p-ISSN: 2086-6917
e-ISSN: 2656-8047

# ANALISA RESIKO MALWARE DENGAN STATIC MOBSF TERHADAP APLIKASI ANDROID APK

**Imam Himawan[1)], Kevin Septianzah[2)], Irawan Setiadi [3) ]**
Fakultas Teknik dan Ilmu Komputer, Universitas Indraprasta PGRI
Email: imamhimawann@gmail.com
Fakultas Teknik dan Ilmu Komputer, Universitas Indraprasta PGRI
Email: kevin.septianzah24@gmail.com
Fakultas Teknik dan Ilmu Komputer, Universitas Indraprasta PGRI
Email: irawan.setiadi91@gmail.com

**Informasi Artikel:**
Submit: 06-05-2023; Accepted: 07-10-2023; Published: 10-10-2023
Doi : http://dx.doi.org/10.31602/tji.v14i4.11460

**Abstrak**

Observasi dalam penelitian ini melibatkan hasil dari aplikasi android, peneliti mencoba melihat dan memahami analisis yang terkandung khususnya dalam sisi resiko. Tujuannya mendapatkan informasi yang menjadi indikator dampak atau konsekuensi jika aplikasi tersebut terpublikasi agar tidak merugikan banyak orang dikemudian hari dalam menggunakan aplikasi. Peneliti menggunakan aplikasi pengecekan yang bernama MobSF (*Mobile sequrity framework)* dengan metode *static*. Mengingat *malware* kerap menyerang dengan berbagai cara seperti halnya dapat mengubah data target menjadi terenkripsi, masalah yang sering ditemui adalah tingkat resiko aplikasi yang berjenis APK (*Application Package File)* terindikasi berapa persentase dampaknya, mengingat jenis bahasa pemrograman yang digunakan oleh *developer* berjenis *native* atau menggunakan *framework* agar menentukan konsekuensinya. *Developer* membuat aplikasi berdasarkan terhadap kerangka acuan kerja serta bisnis proses yang diusulkan, namun tidak memperhatikan terhadap dari sisi resiko terhadap aplikasi yang dibuat, hasil pengujian fungsional menunjukan terkait fungsi, fitur terhadap aplikasi berjalan dengan baik dengan memastikan sudah melewati UAT (*user Accepted test)* yang sehingga resiko atau konsekuensi dapat dipertanggung jawabkan.

***Keywords**: Aplikasi, Mobsf, Android, malware*



## PENDAHULUAN

Dalam perkembangan dunia *industry* khususnya dalam media informasi dan data, informasi dan data adalah bentuk relasi yang tidak dapat terpisahkan dikarenakan data berupa simbol-simbol, informasi data yang diproses agar dapat dimanfaatkan, serta digunakan untuk menjawab tentang siapa (*who*), apa (*what)*, dimana (*where*) dan kapan (*when*), pengetahuan merupakan aplikasi dari data dan informasi[1].

Permasalahan yang kerap ditemui adalah sejauh mana aplikasi yang sudah jadi dan terpublikasi namun celah efek dikatakan lemah maka peneliti mengobservasi aplikasi yang berjenis apk. Terdapat beberapa berbagai jenis sistem operasi yang dapat digunakan dalam *computer* diantaranya adalah *IOS*, *Linux*, dan *Windows* serta masih banyak lagi sitem operasi lainya, proses kinerja dan *setting* konfigurasi pastinya berbeda [2], secara umum pengguna menggunakan sistem operasi *Windows* yang

p-ISSN: 2086-6917
e-ISSN: 2656-8047

jeninsya berbagai kategori sepertinya *windows* 7, 8, 10 dan sekarang *windows* 11.

*Malware* merupakan perangkat lunak atau *software* yang diciptakan untuk menyusup atau merusak sistem *computer*. Penyebaran *malware* saat ini begitu pesat melalui *usb flasdisk,* iklan-iklan tertentu pada *website*, *malware* merupakan *topic* yang masih sangat terbuka luas untuk dijadikan *object* penelitian dan disarankan menggunakan metode teknik analisa program *malware* dengan memanfaatkan sub-teknik statis yang dikenal denga nama *recerse engineering* [3].

Peneliti mengobervasi hasil aplikasi android yang berjudul Rancang Bangun Aplikasi Sistem Informasi Busana Muslimah Pada Produk Kazhimah Berbasis Android, yang dijadikan bahan pengolahaan data yang akan dianalisis menggunakan MOBSF (*Mobile sequrity framework).*

Sebagai rujukan proses analisis yang dilakukan terhadap aplikasi pakar yang berjenis APK (*Application Package File)* menunjukan bahwa tingkat keamanan masih *relative* sama adanya celah-celah keamanan yang ditemukan [4]. Luaran program yang dibuat oleh *developer* dapat menggunakan beberapa bahasa pemrograman yang berakhir berupa *production* sebagai contoh berbasis *desktop*, *website,* dan *mobile*, khsusnya *android* penulis mengangkat pembahasan mengenai hasil program yang terindikasi *mobile* yaitu *android*.

Dengan adanya aplikasi harapannya dapat menyelesaikan pekerjaan konvensional menjadi efektif, efisiensi dan terkonversi terhadap mesin secara terkomputerisasi dalam pengelolaan data menjadi laporan dengan baik [5].

Serta memastikan resiko dari sisi pengguna aplikasi tersebut. Dengan begitu pengguna tidak merasa resah dan pastinya dapat dijadikan *win solution* dalam menyelesaikan pekerjaan.

## METODE PENELITIAN

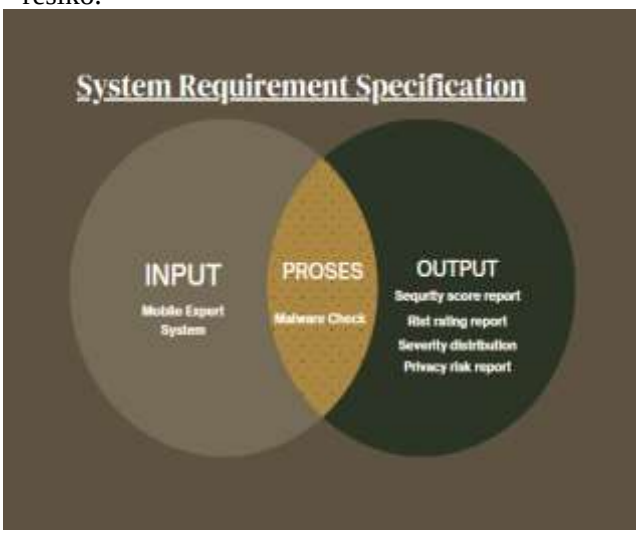
Peneliti melakukan tindakan beberapa tahapan dalam proses pengolahan data sebagai berikut:

1. Pengamatan (*observation)*
   Peneliti mempelajari serta memahami sistem aplikasi *android* serta keterkaitan antara sub sistem satu dengan yang lainya meninjau keamanan sistem.
2. Wawancara (*interview)*
   Merupkan proses diskusi antara peneliti dan *developer* mengenai proses pembuatan aplikasi tersebut.
3. Studi Pustaka
   Teknik pengumpulan data yang bersumber dari informasi yang relevan dalam mendukung proses penilitian berlangsung.

Proses analisa dilakukan dengan mempersiapkan beberapa *tools* sebagai berikut:

a. Install Pyton 3.8-3.9.
b. Install JDK8+.
c. Install Microsoft Visual c++ Build tools.
d. Insttall openssl (Non-light)
e. Install wkhtmltopdf.

Syarat item diatas harus ada serta terinstall dengan baik untuk dapat melakukan pengujian. Dalam hal ini MobSf (*Mobile sequrity framework)* adalah aplikasi *opensource* yang bisa digunakan untuk dapat mengukur kelayakan resiko.




Gambar 1. System requirement specification

## HASIL DAN PEMBAHASAN

Dalam proses tindaklanjut analisa *static* menggunakan MobSf (*Mobile sequrity framework).* Ruang lingkup *file* yang

p-ISSN: 2086-6917
e-ISSN: 2656-8047

dapat digunakan berjenis APK/APKS/XAPK/IPA/ZIPA/APPX. Dalam hal ini peneliti menggunakan yang berjenis APK (*Application Package File)*, implementasi dalam pengerjaan dapat menggunakan kata kunci dalam mesin pencari serta menginput localhost/8000/ terlihat pada gambar dibawah ini.

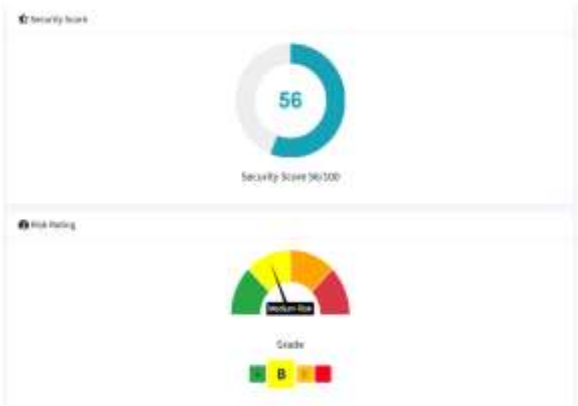
Gambar 2. Halaman utama MobSf

Terlihat halama utama, maka peneliti siap menyisipkan aplikasi yang berekstensi APK terhadap halaman utama agar proses *upload* berjalan yang membutuhkan cukup lama tergantung dengan jumlah *item* data yang ada dalam aplikasi tersebut.


Gambar 3. Hasil analisa terhadap MobSf

Terindikasi beberapa informasi yang dihasilkan oleh sistem terlihat adanya *sequrity score* yang menunjukan nilai 56/100. Informasi yang tersaji dalam kotak *dashboard* sebagai berikut *activities, receveirs, services* dan *providers.* Mengenai analisa ini dapat di kembangkan menjadi metode analisa *dynamic* yang terintegrasi terhadap menu yang tersedia.

Kemudian gambar dibawah ini mengenai laporan *detail* MobSf *application* seperti dibawah ini.




Gambar 4. Sequrity score & Risk rating.

Tersaji dalam hasil analisa bahwa *score* yang didapat terhadap aplikasi *android* yang dijadikan penilaian MobSf dengan menghasilkan angka 56 dari 100. Serta *risk rating* menunjukan nilai di kategori B.


Gambar 5. Severity distribution & Privacy risk.

p-ISSN: 2086-6917
e-ISSN: 2656-8047

Gambar 6. Detailing report MobSf.

Hasil dari analisa aplikasi bernilai 56 dari 100 dan sistem memberikan penilaian resiko di kategori B atau biasa disebut dengan penilaian *medium*. Setiap hasil program *developer* yang sudah terselesaikan dalam proses pengerjaan aplikasi ini dapat di jadikan solusi alternatif dalam penilaian terhadap sisi resiko.

## KESIMPULAN

Setelah melakukan improvisasi dalam proses penilaian maka dapat dijadikan informasi sebagai penilaian akhir bahwa aplikasi *android* yang dibuat nilai resikonya tidak dapat kita prediksi melainkan harus diujikan dalam aplikasi pengujian yang tersedia baik yang berjenis *opensource* atau yang berbayar, agar resiko terlihat.

## REFERENSI

[1] I. Yunianto and K. Adhiyarta, "Jurnal Review: Perbandingan Sistem Operasi Linux Dengan Sistem Operasi Windows," *Jupiter J. Comput. Inf. Technol.*, vol. 1, no. 1, pp. 1–7, 2021, doi: 10.53990/cist.v1i1.77.

[2] M. Mulyadi, "Transisi Data dan Informasi dalam Pengembangan Ilmu Pengetahuan," *Pustakaloka*, vol. 10, no. 1, p. 67, 2018, doi: 10.21154/pustakaloka.v10i1.1237.

[3] T. A. Cahyanto, V. Wahanggara, and D. Ramadana, "Analisis dan Deteksi Malware Menggunakan Metode Malware Analisis Dinamis dan Malware Analisis Statis," *Justindo, J. Sist. Teknol. Inf. Indones.*, vol. 2, no. 1, pp. 19–30, 2017, [Online]. Available: http://jurnal.unmuhjember.ac.id/index.php/JUSTINDO/article/view/1037

[4] I. Himawan, K. Septianzah, and I. Setiadi, "Analisis Keamanan Informasi Malware Terhadap Aplikasi Apk Dengan Metode Static Analysis Menggunakan Mobsf," *JRKT (Jurnal Rekayasa Komputasi Ter.*, vol. 2, no. 02, pp. 122–127, 2022, doi: 10.30998/jrkt.v2i02.6734.

[5] I. Rifai *et al.*, "Tutorial Berwudhu , Dan Mengumandangkan Adzan," vol. 03, no. 01, pp. 94–101, 2022.