klíčů. Navíc každý klíč může být chráněn dalším heslem. Při publikování své aplikace vývojář podepíše balíček svým app signing key a nahraje jej přímo do Google Play, viz Obrázek 2.3.

Když vývojář ztratí přístup ke své klíčence, společnost Google nemá možnost žádným způsobem obnovit jeho app signing key a vývojář tím pádem již nemůže vydávat nové verze ani aktualizace svých aplikací.



Obrázek 2.3: Podepisování aplikací pomocí app signing key [11]

2.4.2 Podpisová schémata

Jak bylo zmíněno výše, každá aplikace na platformě Android musí být podepsaná, respektive musí mít podepsaný svůj balíček. Android poskytuje dvě podpisová schémata, první je založené na podpisu .jar souboru. U druhého podpisového schématu se podepisuje APK [12].

Pro zachování maximální kompatibility by měli vývojáři používat oba dva typy podepisování. Aplikace určené pro verzi systému 6.0 a starší musí být podepsané původním formátem (verze 1), pokud je navíc připojen i druhý podpis (verze 2), aplikace novější formát ignoruje. Pro Android verze 7.0 a novější je primárně určený druhý typ podepisování, v případě, že je aplikace podepsána jen prvním typem, použije se k ověření ten.

Podepisování JAR souboru (verze 1)

V tomto schématu dochází k podepsání pouze .jar souboru. Navíc nepodporuje podepsání některých částí APK. Při ověřování podpisu musí ověřovací proces zpracovat značné množství nedůvěryhodných dat. To zvyšuje riziko útoku. Navíc se musí dekomprimovat všechna komprimovaná data, což vyžaduje značné množství času a paměti.

Podepisování APK (verze 2)

Od Androidu ve verzi 7.0 je k dispozici druhá verze podpisového schématu založeného na podepsání celého APK. Obsah balíčku je hashován, podepsán a následně vložen do původního APK. Ověřování probíhá na celém APK balíčku, tedy i na metadatech zip souborů a dalších datech, které v původním schématu nejsou podepisovány, což snižuje možnosti útoku.

2.5 SELinux

Současné operační systémy poskytují tzv. discretionary access control (DAC, volitelné řízení přístupu). Každý soubor má přidružený access control list (ACL), ve kterém jsou definovány práva vlastníka souboru, skupiny, do které uživatel patří, a ostatních uživatelů, kteří nejsou ani vlastníkem a nespadají ani do stejné skupiny uživatelů jako vlastník. Každý spuštěný proces má stejná práva jako uživatel, který ho spustil, takže může přistupovat ke všem datům daného uživatele. Zde vzniká riziko spuštění škodlivého souboru pod administrátorským účtem, spuštěný malware by tak měl stejná práva jako administrátor. Tomuto se snaží zabránit právě SELinux.

Opakem DAC je mandatory access control (MAC, povinné řízení přístupu). Každý proces je začleněn do tzv. domény, jejíž vrstvu nemůže opustit a může komunikovat pouze s procesy umístěnými ve stejné doméně. Ani v případě, že je proces spuštěný s administrátorskými právy, nemá možnost opustit doménu, do které je přiřazen.

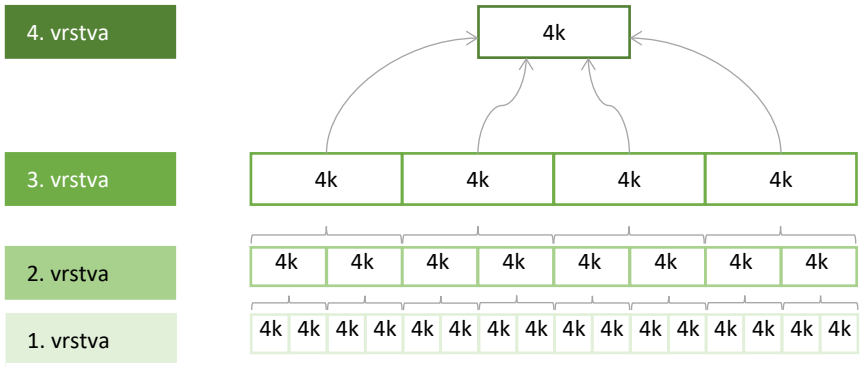
Pro každý uživatelský účet je definován SELinux účet, který má přístup k určitým rolím definujícím, k jakým zdrojům má uživatel přístup. Role je mapována na doménu a doména spojena se subjektem (uživatel, aplikace, proces).

2.6 Verified boot

Android poskytuje ochranu integrity zařízení díky funkci zvané verified boot. Tato funkce kontroluje integritu softwaru zařízení při bootování, tzn., že nebyly provedeny žádné změny kritických částí systému. Verified boot zajišťuje komponenta device-mapper verity (dm-verity). Pro ověření integrity se využívá hashovací funkce SHA256 [13].

2.6.1 Proces ověření integrity

Úložiště operačního systému Android je rozděleno do bloků. Komponenta dm-verity je založena na předpočítané stromové struktuře hashů, kdy na nejnižší úrovni stromu se všechny bloky zahashují funkcí SHA256 a výstup funkce z několika bloků je vstup do hashovací funkce vyšší úrovně (ukázka viz Obrázek 2.4). Na principu, kdy hash rodiče je vypočítán z hashů potomků, je postaven celý strom. Kořenem stromu je tzv. root hash, který se jako jediný používá k ověření integrity. Pokud bude změněn jakýkoliv jediný blok úložiště, změní se jeho hash a tato změna se propíše až do nejvyšší úrovně, tedy do root hashe. Předpočítaný strom hashů je uložen v bootovacím oddílu. Při spuštění zařízení dm-verity provede hashování jednotlivých bloků a použije stejný princip stromové struktury. Jakmile získá root hash, porovná ho s předpočítaným root hashem uloženým v zařízení. Ověřování probíhá pomocí RSA klíčů. Výrobce zařízení si uloží svůj OEM[13] certifikát s veřejným klíčem do úložiště. Výsledkem procesu je rozhodnutí, zda jsou oba hashe shodné [10].



Obrázek 2.4: Stromová struktura ověřování integrity [13]

2.6.2 Stav bootování

Jakmile je proces ověření dokončen, bootovací proces naběhne do jednoho ze čtyř stavů podle toho, jak proces ověření dopadl.

- RED = ověření selhalo, bootloader zobrazí varování a zastaví bootování.
- GREEN = ověřování je úspěšně dokončeno, integrita zařízení je potvrzena OEM certifikátem.
- ORANGE = zařízení je modifikováno, pravděpodobně byl proveden rooting zařízení, je zobrazeno varování a ponecháno na uživateli, jestli chce pokračovat v bootování.
- YELLOW = integrita byla ověřena zabudovaným (ale ne OEM) certifikátem a podpis je validní, bootloader zobrazí varování a otisk veřejného klíče předtím, než umožní bootování.

[13] OEM = Original Equipment Manufacturer, výrobce zařízení, jehož výrobek je prodáván pod jinou obchodní značkou

2.6.3 Stav zařízení

Každé zařízení může být v jednom ze dvou stavů, zamčeném (locked) a odemčeném (unlocked). Každé nové zařízení je ve výchozím stavu LOCKED. Pokud uživatel chce využít rooting zařízení, musí si zařízení přepnout do stavu UNLOCKED.

- LOCKED = rooting zařízení je zakázaný, může nabootovat jen do GREEN, RED a YELLOW stavu.
- UNLOCKED = ověření se neprovádí a zařízení může být rootnuto, nabootuje vždy do stavu ORANGE.

2.6.4 Implementační třídy

Na základě toho, jestli zařízení podporuje UNLOCKED režim a do jakých stavů může nabootovat, rozlišujeme dvě implementační třídy pro verified boot.

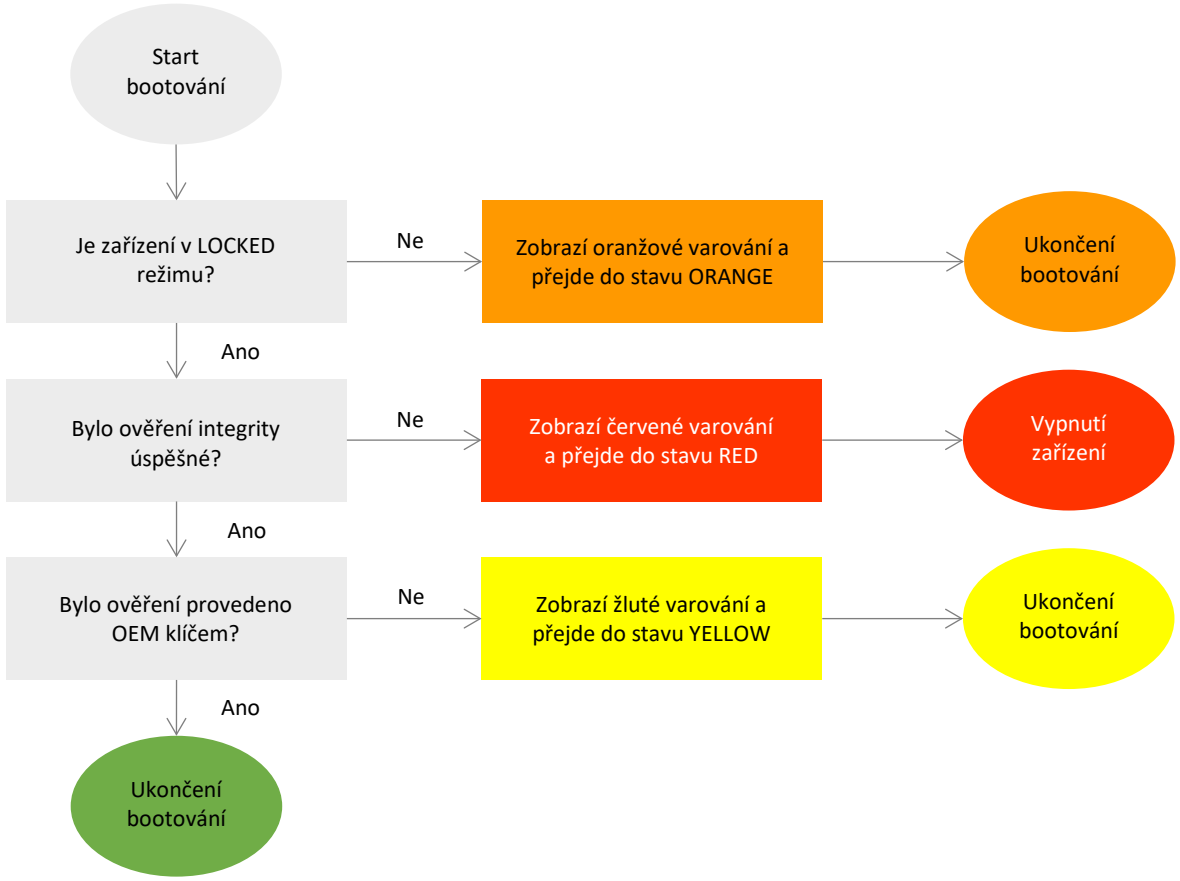
- Třída A = podporuje pouze režim LOCKED a zařízení může nabootovat do stavu GREEN nebo RED.
- Třída B = přidává k třídě A navíc režim UNLOCKED a bootovací stav YELLOW a ORANGE.

2.6.5 Bootovací proces

Při spuštění zařízení je zahájeno bootování (viz Obrázek 2.5). Nejdříve se zjistí, v jakém stavu se zařízení nachází. Pokud je v režimu UNLOCKED, bootovací proces přejde do stavu ORANGE a zobrazí se varování, poté je bootování dokončeno a bootloader načte operační systém (neproběhne tedy proces ověření integrity). V opačném případě, tedy jestliže je zařízení v režimu LOCKED, přejde se k dalšímu kroku, a tím je proces ověření integrity (viz 2.6.1).

Komponenta dm-verity zajišťující ověřovací proces po jeho skončení dospěje k jednomu ze dvou stavů – ověření proběhlo úspěšně a může se přejít k dalšímu kroku, nebo úspěšné nebylo a bootovací proces přejde do stavu RED, zobrazí varovnou hlášku a následně vypne zařízení.

Posledním krokem je zjištění, jakým klíčem bylo ověření provedeno. Pokud OEM klíčem, vše je v pořádku, bootovací proces přejde do stavu GREEN a načte operační systém. Ve druhém případě, kdy k ověření byl použit jiný klíč, bootloader přepne bootavací stav na YELLOW, zobrazí varování a následně ukončí proces bootování a načte operační systém [14].



Obrázek 2.5: Proces bootování [13]

2.7 Šifrování

Pokud v dnešní době dojde k odcizení mobilního zařízení, není důležitá ani tak ztráta samotného zařízení, jako ztráta dat. Ta se dají obnovit ze zálohy, ale je velké riziko, že i přes určitou formu ochrany (např. zámek obrazovky) se k datům dostane někdo nepovolaný. V tomto případě nemluvíme pouze o ztrátě dat, ale o jejich odcizení a kompromitaci útočníkem. Android poskytuje jednoduchou ochranu proti tomuto riziku – šifrování. Nabízí dva způsoby šifrování dat v zařízení, klasické zašifrování celého disku, tzv. Full-disk encryption (FDE), nebo šifrování souborů, kdy různé soubory mohou být šifrovány různými klíči, tzv. File-based encryption (FBE) [15].

2.7.1 Full-disk encryption

Šifrování celého disku bylo představeno jako nová funkce ve verzi 3.0, postupně byly přidávány další funkce až do verze 5.0, kdy byla předělána architektura šifrování. Full-disk encryption nabízí možnost zašifrovat všechna uživatelská data v zařízení. K procesu šifrování se využívá komponenta dm-crypt, která má na starosti také šifrování na linuxových operačních systémech. Šifrovací algoritmus je 128 bitový AES s módem CBC.

Zařízení je šifrováno symetrickým klíčem, tzv. disk encryption key (DEK, také někdy master key). Z uživatelova hesla (PIN, heslo, znak[14]) nebo defaultního hesla[15] (v případě, že uživatel nemá nastavené nic z předešlého) se odvodí další 128bit AES klíč, tzv. key encryption key (KEK). Pomocí KEK je zašifrován master key a uložen do hardwarově zabezpečeného úložiště TEE[16]. Při každém bootování zařízení je vyžadováno, aby uživatel zadal heslo (případně PIN nebo znak) a umožnil tak dešifrování zařízení.

2.7.2 File-based encryption

Druhou možností jak šifrovat uživatelská data je metoda File-based encryption (od verze 7.0). Ta umožňuje šifrovat pouze soubory a to každý jiným klíčem. Jejich dešifrování tak probíhá nezávisle na sobě. U metody FDE musel uživatel zadat svoje heslo, poté teprve zařízení nabootovalo. Do té doby nefungovaly žádné služby operačního systému. Aby se tomuto problému metoda FBE vyhnula, byla představena nová funkce, tzv. direct boot.

Direct boot umožňuje zařízení načíst operační systém až do stavu uzamčené obrazovky. Aby bylo možné v takovém stavu, tedy dříve než uživatel zadá své heslo, přijímat hovory, spouštět budíky a provozovat další důležité služby, bylo potřeba oddělit uživatelská data od potřebných systémových dat. Proto byly vytvořeny dva typy úložiště:

- Credential Encrypted (CE) storage = uložiště, které je výchozí pro ukládání souborů a je přístupné až po tom, co uživatel zadá heslo;
- Device Encrypted (DE) storage = uložiště přístupné už během direct bootu (dříve než uživatel zadá heslo).

DE úložiště obsahuje všechny důležité soubory pro základní chod systému a pro správnou funkčnost výše zmíněných služeb. Poté co uživatel zadá údaje pro dešifrování souborů uložených v CE, je i toto uložiště odemčeno a uživatel má kompletní přístup k zařízení.

2.8 Autentizace uživatele

Nejčastějším způsobem ověření identity uživatele je autentizace pomocí tajné informace, například heslem. Moderní mobilní zařízení kromě toho využívají i autentizaci biometrikou (otisk prstu, sken oční duhovky apod.) V operačním systému tuto funkčnost zajišťují dvě komponenty, autentizaci pomocí tajné informace zaštiťuje gatekeeper, zatímco biometrickou autentizaci fingerprint [16].

2.8.1 Autentizace pomocí tajné informace

Za tajnou informaci můžeme na Androidu považovat tři věci – heslo, PIN a znak či tzv. pattern. Heslo může být složeno z alfanumerických a speciálních znaků, PIN pouze z čísel.

[14] Neboli pattern je jeden ze způsobů, jak zabezpečit zařízení zámkem obrazovky.

[15] „default_password“

[16] Trusted execution environment, více viz [17].

Znak je možnost jak odemknout zařízení spojením určitých bodů na obrazovce ve správném pořadí.[17]

Každý uživatel má přidělené identifikační číslo, tzv. user SID. Při vytváření hesla[18] se uživatelovo SID propojí s jeho heslem. Pokud v budoucnu bude chtít uživatel změnit heslo, musí zadat své staré, aby došlo k správnému propojení user SID s novým heslem a staré bylo zapomenuto.

Proces autentizace probíhá v několika krocích. Nejdříve uživatel zadá své heslo. Gatekeeperdeamon poté pošle heslo protistraně v TEE, kde proběhne ověření a vygeneruje se autentizační token (AuthToken) obsahující uživatelovo SID. Na základě autentizačního tokenu se poté uděluje přístup uživateli k datům.

2.8.2 Autentizace pomocí biometriky

Většina dnešních telefonů má snímač otisku prstů. Pomocí něj se může uživatel autentizovat biometrickou metodou. Druhou možností je využít skener oční duhovky, tato metoda ale ještě není rozšířená zejména z důvodu malé podpory mezi výrobci mobilních telefonů.

Proces ověření otiskem prstu zajišťuje komponenta fingerprint. Pokud zařízení má snímač otisku prstů, fingerprint HAL[19] umožňuje propojit snímač se specifickými knihovnami výrobce. Jakmile uživatel přiloží prst na snímač, je zavolána metoda pro ověření otisku prstu (v případě vytváření vzorů otisků metoda pro zapsání otisku) a knihovna implementovaná výrobcem posoudí, zda se otisk shoduje se vzorem uloženým v zařízení. Výsledek ověřovacího procesu je poslán zpět do fingerprint HAL, odkud informace putuje do fingerprintdeamon, který potvrdí nebo zamítne autentizaci.

Vzory otisků prstů jsou zašifrovány a uloženy v TEE. Navíc jsou podepsány, což znemožňuje zkopírování vzorů do jiného telefonu za účelem kompromitace původního zařízení. K ochraně přístupu k otiskům jsou použity politiky SELinux.

[17] Práce norského studenta Marte Løge ukazuje, jak se dají znaky odhadnout podle fyziologických rysů a podle osobnosti člověka [18].

[18] V následujícím textu se omezíme pouze na používání hesla pro zjednodušení problematiky

[19] Hardware abstraction layer, viz kapitola 1.4.2

3 Hrozby pro uživatele s OS Android

Android je nejpoužívanější operační systém na světě, proto se také setkáváme stále více s útoky na tento systém. Mobilní telefon mají uživatelé většinu času u sebe a zapnutý, nové hrozby se tak mohou šířit mnohem rychleji a k uživateli se mohou dostat dříve, než se o nich stačí dozvědět ze zpravodajství nebo na internetu. Příkladem může být phishing. Více než polovinu e-mailů otevřou uživatelé na mobilním telefonu [19], průměrná doba odpovědi na e-mail z mobilního telefonu se pohybuje pod hranicí třiceti minut [20]. Z toho vyplývá, že podvrhnutý e-mail se k uživateli dostane dříve a rychlost jeho šíření je vyšší než v minulých letech, kdy se k e-mailu lidé dostali večer po příchodu z práce.

Uživatelé mohou přistupovat pomocí mobilního telefonu ke svému bankovnímu účtu, k osobnímu e-mailu nebo třeba k citlivým datům (smlouvy uložené v cloudu, soukromé fotografie apod.) Navíc pokud uživatel má i firemní telefon nebo firma používá politiku BYOD, uživatel může přistupovat také k firemnímu e-mailu nebo k firemním datům.

Pokud se útočníkovi podaří kompromitovat mobilní telefon, může získat také přístup k některým z výše uvedených dat. Jak pro samotné uživatele, tak pro firmy představuje kompromitace mobilního telefonu značný problém. V následujícím textu budou popsány nejčastější hrozby a útoky na operační systém Android a každá hrozba i útok bude celkově ohodnocena. Přesný popis metodiky hodnocení je uveden v kapitole 3.1.

3.1 Metodika modelu hrozeb

Každá hrozba je zařazena do jedné z pěti kategorií. Metodika kategorizace vychází z rozdělení podle společnosti Lookout [21], kdy se jednotlivé kategorie liší tím, z jakého „světa“ hrozba pochází – aplikace, internet, síť a fyzický svět. Podobné rozdělení bylo představeno i společností Skycure [22]. Dále je přidána kategorie „Útoky využívající znalosti telefonního čísla“ kvůli přidání smishingu (viz kapitola 3.11). Celkově jsou tedy hrozby rozděleny do následujících kategorií:

- aplikační hrozby;
- webové hrozby;
- síťové hrozby;
- fyzické hrozby;
- útoky využívající znalosti telefonního čísla.

Abychom mohli nějakým způsobem seřadit hrozby, poslouží nám Common Vulnerability Scoring System[20]. Ten není primárně určený na hodnocení hrozeb, ale v tomto případě může fungovat stejně dobře. K ohodnocení každé zranitelnosti, v našem případě hrozby, vyžaduje nastavení osmi parametrů, které se zranitelnosti, hrozby nebo útoku týkají. Jejich popis a příslušné hodnoty viz Tabulka 3.1**.** Na stránkách projektu [23] je k dispozici kalkulátor.

[20] Zkráceně CVSS [23], je způsob, jakým zachytit principy zranitelností a numericky ohodnotit každou zranitelnost. Tato hodnota reprezentuje závažnost zranitelností.

Tabulka 3.1: Metodika ohodnocení hrozeb a útoků na OS Android

Název metriky	Popis metriky	Možné hodnoty	Popis hodnot
Vektor útoku	Odráží typ hrozby	Vnější síťový	Útok z venkovní sítě (WAN, GSM)
		Vnitřní síťový	Útok z lokální sítě (LAN, Bluetooth, Wi-Fi)
		Lokální	Lokální útok ze zařízení (pomocí číst, zapisovat, spouštět)
		Fyzický	Útočník potřebuje fyzický přístup k zařízení
Komplexita útoku	Jaké podmínky musí být splněny, aby byla hrozba relevantní (nastavení zařízení, získané informace o zařízení apod.)	Nízká	Úspěšnost útoku závisí pouze na útočníkovi a není potřeba splnění jiných podmínek, nebo jen minimálních
		Vysoká	Úspěšnost útoku nezávisí pouze na útočníkovi, ale na splnění podmínek, které nejsou útočníkem ovlivnitelné
Vyžadovaná oprávnění	Jaké oprávnění musí útočník mít, aby byla hrozba relevantní	Žádná	Útok je nezávislý na oprávnění útočníka k zařízení
		Nízká	Útok nevyžaduje žádné vyšší oprávnění k zařízení
		Vysoká	Útok vyžaduje vyšší oprávnění k zařízení
Interakce uživatele	Zda je vyžadována interakce uživatele (např. spuštění aplikace)	Žádná	Útok nevyžaduje interakci uživatele
		Vyžadována	Útok vyžaduje interakci uživatele
Rozsah	Odráží možnost, jestli se útok může rozšířit z jednoho zařízení na druhé	Nezměněný	Hrozba je reálná pouze pro napadené zařízení
		Změněný	Útočník může využít různých mechanizmů k tomu, aby napadl další zařízení
Dopad na důvěrnost	Hodnotí dopad na důvěrnost uložených nebo přenášených dat	Žádný	Žádný dopad na důvěrnost dat
		Nízký	Nízký dopad na důvěrnost dat
		Vysoký	Vysoký dopad na důvěrnost dat
Dopad na integritu	Hodnotí dopad na integritu uložených nebo přenášených dat	Žádný	Žádný dopad na integritu dat
		Nízký	Nízký dopad na integritu dat
		Vysoký	Vysoký dopad na integritu dat
Dopad na dostupnost	Hodnotí dopad na dostupnost uložených nebo přenášených dat	Žádný	Žádný dopad na dostupnost dat
		Nízký	Nízký dopad na dostupnost dat
		Vysoký	Vysoký dopad na dostupnost dat

3.1.1 Seznam hrozeb

- Aplikační hrozby;
  - Malware;
  - Aplikace ohrožující soukromí uživatele;
  - Rooting zařízení;
- Webové hrozby;
  - Drive-by download;
  - Phishing;
- Síťové hrozby;
  - Sniffing Wi-Fi komunikace;
  - Nezabezpečená Wi-Fi;
- Fyzické hrozby;
  - Odcizení nebo ztráta zařízení;
  - Ztráta nebo porušení dat;
- Hrozby využívající znalosti telefonního čísla;
  - Smishing.

3.2 Malware

Mobilní malware je jedním z nejjednodušších způsobů, jak kompromitovat mobilní zařízení. Je mnoho způsobů, jak se může malware do zařízení dostat. Jedním z nich je pomocí phishingového útoku, kdy je uživateli předán odkaz na stránku představující důvěryhodnou službu, ze které si uživatel stáhne škodlivou aplikaci vydávající se za legitimní. Částečnou ochranu proti této hrozbě představuje instalace aplikací jen z oficiálních zdrojů (např. Google Play). Bohužel, ani v těchto oficiálních obchodech s aplikacemi si nemůže být uživatel jistý, že aplikace není škodlivá nebo neobsahuje malware [24].

Co může způsobit?

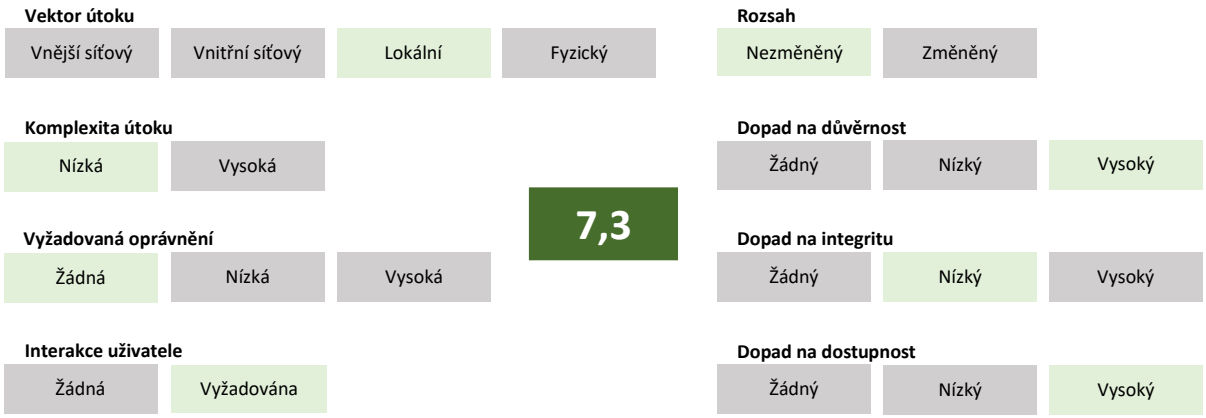
- Instalace dalších škodlivých aplikací;
- Analýza navštěvovaných webových stránek nebo jejich měnění;
- Čtení SMS zpráv a změna odesílaných SMS zpráv;
- Odposlouchávání při přihlašování do aplikací (například aplikace Instagram, VKontakte);
- Zašifrovaní uživatelských dat v telefonu;
- Odposlouchávání přihlašovacích údajů do mobile bankingu a získávání informací o kreditních kartách[21].

[21] Příkladem je malware Acecard. Přidává překryvnou vrstvu při spuštění bankovní aplikace a poté posílá uživatelské přihlašovací údaje útočníkovi. Také má přístup k SMS zprávám, ze kterých může zjistit kód ke dvou-faktorové autentizaci.

Ohodnocení

Tabulka 3.2: Ohodnocení hrozby malware

Metrika	Hodnota	Popis
Vektor útoku	Lokální	Instalace malwaru je lokální, nejedná se o síťový útok
Komplexita útoku	Nízká	Útok je poměrně jednoduchý a častý, stačí, aby si uživatel nainstaloval škodlivou aplikaci (může být i z Google Play)
Vyžadovaná oprávnění	Žádná	K instalaci nejsou vyžadována žádná speciální oprávnění
Interakce uživatele	Vyžadována	Uživatel musí sám nainstalovat aplikaci
Rozsah	Nezměněný	Hrozba se týká ve většině případů pouze zařízení, na které je aplikace nainstalována
Dopad na důvěrnost	Vysoký	Může získávat data ze zařízení, zachytávat přihlašovací údaje apod.
Dopad na integritu	Nízký	Malware obvykle nemění uživatelská data, ale může tímto způsobem pracovat
Dopad na dostupnost	Vysoký	Typickým příkladem je ransomware, který zašifruje data, a uživatel k nim tak ztratí přístup



Obrázek 3.1: Ohodnocení hrozby malware

3.3 Aplikace ohrožující soukromí uživatele

Některé aplikace vyžadují od uživatele schválit oprávnění. Jak bylo vysvětleno v kapitole 2.3, povolit oprávnění je nutné pouze pro typ dangerous. Existují škodlivé aplikace, které zneužívají svých oprávnění ke špehování uživatele. Nejčastěji se jedná o free aplikace, které vyžadují více oprávnění, než potřebují ke svému chodu [25].

Mezi oprávnění, která mohou škodlivé aplikace využít ke sledování uživatele, patří téměř všechna typu dangerous. Mohou sledovat polohu mobilního telefonu, pořizovat fotografie, posílat SMS zprávy a podobně.

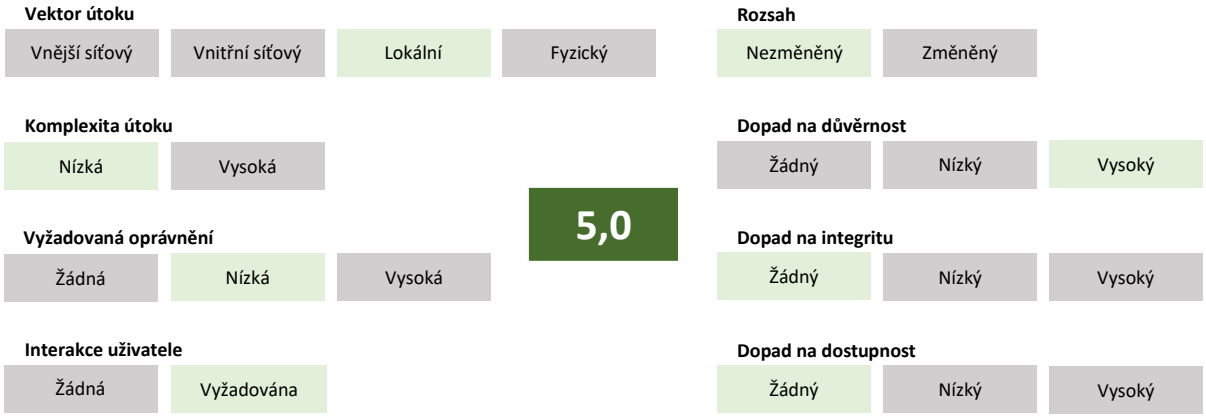
Co může způsobit?

- Přístup k poloze zařízení;
- Pořizování fotografií a videonahrávek;
- Posílání prémiových SMS;
- Přístup k výpisu hovorů;
- Čtení a zapisování na SD kartu;
- Přístup k tělesným senzorům (například měření tepu).

Ohodnocení

Tabulka 3.3: Ohodnocení hrozby aplikace ohrožující soukromí uživatele

Metrika	Hodnota	Popis
Vektor útoku	Lokální	K provedení útoku je potřeba nainstalovat aplikaci na zařízení oběti
Komplexita útoku	Nízká	Útok je poměrně jednoduchý a častý, stačí, aby si uživatel nainstaloval škodlivou aplikaci (může být i z Google Play)
Vyžadovaná oprávnění	Nízká	Uživatel musí povolit aplikaci oprávnění, bohužel někteří uživatelé nekontrolují, jaká oprávnění aplikace požaduje, a automaticky schvalují
Interakce uživatele	Vyžadována	Uživatel musí sám nainstalovat aplikaci
Rozsah	Nezměněný	Hrozba se týká ve většině případů pouze zařízení, na které je aplikace nainstalována
Dopad na důvěrnost	Vysoký	Aplikace může narušit uživatelovo soukromí přístupem k fotoaparátu, posíláním a čtením SMS apod.
Dopad na integritu	Žádný	Útok se obvykle neprojevuje tímto způsobem
Dopad na dostupnost	Žádný	Útok se obvykle neprojevuje tímto způsobem



Obrázek 3.2: Ohodnocení hrozby aplikace ohrožující soukromí uživatele

3.4 Rooting zařízení

Tato hrozba se dá jednoduše popsat jako získání administrátorských práv. Ačkoliv se může zdát, že majitel chytrého telefonu s OS Android má nad telefonem plnou kontrolu, není to tak. Aby správně fungovaly vestavěné bezpečnostní mechanizmy, musí být některé funkce přístupné pouze pro administrátorský účet. Navíc díky omezení uživatelských práv se situace, kdy si uživatel smaže důležitá data nebo provede nevratné změny, stává méně častou a je složitější tyto kroky provést. Root však není ve výchozím nastavení uživateli dostupný. Rooting zařízení (právě od slova „root“) zpřístupní všechny funkce systému a dá uživateli plná práva. Nejčastěji se provádí pomocí různých aplikací dostupných z internetu. Bohužel jsou obvykle infikovány malwarem a jejich instalace je nežádoucí [26].

Pokud se provede rooting správně, aniž by byla nainstalována malwarem infikovaná aplikace, sám o sobě je bezpečný. Bohužel to uživateli dává administrátorská práva a to je obvykle problém. Stává se, že uživatel vypne různé bezpečnostní mechanizmy. Navíc i nainstalovaná škodlivá aplikace má možnost dostat se k oprávnění, která by za normálních okolností nemohla získat.

Co může způsobit?

- Telefon může přestat fungovat[22];
- Aktualizace mohou vyžadovat manuální instalaci (automatická nebude fungovat);
- Bezpečnostní riziko – instalace malwaru (viz kapitola 3.2);
- Ztráta záruky;
- Některé aplikace, například mobilní bankovnictví, kontrolují rootnutí telefonu a v případě, že je kontrola neúspěšná a telefon je rootnutý, automaticky se ukončí.

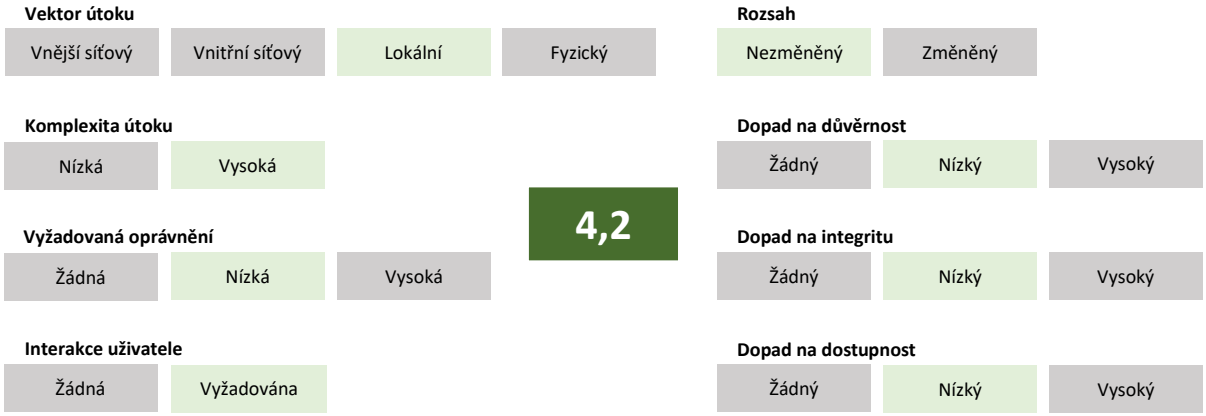
Ohodnocení

Tabulka 3.4: Ohodnocení hrozby rooting zařízení

Metrika	Hodnota	Popis
Vektor útoku	Lokální	Rooting zařízení se provádí nejčastěji pomocí aplikace nebo připojením telefonu k počítači a spuštěním souboru z počítače
Komplexita útoku	Vysoká	Rooting různých zařízení je různě obtížný a provádí se různými technikami, proto je z pohledu útočníka obtížnější zvolit správný postup
Vyžadovaná oprávnění	Nízká	K rootingu jsou potřeba oprávnění, ale ty si aplikace může zajistit sama
Interakce uživatele	Vyžadována	Uživatel musí buď nainstalovat/spustit aplikaci nebo provést rooting pomocí počítače
Rozsah	Nezměněný	Hrozba se týká ve většině případů pouze zařízení, na které je aplikace nainstalována

[22] Při špatně provedeném procesu rootování může dojít k poškození určitých částí paměti. V důsledku tohoto poškození se může stát, že telefon již nepůjde spustit.

Dopad na důvěrnost	Nízký	Může způsobit snížení zabezpečení a tím ohrozit důvěrnost uložených nebo přenášených dat
Dopad na integritu	Nízký	Může způsobit snížení zabezpečení a tím ohrozit integritu uložených nebo přenášených dat
Dopad na dostupnost	Nízký	Může způsobit snížení zabezpečení a tím ohrozit dostupnost uložených nebo přenášených dat



Obrázek 3.3: Ohodnocení hrozby rooting zařízení

3.5 Drive-by download

Tento typ útoku spočívá v začlenění speciálního kódu, který umožňuje provést automatické stažení soboru, do internetové stránky nebo e-mailu. Poté, co oběť klikne na odkaz, se automaticky stáhne malware a spustí se jeho instalace. Příkladem tohoto útoku je malware Svpeng, který byl závislý na spuštění škodlivé stránky v prohlížeči Chrome pro Android. Ve stránce byl vložený javascriptový kód, který vykonával automatické spuštění stahování. To bylo umožněno využitím tehdejší zranitelnosti Chrome, díky které uživatel nemusel ani kliknout na odkaz a stahování se mohlo spustit bez jeho interakce. Další scénář útoku se již mění podle použitého malwaru.

U některých útoků se můžeme setkat s tím, že po stažení se aplikace automaticky spustí (není nutná její instalace) [27] a začne škodit, není tedy nutná interakce uživatele. V opačných případech, kdy je potřebné, aby uživatel aplikaci spustil, se často využívá sociálního inženýrství k přesvědčení uživatele o legitimnosti aplikace.

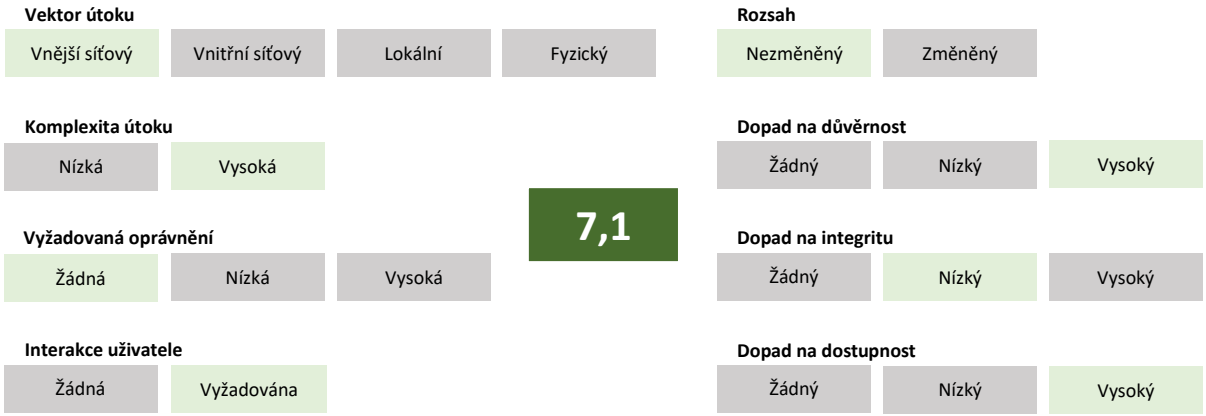
Co může způsobit?

- Instalace malwaru (viz kapitola 3.2);
- Spuštění malwaru bez instalace.

Ohodnocení

Tabulka 3.5: Ohodnocení útoku drive-by download

Metrika	Hodnota	Popis
Vektor útoku	Vnější síťový	Útok se provádí z veřejně dostupných stránek nebo ze škodlivého e-mailu
Komplexita útoku	Vysoká	Útočník musí zajistit spuštění automatického stahování (zranitelnost nebo kliknutí uživatele), poté spuštění nebo instalaci malwaru
Vyžadovaná oprávnění	Žádná	Nejsou vyžadována žádná oprávnění
Interakce uživatele	Vyžadována	Od uživatele mohou být požadovány dvě akce, kliknutí na odkaz a stažení souboru nebo spuštění stažené aplikace, ani jedna z akci však není stoprocentně vyžadována pro úspěšné provedení tohoto útoku
Rozsah	Nezměněný	Hrozba se týká ve většině případů pouze zařízení, na které je aplikace nainstalována
Dopad na důvěrnost	Vysoký	Způsobuje instalaci malwaru a narušuje bezpečnost zařízení
Dopad na integritu	Nízký	
Dopad na dostupnost	Vysoký	



Obrázek 3.4: Ohodnocení útoku drive-by download

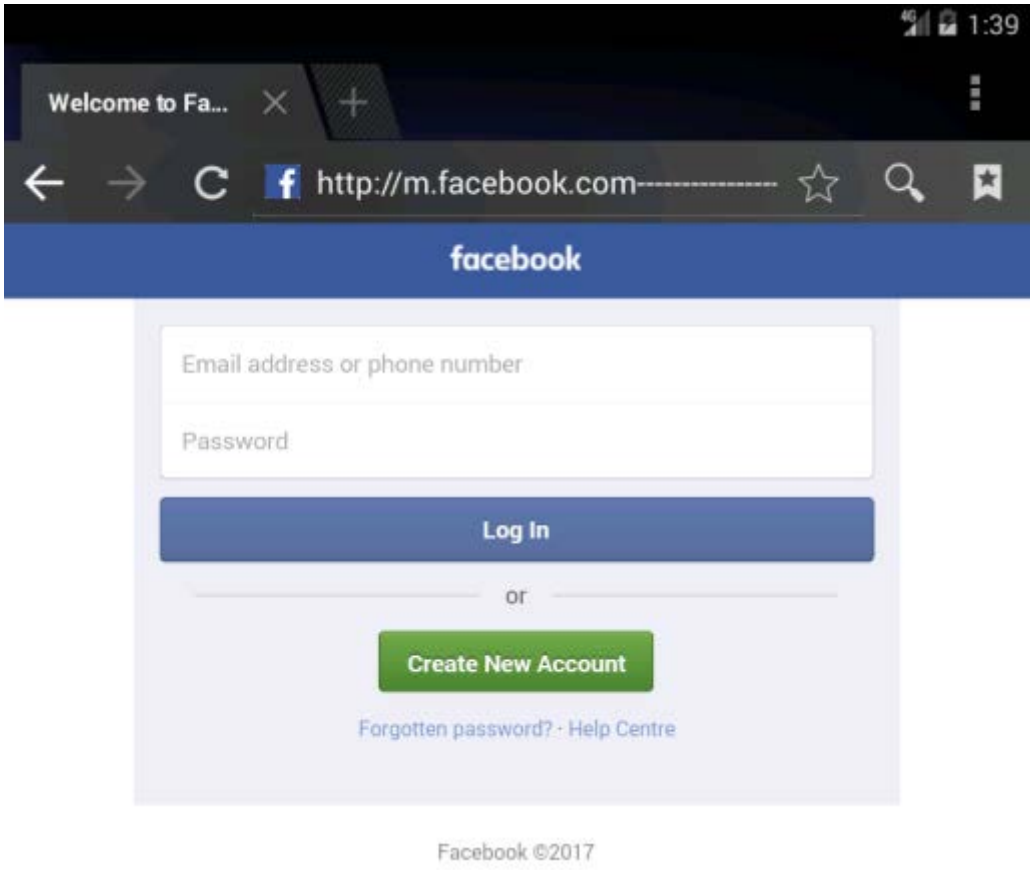
3.6 Phishing

Phishing by se dal popsat jako útok cílený na lidský faktor, který je obecně největší zranitelností většiny systémů. Nejčastěji se phishing využívá pro získání přihlašovacích údajů k internetovým službám, například internet bankingu, e-mailu, cloudu a podobně. Typicky útok probíhá tak, že útočník pošle podvrhnutý e-mail, ve kterém je odkaz na falešné stránky. Na nich

je uživatel nabádán k zadání svých přístupových údajů nebo jiných citlivých informací, což může vést ke krádeži identity.

Doména, na které útočník provozuje falešné stránky, bývá obvykle velmi podobná té originální. Často se zaměňují písmena, která stejně vypadají, nebo se písmena ve slově přehází. Příkladem může být doména `appleid.apple.com`, kterou útočník zamění za `appleid.aqqle.com`. Když se podíváme na české služby, internetové bankovnictví `csob.cz` může být zaměněno za `cscb.cz`.

Dalším způsobem jak zařídit, aby bylo pro uživatele těžší odhalit falešnou doménu, je použití URL paddingu. Principem tohoto útoku je vyplnění URL adresy znaky, například pomlčkami. Původní falešná stránka s URL `https://facebook.com-login.phishing.com` může poté vypadat jako `https://facebook.com----------login.phishing.com`. Útok na mobilních zařízeních je ještě více nebezpečný, protože na displej mobilního zařízení se nevleze celá adresa stránky, viz Obrázek 3.5. Pro uživatele je tedy těžší rozpoznat pravou stránku od falešné.



Obrázek 3.5: Ukázka URL paddingu na mobilním telefonu [28]

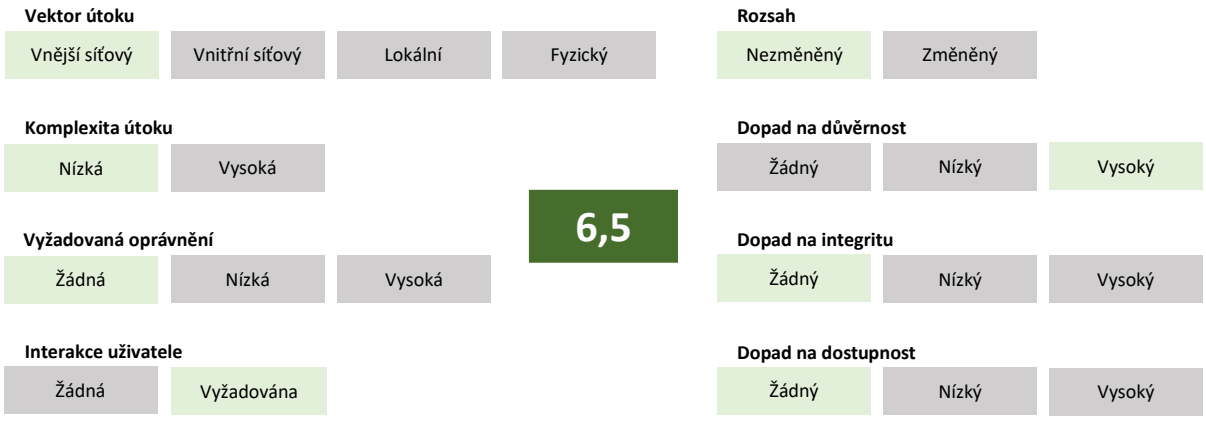
Co může způsobit?

- Odcizení přihlašovacích údajů (internet banking, e-mail, cloud, sociální sítě…);
- Odcizení citlivých informací (rodné číslo, číslo občanského průkazu, telefonní číslo…).

Ohodnocení

Tabulka 3.6: Ohodnocení hrozby phishing

Metrika	Hodnota	Popis
Vektor útoku	Vnější síťový	Útok se provádí z veřejně dostupných stránek nebo ze škodlivého e-mailu
Komplexita útoku	Nízká	Pro útočníka není problém vytvořit falešnou doménu a poté rozeslat hromadné e-maily s odkazem
Vyžadovaná oprávnění	Žádná	Nejsou vyžadována žádná oprávnění
Interakce uživatele	Vyžadována	Uživatel musí otevřít daný odkaz
Rozsah	Nezměněný	Hrozba se týká jen zařízení, na kterém je odkaz otevřen
Dopad na důvěrnost	Vysoký	Může způsobit odcizení přihlašovacích údajů
Dopad na integritu	Žádný	Nemá žádný nebo téměř žádný dopad na integritu dat
Dopad na dostupnost	Žádný	Nemá žádný nebo téměř žádný dopad na dostupnost dat



Obrázek 3.6: Ohodnocení hrozby phishing

3.7 Sniffing Wi-Fi komunikace

Wi-Fi sniffing by se dalo do češtiny přeložit jako odposlouchávání na Wi-Fi síti. Útočník má k dispozici software, tzv. sniffer, díky kterému může odposlouchávat přenášená data. Podstatou Wi-Fi sítě je její bezdrátová charakteristika, čili všechna data se přenášejí vzduchem a jsou teoreticky dostupná pro všechny zařízení v okolí. V případě nešifrované komunikace může mít útočník k dispozici všechna přenášená data (prohlížené webové stránky, přenesená uživatelská jména a hesla, e-maily apod.) Naštěstí se v dnešní době čím dál více rozmáhá používání protokolu https, který webovou komunikaci mezi prohlížečem a webovým serverem šifruje (na rozdíl od původního http protokolu).

Na mobilním telefonu však nejde jen o prohlížené stránky a přenášené údaje v rámci komunikace prohlížeč – server. Některé aplikace mohou komunikovat také přes internet, příkladem může být chatovací aplikace nebo aplikace pro sociální sítě. U těch si ale uživatel nemůže být jistý, zda aplikace šifrují svou komunikaci, nebo ne.

Co může způsobit?

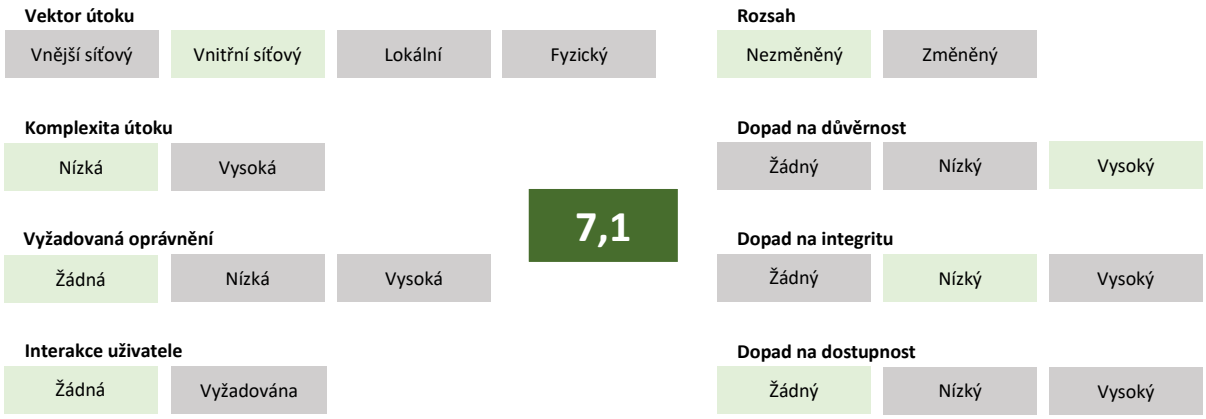
- Sledování internetové komunikace (prohlížené stránky, e-maily, zprávy posílané přes IM klienta[23]);
- Odcizení přenášených dat.

Ohodnocení

Tabulka 3.7: Ohodnocení hrozby Wi-Fi sniffing

Metrika	Hodnota	Popis
Vektor útoku	Vnitřní síťový	Útok je prováděn ve Wi-Fi síti
Komplexita útoku	Nízká	Pro útočníka je velmi jednoduché odchytávat komunikaci
Vyžadovaná oprávnění	Žádná	Nejsou vyžadována žádná oprávnění
Interakce uživatele	Žádná	Není vyžadována žádná interakce uživatele
Rozsah	Nezměněný	Hrozba se týká sice všech zařízení v síti, ale při sledování komunikace nemá útok dopad na jiná zařízení
Dopad na důvěrnost	Vysoký	Útok je primárně určen ke sledování komunikace a tedy k odcizení přenášených dat
Dopad na integritu	Nízký	Může modifikovat přenášená data, ačkoliv to není primárním účelem útoku
Dopad na dostupnost	Žádný	Nemá žádný nebo téměř žádný dopad na dostupnost dat

[23] IM = instant messaging, služba umožňující komunikaci v reálném čase, např. Facebook Messenger, Skype, WhatsApp



Obrázek 3.7: Ohodnocení hrozby Wi-Fi sniffing

3.8 Nezabezpečená Wi-Fi

Útoky pomocí falešné nezabezpečené Wi-Fi sítě jsou velmi časté v kavárnách, restauracích, na letištích a podobně. Útočník si vytvoří vlastní Wi-Fi hotspot, pojmenuje jej například „Free-wifi airport“ a jen čeká, až se připojí uživatelé, kteří hledají onu zdarma dostupnou Wi-Fi.

V tomto případě může útočník provést útok muže uprostřed, tzv. Man-in-the-Middle útok. Při běžné nezabezpečené komunikaci může odposlouchávat veškerá přenášená data. Navíc ani při použití certifikátů není zajištěno, že útočník nemůže komunikaci přečíst. Právě v tomto okamžiku využije metodu Man-in-the-Middle a jako osoba uprostřed získá certifikáty obou stran, což mu zajistí možnost dešifrovat zprávu, a certifikáty pošle opačným stranám. Ani jedna z komunikujících stran nemůže poznat, že se stala terčem útoku a že někdo odposlouchává komunikaci. Tento typ útoku může být nebezpečný pro aplikace, které nemají implementovánu kontrolu serverového certifikátu.

Nezabezpečená Wi-Fi má další rizika. Útočník může podvrhnout DNS záznamy a uživatele při přístupu na webové stránky přesměrovat na své falešné stránky. Na nich se nejčastěji snaží získat přístupové údaje k online službám.

„Domácí uživatel“ se k nezabezpečené Wi-Fi může dostat poměrně snadno kdekoliv ve veřejných prostorách. Co se týká firemních uživatelů, správně by měli být na jakékoliv veřejné Wi-Fi síti připojení na firemní VPN. Bohužel v praxi to tak nefunguje a firemní uživatelé jsou tak vystavěni stejnému riziku jako ti domácí. To stejné platí pro uživatele využívající BYOD zařízení.

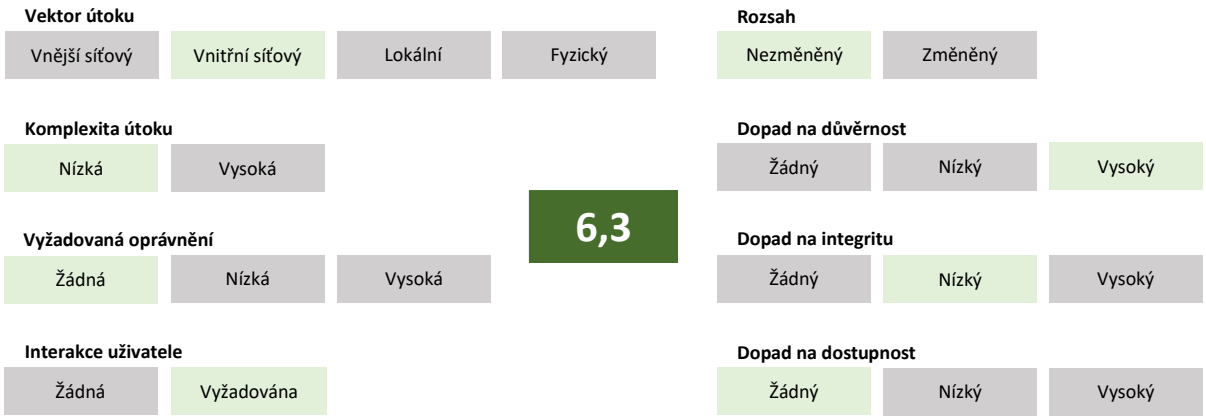
Co může způsobit?

- Odposlouchávání komunikace;
- Přesměrování na falešné stránky a získání přihlašovacích údajů.

Ohodnocení

Tabulka 3.8: Ohodnocení hrozby nezabezpečená Wi-Fi

Metrika	Hodnota	Popis
Vektor útoku	Vnitřní síťový	Útok je prováděn ve Wi-Fi síti
Komplexita útoku	Nízká	Pro útočníka není problém vytvořit falešný přístupový bod, odchytávat komunikaci nebo podvrhnout DNS záznamy
Vyžadovaná oprávnění	Žádná	Nejsou vyžadována žádná oprávnění
Interakce uživatele	Vyžadována	Uživatel se musí připojit na falešnou Wi-Fi síť
Rozsah	Nezměněný	Hrozba se týká sice všech zařízení v síti, ale při sledování komunikace nemá útok dopad na jiná zařízení
Dopad na důvěrnost	Vysoký	Útok je primárně určen ke sledování komunikace, a tedy k odcizení přenášených dat a přístupových údajů
Dopad na integritu	Nízký	Může modifikovat přenášená data, ačkoliv to není primárním účelem útoku
Dopad na dostupnost	Žádný	Nemá žádný nebo téměř žádný dopad na dostupnost dat



Obrázek 3.8: Ohodnocení hrozby nezabezpečená Wi-Fi

3.9 Odcizení nebo ztráta zařízení

Pokud je zařízení uživatele odcizeno, dá se předpokládat, že útočník bude chtít získat všechna uživatelská data. Navíc dokud nedojde k zablokování SIM karty, může útočník posílat prémiové SMS nebo volat na placená čísla, ze kterých profituje. Z běžného telefonu (bez nastaveného zámku obrazovky) může útočník získat přihlašovací údaje uložené v telefonu, SMS zprávy, e-maily, osobní fotky, dokumenty uložené v telefonu, případně v cloudu (pokud má uživatel cloudový účet spárovaný s telefonem).

Rizika uvedená v předchozím odstavci platí i při ztrátě telefonu. Pokud uživatel ztratí telefon, je to stejné, jako by mu byl odcizen. Zařízení může najít útočník se zlými úmysly a dostáváme se opět ke stejným rizikům.

Z výše uvedeného vyplývá, že ztráta zařízení je poměrně velkou hrozbou, její dopady se ale dají jednoduše zmírnit například používáním zámku obrazovky nebo funkcí Anti-Theft (funkce pro ochranu před odcizením telefonu nebo pro zmírnění následků).

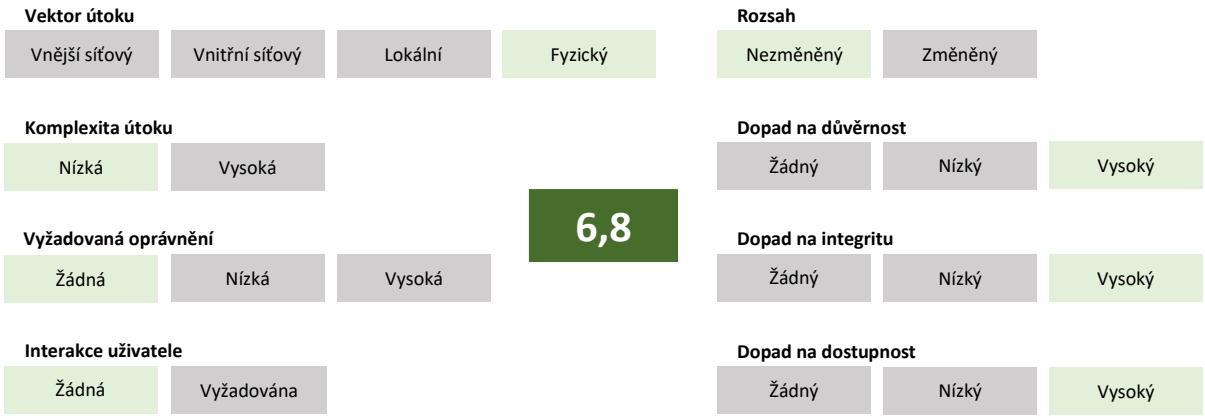
Co může způsobit?

- Odcizení přihlašovacích údajů;
- Odcizení citlivých dat;
- Přístup k fotografiím, SMS zprávám, e-mailům, dokumentům (v zařízení nebo v cloudu);
- Zasílání prémiových SMS zpráv nebo volání na placená čísla;
- Krádež identity;
- Přístup k nezabezpečeným platebním aplikacím;
- Ztráta nezálohovaných dat (především fotografie a videozáznamy);
- Instalace malwaru.

Ohodnocení

Tabulka 3.9: Ohodnocení hrozby odcizení a ztráta zařízení

Metrika	Hodnota	Popis
Vektor útoku	Fyzický	Hrozba je reálná pouze při přímé ztrátě nebo krádeži zařízení
Komplexita útoku	Nízká	Každý desátý majitel chytrého telefonu se stane obětí krádeže zařízení [29], hrozba je tedy poměrně častá
Vyžadovaná oprávnění	Žádná	Nejsou vyžadována žádná oprávnění
Interakce uživatele	Žádná	Není vyžadována žádná aktivita uživatele
Rozsah	Nezměněný	Hrozba se týká pouze ztraceného/odcizeného zařízení
Dopad na důvěrnost	Vysoký	Útočník může z telefonu získat všechna uživatelská data
Dopad na integritu	Vysoký	Útočník může využít přístupu například k sociálním sítím a ukrást identitu majitele telefonu
Dopad na dostupnost	Vysoký	V případě, že uživatel nezálohuje, může přijít o všechna svá data uložená v zařízení



Obrázek 3.9: Ohodnocení hrozby odcizení a ztráta zařízení

3.10 Ztráta nebo porušení dat

Při ztrátě nebo krádeži telefonu jsme předpokládali, že se útočník bude chtít dostat k datům uživatele. U této hrozby můžeme tento faktor pominout, předpokládáme, že zařízení máme stále v držení, pouze se ztratila nebo porušila data. K tomu může dojít mimo jiné některým z výše uvedených útoků (typickým příkladem je ransomware, který zašifruje – poruší data a ztratíme k nim přístup).

Obecně nejdůležitějšími daty (ve smyslu ztráty dat) v telefonu jsou kontakty a fotografie. Aplikace se dají doinstalovat, e-maily jsou uloženy na serverech, v SMS zprávách většinou nejsou důležité informace. Důležitá je tedy záloha primárně kontaktů, fotografií z telefonu a dokumentů, které uživatel nemá nikde jinde uloženy.

Co může způsobit?

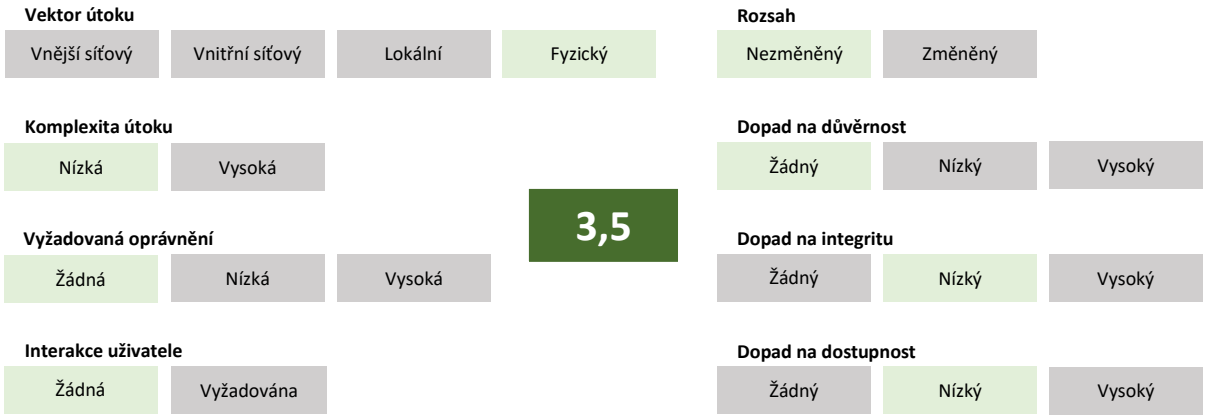
- Ztráta kontaktů, SMS zpráv, fotografií, videozáznamů;
- Porušení uložených dat a nutné obnovení zařízení do továrního nastavení.

Ohodnocení

Tabulka 3.10: Ohodnocení hrozby ztráta nebo porušení dat

Metrika	Hodnota	Popis
Vektor útoku	Fyzický	Hrozba je reálná pouze při přímé ztrátě nebo porušení dat
Komplexita útoku	Nízká	Ke ztrátě nebo porušení dat může dojít mnoha způsoby
Vyžadovaná oprávnění	Žádná	Nejsou vyžadována žádná oprávnění
Interakce uživatele	Žádná	Není vyžadována žádná aktivita uživatele
Rozsah	Nezměněný	Hrozba se týká pouze konkrétního zařízení
Dopad na důvěrnost	Žádný	Hrozba žádným způsobem nemá dopad na důvěrnost dat

Dopad na integritu	Nízký	Riziko se primárně týká ztráty kontaktů a fotografií
Dopad na dostupnost	Nízký	



Obrázek 3.10: Ohodnocení hrozby ztráta nebo porušení dat

3.11 Smishing

Pod pojmem smishing si můžeme představit phishing prováděný pomocí SMS zpráv. Jeden z případů smishingu se odehrál na začátku roku 2017 v České republice [30]. Útočníci vydávající se za Českou poštu poslali obětem SMS zprávu s informací, že jejich balíček je v depu, a pokud si ho chtějí vyzvednout, mohou tak učinit pomocí aplikace, která byla ke stažení pomocí odkazu ve zprávě. Odkaz vedl na falešné stránky vypadající stejně jako stránky České pošty. Po stáhnutí a nainstalování aplikace bylo při každém spuštění jakékoliv aplikace nutné zadat osobní údaje nebo číslo kreditní karty. Falešná aplikace útočníků tato data odesílala na jejich server.

Útok byl velice úspěšný hlavně díky tomu, jakým způsobem pracoval se sociálním inženýrstvím. Samotná SMS zpráva i podvodné stránky, na kterých se dala aplikace stáhnout, vypadaly důvěryhodně a běžný uživatel neměl pochyb o legitimnosti. K tomu se přidala náhoda, protože někteří klienti České pošty balíček opravdu čekali, a úspěšný smishingový útok byl na světě.

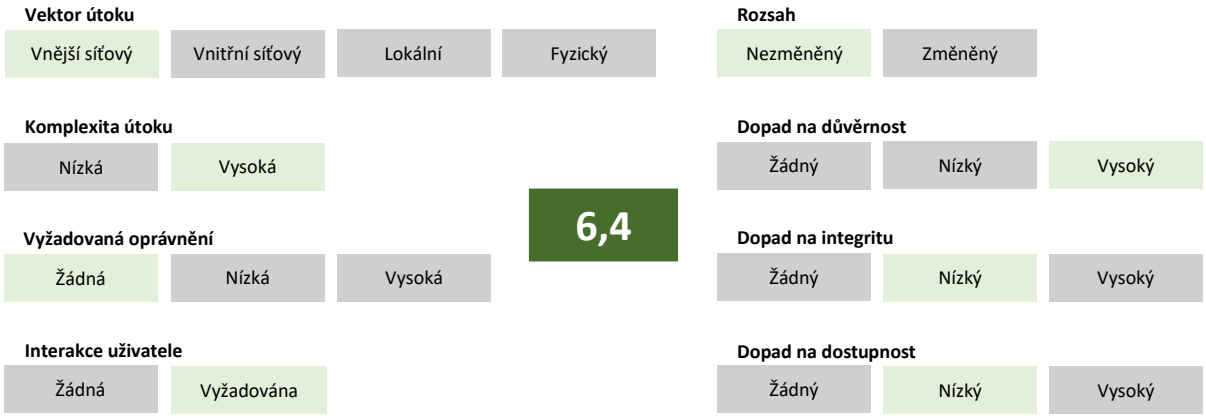
Co může způsobit?

- Odcizení přihlašovacích údajů nebo osobních údajů;
- Instalace malwaru (viz kapitola 3.2).

Ohodnocení

Tabulka 3.11: Ohodnocení útoku smishing

Metrika	Hodnota	Popis
Vektor útoku	Vnější síťový	Útok se provádí z mobilní sítě GSM
Komplexita útoku	Vysoká	Útok není náročný ani tak na přípravu a provedení, ale velmi záleží na uživateli, zda provede požadované kroky
Vyžadovaná oprávnění	Žádná	Nejsou vyžadována žádná oprávnění
Interakce uživatele	Vyžadována	Uživatel musí otevřít daný odkaz, odpovědět na SMS nebo stáhnout škodlivou aplikaci
Rozsah	Nezměněný	Hrozba se týká jen zařízení, na kterém je odkaz otevřen, případně nainstalována aplikace
Dopad na důvěrnost	Vysoký	Může způsobit odcizení přihlašovacích údajů a instalaci malwaru
Dopad na integritu	Nízký	Může způsobit instalaci malwaru
Dopad na dostupnost	Nízký	



Obrázek 3.11: Ohodnocení útoku smishing

4 Ochrana

Pro zabezpečení mobilního zařízení se systémem Android můžeme využít vestavěné funkce. Jedná se například o zámek obrazovky nebo zakázání instalací z jiných než oficiálních zdrojů. Bohužel si v dnešní době s těmito funkcemi už nevystačíme, zvlášť ne ve firemním prostředí, kde je vhodné, aby bezpečnostní správce mohl sledovat stav zabezpečení centrálně na všech zařízeních. Aplikace vhodné k zabezpečení Androidu můžeme rozdělit do dvou základních skupin – aplikace pro správu a bezpečnostní aplikace.

Mezi aplikace pro správu řadíme Enterprise Mobility Management aplikace nebo Mobile Device Management aplikace, které jsou využívané převážně ve firemním prostředí. Slouží k centrální správě mobilních zařízení, k jejich nastavení, sledování aktuálního stavu, využívání Anti-Theft technologie, jednoduššímu šíření požadovaných aplikací a podobně. Díky tomuto typu aplikací nemusí administrátor konfigurovat každé zařízení jednotlivě, stejně tak v případě instalace aplikací může instalovat aplikace centrálně na více zařízení najednou. Tyto aplikace mohou také poskytovat základní bezpečnostní funkce, již zmíněnou technologii Anti-Theft, kontrolu instalovaných aplikací, antivirovou kontrolu a podobně.

Jaký je ale rozdíl mezi EMM a MDM produkty? Jednoduše řečeno, EMM produkty v sobě obsahují nejen MDM funkce, ale také funkce Mobile Application Management (MAM), Mobile Content Management (MCM), možnosti kontejnerizace a další [31]. Čili MDM je jedna z částí EMM, avšak někteří výrobci poskytují pouze MDM řešení. V praxi se oba názvy často zaměňují, respektive se jako MDM označují všechny typy produktů pro centrální správu mobilních zařízení.

Mezi bezpečnostní aplikace řadíme všechny antivirové aplikace, pokročilá řešení Endpoint Protection Mobile, VPN aplikace nebo třeba aplikace pro ochranu před síťovými útoky.

V následujících kapitolách budou uvedeny kategorie aplikací z obou výše zmíněných skupin, které nám mohou pomoci snížit riziko hrozeb nebo snížit pravděpodobnost či dopad útoků popisovaných v kapitole 3. Budou uvedeny příklady jednotlivých aplikací jak pro osobní užití (zde se budeme zaměřovat výhradně na zdarma dostupné aplikace), tak pro firemní užití (možnost centrální správy, vyšší funkčnost vyvážená vyšší cenou).

4.1 Enterprise Mobility Management, Mobile Device Management

Řešení EMM se vyznačují centrální správou obvykle vedenou v cloudu výrobce. Zařízení se do systému mohou připojit nejčastěji odesláním e-mailu nebo SMS zprávy uživateli, kdy v samotné zprávě je mimo jiné odkaz pro stažení klientské aplikace. Ta zajišťuje kompletní funkčnost a komunikaci s centrální správou.

Produkty různých výrobců se mohou lišit v množství nabízených funkcí. Mezi ty základní, které by mělo mít každé EMM řešení, patří bezpochyby možnost vytvářet bezpečnostní politiky pro kontrolu nastavení zařízení (tzv. compliance policy), distribuce aplikací a Anti-Theft funkce (uzamknutí zařízení, lokalizace, smazání dat) [33]. Obecně jsou tyto funkce řazeny pod část MDM. Aplikace většinou neumožňují přímou změnu nastavení

zařízení, ale kontrolují nastavení a porovnávají ho s vynucovanými politikami a v případě, že se nastavení liší, je uživatel informován a požádán o změnu nastavení.

Dále může EMM nabízet řízení přístupu k firemním zdrojům (e-mail, interní aplikace, VPN, sdílené soubory atd.) nebo zakázat přístup na nezabezpečenou Wi-Fi. Pomocí Mobile Application Managementu lze u konkrétních aplikací zakázat funkci Copy&Paste, znemožnit pořizování snímků obrazovky nebo umožnit pouze číst data [34]. To je vhodné v případě, že zaměstnanec musí přistupovat k citlivým dokumentům, ale nechcete zpřístupnit funkci kopírovat. EMM řešení poskytne vlastní aplikace (webový prohlížeč, e-mailový klient, prohlížeč dokumentů), ve kterých lze výše uvedené nastavení vynutit.

Další funkcí je kontejnerizace, která umožňuje zapouzdřit firemní aplikace i data tak, aby k nim nemohly v žádném případě přistupovat ostatní aplikace v zařízení. Navíc v případě ztráty nebo odcizení zařízení lze vymazat pouze tyto kontejnerizovaná data.

Ochrana před hrozbami

Tabulka 4.1: Ochranné funkce aplikací EMM/MDM

Hrozba nebo útok	Ochranná funkce
Instalace malware	Částečně pomocí kontroly nastavení zakázání instalace aplikací ze zdrojů třetích stran
Wi-Fi sniffing	VPN
Nezabezpečená Wi-Fi	VPN
Odcizení nebo ztráta zařízení	Anti-Theft, kontrola nastavení zámku obrazovky
Odcizení firemních dat	Kontejnerizace, MAM

Příklady aplikací

- VMware AirWatch
- BlackBerry
- IBM MaaS360
- MobileIron
- Citrix XenMobile
- Microsoft Intune
- Lookout

4.2 Antivirové aplikace

Základním prvkem zabezpečení mobilního zařízení s Androidem jsou antivirové aplikace. Obsahují nejdůležitější funkce pro ochranu – antivirovou kontrolu nainstalovaných aplikací a ochranu v reálném čase. Někteří výrobci do svých free řešení přidávají další funkce, například funkci Anti-Theft (často omezenou na pouhé uzamčení telefonu a jeho lokalizaci), phishingovou kontrolu a podobně [35].

Tyto aplikace nejsou příliš vhodné do firemního prostředí, protože free verze nenabízí žádnou centrální správu ani možnost integrace na řešení EMM. Do karet těmto aplikacím

hraje pouze fakt, že jsou dostupné zdarma. Jedná se tedy o ideální volbu pro běžné uživatele, nikoliv však pro používání ve firemním prostředí [36].

Ochrana před hrozbami

Tabulka 4.2: Ochranné funkce antivirových aplikací

Hrozba nebo útok	Ochranná funkce
Instalace malware	Antivirová funkce, ochrana v reálném čase
Drive-by download	Ochrana v reálném čase
Odcizení nebo ztráta zařízení	Anti-Theft

Příklady aplikací

- Kaspersky Mobile Antivirus: AppLock & Web Security
- Mobile Security & Antivirus (ESET)
- Avira Antivirus Security
- Mobile Security & Antivirus (Trend Micro)
- Bitdefender Antivirus Free

4.3 Aplikace Mobile Endpoint Protection

Tuto kategorii můžeme pojmout jako placenou nadstavbu nad výše uvedené antivirové aplikace, protože většina z nich po zakoupení licence odemkne další funkce, které zajišťují mnohem komplexnější ochranu mobilního zařízení, než pouhou kontrolu instalovaných aplikací. Kromě výše uvedené antivirové funkce a ochrany v reálném čase obecně tyto aplikace nabízí také kontrolu procházeného webového obsahu (ochrana před phishingem a odkazy se škodlivým obsahem), filtr volání a SMS (ochrana proti smishingu), kontrolu odkazů v SMS zprávách (spadá pod anti-phishing ochranu) a vylepšenou Anti-Theft funkci [37].

Někteří výrobci poskytují také funkci Privacy Protection. Pomocí ní může uživatel kontrolovat přidělené oprávnění jednotlivým aplikacím a sama funkce uživatele automaticky upozorní na podezřelou aplikaci, která má přiřazena oprávnění, jež obvykle tento typ aplikací nepotřebuje nebo která jsou riziková.

App Locker nabízí možnost uzamknout některé aplikace heslem nebo otiskem prstu. To zajistí, že při každém spuštění aplikace je nutné zadat heslo nebo přiložit prst na čtečku otisků. Tento problém obecně řeší zámek obrazovky, protože z hlediska bezpečnosti by k mobilnímu zařízení neměl mít přístup nikdo kromě vlastníka. Existují však situace, kdy může být tato funkce vhodná pro uzamknutí aplikací typu Facebook, Messenger, SMS zprávy a podobně.

Některé produkty nabízí také funkci VPN, umožňují tedy přístup k firemní VPN. Tato funkce může být obsažena přímo v řešení (Bitdefender) nebo se může dát dokoupit (Kaspersky). Tak či tak poskytuje VPN ochranu před Wi-Fi sniffingem či možnost bezpečně přistupovat k internetu na nezabezpečené Wi-Fi.

V neposlední řadě mohou tyto aplikace nabízet kontrolu rootingu zařízení [38]. V případě detekce rootingu může aplikace informovat administrátora, zablokovat zařízení nebo z něj

vymazat všechna firemní data (v případě provázání s EMM nebo jiným řešením umožňujícím kontejnerizaci firemních dat).

Aplikace kategorie Mobile Endpoint Protection jsou vhodné do firemního prostředí. Často poskytují své vlastní MDM řešení nebo umožňují napojení na některé z EMM produktů. Díky tomu je administrátor schopný monitorovat stav zařízení, vzdáleně zařízení lokalizovat nebo zablokovat v případě jeho odcizení nebo ztráty a podobně. V případě integrace s EMM je možnost upravit politiky EMM tak, aby při detekci malwaru nebo jiného škodlivého obsahu klient EMM zablokoval přístup k firemní síti, e-mailům nebo datům.

Ochrana před hrozbami

Tabulka 4.3: Ochranné funkce aplikací Endpoint Protection

Hrozba nebo útok	Ochranná funkce
Instalace malware	Antivirová funkce, ochrana v reálném čase
Drive-by download	Ochrana v reálném čase
Odcizení nebo ztráta zařízení	Anti-Theft, kontrola nastavení zámku obrazovky
Aplikace ohrožující soukromí uživatele	Privacy Protection
Rooting zařízení	Detekce rootingu
Phishing	Anti-phishingová ochrana
Wi-Fi sniffing	VPN
Nezabezpečená Wi-Fi	VPN
Smishing	Filtr volání a SMS, Anti-phishing ochrana
Ochrana před neoprávněným přístupem k aplikacím	App Locker

Příklady aplikací

- ESET Mobile Security pro Android
- Kaspersky Security for Mobile
- Trend Micro Mobile Security for Enterprises
- Sophos Mobile Security
- Bitdefender GravityZone Security for Mobile
- Symantec Endpoint Protection Mobile

4.4 Pokročilé aplikace Mobile Endpoint Protection

Aplikace spadající do této skupiny jsou v základu podobné těm, které byly uvedeny výše ve skupině Mobile Endpoint Protection. Na rozdíl od nich však poskytují pokročilou ochranu před malwarem (známým i neznámým), před síťovými útoky jako je Man-in-the-Middle nebo SSL downgrading či sken operačního systému nebo aplikací vůči zranitelnostem [39].

Pokročila ochrana před malwarem spočívá v analýze aplikací na základě signatur i behaviorální analýze. Analýza na základě signatur se provádí převážně v samotném zařízení, zatímco behaviorální[24] se provádí především v cloudu [40]. Mimo jiné se také zkoumají oprávnění aplikací, zdroj stažení, vydavatel apod. Aplikace nepovolí instalaci aplikace, dokud nejsou známy výsledky z cloudu.

Ochrana před síťovými útoky probíhá na základě skenování síťové komunikace. V případě, kdy je zaznamenán útok nebo možnost odposlechu síťové komunikace, je zablokován přístup k této Wi-Fi síti nebo je automaticky zapnuta VPN a uživatel je přesměrován na firemní sít. O každé takovéto podezřelé síti jsou sbírány statistická data (SSID, MAC adresa přístupového bodu, další podrobnosti o síti), která jsou poté odeslána do cloudové databáze výrobce aplikace.

Pokročilá ochrana spočívá také ve skenování zranitelností jak v samotném operačním systému, tak v nainstalovaných aplikacích. Pomocí provázanosti na cloudovou databázi aplikace umožňuje detekci známých i nově objevených zranitelností. Aplikace dále kontroluje dostupnost záplat a v případě objevení aktualizované verze OS informuje uživatele nebo administrátora.[25]

Ochrana před hrozbami

Tabulka 4.4: Ochranné funkce pokročilých aplikací Endpoint Protection

Hrozba nebo útok	Ochranná funkce
Instalace malware	Pokročilá a sandboxová ochrana v reálném čase
Nezabezpečená Wi-Fi	VPN, kontrola síťové komunikace
Ochrana před síťovými útoky (MitM, SSL downgrading…)	Kontrola síťové komunikace
Exploitace systému	Skenování zranitelností
Smishing	Kontrola odkazů v SMS

Příklady aplikací

- Check Point Sandblast Mobile[26]
- Symantec Endpoint Protection Mobile[27]

[24] Behaviorální analýza může být statická nebo dynamická. Statická zkoumá kód programu, aniž by ho spustila a hledá podezřelé příkazy, které mohou být škodlivé. Dynamická neboli sandboxová se provádí v uzavřeném virtuálním prostředí, kdy se malware spustí a sleduje se jeho chování, skenují se odesílaná i přijímaná data nebo se kontroluje, k jakým datům přistupuje.

[25] Bohužel je aktuálnost operačního systému Android stále nevyřešenou věcí i z pohledu samotného Googlu. Viz kapitola 5.

[26] Aplikaci Sandblast Mobile je vhodné doplnit aplikací, která poskytuje Anti-Theft funkci, detekci rootingu, webovou anti-phidhingovou ochranu apod.

[27] Aplikace poskytuje komplexní portfolio bezpečnostních funkcí pro ochranu mobilních zařízení a hodí se tak ideálně do firemního prostředí.

4.5 Záloha dat

U klasických počítačů si firmy a snad i uživatelé zvykli důležitá data zálohovat. V případě mobilních zařízení to tak ale není. Určitě je důležité při nejmenším zálohovat kontakty. Dále někteří uživatelé požadují zálohovat SMS zprávy, fotografie a videonahrávky, případně další dokumenty vytvořené nebo spravované pomocí telefonu. Pro někoho mohou být důležité poznámky, které má v telefonu uložené, pro jiného zase všechny schůzky, které má uložené v kalendáři.

Nejjednodušším způsobem, jak mít všechna data v telefonu a zároveň v bezpečí zálohovaná, je využívat cloud. Na většině zařízeních s OS Android je předinstalovaný Google Disk, který je napojený na Google účet [43]. Díky tomu lze zálohovat nastavení telefonu, kontakty, nainstalované aplikace, kalendář, poznámky, fotografie apod. Zároveň můžete mít kdykoliv k dispozici své soubory v případě, že je máte uložené v cloudu. Mimo Google Disk existují další cloudové služby, které poskytují klienty na Android, například OneDrive nebo Dropbox.

Existují také firemní aplikace pro správu zálohování mobilních zařízení. Ty umožňují zálohovat data nejen do cloudu, ale také na připravené firemní servery [41]. Přenos dat probíhá zabezpečeně šifrovaným spojením a data mohou být také šifrovaně uložena na serverech. Navíc některé aplikace poskytují i tzv. přírůstkovou zálohu, tedy zálohování pouze změněných nebo nových dat od posledního provedení zálohy. Samozřejmostí je poskytnutí centrální správy, ze které jdou u některých řešení zálohovat nejen mobilní zařízení s Androidem, ale i Apple zařízení či notebooky s Windows.

Pro koncové uživatele, kterým nevyhovuje zálohování do cloudu, jsou určeny aplikace, které se spravují přes počítač a zálohu si tak uživatel může uložit na svůj disk do počítače [44]. Tyto aplikace jsou složitější na ovládání a z principu neposkytují pravidelné zálohování.

Ochrana před hrozbami

Tabulka 4.5: Ochranné funkce zálohovacích aplikací

Hrozba nebo útok	Ochranná funkce
Ztráta nebo porušení dat	Zálohování do cloudu
	Zálohování na interní servery firmy
	Zálohování pomocí PC

Příklady aplikací

- Cloudové řešení (Google Disk, OneDrive, Dropbox atd.)
- Firemní řešení
    - Acronis Backup a Acronis Backup Cloud
    - Asigra Cloud Backup
- Offline řešení
    - MyBackup
    - Helium
    - Manuální záloha pomocí PC

5 Problematika aktualizací OS Android

Jak již bylo nastíněno v kapitole 1.3, udržet operační systém Android aktuální může být pro uživatele či firemního administrátora problém. Google vydává každý rok novou majoritní verzi svého systému, v průběhu roku pak přicházejí minoritní verze nebo bezpečnostní aktualizace. Google je ovšem uvolňuje pouze výrobcům (případně svým vlastním zařízením, které budou zmíněny později), což znamená, že samotné rozhodnutí, zda bude aktualizace dostupná na daný model telefonu, je čistě jen na výrobci mobilního zařízení [45].

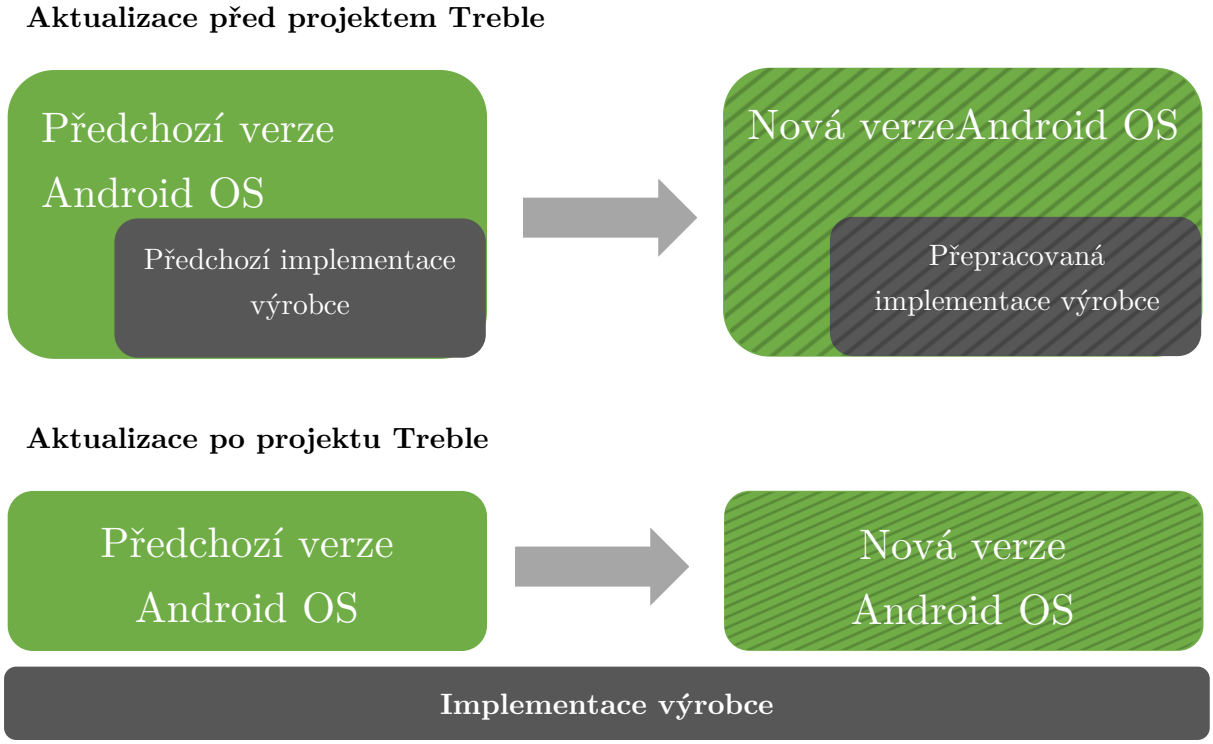
Pokud už se daný výrobce rozhodne, že aktualizace na novou verzi Androidu zpřístupní svým zákazníkům, většinou se to týká jen vybraných modelů, často nejvýkonnějších tzv. vlajkových lodí. Telefony z nižší třídy nedostávají aktualizace na nové verze systému téměř vůbec, střední třída jen u některých výrobců. Vyšší třída má obvykle slíbenou aktualizaci na následující verzi systému. Uživatele s vlajkovými loděmi si výrobci hýčkají a většinou jim nabídnou dvouletou podporu aktualizací, což znamená průměrně dvě majoritní verze.

Bezpečnostní aktualizace jsou na tom o poznání lépe, výrobci je uvolňují pro daleko větší počet svých modelů. Často ale s dlouhým zpožděním, v průběhu kterého je uživatel vystaven hrozbě zneužití zranitelnosti systému.[28]

Nastává tedy otázka, z jakého důvodu výrobci poskytují uživatelům aktualizace s takovým zpožděním, případně je neposkytují vůbec. Vše je způsobeno současnou architekturou systému, kdy operační systém není dostatečně oddělen od vrstvy HAL. Do verze Androidu 8.0 byla architektura systému taková, že vrstva HAL zajišťující komunikaci hardwarových komponent se systémem byla složena z částí, které programovali výrobci jednotlivých hardwarových komponent. Při aktualizaci na novou verzi Android museli nejdříve výrobci všech komponent upravit svou část kódu v HAL, poté výrobce zařízení vše sjednotil a umožnil aktualizaci OS.

Android 8.0 ovšem přinesl velkou změnu v architektuře, označovanou jako Project Treble [47]. Nově umožňuje pomocí HIDL (HAL Interface Definition Language) [48] změnit framework, aniž by bylo nutné provést změny ve vrstvě HAL. Jednoduše řečeno, odděluje kód jednotlivých výrobců od kódu systému tak, že jsou na sobě nezávislé a v případě aktualizace systémů stačí doprogramovat nové metody volání a není nutné měnit kompletně celý kód ve vrstvě HAL. Aktualizace systémů je tak oddělená od aktualizace implementace výrobce.

[28] Na přelomu roku 2017 a 2018 se jednalo především o zranitelnosti KRACK, Spectre a Meltdown.



Obrázek 5.1: Architektura projektu Treble [47]

Dalším důvodem, proč vycházejí aktualizace u některých výrobců mobilních zařízení se zpožděním je fakt, že si většina výrobců do systému přidává tzv. nadstavbu[29] – úpravy grafického rozhraní, přídavné speciální funkce, předinstalované aplikace, které jsou lépe integrované s hardwarem zařízení (například aplikace fotoaparátu dodaná výrobcem zařízení umí lépe využít potenciál fotoaparátu a vytvořit lepší fotky než aplikace v čisté verzi Androidu). Navíc pomocí grafické nadstavby se výrobci odlišují ve vzhledu svého prostředí. Bohužel, to je co do výčtu výhod nejspíše vše.

Nevýhod nadstaveb Androidu není možná ani o tolik více, rozhodně jsou ale důležitější [46]. Předinstalované aplikace nelze běžným způsobem odinstalovat, pouze po rootnutí telefonu je lze smazat, v takovém případě ale uživatel přichází o záruku na telefon. Další nevýhodou je to, že aktualizace na novou verzi Androidu musí jít ruku v ruce s aktualizací nadstavby. Google při programování nové verze OS pracuje tak, aby upgrade proběhl co nejhladčeji a bez problémů. Už se ale nedívá na nadstavby výrobců. Ty jsou někdy tak obsáhlé a těžkopádné, že zkrátka nelze aktualizovat OS, aniž by byla upravena i samotná nadstavba. Z toho důvodu výrobce mobilního telefonu uvolní aktualizaci systému s dlouhým odstupem společně s aktualizací svojí nadstavby. V některých případech to ale není možné vzhledem k náročnosti úpravy staré nadstavby, a tak vytvoří novou, která je kompatibilní s nově vydanou verzí OS, ale už nepovolí aktualizace staré nadstavby na novou.

[29] Mezi používané nadstavby patří například Samsung Experience od Samsungu, EMUI od Huawei nebo MIUI od Xiaomi.

5.1 Řešení problémů s aktualizacemi

Jednou z možností, jak se vyhnout problémům s aktualizacemi, je používat telefony s čistým systémem, tedy bez nadstavby. Update jejich systému je jednodušší, ovšem ani tam není zaručeno, že budou nové verze dostupné ihned po jejich vydání po celou dobu podporované životnosti telefonu. Navíc není vyřešen problém s aktualizací komponent HAL vrstvy.

Druhou možností je používat telefony s Androidem 8.0 Oreo nebo vyšší verzí. Bohužel výrobci na tuto verzi OS přecházejí velmi pomalu. Navíc při aktualizaci si mohou vybrat, zda chtějí zahrnout do aktualizace i projekt Treble. U starších telefonů, které měly starší verzi Androidu, probíhá změna architektury velmi pracně a výrobci se jí vyhýbají. Některé nové telefony, které již od základu mají Android ve verzi 8.0 a vyšší, projekt Treble zabudovaný mají. Stále to je ale poměrně malá část, vůči všem zařízením s touto verzí systému [49].

Další možností je využívat mobilní zařízení zapojené do projektu Android One. Původně jsou to zařízení určená do zemí třetího světa, avšak najdou se i kvalitní zařízení, která jsou do tohoto projektu zapojena. Nabízejí čistý Android a výrobce garantuje podporu aktualizací systému po dobu dvou let. Vzhledem ke stejné architektuře, jakou měly modely s Androidem nižším než verze 8.0, se musí počítat s určitým zpožděním při uvolňování aktualizací.

Poslední a nejlepší možností je využívat mobilní zařízení od samotného Googlu. Díky tomu, že za výrobou jak mobilního telefonu, tak operačního systému stojí tatáž firma, aktualizace probíhají bez problému. Můžeme vidět tedy něco podobného, co nabízí společnost Apple u svých iPhonů a systému iOS. Modelovou třídu těchto zařízení Google pojmenoval Pixel (dříve Nexus). Kromě garantované podpory aktualizací a jejich rychlého vypouštění do světa nabízejí samozřejmě čistý Android bez nadstavby, nejnovější hardware nebo dodatečné funkce, které jiní výrobci nenabízejí. Nevýhodou tohoto řešení je poměrně vysoká cena těchto telefonů. Pro firemní prostředí je to však nejlepší volba.

6 Laboratorní úloha

K představení některých možností, jak ochránit mobilní zařízení před některými z výše uvedených hrozeb, byla vytvořena laboratorní úloha pro studenty. V rámci laboratorní úlohy se student seznámí s řešením Enterprise Mobility Management (EMM) od společnosti IBM, konkrétně se jedná o produkt IBM MaaS360. V tomto produktu nastaví bezpečnostní politiky pro platformu Android dle zadání. Dále nakonfiguruje Mobile Application Management (MAM) a skrz MaaS360 nahraje antivirovou aplikaci Avast Antivir a ochrana mobilu do telefonu. V druhé části si vyzkouší test antivirové aplikace pomocí Mobile Security Virus Test od společnosti TrustPort a zkusí porušit zásady vyžadované bezpečnostní politikou.

Teoretickou část laboratorní úlohy tvoří popis EMM a popis konkrétního řešení MaaS360. Dále se student seznámí s testováním detekce antivirových produktů pomocí testovacího souboru Eicar, který je implementován právě v testovací aplikaci. Poslední teoretickou část tvoří popis CIS politik, k čemu se v praxi využívají a jak vypadá CIS politika pro Android 7.x.

Kompletní laboratorní úloha je uvedena v příloze A.

6.1 Zmírněné hrozby

Úkoly v laboratorní práci směřují ke zmírnění dopadů některých hrozeb. Nastavením zámku obrazovky lze zamezit útočníkovi v přístupu k zařízení v případě jeho ztráty nebo odcizení. Zakázáním instalace aplikací z neznámých zdrojů lze zmírnit hrozbu malwaru, stále je ale riziko instalace malware z Google Play. Instalací antivirové aplikace lze ještě více zmírnit hrozbu malwarové nákazy a mimo jiné obsahuje aplikace také nástroj pro analýzu zabezpečení Wi-Fi sítě.

Samotná aplikace MaaS360 disponuje funkcí Anti-Theft, která umožňuje při ztrátě nebo odcizení zařízení lokalizovat, uzamknout, spustit alarm, poslat na něj zprávu nebo smazat data. Tato funkce umožňuje například zamknout telefon ještě před tím, než se útočníkovi podaří kompromitovat zařízení a získat z něj důležitá data.

6.2 Vybrané produkty

Produkt IBM MaaS360 byl vybrán z toho důvodu, že jako jeden z mála umožňuje získat trial verzi ihned po registraci. Student se tedy zaregistruje na stránkách IBM a na e-mail mu dojde pozvánka do cloudového prostředí aplikace.

Avast Antivir a ochrana mobilu byl vybrán jako zástupce antivirových produktů na platformě Android. Je zdarma a je tedy ideální na testovací účely. Jako testovací aplikace byla vybrána Mobile Security Virus Test, jelikož je účelná a jednoduchá a pro naši demonstraci úplně stačí.

6.3 Předpokládané znalosti studenta

Po studentovi nejsou vyžadovány žádné podrobné znalosti problematiky. Dostačuje základní přehled o fungování operačního systému Android a pasivní znalost anglického jazyka (rozhraní aplikace MaaS360 je v angličtině).

6.4 Výstup laboratorní úlohy

Není vyžadována žádná tvorba laboratorního protokolu. Cílem práce je, aby se student seznámil s řešením EMM, konkrétně s možnostmi nastavení politik, Mobile Application Managementu, správou zařízení apod.

6.5 Prostředí a požadavky

Aplikace MaaS360 je cloudového typu a pro její ovládání stačí běžný prohlížeč. Operační systém Android je dostupný jako ISO image a dá se nahrát do připraveného virtuálního stroje VMware nebo VirtualBox [50].

Pro práci s operačním systémem Android je potřeba vytvořit Google účet pro uživatele. Pomocí tohoto e-mailu bude telefon přidán do systému MaaS360.

7 Závěr

Tato práce měla za cíl přivést čtenáře do světa problematiky bezpečnosti operačního systému Android. Prvním cílem bylo vysvětlit principy základních bezpečnostních mechanizmů na této platformě, například aplikační sandboxing, oprávnění aplikací nebo jak probíhá autentizace uživatele. Tento teoretický popis přináší kapitola 2.

Dále práce poskytla souhrn nejběžnějších hrozeb, se kterými se můžeme na Androidu potkat, a zároveň se pokusila hrozby ohodnotit. Metodika pro ohodnocení hrozeb vycházela z veřejně dostupného hodnocení zranitelností systémem CVSS. Výsledkem tohoto hodnocení je, že za největší hrozbu pro zařízení se systémem Android můžeme pokládat instalaci malwaru, která může probíhat jak ze zdrojů třetích stran, tak ze samotného Google Play. Následně může útočník skrz škodlivou aplikaci instalovat další aplikace, odposlouchávat přihlašovací údaje do jiných aplikací nebo do mobilního bankovnictví, číst SMS zprávy atd. Další dvě velké hrozby, na které si musí uživatelé dávat pozor, jsou drive-by download, který opět vede k instalaci nebo spuštění malwaru, a Wi-Fi sniffing, který může vést nejen k odcizení přihlašovacích údajů, ale ke kompletnímu odposlouchávání komunikace. Seřazené hrozby s jejich hodnocením viz Tabulka 7.1.

Tabulka 7.1: Souhrn ohodnocení hrozeb

Hrozba nebo útok	Ohodnocení dle CVSS
Malware	**7,3**
Drive-by download	**7,1**
Wi-Fi sniffing	**7,1**
Odcizení nebo ztráta zařízení	6,8
Phishing	6,5
Smishing	6,4
Nezabezpečená Wi-Fi	6,3
Aplikace ohrožující soukromí uživatele	5,0
Rooting zařízení	4,2
Ztráta nebo porušení dat	3,5

V návaznosti na uvedené a popsané hrozby následoval popis aplikací, které mohou především ve firemní sféře pomoci se správou mobilních zařízení nebo s jejich zabezpečením. V kapitole 4 jsou uvedeny kategorie bezpečnostních aplikací nebo aplikací pro správu, popsány základní funkce, které by dané aplikace měly mít a příklady takových aplikací. Pro firemní prostředí vyplývá, že vhodnou kombinací je některé EMM řešení společně s aplikací Mobile Endpoint Protection. Také se nesmí zapomenout na zálohování mobilních zařízení, čemuž se věnuje jedna z podkapitol.

V současnosti stále velký problém je roztříštěnost Androidu na jednotlivé verze. Této problematice se věnuje kapitola 5, ve které je popsán důvod roztříštěnosti a těžkopádného šíření aktualizací. Řešení se nabízí několik, z nichž nejlepší je využívání mobilních telefonů

přímo od Googlu. Tyto zařízení mají garantovanou dvouletou podporu. V této lhůtě dostávají bezpečnostní aktualizace i aktualizace na novější verze systému. Vzhledem k tomu, že systém i hardware zařízení má na starosti jeden výrobce, aktualizace jsou vydávány pravidelně a bez zbytečné prodlevy.

Cílem práce bylo také vytvořit laboratorní úlohu pro demonstraci řešení Enterprise Mobility Management a prezentaci jako učební text pro získání znalostí o této problematice. Laboratorní práce je sestavena tak, aby se studenti v průběhu jejího řešení seznámili s produktem pro správu mobilních zařízení. Tuto aplikaci poté využijí pro správu bezpečnostních politik mobilních zařízení, instalaci aplikací a kontrolu souladu s pravidly. V rámci úlohy se také seznámí s testováním antivirových programů pomocí eicaru, aplikaci založenou na tomto projektu si vyzkouší nainstalovat a detekovat funkčnost antivirové aplikace. Dále je popsán princip CIS politik, jakým způsobem se využívají a jak vypadá politika pro Android 7. Prezentace je obsahově totožná s touto prací, ale je přiměřeně zestručněna.

Literatura

[1] Consumer Security Risks Survey 2016. *Kaspersky.com* [online]. 2017 [cit. 2017-12-05]. Dostupné z: https://press.kaspersky.com/files/2017/05/B2C_survey_2016_report_print.pdf

[2] Comodo Mobile Security. *Příspěvek na oficiální stránce společnosti na webu Facebook.com* [online]. [cit. 2017-12-05]. Dostupné z: https://www.facebook.com/ComodoHome/photos/a.10150109766353005.288939.49777068004/10153784872728005/?type=3&theater

[3] UNUCHEK, Roman, Fedor SINITSYN, Denis PARINOV a Alexander LISKIN. IT threat evolution Q3 2017: Statistics. *Securelist* [online]. 2017 [cit. 2017-12-05]. Dostupné z: https://securelist.com/it-threat-evolution-q3-2017-statistics/83131

[4] LOSERT, Filip. BYOD Z PERSPEKTIVY GDPR. *Advokátní kancelář Jelínek* [online]. [cit. 2018-04-25]. Dostupné z: https://www.advokatijelinek.cz/byod-z-perspektivy-gdpr.html

[5] Mobile Operating System Market Share Worldwide. *Statcounter GlobalStats* [online]. 2017 [cit. 2017-10-12]. Dostupné z: http://gs.statcounter.com/os-market-share/mobile/worldwide

[6] Operating System Market Share Worldwide. *Statcounter GlobalStats* [online]. 2017 [cit. 2017-10-12]. Dostupné z: http://gs.statcounter.com/os-market-share

[7] Dashboards. *Android Developers* [online]. [cit. 2017-12-05]. Dostupné z: https://developer.android.com/about/dashboards/index.html

[8] OLMSTEAD, Kenneth a Aaron SMITH. Americans and Cybersecurity. *Pew Research Center* [online]. 2017 [cit. 2017-12-05]. Dostupné z: http://www.pewinternet.org/2017/01/26/americans-and-cybersecurity

[9] FORD, Bryan a Russ COX. *Vx32: Lightweight, User-level Sandboxing on the x86* [online]. Massachusetts Institute of Technology [cit. 2017-12-05]. Dostupné z: https://pdos.csail.mit.edu/papers/vx32:usenix08

[10] ELENKOV, Nikolay. Android security internals: an in-depth guide to Android's security architecture. San Francisco: No Starch Press, 2015. ISBN 9781593275815.

[11] Sign Your App. Android Studio [online]. [cit. 2018-04-07]. Dostupné z: https://developer.android.com/studio/publish/app-signing.html

a. Ikona klíče byla vytvořena autorem SimpleIcon z www.flaticon.com a distribuována pod licencí CC 3.0 BY.
b. Ikona Androidu byla vytvořena autorem Freepik z www.flaticon.com a distribuována pod licencí CC 3.0 BY.

[12] Application Signing. *Security* [online]. [cit. 2018-04-07]. Dostupné z: https://source.android.com/security/apksigning/

[13] Verified Boot. *Security* [online]. [cit. 2018-04-07]. Dostupné z: https://source.android.com/security/verifiedboot/

[14] KHANDELWAL, Swati. What is Strictly Enforced Verified Boot in Android 7.0 Nougat?. The Hacker News [online]. 2016 [cit. 2018-04-07]. Dostupné z: https://thehackernews.com/2016/07/android-verified-boot.html

[15] Encryption. Security [online]. [cit. 2018-04-07]. Dostupné z: https://source.android.com/security/encryption/

[16] Authentication. Security [online]. [cit. 2018-04-07]. Dostupné z: https://source.android.com/security/authentication/

[17] Trusty TEE. Security [online]. [cit. 2018-04-07]. Dostupné z: https://source.android.com/security/trusty/

[18] LØGE, Marte Dybevik. "Tell Me Who You Are and I Will Tell You Your Unlock Pattern". Trondheim, 2015. Diplomová práce. Norwegian University of Science and Technology. Vedoucí práce Lillian Røstad, IDI. Dostupné z: http://hdl.handle.net/11250/2380967

[19] LEWKOWICZ, Kayla. State of email report. *Litmus* [online]. 2017, 80 s. [cit. 2017-12-05]. Dostupné z: https://litmus.com/lp/state-of-email-2017

[20] AIELLO, Luca Maria, Mihajlo GRBOVIC, Amin MANTRACH, Farshad KOOTI a Kristina LERMAN. Evolution of Conversations in the Age of Email Overload. In: *Research Yahoo!* [online]. Příspěvek na International World Wide Web Conference, 2015 [cit. 2017-12-05]. Dostupné z: https://research.yahoo.com/publications/6757/evolution-conversations-age-email-overload

[21] What is a mobile threat? Lookout [online]. [cit. 2017-12-05]. Dostupné z: https://www.lookout.com/know-your-mobile/what-is-a-mobile-threat

[22] AMIT, Yair a Adi SHARABANI. *Mobile Security Attacks: A Glimpse From the Trenches* [online]. Příspěvek na konferenci RSA Conference 2014 Asia Pacific & Japan, 2014 [cit. 2017-12-05]. Dostupné z: https://www.rsaconference.com/writable/presentations/file_upload/mbs-w06-mobile-security-attacks-a-glimpse-from-the-trenches.pdf

[23] Common Vulnerability Scoring System SIG. *First: Improving Security Together* [online]. [cit. 2017-12-05]. Dostupné z: https://www.first.org/cvss

[24] UNUCHEK, Roman. Mobile malware evolution 2016. *Securelist* [online]. 2017 [cit. 2017-12-05]. Dostupné z: https://securelist.com/mobile-malware-evolution-2016/77681

[25] ASHFORD, Warwick. Free mobile apps a threat to privacy, study finds. *ComputerWeekly* [online]. 2012 [cit. 2017-12-05]. Dostupné z: http://www.computerweekly.com/news/2240169770/Free-mobile-apps-a-threat-to-privacy-study-finds

[26] KOROLOV, Maria. „Legitimate“ rooting apps paving way for malware. *ComputerWeekly* [online]. 2015 [cit. 2017-12-05]. Dostupné z: https://www.csoonline.com/article/2993454/mobile-security/legitimate-rooting-apps-paving-way-for-malware.html

[27] APVRILLE, Axelle. New Drive-By Download Android Malware. *Blog Fortinet* [online]. 2014 [cit. 2017-12-05]. Dostupné z: https://blog.fortinet.com/2014/02/16/new-drive-by-download-android-malware

[28] JEZIORSKÝ, Tomáš. URL padding. *S3C.cz* [online]. 2017 [cit. 2017-12-05]. Dostupné z: https://www.s3c.cz/blog/posts/url-padding

[29] Phone Theft in America: What really happens when your phone gets grabbed. *Blog Lookout* [online]. 2014 [cit. 2017-12-05]. Dostupné z: https://blog.lookout.com/phone-theft-in-america

[30] CHECK POINT MOBILE THREAT RESEARCH TEAM. The SMISHING threat – unraveling the details of an attack. *Blog Check Point*[online]. 2017 [cit. 2017-12-05]. Dostupné z: https://blog.checkpoint.com/2017/02/09/smishing-threat-unraveling-details-attack

[31] O'DOWD, Elizabeth. What's the Difference Between EMM and MDM Anyway?. Solutions Review [online]. 2016 [cit. 2018-04-07]. Dostupné z: https://solutionsreview.com/mobile-device-management/whats-the-difference-between-emm-and-mdm-anyway/

[32] MEARIAN, Lucas. What's the difference between MDM, MAM, EMM and UEM?. *ComputerWorld* [online]. 2017 [cit. 2018-04-07]. Dostupné z: https://www.computerworld.com/article/3206325/mobile-wireless/whats-the-difference-between-mdm-mam-emm-and-uem.html

[33] Productivity Apps. VMware AirWatch [online]. [cit. 2018-04-07]. Dostupné z: https://www.air-watch.com/solutions/productivity-apps/

[34] Dokumentace k produktu VMware AirWatch. *Docs VMware* [online]. [cit. 2018-04-07]. Dostupné z: https://docs.vmware.com/en/VMware-AirWatch/index.html

[35] Antivirus pro mobilní zařízení s Androidem. *ESET* [online]. [cit. 2018-04-07]. Dostupné z: https://www.eset.com/cz/domacnosti/android-antivirus/

[36] Popis aplikací na portále Google Play. *Google Play* [online]. [cit. 2018-04-07]. Dostupné z: https://play.google.com

[37] Mobile Security Solutions. *Trend Micro* [online]. [cit. 2018-04-07]. Dostupné z: https://www.trendmicro.com/en_us/forHome/products/mobile-security.html

[38] Kaspersky Security for Mobile. *Kaspersky Lab* [online]. [cit. 2018-04-07]. Dostupné z: https://www.kaspersky.com/small-to-medium-business-security/mobile

[39] Symantec Endpoint Protection Mobile. *Symantec* [online]. [cit. 2018-04-07]. Dostupné z: https://www.symantec.com/products/endpoint-protection-mobile

[40] Mobile Threat Defense. Check Point [online]. [cit. 2018-04-07]. Dostupné z: https://www.checkpoint.com/products/mobile-threat-defense/

[41] Mobile Device Backup and Data Protection Solutions. *Acronis* [online]. [cit. 2018-04-07]. Dostupné z: https://www.acronis.com/en-us/business/enterprise-solutions/mobile/

[42] Data Protection Solutions. *Asigra* [online]. [cit. 2018-04-07]. Dostupné z: https://www.asigra.com/solutions/enterprise-cloud-backup-and-recovery-software-desktops-laptops-tablets-and-smartphones

[43] Manage & restore your device backups in Google Drive. *Google Support* [online]. [cit. 2018-04-07]. Dostupné z: https://support.google.com/drive/answer/6305834?co=GENIE.Platform%3DAndroid&hl=en

[44] HINDY, Joe. 10 best Android backup apps and other ways to backup Android!. *Android Authority* [online]. 2018 [cit. 2018-04-07]. Dostupné z: https://www.androidauthority.com/best-android-backup-apps-and-other-ways-too-608014/

[45] TRLICA, David. Aktualizace na Android Oreo je problémová: Nyní ji pozastavil i Samsung. *Svět Androida* [online]. 2018 [cit. 2018-04-07]. Dostupné z: https://www.svetandroida.cz/aktualizace-na-android-oreo-samsung-201802/

[46] SOBOTKA, Jakub. Čistý Android, nebo upravená nadstavba od dalších výrobců? Ukážeme si klady a zápory. *Techbrain* [online]. 2015 [cit. 2018-04-07]. Dostupné z: https://techbrain.cz/2015/09/cisty-android-nebo-upravena-nadstavba-od-dalsich-vyrobcu-ukazeme-si-klady-a-zapory/

[47] TRIGGS, Robert. Understanding Project Treble and future Android updates. *Android Authority* [online]. 2017 [cit. 2018-04-07]. Dostupné z: https://www.androidauthority.com/project-treble-818225/

[48] HIDL. *Porting* [online]. [cit. 2018-04-07]. Dostupné z: https://source.android.com/devices/architecture/hidl/

[49] DAVENPORT, Corbin. Here are all the phones updated to support Project Treble. *Android Police* [online]. 2017 [cit. 2018-04-07]. Dostupné z: https://www.androidpolice.com/2017/11/26/phones-updated-support-project-treble-continuously-updated/

[50] Download. *Android-x86 - Porting Android to x86* [online]. [cit. 2017-10-12]. Dostupné z: http://www.android-x86.org/download

[51] STALLINGS, William. *Cryptography and network security: principles and practice*. 4th ed. Upper Saddle River, N.J.: Pearson/Prentice Hall, c2006. ISBN 0-13-187316-4

[52] ANDROID ATC TEAM. *Android Security Essentials*. Android ATC, 2015. ISBN 978-0-9900143-5-5.

[53] The Android Story. *Android* [online]. 2014 [cit. 2017-10-12]. Dostupné z: https://www.android.com/history/#/donut

[54] Android 2.2 Platform Highlights. *Android Developers* [online]. [cit. 2017-10-12]. Dostupné z: https://developer.android.com/about/versions/android-2.2-highlights.html

[55] Platform Architecture. *Android Developers* [online]. [cit. 2017-12-05]. Dostupné z: https://developer.android.com/guide/platform/index.html#system-apps

[56] Architecture. *Android Source* [online]. 2017 [cit. 2017-10-12]. Dostupné z: https://source.android.com/devices/architecture

[57] System Permissions. *Android Developers* [online]. [cit. 2017-10-12]. Dostupné z: https://developer.android.com/guide/topics/permissions/index.html

[58] Security. *Android Source* [online]. 2017 [cit. 2017-10-12]. Dostupné z: https://source.android.com/security

[59] LEAVITT, Neal. Mobile Security: Finally a Serious Problem? *Computer* [online]. 2011, 2011(6), 11–14 [cit. 2017-12-05]. DOI: 10.1109/MC.2011.184. ISSN 0018-9162. Dostupné z: http://ieeexplore.ieee.org/document/5875929

[60] The risks of rooting your Android phone – BullGuard. *BullGuard* [online]. [cit. 2017-12-05]. Dostupné z: https://www.bullguard.com/bullguard-security-center/mobile-security/mobile-threats/android-rooting-risks.aspx

[61] MERCATO, Mattia. 5 reasons not to root your Android device. *AndroidPIT* [online]. 2014 [cit. 2017-12-05]. Dostupné z: https://www.androidpit.com/5-reasons-not-to-root-your-device

[62] BAMBURIC, Mihăiță. 5 reasons not to root Android. *Betanews* [online]. 2013 [cit. 2017-12-05]. Dostupné z: https://betanews.com/2013/10/01/5-reasons-not-to-root-android

[63] Introducing IBM MaaS360 with Watson. *IBM* [online]. [cit. 2017-12-05]. Dostupné z: https://www.ibm.com/security/mobile/maas360

[64] CIS benchmarks. *Center for Internet Security* [online]. [cit. 2017-12-05]. Dostupné z: https://www.cisecurity.org/cis-benchmarks

[65] CIS Google Android 7 Benchmark. *Center for Internet Security* [online]. 2017 [cit. 2017-12-05]. Dostupné z: https://www.cisecurity.org/cis-benchmarks

[66] INTENDED USE. *Eicar* [online]. [cit. 2017-12-05]. Dostupné z: http://www.eicar.org/86-0-Intended-use.html

Seznam zkratek

ACL	Access Control List
AES	Advanced Encryption Standard
API	rozhraní pro programování aplikací
APK	formát balíčku, který se používá na Androidu pro distribuci a instalaci aplikací
ART	Android Runtime
BYOD	Bring Your Own Device
CBC	Cipher Block Chaining
CE	Credential Encrypted
CIS	Center for Internet Security
CVSS	Common Vulnerability Scoring System
DAC	Discretionary Access Control
DE	Device Encrypted
DEK	Disk Encryption Key
DNS	systém doménových jmen
EMM	Enterprise Mobility Management
FBE	File-Based Encryption
FDE	Full-Disk Encryption
GSM	globální systém pro mobilní komunikaci
HAL	Hardware Abstraction Layer
HIDL	HAL Interface Definition Language
HTTP	protokol určený pro výměnu hypertextových dokumentů ve formátu HTML
HTTPS	protokol určený pro zabezpečenou výměnu hypertextových dokumentů ve formátu HTML
IM	Instant Messaging
iOS	mobilní operační systém firmy Apple
IPC	meziprocesová komunikace
ISO	soubor obsahující digitální kopii dat z optického disku
JVM	Java Virtual Machine
KEK	Key Encryption Key

KRACK	Key Reinstallation Attack
LAN	lokální počítačová síť
MAC	Mandatory Access Control
MAC adresa	Media Access Control, adresa na druhé vrstvě modelu ISO/OSI
MAM	Mobile Application Management
MCM	Mobile Content Management
MDM	Mobile Device Management
MitM	útok typu Man-in-the-Middle
OEM	Original Equipment Manufacturer
OHA	Open Handset Alliance
OS	operační systém
PC	osobní počítač
PIN	číselný kód pro zabezpečení zařízení
RSA	asymetrická bloková šifra
SIM	Subscriber Identity Module (SIM karta)
SMS	označení pro krátké zprávy posílané pomocí GSM
SSID	Service Set Identifier, označení pro název Wi-Fi sítě
SSL	Secure Sockets Layer
TEE	Trusted Execution Environment
UI	uživatelské rozhraní
UID	identifikátor sloužící k identifikaci jednotlivých aplikací
URL	identifikátor sloužící ke specifikaci umístění zdrojů informací v síti Internet
User ID	identifikátor sloužící k identifikaci jednotlivých uživatelů
user SID	identifikátor sloužící k identifikaci uživatele
VPN	virtuální privátní síť
WAN	počítačová síť, která pokrývá rozsáhlé území
Wi-Fi	standard popisující bezdrátovou komunikaci v počítačových sítích