Analisis Keamanan Data Pada Aplikasi Android Menggunakan HTTP Canary (Studi Kasus : Siakadu UNESA Mobile)

Muhammad Arief Rahman Ismansyah Putra[1], Agus Prihanto[2]

[1,2] Jurusan Teknik Informatika/Teknik Informatika, Universitas Negeri Surabaya
[1]muhammad.17051204047@mhs.unesa.ac.id
[2]agusprihanto@unesa.ac.id

***Abstrak*— Saat ini, Teknologi berkembang pesat dan kita harus beradaptasi dengan cepat, salah satu teknologi tersebut adalah Internet, manusia dapat mencari informasi apapun dengan mudah, transaksi jual beli, dan sebagainya menggunakan *website*. Unesa sebagai Lembaga Pendidikan juga menggunakan *website* sebagai sumber informasi, untuk lebih memudahkan pengguna, Unesa mempunyai Siakad Unesa versi *mobile*, dalam Siakad Unesa versi *mobile* Mahasiswa bisa mendapatkan informasi seperti absensi, jadwal kuliah, lokasi kelas, nama Dosen, hasil nilai studi, riwayat pembayaran spp, profile Mahasiswa, dan lain sebagainya. Pada aplikasi yang menggunakan REST API tentunya membutuhkan koneksi internet untuk mengirim dan mendapatkan informasi dari server. Aplikasi yang tidak menerapkan standar keamanan akan dengan mudahnya dimodifikasi atau diretas oleh pihak yang tidak bertanggung jawab. Dalam penelitian ini penulis menganalisis keamanan *API* pada aplikasi dengan menggunakan *MitM* untuk *sniffing* dan *REST CLIENT* untuk uji coba *REST API* serta saran untuk menghindari hal seperti yang telah disebutkan terjadi. Http Canary sebagai *MitM* dapat merekam seluruh *traffic* antara *client* dengan *server*, Postman sebagai *REST CLIENT* digunakan untuk uji coba *REST API*. Dari hasil pengujian yang telah dilakukan oleh penulis, aplikasi Siakad Unesa versi *Mobile* dinilai kurang aman, yakni hanya dengan menggunakan HTTP Canary sebagai *MitM*, *MitM* dapat merekam seluruh aktifitas aplikasi walaupun ber-*SSL*, dan *server* tidak memerlukan autentikasi pengguna dari aplikasi saat melakukan *request API*, selain itu tidak adanya pembatasan untuk melakukan *request API* sehingga seseorang dapat melakukan request berulang tanpa hambatan serta tidak adanya proses validasi NIM pengguna yang sedang login dengan yang diminta ke *server* sehingga Mahasiswa dapat meminta informasi tentang Mahasiswa lain.**

***Kata Kunci*— Sniff, MitM, REST CLIENT, REST API, Website, Siakad.**

I. PENDAHULUAN

Pada saat ini, Teknologi berkembang pesat dan kita harus beradaptasi dengan cepat, salah satu teknologi tersebut adalah Internet, dengan adanya internet, manusia dapat mencari informasi apapun dengan mudah, transaksi jual beli, berkomunikasi dengan orang lain, streaming film dan lain sebagainya. Dengan salah satunya menggunakan website.

Dengan adanya Website, Lembaga Pendidikan dapat menggunakannya sebagai sumber informasi yang mudah diakses seperti Siakad Unesa. Website Siakad Unesa memudahkan Mahasiswa untuk mendapatkan informasi seperti absensi, jadwal kuliah, lokasi kelas, nama dosen, hasil nilai studi, history pembayaran spp, profil mahasiswa, dan lain sebagainya. Untuk lebih memudahkan lagi, Siakad Unesa membuat versi mobile yang saat ini tersedia pada Platform android, untuk mengakses data Siakad Unesa memerlukan akun gmail yang terintegrasi dengan organisasi unesa.

API (*Application Programming Interface*) adalah suatu aplikasi berupa antarmuka yang berperan sebagai kurir yang melayani permintaan dari aplikasi lain dan mengembalikan respons ke penerima [3]. Pada API sendiri terdapat 2 bagian, yaitu Private API dan Public API. Private API hanya untuk penggunaan internal dan tidak dirilis ke publik sedangkan Public API dirilis ke publik agar pengembang aplikasi lainnya dapat memanfaatkan layanan yang diberikan oleh penyedia API.

Salah satu arsitektur API adalah REST. REST (Representational State Transfer) merupakan seperangkat prinsip arsitektur yang melakukan transmisi data melalui antarmuka yang terstandarisasi seperti HTTP [6]. Hasil dari RESTful API yaitu berupa format pertukaran JSON (*JavaScript Object Notation*) atau XML (*Extensible Markup Language*). Hasil format tersebut dapat digunakan oleh berbagai Bahasa pemrograman seperti PHP, GO, Python, dan lain sebagainya.

Pada aplikasi yang menggunakan REST API tentunya membutuhkan koneksi internet untuk mengirim dan mendapatkan informasi dari server. Aplikasi yang tidak menerapkan standar keamanan akan dengan mudahnya dimodifikasi atau diretas oleh pihak yang tidak bertanggung jawab.

Http Canary merupakan salah satu aplikasi android yang difungsikan untuk penangkapan dan penganalisa paket HTTP/HTTPS/HTTP2/WebSocket/TCP/UDP. Http Canary dapat digunakan untuk pengujian API pada sebuah aplikasi android serta untuk keperluan edukasi. Http Canary dapat menangkap semua traffic yang dikirimkan dan diterima oleh aplikasi. Disaat aplikasi mengirim dan menerima data biasanya terjadi kebocoran data, pencurian data pribadi, pengubahan data tanpa autentikasi, dan tindak kriminal lainnya yang dapat merugikan orang-orang yang terdampak.

Pada Aplikasi Siakadu Mobile ditemukan kerentanan pada *traffic* API yang tidak terenkripsi dan tidak membutuhkan autentikasi untuk mengakses data serta tidak ada pembatasan (*limit*) pada permintaan API.

Berdasarkan permasalahan tersebut, dilakukanlah penelitian ini dengan tujuan untuk mengetahui seberapa aman aplikasi

Siakadu Mobile dan saran untuk menghindari hal seperti yang telah disebutkan terjadi.

II. PENELITIAN TERKAIT

Penelitian ini dilakukan tidak terlepas dari hasil penelitian-penelitian terdahulu yang pernah dilakukan sebagai bahan perbandingan dan kajian. Adapun hasil-hasil penelitian yang dijadikan perbandingan tidak terlepas dari topik penelitian yaitu mengenai sniffing, keamanan, dan API (Application Programming Interface).

Berdasarkan hasil penelitian yang pernah dilakukan Heru Pranata (2015) dimana melakukan penelitian mengenai keamanan protocol Secure Socket Layer (SSL) terhadap proses sniffing di jaringan. Dengan melakukan ujicoba penyadapan (sniffing) terhadap komunikasi data antara client dan server , hasilnya dapat disimpulkan bahwa Protokol SSL merupakan protokol yang aman dari tindakan sniffing, penggunaan protokol SSL pada jaringan juga sangat penting guna mengamankan data di jaringan ini [4].

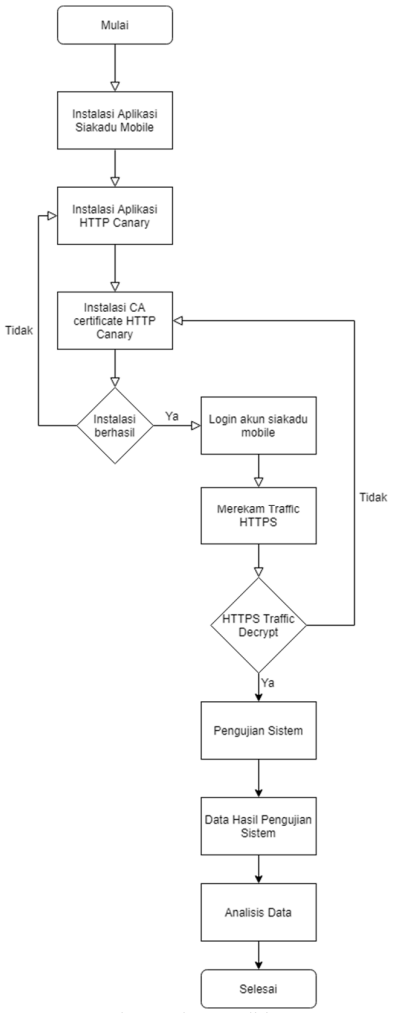
Praful Saxena, dkk (2017) menyajikan penelitian mengenai Analisis network traffic menggunakan packet sniffing tool : Wireshark. Penelitian ini menyimpulkan bahwa Packet Sniffing berguna untuk analisis data saat proses komunikasi ke jaringan, dan berguna untuk monitoring lalu lintas jaringan, analisis lalu lintas jaringan, penyelesaian masalah dan hal lainnya. Packet Sniffers dapat mendapatkan hal seperti password dan username atau informasi sensitif lainnya [5].

Adriant (2015) pernah melakukan implementasi Wireshark untuk penyadapan (SNIFFING) paket data jaringan, Peneliti melakukan sniffing untuk mendapatkan informasi username dan password, dari hasil penelitian Peneliti berhasil mendapatkan informasi username dan password lewat pada jaringan computer [1].

III. METODOLOGI PENELITIAN

3.1 Desain Pengujian

Berikut ini merupakan alur penelitian untuk "Analisis Keamanan Data Pada Aplikasi Android Menggunakan HTTP Canary".



Gbr. 1 Alur Penelitian

3.2 Alat Pendukung Penelitian.

Dalam membangun sistem, dibutuhkan peralatan pendukung yang terdiri dari perangkat keras (hardware) dan perangkat lunak (software). Perangkat yang digunakan dalam penelitian ini adalah sebagai berikut:

3.2.1 *Hardware* (Perangkat Keras)

Perangkat keras yang digunakan dalam mengembangkan sistem ini antara lain:

1. Komputer yang memiliki spesifikasi:
   a. 32/64 *bit Architecture Processor*
   b. 16GB Random Access Memory (RAM)
   c. HDD 2 TB
   d. AMD Ryzen 7 3700x
2. *Mouse*
3. *Keyboard*

3.2.2 Software (Perangkat Lunak)

Agar sistem yang dibangun dapat berjalan dengan baik dan benar maka digunakan beberapa perangkat lunak yang

membantu pengerjaan sistem. Perangkat lunak yang digunakan dalam penelitian ini adalah:

1. *Operating System Windows 10*
2. *Database : MySQL*
3. *Tool Server : Laragon*
4. *Browser Internet : Google Chrome*
5. *Programming language : Hypertext Preprocessor (PHP)*
6. *Editor : Visual Studio Code*
7. *Emulator : MEmu*
8. *Tool packets capture : HTTP Canary*
9. *Tool HTTP client : Postman*

3.3 Pengujian

Adapun tahapan proses yang dilakukan pada penelitian ini yaitu :

A. Proses Merekam *Traffic* pada aplikasi.

Tahap ini adalah sebuah alur proses dimana HTTP Canary sebagai Mitm merekam *traffic* antara Aplikasi Siakadu mobile dengan server Unesa, tahapan ini disebut *sniffing*

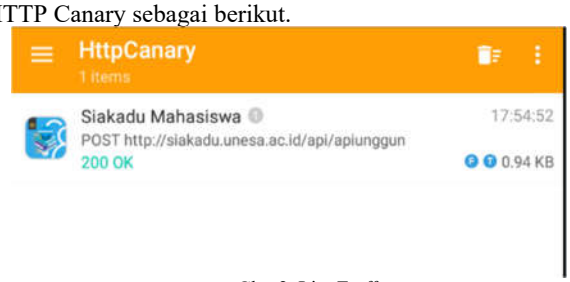
Sniffing merupakan proses pengendusan paket data pada sistem jaringan komputer, yang diantaranya dapat memonitor dan menangkap semua lalu lintas jaringan yang lewat tanpa peduli kepada siapa paket itu di kirimkan.

Aplikasi HTTP Canary memerlukan *root certificate* untuk merekam *SSL/TLS encrypt packets*, tanpa *certificate* tersebut hasil rekaman pada *traffic SSL* akan terenkripsi atau tidak bisa terbaca, SSL berguna untuk mengenkripsi komunikasi antara klien dan *server* [2].

Apabila walaupun sudah melakukan instalasi *root certificate* namun hasil *traffic SSL* masih terenkripsi atau bahkan tidak terekam, berarti keamanan aplikasi tersebut termasuk sudah baik, dan memerlukan metode lain untuk melewati keamanan tersebut.

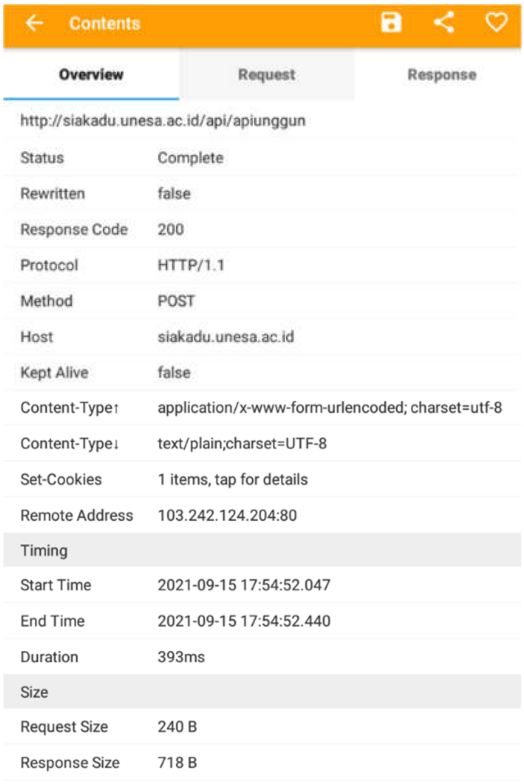
Dalam kasus kali ini *traffic* pada aplikasi siakadu unesa mobile berhasil terekam tanpa tambahan metode lain.

Saat membuka menu yang memerlukan data dari server, aplikasi melakukan request terhadap server dan terekam oleh HTTP Canary sebagai berikut.



Gbr. 2 List *Traffic*

Saat pengguna membuka halaman jadwal perkuliahan, alikasi Siakadu Unesa Mobile melakukan *POST Request* ke *http://siakadu.unesa.ac.id/api/apiunggun* dengan *body kondisi=jadwalperkuliahan&nipd=17051204047* yang dalam artian variabel kondisi menunjukkan halaman yang ingin dibuka dan nipd adalah nomor induk mahasiswa yang ingin dilihat.

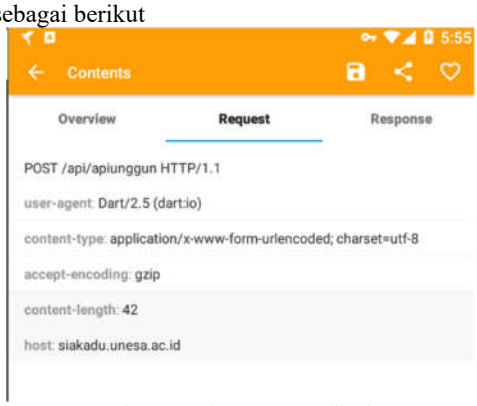


Gbr. 3 Overview Request Aplikasi

B. Proses Pencatatan Semua Request Aplikasi

Pada tahap ini adalah proses untuk merekam semua apa yang dilakukan oleh aplikasi terhadap *server* unesa.

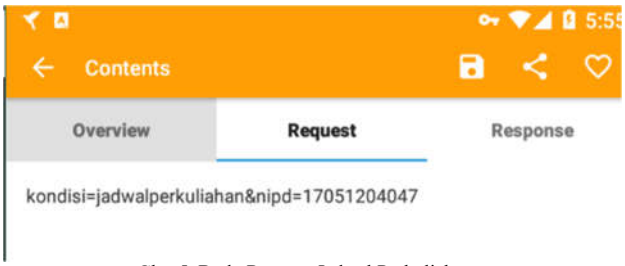
Setelah mencoba semua halaman yang ada pada aplikasi Siakadu Mobile, aplikasi Siakadu Mobile melakukan *POST request* sebagai berikut



Gbr. 4 Header Request Aplikasi

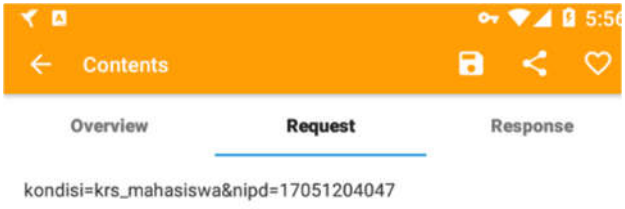
Dengan URL API http://siakadu.unesa.ac.id/api/apiunggun serta *body* yang berbeda disetiap kondisi

1. *kondisi=jadwalperkuliahan&nipd=17051204047* untuk melihat jadwal perkuliahan.



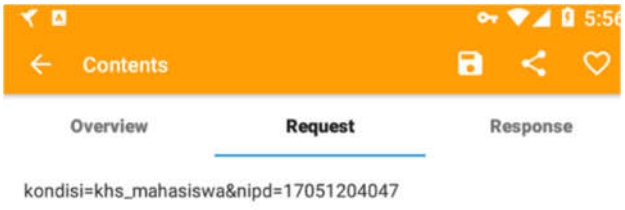
Gbr. 5 Body Request Jadwal Perkuliahan

2. *kondisi=krs_mahasiswa&nipd=17051204047* untuk melihat krs mahasiswa



Gbr. 6 Body Request KRS Mahasiswa

3. *kondisi=khs_mahasiswa&nipd=17051204047* untuk melihat khs mahasiswa

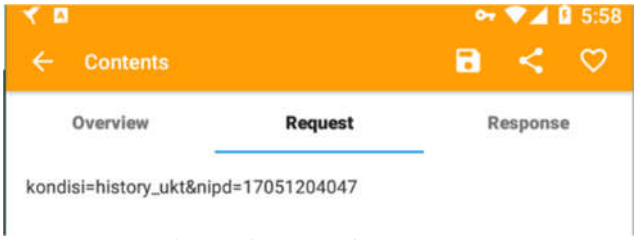


Gbr. 7 Body Request KHS Mahasiswa

4. *kondisi=rekap_absen&nipd=17051204047&action=read* untuk melihat rekap absen

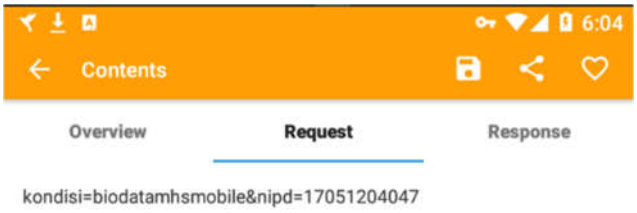
Gbr. 8 Body Request Rekap Absen

5. *kondisi=history_ukt&nipd=17051204047* untuk melihat riwayat UKT



Gbr. 9 Body Request Riwayat UKT

6. *kondisi=biodatamhsmobile&nipd=17051204047* untuk melihat biodata mahasiswa

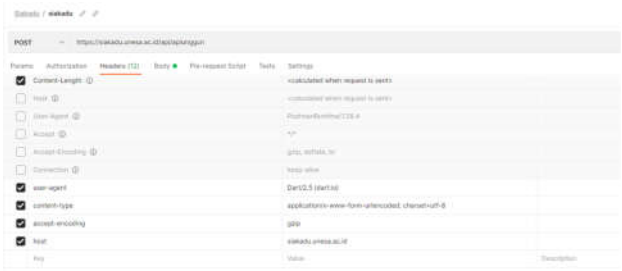


Gbr. 10 Body Request Biodata Mahasiswa

C. Proses Simulasi *Request API*

Proses ini adalah melakukan simulasi request terhadap server unesa tanpa aplikasi Siakadu Mobile dengan menyamakan header dan body menggunakan Software POSTMAN.

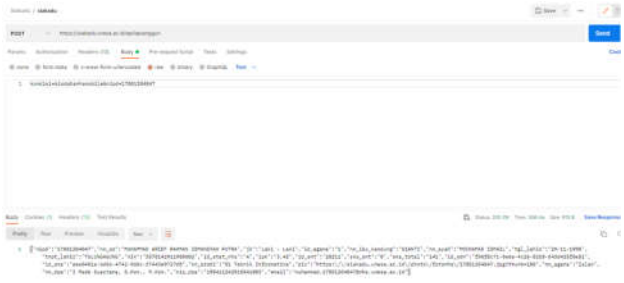
Dikarenakan ternyata aplikasi Siakadu Unesa Mobile tidak menggunakan session pada header untuk melakukan request ke server maka siapapun orang tanpa autentikasi atau akun dapat melakukan request terhadap api siakadu. Disinilah letak kelemahan pada aplikasi, disaat seseorang mempunyai URL API beserta 1 *sample* Nomor Induk Mahasiswa (NIM) maka seseorang dapat melihat data mahasiswa secara massal menggunakan *range* NIM.



Gbr. 11 Postman Header

Gbr. 12 Postman Body

Server unesa mengembalikan data berbentuk JSON.



Gbr. 13 Hasil Postman

D. Proses Pengambilan Data

Apabila simulasi *request API* berhasil, maka tahap proses pengambilan data dapat dilakukan, proses pengambilan data ini menggunakan bahasa pemrograman PHP dengan memanfaatkan CURL untuk simulasi *request API* dan *Json decode* untuk *parsing data*.

```
E:\Data Kuliah\Coding Artikel Ilmiah>php test.php
object(stdClass)#1 (22) {
  ["nipd"]=>
  string(11) "17051204047"
  ["nm_pd"]=>
  string(37) "MUHAMMAD ARIEF RAHMAN ISMANSYAH PUTRA"
  ["jk"]=>
  string(11) "Laki - Laki"
  ["id_agama"]=>
  string(1) "1"
  ["nm_ibu_kandung"]=>
  string(6) "GIANTI"
  ["nm_ayah"]=>
  string(15) "MOCHAMAD ISMAIL"
  ["tgl_lahir"]=>
  string(10) "29-11-1998"
  ["tmpt_lahir"]=>
  string(11) "TULUNGAGUNG"
  ["nik"]=>
  string(16) "3578142911980002"
  ["id_stat_mhs"]=>
  string(1) "A"
  ["ipk"]=>
  string(4) "3.42"
  ["id_smt"]=>
  string(5) "20211"
  ["sks_smt"]=>
  string(1) "0"
  ["sks_total"]=>
  string(3) "141"
  ["id_sdm"]=>
  string(36) "f96f8c71-0e6a-4c26-82b8-d43d425f0e51"
  ["id_sms"]=>
  string(36) "eee0451e-bd6b-4742-9ddc-37443e9727d8"
  ["nm_prodi"]=>
  string(21) "S1 Teknik Informatika"
  ["pic"]=>
  string(67) "https://siakadu.unesa.ac.id/photo/fotomhs/17051204047.jpg?thumb=100"
  ["nm_agama"]=>
  string(5) "Islam"
  ["nm_dpa"]=>
  string(31) "I Made Suartana, S.Kom., M.Kom."
  ["nip_dpa"]=>
  string(18) "198411242015041003"
  ["email"]=>
  string(36) "muhammad.17051204047@mhs.unesa.ac.id"
}
```

Gbr. 14 Hasil CURL PHP

IV. HASIL DAN PEMBAHASAN

A. Data Penelitian

Pada tahap pertama dalam penelitian adalah penentuan dan persiapan yang perlu diuji coba, dalam penelitian ini data yang akan dicoba adalah NIM (nomor induk mahasiswa) lain untuk diambil datanya. Perangkat lunak yang dibuat akan digunakan untuk proses pengambilan data pada mahasiswa lain dalam skala besar.

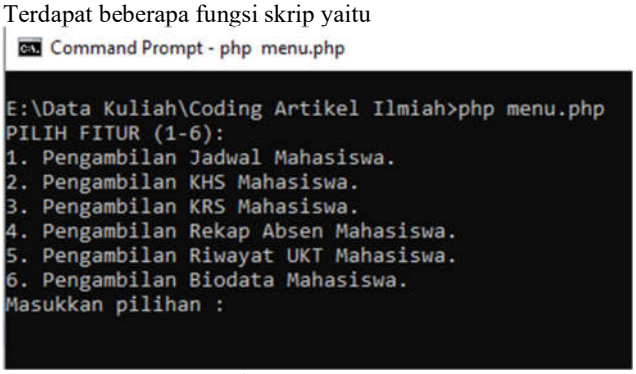
Data yang akan diambil mencakup semua yang ada pada aplikasi Siakadu Mobile Unesa.

Perangkat lunak tersebut akan melakukan pengambilan data pada range NIM (nomor Induk Mahasiswa) yang telah dimasukkan pengguna. Data NIM yang digunakan dalam penelitian ini adalah 17051204000-17051204083.

B. Tampilan Perangkat Lunak

Sistem ini dibangun menggunakan bahasa pemrograman PHP tanpa GUI dan menggunakan CLI (*command-line interface*).

Terdapat beberapa fungsi skrip yaitu

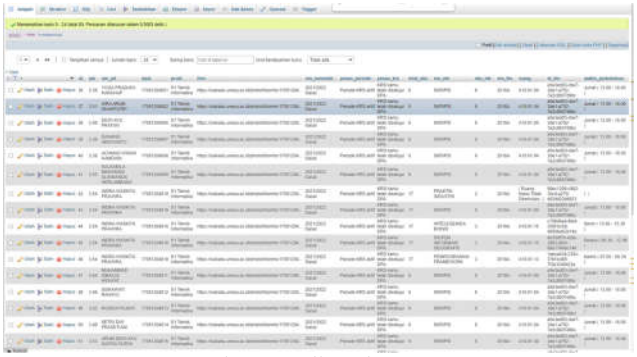


Gbr. 15 Menu CLI

1. Pengambilan Jadwal Mahasiswa.
2. Pengambilan KHS Mahasiswa.
3. Pengambilan KRS Mahasiswa.
4. Pengambilan Rekap Absen Mahasiswa.
5. Pengambilan Riwayat UKT Mahasiswa.
6. Pengambilan Biodata Mahasiswa.
7. Tampilan Halaman hasil.

Skrip proses pengambilan data adalah hasil *export* Postman ke bahasa pemrograman PHP yang diubah dan ditambahkan fitur looping untuk pengambilan data dalam skala besar.

Perangkat lunak akan mengirimkan hasil kedalam database untuk ditampilkan ke halaman hasil, untuk hasil yang telah dimasukkan kedalam database dapat dilihat pada gambar x berikut.

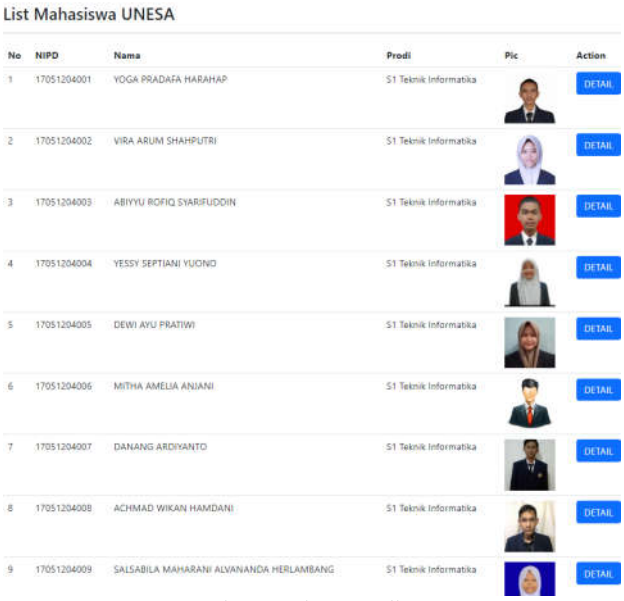


Gbr. 16 Hasil Database

Tampilan halaman hasil yaitu halaman yang menampilkan hasil pengambilan data yang telah dimasukkan dalam database sehingga bisa mudah untuk melihatnya.

Pada halaman tampilan hasil terdapat beberapa fitur yakni pencarian berdasarkan nim, tombol jadwal mahasiswa, krs

mahasiswa, khs mahasiswa, rekap absen mahasiswa, riwayat ukt mahasiswa, dan biodata mahasiswa.



Gbr. 17 Halaman Hasil

Terdapat 4 langkah yang harus dilakukan untuk menjalakan proses pengambilan data.

1. Pengguna terlebih dahulu harus mempunyai instalasi PHP di computer.
2. Pengguna membuka command prompt dan mengarahkan ke folder dimana skrip berada.
3. Pengguna memanggil menu utama yaitu menu.php dengan command "php menu.php" dan memilih fitur.
4. Pengguna memasukkan range Nomor Induk Mahasiswa yang ingin diambil datanya.

Lama waktu dalam proses pengambilan data tergantung dari kecepatan internet pengguna dan banyaknya data yang akan diambil.

C. Hasil Pengujian

Dalam penelitian ini telah dilakukan proses pengambilan data dengan *range* Nomor Induk Mahasiswa 17051204000-17051204083 dan pengujian *limit request*.

Hasil pengujian saat pengambilan data berhasil tanpa hambatan dan dalam waktu yang relatif cepat untuk pengambilan data 83 Mahasiswa yakni 0.7 Menit.

```
E:\Data Kuliah\Coding Artikel Ilmiah>php crawltesttime.php
NIM AWAL = 17051204000
NIM AKHIR = 17051204083
Mahasiswa Tidak ditemukan
Total Data Gathered: 83
Total Execution Time: 0.77760518391927 Mins
E:\Data Kuliah\Coding Artikel Ilmiah>
```

Gbr. 18 Kecepatan Pengambilan Data

Saat pengambilan data server tidak memerlukan autentikasi/*session* pengguna yang sedang aktif sehingga seseorang tanpa otoritas dapat melihat data mahasiswa unesa serta tidak adanya validasi NIM pengguna saat melakukan *request* ke *server* sehingga mahasiswa dapat melihat penuh data mahasiswa lain dengan NIM yang bukan miliknya.


```
object(stdClass)#2 (22) {
  ["nipd"]=>
  string(11) "17051204048"
  ["nm_pd"]=>
  string(19) "JALIS DWI MUTHOHHAR"
  ["jk"]=>
  string(11) "Laki - Laki"
  ["id_agama"]=>
  string(1) "1"
  ["nm_ibu_kandung"]=>
  string(5) "KANTI"
  ["nm_ayah"]=>
  string(12) "JOKO SUPARNO"
  ["tgl_lahir"]=>
  string(10) "20-07-1998"
  ["tmpt_lahir"]=>
  string(5) "NGAWI"
  ["nik"]=>
  string(16) "3521062007980001"
  ["id_stat_mhs"]=>
  string(1) "A"
  ["ipk"]=>
  string(4) "3.62"
  ["id_smt"]=>
  string(5) "20211"
  ["sks_smt"]=>
  string(1) "0"
  ["sks_total"]=>
  string(3) "144"
  ["id_sdm"]=>
  string(36) "f96f8c71-0e6a-4c26-82b8-d43d425f0e51"
  ["id_sms"]=>
  string(36) "eee0451e-bd6b-4742-9ddc-37443e9727d8"
  ["nm_prodi"]=>
  string(21) "S1 Teknik Informatika"
  ["pic"]=>
  string(67) "https://siakadu.unesa.ac.id/photo/fotomhs/17051204048.jpg?thumb=100"
  ["nm_agama"]=>
  string(5) "Islam"
  ["nm_dpa"]=>
  string(31) "I Made Suartana, S.Kom., M.Kom."
  ["nip_dpa"]=>
  string(18) "198411242015041003"
  ["email"]=>
  string(33) "jalis.17051204048@mhs.unesa.ac.id"
}
```


Gbr. 19 Data Mahasiswa lain.

Tidak ditemukan *limit request* dan *server* selalu merespon setelah melakukan percobaan ratusan kali *request* tanpa jeda dengan 1 ip yang sama, dengan demikian dapat dilakukan pengambilan ribuan data mahasiswa tanpa kendala.

```
E:\Data Kuliah\Coding Artikel Ilmiah>php requesttest.php
Masukkan NIM AWAL: 17051204000
Masukkan NIM Akhir: 17051205000

Total Request yang dikirim => 1000 Request
Request Berhasil
E:\Data Kuliah\Coding Artikel Ilmiah>
```

Gbr. 20 Percobaan Request

V. KESIMPULAN

Berdasarkan hasil uji coba yang telah dilakukan mengenai Analisis Keamanan Data Pada Aplikasi Android Menggunakan HTTP Canary, kesimpulan yang diperoleh dari semua proses yang telah diuji dan pembahasan yang telah dilakukan dari penelitian yaitu sebagai berikut.

1. Pada penelitian ini Aplikasi Siakadu UNESA Mobile tidak aman terhadap sniffing. Dalam hal ini Aplikasi sangat mudah untuk direkam trafficnya, hanya menggunakan aplikasi HTTP Canary sebagai Mitm (*Man in the midle*) tanpa metode tambahan lainnya.
2. Telah dilakukan percobaan ratusan *request* tanpa jeda dan dengan 1 ip yang sama namun server merespon semua request API sehingga ditakutkan terjadinya *scraping data* dalam skala besar.
3. Setelah merekam semua aktifitas Aplikasi Siakadu UNESA Mobile terhadap server, ternyata saat Aplikasi Siakadu

UNESA Mobile melakukan request terhadap server, server tidak memerlukan autentikasi atau session pengguna dan langsung merespon apa yang diminta Aplikasi. Sehingga siapapun dapat meniru request aplikasi terhadap server dan mengambil data tanpa autentikasi, hal ini sangatlah berbahaya mengingat data mahasiswa sangatlah banyak.

VI. SARAN

Ada beberapa hal untuk mengatasi sniffing sebagai berikut:

1. Diberlakukan fungsi enkripsi NIM pada aplikasi dan dekripsi NIM pada server dengan tujuan orang lain tidak bisa mengetahui NIM yang sedang direquest oleh aplikasi.
2. Menggunakan JWT (Json Web Tokens), JWT Authentication dapat menghindari mitm (Man in the midle) karena terdapat 3 komponen yaitu Header, Payload, dan Signature sehingga seseorang tidak akan bisa mengubah dan melihat isi saat data akan dikirimkan kepada server karena data terenkripsi dan terdapat verifikasi data pada server, dan sebaliknya seseorang tidak akan bisa melihat hasil data yang dikirim server ke Aplikasi.
3. Diberlakukan validasi NIM pengguna, sehingga saat pengguna mencoba merubah atau melihat data NIM lain server menolak permintaan.
4. Memberi batasan (*limit)* pada request API ke server dalam jumlah wajar yang pengguna gunakan.

UCAPAN TERIMA KASIH

Puji syukur kehadirat Allah SWT atas rahmat serta hidayah yang telah diberikan, sehingga penelitian ini dapat berjalan dengan lancar tanpa halangan apapun. Serta ucapkan terimaksih saya berikan kepada semua pihak yang telah membantu dan memberikan semangat hingga penelitian ini dapat terselesaikan dengan baik.

REFERENSI

[1] Adriant, M. F. (2015) ‘Implementasi Wireshark Untuk Penyadapan (Sniffing) Paket Data Jaringan’, Seminar Nasional Cendekiawan, pp. 224–228.

[2] Arshad Mohammad and Ali Hussain Md. 2016. Secure Framework To Mitigate Man In The Middle Atack Over SSL Protocol. Indian Journal Of Science And Technology, Vol. 9

[3] MuleSoft Videos. (2015). What is an API? [Video]. YouTube., diakses dari https://www.youtube.com/watch?v=s7wmiS2mSXY pada 15 November 2021.

[4] Pranata, H., Abdillah, L. A., & U. E. (2015). Analisis Keamanan Protokol Secure Socket Layer (SSL) Terhadap Proses Sniffing di Jaringan. Proceeding.Student Colloquium Sistem Informasi & Teknik Informatika (SC-SITI). Skripsi, Fakultas Ilmu Komputer. Universitas Bina Darma Palembang.

[5] P. Saxena (2017), Analysis of Network Traffic by using Packet Sniffing Tool : Wireshark,” Int. J. Adv. Res. Ideas Innov. Technol., vol. 3, no. 6, pp. 804–808.

[6] S. Dhingra, "REST vs. SOAP: Choosing the best web service," TechTarget, (2016). [Online]. https://searchapparchitecture.techtarget.com/tip/REST-vs-SOAP-Choosing-the-best-web-service, tanggal akses: 27 November 2021.