TESIS

PERBANDINGAN METODE ANOMALY BASED DAN SIGNATURE BASED DALAM PENDETEKSIAN SCANNING VULNERABILITY PADA WEBSITE

Disusun oleh:

Nama : **Ismail Puji Saputra**
NIM : **21.55.1040**
Konsentrasi : **Digital Transformation Intelligence**

PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2022

TESIS

PERBANDINGAN METODE ANOMALY BASED DAN SIGNATURE BASED DALAM PENDETEKSIAN SCANNING VULNERABILITY PADA WEBSITE

COMPARISON OF ANOMALY BASED AND SIGNATURE BASED METHODS IN DETECTION OF SCANNING VULNERABILITY ON WEBSITES

Diajukan untuk memenuhi salah satu syarat memperoleh derajat Magister

Disusun oleh:

Nama : **Ismail Puji Saputra**
NIM : **21.55.1040**
Konsentrasi : **Digital Transformation Intelligence**

PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA
2022

HALAMAN PENGESAHAN

PERBANDINGAN METODE ANOMALY BASED DAN SIGNATURE BASED DALAM PENDETEKSIAN SCANNING VULNERABILITY PADA WEBSITE

COMPARISON OF ANOMALY BASED AND SIGNATURE BASED METHODS IN DETECTION OF SCANNING VULNERABILITY ON WEBSITES

Dipersiapkan dan Disusun oleh

Ismail Puji Saputra

21.55.1040

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Rabu, 04 Januari 2023

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer

Yogyakarta, 04 Januari 2023
Rektor

Prof. Dr. M. Suyanto, M.M.
NIK. 190302001

HALAMAN PERSETUJUAN

PERBANDINGAN METODE ANOMALY BASED DAN SIGNATURE BASED DALAM PENDETEKSIAN SCANNING VULNERABILITY PADA WEBSITE

COMPARISON OF ANOMALY BASED AND SIGNATURE BASED METHODS IN DETECTION OF SCANNING VULNERABILITY ON WEBSITES

Dipersiapkan dan Disusun oleh

Ismail Puji Saputra

21.55.1040

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Rabu, 04 Januari 2023

Pembimbing Utama

Prof. Dr. Ema Utami, S.Si., M.Kom.
NIK. 190302037

Anggota Tim Penguji

Dr. Andi Sunyoto, M.Kom.
NIK. 190302052

Pembimbing Pendamping

Alva Hendi Muhammad, S.T., M.Eng., Ph.D
NIK. 190302493

Hanafi, S.Kom., M.Eng., Ph.D.
NIK. 190302024

Prof. Dr. Ema Utami, S.Si., M.Kom.
NIK. 190302037

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer

Yogyakarta, 04 Januari 2023
Direktur Program Pascasarjana

Prof.Dr. Kusrini, M.Kom.
NIK. 190302106

HALAMAN PERNYATAAN KEASLIAN TESIS

Yang bertandatangan di bawah ini,

Nama mahasiswa : Ismail Puji Saputra
NIM : 21.55.1040
Konsentrasi : Digital Transformation Intelligence

Menyatakan bahwa Tesis dengan judul berikut:
PERBANDINGAN METODE ANOMALY BASED DAN SIGNATURE BASED DALAM PENDETEKSIAN SCANNING VULNERABILITY PADA WEBSITE
Dosen Pembimbing Utama : **Prof. Dr. Ema Utami, S.St., M.Kom.**
Dosen Pembimbing Pendamping : **Alva Hendi Muhammad, S.T., M.Eng., Ph.D**

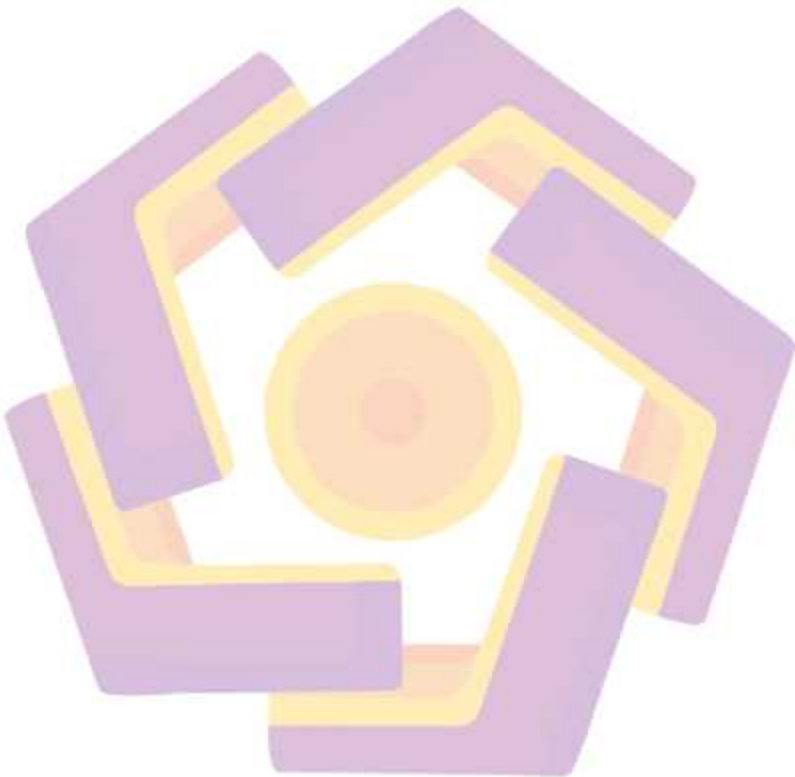
1. Karya tulis ini adalah benar-benar ASLI dan BELUM PERNAH diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian SAYA sendiri, tanpa bantuan pihak lain kecuali arahan dari Tim Dosen Pembimbing
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab SAYA, bukan tanggung jawab Universitas AMIKOM Yogyakarta
5. Pernyataan ini SAYA buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka SAYA bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi

Yogyakarta, 04 Januari 2023
Yang Menyatakan,

Ismail Puji Saputra

HALAMAN PERSEMBAHAN

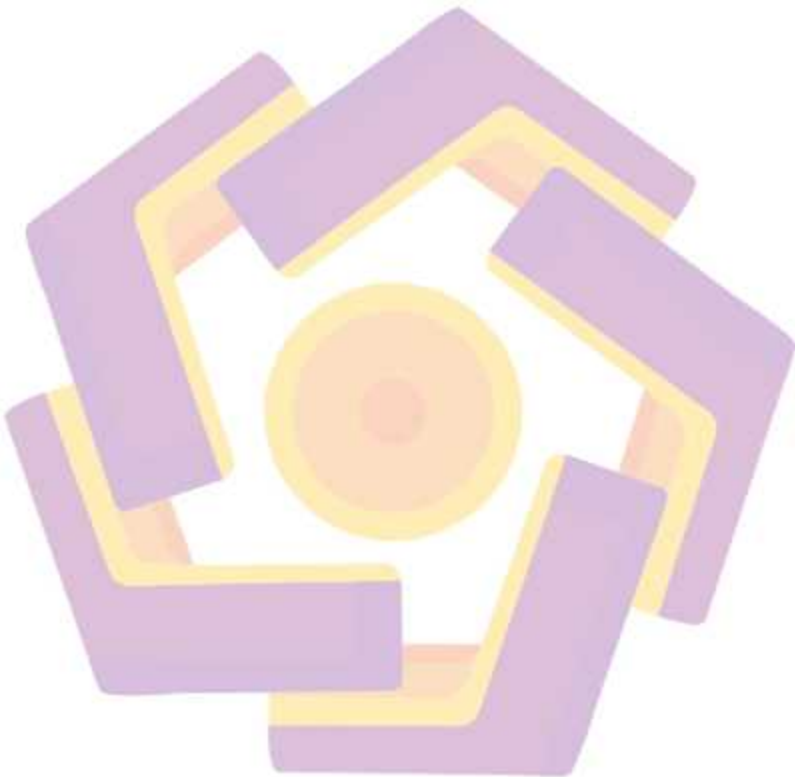
Karya tulis ini ku persembahkan untuk kedua orang tuaku dan kakak-kakakku. Semoga karya tulis ini bermanfaat dan berguna bagi yang membacanya.



HALAMAN MOTTO

"Love is decision not an emotion."

Lao Tzu



KATA PENGANTAR

Puji syukur Alhamdulillah kehadirat Allah SWT, atas segala rahmat-NYA sehingga penulis dapat menyelesaikan Tesis yang berjudul "**PERBANDINGAN METODE ANOMALY BASED DAN SIGNATURE BASED DALAM PENDETEKSIAN SCANNING VULNERABILITY PADA WEBSITE**".

Pada kesempatan ini, penulis menyampaikan terima kasih kepada :

1. Kedua orang tua dan kakak-kakakku atas semua do'a dan dukungan sehingga Tesis ini dapat diselesaikan.
2. Prof. Dr. Ema Utami, S.Si., M.Kom, dan Dr. Suwanto Raharjo, S.Si., M.Kom. yang telah memberikan inspirasi dan bimbingan dalam menulis Tesis ini.
3. Alva Hendi Muhammad, S.T., M.Eng., Ph.D yang memberikan bimbingan dan masukan dalam menulis Tesis ini.

Tidak ada yang sempurna di alam ini, karena kesempurnaan hanyalah milik-NYA, Semoga ketidak sempurnaan Tesis ini, tetap dapat bermanfaat bagi yang membacanya.

Yogyakarta, 04 Januari 2023

Penulis

DAFTAR ISI

TESIS

PERBANDINGAN METODE ANOMALY BASED DAN SIGNATURE BASED DALAM PENDETEKSIAN SCANNING

Halaman Judul ii

Halaman Pengesahan iii

Halaman Persetujuan iv

Halaman Persembahan vi

Halaman Motto vii

Kata Pengantar viii

Daftar Isi ix

Daftar Tabel xi

Daftar Gambar xii

Daftar Istilah xv

Intisari xvi

Abstract xvii

BAB I PENDAHULUAN 1

1.1. Latar Belakang Masalah 1

1.2. Rumusan Masalah 5

1.3. Batasan Masalah 6

1.4. Tujuan Penelitian 7

1.5. Manfaat Penelitian 8

BAB II TINJAUAN PUSTAKA 9

2.1. Tinjauan Pustaka 9

2.2. Keaslian Penelitian 13
2.3. Landasan Teori 22
BAB III METODE PENELITIAN 28
3.1. Metode Penelitian 28
BAB IV HASIL PENELITIAN DAN PEMBAHASAN 36
4.1. *Developing IDS Anomaly-based* 36
4.2. *Developing IDS Signature-based* 44
4.3. *Testing* 50
4.4. Evaluasi *Testing* 65
4.5. Hasil 72
BAB V PENUTUP 78
5.1. Kesimpulan 78
5.1. Saran 79
DAFTAR PUSTAKA 81
LAMPIRAN 84

DAFTAR TABEL

Tabel 2.1. Matriks literatur review 13

Tabel 4.1. Atribut Sniffing Anomaly-bases 40

Tabel 4.2. Data Sample Uji Coba Serangan 50

Tabel 4.3. Tools yang digunakan untuk testing 51

Tabel 4.4. Performa metode *Anomaly-based* 62

Tabel 4.5. Performa metode *Signature-based* 64

Tabel 4.6. Perbandingan Kecepatan Deteksi Acunetix 65

Tabel 4.7. Perbandingan Kecepatan Deteksi Acunetix setelah perubahan interval 67

Tabel 4.8. Hasil perbandingan deteksi berbagai *tools* 67

Tabel 4.9. Hasil perbandingan kecepatan deteksi Acunetix 72

Tabel 4.10. Hasil perbandingan deteksi Acunetix setelah perubahan interval 73

Tabel 4.11. Hasil perbandingan deteksi berbagai *tools* 74

Tabel 4.12. Rekap perbandingan performa deteksi metode 75

Tabel 4.13. Hasil perbandingan penelitian sebelumnya 76

DAFTAR GAMBAR

Gambar 2.1. *Anomaly Based Method* 24

Gambar 2.2. *Signature Based Method* 25

Gambar 2.3. Cara Kerja *Firewall* 26

Gambar 2.4. Terminal MikroTik RouterOS 27

Gambar 3.1. Alur Penelitian 29

Gambar 3.2. Flowchart IDS 32

Gambar 3.3. Proses Eksperimen dan Evaluasi metode 33

Gambar 3.4. Proses Pengumpulan Data Uji 34

Gambar 4.1. Topologi IDS 37

Gambar 4.2. Starting *Sniffing* 37

Gambar 4.3. *File* vul.pcap 38

Gambar 4.4. *Download file* vul.pcap ke *server anomaly-based* 38

Gambar 4.5. *file* vul.pcap pada server 39

Gambar 4.6. *setting scheduller* 39

Gambar 4.7. *command* konversi PCAP ke CSV *anomaly-based* 40

Gambar 4.8. *command insert* data ke *database anomaly-based* 41

Gambar 4.9. Isi tabel vulscan 41

Gambar 4.10. Scripting mikrotik fire.php 42

Gambar 4.11. File fire.php 42

Gambar 4.12. Aturan *Firewall anomaly-based* 43

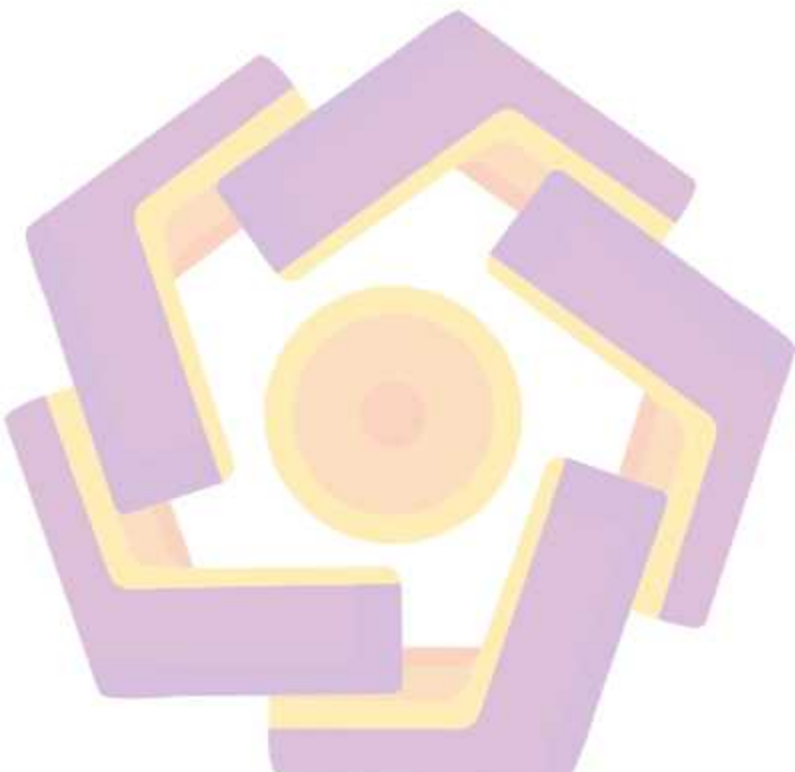
Gambar 4.13. Implementasi aturan *firewall* hasil analisis *anomaly-based* 44

Gambar 4.14. setting scheduler pada script fetch MikroTik 44
Gambar 4.15. langkah collecting signature Acunetix VulScan 45
Gambar 4.16. VulScan Attack untuk mendapatkan pola dari Acunetix 46
Gambar 4.17. Proses seleksi data sebelum di insert ke dalam tabel signature 46
Gambar 4.18. Tabel signature yang berisi request URL pada Acunetix..... 47
Gambar 4.19. Proses pencocokan content fire.php pada signature-based 48
Gambar 4.20. hasil query mekanisme signature-based..... 48
Gambar 4.21. Implementasi aturan firewall hasil analisis signature-based..... 49
Gambar 4.22. proses labeling data dari masing-masing metode..... 49
Gambar 4.23. Acunetix berhenti karena terdeteksi IDS Anomaly-based 51
Gambar 4.24. MikroTik firewall memblokir Acunetix dengan Anomaly-based.. 52
Gambar 4.25. Acunetix berhenti karena terdeteksi IDS Signature-based..... 52
Gambar 4.26. MikroTik firewall memblokir Acunetix dengan Signature-based . 53
Gambar 4.27. MikroTik firewall memblokir Skipfish dengan Anomaly-based ... 54
Gambar 4.28. Skipfish attack pada anomaly-based 54
Gambar 4.29. Skipfish attack pada signature-based 55
Gambar 4.30. Nikto attack pada anomaly-based 56
Gambar 4.31. log firewall rule blocking Nikto attack pada anomaly-based..... 56
Gambar 4.32. firewall rule blocking Nikto attack pada anomaly-based..... 56
Gambar 4.33. result Nikto attack pada signature-based..... 57
Gambar 4.34. result Pentest-Tool pada anomaly-based..... 57
Gambar 4.35. record pada tabel setelah serangan Pentest-Tool..... 58
Gambar 4.36. alur metode anomaly-based dalam menentukan serangan 59

Gambar 4.37. alur menghitung jumlah error 404 pada tanggal 01/08/2023 60
Gambar 4.38. alur menghitung jumlah host yang melakukan error 404 pada tanggal 01/08/2023 60
Gambar 4.39. proses seleksi serangan pada anomaly-based. 61
Gambar 4.40. Alur kerja metode signature-based. 63
Gambar 4.41. signature-based mendeteksi aktivitas vulnerability scanner. 64
Gambar 4.42. Speed deteksi setelah merubah interval mekanisme anomaly-based. 66
Gambar 4.43. Speed deteksi setelah merubah interval mekanisme signature-based. 66
Gambar 4.44. colecting data pola Skipfish signature-based. 68
Gambar 4.45. Pola Skipfish pada tabel signature.. 69
Gambar 4.46. Query Pola Skipfish pada tabel signature.. 70
Gambar 4.47. Contoh penerapan whitelist pada mekanisme.. 71

DAFTAR ISTILAH

Threshold: ambang batas melakukan *error* 404 dalam satu hari



INTISARI

Teknologi informasi sebagian besar terdiri dari website yang di tuntut untuk dapat di akses kapan saja dan di mana saja, hal tersebut dapat meningkatkan resiko serangan pada teknologi informasi berbasis web. Serangan pada website di awali dengan proses pencarian kelemahan pada website (website vulnerability scanning) menggunakan software otomatis khusus guna menemukan kelemahan yang ada pada website, akan sangat berbahaya apabila terdapat seseorang yang tidak memiliki wewenang menemukan kelemahan tersebut, karena akan mengakibatkan kelemahan tersebut di eksploitasi.

Penelitian ini akan membahas keamanan jaringan internet di dalam mendeteksi dan mengantisipasi, fokus pada penelitian ini adalah memanfaatkan metode *anomaly-based* dan *signature-based* di dalam mendeteksi dan mengantisipasi serangan *web vulnerability scanner*, berbeda dengan penelitian sebelumnya yang tidak menganggap *web vulnerability scanning* sebagai serangan.

Anomaly-based sendiri merupakan sebuah metode yang digunakan untuk mendeteksi serangan pada sistem, dengan mengumpulkan profil untuk menemukan nilai normal sebuah fenomena pada sistem, sehingga apabila terdapat fenomena yang tidak sama dengan profil tersebut akan dianggap sebagai sebuah serangan. Sedangkan *Signature-based* adalah sebuah metode yang melakukan proses pembentukan profil sebuah serangan dari profil yang telah di dapatkan dari penyerang, profil ini akan di simpan di dalam basis data, sehingga apabila terdapat profil yang sama dengan profil serangan yang tersimpan pada basis data, profil tersebut akan dianggap sebagai serangan.

Penelitian ini berhasil membangun mekanisme otomatis dalam mendeteksi serangan *web vulnerability scanner*, Selain itu penelitian ini membandingkan kecepatan deteksi, kemapuan deteksi dari berbagai *tools* akurasi, presisi dan sensitivitas dari masing-masing metode yang mencapai kesimpulan di mana dalam hal kecepatan, akurasi dan sensitivitas *signature-based* lebih unggul dari metode *anomaly-based*. Namun dalam hal mendeteksi berbagai *tools anomaly*-based lebih unggul dan dapat mendeteksi serangan yang sebelumnya belum pernah di petakan.

Kata kunci: *Vulnerability Scanner, Anomaly-based, Signature-based, Network Security, MikroTik*

ABSTRACT

Information technology mostly consists of websites that are required to be accessed anytime and anywhere, this can increase the risk of attacks on web-based information technology. The attack on the website begins with the process of searching for weaknesses in the website (website vulnerability scanning) using special automatic software to find weaknesses in the website, it will be very dangerous if someone does not have the authority to find these weaknesses because it will result in these weaknesses being exploited.

This research will discuss internet network security in detecting and anticipating, the focus of this research is to utilize anomaly-based and signature-based methods in detecting and anticipating web vulnerability scanner attacks, in contrast to previous research which did not consider web vulnerability scanning as an attack.

Anomaly-based itself is a method used to detect attacks on the system, by collecting profiles to find the normal value of a phenomenon on the system, so that if there is a phenomenon that is not the same as the profile it will be considered an attack. Whereas Signature-based is a method that performs the process of profiling an attack from the profile that has been obtained from the attacker, this profile will be stored in the database, so that if there is a profile that is the same as the attack profile stored in the database, the profile will be considered as an attack.

This study succeeded in building an automatic mechanism for detecting web vulnerability scanner attacks. In addition, this study compared the speed of detection, the detection capability of various tools, and the accuracy, precision, and sensitivity of each method which concluded in terms of speed, accuracy, and sensitivity of the signature- based is superior to the anomaly-based method. However, in terms of detecting various anomaly-based tools, they are superior and can detect attacks that have never been mapped before.

Keyword: Vulnerability Scanner, Anomaly-based, Signature-based, Network Security, MikroTik

BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Transformasi digital pada setiap aspek kehidupan masyarakat menyebabkan semakin besarnya peran teknologi informasi dan komunikasi. teknologi informasi sebagian besar berbasis *web* yang dituntut untuk dapat diakses kapan saja dan dimana saja sehingga memperbesar peluang serangan terhadap teknologi informasi (BSSN, 2021). Tujuan serangan biasanya untuk mendapatkan data berharga, mendapatkan keuntungan *financial* dan menghancurkan prinsip utama kemanan sistem informasi yaitu *Confidentiality, Integrity,* dan *Availability* (Malkawi, et al., 2021). Banyaknya *tools* yang free dan sangat mudah ditemukan di internet dapat digunakan dalam melakukan serangan yang dapat terjadi kapan saja terhadap suatu teknologi informasi, sehingga keamanan teknologi informasi memerlukan mekanisme pengamanan secara *realtime* guna mencegah terjadinya serangan (Risqiwati, et al., 2018).

Serangan pada *website* biasanya diawali dengan proses menemukan celah terhadap *website* tersebut (*website vulnerability scanner*) dengan menggunakan *software* tertentu yang mampu menemukan kelamahan suatu *website* secara spesifik. Proses *scanning vulnerability* pada suatu *website* sangat penting bagi pengembang *website* tersebut untuk meningkatkan keamanan pada *website* dan menutup celah yang telah terdeteksi oleh *website vulnerability scanner* (Zirwan, 2022). penelitian yang dilakukan oleh Prasetyo dan Lee pada

tahun 2021 menyatakan bahwa *vulnerability scanner* dengan menggunakan *tool* Acunetix dapat mendeteksi kelemahan yang ada pada *website* pay2home. Namun akan sangat berbahaya apabila yang mengetahui kelemahan pada *website* bukanlah orang yang memilki wewenang terhadap *website* tersebut, maka untuk itu perlu pencegahan *vulnerability scanning* dari orang yang tidak bertanggung jawab, sistem kerja *vulnerability scanning* biasanya melakukan serangan secara acak dengan menebak *uniform resource locator* (*url*) pada suatu *website* dan membaca *http response* dari *website* target, untuk itu *website vulnerability scanning* memiliki efek samping yaitu dengan munculnya *http response error* apabila *request* yang diminta tidak sesuai dengan *website* misalnya 403,404 dan sebagainya, selain itu untuk *tool* tertentu seperti Acunetix *web scanner* selalu melakukan *checking* terhadap *link* statis yang dapat dianggap sebagai *signature* (tanda tangan) *software* tersebut ketika melakukan *scanning*. Dari kedua fenomena tersebut dapat dimungkinkan untuk mendeteksi *vulnerability scanner* dengan memanfatkan *anomaly based* dan *signature based*.

Anomaly based adalah salah satu teknik didalam mendeteksi serangan pada sebuah sistem, teknik ini melibatkan pola trafik serangan yang berbeda dengan trafik normal dalam mendeteksi kejanggalan, apabila terjadi kejanggalan didalam sebuah trafik maka trafik tersebut dianggap sebagai sebuah percobaan serangan (Fadhlurrohman, et al., 2021). Dalam Pendeteksian dengan *anomaly based* tidak memerlukan *database* pola dalam mencocokan trafik dalam mendeteksi serangan (Bahtiar, et al., 2021).

Berbeda dengan *anomaly based*, *signature based* memiliki *database pola* yang berisi pola sebuah serangan, menurut penelitian Kumar pada tahun 2021 *signature based* merupakan teknik mencocokan *packet* dari pengguna menuju ke *server*, apabila *packet* tersebut cocok dengan *database pola* maka *packet* akan di *tolak* dan dianggap sebagai serangan. *Signature based* memiliki sebuah kelemahan yaitu tingkat akurasinya ditentukan oleh informasi yang ada pada *database pola* (Stiawan, 2009). Untuk memaksimalkan deteksi serangan *database pola* harus diupdate secara berkala hal ini untuk memperkaya *signature* serangan, sehingga pendeteksian menjadi lebih akurat (Khraisat, et al., 2019).

Pengembangan beberapa metode dalam mendeteksi intrusi pada jaringan terbagi menjadi dua yaitu *anomaly based* dan *signature based* (Maseer, et al., 2021). Beberapa penelitian dapat menggambarkan kelebihan dan kekurangan kedua metode tersebut dalam mendeteksi intrusi pada jaringan, penelitian yang dilakukan Fadhlurrohman pada tahun 2021 menggunakan metode *anomaly based* dalam mendeteksi serangan siber dengan menggunakan algoritma *decision tree*, IDS (*intrusion detection system*) akan mendeteksi sebuah serangan apabila terjadi perubahan lalu lintas paket melebihi *threshold* namun dengan metode ini masih memungkinkan terjadinya *false detection*. Penelitian selanjutnya yang dilakukan oleh Haryani pada tahun 2021 memanfaatkan *firewall* MikroTik dalam mendeteksi serangan *ping flood* dengan memblokir paket ICMP (*internet control message protocol*) yang melebihi *treshold* hanya dalam waktu empat detik. Penelitian serupa juga pernah dilakukan oleh Syahputra pada tahun 2020 yang memanfaatkan MikroTik dalam mengamankan jaringan dari serangan DDOS

(*distribution denial of service*) dengan membatasi paket ICMP dan berhasil menjamin stabilitas koneksi internet pada jaringan. Metode-metode diatas adalah metode yang digunakan dalam mendeteksi anomaly pada sebuah jaringan dengan menandai sebuah paket yang tidak normal sebagai sebuah serangan.

Penelitian tentang *signature base* dalam mendeteksi serangan juga sangat beragam, Hidayat pada tahun 2018 berhasil menandai pola serangan winboxpoc yang selalu mengakses *file* user.dat (berisi *username* dan *password*) pada *router* MikroTik dan memblokir setiap koneksi yang mengakses *file* tersebut, sehingga serangan akses *ilegal* ke MikroTik berhasil digagalkan. Pada tahun 2022 Widodo memanfaatkan SNORT (IDS *signature based*) dalam mendeteksi intrusi dan berhasil, SNORT memberikan notifikasi apabila terdapat aktifitas yang melanggar rule. Penelitian Sau pada tahun 2021 yang memadukan SNORT dengan RITA (*real intelligence threat analytics*) yang memanfaatkan algoritma statistik dan *K-means clustering*, dalam mendeteksi aktifitas komputer *zombie* yang mengirimkan sinyal secara *periodic* ke komputer *server command and control* milik penyerang (*beaconing*), hasilnya RITA mampu mendeteksi serangan yang tidak mampu dideteksi oleh SNORT. Sedangkan penelitian yang dilakukan Iman pada 2019 memadukan algoritma RIPPER dan K-nearest neighbour dengan akurasi yang sangat tinggi yaitu 99,89522%.

Berdasarkan latar belakang tersebut, penelitian ini akan membangun mekanisme otomatis dalam mendeteksi dan mengantisipasi serangan *vulnerability scanner* dengan menggunakan *anomaly based* dan *signature based* dalam mendeteksi *scanning vulnerability* pada website serta menghitung nilai akurasi

dari masing-masing metode. Mekanisme yang di bangun akan di implementasikan menggunakan routerboard MikroTik. Masalah dalam penelitian ini adalah bagaimana membangun mekanisme otomatis yang dapat digunakan untuk mendeteksi serta mengantisipasi trafik yang dianggap sebagai *anomaly* dan mengumpulkan pola serangan guna membangun *database pola* yang akan mendeteksi serangan berbasis *signature*, selanjutnya bagaimana cara mengintegrasikan mekanisme tersebut dengan *firewall routerboard* MikroTik, selain itu penelitian juga akan membandingkan akurasi kedua metode tersebut yang berguna untuk menemukan metode mana yang paling tepat untuk diterapkan dalam mendeteksi *vulnerability scanning*.

1.2. Rumusan Masalah

Dari latar belakang masalah. Dalam membandingkan antara *anomaly based* dan *signature based* dalam mendeteksi *scanning vulnerability* pada website dirumuskan beberapa masalah yang peneliti hadapi yaitu sebagai berikut:

a. Bagaimana cara membangun mekanisme otomatis dalam mendeteksi dan antisipasi *vulnerability scanner* dengan metode *anomaly* dan *signature based*?

b. Bagaimana cara mengimplementasikan mekanisme *anomaly based* dan *signature based* dalam mendeteksi *vulnerability scanner* kedalam *firewall* routerboard MikroTik?

c. Berapa kecepatan mekanisme metode *anomaly based* dalam mendeteksi *scanning vulnerability* pada website?

d. Berapa kecepatan mekanisme metode *signature based* dalam mendeteksi *scanning vulnerability* pada website?

e. Bagaimana kemampuan metode *anomaly-based* dalam mendeteksi *scanning vulnerability* dari dari *tools* yang belum pernah dikenali?

f. Bagaimana kemampuan metode *signature-based* dalam mendeteksi *scanning vulnerability* dari dari *tools* yang belum pernah dikenali?

g. Berapa nilai akurasi,presisi dan sensitivitas dari metode *anomaly-based* dan *signature-based*?

1.3. Batasan Masalah

Untuk mempertajam penelitian ini perlu adanya pembatasan masalah, agar masalah yang diteliti dapat difokuskan pada tujuan penelitian. Batasan masalah dari penelitian ini adalah:

a. Objek penelitian adalah data *traffic packet* yang diperoleh dari proses *sniffing* yang dilakukan MikroTik routerboard tipe RB941-2ND-TC.

b. Proses analisa data *traffic packet* dilakukan pada server dengan sistem operasi CentOS Linux 7 yang terinstal *software* wireshark. Data akan dikonversi dari PCAP ke CSV dengan menggunakan tshark dan diekspor kedalam database MySQL, Dalam menganalisa packet digunakan bahasa pemrograman PHP (*hypertext processor*).

c. Dalam melakukan uji coba perbandingan kecepatan serangan *website vulnerability scanner* penelitian ini menggunakan *software* Acunetix, *sample website* yang diserang adalah website http://sister.umetro.ac.id.

d. Metode *anomaly based* akan menggunakan *threshold* (ambang batas) dalam menentukan sebuah *packet* sebagai serangan atau *packet normal*.

e. *Threshold* didapatkan melalui perhitungan nilai rata-rata *http response* 404 dari setiap *host* dalam rentang waktu satu hari.

f. Metode *signature based* akan menggunakan pola serangan yang didapat dari *software* Acunetix *vulnerability scanner*.

g. Kecepatan dari metode *anomaly based* dan *signature based* akan dilihat dari berapa banyak *request* yang dilakukan Acunetix untuk dapat deteksi oleh *IDS*, semakin sedikit *request* dari Acunetix maka semakin cepat.

h. Kemampuan mendeteksi serangan dari metode *anomaly based* dan *signature based* akan diukur dengan melakukan serangan dengan menggunakan beberapa *tools* untuk melihat apakah metode dapat mendeteksi serangan dari setiap *tool*.

i. Performa dari masi-masing metode akan di dapatkan dari proses perhitungan nilai akurasi, presisi dan sensitivitas.

1.4. Tujuan Penelitian

Tujuan dari penelitian ini adalah:

a. Membangun mekanisme otomatis dalam mendeteksi dan mengantisipasi *vulnerability scanner*.

b. Mengintegrasikan mekanisme otomatis yang telah dibangun kedalam routerboard MikroTik.

c. Mengetahui kecepatan metode *anomaly based* dalam mendeteksi *scanning vulnerability* pada website.

d. Mengetahui kecepatan metode *signature based* dalam mendeteksi *scanning vulnerability* pada website.

e. Mengetahui kemampuan metode *anomaly based* dan *signature based* dalam mendeteksi berbagai *tools* yang digunakan dalam melakukan serangan *vulnerability scanner*.

f. Mengetahui nilai akurasi, presisi dan sensitivitas dari masing-masing metode.

1.5. Manfaat Penelitian

Manfaat penelitian ini adalah menemukan metode terbaik dalam mendeteksi aktivitas *scanning vulnerability* pada website yang bertujuan untuk mengamankan *web server* dari aktivitas *website vulnerability scanning*, yang dilakukan oleh orang yang tidak berwenang.

BAB II

TINJAUAN PUSTAKA

2.1. Tinjauan Pustaka

Pengembangan beberapa metode dalam mendeteksi intrusi pada jaringan terbagi menjadi dua yaitu *anomaly based* dan *signature based* (Maseer, et al., 2021). Beberapa penelitian dapat menggambarkan kelebihan dan kekurangan kedua metode tersebut dalam mendeteksi intrusi pada jaringan, penelitian yang dilakukan Fadhlurrohman pada tahun 2021 menggunakan metode *anomaly based* dalam mendeteksi serangan siber dengan menggunakan algoritma *decision tree*, IDS (*intrusion detection system*) akan mendeteksi sebuah serangan apabila terjadi perubahan lalu lintas paket melebihi *threshold* namun dengan metode ini masih memungkinkan terjadinya *false detection*, *IDS* akan mendeteksi serangan apabila terjadi lalu lintas paket perdetiknya mencapai 350 dan memberikan notifikasi untuk di lakukan pemeriksaan dan antisipasi oleh *administrator*.

Penelitian selanjutnya yang dilakukan oleh Haryani pada tahun 2021 memanfaatkan *firewall* MikroTik dalam mendeteksi serangan *ping flood* dengan memblokir paket ICMP (*internet control message protocol*) yang melebihi *treshold*, apabila pengguna melakukan *request ICMP* lebih dari 10 kali maka request berikutnya akan diblokir. Penelitian serupa juga pernah dilakukan oleh Syahputra pada tahun 2020 yang memanfaatkan MikroTik dalam mengamankan jaringan dari serangan DDOS (*distribution denial of service*) dengan membatasi paket ICMP dan berhasil menjamin stabilitas koneksi internet pada jaringan.

Metode-metode diatas adalah metode yang digunakan dalam mendeteksi anomaly pada sebuah jaringan dengan menandai sebuah paket yang tidak normal sebagai sebuah serangan.

Penelitian tentang *signature base* dalam mendeteksi serangan juga sangat beragam, Hidayat pada tahun 2018 berhasil menandai pola serangan winboxpoc yang selalu mengakses *file* user.dat (berisi *username* dan *password*) pada *router* MikroTik dan memblokir setiap koneksi yang mengakses *file* tersebut, sehingga serangan akses ilegal ke MikroTik berhasil digagalkan, *firewall* pada *router* MikroTik dimanfaatkan untuk memblokir akses *user* ke *file* user.dat sehingga serangan dapat dihentikan.

Pada tahun 2022 Widodo memanfaatkan SNORT (IDS *signature based*) dalam mendeteksi intrusi dan berhasil, SNORT memberikan notifikasi apabila terdapat aktifitas yang melanggar rule, SNORT Menganalisa paket yang telah didapatkan dan mencocokan paket tersebut dengan rule, apabila paket melanggar rule yang telah ada, maka SNORT akan memberikan notifikasi kepada *administrator* untuk segera menindaklanjuti notifikasi tersebut.

Penelitian Sau pada tahun 2021 yang memadukan SNORT dengan RITA (*real intelligence threat analytics*) yang memanfaatkan algoritma statistik dan *K-means clustering*, dalam mendeteksi aktifitas komputer *zombie* yang mengirimkan sinyal secara *periodic* ke komputer *server command and control* milik penyerang (*beaconing*), *log* yang dihasilkan oleh SNORT di konversi menjadi JSON yang akan dianalisis kembali oleh RITA, hasilnya RITA mampu mendeteksi serangan yang tidak mampu dideteksi oleh SNORT.

Sedangkan penelitian yang dilakukan Iman pada 2019 memadukan algoritma RIPPER dan K-nearest neighbor, menurut penelitian tersebut dengan menggabungkan metode deteksi yang berbeda dapat menghasilkan IDS yang memiliki akurasi yang baik, berdasarkan penelitian tersebut *IDS* yang dibangun dengan akurasi yang sangat tinggi yaitu 99,89522%.

Penelitian di atas membangun *IDS* untuk mencegah berbagai serangan yang terjadi pada sebuah *server* maupun jaringan berupa *malware* maupun *DDOS* dimana serangan tersebut sudah masuk ke tahap eksploitasi, berbeda dengan penelitian sebelumnya penelitian yang dilakukan Subandi pada 2021 menjelaskan bahwa *scanning vulnerability* digunakan pada tahap awal sebelum *attacker* melakukan serangan, hal itu dibuktikan dengan hasil dari *tool vulnerability scanner* yaitu Acunetix dapat menemukan celah yang dapat di eksploitasi oleh *attacker*. Dengan menemukan celah yang ada attacker dapat melakukan serangan ke tahap eksploitasi sehingga perlu adanya pencegahan *attacker* untuk mendapatkan *vulnerability* pada *website*.

Senada dengan penelitian yang dilakukan Subandi, Huang pada 2022 menjelaskan *WEB Vulnerability Scanner (WVs)* dapat digunakan untuk menemukan celah dari *website* yang di *scan* dengan cara menganalisa *response* yang diberikan oleh *web server* ketika dilakukan *scanning*. *WVs* baik *opensource* maupun yang berbayar melakukan *scanning* dengan *payload* yang telah di simpan di dalam *database* yang berisi kerentanan yang telah diketahui, sehingga memungkinkan mendapatkan *response error 404* apabila request yang dilakukan tidak ada pada webserver yang diserang.

Dari penelitian yang dilakukan Huang tersebut meyakinkan apabila kerentanan di temukan oleh orang yang tidak bertanggung jawab akan sangat berbahaya, karena penyerang kemungkinan akan melakukan eksploitasi terhadap kerentanan yang telah di dapatkan.

2.2. Keaslian Penelitian

Tabel 2.1. Matriks literatur review dan posisi penelitian
Perbandingan Metode Anomaly Based dan Signature Based Dalam Pendeteksian Scanning Vulnerability Pada Website

No	Judul	Peneliti, Media Publikasi dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
1	Analisis Kinerja Intrusion Detection System pada Deteksi Anomali dengan Metode Decision Tree Terhadap Serangan Siber.	Fadhlurrohman, M., Muliawati, A., & Hananto, B. , Jurnal Ilmu Komputer dan Agri-Informatika, 8 (2), 90-94. , 2021	Menganalisa kinerja IDS deteksi anomali based dengan metode decision tree terhadap serangan siber.	Dalam mendeteksi sebuah anomali, IDS membandingkan sebuah traffic pada sebuah server dalam kondisi normal dengan kondisi saat terjadinya serangan. Jika perubahaan lalu lintas pada paket perdetiknya mencapai 350 paket, IDS akan menganggap itu sebagai anomali dan mendeteksi sebagai suatu serangan. IDS dapat meyebabkan false alarm sehingga IDS mendeteksi suatu lalu lintas yang bukan serangan sebagai sebuah serangan.	Kelemahan: data yang diambil tidak memiliki waktu serangan (tanggal) jika diimpelemntasikan maka akan banyak false positive jika data terus bertambah dari hari ke hari. Saran: data diberi tanggal, agar packet yang di seleksi adalah packet dalam waktu yang sama.	1. Dalam penelitian ini tidak menggunakan algoritma decision tree, melainkan menggunakan threshold dalam mendeteksi anomaly. 2. Penelitian ini akan melakukan perbandingan antara anomaly based dan signature based dalam mendeteksi scanning vulnerability berdasarkan keakuratan dan kecepatan dari hasil pengukuran yang akan dilakukan.

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
2	Manajemen Pada Jaringan MikroTik Menggunakan Metode Hierarchical Token Bucket (HTB) dan Keamanan Firewall Intrusion Detection System (IDS).	Haryani, P., & Raharjo, S. , Jurnal Jarkom, 9(1), 1-9. , 2021	Menangkal serangan ping flood dan memanajemen dengan HTB guna mengamankan router dan menjaga stabilitas bandwith.	1. Hasil dari pengelolahan manajemen bandwidth pada jaringan yang ada di Balai Desa Semali menggunakan metode Hierarchical Token Bucket (HTB) dinilai lebih efektif dalam membagi bandwidth secara adil dan merata kepada client yang diprioritaskan. 2. pembatasan ping flood dan port knocking berhasil menangkal serangan dengan memblokir IP penyerang.	Kelemahan : pembatasan packet ICMP seharusnya tidak dikenakan kepada IP yang sah dalam menggunakan protocol ICMP, sebab protocol ICMP terkadang digunakan untuk melakukan *testing* koneksi dari client ke server, dikhawatirkan akan terjadi false positive ketika admin melakukan uji koneksi ke router menggunakan ICMP protocol. Saran : harus diberikan *whitelist* untuk menandai IP yang berhak melakukan ping misalnya dengan mekanisme port knocking.	1. Penelitian ini akan melakukan perbandingan antara anomaly based dan signature based dalam mendeteksi scanning vulnerability berdasarkan keakuratan dan kecepatan dari hasil pengukuran yang akan dilakukan.

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
3	Analisa Dan Problem Solving Keamanan Router MikroTik Rb750Ra Dan Rb750Gr3 Dengan Metode Penetration *Testing* (Studi Kasus: Warnet Aulia. Net, Tanjung Harapan Lampung Timur).	Hidayat, A., & Saputra, I. P. , Jurnal RESISTOR (Rekayasa Sistem Komputer), 1(2), 118-124. , 2018	Menangkal serangan winboxpoc guna mengamankan router MikroTik dari pencurian akun oleh orang yang tidak bertanggung jawab.	Dalam hal menangani serangan, khususnya winboxpoc, router telah diatur untuk mengenali serangan tersebut, dimana serangan winboxpoc selalu menargetkan file user.dat yang berisi username dan password yang digunakan untuk mengakses server.	Kelemahan: serangan tidak terdokumentasi, sebaiknya diberikan log penyerangan sebelum packet diblokir, dalam penelitian ini hanya memblokir packet tanpa mengetahui siapa yang melakukan serangan tersebut. Saran: akan lebih baik pengguna yang mencoba mengeksploitasi user.dat dimasukan kedalam address-list dan ditandai sebagai pelaku serangan, setelah ditandai sebagai penyerang, firewall akan memblokir IP dari penyerang yang telah ditandai sebelumnya.	1. Penelitian ini akan melakukan perbandingan antara anomaly based dan signature based dalam mendeteksi scanning vulnerability berdasarkan keakuratan dan kecepatan dari hasil pengukuran yang akan dilakukan.

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
4	Pemanfaatan MikroTik Router Board Sebagai Pengaman Serangan DDOS Menggunakan Metode IDS.	Syahputra, W., Diansyah, T. M., & Liza, R. , Seminar Nasional Teknologi Informasi & Komunikasi (Vol. 1, No. 1, pp. 492-499). , 2020	Mendeteksi dan mengantisipasi ping flood untuk mengamankan router dari serangan ping flood dengan membatasi jumlah koneksi ICMP yang diijinkan dalam satu waktu.	IDS yang dibangun mampu mendeteksi dan mengamankan router dari serangan DDOS, dan menjamin kestabilan koneksi internet pada client.	Kelemahan : pembatasan packet ICMP seharusnya tidak dikenakan kepada IP yang sah dalam menggunakan protocol ICMP, sebab protocol ICMP terkadang digunakan untuk melakukan *testing* koneksi dari client ke server, dikhawatirkan akan terjadi false positive ketika admin melakukan uji koneksi ke router menggunakan ICMP protocol. Saran : harus diberikan *whitelist* untuk menandai IP yang berhak melakukan ping misalnya dengan mekanisme port knocking.	1. Penelitian ini akan melakukan perbandingan antara anomaly based dan signature based dalam mendeteksi scanning vulnerability berdasarkan keakuratan dan kecepatan dari hasil pengukuran yang akan dilakukan.

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
5	Analisis Pengguna an Hasil Deteksi IDS Snort pada *Tools* RITA dalam Mendetek si Aktivitas Beacon.	Sau, W. M. T., & Siswantyo, S. , Info Kripto, 15(2), 97-104. , 2021	Memanfaatkan SNORT dan RITA dalam mendeteksi aktivitas mengirimkan sinyal secara periodic dari komputer zombie ke server command & control milik penyerang (beaconing), yang tujuanya untuk menunggu arahan dalam melakukan penyerangan.	RITA (real intelligence threat analytics) mampu menganalisis paket dari SNORT dan dapat mendeteksi aktifitas beaconing.	Kelemahan: penelitian ini menyebutkan SNORT tidak mendeteksi serangan yang di deteksi RITA, namun penulis tidak menjelaskan apakah serangan yang terdeteksi RITA tersebut true positive atau false positive. Saran: Dalam hal sample data bisa didapat melalui percobaan serangan secara real, agar dapat mengetahui secara pasti apakah RITA mendeteksi serangan dengan hasil true positive.	1. Penelitian ini akan melakukan perbandingan antara anomaly based dan signature based dalam mendeteksi scanning vulnerability berdasarkan keakuratan dan kecepatan dari hasil pengukuran yang akan dilakukan.

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
6	Pemanfaatan Network Forensic Investigation Framework untuk Mengidentifikasi Serangan Jaringan Melalui Intrusion Detection System (IDS).	Widodo, T., & Aji, A. S. , JISKA (Jurnal Informatika Sunan Kalijaga), 7(1), 46-55. , 2022	Mengimplementasikan SNORT dengan bantuan Network Forensic Investigation Framework dalam prosedur mendeteksi serangan.	Network Forensic Investigasi framerwork memudahkan proses investigasi ketika terjadi serangan. IDS yang dibangun dengan menggunakan SNORT dapat memberikan notifikasi apabila terjadi aktivitas yang melanggar rule.	Kelemahan: tidak adanya uji akurasi terhadap serangan. pada tahap investigasi penulis mengungkapkan terdapat IP yang mengirim packet sebesar 206MB, didalam jaringan yang kompleks, terdapat router yang memiliki fungsi NAT (network address translation) sangat mungkin sebuah IP (yaitu ip router) mengirimkan paket sebesar itu.	1. Penelitian ini akan melakukan perbandingan antara anomaly based dan signature based dalam mendeteksi scanning vulnerability berdasarkan keakuratan dan kecepatan dari hasil pengukuran yang akan dilakukan.

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
7	Deteksi Anomali Jaringan Menggunakan Hybrid Algorithm.	Imam, R. M., Sukarno, P., & Nugroho, M. A., eProceedings of Engineering, 6(2). , 2019	Menggabungkan IDS anomaly based dan signature based dengan algoritma RIPPER dan algoritma K-Nearest Neighbour dalam mendeteksi serangan.	Algoritma yang dibangun dalam mendeteksi yaitu dengan algortima RIPPER dan K-NN berhasil mendeteksi serangan dengan mengklasifikasikan serangan berdasarkan hasil analisis dari Algoritma RIPPER. Akuasi terbaik ketika *testing* yaitu 99.89522%.	Kelemahan: penelitian hanya menentukan sebuah packet tersebut anomaly atau bukan sehingga tidak diketahui jenis serangan apa yang sedang terjadi. Saran: untuk dapat diterapkan sebagai IDS, pendeteksian harus dapat mengenali jenis serangan, agar *administrator* jaringan dapat mengetahui langkah apa yang harus dilakukan dalam menangani anomaly tersebut.	1. Dalam penelitian ini tidak menggunakan algoritma RIPPER dan K-Nearest Neighbour dalam mendeteksi anomaly. 2. Penelitian ini akan melakukan perbandingan antara anomaly based dan signature based dalam mendeteksi scanning vulnerability berdasarkan keakuratan dan kecepatan dari hasil pengukuran yang akan dilakukan.

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
8	Analisa Serangan Vulnerabilities Terhadap Server Selama Periode WFH di Masa Pandemi Covid-19 Sebagai Prosedur Mitigasi	Subandi, K., & Sugara, V. I. Prosiding Semnastek. 2021	Melakukan penetrasi *testing* kepada web server untuk mengetahui kelemahan dan memperbaiki kelemahan tersebut sebelum diketahui oleh pihak luar.	Penetrasi *testing* dilakukan dengan mencari celah dari website dengan menggunakan Acunetix vulnerability scanner, menghasilkan temuan bug pada website. Temuan tersebut ditindak lanjuti dengan melakukan perbaikan pada celah yang ditemukan.	Pada penelitian ini menggunakan Acunetix sebagai *tool* guna menemukan celah pada website, hal ini tidak menutup kemungkinan ada attacker yang telah menemukan bug terlebih dahulu dari administrator, sehingga perlu adanya penelitian untuk mencegah vulnerability scanner.	Pada penelitian ini menggunakan Acunetix sebagai *tool* yang digunakan untuk menemukan bug, sedangkan penelitian yang akan dilakukan akan menangkal Acunetix untuk mencegah bug ter ekspos dan diketahui oleh pihak luar.

Tabel 2.1. (Lanjutan)

No	Judul	Peneliti, Media Publikasi dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
9	Adaptive Entry Point Discovery for Web Vulnerability Scanning	Huang, H. C., Zhang, Z. K., Chen, C. K., Hong, W. D., Jao, J. C., & Shieh, S. *Journal of Information Science & Engineering*, *38*(1). 2022	Meningkatkan efisiensi web vulnerability scanning dan ketepatan dalam melakukan vulnerability scanning. *Tool* yang dibangun mencari kerentanan berdasarkan fungsi dari suatu halaman.	Dari hasil uji coba yang dilakukan peneliti berhasil mengembangkan *tool* VulCraw yaitu *tool* yang digunakan untuk menemukan celah keamana pada suatu situs, *tool* yang dikembangkan lebih efektif dalam menemukan celah kemanan di bandingkan dengan *tool* yang mencari kerentanan berdasarkan response dari suatu konten halaman.	Kelemahan dari penelitian ini yang meng analisis sebuah halaman berdasarkan fungsinya masih dapat diatasi apabila sebuah halaman mengharuskan pengguna untuk memasukan captcha. Namun dari penelitian ini juga dapat disimpulkan bahwasanya kerentanan website harus di jaga untuk tidak diketahui oleh orang yang tidak berhak melakukanya.	Penelitian berfokus pada peninkatan efektifititas *tool* di dalam menemukan celah suatu website, sedangkan penelitian yang di ajukan akan mencegah terjadinya pengumpulan vulnerability scanning oleh orang yang tidak berhak melakukan.

2.3. Landasan Teori

Terdapat beberapa landasan teori yang dibutuhkan dalam mendukung penelitian ini, mulai dari keamanan komputer, *web vulnerability scanner*, *sniffing*, serta *firewall* dan MikroTik.

2.3.1 Keamanan komputer

Keamanan komputer adalah suatu studi yang dilakukan untuk mengamankan suatu sistem dengan mempelajari pola serangan guna menangkal serangan terhadap komputer (Muhammad, 2021). Penggunaan internet dalam hal komersial menyebabkan perlunya keamanan dalam pertukaran *data* dari akses yang tidak di inginkan seperti *worm* dan *virus* (Gollmann, 2010). Menurut Brinkley pada tahun 1995 The *European community* mendefinisikan keamanan teknologi informasi meliputi *confidentiality* (melindungi informasi dari pihak yang tidak berwenang), *integrity* (mencegah modifikasi terhadap informasi), *availability* (mencegah informasi tetap tersedia dan aman dari gangguan yang tidak diinginkan).

2.3.2 Vulnerability scanning

Vulnerability merupakan suatu kelemahan yang menjadi ancaman nilai *integrity, confidentiality dan availability* dari suatu aset (Subandi., et al., 2021). *Vulnerability scanning* merupakan sebuah metode yang dilakukan untuk mengumpulkan informasi yang berguna untuk menemukan kelemahan dari sebuah *website*, dengan menggunakan *tool* yang telah di konfigurasi operator hanya menunggu waktu untuk mendapatkan hasil dari *scanning* tersebut (Chen., et al., 2021). *Web vulnerability scanner* (WVSs) adalah *tool* yang bekerja secara

otomatis, digunakan untuk mencari celah keamanan, cara kerja dari WVSs yaitu dengan melakukan *generate malicious code* dan melakukan injeksi data ke dalam *website* target. Dengan melakukan observasi terhadap respon dari *website* tersebut celah keamanan dapat diketahui (Huang., et al., 2022).

2.3.3 Intrusion detection system

Intrusion detetction system (IDS) adalah sebuah alat keamanan, yang bertindak layaknya seperti *software antivirus, firewall* dan akses kontrol, yang di fungsikan untuk mengamankan informasi dan komunikasi (Garcia-Teodoro., et al., 2009).

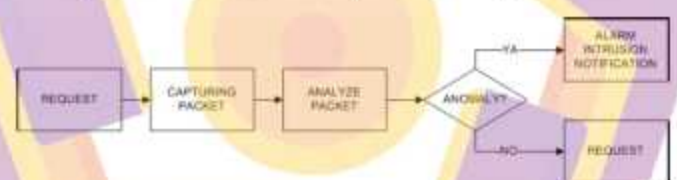
IDS mendeteksi intrusi berdasarkan pola (*signature*), analsisis perilaku (*behavior*) dan aktifitas *malicious* dengan memberikan alarm apabila terjadi sesuatu yang tidak biasa terjadi *(anomaly)* di dalam sebuah jaringan, sehingga pakar keamanan dapat mencegah serangan yang menyebabkan kerusakan (Thakkar., et al., 2021). Maseer pada tahun 2021 mengatakan bahwa terdapat dua *type* IDS yang populer, yaitu *network intrusion detetction system* (NIDS) dan *host based intrusion detection system* (HIDS), IDS mengaplikasikan dua metode dalam mendeteksi serangan yaitu *signature based* dan *anomaly based*.

2.3.4 Anomaly based

Anomaly based adalah metode dalam mendeteksi sebuah intrusi atau serangan, dengan cara menganalisis jumlah *request*, analisis panjang *request* dan menganalisis frekuensi metode *request* yang digunakan (Tekerek., et al., 2014). Keunggulan *anomaly based* adalah dapat mendeteksi serangan yang tidak diketahui sebelumnya, namun perlu algoritma yang baik untuk mendapatkan

hasil yang baik dalam menganalisis paket sehingga hasil menjadi akurat (Otoum., et al., 2021). Imam pada 2019 memberi penjelasan bahwa *anomaly* didapat dari membandingkan kebiasaan seseorang yang melakukan akses internet pada jam kerja dari siang hingga sore hari dan telah diamati selama beberapa waktu dan sistem menetapkan itu sebagai perilaku normal dari pengguna, namun ketika pengguna tersebut akan mengerjakan sesuatu dan mengakses jaringan di luar jam sistem menganggap hal tersebut sebagai sebuah *anomaly*, sehingga sistem mengalami *false positive*. *Anomaly-based* memiliki kelamahan tingginya *false positive* dikarenakan perilaku pengguna yang sangat variatif.

Cara kerja metode *anomaly based* dapat dilihat dari gambar 2.1 betikut ini.



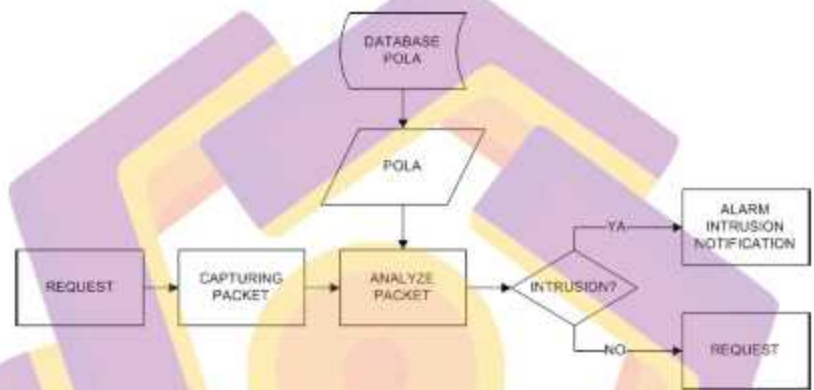
Gambar 2.1. Anomaly based method

2.3.5 Signature based

Signature based adalah sebuah metode deteksi intrusi yang memanfaatkan pola *(pattern)* dari serangan yang telah di simpan di dalam *database pola* (*data* yang berisi pola dari serangan) , apabila pola serangan telah disimpan di dalam *database pola* dan terdapat paket yang cocok dengan pola tersebut maka paket akan di tandai sebagai intrusi (Kumar., 2021). Kelemahan dari *signature base* adalah tidak dapat mendeteksi serangan yang tidak diketahui polanya yang biasa disebut *unknown attack* (Otoum., et al., 2021). Menurut

Imam pada 2019 kelemahan *signature-based* adalah tidak dapat mendeteksi serangan yang belum di petakan atau yang belum pernah di kenal sama sekali, sehingga *signature-based* memerlukan pembaharuan atau *update* terhadap serangan yang belum di petakan ke dalam *database*.

Cara kerja metode signature based dapa dilihat pada gambar 2.2 berikut ini.



Gambar 2.2. signature based method

2.3.6 *Packet sniffing*

Packet sniffing merupakan sebuah proses dalam *capturing packet* yang ada pada *network flow* yang berguna menganalisa kejadian yang ada di dalam jaringan (Dodiya., et al., 2022). Dengan memanfaatkan *data* yang didapat melalui proses *sniffing packet administrator* jaringan dapat mendeteksi *packet* yang tidak benar sehingga pertukaran *data* dapat diamankan (Saroha., 2020). Kumar pada tahun 2021 menjelaskan paket yang di dapatkan dari proses *sniffing* dapat membantu administrator dalam menganalisis guna mendeteksi paket yang tidak normal dan menggunakannya dalam mencari masalah dan memastikan jaringan menjadi aman.

2.3.7 Firewall

Firewall adalah sistem yang digunakan dalam mengamankan informasi dan menghindari terjadi *data loss*, mengamankan jaringan serta aplikasi web (Tekerek., et al., 2014). *Firewall* merupakan perangkat perlindungan dasar untuk mengamankan jaringan dan melindungi perangkat yang terhubung ke internet (Liang., et al., 2022). Efektifitas *firewall* ditentukan oleh lingkungan kerja *firewall* itu sendiri serta tipikal data yang akan dilindungi. *firewall* di tuntut untuk dapat melindungi sistem baik dalam maupun luar jaringannya (Anwar., et al., 2021).

Dua model kebijakan *firewall* dalam mengatur keamanannya yaitu *positive* dan *negative*, *positive security* model hanya mengijinkan trafik tertentu dan memblokir semua trafik, sedangkan *negative security* model adalah memperbolehkan semua trafik dan memblokir trafik tertentu (Clincy., et al., 2018).

Cara kerja *firewall* dapat dilihat pada gambar 2.3 berikut ini.

Gambar 2.3 Cara kerja *firewall*

2.3.8 MikroTik

MikroTik adalah perusahaan yang berasal dari Latvia yang memiliki beberapa produk yang berupa perangkat keras dan perangkat lunak guna memanajemen koneksi internet, produk dari MikroTik antara lain adalah

MikroTik Routerboard yang merupakan pengakat keras *(hardware)* yang berisi MikroTik RouterOS dan MikroTik RouterOS yang merupakan sistem operasi khusus untuk perangkat jaringan komputer(Saputra et al., 2021).

Gambar 2.4 terminal MikroTik RouterOS

MikroTik RouterOS dapat digunakan sebagai *firewall* yang berfungsi mengatur konten apa saja yang di akses serta mencegah penyalahgunaan jaringan dari hal yang melanggar keamanan jaringan.

2.3.9 Wireshark

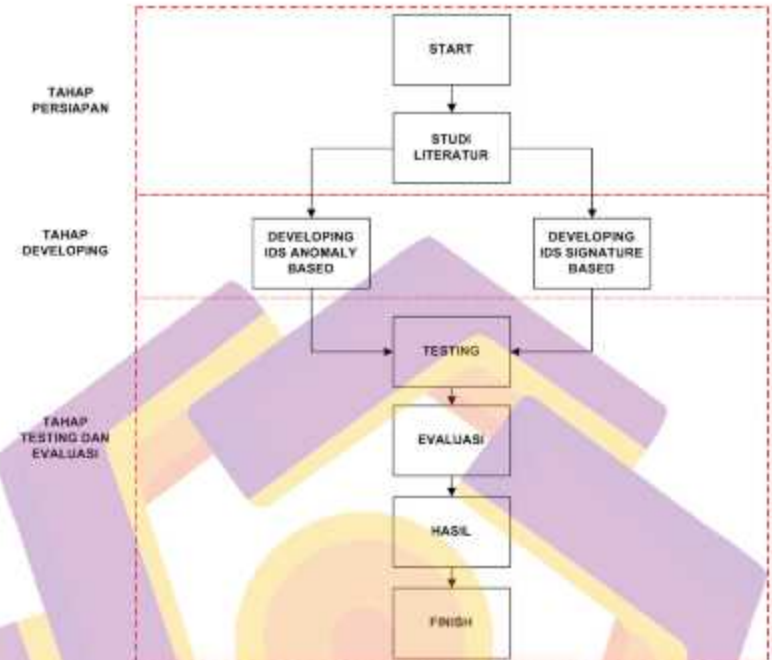
Wireshark merupakan alat yang digunakan untuk melakukan *sniffing data* pada jaringan yang digunakan untuk proses analisis paket yang ada pada jaringan, cara kerja wireshark dengan menangkap data pada perangkat jaringan dan menangkapnya dalam bentuk *frames, segments,* dan *packets*, dapat memahami dari mana paket tersebut berasal dan dimana paket tersebut berhenti, paket dapat diamati secara langsung melalui wireshark maupun di simpan untuk di amati kemudian hari ke PCAP file (Bullock, 2017).

BAB III

METODE PENELITIAN

3.1. Metode Penelitian

Metode penelitian dari penelitian ini terdiri dari beberapa tahapan, dimulai dari studi pustaka, perancangan dan pengembangan *IDS* dengan metode yang berbeda yaitu *anomaly based* dan *signature based*, dilanjutkan ke tahap eksperimen dan evaluasi, penelitian ini akan menghasilkan mekanisme otomatis dalam mendeteksi serangan *vulnerability scanner* dengan metode *anomaly-based* dan *signature-based* serta membandingkan kedua mekanisme tersebut dari segi kecepatan deteksi, kemampuan mekanisme dalam mendeteksi berbagai *tools* dan menghitung performa masing-masing metode dengan menghitung nilai akurasi, presisi dan sensitivitas dari setiap metode. Metode penelitian sendiri digunakan untuk memetakan alur penelitian dari awal hingga akhir hingga maksud dan tujuan penelitian ini tercapai. Adapun tahapan penelitian didalam penelitian ini dapat dilihat melalui dilihat pada Gambar 3.1 berikut ini.



Gambar 3.1 Alur penelitian

Proses *developing* atau pengembangan dilakukan dengan membangun IDS dengan memanfaatkan dua buah metode, metode yang pertama *anomaly based* dengan menandai paket yang memiliki *http response* 404 *(not found)* yang melebihi ambang *(threshold)* sebagai sebuah serangan, sedangkan IDS kedua akan menggunakan metode *signature based* yaitu dengan menandai paket yang melakukan *request* ke sistem Acunetix sebagai sebuah serangan. Paket yang akan dianalisis oleh *IDS* yang dibangun adalah paket yang didapat dari proses *sniffing* secara langsung di *router* MikroTik yang terhubung ke website http://sister.ummetro.ac.id.

Dalam tahap *testing* akan dilakukan serangan kepada website dengan

menggunakan *tool* Acunetix, dari hasil *testing* tersebut akan di hitung kecepatan dari masing-masing mekanisme yaitu *anomaly-based* dan *signature-based*, selain itu IDS yang dibangun akan di coba untuk mencegah serangan dari berbagai *tools* website akan diserang menggunakan beberapa *tools scanning* yang berbeda sehingga dapat menyimpulkan kemampuan dari masing-masing metode di dalam mendeteksi *vulnerability scanner*, dalam hal pengukuran performa akan di lakukan dengan melabeli data set secara manual untuk menentukan sebuah serangan atau bukan serangan, mekanisme akan di uji untuk menemukan nilai *true positive* (TP) yaitu kemampuan mendeteksi serangan dengan benar, serta menemukan nilai *true negative* (TN) kemampuan mendeteksi bukan serangan dengan benar, hal tersebut bertujuan untuk menemukan nilai akurasi, presisi dan sensitivitas dari masing-masing metode.

Pada tahap evaluasi akan melakukan evaluasi terhadap hasil *testing* yang sudah di lakukan, sehingga penelitian ini akan menghasilkan mekanisme mana yang paling efektif dan tepat di dalam mendeteksi serangan *vulnerability scanning*.

3.2. Studi Literatur

Pada tahap studi literatur penulis mencari pengertian dan penjelasan teoritis tentang *IDS* dengan metode *anomaly based* dan *signature based*, cara kerja dari kedua metode tersebut, mempelajari tentang *vulnerability scanner,* jenis-jenis *vulnerability scanner* dan cara kerja *vulnerability scanner*. Proses studi literatur digunakan untuk mempermudah penelitian dalam mengembangkan IDS yang dijadikan perbandingan didalam penelitian ini.

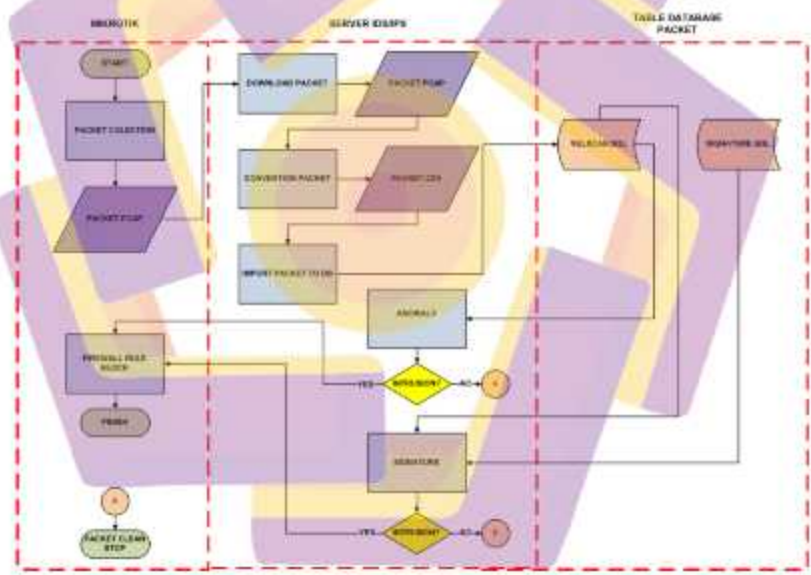
3.3. Eksperimen Perancangan IDS

Pada penelitian ini akan melakukan eksperimen dengan membuat dua *IDS* yang dapat mendeteksi sebuah aktivitas *scanning vulnerability* pada sebuah website yaitu https://sister.ummetro.ac.id, *IDS* pertama yang dibangun menggunakan *anomaly based* dengan menentukan sebuah *threshold,* menurut Imam pada tahun 2019 menjelaskan bahwa *anomaly* di dapatkan dengan membandingkan kebiasaan normal pengguna. *Threshold* mewakili suatu kejadian normal, untuk itu dalam menentukan sebuah *threshold* dilakukan dengan menggunakan metode statistik yaitu menghitung nilai rata-rata dari *http response error* 404 (halaman tidak ditemukan) dalam satu hari, nilai rata-rata inilah yang akan di jadikan dasar dalam menentukan sebuah *anomaly*, merujuk Huang tahun 2022 bahwa *web vulnerability scanner* melakukan *request* dengan *payload* yang telah tersimpan di *database* yang berisi kerentanan yang telah di ketahui sehingga efek samping yang di dapatkan adalah *error 404* apabila *payload* yang di minta tidak ditemukan. Sehingga apabila terdapat *IP* yang menghasilkan *error* 404 melebihi *threshold* akan dianggap sebuah serangan.

IDS kedua akan dibangun dengan metode *signature based,* menurut Kumar tahun 2021 menjelaskan untuk menentukan sebuah serangan, *signature-base* melakukan pencocokan *packet* dengan pola yang telah di simpan di dalam *database*, apabila terdapat *packet* yang cocok dengan pola serangan maka *packet* tersebut akan dianggap sebagai serangan. *Signature* (pola) akan didapatkan dari proses pengamatan aktifitas Acunetix *vulnerability scanner* dengan menggunakan Wireshark, cara ini merujuk pada penelitian Laabid tahun 2021

yang melakukan analisis pada *packet flow* untuk mendapatkan trafik dari *malware* dengan menjalankan *malware* pada sistem yang di isolasi. Setelah pola serangan di dapatkan pola akan di simpan ke dalam tabel signature. apabila terdapat *packet* pada table vulscan yang memiliki pola yang sama dengan pola yang tersimpan pada tabel signature, maka *packet* akan dianggap sebagai aktivitas *scanning vulnerability*.

Flowchart dari IDS *anomaly-based dan signature-based* dapat dilihat pada Gambar 3.2 berikut ini.



Gambar 3.2 *Flowchart* IDS *anomaly dan signature based*

Setelah kedua *IDS* di bangun selanjutnya kedua *IDS* akan dibandingkan dalam hal kecepatan mendeteksi serangan Acunetix dan kemampuan dalam mendeteksi berbagai *tools* yang biasa digunakan untuk *scanning vulnerability*.

3.4. Eksperimen dan evaluasi

Pada tahapan eksperimen dan evaluasi akan dilakukan proses uji coba serangan *vulnerability scanning* dengan menggunakan *tool* Acunetix, hasil uji coba tersebut akan di evaluasi untuk menentukan perbandingan kecepatan dari kedua metode dalam mendeteksi serangan Acunetix.

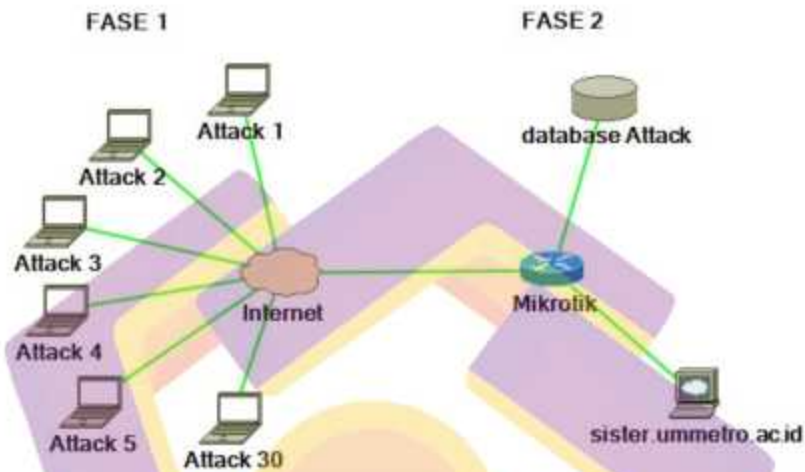
Selain itu kedua metode akan di uji menggunakan berbagai *web vulnerability scanner tools* untuk membandingkan kemampuan dalam mendeteksi berbagai *tools*. Dari hasil evaluasi akan di dapatkan kesimpulan metode mana yang dapat melakukan antisipasi serangan secara cepat. Selain kecepatan pada proses evaluasi juga akan mengukur kemampuan dari setiap mekanisme di dalam mendeteksi berbagai *tools* yang digunakan dalam melakukan serangan *vulnerability scanner*.

Proses eksperimen dan evaluasi dapat dilihat pada Gambar 3.3 berikut ini.

Gambar 3.3. Proses eksperimen dan evaluasi metode

Selain dari kecepatan dan kemampuan dalam mendeteksi berbagai serangan, kedua metode akan di ukur nilai akurasi, presisi dan sensitivitas guna

menemukan nilai performa dari masing-masing metode. Berikut gambar 3.4 Menjelaskan proses pengumpulan data guna menguji masing-masing metode.



Gambar 3.4. Proses pengumpulan data uji

Proses pengumpulan data uji di lakukan dengan mencatat alamat *Internet Protocol (IP Addresses)* dari masing-masing komputer yang melakukan uji coba serangan, uji coba serangan di lakukan sebanyak 30 kali dengan menggunakan Acunetix *web vulnerability tool scanner*, data akan di labeli serangan apabila menggunakan *IP Adresses* yang telah di catat sebelumnya dan mengakses URL yang identik dengan URL Acunetix. Dari proses *testing* yang dilakukan akan di evaluasi sehingga mendapatkan hasil berupa nilai akurasi, presisi dan sensivitas dari seiap model, sehingga dapat di bangun metode yang tepat untuk melakukan deteksi dan antisipasi serangan *web vulnerability scanner*. Proses perhitungan performa dari setiap metode dapat di hitung dengan menggunakan beberapa

formula yang ada di bawah ini.

$$\text{Akurasi} = (TP + TN) / (TP+FP+FN+TN) \quad (1)$$

$$\text{Presisi} = (TP) / (TP + FP) \quad (2)$$

$$\text{Sensitivitas} = TP / (TP + FN) \quad (3)$$

Keterangan :

TP= Jumlah seleksi serangan dan benar serangan

TN= Jumlah seleksi bukan serangan dan benar bukan serangan

FP = Jumlah seleksi serangan dan merupakan bukan serangan

TN= Jumlah seleksi bukan serangan dan merupakan bukan serangan

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

4.1. *Developing IDS Anomaly-based*

Pada tahap *developing* IDS *anomaly-based* dilakukan setelah mengetahui cara kerja *anomaly-based*, dimana IDS *anomaly-based* akan mendeteksi sebuah serangan dari perilaku pengguna yang didapatkan dari proses *sniffing* yang dilakukan secara otomatis oleh router MikroTik. Berikut ini adalah persamaan dalam menentukan anomaly:

$$T = x / y$$

Keterangan:

T = Threshold
x = Jumlah error 404 dalam satu hari
y = Jumlah host yang melakukan error 404 dalam satu hari

Berdasarkan pendapat Imam pada tahun 2019 profil yang digunakan untuk perbandingan di dapatkan dari proses perhitungan *statistical* dan *historical network traffic data*. Proses penentuan profil *(threshold)* dilakukan pada hari sebelumnya, sehingga akan dibandingkan dengan current activity, sehingga apabila terdapat *IP Address* yang melakukan *error* 404 melebihi *threshold* akan dianggap sebagai sebuah *anomaly*.

4.1.1. Proses *Packet Collection Anomaly-based*

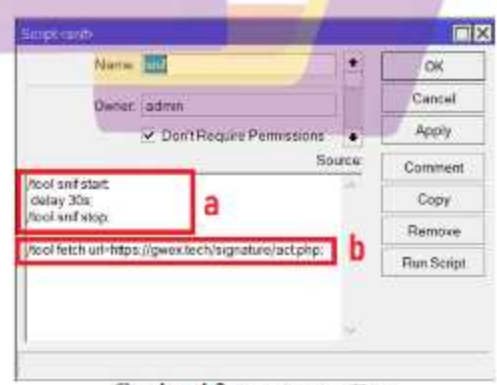
Proses *packet collection* adalah proses mengumpulkan *packet* yang melalui MikroTik baik dari luar jaringan maupun dalam jaringan itu sendiri. Proses ini akan dilakukan secara otomatis menggunakan fitur *scripting* yang ada pada *router* MikroTik. Berikut ini adalah gambar 4.1 yaitu gambar topologi

penempatan *router* MikroTik yang digunakan dalam mendapatkan *packet flow* IDS *Anomaly-based*:

Gambar 4.1. Topologi jaringan IDS *anomaly-based*

Menurut Kumar 2021 menjelaskan bahwa proses *sniffing* dilakukan dengan menangkap paket dari komputer aplikasi yang terhubung ke *network device*, dimana *device* disini berfungsi untuk menangkap paket yang berjalan kearah komputer aplikasi maupun keluar aplikasi sehingga penempatan MikroTik tersebut berada di antara pengguna dan komputer aplikasi (*server*), mekanisme ini disebut dengan *ethernet sniffer* atau *network analyzer*.

MikroTik yang berada ditengah antara internet dengan *webserver* akan melakukan *capture packet* antara keduanya, berikut ini adalah gambar 4.2 yang berisi *scripting* yang dilakukan pada *router* MikroTik untuk melakukan sniffing:



Gambar 4.2. *starting sniffing*

Penjelasan gambar 4.2:

a. *Start* dan *Stop Sniffing Anomaly-based*

Script digunakan untuk menjalankan *sniffing* dengan durasi 30 detik, setelah 30 detik *script* akan menjalankan perintah untuk memberhentikan *sniffing*. Setelah proses *sniffing* berhenti maka akan terbentuk sebuah *file* yang diberi nama VUL, dengan format PCAP dan tersimpan di dalam *storage* MikroTik, seperti ditunjukan pada gambar 4.3 berikut ini:

Gambar 4.3. *file* vul.pcap

b. Akses *file* act.php untuk mengunduh vul.pcap pada *server Anomaly-based*

Script yang digunakan untuk menjalankan *file* act.php yang ada pada *server anomaly-based*, perintah tersebut akan menjalankan *command* yang ada pada act.php, fungsi pertama *file* act.php adalah mengunduh *file* act.php yang ada pada *storage* MikroTik. Berikut ini gambar 4.4 yang berisi perintah download file vul.pcap pada *file* act.php:

```
$local_file = 'vul.pcap';
$server_file = 'vul';

// set up basic connection
$ftp = ftp_connect($ftp_server);

// login with username and password
$login_result = ftp_login($ftp, $ftp_user_name, $ftp_user_pass);
// turn passive mode on
ftp_pasv($ftp, true);

// try to download $server_file and save to $local_file
if (ftp_get($ftp, $local_file, $server_file, FTP_BINARY)) {
    echo ":v";
} else {
    echo "There was a problem\n";
}
```

Gambar 4.4 *download file* vul.pcap ke *server anomaly-based*

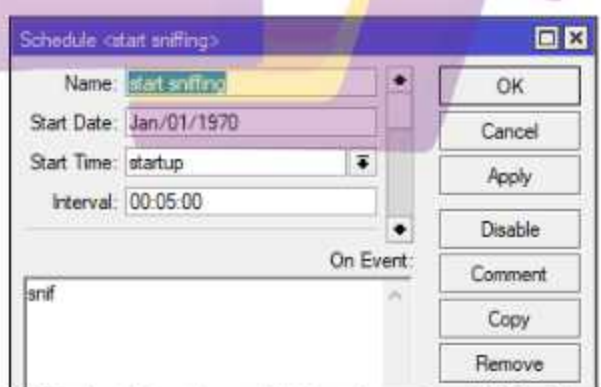
Perintah pada *script* tersebut adalah mendownload file vul yang ada pada *File Transfer Protocol* (FTP) MikroTik dan merubahnya menjadi *file* vul.pcap. berikut ini gambar 4.5 dimana *file* vul.pcap telah berada pada *server anomaly-based* :

Gambar 4.5. *file* vul.pcap telah di *download anomaly-server*

c. Menjalankan *scripting* snif dengan *scheduler*.

Scripting pada MikroTik tidak dapat berjalan secara otomatis tanpa menggunakan *scheduler*, *scheduler* sendiri fungsinya seperti *cron job* yaitu menjalankan sebuah perintah secara otomatis dengan *interval* tertentu.

Berikut ini *scheduler* yang digunakan untuk menjalankan *script* snif secara otomatis.



Gambar 4.6. *setting scheduler* pada *script* snif MikroTik

Penjelasan gambar 4.6 :

a. *Script date* adalah tanggal *script*

b. *Start time* adalah waktu yang ditentukan untuk menjadi awal waktu, karena di dalam *schedule* tersebut di atur *startup* maka *script* akan berjalan ketika *router* pertama kali di hidupkan.

c. *Interval* berisi 00:05:00 yang artinya setiap 5 menit sekali *script* akan dijalankan secara otomatis.

4.1.2. Proses *Convert Packet Anomaly-based*

Packet yang telah di *download* pada langkah sebelumnya masih dalam format *file* PCAP, untuk dapat di *insert* ke dalam tabel *database* MySQL perlu di konversi terlebih dahulu ke format CSV. Berikut ini adalah *command* pada *file* act.php yang berfungsi mengkonversi *file* vul.pcap menjadi *file* vul.csv seperti ditunjukan oleh gambar 4.7 berikut ini:

```
30  //CONVERT PCAP TO CSV
31  $cmd = 'sudo tshark -r vul.pcap -Y "http.host" -T fields
32  -e  frame.number  -e ip.src -e tcp.srcport
33  -e ip.dst -e tcp.dstport  -e http.request.uri
34  -e http.response.code -E quote=d  -E separator=, > vulscan.csv';
35
36  echo shell_exec($cmd);
```

Gambar 4.7. *command* konversi PCAP menjadi CSV pada *anomaly-server*

Data yang akan di *insert* ke dalam *database* yang di dapatkan dari proses *sniffing* memiliki beberapa atribut yaitu seperti pada tabel 4.1 berikut ini.

Tabel 4.1 Atribut *sniffing*

Atribut tshark	Column tabel	Keterangan
frame.number	No.	Merupakan nomor *packet* yang di *sniffing*
ip.src	Source	Merupakan alamat *IP Address* sumber *packet*
tcp.srcport	Srcport	Merupakan *port user* sumber *packet*
ip.dst	Destination	Merupakan alamat *IP Address* sumber *packet*
tcp.dstport	Dstport	Merupakan alamat *port* sumber *packet*
http.request.uri	Content	*URL* yang di tuju *IP source*

4.1.3. Proses *Insert Packet* ke *Database Anomaly-based*

File vul.pcap yang telah dikonversi menjadi *file* CSV akan di *insert* ke dalam tabel vulscan yang ada pada *database anomaly*, tabel vulscan inilah yang nantinya akan dianalisis oleh proses selanjutnya. Berikut ini gambar 4.8 yaitu *command* yang ada pada *file* act.php yang berfungsi *inserting data* ke tabel vulscan.

Gambar 4.8. *command inserting data* ke table *database* pada *anomaly-server*

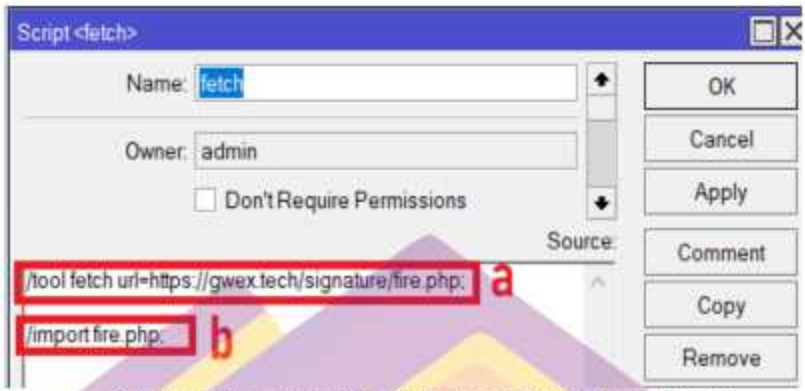
Setelah proses inserting data ke tabel vulscan selesai maka tabel vulscan akan terisi, seperti ditunjukan pada gambar 4.9 berikut ini:

No.	Source	Srcport	Destination	Dstport	Content
153	146.66.178.78	49084	103.213.116.82	80	mkzaim.ru:443
154	146.66.178.78	49084	192.168.1.4	80	mkzaim.ru:443
211	90.151.171.106	43660	103.213.116.82	8291	ip.bablosoft.com:443
180	90.151.171.108	31679	103.213.116.82	80	ip.bablosoft.com:443

Gambar 4.9. isi tabel vulscan

4.1.5 Proses Analisis *Database Anomaly-based*

Table vulscan yang telah terisi data dari proses *sniffing router* MikroTik akan dianalisa oleh *server anomaly-based*. Proses ini akan dilakukan dengan mengakses file fire.php yang ada pada server *anomaly-based*. Berikut ini adalah gambar 4.10 yang menunjukan isi *script*ing pada router MikroTik:

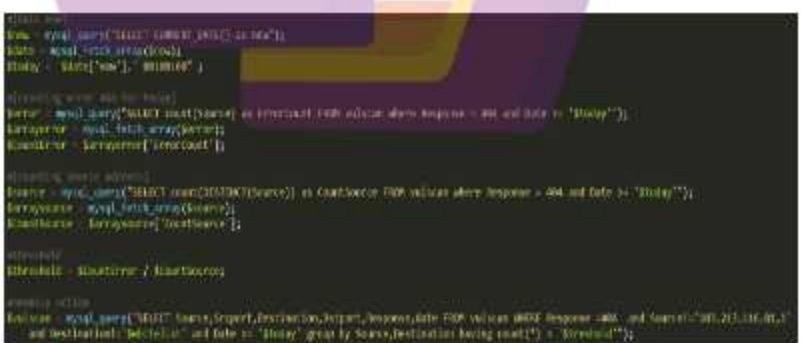


Gambar 4.10. *scripting* MikroTik yang menjalankan *file* fire.php

Penjelasan gambar 4.10 :

a. Proses menjalankan *file* fire.php pada server *Anomaly-based*

Ketika MikroTik menjalankan *script* baris pertama maka MikroTik akan mengakses *file* fire.php pada *server anomaly-based*, *file* fire.php ini memiliki beberapa fungsi yaitu menentukan tanggal hari ini, menghitung jumlah *error* 404, menghitung jumlah *source address* yang mengakses *webserver*, menentukan jumlah *threshold*, dan melakukan seleksi *source address* yang melebihi *threshold* setiap harinya.



Gambar 4.11. proses seleksi *anomaly-based* pada *file* fire.php

Proses seleksi dilakukan dengan kalkulasi rata-rata *error* 404 dari setiap harinya, sehingga *threshold* akan bervariasi sesuai dengan kebiasaan pengguna, hal ini yang membedakan dari penelitian yang dilakukan sebelumnya yaitu penelitian oleh Haryani maupun Syahputra yang menentukan *threshold* secara *static*, selain itu proses perhitungan rata-rata *error* 404 dilakukan dengan penentuan tanggal hal ini berbeda dengan yang dilakukan Fadhlurrohman yang menghitung semua data tanpa penentuan tanggal, hal tersebut apabila di terapkan pada mekanisme *anomaly-based* akan mengakibatkan lambatnya mekanisme dalam menentukan *threshold* akibat data yang terlalu banyak.

b. Proses import *rule firewall* MikroTik *Anomaly-based*

Scripting pada MikroTik yang dilakukan pada baris ke dua pada gambar 4.10 akan mengunduh *file* fire.php, *file* tersebut berisi aturan *firewall* yang terbentuk dari proses analisis yang ditunjukan pada gambar 4.11. Berikut ini adalah tampilan gambar 4.12 yang menunjukan aturan *firewall* hasil dari proses analisis *anomaly-based*.

```
C:\Users\ismai>curl https://gwex.tech/signature/fire.php
/ip firewall filter
rem [find comment=vulscan-anomaly-detection]
add chain=forward protocol=tcp comment=vulscan-anomaly-detection action=drop
 src-address=104.128.64.53  dst-address=192.168.1.4 dst-port=80
```

Gambar 4.12. aturan *firewall* hasil analisis *anomaly-based* pada fire.php

Aturan *firewall* yang telah di *download* akan di implementasikan ke dalam aturan *firewall router* MikroTik sehingga MikroTik akan melakukan *blocking* terhadap *IP Address* yang dianggap sebagai *attacker*. Seperti yang ditunjukan pada gambar 4.13 dibawah ini:

Gambar 4.13. Implementasi aturan *firewall* hasil analisis *anomaly-based*

c. Menjalankan *scripting* fetch dengan *scheduler*

Scheduler ini digunakan untuk menjalankan *script* fetch yang telah dibuat sebelumnya, *script* ini dijalankan dengan *interval* 5 menit sekali sama seperti pada langkah yang ada pada gambar 4.6

Gambar 4.14. *setting scheduler* pada *script* fetch MikroTik

4.2. *Developing IDS Signature-based*

Proses *developing IDS Signature-based* memiliki beberapa perbedaan dengan proses *developing Anomaly-based* diantaranya dalam proses penentuan

serangan, *signature-based* akan mencocokan tabel vulscan yang berisi *data flow* yang ada pada jaringan dengan tabel signature yang sebelumnya telah di simpan di *database*. Berikut ini akan menjelaskan beberapa langkah yang dilakukan di dalam mengembangkan *IDS Signature-based.*

4.2.1 Mengumpulkan *signature* Acunetix *website vulnerability scanner*

Berdasarkan penelitian yang dilakukan Laabid 2021 yang melakukan pengawasan terhadap perilaku *Botnet*, pengawasan di lakukan dengan menganalisis *packet* yang di kirimkan *Botnet*, hal ini menginspirasi penulis untuk mengamati perilaku dari Acunetix guna mendapatkan *signature* dari aplikasi Acunetix *website vulnerability scanner*, berikut ini gambar 4.15 yang menunjukan proses *collecting signature*.

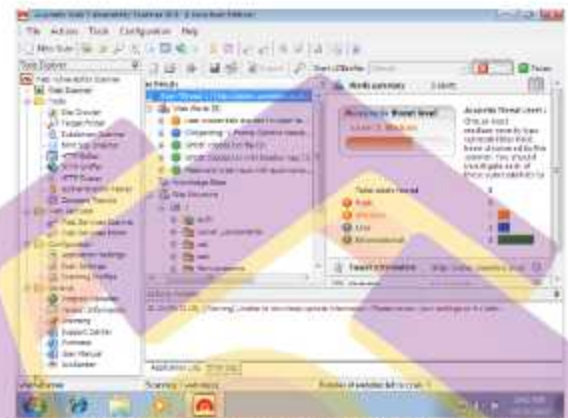
Gambar 4.15. langkah *collecting signature* Acunetix VulScan

Penjelasan gambar 4.15:

a. Proses VulScan *attack* ke webserver

Pada proses ini komputer melakukan VulScan ke *website* menggunakan Acunetix, proses ini diawasi oleh Wireshark yang dapat menangkap *packet*

yang dihasilkan dari proses serangan ke *webserver*, Sehingga Wireshark dapat mengawasi *content* atau *URL* apa saja yang dihasilkan oleh Acunetix ketika melakukan serangan.



Gambar 4.16. VulScan *Attack* untuk mendapatkan pola dari Acunetix

b. Proses *Sniffing packet* menggunakan Wireshark

Proses ini digunakan untuk mendapatkan *content* yang dihasilkan oleh Acunetix ketika melakukan serangan, *content* akan diseleksi secara manual dengan membersihkan data dari duplikasi dan URL yang bersifat umum. pola yang di dapatkan dari Acunetix akan di konversi menjadi CSV dan di *insert* ke database, proses Berikut ini gambar 4.15 yang merupakan hasil *sniffing* yang dilakukan menggunakan Wireshark pada saat Acunetix melakukan serangan.

Gambar 4.17. Proses seleksi data sebelum di *insert* ke dalam tabel *signature*

Setelah data bersih dari duplikasi dan URL yang bersifat umum, maka file akan siap untuk di *insert* ke dalam tabel signature. Dalam mendapatkan pola atau signature dari *tool* Acunetix penulis harus melakukan langkah *collecting* seperti pada gambar 4.13, hal ini karena belum ada *dataset opensource* yang berisi pola dari Acunetix.

c. Proses *import* Acunetix signature ke tabel signature.

File yang telah diseleksi dan dikonversi akan di *import* ke *database* pada tabel signature, tabel ini akan dicocokan dengan tabel vulscan, apabila terdapat *IP Address* yang mengakses *content* yang identik dengan tabel signature maka *IP Address* tersebut akan dianggap sebagai seorang *attacker*.

Gambar 4.18. Tabel signature yang berisi *request URL* pada Acunetix

4.2.2 Proses analisis serangan menggunakan *signature-based* pada fire.php

Signature-based menganalisa paket dengan mencocokan content yang ada pada tabel vulscan dengan tabel signature, berikut ini adalah gambar 4.19 yang berisi *command* seleksi content yang sama antara tabel signature dan tabel vulscan.

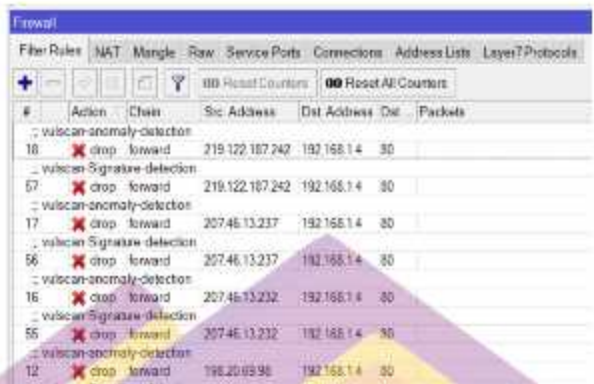
Gambar 4.19 Proses pencocokan content fire.php pada *signature-based*

Apabila terdapat kecocokan antara content yang ada pada tabel vulscan dengan content yang ada pada tabel signature, maka source yang berisi *IP Address* akan dianggap melakukan serangan. Berikut ini gambar 4.20 yang menunjukan apabila *query* tersebut dijalankan.

No.	Source	Sreport	Destination	Dsiport	Content	No.	Name	Content
734	103.135.24.162	3369	103.213.116.82	80		14	ACUNETIX	
735	103.135.24.162	3369	192.168.1.6	80		14	ACUNETIX	
764	103.135.24.162	3370	103.213.116.82	80		10	ACUNETIX	

Gambar 4.20. hasil *query* mekanisme *signature-based*

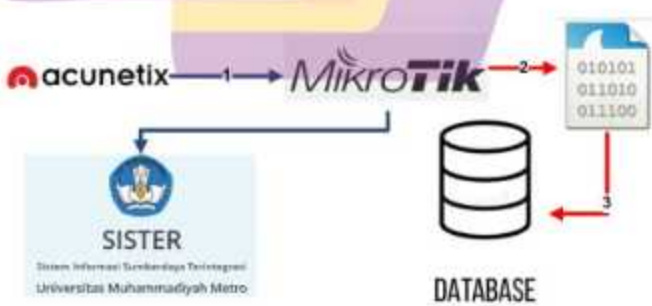
Apabila terdapat *source address* dari table vulscan yang mengakses content seperti pada tabel signature mekanisme *signature-based* akan membuat *rule firewall* yang akan di *import* ke dalam MikroTik seperti yang ditunjukan pada gambar 4.21 berikut.



Gambar 4.21. Implementasi aturan *firewall* hasil analisis *signature-based*

4.2.3 Proses pengumpulan dan *labeling* data yang digunakan untuk menguji metode.

Proses pengumpulan data yang dilakukan di fungsikan untuk melakukan *labeling* antara data yang merupakan serangan dan bukan serangan, proses serangan di lakukan sebanyak 30 kali menggunakan *IP Address* yang berbeda dan telah di catat sebelumnya, sehingga dalam *labeling* data yang menggunakan *IP address* yang telah di catat atau mengakses URL yang identik dengan *tool* Acunetix akan di labeli sebagai serangan. Berikut ini gambar xx proses *labeling* data dari masing-masing metode.



Gambar 4.22. proses *labeling* data dari masing-masing metode

Proses yang di lakukan pada gambar 4.22 dilakukan dengan melakukan simulasi serangan sebanyak 30 kali dengan menggunakan *tool* Acunetix melalui *IP Addresses* yang berbeda, dari hasil uji serangan dan *labeling* data yang di lakukan, di temukan data yang dapat di tunjukan pada tabel 4.2 di bawah ini.

Tabel 4.2. Data sample dari uji coba serangan yang telah di labeli

Nama data	Jumlah baris	Total IP	JUMLAH IP Serangan	JUMLAH IP Bukan serangan
Sample serangan *anomaly-based*	5015	48	38	10
Sample serangan *signature-based*	5301	51	38	13

Pada tabel 4.2 terdapat beberapa variabel yaitu jumlah baris, total IP, jumlah IP serangan dan jumlah IP bukan serangan, sample data dari masing-masing metode di dapatkan dari proses uji coba serangan yang sama, namun memiliki perbedaan pada jumlah baris dan total IP, hal ini di karenakan proses seleksi data yang berbeda, di mana pada saat proses konversi data dari PCAP (data *sniffing*) ke CSV (data yang akan di *import* ke *database*) data pada *anomaly-based* di lakukan dengan menggunakan perintah "*http.response*" sehingga IP yang memiliki *http response* saja yang terbaca, sedangkan pada data *signature-based* proses konversi di lakukan dengan perintah "*http.host*" sehingga semua *IP address* yang melakukan *request* pada *webserver* akan terbaca, sehingga jumlah *IP address* yang terbaca lebih banyak dari data sample *anomaly-based.*

4.3. *Testing*

Proses *testing* dilakukan dengan menyerang *website* yang sudah dilindungi dengan IDS *anomaly-based* dan *signature-based*, tujuanya adalah menemukan metode yang paling baik didalam mendeteksi dan mengantisipasi serangan

VulScan, tabel 4.3 adalah list dari *tools* yang akan digunakan di dalam melakukan serangan kepada *website* target.

Tabel 4.3. *tools* yang digunakan untuk attack *testing*

No	Nama *Tool*	Jenis *Tool*	URL *Tool*
1	Acunetix	Installed Software	
2	Skipfish	Installed software	
3	Nikto	Installed software	
4	Pentest-*Tool*	Online scanner	https://pentest-*tools*.com

Masing-masing *tools* akan menyerang *website* yang dilindungi dengan IDS *anomaly-based* dan *signature-based* secara bergiliran, berikut ini adalah proses *testing* yang dijalankan.

4.3.1 Komparasi kecepatan deteksi serangan Acunetix

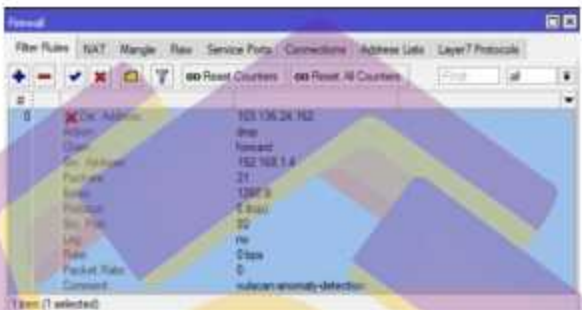
Pada proses *testing* menggunakan Acunetix *vulnerability scanner* dengan menargetkan situs sister.ummetro.ac.id dimana pada serangan pertama situs dilindungi dengan *IDS Anomaly-based* dan *signature-based*.

a. Hasil serangan pada *anomaly-based*

Serangan ke situs sister.ummetro.ac.id menggunakan Acunetix *vulnerability scanner* dapat di deteksi dan antisipasi oleh *IDS anomaly-based*.

Gambar 4.23. Acunetix berhenti karena terdeteksi *IDS Anomaly-based*.

Serangan Acunetix ke situs yang dilindungi oleh mekanisme *anomaly-based* berhenti, hal ini terjadi karena mekanisme mendeteksi serangan dan *firewall rule* di implementasikan ke router MikroTik sehingga *IP Address attacker* terblokir, seperti ditunjukan pada gambar 4.24 dibawah ini.



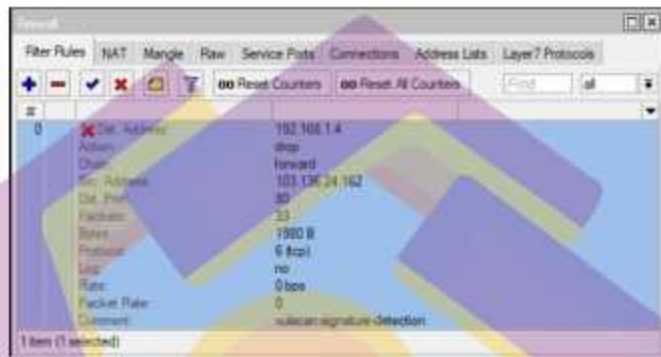
Gambar 4.24. MikroTik *firewall* memblokir Acunetix dengan *Anomaly-based.*

b. Hasil serangan pada *signature-based*

Serangan ke situs sister.ummetro.ac.id menggunakan Acunetix *vulnerability scanner* dapat di deteksi dan antisipasi oleh *IDS signature-based.*

Gambar 4.25. Acunetix berhenti karena terdeteksi *IDS Signature-based.*

Mekanisme dengan *signature-based* dapat mendeteksi serangan VulScan dan mengimplementasikan *rule firewall* ke MikroTik sehingga serangan VulScan pada situs dapat diblokir, seperti yang ditunjukan pada gambar 4.26 dibawah ini.



Gambar 4.26. MikroTik *firewall* memblokir Acunetix dengan *Signature-based.*

Dari hasil *testing* yang dilakukan, mekanisme *anomaly-based* maupun *signature-based* memiliki keunggulan dari penelitian sebelumnya, dimana pada penelitian ini memiliki kemampuan mendeteksi dan mengantisipasi serangan *vulnerability scanning* secara otomatis.

4.3.2 Komparasi kemampuan deteksi serangan berbagai *tools*

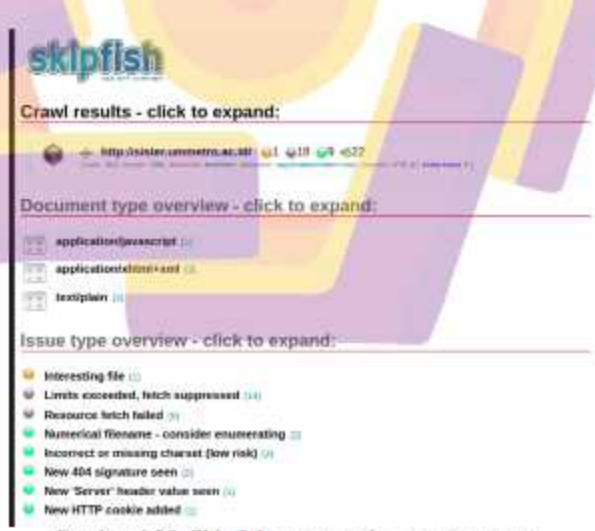
a. Serangan VulScan menggunakan Skipfish

Pada *IDS anomaly-based* serangan Skipfish dapat dideteksi dengan baik, seperti ditunjukan pada gambar 4.27 yang menunjukan *rule firewall* dari *anomaly-based* ketika terdapat serangan VulScan.

Gambar 4.27. MikroTik *firewall* memblokir Skipfish dengan *Anomaly-based.*

Meskipun Skipfish terdeteksi oleh *anomaly-based* namun informasi tentang vulnerability scanner masih menampilkan informasi vulnerability pada situs dengan jumlah informasi yang lebih sedikit. Hal ini terjadi karena mekanisme melakukan update tabel vulscan dengan *interval* 5 menit, sehingga masih ada waktu jeda untuk melakukan serangan.

Berikut ini hasil *testing* yang dilakukan skipfish pada mekanisme *anomaly-based.*



Gambar 4.28. Skipfish *attack* pada *anomaly-based.*

Sedangkan pada saat menggunakan *signature-based*, serangan tidak dapat di deteksi oleh mekanisme *signature-based*. Skipfish sukses mendapatkan informasi vulnerability dari website tanpa terdeteksi. pada gambar 4.29 yang menunjukan informasi vulnerability dari website.

Gambar 4.29. Skipfish *attack* pada *signature-based*.

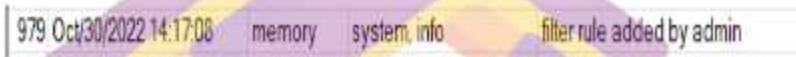
Dari hasil *testing* serangan diatas dapat disimpulkan bahwa *anomaly-based* lebih unggul didalam mendeteksi serangan Skipfish di bandingkan dengan *signature-based*.

b. Serangan VulScan menggunakan Nikto

Serangan *vulnerability scanning* yang dilakukan dengan Nikto dapat dideteksi oleh *anomaly-based*, serangan dilakukan pada pukul 03:10:11 pada jam *PC attacker* yang memiliki selisih waktu 1 jam lebih cepat dari jam pada routerboard MikroTik.

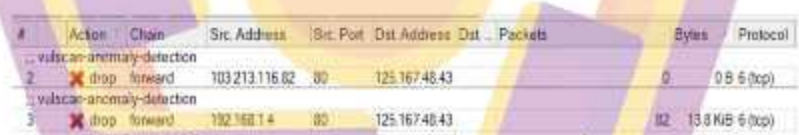
Gambar 4.30. Nikto *attack* pada *anomaly-based.*

Serangan tersebut di block oleh *firewall* pada pukul 14:17:02 jam *router*, seperti pada gambar 4.31 berikut ini.



Gambar 4.31. *log firewall rule blocking* Nikto *attack* pada *anomaly-based.*

Artinya serangan Nikto pada *website* yang dilindung mekanisme *anomaly-based* dapat di deteksi dalam waktu 6 menit 51 detik. Serangan Nikto terhenti ketika *firewall* memblokir *webserver* untuk mengirimkan *response* kepada *attacker* seperti di tunjukan pada gambar 4.32 di bawah ini.



Gambar 4.32. *firewall rule blocking* Nikto *attack* pada *anomaly-based.*

Sedangkan serangan Nikto pada website yang di lindungi mekanisme *signature-based* tidak dapat di deteksi, dan menampilkan info vulnerability dari website, seperti yang terlihat pada gambar 4.33 dibawah ini.

Gambar 4.33. *result* Nikto *attack* pada *signature-based*.

c. Serangan VulScan menggunakan Pentest-*Tool*

Serangan menggunakan *online* vulnerability scanner dari Pentest-*Tool* tidak dapat di deteksi menggunakan mekanisme *anomaly-based* maupun *signature-based*, serangan berhasil mendapatkan informasi yang cukup lengkap untuk melakukan serangan ke tahap eksploitasi vulnerability yang telah didapatkan.

Gambar 4.34. *result* Pentest-*Tool* pada *anomaly-based*

Serangan menggunakan *online* VulScan menggunakan Pentest-*Tool* tidak menghasilkan *error* 404 sehingga serangan tidak dapat di deteksi oleh

Mekanisme *Anomaly-based,* berikut ini gambar 4.35 yang menampilkan tabel vulscan setelah serangan selesai dilakukan.

Gambar 4.35. tidak terdapat *record* pada tabel setelah serangan Pentest-*Tool*.

Gambar 4.35 menunjukan bahwa tabel vulscan tidak memiliki record apapun, ini menunjukan serangan dari Pentest-*Tool* bersih dari error 404, sehingga *anomaly-based* maupun *signature-based* tidak dapat mendeteksi serangan.

4.3.3 Performa Metode anomaly-based dan signature-based

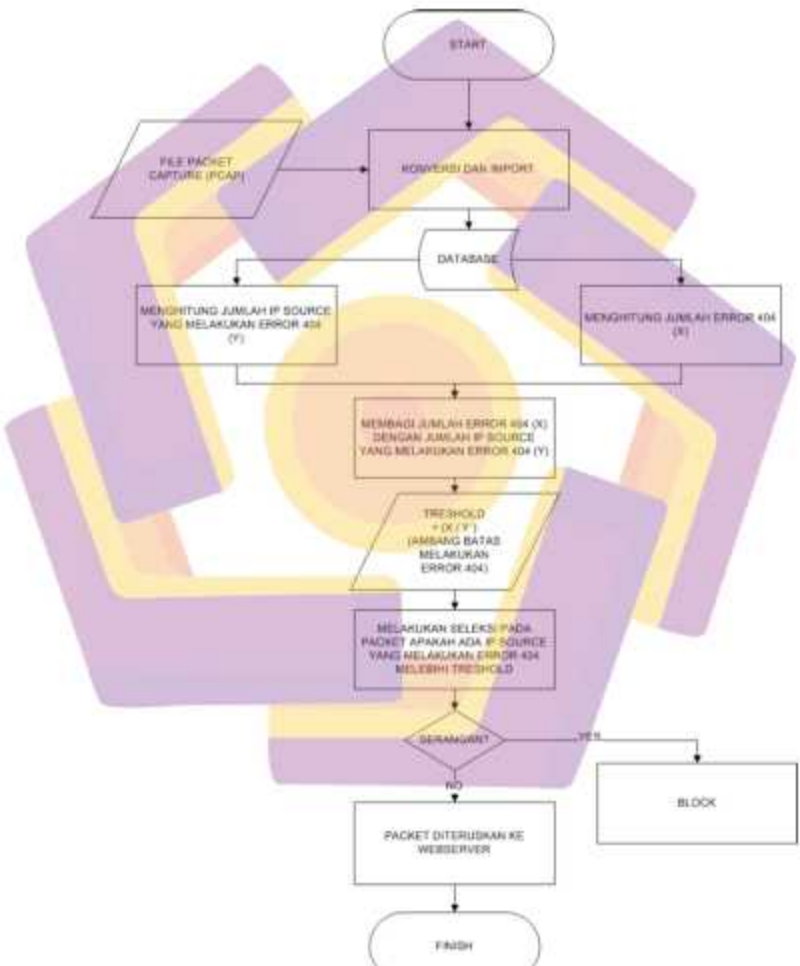
Setelah menjalani *testing* baik metode *anomaly-based* dan *signature-based* mampu mendeteksi aktivitas *web vulnerability scanner*, sehingga perlu di uji performa dari masing-masing metode sehingga menemukan nilai akurasi, presisi dan sensitivitas dari kedua metode. Berikut ini hasil uji performa dari masing-masing metode.

a. Performa metode *anomaly-based*

Metode *anomaly-based* menentukan sebuah serangan dengan melakukan perhitungan statistik, yaitu menghitung jumlah error 404 dalam satu hari. Data sample yang telah di kumpulkan dan di labeli sebelumnya akan di uji untuk

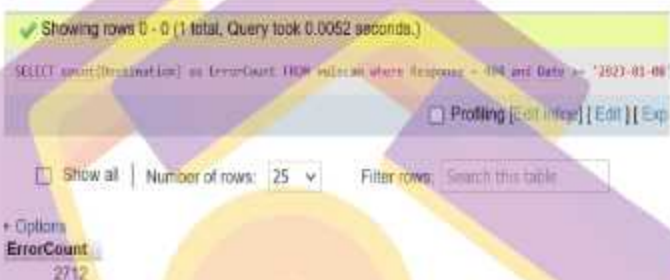
menemukan ke sesuaian deteksi dari metode *anomaly-based* dengan label pada data.

Berikut ini adalah gambar 4.33 yang menunjukan alur kerja dari *anomaly-based* dalam mendeteksi sebuah aktivitas *vulnerability scanner*.



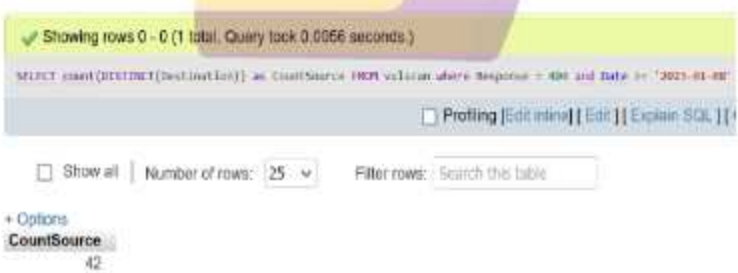
Gambar 4.36. alur metode *anomaly-based* dalam menentukan serangan.

Proses awal yang di lakukan dalam pengujian ini dengan menentukan tanggal data yang akan di analisis, sesuai dengan data yang di ambil pada tanggal 08 bulan januari 2023 maka variabel tanggal yang di gunakan adalah "2023-01-08", setelah menentukan tanggal langkah selanjutnya adalah menemukan jumlah *error 404* pada tanggal tersebut. Berikut ini adalah gambar 4.37 yang menunjukan proses perhitungan jumlah error 404 pada tanggal tersebut.



Gambar 4.37. alur menghitung jumlah error 404 pada tanggal 01/08/2023.

Pada gambar 4.37 hasil query menunjukan jumlah *error 404 (ErrorCount)* pada tanggal 01 januari 2023 memiliki nilai 2712. Langkah selanjutnya metode *anomaly-based* akan menghitung jumlah *IP Address* yang melakukan *error 404* pada tanggal tersebut. Proses penentuan jumlah host yang melakukan *error 404* di tunjukan pada gambar 4.38 di bawah ini.



Gambar 4.38. alur menghitung jumlah host yang melakukan error 404 pada tanggal 01/08/2023.

Hasil perhitungan jumlah host yang melakukan *error 404 (CountSource)* yaitu sebanyak *42 host*. Setelah menghitung jumlah host yang melakukan *error 404* langkah selanjutnya yang di lakukan adalah menghitung jumlah *threshold (ambang batas toleransi error 404)* threshold di dapatkan dengan formula 4 yang ada di bawah ini.

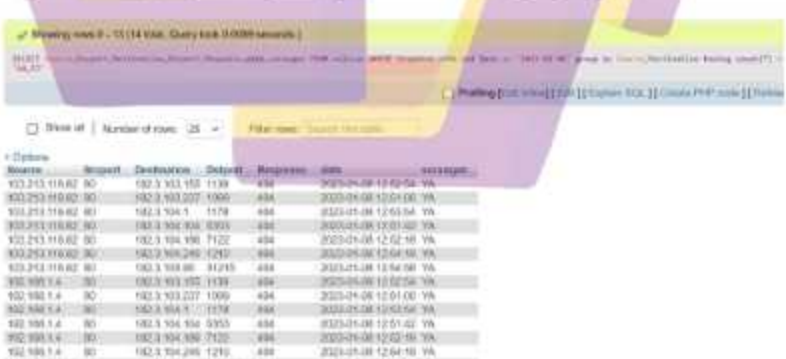
$$Threshold = (x/y) \quad (4)$$

Keterangan :

x = *ErrorCount*

y = *CountSource*

Threshold di dapatkan dari hasil bagi *ErrorCount* (x) yaitu 2712 dengan *CountSource* (y) yaitu 42, maka nilai *threshold* yang di dapatkan adalah 64,57. Setelah nilai *threshold di* dapatkan proses selanjutnya adalah proses seleksi host yang melakukan serangan, host dikatakan melakukan serangan apabila melakukan *error 404* melebihi dari jumlah *threshold*, di bawah ini merupakan gambar 4.39 yang menunjukan proses seleksi serangan yang di lakukan.



Gambar 4.39. proses seleksi serangan pada *anomaly-based*.

Dari proses seleksi metode *anomaly-based* yang di lakukan pada gambar 4.39 di temukan 14 host yang melakukan serangan dan memiliki label "YA" artinya hasil seleksi telah sesuai, namun merujuk pada tabel 4.2 yaitu data *sample* dari uji coba serangan yang telah di labeli, terdapat 38 host yang melakukan serangan. Maka nilai akurasi, presisi dan sensitivitas dari metode *anomaly-based* dapat di lihat pada tabel 4.4 di bawah ini.

Tabel 4.4. performa dari metode *anomaly-based* dalam mendeteksi serangan *web vulnerability scanner*.

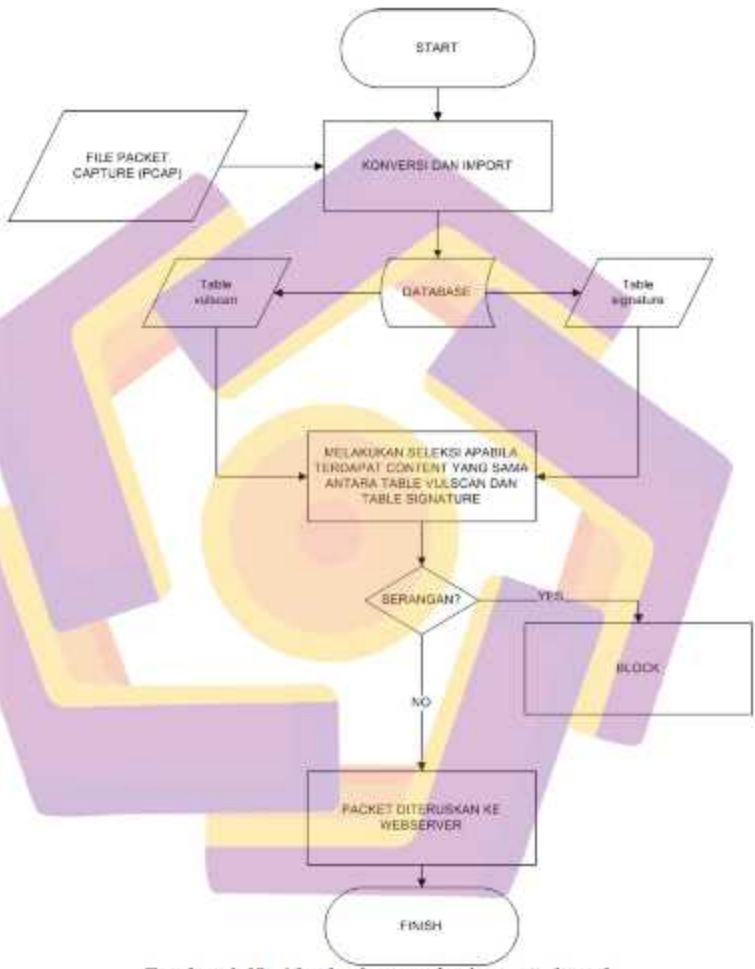
Total IP = 48	Aktual Positif (YES)	Aktual Negatif
Data sample *anomaly-based*	38	10
Hasil Seleksi *anomaly-based*	TP = 14	FP = 0
	FN = 24	TN = 10
Nilai akurasi	(14+10) / (14+0+24+10) = 24/48 = 0,5	
Nilai presisi	(14)/(14+0)	= 14/14 = 1
Nilai sensitivitas	(14)/(14+24)	= 14/38 = 0,36

Dari tabel 4.4 di proleh informasi performa dari metode *anomaly-based* dalam mendeteksi serangan *web vulnerability scanner* yaitu nilai akurasi sebesar 0,5 , nilai presisi sebesar 1 dan nilai sensitivitas sebesar 0,36.

b. Akurasi metode *signature-based*

Berbeda dengan metode *anomaly-based*, *signature-based* bekerja dengan membandingkan *packet* yang ada pada tabel *vulscan* yaitu tabel yang berisi packet yang di dapatkan dari proses *sniffing*, terdapat kolom *content* pada tabel *vulscan* yang berisi *URL* dimana kolom *content* ini akan di bandingkan dengan kolom *content* yang ada pada tabel *signature* yaitu tabel yang berisi *URL* yang di dapatkan dari mengamati URL yang di *request* Acunetix.

Berikut ini gambar 4.40 yang menunjukan cara kerja dari metode *signature-based.*



Gambar 4.40. Alur kerja metode signature-based.

Signature-based memiliki akurasi yang sangat baik dalam mendeteksi aktivitas *vulnerability scanner*, berikut ini gambar 4.41 yang menunjukan kinerja dari metode signature-based dalam mendeteksi sebuah serangan.

Gambar 4.41. *signature-based* mendeteksi aktivitas *vulnerability scanner*

Dari proses seleksi metode *signature-based* yang di lakukan pada gambar 4.41 di temukan 38 *host* yang melakukan serangan dan memiliki label "YA" artinya hasil seleksi telah sesuai, merujuk pada tabel 4.2 yaitu data *sample* dari uji coba serangan yang telah di labeli, terdapat 38 host yang melakukan serangan. Maka nilai akurasi, presisi dan sensitivitas dari metode *anomaly-based* dapat di lihat pada tabel 4.5 di bawah ini.

Tabel 4.5. performa dari metode *signature-based* dalam mendeteksi serangan *web vulnerability scanner*.

Total IP = 51	Aktual Positif (YES)	Aktual Negatif
Data sample *signature-based*	38	13
Hasil Seleksi *signature-based*	TP = 38	FP = 0
	FN = 0	TN = 13
Nilai akurasi	(38+13) / (38+0+0+13) = 51/51 = 1	
Nilai presisi	(38)/(38+0) = 38/38 = 1	
Nilai sensitivitas	(38)/(38+0) = 38/38 = 1	

Dari tabel 4.5 di proleh informasi performa dari metode *signature-based* dalam mendeteksi serangan *web vulnerability scanner* yaitu nilai akurasi sebesar 1, nilai presisi sebesar 1 dan nilai sensitivitas sebesar 1.

4.4. Evaluasi *Testing*

Dari hasil *testing* yang telah di lakukan, ditemukan bahwa antara metode *anomaly-based* dan *signature-based* memiliki perbedaan kecepatan dalam mendeteksi dan mengantisipasi serangan, selain dari kecepatan metode *anomaly-based* dan *signature-based* memiliki perbedaan di dalam mendeteksi serangan.

Serangan *vulnerability scanner* yang dilakukan menggunakan Acunetix berhasil di deteksi oleh ke dua metode yaitu *anomaly-based* dan *signature based*, namun terdapat perbedaan dalam kecepatan yang dapat dilihat pada table 4.6 di bawah ini.

Tabel 4.6. Perbandingan kecepatan deteksi VulScan *attack* dengan Acunetix

No	Metode	Time	Request
1	*Anomaly-based*	11 Menit,59 Detik	1202
2	*Signature-based*	9 Menit, 16 Detik	855
GAP		2 Menit, 44 Detik	347

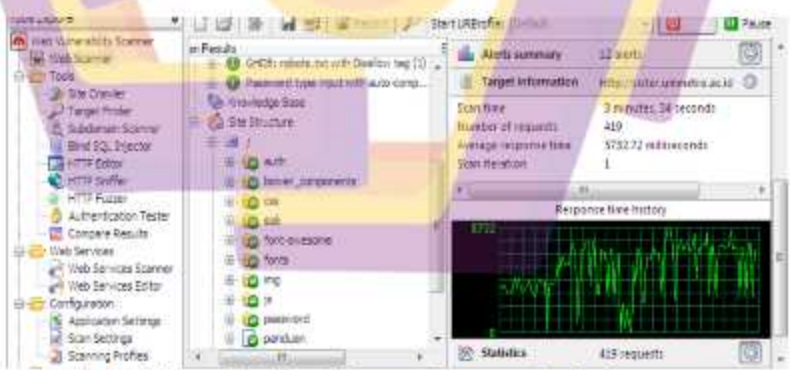
Dari hasil *testing* yang dilakukan, meskipun *anomaly-based* dapat mendeteksi serangan Acunetix namun waktu 12 Menit sangatlah cukup untuk menemukan celah pada *website*, untuk itu perlu adanya penentuan *interval scheduler* pada MikroTik yang tepat, hal ini di maksudkan untuk mengurangi jeda sehingga kelemahan pada website tidak terlalu banyak ter ekspos, dari *testing* tersebut terlihat serangan terdeteksi pada *interval* ke tiga untuk *anomaly-based* dan *interval* ke dua untuk *signature-based*, dimana setiap *interval* berlangsung selama 5 menit.

Dalam meningkatkan kecepatan deteksi penulis mencoba untuk meningkatkan *interval scheduler* MikroTik menjadi 3 menit dan menemukan

mekanisme *anomaly-based* dapat mendeteksi serangan dengan durasi 6 menit 12 detik atau pada *interval* ke dua, seperti ditunjukan pada gambar 4.42 dibawah ini.

Gambar 4.42. Speed deteksi setelah merubah *interval* mekanisme *anomaly-based*.

Berbeda dengan *anomaly-based*, *signature-based* dapat mendeteksi serangan dengan waktu 3 menit 34 detik atau pada *interval* pertama seperti ditunjukan pada gambar 4.43 dibawah ini



Gambar 4.43. Speed deteksi setelah merubah *interval* mekanisme *signature-based*.

Setelah meningkatkan *interval* pada *scheduler* MikroTik menjadi 3 Menit, Mekanisme *anomaly-based* dan *signature-based* dapat mendeteksi menjadi lebih

cepat, Berikut ini tabel 4.7 yang menunjukan perbedaan kecepatan deteksi setelah *interval* dirubah menjadi 3 menit.

Tabel 4.7. Perbandingan kecepatan deteksi VulScan *attack* dengan Acunetix setelah *interval scheduler* dirubah

No	Metode	Time	Request
1	*Anomaly-based*	6 Menit,12 Detik	599
2	*Signature-based*	3 Menit, 34 Detik	419
GAP		2 Menit, 38 Detik	180

Dari segi kecepatan dalam mendeteksi serangan VulScan *Signature-based* lebih unggul dari *Anomaly-based* seperti pada tabel 4.7 diatas. Namun dalam segi kemampuan mendeteksi serangan, *Anomaly-based* lebih unggul di bandingkan dengan *signature-based*.

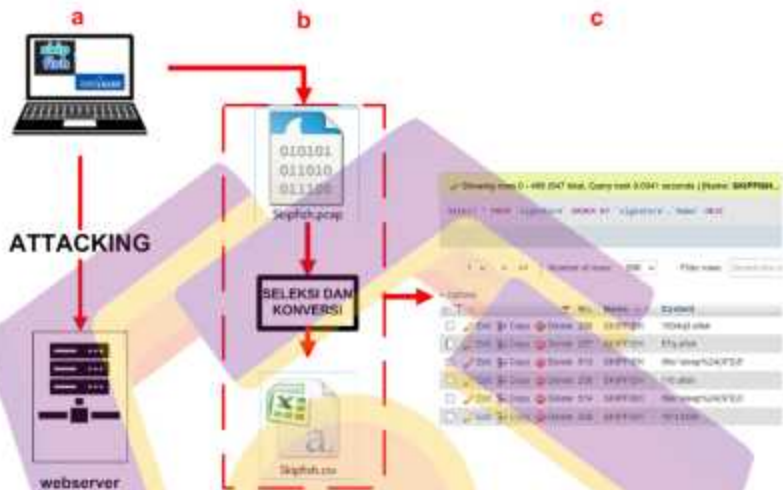
Berikut tabel 4.8 yang menjelaskan perbandingan dalam mendeteksi serangan dari setiap *tools*.

Tabel 4.8. perbandingan mendeteksi serangan dari berbagai *tools*

No	Nama *Tool*	*Anomaly-based*	*Signature-based*
1	Acunetix	Terdeteksi	Terdeteksi
2	Skipfish	Terdeteksi	Tidak Terdeteksi
3	Nikto	Terdeteksi	Tidak Terdeteksi
4	Pentest-*Tool*	Tidak Terdeteksi	Tidak Terdeteksi

Dari hasil tersebut, penulis mencoba menambahkan pola dari Skipfish ke tabel signature dengan tujuan meningkatkan kemampuan mekanisme *signature-based* dalam medeteksi serangan dari Skipfish, dalam mengumpulkan pola Skipfish cara yang dilakukan sama dengan mendapatkan pola Acunetix, yaitu menggunakan wireshark dan membersihkan data dari duplikat dan mengimport

pola ke dalam tabel signature yang ada pada database. Langkah tersebut dapat dilihat pada gambar 4.44 dibawah ini.

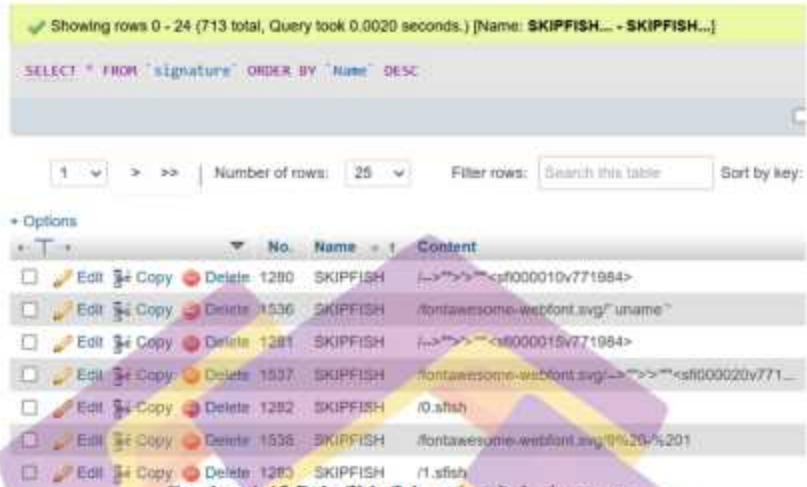


Gambar 4.44. colecting data pola Skipfish *signature-based*.

Penjelasan gambar 4.44:

a. Serangan yang dilakukan dengan menggunakan *tool* skipfish kepada situs target, diawasi dengan wireshark.
b. Data yang telah didapatkan dikonversi menjadi file CSV, dibersihkan dari duplikasi dan URL yang tidak mengandung pola dari skipfish.
c. Pola yang telah bersih dari duplikasi dan URL yang tidak perlu di import ke dalam database.

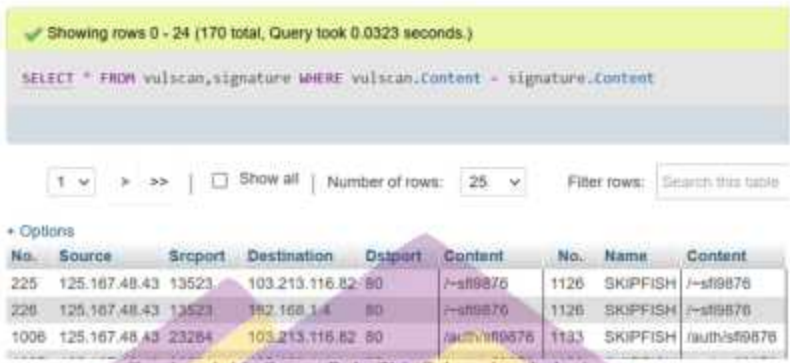
Berikut ini gambar 4.45 yang menunjukan pola skipfish telah di insert ke dalam tabel signature.



Gambar 4.45 Pola Skipfish pada tabel signature.

Setelah menambahkan pola Skipfish ke dalam tabel signature, mekanisme signature base dapat mendeteksi serangan yang dilakukan menggunakan *tool* Skipfish, hal ini membuktikan pendapat Otum pada 2021 bahwa dengan melakukan update pola pada tabel signature akan meningkatkan kemampuan mekanisme dalam mendeteksi serangan, karena *signature-base* menentukan serangan berdasarkan pola yang sebelumnya telah di ketahui.

berikut ini gambar 4.46 yang merupakan *query* yang menunjukan terdapat beberapa *packet* pada tabel vulscan yang *match* dengan pola yang tersimpan pada table signature.


Showing rows 0 - 24 (170 total, Query took 0.0323 seconds.)

```
SELECT * FROM vulscan,signature WHERE vulscan.Content = signature.Content
```

1 > >> | Show all | Number of rows: 25 Filter rows: Search this table

+ Options

No.	Source	Srcport	Destination	Dstport	Content	No.	Name	Content
225	125.167.48.43	13523	103.213.116.82	80	/~sfi9876	1126	SKIPFISH	/~sfi9876
226	125.167.48.43	13523	192.168.1.4	80	/~sfi9876	1126	SKIPFISH	/~sfi9876
1006	125.167.48.43	23264	103.213.116.82	80	/auth/sfi9876	1133	SKIPFISH	/auth/sfi9876

Gambar 4.46. Query Pola Skipfish pada tabel signature.

Setelah mekanisme *anomaly-based* dan *signature-based* dapat bekerja dengan baik, selain itu mekanisme yang di bangun dapat berjalan secara otomatis sehingga pengamanan tidak memerlukan campur tangan administrator di dalam langkah antisipasi, hal ini perlu sesuai dengan yang di ungkapkan oleh Risqiwati 2018 bahwa sistem informasi memerlukan mekanisme pengamanan secara *realtime*. Penelitian yang ada pada tabel 2.1 yaitu tabel matrix literatur *review* tidak melakukan deteksi pada serangan *vulnerability-scanner*, mengutip dari Subandi dan Huang proses *Vulnerability scanner* mampu digunakan untuk menemukan celah keamanan pada *web server*, untuk itu dengan adanya mekanisme yang di bangun ini, *vulnerability scanning* oleh orang yang tidak bertanggung jawab dapat dicegah guna menghindari serangan pasca ditemukan nya kerentanan pada *web server*.

Selain itu mekanisme yang di bangun perlu menambahkan *whitelist IP address*, hal ini digunakan untuk mengecualikan penetrasi *testing* yang dilakukan oleh orang yang berwenang, *script* fire.php pada *anomaly-based* dan *signature-based* dapat mengecualikan *IP Address* tertentu yang di ijinkan untuk melakukan

penetrasi *tester*, sehingga *pentester* yang diberikan wewenang tidak terblokir.

Berikut ini adalah gambar 4.47 yang menggambarkan bagaimana *whitelist* di implementasikan pada mekanisme yang telah dibangun.

Gambar 4.47. Contoh penerapan *whitelist* pada mekanisme.

Penjelasan gambar 4.47:

a. *Rule firewall* melakukan blocking kepada *IP Address* 125.167.48.43
b. *Script* fire.php dirubah untuk menambahkan *IP Address* 125.167.48.43 hal ini digunakan untuk mencegah *IP Address* tersebut tidak di blokir ketika terdeteksi mekanisme yang dibangun.
c. *Whitelist* berhasil mengecualikan *IP Address* yang telah di insert ke dalam *script* fire.php, sehingga *rule firewall* tidak memblokir *IP Address* yang telah di *whitelist*.

Dengan adanya *whitelist* aktifitas *vulnerability scanner* masih dapat di lakukan oleh orang yang berwenang, hal ini merujuk pada Subandi bahwa proses *vulnerability scanner* digunakan untuk upaya mitigasi terhadap celah yang di temukan.

4.5. Hasil

Hasil *testing* dan evaluasi di dapatkan bahwa mekanisme yang di bangun dengan metode *anomaly-based* maupun *signature-based* dapat berjalan dengan baik, selain itu beberapa temuan dari hasil *testing* dan evaluasi adalah sebagai berikut.

4.5.1 Komparasi kecepatan dalam mendeteksi *tool* Acunetix

Perbandingan kecepatan dalam mendeteksi serangan Acunetix antara Metode *anomaly-based* dan *signature-based* dapat di lihat pada tabel 4.9 di bawah ini.

Tabel 4.9. Hasil perbandingan kecepatan deteksi VulScan *attack* dengan Acunetix

No	Metode	Time	Request
1	*Anomaly-based*	11 Menit,59 Detik	1202
2	*Signature-based*	9 Menit, 16 Detik	855
GAP		**2 Menit, 44 Detik**	**347**

Signature-based lebih unggul dalam hal kecepatan mendeteksi dan mengantisipasi serangan di bandingkan dengan *anomaly-based* dengan selisih waktu 2 menit, 44 detik selain itu *requests* yang di hasilkan Acunetix lebih sedikit pada *signature-based* yaitu sebanyak 855 *requests* selisih 347 *requests* di bandingkan *anomaly-based*.

4.5.2 Ketepatan *interval scheduller* menentukan kecepatan deteksi

Interval scheduler yang di tingkatkan menjadi 3 menit berdampak pada peningkatan waktu deteksi, penentuan 3 menit ini ditentukan dengan mempertimbangkan proses *sniffing data*, proses *upload data*, proses konversi *data*, proses *import data* ke *database*, proses seleksi dan proses *import firewall rule*, waktu 3 menit dapat mengakomodir semua proses yang dilakukan tanpa

kendala, namun jika waktu di perkecil kurang dari 3 menit di khawatirkan akan mengakibatkan mekanisme tidak berjalan dengan baik, karena *file* yang akan di konversi di *replace* dengan file yang baru sehingga *data* yang di *import* ke dalam *database* menjadi tidak *valid*.

Hasil dengan meningkatkan interval *scheduler* menjadi 3 menit ini menghasilkan waktu deteksi yang lebih cepat, berikut ini tabel perbandingan antara *anomaly-based* dan *signature-based* setelah meningkatkan interval *scheduler*.

Tabel 4.10. Hasil perbandingan kecepatan deteksi VulScan *attack* dengan Acunetix setelah *interval scheduler* dirubah

No	Metode	Time	Request
1	*Anomaly-based*	6 Menit,12 Detik	599
2	*Signature-based*	3 Menit, 34 Detik	419
GAP		**2 Menit, 38 Detik**	**180**

Signature-based tetap lebih unggul dalam hal kecepatan mendeteksi dan mengantisipasi serangan di bandingkan dengan *anomaly-based* dengan selisih waktu 2 menit, 38 detik selain itu *requests* yang di hasilkan Acunetix lebih sedikit pada *signature-based* yaitu sebanyak 419 *requests* selisih 180 *requests* di bandingkan *anomaly-based*.

4.5.3 Komparasi kemampuan dalam mendeteksi serangan dari berbagai *tools Web Vulnerability scanner*

Hasil *testing* yang dilakukan dengan melakukan serangan dengan menggunakan berbagai *tools* pada masing-masing metode menunjukan beberapa temuan. berikut ini merupakan tabel 4.11 yang menunjukan hasil dari testing tersebut

Tabel 4.11. Hasil perbandingan mendeteksi serangan dari berbagai *tools*

No	Nama *Tool*	*Anomaly-based*	*Signature-based*
1	Acunetix	Terdeteksi	Terdeteksi
2	Skipfish	Terdeteksi	Tidak Terdeteksi
3	Nikto	Terdeteksi	Tidak Terdeteksi
4	Pentest-*Tool*	Tidak Terdeteksi	Tidak Terdeteksi

Tabel 4.11 menunjukan bahwa metode *anomaly-based* mampu mendeteksi 3 *tools* yang digunakan sehingga menyimpulkan metode *anomaly-based* mampu mendeteksi serangan yang belum diketahui sebelumnya dengan mengandalkan perbandingan *threshold* dengan *current activity*, sedangkan *signature-based* hanya mampu mendeteksi serangan dari *tool* Acunetix yang pola seranganya telah ada pada *database signature*. Terdapat *tool* yang tidak mampu di deteksi kedua metode yaitu *online pentest-tool*. Perlu penelitian selanjutnya untuk menemukan kenapa hal ini terjadi.

4.5.4 Peningkatan kemampuan *signature-based* dengan menambahkan pola ke tabel *signature*

Setelah menambahkan pola Skipfish pada table *signature*, mekanisme menggunakan metode *signature-based* dapat mendeteksi serangan Skipfish yang sebelumnya tidak dapat terdeteksi, hal ini membuktikan pendapat Imam tahun 2019 pembaharuan pola pada *database* pola dapat meningkatkan kemampuan *signature-base* dalam mendeteksi serangan.

4.5.5 Hasil perbandingan performa metode *anomaly-based* dan *signature-based.*

Setelah melakukan testing kecepatan dan komparasi kemampuan mendeteksi serangan berbagai *tools*, berikut ini adalah hasil perbandingan dari kedua metode dalam hal akurasi, presisi dan sensitivitas dalam mendeteksi serangan *web vulnerability scanner*.

Tabel 4.12. Rekap perbandingan performa metode *anomaly-based* dan *signature-based*

Metode	**Akurasi**	**Presisi**	**Sensitivitas**	**Kemampuan deteksi *Vulnerability scanner* (*known attack*)**	**Kemampuan deteksi *Vulnerability scanner* (*unknown attack*)**
Anomaly-based	0,5	1	0,36	Mampu mendeteksi serangan dari berbagai *tools*	Mampu mendeteksi serangan yang sebelumnya belum pernah di petakan
Signature-based	1	1	1	Mampu mendeteksi serangan yang sudah ada pada *database signature*	Tidak mampu mendeteksi apabila pola belum tersedia pada *database signature*

Tabel 4.12 merupakan perbandingan dari kedua metode selain dari segi kecepatan, *anomaly-based* memiliki performa yang lebih rendah dengan nilai akurasi 0,5, presisi 1 dan sensitivitas 0,36. Sedangkan metode signature-based memiliki nilai akurasi yang lebih baik yaitu ber nilai 1, presisi 1 dan nilai sensitivitas 1. *Signature-based* memiliki akurasi tinggi karena pola telah tersedia pada *database signature, n*amun *signature-based* tidak dapat mendeteksi serangan

apabila pola tidak tersedia di dalam database signature. *Anomaly-based* mampu mendeteksi serangan meskipun tanpa pola karena mengandalkan analisis perilaku pengguna dan membandingkan dengan *threshold* yang telah di dapatkan.

4.5.6 Perbandingan dengan penelitian sebelumnya.

Perbedaan yang paling utama adalah penelitian ini berfokus pada deteksi dan antisipasi serangan *vulnerability scanner* yang tidak di lakukan pada penelitian yang menjadi rujukan pada tabel 2.1 yaitu tabel matrix literatur review, selain itu beberapa ke ungulan penelitian ini dari beberapa penelitian yang menjadi rujukan dapat di lihat pada tabel 4.11 di bawah ini.

Tabel 4.13. Hasil perbandingan dengan penelitian sebelumnya

No	Peneliti	*Mekanisme otomatis*	*Fitur whitelist*
1	Fadhlurrohman, M.,Muliawati, A., &Hananto, B. , Jurnal Ilmu Komputer danAgri-Informatika, 8(2), 90-94. , 2021	TIDAK	TIDAK
2	Haryani, P., &Raharjo, S. , Jurnal Jarkom, 9(1), 1-9. , 2021	YA	TIDAK
3	Hidayat, A., &Saputra, I. P. ,Jurnal RESISTOR(Rekayasa Sistem Komputer), 1(2), 118-124. , 2018	YA	TIDAK
4	Syahputra, W.,Diansyah, T. M., &Liza, R. , Seminar Nasional Teknologi Informasi &Komunikasi (Vol.1, No. 1, pp. 492-499), , 2020	YA	TIDAK
5	Sau, W. M. T., &Siswantyo, S. , Info Kripto, 15(2), 97-104. , 2021	TIDAK	TIDAK
6	Widodo, T., & Aji,A. S. , JISKA (Jurnal Informatika Sunan Kalijaga),7(1), 46-55. , 2022	TIDAK	TIDAK

Penelitian yang ada pada tabel 4.10 beberapa tidak memiliki kemampuan dalam mengantisipasi serangan secara otomatis dan masih membutuhkan campur tangan *administrator* dalam antisipasi serangan, sedangkan penelitian ini telah

berhasil membangun mekanisme otomatis yang digunakan untuk memblokir akses *attacker* ke dalam sistem, hal ini merujuk pendapat Risqiwati tahun 2018 bahwa serangan pada sistem dapat terjadi kapan saja dan harus di antisipasi sepanjang waktu (*realtime*), sehingga mekanisme otomatis akan bermanfaat apabila *administrator* sedang tidak *standby* memonitoring sistem.

Selain itu mekanisme yang dibangun memiliki fitur *whitelist*, hal ini digunakan untuk memberikan pengecualian pada *pentester* yang memiliki wewenang untuk melakukan *vulnerability scanning*, hal ini mengacu pada pendapat Zirwan tahun 2022 bahwa *scanning vulnerability* pada suatu *website* sangat penting bagi pengembang *website* tersebut untuk meningkatkan keamanan pada *website* dan menutup celah yang telah terdeteksi oleh *website vulnerability scanner*, sebelum celah tersebut di ketahui dan di eksploitasi orang lain.

BAB V

PENUTUP

5.1. Kesimpulan

1. Berdasarkan hasil penelitian yang telah dilakukan, mekanisme otomatis yang dibangun baik menggunakan *Anomaly-based* maupun *Signature-based* dapat berjalan dengan baik di dalam mendeteksi serangan *vulnerability scanner* yang dilakukan oleh Acunetix, proses deteksi dilakukan secara otomatis tanpa campur tangan *administrator*, hal ini lebih unggul dari beberapa penelitian sebelumnya yang masih membutuhkan campur tangan *administrator* dalam memblokir serangan, selain itu mekanisme yang dibangun memiliki keunggulan yang mampu melakukan pengecualian terhadap *IP Address* tertentu dengan menggunakan *whitelist*.
2. Dalam hal kecepatan mendeteksi serangan dengan *tools* Acunetix metode *Signature-based* lebih unggul dengan selisih 2 menit 44 detik lebih cepat di bandingkan dengan metode *Anomaly-based*. Jumlah *request* yang terkirim pada *website* yang dilindungi mekanisme *Signature-based* lebih sedikit 347 *request* di bandingkan dengan *Anomaly-based* sehingga dalam hal kecepatan *Signature-based* lebih unggul di banding *Anomaly-based*.
3. *Interval scheduler* pada MikroTik mempengaruhi kecepatan mekanisme dalam mendeteksi serangan, setelah *interval* di kecilkan menjadi 3 menit, mekanisme *anomaly-based* dan *signature-based* menjadi lebih cepat di dalam mendeteksi serangan, *anomaly-based* dapat mendeteksi serangan dalam waktu 6 menit 12

detik, sedangkan *signature-based* dapat mendeteksi serangan lebih cepat dari mekanisme *anomaly-based* dengan selisih waktu 2 menit 34 detik dan dengan selisih *request* sebanyak 180 *request*.

4. Dalam hal kemampuan mendeteksi serangan *Anomaly-based* dapat mendeteksi serangan dari *tools* yang belum pernah diketahui sebelumnya, sedangkan dalam *Signature-based* tidak dapat mendeteksi serangan yang belum pernah tersimpan di dalam tabel database signature. Hal ini di sebabkan *signature-base* tidak memiliki pola guna melakukan pencocokan paket, sehingga setelah dilakukan penambahan pola dari *tool* Skipfish, *signature-based* dapat mendeteksi serangan Skipfish dengan baik.
5. Performa dalam mendeteksi aktivitas *vulnerability scanning* metode *signature-based* lebih baik dari metode *anomaly-based,* akurasi *anomaly-based* hanya 0,5 sedangkan pada *signature-based* memiliki nilai 1, dalam perbandingan presisi kedua metode memiliki nilai yang sama yaitu 1, dan dalam hal sensitivitas *signature-based* lebih unggul dengan nilai 1, sedangkan *anomaly-based* hanya mendapatkan nilai 0,36.

5.1. Saran

1. Mekanisme yang dibangun baik *Anomaly-based* maupun *Signature-based* bekerja dengan *interval* 5 menit sekali, sehingga dalam waktu 5 menit tersebut serangan masih dapat dijalankan, untuk itu *interval* diperkecil menjadi 3 menit sehingga serangan dapat segera di antisipasi.

2. mekanisme *signature-based* harus selalu mendapatkan pembaharuan apabila telah terjadi serangan dari *tools* yang belum pernah ada di dalam table signature pada *database.*
3. Penelitian selanjutnya di harapkan dapat meningkatkan performa dari metode *anomaly-based*.
4. Penelitian ini memiliki kekurangan pada *testing* serangan VulScan yang dilakukan menggunakan *online* Pentest-*Tool* dimana serangan tersebut tidak dapat di deteksi oleh kedua mekanisme, masih belum diketahui penyabab dari temuan tersebut, oleh sebab itu akan di rekomendasikan pada penelitian selanjutnya.

DAFTAR PUSTAKA

BSSN, 2021 ,Monitoring keamanan siber, Laporan tahunan, BSSN , Depok Jawa Barat.

Malkawi, M., Özyer, T., & Alhajj, R. (2021, November). Automation of active reconnaissance phase: an automated API-based port and vulnerability scanner. In Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (pp. 622-629).

Risqiwati, D., & Irawan, E. A. (2018). Realtime Pencegahan Serangan Brute Force dan DDOS Pada Ubuntu Server. Techno. Com, 17(4), 347-354.

Prasetyo, S. E., & Lee, R. C. (2021, March). Analisis Keamanan Jaringan Pada Pay2home Menggunakan Metode Penetration *Testing*. In CoMBInES-Conference on Management, Business, Innovation, Education and Social Sciences (Vol. 1, No. 1, pp. 710-718).

Fadhlurrohman, M., Muliawati, A., & Hananto, B. (2021). Analisis Kinerja Intrusion Detection System pada Deteksi Anomali dengan Metode Decision Tree Terhadap Serangan Siber. Jurnal Ilmu Komputer dan Agri-Informatika, 8(2), 90-94.

Bahtiar, R. (2021). ANALISIS PERBANDINGAN DETECTION TRAFFIC ANOMALY DENGAN METODE NAIVE BAYES DAN DBSCAN. JATIMIKA: Jurnal Kreativitas Mahasiswa Informatika, 1(2).

Kumar, S., Jigyasu, S., & Singh, V. Network Traffic Monitor and Analysis Using Packet Sniffer.

Deris, S. Intrusion Prevention System (IPS) dan Tantangan dalam pengembangannya. FASILKOM, UNSRI, Palembang, Indonesia.

Khraisat, A., Gondal, I., Vamplew, P., & Kamruzzaman, J. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity, 2(1), 1-22.

Maseer, Z. K., Yusof, R., Bahaman, N., Mostafa, S. A., & Foozy, C. F. M. (2021). Benchmarking of machine learning for anomaly based intrusion detection systems in the CICIDS2017 dataset. IEEE access, 9, 22351-22370.

Haryani, P., & Raharjo, S. (2021). MANAJEMEN PADA JARINGAN MIKROTIK MENGGUNAKAN METODE HIERARCHICAL TOKEN BUCKET (HTB) DAN KEAMANAN FIREWALL INTRUSION DETECTION SYSTEM (IDS). Jurnal Jarkom, 9(1), 1-9.

Syahputra, W., Diansyah, T. M., & Liza, R. (2020). PEMANFAATAN MIKROTIK ROUTER BOARD SEBAGAI PENGAMAN SERANGAN DDOS MENGGUNAKAN METODE IDS. In SEMINAR NASIONAL TEKNOLOGI INFORMASI & KOMUNIKASI (Vol. 1, No. 1, pp. 492-499).

Hidayat, A., & Saputra, I. P. (2018). Analisa Dan Problem Solving Keamanan Router MikroTik Rb750Ra Dan Rb750Gr3 Dengan Metode Penetration *Testing* (Studi Kasus: Warnet Aulia. Net, Tanjung Harapan Lampung Timur). Jurnal RESISTOR (Rekayasa Sistem Komputer), 1(2), 118-124.

Widodo, T., & Aji, A. S. (2022). Pemanfaatan Network Forensic Investigation Framework untuk Mengidentifikasi Serangan Jaringan Melalui Intrusion Detection System (IDS). JISKA (Jurnal Informatika Sunan Kalijaga), 7(1), 46-55.

Sau, W. M. T., & Siswantyo, S. (2021). Analisis Penggunaan Hasil Deteksi IDS Snort pada *Tools* RITA dalam Mendeteksi Aktivitas Beacon. Info Kripto, 15(2), 97-104.

Muhammad, A. H. (2021). Penerapan Kombinasi Algoritma Caesar Cipher pada Block Acak dan Cipher Transposisi Dalam Mengamankan Pesan. Journal of Information Technology, 1(1), 22-28.

Gollmann, D. (2010). Computer security. Wiley Interdisciplinary Reviews: Computational Statistics, 2(5), 544-554..

Brinkley, D. L., & Schell, R. R. (1995). Concepts and terminology for computer security. Information security: An integrated collection of essays, 40-97.

Subandi, K., & Sugara, V. I. (2021). Analisa Serangan Vulnerabilities Terhadap Server Selama Periode WFH di Masa Pandemi Covid-19 Sebagai Prosedur Mitigasi. Prosiding Semnastek.

Chen, J., Chen, X., & Yu, B. (2021). Design of web vulnerability scanner based on go language. In MATEC Web of Conferences (Vol. 336, p. 08010). EDP Sciences.

Huang, H. C., Zhang, Z. K., Chen, C. K., Hong, W. D., Jao, J. C., & Shieh, S. (2022). Adaptive Entry Point Discovery for Web Vulnerability Scanning. Journal of Information Science & Engineering, 38(1).

Garcia-Teodoro, P., Diaz-Verdejo, J., Maciá-Fernández, G., & Vázquez, E. (2009). *Anomaly-based* network intrusion detection: Techniques, systems and challenges. computers & security, 28(1-2), 18-28.

Thakkar, A., & Lohiya, R. (2021). A survey on intrusion detection system: feature selection, model, performance measures, application perspective, challenges, and future research directions. Artificial Intelligence Review, 1-111.

Tekerek, A., Gemci, C., & Bay, O. F. (2014, October). Development of a hybrid web application firewall to prevent web based attacks. In 2014 IEEE 8th International Conference on Application of Information and Communication Technologies (AICT) (pp. 1-4). IEEE.

Otoum, Y., & Nayak, A. (2021). As-ids: Anomaly and signature based ids for the internet of things. Journal of Network and Systems Management, 29(3), 1-26.

Dodiya, B., & Singh, U. K. Malicious Traffic analysis using Wireshark by collection of Indicators of Compromise. International Journal of Computer Applications, 975, 8887.

Saroha, S. Restraining Packet Sniffing & Security: A Brief Overview.

Liang, J., & Kim, Y. (2022, January). Evolution of Firewalls: Toward Securer Network Using Next Generation Firewall. In 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC) (pp. 0752-0759). IEEE.

Anwar, R. W., Abdullah, T., & Pastore, F. (2021). Firewall Best Practices for Securing Smart Healthcare Environment: A Review. Applied Sciences, 11(19), 9183.

Clincy, V., & Shahriar, H. (2018, July). Web application firewall: Network security models and configuration. In 2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC) (Vol. 1, pp. 835-836). IEEE.

Imam, R. M., Sukarno, P., & Nugroho, M. A. (2019). Deteksi Anomali Jaringan Menggunakan Hybrid Algorithm. eProceedings of Engineering, 6(2).

Laabid, N. (2021). Botnet Command & Control Detection in IoT Networks (Master's thesis, Itä-Suomen yliopisto).

Saputra, I. P., Yusuf, R., & Saprudin, U. (2021). IMPLEMENTASI CLOUD COMPUTING SEBAGAI RADIUS SERVER PADA JARINGAN INTERNET ROUTER MIKROTIK. *Journal Computer Science and Information Systems: J-Cosys*, *1*(2), 81-86.

Bullock, J., & Parker, J. T. (2017). *Wireshark for Security Professionals: Using Wireshark and the Metasploit Framework*. John Wiley & Sons.

LAMPIRAN