PERBANDINGAN DAN KEBERKESANAN SISTEM PENGESANAN PENCEROBOHAN PERISIAN SUMBER TERBUKA: SNORT, SURICATA DAN ZEEK

NUR HAYATI BINTI AHMAD

UNIVERSITI KEBANGSAAN MALAYSIA

PERBANDINGAN DAN KEBERKESANAN SISTEM PENGESANAN PENCEROBOHAN PERISIAN SUMBER TERBUKA: SNORT, SURICATA DAN ZEEK

NUR HAYATI BINTI AHMAD

PROJEK YANG DIKEMUKAKAN UNTUK MEMENUHI SEBAHAGIAN
DARIPADA SYARAT MEMPEROLEH IJAZAH
SARJANA KESELAMATAN SIBER

FAKULTI TEKNOLOGI DAN SAINS MAKLUMAT
UNIVERSITI KEBANGSAAN MALAYSIA
BANGI

2024

PENGAKUAN

Saya akui karya ini adalah hasil kerja saya sendiri kecuali nukilan dan ringkasan yang tiap-tiap satunya telah saya jelaskan sumbernya.

27 Februari 2024

NUR HAYATI AHMAD
P95384

PENGHARGAAN

Alhamdulillah, syukur ke hadrat Ilahi dengan limpah kurniaNya, laporan projek ini telah berjaya disiapkan dengan pelbagai cabaran yang dihadapi. Setinggi-tinggi penghargaan dan ucapan terima kasih kepada Profesor Madya Dr. Khairul Akram Zainol Ariffin, selaku penyelia projek yang telah banyak membantu dan memberikan tunjuk ajar, perhatian, semangat dan nasihat dalam melaksanakan projek ini. Tidak dilupakan kepada barisan pensyarah dan staf teknikal di Fakulti Teknologi dan Sains Maklumat, Universiti Kebangsaan Malaysia yang sudi berkongsi pengalaman sepanjang pengajian ini. Segala pengalaman yang dilalui pasti tidak dapat dilupakan

Begitu juga kepada pengurusan Lembaga Perindustrian Kayu Malaysia yang telah memberikan sokongan, semangat dan kewangan untuk saya menyelesaikan program ini.

ABSTRAK

Dalam zaman digital hari ini, ancaman serangan siber dan pencerobohan merupakan kebimbangan yang berterusan bagi organisasi dan individu. Oleh itu, keperluan untuk sistem pengesan pencerobohan (IDS) yang berkesan telah menjadi amat penting. Penyelesaian IDS tradisional, berdasarkan perisian eksklusif, boleh menjadi mahal dan mungkin tidak selalu menyediakan tahap perlindungan yang diingini. Sebagai alternatif, penyelesaian IDS sumber terbuka telah mendapat popularity kerana kos yang berkesan dan fleksibilitinya. Kajian ini bertujuan untuk membandingkan keberkesanan tiga sistem IDS sumber terbuka yang popular: Snort, Suricata, dan Zeek. Tiga perisian ini dipilih berdasarkan penggunaan meluas dan reputasinya yang boleh dipercayai dan kaya dengan ciri-ciri. Metodologi yang digunakan dalam kajian ini melibatkan kajian kesusasteraan yang teliti dan penilaian praktikal terhadap ciri-ciri dan keupayaan setiap sistem. Kaedah pelaksanaan eksperimen melibatkan *replay* set data dan output disimpan di direktori log seterusnya dianalisis. Kriteria penilaian termasuk kemudahan pemasangan, prestasi, keupayaan untuk mengesan ancaman dan fleksibiliti dalam penyesuaian. Kajian ini untuk menilai penilaian prestasi dari segi penggunaan sumber iaitu CPU, memori dan kapasiti perkakasan. Selain itu, ia juga untuk menilai keberkesanan pengesanan trafik rangkaian untuk mengurangkan risiko ancaman. Eksperimen yang dilaksana menggunakan set data CIC-IDS2017 dan CICDDoS2019. Hasil daripada eksperimen tersebut menunjukkan bahawa Suricata menunjukkan prestasi yang baik dari segi penggunaan sumber yang rendah iaitu CPU antara 25% hingga 26% dan RAM 5% hingga 8%. Kadar ketepatan pengesanan pula antara 95% hingga 96%. Penggunaan sumber oleh Zeek adalah tinggi iaitu 80% hingga 100% dan RAM 75% hingga 96% tetapi kadar ketepatan adalah 90% hingga 94%. Suricata mempunyai *throughput* yang lebih tinggi dan ciri-ciri yang lebih maju, manakala Zeek menawarkan pendekatan unik dalam pengesanan pencerobohan dengan menganalisis trafik rangkaian secara masa nyata. Selain itu, sifat sumber terbuka sistem-sistem ini menyediakan komuniti pembangun yang sentiasa bekerja untuk meningkatkan dan mengemaskini perisian, menjadikannya lebih tahan terhadap ancaman yang muncul. Secara keseluruhannya, kajian ini menonjolkan keberkesanan dan nilai penyelesaian IDS sumber terbuka dalam menyediakan perlindungan yang berkesan dan berpatutan terhadap pencerobohan siber. Kajian seterusnya perlu dilanjutkan dengan penggunaan set peraturan Suricata di Snort dan sebaliknya bagi melihat keberkesanan yang lebih baik.

COMPARISON AND EFFECTIVENESS OF OPEN SOURCE SOFTWARE INTRUSION DETECTION SYSTEMS: SNORT, SURICATA AND ZEEK

ABSTRACT

In today's digital age, the threat of cyberattacks and intrusions is a constant concern for organizations and individuals. Therefore, the need for an effective intrusion detection system (IDS) has become crucial. Traditional IDS solutions, based on exclusive software, can be expensive and may not always provide the desired level of protection. Alternatively, the open source IDS solution has gained popularity due to its cost-effectiveness and flexibility. The study aims to compare the effectiveness of three popular open source IDS systems: Snort, Suricata, and Zeek. These three programs were selected based on their widespread use and reputation as reliable and feature-rich. The methodology used in this study involves a thorough literary study and a practical assessment of the characteristics and capabilities of each system. The experiment execution method involves replaying data sets and outputs stored in the next log directory being analyzed. The evaluation criteria include ease of installation, performance, ability to detect threats and flexibility in adaptation. This study is to evaluate performance assessments in terms of resource usage such as CPU, memory and hardware capacity. In addition, it is also to assess the effectiveness of network traffic detection to reduce threat risk. Experiments were conducted using CIC-IDS2017 and CICDDoS2019 data sets. The results of the experiment showed that Suricata showed good performance in terms of low resource consumption of between 25% to 26% CPU and 5% to 8% RAM. The detection accuracy is between 95% and 96%. Zeek's resource usage is high, 80 percent to 100 percent and RAM 75 percent to 96 percent but the accuracy rate is 90 percent to 94 percent. Suricata has higher throughput and more advanced features, while Zeek offers a unique approach in intrusion detection by analyzing network traffic in real time. In addition, the open-source nature of these systems provides a community of developers who are constantly working to improve and update the software, making it more resistant to emerging threats. Overall, the study highlights the effectiveness and value of open source IDS solutions in providing effective and affordable protection against cyber intrusion. Further research needs to be continued with the use of a Suricata set of rules in Snort and vice versa to see better effectiveness.

KANDUNGAN

Halaman

SENARAI JADUAL

SENARAI ILUSTRASI

SENARAI SINGKATAN

IDS	Intrusion Detection System
IPS	Intrusion Protection System
FPR	False Positive Rate
FNR	False Negative Rate
NSM	Network Security Management
NIDS	Network Intrusion Detection System
CPU	Central Processing Unit
MyCERT	Malaysian Computer Emergency Response Team
VM	Virtual Machine
IP	Internet Protocol
RAM	Random Access Memory
PIBD	Pattern Based Intrusion Detection

BAB I

PENDAHULUAN

1.1 PENGENALAN

Evolusi teknologi digital berkembang pesat memberikan faedah dan manfaat kepada masyarakat. Namun, penggunaan teknologi dan kebergantungan terhadap teknologi turut mendedahkan masyarakat kepada pelbagai risiko ancaman dan serangan siber. Sejajar dengan perkembangan teknologi digital, keselamatan siber juga telah berkembang ke tahap yang lebih tinggi dengan penekanan pada *zero trust*, kecerdasan buatan dan teknologi awan. Serangan siber di seluruh dunia meningkat 40% hingga 45%, menjadikan ancaman yang perlu ditangani juga semakin meningkat bergantung kepada teknologi keselamatan (Forbes, 2023). Bagi melindungi data sensitif dan mencegah serangan siber, penggunaan sistem pengesanan pencerobohan (IDS) telah menjadi komponen penting dalam infrastruktur keselamatan siber. Sistem ini bertanggungjawab untuk memantau trafik rangkaian, mengenal pasti dan memberi amaran kepada potensi ancaman keselamatan dan akhirnya melindungi data sensitif daripada ancaman serangan siber.

Menurut kajian Cybersecurity Ventures, serangan siber akan berlaku setiap 11 saat dan dianggarkan kos jenayah siber akan mencecah USD10 trilion setahun pada tahun 2025 (Herath, McNeil, 2020). Berdasarkan statistik yang dikeluarkan oleh Malaysian Computer Emergency Response Team (MyCERT), terdapat lebih 10,000 kes berkaitan keselamatan siber dilaporkan berlaku setiap tahun sejak dari tahun 2018 dan tahun 2022 sebanyak 5,917 kes (MYCERT, 2023). Daripada jumlah kes tersebut, kebanyakannya merupakan kes melibatkan penipuan dan pencerobohan sistem operasi komputer dengan kerugian mencecah jutaan ringgit. Statistik yang membimbangkan ini

merupakan salah satu sebab keperluan IDS yang mantap untuk mengesan dan mencegah serangan ini.

Salah satu perisian sumber terbuka yang paling banyak digunakan untuk pengesanan pencerobohan ialah Snort (Thapa et al. 2020). Ia dibangunkan oleh Cisco, Snort telah wujud selama lebih dari dua dekad dan telah terbukti menjadi perisian yang boleh dipercayai dan berkesan untuk mengesan dan mencegah ancaman siber.

Satu lagi perisian sumber terbuka yang telah mendapat daya tarikan dalam beberapa tahun kebelakangan ini ialah Suricata. Ia dibangunkan oleh Open Information Security Foundation (OISF), Suricata terkenal dengan kelajuan dan skalabilitinya, menjadikannya pilihan utama untuk rangkaian berprestasi tinggi. Menurut laporan oleh Grand View Research, pasaran global Suricata dijangka mencecah USD 2.24 bilion menjelang 2023, seterusnya menonjolkan popularitinya dan keberkesanannya yang semakin meningkat.

Akhir sekali, satu lagi perisian sumber terbuka yang telah mendapat perhatian dalam dunia pengesanan pencerobohan ialah Zeek (dahulunya dikenali sebagai Bro). Zeek terkenal dengan keupayaan analisis rangkaiannya yang kuat dan telah digunakan dalam pelbagai industri termasuk kewangan, penjagaan kesihatan dan kerajaan. Menurut laporan oleh MarketsandMarkets, pasaran analisis trafik rangkaian global (termasuk Zeek) dijangka mencapai USD 3.2 bilion menjelang 2023, mengukuhkan lagi kepentingan dan keberkesanan perisian sumber terbuka ini.

1.2 LATAR BELAKANG KAJIAN

Intrusion Detection System (IDS) atau Sistem Pengesanan Pencerobohan digunakan untuk mengenal pasti penceroboh dalam rangkaian dengan memantau kegiatan melalui rangkaian komunikasi. Apabila sistem pengesanan pencerobohan mendapati adanya aktiviti tidak normal dalam rangkaian, ia akan menghantar isyarat kepada pentadbir yang mengingatkan akan kemungkinan adanya penceroboh dalam sistem rangkaian. Oleh itu, terdapat keperluan sistem pengesanan pencerobohan (IDS) yang dapat memantau dan menganalisis trafik melalui rangkaian dengan berkesan untuk

mengetahui ancaman pencerobohan sebelum mereka merosakkan rangkaian dengan melakukan sebarang aktiviti berbahaya.

IDS digunakan untuk memantau trafik pada rangkaian komputer untuk mengesan sebarang aktiviti yang mencurigakan. Ia menganalisis data yang mengalir melalui rangkaian untuk mencari corak dan tanda-tanda tingkah laku yang tidak normal. IDS membandingkan aktiviti rangkaian dengan satu set peraturan dan corak yang mungkin menunjukkan serangan atau pencerobohan. Jika IDS mengesan sesuatu yang sepadan dengan salah satu peraturan (rule set), ia akan menghantar amaran kepada pentadbir sistem. Pentadbir sistem kemudiannya boleh menyiasat amaran dan mengambil tindakan terhadap sebarang kerosakan atau pencerobohan selanjutnya.

IDS dikategorikan kepada lima (5) jenis iaitu Sistem Pengesanan Pencerobohan Rangkaian (NIDS), Sistem Pengesanan Pencerobohan Hos (HIDS), Sistem Pengesanan Pencerobohan Berasaskan Protokol (PIDS), Sistem Pengesanan Berasaskan Protokol Aplikasi (APIDS) dan Sistem Pengesanan Pencerobohan Hibrid.

Terdapat dua pengesanan ancaman utama iaitu pengesanan berasaskan tandatangan (signature-based detection) dan pengesanan berasaskan anomali (anomaly-based detection). Pengesanan berasaskan tandatangan adalah kaedah yang digunakan dalam sistem pengesanan pencerobohan (IDS) yang membandingkan tandatangan ancaman yang diketahui dengan peristiwa yang diperhatikan untuk mengenal pasti insiden. Ia bergantung pada teknik pemadanan corak untuk mencari pencerobohan sebelumnya dengan membandingkan tandatangan peristiwa semasa dengan tandatangan yang disimpan dalam pangkalan data (Mouli, K et al. 2023). Pendekatan ini berkesan dalam mengesan ancaman yang diketahui tetapi tidak berkesan dalam mengesan serangan yang tidak diketahui atau baru (Khraisat et al. 2019). Sebaliknya, pengesanan berasaskan anomali adalah kaedah lain yang digunakan dalam IDS yang membandingkan peristiwa yang diperhatikan dengan definisi apa yang dianggap sebagai aktiviti normal untuk mengenal pasti penyimpangan yang signifikan. Ini melibatkan pemantauan ciri-ciri aktiviti biasa dari masa ke masa dan membandingkan aktiviti semasa dengan ambang yang berkaitan dengan profil. Pengesanan berasaskan anomali boleh berkesan dalam mengesan ancaman yang tidak diketahui sebelumnya

tetapi boleh menghasilkan positif palsu dan mengalami kesukaran untuk menentukan sempadan antara tingkah laku normal dan tidak normal (Safana Hyder Abbas et al, 2023).

Sehubungan itu, kajian ini akan menumpukan kepada Sistem Pengesanan Pencerobohan Rangkaian (NIDS) menggunakan perisian sumber terbuka antaranya Snort, Suricata dan Zeek untuk melihat perbandingan dan keberkesanan sesuatu perisian tersebut. Kajian ini akan menilai dari segi prestasi penggunaan sumber seperti CPU, memori dan kapasiti rangkaian. Selain itu, kajian ini juga akan menilai ketepatan pengesanan ancaman siber termasuk perisian hasad dan serangan seperti DoS, DDoS. NIDS memeriksa trafik dari semua peranti pada rangkaian dan diletakkan di luar tembok api di mana keseluruhan trafik luaran boleh dipantau dengan mengesan aktiviti anomali. Perisian tersebut telah mendapat penerimaan meluas sama ada dalam bidang akademik mahupun industri kerana keteguhan dan kebolehpercayaannya.

Pelbagai kajian telah dilaksanakan untuk menilai prestasi dan keberkesanan perisian IDS sumber terbuka ini. Satu kajian oleh Karim et al. (2019) membandingkan kadar pengesanan Snort, Suricata dan Zeek. Ianya mendapati bahawa Zeek mempunyai kadar pengesanan tertinggi untuk serangan berasaskan HTTP. Selain itu, kajian oleh Abubakar et al. (2019) menilai ketepatan dan kecekapan Snort, Suricata dan Zeek dalam mengesan pelbagai jenis serangan dan menyimpulkan bahawa Zeek mempunyai prestasi keseluruhan yang terbaik.

Sebagai tambahan kepada kajian akademik, laporan industri dan *whitepapers* juga menyerlahkan keberkesanan perisian IDS sumber terbuka ini. Sebagai contoh, laporan oleh Gartner (2020) menyenaraikan Zeek sebagai 'Cool Vendor' dalam bidang rangkaian dan keselamatan awan, menonjolkan keupayaannya dalam mengesan serangan canggih. Begitu juga, *whitepapers* oleh FireEye (2019) membincangkan penggunaan Suricata dalam platform pengesanan dan tindak balas ancaman serta mempamerkan keberkesanannya dalam senario dunia sebenar. Selain itu, pembangunan dan kemas kini berterusan perisian IDS sumber terbuka ini telah menjadikannya lebih cekap dan berkesan.

Dengan landskap ancaman yang semakin meningkat dan evolusi serangan yang berterusan adalah penting untuk terus menilai dan meningkatkan prestasi dan keberkesanan perisian IDS. Oleh itu, penyelidikan berterusan dalam bidang ini adalah penting. Satu kajian oleh Iqbal et al. (2020) mencadangkan rangka kerja baru untuk meningkatkan keupayaan pengesanan dan tindak balas Snort, Suricata dan Zeek, dengan menggabungkannya dengan algoritma pembelajaran mesin. Satu lagi kajian oleh Alghamdi et al. (2020) memberi tumpuan kepada mengoptimumkan set peraturan perisian ini untuk meningkatkan ketepatan pengesanan.

1.3 PERNYATAAN MASALAH

Pernyataan masalah prestasi dan keberkesanan dalam sistem pengesanan pencerobohan (IDS) adalah isu penting dalam bidang keselamatan rangkaian (Ru-Xin Wang et al., 2023). Dengan jumlah dan kecanggihan serangan siber yang terus meningkat, organisasi menghadapi ancaman berterusan terhadap data sensitif dan sistem aplikasi. Ini telah menyebabkan permintaan yang meningkat untuk penyelesaian IDS yang efisien dan boleh dipercayai yang dapat mengenal pasti dan mengurangkan penembusan berpotensi secara langsung. Perisian sumber terbuka seperti Snort, Suricata, dan Zeek telah mendapat perhatian daripada organisasi kerana keberkesanan kos dan fleksibiliti dalam pelaksanaannya. Penyelesaian perisian ini adalah sistem pengesanan dan pencegahan pencerobohan rangkaian (NIDPS) yang menawarkan pertahanan terhadap potensi ancaman terhadap data rangkaian. Snort dan Suricata adalah NIDPS sumber terbuka yang paling terkenal di pasaran (Gueltoum et al. 2023).

Namun, walaupun populariti perisian-perisian ini, masih terdapat kekurangan penyelidikan menyeluruh tentang prestasi dan keberkesanan mereka dalam mengesan dan mencegah penembusan. Ini menimbulkan cabaran bagi organisasi dalam memilih penyelesaian IDS yang paling sesuai untuk keperluan khusus mereka. Oleh itu, pernyataan masalah untuk penyelidikan ini adalah untuk menilai prestasi dan keberkesanan perisian IDS sumber terbuka, khususnya Snort, Suricata, dan Zeek, dalam mengesan dan mencegah penembusan dalam persekitaran rangkaian.

Salah satu cabaran dalam menilai prestasi IDS adalah penggunaan sumber perkakasan seperti CPU, memori dan rangkaian bagi memproses trafik dan mengesan

trafik yang masuk. Zeek memerlukan sumber pemprosesan yang tinggi berbanding Snort dan Suricata bagi mengesan dan menganalisis serangan ke atas rangkaian (Azad et al. 2019).

Selain prestasi, keberkesanan IDS juga merupakan faktor penting dalam memilih perisian yang tepat untuk pengesan penembusan. Ini merujuk kepada kemampuan IDS untuk mengesan dan mengurangkan pelbagai jenis serangan dengan tepat, termasuk ancaman yang diketahui berdasarkan pengesanan tandatangan yang telah ditetapkan pada set peraturan dan tidak dikenali tiada dalam set peraturan. Beberapa kajian (Abdelkader et al., 2019; Chen et al., 2021) telah menyoroti kelemahan perisian IDS sumber terbuka dalam mengesan serangan yang baru dan canggih. Dalam beberapa kes, IDS mungkin gagal mengesan serangan disebabkan pangkalan data tandatangan yang sudah lapuk atau ketidakmampuan untuk mengenal pasti corak serangan baru. Oleh itu, perlu ada penyelidikan untuk menilai keberkesanan perisian IDS sumber terbuka dalam mengesan pelbagai jenis serangan, termasuk ancaman yang dikenali dan tidak dikenali.

Selanjutnya, pelaksanaan dan konfigurasi IDS juga memainkan peranan penting dalam prestasi dan keberkesanannya. Setiap perisian IDS sumber terbuka mempunyai set ciri dan kebolehan sendiri, dan adalah penting untuk memahami bagaimana konfigurasi untuk mencapai hasil yang optimum. Ini memerlukan pemahaman menyeluruh tentang persekitaran rangkaian, beban trafik dan jenis serangan yang mungkin dihadapi oleh IDS. Kajian oleh Ahmed et al. (2020) telah mengenal pasti keperluan set peraturan dalam pelaksanaan dan konfigurasi perisian IDS sumber terbuka. Oleh itu, penyelidikan ini juga akan bertujuan untuk memberikan cadangan untuk pelaksanaan dan konfigurasi optimum Snort, Suricata, dan Zeek dalam persekitaran rangkaian yang berbeza..

1.4 OBJEKTIF KAJIAN

Objektif kajian adalah seperti berikut:

1. Untuk menilai prestasi penggunaan sumber seperti CPU, memori dan kapasiti rangkaian.

2. Untuk menilai ketepatan pengesanan ancaman siber termasuk perisian hasad dan serangan, contohnya DoS, DDoS, *web attack*.

1.5 PERSOALAN KAJIAN

Berdasarkan daripada pernyataan masalah yang telah dinyatakan terdapat dua persoalan kajian yang akan dijawab menerusi kajian ini seperti berikut:

1. Bagaimanakah prestasi Snort, Suricata dan Zeek dari segi ketepatan pengesanan dan penggunaan sumber?

2. Apakah keberkesanan Snort, Suricata dan Zeek dalam mengesan dan mencegah pelbagai jenis serangan siber?

1.6 SKOP KAJIAN

Skop kajian ini melibatkan perbandingan tiga perisian sumber terbuka iaitu Snort, Suricata dan Zeek. Kriteria yang digunakan untuk membandingkan ketiga-tiga IDS ini adalah jumlah serangan yang mampu dikesan, keberkesanan serangan dan penggunaan sumber yang digunakan untuk memastikan perisian IDS mana yang lebih baik iaitu dari jumlah serangan yang paling banyak dikesan serta penggunaan sumber yang ringan.

Kajian ini akan menggunakan set data CIC-IDS2017 dan CICDDoS2019 yang telah dibangunkan oleh Canadian Institute for Cybersecurity. Set data tersebut mengandungi set data rangkaian trafik yang komprehensif dalam format .pcap yang mengandungi serangan DDOS, Infilteration, Web Attacks, PortScan. Set data ini mempunyai 80 *features* berbeza serta 3,111,388 trafik rangkaian. Manakala set data CICDDoS2019 mengandungi 50,063,112 rekod trafik termasuk 50,006,249 baris serangan DDoS dan 56,863 *benign* dalam format pcap

1.7 KEPENTINGAN KAJIAN

Signifikan kajian ini memberikan penekanan kepada beberapa aspek penting seperti berikut:

1. Kajian ini menyumbang kepada peningkatan keselamatan rangkaian dengan menilai prestasi dan keberkesanan perisian pengesanan pencerobohan sumber terbuka (Snort, Suricata dan Zeek). Ini adalah penting dalam menghadapi ancaman siber yang semakin meningkat kerana ia memberi idea kepada organisasi tentang cara melindungi data dan sistem sensitif dengan lebih baik.

2. Kajian ini membantu organisasi memilih sistem IDS terbaik untuk keperluan organisasi. Pemahaman mengenai kebaikan dan keburukan Snort, Suricata dan Zeek dapat membantu organisasi membuat keputusan yang bijak tentang cara melaksanakan dan mengkonfigurasi perisian ini.

3. Kajian ini membantu dalam menggalakkan kerjasama dan perkongsian pengetahuan di kalangan profesional keselamatan, yang membawa kepada pemahaman yang lebih komprehensif mengenai keselamatan rangkaian.

4. Kajian ini menyumbang kepada pengetahuan akademik dan praktikal tentang keselamatan rangkaian. Ini akan menambah bilangan kajian akademik mengenai perisian pengesanan pencerobohan sumber terbuka. Kajian inni memberikan maklumat dan cadangan yang boleh digunakan oleh organisasi untuk meningkatkan kecekapan sistem pengesanan pencerobohan.

1.8 KESIMPULAN

Melalui kajian ini, satu perbandingan dan keberkesanan penggunaan IDS dalam membantu organisasi atau bisnes mengesan dan mencegah capaian tanpa kebenaran dalam rangkaian. Dengan pembuktian IDS boleh mengenal pasti aktiviti yang diragui dan amaran akan dihantar kepada pentadbir rangkaian serta penggunaan set peraturan yang lebih berkesan akan dibincangkan dalam Bab 4 dan 5. IDS boleh menjadi perisian tambahan kepada infrastruktur keselamatan organisasi dan meningkatkan prestasi rangkaian.

BAB II

KAJIAN KESUSASTERAAN

2.1 PENGENALAN

Bab II membincangkan mengenai ringkasan menyeluruh kajian terdahulu mengenai sistem pengesanan pencerobohan melalui artikel ilmiah, buku dan sumber lain yang relevan. Melalui kajian kesusasteraan ini, keupayaan dan kelemahan IDS serta model seterusnya penggunaan set data yang bersesuaian dapat dikenal pasti termasuk teknologi, metodologi dan perisian IDS.

2.2 SISTEM PENGESANAN PENCEROBOHAN

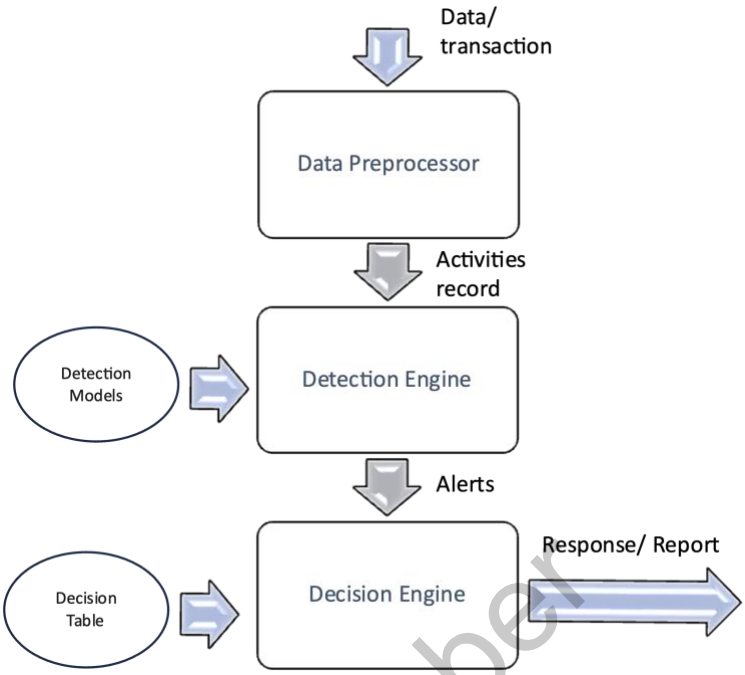
Perkembangan teknologi dan internet serta ancaman siber meningkat. Aspek penting dalam keselamatan siber ialah pengesanan dan pencegahan akses yang tidak dibenarkan dan aktiviti jahat dalam rangkaian komputer. Sistem Pengesanan Pencerobohan (IDS) merupakan komponen penting dalam infrastruktur keselamatan rangkaian yang direka untuk menganalisis trafik rangkaian bagi aktiviti mencurigakan atau potensi pencerobohan keselamatan (Dini et al. 2023). Fungsi utama adalah untuk mengenal pasti akses tidak dibenarkan, penyalahgunaan atau anomali yang mungkin menunjukkan kehadiran serangan siber atau kejadian keselamatan dalam rangkaian.

IDS dikategorikan kepada dua jenis utama berdasarkan fungsian iaitu IDS berasaskan tandatangan dan IDS berasaskan anomali (Gupta et al. 2019). IDS berasaskan tandatangan mengesan corak atau tandatangan yang diketahui aktiviti berbahaya dengan membandingkan trafik rangkaian dengan pangkalan data tandatangan yang telah ditentukan sebelumnya. Sekiranya terdapat padanan, amaran dijana. Manakala IDS berasaskan anomali menetapkan titik asas tingkah laku rangkaian yang normal dan menandakan sebarang penyimpangan daripada titik asas ini sebagai

pencerobohan berpotensi. Seterusnya IDS boleh ditempatkan pada pelbagai titik dalam rangkaian termasuk IDS berasaskan rangkaian (NIDS) untuk memantau trafik rangkaian pada titik strategik seperti sempadan rangkaian, *switch* atau *router* dan IDS berasaskan hos (HIDS) dipasang pada hos individu atau titik akhir untuk memantau aktiviti khusus hos tersebut termasuk perubahan sistem fail, log masuk dan penggunaan aplikasi (Safana Hyder Abbas et al, 2023).

Komponen utama IDS adalah *sensor* pencerobohan, mikropengawal dan pemberitahuan (Swati, Mirlekar et al, 2022). IDS mempunyai kekuatan mengesan penceroboh, memberikan amaran dan mempertahankan diri daripada serangan siber (Nitesh, Singh et al, 2020). Walau bagaimanapun, IDS juga mempunyai kelemahan seperti menghasilkan positif palsu, memerlukan pemantauan berterusan dan terdedah kepada serangan baru (Jiannan, Liu., 2020).

IDS berdasarkan teknik pemantauan, rekod data trafik, pengesanan serangan dan pelaporan amaran kepada pentadbir keselamatan. IDS yang berkesan dan efisien memastikan komponen IDS dilindungi dengan betul, termasuk pengguna, *sensor*, pelayan pangkalan data, pelayan pengurusan dan rangkaian seperti Rajah 2.1. Komponen ini adalah penting kerana ia disasarkan oleh penyerang yang ingin menghalang IDS daripada akses maklumat penting antaranya *vulnerabilities* atau mengesan serangan. Antara teknologi IDS seperti berasaskan rangkaian, tanpa wayar dan berasaskan hos. Setiap teknologi tersebut berbeza dari segi pengumpulan maklumat, rekod, pengesanan dan keupayaan pencegahan. Setiap teknologi menawarkan kelebihan seperti mengesan aktiviti tertentu dengan lebih cekap atau dengan ketepatan yang lebih tinggi. Resolusi serangan yang berkesan apabila menggunakan teknologi IDS memerlukan ketahanan dan kebolehpercayaan sistem, pengesanan pantas, positif palsu minimum, kadar pengesanan maksimum dan keupayaan untuk mengesan lokasi pencerobohan dengan tepat.



Rajah 2.1 Komponen Sistem Pengesanan Pencerobohan

Sumber: omcs notes

Fungsi utama IDS adalah untuk mengesan dan bertindak balas terhadap potensi pelanggaran keselamatan dalam masa nyata. IDS melakukan tugas pengesanan anomali, pengesanan tandatangan dan penjanaan amaran keselamatan. Pelbagai algoritma pembelajaran mesin seperti Support Vector Machine (SVM), Random Forest dan Decision Trees, boleh digunakan untuk pengesanan pencerobohan (Dipali Mane et al., 2023). IDS menganalisis trafik rangkaian dan mengkategorikannya sebagai normal atau tidak normal, menawarkan perisai untuk hos atau rangkaian tertentu. Ia mengkaji dan meramalkan aktiviti akses rangkaian pelanggan dan memproses komunikasi masuk melalui rangkaian.

2.2.1 Metrik Penilaian IDS

Menurut Merve Ozkan-Okay, Refik Samet (2020), untuk menilai IDS model yang dibangunkan dan membandingkan prestasi, metrik seperti *recall, false positive, false negative, precision, f-measure* dan ketepatan menggunakan matriks kekeliruan (confusion matrix) yang merangkumi *True Positive* (TP), *True Negative* (TN), *False Positive* (FP) dan *False Negative* (FN) seperti Jadual 2.1. *Precision* (juga dikenali

sebagai positive predictive value) mengukur nisbah sampel yang relevan yang dikenalpasti dengan betul sebagai positif. *Recall* (juga disebut sebagai ramalan positif) mengukur nisbah sampel yang relevan di antara sampel yang diambil. F-measure pula adalah min ketepatan dan penarikan semula harmoni, memberikan ukuran seimbang kedua-dua metrik. Manakala ketepatan mengukur ketepatan keseluruhan klasifikasi dengan mempertimbangkan kedua-dua *true positive* dan *true negative*. Metrik ini membantu dalam menilai prestasi model IDS dan membandingkan keberkesanannya dalam mengesan pencerobohan.

Jadual 2.1 Matriks Kekeliruan (Confusion Matrix)

Actual	**Predicted (Positive)**	**Predicted (Negative)**
Positive	TP	FN
Negative	FP	TN

Recall=TP/(TP+FN)
Precision=TP/(TP+FP)
F-Measure=(2*precision*recall)/(precision+recall)
Accuracy=TP+TN/(TAPI+TN+FP+FN)

Sumber: Merve Ozkan-Okay, Refik Samet (2021)

2.2.2 Cabaran IDS

Penggunaan teknologi dalam kehidupan semakin meningkat yang membawa kepada peningkatan ancaman dan serangan siber. Akibatnya, organisasi sentiasa mencari cara untuk melindungi sistem dan rangkaian daripada serangan siber. Salah satu penyelesaian utama ialah sistem pengesanan pencerobohan (IDS). IDS ialah mekanisme keselamatan yang direka untuk mengenal pasti dan mengelakkan akses yang tidak dibenarkan kepada rangkaian komputer.

Salah satu cabaran utama ialah keupayaan untuk mengesan dan mencegah serangan baru yang canggih dan sentiasa berkembang. Menurut kajian penyelidikan yang dijalankan pada tahun 2019 oleh Kaur dan Singh, IDS berasaskan tanda tangan telah menjadi kurang berkesan dalam mengesan serangan baru dan tidak diketahui. Ini kerana IDS berasaskan tanda tangan bergantung kepada satu set peraturan yang ditakrifkan untuk mengenal pasti serangan, menjadikannya rentan kepada serangan baru dan tidak diketahui.

Selain itu, penggunaan teknik enkripsi yang semakin meningkat dalam trafik rangkaian adalah sukar bagi IDS untuk mengesan dan menafsirkan trafik rangkaian dengan tepat. Seperti yang dicatat oleh kajian yang dijalankan oleh Dianatnia dan Shafiei pada tahun 2020, trafik yang dienkripsi merupakan cabaran utama bagi IDS kerana ia menghalang sistem daripada mengesan aktiviti dan corak jahat. Ini kerana trafik yang dienkripsi tidak dapat dibaca oleh IDS, menjadikannya sukar untuk menganalisis dan mengenal pasti apa-apa ancaman yang berpotensi.

Satu lagi cabaran yang dihadapi oleh IDS ialah kadar tinggi amaran dan amaran palsu, juga dikenali sebagai positif palsu. Amaran palsu ini disebabkan oleh aktiviti rangkaian yang sah atau anomali baik. Menurut satu kajian oleh Zhang et al. pada tahun 2021, positif palsu boleh mewakili sehingga 80% daripada semua amaran IDS, yang mengakibatkan pembaziran sumber dan sukar bagi pasukan keselamatan untuk memberi keutamaan dan bertindak balas kepada ancaman sebenar dalam masa yang tepat.

Selain itu, kurangnya integrasi dan koordinasi antara komponen IDS yang berbeza dan alat keselamatan lain merupakan cabaran utama lain. Seperti yang dinyatakan dalam kajian penyelidikan oleh Korba et al. pada tahun 2022, kurangnya keseragaman dan saling kendali antara IDS dan alat keselamatan lain boleh menyebabkan jurang dalam cakupan keselamatan dan sukar untuk diuruskan dan mengaitkan amaran dari sistem yang berbeza. Kekurangan kohesi ini boleh membawa kepada ancaman yang terlepas dan meningkatkan kerumitan keseluruhan pengurusan dan pemantauan keselamatan.

Akhirnya, evolusi dan kemajuan teknologi yang berterusan, bersama-sama dengan peningkatan ancaman siber, mewakili cabaran yang terus-menerus bagi IDS. Seperti yang dinyatakan dalam kajian oleh Mithun et al. pada tahun 2023, penyerang siber sentiasa mencari cara baru dan inovatif untuk mengelakkan langkah-langkah keselamatan menjadikannya sukar bagi IDS untuk mengikuti ancaman yang berkembang ini. Ini memerlukan kemas kini dan peningkatan berterusan kepada sistem IDS yang boleh menjadi mahal dan memakan masa bagi organisasi.

2.3 TEKNOLOGI PENGESANAN PENCEROBOHAN

Teknologi pengesanan pencerobohan adalah mekanisme keselamatan dinamik yang mengesan dan menghalang pencerobohan sistem. Ia melengkapkan mekanisme keselamatan tradisional seperti penyulitan dan *firewall* dengan menyediakan pemantauan pintar, pengesanan masa nyata dan keupayaan tindak balas dinamik (Ru-Xin Wang et al., 2023). Sistem pengesanan pencerobohan (IDS) mengimbas rangkaian atau sistem untuk aktiviti yang mencurigakan dan menyiasat tingkah laku bermusuhan (B.Ravi Teja et al., 2023). Teknologi IDS berbeza dalam pengesanan, konfigurasi, dan kos, dan boleh bergantung pada pengetahuan pakar keselamatan atau menggunakan teknik pembelajaran data dan mesin untuk mengurangkan pergantungan pada kepakaran manusia (Hind Tribak et al., 2022). Kaedah pembelajaran mesin, seperti *Bayesian classification, neural networks*, perlombongan data, dan *support vector machine* telah digunakan untuk sistem pengesanan pencerobohan untuk meningkatkan prestasi IDS (Lu Xin et al., 2021). Pengesanan pencerobohan adalah pelengkap berkesan untuk firewall dan membantu mengenal pasti penggunaan sumber yang tidak normal dan memberikan amaran awal mengenai serangan yang berpotensi.

2.3.1 Sistem Pengesanan Pencerobohan Berasaskan Rangkaian (NIDS)

Sistem pengesanan penyusupan rangkaian (NIDS) merupakan alat penting dalam landskap keselamatan siber yang sentiasa berkembang hari ini. Ia direka untuk mengesan akses tidak dibenarkan ke rangkaian serta mengenal pasti dan bertindak balas kepada apa-apa aktiviti berbahaya. Dengan meningkatnya bilangan serangan siber dan kerumitan seni bina rangkaian, NIDS yang berkesan adalah penting dalam melindungi data sensitif dan memastikan kesinambungan perniagaan. Menurut penyelidikan yang dijalankan pada tahun 2019, NIDS telah terbukti menjadi mekanisme pertahanan yang berkesan terhadap serangan rangkaian dengan kadar ketepatan lebih daripada 90%. (Wang et al., 2019). Ini menyoroti pentingnya mengemas kini dan meningkatkan NIDS secara berterusan untuk mengikuti kaedah penjenayah siber yang sentiasa berubah. Ia melibatkan penggunaan sensor untuk mengumpul paket dari rangkaian dan pelayan untuk menganalisis paket yang dikumpulkan dan mengesan sebarang tingkah laku yang tidak normal. Terdapat pelbagai jenis IDS berasaskan rangkaian, termasuk IDS berasaskan tandatangan dan IDS berasaskan anomali. IDS berasaskan tandatangan

bergantung pada tandatangan serangan yang telah ditetapkan untuk mengenal pasti tingkah laku berniat jahat, sementara IDS berasaskan anomali menggunakan teknik pembelajaran mesin untuk mengesan penyimpangan daripada tingkah laku rangkaian biasa. Perisian sumber terbuka seperti Snort dan Suricata biasanya digunakan untuk IDS berasaskan rangkaian. Penempatan strategik IDS berbilang sensor dalam persekitaran WAN bersama NIDS dapat meningkatkan pengesanan ancaman dan perlindungan sistem (Navya Iyengar, 2020; K.Azarudeen et al., 2023; Md.Shamim Towhid et al., 2023).

Dalam tahun-tahun kebelakangan ini, terdapat peralihan yang signifikan ke arah memasukkan teknik pembelajaran mesin dalam NIDS. Algoritma pembelajaran mesin mempunyai keupayaan untuk menganalisis sejumlah besar data dan mengenal pasti corak, menjadikannya sesuai untuk mengesan anomali dalam trafik rangkaian. Dalam kajian yang dijalankan pada tahun 2020, penyelidik mencadangkan NIDS berasaskan pembelajaran mendalam yang menunjukkan hasil yang menjanjikan dalam mengesan serangan yang diketahui dan tidak diketahui (Liu et al., 2020). Ini menunjukkan potensi pembelajaran mesin dalam meningkatkan ketepatan dan keberkesanan NIDS. Wang et al. (2022) mencadangkan IDS yang menggabungkan model deep neural network (DNN) dengan *principle component analysis* (PCA) untuk meningkatkan pengesanan ancaman dan meningkatkan keselamatan dan prestasi. Model yang dicadangkan mencapai kadar ketepatan 98%, yang merupakan kadar ketepatan tertinggi yang dilaporkan berdasarkan sejumlah besar serangan.

Aspek utama NIDS ialah keupayaan untuk menyesuaikan diri dengan ancaman baru dan muncul. Dengan peningkatan penggunaan perkhidmatan berasaskan awan dan *Internet of Things* (IoT), permukaan serangan bagi penjenayah siber telah berkembang. Ini memerlukan NIDS untuk dapat mengesan dan mencegah serangan yang menyasarkan teknologi baru ini. Dalam kajian yang dijalankan pada tahun 2022, penyelidik mencadangkan rangka kerja NIDS yang boleh menyesuaikan dinamik peraturan dan mekanisme pengesanan berdasarkan persekitaran rangkaian (Zhang et al., 2022). Ini menyoroti pentingnya mengemas kini dan meningkatkan NIDS secara berterusan untuk mengikuti landskap teknologi dan ancaman yang berubah.

Selain itu, keberkesanan NIDS juga bergantung kepada keupayaan untuk bertindak balas dengan cepat terhadap ancaman yang dikesan. Dalam kes serangan, penundaan dalam tindak balas boleh menyebabkan kerosakan yang signifikan kepada rangkaian. Untuk menangani isu ini, penggunaan kecerdasan buatan untuk mengautomasikan proses respons insiden (Chen et al., 2023). Ini dapat mengurangkan masa respons dan membolehkan pertahanan yang lebih cekap dan berkesan terhadap serangan.

2.3.2 Sistem Pengesanan Pencerobohan Berasaskan Hos (HIDS)

HIDS mempunyai pelbagai ciri keselamatan untuk melindungi sistem hos daripada pencerobohan dan serangan. HIDS mengumpulkan maklumat mengenai sistem hos termasuk pemantauan integriti fail, log sistem dan log aktiviti pengguna. Seterusnya menganalisis log sistem untuk mengesan sebarang aktiviti yang mencurigakan atau tidak dibenarkan seperti percubaan akses yang tidak dibenarkan atau perubahan pada fail sistem kritikal. HIDS memberikan makluman masa nyata kepada pentadbir sistem keselamatan apabila terdapat aktiviti yang mencurigakan atau pencerobohan yang berpotensi dikesan, membolehkan tindak balas dan mitigasi tepat pada masanya.

Chawla et al. (2018) mencadangkan HIDS baru yang mengenal pasti tingkah laku normal sistem berdasarkan urutan sistem panggilan. Kajian ini menerangkan sistem IDS berasaskan anomali yang cekap dari segi pengiraan berdasarkan *Recurrent Neural Network* (RNN). RNN adalah *neural network* yang dapat memproses data berurutan dengan menggunakan gelung maklum balas untuk menyampaikan maklumat dari satu langkah ke langkah seterusnya. Dengan menggunakan *Gated Repetitive Units* (GRU) dan bukan menggunakan rangkaian LSTM (Memori Jangka Panjang) biasa untuk mencapai hasil dengan penurunan masa latihan. Menggabungkan GRU dan *Convolutional Neural Network* (CNN) meningkatkan IDS berasaskan anomali. Teknik yang dicadangkan diterapkan pada set data ADFA (Australian Defense Force Academy) yang merupakan set data yang tersedia secara umum untuk menilai IDS. Keputusan yang diperoleh menunjukkan bahawa CNN/GRU yang disusun kira-kira 10 kali lebih cepat daripada LSTM kerana penumpuan yang lebih cepat dalam latihan. Selain itu, Kadar Pengesanan Benar (True Detection Rate) telah mencapai 100% dan 60% kadar penggera palsu (false alarm rate) dengan sistem yang dicadangkan. Kajian ini dapat

ditingkatkan dengan meningkatkan jumlah sampel latihan atau set data yang berbeza dapat membantu mengesahkan teknik yang dicadangkan.

Menurut Byrnes et al. (2020), set data yang lama boleh menjadi usang atau tidak sah kerana perubahan sistem pengoperasian dan kompleksiti, menjadikan HIDS sukar untuk mengenal pasti ancaman terkini. Kajian ini untuk merapatkan jurang antara model teori dan persekitaran aplikasi dunia nyata dengan memeriksa versi terbaru kernel Linux 5.7.0-rc1 untuk menjana data dan meningkatkan model pengesanan. Mengikuti perkembangan terkini dalam sistem pengoperasian dan penyesuaian HIDS dapat memastikan keberkesanan berterusan dalam mengesan dan mencegah pencerobohan.

Jenis serangan yang dikesan oleh HIDS bergantung kepada teknik pengesanan yang digunakan dalam sistem. HIDS mempunyai maklumat terperinci ciri-ciri hos dan konfigurasi. Agen IDS boleh mengenal pasti sama ada serangan ke atas hos akan berjaya jika tidak dihentikan.

2.3.3 Sistem Pengesanan Pencerobohan Sumber Terbuka

Penggunaan teknologi sumber terbuka dalam sistem pengesanan pencerobohan rangkaian (NIDS) telah mendapat perhatian yang signifikan. NIDS merujuk kepada koleksi alat dan teknik yang digunakan untuk mengesan dan mencegah akses yang tidak dibenarkan ke rangkaian komputer. NIDS sumber terbuka menggunakan perisian yang tersedia secara bebas dan boleh diubahsuai oleh pengguna untuk memenuhi keperluan spesifik. Pendekatan ini mempunyai beberapa kelebihan termasuk fleksibiliti, kecekapan kos dan sokongan komuniti. Penggunaan NIDS sumber terbuka telah menjadi semakin popular kerana permintaan yang semakin meningkat untuk penyelesaian keselamatan yang boleh disesuaikan dan berpatutan dalam organisasi (Bahl et al. 2020).

Salah satu kelebihan utama menggunakan teknologi sumber terbuka dalam NIDS ialah fleksibiliti. Dengan perisian sumber terbuka, pengguna mempunyai akses kepada kod sumber, membolehkan pengubahsuaian sistem untuk memenuhi keperluan keselamatan rangkaian. Ini bertentangan dengan NIDS komersial, di mana kod sumber tidak boleh diakses, membatasi keupayaan pengguna untuk membuat perubahan.

Seperti yang dinyatakan oleh Goyal et al. (2019), fleksibiliti NIDS sumber terbuka membolehkan integrasi yang lebih baik dengan alat-alat keselamatan lain dan membenarkan pengguna untuk menangani keperluan keselamatan rangkaian yang unik secara berkesan.

Satu lagi kelebihan utama NIDS sumber terbuka ialah kosnya. Ini adalah disebabkan perisian ini boleh didapati secara percuma dan penjimatan kos untuk membeli NIDS komersial. Ini sangat menguntungkan untuk perniagaan kecil dan menengah (SME) yang mungkin tidak mempunyai sumber kewangan untuk melabur dalam penyelesaian keselamatan yang mahal. Menurut kajian oleh Mishra et al. (2022), keberkesanan kos NIDS sumber terbuka telah menjadikannya pilihan yang popular di kalangan SME dan membolehkan keselamatan rangkaian ditingkatkan dengan kos minima.

Walaupun mempunyai banyak kelebihan, NIDS sumber terbuka juga mempunyai beberapa batasan. Salah satu kebimbangan utama dengan teknologi sumber terbuka ialah kurangnya sokongan teknikal rasmi. Seperti yang dinyatakan oleh Nandi dan Sen (2021), kurangnya sokongan profesional boleh menjadi kelemahan yang signifikan bagi organisasi yang mungkin tidak mempunyai kepakaran untuk menangani isu-isu teknikal yang kompleks. Ini boleh menyebabkan penundaan dalam penyelesaian ancaman keselamatan yang berpotensi membahayakan rangkaian organisasi. Oleh itu, pasukan profesional yang mahir yang boleh mengurus dan mengekalkan NIDS sumber terbuka diperlukan dalam sesuatu organisasi.

Kesimpulannya, penggunaan teknologi sumber terbuka dalam sistem pengesanan pencerobohan rangkaian telah muncul sebagai penyelesaian yang berkesan dan fleksibel bagi organisasi. Keupayaan untuk menyesuaikan perisian, ketersediaan, dan sokongan komuniti yang bersemangat menjadikannya pilihan yang menarik untuk organisasi semua saiz. Walau bagaimanapun, kurangnya sokongan teknikal rasmi boleh menimbulkan cabaran bagi sesetengah organisasi, memerlukan mereka untuk mempunyai pasukan yang mahir untuk menguruskan NIDS sumber terbuka secara berkesan. Apabila teknologi terus berkembang, diharapkan bahawa NIDS sumber

terbuka akan memainkan peranan penting dalam melindungi rangkaian terhadap ancaman siber yang muncul.

2.4 METODOLOGI PENGESANAN PENCEROBOHAN

Metodologi pengesanan pencerobohan adalah pendekatan yang digunakan untuk mengesan dan mencegah serangan siber dalam rangkaian komputer atau sistem komputer. Ianya melibatkan perlombongan data, analisis statistik dan teknik pembelajaran mesin.

Metodologi pengesanan pencerobohan terdiri daripada tiga kategori iaitu Model berasaskan tandatangan (Signature-based model), Model berasaskan anomali (Anamoly-based model) dan Analisis protokol stateful (Stateful protocol analysis). Setiap metodologi IDS menggunakan teknik yang berbeza untuk mengenal pasti serangan rangkaian. Model berasaskan tandatangan dapat mengesan serangan yang diketahui dengan pantas dan berkesan tetapi gagal untuk mengesan serangan zero-day. Model berasaskan anomali berkesan untuk mengesan serangan rangkaian tidak diketahui sebelumnya tetapi menimbulkan penggera palsu. Dalam erti kata lain, ia mengklasifikasi trafik biasa sebagai serangan. Manakala model protokol *stateful* boleh mengesan sebahagian daripada serangan baru, ia adalah intensif sumber, kompleks dan tidak dapat mengesan serangan pintar (Khraisat et al. 2019).

2.4.1 Model Berasaskan Tandatangan

Pengesanan berasaskan tandatangan dikenali juga sebagai pengesanan berasaskan pengetahuan yang mempunyai satu pangkalan data ancaman yang telah dikenal pasti di peringkat awal. Pengesanan berasaskan tandatangan ialah kaedah pengesanan yang paling mudah kerana peristiwa atau aktiviti yang diperhatikan disemak terhadap senarai tandatangan dengan menggunakan proses perbandingan. Jika terdapat aktiviti serangan yang ditakrifkan sebelum ini dalam senarai, amaran akan dihasilkan. IDS berasaskan tandatangan sangat berkesan untuk mengesan ancaman yang diketahui berdasarkan senarai tandatangan, tetapi sebahagian besarnya tidak berkesan untuk mengesan ancaman atau varian yang diketahui sebelum ini adalah ancaman.

IDS berasaskan tanda tangan telah terbukti berkesan dalam mengesan serangan yang diketahui dengan ketepatan yang tinggi (Sinha & Saha, 2019). Walau bagaimanapun, dengan kemajuan teknik serangan dan munculnya ancaman baru yang berterusan, model-model ini menghadapi cabaran dalam mengekalkan landskap serangan siber yang sentiasa berubah. Salah satu peningkatan pendekatan tersebut ialah penggunaan algoritma pembelajaran mesin untuk meningkatkan keupayaan pengesanan IDS. Menurut kajian Alam et al. (2020), IDS berasaskan satu set pengelas pembelajaran mesin telah dicadangkan, yang menunjukkan peningkatan yang signifikan dalam mengesan serangan yang tidak diketahui. Ini menyoroti potensi menggabungkan teknik pembelajaran mesin ke dalam IDS berasaskan tanda tangan untuk menjadikannya lebih kukuh dan boleh disesuaikan.

Aspek lain daripada metodologi model berasaskan tanda tangan untuk IDS yang telah mendapat perhatian adalah pemilihan tanda tangan yang bersesuaian. Apabila bilangan tanda tangan meningkat, masa pemprosesan dan kadar positif palsu mungkin meningkat. Oleh itu, memilih tanda tangan yang paling relevan dan berkesan adalah penting untuk berfungsi dengan cekap IDS. Satu kajian oleh Ahmed et al. (2020) mencadangkan teknik pemilihan ciri menggunakan algoritma genetik untuk mengenal pasti tanda tangan yang paling penting, yang membawa kepada ketepatan pengesanan yang lebih baik dan masa pemprosesan yang lebih pendek. Kajian oleh Kaur et al. (2021) mencadangkan seni bina IDS berasaskan tanda tangan yang membahagikan pangkalan data tanda tangan di antara pelbagai nod, yang membawa kepada kebolehskalaan yang lebih baik dan masa pemprosesan yang berkurangan.

Landskap ancaman terus berkembang, ia adalah penting bagi IDS berasaskan tanda tangan untuk mempunyai keupayaan untuk mengesan serangan yang tidak diketahui atau *zero-day*. Untuk menangani isu ini, penyelidik telah mencadangkan penggunaan IDS hibrid, yang menggabungkan kekuatan pendekatan berasaskan tanda tangan dan anomali. Dalam kajian oleh Sood et al. (2019), IDS hibrid telah dicadangkan, yang menunjukkan ketepatan yang lebih tinggi dalam mengesan kedua-dua serangan yang diketahui dan tidak diketahui berbanding dengan IDS berasaskan tanda tangan tradisional.

Malek et al. (2020) mencadangkan sistem baru untuk mengesan pencerobohan dalam rangkaian komputer. Sistem ini menggunakan satu set peraturan sebagai enjin pengecaman corak yang dipanggil model Pattern Based Intrusion Detection (PIBD). Model PIBD diuji pada set data yang dihasilkan dalam skop kajian dan mencapai kadar ketepatan 75%. Gabungan hasil eksperimen, pendekatan Statistical Based Intrusion Detection (SIBD) dan PBID menyediakan sistem komprehensif untuk pengesanan pencerobohan. Walau bagaimanapun, hasil yang berkesan tidak dapat diperoleh dengan hanya pengesanan serangan berasaskan tandatangan. Oleh itu,.

IDS berasaskan tandatangan mengesan serangan berdasarkan corak tertentu seperti bilangan bait atau nombor 1s atau bilangan 0s dalam trafik rangkaian. Ia juga mengesan berdasarkan urutan arahan berniat jahat yang sudah diketahui yang digunakan oleh perisian berniat jahat (malicious). Corak yang dikesan dalam IDS dikenali sebagai tandatangan. IDS berasaskan tandatangan dengan mudah boleh mengesan serangan yang coraknya (tandatangan) sudah wujud dalam sistem tetapi agak sukar untuk mengesan serangan perisian berniat jahat (malicious) baharu kerana corak (tandatangan) mereka tidak diketahui. Salah satu IDS yang menggunakan berasaskan tandatangan adalah Suricata yang mengandungi *action*, *header* dan *rule options*. Contoh peraturan seperti Rajah 2.2.


```
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"HTTP GET Request
Containing Rule in URI"; flow:established,to_server; http.method;
content:"GET"; http.uri; content:"rule"; fast_pattern; classtype:bad-
unknown; sid:123; rev:1;)
```

Rajah 2.2 Contoh peraturan Suricata menggunakan teknik pengesanan berasaskan tandatangan

Sumber: Suricata Documentation

2.4.2 Model Berasaskan Anomali

Pengesanan berasaskan anomali membandingkan aktiviti yang diperhatikan dengan definisi biasa untuk mengenal pasti keadaan atau aktiviti yang tidak normal. IDS menggunakan pengesanan berasaskan anomali mempunyai peraturan yang mewakili

tingkah laku normal dan menggunakan kaedah statistik untuk mengesan aktiviti yang jauh lebih tinggi daripada yang dijangkakan. Pengesanan berasaskan anomali berkesan dalam mengesan jenis serangan yang tidak diketahui sebelumnya. Peraturan untuk pengesanan berasaskan anomali boleh menjadi statik atau dinamik dengan profil dinamik sentiasa dikemas kini. IDS berasaskan anomali dapat menghasilkan positif palsu dan mengalami kesukaran menentukan punca amaran dalam persekitaran dinamik.

Ia juga digunakan berdasarkan perubahan tingkah laku (behavior) ancaman. Dalam IDS berasaskan anomali terdapat penggunaan pembelajaran mesin untuk mencipta model aktiviti yang boleh dipercayai dan apa-apa yang datang dibandingkan dengan model itu dan ia diisytiharkan mencurigakan jika ia tidak terdapat dalam model. Kaedah berasaskan pembelajaran mesin mempunyai sifat umum yang lebih baik berbanding dengan IDS berasaskan tandatangan kerana model ini boleh dilatih mengikut aplikasi dan konfigurasi perkakasan. Kaedah pembelajaran mesin ini boleh menangani kelemahan kepada serangan sifar hari (zero day attack) dan lebih mudah untuk di konfigurasi dan sentiasa dikemaskini.

Menurut kajian oleh Al-Shaer et al. (2019), model berasaskan anomali boleh mengesan serangan yang tidak diketahui dan terkini yang tidak boleh dikesan oleh model berdasarkan tanda tangan. Di samping itu, Kim et al. (2020) menggunakan pendekatan berasaskan pembelajaran mendalam untuk pengesanan anomali dalam sistem pengesanan pencerobohan menunjukkan hasil yang baik dalam mengesan serangan berbanding kaedah pembelajaran mesin tradisional.

Satu kajian oleh Tama et al. (2019) mencadangkan kaedah untuk memilih ciri yang relevan dan sistem pengesanan pencerobohan menggunakan *ensemble* pengklasifikasian dua peringkat, mencapai kadar ketepatan tinggi 85.8% dan 91.3% pada set data yang berbeza mengatasi teknik klasifikasi lain.

Selain itu, kajian oleh Jiang et al. (2021) mencadangkan model hibrid yang menggabungkan pembelajaran mendalam dan analisis data besar untuk pengesanan pencerobohan. Hasil menunjukkan bahawa model ini mampu mencapai kadar

pengesanan yang tinggi dan kadar positif palsu yang rendah. Ini menekankan potensi penggunaan data besar dan teknik pembelajaran mesin canggih dalam pembangunan model berasaskan anomali.

2.4.3 Analisis Protokol Stateful

Analisis protokol stateful membandingkan keadaan atau aktiviti yang diperhatikan dengan profil yang telah ditetapkan untuk mengenal pasti penyimpangan dan berdasarkan profil yang menentukan bagaimana protokol harus digunakan. Ianya membolehkan IDS memahami dan memantau keadaan protokol rangkaian, pengangkutan dan aplikasi membezakan antara keadaan yang tidak disahkan dan disahkan.

Rashid et al. (2020) menjalankan analisis komprehensif dan perbandingan dua set data iaitu NSL-KDD dan CIDDS-001 untuk menilai prestasi IDS. Network Neural dan kaedah pemilihan ciri hibrid digunakan untuk mendapatkan hasil yang ideal sebelum menerapkan pendekatan klasifikasi seperti Naïve Bayes, SVM k-NN, DAE, dan DNN. Kaedah pemilihan dan kedudukan ciri hibrid menggunakan gabungan teknik analisis protokol berasaskan tandatangan, berasaskan anomali dan *stateful* untuk memilih ciri yang paling relevan untuk klasifikasi. Hasil eksperimen menunjukkan bahawa pengelas k-NN, SVM, NN, dan DNN mempunyai ketepatan kira-kira 100% dalam set data NSL-KDD dan kira-kira 99% ketepatan dalam kumpulan data pengelas K-nn dan Naïve Bayes CIDDS-001. Kajian menunjukkan bahawa menerapkan pendekatan pembelajaran mendalam yang berbeza pada set data semasa seperti CIC-IDS 2017, CIC-IDS 2018 dapat meningkatkan lagi prestasi IDS.

Kaedah analisis protokol stateful/keadaan menggunakan model protokol standard untuk mengesan urutan arahan yang tidak dijangka dan mengikuti profil dalam kedua-dua lapisan rangkaian dan aplikasi. Walau bagaimanapun, kekurangan maklumat protokol terperinci dan ketidakpatuhan terhadap piawaian atau penambahan ciri khas oleh vendor boleh mengehadkan ketepatan analisis.

Kaedah analisis protokol yang berkenaan mempunyai kelemahan seperti proses analisis yang kompleks, penggunaan sumber yang tinggi, ketidakupayaan untuk

mengesan jenis serangan tertentu yang tidak melanggar tingkah laku protokol dan variasi dalam pelaksanaan model protokol merentasi sistem operasi atau pelanggan/pelayan yang berbeza.

2.5 PENDEKATAN PENGESANAN PENCEROBOHAN

Terdapat beberapa pendekatan untuk mengesan pencerobohan dalam rangkaian komputer, yang merangkumi pendekatan statistik, berasaskan peraturan, heuristik, berasaskan corak, berasaskan awan, berasaskan pembelajaran mesin, dan pendekatan berasaskan pembelajaran mendalam.

IDS berasaskan statistik memantau transaksi biasa untuk membuat profil yang sah, memberikan skor kepada penyimpangan dari profil ini dan mencetuskan penggera jika skor melebihi nilai ambang berdasarkan bilangan peristiwa dalam tempoh masa tertentu. Pelbagai metrik statistik digunakan untuk membina profil normal, dan tekniknya dapat dikelaskan kepada model *univariate*, *multivariate* dan siri masa (Khatri et al., 2023).

IDS berasaskan peraturan menggunakan teknologi untuk mengesan potensi pencerobohan dalam lalu lintas rangkaian dengan menerapkan peraturan yang telah ditetapkan boleh diperoleh daripada corak serangan menggunakan kecerdasan buatan. Pendekatan ini memerlukan peraturan yang lebih sedikit berbanding dengan IDS berasaskan tandatangan dan membolehkan penyelenggaraan dan pengesanan serangan baru dengan mudah (Khraisat et al., 2019).

IDS berasaskan heuristik mengesan pencerobohan dengan menganalisis jejak pelaksanaan untuk tingkah laku yang mencurigakan, menimbulkan amaran apabila corak dijumpai, dan dapat mengesan serangan yang diketahui dan tidak diketahui, walaupun beberapa serangan mungkin menggunakan teknik persembunyian untuk mengelakkan pengesanan (Cyntia Vargas Martinez et al., 2019).

Pendekatan berasaskan corak dalam sistem pengesanan pencerobohan mengenal pasti dan mengekstrak corak yang bermakna dalam data yang dikumpulkan untuk mengesan serangan yang diketahui dengan cekap, tetapi mungkin tidak berkesan

terhadap serangan yang tidak diketahui. Algoritma pemadanan corak berganda lebih cekap untuk IDS semasa, tetapi memerlukan lebih banyak memori dan masa pemprosesan (Narendar Kumboji, 2020).

IDS berasaskan awan terdiri daripada pengumpul data pengguna, perkhidmatan awan, dan komponen pengesanan pencerobohan awan, yang menganalisis dan mengesahkan data untuk mengesan pencerobohan. IDS di awan membawa faedah seperti mengenal pasti aktiviti rangkaian berniat jahat dan mengesan pelbagai jenis serangan rangkaian secara selari (Suman et al., 2022).

Pembelajaran mesin (ML) adalah pendekatan algoritma untuk mengautomasikan analisis data, dengan teknik yang berbeza seperti pembelajaran yang diselia, tanpa pengawasan, dan separa diselia yang digunakan dalam IDS. Antara kelebihan seperti kebolehsuaian, prestasi tinggi, fleksibiliti, dan keupayaan untuk mengesan jenis serangan baru. Walau bagaimanapun, ia juga mempunyai kelemahan seperti membuat andaian mengenai data, terdedah kepada berat sebelah, kesukaran menangani kelebihan, dan cabaran dalam mengesan dan mencegah serangan yang tidak diketahui (Ayesha S Dina et al., 2021; Ashish Tripathi et al., 2021).

Pembelajaran mendalam (Deep Learning) adalah subbidang pembelajaran mesin yang menghilangkan keperluan untuk kejuruteraan ciri dan menggunakan pelbagai lapisan untuk mengekstrak ciri hierarki. Ia telah berjaya diterapkan dalam pelbagai bidang seperti pemprosesan imej, pengiktirafan manusia, dan pengesanan perisian hasad. Sistem pengesanan pencerobohan berasaskan pembelajaran mendalam dapat mengesan anomali, mengklasifikasikan lalu lintas sebagai normal atau serangan dan menentukan jenis serangan (Wen-hao Lv et al., 2020; Zeeshan Ahmad et al., 2021). Kelebihan pembelajaran mendalam dalam IDS termasuk pengekstrakan ciri automatik, mengendalikan set data yang besar, dan mengurangkan ruang ciri, sementara kelemahannya termasuk keperluan pengetahuan konteks, kekurangan pakar yang terlatih, dan kerentanan terhadap serangan pengelakan IDS berasaskan pembelajaran mendalam menggunakan algoritma pembelajaran mendalam untuk mengesan serangan. Algoritma ini dapat belajar dari sejumlah besar data dan mengenal pasti corak serangan kompleks yang mungkin sukar dikesan menggunakan pendekatan lain.

Pendekatan pengesanan pencerobohan yang berbeza mempunyai kelebihan dan kekurangannya sendiri dan prestasinya berbeza-beza bergantung pada set data; pendekatan statistik, heuristik dan berasaskan corak yang biasa digunakan, tetapi fokus diberikan kepada pendekatan awan, pembelajaran mesin dan pembelajaran mendalam, serta mempertimbangkan pelbagai teknik pengelakan.

2.6 SET DATA DIGUNAKAN DALAM IDS

IDS digunakan untuk mengesan dan mengklasifikasikan serangan rangkaian. Penilaian keberkesanan IDS diuji menggunakan set data. Set data IDS dibangunkan dan boleh dipercayai dimana aliran rangkaian dikumpulkan menggunakan perisian analisis paket. Aliran rangkaian terdiri daripada alamat IP sumber dan destinasi, port sumber dan destinasi, panjang paket, jenis perkhidmatan rangkaian, dan percubaan log masuk yang gagal (G. Bovenzi et al., 2020). Aliran rangkaian yang dikumpulkan kemudian dianalisis secara manual atau secara automatik. Ciri-ciri ini digunakan oleh IDS untuk mengekstrak corak serangan dan mengesan serangan rangkaian.

Siasatan ke atas beberapa set data yang tersedia secara umum KDD '99, CAIDA, NSL-KDD, ADFA-LD dan ADFA-WD, AWID, UNSW-NB15 dan CICIDS untuk sistem pengesanan pencerobohan. Set data ini terkenal untuk pengesanan pencerobohan rangkaian dan digunakan dalam banyak kajian saintifik dan perniagaan. Set data dan ciri-ciri IDS dikemas kini dari semasa ke semasa kerana jenis serangan rangkaian berubah mengikut teknologi dan kepakaran penceroboh. Ini adalah perlu untuk memenuhi keperluan semasa IDS (J. Verma et al., 2020).

Penggunaan set data untuk menilai perisian IDS dari segi prestasi dan keberkesanan sesuatu IDS. Sehubungan itu, penerangan ringkas ciri-ciri, kelebihan dan kekurangan sesuatu set data seperti Jadual 2.2.

2.6.1 Set Data KDD’99

Set data pertama yang digunakan untuk sistem pengesanan pencerobohan iaitu set data Defense Advanced Research Project Agency (DARPA) (A.Thakkar et al., 2020). Set data DARPA dibangunkan pada tahun 1998 oleh Agensi Projek Penyelidikan Lanjutan

Pertahanan di MIT Lincoln LAB (A.Khairasat et al., 2019). Set data terdiri daripada pembuangan paket TCP data mentah yang bermaksud bahawa ia mengandungi sejumlah besar data yang tidak diproses. Oleh kerana sifat mentah data, algoritma klasifikasi pembelajaran mesin tidak dapat dilakukan pada set data tanpa mengekstrak ciri terlebih dahulu.

Pada tahun 1999, versi set data DARPA yang diekstrak dari ciri telah dicadangkan, yang disebut set data Penemuan Pengetahuan dan Perlombongan Data (KDD '99). Set data KDD '99 mempunyai kelas berlabel yang mengkategorikan lalu lintas rangkaian kepada lima kumpulan iaitu serangan DoS (penolakan perkhidmatan), serangan jauh ke tempatan (R2L), serangan pengguna ke jauh (U2R), serangan siasatan dan normal. Set data mengandungi 24 serangan berbeza untuk latihan dan 14 serangan yang tidak diketahui untuk ujian. Walau bagaimanapun, jenis serangan dan label kelas biasa tidak sama rata diedarkan dalam set data, yang boleh menjejaskan ketepatan sistem pengesanan pencerobohan (Siddique et al. 2019).

Akhirnya, set data tidak mengandungi serangan rangkaian baru-baru ini, yang dapat mengehadkan keberkesanan sistem pengesanan pencerobohan dalam mengesan ancaman baru dan muncul.

2.6.2 Set Data CAIDA

Set data CAIDA adalah koleksi jejak trafik rangkaian dari tahun 2007 yang telah dinyatakan tanpa nama untuk melindungi privasi pengguna (Ahmed et al. 2021). Set data ini termasuk contoh serangan DDoS, yang merupakan percubaan berniat jahat untuk menyekat pengguna yang sah daripada mengakses pelayan yang disasarkan.

Walau bagaimanapun, set data CAIDA mempunyai beberapa batasan. Sebagai contoh, ia tidak mengandungi pelbagai serangan rangkaian yang bermaksud bahawa ia mungkin tidak mewakili semua jenis serangan yang mungkin berlaku dalam senario dunia nyata. Di samping itu, set data CAIDA tidak termasuk data rangkaian lengkap yang boleh mengehadkan kegunaannya untuk jenis soalan penyelidikan tertentu. Kelemahan lain dari set data CAIDA adalah bahawa ia tidak dilabel dan penyelidik

perlu mengenal pasti serangan atau peristiwa lain yang relevan dalam data secara manual.

Akhirnya, set data CAIDA hanya merangkumi 20 ciri, yang mungkin tidak mencukupi untuk beberapa jenis tugas analisis atau pemodelan.

2.6.3 Set Data NSL-KDD

NSL-KDD adalah set data yang merupakan versi set data KDD99 yang lebih baik. Set data KDD mempunyai beberapa ciri yang diulang menyebabkan penurunan prestasi pembelajaran mesin dari masa ke masa. Peratusan rekod pendua dalam set data KDD sangat besar yang menjadikannya cabaran untuk menganalisis menggunakan pembelajaran mesin.

Pada tahun 2009, Tavallaee et al. (2009) mencipta set data NSL-KDD dari KDD, yang membuang rekod pendua dan mengurangkan saiz set data. Set data latihan NSL-KDD mempunyai 125,973 contoh dan set data ujian mempunyai 22,544 contoh. Set data NSL-KDD mempunyai 22 label serangan pencerobohan untuk latihan dengan 41 ciri. Set data NSL-KDD sesuai untuk menguji IDS semasa.

Untuk meningkatkan prestasi pembelajaran mesin, kejuruteraan ciri dapat diterapkan pada set data NSL-KDD sebelum melakukan algoritma pembelajaran mesin.

2.6.4 Set Data UNSWB-NB15

UNSW-NB15 adalah set data yang dibangunkan pada tahun 2015 untuk menguji IDS. Set data terdiri daripada trafik rangkaian komprehensif yang dihasilkan menggunakan alat IXIA (Sarhan et al., 2021). Trafik rangkaian mentah direkod menggunakan tcpdump dan mengandungi 49 ciri, 9 jenis serangan unik dan 2,540,044 rekod.

Set data dibahagikan kepada empat fail csv, iaitu UNSW-NB15_1.csv, UNSW-NB15_2.csv, UNSW-NB15_3.csv, dan UNSW-NB15_4.csv. Set latihan set data mengandungi 175,341 rekod, sementara set ujian mengandungi 82,332 rekod dari 9 kategori serangan dan trafik biasa. Set data UNSW-NB15 dianggap baik untuk menguji

IDS moden kerana mengandungi jenis serangan baru dan ciri komprehensif. Set data boleh digunakan untuk menilai prestasi IDS dalam mengesan pelbagai jenis serangan, termasuk serangan DoS, siasatan, malware, dan shellcode.

Set data telah digunakan secara meluas dalam kajian penyelidikan untuk membangunkan dan menilai IDS baru dan untuk membandingkan prestasi yang sedia ada. Ketersediaan set data telah menyumbang kepada kemajuan penyelidikan dalam bidang pengesanan pencerobohan dan telah memudahkan pengembangan ID yang lebih berkesan dan cekap.

2.6.5 Set Data CIC-IDS2017

Set data CIC-IDS dibangunkan pada tahun 2017 untuk tujuan menguji dan menilai IDS. Set data mengandungi kedua-dua trafik rangkaian biasa dan pelbagai jenis serangan, menjadikannya sumber yang berharga untuk penyelidik di lapangan. Data dikumpulkan dari pelbagai sumber, termasuk modem, switch, router, firewall dan sistem operasi yang berbeza seperti Windows, Linux dan macOS.

Set data merangkumi beberapa profil serangan yang berbeza, seperti serangan kekuatan kasar pada FTP dan SSH, serangan penyusupan, serangan penolakan perkhidmatan (DoS), serangan penolakan perkhidmatan (DDoS) yang diedarkan, serangan Heartbleed, serangan botnet, dan serangan web (A.R. Sonule et al., 2020). Profil serangan ini mewakili pelbagai ancaman yang berpotensi terhadap keselamatan rangkaian menjadikan set data perisian yang berguna untuk menilai keberkesanan sistem pengesanan pencerobohan. Set data CICIDS terdiri daripada 80 ciri yang boleh digunakan untuk melatih model pembelajaran mesin untuk pengesanan pencerobohan.

2.6.6 Set Data CIC-DDoS2019

Set data CIC-DDOS2019 adalah koleksi data yang merangkumi kedua-dua trafik biasa dan berniat jahat khususnya serangan Distributed Denial of Service (DDoS). Set data besar dengan ratusan ribu rekod dan 88 ciri. Data disusun ke dalam fail.csv berasingan berdasarkan jenis serangan DDoS, dengan kategori termasuk NTP, DNS, LDAP,

MSSQL, NetBIOS, SNMP, SSDP, UDP, UDP-LAG, WebDDoS, SYN, dan TFTP (Iman et al. 2019).

Satu aspek unik dari set data ini ialah ia merangkumi serangan yang tidak diketahui, yang bermaksud bahawa ia dapat digunakan untuk menguji keberkesanan IDS dalam mengesan ancaman yang belum pernah dilihat sebelumnya. Set data ini digunakan untuk menilai prestasi IDS sedia ada dan berpotensi mengembangkan yang baru yang lebih dilengkapi untuk menangani serangan DDoS.

Jadual 2.2 Ringkasan set data digunakan untuk IDS

Set Data	**Tujuan**	**Ciri-ciri Utama**	**Jenis Serangan**	**Bil. data**	**Kelebihan**	**Kekurangan**
KDD'99	Kajian IDS	Aliran trafik rangkaian, lapisan jaringan dan lapisan aplikasi	DoS, pengimejan, *user-to-root* (U2R), *root-to-local* (R2L) dan lain-lain	5 juta rekod	Saiz set data besar dan dilabel dengan pelbagai jenis serangan	Ciri-ciri berlebihan dan tidak relevan serta persekitaran yang disimulasikan mungkin tidak mewakili trafik dunia nyata
CAIDA	Analisis rangkaian dan prestasi	Data aliran rangkaian IP dan TCP/UDP, ciri-ciri pakej transaksi dan metadata	Tiada	Berbeza ikut pada set data	Data trafik internet sebenar dan pandangan ke atas tingkah laku rangkaian sebenar	Data berlabel terhad atau tidak ada untuk tugas keselamatan dan data sukar didapati atau mempunyai sekatan penggunaan
NSL-KDD	Kajian IDS	Aliran trafik rangkaian, kurangkan ciri-ciri berlebihan dari KDD'99	DoS, pengimejan, *user-to-root* (U2R), *root-to-local* (R2L) dan lain-lain	125,973 rekod	Peningkatan versi dan dilabel dengan pelbagai serangan	Simulasi persekitaran
UNSW-NB-15	Menilai IDS	Aliran trafik rangkaian dan ciri-ciri aplikasi	DoS, pengintaian dan lai-lain	2.5 juta rekod	Senario serangan pelbagai dalam makmal terkawal dan set data besar	Kerelevanan dunia nyata terhad

bersambung ...

sambungan ...

CIC-IDS2017	Kajian IDS	Data aliran rangkaian dan ciri-ciri pakej transaksi	DoS, serangan brute force, perisian hasad dan lain-lain	2.8 juta rekod	Set data komprehensif dan dihasilkan dalam makmal terkawal	Ketiadaan keragaman berbanding trafik dunia nyata
CIC-DDOS2019	Teknik pengesanan dan pengekalan DDoS	Aliran trafik rangkaian dan ciri-ciri serangan DDoS	Serangan penafian perkhidmatan teragih (DDoS) seperti UDP *flood*, SYN *flood* dan lain-lain	16 set data	Tumpuan kepada serangan DDoS dan *instance* berlabel untuk pembelajaran berbimbing	Mungkin tidak meliputi jenis ancaman keselamatan lain

2.7 SISTEM PENGESANAN PENCEROBOHAN RANGKAIAN (NIDS) SUMBER TERBUKA

Sistem Pengesanan Pencerobohan Rangkaian (NIDS) merupakan alat keselamatan penting yang digunakan untuk memantau dan melindungi rangkaian komputer daripada aktiviti berbahaya. NIDS berfungsi menganalisis trafik rangkaian dalam masa nyata, mengesan tingkah laku yang mencurigakan dan memberi amaran kepada pentadbir sistem atau rangkaian. Sistem NIDS komersial adalah mahal dan menjadikannya sukar bagi organisasi kecil untuk mengakses dan mengubahsuai mengikut keperluan. Walau bagaimanapun, kemunculan perisian sumber terbuka (OSS) telah menyediakan penyelesaian alternatif untuk keselamatan rangkaian. NIDS sumber terbuka menjadi semakin popular kerana kos, fleksibiliti dan proses pembangunan kolaboratif mereka.

Pada tahun 2019, satu kajian oleh Chen et al. menilai prestasi NIDS sumber terbuka berbanding komersial. Penyelidikan mendapati bahawa NIDS sumber terbuka sama-sama berkesan dalam mengesan dan menghalang pencerobohan rangkaian bahkan melebihi produk komersial. Ini menyoroti potensi NIDS sumber terbuka untuk menyediakan penyelesaian yang sama berkesan tetapi lebih mudah diakses untuk keselamatan rangkaian.

Selain itu, komuniti sumber terbuka aktif untuk meningkatkan keupayaan NIDS melalui pembangunan dan kemas kini yang berterusan. Pada tahun 2020, projek

Suricata, NIDS sumber terbuka, mengeluarkan kemas kini utama dengan ciri-ciri yang ditingkatkan seperti peraturan pengesanan yang lebih baik, pemprosesan *multi-thread*, dan sokongan untuk TLS 1.3. Perkembangan berterusan ini memastikan bahawa NIDS sumber terbuka kekal relevan dan berkesan dalam mengesan teknik pencerobohan yang sentiasa berkembang.

Salah satu kelebihan utama NIDS sumber terbuka ialah keupayaan untuk pengubahsuaian untuk keperluan rangkaian tertentu. Pada tahun 2021, Al-Mafarjeh et al. mencadangkan rangka kerja NIDS hibrid yang menggabungkan ciri-ciri NIDS komersial dan sumber terbuka. Kerangka kerja itu memanfaatkan fleksibiliti NIDS sumber terbuka untuk membolehkan penyesuaian, manakala juga menggabungkan ketahanan NIDS komersial untuk menangani trafik tinggi. Kajian ini menunjukkan bagaimana NIDS sumber terbuka boleh disesuaikan dan disepadukan ke dalam sistem komersial sedia ada, menyediakan pendekatan yang lebih komprehensif dan tersuai untuk keselamatan rangkaian.

Sifat kolaboratif projek sumber terbuka juga membolehkan pelbagai perspektif dan kepakaran untuk menyumbang kepada pembangunan NIDS. Pada tahun 2022, satu kajian oleh Anwar et al. mengkaji keberkesanan penggunaan teknik pembelajaran mesin dalam NIDS. Pasukan penyelidikan menggunakan NIDS sumber terbuka, Snort, dan melatihnya dengan algoritma pembelajaran mesin untuk meningkatkan keupayaan pengesanan. Hasil menunjukkan bahawa penyertaan pembelajaran mesin secara signifikan meningkatkan prestasi NIDS, menyoroti potensi NIDS sumber terbuka untuk menggabungkan teknologi baru dan kekal di hadapan.

NIDS sumber terbuka dijangka terus berkembang dan menyesuaikan diri dengan landskap keselamatan rangkaian yang sentiasa berubah. Pada tahun 2023, satu kajian oleh Akhavan et al. mencadangkan NIDS berasaskan pembelajaran mendalam yang menggunakan gabungan alat dan teknik sumber terbuka. NIDS yang diusulkan menunjukkan hasil yang menjanjikan dalam mengesan dan mencegah pencerobohan rangkaian. Kajian ini menunjukkan usaha berterusan untuk memindahkan sempadan NIDS sumber terbuka dan potensi untuk menjadi penyelesaian terkemuka dalam keselamatan rangkaian.

Kemunculan NIDS sumber terbuka telah menyediakan alternatif yang cekap kos dan fleksibel untuk keselamatan rangkaian. Melalui pembangunan berterusan, penyesuaian, penyertaan teknologi baru dan kerjasama dalam komuniti sumber terbuka, NIDS telah menunjukkan keupayaan yang mengesankan dalam mengesan dan mencegah pencerobohan rangkaian. Antara sumber terbuka NIDS adalah Snort, Suricata, Bro/Zeek dan Security Onion. Dengan penyelidikan dan kemajuan yang berterusan, NIDS sumber terbuka dijangka terus berkembang dan memainkan peranan penting dalam melindungi rangkaian komputer. Perbandingan ciri-ciri, kelebihan dan kekurangan perisian sumber terbuka NIDS seperti Jadual 2.3.

2.7.1 Snort

Snort ialah sistem pengesanan pencerobohan rangkaian sumber terbuka (NIDS) yang digunakan untuk memantau trafik jaringan dan mengesan ancaman keselamatan yang berpotensi. Ia pertama kali dibangunkan pada tahun 1998 oleh Martin Roesch dan telah menjadi alat yang digunakan secara meluas dalam industri keselamatan siber. Apabila teknologi terus maju dan ancaman siber menjadi lebih canggih, keperluan untuk langkah-langkah keselamatan rangkaian yang berkesan telah menjadi semakin penting.

Satu kajian telah dijalankan mengenai keupayaan Snort untuk mengesan dan mencegah serangan DDoS (Distributed Denial of Service). Serangan DDoS telah menjadi masalah utama dengan insiden profil tinggi menyebabkan kerosakan yang signifikan kepada perniagaan dan organisasi. Kajian ini mendapati bahawa Snort mampu mengesan dan mengurangkan serangan DDoS secara berkesan (Muniz et al., 2019).

Manakala satu artikel kajian yang memberi tumpuan kepada peningkatan prestasi Snort melalui penggunaan *multi-threading*. Dengan menggunakan pelbagai *thread*, Snort mampu memproses trafik rangkaian dengan lebih cekap dan berkesan, meningkatkan prestasi keseluruhan dan keupayaan pengesanan (Kiani et al., 2020). Kajian ini menekankan pentingnya peningkatan dan pengemaskinian Snort untuk mengikuti perkembangan ancaman siber.

Satu lagi kajian telah dijalankan mengenai keberkesanan Snort dalam mengesan pelbagai jenis perisian hasad (malware). Ini adalah disebabkan perisian hasad terus menjadi ancaman utama dalam landskap keselamatan siber. Ia adalah penting bagi sistem NIDS untuk dapat mengesan dan mengelakkan program atau perisian berbahaya ini. Kajian ini mendapati bahawa Snort mampu mengesan pelbagai perisian hasad, termasuk ransomware, trojan, dan cacing. (Khan et al., 2021).

Selain daripada penggunaan dalam keselamatan rangkaian, Snort juga telah dipelajari untuk potensi dalam persekitaran Internet of Things (IoT). Pada tahun 2022, sebuah artikel kajian diterbitkan yang mengkaji penggunaan Snort dalam mengamankan peranti dan rangkaian IoT. Kajian ini mendapati bahawa Snort boleh digunakan untuk mengesan dan mencegah serangan pada peranti IoT secara berkesan, menyediakan lapisan keselamatan tambahan untuk peranti yang semakin terhubung ini. (Saha et al., 2022).

Seterusnya, potensi penggunaan pembelajaran mesin di Snort telah dikaji untuk pengesanan ancaman yang lebih baik. Dengan peningkatan kompleksiti ancaman siber, pembelajaran mesin telah muncul sebagai penyelesaian untuk meningkatkan keselamatan rangkaian. Kajian ini mendapati bahawa memasukkan algoritma pembelajaran mesin ke dalam Snort boleh meningkatkan keupayaan pengesanan secara signifikan, menekankan keperluan untuk kajian berterusan dalam bidang ini. (Chauhan et al., 2023).

Selain daripada mengesan serangan, Snort juga memainkan peranan penting dalam menghalang serangan daripada berjaya memasuki rangkaian organisasi. Ini dicapai melalui penggunaan mekanisme tindak balas aktif seperti menghalang atau menghapuskan trafik berbahaya. Walau bagaimanapun, keberkesanan mekanisme ini bergantung kepada ketepatan ancaman yang dikesan. Untuk meningkatkan ketepatan tindak balas aktif, satu kaedah telah dicadangkan untuk korelasi amaran pintar dan keutamaan. Dalam kajian oleh Fajardo et al. (2021), rangka kerja telah dicadangkan yang menggunakan pembelajaran mesin untuk memberi keutamaan kepada amaran Snort berdasarkan kepentingan dan kesan yang berpotensi terhadap rangkaian. Hasilnya

menunjukkan bahawa rangka kerja yang dicadangkan boleh mengurangkan positif palsu dan mengoptimumkan penggunaan mekanisme tindak balas aktif.

Penggunaan teknologi awan dan persekitaran maya semakin berkembang. Sehubungan itu, terdapat satu kajian yang memberi tumpuan kepada meningkatkan keluasan dan penyesuaian Snort kepada persekitaran ini. Sistem pengurusan peraturan dinamik telah dicadangkan oleh Ahumada et al. (2020) untuk Snort dalam persekitaran maya. Sistem ini mampu menyesuaikan diri dengan perubahan dalam persekitaran maya dan secara berkesan menguruskan peraturan Snort untuk mengoptimumkan prestasi dan mengurangkan amaran palsu.

2.7.2 Suricata

Suricata ialah sistem pengesanan dan pencegahan pencerobohan rangkaian sumber terbuka yang telah mendapat populariti kerana prestasi yang tinggi dan seni bina *multi-thread*. Pembangunan Suricata bermula pada tahun 2008 dan sejak itu telah terus diperbaiki dan dikemas kini untuk mengikuti landskap ancaman yang sentiasa berkembang. Pada tahun 2019, satu kajian oleh Paul dan Raj (2019) menilai keberkesanan Suricata dalam mengesan pelbagai jenis perisian hasad yang berbeza dan mendapati bahawa ia melebihi sistem pengesanan pencerobohan lain.

Pada tahun 2020, satu kajian oleh Kandpal et al. (2020) memberi tumpuan kepada mengoptimumkan prestasi Suricata dengan melaksanakan teknik pemprosesan paket baharu. Hasil menunjukkan peningkatan yang signifikan dalam kadar pengesanan dan penurunan positif palsu, menunjukkan potensi untuk kemajuan lebih lanjut dalam sistem. Selain itu, pada tahun yang sama, penyelidikan oleh Nain dan Verma (2020) mencadangkan pendekatan hibrid yang menggabungkan Suricata dengan teknik pembelajaran mesin untuk pengesanan lebih tepat ancaman lanjutan yang berterusan.

Tahun 2021 menyaksikan perkembangan yang signifikan dalam komuniti Suricata dengan Suricata versi 6.0. Kemas kini ini termasuk ciri-ciri baru seperti sokongan TLS 1.3, penyahkodan HTTP / 2, dan keupayaan pengesanan yang lebih baik untuk penambangan DNS dan serangan berasaskan PowerShell. Pada tahun yang sama, kajian oleh Li et al. (2021) memberi tumpuan kepada meningkatkan keluasan dan

penggunaan sumber Suricata dengan melaksanakan seni bina yang teragih. Hasil menunjukkan peningkatan ketara dalam prestasi sistem, menjadikannya lebih sesuai untuk persekitaran rangkaian berskala besar.

Bergerak ke 2022, satu kajian oleh Gupta et al. (2022) mencadangkan sistem tindak balas insiden baharu menggunakan Suricata dan kecerdasan ancaman untuk mengautomasikan tindak balas kepada insiden keselamatan. Pendekatan ini bukan sahaja mengurangkan masa tindak balas tetapi juga meminimumkan kesan serangan yang berpotensi. Di samping itu, kajian oleh Lee et al. (2022) menilai penggunaan Suricata untuk persekitaran berasaskan awan dan mendapati bahawa ia boleh mengesan dan mencegah serangan dalam persekitaran tersebut dengan berkesan, menjadikannya pilihan yang sesuai untuk organisasi yang beralih ke awan.

Seterusnya, satu kajian oleh Sharma et al. (2023) mencadangkan pendekatan berasaskan ramalan menggunakan Suricata untuk meramalkan serangan yang berpotensi dan mengambil langkah-langkah proaktif untuk menghalang serangan. Teknik ini membuktikan bahawa ia boleh mengubah permainan dalam bidang pengesanan pencerobohan, membolehkan organisasi satu langkah di hadapan daripada ancaman siber. Selain itu, kajian oleh Mishra et al. (2023) memberi tumpuan kepada menggabungkan teknologi blok rantai ke dalam Suricata untuk menyediakan sistem pengurusan log yang lebih selamat. Integrasi ini boleh meningkatkan kebolehpercayaan sistem, menjadikannya pilihan yang lebih kukuh untuk organisasi.

Salah satu komponen utama Suricata ialah enjin peraturan, yang bertanggungjawab untuk menganalisis trafik rangkaian dan mengesan ancaman keselamatan yang berpotensi. Enjin peraturan sentiasa dikemas kini dengan peraturan baru untuk mengikuti landskap ancaman yang sentiasa berubah. Kaedah automatik untuk menghasilkan peraturan Suricata dengan menganalisis corak serangan yang diketahui dan mewujudkan peraturan berdasarkan corak tersebut (Chen et al., 2019). Pendekatan ini secara signifikan mengurangkan masa dan usaha yang diperlukan untuk mewujudkan peraturan, menjadikan Suricata lebih cekap dalam mengesan ancaman baru dan muncul.

Komponen Suricata lain ialah seni bina *multi-threaded*, yang membolehkan ia memproses trafik rangkaian secara bersamaan pada beberapa CPU. Seni bina ini penting untuk mengendalikan jumlah trafik rangkaian yang tinggi dan memastikan pengesanan ancaman yang cekap dan cepat. Seni bina *multi-threaded* Suricata secara signifikan meningkatkan prestasi dalam mengesan serangan rangkaian berbanding dengan sistem IDPS lain (Bianchi et al., 2020).

Selain daripada enjin peraturan dan seni bina *multi-threaded*, Suricata juga menggunakan pelbagai teknik pengesanan untuk mengenal pasti ancaman yang berpotensi. Ini termasuk pengesan berasaskan tanda tangan, pengesanan berasaskan anomali dan analisis protokol. Penilaian komprehensif keupayaan pengesanan Suricata dan mendapati bahawa ia sangat berkesan dalam mengesan pelbagai serangan rangkaian. (Peng et al., 2021).

Untuk meningkatkan keupayaan pengesanan, Suricata juga menyokong penggunaan pembelajaran mesin dan algoritma kecerdasan buatan (AI). Teknik ini boleh digunakan untuk menganalisis trafik rangkaian dan mengenal pasti corak yang mungkin menunjukkan aktiviti berbahaya. Dalam satu kajian didapati bahawa menggabungkan Suricata dengan algoritma pembelajaran mesin meningkatkan keupayaan Suricata untuk mengesan dan mencegah serangan yang tidak diketahui. (Hosseini et al., 2022).

2.7.3 Zeek

Zeek ialah platform keselamatan rangkaian sumber terbuka dan prestasi tinggi yang menggunakan bahasa skrip untuk menganalisis trafik rangkaian dan mengesan ancaman keselamatan yang berpotensi. Zeek berkesan dalam mengesan dan mengurangkan pelbagai jenis serangan rangkaian.

Zeek mampu mengesan dan mengurangkan serangan DDoS dengan tepat, menjadikannya penyelesaian yang layak untuk perlindungan terhadap jenis ancaman DDoS (Jain, 2019). Demikian juga, keberkesanan Zeek dalam mengesan dan mencegah pelbagai serangan perisian hasad (Agrawal, 2020).

Zeek berkeupayaan untuk mengintegrasikan dengan alat dan sistem keselamatan lain. Terdapat kajian memberi tumpuan kepada integrasi Zeek dengan Snort. Hasil menunjukkan bahawa gabungan Zeek dan Snort menyediakan pendekatan yang lebih komprehensif dan berkesan untuk keselamatan rangkaian kerana ia melengkapi kekuatan satu sama lain. (Dey, 2021). Demikian juga, kajian integrasi Zeek dengan teknik pembelajaran mesin untuk meningkatkan pengesanan anomali. Gabungan analisis trafik rangkaian Zeek dan algoritma pembelajaran mesin membawa kepada pengesanan yang lebih tepat dan efisien dalam pencerobohan rangkaian (Singh, 2022).

Selain itu, terdapat juga kajian memberi tumpuan kepada meningkatkan prestasi Zeek dengan mengoptimumkan bahasa skripnya dimana algoritma baru yang meningkatkan kecekapan enjin peristiwa Zeek, yang membawa kepada pengesanan yang lebih cepat dan tepat anomali rangkaian. (Gupta, 2023).

Satu komponen penting ialah rangka kerja skrip Zeek, yang membolehkan penyesuaian dan keluasan sistem. Menurut kajian Kaur et al.(2019), penggunaan skrip Zeek telah terbukti meningkatkan keupayaan pengesanan sistem serta ketepatan dan kecekapan proses analisis.

Komponen utama lain daripada sistem pengesanan pencerobohan rangkaian Zeek ialah penggunaan analisis trafik jaringan. Ini melibatkan pemantauan dan analisis trafik rangkaian untuk mengenal pasti sebarang aktiviti yang mencurigakan atau berbahaya. Sharma et al. (2020) menekankan pentingnya komponen ini, kerana ia membolehkan pengesanan kedua-dua ancaman yang diketahui dan tidak diketahui di rangkaian. Selain itu, dengan kemajuan dalam teknologi dan peningkatan kerumitan serangan siber, penggunaan analisis trafik rangkaian telah menjadi penting dalam memastikan keselamatan rangkaian.

Sistem Zeek juga menggunakan algoritma pembelajaran mesin sebagai komponen untuk mengesan dan menganalisis ancaman. Algoritma ini dilatih pada set data besar ancaman yang diketahui dan kemudian boleh mengesan corak yang serupa dalam trafik rangkaian masa nyata. Satu kajian Kumar et al. (2021) mendapati bahawa

penggunaan algoritma pembelajaran mesin di Zeek secara signifikan meningkatkan ketepatan dan kelajuan pengesanan ancaman. Selain itu, dengan evolusi terus-menerus ancaman siber, penggunaan pembelajaran mesin di Zeek membolehkan sistem untuk disesuaikan dengan ancaman baru.

2.7.4 Security Onion

Security Onion ialah sistem pemantauan keselamatan rangkaian dan pengesanan pencerobohan sumber terbuka. Security Onion telah dikembangkan oleh Doug Burks pada tahun 2008, Security Onion telah menjadi alat penting bagi organisasi yang ingin meningkatkan sikap keselamatan siber mereka dan melindungi rangkaian mereka daripada pelbagai ancaman. Kerangka kerja ini dibina di atas Ubuntu dan diintegrasikan dengan pelbagai alat keselamatan seperti Snort, Suricata, Zeek dan Squert untuk menyediakan penyelesaian keselamatan rangkaian yang komprehensif. Security Onion telah diperbaharui dan dikemas kini dengan ciri-ciri dan keupayaan baru yang ditambah dengan setiap versi baru.

Satu kajian oleh Ruifang et al. (2019) mencadangkan sistem pengesanan pencerobohan yang lebih baik untuk Security Onion dengan menggunakan teknik pembelajaran mesin untuk mengesan anomali dan aktiviti jahat dalam rangkaian. Pendekatan ini membawa kepada sistem pengesanan yang lebih tepat dan berkesan, yang mampu mengenal pasti serangan *zero day* dan ancaman terkini. Selain itu, Security Onion dikaji dengan persekitaran awan kerana kebanyakan organisasi ke arah infrastruktur berasaskan awan. Satu kajian oleh Liu et al. (2019) memperkenalkan seni bina baru Security Onion yang direka khas untuk persekitaran awan yang lebih baik dan keselamatan rangkaian berasaskan awan.

Selain itu, Bao et al. (2020) telah memperkenalkan seni bina ringan untuk Security Onion yang secara signifikan mengurangkan penggunaan memori dan CPU sambil mengekalkan keberkesanan dalam mengesan aktiviti berbahaya. Liu et al. (2020) juga telah memperkenalkan seni bina teragih untuk Security Onion yang membolehkan penyebaran kerangka dalam persekitaran yang diagihkan untuk meningkatkan keluasan dan prestasi Security Onion.

Chen et al. (2022) mencadangkan sistem pengesanan pencerobohan berasaskan pembelajaran mendalam untuk Security Onion mencapai kadar pengesan yang lebih tinggi dan kadar positif palsu yang lebih rendah berbanding kaedah biasa. Selain itu, siasatan penggunaan Security Onion dalam persekitaran Internet of Things (IoT) terus meningkat. Satu kajian oleh Zhang et al. (2023) memperkenalkan rangka kerja menggunakan Security Onion untuk memantau dan melindungi rangkaian IoT dan menyediakan penyelesaian yang komprehensif dan berpusat.

Jadual 2.3 Ringkasan Perisian Sumber Terbuka NIDS

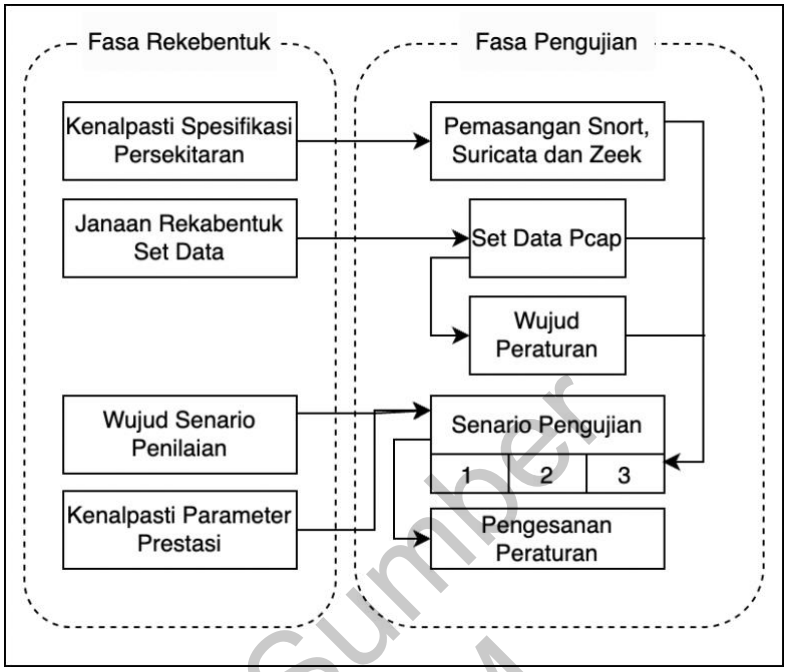
Kategori	Snort	Suricata	Zeek	Security Onion
Komponen	Pengecam paket, enjin pengesanan, penterjemah peraturan	IDS Rangkaian, enjin IPS, pengesanan ancaman baru	Rangkaian analisis, sokongan skrip	Pengedaran Linux dengan pelbagai alat keselamatan (Snort, Suricata, Zeek, Squert)
Ciri-ciri	Pengesanan berdasarkan tandatangan, analisis protokol	Multi-threaded, pengekstrakan fail, sokongan ancaman baru	Analisis protokol, sokongan skrip	Integrasi pelbagai alat keselamatan, pengurusan terpusat
Kelebihan	Matang, set peraturan meluas, ringan	Prestasi tinggi, sokongan protokol	Kelenturan, pengetahuan menyeluruh	Penyelesaian serba satu, kemudahan pengedaran
Kekurangan	Sokongan protokol terhad, skalabiliti terhad	Kurva pembelajaran, penggunaan sumber tinggi	Kurva pembelajaran yang curam, keperluan sumber	Keperluan sumber, kompleksiti

2.8 MOTIVASI KAJIAN

Motivasi kajian berdasarkan kajian kesusasteraan yang telah di laksana untuk menilai prestasi perisian sumber terbuka IDS dari segi ketepatan pengesanan, kelajuan dan penggunaan sumber. Ini termasuk menilai keupayaan mengenal pasti dan mengesan jenis serangan. Selain itu, perbandingan pilihan perisian IDS yang sesuai untuk memenuhi keperluan keselamatan rangkaian.

Sehubungan itu, kajian akan di laksana mengandungi dua fasa iaitu reka bentuk dan pengujian seperti Rajah 2.3. Fasa reka bentuk termasuk penilaian persekitaran pengujian, janaan reka bentuk set data, reka bentuk senario penilaian dan parameter prestasi. Manakala fasa pengujian melibatkan *outcome* fasa reka bentuk akan

dilaksanakan untuk menghasilkan janaan set data, peraturan daripada set data pcap dan peratusan pengesanan bagi setiap senario.



Rajah 2.3 Peringkat Kajian

2.9 ISU DAN JURANG KAJIAN

Hasil daripada kajian kesusasteraan yang telah dilaksanakan terdapat beberapa isu dan jurang kajian mengenai sistem pengesanan pencerobohan sumber terbuka. Antaranya adalah seperti berikut:

1. Cabaran dari segi prestasi dan keberkesanan IDS sumber terbuka. Serangan intensif rangkaian boleh membebankan sumber pemantauan dan analisis serta mempengaruhi prestasi IDS.

2. Rangkaian berkelajuan tinggi dengan jumlah lalu lintas yang besar dan pelbagai menimbulkan kesulitan teknikal bagi IDS seterusnya mengakibatkan penurunan paket dan penurunan ketepatan pengesanan.

3. Kadar pengesanan masa nyata dan kadar positif palsu adalah faktor penting untuk sistem pengesanan pencerobohan terutama dalam persekitaran produksi.

4. Keupayaan IDS untuk mengklasifikasikan serangan yang diketahui dan mengenali serangan yang tidak diketahui sangat penting untuk meningkatkan keberkesanan dan kecekapannya

Secara keseluruhan, sistem pengesanan pencerobohan sumber terbuka perlu menangani cabaran ini untuk meningkatkan prestasi dan keberkesanannya dalam mengesan dan mencegah serangan rangkaian.

2.10 KESIMPULAN

Kajian kesusasteraan yang dibuat adalah mengambil kira proses pengesanan pencerobohan dan menilai serta menguji IDS melibatkan perisian sumber terbuka yang telah dikenalpasti iaitu Snort, Suricata dan Zeek. Selain itu, ciri-ciri pengesanan yang melibatkan penetapan set peraturan bagi memastikan keberkesanan perisian tersebut termasuk pengurusan sumber melibatkan penggunaan CPU dan memori.

Serangan yang berkaitan dengan siber meningkat secara eksponen dan tidak ada kaedah untuk menghentikan serangan. IDS sangat penting dalam mengurangkan atau mencegah serangan siber, tetapi penyerang menggunakan perisian canggih untuk mengelakkannya. Untuk meningkatkan pengesanan serangan siber baru dan kompleks, IDS semasa perlu disepadukan dengan teknologi seperti awan, pembelajaran mesin, dan pembelajaran mendalam. IDS berasaskan rangkaian berkesan dalam mengesan pencerobohan pada rangkaian komputer, sementara IDS berasaskan hos mengesan serangan ke atas hos individu.

Teknik pengesanan berasaskan tandatangan adalah cepat dan berkesan untuk serangan yang diketahui, sementara IDS berasaskan anomali dapat mengesan serangan yang tidak diketahui tetapi mungkin menghasilkan penggera palsu.

Penggunaan perisian sumber terbuka NIDS adalah salah satu pilihan dalam mengesan dan menghalang serangan siber. Sehubungan itu, pemilihan penggunaan IDS adalah amat penting bagi mengurangkan kerosakan kepada sesuatu organisasi.

BAB III

METODOLOGI

3.1 PENGENALAN

Bab III membincangkan metodologi yang diguna pakai bagi melaksanakan kajian ini. Sasaran kajian ini adalah NIDS sumber terbuka yang sedia ada sebagai perisian yang berasingan dan sebagai sebahagian daripada produk komersial. Perisian telah ditetapkan untuk memberikan maklumat dan keputusan tentang penggunaannya dalam tetapan rangkaian. Snort, Suricata dan Zeek adalah NIDS sumber terbuka generik yang tersedia dalam pasaran sumber terbuka pada masa ini.

3.2 RANGKA KERJA KAJIAN

Kajian ini akan menggunakan pendekatan kaedah campuran dimana menggabungkan kedua-dua kaedah penyelidikan kuantitatif dan kualitatif. Rangka kerja kajian akan terdiri daripada langkah-langkah berikut:

1. Kajian Literatur: Kajian komprehensif mengenai kesusasteraan sedia ada mengenai perisian IDS sumber terbuka, keupayaan dan keberkesanannya dilaksana.

2. Pemilihan Perisian IDS: Tiga perisian IDS sumber terbuka, Snort, Suricata, dan Zeek, akan dipilih untuk penilaian berdasarkan popularity, keupayaan dan ketersediaan mereka.

3. Pengumpulan data: Penyelidikan akan menggunakan sumber data primer dan sekunder. Data utama akan dikumpulkan melalui eksperimen, sementara data sekunder akan dikumpulkan dari kajian sedia ada, sumber dalam talian, dan dokumentasi perisian IDS yang dipilih.

4. Reka Bentuk Eksperimen: Penyelidikan akan menjalankan eksperimen untuk menilai prestasi dan keberkesanan setiap perisian IDS. Eksperimen ini akan melibatkan simulasi pelbagai jenis serangan siber dan memantau pengesanan dan tindak balas setiap perisian.

5. Analisis Data: Data yang dikumpul akan dianalisis menggunakan teknik statistik untuk menentukan prestasi setiap perisian IDS dari segi kadar pengesanan, positif palsu dan negatif palsu.

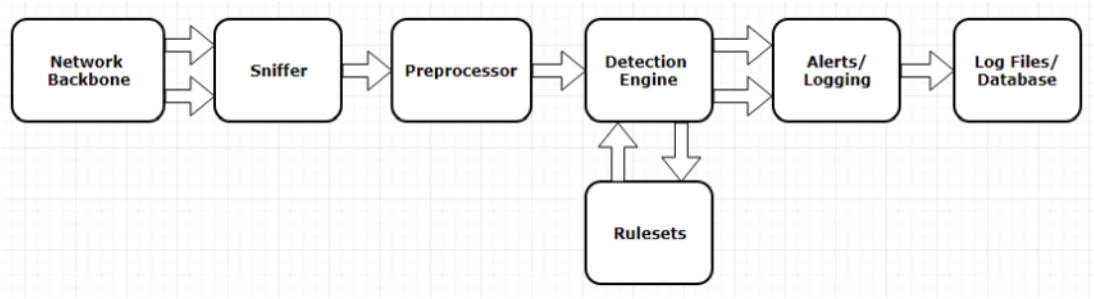
6. Analisis Kualitatif: Data kualitatif yang dikumpul daripada eksperimen akan dianalisis untuk mengenal pasti kekuatan dan kelemahan setiap perisian IDS.

7. Perbandingan dan Cadangan: Keputusan analisis akan dibandingkan dengan mengenal pasti kekuatan dan kelemahan setiap perisian IDS. Berdasarkan penemuan, cadangan untuk meningkatkan prestasi dan keberkesanan setiap perisian akan dicadangkan.

3.3 PERISIAN SUMBER TERBUKA NIDS

3.3.1 Snort

Snort adalah sistem pengesanan dan pencegahan pencerobohan sumber terbuka yang sering digunakan untuk mengesan dan mencegah serangan rangkaian. Ia dicipta oleh Martin Roesch dan dikekalkan oleh Cisco. Snort menggunakan bahasa berdasarkan peraturan untuk menganalisis trafik rangkaian dan boleh dikonfigurasikan untuk melakukan pelbagai tindakan berdasarkan tanda atau corak yang dikesan. Snort adalah perisian pengesanan berasaskan tandatangan yang mempunyai set peraturan yang ditetapkan Gambaran umum tentang seni bina Snort seperti Rajah 3.1. Komponen utama termasuk pengekod pakej, pra-pemproses dan enjin pengesanan mengaplikasikan pengesanan berasaskan tanda tangan untuk mengenal pasti ancaman keselamatan. Set peraturan membandingkan atribut pakej dengan peraturan pengguna untuk memutuskan sama ada terdapat padanan, sementara pencatatan dan pemberitahuan menyediakan rekod peristiwa dan memberitahu pentadbir. Modul output dan sokongan pangkalan data membolehkan penyesuaian fleksibel dan analisis sejarah, manakala peraturan komuniti memberikan manfaat daripada pengetahuan kolektif untuk meningkatkan keberkesanan Snort.



Rajah 3.1 Seni bina Snort

Sumber: Shah, Syed et al(2018)

Setiap komponen Snort memainkan peranan penting dalam mengesan serangan tetapi kebergantungan kepada set peraturan yang ditetapkan. Walau bagaimanapun, set peraturan perlu dikemaskini bagi memastikan keberkesanan Snort. Contoh set peraturan adalah seperti Rajah 3.2.

```
alert tcp $HOME_NET 20034 -> $EXTERNAL_NET any ( msg:"MALWARE-BACKDOOR NetBus Pro 2.0 connection established";
flow:to_client,established; flowbits:isset,backdoor.netbus_2.connect; content:"BN|10 00 02 00|",depth 6; content:"|
05 00|",depth 2,offset 8; metadata:ruleset community; classtype:trojan-activity; sid:115; rev:15; )
alert tcp $HOME_NET any -> $EXTERNAL_NET any ( msg:"MALWARE-BACKDOOR Infector.1.x"; flow:to_client,established;
content:"WHATISIT",depth 9; metadata:impact_flag red,ruleset community; reference:nessus,11157; classtype:misc-
activity; sid:117; rev:17; )
alert tcp $HOME_NET 666 -> $EXTERNAL_NET any ( msg:"MALWARE-BACKDOOR SatansBackdoor.2.0.Beta";
flow:to_client,established; content:"Remote|3A| ",depth 11,nocase; content:"You are connected to me.|0D 0A|Remote|
3A| Ready for commands",distance 0,nocase; metadata:ruleset community; reference:url,www.megasecurity.org/trojans/s/
satanzbackdoor/SBD2.0b.html; reference:url,www3.ca.com/securityadvisor/pest/pest.aspx?id=5260; classtype:trojan-
activity; sid:118; rev:12; )
```

Rajah 3.2 Contoh Set Peraturan Snort

Sumber: Snort.org

Pelaksanaan simulasi/eksperimen ini menggunakan fail pcap daripada set data CIC-IDS2017. Ketika memproses fail pcap, Snort akan menganalisis trafik rangakian untuk mengesan dan memberikan amaran jika terdapat aktiviti berbahaya. Snort memproses fail pcap adalah seperti berikut:

1. Konfigurasi

    a. Snort menggunakan fail konfigurasi iaitu snort.conf yang mengandungi peraturan, prepemprosesan, pilihan *logging* dan pelbagai tetapan.

    b. Konfigurasi boleh diubahsuai mengikut keperluan keselamatan.

2. Prepemprosesan

a. Prepemprosesan untuk menormalkan dan pre proses trafik rangkaian sebelum di analisis. Prepemprosessn melaksana penormalan paket dan pengurusan *fragmentation.*

3. Padanan peraturan

a. Fungsi utama Snort berasaskan padanan peraturan. Peraturan di tulis dengan bahasa yang spesifik dan syarat untuk memastikan pencetus amaran atau maklum balas sekiranya terdapat aktiviti berbahaya.

4. Penjanaan Amaran

a. Peraturan dicetus, Snort akan jana amaran. Amaran mengnadungi maklumat tentang spesifik peraturam yang telah dipadankan, maklumat terperinci paket dan maklumat lain yang berkaitan.

b. Amaran akan disimpan ke dalam fail dan dihantar ke server syslog dan terdapat juga trafik disekat.

5. Logging dan output

a. Snort menyediakan pelbagai pilihan untuk logging dan output termasuk log amaran ke fail, output ke console atau penghantaran samaran ke sistem log berpusat.

3.3.2 Suricata

Suricata adalah salah satu lagi perisian untuk mengesan dan mencegah serangan daripada pencerobohan mahupun ancaman. Suricata menggunakan enjin *multithread* untuk memproses trafik rangkaian berdasarkan teknik pengesanan berasaskan tnadatangan. Suricata menyokong peraturan dan bahasa tandatangan disamping mempunyai keupayaan menggunakan skrip Lua untuk mengesan ancaman yang kompleks. Gambaran umum tentang senibina Suricata seperti Rajah 3.3. Komponen utama melibatkan *packet acquisition, decode and stream, detect* dan *output. Detect engine* akan berkomunikasi dengan set peraturan bagi mengesan aktiviti normal atau ancaman seperti Snort. Contoh set peraturan seperti Rajah 2.2. Antara ciri-ciri Suricata adalah seperti berikut:

1. Analisis fail pcap secara offline.

2. Rekod trafik menggunakan pcap logger.

3. Fail konfigurasi YAML yang mudah dibaca dengan format seperti XML.

4. Menyokong sepenuhnya IPv6.

5. Enjin TCP yang mampu mengesan dan menyusun semula sesi.

6. Menyokong dekod IPv4, IPv6, TCP, UDP, SCTP, ICMPv4, ICMPv6, GRE, Ethernet, PPP dan paket VLAN.

7. Menyokong dekod protokol lapissan aplikasi HTTP, SSL, TLS, SMTP, FTP, SSH, DNS dan DHCP.

8. Enjin FTP, HTTP dan SMTP mampu melakukan perekodan transaksi dan pengenalpastian, pengeluaran dan perekodan fail.

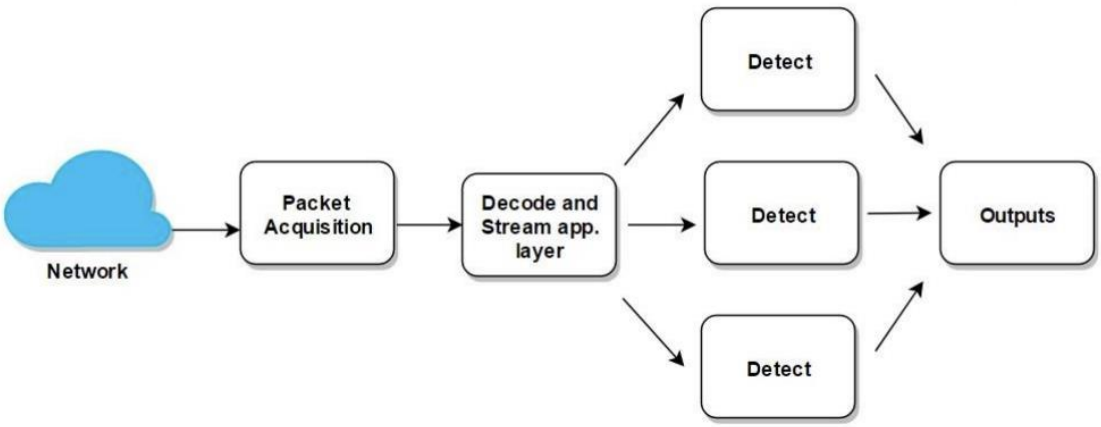
9. Peraturan real time kemaskini tanpa perlu memulakan semula service Suricata.

10. Logging EVE dan JSON berupaya menjana format personal output dangan skrip Lua.

11. Menapis aktiviti berdasarkan peraturan atau *thresholds.*

12. Menyokong *multi-threading*.

13. Mekanisma reputasi IP integrasi dengan peraturan



Rajah 3.3 Senibina Suricata

Sumber: Shah, Syed e tal(2018)

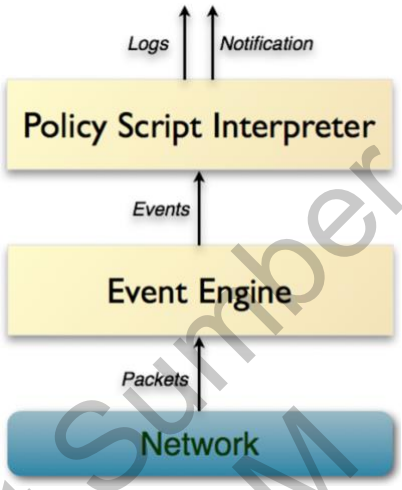
3.3.3 Zeek

Zeek adalah pengesan trafik rangkaian sumber terbuka dan pasif serta digunakan sebagai pemantau keselamatan rangkaian (NSM) untuk menyokong penyiasatan terhadap aktiviti yang mencurigakan atau berbahaya. Zeek juga menyokong pelbagai tugas analisis trafik di luar domain keselamatan termasuk pengukuran prestasi dan penyelesaian masalah. Zeek adalah perisian pengesanan berasaskan anomali dimana laksana pengesanan berasaskan persekitaran skrip dengan bahasa pengaturcaraan tersendiri. Bahasa skrip Zeek adalah *event-driven.* Sebagai contoh, aktiviti adalah ekstrak daripada trafik rangkaian. Ia membenarkan untuk mewujudkan fungsi untuk spesifik aktiviti untuk laksana analisis.

Zeek mempunyai dua komponen utama iaitu *event engine* dan *script interpreter*. *Event engine* digunakan untuk menukar aliran input kepada siri *event* peringkat tinggi yang menggambarkan aktiviti rangkaian. Zeek juga menggunakan bahasa scripting untuk mentafsirkan *event* dan melaksanakan dasar keselamatan. Enjin ini melibatkan analisis paket, analisis sesi dan analisis fail untuk memproses data rangkaian pada pelbagai peringkat. Zeek juga boleh mencipta amaran masa nyata dan log maklumat terperinci untuk analisis selepas insiden. Log Zeek seperti *conn.log, http.log* dan *dhcp.log* mempunyai nilai yang penting kerana mereka menyediakan maklumat terperinci tentang hubungan yang dibangunkan dan spesifikasi rangkaian Kelebihannya terletak dalam reka bentuk modular dan keupayaan untuk memperluaskan seni bina

plugin seterusnya menjadikannya perisian serba guna untuk pemantauan rangkaian dan keselamatan. Seni bina Zeek seperti Rajah 3.4.

Berbeza daripada Snort, Zeek hanya laksana IDS sahaja dan bukannya IPS. Zeek akan memantau trafik dan menghasilkan fail log dengan amaran seterusnya disemak secara manual atau melalui sistem.



Rajah 3.4 Senibina Zeek

Sumber: Zeek

Zeek melalui beberapa langkah untuk mengekstrak maklumat dan menghasilkan log rangkaian yang terperinci hasil daripada memproses fail pcap. Zeek memproses fail pcap adalah seperti berikut:

1. Konfigurasi

    a. Zeek menggunakan fail konfigurasi iaitu local.zeek untuk tetapan, polisi dan pilihan. Konfigurasi boleh diubahsuai mengikut keperluan.

2. Paket Dekod

    a. Zeek menganalisis setiap paket dalam fail pcap untuk mengekstrak maklumat yang relevan. Ia mendekod pelbagai protokol rangkaian termasuk protokol TCP, UDP, HTTP, DNS dan lain-lain.

3. Jejak Sambungan

a. Zeek mengekalkan penjejakan sambungan yang bersifat mengekalkan keadaan, membolehkan ia merekonstruk dan menganalisis protokol.

4. Logging

a. Zeek menghasilkan pelbagai log yang menyediakan ringkasan mengenai aktiviti rangkaian atau butiran penting seperti IP sumber dan destinasi, nombor port. Log biasa termasuk conn.log, dns.log, http.log.

5. Pelaksanaan Skrip

a. Penulisan tambahan skrip boleh dilaksana dan disesuaikan mengikut tingkah laku Zeek berdasarkan keperluan keselamatan.

6. Pengekstrakan Fail

a. Zeek boleh mengekstrak dan menyimpan fail yang dihantar melalui rangkaian. Fail yang diekstrak boleh dianalisis atau digunakan untuk tujuan forensik.

7. Kerangka Notis

a. Notis bersarkan dasar atau peraturan yang diubahsuai. Ini membantu mengenalpasti aktiviti yang mencurigakan.

8. Output

a. Zeek menghasilkan output dalam pelbagai format termasuk teks biasa, JSON, CSV dan lain-lain.

3.4 PERSEKITARAN PENGUJIAN

Persekitaran pengujian diwujudkan dengan persekitaran maya menggunakan perisian virtual iatu VirtualBox. Perisian VirtualBox membolehkan menjalankan semua hos dalam ujian pada mesin yang sama dan mengurangkan kerumitan penyediaan ujian. VirtualBox ialah projek sumber terbuka dan percuma digunakan untuk peribadi, pendidikan dan kajian. Eksperimen akan dijalankan dalam persekitaran maya ini untuk memastikan mudah alih dan keselamatan yang membolehkan permulaan eksperimen

lebih cepat. Reka bentuk eksperimen merangkumi metrik empirikal, tekanan sistem, penjanaan trafik dengan menggunakan set data.

Tiga virtual machine (VM) telah diwujudkan masing-masing dipasang dengan perisian IDS iaitu Snort, Suricata dan Zeek. Setiap VM dilengkapi dengan sistem pengoperasian Ubuntu 20.04, storan 40GB, 4GB memori dan pemproses 1 CPU. Versi setiap perisian IDS tersebut adalah Snort 3.1.6.0, Suricata 7.0.2 dan Zeek 6.1.0. Spesifikasi perkakasan (dalam persekitaran maya) menggunakan perisian VirtualBox versi 6.1.12 seperti Jadual 3.1. Setiap perisian IDS tersebut telah dipasang dan di konfigurasi pada VM yang telah disediakan bagi memastikan setiap perisian tersebut berfungsi mengikut keperluan. Selain daripada perisian utama, perisian sokongan diperlukan untuk tujuan pemantauan prestasi iaitu load. Langkah-langkah pemasangan dan konfigurasi bagi setiap IDS seperti Lampiran A.

Jadual 3.1 Spesifikasi perkakasan (persekitaran maya)

Spesifikasi	Snort	Suricata	Zeek
CPU	1 core (Intel i5@2.9Ghz)	1 core (Intel i5@2.9Ghz)	1 core (Intel i5@2.9Ghz)
Sistem Pengoperasian	Ubuntu 20.04	Ubuntu 20.04	Ubuntu 20.04
Storan	40 GB	40 GB	40 GB
RAM	4GB	4GB	4GB
Versi Perisian NIDS	3.1.78	6.0.5	6.1.0

3.4.1 Set data

Kajian ini akan menggunakan fail pcap sebagai set data. Penggunaan fail pcap daripada set data CIC-IDS2017 dan CICDDoS2019 yang mengandungi parameter IP sumber, IP destinasi, port, *timestamp*, protokol dan transaksi atau trafik yang mengandungi benign atau normal dan serangan.

Set Data yang digunakan dalam kajian ini adalah CIC-IDS2017 telah dibangunkan oleh Canadian Institute for Cybersecurity. Ia mengandungi set data trafik rangkaian yang komprehensif dan digunakan oleh penyelidik keselamatan siber seluruh dunia untuk penilaian IDS dan ML-analysis. Set data CIC-IDS2017 mengandungi tiga jenis fail iaitu *raw* data rangkaian, Generated Labelled Flows dan Machine Learning

csv dan mempunyai 80 *features* berbeza serta 3,111,388 trafik rangkaian. Aliran trafik yang ditangkap ini terdiri daripada trafik normal dan trafik berbahaya. Set data ini terdiri daripada pelbagai serangan antaranya DDOS, Infilteration, Web Attacks, PortScan. Senarai set data ini menggunakan set data yang telah dilabelkan seperti Jadual 3.2.

Jadual 3.2 Set Data CIC-IDS2017

Bil	Nama Fail	Saiz
1	Monday-WorkingHours.pcap_ISCX.csv	8.8MB
2	Tuesday-WorkingHours.pcap_ISCX.csv	6.8MB
3	Wednesday-WorkingHours.pcap_ISCX.csv	11.3MB
4	Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX.csv	2.6MB
5	Thursday-WorkingHours-Afternoon-Infilteration.pcap_ISCX.csv	4.2MB
6	Friday-WorkingHours-Morning.pcap_ISCX.csv	2.9MB
7	Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX.csv	3.8MB
8	Friday-WorkingHours-Afternoon-DDos.pcap_ISCX.csv	3.9MB

Seterusnya set data yang digunakan dalam kajian ini adalah CICDDoS2019. Set data ini juga telah dibangunkan oleh Canadian Institute for Cybersecurity. Ia mengandungi rekod terperinci serangan DDoS yang berlaku pada tahun 2019. Set data ini digunakan untuk memahami ciri-ciri dan corak serangan dan membangunkan mekanisme pertahanan yang berkesan. Set data termasuk maklumat mengenai jenis serangan, tempoh, alamat IP sumber dan destinasi serta beban data yang digunakan dalam serangan. Ia mengandungi 50,063,112 rekod trafik termasuk 50,006,249 baris serangan DDoS dan 56,863 *benign* dalam format pcap. Setiap baris mempunyai 88 ciri.

3.4.2 Penerangan Set Data Yang Digunakan

Eksperimen ini menggunakan dua set data daripada repositori Canadian Institute of CyberSecurity, CIC-IDS2017. Set data ini menggunakan kaedah baru seperti sistem B-Profile, untuk menghasilkan trafik rangkaian yang realistik berdasarkan tingkah laku manusia yang abstrak. Set data CIC-IDS2017 mengandungi data trafik rangkaian dari 3 hingga 7 Julai 2017 (5 hari). Ia mengandungi pelbagai jenis serangan seperti Brute Force-FTP, Brute Force-SSH, DoS, Botnet, DDoS dan Serangan Web. Set data yang

digunakan adalah daripada kategori Machine Learning CVE dimana datanya telah diproses dan dilabel menggunakan CICFlowMeter. Jadual 3.3 menunjukkan keterangan fail dan aliran set data CIC-IDS2017. Jadual 3.4 pula menunjukkan bilangan aliran merangkumi aktiviti rangkaian termasuk aktiviti normal dan berbahaya. Pembahagian set data tersebut menggunakan kaedah ScikitLearn.

Jadual 3.3 Keterangan fail dan aliran

Nama Fail	Aliran Normal	Aliran Berbahaya	Label
Monday-WorkingHours.pcap_ISCX.csv	529,918	0	Benign
Tuesday-WorkingHours.pcap_ISCX.csv	432,074	13,835	Benign, FTP-Patator, SSH-Patator
Wednesday-WorkingHours.pcap_ISCX.csv	432,074	252,672	Benign, DoS GoldenEye, DoS Hulk, DoS Slowhttptest, DoS Slowris, Heartbleed
Thursday-WorkingHours-Morning-WebAttacks.pcap_ISCX.csv	168,186	290,782	Benign, Web attack-brute force, Web Attack-Sql Injection, Web Attack-XSS
Thursday-WorkingHours-Afternoon-Infilteration.pcap_ISCX.csv	288,566	36	Benign, Infiltration
Friday-WorkingHours-Morning.pcap_ISCX.csv	189,067	1,966	Benign, Bot
Friday-WorkingHours-Afternoon-PortScan.pcap_ISCX.csv	183,910	41,835	Benign, Portscan
Friday-WorkingHours-Afternoon-DDos.pcap_ISCX.csv	127,537	158,930	Benign, DDoS

Jadual 3.4 Bilangan aliran berbahaya

Label	Bilangan aliran
Benign	2,359,087
DoS Hulk	231,072
Portscan	158,930
DDoS	41,835
DoS GeldenEye	10,293
FTP-Patator	7938
SSH-Patator	5897
DoS Slowloris	5796
DoS Slowhttptest	5499
Bot	1966
Web Attack-Brute Force	1507
Web Attack-XSS	652

bersambung...

sambungan...

Infiltration	36
Web Attack-Sql Injection	21
Heartbleed	11

Langkah-langkah untuk pembahagian set data CIC-IDS2017 kepada set data latihan dan pengujian seperti Jadual 3.5 dimana test_ratio adalah 20% dan train_ratio 80% daripada data yang telah diproses seperti Jadual 3.4. Ini bermakna data latihan berjumlah 2,264,432 dan data pengujian berjumlah 566,108. Berdasarkan pembahagian data tersebut didapati bahawa sebanyak 2,359,087 adalah benign dan 471,453 adalah serangan.

Jadual 3.5 Langkah-langkah pembahagian set data latihan dan pengujian

Algoritma pembahagian data CIC-IDS2017

```
function split_dataset(dataset, test_ratio):
    // Shuffle the dataset randomly
    shuffle(dataset)

    // Calculate the number of samples for the test set
    test_size = length(dataset) * test_ratio

    // Split the dataset into training and testing sets
    training_set = dataset[:length(dataset) - test_size]
    testing_set = dataset[length(dataset) - test_size:]

    return training_set, testing_set

// Usage example
dataset = load_cicids2017_dataset()
test_ratio = 0.2  // 20% of the data will be used for testing

training_set, testing_set = split_dataset(dataset, test_ratio)
```


Selain itu, set data CICDDoS2019 mengandungi data trafik rangkaian selama 2 hari. Ringkasan set data yang mengandungi serangan dalam latihan dan pengujian seperti Jadual 3.6. Set data latihan mengandungi 12 serangan DDoS termasuk Network Time Protocol (NTP), Domain Name System (DNS), Lightweight Directory Access Protocol (LDAP), Microsoft SQL Server (MSSQL), Network Basric Input Output System (NetBIOS), Simple Network Management Protocol (SNMP), Simple Service Discovery Protocol (SSDP), User Datagram Protocol (UDP), UDP-Lag, WebDDoS,

SYN dan TFTP. Manakala set data untuk pengujian mengandungi 7 serangan iaitu MSSQL, NetBIOS, Portscan, LDAP, UDP, UDP-Lag and SYN. Manakala ringkasan serangan untuk latihan dan pengujian untuk setiap set data seperti Jadual 3.7. Seterusnya langkah-langkah pembahagian data latihan dan pengujian seperti Jadual 3.8 dengan pembahagian test_ratio 80% dan train_ratio 20%.

Jadual 3.6 Jenis serangan set data CICDDoS2019

Jenis Serangan	Bilangan Aliran
Benign	56,863
DDoS_DNS	5,071,011
DDoS_LDAP	2,179,930
DDoS_MSSQL	4,522,492
DDoS_NetBios	4,093,279
DDoS_NTP	1,202,642
DDoS_SNMP	5,159,870
DDoS_SSDP	2,610,611
DDoS_SYN	1,582,289
DDoS_TFTP	20,082,580
DDoS_UDP	3,134,645
DDoS_UDP-Lag	366,461
DDoS_WebDDoS	439

Jadual 3.7 Kategori set data untuk latihan dan pengujian

Kategori	Latihan	Pengujian
Benign	56,101	17,146
Serangan	997,054	314,716

Jadual 3.8 Langkah-langkah pembahagian set data latihan dan pengujian

Algoritma pembahagian data CICDDoS2019

```
function split_dataset(dataset, test_ratio):
    // Shuffle the dataset randomly
    shuffle(dataset)

    // Calculate the number of samples for the test set
    test_size = length(dataset) * test_ratio

    // Split the dataset into training and testing sets
    training_set = dataset[:length(dataset) - test_size]
    testing_set = dataset[length(dataset) - test_size:]

    return training_set, testing_set
```

bersambung...

...sambungan

```
// Usage example
dataset = load_cicddos2019_dataset()
test_ratio = 0.2  // 20% of the data will be used for testing

training_set, testing_set = split_dataset(dataset, test_ratio)
```

3.4.3 Penilaian Snort

Proses menilai Snort untuk eksperimen ini menggunakan dataset CIC-IDS2017 dan CICDDoS2019. Snort akan menganalisis fail pcap dengan set aturan dan konfigurasi ketika menganalisis CIC-IDS2017 dan CICDDoS2019. Amaran yang dihasilkan oleh Snort kemudian dihantar dan disimpan dalam fail CSV. Fail pcap yang mengandungi semua paket yang diambil semasa eksperimen ditukar menjadi fail CSV. Proses ini boleh dirangkumkan seperti berikut:

1. Tentukan tetingkap masa untuk setiap serangan.
2. Simulasikan serangan dan rekod trafik.
3. Proses data dengan Snort.
4. Parse amaran Snort
5. Tukar rakaman paket ke dalam fail CSV.
6. Labelkan paket berdasarkan kejadian dalam peringatan.
7. Hasilkan metrik penilaian.

Selain daripada proses yang dinyatakan di atas untuk laksana *replay* set data yang telah dikenal pasti, berikut merupakan langkah-langkah atau gambaran Snort beroperasi untuk menganalisis atau mengesan set data yang mengandungi benign mahupun serangan seperti Jadual 3.9.

Jadual 3.9 Pseudocode replay fail pcap dengan Snort 3

Pseudocode : Snort 3
1. Muatkan fail konfigurasi Snort 3.

bersambung...

sambungan...

2. Inisialisasikan Snort 3 dengan konfigurasi yang dimuat.
3. Buka fail pcap untuk dibaca menggunakan command.
   $sudo snort -c /usr/local/etc/snort/snort.lua -r <file.pcap> -k none -A csv -q
4. Selagi tidak tamatnya fail pcap:
   4.1 Baca paket seterusnya dari fail pcap.
   4.2 Hantarkan paket kepada Snort 3 untuk diproses
   4.3 Ulangi sehingga semua paket diproses
5. Tutup fail pcap.
6. Matikan Snort 3
7. Log fail pcap disimpan dalam folder log.
   /var/log/snort
8. Log boleh dianalisis untuk melihat alert dan ketepatan pengesenan.

3.4.4 Penilaian Suricata

Berikut adalah langkah-langkah untuk menilai Suricata menggunakan set data CIC-IDS2017 dan CICDDoS2019:

1. Sedia persekitaran
   a. Persekitaran pembangunan yang sesuai dengan keperluan Suricata
2. Muat turun dan sedia dataset
   a. Muat turun set data CIC-IDS2017 dan CICDDoS2019 dari sumber yang sah dan set data merangkumi serangan yang ingin dinilai.
3. Konfigurasi suricata
   a. Konfigurasi Suricata dengan aturan dan konfigurasi yang sesuai dengan set data CIC-IDS2017 dan CICDDoS2019 serta pengaktifan set peraturan yang relevan.
4. Proses data dengan Suricata
   a. Proses data pcap menggunakan Suricata serta memastikan set peraturan dan konfigurasi mengikut keperluan.
5. *Parse* hasil Suricata

a. Hasil pengesanan berdasarkan kadar FPR, FNR dan prestasi sumber perkakasan.

6. Tukar format pcap ke csv dan pelabelan

a. Tukar format fail pcap kepada csv untuk memudakan dibaca dan dianalisis seterusnya pelabelan paket sama ada benign ataupun serangan

7. Analisis hasil

a. Analisis hasil penilaian dan perlu dinilai kelebihan dan kekurangan IDS.

Selain daripada proses yang dinyatakan di atas untuk laksana *replay* set data yang telah dikenal pasti, berikut merupakan langkah-langkah atau gambaran Suricata beroperasi untuk menganalisis atau mengesan set data yang mengandungi *benign* mahupun serangan seperti Jadual 3.10.

Jadual 3.10 Pseudocode replay fail pcap dengan Suricata

Pseudocode : Suricata

```
# Langkah 1: Pasang Suricata (seperti Lampiran A)

# Langkah 2: Sediakan Fail Konfigurasi
fail_konfigurasi = """
suricata:
  interfaces:
    - eth0
  logging:
    - fast:
        filename: fast.log
        enabled: yes
    - eve-log:
        enabled: yes
        filetype: regular
        filename: eve.json
        types:
          - alert
          - http:
              extended: yes
"""

# Langkah 3: Muat turun Set Peraturan Suricata
            /var/lib/suricata/rules/suricata.rules

# Langkah 4: Mulakan Suricata dalam Mod Luring
fail_pcap = "/media/sf_folder/Monday-WorkingHours.pcap "
```

bersambung...

```
sambungan...
arahan_suricata = "suricata -c /etc/suricata/suricata.yaml -r "/media/sf_folder/Monday-
WorkingHours.pcap ".format(fail_konfigurasi, fail_pcap)
jalankan_arahan(arahan_suricata)

# Langkah 5: Pantau Keluaran
# Memantau log/amaran yang dihasilkan oleh Suricata secara langsung atau menyemak fail log yang
dihasilkan berdasarkan konfigurasi.
/var/log/suricata/
```

3.4.5 Penilaian Zeek

Untuk menilai dengan tepat keupayaan Zeek, penggunaan set data CIC-IDS2017 dan CICDDoS2019 yang mengandungi pelbagai senario trafik rangkaian dunia nyata. Proses adalah seperti berikut:

1. Memahami ciri-ciri dan keupayaannya dan keperluan teknikal untuk pemasangan. Zeek.
2. Muat turun dataset CIC-IDS2017 dan CICDDoS2019.
3. Pemasangan Zeek dalam persekitaran maya dan konfigurasi untuk menganalisis trafik rangkaian daripada dataset CIC-IDS2017 dan CICDDoS2019.
4. Memantau dan analisis log yang dihasilkan. Perbandingan hasil yang dijangka daripada dataset dengan log sebenar yang dihasilkan oleh Zeek untuk menilai ketepatan dan keberkesanannya.
5. Interpretasi hasil dan membuat kesimpulan tentang prestasi dan keupayaan Zeek.

Proses pengesanan trafik berasaskan Zeek berdasarkan paket yang terkandung dalam set data CIC-IDS2017 dan CICDDoS2019 dan akan menghasilkan satu fail yang mengandungi aliran trafik. Setiap aliran yang mengandungi atribut akan disimpan dalam conn.log yang mengandungi sambungan yang dikesan pada peringkat protokol IP, TCP, UDP, ICMP seperti Jadual 3.11.

Jadual 3.11 Atribut Zeek dalam fail conn.log

Atribut		
Timestamp	Duration	Source IP Bytes
Source IP	Source Bytes	Response Packets
Source Port	Response Bytes	Response IP Bytes
Destination IP	Conn_state	Tunnel Parents
Destination Port	Missed Bytes	Label
Protocol	History	
Service	Soutce Packets	

Selain daripada proses yang dinyatakan di atas untuk laksana *replay* set data yang telah dikenal pasti, berikut merupakan langkah-langkah atau gambaran Zeek beroperasi untuk menganalisis atau mengesan set data yang mengandungi *benign* mahupun serangan seperti Jadual 3.12.

Jadual 3.12 Pseudocode replay fail pcap dengan Zeek

Pseudocode : Zeek

```
# Load necessary modules
@load base/frameworks/packet-analysis

# Define event handlers for analyzing packets
event packet_handler(packet: Packet) {
   # Extract relevant fields from the packet
   local src_ip = packet$ip_src;
   local dst_ip = packet$ip_dst;
   local src_port = packet$src_port;
   local dst_port = packet$dst_port;
   local protocol = packet$proto;

   # Perform analysis based on packet contents
   if (protocol == TCP) {
      # TCP analysis
      print "TCP packet from", src_ip, "to", dst_ip, "on port", dst_port;
      # Add more analysis here as needed
   } else if (protocol == UDP) {
      # UDP analysis
      print "UDP packet from", src_ip, "to", dst_ip, "on port", dst_port;
      # Add more analysis here as needed
   } else if (protocol == ICMP) {
      # ICMP analysis
      print "ICMP packet from", src_ip, "to", dst_ip;
      # Add more analysis here as needed
   } else {
      # Other protocols
      print "Unknown protocol packet from", src_ip, "to", dst_ip;
   }
```

bersambung...

```
sambungan...
}
# Main entry point
event zeek_init() {
    # Enable packet analysis on the provided pcap file
    local pcap_file = "/media/sf_folder/Monday-WorkingHours.pcap ";
    print "Analyzing pcap file:", pcap_file;
    pcap_open(pcap_file);
```

3.4.6 Set Peraturan

Snort dan Suricata merupakan IDS berasaskan tandatangan yang memerlukan pangkalan data set peraturan untuk mengenal pasti dan mengesan ancaman yang diketahui. Manakala berasaskan anomali, ia membentuk garis panduan tingkah laku normal dan memberi amaran atau mengambil tindakan apabila penyimpangan daripada garis panduan ini dikesan. Ia juga mencari corak atau aktiviti luar biasa yang mungkin menunjukkan serangan.

Sehubungan itu, set peraturan ini terpakai pada perisian Snort dan Suricata. Eksperimen ini akan menggunakan peraturan snort3-community bagi Snort seperti Rajah 3.5 manakala Emerging Threats (ET) Open untuk Suricata seperti Rajah 3.6. Snort menawarkan tiga set peraturan yang berbeza iaitu komuniti, pengguna berdaftar dan set peraturan langganan. Set peraturan komuniti adalah percuma dan dikemas kini setiap hari, manakala set peraturan berdaftar adalah 30 hari di belakang set peraturan pelanggan yang paling komprehensif dan dibayar oleh mengikut tempoh langganan. Manakala Emerging Threats, Proofpoint, telah menyediakan set peraturan ET Open dan ET Pro. Set peraturan ET Open adalah percuma dan dikekalkan oleh komuniti keselamatan, sementara set peraturan ET Pro dikekalkan oleh pasukan penyelidikan Emerging Threats. Set peraturan ini direka terutamanya untuk Suricata tetapi boleh digunakan dalam Snort 3. Pecahan set peraturan dan bilangan peraturan ditunjukkan dalam Jadual 3.13.

Jadual 3.13 Set Peraturan Terperinci

Set Peraturan	Bilangan Peraturan
Snort3-community	4,234
ET Open	46,886

Zeek tiada set peraturan tetapi menggunakan skrip dalam bahasa yang direka untuk analisis rangkaian dan maklumat yang diperlukan. Bahasa skrip ini membenarkan pengguna untuk mewujudkan peraturan dan polisi yang spesifik kepada keperluan. Antara contoh asas skrip dan peraturan serta *logging* adalah seperti Rajah 3.7.

```
alert tcp $EXTERNAL_NET [80,143] -> $HOME_NET any
(
    msg:"MALWARE-OTHER Win.Ransomware.Agent payload download attempt";
    flow:to_client,established;
    file_data; content:"secret_encryption_key",fast_pattern,nocase;
    service:http, imap;
    classtype:trojan-activity;
    sid:1;
)
```

Rajah 3.5 Contoh Format Set Peraturan Snort3

Sumber: Snort

```
alert http $HOME_NET any -> $EXTERNAL_NET any (msg:"HTTP GET
Request Containing Rule in URI"; flow:established,to_server;
http.method; content:"GET"; http.uri; content:"rule"; fast_pattern;
classtype:bad-unknown; sid:123; rev:1;)
```

Rajah 3.6 Contoh Format Set Peraturan Suricata

Sumber : Suricata

```
module BugzBruteforcing;

export {
    redef enum Notice::Type += {
        ## Indicates that a host performing HTTP requests leading to
        ## excessive HTTP auth errors was detected.
        HTTP_BugzBruteforcing_Attacker,
        ## Indicates that a host was seen to respond excessive HTTP
        ## auth errors. This is tracked by IP address as opposed to
        ## hostname.
        HTTP_BugzBruteforcing_Victim,
    };

    const ports_int: set[port] = { 80/tcp, 443/tcp } &redef;

    redef enum Log::ID += { LOG };

    # Let's tag the http item
    redef enum HTTP::Tags += {
        ## HTTP status code 401, describing a HTTP auth error
        HTTP_AUTH_ERROR,
        ## HTTP describing a successful HTTP auth
        HTTP_AUTH_SUCCESS,
    };

    type Info: record {
        ts:                time        &log;
        uid:               string      &log;
        id:                conn_id     &log &optional;
        cluster_client_ip: string      &log &optional;
        status_code:       count       &log &optional;
        host:              string      &log &optional;
        uri:               string      &log &optional;
        username:          string      &log &optional;
        auth_success:      bool        &log &optional;
    };
```

Rajah 3.7 Contoh Bahasa Skrip Zeek untuk Ancaman Bruteforce

Sumber : Github

3.4.7 Kaedah Pelaksanaan

Kaedah yang digunakan semasa penghasilan kajian ini adalah penting bagi memastikan kelancaran serta sebarang kelemahan dapat dikenal pasti pada peringkat awal. Snort 3.1.6.0, Suricata 7.0.2 dan Zeek 6.1.0 dipasang dengan konfigurasi lalai dan fail output yang dikenal pasti disimpan pada direktori log lalai. Setiap NIDS dipasang pada mesin maya tanpa sebarang aktiviti untuk mendapat keputusan yang tepat. Setiap IDS di laksana menggunakan set peraturan dan skrip yang telah ditetapkan awal sama ada menggunakan Snort3-Community atau *emerging threats* atau skrip seperti dinyatakan pada para 3.4.6. Penilaian di laksana berdasarkan prestasi penggunaan sumber, amaran dan ketepatan pengesanan.

Metrik prestasi dikira dan dibandingkan antara tiga NIDS menggunakan dua set data CIC-IDS2017 dan CICDDoS2019 menggunakan kaedah *replay* set data berdasarkan parameter penilaian berikut:

1. Penggunaan CPU dinilai dari segi peratusan jumlah sumber pemprosesan yang digunakan semasa trafik diproses menggunakan alatan sokongan seperti top, multi-load.

2. Penggunaan memori (RAM) dinilai dari segi peratusan jumlah memori fizikal yang digunakan semasa trafik diproses menggunakan alatan sokongan seperti top, multi-load.

3. Kadar pengesanan diukur mengikut setiap NIDS berdasarkan konfigurasi dan set peraturan mahupun skrip yang telah ditetapkan (default).

Kaedah ujian dan rakaman ini menyediakan cara yang ringkas namun berkesan untuk membandingkan keupayaan pengesanan perisian NIDS. Kelebihan pendekatan ini terletak pada kesederhanaan dan kebolehgunaan langsung, membolehkan untuk fokus penilaian setiap alat tanpa memerlukan analisis statistik yang kompleks atau meluas tetapan eksperimen.

3.5 KESIMPULAN

Metodologi kajian ini melibatkan empat peringkat iaitu analisis data, pra-pemprosesan data, pengekstrakan ciri-ciri dan pengujian. Kajian ini menggunakan set data CIC-IDS 2017 dan CICDDoS2019 yang mengandungi pelbagai jenis serangan. Ianya untuk melihat keberkesanan perisian IDS sumber terbuka yang telah dikenal pasti. Kaedah pelaksanaan merupakan panduan bagi pelaksanaan eksperimen ke atas perisian IDS sumber terbuka. Pada akhir kajian, satu perbandingan dan keberkesanan perisian sistem pengesanan pencerobohan (IDS) sumber terbuka iaitu Snort, Suricata dan Zeek dapat dikenal pasti dan pemilihan perisian yang bersesuaian bagi memastikan aktiviti berniat jahat dapat dikesan di peringkat rangkaian organisasi atau bisnes.

BAB IV

PELAKSANAAN SISTEM PENGURUSAN PENCEROBOHAN RANGKAIAN (NIDS) MENGGUNAKAN PERISIAN SUMBER TERBUKA

4.1 PENGENALAN

Bab ini akan membincangkan pelaksanaan sistem pengesanan pencerobohan rangkaian (NIDS) menggunakan perisian sumber terbuka berdasarkan objektif dan metodologi yang dipilih. Pelaksanaan ini melibatkan output, keputusan diperolehi dan analisis secara keseluruhan.

4.2 SENIBINA EKSPERIMEN

Eksperimen ini dijalankan pada mesin maya (virtual machine) iaitu VirtualBox Versi 6.1.12 dipasang dengan sistem pengoperasian Ubuntu 20.04, 4GB RAM dan 40GB storan. Snort, Suricata dan Zeek dipasang pada mesin maya dan diuji secara individu dengan versi masing-masing Snort 3.1.6.0, Suricata 7.0.2 dan Zeek 6.1.0.

Selain itu, PulledPork versi 0.8.0 telah dipasang pada mesin maya Snort untuk menguruskan set peraturan terkini daripada laman web Snort terutamanya peraturan yang dilanggan. Set peraturan yang dipasang pada Snort juga adalah set peraturan daripada komuniti perisian pengurusan peraturan untuk Snort. Ia memuat turun peraturan terkini dari laman web Snort. PulledPork ditulis semula dari Perl ke Python 3 dan kini dipanggil PulledPork. Kedua-dua versi dipasang, tetapi PulledPork versi 0.8.0 digunakan sebagai perisian pengurusan peraturan kerana ciri-ciri dan keserasian yang lebih baik dengan Snort 3. Suricata versi 4 ke atas dilengkapi dengan pengurusan peraturan dan tidak memerlukan perisian pengurusan peraturan lain.

Snort dan Suricata dipasang pada mesin maya yang sama dengan *extra logging* yang diaktifkan untuk mengumpul tingkah laku amaran. Penggunaan sumber dan bilangan amaran yang dihasilkan direkodkan untuk kedua-dua Snort dan Suricata semasa berjalan dengan fail pcap yang berbeza. Fail log diperiksa dan disiasat untuk menganalisis amaran yang dihasilkan oleh Snort dan Suricata semasa eksperimen.

4.3 KEPUTUSAN

Eksperimen memberi tumpuan kepada tiga aspek iaitu penggunaan sumber, tingkah laku amaran dan keberkesanan pengesanan serangan. Penggunaan sumber merujuk kepada seberapa cekap IDS menggunakan sumber sistem seperti CPU, memori dan ruang storan. Ianya untuk membandingkan penggunaan sumber Snort dan Suricata untuk menentukan IDS mana yang lebih cekap dari segi penggunaan sumber. Manakala tingkah laku amaran merujuk kepada seberapa berkesan IDS mengesan dan melaporkan aktiviti atau serangan *malicious* pada rangkaian. Ianya untuk membandingkan tingkah laku amaran Snort dan Suricata yang dapat mengesan dan memberi amaran mengenai serangan siber yang berpotensi.

Hasil penggunaan sumber dan ujian tingkah laku amaran dianalisis untuk membuat kesimpulan mengenai prestasi dan keberkesanan Snort dan Suricata. Analisis ini membantu dalam memahami kekuatan dan kelemahan setiap IDS dan dapat membantu dalam membuat keputusan yang tepat mengenai IDS yang hendak digunakan dalam perlindungan rangkaian.

4.3.1 Objektif 1: Untuk Menilai Prestasi Penggunaan Sumber seperti CPU dan Memori

Penilaian prestasi tiga NIDS ke atas CPU dan penggunaan memori iaitu kadar pemprosesan dan sumber memori yang digunakan oleh proses NIDS semasa menganalisis paket yang diterima dan menghasilkan amaran. Maksimum beban pemproses dan penggunaan memori ialah 100%. Perisian yang paling berkesan ialah penggunaan lebih sedikit sumber. Untuk menentukan kadar pengeluaran CPU dan memori catatan kadar penggunaan CPU, mengambil nilai tertinggi yang muncul pada antara muka TOP semasa tempoh ujian.