

**IMPLEMENTASI SENSOR MALTRAIL DAN FAIL2BAN UNTUK
MENDETEKSI DAN MENCEGAH SERANGAN *MALWARE* PADA
JARINGAN SERVER DISKOMINFO SUMEDANG DENGAN PUSH
NOTIFIKASI**

*Implementation of Maltrail Sensor and Fail2Ban For Detection and Prevention Malware
Attack on Server Network at Diskominfo Of Sumedang With Push Notification*

PROPOSAL PROYEK AKHIR

Diajukan sebagai syarat untuk mengambil Mata Kuliah Proyek Akhir

oleh :

RAMA WIJAYA SHIDDIQ

6705184073



**D3 TEKNOLOGI TELEKOMUNIKASI
FAKULTAS ILMU TERAPAN
UNIVERSITAS TELKOM
2021**

LEMBAR PENGESAHAN

Proposal Proyek Akhir dengan judul :

IMPLEMENTASI SENSOR MALTRAIL DAN FAIL2BAN UNTUK MENDETEKSI DAN MENCEGAH SERANGAN *MALWARE* PADA JARINGAN SERVER DISKOMINFO SUMEDANG DENGAN PUSH NOTIFIKASI

*Implementation of Maltrail Sensor and File2Ban For Detection and Prevention Malware Attack
on Server Network at Diskominfo of Sumedang With Push Notification*

oleh :

RAMA WIJAYA SHIDDIQ

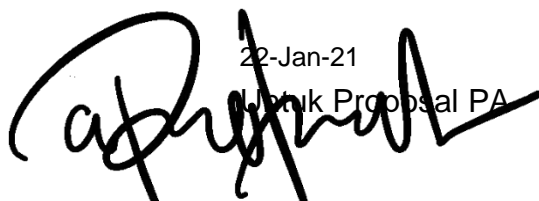
6705184073

Telah diperiksa dan disetujui untuk diajukan sebagai syarat mengambil
Mata Kuliah Proyek Akhir
pada Program Studi D3 Teknologi Telekomunikasi Universitas Telkom


Bandung, 20 Januari 2021

Menyetujui,

Pembimbing I

22-Jan-21
Untuk Proposal PA

ROHMAT TULLOH, S.T., M.T.
NIP. 06830002

Pembimbing II


ASEP MULYANA, S.T., M.T.
NIP. 945700113

ABSTRAK

Diskominfo Sumedang mempunyai tugas menyelenggarakan urusan pemerintahan di bidang Komunikasi, bidang Informatika, bidang Persandian dan bidang Statistik. Diskominfo Sumedang bertugas mengurus dan menjamin keamanan TIK, khususnya di bidang Persandian *cyber security*, seperti *me-monitoring* seluruh keamanan aktivitas jaringan, perlindungan sistem, data, dan peningkatan kualitas keamanan jaringan. Dalam usaha peningkatan kualitas keamanan jaringan, dibutuhkan sebuah sistem yang dapat mendeteksi dan memblokir *malware-malware* yang berusaha masuk ke dalam jaringan pemerintahan di Diskominfo Sumedang.

Pada penelitian ini akan dirancang suatu sistem implementasi sensor Maltrail dan Fail2Ban untuk mendeteksi dan mencegah serangan *malware* pada jaringan *server* Diskominfo Sumedang dengan push notifikasi. *Software* yang digunakan untuk melakukan pendeteksian *malware* yaitu Maltrail (Malware Trail) yang merupakan solusi dari permasalahan tersebut, cara kerja dari *software* ini mirip dengan mekanisme “sensor” yang memindai seluruh aktivitas *traffic* pada jaringan *server*. Kemudian, *software* yang digunakan untuk melakukan *blocking* ‘pencegahan’ dari serangan *malware*, yaitu Fail2Ban. Dan sebagai fitur notifikasi menggunakan via bot telegram dan email. Sistem tersebut dikolaborasikan dan disesuaikan dengan jaringan *server*.

Dengan dibuatnya sistem ini diharapkan dapat membantu mendeteksi dan mencegah serangan *malware* dalam meningkatkan kualitas keamanan jaringan dan mencegah kerugian-kerugian yang disebabkan oleh *malware* khususnya dalam mengakses *browsing* di jaringan internet, karena hampir semua penyebab utama masalah keamanan internet adalah *malware*.

Kata Kunci: Maltrail, Fail2Ban, *malware*, mendeteksi, dan mencegah.

DAFTAR ISI

LEMBAR PENGESAHAN.....	i
ABSTRAK	ii
DAFTAR ISI	iii
DAFTAR GAMBAR	v
DAFTAR TABEL	vi
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Tujuan dan Manfaat	2
1.3 Rumusan Masalah.....	2
1.4 Batasan Masalah	3
1.5 Metodologi	3
BAB II DASAR TEORI.....	6
2.1 <i>Malware</i>	6
2.2 Malware Trail	6
2.3 Cara Kerja	7
2.4 Fitur.....	8
2.5 Fail2Ban	8
2.6 OS Debian Server	9
2.7 Sendemail (SMTP)	9
2.8 Telegram	10
2.9 Crontab.....	10
2.10 Super Simple Auto Refresh.....	11
BAB III MODEL SISTEM	12
3.1 Blok Diagram Sistem.....	12
3.2 Tahapan Perancangan	13
3.3 Perancangan.....	14

BAB IV BENTUK KELUARAN YANG DIHARAPKAN.....	18
4.1 Keluaran yang Diharapkan.....	18
4.2 Parameter Keberhasilan	18
4.3 Jadwal Pelaksanaan.....	20
DAFTAR PUSTAKA	21

DAFTAR GAMBAR

<i>Gambar 1.1 Metode Kerja Penerapan Sistem.....</i>	<i>3</i>
<i>Gambar 2.1 Malware Trail.....</i>	<i>6</i>
<i>Gambar 2.2 Fail2Ban.....</i>	<i>8</i>
<i>Gambar 2.3 OS Debian Server.....</i>	<i>9</i>
<i>Gambar 2.4 Sendemail.....</i>	<i>9</i>
<i>Gambar 2.5 Telegram.....</i>	<i>10</i>
<i>Gambar 2.6 Crontab.....</i>	<i>10</i>
<i>Gambar 2.7 Super Simple Auto Refresh.....</i>	<i>11</i>
<i>Gambar 3.1 Blok Diagram Sistem.....</i>	<i>12</i>
<i>Gambar 3.2 Flowchart.....</i>	<i>13</i>
<i>Gambar 3.3 Konsep Desain Sistem.....</i>	<i>14</i>
<i>Gambar 3.4 Skema Topologi Jaringan Diskominfo Sumedang.....</i>	<i>15</i>
<i>Gambar 4.2 Tampilan Maltrail.....</i>	<i>19</i>

DAFTAR TABEL

<i>Tabel 3.1 Perangkat Keras Yang di Gunakan.....</i>	<i>16</i>
<i>Tabel 3.2 Perangkat Lunak Yang di Gunakan.....</i>	<i>16</i>
<i>Tabel 4.1 Parameter Keberhasilan.....</i>	<i>18</i>
<i>Tabel 4.2 Jadwal Pelaksanaan.....</i>	<i>20</i>

BAB I

PENDAHULUAN

1.1 Latar Belakang

Salah satu ancaman utama di Internet saat ini yaitu *software* berbahaya yang sering disebut sebagai *malware*. Faktanya, sebagian besar masalah keamanan Internet disebabkan oleh *malware*. *Malware* hadir dalam berbagai bentuk dan variasi, seperti *virus*, *worm*, *botnet*, *rootkit*, *trojan horse*, dan program *denial tools* lainnya. Dalam penyebarannya, *malware* mengeksploitasi kerentanan *software* di *browser* dan sistem operasi, atau menggunakan teknik *social engineering* untuk mengelabui pengguna agar dapat menjalankan program-program berbahaya. (Bayer et al. 2009).

Perusahaan anti-virus Malwarebytes (2019) merilis laporan tahunan tentang kondisi *malware* di seluruh dunia dalam jurnal “2019 States of Malware”. Laporan tersebut menyatakan bahwa terdapat kurang lebih 750 juta serangan *malware* yang terdeteksi menyerang komputer end-user (personal) sepanjang tahun 2017–2018 di seluruh dunia. Kemudian, terdapat kurang lebih 71 juta *malware* yang terdeteksi menyerang pengguna *business-user* (perusahaan/industri/lembaga) sepanjang tahun 2017–2018. Sayangnya, jumlah yang meningkat dan keragaman *malware* membuat teknik keamanan klasik, seperti pemindai *anti-virus*, tidak efektif dan, sebagai konsekuensinya, jutaan host di Internet saat ini terinfeksi dengan perangkat lunak berbahaya (Microsoft, 2009; Symantec, 2009).

Berdasarkan penelitian tersebut, diperlukan adanya sistem keamanan jaringan untuk *monitoring* dan pencegahan dari serangan *malware* yang melintasi jaringan *server* pada sektor perangkat jaringan di Diskominfo Sumedang. Beberapa contohnya adalah seperti mengakses informasi sensitif, mengakses situs-situs berbahaya yang mengandung *malware* atau bahkan mengubah data penting dan membanjiri jaringan server dengan merugikan banyak pihak oleh oknum yang tidak bertanggung jawab. Pada struktur yang akan diterapkan, *software* Maltrail dan Fail2Ban yang telah ada, dikonfigurasi sedemikian rupa agar dapat saling berkolaborasi pada sistem keamanan jaringan. Dari beberapa penelitian lanjutan yang telah dilakukan, sistem pendeteksi serangan *malware* ini dapat dikembangkan lebih jauh pada sisi otomatisasi dalam monitoring dan pencegahan *malware* traffic yang

ada secara real-time beserta fitur notifikasi melalui aplikasi Telegram dan pelaporan berkas melalui e-mail secara berkala.

Beberapa penelitian yang telah dilakukan berkaitan dengan sistem Maltrail sebagai basis dari sistem malware monitoring, yaitu (Hudzaifah et al. 2019; Suci et al. 2019; Bayer et al. 2009; McGraw dan Morrisett 2000; Idika dan Mathur 2007). Pada penelitian tersebut, sistem pada sensor Maltrail dengan fungsionalitas berbasis *malware monitoring* belum tersedia fitur pencegahan dan fitur notifikasi untuk memblokir *malware-malware* dan melaporkan berkas secara real-time melalui media sosial seperti aplikasi telegram dan *e-mail*.

1.2 Tujuan dan Manfaat

Adapun tujuan dan manfaat dari Proyek Akhir ini, sebagai berikut:

1. Dapat melakukan monitoring serta mendeteksi paket-paket yang masuk melalui jalur *network server* yang terindikasi dan terdeteksi sebagai *malware* secara otomatis
2. Dapat melaporkan IP yang diblokir oleh sistem melalui notifikasi Telegram, sehingga administrator dapat *me-monitoring* sistem secara *real-time*
3. Dapat membantu administrator dalam merekapitulasi hasil pemindaian *data log traffic* malware melalui e-mail secara otomatis berdasarkan periodik yang ditetapkan
4. Dapat mengintegrasikan seluruh komponen *software* dengan baik untuk tingkat keberhasilan dalam mendeteksi dan mencegah serangan malware

1.3 Rumusan Masalah

Adapun rumusan masalah dari Proyek Akhir ini, sebagai berikut:

1. Apa saja fungsi atau fitur yang akan diterapkan pada sistem sensor Maltrail dan Fail2Ban?
2. Bagaimana cara melakukan konfigurasi pada sistem sensor Maltrail dan Fail2Ban?
3. Bagaimana implementasi sistem penggunaan sensor Maltrail dan Fail2Ban dalam mendeteksi dan mencegah serangan *malware* dengan push notifikasi?
4. Bagaimana hasil rekapitulasi laporan data *log traffic malware* pada sensor

Maltrail berdasarkan parameter Fail2Ban yang dikirimkan ke email?

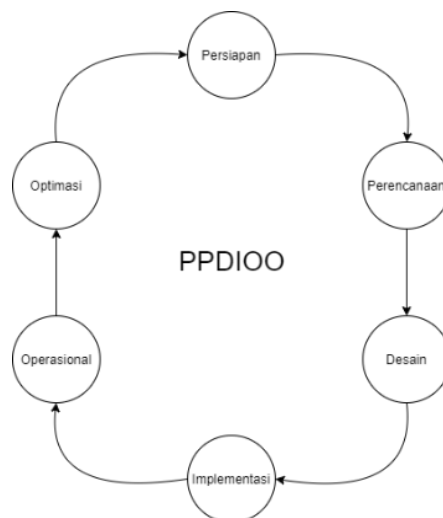
1.4 Batasan Masalah

Dalam Proyek Akhir ini, dilakukan pembatasan masalah sebagai berikut:

1. *Server* harus selalu dalam keadaan menyala dan terhubung dengan internet
2. Sistem maltrail hanya bisa mendeteksi *malware*
3. Sumber *malware blacklists* yang digunakan hanya bersumber dari situs Github resmi Developer *software* Maltrail
4. Pengujian hanya dilakukan menggunakan metode PING terhadap *malware*

1.5 Metodologi

Metodologi yang digunakan adalah PPDIOO (Prepare, Plan, Design, Implement, Operate, dan Optimize). PPDIOO adalah metodologi Cisco yang mendefinisikan siklus hidup berkelanjutan dari layanan yang dibutuhkan terhadap suatu jaringan. (Sivasubramanian *et al*, 2010). Metode ini cocok untuk mendesain pengembangan keamanan jaringan yang pendekatannya terpusat pada administrator. Pada metode ini terdapat 6 tahapan, yaitu persiapan, perencanaan, desain, implementasi, operasional, dan optimasi. Gambar 1 menunjukkan alur dari metode kerja yang digunakan.



Gambar 1.1 Metode Kerja Penerapan Sistem

1. Persiapan

Tahap persiapan merupakan tahap menentukan kebutuhan sistem keamanan di dalam sebuah jaringan *server* dengan mengusulkan suatu konsep arsitektur jaringan, serta bertujuan untuk strategi pengembangan sistem keamanan jaringan. Pada tahap ini, dilakukan penentuan apa saja yang menjadi kebutuhan untuk pengembangan sistem keamanan jaringan.

2. Perencanaan

Tahap perencanaan merupakan tahap yang dilakukan untuk mengidentifikasi hal-hal yang harus dipenuhi terhadap sistem keamanan jaringan berdasarkan tujuan, fasilitas/manfaat, kebutuhan administrator, serta analisis Gap. Analisis Gap merupakan sebuah metode pengukuran untuk mengetahui kesenjangan (*gap*) antara kinerja suatu variabel dengan tujuan yang diharapkan *user* terhadap variabel tersebut. Tahap ini harus sejalan dengan ruang lingkup (batasan), dan parameter sumber daya yang disesuaikan dengan kebutuhan sistem keamanan jaringan. Rencana proyek ini dapat diikuti dan diperbarui selama tahap-tahap dalam siklus berlangsung.

3. Desain

Tahap desain jaringan dikembangkan berdasarkan persyaratan teknis fungsionalitas yang diperoleh dari kondisi sebelumnya. Spesifikasi desain jaringan adalah desain yang bersifat komprehensif dan terperinci, sehingga dapat memenuhi persyaratan teknis dan fungsionalitas yang dibutuhkan pada sistem keamanan jaringan.

4. Implementasi

Tahapan implementasi merupakan tahap instalasi dan konfigurasi terhadap peralatan-peralatan keamanan jaringan yang sesuai dengan spesifikasi desain. Pada tahap ini akan melakukan pengujian desain, konfigurasi dan topologi arsitektur jaringan yang telah gambarkan dan setelah selesai diimplementasikan dilakukan pengujian terhadap jaringan untuk memastikan bahwa pengoperasian jaringan sesuai dengan yang diharapkan.

5. Operasional

Tahapan operasional merupakan tahap terpanjang yang dibutuhkan pada metode PPDIOO, karena diperlukan pemantauan terhadap jalannya alur data dan konfigurasi keamanan jaringan. Operasional pada tahap ini meliputi pengelolaan dan memonitor komponen-komponan keamanan jaringan, mengelola kegiatan *upgrade*, mengelola kinerja, serta mengidentifikasi dan mengoreksi kesalahan jaringan. Pada tahap ini, penyedia akan melakukan pemantuan proaktif dan reaktif, serta manajemen kerja keamanan jaringan. Pada tahap ini kemungkinan akan adanya perubahan, penambahan dan perubahan menyesuaikan pada kondisi.

6. Optimasi

Tahapan optimasi dapat terjadi kapan saja setelah jaringan beroperasi. Tahap ini memungkinkan untuk memodifikasi desain jaringan, memperbaiki masalah kinerja, atau menyelesaikan masalah-masalah pada aplikasi (*software*). Tahap ini terjadi biasanya karena adanya perubahan teknis atau persyaratan teknis dan perawatan sistem keamanan jaringan. Tahapan ini akan dilakukan uji perbandingan pada tahap sebelumnya. Jika terdapat perubahan, tahap ini akan diperbarui untuk memastikan sistem keamanan jaringan dapat berjalan dengan konsisten serta sesuai dengan desain dan perencanaan.

BAB II

DASAR TEORI

2.1 *Malware*

Malware (malicious software) adalah sebuah program yang berisi kode-kode yang ditambahkan, diubah, atau dihapus dari suatu sistem *software* untuk secara sengaja menyebabkan kerusakan atau menumbangkan fungsi sistem tersebut. Meskipun masalah *malware* memiliki sejarah yang panjang, sejumlah serangan *malware* baru-baru ini yang dipublikasikan secara luas dan tren ekonomi tertentu menunjukkan bahwa *malware* cepat menjadi masalah penting bagi industri, pemerintah, dan individu. (McGraw dan Morrisett 2000).

Malware menjadi istilah umum yang mencakup *virus*, *trojan*, *spywares*, dan kode intrusif lainnya yang tersebar luas saat ini. Analisis *malware* adalah proses multi-langkah yang memberikan wawasan tentang struktur dan fungsi *malware*, memfasilitasi pengembangan penangkal racun. (Vasudevan dan Yerraballi 2006). Secara teknis, *malicious traffic* ialah suatu kejadian abnormal pada lalu lintas jaringan dan merupakan perbuatan user yang tidak bertanggung jawab tanpa sepengetahuan pengguna komputer yang sah. (Hudzaifah *et al.* 2018).

2.2 *Malware Trail*



Gambar 2.1 *Malware Trail*

Malware Trail (Maltrail) adalah sistem pendeteksi lalu lintas *malware* berbahaya yang menggunakan daftar hitam (*blacklists*) yang *repository*-nya disediakan oleh pihak ketiga yang berisi daftar *malware* berbahaya dan mencurigakan, serta jalur statis atau lokal yang disusun dari berbagai laporan anti-virus dan daftar yang ditentukan pengguna khusus. Maltrail didasarkan pada *traffic sensor-sensor-server-client architecture*. Sensor adalah komponen mandiri yang berjalan pada *monitoring mode* atau di mesin mandiri. Ia memantau paket-paket terdaftar sebagai *blacklist packet* (berupa nama domain, URL, atau alamat IP) yang melewati jaringan tersebut. (Stampar 2014)

2.3 Cara Kerja

Arsitektur maltrail didasarkan pada Sensor > Server > Klien.

1. Sensor

Sensor adalah komponen mandiri yang berjalan pada node pemantau yang bertugas memantau traffic yang lewat untuk jalur yang masuk daftar hitam (URL atau IP) pada jaringan. Sensor akan mengirimkan detail acara ke Server.

2. Server

Server merupakan komponen yang menyimpan semua peristiwa yang terjadi dalam periode (24h) dan memberikan data ke Klien untuk aplikasi web pelaporan. Data dikirim ke klien dalam potongan terkompresi, dan diproses secara berurutan.

3. Client

Client berupa web browser (IE, Chrome, Firefox, dll.). Semua peristiwa (yaitu entri log) dalam periode (24h) akan ditransfer ke client, dan aplikasi web pelaporan yang bertanggung jawab penuh atas bagian presentasi seperti ancaman, kejadian, sumber, dan jejak.

2.4 Fitur

Maltrail mempunyai fitur antara lain :

1. Menggunakan banyak daftar hitam publik yang berisi jalur mencurigakan (alientvault, autoshun, badips, sblam, dll) dan jejak statis yang dikumpulkan dari berbagai laporan Anti Virus dan daftar yang ditetapkan pengguna khusus.
2. Memiliki jalur statis yang luas untuk identifikasi (nama domain, URL, atau alamat IP). Sistem maltrail dapat menampilkan informasi berupa DNS dan WHOIS dari RIPE sebagai penyedia informasi.
3. Interface presentasi elaporan memakai aplikasi web browser. Ketika data semua detail peristiwa(event) diterima oleh klien, maka klien akan mempresentasikan laporan data tersebut dengan memakai web browser. Mulai dari waktu pertama kejadian, waktu kejadian terakhir, protokol yang dipakai, sumber alamat IP, dan alamat IP tujuan.

2.5 Fail2Ban



Gambar 2. 1 Fail2Ban

Fail2Ban merupakan *software* yang dapat memindai *file log* dan melarang alamat IP yang menunjukkan tanda-tanda paket berbahaya, banyaknya kegagalan kata sandi, dan lain-lain. Umumnya, Fail2Ban digunakan untuk menambahkan *firewall rules* untuk menolak alamat IP untuk jumlah waktu tertentu, meskipun tindakan pencegahan lainnya juga dapat dikonfigurasi lebih lanjut. (Fail2ban.org 2020).

Fail2Ban berfungsi sebagai monitor jumlah kegagalan *login* SSH di *server*, yang selanjutnya alamat IP akan diblokir. Dalam kasus ini Fail2Ban menangani serangan *brute-force*, untuk itu diperlukan beberapa tahap dalam konfigurasi Fail2Ban. (Kurniawan *et al.* 2016).

2.6 OS Debian server



Gambar 2.3 OS Debian Server

OS Debian server digunakan untuk menjalankan segala macam aktivitas layanan *server* pada jaringan.

2.7 Sendemail (SMTP)



Gambar 2.4 Sendemail

SendEmail adalah barisan perintah SMTP *e-mail client* yang ringan. Program ini dirancang untuk digunakan di dalam *bash scripts*, *batch files*, program Perl dan situs *web* yang cukup mudah diterapkan untuk memenuhi kebutuhan penggunaanya. SendEmail ditulis dengan Bahasa Perl yang unik, karena program ini tidak

memerlukan modul. Program ini memiliki seperangkat opsi baris perintah yang intuitif dan fleksibel yang membuatnya sangat mudah dipelajari dan digunakan. (Caspian 2009).

2.8 Telegram



Gambar 2.5 Telegram

Telegram adalah platform IM (*instant messaging*) yang memungkinkan penggunaannya untuk bertukar pesan, menggunakan berbagai skema komunikasi (yaitu *one-to-one*, *one-to-many*, dan *many-to-many*), serta untuk melakukan panggilan suara, menggunakan berbagai teknik menjaga keamanan/privasi. Telegram mendukung pertukaran data berupa pesan teks (yang isinya teks biasa) dan pesan non-teks (dari jenis apa pun, termasuk informasi kontak, koordinat geografis, dan fail jenis apa pun). (Anglano *et al.* 2017).

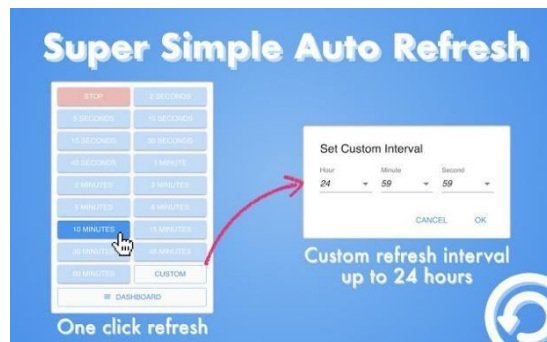
2.9 Crontab



Gambar 2.6 Crontab

Cron merupakan program yang dapat menjalankan instruksi secara otomatis di latar belakang secara berkala. Cron juga dapat digunakan untuk membuat *backup* data, sinkronisasi fail, menjadwalkan pembaruan sistem, dan lain-lain. Crontab (*cron* berasal dari kata *chronos* (Bahasa Yunani) yang artinya waktu, sedangkan *tab* berasal dari kata *table* (Bahasa Inggris) yang artinya tabel. Aplikasi Crontab tersedia pada sistem operasi UNIX/LINUX, yang digunakan untuk menjadwalkan perintah yang akan dieksekusi secara berkala. (Kishore dan Sachin, tanpa tahun).

2.10 Super Simple Auto Refresh



Gambar 2.7 Super Simple Auto Refresh

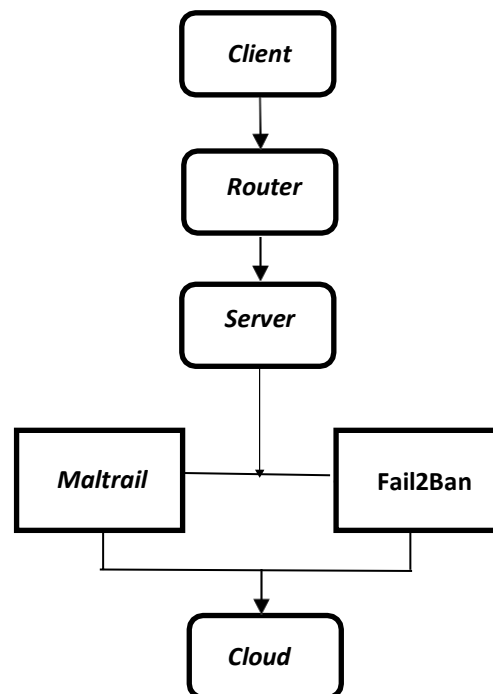
Super Simple Auto Refresh merupakan program ekstensi yang tersedia untuk *browser* Google Chrome yang dapat memperbarui halaman web secara otomatis berdasarkan interval waktu yang dapat diatur sesuai dengan kebutuhan pengguna hanya dengan menggunakan satu klik. (Super Simple Auto Refresh, 2020).

BAB III

MODEL SISTEM

3.1 Blok Diagram Sistem

Pada bab ini akan dijelaskan mengenai blok diagram sensor Maltrail dan Fail2Ban untuk mendeteksi dan mencegah serangan *malware*. Sistem Maltrail dan Fail2Ban ini diletakkan sebagai sistem pendeteksi paket- paket data yang melewati server. Seluruh aktivitas akses internet pada jaringan server diskominfo sumedang, akan melewati tahap pemindaian oleh sistem Maltrail ini karena selayak dengan sensor yang mengecek apakah terdapat paket malware yang melewati traffic pada jaringan tersebut atau tidak. Sedangkan sistem Fail2Ban diletakkan berdampingan dengan Maltrail, karena Fail2Ban berperan sebagai program yang mengeksekusi, mencegah dan melarang malware yang lewat berdasarkan dari daftar log yang telah dicatat dan dideteksi oleh Maltrail dengan parameter intensitas penyerangan *malware* yang dilakukan.



Gambar 3. 1 Blok Diagram Sistem

3.2 Tahapan Perancangan

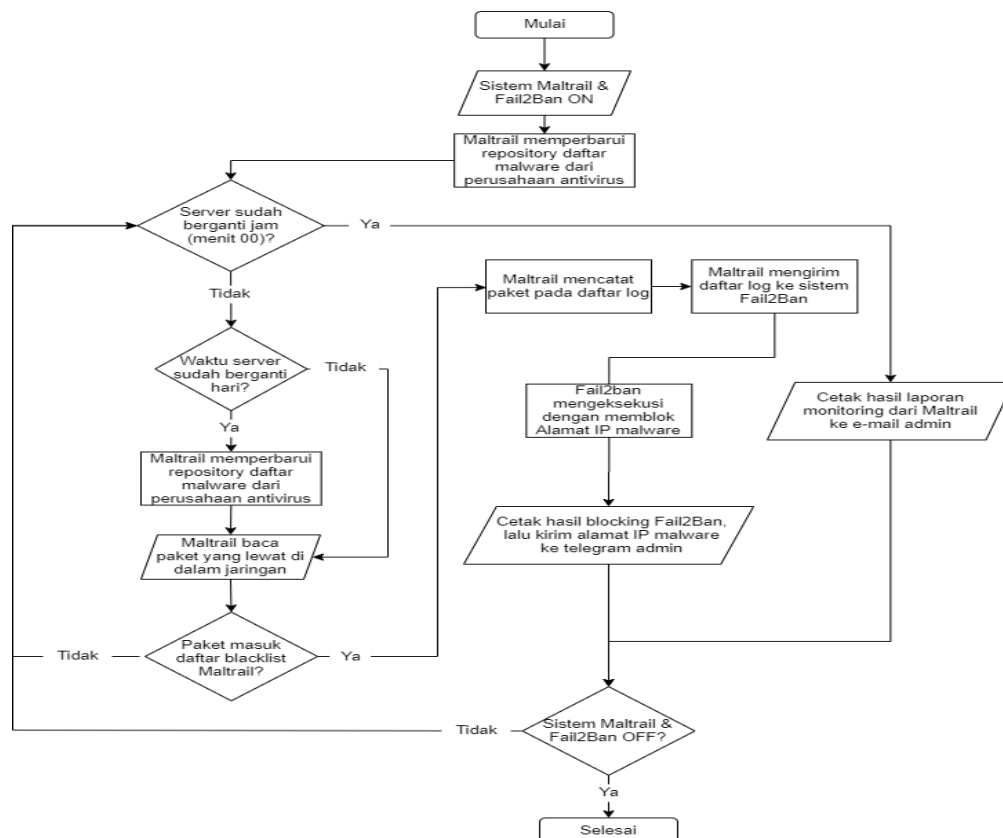
Proses perancangan perangkat ini dilakukan dengan metode eksperimental, tahapan pembuatannya adalah sebagai berikut:

1. Penentuan spesifikasi

Langkah awal dalam pembuatan sistem ini adalah dengan menentukan rancangan untuk mengintegrasikan semua software agar dapat terintegrasi dengan di atur oleh Maltrail dan Fail2Ban, sistem tersebut dapat menampilkan data pada GUI atau diagram di Maltrail dan dapat mengirimkan laporan log *malware* ke Telegram dan *e-mail*

2. Penyusunan Komponen

Semua komponen *software* akan diintegrasikan dengan dengan cara membuat mengintegrasikan beberapa script codingan. untuk tahapan penyusunan komponennya dapat dibuat *flowchart* sebagai berikut



Gambar 3. 2 Flowchart

3.3 Perancangan

Pada proyek akhir ini akan menggabungkan beberapa komponen software sehingga akan menjadi suatu sistem yang diharapkan, Sistem yang dimaksud adalah sebagai berikut :

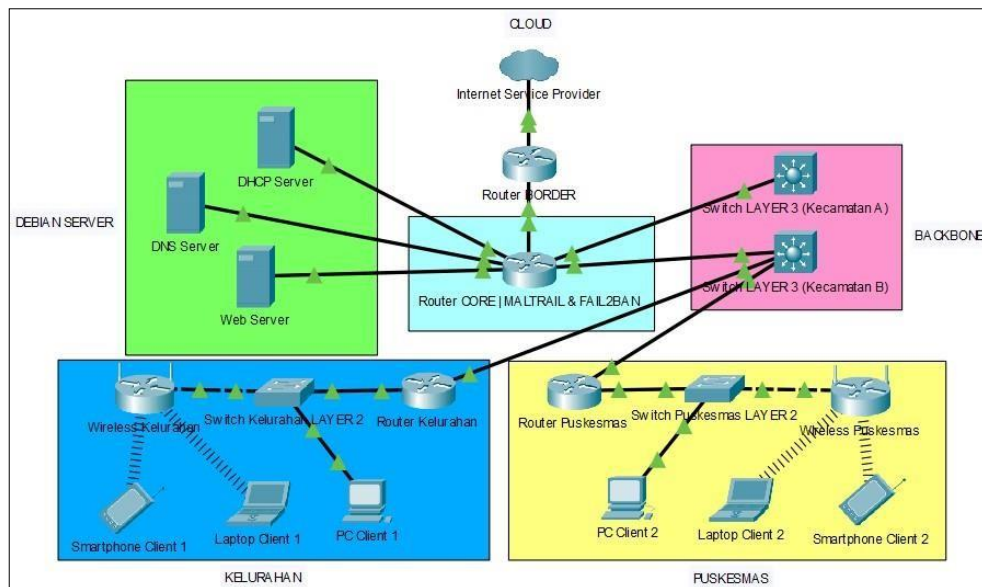
3.3.1 Gambaran Sistem Usulan



Gambar 3.3 Konsep Desain Sistem

Pada Gambar 3.3 dapat dijelaskan sistem secara garis besar ketika cloud menerima packet request lalu mengirimnya ke client, paket tersebut akan melewati serangkaian tahap pemindaian paket oleh sensor Maltrail berdasarkan database yang tersedia pada datanya. Jika paket tersebut terindikasi terdapat *malware*, sistem Maltrail akan melakukan pencatatan berupa malware log. Selanjutnya, log tersebut akan dieksekusi oleh sistem Fail2Ban dalam bentuk alamat IP blocking atau pembatasan akses paket oleh sebuah alamat IP di dalam jaringan tersebut. Setelah alamat IP tersebut diblokir, Fail2Ban akan mengirimkan informasi pemblokirannya ke telegram dan akan muncul notifikasi bahwa IP dalam jaringan tersebut terindikasi *malware*. Untuk notifikasinya dikirimkan secara realtime. Server akan menampilkan data dari log Fail2Ban tersebut setelah diblokir. IP yang telah diblokir bisa di unbanned dalam interval waktu tertentu. Kemudian hasil rekapitulasi data keseluruhan dari *malware* tersebut akan ditampilkan di server Maltrail dalam bentuk GUI atau diagram untuk mengidentifikasi jenis jenis malware yang masuk. Dan sistem akan mengirimkan ke *e-mail* untuk melihat laporan rekapitulasi dari sensor Maltrail.

3.3.2 Topologi Sistem



Gambar 3.4 Skema Topologi Jaringan Diskominfo Sumedang

Perencanaan topologi jaringan yang akan diterapkan dapat dilihat pada Gambar 3.4 Topologi tersebut dibuat berdasarkan topologi yang telah diterapkan di Diskominfo Sumedang. Pada Gambar 3.4, terdapat tiga komponen utama untuk pembuatan skema topologi jaringan *malware blocking* menggunakan Maltrail dan Fail2Ban, yakni sebuah *router-server*, program Maltrail, dan program Fail2Ban. Dalam penerapannya, *server* yang digunakan, yakni Debian *Server* seri 9. Untuk sistem Maltrail dan Fail2Ban diinstal secara bersamaan, lalu diintegrasikan agar dapat saling berkolaborasi. Kedua sistem tersebut diterapkan pada Router CORE (*router-server*).

Pada Gambar 3.4, sistem Maltrail berperan seperti sensor yang mengecek apakah terdapat malware yang melewati traffic pada jaringan tersebut atau tidak, sedangkan sistem Fail2Ban diletakkan berdampingan dengan Maltrail sebagai program yang mengeksekusi, mencegah, dan melarang malware yang lewat berdasarkan dari daftar malware log yang telah dideteksi dicatat dan oleh Maltrail dan Fail2Ban ini memiliki kemampuan memblokir paket-paket ilegal sekaligus menyediakan fitur report forwarding.

3.3.2 Spesifikasi Sistem

Berikut ini adalah kebutuhan perangkat keras dan perangkat lunak yang dibutuhkan dalam proyek akhir ini.

3.2.2.1 Perangkat Keras

Adapun beberapa perangkat keras yang di gunakan di sistem ini, yaitu :

Tabel 3.1 Perangkat Keras yang di Gunakan

No.	Hardware	Unit	Keterangan
1.	Laptop	1	Penempatan Sensor, server, dan client
2.	Modem	1	Untuk koneksi internet

3.2.2.1 Perangkat Lunak

Adapun beberapa perangkat lunak yang di gunakan di sistem ini, yaitu :

Tabel 3.2 Perangkat Lunak Yang di Gunaka

No.	Software	Spesifikasi	Keterangan
1.	OS Debian Server	Distro Debian Linux versi 9.12	Sensor dan Server
2.	Windows	Windows 10	Client
3.	Python	Python 2.6 atau 2.7	Bahasa pemrograman yang dipakai
4.	Maltrail	Versi 0.17.5	<i>Software</i> sistem pendeteksi <i>malware traffic</i> pada jaringan <i>server</i>
5.	Fail2ban	Versi 0.10.4	<i>Software</i> sistem pencegah dan pelindung server komputer dari serangan <i>brute-force malware</i> berdasarkan intensitas serangan <i>malware</i>
6	Cisco Packet Tracer	Versi 7.3.0	<i>Software</i> untuk desain topologi jaringan
7	Telegram	Versi 5.15.0	Aplikasi media sosial untuk pelaporan status keadaan sistem beserta IP yang diblokir
8	Super Simple Auto Refresh	Versi 7.3.0	<i>Software auto-refresh</i> situs web Google Chrome berdasarkan periodik yang diterapkan pengguna
9	Crontab	Versi 0.22.6	Aplikasi untuk penjadwalan perintah yang akan dieksekusi secara berkala

10	Sendemail	Versi 1.56	Aplikasi untuk pengiriman <i>e- mail</i> dari sistem kepada administrator
----	-----------	------------	---

BAB IV

BENTUK KELUARAN YANG DIHARAPKAN

4.1 Keluaran yang Diharapkan

Perancangan dan realisasi pada Proyek Akhir akan dibuat sistem software dengan spesifikasi sebagai berikut :

- a) Dapat mendeteksi paket-paket yang terindikasi *malware*
- b) Dapat mencegah *malware log* dengan alamat IP *blocking*
- c) Dapat menampilkan *malware log* pada browser dan e-mail
- d) Dapat menampilkan GUI untuk keluaran data yang telah di proses pada sensor maltrail.
- e) Dapat melaporkan status sistem dan alamat IP *malware* ke aplikasi telegram
- f) Dapat berjalan dengan baik keintegrasian seluruh komponen software yang digunakan untuk tingkat keberhasilan dalam mendeteksi dan mencegah serangan malware

4.2 Parameter Keberhasilan

Adapun parameter keberhasilan dapat dilihat pada table berikut :

Tabel 4.1 Parameter Keberhasilan

No	Domain	Alamat IP	Alamat DNS	Sampel Domain
1	x	x	x	x
2	x	X	x	x
3	x	X	x	x
4	x	x	x	x

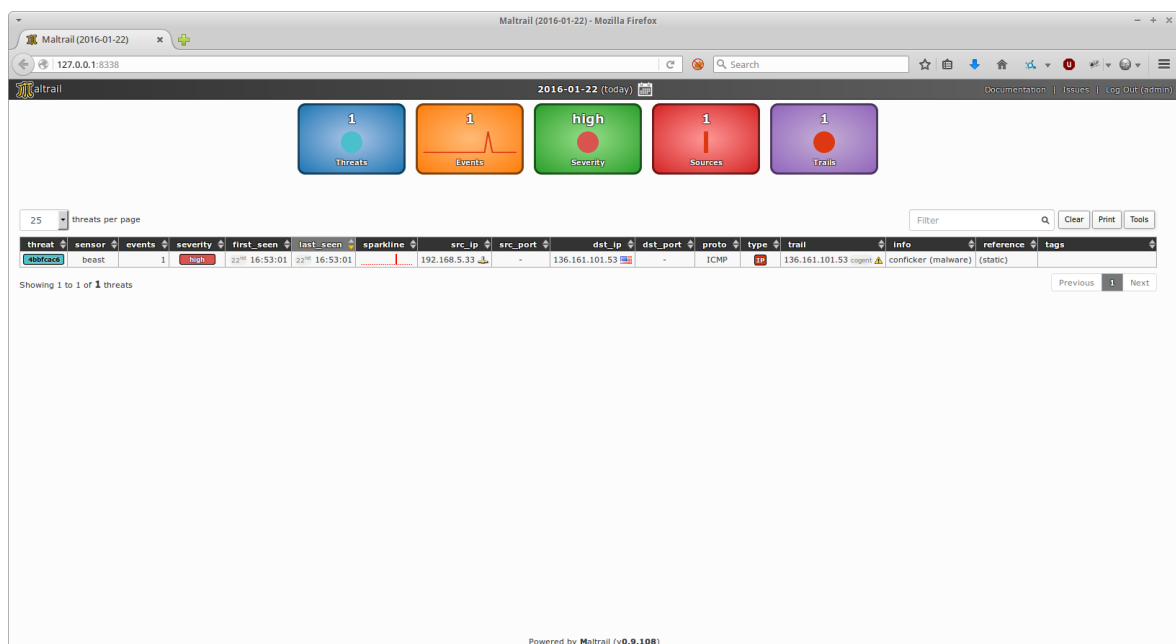
Parameter pengujian berdasarkan rules yang terdapat pada Fail2ban

1. Currently Banned
2. Total Banned
3. Banned IP List

4. Maxretry

5. Bantime

Pengujian *malware* dilakukan dengan melakukan akses browsing terhadap domain-domain atau website yang terindikasi sebagai *malware* dan tidak terindikasi. Hasilnya, domain yang meminta access request secara terus menerus itu akan terdeteksi dan terblokir oleh Maltrail dan Fail2Ban berdasarkan parameter yang digunakan pada Fail2Ban yakni *maxretry* dan *bantime*. *Maxretry* parameter yang digunakan untuk memlimit jumlah percobaan login dari sebuah host, jika melewati batas, maka ip akan dibanned. Sedangkan *bantime* merupakan parameter yang digunakan untuk menset durasi dalam satuan detik untuk IP yang sedang di blokir.



Gambar 4.2 Tampilan maltrail

Tampilan dashboard Maltrail pada saat sebelum pengujian yang akan menampilkan GUI atau diagram yang berisi threats (tingkat ancaman), events (jumlah kejadian), severity (tingkat kesulitan), sources (sumber malware), dan trails (jejak malware melewati alamat IP). Selain itu juga terdapat fitur pencarian daftar *malware* dan fitur mencetak tabel data *malware*, dan hasil rekapitulasi *malware* secara lengkap yang dikirimkan ke *e-mail* dari hasil bloking berdasarkan rules dari Fail2ban yaitu *maxretry* dan *bantime* menggunakan notifikasi telegram dan dikirimkan ke sensor Maltrail.

4.3 Jadwal Pelaksanaan

Adapun jadwal pengerjaan Proyek Akhir bisa dilihat pada tabel berikut:

Tabel 4.2 Jadwal Pelaksanaan

Judul Kegiatan	Waktu							
	Des	Jan	Feb	Mar	Apr	Mei	Jun	Jul
Studi Literatur								
Perancangan dan Pembuatan Alat								
Pengujian								
Analisa								
Pembuatan Laporan								

DAFTAR PUSTAKA

- [1] Stampar M. 2016. Malicious Traffic Detection System. Github. [diunduh 2020 Mar 16]. Tersedia pada: <https://github.com/stamparm/maltrail>.
- [2] Anglano C, Massimo C, Marco G. 2017. Forensic Analysis of Telegram Messenger on Android Smartphones”. Alessandria (IT): DiSIT–Computer Science Institute, Università del Piemonte Orientale. Vol 23: 31—49. <https://doi.org/10.1016/j.diin.2017.09.002>.
- [3] Caspian. 2009. An Email Program for Sending SMTP Mail from a Command Line. [diunduh 2020 Mar 16]. Tersedia pada: <http://http://caspian.dotconf.net/menu/Software/SendEmail>.
- [4] Hudzaifah, Anang S, Devie RS. 2018. Membangun Sistem Monitoring Malicious Traffic di Jaringan dengan Maltrail. Bandung (ID): Telkom University. Vol 4 No.3: 2013—2018.
- [5] Jaquier C. 2004. Fail2Ban Intrusion and Prevention Software. GitHub. [diunduh 2020 Mar 16].
- [6] Super Simple Auto Refresh. 2020. Overview: Super Simple Auto Refresh. Chrome Web Store. [diunduh 2020 Mar 16]. Tersediapada: <https://chrome.google.com/webstore/detail/super-simple-auto-refreshgljclgacfa lmnebgmhknodlplmngmfpi>.
- [7] Kurniawan I, Ferry Mulyanto, Fuad Nandiasa. 2016. Sistem Pencegah Serangan Bruteforce pada Ubuntu *Server* dengan menggunakan Fail2Ban. Bandung (ID)
- [8] Sivasubramanian B, Erum F, Richard Froom. 2010. Analyzing the Cisco Enterprise Campus Architecture. Cisco Press. [diunduh 2020 Mar 16]. Tersediapada: <https://www.ciscopress.com/articles/article.asp?p=1608131>.
- [9] Bayer U, Paolo MC, Clemens H, Christoper K, Engin K. 2006. Scalable, Behavior-Based Malware Clustering. Vienna (AZ): Technical University Vienna. 1—18.
- [10] Kujawa A, Wendy Z, Jovi U, Jerome S, William T, Pieter A, Chris B. 2019. *2019 State of Malware*. California (US): Malwarebytes Corporation. 6—7.
- [11] Kishore A, Sachin. Tanpa tahun. Crontab: Everything You Want To Know. Oracle E-Business Suite 11/Certified Professional. 1.



UNIVERSITAS TELKOM
FAKULTAS ILMU TERAPAN
KARTU KONSULTASI
SEMINAR PROPOSAL PROYEK AKHIR

NAMA / PRODI : RAMA WIJAYA SHIDDIQ/D3 TEKNOLOGI
TELEKOMUNIKASI

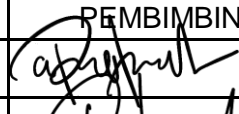
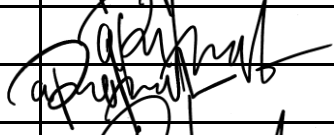
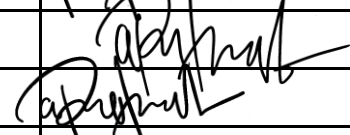
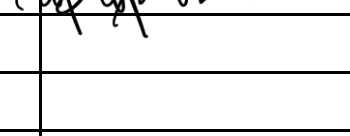


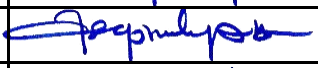
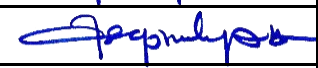

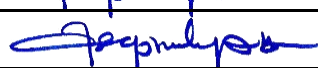
NIM : 6705184073

JUDUL PROYEK AKHIR :

IMPLEMENTASI SENSOR MALTRAIL DAN FAIL2BAN UNTUK MENDETEKSI DAN MENCEGAH SERANGAN MALWARE PADA JARINGAN SERVER DISKOMINFO SUMEDANG DENGAN PUSH NOTIFIKASI

CALON PEMBIMBING : I. Rohmat Tulloh, S.T., M.T.

II. Asep Mulyana, S.T., M.T.

NO	TANGGAL	CATATAN HASIL KONSULTASI	TANDA TANGAN CALON PEMBIMBING I
1	22 Januari 2021	BAB 1 (SELESAI)	
2	22 Januari 2021	BAB 2 (SELESAI)	
3	22 Januari 2021	BAB 3 (SELESAI)	
4	22 Januari 2021	BAB 4 (SELESAI)	
5	22 Januari 2021	FINALISASI PROPOSAL	
6			
7			
8			
9			
10			
NO	TANGGAL	CATATAN HASIL KONSULTASI	TANDA TANGAN CALON PEMBIMBING II
1	20 Januari 2021	BAB 1 (SELESAI)	
2	20 Januari 2021	BAB 2 (SELESAI)	
3	20 Januari 2021	BAB 3 (SELESAI)	
4	20 Januari 2021	BAB 4 (SELESAI)	
5	20 Januari 2021	FINALISASI PROPOSAL	
6			
7			
8			
9			
10			



PEMERINTAH DAERAH KABUPATEN SUMEDANG
**DINAS KOMUNIKASI, INFORMATIKA,
PERSANDIAN DAN STATISTIK**

Alamat : Jl. Angkrek No.103 Sumedang, No.Tlp: (0261) 201255,
Website : sumedangkab.go.id E-mail : diskominfosanditik@sumedangkab.go.id , 45323

Sumedang, 11 Januari 2021

Nomor : 800/ 17 /Umpeg
Sifat : Biasa
Lampiran : -
Hal : Rekomendasi Penelitian

Kepada
Yth. Rama Wijaya Shiddiq
di

Tempat

Dipermaklumkan dengan hormat, menindaklanjuti Surat Permohonan dari Sdr.
Rama Wijaya Shiddiq tanggal 10 Januari 2021 perihal Permohonan Ijin Penelitian.

Berdasarkan hal tersebut diatas, pada dasarnya kami tidak berkeberatan dan
memberikan rekomendasi/ijin kepada :

Nama : Rama Wijaya Shiddiq

NIP 6705184073

Program Studi : D3 Teknologi Telekomunikasi

untuk melakukan penelitian dengan judul *“Implementasi Sensor Maltrail dan
Fail2ban untuk Mendeteksi dan Mencegah Serangan Malware Pada Jaringan Server
Diskominfo Sumedang Dengan Push Notifikasi.”* pada Kantor Dinas Komunikasi dan
Informatika, Persandian dan Statistik Kabupaten Sumedang.

Demikian Surat Rekomendasi kami buat, untuk dapat dipergunakan
sebagaimana mestinya.



Ditandatangani Secara Elektronik Oleh:

Dr. IWA KUSWAERI

NIP. 196203031988031012

Kepala Dinas Komunikasi,
Informatika, Persandian dan Statistik

Catatan :

- ✓ UU ITE No 11 Tahun 2008 Pasal 5 ayat 1
“Informasi Elektronik dan/atau Dokumen Elektronik dan/atau hasil cetaknya merupakan alat bukti hukum yang sah.”
- ✓ Dokumen ini telah ditandatangani secara elektronik menggunakan **sertifikat elektronik** yang di terbitkan **BSrE**.
- ✓ Surat ini dapat dibuktikan keasliannya dengan terdaftar di <http://e-office.sumedangkab.go.id>, kode: MTVINTFM