Kode/Rumpun Ilmu: 458/Teknik Informatika
Bidang : TIK

LAPORAN PENELITIAN INTERNAL
PENELITIAN PEMULA

MONITORING INTERFACES FASTETHERNET PADA CISCO CATALYST 3750 UNTUK MENJAMIN KEAMANAN PENGGUNAAN JARINGAN KOMPUTER DI LABORATORIUM KOMPUTASI STTA

Disusun oleh:

Sudaryanto, S.T., M.Eng. **NIDN. 0511097901**
Dwi Nurhayati **NIM. 15030016**

Dibiayai oleh
Pusat Penelitian dan Pengabdian Pada Masyarakat
Sekolah Tinggi Teknologi Adisutjipto Yogyakarta
Sesuai Dengan Kontrak Penelitian Tahun Anggaran 2018/2019

PROGRAM STUDI TEKNIK INFORMATIKA
DEPARTEMEN INFORMATIKA
SEKOLAH TINGGI TEKNOLOGI ADISUTJIPTO
YOGYAKARTA
2019

HALAMAN PENGESAHAN

1.	Judul Pengabdian	:	*Monitoring Interface Fastethernet Pada Cisco Catalyst 3750 Untuk Menjamin Keamanan Penggunaan Jaringan Komputer di Laboratorium Komputasi STTA*
2.	Bidang Penelitian	:	Informatika
3.	Identitas Peneliti		
	a. Ketua Peneliti		
	Nama	:	Sudaryanto, S.T., M.Eng.
	NIDN	:	0511097901
	Jabatan Fungsional	:	Asisten Ahli
	Bidang Keahlian	:	Teknik Informatika
	b. Asisten Peneliti		
	Nama	:	Dwi Nurhayati
	Nim	:	15030016
4.	Lokasi Penelitian		
	a. Tempat	:	Sekolah Tinggi Teknologi Adisutjipto
	b. Kecamatan	:	Banguntapan
	c. Kabupaten	:	Bantul
	d. Propinsi	:	Daerah Istimewa Yogyakarta
5.	Biaya dan Sumber Dana Penelitian	:	Rp. 3.000.000,- (STTA)
6.	Publikasi/Bukti Luaran	:	SENATIK (Seminar Nasional Teknologi Informasi dan Kedirgantaraan) STTA Yogyakjarta

Mengetahui
Waket I

Dedet Hermawan, S.T., M.T.
NIDN. 0521047001

Yogyakarta, 8 November 2019
Peneliti

Sudaryanto, S.T., M.Eng
NIDN. 0511097901

Mengetahui
Ka.P3M

Dr. Okto Dinaryanto, ST, M.M, M.Eng.
NIDN. 0504107202

SURAT KETERANGAN PERPUSTAKAAN

Yang bertanda tangan dibawah ini:

Nama	: Hero Wintolo, S.T., M.Kom.
NIP	: 010303032
Jabatan	: Kepala Perpustakaan
Unit Kerja / PTS	: Perpustakaan STTA

Menerangkan bahwa telah menerima hasil penelitian Sudaryanto S.T., M.Eng. dan Dwi Nurhayati dengan judul:

"*MONITORING INTERFACES FASTETHERNET* PADA CISCO CATALYST 3750 UNTUK MENJAMIN KEAMANAN PENGGUNAAN JARINGAN KOMPUTER DI LABORATORIUM KOMPUTASI STTA"

Dan digunakan sebagai Buku Pustaka dan Bahan Bacaan di Perpustakaan Sekolah Tinggi Teknologi Adisutjipto.

Demikian untuk dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 12 November 2019

Perpustakaan STTA

Kepala

Hero Wintolo, S.T., M.Kom

NIP. 010303032

Ringkasan

Di laboratorium komputasi STTA (Sekolah Tinggi Teknologi Adisutjipto) terdapat banyak perangkat komputer yang terhubung dengan jaringan, hal tersebut berdampak pada peningkatan kebutuhan jaringan dalam hal ini penggunaan *bandwidth*. Sehingga administrator membutuhkan banyak waktu untuk mengelola jaringan seperti *monitoring* dan *management*, contohnya pada perangkat Cisco Catalyst 3750. Hal ini menuntut administrator harus berada di tempat yang sama dengan perangkat jaringan yang sedang dikelola. Oleh karena itu, diperlukan sistem yang dapat memonitoring jaringan dari jarak jauh. Dimana sistem *monitoring interfaces ethernet* bekerja secara *real time* dan mampu mengatasi penggunaan jaringan komputer yang terus mengalami peningkatan. Sistem *monitoring interfaces fastethernet* menerapkan *port security* pada sistem jaringan komputer yang berfungsi sebagai sistem keamanan jaringan komputer agar dapat mengurangi penggunaan jaringan di luar perangkat komputer yang telah didaftarkan atau diijinkan. Sistem *monitoring interfaces fastethernet* yang berbasis Web ini nantinya akan digunakan untuk *monitoring* dan *management* perangkat Cisco Catalyst 3750 dengan memanfaatkan API. Hasil dari pengujian menunjukkan bahwa aplikasi dapat memudahkan administrator dalam mengelola jaringan, karena semua fungsi atau perintah yang ada di sistem *monitoring interfaces fastethernet* sama dengan perintah yang ada di-*command line*. Selain itu, aplikasi *monitoring interfaces fastethernet* dapat berjalan di Browser pada perangkat personal komputer dan *smartphone* secara *responsive*.

Kata kunci : Keamanan Jaringan, *Port security, monitoring, interfaces fastethernet* , Cisco Catalyst, API.

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I
PENDAHULUAN

1.1 Latar Belakang

Teknologi jaringan komputer pada saat ini berkembang dengan sangat cepat, hampir disemua instansi sudah memanfaatkan teknologi jaringan komputer sebagai pendukung perkembangan teknologi informasi yang digunakan. Misalnya di laboratorium komputer yang memanfaatkan teknologi jaringan komputer sebagai pendukung proses pembelajaran saat sedang melakukan praktikum. Dengan banyaknya perangkat komputer yang berada di laboratorium tersebut, maka diperlukan kualitas koneksi yang stabil. Oleh karena itu, digunakan *Local Area Network* (LAN) untuk menghubungkan komputer ke *switch*. Semakin banyak perangkat yang terhubung dengan jaringan komputer, maka kebutuhan jaringan terus mengalami peningkatan dalam penggunaan *bandwidth*. Untuk mengatasi penggunaan jaringan yang di luar perangakat yang telah diijinkan atau didaftarkan, maka diperlukan sebuah sistem yang dapat memonitoring jaringan secara *real time*.

Sudaryanto (2018:258) dalam penelitiannya tentang implementasi *port security* pada sistem keamanan jaringan untuk mengurangi pengguna yang memanfaatkan jaringan Laboratorium Komputasi untuk penggunaan *bandwidth* di luar perangkat komputer yang telah diijinkan atau didaftarkan, tetapi konfigurasinya masih secara manual yaitu menggunakan *command line*. Oleh karena itu, pada penelitian ini akan mencoba membuat Web yang bisa digunakan untuk *monitoring* dan *management port security* dari jarak jauh dengan menggunakan Telnet dan memanfaatkan API yang ada di perangkat Cisco Catalyst 3750.

Monitoring merupakan sebuah kegiatan yang bertujuan untuk memantau tentang perubahan status yang ada disuatu perangkat jaringan. Banyak hal dalam jaringan yang bisa di *monitoring*, salah satu diantaranya adalah status *up* atau *down* dari sebuah perangkat jaringan. Adanya sistem *monitoring* dapat mempermudah administrator jaringan dalam memantau sistem jaringan yang

berada di lapangan dari tempat yang berbeda tanpa harus mengecak secara berkala dan bersentuhan langsung dengan perangkat tersebut.

Faktor yang mempengaruhi kualitas dalam jaringan bisa berasal dari sistem keamanannya, salah satu teknik yang digunakan dalam meningkatkan keamanan jaringan yaitu dengan *port security*. *Port security* ini bertujuan sebagai sistem keamanan jaringan komputer yang ada di laboratorium komputer pada suatu instansi agar mengurangi penggunaan jaringan di luar perangkat komputer yang telah didaftarkan atau diijinkan agar tidak sembarangan menggunakan jaringan yang ada di tempat tersebut.

Switch merupakan sebuah perangkat jaringan yang beroperasi di OSI Layer 2 yaitu layer data *link*. *Switch* digunakan sebagai penyambung atau *concentrator* dalam jaringan. Kelebihan dari *switch* salah satunya yaitu tidak dapat mengalami *collision* karena *switch* dapat mengenal MAC *Address* atau *Physical Address* sehingga dapat memilih data yang akan ditransmisikan. Ada banyak *series* dari *switch*, salah satunya adalah Cisco Catalyst 3750. Cisco Catalyst 3750 ini merupakan *series* terbaru dan *switch* jenis ini dapat digunakan sebagai *switch multilayer*.

Penelitian ini membahas tentang bagaimana *monitoring* dan *management* perangkat Cisco Catalyst 3750 untuk menjamin penggunaan jaringan komputer menggunakan Web yang bisa diakses dari tempat yang berbeda. Untuk mengurangi penggunaan jaringan pada perangkat komputer yang tidak memiliki ijin, maka digunakan kemampuan yang dimiliki pada *switch* berupa *port security* sebagai sistem keamanan jaringan yang ada di laboratorium komputer pada Cisco Catalyst 3750 dengan memprioritaskan MAC *address* yang lama.

1.2 Rumusan Masalah

Dari latar belakang masalah diatas dapat dirumuskan sebuah masalah yaitu sebagai berikut:

1. Bagaimana cara *monitoring* perangkat Cisco Catalyst 3750 secara *real time* dari tempat yang berbeda tanpa harus bersentuhan langsung dengan perangkatnya.

2. Bagaimana cara *management* perangkat Cisco Catalyst 3750 dari tempat yang berbeda tanpa harus bersentuhan langsung dengan perangkatnya.
3. Bagaimana membuat Web yang dapat digunakan untuk *monitoring* dan *management* Cisco Catalyst 3750 dari tempat yang berbeda tanpa harus bersentuhan langsung dengan perangkatnya.

1.3 Batasan Masalah

Berdasarkan rumusan masalah yang telah disebutkan, maka didapatkan batasan masalah sebagai berikut:

1. Web *monitoring* digunakan untuk *management port security* pada Cisco Catalyst 3750.
2. Web *monitoring* digunakan untuk *monitoring port security* pada Cisco Catalyst 3750.
3. Dalam penelitian ini jenis jaringan komputer yang digunakan adalah *Local Area Network* (LAN).

1.4 Tujuan dan Manfaat Penelitian

Adapun tujuan dan manfaat yang diperoleh dari penelitian ini adalah sebagai berikut :

1. Mengurangi penggunaan jaringan komputer di luar perangkat komputer yang telah didaftarkan .
2. Membuat Web yang digunakan untuk *monitoring* Cisco Catalyst 3750 dari tempat yang berbeda tanpa harus bersentuhan langsung dengan perangkatnya.
3. Membuat Web yang digunakan untuk *management* Cisco Catalyst 3750 dari tempat yang berbeda tanpa harus bersentuhan langsung dengan perangkatnya.
4. Meningkatkan sistem keamanan jaringan dengan menggunakan *port security*.
5. Mengurangi peningkatan dalam penggunaan *bandwidth* karena penggunaan jaringan pada perangkat komputer yang tidak diijinkan atau didaftarkan.

6. Membantu administrator jaringan dalam memonitoring *switch* secara *real time* dari tempat yang berbeda tanpa harus bersentuhan langsung dengan perangkatnya.
7. Mengurangi peningkatan dalam penggunaan *bandwidth* karena penggunaan jaringan pada perangkat komputer yang tidak diijinkan atau didaftarkan.

1.5 Metodologi Penelitian

Metodologi penelitian yang digunakan dalam penelitian ini adalah :

1. Pengumpulan Data
   a. Observasi

      Pada metode ini penulis mengamati, menyaksikan dan memperhatikan secara langsung kejadian-kejadian yang ada disuatu laboratorium.

   b. Wawancara

      Data diperoleh dari beberapa proses yaitu data dari proses wawancara dengan administrator yang berada disuatu laboratorium.

   c. Studi Literatur

      Bertujuan untuk mempelajari teori-teori dengan membaca beberapa buku dan jurnal yang berhubungan dengan permasalahan yang dibahas. Kususnya kajian mengenai *switch port security*.
2. Perancangan
   a. Menguji fungsi atau perintah secara manual yaitu langsung pada perangkat Cisco Catalyst 3750. Beberapa perintah yang dicoba atau diuji adalah *sticky port security, violation* dan menghapus konfigurasi.
   b. Merancang fungsi atau perintah yang telah dicoba secara manual dan mengimplementasikannya ke dalam bahasa pemrograman PHP dengan memanfaatkan API yang dimiliki oleh Cisco Catalyst 3750.
3. Implementasi dan Uji Coba

   Pada tahapan metode ini penulis mengimplementasikan dan menguji aplikasi yang dibuat pada perangkat Cisco Catalyst 3750 yang akan di *monitoring*.

BAB II
TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Menurut Ocanitra dan Ryansyah (2019:53) *Switch* adalah perangkat yang juga berfungsi untuk menghubungkan *multiple* komputer. *Switch* secara fisik sama dengan hub tetapi logikalnya sama dengan barisan *bridge*. Peningkatan kecerdasan dibandingkan hub, yaitu memiliki pengertian terhadap alamat *Medium Access Control* (MAC) atau pada *link* layer model OSI sehingga hanya mengirimkan data pada *port* yang dituju (*unicast*). Proses kerjanya adalah apabila paket data datang, *header* dicek untuk menentukan di *segment* mana tujuan paket datanya. Kemudian data akan dikirim kembali (*forwaded*) ke *segment* tujuan tersebut.

Sudaryanto (2018:86) menyatakan bahwa *Switch Non Multilayer* merupakan jenis *switch* yang hanya bekerja dilayer data *link*, fitur yang paling sering digunakan adalah kemampuan *switch* dalam konfigurasi *Virtual* LAN (VLAN) dan *traffic* jaringan yang bisa dikontrol atau diatur. Sedangkan *switch multilayer* merupakan jenis *switch* yang mampu bekerja dilayer data *link* dan layer *network*. Pada layer data *link* kemampuan *switch* sama dengan *Switch Non Multilayer* sedangkan pada layer *network* bisa di konfigurasi sesuai dengan keinginan administrator dan juga dapat dilakukan proses *routing* ataupun menghubungkan alamat *network* yang berbeda.

Sudaryanto (2018:258) menyebutkan bahwa *switch manageable* yang ditunjukkan pada Gambar 2.2 mempunyai fungsi yang sama dengan *switch unmanageable* namun banyak fitur-fitur tambahan yang dapat membedakan *switch unmanageable* dalam meningkatkan kualitas dari jaringan tersebut, contoh fitur yang paling sering digunakan adalah kemampuan *switch* dalam konfigurasi *Virtual* LAN (VLAN) dan *traffic* jaringan yang bisa dikontrol atau diatur, *switch* ini juga dapat melakukan proses *routing*, *switch* ini juga dapat digunakan untuk meningkatkan keamanan dengan menggunakan kemampuan *switch port security* yang berfungsi untuk menangani hak akses ke jaringan tersebut berdasarkan *port–*

port yang dimiliki oleh *switch* tersebut, berbeda halnya dengan *switch unmanageable* ditunjukkan pada Gambar 2.1 yang hanya bekerja di layer data *link* atau layer 2 pada *switch* jenis ini tidak bisa melakukan konfigurasi.

Gambar 2.1 *Switch Unmanageable*
(Sumber : https://www.cisco.com/c/en/us/support/*switch*es/sg100-24-24-port-gigabit-*switch*/model.html)

Gambar 2.2 *Switch Manageable*
(Sumber : https://www.cisco.com/c/en/us/support/*switch*es/sf350-24p-24-port-10-100-poe-managed-*switch*/model.html#~tab-downloads)

Salah satu faktor yang mempengaruhi kualitas dalam jaringan adalah sistem keamanan, banyak teknik yang dapat dilakukan dalam meningkatkan keamanan jaringan, baik dengan menggunakan layer 7 *protocol*, dengan membangun sistem *firewall* maupun dengan *port security*. Dengan adanya *port security*, *port–port* yang ada dapat dimanfaatkan untuk mengizinkan akses ke jaringan. *Switch port security* merupakan suatu kemampuan perangkat *switch* untuk mengamankan jaringan *Local Area Network* (LAN).

Sedangkan menurut Sulaiman (2016:10) kemampuan *switch manageable* untuk meningkatkan keamanan jaringan dengan menggunakan *port-port* yang tersedia pada *switch* ada 3 jenis *switch port security* yaitu:

1. *Default/static port security*, ketika *port security* ini difungsikan maka *mac address port security* akan diaktifkan pada *port switch,* sehingga *port* tidak akan mem-forward paket jika *source address* bukanlah *address* yang telah kita definisikan atau tentukan sebelumnya. Menentukan alamat MAC tertentu yang diperbolehkan untuk terhubung ke *port* tersebut secara manual.
2. *Port security dynamic learning,* MAC *address* dipelajari secara dinamis ketika perangkat terhubung ke *switch*, MAC *address* tersebut disimpan di MAC *address table*.
3. *Sticky port security,* sebuah kemampuan *switch* dalam mengenal MAC *address* tiap-tiap perangkat yang terhubung dan akan memblok setiap MAC yang melebihi dari MAC yang telah terdaftar.

Herliana dan Rasyid (2016:43) menyatakan bahwa *monitoring* merupakan sebuah kegiatan untuk menjamin akan tercapainya semua tujuan organisasi dan manajemen. Dalam kesempatan lain, *monitoring* juga didefinisikan sebagai langkah untuk mangkaji apakah kegiatan yang dilaksanakan telah sesuai dengan rencana, mengidentifikasi masalah yang timbul agar langsung diatasi, melakukan penilaian apakah pola kerja dan manajemen yang digunakan sudah tepat untuk mencapai tujuan untuk memperoleh ukuran kemajuan. Dengan kata lain, *monitoring* merupakan salah satu proses didalam kegiatan organisasi yang sangat penting yang dapat menentukan terlaksana atau tidaknya sebuah tujuan organisasi. Tujuan dilakukannya *monitoring* adalah untuk memastikan agar tugas pokok organisasi dapat berjalan sesuai dengan rencana yang telah ditentukan.

Menurut gobel et al. (2019:79) *notification* adalah pemberitahuan, dalam hal ini pemberitahuan yang dimaksud adalah pemberitahuan kepada administrator. Notifikasi dalam hal ini berisi tentang informasi-informasi yang dibutuhkan oleh pengelola jaringan untuk tujuan-tujuan tertentu misalnya dalam pengawasan jaringan. Informasi yang terdapat didalam notifikasi terdiri dari beberapa bagian dari paket data yang keluar masuk dijaringan komputer melalui *proxy*. Informasi yang diberitahukan adalah sebagai berikut:

1. *Time,* waktu pada saat paket data dikirimkan.

1. IP *Source,* nomor IP pengirim paket data.
2. *Host Name*, nama komputer pengirim paket data.
3. *Service, port service* yang digunakan (Telnet, SSH, HTTP, dll).
4. *Port,* nomor *port* yang digunakan.

Informasi ini dikirimkan sebagai notifikasi kepada pengelola yang menjadi bahan untuk pengawasan jaringan komputer atau tujuan lainnya.

2.2 Landasan Teori

2.2.1 Cisco Catalyst 3750

Switch Cisco Catalyst 3750 Series adalah garis utama *switch* kelas perusahaan, *stackable*, *multilayer* yang menyediakan ketersediaan tinggi, keamanan, dan kualitas layanan (QoS) untuk meningkatkan operasi jaringan. Manajemen tumpukan terpadu yang inovatif meningkatkan standar dalam manajemen tumpukan, redundansi, dan *failover*. Dengan serangkaian konfigurasi Fast Ethernet dan Gigabit Ethernet, Cisco Catalyst 3750 Series dapat berfungsi sebagai saklar lapisan akses yang kuat untuk lemari kabel perusahaan menengah dan sebagai saklar tulang punggung untuk jaringan ukuran menengah. Pelanggan dapat menggunakan layanan cerdas jaringan yang luas, seperti QoS tingkat lanjut, pembatasan tingkat, daftar kontrol akses keamanan Cisco (ACL), manajemen multicast, dan perutean IP berkinerja tinggi sambil mempertahankan kesederhanaan *switching* LAN tradisional. Tertanam dalam Cisco Catalyst 3750 Series adalah perangkat lunak Cisco *Cluster Management Suite* (CMS), yang memungkinkan pengguna untuk secara bersamaan mengkonfigurasi dan memecahkan masalah beberapa sakelar desktop Cisco Catalyst menggunakan Browser Web standar (Cisco, 2012).

2.2.2 Ethernet

Ethernet adalah metode media akses agar memperbolehkan semua host di dalam jaringan untuk *share bandwidth* dalam suatu *link* . Ethernet merupakan salah satu alat (media komunikasi) yang dipasang di dalam CPU pada PCI *slot*. Ini berfungsi untuk menghubungkan kabel dalam jaringan dan memungkinkan terjadi

koneksi internet, intranet, atau ekstranet. Ethernet adalah salah satu skenario pengkabelan dan pemrosesan sinyal untuk data dalam jaringan. Sebenarnya ada berbagai metode akses yang digunakan dalam jaringan diantaranya, Ethernet, FDDI, Token Ring, *Wireless* LAN, *Bridging*, dan Virtual Bridged LAN. Masing-masing metode mempunyai *interface* yang berbeda beda. *Interface* yang digunakan pada ethernet disebut ethernet card. Ada berbagai macam *interface* untuk ethernet berdasarkan media transmisi yang digunakan, ini akan dibahas pada topik selanjutnya. Ethernet menjadi populer karena mudah sekali disesuaikan dengan kebutuhan (*scalable*), artinya cukup mudah untuk mengintegrasikan teknologi baru ke dalam infrastruktur *network* yang ada. Ada banyak metode-metode lain yang lebih cepat dari ethernet, namun dari sisi harga untuk *interface-interface* ethernet sangat terjangkau sehingga sampai sekarang ethernet masih menjadi pilihan kebanyakan orang. Selain murah, ethernet sangat banyak beredar di pasaran, tidak terlalu sulit untuk mendapatkannya (Zulfi).

2.2.3 *Aplication Programming Interface* (API)

Menurut Pratama (2014:150) agar proses di *client* dan proses di *server* dapat saling berkomunikasi dengan baik, maka perlu adanya sejumlah instruksi di dalam program atau aplikasi yang dijalankan. Program ini ditulis dengan bahasa pemrograman, yang di dalamnya berisi sekumpulan instruksi untuk operasi matematika, manipulasi *string*, dan lain-lain. Sehubungan dengan bahasa pemrogramandan pembuatan sebuah aplikasi dengan menggunakan bahasa pemrograman, tentulah terdapat API di dalamnya. API merupakan sekumpulan instruksi yang berada didua *entitas*, yaitu *entitas* yang berjalan disisi *Application Layer* dan yang berjalan di sisi sistem operasi. Definisi lain dari API merupakan kumpulan dari beragam perintah, fungsi, dan protokol di dalam jaringan komputer, yang bekerja sama di dalam menjadikan aplikasi untuk dapat berkomunikasi dengan sistem operasi, perangkat keras komputer, dan komputer lainnya dalam jaringan komputer. Selain itu API juga merupakan kombinasi dari beragam perintah dan prosedur yang terurut, sehingga memudahkan di dalam Teknik-teknik penyerangan, misalkan dengan DNS *Poisoning*, SSL *Vulnerability*,

pengembangan perangkat lunak. Ada beberapa manfaat dari API antara lain sebagai berikut:

1. API memudahkan di dalam proses komunikasi. Beberapa diantaranya meliputi *socket interface, Transport Layer Interface* (TLI), dan *Stream*.
2. API memudahkan pengembang (*programmer*) di dalam pembuatan aplikasi yang mana aplikasi tersebut nantinya dapat berjalan dengan baik sebagaimana mestinya. Misalkan melalui sejumlah prosedur, fungsi, tombol, pustaka, yang disediakan dan digunakan.
3. API membantu aplikasi yang dikembangkan untuk dapat berkomunikasi dengan baik kepada sistem operasi, perangkat keras komputer, dan juga komputer-komputer lainnya yang terhubung di dalam jaringan.

2.2.4 Jaringan Komputer

Sofana (2015:3) menyatakan bahwa jaringan komputer (*computer networks*) adalah suatu himpunan interkoneksi sejumlah komputer *autonomous*. Dalam bahasa yang populer dapat dijelaskan bahwa jaringan komputer adalah sekumpulan beberapa komputer (dan perangkat lain seperti *router, switch*, dan sebagainya) yang saling terhubung satu sama lain melalui media perantara. Media perantara ini bisa berupa media kabel ataupun media tanpa kabel (*nirkabel*). Informasi berupa data akan mengalir dari satu komputer ke komputer lainnya atau dari satu komputer ke perangkat yang lain, sehingga masing-masing komputer yang terhubung tersebut bisa saling bertukar data atau berbagi perangkat keras.

2.2.5 Keamanan Jaringan

Menurut Pratama (2014:628) keamanan sistem secara umum terbagi atas tiga aspek utama yaitu sebagai berikut:

2.2.5.1 Keamanan Jaringan Komputer Dilihat dari Sisi Sistem

Keamanan jaringan komputer dilihat dari sisi sistem memiliki arti bahwa sistem menjadi fokus utama di dalam mewujudkan keamanan jaringan komputer. Sistem dalam hal ini meliputi segala perangkat keras komputer (*hardware*),

perangkat lunak komputer (*software*), perangkat keras penghubung jaringan, pustaka (*library*), jaringan komputer (intranet, internet), protokol di dalam jaringan komputer, dan media transmisi (kabel, *wireless*) yang digunakan, yang membentuk sistem dari jaringan komputer itu sendiri. Kajian keamanan jaringan komputer dari sisi sistem merupakan kajian yang paling banyak diminati dan dibahas, baik oleh para ilmuan, akademisi, praktisi, hingga masyarakat umum. Fokus keamanan pada jaringan komputer disisi sistem tidak akan habis untuk dibahas. Hal ini disebabkan oleh enam buah faktor berikut ini:

a. Perkembangan perangkat keras komputer (*hardware*) dan perangkat lunak komputer (*software*) dari waktu kewaktu, sehingga memunculkan banyak *Bugs* dan *Vulnerability* (celah keamanan) terutamanya pada versi lama. penemuan-penemuan celah keamanan ini dapat dilakukan oleh *Pentester* dari kalangan komunitas (cuma-cuma) maupun Profesional (berbayar), *hacker*, maupun *cracker* (yang memanfaatkan untuk tujuan pribadi).
b. Luasnya cakupan bidang yang dibahas di dalam sistem. Terdapat banyak sekali, ratusan hingga ribuan, celah dan ancaman keamanan yang ada disejumlah protokol di dalam jaringan komputer (FTP, HTTP, SSL), pada setiap *layer* di dalam jaringan komputer (*Application Layer* hingga *Physical Layer*), dan pada setiap aplikasi dan layanan pada jaringan komputer (Web *Server, Mail Server,* DNS *Server*).
c. Pada industri atau perusahaan, investasi dalam jumlah besar dilakukan pada sisi sistem. Misalkan saja layanan Bank yang mengelontarkan dana dalam jumlah besar untuk peningkatan keamanan PIN dan *E-banking* (untuk menunjang keamanan transaksi nasabah), penyedia Data *Center*, dan sebagainya.
d. Beragam Web di dalam dan di luar, media cetak, hingga media elektronik, di dalam membahas tentang keamanan jaringan komputer, lebih banyak membahas disisi keamanan sistem. Misalkan keamanan Web *server* (kasus *Deface* pada sejumlah Web pemerintah), DNS *server* (kasus situs Presiden SBY) dan sebagainya.

e. Makin banyaknya perangkat lunak komputer (aplikasi) siap pakai yang dapat memudahkan pengguna komputer untuk melakukan eksplorasi terhadap keamanan di dalam jaringan komputer.
f. Masih banyak pengguna komputer yang tidak memiliki pemahaman yang cukup mengenai keamanan jaringan komputer dari sisi sistem.

Teknik-teknik penyerangan, misalkan dengan DNS *Poisoning*, SSL *Vulnerability*, ARP *Attack*, *Man in the middle Attack*, IP *Spoofing*, dan sebagainya, merupakan bentuk-bentuk ancaman keamanan pada jaringan komputer disisi sistem. Sedangkan solusi-solusi berupa ARP *Table, Honeypot, Honeynet, Honeywall, Firewall, Deep Packet Inspection*, dan sebagainya, merupakan bentuk-bentuk solusi yang diberikan terkait dengan keamanan pada jaringan komputer disisi sistem.

2.2.5.2 Keamanan Jaringan Komputer Dilihat dari Sisi Pengguna

Keamanan jaringan komputer dilihat dari sisi pengguna memiliki arti bahwa pengguna (*user*) menjadi fokus utama di dalam mewujudkan keamanan jaringan komputer. Pengguna dalam hal ini meliputi hirarki pengguna tertinggi (misalkan *System Administrator*) hingga pengguna terbawah (pengguna biasa). Di dalam suatu organisasi (perusahaan, instansi pemerintahan, perguruan tinggi, sekolah dan sebagainya), baik dari tingkat pimpinan hingga pegawai terbawah dan semua pengguna lainnya, wajib peduli terhadap masalah keamanan jaringan komputer. Pengguna pada konteks jaringan komputer merupakan individu yang memanfaatkan fitur dan menu yang disediakan aplikasi dan layanan pada jaringan komputer sesuai dengan kebutuhan masing-masing, hingga yang turut melakukan pengujian sistem dan pengembangan di dalamnya.

Keamanan di dalam jaringan komputer tidak akan terwujud dengan baik jika hanya ditunjang dengan sistem yang aman saja tanpa adanya pengguna yang peduli (*Aware*) terhadap masalah keamanan. Pengguna dari tingkat tertinggi hingga pengguna biasa pun wajib peduli dengan masalah keamanan pada jaringan komputer, khususnya keamanan data dan informasi. Beberapa contoh antara lain sebagai berikut:

a. Sebuah layanan *E-commerce* menyediakan keamanan sistem yang sangat baik untuk transaksi elektronik, namun tidak didukung dengan konsumen (pengguna internet) yang peduli dengan keamanan pada jaringan komputer. Sehingga memunculkan kasus dimana terjadi sabotase akun salah satu nasabah karena *password* yang lemah atau tidak diperbarui secara berkala.
b. Sebuah Bank telah berinvestasi besar-besaran dibidang Teknologi Informasi untuk penyedia layanan keamanan jaringan komputer. Terbukti bahwa sistem pada bank ini mampu memberikan keamanan untuk PIN dan *E-Banking* (SMS *Banking,* Internet *Banking*). Hanya saja pengguna (nasabah) dalam hal initidak peduli dengan keamanan pada jaringan komputer. Sehngga memunculkan kasus dimana ada nasabah yang dikuras isi rekeningnya karena teledor menyimpan PIN, tidak melakukan pergantian PIN secara berkala, tidak teliti sebelum menggunakan kartu kredit dan kartu ATM (kasus *Skimming*), maupun tidak teliti di dalam layanan berbasis Web dan sms (penipuan, *Man in the middle Attack*).

2.2.5.3 Keamanan Jaringan Komputer Dilihat dari Sisi Kebijakan

Keamanan jaringan komputer dilihat dari sisi kebijakan memiliki arti bahwa di dalam proses mewujudkan keamanan pada jaringan komputer, fokus yang diutamakan adalah pada sisi kebijakan. Kebijakan dalam hal ini dapat berupa perancanaan, peraturan (regulasi), dan tindakan-tindakan, yang harus ditaati bersama. Pada sisi kebijakan ini, terdapat sejumlah level yang harus diperhatikan antara lain:

a. Level Regulasi

   Level regulasi memuat sejumlah regulasi atau aturan-aturan yang mengatur tentang penyedia dan penggunaan layanan internet, elektronik, digital termasuk juga transaksi digital di dalamnya. Misalkan satu saja untuk di Indonesia, level regulasi ini diwujudkan dalam bentuk Undang-Undang Informasi dan Transaksi Elektronik (UU ITE), Peraturan

Pemerintah (PP) No 82 Tahun 2012, PP No 5 Kominfo 2011, dan Peraturan Bank Indonesia (PBI) Tahun 2007.

b. Level Standarisasi

Pada level standarisasi terdapat satu atau beberapa buah standarisasi yang umum digunakan di dalam keamanan pada jaringan komputer. Misalkan standarisasi untuk menajemen risiko keamanan Teknologi Informasi (IT *Risk Management*).

c. Level Pedoman

Pada level pedoman disediakan satu atau beberapa buah patokan atau pedoman dan bentuk format (*Templete*) penulisan sebuah aturan, standarisasi, Undang-Undang, prosedur, di dalam mewujudkan kebijakan untuk keamanan di dalam jaringan komputer. Misalkan dokumen pedoman yang memuat tata cara untuk melakukan transaksi elektronik (*E-commerce, E-business*) yang baik dan benar atara pihak penjual dan pihak pembeli.

d. Level Prosedur

Pada level prosedur diatur tentang urutan langkah-langkah yang harus dilakukan oleh pengguna jaringan komputer di dalam mewujudkan keamanan pada jaringan komputer. Urutan langkah-langkah pada level prosedur ini dibuat berdasarkan regulasi, standarisasi, dan pedoman yang telah dibahas diatas. Misalkan saja, bagaimana prosedur di dalam melakukan pengiriman data dan informasi pada jaringan internet (sebagai jaringan publik) secara terenkripsi agar lebih aman, bagaimana prosedur di dalam membuat *password* yang kuat dan penggantiannya secara berkala.

2.2.6 *Security*

Menurut Sofana (2012:362) *security* pada *network* lebih banyak terfokus pada penggunaan *firewall*, *proxy,* atau berbagai cara untuk mengatasi serangan *Layer* 3. *Security* semacam ini menggunakan sebuah asumsi bahwa serangan akan selalu dilakukan dari *network* eksternal. Diasumsikan bahwa tidak ada yang seorang pun yang akan melakukan serangan “jarak dekat”. Sebenarnya *network*

juga harus terlindungi dari berbagai serangan *Layer* 2. Misalkan saja, seorang karyawan yang nakal, dapat menambahkan perangkat disalah satu *switch* untuk mendapatkan lebih banyak akses *bandwidth* LAN atau VLAN. Perangkat ilegalyang dapat ditambahkan antara lain: *Wireless routers* atau *Access point*, *Access switch,* dan *Hub.* Bisa saja perangkat tersebut menjadi *Root bridge* yang menyebabkan masalah pada *network traffic*. Beberapa jenis serangan lain yang dapat menimbulkan masalah pada *switch* yaitu: MAC *address-based attacks, Spoofing attacks,* VLAN-*based attacks*.

1. MAC *address-based attacks*

   Serangan yang dilakukan dengan cara "membanjiri" MAC *address* (MAC *address flooding*). Teknik yang dilakukan pada MAC *address flooding* yaitu si penyerang mengisi tabel *Content Addressable Memory* (CAM) pada *switch*, dengan MAC *address* yang tidak *valid*. Setelah tabel CAM penuh, *traffic* yang *address*-nya tidak ada ditabel akan di-*flood* kesemua *interface*. Efek yang ditimbulkan adalah meningkatnya beban (*load*) pada *traffic* dan meningkatnya *traffic* pada LAN. Kondisi ini juga dapat menyebabkan tabel CAM milik *switch* yang berdekatan mengalami *overflow*. Untuk mengatasi atau mengurangi MAC *address attack* dapat digunakan teknik yang disebut *Port Security* dan *Port-based authentication.*

2. *Spoofing attacks*

   Spoofing dilakukan oleh penyerang untuk "menipu" *switch* atau *host* lain, sehingga pengguna mengira bahwa komputer penyerang adalah sebuah *gateway*. Penyerang dapat mengutak-atik paket yang datang sebelum mengirimkannya kembali. Teknik yang digunakan untuk mengatasi *spoofing* adalah DHCP *snooping,* IP *source guard, dynamic* ARP *inspection*.

3. VLAN-*based attacks*

   Beberapa jenis serangan VLAN dan teknik pencegahannya yaitu *Switch Spoofing, 802.1Q Double-Tagging, Private* VLAN, *Protected Port.*

2.2.7 *Port Security*

Menurut Sofana (2012:364) *Port Security* membatasi jumlah MAC *address* yang diizinkan terhubung dengan tiap *port* dan juga dapat membatasi MAC *address* mana saja yang diizinkan. Perintah yang digunakan untuk menampilkan informasi *port-security* dapat dilihat pada Gambar 2.3.

```
Switch>en
Switch#sh port-security int fa0/1
Port Security              : Disabled
Port Status                : Secure-down
Violation Mode             : Shutdown
Aging Time                 : 0 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 1
Total MAC Addresses        : 0
Configured MAC Addresses   : 0
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0000.0000.0000:0
Security Violation Count   : 0
```

Gambar 2.3 Perintah Menampilkan Informasi *Port Security*

Secara *default* fitur *port security* ini dalam kondisi *disable* atau belum aktif. *Port security* diimplementasikan pada *interface* tertentu. Untuk mengaktifkannya digunakan perintah yang dapat dilihat pada Gambar 2.4.

```
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int fa0/1
Switch(config-if)#switchport mode access
Switch(config-if)#switchport port-security
Switch(config-if)#end
Switch#
%SYS-5-CONFIG_I: Configured from console by console

Switch#sh port-security int fa0/1
Port Security              : Enabled
Port Status                : Secure-down
Violation Mode             : Shutdown
Aging Time                 : 0 mins
Aging Type                 : Absolute
SecureStatic Address Aging : Disabled
Maximum MAC Addresses      : 1
Total MAC Addresses        : 0
Configured MAC Addresses   : 0
Sticky MAC Addresses       : 0
Last Source Address:Vlan   : 0000.0000.0000:0
Security Violation Count   : 0
```

Gambar 2.4 Perintah Mengaktifkan *Port Security*

Menentukan jumlah MAC *address* yang diizinkan mengakses *port*. *Default*-nya adalah hanya 1 buah MAC *address*. Misalkan akan diubah menjadi 4 buah. Contoh perintahnya dapat dilihat pada Gambar 2.5.

```
Switch(config-if)#sw
Switch(config-if)#switchport po
Switch(config-if)#switchport port-security ma
Switch(config-if)#switchport port-security max
Switch(config-if)#switchport port-security maximum 4
```

Gambar 2.5 Perintah Jumlah MAC *Address*

Kita juga bisa menentukan langsung MAC *address* yang dibolehkan untuk mengakses *port*. Misalkan saja MAC *address*-nya 0016.4302.A742 dan apabila tidak tahu MAC *address*-nya bisa dengan perintah *dynamic* yaitu *sticky*. Contoh perintahnya dapat dilihat pada Gambar 2.6.

```
Switch(config-if)#sw
Switch(config-if)#switchport po
Switch(config-if)#switchport port-security mac
Switch(config-if)#switchport port-security mac-address
0016.4302.A742 | sticky
```

Gambar 2.6 Perintah Menentukan MAC *Address*

Apabila menjumpai ada MAC *address* asing atau jumlah MAC *address* melebihi apa yang diizinkan, maka aksi apa yang akan dilakukan selanjutnya dapat ditentukan dengan perintah yang dapat dilihat pada Gambar 2.7.

```
Switch(config-if)#sw
Switch(config-if)#switchport po
Switch(config-if)#switchport port-security vio
Switch(config-if)#switchport port-security violation shutdown |
restrict | protect
```

Gambar 2.7 Perintah Menentukan Aksi

Setting aksi *default*-nya adalah *shutdown*. Berikut merupakan penjelasan dari masing-masing opsi sebagai berikut:

1. *Shutdown* : port akan segera pindah ke *Err-disable state*, yaitu secara efektif *port* di-*shutdown*.Untuk meng-*enable*-nya kembali harus dilakukan secara manual. Contoh perintahnya dapat dilihat pada Gambar 2.8.

```
Switch(config)#int fa0/1
Switch(config-if)#shutdown

%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to
administratively down
Switch(config-if)#no shutdown

Switch(config-if)#
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1,
changed state to up
```

Gambar 2.8 Perintah Meng-e*nable*-kan *Port*

2. *Restrict* : *port* tetap dalam kondisi *up*, namun semua paket yang datang dari MAC *address* yang tidak diizinkan akan langsung di-*drop*. *Switch* dapat me-*record* paket yang terlarang, kemudian mengirim pesan *violation.*
3. *Protect* : mirip dengan opsi *restrict* hanya saja tidak ada pesan apapun.

Port yang dalam kondisi terproteksi dapat dikembalikan kekondisi semula dengan perintah yang dapat dilihat pada Gambar 2.9.

```
Switch#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#int fa0/1
Switch(config-if)#switchport mode access
Switch(config-if)#switchport port-security
Switch(config-if)#no switchport port-security
```

Gambar 2.9 Perintah Menghapus *Port Security*

2.2.8 *Switch*

Menurut Pratama (2014:487) *Switch* merupakan perangkat keras penghubung di dalam jaringan komputer yang lebih banyak digunakan saat ini dibandingkan hub. Hal ini disebabkan karena dengan fungsi yang serupa dengan hub, *Switch* memiliki dua buah kelebihan utama dibandingkan hub. Kelebihan-kelebihan yang dimiliki oleh *switch* yaitu:

4. *Switch* memiliki kemampuan untuk membaca alamat fisik (MAC *Address*) dari setiap komputer yang terhubung ke dalam *switch* bersangkutan. *Switch* menyimpan alamat fisik (MAC *Address*) dari setiap komputer yang terhubung ke dalam *switch* tersebut beserta dengan nomor *port switch* yang digunakan oleh komputer bersangkutan. Misalkan pada *switch* dengan jumlah *port* sebanyak 8, terdapat komputer pertama yang

menggunakan *port* nomor empat pada *switch* dan komputer bersangkutan memiliki alamat fisik di dalam jaringan komputer (MAC *Address*) 66:2B:18:55:78. Kemudian pada *port* nomor enam *switch* digunakan oleh komputer kedua dengan alamat fisik dalam jaringan komputer (MAC *Address*) 88:4G:43:66:28. Maka oleh *switch*, data pemakaian nomor *port* dan alamat fisik (MAC *Address*) dari kedua komputer ini akan disimpan ke dalam sebuah *Switching Table*. *Switching Table* memuat dua buah kolom, yaitu kolom untuk *Address* (MAC *Address*) dan kolom kedua untuk nomor *port* yang digunakan pada *switch*. Contoh *Switching Table* dapat dilihat pada Tabel 2.1.

Tabel 2.1 Contoh *Switching Table*
(Sumber : Pratama, I. P. A. E, 2014)

No	*Address* (MAC *Address*)	Nomor *Port Switch*
1	66:2B:18:55:78	4
2	88:4G:43:66:28	6

5. *Switch* memiliki kemampuan untuk melakukan filter terhadap paket data yang keluar masuk *switch*. Hal ini akan memberikan keamanan paket data (terkait dengan pengendusan paket data di dalam jaringan komputer). Hal terpenting lainnya adalah memberikan kemudahan di dalam memberikan informasi mengenai alamat tujuan untuk komputer penerima (*Destination Address*) serta kemampuan untuk ikut menentukan *port* mana yang digunakan untuk keluar menuju ke komputer tujuan (*Outgoing Port*).

Switch bekerja di dua buah layer pada jaringan komputer, yaitu Data *Link Layer* dan *Physical Layer*. Pada Data *Link Layer*, terjadi proses pengecekan terhadap alamat fisik jaringan (MAC *Address*) untuk otentikasi alamat fisik komputer yang terhubung ke *switch*, untuk kemudian disesuaikan dengan alamat jaringan pada *Network Layer* (IP *Address*). Pada *Physical Layer* terjadi proses pengolahan sinyal digital.

2.2.9. *Telecommunications Network* (Telnet)

Menurut Pratama (2014:109) Telnet merupakan protokol di dalam jaringan komputer yang digunakan untuk komunikasi, kendali dan konfigurasi komputer jarak jauh (*remote*), bahkan pada beberapa kasus Telnet juga dapat digunakan untuk menikmati layanan hiburan (meskipun dalam Mode *Text*/ASCII). telnet terkoneksi ke dalam jaringan komputer dengan menggunakan *port* 23. Sebagaimana SSH, Telnet menyajikan kemampuan untuk membantu anda di dalam kendali komputer jarak jauh (*remote*), baik untuk pengambilan dan transfer file maupun konfigurasi sistem. Hanya saja, satu hal yang menjadi kekurangan Telnet adalah kesederhanaan sistemnya, sehingga tidak ada enkripsi terhadap akun yang kita gunakan saat *login* ke dalam sistem.

Beda dengan SSH, dimana proses *login* ke mesin *remote* dilakukan proses enkripsi sehingga menjadi lebih aman, terutama untuk jenis serangan *Man In The Middle* (MITM). Seiring dengan perkembangan teknologi, protokol Telnet pun juga mulai dikembangkan menjadi salah satu bentuk layanan multimedia jaringan. Misalkan saja untuk menonton film secara *online* (*streaming*). Tentu saja, dengan keterbatasan protokol Telnet, film animasi yang disajikan dalam bentuk ASCII (sejumlah karakter). Telnet dan SSH merupakan dua buah protokol sekaligus aplikasi *client server* di internet yang umum digunakan untuk melakukan *remote login* ke komputer lain. Dibandingkan SSH, Telnet lebih rentan dari sisi keamanan, Telnet memerlukan akun untuk *login* (*username* dan *password*), namun Telnet mengirimkan semua data kepada pengguna (termasuk juga *password plaintext* tanpa enkripsi.

2.2.10. *HyperText Preprocessor* (PHP)

Sidik (2014:4) menyatakan bahwa PHP merupakan secara umum dikenal sebagai bahasa pemrograman *script-script* yang membuat dokumen HTML secara *on the fly* yang dieksekusi di *server* Web, dokumen HTML yang dihasilkan dari suatu aplikasi bukan dokumen HTML yang dibuat dengan menggunakan editor teks atau editor HTML. Dikenal juga sebagai bahasa pemrograman *server* side. Dengan menggunakan PHP maka *maintenance* suatu situs Web menjadi lebih

mudah. Proses *update data* dapat dilakukan dengan menggunakan aplikasi yang dibuat dengan menggunakan *script* PHP. PHP/FI merupakan nama awal dari PHP. *Personal Home Page* (PHP), FI adalah *Form Interface*. Dibuat pertama kali oleh Rasmus Lerdoff. PHP awalnya merupakan program CGI yang dikhususkan untuk menerima input melalui form yang ditampilkan dalam browser Web. *Software* ini disebarkan dan dilisensikan sebagai perangkat lunak Open Source. Integrasi PHP dengan *server* Web dilakukan dengan teknik CGI, FastCGI, dan modul *server* Web. Teknik CGI dan FastCGI memisahkan antara *server* Web PHP; sedangkan modul *server* Web menjadi PHP sebagai bagian dari *server* Web. Kini, PHP adalah kependekan dari PHP:*HyperText Preprocessor* (rekursif, mengikuti gaya penamaan di *nix), merupakan bahasa utama *script server-side* yang disisipkan pada HTML yang dijalankan di *server*, dan juga bisa digunakan untuk membuat aplikasi desktop.

2.2.11 *Unified Modeling Language* (UML)

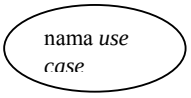
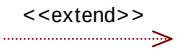
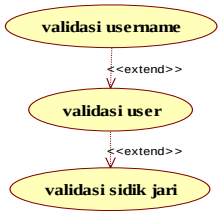
Menurut Rosa dan Salahuddin (2015:133) UML adalah bahasa yang digunakan untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemroograman berorientasi objek. UML muncul karena adanya kebutuhan pemodelan visual untuk mempresentasikan, menggambarkan, membangun dan dokumentasi dari sistem perangkat lunak. UML merupakan bahasa visual untuk pemodelan dan komunikasi mengenai sebuah sistem dengan menggunakan diagram dan teks-teks pendukung. UML hanya berfungsi untuk melakukan pemodelan. Jadi penggunaan UML tidak terbatas pada metodologi tertentu, meskipun pada kenyataannya UML paling banyak digunakan pada metodologi berorientasi objek.

2.2.11.1 *Use Case* Diagram

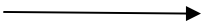
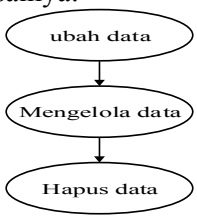
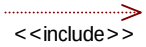
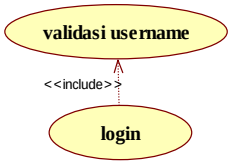
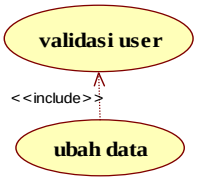
Use Case atau diagram *use case* merupakan pemodelan untuk melakukan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada di

dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Simbol-simbol yang biasa digunakan dalam membuat *use case* diagram dapat dilihat pada Tabel 2.2.

Tabel 2.2 Simbol *Use Case* Diagram
(Sumber : Rossa A.S dan M. Salahuddin, 2015)

Simbol	Deskripsi
Use Case 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal *frase* nama *use case*.
Aktor/*actor* 	Orang, proses, atau sistem lain yang berinteraksi dengan sistem di luar sistem yang akan dibuat itu sendiri. Walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal *frase* nama aktor.
Asosiasi/*association* ————	Komunikasi antara aktor dan *use case* yang berpartisipasi pada *use case* atau *use case* memiliki interaksi dengan aktor.
Ekstensi / *extend* 	Relasi *use case* tambahan kesebuah *use case* dimana *use case* yang ditambahkan dapat berdiri sendiri walaupun tanpa *use case* tambahan itu; mirip dengan prinsip *inheritence* pada pemrograman berorientasi objek; biasanya *use case* tambahan memiliki nama depan yang sama dengan *use case* yang di tambahkan, misal:  Arah panah mengarah *use case* yang ditambahkan; biasanya *use case* yang menjadi *extend*-nya merupakan jenis yang sama dengan *use case* yang menjadi induknya.

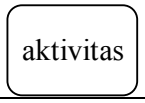
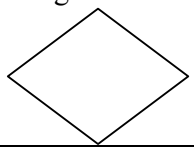
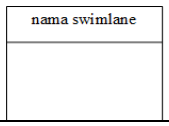
Tabel 2.2 Lanjutan

Simbol	Deskripsi
Generalisasi / *generalitazion* 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah *use case* dimana fungsi yang satu adalah fungsi yang lebih umum dari yang lainnya, misalnya:  Arah panah mengarah pada *use case* yang menjadi generalisasinya(umum).
Menggunakan/ *include / uses*  	Relasi *use case* tambahan kesebuah *use case* dimana *use case* yang ditambahkan memerlukan *use case* ini untuk menjalankan fungsinya atau sebagai syarat dijalankan *use case* ini. Ada dua sudut pandang yang cukup besar mengenai *include* di-*use case*: -*include* berarti *use case* yang ditambahkan akan selalu dipanggil saat *use case* tambahan dijalankan, misal pada kasus berikut:  -*include* berarti *use case* yang tambahan akan selalu melakukan pengecekan apakah *use case* yang ditambahkan telah dijalankan sebelum *use case* tambahan dijalankan, misal pada kasus berikut:  Kedua interpretasi diatas dapat dianut salah satu atau keduanyatergantung pada pertimbangan dan interpretasi yang dibutuhkan.

2.2.11.2 *Activity* Diagram

Diagram aktivitas atau *Activity* diagram menggambarkan alur kerja (*work flow*) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem. Simbol-simbol yang biasa digunakan dalam membuat *use case* diagram dapat dilihat pada Tabel 2.3.

Tabel 2.3 Simbol *Activity* Diagram
(Sumber : Rossa A.S dan M. Salahuddin, 2015)

Simbol	Deskripsi
Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal
Aktivitas 	Aktivitas yang dilakukan sistem, Aktivitas biasanya diawali dengan kata kerja
Percabangan / *decision* 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu
Penggabungan / *join* 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu
Status akhir 	Status akhir yang dilakukan sistem, Sebuah diagram aktivitas memiliki sebuah status akhir
Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi
Fork 	*Fork* digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel
Join 	*Join* digunakan untuk menunjukkan kegiatan yang digabungkan

2.2.12 Xampp

Xampp adalah perangkat yang menggabungkan tiga aplikasi kedalam satu paket, yaitu Apache, MySQL, dan PHPMy Admin. Dengan Xampp pekerjaan akan sangat dimudahkan karena dapat menginstalasi dan mengkonfigurasi ketiga aplikasi tersebut dengan sekaligus dan otomatis. Aplikasi utama dalam paket Xampp yaitu terdiri atas Web server Apache, MySQL, PHP, dan PHPMyAdmin. Apache adalah sebuah Web server *open source*, jadi pengguna dapat menggunakannya secara gratis bahkan bisa mengedit kode programnya. Fungsi utama dari Apache yaitu menghasilkan halaman Web yang benar sesuai dengan yang dibuat oleh seorang Web programmer menggunakan kode PHP (Affandi).

2.2.13 Serveo

Serveo adalah sebuah SSH *server* yang hanya berfungsi sebagai *remote port forwading*. Ketika *user* melakukan koneksi pada Serveo, *user* mendapatkan *public* URL yang dapat diakses dari internet dan menampilkan *server* lokal. Kelebihan dari Serveo adalah tidak dibutuhkannya aplikasi yang harus didownload, yang penting perangkat yang digunakan sudah *support* SSH (Imanudin).

BAB III
PERANCANGAN PERANGKAT LUNAK

3.1. Spesifikasi Kebutuhan *Hardware* dan *Software*

Pada subbab ini dijelaskan komponen-komponen *hardware* dan *software* yang diperlukan dalam pembuatan aplikasi *Monitoring Interfaces Ethernet* pada Cisco Catalyst 3750 untuk Menjamin Keamanan Penggunaan Jaringan Komputer.

3.1.1. Spesifikasi Perangkat Keras (*Hardware*)

Perangkat keras yang digunakan dalam pembuatan sistem ini adalah sebagai berikut:

1. Cisco Catalyst 3750
2. *Processor* AMD A6-6310
3. RAM 2 GB
4. Kabel LAN
5. Kabel Console
6. *Mouse*

3.1.2. Spesifikasi Perangkat Lunak (*Software*)

Perangkat lunak yang digunakan dalam pembuatan sistem ini adalah sebagai berikut:

1. Windows 10 Pro 64 Bit
2. Xampp v3.2.2
3. Browser
4. Sublime Text 3
5. Tera Term
6. *Command Prompt*

3.2. Perancangan Sistem *Monitoring Interfaces Ethernet*

Perancangan sistem adalah merancang komponen-komponen yang akan dibangun untuk membentuk sistem yang akan dibuat nantinya. Perancangan

sistem ini juga akan membantu dalam pembuatan program agar tidak melenceng dari perancangan yang telah dibuat. Perancangan sistem dalam pembuatan aplikasi "*Monitoring Interfaces Ethernet* pada Cisco Catalyst 3750 untuk Menjamin Keamanan Penggunaan Jaringan Komputer" ini melibatkan beberapa tahapan *Unified Modeling Language* (UML) yang terdiri dari *Use Case* Diagram dan *Activity* Diagram serta perancangan antarmuka. Dengan adanya perancangan sistem ini dapat mempermudah proses pembuatan aplikasi, karena jalannya aplikasi yang akan dibuat sudah dirancang yang kemudian dapat langsung diimplementasikan kedalam program. Sehingga jika mengalami kendala dalam proses implementasinya bisa dilihat lagi pada perancangan sistem.

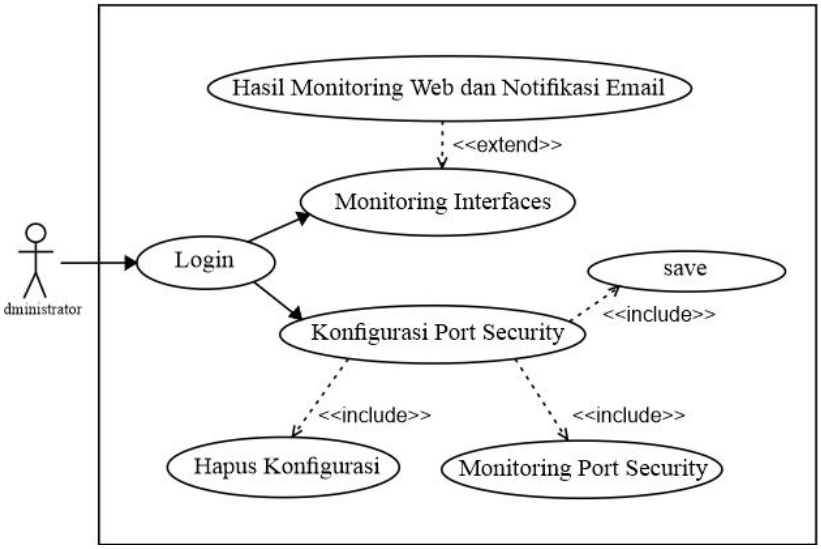
3.3 Perancangan *Telecommunications Network* (Telnet) pada Perangkat Cisco Catalyst 3750

Dalam pembuatan aplikasi "*Monitoring Interfaces Ethernet* pada Cisco Catalyst 3750 untuk Menjamin Keamanan Penggunaan Jaringan Komputer" langkah pertama yang dilakukan adalah membuat atau merancang Telnet pada perangkat Cisco Catalyst 3750 agar bisa di-*monitoring* dari jarak jauh. Saat konfigurasi atau pembuatan Telnet yang perlu diatur adalah *ip address, username, password* dan jumlah *user* (*vty*). *Ip address* digunakan untuk memberi alamat *ip* pada perangkat Cisco Catalyst 3750, yang nantinya digunakan untuk pemanggilan pada saat menggunakan perangkat tersebut. *Username* dan *password* digunakan untuk *login* saat akan menggunakan perangkat Cisco Catalyst 3750. Sedangkan jumlah *user* (*vty*) digunakan untuk membatasi *user* yang dapat terhubung dengan perangkat Cisco Catalyst 3750.

3.4 *Use Case* Diagram pada Sistem *Monitoring Interfaces Ethernet*

Use Case Diagram digunakan untuk menggambarkan secara ringkas siapa yang dapat menggunakan sistem dan apa saja yang dapat dilakukan. Dalam pembuatan aplikasi "*Monitoring Interfaces Ethernet* pada Cisco Catalyst 3750 untuk Menjamin Keamanan Penggunaan Jaringan Komputer" ini hanya menggunakan satu aktor (pengguna) yang dapat berinteraksi dengan sistem yang

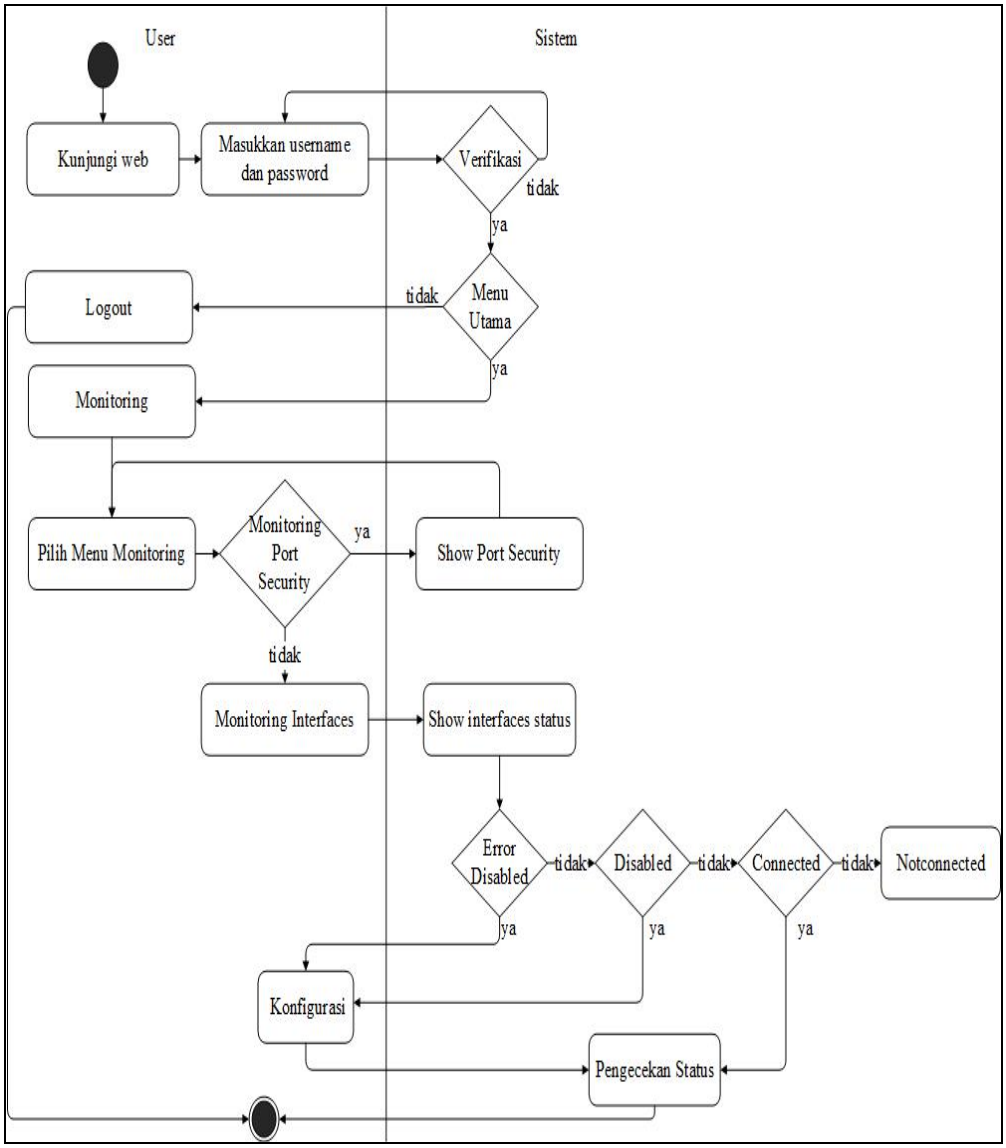
akan dibuat. Aktor dalam *use case* ini merupakan admin yang bertindak sebagai pengelola jaringan. Gambar dari *use case* diagram dapat dilihat pada Gambar 3.1.



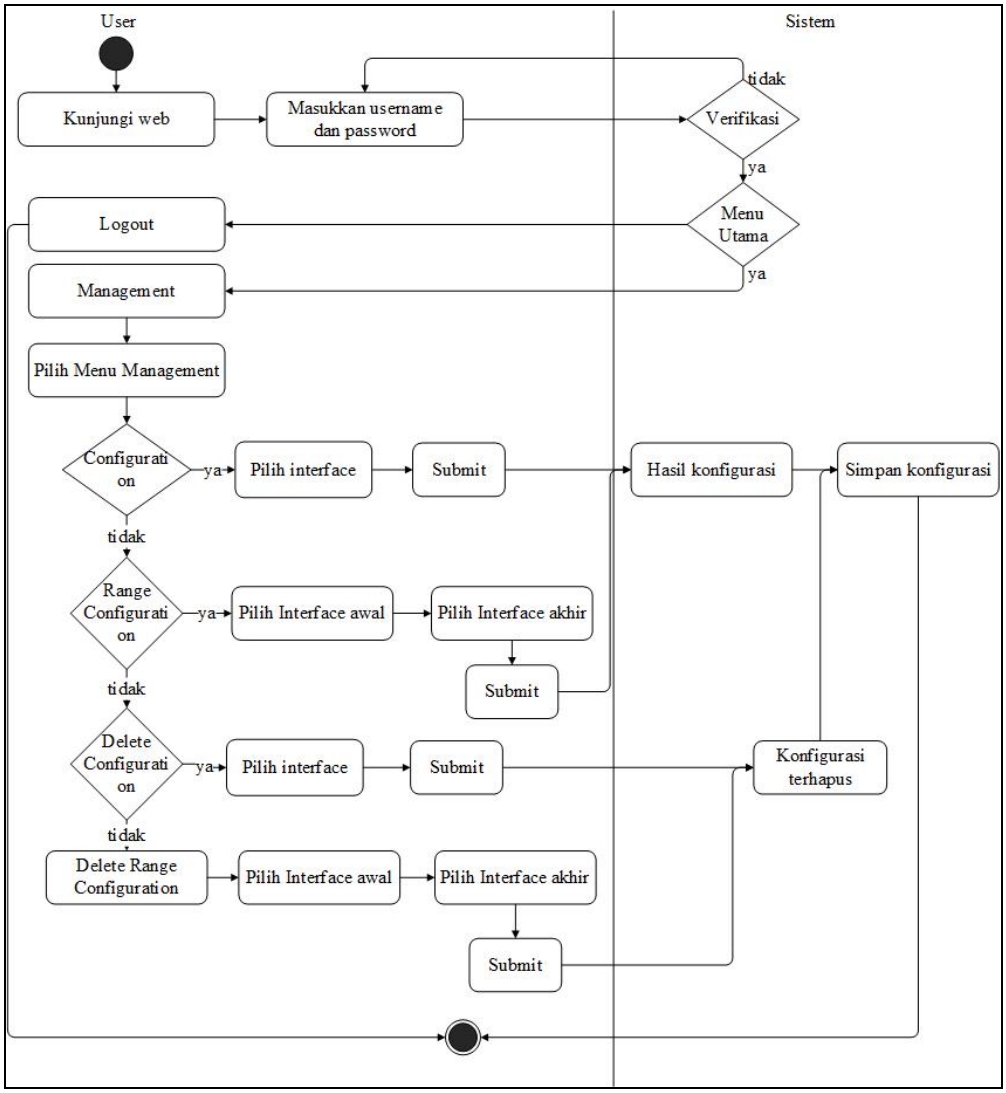
Gambar 3.1 *Use Case* Diagram Sistem *Monitoring Interfaces Ethernet*

3.5 *Activity* Diagram pada Sistem *Monitoring Interfaces Ethernet*

Activity Diagram menggambarkan rangkaian aliran dari aktivitas apa saja yang dapat dilakukan oleh pengguna sistem. Dalam sistem ini terdapat dua *activity* diagram antara lain *activity* diagram *monitoring* yang terlihat pada Gamabr 3.2 digunakan untuk memantau atau mengawasi jaringan yang ada, *activity* diagram *management* yang terlihat pada Gambar 3.3 digunakan untuk mengelola jaringan yang ada.



Gambar 3.2 *Activity* Diagram *Monitoring* pada Sistem *Monitoring Interfaces Ethernet*



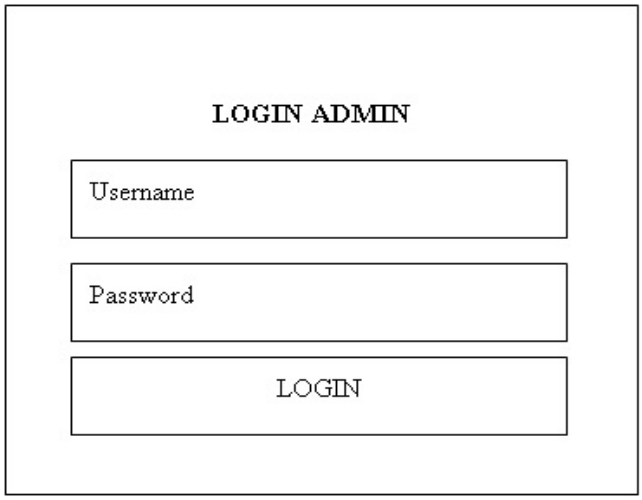
Gambar 3.3 *Activity* Diagram *Management* pada Sistem *Monitoring Interfaces Ethernet*

3.6 Perancangan Antarmuka pada Sistem *Monitoring Interfaces Ethernet*

Perancangan antarmuka ini bertujuan untuk memberikan gambaran tentang aplikasi yang akan dibangun, sehingga dapat mempermudah dalam pembuatan aplikasi "*Monitoring Interfaces Ethernet* pada Cisco Catalyst 3750 untuk Menjamin Keamanan Penggunaan Jaringan Komputer". Berikut akan dijelaskan masing-masing tampilan perancangan antarmuka secara keseluruhan, yaitu sebagai berikut:

3.6.1. Perancangan Tampilan *Login* pada Sistem *Monitoring Interfaces Ethernet*

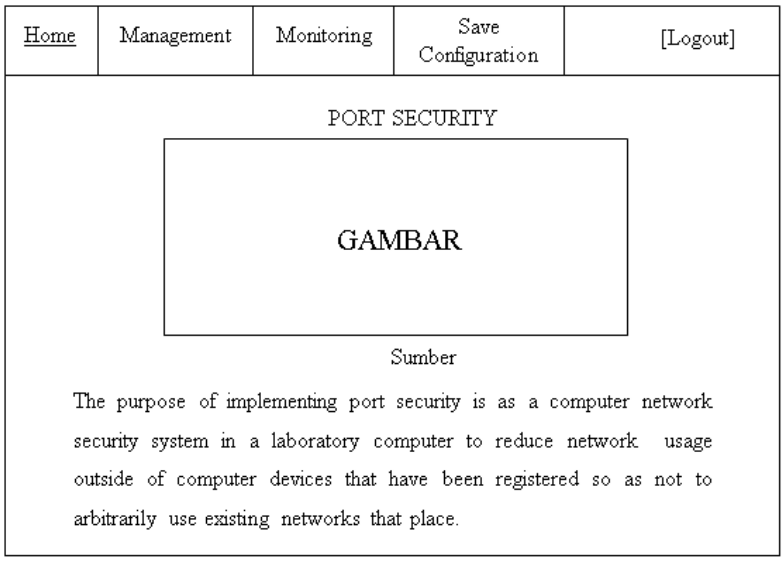
Tampilan *login* adalah tampilan yang muncul pertama kali saat *administrator* atau user mengunjungi Web *Monitoring Interfaces Ethernet*. Pada tampilan menu *login* di Gambar 3.4 menampilkan *username* dan *password* yang harus diisi dengan benar oleh *user* agar bisa mengakses Web *Monitoring Interfaces Ethernet*. Jika *user* memasukkan *username* dan *password* yang salah maka akan tetap berada di menu *login* sampai *username* dan *password* yang diminta sesuai atau benar. Pada perancangan *login Monitoring Interfaces Ethernet* ini *username* dan *password* sudah diatur secara statis pada program.



Gambar 3.4 Perancangan Tampilan *Login* pada Sistem *Monitoring Interfaces Ethernet*

3.6.2. Perancangan Tampilan *Home* pada Sistem *Monitoring Interfaces Ethernet*

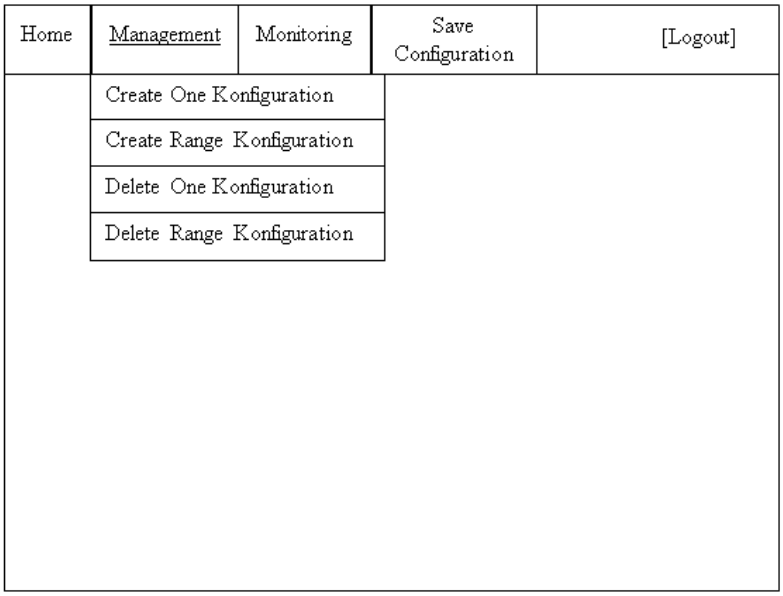
Perancangan tampilan *home* merupakan tampilan yang muncul setelah *user* berhasil melakukan proses *login*. Pada tampilan *home* di Gambar 3.5 terdapat gambar dan informasi tentang tujuan implementasi *port security* pada perangkat Cisco Catalyst 3750.



Gambar 3.5 Perancangan Tampilan *Home* pada Sistem *Monitoring Interfaces Ethernet*

3.6.3. Perancangan Tampilan Menu *Management* pada Sistem *Monitoring Interfaces Ethernet*

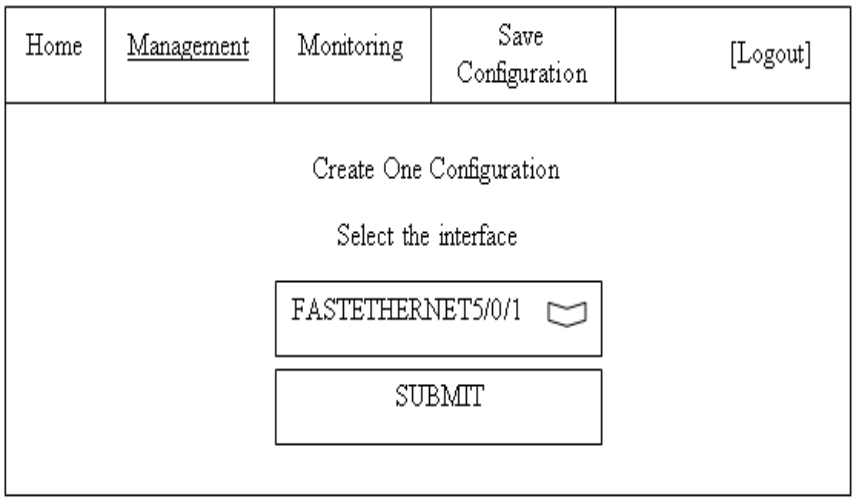
Pada perancangan tampilan menu *management* pada sistem *monitoring interfaces ethernet* yang dapat dilihat pada Gambar 3.6 terdapat empat sub menu di dalamnya.



Gambar 3.6 Perancangan Tampilan Menu *Management* pada Sistem *Monitoring Interfaces Ethernet*

3.6.3.1. Perancangan Tampilan Submenu *Create One Configuration* pada Sistem *Monitoring Interfaces Ethernet*

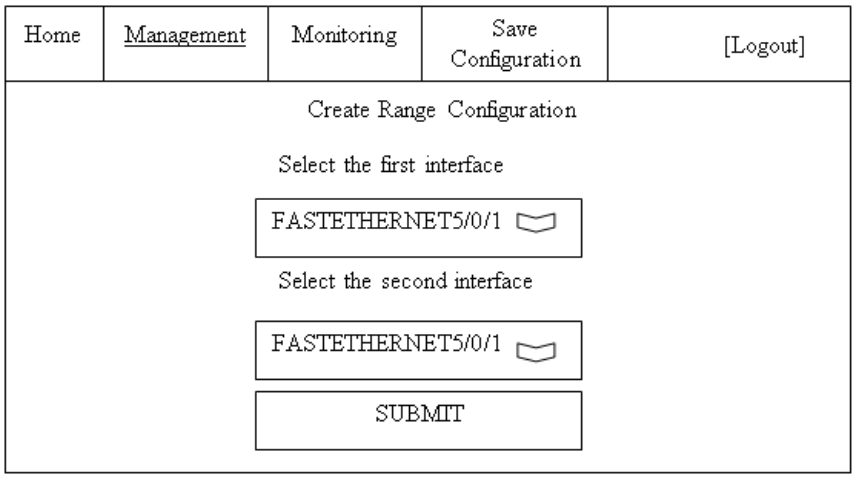
Perancangan tampilan submenu *create one configuration* merupakan tampilan yang dapat digunakan admin untuk mengkonfigurasi *port security* pada *interface* yang ingin dikonfigurasi. Pada tampilan konfigurasi di Gambar 3.7 admin cukup memilih *interface* yang ingin dikonfigurasi pada *option* dan langsung menekan tombol *submit*, maka *interface* yang dipilih sudah terkonfigurasi *port security*.



Gambar 3.7 Perancangan Tampilan Submenu *Create One Configuration* pada Sistem *Monitoring Interfaces Ethernet*

3.6.3.2. Perancangan Tampilan Submenu *Create Range Configuration* pada Sistem *Monitoring Interfaces Ethernet*

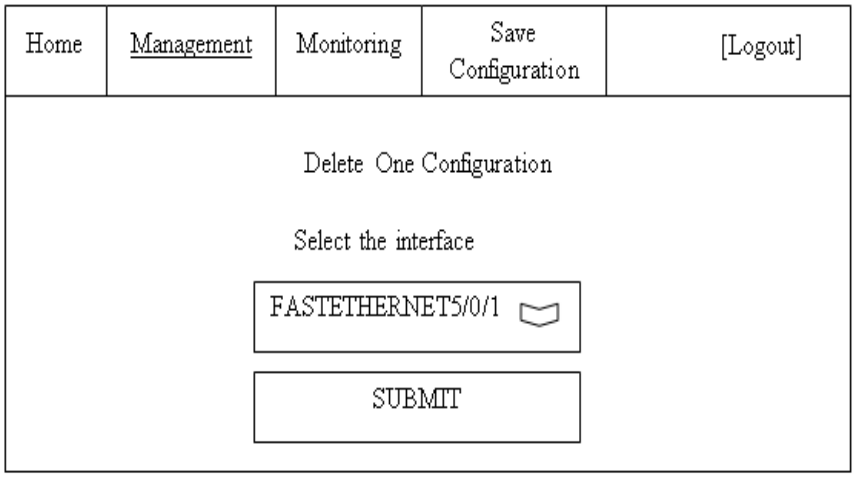
Perancangan tampilan submenu *create range configuration* merupakan tampilan yang dapat digunakan admin untuk mengkonfigurasi *port security* pada *interface* yang ingin dikonfigurasi secara kelompok. Pada tampilan konfigurasi di Gambar 3.8 admin cukup memilih *interface* awal dan *interface* akhir yang ingin dikonfigurasi pada *option* dan langsung menekan tombol *submit*, maka *interface* yang dipilih sudah terkonfigurasi *port security*.



Gambar 3.8 Perancangan Tampilan Submenu *Create Range Configuration* pada Sistem *Monitoring Interfaces Ethernet*

3.6.3.3. Perancangan Tampilan Submenu *Delete One Configuration* pada Sistem *Monitoring Interfaces Ethernet*

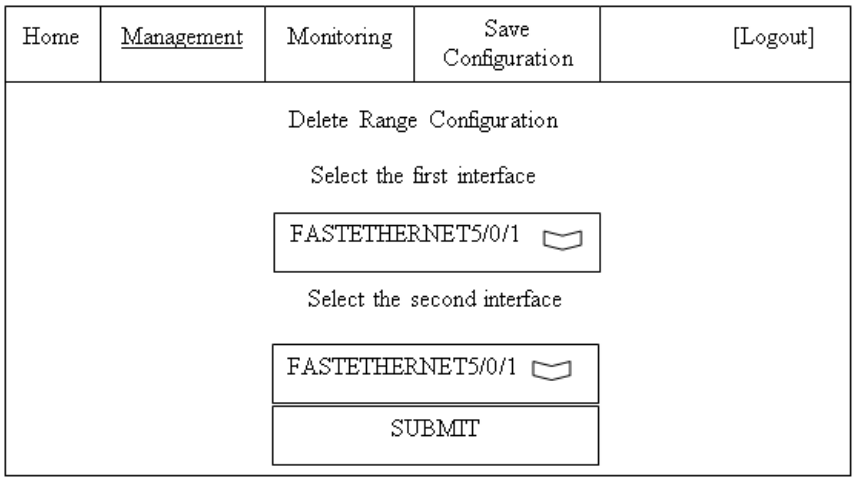
Perancangan tampilan submenu *delete one configuration* merupakan tampilan yang dapat digunakan admin untuk menghapus konfigurasi *port security* pada *interface* yang telah dikonfigurasi. Pada tampilan hapus konfigurasi di Gambar 3.9 admin cukup memilih *interface* yang ingin dihapus konfigurasinya pada *option* dan langsung menekan tombol *submit*, maka *interface* yang dipilih sudah kembali normal yaitu tanpa konfigurasi *port security*.



Gambar 3.9 Perancangan Tampilan Submenu *Delete One Configuration* pada Sistem *Monitoring Interfaces Ethernet*

3.6.3.4. Perancangan Tampilan Submenu *Delete Range Configuration* pada Sistem *Monitoring Interfaces Ethernet*

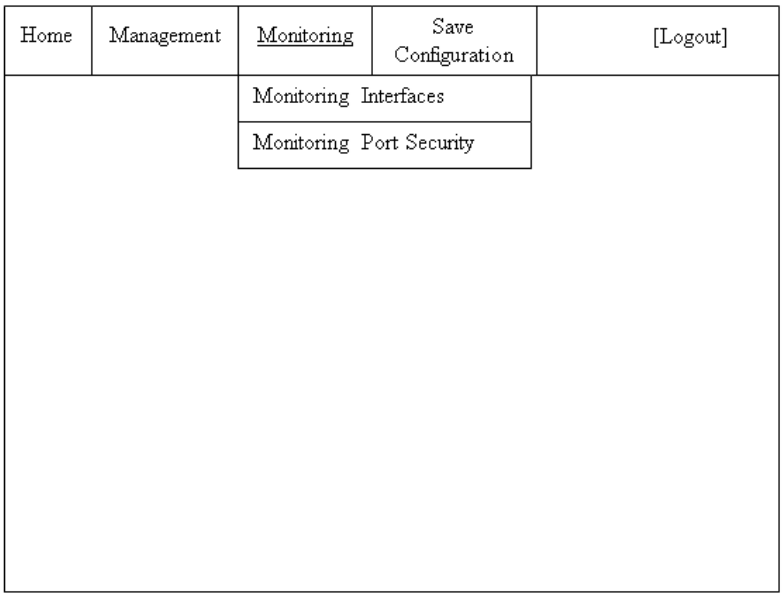
Perancangan tampilan submenu *delete range configuration* merupakan tampilan yang dapat digunakan admin untuk menghapus konfigurasi *port security* pada *interface* yang telah dikonfigurasi secara kelompok tetapi harus secara berurutan. Pada tampilan *delete range configuration* di Gambar 3.10 admin cukup memilih *interface* awal dan *interface* akhir yang ingin dihapus konfigurasinya pada *option* dan langsung menekan tombol *submit*, maka *interface* yang dipilih sudah kembali normal yaitu tanpa konfigurasi *port security*.



Gambar 3.10 Perancangan Tampilan Submenu *Delete Range Configuration* pada Sistem *Monitoring Interfaces Ethernet*

3.6.4. Perancangan Tampilan Menu *Monitoring* pada Sistem *Monitoring Interfaces Ethernet*

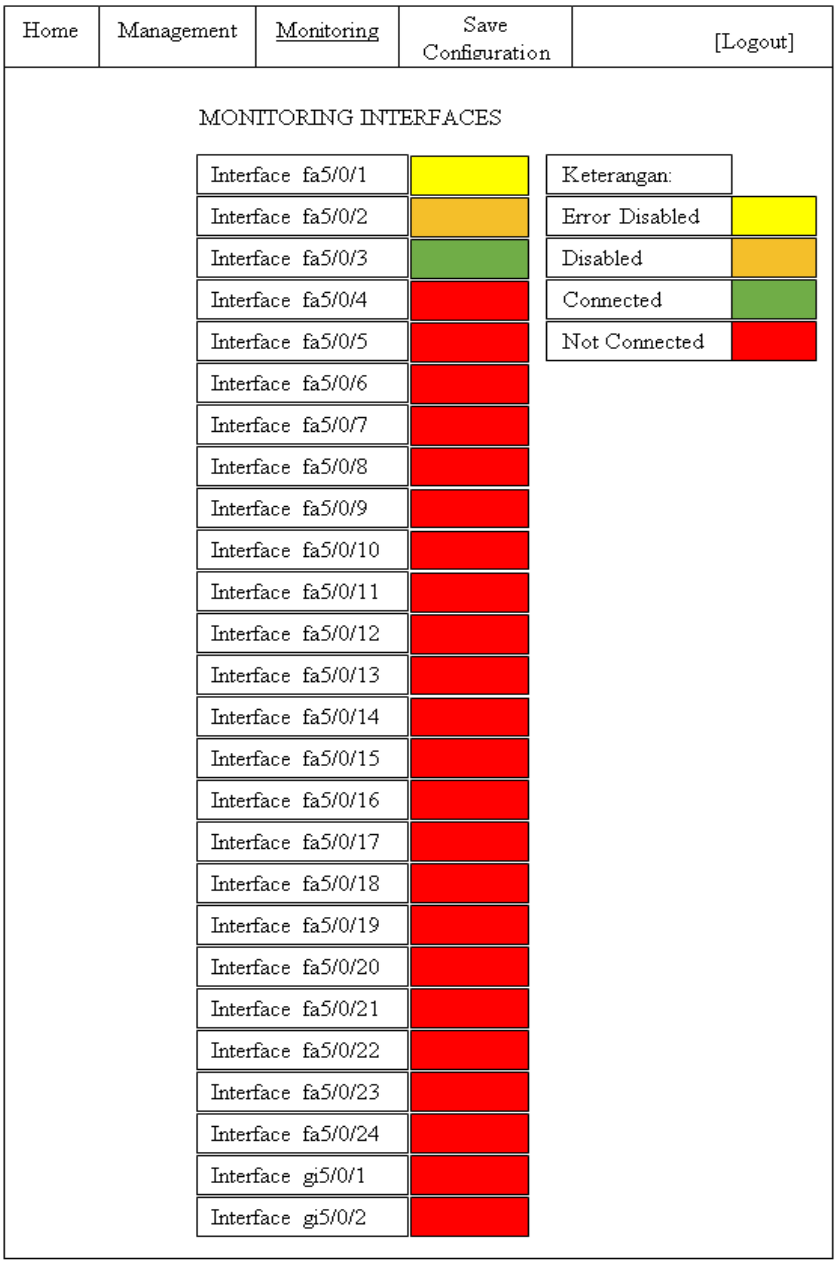
Pada perancangan tampilan menu *monitoring* pada sistem *monitoring interfaces ethernet* yang dapat dilihat pada Gambar 3.11 terdapat dua submenu di dalamnya.



Gambar 3.11 Perancangan Tampilan Menu *Monitoring* pada Sistem *Monitoring Interfaces Ethernet*

3.6.4.1. Perancangan Tampilan Submenu *Monitoring Interfaces* pada Sistem *Monitoring Interfaces Ethernet*

Perancangan tampilan submenu *monitoring interfaces* ini merupakan tampilan dimana admin dapat memantau aktifitas yang ada di *interfaces* Cisco Catalyst 3750. Pada tampilan yang terlihat di Gambar 3.12 terdapat nama *interface* dan status *interface*. Aktifitas atau perubahan status *interface* dalam *monitoring* ini adalah *Error-disabled* dengan tombol warna *orange, disabled* dengan tombol warna kuning, *connected* dengan tombol warna hijau, dan *not-connected* dengan tombol warna merah. Pada submenu tampilan ini admin dapat mengetahui *interfaces* mana saja yang digunakan dan jika terdapat *error* berupa *error-disabled* atau *disabled* maka dapat langsung mengatasinya atau mengaktifkan dengan menekan tombol berupa warna yang terdapat di sebelah nama *interfaces*-nya.

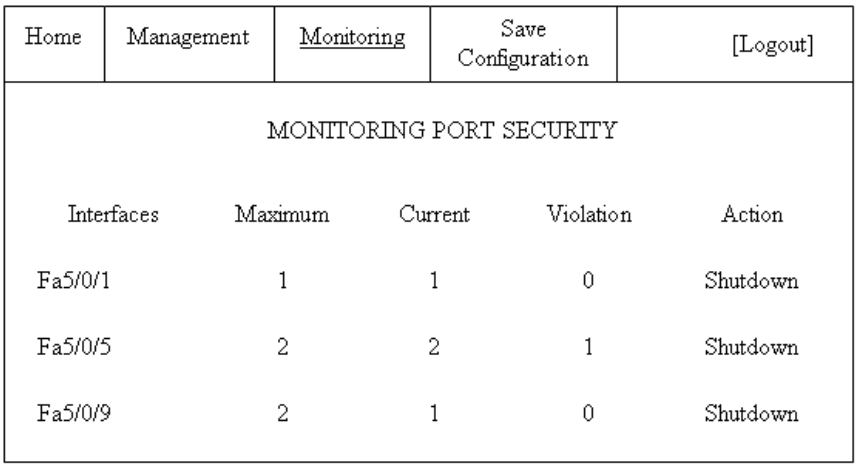


Gambar 3.12 Perancangan Tampilan Submenu *Monitoring Interfaces* pada Sistem *Monitoring Interfaces Ethernet*

3.6.4.2. Perancangan Tampilan Submenu *Monitoring Port Security* pada Sistem *Monitoring Interfaces Ethernet*

Perancangan tampilan submenu *monitoring port security* ini merupakan tampilan dimana admin dapat memantau *interfaces* Cisco Catalyst 3750 yang sudah terkonfigurasi *port security*. Pada tampilan yang terlihat di Gambar 3.13

terdapat nama *interface*, jumlah *maximum* perangkat yang terhubung, *current* yang berisi jumlah perangkat yang sudah terhubung dengan Cisco Catalyst 3750, *violation* yang berisi jika perangkat yang terhubung dengan Cisco lebih dari *maximum* maka akan tertulis angka 1 yang berarti port tersebut melebihi batas *maximum*, dan *action* yang berisi aksi dari *interface* tersebut yaitu *shutdown*.



Gambar 3.13 Perancangan Tampilan Submenu *Monitoring Port Security* pada Sistem *Monitoring Interfaces Ethernet*

BAB IV
HASIL PENELITIAN DAN PEMBAHASAN

4.1. Tampilan Pembuatan Telnet menggunakan Tera-Term

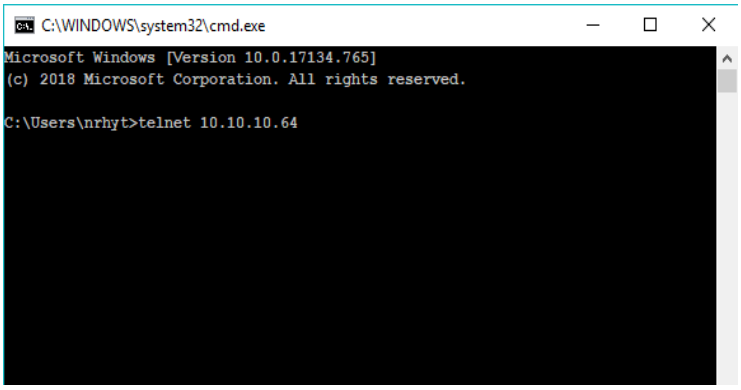
Tampilan ini merupakan tampilan saat membuat atau mengkonfigurasi perangkat Cisco Catalyst 3750 dengan Telnet agar bisa diakses dari jarak jauh. perintah yang digunakan untuk mengkonfigrasi Telnet yang pertama adalah mengetikkan "*enable*" yang digunakan untuk masuk ke perangkat *switch* atau agar berganti tandanya menjadi "#". Kemudian mengetikkan perintah "*configure terminal*" untuk mengkonfigurasi, "*interface* vlan 1" digunakan untuk masuk ke *interface* Vlan 1, "*ip address* 10.10.10.64 255.255.255.0" yang digunakan untuk memberi alamat ip pada *interface* Vlan 1, "*username* stta *password* 12345" yang digunakan untuk memberi nama dan *password* saat akan mengakses Telnet, "line vty 0 1" untuk membatasi akses ke Telnet. Untuk lebih jelasnya langkah-langkah pembuatan Telnet pada Tera Term dapat dilihat pada Gambar 4.1.

```
Switch#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Switch(config)#interface vlan 1
Switch(config-if)#ip address 10.10.10.64 255.255.255.0
Switch(config-if)#no shutdown
Switch(config-if)#exit
Switch(config)#username stta password 12345
Switch(config)#line vty 0 1
Switch(config-line)#login local
Switch(config-line)#
```

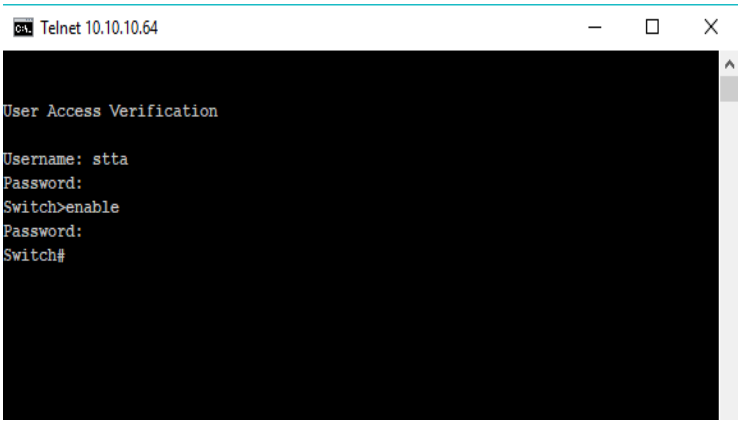
Gambar 4.1 Tampilan Pembuatan Telnet pada Tera Term

4.2. Tampilan Pemanggilan Telnet pada *Command Prompt*

Perintah ini digunakan untuk mengaktifkan Telnet pada *command prompt* dengan mengetikkan "telnet 10.10.10.64". 10.10.10.64 merupakan alamat ip dari perangkat Cisco Catalyst 3750. Tampilan pemanggilan atau pengaktifan Telnet dapat dilihat pada Gambar 4.2. Sedangkan tampilan setelah berhasil masuk Telnet dapat dilihat pada Gambar 4.3.



Gambar 4.2 Tampilan Pemanggilan Telnet pada *Command Prompt*



Gambar 4.3 Tampilan Telnet pada *Command Prompt*

4.3. Tampilan *Source Code Connection* Aplikasi dengan Perangkat Cisco Catalyst 3750

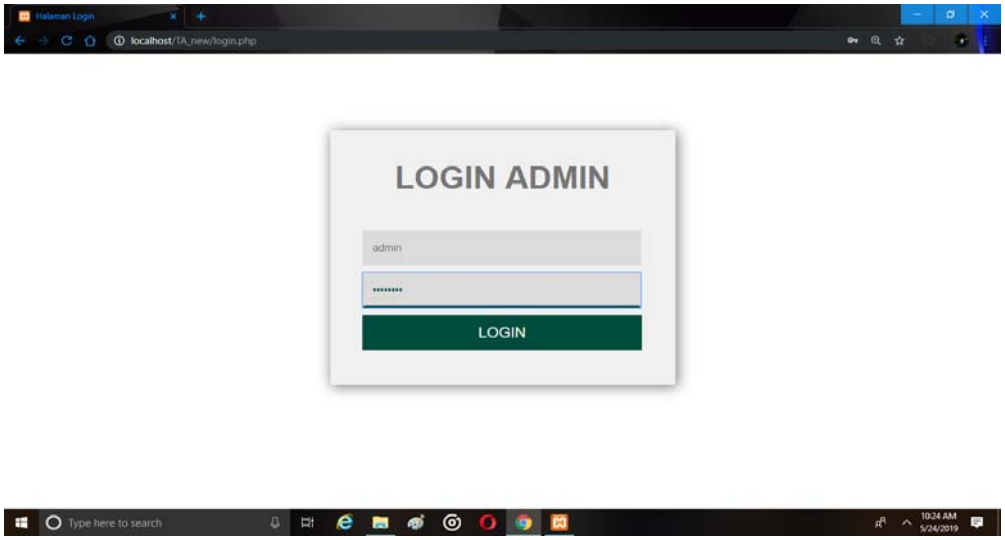
Source code connection ini yang nantinya akan digunakan untuk mengkoneksikan antara aplikasi yang dibuat dengan perangkat Cisco Catalyst 3750 yang digunakan. Pada dasarnya *source code* ini sama dengan perintah Telnet pada *command line* yang ada di perangkat Cisco Catalyst 3750. Tampilan *source code connection* antara aplikasi dengan perangkat Cisco Catalyst 3750 dapat dilihat pada Gambar 4.4 mulai dari baris ke 29 sampai baris ke 49.

```
29      public function connect()
30      {
31
32          $this->_connection = fsockopen($this->_hostname, 23, $errno, $errstr, $this->_timeout);
33          if ($this->_connection === false) {
34              die("Error: Connection Failed for $this->_hostname\n");
35          } // if
36          stream_set_timeout($this->_connection, $this->_timeout);
37          $this->_readTo(':');
38          if (substr($this->_data, -9) == 'Username:') {
39              $this->_send($this->_username);
40              $this->_readTo(':');
41          } // if
42          $this->_send($this->_password);
43          $this->_prompt = '>';
44          $this->_readTo($this->_prompt);
45          if (strpos($this->_data, $this->_prompt) === false) {
46              fclose($this->_connection);
47              die("Error: Authentication Failed for $this->_hostname\n");
48          } // if
49      } // connect
```

Gambar 4.4 Tampilan *Source Code Connection* antara Aplikasi dengan Perangkat Cisco Catalyst 3750

4.4. Tampilan Halaman *Login* pada Sistem *Monitoring Interfaces Ethernet*

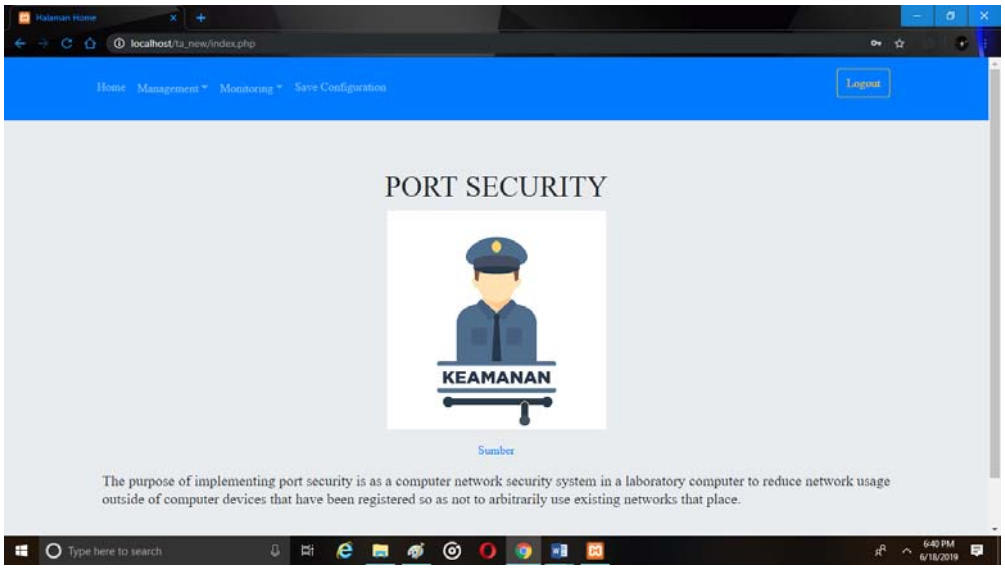
Halaman *login* merupakan halaman pertama yang akan tampil saat *user* atau administrator mengunjungi Web *Monitoring Interfaces Ethernet*. Di dalam halaman *login* terdapat *username* dan *password* yang harus diisi oleh administrator. *Username* dan *password* yang dimasukkan harus sesuai yaitu *username* = "admin" dan *password* = "admin123", jika *username* dan *password* diisi salah atau dikosongkan maka akan muncul *alert* atau peringatan. Setelah administrator memasukkan *username* dan *password* yang sesuai maka langsung menekan tombol *login* agar bisa mengakses menu yang ada di Web *Monitoring Interfaces Ethernet*. Tampilan halaman *login* pada sistem *monitoring interfaces ethernet* dapat dilihat pada Gambar 4.5.



Gambar 4.5 Tampilan *Login* pada Sistem *Monitoring Interfaces Ethernet*

4.5. Tampilan Menu *Home* pada Sistem *Monitoring Interfaces Ethernet*

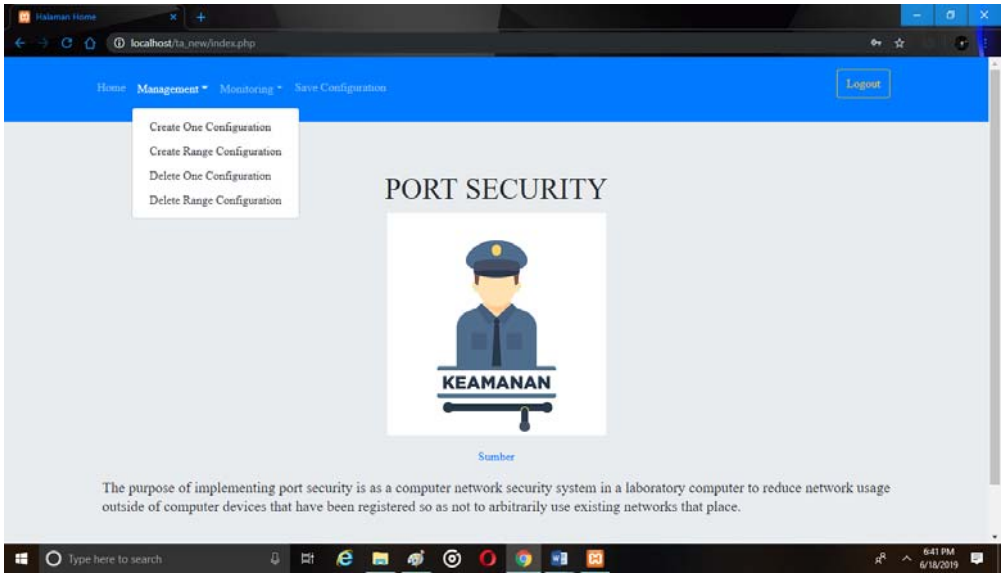
Menu *home* merupakan halaman awal saat administrator berhasil *login*. Di dalam menu *home* di Gambar 4.5 terdapat gambar dan informasi tentang tujuan dari implementasi *port security* pada perangkat Cisco Catalyst 3750. Tampilan menu *home* pada sistem *monitoring interfaces ethernet* dapat dilihat pada Gambar 4.6.



Gambar 4.6 Tampilan Menu *Home* pada Sistem *Monitoring Interfaces Ethernet*

4.6. Tampilan Menu *Management* pada Sistem *Monitoring Interfaces Ethernet*

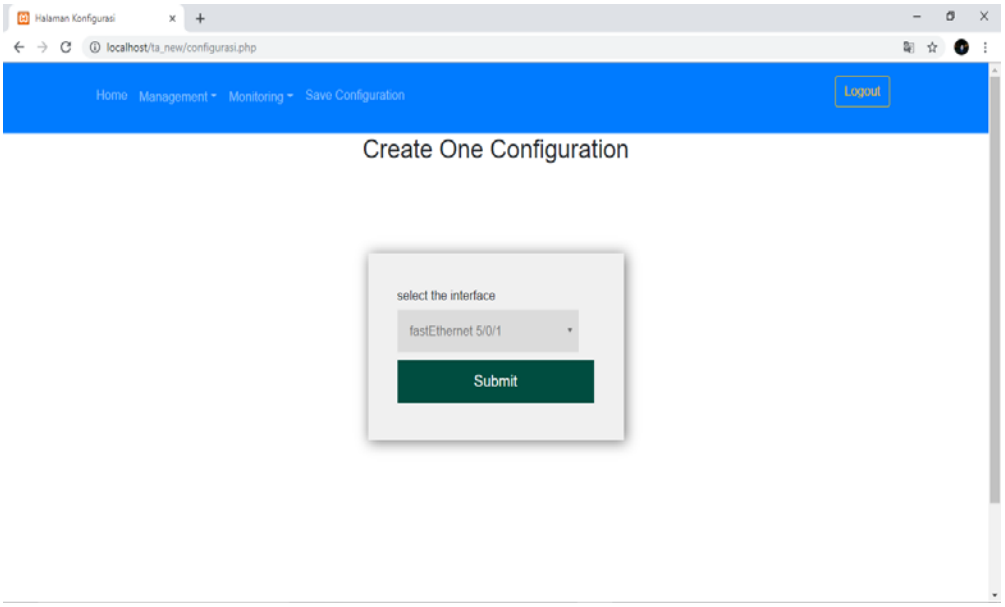
Tampilan menu *management* adalah menu yang dapat dipergunakan untuk pengelolaan terhadap perangkat Cisco Catalyst 3750. Pengelolaan yang dapat dilakukan berupa membuat dan menghapus konfigurasi. Di dalam menu *management* pada sistem *monitoring interfaces ethernet* terbagi atas 4 submenu yang dapat dilihat pada Gambar 4.7.



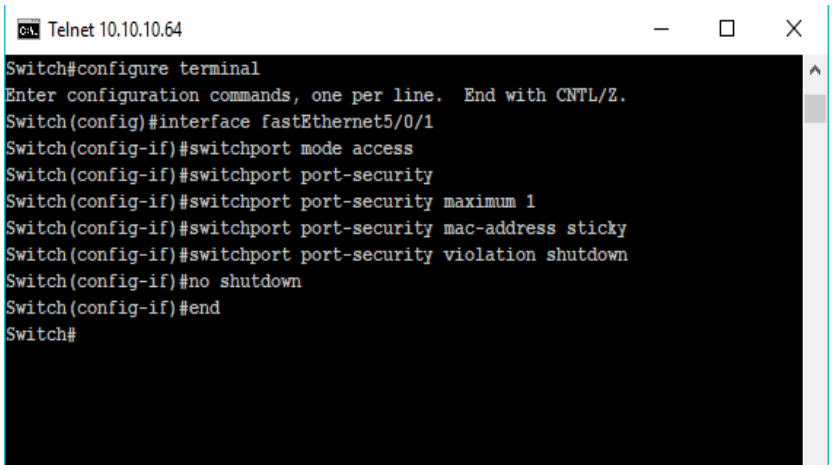
Gambar 4.7 Tampilan Menu *Management* pada Sistem *Monitoring Interfaces Ethernet*

4.6.1. Tampilan Submenu *Create One Configuration* pada Sistem *Monitoring Interfaces Ethernet*

Submenu *create one configuration* merupakan salah satu dari menu yang ada di menu *management*. Di dalam submenu ini administrator dapat mengkonfigurasi *interfaces* dengan sangat mudah, yaitu hanya dengan memilih *interface* yang ingin dikonfigurasi pada *option* atau pilihan dan setelah itu tinggal menekan tombol *submit*. Setelah proses selesai maka *interface* yang dipilih tersebut sudah terkonfigurasi *port security*. Tampilan submenu *create one configuration* pada sistem *monitoring interfaces ethernet* dapat dilihat pada Gambar 4.8. Sedangkan pada Gambar 4.9 adalah tampilan *create one configuration* yang ada di-*command line*.



Gambar 4.8 Tampilan Submenu *Create One Configuration* pada Sistem *Monitoring Interfaces Ethernet*

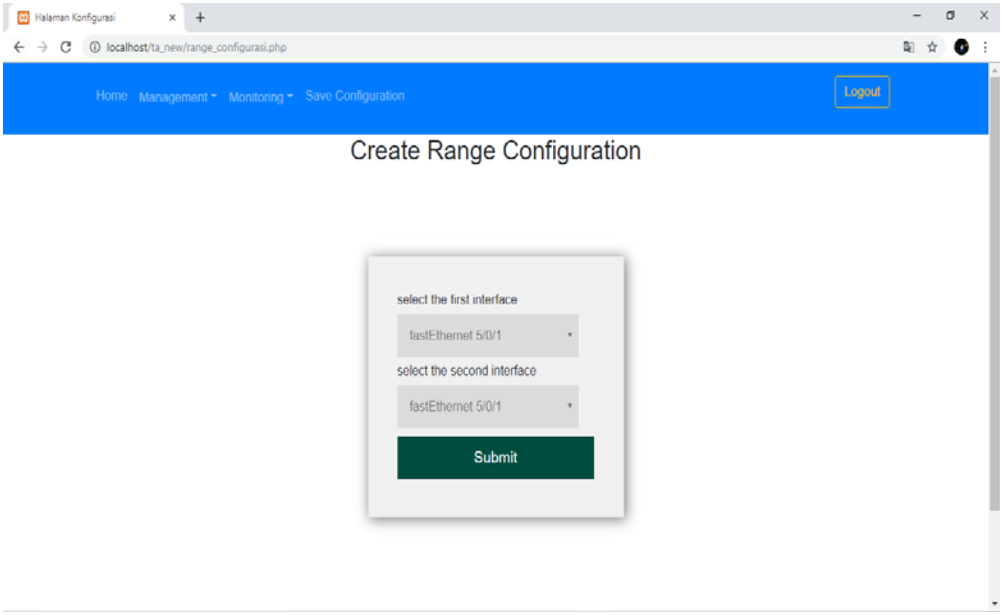


Gambar 4.9 Tampilan *Create One Configuration* pada *Command Line*

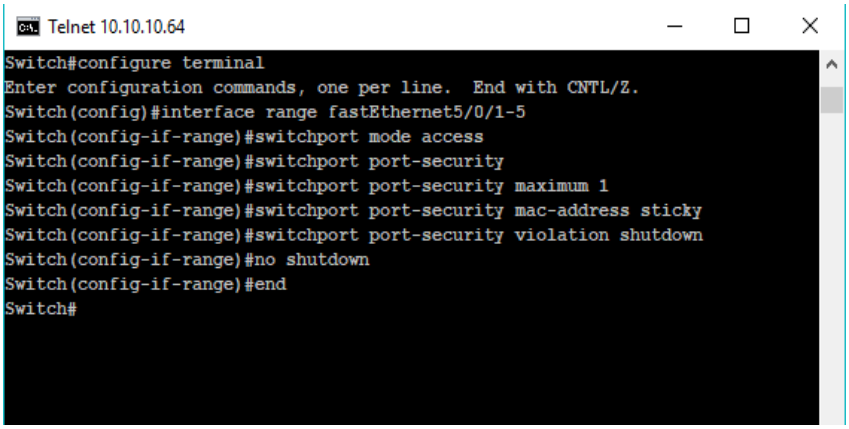
4.6.2. Tampilan Submenu *Create Range Configuration* pada Sistem *Monitoring Interfaces Ethernet*

Submenu *create range configuration* merupakan salah satu dari menu yang ada di menu *management*. Di dalam submenu ini administrator dapat mengkonfigurasi *interfaces* dengan sangat mudah, yaitu hanya dengan memilih *interface* awal dan *interface* akhir yang ingin dikonfigurasi pada *option* atau pilihan dan setelah itu tinggal menekan tombol *submit*. Saat memilih *interface*

terdapat beberapa syarat, yaitu *interface* awal tidak boleh lebih kecil daripada *interface* akhir, *interface* awal dan akhir tidak boleh sama, dan jenis *interface* awal dan akhir harus sama. Setelah proses selesai maka *interface* yang dipilih tersebut sudah terkonfigurasi *port security*. Tampilan submenu *create range configuration* pada sistem *monitoring interfaces ethernet* dapat dilihat pada Gambar 4.10. Sedangkan pada Gambar 4.11 adalah tampilan *create range configuration* yang ada di-*command line*.



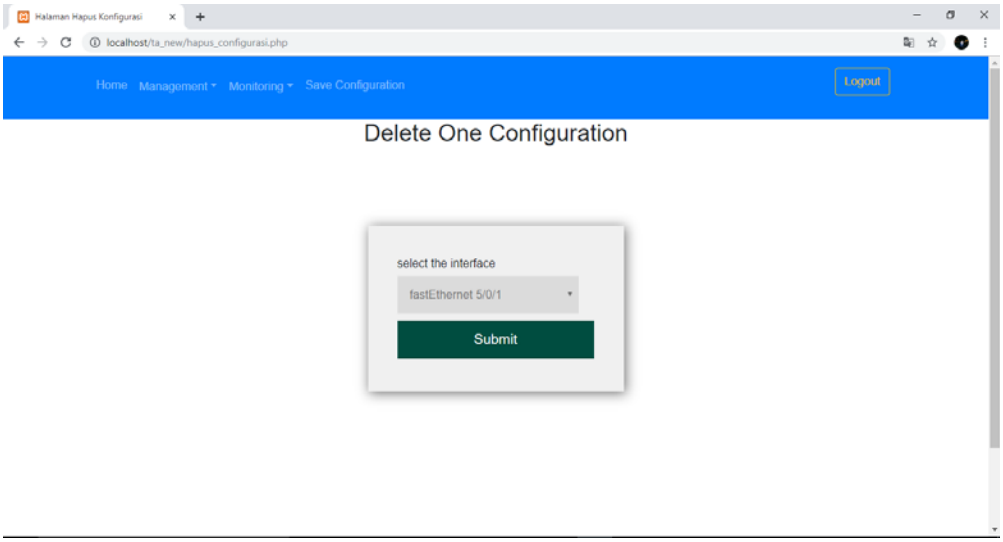
Gambar 4.10 Tampilan Submenu *Create Range Configuration* pada Sistem *Monitoring Interfaces Ethernet*



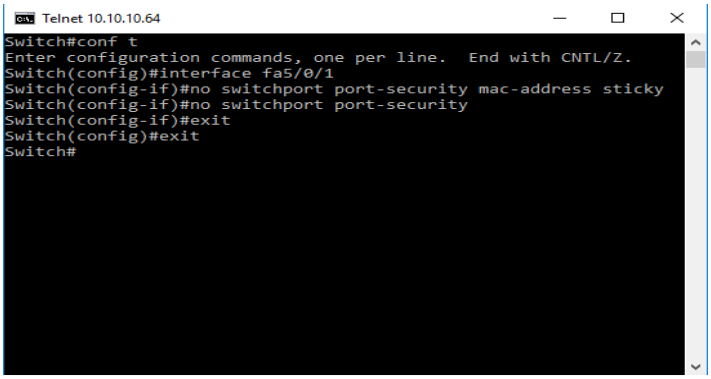
Gambar 4.11 Tampilan *Create Range Configuration* pada *Command Line*

4.6.3. Tampilan Submenu *Delete One Configuration* pada Sistem *Monitoring Interfaces Ethernet*

Submenu *delete one configuration* merupakan salah satu dari menu yang ada di menu *management*. Di dalam submenu ini administrator dapat menghapus konfigurasi *interfaces* dengan sangat mudah, yaitu hanya dengan memilih *interface* yang ingin dihapus konfigurasinya pada *option* atau pilihan dan setelah itu tinggal menekan tombol *submit*. Setelah proses selesai maka *interface* yang dipilih tersebut sudah kembali seperti semula atau tanpa konfigurasi *port security*. Tampilan submenu *delete one configuration* pada sistem *monitoring interfaces ethernet* dapat dilihat pada Gambar 4.12. Sedangkan pada Gambar 4.13 adalah tampilan *delete one configuration* yang ada di-*command line*.



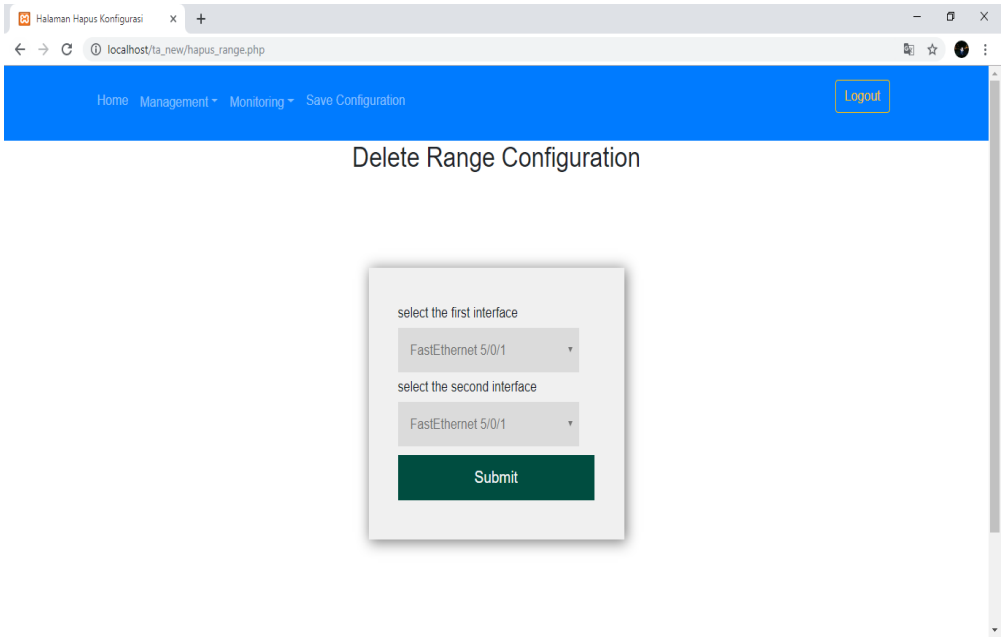
Gambar 4.12 Tampilan Submenu *Delete One Configuration* pada Sistem *Monitoring Interfaces Ethernet*



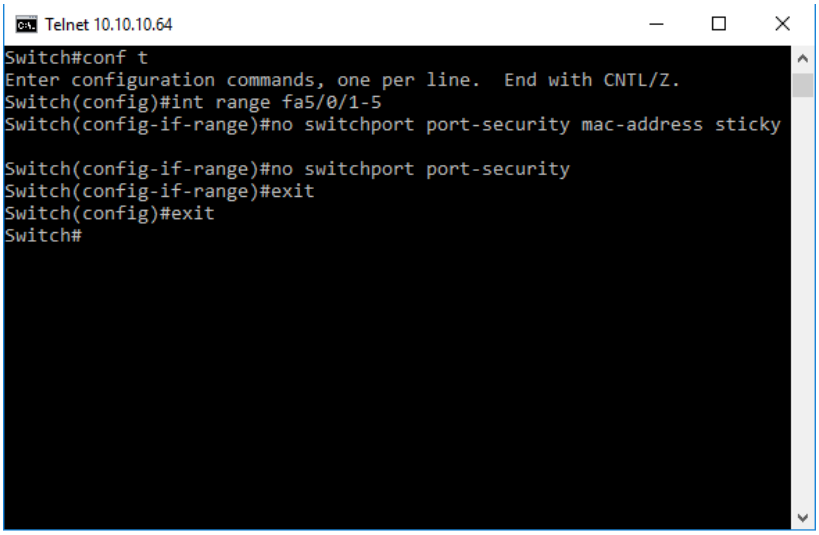
Gambar 4.13 Tampilan *Delete One Configuration* pada *Command Line*

4.6.4. Tampilan Submenu *Delete Range Configuration* pada Sistem *Monitoring Interfaces Ethernet*

Submenu *delete range configuration* merupakan salah satu dari menu yang ada di menu *management*. Di dalam submenu ini administrator dapat menghapus konfigurasi *interfaces* dengan sangat mudah, yaitu hanya dengan memilih *interface* awal dan *interface* akhir yang ingin dihapus konfigurasinya pada *option* atau pilihan dan setelah itu tinggal menekan tombol *submit*. Saat memilih *interface* terdapat beberapa syarat, yaitu *interface* awal tidak boleh lebih kecil daripada *interface* akhir, *interface* awal dan akhir tidak boleh sama, dan jenis *interface* awal dan akhir harus sama. Setelah proses selesai maka *interface* yang dipilih tersebut sudah kembali seperti semula atau tanpa konfigurasi *port security*. Tampilan submenu *delete range configuration* pada sistem *monitoring interfaces ethernet* dapat dilihat pada Gambar 4.14. Sedangkan pada Gambar 4.15 adalah tampilan *delete range configuration* yang ada di-*command line*.



Gambar 4.14 Tampilan Submenu *Delete Range Configuration* pada Sistem *Monitoring Interfaces Ethernet*



Gambar 4.15 Tampilan *Delete Range Configuration* pada *Command Line*

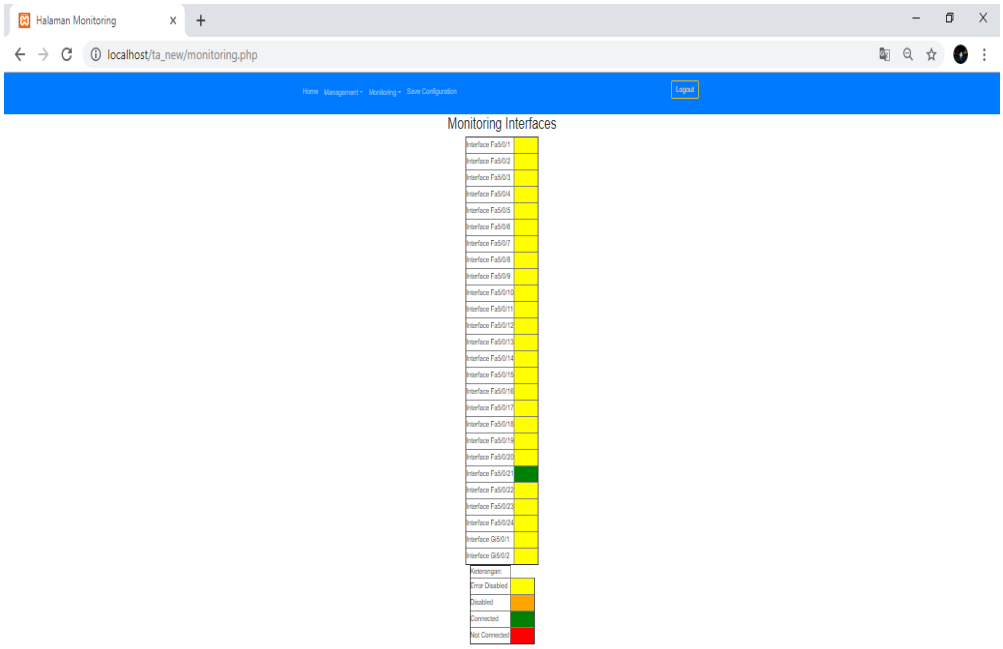
4.7. Tampilan Menu *Monitoring* pada Sistem *Monitoring Interfaces Ethernet*

Tampilan menu *monitoring* adalah menu yang dapat dipergunakan untuk pemantauan terhadap perangkat Cisco Catalyst 3750. Pemantauan yang dapat dilakukan berupa pemantauan *interfaces* dan pemantauan *port security* (*interfaces* yang telah terkonfigurasi). Di dalam menu *monitoring* pada sistem *monitoring interfaces ethernet* terbagi atas 2 submenu yang dapat dilihat pada Gambar 4.16.

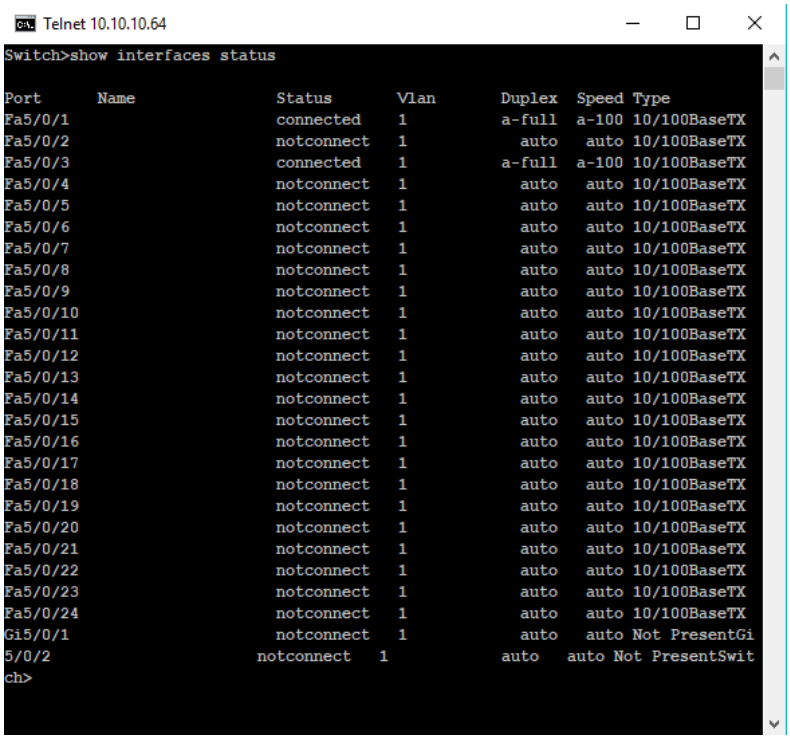
Gambar 4.16 Tampilan Menu *Monitoring* pada Sistem *Monitoring Interfaces Ethernet*

4.7.1. Tampilan Submenu *Monitoring Interfaces* pada Sistem *Monitoring Interfaces Ethernet*

Tampilan submenu *monitoring interfaces* merupakan salah satu menu yang terdapat pada menu *monitoring*. Di dalam submenu ini administrator dapat memantau atau melihat aktivitas yang ada di perangkat Cisco Catalyst 3750. Aktivitas yang dapat dilihat berupa *interface* mana saja yang terhubung (*connected*) dengan perangkat komputer yang disimbolkan dengan tombol warna hijau, *interfaces* mana saja yang tidak terhubung (*not connected*) dengan perangkat komputer yang disimbolkan dengan tombol warna merah, *interfaces* yang *error disabled* yang disimbolkan dengan tombol warna kuning, dan *interfaces* yang *disabled* yang disimbolkan dengan tombol warna *orange*. *Error disabled* merupakan *error* yang *interface*-nya terhubung dengan perangkat komputer yang tidak diijinkan atau tidak terdaftar di perangkat Cisco Catalyst 3750, apabila *interface* tersebut ingin digunakan kembali harus dihubungkan dengan perangkat komputer yang diijinkan dan sebelum bisa dipergunakan harus diaktifkan ulang dengan menekan tombol warna kuning dan *interrface* sudah aktif kembali. Sedangkan *disabled* adalah *error* yang *interface*-nya sengaja dimatikan, apabila ingin mengaktifkan *interface* tersebut hanya dengan menekan tombol warna *orange* dan *interface* sudah aktif kembali. Tampilan *monitoring interfaces* pada sistem *monitoring interfaces ethernet* dapat dilihat pada Gambar 4.17. Sedangkan pada Gambar 4.18 merupakan tampilan *monitoring interfaces* yang ada di-*command line*.



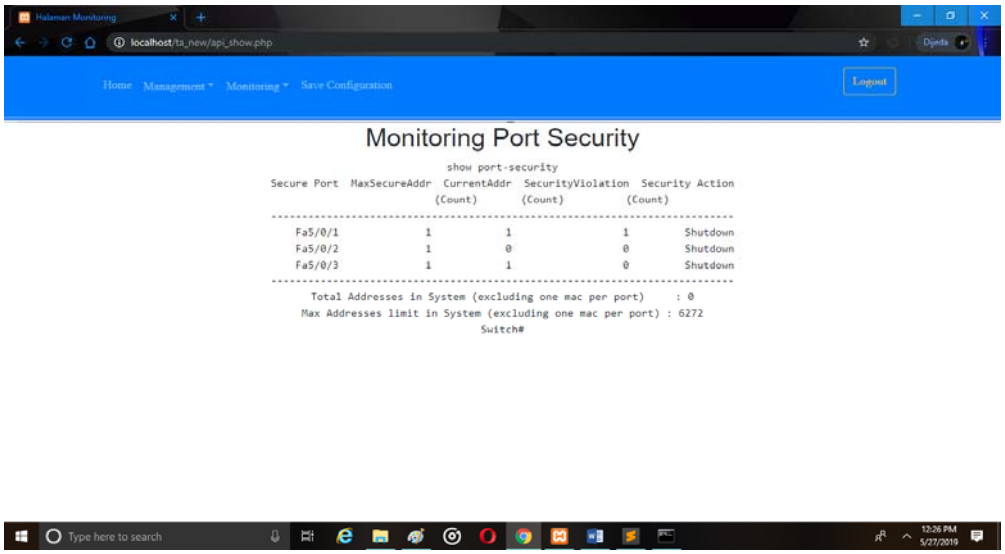
Gambar 4.17 Tampilan Submenu *Monitoring Interfaces* pada Sistem *Monitoring Interfaces Ethernet*



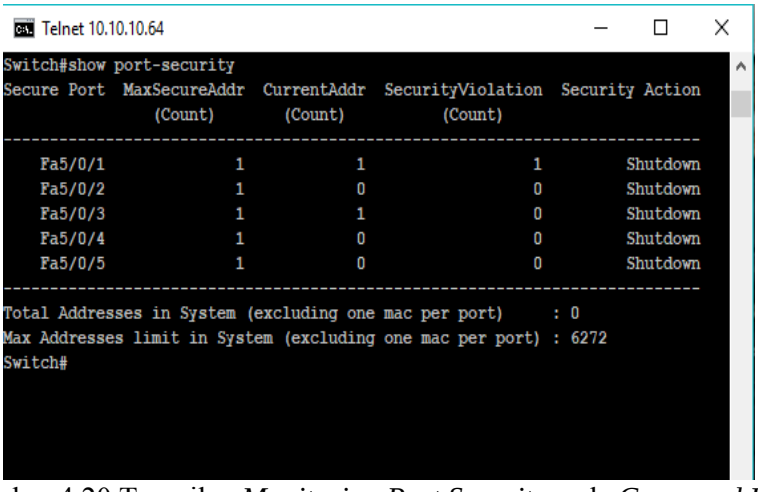
Gambar 4.18 Tampilan *Monitoring Interfaces* pada *Command Line*

4.7.2. Tampilan Submenu *Monitoring Port Security* pada Sistem *Monitoring Interfaces Ethernet*

Tampilan submenu m*onitoring port security* merupakan salah satu tampilan yang ada di menu *monitoring*. Di dalam submenu ini administrator dapat dengan mudah memantau atau melihat *interfaces* mana saja yang terkonfigurasi *port security*. Tampilan submenu *monitoring port security* pada sistem *monitoring interfaces ethernet* dapat dilihat pada Gambar 4.19. Sedangkan pada Gambar 4.20 merupakan tampilan *monitoring port security* pada *command line.*



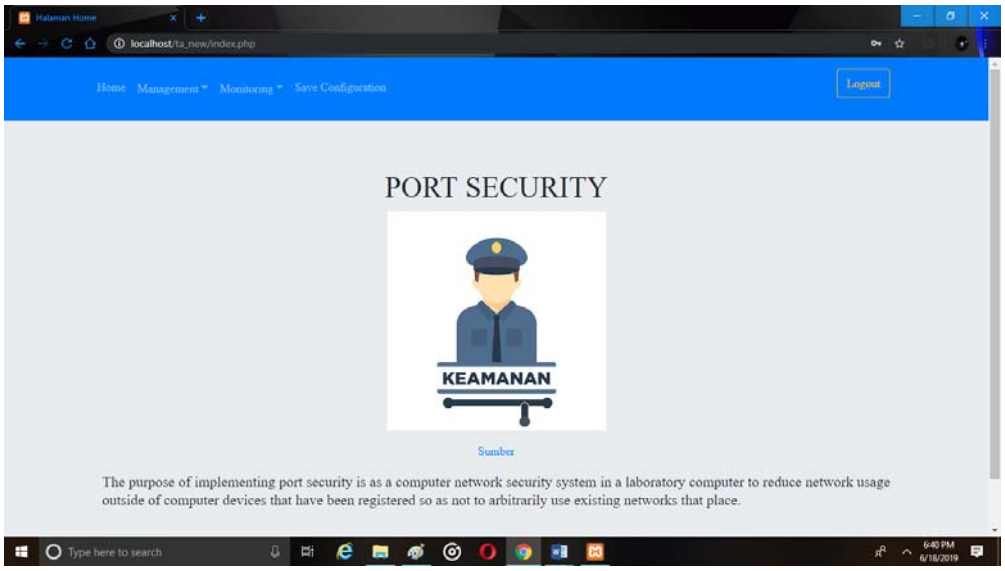
Gambar 4.19 Tampilan Submenu *Monitoring Port Security* pada Sistem *Monitoring Interfaces Ethernet*



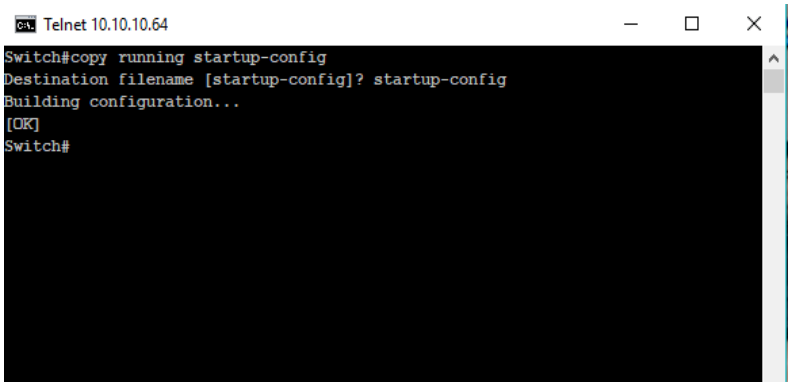
Gambar 4.20 Tampilan *Monitoring Port Security* pada *Command Line*

4.8. Tampilan *Save Configuration* pada Sistem *Monitoring Interfaces Ethernet*

Tampilan *save configuration* pada sistem *monitoring interfaces ethernet* ini digunakan untuk menyimpan hasil konfigurasi atau semua *history management* yang telah dilakukan pada perangkat Cisco Catalyst 3750. Pada Web *monitoring interfaces ethernet* ini menyimpan hasil konfigurasinya sangat mudah yaitu hanya dengan menekan tombol *save* dan semua hasil konfigurasinya sudah tersimpan pada perangkat Cisco Catalyst 3750. Tampilan *save configuration* pada sistem *monitoring interfaces ethernet* dapat dilihat pada Gambar 4.21. Sedangkan *save configuration* pada *command line* dapat dilihat pada Gambar 4.22.



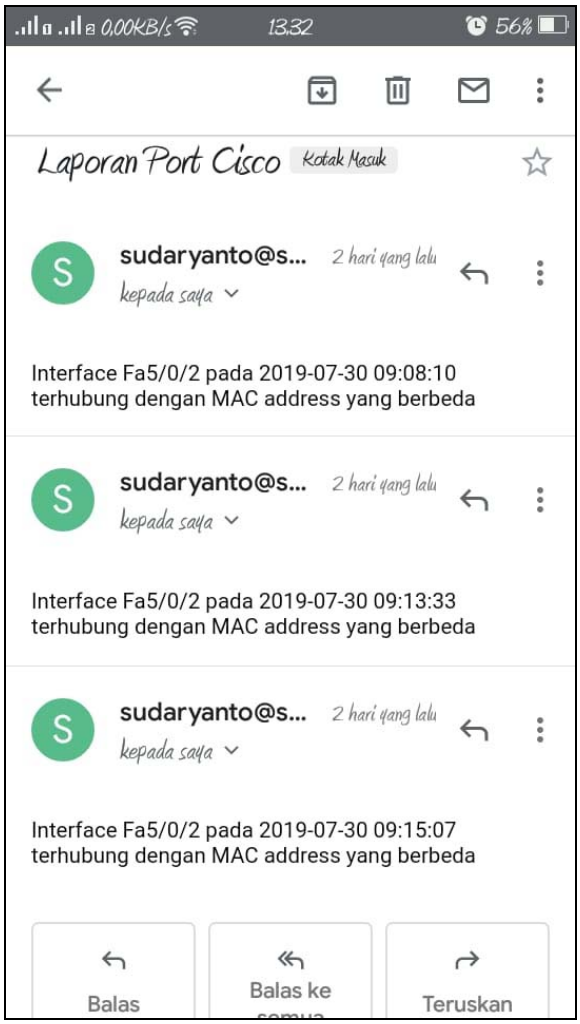
Gambar 4.21 Tampilan *Save Configuration* pada Sistem *Monitoring Interfaces Ethernet*



Gambar 4.22 Tampilan *Save Configuration* pada *Command Line*

4.9. Tampilan *Error Notification* pada Email

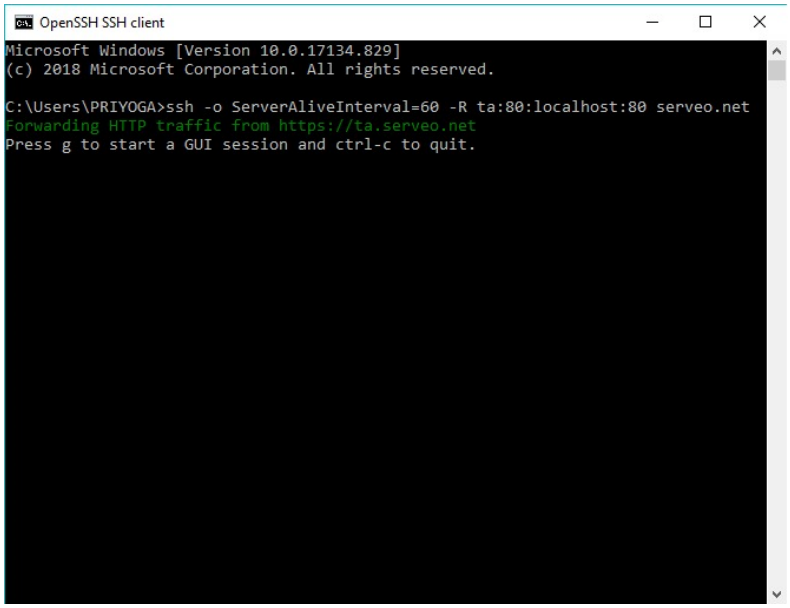
Notifikasi digunakan untuk memudahkan administrator dalam mengetahui perubahan yang terdapat di dalam perangkat Cisco Catalyst 3750yang sedang dikelolanya. Dengan adanya notifikasi administrator dimudahkan agar tidak selalu mengecek secara berkala, apabila terjadi perubahan aktifitas berupa terhubungnya perangkat komputer dengan perangkat Cisco Catalyst 3750 dimana MAC *Address* yang ada di perangkat komputer tidak dikenali oleh perangkat Cisco Catalyst 3750. Dengan demikian apabila terdapat perubahan tersebut maka sistem akan langsung mengirim notifikasi ke email administrator yang sudah diatur dalam program. Tampilan *error notification* di Email dapat dilihat pada Gambar 4.23.



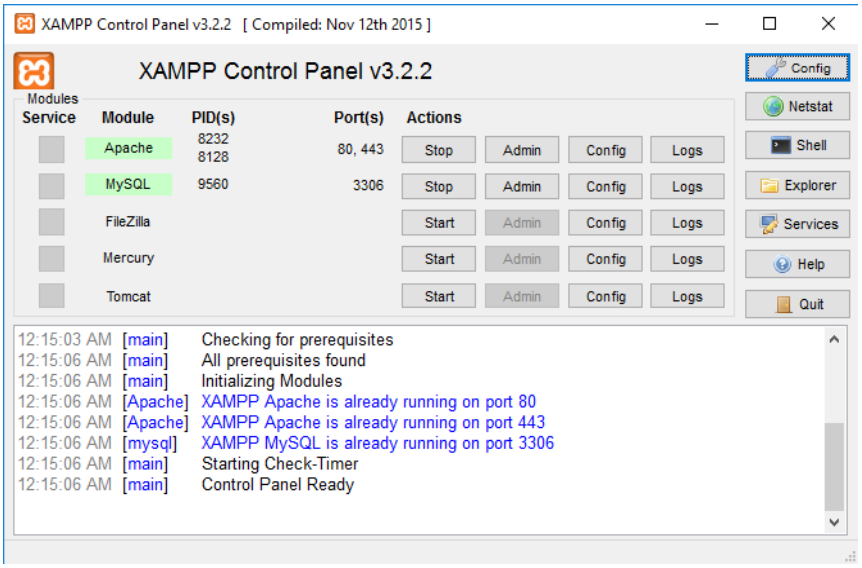
Gambar 4.23 Tampilan *Error Notification* pada Email

4.10. Tampilan *Server* Menggunakan Serveo dan Xampp

Serveo merupakan *server* secara lokal yang digunakan untuk mengakses atau meremote Web. Sedangkan Xampp digunakan untuk menghasilkan halaman Web yang benar kepada *user* berdasarkan kode PHP yang dituliskan oleh pembuat halaman Web. Tampilan *server* menggunakan Serveo dapat dilihat pada Gambar 4.24. Sedangkan tampilan Xampp dapat dilihat pada Gambar 4.25.



Gambar 4.24 Tampilan *Server* Menggunakan Serveo



Gambar 4.25 Tampilan *Server* Menggunakan Xampp

4.11. Pembahasan

Pengujian dilakukan dengan cara tes ping pada setiap komputer untuk mengetahui balasan dari setiap kondisi. Pengujian dilakukan di laboratorium Komputasi Sekolah Tinggi Teknologi Adisutjipto seperti yang terlihat pada Gambar 4.26 dan Gambar 4.27.

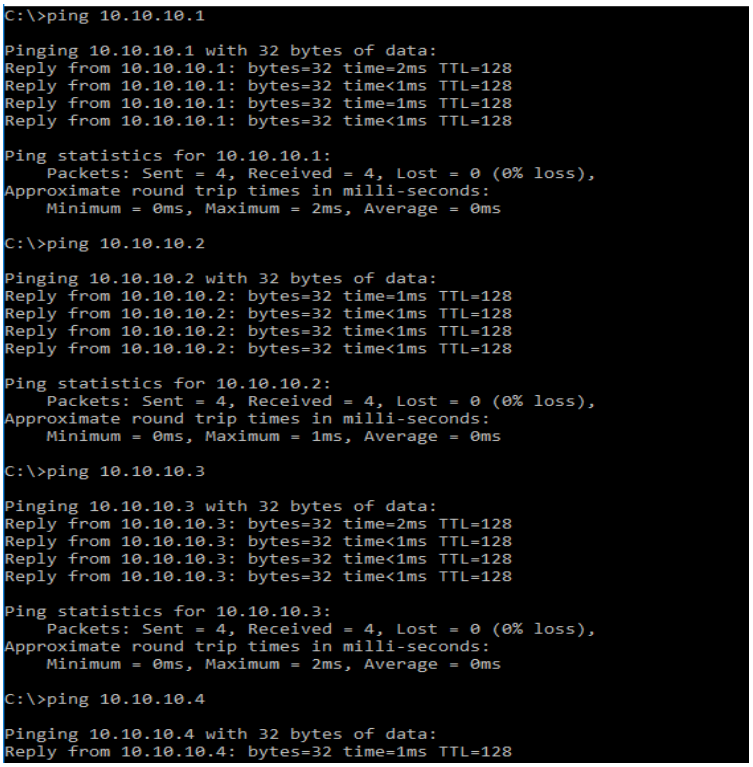
Gambar 4.26 Laboratorium Komputasi STTA

Gambar 4.27 *Switch* Cisco Catalyst 3750

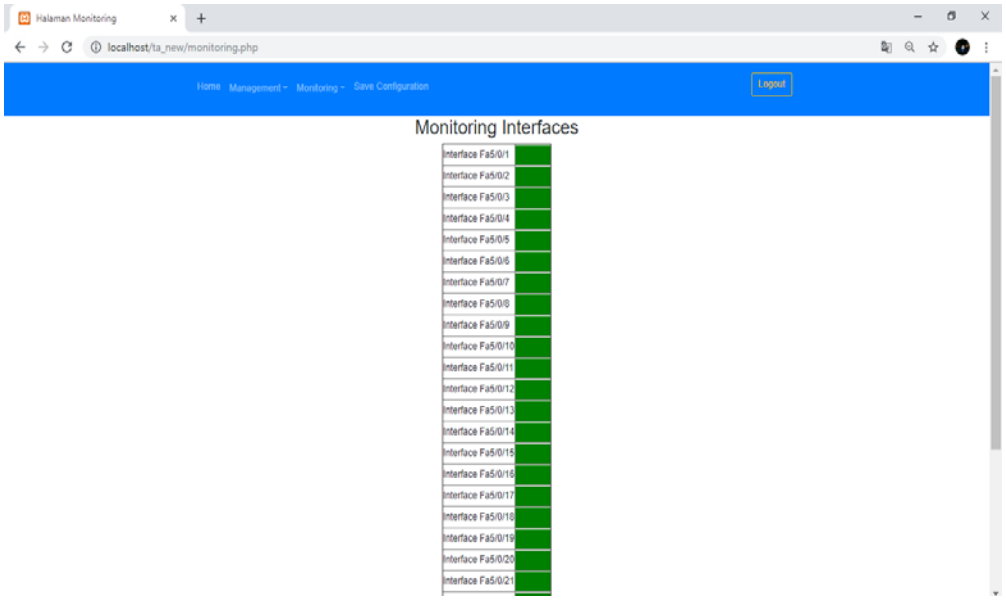
4.11.1 Hubungan antar Komputer Tanpa Konfigurasi *Port Security*

Pada pengujian ini, kondisi semua personal komputer belum terkonfigurasi *port security*. Oleh karena itu, jika personal komputer melakukan *request* atau ping pada semua personal komputer dengan *network* yang sama maka akan mendapat balasan *reply*. Proses pengujian (ping) yang terlihat pada Gambar 4.28 dari personal komputer dengan IP *address* 10.10.10.1 dan diulang untuk 20 personal komputer yang IP *address* dan hasilnya dapat dilihat pada Tabel 4.1.

Hasil *monitoring* menggunakan perangkat lunak (*software*) yang dirancang seperti terlihat pada Gambar 4.29 menunjukkan bahwa hasil ping yang interface-nya terhubung dengan personal komputer berwarna hijau atau yang berarti berhasil (*connected*) dimana datanya terlihat pada Tabel 4.1.



Gambar 4.28 Tes Ping antar Komputer tanpa Konfigurasi *Port Security* pada IP 10.10.10.1-4



Gambar 4.29 Tampilan Web *Monitoring Interfaces* Tanpa Konfigurasi *Port Security*

Tabel 4.1 Tes Ping antar Komputer Tanpa Konfigurasi *Port Security*

No	IP Tujuan	*Interface*	Hasil
1.	10.10.10.1	FastEthernet 5/0/1	√
2.	10.10.10.2	FastEthernet 5/0/2	√
3.	10.10.10.3	FastEthernet 5/0/3	√
4.	10.10.10.4	FastEthernet 5/0/4	√
5.	10.10.10.5	FastEthernet 5/0/5	√
6.	10.10.10.6	FastEthernet 5/0/6	√
7.	10.10.10.7	FastEthernet 5/0/7	√
8.	10.10.10.8	FastEthernet 5/0/8	√
9.	10.10.10.9	FastEthernet 5/0/9	√
10.	10.10.10.10	FastEthernet 5/0/10	√
11.	10.10.10.11	FastEthernet 5/0/11	√
12.	10.10.10.12	FastEthernet 5/0/12	√
13.	10.10.10.13	FastEthernet 5/0/13	√
14.	10.10.10.14	FastEthernet 5/0/14	√

Tabel 4.1 Lanjutan

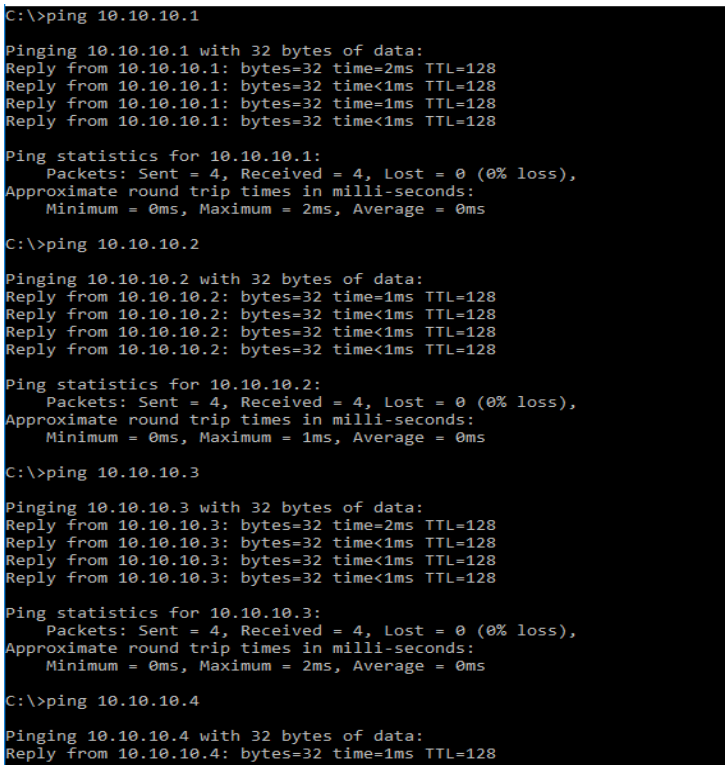
No	IP Tujuan	*Interface*	Hasil
15.	10.10.10.15	FastEthernet 5/0/15	√
16.	10.10.10.16	FastEthernet 5/0/16	√
17.	10.10.10.17	FastEthernet 5/0/17	√
18.	10.10.10.18	FastEthernet 5/0/18	√
19.	10.10.10.19	FastEthernet 5/0/19	√
20.	10.10.10.20	FastEthernet 5/0/20	√

Dengan Keterangan:
√ = berhasil
x = gagal

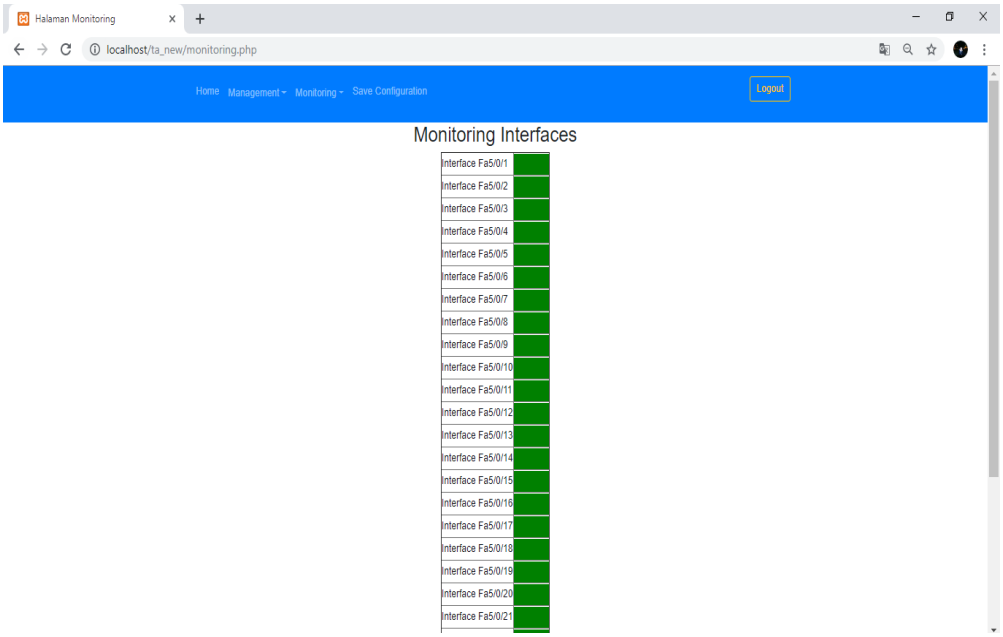
4.11.2 Hubungan antar Komputer Sudah Terkonfigurasi *Port Security*

Pada pengujian ini, semua *interface* sudah terkonfigurasi *port security*. Oleh karena itu, jika personal komputer melakukan *request* atau ping pada semua personal komputer dengan *network* yang sama maka akan mendapat balasan *reply*. Proses pengujian (ping) yang terlihat pada Gambar 4.30 dari personal komputer dengan IP *address* 10.10.10.1 dan diulang untuk 20 personal komputer yang IP *address* dan hasilnya dapat dilihat pada Tabel 4.2.

Hasil *monitoring* menggunakan perangkat lunak (*software*) yang dirancang seperti terlihat pada Gambar 4.31 menunjukkan bahwa hasil ping yang interface-nya terhubung dengan personal komputer berwarna hijau atau yang berarti berhasil (*connected*) dimana datanya terlihat pada Tabel 4.2.



Gambar 4.30 Tes Ping antar Komputer Sudah Terkonfigurasi *Port Security* pada IP 10.10.10.1-4



Gambar 4.31 Tampilan Web *Monitoring Interfaces* Sudah Terkonfigurasi *Port Security*

Tabel 4.2 Tes Ping antar Komputer Sudah Terkonfigurasi *Port Security*

No	IP Tujuan	*Interface*	Hasil
1.	10.10.10.1	FastEthernet 5/0/1	√
2.	10.10.10.2	FastEthernet 5/0/2	√
3.	10.10.10.3	FastEthernet 5/0/3	√
4.	10.10.10.4	FastEthernet 5/0/4	√
5.	10.10.10.5	FastEthernet 5/0/5	√
6.	10.10.10.6	FastEthernet 5/0/6	√
7.	10.10.10.7	FastEthernet 5/0/7	√
8.	10.10.10.8	FastEthernet 5/0/8	√
9.	10.10.10.9	FastEthernet 5/0/9	√
10.	10.10.10.10	FastEthernet 5/0/10	√
11.	10.10.10.11	FastEthernet 5/0/11	√
12.	10.10.10.12	FastEthernet 5/0/12	√
13.	10.10.10.13	FastEthernet 5/0/13	√
14.	10.10.10.14	FastEthernet 5/0/14	√
15.	10.10.10.15	FastEthernet 5/0/15	√
16.	10.10.10.16	FastEthernet 5/0/16	√
17.	10.10.10.17	FastEthernet 5/0/17	√
18.	10.10.10.18	FastEthernet 5/0/18	√
19.	10.10.10.19	FastEthernet 5/0/19	√
20.	10.10.10.20	FastEthernet 5/0/20	√

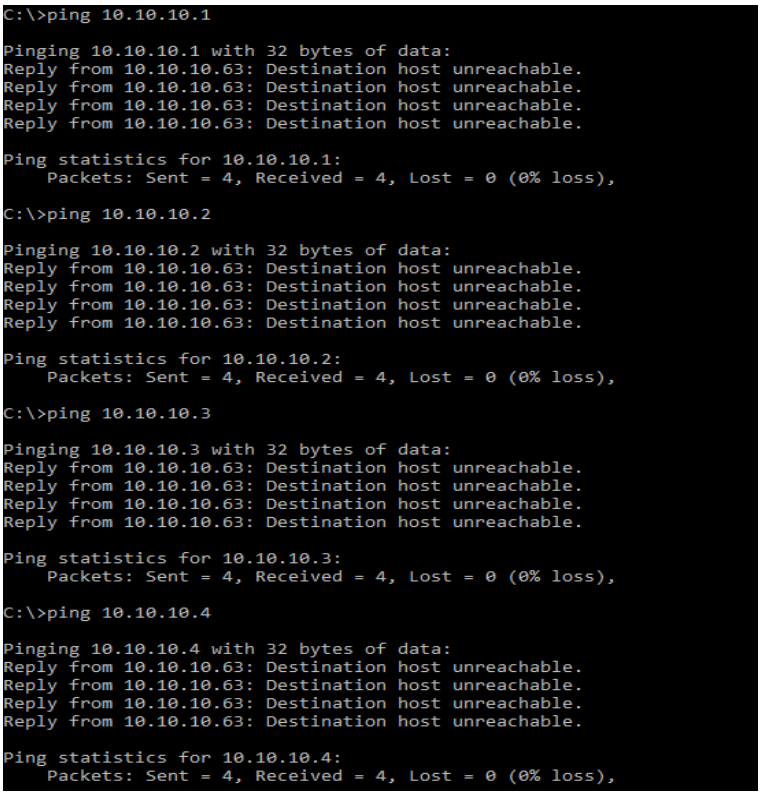
Dengan Keterangan:
√ = berhasil
x = gagal

4.11.3 Hubungan antar Komputer setelah Ditukar *Interface*-nya

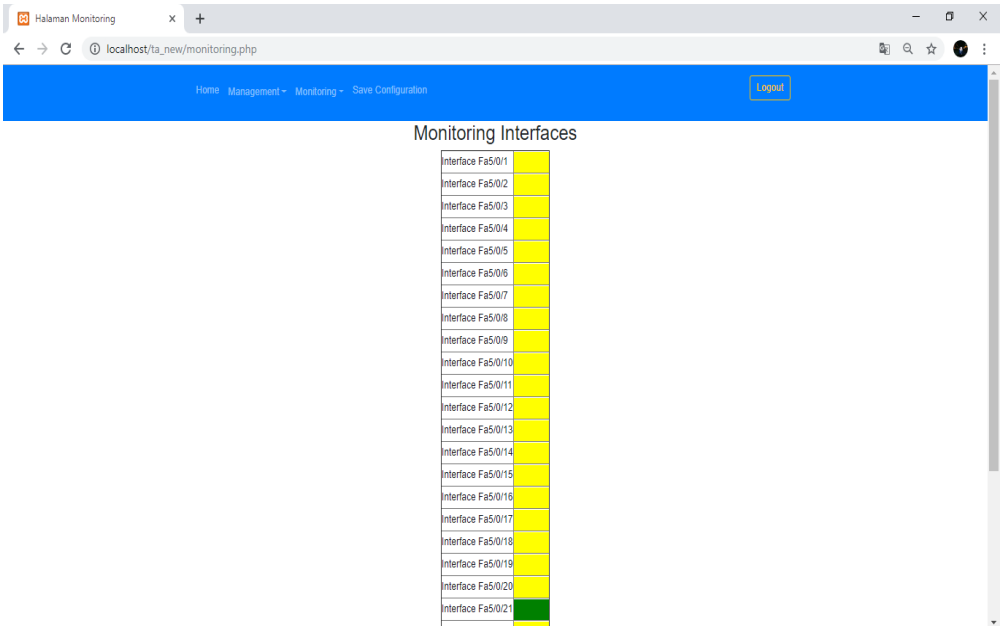
Pada pengujian ini, komputer yang seharusnya berada di *interface* FastEthernet 5/0/1 dipindah ke *interface* FastEthernet 5/0/20 dan begitu juga dengan *interface* yang lain. Oleh karena itu, jika personal komputer melakukan *request* atau ping pada semua personal komputer dengan *network* yang sama maka akan mendapat balasan *destination host unreachable* sekaligus *interface*-nya akan

otomatis mati (*shutdown*). Hal ini dikarenakan MAC *address* yang baru masuk dibandingkan dengan MAC *address* yang ada di *switching table* pada *interface* tersebut, jika MAC *address*-nya berbeda maka *action*-nya akan dijalankan. Proses pengujian (ping) yang terlihat pada Gambar 4.32 dari personal komputer(*server*) dengan IP *address* 10.10.10.63 dan diulang untuk 20 personal komputer yang IP *address* dan hasilnya dapat dilihat pada Tabel 4.3.

Hasil *monitoring* menggunakan perangkat lunak (*software*) yang dirancang seperti terlihat pada Gambar 4.33 menunjukkan bahwa hasil ping yang interface-nya terhubung dengan personal komputer berwarna kuning atau yang berarti gagal (*error disabled*) dimana datanya terlihat pada Tabel 4.3.



Gambar 4.32 Tes Ping antar Komputer setelah Ditukar *Interface*-nya pada IP 10.10.10.1-4



Gambar 4.33 Tampilan Web *Monitoring Interfaces* setelah Ditukar *Interface*-nya

Tabel 4.3 Tes Ping antar Komputer setelah Ditukar *Interface*-nya

No	IP Tujuan	*Interface*	Hasil
1.	10.10.10.1	FastEthernet 5/0/20	x
2.	10.10.10.2	FastEthernet 5/0/19	x
3.	10.10.10.3	FastEthernet 5/0/18	x
4.	10.10.10.4	FastEthernet 5/0/17	x
5.	10.10.10.5	FastEthernet 5/0/16	x
6.	10.10.10.6	FastEthernet 5/0/15	x
7.	10.10.10.7	FastEthernet 5/0/14	x
8.	10.10.10.8	FastEthernet 5/0/13	x
9.	10.10.10.9	FastEthernet 5/0/12	x
10.	10.10.10.10	FastEthernet 5/0/11	x
11.	10.10.10.11	FastEthernet 5/0/10	x
12.	10.10.10.12	FastEthernet 5/0/9	x
13.	10.10.10.13	FastEthernet 5/0/8	x
14.	10.10.10.14	FastEthernet 5/0/7	x

Tabel 4.3 Lanjutan

No	IP Tujuan	*Interface*	Hasil
15.	10.10.10.15	FastEthernet 5/0/6	x
16.	10.10.10.16	FastEthernet 5/0/5	x
17.	10.10.10.17	FastEthernet 5/0/4	x
18.	10.10.10.18	FastEthernet 5/0/3	x
19.	10.10.10.19	FastEthernet 5/0/2	x
20.	10.10.10.20	FastEthernet 5/0/1	x

Dengan Keterangan:
√ = berhasil
x = gagal

BAB V
PENUTUP

5.1 Kesimpulan

Dari hasil penelitian dan pembahasan yang ada di *Monitoring Interfaces Ethernet* pada Cisco Catalyst 3750 untuk Menjamin Keamanan Penggunaan Jaringan Komputer dapat diambil beberapa kesimpulan sebagai berikut:

1. Sistem *Monitoring Interfaces Ethernet* pada Cisco Catalyst 3750 dapat membantu administrator dalam memantau jaringan komputer secara *real time* dari tempat yang berbeda tanpa harus bersentuhan langsung dengan perangkatnya.
2. Berdasarkan uji coba program yang telah dilakukan, didapatkan bahwa sistem *Monitoring Interfaces Ethernet* pada Cisco Catalyst 3750 memiliki fungsi atau perintah yang sama dengan perintah yang ada di *command line*.
3. Berdasarkan uji coba program yang telah dilakukan, didapatkan bahwa sistem dapat berjalan di Browser pada perangkat personal komputer dan *smartphone* secara *responsive* pada *device* yang diujikan.

5.2 Saran

Adapun saran untuk pengembangan lebih lanjut terhadap tugas akhir ini yaitu sebagai berikut:

1. Sistem *Monitoring Interfaces Ethernet* dapat dikembangkan dengan berbasis Android.
2. *Monitoring* yang dilakukan pada perangkat Cisco Catalyst 3750 tidak hanya *port security*.

DAFTAR PUSTAKA

Affandi, K. *Mengenal Apa itu Xampp, Apache, PHP, dan MySQL*. http://khaerulaffandi.weebly.com/mengenal-apa-itu-xamppapachephp-dan-mysql.html. 24 Mei 2019 (17:40).

Cisco. *Cisco SF350-24P 24-Port 10/100 POE Managed Switch*. https://www.cisco.com/c/en/us/support/switches/sf350-24p-24-port-10-100 -poe-managed-switch/model.html#~tab-documents. 24 Mei 2019 (17:15).

Cisco. *Cisco SG100-24 24-Port Gigabit Switch*. https://www.cisco.com/c/en/us/ support/switches/sf350-24p-24-port-10-100-poe-managed-switch/model. html#~tab-documents. 24 Mei 2019 (17:25).

Cisco (2012). *Overview on Cisco Catalyst 3750 Switches:Features, Technology, Intelligent Switching, Network Management*. http://ciscorouterswitch. over-blog.com/article-overview-on-cisco-catalyst-3750-switches-features-technology-intelligent-switching-network-manag-101871689.html. 31 Juli 2019 (17:53).

Gobel. M. A. A., Sumarsono, dan Y. Indrianingsih. (2012). Notification Of Security Threats On The Internet Proxy Server Is A Server-Based Short Message Service (SMS). *In Compiler STT Adisutjipto Yogyakarta*, 1(1), 77-90.

Herliana, A., dan P.M. Rasyid. (2016). Sistem Informasi Monitoring Pengembangan Software pada Tahap Development Berbasis Web. *Jurnal Informatika*, 3(1), 41-50.

Imanudin, A. (2019) . Akses Server Lokal dari internet Menggunakan Serveo. http://www.google.com/amp/s/imanudin.com/2019/01/24/akses-server-lokal-dari-internet-menggunakan-serveo/amp/. 24 Mei 2019 (17:50).

Ocanitra, R, dan M. Ryansyah. (2019). Implementasi Sistem Keamanan Jaringan Menggunakan Firewall Security Port pada Vitaa Multi Oxygen. *Jurnal Sistem dan Teknologi Informasi*, 7(1), 52-59.

Pratama, I. P. A. E. (2014). *Handbook Jaringan Komputer Teori dan Praktik Berbasiskan Open Source*. Informatika. Bandung.

Sidik, B. (2014). *Pemrograman WEB dengan PHP*. Edisi Revisi Kedua. Informatika. Bandung.

_____., Pohan, H., I. (2001). *Pemrograman Web dengan HTML*. Edisi 3. Informatika. Bandung.

Sofana, I. (2015). *Membangun Jaringan Komputer Mudah Membuat Jaringan Komputer (Wire & Wireless) untuk Pengguna Windows dan Linux*. Cetakan Pertama. Informatika. Bandung.

_____. (2012). *CISCO CCNP dan Jaringan Komputer (Materi Router, Switch, & Troubleshooting)*. Informatika. Bandung.

S, Rossa, A., dan M. Shalahuddin. (2015). *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Informatika. Bandung.

Sudaryanto, S. (2018). Implementation Port Security For Security System Network At The Computing Laboratory Of Adisutjipto Technology College. *In Conference SENATIK STT Adisutjipto Yogyakarta*, 4, 257-265.

_____. (2018). The Effect Of Multi Layer Switch For Data Transfer Speeds On Computer Network. *In Compiler STT Adisutjipto Yogyakarta*, 7(2), 85-90.

Sulaiman, O. K. (2016). Analisis Sistem Keamanan Jaringan Dengan Menggunakan Switch Port Security. *CESS (Journal Of Computer Engineering, System And Science)*, 1(1), 9-14.

Zulfi, I. (2014). *Makalah Ethernet*. http://ineshazulfi1.blogspot.com/2014/11/ makalah-ethernet.html. 31 Juli 2019 (17:35)

LAMPIRAN

MONITORING INTERFACES FASTETHERNET ON CISCO CATALYST 3750 TO ENSURE USE OF THE SECURITY COMPUTER NETWORK IN STTA COMPUTING LABORATORIES

Sudaryanto[1], Dwi Nurhayati[2]
Program Studi Informatika
Sekolah Tinggi Teknologi Adisutjipto
Jl. Janti, Blok R, Lanud Adisutjipto Yogyakarta
Email : [1]sudaryanto@stta.ac.id, [2]nrhyt78@gmail.com

Abstract

Cisco is a company that concentrates on hardware and software related to computer networks. One of the hardware produced by Cisco is a switch device that can be used for management of a computer network. Many types of switches that have been produced by Cisco, one of which is the Cisco Catalyst 3750. In the configuration and monitoring of the Cisco Catalyst 3750 user / administrator must do management with command line based configuration. This is because Cisco has not facilitated its users with user interface based configurations. So the user is required to know the code syntax to execute a command on the switch. In this study, researchers will discuss how to create a user interface for monitoring web-based Fastethernet Interfaces on the Cisco Catalist 3750 and use notifications E-mail by utilizing an API to determine the up or down status of a network device. The results of the test show that the application can be monitored by the administrator remotely in real time, the user interface can run well on the personal computer browser and smartphone responsively.

Keywords: Port security, monitoring, interfaces ethernet, Cisco Catalyst 3750, API.

Abstrak

Cisco merupakan salah satu perusahan yang berkonsentrasi pada *hardware* dan *software* yang berhubungan dengan jaringan komputer. Salah satu *hardware* yang diproduksi oleh *Cisco* adalah sebuah perangkat *switch* yang dapat digunakan untuk manajemen suatu jaringan komputer. Banyak jenis *switch* yang telah di produksi oleh *Cisco* salah satunya adalah Cisco Catalyst 3750. Dalam konfigurasi dan monitoring Cisco Catalyst 3750 pengguna/administrator harus melakukannya manajemen dengan konfigurasi berbasis *command line*. Hal ini dikarenakan *Cisco* belum memfasilitasi penggunanya dengan konfigurasi berbasis *user interface*. Sehingga pengguna diharuskan mengetahui sintak kode untuk mengeksekusi suatu perintah pada *switch* tersebut. Pada penelitian ini peneliti akan membahas tentang bagaimana membuat *user interface* untuk monitoring *Interfaces Fastethernet* berbasis web pada Cisco Catalist 3750 dan menggunakan notifikasi *E-mail* dengan memanfaatkan API untuk mengetahui status *up* atau *down* dari sebuah perangkat jaringan. Hasil dari pengujian menunjukkan bahwa aplikasi dapat dimonitoring administrator dari jarak jauh secara *real time*, *user interface* dapat berjalan dengan baik pada browser perangkat personal komputer dan *smartphone* secara *responsive*.

Kata kunci : *Port security*, *monitoring*, *interfaces ethernet,* Cisco Catalyst, API.

1. Pendahuluan

Penelitian penelitian sebelumnya banyak membahas tentang manajemen keamanan jaringan baik menggunakan port-port yang tersedia pada *switch* yaitu : *default / static port security, port security dynamic learning, sticky port security* [1], dan juga monitoring jaringan dengan menggunakan menggunakan SNMP [2][3], mikrotik [4], sms [5] dan menggunakan web [6] dimana monitoring jaringan mengunakan piranti mikrotik dengan api-mikrotik, belum ada penelitian yang memanage dan memonitoring jaringan dengan menggunakan piranti *cisco* yang melibatkan *api-cisco*. [7][8] Dalam penelitiannya membahas tentang switch multilayer dan implementasi *port security* pada sistem keamanan jaringan untuk mengurangi pengguna yang memanfaatkan jaringan Laboratorium Komputasi untuk penggunaan *bandwidth* di luar perangkat komputer yang telah diijinkan atau didaftarkan, tetapi konfigurasinya masih secara manual yaitu menggunakan *command line.* Pada penelitian ini peneliti akan membahas tentang bagaimana membuat *user interface* untuk manajemen *Port Security* pada *Cisco Catalyst 3750* berbasis *web* sehingga konfigurasi dalam manajemen *Port Security Cisco* dapat beralih dari konfigurasi *command line* menjadi konfigurasi berbasis *User Interface* dan manajemen Catalyst 3750 dapat dilakukan dari mana saja dan kapanpun.

Selain melakukan manajemen peneliti juga akan melakukan *monitoring* yang merupakan sebuah kegiatan yang bertujuan untuk memantau tentang perubahan status yang ada di suatu perangkat jaringan, sehingga penelitian ini mempunyai tujuan untuk *monitoring* perangkat Cisco Catalyst 3750 secara *real time* dari tempat yang berbeda tanpa harus bersentuhan langsung dengan perangkatnya dengan menggunakan *web* dan *notifikasi E-mail* sebagai monitoringnya. Banyak hal dalam jaringan yang bisa dimonitoring, salah satu diantaranya adalah status *up* atau *down* dari sebuah perangkat jaringan.

Adanya sistem manajemen berbasis *user interface (web)* dan monitoring dapat mempermudah administrator jaringan dalam memantau sistem jaringan yang berada di lapangan dari tempat yang berbeda tanpa harus mengecek secara berkala dan bersentuhan langsung dengan perangkat tersebut.

2. Metodologi Penelitian

2.1 *Switch*

Switch merupakan perangkat keras penghubung di dalam jaringan komputer yang lebih banyak digunakan saat ini dibandingkan hub [9]. Hal ini disebabkan karena dengan fungsi yang serupa dengan hub, *Switch* memiliki dua buah kelebihan utama dibandingkan hub. Kelebihan-kelebihan yang dimiliki oleh *switch* yaitu:

a. *Switch* memiliki kemampuan untuk membaca alamat fisik (MAC *Address*) dari setiap komputer yang terhubung ke dalam *switch* bersangkutan. *Switch* menyimpan alamat fisik (MAC *Address*) dari setiap komputer yang terhubung ke dalam *switch* tersebut beserta dengan nomor *port switch* yang digunakan oleh komputer bersangkutan.
b. *Switch* memiliki kemampuan untuk melakukan filter terhadap paket data yang keluar masuk *switch*. Hal ini akan memberikan keamanan paket data (terkait dengan pengendusan paket data di dalam jaringan komputer).

Switch bekerja di dua buah layer pada jaringan komputer, yaitu Data *Link Layer* dan *Physical Layer*. Pada Data *Link Layer*, terjadi proses pengecekan terhadap alamat fisik jaringan (MAC *Address*) untuk otentikasi alamat fisik komputer yang terhubung ke

switch, untuk kemudian disesuaikan dengan alamat jaringan pada *Network Layer* (IP *Address*). Pada *Physical Layer* terjadi proses pengolahan sinyal digital.

2.2 *Port Security*

Port Security membatasi jumlah MAC *address* yang diizinkan terhubung dengan tiap *port* dan juga dapat membatasi MAC *address* mana saja yang diizinkan [10][11].

2.3 Perangkat yang Dipergunakan

Dalam pembuatan sistem *management interfaces ethernet* diperlukan *hardware* dan *software* yang digunakan sebagai proses penunjang dalam pembuatan sistem *management interfaces ethernet.*

a. *Hardware* (perangkat keras) merupakan komponen perangkat yang dapat dilihat secara kasat mata dan dapat disentuh secara fisik. Adapun spesifikasi *hardware* yang digunakan dalam pembuatan sistem ini, sebagai berikut:

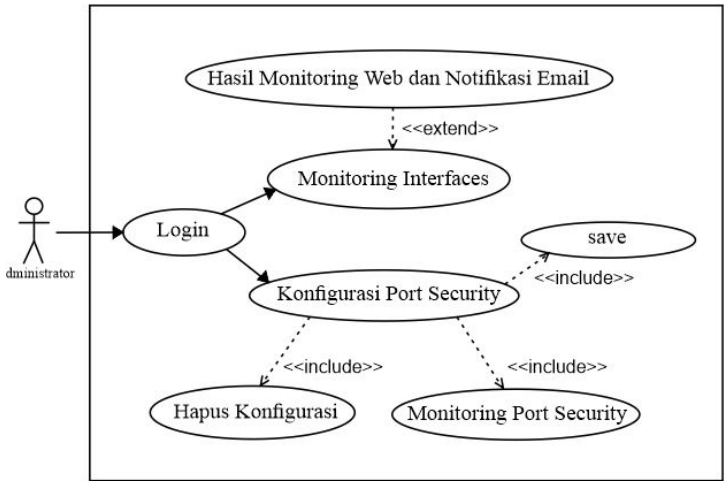
1) Cisco Catalyst 3750
2) Laptop (HP Probook 4321S)

b. *Software* (perangkat lunak) merupakan komponen yang tidak terkihat secara fisik, tetapi terdapat dalam sebuah komputer.

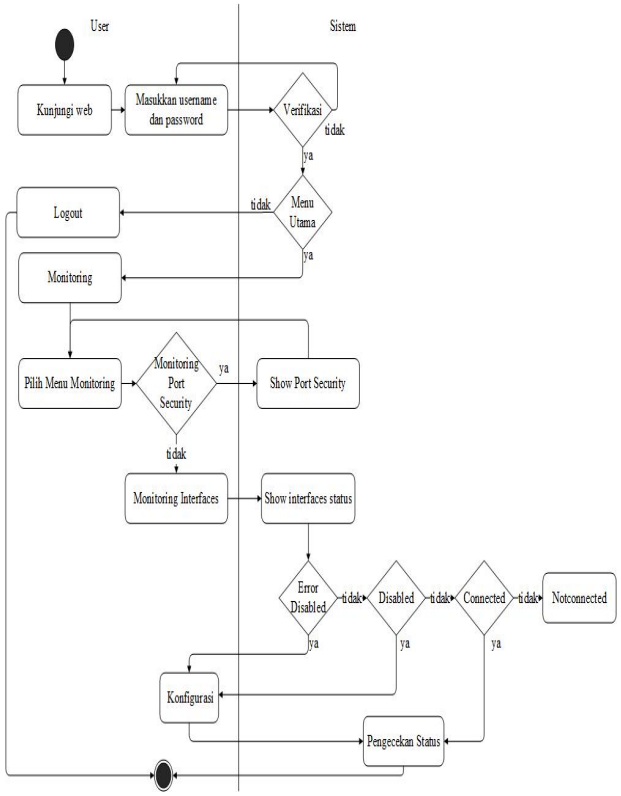
1) Sistem Operasi Windows 10 Pro
2) Bahasa Pemrograman PHP dan HTML

2.4 Metode Penelitian

Pada Gambar 1 dan 2 dijelaskan bahwa administrator bisa melakukan *monitoring* untuk mengetahui status *up* atau *down* dari sebuah perangkat jaringan melalui web dan notifikasi email selain itu administrator juga bisa melakukan konfigurasi untuk melakukan perubahan status pada perangkat jaringan dari status *up* ke *down* ataupun sebaliknya.



Gambar 1. *Use Case* Diagram Sistem *Management Interfaces Ethernet*

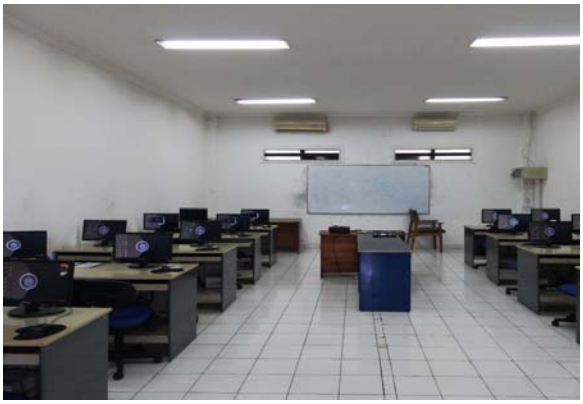


Gambar 2. *Activity* Diagram *Monitoring* pada Sistem *Management Interfaces Ethernet*

3. Hasil dan Pembahasan

3.1 Pengujian *interface fasethernet* dengan *Port Security*

Pengujian dilakukan dengan cara tes ping pada setiap komputer untuk mengetahui balasan dari setiap kondisi. Pengujian dilakukan di laboratorium Komputasi Sekolah Tinggi Teknologi Adisutjipto seperti yang terlihat pada Gambar 3 dan Gambar 4.



Gambar 3. Laboratorium Komputasi STTA

Gambar 4. *Switch* Cisco Catalyst 3750

Pada pengujian ini, kondisi semua personal komputer belum terkonfigurasi *port security* dan diulang dengan kondisi semua personal komputer terkonfigurasi *port security*. Oleh karena itu, jika personal komputer melakukan *request* atau ping pada semua personal komputer dengan *network* yang sama maka akan mendapat balasan *reply*. Proses pengujian (ping) yang terlihat pada Gambar 5 dari personal komputer dengan IP *address* 10.10.10.1 dan diulang untuk 14 personal komputer yang IP *address* dan hasilnya dapat dilihat pada Tabel 1.

Tabel 1. Ping antar Komputer Tanpa dan Dengan Konfigurasi *Port Security*

No	IP Tujuan	*Interface*	Hasil
1.	10.10.10.1	FastEthernet 5/0/1	√
2.	10.10.10.2	FastEthernet 5/0/2	√
3.	10.10.10.3	FastEthernet 5/0/3	√
4.	10.10.10.4	FastEthernet 5/0/4	√
5.	10.10.10.5	FastEthernet 5/0/5	√
6.	10.10.10.6	FastEthernet 5/0/6	√
7.	10.10.10.7	FastEthernet 5/0/7	√
8.	10.10.10.8	FastEthernet 5/0/8	√
9.	10.10.10.9	FastEthernet 5/0/9	√
10.	10.10.10.10	FastEthernet 5/0/10	√
11.	10.10.10.11	FastEthernet 5/0/11	√
12.	10.10.10.12	FastEthernet 5/0/12	√
13.	10.10.10.13	FastEthernet 5/0/13	√
14.	10.10.10.14	FastEthernet 5/0/14	√

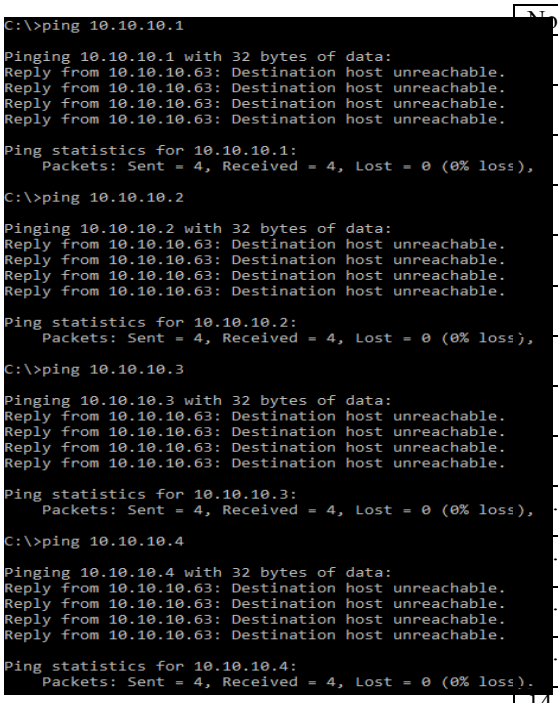
```
C:\>ping 10.10.10.1

Pinging 10.10.10.1 with 32 bytes of data:
Reply from 10.10.10.1: bytes=32 time=2ms TTL=128
Reply from 10.10.10.1: bytes=32 time<1ms TTL=128
Reply from 10.10.10.1: bytes=32 time=1ms TTL=128
Reply from 10.10.10.1: bytes=32 time<1ms TTL=128

Ping statistics for 10.10.10.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 2ms, Average = 0ms

C:\>ping 10.10.10.2

Pinging 10.10.10.2 with 32 bytes of data:
Reply from 10.10.10.2: bytes=32 time=1ms TTL=128
Reply from 10.10.10.2: bytes=32 time<1ms TTL=128
Reply from 10.10.10.2: bytes=32 time<1ms TTL=128
Reply from 10.10.10.2: bytes=32 time<1ms TTL=128

Ping statistics for 10.10.10.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms

C:\>ping 10.10.10.3

Pinging 10.10.10.3 with 32 bytes of data:
Reply from 10.10.10.3: bytes=32 time=2ms TTL=128
Reply from 10.10.10.3: bytes=32 time<1ms TTL=128
Reply from 10.10.10.3: bytes=32 time<1ms TTL=128
Reply from 10.10.10.3: bytes=32 time<1ms TTL=128

Ping statistics for 10.10.10.3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 2ms, Average = 0ms

C:\>ping 10.10.10.4

Pinging 10.10.10.4 with 32 bytes of data:
Reply from 10.10.10.4: bytes=32 time=1ms TTL=128
```

Gambar 5. Tes Ping antar komputer

3.2 Pengujian hubungan antara komputer setelah ditukar interfacenya

Pada pengujian ini, komputer yang seharusnya berada di *interface* FastEthernet 5/0/1 dipindah ke *interface* FastEthernet 5/0/20 dan begitu juga dengan *interface* yang lain. Oleh karena itu, jika personal komputer melakukan *request* atau ping pada semua personal komputer dengan *network* yang sama maka akan mendapat balasan *destination host unreachable* sekaligus *interface*-nya akan

otomatis mati (*shutdown*). Hal ini dikarenakan MAC *address* yang baru masuk dibandingkan dengan MAC *address* yang ada di *switching table* pada *interface* tersebut, jika MAC *address*-nya berbeda maka *action*-nya akan dijalankan. Proses pengujian (ping) yang terlihat pada Gambar 6 dari personal komputer(*server*) dengan IP *address* 10.10.10.63 dan diulang untuk 14 personal komputer yang IP *address* dan hasilnya dapat dilihat pada Tabel 2.



Gambar 6. Tes Ping antar komputer

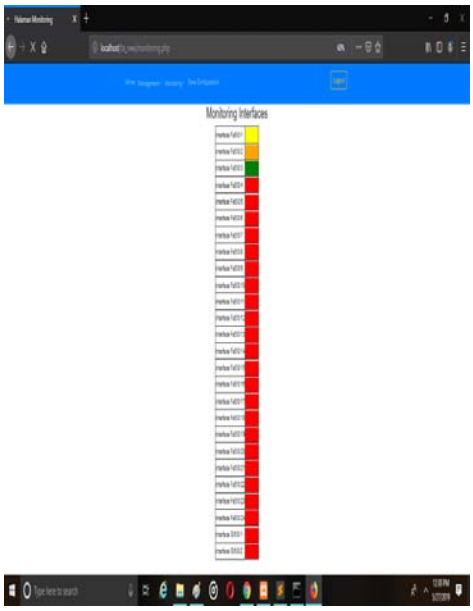
Tabel 2. Tes Ping antar Komputer setelah Ditukar *Interface*-nya

No	IP Tujuan	*Interface*	Hasil
	10.10.10.1	FastEthernet 5/0/20	x
	10.10.10.2	FastEthernet 5/0/19	x
	10.10.10.3	FastEthernet 5/0/18	x
	10.10.10.4	FastEthernet 5/0/17	x
	10.10.10.5	FastEthernet 5/0/16	x
	10.10.10.6	FastEthernet 5/0/15	x
	10.10.10.7	FastEthernet 5/0/14	x
	10.10.10.8	FastEthernet 5/0/13	x
	10.10.10.9	FastEthernet 5/0/12	x
.	10.10.10.10	FastEthernet 5/0/11	x
.	10.10.10.11	FastEthernet 5/0/10	x
.	10.10.10.12	FastEthernet 5/0/9	x
.	10.10.10.13	FastEthernet 5/0/8	x
14.	10.10.10.14	FastEthernet 5/0/7	x

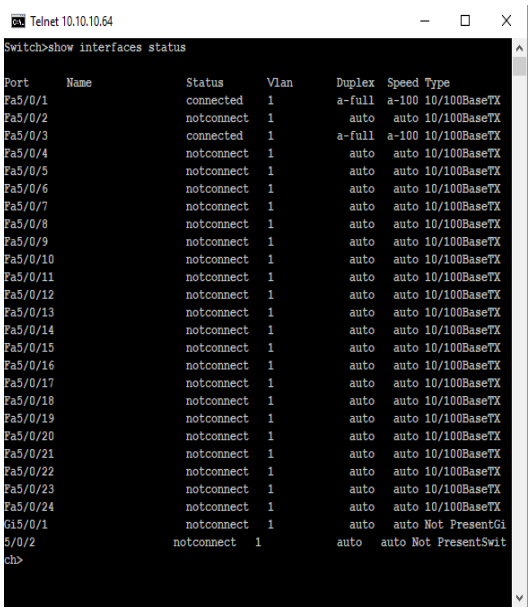
3.3 Tampilan *Monitoring Interfaces* di Sistem *Management Interface Ethernet* dan di *Command Line*

Tampilan *monitoring interfaces* digunakan untuk melihat status dari setiap *interface*. Status yang dimaksud dalam sistem ini adalah *connected, notconnected, disabled, error disabled.* Tampilan *monitoring interfaces* yang ada di sistem *management interfaces ethernet* seperti yang terlihat pada Gambar 7 menunjukkan bahwa status *connected* (warna hijau), *notconnected* (warna merah), *disabled* (warna *orange*), dan *error disabled* (warna kuning). Sedangkan tampilan yang

terlihat pada Gambar 8 merupakan tampilan *monitoring interfaces* yang ada di *command line*, jika ingin melihat status dari *interfaces* maka harus mengetikkan perintahnya terlebih dahulu.



Gambar 7. Tampilan *Monitoring Interfaces* di Sistem *Management Interface Ethernet*

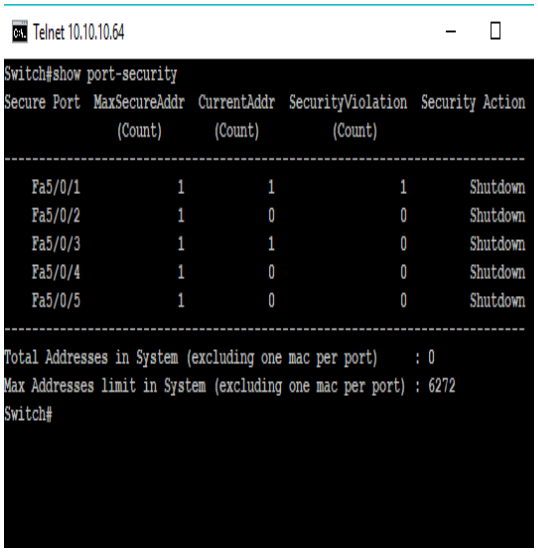


Gambar 8. Tampilan *Monitoring Interfaces* di *Command Line*

3.4 Tampilan *Monitoring Port Security* di Sistem *Management Interface Ethernet* dan di *Command Line*

Tampilan *monitoring port security* digunakan untuk melihat status dari *interfaces* mana saja yang sudah ada konfigurasi *port security*-nya. Tampilan yang terlihat pada Gambar 9 merupakan tampilan *monitoring port security* di sistem *management interface ethernet*. Sedangkan pada Gambar 10 merupakan tampilan *monitoring port security* di *Command Line*.

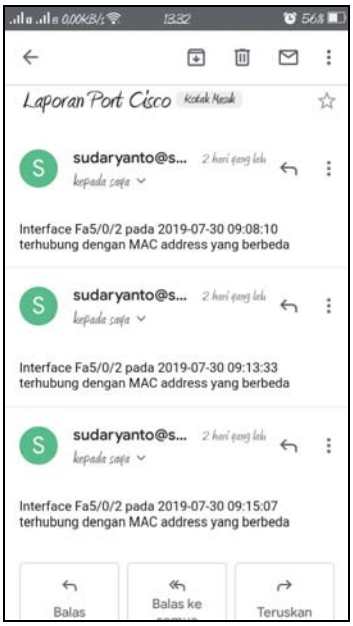
Gambar 9. Tampilan *Monitoring Port Security* di Sistem *Management Interface Ethernet*



Gambar 10. Tampilan *Monitoring Port Security* di *Command Line*

3.5. Tampilan *Notification* pada E-mail

Notifikasi digunakan untuk memudahkan administrator dalam mengetahui perubahan yang terdapat di dalam perangkat Cisco Catalyst 3750, tampilan *notification* di Email dapat dilihat pada Gambar 7.



Gambar 7 Tampilan *Notification* pada Email

Dengan menggunakan *monitoring* berbasis *notifikasi E-mail* administrator tidak perlu selalu mengecek secara berkala untuk mengetahui terjadi perubahan

aktifitas (status *up* dan *down*) ataupun penggunaan *port* yang tidak diijinkan pada sebuah perangkat jaringan (terhubungnya perangkat komputer dengan perangkat Cisco Catalyst 3750 dimana MAC *Address* yang ada di perangkat komputer tidak dikenali oleh perangkat Cisco Catalyst 3750) karena apabila terdapat perubahan tersebut maka sistem akan langsung mengirim *notifikasi* ke E-*mail* administrator yang sudah diatur dalam program. Selain itu administrator juga tidak perlu datang ke peralatan jaringan untuk memastikan bahwa kondisi peralatan sudah berjalan dengan baik atau tidak karena sudah bisa dilakukan dari jarak jauh secara *real time* baik konfigurasi *port* maupun *monitoring port interface Fastethernet*.

4 Kesimpulan

Berdasarkan hasil dari penelitian dengan judul "*Monitoring Interfaces Fastethernet On Cisco Catalyst 3750 To Ensure Use Of The Security Computer Network In Stta Computing*" maka dapat diambil beberapa kesimpulan sebagai berikut:

a. Berdasarkan uji coba program yang telah dilakukan, didapatkan untuk melakukan konfigurasi dan monitoring administrator tidak perlu bersentuhan langsung dengan perangkat jaringan.
b. Berdasarkan uji coba program yang telah dilakukan, didapatkan bahwa jika terjadi perubahan data di *table mac address* pada *switch* port *interface Fastethernet* maka port akan *shutdown.*
c. Berdasarkan uji coba yang telah dilakukan, jika ada perubahan data di *table mac address* pada *switch* yang menyebabkan status *port Up* ataupun *Down*, sistem akan mengirimkan informasi perubahan ke *E-mail* yang sudah ditentukan.

Daftar Pustaka

[1] Sulaiman, K. (2016). ANALISIS SISTEM KEAMANAN JARINGAN DENGAN MENGGUNAKAN *SWITCH PORT SECURITY*. CESS (*Journal Of Computer Engineering, System And Science*) (Vol. 1, ISSN :2502-7131)

[2] Pradikta, R., Affandi, A., & Setijadi, E. (2013). RANCANG BANGUN APLIKASI MONITORING JARINGAN DENGAN MENGGUNAKAN *SIMPLE NETWORK MANAGEMENT PROTOCOL*. Jurnal Teknik Pomits, 2(1), 154-159.

[3] Taftazanie, S., Prasetijo, A. B., & Widianto, E. D. (2017). APLIKASI PEMANTAU PERANGKAT JARINGAN BERBASIS *WEB* MENGGUNAKAN PROTOKOL SNMP DAN NOTIFIKASI SMS. Jurnal Teknologi dan Sistem Komputer, 5(2), 62-68.

[4] Rinaldo, R. (2016). IMPLEMENTASI SISTEM MONITORING JARINGAN MENGGUNAKAN MICROTIK ROUTER OS DI UNIVERSITAS ISLAM BATIK SURAKARTA. Jurnal Emitor, 16(2), 5-12.

[5] Gobel. M. A. A., Sumarsono. S., & Indrianingsih. Y. (2012). *NOTIFICATION OF SECURITY THREATS ON THE INTERNET PROXY SERVER IS A SERVER-BASED SHORT MESSAGE SERVICE* (SMS). In Compiler STT Adisutjipto Yogyakarta, 1(1), 77-90.

[6] Herliana, A., Rasyid, P.M. (2016). SISTEM INFORMASI MONITORING PENGEMBANGAN *SOFTWARE* PADA TAHAP DEVELOPMENT BERBASIS *WEB*. Jurnal Informatika, 3(1), 41-50.

[7] Sudaryanto, S. (2018). *IMPLEMENTATION PORT SECURITY FOR SECURITY SYSTEM NETWORK AT THE COMPUTING LABORATORY OF ADISUTJIPTO TECHNOLOGY COLLEGE*. In Conference SENATIK STT Adisutjipto Yogyakarta, 4, 257-265.

[8] Sudaryanto, S. (2018). *THE EFFECT OF MULTI LAYER SWITCH FOR DATA TRANSFER SPEEDS ON COMPUTER NETWORK*. In Compiler STT Adisutjipto Yogyakarta, 7(2), 85-90.

[9] Pratama, I. P. A. E. (2014). *Handbook Jaringan Komputer Teori dan Praktik Berbasiskan Open Source*. Informatika. Bandung.

[10] Sofana, I. (2015). *Membangun Jaringan Komputer Mudah Membuat Jaringan Komputer (Wire & Wireless) untuk Pengguna Windows dan Linux*. Cetakan Pertama. Informatika. Bandung

[11] Sofana, I. (2012). *CISCO CCNP dan Jaringan Komputer (Materi Router, Switch, & Troubleshooting)*. Informatika. Bandung.

[12] Sulaiman, O. K. (2016). Analisis Sistem Keamanan Jaringan Dengan Menggunakan Switch Port Security. *CESS (Journal Of Computer Engineering, System And Science)*, 1(1), 9-14.