SEMINAR NASIONAL TEKNOKA

Vol. 2, 2017
ISSN No. 2502-8782

Membangun Sistem Mobile Monitoring Keamanan Web Aplikasi Menggunakan Suricata dan Bot Telegram Channel

Dias Utomo, Muchammad Sholeh, Arry Avorizano

Program Studi Teknik Informatika Fakultas Teknik
Universitas Muhammadiyah Prof. Dr. HAMKA
Jl. Tanah Merdeka No. 6, Kp. Rambutan, Ps. Rebo, Jakarta Timur
Tlp. 021-8400941, 021-87782739, 87783818
Email: m.sholeh@uhamka.ac.id

Abstrak – *Keamanan suatu informasi sangatlah penting, terlebih lagi pada suatu jaringan yang terkoneksi dengan internet. Keamanan jaringan seringkali terganggu dengan adanya ancaman dari dalam ataupun luar. Serangan tersebut dapat berupa serangan yang bermaksud merusak jaringan ataupun mencuri informasi penting yang ada pada jaringan tersebut sehingga monitoring jaringan sangat di perlukan yang mampu bekerja secara realtime. Menjadi suatu tuntutan bagi sistem administrator dalam melakukan pengawasan secara terus menerus untuk keamanan jaringan yang menjadi titik masalah penting. Penulis akan membahas tentang membangun sistem monitoring keamanan web aplikasi pada PC server. Lalu lintas jaringan komputer di pantau dengan sebuah aplikasi pendeteksi serangan yaitu Suricata, yakni aplikasi berbasis opensource yang mendeteksi aktifitas mencurigakan ke dalam PC server. Sehingga bentuk ancaman atau serangan yang masuk akan dibuat batasan berdasarkan klasifikasi pada Suricata. Peringatan bahaya dikirim berupa pesan sebagai informasi adanya ancaman atau serangan dan di integrasikan pada aplikasi telegram yang terdapat pada smartphone dalam berbentuk log alert kejadian yang berisikan waktu, port server, attacker port dan jenis serangan. Dengan terapkannya sistem monitoring yang dilengkapi peringatan ancaman ini, pengawasan terhadap ancaman yang menyusup ke jaringan lebih maksimal, karena terintegrasinya antara sistem yang langsung terhubung dengan sistem administrator.*

***Kata kunci**: Monitoring, opensource, Suricata, Aplikasi Telegram*

1 Pendahuluan

Keamanan jaringan komputer merupakan suatu hal yang sangat penting dilakukan sebagai pencegah penyalahgunaan sumber daya jaringan yang tidak sah, mengantisipasi resiko ancaman baik secara langsung ataupun tidak langsung. Demi menjamin terjaganya suatu lalu lintas jaringan komputer maka perlu adanya suatu *software* jaringan yang berfungsi sebagai pemantau aktifitas jaringan.

Pencegahan yang paling sering dilakukan terhadap serangan jaringan adalah dengan menempatkan seorang administrator, namun masalah akan timbul ketika administrator sedang tidak mengawasi jaringan, maka untuk mengatasi permasalahan tersebut diletakkanlah suatu sistem yaitu IDS *(Intrusion Detection System)* yang dapat mendeteksi aktifitas yang mencurigakan didalam jaringan dengan cara mengautomatisasikan fungsi kerja dari seorang administrator.

Berawal dari munculnya perangkat lunak bekerja sebagai aplikasi untuk monitoring jaringan yang bebas digunakan dan di modifikasi sesuai kebutuhan. Saat ini telah muncul aplikasi baru perkembangan dari aplikasi berbasis IDS, dalam pengamatan penulis, Snort paling banyak digunakan karena merupakan standar bagi IDS di dunia, namun alternatif lain yakni Suricata sebagai salah satu IDS *engine open source* masih belum banyak digunakan.

Aplikasi ini bekerja memberikan peringatan dini pada saat terjadi ancaman terhadap target atau terhadap *host* yang dilindungi oleh aplikasi tersebut. Peringatan dini dikirim melalui aplikasi *Telegram messenger* yang berupa *chat* di kirim ke *Smartphone*. Sebagai referensi sebelumnya telah di teliti pula :

1. "Sistem Monitoring Keamanan Jaringan Komputer Dengan Pemanfaatan SMS Alert" oleh Nisa Aulia Rahman pada tahun 2015 berupa jurnal skripsi.

Manuscript received 20 Sepetember 2017, revised 29 September 2017

2. "Analisis Dan Implementasi Suricata, Snorby, Dan Barnyard2 Pada VPS Ubuntu" oleh Alim Nuryanto pada tahun 2015 berupa jurnal skripsi.

Dengan ini maka penulis mencoba membahas suatu masalah dengan judul "Membangun Sistem Mobile Monitoring Keamanan Web Aplikasi Menggunakan Suricata dan Bot Telegram Channel".

2 Dasar Teori

2.1. Ancaman Jaringan Komputer

Vulnerability adalah segala sesuatu yang berjalan pada komputer yang dapat secara langsung atau tidak langsung memicu kebocoran kerahasiaan, integritas, ketersediaan informasi atau layanan di manapun pada jaringan.[10]

Secara umum tipe *vulnerability* meliputi hal-ha berikut *Buffer Overflows, Kesalahan format string, Malicius Content, Web Application.*

a. Scanning

Scanning merupakan tanda dimulainya serangan *hacker*. Melalui scanning, hacker akan mencari berbagai kemungkinan yang bisa digunakan untuk mengambil alih komputer korban. Melalui informasi yang diperoleh pada tahapan *scanning*, *hacker* bisa mencari jalan masuk untuk menguasai komputer korban. Berbagai *tool* biasanya digunakan oleh *hacker* dalam membantu proses pencarian ini.11]

b. Web Vulnerability Scanning

Vulnerability scanning merupakan *scanning* yang bertujuan menemukan kelemahan dari sebuah sistem. Beberapa *software vulnerability scanning* yang banyak di gunakan antara lain, Saint, Nessus, Nmap, Nikto, SQLmap, Scrawlr (*SQL Injection Scanner*), WPScan, Wikto dan lain-lain. [11]

c. Reconnaissance

Reconnaissance adalah tahap mengumpulkan data dimana *hacker* akan mengumpulkan semua data sebanyak-banyaknya mengenai target. *Reconnaissance* masih dibagi lagi menjadi dua, yaitu : [11]

1. *Active reconnaissance*
   Reconnaissance yang dilakukan secara aktif, dimana hacker melakukan aktifitas terhadap korban untuk mendapatkan data tersebut.
2. *Pasive reconnaissance*
   Reconnaissance yang tanpa berhubungan secara langsung dengan korban, anda tidak akan terdeteksi oleh korban.

2.2. Monitoring

Adapun pengertian monitoring menurut para ahli :

Cassely dan Kumar

Monitoring merupakan program yang terintegrasi, bagian penting dipraktek manajemen yang baik dan arena itu merupakan bagian integral di manajemen sehari-hari [7].

Calyton dan Petry

Monitoring sebagai suatu proses mengukur, mencatat, mengumpulkan, memproses dan mengkomunikasikan informasi untuk membantu pengambilan keputusan manajemen program/proyek [8].

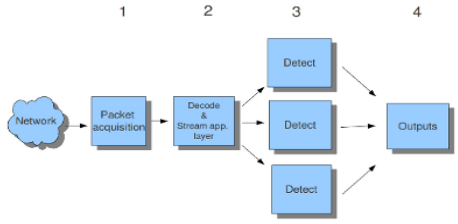
2.3. Intrusion Detection System (IDS)

Intrusion Detected System merupakan sebuah sistem yang melakukan pengawasan terhadap *traffic* jaringan dan pengawasan terhadap kegiatan-kegiatan didalam sebuah sistem jaringan. Jika ditemukan kegiatan-kegiatan yang mencurigakan berhubungan dengan *traffic* jaringan maka IDS akan memberikan peringatan kepada sistem atau administrator jaringan. Dalam banyak kasus IDS juga merespon terhadap *traffic* yang tidak normal / anomali pemblokiran seorang *user* atau alamat IP (*Internet Protocol*) sumber dari usaha pengaksesan jaringan.[9]

2.4. Suricata

Suricata merupakan *Network* IDS, IPS dan sebuah mesin monitor keamanan jaringan dengan peforma tinggi. Suricata adalah IDS *opensource* dan dimiliki oleh sebuah komunitas non-profit, yaitu *Open Information Security Foundation* (OISF). Suricata di kembangkan oleh OISF dan vendor pendukungnya. Suricata *engine* merupakan *open source next generation intrusion detection and prevention engine*. Suricata merupakan engine yang memiliki kemampuan *Multi-threaded*. [14]

2.4.1. Alur Kerja Suricata



Gambar 1 Alur Kerja Suricata

Packet acquisition : Membaca paket – paket yang masuk
Decode : Men-Decode paket
Stream App.Layer : Melakukan *stream-tracking* dan *reassembly*
Detect : Melakukan pencocokan *Signatures* dengan database
Outputs : Memproses semua kejadian dan memberikan peringatan. [14]

2.5 Aplikasi Telegram Messenger

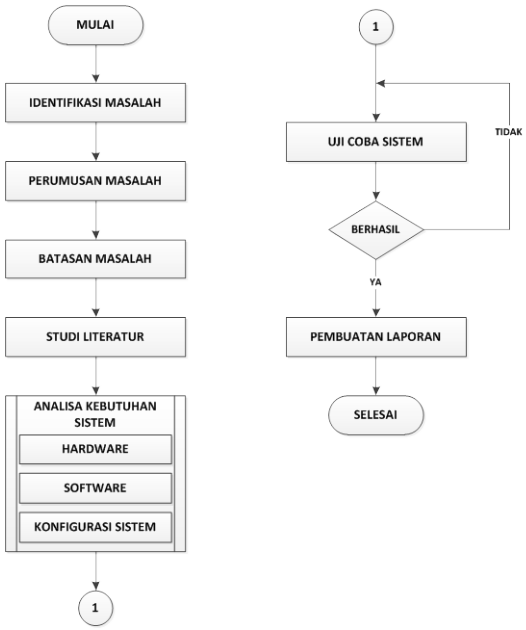
Telegram adalah sebuah sistem perpesanan yang lintas *platform* dan berpusat pada keamanan kerahasiaan pribadi penggunanya, sedangkan bot adalah program komputer yang melakukan pekerjaan tertentu secara otomatis. Bot adalah

ISSN No. 2502-8782

sebuah mesin, dibuat memudahkan kehidupan keseharian kita tanpa harus terpaku di depan komputer. Jika ingin membuat bot telegram, ia perlu komunikasi utama dengan peladen (server) telegram dilakukan melalui protokol MTProto, sebuah protokol biner buatan telegram sendiri.

Bot yang paling terkenal adalah telegram-bot buatan Yugo Perez. Bot-telegram cli bekerja layaknya akun pribadi dan manfaat bot ini diamini juga oleh telegram yang kemudian meluncurkan bot API (*Application Progamming Interface*) agar orang banyak dapat membangun bot menggunakan bahasa pemrograman yang mereka kuasai tanpa harus berhubungan dengan telegram-cli atau MTProto.[19]

3 Metodologi Penelitian



Gambar 2 Metode Membangun

4 Analisis dan Membangun Sistem

4.1. Analisa Sistem

1x24 jam pengawasan jaringan menjadi keharusan bagi sistem administrator jaringan dalam pemantauan ancaman yang akan masuk. Namun ancaman jaringan komputer tidak dapat di ketahui kapan akan terjadinya, oleh sebab itu perlu adanya sebuah sitem sebagai notifikasi ancaman yang langsung terhubung pada sistem administrator.

Penelitian ini difokuskan kepada penerapan *alert* pada aplikasi *smartphone* sebagai notifikasi dalam pemantauan keamanan *web* aplikasi. Sistem yang dibangun ini menggunakan suricata sebagai pendeteksi aktivitas yang mencurigakan dalam sebuah server yang terdapat *web* di dalamnya. Bagaimana pesan singkat yang masuk berisikan pemberitahuan bahwa adanya ancaman yang ditujukan pada sistem administrator.

4.1.1. Kebutuhan Perangkat Keras

Perancangan perangkat keras di sini hanya meliputi *Smartphone* dan PC Server yaitu :

a. Spesifikasi minimum PC Server
- *Processor* : Intel core i3
- RAM : 4 Giga Byte
- *Hard Disk* : Minimal 80 Giga Byte

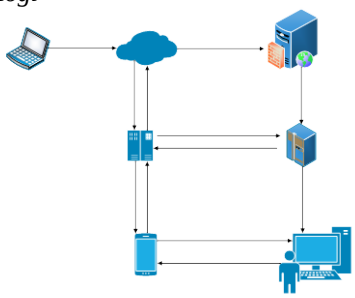
b. Spesifikasi minimum *smartphone*
- *Operating system* : Android 4.1.2 versi Jellybean
- *Processor* : Dual-core 1.2 Ghz Cortex A5
- RAM : 512Mega Byte

4.1.2. Kebutuhan Perangkat Lunak

Untuk menerapkan skema yang akan dirancang dibutuhkan beberapa perangkat lunak sebagian besar perangkat yang digunakan berbasis *Open Source*.

1) VirtualBox
2) Ubuntu 17.04LTS
3) *Wordpress*
4) *Suricata*
5) *Telegram*
6) *Java*
7) *Putty*
8) *Filezilla*

4.2. Topologi



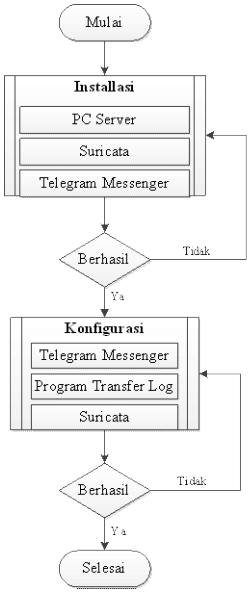
Gambar 3 Topologi Sesudah Perancangan

Setelah menganalisa bentuk dan karakteristik jaringan sebelumnya maka penulis mulai merancang topologi jaringan dengan mengintegrasikan *alert* ke *Telegram Messenger*, yang diharapkan dapat mempermudah sistem administrator dalam memonitoring keamanan *web* aplikasi pada saat tidak berada pada ruang monitoring. dimana sistem administrator dapat melakukan monitoring dengan dua acara yaitu:

1. Sistem administrator dapat memonitoring ancaman secara langsung pada PC server saat berada di ruang monitoring.
2. Sistem administrator jaringan dapat memonitoring ancaman melalui smartphone pada saat administrator tidak berada di tempat monitoring jaringan.

ISSN No. 2502-8782

4.3. Alur Membangun Sistem



Gambar 4 Alur Membangun Sistem

Sub bab ini menjelaskan tahapan-tahapan dalam pembuatan sistem monitoring keamanan jaringan mulai dari pemasangan sistem operasi yang digunakan pada PC server lalu instalasi aplikasi pendukung. Pada perancangan sistem ini server menggunakan sistem operasi UBUNTU 17.04 LTS, Setelah UBUNTU 17.04 LTS sudah terinstall pada server maka dilanjutkan dengan instalasi wordpress, suricata dan telegram pada *smartphone*, konfigurasi sistem dan pembuatan aplikasi *transfer log* pada telegram.

5 Hasil dan Pembahasan

Pada bab V Hasil dan Pembahasan ini akan menjelaskan tentang uji sistem mobile monitoring keamanan yang akan dilakukan oleh PC *Attacker*, pengujian sistem deteksi dengan *tools* suricata, uji serang pada sistem yang sudah di bangun di Bab IV Analisa dan Membangun Sistem.

5.1. Pengujian Serangan Ke Server Web Vulnerability

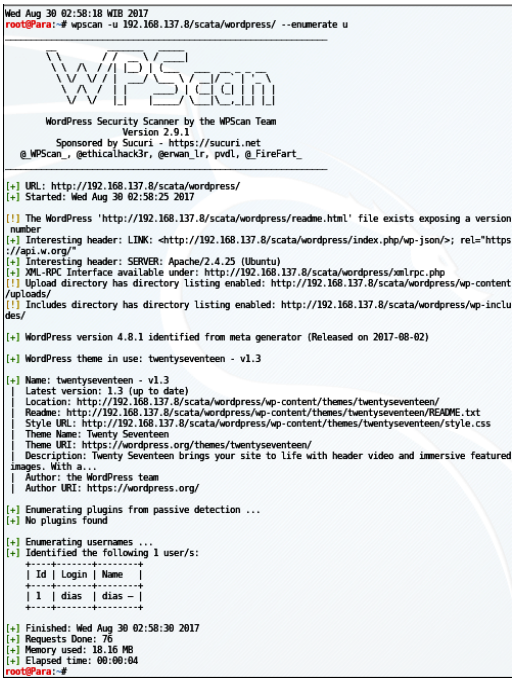
Pengujian serangan terhadap *Web Vulnerability* PC server yang di install suricata dilakukan dengan beberapa *tools* untuk melakukan serangan pada *web* aplikasi yang dilakukan oleh PC *Attacker*. Untuk memastikan bahwa sistem monitoring yang di bangun berjalan.

5.1.1 Menggunakan WPScan

WPScan merupakan *tools vulnerability scanner* untuk CMS wordpress WPScan mampu mendeteksi kerentanan umum serta daftar semua *plugin* dan *themes* yang digunakan oleh sebuah *website* yang menggunakan CMS wordpress yang sudah di bangun di PC server

Gambar 6 Wordpress PC Server

Pada gambar 6 wordpress pada PC server yang menjadi target pengujian serangan *web vulnerability*.



Gambar 7 WPScan *Scanning* Wordpress

Masuk terminal WPScan dengan menjalankan perintah :

```
wpscan –u
192.168.137.8/scata/wordpress/ --
enumerate u
```

-u adalah URL target

--enumerate u adalah untuk *scan username*.

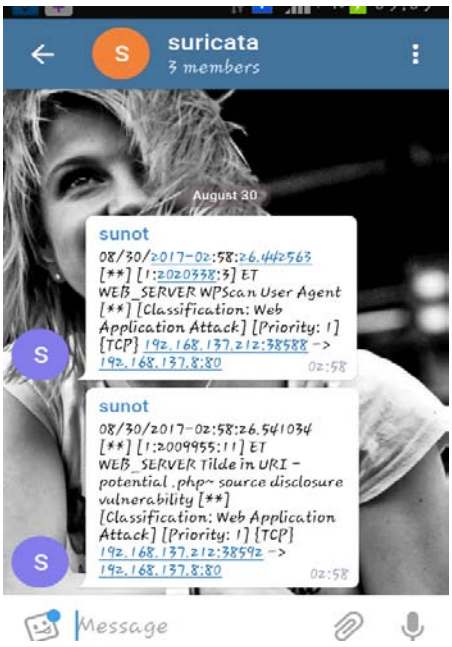
WPScan berhasil mendeteksi *username* admin pada

ISSN No. 2502-8782

wordpress PC server dengan login menggunakan nama dias. Dan suricata berhasil mendeteksi *scanning attacker*. *Log* deteksi suricata dari hasil serangan WPScan.

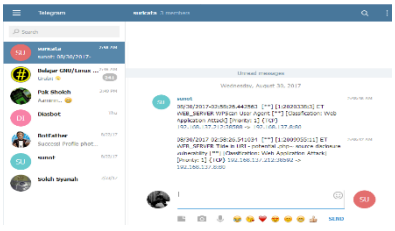
```
08/30/2017-02:58:26.442563  [**] [1:2020338:3] ET WEB_SERVER W
PScan User Agent [**] [Classification: Web Application Attack]
 [Priority: 1] {TCP} 192.168.137.212:38588 -> 192.168.137.8:80
08/30/2017-02:58:26.541034  [**] [1:2009955:11] ET WEB_SERVER
Tilde in URI - potential .php~ source disclosure vulnerability
 [**] [Classification: Web Application Attack] [Priority: 1] {
TCP} 192.168.137.212:38592 -> 192.168.137.8:80
```

Gambar 8 Log Suricata Deteksi WPScan



Gambar 9 WPScan Notifikasi *Telegram Messenger*

Tampilan notifikasi yang masuk pada telegram saat pengujian menggunakan WPScan. Notifikasi saat terjadinya ancaman juga dapat masuk saat meggunakan *Web* telegram *messenger* maupun aplikasi telegram *messenger* pada *smartphone*.berikut tampilan pada *web* telegram.



Gambar 10 Tampilan *Web Telegram* Deteksi WPScan

5.1.2 Menggunakan Nikto

Nikto adalah alat *scanning* aplikasi *web* yang mencari kesalahan konfigurasi, direktori *web* diakses secara terbuka dan sejumlah kerentanan aplikasi *web*. Dalam pengujian kali ini *tools* nikto sudah terinstall di Kali Linux. Target pengujian nikto terhadap *web* PC server dengan *IP address* 192.168.137.8.

Lalu jalankan nikto dengan perintah –h untuk mencari *hostname* pada target.

```
Wed Aug 30 03:06:15 WIB 2017
root@Para:~# nikto -h 192.168.137.8
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          192.168.137.8
+ Target Hostname:    192.168.137.8
+ Target Port:        80
+ Start Time:         2017-08-30 03:06:31 (GMT7)
---------------------------------------------------------------------------
+ Server: Apache/2.4.25 (Ubuntu)
+ Server leaks inodes via ETags, header found with file /, fields:
 0x2aa6 0x555dd75ab3ab5
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint
 to the user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow t
he user agent to render the content of the site in a different fas
hion to the MIME type
+ No CGI Directories found (use '-C all' to force check all possib
le dirs)
+ Allowed HTTP Methods: HEAD, GET, POST, OPTIONS
+ Uncommon header 'x-robots-tag' found, with contents: noindex, no
follow
+ Uncommon header 'x-ob_mode' found, with contents: 1
+ Uncommon header 'x-permitted-cross-domain-policies' found, with
contents: none
+ OSVDB-3233: /icons/README: Apache default file found.
+ /phpmyadmin/: phpMyAdmin directory found
+ 7686 requests: 0 error(s) and 10 item(s) reported on remote host
+ End Time:           2017-08-30 03:06:48 (GMT7) (17 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested


     ************************************************************
*********
```

Gambar 11 Tampilan *Scanning* Nikto

Dari hasil *scanning* dengan nikto, *attacker* mendapatkan informasi *hostname* dan *port* yang digunakan pada PC server, serta *hostname* menggunakan Apache versi 2.4.5 dan phpmyadmin

Pergerakan *attacker scanning* nikto terdeteksi oleh suricata berhasil mengcapturenya.

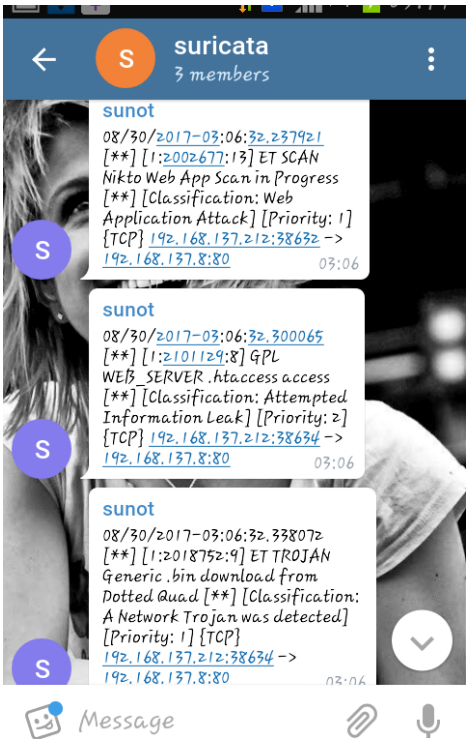
```
08/30/2017-03:06:32.237921  [**] [1:2002677:13] ET SCAN Nikto Web
 App Scan in Progress [**] [Classification: Web Application Attac
k] [Priority: 1] {TCP} 192.168.137.212:38632 -> 192.168.137.8:80

08/30/2017-03:06:32.300065  [**] [1:2101129:8] GPL WEB_SERVER .ht
access access [**] [Classification: Attempted Information Leak] [
Priority: 2] {TCP} 192.168.137.212:38634 -> 192.168.137.8:80
08/30/2017-03:06:32.338072  [**] [1:2018752:9] ET TROJAN Generic
.bin download from Dotted Quad [**] [Classification: A Network Tr
ojan was detected] [Priority: 1] {TCP} 192.168.137.212:38634 -> 1
92.168.137.8:80
08/30/2017-03:06:32.377595  [**] [1:2100977:13] GPL EXPLOIT .cnf
access [**] [Classification: access to a potentially vulnerable w
eb application] [Priority: 2] {TCP} 192.168.137.212:38634 -> 192.
168.137.8:80
```

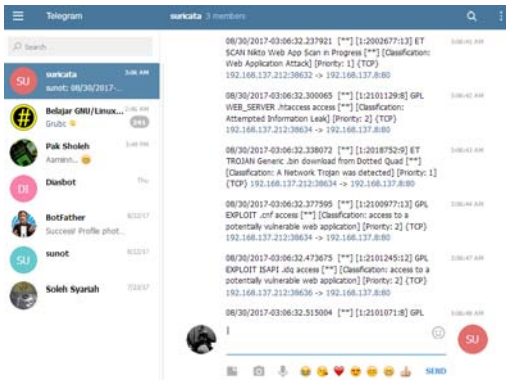
Gambar 12 *Log* Suricata Deteksi Nikto

Log deteksi serangan suricata sampai pada telegram

ISSN No. 2502-8782



Gambar 13. Nikto Notifikasi Telegram *Messenger* Tampilan notifikasi pada web telegram dari hasil deteksi suricata terhdap serangan menggunakan nikto *web scanning*.



Gambar 14 Tampilan *Web Telegram* Deteksi Nikto

Pengujian *Web Vulnerability* dari beberapa jenis *tools* yang di gunakan *attacker* seperti *port scanning* menggunakan Nmap, *web exploite* menggunakan WPScan, *web scanning* menggunakan Nikto telah berjalan. Bot *engine transfer log* yang sudah dibuat dan diaktifkan dan di jalankan pada *operating system* ubuntu server. Bot *engine* bekerja susuai perintah yang di input dalam program (*Source Code*) untuk mengolah dan menstransfer *file* fast.log suricata kepada bot API telegram lalu mengirim *file* tersebut ke telegram *user* dalam bentuk tampilan *chats* berisi informasi deskripsi kejadian ancaman yang sudah di deteksi dan *capture* oleh suricata. Maka dari itu penulis dapat menarik kesimpulan yang akan di bahas pada bab selanjutnya serta saran untuk penelitian selanjutnya dari kekurangan sistem yang dibangun.

6 Kesimpulan dan Saran

6.1. Kesimpulan

1. Menggunakan suricata dengan mengaktifakan *rules* serta mengkonfigurasi alamat *host* yang akan di lindungi, monitoring terhadap PC sever dimana terdapat *web* aplikasi di dalamnya dapat berjalan. Suricata berhasil mendekteksi beberapa jenis ancaman yang di lakukan *attacker*.
2. Dari hasil uji coba sistem, *telegram messenger* dapat menerima informasi serangan yang sudah di deteksi suricata sehingga sistem berjalan sesuai perancangan.
3. Dengan menerapkan sistem deteksi yang terintegrasi adanya notifikasi melalui aplikasi *telegram messenger* monitoring secara mobile dapat di lakukan menggunakan *smartphone*.

6.2 Saran

1. Dalam pengembangannya diharapkan dapat memanfaatkan aplikasi yang lebih baru seperti pemanfaatan teknologi berbasis android.
2. Diharapkan aplikasi ini dapat dikembangkan dengan penanggulangan serangan pada jaringan komputer secara langsung melalui metode yang sama atau metode yang lebih baik lagi.
3. Pengembangan aplikasi dengan memiliki banyak fungsi bukan hanya sebagai sistem monitoring.

Kepustakaan

[1]. Sopandi, Dede. 2010. ”Instalasi dan Konfigurasi Jaringan Komputer”. Bandung: Informatika Bandung.
[2]. Pratama, I Putu Agus Eka.2014. “Handbook Jaringan Komputer”. Bandung: Informatika.
[3]. Catur L., Azis & Herlambang, Moch. Linto.2008. “Panduan lengkap mennguasai router masa depan menggunakan mikrotik routerOS”. Yogyakarta : Andi Publisher Yogyakarta.
[4]. Sofana, Iwan.2008.”Membangun Jaringan Komputer”. Bandung : Informatika Banndung.
[5]. Ariyus, Dony, 2007 “Intrusion Detection System” Yogyakarta : Pernerbit Andi.
[6]. Sukamanji, Anjik dan Rianto 2008. “ Jaringan Komputer” Yogyakarta : Andi
[7]. Casley, J., and Kumar, 1989. The collection, analysis and use of monitoring and evaluation data. A word bank publication.
[8]. Clayton, Eric, Petry Francoise. 1983, Monitoring for agricultural and rural development project. The macmillan. London.
[9]. Scarfone Karen, Mell Peter, “Network Based IDPS” in Guide to Intrusion Detection and Prevention Systems (IDPS), Special Publication 800-94, National Institute of Standards and Technology, 2007
[10]. Saint 2008, Integrated Network Vulnerability And Penetration Testing http://www.saintcorporation.com/resources/SAINT_integrated_pen_testing.pdf
[11]. S’to, 2009, CEH : 100% Illegal, Jakarta : Jasakom.

ISSN No. 2502-8782

[12]. M. K. S. M. Alim Nuryanto, "Analisis Dan Implementasi Suricata, Snorby, Dan Barnyard2," 2015.
[13]. Von Hagen, William. 2007 Ubuntu Linux Bible. Indiana: Wiley Publishing, Inc.
[14]. OISF, 2015. "Suricata Documentation" Diakses dari http://redmine.openinfosecfoundation.org/projects/suricata/wiki (tanggal 4 juni 2016).
[15]. Wirawan, 2007, "Langkah mudah membangun jaringan", andi : Yogyakarta.
[16]. Arzikin, Hasnul, 2011 "Kitab Suci Jaringan Komputer Dan Koneksi Internet", Yogyakarta :Mediakom.
[17]. http://sapacerita.blogspot.co.id/2016/01/kelebihan-dan-kekurangan-telegram.html (di akses pada tanggal 28 juli 2016).
[18]. https://core.telegram.org/bots (di akses pada tanggal 6 juli 2017).
[19]. https://rizaumi.github.io/2015/12/11/mengenal-bot-telegram (di akses pada tanggal 6 juli 2017)..