Analisis Perbandingan Performansi Deep Packet Inspection Firewall Antara L7-Filter dan nDPI

Deep Packet Inspection Firewall Performance Comparison Analysis between L7-Filter and nDPI

Faizal Eko Nugroho[1], Gandeva bayu Satrya, ST., MT.[2], Tri Brotoharsono, Ir., MT.[3]

[1,2,3] Prodi S1 Teknik Informatika, Fakultas Informatika, Universitas Telkom
Jl. Telekomunikasi, Dayeuh Kolot, Bandung 40257 Telp. (62-22) 7564108 ext. 2333

faizal.eko@students.telkomuniversity.ac.id[1], gbs@telkomuniversity.ac.id[2], tribrotoharsono@telkomuniversity.ac.id[3]

ABSTRAK

Pengklasifikasian trafik data pada firewall tidak cukup bila hanya menggunakan parameter seperti nomor port. Saat ini banyak aplikasi yang bisa digunakan tanpa harus memakai nomor port yang seharusnya aplikasi tersebut gunakan. Selain itu, pengaturan akses pada web aplikasi menggunakan firewall juga tidak bisa bila hanya digunakan tiga parameter yang disebutkan di awal.

Metode deep packet inspection bisa dipakai pada firewall sebagai metode pengklasifikasian untuk digunakan dalam pengaturan akses trafik data. L7-Filter dan nDPI merupakan deep packet inspection library yang dapat digabungkan dengan aplikasi firewall. Tingkat kualitas L7-Filter dan nDPI sebagai pengklasifikasian data pada firewall akan diukur pada tugas akhir ini.

Berdasarkan hasil pengujian pada tugas akhir ini, nDPI memiliki performansi yang lebih bagus daripada L7-Filter, dalam hal nilai sensitivitas dan nilai spesifisitas. Nilai sensitivitas rule firewall yang menggunakan nDPI lebih tinggi 2,1% bila digunakan untuk menghentikan akses dan lebih tinggi 2.31% bila digunakan untuk menerima akses layanan, dibandingkan nilai sensitivitas L7-Filter. Sementara nilai spesifisitas rule firewall dengan nDPI lebih tinggi 2.26% saat digunakan untuk menghentikan layanan, dan lebih tinggi 6.66% jika digunakan untuk menerima akses layanan. Meskipun demikian, rule firewall yang menggunakan L7- Filter memiliki waktu pengeksekusian rata-rata yang lebih cepat 0.0298 ms dibandingkan nDPI.

Kata Kunci : pengklasifikasian paket data, firewall, deep packet inspection, L7-Filter, nDPI.

ABSTRACT

Traffic data classification for access management in firewall is insufficient when using parameter like port number. Nowadays, many applications can be properly executed without using its standard port. Web application blocking access using firewall, can't be done using ip address as the only classification parameter.

Deep packet inspection methods can be used in firewall as packet data classifier for traffic data access management. L7-Filter and nDPI are deep packet inspection library which can be combined with firewall, as packet data classifier. The classification quality of L7-Filter and nDPI as firewall will be tested in this thesis (final task).

According to L7-Filter and nDPI testing scenario result in this thesis, nDPI has better performance in firewall rule sensitivity and specificity. nDPI's sensitivity is better than L7-Filter's, 2.1% when used for blocking access and 2.31% when used for accepting access. nDPI's specificity is better than L7-Filter's, 2.26% when used for blocking access and 6.66% when used for accepting access. The L7-Filter firewall rule execution time, however, is faster 0.0298 ms than nDPI.

Key word : packet data classification, firewall, deep packet inspection, L7-Filter, nDPI.

1. Latar Belakang

Saat ini, beragam jenis trafik data di Internet menyulitkan proses klasifikasi trafik untuk digunakan di firewall. Tidak jarang terdapat aplikasi Internet yang menggunakan port yang sama dengan yang digunakan service standar, seperti aplikasi Skype yang menggunakan port 80, sama dengan http. Selain itu, aplikasi peer-to-peer sharing seperti BitTorrent client bisa menggunakan port yang tidak sama dengan port yang digunakan sebagai port standar BitTorrent client. Begitu juga untuk pembatasan akses pada aplikasi web tertentu, yang tidak bisa dilakukan hanya dengan menggunakan firewall dengan menghentikan akses pada alamat ip web tersebut.

Deep packet inspection (DPI) adalah salah satu solusi yang bisa digunakan untuk mengklasifikasikan trafik data pada firewall. DPI bekerja dengan memeriksa paket hingga application layer pada OSI layer model, untuk mendapatkan informasi jenis trafik data paket tersebut. Hal ini bisa mengatasi masalah klasifikasi data pada stateful firewall yang dapat mengklasifikasikan trafik data hingga berdasarkan nomor port yang digunakan. Selain itu, metode DPI lebih banyak dipakai dibandingkan metode yang sama waktu munclnya, yaitu metode statistikal, disebabkan keterbatasan dan perlunya proses pelatihan metode tersebut [1]. Selain itu, penggunaan DPI pada firewall sendiri sudah dimanfaatkan untuk firewall yang dibuat oleh berbagai perusahaan besar seperti SonicWall [2] dan Cisco. Pada aplikasi open source firewall, juga bisa diterapkan DPI sebagai metode pengklasifikasian data, sebagai contohnya dengan menambahkan nDPI atau L7-Filter pada aplikasi firewall Netfilter/Iptables.

nDPI dan L7-Filter adalah open source DPI library. Keduanya ini bisa digabungkan dengan Netfilter/Iptables pada Linux untuk membuat DPI firewall. Diharapkan dengan ini, nDPI firewall atau L7-Filter firewall bisa digunakan untuk mengatur akses terhadap web aplikasi tertentu, maupun aplikasi seperti BitTorrent client yang bisa berfungsi meskipun diatur untuk menggunakan port yang tidak sama dengan ketentuan untuk port BitTorrent.

2. Dasar Teori

2.1. Deep Packet Inspection Firewall

Firewall adalah sistem, baik perangkat keras ataupun perangkat lunak, yang mengendalikan alur trafik data masuk dan keluar jaringan dengan menganalisa paket data dalam trafik lalu membolehkan atau memblokir paket data tersebut berdasarkan aturan yang sudah ada.

Perkembangan firewall dapat dibedakan menjadi tiga generasi. Pada generasi awal, mulai tahun 1980, firewall menyediakan filterisasi paket data berdasarkan kriteria seperti port, protokol, dan alamat MAC/IP. Generasi setelahnya, sejak tahun 1990, muncul stateful packet inspection firewall, yang bisa mengklasifikasikan trafik berdasarkan state yang ada pada trafik masuk atau keluar yang dibandingkan dengan table state. Pada generasi ini, firewall bekerja pada OSI model layer 2, 3 dan 4. Generasi ketiga, yaitu generasi sejak tahun 2000, adalah era firewall memiliki kemampuan lebih banyak dengan cakupan lebih luas, mulai dari deep packet inspection (DPI) pada seluruh trafik data, pencegahan intrusi, deteksi malware, trafik analisis, pengendali aplikasi, IPSec dan SSL VPN. Singkat kata, pada generasi ini, firewall bekerja pada OSI model layer 2, 3, 4, serta 7, yang tidak hanya bisa mempertahankan sistem dari penyerangan, tetapi juga filterisasi konten trafik data. [2]

Deep Packet Inspection (DPI) adalah teknologi untuk memeriksa dan menganalisa sebagian paket dari keseluruhan paket dalam sebuah trafik, hingga pada bagian data paket di dalam frame TCP/UDP, serta mengumpulkan dan mengambil tindakan berdasarkan informasi yang dikumpulkan tersebut. Contoh penggunaan data yang dikumpulkan ini, selain untuk proses klasifikasi, DPI bisa digunakan untuk sistem ketahanan suatu jaringan, terbukti dengan adanya network-based DPI yang bisa mengidentifikasi serangan Denial of Service dan malware worms. Begitu pula terdapat penggunaan DPI pada firewall, seperti yang telah dikembangkan oleh perusahaan seperti SonicWall dan komunitas open source seperti L7-Filter. [3]

2.4.1. L7-Filter

L7-Filter adalah *packet classifier* pada Linux, pada awalnya dibuat sebagai modul tambahan Netfilter untuk mendeteksi trafik data aplikasi *peer-to-peer sharing*. *Library* ini menggunakan proses *regular expression matching* untuk pencocokan antara protokol yang digunakan dengan data pada *application layer* di OSI model. [1]

Pengembangan L7-Filter ini mulai pada tahun 2003, akibat kebanyakan aplikasi untuk mengendalikan jumlah *bandwidth* yang digunakan pada suatu protokol adalah berbayar dan mahal. Kemudian, pada tahun 2005, dikembangkan versi yang bisa digabungkan dengan kernel Linux dan Netfilter. Setelah itu, pada tahun 2006, dikembangkan versi yang dapat berjalan tanpa harus digabungkan dengan kernel Linux. Saat ini, pengembangan L7-Filter diambil alih oleh ClearFoundation, dan versi terbaru L7-Filter adalah versi 2.23, yang dirilis pada bulan September 2013 [2].

2.4.2 nDPI

nDPI adalah DPI *library* yang dikembangkan oleh Ntop. nDPI sendiri dikembangkan dari OpenDPI, yang mana OpenDPI merupakan versi *open source* DPI *library* yang dibuat oleh Ipoque untuk *firewall*. Pada nDPI digunakan algoritma *string matching* Aho-Corasick, untuk proses pencocokan paket data dengan *signature* yang dimiliki nDPI [3].

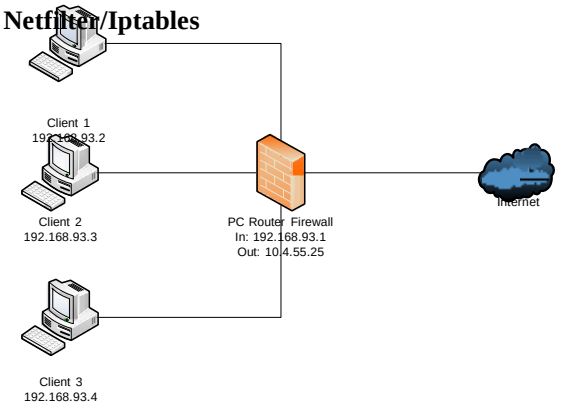
Beberapa kemampuan nDPI antara lain [4]:

1) Mendukung pendeteksian lebih dari 170 protokol

2) Memiliki dekoder sertifikat SSL, sehingga bisa mendukung pendeteksian pada koneksi terenkripsi
3) Memiliki kemampuan untuk mendukung sub-protokol menggunakan proses *string-based* matching

3. Pembahasan

3.1. L7-Filter dan nDPI pada Netfilter/Iptables



Gambar 0-1 Rancangan Jaringan

L7-Filter dan nDPI akan digabungkan dengan Netfilter/Iptables sebagai modul untuk proses pencocokan dengan metode *deep packet inspection*. Keduanya akan digunakan pada skenario penghentian akses dan penerimaan akses layanan data tertentu. L7-Filter bisa digabungkan langsung dengan Netfilter/Iptables, sementara nDPI membutuhkan aplikasi tambahan, yaitu ndpi-netfilter buatan *user*github ewildgoose

3.2 Skenario Pengujian

Pada bagian ini akan dibahas mengenai penentuan cara akses layanan yang akan dijalankan untuk menguji nDPI firewall dan L7-Filter firewall. Kedua firewall ini akan digunakan sebagai berikut:

a. Penghentian akses
   Firewall yang diuji akan ditambahkan rule untuk menerima semua akses kecuali pada akses layanan trafik data yang sedang diujikan
b. Penerimaan akses
   Firewall yang diuji akan ditambahkan rule untuk menolak semua akses kecuali pada akses layanan trafik data yang sedang diujikan

Setiap skenario akses ini dilakukan dalam setiap komputer client yang dihubungkan dengan firewall. Dalam setiap skenario juga akan ditambahkan trafik data yang seharusnya tidak akan dimasukan ke dalam rule firewall yang sedang diujikan. Selain itu, setiap skenario ini akan dijalankan dalam waktu sepuluh menit dan dilakukan selama lima kali.

3.2.1 Skenario Akses BitTorrent

Pada skenario ini akan dilakukan pengujian terhadap pengaksesan trafik data BitTorrent. Sebelum rule firewall diberlakukan, dalam aplikasi BitTorrent client sudah melakukan download torrent. Kemudian, setelah rule firewall diterapkan, ditambahkan download torrent baru.

Sebagai tambahan trafik data pada skenario ini, diakses juga aplikasi Soulseek client setelah rule firewall ditambahkan, dan dilakukan download file dari jaringan Soulseek.

3.2.2 Skenario Akses BitTorrent Terenkripsi

Pada skenario ini, sama seperti skenario akses BitTorrent, sebelum rule firewall diberlakukan, dalam aplikasi BitTorrent client sudah melakukan download torrent. Kemudian, setelah rule firewall diterapkan, ditambahkan download torrent baru. Hanya saja, pada skenario ini mode enkripsi protocol BitTorrent di aplikasi BitTorrent client diaktifkan.

Sebagai tambahan trafik data pada skenario ini, diakses juga aplikasi Soulseek client setelah rule firewall ditambahkan, dan dilakukan download file dari jaringan Soulseek.

3.2.3 Skenario Akses eDonkey

Pada skenario ini, sebelum rule firewall diberlakukan, dalam aplikasi eMule sudah melakukan akses ke jaringan eDonkey, dan melakukan download file. Kemudian, setelah rule firewall diterapkan, ditambahkan proses download file yang baru.

Pada skenario ini, diakses juga aplikasi Soulseek client setelah rule firewall ditambahkan, sebagai trafik data tambahan yang seharusnya tidak masuk dalam rule firewall, Melalui aplikasi ini, dilakukan download file dari jaringan Soulseek.

3.2.4 Skenario Akses eDonkey Obfuscated Mode

Pada skenario ini, sebelum rule firewall diberlakukan, dalam aplikasi eMule sudah melakukan akses ke jaringan eDonkey, dan melakukan download file. Kemudian, setelah rule firewall diterapkan, ditambahkan proses download file yang baru. Dalam skenario ini, mode penyembunyian protokol eDonkey, atau disebut juga obfuscated mode pada eMule diaktifkan.

Pada skenario ini, sebagai sumber trafik data tambahan, digunakan Soulseek client setelah rule firewall ditambahkan, dan dilakukan download file dari jaringan Soulseek.

3.2.5 Skenario Akses Skype

Pada skenario ini, sebelum rule firewall diberlakukan, aplikasi Skype sudah aktif, user dalam keadaan login, serta melakukan chatting melalui aplikasi ini. Setelah lima menit dari diberlakukannya rule firewall, aplikasi Skype dihentikan kemudian diaktifkan kembali, user melakukan login, lalu digunakan untuk chatting kembali.

Pada skenario ini, aplikasi yang digunakan sebagai tambahan trafik data yang seharusnya tidak dimasukan ke dalam rule firewall yang diujikan adalah Yahoo Messenger. Aplikasi ini akan digunakan untuk mengirimkan pesan dan chatting selama pengujian berlangsung.

3.2.6 Skenario Akses Yahoo Messenger

Pada skenario ini, sebelum rule firewall diberlakukan, aplikasi Yahoo Messenger sudah aktif, user dalam keadaan login, serta melakukan chatting melalui aplikasi ini. Lima menit setelah rule firewall diterapkan, aplikasi Yahoo Messenger dihentikan kemudian diaktifkan kembali, user melakukan login, lalu digunakan untuk chatting kembali.

Pada skenario ini, aplikasi yang digunakan sebagai tambahan trafik data yang seharusnya tidak dimasukan ke dalam rule firewall yang diujikan adalah Skype. Aplikasi ini akan digunakan untuk mengirimkan pesan dan chatting selama pengujian berlangsung.

3.2.7 Skenario Akses Facebook

Pada skenario ini, sebelum rule firewall diberlakukan, sudah dilakukan login ke dalam situs web Facebook (http://www.facebook.com), dan melakukan aktivitas terkait aplikasi web Facebook, seperti melihat notifikasi, setelah rule diterapkan. Lima menit setelah rule firewall diterapkan, web browser yang digunakan untuk mengakses web Facebook ditutup, kemudian dibuka kembali untuk user login ke dalam Facebook dan melakukan aktivitas seperti melihat notifikasi Facebook.

Pada skenario ini, aplikasi yang digunakan sebagai tambahan trafik data yang seharusnya tidak dimasukan ke dalam rule firewall yang diujikan adalah trafik data akses ke situs web Reddit (http://www.reddit.com). Pengaksesan pada Reddit ini meliputi mengakses salah satu link dan mengakses Reddit melalui url http://www.reddit.com/search?q=facebook.com

3.2.8 Skenario Akses Twitter

Pada skenario ini, sudah dilakukan login ke dalam web Twitter (http://www.twitter.com) sebelum rule firewall diberlakukan, dan melakukan aktivitas terkait, seperti melihat timeline Twitter, setelah rule diberlakukan. Lima menit setelah rule firewall diterapkan, web browser yang digunakan untuk mengakses web Twitter ditutup, kemudian dibuka kembali untuk user login ke dalam Twitter dan melakukan aktivitas terkait aplikasi web Twitter kembali, seperti melihat timeline.

Pada skenario ini, aplikasi yang digunakan sebagai tambahan trafik data yang adalah trafik data akses ke situs web Reddit (http://www.reddit.com). Pengaksesan pada Reddit ini meliputi mengakses salah satu link dan mengakses Reddit melalui url http://www.reddit.com/search?q=twitter.com

3.2.9 Skenario Akses Google

Pada skenario ini, sebelum rule firewall diberlakukan, sudah dilakukan login ke dalam akun Google (http://accounts.google.com), dan mengakses Google+ serta pencarian Google setelah rule diaktifkan. Lima

menit setelah rule firewall diterapkan, web browser yang digunakan untuk mengakses web Google ditutup, setelah sebelumnya dilakukan logout, kemudian dibuka kembali untuk user login ke dalam akun Google dan melakukan aktivitas seperti melihat profil akun Google+.

Pada skenario ini, aplikasi yang digunakan sebagai tambahan trafik data yang seharusnya tidak dimasukan ke dalam rule firewall yang diujikan adalah trafik data akses ke situs web Reddit (http://www.reddit.com). Pengaksesan pada Reddit ini meliputi mengakses salah satu link dan mengakses Reddit melalui url http://www.reddit.com/search?q=google.com

3.2.10 Skenario Akses Youtube

Pada skenario ini, sebelum rule firewall diberlakukan, sudah login ke dalam situs web Youtube (http://www.youtube.com), dan melakukan streaming salah satu video Youtube. Lima menit setelah rule firewall diterapkan, web browser yang digunakan untuk mengakses Youtube ditutup, kemudian dibuka kembali untuk mengakses Youtube dan melakukan streaming salah satu video Youtube.

Pada skenario ini, aplikasi yang digunakan sebagai tambahan trafik data yang seharusnya tidak dimasukan ke dalam rule firewall yang diujikan adalah trafik data akses ke situs web Reddit (http://www.reddit.com). Pengaksesan pada Reddit ini meliputi mengakses salah satu link dan mengakses Reddit melalui url http://www.reddit.com/search?q=youtube.com

3.2.11 Skenario Akses Youtube (https)

Pada skenario ini, sebelum rule firewall diberlakukan, sudah login ke dalam situs web Youtube (https://www.youtube.com), dan melakukan streaming salah satu video Youtube. Lima menit setelah rule firewall diterapkan, web browser yang digunakan untuk mengakses Youtube ditutup, kemudian dibuka kembali untuk mengakses Youtube (https://www.youtube.com) dan melakukan streaming salah satu video Youtube.

Pada skenario ini, aplikasi yang digunakan sebagai tambahan trafik data yang seharusnya tidak dimasukan ke dalam rule firewall yang diujikan adalah trafik data akses ke situs web Reddit (http://www.reddit.com). Pengaksesan pada Reddit ini meliputi mengakses salah satu link dan mengakses Reddit melalui url http://www.reddit.com/search?q=youtube.com

Berdasarkan data mengenai paket trafik data yang diujikan dalam suatu skenario akses, dapat dihitung nilai sensitivitas dan spesifisitas *firewall* pada skenario tersebut, serta penghitungan lama waktu eksekusi *rule firewall* yang menggunakan *deep packet inspection* dalam proses pengklasifikasian paket data.

a. True Positive
   Paket data layanan yang harus dimasukan pada rule firewall yang menggunakan deep packet inspection, dan digolongkan dalam rule firewall tersebut
b. True Negative
   Paket data layanan yang harus tidak dimasukan pada rule firewall yang menggunakan deep packet inspection, dan tidak digolongkan dalam rule firewall tersebut
c. False Positive
   Paket data layanan yang harus tidak dimasukan pada rule firewall yang menggunakan deep packet inspection, dan digolongkan dalam rule firewall tersebut
d. False Negative
   Paket data layanan yang harus dimasukan pada rule firewall yang menggunakan deep packet inspection, dan tidak digolongkan dalam rule firewall tersebut

Penghitungan nilai sensitivitas didefinisikan sebagai berikut

$$\frac{\sum}{\sum}$$

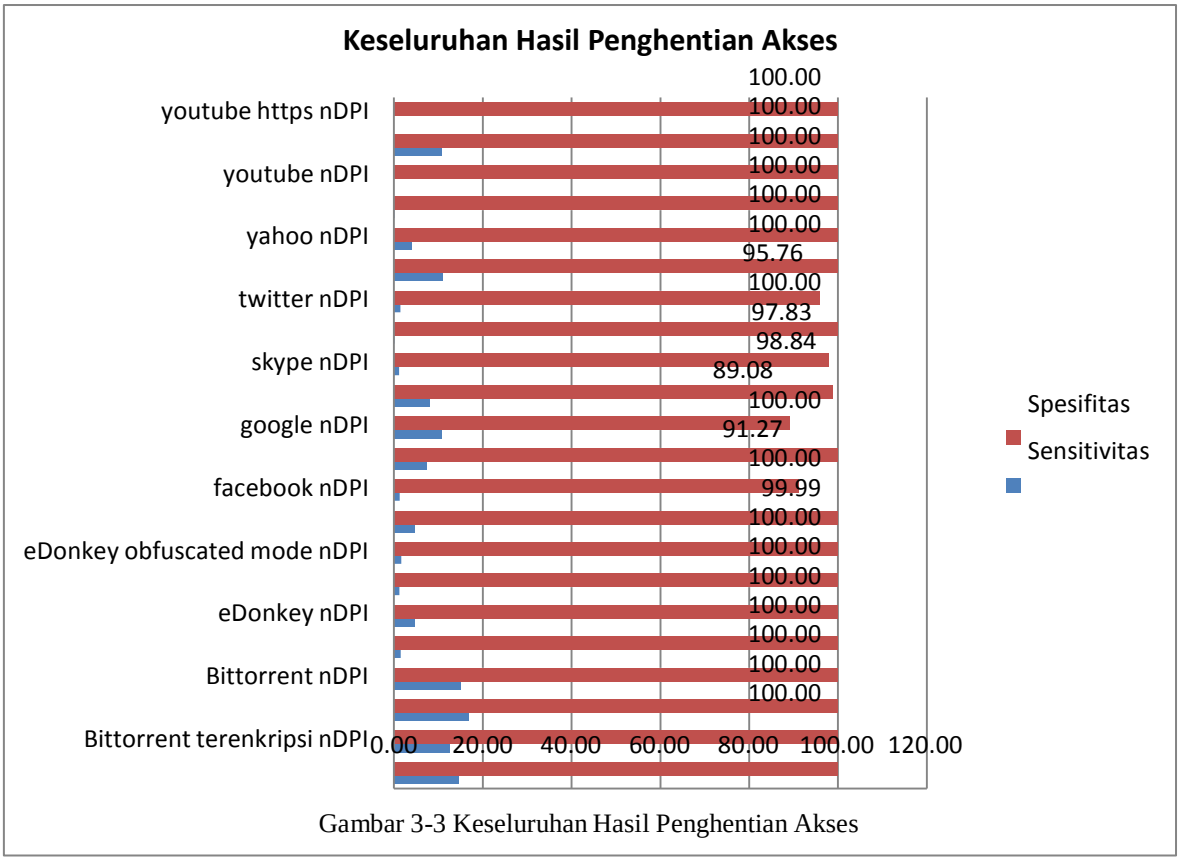
Penghitungan nilai spesifisitas didefinisikan sebagai berikut

$$\frac{\sum}{\sum}$$

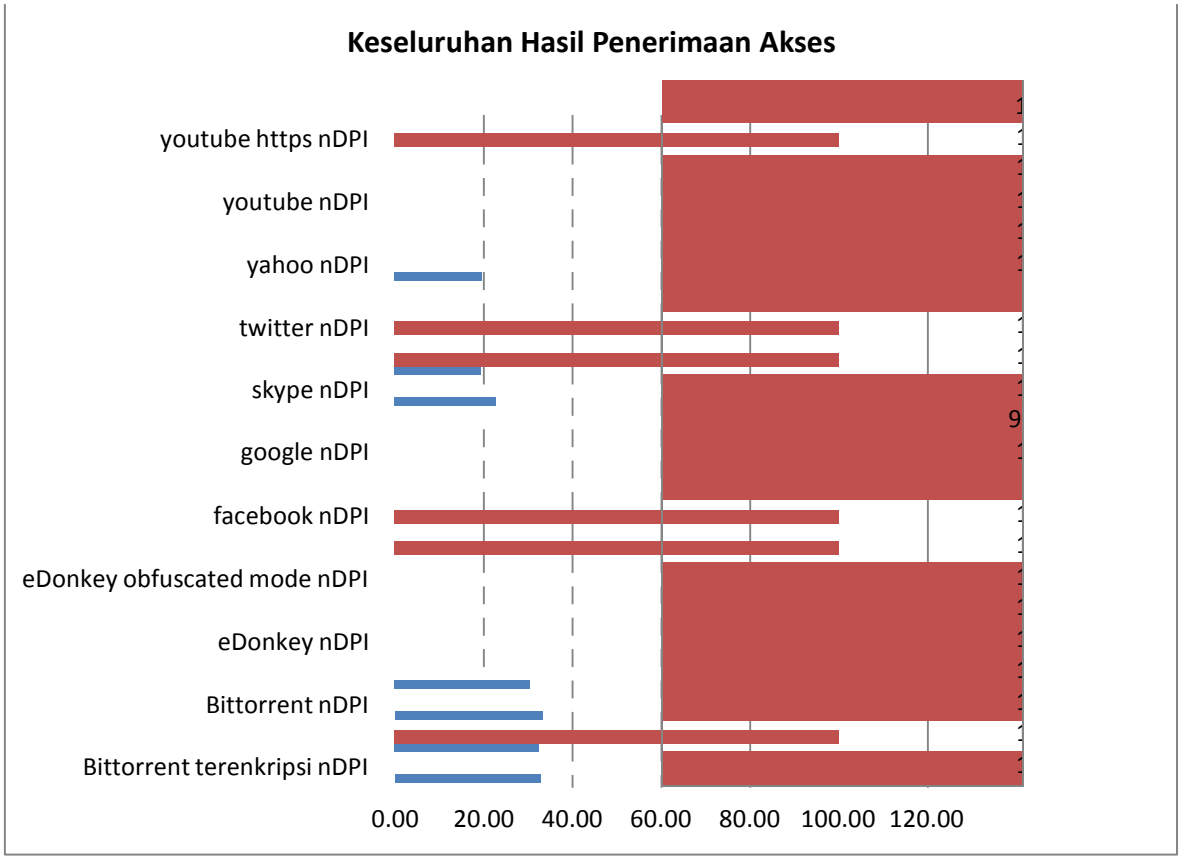
Penghitungan waktu suatu paket diperiksa dalam *rule firewall* yang menggunakan *deep packet inspection* (DPI) pada nDPI *firewall* dan L7-Filter *firewall*, yaitu dengan waktu selesai diperiksa *rule firewall* dengan DPI dan waktu selesai diperiksa *rule* sebelum *rule* dengan DPI.

3.3 Analisis Keseluruhan Pengujian

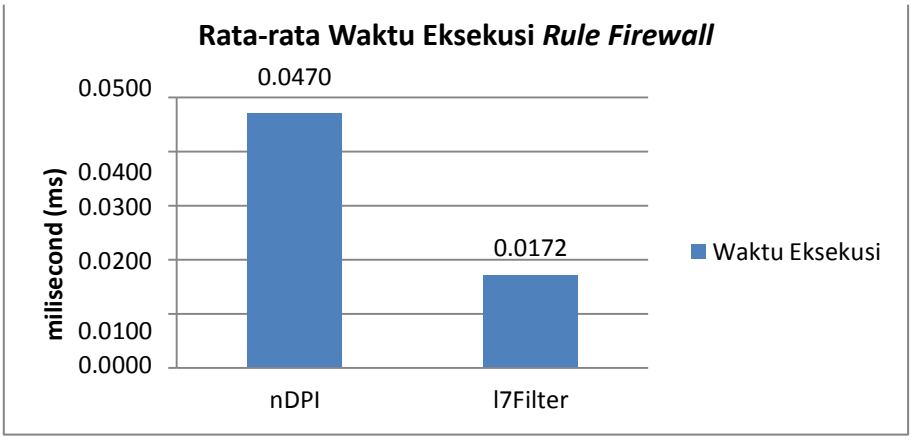
Berdasarkan semua data hasil pengujian skenario akses, dapat dilihat keunggulan nDPI firewall maupun L7-Filter firewall pada data nilai sensitivitas dan spesifisitas setiap skenario.



Gambar 3-3 Keseluruhan Hasil Penghentian Akses



Gambar 3-4 Keseluruhan Hasil Penerimaan Akses



Gambar 3-5 Rata-rata Waktu Eksekusi Rule Firewall

Pada gambar 3-3 dan gambar 3-4 dapat diperhatikan bahwa nDPI firewall memiliki lebih banyak nilai sensitivitas dan spesifisitas yang lebih tinggi daripada L7-Filter firewall. Hal ini disebabkan, meskipun kedua firewall mempunyai dasar cara pengklasifikasian yang sama, mencocokan data dalam paket dengan data signature paket yang dimiliki, data signature yang ada pada nDPI lebih banyak yang sesuai dengan paket data pada setiap skenario akses. Selain itu, hal yang membuat nDPI lebih unggul dalam nilai spesifisitas adalah kesalahan pengklasifikasian paket data pada L7-Filter. Sebagai buktinya adalah dihentikannya akses ke situs web dengan url http://www.reddit.com/search?q=facebook.com saat pengujian penghentian akses Facebook. Meskipun demikian, signature paket data yang dimiliki L7-Filter untuk trafik data eDonkey lebih sesuai

daripada nDPI, berdasarkan data hasil pengujian skenario akses eDonkey dan akses eDonkey obfuscated mode.

Data sensitivitas skenario akses BitTorrent tidak dienkripsi lebih tinggi daripada pada skenario akses BitTorrent terenkripsi, begitu pula pada skenario akses eDonkey tanpa obfuscated mode dan pada skenario akses eDonkey obfuscated mode. Hal ini disebabkan proses enkripsi paket data pada suatu protokol serta proses penyembunyian protokol pada obfuscated mode, menyebabkan paket data yang menggunakan protokol tersebut tidak sesuai dengan data signature yang ada pada nDPI maupun L7-Filter. Meskipun demikian, pada data sensitivitas skenario akses Youtube lebih rendah daripada Youtube (https) pada nDPI firewall, sebab, signature pada nDPI untuk Youtube, didesain untuk lebih mengenali pengaksesan situs Youtube dengan https.

Pada data sensitivitas setiap hasil pengujian skenario, nilai yang didapat bernilai kurang dari 50%. Dengan kata lain, lebih banyak paket data yang seharusnya dideteksi dan ditindak, tetapi tidak terdeteksi oleh nDPI firewall dan L7-Filter firewall. Hal ini disebabkan dalam sistem nDPI maupun L7-Filter apabila terdapat sejumlah paket data pada flow paket data, dengan batasan jumlah paket data pada sistem nDPI dan L7-Filter berbeda, dalam suatu flow dideteksi termasuk sebuah kategori, maka seluruh flow paket data tersebut akan digolongkan dalam kategori yang sama. Dengan demikian, apabila ada paket data yang seharusnya diteruskan firewall, bila paket tersebut tidak diteruskan oleh nDPI firewall atau L7-Filter firewall, maka paket data lain yang dalam satu flow tidak akan diteruskan. Begitu pula jika terdapat paket data yang seharusnya dihentikan dalam skenario penghentian akses tetapi tidak dihentikan, maka paket data lain yang berada dalam satu flow dengan paket data tersebut tidak akan dihentikan, dan hal ini menyebabkan pula jumlah paket data yang seharusnya dihentikan lebih banyak sehingga nilai sensitivitas lebih sedikit daripada pada skenario penerimaan akses.

Bila diperhatikan pada data lama pengeksekusian rule firewall, nDPI firewall memiliki nilai rata-rata pengeksekusian yang lebih lama dibandingkan pengeksekusian rule pada L7-Filter firewall. Hal ini dikarenakan proses pengklasifikasian pada nDPI tidak hanya menggunakan perbandingan signature paket data, tetapi juga pemeriksaan panjang payload pada suatu paket, atau ukuran suatu field, tergantung pada fungsi pencarian jenis paket yang sedang dieksekusi. Selain itu, hal ini juga menjadi faktor lain penyebab nilai sensitivitas dan nilai spesifisitas keseluruhan untuk nDPI firewall lebih tinggi dibandingkan L7-Filter firewall.

4 Kesimpulan

Bedasarkan tujuan serta hasil pengujian dan analisis yang telah dilakukan pada penggunaan nDPI dan L7-Filter sebagai deep packet inspection firewall, maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Nilai sensitivitas rule firewall yang menggunakan nDPI lebih tinggi 2,1% bila digunakan untuk menghentikan akses dan lebih tinggi 2,31% bila digunakan untuk menerima akses layanan, dibandingkan nilai sensitivitas L7-Filter.
2. Nilai spesifisitas rule firewall dengan nDPI lebih tinggi 2,26% saat digunakan untuk menghentikan layanan, dan lebih tinggi 6,66% jika digunakan untuk menerima akses layanan.
3. Waktu pengeksekusian rule firewall yang menggunakan L7-Filter memiliki waktu pengeksekusian rata-rata yang lebih cepat 0,0298 ms dibandingkan nDPI

DAFTAR PUSTAKA

[1] ntop. (2013). *nDPI - Quick Start Guide.* ntop.

[2] Thomason, S. (2012). *Improving Network Security: Next Generation Firewalls and Advanced Packet Inspection Devices.* Global Journals Inc.

[3] Ayoub, D. (2009). *Why Protection and Performance Matter : The Benefits of Multi-core Reassembly-Free Deep Packet Inspection.* SonicWall.

[4] Bujlow, T., & Carela, V. (2013). *Comparison of Deep Packet Inspection (DPI) Tools for Traffic Classification.* Barcelona: Universitat Politecnica De Catalunya.

[5] Gheorghe, L. (2006). *Designing and Implementing Linux Firewalls and QoS using netfilter, iproute2, NAT, and L7-filter.* Birmingham: Packt Publishing.

[6] Alcock, S., & Nelson, R. (2012). *Measuring the Accuracy of Open-Source Payload-Based Traffic Classifiers Using Popular Internet Applications.* Hamilton: University of Waikato.

[7] Bujlow, T., Carela-Espanol, V., & Barlet-Ros, P. (2014). *Extended Independent Comparison of Popular Deep Packet Inspection (DPI) Tools for Traffic Classification.* Barcelona: Universitat Politècnica de Catalunya.

[8] Scarfone, K., & Hoffman, P. (2009). *Guidelines on Firewalls and Firewall Policy.* Gaithersburg: National Institute of Standards and Technology.

[9] Ou, G. (2009). *Understanding Deep Packet Inspection (DPI) Technology.* Digital Soceity.

[10] Deri, L., & Maurizio, M. (2014). *nDPI: Open-Source High-Speed Deep Packet Inspection.* Pisa: Institute of Informatics and Telematics.

[11] Cisco. (2010). *WAN and Application Optimization Solution Guide.* Cisco.

[12] Radisys. (2010). *DPI: Deep Packet Inspection Motivations, Technology, and Approaches for Improving, Broadband Service Provider ROI.* Radisys White Paper.

[13] Bober, A. *Introduction to Layer 7-Filter.*

[14] L7-Filter. (2008). *L7-filter Kernel Version HOWTO.* http://l7-filter.sourceforge.net/HOWTO-kernel.

[15] Zhu, W., & Zeng, N. (2010). *Sensitivity, Specificity, Accuracy, Associated Confidence Interval and ROC Analysis with Practical SAS® Implementations.* Fort Washington: K&L consulting services, Inc.

[16] Wildgoose, E. *ndpi-netfilter.* https://github.com/ewildgoose/ndpi-netfilter.