

**ANALISA PENERAPAN DMARC YANG DIINTEGRASIKAN  
DENGAN ANTI SPAM DAN ANTI VIRUS UNTUK  
PENGAMANAN MAIL SERVER**

**SKRIPSI**



**PROGRAM STUDI ILMU KOMPUTER  
FAKULTAS TEKNIK DAN DESAIN  
UNIVERSITAS BUMIGORA  
MATARAM  
2021**

**ANALISA PENERAPAN DMARC YANG DIINTEGRASIKAN  
DENGAN ANTI SPAM DAN ANTI VIRUS UNTUK  
PENGAMANAN MAIL SERVER**

**SKRIPSI**



Diajukan Sebagai Salah Satu Syarat untuk Memenuhi Kebulatan Studi  
Jenjang Strata Satu (S1) Program Studi Ilmu Komputer  
Pada Universitas Bumigora

**Oleh:**

**RUDI KURNIAWAN**  
**1710510157**

**PROGRAM STUDI ILMU KOMPUTER**  
**FAKULTAS TEKNIK DAN DESAIN**  
**UNIVERSITAS BUMIGORA**  
**MATARAM**  
**2021**

**ANALISA PENERAPAN DMARC YANG DIINTEGRASIKAN  
DENGAN ANTI SPAM DAN ANTI VIRUS UNTUK  
PENGAMANAN MAIL SERVER**

**SKRIPSI**

Diajukan Sebagai Salah Satu Syarat untuk Memenuhi Kebulatan Studi  
Jenjang Strata Satu (S1) Program Studi Ilmu Komputer  
Pada Universitas Bumigora

Oleh:

**RUDI KURNIAWAN  
1710510157**

19 87

**B G Dekan Fakultas Teknik dan Desain**

**Ahmat Adil, S.Kom., M.Sc.  
NIK. 96.6.63**



UNIVERSITAS BUMIGORA

Universitas Bumigora  
Pendidikan dan Kebudayaan  
Kampus Cikarang Selatan  
Jl. Raya Cikarang Selatan Km. 10  
Cikarang Selatan, Bekasi 17521

## SKRIPSI

ANALISA PENERAPAN DMARC YANG  
DIINTEGRASIKAN DENGAN ANTI SPAM DAN  
ANTI VIRUS UNTUK PENGAMANAN MAIL  
SERVER

NAMA : Rudi Kurniawan

NIM : 1710510157

NPM : 17.8.349.74.75.0.5.0157

PROGRAM STUDI : Ilmu Komputer

JENJANG : Strata Satu (S1)

DIUJIKAN : Senin, 9 Agustus 2021

Menyetujui,

Khairan Marzuki, S.T, M.Kom

Pembimbing I

Tanggal menyetujui 18-8-2021

Lilik Widyawati, M.Kom

Pembimbing II

Tanggal menyetujui 18/8/21

Telah diterima dan disetujui sebagai salah satu syarat untuk memperoleh  
Gelar Akademik Strata Satu Komputer (S.Kom).

Mengetahui :

Lilik Widyawati, M.Kom

Ketua Program Studi S1 Ilmu Komputer

Tanggal Mengetahui: 18/8/21

**ANALISA PENERAPAN DMARC YANG DIINTEGRASIKAN  
DENGAN ANTI SPAM DAN ANTI VIRUS UNTUK  
PENGAMANAN MAIL SERVER**

**LEMBAR PENGESAHAN PENGUJI**

Diajukan Sebagai Salah Satu Syarat untuk Memenuhi Kebulatan Studi  
Jenjang Strata Satu (S1) Program Studi Ilmu Komputer  
Pada Universitas Bumigora

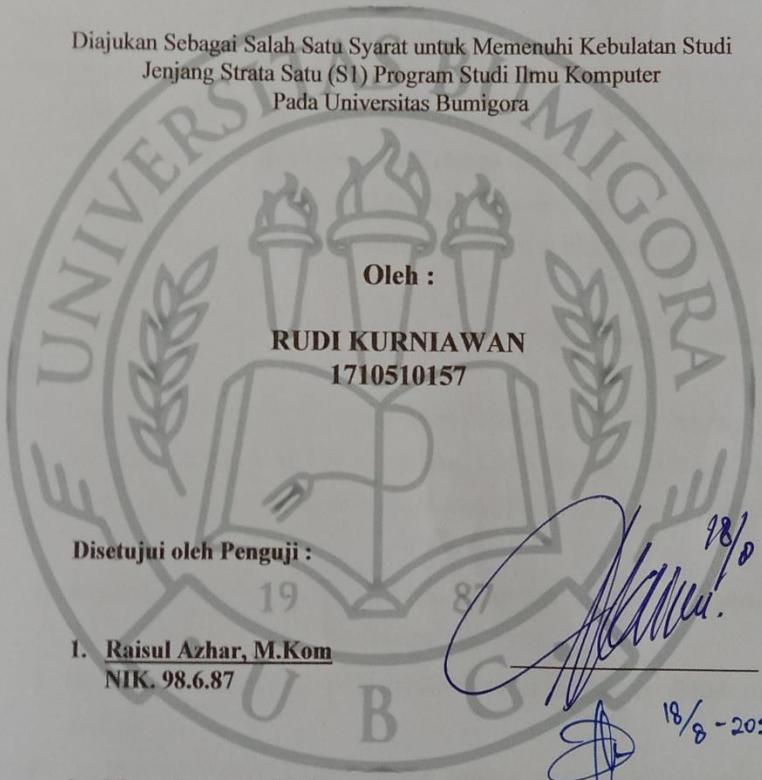
Oleh :

**RUDI KURNIAWAN**  
**1710510157**

Disetujui oleh Penguji :

1. Raisul Azhar, M.Kom  
NIK. 98.6.87

2. I Putu Haryadi, M.Kom  
NIK. 09.6.124

  
1987-2021  
Raisul Azhar, M.Kom  
I Putu Haryadi, M.Kom

18/8 - 2021

## KATA PENGANTAR

Dengan nama Tuhan Yang Maha Pengasih lagi Maha Penyayang, penulis panjatkan puji syukur atas rahmatnya, nikmat kebahagian, serta seluruh anugerah dalam bentuk apapun yang telah dilimpahkan kepada seluruh makhluk hidup, penulis dapat menyelesaikan Skripsi yang berjudul “**ANALISA PENERAPAN DMARC YANG DIINTEGRASIKAN DENGAN ANTI SPAM DAN ANTI VIRUS UNTUK PENGAMANAN MAIL SERVER**”

Dengan selesainya Skripsi ini, penulis ingin mengucapkan terimakasih kepada pihak-pihak yang telah banyak membantu dalam penyelesaian Skripsi ini. Dalam kesempatan ini penulis ingin menyampaikan ucapan terima kasih kepada:

1. Kedua Orang Tua yang telah memberikan dukungan berupa Bimbingan, Materi dan Doa. Tidak Terlupakan Keluarga Besar yang telah Memberikan Semangat dan Doa untuk menyelesaikan Tugas Akhir ini.
2. Bapak Dr.Ir. Anthony Anggrawan, MT., Ph.D. selaku Rektor Universitas Bumigora.
3. Ibu Ni Gusti Ayu Dasriani, M.Kom, selaku Wakil Rektor I Universitas Bumigora.
4. Bapak Ahmat Adil, M.Sc, selaku Dekan Fakultas Teknik dan Desain.
5. Ibu Lilik Widyawati., M.Kom selaku Ketua Program Studi S1 Ilmu Komputer dan Selaku Pembimbing Kedua dalam membantu mengerjakan Skripsi ini.
6. Bapak Khairan Marzuki S.T, M.Kom, selaku dosen pembimbing pertama dalam membantu mengerjakan Skripsi ini.
7. Bapak/Ibu dosen yang telah memberikan ilmu selama dalam masa perkuliahan.
8. Teman-teman serta sahabat-sahabat penulis yang selalu memberikan semangat kepada penulis dalam menyelesaikan Skripsi ini.

Semoga Skripsi ini dapat memberikan manfaat yang sebesar-besarnya pada kita semua. Sebagai manusia biasa yang mempunyai keterbatasan dan

kekurangan, maka penulis menyadari bahwa Skripsi ini masih banyak kekurangan-kekurangan baik dalam teknik penulisan, pembahasan dan penyajian untuk itu penulis senantiasa mengharapkan teguran, kritik serta saran yang sifatnya membangun untuk dapat lebih sempurnanya pembuatan Skripsi atau sejenisnya pada masa-masa yang akan datang. Akhirnya penulis berharap semoga tulisan ini bermanfaat bagi semua pihak, terimakasih.

**Mataram, 13 Juni 2021**

**Penulis,**





### LEMBAR PERNYATAAN KEASLIAN

Saya bertanda tangan dibawah ini:

Nama : Rudi Kurniawan

NIM : 1710510157

Program studi : Ilmu Komputer

Kompetensi : Jaringan Komputer

Menyatakan bahwa skripsi yang berjudul:

#### **ANALISA PENERAPAN DMARC YANG DIINTEGRASIKAN DENGAN ANTI SPAM DAN ANTI VIRUS UNTUK PENGAMANAN MAIL SERVER**

Benar-benar merupakan hasil karya pribadi dan seluruh sumber yang dikutip maupundirujuk telah saya nyatakan dengan benar dan apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik sesuai dengan aturan yang berlaku.

Mataram, 13 Juni 2021  
  
RUDI KURNIAWAN  
NIM. 1710510157

## **IZIN PENGGUNAAN**

Skripsi ini merupakan syarat kelulusan pada Program Studi S1 Ilmu Komputer Universitas Bumigora, dengan ini penulis setuju jika Skripsi ini digandakan (diduplikasi) baik sebagian maupun seluruhnya, ataupun dikembangkan untuk kepentingan akademis yang disetujui oleh pembimbing penulis, Ketua Program Studi, Dekan Fakultas Teknik dan Desain.

Untuk dimaklumi, bahwa menduplikasi, mempublikasikan atau menggunakan Tugas Akhir ini, maupun bagian-bagiannya dengan tujuan komersional / keuntungan finansial, tidak diizinkan tanpa adanya izin tertulis dari Universitas Bumigora. Jika hal ini dilanggar maka Universitas Bumigora akan memberikan sanksi sesuai dengan hukum yang berlaku.

Penghargaan akademis terkait isi dari skripsi ini adalah pada penulis dan Universitas Bumigora.

Permintaan izin untuk menduplikasi atau menggunakan materi dari Skripsi ini baik sebagian maupun seluruhnya harus ditunjukan pada:

Dekan Fakultas Teknik dan Desain  
Ketua Program Studi S1 Ilmu Komputer  
Universitas Bumigora

## ABSTRAK

*Email spam, email spoofing, dan virus yang didistribusikan melalui email merupakan hal yang tidak diinginkan oleh pengguna email. Email spam akan sangat mengganggu pengguna email dan akan menghabiskan banyak sumber daya mail server. Email spoofing merupakan tindakan kejahatan yang memanfaatkan email sebagai sarana untuk melakukan penipuan. Sedangkan virus yang didistribusikan melalui email biasanya dikirimkan oleh pihak yang tidak bertanggung jawab yang bertujuan untuk menginfeksi mail server ataupun komputer pengguna email. Email spam, email spoofing, dan email yang mengandung virus dapat menimbulkan kerugian yang sangat besar baik bagi penyedia layanan email maupun bagi pengguna email. Berdasarkan latar belakang tersebut maka mendorong penulis untuk menganalisa penerapan protocol DMARC, Anti Spam, dan Anti Virus sehingga mail server dapat terhindar dari email spam, virus dan pengguna email dapat terhindar dari aktifitas spoofing.*

Perancangan dan analisa penerapan Protocol DMARC, anti spam, dan anti virus ini menggunakan metodologi NDLC, yaitu metode pengembangan jaringan komputer yang diawali dengan menganalisa artikel ilmiah, buku, dan jurnal untuk mendapatkan informasi mengenai DMARC, anti spam, dan anti virus. Merancang sistem filtering email spam, spoofing, dan virus, melakukan simulasi instalasi dan konfigurasi. Tahap berikutnya adalah implementasi dimana pada tahap ini dilakukan penerapan sistem yang telah dirancang sebelumnya dan melakukan uji coba pada sistem filtering email spam, spoofing, dan virus. Tahapan yang terakhir adalah tahap monitoring dimana akan dilakukan pengawasan terhadap sistem yang telah dibuat untuk mengetahui tingkat keberhasilan sistem yang telah dibuat.

Hasil atau keluaran yang akan dicapai yaitu mail server dapat terhindar dari email spam, email spoofing, dan virus untuk memastikan keamanan dan kenyamanan pengguna email serta menghindari dampak kerugian yang dapat ditimbulkan oleh email spam, email spoofing, dan virus.

Kesimpulan dari penelitian ini adalah penerapan protokol *Domain-based Message Authentication, Reporting & Conformance (DMARC)* dapat mencegah email spoofing dengan cara melakukan otentikasi menggunakan metode pencocokan *private key* dan *public key (Asymmetric keys)*. Sedangkan penerapan protokol *Sender Policy Framework* dapat mencegah email spoofing dengan cara melakukan otorisasi menggunakan metode pencocokan alamat IP server pengirim. Sebaliknya penerapan *SpamAssassin*, *ClamAV*, dan *Amavisd-New* dapat mencegah masuknya email spam dan virus dengan cara melakukan pengecekan *header*, *body*, dan *attachment email*.

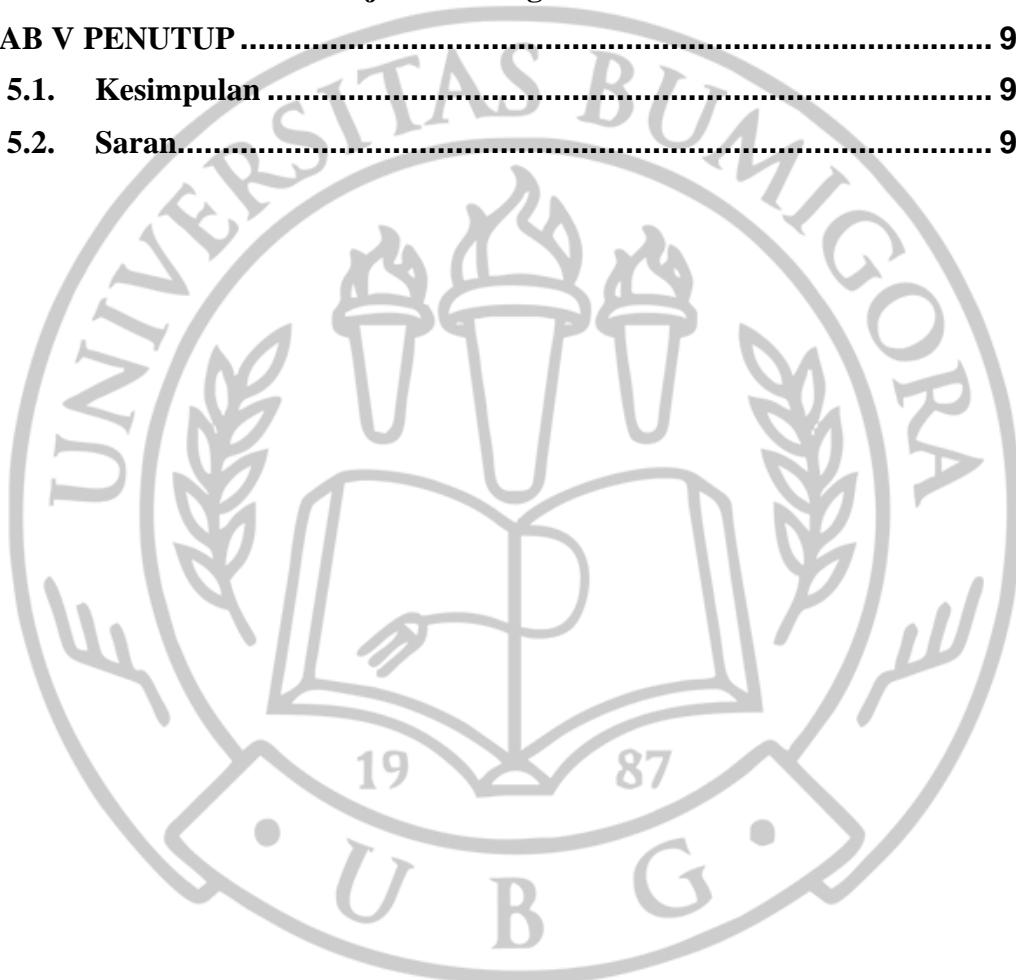
**Kata Kunci:** Dmarc Anti Spam Dan Anti Virus Untuk Pengamanan Mail Server

## DAFTAR ISI

<b>KATA PENGANTAR .....</b>	<b>v</b>
<b>ABSTRAK.....</b>	<b>vi</b>
<b>DAFTAR TABEL.....</b>	<b>xii</b>
<b>BAB I PENDAHULUAN .....</b>	<b>1</b>
1.1.    Latar Belakang.....	1
1.2.    Perumusan Masalah .....	3
1.4.    Tujuan dan Manfaat Penulisan.....	4
1.4.1.    Tujuan .....	4
1.4.2.    Manfaat .....	4
1.5.    Metodologi penelitian .....	5
1.6.    Sistematika Penulisan .....	7
<b>BAB II LANDASAN TEORI .....</b>	<b>9</b>
2.1    Jaringan Komputer .....	9
2.2    Model Lapisan <i>OSI</i> .....	9
2.3 <i>Transmission Control Protocol / Internet Protocol (TCP/IP)</i> .....	13
2.4    Keamanan Jaringan Komputer.....	15
2.5    Jenis-jenis Layanan Keamanan Jaringan .....	15
2.6 <i>Email Spoofing</i> dan <i>Phising</i> .....	17
2.7 <i>Server</i> .....	17
2.8    Linux.....	17
2.9    Linux <i>CentOS</i> .....	18
2.10 <i>Centos Web Panel</i> .....	18
2.11    Surat Elektronik.....	19
2.12 <i>Mail Server</i> .....	19
2.13 <i>Mail Protocol</i> .....	21
2.14 <i>Postfix</i> .....	22
2.15 <i>Dovecot</i> .....	23
2.16 <i>Roundcube</i> .....	23
2.17 <i>Domain Name System (DNS)</i> .....	23
2.18 <i>DNS Server</i> .....	23

<b>2.19</b>	<i>Bind9</i> .....	<b>24</b>
<b>2.20</b>	<i>HTTP</i> .....	<b>24</b>
<b>2.21</b>	<i>HTTP Server</i> .....	<b>25</b>
<b>2.22</b>	<i>Apache HTTP Server</i> .....	<b>25</b>
<b>2.23</b>	<i>Email Spam</i> .....	<b>25</b>
<b>2.24</b>	<i>Spam Filter</i> .....	<b>25</b>
<b>2.25</b>	<i>SpamAssassin, ClamAV, dan Amavisd-New</i> .....	<b>26</b>
<b>2.26</b>	<i>DomainKeys Identified Mail (DKIM) dan OpenDKIM</i> .....	<b>27</b>
<b>2.27</b>	<i>Sender Policy Framework (SPF)</i> .....	<b>28</b>
<b>2.28</b>	<i>Domain-Based Message Authentication, Reporting &amp; Conformance (DMARC)</i> .....	<b>29</b>
<b>2.29</b>	<i>Gmail</i> .....	<b>31</b>
<b>2.30</b>	<i>Emkei's Mailer</i> .....	<b>31</b>
<b>2.31</b>	<i>Yahoo! Mail</i> .....	<b>31</b>
<b>BAB III METODOLOGI DAN PERANCANGAN</b> .....		<b>32</b>
<b>3.1.</b>	<i>Tahap Analisa (Analysis)</i> .....	<b>32</b>
<b>3.1.1.</b>	<i>Pengumpulan Data</i> .....	<b>32</b>
<b>3.1.2.</b>	<i>Analisa Data</i> .....	<b>34</b>
<b>3.2.</b>	<i>Tahap Desain (Design)</i> .....	<b>35</b>
<b>3.2.1</b>	<i>Rancangan Sistem Filtering Email Spam, Virus dan Spoofing</i> ..	<b>35</b>
<b>3.2.2</b>	<i>Rancangan jaringan Uji coba</i> .....	<b>37</b>
<b>3.2.3</b>	<i>Rancangan Pengalamatan IP</i> .....	<b>38</b>
<b>3.2.4</b>	<i>Rancangan Akun Email</i> .....	<b>38</b>
<b>3.2.5</b>	<i>Kebutuhan Perangkat Keras dan Perangkat Lunak</i> .....	<b>38</b>
<b>3.3.</b>	<i>Tahap Simulasi (Prototyping)</i> .....	<b>40</b>
<b>3.3.1.</b>	<i>Instalasi Dan Konfigurasi</i> .....	<b>41</b>
<b>3.3.2.</b>	<i>Uji Coba</i> .....	<b>41</b>
<b>BAB IV HASIL DAN PEMBAHASAN</b> .....		<b>42</b>
<b>4.1</b>	<i>Hasil Instalasi Dan Konfigurasi</i> .....	<b>42</b>
<b>4.1.1</b>	<i>Hasil Instalasi Dan Konfigurasi Server</i> .....	<b>42</b>
<b>4.1.5</b>	<i>Hasil Konfigurasi Client</i> .....	<b>51</b>
<b>4.2</b>	<i>Hasil Uji Coba</i> .....	<b>51</b>

4.2.1	Verifikasi Konfigurasi .....	51
4.3	Skenario Uji Coba.....	56
4.4	Analisa Hasil Uji Coba.....	<b>87</b>
4.4.1	Analisa Hasil Uji Coba Pengiriman <i>Email Spoofing</i> .....	88
4.4.2	Analisa Hasil Uji Coba Pengiriman <i>Email Spam</i> .....	89
4.4.3	Analisa Hasil Uji Coba Mengirim <i>Email</i> Mengandung Virus...	90
4.4.4	Analisa Hasil Uji Coba Pengecekan <i>Header Email</i> .....	90
<b>BAB V</b>	<b>PENUTUP .....</b>	<b>93</b>
5.1.	Kesimpulan .....	93
5.2.	Saran.....	93



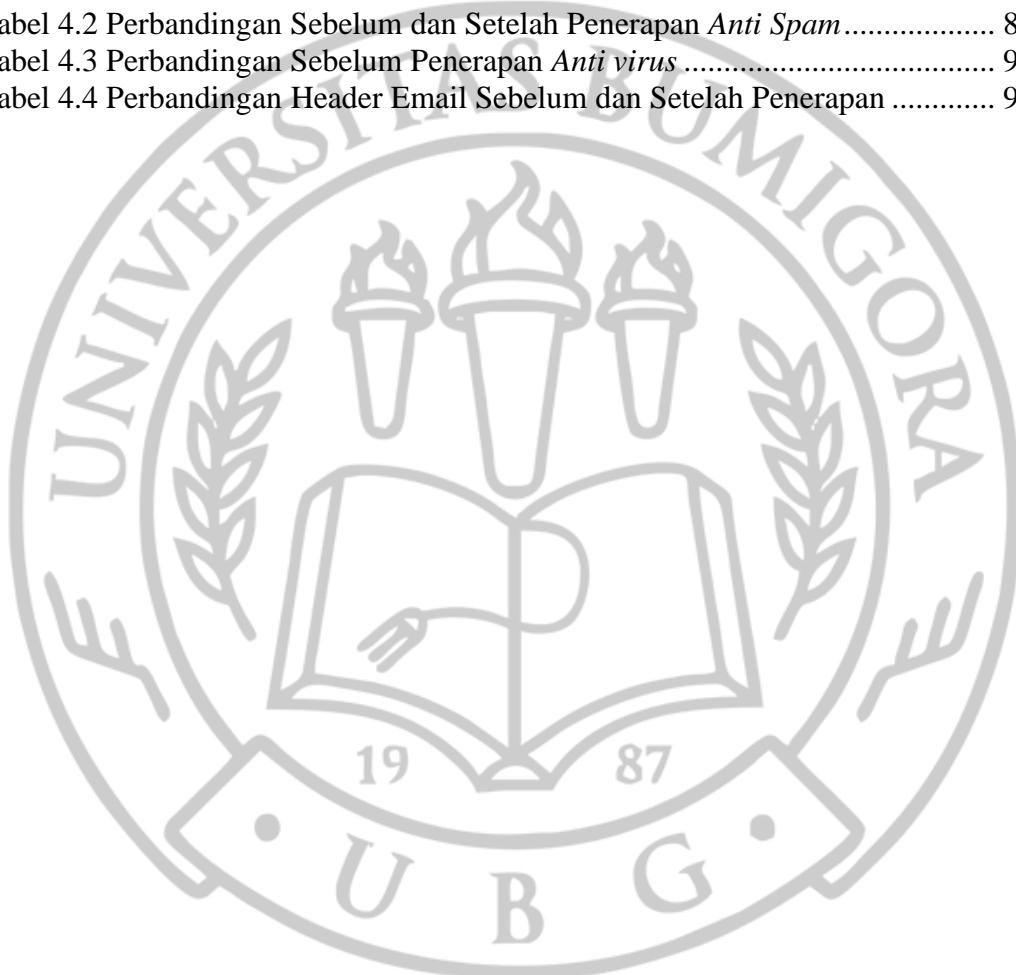
## DAFTAR GAMBAR

Gambar 2.1 Layer OSI .....	10
Gambar 2.2 Perbandingan Layer TCP/IP dan Layer OSI.....	14
Gambar 2.3 Proses pengiriman email .....	21
Gambar 2.4 Cara Kerja SpamAssassin, ClamAV, dan Amavisd-New.....	27
Gambar 2.5 Cara Kerja DKIM.....	28
Gambar 2.6 Cara Kerja SPF.....	29
Gambar 2. 7 Cara Kerja DMARC.....	30
Gambar 3.1 Rancangan Sistem Filtering Email Spam, Spoofing, dan Virus .....	35
Gambar 3. 2 Rancangan Topologi Uji Coba.....	37
Gambar 4.1 Linux CentOS release 7-9 .....	42
Gambar 4.2 Hasil Instalasi CWP .....	43
Gambar 4.3 Konfigurasi Interface.....	43
Gambar 4.4 Konfigurasi Name Server.....	44
Gambar 4.5 Konfigurasi Domain .....	45
Gambar 4.6 File skripsirudi.my.id.db .....	46
Gambar 4. 7 File 207.41.103.in-addr.arpa.db .....	46
Gambar 4.8 File named.conf .....	47
Gambar 4.9 Membuat Akun Email .....	48
Gambar 4.10 Instalasi DKIM, SPF, Anti Spam, dan Anti Virus .....	48
Gambar 4.11 Menambah DKIM Record pada File Zone.....	49
Gambar 4.12 Menambah SPF Record pada File Zone.....	49
Gambar 4.13 DKIM dan SPF Record .....	49
Gambar 4.14 Konfigurasi File TrustedHosts .....	50
Gambar 4.15 Konfigurasi File main.cf .....	50
Gambar 4.16 Terhubung ke Internet .....	51
Gambar 4.17 Verifikasi Konfigurasi DNS Server .....	52
Gambar 4.18 Verifikasi Konfigurasi Mail Server.....	53
Gambar 4.19 Mengirim Email pada User Email Local.....	53
Gambar 4.20 Mengirim Email Pada Mail Server Lain .....	54
Gambar 4.21 Verifikasi Fungsi DKIM, SPF, dan DMARC .....	54
Gambar 4.22 Verifikasi Fungsi ClamAV.....	54
Gambar 4.23 Ping Mail Server.....	55
Gambar 4.24 Akses MUA roundcube .....	55
Gambar 4. 25 Emke'I Fake mailer .....	57
Gambar 4. 26 Email spoofing .....	58
Gambar 4. 27 mengirim spoofing ke yahoo mail.....	59
Gambar 4. 28 Email Spoofing Terkirim ke Yahoo! Mail .....	60
Gambar 4. 29 mengirim spoofing ke skripsirudi.my.id .....	60
Gambar 4. 30 Email Spoofing Terkirim ke User skripsirudi.my.id.....	61
Gambar 4. 31 Mengirim Email Spam dari skripsirudi.my.id.....	61
Gambar 4. 32 email spam dari skripsirudi.my.id terkirim .....	62
Gambar 4. 33 Mengirim Email Spam dari Yahoo! Mail .....	62
Gambar 4. 34 Email Spam dari Yahoo! Mail Terkirim .....	63

Gambar 4. 35 Mengirim Email Spam dari Gmail .....	63
Gambar 4. 36 Email Spam dari Gmail Terkirim.....	63
Gambar 4. 37 EICAR Test dari skripsirudi.my.id .....	64
Gambar 4. 38 Email Mengandung Virus dari skripsirudi.my.id Terkirim.....	65
Gambar 4. 39 EICAR Test dari Yahoo! Mail .....	65
Gambar 4. 40 Email Mengandung Virus dari Yahoo! Mail Terkirim .....	66
Gambar 4. 41 EICAR Test dari Gmail.....	66
Gambar 4. 42 Email Mengandung Virus dari Gmail Terkirim.....	67
Gambar 4. 43 Cuplikan Header Email pada Gmail Sebelum Penerapan .....	68
Gambar 4. 44 Cuplikan Header Email pada Yahoo! Mail Sebelum Penerapan ...	68
Gambar 4. 45 Cuplikan Header Email pada skripsi Sebelum Penerapan .....	69
Gambar 4. 46 Private Key pada skripsirudi.my.id .....	71
Gambar 4.47 Public Key pada DNS Server skripsirudi.my.id.....	71
Gambar 4.48 Cuplikan Header Email .....	72
Gambar 4.49 SPF Record pada skripsirudi.my.id.....	72
Gambar 4. 50 DMARC Record pada skripsirudi.my.id .....	72
Gambar 4. 51 terindikasi spam oleh gmail.....	73
Gambar 4. 52 Email Spoofing Masuk ke Folder Spam .....	73
Gambar 4. 53 Email Spoofing Masuk pada Folder Inbox .....	74
Gambar 4. 54 Mengirim Email Spam dari skripsirudi.my.id Setelah Penerapan .	75
Gambar 4. 55 Email dari skripsirudi.my.id Terindikasi Spam .....	75
Gambar 4. 56 Mengirim Email Spam dari Yahoo! Mail Setelah Penerapan .....	76
Gambar 4. 57 Email dari Yahoo! Mail Terindikasi Spam .....	76
Gambar 4. 58 Mengirim Email Spam dari Gmail Setalah Penerapan.....	77
Gambar 4. 59 Email dari Gmail Terindikasi Spam.....	77
Gambar 4. 60 Email Dengan Format Spam .....	78
Gambar 4. 61 Email Terindikasi Sebagai Spam oleh Yahoo! Mail .....	78
Gambar 4. 62 header dmarc .....	79
Gambar 4. 63 hasil report DMARC setelah protocol di terapkan.....	79
Gambar 4. 64 EICAR Test dari skripsirudi.my.id Setelah Penerapan .....	80
Gambar 4. 65 email local dari skripsirudi.my.id terblok .....	81
Gambar 4. 66 report email yang mengandung virus .....	81
Gambar 4. 67 EICAR Test dari Yahoo! Mail Setelah Penerapan.....	82
Gambar 4. 68 email dari rudi.masterqq3@yahoo.com ke skripsirudi.my.id terblok .....	82
Gambar 4. 69 report email yang mengandung virus .....	82
Gambar 4. 70 EICAR Test dari Gmail Setelah Penerapan .....	83
Gambar 4. 71 email di blok oleh antivirus client .....	84
Gambar 4. 72 report email yang mengandung virus .....	84
Gambar 4. 73 Cuplikan Header Email pada Gmail Setelah Penerapan .....	85
Gambar 4. 74 Cuplikan Header Email pada Yahoo! Mail Setelah Penerapan.....	86
Gambar 4. 75 Cuplikan Header Email skripsirudi.my.id Setelah Penerapan .....	87

## DAFTAR TABEL

Tabel 3. 1 Jurnal Ilmiah Tentang <i>Email Spam, Spoofing, dan Virus</i> .....	32
Tabel 3.2 Pengalamatan <i>IP</i> .....	38
Tabel 3.3 Kebutuhan Akun <i>Email</i> .....	38
Tabel 3.4 Spesifikasi <i>VPS</i> .....	39
Tabel 3.5 Spesifikasi <i>Client</i> .....	39
Tabel 4. 1 Perbandingan Sebelum dan Setelah Penerapan protocol <i>DMARC, DKIM</i> dan <i>SPF</i> .....	88
Tabel 4.2 Perbandingan Sebelum dan Setelah Penerapan <i>Anti Spam</i> .....	89
Tabel 4.3 Perbandingan Sebelum Penerapan <i>Anti virus</i> .....	90
Tabel 4.4 Perbandingan Header Email Sebelum dan Setelah Penerapan .....	91



# **BAB I**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Perkembangan teknologi saat ini sudah begitu pesat sehingga teknologi dapat memudahkan pekerjaan manusia hampir di segala bidang, surat elektronik adalah salah satu dari kemajuan teknologi dalam bidang komunikasi sehingga fungsi dari surat dapat digantikan dengan adanya surat elektronik, efisiensi biaya dan waktu menjadi alasan yang membuat banyak orang beralih dari surat menuju surat elektronik.

Mengingat betapa pentingnya media komunikasi di zaman sekarang ini maka beberapa orang melakukan penelitian terutama di bidang keamanan jaringan. menyebutkan bahwa salah satu layanan internet yang banyak digunakan adalah email. Email merupakan surat elektronik yang berbasis file teks, namun dengan perkembangan teknologi, email lebih atraktif terhadap penggunanya, tidak hanya dapat mengirim file teks, tetapi juga dapat mengirim file audio, video, foto dan file ekstensi lainnya. Terdapat ancaman serius mengiringi kemudahan yang diberikan oleh email dengan memanfaatkan email sebagai media untuk melakukan tindak kejahatan di dunia siber, karena email merupakan alat transportasi utama bagi spam, virus dan malware dalam jaringan (Hoiriyah, Sugiantoro, and Prayudi 2016). penerapan protokol DomainKeys Identified Mail dapat mencegah email spoofing dengan cara melakukan otentifikasi menggunakan metode pencocokan private key dan public key (Asymmetric keys). Sedangkan penerapan protokol Sender Policy Framework dapat mencegah email spoofing dengan cara melakukan otorisasi menggunakan metode pencocokan alamat IP server pengirim. Hasil atau keluaran yang dicapai yaitu mail server dapat terhindar dari email spam, email spoofing, dan virus untuk memastikan keamanan dan kenyamanan pengguna email serta menghindari dampak kerugian yang dapat ditimbulkan oleh email spam,

email spoofing, dan virus (Hanif 2018). pendeksi spoofing pada email menggunakan penerapan *DKIM*, *SPF* dan *DMARC* yang pada penelitian di gunakan Sebuah metode untuk melakukan deteksi diperlukan untuk melihat apakah sebuah email terindikasikan sebagai *spoof* atau tidak, (Nadzifan, Nazihullah, and . 2018). Forensik email dengan metode Header Analysis dianggap efektif untuk melacak alamat IP pengirim email, namun hal ini tidak dapat melacak posisi pengirim email secara akurat. Mengintegrasikan email forensik klasik dengan data mining dari Twitter data stream telah terbukti efektif untuk mendapatkan informasi geografis dan memperkecil luas dari seluas kota menjadi seluas lingkungan, yang sangat berharga bagi pihak berwajib dalam menghemat waktu dan juga usaha untuk mengadili pelaku tindak kejahatan cyber, (Ardhi 2020).

Dari kutipan di atas ada beberapa kekurangan seperti *DKIM* memiliki masalah yang tidak dapat menentukan apakah tanda tangan itu sah dan juga tidak dapat memberi laporan apabila terjadi pemalsuan email. Pertimbangan ini lah yang membuat penulis untuk menerapkan *Protocol DMARC* yang berfungsi untuk mendeteksi email palsu dan memberi tahu pengguna tanpa *DKIM* tanda tangan dengan memanfaatkan *DMARC* dan menerapkan sistem itu mengirimkan hasil verifikasi *DMARC* ke penerima, *ClamAV* sebagai tools *anti spam* dan *spoofing* yang dapat melakukan otorisasi bukan hanya melalui alamat IP saja namun juga dapat melalui URL dan antivirus *ClamAV* untuk mengatasi virus yang sangat tidak diinginkan oleh pengguna maupun penyedia layanan email. Sistem pencegahan *email spam*, *spoofing*, dan *virus* diharapkan dapat mengurangi dampak kerugian yang diakibatkan oleh email spam, spoofing, dan virus.

*DMARC* (*Domain-based Message Authentication, Reporting and Conformance*) dapat digunakan sebagai otentikasi dan otorisasi email sehingga email client akan terbebas dari tindakan *spoofing*. Penerapan Anti Spam dan Anti Virus *ClamAV* juga diperlukan agar email server terhindar dari email spam dan virus, metode yang diterapkan oleh Anti Spam dan Anti

Virus *ClamAV* yaitu dengan melakukan pengecekan header, body, dan attachment email kemudian disampaikan ke pengguna.

Manfaat dari penerapan *DMARC*, Anti *Spam* dan Anti Virus *ClamAV* adalah untuk mengoptimalkan sistem keamanan jaringan server mail, dengan cara memblokir surat elektronik yang dianggap sebagai spam atau virus, meningkatkan kualitas keamanan surat elektronik sehingga pengguna dapat terhindar dari aktivitas spoofing dan virus yang disisipkan melalui surat elektronik (Hanif 2018).

### **1.2. Perumusan Masalah**

Sesuai dari latar belakang yang telah dipaparkan di atas maka rumusan masalah yang akan dikaji adalah bagaimana menganalisa penerapan *DMARC* (*Domain-based Message Authentication, Reporting and Conformance*), Anti *Spam*, dan Anti Virus *ClamAV* pada mail server agar mail server dapat terhindar dari email spam, virus dan pengguna email dapat terhindar dari aktivitas *spoofing*, (Hanif 2018).

### **1.3. Batasan Masalah**

Batasan masalah yang digunakan dalam penyusunan skripsi ini untuk menjadikan pembahasan menjadi lebih terarah dan fokus adalah sebagai berikut:

1. Rancangan uji coba diimplementasikan menggunakan VPS yang disewa pada penyedia layanan VPS. Pada VPS akan dilakukan instalasi CentOS Web Panel, konfigurasi DNS server, konfigurasi Mail server, dan komputer client digunakan untuk mengakses Mail User Agent berbasis web (Zimbra).
2. Sistem operasi VPS yang digunakan adalah CentOS 7-9.
3. Aplikasi yang digunakan untuk memudahkan instalasi dan konfigurasi server adalah CentOS Web Panel.
4. Aplikasi MTA yang digunakan adalah Postfix untuk mengirim email.

5. Aplikasi MDA yang digunakan adalah Dovecot untuk menerima email.
6. Aplikasi MUA yang digunakan adalah Roundcube sebagai aplikasi email di sisi pengguna.
7. Aplikasi DNS server yang digunakan adalah bind9 agar email server dapat diakses menggunakan nama domain
8. Aplikasi HTTP server yang digunakan adalah Apache agar Mail Transfer Agent berbasis web dapat diakses melalui browser.
9. Pengujian yang dilakukan dengan mengirim surat elektronik yang terindikasi sebagai spam, kemudian melakukan pengiriman email spoofing, dan email yang mengandung virus, serta mengecek header setelah penerapan DMARC, anti spam, dan anti virus.
10. Pengujian DMARC dilakukan dengan cara mengirim email spoofing menggunakan Emkei's Fake Mailer kemudian email spoofing tersebut dikirim ke Gmail dan Yahoo! Mail.
11. Pengujian Anti Spam dan Anti Virus dilakukan dengan cara mengirim email spam dan email yang mengandung virus ke mail server.
12. Hasil report DMARC di kirim ke email root@skripsi.rudi.my.id.

## **1.4. Tujuan dan Manfaat Penulisan**

### **1.4.1. Tujuan**

Adapun tujuan dari penulisan skripsi ini adalah untuk menganalisa Penerapan Protocol DMARC, Anti Spam dan Anti Virus agar mail server terhindar dari spam dan virus dilakukan dengan cara mengirim email yang mengandung spam dan virus ke mail server.

### **1.4.2. Manfaat**

Adapun manfaat dari penelitian ini adalah sebagai berikut:

1. Bagi Diri Sendiri

- a. Dapat menambah ilmu pengetahuan pengetahuan baru yang dapat di terapkan di dunia kerja.
- b. Dapat menjadi tempat untuk mengimplementasikan ilmu pengetahuan yang telah didapat selama berada dibangku perkuliahan.
- c. Sebagai syarat untuk menyelesaikan jenjang pendidikan Strata 1 (S1) pada program studi Teknik Informatika di Universitas Bumigora Mataram.

## 2. Bagi Keilmuan

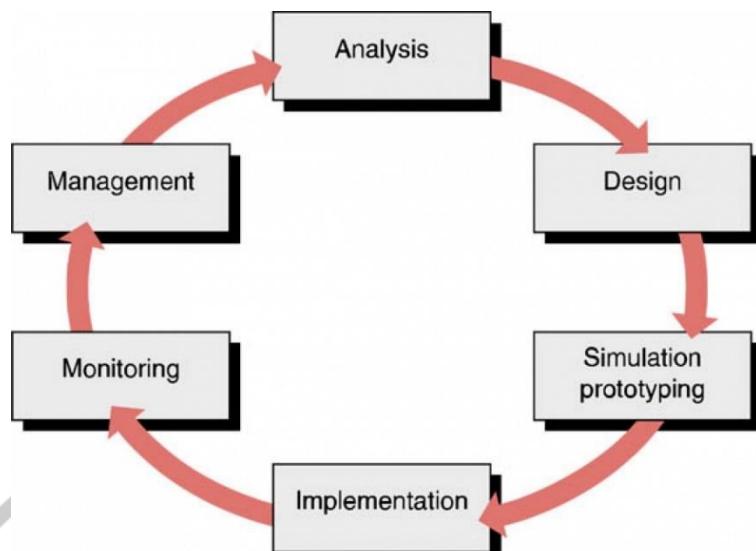
- a. Dapat menjadi bahan rujukan untuk pengembangan penelitian berikutnya terutama dalam bidang yang sama.
- b. Dapat menjadi sarana untuk melatih kemampuan dalam menulis karya ilmiah.

## 3. Bagi Masyarakat

- a. Dapat memberikan pengetahuan terkait dengan analisa penanganan *email spam*, *virus* dan aktifitas *spoofing* menggunakan *Protocol DMARC*, *Anti Spam*, dan *Anti Virus*.
- b. Dapat memberikan solusi cara penerapan *Protocol DMARC*, *Anti Spam*, dan *Anti Virus* pada *mail server*.

### 1.5. Metodologi penelitian

(Puspita et al. 2015) menyebutkan bahwa *Network Development Life Cycle* adalah suatu metode yang digunakan dalam mengembangkan atau merancang jaringan infrastruktur yang memungkinkan terjadinya pemantauan jaringan untuk mengetahui statistik dan kinerja jaringan. *NDLC* mempunya enam fase, keenam fase tersebut dapat dilihat seperti pada gambar 1.1 berikut.



**Gambar 1. 1 Fase NDLC**

Sumber: Nurfajar, Kurniawan, dan Yunan, 2015

Dari keenam fase yang terdapat pada *NDLC*, penulis hanya menggunakan tiga fase antara lain sebagai berikut:

### 1. *Analysis*

Pada fase ini penulis melakukan pengumpulan data dengan cara studi literatur, yaitu penulis membaca artikel ilmiah, buku, dan jurnal untuk mendapatkan informasi mengenai *DMARC*, *DKIM*, *SPF*, *Anti Spam*, dan *Anti Virus*. Data-data yang telah terkumpul kemudian dianalisa.

### 2. *Design*

Pada fase ini penulis membuat rancangan yang meliputi rancangan jaringan uji coba, rancangan pengalamatan *IP*, rancangan sistem *filtering*, otentikasi, dan otorisasi *email* menggunakan *DMARC*, *DKIM*, *SPF*, *Anti Spam*, dan *Anti Virus*, serta kebutuhan perangkat keras dan perangkat lunak.

### 3. *Simulation Prototyping*

Setelah melakukan analisa dan desain, tahap berikutnya adalah melakukan simulasi dan membuat *prototype* berdasarkan pada desain yang telah dirancang sebelumnya. Pada fase ini dilakukan instalasi dan konfigurasi serta uji coba *DMARC*, *DKIM*, *SPF Anti Spam*, dan *Anti Virus* menggunakan berbagai macam skenario.

## 1.6. Sistematika Penulisan

Adapun sistematika penulisan yang digunakan pada skripsi ini adalah sebagai berikut:

### BAB I Pendahuluan

Bab ini berisi latar belakang, perumusan masalah, batasan masalah, tujuan dan manfaat penulisan, metodologi penelitian, dan sistematika penulisan.

### BAB II Landasan Teori

Bab ini berisi tentang teori-teori yang melandasi penelitian ini, antara lain jaringan komputer, *OSI*, *TCP/IP*, keamanan jaringan komputer, tujuan keamanan jaringan komputer, *Spoofing*, *Phising*, *Server*, Linux, Linux *CentOS*, *CentOS Web Panel*, *Email*, *Mail Server*, *POP3*, *IMAP*, *SMTP*, *Postfix*, *Dovecot*, *Roundcube*, *DNS*, *DNS Server*, *BIND9*, *HTTP* dan *HTTPS*, *HTTP Server*, Apache *HTTP Server*, *Email Spam*, *Anti Spam*, *SpamAssassin*, *ClamAV*, *DMARC*, *Amavisd-New*, *DKIM*, *OpenDKIM*, *SPF*, *Gmail*, *Emkei's Mailer*, *Yahoo! Mail*.

### BAB III Metodologi Penelitian

Bab ini berisi tentang metodologi penelitian yang digunakan dan fase-fase dari metodologi penelitian yang digunakan pada penelitian ini.

## **BAB IV Hasil dan Pembahasan**

Bab ini berisi tentang pembahasan hasil konfigurasi, uji coba, dan analisa terhadap uji coba yang telah dilakukan.

## **BAB V Penutup**

Bab ini berisi tentang kesimpulan dan saran untuk pengembangan skripsi ini selanjutnya.



## BAB II

### LANDASAN TEORI

#### 2.1 Jaringan Komputer

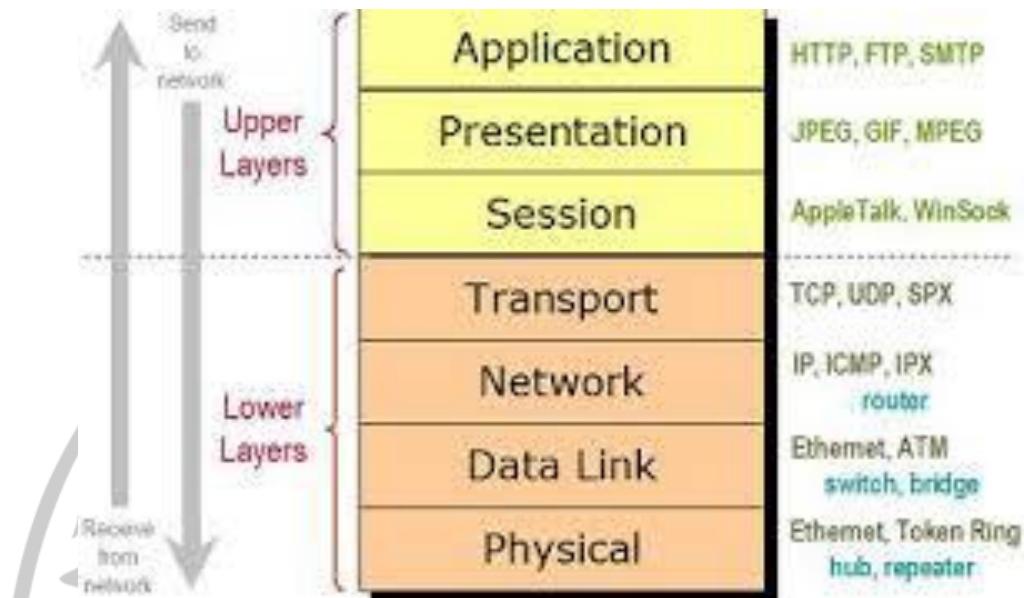
Menurut (Haryanto and Riadi 2014) sebuah jaringan komputer biasanya terdiri dari dua atau lebih komputer yang saling terhubung satu sama lain serta dapat saling berbagi sumber daya seperti *CDROM*, *printer*, pertukaran *file*, atau memungkinkan untuk saling berkomunikasi secara elektronik, komputer dapat terhubung melalui media transmisi seperti kabel, saluran telepon, gelombang radio, satelit atau infrared. Jaringan komputer merupakan sekumpulan perangkat yang dapat digunakan untuk menyimpan dan memanipulasi data elektronis serta pesan-pesan, saling terkait sehingga dapat berbagi pakai berupa data, perangkat keras, dan perangkat lunak. Jaringan komputer adalah sekelompok komputer otonom yang saling berhubungan satu sama lain menggunakan protokol komunikasi sehingga dapat saling berbagi informasi, aplikasi, dan perangkat keras secara bersama-sama, tujuan membangun jaringan komputer adalah untuk membawa secara tepat tanpa adanya kesalahan dari sisi pengirim menuju ke sisi penerima melalui media komunikasi (Erfanti, Taufik, and Joko 2016).

Berdasarkan dari ketiga pengertian tersebut dapat disimpulkan bahwa jaringan komputer adalah sebuah sistem yang menghubungkan *node-node* yang terdapat pada jaringan komputer dengan menggunakan media komunikasi tertentu sehingga *node-node* pada jaringan komputer dapat saling berbagi sumber daya, berkomunikasi, dan saling bertukar informasi untuk mencapai suatu tujuan yang sama (Hanif 2018).

#### 2.2 Model Lapisan *OSI*

Model *Open System Interconnection (OSI)* diciptakan oleh *International Organisation for Standardization (ISO)* yang menyediakan kerangka logika terstruktur bagaimana proses komunikasi data berinteraksi

melalui jaringan, standar ini dikembangkan untuk industri komputer agar komputer dapat berkomunikasi pada jaringan yang berbeda secara efisien (Sinsuw 2014).



Gambar 2.1 Layer OSI  
Sumber: Sujana, 2014

Menurut Sujana (2014) terdapat tujuh *layer* pada model *OSI* dan setiap *layer* memiliki tanggung jawab khusus pada proses komunikasi data:

### 1. *Physical*

Pada *physical layer* tidak memiliki protokol yang spesifik, karena pada *physical layer* hanya mengirimkan *bit* data.

### 2. *Data Link*

Terdapat dua protokol pada *data link layer* yaitu:

- *PPP (Point to Point Protocol)*

Protokol yang digunakan untuk komunikasi *point to point* pada suatu jaringan.

- *SLIP (Serial Line Internet Protocol)*

Protokol yang digunakan untuk menghubungkan *serial*.

### **3. Network**

Terdapat tiga protokol pada *network layer* yaitu:

- *IP (Internetworking Protocol)*

Mekanisme transmisi yang digunakan untuk mentransportasikan data dalam paket yang disebut *datagram*.

- *ARP (Address Resolution Protocol)*

Protokol yang digunakan untuk mengetahui alamat *IP* berdasarkan alamat fisik dari sebuah komputer.

- *RARP (Reverse Address Resolution Protocol)*

Protokol yang digunakan untuk mengetahui alamat fisik melalui alamat *IP* komputer.

- *ICMP (Internet Control Message Protocol)*

Mekanisme yang digunakan oleh sejumlah *host* untuk mengirim notifikasi datagram yang mengalami masalah pada *hostnya*.

- *IGMP (Internet Group Message Protocol)*

Protokol yang digunakan untuk memberi fasilitas pesan yang simultan kepada grup penerima.

### **4. Transport**

Terdapat dua protokol pada *transport layer* yaitu:

- *TCP (Transmission Control Protocol)*

Protokol yang menyediakan layanan penuh pada lapisan *transport* untuk aplikasi.

- *UDP (User Datagram Protocol)*

Protokol *connectionless* dan *proces-to-proces* yang hanya menambahkan alamat *port*, *checksum error control* dan panjang informasi data pada *layer* diatasnya.

### **5. Session**

Terdapat empat protokol pada *session layer* yaitu:

- *NETBIOS*

Berfungsi sebagai penyiaran pesan, maksudnya adalah memungkinkan *user* mengirim pesan tunggal secara serempak

ke komputer lain yang terkoneksi. *NETBEUI (NETBIOS Extended User Interface)* berfungsi sama dengan *NETBIOS* hanya sedikit dikembangkan lagi dengan menambah fungsi yang memungkinkan bekerja dengan perangkat keras dan perangkat lunak.

- *ADSP (AppleTalk Data Stream Protocol)*

Fungsi dari protokol ini adalah untuk memantau aliran data dianatara dua komputer dan untuk memeriksa aliran data tersebut tidak terputus.

- *PAP (Printer Access Protocol)*

Berfungsi sebagai *printer postscript* untuk melakukan akses pada jaringan *Apple Talk* dan untuk mengendalikan bagaimana pola komunikasi antar *node*.

- *SPDU (Session Protocol Data Unit)*

Berfungsi sebagai penghubung antara dua *session service user*.

## 6. *Presentation*

Terdapat tiga protokol pada *layer presentation* yaitu:

- *TELNET*

Protokol yang digunakan untuk melakukan *remote access* ke suatu *host*.

- *SMTP (Simple Mail Transfer Protocol)*

Salah satu protokol yang digunakan dalam pengiriman *email* di internet atau untuk mengirim data dari komputer pengirim *email* ke *server email* penerima.

- *SNMP (Simple Network Management Protocol)*

Protokol yang digunakan dalam suatu manajemen jaringan.

## 7. *Application*

Terdapat sembilan protokol pada *layer application* yaitu:

- *HTTP (Hyper Text Transfer Protocol)*

Protokol yang digunakan untuk mentransfer dokumen dan web dalam sebuah *web browser* melalui *www*.

- *FTP (File Transfer Protocol)*

Protokol *internet* yang berjalan dalam lapisan aplikasi yang merupakan standar untuk mentransfer *file* komputer dalam sebuah jaringan *internet*.

- *NFS (Network File System)*

Jaringan komputer yang memungkinkan pengguna di klien komputer untuk mengakses *file* melalui jaringan dengan cara yang sama saat mengakses *file* pada sumber penyimpanan lokal.

- *DNS (Domain Name System)*

Protokol yang digunakan untuk memberikan suatu nama *domain* pada sebuah alamat *IP* agar lebih mudah diingat.

- *POP3 (Post Office Protocol)*

Protokol yang digunakan untuk mengambil *mail* dari suatu *mail transfer agent* yang akhirnya *mail* tersebut akan *download* kedalam jaringan lokal.

- *MIME (Multipurpose Internet Mail Extension)*

Protokol yang digunakan untuk mengirim *file binary* dalam bentuk teks.

- *SMB (Server Message Block)*

Protokol yang digunakan untuk mentransfer *server-server file* ke DOS dan Windows.

- *NNTP (Network News Transfer Protocol)*

Protokol yang digunakan untuk menerima dan mengirim *newsgroup*.

- *DHCP (Dynamic Host Configuration Protocol)*

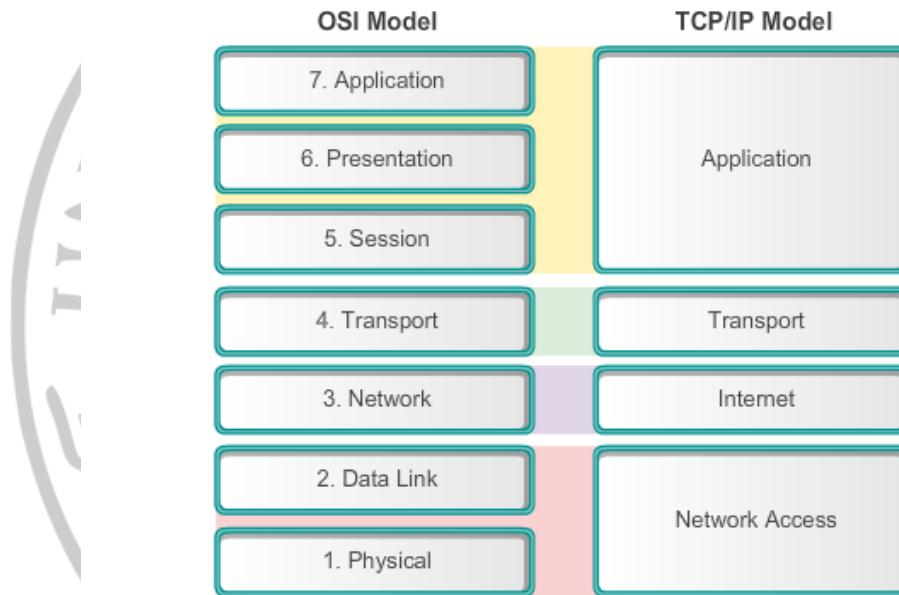
Layanan yang memberikan alamat *IP* kepada komputer yang memintanya secara otomatis.

### 2.3 *Transmission Control Protocol / Internet Protocol (TCP/IP)*

*TCP/IP* didefinisikan sebagai protokol jaringan yang berperan dalam membangun *environment* jaringan global seperti *internet*. Protokol direferensikan pula sebagai suit *protocol DoD* (“*deehoohdee*”) karena mereka

pada dasarnya dikembangkan oleh komunitas riset *Advanced Research Projects Agency* (ARPA) dari *US Department of Defense* (DoD) (Hanif 2018).

Nama *TCP/IP* diambil dari dua ‘Keluarga’ protokol fundamental, yaitu *TCP* dan *IP*. Meskipun demikian, suit masih memiliki protokol utama lainnya seperti *UDP* dan *ICMP*. Protokol bekerja sama dalam memberikan *framework networking* yang digunakan oleh banyak protokol aplikasi berbeda, di mana masing-masing digunakan untuk tujuan berbeda (Kader, Najoan, dan Sinsuw, 2014).



Gambar 2.2 Perbandingan Layer *TCP/IP* dan Layer *OSI*

Sumber: Wardoyo, Ryadi, dan Fahrizal, 2014

Berikut fungsi dari masing-masing *layer* pada protokol *TCP/IP* (Ryadi, 2011):

### 1. *Network Access Layer*

*Layer network access* merupakan gabungan antara dua *layer* yaitu *network interface layer* dan *physical layer*, *network interface layer* berfungsi untuk mengirim data ke *layer physical* melalui device jaringan kemudian dilanjutkan oleh *layer physical* yang

*merupakan sistem kabel yang digunakan untuk proses mengirim dan menerima data.*

### **2. Internet Layer**

Pada lapisan *internet* terjadi proses pengambilan paket dari lapisan *transport* dan menambahkan informasi alamat sebelum mengirimkannya ke lapisan *network interface*.

### **3. Transport Layer**

Pada lapisan *transport* terdapat protokol seperti *TCP* dan *UDP* yang berfungsi menambahkan data *transport* ke paket dan melewatkannya ke lapisan *Internet*.

### **4. Application Layer**

Pada lapisan *application* terdapat protokol seperti *FTP*, *Telnet*, *SMTP*, dan *NFS* dilaksanakan.

## **2.4 Keamanan Jaringan Komputer**

Menurut Fitriani (2014) Keamanan jaringan komputer merupakan suatu proses untuk mencegah dan mengidentifikasi penggunaan yang tidak sah dari jaringan komputer. Langkah-langkah pencegahan membantu menghentikan pengguna yang tidak sah yang disebut “penyusup” untuk mengakses setiap bagian dari sistem jaringan komputer. Tujuan keamanan jaringan komputer adalah untuk mengantisipasi resiko jaringan komputer berupa bentuk ancaman fisik maupun logik baik langsung ataupun tidak langsung mengganggu aktivitas yang sedang berlangsung dalam jaringan komputer.

## **2.5 Jenis-jenis Layanan Keamanan Jaringan**

Menurut Fitriani (2014) terdapat beberapa jenis layanan keamanan jaringan, diantaranya:

### **1. Otentikasi (*Authentication*)**

Layanan Otentikasi ada 2 macam. Pertama disebut dengan Otentikasi Entitas (*Entity Authentication*) yaitu layanan keamanan jaringan yang memberikan kepastian terhadap identitas sebuah

entitas yang terlibat dalam komunikasi data. Kedua adalah Otentikasi Keaslian Data (*Data Origin Authentication*) yaitu layanan yang memberikan kepastian terhadap sumber sebuah data.

## 2. Kendali Akses (*Access Control*)

Kendali Akses adalah layanan keamanan jaringan yang menghalangi penggunaan tidak terotorisasi terhadap sumber daya. Pada aplikasi jaringan umumnya kebijakan kemampuan (baca, modifikasi, tulis dan eksekusi sebuah data/layanan sistem) ditentukan oleh jenis pengguna.

## 3. Kerahasiaan Data (*Data Confidentiality*)

Kerahasiaan data adalah layanan keamanan jaringan yang memproteksi data tertransmisi terhadap pengungkapan oleh pihak yang tidak berwenang / berhak.

## 4. Keutuhan Data (*Data Integrity*)

Keutuhan data adalah layanan keamanan jaringan yang memastikan bahwa data yang diterima oleh penerima adalah benar-benar sama dengan data yang dikirim oleh pengirim.

## 5. *Non-Repudiation*

Layanan *non-repudiation* adalah layanan keamanan jaringan yang menghindari penolakan atas penerima atau pengirim data yang telah dikirim.

## 6. Ketersediaan (*Availability*)

Layanan *Availability* adalah layanan sistem yang membuat sumber daya sistem tetap dapat diakses dan digunakan ketika ada permintaan dari pihak yang berwenang. Serangan seperti *Denial of Service* membuat sistem tidak dapat diakses oleh pihak yang berwenang.

## **2.6 Email Spoofing dan Phising**

(Suryana, Akbar, and Widiyasono 2016) *Email Spoofing* adalah kegiatan melakukan manipulasi data pada *header email*. Serangan yang paling populer dari *email spoofing* adalah serangan *phising*. *Email spoofing* dianggap sebagai tindakan yang berbahaya, karena melakukan manipulasi data pada *header email* untuk menyamar sebagai orang atau organisasi yang berwenang, contohnya seperti melakukan pengiriman *email* dengan nama pengirim seolah-olah *email* tersebut dikirim oleh administrator suatu organisasi. Pengirim *email spoofing* menyerang dengan berbagai macam isi pesan untuk meyakinkan korbannya.

(Suryana, Akbar, and Widiyasono 2016) *Phising* adalah bentuk pencurian identitas secara *online* yang bertujuan untuk mencuri informasi sensitif seperti sandi dan informasi kartu kredit. Serangan *phising* menggunakan kombinasi teknik *social engineering* dan teknik *spoofing* untuk membujuk pengguna agar memberikan informasi sensitif yang dapat digunakan untuk memperoleh keuntungan pribadi, salah satu contohnya adalah keuntungan finansial. *Phiser* biasanya membajak sebuah halaman web dari bank, kemudian mengirim *email* kepada korbannya supaya korbannya mengunjungi situs berbahaya dengan tujuan untuk mengumpulkan informasi rekening bank dan nomor kartu milik korbannya.

## **2.7 Server**

*Server* adalah sebuah sistem komputer yang menyediakan jenis layanan tertentu dalam sebuah jaringan komputer. Beberapa contoh layanan *server* adalah *DHCP Server*, *DNS Server*, *FTP Server*, *Web Server*, *Mail server*, *Database Server* dan lain-lain (Saputra & Syafrizal, 2012).

## **2.8 Linux**

Menurut Harjono (2016) Linux adalah sebuah aplikasi atau program yang menggunakan kernel sebagai sistem operasi. *Script* pertama Linux dirancang dan ditulis oleh seorang mahasiswa dari Finlandia bernama "Linus Torvalds" untuk arsitektur Intel 80386. Banyak orang memiliki

peran penting dalam mengembangkan dan memperluas Linux di berbagai belahan dunia. Peralatan sistem dan pustakanya umumnya berasal dari sistem operasi GNU yang diumumkan tahun 1983 oleh Richard Stallman. Kontribusi GNU merupakan dasar dari munculnya nama alternatif GNU/LINUX. Dia menggunakan alat proyek GNU dan dengan demikian sistem operasi dikembangkan melalui proyek GNU/LINUX.

## 2.9 Linux *CentOS*

*CentOS* merupakan singkatan dari *Community ENTerprise Operating System* yang merupakan sebuah distribusi Linux sebagai bentuk dari usaha untuk menyediakan *platform* komputasi berkelas *enterprise* yang memiliki kompatibilitas kode biner sepenuhnya dengan kode sumber yang menjadi induknya, *Red Hat Enterprise Linux (RHEL)*. *RHEL* merupakan distribusi Linux berbayar yang menyediakan akses update atas perangkat lunak dan beragam jenis dukungan teknis. Distribusi Linux ini sebenarnya merupakan gabungan dari sejumlah perangkat lunak yang didistribusikan di bawah lisensi perangkat lunak yang bebas dan kode sumber atas paket perangkat lunak ini dirilis ke publik oleh *Red Hat*. *CentOS* tersedia secara gratis, dukungan teknis utamanya disediakan terhadap para pengguna melalui *mailing list*, forum berbasis web, ataupun *chat*. Proyek *CentOS* tidak berafiliasi dengan *Red Hat*, sehingga proyek *CentOS* berjalan tanpa mendapatkan bantuan apapun dari *Red Hat*. Untuk penggalangan dana, *CentOS* berbasis donasi dari para pengguna serta sponsor dari perusahaan-perusahaan yang menggunakaninya (Wicitra, Utomo, dan Wardana, 2014).

## 2.10 *Centos Web Panel*

*CentOS Web Panel* adalah panel kontrol untuk *web hosting* yang dapat digunakan secara gratis dan dirancang untuk memanajemen *VPS* maupun *Dedicated Server* dengan cepat dan mudah tanpa harus menggunakan aplikasi *SSH Client*, menawarkan sejumlah besar opsi dan fitur untuk manajemen *server* dalam paket panel kontrolnya.

Centos Web Panel system administrator dapat menggunakan satu server untuk membuat banyak server virtual untuk kebutuhan yang berbeda-beda, seperti menggunakan server untuk kebutuhan web server, DNS Server, mail server dan lain sebagainya. Hal ini membuat penggunaan server lebih efektif dan efisien dan tidak ada resource yang terbuang sia-sia karena tidak terpakai. lalu setiap server penggunaannya akan menggunakan remote untuk setiap administrator dengan pengaturan autentikasi yang berbeda-beda (Ginantra et al. n.d.).

## 2.11 Surat Elektronik

Surat elektronik adalah layanan yang diberikan oleh *internet* yang berkembang sejak tahun 1960, pada saat itu *internet* belum terbentuk, yang ada hanyalah kumpulan *mainframe* yang terbentuk sebagai jaringan. Mulai tahun 1980-an, surat elektronik sudah bisa dinikmati oleh khalayak umum. Surat elektronik adalah salah satu proses pengiriman surat melalui *internet* dengan menggunakan waktu yang sangat singkat. Surat elektronik merupakan salah satu dari sekian banyak layanan *internet* yang ada saat ini selain *Netnews*, *Telnet*, *File Transfer Protokol (FTP)* dan *World Wide Web (www)* dan masih banyak layanan yang lainnya. Layanan *internet* adalah berbagai program atau fasilitas yang disediakan oleh *internet*, dari layanan *internet* tersebut yang paling banyak digunakan adalah layanan surat elektronik. Penggunaan *electronic mail* (surat elektronik) sebagai media komunikasi yang ditunjang oleh banyaknya penyedia layanan di*internet* seperti Yahoo, Google, MSN, Wordpress, dan yang lainnya menunjukkan bahwa banyak orang melakukan komunikasi karena dengan komunikasi orang dapat beraktivitas dan meningkatkan kariernya (Mawarsih, 2014).

## 2.12 Mail Server

(Desmira, Sumarto, and Yuliani 2017) *Mail server* dikenal sebagai sebuah *mail transfer agent* atau *MTA*, *mail router* atau *mailer Internet* adalah sebuah aplikasi yang akan menerima *email masuk* dari pengguna

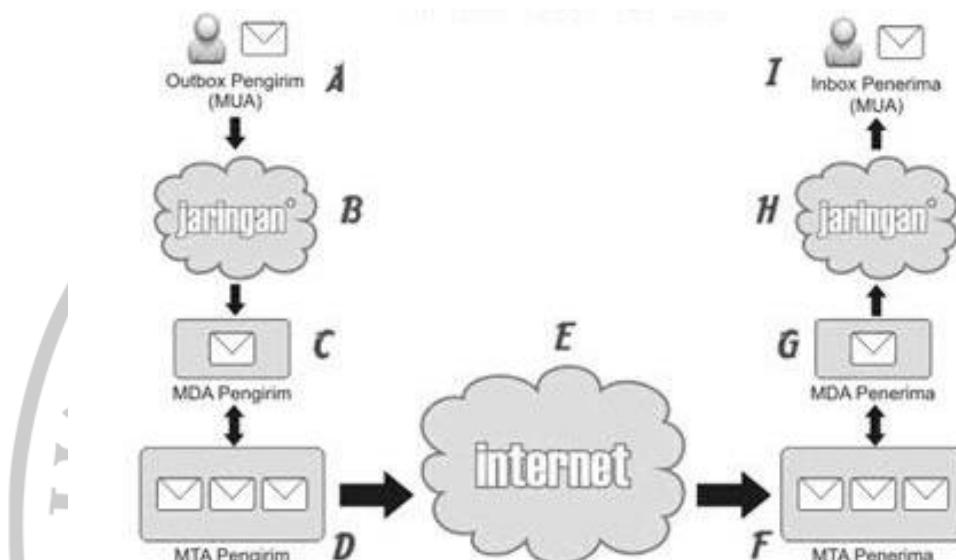
*lokal (orang-orang dalam satu domain) dan jarak jauh pengirim dan meneruskan email keluar untuk pengiriman. Sebuah komputer yang didedikasikan untuk menjalankan aplikasi tersebut juga disebut sebagai mail server. Mail Server bisa diartikan sebagai induk atau rumah dari email, Setiap email yang dikirimkan dibuat untuk melewati sejumlah server mail sepanjang perjalanan ke penerima. Untuk user biasa, surat tersebut dikirim langsung tetapi proses adalah sesuatu yang dimengerti. Tanpa rangkaian Server Mail, pengguna hanya akan dapat mengirim email ke orang-orang yang memiliki alamat email dengan domain yang sama.*

Menurut (Muarif and Irwan 2018) *Mail Server memiliki tiga komponen utama yang membentuknya, yakni Mail Transfer Agent (MTA), Mail Delivery Agent (MDA), dan Mail User Agent (MUA):*

1. Menurut Sadikin (2014) *Mail User Agent (MUA)* merupakan program yang digunakan oleh pemakai untuk membaca dan mengirim *email* pada komputer pribadinya. Contoh program atau perangkat lunak *Mail User Agent (MUA)* ini misalnya Microsoft Outlook, Microsoft Outlook Express, Lotus Notes, Pegasus Mail dan Thunderbird. *Mail User Agent (MUA)* mengambil *email* dari *email server* menggunakan protokol *Post Office Protocol (POP)* dan *Internet Message Access Protocol (IMAP)*.
2. *Mail Tranfer Agent (MTA)* *Mail Transfer Agent* merupakan salah satu komponen penting pada server internet. *Mail Transfer Agent* bertanggung jawab untuk mentransfer *email* dari *mail server* mengirimkan sampai ke *server penerima email*. Kebutuhan pengguna atas jenis MTA yang digunakan juga beragam. Berbagai kriteria biasa digunakan untuk pertimbangan. Tiap-tiap program *mail server* memiliki kelebihan dan kekurangan tersendiri. Beberapa MTA memiliki fasilitas yang sangat hebat sehingga mampu digunakan untuk

*menangani email dalam jumlah ratusan bahkan sampai ribuan perhari* (Desmira, Sumarto, and Yuliani 2017).

3. Menurut Crocker (2009) *Mail delivery agent* atau *message delivery agent (MDA)* adalah komponen perangkat lunak komputer yang bertanggung jawab atas pengiriman pesan *email* ke kotak pesan penerima lokal.



**Gambar 2.3 Proses pengiriman email**

Sumber: Pratama, 2008

Pada gambar 2.3 dapat dijelaskan proses pengiriman *email* dimulai dari proses A yaitu pengirim *email* mengirim *email* menggunakan *MUA*, kemudian *email* diteruskan pada *MDA* yang berfungsi untuk mengatur pengiriman *email* pada *mail server* lokal (proses C), jika *email* tersebut dikirim kepada penerima yang berada pada *mail server* yang berbeda maka *email* akan dikirim melalui *MTA* untuk diteruskan ke *mail server* penerima melalui jaringan *internet* (proses E) kemudian *email* tersebut diterima oleh *MTA* pada *mail server* penerima (proses F) dan dilanjutkan ke *MDA mail server* penerima (proses G) agar *email* dapat di unduh oleh penerima *email* melalui jaringan lokal (proses H dan I).

### 2.13 *Mail Protocol*

Menurut (Desmira, Sumarto, and Yuliani 2017) terdapat tiga *Mail Protocol*, yaitu:

### **1. POP3 (Post Office Protocol version 3)**

*POP3 merupakan protokol yang digunakan untuk pengelolaan email. POP3 memudahkan seseorang dalam mendapatkan email mereka dari sebuah mail server tanpa perlu koneksi yang lama dengan internet yang tentu saja memakan biaya.*

### **2. IMAP (Internet Message Access Protocol)**

*IMAP (Internet Message Access Protocol) sama halnya dengan POP3, maka pesan email akan sepenuhnya disimpan dalam server email dan menggunakan komputer lokal untuk mengirim dan mengambilnya kapanpun di inginkan. Tergantung dari keinginan user. IMAP adalah protocol standar untuk mengakses atau mengambil email dari server.*

### **3. SMTP (Simple Mail Transfer Protocol)**

*SMTP merupakan salah satu jenis protocol yang bekerja dalam hal pengiriman pesan-pesan berupa surat elektronik atau email pada sebuah jaringan internet.*

## **2.14 Postfix**

Menurut (Wahyu noer hidayat 2010) “Postfix adalah *Mail Transfer Agent* yang dapat diperoleh dengan gratis dan bersifat *open source*. Postfix merupakan *mail transfer agent default* untuk sejumlah sistem operasi yang bertipe unix. Postfix didistribusikan menggunakan lisensi umum IBM 1.0 yang merupakan lisensi perangkat lunak bebas tetapi tidak kompatibel dengan *GPL*”.

Menurut (Kusmaya 2016) Postfix ditulis oleh Wietse Venema dan termasuk salah satu proyek *freeware*. Mulai digarap Wietse saat berkunjung ke *IBM T. J. Watson Research*. Wietse diberi kesempatan oleh IBM untuk menulis *software* ini. *Original software* tersebut diberi nama Vmailer, namun diganti menjadi *Postfix* atas saran *IBM*.

## **2.15 Dovecot**

Menurut (Kusmaya 2016) “Dovecot adalah *open source server POP3* dan *IMAP* untuk Linux atau Unix. Program ini melengkapi *Postfix* dengan kinerja yang tinggi, kemudahan administrasi, dan keamanan yang solid. Dovecot merupakan sebuah aplikasi yang dijalankan untuk mengikuti protokol *IMAP* dan *POP3*.

## **2.16 Roundcube**

*Roundcube* adalah solusi *webmail* gratis dan *open source* dengan antarmuka pengguna mirip *desktop* yang mudah dipasang atau dikonfigurasi dan berjalan pada *server LAMPP* standar. Tampilan menggunakan standar web terbaru untuk merender antar muka yang fungsional dan dapat disesuaikan. *Roundcube* menyertakan *library open-source* canggih lainnya seperti *PEAR*, *IMAP* yang berasal dari *IlohaMail*, pustaka *Googiespell* untuk pemeriksaan ejaan atau pembersih *WasHTML* oleh *Frederic Motte* (*Roundcube Open Source Webmail Software*, n.d.) (Hanif 2018).

## **2.17 Domain Name System (DNS)**

Menurut (Saputra and Syafrizal 2012) *Domain Name System* adalah sebuah sistem yang menyimpan dan mengatur suatu informasi tentang penamaan *host* dari sebuah alamat *IP* menjadi sebuah karakter atau angka dalam sebuah jaringan internet yang di distribusikan pada *database*. *Domain name system* memiliki pengelolaan komponen inti yang terdiri dari *DNS resolver*, *Recursive DNS server* dan *Authoritative DNS server*. pada awal penggunaan *DNS* didalam jaringan komputer menggunakan *HOSTS.TXT* dari *SRI* (sekarang *SIR International*) yang berisi informasi dari nama komputer dan *IP address*.

## **2.18 DNS Server**

Menurut (Kusmaya 2016) *DNS server* adalah *distribute database system* yang digunakan untuk pencarian nama komputer di jaringan yang menggunakan *TCP/IP* (*Transmission Control Protocol/Internet Protocol*). *DNS server* biasa digunakan pada aplikasi yang terhubung ke *internet*

seperti *web browser* atau *email*, dimana *DNS server* dapat membantu memetakan *hostname* sebuah komputer ke *IP Address*.

### 2.19 *Bind9*

*BIND9* adalah aplikasi *DNS server* yang paling umum digunakan di *internet*, khususnya di sistem unix, *bind9* merupakan standar *DNS server*. *BIND9* awalnya dibuat oleh empat orang mahasiswa dengan menggunakan CSRG di Universitas California, Berkeley dan pertama kali dirilis di dalam 4.3 BSD. Paul Vixie kemudian meneruskan pemrogramannya pada tahun 1988 saat bekerja di *DEC*. Saat ini, *Bind9* dikelola oleh Konsorsium sistem *internet*. *BIND9* awalnya ditulis pada awal 1980 dan didanai oleh *DARPA* (*Defense Advanced Research Projects Agency*). Pada pertengahan 1980-an, *DEC* (*Digital Equipment Corporation*) mengambil alih pengembangan *BIND9*. Satu dari pekerja itu adalah Paul Vixie, yang terus mengerjakan *BIND9* sesudah meninggalkan *DEC* (Wahyu noer hidayat 2010).

### 2.20 *HTTP*

Menurut (Zabar and Novianto 2015) *HTTP* adalah sebuah protokol yang bekerja dengan cara meminta atau menjawab antara *client* dan *server*. Sebuah *client* *HTTP* seperti *web browser*, biasanya memulai permintaan dengan membuat hubungan *TCP/IP* ke *port* tertentu di tuan rumah yang jauh (biasanya *port* 80). Sebuah *server* *HTTP* yang mendengarkan di *port* tersebut menunggu *client* mengirim kode permintaan (*request*), seperti "GET / HTTP/1.1" (yang akan meminta halaman yang sudah ditentukan), diikuti dengan pesan *MIME* yang memiliki beberapa informasi kode kepala yang menjelaskan aspek dari permintaan tersebut dan diikuti dengan badan dari data tertentu. Beberapa kepala (*header*) juga dapat ditulis atau tidak, sementara yang lainnya (seperti tuan rumah) diperlukan oleh protokol *HTTP/1.1*. Begitu menerima kode permintaan (dan pesan bila ada), *server* mengirim kembali kode jawaban, seperti "200 OK", dan sebuah pesan yang diminta, atau sebuah pesan *error* atau pesan lainnya. Pengembangan *HTTP* dikordinasi oleh Konsorsium *World Wide Web* (*W3C*) dan grup kerja

*Internet Engineering Task Force (IETF)*, bekerja dalam publikasi satu seri *RFC*, yang paling terkenal *RFC 2616*, yang menjelaskan *HTTP/1.1*, versi *HTTP* yang umum digunakan sekarang.

## **2.21 *HTTP Server***

Menurut (Saputra and Syafrizal 2012) *HTTP Server* adalah sebuah *software* yang melayani permintaan berupa *Hypertext Transfer Protocol (HTTP)* atau *Hypertext Transfer Protocol Secure (HTTPS)* dari komputer atau *client* yang terhubung dalam jaringan *internet* atau *intranet*.

## **2.22 *Apache HTTP Server***

Menurut (Saputra and Syafrizal 2012) Apache *HTTP Server* adalah *web server* yang dapat dijalankan di banyak sistem operasi, seperti Unix, *BSD*, Linux, Microsoft Windows dan Novell Netware serta *platform* lainnya yang berguna untuk melayani dan memfungsikan situs web.

## **2.23 *Email Spam***

Menurut (Chandra, Indrawan, and Sukajaya 2016) *Spam email* dapat didefinisikan sebagai “*unsolicited bulk email*” yaitu *email* yang dikirimkan kepada ribuan penerima. *Spam email* biasanya dikirimkan oleh suatu perusahaan untuk mengiklankan produknya. Hal ini menyebabkan semakin padatnya antrian dari *mail server*. Banyak waktu yang terbuang untuk menghapus *email spam* dari kotak masuk, *spam* juga menyebabkan pemborosan biaya bagi pengguna yang menggunakan koneksi *dial-up*. Selain itu *spam* juga dapat membuang *bandwidth* dan dapat menyebabkan penerima di bawah umur mengakses situs-situs yang memiliki konten negatif. Banyaknya *spam* menyebabkan kerugian dalam hal sumber daya dan memerlukan banyak waktu untuk menghapusnya.

## **2.24 *Spam Filter***

*Spam filter* merupakan *software anti spam*, *Software anti spam* bekerja dengan cara menganalisa *email* yang datang dan menggunakan sejumlah metode untuk menentukan apakah *email* yang diterima adalah

*email spam* atau bukan. Keberhasilan *spam filter* dalam mencegah masuknya *email spam* tergantung dari *software anti spam* yang digunakan serta metode-metode yang diterapkan oleh *software anti spam* untuk mendeteksi dan mencegah *email spam* (Fachrurrazi 2014).

## 2.25 *SpamAssassin, ClamAV, dan Amavisd-New*

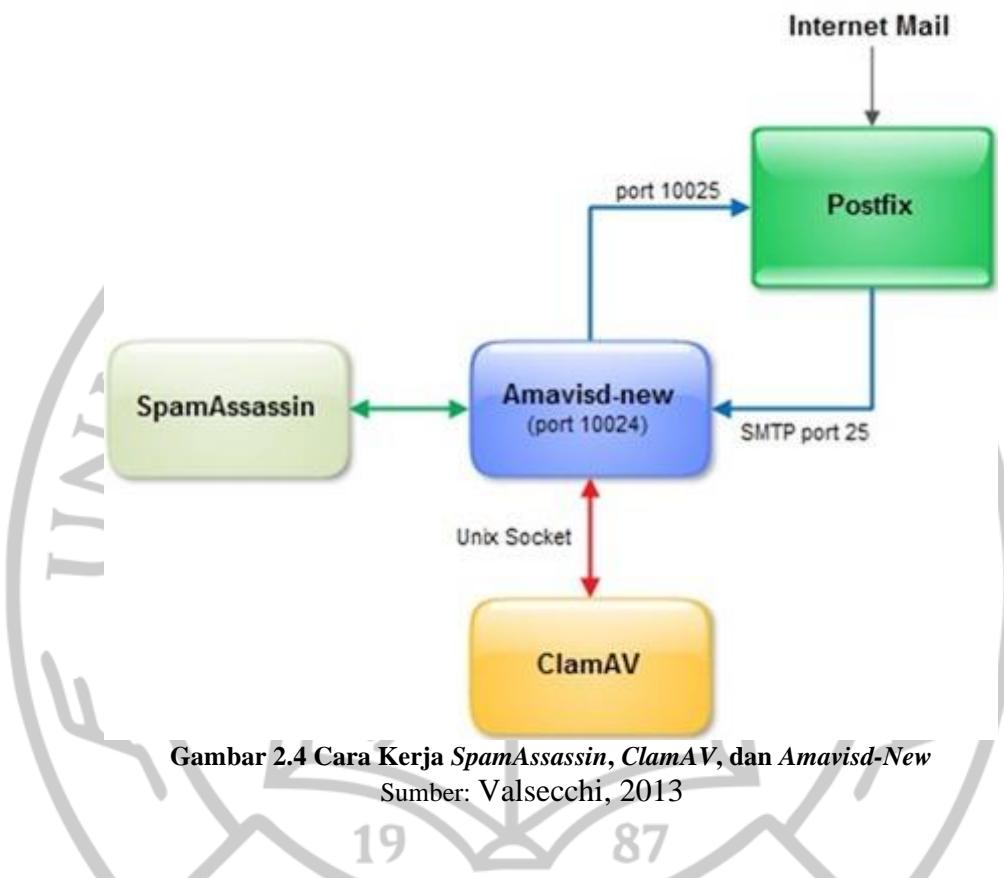
Menurut (Nurlina and Irmayana 2015) *SpamAssassin* adalah aplikasi yang sudah teruji secara luas menggunakan proyek *open source* yang berfungsi sebagai *mail filter* untuk mendeteksi *spam*. *SpamAssassin* berjalan pada *server* dan sebagai *filter spam* sebelum sampai pada kotak masuk pengguna. *SpamAssassin* diintegrasikan dengan *mail server* agar secara otomatis menyaring semua *email spam* dan aturan penggunaan atau tes untuk menentukan *email spam* atau *ham*. *SpamAssassin* dapat memberikan tanda dengan mengubah *subject email* atau langsung menghapus *email spam* yang masuk.

*SpamAssassin* menggunakan berbagai mekanisme untuk menangani *email spam*, berikut mekanisme yang diterapkan *SpamAssassin*:

1. Pengecekan *header email*.
2. Pengecekan isi *email*.
3. Pengelompokan *email address* secara manual kedalam *whitelist* atau *blacklist*.
4. *Bayesian filtering*.
5. Penyaringan *database spam* kolaboratif (*DCC*, *Pyzor*, dan *Razor2*).
6. Berbasis jaringan seperti *blacklist URL*, *blacklist DNS*, *checksum* berbasis *filter*, dan algoritma *Hash*.

Menurut (Kusmaya 2016) *ClamAV* adalah *anti virus open source* (*GPL*) yang dirancang untuk mendeteksi *trojan*, *virus*, *malware*, dan ancaman berbahaya lainnya. Secara *de facto* *ClamAV* adalah standar untuk pemindaian *mail gateway*.

*Amavisd-new* adalah antarmuka yang memiliki kinerja yang tinggi dan dapat diandalkan. *Amavisd-new* memiliki beberapa fitur seperti pemindai virus dan modul *SpamAssassin*. *Amavisd-new* berkomunikasi ke MTA melalui protokol *SMTP* atau *LMDP*, atau dengan menggunakan program pembantu .



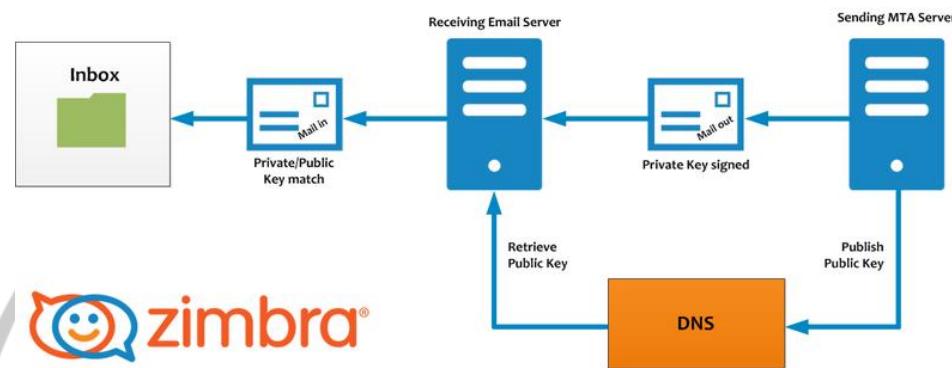
Cara kerja *SpamAssassin*, *ClamAV*, dan *Amavisd-New* dapat dilihat seperti gambar 2.4 yaitu *Amavisd-New* menerima *email* dari *Postfix* (*MTA*), kemudian menyebakannya ke *ClamAV* dan *SpamAssassin* untuk memeriksa spam dan virus lalu mengembalikan *email* ke *Postfix* (*MTA*) untuk diteruskan ke penerima *email*.

## 2.26 *Domain Keys Identified Mail (DKIM)* dan *OpenDKIM*

*Domain Keys Identified Mail (DKIM)* adalah metode otentifikasi *email* yang dirancang untuk mendeteksi *spoofing email*. Ini memungkinkan penerima untuk memeriksa bahwa *email* yang diklaim berasal dari *domain* tertentu memang diotorisasi oleh pemilik *domain* tersebut. Hal ini

dimaksudkan untuk mencegah alamat pengirim palsu dalam *email* yang sering digunakan untuk melakukan *phishing* dan *spam email*. (Hansen, Crocker, Baker, 2009).

Menurut Barovih (2011) *OpenDKIM* adalah pengiriman *email* yang menggunakan mekanisme otentikasi *framework* menggunakan kunci publik yang dimasukan ke dalam *DNS* maupun *email*.

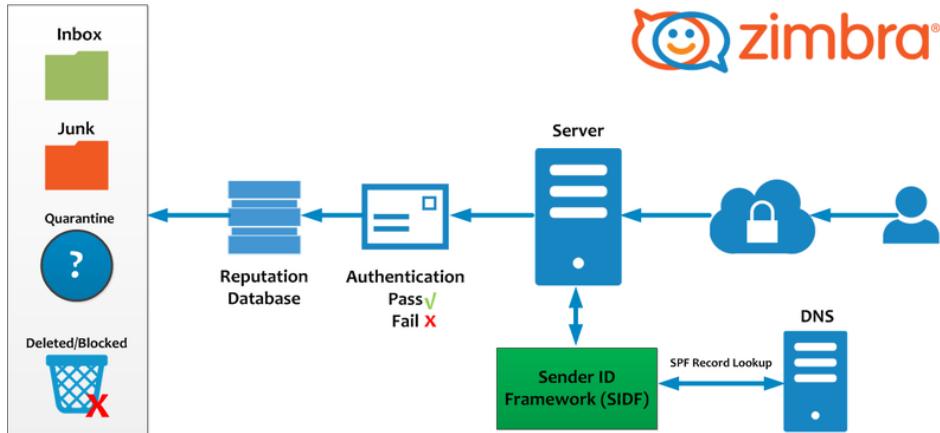


Gambar 2.5 Cara Kerja *DKIM*  
Sumber: Zimbra Incorporation, 2005

Cara kerja *DKIM* dapat dilihat seperti pada gambar 2.5 yaitu mail server pengirim mempublish *public key* pada pada *DNS server* pengirim, setiap *email* yang dikirim melalui *mail server* pengirim akan diberikan *private key*, setelah *email* sampai pada *mail server* penerima maka *mail server* penerima akan mencocokan *private key* yang terdapat pada *email* dengan *public key* yang terdapat pada *DNS server* penerima, jika *public* dan *private key* cocok maka *email* tersebut dapat dipastikan berasal dari pengirim yang asli, namun jika *public* dan *private key* tidak cocok maka *email* tersebut dapat dipastikan sebagai *email spoofing*.

## 2.27 Sender Policy Framework (SPF)

*Sender Policy Framework (SPF)* adalah sistem validasi *email*, yang dirancang untuk mencegah *email* yang tidak diinginkan menggunakan sistem *spoofing*. Untuk memeriksa masalah keamanan umum ini, *SPF* akan memverifikasi *IP* sumber *email* dan membandingkannya dengan data *TXT DNS* dengan konten *SPF* (Zimbra Incorporation, *Best Practices on Email Protection: SPF, DKIM and DMARC*, 2005).



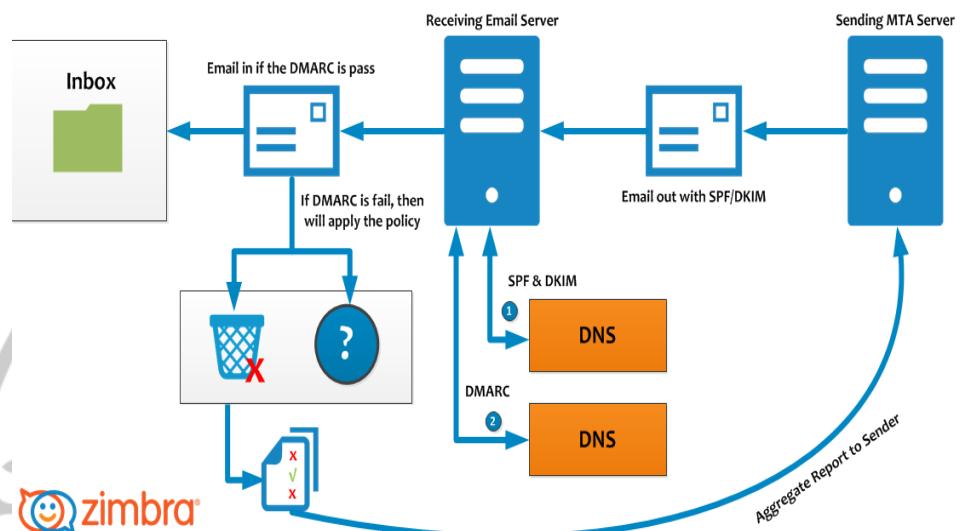
**Gambar 2.6 Cara Kerja SPF**  
Sumber: Zimbra Incorporation, 2005

Cara kerja *SPF* dapat dilihat seperti pada gambar 2.6 yaitu *email* yang dikirim oleh pengirim akan diteruskan pada *mail server* penerima, selanjutnya *mail server* penerima akan mengecek *Sender ID Framework* yang berada pada *DNS server* pengirim, jika alamat *IP server* pengirim *email* sesuai dengan *alamat IP* yang telah diotorisasi oleh *SPF record* pada *DNS server* pengirim *email* maka *email* tersebut akan diberi nilai *PASS*, namun jika alamat *IP server* pengirim *email* tidak sesuai dengan alamat *IP* yang telah diotorisasi oleh *SPF record* pada *DNS server* pengirim *email* maka *email* tersebut akan diberi nilai *FAIL* atau *SOFTFAIL* dan selanjutnya database reputasi akan memberi nilai pada *email* tersebut berdasarkan pada laporan *SPF* masing-masing *email* untuk dijadikan pertimbangan tindakan apa yang akan dilakukan pada *email* tersebut.

## 2.28 *Domain-Based Message Authentication, Reporting & Conformance (DMARC)*

Domain-based Message Authentication, Reporting, and Conformance (DMARC) adalah metode autentikasi email standar. DMARC membantu administrator organisasi mencegah peretas dan penyerang lain melakukan spoofing terhadap organisasi dan domain. Spoofing adalah jenis serangan yang memalsukan alamat Dari dalam pesan email. Pesan palsu seolah tampak berasal dari organisasi atau domain yang ditiru identitasnya.

DMARC juga memungkinkan Anda meminta laporan dari server email yang menerima pesan dari organisasi atau domain Anda. Laporan ini berisi informasi untuk membantu Anda mengidentifikasi kemungkinan masalah autentikasi dan aktivitas berbahaya untuk pesan yang dikirim dari domain Anda (Nightingale 1945).



**Gambar 2. 7 Cara Kerja DMARC**  
Sumber: Zimbra Incorporation, 2005

DMARC melalukan pengecekan melalui mekanisme yang sama dengan SPF Record dan DKIM Record. Saat pengguna mengirim email (melalui server indoglobal.com), maka mail server akan membubuhkan tandatangan yang dapat dicek oleh mail server penerima untuk memastikan keabsahan email tersebut (Nightingale 1945).

Mail server penerima akan melakukan pengecekan melalui dua cara:

- Dengan mekanisme SPF, yaitu dengan cara melihat IP address mail server yang mengirimkan email. Jika IP address tersebut tertera pada record SPF, maka email tersebut dianggap sah (Nightingale 1945).
- Dengan mekanisme DKIM, yaitu dengan cara melihat tanda tangan yang ada di email. Jika tanda tangan di email sesuai dengan yang dipublikasikan pada record DKIM, maka email dianggap sah (Nightingale 1945).

Pengecekan DMARC dianggap sah jika paling tidak salah satu dari kondisi di atas terpenuhi.

### **2.29 Gmail**

*Gmail* adalah layanan *email* yang intuitif dan efisien. *Gmail* menyediakan penyimpanan sebesar 15 GB, dengan lebih sedikit *spam*, dan dapat diakses melalui perangkat seluler (*Gmail* .n.d).

### **2.30 Emkei's Mailer**

*Emkei's Mailer* adalah *Mailer* palsu *online* gratis dengan berbagai fitur seperti lampiran, enkripsi, *Editor HTML*, dan pengaturan lanjutan. (*Emkei's Mailer*, 2009).

*Emkei's Mailer* dapat digunakan untuk mengirim *email spoofing* dengan memalsukan alamat *email* pengirim pesan. *Emkei's Mailer* dapat diakses menggunakan *browser* dengan alamat *domain* www.emkei.cz. *Emkei's Mailer* dapat diakses secara gratis sehingga memberikan kemudahan dalam mengirim *email spoofing*.

### **2.31 Yahoo! Mail**

Yahoo! Mail merupakan sebuah penyedia surat elektronik (webmail) dari Yahoo!. Yahoo! Mail merupakan penyedia surat elektronik terbesar di internet dengan jutaan pengguna. Saingan utama Yahoo! Mail ialah Windows Live Hotmail, Gmail dan AOL Mail (Arrington, 2006).

## **BAB III**

### **METODOLOGI DAN PERANCANGAN**

Metode penelitian yang digunakan adalah *Network Development Life Cycle* (*NDLC*). Dari enam tahapan yang ada pada *NDLC*, penulis hanya menggunakan 3 tahapan yaitu *Analysis, Design, Simulation Prototyping* (Hanif 2018).

#### **3.1. Tahap Analisa (*Analysis*)**

Pada fase ini penulis melakukan pengumpulan data dengan cara studi literatur, yaitu penulis membaca artikel ilmiah, buku, dan jurnal untuk mendapatkan informasi mengenai *email spam*, *email spoofing*, dan *virus*. Data-data yang telah terkumpul kemudian dianalisa. Tahap ini terdiri dari dua bagian yaitu pengumpulan data dan analisa data (Hanif 2018).

##### **3.1.1. Pengumpulan Data**

Pada tahap pengumpulan data, penulis menggunakan metode studi literatur yaitu dengan mempelajari beberapa jurnal ilmiah yang membahas tentang *email spam*, *email spoofing*, dan *virus*, selain itu penulis juga menggunakan *e-book* yang membahas tentang *email spam*, *virus*, dan *email spoofing*. Setelah membaca beberapa jurnal ilmiah diperoleh informasi tentang beberapa jurnal ilmiah yang berkaitan dengan *email spam*, *virus*, dan *email spoofing* seperti terlihat pada tabel 3.1 berikut.

**Tabel 3. 1 Jurnal Ilmiah Tentang *Email Spam, Spoofing, dan Virus***

No	Penulis	Tahun	Judul	Pembahasan
1	Andri Lesmana Suryana, R. Reza El Akbar, dan Nur Widiyasono	2016	Investigasi <i>Email Spoofing</i> dengan Metode <i>Digital Forensics Research Workshop (DFRWS)</i>	Mengidentifikasi <i>email spoofing</i> menggunakan metode <i>DFRWS</i>

No	Penulis	Tahun	Judul	Pembahasan
2	Naufal hanif.S.Kom	2018	Analisa Penerapan <i>Domainkeys Identified Mail (Dkim), Sender Policy Framework (Spf), Anti Spam, Dan Anti Virus Pada Mail Server</i>	menganalisa penerapan <i>DKIM, SPF, Anti Spam dan Anti Virus</i> pada <i>mail server</i> agar <i>mail server</i> terhindar dari <i>email spam, virus</i> dan aktifitas <i>spoofing</i>
3	Andrian Maftuh Nadzifan, Farih Nazihullah	2018	Aplikasi Untuk Deteksi Adanya Spoof Pada Email	Meneruskan penelitian sebelumnya <i>Forensic Analysis of E-mail Address Spoofing</i> dengan algoritma deteksi.
4	Abidarin Rosidi, Heri Sismoro, Emha Taufiq Luthfi, Hanif Al Fatta, Hastari Utama	2016	Data Manajemen Dan Teknologi Informasi	mendeteksi adanya <i>email spoofing</i> , maka perlu adanya investigasi forensik email terhadap <i>email spoofing</i> .
5	Daniel Adi Putra Sitorus, Harun Mukhtar, Yulia Fatma	2020	Analisa Dan Implementasi Security Mail Server	Analisis dan implementasi serangan email spam pada mail server zimbra.

### **3.1.2. Analisa Data**

Berdasarkan hasil dari pengumpulan data maka dapat diperoleh hasil analisa sebagai berikut:

1. Jurnal ilmiah pertama membahas tentang investigasi *email spoofing* menggunakan metode *DFRWS* yaitu dengan melakukan pengecekan *header email* secara manual.
2. Jurnal ilmiah kedua pembahasan tentang penerapan protokol *DomainKeys Identified Mail* dapat mencegah *email spoofing* dengan cara melakukan otentikasi menggunakan metode pencocokan *private key* dan *public key* (*Asymmetric keys*). Sedangkan penerapan protokol *Sender Policy Framework* dapat mencegah *email spoofing* dengan cara melakukan otorisasi menggunakan metode pencocokan alamat *IP server* pengirim. Sebaliknya penerapan *SpamAssassin*, *ClamAV*, dan *Amavisd-New* dapat mencegah masuknya *email spam* dan *virus* dengan cara melakukan pengecekan *header*, *body*, dan *attachment email*.
3. Jurnal ilmiah ketiga membahas tentang Meneruskan penelitian sebelumnya *Forensic Analysis of E-mail Address Spoofing* dengan algoritma deteksi.
4. Jurnal ilmiah keempat membahas tentang mendeteksi adanya *email spoofing*, maka perlu adanya investigasi forensik email terhadap *email spoofing*.
5. Jurnal ilmiah kelima membahas tentang Analisis dan implementasi serangan email spam pada mail server zimbra.
6. Penanganan *email spoofing* belum menerapkan metode otentikasi dan otorisasi untuk menambah informasi pada *email header*.
7. Belum terdapat uji coba *ClamAV* sebagai *anti virus* pada *mail server*.

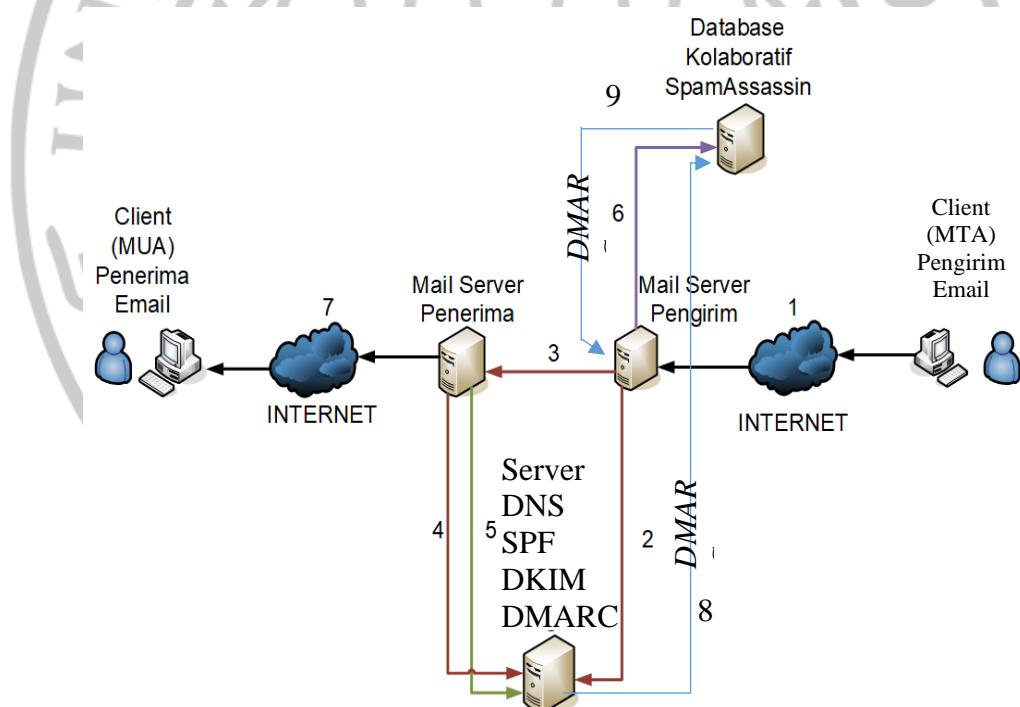
Dari hasil analisa tersebut maka mendorong penulis untuk melakukan penelitian tentang Analisa Penerapan *Dmarc* Yang Diintegrasikan Dengan *Anti Spam* Dan *Anti Virus* Untuk Pengamanan Mail Server.

### 3.2. Tahap Desain (*Design*)

Tahap ini terdiri dari 4 (empat) bagian yaitu rancangan sistem *filtering email spam, virus, dan spoofing*, rancangan jaringan ujicoba, rancangan pengalaman IP, rancangan akun *email*, serta kebutuhan perangkat keras dan perangkat lunak (Hanif 2018).

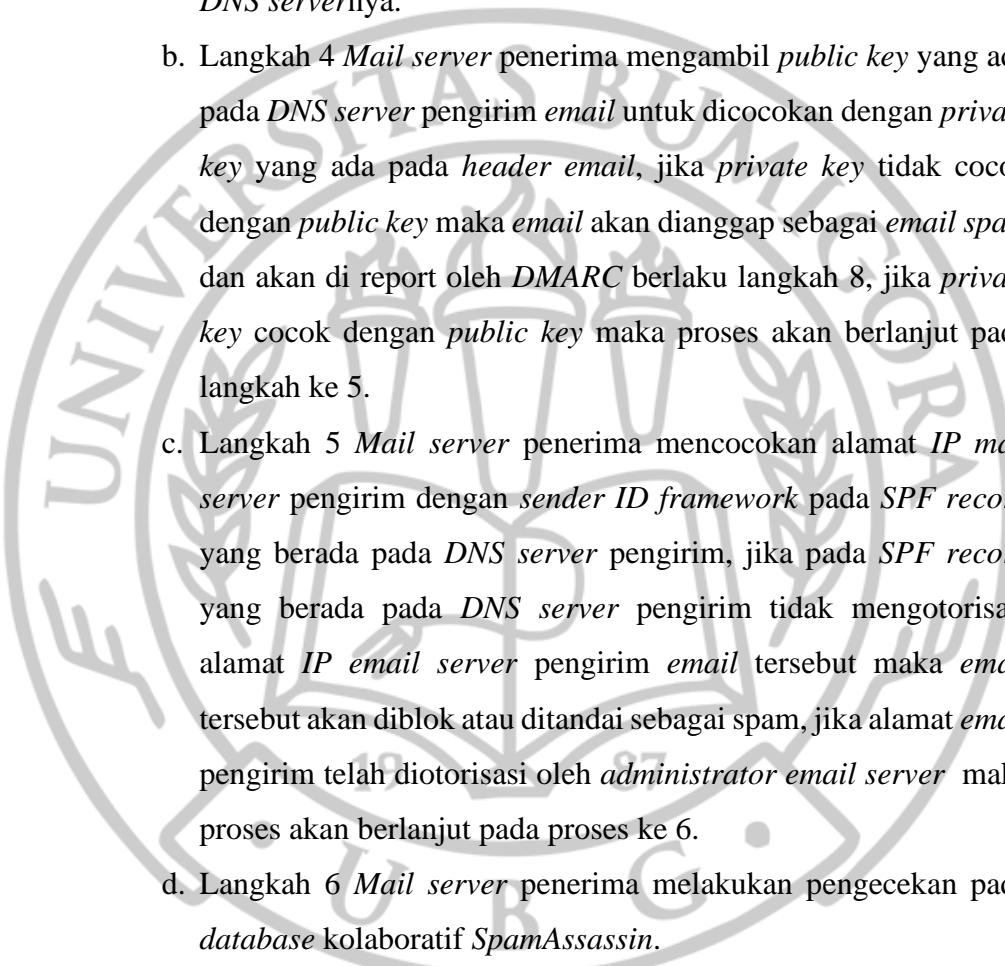
#### 3.2.1 Rancangan Sistem *Filtering Email Spam, Virus dan Spoofing*

Rancangan sistem *filtering email spam, virus* dan *spoofing* yang digunakan seperti terlihat pada gambar 3.1 berikut.



Gambar 3.1 Rancangan Sistem *Filtering Email Spam, Spoofing, dan Virus*

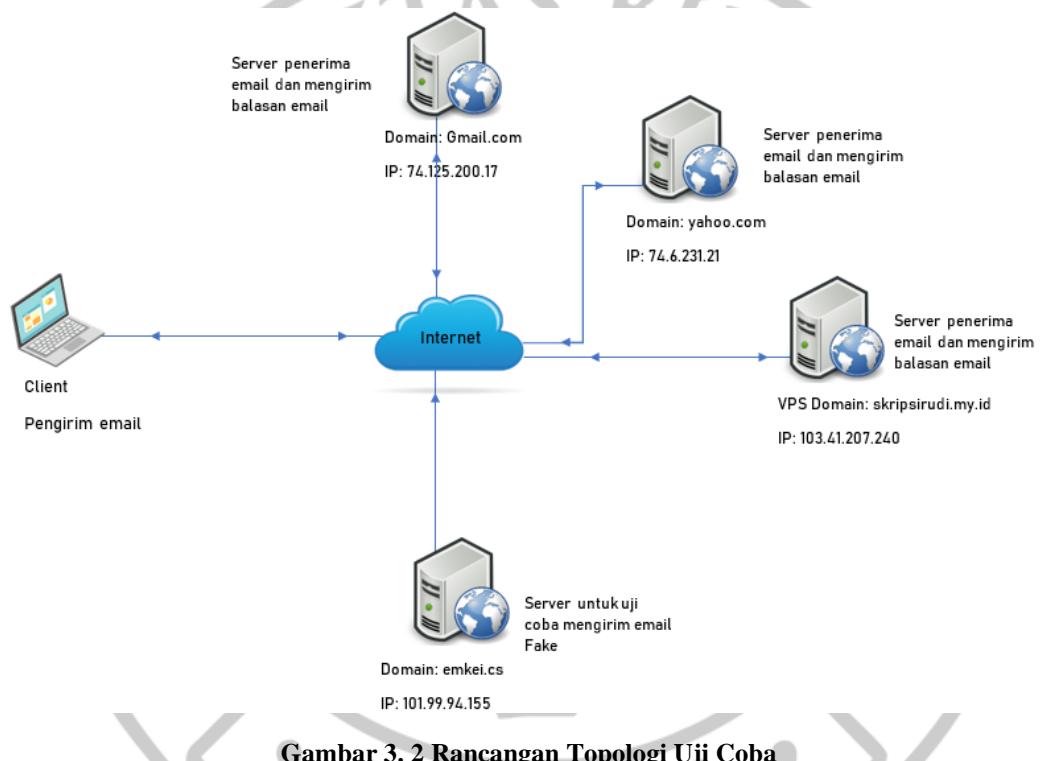
Berdasarkan gambar 3.1 tersebut maka rancangan sistem *filtering email spam, virus* dan *email spoofing* dapat dijelaskan sebagai berikut.

- 
- a. Langkah 1 *user* mengirim *email* dengan menggunakan *Mail User Agent* berbasis web (*Roundcube*), *user* mengakses *Roundcube* menggunakan *browser*.
  - b. Langkah 2 *Mail server* pengirim meneruskan *email* ke *mail server* penerima dengan menambahkan *private key* pada *header email*.
  - c. Langkah 3 *Mail server* pengirim mempublish *public key* pada *DNS servernya*.
  - b. Langkah 4 *Mail server* penerima mengambil *public key* yang ada pada *DNS server* pengirim *email* untuk dicocokan dengan *private key* yang ada pada *header email*, jika *private key* tidak cocok dengan *public key* maka *email* akan dianggap sebagai *email spam* dan akan di report oleh *DMARC* berlaku langkah 8, jika *private key* cocok dengan *public key* maka proses akan berlanjut pada langkah ke 5.
  - c. Langkah 5 *Mail server* penerima mencocokan alamat *IP mail server* pengirim dengan *sender ID framework* pada *SPF record* yang berada pada *DNS server* pengirim, jika pada *SPF record* yang berada pada *DNS server* pengirim tidak mengotorisasi alamat *IP email server* pengirim *email* tersebut maka *email* tersebut akan diblok atau ditandai sebagai spam, jika alamat *email* pengirim telah diotorisasi oleh *administrator email server* maka proses akan berlanjut pada proses ke 6.
  - d. Langkah 6 *Mail server* penerima melakukan pengecekan pada *database kolaboratif SpamAssassin*.
  - e. Proses pemfilteran *email spam* selanjutnya adalah menggunakan *SpamAssassin* dan *ClamAV* sebagai *anti spam* dan *anti virus* *email* dengan *Amavisd-New* sebagai penghubung antara *SMTP server* dengan *SpamAssassin* dan *ClamAV*.
  - f. Proses report *email* dari *DMARC* yang dikirim ke folder spam, kemudian proses 9 yang mengirim balik *email spam* ke pengirim semula.

Keterangan: garis merah mewakili proses *DKIM* (nomor 2, 3, dan 4), garis hijau mewakili proses *SPF* (nomor 5), dan garis ungu mewakili proses *SpamAssassin* (nomor 6), (no 8 dan 9) garis biru mewakili *protocol DMARC*.

### 3.2.2 Rancangan jaringan Uji coba

Rancangan jaringan uji coba yang digunakan seperti terlihat pada gambar 3.2 berikut.



Rancangan ini diimplementasikan menggunakan *VPS* yang disewa pada penyedia layanan *VPS* dan pada *VPS* telah terinstal sistem operasi *CentOS Linux release 7*, *VPS* yang telah disewa diberikan satu alamat *IP public* oleh penyedia layanan *VPS* yaitu 103.41.207.240. Pada *VPS* akan dilakukan instalasi *CentOS Web Panel*, konfigurasi *DNS server*, konfigurasi *Mail server*, dan pada komputer *client* telah terinstal system operasi windows 10 dan aplikasi browser Google Chrome untuk mengakses *Mail User Agent* berbasis web (Roundcube).

### **3.2.3 Rancangan Pengalamatan IP**

Pengalamatan *IP* merupakan salah satu bagian yang penting karena merupakan suatu identitas pengalamatan suatu *interface*. Berikut adalah pengalamatan *IP* pada masing-masing *interface* agar dapat saling berkomunikasi antar perangkat yang terhubung (Hanif 2018). Pengalamatan *IP* dapat dilihat seperti pada tabel 3.2 berikut.

**Tabel 3.2 Pengalamatan IP**

No	Perangkat	IP Address	Network	Interface
1	<i>DNS Server, HTTP Server, SMTP Server, POP3/IMAP Server (VPS)</i>	192.168.43.24/25	192.168.43.1	eth0
2	<i>Client</i>	<i>DHCP</i>	<i>DHCP</i>	-

### **3.2.4 Rancangan Akun Email**

Berikut adalah kebutuhan akun *email* untuk mendukung apa yang akan dilakukan dalam membangun atau mempersiapkan implementasi seperti terlihat pada tabel 3.3 berikut, (Hanif 2018).

**Tabel 3.3 Kebutuhan Akun Email**

No	Alamat Email	Domain
1	Rudi.masterqq3@gmail.com	gmail.com
2	Rudi.masterqq3@yahoo.com	yahoo.com
3	Root@SkripsiRudi.my.id	SkripsiRudi.my.id
4	admin@ SkripsiRudi.my.id	SkripsiRudi.my.id
5		
6		

### **3.2.5 Kebutuhan Perangkat Keras dan Perangkat Lunak**

(Hanif 2018) Berikut adalah kebutuhan perangkat keras dan perangkat lunak untuk mendukung apa yang akan dilakukan dalam membangun atau mempersiapkan implementasi yaitu:

## 1. Kebutuhan Perangkat Keras

Satu unit *VPS* dengan spesifikasi seperti terlihat pada tabel 3.4 berikut.

**Tabel 3.4 Spesifikasi *VPS***

Komponen	Spesifikasi
<i>CPU</i>	<i>Virtual CPU</i> <i>2 Core</i>
<i>RAM</i>	2 GB
<i>Hard Drive</i>	40 GB

Satu unit laptop dengan spesifikasi seperti terlihat pada tabel 3.5 berikut.

**Tabel 3.5 Spesifikasi *Client***

Komponen	Spesifikasi
<i>CPU</i>	Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz - 1.80 GHz
<i>RAM</i>	12 GB
<i>Hard Drive</i>	1 TB

## 2. Kebutuhan Perangkat Lunak

Adapun perangkat lunak yang dibutuhkan adalah sebagai berikut:

- a. *Linux CentOS release 7* sebagai sistem operasi *VPS*.
- b. *CentOS Web Panel* sebagai *tool* untuk memudahkan dalam melakukan konfigurasi *server*.
- c. Dovecot sebagai *Mail Delivery Agent*.
- d. Postfix sebagai *Mail Transfer Agent*.
- e. Roundcube sebagai *Mail User Agent*.
- f. Apache sebagai web *server*.
- g. Bind9 sebagai *DNS server*.
- h. Microsoft Windows 10 sebagai sistem operasi *client*.

- i. Google Chrome sebagai *browser client* untuk mengakses *Roundcube*.

### **3.3. Tahap Simulasi (*Prototyping*)**

(Hanif 2018) Tahap ini terdiri dari 2 bagian yaitu instalasi dan konfigurasi pada *VPS* dan *client* serta melakukan uji coba menggunakan berbagai skenario dan memverifikasi hasil uji coba tersebut.

Uji coba pertama dilakukan dengan mengirim *email spoofing* melalui *Emkei's Mailer* dengan mengatasnamakan salah satu *user* yang berada pada *domain* skripsirudi.my.id kemudian mengirim *email spoofing* tersebut ke *mail server Gmail, Yahoo! Mail*, dan skripsirudi.my.id setelah penerapan protocol *DMARC, DKIM* dan *SPF* pada *mail server* skripsirudi.my.id (Hanif 2018).

Uji coba kedua dilakukan untuk menguji kinerja *Anti Spam* pada *mail server* skripsirudi.my.id dengan mengirim *email spam* melalui *Emkei's Fake Mailer, Gmail*, dan *Yahoo! Mail* kemudian mengirim *email spam* tersebut ke salah satu *user* yang berada pada *mail server* skripsirudi.my.id sebelum dan setelah penerapan *SpamAssassin* (Hanif 2018).

Uji coba ketiga dilakukan untuk menguji kinerja *anti virus* pada *mail server* skripsirudi.my.id dengan cara mengirim *email* yang mengandung *virus* melalui *Emkei's Fake Mailer, Gmail*, dan *Yahoo! Mail* ke salah satu *user* yang ada pada *mail server* skripsirudi.my.id setelah penerapan *ClamAV* (Hanif 2018).

Uji coba ketiga dilakukan dengan cara membandingkan *header email* yang dikirim oleh salah satu *user* yang berada pada *mail server* skripsirudi.my.id ke *Gmail, Yahoo! Mail*, dan skripsirudi.my.id setelah penerapan protocol *DMARC, DKIM, SPF, Anti Spam*, dan *Anti Virus* (Hanif 2018).

### **3.3.1. Instalasi Dan Konfigurasi**

Instalasi dan konfigurasi *DKIM*, *SPF*, *Anti Spam*, *Anti Virus* dan *DMARC* dilakukan pada *VPS* yang berfungsi untuk memfilter *email spam* dan *virus* yang masuk serta untuk mencegah adanya *email spoofing* yang mengatasnamakan skripsi.rudi.my.id, sedangkan pada komputer *client* sudah terinstal sistem operasi Windows 10 dan browser Google Chrome untuk mengakses *Mail User Agent* berbasis web (Roundcube), *client* harus terkoneksi dengan jaringan *internet* agar dapat mengakses *Mail User Agent* yang telah disediakan oleh *mail server* skripsi.rudi.my.id. (Hanif 2018).

### **3.3.2. Uji Coba**

Pada tahap ujicoba ini terdiri dari 2 bagian yaitu verifikasi konfigurasi dan ujicoba menggunakan berbagai skenario. Verifikasi konfigurasi dilakukan untuk memverifikasi fungsi *DNS server* dan *Mail server* dengan melakukan *nslookup* untuk memverifikasi fungsi *DNS server* dan melakukan pengiriman *email* antar pengguna yang berada pada *mail server* yang telah dibangun serta melakukan pengiriman *email* dari *server* yang telah dibangun ke *email server* yang lainnya untuk memverifikasi fungsi *Mail server*. Sedangkan skenario ujicoba yang dilakukan meliputi pembuatan skenario yang terdiri dari beberapa skenario seperti uji coba sebelum diterapkannya *filtering email spam*, *virus*, dan *spoofing*, serta ujicoba sesudah diterapkannya *filtering email spam*, *virus* dan *spoofing*.

## BAB IV

### HASIL DAN PEMBAHASAN

Bab ini memuat tentang pembahasan dan instalasi, konfigurasi mail server, serta Analisa dan hasil uji coba.

#### 4.1 Hasil Instalasi Dan Konfigurasi

Pada tahap hasil dan implementasi ini terdiri dari dua bagian yaitu hasil instalasi dan konfigurasi *server* dan hasil konfigurasi *client* (Hanif 2018).

##### 4.1.1 Hasil Instalasi Dan Konfigurasi *Server*

Tahap instalasi dan konfigurasi server berisikan instalasi CentOS Web Panel, konfigurasi DNS server, dan konfigurasi Mail server. Server yang digunakan pada penelitian ini adalah Virtual Private Server yang telah di sewa pada salah satu penyedia jasa layanan VPS, Alamat IP VPS yang diberikan oleh penyedia jasa layanan VPS adalah 103.41.207.240 dengan sistem operasi Linux CentOS release 7-9, pada VPS telah terinstal SSH Server agar VPS dapat diakses melalui perangkat lain melalui jaringan internet seperti terlihat pada gambar 4.1 berikut: (Hanif 2018).

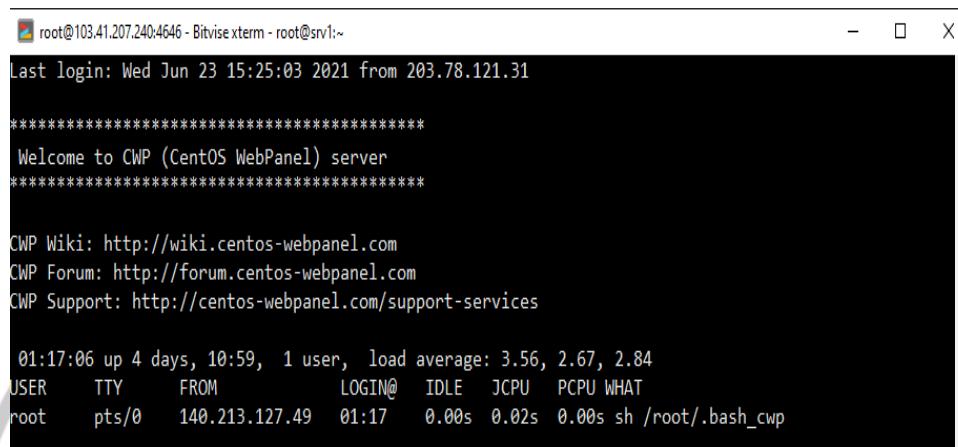
```
[root@srv1 ~]# rpm --query centos-release
centos-release-7-9.2009.1.el7.centos.x86_64
[root@srv1 ~]#
```

Gambar 4.1 Linux CentOS release 7-9

##### 4.1.2 Hasil Instalasi *CentOS Web Panel*

CentOS Web Panel digunakan untuk memudahkan dalam melakukan instalasi dan konfigurasi server karena proses instalasi server akan dilakukan secara otomatis dan proses konfigurasi server dapat dilakukan dengan mudah melalui halaman konfigurasi CentOS Web Panel yang berbasis web. Tahap instalasi CentOS Web Panel berisikan tiga perintah yaitu perintah untuk masuk pada direktori src yang bertujuan sebagai lokasi penyimpanan file *installer CWP* dengan perintah `#cd /usr/local/src`, perintah untuk mendownload file *installer CWP* versi terbaru

dengan perintah `#wget http://centos-webpanel.com/cwp-latest`, perintah untuk menginstal *file installer* yang telah di *download* dengan perintah `#sh cwp-latest`, hasil instalasi CWP seperti terlihat pada gambar 4.2 berikut: (Hanif 2018).



```
root@103.41.207.240:4646 - Bitvise xterm - root@srv1:~  
Last login: Wed Jun 23 15:25:03 2021 from 203.78.121.31  
*****  
Welcome to CWP (CentOS WebPanel) server  
*****  
  
CWP Wiki: http://wiki.centos-webpanel.com  
CWP Forum: http://forum.centos-webpanel.com  
CWP Support: http://centos-webpanel.com/support-services  
  
01:17:06 up 4 days, 10:59, 1 user, load average: 3.56, 2.67, 2.84  
USER TTY FROM LOGIN@ IDLE JCPU PCPU WHAT  
root pts/0 140.213.127.49 01:17 0.00s 0.02s 0.00s sh /root/.bash_cwp
```

Gambar 4.2 Hasil Instalasi CWP

#### 4.1.3 Hasil Konfigurasi *DNS Server*

Tahap konfigurasi DNS server berisikan konfigurasi interface, konfigurasi name server dan konfigurasi file revers lookup zone. Konfigurasi interface sudah dilakukan oleh pihak penyedia jasa VPS, yang perlu dilakukan penyesuaian kebutuhan *mail server* yang di atur pada CWP yang telah di install (Hanif 2018).

```
# Created by cloud-init on instance boot automatically, do not edit.  
#  
BOOTPROTO=none  
DEFROUTE=yes  
DEVICE=eth0  
GATEWAY=103.41.207.1  
HWADDR=8a:9c:46:3d:99:de  
IPADDR=103.41.207.240  
NETMASK=255.255.255.0  
ONBOOT=yes  
STARTMODE=auto  
TYPE=Ethernet  
USERCTL=no
```

Gambar 4.3 Konfigurasi Interface

Untuk melakukan konfigurasi *name server* dilakukan pada halaman konfigurasi *CWP* dengan memilih menu *DNS Functions* kemudian pilih menu *Edit Nameservers IPs* seperti terlihat pada gambar 4.4 berikut. (Hanif 2018)

### Edit Nameservers

Don't forget to restart BIND DNS Server once you are done with the changes.  
Changes made here will apply only to the newly added domains.  
[For more information's check this instructions.](#)

**srv1.skripsirudi.my.id** resolves to **103.41.207.240**  
**srv2.skripsirudi.my.id** resolves to **103.41.207.240**

Edit Name Servers		c + x
Name Server 1:	<input type="text" value="srv1.skripsirudi.my.id"/> Enter NS1 SubDomain, example: ns1.centos-webpanel.com	<input type="text" value="103.41.207.240"/> Enter NS1 IPv4 Address
Name Server 2:	<input type="text" value="srv2.skripsirudi.my.id"/> Enter NS2 SubDomain, example: ns2.centos-webpanel.com	<input type="text" value="103.41.207.240"/> Enter NS2 IPv4 Address
Options:	<input checked="" type="checkbox"/> Update DNS zone file, overwrite ns1 and ns2 zone files with the new changes <input checked="" type="checkbox"/> Restart DNS Server, restart DNS server after making changes	
<b>Save changes</b>		

Gambar 4.4 Konfigurasi Name Server

Untuk dapat memetakan nama *domain* ke alamat *IP* dan agar *domain* dapat diakses dengan nama alias maka perlu dibuat *file forward-lookup zone* dengan cara masuk pada menu *Domains* lalu masuk pada *sub menu Add Domain* seperti terlihat pada gambar 4.5 berikut. (Hanif 2018)

## Add Domain

Path must be /home/USERNAME eg. /home/mywebsite/...  
If you enter / then the home path will be eg. /home/mywebsite/  
If you enter /public\_html/addondomain1.com then the path will be /home/mywebsite/public\_html/addondomain1.com

Add a New Domain

Add Domain: skripsirudi.my.id  
Enter domain name without www.

to User: rudi  
Select user to which you want to add this domain

Folder Path: /public\_html  
Enter path to a folder in user homedir

AutoSSL: Install SSL certificate, domain and www. subdomain must be pointed to the server!

Create

Gambar 4.5 Konfigurasi Domain

Jika konfigurasi *domain* berhasil maka secara otomatis akan terbuat *file forward-lookup zone* yang diberi nama skripsirudi.my.id.db, pada baris record @ IN A 103.41.207.240 yang berfungsi untuk memetakan nama *host* ke alamat *IP*, pada baris record @ IN NS srv1.skripsirudi.my.id. yang berfungsi untuk memetakan sebuah nama domain ke dalam satu daftar dari server DNS untuk domain skripsirudi.my.id, pada baris record @ IN MX 10 srv1.skripsirudi.my.id. yang berfungsi untuk memetakan sebuah nama domain ke dalam daftar mail exchange server untuk domain skripsirudi.my.id, pada 4 baris terakhir merupakan record CNAME atau Canonical Name yang berfungsi agar nama domain skripsirudi.my.id dapat diakses menggunakan nama alias www.skripsirudi.my.id, dan mail.skripsirudi.my.id seperti terlihat pada gambar 4.6 berikut. (Hanif 2018)

```

$TTL 14400
@ 86400 IN SOA srv1.skripsirudi.my.id. postmaster.skripsirudi.my.id. (
    2021062472 ; serial, todays date+todays
    3600       ; refresh, seconds
    7200       ; retry, seconds
    1209600    ; expire, seconds
    86400 )    ; minimum, seconds
@ 86400 IN NS   srv1.skripsirudi.my.id.
@ 86400 IN NS   srv2.skripsirudi.my.id.
@ IN A 103.41.207.240
@ IN NS srv1.skripsirudi.my.id.
@ IN MX 10 srv1.skripsirudi.my.id.

srv1 14400 IN A 103.41.207.240
srv2 14400 IN A 103.41.207.240
srv1.skripsirudi.my.id. 14400 IN A 103.41.207.240
srv2.skripsirudi.my.id. 14400 IN A 103.41.207.240

```

Gambar 4.6 File skripsirudi.my.id.db

Untuk dapat memetakan alamat *IP* ke nama *domain* maka perlu dibuat *file reverse-lookup zone* pada terminal dengan perintah `#nano /var/named/ 207.41.103.in-addr.arpa.db`. pada baris *record* @ IN NS srv1.skripsirudi.my.id. yang berfungsi untuk memetakan sebuah nama domain ke dalam satu daftar dari server DNS untuk domain skripsirudi.my.id, pada 2 baris setelahnya terdapat record pointer yang berfungsi untuk memetakan nama domain ke dalam alamat IP seperti terlihat pada gambar 4.7 berikut. (Hanif 2018)

```

$TTL 144000
@ IN SOA srv1.skripsirudi.my.id. root.skripsirudi.my.id. (
    2017082100 ;serial, todays date+todays
    86400 ;refresh, seconds
    7200 ;retry, seconds
    3600000 ;expire, seconds
    86400 ;minimum, seconds
)
@ IN NS srv1.skripsirudi.my.id.
240 IN PTR skripsirudi.my.id.
240 IN PTR srv1.skripsirudi.my.id.
240 IN PTR srv2.skripsirudi.my.id.

```

Gambar 4.7 File 207.41.103.in-addr.arpa.db

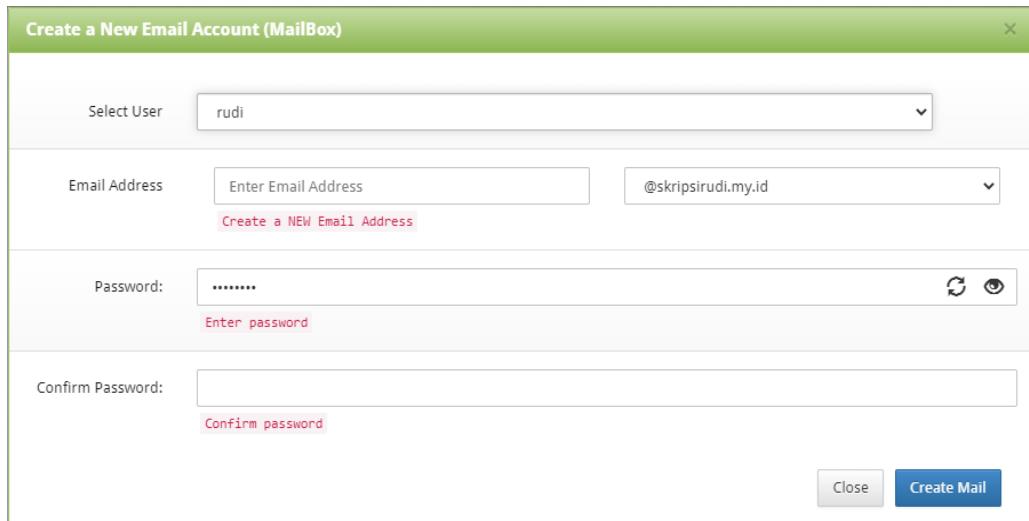
Pada file named.conf ditambahkan beberapa perintah yaitu zone "skripsirudi.my.id" adalah nama domain yang akan digunakan, selanjutnya yaitu type master adalah tipe DNS server yaitu primary DNS, pada baris lokasi file forward zone, kemudian pada baris zone "207.41.103.in-addr.arpa" IN adalah lingkup network dalam domain yang akan digunakan sebagai reverse, baris type master adalah tipe DNS server yaitu primary DNS, baris 8 adalah lokasi file reverse zone, setelah semua konfigurasi selesai disarankan untuk merestart DNS server yang terdapat pada dashboard CPW pada bagian server service. Seperti terlihat pada gambar 4.8 berikut. (Hanif 2018)

```
zone "." IN {  
    type hint;  
    file "named.ca";  
};  
zone "207.41.103.in-addr.arpa" IN {  
    type master;  
    file "/var/named/207.41.103.in-addr.arpa.db";  
    allow-update { none };  
};
```

Gambar 4.8 File named.conf

#### 4.1.4 Hasil Konfigurasi Mail Server

Untuk dapat mengecek fungsi mail server maka perlu membuat akun email pada mail server dengan cara masuk pada menu Email kemudian pilih sub menu Add Email Account seperti terlihat pada gambar 4.9 berikut. (Hanif 2018)



Gambar 4.9 Membuat Akun Email

Proses instalasi *DKIM*, *SPF*, *Anti Spam*, dan *Anti Virus* dilakukan pada menu *Email* kemudian masuk pada sub menu *MailServer Manager*, centang check box AntiSpam/AntiVirus, Install DKIM & SPF, dan rDNS Check untuk melakukan instalasi Spam-Assassin, ClamAV, Amavis, DKIM, dan SPF seperti terlihat pada gambar 4.10 berikut. (Hanif 2018)

#### ■ Rebuild Postfix Configuration [Last Rebuild: 2021-04-07 16:00:19]

AntiSpam/AntiVirus (recommended):	<input checked="" type="checkbox"/> ClamAV, Amavis & Spamassassin, Requires 2Gb+ RAM
rDNS Check (recommended):	<input checked="" type="checkbox"/> Drop all emails if no rDNS/PTR
Install DKIM & SPF (recommended):	<input checked="" type="checkbox"/> Installs DKIM & SPF, enables DKIM for New Accounts and Domains
Install Policyd (recommended):	<input type="checkbox"/> Installs Policyd, enables hourly email limit per domain.
Reject Unknown Hostname ( <b>NOT recommended</b> ):	<input type="checkbox"/> <b>WARNING:</b> Reject unless the hostname has valid DNS record.
Current Settings in Postfix:	
<pre>mydomain = skripsirudi.my.id myhostname = srv1.skripsirudi.my.id</pre>	
Hostname:	<input type="text" value="srv1.skripsirudi.my.id"/> (change hostname)
Domain:	<input type="text" value="my.id"/> (must be the main domain of the server hostname)
<input type="button" value="Rebuild Mail Server"/>	

Gambar 4.10 Instalasi *DKIM*, *SPF*, *Anti Spam*, dan *Anti Virus*

Konfigurasi *DKIM* dilakukan pada menu *Email* kemudian masuk pada sub menu *DKIM Manager*, seperti terlihat pada gambar 4.11 berikut.  
(Hanif 2018)

**Gambar 4.11 Menambah *DKIM Record* pada *File Zone***

Konfigurasi SPF dilakukan pada menu Email kemudian masuk pada sub menu SPF Manager, seperti terlihat pada gambar 4.12 berikut.(Hanif 2018)

**Gambar 4.12 Menambah *SPF Record* pada *File Zone***

Pada file skripsirudi.my.id.db akan terlihat tambahan dua baris dibagian paling bawah seperti terlihat pada gambar 4.13 berikut. (Hanif 2018)

```
skripsirudi.my.id. IN TXT "v=spf1 mx a ip4:103.41.207.240/32 a:srv1.skripsirudi.my.id -all"
default._domainkey 14400 IN TXT "v=DKIM1; k=rsa;
p=MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQC7b7KUq0Ya9jz0iDa3NF+WJquppUdry0MjttVjkAQ8eoUloMU4Y8RFst
71mdjJcQzW51P+p0iFJ2MKMz0MtLkz2xAkr98epATFKfb9GIK0Olv45WMJOFlQpHk/O76iBiKogTLcxZ7RMENpIt9TV50yf1JZ1xZi
dVEhHXgR9YGwIDAQAB"
```

**Gambar 4.13 *DKIM* dan *SPF Record***

Pada baris skripsi.rudi.my.id. IN TXT "v=spf1 mx a ip4:103.41.207.240/32 a:srv1.skripsi.rudi.my.id -all" merupakan *record SPF*, v=spf1 berarti versi *SPF* yang digunakan adalah *SPF* versi, a:srv1.skripsi.rudi.my.id berarti hanya mengizinkan pengiriman *email* dengan *hostname* srv1.skripsi.rudi.my.id, ip4: 103.41.207.240/32 yang berarti hanya mengizinkan pengiriman email dari *server* dengan alamat *IP* 103.41.207.240, -all berarti menolak semua *email* yang tidak sesuai dengan aturan tersebut, (Hanif 2018).

Pada baris *record DKIM*, dimana parameter v=DKIM1 berarti versi *DKIM* yang digunakan yaitu *DKIM* versi 1, parameter k=rsa berarti jenis kriptografi yang digunakan adalah rsa, dan parameter p yaitu *public key* yang digunakan (Hanif 2018).

(Hanif 2018) Agar srv1.skripsi.rudi.my.id menjadi *host* yang dipercaya maka harus ditambahkan *hostname* pada baris 2 pada *file TrustedHosts* dibaris yang paling bawah dengan perintah #nano /etc/opendkim/TrustedHosts seperti pada gambar 4.14 berikut.

```
#host.example.com
srv1.skripsi.rudi.my.id
#192.168.1.0/24
skripsi.rudi.my.id
```

Gambar 4.14 Konfigurasi File TrustedHosts

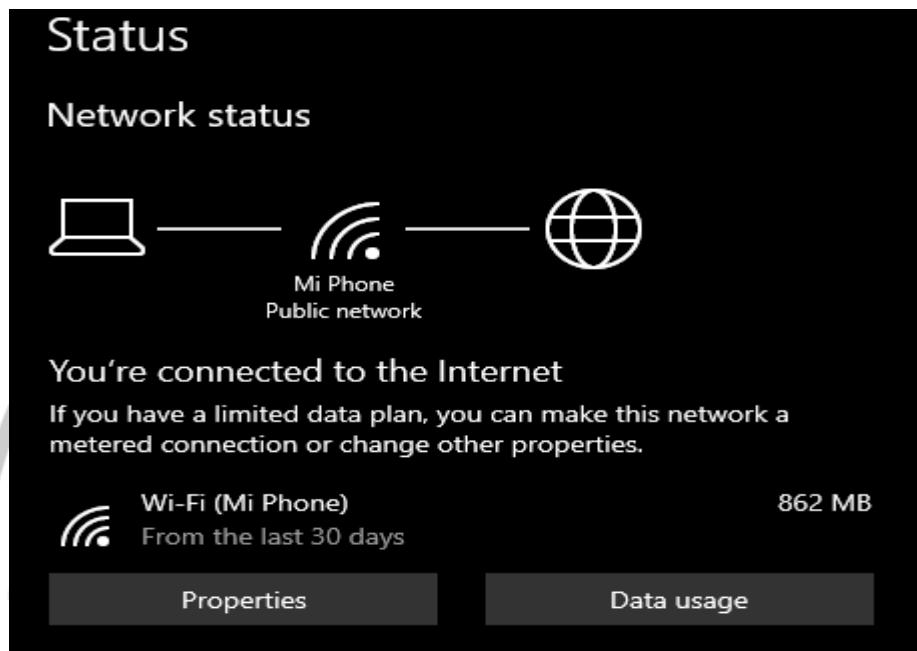
(Hanif 2018) Pada *file main.cf* ditambahkan beberapa parameter seperti smtpd\_milters = inet:127.0.0.1:8891, non\_smtpd\_milters = \$smtpd\_milters, milter\_default\_action = accept, dan milter\_protocol = 2 yang berfungsi untuk memfilter *email*, seperti terlihat pada gambar 4.15 berikut.

```
recipient_bcc_maps = proxy:mysql:/etc/postfix/mysql-virtual_vacation.cf
smtpd_milters = inet:127.0.0.1:8891
non_smtpd_milters = $smtpd_milters
milter_default_action = accept
milter_protocol = 2
tls_server_sni_maps = hash:/etc/postfix/vmail_ssl.map
```

Gambar 4.15 Konfigurasi File main.cf

#### 4.1.5 Hasil Konfigurasi *Client*

(Hanif 2018) Komputer *client* berfungsi sebagai *Mail User Agent (MUA)*, untuk dapat mengakses *mail server* maka komputer *client* harus terkoneksi dengan jaringan *internet* seperti terlihat pada gambar 4.16 berikut.



Gambar 4.16 Terhubung ke *Internet*

### 4.2 Hasil Uji Coba

Tahap uji coba ini terdiri dari 2 bagian yaitu verifikasi konfigurasi dan uji coba menggunakan berbagai macam skenario (Hanif 2018).

#### 4.2.1 Verifikasi Konfigurasi

Pada tahap hasil verifikasi konfigurasi ini dilakukan untuk mengetahui apakah hasil konfigurasi yang dilakukan sebelumnya berhasil atau tidak (Hanif 2018).

#### 4.2.2 Verifikasi Konfigurasi *DNS Server*

(Hanif 2018) Untuk mengecek fungsi forward-lookup, CNAME, reverse-lookup, dan fitur *mail exchanger* dapat digunakan perintah nslookup dan host -t mx pada terminal seperti terlihat pada gambar 4.17 berikut.

```
[root@srv1 ~]# nslookup mail.skripsirudi.my.id
Server:          8.8.8.8
Address:         8.8.8.8#53

Non-authoritative answer:
mail.skripsirudi.my.id canonical name = skripsirudi.my.id.
Name:   skripsirudi.my.id
Address: 103.41.207.240

[root@srv1 ~]# nslookup skripsirudi.my.id
Server:          8.8.8.8
Address:         8.8.8.8#53

Non-authoritative answer:
Name:   skripsirudi.my.id
Address: 103.41.207.240

[root@srv1 ~]# nslookup www.skripsirudi.my.id
Server:          8.8.8.8
Address:         8.8.8.8#53

Non-authoritative answer:
www.skripsirudi.my.id canonical name = skripsirudi.my.id.
Name:   skripsirudi.my.id
Address: 103.41.207.240

[root@srv1 ~]# host -t mx skripsirudi.my.id
skripsirudi.my.id mail is handled by 10 srv1.skripsirudi.my.id.
skripsirudi.my.id mail is handled by 0 skripsirudi.my.id.
[root@srv1 ~]# nslookup srv1.skripsirudi.my.id
Server:          8.8.8.8
Address:         8.8.8.8#53

Non-authoritative answer:
Name:   srv1.skripsirudi.my.id
Address: 103.41.207.240
```

Gambar 4.17 Verifikasi Konfigurasi *DNS Server*

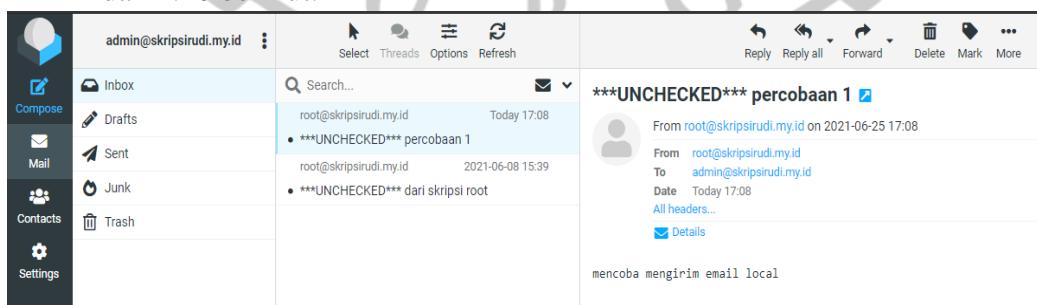
#### 4.2.3 Verifikasi Konfigurasi *Mail Server*

(Hanif 2018) Verifikasi konfigurasi *mail server* dapat dilakukan dengan cara menulis perintah `#service dovecot status` dan `#service postfix status` pada terminal seperti terlihat pada gambar 4.18 berikut.

```
[root@srv1 ~]# service dovecot status
Redirecting to /bin/systemctl status dovecot.service
● dovecot.service - Dovecot IMAP/POP3 email server
  Loaded: loaded (/usr/lib/systemd/system/dovecot.service; enabled; vendor preset: disabled)
  Active: active (running) since Sun 2021-06-20 14:17:21 WITA; 5 days ago
    Docs: man:dovecot(1)
          http://wiki2.dovecot.org/
Main PID: 1283 (dovecot)
   CGroup: /system.slice/dovecot.service
           └─ 1283 /usr/sbin/dovecot
             ├ 1297 dovecot/anvil
             ├ 1298 dovecot/log
             ├ 1305 dovecot/config
             ├ 19748 dovecot/auth
             ├ 20032 dovecot/auth -w
             ├ 26337 dovecot/auth -w
             ├ 26356 dovecot/auth -w
             ├ 26357 dovecot/auth -w
             └─ 26358 dovecot/auth -w
[root@srv1 ~]# service postfix status
Redirecting to /bin/systemctl status postfix.service
● postfix.service - Postfix Mail Transport Agent
  Loaded: loaded (/usr/lib/systemd/system/postfix.service; enabled; vendor preset: disabled)
  Active: active (running) since Sun 2021-06-20 14:17:20 WITA; 5 days ago
Main PID: 1185 (master)
   CGroup: /system.slice/postfix.service
           └─ 1185 /usr/libexec/postfix/master -w
             ├ 1191 qmgr -l -t fifo -u
             ├ 2242 tlsmgr -l -t unix -u
             ├ 2243 anvil -l -t unix -u
             ├ 20391 pickup -l -t fifo -u -o content_filter= -o receive_override_options=no_header_...
             ├ 22710 smtpd -n smtp -t inet -u -o stress= -s 2 -o content_filter=smtp-amavis:127.0.0...
             ├ 23893 smtpd -n smtp -t inet -u -o stress= -s 2 -o content_filter=smtp-amavis:127.0.0...
             ├ 24891 smtpd -n smtp -t inet -u -o stress= -s 2 -o content_filter=smtp-amavis:127.0.0...
             └─ 25371 smtpd -n smtp -t inet -u -o stress= -s 2 -o content_filter=smtp-amavis:127.0.0...
```

Gambar 4.18 Verifikasi Konfigurasi *Mail Server*

(Hanif 2018) Melakukan pengiriman *email* dengan cara mengirim *email* antar pengguna pada *mail server* yang sama dan mengirim *email* antar pengguna pada *mail server* yang berbeda seperti terlihat pada gambar 4.19 dan 4.20 berikut.



Gambar 4.19 Mengirim *Email* pada *User Email Local*



Verifikasi fungsi *DKIM*, *SPF*, dan *DMARC* dapat dilakukan dengan cara mengirim email ke alamat email rudi.masterqq3@yahoo.com seperti pada gambar 4.21 berikut.

```
Received: from 10.197.39.136
by atlas108.free.mail.bf1.yahoo.com with HTTPS; Sat, 26 Jun 2021 08:27:34 +0000
Return-Path: <admin@skripsirudi.my.id>
X-Originating-Ip: [103.41.207.240]
Received-SPF: pass (domain of skripsirudi.my.id designates 103.41.207.240 as permitted sender)
Authentication-Results: atlas108.free.mail.bf1.yahoo.com;
dkim=pass header.i=@skripsirudi.my.id header.s=default;
dkim=pass header.i=@skripsirudi.my.id header.s=default;
spf=pass smtp.mailfrom=skripsirudi.my.id;
dmarc=pass(p=QUARANTINE) header.from=skripsirudi.my.id;
X-Apparently-to: rudi.masterqq3@yahoo.com; Sat, 26 Jun 2021 08:27:35 +0000
```

Gambar 4.21 Verifikasi Fungsi *DKIM*, *SPF*, dan *DMARC*

Verifikasi fungsi ClamAV dilakukan dengan menggunakan perintah `#systemctl status clamd.service` pada remote ssh sever, seperti terlihat pada gambar 4.22 berikut.

```
[root@srv1 ~]# systemctl status clamd.service
● clamd.service - clamd scanner () daemon
   Loaded: loaded (/usr/lib/systemd/system/clamd.service; enabled; vendor preset: disabled)
   Active: active (running) since Sat 2021-06-26 19:03:16 WITA; 9s ago
     Docs: man:clamd(8)
           man:clamd.conf(5)
           https://www.clamav.net/documents/
 Main PID: 29511 (clamd)
    CGroup: /system.slice/clamd.service
             └─29511 /usr/sbin/clamd -c /etc/clamd.d/amavisd.conf --foreground=yes
```

Gambar 4.22 Verifikasi Fungsi *ClamAV*

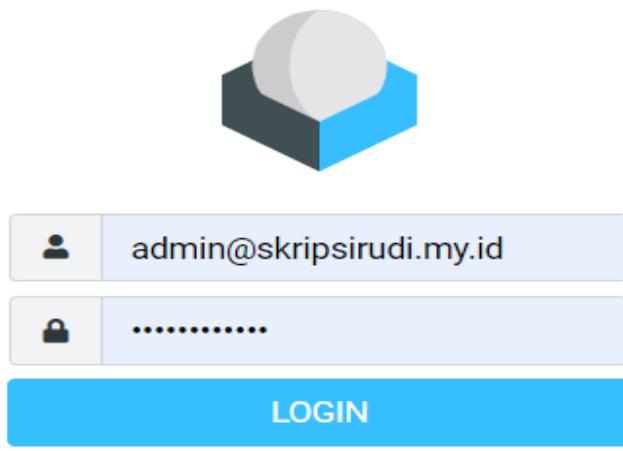
#### 4.2.4 Verifikasi Konfigurasi Client

(Hanif 2018) Verifikasi konfigurasi *client* dilakukan agar memastikan domain yang di buat telah connect secara online dengan server dengan cara melakukan ping pada *mail server* dengan perintah >ping *skripsirudi.my.id* seperti pada gambar 4.23 berikut.

```
Pinging skripsirudi.my.id [103.41.207.240] with 32 bytes of data:  
Reply from 103.41.207.240: bytes=32 time=54ms TTL=52  
Reply from 103.41.207.240: bytes=32 time=72ms TTL=52  
Reply from 103.41.207.240: bytes=32 time=57ms TTL=52  
Reply from 103.41.207.240: bytes=32 time=68ms TTL=52  
  
Ping statistics for 103.41.207.240:  
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),  
Approximate round trip times in milli-seconds:  
    Minimum = 54ms, Maximum = 72ms, Average = 62ms
```

Gambar 4.23 Ping Mail Server

(Hanif 2018) Verifikasi konfigurasi pada *client* juga dapat dilakukan dengan cara mengakses *Mail User Agent* dengan menggunakan aplikasi *browser* kemudian mengakses alamat *URL* <http://skripsirudi.my.id/webmail/> seperti terlihat pada gambar 4.24 berikut.



Roundcube Webmail

Gambar 4.24 Akses MUA *roundcube*

### **4.3 Skenario Uji Coba**

Pada tahap skenario hasil uji coba ini berisikan tentang uji coba sebelum diterapkannya filtering, otentikasi, dan otorisasi email spam.

#### **4.3.1 Uji Coba Sebelum Diterapkan Filtering, Otentikasi, dan Otorisasi**

Adapun uji coba yang dilakukan sebelum diterapkan *filtering*, otentikasi, dan otorisasi *email* adalah uji coba mengirim *email spoofing*, uji coba mengirim *email spam*, uji coba mengirim *email* yang mengandung *virus*, dan uji coba pengecekan *header email*.

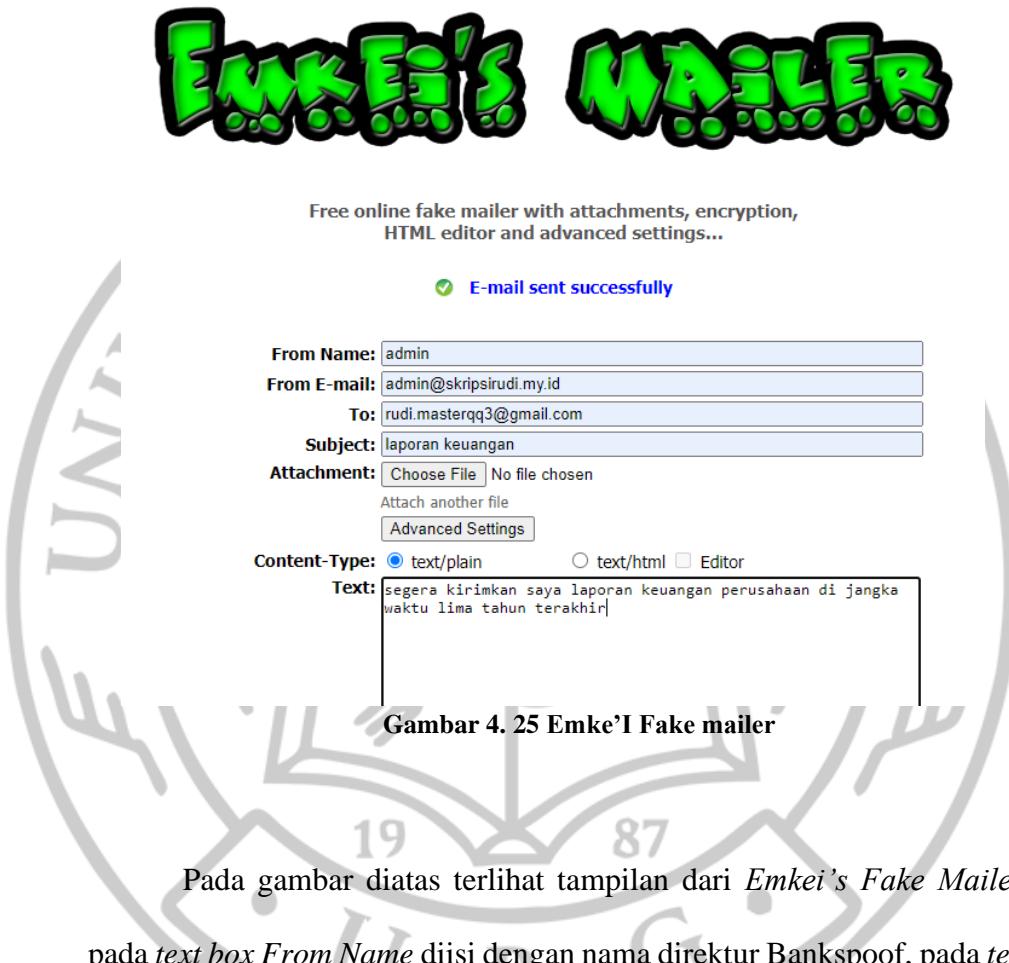
#### **4.3.2 Uji Coba mengirim *Email Spoofing***

Uji coba mengirim *email spoofing* dilakukan dengan mengirim *email spoofing* menggunakan *Emkei's Fake Mailer* ke *Gmail*, *Yahoo! Mail*, dan *skripsirudi.my.id*

#### **4.3.3 Uji Coba Mengirim *Email Spoofing* ke *Gmail***

Scenario uji coba untuk menguji protocol *DMARC*, *DKIM* dan *SPF* dilakukan dengan cara mengirim *email spoofing*, misalakan ada perusahaan bank yang bernama *Bnkspoof* yang bergerak di bidang perbankan untuk masyarakat. *Banksspoof* mempunyai pesaing dalam menjalankan bisnis perbankannya, dimna pesaing tersebut berusaha menjatuhkan *Banksspoof* dengan cara melakukan penipuan dengan mengirim *email spoofing* dengan *fake mailer*, sebelum melakukan penipuan terlebih dahulu mencari tahu *email* perusahaan *Banksspoof*, setelah mengetahui *email* dari perusahaan *Banksspoof* maka pesaing tersebut memulai aksinya mengirim *email spoofing* yang mengatasnamakan direktur *Banksspoof*, pesaing tersebut menunjukan *email* ke staf keuagan *Banksspoof*, *email spoofing* tersebut berisikan perintah mengirim laporan keuangan Bank spoof dalam jangka waktu lima tahun terahir. Setelah semua informasi yang di perlukan telah di dapat, dimana nama direkturnya admin dan nama pegawainya root maka pesaing tersebut memulai *email spoofing* dengan cara membuka situs [www.emkei.cz](http://www.emkei.cz) kepada pegawai staf keuangan root dengan alamat *email*

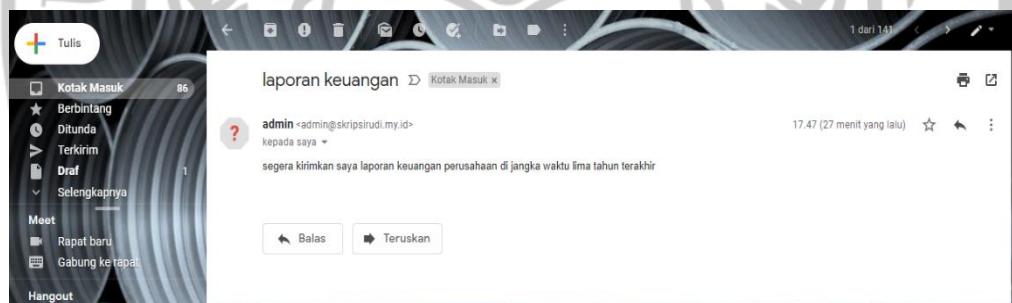
rudi.masterqq3@gmail.com dengan mengatasnamakan direktur Bankspoof yang bernama admin dengan alamat email admin@skripsirudi.my.id, isi email terbebut memerintahkan staf keuangan Bankspoof untuk mengirimkan laporan keuangan Bankspoof lima tahun terakhir, seperti pada gambar 4.25 berikut.



Gambar 4. 25 Emkei's Fake mailer

Pada gambar diatas terlihat tampilan dari *Emkei's Fake Mailer*, pada *text box From Name* diisi dengan nama direktur Bankspoof, pada *text box From E-mail* diisi dengan alamat *email* direktur Bankspoof, pada *text box To* diisi dengan alamat *email* staf keuangan Bankspoof, pada *text box Subject* diisi dengan subjek *email*, dan *text box Text* diisi dengan pesan dari *email spoofing*, jika pada *email* Bankspoof belum menerapkan protokol *DMARC*, *SPF* dan *DKIM* maka *email* tersebut berhasil terkirim dengan proses sebagai berikut:

1. Pesaing tersebut melakukan pengiriman *email spoofing* menggunakan *Emkei's Fake Mailer* dengan cara memasukan URL [www.emkei.cz](http://www.emkei.cz) pada *browser*, kemudian pada situs Emkei's Fake Mailer pesaing tersebut menuliskan alamat pengirim *email* yaitu [root@skripsirudi.my.id](mailto:root@skripsirudi.my.id) dan alamat penerima email adalah [rudi.masterqq3@gmail.com](mailto:rudi.masterqq3@gmail.com).
2. Email spoofing tersebut di akses melalui *mail server* [gmail.com](mailto:gmail.com) oleh alamat email [rudi.masterqq3@gmail.com](mailto:rudi.masterqq3@gmail.com) tanpa adanya proses otentikasi dan otorisasi oleh protokol *DMARC*, *DKIM* dan *SPF* sehingga *email spoofing* tersebut berhasil terkirim kealamat *email* [rudi.masterqq3@gmail.com](mailto:rudi.masterqq3@gmail.com) seperti terlihat pada gambar 4.26 berikut.



Gambar 4. 26 Email spoofing

Setelah *email* tersebut masuk ke *inbox* maka staf keuangan Bankspoof akan membaca *email spoofing* tersebut.

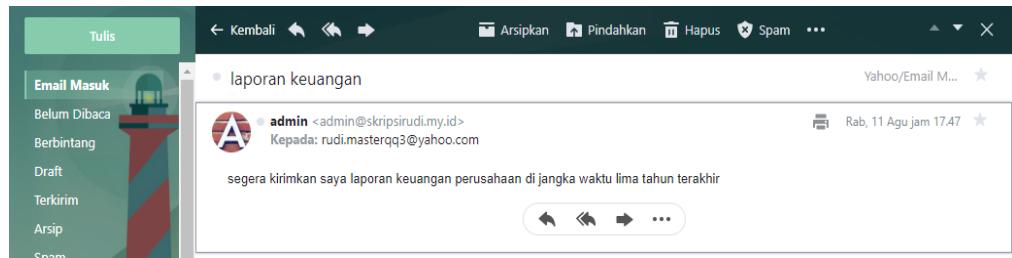
#### 4.3.4 Uji Coba Mengirim *Email Spoofing* ke *Yahoo! Mail*

Dengan menggunakan skenario yang sama seperti pada uji coba mengirim *email spoofing* ke *Gmail*, namun pada uji coba ini akan di uji pengiriman *email spoofing* pada layanan *email Yahoo! Mail* dengan mengatasnamakan salah satu *user* yang ada pada *mail server* skripsi.rudi.my.id, proses otentikasi dan otorisasi akan sama dengan proses otorisasi dan otentikasi pada uji coba pertama terlihat seperti gambar 4.27 berikut.



Gambar 4. 27 mengirim spoofing ke yahoo mail

*Email spoofing* diatas berhasil terkirim ke alamat *email* rudi.masterqq3@yahoo.com terlihat seperti pada gambar 4.28 berikut.



Gambar 4. 28 Email Spoofing Terkirim ke Yahoo! Mail

#### 4.3.5 Uji Coba Mengirim Email Spoofing pada skripsirudi.my.id

Dengan menggunakan skenario yang sama seperti pada uji coba mengirim *email spoofing* ke layanan *email Gmail*, namun pada uji coba ini akan di uji pengiriman *email spoofing* pada *mail server* skripsirudi.my.id dengan mengatasnamakan salah satu *user* yang ada pada *mail server* skripsirudi.my.id, proses otentikasi dan otorisasi akan sama dengan proses otorisasi dan otentikasi pada uji coba mengirim *email spoofing* ke *Gmail* terlihat seperti gambar 4.29 berikut.

Enkeej's MAILER

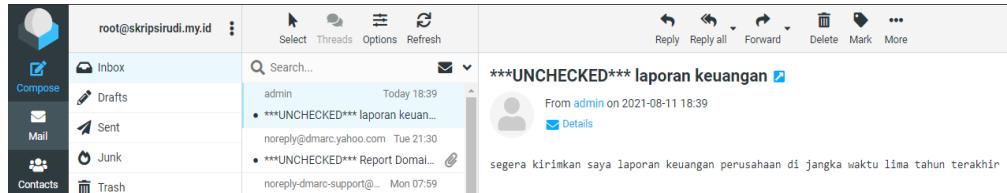
Free online fake mailer with attachments, encryption, HTML editor and advanced settings...

E-mail sent successfully

**From Name:** admin  
**From E-mail:** admin@skripsirudi.my.id  
**To:** root@skripsirudi.my.id  
**Subject:** laporan keuangan  
**Attachment:** Choose File No file chosen  
Attach another file  
Advanced Settings  
**Content-Type:**  text/plain  text/html  Editor  
**Text:** segera kirimkan saya laporan keuangan perusahaan di jangka waktu lima tahun terakhir

Gambar 4. 29 mengirim spoofing ke skripsirudi.my.id

*Email spoofing* tersebut diatas berhasil terkirim ke alamat *email* naufalhanif@skripsian.online terlihat seperti pada gambar 4.30 berikut.



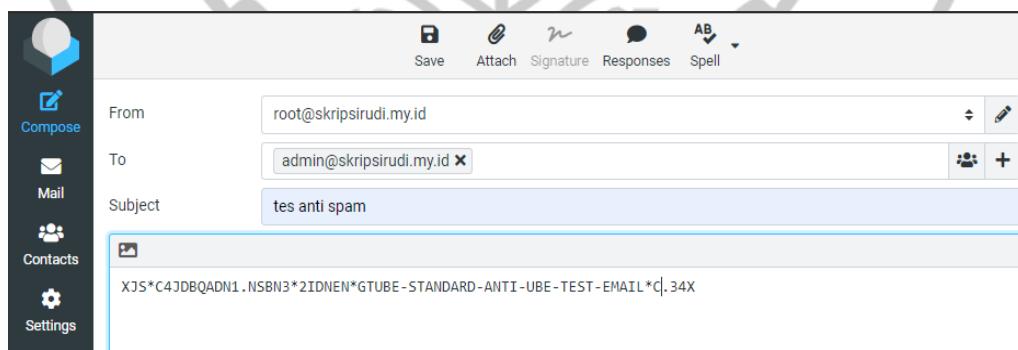
Gambar 4. 30 *Email Spoofing* Terkirim ke User skripsirudi.my.id

#### 4.3.6 Uji Coba Mengirim *Email Spam*

Uji coba mengirim *email spam* dilakukan dengan mengirim *email spam* menggunakan layanan *email* skripsirudi.my.id, *Yahoo! Mail*, dan *Gmail* ke layanan *email* skripsirudi.my.id.

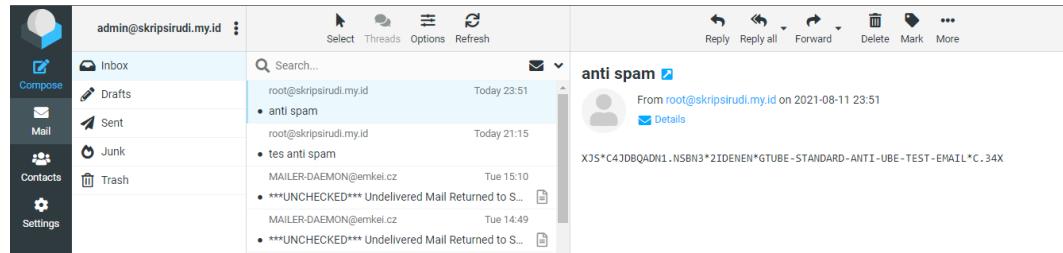
#### 4.3.7 Mengirim *Email Spam* dari skripsirudi.my.id

Uji coba kedua adalah dengan mengirim *email spam* menggunakan layanan *email* skripsian.online ke layanan *email* skripsian.online, isi pesan yang digunakan adalah XJS\*C4JDBQADN1.NSBN3\*2IDNEN\*GTUBE-STANDARD-ANTI-UBE-TEST-EMAIL\*C.34X yang merupakan standar GTUBE untuk menguji kinerja *anti spam* terlihat seperti gambar 4.31 berikut. (Klop and Csuka 2018)



Gambar 4. 31 Mengirim *Email Spam* dari skripsirudi.my.id

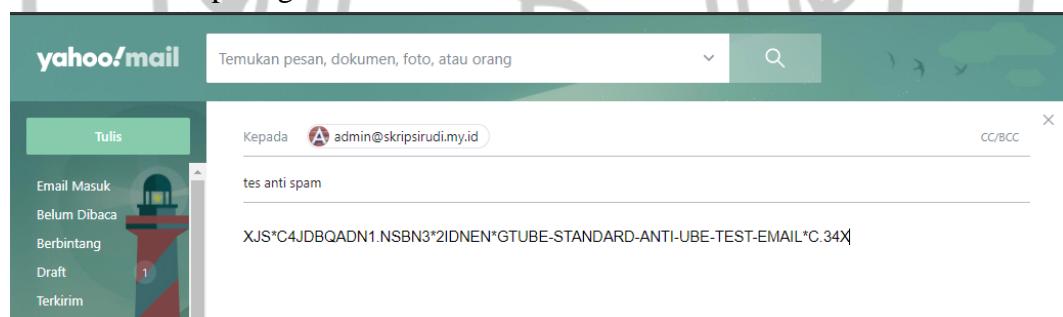
*Email spam* tersebut diatas berhasil terkirim ke alamat *email* admin@skripsirudi.my.id karena belum ada penerapan *anti spam* pada *mail server* skripsirudi.my.id terlihat seperti pada gambar 4.32 berikut.



Gambar 4. 32 email spam dari skripsirudi.my.id terkirim

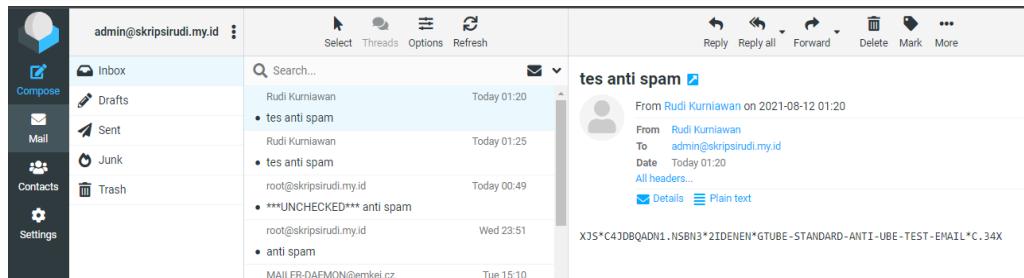
#### 4.3.8 Mengirim *Email Spam* dari *Yahoo! Mail*

Dengan menggunakan skenario yang sama seperti pada uji coba mengirim *email spam* dari layanan *email* skripsiian.online ke layanan *email* skripsiian.online, namun pada uji coba ini akan di uji pengiriman *email spam* ke layanan *email* skripsiian.online dari layanan *email* *Yahoo! Mail* terlihat seperti gambar 4.33 berikut.



Gambar 4. 33 Mengirim *Email Spam* dari *Yahoo! Mail*

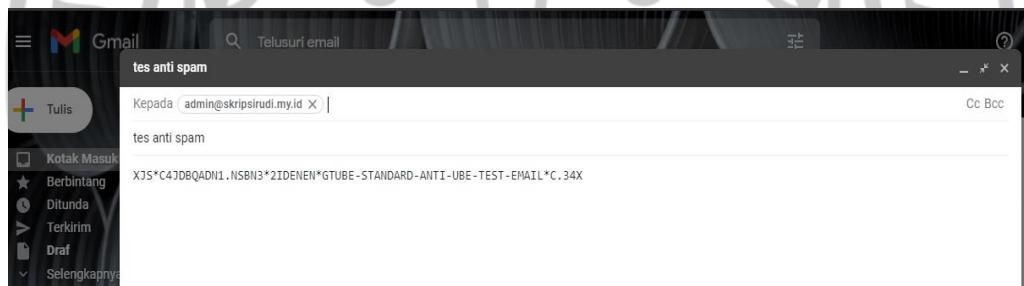
*Email spam* tersebut diatas berhasil terkirim ke alamat *email* admin@skripsirudi.my.id karena belum ada penerapan *anti spam* pada *mail server* skripsirudi.my.id terlihat seperti pada gambar 4.34 berikut.



**Gambar 4. 34 Email Spam dari Yahoo! Mail Terkirim**

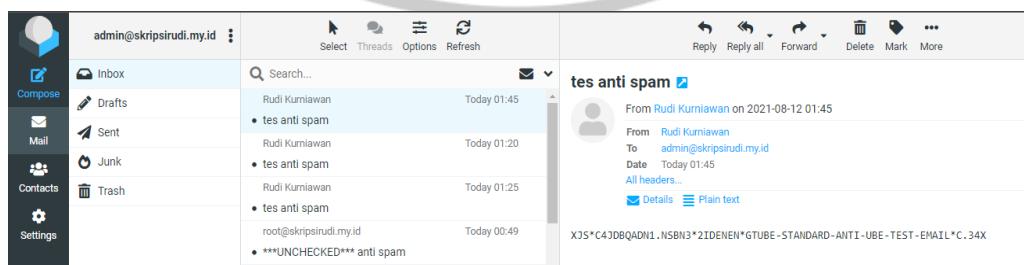
#### 4.3.9 Uji Coba Mengirim *Email Spam* dari *Gmail*

Dengan menggunakan skenario yang sama seperti pada uji coba mengirim *email spam* dari layanan *email* skripsirudi.my.id ke layanan *email* skripsirudi.my.id, namun pada uji coba ini akan di uji pengiriman *email spam* ke layanan *email* skripsirudi.my.id dari layanan *email Gmail* terlihat seperti gambar 4.35 berikut.



**Gambar 4. 35 Mengirim *Email Spam* dari *Gmail***

*Email spam* tersebut diatas berhasil terkirim ke alamat *email* admin@skripsirudi.my.id karena belum ada penerapan *anti spam* pada *mail server* skripsirudi.my.id terlihat seperti pada gambar 4.36 berikut.



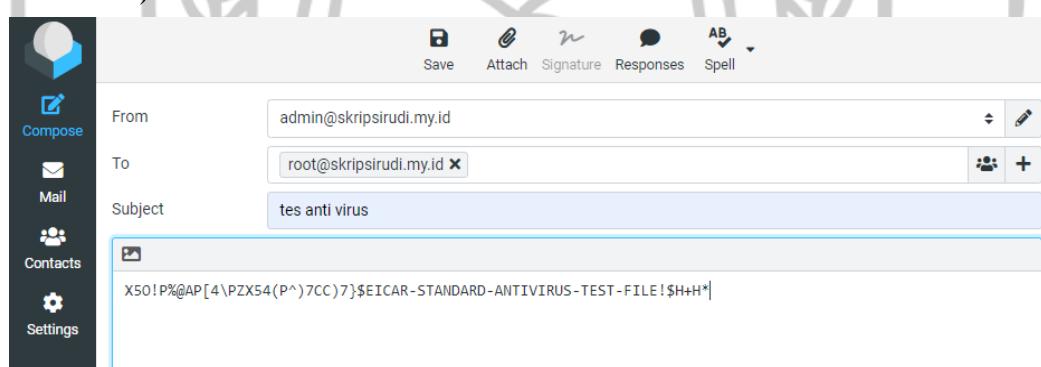
**Gambar 4. 36 Email Spam dari *Gmail* Terkirim**

#### **4.3.10 Uji Coba Mengirim Email yang Mengandung Virus**

Uji coba pengiriman *email* yang mengandung *virus* dilakukan dengan mengirim *email* yang mengandung *virus* dari layanan *email* skripsi.rudi.my.id, *Yahoo! Mail*, dan *Gmail* ke layanan *email* skripsi.rudi.my.id.

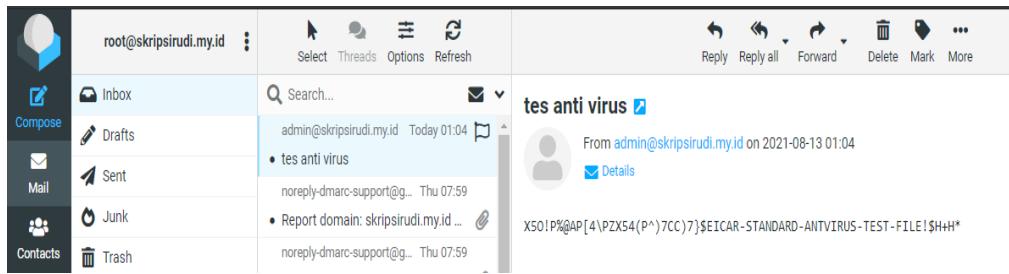
#### **4.3.11 Uji Coba Mengirim Email yang Mengandung Virus dari skripsi.rudi.my.id**

Uji coba pengiriman *email* yang mengandung *virus* dilakukan dengan mengirim *email* yang berisi X5O!P% @AP[4\ZX54(P^)7CC]7 }\$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!\$H+H\* yang merupakan standar EICAR untuk melakukan tes anti virus mail server, email dikirim dari layanan email skripsi.rudi.my.id ke layanan email skripsi.rudi.my.id, seperti pada gambar 4.37 berikut. (Abrams 1999)



**Gambar 4. 37 EICAR Test dari skripsi.rudi.my.id**

Setelah *email* tersebut dikirim pada salah satu *user email* yang ada pada *mail server* skripsi.rudi.my.id maka *email* yang mengandung *virus* tersebut berhasil terkirim ke *user* yang berada pada *mail server* skripsi.rudi.my.id seperti terlihat pada gambar 4.38 berikut



Gambar 4. 38 Email Mengandung Virus dari skripsirudi.my.id Terkirim

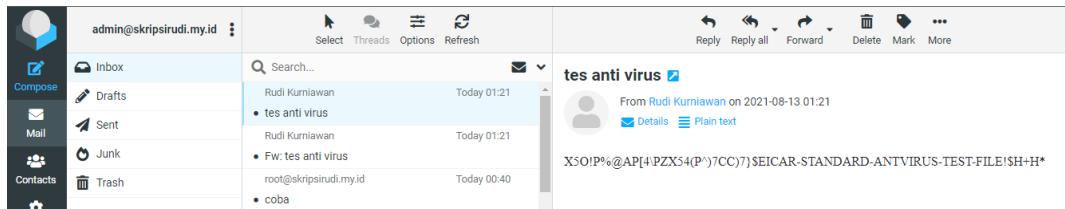
#### 4.3.12 Uji Coba Mengirim Email yang Mengandung Virus dari Yahoo! Mail

Uji coba pengiriman *email* yang mengandung *virus* dilakukan dengan mengirim *email* dengan isi X5O!P%@AP[4\PZX54(P^)7CC7}\\$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!\\$H+H\* yang merupakan standar EICAR untuk melakukan tes anti virus mail server, email dikirim dari layanan email Yahoo! Mail ke layanan email skripsirudi.my.id, seperti pada gambar 4.39 berikut. (Abrams 1999)



Gambar 4. 39 EICAR Test dari Yahoo! Mail

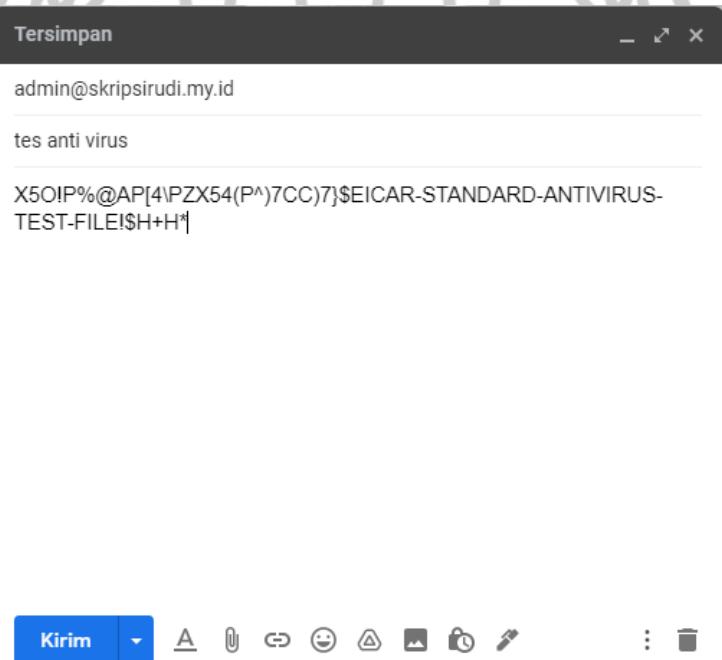
Setelah *email* tersebut dikirim pada salah satu *user email* yang ada pada *mail server* skripsiian.online, maka *email* yang mengandung *virus* tersebut berhasil terkirim ke *user email* yang berada pada *mail server* skripsiian.online seperti terlihat pada gambar 4.40 berikut.



Gambar 4. 40 Email Mengandung Virus dari Yahoo! Mail Terkirim

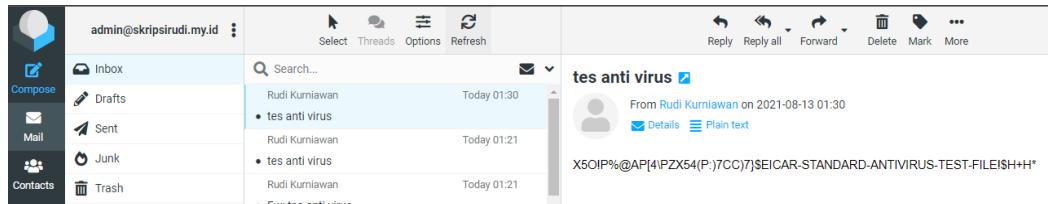
#### 4.3.13 Uji Coba Mengirim Email yang Mengandung Virus dari Gmail

Uji coba pengiriman *email* yang mengandung *virus* dilakukan dengan mengirim *email* dengan isi X5O!P% @AP[4\|PZX54(P^)7CC)7}\\$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!\$H+H\* yang merupakan standar EICAR untuk melakukan tes anti virus mail server, email dikirim dari layanan email Gmail ke layanan email skripsi.rudi.my.id, seperti pada gambar 4.41 berikut.(Abrams 1999)



Gambar 4. 41 EICAR Test dari Gmail

Setelah *email* tersebut dikirim pada salah satu pengguna *email* yang ada pada *mail server* skripsirudi.my.id maka *email* yang mengandung *virus* tersebut berhasil terkirim ke penerima yang berada pada *mail server* skripsirudi.my.id seperti terlihat pada gambar 4.42 berikut.



Gambar 4.42 Email Mengandung Virus dari Gmail Terkirim

#### 4.3.14 Uji Coba Pengecekan *Header Email*

Uji coba pengecekan *header email* dilakukan dengan membandingkan *header email* yang dikirim dari layanan *email* skripsirudi.my.id ke layanan *email* *Gmail*, *Yahoo! Mail*, dan skripsirudi.my.id sebelum dan setelah penerapan *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus*.

#### 4.3.15 *Header Email* pada *Gmail*

Uji coba ini dilakukan dengan mengirim *email* dari salah satu *user email* yang ada pada skripsirudi.my.id ke salah satu *user email* yang ada pada *Gmail* kemudian melakukan pengecekan *header email* tersebut dan melakukan perbandingan terhadap *header email* sebelum dan setelah penerapan *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus*, *header email* sebelum diterapkannya *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus* terlihat seperti gambar 4.43 berikut.

```

ARC-Authentication-Results: i=1; mx.google.com;
dkim=temperror (no key for signature) header.i=@skripsirudi.my.id header.s=default header.b=X5TPJCmt;
dkim=temperror (no key for signature) header.i=@skripsirudi.my.id header.s=default header.b=iqCe0fxw;
spf=neutral (google.com: 103.41.207.240 is neither permitted nor denied by best guess record for domain of
admin@skripsirudi.my.id) smtp.mailfrom=admin@skripsirudi.my.id
Return-Path: <admin@skripsirudi.my.id>
Received: from srv1.skripsirudi.my.id ([103.41.207.240])
by mx.google.com with ESMTPS id l17si293887pjz.83.2021.08.12.10.40.21
for <rudi.masterqq3@gmail.com>
(version=TLS1_2 cipher=ECDHE-ECDSA-AES128-GCM-SHA256 bits=128/128);
Thu, 12 Aug 2021 10:40:21 -0700 (PDT)
Received-SPF: neutral (google.com: 103.41.207.240 is neither permitted nor denied by best guess record for domain of
admin@skripsirudi.my.id) Client-ip=103.41.207.240;
Authentication-Results: mx.google.com;
dkim=temperror (no key for signature) header.i=@skripsirudi.my.id header.s=default header.b=X5TPJCmt;
dkim=temperror (no key for signature) header.i=@skripsirudi.my.id header.s=default header.b=iqCe0fxw;
spf=neutral (google.com: 103.41.207.240 is neither permitted nor denied by best guess record for domain of
admin@skripsirudi.my.id) smtp.mailfrom=admin@skripsirudi.my.id

```

**Gambar 4. 43 Cuplikan Header Email pada Gmail Sebelum Penerapan**

Pada gambar 4.43 terlihat pada cuplikan *header email* hanya terdapat parameter *DKIM =temperror* dan Received-SPF: neutral belum terdapat parameter *DMARC* atau tanda tangan *digital* dan *X-Virus-Scanned* karena belum ada penerapan *DMARC*, *DKIM*, *ClamAV*, dan *Amavisd-New*.

#### 4.3.16 Header Email pada Yahoo! Mail

Uji coba ini dilakukan dengan mengirim *email* menggunakan salah satu *user email* yang ada pada skripsi.rudi.my.id ke salah satu *user* yang ada pada *Yahoo! Mail* kemudian melakukan pengecekan *header email* tersebut dan melakukan perbandingan terhadap *header email* sebelum dan setelah penerapan *DMAR*, *DKIM*, *SPF*, *anti spam*, dan *anti virus*, *header email* sebelum diterapkannya *DMAR*, *DKIM*, *SPF*, *anti spam*, dan *anti virus* terlihat seperti gambar 4.44 berikut.

```

Received: from 10.222.142.149
by atlas301.free.mail.ne1.yahoo.com with HTTPS; Thu, 12 Aug 2021 18:17:56 +0000
Return-Path: <root@skripsi.rudi.my.id>
X-Originating-Ip: [103.41.207.240]
Received-SPF: none (domain of skripsi.rudi.my.id does not designate permitted sender hosts)
Authentication-Results: atlas301.free.mail.ne1.yahoo.com;
dkim=perm_fail header.i=@skripsi.rudi.my.id header.s=default;
dkim=perm_fail header.i=@skripsi.rudi.my.id header.s=default;
spf=none smtp.mailfrom=skripsi.rudi.my.id;
dmarc=unknown header.from=skripsi.rudi.my.id;
X-Apparently-To: rudi.masterqq3@yahoo.com; Thu, 12 Aug 2021 18:17:56 +0000
X-YMailISG: FX_DJPKwLDT.jRXPrZ4qA1_5vuZthhvze7JL0gNoiylFdXU0

```

**Gambar 4. 44 Cuplikan Header Email pada Yahoo! Mail Sebelum Penerapan**

Pada gambar 4.44 dapat dilihat cuplikan *header email* belum terdapat parameter *X-Virus-Scanned* karena belum ada penerapan *ClamAV* dan *Amavisd-New*, parameter *Received-SPF* bernilai *none* karena belum ada penerapan *SPF*, dan parameter *dkim=perm\_fail* dan *dmarc=unknown* yang berarti belum ada tanda tangan *digital* karena belum diterapkan *DMARC* dan *DKIM*.

#### 4.3.17 Header Email pada skripsirudi.my.id

Uji coba ini dilakukan dengan mengirim *email* menggunakan salah satu *user email* yang ada pada skripsirudi.my.id ke salah satu *user email* yang ada pada skripsirudi.my.id kemudian melakukan pengecekan *header email* dan melakukan perbandingan terhadap *header email* sebelum dan setelah penerapan *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus*, *header email* sebelum diterapkannya *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus* terlihat seperti gambar 4.45 berikut.

**Authentication-Results:** srv1.skripsirudi.my.id (amavisd-new); dkim=neutral  
reason="invalid (public key: not available)"  
header.d=skripsirudi.my.id

**Received:** from srv1.skripsirudi.my.id ([127.0.0.1])  
by localhost (srv1.skripsirudi.my.id [127.0.0.1]) (amavisd-new, port 10024)  
with ESMTP id CvWI7sq-udlU for <admin@skripsirudi.my.id>;  
Fri, 13 Aug 2021 03:51:52 +0800 (WITA)

**Received:** from localhost (skripsi.rudi.com [IPv6:::1])  
by srv1.skripsirudi.my.id (Postfix) with ESMTPA id 132911163CE  
for <admin@skripsirudi.my.id>; Fri, 13 Aug 2021 03:51:52 +0800 (WITA)

**DKIM-Signature:** v=1; a=rsa-sha256; c=relaxed/simple; d=skripsirudi.my.id;  
s=default; t=1628797912;  
bh=0nLVVXLwOLbC9inyoGwFCTTHs5+Kc/F/uvToZ4sZEsQ=;  
h=Date:From:To:Subject;  
b=sDAnWGLh3cyodXAmiHe56GDwol/cOd7o8GvNIECe0MYdwh2JuUPKH94bfnP9Nor  
sFWclft69UHTxZ//TXUPCvHiUDzKg7tOCEzFmAQHQoHYujwYAEpW2fl2U4YPa4ili9  
YgzyXfFa16ejvb+VaMCn1QEpjMkZy2t3tUfbhY=

**MIME-Version:** 1.0

**Date:** Fri, 13 Aug 2021 03:51:51 +0800

**From:** root@skripsirudi.my.id

**To:** admin@skripsirudi.my.id

**Subject:** coba header

**User-Agent:** Roundcube Webmail/1.4.11

**Message-ID:** <8f3d93e611d6c61a6ae42b44dd3e862c@skripsirudi.my.id>

Gambar 4. 45 Cuplikan *Header Email* pada skripsi an Sebelum Penerapan

Pada gambar 4.45 terlihat *header email* belum terdapat parameter *X-Virus-Scanned* dan *DKIM* masih bernilai dkim=neutral karena belum diterapkan *Amavisd-New* dan protocol *DMARC* dan *DKIM*.

#### **4.3.18 Setelah Diterapkan Filtering, Otentikasi dan Otorisasi**

Uji coba yang dilakukan setelah diterapkan *filtering*, otentikasi, dan otorisasi *email* adalah uji coba mengirim *email spoofing*, uji coba mengirim *email spam*, uji coba mengirim *email* yang mengandung *virus*, dan uji coba pengecekan *header email* (Hanif 2018).

#### **4.3.19 Uji Coba Mengirim *Email Spoofing***

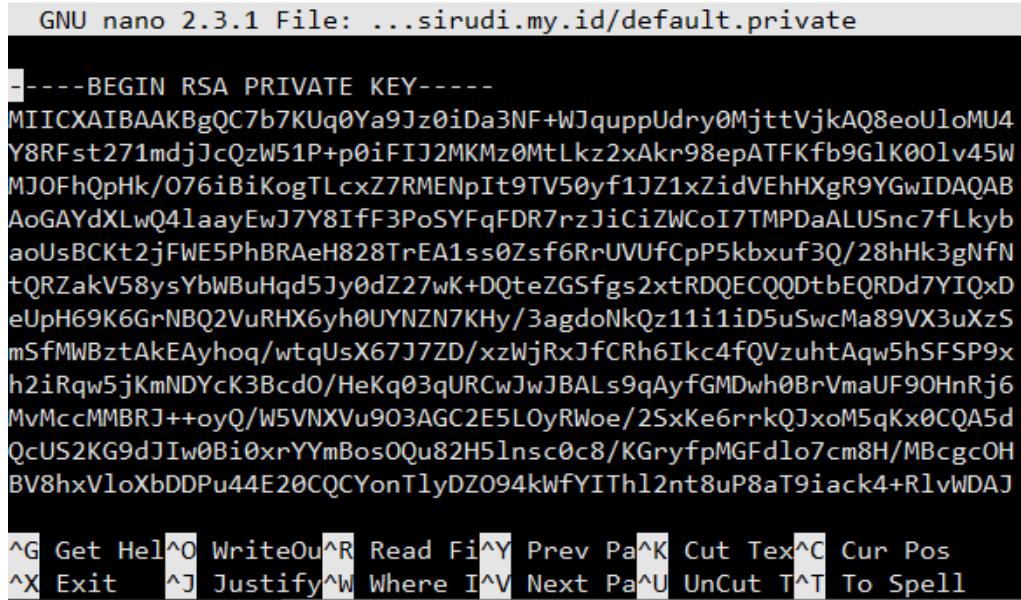
Uji coba mengirim *email spoofing* dilakukan dengan mengirim *email spoofing* dari *Emkei's Fake Mailer* ke *Gmail*, *Yahoo! Mail*, dan *skripsi.rudi.my.id* dengan mengatasnamakan salah satu *user email* yang ada pada *skripsi.rudi.my.id* (Hanif 2018).

#### **4.3.20 Uji Coba Mengirim *Email Spoofing* ke *Gmail***

(Hanif 2018) Proses uji coba mengirim *email spoofing* pada *email server* akan berbeda setelah protocol *DMARC*, *SPF* dan *DKIM* diterapkan pada *mail server* karena protokol *DMARC*, *SPF* dan *DKIM* akan melakukan otentikasi dan otorisasi pada setiap *email* yang datang dari *mail server* *skripsi.rudi.my.id*. Proses yang terjadi setelah penerapan protokol *DMARC*, *SPF* dan *DKIM* adalah sebagai berikut:

1. Pesaing tersebut melakukan pengiriman *email spoofing* menggunakan *Emkei's Fake Mailer* dengan cara membuka situs [www.emkei.cz](http://www.emkei.cz) menggunakan *browser* kemudian pada situs [www.emkei.cz](http://www.emkei.cz) pesaing tersebut menuliskan alamat pengirim *email* yaitu *admin@skripsi.rudi.my.id* dan alamat penerima *email* yaitu *rudi.masterqq3@gmail.com*.
2. Ketika *email spoofing* tersebut melewati *mail server Emkei's Fake Mailer* maka *email spoofing* tersebut tidak mendapatkan *private key*

yang hanya terdapat pada *mail server* skripsirudi.my.id terlihat seperti gambar 4.56 berikut.



```
GNU nano 2.3.1 File: ...sirudi.my.id/default.private

-----BEGIN RSA PRIVATE KEY-----
MIICXAIBAAKBgQC7b7KUq0Ya9Jz0iDa3NF+WJquppUdry0MjttVjkAQ8eoUloMU4
Y8RFst271mdjJcQzW51P+p0iFIJ2MKMz0MtLkz2xAkr98epATFKfb9G1K001v45W
MJOFhQpHk/076iBiKogTLcxZ7RMENpIt9TV50yf1JZ1xZidVEhHXgR9YGwIDAQAB
AoGAYdXLwQ4laayEwJ7Y8IfF3PoSYFqFDR7rzJiCizWCoI7TMPDaALUSnc7fLkyb
aoUsBCKt2jFWE5PhBRAeH828TrEA1ss0Zsf6RrUVUFcpP5kbxuf3Q/28hHk3gNfN
tQRZakV58ysYbWBuHqd5Jy0dZ27wK+DQteZG5fgs2xtRDQECQQDtbeQRDd7YI0xD
eUpH69K6GrNBQ2VuRHX6yh0UYNZN7KHy/3agdoNkQz11i1iD5uSwcMa89VX3uXzS
m5fMWBztAkEAyhoq/wtqUsX67J7ZD/xzWjRxJfCRh6IkC4fQVzuhtAqw5hSFSP9x
h2iRqw5jKmNDYcK3Bcd0/HeKq03qURcwJwJBALS9qAyfGMDwh0BrVmaUF90HnRj6
MvMccMMBRJ++oyQ/W5VNXXu903AGC2E5LOyRWoe/2SxKe6rrkQJxoM5qKx0CQA5d
QcUS2KG9dJIw0Bi0xrYYmBos0Qu82H5lnsc0c8/KGryfpMGFdlo7cm8H/MBcgcOH
BV8hxVloXbDDPu44E20CQCYonTlyDZ094kWfYITH12nt8uP8aT9iack4+R1vWD AJ

^G Get Help ^O Write Out ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit      ^J Justify ^W Where ^I ^V Next Page ^U Uncut ^T To Spell
```

Gambar 4.46 Private Key pada skripsirudi.my.id

3. Ketika *email spoofing* tersebut masuk ke *mail server Gmail* maka *email* tersebut akan dianggap sebagai *spam* karena *email* tersebut tidak mempunyai *private key* yang ada pada *mail server* skripsirudi.my.id yang cocok dengan *public key* yang telah diletakan pada *DNS server* skripsirudi.my.id sehingga pesan tersebut tidak memiliki tanda tangan *digital* pada *header email* (proses *DKIM*) terlihat seperti gambar 4.47 dan 4.48 berikut.

```
default._domainkey 14400 IN TXT "v=DKIM1; k=rsa;
p=MIGfMA0GCSqGSIb3DQEBAQUAA4GNADCBiQKBgQC7b7KUq0Ya9Jz0iDa3NF+WJquppUdry0MjttVjkAQ8eoUloMU4Y8RFst2
71mdjJcQzW51P+p0iFIJ2MKMz0MtLkz2xAkr98epATFKfb9G1K001v45WMJOFhQpHk/076iBiKogTLcxZ7RMENpIt9TV50yf1JZ1xZi
dVEhHXgR9YGwIDAQAB"
```

Gambar 4.47 Public Key pada DNS Server skripsirudi.my.id

Pesan Asli

ID Pesan	<ebebe14c54b7ee1cd1e1f839d4d286aa@skripsirudi.my.id>
Dibuat pada:	27 Juni 2021 14.58 (Dikirim setelah 69 detik)
Dari:	admin@skripsirudi.my.id
Kepada:	rudi.masterqq3@gmail.com
Subjek:	percobaan ke 3
SPF:	PASS dengan IP 103.41.207.240 Pelajari lebih lanjut
DKIM:	'PASS' dengan domain skripsirudi.my.id Pelajari lebih lanjut
DMARC:	'PASS' Pelajari lebih lanjut

Gambar 4.48 Cuplikan Header Email

4. Selanjutnya *mail server Gmail* akan melakukan pengecekan *SIDF* (*Sender ID Framework*) pada *record DNS server* skripsirudi.my.id, karena alamat *IP Emkei's Fake Mailer* adalah 101.99.94.155 maka *email spoofing* tersebut dianggap sebagai *spam* dikarenakan *record SPF* dan *DMARC* yang ada pada *DNS server* skripsirudi.my.id hanya mengizinkan pengiriman *email* dari alamat yang telah diotorisasi yaitu alamat *IP* 103.41.207.240 yang merupakan alamat mail server skripsirudi.my.id dan nilai dari parameter *Received-SPF* dan *DMARC* pada *header email* adalah *fail* sehingga *email* tersebut akan ditandai sebagai *email spam* oleh *server Gmail* dan *email spoofing* tersebut tidak diblok oleh *Gmail* di keranakan *record* pada *DMARC* untuk mengkarantina *email* sehingga *email* tersebut di masukan ke folder *spam* (proses *SPF* dan *DMARC*) terlihat seperti gambar 4.49 dan 4.50 berikut.

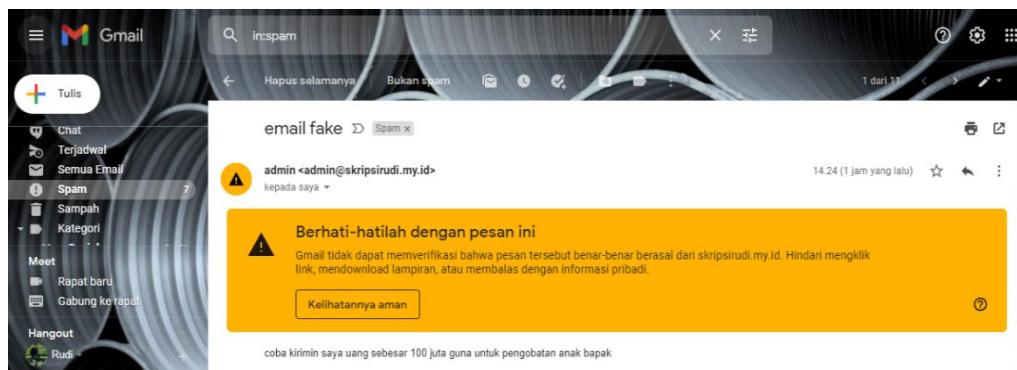
skripsirudi.my.id. IN TXT "v=spf1 mx a ip4:103.41.207.240/32 a:srv1.skripsirudi.my.id -all"

Gambar 4.49 SPF Record pada skripsirudi.my.id

\_dmarc 14400 IN TXT "v=DMARC1; p=quarantine; pct=100; rua=mailto:root@skripsirudi.my.id"

Gambar 4. 50 DMARC Record pada skripsirudi.my.id

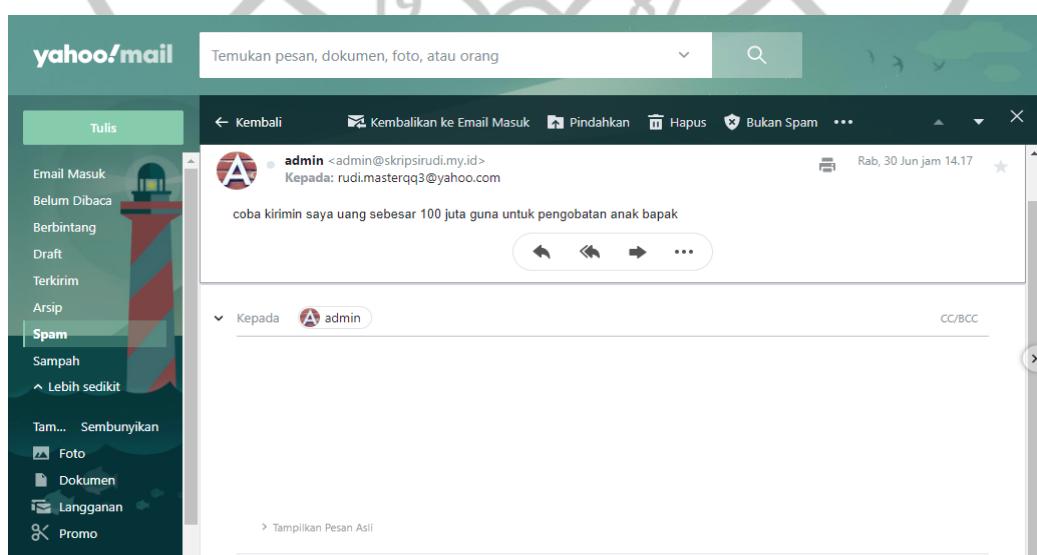
- Hasil verifikasi pengiriman email fake dari emkei.cz ke mail dengan mengatas nama kan salah satu user email yaitu admin@skripsirudi.my.id yang kemudian terindikasi sebagai spam oleh gmail terlihat seperti gambar 4.51 berikut.



Gambar 4. 51 terindikasi spam oleh gmail

#### 4.3.21 Uji Coba Mengirim *Email Spoofing* ke *Yahoo! Mail*

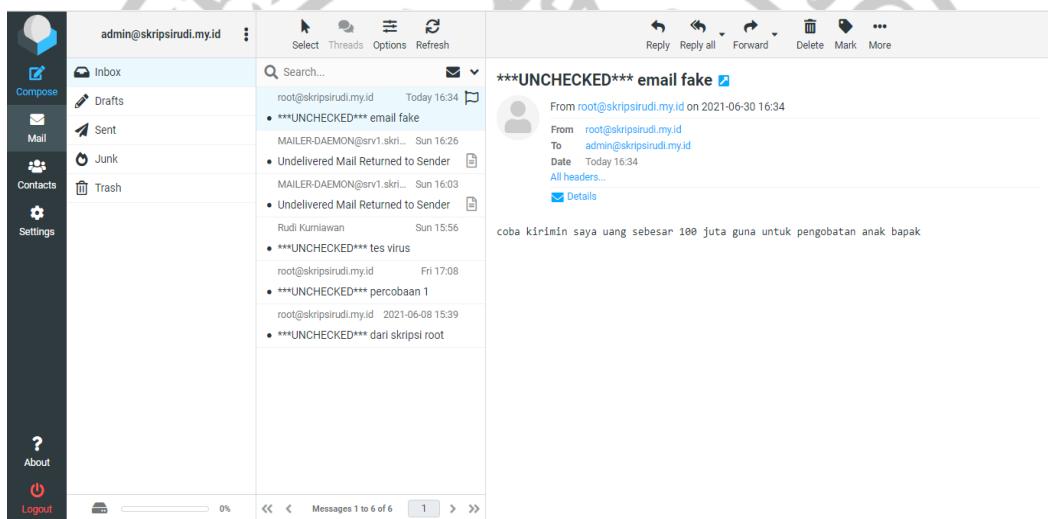
Selain mengirim *email spoofing* pada *Gmail*, pengiriman *email spoofing* juga dilakukan pada *Yahoo! Mail* untuk membandingkan perlakuan yang diberikan pada *email spoofing* antara dua layanan *email* tersebut. Pada *Gmail*, *email spoofing* yang masuk langsung di karantina sehingga *email spoofing* masuk pada *folder spam* penerima *email* sedangkan pada *Yahoo! Mail*, *email spoofing* dimasukan kedalam *folder spam* penerima *email* seperti gambar 4.52 berikut. (Hanif 2018)



Gambar 4. 52 *Email Spoofing* Masuk ke *Folder Spam*

#### 4.3.22 Coba Mengirim *Email Spoofing* ke skripsirudi.my.id

Selain mengirim *email spoofing* ke *Gmail* dan *Yahoo! Mail*, pengiriman *email spoofing* juga dilakukan pada *skripsirudi.my.id* untuk membandingkan perlakuan yang diberikan pada *email spoofing* antara dua layanan *email* tersebut. Pada *Gmail*, *email spoofing* yang masuk langsung langsung masuk pada folder spam, pada *Yahoo! Mail*, *email spoofing* dimasukan kedalam folder *spam* penerima *email*, dan pada *skripsirudi.my.id* *email spoofing* masuk pada folder *inbox*, seperti gambar 4.53 berikut. (Hanif 2018).



Gambar 4. 53 *Email Spoofing* Masuk pada Folder *Inbox*

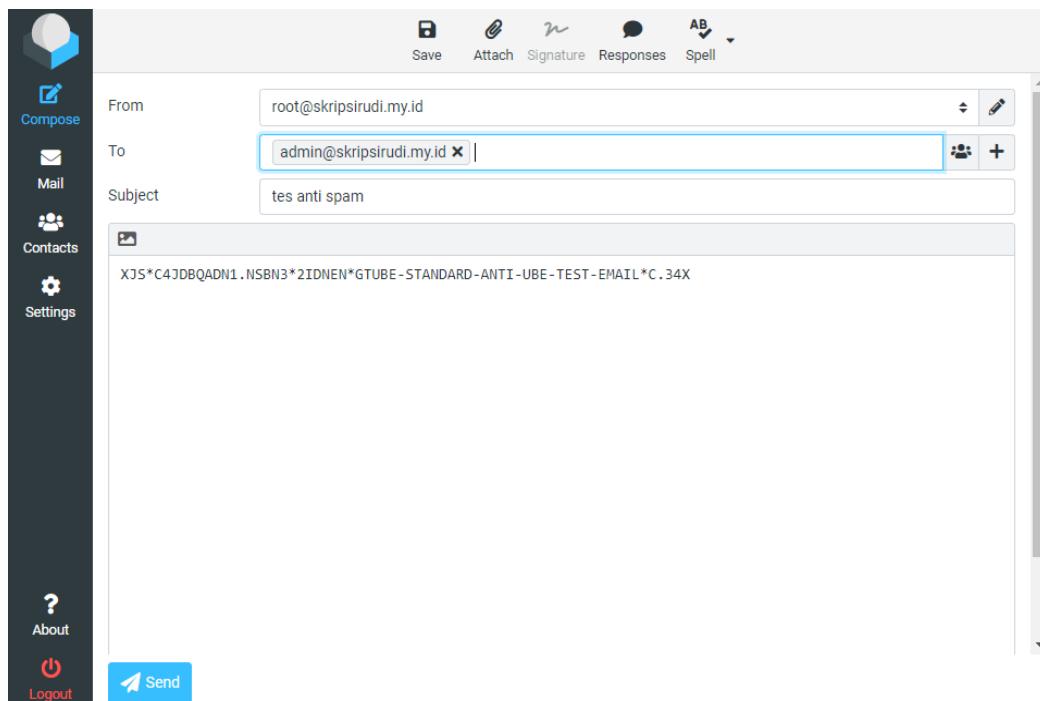
#### 4.3.23 Uji Coba Mengirim *Email Spam*

Uji coba mengirim *email spam* dilakukan dengan mengirim *email spam* menggunakan layanan *email* *skripsirudi.my.id*, *Yahoo! Mail*, dan *Gmail* ke layanan *email* *skripsirudi.my.id*.

#### 4.3.24 Uji Coba Mengirim *Email Spam* dari *skripsirudi.my.id*

Uji coba mengirim *email spam* dari layanan *email* *skripsirudi.my.id* ke layanan *email* *skripsirudi.my.id* adalah dengan mengirim *email spam* menggunakan layanan *email* *skripsirudi.my.id* ke layanan *email* *skripsirudi.my.id*, isi pesan yang digunakan adalah

XJS\*C4JDBQADN1.NSBN3\*2IDNEN\*GTUBE-STANDARD-ANTI-UBE-TEST-EMAIL\*C.34X yang merupakan standar *GTUBE* untuk menguji kinerja *anti spam* terlihat seperti gambar 4.54 berikut. (Hanif 2018). (Klop and Csuka 2018)



**Gambar 4. 54 Mengirim Email Spam dari skripsirudi.my.id Setelah Penerapan**

*Email spam* tersebut diatas diblokir oleh *Amavisd-New* karena terindikasi sebagai *email spam* oleh *SpamAssassin*, hasil pemfilteran *email spam* dapat dilihat pada *mail log* dengan menggunakan perintah #cat /var/log/maillog seperti terlihat pada gambar 4.55 berikut. (Hanif 2018).

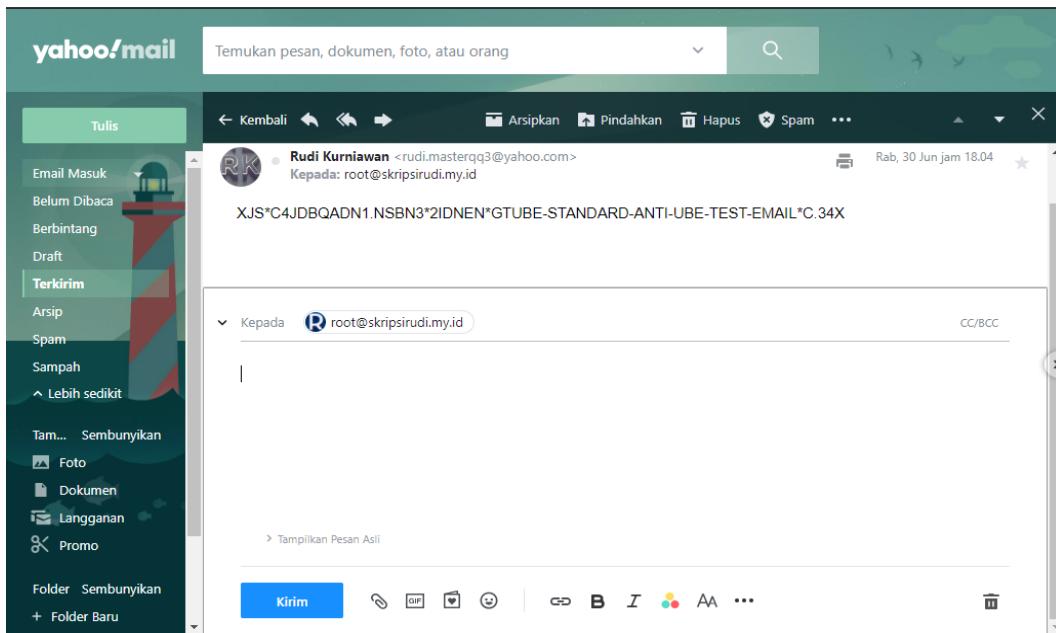
```
Jun 30 17:12:12 srv1 amavis[11541]: (11541-05) Blocked [PAM {DiscardedInternal,Quarantined}], MYNETS LOCAL [:1]:38244 <root@skripsirudi.my$  
Jun 30 17:12:12 srv1 postfix/smtp[17750]: 2DA85C0779: to=<admin@skripsirudi.my.id>, relay=127.0.0.1[127.0.0.1]:10024, delay=10, delays=0.1$  
Jun 30 17:12:12 srv1 postfix/qmgr[1191]: 2DA85C0779: removed
```

**Gambar 4. 55 Email dari skripsirudi.my.id Terindikasi Spam**

#### 4.3.25 Uji Coba Mengirim Email Spam dari Yahoo mail

Uji coba mengirim *email spam* dari layanan *email Yahoo! Mail* ke skripsirudi.my.id adalah dengan mengirim *email spam* menggunakan layanan *email Yahoo! Mail* ke layanan *email* skripsirudi.my.id, isi pesan

yang digunakan adalah XJS\*C4JDBQADN1.NSBN3\*2IDNEN\*GTUBE-STANDARD-ANTI-UBE-TEST-EMAIL\*C.34X yang merupakan standar GTUBE untuk menguji kinerja *anti spam* terlihat seperti gambar 4.56 berikut. (Klop and Csuka 2018)



Gambar 4. 56 Mengirim *Email Spam* dari Yahoo! Mail Setelah Penerapan

*Email spam* tersebut diatas diblokir oleh *Amavisd-New* karena terindikasi sebagai *email spam* oleh *SpamAssassin*, hasil pemfilteran *email spam* dapat dilihat pada *mail log* dengan menggunakan perintah #cat /var/log/maillog seperti terlihat pada gambar 4.57 berikut.

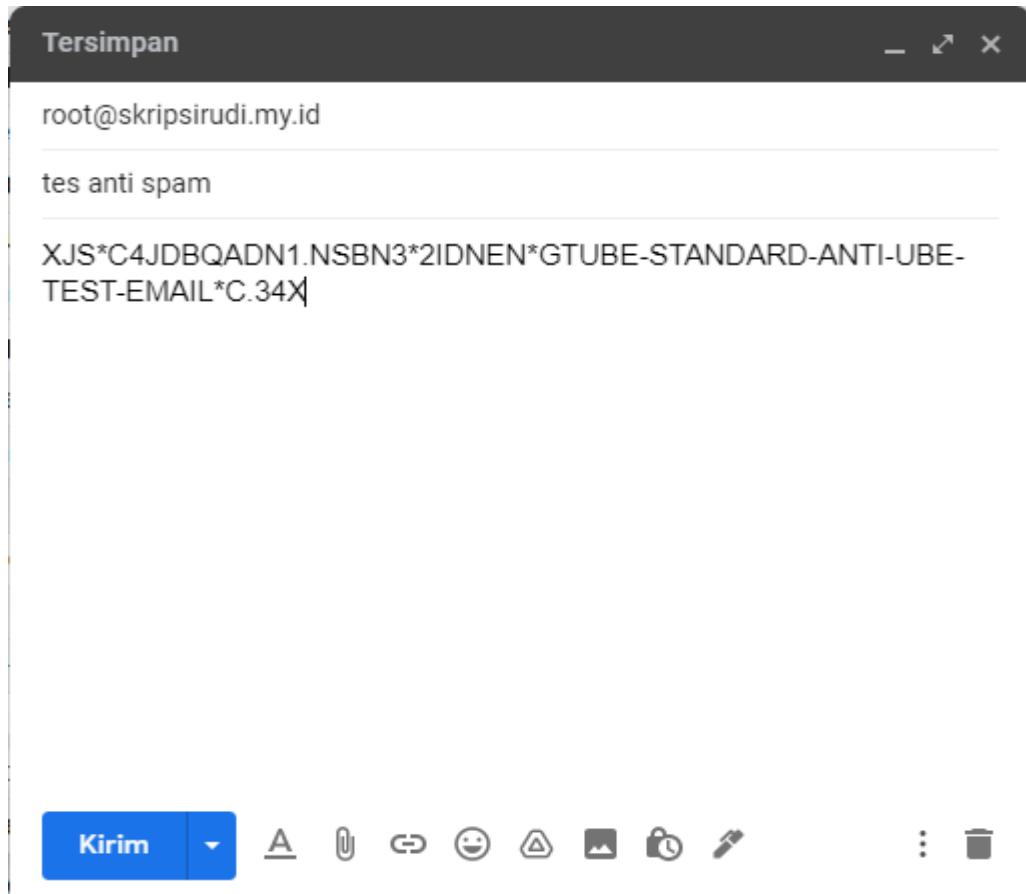
```
Jun 30 18:04:31 srv1 amavis[11610]: (11610-06) [Blocked SPAM {DiscardedInbound,Quarantined}, [106.10.241.209]:43256 [106.10.241.209] <rudi.masterqq3$ Jun 30 18:04:31 srv1 postfix/smtp[24517]: 2EBFD1027E4: to=<root@skripsirudi.my.id>, relay=127.0.0.1[127.0.0.1]:10024, delay=20, delays=0.22/0.02/0.0 Jun 30 18:04:31 srv1 postfix/qmgr[1191]: 2EBFD1027E4: removed
```

Gambar 4. 57 *Email* dari Yahoo! Mail Terindikasi *Spam*

#### 4.3.26 Uji Coba Mengirim *Email Spam* dari Gmail

Uji coba mengirim *email spam* dari layanan *email Gmail* ke layanan *email* skripsirudi.my.id adalah dengan mengirim *email spam* menggunakan layanan *email Gmail* ke layanan *email* skripsirudi.my.id, isi pesan yang digunakan adalah XJS\*C4JDBQADN1.NSBN3\*2IDNEN\*GTUBE-STANDARD-ANTI-UBE-TEST-EMAIL\*C.34X yang merupakan standar

*GTUBE* untuk menguji kinerja *anti spam* terlihat seperti gambar 4.58 berikut.  
(Klop and Csuka 2018)



Gambar 4. 58 Mengirim Email *Spam* dari *Gmail* Setalah Penerapan

*Email spam* tersebut diatas diblokir oleh *Amavisd-New* karena terindikasi sebagai *email spam* oleh *SpamAssassin*, hasil pemfilteran *email spam* dapat dilihat pada *mail log* dengan menggunakan perintah `#cat /var/log/maillog` seperti terlihat pada gambar 4.59 berikut.

```
Jun 30 18:29:39 srv1 amavis[11541]: (11541-07) Blocked SPAM [DiscardedInbound,Quarantined], [209.85.166.43]:38693 [209.85.166.43] <crudi.masterqq3@gmail.com>  
Jun 30 18:29:39 srv1 postfix/smtp[24931]: A66E21027E4: to=<root@skripsirudi.my.id>, relay=127.0.0.1[127.0.0.1]:10024, delay=41, delays=5.2/0.02/0.02$  
Jun 30 18:29:39 srv1 postfix/qmgr[1191]: A66E21027E4: removed
```

Gambar 4. 59 Email dari *Gmail* Terindikasi *Spam*

#### 4.3.27 Uji Coba Mengirim Email *Spam* Tanpa *GTUBE* Test

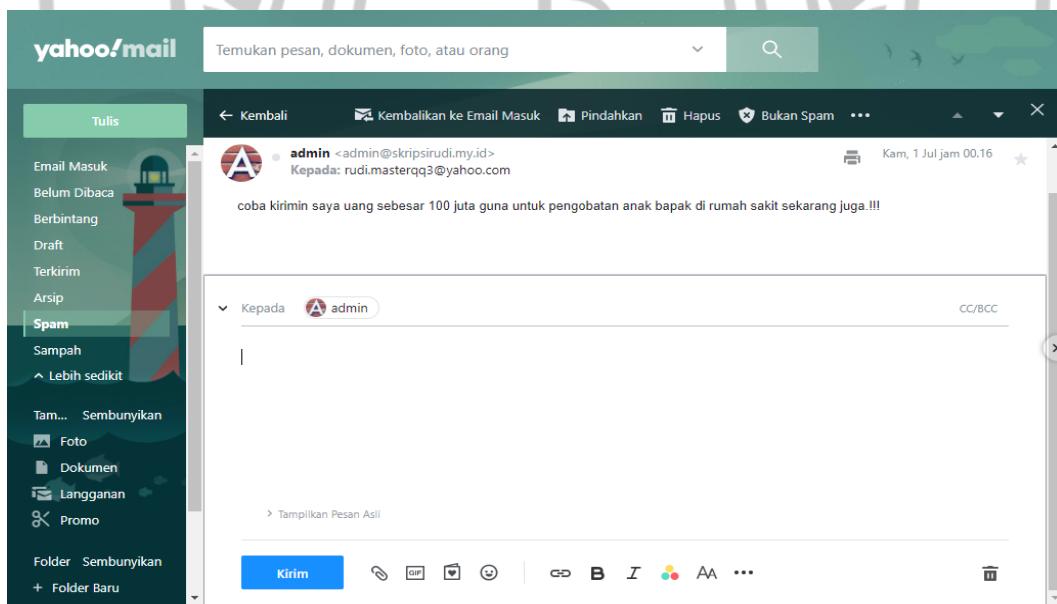
Uji coba mengirim *email* yang terindikasi *spam* oleh *Yahoo! Mail* adalah dengan mengirim *email* melalui *Emkei's Fake Mailer* dengan format

*email spam* yang berisi penipuan atau promosi suatu produk seperti terlihat pada gambar 4.60 berikut.



Gambar 4. 60 Email Dengan Format Spam

Email dengan format *spam* tersebut terkirim ke *Yahoo! Mail* dan masuk ke dalam *folder spam* seperti terlihat pada gambar 4.61 berikut.



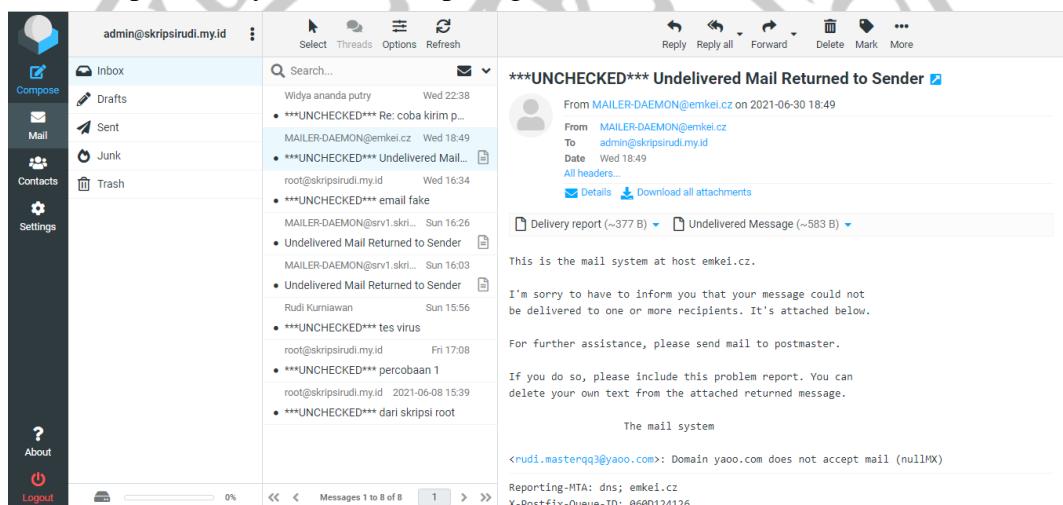
Gambar 4. 61 Email Terindikasi Sebagai *Spam* oleh *Yahoo! Mail*

*Yahoo! Mail* mengindikasi bahwa *email* diatas merupakan *spam* sehingga protocol *dmarc* mengkarantina email tersebut sehingga pesan terindikasi sebagai email *spam* terlihat pada header email bagian *dmarc=fail(p=QUARANTINE)* seperti pada gambar 4.62 berikut.

```
Authentication-Results: atlas308.free.mail.gq1.yahoo.com;
dkim=unknown;
spf=fail smtp.mailfrom=skripsirudi.my.id;
dmarc=fail(p=QUARANTINE) header.from=skripsirudi.my.id;
```

Gambar 4. 62 header *dmarc*

Pada email *skripsirudi.my.id* terlihat protocol *DMARC* yang melaporkan pesan email sebelumnya yang mengandung spoofing yang dikirim melalui emal *emkei.cz* telah di verifikasi bahwa pesan tersebut mengandung spoofing kemudian pesan tersebut di kirim kembali ke email *skripsirudi.my.id*, terlihat seperti gambar 4.63 berikut.



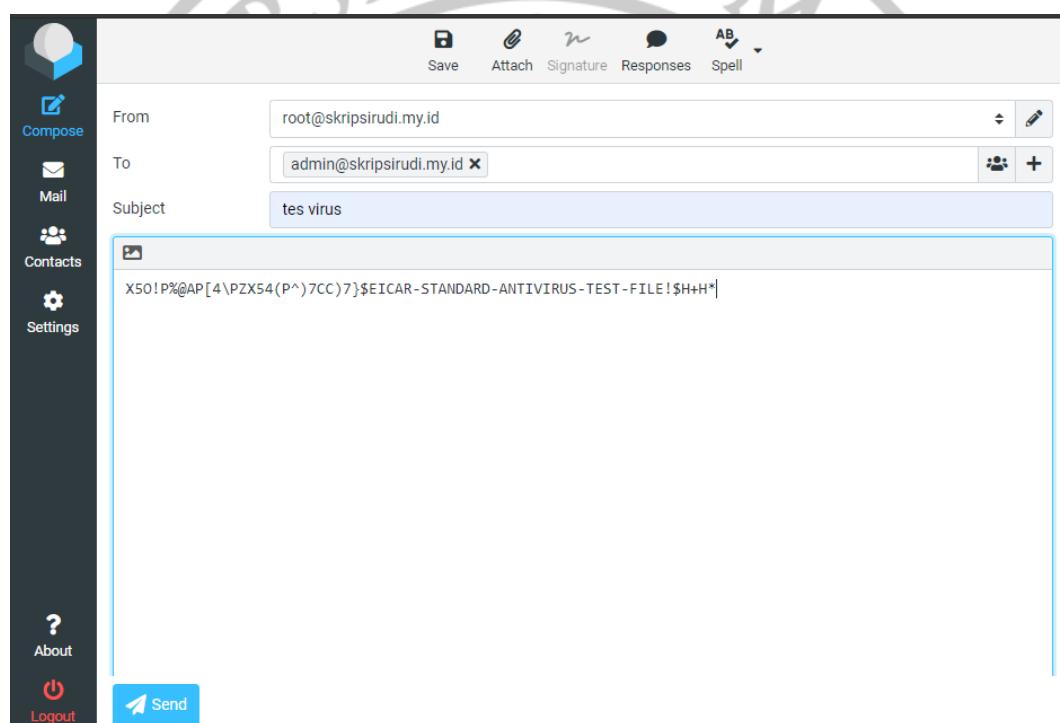
Gambar 4. 63 hasil report *DMARC* setelah protocol di terapkan

#### 4.3.28 Uji Coba Mengirim *Email* yang Mengandung *Virus*

Uji coba mengirim *email* yang mengandung *virus* dilakukan dengan mengirim *email* yang mengandung *virus* dari layanan *email* *skripsirudi.my.id*, *Yahoo! Mail*, dan *Gmail* ke layanan *email* *skripsirudi.my.id*.

#### 4.3.29 Uji Coba Mengirim Email yang Mengandung Virus dari skripsirudi.my.id

Uji coba pengiriman *email* yang mengandung *virus* dilakukan dengan mengirim *email* dengan isi X5O!P% @AP[4\PZX54(P^)7CC7}\$\_EICAR-STANDARD-ANTIVIRUS-TEST-FILE!\$H+H\* yang merupakan standar *EICAR* untuk melakukan tes *anti virus mail server*, *email* yang mengandung virus dikirim dari layanan email skripsirudi.my.id ke layanan email skripsirudi.my.id, seperti pada gambar 4.64 Berikut. (Abrams 1999)



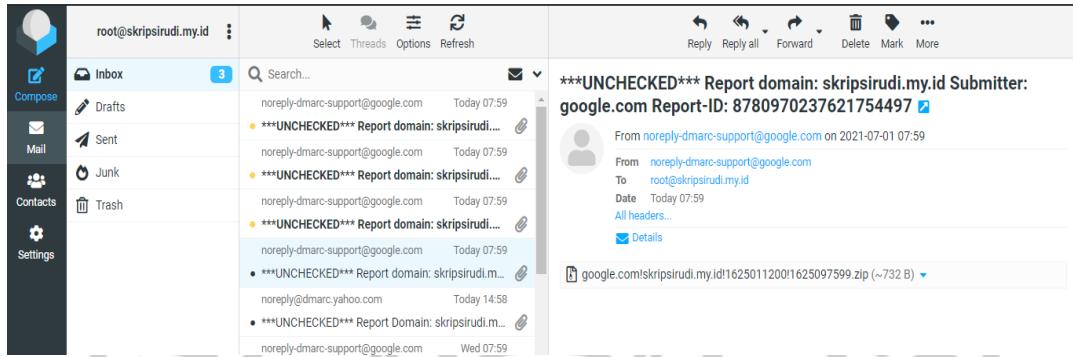
Gambar 4. 64 *EICAR Test* dari skripsirudi.my.id Setelah Penerapan

Setelah *email* yang mengandung *virus* tersebut dikirim ke salah satu *user email* yang ada pada *mail servrer* skripsirudi.my.id maka *email* tersebut akan di blok oleh *Amavis-New* dan di deteksi sebagai virus oleh *ClamAV*, namun terdapat report *DMARC* yang memberitahukan bahwa *email* tersebut mengandung virus

sehingga pesan yang masuk tidak dapat dibaca, hanya report pesan, seperti terlihat pada gambar 4.65 dan gambar 4.66 berikut.

```
Aug  8 00:24:53 skripsirudi amavis[1716]: (01716-03) Blocked INFECTED (Eicar-Signature) [DiscardedInternal,Quarantined], MYN  
'ETS LOCAL [:1]:43546 <root@skripsirudi.my.id> -> <admin@skripsirudi.my.id>, Queue-ID: 9075E115479, Message-ID: <91a096728f2  
a37527487761424d57e88@skripsirudi.my.id>, mail_id: LzMm6d0yGUJ4, Hits: -, size: 998, dkim_sd=default:skripsirudi.my.id, 177  
lms  
Aug  8 00:24:53 skripsirudi postfix/smtp[22700]: 9075E115479: to=<admin@skripsirudi.my.id>, relay=127.0.0.1[127.0.0.1]:10024  
, delay=0.35, delays=0.11/0.05/0.02/0.17, dsn=2.7.0, status=sent (250 2.7.0 Ok, discarded, id=01716-03 - INFECTED: Eicar-Sig  
nature)
```

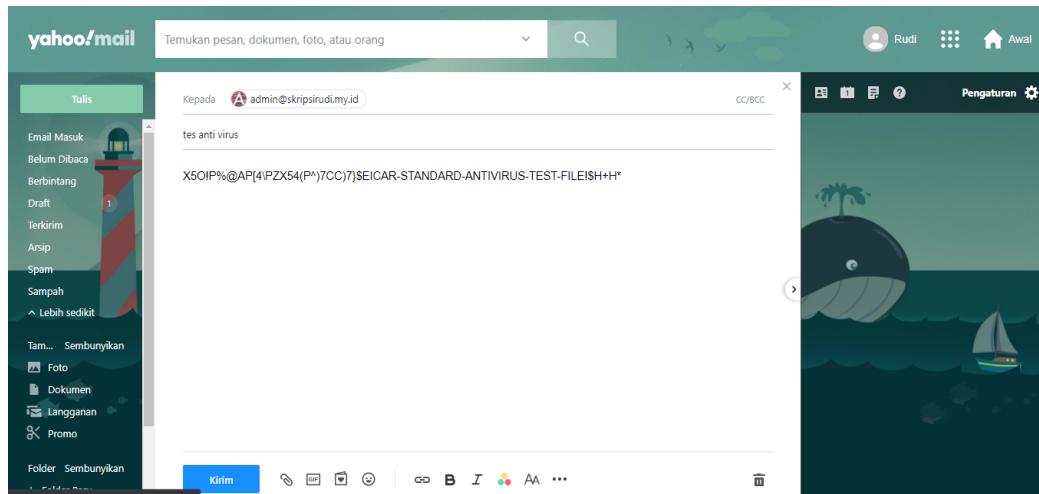
Gambar 4. 65 email local dari skripsirudi.my.id terblok



Gambar 4. 66 report email yang mengandung virus

#### 4.3.30 Uji Coba Mengirim Email yang Mengandung Virus dari Yahoo! Mail

Uji coba mengirim *email* yang mengandung *virus* dilakukan dengan mengirim *email* dengan isi X5O!P% @AP[4\PZX54(P^)7CC)7}\$\$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!\$H+H\* yang merupakan standar *EICAR* untuk melakukan tes anti *virus mail server*, *email* yang mengandung virus dikirim dari layanan *email* Yahoo! Mail ke layanan *email* skripsirudi.my.id, seperti pada gambar 4.67 berikut.(Abrams 1999)



Gambar 4. 67 EICAR Test dari Yahoo! Mail Setelah Penerapan

Setelah *email* yang mengandung *virus* tersebut dikirim ke salah satu *user email* yang ada pada *mail server* skripsirudi.my.id maka *email* tersebut akan di blok oleh *Amavis-New* dan di deteksi sebagai virus oleh *ClamAV*, namun terdapat report *DMARC* yang memberitahukan bahwa *email* tersebut mengandung virus sehingga pesan yang masuk tidak dapat di baca, hanya report pesan, seperti terlihat pada gambar 4.68 dan gambar 4.69 berikut

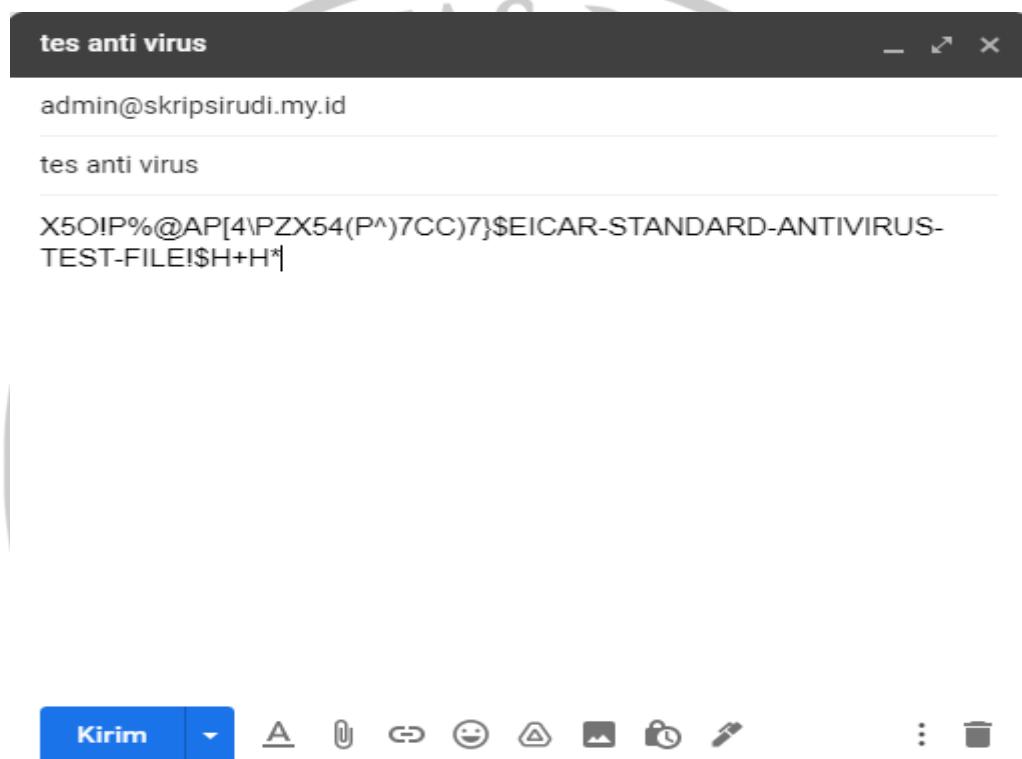
```
Aug  8 00:46:46 skripsirudi amavis[1715]: (01715-03) Blocked INFECTED (Eicar-Signature) [DiscardedInbound,Quarantined], [106.10.241.210]:35065 [106.10.241.210] <rudi.masterqq3@yahoo.com> -> <admin@skripsirudi.my.id>, Queue-ID: E5B5E115479, Message-ID: <941185908.172342.1628354801611@mail.yahoo.com>, mail_id: xlcaUXxEgIvs, Hits: -, size: 5616, dkim_sd=s2048@yahoo.com, 158 ms
Aug  8 00:46:46 skripsirudi postfix/smtp[23325]: E5B5E115479: to=<admin@skripsirudi.my.id>, relay=127.0.0.1[127.0.0.1]:10024, delay=0.43, delays=0.23/0.02/0.02/0.15, dsn=2.7.0, status=sent (250 2.7.0 Ok, discarded, id=01715-03 - INFECTED: Eicar-Signature)
```

Gambar 4. 68 email dari rudi.masterqq3@yahoo.com ke skripsirudi.my.id terblok

Gambar 4. 69 report email yang mengandung virus

#### 4.3.31 Uji Coba Mengirim *Email* yang Mengandung *Virus* dari *Gmail*

Uji coba mengirim *email* yang mengandung *virus* dilakukan dengan mengirim *email* dengan isi X5O!P%@AP[4\ZX54(P^)7CC)7}\$.EICAR-STANDARD-ANTIVIRUS-TEST-FILE!\$H+H\* yang merupakan standar *EICAR* untuk melakukan tes anti virus mail server, *email* yang mengandung virus di kirim dari layanan email Gmail Mail ke layanan email skripsirudi.my.id, seperti pada gambar 4.70 berikut. (Abrams 1999)



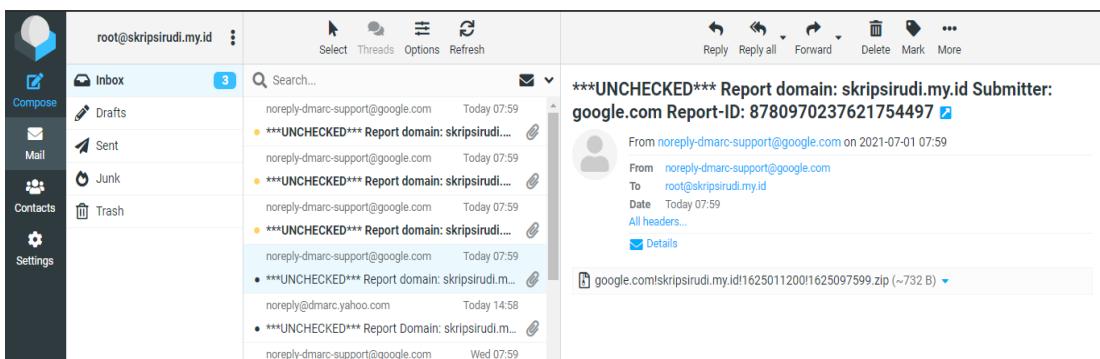
Gambar 4. 70 *EICAR Test* dari *Gmail* Setelah Penerapan

Setelah *email* yang mengandung *virus* tersebut dikirim ke salah satu *user email* yang ada pada *mail server* skripsirudi.my.id maka *email* tersebut akan diblok oleh *Amavis-New* dan di deteksi sebagai virus oleh *ClamAV*, namun terdapat report *DMARC* yang memberitahukan bahwa *email* tersebut mengandung virus

sehingga pesan yang masuk tidak dapat dibaca, hanya report pesan, seperti terlihat pada gambar 4.71 dan gambar 72 berikut.

```
Aug  8 01:00:23 skripsirudi amavis[1716]: (01716-04) Blocked INFECTED (Eicar-Signature) [DiscardedInbound,Quarantined], [209.85.166.54]:40878 [209.85.166.54] <rudi.masterqq@gmail.com> -> <admin@skripsirudi.my.id>, Queue-ID: 07F44115479, Message-ID: <CAM2Yke6mK8=TftU2b+SSHm438Z7wPUKEZBddqxr9mDechZS-Q@mail.gmail.com>, mail_id: mnApAM9Jv96f, Hits: -, size: 3244, dkim_sdn=20161025:gmail.com, 145 ms
Aug  8 01:00:23 skripsirudi postfix/smtp[23809]: 07F44115479: to=<admin@skripsirudi.my.id>, relay=127.0.0.1[127.0.0.1]:10024, delay=0.41, delays=0.22/0.04/0.02/0.13, dsn=2.7.0, status=sent (250 2.7.0 Ok, discarded, id=01716-04 - INFECTED: Eicar-Signature)
```

Gambar 4. 71 email di blok oleh antivirus client



Gambar 4. 72 report email yang mengandung virus

#### 4.3.32 Uji coba pengecekan *header email*

Uji coba pengecekan *header email* dilakukan dengan membandingkan *header email* yang dikirim dari skripsirudi.my.id ke *Gmail*, *Yahoo! Mail*, dan skripsirudi.my.id sebelum dan setelah penerapan protocol *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus*.

#### 4.3.33 *Header Email* pada *Gmail*

Uji coba ini dilakukan dengan mengirim *email* menggunakan salah satu *user email* yang ada pada layanan *email* skripsirudi.my.id ke salah satu *user email* yang ada pada layanan *email* *Gmail* kemudian melakukan pengecekan *header email* tersebut dan melakukan perbandingan terhadap *header email* setelah penerapan protocol *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus*, *header email* terlihat seperti gambar 4.73 berikut.

```
Authentication-Results: mx.google.com;
dkim=pass header.i=@skripsirudi.my.id header.s=default header.b=rflDG5ip;
dkim=pass header.i=@skripsirudi.my.id header.s=default header.b="m@n/OfeF";
spf=pass (google.com: domain of admin@skripsirudi.my.id designates 103.41.207.240 as permitted sender)
smtp.mailfrom=admin@skripsirudi.my.id;
dmarc=pass (p=QUARANTINE sp=QUARANTINE dis=NONE) header.from=skripsirudi.my.id
Received: from localhost (unknown [127.0.0.1]) by srv1.skripsirudi.my.id (Postfix) with ESMTP id E93DEC0779 for
<rudi.masterqq@gmail.com>; Sun, 27 Jun 2021 06:59:38 +0000 (UTC)
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=skripsirudi.my.id; s=default; t=1624777178;
bh=0nLVVXLw0Lbc9inyoGwFCTTHs5+Kc/F/uvToZ4sZEsQ=; h=Date:From>To:Subject;
b=rflDG5ipTJdm5ClzZUco0tLAjzV8K1BzboLSegHnF6n/GCbnyC8e/t8mb0Nk8tP
TCCwjACPnDDJMnja5RphKzbjmTKmhawC9JeBZH8pozme6hIFQITtOzGLEP0qaQyHa
YUddrY0zVqSQZoRiaB9LDBxp1uhj/rJWJKCtLi@=
X-Virus-Scanned: amavisd-new at skripsirudi.my.id
```

**Gambar 4. 73 Cuplikan Header Email pada Gmail Setelah Penerapan**

Pada gambar 4.73 terlihat perbedaan *header email* setelah penerapan protocol *DMARC*, *Dkim*, *SPF*, dan *anti virus* yaitu terdapat tambahan parameter *DMARC* yang bernilai *dmarc= pass*, *X-Virus-Scanned*, dan *Dkim-Signature* yang bernilai *dkim=pass*.

#### **4.3.34 Header Email pada Yahoo! Mail**

Uji coba ini dilakukan dengan mengirim *email* menggunakan salah satu *user email* yang ada pada layanan *email* skripsirudi.my.id ke salah satu *user* yang ada pada layanan *email* *Yahoo! Mail* kemudian melakukan pengecekan *header email* tersebut setelah penerapan protocol *DMARC*, *Dkim*, *SPF*, *anti spam*, dan *anti virus*, *header email* terlihat seperti gambar 4.74 berikut.

```

Received-SPF: pass (domain of skripsirudi.my.id designates 103.41.207.240 as permitted sender)
Authentication-Results: atlas310.free.mail.bf1.yahoo.com;
dkim=pass header.i=@skripsirudi.my.id header.s=default;
dkim=pass header.i=@skripsirudi.my.id header.s=default;
spf=pass smtp.mailfrom=skripsirudi.my.id;
dmarc=pass(p=QUARANTINE) header.from=skripsirudi.my.id;
X-Apparently-To: rudi.masterqq3@yahoo.com; Sat, 26 Jun 2021 09:45:19 +0000
X-YMailISG: zTAGvAIWLDtQvpCynw08J_uWVwlujrMtA5sCth7RlgCRCMq
_kt8VBfhw5q16ccKam1NVkXz7m1HV7INz0Fes_NR2bXYtf9SXKmxfcUYVV_
c09JnBMZVLmp0Zcu6XBC57l01ilitEQvQbRGUh8xH1E0uWjvZohfpjB1diSI
WXBe58mdoZ7KxZgBLT0aff8aKrud2v5kxV3KmNKNu0LZig4Jc91kSmsNc0F4
7hIQwtZ1R47Qtc4qXejM66FW7KSjvX.zFmTYIcn1rdRKohrl_iSrqlvjRPok
.HhinAB5g9a_eLeiaTBW0XGJ21ZTI2oScb6FhhXx6Nk9E8xLD2yEdgjPGZN
.vVAu1ltN8RL4ctlCh2w1EuDQm3HkmA7xB030aXw8IPwZ_f_p4k6Mk3ysWKjh
oSryh6cIW3_b_cXIfxWZDL3KxNEP_m8vCIBGw0JA5k3Edz5Cvfnf.p3DuoTa1
XBSoQ33VEAB.LV_nzzTe.fCcFCwbQIuqWMCMBet1oenkXMTc3iEJncjBMqRg
pcSz2.03Ix02p0qpgtQYDmpErYwlpANDYKdEROAw9906cpYf2nD._giGnyr
zV1Wvw6cdqsd_bsI3gvLmhwZ14chJK95nCHfc_3wk51fEpVH8hLnadI1x7t
oQ8rR1lpCCs72qqfxqMD.yTcrGFbpTF_ArvIPBQd98zuhG2v6dXMS22gd6hn
oydFEb79pHj2Nvbj5kcGBPkeZ1jv73L1EwETT2cONhIZZU0tdO60QFC1V4L9w
Woy2am1iyJxdqv7M4Km0o12GXgle0exaeJNGUmus4zbhijnjJ43ZlgP0pEHg
1qjuRqcEV.h2UMtr9Kn8eDGBybCY7tE7TtWLuPRjDVnIigu1vCTBdrdzCz9n
irbekyqt76vTA1kg5bkA1zHGCUlq9bhIlCdNL_90d00ZDGYY3RidBusmDI
q_Gz.7nW.M0g5a6idxXCCSkP.WuS3718mJ3XFLTd8tuXsxWHpZpXayYHovMb
qHPE7WnQajvuUF0p8YHFV2OSZML8ssVug7u1af.EF7sAaP_zU0crCYHyxamB
OF1U309TM8CnHrePBvtx6MId33Zraw--
Received: from 103.41.207.240 (EHLO srv1.skripsirudi.my.id)
by 10.197.39.201 with SMTPs
(version=TLS1_2 cipher=TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256);
Sat, 26 Jun 2021 09:45:18 +0000
Received: from localhost (unknown [127.0.0.1])
by srv1.skripsirudi.my.id (Postfix) with ESMTP id 58D87C0778
for <rudi.masterqq3@yahoo.com>; Sat, 26 Jun 2021 09:45:16 +0000 (UTC)
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=skripsirudi.my.id;
s=default; t=1624700716;
bh=Sa2S8055UojRvdFrdHMTAzofIe5wi3d9s1+uxltMZvQ=;
h=Date:From:To:Subject;
b=XbIK1ScKwlijajATc9UQTUSGZQ70709rh5Ka312DxweRjLuq9zN949hap/TSRHynfY2
ATcT0aUGbU6Wh+b0YJsjKk0EtoFY0VPYS87aUTNGR0xCd2YJ0ULF0EYk8oDheW3ZgZ
LW9C2/xNvhSeeac96YzzMMZPOTrLqpt0e8muQsg0=
X-Virus-Scanned: amavisd-new at skripsirudi.my.id

```

**Gambar 4.74 Cuplikan Header Email pada Yahoo! Mail Setelah Penerapan**

Pada gambar 4.53 terlihat perbedaan *header email* setelah penerapan protocol *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus* yaitu nilai dari parameter *Received-SPF* yang awalnya *none* menjadi *pass*, parameter *dkim* yang awalnya neutral menjadi *pass*, parameter *dmarc* yang menjadi *pass* dan terdapat tambahan parameter *DKIM-Signature* dan *X-Virus-Scanned*, (Hanif 2018).

### 4.3.35 Header Email pada skripsirudi.my.id

Uji coba ini dilakukan dengan mengirim *email* menggunakan salah satu *user email* yang ada pada layanan *email* skripsirudi.my.id ke salah satu *user* yang ada pada layanan *email* skripsirudi.my.id kemudian melakukan pengecekan *header email* tersebut setelah penerapan protocol *DMARC*,

*DKIM, SPF, anti spam, dan anti virus, header email* terlihat seperti gambar 4.75 berikut.

X-Virus-Scanned: amavisd-new at skripsirudi.my.id  
Authentication-Results: srv1.skripsirudi.my.id (amavisd-new);  
dkim=pass (1024-bit key) header.d=skripsirudi.my.id

Received: from srv1.skripsirudi.my.id ([127.0.0.1])  
by localhost (srv1.skripsirudi.my.id [127.0.0.1]) (amavisd-new port 10024)  
with ESMTP id rPXL1Plqn7Nz for <admin@skripsirudi.my.id>;  
Fri, 16 Jul 2021 21:50:31 +0800 (WITA)

Received: from localhost (skripsi.rudi.com [IPv6:::1])  
by srv1.skripsirudi.my.id (Postfix) with ESMTPA id 25EDF115453  
for <admin@skripsirudi.my.id>; Fri, 16 Jul 2021 21:50:31 +0800 (WITA)

**DKIM-Signature:** v=1; a=rsa-sha256; c=relaxed/simple; d=skripsirudi.my.id;  
s=default; t=1626443431;  
bh=bvGp0CSqPLZq0x7LiUJwPnzORqxk3whkwyFpYKfdkyw=;  
h=Date:From:To:Subject;  
b=eMupgNm5fsPe4FNYEMgBR54Ai/+H9tdMkQ2WJwxXB14PzhIRrVmZtm8QvVViNdwrZ  
+Mo9KXFmFbdR1SczMqjUP9uhWT1oxHUHje03KUT5Z1RLcTV4cwxQmsHaYacZygJ9+v  
B424nQij+yGRbJeWD8newiARBSSyjfF4n4n5kh9g=

Gambar 4. 75 Cuplikan *Header Email* skripsirudi.my.id Setelah Penerapan

Pada gambar 4.76 terlihat perbedaan *header email* setelah penerapan protocol *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus* yaitu terdapat tambahan parameter *DKIM-Signature* dan *X-Virus-Scanned*, serta terlihat *port* yang menghubungkan antara *MTA* dengan *Amavisd-New* yaitu *port* 10024.

#### 4.4 Analisa Hasil Uji Coba

Pada tahap ini akan dilakukan analisa hasil uji coba yang telah dilakukan sebelumnya. Pada analisa hasil uji coba akan ditampilkan analisa hasil uji coba pengiriman *email spoofing* sebelum dan setelah penerapan protocol *DMARC*, *DKIM* dan *SPF*, pengiriman *email spam* sebelum dan setelah penerapan *anti spam*, pengiriman *email* yang mengandung *virus* sebelum dan setelah penerapan *anti virus*, dan pengecekan *header email* sebelum dan setelah penerapan protocol *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus* (Hanif 2018).

#### 4.4.1 Analisa Hasil Uji Coba Pengiriman *Email Spoofing*

Cara yang dapat digunakan untuk mengetahui apakah sudah dilakukan proses otorisasi dan otentikasi oleh protocol *DMARC*, *DKIM* dan *SPF* adalah dengan melakukan pengiriman *email spoofing* menggunakan *Emkei's Fake Mailer* dengan mengatasnamakan salah satu *user email* pada *mail server* skripsi.rudi.my.id, kemudian *email* tersebut dikirim ke layanan *email Gmail*, *Yahoo! Mail*, dan skripsi.rudi.my.id. Berikut Analisa hasil ujicoba perbandingan sebelum diterapkan protocol *DMARC*, *DKIM* dan *SPF* dan setelah diterapkan protocol *DMARC*, *DKIM* dan *SPF* yang dilakukan pada uji coba sebelumnya, seperti terlihat pada tabel 4.1 berikut. (Hanif 2018).

Tabel 4. 1 Perbandingan Sebelum dan Setelah Penerapan protocol *DMARC*, *DKIM* dan *SPF*

N O	Fake Mailer	Layanan Email yang diatasnamakan	Layanan Email Penerima	Sebelum Penerapan	Setelah Penerapan
1	<i>Emkei's Fake Mailer</i>	Skripsi.rudi.my.id	<i>Gmail</i>	Masuk <i>Folder Inbox</i>	Diblokir Dan di report oleh <i>DMARC</i>
2	<i>Emkei's Fake Mailer</i>	Skripsi.rudi.my.id	<i>Yahoo! Mail</i>	Masuk <i>Folder Inbox</i>	Masuk <i>Folder Spam</i>
3	<i>Emkei's Fake Mailer</i>	Skripsi.rudi.my.id	Skripsi.rudi.my.id	Masuk <i>Folder Inbox</i>	Masuk <i>Folder Inbox</i>

Berdasarkan tabel 4.1 perbandingan sebelum dan setelah penerapan protocol *DMARC*, *DKIM* dan *SPF* dengan melakukan pengiriman *email spoofing* yang dikirim menggunakan *Emkei's Fake Mailer* ke layanan *email Gmail*, *Yahoo! Mail*, dan skripsi.rudi.my.id sebelum penerapan protocol *DMARC*, *DKIM* dan *SPF* yaitu *email spoofing* berhasil masuk ke *folder inbox* penerima *email* yang berada pada *mail server Gmail*, *Yahoo!*

*Mail*, dan skripsirudi.my.id sedangkan setelah penerapan protocol *DMARC*, *DKIM* dan *SPF*, *email spoofing* tersebut diblokir dan *dmarc* mereport email, dimasukan ke *folder spam* oleh layanan *email Yahoo! Mail* dan dimasukan ke *folder inbox* oleh layanan *email* skripsirudi.my.id. (Hanif 2018).

#### 4.4.2 Analisa Hasil Uji Coba Pengiriman *Email Spam*

Analisa penerapan *anti spam* dilakukan dengan mengirim *email spam* dengan menggunakan layanan *email* skripsirudi.my.id, *Yahoo! Mail*, dan *Gmail* ke layanan *email* skripsirudi.my.id untuk menguji kinerja *anti spam* sebelum dan setelah penerapan *anti spam* seperti terlihat pada tabel 4.2 berikut.(Hanif 2018).

**Tabel 4.2 Perbandingan Sebelum dan Setelah Penerapan *Anti Spam***

NO	Layanan <i>Email Pengirim</i>	Layanan <i>Email Penerima</i>	Sebelum Penerapan	Setelah Penerapan
1	<i>Yahoo! Mail</i>	SkripsiRUDI.my.id	Masuk <i>Folder Inbox</i>	Diblokir
2	<i>Gmail</i>	SkripsiRUDI.my.id	Masuk <i>Folder Inbox</i>	Diblokir
3	SkripsiRUDI.my.id	SkripsiRUDI.my.id	Masuk <i>Folder Inbox</i>	Diblokir

Berdasarkan tabel 4.2 dapat disimpulkan bahwa sebelum penerapan *anti spam*, tidak terjadi pemblokiran *email spam* oleh *Amavisd-New* sehingga *email spam* dapat masuk pada *folder inbox* pengguna yang berada pada *mail server* skripsiRUDI.my.id, sedangkan setelah penerapan *anti spam*, terjadi proses pemblokiran *email spam* oleh *Amavisd-New* sehingga *email* yang terindikasi sebagai *spam* langsung diblokir sebelum sampai pada *folder* penerima *email* (Hanif 2018).

#### 4.4.3 Analisa Hasil Uji Coba Mengirim Email Mengandung Virus

Analisa penerapan *anti virus* dilakukan dengan mengirim *email spam* dengan menggunakan layanan *email* skripsi.rudi.my.id, *Yahoo! Mail*, dan *Gmail* ke layanan *email* skripsi.rudi.my.id untuk menguji kinerja *anti spam* sebelum dan setelah penerapan *anti spam* seperti terlihat pada tabel 4.3. (Hanif 2018).

**Tabel 4.3 Perbandingan Sebelum Penerapan Anti virus**

N O	Layanan Email Pengirim	Layanan Email Penerima	Sebelum Penerapa n	Setelah Penerapa n
1	<i>Yahoo! Mail</i>	Skripsi.rudi.my.id	Masuk <i>Folder Inbox</i>	Diblokir dan di report oleh dmarc
2	<i>Gmail</i>	Skripsi.rudi.my.id	Masuk <i>Folder Inbox</i>	Diblokir dan di report oleh dmarc
3	Skripsi.rudi.my.id	Skripsi.rudi.my.id	Masuk <i>Folder Inbox</i>	Diblokir dan di report oleh dmarc

Berdasarkan tabel 4.3, dapat disimpulkan bahwa sebelum penerapan *anti virus*, tidak terjadi proses pemblokiran *email* yang mengandung *virus* oleh *Amavisd-New* sehingga *email* yang mengandung *virus* dapat masuk pada *folder inbox* pengguna *email* yang berada pada *mail server* skripsi.rudi.my.id, sedangkan setelah penerapan *anti virus*, tidak terjadi pemblokiran *email* namun ada report *DMARC* yang mereport *email* yang mengandung *virus* sehingga *email* yang terindikasi mengandung *virus* langsung di report (Hanif 2018).

#### 4.4.4 Analisa Hasil Uji Coba Pengecekan Header Email

Analisa pengecekan *header email* dilakukan dengan melihat *header email* sebelum dan setelah penerapan protocol *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus*. Perbedaan *header email* sebelum dan setelah

diterapkan protocol *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus* terlihat seperti pada tabel 4.4 berikut.

**Tabel 4.4 Perbandingan Header Email Sebelum dan Setelah Penerapan**

NO	Uji Coba	Layanan Email	DKIM-Signature	DMARC	X-Virus-Scanned	Nilai Received-SPF
1	Sebelum Penerapan	Gmail	<i>DKIM =temperror</i>	-	-	neutral
		Yahoo! Mail	<i>dkim=perm_fail</i>	-	-	none
		Skripsiруди.my.id	<i>dkim=neutral</i>	-	-	-
2	Setelah Penerapan	Gmail	Pass	Pass	Ada	Pass
		Yahoo! Mail	Pass	Pass	Ada	Pass
		Skripsiруди.my.id	Pass	-	Ada	-

Catatan : keterangan “-“ bermakna tidak terdapat pengaturan parameter tersebut.

Berdasarkan tabel 4.4, sebelum penerapan protocol *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus* terdapat parameter *DKIM-Signature* yang bernilai *temperror* dan tidak terdapat parameter *X-Virus-Scanned*, namun *Received-SPF* bernilai *neutral* pada *header email* di *Gmail*, sedangkan setelah penerapan protocol *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus* terdapat parameter *dkim=pass*, *DKIM-Signature* dan *X-Virus-Scanned*, serta *Received-SPF* bernilai *Pass* pada *header email* di *Gmail*.

Sebelum penerapan protocol *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus* tidak terdapat parameter *DKIM-Signature* yang bernilai *perm\_fail* dan tidak ada parameter *X-Virus-Scanned*, serta *Received-SPF* bernilai *none* pada *header email* di *Yahoo! Mail*, sedangkan setelah penerapan protocol *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus* terdapat parameter *dkim=pass*, *DKIM-Signature* bernilai *Pass* dan *X-Virus-Scanned*, serta *Received-SPF* bernilai *Pass* pada *header email* di *Yahoo! Mail*.

Sebelum penerapan protocol *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus* terdapat parameter *DKIM-Signature* yang bernilai *neutral* dan tidak ada parameter *X-Virus-Scanned*, sedangkan parameter *SPF-Received* juga tidak ada pada *header email* di *skripsiруди.my.id*, sedangkan setelah penerapan protocol *DMARC*, *DKIM*, *SPF*, *anti spam*, dan *anti virus* terdapat parameter

*DKIM-Signature* yang benilai Pass dan *X-Virus-Scanned*, namun tetap tidak terdapat parameter *Received-SPF* dan *DMARC* pada *header email* di skripsirudi.my.id. (Hanif 2018).



## **BAB V**

### **PENUTUP**

#### **5.1. Kesimpulan**

Berdasarkan hasil ujicoba penerapan protocol DMARC yang telah dilakukan maka dapat diperoleh kesimpulan sebagai berikut:

1. Penerapan protokol *DomainKeys Identified Mail* dapat mencegah *email spoofing* dengan cara melakukan otentikasi menggunakan metode pencocokan *private key* dan *public key* (*Asymmetric keys*).
2. Penerapan protokol *Sender Policy Framework* dapat mencegah *email spoofing* dengan cara melakukan otorisasi menggunakan metode pencocokan alamat *IP server* pengirim.
3. Penerapan protocol DMARC dapat mencegah email spoofing dan memberikan sebuah laporan kepada email pengguna asli dengan cara memlakukan otentikasi pencocokan alamat IP server dan header email.
4. Penerapan *SpamAssassin*, *ClamAV*, dan *Amavisd-New* dapat mencegah masuknya *email spam* dan *virus* dengan cara melakukan pengecekan *header*, *body*, dan *attachment email*.

#### **5.2. Saran**

Adapun saran-saran untuk pengembangan penelitian lebih lanjut adalah sebagai berikut :

1. Mengembangkan sistem *anti spam* dengan menggunakan *database* kolaboratif SpamAssassin yaitu Pyzor, Razor2, dan DCC serta menggunakan fitur *blacklist* dan *whitelist* SpamAssassin untuk memaksimalkan kinerja SpamAssassin.
2. Mengembangkan sistem *anti spam* dengan menambahkan *tools anti spam* lainnya seperti *Barracuda Central*, *Spamhaus*, *SpamCop*, *SORBS*, dan lain-lain.

3. Penambahan mekanisme email agar rentan waktu untuk DMARC dapat mengirimkan report email terlebih dahulu daripada pesan email.
4. Penambahan kebijakan di dalam protocol DMARC seperti kebijakan none dan reject.



## DAFTAR REFERENSI

- Abrams, Randy. 1999. "GIVING THE EICAR TEST FILE SOME TEETH."
- Ardhi, Naufal Herdyputra. 2020. "Pelacakan Geolocation Pada Forensik Email Terintegrasi Dengan Twitter Geo-Social Network." jakarta.  
[https://repository.uinjkt.ac.id/dspace/bitstream/123456789/53623/1/NAUFAL\\_HERDYPUTRA\\_ARDHI-FST.pdf](https://repository.uinjkt.ac.id/dspace/bitstream/123456789/53623/1/NAUFAL_HERDYPUTRA_ARDHI-FST.pdf).
- Chandra, Wirawan Nathaniel, Gede Indrawan, and I Nyoman Sukajaya. 2016. "Spam Filtering Dengan Metode Pos Tagger Dan Klasifikasi Naïve Bayes." *Jurnal Ilmiah Teknologi Informasi Asia* 10(1): 47–55.
- Desmira, Dwi Sumarto, and Ririn Yuliani. 2017. "Rancang Bangun Mail Server Berbasis Squirrelmail Menggunakan MTA (Mail Transfer Agent) Pada PT. Teras Inti Media." *Jurnal Prosisko* 4(2): 55–59. <http://ejurnal.ippmunsera.org/index.php/PROSISKO/article/view/392>.
- Erfanti, Fatkhiyah;,, Ardiantoro; Taufik, and Triyono Joko. 2016. "Jurnal JARKOM Vol. 4 No. 1 Desember 2016 OPTIMALISASI RANCANGAN JARINGAN KOMPUTER MENGGUNAKAN GOOGLE SKETCHUP - PDF Free Download.Pdf." 4(2): 82–89.
- Fachrurrazi, Sayed. 2014. "Penerapan Metode Pembeda Markov Pada Proses Pemfilteran Email Spam." *Jurnal Penelitian Teknik Informatika* 4.
- Ginantra, I Ketut, Ahmad Adil, Lilik Widyawati, and I Putu Hariyadi. "ANALISIS MERANCANG CENTOS WEB PANEL UNTUK MANAJEMEN VIRTUAL PRIVATE SERVER." 3: 1–12.  
<http://repository.universitasbumigora.ac.id/505/9/1610530197-Artikel.pdf>.
- Hanif, Naufal. 2018. "Analisa Penerapan Domainkeys Identified Mail ( Dkim ), Sender Policy Framework ( Spf ), Anti Spam , Dan Anti Virus Pada Mail Server."
- Haryanto, Muhammad Dedy, and Imam Riadi. 2014. "Analisa Dan Optimalisasi Jaringan Menggunakan Teknik Load Balancing." *Jurnal Sarjana Teknik Informatika* 2: 1370–78. <http://www.mendeley.com/research/analisa-dan-optimalisasi-jaringan-menggunakan-teknik-load-balancing>.
- Hoiriyah, Bambang Sugiantoro, and Yudi Prayudi. 2016. "Investigasi Forensik Pada E-Mail Spoofing Menggunakan Metode Header Analysis." *jurnal ilmiah dasi* 17(4): 20–25.
- Klop, Isaac, and Kevin Csuka. 2018. "Automated End-to-End e-Mail Component Testing."
- Kusmaya. 2016. "Implementasi Mail Server Menggunakan Postfix." *Jurnal Informasi VIII*(2): 49–69.
- Muarif, Muhammad Imam, and Dadan Irwan. 2018. "Sistem Auto Backup Elektronik Mail Pada Mail Server Menggunakan Cron Job." *PIKSEL : Penelitian Ilmu*

*Komputer Sistem Embedded and Logic* 5(2): 79–90.

Nadzifan, Andrian Maftuh, Farih Nazihullah, and . Syaifuddin . 2018. “Aplikasi Untuk Deteksi Adanya Spoof Pada Email.” *Sistemasi* 7(3): 268.

Nightingale, Stephen. 1945. “Email Authentication Mechanisms : NIST Technical Note 1945 Email Authentication Mechanisms : DMARC , SPF and DKIM.”

Nurlina, Nurlina, and Irmayana Irmayana. 2015. “Studi Banding Spam-Assassin Mail Server Dengan Dan Tanpa Filter Di Sisi Mail Client.” *Creative Information Technology Journal* 1(2): 77.

Puspita, Okta, Dwi Anggorowati, M Teguh Kurniawan, and Umar Yunan K S H. 2015. “Desain Dan Analisa Infrastruktur Jaringan Wireless Di Pdii-Lipi Jakarta Dengan Menggunakan Metode Network Development Life Cycle ( Ndlc ) Design and Analysis of Infrastructure Wireless Network in Pdii-Lipi Jakarta Using Network Development Life Cycle ( Nd.” *Telkom University* 2(2): 5811–19.

Saputra, Andika, and Melwin Syafrizal. 2012. “Perancangan Dan Implementasi Mail Server Pada Cv. Sanjaya Anugerah Sejahtera (Isp Jogjarigan) Berbasis Open Source.” *Data Manajemen dan Teknologi Informasi (DASI)* 13(2): 1.

Sinsuw, A E. 2014. “Analisa Performansi Algoritma Routing Di Jaringan Komputer Unsrat.” *Jurnal Teknik Elektro dan Komputer* 3(4): 28–39.

Suryana, Andri Lesmana, Reza El Akbar, and Nur Widiyasono. 2016. “Investigasi Email Spoofing Dengan Metode Digital Forensics Research Workshop (DFRWS).” *Jurnal Edukasi dan Penelitian Informatika (JEPIN)* 2(2): 111–17.

Wahyu noer hidayat. 2010. 7 Theoretical and Applied Genetics “Membangun Mail Server Berbasis Linux Menggunakan Postfix Dengan Client Squirrelmail.” Yogyakarta. <http://dx.doi.org/10.1016/j.tplants.2011.03.004%0Ahttp://dx.doi.org/10.1016/j.pbi.2010.01.004%0Ahttp://www.biomedcentral.com/1471-2156/12/42%0Ahttp://dx.doi.org/10.1016/j.biotechadv.2009.11.005%0Ahttp://www.sciencemag.org/content/323/5911/240.short%0Apape>.

Zabar, Adzan Abdul, and Fahmi Novianto. 2015. “Keamanan Http Dan Https Berbasis Web Menggunakan Sistem Operasi Kali Linux.” *Komputa : Jurnal Ilmiah Komputer dan Informatika* 4(2): 69–74.

Martinec, M. (2016), Internet. Amavisd-New. Diperoleh Juni 20, 2018, dari <https://amavis.org/#intro>



## **Header Email pada Gmail Sebelum Penerapan**

Delivered-To: rudi.masterqq3@gmail.com  
Received: by 2002:ac0:c78a:0:0:0:0:0 with SMTP id j10csp1004246imk;  
Thu, 12 Aug 2021 10:40:22 -0700 (PDT)  
X-Google-Smtp-Source:  
ABdhPJxSSb1m6/5lizzIzi9BajWWefNIizh/4RDgDxdJidlpCSXcuT1eJkKH88BEU4  
d3u4AwIZFi  
X-Received: by 2002:a63:ef12:: with SMTP id  
u18mr4755274pgh.331.1628790022825;  
Thu, 12 Aug 2021 10:40:22 -0700 (PDT)  
ARC-Seal: i=1; a=rsa-sha256; t=1628790022; cv=none;  
d=google.com; s=arc-20160816;  
b=Q9Wz6VrdsanWVmUxr719Q6soSEAdrDdYwLRSUzLHtAPNZXKEF5f3WM/cAvCUjEV  
BL  
k+x3gRtsS2PmpY4SW1Lpw5fRZQEfyFNK2Yet3HVYk6Bg4EDPHWsmA+Rbg02B3Adxxx  
pf  
dHzUiQ0mEvutr7OdCecmNdlpYRcaud/TDzuldDkbCei1Ri4PuhE17/BoCm3/u7CdCX  
mT  
0kg6YPaurrT1h87MTAiCUCeQjVnXToaAuop3X/Dw1iZJgNybgv9ekmpPVz82jDpTjm  
nR  
Y0dgkmd7gYtGO1fPV5SnSGcncJjHne2YE1ql7vQLlcafDOGIkz1MRGmMGGwXajbRSj  
+y  
K5vg==  
ARC-Message-Signature: i=1; a=rsa-sha256; c=relaxed/relaxed;  
d=google.com; s=arc-20160816;  
h=content-transfer-encoding:message-id:user-agent:subject:to:from  
:date:mime-version:dkim-signature:dkim-signature;  
bh=0nLVVXLwOLbC9inyoGwFCTTHs5+Kc/F/uvToZ4sZEsQ=;  
b=baUemm0XiENSag6oKF6qa4LRWHQ8UzD3/OmbUYEfQOwJF19UqfG38FkoQ4CGHyel  
d3  
ufc5Krs4+c9zbxe03Strp4WbIaJVWQc5r6EgXOdLwEEw0x1OdXmNkuPyTMDaphanIj  
OJ  
cTZ2kuFM6gq35m7BpDJOrkqQ4NiSiAq37K1k5mAmG4RInCcaHbcm0DqFHX5jVPtu4A  
mh  
rdaOm1f93fZtMYacyq1laCro/yWcatAP94pHaJy20EfPmvbNIRIKyvQ/UfCLl4hdTD  
L+  
19DMIyNmfsVQMe1SY8NNn5n8pbrboigEmwf9top681CR/bPoiDCSm9/Oq3/zVSZ81w  
vb  
L5Kw==  
ARC-Authentication-Results: i=1; mx.google.com;  
dkim=temperror (no key for signature) header.i=@skripsirudi.my.id  
header.s=default header.b=X5TPJCmt;  
dkim=temperror (no key for signature) header.i=@skripsirudi.my.id  
header.s=default header.b=iqCe0fxw;  
spf=neutral (google.com: 103.41.207.240 is neither permitted nor  
denied by best guess record for domain of admin@skripsirudi.my.id)  
smtp.mailfrom=admin@skripsirudi.my.id  
Return-Path: <admin@skripsirudi.my.id>  
Received: from srv1.skripsirudi.my.id ([103.41.207.240])  
by mx.google.com with ESMTPS id  
117si293887pjz.83.2021.08.12.10.40.21

for <rudi.masterqq3@gmail.com>  
(version=TLS1\_2 cipher=ECDHE-ECDSA-AES128-GCM-SHA256  
bits=128/128);  
Thu, 12 Aug 2021 10:40:21 -0700 (PDT)  
Received-SPF: neutral (google.com: 103.41.207.240 is neither  
permitted nor denied by best guess record for domain of  
admin@skripsirudi.my.id) client-ip=103.41.207.240;  
Authentication-Results: mx.google.com;  
dkim=temperror (no key for signature) header.i=@skripsirudi.my.id  
header.s=default header.b=X5TPJCmt;  
dkim=temperror (no key for signature) header.i=@skripsirudi.my.id  
header.s=default header.b=iqCe0fxw;  
spf=neutral (google.com: 103.41.207.240 is neither permitted nor  
denied by best guess record for domain of admin@skripsirudi.my.id)  
smtp.mailfrom=admin@skripsirudi.my.id  
Received: from localhost ([127.0.0.1]) by  
srv1.skripsirudi.my.id (Postfix) with ESMTP id E9F1F1163D6 for  
<rudi.masterqq3@gmail.com>; Thu, 12 Aug 2021 17:40:19 +0000 (UTC)  
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple;  
d=skripsirudi.my.id; s=default; t=1628790020;  
bh=0nLVVXLwOLbC9inyoGwFCTTHs5+Kc/F/uvToZ4sZEsQ=;  
h=Date:From:To:Subject;  
b=X5TPJCmt8Fa0gxb0Eg/rnfodk3hluRJb+4Jsg5eGD/me61tvvhq1COuOn8rlIdSj  
I  
KuSeiErGpBf3SBmAvUm0z/uZvlpK1FkULL2cgHtxdG1wpUGOG1GkPCKg42clfRVZkt  
1EIZ0t6zXWeRVJi9wVjBm3oRAuH4amo2GA6cXbWA=  
X-Virus-Scanned: amavisd-new at skripsirudi.my.id  
Received: from srv1.skripsirudi.my.id ([127.0.0.1]) by localhost  
(srv1.skripsirudi.my.id [127.0.0.1]) (amavisd-new, port 10024) with  
ESMTP id oz50k3hkN\_9K for <rudi.masterqq3@gmail.com>; Fri, 13 Aug  
2021 01:40:18 +0800 (WITA)  
Received: from localhost (skripsi.rudi.com [IPv6:::1]) by  
srv1.skripsirudi.my.id (Postfix) with ESMTPA id 104861163D2 for  
<rudi.masterqq3@gmail.com>; Fri, 13 Aug 2021 01:40:18 +0800 (WITA)  
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple;  
d=skripsirudi.my.id; s=default; t=1628790018;  
bh=0nLVVXLwOLbC9inyoGwFCTTHs5+Kc/F/uvToZ4sZEsQ=;  
h=Date:From:To:Subject;  
b=iqCe0fxwch1u812TNTtUn2xMwiVzJvu01vfo9A43jdwR7gLKU+5v2Ec/baahVrWx  
o  
3pCdaIvAkq7zXLFDKMGHfSBpKifCtH+JpZNQXt3jZfmKDGA1N/Kmh90rgk0or94m3H  
AHICRt2kmFu+m92iD449me5pLAREHfGR8OqfaePo=  
MIME-Version: 1.0  
Date: Fri, 13 Aug 2021 01:40:17 +0800  
From: admin@skripsirudi.my.id  
To: rudi.masterqq3@gmail.com  
Subject: cek header  
User-Agent: Roundcube Webmail/1.4.11  
Message-ID: <070c35ae979e26db1108cb4c50229f11@skripsirudi.my.id>  
X-Sender: admin@skripsirudi.my.id  
Content-Type: text/plain; charset=US-ASCII; format=flowed  
Content-Transfer-Encoding: 7bit

Hallo

### **Header Email pada Gmail Setelah Penerapan**

Delivered-To: rudi.masterqq3@gmail.com  
Received: by 2002:ac0:9b09:0:0:0:0:0:0 with SMTP id p9csp3352743imd;  
Sun, 4 Jul 2021 08:26:04 -0700 (PDT)  
X-Google-Smtp-Source:  
ABdhPJxVCFbzN8G/wFBbGK3Mjk/sLKS95jBj5j7sttVAnbGwsLYDV0Jfb/Xx50m4s  
V2ZjgpJZQe  
X-Received: by 2002:a17:90a:d713:: with SMTP id  
y19mr10108676pju.42.1625412364751;  
Sun, 04 Jul 2021 08:26:04 -0700 (PDT)  
ARC-Seal: i=1; a=rsa-sha256; t=1625412364; cv=none;  
d=google.com; s=arc-20160816;  
  
b=B4AQbFcCa31uvOkBoMN56LqQ/E1G5yVM3ISzqOde3hVNmaxyREUz+itD0NKoo0J6  
T7  
  
1WLQxzfHPOJWxT3UcVIrffpvfhcbRtjR+/9x2uJUXgal6mteXYw77X1Au7z0kachZE  
12  
  
DqO/7ategYEaYvOEuKrd5yCKC/YgyQcLxWtdjmYcGsqFe4ayehuFFZnfCwJsgx6Dc/  
fz  
  
jNAri9899Rme6qsnAikcyseDwS1gIKvEo14KtB+sJx8PH0HYFZIK/XZPbM/23oH9je  
i3  
  
Z53E6g9daOEOf5V0jdBj4l033w4dZeWt9k8pULnGC3IU8c91aImq8S3kQZhULZGwa  
Zj  
O+aQ==  
ARC-Message-Signature: i=1; a=rsa-sha256; c=relaxed/relaxed;  
d=google.com; s=arc-20160816;  
h=content-transfer-encoding:message-id:user-  
agent:subject:to:from  
:date:mime-version:dkim-signature:dkim-signature;  
bh=U1B15v6ZtlHGESAwKHttp70/7Ndg3HwEZbXrvJEwlzhY=;  
  
b=mKB6v2+SQMXIUV5v7pbVotVTNqHqdFwRQdbzcLYTcyLyHkKzpwoLRLA7977Tjtuf  
4m  
  
m0IynUKYVeDEQw8HjaUyvWH9ZVTkHpmGIgSMHw7gJt6XFGLl9oriqRpvdmn+glezg  
Fo  
  
xijjigSF15h9M1Ea50KMYXzDweRGD77/fQoJo+0Qt1p4jupJUuLP139d3A2rJltuvJ  
Qt  
  
KOFnF3kWaUbqa2q1+0rKzifK3EwcuwYOZd9VSDrSKQni+evo4imvJN2k+XWt0+6vx5  
1L

iQKy154nXYbli0/3YRINYWSPY8hVADIaSitrj+AmKdP7Sb4VbfC8eBQzlOe+Utl3mD  
pd

1/Jg==

ARC-Authentication-Results: i=1; mx.google.com;  
dkim=pass header.i=@skripsirudi.my.id header.s=default  
header.b=sR9QIzFn;  
dkim=pass header.i=@skripsirudi.my.id header.s=default  
header.b=KWPsjVRI;  
spf=pass (google.com: domain of root@skripsirudi.my.id  
designates 103.41.207.240 as permitted sender)  
smtp.mailfrom=root@skripsirudi.my.id;  
dmarc=pass (p=QUARANTINE sp=QUARANTINE dis=NONE)  
header.from=skripsirudi.my.id  
Return-Path: <root@skripsirudi.my.id>  
Received: from srv1.skripsirudi.my.id ([103.41.207.240])  
by mx.google.com with ESMTPS id  
u5si24495287pjm.151.2021.07.04.08.26.03  
for <rudi.masterqq3@gmail.com>  
(version=TLS1\_2 cipher=ECDHE-ECDSA-AES128-GCM-SHA256  
bits=128/128);  
Sun, 04 Jul 2021 08:26:03 -0700 (PDT)  
Received-SPF: pass (google.com: domain of root@skripsirudi.my.id  
designates 103.41.207.240 as permitted sender) client-ip=103.41.207.240;  
Authentication-Results: mx.google.com;  
dkim=pass header.i=@skripsirudi.my.id header.s=default  
header.b=sR9QIzFn;  
dkim=pass header.i=@skripsirudi.my.id header.s=default  
header.b=KWPsjVRI;  
spf=pass (google.com: domain of root@skripsirudi.my.id  
designates 103.41.207.240 as permitted sender)  
smtp.mailfrom=root@skripsirudi.my.id;  
dmarc=pass (p=QUARANTINE sp=QUARANTINE dis=NONE)  
header.from=skripsirudi.my.id  
Received: from localhost (unknown [127.0.0.1]) by  
srv1.skripsirudi.my.id (Postfix) with ESMTP id ECABC102800 for  
<rudi.masterqq3@gmail.com>; Sun,  
4 Jul 2021 15:26:01 +0000 (UTC)  
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple;  
d=skripsirudi.my.id; s=default; t=1625412361;  
bh=U1B15v6Zt1HGESAwKHttp70/7Ndg3HwEZbXrvJEw1zhY=;  
h=Date:From:To:Subject;  
b=sR9QIzFnxEat3Zx+MCKtzLHyTwGI9fkOsOQ3jae0CFxlyFiSdsMgJmNf5mfrZC72  
3

j+PJMhvZCjwHTHv7fZyq4qJCanGULEguHZhbGwFHycFPQkU7q95hwDDZdEMzwblkeS  
Z3MhJ1aP1y3aZ+pAxnZ8UdPY1G59CfTeSUC/0iiI=  
X-Virus-Scanned: amavisd-new at skripsirudi.my.id  
Received: from srv1.skripsirudi.my.id ([127.0.0.1]) by localhost  
(srv1.skripsirudi.my.id [127.0.0.1]) (amavisd-new, port 10024) with  
ESMTP id j-QbgAbpYhZ0 for <rudi.masterqq3@gmail.com>; Sun,

4 Jul 2021 23:25:28 +0800 (WITA)  
Received: from localhost (skripsi.rudi.com [IPv6:::1]) by  
srv1.skripsirudi.my.id (Postfix) with ESMTPA id 407D0AE18B for  
<rudi.masterqq3@gmail.com>; Sun,  
4 Jul 2021 23:25:28 +0800 (WITA)  
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple;  
d=skripsirudi.my.id; s=default; t=1625412328;  
bh=U1B15v6Zt1HGESEAwKHttp7O/7Ndg3HwEZbXrvJEw1zhY=;  
h=Date:From:To:Subject;  
b=KWPsjVRiMVy2sU4UEjdaTrlWog7MEfI9f5DP13Bv+i7x1ivQe9i+f1M5cSfz30X4  
A  
  
vs18J80zDFQgxQ4eKMxxsdHBoLTVsTOdLEb7VC1gqanQXi1FpF9UxAq3+bkYTrrSeT  
DQy9/IOja3hAWTW0gWiYnUUQmE+eRQH1A/MtOQxk=  
MIME-Version: 1.0  
Date: Sun, 04 Jul 2021 23:25:28 +0800  
From: root@skripsirudi.my.id  
To: rudi.masterqq3@gmail.com  
Subject: untuk rudi  
User-Agent: Roundcube Webmail/1.4.11  
Message-ID: <56616ab8ce106551942ed8e265e624c9@skripsirudi.my.id>  
X-Sender: root@skripsirudi.my.id  
Content-Type: text/plain; charset=US-ASCII; format=flowed  
Content-Transfer-Encoding: 7bit

Hallloooooooooooooooooooooooooooooooooooo

### ***Header Email pada Yahoo! Mail Sebelum Penerapan***

Received: from 10.222.142.149  
by atlas301.free.mail.nel.yahoo.com with HTTPS; Thu, 12 Aug 2021  
18:17:56 +0000  
Return-Path: <root@skripsirudi.my.id>  
X-Originating-Ip: [103.41.207.240]  
Received-SPF: none (domain of skripsirudi.my.id does not designate  
permitted sender hosts)  
Authentication-Results: atlas301.free.mail.nel.yahoo.com;  
dkim=perm\_fail header.i=@skripsirudi.my.id header.s=default;  
dkim=perm\_fail header.i=@skripsirudi.my.id header.s=default;  
spf=none smtp.mailfrom=skripsirudi.my.id;  
dmarc=unknown header.from=skripsirudi.my.id;  
X-Apparently-To: rudi.masterqq3@yahoo.com; Thu, 12 Aug 2021  
18:17:56 +0000  
X-YMailISG: FX\_DJPkWLdt.jRXPrZ4qA1\_5vuZthhvze7JL0gNoiylFdXU0  
bObb\_dG7ezBsPtRSPTiC9179\_b4hggC\_u7EE6Var0UxFU6SzZ76Xt0roJtm  
nTgsjXyg1Numt921PIuLfQ2gTGb\_dkimHj5xXbUmv\_N1GbwsC43K5dex3y43  
KUVJEQXv.O21VMQ9Ve0I0djz1I0Hh8lj.Etw9pgU3eCzN.Z1DyKMCUrdlzTN  
9nlmT1C9NjsogoBVrrm4r6VnO.XGXCjZO4jhHjxxqOyf7C2TC9z4\_Nr9PwOf  
dVys6ZkPa8wsDopNvSDH0uqES5wfrxQsSFJ0NTc6606Y5HTA7n.Aoz4xL9x  
mLPW8BeTrsmEZzwJuz\_sTycb73co6ACKQ4omo2msCN4bx100B4.sEqUMr6U5  
zya2zcbzYkWLPsLTEmcBbiERUULEXCWenqiygNDklPeXLQN5vdWJ2cC7Y0k

iFUQOqDTfZwOSUWGuFbJZePDBFrjbdYGxF1yd0q2Faek3nWs iMdCpNHO bQcK  
FnXn7HQjMIGvhP5ZaYYe.TtPr4CJRxaVDDh.zZhqe7H6iULamOxMkzS1Ffy  
a8eCLFauxI76.bdX2EsdubHd5wuNuNPntZC\_8KuW8notil2SQGxawwYrcQhw  
P8tRB9Y7mH0g7uK4XA0QaeKx2IYgPIaTNahZw8F3e8IecQnLETpPy8wpkk6  
FuPXIo33NH8JVZ4syg70hcs1RRQH5rh2pWMWvxUe6BVTg.Hy91oFqZti0\_\_IE  
r1V8SwmlkhF5EweH4bcWgoz4E0XYIOm\_c3.D3fs157athrls.\_NBewnEn9r  
5SLRMYfKh3EWXAr0NzyIeI1Jh44K.P8o2M0Bc271AnpyDRSZbi7Lh99wuq0h  
uW\_uzRQXI Kou5KAzRqGITud\_IMZbabSFaX9pprvpOmCIq8EP3dhCPdyrQfo9  
AEEUosMTLWMOTb8ovGfB1UD7Tjh1Uycs7Q--

Received: from 103.41.207.240 (EHLO srv1.skripsirudi.my.id)  
by 10.222.142.149 with SMTPs  
(version=TLS1\_2 cipher=TLS\_ECDHE\_RSA\_WITH\_AES\_128\_GCM\_SHA256);  
Thu, 12 Aug 2021 18:17:56 +0000

Received: from localhost (unknown [127.0.0.1])  
by srv1.skripsirudi.my.id (Postfix) with ESMTP id BDAE21163D6  
for <rudi.masterqq3@yahoo.com>; Thu, 12 Aug 2021 18:17:51  
+0000 (UTC)

DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple;  
d=skripsirudi.my.id;  
s=default; t=1628792271;  
bh=0nLVVXLwOLbC9inyoGwFCTTHs5+Kc/F/uvToZ4sZEsQ=;  
h=Date:From:To:Subject;  
b=aM4PWQ+129nITtg9PgSNpECQcy+auWCyPUek33tCyiSD8liwgYdvogMTaq  
vB7s/V1

M9+YStCqi5DW/CurpLmR+95QRcJhGokbEB1I089rLsmo1x4+Se7Jh7MZTTjd3m3uKD  
WmHCcqzRHi4xNRwCGEGcU2nkprD5Fd4AVZS6clrs=  
X-Virus-Scanned: amavisd-new at skripsirudi.my.id

Received: from srv1.skripsirudi.my.id ([127.0.0.1])  
by localhost (srv1.skripsirudi.my.id [127.0.0.1]) (amavisd-new, port 10024)  
with ESMTP id rvSkEcEMmAB7 for <rudi.masterqq3@yahoo.com>;  
Fri, 13 Aug 2021 02:17:37 +0800 (WITA)

Received: from localhost (skripsi.rudi.com [IPv6:::1])  
by srv1.skripsirudi.my.id (Postfix) with ESMPA id  
AAA251163D2  
for <rudi.masterqq3@yahoo.com>; Fri, 13 Aug 2021 02:17:37  
+0800 (WITA)

DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple;  
d=skripsirudi.my.id;  
s=default; t=1628792257;  
bh=0nLVVXLwOLbC9inyoGwFCTTHs5+Kc/F/uvToZ4sZEsQ=;  
h=Date:From:To:Subject;  
b=S6T2KumTP4vqj76oapORPwg1AhaSdNleurz6vRnlUWcqEaBv+jx/Oq6d1a  
pdQeJEP

B0yQ03TSzgG1VEcirTRis/n2hDiobIZtgOsOE9Z0dVHp6fpgh2axp7BclMR7h+KPd/  
Vn78nAdP7n1eKUqZhYW8HiUiUIJkR2vWnHqlt+4g=  
MIME-Version: 1.0  
Date: Fri, 13 Aug 2021 02:17:37 +0800  
From: root@skripsirudi.my.id  
To: rudi.masterqq3@yahoo.com

Subject: cek header  
User-Agent: Roundcube Webmail/1.4.11  
Message-ID: <851bcab84a2ab60ce22bf2134dc56a9f@skripsirudi.my.id>  
X-Sender: root@skripsirudi.my.id  
Content-Type: text/plain; charset=US-ASCII;  
format=flowed  
Content-Transfer-Encoding: 7bit  
Content-Length: 6

Hallo

### ***Header Email pada Yahoo! Mail Setelah Penerapan***

Received: from 10.197.39.201  
by atlas310.free.mail.bf1.yahoo.com with HTTPS; Sat, 26 Jun 2021  
09:45:18 +0000  
Return-Path: <admin@skripsirudi.my.id>  
X-Originating-Ip: [103.41.207.240]  
Received-SPF: pass (domain of skripsirudi.my.id designates  
103.41.207.240 as permitted sender)  
Authentication-Results: atlas310.free.mail.bf1.yahoo.com;  
dkim=pass header.i=@skripsirudi.my.id header.s=default;  
dkim=pass header.i=@skripsirudi.my.id header.s=default;  
spf=pass smtp.mailfrom=skripsirudi.my.id;  
dmarc=pass (p=QUARANTINE) header.from=skripsirudi.my.id;  
X-Apparently-To: rudi.masterqq3@yahoo.com; Sat, 26 Jun 2021  
09:45:19 +0000  
X-YMailISG: zTAGvAIWLDTQvpCynw08J\_uWVwlujrMtA5sCth7RlgCRCMq  
\_Kt8VBfhw5qi6ccKam1NVkcXz7m1HV7INz0Fes\_WR2bXYtf9SXKmxFCUYYV\_  
cO9JnBMZVLmPfOZcu6XBc5710iliteQvQbRGUh8xH1E0uWjvZoHfpjB1diSI  
WXBe58mdoZ7KxZgBLTOaff8aKrud2v5kxV3KmNKNU0LZig4Jc9IkSmsNc0F4  
7hIQWtz1R47Qtc4qXejM66FW7KSjvx.zFmTYIcn1rdRKohrl\_iSrqlvjRPok  
.HHinABsg9a\_\_eLeiatTBWOXGJ2IZTI2oScb6FhhXx6Nk9E8xLD2yEdGjPGZN  
vVAulltN8RL4ct1Ch2wWEuDQm3HkmA7xB030aXw8IPwZ\_f\_p4k6Mk3ysWKjh  
oSryh6cIW3\_b\_cX1FxWZDL3KxNEP\_m8vCIBGw0JA5k3Edz5Cvfng.p3DuTal  
XBSoQ33VEAB.LV\_nzzTe.fCcFCwbQIuqWMCMBet1oenkXMTc3iEJncjBMqRg  
pcSz2.O3Ix02p0qpgtQYDmpErYwlPANDYKdEROAw9906cpYf2nD.\_giGnyr  
zv1Wvw6cdqsd\_\_bsI3gvLmhwZ14chJK95nCHfc\_3wk5lfEpVH8hLnadI1x7t  
oQ8rR1LpCCs72qqfXqmD.yTcrGFbpTF\_ArvIPBQd98zuhG2v6dXMS2Zgd6hn  
oydFEb79pHj2Nvj5kcGPkeZ1Jv7JL1EWETT2cONnIZZU0tdO60QFC1V4L9w  
Woy2ami1yJXdqv7M4KmOo12GXg1eOexaeJNGUmus4zbhijnjJ43Z1gPQpEHg  
1qjuRqcEV.h2UMtr9KN8eDGBybCY7tE7TtWLuPRjDVnIigu1vCTBdrdzZc9n  
irbekyqtp76vTA1kgbsbkAIzHGCuLq9bhIlCdNL\_90d00ZDZGYY3RidBusmdI  
q\_Gz.7nW.M0g5a6idxXCCSkP.WuS3718mJ3XFLTd8tuXsxWHpZpXayYHovMb  
qHPE7WWQajvuF0p8YHFV2OSZMIL8ssVug7u1af.EF7sAaP\_zU0crCYHyxamB  
OF1U309TM8CnHrePBvtx6mId33Zraw--  
Received: from 103.41.207.240 (EHLO srv1.skripsirudi.my.id)  
by 10.197.39.201 with SMTPs  
(version=TLS1\_2 cipher=TLS\_ECDHE\_RSA\_WITH\_AES\_128\_GCM\_SHA256);  
Sat, 26 Jun 2021 09:45:18 +0000  
Received: from localhost (unknown [127.0.0.1])

by srv1.skripsirudi.my.id (Postfix) with ESMTP id 5BD87C0778  
for <rudi.masterqq3@yahoo.com>; Sat, 26 Jun 2021 09:45:16  
+0000 (UTC)  
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple;  
d=skripsirudi.my.id;  
s=default; t=1624700716;  
bh=Sa2S8055UojRvdFrdHMTAzofIe5wi3d9sl+uxltMZvQ=;  
h=Date:From:To:Subject;  
b=XbIK1ScKWjajATc9UQTUSGZQT07O9rhSKa312DxweRjLuq9zN949hap/TS  
RHynfY2

ATctT0aUGbU6Wh+b0YJsKk0EtoFY0VPYS87aUTNGR0xCd2YkJ0U1F0EYk8oDheW3ZgZ  
LW9C2/xNvhSeeac96YzzMMZPOTrLqptOe8mUQsg0=  
X-Virus-Scanned: amavisd-new at skripsirudi.my.id  
Received: from srv1.skripsirudi.my.id ([127.0.0.1])  
by localhost (srv1.skripsirudi.my.id [127.0.0.1]) (amavisd-new, port 10024)  
with ESMTP id SpwXMnEDiF7B for <rudi.masterqq3@yahoo.com>;  
Sat, 26 Jun 2021 17:45:02 +0800 (WITA)  
Received: from localhost (skripsi.rudi.com [IPv6:::1])  
by srv1.skripsirudi.my.id (Postfix) with ESMTPA id B5DCEC0773  
for <rudi.masterqq3@yahoo.com>; Sat, 26 Jun 2021 17:45:01  
+0800 (WITA)  
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple;  
d=skripsirudi.my.id;  
s=default; t=1624700701;  
bh=Sa2S8055UojRvdFrdHMTAzofIe5wi3d9sl+uxltMZvQ=;  
h=Date:From:To:Subject;  
b=VHRFzTLAkqRfax51BeP968jDFdtNx1+4BJMMwetNQTJvSnvfoqpa0vSHXu  
IHX3ouV

3G8hViRN+R+TYIXyC9/fXWS3t1GSBNN/rWblbPQKbYtuSU+8vVpDj64EKa1YENE+Dm  
f8OfS/ep+v7FwfUgyRGZLQYtlOSPByjkfEcDBFek=  
MIME-Version: 1.0  
Date: Sat, 26 Jun 2021 17:45:01 +0800  
From: admin@skripsirudi.my.id  
To: rudi.masterqq3@yahoo.com  
Subject: coba 5  
User-Agent: Roundcube Webmail/1.4.11  
Message-ID: <3c03c134e3f074eae1882a0ef6129b75@skripsirudi.my.id>  
X-Sender: admin@skripsirudi.my.id  
Content-Type: text/plain; charset=US-ASCII;  
format=flowed  
Content-Transfer-Encoding: 7bit  
Content-Length: 21

ini pesan bukan spam

## **Header Email pada skripsirudi.my.id Sebelum Penerapan**

Return-Path: <root@skripsirudi.my.id>  
Delivered-To: admin@skripsirudi.my.id  
Received: from localhost (unknown [127.0.0.1])  
by srv1.skripsirudi.my.id (Postfix) with ESMTP id B32D81163D7  
for <admin@skripsirudi.my.id>; Thu, 12 Aug 2021 19:51:54 +0000  
(UTC)  
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple;  
d=skripsirudi.my.id;  
s=default; t=1628797914;  
bh=0nLVVXLwOLbC9inyoGwFCTTHs5+Kc/F/uvToZ4sZEsQ=;  
h=Date:From:To:Subject;  
  
b=jXpaCSi+uBzCiY1R6ETfkD3RLvKOXKv34aXFMcCFd7W1fiu+qKqReJ4cXNFdTnkw  
B  
  
D5wxbdzBfSrWDcxkvI/inSizFo8ZwnuQQviDLa8aoFiOMDFgGs3su/OmmEvf1O3mUx  
HH9/GZXLqtjeMcko2017iYYbr0rpGpOVduniayac=  
X-Virus-Scanned: amavisd-new at skripsirudi.my.id  
Authentication-Results: srv1.skripsirudi.my.id (amavisd-new);  
dkim=neutral  
reason="invalid (public key: not available)"  
header.d=skripsirudi.my.id  
Received: from srv1.skripsirudi.my.id ([127.0.0.1])  
by localhost (srv1.skripsirudi.my.id [127.0.0.1]) (amavisd-new,  
port 10024)  
with ESMTP id CvWl7sq-udlU for <admin@skripsirudi.my.id>;  
Fri, 13 Aug 2021 03:51:52 +0800 (WITA)  
Received: from localhost (skripsi.rudi.com [IPv6:::1])  
by srv1.skripsirudi.my.id (Postfix) with ESMTPA id 132911163CE  
for <admin@skripsirudi.my.id>; Fri, 13 Aug 2021 03:51:52 +0800  
(WITA)  
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple;  
d=skripsirudi.my.id;  
s=default; t=1628797912;  
bh=0nLVVXLwOLbC9inyoGwFCTTHs5+Kc/F/uvToZ4sZEsQ=;

h=Date:From:To:Subject;  
  
b=sDAnWGLh3cyodXAmiHe56GDwol/cOd7o8GvN1ECeq0MYdwh2JuUPKH94bfnHP9No  
r  
  
sFWclft69UHTxZ//TXUPCvHiUDzKg7tOCEzFmAQHQoHYujwYAEpW2f12U4YPa4iIi9  
YgzyXfFa16ejvb+VaMCn1QEpqjMKZy2t3tUfbhY=  
  
MIME-Version: 1.0  
  
Date: Fri, 13 Aug 2021 03:51:51 +0800  
  
From: root@skripsirudi.my.id  
  
To: admin@skripsirudi.my.id  
  
Subject: coba header  
  
User-Agent: Roundcube Webmail/1.4.11  
  
Message-ID: <8f3d93e611d6c61a6ae42b44dd3e862c@skripsirudi.my.id>  
  
X-Sender: root@skripsirudi.my.id  
  
Content-Type: text/plain; charset=US-ASCII;  
  
format=flowed  
  
Content-Transfer-Encoding: 7bit

### ***Header Email pada skripsirudi.my.id Setelah Penerapan***

Return-Path: <root@skripsirudi.my.id>  
Delivered-To: admin@skripsirudi.my.id  
Received: from localhost (unknown [127.0.0.1])  
by srv1.skripsirudi.my.id (Postfix) with ESMTP id 70DD7115453  
for <admin@skripsirudi.my.id>; Sun, 25 Jul 2021 10:21:43 +0000  
(UTC)  
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple;  
d=skripsirudi.my.id;  
s=default; t=1627208503;  
bh=tn1VgA14qoyFxkr50v+nGh+plsSSbpFhPx/h930jhD0=;  
h=Date:From:To:Subject;  
  
b=PCrzrWwyC3II7ykvOtMLM21XAk71ArLYO3g1kQGDQmQ2FS8k8AJeNp7ebEgkBbvP  
D  
  
q4MbNH+AGrq1LNN66wLzs9YrK2+DIS8QvedvFQ6emAXdsm0T7JaWQI5ZDVDtxCghSp  
6KDPR3dg2xyURw5v7rNa3Kshx1RrDnc8Um4WUQKo=  
X-Virus-Scanned: amavisd-new at skripsirudi.my.id  
Authentication-Results: srv1.skripsirudi.my.id (amavisd-new);  
dkim=pass (1024-bit key) header.d=skripsirudi.my.id  
Received: from srv1.skripsirudi.my.id ([127.0.0.1])

by localhost (srv1.skripsirudi.my.id [127.0.0.1]) (amavisd-new,  
port 10024)  
with ESMTP id Vq7CYZi30Y5q for <admin@skripsirudi.my.id>;  
Sun, 25 Jul 2021 18:21:41 +0800 (WITA)  
Received: from localhost (skripsi.rudi.com [IPv6:::1])  
by srv1.skripsirudi.my.id (Postfix) with ESMTPA id 159D1115454  
for <admin@skripsirudi.my.id>; Sun, 25 Jul 2021 18:14:19 +0800  
(WITA)  
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple;  
d=skripsirudi.my.id;  
s=default; t=1627208059;  
bh=tn1VgAl4qoyFxkr50v+nGh+plsSSbpFhPx/h930jhD0=;  
h=Date:From:To:Subject;  
  
b=bjDXIFipjJf9KRPhkTlUWzft3KI15rLZ5aroCpG+rfg1QhhokYN2fKBAK3wJVrt  
+  
  
asalHad0WrL4Y+Gf8Kf18pGfO1sdsEIReVhmMHX6PA/XirebbQNO/QrIiBZsX14E/4  
YX2ErOBn9SA1xaRTYt389x3OClj+WMBHuw2h5Fo=  
MIME-Version: 1.0  
Date: Sun, 25 Jul 2021 18:14:18 +0800  
From: root@skripsirudi.my.id  
To: admin@skripsirudi.my.id  
Subject: testing lagi  
User-Agent: Roundcube Webmail/1.4.11  
Message-ID: <2d0a7bcd9f2339158bf38e62e1547829@skripsirudi.my.id>  
X-Sender: root@skripsirudi.my.id  
Content-Type: text/plain; charset=US-ASCII;  
format=flowed  
Content-Transfer-Encoding: 7bit