REPOSITOR, Vol. 2, No. 11, November 2020, Pp. 1554-1561
ISSN : 2714-7975
E-ISSN : 2716-1382

Analisis Patch Keamanan Android Terhadap Serangan Juice Jacking

Amru Rizal F.S[1], Dwi Setiawan[2]
Universitas Amikom Yogyakarta[1, 2]
amru.21@students.amikom.ac.id[1], dwi.setiawan@ students.amikom.ac.id[2]

Abstrak

Saat ini smartphone merupakan alat bantu terpenting dalam mendukung aktifitas sosial dan pekerjaan sehari-hari. Melihat pentingnya smartphone untuk melakukan aktifitas tersebut, banyak perusahaan memberikan layanan gratis untuk memendukung kebutuhan smartphon, seperti halnya layanan pengisian baterai pada tempat tempat umum. Namun, pengisian daya di tempat umum belum bisa dikatakan aman karena bisa menjadi tempat bagi para penjahat cyber untuk melakukan serangan yang terutama serangan saat pengisian, untuk mencuri informasi pribadi pengguna ponsel dengan memanfaatkan port USB pada smartphone. Serangan Juice Jacking adalah salah satu ancaman yang dapat merujuk informasi pribadi pengguna dari perangkat Android OS, dengan melakukan pemantauan dan perekaman layar ponsel secara otomatis selama pengisian daya smartphone. Karena potensi yang ditimbulkan oleh Juice Jacking memiliki pengaruh yang besar, kami melakukan analisis keamanan patch android terbaru dengan tujuan untuk mengetahui tingkat keamanan android terhadap serangan juice jacking, serta memberikan kesadaran kepada para pengguna smartphone untuk lebih berhati-hati dalam memanfaatkan fasilitas umum yang belum terjamin.

Kata kunci: *Juice Jacking, Malware,Android.*

Abstract

Nowadays smartphones are the main tools to support social activities and daily work. Seeing the importance of smartphones to carry out these activities, many companies provide free services to support smartphone needs, as well as battery charging support services in public places. However, charging in public places cannot be safely resolved because it can be a place for cyber criminals to carry out attacks that require attacks, to access personal information of mobile users by using a USB port on a smartphone. Juice Jacking Attack is one of the protections that can help users' personal information from an Android OS device, by monitoring and recording the cellphone screen using automatically during smartphone charging. Because the potential caused by Juice Jacking has a great effect, we conducted a security analysis of the latest Android patch with the aim to determine the level of Android security against juice jacking attacks, as well as giving awareness to smartphone users to be more careful in using facilities that have not been guaranteed.

Keywords*: Juice Jacking, Malware, Android.*

1. Pendahuluan

Android merupakan ponsel sumber terbuka komprehensif pertama yang sistem operasinya ditujukan untuk pasar konsumen secara global. Ini tentu menangkap perhatian para penyedia layanan di Eropa dan AS karena fitur ini yang sangat menarik. Kami melakukan analasisa terhadap Android, karena penerimaannya di seluruh dunia, basis pengetahuannya tentang kebijakan keamanan, dan keterbukaannya. Keterbukaan Android mengarah pada pertumbuhan yang cepat, yang tentunya memberikan booming ke pasar aplikasi Android. Konsumen berharap perangkat seluler menjadi cukup aman atau setidaknya cukup untuk memberi tahu konsumen tentang potensi serta bahaya apa pun dari aplikasi yang dikembangkan oleh pengembang aplikasi pihak ketiga. Masalah keamanan di smartphone menjadi lebih mirip dengan sistem desktop. Dalam beberapa tahun terakhir, sejumlah malware telah dirancang untuk menargetkan dan menyerang perangkat seluler

Kami melakukan analisis terhadap sistem keamanan Android untuk mengurangi malware dan Trojan yang membahayakan dan merugikan. Dalam desain, evaluasi, dan implementasi skema yang diusulkan, jurnal ini memberikan kontribusi sebagai berikut:

- Mengidentifikasi beberapa batasan dalam izin aplikasi Android dan kontrol aksesnya. Android tidak melakukan evaluasi pada sistem keamanan untuk mengurangi malware. Sumber daya aplikasi di Android hanya dapat dicabut dengan menghapus instalan aplikasi tersebut.
- Melakukan analisis terhadap perilaku aplikasi. Didasarkan pada basis pengetahuan yang disediakan oleh Android, yang membuatnya efisien dan kompatibel dengan sistem.

Android adalah sistem operasi berbasis Linux yang dirancang terutama untuk perangkat seluler layar sentuh seperti smartphone dan komputer tablet. Sistem operasi telah berkembang pesat dalam 15 tahun terakhir mulai dari ponsel hitam putih hingga smartphone atau komputer mini terbaru. Salah satu OS seluler yang paling banyak digunakan saat ini adalah android. Android adalah perangkat lunak yang didirikan di Palo Alto of California pada tahun 2003. Android mengandalkan Linux versi 2.6 untuk sistem inti layanan seperti keamanan, manajemen memori, proses manajemen, tumpukan jaringan dan model driver.[1]

Sistem Android menggunakan alat dx yang berfungsi mengubah kode java menjadi kode byte Dalvik agar dapat dimengerti. Semua aplikasi pada Android diidentifikasi dengan ID pengguna Linux (UID) yang unik, yang memungkinkan Dalvik menjalankan beberapa aplikasi, masing-masing dalam proses yang terpisah. layar visual adalah tempat pengguna berinteraksi dengan aplikasi. berupa daftar objek, seperti, label ke menu top up atau beberapa Gambar yang ditampilkan ke kotak dialog di layar ponsel. Aplikasi terdiri dari satu atau lebih kegiatan tergantung pada arsitektur dan desain yang digunakan. Layanan ini tidak memiliki antarmuka visual. Ini berjalan ke latar belakang, seperti memutar musik, mengambil data melalui jaringan[2]

Mekanisme kebijakan keamanan Android, suatu komponen yang dapat berinteraksi dengan komponen lain dalam aplikasi yang sama menggunakan mekanisme ICC khusus[3]. Intent adalah sebuah mekanisme penyampaian pesan yang berisi deskripsi tindakan yang harus dilakukan. Intents dapat dikirim ke komponen tertentu (disebut niat eksplisit) atau disiarkan ke kerangka kerja Android, yang kemudian meneruskannya ke komponen yang sesuai (disebut niat tersirat). String digunakan untuk menentukan maksud secara implisit; sebenarnya, action string menyajikan jenis tindakan yang harus dilakukan, dan kemudian sistem Android memutuskan komponen mana yang cocok untuk melakukan tindakan tersebut

Juice Jacking merupakan sebuah serangan yang menyerang telepon genggam anroid/ios dengan memanfaatkan Port USB sebagai perantara[4]. Serangan juice jacking sering dimanfaatkan pelaku kejahatan untuk mencuri data dari para korban yang tidak waspada. Serangan itu memanfaatkan celah yang diberikan, seperti pada tempat umum yang menyediakan layanan charging secara gratis, terkadang banyak korban yang tidak sadar akan bahaya tersebut. Virus yang ditanamkan pada divice dan memanfaatkan port pada kabel USB dapat masuk ke ponsel tampa diketahui oleh korban.

Kami menganalisa arsitektur OS Android dan mekanisme sistem keamanan yang mungkin memiliki pengaruh pada kepercayaan sumber aplikasi. Kami juga menyelidiki kerentanan potensial yang dapat timbul karena jumlah kemungkinan serangan dan bagaimana mereka dapat memengaruhi keamanan data pengguna. kami melakukan analisa untuk mengevaluasi keamanan data seperti: kerentanan akses root, kerentanan bootloader yang tidak dikunci, kunci perangkat, versi Android OS, versi patch keamanan, model perangkat, sumber aplikasi yang tidak diketahui, aplikasi yang diinstal, menu opsi pengembang, peringkat aplikasi yang diinstal, kerentanan sistem, peringkat perangkat.

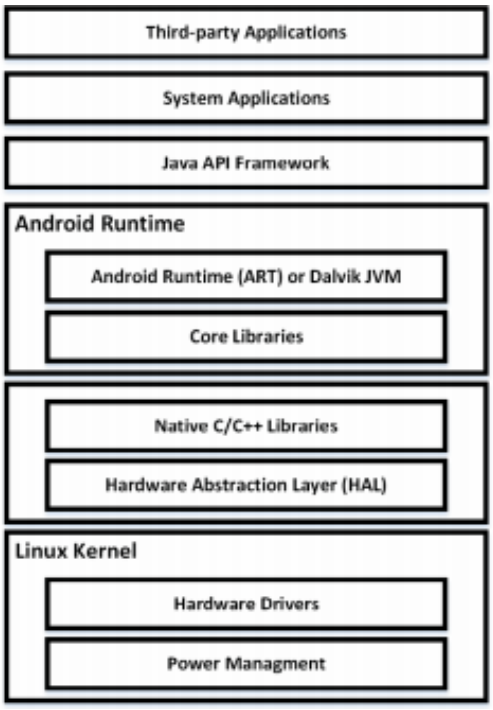
2. Metode Penelitian

Android tidak menyelidiki perilaku aplikasi untuk memastikan kepercayaannya dan mencegah aplikasi dari fungsi yang sebenarnya. Kami mengusulkan untuk menggunakan teknik Android Enhanced Security (SEAF) guna menganalisis perilaku dinamis suatu aplikasi. Teknik itu bertujuan untuk memberi tahu pengguna tentang aktivitas yang berbahaya dari suatu aplikasi, sehingga pengguna dapat membatasi akses aplikasi ke sumber daya yang rumit saat runtime. Selain itu, SEAF juga memberikan kontrol kepada pengguna untuk melakukan kustomisasi izin setelah instalasi dan memberikan kebijakan pada sebuah aplikasi.

Mekanisme izin untuk akses Android saat ini memberlakukan beberapa fungsi MAC primitif. Kami telah mengamati beberapa batasan dalam Android, terkait kebijakan keamanan dan kontrol akses.

1. Model izin Android tidak memiliki mekanisme untuk penyesuaian izin. Ini berarti bahwa pengguna harus menerima semua izin (disebutkan oleh aplikasi pada saat instalasi) agar instalasi aplikasi berhasil. Pengguna tidak memiliki pilihan untuk menolak izin tertentu yang diminta oleh suatu aplikasi.
2. Android tidak menyelidiki perilaku aplikasi untuk mencegah pada saat aplikasi mengalami kegagalan fungsi dan tidak memastikan integritas sistem. sebuah tindakan untuk menangani evaluasi perilaku dari aplikasi pengembang pihak ketiga untuk memastikan kepercayaan sangat dibutuhkan.

Arsitektur Android terdiri dari lapisan dasar yang merupakan kernel Linux yang berkomunikasi dengan platform perangkat keras dan sensor seperti pada Gambar 1. Hardware Abstraction Layer (HAL) menyediakan antarmuka standar komponen perangkat keras. Lapisan C / C ++ Native Library berisi pustaka berkinerja tinggi. Android Runtime (ART) dan pendahulunya Dalvik (untuk Android OS versi 5.0) mengeksekusi kode Java.



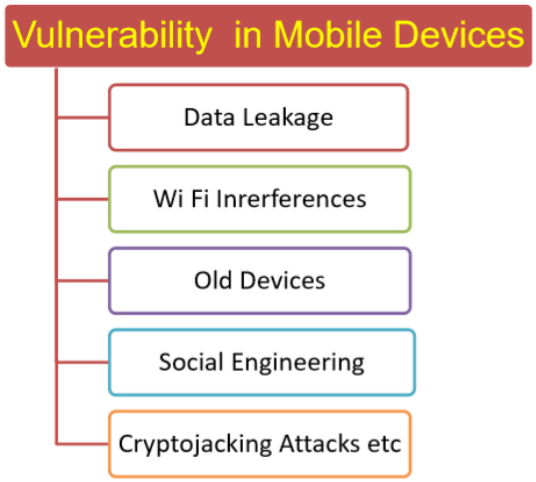
Gambar 1 Tipe Arsitektur Android

Android berasal dari kernel Linux. Mengimplementasikan Linux Discretionary Access Control (DAC)[5]. Setiap aplikasi diberi ID Unik (UID) dan memiliki sandbox sendiri. Tindakan semacam itu guna mencegah aplikasi untuk berinteraksi satu sama lain dan mewakili salah satu mekanisme perlindungan utama dalam sistem. Izin sistem itu menghadirkan perlindungan penting lainnya. Izin dapat menahan aplikasi dari akses tidak sah ke sumber daya sistem. Mereka dideklarasikan sebagai file Manifest, yang merupakan sebuah bagian yang tidak terpisahkan dari aplikasi OS Android. Izin sistem diklasifikasikan menjadi empat tingkatan: Normal, dangerous, signature and signature systems. Masa sebelum Android OS versi 6.0, pengguna wajib menerima semua izin yang dinyatakan selama proses instalasi. Dalam versi 6.0 Izin Runtime baru diperkenalkan. Sekarang pengguna dapat menginstal aplikasi dengan izin dengan notifikasi berbahaya namun juga dapat menginstal tanpa memberikan izin tersebut.

Ada beberapa jenis kerentanan yang dapat dicatat dalam perangkat smartphone. Saat ini smartphone sering menjadi target serangan penjahat melalui internet dan kami menyadari bahwa malware adalah program yang merusak sistem perhatikan Gambar 2. Penelitian mencatat

bahwa, varian malware telah meningkat hingga 54%[6]. Worms, virus dianggap penting dalam kategori ini. Serangan malware dapat dibagi menjadi dengan tiga jenis yaitu.

1. Infeksi sistem dan Izin aplikasi Eksplisit, Izin Tersirat, Interaksi Umum, Tanpa Interaksi. Semua kerentanan ini sangat berbahaya bagi perangkat seluler.
2. Mempengaruhi sistem setelah malware masuk pada perangkat lunak hingga masuk ke dalam sistem dan itu dilakukan oleh kerusakan moneter, data kerusakan dan / atau perangkat, serta kerusakan tersembunyi.
3. Penyebaran malware ke sistem lain adalah cara lain untuk membuat sistem rentan. Di sini Wi-Fi, Bluetooth dan inframerah atau bahkan menggunakan jaringan jarak jauh seperti, panggilan telepon atau penyebaran SMS atau email.



Gambar 2 Sekilas tentang Kerentanan Perangkat Seluler

Kesadaran pengguna sangat penting untuk menjalankan sistem berbasis Android yang aman dan terlindungi. Ini adalah fakta bahwa sebagian besar pengguna tidak berhati-hati ketika membaca pesan dan detail aplikasi. Di sini reputasi penyedia aplikasi, pesan keamanan, pesan perjanjian sangat penting. Bahkan mungkin ada beberapa perangkat lunak atau aplikasi dengan phishing yang intens, dll. Merawat telepon dengan pemilik juga merupakan langkah pengamanan penting dalam beberapa kasus. Berbagai organisasi dan asosiasi telah menyediakan kerangka kerja dan pedoman yang berbeda, yang semuanya harus diikuti oleh pemangku kepentingan yang berbeda (lihat Gambar: 2 untuk detail). Aplikasi dan sistem yang berbeda perlu ditutup jika tidak diperlukan, yaitu

1. Kamera
2. GPS
3. Bluetooth
4. USB
5. Penyimpanan eksternal, dll.

Secara keseluruhan, Keamanan Seluler membutuhkan langkah-langkah dan mekanisme pertahanan yang berbeda tetapi masih ada masalah tertentu yang sulit untuk melakukan keamanan. Dan di antara beberapa yang penting ini adalah Sistem Operasi. Perlu dicatat bahwa beberapa sistem operasi adalah penugasan tunggal sehingga tidak mampu melakukan tugas bersama dengan firewall atau antivirus. Perlu diperhatikan bahwa pemanfaatan jaringan seluler tidak boleh terlalu tinggi karena alasan keamanan. Selain dari penanggulangan teknologi, penting juga untuk memiliki minat dan kesadaran pengguna akan perhatian terkait keamanan.

Juice Jacking merupakan sebuah serangan yang menyerang telepon genggam android/ios dengan memanfaatkan Port USB sebagai perantara. Serangan juice jacking sering dimanfaatkan pelaku kejahatan untuk mencuri data dari para korban yang tidak waspada. Serangan itu memanfaatkan celah yang diberikan, seperti pada tempat umum yang menyediakan layanan charging secara gratis, terkadang banyak korban yang tidak sadar akan bahaya tersebut. Virus yang ditanamkan pada divice dan memanfaatkan port pada kabel USB dapat masuk ke ponsel tampa diketahui oleh korban.

Telepon genggam dengan OS iPhone, BlackBerry ataupun Android, tetap saja memiliki satu celah keamanan, yaitu Port daya dan aliran data melewati satu kabel yang sama. Dengan satu aliran yang sama antara port pengisian data dan transfer data maka akan muncul sebuah masalah. Ketika ponsel terhubung dengan perangkat lain, maka perangkat tersebut dapat berpotensi mengambil data yang ada dalam ponsel kita. Data/power pada kabel yang sama, menjadi jalan bagi peretas untuk masuk dan mendapatkan akses ke ponsel selama proses pengisian, memanfaatkan USB kabel data/power untuk mengakses data telepon dan/atau menyuntikkan kode berbahaya ke perangkat, teknik ini dikenal sebagai Juice Jacking. Pencurian data dengan metode ini telah menjadi ancaman terbesar bagi pengguna gadget di mana pun Anda berada setiap kali mengisi ulang baterai.

Dengan melihat kerentanan yang ada, maka kami akan membuat analisis kerentanan dengan metode JuiceJacking, dengan memanfaatkan hak akses yang dibatasi, kami membuat sebuah virus dimana virus itu ditanamkan pada Port USB untuk masuk ke dalam sistem dan mendapatkan Akses sebagai Administrator sehingga saat virus tersebut bekerja, ia akan menjadi root yang tidak terdeteksi oleh AntiVirus. Virus yang kami masukkan kedalam USB adalah Remote Access Trojan, dimana kami memanfaatkan celah yang ada pada Remote Code Execution. Dengan menyelinap dan masuk menjadi akses root, maka kami akan mendapatkan hak akses sepenuhnya pada device korban. Dibawah ini merupakan bentuk dari layer pertama alat yang dbuat, berfungsi untuk menyelipkan kebel dan pemancaar sinyal yang digunakan untuk menghubungkan dengan server[7].

3. Hasil Penelitian dan Pembahasan

Catatan Rilis Keamanan Android berikut ini berisi rincian kerentanan keamanan yang mempengahuri perangkan android yang ditunjukkan sebagai bagaian dari Android 10. Perangkat Android 10 dengan tingkat patch keamanan 2019-09-01 atau lebih aman dari masalah ini seperti pada Gambar 3 (Android 10, seperti dirilis pada AOSP, memiliki tingkat update keamanan default 2019-09-01)

Sistem

Kerentanan paling parah di bagian ini dapat memungkinkan penyerang jarak jauh menggunakan transmisi yang dibuat khusus untuk mengeksekusi kode arbitrer dalam konteks proses istimewa.

CVE	Referensi	Tipe	Kerasnya	Versi AOSP yang diperbarui
CVE-2020-0117	A-151155194	RCE	Kritis	8.0, 8.1, 9, 10
CVE-2020-8597	A-151153886	RCE	Kritis	8.0, 8.1, 9, 10
CVE-2020-0116	A-151330809	Indo	Tinggi	10
CVE-2020-0119	A-150500247 [2]	Indo	Tinggi	10

Gambar 3 AOSP patch 2019-09-01

.

3.1. Perancangan Sistem

Pemanfaatan tempat pengisian untuk tindakan kejahatan terhadap perangkat, dengan alasan memberikannya dengan cuma-cuma atau gratis. Ketika anda mengisi batrai ponsel melalui port USB komputer atau device anda, malware ini juga membuka opsi untuk memindahkan file antara kedua sistem. Itu karena port USB bukan cuma sekedar pengisian daya. Konektor USB biasa memiliki lima pin, dimana hanya satu yang diperlukan untuk mengisi daya ujung penerima. Dua lainnya digunakan secara default untuk transfer data.

Sebenarnya banyak tolls bahkan device yang mendukung untuk melakukan teknik juice jacking, seperti halnya HTC Dream spesifik (juga dikenal sebagai Android G1)[8], tersedia koneksi Serial over USB; pengkabelan direkayasa terbalik (kabel serial Android G1 ke USB) memungkinkan beberapa kemampuan (mis., pengintaian USB) untuk menguping data yang disampaikan melalui kabel. Namun, pendekatan ini masih belum lengkap dan membutuhkan lebih banyak pengujian; beberapa model perangkat lain (mis., HTC Magic, Samsung i7500 Galaxy) tidak dijamin kompatibel dan rekonstruksi protokol yang digunakan masih dalam proses.

Sebagian besar penyerang melakukan serang untuk mengincar data yang menurut mereka penting, seperti data pada File Manager, Pesan, Galeri, Contact, dan Note, seperti Gambar 4.

Gambar 4 target pencurian

XML yang merupakan nomenklatur Extensible benar-benar didefinisikan karena kumpulan aturan untuk mengode beberapa catatan dalam format yang dapat dibaca manusia dan dapat pula dibaca oleh mesin[9]. XML ditentukan sebagian besar sebagai spesifikasi XML 1.0 World Wide Web Consortium yang dikenalkan pada 1998. Meskipun banyak spesifikasi standar yang tersedia di luar sana untuk nomenklatur yang dapat dikembangkan.

Tujuan dari memperkenalkan nomenklatur XML adalah untuk menekankan kesederhanaan, generalitas, dan kemudahan di internet. ini adalah sebuah format teks dengan dukungan stabil melalui Unicode untuk berbagai jenis bahasa di internet. Meskipun yang menjadi fokus dari perencanaan XML adalah dokumen tetapi yang paling utama pengembang menggunakan bahasa ini untuk representasi struktur data yang berubah-ubah sebagai contoh yang digunakan dalam layanan web.

Disini kami akan merancang sebuah alat yang digunakan untuk melakukan analisa terhdap kerentanan android pada port usb yang nantinya akan menajadi sebuah pembelajaran bagi pengguna ponsel pintar di masa depan. Dengan melihat kerentanan yang sudah dijelaskan dan memanfaatkan kemajuan teknologi yang mulai berkembang dan kurangnya kesadaran, diharapkan dengan adanya tulisan ini, mampu membuat pengguna smartphone sadar akan bahayanya memberi kepercayaan kepada sebuah platform yang diberikan secara gratis

Dengan menggunakan sebuah software yang sudah kami buat,script aplikasi pada Gambar 5. Kami mencoba melakukan sebuah analisa dengan memanfaatkan kelemahan pengguna dalam memberikan kepercayaan kepada smartphone mereka. Banyak pengguna smartphone masih lalai dalam menjaga keamanan data pribadi

```
#!/bin/bash
base_count=`/sbin/lsusb | wc –l`;
last_count=$base_count;
interval=2;
while ( sleep $interval; ) do
  count = `/sbin/lsusb | wc –l`;
  if [ $last_count != $count ] && [ $count != $base_count ]
    then
  # Do your code here
  fi
done
```

Gambar 5 Script Aplikasi

3.2. Hasil Analisa Aplikasi

Memanfaatkan tempat pengisian untuk tindak kejahatan terhadap perangkat device, dengan alasan memberikannya dengan gratis kami mengidentifikasi kerentanan pengisian daya ponsel cerdas dan merancang jenis serangan pengisian daya baru (disebut serangan jus jacking) berdasarkan konektor USB standar dan HDMI, yang dapat mencuri rahasia pengguna melalui pengambilan video secara otomatis input mereka (misalnya, pola buka kunci, PIN kode) [10]. Dengan membangun sebuah desain box charging yang kami buat seperti pada Gambar 6.

Gambar 6 Design Box Charging

Ketika anda mengisi batrai ponsel melalui port USB device anda, malware ini juga membuka opsi untuk memindahkan file antara kedua sistem. Ini dikarenakan port USB bukan Cuma sekedar pengisian daya. Konektor USB biasa memiliki 5 pin, pada Gambar 7, dimana hanya satu yang diperlukan untuk mengisi daya, Dua lainnya digunakan secara default untuk transfering data.

Pin	Name	Cable color	Description
1	V_{BUS}	Red	+5 V
2	D−	White	Data −
3	D+	Green	Data +
4	ID	N/A	Permits distinction of a host connection from device connection: • host: connected to the signal ground • device: not connected
5	GND	Black	Signal ground

Gambar 7 Deskripsi Konektor USB

Virus yang kami gunakanakan menyerang kedalam sistem remote dan mengambil alih device sebagai administrator seperti pada Gambar 8. Dan kita dapat mengoperasikan melalui server yang ada pada laptop kita.

Gambar 8 Laptop Server Remote.

4. Kesimpulan

Perangkat berbasis Android menjadikannya sebuah platform yang sangat nyaman untuk digunakan. Namun, karena banyak kerentanan pada sistem Android, beberapa skenario manipulasi data dimungkinkan. Makalah ini menyelidiki arsitektur OS Android dan mekanisme keamanan utama yang mungkin memiliki pengaruh pada kepercayaan data yang berasal dari keteledoran pengguna. Selain itu, kami menyelidiki kerentanan potensial yang dapat timbul karena serangan malware dan bagaimana mereka dapat mempengaruhi keamanan data pengguna. percobaan dilakukan untuk menganalisa keamanan data, seperti: kerentanan akses root, kerentanan bootloader yang tidak dikunci, kunci perangkat, versi Android OS, versi patch keamanan, model perangkat, sumber aplikasi yang tidak diketahui, aplikasi yang diinstal, menu opsi pengembang, peringkat aplikasi yang diinstal, kerentanan sistem, peringkat perangkat.

Dengan melihat hasil diatas dapat disimpulkan bahwa kerentanan yang ada pada android terletak pada sistem RCE dan kerentanan pada Port USB yang masih rentan terkena serangan. Dengan menggunakan teknik juice jacking , penyerang dapat masuk ke sistem korban dan mengambil alih device korban. Hal yang dapat dilakukan untuk mengantisipasi hal tersebut terjadi dengan memberikan device anda anntivirus dan selalu mengupdate ke versi terbaru. Dan matikan izin untuk aplikasi yang tidak dikenali, serta selalu mengunduh data pada sumber yang terpercaya.

Refrensi

[1] A. Cochereau, “What is Android,” *Soins. Pediatr. Pueric.*, no. 257, p. 8, 2008.
[2] H. Banuri *et al.*, “An Android runtime security policy enforcement framework,” *Pers. Ubiquitous Comput.*, vol. 16, no. 6, pp. 631–641, 2012.
[3] M. Nauman, S. Khan, and X. Zhang, “Apex: Extending Android permission model and enforcement with user-defined runtime constraints,” *Proc. 5th Int. Symp. Information, Comput. Commun. Secur. ASIACCS 2010*, pp. 328–332, 2010.
[4] Y. Kumar, “Juice Jacking - The USB Charger Scam,” *SSRN Electron. J.*, no. April, 2020.
[5] I. Khokhlov and L. Reznik, “Data security evaluation for mobile android devices,” *Conf. Open Innov. Assoc. Fruct*, vol. 2017-April, pp. 154–160, 2017.
[6] P. Paul and P. S. Aithal, “Database Security: An Overview and Analysis of Current Trend,” *SSRN Electron. J.*, pp. 112–121, 2019.
[7] R. Rowley, “Juice jacking unearthed,” no. August, 2013.
[8] A. Carroll, “An Analysis of Power Consumption in a Smartphone.”
[9] S. Khatun, S. Sarkar, and M. Biswas, “SecureIT – A Weapon to Protect You,” *SSRN Electron. J.*, no. January, 2020.
[10] W. Meng, W. H. Lee, S. R. Murali, and S. P. T. Krishnan, “Charging me and I know your secrets! Towards juice filming attacks on smartphones,” *CPSS 2015 - Proc. 1st ACM Work. Cyber-Physical Syst. Secur. Part ASIACCS 2015*, pp. 89–98, 2015.