Studi Tentang *Virtual Private Network* (VPN) Pada Sistem *Monitoring* Data *Frame* Video Menggunakan Raspberry Pi3

Study of Virtual Private Network (VPN) on Video Frame Data Monitoring Systems Using Raspberry Pi

M. Gopur Riswanto[1*], Afaf Fadhil Rifa'i[2], Nanang Ismail[3]
[1,3]Universitas Sunan Gunung Djati Bandung
Jl. A. H. Nasution No. 105, Bandung 40614 INDONESIA. (tlp: 022-7800525; fax: 022-7803936)
[2]Politeknik Manufaktur Bandung
Jl. Kanayakan No. 21, Bandung 40135 INDONESIA (tlp:022-2500241; fax:022-2502649)
gofurriswanto39@gmail.com[1], afaf_fadhil@yahoo.com[2], nanang.is@uinsgd.ac.id[3]

Abstrak – *Sistem monitoring akuisisi data berbasis web server mempunyai beberapa permasalahan terutama dalam hal keamanan jaringan. Salah satu solusi pengamanannya dengan menggunakan virtual private network. Studi ini mengujicoba skema keamanan web server dengan VPN menggunakan dua PC. Raspberry Pi3 digunakan sebagai kontroler dari deteksi motion dan web server. VPN server menggunakan Sistem Operasi Raspbian Jessie dan VPN client menggunakan Windows 7. Pengujian yang dilakukan meliputi pengujian QoS dan akuisisi data pada web server, serta pengujian keamanan web server dengan melakukan tes penetrasi menggunakan OWASP ZAP. Pengujian dilakukan dalam dua keadaan, yaitu pada saat terhubung dengan jaringan VPN dan saat tidak terhubung dengan jaringan VPN. Hasil pengujian QoS dan akuisisi data pada kedua kondisi tersebut menunjukkan adanya beberapa perbedaan nilai parameter: throughput sebesar 2.150 bits, delay sebesar 0,081 s, dan selisih 2 data frame yang hilang. Untuk pengujian keamanan web server, VPN memberikan hasil yang sangat signifikan. Jaringan yang terhubung dengan VPN memberikan peningkatan keamanan pada web server dibandingkan dengan jaringan tanpa VPN.*

Kata Kunci*: VPN, QoS, Akuisisi Data, Keamanan Web Server*

Abstract – *Raspberry Pi3 is used as a controller for motion detection and a web server. The VPN server uses the Raspbian Jessie Operating System and the VPN client uses Windows 7. The tests included QoS testing and data acquisition on the web server, as well as testing the web server security by conducting a penetration test using OWASP ZAP. Testing is done in two circumstances, namely when connected to a VPN network and when not connected to a VPN network. The results of QoS testing and data acquisition in the two conditions showed some differences in parameter values: throughput of 2,150 bits, delay of 0.081 s, and difference of 2 missing data frames. For web server security testing, VPN provides very significant results. Networks that are connected with VPNs provide increased security on web servers compared to networks without VPNs.*

Keywords*: VPN, QoS, Data Acquisition, Web Server Security.*

1. Pendahuluan

Beberapa aktivitas industri membutuhkan fasilitas *remote monitoring* yang efisien dan aman dalam melakukan pekerjaan mereka. Saat ini teknologi *monitoring* semakin berkembang, salah satunya pengembangan teknologi *video monitoring*. Dengan pengembangan teknologi *video monitoring*, pengguna bisa menggunakannya di berbagai kegiatan, salah satunya dalam *real time monitoring*[1].

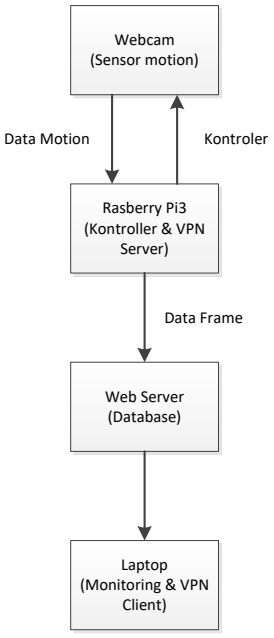
Remote Monitoring biasanya menggunakan sistem yang dapat mendukung dalam pengiriman data dan penerimaan data, seperti halnya *web server*. Teknologi *web server* dihadirkan sebagai upaya untuk memungkinkan akses sumber daya dari mana saja melalui internet[2]. Pada jaringan publik sering terjadinya permasalahan keamanan komunikasi. Kemudian dalam informasi pengiriman data dan penerimaan data berbasis *web server*, sangat penting untuk menjamin bahwa *web server* mempunyai keamanan yang baik[3]. Salah satu cara untuk mengatasi permasalahan itu adalah dengan mengimplementasikan *Virtual Private Network* (VPN), sehingga dapat membuat sebuah jaringan bersifat *private* dan aman dengan menggunakan jaringan publik maupun lokal serta menggunakan internet[4].

Sistem *monitoring* yang diakses secara *private* biasanya digunakan untuk me-*monitoring* data. Data yang didapatkan pada studi ini merupakan data *frame* yang terbentuk dari deteksi *motion*, sehingga dibutuhkan suatu komponen/alat yang mempunyai kapasitas *core* yang cukup besar dalam pengambilan data tersebut. Salah satu komponen/alat yang sering digunakan adalah Raspberry Pi3. Raspberry Pi3 menggunakan OS Linux dan sudah ter-*intergratted* dengan *WiFi*. Penggunaan dari Raspberry Pi3 ini sebagai pengganti PC *Desktop*. Raspberry Pi3 mampu menggantikan PC *Desktop* dalam melakukan *monitoring*[5].

Fokus penelitian ini membahas tentang pengaruh VPN pada kinerja sistem *monitoring* akuisisi data berbasis VPN dengan menggunakan *minicomputer* Raspberry Pi3 sebagai kontrol *webcam* dan pengambilan data. Dengan bertujuan mengetahui selisih nilai QoS dan nilai akuisisi data yang didapat, serta mengetahui pengaruh VPN pada kemanan *web server*.

2. Metode Penelitian

Perancangan VPN pada sistem *monitoring* akuisisi data menggunakan Raspberry Pi3 terbagi kedalam beberapa tahap perancangan. Seperti pada Gambar 1



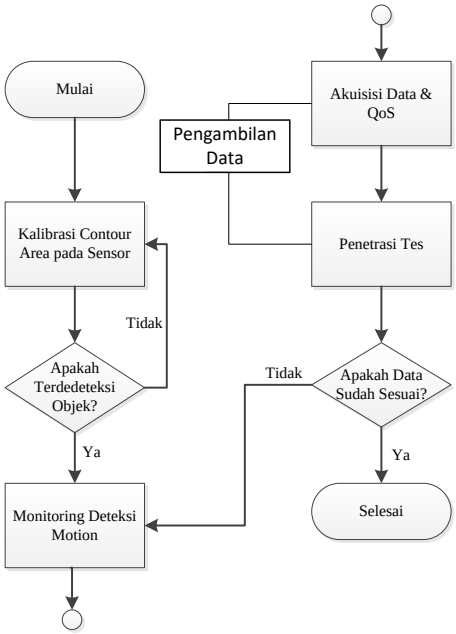
Gambar 1. Blok Diagram Penelitian

ISBN: 978-602-60581-1-9

Pada blok diagram, Raspberry Pi3 di-*install* dengan jaringan VPN agar menjadi VPN *server*. Kemudian VPN *client* di-*install* pada *platform* yang digunakan sebagai alat untuk melakukan *monitoring*, sehingga tercipta jaringan VPN berbasis *tunneling*.

Input yaitu berupa data *motion.* Data *motion* merupakan hasil dari deteksi objek menggunakan *webcam.* Raspberry Pi3 akan mengontrol *webcam* dengan pemrograman bahasa Python untuk mendapatkan deteksi terbaik. Raspberry Pi3 akan mengirimkan *output* berupa data *frame* menuju *web* server. Sehingga Raspberry Pi3 membutuhkan jaringan internet untuk melakukan pengiriman data *frame* menuju *database.* Proses *monitoring* dilakukan melalui laptop yang sudah terhubung dengan jaringan VPN dan tanpa VPN. Untuk VPN, dipasangkan pada Raspberry Pi3 sebagai VPN *server* dan PC sebagai VPN client.

Berdasarkan blok diagram penelitian pada Gambar 1 didapatkan algorima penelitian seperti Gambar 2



Gambar 2. Algoritma Penelitian Sistem *Monitoring* Akuisisi Data

Berdasarkan Gambar 2, nilai *contour area* dikalibrasi agar sensor dapat mendeteksi objek dengan baik. Hasil deteksi *motion* di-*convert* menjadi bentuk data *frame* yang akan dikirimkan menuju *web server*, sehingga terbentuk *monitoring* akuisisi data. Kemudian pada QoS, akuisisi data, dan tes penetrasi dilakukan pengambilan data yang dibutuhkan dengan tujuan agar mengetahui pengaruh dari VPN. Akuisisi data dilakukan pengambilan data dengan cara menghitung data yang ada *shell* Python dan yang ada *web server*, kemudian dicari jumlah selisih data yang didapatkan. Pada QoS, menggunakan *software* WireShark agar dapat melihat data *througput, delay,* dan *packet loss.* Kemudian pada tes penetrasi, dilakukan penetrasi keamanan pada alamat *web server* akuisisi data agar mendapatkan data peringatan keamanana pada Raspberry Pi3.

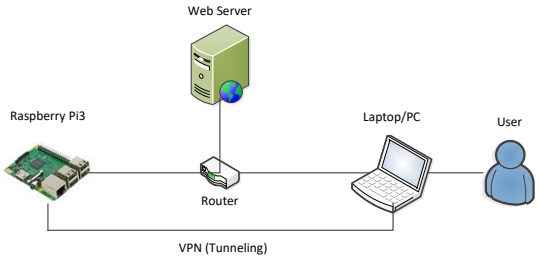
Perangkat yang digunakan (Raspberry Pi3) dihubungkan dengan jaringan VPN dan dipasang *web server* sebagai *database.* Kemudian didapatkan IP *address* dan dilakukan 4 pengujian untuk mendapatkan data perbandingan, seperti pengujian perbandingan IP *address* dan *packet loss*, Perbandingan *database* pada *web server*, serta perbandingan keamaan *web* server.

2.1. Metode *Tunneling*

Penelitian terkait pengujian VPN telah banyak dilakukan. Salah satu penelitian terkait VPN dibahas pada [4]. Paper tersebut melakukan pengujian terhadap jaringan VPN, dengan cara membandingkan pengaruh antar protokol satu dengan yang lainnya terhadap jaringan. Jaringan

ISBN: 978-602-60581-1-9

yang dibandingkan adalah PPTP dan L2TP. PPTP dan L2TP merupakan metode *tunneling* yang digunakan untuk membuat jaringan VPN, seperti pada Gambar 3



Gambar 3 Desain Sistem VPN *Tunneling*

Gambar 3 menunjukkan desain sistem VPN menggunakan metode *tunneling*.
VPN mempunyai beberapa metode dalam pembutatannya, seperti: metode *tunneling*, metode enkripsi, metode otentikasi *user*, dan metode integritas data. Referensi [6] menggunakan metode *tunneling* dalam membuat VPN. Referensi tersebut membandingkan nilai QoS dan keamanan jaringan antar protokol VPN.

2.2. Pengukuran QoS

Pengujian QoS bertujuan untuk mengetahui performa dari VPN, seperti *througput*, *delay*, dan *packet loss* [7]. Persamaan untuk mendapatkan nilai-nilai tersebut dijabarkan pada Persamaan (1), Persamaan (2), dan Persamaan (3).
Througput,

$$Througput = \frac{\text{Besar ukuran paket data}}{\text{Waktu Pengiriman data}} x1000 \quad (1)$$

Delay,

$$Delay = \frac{\text{Total } Delay}{\text{Total Paket yang Diterima}} \quad (2)$$

Packet loss,

$$Packet\ Loss = \frac{\text{Paket yang Dikirim} - \text{Paket yang Diterima}}{\text{Paket yang Dikirim}} X100\% \quad (3)$$

2.3. Metode Tes Penetrasi Keamanan *Web Server*

Berdasarkan referensi [8], untuk mengetahui kehandalan keamanan dapat dilakukan dengan tes penetrasi dari software OWASP ZAP. OWASP ZAP akan melakukan tes penetrasi keamanan terhadap *web* yang akan diuji.

3. Hasil dan Pembahasan

Rancang Bangun VPN pada sistem *monitoring* akuisisi data menggunakan beberapa percobaan.

3.1. Pengujian dan Analisis QoS

Pengujian selanjutnya, menggunakan *software* WireShark yang terhubung dengan *Wi-Fi* Raspberry Pi3 dengan IP *Address* yang sudah diketahui, yaitu 192.168.1.8 dengan terhubung VPN dan 192.18.1.10 tanpa terhubung VPN.

Pengujian ini bertujuan agar mengetahui pengaruh baik atau buruk VPN pada QoS.

ISBN: 978-602-60581-1-9

Tabel 1 Hasil QoS (Dengan VPN)

***Througput* (bit/s)**	***Delay* (s)**	***Packet Loss* (%)**
2.061	0,348	0
1.953	0,338	0
2.096	0,336	0
2.152	0,311	0
1.987	0,335	0
3.999	0,181	0
2.035	0,323	0
2.243	0,309	0
2.125	0,312	0
2.255	0,303	0
22.906	**3,096**	**0**

Tabel 2 Hasil QoS (Tanpa VPN)

Uji ke-	***Througput* (bit/s)**	***Delay* (s)**	***Packet Loss* (%)**
1	1.918	0,344	0
2	4.935	0,153	0
3	1.881	0,356	0
4	2.068	0,312	0
5	1.882	0,356	0
6	2.047	0,321	0
7	4.042	0,172	0
8	1.882	0,356	0
9	2.521	0,289	0
10	1.880	0,356	0
Rerata	**25.056**	**3,015**	**0**

Tabel 1dan 2 menunjukkan hasil QoS pada Raspberry Pi3 yang terhubung dengan jaringan VPN dan yang tidak terhubung dengan VPN. Terdapat selisih hasil QoS antara Raspberry Pi3 yang tidak terhubung VPN dan Raspberry Pi3 yang terhubung VPN. Seperti pada Tabel 3

Tabel 3 Selisih Hasil QoS

Keadaan	***Througput* (bit/s)**	***Delay* (s)**	***Packet Loss* (%)**
Dengan VPN	22.906	3,096	0
Tanpa VPN	25.056	3,015	0
Selisih	2.150	0,081	0

Tabel 3 menunjukan hasil QoS dengan menggunakan WireShark pada masing-masing keadaan dalam jangka waktu yang berbeda. Analisis yang dilakukan pada QoS yaitu:

1) *Througput* pada jaringan VPN lebih kecil daripada *througput* jaringan tanpa VPN, yaitu 22.906 bps untuk yang terhubung VPN dan 25.056 untuk yang tidak terhubung VPN. Ini berarti VPN mempengaruhi pada pengiriman data (bit). Saat pengiriman data (bit), terjadi penurunan kecepatan rata-rata yang diterima oleh suatu *node* (*web* server) dalam selang waktu tertentu dan banyaknya pengguna jaringan. Perbandingan bps yang didapatkan dengan menggunakan WireShark adalah 2.150 bit per sekon.

ISBN: 978-602-60581-1-9

2) Pengujian dilakukan dengan lama waktu 306,421 detik untuk yang terhubung dengan VPN dan 306,699 detik untuk yang tanpa terhubung dengan VPN. Didapat perbandingan hasil *delay* yaitu 0,081. Perbandingan tersebut didapat dari hasil data rata-rata *delay* saat paket yang dikirimkan menuju *node* (*web server*). Saat terhubung jaringan VPN, *delay* yang dihasilkan lebih besar. Disebabkan adanya perubahan kecepatan antar jaringan dan pemadatan *bandwidth* secara tiba-tiba.
3) Dari kedua keadaan tersebut, *packet received* dan *packet transmitted* mempunyai nilai yang sama atau tidak ada paket data yang hilang. Sehingga dari kedua keadaan tersebut mempunyai nilai *packet loss* 0%. Raspberry Pi3 tidak terhubung dengan jaringan VPN, tidak mengalami *packet loss* dan mempunyai persentase *packet loss* 0 %. Serta pada saat Raspberry Pi3 terhubung dengan jaringan VPN, Raspberry Pi3 juga tidak mengalami *packet loss* atau mempunyai persentase *packet loss* 0 %.

VPN tidak mempengaruhi performa Raspberry Pi3 dalam pengiriman paket data, namun mempengaruhi pada pengiriman sejumlah data dan waktu pengiriman data pada *node* (*web server*).

3.2. Pengujian dan Analisis Selisih Data *Frame*

Pengujian selanjutnya, mencari selisih jumlah data yang ada pada *web server* pada saat jaringan terhubung dengan VPN dan tanpa terhubung dengan VPN.

Pengujian ini bertujuan mengetahui jumlah dan selisih data *frame* pada *shell* Python dan *web server* dalam kurun waktu tertentu ketika melakukan deteksi motion, serta mengetahui pengaruh VPN pada jumlah data yang didapatkan pada *web server*. Sehingga data yang didapatkan yaitu:

1) Selisih data *frame* dengan VPN

$$\begin{aligned} Jumlah\ data\ frame\ shell\ Python - Jumlah\ data\ frame\ web\ server \\ = Jumlah\ Data\ Frame\ Hilang \\ 373\ data\ frame - 348\ data\ frame = \mathbf{25\ data\ frame} \end{aligned} \quad (4)$$

Tabel 4 Selisih Akuisisi Data *Web Server* Dengan VPN

Keadaan	Dengan VPN
Waktu Pengambilan Data	6 detik
***Delay* Video**	5,9 detik
Data *Frame* Hilang	25 data *frame*

2) Selisih data *frame* tanpa VPN:

$$\begin{aligned} Jumlah\ data\ frame\ shell\ Python - Jumlah\ data\ frame\ web\ server \\ = Jumlah\ Data\ Frame\ Hilang \\ 383\ data\ frame - 360\ data\ frame = \mathbf{23\ data\ frame} \end{aligned}$$

Tabel 5 Selisih Akuisisi Data *Web Server* Tanpa VPN

Keadaan	Tanpa VPN
Waktu Pengambilan Data	6 detik
***Delay* Video**	5,7 detik
Data *Frame* Hilang	23 data *frame*

ISBN: 978-602-60581-1-9

Tabel 4 dan Tabel 5 menunjukkan adanya selisih akuisisi data yang tidak signifikan. Sehingga didapatkan analisis yaitu sebagai berikut:

1) Perbandingan ini disebabkan adanya *delay* video pada saat kedua kedaan tersebut (dengan VPN dan tanpa VPN) melakukan *running program* deteksi *motion*.
2) Raspberry Pi3 yang dihubungkan dengan jaringan VPN, menyebabkan data *frame* yang dikirimkan pada *web server* harus melewati jaringan VPN terlebih dahulu sehingga menyebabkan data yang hilang menjadi lebih banyak yaitu 2 data *frame*.

Raspberry Pi3 yang dihubungkan dengan jaringan VPN menyebabkan *delay* yang lebih lama sebanyak 0,2 detik. Disebabkan ketika pemrograman di-*running* untuk melakukan deteksi *motion*, data *frame* yang masuk pada *web server* terhenti sejenak pada saat mensinkronkan dengan *database* (*web server*).

3.3. Pengujian dan Analisis Keamanan *Web Server* Menggunakan OWASP ZAP

Pengujian dan analisis ini menggunakan *software* OWASP ZAP untuk melakukan penetrasi keamanan pada *web server*. Dalam melakukan tes penetrasi OWASP ZAP tidak terpaku terhadap waktu tes penetrasi. OWASP ZAP secara otomatis akan mencoba pengetesan menggunakan kategori-kategori yang disediakan oleh OWASP ZAP itu sendiri. Pengujian ini bertujuan untuk mengetahui pengaruh VPN terhadap keamanan *web server*. Sehingga OWASP ZAP memberikan data hasil tes penetrasi kemanan. Data hasil uji ditunjukkan pada Tabel 6 dan Tabel 7.

Tabel 6 Data Hasil Tes penetrasi Dengan VPN

Jenis Tes penetrasi	*Request*	*Respon Alert*	Waktu Uji (detik)
Relative Path Confusion	1	1	14,222
Cookie Slack Detector	0	0	0,003
.htaccess *Information Leak*	1	0	0,010
User Agent Fuzzer	6	6	27,203
Cross Site Scripting (*Reflected*)	14	0	27,906
Total	**22**	**7**	**69,344**

Tabel 7 Data Hasil Tes penetrasi Tanpa VPN

Jenis Tes penetrasi	*Request*	*Respon Alert*	Waktu Uji (detik)
Relative Path Confusion	1	0	2,743
Cookie Slack Detector	1	1	0,337
.htaccess *Information Leak*	1	1	0,271
User Agent Fuzzer	7	7	2,310
Cross Site Scripting (*Reflected*)	16	1	1,109
Total	**26**	**10**	**6,770**

Tabel 6 dan Tabel 7 menunjukkan hasil tes penetrasi keamanan *web server* menggunakan *software* OWASP ZAP. Hasil data tersebut menunjukkan selisih data seperti pada Tabel 8

Tabel 8 Selisih Data Hasil Tes penetrasi

Selisih Jumlah *Request*	4 *request*
Selisih Jumlah *Respon Alert*	3 *respon alert*
Selisih Waktu Uji (detik)	62,574

ISBN: 978-602-60581-1-9

Tabel 8 menunjukkan selisih data hasil tes penetrasi, sehingga didapatkan analisis sebagai berikut:

1) Analisis selisih *request*, tes penetrasi yang digunakan bersifat otomatis, ketika tes penetrasi sudah cukup untuk dalam melakukan penetrasi kerentanan keamanan maka secara otomatis tes penetrasi akan langsung berhenti. *Request* merupakan banyaknya permintaan dari tes penetrasi menuju *web server* yang di penetrasi kerentanan keamanannya untuk menghasilkan gangguan dan peringatan pada *web server*. Semakin kecil jumlah *request* yang diminta oleh tes penetrasi, itu menandakan *web server* mempunyai keamanan yang baik. Sebaliknya, semakin besar jumlah *request* yang diminta oleh tes penetrasi, itu menandakan *web server* mempunyai keamanan yang tidak baik.
2) Analisis selisih *respon alert, alert* merupakan hasil tes penetrasi dari *request* yang dikirimkan pada *web server*. Semakin kecil jumlah *alert* yang dihasilkan oleh tes penetrasi, itu menandakan *web server* mempunyai keamanan yang baik. Sebaliknya, semakin besar jumlah *alert* yang dihaslkan oleh tes penetrasi, itu menandakan *web server* mempunyai keamanan yang tidak baik
3) Analisis selisih waktu uji (detik), waktu merupakan lamanya tes penetrasi yang dilakukan pada *web server* agar dapat merentankan keamanan *web server*. Semakin kecil waktu yang dibutuhkan oleh tes penetrasi untuk mengetes keamanan *web server*, itu menandakan *web server* mempunyai keamanan yang baik. Sebaliknya, Semakin besar waktu yang dibutuhkan untuk melakukan tes penetrasi untuk mengetes keamanan *web server*, itu menandakan *web server* mempunyai keamanan yang tidak cukup baik.

Dari beberapa perbandingan hasil tes penetrasi, dapat disimpulkan bahwa VPN mempunyai pengaruh yang baik untuk meningkatkan keamanan *web server*.

4. Kesimpulan

Hasil penelitian ini adalah VPN memberikan pengaruh pada sistem *monitoring* akuisisi data. IP *address* menjadi statis dengan IP *address* 192.168.1.8. Mempengaruhi pada *througput* dan *delay*, yaitu menyebabkan penurunan kecepatan pada waktu tertentu, pemadatan *bandwidth* secara tiba-tiba. Namun tidak menyebabkan *packet* data yang hilang, sehingga *packet loss*-nya sebesar 0%. VPN menyebabkan *delay* video pada saat proses *monitoring* akuisisi data dengan selisih 0,2 detik dan menyebabkan data *frame* yang hilang sebanyak 2 data *frame*.

VPN juga memberikan peningkatan keamanan pada *web server* dibandingkan tanpa VPN, *alerts* yang dihasilkan lebih sedikit dengan selisih 3 *alerts* dan waktu tes penetrasi keamanan yang dihasilkan menjadi semakin lama dengan selisih 62,574 detik. Semakin lama OWASP ZAP melakukan tes penetrasi keamana, itu berarti tes penetrasi keamanan mengalami kesulitan untuk melakukan *attack* pada *web server*.

Referensi

[1] Sampoerna, Zainuri, Nurussa'adah. "Rancang Bangun Sistem Monitoring Keamanan Rumah Bebasis Web". *J. Mahasiswa TEUB.* vol.4. 2017.
[2] Ashari. A, Setiawan. H. "*Cloud Computing*: Solusi ICT". *J. Sistem Informasi* (JSI), *Vol.3, No.2.* 2011.
[3] Triyono. J, Rr. Yuliana, Fahmi Dimas. "Analisis Perbandingan Kinerja Jaringan VPN Berbasis Mikrotik Menggunakan Protokol PPTP dan L2TP Sebagai Media Transfer Data". *J. JARKOM Vol. 1, No. 2, 2338-6312.* 2014.
[4] Afrianto. I, Setiawan Eko. "Kajian *Virtual Private Network* (VPN) Sebagai Sistem Pengamanan Data Jaringan Komputer". *Majalah Ilimiah UNIKOM Vol.12, No.1.* 2011.

ISBN: 978-602-60581-1-9

[5] Permana, Tauriq Djasa. "Sistem Monitoring Menggunakan Mini Raspberry Pi". *J. Teknik Komputer Unikom, Vol.3 No.1*. 2014.
[6] Oktivasari, Prihatin, Andri B. Utomo. "Analisa *Virtual Private Network* (VPN) Menggunakan OpenVPN dan *Point to Point Tunneling Protocol*". *J.Penelitian omunikasi dan Opini Publik, Vol. 20 No.2 (185-202)*. 2016.
[7] Mubarok, Hari. "Analisis *Quality of Service* (QoS) Jaringan Komputer PLN Area Surakarta". Skripsi, Universitas Muhamammadiyah, Surakarta, Indonesia. 2016.
[8] OWASP. "OWASP Zad Attack Proxy", *owasp.org*. [Online]. Available: https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project/. [Accessed: 14-Jun-2018].
[9] Satyanarayana, Reddy, Sai Teja, Basit Habibuddin. "IOT *Based Weather Station Using* Raspberry Pi3". *J. Of Chemical and Pharmaceutical Science (0974-2115)*. 2016.
[10] M. B. Kalpana, M. Tech Student. "*Online Monitoring Of Water Quality Using* Raspberry pi3 *Model* B". *International J. of Innovate Technology and Research, Vol.4 No.6, (4790-4795)*. 2016.

ISBN: 978-602-60581-1-9