Implementasi *SMS-Based One-Time Password Stealing Attack* pada Akun Aplikasi Android menggunakan *Digispark Attiny85*

Asep Setiawan[1)], Ryan Muhammad Azizulfiqar Kamajaya[2)]

1) Rekayasa Keamanan Siber, Keamanan Siber, Politeknik Siber dan Sandi Negara, asep.setiawan@bssn.go.id

2) Rekayasa Keamanan Siber, Keamanan Siber, Politeknik Siber dan Sandi Negara, ryan.muhammad@student.polteksn.ac.id

Abstrak

One-Time Password (OTP) yang mendominasi bidang otentikasi pengguna selama dekade terakhir, merupakan musuh utama bagi setiap penyerang yang mencoba mengakses informasi sensitif. Pakar keamanan khawatir bahwa spoofing pesan SMS dan serangan man-in-the-middle (MITM) dapat digunakan untuk merusak sistem 2FA yang mengandalkan one-time password. Pencurian OTP dapat dilakukan dengan berbagai saluran serangan, salah satunya adalah akses fisik ke perangkat, yakni malware seluler yang mencuri pesan SMS OTP. Serangan telepon seluler dapat dilakukan juga melalui micro-USB dengan menggunakan tampilan antar muka yang umum pada port micro-USB pada telepon seluler. Tujuan dari penelitian ini adalah untuk mengimplementasikan serangan SMS-based OTP stealing attack pada aplikasi Android menggunakan Digispark Attiny85. Metode yang digunakan menggunakan Software Development Lifecycle dengan pendekatan extreme programming. Pada penelitian ini, penulis membuat dua buah skenario penyerangan untuk mencuri OTP berbasis SMS milik korban. Skenario live attack beroperasi dengan memanfaatkan injeksi langsung Digispark Attiny85 ke device korban, sedangkan skenario remote attack memanfaatkan malware yang diunduh melalui script yang diinjeksikan dari Digispark Attiny85. Pengujian menggunakan metode path testing untuk pengujian hardware dan scenario testing untuk pengujian software. Pengujian skenario diterapkan pada beberapa aplikasi antara lain aplikasi mobile banking, dompet digital, instant messaging, dan e-commerce. Hasil penelitian menunjukkan keberhasilan implementasi dari SMS-based OTP Stealing Attack berbasis SMS dalam mengambil OTP akun dari beberapa aplikasi Android korban terbukti dari hasil path testing dan scenario testing.

Kata kunci: android, bad USB, digispark attiny85, eksploitasi, pencurian OTP.

Abstract

One-Time Passwords (OTPs), which have dominated the field of user authentication for the past decade, are a major enemy for any attacker trying to access sensitive information. Security experts are concerned that SMS message spoofing and man-in-the-middle (MITM) attacks can be used to undermine 2FA systems that rely on one-time passwords. OTP theft can be carried out by various attack channels, one of which is physical access to the device, namely mobile malware that steals OTP SMS messages. Mobile phone attacks can also be carried out via micro-USB by using a common interface on the micro-USB port of a mobile phone. The purpose of this research is to implement an SMS-based OTP stealing attack on Android applications using Digispark Attiny85. The method used is a Software Development Lifecycle with an extreme programming approach. In this research, the author created two attack scenarios to steal the victim's SMS-based OTP. The live attack scenario operates by utilizing the direct injection of Digispark Attiny85 into the victim's device, while the remote attack scenario utilizes malware downloaded through a script injected from Digispark Attiny85. Testing uses the path testing method for hardware testing and scenario testing for software testing. Scenario testing is applied to several applications including mobile banking, digital wallet, instant messaging, and e-commerce applications. The results showed the successful implementation of SMS-based OTP Stealing Attack in retrieving account OTPs from several victim Android applications as evidenced by the results of path testing and scenario testing.

Keywords: android, bad USB, digispark attiny85, exploitation, OTP stealing.

1. PENDAHULUAN

One-Time Password (OTP) yang mendominasi bidang otentikasi pengguna selama dekade terakhir, merupakan musuh utama bagi setiap penyerang yang mencoba mengakses informasi sensitif [1]. OTP bertindak untuk menambah identitas (ID) pengguna dan kata sandi yang ada dengan lapisan otentikasi tambahan untuk meningkatkan keamanan pada semua aplikasi yang diotentikasi. Alasannya adalah jika kata sandi diketahui, OTP masih harus dipecahkan juga untuk mendapatkan akses. Otentikasi semacam ini adalah disebut otentikasi dua faktor, jenis otentikasi yang lebih kuat [2]. OTP merupakan salah satu faktor kepemilikan paling populer dalam otentikasi dua faktor (TFA atau 2FA). OTP merupakan kode otentikasi untuk akun-akun aplikasi google, *e-mail*, *mobile banking, instant messaging*, dompet digital, *e-commerce*, dan aplikasi lain yang mengandung data sensitif [1] [3]. Untuk membuat OTP, berbagai metode transmisi seperti transmisi SMS, aplikasi ponsel, dan token tipe perangkat keras [4]. Saat ini, hampir semua transaksi perbankan memerlukan satu tambahan verifikasi SMS *One-Time Password* (OTP). Persyaratan ini telah diwajibkan oleh bank selama beberapa waktu ini [5].

Pakar keamanan telah lama khawatir bahwa *spoofing* pesan SMS dan serangan *man-in-the-middle* (MITM) dapat digunakan untuk merusak sistem 2FA yang mengandalkan *one-time password.* Institut Standar dan Teknologi Nasional (NIST) AS mengumumkan rencana untuk menghentikan penggunaan SMS untuk 2FA dan kata sandi satu kali, karena metode ini rentan terhadap bermacam-macam serangan yang dapat membahayakan kata sandi dan kode tersebut [6]. Peretas dapat memanfaatkan celah keamanan untuk mengalihkan pesan teks penting seperti SMS yang berisi OTP, atau tautan masuk untuk layanan seperti WhatsApp [7].

Serangan telepon seluler dapat dilakukan melalui *micro-USB* dengan menggunakan tampilan antar muka yang umum pada *port micro-USB* pada telepon seluler [8]. *BadUSB* adalah salah satu contoh perkembangan *internet of things* yang masuk ke dalam kelas serangan *firmware* yang terkenal [9]. Sebuah penemuan baru untuk menggantikan *keystroke injection tool* atau USB *Rubber Ducky* yakni menggunakan *Digispark* [8]. Dalam penelitian ini, penulis akan menggunakan *Digispark* (*Attiny85 based)* dimana memiliki kelebihan dari aspek bentuk yang lebih kecil dan harga yang lebih murah akan tetapi performa tidak berbeda jauh dengan arduino sebagai media untuk menginjeksikan perintah untuk melakukan serangan *Stealing SMS One-Time Password* pada Android.

2. LANDASAN TEORI

2.1 *One-Time Password*

One-Time Password (OTP) adalah kata sandi yang hanya berlaku untuk satu sesi login atau transaksi. OTP menghindari sejumlah kekurangan yang terkait dengan kata sandi tradisional (statis). Kelemahan terpenting yang ditangani oleh OTP adalah tidak rentan terhadap serangan *replay*. Artinya potensi penyusup yang berhasil merekam OTP yang sudah digunakan untuk masuk ke layanan atau melakukan transaksi tidak dapat disalahgunakan, karena tidak berlaku lagi [10]. Tujuan dari *one-time password* (OTP) adalah untuk membuat penyerang lebih sulit untuk mendapatkan akses tidak sah ke sumber daya yang dibatasi, seperti akun komputer. [11].

2.2 *Malicious Software*

Malicious Software (*Malware*) adalah program komputer yang dirancang untuk membuat efek berbahaya dan tidak diinginkan pada perangkat korban. *Malware* dianggap menjadi salah satu dari banyak ancaman yang berbahaya bagi pengguna internet. Penyerang biasanya merancang *malware* dengan tujuan dan fungsi tertentu yang sesuai dengan kebutuhan penyerang. Saat sudah diaktifkan, *malware* dapat menyebar melalui internet dan menyebabkan kerusakan pada sistem operasi. *Malware* memanfaatkan kerentanan yang ada pada perangkat dan sistem operasi untuk mengeksploitasi data.

2.3 Digispark

Digispark adalah salah satu papan Arduino terkecil yang pernah diproduksi dan dilindungi hak cipta oleh Digistump LLC. Meskipun kecil, ia juga sangat kuat dan ditenagai oleh *chip* Attiny85 yang memiliki *clock* hingga 16,5Mhz (kira-kira kecepatan yang sama dengan Arduino Uno Board). Jadi Digispark hanyalah sebuah papan mikrokontroler berbasis MCU Attiny85 yang dapat diprogram menggunakan Arduino IDE. Digispark mirip dengan garis Arduino terutama dalam hal cara pemrograman, lebih murah, lebih kecil, dan cukup kuat. Sama seperti kebanyakan papan Arduino yang dilengkapi dengan *port* USB untuk pemrograman dan kadang-kadang sebagai sumber daya, Digispark dilengkapi dengan konektor USB *onboard* yang dapat dicolokkan langsung ke komputer untuk pemrograman perangkat. Papan dapat diberi daya melalui *port* USB yang akan menyalurkan 5V ke papan atau dari sumber eksternal melalui pin VIN yang dapat menerima 7 hingga 35V yang akan diatur ke 5V melalui regulator tegangan 78M05 *onboard* [12].

2.4 Arduino Integrated Development Environment

IDE adalah singkatan dari *Integrated Development Environment*. Arduino IDE adalah perangkat lunak *open-source*, yang digunakan untuk menulis dan mengunggah kode ke Arduino *board*. Aplikasi IDE cocok untuk berbagai sistem operasi seperti Windows, Mac OS X, dan Linux serta mendukung bahasa pemrograman C dan C++. Selain dalam bentuk perangkat lunak, Arduino IDE juga tersedia dalam bentul *Command Line Interface* (CLI). Namun, fungsi dari Arduino IDE berbasis CLI memiliki fungsi yang terbatas diantaranya *compile* dan *upload* [13]. Bahasa pemrograman yang digunakan pada Arduino IDE adalah Arduino *programming language*. Arduino *programming language* dapat dibagi menjadi tiga bagian utama yaitu *functions*, *values*, dan *structure* [14],

2.5 BadUSB

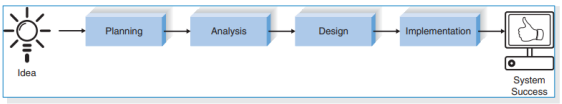
BadUSB adalah *malware* yang baru-baru ini diperkenalkan oleh penulis keamanan pada Juli 2014. *BadUSB* mengeksploitasi lubang keamanan utama dalam desain perangkat USB yang memungkinkan penemunya untuk menipu *network card*, mengalihkan lalu lintas internet, dan menginstal *malware* tambahan. *Keyboard* virtual yang berbahaya bisa menyuntikkan perintah yang berakhir dengan *malware* yang menginfeksi *host*. Perintah jahat ini dapat mengunduh *malware* dari internet, tetapi juga dapat membuatnya dengan cepat, misalnya, dengan "mengetik" konten skrip jahat dan menjalankannya secara langsung. Di konteks ini, antivirus biasa sebagian besar tidak efektif, karena perangkat USB

berbahaya mengeksploitasi kemampuan dasar (contohnya mengetikkan perintah sistem operasi) yang biasa digunakan pengguna diperbolehkan untuk digunakan [15].

3. METODE PENELITIAN

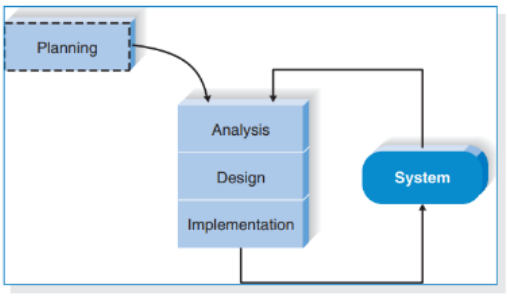
3.1 Desain Penelitian

Metode yang digunakan pada penelitian ini adalah metodologi *System Development Life Cycle* (SDLC). SDLC adalah sebuah proses untuk memahami bagaimana sebuah sistem didesain dan dibangun. Metode SDLC yang digunakan pada penelitian ini memiliki empat tahap, yaitu *planning, analysis, design, dan implementation* [13]. Tahapan–tahapan pada SDLC digambarkan pada Gambar 1.



Gambar 1. Tahapan pada SDLC [16]

Pada pengembangan SDLC terdapat tiga metodologi yang berbeda yaitu *Waterfall Development*, *Rapid Application Development* (RAD) dan *Agile Development.* Gambar 2 memperlihatkan metodologi *Agile Development* yang berfokus pada perampingan SDLC. Ada beberapa pendekatan populer untuk *agile development*, termasuk *extreme programming* (XP), Scrum, dan metode pengembangan sistem dinamis (DSDM) [16]. XP menekankan kepuasan pelanggan dan kerja sama tim, komunikasi, kesederhanaan, umpan balik, dan keberanian adalah nilai inti.



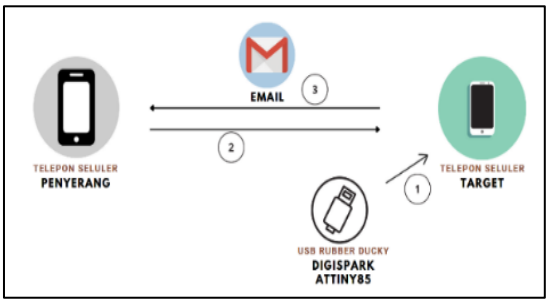
Gambar 2. Tahapan metodologi *Agile Development* [16]

3.2 Perencanaan

Pada penelitian ini, penulis merencanakan dari awal bagaimana skenario serangan yang akan diterapkan untuk menerapkan OTP *stealing attack* menggunakan Digispark Attiny85. Kemudian, penulis membuat perencanaan tentang bentuk program yang akan mendukung pengaplikasian skenario serangan yang telah dibuat sebelumnya. Program yang dibangun menggunakan bahasa Digispark yang kemudian di-*upload* ke perangkat Digispark Attiny85 menggunakan aplikasi perangkat lunak Arduino IDE. Digispark pada penelitian ini digunakan sebagai media implementasi terhadap program yang sudah dibuat dan sebagai media penyerangan.

a. *Live Attack*

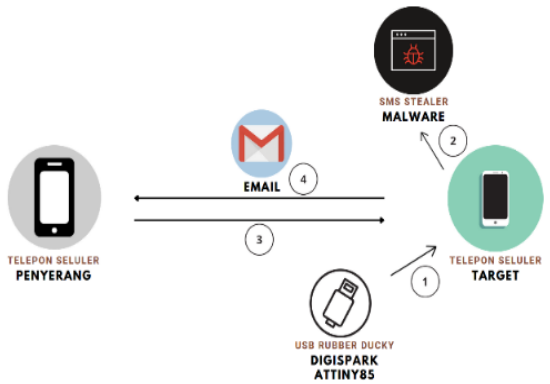
Gambar 3 memperlihatkan tahapan pada skenario serangan *live attack.* Pada skenario serangan pertama, fokus penyerang adalah pemanfaatan waktu yang cukup lama pada tahapan ketika injeksi Digispark Attiny85 ke telepon seluler korban. Keseluruhan tahapan pengambilan data dilakukan oleh Digispark Attiny85 tanpa menggunakan bantuan *malware.* Tahap pertama dimulai dengan penyerang yang harus menginjeksikan Digispark Attiny85 ke telepon seluler korban. Saat sudah terhubung, Digispark Attiny85 akan mengirimkan *script* perintah untuk membuka aplikasi SMS yang akan dieksekusi secara langsung oleh telepon seluler korban. Bersamaan dengan itu, penyerang akan me-*request* OTP untuk masuk ke akun *instant message* korban. *Script* perintah Digispark Attiny85 akan memerintahkan telepon seluler korban untuk menyalin OTP yang ada di SMS kemudian mengirimkannya kepada penyerang melalui *e-mail*. Setelah itu, perintah selanjutnya adalah penghapusan histori SMS OTP dan pengiriman *e-mail* untuk menghilangkan jejak.



Gambar 3. Tahapan pada skenario serangan *live attack*

b. *Remote Attack*

Alasan utama penggunaan serangan jarak jauh adalah untuk melihat atau mencuri data secara ilegal, memasukkan virus atau perangkat lunak berbahaya lainnya ke komputer, jaringan atau sistem lain, dan merugikan pemilik komputer atau jaringan yang ditargetkan [17]. Pada skenario serangan kedua seperti divisualisasikan pada Gambar 4, fokus penyerangannya adalah pemanfaatan serangan jarak jauh yang lebih fleksibel dibanding dengan skenario pertama. Langkah pertama dimulai dengan injeksi Digispark Attiny85 ke telepon seluler korban. Injeksi ini bertujuan untuk mengirimkan *script* yang berisi perintah mengunduh *SMS stealer malware* yang akan bertindak sebagai *spyware* yang akan di-*install* di telepon seluler korban. Apabila sudah terunduh dan ter-*install* maka Digispark Attiny85 dapat dicabut dari telepon seluler korban. Dengan demikian, penyerang bisa me-*request* kapan pun OTP ke ponsel korban, karena akan langsung dikirimkan oleh *spyware* kepada penyerang.

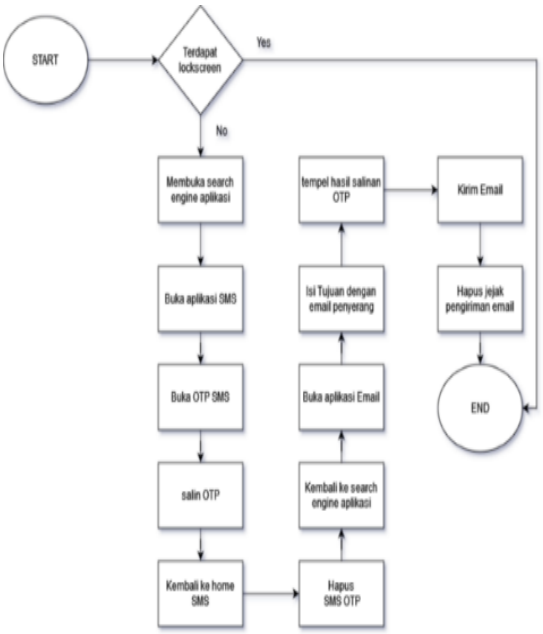


Gambar 4. Tahapan pada skenario serangan *remote attack*

3.3 Desain

a. Program *Live Attack*

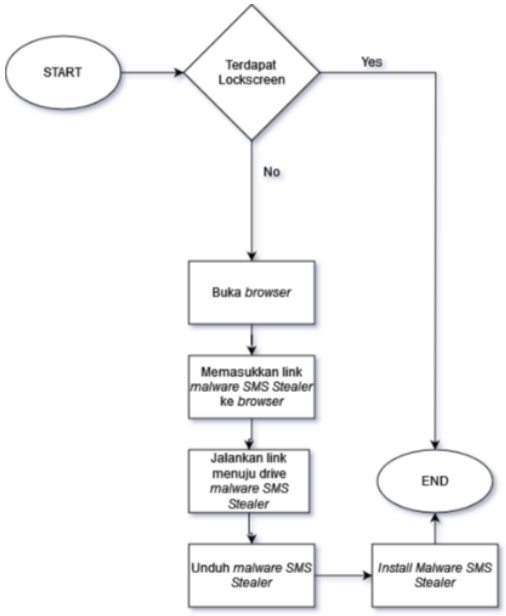
Pada skenario *live attack,* program dibangun untuk membuka telepon seluler korban, kemudian masuk ke dalam aplikasi SMS milik korban, membuka SMS OTP, menyalin OTP, kembali ke *home,* membuka *e-mail*, mengetikkan alamat *e-mail* penyerang, menempel hasil salinan OTP, mengirim *e-mail* ke penyerang, dan menghapus jejak pengiriman *e-mail*. Tahapan tersebut dapat dilihat pada Gambar 5.



Gambar 5. Diagram alir program *live attack*

b. Program *Remote Attack*

Pada skenario *remote attack,* program yang dibangun lebih sederhana, karena yang akan bekerja mencuri OTP adalah *malware* yang ditanam pada telepon seluler korban melalui Digispark Attiny85. Tahapan rinci terlihat pada Gambar 6.



Gambar 6. Diagram alir program *remote attack*

3.4 Implementasi

a. *Path Testing*

Path testing merupakan teknik komprehensif yang menguji setiap jalur program dan memastikan jalur program tersebut dapat berjalan sesuai dengan semestinya [18]. Langkah-langkah teknik *path testing* dijelaskan sebagai berikut:

1) Pembuatan *Flowgraph*

Flowgraph merupakan alur logika dari sebuah *program*. *Flowgraph* dibuat dengan cara memodifikasi notasi pada *flowchart* program yang sudah dibuat. Notasi pada *flowgraph* berupa simbol lingkaran dan panah. Lingkaran atau *node* menyatakan prosedur yang dilakukan program dan panah atau *edge* menyatakan alur dari *path* logika. Angka yang terdapat dalam *node* merupakan *action* yang dilakukan program yang merujuk pada *flowchart* program.

2) Perhitungan *Cyclomatic Complexity* (CC)

Cyclomatic complexity merupakan hasil pengukuran yang menunjukkan ukuran kompleksitas logika dari sebuah program. *Cyclomatic complexity* dihitung berdasarkan jumlah *node* dan *edge* yang terdapat pada *flowgraph*. *Cyclomatic complexity* dilambangkan dengan V(G). Adapun perhitungan *cyclomatic complexity* menggunakan rumus sebagai berikut:

$$V(G) = E - N + 2$$

Keterangan
V(G) = *Cyclomatic Complexity*
E = jumlah *edge* yang terdapat pada *flowgraph*
N = jumlah *node* yang terdapat pada *flowgraph*

3) Penentuan *Independent Path*

Independent path merupakan jalur *flowgraph* yang menghubungkan *node* awal dengan *node* akhir. Oleh karena itu, *independent path* satu dengan *independent path* lainnya bisa saja berbeda. Jumlah *independent path* didapatkan dari hari perhitungan *cyclomatic complexity* yang telah dilakukan pada tahap sebelumnya.

4) Pengujian *Test Case*

Pengujian *test case* dijalankan dengan cara menjalankan semua alur logika yang dibuat. Berdasarkan hasil pengujian tersebut akan terlihat apakah program berjalan sesuai dengan semestinya atau tidak. Pada pengujian *test case* ini, setiap program pada *live* dan *remote attack* akan diuji untuk menentukan apakah setiap fungsi berjalan secara sistematik untuk menjalankan masing-masing serangan.

b. *Scenario Testing*

Skenario adalah cerita hipotetis, digunakan untuk membantu seseorang memikirkan masalah yang kompleks atau sistem [19]. ***Scenario Testing*** dalam pengujian perangkat lunak adalah metode di mana skenario aktual digunakan untuk menguji aplikasi perangkat lunak alih-alih kasus uji.

1) *Mobile Banking*

Aplikasi jenis ini merupakan aplikasi dengan tingkat resiko tertinggi apabila akun tersebut dicuri. Keamanan OTP menjadi faktor penting dalam menjaga keamanan dari akun *Mobile Banking.* Dilansir dari kompas.com, Direktorat Tindak Pidana Siber Bareskrim Polri meringkus 10 pelaku pembobolan sebanyak 3.070 rekening dengan modus menipu korban demi mendapatkan kode *one-time password* (OTP). Total kerugian yang diderita para nasabah mencapai Rp21 miliar [20]. Menurut laporan dari CNBC Indonesia (2022), beberapa bank digital terbesar di Indonesia diduduki oleh SeaBank, disusul oleh Bank Neo, Bank Jago, dan Bank BRI [21]. Pada pengujian ini, penulis menggunakan 3 aplikasi *mobile banking* yaitu SeaBank, Bank Jago, dan BRI Mobile.

2) Dompet Digital

Aplikasi dompet digital merupakan aplikasi yang marak digunakan saat ini untuk pembayaran berbagai keperluan sehari-hari. Hasil riset yang berjudul "*Consistency That Leads: 2023 E-Wallet Industry Outlook*" menunjukkan 71% responden aktif menggunakan dompet digital untuk berbagai macam transaksi mereka [22]. Aplikasi ini menggunakan OTP juga sebagai metode otentikasinya. Menurut laporan dari Bank Indonesia (2023), 5 dompet digital dengan pengguna terbanyak antara lain Shopeepay, OVO, GoPay, DANA, dan LinkAja [23]. Pada pengujian ini, penulis menggunakan empat aplikasi di antaranya adalah OVO, DANA, LinkAja, dan Gopay.

3) *E-Commerce*

Pada tahun 2021 yang lalu, Sebanyak 88,1% pengguna internet di Indonesia memakai layanan *e-commerce* untuk membeli produk tertentu dalam beberapa bulan terakhir. Persentase tersebut merupakan yang tertinggi di dunia dalam hasil survei *We Are Social.* Aplikasi jenis ini menggunakan OTP untuk otentikasi masuk ke akunnya. Menurut data dari Databoks (2023), lima aplikasi *e-commerce* dengan pengguna terbanyak di Indonesia antara lain Shopee, Tokopedia, Lazada, BliBli, dan Bukalapak [24]. Pada penelitian ini, penulis menggunakan empat aplikasi antara lain Shopee, Tokopedia, Lazada, dan Bukalapak.

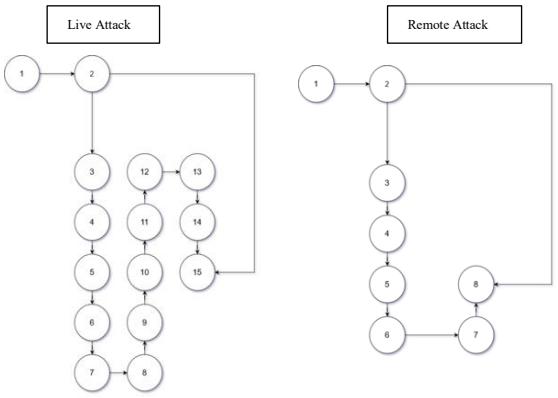
4) *Instant Messaging*

Aplikasi *instant messaging* tidak bisa lepas dari kehidupan masyarakat saat ini. Aplikasi pesan instan memungkinkan penggunanya untuk menggunakan berbagai fitur seperti *group chat*, mengirim pesan gambar, video, bahkan stiker. Kaya akan fitur serta dapat diakses secara gratis maupun berbayar dengan biaya rendah, aplikasi pesan instan dengan cepat meraup popularitas di kalangan masyarakat dunia [25]. Aplikasi ini juga menggunakan OTP untuk otentikasi terakhir sebelum berhasil masuk ke akun pribadi. Dilansir dari *Pewarta.co.id* (2022), lima aplikasi *instant messaging* dengan pengguna terbanyak di Indonesia antara lain *Whatsapp, Messenger, WeChat, Telegram, dan Line* [26]. Pada penelitian ini, penulis menggunakan tiga aplikasi yakni *Whatsapp, Mesenger, dan Line.*

4. HASIL DAN PEMBAHASAN

a. *Path Testing*

Pada tahapan awal, Gambar 7 memperlihatkan hasil dari pembuatan *flowgraph* dari masing-masing jenis serangan.



Gambar 7. *Flowgraph* dari kedua skenario

Setelah didapatkan bentuk *flowgraph*, maka

selanjutnya dihitung nilai *Cyclomatic Complexity* yang menghasilkan nilai 2 pada setiap skenario. Selanjutnya dari nilai tersebut dibuatlah *Independend Path* untuk pengujian *Test Case.*

b. *Scenario Test*

Pada serangan *live attack* program hanya membutuhkan waktu kurang lebih 37 detik untuk mengambil OTP dari *handphone* korban, sehingga berpengaruh terhadap keberhasilan skenario apabila waktu berlaku OTP lebih kecil dibandingkan dengan waktu yang dibutuhkan untuk serangan. Sedangkan untuk melakukan *remote attack,* setelah *malware* dipasang di *handphone* korban, maka pesan akan masuk secara langsung ke penyerang sesaat setelah pesan OTP masuk ke *handphone* korban. Tabel 1 dan Tabel 2 merupakan hasil kesesuaian terhadap hasil yang diharapkan berdasarkan beberapa *test case.*

Tabel 1 Pengujian *node* pada program *live attack*

Node	Deskripsi *Test Case*	Hasil yang Diharapkan	Hasil
1	*Start*	Perangkat Digispark tersambung ke *handphone* korban	Sesuai
2	*Choose next step with lockscreen or not*	Langkah selanjutnya terpilih bergantung dari *lockscreen* atau tidak	Sesuai
3	*Open application search engine*	*Search engine* aplikasi *handphone* terbuka	Sesuai
4	*Open SMS application*	Aplikasi SMS terbuka	Sesuai
5	*Open SMS when containing the OTP*	SMS yang berisi OTP terbuka	Sesuai
6	*Copy OTP*	OTP tersalin	Sesuai
7	*Back to home of SMS application*	Kembali ke halaman awal aplikasi SMS	Sesuai
8	*Delete SMS OTP*	SMS yang berisi OTP terhapus	Sesuai
9	*Back to application search engine*	Kembali ke beranda awal search engine aplikasi	Sesuai
10	*Open email application*	Aplikasi *email* (gmail) terbuka	Sesuai
11	*Fill the destination email address*	Alamat penyerang dimasukkan pada kolom alamat tujuan	Sesuai
12	*Paste the OTP*	OTP yang sudah disalin ditempelkan pada isi *email*	Sesuai
13	*Send the email*	*Email* yang berisi OTP terkirim ke alamat penyerang	Sesuai
14	*Delete the send email*	Jejak pengiriman *email* yang terkirim dihapus	Sesuai
15	*End*	Program selesai, perangkat Digispark dapat dicabut dari *handphone*	Sesuai

Tabel 2 Pengujian Node pada program *remote attack*

Node	Deskripsi *Test Case*	Hasil yang Diharapkan	Hasil
1	*Start*	Perangkat Digispark tersambung ke *handphone* korban	Sesuai
2	*Choose next step with locksreen or not*	Langkah selanjutnya terpilih bergantung dari *lockscreen* atau tidak	Sesuai
3	*Open application search engine*	*Search engine* aplikasi *handphone* terbuka	Sesuai
4	*Open browser*	Aplikasi *browser* terbuka	Sesuai
5	*Enter the link*	Link berisi *malware* dimasukkan ke *browser*	Sesuai
6	*Running the link*	Link menuju drive berisi *malware* dijalankan	Sesuai
7	*Download the malware*	*Malware* terunduh	Sesuai
8	*Install the malware*	*Malware* terpasang pada *handphone*	Sesuai
9	*End*	Program selesai, perangkat Digispark dapat dicabut dari *handphone*	Sesuai

1) *Mobile Banking*

Hasil pengujian menunjukkan bahwa aplikasi BRI Mobile tidak dapat dilakukan serangan karena OTP yang telah di-*request* tidak masuk ke perangkat korban sehingga belum bisa mendapatkan hasil pengujian. Aplikasi SeaBank berhasil dicuri OTPnya melalui serangan *live attack* maupun *remote attack* dengan waktu OTP yang diberikan selama 1 menit. Begitupun dengan Bank Jago, memiliki waktu 1 menit untuk memasukkan OTP dan tidak ada waktu kedaluwarsa untuk OTP, maka pengujian *live* dan *remote attack* berhasil.

2) *E-Commerce*

Hasil pengujian pada aplikasi *e-commerce* mendapatkan keberhasilan pada aplikasi Lazada dan Bukalapak dengan waktu tunggu OTP selama 1 menit dan 2 menit. Sedangkan pada aplikasi Shopee, program tidak dapat diimplementasikan karena OTP hanya bisa dikirimkan melalui Whatsapp. Sedangkan pada aplikasi Tokopedia, program berhasil dengan syarat, yakni waktu tunggu OTP hanya 30 detik namun akibat status OTP yang masuk tidak memiliki waktu kedaluwarsa sehingga OTP masih bisa dimasukkan bahkan setelah melewati batas waktu yang diberikan.

3) *Dompet Digital*

Pada aplikasi dompet digital, OVO merupakan satu-satunya aplikasi yang gagal diimplementasikan program OTP *stealing attack*

karena OTP hanya bisa dikirimkan melalui *e-mail* bukan SMS. Sedangkan untuk Dana, LinkAja, dan Gopay memiliki sistem yang berhasil diterapkan serangan OTP *stealing attack*. Dana membutuhkan paling tidak 35 detik dalam menjalankan keseluruhan program.

2. Pada program *remote attack,* OTP akan dikirimkan kepada penyerang tepat setelah OTP masuk ke *handhpone* korban, karena *malware* bertindak sebagai *spyware* yang mengawasi setiap SMS yang masuk.
3. Aplikasi Tokopedia merupakan aplikasi paling aman dari program OTP *stealing attack* di antara keseluruhan aplikasi yang diuji oleh penulis dari segi penerapan sistem OTP yang kedaluwarsa ketika melewati batas waktu. Tokopedia hanya mempunyai waktu 30 detik dengan maksimal percobaan OTP hanya dua kali sehingga akan menyulitkan penyerang ketika salah memasukkan OTP lebih dari 2 kali. Gopay dan Lazada juga merupakan aplikasi paling aman karena memiliki waktu tunggu 1 menit dan terdapat kedaluwarsa untuk masa aktif OTP-nya. Namun, di samping itu aplikasi memiliki waktu tunggu yang cukup lama yakni 1 menit dan bisa ditebak OTP-nya untuk beberapa kali.
4. Aplikasi yang gagal diterapkan program OTP *stealing attack* merupakan aplikasi yang tidak menggunakan SMS dalam pengiriman OTP dan satu aplikasi tidak mendapatkan OTP dari aplikasi terkait.

Tabel 3 Hasil pengujian *scenario testing*

No	Jenis Aplikasi	Nama Aplikasi	Live Attack	Remote Attack	Waktu OTP	Status OTP Setelah Waktu Habis	Keterangan
1	Mobile Banking	BRI Mobile	G	G	2 menit 30 detik	Aktif	OTP tidak masuk
		SeaBank	B	B	1 menit	Aktif	
		Bank Jago	B	B	1 menit	Aktif	
2	Dompet Digital	OVO	B	B	1 menit	Aktif	
		Dana	B	B	1 menit	Aktif	
		LinkAja	B	B	1 menit 30 detik	Aktif	
		Gopay	B	B	1 menit	Kedaluwarsa	
3	E-Commerce	Shopee	G	G	30 detik	Kedaluwarsa	OTP masuk ke *email*
		Tokopedia	BS	B	30 detik	Aktif untuk 2 kali percobaan	
		Lazada	B	B	1 menit	Kedaluwarsa	
		Bukalapak	B	B	2 menit	Aktif	
4	Instant Messaging	WhatsApp	B	B	2 menit 5 detik	Aktif	
		Messenger	B	B	Tanpa waktu	Aktif	
		Line	B	B	Tanpa waktu	Aktif	

Keterangan:
- G = Gagal
- B = Berhasil
- BS = Berhasil dengan Syarat

memiliki waktu tunggu 1 menit, LinkAja 1 menit 30 detik, dan Gopay selama 1 menit. Namun di antara keempat aplikasi yang diuji, LinkAja memiliki keamanan paling baik karena OTP akan kedaluwarsa setelah waktu tunggu telah habis.

4) *Instant Messaging*

Tabel 3 merupakan hasil pengujian terakhir. Berdasarkan hasil tersebut, aplikasi *instant messaging* menjadi jenis aplikasi yang paling rentan terhadap OTP *stealing attack.* Dibuktikan dengan berhasilnya serangan pada ketiga aplikasi yang diuji. Whatsapp memiliki waktu 2 menit 5 detik untuk waktu tunggu OTP. Kemudian Messenger tidak memiliki waktu tunggu OTP, dan Line memiliki batas waktu 30 menit. Namun ketiga aplikasi tersebut, memiliki status OTP yang masih aktif meski telah melewati batas waktu yang telah ditentukan. Pada aplikasi Tokopedia, program berhasil dengan syarat, yakni waktu tunggu OTP hanya 30 detik namun akibat status OTP yang masuk tidak memiliki waktu kedaluwarsa sehingga OTP masih bisa dimasukkan bahkan setelah melewati batas waktu yang diberikan.

c. Pembahasan

Setelah program diimplementasikan dan diuji pada sistem, akan dilakukan analisis mengenai kekurangan dan keberhasilan dari program yang telah diterapkan. Berikut adalah hasil analisis dari penerapan program OTP *stealing attack*.

1. Program *live attack* sangat bergantung pada waktu yang ditetapkan oleh aplikasi dalam meng-*input* OTP, hal itu dikarenakan proses *live attack*

5. KESIMPULAN

Berdasarkan hasil implementasi dan uji coba program OTP *stealing attack* menggunakan Digispark Attiny85 pada *handphone* Android, dapat disimpulkan bahwa program tersebut sukses dijalankan dengan baik. Pengujian kompatibilitas perangkat dan skenario terhadap 14 aplikasi menunjukkan bahwa 11 aplikasi berhasil diterapkan dengan program tersebut, satu aplikasi berhasil

dengan syarat, dan dua aplikasi gagal. Hasil pengujian juga mengungkapkan bahwa keberhasilan program SMS-*Based* OTP *Stealing Attack* dipengaruhi oleh beberapa faktor, termasuk kecepatan internet pada *handphone* korban yang memengaruhi skema serangan jarak jauh, kecepatan aplikasi dalam mengirim OTP, waktu tunggu dari OTP yang mempengaruhi kinerja serangan langsung, serta status kedaluwarsa OTP setelah batas waktu yang juga memengaruhi pencurian OTP.

REFERENSI

[1] I. Tzemos, A. P. Fournaris and N. Sklavos, "Security and Efficiency Analysis of One Time Password Techniques," *Proceedings of the 20th Pan-Hellenic Conference on Informatics,* no. 67, pp. 1-5, 2016.

[2] Y. Huang, Z. Huang, H. Zhao and X. Lai, "A new One-time Password Method," *International Conference on Electronic Engineering and Computer Science ,* vol. 4, pp. 32-37, 2013.

[3] Thales, "One Time Password (OTP, TOTP) : definition, examples," Thales, 2018. [Online]. Available: https://www.thalesgroup.com/en/markets/digital-identity-and-security/technology/otp. [Accessed 11 Desember 2022].

[4] C. Yoo, B.-T. Kang and H. K. Kim, "Case study of the vulnerability of OTP implemented in internet banking systems of South Korea," *Springer Science+Business Media New York ,* pp. 1-15, 2014.

[5] S. Hamdare, V. Nagpurkar and J. Mittal, "Securing SMS Based One Time Password Technique from Man in the Middle Attack," *International Journal of Engineering Trends and Technology (IJETT),* vol. 11, no. 3, pp. 154- 158, 2014.

[6] K. Richard, "One-Time Password (OTP)," Techtarget, September 2021. [Online]. Available: https://www.techtarget.com/searchsecurity/definition/one-time-password-OTP. [Accessed 11 Desember 2022].

[7] Y. Cox, "Your OTP is hacked! Here's how hackers are stealing your personal information," Zeenews, 19 Maret 2021. [Online]. Available: https://zeenews.india.com/technology/your-otp-is-hacked-here-s-how-hackers-are-stealing-your-personal-information-2349019.html. [Accessed 21 November 2022].

[8] A. N. Firdaus, H. E. Wahanani and M. Idhom, "Uji Serangan Remote Exploit Pada Telepon Seluler IOS Menggunakan Digispark Attiny85," *Jurnal Informatika dan Sistem Informasi,* vol. 1, no. 2, pp. 557-562, 2020.

[9] H. Lu, Y. Wu, S. Li, Y. Lin, C. Z. and F. Zhang, "BADUSB-C: Revisiting BadUSB with Type-C," *Southern University of Science and Technology,* pp. 1-12, 2021.

[10] K. Aravindhan and R. R. Karthiga, "One-time Password: A Survey," *International Journal of Emerging Trends in Engineering and Development,* vol. 1, no. 3, pp. 613-623, 2013.

[11] K. Alghathbar and H. A. Mahmoud, "Noisy Password Scheme: A New One Time Password System," *IEEE,* pp. 841-846, 2009.

[12] electronics-lab.com, "Intruduction to Digispark-A Smaller, Cheaper and Powerful Arduino Board," electronics-lab.com, 4 April 2018. [Online]. Available: https://www.electronics-lab.com/introduction-digispark-smaller-cheaper-powerful-arduino-board/. [Accessed 23 November 2022].

[13] S. Potocký and J. Štulrajter, "The Human Interface Device (HID) Attack on Android Lock Screen Non-Biometric Protections and It Computational Complexity," *Science and Military,* pp. 29-36, 2022.

[14] J. Sanal, "How DIY USBs are used to Hack Computers? HID Attack using Digispark & Arduino," Hackers Grid, 25 Mei 2022. [Online]. Available: https://hackersgrid.com/2022/05/usb-hid-attacks.html. [Accessed 29 November 2022].

[15] F. Griscioli, M. Pizzonia and M. Sacchetti, "USBCheckIn: Preventing BadUSB attacks by forcing human-device interaction," 2016 14th Annual Conference on Privacy, Security and Trust (PST), Auckland, New Zealand, 2016, pp. 493-496, doi: 10.1109/PST.2016.7907004..

[16] A. Dennis, B. H. Wixom and R. M. Roth, System Analysis and Design, United States of America: John Wiley & Sons, Inc., 2012.

[17] M. Rouse, "Remote Attack," Techopedia, 9 September 2013. [Online]. Available: https://www.techopedia.com/definition/4078/remote-attack. [Accessed 11 April 2023].

[18] V. P. Katiyar and M. S. Patel, "White-Box Testing Technique For Finding Defects," *Globa Journal For Research Analysis ,* vol. 8, no. 7, pp. 1-3, 2019.

[19] C. Kaner, "An Introduction to Scenario Testing," 2013. [Online].

[20] A. R. Pasha, "Awas Rekening Dibobol, Begini 6 Cara Jaga Kode OTP," Cermati, 3 Mei 2023. [Online]. Available: https://www.cermati.com/artikel/cara-jaga-kode-otp. [Accessed 8 Mei 2023].

[21] A. Hidayah, "Top 7 Bank Digital Terbesar, Juaranya Melesat Sendirian," CNBC Indonesia,

23 November 2022. [Online]. Available: https://www.cnbcindonesia.com/market/20221123093109-17-390387/top-7-bank-digital-terbesar-juaranya-melesat-sendirian. [Accessed 18 Agustus 2023].

[22] A. P. Brilian, "Survei: 71% Orang RI Pakai Dompet Digital, Mana yang Paling Laris?," detikfinance, 29 November 2022. [Online]. Available: https://finance.detik.com/fintech/d-6433675/survei-71-orang-ri-pakai-dompet-digital-mana-yang-paling-laris. [Accessed 8 Mei 2023].

[23] C. S. Wulandari, "Dompet Digital Naik Daun, Membetot Minat kala Pandemi," BI Institute, 31 Maret 2023. [Online]. Available: https://www.bi.go.id/id/bi-institute/BI-Epsilon/Pages/Dompet-Digital--Naik-Daun,-Membetot-Minat-Kala-Pandemi.aspx. [Accessed 18 Agustus 2023].

[24] A. Ahdiat, "5 E-Commerce dengan Pengunjung Terbanyak Kuartal IV 2022," Databoks, 31 Januari 2023. [Online]. Available: https://databoks.katadata.co.id/datapublish/2023/01/31/5-e-commerce-dengan-pengunjung-terbanyak-kuartal-iv-2022. [Accessed 18 Agustus 2023].

[25] D. Angelia, "Aplikasi Pesan Instan dengan Pengguna Terbanyak di Dunia 2022," Goodstats, 20 April 2022. [Online]. Available: https://goodstats.id/article/aplikasi-pesan-instan-dengan-pengguna-terbanyak-di-dunia-2022-3tggF. [Accessed 5 Mei 2023].

[26] Tim Redaksi, "Aplikasi Chatting Paling Banyak Pengguna di Indonesia," Pewarta.co.id, 10 Maret 2022. [Online]. Available: https://www.pewarta.co.id/2022/03/aplikasi-chatting-paling-banyak-pengguna-di-indonesia.html. [Accessed 18 Agustus 2023].