Overview Keamanan Dalam Komunikasi D2D (*Device-to-Device*) Pada Jaringan Komunikasi Bergerak Nirkabel

Tugas Akhir Mata Kuliah Keamanan Informasi dan Jaringan
EL5241

Bambang Supriadi
(23214330)

Dosen :
Ir. Budi Rahardjo, MSc., PhD

Magister Teknik Elektro
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
2015

Abstrak

D2D (*Device-to-Device*) merupakan bentuk komunikasi yang memungkinkan antar UE (*User Equipment*) berkomunikasi secara langsung dengan atau tanpa supervisi dari eNB (*evolved Node B*). D2D merupakan teknologi yang dikenalkan melalui 3GPP Rel. 12 untuk LTE. Dan teknologi ini menjadi teknologi yang menjanjikan dimasa depan yang akan diadopsi oleh sistem 5G karena dapat meningkatkan *latency* dan kapasitas *bandwidth*. Akan tetapi fakta bahwa kanal *wireless* (nirkabel) dianggap sebagai kanal yang rentan terhadap berbagai macam serangan, maka aspek keamanan merupakan salah satu hal yang menjadi perhatian utama dalam komunikasi *Device-to-Device* (D2D). Urgensi untuk membentuk suatu protokol dan mekanisme komunikasi yang *secure* (aman) sangat diperlukan. Oleh sebab itu telah dilakukan beberapa kajian oleh para peneliti untuk membangun suatu komunikasi D2D yang aman. Beberapa kajian tersebut telah menghasilkan algoritma dan solusi untuk mengatasi masalah keamanan komunikasi D2D. Namun demikian belum ada yang diadopsi menjadi standar resmi untuk keamanan teknologi D2D yang baru muncul ini. Dalam makalah ini dibahas bagaimana suatu bentuk komunikasi yang aman dalam D2D dengan menggunakan suatu protokol yang mengharuskan *authentication* antara kedua entitas UE. Kemudian juga dibahas metode dalam mengatasi masalah *eavesdropper*, yaitu seorang yang ikut mendengarkan percakapan orang lain, pada kanal *relay* D2D. Bahasan berikutnya adalah terkait serangan *Denial-of-Service* (DoS) pada komunikasi D2D.

Kata Kunci : *D2D (Device-to-Device) communication, security, Denial-of-Service (DoS), LTE*

Daftar Isi

Daftar Gambar

Daftar Tabel

Overview Keamanan Dalam Komunikasi D2D (*Device-to-Device*) Pada Jaringan Komunikasi Bergerak Nirkabel

1 Pendahuluan

Dewasa ini telah kita rasakan bagaimana suatu teknologi nirkabel telah dan akan terus berevolusi untuk memberikan akses konektivitas tanpa batas yang pada gilirannya memberikan kemudahan pada umat manusia. Dari awalnya yang hanya menyalurkan komunikasi analog berupa suara, kemudian menuju komunikasi suara digital dan hingga saat ini berupa teknologi data *broadband* berkecepatan tinggi. 4G merupakan generasi teknologi nirkabel berbasis seluler terkini yang dikenal dengan jargonnya "*The Future of Mobile Broadband*". 4G harus bisa mencapai kapasitas *bandwidth* sebesar 1000 Mbps sebelum menuju transisi ke sistem 5G. Kapasitas sebesar itu dibutuhkan untuk mengatasi dampak pertumbuhan pengguna *mobile* (bergerak) yang terus tumbuh. Hasil laporan *Cisco Visual Network Index* (VNI) [1] menyatakan bahwa trafik data *mobile* tumbuh 69% pada tahun 2014. Traffik data *mobile* mencapai 2.5 exabyte per bulan pada akhir tahun 2014. Total data trafik *mobile* tahun 2014 hampir 30 kali dari total data trafik internet global tahun 2000.

LTE-A (*Long Term Evolution Advanced*) merupakan teknologi *mobile broadband* yang menjadi basis sistem 4G. 3GPP(*Third Generation Partnership Project*) yang merupakan lembaga standardisasi teknologi *mobile*, seperti LTE memperkenalkan suatu teknologi yaitu D2D (*Device-to-Device*) pada *3GPP Release 12* [2]. Teknologi ini memungkinkan UE(*User Equipment*) untuk berkomunikasi langsung dengan atau tanpa supervisi dari eNB (*evolved Node B*). Teknologi ini muncul untuk meningkatkan kapasitas *bandwidth* dan *latency*. Sehingga D2D merupakan bagian dari sistem 4G untuk mencapai kapasitas *bandwidth* dan mengatasi masalah pertumbuhan data trafik *mobile* seperti disampaikan pada paragraf sebelumnya.

Kanal *wireless* (nirkabel) dianggap sebagai koneksi yang rentan terhadap serangan. Hal ini didasarkan pada sifatnya yang *broadcast* sehingga siapapun atau perangkat apapun dapat membangun koneksi ke perangkat pemancar dan melakukan serangan. Dalam [3], para penulisnya menyampaikan bahwa dibandingkan koneksi UE langsung ke BTS/eNB, koneksi langsung antar perangkat *proximity* (D2D) lebih rentan karena : (1) keterbatasan kapasitas komputasi dari perangkat *mobile* untuk komputasi terkait keamanan; (2) manajemen keamanan *autonomous* (swatantra) seperti *mutual authentication*; (3) struktur transmisi *relay*. Maka disamping kelebihan yang diberikan terdapat kekurangan pada teknologi D2D yaitu terkait aspek keamanan. Aspek ini harus sudah ada solusinya sebelum teknologi ini dimplementasikan secara luas.

2 Bentuk Komunikasi D2D (*Device-to-Device*)

Bentuk komunikasi D2D (*Device-to-Device*) sudah ada sebelum teknologi LTE untuk standar sistem 4G digunakan. Beberapa contoh teknologi pendukung untuk memungkinkan komunikasi nirkabel *device-to-device* adalah: *infrared, bluetooth dan WiFi*. Teknologi tersebut telah mendukung komunikasi nirkabel *device-to-device* lokal pada jarak yang pendek. Namun untuk dapat digunakan dalam mendukung layanan ProSe (*Proximity-Based Services*), istilah yang digunakan 3GPP Release 12 untuk D2D, masih terdapat kekurangan. Dalam [4], penulisnya menyampaikan beberapa kekurangan tersebut adalah sebagai berikut :

- *unlicensed spectrum*. *WiFi* dan *Bluetooth* beroperasi pada *unlicensed spectrum*, tanpa adanya kendali. Hal ini tidak menjadi masalah ketika kepadatan pengguna rendah, namun akan menjadi masalah ketika layanan ProSe berkembang. *Throughput*, jangkauan dan kehandalan semuanya akan menderita.

- *manual pairing*. *WiFi* dan *Bluetooth* tergantung pada *manual pairing* untuk memungkinkan komunikasi antar perangkat. Hal ini akan menjadi batu sandungan untuk layanan *autonomous* (swatantra) D2D.

- *Security*. Fitur keamanan pada *WiFi* dan *Bluetooth* tidak begitu aman sehingga tidak akan cocok untuk layanan aplikasi keamanan publik.

- *independence from cellular networks*. *WiFi* dan *Bluetooth* beroperasi secara *independent* tanpa menggunakan teknologi seluler, seperti LTE. Bentuk komunikasi *device-to-device discovery* seperti itu berjalan paralel dengan komunikasi radio seluler, sehingga tidak efisien dan menguras penggunaan baterai.

2.1 Komponen *Device-to-Device*

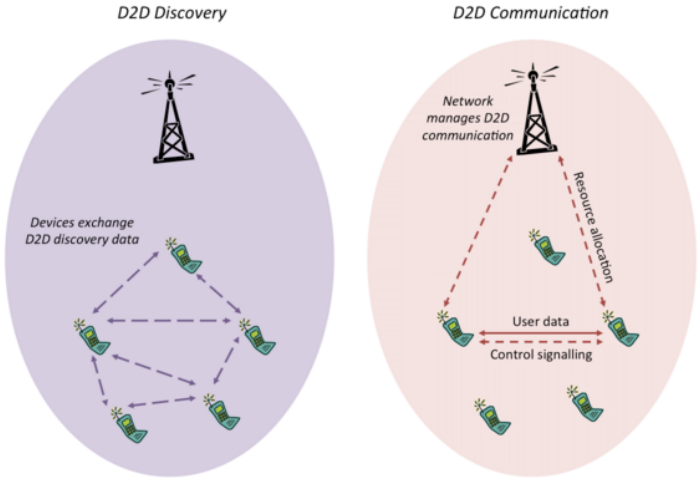
Dalam membentuk suatu komunikasi *device-to-device* terdapat dua komponen utama. Dalam [4], dua komponen tersebut (lihat Gambar 1), dinyatakan sebagai berikut :

- *D2D Discovery*, memungkinkan perangkat bergerak untuk menggunakan antarmuka radio LTE (*LTE licensed spectrum*) untuk menemukan (*discover*) kehadiran perangkat D2D lainnya.

- *D2D Communication* adalah fasilitas perangkat D2D untuk menggunakan antarmuka radio LTE (*LTE licensed spectrum*) tanpa melakukan *routing* trafik ke eNB.

Untuk layanan publik umum, D2D akan tersedia hanya ketika UE berada dalam jangkauan jaringan telekomunikasi seluler, seperti LTE. Dengan demikian jaringan LTE dapat melakukan kendali untuk *radio resource sharing* dan keamanan. Namun dalam kasus khusus terkait *public safety* (PS)/keselamatan

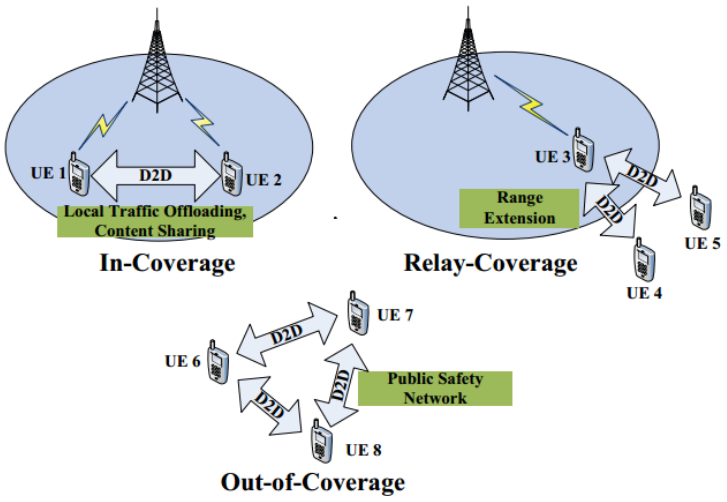
publik, contohnya saat terjadinya bencana yang merusak jaringan telekomunikasi seluler atau serangan teroris, maka layanan D2D dapat beroperasi tanpa kehadiran jaringan seluler.



Gambar 1: Komponen D2D. Sumber: [4]

2.2 Skenario *Device-to-Device*

Dari dua komponen *device-to-device* yaitu *D2D discovery* dan *D2D communication* dapat dibentuk suatu skenario yang menempatkan teknologi *device-to-device* berada pada jaringan seluler maupun di luar jaringan seluler, seperti: jaringan keamanan nasional dan keselamatan publik. Dengan demikian teknologi *device-to-device* harus mampu bekerja pada kedua jaringan tersebut dengan sangat baik. [3] mengkategorikan skenario komunikasi *device-to-device* kedalam tiga tipe representatif berdasarkan keikutsertaan dari beberapa entitas jaringan seperti jaringan seluler dan tipe utilisasi sumber daya spektrum, seperti diilustrasikan pada Gambar 2.



Gambar 2: Skenario dan *use cases* D2D. Sumber: [3]

Sesuai dengan yang disampaikan oleh [3] ketiga tipe representatif skenario *device-to-device* adalah sebagai berikut :

- *In-Coverage*: dalam skenario ini, UE berada dalam jangkauan jaringan telekomunikasi seluler. Dengan skenario ini perangkat-perangkat *device-to-device* dikendalikan oleh entitas jaringan, seperti eNB. Operator mengendalikan *authentication*, pembangunan koneksi, alokasi sumber daya dan manajemen keamanan. Dalam skenario ini koneksi antar *device-to-device* menggunakan media komunikasi radio yang sama dengan yang digunakan oleh entitas jaringan (*licensed spectrum* dari jaringan seluler). Skenario ini dapat menjadi solusi dari jumlah trafik yang besar dan kerapatan pengguna UE yang padat, sebagaimana disampaikan dalam laporan Cisco Visual Network Index (VNI) [1]. Jumlah data dan kerapatan pengguna yang besar membebani jaringan inti (*core network*), dengan skenario ini operator dapat melakukan *offloading* sumberdaya frekuensinya ke jaringan lokal *device-to-device* sehingga *core network* tidak terbebani oleh trafik yang padat. Contoh penggunaanya adalah pada *local content sharing* dan komunikasi *machine-to-machine* (M2M).

- *Relay Coverage*: ketika perangkat pengguna (mis: UE4 dan UE5, lihat Gambar 2) berada di luar jangkauan dari jaringan seluler, seperti eNB, maka UE4 dan UE5 dapat berkomunikasi dengan eNB melalui eNB3 yang berada dalam jangkauan eNB. Kontrol penuh pada perangkat *device-to-device* dilakukan oleh eNB sama seperti skenario *In-Coverage* begitu juga dengan *link* yang digunakan antar *device-to-device* menggunakan *licensed spectrum* dari eNB.

- *Out-of-Coverage*: Skenarioi ini berlaku jika jaringan seluler tidak ada. Hal ini terjadi pada kondisi darurat seperti adanya bencana alam yang menghancurkan infrastruktur jaringan seluler. Perangkat *device-to-device* mis UE6, UE7 dan UE8 seperti pada gambar 2, dapat membentuk suatu koneksi dan melakukan komunikasi *device-to-device*. Skenario ini dapat dimanfaatkan pada layanan perlindungan publik, penanggulangan bencana, keamanan nasional dan keselamatan publik. Skenario ini terlihat mirip dengan *Mobile Ad-hoc Network* (MANET). Namun perbedaan keduanya terletak pada penggunaan media radio untuk membentuk *link* koneksi. MANET menggunakan frekuensi ISM (*Industrial, Scientific and Medical*) sementara D2D menggunakan *reserved cellular licensed* (frekuensi radio terlisensi milik operator).

2.3 Keuntungan *Device-to-Device*

Dari skenario representatif D2D pada subbagian 2.2 sebelumnya, sebenarnya dapat dilihat beberapa keuntungan D2D seperti mengurangi beban trafik pada jaringan inti, meningkatkan jangkauan (*coverage*) karena kemampuannya untuk membentuk suatu *relay* dan dapat digunakan dalam *public safety* (keselamatan publik).

Ericson Research, suatu lembaga penelitian milik Ericson, telah mengidentifikasi dan mengevaluasi *potential gains* (keuntungan yang diperoleh) dari D2D [5]. Keuntungan yang diperoleh adalah sebagai berikut :

- Peningkatan Kapasitas: hal ini bisa diperoleh karena adanya penggunaan bersama sumber daya frekuensi antara seluler dan D2D.
- Peningkatan *Data Rate*: terkait dengan *proximity* yang dekat diharapkan dapat menaikkan *data rate*
- Peningkatan *latency* : ketika perangkat-perangkat berkomunikasi melalui *direct link*, *delay* dari ujung ke ujung dapat dikurangi.

D2D diinisiasi pada 3GPP *Release* 12, teknologi ini dianggap sebagai salah satu teknologi yang menjanjikan disamping teknologi lainnya yang mendukung komunikasi seluler masa depan (seperti : SDN, MIMO dll). *International Telecommunication Union* (ITU) melalui grup pembahasan *Working Party 5D* (WP5D) yang diwakili oleh setiap negara anggotanya telah memasukkan D2D sebagai teknologi untuk meningkatkan skenario *mobile broadband* di masa mendatang. WP5D bertanggungjawab untuk menciptakan regulasi dan *roadmap* untuk IMT2020, yaitu sistem 5G yang akan digelar pada tahun 2020 [6].

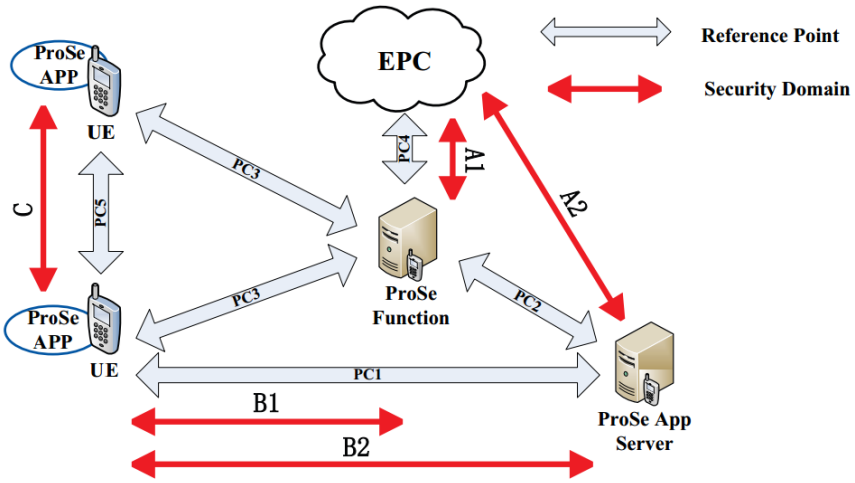
Dengan keuntungan yang dapat diperoleh dari D2D, maka keseluruhan tujuan teknis dalam sistem 5G seperti yang telah dikonsep oleh METIS dalam [7] diharapkan dapat dicapai dengan optimal. Disamping keuntungan tersebut, namun terdapat satu aspek penting yang belum benar-benar menjadi perhatian yaitu aspek keamanan komunikasi D2D. Pada bagian berikutnya akan dibahas beberapa aspek teknis yang sudah dikaji oleh para peneliti.

3 Persyaratan Keamanan *Device-to-Device*

Dalam [3], para penulisnya mengajukan sebuah arsitektur keamanan dalam *framework* D2D. Berdasarkan arsitektur keamanan inilah nantinya akan dibahas persyaratan keamanan yang utama untuk teknologi dan aplikasi D2D. Berdasarkan hal tersebut pembahasan mengenenai persyaratan keamanan dalam D2D ini akan dibahas menjadi dua sub bagian.

3.1 Arsitektur Keamanan D2D

Arsitektur komunikasi D2D telah didefinisikan oleh 3GPP dalam [8]. Dalam arsitektur tersebut terdapat dua komponen fungsionalitas yaitu *ProSe Function* dan *ProSe App Server* dan lima koneksi penghubung (PC1, PC2, PC3, PC4 dan PC5) baik antara UE dengan *ProSe Function/ProSe App Server* maupun antar *ProSe Function* dan *ProSe App Server* atau antar UE. Berdasarkan arsitektur *Proximity-Based Services* ini lalu [3] mengembangkan arsitektur keamanan seperti yang terlihat pada Gambar 3.



Gambar 3: Arsitektur Keamanan Komunikasi D2D. Sumber: [3]

Wang dkk. mengajukan tiga *domain* keamanan pada jaringan komunikasi D2D seperti terlihat pada gambar 3. Ketiga *domain* tersebut adalah sebagai berikut :

- Keamanan D2D antara jaringan 3GPP dan *ProSe Function/ProSe App Server*. Dibagi menjadi dua bagian yaitu (A1) Keamanan D2D antara jaringan 3GPP dan server *ProSe Function*, yang menangani keamanan PC4; (A2) Keamanan D2D antara jaringan 3GPP dan D2D *ProSe App Server*, yang menangani keamanan PC2 dan PC4.

- Keamanan D2D antara perangkat D2D dan *ProSE Function/App Server*. Dibagi menjadi dua juga yaitu : (B1) Keamanan D2D antara perangkat D2D dan server *ProSe Function*, yang menangani keamanan pada PC3; (B2) Keamanan D2D antara perangkat D2D dan *Prose App Server*, yang bekerjasama menangani keamanan terkait PC3 dan PC2.

- Keamanan antar perangkat D2D. Merupakan keamanan untuk menangani koneksi PC5.

3.2 Persyaratan Keamanan D2D

Untuk menjadikan komunikasi D2D aman maka diperlukan suatu persyaratan. Persyaratan teknis keamanan ini yang akan membentengi koneksi D2D melalui media nirkabel, menjadi lebih aman. Dengan demikian pengguna tidak merasa khawatir saat melakukan komunikasi D2D. Kekhawatiran mengenai pencurian data, pengungkapan *privasi* dan jenis serangan lainnya dapat diminimalisasi jika komunikasi D2D telah memenuhi persyaratan keamanan yang distandarkan. Namun sampai dengan saat ini belum ada persyaratan keamanan standar untuk komunikasi D2D. [3] merangkum beberapa syarat teknis tersebut sebagai berikut :

- *Confidentiality and Integrity (C/I)*: Hal ini untuk memastikan tidak terjadi perubahan data dan kebocoran saat dilakukan transmisi data.

- *Authentication (Au)* : adalah bentuk *access control* untuk mengkonfirmasi perangkat yang membangun hubungan adalah perangkat sebenarnya, bukan perangkat penyerang seperti: MITMA (*man-in-the-middle attack*).
- *Privacy (Pr)*; Data pribadi seperti identitas, lokasi dan data personal lainnya harus dirahasiakan dari pihak yang berwenang. Dengan demikian D2D harus mampu menghindari kebocoran data pribadi ke pihak lain.
- *Non-Repudiation (NR)*: Mekanisme ini sebenarnya untuk memastikan data yang dikirim dari A ke B dapat dibuktikan kebenarannya bahwa data itu dikirim dari A dan diterima B tanpa penolakan.
- *Revocability (Re)* : Digunakan untuk mencabut hak pengguna layanan D2D jika pengguna terdeteksi sebagai pengguna yang berbahaya.
- *Availability and Dependability (A/D)*: Layanan D2D harus selalu tersedia. Beberapa serangan seperti DoS dapat mematikan layanan, oleh sebab itu komunikasi D2D harus dapat mengantisipasi serangan seperti ini.

4 Keamanan Pada Komunikasi D2D

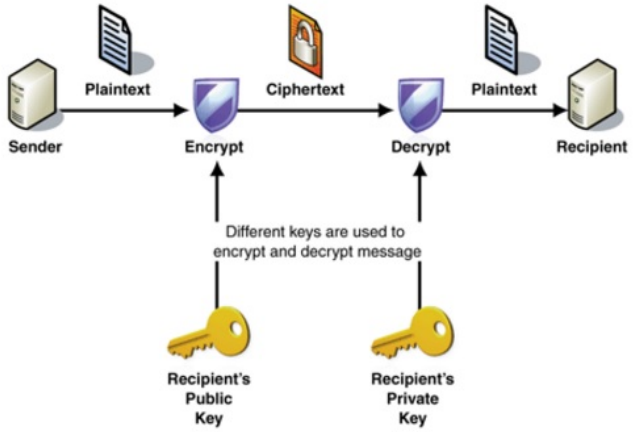
Komunikasi D2D membawa manfaat yang besar seperti telah disampaikan bada bagian 2. Efisiensi penggunaan spektrum frekuensi radio sebagai sumber daya yang terbatas dan peningkatan *data rate* serta *latency* merupakan hal yang mendasari teknologi ini menjadi bagian dari teknologi *mobile* di masa mendatang. Namun keuntungan tersebut tidak akan pernah diperoleh jika protokol komunikasi D2D itu sendiri tidak aman. Pengguna akan merasa khawatir dalam penggunaan teknologi ini jika aspek keamanannya tidak dikaji secara mendalam. Berbagai kejahatan di dunia internet seperti pencurian data, pengungkapan data pribadi, penyebaran aplikasi dan layanan yang berbahaya bagi perangkat dan data yang tersimpan di perangkat merupakan beberapa kekhawatiran yang akan dialami pengguna D2D.

Security dan *availability* merupakan salah satu aspek penting suatu teknologi layak digunakan secara massal. Keamanan memastikan koneksi atau teknologi yang kita gunakan tidak membahayakan dan ketersedian (*availability*) memastikan layanan tersebut tetap dapat bertahan dan berjalan meskipun sedang melawan *attacker*. Sistem yang *intermittent* tentu akan ditinggalkan oleh penggunanya.

4.1 Mekanisme Pembentukan dan Pertukaran Kunci

Mekanisme pembentukan kunci dan pertukran kunci diperlukan dalam membangun suatu komunikasi yang aman. Ini merupakan bagian dari persyaratan teknis yang dijabarkan dalam sub bagian 3.2 yaitu *authentication*. Dan suatu mekanisme *non-repudiation* diperlukan untuk mengantisipasi *man-in-the-middle attack* (MITA). MITA dimungkinkan karena pertukaran kunci publik dilakukan melalui kanal publik *insecure*. Mekanisme kunci publik diperlihatkan

pada gambar 4. Kunci publik dapat diketahui oleh siapa saja namun kunci pribadi dipegang oleh pengguna dan bersifat rahasia. Pengirim mengirim pesan (mis:*plaintext*) dan dienkripsi menggunakan kunci publik penerima, pesan *plaintext* terenkripsi (*chipertext*) dikirim dan sampai di penerima didekripsi menggunakan *private key* sehingga diperoleh pesan *plaintext* kembali.



Gambar 4: Kunci Public, Enkripsi dan Dekripsi Data.
Sumber: Microsoft, available at:
https://msdn.microsoft.com/en-us/library/ff647097.aspx

Salah satu algoritma *public key* adalah *"Diffie-Hellman cryptosystem"* yang merupakan sistem *public key* yang termasuk sudah tua namun masih digunakan sampai sekarang. Cara kerja protokol *key agreement* Diffie-Hellman seperti yang dipaparkan dalam [9] adalah sebagai berikut: anggap p dan q secara publik dikenal oleh perangkat A dan B (jika tidak, A dapat mengirim kunci publiknya ke B), A dan B keduanya secara *random* menghasilkan sebuah nilai a dan b. A menghitung g^a mod p dan mengirimnya ke B, sebaliknya B menghitung g^b mod p dan mengirimnya ke A. Kemudian A menghitung $s{=}(g^b)^a$ mod p dan B menghitung $s{=}(g^a)^b$ mod p. Kedua entitas A dan B akan memperoleh nilai s yang sama. $(g^a)^b$ mod p akan digunakan sebagai *shared secret* antara A dan B. Implementasi dari Diffie-Hellman akan memerlukan kapasitas komputasi yang cukup besar karena p, a dan b dapat berupa angka yang besar. Namun kapasitas komputasi dari perangkat bergerak (*mobile device*) saat ini sudah mumpuni untuk melakukan komputasi tersebut.

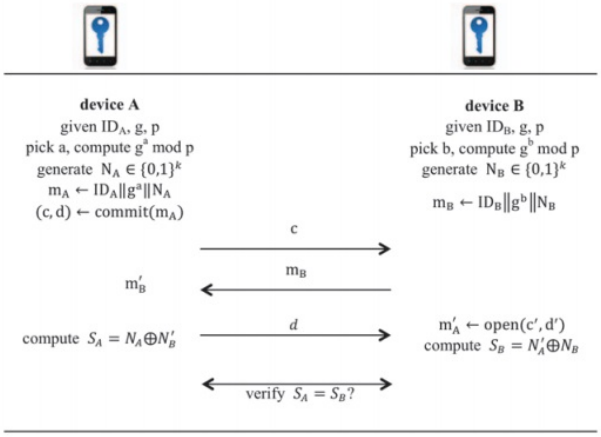
Dalam [9], para penulisnya juga menyampaikan kelemahan Diffie-Hellman terhadap MITMA. Hal ini karena g^a dan g^b ditransmisikan melalui kanal publik, tidak ada cara bagi perangkat A untuk mengetahui secara pasti apakah g^b berasal dari perangkat B dan sebaliknya. Perangkat A akan berbagi suatu *shared secret* dengan siapa saja yang mentransmisikan g^b, meskipun itu bukan dari perangkat B. Alasan utama MITMA dapat bekerja adalah tidak adanya *mutual authentication* antara dua perangkat ini. Salah satu solusi untuk mengatasi hal ini adalah kedua perangkat melakukan *hash secret key* nya (mis. menggunakan MD5) untuk menghasilkan nilai h(K), lalu melakuan komparasi nilai *hash* melalui kanal terpercaya. Jika *mutual authentication* cocok maka kedua

perangkat dapat menerbitkan suatu *shared secret key*. Ini merupakan solusi *Non-Repudiation*.

Mutual authentication yang disampaikan sebelumnya ternyata memiliki masalah terkait jumlah bit yang harus dicek oleh pengguna terlalu besar [9]. Keluaran dari fungsi *hash* bisa mencapai lebih dari 128 bits (32 hexadecimal digits), dan pengecekan secara visual dan verbal terhadap hal tersebut adalah tugas yang tidak sepele. Pemotongan terhadap kode *hash* bisa mengurangi jumlah digit untuk dicek, namun memunculkan kelemahan keamanan. 32 bit kode *hash* yang dipotong dapat dipecahkan dalam waktu kurang dari 1 detik.

4.1.1 Usulan Perbaikan Protokol Kunci Keamanan

Dalam [9], para penulisnya menyampaikan perbaikan protokol untuk pertukaran kunci Diffie-Hellman. Skenario penelitian mereka adalah dua perangkat *mobile* ingin membangun *shared secret key* untuk komunikasi D2D. Kedua perangkat tidak berbagi informasi kriptografi sebelumnya dan tidak ada pihak ketiga terpercaya (mis. lembaga CA dan RA). *mutually authentication* dilakukan secara visual atau verbal.



Gambar 5: Protokol Pertukaran Kunci. Sumber: [9]

Dalam protokol ini diperkenalkan adanya skema komitmen (*commitment schemes*). Salah satu *user* dapat memilih satu nilai komitmen dan menjaganya tetap tersembunyi *hidden* hingga sampai saatnya untuk dibuka. Alur pengiriman skema komitmen dapat dilihat pada gambar 5. Algoritma dari skema komitmen adalah sebagai berikut:
Commit. $(c,d) \leftarrow m$ mentransformasikan nilai m ke dalam pasangan *commitment/open* (c,d). nilai *commit* c tidak dapat mengungkap m kecuali digabungkan dengan nilai *open* d. Sehingga (c,d) dapat mengungkap nilai m.
Open $(c,d) \leftarrow m$ menghasilkan keluaran nilai m jika (c,d) dihasilkan oleh Commit(m).

Keseluruhan protokol dibangun berdasarkan protokol *key agreement* Diffie-Hellman dan skema komitmen. Di luar protokol perangkat A dan B menghasilkan k-bit *random string* N_A dan N_B, $N_A \oplus N_B$ merupakan simbol untuk

mutual authentication atara A dan B. Alur dari protokol dapat dilihat pada gambar 5.

Pada tahap inisial, A dan B memilih parameter Diffie-Hellman a dan b, lalu menghitung g^a dan g^b. A dan B secara acak menghasilkan k-bit *string* N_A dan N_B. $m_A = \mathrm{ID}_A \parallel g^a \parallel \mathrm{N}_A$ dan $m_B = \mathrm{ID}_B \parallel g^b \parallel \mathrm{N}_B$ dibentuk dengan rangkaian (*concatenation*), dimana ID_A dan ID_B adalah identitas dari pengguna A dan B berupa nama dan alamat *email*. A juga perlu menghitung *commitment/opening* (c,d) untuk $m_A = \mathrm{ID}_A \parallel g^a \parallel \mathrm{N}_A$.

Pertukaran data dilakukan oleh pengguna A dan B melalui kanal komunikasi D2D. Pengguna A mengirim c (nilai komitmen dari m_A ke pengguna B; setelah menerima c, pengguna B mengirim m_B ke pengguna A. Sebagai balasan, pengguna A mengirim nilai *decommit* d ke pengguna B. Pengguna B membuka *commitment* dan memperoleh $m_A = \mathrm{ID}_A \parallel g^a \parallel \mathrm{N}_A$.

Pada tahap akhir, pengguna A dan B menghasilkan k-bit *authentication string* $S_A = \mathrm{N}_A \oplus \mathrm{N}'_B$ dan $S_B = \mathrm{N}'_A \oplus \mathrm{N}_B$, dimana N'_A dan N'_B diperoleh dari pesan yang diterima oleh A dan B. Lalu pengguna A dan B melakukan verifikasi apakah $S_A = S_B$ melalui kanal terpercaya (perbandingan visual atau verbal). Jika *authentication string* cocok, A dan B menerima parameter Diffie-Helman masing-masing dan *shared secret key* $K = g^a \bmod p$. Jika *authentication string* tidak cocok kedua belah pihak bisa membatalkan komputasi *secret key generation*.

Perbaikan lainnya dari algoritma *Diffie-Hellman Key Exchange* (DHKE) disampaikan oleh Zhang dkk dalam [10], Zhang dkk menamakan protokolnya dengan SeDS (*Secure Data Sharing Strategy*). Inisialisasi dari sistemnya adalah sebagai berikut :

1. Menghasilkan parameter sistem
   Otoritas terpercaya eNB, diberikan parameter keamanan k, menghasilkan *tuple* $(q,g,g_1, \mathbb{G}, \mathbb{G}_T, \hat{e})$ dengan menjalankan *Gen(k)*. Lalu eNB memilih satu algoritma enkripsi simetrik $Enc_s()$ dan dua fungsi *hash* H_0 dan H_1, dimana H_0:{0,1}* $\rightarrow \mathbb{Z}_q^*$, H_1:{0,1}* $\rightarrow \mathbb{G}$.
   Parameter sistem, $params = (q, g, g_1, \mathbb{G}, \mathbb{G}_T, \hat{e}, Enc_s(), H_0, H_1)$

2. Registrasi *Service Provider* (SP)
   eNB menghitung $PID_0 = H_0(RID_0)$ sebagai identitas *pseudo* untuk SP. Lalu secara acak memilih sebuah integer $x_0 \in \mathbb{Z}_q^*$ sebagai kunci pribadi dan $X_0 = g^{x_0}$ sebagai kunci publik. Terakhir eNB mengirim pasangan kunci publik/pribadi $(X - 0, x_0)$ melalui kanal yang aman.

3. Registrasi *User Equipment* (UE)
   Ketika UE_i mendaftar ke sistem dengan identitas asli RID_i, eNB menetapkan $PID_i = H_0(RID_i)$ sebagai identitas *pseudo*. Lalu secara acak memilih integer $x_i \in \mathbb{Z}_q^*$ sebagai kunci *private* dan menghitung kunci publik untuk UE_i melalui $X_i = g^{xi}$. Kunci publik/*private* (X_i, x_i) dikirim ke UE_i melalui kanal aman (*secure*). Sementara itu, *tuple* (X_0, PID_0) juga dikirim ke *register*

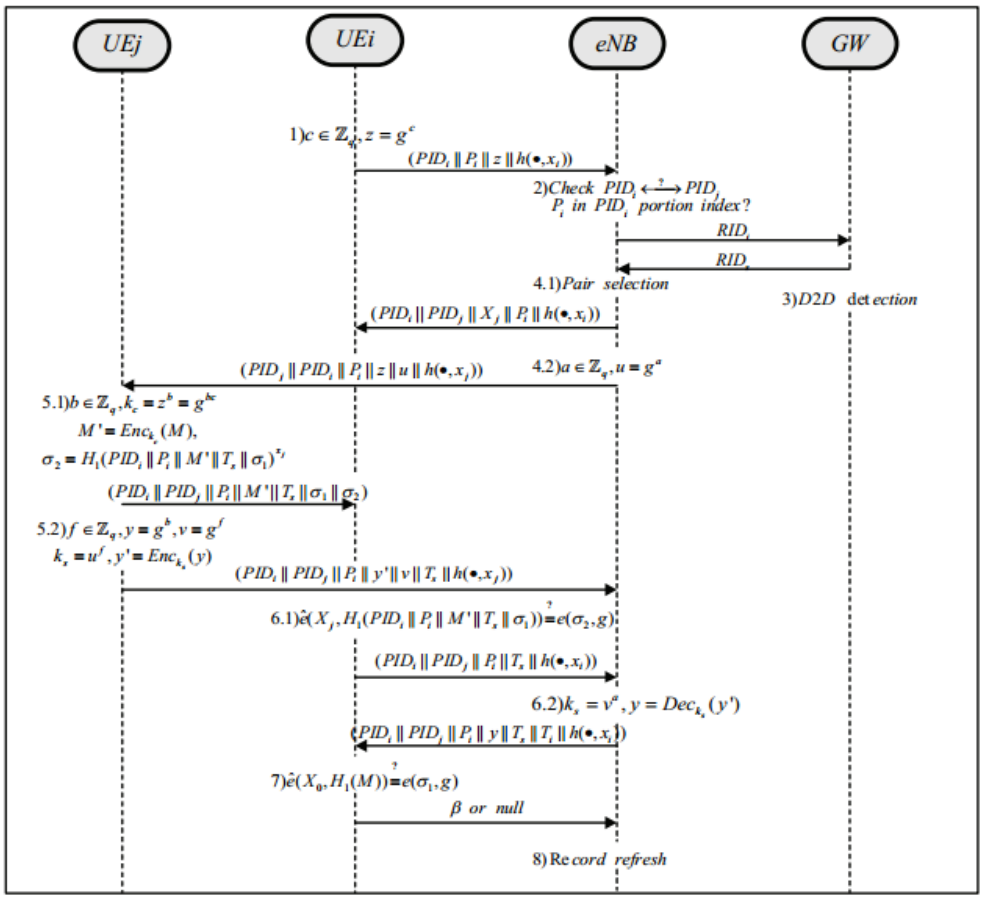
Table 1: Member yang tercatat di eNB.

RID	PID	Pubilic key	Portion index (P_i)	Share frequency	Malicious behavior amount
RID_0	PID_0	X_0	$0, 1, 2, \cdots p$	0	0
RID_1	PID_1	X_1	0	0	0
RID_2	PID_2	X_2	0	0	0
...	...	...	...	...	...

Sumber: [10]

4. Pembangunan Sistem
   eNB terus mencatat status untuk entitas dan frekuensi tempat berbagi data dengan pasangannya sebagaimana diperlihatkan pada tabel 1.

Protokol *secure data sharing* bekerja berdasarkan langkah-langkah berikut. Langkah-langkah tersebut dapat dilihat pada gambar 6.



Gambar 6: Protokol SeDS (*Secure Data Sharing).* Sumber: [10]

Langkah 1 *Service request* (permintaan layanan). Sebuah UE (anggap UE_i), secara acak memilih $c \in \mathbb{Z}_q^*$ dan menghitung nilai $z = g^c$ untuk menghasilkan kunci komunikasi k_c.

Langkah 2 *Atuthentication.* Selama menerima permintaan pesan, eNB pada awalnya memverifikasi integritas dengan melakukan komputasi terhadap nilai

Table 2: Format data selama pengiriman.

PID_i	PID_j	Portin index (P_i)	$Enc(M)$ (M')	Timestamp (T_s)	Signature (σ_1)	Signature (σ_2)
2 Bytes	2 Bytes	2 Bytes	L Bytes	2 Bytes	20 Bytes	20 Bytes

Sumber: [10]

hash dari pesan, dan autentikasi pemohon dengan mode komunikasi seluler normal. Diperoleh nilai RID_i (contohnya, nomor kartu SIM). Lalu merujuk ke tabel 1 untuk melakukan pengecekan apakah PID_i yang dikirim UE_i dan RID_i adalah pemetaan satu-ke-satu.

Langkah 3 *Candidate detection* (deteksi kandidat). *Proximity Service Control Function* (PSCF) pada GW melakukan deteksi *proximity service* dan mencari pasangan D2D potensial untuk UE.

Langkah 4 *Pair selection* (pemilihan pasangan). eNB memilih kandidat yang tepat (anggap UE_j). Lalu eNB secara acak memilih $a \in \mathbb{Z}_q^*$ dan menghitung $u = g^a$ sebagai kunci petunjuk. Komunikasi pesan permintaan $(PID_j \| PID_i \| z \| u \| P_i \| h(\bullet, x_j))$ dikirim ke entitas terpilih. Secara bersamaan eNB menjawab permintan UE dengan *pseudo-identity* PID_j dan kunci publik X_j dari pengirim, dengan kata lain $(PID_i \| PID_j \| X_j \| P_i \| \ h(\bullet, x_i))$ dikirim ke UE_i.

Langah 5 *Data transmission* (transmisi data). Ketika menerima komunikasi permintaan pesan

$$PID_j \| PID_i \| z \| u \| P_i \| h(\bullet, x_j)$$

, entitas secara acak memilih $b \in \mathbb{Z}_q^*$ dan menghasilkan kunci komunikasi $k_c = z^b = g^{cb}$. Dengan k_c entitas yang mengenkripsi material M dan diperoleh $M\prime = Enc_{k_c}(M)$. Sebelum mengirim pesan M, entitas menandai pesan dengan menghitung

$$\sigma_2 = H_1(PID_j \| P_i \| M' \| T_s | \sigma_1)^{xj}$$

.Data tersebut dibentuk dalam format seperti tabel 2 dan dikirim ke UE tujuan.

Langkah 6 *Entity verification* (verifikasi entitas) ketika suatu paket diterima, UE_i mengektraksi PID_j dari pesan. PID_j dibandingkan dengan *pseudo-identity* yang diperoleh dari eNB. Jika keduanya tidak cocok, paket dibuang. Sebaliknya jika cocok maka dilakukan verifikasi tandatangan dengan melakukan pengecekan

$$\hat{e}(X_j, H_1(LID_j \| P_i \| M' \| T_s \| \sigma_1) \stackrel{?}{=} \hat{e}(\sigma_2, g).$$

Untuk mendekrip pesan M', UE_i mengirim pesan permintaan kunci petunjuk berupa $(PID_i \| PID_j \| P_i \| T_s \| h(\bullet, x_i))$ kepada eNB.

Ketika pesan permintaan kunci petunjuk sampai , eNB mengecek jika informasi waktu dari pesan masih dalam waktu yang diperbolehkan *window*. Jika iya, lakukan dekripsi $Enc_{k_s}(y)$ dengan $k_s = v^a$ dan sebuah jawaban dengan dikirimnya $(PID_i \| PID_j \| P_i \| y \| T_s \| T_i \| h(\bullet, x_i))$. *Timestamp* T_i digunakan untuk merekam waktu umpan balik yang dianalisa kemudian pada langkah 8.

Langkah 7 Data verfication (verifikasi data). Dengan diterimanya kunci petunjuk y, UE_i dapat kunci komunikasi dengan melakukan komputasi $k_c =$

$y^c = g^b c$. Sehingga *payload* M' didekripsikan dan material asal M terungkap. Untuk memastikan otoritas data, tanda tangan diverifikasi dengan pengecekan

$$\hat{e}(X_0, H_1(P_i|M)) \stackrel{?}{=} \hat{e}(\sigma_1, g)$$

Jika persamaan ditahan, data diterima. Sebaliknya, maka kemungkinan terjadi serangan. Lalu UE_i melaporkan *beacon*

$$\beta = (PID_i \| PID_j \| P_i \| M' \| T_s \| \sigma_1 \| \sigma_2 \| h(\bullet, x_i))$$

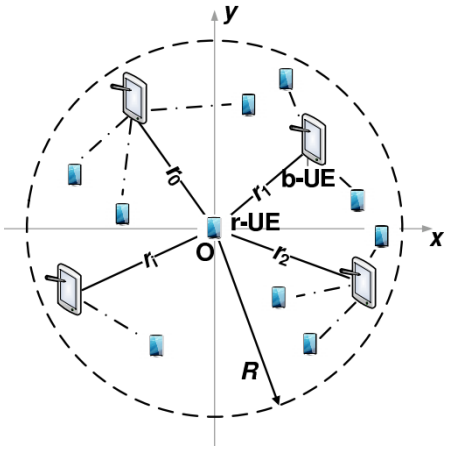
ke eNB dalam *timestamp* T_i', yang memenuhi $T_i' < T_i + \Delta T$ (ΔT, skala waktu yang didefinisikan sebelumnya)

Langkah 8 Record refresh. eNB menunggu ΔT setelah mengirim jawaban kunci petunjuk ke UE_i.

4.1.2 Kunci Keamanan Untuk Aplikasi Keselamatan Umum

Teknologi D2D didesain untuk terintegrasi dengan sistem seluler, namun dalam kasus tertentu seperti bencana alam yag menghancurkan infrastruktur jaringan telekomunikasi, D2D dapat beroperasi tanpa kehadiran layanan jaringan telekomunikasi seluler.

Kapabilitasnya yang mampu membangun layanan komunikasi tanpa kehadiran jaringan telekomunikasi membuat teknologi ini berguna dalam layanan keselamatan umum (*public safety*). Disaat beberapa UE tidak mendapat layanan dari jaringan seluler seperti eNB maka salah satu dari UE bertindak sebagai *beacon broadcaster* yang kemudian diterima oleh beberapa UE yang lainnya. Mekanisme ini dijelaskan oleh Gorratti dkk dalam [11]. Gambaran *Secure Direct-beacons Broadcasting* digambarkan dalam gambar 7.



Gambar 7: Kemungkinan pemasangan *beaconing* dan *regular UE*. Sumber: [11]

Mekanisme *Secure D-beacons Broadcasting* adalah b-UE melakukan *broadcasting D-beacons*, lalu *regular UE* dapat memulai prosedur asosiasi dengan jaringan D2D. Untuk membangun komunikasi D2D yang aman, autentikasi dan keamanan didasarkan pada protokol enkripsi pertukaran kunci yang terindeks

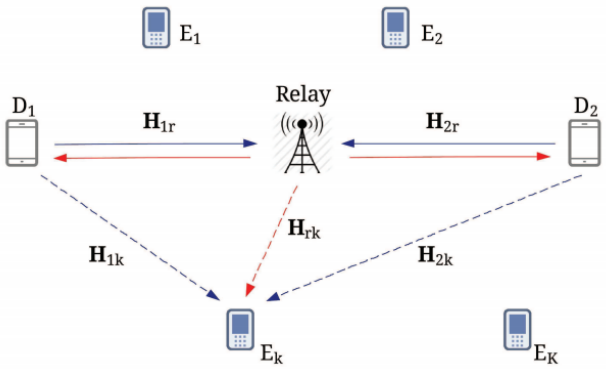
pada *D-beacon*. Pendekatan ini bertujuan untuk mengurangi jumlah pertukran paket yang dapat menyebabkan kenaikan hasil agregat interferensi melalui kanal nirkabel.

Untuk menghindari akibat buruk inteferensi dan sekaligus menyediakan protokol D2D yang aman, maka Gorratti dkk mengajukan mekanisme dimana hanya *r-UEs* yang bisa berkomunikasi dengan *b-UE*. Inti dari mekanisme tergantung pada skema enkripsi acak *key pre-distribution*. Ide ini sebenarnya sudah digunakan pada jaringan sensor (*sensor network*). Karena alasan kapasitas memori sensor yang terlalu kecil, nilai acak *pre-distribution* dari sejumlah relatif kunci diambil dari sejumlah besar kunci terjamin yang tahan terhadap *hacking*.

4.2 Keamanan D2D pada Lapisan Fisik

Keamanan pada lapisan fisik dibutuhkan untuk menghindari *eavesdropper* pada komunikasi D2D. Menghindari *eavesdropper* merupakan bentuk pemenuhan persyaratan keamanan D2D (seperti dijabarkan pada bagian 3.2) yaitu pada poin *confidentiality and integrity*. Beberapa kajian keamanan pada lapisan fisik untuk mengatasi *eavesdropper* ini, telah dilakukan oleh para peneliti. Beberapa diantaranya adalah dengan (1) menyediakan alokasi transmisi daya optimal untuk memaksimalkan kerahasiaan; (2) kualitas layanan (QoS) yang berorientasi skema keamanan *beamforming* optimal; (3) masalah rasio kerahasiaan maksimum terkait keberadaan beberapa *eavesdropper* menggunakan *imperfect channel state information* (CSI).

Para penulis dalam [12], melakukan kajian keamanan pada lapisan fisik komunikasi D2D berbasiskan *physical layer network coding* (PNC). Dalam komunikasi D2D berbasiskan PNC, kedua perangkat mengirimkan simbol-simbol selama tahap *multiple access* (MA). Ketika *eavesdropper* berusaha meng-*intercept* (menangkap) satu simbol, komunikasi mengalamai interferensi/gangguan dari yang lainnya. Selama tahap *broadcasting* (BC), *relay* mengirimkan simbol XOR yang merupakan simbol terenkripsi.



Gambar 8: Komunikasi D2D menggunakan *relay* pada lingkungan yang tidak aman. Sumber: [12].

Berdasarkan gambar 8, terdapat dua perangkat yang sedang berkomunikasi melalui sebuah *relay* yang terpercaya. *Relay* melakukan pemetaan PNC, dan

komunikasi dua arah dari perangkat D_1 dan D_2 disediakan dengan menggunakan mode *time division duplex* (TDD). Beberapa *eavesdropper* (K) berusaha menangkap informasi dari komunikasi D2D. Semua simpul dilengkapi dengan beberapa antenna. N_i, N_r dan N_k masing-masingnya adalah jumlah antenna dari perangkat ke-i (dinotasikan D_i dengan i = 1,2), *relay* dan *eavesdropper* ke-k (dinotasikan E_k dengan k=1,2...,K).

Ada dua tahapan pada proses komunikasi D2D berdasarkan gambar 8 tersebut, yaitu:

- tahap *multiple access* (MA)
  Selama tahap MA, kedua perangkat mengirimkan informasi modulasi $s_i \in \mathbb{C}$ di-*beamform* oleh $\mathbf{w}_i \in \mathbb{C}^{N_i \times 1}$ (i=12). *Relay* memperkirakan jumlah dari dua simbol yang ditransmisikan setelah melewati vektor *beamforming* $\mathbf{w}_r \in \mathbb{C}^{N_r \times 1}$

- tahap *broadcasting* (BC)
  Selama dalam slot waktu ke dua, *relay* memancarkan (*broadcast*) s_r di-*beamformed* oleh $\mathbf{v}_r \in \mathbb{C}^{N_r \times 1}$. D_i menghitung s_r setelah melalui vektor *beamforming* $\mathbf{v}_i \in \mathbb{C}^{N_i \times 1}$ (i=1,2).

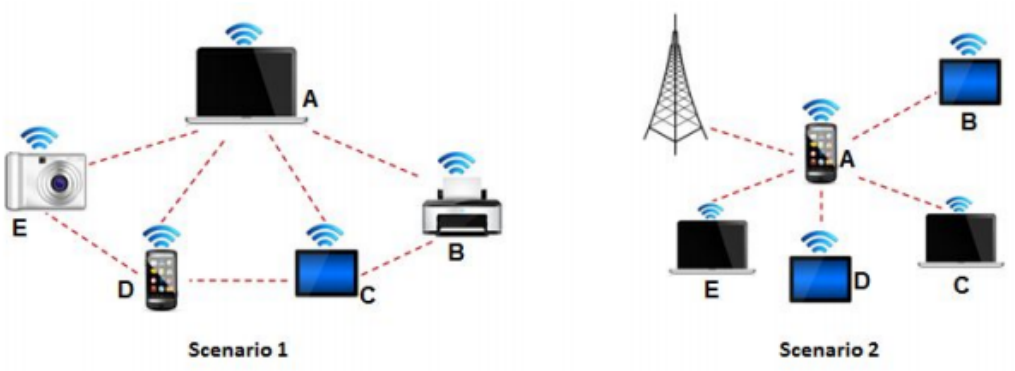
Vektor *beamforming* didesain untuk menyediakan kerahasian pada komunikasi D2D. Keamanan lapisan fisik selama tahap MA lebih penting untuk mencegah adanya *eavesdropping*, karena setidaknya hanya dibutuhkan satu pesan dari s_1 atau s_2 untuk mendekodekan s_r. Dan penyempurnaan kerahasian dengan desain *beamforming* dilakukan selama tahap BC.

Dalam [12], para penulisnya mengajukan dua buah algoritma untuk mengamankan komunikasi D2D melalui suatu *relay* yaitu: algoritma 1 *secure beamforming design for MA stage* dan algoritma 2 *secure beamforming design for BC stage*. Performa keamanan komunikasi D2D diperbaiki dengan meminimalkan *mean square error* (MSE) pada *relay*, baik pada tahap MA maupun BC dan dengan mempertimbangkan kendala pada SINR untuk mencegah *eavesdropping*. *Channel state information* (CSI) pada kanal-kanal *device-to-eavesdropper* dan *relay-to-eavesdropper imperfect* pada perangkat dan *relay*. *Error* pada CSI diasumsikan mengikuti model *Gauss Markov uncertainty*. Dengan demikian kedua algoritma tersebut diajukan dengan memperhitungkan sifat kesalahan Gaussian (*Gaussion nature of error*). Kedua algoritma tersebut digunakan untuk menganalisa performa komunikasi D2D dan menyelidiki distribusi SINR pada *eavesdropper*.

4.3 Serangan DoS pada D2D

Denial-of-service (DoS) merupakan serangan ke suatu komputer atau *host* yang terhubung ke suatu jaringan. Serangan ini menghabiskan *resource* atau sumber daya dari perangkat yang diserang. Dengan habisnya sumber daya yang dimiliki komputer/*server* yang diserang, maka komputer/*server* tidak bisa melayani pengguna lain. D2D merupakan komunikasi yang sangat memungkinkan menerima serangan DoS.

Dalam [13], para penulis mempelajari bagaimana kemungkinan serangan DoS maupun DDoS (*Distributed* DoS) pada komunikasi D2D. Namun dalam eksperimennya para penulis menggunakan *legacy D2D* bukan D2D yang memanfaatkan spektrum frekuensi dari jaringan seluler, spt LTE. Dalam eksperimennya, mereka menggunakan *WiFi Direct* dengan skenario yang ditampilkan pada gambar 9.



Gambar 9: Skenario *WiFi Direct D2D*. Sumber: [13]

Eksperimen dilakukan menggunakan "Skenario 2" sesuai gambar 9. Adapun skenario serangan dibagi dua bagian besar yaitu serangan DoS dan DDoS, dengan rincian sebagai berikut:

1. Serangan DoS
   Skenario A. Serangan dilakukan menggunakan perangkat B. Serangan ditujukan ke perangkat *server* pada jaringan Internet. Efek pada *server* Internet tidak membawa dampak yang signifikan, hal ini karena kapasitas pita yang lebih besar pada *server* Internet. Namun jika dilihat trafik *upload* pada perangkat A hampir seluruhnya terpakai. Sehingga jika prioritas akses ada pada perangkat A, maka perangkat C,D dan E akan mengalami gangguan akses ke *server* Internet. Sebaliknya jika A tidak memiliki prioritas maka A akan mengalami penurunan kecepatan akses ke *server* Internet.

   Skenario B. Pada skenario ini serangan berasal dari perangkat B dan ditujukan ke perangkat D. Semua perangkat menerima serangan dari perangkat B karena ketiadaan *router* dalam jaringan tersebut. Namun yang bereaksi atas serangan tersebut hanya perangkat D. Serangan ini tidak mempengaruhi akses perangkat A ke jaringan Internet. Hasil pengamatan yang dilakukan oleh Hadiks dkk adalah sebagai berikut:

   - perangkat penyerang mengalami *overload* pada prosesor dan memaksa untuk menghentikan *script* serangan;
   - *webserver* pada perangkat D mengalami *restart*;
   - *webserver* pada perangkat D (korban) tidak *restart* akan tetapi perangkat *WiFi* terputus dari koneksi;

- Kinerja CPU pada perangkat korban berada pada utilisasi 100%.

2. Serangan DDoS
   Pada serangan DDoS (*Distributed Denial-of-Service*) penyerangan berasal dari perangkat B dan D dengan kapasitas serangan masing-masingnya adalah 100 serangan, 512 KB dan 100 serangan 256 KB. Korban dari serangan adalah perangkat E. Pengamatan selama serangan adalah sebagai berikut:

   - *webserver* pada perangkat korban dipaksa untuk *restart*, memutus semua koneksi aktiv;
   - *webserver* tidak terputus akan tetapi *WiFi adapter* tidak mendapatkan koneksi;
   - Beban CPU pada perangkat korban selalu 100%.

Hadiks dkk telah membuat beberapa skenario serangan DoS pada jaringan komunikasi D2D. Namun dalam artikelnya tidak disampaikan solusi atas eksperimen tersebut. Meskipun demikian solusi *security* dan *availability* pada komunikasi D2D telah dipaparkan dalam bagian 4.1.1 yang merupakan bentuk perbaikan dari algoritma Diffie-Hellman.

5 Kesimpulan

D2D dianggap sebagai teknologi yang menjanjikan yang mulai diperkenalkan pada LTE melalui 3GPP Rel. 12. Teknologi ini akan digunakan pada sistem 5G karena memberikan efisiensi penggunaan spektrum frekuensi, meningkatkan kapasitas, kecepatan data dan *latency*. Namun dibalik kelebihan tersebut terdapat celah keamanan karena sifat dari teknologi D2D yang terbuka (*open air*). Di dalam makalah ini telah disampaikan beberapa kajian terkait keamanan komunikasi D2D. Kajian tersebut sesuai dengan persyaratan keamanan D2D. Mekanisme pertukaran kunci pada proses autentikasi, serangan DoS pada D2D, keamanan pada *public safety* dan proses menangkal *eavesdropping* melalui lapisan fisik dikaji untuk memenuhi persyaratan keamanan D2D tersebut. Namun dalam kajian tersebut masih terkelompok-kelompok berdasarkan skenario komunikasi D2D (*discovery, relay* dan *public safety*). Perlu adanya sistem keamanan terintegrasi yang mencakup ketiga skenario tersebut. Hal tersebut merupakan *chalenge* (tantangan) dalam pengembangan keamanan komunikasi D2D.

Daftar Pustaka

[1] "Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2014-2019," pp. 2010–2015, 2011. [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white{_}paper{_}c11-520862.html

[2] "Third Generation Partnership Project 2." [Online]. Available: http://www.3gpp.org/specifications/releases/68-release-12

[3] M. Wang and Z. Yan, "Security in D2D Communications: A Review," *2015 IEEE Trustcom/BigDataSE/ISPA*, pp. 1199–1204, 2015. [Online]. Available: http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7345413

[4] A. BRYDON, "Opportunities and threats from LTE Device-to-Device (D2D) communication," 2014. [Online]. Available: http://www.unwiredinsight.com/2014/lte-d2d

[5] G. Fodor, "D2D Communications What Part Will It Play in 5G?" 2014. [Online]. Available: http://www.ericsson.com/research-blog/5g/device-device-communications/

[6] ITU, "IMT Vision - Framework and overall objectives of the future development of IMT for 2020 and beyond," 2015. [Online]. Available: http://www.itu.int/rec/R-REC-M.2083

[7] Z. Li, M. Moisio, M. a. Uusitalo, C. Wijting, F. S. Moya, and A. Yaver, "Overview on initial METIS D2D Concept," *METIS Deliverable*, pp. 203–208, 2014.

[8] 3GPP, "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Study on architecture enhancements to support Proximity-based Services (ProSe) (Release 12)," Tech. Rep., 2014.

[9] W. Shen, W. Hong, X. Cao, B. Yin, D. M. Shila, and Y. Cheng, "Secure key establishment for Device-to-Device communications," *Global Communications Conference (GLOBECOM), 2014 IEEE*, pp. 336–340, 2014.

[10] A. Zhang, J. Chen, R. Q. Hu, and Y. Qian, "SeDS: Secure Data Sharing Strategy for D2D Communication in LTE-Advanced Networks," *IEEE Transactions on Vehicular Technology*, vol. 9545, no. c, pp. 1–1, 2015.

[11] L. Goratti, G. Steri, K. M. Gomez, and G. Baldini, "Connectivity and Security in a D2D Communication Protocol for Public Safety Applications," *Wireless Communications Systems (ISWCS), 2014 11th International Symposium on*, pp. 548–552, 2014. [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp={&}arnumber=6933414{&}isnumber=6933305

[12] K. Jayasinghe, P. Jayasinghe, N. Rajatheva, and M. Latva-aho, "Physical Layer Security for Relay Assisted MIMO D2D Communication," pp. 651–656, 2015.

[13] A. Hadiks, Y. Chen, F. Li, and B. Liu, "A Study of Stealthy Denial-of-Service Attacks in Wi-Fi Direct Device-to-Device Networks," *Consumer Communications and Networking Conf. (CCNC)*, pp. 929–930.