ANALISIS KEAMANAN APLIKASI *EMAIL* BAWAAN *ANDROID* DAN *GMAIL* PADA JARINGAN NIRKABEL

Hamid

Jurusan Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Indonesia
Jl. Kaliurang Km. 14,5 Sleman, Yogyakarta
E-Mail : hamid@uii.ac.id

ABSTRACT

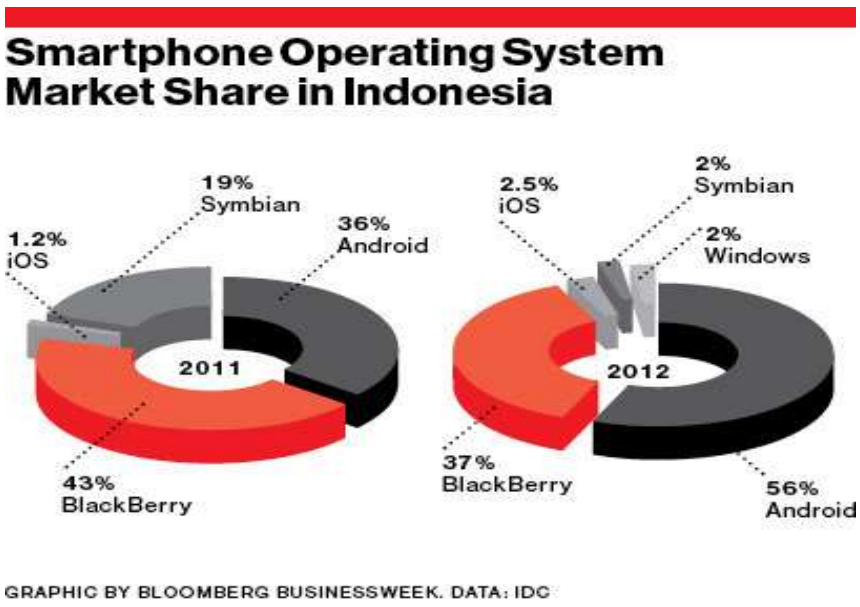
ARP Poisoning / ARP Spoofing and sniffing will always threaten wireless network user. The rise of android-based smartphone user and internet wireless network provider, allowing smartphone users to access anything including email from anywhere they want. This paper provides an overview of comparison of security analysis between use the embedded android email application with gmail email application that is connected to wireless network, as well as providing an overview of possible attack methods. This paper also provides solutions to prevent attacks from the vulnerability found.

Keywords : *ARP Poisoning, ARP Spoofing, Sniffing, Wireless Network, Email Application.*

1. LATAR BELAKANG

Android merupakan salah satu sistem operasi yang dikembangkan untuk perangkat bergerak yang bersifat *open source*. Sejak dirilis pada tahun 2007 (Alliance, 2007), ponsel pintar berbasis *android* langsung menguasai pasar ponsel pada kuartal keempat tahun 2010 dengan penguasaan pasar sebesar 32,5 % dari total penjualan ponsel pintar di dunia (Ricknäs, 2011).

Pada kuartal ketiga tahun 2013 ponsel pintar berbasis *android* menguasai pasar dunia dengan penguasaan pasar sebesar 81,9 % dengan total penjualan 205.222 unit (Meulen, 2013). Di Indonesia sendiri ponsel pintar berbasis *android* menguasai pasar dengan penguasaan pasar sebesar 56 % pada tahun 2012 (Einhorn, 2012).



Gambar 1. Penguasaan Pasar Ponsel Pintar Berbasis *Android* di Indonesia.
(Einhorn, 2012)

Ponsel pintar terutama ponsel pintar berbasis *android* yang semakin mampu menggantikan tugas yang sebelumnya dilakukan oleh *notebook* ataupun komputer personal, menghadirkan fenomena ketergantungan dan menjadikan ponsel pintar sebagai kebutuhan primer. Fitur yang sudah umum tertanam dan digunakan oleh pengguna ponsel pintar adalah fitur *email.* Pada ponsel berbasis *android* fitur *email* adalah fitur wajib yang ada dan sudah tertanam otomatis bersama dengan sistem operasi *android* itu sendiri. Selain itu pengguna sistem operasi *android* diwajibkan menggunakan minimal 1 (satu) akun email dari *google / gmail* guna mengaktifkan sistem operasi dan fitur - fitur yang ada.

Selain aplikasi *email* bawaan pada sistem operasi *android,* pengguna dapat memasang aplikasi *email* lain yang disediakan oleh penyedia *email* masing - masing. *Yahoo* dan *google* merupakan contoh penyedia *email* yang menyediakan aplikasi *email* yang dapat dipasang pada ponsel *android*. Bagi pengguna *email* yang tidak menyediakan aplikasi yang berbasis *android* tersendiri, biasanya menggunakan aplikasi bawaan yang tertanam dalam ponsel *android.*

Semakin menjamurnya pengguna ponsel *android* maupun perangkat bergerak lain, menjadikan semakin tinggi pula kebutuhan akan *internet. Internet* yang cepat dan murah sangat diharapkan oleh pengguna, penyedia layanan *internet* juga menangkap peluang fenomena yang ada dengan memberikan banyak layanan koneksi nirkabel pada lokasi pusat keramaian. Selain penyedia jasa layanan *internet,* pengusaha yang membutuhkan keramaian orang juga memanfaatkan fenomena ini dengan memasang berbagai layanan *internet hotspot* baik gratis maupun yang berbayar.

Kemudahan berbanding terbalik dengan keamanan. Fenomena menjamurnya layanan *internet hotspot* menjadikan titik - titik *hotspot* maupun pengguna yang memanfaatkan titik tersebut menjadi target kejahatan bagi beberapa orang yang tidak bertanggung jawab.

Beberapa orang yang tidak bertanggung jawab tersebut dengan sedikit kemampuan lebih, mampu mengintip ataupun mencuri data dari pengguna layanan *internet hotspot.* Kemampuan yang dibutuhkan untuk melakukan kejahatan *internet* model seperti ini tidaklah tinggi. Bahkan sudah banyak tutorial - tutorial yang tersebar baik dalam bentuk buku maupun tutorial di *internet* yang dengan mudah dapat diakses oleh semua orang. Data yang diintip maupun dicuri bisa berupa *username,* alamat *email,* bahkan *password.*

2. LANDASAN TEORI

2.1. *Email*

Email atau surat elektronik merupakan layanan pengiriman surat digital yang disediakan oleh *Internet Service Provider* (ISP). ISP menyediakan *server email* atau *mail server* yang berfungsi untuk melakukan pendeteksian pesan dan mengirimkannya pada *email* tujuan. Layanan surat elektronik sendiri terbagi kedalam dua bagian layanan surat elektronik bebas (*free*) dan layanan surat elektronik terbatas. Layanan surat elektronik ini sendiri dapat diakses melaui berbagai cara :

1. *Web Mail*

Merupakan aplikasi berbasis *website* yang disediakan oleh penyedia layanan *email* agar para pengguna dapat dengan mudah mengakses layanan yang diberikan. Pengguna surat elektronik hanya membutuhkan *web browser* serta koneksi *internet* untuk melakukan pengaksesan layanan tersebut.

2. Aplikasi *email Client*

Merupakan aplikasi yang dibuat khusus untuk melakukan pengaksesan layanan surat elektronik. Dengan menggunakan aplikasi *mail client* ini pengguna dapat dengan mudah melakukan manajemen surat elektronik yang dimiliki, bahkan dapat melakukan penulisan surat elektronik

meskipun tidak terdapat koneksi *internet* (Butterfield, Tracy, & Jansen, 2007).

Fungsi yang dilakukan oleh *email client* adalah sebagai berikut (EC Council, 2010) :

a. Mengambil *email* dari kotak surat.
b. Menampilkan *header* dari pesan-pesan yang ada di kotak surat. Pada beberapa kasus, sebagian isi *email* juga ditampilkan.
c. Menulis *email* baru.

Beberapa *email client* yang sering digunakan antara lain : *Mozilla Thunderbird, Microsoft Outlook, Eudora Mail* dan beberapa aplikasi *email client* yang terintegrasi dengan sistem operasi perangkat bergerak.

3. *Email Server*

Email server atau yang biasa disebut dengan *mail server* adalah komputer yang terhubung ke jaringan yang berfungsi sebagai kantor pos virtual, dalam hal komputer yang berfungsi sebagai penyimpan dan penyampai surat elektronik (EC Council, 2010). Proses komunikasi jaringan antara *email client* dan *email server* dapat dilihat pada gambar 2.

Ketika *email client* meminta *email* baru ke *email server*, *email server* meminta *username* dan *password* akun *email.* Setelah *email server* mencocokkan keduanya, *email server* akan mengirimkan *header email* baru ke *email client.* Proses pengiriman *email* sendiri melalui beberapa protokol, yaitu:

1. SMTP

SMTP (*Simple Mail Transfer Protocol*) mekanisme yang digunakan untuk melakukan pengiriman surat elektronik antar *host* dalam jaringan komputer dengan menggunakan TCP / IP (Riabov & College, 2005). SMTP merupakan protokol yang handal dan efisien yang menggunakan *port* 25 untuk operasinya (Jonathan B. Postel, 1982). SMTP melakukan koneksi dengan melakukan pembukaan koneksi melalui SMTP *client* untuk melakukan koneksi ke *server* SMTP, setelah *server* mendapatkan koneksi dari klien SMTP *server* akan mencari keberadaan SMTP *server* tujuan dan mengirimkan surat elektronik tersebut.

2. POP3

POP (*Post Office Protocol*) versi 3 merupakan protokol yang dibuat pada tahun 1984 yang berfungsi untuk melakukan penerimaan surat elektronik. POP sendiri merupakan mekanisme penarikan surat elektronik dari mail *server* ke aplikasi *email* milik pengguna. POP pada dasarnya bekerja mirip dengan kotak surat konvensional. Dengan memanfaatkan protokol POP ini surat elektronik yang berada pada *mail server* akan terhapus. Seperti halnya protokol *email* yang lain POP juga menggunakan perintah dalam operasinya (Butterfield et al., 2007). Perintah yang digunakan dapat dilihat pada gambar 3.

Gambar 2. Proses Komunikasi *Email Client* dan *Email Server.*
(EC Council, 2010)

3. IMAP

IMAP (*Internet* Message Access Protocol) merupakan pengembangan dari protokol POP versi 2. IMAP memiliki fungsi yang sama dengan POP yaitu digunakan untuk melakukan pembacaan surat elektronik. Perbedaan mendasar antara POP versi 3 dan IMAP terletak pada surat yang disimpan, jika POP3 akan menyalin seluruh surat dari *server* dan menyimpannya pada sisi klien, IMAP tidak melakukan penyalinan dan penghapusan surat elektronik dari *mail server* (Butterfield et al., 2007). Perintah - perintah yang dijalankan oleh protokol IMAP dapat dilihat pada gambar 4.

2.2. Keamanan *Email*

Aspek yang penting dalam keamanan *email* adalah, kerahasiaan (*Confidentiality*), keaslian (*Authentication*), integritas (*Integrity*), anti penyangkalan (*Non-repudiation*) (Stallings, 2011). Surat elektronik atau *email* itu sendiri bagaikan surat konvensional, jalur yang dilalui dari pengirim ke penerima sangat panjang melalui beberapa kantor pos cabang, pusat dan dibawa oleh beberapa petugas pengirim surat. *Email* juga demikian, jalur yang dilalui dari pengirim ke penerima melalui beberapa *router*, *mail servers*, dan beberapa jaringan komputer.

Email sangat rentan dengan serangan baik pasif maupun aktif. Contoh ancaman pasif yang mungkin adalah :

1. Pembukaan isi *email*
   Kebanyakan *email* ditransmisikan dalam bentuk jelas (tanpa enkripsi), artinya beberapa orang dengan aplikasi tertentu bisa melihat isi *email.*
2. Analisa lalu lintas data
   Beberapa negara secara rutin memantau isi *email*.

Sedangkan ancaman serangan aktif antara lain sebagai berikut :

1. Modifikasi isi *email*
   Isi *email* dapat dimodifikasi pada saat *transport*asi atau penyimpanan. Selama penyerang ada dalam satu jaringan, penyerang bisa menggunakan *ARP spoofing* untuk mencegat lalu memodifikasi isi *email* ke *mail server* maupun dari *mail server*. Teknik ini yang nantinya akan digunakan untuk pengujian.
2. *Masquerade* (Penyamaran)
   Dimungkinkan untuk mengirim pesan sebagai orang atau organisasi lain.
3. *Spoofing*
   Pesan palsu dapat dimasukkan ke dalam sistem *mail* pengguna lain.
4. *Denial of Service*
   Dimungkinkan untuk membuat *mail server* sibuk dan *overload* sehingga membuat *mail server* tersebut tidak bisa melayani pengguna lain (Toorani, 2008).

Basic Commands from RFC 918	
USER <name>	Set username
PASS <password>	Set password
STAT	Check the status of the mailbox, typically retrieves number of messages
LIST [msg]	List messages in the mailbox; Optional argument for message [msg]
RETR <msg>	Retrieve message <msg>
DELE <msg>	Delete message <msg>
QUIT	Quit
NOOP	No operation
RSET	Reset
Optional Commands from RFC 1939	
TOP <msg> <n>	Retrieve the top <n> lines of message <msg>
UIDL [msg]	Retrieve unique id for [msg]
APOP <name> <digest>	A more robust form of authentication than USER/PASS
Extension Command from RFC 2449	
CAPA	Retrieve a list of capabilities supported by the POP3 server

Gambar 3. Perintah Pada Protokol POP. (Butterfield et al., 2007)

NOOP	Perform no operation
STARTTLS	Establish confidentiality and integrity protection
AUTHENTICATE <type>	Choose authentication method
LOGIN <user> <passwd>	Login with username and password
LOGOUT	Logout the current user
SELECT <mailbox>	Select the desired mailbox to access
EXAMINE <mailbox>	Same as SELECT except opens mailbox for read-only
CREATE <mailbox>	Create a mailbox with the name <mailbox>
DELETE <mailbox>	Delete selected mailbox
RENAME <mailbox> <newmailbox>	Rename mailbox
SUBSCRIBE <mailbox>	Subscribe to selected mailbox
UNSUBSCRIBE <mailbox>	Unsubscribe from selected mailbox
LIST <reference> [pattern]	List contents of current reference based on an optional pattern
LSUB <reference> [pattern]	List a set of mailboxes matching the pattern
STATUS <mailbox> <item>	Show the status of specific items in the selected mailbox
APPEND <mailbox> [flags] <msg>	Append a message to the selected mailbox
CHECK	Perform a checkpoint on the currently selected mailbox
CLOSE	Close the currently selected mailbox
EXPUNGE	Expunge deleted messages from the mailbox
SEARCH <criteria>	Search the mailbox based on certain criteria
FETCH <message> <item>	Fetch the specified item from the selected message
STORE <message> <item> <newvalue>	Update the selected item in a message
COPY <message> <mailbox>	Copy a message to the provided mailbox
UID <command> [args]	Perform an operation on a message based on its UID
CAPABILITY	Query the server for its capabilities

Gambar 4. Perintah Pada Protokol IMAP. (Butterfield et al., 2007)

Guna mengatasi ancaman - ancaman tersebut diatas, maka dikembangkan metode pengamanan *email* yang terenkripsi. Pada dasarnya ada 2 metode dalam enkripsi *email*, yaitu (Stallings, 2011) :

1. *Pretty Good Privacy (PGP)*
   Metode ini merupakan metode yang sangat sering digunakan. PGP dikembangkan oleh Phil Zimmerman dan dirilis pertama kali pada tahun 1991. PGP tersedia baik gratis maupun yang berbayar. PGP mendukung enkripsi dari 5 layanan, yaitu *authentication, confidentiality, compression, e-mail compatibility* dan *segmentation*. Selain 5 layanan itu, PGP juga mendukung *digital signature.*
2. S/MIME (*Secure / Multipurpose Internet Mail Extensions*)
   S/MIME dicetuskan oleh RSA Data *Security* pada tahun 1995. Dalam hal fungsionalitas umum, S/MIME sangat menyerupai PGP. Keduanya menawarkan kemampuan untuk menandatangai dan atau mengenkripsi pesan. Layanan - layanan yang dienkripsi juga sama dengan PGP. Pada sisi penyedia layanan, enkripsi bisa dilakukan dalam komunikasi *client* ke *server* ataupun komunikasi antar mail server. Enkripsi ini biasanya menggunakan standar enkripsi TLS (*Transport Layer Security*). Protokol TLS sendiri mirip dengan protokol *Secure Sockets Layer* (SSL) yang dikombinasikan dengan protokol POP (995), IMAP (993), dan SMTP (465) yang berfungsi untuk enkripsi komunikasi antara aplikasi *email client* dan *email server*.

2.3. Aplikasi *Email* Pada *Android*

Aplikasi *email* pada *android* dibagi menjadi 2 macam, yaitu :

1. Aplikasi *email* bawaan *android*
   Aplikasi *email* bawaan *android* ini terpasang bersama dengan sistem operasi *android.* Pengguna bisa memasang akun *email private* maupun akun *email public* yang disediakan oleh penyedia jasa *email* seperti *yahoo* ataupun *hotmail.* Hanya *email* dari *google* yang tidak dapat dipasang pada aplikasi *email* bawaaan ini dikarenakan aplikasi *email google* juga merupakan aplikasi bawaan sistem operasi *android.* Pada aplikasi *email* bawaan android ini terdapat dua macam mode pengaturan, yaitu pengaturan secara otomatis maupun pengaturan secara manual.
2. Aplikasi *email* dari penyedia jasa *email*.
   Aplikasi *email* dari penyedia jasa email biasanya disediakan oleh penyedia *email public* seperti *yahoo* ataupun *google.* Aplikasi *email* ini hanya bisa memasang *email* yang disediakan oleh penyedia *email* tersebut.

2.4. Aplikasi *Email Gmail*

Salah satu aplikasi *email* yang disediakan oleh *google* ini merupakan aplikasi *email* yang sangat popular digunakan oleh pengguna ponsel *android*. Fitur - fitur yang disediakan antara lain sebagai berikut :

1. Dapat mengatur lebih dari satu *email*.
2. Dapat membaca *email* pada saat *online* maupun *offline*.
3. Dapat membalas *email* saat *offline* lalu otomatis mengirim saat *online*.
4. Notifikasi yang dapat diatur.
5. *Attachment* dapat diihat ataupun langsung *download*.

Dengan beberapa fitur yang ditawarkan serta aplikasi yang bisa didapatkan dengan gratis ini maka banyak pengguna ponsel *android* memasang aplikasi ini pada ponselnya.

2.5. Komunikasi Nirkabel

Komunikasi nirkabel adalah perpindahan data atau informasi dari dua titik atau lebih yang tidak terhubung dengan konduktor listrik. Pemancar nirkabel pertama kali mengudara diawal abad 20 menggunakan *radiotelegraphy* (kode *morse*). Semenjak itu, dimungkinkan untuk memancarkan suara, musik, video melalui jaringan nirkabel (Rouse, 2006).

Teknologi nirkabel yang paling umum digunakan adalah teknologi nirkabel elektromagnetik seperti radio. Dengan digunakannya gelombang radio, penyesuaian jarak perangkat bisa lebih fleksibel. Penggunaan gelombang radio jarak pendek bisa digunakan untuk *remote control tv*, *remote control ac*, dan beberapa perangkat nirkabel rumah tangga yang lain. Penggunaan jaringan nirkabel gelombang radio jarak jauh seperti contoh tv satelit, telepon seluler bahkan alat *gps* sudah banyak diterapkan saat ini.

Kelebihan penggunaan jaringan nirkabel antara lain (EC Council, 2010) :

1. Tidak perlu menarik kabel.
2. Mobilitas perangkat yang tinggi.
3. Pemeliharaan jaringan relatif lebih mudah.
4. Rancangan fleksibel (jarak pendek maupun jauh).
5. Mengikuti perkembangan jaman.

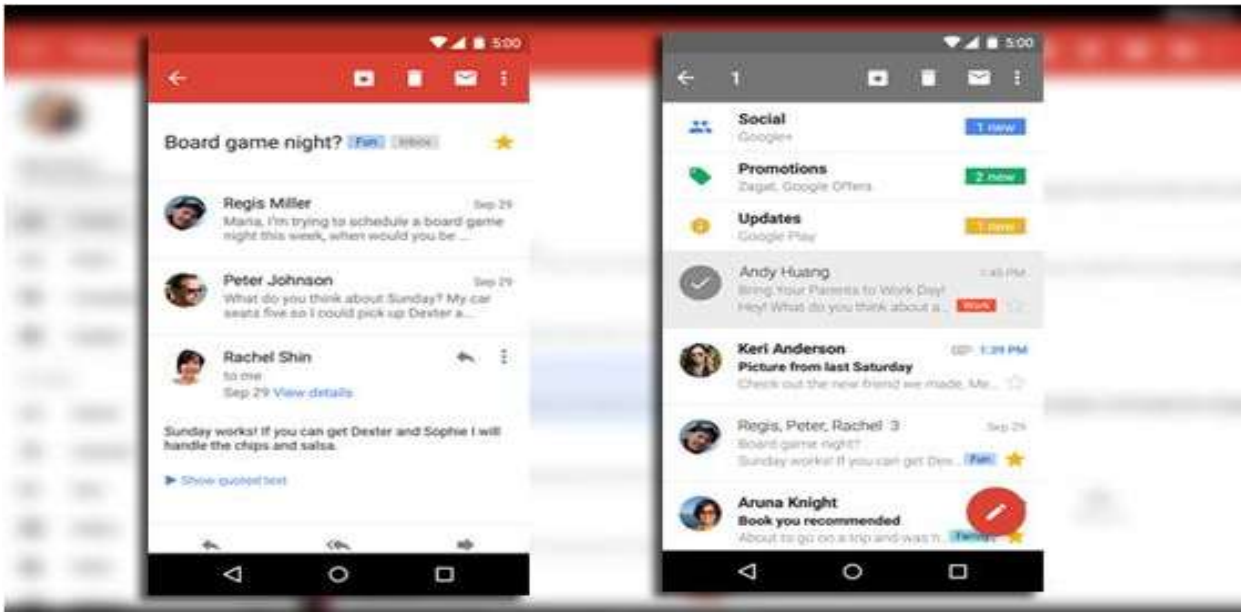
Selain kelebihan komunikasi nirkabel juga memiliki kelemahan, antara lain :

1. Isu keamanan yang masih rentan dibandingkan dengan teknologi kabel.
2. *Bandwidth* yang dibutuhkan lebih besar.
3. Beberapa perangkat elektronik bisa melemahkan sinyal nirkabel.

2.6. ARP *Poisoning* / ARP *Spoofing*

ARP poisoning / ARP spoofing merupakan sebuah serangan yang menyerang transisi antara layer 3 (*layer network*) ke layer 2 (*layer data link*) dalam referensi model OSI . *ARP poisoning* mengubah *MAC address* dari korban yang akan diserang. *ARP poisoning* juga dikenal dengan nama *ARP spoofing*. Teknik *ARP poisoning* sangat efektif digunakan pada jaringan kabel maupun *wireless*. Tujuan serangan dari *ARP poisoning* ini pada umumnya adalah *sniffing*.

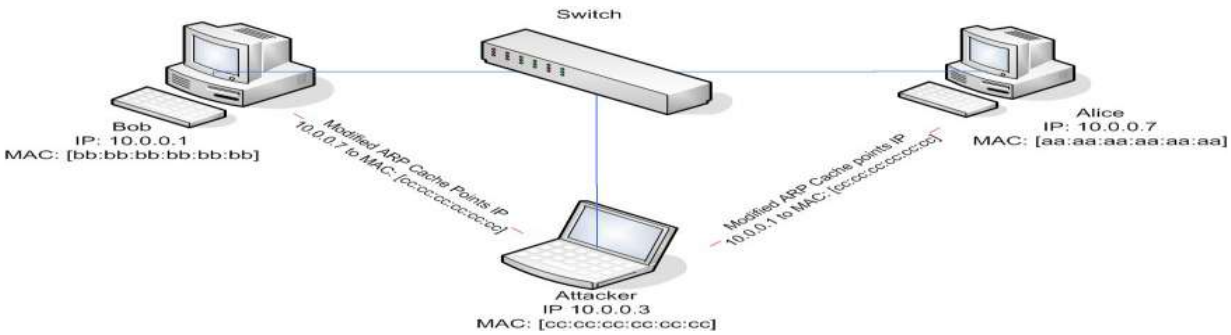
Tipe serangan *ARP poisoning* ini akan membuat penyerang menjadi *gateway* dari sebuah jaringan (EC Council, 2008). Pada saat penyerang melakukan serangan jaringan yang diserang akan menganggap penyerang adalah *gateway* dari jaringan tersebut.



Gambar 5. Aplikasi *Email Gmail* Pada Ponsel Android.

Tabel 1. Informasi Yang Dapat Diambil Dari Serangan *Sniffing* (ECCouncil, 2011)

No	Protokol	Informasi yang Bisa Di dapat
1	TELNET	*Key Stroke*
2	HTTP	Data *Sent in Clear Text*
3	SMTP	*Password* and Data *sent in Clear Text*
4	NNTP	*Password* and Data *sent in Clear Text*
5	POP	*Password* and Data *sent in Clear Text*
6	FTP	*Password* and Data *sent in Clear Text*
7	IMAP	*Password* and Data *sent in Clear Text*
8	SMB	*Data Sent*



Gambar 6. Proses Terjadinya *ARP Poisoning*.
(EC Council, 2008)

2.7. *Sniffing*

Sniffing dalam pengertian berarti mengendus, sedangkan dalam ilmu keamanan jaringan *sniffing* merupakan aktifitas menangkap paket - paket data yang lewat dalam sebuah jaringan (Susanto, 2007). *Sniffing* sendiri biasanya digunakan untuk menangkap informasi - informasi vital dari sebuah jaringan seperti *password*, *email text*, dan *File transfer*. *Sniffing* biasanya menyerang protocol - protokol seperti Telnet, HTTP, POP, IMAP, SMB, FTP, dan lain - lain. Informasi yang didapat dari beberapa protokol di atas dapat dilihat pada tabel 1.

Dalam metode *hacking*, *sniffing* dibagi menjadi dua bagian yaitu *passive sniffing* dan *active sniffing* (Ornaghi & Valleri, 2013).

1. *Passive Sniffing*

Passive sniffing merupakan aktifitas *sniffing* yang dilakukan pada jaringan dengan media penghubung hub. Dimana hub akan melakukan *broadcast* seluruh paket yang melewatinya ke seluruh *node* yang terhubung ke *hub* tersebut. *Hub* merupakan perangkat komputer yang melakukan *broadcast* paket data ke seluruh jaringan sehingga *sniffing* pada jaringan dengan hub sangat mudah dilakukan.

2. *Active Sniffing*

Active sniffing merupakan aktifitas *sniffing* yang dilakukan pada jaringan dengan media penghubung *switch* atau sejenisnya. *Switch* sendiri merupakan sebuah perangkat penghubung (*Concentrator*) yang memiliki *chip* untuk menyimpan tabel *MAC address*. *Switch* tidak lagi mem-*broadcast* paket ke seluruh jaringan namun paket data yang dikirim hanya melalui port asal dan port tujuan saja. Sehingga sangat sulit untuk melakukan *sniffing* pada *switch*. Diperlukan metode khusus untuk melakukan *sniffing* pada *switch*. Untuk melakukan *sniffing* pada jaringan dengan *switch* kita perlu membuat membanjiri media penyimpanan pada *switch* dengan *MAC address* sehingga *switch* tersebut tidak ada bedanya dengan hub. Untuk membanjiri media penyimpanan dapat menggunakan *ARP poisoning* ataupun *MAC Flooding* (Susanto, 2007).

3. HASIL DAN ANALISIS

3.1. Pengujian

Pengujian dilakukan untuk membandingkan tingkat keamanan aplikasi *email* yang otomatis terpasang pada sistem operasi *android* serta aplikasi *email gmail* yang keduanya terhubung dengan jaringan nirkabel. Dalam pengujian ini dilakukan proses *sniffing* dengan digabungkan dengan teknik *ARP spoofing / poisoning*. Skema pengujian dapat dilihat pada gambar 7.

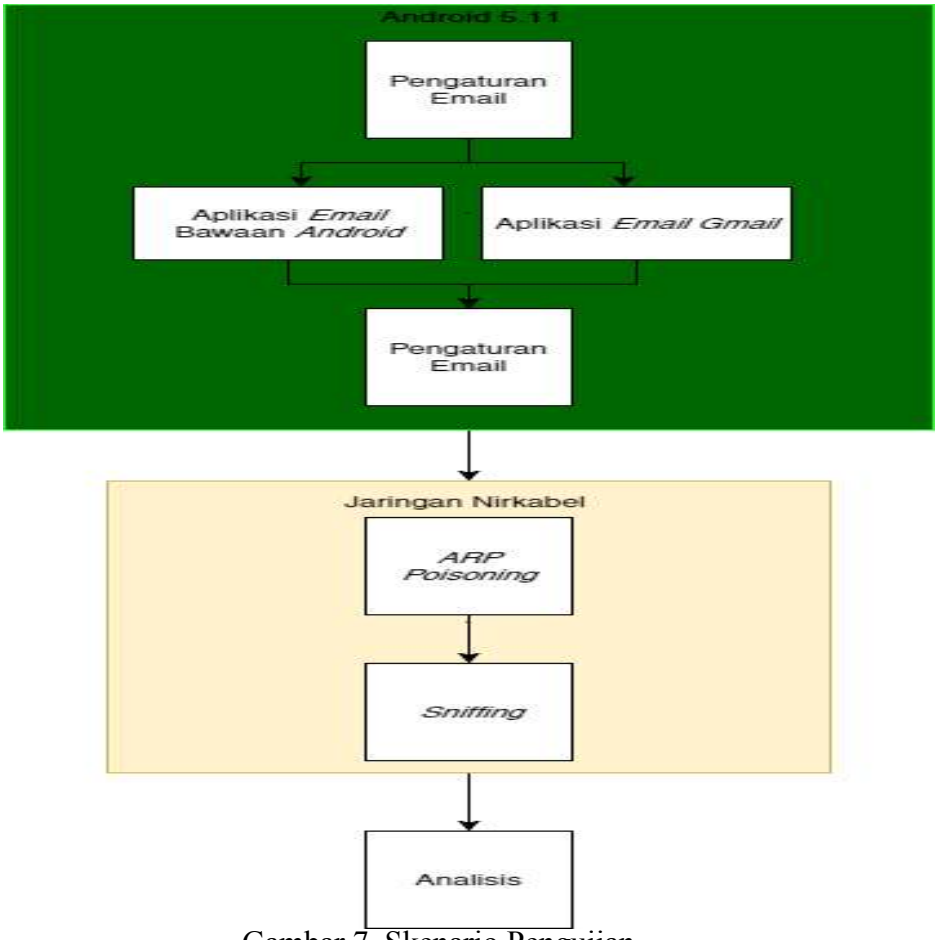
Melakukan *ARP spoofing* atau *ARP poisoning* merupakan suatu kewajiban dalam tujuan untuk melakukan *active sniffing. ARP poisoning* dilakukan pada jaringan yang dipasang untuk skenario pengujian yaitu jaringan dengan akses poin yang difungsikan sekaligus sebagai *gateway router*.

Pengujian pada penelitian ini terdiri dari 3 (tiga) skenario pengujian sebagai berikut :

1. Pengujian pada aplikasi *email* bawaan dengan menggunaan protokol POP (*Post Office Protocol*).
2. Pengujian pada aplikasi *email* bawaan dengan menggunakan protokol IMAP (*Internet Message Access Protocol*).
3. Pengujian pada aplikasi *email gmail*.

Aplikasi *ettercap* digunakan untuk melakukan proses *ARP Poisoning. Ettercap* merupakan *tool* yang dibangun untuk melakukan proses *sniffing* pada jaringan (Ornaghi & Valleri, 2013).

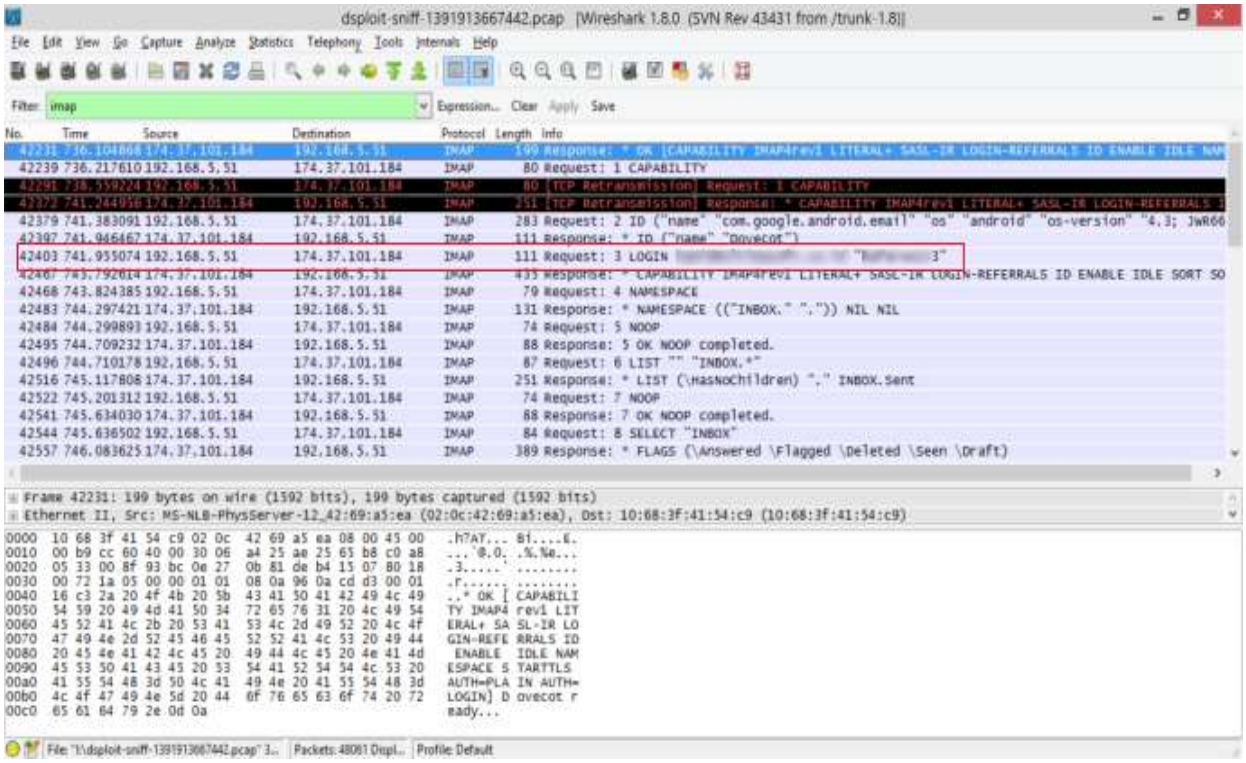
Meskipun aplikasi *ettercap* dapat melakukan *ARP poisoning* sekaligus *sniffing,* namun aplikasi *wireshark* digunakan dalam melakukan proses *sniffing. Wireshark* sendiri merupakan aplikasi yang digunakan aplikasi yang dapat digunakan untuk melakukan *sniffing* sekaligus analisa hasilnya (Wireshark, 2013).



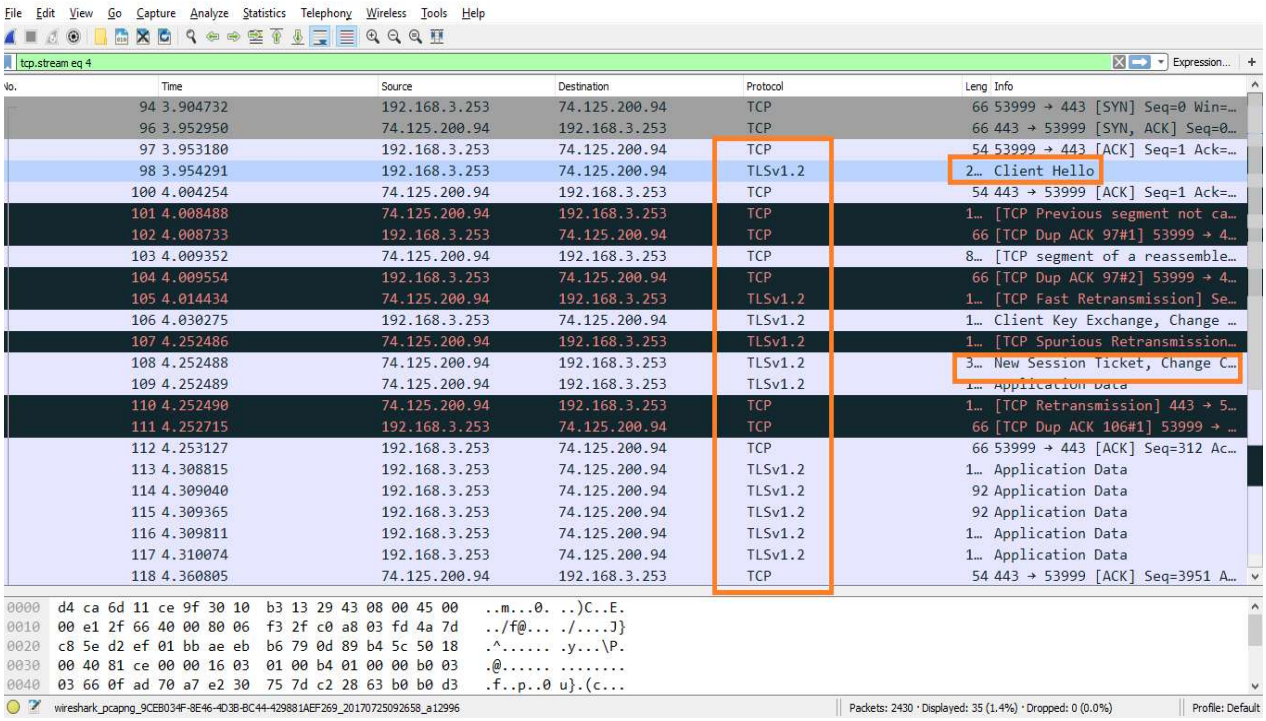
Gambar 7. Skenario Pengujian.

Tabel 2. Hasil Pengujian Protokol *Email*

PROTOKOL	DATA OTENTIKASI
POP	*CLEAR TEXT*
IMAP	*CLEAR TEXT*
POP TLS	*ENCRYPTED*
IMAP TLS	*ENCRYPTED*



Gambar 8. Komunikasi Data Aplikasi *Email* Bawaan *Android* Protokol IMAP.



Gambar 9. Komunikasi Data Aplikasi *Email Gmail*.

Setelah *ARP poisoning* berhasil dilakukan, komunikasi datang pengguna yang menggunakan jaringan nirkabel yang sama akan terekam. Contoh hasil *sniffing* dapat dilihat pada gambar 8.

Gambar diatas menampilkan komunikasi data aplikasi *email* bawaan dengan menggunakan protokol IMAP. Gambar tersebut memperlihatkan bahwa pengujian ke-2 menghasilkan komunikasi data dengan *username* dan *password* dapat dibaca dalam bentuk *cleartext* tanpa enkripsi apapun. Komunikasi data tersebut juga menunjukkan bahwa proses *login* dan *listing email* berhasil.

1. Pengujian pada aplikasi e*mail gmail.*

Hasil *sniffing* pada aplikasi *email gmail* dapat dilihat pada gambar 9. Terlihat pada gambar bahwa proses *login email* di enkripsi, yang terlihat hanya informasi bahwa proses *login* berhasil.

Dari hasil uji yang dilakukan pada 3 (tiga) skenario dengan langkah - langkah sebelumnya maka didapat hasil pada tabulasi pengujian. Hasil tabulasi dapat dilihat pada Tabel 2.

3.2. Analisis Hasil Uji

Pengujian dengan 3 (tiga) skenario telah dijalankan lalu pengujian serta analisa dilakukan lebih lanjut dengan hasil sebagai berikut :

1. Komunikasi data pada pengujian aplikasi *email* bawaan dengan menggunakan opsi protokol POP menunjukkan bahwa *username* dan *password* dapat dilihat secara *cleartext.* Pengujian juga dilakukan dengan menunjukkan proses memasukan *username* dan *password* yang benar sehingga *login* berhasil dengan *username* dan *password* tersebut. Hal ini juga berarti bahwa *username* dan *password* yang terekam juga dapat dibuktikan kebenarannya.
2. Komunikasi data pada pengujian aplikasi *email* bawaan dengan menggunakan opsi protokol IMAP menunjukkan bahwa *username* dan *password* dapat diilihat secara *cleartext.* Pengujian juga dilakukan dengan menunjukkan proses memasukan *username* dan *password* yang benar sehingga *login* berhasil dengan *username* dan *password* tersebut. Hal ini juga berarti bahwa *username* dan *password* yang terekam juga dapat dibuktikan kebenarannya.
3. Komunikasi data pada pengujian aplikasi *email gmail* menghasilkan *username* dan *password* yang terenskripsi. Komunikasi data yang dapat dilihat hanya proses *login* ke *server* serta proses *login* berhasil.

Pada pengujian ke-1 dan pengujian ke-2 digunakan 2 (dua) *private email*, sedangkan pada pengujian ke-3 digunakan *google mail.* Dari pengujian tersebut juga ditemukan bahwa :

1. Dua *private email* yang digunakan untuk pengujian mendukung penggunaan protokol POP dan IMAP.
2. Dua *private email* yang digunakan untuk pengujian tidak mendukung protokol POP dengan *security* dan IMAP dengan *security* (untuk pengaturan lebih lanjut).
3. *Google mail* pada pengujian (3) tidak perlu melakukan konfigurasi lebih lanjut dan otomatis menggunakan protokol TLS.

Walaupun aplikasi *email gmail* tidak dapat dilihat datanya, namun hal ini paling tidak membuktikan bahwa serangan *ARP poisoning / spoofing* sangat mungkin dilakukan pada jaringan nirkabel.

4. KESIMPULAN

Setelah dilakukan pengujian dan analisis dapat ditarik beberapa kesimpulan, sebagai berikut :

1. Salah satu serangan yang harus diwaspadai dalam jaringan nirkabel adalah serangan *ARP spoofing / poisoning.* Serangan ini bisa digunakan untuk mendukung serangan lain seperti *sniffing.*
2. Secara *default* saat melakukan konfigurasi aplikasi *email* bawaan *android* secara otomatis akun *email* menggunakan protokol POP ataupun IMAP sesuai dengan penyedia. Konfigurasi otomatis ini tanpa pengaturan *security* (TLS).
3. Pengaturan konfigurasi aplikasi *email* bawaan *android* secara *default* ataupun otomatis mengakibatkan *username* dan *password* dapat dibaca secara *cleartext.*
4. TLS sudah dipublikasikan sebagai kesepakatan *internet* protokol pada tahun 1999 (Chandrataruna & Ngazis, 2013), namun masih banyak *private email* tidak menyertakan protokol TLS untuk tambahan keamanan pada protokol POP dan IMAP. Ini terjadi dikarenakan mahalnya infrastruktur maupun investasi yang harus disediakan untuk pengadaaan protokol TLS ini. *Google* sendiri mulai menggunakan protokol TLS pada POP dan IMAP pada tahun 2004, dan menggunakan TLS pada SMTP pada tahun 2011 (Chandrataruna & Ngazis, 2013).
5. Pencegahan untuk menghindari penyadapan data *email* dapat dilakukan dengan cara sebagai berikut :

    a. Melakukan konfigurasi *advanced* / lanjut apabila menggunakan aplikasi *email* bawaan *android* sehingga dapat menambahkan protokol TLS. Hal ini tentu saja dapat dilakukan dengan catatan *server private email* tersebut mendukung protokol keamanan TLS.
    b. Untuk *private email* yang tidak menyediakan ataupun mendukung protokol keamanan TLS, bisa dipertimbangkan 2 (dua) alternatif berikut :

        - Menggunakan fitur *forwarding email google*, yang merupakan satu - satunya penyedia layanan *email* yang menggunakan protokol TLS pada penerimaan maupun pengiriman *email* (Chandrataruna & Ngazis, 2013).
        - Menggunakan aplikasi *email gmail* untuk *private email. Gmail* mendukung penggunaan akun *email* selain akun *gmail* itu sendiri untuk ditambahkan dalam aplikasi *email gmail*.

Standard operating procedure (SOP) juga diperlukan dalam mengantisipasi serangan *ARP spoofing / poisoning* ini. SOP ini berlaku baik untuk pengguna ataupun penyedia layanan jaringan nirkabel. Antara lain sebagai berikut :

1. Bagi pengguna layanan jaringan nirkabel, sebagai berikut :

    a. Tidak sembarangan membuka koneksi ke jaringan nirkabel yang belum dikenal.
    b. Memastikan *anti virus* ataupun *firewall* selalu dalam pembaruan terbaru.

2. Bagi penyedia layanan jaringan nirkabel, sebagai berikut :

    a. Pemasangan *router* atau akses poin yang mendukung *anti ARP spoofing / poisoning*.
    b. Menindak pelaku kejahatan komputer dalam wewenang jaringan penyedia.
    c. *Update* berkala *firmware* akses poin ataupun *router*.

DAFTAR PUSTAKA

Alliance, O. H., 2007. Industry Leaders Announce Open Platform for Mobile Devices. Retrieved January 02, 2014, from *http://www.openhandsetalliance.com/press_110507.html*.

Butterfield, J., Tracy, M., & Jansen, W., 2007. Guidelines on Electronic Mail Security Recommendations of the National Institute of Standards and Technology.

Chandrataruna, M., & Ngazis, A. N., 2013. Mengapa Badan Keamanan AS Bisa Sadap E-mail? Retrieved March 01, 2014, from *http://m.news.viva.co.id/news/read/423514-mengapa-badan-keamanan-as-bisa-sadap-e-mail.*

EC Council., 2008. Ethical Hacking and Countermeasures Module XIII Hacking Email Accounts News.

EC Council., 2010a. CHFI v8 Module 17 Investigating Wireless Attacks.pdf.

EC Council., 2010b. CHFI v8 Module 19 Tracking Emails and Investigating Email Crimes.pdf.

EC Council., 2011. CHFI v8 Module 17 Investigating Wireless Attacks.pdf. ECCouncil.

Einhorn, B., 2012. Indonesians Still Love Their BlackBerrys - Businessweek. Retrieved January 04, 2014, from *http://www.businessweek.com/articles/2012-12-06/indonesians-still-love-their-blackberrys*.

Jonathan B. Postel., 1982. SIMPLE MAIL TRANSFER PROTOCOL. *RFC*, *90291*(August).

Meulen, R. van der., 2013. Gartner Says Smartphone Sales Accounted for 55 Percent of Overall Mobile Phone Sales in Third Quarter of 2013. Retrieved January 03, 2014, from *http://www.gartner.com/newsroom/id/2623415*.

Ornaghi, A., & Valleri, M., 2013. Ettercap. Retrieved March 23, 2013, *from http://ettercap.github.io/ettercap/*.

Riabov, V. V, & College, R., 2005. SMTP (*Simple Mail Transfer Protocol*).

Ricknäs, M., 2011. Android Becomes Best - selling Smartphone OS, Says Canalys PC World. Retrieved January 04, 2014, *from http ://www.pcworld.com/article/218219/android_becomes_best_selling_smartphone_os_says_canalys.html*.

Rouse, M., 2006. Definition of Wireless. Retrieved March 27, 2013, *from http://www.techtarget.com/*.

Stallings, W., 2011. *Network Security Essentials : Applications And Standards* (4th ed.).

Susanto., 2007. *Seni Teknik Hacking 2* (Edisi Dua.). Jakarta: Jasakom.

Toorani, M., 2008. SMEmail - A New Protocol for the Secure E-mail in Mobile Environments. *2008 Australasian Telecommunication Networks and Applications Conference*, 39–44. doi:10.1109 / ATNAC. 2008. 4783292.