**ANALISA PENERAPAN *DOMAINKEYS IDENTIFIED MAIL* (*DKIM*), *SENDER POLICY FRAMEWORK* (*SPF*), *ANTI SPAM*, DAN *ANTI VIRUS* PADA *MAIL SERVER***

**SKRIPSI**



**Oleh :**

**NAUFAL HANIF**

**1410530129**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER**

**(STMIK) BUMIGORA**

**MATARAM**

**2018**

**ANALISA PENERAPAN *DOMAINKEYS IDENTIFIED MAIL* (*DKIM*), *SENDER POLICY FRAMEWORK* (*SPF*), *ANTI SPAM*, DAN *ANTI VIRUS* PADA *MAIL SERVER***

**SKRIPSI**



Diajukan Sebagai Salah Satu Syarat untuk Memenuhi Kebulatan Studi

Jenjang Strata Satu (S1) Program Studi Teknik Informatika

Pada Sekolah Tinggi Manajemen Informatika dan Komputer

(STMIK) Bumigora Mataram

**Oleh :**

**NAUFAL HANIF**

**1410530129**

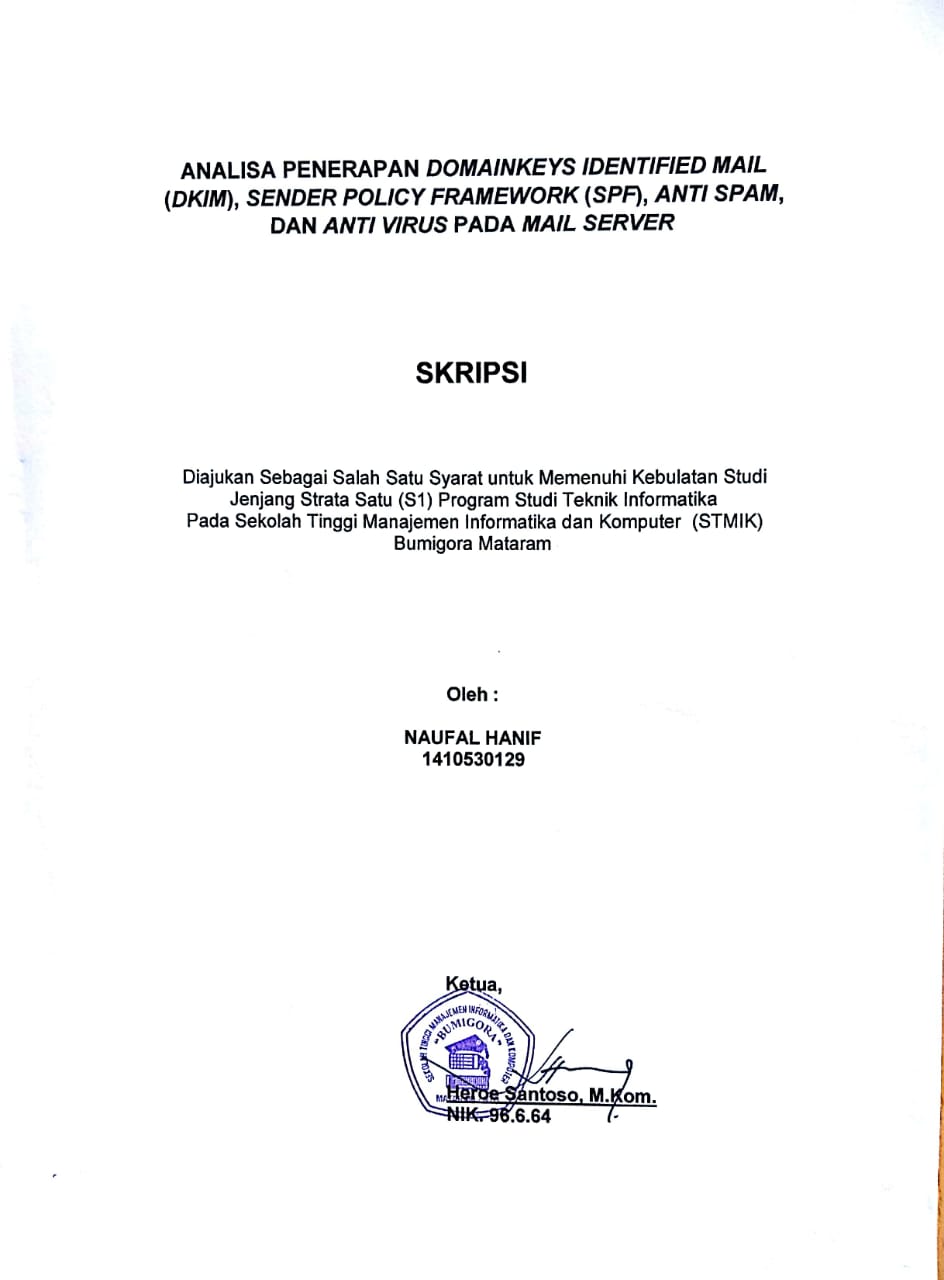
**PROGRAM STUDI TEKNIK INFORMATIKA**

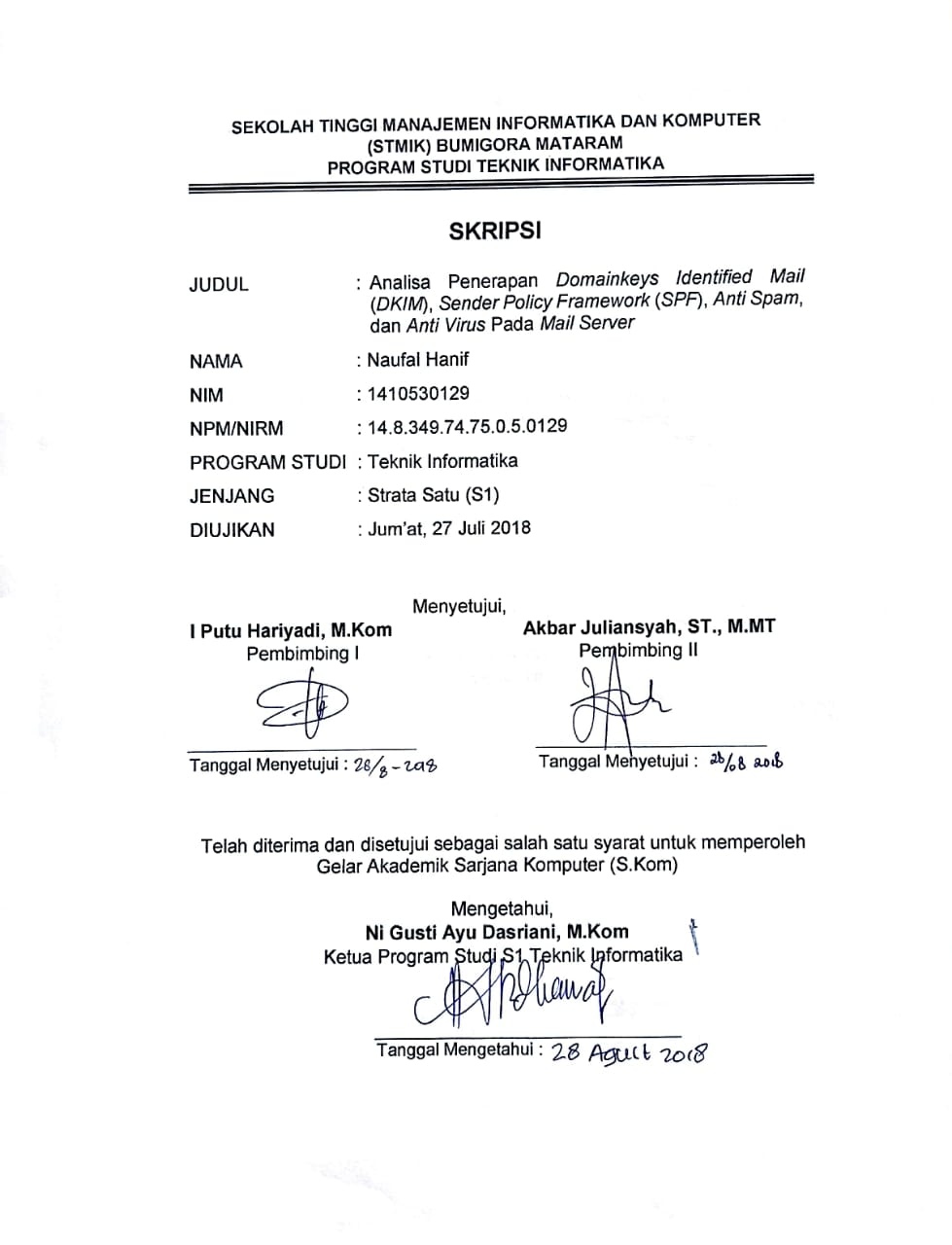
**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER**

**(STMIK) BUMIGORA**

**MATARAM**

**2018**



****

****

# **KATA PENGANTAR**

Puji syukur penulis panjatkan kepada Allah SWT yang telah memberikan rahmat-Nya sehingga penulis dapat menyelesaikan Skripsi yang berjudul “Analisa Penerapan *DomainKeys Identified Mail* (*DKIM*), *Sender Policy Framework* (*SPF*), *Anti Spam*, dan *Anti Virus* pada *Mail Server*”.

Terselesaikannya skripsi ini tidak terlepas dari bantuan berbagai pihak, dan dengan segala kerendehan hati penulis ingin menyampaikan ucapan terima kasih serta penghargaan setinggi-tingginya kepada:

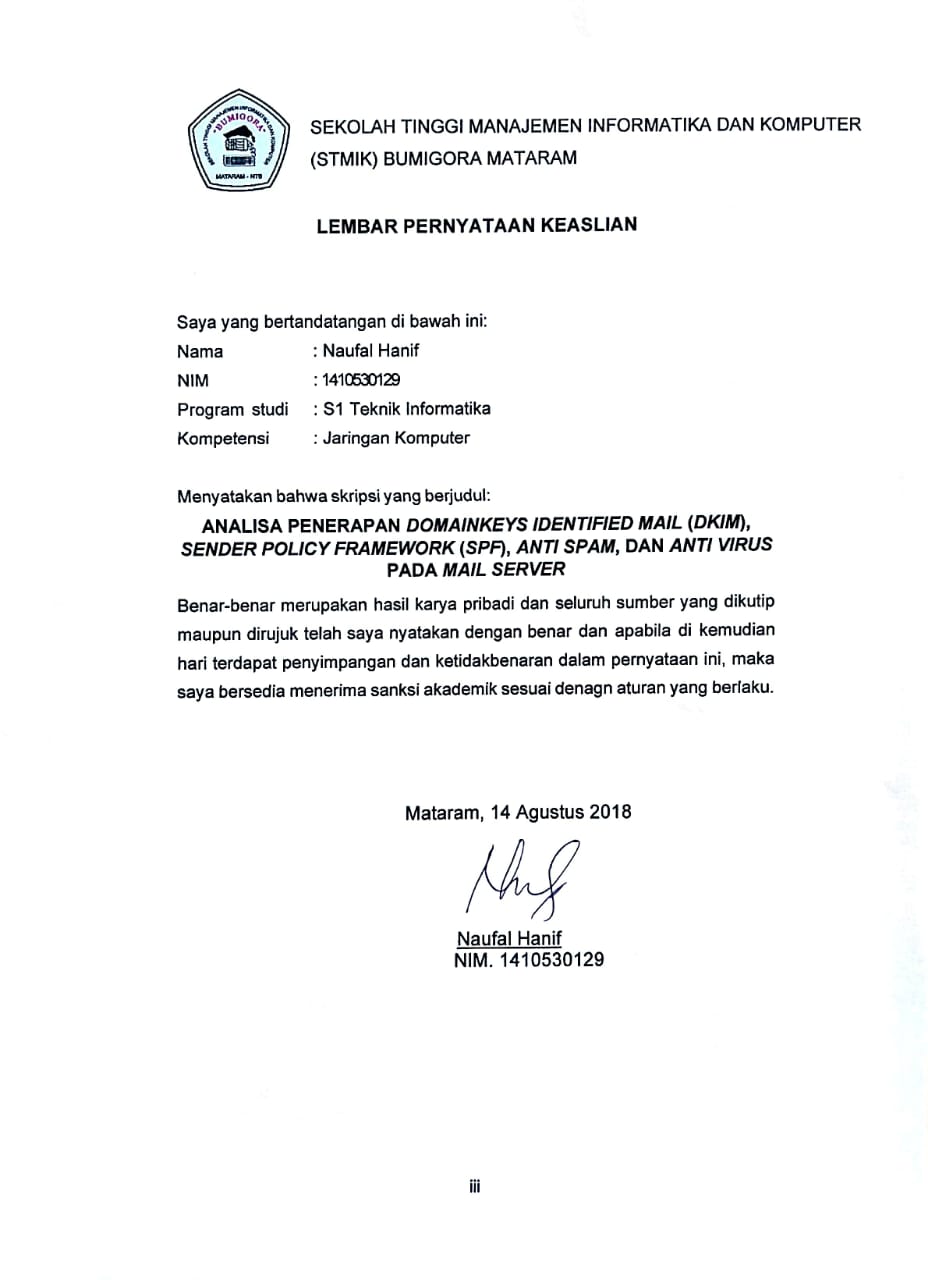
1. Bapak Heroe Santoso, M.Kom, selaku ketua Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Bumigora Mataram.
2. Ibu Ni Gusti Ayu Dasriani, M.Kom, selaku ketua Program Studi S1 Teknik Informatika.
3. Bapak I Putu Hariyadi, M.Kom, selaku Pembimbing Utama dalam mengerjakan skripsi ini.
4. Bapak Akbar Juliansyah, ST., M.MT, selaku Pembimbing Kedua dalam mengerjakan skripsi ini.
5. Bapak/Ibu dosen yang telah memberikan ilmu selama dalam perkuliahan.
6. Tidak terlupakan Ibu dan Bapak tercinta serta Saudara dan Keluarga yang telah memberikan dukungan moril dan materi serta mendo’akan, memberikan semangat dan dorongan dalam penyelesaian skripsi ini.
7. Semua teman-teman dan sahabat yang selalu setia memberikan motivasi dan setia membantu selama penyusunan skripsi ini.
8. Serta semua pihak yang tidak dapat penulis sebutkan satu persatu yang turut membantu dan mendukung kelancaran penyusunan 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, baik dalam teknik penulisan, pembahasan, dan penyajian, untuk itu penulisa terbuka untuk menerima kritik dan saran yang membangun dari pembaca untuk kesempurnaan dari skripsi ini.

**Mataram, Agustus 2018**

**Penulis,**

# 



# **IZIN PENGGUNAAN**

Skripsi ini merupakan syarat kelulusan pada Program Studi S1 Teknik Informatika STMIK Bumigora Mataram, dengan ini penulis setuju jika skripsi ini digandakan (diduplikasi) baik sebagian maupun seluruhnya, ataupun dikembangkan untuk kepentingan akademis yang disetujui oleh pembimbing penulis, Pembantu Ketua I atau Ketua STMIK Bumigora*.*

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

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

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

Pembantu Ketua I

Ketua Program Studi S1 Teknik Informatika

STMIK Bumigora Mataram

# **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 *DKIM*, *SPF*, *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 *DKIM*, *SPF*, *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 *DKIM*, *SPF*, *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 *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*.

**Kata Kunci**: *DomainKeys Identified Mail* (*DKIM*), *Sender Policy Framework* (*SPF*), *SpamAssassin*, *ClamAV*, *Amavisd-New*, *Mail Server*

# **DAFTAR ISI**

**Halaman**

**HALAMAN SAMPUL**

**HALAMAN JUDUL**

**HALAMAN PENGESAHAN**

[KATA PENGANTAR i](#_Toc518896062)

[LEMBAR PERNYATAAN KEASLIAN ii](#_Toc518896063)

[IZIN PENGGUNAAN iv](#_Toc518896064)

[ABSTRAK v](#_Toc518896065)

[DAFTAR ISI vi](#_Toc518896066)

[DAFTAR GAMBAR ix](#_Toc518896067)

[DAFTAR TABEL xii](#_Toc518896068)

[DAFTAR LAMPIRAN xiii](#_Toc518896069)

[BAB I](#_Toc518896070) [PENDAHULUAN 1](#_Toc518896071)

[1.1. Latar Belakang 1](#_Toc518896072)

[1.2. Perumusan Masalah 3](#_Toc518896073)

[1.3. Batasan Masalah 3](#_Toc518896074)

[1.4. Tujuan dan Manfaat Penulisan 5](#_Toc518896075)

[1.4.1. Tujuan 5](#_Toc518896076)

[1.4.2. Manfaat 5](#_Toc518896077)

[1.5. Metodologi penelitian 6](#_Toc518896078)

[1.6. Sistematika Penulisan 8](#_Toc518896079)

[BAB II LANDASAN TEORI 10](#_Toc518896080)

[2.1. Jaringan Komputer 10](#_Toc518896081)

[2.2. Model Lapisan *OSI* 11](#_Toc518896082)

[2.3. *Transmission Control Protocol / Internet Protocol (TCP/IP)* 16](#_Toc518896083)

[2.4. Keamanan Jaringan Komputer 17](#_Toc518896084)

[2.5. Jenis-jenis Layanan Keamanan Jaringan 18](#_Toc518896085)

[2.6. *Email Spoofing* dan *Phising* 19](#_Toc518896086)

[2.7. *Server* 20](#_Toc518896087)

[2.8. Linux 20](#_Toc518896088)

**Halaman**

[2.9. Linux *CentOS* 21](#_Toc518896089)

[2.10. *Centos Web Panel* 22](#_Toc518896090)

[2.11. Surat Elektronik 22](#_Toc518896091)

[2.12. *Mail Server* 23](#_Toc518896092)

[2.13. *Mail Protocol* 25](#_Toc518896093)

[*2.14.* *Postfix* 26](#_Toc518896094)

[2.15. Dovecot 26](#_Toc518896095)

[2.16. *Roundcube* 27](#_Toc518896096)

[2.17. *Domain Name System (DNS)* 27](#_Toc518896097)

[2.18. *DNS Server* 27](#_Toc518896098)

[2.19. *Bind9* 28](#_Toc518896099)

[2.20. *HTTP* 28](#_Toc518896100)

[2.21. *HTTP Server* 29](#_Toc518896101)

[2.22. Apache HTTP Server 29](#_Toc518896102)

[2.23. *Email Spam* 29](#_Toc518896103)

[2.24. *Spam Filter* 30](#_Toc518896104)

[2.27. *SpamAssassin, ClamAV, dan Amavisd-New* 30](#_Toc518896105)

[2.28. *DomainKeys Identified Mail (DKIM)* dan *OpenDKIM* 32](#_Toc518896106)

[*2.29.* *Sender Policy Framework (SPF)* 33](#_Toc518896107)

[2.30. *Gmail* 35](#_Toc518896108)

[2.31. *Emkei’s* *Mailer* 35](#_Toc518896109)

[2.32. *Yahoo! Mail* 35](#_Toc518896110)

[BAB III](#_Toc518896111) [METODOLOGI DAN PERANCANGAN 36](#_Toc518896112)

[3.1. Tahap Analisa (*Analysis*) 36](#_Toc518896113)

[3.1.1. Pengumpulan Data 36](#_Toc518896114)

[3.1.2. Analisa Data 37](#_Toc518896115)

[3.2. Tahap Desain *(Design)* 39](#_Toc518896116)

[3.2.1 Rancangan Sistem *Filtering Email Spam, Virus* dan *Spoofing* 39](#_Toc518896117)

[3.2.2 Rancangan Jaringan Uji Coba 41](#_Toc518896118)

[3.2.3 Rancangan Pengalamatan *IP* 42](#_Toc518896119)

[3.2.4 Rancangan Akun *Email* 42](#_Toc518896120)

**Halaman**

[3.2.5 Kebutuhan Perangkat Keras dan Perangkat Lunak 43](#_Toc518896121)

[3.3. Tahap Simulasi (*Prototyping*) 44](#_Toc518896122)

[3.3.1. Instalasi Dan Konfigurasi 45](#_Toc518896123)

[3.3.2. Ujicoba 45](#_Toc518896124)

[3.4. Tahap Implementasi 46](#_Toc518896125)

[3.5. Tahap *Monitoring* 46](#_Toc518896126)

[BAB IV HASIL DAN PEMBAHASAN 48](#_Toc518896127)

[4.1. Hasil Instalasi Dan Konfigurasi 48](#_Toc518896128)

[4.1.1. Hasil Instalasi Dan Konfigurasi *Server* 48](#_Toc518896129)

[4.1.2. Hasil Konfigurasi *Client* 58](#_Toc518896130)

[4.2. Hasil Uji Coba 58](#_Toc518896131)

[4.2.1. Verifikasi Konfigurasi 58](#_Toc518896132)

[4.2.2. Skenario Uji Coba 62](#_Toc518896133)

[4.3. Analisa Hasil Uji Coba 105](#_Toc518896134)

[4.3.1. Analisa Hasil Uji Coba Pengiriman *Email Spoofing* 105](#_Toc518896135)

[4.3.2 Analisa Hasil Uji Coba Pengiriman *Email Spam* 106](#_Toc518896136)

[4.3.3 Analisa Hasil Uji Coba Mengirim *Email* Mengandung *Virus* 107](#_Toc518896137)

[4.3.4 Analisa Hasil Uji Coba Pengecekan *Header Email* 108](#_Toc518896138)

[BAB V PENUTUP 110](#_Toc518896139)

[5.1. Kesimpulan 110](#_Toc518896140)

[5.2. Saran 110](#_Toc518896141)

[DAFTAR REFERENSI 1](#_Toc518896142)

**LAMPIRAN**

# **DAFTAR GAMBAR**

**Halaman**

[Gambar 1.1 Fase *NDLC* 6](#_Toc519143570)

[Gambar 2.1 *Layer OSI* 11](#_Toc518829686)

[Gambar 2.2 Perbandingan *Layer TCP/IP* dan *Layer OSI* 16](#_Toc518829687)

[Gambar 2.3 Proses pengiriman *email* 24](#_Toc518829688)

[Gambar 2.4 Cara Kerja *SpamAssassin*, *ClamAV*, dan *Amavisd-New* 32](#_Toc518829689)

[Gambar 2.5 Cara Kerja *DKIM* 33](#_Toc518829690)

[Gambar 2.6 Cara Kerja *SPF* 34](#_Toc518829691)

[Gambar 3.1 Rancangan Sistem *Filtering Email Spam,* *Spoofing,* dan *Virus* 39](#_Toc518829682)

[Gambar 3.2 Rancangan Topologi Uji Coba 41](#_Toc518829683)

[Gambar 4.1 Linux CentOS release 7.3.1611 48](#_Toc521091290)

[Gambar 4.2 Hasil Instalasi CWP 49](#_Toc521091291)

[Gambar 4.3 Konfigurasi Interface 50](#_Toc521091292)

[Gambar 4.4 Konfigurasi Name Server 51](#_Toc521091293)

[Gambar 4.5 Konfigurasi *Domain* 51](#_Toc521091294)

[Gambar 4.6 *File* skripsian.online.db 52](#_Toc521091295)

[Gambar 4. 7 *File* named.ip4.skripsian.online.db 53](#_Toc521091296)

[Gambar 4.8 *File* named.conf 53](#_Toc521091297)

[Gambar 4.9 Membuat Akun *Email* 54](#_Toc521091298)

[Gambar 4.10 Instalasi *DKIM*, *SPF, Anti Spam,* dan *Anti Virus* 55](#_Toc521091299)

[Gambar 4.11 Menambah *DKIM Record* pada *File Zone* 55](#_Toc521091300)

[Gambar 4.12 Menambah *SPF Record* pada *File Zone* 56](#_Toc521091301)

[Gambar 4.13 *DKIM* dan *SPF Record* 56](#_Toc521091302)

[Gambar 4.14 Konfigurasi *File* TrustedHosts 57](#_Toc521091303)

[Gambar 4.15 Konfigurasi *File* main.cf 57](#_Toc521091304)

[Gambar 4.16 Terhubung ke *Internet* 58](#_Toc521091305)

[Gambar 4.17 Verifikasi Konfigurasi *DNS Server* 59](#_Toc521091306)

[Gambar 4.18 Verifikasi Konfigurasi *Mail Server* 60](#_Toc521091307)

[Gambar 4.19 Mengirim *Email* pada *User Email Local* 60](#_Toc521091308)

[Gambar 4.20 Mengirim *Email* Pada *Mail Server* Lain 60](#_Toc521091309)

[Gambar 4.21 Verifikasi Fungsi *DKIM*, *SPF,* dan *SpamAssassin* 61](#_Toc521091310)

[Gambar 4.22 Verifikasi Fungsi *ClamAV* 61](#_Toc521091311)

[Gambar 4.23 Ping *Mail Server* 62](#_Toc521091312)

[Gambar 4.24 Akses *MUA* 62](#_Toc521091313)

[Gambar 4.25 *Emkei’s Fake Mailer* 65](#_Toc521091314)

[Gambar 4.26 *Email Spoofing* 66](#_Toc521091315)

[Gambar 4.27 Balasan *Email Spoofing* 67](#_Toc521091316)

[Gambar 4. 28 Mengirim *Email Spoofing* ke *Yahoo! Mail* 68](#_Toc521091317)

[Gambar 4.29 *Email Spoofing* Terkirim ke *Yahoo! Mail* 68](#_Toc521091318)

[Gambar 4. 30 Mengirim *Email Spoofing* ke skripsian.online 69](#_Toc521091319)

**Halaman**

[Gambar 4. 31 *Email Spoofing* Terkirim ke *User* skripsian.online 70](#_Toc521091320)

[Gambar 4.32 Mengirim *Email Spam* dari skripsian.online 71](#_Toc521091321)

[Gambar 4.33 *Email Spam* dari sripsian.online Terkirim 71](#_Toc521091322)

[Gambar 4.34 Mengirim *Email Spam* dari *Yahoo! Mail* 72](#_Toc521091323)

[Gambar 4.35 *Email Spam* dari *Yahoo! Mail* Terkirim 72](#_Toc521091324)

[Gambar 4.36 Mengirim *Email Spam* dari *Gmail* 73](#_Toc521091325)

[Gambar 4.37 *Email Spam* dari *Gmail* Terkirim 73](#_Toc521091326)

[Gambar 4.38 Mengirim *Email Spam* Tanpa *GTUBE TEST* 74](#_Toc521091327)

[Gambar 4.39 *Email* Terindikasi Sebagai *Spam* oleh *Yahoo! Mail* 75](#_Toc521091328)

[Gambar 4.40 Mengirim *Email Spam* Tanpa *Format GTUBE Test* 76](#_Toc521091329)

[Gambar 4. 41 *Email Spam* Tanpa *GTUBE Test* Terkirim 76](#_Toc521091330)

[Gambar 4. 42 *EICAR Test* dari skripsian.online 77](#_Toc521091331)

[Gambar 4.43 Email Mengandung Virus dari skripsian.online Terkirim 78](#_Toc521091332)

[Gambar 4.44 *EICAR Test* dari *Yahoo! Mail* 78](#_Toc521091333)

[Gambar 4. 45 *Email* Mengandung *Virus* dari *Yahoo! Mail* Terkirim 79](#_Toc521091334)

[Gambar 4.46 *EICAR Test* dari *Gmail* 80](#_Toc521091335)

[Gambar 4.47 *Email* Mengandung *Virus* dari *Gmail* Terkirim 80](#_Toc521091336)

[Gambar 4.48 Lampiran Terdeteksi Sebagai *Virus* 81](#_Toc521091337)

[Gambar 4.49 Mengirim *Virus* Sebelum Penerapan *Anti Virus* 82](#_Toc521091338)

[Gambar 4.50 *Virus* Berhasil Terkirim ke Kotak Masuk Pengguna *Email* 82](#_Toc521091339)

[Gambar 4.51 *Header Email* Tanpa Parameter *X-Virus-Scanned* 82](#_Toc521091340)

[Gambar 4.52 Cuplikan *Header Email* pada Gmail Sebelum Penerapan 83](#_Toc521091341)

[Gambar 4.53 Cuplikan *Header Email* pada *Yahoo! Mail* Sebelum Penerapan 84](#_Toc521091342)

[Gambar 4.54 Cuplikan *Header Email* pada skripsian Sebelum Penerapan 85](#_Toc521091343)

[Gambar 4.55 *Private Key* pada skripsian.online 86](#_Toc521091344)

[Gambar 4.56 *Public Key* pada *DNS Server* skripsian.online 87](#_Toc521091345)

[Gambar 4.57 Cuplikan *Header Email* 87](#_Toc521091346)

[Gambar 4.58 *SPF Record* pada skripsian.online 88](#_Toc521091347)

[Gambar 4.59 Parameter *Received-SPF* pada *Header Email* 88](#_Toc521091348)

[Gambar 4.60 Pemberitahuan dari *Mail System* *Emkei’s Fake Mailer* 88](#_Toc521091349)

[Gambar 4.61 *Email Spoofing* Masuk ke *Folder Spam* 89](#_Toc521091350)

[Gambar 4.62 *Email Spoofing* Masuk pada *Folder Inbox* 90](#_Toc521091351)

[Gambar 4.63 Mengirim *Email Spam* dari skripsian.online Setelah Penerapan 91](#_Toc521091352)

[Gambar 4.64 *Email* dari skripsian.online Terindikasi *Spam* 91](#_Toc521091353)

[Gambar 4.65 Mengirim *Email Spam* dari *Yahoo! Mail* Setelah Penerapan 92](#_Toc521091354)

[Gambar 4.66 *Email* dari *Yahoo! Mail* Terindikasi *Spam* 92](#_Toc521091355)

[Gambar 4.67 Mengirim *Email Spam* dari *Gmail* Setalah Penerapan 93](#_Toc521091356)

[Gambar 4.68 *Email* dari *Gmail* Terindikasi *Spam* 93](#_Toc521091357)

[Gambar 4.69 *Email* Dengan Format *Spam* 94](#_Toc521091358)

[Gambar 4.70 *Email* Terindikasi Sebagai *Spam* oleh *Yahoo! Mail* 95](#_Toc521091359)

[Gambar 4.71 Mengirim *Email Spam* Tanpa *Format GTUBE Test* 96](#_Toc521091360)

[Gambar 4. 72 *Default Score SpamAssassin* 96](#_Toc521091361)

**Halaman**

[Gambar 4.73 Skor *Email* yang Terindikasi Sebagai *Spam* 97](#_Toc521091362)

[Gambar 4.74 *EICAR Test* dari skripsian.online Setelah Penerapan 97](#_Toc521091363)

[Gambar 4.75 *Email* dari skripsian.online Terblok 98](#_Toc521091364)

[Gambar 4.76 *EICAR Test* dari *Yahoo! Mail* Setelah Penerapan 99](#_Toc521091365)

[Gambar 4.77 *Email* dari *Yahoo! Mail* Terblok 99](#_Toc521091366)

[Gambar 4.78 *EICAR Test* dari *Gmail* Setelah Penerapan 100](#_Toc521091367)

[Gambar 4.79 *Email* dari *Gmail* Terblok 101](#_Toc521091368)

[Gambar 4.80 Lampiran Terdeteksi Sebagai *Virus* 101](#_Toc521091369)

[Gambar 4.81 Mengirim *Virus* Setelah Penerapan *Anti Virus* 102](#_Toc521091370)

[Gambar 4.82 *Email* Terdeteksi Mengandung *Virus* 102](#_Toc521091371)

[Gambar 4.83 Cuplikan *Header Email* pada *Gmail* Setelah Penerapan 103](#_Toc521091372)

[Gambar 4. 84 Cuplikan *Header Email* pada *Yahoo! Mail* Setelah Penerapan 104](#_Toc521091373)

[Gambar 4.85 Cuplikan *Header Email* pada skripsian.onlineSetelah Penerapan 105](#_Toc521091374)

# **DAFTAR TABEL**

**Halaman**

[Tabel 3.1 Jurnal Ilmiah Tentang *Email Spam, Spoofing,* dan *Virus* 36](#_Toc518829915)

[Tabel 3.2 Pengalamatan *IP* 42](#_Toc518829916)

[Tabel 3.3 Kebutuhan Akun *Email* 42](#_Toc518829917)

[Tabel 3.4 Spesifikasi *VPS* 43](#_Toc518829918)

[Tabel 3.5 Spesifikasi *Client* 43](#_Toc518829919)

[Tabel 4.1 Perbandingan Sebelum dan Setelah Penerapan *DKIM* dan *SPF* 106](#_Toc518829901)

[Tabel 4.2 Perbandingan Sebelum dan Setelah Penerapan *Anti Spam* 107](#_Toc518829902)

[Tabel 4.3 Perbandingan Sebelum Penerapan *Anti virus* 107](#_Toc518829903)

[Tabel 4.4 Perbandingan Header Email Sebelum dan Setelah Penerapan 108](#_Toc518829904)

# **DAFTAR LAMPIRAN**

**Halaman**

LAMPIRAN A: *HEADER EMAIL* ......................................................................... A.1

# 

# **BAB I**

# **PENDAHULUAN**

## **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.

Hoiriyah, Sugiantoro, dan Prayudi (2016) 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. *Spam* adalah e*mail* yang tidak diinginkan, *email spam* dikirim kepadaseseorang penerima dan pesantersebut tidak ada gunanya untuk penerima. *Spam* dikirim pada jaringan untuk meningkatkan konsumsisumber daya, dengan kata lain untuk meningkatkanlalu lintas jaringan (Suryana, Akbar, Widiyasono, 2016). Nurlina & Irmayana (2014) menyebutkan bahwa tidak semua *email spam* masuk pada *folder spam* yang telah disediakan dan *email* yang bukan *spam* terkadang masuk pada *folder spam* sehingga *email* penting terkadang tidak dibaca oleh penerima *email*. *Email* juga merupakan sumber utama dari kebanyakan aktivitas kriminal pada *internet*, salah satu ancaman dari tindak kejahatan yang menggunakan *email* adalah emailspoofing. *Email spoofing* dianggap sebagai tindakanberbahaya karena melakukan manipulasi data pada *header email* untuk menyamar sebagai orang atau organisasiyang sah, contohnya seperti melakukan pengiriman *email* dengan nama pengirim seolah dari administrator suatuorganisasi. Pengirim email spoofing menyerang denganberbagai macam isi pesan untuk membuat korbannya percaya.

*Email spam,* *spoofing*, dan *virus* sangat tidak diinginkan oleh pengguna maupun penyedia layanan *email* sehingga perlu diterapkan suatu sistem yang dapat mencegah *email spam, spoofing,* dan *virus*. Sistem pencegahan *email spam, spoofing,* dan *virus* diharapkan dapat mengurangi dampak kerugian yang diakibatkan oleh *email spam, spoofing,* dan *virus*.

*DKIM* dan *SPF* dapat digunakan untuk melakukan otentikasi dan otorisasi *email* sehingga akun *email client* akan terbebas dari tindakan *spoofing*. Penerapan *Anti Spam* dan *Anti Virus* juga diperlukan agar *email server* terhindar dari *email spam* dan *virus*, metode yang diterapkan oleh *Anti Spam* dan *Anti Virus* yaitu dengan melakukan pengecekan *haeder*, *body*, dan *attachment* *email*.

Manfaat dari penerapan *DKIM, SPF*, *Anti Spam* dan *Anti Virus* adalah untuk menghemat sumber daya *mail server* dengan cara memblokir surat elektronik yang dianggap sebagai *spam*, meningkatkan kualitas keamanan surat elektronik sehingga pengguna dapat terhindar dari aktifitas *spoofing* serta *virus* dan *malware* yang disisipkan melalui surat elektronik.

## **Perumusan Masalah**

Adapun rumusan masalah pada skripsi ini adalah bagaimana menganalisa penerapan *DKIM*, *SPF*, *Anti Spam,* dan *Anti Virus* pada *mail server* agar *mail server* dapat terhindar dari *email spam, virus* dan pengguna *email* dapat terhindar dari aktifitas *spoofing*.

## **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 (*Roundcube*).
2. Sistem operasi *VPS* yang digunakan adalah *CentOS* 7.3.1611.
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* skripsian.online.
8. Aplikasi *HTTP server* yang digunakan adalah Apache agar *Mail Transfer Agent* berbasis web dapat diakses melalui *browser*.
9. Aplikasi yang digunakan untuk membuat *DKIM record key* adalah *OpenDKIM*.
10. Validasi surat elektronik menggunakan *SpamAssassin, ClamAV*, dan *Amavisd-New* sebagai *anti spam* dan *anti virus*.
11. *Email* yang digunakan untuk uji coba adalah *email server* skripsian.online, *Emkei’s Fake Mailer*, *Yahoo! Mail* *Gmail*, dan ridho.org.
12. Pengujian yang dilakukan dengan mengirim surat elektronik yang terindikasi sebagai *spam* dan *ham*, kemudian melakukan pengiriman *email* *spoofing,* dan email yang mengandung virus, serta mengecek header email sebelum dan setelah penerapan *DKIM*, *SPF*, *anti spam,* dan *anti virus*.
13. Pengujian *DKIM* dan *SPF* dilakukan dengan cara mengirim *email spoofing* menggunakan *Emkei’s Fake Mailer* kemudian *email spoofing* tersebut dikirim ke *Gmail* dan *Yahoo! Mail*.
14. Pengujian *Anti Spam* dan *Anti Virus* dilakukan dengan cara mengirim *email spam* dan *email* yang mengandung *virus* ke *mail server* skripsian.online.

## **Tujuan dan Manfaat Penulisan**

### **Tujuan**

Adapun tujuan dari penulisan skripsi ini adalah untuk menganalisa penerapan *DKIM*, *SPF*, *Anti Spam* dan *Anti Virus* pada *mail server* agar *mail server* terhindar dari *email spam*, *virus* dan aktifitas *spoofing* yang dilakukan oleh pihak yang tidak bertanggung jawab.

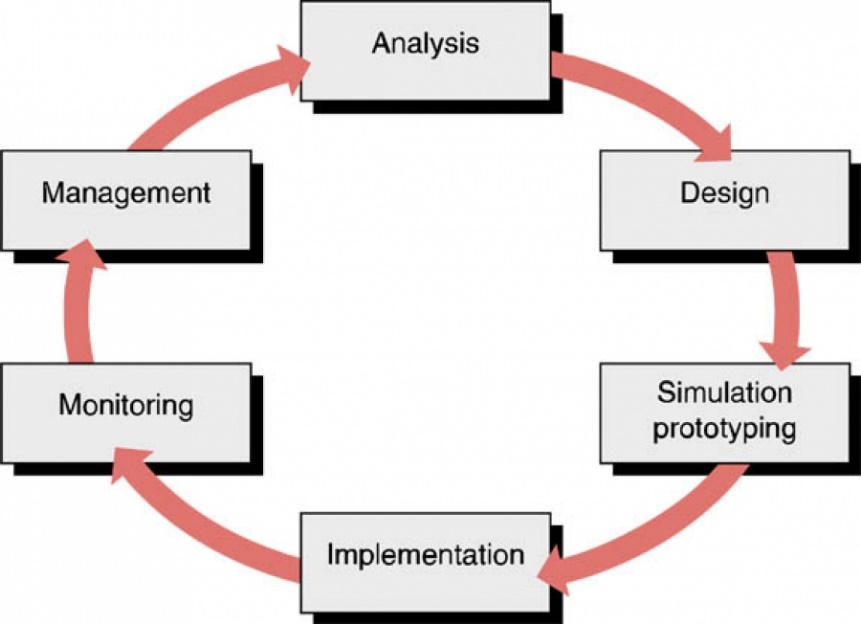
### **Manfaat**

Adapun manfaat dari penelitian ini adalah sebagai berikut:

1. Bagi Diri Sendiri
2. Dapat menambah ilmu pengetahuan penulis sehingga dapat diterapkan pada dunia kerja.
3. Dapat menjadi tempat untuk mengimplementasikan ilmu pengetahuan yang telah didapat selama berada dibangku perkuliahan.
4. Sebagai syarat untuk menyelesaikan jenjang pendidikan Strata 1 (S1) pada program studi Teknik Informatika di STMIK Bumigora.
5. Bagi Keilmuan
6. Dapat menjadi bahan rujukan untuk pengembangan penelitian berikutnya terutama dalam bidang yang sama.
7. Dapat menjadi sarana untuk melatih kemampuan dalam menulis karya ilmiah.
8. Bagi Masyarakat
9. Dapat memberikan pengetahuan terkait dengan analisa penanganan *email spam, virus* dan aktifitas *spoofing* menggunakan *DKIM*, *SPF*, *Anti Spam,* dan *Anti Virus*.
10. Dapat memberikan solusi cara penerapan *DKIM*, *SPF*, *Anti Spam,* dan *Anti Virus* pada *mail server*.

## **Metodologi penelitian**

Nurfajar, Kurniawan, dan Yunan (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 lima 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 *DKIM*, *SPF*, *Anti Spam*, dan *Anti Virus*. Data-data yang telah terkumpul kemudian dianalisa.

1. *Design*

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

1. *Simulation Prototyping*

Setelah melakukan analisa dan desain, tahap berikutnya adalah melakukan simulasi dan membuat *prototype* berdasarkan pada desain yang telah dirancang sebelumnya (Nurfajar, Kurniawan, dan Yunan, 2015). Pada fase ini dilakukan instalasi dan konfigurasi serta uji coba *DKIM*, *SPF* *Anti Spam*, dan *Anti Virus* menggunakan berbagai macam skenario.

1. *Implementation*

Pada fase ini penulis akan menerapkan semua yang telah direncanakan dan di desain pada tahapan sebelumnya. Pada fase ini penulis akan membangun sebuah *mail server* kemudian pada *mail server* tersebut akan diterapkan *DKIM*, *SPF*, *Anti Spam*, dan *Anti Virus* untuk mengotentikasi, mengotorisasi dan memvalidasi *email* dan penulis akan melakukan analisa pada *mail server* sebelum dan sesudah penerapan *DKIM*, *SPF*, *Anti Spam*, dan *Anti Virus*.

1. *Monitoring*

Setelah melakukan implementasi, tahapan *monitoring* adalah tahapan penting dalam merancang desain jaringan, tujuan dari tahapan *monitoring* adalah untuk memastikan jaringan komputer berjalan sesuai dengan tujuan pada tahap analisis (Nurfajar, Kurniawan, dan Yunan, 2015). Pada fase ini penulis akan melakukan *monitoring* terhadap aktifitas *spam*, dan *virus* pada *mail server* skripsian.online dengan menggunakan *maillog server* skripsian.online serta melakukan *monitoring* terhadap aktifitas *spoofing* pada penyedia layanan *email* yaitu *Yahoo! Mail* dan *Gmail*.

## **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*, *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**

## **Jaringan Komputer**

Menurut Haryanto & 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. Sedangkan menurut Masero, Triyono, dan Andayati (2013) 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 (Ardiantoro, Triyono, Fatkhiyah, 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.

## **Model Lapisan *OSI***

Model *Open System Interconnection* (*OSI*) diciptakan oleh *International Organitation 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 (Kader, Najoan, dan 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 pada proses komunikasi data:

1. *Physical*

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

1. *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*.

1. *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 mengalamai masalah pada *host*nya.

* *IGMP (Internet Group Message Protocol)*

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

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

1. *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*.

1. *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.

1. *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 di*download* kedalam jaringan lokal.

* *MIME (*Multipurpose Internet Mail Extension*)*

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

* *SMB (*Server Messange 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*.

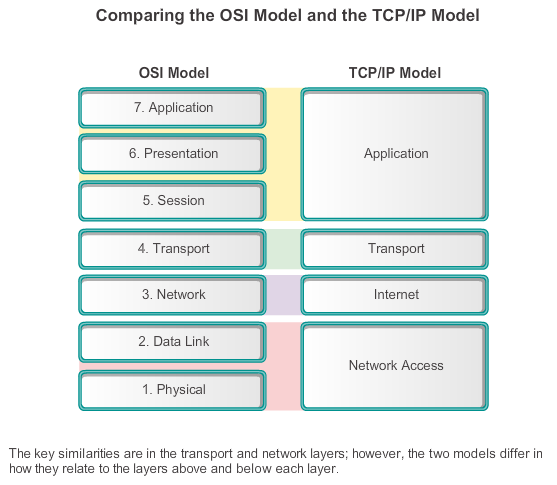
* *DHCP (Dynamic Host Configuration Protocol)*

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

## ***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* (“*deeohdee*”) karena mereka pada dasarnya dikembangkan oleh komunitas riset *Advanced Research Projects Agency* (*ARPA*) dari *US Department of Defense* (*DoD*).

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* (Riadi, 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 untukproses mengirim dan menerima data*.*

1. *Internet Layer*

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

1. *Transport Layer*

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

1. *Application Layer*

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

## **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.

## **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 kemanan 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.

1. Kendali Akses (*Access Control*)

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

1. Kerahasiaan Data (*Data Confidentility*)

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

1. 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.

1. *Non-Repudiation*

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

1. Ketersediaan (*Availability*)

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

## ***Email Spoofing* dan *Phising***

Menurut Suryana, Akbar, dan 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.

Menurut Suryana, Akbar, dan 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.

## ***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).

## **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.

## **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 menggunakannya (Wicitra, Utomo, dan Wardana, 2014).

## ***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 (Control Web Panel, n.d.).

## **Surat Elektronik**

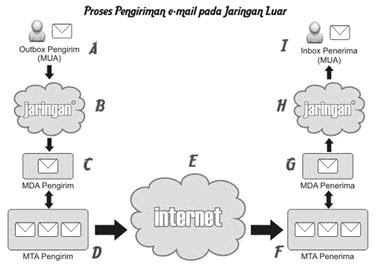
Surat elektronikadalah layanan yang diberikan oleh *internet* yang berkembang sejak tahun 1960, pada saat itu *internet* belum terbentuk, yang ada hanyalah kumpulan *mainframe* yang terbentuk sebagai jaringan. Mulai tahun 1980-an, surat elektroniksudah bisa dinikmati oleh khalayak umum. Surat elektronikadalah salah satu proses pengiriman surat melalui *internet* dengan menggunakan waktu yang sangat singkat.Surat elektronikmerupakan salah satu dari sekian banyak layananinternet 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 olehinternet, dari layanan internet tersebut yang paling banyak digunakan adalahlayanan 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).

## ***Mail Server***

Menurut Desmira, Sumarto, dan 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-orangdalam satu domain) dan jarak jauh pengirim danmeneruskan *email* keluar untuk pengiriman. Sebuahkomputer yang didedikasikan untuk menjalankanaplikasi tersebut juga disebut sebagai *mail server. Mail Server* bisa diartikan sebagai induk atau rumahdari *email,* Setiap *email* yang dikirimkan dibuat untukmelewati sejumlah *server mail* sepanjang perjalanan kepenerima. Untuk *user* biasa, surat tersebut dikirimlangsung tetapi proses adalah sesuatu yangdimengerti. Tanpa rangkaian *Server Mail*, penggunahanya akan dapat mengirim email ke orang-orangyang memiliki alamat *email* dengan *domain* yang sama.

Menurut Muarif & Irwan (2017) *Mail Server* memiliki tiga komponen utama yangmembentuknya, 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 satukomponen 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 biasadigunakan untuk pertimbangan. Tiap-tiap program *mail server* memiliki kelebihan dan kekurangantersendiri. Beberapa *MTA* memiliki fasilitas yangsangat hebat sehingga mampu digunakan untukmenangani email dalam jumlah ratusan bahkansampai ribuan perhari (Desmira, Sumarto, 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).

## ***Mail Protocol***

Menurut Desmira, Sumarto, dan 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.

1. *IMAP (Internet Message Access Protocol)*

*IMAP (Internet Message Access Protocol)* samahalnya dengan *POP3*, maka pesan *email* akansepenuhnya disimpan dalam *server email* danmenggunakan komputer lokal untuk mengirim danmengambilnya kapanpun di inginkan. Tergantung darikeinginan *user. IMAP* adalah *protocol* standar untuk mengakses ataumengambil *email* dari *server*.

1. *SMTP (Simple Mail Transfer Protocol)*

*SMTP* merupakan salah satu jenis *protocol* yangbekerja dalam hal pengiriman pesan-pesan berupa suratelektronik atau *email* pada sebuah jaringan *internet.*

## ***Postfix***

Menurut Hidayat (2010) “Postfix adalah *Mail Tranfer 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*.

## **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*.

## ***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 me*render* 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.).

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

Menurut Saputra & 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*.

## ***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.

## ***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 di tulis pada awal 1980 dan didanai oleh *DARPA* (Defense AdvancedResearch Projects Agency). Pada pertengahan 1980-an, *DEC* (Digital EquipmentCorporation) mengambil alih pengembangan *BIND9*. Satu dari pekerja itu adalah Paul Vixie, yang terus mengerjakan *BIND9* sesudah meninggalkan *DEC* (Hidayat, 2010).

## ***HTTP***

Menurut Zabar dan 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* dikoordinasi 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.

## ***HTTP Server***

Menurut Syafrizal & Saputra (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*.

## **Apache HTTP Server**

Menurut Syafrizal dan Saputra (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.

## ***Email Spam***

Menurut Chandra, Indrawan, dan 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 *bandwith* 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.

## ***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).

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

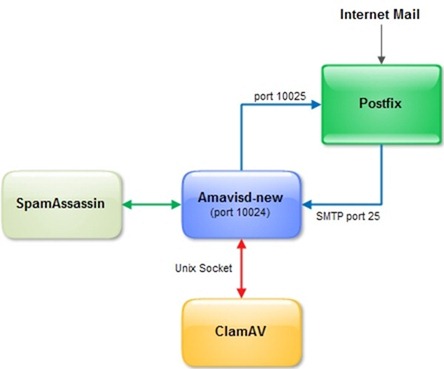
Menurut Irmayana & Nurlina (2014) *SpamAssassin* adalah aplikasi yang sudah teruji secara luas menggunakan proyek *open source* yang berfungsi sebagai *mail filter* untuk mendeteksi *spam*. *SpamAssassin* berjalan pada *server* dan sebagai *filter spam* sebelum sampai pada pada kotak masuk pengguna. *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 *LMTP*, atau dengan menggunakan program pembantu (Martinec, 2016).



Gambar 2.4 Cara Kerja *SpamAssassin*, *ClamAV*, dan *Amavisd-New*

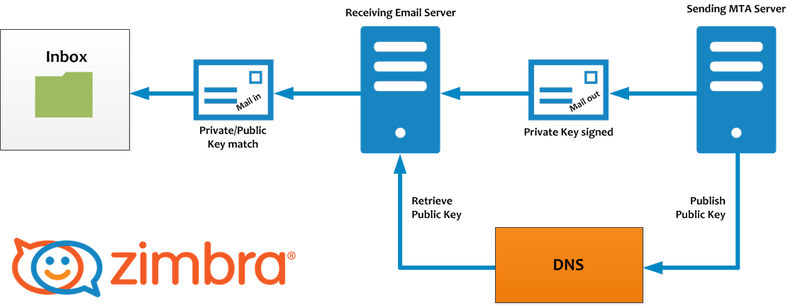
Sumber: [Valsecchi](https://nolabnoparty.com/en/author/paolo-valsecchi/), 2013

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

## ***DomainKeys Identified Mail (DKIM)* dan *OpenDKIM***

*Domain Keys Identified Mail (DKIM)* adalah metode otentikasi *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*.



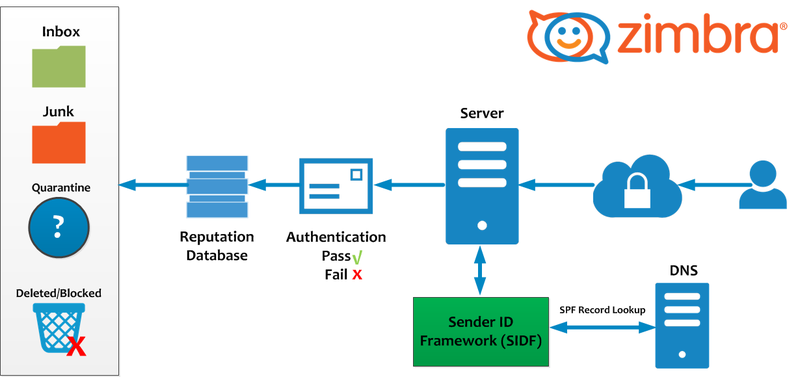
Gambar 2.5 Cara Kerja *DKIM*

Sumber: *Zimbra Incorporation*, 2005

Cara kerja *DKIM* dapat dilihat seperti pada gambar 2.5 yaitu *mail server* pengirim mem*publish 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*.

## ***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.

## ***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).

## ***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*.

## ***Yahoo! Mail***

*Yahoo! Mail* merupakan sebuah penyedia surat elektronik (*webmail*) dari [Yahoo!](https://id.wikipedia.org/wiki/Yahoo!). *Yahoo! Mail* merupakan penyedia surat elektronik terbesar di [*internet*](https://id.wikipedia.org/wiki/Internet) dengan jutaan pengguna. Saingan utama *Yahoo! Mail* ialah [*Windows Live Hotmail*](https://id.wikipedia.org/wiki/Windows_Live_Hotmail), [*Gmail*](https://id.wikipedia.org/wiki/Gmail) dan [*AOL Mail*](https://id.wikipedia.org/wiki/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 5 tahapan yaitu *Analysis, Design, Simulation Prototyping,* *Implementation,* dan *Monitoring.*

## **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.

### **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 *Email Spoofing* dengan Metode *Digital Forensics Research Workshop (DFRWS)* | Mengidentifikasi *email* *spoofing* menggunakan metode *DFRWS* |
| 2 | Hoiriyah, Bambang Sugiantoro, dan Yudi Prayudi | 2016 | Investigasi Forensik pada E-mail Spoofing menggunakan MetodeHeader Analysis | Membuat algoritma untuk membandingkan *header email spoofing* dan *email legitimate* |
| 3 | Nurlina dan Irmayana | 2014 | Studi Banding *Spam-Assassin* *Mail Server* Dengan dan Tanpa *Filter* di Sisi *Mail Client* | Membandingkan metode pengklasifikasian *email spam* dengan dan tanpa *filter* di sisi *client* menggunakan metode *naive bayes* dan *spamassassin* |
| 4 | Wirawan Nathaniel Chandra, Gede Indrawan, dan I Nyoman Sukajaya | 2016 | *Spam Filtering* Dengan Metode *Pos Tagger* dan Klasifikasi *Naive Bayes* | Memfilter kata penting pada email menggunakan metode Pos Tagger untuk dijadikan pembelajaran pada klasifikasi Naive Bayes |
| 5 | Ratih Yulia Hayuningtyas | 2017 | Aplikasi *Filtering of Spam Email* Menggunakan *Naive Bayes* | Membuat aplikasi *Mail User Agent* yang dapat mem*filter spam* dengan metode *Naive Bayes* |

### **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 membahas tentang investigasi *email spoofing* menggunakan metode *haeder analysis* untuk menemukan pola *haeder email spoofing* kemudian membuat algoritma untuk mengklasifikasikan *email spoofing* dan *email legitimate*.
3. Jurnal ilmiah ketiga membahas tentang penerapan algoritma *naive bayes* disisi *client* yang akan meningkatkan akurasi pendeteksian *email spam* sebesar 99,98%.
4. Jurnal ilmiah keempat membahas tentang pemfilteran kata penting pada email menggunakan metode Pos Tagger untuk dijadikan pembelajaran pada klasifikasi Naive Bayes.
5. Jurnal ilmiah kelima membahas tentang pembuatan aplikasi *Mail User Agent* yang dapat mem*filter spam* dengan metode *Naive Bayes*.
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 *Domainkeys identified mail (DKIM), sender policy framework (SPF) Anti Spam*, dan *Anti Virus* pada *Mail Server.*

## **Tahap Desain *(Design)***

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

### **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*

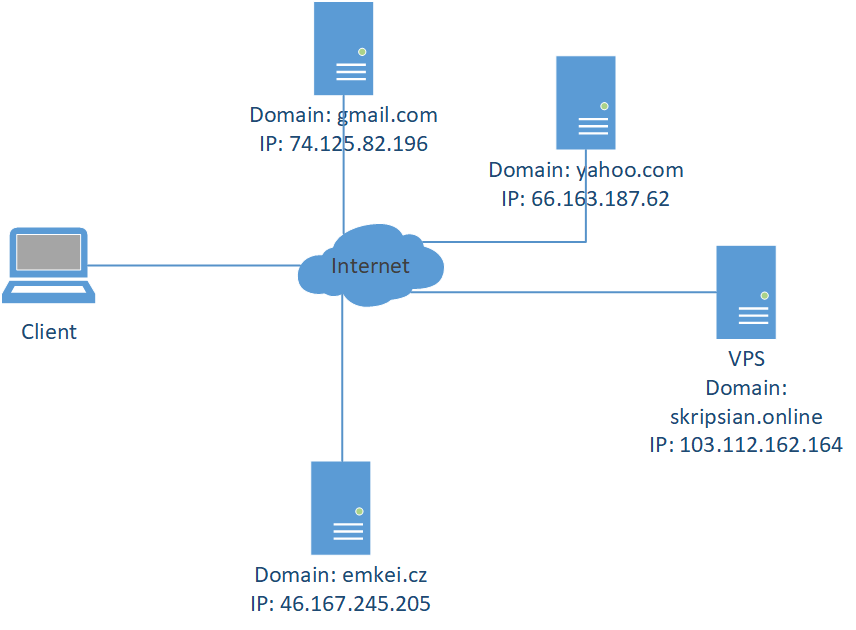
Berdasarkan gambar 3.1 tersebut maka rancangan sistem *filtering email spam*, *virus* dan *email* *spoofing* dapat dijelaskan sebagai berikut.

* + - * 1. Langkah 1 *user* mengirim *email* dengan menggunakan *Mail User Agent* berbasis web (*Roundcube*), *user* mengakses *Roundcube* menggunakan *browser.*
        2. Langkah 2 *Mail server* pengirim meneruskan *email* ke *mail server* penerima dengan menambahkan *private key* pada *header email*.
        3. Langkah 3 *Mail server* pengirim mem*publish public key* pada *DNS server*nya.
  1. 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*, jika *private key* cocok dengan *public key* maka proses akan berlanjut pada langkah ke 5.
  2. 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.
  3. Langkah 6 *Mail server* penerima melakukan pengecekan pada *database* kolaboratif *SpamAssassin*.
  4. Proses pem*filter*an *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*.

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).

### **Rancangan Jaringan Uji Coba**

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



Gambar 3.2 Rancangan Topologi Uji Coba

Rancangan ini diimplementasikan menggunakan *VPS* yang disewa pada penyedia layanan *VPS* dan pada *VPS* telah terinstal sistem operasi *CentOS* Linux *release* 7.3.1611, *VPS* yang telah disewa diberikan satu alamat *IP public* oleh penyedia layanan *VPS* yaitu 103.112.162.164. Pada *VPS* akan dilakukan instalasi *CentOS Web Panel*, konfigurasi *DNS server*, konfigurasi *Mail server*, dan pada komputer *client* telah terinstal sistem operasi windows 10 dan aplikasi browser Google Chrome untuk mengakses *Mail User Agent* berbasis web (Roundcube).

### **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. Pengalamatan *IP* dapat dilihat seperti pada tabel 3.2 berikut.

Tabel 3.2 Pengalamatan *IP*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Perangkat** | ***IP Address*** | ***Network*** | ***Interface*** |
| 1 | *DNS Server*, *HTTP Server*, *SMTP Server*, *POP3/IMAP* *Server (VPS)* | 103.112.162.228/25 | 103.112.162.128 | eth0 |
| 2 | *Client* | *DHCP* | *DHCP* | - |

### **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.

Tabel 3.3 Kebutuhan Akun *Email*

|  |  |  |
| --- | --- | --- |
| **No** | **Alamat *Email*** | ***Domain*** |
| 1 | [naufalhanif1477.nh@gmail.com](mailto:naufalhanif1477.nh@gmail.com) | gmail.com |
| 2 | [naufalhanif74@yahoo.com](mailto:naufalhanif74@yahoo.com) | yahoo.com |
| 3 | [hendarto@skripsian.online](mailto:hendarto@skripsian.online) | skripsian.online |
| 4 | [hendarto@ridho.org](mailto:hendarto@ridho.org) | ridho.org |
| 5 | [yunita@skripsian.online](mailto:yunita@skripsian.online) | skripsian.online |
| 6 | [naufalhanif@skripsian.online](mailto:naufalhanif@skripsian.online) | skripsian.online |

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

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** |
| *CPU* | *Virtual CPU* a7769a6388d5  1 *Core* (2400 MHz) |
| *RAM* | 1 GB |
| *Hard Drive* | 25 GB |

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

Tabel 3.5 Spesifikasi *Client*

|  |  |
| --- | --- |
| **Komponen** | **Spesifikasi** |
| *CPU* | Intel Core i3-7100U, 2.4GHz |
| *RAM* | 4 GB |
| *Hard Drive* | 1 TB |

1. Kebutuhan Perangkat Lunak

Adapun perangkat lunak yang dibutuhkan adalah sebagai berikut:

1. *Linux CentOS release* 7.3.1611 sebagai sistem operasi *VPS*.
2. *CentOS Web Panel* sebagai *tool* untuk memudahkan dalam melakukan konfigurasi *server*.
3. Dovecot sebagai *Mail Delivery Agent*.
4. Postfix sebagai *Mail Transfer Agent*.
5. Roundcube sebagai *Mail User Agent*.
6. Apache sebagai web *server*.
7. Bind9 sebagai *DNS server*.
8. Microsoft Windows 10 sebagai sistem operasi *client*.
9. Google Chrome sebagai *browser client* untuk mengakses *Roundcube*.

## **Tahap Simulasi (*Prototyping*)**

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* skripsian.online kemudian mengirim *email spoofing* tersebut ke *mail server* *Gmail*, *Yahoo! Mail,* dan skripsian.online sebelum dan setelah penerapan *DKIM* dan *SPF* pada *mail server* skripsian.online.

Uji coba kedua dilakukan untuk menguji kinerja *Anti Spam* pada *mail server* skripsian.online 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* skripsian.online sebelum dan setelah penerapan *SpamAssassin*.

Uji coba ketiga dilakukan untuk menguji kinerja *anti virus* pada *mail server* skripsian.online 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* skripsian.online sebelum dan setelah penerapan *ClamAV*.

Uji coba ketiga dilakukan dengan cara membandingkan *header email* yang dikirim oleh salah satu *user* yang berada pada *mail server* skripsian.online ke *Gmail, Yahoo! Mail,* dan skripsian.online sebelum dan setelah penerapan *DKIM*, *SPF*, *Anti Spam*, dan *Anti Virus*.

### **Instalasi Dan Konfigurasi**

Instalasi dan konfigurasi *DKIM*, *SPF*, *Anti Spam,* dan *Anti Virus* dilakukan pada *VPS* yang berfungsi untuk mem*filter email spam* dan *virus* yang masuk serta untuk mencegah adanya *email spoofing* yang mengatasnamakan skripsian.online, 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* skripsian.online.

### **Uji Coba**

Pada tahap ujicoba ini tediri 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*.

## **Tahap Implementasi**

Tahap implementasi merupakan tahap penerapan sistem yang sudah dirancang, agar sistem yang telah dirancang dapat dioperasikan dan digunakan secara optimal sesuai dengan kebutuhan. Selain tahap implementasi akan dilakukan pengujian terhadap sistem yang baru dan akan dilihat kekurangan-kekurangan pada sistem yang baru untuk pengembangan sistem selanjutnya. Pada fase ini penulis akan membangun sebuah *mail server* kemudian pada *mail server* tersebut akan diterapkan *DKIM*, *SPF*, *Anti Spam* dan *Anti Virus* untuk mem*filter*, mengotorisasi, dan mengotentikasi *email*, kemudian penulis akan melakukan analisa pada *mail server* sebelum dan sesudah penerapan *DKIM*, *SPF*, *Anti Spam,* dan *Anti Virus*.

## **Tahap *Monitoring***

Setelah melakukan implementasi, tahapan *monitoring* adalah tahapan penting dalam merancang desain jaringan, tujuan dari tahapan *monitoring* adalah untuk memastikan jaringan komputer berjalan sesuai dengan tujuan pada tahap analisis. Pada fase ini penulis akan melakukan *monitoring* terhadap aktifitas *spam* pada *mail server* skripsian.online dengan menggunakan *maillog server* skripsian.online serta melakukan *monitoring* terhadap aktifitas *spoofing* pada layanan *email* *Yahoo! Mail*, *Gmail,* dan skripsian.online.

# **BAB IV**

# **HASIL DAN PEMBAHASAN**

Bab ini memuat tentang pembahasan dari hasil instalasi dan konfigurasi*,* uji coba dan analisa hasil uji coba.

## **Hasil Instalasi Dan Konfigurasi**

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

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

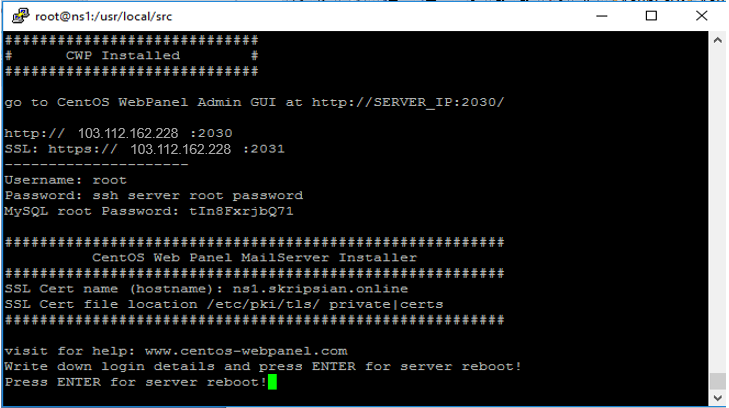
Tahap instalasi dan konfigurasi *server* berisikan instalasi *CentOS Web Panel*, konfigurasi *DNS server*, dan konfigurasi *Mail server*. *Server* yang digunakan pada penelitian ini adalah *Virtual Private Server* yang telah di sewa pada salah satu penyedia jasa layanan *VPS*, Alamat *IP VPS* yang diberikan oleh penyedia jasa layanan *VPS* adalah 103.112.162.228 dengan sistem operasi Linux *CentOS release* 7.3.1611, pada *VPS* telah terinstal *SSH Server* agar *VPS* dapat diakses melalui perangkat lain melalui jaringan *internet* seperti terlihat pada gambar 4.1 berikut.



Gambar 4.1 Linux CentOS release 7.3.1611

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

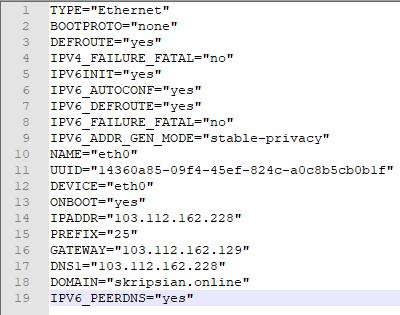
*CentOS Web Panel* digunakan untuk memudahkan dalam melakukan instalasi dan konfigurasi *server* karena proses instalasi *server* akan dilakukan secara otomatis dan proses konfigurasi *server* dapat dilakukan dengan mudah melalui halaman konfigurasi *CentOS Web Panel* yang berbasis web. Tahap instalasi *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 men*download* *file installer* CWP versi terbaru dengan perintah *#wget* [*http://centos-webpanel.com/cwp-latest*](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.



Gambar 4.2 Hasil Instalasi CWP

#### **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 hanya menambahkan parameter DNS1=”103.112.162.228” yang berada pada baris 17 yang berfungsi agar *VPS* menjadi *DNS server*, parameter lain yang perlu ditambahkan adalah DOMAIN=”skripsian.online” yang berada pada baris 18 yang bertujuan agar skripsian.online menjadi domain VPS seperti te dilakukan pada *file* ifcfg-eth0 dengan perintah *#nano /etc/sysconfig/network-scripts/ifcfg-eth0* seperti terlihat pada gambar 4.3 berikut.



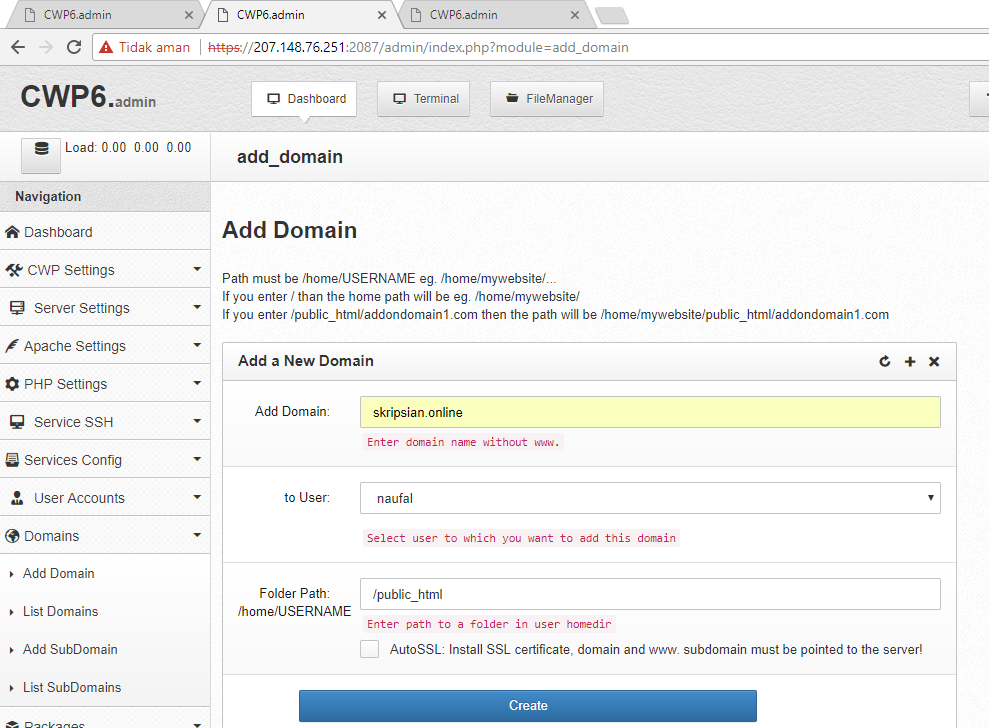
Gambar 4.3 Konfigurasi Interface

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.



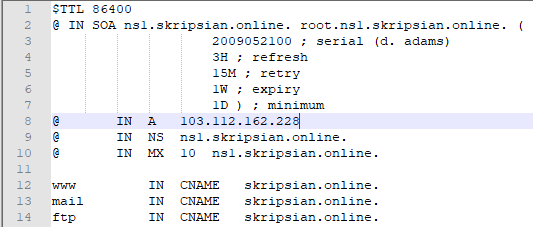
Gambar 4.4 Konfigurasi Name Server

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.



Gambar 4.5 Konfigurasi *Domain*

Jika konfigurasi *domain* berhasil maka secara otomatis akan terbuat *file forward-lookup zone* yang diberi nama skripsian.online.db, pada baris 8 terdapat record @ IN A 103.112.162.228 yang berfungsi untuk memetekan nama *host* ke alamat *IP*, pada baris 9 terdapat *record* @ IN NS ns1.skripsian.online. yang berfungsi untuk memetakan sebuah nama *domain* ke dalam satu daftar dari *server DNS* untuk *domain* skripsian.online, pada baris 10 terdapat *record* @ IN MX 10 ns1.skripsian.online. yang berfungsi untuk memetakan sebuah nama *domain* ke dalam daftar *mail exchange server* untuk *domain* skripsian.online, pada baris 12 sampai 14 merupakan *record CNAME* atau *Canonical Name* yang berfungsi agar nama *domain* skripsian.online dapat diakses menggunakan nama alias [www.skripsian.online](http://www.skripsian.online), [ftp.skripsian.online](ftp://ftp.skripsian.online), dan mail.skripsian.online seperti terlihat pada gambar 4.6 berikut.



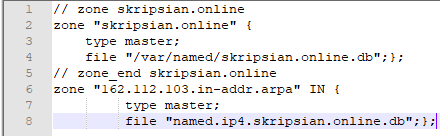
Gambar 4.6 *File* skripsian.online.db

Untuk dapat memetakan alamat *IP* ke nama *domain* maka perlu dibuat *file reverse-lookup zone* pada terminal dengan perintah *#nano /var/named/named.ip4.skripsian.online.db.* pada baris 8 terdapat *record* @ IN NS ns1.skripsian.online. yang berfungsi untuk memetakan sebuah nama *domain* ke dalam satu daftar dari *server DNS* untuk *domain* skripsian.online, pada baris 9 dan 10 terdapat *record pointer* yang berfungsi untuk memetakan nama *domain* ke dalam alamat *IP* seperti terlihat pada gambar 4.7 berikut.



Gambar 4. 7 *File* named.ip4.skripsian.online.db

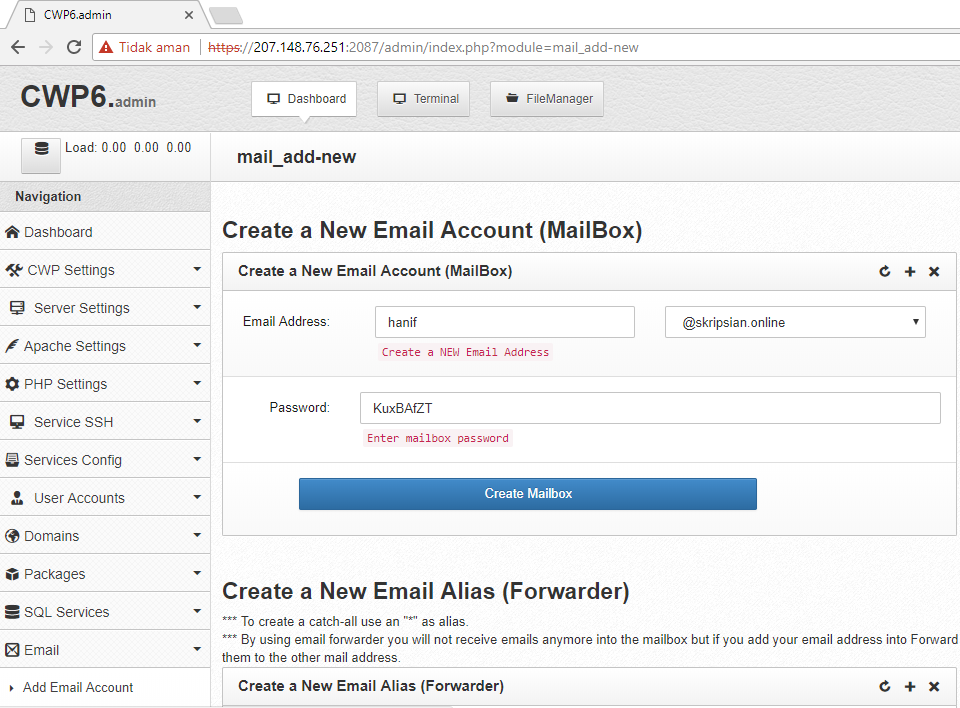
Pada *file* named.conf ditambahkan perintah pada baris 2 yaitu zone "skripsian.online" adalah nama *domain* yang akan digunaka, baris 3 yaitu type master adalah tipe *DNS server* yaitu *primary DNS*, baris 4 adalah lokasi *file forward zone*, kemudian pada baris 6 yaitu zone "162.112.103.in-addr.arpa" IN adalah lingkup *network* dalam *domain* yang akan digunakan sebagai *reverse*, baris 7 yaitu type master adalah tipe *DNS server* yaitu primary DNS, baris 8 adalah lokasi *file reverse zone*, setelah semua konfigurasi selesai ketikan perintah *#service named restart* untuk me*restart DNS server*. Seperti terlihat pada gambar 4.8 berikut.



Gambar 4.8 *File* named.conf

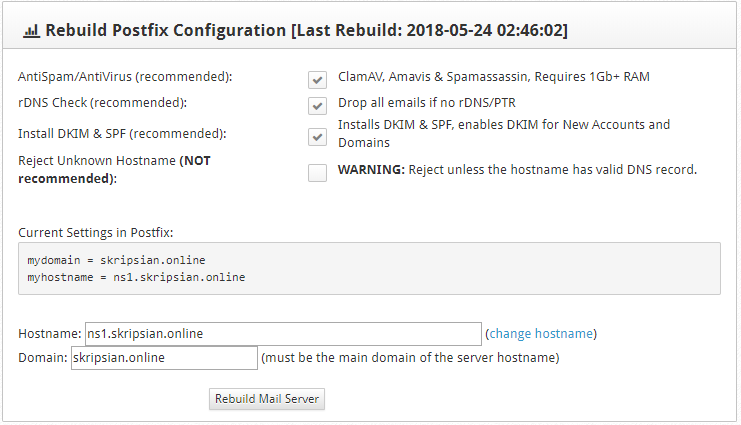
#### **Hasil Konfigurasi Mail Server**

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.



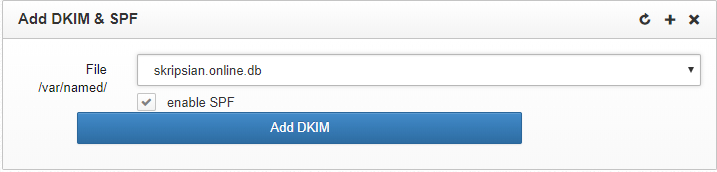
Gambar 4.9 Membuat Akun *Email*

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.



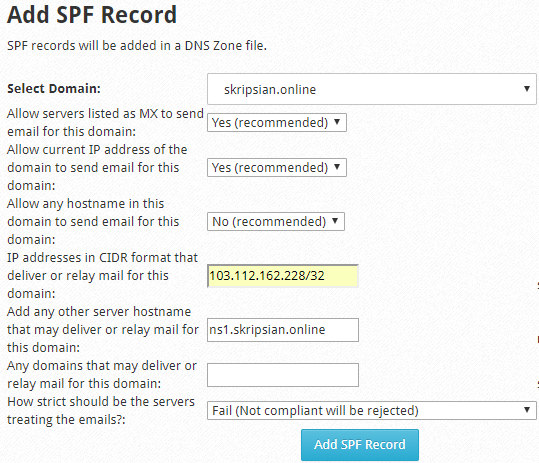
**Gambar 4.10 Instalasi *DKIM*, *SPF, Anti Spam,* dan *Anti Virus***

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



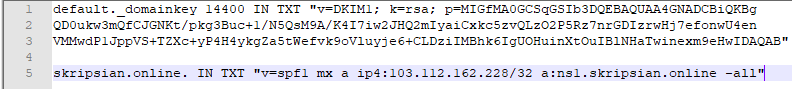
Gambar 4.11 Menambah *DKIM Record* pada *File Zone*

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



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

Pada *file* skripsian.online.db akan terlihat tambahan dua baris dibagian paling bawah seperti terlihat pada gambar 4.13 berikut.



Gambar 4.13 *DKIM* dan *SPF Record*

Pada baris 1 skripsian.online. IN TXT "v=spf1 mx a ip4:103.112.162.228/32 a:ns1.skripsian.online -all" merupakan *record SPF,* v=spf1 berarti versi *SPF* yang digunakan adalah *SPF* versi, a:ns1.skripsian.online berarti hanya mengizinkan pengiriman *email* dengan *hostname* ns1.skripsian.online, ip4:103.112.162.228/32 yang berarti hanya mengizinkan pengiriman email dari *server* dengan alamat *IP* 103.112.162.228, -all berarti menolak semua *email* yang tidak sesuai dengan aturan tersebut.

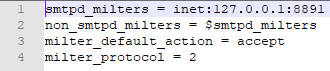
Pada baris 2 adalah *record* *DKIM,* dimana parameter v=DKIM1 berarti versi *DKIM* yang digunakan yaitu *DKIM* versi 1, parameter k=rsa berarti jenis kriptografi yang digunakan adalah rsa, dan parameter p yaitu *public key* yang digunakan.

Agar ns1.skripsian.online 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.



Gambar 4.14 Konfigurasi *File* TrustedHosts

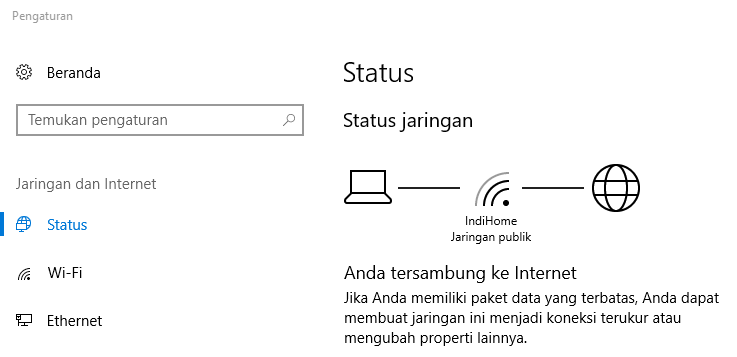
Pada *file* main.cf ditambahkan beberapa parameter seperti smtpd\_milters = inet:127.0.0.1:8891, non\_smtpd\_milters = $smtpd\_milters, milter\_default\_action = accept, dan milter\_protocol = 2 yang terdapat pada baris 1 sampai 4 yang berfungsi untuk mem*filter email*, seperti terlihat pada gambar 4.15 berikut.



Gambar 4.15 Konfigurasi *File* main.cf

### **Hasil Konfigurasi *Client***

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*

## **Hasil Uji Coba**

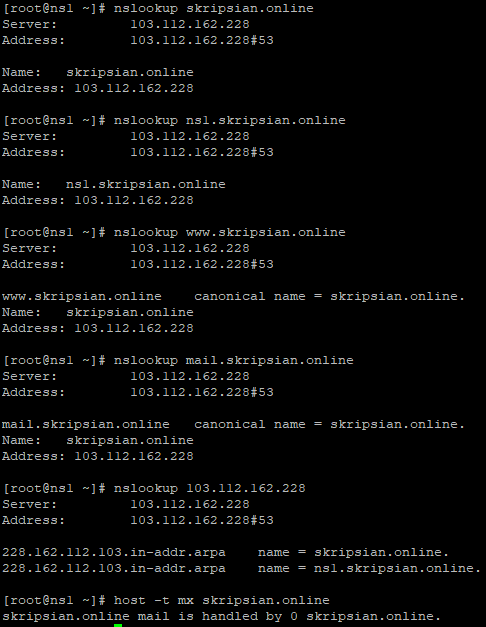
Pada tahap uji coba ini terdiri dari 2 bagian yaitu verifikasi konfigurasi dan uji coba menggunakan berbagai macam skenario.

### **Verifikasi Konfigurasi**

Berikut ini adalah hasil verifikasi konfigurasi yang telah dilakukan sebelumnya untuk mengetahui apakah hasil konfigurasi yang dilakukan sebelumnya berhasil atau tidak.

#### **Verifikasi Konfigurasi DNS Server**

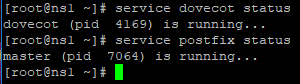
Untuk mengecek fungsi forward-lookup, CNAME, reverse-lookup, dan fitur *mail exchanger* dapat digunakan perintah nslookup dan host –t mx pada terminal seperti terlihat pada gambar 4.17 berikut.



Gambar 4.17 Verifikasi Konfigurasi *DNS Server*

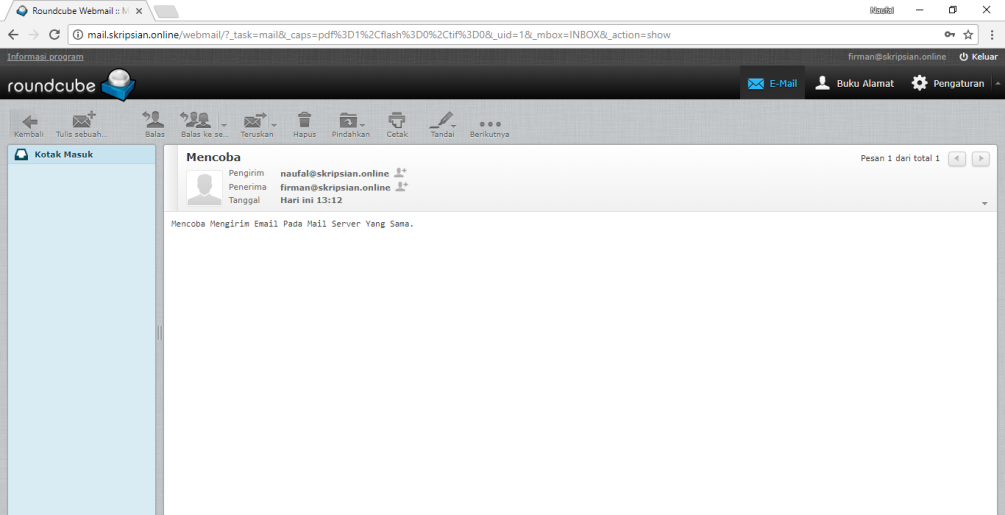
#### **Verifikasi Konfigurasi Mail Server**

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.



Gambar 4.18 Verifikasi Konfigurasi *Mail Server*

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.

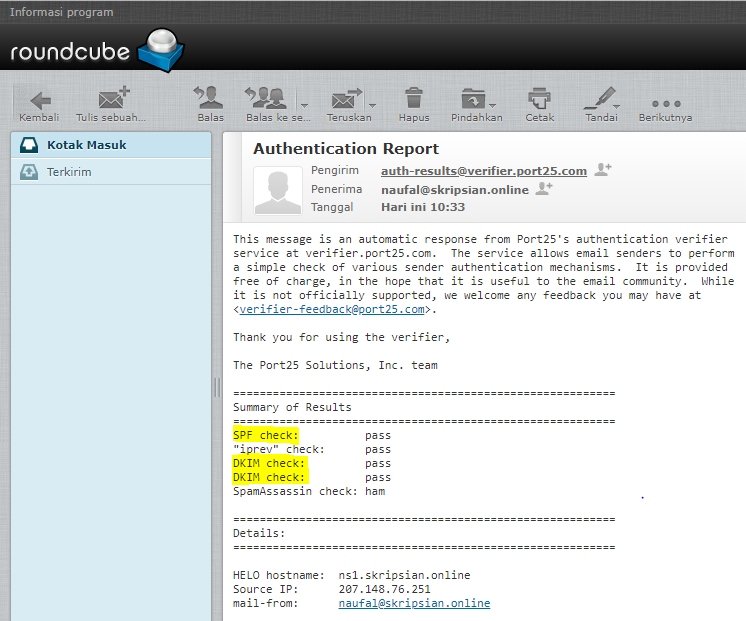


Gambar 4.19 Mengirim *Email* pada *User Email Local*



**Gambar 4.20 Mengirim *Email* Pada *Mail Server* Lain**

Verifikasi fungsi *DKIM*, *SPF* dan SpamAssassin dapat dilakukan dengan cara mengirim email ke alamat check-auth@verifier.port25.com seperti pada gambar 4.21 berikut.



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

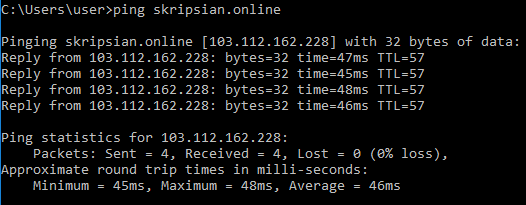
Verifikasi fungsi ClamAV dilakukan dengan cara masuk pada menu Dashboard di CentOS Web Panel, seperti terlihat pada gambar 4.22 berikut.



Gambar 4.22 Verifikasi Fungsi *ClamAV*

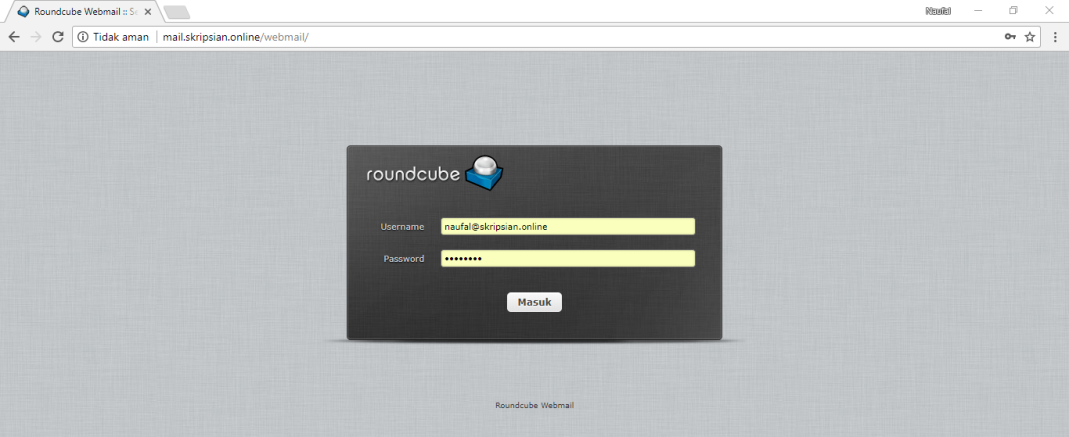
#### **Verifikasi Konfigurasi Client**

Verifikasi konfigurasi *client* dilakukan dengan cara melakukan ping pada *mail server dengan perintah >ping skripsian.online* seperti pada gambar 4.23 berikut.



Gambar 4.23 Ping *Mail Server*

Verifikasi konfigurasi pada *client* juga dapat dilakukan dengan cara mengakses *Mail User Agent* dengan menggunakan aplikasi *browser* kemudian mengakses alamat *URL* skripsian.online/webmail/ seperti terlihat pada gambar 4.24 berikut.



Gambar 4.24 Akses *MUA*

### **Skenario Uji Coba**

Adapun pada tahap skenario hasil uji coba ini berisikan tentang uji coba sebelum diterapkannya *filtering*, otentikasi, dan otorisasi *email spam* dan uji coba setelah diterapkannya *filtering*, otentikasi, dan otorisasi *email spam*.

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

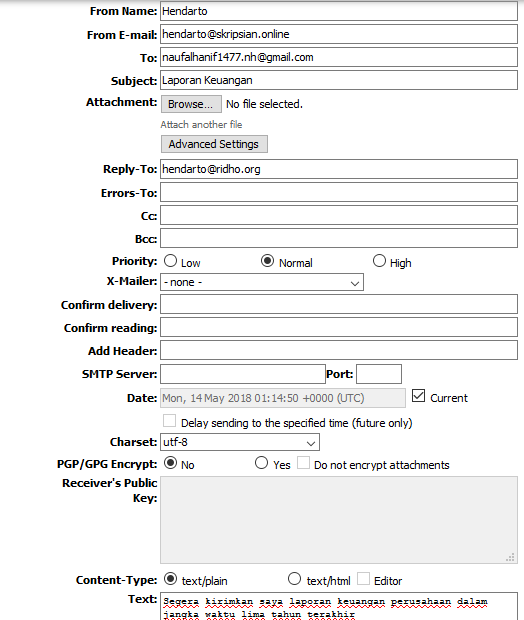
Adapun uji coba yang dilakukan sebelum diterapkan *filtering*, otentikasi, dan otorisasi *email* adalah uji coba mengirim *email spoofing*, uji coba mengirim *email spam*, uji coba mengirim *email* yang mengandung *virus*, dan uji coba pengecekan *header email*.

##### **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 skripsian.online.

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

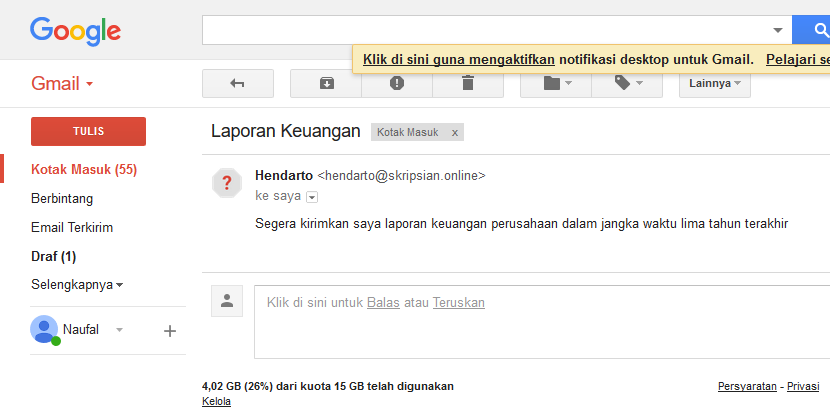
Skenario uji coba untuk menguji kinerja protokol *DKIM* dan *SPF* dilakukan dengan cara melakukan pengiriman *email spoofing*, misalkan skripsian *education center* adalah perusahaan yang bergerak dibidang pendidikan yaitu membuka kursus untuk para mahasiswa semester akhir yang sedang mengerjakan skripsi. Skripsian *education center* mempunyai pesaing didalam menjalankan bisnisnya sehingga pesaing tersebut berusaha untuk menjatuhkan skripsian *education center* dengan cara melakukan penipuan dengan mengirim *email spoofing* menggunakan *Fake Mailer*, sebelum melakukan *email* *spoofing*, pesaing tersebut terlebih dahulu mencari tahu alamat *email* bagian keungan dan direktur skripsian *education center*, setelah pesaing tersebut mengetahui alamat *email* bagian keuangan dan direktur skrpsian *education center* maka pesaing tersebut mulai mengirim *email* *spoofing* yang mengatasnamakan direktur skripsian *education center*, pesaing tersebut menujukan *email spoofing* tersebut ke staf keuangan skripsian *education center*, *email spoofing* tersebut berisi perintah untuk mengirim laporan keuangan skripsian *education center* dalam jangka waktu lima tahun terakhir. Misalkan setelah pesaing tersebut mengumpulkan informasi yang diperlukan untuk melakukan *email spoofing* dan mengetahui bahwa nama pegawai dibagian keuangan tersebut adalah Naufal Hanif dengan alamat naufalhanif1477.nh@gmail.com kemudian telah diketahui bahwa nama direktur skripsian *education center* adalah Hendarto dengan alamat email [hendarto@skripsian.online](mailto:hendarto@skripsian.online) maka pesaing tersebut mulai melakukan *email spoofing* dengan cara membuka situs www.emkei.cz yang digunakan untuk mengirim *email spoofing*, pesaing tersebut mengirim *email spoofing* melalui www.emkei.cz kepada pegawai staf keuangan yaitu Naufal Hanif dengan alamat *email* naufalhanif1477.nh@gmail.com dengan mengatasnamakan direktur skripsian *education center* yang bernama Hendarto dengan alamat *email* [hendarto@skripsian.online](mailto:hendarto@skripsian.online), isi *email* tersebut memerintahkan Naufal Hanif sebagai staf keuangan skripsian *education center* untuk mengirimkannya laporan keuangan skripsian *education center* dalam jangka waktu lima tahun terakhir, balasan *email spoofing* tersebut di arahkan ke *email server* yang dibuat sementara dengan domain ridho.org untuk menampung *email* balasan yang berisi laporan keuangan seperti pada gambar 4.25 berikut.



Gambar 4.25 *Emkei’s Fake Mailer*

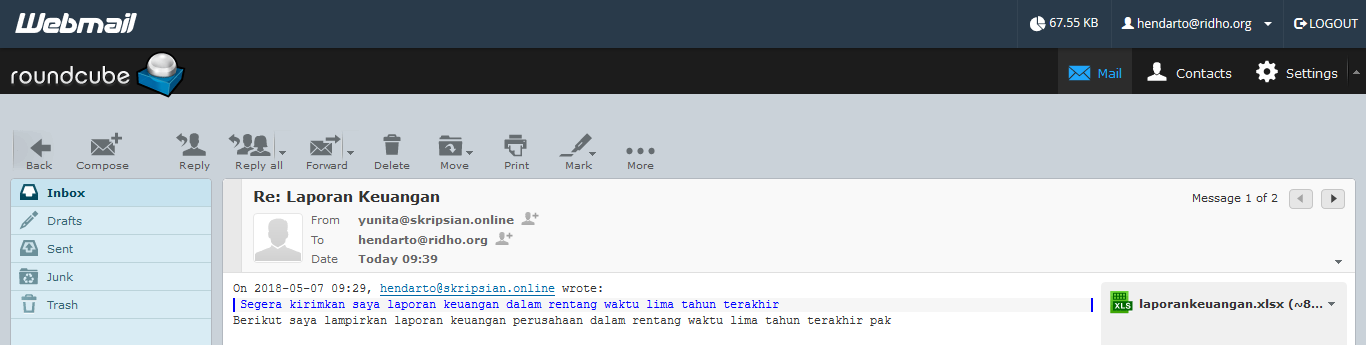
Pada gambar diatas terlihat tampilan dari *Emkei’s Fake Mailer*, pada *text box From Name* diisi dengan nama direktur skripsian *education center*, pada *text box From E-mail* diisi dengan alamat *email* direktur skripsian *education center*, pada *text box To* diisi dengan alamat *email* staf keungan skripsian *education center*, pada *text box Subject* diisi dengan subjek *email*, pada *text box Reply-To* diisi dengan alamat *email* yang digunakan untuk menerima *email* balasan dari *email spoofing* tersebut, dan *text box Text* diisi dengan pesan dari *email spoofing*, jika pada *email server* skripsian *education center* belum menerapkan protokol *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 pada *browser,* kemudian pada situs Emkei’s Fake Mailer pesaing tersebut menuliskan alamat pengirim *email* yaitu [hendarto@skripsian.online](mailto:hendarto@skripsian.online) dan alamat penerima email adalah naufalhanif1477.nh@gmail.com.
      2. Email spoofing tersebut di akses melalui *mail server* gmail.com oleh alamat email naufalhanif1477.nh@gmail.com tanpa adanya proses otentikasi dan otorisasi oleh protokol *DKIM* dan *SPF* sehingga *email spoofing* tersebut berhasil terkirim kealamat *email* naufalhanif1477.nh@gmail.com seperti terlihat pada gambar 4.26 berikut.



Gambar 4.26 *Email Spoofing*

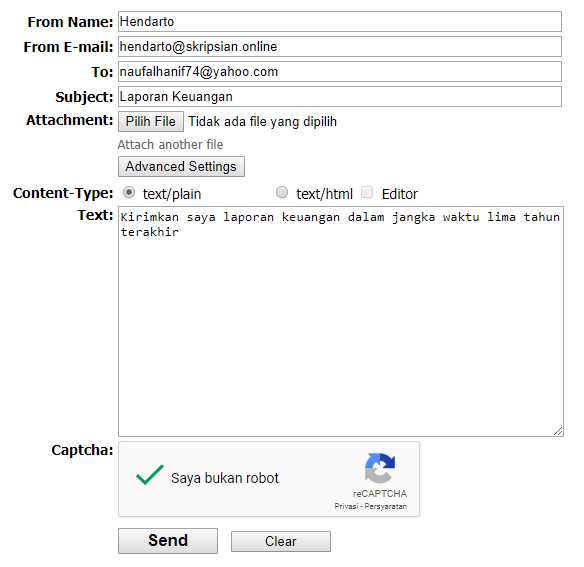
* + - 1. Setelah *email* tersebut masuk ke *inbox* Naufal Hanif maka Naufal Hanif akan membaca *email spoofing* tersebut kemudian *email spoofing* tersebut akan dibalas oleh Naufal Hanif dengan melampirkan laporan keungan yang akan terkirim kealamat *email* [hendarto@ridho.org](mailto:hendarto@ridho.org) sebagai alamat *email* untuk menerima balasan *email spoofing* seperti gambar 4.27 berikut.



Gambar 4.27 Balasan *Email Spoofing*

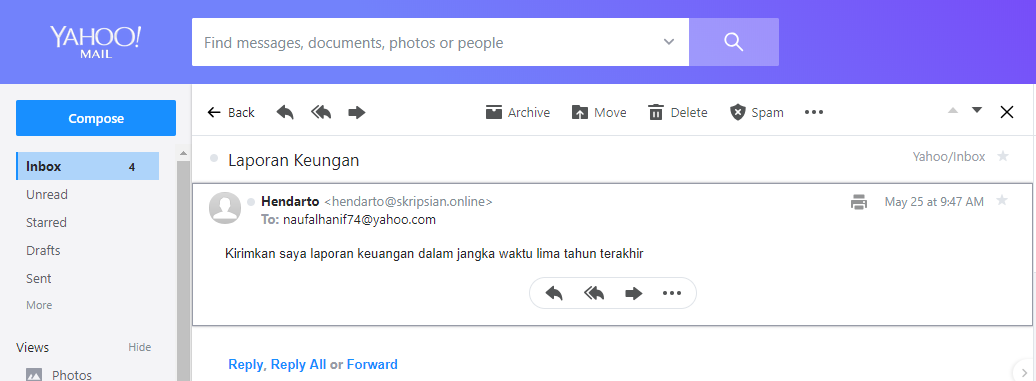
###### **Uji Coba Mengirim *Email Spoofing* ke *Yahoo! Mail***

Dengan menggunakan skenario yang sama seperti pada uji coba mengirim *email spoofing* ke *Gmail*, namum 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* skripsian.online, proses otentikasi dan otorisasi akan sama dengan proses otorisasi dan otentikasi pada uji coba pertama terlihat seperti gambar 4.28 berikut.



Gambar 4. 28 Mengirim *Email Spoofing* ke *Yahoo! Mail*

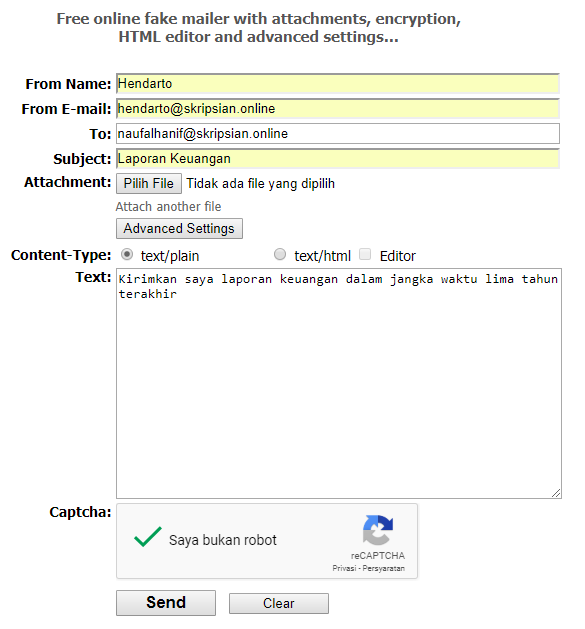
*Email spoofing* tersebut diatas berhasil terkirim ke alamat *email* [naufalhanif74@yahoo.com](mailto:naufalhanif74@yahoo.com) terlihat seperti pada gambar 4.29 berikut.



Gambar 4.29 *Email Spoofing* Terkirim ke *Yahoo! Mail*

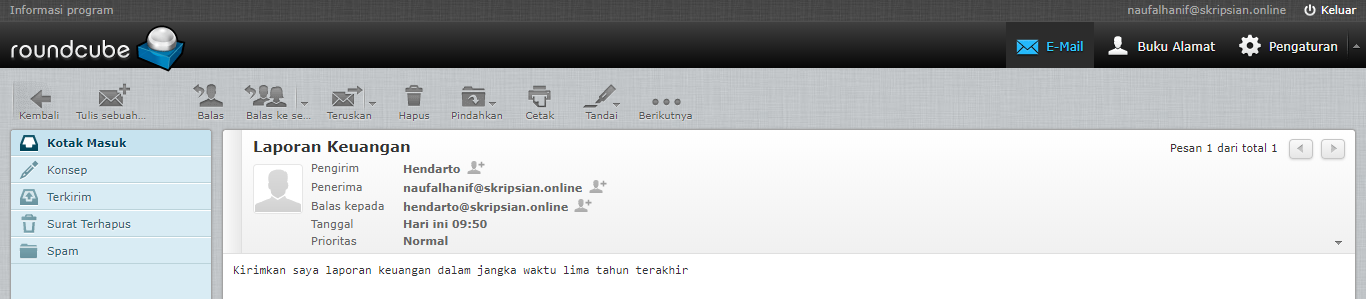
###### **Uji Coba Mengirim *Email Spoofing* pada skripsian.online**

Dengan menggunakan skenario yang sama seperti pada uji coba mengirim *email spoofing* ke layanan *email* *Gmail,* namum pada uji coba ini akan di uji pengiriman *email spoofing* pada *mail server* skripsian.online dengan mengatasnamakan salah satu *user* yang ada pada *mail server* skripsian.online, proses otentikasi dan otorisasi akan sama dengan proses otorisasi dan otentikasi pada uji coba mengirim *email spoofing* ke *Gmail* terlihat seperti gambar 4.30 berikut.



Gambar 4. 30 Mengirim *Email Spoofing* ke skripsian.online

*Email spoofing* tersebut diatas berhasil terkirim ke alamat *email* [naufalhanif@skripsian.online](mailto:naufalhanif@skripsian.online) terlihat seperti pada gambar 4.31 berikut.



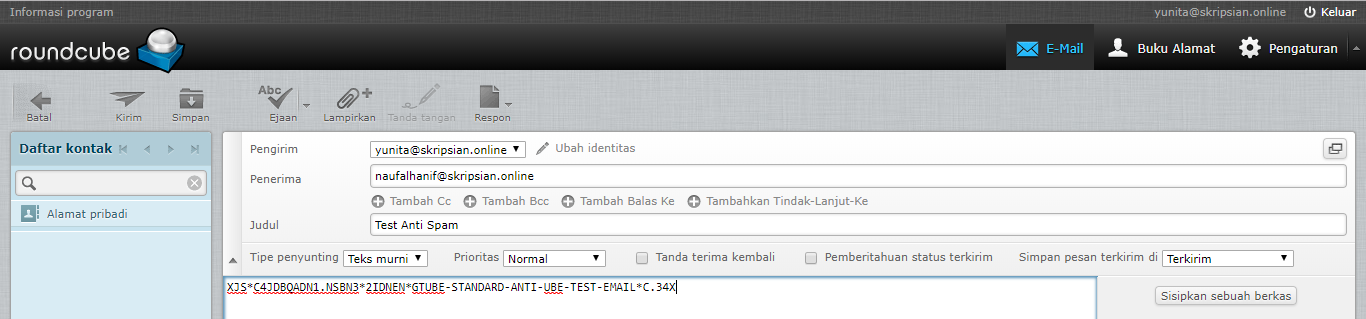
Gambar 4. 31 *Email Spoofing* Terkirim ke *User* skripsian.online

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

Uji coba mengirim *email spam* dilakukan dengan mengirim *email spam* menggunakan layanan *email* skripsian.online, *Yahoo! Mail*, dan *Gmail* ke layanan *email* skripsian.online.

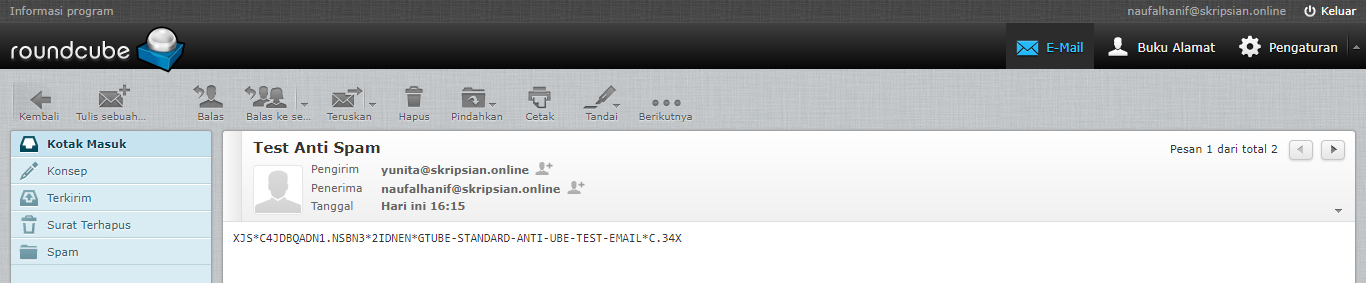
###### **Uji Coba Mengirim *Email Spam* dari skripsian.online**

Uji coba kedua adalah dengan mengirim *email spam* menggunakan layanan *email* skripsian.online ke layanan e*mail* 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 4.32 berikut.



Gambar 4.32 Mengirim *Email Spam* dari skripsian.online

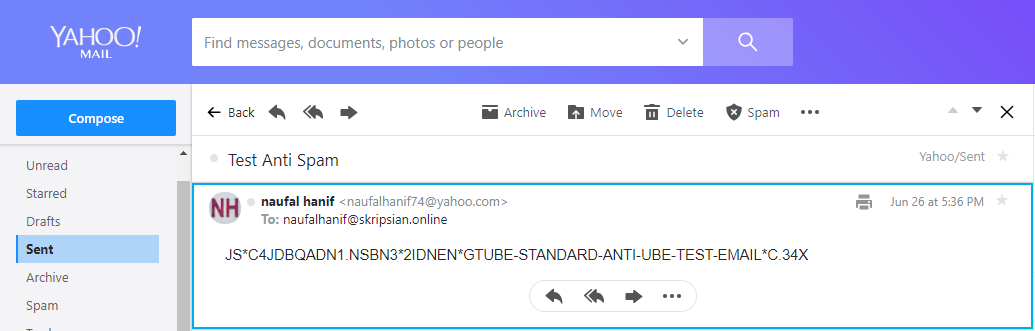
*Email spam* tersebut diatas berhasil terkirim ke alamat *email* [naufalhanif@skripsian.online](mailto:naufalhanif@skripsian.online) karena belum ada penerapan *anti spam* pada *mail server* skripsian.online terlihat seperti pada gambar 4.33 berikut.



Gambar 4.33 *Email Spam* dari sripsian.online Terkirim

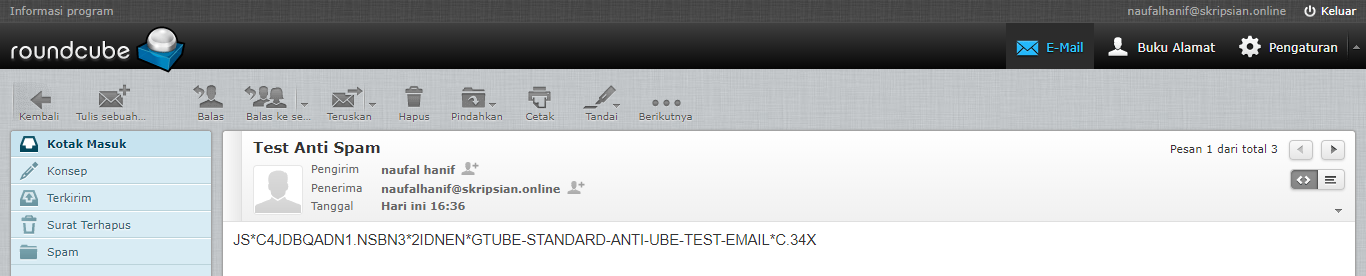
###### **Uji Coba Mengirim *Email Spam* dari *Yahoo! Mail***

Dengan menggunakan skenario yang sama seperti pada uji coba mengirim *email spam* dari layanan *email* skripsian.online ke layanan *email* skripsian.online, namum pada uji coba ini akan di uji pengiriman *email spam* ke layanan e*mail* skripsian.online dari layanan *email* *Yahoo! Mail* terlihat seperti gambar 4.34 berikut.



Gambar 4.34 Mengirim *Email Spam* dari *Yahoo! Mail*

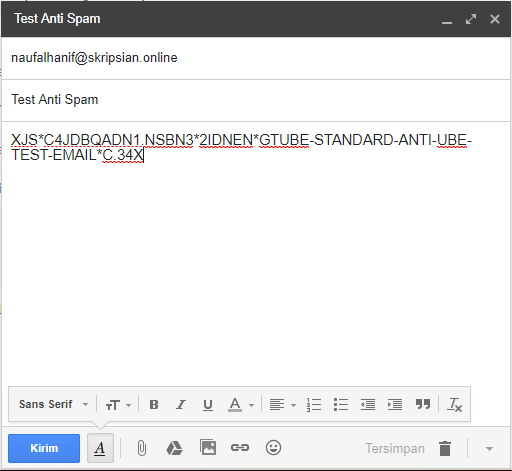
*Email spam* tersebut diatas berhasil terkirim ke alamat *email* [naufalhanif@skripsian.online](mailto:naufalhanif@skripsian.online) karena belum ada penerapan *anti spam* pada *mail server* skripsian.online terlihat seperti pada gambar 4.35 berikut.



**Gambar 4.35 *Email Spam* dari *Yahoo! Mail* Terkirim**

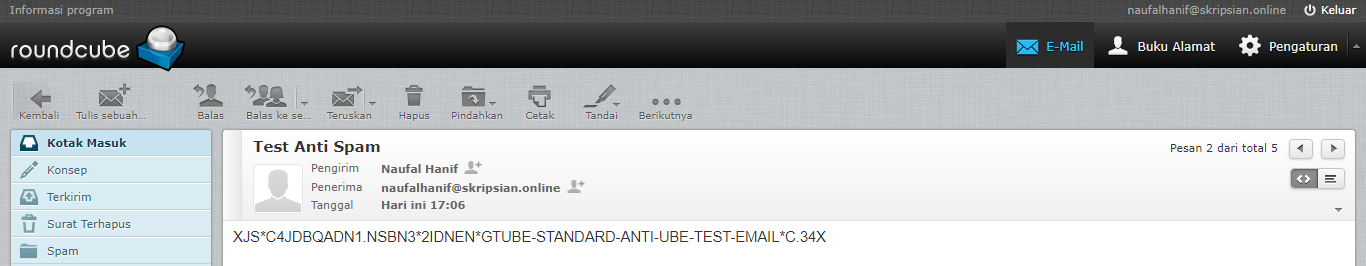
###### **Uji Coba Mengirim *Email Spam* dari *Gmail***

Dengan menggunakan skenario yang sama seperti pada uji coba mengirim *email spam* dari layanan *email* skripsian.online ke layanan *email* skripsian.online, namum pada uji coba ini akan di uji pengiriman *email* *spam* ke layanan e*mail* skripsian.online dari layanan *email Gmail* terlihat seperti gambar 4.36 berikut.



Gambar 4.36 Mengirim *Email Spam* dari *Gmail*

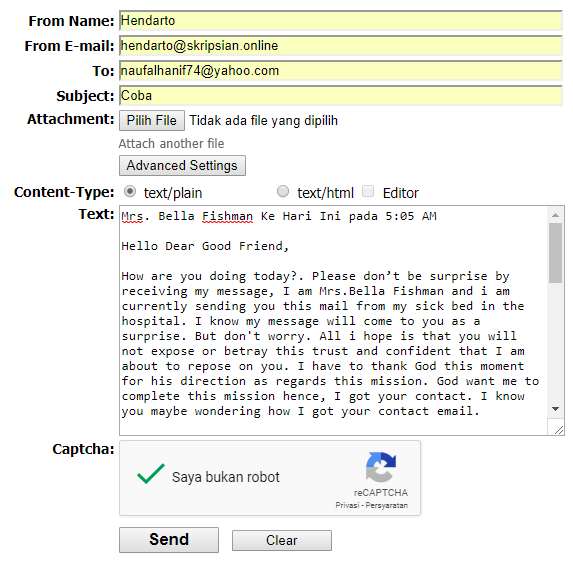
*Email spam* tersebut diatas berhasil terkirim ke alamat *email* [naufalhanif@skripsian.online](mailto:naufalhanif@skripsian.online) karena belum ada penerapan *anti spam* pada *mail server* skripsian.online terlihat seperti pada gambar 4.37 berikut.



Gambar 4.37 *Email Spam* dari *Gmail* Terkirim

###### **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 4.38 berikut.



Gambar 4.38 Mengirim *Email Spam* Tanpa *GTUBE TEST*

*Email* dengan *format spam* tersebut terkirim ke *Yahoo! Mail* dan masuk ke dalam *folder spam* seperti terlihat pada gambar 4.39 berikut.



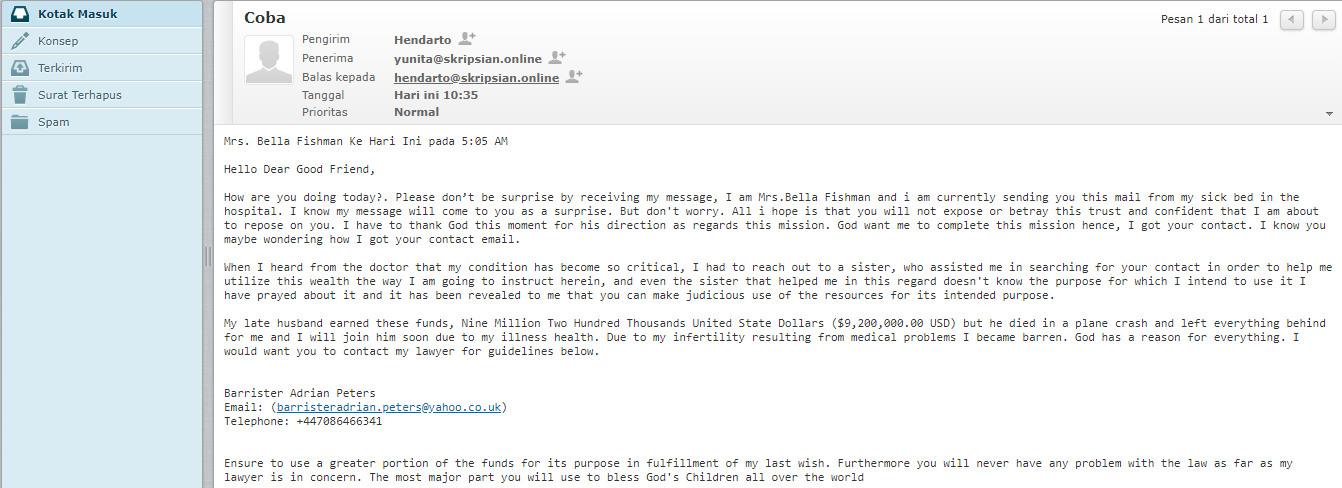
Gambar 4.39 *Email* Terindikasi Sebagai *Spam* oleh *Yahoo! Mail*

*Yahoo! Mail* mengindikasi bahwa *email* tersebut diatas merupakan *spam* sehingga *format email* tersebut diatas dapat dijadikan acuan untuk melakukan uji coba mengirim *email spam* tanpa *GTUBE test* dengan cara mengirim *email* seperti *format email* tersebut diatas dari *Emkei’s Fake Mailer* ke salah satu *user* yang ada pada layanan *email* skripsian.online seperti terlihat pada gambar 4.40 berikut.



Gambar 4.40 Mengirim *Email Spam* Tanpa *Format GTUBE Test*

*Email spam* tanpa *format GTUBE Test* tersebut terindikasi sebagai *email ham* dan masuk pada kotak masuk pengguna layanan *email* skripsian.online karena belum ada penerapan *SpamAssassin* dan *Amavisd-New* seperti terlihat pada gambar 4.41 berikut.



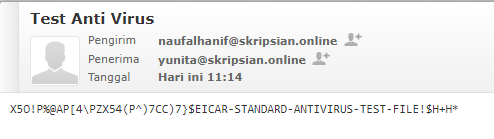
Gambar 4. 41 *Email Spam* Tanpa *GTUBE Test* Terkirim

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

Uji coba pengiriman *email* yang mengandung *virus* dilakukan dengan mengirim *email* yang mengandung *virus* dari layanan *email* skripsian.online, *Yahoo! Mail*, dan *Gmail* ke layanan *email* skripsian.online.

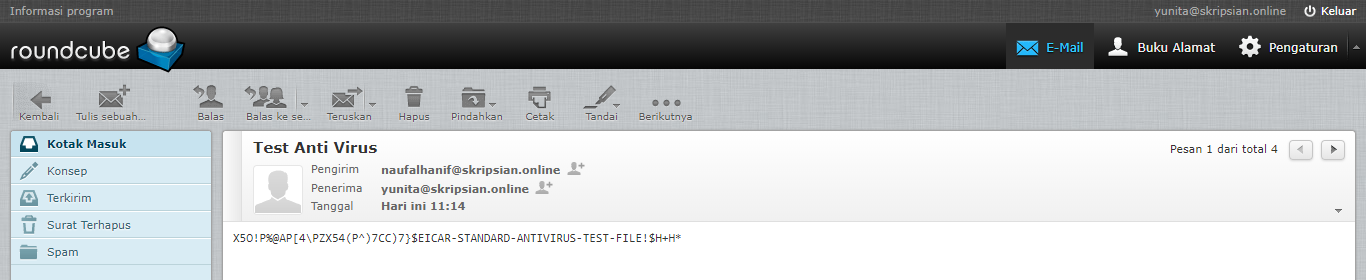
###### **Uji Coba Mengirim *Email* yang Mengandung *Virus* dari skripsian.online**

Uji coba pengiriman *email* yang mengandung *virus* dilakukan dengan mengirim *email* yang berisi 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* skripsian.online ke layanan *email* skripsian.online, seperti pada gambar 4.42 berikut.



Gambar 4. 42 *EICAR Test* dari skripsian.online

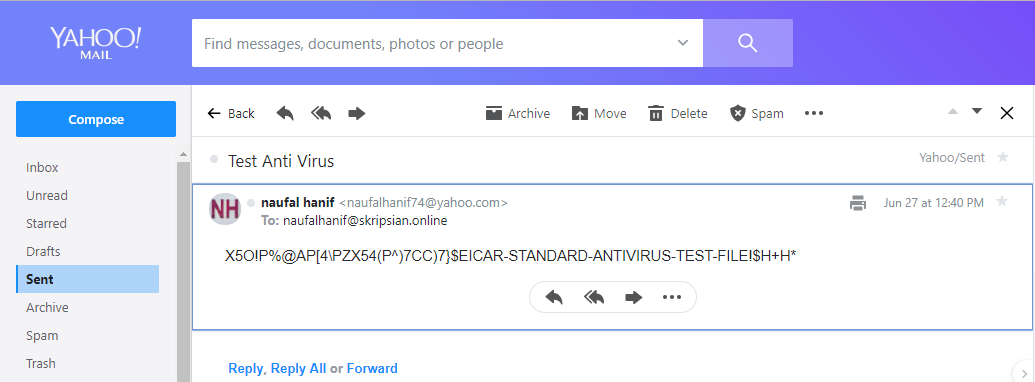
Setelah *email* tersebut dikirim pada salah satu *user* *email* yang ada pada *mail server* skripsian.online maka *email* yang mengandung *virus* tersebut berhasil terkirim ke *user* yang berada pada *mail server* skripsian.online seperti terlihat pada gambar 4.43 berikut.



**Gambar 4.43 Email Mengandung Virus dari skripsian.online Terkirim**

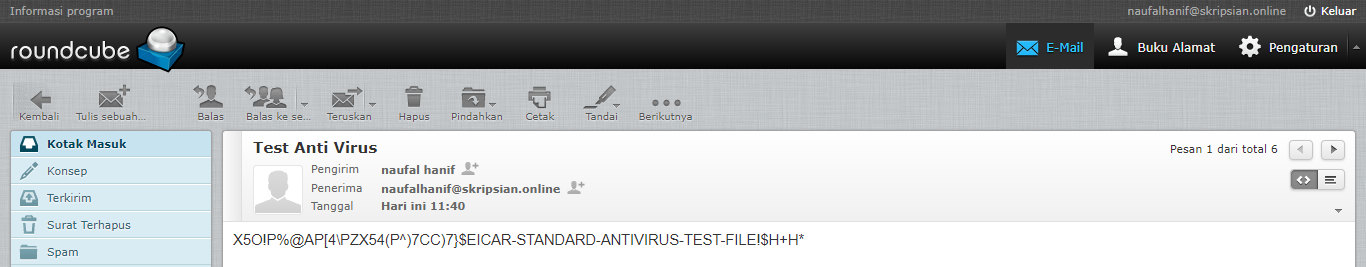
###### **Uji Coba Mengirim *Email* yang Mengandung *Virus* dari *Yahoo! Mail***

Uji coba pengiriman *email* yang mengandung *virus* dilakukan dengan mengirim *email* dengan isi X5O!P%@AP[4\PZX54(P^)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* *Yahoo! Mail* ke layanan *email* skripsian.online, seperti pada gambar 4.44 berikut.



**Gambar 4.44 *EICAR Test* dari *Yahoo! Mail***

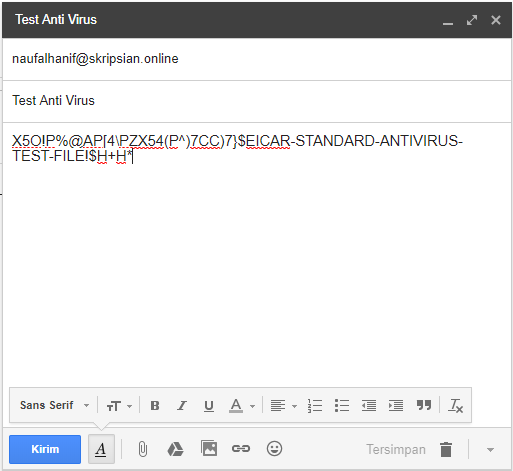
Setelah *email* tersebut dikirim pada salah satu *user* *email* yang ada pada *mail server* skripsian.online, maka *email* yang mengandung *virus* tersebut berhasil terkirim ke *user email* yang berada pada *mail server* skripsian.online seperti terlihat pada gambar 4.45 berikut.



Gambar 4.45 *Email* Mengandung *Virus* dari *Yahoo! Mail* Terkirim

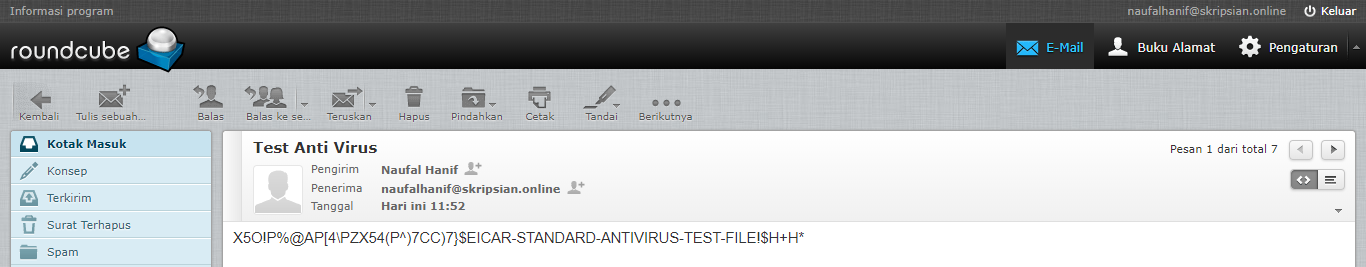
###### **Uji Coba Mengirim *Email* yang Mengandung *Virus* dari *Gmail***

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* skripsian.online, seperti pada gambar 4.46 berikut.



Gambar 4.46 *EICAR Test* dari *Gmail*

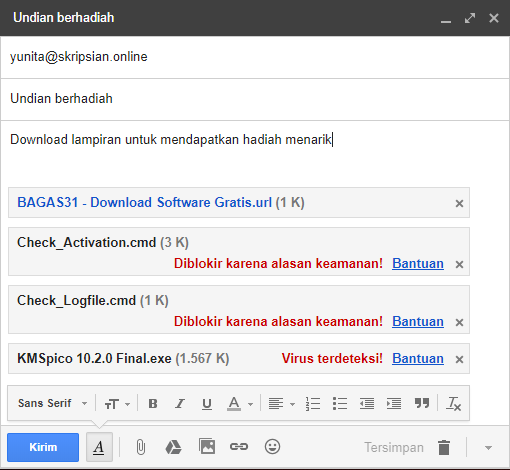
Setelah *email* tersebut dikirim pada salah satu pengguna *email* yang ada pada *mail server* skripsian.online maka *email* yang mengandung *virus* tersebut berhasil terkirim ke penerima yang berada pada *mail server* skripsian.online seperti terlihat pada gambar 4.47 berikut.



**Gambar 4.47 *Email* Mengandung *Virus* dari *Gmail* Terkirim**

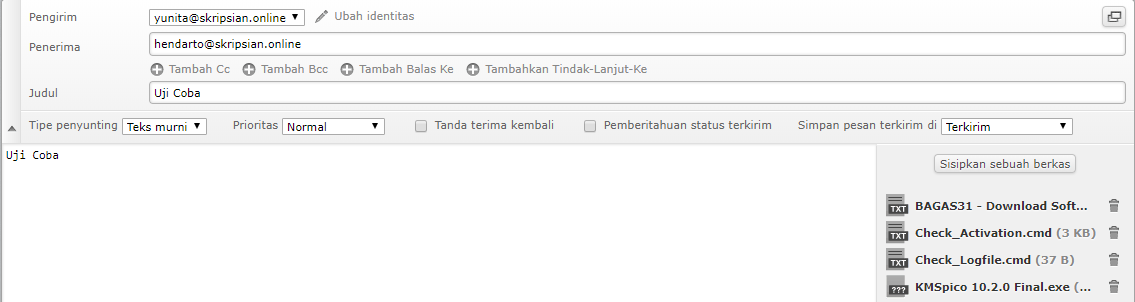
###### **Uji Coba Mengirim *Email* *Virus* Tanpa *EICAR Test***

Uji coba mengirim *email* yang mengandung *virus* dilakukan dengan mengirim *email* dari layanan *email gmail*, *email* yang dikirim diberi lampiran berupa program *crack* dengan ekstensi .exe yang akan terdeteksi sebagai *virus* oleh *gmail* seperti terlihat pada gambar 4.48 berikut.



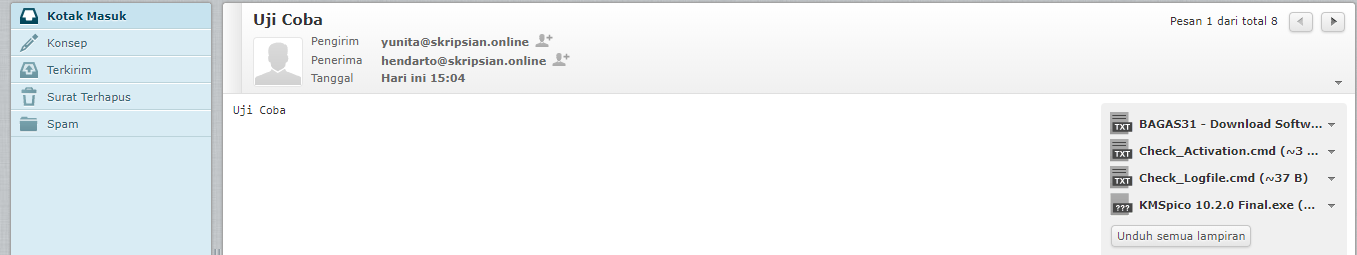
Gambar 4.48 Lampiran Terdeteksi Sebagai *Virus*

Lampiran yang telah terdeteksi sebagai *virus* oleh layanan *email gmail* akan dikirim ke layanan *email* skripsian.online untuk menguji layanan *email* skripsian.online sebelum penerapan *anti virus* seperti terlihat pada gambar 4.49 berikut.

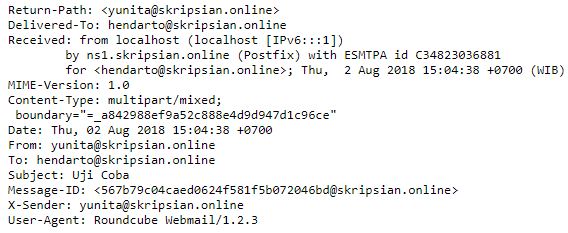


Gambar 4.49 Mengirim *Virus* Sebelum Penerapan *Anti Virus*

*Email* yang mengandung *virus* tersebut berhasil terkirim dan masuk pada kotak masuk pengguna karena tidak ada proses pendeteksian *virus* yang dilakukan oleh *Amavisd-New*, proses pendeteksian virus juga dapat dilihat pada *header email* dengan mengecek parameter *X-Virus-Scanned*, apabila tidak terdapat parameter *X-Virus-Scanned* pada *header email*, maka belum terjadi proses pendeteksian *virus* pada *email* seperti terlihat pada gambar 4.50 dan 4.51 berikut.



Gambar 4.50 *Virus* Berhasil Terkirim ke Kotak Masuk Pengguna *Email*



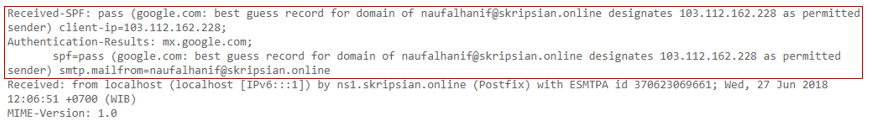
Gambar 4.51 *Header Email* Tanpa Parameter *X-Virus-Scanned*

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

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

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

Uji coba ini dilakukan dengan mengirim *email* dari salah satu *user email* yang ada pada skripsian.online 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 *DKIM*, *SPF, anti spam,* dan *anti virus*, *header email* sebelum diterapkannya *DKIM*, *SPF, anti spam,* dan *anti virus* terlihat seperti gambar 4.52 berikut.

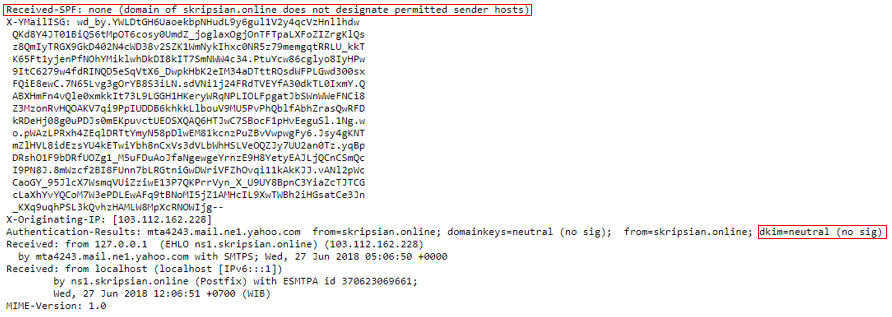


Gambar 4.52 Cuplikan *Header Email* pada Gmail Sebelum Penerapan

Pada gambar 4.52 terlihat pada cuplikan *header email* hanya terdapat parameter Received-SPF dan belum terdapat parameter *DKIM-Signature* atau tanda tangan *digital* dan *X-Virus-Scanned* karena belum ada penerapan *DKIM*, *ClamAV*, dan *Amavisd-New*.

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

Uji coba ini dilakukan dengan mengirim *email* menggunakan salah satu *user email* yang ada pada skripsian.online 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 *DKIM*, *SPF, anti spam,* dan *anti virus*, *header email* sebelum diterapkannya *DKIM*, *SPF, anti spam,* dan *anti virus* terlihat seperti gambar 4.53 berikut.

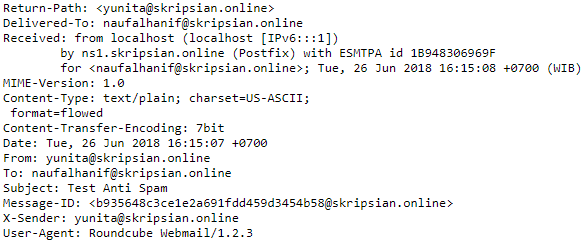


Gambar 4.53 Cuplikan *Header Email* pada *Yahoo! Mail* Sebelum Penerapan

Pada gambar 4.53 dapat dilihat cuplikan *haeder 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=neutral (no sig) yang berarti belum ada tanda tangan *digital* karena belum diterapkan *DKIM*.

###### **Header *Email* pada skripsian.online**

Uji coba ini dilakukan dengan mengirim *email* menggunakan salah satu *user email* yang ada pada skripsian.online ke salah satu *user email* yang ada pada skripsian.online kemudian melakukan pengecekan *header email* dan melakukan perbandingan terhadap *header email* sebelum dan setelah penerapan *DKIM*, *SPF, anti spam,* dan *anti virus*, *header email* sebelum diterapkannya *DKIM*, *SPF, anti spam,* dan *anti virus* terlihat seperti gambar 4.54 berikut.



Gambar 4.54 Cuplikan *Header Email* pada skripsian Sebelum Penerapan

Pada gambar 4.54 terlihat *haeder email* belum terdapat parameter *X-Virus-Scanned* dan *DKIM-Signature* karena belum diterapkan *Amavisd-New* dan *DKIM*.

#### **Uji Coba Setelah Diterapkan Filtering, Otentikasi dan Otorisasi**

Adapun 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*.

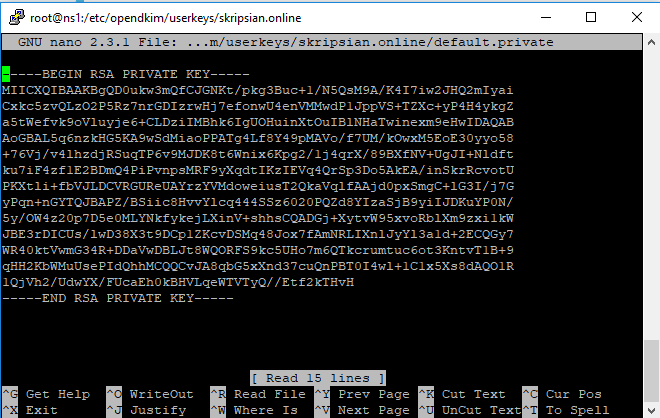
##### **Uji Coba Mengirim *Email Spoofing***

Uji coba mengirim *email spoofing* dilakukan dengan mengirim *email spoofing* dari *Emkei’s Fake Mailer* ke *Gmail*, *Yahoo! Mail*, dan skripsian.online dengan mengatasnamakan salah satu *user email* yang ada pada skripsian.online.

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

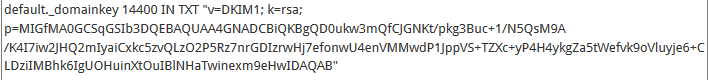
Proses uji coba mengirim *email spoofing* pada *email server* akan berbeda setelah protokol *SPF* dan *DKIM* diterapkan pada *mail server* karena protokol *SPF* dan *DKIM* akan melakukan otentikasi dan otorisasi pada setiap *email* yang datang dari *mail server* skripsian.online. Proses yang terjadi setelah penerapan protokol *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 menggunakan *browser* kemudian pada situs www.emkei.cz pesaing tersebut menuliskan alamat pengirim *email* yaitu [hendarto@skripsian.online](mailto:hendarto@skripsian.online) dan alamat penerima *email* yaitu naufalhanif1477.nh@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* skripsian.online terlihat seperti gambar 4.55 berikut.



Gambar 4.55 *Private Key* pada skripsian.online

1. 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* skripsian.online yang cocok dengan *public key* yang telah diletakan pada *DNS server* skripsian.online sehingga pesan tersebut tidak memiliki tanda tangan *digital* pada *header email* (proses *DKIM*) terlihat seperti gambar 4.56 dan 4.57 berikut.



Gambar 4.56 *Public Key* pada *DNS Server* skripsian.online



Gambar 4.57 Cuplikan *Header Email*

1. Selanjutnya *mail server* *Gmail* akan melakukan pengecekan *SIDF* (*Sender ID Framework*) pada *record DNS server* skripsian.online, karena alamat *IP* *Emkei’s Fake Mailer* adalah 46.167.245.205 maka *email spoofing* tersebut dianggap sebagai *spam* dikarenakan *record SPF* yang ada pada *DNS server* skripsian.online hanya mengizinkan pengiriman *email* dari alamat yang telah diotorisasi yaitu alamat *IP* 103.112.162.228 yang merupakan alamat mail server skripsian.online dan nilai dari parameter *Received-SPF* pada *header email* adalah *fail* sehingga *email* tersebut akan ditandai sebagai *email spam* oleh *server Gmail* dan *email spoofing* tersebut diblok oleh *Gmail* (proses *SPF*) terlihat seperti gambar 4.58 dan 4.59 berikut.

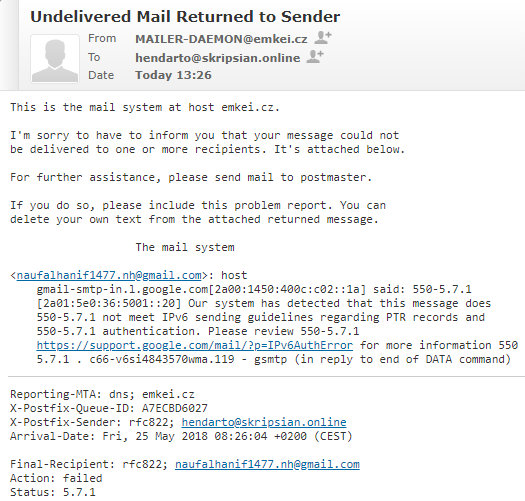


Gambar 4.58 *SPF Record* pada skripsian.online



Gambar 4.59 Parameter *Received-SPF* pada *Header Email*

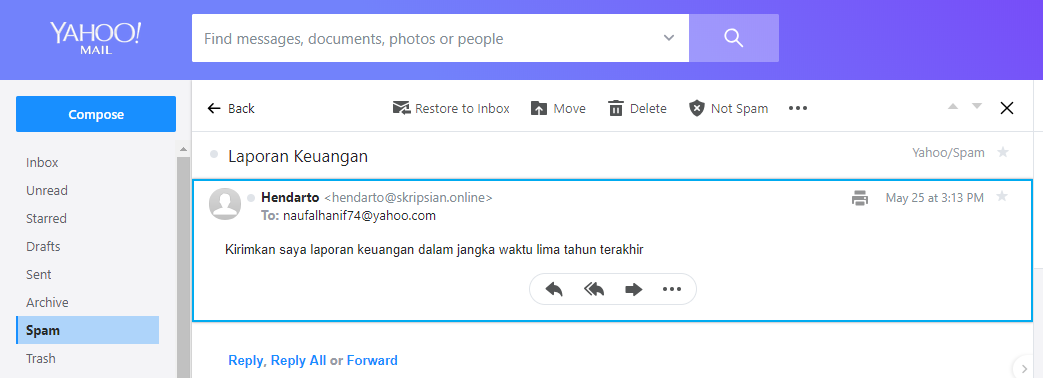
1. *Mail system* *Emkei’s Fake Mailer* menerima kode *error* 550-5.7.1 dari *server Gmail* kemudian *mail system* *Emkei’s Fake Mailer* mengirimkan pemberitahuan *error* tersebut ke alamat *email user* yang asli sehingga *user email* yang asli dapat mengetahui bahwa *email*nya telah dimanfaatkan oleh orang yang tidak bertanggung jawab seperti terlihat pada gambar 4.60 berikut.



Gambar 4.60 Pemberitahuan *Mail System* *Emkei’s Fake Mailer*

###### **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 diblokir sehingga *email spoofing* tidak masuk pada *folder* *inbox* ataupun *folder spam* penerima *email* sedangkan pada *Yahoo! Mail,* *email spoofing* tidak diblokir tetapi dimasukan kedalam *folder spam* penerima *email* seperti gambar 4.61 berikut.



Gambar 4.61 *Email Spoofing* Masuk ke *Folder Spam*

###### **Uji Coba Mengirim *Email Spoofing* ke skripsian.online**

Selain mengirim *email spoofing* ke *Gmail* dan *Yahoo! Mail*, pengiriman *email spoofing* juga dilakukan pada skripsian.online untuk membandingkan perlakuan yang diberikan pada *email spoofing* antara dua layanan *email* tersebut. Pada *Gmail*, *email spoofing* yang masuk langsung diblokir sehingga *email spoofing* tidak masuk pada *folder* *inbox* ataupun *folder spam* penerima *email*, pada *Yahoo! Mail*, *email spoofing* tidak diblokir tetapi dimasukan kedalam *folder spam* penerima *email*, dan pada skripsian.online *email spoofing* masuk pada *folder inbox*, seperti gambar 4.62 berikut.



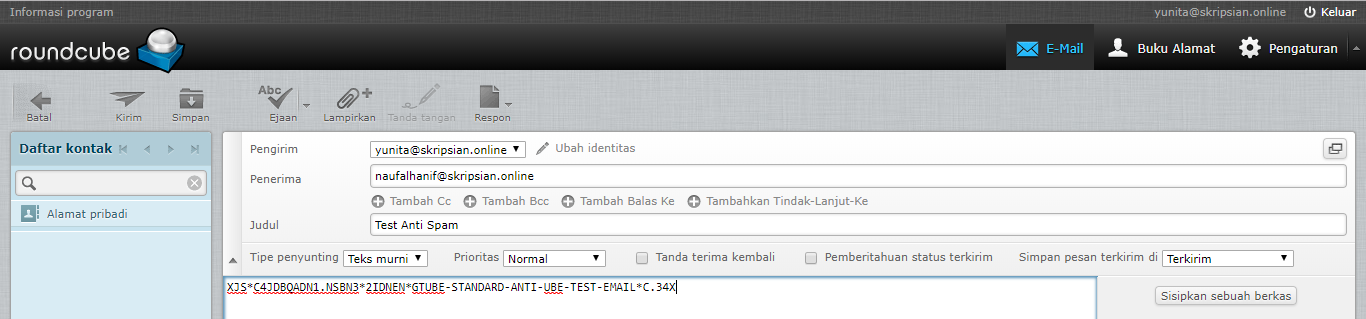
Gambar 4.62 *Email Spoofing* Masuk pada *Folder Inbox*

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

Uji coba mengirim *email spam* dilakukan dengan mengirim *email spam* menggunakan layanan *email* skripsian.online, *Yahoo! Mail*, dan *Gmail* ke layanan *email* skripsian.online.

###### **Uji Coba Mengirim *Email Spam* dari skripsian.online**

Uji coba mengirim *email spam* dari layanan *email* skripsian.online ke layanan *email* skripsian.online 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 4.63 berikut.



Gambar 4.63 Mengirim *Email Spam* dari skripsian.online Setelah Penerapan

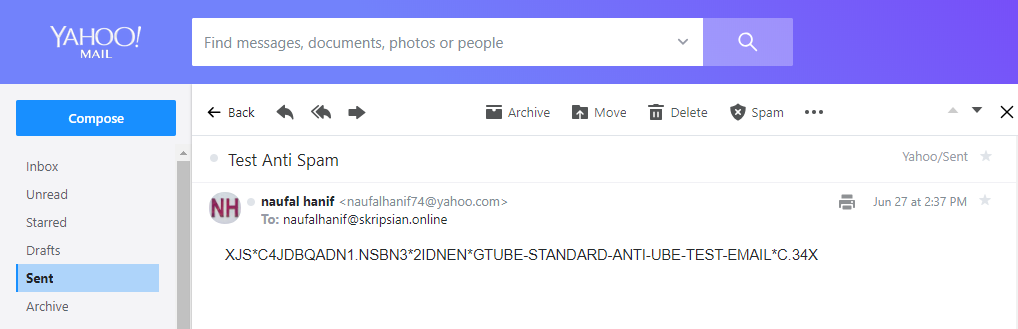
*Email spam* tersebut diatas diblokir oleh *Amavisd-New* karena terindikasi sebagai *email spam* oleh *SpamAssassin*, hasil pem*filter*an *email spam* dapat dilihat pada *mail log* dengan menggunakan perintah *#cat /var/log/maillog* seperti terlihat pada gambar 4.64 berikut.



Gambar 4.64 *Email* dari skripsian.online Terindikasi *Spam*

###### **Uji Coba Mengirim *Email Spam* dari *Yahoo! Mail***

Uji coba mengirim *email spam* dari layanan *email* *Yahoo! Mail* ke skripsian.online adalah dengan mengirim *email spam* menggunakan layanan *email Yahoo! Mail* 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 4.65 berikut.



**Gambar 4.65 Mengirim *Email Spam* dari *Yahoo! Mail* Setelah Penerapan**

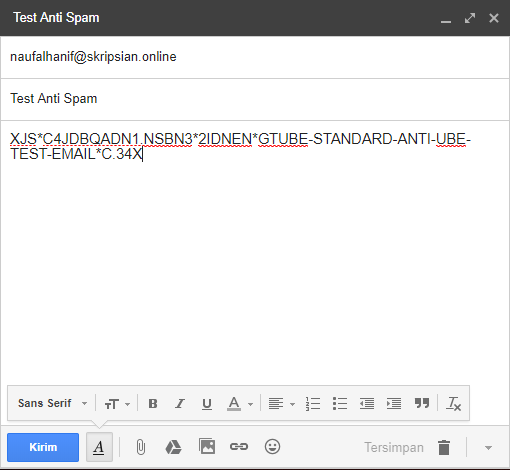
*Email spam* tersebut diatas diblokir oleh *Amavisd-New* karena terindikasi sebagai *email spam* oleh *SpamAssassin*, hasil pem*filter*an *email spam* dapat dilihat pada *mail log* dengan menggunakan perintah *#cat /var/log/maillog* seperti terlihat pada gambar 4.66 berikut.



Gambar 4.66 *Email* dari *Yahoo! Mail* Terindikasi *Spam*

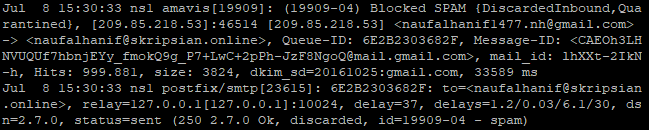
###### **Uji Coba Mengirim *Email Spam* dari *Gmail***

Uji coba mengirim *email spam* dari layanan *email* *Gmail* ke layanan *email* skripsian.online adalah dengan mengirim *email spam* menggunakan layanan *email Gmail* 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 4.67 berikut.



Gambar 4.67 Mengirim *Email Spam* dari *Gmail* Setalah Penerapan

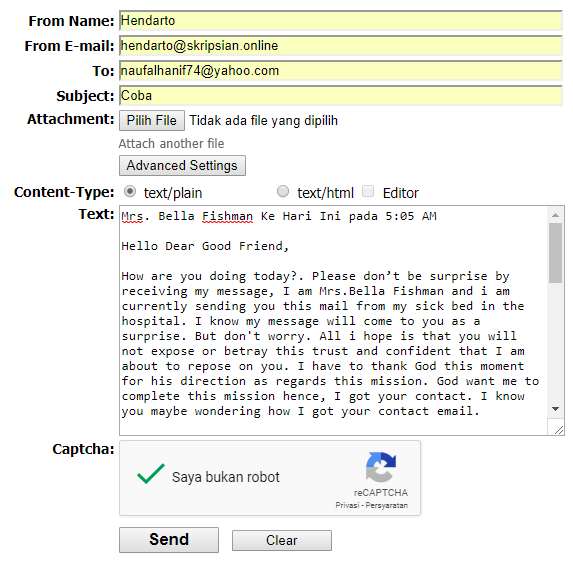
*Email spam* tersebut diatas diblokir oleh *Amavisd-New* karena terindikasi sebagai *email spam* oleh *SpamAssassin*, hasil pem*filter*an *email spam* dapat dilihat pada *mail log* dengan menggunakan perintah *#cat /var/log/maillog* seperti terlihat pada gambar 4.68 berikut.



Gambar 4.68 *Email* dari *Gmail* Terindikasi *Spam*

###### **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 4.69 berikut.



Gambar 4.69 *Email* Dengan Format *Spam*

*Email* dengan *format spam* tersebut terkirim ke *Yahoo! Mail* dan masuk ke dalam *folder spam* seperti terlihat pada gambar 4.70 berikut.



Gambar 4.70 *Email* Terindikasi Sebagai *Spam* oleh *Yahoo! Mail*

*Yahoo! Mail* mengindikasi bahwa *email* tersebut diatas merupakan *spam* sehingga *format email* tersebut diatas dapat dijadikan acuan untuk melakukan uji coba mengirim *email spam* tanpa *GTUBE test* dengan cara mengirim *email* seperti *format email* tersebut diatas dari *Emkei’s Fake Mailer* ke salah satu *user* yang ada pada layanan *email* skripsian.online seperti terlihat pada gambar 4.71 berikut.

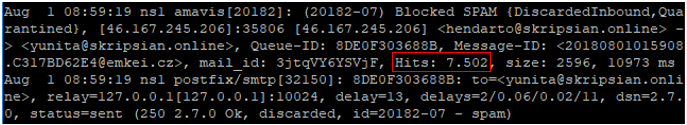


Gambar 4.71 Mengirim *Email Spam* Tanpa *Format GTUBE Test*

*Email spam* tanpa *format GTUBE Test* tersebut terindikasi sebagai *spam* oleh *mail server* skripsian.online karena memiliki skor 7,502, sedangkan skor yang dapat ditoleransi oleh *SpamAssassin* adalah 5,0 (*default score*) sehingga *email* yang memiliki skor lebih besar atau sama dengan 5,0 akan langsung diblokir karena terindikasi sebagai *spam* dan *email* yang memiliki skor lebih rendah dari 5,0 akan masuk pada kotak masuk karena dianggap sebagai *email ham* seperti terlihat pada gambar 4.72 dan 4.73 berikut.



Gambar 4. 72 *Default Score SpamAssassin*



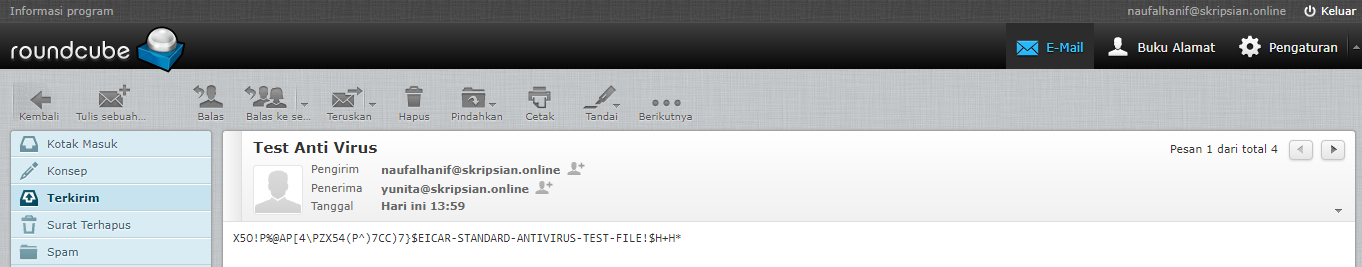
Gambar 4.73 Skor *Email* yang Terindikasi Sebagai *Spam*

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

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

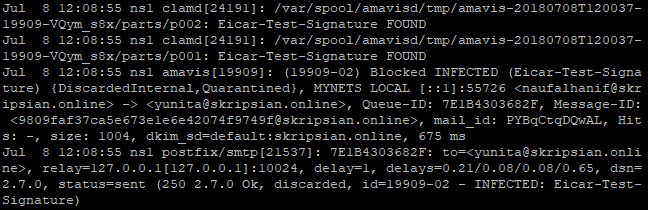
###### **Uji Coba Mengirim *Email* yang Mengandung *Virus* dari skripsian.online**

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* yang mengandung *virus* dikirim dari layanan *email* skripsian.online ke layanan *email* skripsian.online, seperti pada gambar 4.74 berikut.



Gambar 4.74 *EICAR Test* dari skripsian.online Setelah Penerapan

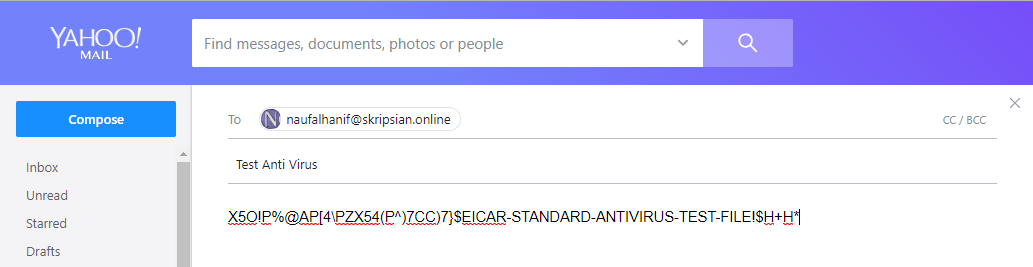
Setelah *email* yang mengandung *virus* tersebut dikirim ke salah satu *user email* yang ada pada *mail server* skripsian.online maka *email* tersebut akan di*block* oleh *Amavisd-New* karena *email* tersebut telah terdeteksi mengandung *virus* oleh *ClamAV* yang merupakan *anti virus mail server* skripsian.online. cara untuk mengecek bahwa *email* tersebut telah terblok adalah dengan membuka *mail log server* dengan perintah *#cat /var/log/maillog*, seperti terlihat pada gambar 4.75 berikut.



Gambar 4.75 *Email* dari skripsian.online Terblok

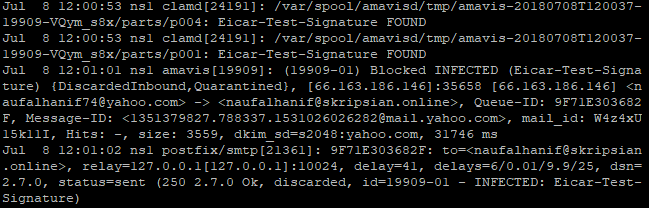
###### **Uji Coba Mengirim *Email* yang Mengandung *Virus* dari *Yahoo! Mail***

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* skripsian.online, seperti pada gambar 4.76 berikut.



Gambar 4.76 *EICAR Test* dari *Yahoo! Mail* Setelah Penerapan

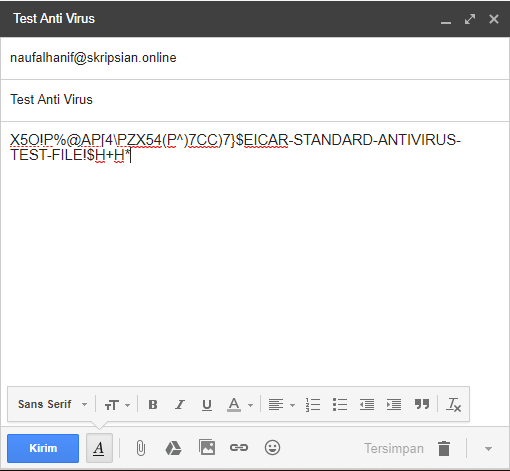
Setelah *email* yang mengandung *virus* tersebut di kirim pada salah satu *user* *email* yang ada pada *mail server* skripsian.online maka *email* tersebut akan di*block* oleh *Amavisd-New* karena *email* tersebut telah terdeteksi mengandung *virus* oleh *ClamAV* yang merupakan *anti virus mail server* skripsian.online. cara untuk mengecek bahwa *email* tersebut telah terblok adalah dengan membuka *mail log server* dengan perintah *#cat /var/log/maillog*, seperti terlihat pada gambar 4.77 berikut.



Gambar 4.77 *Email* dari *Yahoo! Mail* Terblok

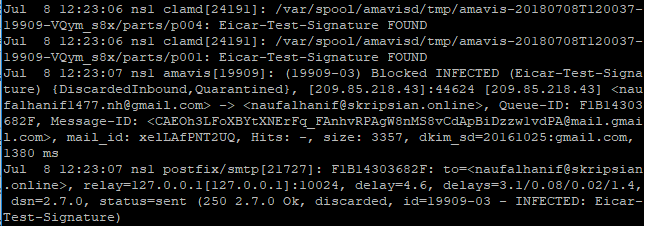
###### **Uji Coba Mengirim *Email* yang Mengandung *Virus* dari *Gmail***

Uji coba mengirim *email* yang mengandung *virus* dilakukan dengan mengirim *email* dengan isi X5O!P%@AP[4\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* di kirim dari layanan *email Gmail Mail* ke layanan *email* skripsian.online, seperti pada gambar 4.78 berikut.



**Gambar 4.78 *EICAR Test* dari *Gmail* Setelah Penerapan**

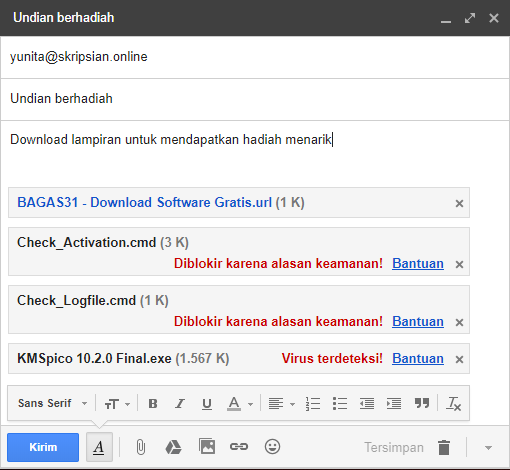
Setelah *email* yang mengandung *virus* tersebut di kirim pada salah satu *user* *email* yang ada pada *mail server* skripsian.online maka *email* tersebut akan di*block* oleh *Amavisd-New* karena *email* tersebut telah terdeteksi mengandung *virus* oleh *ClamAV* yang merupakan *anti virus mail server* skripsian.online. cara untuk mengecek bahwa *email* tersebut telah terblok adalah dengan membuka *mail log server* dengan perintah *#cat /var/log/maillog*, seperti terlihat pada gambar 4.79 berikut.



Gambar 4.79 *Email* dari *Gmail* Terblok

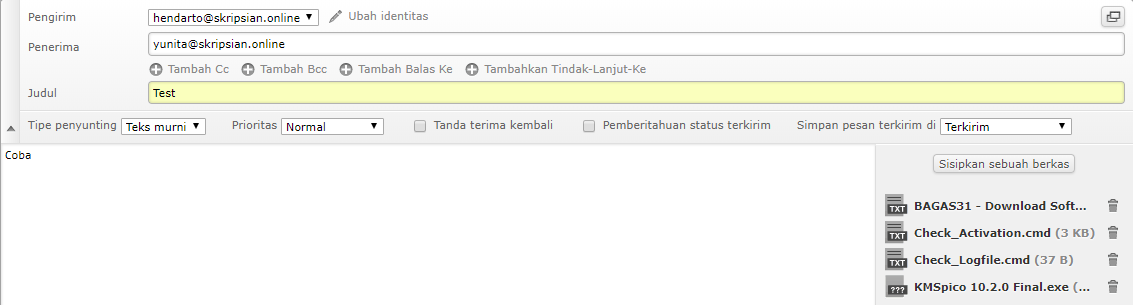
###### **Uji Coba Mengirim *Email Virus* Tanpa Menggunakan *EICAR Test***

Uji coba mengirim *email* yang mengandung *virus* dilakukan dengan mengirim *email* dari layanan *email gmail*, *email* yang dikirim diberi lampiran berupa program *crack* dengan ekstensi .exe yang akan terdeteksi sebagai *virus* oleh *gmail* seperti terlihat pada gambar 4.80 berikut.



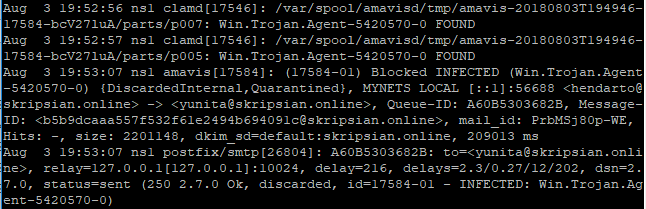
Gambar 4.80 Lampiran Terdeteksi Sebagai *Virus*

Lampiran yang telah terdeteksi sebagai *virus* oleh layanan *email gmail* akan dikirim ke layanan *email* skripsian.online untuk menguji layanan *email* skripsian.online sebelum penerapan *anti virus* seperti terlihat pada gambar 4.81 berikut.



Gambar 4.81 Mengirim *Virus* Setelah Penerapan *Anti Virus*

*Email* yang mengandung *virus* tersebut diblokir oleh *Amavisd-New* karena terdeteksi mengandung *virus*. seperti terlihat pada gambar 4.82 berikut.



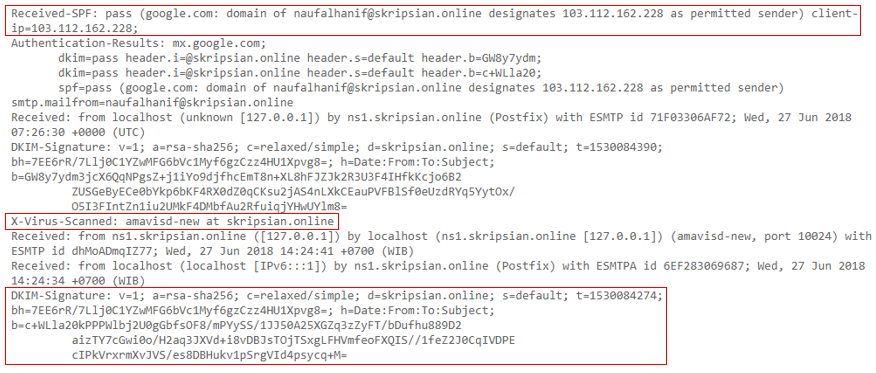
Gambar 4.82 *Email* Terdeteksi Mengandung *Virus*

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

Uji coba pengecekan *header email* dilakukan dengan membandingkan *header email* yang di kirim dari skripsian.online ke *Gmail, Yahoo! Mail*, dan skripsian.online sebelum dan setelah penerapan *DKIM, SPF, anti spam*, dan *anti virus*.

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

Uji coba ini dilakukan dengan mengirim *email* menggunakan salah satu *user email* yang ada pada layanan *email* skripsian.online ke salah satu *user email* yang ada pada layanan *email* *Gmail* kemudian melakukan pengecekan *header email* tersebut dan melakukan perbandingan terhadap *header email* sebelum dan setelah penerapan *DKIM*, *SPF, anti spam,* dan *anti virus*, *header email* setelah diterapkannya *DKIM*, *SPF, anti spam,* dan *anti virus* terlihat seperti gambar 4.83 berikut.



Gambar 4.83 Cuplikan *Header Email* pada *Gmail* Setelah Penerapan

Pada gambar 4.62 terlihat perbedaan *header email* setelah penerapan *DKIM, SPF*, dan *anti virus* yaitu terdapat tambahan parameter *X-Virus-Scanned*, dan *DKIM-Signature*.

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

Uji coba ini dilakukan dengan mengirim *email* menggunakan salah satu *user email* yang ada pada layanan *email* skripsian.online ke salah satu *user* yang ada pada layanan *email Yahoo! Mail* kemudian melakukan pengecekan *header email* tersebut dan melakukan perbandingan terhadap *header email* sebelum dan setelah penerapan *DKIM*, *SPF, anti spam,* dan *anti virus*, *header email* setelah diterapkannya *DKIM*, *SPF, anti spam,* dan *anti virus* terlihat seperti gambar 4.84 berikut.



Gambar 4. 84 Cuplikan *Header Email* pada *Yahoo! Mail* Setelah Penerapan

Pada gambar 4.84 terlihat perbedaan *header email* setelah penerapan *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* (*OK*) dan terdapat tambahan parameter *DKIM-Signature* dan *X-Virus-Scanned*.

###### ***Header Email* pada skripsian.online**

Uji coba ini dilakukan dengan mengirim *email* menggunakan salah satu *user email* yang ada pada layanan *email* skripsian.online ke salah satu *user email* yang ada pada layanan *email* skripsian.online kemudian melakukan pengecekan *header email* tersebut dan melakukan perbandingan terhadap *header email* sebelum dan setelah penerapan *DKIM*, *SPF, anti spam,* dan *anti virus*, *header email* setelah diterapkannya *DKIM*, *SPF, anti spam,* dan *anti virus* terlihat seperti gambar 4.85 berikut.



Gambar 4.85 Cuplikan *Header Email* skripsian.onlineSetelah Penerapan

Pada gambar 4.85 terlihat perbedaan *header email* setelah penerapan *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.

## **Analisa Hasil Uji Coba**

Pada tahap ini akan dilakukan analisa hasil uji coba yang telah di lakukan sebelumnya. Pada analisa hasil uji coba akan di tampilkan analisa hasil uji coba pengiriman *email spoofing* sebelum dan setelah penerapan *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 *DKIM*, *SPF*, *anti spam*, dan *anti virus.*

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

Cara yang dapat digunakan untuk mengetahui apakah sudah dilakukan proses otorisasi dan otentikasi oleh protokol *DKIM* dan *SPF* adalah dengan melakukan pengiriman *email spoofing* menggunakan *Emkei’s Fake Mailer* dengan mengatasnamakan salah satu *user email* pada *mail server* skripsian.online, kemudian *email* tersebut dikirim ke layanan *email Gmail*, *Yahoo! Mail*, dan skripsian.online. Berikut analisa hasil ujicoba perbandingan sebelum diterapkan *DKIM* dan *SPF* dan setelah diterapakan *DKIM* dan *SPF* yang dilakukan pada uji coba sebelumnya, seperti terlihat pada tabel 4.1 berikut.

Tabel 4.1 Perbandingan Sebelum dan Setelah Penerapan *DKIM* dan *SPF*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **NO** | ***Fake Mailer*** | **Layanan *Email* yang diatasnamakan** | **Layanan *Email* Penerima** | **Sebelum Penerapan** | **Setelah Penerapan** |
| 1 | *Emkei’s Fake Mailer* | skripsian.online | *Gmail* | Masuk *Folder Inbox* | Diblokir |
| 2 | *Emkei’s Fake Mailer* | skripsian.online | *Yahoo! Mail* | Masuk *Folder Inbox* | Masuk *Folder Spam* |
| 3 | *Emkei’s Fake Mailer* | skripsian.online | skripsian.online | Masuk *Folder Inbox* | Masuk *Folder Inbox* |

Berdasarkan tabel 4.1 perbandingan sebelum dan setelah penerapan *DKIM* dan *SPF* dengan melakukan pengiriman *email spoofing* yang dikirim menggunakan *Emkei’s Fake Mailer* ke layanan *email* *Gmail*, *Yahoo! Mail*, dan skripsian.online sebelum penerapan *DKIM* dan *SPF* yaitu *email spoofing* berhasil masuk ke *folder inbox* penerima *email* yang berada pada *mail server* *Gmail*, *Yahoo! Mail*, dan skripsian.online sedangkan setelah penerapan *DKIM* dan *SPF*, *email spoofing* tersebut diblokir oleh layanan *email Gmail*, dimasukan ke *folder spam* oleh layanan *email Yahoo! Mail* dan dimasukan ke *folder inbox* oleh layanan *email* skripsian.online.

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

Analisa penerapan *anti spam* dilakukan dengan mengirim *email spam* dengan menggunakan layanan *email* skripsian.online, *Yahoo! Mail*, dan *Gmail* ke layanan *email* skripsian.online untuk menguji kinerja *anti spam* sebelum dan setelah penerapan *anti spam* seperti terlihat pada tabel 4.2 berikut.

Tabel 4.2 Perbandingan Sebelum dan Setelah Penerapan *Anti Spam*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **NO** | **Layanan *Email* Pengirim** | **Layanan *Email* Penerima** | **Sebelum Penerapan** | **Setelah Penerapan** |
| 1 | *Yahoo! Mail* | skripsian.online | Masuk *Folder Inbox* | Diblokir |
| 2 | *Gmail* | skripsian.online | Masuk *Folder Inbox* | Diblokir |
| 3 | skripsian.online | skripsian.online | Masuk *Folder Inbox* | 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* skripsian.online, 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*.

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

Analisa penerapan *anti virus* dilakukan dengan mengirim *email spam* dengan menggunakan layanan *email* skripsian.online, *Yahoo! Mail*, dan *Gmail* ke layanan *email* skripsian.online untuk menguji kinerja *anti spam* sebelum dan setelah penerapan *anti spam* seperti terlihat pada tabel 4.3.

Tabel 4.3 Perbandingan Sebelum Penerapan *Anti virus*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **NO** | **Layanan *Email* Pengirim** | **Layanan *Email* Penerima** | **Sebelum Penerapan** | **Setelah Penerapan** |
| 1 | *Yahoo! Mail* | skripsian.online | Masuk *Folder Inbox* | Diblokir |
| 2 | *Gmail* | skripsian.online | Masuk *Folder Inbox* | Diblokir |
| 3 | skripsian.online | skripsian.online | Masuk *Folder Inbox* | Diblokir |

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* skripsian.online, sedangkan setelah penerapan *anti virus*, terjadi pemblokiran *email* yang mengandung *virus* oleh *Amavisd-New* sehingga *email* yang terindikasi mengandung *virus* langsung diblokir sebelum sampai pada *folder* penerima *email*.

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

Analisa pengecekan *header email* dilakukan dengan melihat *header email* sebelum dan setelah penerapan *DKIM*, *SPF*, *anti spam*, dan *anti virus*. Perbedaan *header email* sebelum dan setelah diterapkan *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** | **X-Virus-Scanned** | **Received-SPF** |
| 1 | Sebelum Penerapan | Gmail | Tidak Ada | Tidak Ada | Pass |
| Yahoo! Mail | Tidak Ada | Tidak Ada | None |
| skripsian.online | Tidak Ada | Tidak Ada | Tidak Ada |
| 2 | Setelah Penerapan | Gmail | Ada | Ada | Pass |
| Yahoo! Mail | Ada | Ada | Pass |
| skripsian.online | Ada | Ada | Tidak Ada |

Berdasarkan tabel 4.4, sebelum penerapan *DKIM*, *SPF*, *anti spam*, dan *anti virus* tidak terdapat parameter *DKIM-Signature* dan *X-Virus-Scanned*, namun *Received-SPF* sudah bernilai *Pass* pada *header email* di *Gmail*, sedangkan setelah penerapan *DKIM*, *SPF*, *anti spam*, dan *anti virus* terdapat parameter *DKIM-Signature* dan *X-Virus-Scanned*, serta *Received-SPF* bernilai *Pass* pada *header email* di *Gmail*.

Sebelum penerapan *DKIM*, *SPF*, *anti spam*, dan *anti virus* tidak terdapat parameter *DKIM-Signature* dan *X-Virus-Scanned*, serta *Received-SPF* bernilai *none* pada *header email* di *Yahoo! Mail*, sedangkan setelah penerapan *DKIM*, *SPF*, *anti spam*, dan *anti virus* terdapat parameter *DKIM-Signature* dan *X-Virus-Scanned*, serta *Received-SPF* bernilai *Pass* pada *header email* di *Yahoo! Mail*.

Sebelum penerapan *DKIM*, *SPF*, *anti spam*, dan *anti virus* tidak terdapat parameter *DKIM-Signature*, *X-Virus-Scanned,* dan *SPF-Received* pada *header email* di skripsian.online, sedangkan setelah penerapan *DKIM*, *SPF*, *anti spam*, dan *anti virus* terdapat parameter *DKIM-Signature* dan *X-Virus-Scanned*, namun tetap tidak terdapat parameter *Received-SPF* pada *header email* di skripsian.online.

# **BAB V**

# **PENUTUP**

## **Kesimpulan**

Berdasarkan hasil ujicoba 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 *SpamAssassin*, *ClamAV*, dan *Amavisd-New* dapat mencegah masuknya *email spam* dan *virus* dengan cara melakukan pengecekan *header*, *body*, dan *attachment email*.

## **Saran**

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

1. Mengembangkan sistem otentikasi dan otorisasi *email spoofing* dengan menambahkan protokol *DMARC*.
2. 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.
3. Mengembangkan sistem *anti spam* dengan menambahkan *tools anti spam* lainnya seperti *Barracuda Central*, *Spamhaus*, *SpamCop*, *SORBS*, dan lain-lain.

# **DAFTAR REFERENSI**

Ardiantoro, T., Triyono, J., Fatkhiyah, E. (2016). Optimalisasi rancangan jaringan komputer menggunakangoogle sketchup. Jurnal JARKOM, 4, 81-88.

Arrington, M. (2006), Internet. *Single Ajax Interface for Yahoo Mail & IM Coming*. Diperoleh Juli 22, 2018, dari https://www.techcrunch.com/2006/11/09/single-ajax-interface-for-yahoo-mail-im-coming/

Barovih, G. (2011). Desain dan implementasi mail server berbasis webbeserta pengamannya pada PT. PLN (persero) sektorpembangkitan bukit asamtanjung enim. Sekolah Tinggi Manajemen Informatika dan Komputer PalComTech Palembang. 1-10.

Chandra, W. N., Indrawan, G., Sukajaya, I. N. (2016). *Spam filtering* dengan metode *pos tagger* dan klasifikasi *naïve**bayes*. Jurnal Ilmiah Teknologi dan Informasia ASIA, 10, 47-55.

CentOS Web Panel. (n.d.), Internet. *Control Web Panel*. Diperoleh 30 Juni, 2018, dari http://centos-webpanel.com/

Crocker, D. (2009). *Internet Mail Architecture*. *Request for Comments: 5598*, 1-39.

Desmira., Sumarto, D., Yuliani, R. (2017). Rancang bangun mail server berbasis squirrelmailmenggunakan MTA (Mail Transfer Agent)pada PT. Teras Inti Media. Jurnal PROSISKO, 4, 55-59.

Emkei’s Mailer, Internet. *Free Online Fake Mailer with Attachments, Encryption, HTML Editor and Advanced Settings*. Diperoleh Juni 20, 2018, dari https://emkei.cz/

Fitriani, R. E., Riyono, A. (2015). Mendesain keamanan sistem jaringan. Jurnal Ilmiah, 1-21.

Google. (n.d.), Internet. Gmail. Diperoleh 21 Juni, 2018, dari https://www.google.com/gmail/

Hansen, T., Crocker, D., Baker, P. H. (2009). *DomainKeys Identified Mail (DKIM) Service Overview*. *Request for Comments: 5585*, 1-24.

Harjono, E. B. (2016). Analisa dan implementasi dalam membangun sistem operasi linuxmenggunakan metode *LSF* dan *remaster*. Jurnal & Penelitian Teknik Informatika, 1, 30-35.

Haryanto, M. D., Riadi, I. (2014). Analisis dan optimalisasi jaringanmenggunakan teknik load balancing(studi kasus: jaringan UAD kampus 3). Jurnal Sarjana Teknik Informatika, 2, 1370-1378.

Hayuningtyas, R. Y. (2017). Aplikasi filtering of spam email menggunakan naïvebayes. *Indonesian Journal on Computer and Information Technology*, 2, 53-60.

Hidayat, W. N., Pangera, A. A. (2010). Membangun *mail server* berbasis linux menggunakn *postfix*dengan *client* squirrelmail. Sekolah Tinggi Manajemen Informatika dan KomputerAmikomYogyakarta.

Hoiriyah., Sugiantoro, B., Prayudi, Y. (2016). Investigasi forensik pada e-mail spoofing menggunakan metodeheader analysis. Jurnal Ilmiah DASI, 17, 20-25.

Kader, D. P., Najoan, M. E. I., Sinsuw, A. A. E. (2014). Analisa performansi algoritma *routing* di jaringan komputer UNSRAT. *E-journal* Teknik Elektro dan Komputer, 28-39.

Kusmaya. (2016). Implementasi mail server menggunakan *postfix*. Jurnal Informasi, 8, 49-69.

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

Masero, A. P., Triyono, J., Andayati, D. (2013). Perancangan pengelolaan jaringan *IT* pada institut sains & teknologiAKPRIND menggunakan teknologi *VPN* (Virtual Private Network). Jurnal JARKOM, 1, 20-30.

Mawarsih, A. (2014). Pengaruh *electronic mail* sebagai media komunikasiterkadap mengerjakan tugas kuliah mahasiswa. *Ejournal* Ilmu Komunikasi, 2, 334-348.

Muarif, M. I., Irwan, D. (2017). Sistem *auto backup* elektronik *mail* pada *mail server*menggunakan *cron job*. Jurnal Penelitian Ilmu Komputer, 5, 79-90.

Nurfajar, A., Kurniawan, M. T., Hediyanto, U. Y. K. S. (2015). Desain dan analisa infrastruktur jaringan wired di PDII-LIPIjakarta dengan menggunakan metode Network Development LifeCycle (*NDLC*). *e-Proceeding of Engineering*, 2, 5359-5365.

Nurlina., Irmayana. (2014). Studi banding *spam-assassin mail server*dengan dan tanpa *filter* di sisi *mail client*. *Citec Journal*, 1, 77-88.

Pratama, A. M. R. (2008). Perancangan dan implementasi *mail server* berbasis *qmail*pada *jcpanel web hosting control panel*. Seminar Nasional Aplikasi Teknologi Informasi 2008, 1-7.

Riadi, I. (2011). Optimalisasi keamanan jaringanmenggunakan pem*filter*an aplikasiberbasis mikrotik. JUSI, 1, 71-80.

Sadikin, N. (2014). Implementasi *e-mail server* terdistribusipada jaringan *Local Area Network (LAN)*dan *Wide Area Network (WAN)*. Seminar Nasional Teknologi Informasi dan Multimedia 2014, 7-12.

Saputra, A., Syafrizal, M. (2012). Perancangan dan implementasi *mail server* pada CV. SanjayaAnugerah Sejahtera (*ISP* Jogjaringan) berbasis *open source*. Jurnal DASI, 13, 1-6.

Sujana, A. P. (2014). Perangkat pendukung forensik lalu lintas jaringan. Jurnal Teknik Komputer Unikom, 3, 31-37.

Suryana, A. L., Akbar, R. R. E., Widiyasono, N. (2016). Investigasi email spoofing dengan metodeDigital Forensics Research Workshop (*DFRWS*). Jurnal Edukasi dan Penelitian Informatika, 2, 112-117.

The Roundcube Team. (n.d.), Internet. *About the Roundcube webmail project*. Diperoleh 30 Juni, 2018, dari <https://roundcube.net/about/>

Valsecchi, P. (2013), Internet. *Secure Postfix with Amavisd, ClamAV, SpamAssassin*. Diperoleh Juni 20, 2018, dari https://nolabnoparty.com/en/secure-postfix-amavisd-clamav-spamassassin/

Wardoyo, S., Ryadi, T., Fahrizal, R. (2014). Analisis performa file transport protocol padaperbandingan metode IPv4 murni, IPv6 murni dantunneling 6to4 berbasis router mikrotik. Jurnal Nasional Teknik Elektro, 3, 106-117.

Wicitra, A., Utomo, D., Wardana, H. K. (2014). Membangun infrastruktur komputasi awanprivat single cluster dan multi cluster denganmenggunakan *linux CentOS*. Techné Jurnal Ilmiah Elektroteknika, 13, 185-194.

Zabar, A. A., Novianto, F. (2015). Keamanan *HTTP* dan *HTTPS* berbasis web menggunakansistem operasi kali linux. Jurnal Ilmiah Komputer dan Informatika, 4, 69-74.

Zimbra a Synacor Product. (2005), Internet. *Best Practices on Email Protection: SPF, DKIM and DMARC*. Diperoleh 22 Juni, 2018, dari <https://wiki.zimbra.com/wiki/Best_Practices_on_Email_Protection:_SPF,_DKIM_and_DMARC>

# **LAMPIRAN A *HEADER EMAIL***

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

Delivered-To: [naufalhanif1477.nh@gmail.com](mailto:naufalhanif1477.nh@gmail.com)

Received: by 2002:a9d:1b90:0:0:0:0:0 with SMTP id z16-v6csp2930798otd; Thu, 24 May 2018 20:14:27 -0700 (PDT)

X-Google-Smtp-Source: AB8JxZrCk7AjzJqbCSSgEp+7rgOmLXunmoGj1lvVEo1FspoHrtYpNYELxfl8LaWU6ok+6R73tPiL

X-Received: by 2002:a63:77c9:: with SMTP id s192-v6mr563891pgc.140.1527218067862; Thu, 24 May 2018 20:14:27 -0700 (PDT)

ARC-Seal: i=1; a=rsa-sha256; t=1527218067; cv=none; d=google.com; s=arc-20160816; b=e585ODLYx1AK4rTlEGJsT68AAYr8whzBt/hsWmU2elV5FCUbhEmHYrWQjpdetXuHL177lGuPOYu1DLo2lwmwu81fNMmFWtyZXzeQiKgil/xeAPSb0p1mk6wwYI2jOtOagA6VTkrDD4S0tdPMY3aiDXE1Ar7J5vAxO3JFgzxBdpMxPG9/uXvaytGItmBovM4qPH8wNBSv5hxpcxNBvAkmTpbDlpShGfeKTMfHGQU6FIn1WqAdqIJEd0WUhgSoDC+Ce1RcqrOLgRdFzOLFRMHqru/g6Yan/CUtnvDofpeJeBXrOaWOQfMdPIr8w==

ARC-Message-Signature: i=1; a=rsa-sha256; c=relaxed/relaxed; d=google.com; s=arc-20160816; h=user-agent:message-id:subject:to:from:date:content-transfer-encoding:mime-version:arc-authentication-results; bh=5JtrnNw9RJpLzzBAVfY5dy+L5y+H+RNH4s8DSbVA2Ek=; b=gP0Me9Pz98ZG2UgGIwWWzPGpiRuMB4l0sm99JEpviu9f+u1d3UX6kUVcK6Ktgt2ka21xUOq1ynPm5ujsmTtUL1JYp1gADT5etLH8Zd0JLeqXGAlIxNWg0PloHg5FmvHzX5jSzBzvRevKfLlKl8aYrnr3sKQyzo+L3RuSnplzTmiTPOqjqDefB4ZFjQ6Y27SU1deG6szzPtO/t+kfUKb39M8lyOgiAYEZuoUWsSopcep80X5+F6nvySKR67AgjPSIw9jZC0Lt6ADlEarq6oyb7FQbOwujs9ztMT0Isct7WbqhsCPucin4ct9XXHxz8Hxic3z1yJPbYBPNpMhBTCHCOQ==

ARC-Authentication-Results: i=1; mx.google.com; spf=pass (google.com: best guess record for domain of hendarto@skripsian.online designates 103.112.162.164 as permitted sender)

Return-Path: <[hendarto@skripsian.online](mailto:hendarto@skripsian.online)>

Received: from ns1.skripsian.online (ns1.skripsian.online. [103.112.162.164]) by mx.google.com with ESMTPS id l3-v6si22345139pld.96.2018.05.24.20.14.26 for <[naufalhanif1477.nh@gmail.com](mailto:naufalhanif1477.nh@gmail.com)> (version=TLS1\_2 cipher=ECDHE-RSA-AES128-GCM-SHA256 bits=128/128); Thu, 24 May 2018 20:14:27 -0700 (PDT)

Received-SPF: pass (google.com: best guess record for domain of hendarto@skripsian.online designates 103.112.162.164 as permitted sender) client-ip=103.112.162.164; Authentication-Results: mx.google.com; spf=pass (google.com: best guess record for domain of hendarto@skripsian.online designates 103.112.162.164 as permitted sender) [smtp.mailfrom=hendarto@skripsian.online](mailto:smtp.mailfrom=hendarto@skripsian.online)

Received: from localhost (localhost [IPv6:::1]) by ns1.skripsian.online (Postfix) with ESMTPA id 77ED630535C8 for <naufalhanif1477.nh@gmail.com>; Fri, 25 May 2018 10:13:57 +0700 (WIB)

MIME-Version: 1.0

Content-Type: text/plain; charset=US-ASCII; format=flowed

Content-Transfer-Encoding: 7bit

From: [hendarto@skripsian.online](mailto:hendarto@skripsian.online)

To: [naufalhanif1477.nh@gmail.com](mailto:naufalhanif1477.nh@gmail.com)

Subject: Cek Header Email

Message-ID: [4af09547044f1ef8842fd8e66b388f48@skripsian.online](mailto:4af09547044f1ef8842fd8e66b388f48@skripsian.online)

X-Sender: [hendarto@skripsian.online](mailto:hendarto@skripsian.online)

User-Agent: Roundcube Webmail/1.2.3

Melakukan pengecekan header email

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

Delivered-To: [naufalhanif1477.nh@gmail.com](mailto:naufalhanif1477.nh@gmail.com)

Received: by 2002:a9d:2005:0:0:0:0:0 with SMTP id n5-v6csp3905089ota; Wed, 27 Jun 2018 00:26:30 -0700 (PDT)

X-Google-Smtp-Source: ADUXVKLvbd8NK4fhsnz+IQuWVVM8h1geLLx16dzztNDSdcpEEzvt056eVbwLd9ZOI6MbNWTD/FWZ

X-Received: by 2002:a65:418b:: with SMTP id a11-v6mr4169890pgq.118.1530084390645; Wed, 27 Jun 2018 00:26:30 -0700 (PDT)

ARC-Seal: i=1; a=rsa-sha256; d=google.com; s=arc-20160816; b=pMob8JnJiBb78YxyRnTeyk/s9WoS/aJ3am7K0qR8UY+hiaBt/aCcHnhYIZt/2yjq6ZJFdh+M5VA4QBGksqcWF2M5nXTK7YbUtBlC5go9X26DX8cEqtb91JDjBpcfwG4WG3Oc6xdj1td2UGMQPhqh5OQjE6OkId0Jhknirr4NT3qEQekAfSDlMNOA5u9COIzCatFWU+lSsJT4AzA6GwjPx0gnIz1PwkAtDgamuo1OoxvvdmH518W9+IRVynSVal+aioDz2IQ51emhVvuPO2gYeo777sAlH3dE3Eko5UaInr9VJsLdbBk3aZ5F7ltz/hInrooIBnLwqBhFySoedM s/pw==

ARC-Message-Signature: i=1; a=rsa-sha256; c=relaxed/relaxed; d=google.com; s=arc-20160816; h=user-agent:message-id:subject:to:from:date:content-transfer-encoding:mime-version:dkim-signature:dkim-signature:arc-authentication-results; bh=7EE6rR/7Llj0C1YZwMFG6bVc1Myf6gzCzz4HU1Xpvg8=; b=xuhJWKfGtNgte5iUxg197I0jEZrqQzjVQb7+XDJZl0xw86wnOzpDpWgGkscTE0VeeEQtaZqJ8jxhMphJswy81aeLliKfUgBVnERMtgH5d2dSIFdGxvmLGlNe66q0UZM/J+uy7Yxrak7ww/Wk3XKZlaAa+Htca4HYlo7eyxXFRlPoF3TNnVaSr8DxK7nR/90XDefLVcE2Z1joW+Zurlcxoswc1ugwe0dtboDxbd1Wo8gD0y+zK8zxXjTPUxq7S6VTYfKbX/W/sy22sQ1lMyw507pNqP4TAbs+KoU2n6hj57DtsJAana2fddl1Gm8zT9Wpj02pI0ddhOwujgFXTM9OKA==

ARC-Authentication-Results: i=1; mx.google.com; dkim=pass header.i=@skripsian.online header.s=default header.b=GW8y7ydm; dkim=pass header.i=@skripsian.online header.s=default header.b=c+WLla20; spf=pass (google.com: domain of naufalhanif@skripsian.online designates 103.112.162.228 as permitted sender) [smtp.mailfrom=naufalhanif@skripsian.online](mailto:smtp.mailfrom=naufalhanif@skripsian.online)

Return-Path: <[naufalhanif@skripsian.online](mailto:naufalhanif@skripsian.online)>

Received: from ns1.skripsian.online (skripsian.online. [103.112.162.228]) by mx.google.com with ESMTPS id x2-v6si3356094plv.388.2018.06.27.00.26.29 for <[naufalhanif1477.nh@gmail.com](mailto:naufalhanif1477.nh@gmail.com)> (version=TLS1\_2 cipher=ECDHE- Wed, 27 Jun 2018 00:26:29 -0700 (PDT)

Received-SPF: pass (google.com: domain of naufalhanif@skripsian.online designates 103.112.162.228 as permitted sender) client-ip=103.112.162.228;

Authentication-Results: mx.google.com; dkim=pass header.i=@skripsian.online header.s=default header.b=GW8y7ydm; dkim=pass header.i=@skripsian.online header.s=default header.b=c+WLla20; spf=pass (google.com: domain of naufalhanif@skripsian.online designates 103.112.162.228 as permitted sender) [smtp.mailfrom=naufalhanif@skripsian.online](mailto:smtp.mailfrom=naufalhanif@skripsian.online)

Received: from localhost (unknown [127.0.0.1]) by ns1.skripsian.online (Postfix) with ESMTP id 71F03306AF72; Wed, 27 Jun 2018 07:26:30 +0000 (UTC)

DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=skripsian.online; s=default; t=1530084390; bh=7EE6rR/7Llj0C1YZwMFG6bVc1Myf6gzCzz4HU1Xpvg8=; h=Date:From:To:Subject; b=GW8y7ydm3jcX6QqNPgsZ+j1iYo9djfhcEmT8n+XL8hFJZJk2R3U3F4IHfkKcjo6B2ZUSGeByECe0bYkp6bKF4RX0dZ0qCKsu2jAS4nLXkCEauPVFBlSf0eUzdRYq5YytOx/ O5I3FIntZn1iu2UMkF4DMbfAu2RfuiqjYHwUYlm8=

X-Virus-Scanned: amavisd-new at skripsian.online

Received: from ns1.skripsian.online ([127.0.0.1]) by localhost (ns1.skripsian.online [127.0.0.1]) (amavisd-new, port 10024) with ESMTP id dhMoADmqIZ77; Wed, 27 Jun 2018 14:24:41 +0700 (WIB)

Received: from localhost (localhost [IPv6:::1]) by ns1.skripsian.online (Postfix) with ESMTPA id 6EF283069687; Wed, 27 Jun 2018 14:24:34 +0700 (WIB)

DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=skripsian.online; s=default; t=1530084274; bh=7EE6rR/7Llj0C1YZwMFG6bVc1Myf6gzCzz4HU1Xpvg8=; h=Date:From:To:Subject; b=c+WLla20kPPPWlbj2U0gGbfsOF8/mPYySS/1JJ50A25XGZq3zZyFT/bDufhu889D2aizTY7cGwi0o/H2aq3JXVd+i8vDBJsTOjTSxgLFHVmfeoFXQIS//1feZ2J0CqIVDPEcIPkVrxrmXvJVS/es8DBHukv1pSrgVId4psycq+M=

MIME-Version: 1.0

Content-Type: text/plain; charset=US-ASCII; format=flowed

Content-Transfer-Encoding: 7bit

Date: Wed, 27 Jun 2018 14:24:32 +0700

From: [naufalhanif@skripsian.online](mailto:naufalhanif@skripsian.online)

To: Naufal Hanif <naufalhanif1477.nh@gmail.com>, naufalhanif74@yahoo.com, [yunita@skripsian.online](mailto:yunita@skripsian.online)

Subject: Cek Header Email

Message-ID: <15f73f4a3d9cb5af642565d45702f46a@skripsian.online

X-Sender: [naufalhanif@skripsian.online](mailto:naufalhanif@skripsian.online)

User-Agent: Roundcube Webmail/1.2.3

Melakukan Pengecekan Header Email

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

X-Apparently-To: naufalhanif74@yahoo.com; Wed, 27 Jun 2018 05:06:51 +0000

Return-Path: <naufalhanif@skripsian.online>

X-YahooFilteredBulk: 103.112.162.228

Received-SPF: none (domain of skripsian.online does not designate permitted sender hosts)

X-YMailISG: wd\_by.YWLDtGH6UaoekbpNHudL9y6gul1V2y4qcVzHnllhdw

QKd8Y4JT01BiQS6tMpOT6cosy0UmdZ\_joglaxOgjOnTFTpaLXFoZIZrgKlQs

z8QmIyTRGX9GkD402N4cWD38v2SZK1WmNykIhxc0NR5z79memgqtRRLU\_kkT

K65Ft1yjenPfNOhYMiklwhDkDI8kIT7SmNWW4c34.PtuYcw86cglyo8IyHPw

9ItC6279w4fdRINQD5eSqVtX6\_DwpkHbK2eIM34aDTttROsdWFPLGwd300sx

FQiE8ewC.7N65Lvg3gOrYB8S3iLN.sdVNi1j24FRdTVEYfA30dkTL0IxmY.Q

ABXHmFn4vQle0xmkkIt73L9LGGH1HKeryWRqNPLIOLFpgatJbSWnWWeFNCi8

Z3MzonRvHQOAKV7qi9PpIUDDB6khkkLlbouV9MU5PvPhQblfAbhZrasQwRFD

kRDeHj08g0uPDJs0mEKpuvctUEOSXQAQ6HTJwC7SBocF1pHvEeguSl.1Ng.w

o.pWAzLPRxh4ZEqlDRTtYmyN58pDlwEM81kcnzPuZBvVwpwgFy6.Jsy4gKNT

mZlHVL8idEzsYU4kETwiYbh8nCxVs3dVLbWhHSLVeOQZJy7UU2an0Tz.yqBp

DRshO1F9bDRfUOZg1\_M5uFDuAoJfaNgewgeYrnzE9H8YetyEAJLjQCnCSmQc

I9PN8J.8mWzcf2BI8FUnn7bLRGtniGwDWriVFZhOvqi11kAkKJJ.vANl2pWc

CaoGY\_95JlcX7WsmqVUiZziwE13P7QKPrrVyn\_X\_U9UY8BpnC3YiaZcTJTCG

cLaXhYvYQCoM7W3ePDLEwAFq9tBNoMI5jZ1AMHcIL9XwTWBh2iHGsatCe3Jn

\_KXq9uqhPSL3kQvhzHAMLW8MpXcRNOWIjg--

X-Originating-IP: [103.112.162.228]

Authentication-Results: mta4243.mail.ne1.yahoo.com from=skripsian.online; domainkeys=neutral (no sig); from=skripsian.online; dkim=neutral (no sig)

Received: from 127.0.0.1 (EHLO ns1.skripsian.online) (103.112.162.228) by mta4243.mail.ne1.yahoo.com with SMTPS; Wed, 27 Jun 2018 05:06:50 +0000

Received: from localhost (localhost [IPv6:::1]) by ns1.skripsian.online (Postfix) with ESMTPA id 370623069661; Wed, 27 Jun 2018 12:06:51 +0700 (WIB)

MIME-Version: 1.0

Content-Type: text/plain; charset=US-ASCII; format=flowed

Content-Transfer-Encoding: 7bit

Date: Wed, 27 Jun 2018 12:06:51 +0700

From: naufalhanif@skripsian.online

To: Naufal Hanif <naufalhanif1477.nh@gmail.com>, [naufalhanif74@yahoo.com](mailto:naufalhanif74@yahoo.com), yunita@skripsian.online

Subject: Cek Header Mail

Message-ID: <5a808340b5c6f19cc349b0198c116a1f@skripsian.online>

X-Sender: naufalhanif@skripsian.online

User-Agent: Roundcube Webmail/1.2.3

Content-Length: 33

Uji coba pengecekan header email

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

X-Apparently-To: naufalhanif74@yahoo.com; Fri, 25 May 2018 08:53:57 +0000

Return-Path: <yunita@skripsian.online>

X-YahooFilteredBulk: 103.112.162.164

Received-SPF: pass (domain of skripsian.online designates 103.112.162.164 as permitted sender)

X-YMailISG: ffFxpsMWLDu.4FqQJakpVVaJoFeW\_P3ygeM2RLljE4o5Tr\_F

\_Db9mnVz3MDqu4E3\_m1lltQB0NTlupgKveyN0D92rGlz7nfdQJzVXzoYX4ZT

ei3pww.80HGftoYT0ykpLvaGZwkrz2UU0NUJCPXoHx5V0\_cS85HKV\_7sQu\_l

sZKWFq21WqmiK83INbViHAQoKzySzzAKFcw5GPm9zHJdL2ACBEi9fpTf7F4K

51qunt.ZsonrFTIpnk\_gUffrNHin\_XaTyuc9.qquDarZ0JqZddlvaIQ48OeJ

o88MBZPyfV6oG9wt6.e7G.Yl5yW7\_RTUQzi0M4GCvLyKC0d34QlLneVdMUxI

QOPMhOXAyrMeafehQRPoX22\_dL227yPzcoa16CEw35XPeS9BHrAiHBJqWzY6

deWJLl.bcD5pd7MEg0P\_XX9E5nvxMeLC0PCEzovUUhXvjKHQ61aI5v7oDLHm

cZRISC5RjMs.U6M3y1TesE41ptgxQ4ceqNb801E5Evt2HmjCiBgegyT6yZmE

GG6Zu9Igd5hZwfLucwAef5YGCSZCaJwpQmPqyvJ7jT7Yiso\_TcOoztPAgPH0

h.w.3D9X4fPOXjdO30FX27Qqh2dh9c5Btdls24\_hKR6ET5WjX5ZcJrRI2maU

Mk.HTfUomATFfKmio3PTiCrEwkg1E1qJu8xZSr.PzOHluUm7PqJ\_wmavYiVk

glrGR9QZMGSZMJ5aOejshs5c10I14X.fXKfLuhkDK9g.LjPx7fnIdacXb5Vu

t5VJ6ijM76.46xPzyvNodrrSK3JE\_2N0YSXGJlH5jVbLc2QtYKsjeLRRDQhM

qCc.xxtDptKSbNYaVPiKYRiO\_5CZGRKqylM83hf.OUj0d8K7ez9\_\_GrV7SXl

7U0Ns67imBGtWhu6WxcEyQu.3Z8oKTePyE2D5Rf6rUw9XTCvWHN5aHiNIA\_u

T3LBrHovUE0Md5SQBquBckVTedrShRhGKzvgrymFyOMRUhLyLO5UiqNj1u1\_

lIJkvbqf0nanftQeOET2QupfgcJCUR2hlH2A0Twl0l516CPIr.9838.hiLCf

HBnCUtIGlsAXUw--

X-Originating-IP: [103.112.162.164]

Authentication-Results: mta4247.mail.gq1.yahoo.com from=skripsian.online; domainkeys=neutral (no sig); from=skripsian.online; dkim=pass (ok)

Received: from 127.0.0.1 (EHLO ns1.skripsian.online) (103.112.162.164) by mta4247.mail.gq1.yahoo.com with SMTPS; Fri, 25 May 2018 08:53:56 +0000

Received: from localhost (unknown [127.0.0.1]) by ns1.skripsian.online (Postfix) with ESMTP id B251D30535CB for <naufalhanif74@yahoo.com>; Fri, 25 May 2018 08:53:32 +0000 (UTC)

DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=skripsian.online; s=default; t=1527238412; bh=IHq60PwVrrx751B/xqUvHPs4OMlToRpO/EFxCboljFk=; h=Date:From:To:Subject; b=QzXtxy5pDfeBe8Z5uPfzzvKJj5ZI+5DzUcASE1aK41YDxl4MQsbz4MOogyEyYWiDavY7XMigCx1UT2Wn26eTQsA5g/4rXJmEsKwoYXHCbKidqzF8BVToj7czj71Fkd/X9aag2bVxq5UBoRcF09ZGdkc7nB4jELC5HeNM74orcmA=

X-Virus-Scanned: amavisd-new at skripsian.online

Received: from ns1.skripsian.online ([127.0.0.1]) by localhost (ns1.skripsian.online [127.0.0.1]) (amavisd-new, port 10024) with ESMTP id kFD1twHtwkMA for <naufalhanif74@yahoo.com>; Fri, 25 May 2018 15:53:26 +0700 (WIB)

Received: from localhost (localhost [IPv6:::1]) by ns1.skripsian.online (Postfix) with ESMTPA id 6056430535C3 for <naufalhanif74@yahoo.com>; Fri, 25 May 2018 15:53:26 +0700 (WIB)

DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=skripsian.online;s=default; t=1527238406; bh=IHq60PwVrrx751B/xqUvHPs4OMlToRpO/EFxCboljFk=; h=Date:From:To:Subject; b=gGQZBtBRyer1QIWazoCb7vMFyelGwH6D/f3HjqHNIZ7uVAWZdnvTNqRFoXRGvwFHaiFuEnAAPQjfY33x9yd3/xpce9LXgRNY/zgiBKo7JtCiT6JeXJ5bYVimHp9L9VE+ihMfrsz5KfPeuaWgySPK364Yw3WsOZXOEK9s3AtriK0=

MIME-Version: 1.0

Content-Type: text/plain; charset=US-ASCII;

format=flowed

Content-Transfer-Encoding: 7bit

Date: Fri, 25 May 2018 15:53:26 +0700

From: yunita@skripsian.online

To: naufalhanif74@yahoo.com

Subject: Cek Header Mail

Message-ID: <b3ab19d7e9a37d85ba586b7ea5752241@skripsian.online>

X-Sender: yunita@skripsian.online

User-Agent: Roundcube Webmail/1.2.3

Content-Length: 48

Cek header email setelah penerapan DKIM dan SPF

***Header Email* pada skripsian.online Sebelum Penerapan**

Return-Path: <yunita@skripsian.online>

Delivered-To: naufalhanif@skripsian.online

Received: from localhost (localhost [IPv6:::1]) by ns1.skripsian.online (Postfix) with ESMTPA id 1B948306969F for <naufalhanif@skripsian.online>; Tue, 26 Jun 2018 16:15:08 +0700 (WIB)

MIME-Version: 1.0

Content-Type: text/plain; charset=US-ASCII; format=flowed

Content-Transfer-Encoding: 7bit

Date: Tue, 26 Jun 2018 16:15:07 +0700

From: yunita@skripsian.online

To: naufalhanif@skripsian.online

Subject: Test Anti Spam

Message-ID: <b935648c3ce1e2a691fdd459d3454b58@skripsian.online>

X-Sender: yunita@skripsian.online

User-Agent: Roundcube Webmail/1.2.3

XJS\*C4JDBQADN1.NSBN3\*2IDNEN\*GTUBE-STANDARD-ANTI-UBE-TEST-EMAIL\*C.34X

***Header Email* pada skripsian.online Setelah Penerapan**

Return-Path: <yunita@skripsian.online>

Delivered-To: naufalhanif@skripsian.online

Received: from localhost (unknown [127.0.0.1]) by ns1.skripsian.online (Postfix) with ESMTP id A89C1306969F for <naufalhanif@skripsian.online>; Fri, 29 Jun 2018 02:31:32 +0000 (UTC)

DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=skripsian.online; s=default; t=1530239492; bh=5JtrnNw9RJpLzzBAVfY5dy+L5y+H+RNH4s8DSbVA2Ek=; h=Date:From:To:Subject; b=asreqFSmSoOWmABCcP5PyLZmrQZbv7rg0OuoRQn0/S48QmzuHioDB7iFbWLWHIN2KJPrZDqaCb+GSoyBvoFJyWAUVDtOENA6PlMV+x0+JK10QZeGmuJA/K78twCZQXQFlXU13cb1vw42pR9jCWwigDRuVdKTnoEHkIvWXZEZWpY=

X-Virus-Scanned: amavisd-new at skripsian.online

Authentication-Results: ns1.skripsian.online (amavisd-new); dkim=pass (1024-bit key) header.d=skripsian.online

Received: from ns1.skripsian.online ([127.0.0.1])by localhost (ns1.skripsian.online [127.0.0.1]) (amavisd-new, port 10024) with ESMTP id fAERE5f6jNzQ for <naufalhanif@skripsian.online>; Fri, 29 Jun 2018 09:31:09 +0700 (WIB)

Received: from localhost (localhost [IPv6:::1]) by ns1.skripsian.online (Postfix) with ESMTPA id 22531306969D for <naufalhanif@skripsian.online>; Fri, 29 Jun 2018 09:31:09 +0700 (WIB)

DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/simple; d=skripsian.online; s=default; t=1530239469; bh=5JtrnNw9RJpLzzBAVfY5dy+L5y+H+RNH4s8DSbVA2Ek=; h=Date:From:To:Subject; b=plum4Fjn67DX1TWaQS1fbibLTu6r+aKoAOguEhX/bU3Mf+tnape0FivcCa3reD8mrm8OZEn0Mv2swvV9UiyN6J47gbS5YVXMUd84FMXweF5QNieTwJUjDyScK+b1PIs4929zHUOtyQI4vNdOD8d3yEipK04LcJvZ9YFW+eMUXgQ=

MIME-Version: 1.0

Content-Type: text/plain; charset=US-ASCII; format=flowed

Content-Transfer-Encoding: 7bit

Date: Fri, 29 Jun 2018 09:31:08 +0700

From: yunita@skripsian.online

To: naufalhanif@skripsian.online

Subject: Cek Header Mail

Message-ID: <55a57f675639d913ec310d949a052be7@skripsian.online>

X-Sender: yunita@skripsian.online

User-Agent: Roundcube Webmail/1.2.3

Melakukan pengecekan header email