Indonesian Journal of Computer Science
ISSN 2302-4364 (*print*) dan 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam, No. 1, Padang, Indonesia, Telp. (0751) 7056199, 7058325
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

Deteksi *Malware* Dridex Menggunakan *Signature-based Snort*

Adhitya Nugraha, Dinda Aulia Gustian
adhitya@dsn.dinus.ac.id, daulia810@gmail.com
Universitas Dian Nuswantoro Semarang

Informasi Artikel

Diterima : 10-02-2021
Direview : 20-02-2021
Disetujui : 13-03-2021

Kata Kunci

dridex;*malware*; IDS; snort; berbasis *signature*

Abstrak

Saat ini *malware* merupakan aplikasi berbahaya dan terus berkembang sehingga menjadi ancaman pada saat menggunakan layanan internet. Salah satu *malware* yang sangat berbahaya pada 2020 ini adalah Dridex yang menargetkan dan mencuri data kredensial perbankan serta informasi pribadi mengenai catatan keuangan seseorang. Dridex memanfaatkan spam email dan *social engineering* untuk penyebarannya. Tercatat bahwa *malware* ini telah membuat kerugian mencapai $100 juta. Penelitian ini berfokus untuk melakukan analisis terhadap aktivitas Dridex melalui dataset lalu lintas jaringan dan kemudian mengembangkan *rules snort* berdasarkan *signature* Dridex yang telah ditemukan. Penelitian ini telah mengembangkan 12 (dua belas) rules yang diimplementasikan pada Snort untuk mendeteksi keberadaan *signature* Dridex. Pengujian keberhasilan deteksi Dridex dilakukan menggunakan teknik confusion matrix dan menghasilkan nilai akurasi sebesar 88,5%, nilai *recall* atau *decision rate* sebesar 100%, dan nilai presisi sebesar 84,75%.

Keywords

dridex;malware;ids;snort; signature-based

Abstract

Currently malware is a dangerous application and continues to grow so that it becomes a threat when using internet services. One of the most dangerous malware in 2020 is Dridex which targets and steals banking credentials and personal information regarding a person's financial records. Dridex makes use of email spam and social engineering for its distribution. It is noted that this malware has made a loss of up to $100 million. This study focuses on analyzing Dridex activity through a network traffic dataset and then developing snort rules based on the Dridex signatures that have been found. This study has developed 12 (twelve) rules that are implemented on Snort to detect the presence of Dridex signatures. Testing the success of Dridex detection was carried out using confusion matrix techniques and resulted in an accuracy value of 88.5%, a recall or decision rate of 100%, and a precision value of 84.75%.

A. Pendahuluan

Internet merupakan suatu jaringan global yang menghubungkan berbagai perangkat komputer dari berbagai macam jenis dan tipe. Di dalam internet terjadi berbagai interaksi dan komunikasi dari pengguna yang menciptakan banyaknya pertukaran data dan informasi yang dapat diakses dengan mudah. Keterbukaan akses informasi dan data telah memberi banyak kemudahan bagi pengguna internet. Namun, keterbukaan akses informasi ini juga telah mendorong pihak-pihak yang tidak berhak mencoba untuk memperolah data dan informasi yang beredar pada internet. Oleh karena itu, faktor keamanan dalam jaringan komputer merupakan salah satu aspek penting yang harus diperhatikan.

Ancaman dan penyerangan yang dilakukan oleh pihak yang tidak bertanggung jawab sangat beragam. Salah satu upaya kerusakan yang sering dilakukan adalah menggunakan *malware*. *Malware* merupakan istilah umum untuk perangkat lunak yang dimasukkan ke dalam suatu sistem informasi kemudian dikontrol secara jarak jauh dengan tujuan untuk merusak sistem dan mencuri data ataupun informasi penting[1].

Organisasi kejahatan dunia maya yang berbasis di Rusia, Evil Corp, yang dikenal dengan berbagai nama lain seperti TA505, SectorJ04, dan Indrik Spider, merupakan pencipta *malware* Dridex yang telah digunakan untuk peretasan dan penipuan bank yang mengakibatkan kerugian mencapai $100 juta. Selain itu, para peneliti di ESET diawal tahun 2018 menetapkan bahwa para pengembang Dridex juga berada di balik pengembangan *ransomware* yang dikenal sebagai BitPaymer (atau FriedEx). BitPaymer adalah salah satu bentuk utama *ransomware* yang memengaruhi sektor *Healthcare and Public Health* (HPH) dan analisis lebih lanjut oleh Trend Micro mengungkapkan hubungan antara *malware* Dridex, BitPaymer, Emotet, dan Ursnif[1]–[3].

Dridex *Malware* pertama kali muncul pada pertengahan tahun 2014 dengan melakukan penyerangan kepada perusahaan di industri keuangan dan perbankan. Tujuan *malware* dridex adalah untuk mencuri data kredensial pribadi dan akses ke catatan keuangan. *Malware* ini berusaha memanfaatkan dokumen makro agar korban mengunduh *file* yang dapat dieksekusi. Padahal makro sendiri sudah dinonaktifkan secara *default* sejak Office 2007. Dridex memanfaatkan teknik rekayasa sosial untuk mengelabui pengguna agar mau mengaktifkan makro. Pengguna biasanya tertipu melihat faktur penting, tagihan atau dokumen sensitif lainnya. Bahkan para penyerang juga menyertakan petunjuk untuk mengaktifkan makro. Oleh karena itu, pengguna yang tidak mengetahui fungsi makro juga dapat berisiko terinfeksi[4]–[7].

Beberapa penelitian dilakukan untuk membahas secara detail mengenai ancaman dan bahaya Dridex, aktifitas serangan, ciri-ciri serangan dan fitur-fitur yang dimiliki oleh Dridex. Dalam penelitian yang dilakukan oleh Sans Institute [4], dijabarkan secara terperinci bagaimana langkah-langkah analisis terhadap dataset lalu lintas Dridex untuk menemukan *signature* maupun anomali yang berkaitan dengan aktifitas Dridex. Sedangkan Rudman dan Irwin [5] menganalisis sampel lalu lintas data Dridex untuk menemukan *Indicator of Compromise* (IOC) atau *signature* dari aktifitas Dridex. Dari beberapa laporan teknis yang dibuat oleh beberapa lembaga seperti CISA[8], SophosLab [3,9] dan *Health Sector Cybersecurity Coordination Center (HC3)*[2] yang telah menjelaskan secara

terperinci perihal cara kerja, ciri-ciri aktifitas Dridex dan juga cara mencegah Dridex untuk dapat masuk dan mengambil alih sistem yang ada. Salah satu cara preventif yang dilakukan adalah dengan memanfaatkan *Intrusion Detection System* (IDS) sebagai pemberi peringatan apabila terdeteksi adanya aktifitas Dridex.

Intrusion Detection System (IDS) merupakan sebuah aplikasi perangkat lunak yang dapat melakukan inspeksi terhadap lalu lintas *inbound* dan *outbound* dalam sebuah sistem atau jaringan untuk mendeteksi adanya percobaan intrusi (penyusupan) atau aktivitas yang mencurigakan[10], [11]. Dalam mendeteksi adanya ancaman atau aktifitas mencurigakan dalam sebuah jaringan komputer, IDS menerapkan pendekatan deteksi berdasarkan *signature* (*signature-based detection*) dan berdasarkan aktifitas anomali (*anomaly-based detection*).

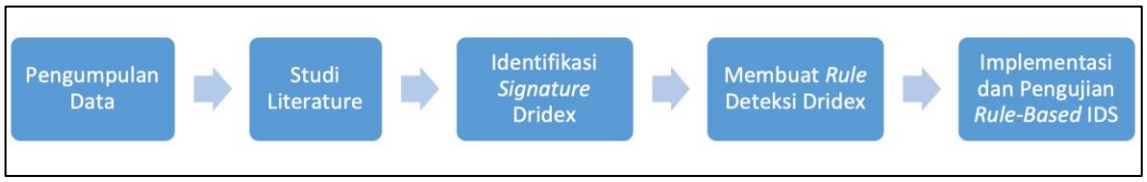
Pada pendekatan *anomaly-based detection*, IDS difungsikan untuk mendeteksi adanya aktifitas sistem yang bukan berdasarkan behavior dan konfigurasi yang telah ditentukan. IDS akan melakukan *baselining* dan *learning* pada *pattern* dari aktifitas normal sistem untuk mendeteksi adanya intrusi. Pada IDS, penyimpangan dari *baseline behavior* tersebut akan menyebabkan munculnya alarm yang menandai adanya aktifitas mencurigakan[12]–[14]. Sedangkan pada *signature-based detection*, IDS akan menganalisis setiap aktifitas (berupa packet, log, dll) pada lalu lintas jaringan untuk dibandingkan dengan ciri-ciri *signature* yang dikenali sebagai aktifitas ancaman atau serangan. Suatu aktifitas (khususnya yang berupa serangan) memiliki karakterisitik dan ciri jejak (*footprint*) tersendiri yang membedakan antara satu dengan yang lainnya. *Footprint* tersebut dapat diistilahkan sebagai *signature*. *Signature* tersebut lalu dikumpulkan menjadi sebuah '*database*' *signature*" yang kemudian digunakan untuk mendeteksi serangan yang sama di kemudian hari. Dalam implementasinya, pendekatan *signature-based detection* memiliki *false positive* lebih kecil dibandingkan teknik lainnya[10], [11], [13], [14].

Salah satu aplikasi *Intrusion Detection System* (IDS) adalah *Snort*. *Snort* merupakan sebuah perangkat yang digunakan untuk mengawasi aktifitas dalam suatu jaringan komputer[11], [13], [15]. *Snort* memberikan fasililtas untuk penggunanya agar dapat membuat *rule* sendiri yang sesuai dengan kondisi dan kebutuhan *user*. Kelebihan ini kemudian dapat dimanfaatkan untuk menerapkan *signature-based* detection IDS dan menyalakan alarm yang menandai adanya aktifitas mencurigakan.

Berdasarkan keterangan di atas, maka penulis bermaksud untuk menganalisis lalu lintas jaringan atas serangan Dridex untuk menemukan perilaku/karakteristik (*signature*) dari *malware* Dridex. Berdasarkan *signature* tersebut, akan dibuatkan *rule-snort* yang dikustomisasi untuk mendeteksi adanya ancaman Dridex pada lalu lintas jaringan.

B. Metode Penelitian

Pada penelitian ini dijelaskan beberapa tahapan yang dilakukan untuk mencapai tujuan penelitian. Adapun tahapannya adalah pengumpulan data, identifikasi *signature* Dridex, membuat *rule* deteksi Dridex, implementasi *rule*-based IDS, pengujian.



Gambar 1. Metode Penelitian

1. Pengumpulan Data

   Data yang digunakan dalam penelitian ini adalah data publik yang tersedia dan bisa diakses melalui www.malware-traffic-analysis.net dengan judul halaman website "*Dridex Malspam with PDF Attachments Containing Embedded Word Docs".* Data yang didapatkan merupakan data lalu lintas jaringan berupa file berukuran 1,48 Mb dengan ekstensi .pcap.
2. Studi Literatur

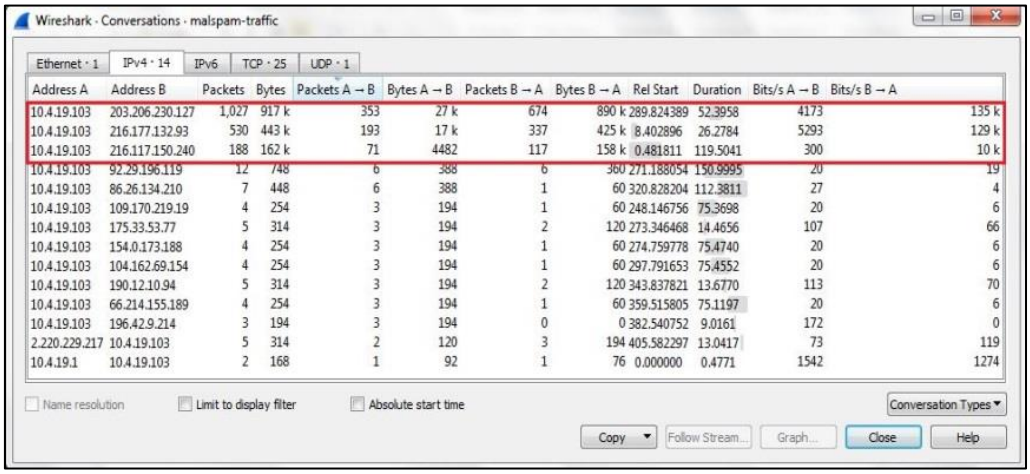
   Dalam tahapan ini dilakukan pencarian dari literature mengenai *signature* dari Dridex. Pada penelitian yang dilakukan oleh [5] menjelaskan penemuan 50 sampel lalu lintas Dridex yang diambil selama proses 30 menit. 40 sampel menggunakan protokol DNS (port 53 UDP), 33 sampel mencoba mengkases nama domain icanhazip.com dengan IP address 64.182.208.185 & 64.182.208.184. Adapun sampel lainnya mencoba mengakses beberapa nama domain lain.

   Dari laporan teknis SANS Institute[4], protokol yang digunakan Dridex adalah SSL dan TCP port yang digunakan 80, 443, 8000, 8080. Terdapat beberapa kemungkinan *signature* yaitu ukuran paket 543 *bytes* dan 828 *bytes*, *destination IP address* 46.36.217.227.
3. Identifikasi *Signature* Dridex pada file pcap

   Dalam proses ini dilakukan analisis terhadap *file* pcap yang didapatkan dengan bantuan aplikasi *Wireshark*[4], [5], [16]. Adapun tahapan lebih detail mengenai analisa lalu lintas jaringan terbagi menjadi beberapa tahapan berikut.

   a. Frekuensi transaksi data berdasarkan IP address

   Pada analisa *traffic* jaringan, terjadi kecurigaan jika suatu *user* terhubung atau berkomunikasi dengan server tertentu secara berulang dengan jarak waktu yang sangat singkat. Dari proses analisis 1800 paket, ditemukan 14 IP sumber yang berbeda yang kemudian diurutkan berdasarkan IP address yang memiliki frekuensi aktifitas paling banyak di dalam lalu lintas jaringan.

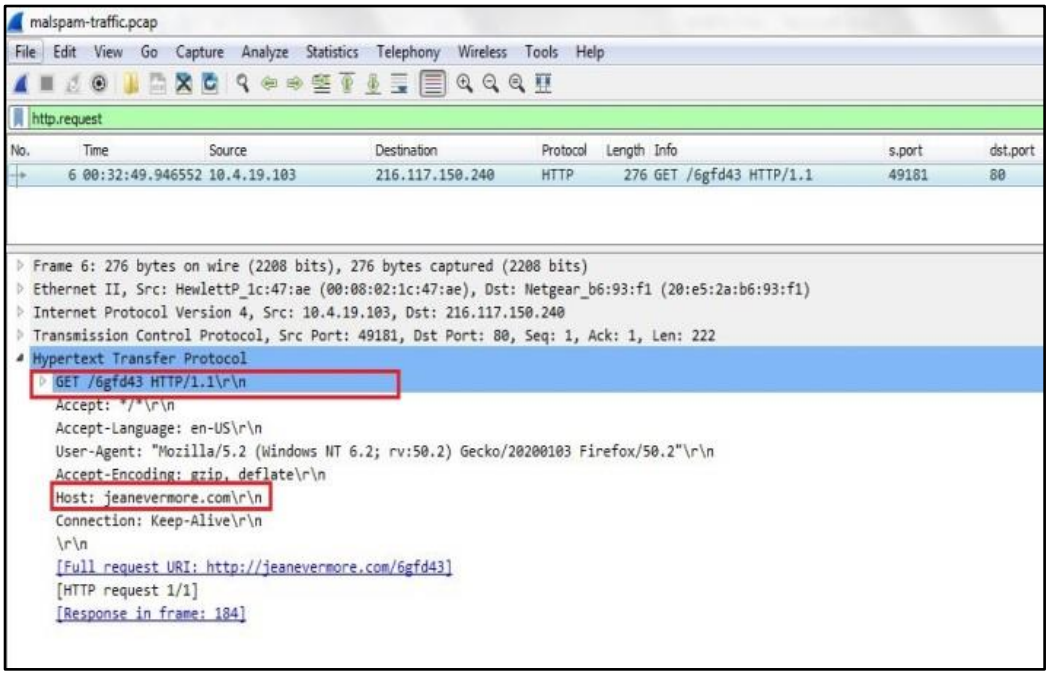


Gambar 2. Jumlah Aktifitas Terbanyak pada Dataset

Berdasarkan ringkasan pada Gambar 2, disimpulkan bahwa terdapat 1 IP sumber yaitu 14.0.17.103 yang memiliki komunikasi ke 3 IP tujuan dengan jumlah paket yang sangat banyak dan dalam waktu yang sangat dekat. Hal ini jelas menimbulkan kecurigaan dimana jumlah paket yang besar serta tujuan ip yang berbeda dapat diakses dalam waktu yang sangat dekat.

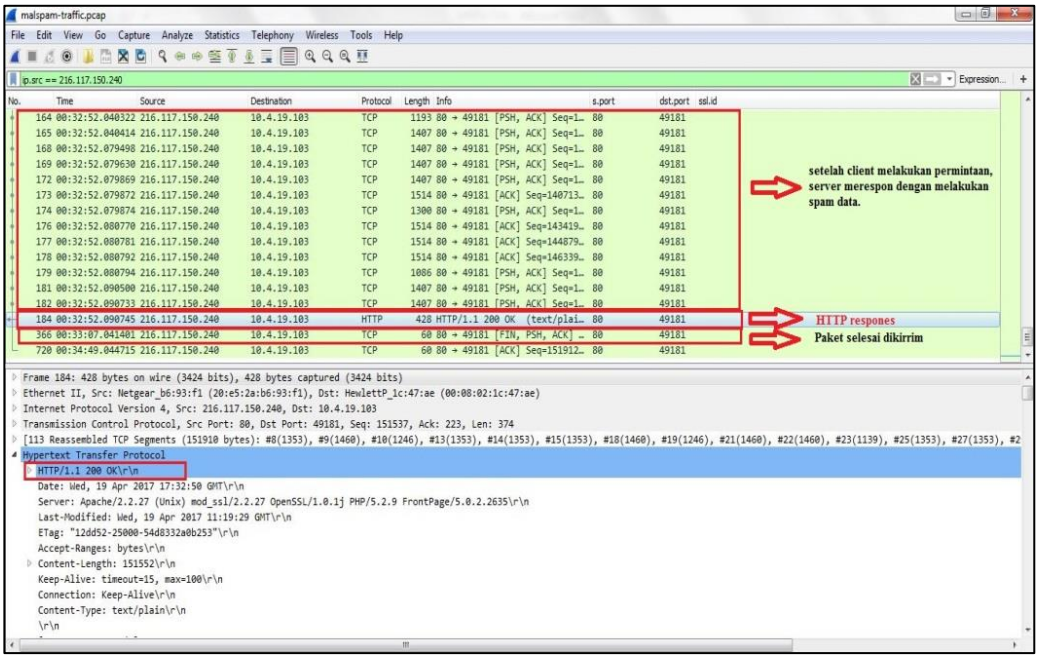
b. Analisa HTTP Request

Berdasarkan temuan sebelumnya terdapat transaksi data dari 1 sumber IP menuju ke beberapa tujuan IP. Filter *Wireshark* yang kemudian digunakan untuk menyaring aktifitas pada *user* adalah "http.request". Berikut adalah hasil filter pada salah satu paket yang dicurigai.



Gambar 3. Isi "Http.Request" pada Salah Satu Paket yang Dicurigai

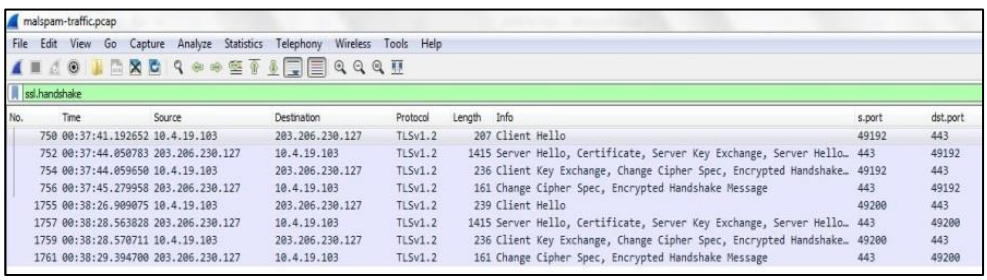
Dari hasil temuan Gambar 3, terdapat aktifitas yang cukup mencurigakan yaitu klien mencoba mengakses sebuah situs yang tidak populer yaitu 'jeanevermore.com" dan dicurigai sebagai server *malware*. Terbukti setelah request tersebut diterima oleh server, balasannya adalah memberikan banyak paket yang membanjiri klien dalam kurun waktu yang sangat singkat. Aktifitas ini ditunjukkan pada Gambar 4.



Gambar 4. Aktifitas Tidak Normal pada Server

c. Analisa SSL/TLS Handshake

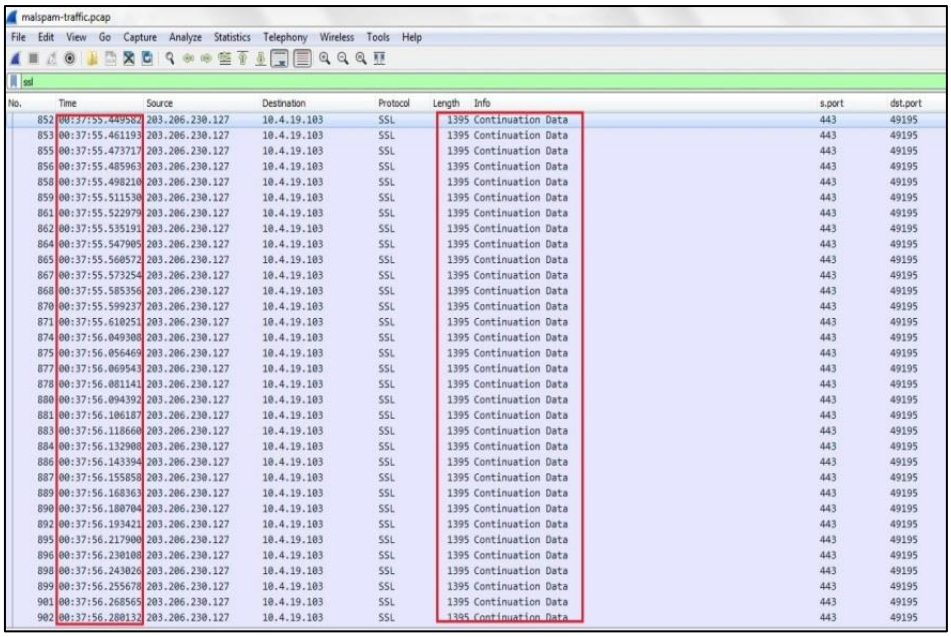
Berdasarkan penelitian sebelumnya oleh SANS Institute [4], salah satu bentuk serangan Dridex adalah melalui protokol SSL port 443. Secure Sockets Layer (SSL) dan Transport Layer Security (TLS) keduanya merupakan protokol yang digunakan untuk enkripsi data jaringan. SSL/TLS *handshake* dimulai ketika klien mengirim Pesan ClientHello. Gambar 5 adalah tampilan pada saat klien memulai percakapan dengan server Dridex.



Gambar 5. Proses SSL/TLS *Handshake*

Sesaat setelah server mengkonfirmasi request dari klien melalui ssl-handshake, server melakukan pengiriman paket dalam jumlah yang

banyak serta ukuran yang sama hanya dengan durasi kurang dari 1 detik. *Wireshark* tidak dapat membaca pesan terenkripsi yang dikirim oleh server, untuk itu *Wireshark* memberi nama "Application data" dan "Continuation data". Hal ini ditampilkan pada gambar 6.



Gambar 6. Aktifitas Tidak Normal pada Komunikasi Jaringan

4. Membuat *Rules* Deteksi Dridex

Berdasarkan temuan *signature* Dridex sebelumnya, maka tahap berikutnya adalah membuat *rule* yang digunakan untuk mendeteksi serta memberi peringatan apabila ditemukan adanya aktifitas Dridex. Dari hasil analisa yang sudah dilakukan maka terbentuklah 12 *rules* yang akan digunakan untuk mendeteksi adanya Dridex. Adapun *rules* yang dibuat adalah sebagai berikut.

Tabel 1. Dua Belas *Rules* untuk Deteksi Dridex

No	Identitas	*Rule*
1	*Rule1*	alert tcp any any -> 216.117.150.240 80 (msg:"ET TROJAN Known Malicious Doc Downloading Payload"; flow:to_server,established; content:"GET"; nocase; http_method; sid:1000001;)
2	*Rule2*	alert tcp 216.117.150.240 80 -> any any (msg:"ET TROJAN abused transfer data"; flags:A; sid:1000003; rev:1;)
3	*Rule3*	alert tcp 216.117.150.240 80 -> any any (msg:" ET TROJAN abused transfer data"; flags:PA; sid:1000005; rev:1;)
4	*Rule4*	alert tcp 216.117.150.240 80 -> any any (msg:" ET TROJAN abused transfer data "; flags:FPA; sid:1000025; rev:1;)
5	*Rule5*	alert tcp any any -> 203.206.230.127 443 (msg:" ET TROJAN ABUSE SSL Blacklist Malicious SSL certificate detected (Dridex)"; flags:PA; flow:to_server,established; sid:1000006;)
6	*Rule6*	alert tcp 203.206.230.127 443 -> any any (msg:" ET TROJAN ABUSE SSL Blacklist Malicious SSL certificate detected (Dridex)"; flags:A; sid:1000007; rev:1;)
7	*Rule7*	alert tcp 203.206.230.127 443 -> any any (msg:" ET TROJAN ABUSE

		SSL Blacklist Malicious SSL certificate detected (Dridex)"; flags:PA; sid:1000007; rev:1;)
8	*Rule8*	alert tcp 203.206.230.127 443 -> any any (msg:" ET TROJAN ABUSE SSL Blacklist Malicious SSL certificate detected (Dridex)"; flags:FPA; sid:1000007; rev:1;)
9	*Rule9*	alert tcp any any -> 216.177.132.93 4143 (msg:"ET TROJAN ABUSE Malicious Content Detect "; flags:PA; flow:to_server,established; sid:1000011;) rev:1;)
10	*Rule10*	alert tcp 216.177.132.93 4143 -> any any (msg:" ET TROJAN ABUSE Malicious Content Detect "; flags:A; sid:1000012; rev:1;)
11	*Rule11*	alert tcp 216.177.132.93 4143 -> any any (msg:" ET TROJAN ABUSE Malicious Content Detect "; flags:PA; sid:1000013; rev:1;)
12	*Rule12*	alert tcp 216.177.132.93 4143 -> any any (msg:" ET TROJAN ABUSE Malicious Content Detect "; flags:FPA; sid:1000014; rev:1;)

5. Implementasi dan pengujian *Rule-Based* IDS

   Pada proses ini dilakukan implementasi dua tahap. Tahap pertama adalah melakukan komparasi atas deteksi Snort menggunakan *rules* standar dan deteksi Snort yang ditambahkan *rules* anti-Dridex. Tahap berikutnya adalah melakukan komparasi setiap *rule* pada anti Dridex pada saat diimplementasikan pada Snort dan juga dilakukan filtering pada *Wireshark*.

C. Hasil dan Pembahasan

Hasil pengujian pertama adalah mengkomparasi akurasi Snort standar dan Snort Anti-Dridex. Percobaan pertama adalah menjalankan aplikasi snort dengan menggunakan *rules* standar yang berjumlah 118 file *rules*. Pada percobaan ini telah dilakukan analisa terhadap 1800 paket dan mampu memberikan 1 peringatan yang mengindikasikan adanya aktifitas mencurigakan. Berikutnya dilakukan percobaan terhadap *file* pcap yang sama namun menggunakan snort dengan *rules* yang telah ditambahkan dengan *rules* anti Dridex. *Rules* anti Dridex dibuat berdasarkan temuan dan analisa atas *signature* Dridex yang sudah didapatkan dari penelitian ini. Hasil dari percobaan ini menemukan bahwa terdapat 1352 paket yang dicurigai membawa *malware* Dridex. Gambar 7.a menunjukan hasil deteksi snort menggunakan *rules* standar dan gambar 7.b menunjukan hasil deteksi Dridex menggunakan *rules* yang telah ditambahkan *rules* anti Dridex

(a) (b)

Gambar 7. (a) Hasil Deteksi Snort Standar (b) Hasil Deteksi Snort Anti-Dridex

Hasil komparasi Gambar 7. menunjukan bahwa *snort* menggunakan *rules* anti-Dridex telah meningkatkan hasil deteksi atas adanya aktifitas mencurigakan dari Dridex jika dibandingkan dengan *snort* menggunakan *rules* standar. Berdasarkan kesimpulan tersebut maka dapat disimpulkan bahwa snort anti-Dridex menghasilkan deteksi lebih baik daripada *snort* standar.

Tahap berikutnya adalah melakukan komparasi hasil deteksi setiap *rules* anti Dridex pada saat diimplementasikan di Snort dan filter *Wireshark*. Sebagaimana diketahui filter pada *Wireshark* merupakan salah satu fungsi yang tersedia pada aplikasi dan tidak dapat dirubah oleh *user*, sedangkan *rules* pada *snort* dapat dikustomisasi sesuai keinginan dan keadaan *user* sehingga tingkat akurasi dan kepercayaannya lebih rendah apabila dibandingkan filter pada aplikasi yang sudah stabil seperti *Wireshark*. Oleh karena itu, percobaan ini dilakukan untuk mengukur seberapa akurat implementasi *rules* pada *snort* jika dibandingkan dengan filter *Wireshark*. Tabel 2. menunjukan komparasi hasil deteksi *signature* Dridex antara filter *Wireshark* dan snort rules anti-Dridex.

Tabel 2. Hasil Komparasi Deteksi Filter dan Snort

No	Identitas	Hasil Deteksi Filter *Wireshark*	Hasil Deteksi Snort anti Dridex	TP	TN	FP	FN
1	Rule1	1	1	1			
2	Rule2	29	30	29		1	
3	Rule3	88	88	88			
4	Rule4	1	2	1		1	
5	Rule5	14	14	14			
6	Rule6	54	258	54		204	
7	Rule7	604	604	604			
8	Rule8	8	8	8			
9	Rule9	13	13	13			
10	Rule10	104	104	104			
11	Rule11	225	225	225			
12	Rule12	4	4	4			
	Jumlah	1145	1351	1145	0	206	0

Berdasarkan Tabel 2. ditemukan perbedaan atas hasil deteksi *signature* Dridex melalui filter *Wireshark* dan deteksi *snort*. Pada filter *Wireshark* ditemukan 1145 paket yang mengindikasikan adanya *signature* Dridex. Sedangkan pada deteksi *snort* telah ditemukan 1351 paket memicu peringatan snort sebagai tanda adanya *signature* Dridex. Terjadinya perbedaan tersebut dikarenakan adanya beberapa *rules snort* yang dijalankan bersamaan mendeteksi paket yang terindikasi Dridex dihitung lebih dari satu kali sebagai tanda peringatan.

Hasil dari filter 6 dan *rules* 6 menunjukan jumlah deteksi yang sangat berbeda yaitu filter *Wireshark* menemukan 54 peringatan sedangkan snort menemukan 258 peringatan. Filter 6 dan rules 6 menganalisis aliran data dari "ssl.handshake" dengan membaca penanda "*flag"* yang ada pada paket. Perintah-perintah ini difungsikan untuk memberikan peringatan ketika IP 203.206.230.127 dengan port 443 mengirim data ke alamat IP dan port manapun, dengan "*flag*" A yaitu ACKnowledgement.

Perintah filter *Wireshark* yang digunakan yaitu *"ip.src == 203.206.230.127 && tcp.flags.push == 0 && tcp.flags.ack == 1 && tcp.flags.syn == 0"* dapat secara spesifik memfilterkan "*flag"* ACKnowledgement. Sedangkan pada rules 6 snort "*alert tcp*

203.206.230.127 443 -> any any (msg:" ET TROJAN ABUSE SSL Blacklist Malicious SSL certificate detected (Dridex)"; flags:A; sid:1000007; rev:1;), PUSH dan SYN masih ikut masuk kedalam hasil proses filter. Selain itu adanya redudansi proses deteksi yang menyebabkan hasil terdeteksi Dridex menjadi lebih banyak. Proses selanjutnya adalah mengukur hasil deteksi menggunakan confusion matrix untuk menemukan nilai akurasi, presisi dan reccall. Hasil pembahasan tersebut ditampilkan pada Tabel 3.

Tabel 3. Hasil Perhitungan Akhir

Confusion Matrix	Rumus	Hasil
Akurasi	((TP+TN)/(TP+TN+FP+FN))*100%	88,5%
Presisi	((TP)/(FP+TP))*100%	84,75%
Recall	((TP)/(FN+TP))*100%	100%

Hasil perhitungan didapatkan nilai sebesar 88,5% untuk akurasi sehingga dapat dikegorikan sebagai baik dan karena mendekati nilai 1 (=100%) dan memiliki nilai antara 80% – 100%. Nilai presisi menunjukkan jumlah total serangan yang terklasifikasi secara benar sebesar 84,75%. Sedangkan nilai recall atau *detection rate* menggambarkan jumlah total serangan yang terdeteksi selama data yang dipindai yaitu 100%.

D. Simpulan

Penelitian ini telah memberikan penjelasan terkait analisis dataset lalu lintas jaringan untuk menemukan *signature* dari *malware* Dridex. Berdasarkan hasil temuan signature Dridex, maka disusunlah 12 (dua belas) rules yang diimplementasikan pada *snort* untuk mendeteksi keberadaan aktifitas mencurigakan Dridex pada lalu lintas jaringan. Sebagai hasil, deteksi Dridex berbasis *signature* berhasil memberikan peringatan yang lebih banyak dari snort *rule* standar. Uji coba akurasi, presisi dan recall dilakukan guna mengukur tingkat keberhasilan implementasi 12 *rules* berbasis *signature* pada *snort* dikomparasikan dengan deteksi menggunakan filter *Wireshark* yang menghasilkan nilai sebesar 88,5% untuk akurasi, 84,75% untuk presisi dan 100% untuk nilai *detection rate*.

E. Referensi

[1] Y. Zhang, J. Niu, D. Guo, Y. Teng, and X. Bao, “Unknown Network Attack Detection Based on Open Set Recognition,” *Procedia Comput. Sci.*, vol. 174, no. 2019, pp. 387–392, 2020.

[2] Health Sector Cybersecurity Coordination Center (HC3), “Health Sector Cybersecurity Coordination Center (HC3) Sector Alert,” 2020.

[3] SophosLabs, Sophos Managed Threat Response, Sophos Rapid Response, Sophos IA, and Cloud Security, “Sophos 2021 Threat Report,” 2020.

[4] L. Teo, “Learning from the Dridex Malware - Adopting an Effective Strategy,” *SANS Inst.*, p. 34, 2015.

[5] L. Rudman and B. Irwin, “Dridex: Analysis of the traffic and automatic generation of IOCs,” *2016 Inf. Secur. South Africa - Proc. 2016 ISSA Conf.*, pp. 77–84, 2016.

[6] J. Gajek, “Macro malware: dissecting a malicious Word document,” *Netw. Secur.*, vol. 2017, no. 5, pp. 8–13, 2017.

[7] Y. Cohen, D. Hendler, and A. Rubin, "Detection of malicious webmail attachments based on propagation patterns," *Knowledge-Based Syst.*, vol. 141, pp. 67–79, 2018.
[8] C. and I. S. A. (CISA), "National Cyber Awareness System Alerts Dridex Malware Alert (AA19-339A)," 2019.
[9] SophosLabs Research Team, "Emotet exposed: looking inside highly destructive malware," *Netw. Secur.*, vol. 2019, no. 6, pp. 6–11, 2019.
[10] L. Dali *et al.*, "A survey of intrusion detection system," *2015 2nd World Symp. Web Appl. Networking, WSWAN 2015*, 2015.
[11] A. Garg and P. Maheshwari, "Performance analysis of Snort-based Intrusion Detection System," *ICACCS 2016 - 3rd Int. Conf. Adv. Comput. Commun. Syst. Bringing to Table, Futur. Technol. from Arround Globe*, pp. 0–4, 2016.
[12] A. Nugraha and N. Rijati, "Penerapan Metode Principal Component Analysis (PCA) Untuk Deteksi Anomali Pada Jaringan Peer-To-Peer (P2P) Botnet," *Techno.COM*, vol. 14, no. 3, pp. 212–217, 2015.
[13] S. M. Hussein, "Performance Evaluation of Intrusion Detection System Using Anomaly and Signature Based Algorithms to Reduction False Alarm Rate and Detect Unknown Attacks," *Proc. - 2016 Int. Conf. Comput. Sci. Comput. Intell. CSCI 2016*, pp. 1064–1069, 2017.
[14] R. M. A. Ujjan, Z. Pervez, and K. Dahal, "Suspicious Traffic Detection in SDN with Collaborative Techniques of Snort and Deep Neural Networks," *Proc. - 20th Int. Conf. High Perform. Comput. Commun. 16th Int. Conf. Smart City 4th Int. Conf. Data Sci. Syst. HPCC/SmartCity/DSS 2018*, pp. 915–920, 2019.
[15] A. Sagala, "Automatic SNORT IDS rule generation based on honeypot log," *Proc. - 2015 7th Int. Conf. Inf. Technol. Electr. Eng. Envisioning Trend Comput. Inf. Eng. ICITEE 2015*, pp. 576–580, 2015.
[16] A. S. Shekhawat, F. Di Troia, and M. Stamp, "Feature analysis of encrypted malicious traffic," *Expert Syst. Appl.*, vol. 125, pp. 130–141, 2019.