BAB II

TINJAUAN PUSTAKA

2.1 Tinjauan Penelitian Terdahulu

Dalam Penalitian tentang pemanfaatan *telegram-bot* sebagai *monitoring file* pada *root directory web server* untuk mendeteksi serangan siber, peneliti perlu melakukan tinjauan terhadap penetian terdahulu yang digunakan sebagai acuan dan pedoman untuk melakukan penelitian selanjutnya.

Salah satu data pendukung yang perlu di gunakan peneliti adalah penelitian terdahulu yang relevan dengan penelitian yang yang di bahas pada Table 2.1

2.1.1 Penelitian Terdahulu I

Penelitian terdahulu dilakukan oleh A.Dargahi pada tahun (2017) , dalam penetianya tentang " *Analysis of Telegram, An Instant Messaging Service* " dengan menggunakan metode *Neural Network* dan *Decision Tree*. Hasil dari penelitian tersebut didapatkan hasil bahwa *Instagram Messaging* merupakan salah satu plaftrom dari beberapa *plaftrom* penyedia layanan *messanging* yang sukses memeberikan layanan pesan terbaik, yang di lengkapi dengan fitur yang kaya seperti *private chat, grup,chanel,bot.* Dari penelitian tersebut di simpulkan bahwa telegram merupakan platform yang sangat handal dan aman[3].

2.1.2 Penelitian Terdahulu II

Penelitian terdahulu dilakukan oleh Jefree Fahana padatahun (2017), dalam penelitianya tentang " Pemanfaatan Telegram Sebagai Notifikasi Serangan untuk Keperluan Forensik Jaringan " Hasil dari penelitian tersebut menunjukkan aplikasi yang di bangun dengan memanfaatkan *Snort Sensor, IDS (Intrusion Detection System)* dan *telegram* untuk medeteksi serangan pada jaringan berjalan dengan baik dan lancar, Penggunaan aplikasi *SNORT* sebagai aplikasi pendeteksi serangan bekerja dengan baik dan dapat mengirimkan data log *ke* database yang selanjutnya informasi *log* tersebut di teruskan mengunakan aplikasi telegram secara *real time*. Hasil dari rancangan peneliti dapat membantu *administrator* jaringan mengawasi lalu lintas data dan dapat mengumpulkan bukti data serangan untuk keperluan persidangan.

Perbedaan penelitian Jefree Fahana padatahun (2017), dengan peneleitian pemanfaatan *telegram-bot* sebagai *monitoring file* pada root *webserver* untuk mendeteksi serangan siber adalah, peneliti ingin memonitoring perubahan file pada *root directory web server* menggunakan *bot telegram*, di harapkan admin *website* dapat mengetahui perubahan file vital pada *website* yang di kelola secara mudah[2].

2.1.3 Penelitian Terdahulu III

Penelitian terdahulu dilakukan oleh Budi Kurniawan pada tahun (2016), dalam penelitianya tentang " Analisis Pendeteksian dan Pencegahan Serangan *Backdoor* Pada Layanan *Web Server* " Hasil yang di penelitian tersebut cara kerja dan prilaku *backdoor* atau yang biasa di sebut dengan pintu belakang yang sudah di tanam pada suatu website bisa terdeteksi menggunakan bantuan *SNORT IDS* yang sudah di pasang *rule* yang di khususkan untuk *backdoor website*. Dari hasil beberapa percobaan menggunakan *rule backdoor* pada *snort*, *backdoor* yang terenkripsi pada skripnya *SNORT* belum bisa mendeteksinya.

Karena *backdoor* yang ternkripsi skripnya maka peneliti menggunakan aplikasi *backkdor scanner* buatan dari forum kemanan Indonesia yaitu "*DevilCode*" , maka di dapatkan hasil bahwa backkdor yang sebelumnya tidak terdeteksi pada *SNORT* menjadi terdeteksi dan di ketahui di mana lokasi *file backdoor* tersebut diletakkan[4].

Perbedaan penelitian Budi Kurniawan pada tahun (2016), dengan peneleitian pemanfaatan *telegram-bot* sebagai *monitoring file* pada *root webserver* untuk mendeteksi serangan siber adalah, peneliti dapat memberi peringatan dini jika ada pelaku kejahatan siber melakukan aktifitas penanaman backdoor pada root *directory webserver* yang tidak di ketahui oleh admin *website*.

2.1.4 Penelitian Terdahulu IV

Penelitian terdahulu dilakukan oleh Komang Aryasa pada tahun (2014), dalam penelitianya tentang " Implemestasi *Hash Algorithm-1* Untuk Pengaman Data Dalam Library Pada Pemograman Java " Hasil penelitian dan penerapan hashing atau enkripsi menggunakan alogaritma *SHA-1* terdapat perubahan data yang sebelumnya *plain text* berubah menjadi suatu pesan digest yang memiliki Panjang karakter biner yang tetap yaitu 160 bit, Dan jika sebelumnya telah di lakukan hashing atau enkripsi menggunakan SHA-1 dan terdapat perubahan sekecil apapun karakter pada *plain text* yang sudah di hash maka otomatis terjadi perubahan pada hasil *hashing SHA-1*[5].

Perbedaan dengan penelitian Komang Aryasa pada tahun (2014), dengan peneleitian pemanfaatan *telegram-bot* sebagai *monitoring file* pada *directory root webserver* untuk mendeteksi serangan siber adalah,penggunaan tipe hash untuk melakukan enkripsi file, jika peneliti sebelumnya menggunakan enkripsi SHA-1, untuk penelitian yang di lakukan sekarang munggunakan tipe enkripsi SHA512.

No	Judul/Peneliti/Tahun	Metode	Perumusan Masalah	Hasil Penelitian
1	Analysis of Telegram, An Instant Messaging Service / Arash Dargahi Nobari Negar Reshadatmand Mahmood Neshati / 2017	Metode Neural Network & Decision Tree	Mempelajari secara detail cara kerja fitur pada telegram dan menjabarkan secara detail kelebihan dan kekurangan instan massanging telegram	Hasil dari penelitian menujukan telegram menjadi salah satu instan massanger yang cukup kaya fitur seperti, telegram bot, group, chanel dll, kesimpulan dari penelitian juga menewangkan bahwa telegram memiliki performa yang cukup baik dan memiliki penyimpanan yang berbasis cloud cukuk baik dan aman
2	Pemanfaatan Telegram Sebagai Notifikasi Serangan untuk Keperluan Forensik Jaringan / Jefree Fahana Rusydi Umar Faizin Ridho / 2017	Metode Network Forensic	Pemanfaatan telegrambot menjadi notifikasi serangan menggunakan tools snort untuk memantau traffic serangan yang terjadi, agar log yang tertangkap di oleh IDS dapat di olah menjadi data yang bisa di pahami serta bisa menjadi bahan pendukung forensic jika terjadi serangan siber	Hasil dari penelitian menujukkan bahwa penggabungan *IDS SNORT* dapat mendeteksi *traffic* serangan pada jaringan, pada hasil *log* yang di tangkap *IDS* di tangkap dan di salurkan menggunakan *BOT telegram* sebagai notifikasi *realtime,* Log yang di dapat juga di manfaaatkan untuk barang bukti kejahatan siber

3	Analisis Pendeteksian dan Pencegahan Serangan Backdoor Pada Layanan Web Server / Budi Kurniawan Muhamad Akbar Edi Surya Negara / 2016	Metode Experiment	Menerapkan cara pendeteksian *backdoor* yang terdapat pada webserver menggunakan snort	Hasil menujukan penggunaan *SNORT* dapat mendeteksi beberapa jenis *backdoor website*, sedangkan untuk mendeteksi *backdoor* yang terenkripsi perlu di gunakan tools lain yang di buat oleh *forum* kemanan jaringan yaitu "*DevilCode*"
4	Implemestasi Hash Algorithm-1 Untuk Pengaman Data Dalam Library Pada Pemograman Java / Komang Aryasa Yesaya Tommy Paulus / 2014	Metode Secure Hash	Memnerapkan metode enkripsi pada pemrosesan data pada pemograman *java*	Hasil penerapan metode hash pada plain text menujukkan bahwa perubahan sekecil apapun *plain text* sebelum di *hash* dapat merubah hasil *bit* enkripsi

Tabel 2. 1 Penelitian terdahulu

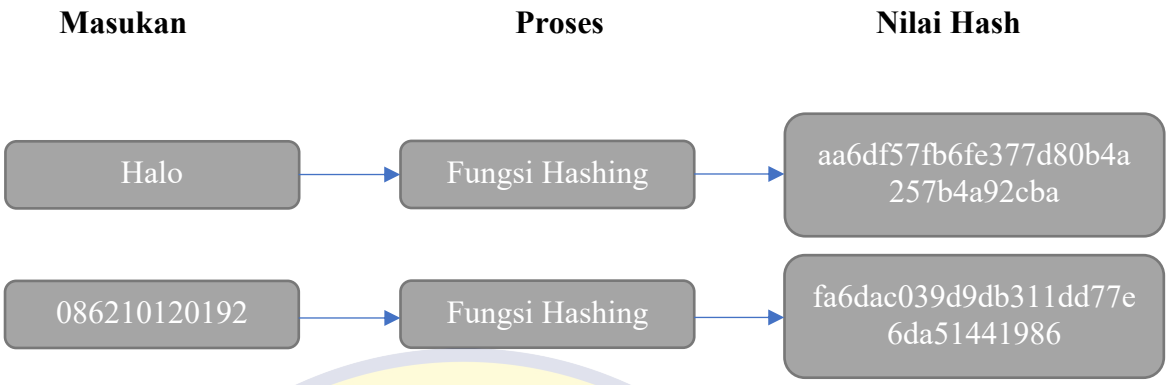
2.2 Teori Dasar Yang Digunakan

2.2.1 Metode Hashing

Sebuah fungsi hash (*hash function* atau *hash algorithm*) adalah suatu cara untuk menghasilkan sebuah digital "*fingerprint"* kecil dari sembarang data [1,2,3]. Fungsi ini memecahkan dan mencampurkan data untuk menghasilkan *fingerprint* yang sering disebut sebagai nilai *hash* (*hash value).* Nilai *hash* ini sering direpresentasikan dengan sebuah *string* pendek dari huruf-huruf dan angka-angka yang kelihatan acak (berbentuk *heksadesimal).* Sebuah fungsi hash yang baik adalah suatu fungsi yang tidak (jarang) memiliki output nilai *hash* yang sama untuk *input* yang berbeda[5]. (Komang Aryasa, 2014)

Fungsi hash :

- menerima masukan string yang panjangnya sembarang,
- lalu mentransformasikannya menjadi string keluaran yang panjangnya tetap (fixed) (umumnya berukuran jauh lebih kecil daripada ukuran string semula).



Persamaan fungsi hash:

h = H(M)

M = pesan kuran sembarang

h = nilai hash atau pesan-ringkas (message-digest)

h <<<< M

Contoh: size(M) = 1 MB size(h) = 128 bit !!!!

Nama lain fungsi hash adalah:
- fungsi kompresi (compression function)
- cetak-jari (fingerprint)
- cryptographic checksum
- message integrity check (MIC)
- manipulation detection code (MDC)

Fungsi Hash Satu-Arah (one-way function) :
- fungsi hash yang bekerja dalam satu arah.
- satu arah: pesan yang sudah diubah menjadi message digest tidak dapat dikembalikan lagi menjadi pesan semula (irreversible).

Sifat-sifat fungsi hash satu-arah adalah sebagai berikut:
1. Fungsi H dapat diterapkan pada blok data berukuran berapa saja.
2. H menghasilkan nilai (h) dengan panjang tetap (fixedlength output).
3. H(x) mudah dihitung untuk setiap nilai x yang diberikan.
4. Untuk setiap h yang dihasilkan, tidak mungkin dikembalikan nilai x sedemikian sehingga H(x) = h. Itulah sebabnya fungsi H dikatakan fungsi hash satu-arah (oneway hash function).
5. Untuk setiap x yang diberikan, tidak mungkin mencari y □
x sedemikian sehingga H(y) = H(x).
6. Tidak mungkin mencari pasangan x dan y sedemikian
sehingga H(x) = H(y).

Masukan fungsi hash adalah blok pesan (M) dan keluaran dari hashing blok pesan sebelumnya,

$$hi = H(Mi , hi - 1)$$

Skema fungsi hash ditunjukkan pada Gambar di bawah:

Gambar Fungsi hash satu-arah

- Fungsi hash satu arah tidak tepat disebut sebagai sebuah proses enkripsi, meskipun nilai hash tidak memiliki makna,
- sebab, nilai hash tidak dapat ditransformasi balik menjadi pesan semula.
- Alasan lainnya, proses hashing tidak menggunakan kunci.

- Ada beberapa fungsi hash satu-arah yang terdapat di dalam kriptografi: - MD2, MD4, MD5, - Secure Hash Function (SHA), - Snefru, - N-hash, - RIPE-MD, dan lain-lain

Fungsi hash satu arah :

- Menjaga integritas data

- Fungsi hash sangat peka terhadap perubahan 1 bit pada pesan

- Pesan berubah 1 bit, nilai hash berubah sangat signifikan.

- Bandingkan nilai hash baru dengan nilai hash lama. Jika sama, pesan masih asli. Jika tidak sama, pesan sudah dimodifikasi.

Contoh:

(i) Pesan (berupa *file*) asli

```
Pada bulan Oktober 2004 ini, suhu udara kota Bandung terasa
lebih panas dari hari-hari biasanya. Menurut laporan Dinas
Meteorologi Kota Bandung, suhu tertinggi kota Bandung
adalah 33 derajat Celcius pada Hari Rabu, 17 Oktober yang
lalu. Suhu terseut sudah menyamai suhu kota Jakarta pada
hari-hari biasa. Menurut Kepala Dinas Meteorologi,
peningkatan suhu tersebut terjadi karena posisi bumi
sekarang ini lebih dekat ke matahari daripada hari-hari
biasa.

Sebutan Bandung sebagai kota sejuk dan dingin mungkin tidak
lama lagi akan tinggal kenangan. Disamping karena faktor
alam, jumlah penduduk yang padat, polusi dari pabrik di
sekita Bandung, asap knalpot kendaraan, ikut menambah
kenaikan suhu udara kota.
```

Nilai MD5: **2F82D0C845121B953D57E4C3C5E91E63**

Gambar hasing file asli

(ii) Misal 33 diubah menjadi 32

```
Pada bulan Oktober 2004 ini, suhu udara kota Bandung terasa
lebih panas dari hari-hari biasanya. Menurut laporan Dinas
Meteorologi Kota Bandung, suhu tertinggi kota Bandung
adalah 32 derajat Celcius pada Hari Rabu, 17 Oktober yang
lalu. Suhu terseut sudah menyamai suhu kota Jakarta pada
hari-hari biasa. Menurut Kepala Dinas Meteorologi,
peningkatan suhu tersebut terjadi karena posisi bumi
sekarang ini lebih dekat ke matahari daripada hari-hari
biasa.

Sebutan Bandung sebagai kota sejuk dan dingin mungkin tidak
lama lagi akan tinggal kenangan. Disamping karena faktor
alam, jumlah penduduk yang padat, polusi dari pabrik di
sekita Bandung, asap knalpot kendaraan, ikut menambah
kenaikan suhu udara kota.
```

Nilai MD5 : **2D1436293FAEAF405C27A151C0491267**

Sebelum diubah : $MD5_1$ = **2F82D0C845121B953D57E4C3C5E91E63**
Sesudah diubah : $MD5_2$ = **2D1436293FAEAF405C27A151C0491267**
Verifikasi: $MD5_1 \neq MD5_2$ (arsip sudah diubah)

Gambar hasing file yang sudah di modifikasi

Di bawah ini merupakan contoh macam macam fungsi hash .

Algoritma	Ukuran *message digest* (bit)	Ukuran blok pesan	Kolisi
MD2	128	128	Ya
MD4	128	512	Hampir
MD5	128	512	Ya
RIPEMD	128	512	Ya
RIPEMD-128/256	128/256	512	Tidak
RIPEMD-160/320	160/320	512	Tidak
SHA-0	160	512	Ya
SHA-1	160	512	Ada cacat
SHA-256/224	256/224	512	Tidak
SHA-512/384	512/384	1024	Tidak
WHIRLPOOL	512	512	Tidak

2.2.2 Kriptografi

Kriptografi adalah bidang ilmu pengetahuan yang mempelajari pemakaian persamaan matematika untuk melakukan proses penyandian data (Onno, 2000). Kriptografi bertujuan untuk mengamankan isi data atau menjaga kerahasiaan informasi dari orang yang tidak berhak untuk mengetahui isi data tersebut. Dengan teknik atau algoritma tertentu yang disebut proses enkripsi (encrypt), data diubah menjadi data sandi yang bentuknya berbeda dengan data aslinya. Orang yang berhak menerima data akan mengetahui algoritma dan memiliki kunci untuk mengembalikan data sandi menjadi bentuk data aslinya, proses ini disebut dekripsi (decrypt). Bentuk data sandi diperlukan pada saat proses penyimpanan atau proses pengiriman data. Untuk dapat melakukan proses enkripsi dan dekripsi maka pihak pengirim dan penerima harus mengetahui algoritma kriptografi yang digunakan serta memiliki kunci yang sesuai. Tingkat keamanan dari data sandi terhadap upaya proses dekripsi secara paksa oleh orang yang tidak berhak ditentukan oleh kekuatan algoritma yang digunakan dan kerahasiaan kunci. Kekuatan algoritma yang digunakan untuk proses enkripsi dan dekripsi berhubungan erat dengan penggunaan persamaan matematika. Semakin banyak dan rumit perhitungan dari persamaan matematika yang digunakan maka data sandi semakin aman (Alfred, 1997).

Pemanfaatan kecepatan dan ketelitian dari kerja komputer sangat membantu untuk proses ini. Kerahasiaan kunci adalah bagaimana cara kunci tersebut

disimpan dan didistribusikan kepada pihak yang berhak menerima data, karena kunci ini akan digunakan untuk melakukan dekripsi. Semakin rapi kunci disimpan dan didistribusikan maka data sandi semakin aman. Berikut ini adalah istilah-istilah yang berhubungan erat dengan kriptografi :

1. Plaintext

Pesan yang hendak dikirimkan (berisi data asli).

2. Ciphertext

Pesan ter-enkrip (tersandi) yang merupakan hasil enkripsi.

3. Enkripsi

Proses pengubahan plaintext menjadi ciphertext.

4. Dekripsi

Merupakan kebalikan dari enkripsi yakni mengubah ciphertext menjadi plaintext, sehingga

berupa data awal/asli.

5. Kunci

Suatu bilangan yang dirahasiakan yang digunakan dalam proses enkripsi dan dekripsi.

2.2.3 Keamanan Website

Pada saat ini *website* menjadi salah satu media informasi modern yang berkembang sangat cepat. Dalam pembuatan website tidak hanya sisi desain dan informasi yang dipentingkan tetapi aspek keamanan dari sebuah website itu sendiri mempunyai peranan yang sangat penting dalam sebuah website.

Kebutuhan keamanan sebuah website timbul dari kebutuhan untuk melindungi data. Pertama, dari kehilangan dan kerusakan data. Kedua, adanya pihak yang tidak hendak mengakses dan merubah data. Permasalahan lainnya mencakup perlindungan data dari *delay* yang berlebih pada saat mengakses atau menggunakan data, atau mengatasi gangguan *Denial of Service*.

Menerapkan keamanan pada *website* dilakukan pada pengujian menggunakan tool berupa perangkat lunak dan cara-cara tertentu yang digunakan untuk menguji keamanan sebuah website. Hasil dari pengujian adalah ditemukannya berbagai level kerentanan dari level kerentanan Low sampai level kerentanan High. (Detty Metasari, 2014)

2.2.4 Keamanan Data

Keamanan data adalah hal yang sangat penting untuk dipertimbangkan pada setiap kegiatan yang berhubungan dengan data rahasia atau terbatas pada

komunitas tertentu. Data yang berkaitan dengan informasi sensitif dan berharga akan beresiko jika diakses oleh orang yang tidak berhak. Dalam dunia perbankan banyak data pelanggan yang harus dilindungi dan hatihati mempertimbangkan faktor keamanan seperti ebanking, sms banking, internet banking, dll Salah satu cara untuk meningkatkan keamanan data dengan kriptografi. Teknik kriptografi ini digunakan untuk melakukan enkripsi dan dekripsi data, mengkonversi atau mengubah data menjadi kode kode tertentu. Hal ini dilakukan agar informasi yang tersimpan dan ditransmisikan melalui jaringan yang paling aman, misalnya melalui Internet. Teknik ini lebih aman karena tidak dapat dibaca oleh siapa pun kecuali oleh mereka yang berhak. Penelitian ini bertujuan untuk meningkatkan keamanan data dengan metode enkripsi keamanan Tanam Pemilihan Pseudorandom. Keuntungan dari teknik enkripsi ini yang menggunakan algoritma enkripsi sangat ringan namun aman dalam arti bahwa hasil enkripsi dapat menyembunyikan data asli menjadi bentuk yang sulit diterjemahkan. Hal lain yang membuat ini metode yang sangat aman dari enkripsi adalah proses algoritma acak atau random sehingga menjadi sulit untuk memprediksi dan dibongkar. Hasil yang dicapai dengan algoritma ini diperoleh akurasi di atas 97% untuk kembali ke bentuk awal. (Pratiwi, 2016)

2.2.5 Backdoor

Backdoor adalah perangkat lunak yang digunakan untuk mengakses sistem, aplikasi, atau jaringan tanpa harus menangani proses autentikasi. Backdoor dapat membantu user yang membuat backdoor (peretas) dapat masuk ke dalam sistem tanpa harus melewati proses autentifikasi. Backdoor juga dapat diartikan sebagai mekanisme yang digunakan untuk mengakses sistem atau jaringan.

Awalnya backdoor dibuat para programmer untuk mendapatkan akses khusus untuk masuk ke dalam program yang mereka kembangkan. Apalagi saat terjadi masalah pada program mereka, seperti crash maupun masalah yang diakibatkan oleh bug. Pada saat terjadi masalah itu, backdoor menjadi salah satu solusi yang berhasil dibuat.

Namun, sejalan dengan perkembangan teknologi informasi, backdoor yang sebelumnya menjadi solusi, saat ini menjadi salah satu celah yang digunakan mengambil akses sistem secara paksa. Backdoor disisipkan ke dalam kode sistem maupun sebuah program secara diam-diam sehingga pengguna tidak mengetahui ada backdoor pada sistemnya. Akibatnya, pembuat backdoor tadi dapat masuk dan mendapatkan akses ke dalam sistem pengguna bahkan dapat mengakses keseluruhan sistem. (Yasin K, 2017, www.niagahoster.co.id

2.2.6 Monitoring

Keberhasilan sebuah program dapat dilihat dari apa yang direncanakan dengan apa yang dilakukan, apakah hasil yang diperoleh berkesesuaian dengan hasil perencanaan yang dilakukan. Untuk dapat memperoleh implementasi sebuah acara yang sesuai dengan apa yang direncanakan manajemen harus menyiapkan

sebuah program yaitu monitoring, monitoring ditujukan untuk memperoleh fakta, data dan informasi tentang pelaksanaan program, apakah proses pelaksanaan kegiatan dilakukan seusai dengan apa yang telah direncakan. Selanjutnya temuan-temuan hasil monitoring adalah informasi untuk proses evaluasi sehingga hasilnya apakah program yang ditetapkan dan dilaksanakan memperoleh hasil yang berkesuaian atau tidak. Beberapa pakar manajemen mengemukakan bahwa fungsi monitoring mempunyai nilai yang sama bobotnya dengan fungsi perencanaan. Conor (1974) menjelaskan bahwa keberhasilan dalam mencapai tujuan, separuhnya ditentukan oleh rencana yang telah ditetapkan dan setengahnya lagi fungsi oleh pengawasan atau monitoring. Pada umumnya, manajemen menekankan terhadap pentingnya kedua fungsi ini, yaitu perencanaan dan pengawasan.

Proses dasar dalam monitoring ini meliputi tiga tahap yaitu: (1) menetapkan standar pelaksanaan; (2) pengukuran pelaksanaan; (3) menentukan kesenjangan (deviasi) antara pelaksanaan dengan standar dan rencana. Menurut Dunn (1981), monitoring mempunya empat fungsi, yaitu:

a. Ketaatan (compliance). Monitoring menentukan apakah tindakan administrator, staf, dan semua yang terlibat mengikuti standar dan prosedur yang telah ditetapkan.
b. Pemeriksaan (auditing). Monitoring menetapkan apakah sumber dan layanan yang diperuntukkan bagi pihak tertentu bagi pihak tertentu (target) telah mencapai mereka.
c. Laporan (accounting). Monitoring menghasilkan informasi yang membantu "menghitung" hasil perubahan sosial dan masyarakat sebagai akibat implementasi kebijaksanaan sesudah periode waktu tertentu.
d. Penjelasan (explanation). Monitoring menghasilkan informasi yang membantu menjelaskan bagaimana akibat kebijaksanaan dan mengapa antara perencanaan dan pelaksanaannya tidak cocok.

2.2.7 Telegram

Telegram adalah sebuah aplikasi layanan pengirim pesan instan multiplatform berbasis awan yang bersifat gratis dan nirlaba. Klien Telegram tersedia untuk perangkat telepon seluler (Android, iOS, Windows Phone, Ubuntu Touch) dan sistem perangkat komputer (Windows, OS X, Linux). Para

pengguna dapat mengirim pesan dan bertukar foto, video, stiker, audio, dan tipe berkas lainnya. Telegram juga menyediakan pengiriman pesan ujung ke ujung terenkripsi opsional.

Telegram dikembangkan oleh Telegram Messenger LLP dan didukung oleh wirausahawan Rusia Pavel Durov. Kode pihak kliennya berupa perangkat lunak sistem terbuka namun mengandung blob binari, dan kode sumber untuk versi terbaru tidak selalu segera dipublikasikan, sedangkan kode sisi servernya bersumber tertutup dan berpaten. Layanan ini juga menyediakan API kepada pengembang independen. Pada Februari 2016, Telegram menyatakan bahwa mereka memiliki 100 juta pengguna aktif bulanan, mengirimkan 15 miliar pesan per hari.

Keamanan Telegram telah menghadapi pemeriksaan teliti yang menjadi perhatian; para kritikus mengklaim bahwa model keamanan Telegram dirusak oleh penggunaan protokol enkripsi yang dirancang khusus yang belum terbukti andal dan aman, dan dengan tidak mengaktifkan percakapan aman secara default. Telegram juga menghadapi kritik karena penggunaan skala luas oleh organisasi teroris Negara Islam (NIIS). NIIS telah merekomendasikan Telegram kepada para pendukung dan anggotanya dan pada Oktober 2015 mereka mampu melipatgandakan jumlah pengikut saluran resmi mereka menjadi 9.000 orang. (https://id.wikipedia.org, 2013-2018)

2.2.8 Telegram API (BOT)

Telegram menyediakan 2 bentuk API, API yang pertama adalah klien IM Telegram, yang berarti semua orang dapat menjadi pengembang klien IM Telegram jika diinginkan. Ini berarti jika seseorang ingin mengembangkan Telegram versi mereka sendirimereka tidak harus memulai semua dari awal lagi. Telegram menyediakan source code yang mereka gunakan saat ini. Tipe API yang kedua adalah Telegram Bot API. API jenis kedua ini memungkinkan siapa saja untuk membuat bot yang akan membalas semua penggunanya jika mengirimkan pesan perintah yang dapat diterima oleh Bot tersebut. Layanan ini masih hanya tersedia bagi pengguna yang menggunakan aplikasi Telegram saja. Sehingga pengguna yang ingin menggunakan Bot harus terlebih dahulu memiliki akun Telegram. Bot juga dapat dikembangkan oleh siapa saja.

Metode Pengiriman yang Disediakan oleh Telegram Bot API Ada beberapa metode yang dapat digunakan untuk merancang sebuah Bot di Telegram.

Beberapa diantaranya adalah:

- sendMessage
- forwardMessage
- sendPhoto
- sendAudio
- sendDocument
- sendSticker
- sendVideo
- sendVoice
- sendLocation
- sendVenue

- sendContact
- sendChatAction
- getUserProfilePhotos
- getFile
- kickChatMember
- leaveChat
- unbanChatMember
- getChat
- getChatAdministrator
- getChatMember

Bot juga dapat menggunakan custom keyboard untuk penggunanya. Hal ini akan mempermudah interaksi antara botdan penggunanya. Semua dasar pengiriman data yang digunakan oleh server Telegram akan menggunakan JSON, sehingga pengembang bot harus juga menggunakan bentuk data JSON. Bot Telegram tidak terbatas oleh bahasa pemrograman. Hampir semua bahasa pemrograman bisa digunakan untuk merancang suatu bot. Telegram juga menyediakan contoh bot yang menggunakan berbagai bahasa pemrograman.

2.2.9 Enkripsi dan Dekripsi

Enkripsi yaitu suatu proses pengaman suatu data yang disembunyikan atau proses konversi data (plaintext) menjadi bentuk yang tidak dapat dibaca/ dimengerti. Enkripsi telah digunakan untuk mengamankan komunikasi di berbagai negara, namun, hanya organisasi-organisasi tertentu dan individu yang memiliki kepentingan yang sangat mendesak akan kerahasiaan yang menggunakan enkripsi. Di pertengahan tahun 1970an enkripsi kuat dimanfaatkan untuk pengamanan oleh sekretariat agen pemerintah Amerika

Serikat pada domain publik, dan saat ini enkripsi telah digunakan pada sistem secara luas, seperti Internet, e-commerce, jaringan telepon bergerak dan ATM pada bank. Enkripsi dapat digunakan untuk tujuan keamanan, tetapi teknik lain masih diperlukan untuk membuat komunikasi yang aman, terutama untuk memastikan integrasi dan autentikasi dari sebuah pesan. Untuk menampilkan enkripsi dan kebalikannya dekripsi, digunakan algoritma yang biasa disebut Cipher dengan menggunakan metode serangkaian langkah yang terdefinisi yang diikuti sebagai prosedur. Alternatif lain ialah Encipherment. Informasi yang asli disebuh sebagai plaintext, dan bentuk yang sudah dienkripsi disebut sebagai chiphertext. Pesan chipertext berisi seluruh informasi dari pesan plaintext, tetapi tidak dalam format yang didapat dibaca manusia ataupun komputer tanpa menggunakan mekasnisme yang tepat untuk melakukan dekripsi. (Gadih, 2013)

2.2.10 Python

Python adalah bahasa pemrograman interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. Python diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif.

Python mendukung multi paradigma pemrograman, utamanya; namun tidak dibatasi; pada pemrograman berorientasi objek, pemrograman imperatif, dan

pemrograman fungsional. Salah satu fitur yang tersedia pada python adalah sebagai bahasa pemrograman dinamis yang dilengkapi dengan manajemen memori otomatis. Seperti halnya pada bahasa pemrograman dinamis lainnya, python umumnya digunakan sebagai bahasa skrip meski pada praktiknya penggunaan bahasa ini lebih luas mencakup konteks pemanfaatan yang umumnya tidak dilakukan dengan menggunakan bahasa skrip. Python dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai platform sistem operasi.

Saat ini kode python dapat dijalankan di berbagai platform sistem operasi, beberapa diantaranya dalah:

- Linux/Unix
- Windows
- Mac OS X
- Java Virtual Machine
- OS/2
- Amiga

Python didistribusikan dengan beberapa lisensi yang berbeda dari beberapa versi. Lihat sejarahnya di Python Copyright. Namun pada prinsipnya Python dapat diperoleh dan dipergunakan secara bebas, bahkan untuk kepentingan komersial. Lisensi Python tidak bertentangan baik menurut definisi Open Source maupun General Public License (GPL)

Python dikembangkan oleh Guido van Rossum pada tahun 1990 di CWI, Amsterdam sebagai kelanjutan dari bahasa pemrograman ABC. Versi terakhir yang dikeluarkan CWI adalah 1.2.

Tahun 1995, Guido pindah ke CNRI sambil terus melanjutkan pengembangan Python. Versi terakhir yang dikeluarkan adalah 1.6. Tahun 2000, Guido dan para pengembang inti Python pindah ke BeOpen.com yang merupakan sebuah perusahaan komersial dan membentuk BeOpen PythonLabs. Python 2.0 dikeluarkan oleh BeOpen. Setelah mengeluarkan Python 2.0, Guido dan beberapa anggota tim PythonLabs pindah ke DigitalCreations.

Saat ini pengembangan Python terus dilakukan oleh sekumpulan pemrogram yang dikoordinir Guido dan Python Software Foundation. Python Software Foundation adalah sebuah organisasi non-profit yang dibentuk sebagai pemegang hak cipta intelektual Python sejak versi 2.1 dan dengan demikian mencegah Python dimiliki oleh perusahaan komersial. Saat ini distribusi Python sudah mencapai versi 2.6.1 dan versi 3.0.

Nama Python dipilih oleh Guido sebagai nama bahasa ciptaannya karena kecintaan Guido pada acara televisi Monty Python's Flying Circus. Oleh karena itu seringkali ungkapan-ungkapan khas dari acara tersebut seringkali muncul dalam korespondensi antar pengguna Python. (https://id.wikipedia.com)

2.2.11 PHP

PHP adalah bahasa pemrograman script yang paling banyak dipakai saat ini. PHP banyak dipakai untuk memrogram situs web dinamis, walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain. Contoh terkenal dari aplikasi PHP adalah forum (phpBB) dan MediaWiki (software di belakang Wikipedia). PHP juga dapat dilihat sebagai pilihan lain dari ASP.NET/C#/VB.NET Microsoft, ColdFusion Macromedia, JSP/Java Sun Microsystems, dan CGI/Perl. Contoh aplikasi lain yang lebih kompleks berupa CMS yang dibangun menggunakan PHP adalah Mambo, Joomla!, Postnuke, Xaraya, dan lainlain.

Sejarah PHP

Pada awalnya PHP merupakan kependekan dari Personal Home Page (Situs Personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama FI (Form Interpreted), yang wujudnya berupa sekumpulan script yang digunakan untuk mengolah data form dari web. Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilisan kode sumber ini menjadi open source, maka banyak programmer yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini interpreter PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP. (Adis Lena, 2014)

2.2.12 Lampp

LAMP adalah istilah yang merupakan singkatan dari Linux, Apache, MySQL dan Perl/PHP/Phyton. Merupakan sebuah paket perangkat lunak bebas yang digunakan untuk menjalankan sebuah aplikasi secara lengkap.

1. Komponen-komponen dari LAMP:
2. Linux sebagai sistem operasi
3. Apache HTTP Server sebagai *web server*
4. MySQL sebagai sistem basis data
5. Perl atau PHP atau Pyton sebagai bahasa pemrograman yang dipakai

Beberapa perangkat lunak yang menggunakan konfigurasi LAMP antara lain MediaWiki dan Bugzilla.

LAMP adalah singkatan untuk beberapa, perangkat lunak *open source*, awalnya diciptakan dari huruf pertama dari Linux (sistem operasi), Apache HTTP *Server*, MySQL (*software database*) dan Perl / PHP Python /, komponen utama untuk membangun layak *web server* tujuan umum.

Kombinasi yang tepat dari perangkat lunak yang disertakan dalam paket LAMP mungkin berbeda, terutama berkenaan dengan perangkat lunak

web scripting, seperti PHP dapat diganti atau dilengkapi dengan Perl dan / atau Python. istilah serupa ada untuk dasarnya suite perangkat lunak yang sama (AMP) yang berjalan pada sistem operasi lain, seperti Microsoft Windows (WAMP), Mac OS (MAMP), Solaris (Samp), atau OpenBSD (OAMP). Meskipun penulis asli program ini tidak merancang mereka semua untuk bekerja secara khusus satu sama lain, filosofi pengembangan dan set alat dibagi dan dikembangkan bersama dekat. Kombinasi perangkat lunak telah menjadi populer karena bebas biaya, sumber terbuka, dan karena itu mudah beradaptasi, dan karena di mana-mana komponen yang dibundel dengan kebanyakan distribusi Linux saat ini. (https://kangoby.wordpress.com, 2012)