Analisis Keamanan Arsitektur Jaringan Masa Depan Berbasis *Protocol Blocks* Sebagai Pedoman Untuk Menggantikan Arsitektur Jaringan Masa Kini

Beny Nugraha

Teknik Elektro, Universitas Mercu Buana, Jakarta
beny.nugraha@gmail.com

Abstrak

Arsitektur Jaringan Masa Depan (*Future Network Architectures*) sedang dikembangkan di berbagai negara untuk mengatasi masalah pada jaringan internet saat ini seperti tidak fleksibel (Jika ingin mengubah satu protokol dalam satu layer maka memerlukan perubahan protokol di layer yang berbeda lain. Sebagai contoh, untuk mengubah dari IPv4 ke IPv6 memerlukan versi modifikasi dari TCP) dan tidak mampu memberikan mekanisme keamanan secara intrinsik atau dengan kata lain mekanisme keamanan perlu ditambahkan untuk mengatasi ancaman keamanan yang baru. Sebagai contoh, IPsec untuk melindungi komunikasi IP. Salah satu basis yang digunakan untuk mendeseain Jaringan Masa Depan adalah *protocol blocks*, di mana jaringan akan terdiri dari sekumpulan protokol-protokol yang bersesuaian untuk menjalankan aplikasi tertentu. Dalam penelitian ini akan dibahas dua buah Jaringan Masa Depan yang berbasis *protocol blocks*, yaitu Service Oriented Network Architectures (SONATE) dan Netlet-based Node Architecture (NENA). Selain akan dibahas konsep atau cara kerja dari kedua arsitektur tersebut, juga akan dibahas bagaimana kedua jaringan tersebut mengamankan jaringannya, kemudian akan dibandingkan dengan keamanan pada jaringan Internet masa kini. Metodologi yang digunakan untuk melakukan analisa keamanan adalah kombinasi antara metode *system-centric* dan *attack centric*. Masalah keamanan pada setiap arsitektur didapatkan dengan membandingkan antara mekanisme keamanan pada setiap arsitektur dan mekanisme keamanan untuk menangani serangan keamanan. Hasil yang didapatkan adalah, kedua Jaringan Masa Depan tersebut mampu memberikan level keamanan yang sama dengan jaringan internet masa kini, namun keuntungannya adalah, kedua jaringan tersebut mampu berjalan secara fleksibel karena dapat menggunakan protokol-protokol keamanan yang diinginkan secara bebas dan dapat ditukar-tukar karena protokol-protokol tersebut ditempatkan ke dalam *protocol blocks*. Oleh karena itu, kedua Jaringan Masa Depan tersebut memiliki potensi untuk menggantikan jaringan internet masa kini.

***Keywords**:* Internet, Arsitektur jaringan masa depan, Keamanan jaringan, *Protocol blocks*

Received April 2014
Accepted for Publication May 2014

1. PENDAHULUAN

Internet telah menjadi fenomena di kehidupan sehari-hari, contohnya, hampir seluruh sektor industri memanfaatkan internet dalam melakukan pekerjaan mereka (contoh: saling bertukar data antar cabang perusahaan). Sektor-sektor industri yang menggunakan internet contohnya adalah perusahaan *software* seperti Microsoft dan Apple, perusahaan otomotif seperti Toyota dan Honda, juga perusahaan penyedia layanan telekomunikasi seperti Telkomsel dan Indosat. Lebih jauh lagi, internet digunakan untuk berkomunikasi antar pelanggan, contohnya dengan menggunakan aplikasi e-mail maupun Skype..

Walaupun memiliki banyak keuntungan, internet juga tidak luput dari beberapa masalah. Masalah-masalah ini tidak kasat mata oleh pengguna, contohnya adalah jaringan internet saat ini tidak fleksibel, dan juga tidak mampu memberikan keamanan secara intrinsik, sehingga membutuhkan mekanisme keamanan yang baru apabila muncul serangan keamanan yang baru. Masalah ini umumnya muncul dikarenakan prinsip desain jaringan internet yang sangat sulit untuk diubah-ubah [1]. Contoh prinsip desain yang dipakai pada internet masa kini adalah sebagai berikut:

1. Menggunakan struktur yang ber-layer (TCP/IP Layer). Apabila ditambahkan sebuah mekanisme keamanan pada sebuah layer, maka mekanisme tersebut hanya mengamankan layer tersebut saja, tidak mengamankan jaringan secara keseluruhan. Hal ini juga mengakibatkan internet masa kini tidak mampu memberikan keamanan yang intrinsik karena dibutuhkan mekanisme keamanan yang baru setiap muncul serangan keamanan yang baru.
2. Skema pengalamatan pada jaringan internet berbasiskan alamat dari pengguna (Alamat IP), oleh karena itu akan sulit untuk membuat internet menjadi *mobile* karena alamat IP tersebut telah ditentukan sesuai dengan lokasi penggunanya.

Beberapa arsitektur Jaringan Masa Depan sedang dikembangkan untuk mengatasi masalah-masalah pada jaringan internet masa kini, namun hanya dua buah jaringan yang akan dibahas yaitu Service Oriented Network Architectures (SONATE) dan Netlet-based Node Architecture (NENA). Kedua jaringan tersebut berbasis *protocol blocks*. Akan dibahas bagaimana kedua jaringan tersebut mengamankan jaringannya, serta akan dibandingkan dengan bagaimana jaringan internet masa kini mengamankan jaringan, sehingga akan diketahui apakah kedua jaringan tersebut berpotensi untuk menggantikan jaringan internet atau tidak.

2. METODOLOGI PENELITIAN

Metodologi yang digunakan pada penelitian ini adalah penggabungan antara metodologi *system centric* dan metodologi *attack centric*. Penjelasan lebih detail mengenai metodologi *system centric*, *attack centric*, dan juga metodologi yang digunakan pada penelitian ini akan diberikan pada sub-bab selanjutnya.

2.1 Metodologi *System Centric*

Metodologi ini dilakukan dengan cara memodelkan sebuah sistem tertentu secara menyeluruh. Pemodelan sistem tersebut dilakukan dengan cara menganalisis mekanisme-mekanisme yang ada pada sistem tersebut serta mengidentifikasi mekanisme-mekanisme keamanan yang diimplementasi ke dalam sistem tersebut [2]. Sebagai contoh, jika dalam sebuah sistem telah diidentifikasi baha sistem tersebut menggunakan mekanisme pengalamatan seperti alamat port maupun alamat IP, maka sistem tersebut akan mudah diserang oleh serangan keamanan yang memulai serangannya dengan cara mendapatkan alamat sistem tersebut. Jika alamat sistem tersebut telah didapatkan, maka penyerang dapat melihat paket atau data yang masuk dan keluar dari sistem tersebut. Contoh lainnya, apabila diidentifikasi bahwa sebuah sistem memiliki mekanisme enkripsi data, maka sistem tersebut akan tahan terhadap serangan-serangan yang bertujuan untuk memodifikasi data yang masuk dan keluar dari sistem.

2.2 Metodologi *Attack Centric*

Metodologi ini dilakukan dengan cara memodelkan sebuah serangan tertentu. Pemodelan sebuah serangan tersebut dapat dilakukan dengan cara menganalisis bagaimana serangan tersebut bekerja serta apakah tujuan dari serangan tersebut. Contohnya, apabila diidentifikasi bahwa sebuah serangan bekerja dengan cara mengambil dan memodifikasi paket atau data, maka selanjutnya dapat ditentukan bahwa mekanisme yang tepat untuk menangani serangan tersebut adalah dengan mekanisme enkripsi.

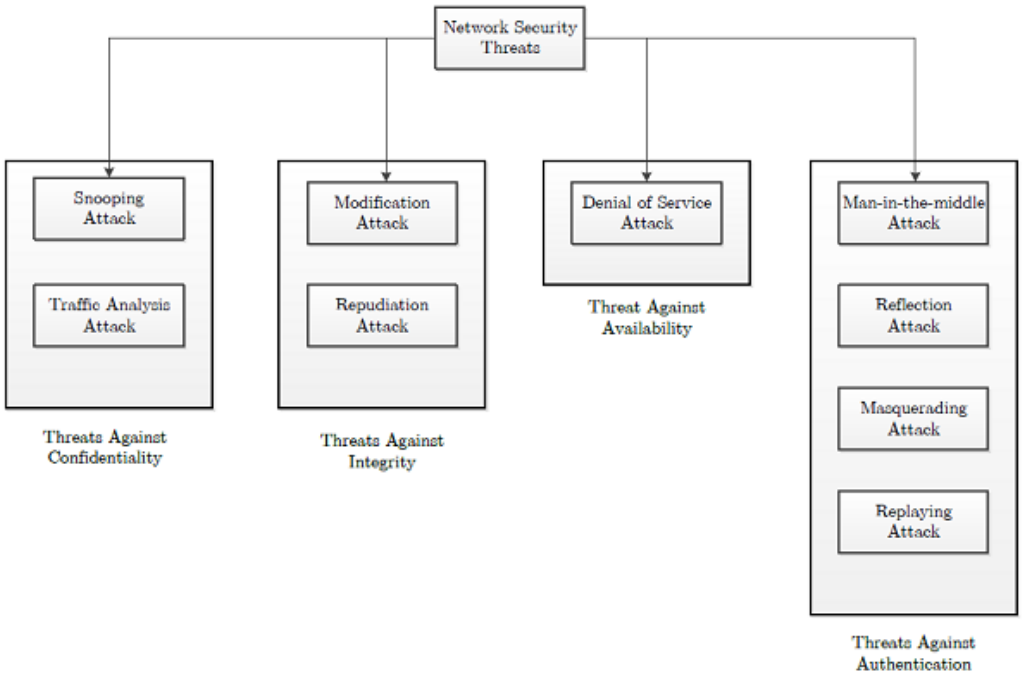
2.3 Metodologi Yang Digunakan Pada Penelitian Ini

Sudah disebutkan bahwa metodologi yang digunakan pada penelitian ini adalah metode yang menggabungkan metode system centric dengan metode *attack centric*. Metode tersebut terdiri dari beberapa langkah berikut:

1. Mengidentifikasi beberapa serangan keamanan, kemudian mengidentifikasi mekanisme yang diperlukan untuk menangkal serangan tersebut.
2. Menganalisis mekanisme-mekanisme keamanan yang ada pada SONATE, NENA, dan jaringan internet masa kini.
3. Menentukan masalah-masalah keamanan yang sudah bisa diatasi oleh SONATE, NENA, dan jaringan internet masa kini, dan kemudian membandingkan ketiga-nya.

3. IDENTIFIKASI SERANGAN KEAMANAN

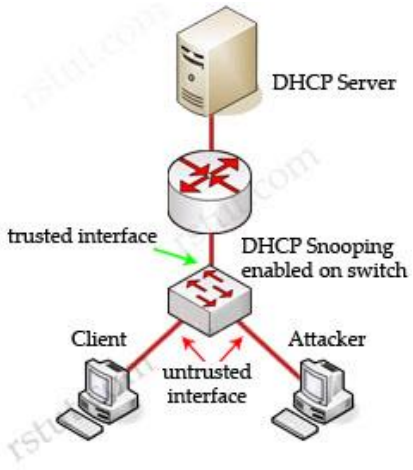
Terdapat sembilan buah serangan keamanan yang dibahas pada penelitian ini, ke-sembilan keamanan tersebut bertujuan untuk menyerang empat buah *network security goals* yaitu *confidentiality*, *integrity*, *availability*, dan *authentication*. Bagan yang mengelompokkan ke-sembilan serangan keamanan berdasarkan network security goals yang mereka serang dapat dilihat pada Gambar 1.



Gambar 1. Bagan Serangan Keamanan Jaringan

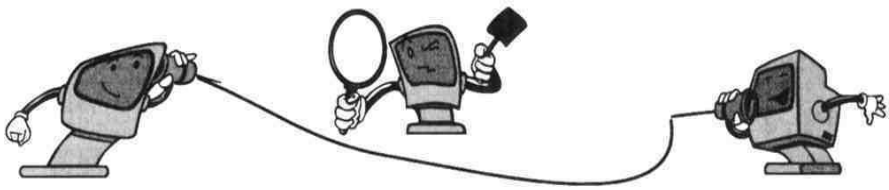
Terlihat dari Gambar 1, ke-sembilan serangan tersebut adalah *snooping attack* dan *traffic analysis attack* (menyerang *confidentiality*), *modification attack* dan *repudiation attack* (menyerang *integrity*), *denial-of-service attack* (menyerang *availability*), *man-in-the-middle attack, reflection attack, masquerading attack*, dan *replaying attack* (menyerang *authentication*). Penjelasan lebih lengkap mengenai ke-sembilan serangan keamanan tersebut adalah sebagai berikut:

1. *Snooping Attack*
   Serangan ini bertujuan untuk melihat paket atau data yang masuk dan keluar dari sebuah sistem. Salah satu cara untuk melakukan serangan ini adalah dengan mendapatkan alamat port atau alamat IP dari sistem yang diinginkan, cara lain adalah dengan memonitor paket-paket yang bergerak di dalam jaringan kemudian mengambilnya. Apabila paket-paket ini tidak terproteksi, maka penyerang dapat melakukan *snooping attack* [3]. Dengan mengetahui cara kerja dari serangan ini, maka dapat diidentifikasi mekanisme serangan yang dibutuhkan untuk menangkal serangan ini adalah mekanisme enkripsi. Dengan mengekripsi paket-paket yang mengalir di jaringan, si penyerang tidak akan mampu melihat isi dari paket tersebut, sehingga *snooping attack* tidak dapat dilakukan. Pada Gambar 2 diperlihatkan ilustrasi untuk *snooping attack*. Pada Gambar 2 diperlihatkan seorang attacker menggunakan DHCP *snooping* pada *switch* sehingga penyerang dapat mengambil paket dari *client*.



Gambar 2. Ilustrasi *Snooping Attack* [4]

2. *Traffic Analysis Attack*
   Untuk melakukan serangan ini, penyerang akan mengambil dan menganalisis pergerakan dari paket-paket yang berada pada jaringan, dengan tujuan untuk mendapatkan informasi dari pola trafik paket-paket tersebut. Semakin banyak paket yang diobservasi, semakin banyak pula informasi yang didapatkan. Salah satu cara untuk melakukan serangan ini adalah dengan menggunakan software Wireshark, software ini dapat melihat paket-paket yang melintas pada jaringan, dan dengan melihat pola trafiknya, maka dapat diketahui seorang pengguna sedang mengunduh sesuatu dari sebuah situs [5]. Mekanisme keamanan yang dapat diterapkan untuk mengatasi serangan ini adalah dengan cara menyembunyikan alamat dari user dan sistem yang sedang berkomunikasi, sehingga penyerang tidak dapat menentukan di titik mana dia harus melakukan analisa trafik. Pada Gambar 3 diperlihatkan ilustrasi dari *traffic analysis attack.*



Gambar 3. Ilustrasi *Traffic Analysis Attack* [6]

3. *Modification Attack*
   Pada serangan ini, penyerang berusaha untuk menghapus atau mengubah informasi yang ada pada sebuah paket secara ilegal, namun tetap dilihat sebagai informasi yang valid oleh target. Contohnya, seorang penyerang menangkap paket yang datang dari user bernama Alice, kemudian mengubahnya terlebih dahulu sebelum dikirim ke Bob, dan Bob akan

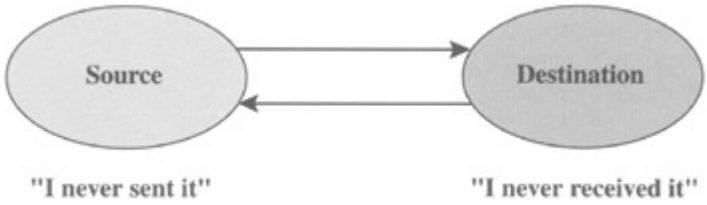
mengira paket ini benar-benar datang dari Alice [7]. Mekanisme keamanan yang dibutuhkan untuk mengatasi serangan ini adalah dengan melakukan proses hash kepada pesan tersebut, atau dengan menggunakan digital signature sehingga penerima. dalam contoh bernama Bob, dapat yakin bahwa pesan tersebut benar-benar datang dari Alice dan belum pernah diubah-ubah. Ilustrasi dari *modification attack* dapat dilihat pada Gambar 4.

Gambar 4. Ilustrasi terjadinya *modification attack* [8].

Dari Gambar 4 di atas terlihat bahwa penyerang yang bernama Alex mengambil pesan dari Halim, dan kemudian mengubahnya terlebih dahulu sebelum dikembalikan ke Anita. Serangan ini berhasil apabila Anita menganggap pesan yang telah diubah tersebut benar-benar datang dari Halim.

4. *Repudiation Attack*

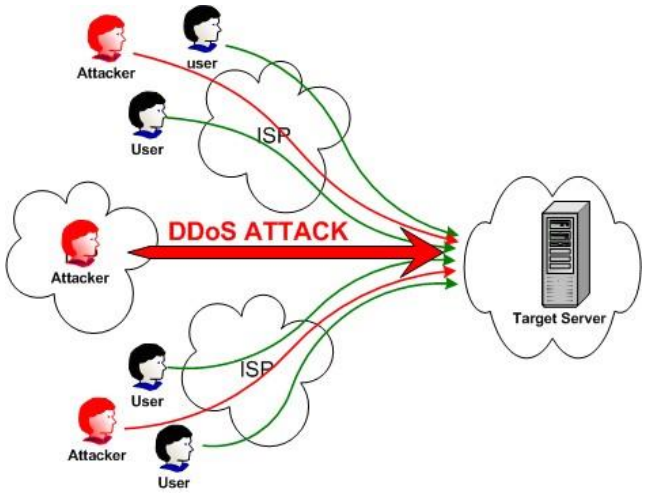
Repudiation adalah sebuah serangan di mana seorang user tidak dapat membuktikan bahwa transmisi data telah dilakukan antara dia dengan user yang lainnya, sehingga user lain dapat menyangkal bahwa dia telah mengirim atau menerima data [7]. Mekanisme yang dibutuhkan untuk mengatasi serangan ini adalah dengan menempatkan trusted third party (entitas ke-tiga yang terpercaya) sehingga entitas tersebut dapat membuktikan bahwa transmisi data antara dua user memang telah terjadi. Ilustrasi dari repudiation attack dapat dilihat pada Gambar 5.



Gambar 5. Ilustrasi terjadinya *repudiation attack* [9].

5. *Denial-of-Service Attack*

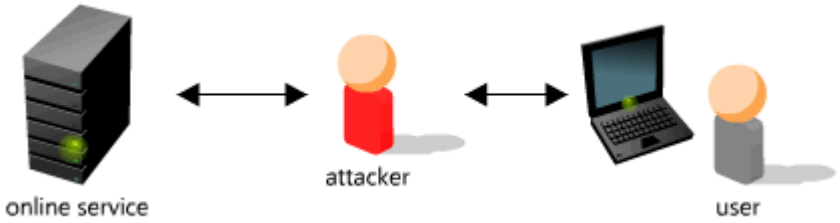
Denial-of-service (DoS) attack bertujuan untuk membuat sebuah sistem atau server menjadi tidak dapat diakses untuk sementara waktu. Cara yang dapat dilakukan seorang penyerang untuk melakukan *DoS attack* adalah dengan membanjiri sistem atau server tersebut dengan paket-paket sampah yang banyak jumlahnya. Serangan ini dapat dicegah dengan mengunakan mekanisme *flow control*, sehingga dapat membatasi jumlah paket yang dapat masuk ke dalam sebuah server, dan apabila terdapat anomali dari jumlah paket yang masuk, server akan memutuskan hubungannya [10]. Ilustrasi dari *DoS attack* di mana terdapat seorang penyerang yang membanjiri sebuah server dapat dilihat pada Gambar 6.



Gambar 6. Ilustrasi *DoS Attack* [11].

6. *Man-in-the-middle Attack*

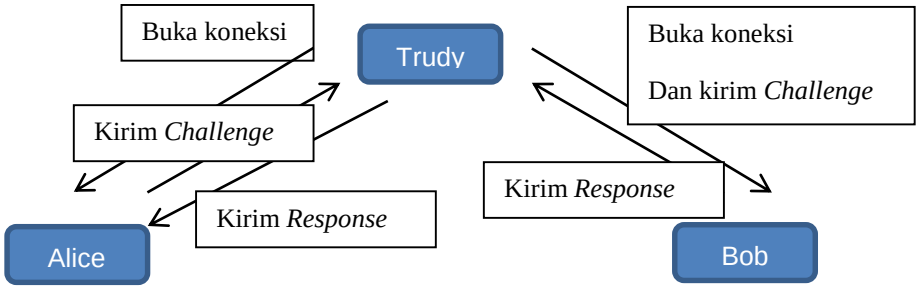
Tipe serangan ini membuat kelihatannya dua user yang legal sedang berkomunikasi satu sama lain, padahal sebenarnya terdapat seseorang di antara mereka yang mengirim dan menerima data dari kedua user tersebut. Mekanisme keamanan yang dapat diterapkan untuk mengatasi serangan ini adalah dengan mengautentikasi seluruh user dengan menggunakan digital signature. Ilustrasi *man-in-the-middle attack* antara seorang user dan sebuah server dapat dilihat pada Gambar 7.



Gambar 7. Ilustrasi man-in-the-middle attack antara user dan server [12].

7. *Reflection Attack*

Contoh skenario *reflection attack* adalah sebagai berikut: Terdapat seorang penyerang bernama Trudy dan Trudy membuat sebuah koneksi dengan seorang user bernama Alice. Alice mencoba untuk mengetahui identitas Trudy dengan mengirim pesan bernama *challenge*. Sebelum Trudy menjawab pesan dari Alice, Trudy membuat sebuah koneksi baru dengan Bob, dan Trudy mengirimkan *challenge* dari Alice ke Bob. Bob akan menjawab *challenge* tersebut dengan mengirimkan pesan bernama *response* ke Trudy, *response* ini diteruskan oleh Trudy ke Alice, sehingga sekarang Trudy dapat berkomunikasi dengan Alice dengan menggunakan identitas Bob [13]. Serangan ini dapat diatasi dengan menggunakan *digital signature* yang akan mengautentikasi seluruh user yang valid. Ilustrasi dari serangan ini dapat dilihat pada Gambar 8.



Gambar 8. Ilustrasi *Reflection Attack*.

8. *Masquerading Attack*

Seorang penyerang akan berpura-pura menjadi seorang user yang valid untuk mengakses sebuah sistem. Contoh sederhananya, seorang penyerang mampu mendapatkan informasi log in dan password facebook dari seseorang, dan kemudian mengakses halaman facebook orang tersebut. Mekanisme yang dibutuhkan untuk mengatasi ini adalah dengan mengautentikasi seluruh user dengan menggunakan digital signature, atau bisa juga dengan menyembunyikan informasi log in dan password sehingga penyerang akan mengalami kesulitan untuk berpura-pura menjadi orang tersebut [14].

9. *Replaying Attack*

Replaying attack memiliki mekanisme yang mirip dengan *reflection attack*, perbedaannya adalah message yang telah ditangkap akan di-*replay* pada sesi komunikasi yang berbeda, pada *reflection attack*, message akan di-*reflect* pada sesi yang sama [15]. Dari cara kerja serangan tersebut, maka mekanisme keamanan yang bisa digunakan untuk mengatasi *replaying attack* adalah dengan menambahkan sebuah parameter sebagai penanda sebuah sesi komunikasi, misalnya dengan menggunakan sebuah *random number* (nonce), *session key*, atau *timestamp*.

4. ANALISIS KEAMANAN PADA ARSITEKTUR JARINGAN MASA DEPAN BERBASIS *PROTOCOL BLOCKS*

Sudah disebutkan bahwa pada penelitian ini akan diteliti dua buah jaringan masa depan yang berbasis *protocol blocks* yaitu SONATE dan NENA. *Protocol blocks* secara umum berarti bahwa dalam menjalankan aplikasi yang diinginkan pada SONATE atau NENA, maka akan dipilih protokol-protokol yang sesuai dengan persyaratan aplikasi tersebut. Protokol-protokol tersebut dikumpulkan ke dalam satu atau beberapa blok, pada SONATE, protokol tersebut dikumpulkan ke dalam kompoenen yang bernama *Building Block*, sedangkan pada NENA, protokol tersebut dikumpulkan ke dalam komponan yang bernama *netlet*. Penjelasan lebih lengkap mengenai mekanisme SONATE dan NENA akan dijelaskan pada subbab 4.1 dan 4.2.

Analisis keamanan dilakukan dengan menganalisis cara kerja kedua jaringan tersebut, mengidentifikasi mekanisme keamanan pada kedua jaringan tersebut, dan kemudian menentukan serangan mana saja (dari sembilan serangan keamanan di atas) yang dapat diatasi oleh kedua jaringan tersebut.

4.1 SONATE (Service Oriented Network Architecture)

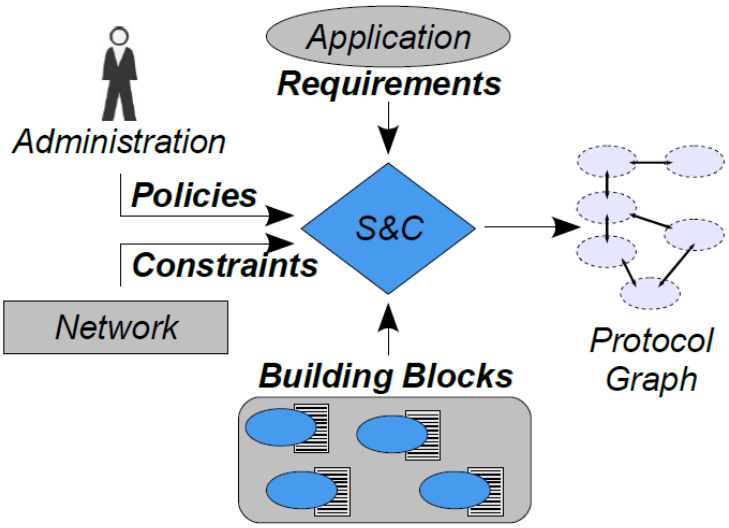
4.1.1 Mekanisme Jaringan SONATE

SONATE adalah arsitektur jaringan masa depan yang menganut prinsip *Service Oriented Architecture* (SOA). SOA menggunakan sekumpulan layanan-layanan yang merupakan efek dari pengimplementasian sebuah protokol. Contoh layanan yang dapat digunakan antara lain data enkrispi, *digital signature*, dan *flow control* [16] [17].

SONATE merupakan tipe arsitektur *loosy-coupled*, yang bersifat fleksibel sehingga mekanisme-mekanisme baru dapat segera diimplementasikan ke dalam jaringan. Lebih lanjut lagi, SONATE menggunakan komponen yang bernama *Bulding Block* (BB) yang merupakan implementasi dari beberapa layanan atau protocol [18].

Untuk menyediakan proses komunikasi secara komplit, SONATE menggunakan proses pemilihan dan pembentukan dari *Bulding Block*. Ilustrasi untuk proses pemilihan dan pembentukan *Building Block* ini dapat dilihat pada Gambar 9.

Terlihat dari Gambar 9 bahwa komponen utama dari proses pemilihan dan pembentukan *Building Block* adalah persyaratan dari aplikasi yang diinginkan, deskripsi layanan dari *Building Block* yang tersedia, keterbatasan jaringan, serta kebijakan administrasi. Tujuan utama dari keseluruhan proses ini adalah untuk menciptakan *protocol graph*, yang terdiri dari sekumpulan protokol, yang efisien dan optimal sesuai dengan persyaratan yang diinginkan. Seluruh proses pemilihan dan pembentukan ini terjadi di saat aplikasi membutuhkan layanan komunikasi, atau pada saat *run time* [18]. Dari mekanisme SONATE di atas, dapat disimpulkan bahwa SONATE memiliki keuntungan yaitu lebih fleksibel daripada jaringan internet masa kini.



Gambar 9. Pemilihan dan Pembentukan Building Blocks [18].

4.1.2 Mekanisme Keamanan Jaringan SONATE

Oleh karena *Building Block* adalah implementasi dari layanan-layanan komunikasi, termasuk juga layanan keamanan, maka tingkat keamanan pada SONATE bergantung pada pemilihan *Building Block* yang sesuai. Layanan keamanan yang bisa digunakan contohnya adalah data enkripsi, *digital signature*, PKI, dan sebagainya. Selanjutnya akan ditentukan vulnerabilitas SONATE terhadap sembilan serangan keamanan. Penjelasannya adalah sebagai berikut:

1. *Snooping attack*
   SONATE mampu memilih *Building Block* yang menyediakan layanan data enkripsi, sehingga walaupun seorang penyerang mampu mendapatkan data yang dia inginkan, tapi dia tidak akan bisa membaca-nya. Oleh karena itu, SONATE dapat mengatasi *snooping attack*.

2. *Traffic analysis attack*
   Serangan ini tidak dapat diatasi oleh SONATE karena seorang penyerang masih bisa mendapatkan alamat (*IP address*) dari targetnya. Salah satu cara untuk mengatasi serangan ini adalah dengan menyembunyikan alamat atau identitas user, namun mekanisme ini belum tersedia pada SONATE.

3. *Modification attack*
   Pemakai jaringan SONATE dapat memilih *Building Block* yang menyediakan layanan *digital signature*. Dengan bantuan *digital signature*, integritas dari data dapat dijamin karena hanya user yang legal saja yang bisa memodifikasi data-nya, apabila user lain (yang ilegal) memodifikasi data-nya, maka akan dengan mudah terdeteksi.

4. *Repudiation attack*
   Bulding Block yang menyediakan layanan *digital signature* juga dapat menciptakan pesan/data yang *non-repudiation* (identitas dari kedua pengirim tersedia di dalam pesan/data), namun hal ini tidak cukup karena salah satu atau kedua user dapat menyangkal telah terjadi pertukaran data antara mereka, sehingga dibutuhkan suatu komponen ketiga yaitu *Trusted Third Party*. Oleh karena *Trusted Third Party* tidak tersedia pada SONATE, maka *repudiation attack* masih belum bisa diatasi.

5. *Denial-of-Service (DoS) attack*
   Serangan ini dapat diatasi dengan cara memilih *Building Block* yang menyediakan layanan *flow control*. Dengan layanan ini, jumlah paket yang mengalir dalam jaringan bisa dikendalikan, sehingga apabila dalam suatu waktu terdapat seseorang yang membanjiri jaringan dengan paket-paket sampah, maka hal tersebut dapat dideteksi dengan mudah, dan kemudian koneksi akan diputus.

6. *Man-in-the-middle attack*
   Mekanisme autentikasi seperti *digital signature* dapat digunakan pada SONATE, sehingga seluruh user yang berada dalam jaringan telah terautentikasi. Mekanisme ini dapat digunakan untuk mencegah *man-in-the-middle attack.*

7. *Reflection attack*
   Reflection attack memiliki basis serangan yang hampir sama dengan *man-in-the-middle attack*, oleh karena itu, penggunaan *digital signature* dapat mengatasi serangan ini.

8. *Masquerading attack*
   Masquerading attack juga memiliki mekanisme yang hampir sama dengan *man-in-the-middle attack* dan *reflection attack*, oleh karena itu, pemilihan *Building Block* yang menyediakan layanan *digital signature* dapat mencegah serangan ini.

9. *Replaying attack*
   Sudah dijelaskan bahwa pada SONATE akan terbentuk sebuah *protocol graph* untuk sebuah sesi komunikasi. Hal ini berarti satu *protocol graph* hanya digunakan untuk sebuah sesi, sehingga *replaying attack* dapat diatasi.

Tabel yang menyimpulkan hasil analisis keamanan di atas dapat dilihat sebagai berikut:

Tabel 1. Vulnerabilitas Keamanan Pada SONATE

Security Attacks	Mekanisme Keamanan
Snooping	*Building Block* dengan data enkripsi
Traffic Analysis	**Belum tersedia**
Modification	*Building Block* dengan *digital signature*
Repudiation	**Belum tersedia**
Denial of Service	*Building Block* dengan *flow control*
Man-in-the-Middle	*Building Block* dengan *digital signature*
Reflection	*Building Block* dengan *digital signature*
Masquerading	*Building Block* dengan *digital signature*
Replaying	Satu *protocol graph* untuk satu sesi komunikasi

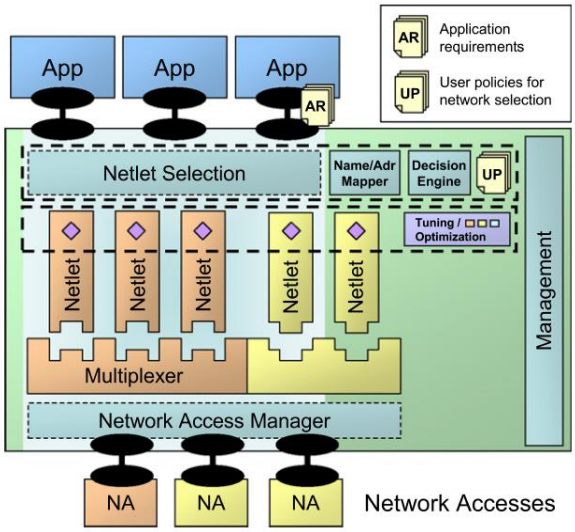
Dari hasil analisis di atas dapat disimpulkan bahwa tingkat keamanan pada SONATE sangat bergantung pada pemilihan *Building Block* yang sesuai. SONATE masih belum memiliki mekanisme untuk menangani *traffic analysis attack* dan *repudiation attack* karena SONATE tidak menyediakan mekanisme untuk menyembunyikan alamat atau identitas user, serta tidak memiliki *Trusted Third Party*. Walaupun masih rentan terhadap dua serangan tersebut, SONATE adalah sebuah arsitektur jaringan yang fleksibel, sehingga mekanisme untuk mengatasi dua serangan tersebut masih bisa ditemukan sebelum diluncurkan ke publik.

4.2 NENA (Netlet-based Node Architecture)

4.2.1 Mekanisme Jaringan NENA

Sudah dijelaskan di atas bahwa NENA adalah arsitektur jaringan masa depan yang menganut prinsip protocol blocks, yaitu beberapa set protokol yang dikumpulkan ke dalam sebuah block. Pada NENA, block ini disebut dengan netlet [19].

Sama seperti SONATE, NENA juga merupakan arsitektur *loosy-coupled*, yang berarti protocol-protokol yang berada di dalam netlet dapat dimodifikasi dengan mudah sehingga NENA adalah jaringan yang fleksibel. Ilustrasi dari NENA dapat dilihat pada Gambar 10.



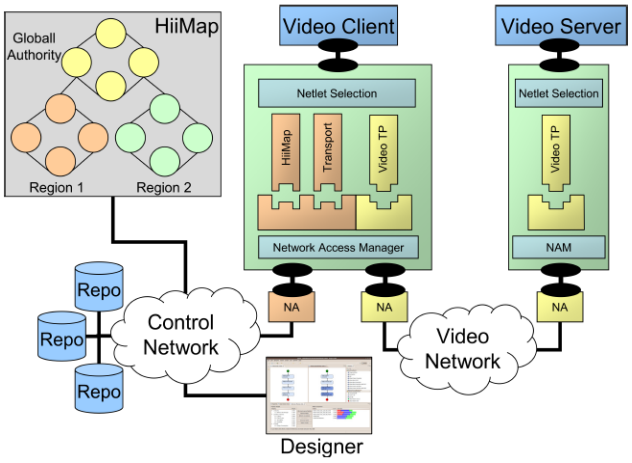
Gambar 10. Arsitektur NENA [19].

Netlet dapat dilihat sebagai sebuah wadah tempat dikumpulkannya protokol-protokol yang diinginkan. User tidak mengakses *netlet* secara langsung, melainkan hanya perlu menyebutkan persyaratan-persyaratan yang mereka inginkan, kemudian komponen *Netlet Selection* akan memilih *netlet* yang tepat sesuai dengan keinginan user. *Netlet* yang terpilih ini yang akan digunakan untuk menjalankan komunikasi yang diinginkan user. User dapat memilih *netlet* sesuai dengan spesifikasi yang mereka inginkan, contohnya apabila user menginginkan menjalankan komunikasi dengan menggunakan layanan data enkripsi, maka *netlet* yang dipilih adalah *netlet* yang mengandung protokol untuk data enkripsi [18].

Pembentukan dari netlet tersebut dijalankan pada saat sebelum user menjalankan komunikasi, atau pada saat design time. Sedangkan pemilihan netlet dilakukan pada saat user menjalankan komunikasi, atau pada saat run time [18].

Terdapat tiga buah mekanisme keamanan pada NENA, yaitu:

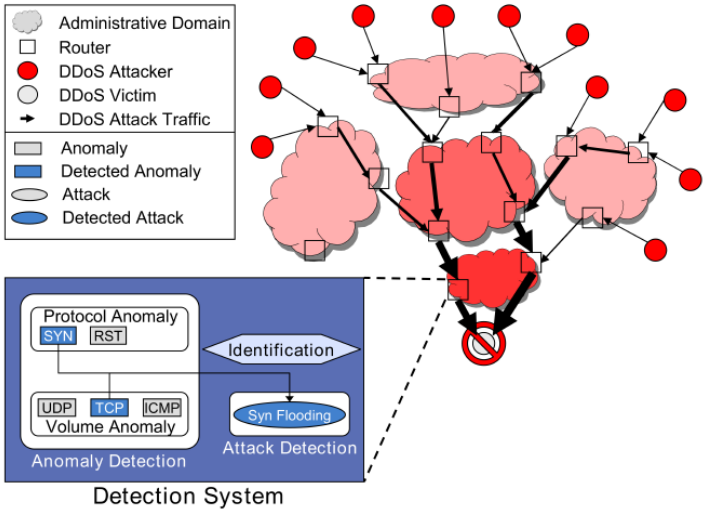
1. Pemilihan netlet yang tepat akan menentukan tingkat keamanan NENA, hal ini dikarenakan protokol-protokol yang terdapat pada netlet bisa berupa protokol untuk keamanan seperti data enkripsi atau digital signature.
2. Protokol-protokol yang digunakan pada NENA telah dipastikan aman dari pemodifikasian yang ilegal. Sebagai tambahan, integritas dari setiap protokol juga terjamin karena setiap protokol mendapat Protocol ID yang unik. Gambar 11 mengilustrasikan proses untuk penggunakan protokol tersebut.



Gambar 11. Proses Penggunaan Protokol Pada NENA [20].

3. NENA menggunakan suatu mekanisme yang bernama *collaborative attack detection* [21]. Mekanisme ini terbagi menjadi tiga bagian yaitu:
   a. Deteksi lokal terhadap anomali trafik
   b. Identifikasi lokal terhadap serangan keamanan
   c. Menggunakan kolaborasi dari beberapa sistem deteksi untuk menangani serangan keamanan

   Gambar 12 mengilustrasikan proses *collaborative attack detection*.



Gambar 12. Ilustrasi *Collaborative Attack Detection* [21].

Secara umum, mekanisme *collaborative attack detection* digunakan untuk mengatasi DoS attack dengan cara mengidentifikasi anomali trafik secepat mungkin, kemudian memutuskan koneksinya.

4.2.2 Mekanisme Keamanan Jaringan NENA

Setelah mengetahui mekanisme-mekanisme keamanan pada NENA, langkah selanjutnya adalah menentukan vulnerabilitas NENA terhadap sembilan serangan keamanan. Penjelasannya adalah sebagai berikut:

1. *Snooping attack*
   Serangan ini dapat diatasi oleh NENA karena NENA mampu memilih *netlet* yang menyediakan layanan data enkripsi.

2. *Traffic analysis attack*
   Sama seperti SONATE, serangan ini tidak dapat diatasi oleh NENA karena seorang penyerang masih bisa mendapatkan alamat (*IP address*) dari targetnya. Salah satu cara untuk mengatasi serangan ini, yaitu dengan menyembunyikan alamat atau identitas user tidak tersedia pada NENA.

3. *Modification attack*
   Integritas dari data dapat dijamin dengan memilih *netlet* yang mengandung layanan *digital signature*. Penggunaan digital signature akan menjamin hanya user yang legal saja yang bisa memodifikasi data-nya, apabila penyerang yang mencoba memodifikasi data-nya, maka akan terdeteksi dengan mudah.

4. *Repudiation attack*
   Sama seperti SONATE, NENA mampu memilih *netlet* yang menyediakan layanan *digital signature* juga dapat menciptakan pesan/data yang *non-repudiation* d mana identitas dari kedua pengirim tersedia di dalam pesan/data, namun hal ini tidak cukup karena salah satu atau kedua user dapat menyangkal telah terjadi pertukaran data antara mereka. Untuk mengatasi *repudiation attack* dibutuhkan suatu komponen ketiga yaitu *Trusted Third Party*, komponen ini tidak tersedia pada NENA, sehingga *repudiation attack* masih belum bisa ditangani oleh NENA.

5. *Denial-of-Service (DoS) attack*
   Serangan ini dapat diatasi menggunakan mekanisme *collaborative attack detection*. Seperti telah dijelaskan di atas, mekanisme ini akan dengan mudah mengidentifikasi kejanggalan pada jumlah trafik atau paket dalam jaringan, kemudian mengidentifikasi alamat pengirim paket-paket tersebut, dan kemudian memutuskan koneksinya.

6. *Man-in-the-middle attack*
   Dengan memilih *netlet* yang menyediakan layanan *digital signature,* maka seluruh user dalam jaringan akan terautentikasi. Mekanisme ini dapat digunakan untuk mencegah *man-in-the-middle attack.*

7. *Reflection attack*
   Reflection attack memiliki basis serangan yang hampir sama dengan *man-in-the-middle attack*, oleh karena itu, pemilihan *netlet* dengan *digital signature* dapat mengatasi serangan ini.

8. *Masquerading attack*
   Masquerading attack juga memiliki mekanisme yang hampir sama dengan *man-in-the-middle attack* dan *reflection attack*, oleh karena itu, pemilihan *netlet* yang menyediakan layanan *digital signature* dapat mencegah serangan ini.

9. *Replaying attack*
   Mekanisme untuk mengatasi serangan ini dapat dipilih sesuai dengan persyaratan *netlet* dari user. Mekanisme yang bisa dipilih diantaranya adalah *timestamp* atau *random number* (*nonce*). Timestamp maupun random number sama-sama memiliki fungsi untuk mengikat sebuah sesi komunikasi, sehingga user dapat dengan mudah mengecek apakah paket yang dia terima berasal dari sesi komunikasi yang sama atau tidak.

Tabel yang menyimpulkan analisis dari vulnerabilitas keamanan pada NENA dapat dilihat sebagai berikut:

Tabel 2. Vulnerabilitas Keamanan Pada NENA

Security Attacks	**Mekanisme Keamanan**
Snooping	*Netlet* dengan data enkripsi
Traffic Analysis	**Belum tersedia**
Modification	*Netlet* dengan *digital signature*
Repudiation	**Belum tersedia**
Denial of Service	*Collaborative Attack Detection*
Man-in-the-Middle	*Netlet* dengan *digital signature*
Reflection	*Netlet* dengan *digital signature*
Masquerading	*Netlet* dengan *digital signature*
Replaying	*Netlet* dengan *timestamp* atau *random number* (*nonce*)

Dari hasil analisis di atas dapat disimpulkan bahwa pemilihan *netlet* menjadi faktor utama yang menentukan tingkat keamanan NENA. Sama seperti SONATE,

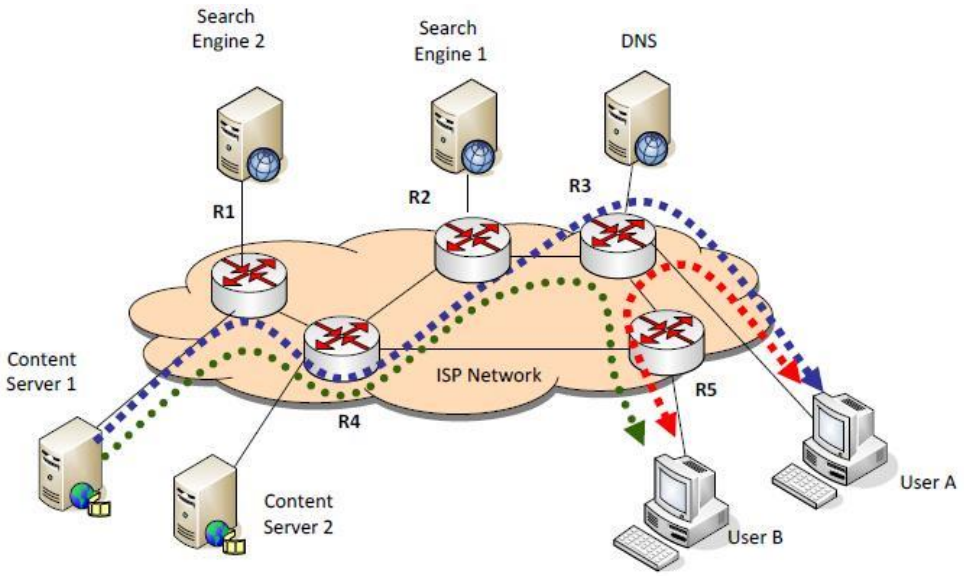
NENA masih belum mampu mengatasi traffic analysis attack serta *repudiation attack*, hal ini dikarenakan NENA tidak memiliki mekanisme untuk menyembunyikan identitas user serta tidak memiliki *Trusted Third Party*. Namun demikian, NENA merupakan arsitektur jaringan yang fleksibel, sehingga layanan-layanan yang bisa didapat pada *netlet* juga mudah untuk ditambah dan diimplementasikan, oleh karena itu diharapkan NENA mampu mengatasi seluruh serangan keamanan jaringan sebelum dikomersialkan.

5. ANALISIS KEAMANAN PADA JARINGAN INTERNET MASA KINI

5.1 Mekanisme Jaringan Internet

Jaringan internet saat ini menggunakan desain prinsip *address based*, dengan prinsip desain ini, seorang user meminta konten yang diinginkan dengan cara mengakses alamat (*IP Address*) dari server penyedia konten. Gambar 13 mengilustrasikan cara user meminta konten dari sebuah server.

Gambar 13 mengilustrasikan bagaimana user mencoba untuk mendapatkan konten yang dia inginkan dari sebuah server. Pertama-tama, sebuah komponen yang bernama *search engine* akan menelusuri jaringan untuk mencari, menyimpan, dan mengklasifikasikan konten-konten yang ada dalam jaringan tersebut.



Gambar 13. Diagram Jaringan Internet

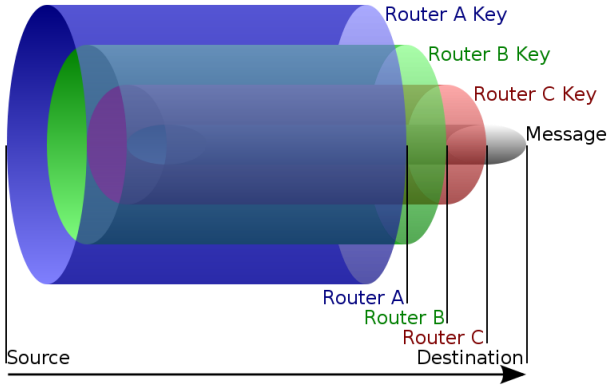
Langkah kedua adalah user menggunakan fasilitas *search engine* untuk mencarikan konten yang dia inginkan, user tidak perlu mengetahui di mana lokasi dari server penyedia konten. Langkah terakhir adalah konten yang sesuai dengan keinginan user akan dikirimkan dari server ke user. Konten dapat terkirim dengan cara sebagai berikut: User akan memilih alamat (IP Address) dari server yang telah disediakan *search engine* dan konten akan terkirim ke user.

5.2 Mekanisme Keamanan Jaringan Internet

Pada awalnya, tidak ada mekanisme keamanan pada tiap-tiap layer pada jaringan internet. Hal ini dikarenakan tujuan utama dari adanya layer tersebut hanyalah untuk mengirim dan menerima data. Namun, seiring berjalannya waktu, ditambahkanlah berbagai mekanisme keamanan pada tiap-tiap layer sehingga data yang dikirim dan diterima akan menjadi lebih aman. Oleh karena itu, mekanisme keamanan pada Internet tidaklah intrinsik atau *built-in* di dalam jaringannya, melainkan harus ditambal-sulam. Hal ini mengakibatkan kurangnya fleksibilitas dari keamanan Internet, karena apabila muncul jenis serangan yang baru, maka mekanisme keamanan yang baru perlu diaplikasikan.

Beberapa mekanisme keamanan yang saat ini diaplikasikan pada internet adalah sebagai berikut:

1. *Onion Routing*
   Onion routing adalah teknik yang digunakan untuk menyembunyikan identitas user, sehingga tercipta suatu *anonymous communication* antar user. Ilustrasi dari *onion routing* adalah sebagai berikut:



Gambar 14. Ilustrasi Onion Routing [22]

   Cara kerja dari *onion routing* secara singkat adalah sebagai berikut: setiap paket yang mengalir akan dienkripsi dan dikirimkan ke beberapa node yang bernama *onion router*. Setiap *onion router* yang mendapat paket akan mendeskripsi paket tersebut untuk menentukan informasi routing sehingga paket dapat diteruskan ke onion router selanjutnya.

2. *Transport Layer Security* (TLS)
   Sesuai namanya, teknik ini diaplikasikan pada layer transport. Teknik ini mengamankan internet dengan cara mengautentikasi user dengan menggunakan Message Authentication Code (MAC) serta mengenkripsi data yang mengalir dengan menggunakan teknik *symmetric key encryption*. Selain dua mekanisme ini, TLS juga bisa digunakan untuk menghasilkan nilai hash dari sebuah paket. TLS pertama kali diaplikasikan pada Internet

pada Januari 1999.

3. *Internet Protocol Security* (IPSec)
   IPSec adalah mekanisme keamanan yang diaplikasikan pada layer IP. Teknik ini akan mengautentikasi serta mengenkripsi setiap paket yang mengalir pada jaringan IP. IPSec pertama kali distandarkan pada tahun 2005.

4. *Digital Signature* dan *Public Key Infrastructure* (PKI)
   Kombinasi antara *digital signature* dan PKI digunakan secara luas sebagai cara untuk memverifikasi user atau sumber. Dengan mengecek *digital signature* maupun sertifikat yang dihasilkan PKI, setiap user dapat yakin bahwa lawan bicara-nya adalah user legal. *Digital signature* pertama kali diaplikasikan pada tahun 1989 sementara PKI pertama kali diaplikasikan pada tengah tahun 1990an.

5. *Timestamp*
   Timestamp adalah sebuah mekanisme untuk membedakan paket yang datang dari sesi yang lama dengan paket yang datang dari sesi komunikasi yang sedang dijalani. Cara kerjanya adalah dengan membandingkan waktu paket dikirim dengan waktu paket diterima, apabila rentang waktu-nya terlalu jauh, maka paket tersebut dikategorikan sebagai paket yang berasal dari sesi komunikasi yang lama. Kesulitan dalam mengaplikasikan timestamp adalah perlunya sinkronisasi waktu yang presisi antara pengirim dan penerima.

Dapat dilihat dari ke-lima mekanisme di atas, mekanisme-mekanisme tersebut tidak diterapkan pada saat yang bersamaan, atau dengan kata lain, terjadi penambahan mekanisme-mekanisme keamanan seiring dengan bertambahnya jenis-jenis serangan keamanan. Oleh karena itu, sesuai dengan apa yang disebutkan di atas, jaringan internet masa kini memiliki kelemahan dalam fleksibilitas dan menyebabkan berkurangnya tingkat efisiensi dalam melakukan penelitian keamanan jaringan.

Setelah mengetahui mekanisme keamanan jaringan yang ada pada Internet saat ini, akan ditentukan vulnerabilitas Internet terhadap sembilan serangan keamanan. Penjelasannya adalah sebagai berikut:

1. *Snooping attack*
   Snooping attack dapat diatasi dengan mekanisme IPSec. IPSec dapat mengenkripsi paket data, sehingga penyerang tidak akan bisa melihat isi di dalam paket data tersebut.

2. *Traffic analysis attack*
   Serangan ini diatasi dengan menggunakan mekanisme onion routing. Telah dijelaskan di atas bahwa *onion routing* mampu menciptakan suatu

komunikasi yang anonymous, sehingga penyerang tidak akan bisa menentukan lokasi di mana dia harus melakukan analasis trafik.

3. *Modification attack*
   Modification attack dapat diatasi dengan menggunakan *hash function* yang diberikan oleh TLS. Dengan *hash function* ini, penerima dapat mengecek apakah paket yang diterimanya mengalami perubahan atau tidak.

4. *Repudiation attack*
   Pada mekanisme PKI terdapat komponen *trusted third party*. Komponen ini dapat memberikan jaminan bahwa tidak ada user yang bisa menyangkal komunikasi telah terjadi.

5. *Denial-of-Service (DoS) attack*
   Dalam Internet terdapat mekanisme *flow control,* yang digunakan untuk mengatur batas dari paket yang bisa melintas pada jaringan. Apabila batasnya terlewati, contoh karena adanya paket-paket sampah yang memenuhi jaringan, maka paket-paket tersebut dapat dibuang, atau sesi komunikasinya dapat diputus.

6. *Man-in-the-middle attack*
   Serangan ini diatasi dengan kombinasi antara *digital signature* dengan PKI. Dengan kombinasi dua mekanisme tersebut, maka setiap user dapat menjamin bahwa lawan bicaranya sudah terautentikasi/legal.

7. *Reflection attack*
   Reflection attack memiliki basis serangan yang hampir sama dengan *man-in-the-middle attack*, oleh karena itu, kombinasi antara *digital signature* dan PKI yang diaplikasikan pada Internet dapat mengatasi serangan ini.

8. *Masquerading attack*
   Masquerading attack juga memiliki mekanisme yang hampir sama dengan *man-in-the-middle attack* dan *reflection attack,* oleh karena itu, kombinasi antara *digital signature* dan PKI yang diaplikasikan pada Internet dapat mengatasi serangan ini.

9. *Replaying attack*
   Internet mampu mengatasi serangan ini dengan adanya *timestamp* maupun *random number* (yang unik diberikan di setiap paket data). dengan mekanisme tersebut, user dapat membedakan paket data yang datang dari

sesi komunikasi yang sekarang dengan yang datang dari sesi komunikasi yang sebelumnya.

Tabel yang menyimpulkan vulnerabilitas keamanan pada jaringan internet masa kini adalah sebagai berikut:

Tabel 3. Vulnerabilitas Keamanan Pada Jaringan Internet Masa Kini

Security Attacks	**Mekanisme Keamanan**
Snooping	Proses enkripsi-dekripsi paket data oleh IPSec
Traffic Analysis	Mekanisme *onion routing*
Modification	Proses *hash function* yang diberikan TLS
Repudiation	Dengan *Trusted Third Party* yang ada pada PKI
Denial of Service	Menggunakan *Flow Control*
Man-in-the-Middle	PKI dan *Digital Signature*
Reflection	PKI dan *Digital Signature*
Masquerading	PKI dan *Digital Signature*
Replaying	*Timestamp*

Dari tabel terlihat bahwa jaringan internet masa kini mampu mengatasi ke-sembilan serangan kemanan jaringan. namun, mekanisme-mekanisme yang dibutuhkan tersebut hanya berlaku hingga saat ini, apabila di masa depan muncul serangan keamanan yang baru, maka mekanisme yang baru harus diaplikasikan. hal ini mengurangi tingkat fleksibilitas dan ke-efisienan jaringan internet.

6. KESIMPULAN

Beberapa kesimpulan yang dapat ditarik dari penelitian ini adalah sebagai berikut:

1. Jaringan internet masa kini telah mempunyai mekanisme-mekanisme keamanan untuk mengatasi ke-sembilan serangan keamanan jaringan yang diteliti, namun hal tersebut hanya berlaku saat ini saja. Apabila di masa depan muncul serangan keamanan yang baru, maka diperlukan mekanisme keamanan yang baru juga. Hal ini akan mengurangi tingkat fleksibilitas dari jaringan, dan tentu juga akan menyebabkan meningkatnya sumber daya manusia dan dana yang dibutuhkan untuk melakukan penelitian untuk menangani serangan keamanan yang baru tersebut.
2. Kedua arsitektur jaringan masa depan yang diteliti pada penelitian ini, SONATE dan NENA, belum mempunyai mekanisme-mekanisme

keamanan untuk mengatasi ke-sembilan serangan keamanan jaringan yang diteliti, namun hal ini tidak menjadi masalah dikarenakan kedua arsitektur jaringan tersebut menganut desain protocol blocks, di mana dalam menjalankan fungsi jaringannya, protokol-protokol yang sesuai akan dipilih berdasarkan persyaratan yang diinginkan.

3. Peneliti yang mengembangkan SONATE dan NENA dapat menggunakan hasil penelitian ini untuk mengembangkan mekanisme-mekanisme yang dibutuhkan untuk mengatasi serangan keamanan jaringan pada masing-masing jaringan. Sehingga diharapkan pada saat SONATE dan NENA diluncurkan ke publik, kedua jaringan tersebut telah benar-benar aman. Apabila keamanan jaringan dari kedua jaringan tersebut bisa terjamin levelnya, minimal memiliki level keamanan yang sama dengan jaringan internet masa kini, maka kedua jaringan tersebut memiliki potensi untuk menggantikan jaringan internet masa kini, hal ini dikarenakan SONATE and NENA memiliki tingkat fleksibilitas yang lebih tinggi dibandingkan jaringan internet masa kini, serta SONATE dan NENA memiliki mekanisme keamanan yang intrinsik.

REFERENCES

[1] Anja Feldmann. Internet clean-slate design: What and why?. In *SIGCOMM Computer Communication Review*, pages 59–64. Volume 37, Number 3. ACM, 2007.

[2] Rowan Klöti. Open flow: A security analysis. Master's thesis, Eidgenössische Technische Hochschule Zürich, 2013.

[3] Ltd Hangzhou H3C Technologies Co. Attack prevention technology white paper. Pages 1–13, 2008.

[4] http://technicafe.net/2013/05/what-is-dhcp-snooping.html

[5] George Danezis. Technical report. introducing traffic analysis: Attacks, defences and public policy issues. pages 1–25, 2005.

[6] http://codeidol.com/community/security/passive-attacks/22774/

[7] Emmett Dulaney. *CompTIA Security+ Study Guide*. Wiley, Indianapolis, 4th edition, 2009.

[8] http://wiki.olc.edu/index.php/Active_attacks

[9] http://flylib.com/books/en/4.154.1.31/1/

[10] Jelena Mirkovic, Sven Dietrich, David Dittrich, and Peter Reiher. *Internet Denial of Service: Attack and Defense Mechanisms*. Prentice Hall, 2005.

[11] http://ardyutomo25.blogspot.com/2012/09/serangan-dos-dan-cara-mengatasinya.html

[12] http://www.passwindow.com/security.html

[13] Ling Dong and Kefei Chen. *Cryptographic Protocol: Security Analysis Based on Trusted Freshness*. Springer, 2012.

[14] Hamid Jahankhani, David Lilburn Watson, Gianluigi Me, and Frank Leonhardt. *Handbook of Electronic Security and Digital Forensics*. 2010.

[15] Hannes Gredler and Walter Goralski. *The Complete IS-IS Routing Protocol*. Springer, 2004.

[16] Rahamatullah Khondoker. Service composition and selection in service oriented network architecture. In *Research Seminar, Institute of Telematics, Karlsruhe Institute of Technology, Germany*, pages 1–44, 2010.

[17] Paul Müller, Bernd Reuther, and Markus Hillenbrand. Future internet: A service oriented approach - sonate. In *Würzburg Workshop on Visions of Future Generation Networks (EuroView2007)*, pages 1–35, 2007.

[18] Rahamatullah Khondoker, Abbas Siddiqui, Bernd Reuther, and Paul Müller. Service orientation paradigm in future network architectures. In *Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS-2012)*, pages 346–351, 2012.

[19] Denis Martin, Lars Völker, and Martina Zitterbart. A flexible framework for future internet design, assessment, and operation. *Journal Computer Networks: The International Journal of Computer and Telecommunications Networking*, pages 910–918. Volume 55 Issue 4, March 2010.

[20] Hans Wippel and Oliver Hanka. Deployment of application-tailored protocols in future networks. In *11th Würzburg Workshop on IP: Joint ITG and Euro-NF Workshop "Visions of Future Generation Networks" (EuroView2011)*, pages 1–2, 2011.

[21] Thomas Gamer and Hans Wippel. A collaborative attack detection and its challenges in the future internet. In *Proceedings of the Joint ITG, ITC, and Euro-NF Workshop "Visions of Future Generation Networks" (EuroView)*, pages 1–2, 2010.

[22] http://en.wikipedia.org/wiki/Onion_routing