Available *online* at : http://bit.ly/InfoTekJar

InfoTekJar : Jurnal Nasional Informatika dan Teknologi Jaringan

ISSN (Print) 2540-7597 | ISSN (Online) 2540-7600

Pengaruh File Apk Terhadap Keamanan Sistem Operasi Android Berdasarkan Analisis Statik dan Dinamik

Tri Agus Riyadi

Universitas Gunadarma, Jl. Margonda Raya No. 100, Depok, 16424, Indonesia

KEYWORDS

Kerentanan, Statik, Dinamik, Android, Apk

CORRESPONDENCE

Phone: +6287888936527

E-mail: ta_riyadi@staff.gunadarma.ac.id

A B S T R A C T

Sistem operasi Android saat ini banyak digunakan pada perangkat elektronik seperti ponsel pintar, tablet dan sebagainya. Penggunaan Android pada perangkat elektronik seperti ponsel pintar dapat memudahkan pengguna untuk mengaktifkan atau menonaktifkan perangkat lainnya dengan teknologi *Internet of Things* (IoT). Untuk dapat menggunakan fasilitas tersebut, pengguna biasanya diwajibkan untuk menginstal suatu file dalam Androidnya. File yang di*instal* sebenarnya bisa saja menjadi berbahaya bagi Android itu sendiri. File ini biasanya dinamakan dengan file apk. Tujuan dari penelitian ini adalah untuk mengetahui seberapa besar pengaruh dari suatu file apk terhadap sistem operasi Android dengan menggunakan analisis statik dan dinamik. Metode pertama yang digunakan adalah analisis statik dimana file apk diuji dengan suatu *tools* yang bernama Qark. Qark melakukan analisis terhadap file apk yang telah dibuat dan memberikan rekomendasi dari pengujian tersebut. Rekomendasi yang dihasilkan ada tiga yaitu rentan, peringatan dan informasi. Metode kedua adalah analisis dinamik. Pada analisis dinamik, file apk langsung diinstal pada Android dan diberikan *permission*nya. Tanpa pemilik perangkat Android sadari, orang lain sudah berhasil masuk ke dalam hampir seluruh sumber daya yang ada pada Android salah satunya adalah dapat mengakses sms.

Many devices using Android such as handphone, tablet and others. Android makes the daily works become so easy to activate other devices using IoT technology. To use those facilitate, Android user should install a file such as an apk file. The objective of this research is to examine the impact of an apk file in Android using static and dynamic analysis. First, static analysis was done using Qark and the results are three recommendation such as vulnerable, warning and information. Second, dynamic analysis was done by giving the permission to the apk file when it was installed in Android. The impact of this condition was anyone could access almost all the resources in the Android such as SMS when connected to internet.

PENDAHULUAN

Inovasi teknologi pada peralatan pintar saat ini sedang berkembang dengan pesat. Peralatan pintar ini banyak digunakan oleh masyarakat seperti ponsel pintar, peralatan rumah tangga pintar, jam tangan pintar, peralatan yang terhubung dengan *Internet of Things* (IoT) dan sebagainya. Seluruh peralatan ini dapat membuat manusia dimudahkan dalam pemantauan dan pengaksesan sumber daya yang diinginkannya. Setiap peralatan ini pasti memiliki sistem operasi yang mampu mengatur sumber daya didalamnya. Salah satu sistem operasi yang banyak digunakan adalah sistem operasi Android. Sistem operasi Android digunakan pada ponsel pintar, tablet dan peralatan lainnya. Sistem operasi Android saat ini juga mendukung untuk peralatan rumah tangga berbasis IoT. Peralatan rumah tangga yang mendukung IoT dapat terhubung dengan ponsel pintar sehingga bisa dipantau dan dikendalikan dari jarak jauh[1].

Sistem operasi Android yang digunakan pada ponsel pintar, tablet dan peralatan lainnya dikembangkan oleh Google. Arsitektur Android menggunakan kernel Linux. Berbagai varian dari kernel Linux digunakan sehingga memiliki perubahan arsitektur Android lebih lanjut yang diterapkan oleh Google. Penerapan ini diluar siklus pengembangan dari kernel Linux biasa[2]. Tujuan dibuatnya Android adalah penggunaan sistem layar sentuh pada ponsel pintar dan tablet. Android memiliki karakteristik perangkat lunak yang *free* dan *open source*[3]. Android pertama kali dikeluarkan dengan versi 1.0 yang dinamakan dengan Alpha pada tahun 2008[4]. Hampir setiap tahun dikeluarkan versi terbaru dari Android. Versi terbaru Android adalah versi 11.

Sistem operasi Android sendiri menawarkan berbagai fasilitas kemudahan antar muka yaitu dengan menggunakan sentuhan jari

https://doi.org/10.30743/infotekjar.v6i2.4677

pengguna dapat melakukan hal yang diinginkannya yang telah tersedia dalam Android. Sebagai contoh, adanya virtual keyboard pada sistem operasi Android memudahkan pengguna melakukan pengetikan pada suatu aplikasi. Contoh lainnya, jika pengguna ingin menggunakan kamera, hanya menekan icon kamera dan kamera bisa digunakan. Penggunaan sistem operasi Android juga memudahkan penggunanya dalam hal pencetakan dokumen ke printer. Android memiliki fasilitas Wi-Fi dan Bluetooth yang dapat digunakan untuk mengirimkan dokumen yang ingin dicetak ke mesin pencetak atau printer yang juga memiliki fitur Wi-Fi atau Bluetooth. Pemanfaatan fasilitas tersebut membuat pengguna tidak perlu menghubungkan ponsel pintar ke komputer untuk mencetak dokumen melalui komputer. Selain beberapa kemudahan tersebut, masih banyak lagi kemudahan yang didapatkan dengan menggunakan sistem operasi Android pada ponsel pintar[5].

Kemudahaan yang diberikan oleh sistem operasi Android terhadap penggunanya tentu saja akan memberikan pengaruh terhadap keamanan dan privasi dari pengguna ponsel pintar. Berbagai aplikasi yang dapat digunakan pada sistem operasi Android telah tersedia secara legal dalam Android ataupun melalui pihak ketiga. Aplikasi ini digunakan oleh pengguna karena memudahkan pengguna untuk melakukan aktivitas yang berhubungan dengan aplikasi tersebut. Tanpa disadari, aplikasi yang digunakan pengguna bisa saja menjadi ancaman bagi pengguna ponsel pintar. Ancaman yang mungkin terjadi seperti pencurian data penting pengguna, pengaksesan lokasi pengguna, pengaksesan kamera pengguna dan aktivitas lainnya yang berkaitan dengan internet. Selain itu, ancaman serangan ransomware juga menghantui pengguna ponsel pintar dengan cara mengunci ponsel pengguna atau mengenkripsi data pengguna atau melakukan pengaksesan ponsel dari jarak jauh. Keadaan ini dapat membuat pengguna kehilangan data atau hal yang lebih buruh adalah tidak dapat mengakses ponsel pintarnya sehingga Android harus direset ulang[6].

Keamanan merupakan hal yang sangat penting pada sistem operasi Android. Android tidak memberikan ijin kepada aplikasi eksternal untuk melakukan perubahan atau modifikasi terhadap file yang telah terinstal. Namun Android memberikan ijin terhadap aplikasi tertentu untuk menggunakan perangkat keras dan perangkat lunak dari perangkat yang menggunakan sistem operasi Android. Pengguna Android juga memiliki hak untuk menerima atau menolak ijin dari aplikasi yang terinstal. Contohnya jika ada suatu aplikasi yang terinstal meminta ijin untuk mengakses contact pada Android, biasanya aplikasi tersebut akan meminta ijin dan pengguna dapat memberikan ijin atau menolak ijin tersebut. Namun, pada Android memungkinkan penggunaan IPC (*Inter-Process Communication*) dimana dapat membuat perangkat Android menjadi rentan terhadap ancaman keamanan yang dapat menyebabkan pembajakan[7].

Ancaman keamanan yang saat ini tengah dihadapi oleh ponsel pintar yang menggunakan Android terbagi dalam tiga yaitu berasal dari *website*, aplikasi yang berisi *malware*, resiko hilangnya data dan keamanan jaringan[8]. Website banyak digunakan oleh pengguna Android untuk mencari informasi melalui internet. Beberapa *website* menjalankan kode tertentu ketika ada pemicu yang diberikan. Jika pengguna menyentuh pemicu tersebut maka kode tersebut akan mengaktifkan apa yang diperintahkan didalam kode tersebut. Keadaan ini tentu saja menjadi bahaya jika kode tersebut merupakan kode yang membahayakan perangkat Android. Untuk menghindari kejadian ini, dapat menonaktifkan *Java Script* pada *browser* dan menggunakan *browser* yang aman dengan fitur AdBlocking didalamnya[9]. Aplikasi yang berisi *malware* telah banyak ditemukan dalam pencarian aplikasi legal Android yaitu *Google Play Store*. Namun, Google telah menghapus banyak aplikasi yang berisi malware dan menyebabkan aplikasi pihak ketiga menjadi sumber utama dari kode dan aplikasi yang berbahaya bagi perangkat Android. Aplikasi ini mampu melakukan apapun seperti mengakses *contact*, pesan, file pribadi, gambar dan sebagainya. Cara yang paling mudah untuk menghindari keadaan ini adalah hanya menginstal aplikasi dari sumber yang terpercaya. Namun jika dengan terpaksa harus menginstal aplikasi dari pihak ketika, disarankan untuk menggunakan *website* yang terpercaya seperti apkmirror.com[10].

Hilangnya ponsel pintar akan menyebabkan hilangnya data yang disimpan dalam ponsel tersebut. Pemulihan data untuk kasus seperti ini sangat tidak mungkin dilakukan karena ponsel pintar yang hilang tidak dapat diketemukan kembali. Untuk mengatasi hal ini, sangat disarankan untuk menggunakan penyimpanan cadangan data secara online sehingga jika ponsel pintar tersebut hilang, data masih dapat dipulihkan. Pengguna ponsel pintar sering mencari koneksi internet di tempat umum dengan memanfaatkan Wi-Fi pada tempat tersebut. Biasanya pengguna diwajibkan memasukkan *username* dan *password* untuk dapat terhubung dengan internet melalui Wi-Fi. Jika Wi-Fi tersebut dibuat oleh orang yang tidak bertanggung jawab, kemudian pengguna masuk ke dalam jaringan tersebut, maka jika pengguna tersebut melakukan aktivitas perbankan sudah sangat bisa dipastikan seluruh data perbankan dapat dicuri oleh orang tersebut[8].

Kerentanan sistem operasi Android yang berasal dari aplikasi Android merupakan suatu tantangan tersendiri karena informasi mengenai aplikasi tersebut masih kurang tersedia pada penyedia aplikasi Android (market application). Banyak aplikasi turunan dari suatu aplikasi yang membentuk aplikasi baru. Perekonstruksian yang dilakukan oleh Gao et al.[11] dari 465.037 aplikasi menghasilkan aplikasi turunan sebanyak 28.564. Dengan melakukan investigasi terhadap kerentanan Android dari aplikasi turunannya, dapat diketahui bahwa kerentanan dapat bertahan hingga paling sedikit tiga update, perpustakaan aplikasi dari pihak ketiga memberikan kontribusi yang cukup besar dalam kerentanan Android, melakukan pengenalan kembali kerentanan berlaku untuk semua kerentanan dan kerentanan menandakan adanya *malware*.

Model metadata diperkenalkan oleh Joshi dan Parekh[12] untuk memahami semua yang berhubungan dengan kerentanan Android seperti tanggal ditemukannya kerentanan, jalan keluarnya dan tingkat keparahan dari suatu kerentanan. Data kerentanan yang digunakan didapatkan dari berbagai sumber data dimana memberikan informasi yang lebih detail mengenai karakteristik dari kerentanan. Selain itu juga berpengaruh pada aspek keamanan seperti ijin, hak istimewa dan pengendalian akses. Berdasarkan data-data tersebut, dibuatlah suatu aplikasi pemindaian kerentanan Android yang dapat mengidentifikasi kerentanan dari versi terbaru dan dapat memperbaiki kerentanan

 https://doi.org/10.30743/infotekjar.v6i2.4677

yang ada sehingga meningkatkan keamanan dari aplikasi bergerak.

Analisis pendeteksian terhadap kerentanan suatu perangkat lunak yang ditanamkan pada Android dapat menggunakan dua metode yaitu Analisis Statik dan Analisis Dinamik. Analisis statik dapat digunakan untuk mendeteksi dan memperbaiki jika terjadi ancaman terhadap Android. Pada Analisis statik, program aplikasi atau program aslinya dianalisis untuk mengetahui apakah ada aktivitas yang berbahaya tanpa harus mengeksekusi program aplikasi tersebut. Analisis statik dijalankan dengan mengekstraksi informasi dari paket apk viz. Program aplikasi harus diekstraksi dari apk. Berbagai tools yang bisa digunakan untuk melakukan analisis terhadap program aplikasi seperti DroidMat, dex2jar dan masih banyak lagi. Selain analisis statik, analisis dinamik juga bisa digunakan untuk mengetahui apakah program aplikasi yang dimasukkan dalam Android berbahaya atau tidak. Analisis dinamik melakukan pengujian dan evaluasi dengan memberikan data ke aplikasi dalam keadaan *real time* dengan kondisi lingkungan yang terpantau dengan baik. Analisis dinamik digunakan untuk menganalisis dan memantau situasi dibalik aplikasi yang sedang berjalan. Analisis dinamik mampu bekerja terhadap file yang dienkripsi, *polymorphic* dan *malware* yang mengubah kodenya. Beberapa aplikasi yang dapat digunakan adalah ARTDroid, VetDroid dan sebagainya[13].

Berbagai penelitian dilakukan untuk dapat mencari kerentanan Android sehingga suatu tools dapat dibuat untuk mencegah adanya serangan dari celah pada Android. Tujuan penelitian ini adalah menganalisis pengaruh dari suatu file apk dengan menggunakan analisis statik dan dinamik.

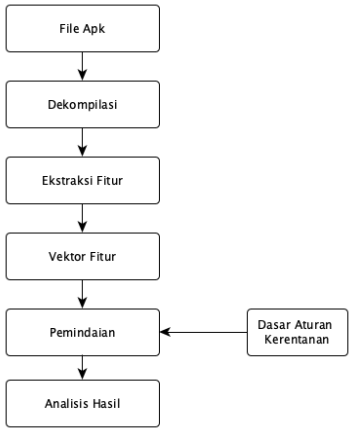
METODOLOGI

Tahapan dari penelitian ini dapat dilihat pada gambar 1.

Gambar 1. Tahapan Penelitian

Tahap pertama yang dilakukan adalah membuat satu file apk dengan menggunakan perangkat lunak yang ada. Fitur yang dimasukkan dalam file tersebut diantaranya *view contacts, File explorer* dan *Downloader, Call logs* dan *Send SMS*. Tahap selanjutnya dilakukan pengujian terhadap file apk yang telah dibuat. Pengujian terhadap file apk tersebut dilakukan dengan menggunakan analisis statik dengan menggunakan *software* Qark dan analisis dinamik. Pengujian dengan menggunakan analisis statik menguji file apk yang telah dibuat dengan perangkat lunak Qark. Tahap akhir dilakukan analisis dari hasil yang diperoleh berdasarkan analisis statik dan dinamik.

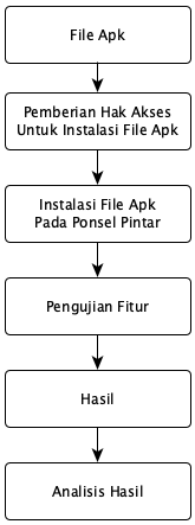
Pengujian pertama dengan menggunakan analisis statik. Proses pengujian dengan analisis statik dapat dilihat pada gambar 2.



Gambar 2. Proses Analisis Statik[14]

File Apk atau file aplikasi Android yang telah dibuat selanjutnya didekompilasi dan dipreprosesing. Dekompilasi dilakukan untuk mendapatkan kode program dari aplikasi Android seperti Java source code dan AndroidManifest.XML. Selanjutnya dilakukan fitur ekstraksi dari kode program aplikasi Android sehingga menghasilkan vektor fitur seperti API *function*, *Privilege*, *Component* dan File *Library*. Tahap selanjutnya adalah dilakukan pemindaian berdasarkan dasar aturan kerentanan yang telah dibuat oleh *tools* yang digunakan berdasarkan analisis aliran data, pencocokan ekspresi regular dan deteksi file sehingga dihasilkan suatu laporan yang dapat dianalisis. *Tools* yang digunakan untuk analisis statik adalah Qark.

Analisis dinamik pada penelitian ini dilakukan setelah mengetahui hasil dari analisis statik. Analisis dinamik biasanya dilakukan untuk memverifikasi hasil yang didapatkan pada analisis statik. Proses analisis dinamik pada penelitian ini diperlihatkan pada gambar 3.



Gambar 3. Proses Analisis Dinamik

File apk yang telah dibuat diberikan hak akses supaya dapat diinstal pada ponsel pintar Android. Sistem Operasi Android yang digunakan pada penelitian ini adalah Android 8. Setelah hak akses diberikan terhadap file tersebut, maka file aplikasi tersebut dapat diinstal pada Android 8. Pengujian dilakukan terhadap fitur yang telah ditanamkan pada file apk. Selanjutnya dilakukan analisis terhadap hasil dari pengujian.

HASIL DAN PEMBAHASAN

Berdasarkan hasil pengujian terhadap file apk yang telah dibuat dengan menggunakan analisis statik, didapatkan beberapa kondisi yang membuat file apk tersebut berbahaya jika diinstal ke Android. Ada tiga rekomendasi yang dihasilkan dari pengujian ini yaitu rentan, peringatan dan informasi. Rekomendasi yang dianggap sebagai kerentanan adalah *Manifest is manually set to debug*. Rekomendasi yang dianggap sebagai peringatan jika file apk tersebut diinstal ke Android adalah *Broadcast sent without receiver permission, Insecure functions found, Logging found dan Potential Vulnerable check permission function called*. Rekomendasi yang dianggap sebagai informasi adalah *Exported tag with permission* dan *Hardcoded HTTP url found*.

Tabel 1. Hasil Pengujian Analisis Statik dengan Qark

Masalah	Hasil Rekomendasi	File yang terdeteksi
Manifest is manually set to debug	vulnerability	AndroidManifest.xml
Broadcast Sent without receiver Permission	warning	MainService.java
Insecure functions found	warning	ConnectionManager.java Polling.java Emitter.java Polling.java Parser.java Emitter.java Ack.java WebSocket.java
Logging found	warning	WifiScanner.java FileManager.java FileManager.java MicManager.java MicManager.java
Potential Vulnerable check permission function called	warning	LocManager.java PermissionManager.java
Exported tag with permission	Information	AndroidManifest.xml
Hardcoded HTTP url found	Information	IOSocket.java

Permasalahan pada *manifest is manually set to debug* menandakan bahwa the Android:debuggable flag yang berada pada file AndroidManifest.xml diset benar secara manual. Hal ini dapat menyebabkan aplikasi android dapat didebug ketika dalam pengembangan dan dapat mengakibatkan kebocoran data dan isu keamanan lainnya. Rekomendasi dari permasalahan ini adalah rentan jika file Apk ini tetap diinstal pada Android. Permasalahan selanjutnya adalah pada *broadcast sent without receiver permission* menandakan bahwa aplikasi yang ada pada perangkat dapat menerima broadcast pesan tanpa sepengetahuan pemilik perangkat android. Ini bisa menyebabkan kebocoran data. Rekomendasi terhadap permasalahan ini adalah peringatan karena jika dilakukan maka kebocoran data akan sangat mungkin terjadi.

Permasalahan berikutnya adalah *Insecure function found* dimana *framework* tidak memiliki pemeriksaan *permission*. Ini dapat mengakibatkan komponen yang tidak terotorisasi dapat berinteraksi dengan penyedia konten. Rekomendasi dari permasalahan ini adalah peringatan. Permasalahan selanjutnya adalah *logging found*. *Log* seharusnya tidak boleh terkompilasi dalam suatu aplikasi kecuali pada saat pengembangan. Hal ini bisa mengakibatkan kebocoran informasi yang disebabkan dari aplikasi Android tersebut. Rekomendasi dari permasalahan ini merupakan peringatan. Permasalahan selanjutnya adalah *Potential Vulnerable check permission function called*. Aplikasi berbahaya tanpa *permission* yang tepat sehingga dapat mengabaikan *permission* tersebut dengan menggunakan *applicationpermission* untuk mendapatkan akses, selain itu sumber daya akan menolaknya. Rekomendasi yang diberikan dari *tools* ini adalah peringatan.

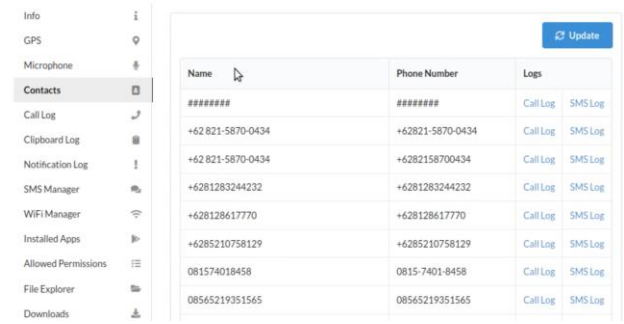
Hasil rekomendasi terhadap permasalahan selanjutnya adalah informasi. Permasalahan pertama adalah *exported tag with permission*. Kegagalan dalam memproteksi *service tag*s akan mengakibatkan kerentanan sehingga dapat diserang oleh aplikasi berbahaya. Permasalahan terakhir adalah *hardcoded HTTP url found* dimana permintaan ini dapat dicegat dan dimodifikasi oleh *man-in-the-middle-attack*.

Pengujian kedua adalah dengan menggunakan analisis dinamik dimana file apk yang telah dibuat langsung diinstall pada perangkat yang menggunakan sistem operasi Android dan diujicobakan. Pada pengujian di penelitian ini, Android 8 digunakan sebagai sistem operasi Android yang diuji untuk file apk tersebut. File apk yang diinstal ke perangkat Android, harus diberikan permissionnya supaya file tersebut dapat terinstal. Setelah file apk tersebut terinstal pada ponsel pintar Android dan terhubung dengan internet maka sebuah komputer yang juga terhubung dengan internet dapat mendeteksi ponsel pintar Android tersebut dengan menggunakan suatu aplikasi. Aplikasi pada komputer tersebut dapat langsung memberikan tampilan ponsel pintar Android yang terdeteksi dan pengguna komputer dapat mengaksesnya. Hal ini mengakibatkan hampir seluruh sumber daya yang ada pada perangkat Android dapat digunakan pada komputer tanpa sepengetahuan pemilik ponsel pintar Android. Fitur yang terdapat pada file apk seperti *view contacts, File explorer* dan *Downloader*, *Call logs, Send SMS* dan masih banyak lainnya dimana seluruhnya dapat diakses secara mudah dari komputer seperti yang terlihat pada gambar 4.

 https://doi.org/10.30743/infotekjar.v6i2.4677

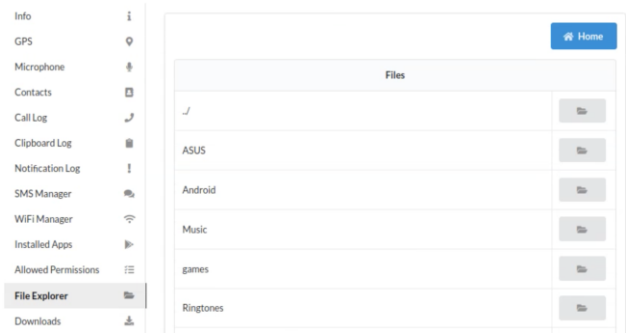
Gambar 4. Tampilan Pada Komputer yang Terhubung dengan Ponsel Pintar Android

Pengguna komputer juga dapat mengetahui siapa saja yang ada pada *contacts* di ponsel pintar Android tersebut dan dapat juga mengetahui nomor telepon yang ada di *contact* tersebut seperti yang terlihat pada gambar 5.



Gambar 5. Pengaksesan *Contact* yang Ada di Ponsel Pintar Android

Selain itu, pengguna komputer dapat mengakses *file explorer* sehingga seluruh file yang tersimpan dalam ponsel pintar Android dapat dilihat dan di*download* oleh pengguna komputer yang mengaksesnya seperti yang terlihat pada gambar 6. Pengguna komputer dengan mudah dapat memasuki setiap *folder* yang ada pada *file explorer* tanpa diketahui oleh pemilik ponsel pintar Android dengan mengklik icon yang terdapat pada sisi paling kanan.



Gambar 6. Pengaksesan File Explorer yang Ada di Ponsel Pintar Android

Berdasarkan hasil pengujian analisis dinamik memperlihatkan bahwa file apk yang telah dibuat dan diinstal dalam ponsel pintar Android memberikan kerentanan terhadap ponsel tersebut karena hampir seluruh sumber daya dalam ponsel pintar Android dapat digunakan secara *remote* oleh orang lain yang terhubung internet dan menggunakan suatu aplikasi. Pada hasil pengujian analisis statik terhadap file apk sudah diberikan rekomendasi berupa rentan, peringatan dan informasi dimana kesemuanya terbukti ketika analisis dinamis diujikan terhadap file apk tersebut.

KESIMPULAN

File apk merupakan file yang tidak resmi dibuat dan dapat diinstal pada ponsel pintar Android. File tersebut bisa menjadi berbahaya bagi pengguna ponsel pintar Android jika *permission* untuk diinstal pada ponsel pintar Android diberikan terhadap file apk tersebut. Berdasarkan pengujian dengan menggunakan analisis statik dan dinamik, dihasilkan bahwa file apk yang telah dibuat dapat memberikan kerentanan terhadap ponsel pintar Android. Pada analisis statik, sudah diberikan rekomendasi terhadap file tersebut yaitu rentan, peringatan dan informasi. Pada analisis dinamik, terlihat bahwa hampir seluruh sumber daya yang ada pada ponsel pintar dapat diakses dari jarak jauh oleh orang lain tanpa diketahui oleh pemilik ponsel pintar Android tersebut.

DAFTAR PUSTAKA

[1] Umasankar, "Analysis of latest vulnerabilities in android," in *2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2017, pp. 1236–1241, doi: 10.1109/ICACCI.2017.8126011.

[2] M. Linares-Vásquez, G. Bavota, and C. Escobar-Velásquez, "An Empirical Study on Android-Related Vulnerabilities," in *2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR)*, 2017, pp. 2–13, doi: 10.1109/MSR.2017.60.

[3] W. J. Buchanan, S. Chiale, and R. Macfarlane, "A methodology for the security evaluation within third-party Android Marketplaces," *Digit. Investig.*, vol. 23, pp. 88–98, 2017, doi: https://doi.org/10.1016/j.diin.2017.10.002.

[4] B. M. Al-Zadjali, "A Critical Evaluation of Vulnerabilities in Android OS: (Forensic Approach)," *Int. J. Comput. Appl.*, vol. 130, pp. 38–42, 2015.

[5] A. Khandelwal and A. K. Mohapatra, "An insight into the security issues and their solutions for android phones," in *2015 2nd International Conference on Computing for Sustainable Global Development (INDIACom)*, 2015, pp. 106–109.

[6] T. Yang, Y. Yang, K. Qian, D. C.-T. Lo, Y. Qian, and L. Tao, "Automated Detection and Analysis for Android Ransomware," in *2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security, and 2015 IEEE 12th International Conference on Embedded Software and Systems*, 2015, pp. 1338–1343, doi: 10.1109/HPCC-CSS-ICESS.2015.39.

[7] A. Sarkar, A. Goyal, D. Hicks, D. Sarkar, and S. Hazra, "Android Application Development: A Brief Overview of Android Platforms and Evolution of Security Systems," in *2019 Third International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 2019, pp. 73–79, doi: 10.1109/I-SMAC47947.2019.9032440.

[8] S. Yadav, A. Apurva, P. Ranakoti, S. Tomer, and N. R. Roy, "Android vulnerabilities and security," in *2017 International Conference on Computing and Communication Technologies for Smart Nation (IC3TSN)*, 2017, pp. 204–208, doi: 10.1109/IC3TSN.2017.8284477.

[9] Q. Kennemer, "New Android Malware Can Infect Your Phone By Simply Visiting a Website," *Phandroid*, Apr. 2016.

[10] J. Smith, "Google Play Protect rolling out to Android

devices for better security," *iGeneration*, Jul. 2017.
[11] J. Gao, L. Li, P. Kong, T. F. Bissyandé, and J. Klein, "Understanding the Evolution of Android App Vulnerabilities," *IEEE Trans. Reliab.*, vol. 70, no. 1, pp. 212–230, 2021, doi: 10.1109/TR.2019.2956690.
[12] J. Joshi and C. Parekh, "Android smartphone vulnerabilities: A survey," in *2016 International Conference on Advances in Computing, Communication, Automation (ICACCA) (Spring)*, 2016, pp. 1–5, doi: 10.1109/ICACCA.2016.7578857.
[13] K. Kulkarni and A. Y. Javaid, "Open Source Android Vulnerability Detection Tools: A Survey," 2018, [Online]. Available: http://arxiv.org/abs/1807.11840.
[14] W. Chao *et al.*, "An Android Application Vulnerability Mining Method Based On Static and Dynamic Analysis," in *2020 IEEE 5th Information Technology and Mechatronics Engineering Conference (ITOEC)*, 2020, pp. 599–603, doi: 10.1109/ITOEC49072.2020.9141575.