BAB I
PENDAHULUAN

1.1 Latar Belakang Masalah

Logging / pencatatan kejadian merupakan hal yang sangat penting dalam sebuah sistem komputer, hampir semua sistem di dunia IT / komputer mempunyai *logger* (sistem yang menjalankan *logging*), baik di MikroTik itu sendiri. Ouput dari *logging* tersimpan di dalam file, seringkali di sebut "log file", dimana di dalam log file, khususnya di dalam log file mikrotik berisi timestamp / waktu kejadian sebuah "event / kejadian" terjadi, topik / kategori dari kejadian dan juga message / pesan rinci dari kejadian itu sendiri. Dan pada jaringan dengan perangkat yang banyak dan kompleks, seorang system administrator seringkali mengirim log file ke sebuah sistem terpusat untuk memudahkan system administrator untuk memantau dan menganalisa nya. Dan karena sangat penting dan sensitifnya nya informasi di dalam log file, system administrator di haruskan meng-enkripsi log-log tersebut sebelum di kirim ke sistem terpusat agar saat proses pengiriman tidak ada orang yang dapat melihat atau mengganti log file tersebut. Tetapi sayangnya, di perangkat MikroTik, hanya terdapat fitur remote syslog, dan tidak ada nya fitur untuk meng-enkripsi log-log tersebut saat akan di kirimkan ke sistem terpusat

Permasalahan di atas dapat di selesaikan dengan berbagai cara, namun hingga saat ini hanya ada 1 cara yang umum, yaitu dengan menggunakan TLS-Based Remote Syslog (rsyslog) ditambah dengan TLS-Proxy pada server standalone yang berada pada jaringan lokal. Secara singkat, cara kerja MikroTik menggunakan TLS-Proxy ini adalah dengan cara mengirim log dari mikrotik ke server TLS-Proxy yang masih berada pada jaringan lokal yang sama, kemudian dari server TLS-Proxy log tersebut akan di enkripsi lalu di kirimkan ke server log terpusat yang berada di jaringan internet / cloud. Yang nantinya di harapkan dengan adanya enkripsi dari server TLS-Proxy tersebut, log-log yang di kirim ke server terpusat melalui jaringan WAN / internet dari server TLS-Proxy akan aman dari attacker yang ingin mengganti / melihat isi file log tersebut. Tetapi sayangnya solusi menggunakan TLS-Proxy dinilai terlalu susah untuk diterapkan, dirawat

dan terlalu mahal dari sisi operasional. Di sinilah penulis membuat sebuah terobosan yang di harapkan dapat menyederhanakan metode TLS-Proxy di atas dan memudahkan system administrator itu sendiri.

Pada karya tulis ini penulis membuat sebuah tool / program yang pada dasarnya mempunyai tujuan yang sama, yaitu mengambil / *gathering* log pada perangkat-perangkat MikroTik dengan enkripsi, tetapi disini penulis menggunakan metode yang sederhana, mudah dan murah di bandingkan dengan menggunakan TLS-Proxy. Tools ini nantinya akan memanfaatkan sistem protokol dari layanan SSH (secure socket shell). Di mana layanan SSH ini seringkali di gunakan untuk me-remote / mengakses perangkat MikroTik dengan tampilan CLI (command line interface). Secara bawaan, SSH akan membuat sebuah koneksi berupa tunnel / terowongan yang di enkripsi, inilah nilai lebih yang ada pada layanan SSH. Dan nantinya alat / program ini akan menggunakan metode pull-based system, yang mana seluruh pekerjaan dan pengaturan akan berada pada aplikasi ini, sehingga perangkat-perangkat MikroTIk yang ada tidak perlu di lakukan pengaturan yang banyak dan rumit, seperti yang di gunakan pada metode TLS-Proxy. Harapan penulis dengan adanya program ini, akan memudahkan *system administrator* dalam mengumpulkan log-log dari perangkat-perangkat MikroTik yang ada tanpa harus menyediakan, mempersiapkan dan merawat server sendiri untuk TLS-Proxy pada masing-masing jaringan lokal.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, maka pokok permasalahan yang akan diteliti adalah Bagaimana cara menerapkan enkripsi pada log yang dikirim oleh mikrotik ke server terpusat tanpa memerlukan server / program perantara yang akan mengakibatkan bertambahnya biaya, waktu dan upaya dalam menerapkan sistem tersebut.

Oleh karena itu, dalam mengatasi permasalahan diatas, perlu diimplementasikan program python mikrotik *logger* menggunakan metode *pull-based system* untuk mengetehaui hasilnya, yang nantinya penulis harapkan dapat

membantu administrator dalam membuat sistem yang aman tetapi dengan biaya, waktu dan upaya yang minim

1.3 Batasan Masalah

Untuk mempersempit pembahasan pada skripsi ini, maka dibuat batasan-batasan sebagai berikut:

a. Aplikasi di rancang menggunakan bahasa Python versi 3
b. *Libraries* python-paramiko di gunakan untuk SSH
c. *Libraries* python-sqlite / python-mysql di gunakan untuk berkomunikasi dengan database
d. Perangkat MikroTik yang di gunakan versi RB951Ui-2HnD
e. Basis data yang di gunakan menggunakan sqlite dan mysql / mariadb
f. Sistem operasi yang di gunakan untuk menjalakan aplikasi menggunakan linux, yaitu arch linux dan armbian
g. Pengguna aplikasi ini di tujukan kepada *network / system administrator*
h. Metode autentikasi pada SSH menggunakan password
i. Algoritma KEX pada SSH menggunakan deffie-helman-sha256
j. Cipher pada SSH menggunakan aes128
k. Penelitian mencakup cara kerja program, analisis trafik dengan dan tanpa enkripsi, dan hasil analisa parsing dari program yang ada di dalam database

1.4 Tujuan Penelitian

Tujuan dari penelitan ini adalah untuk mengimplementasikan program *python mikrotik logger* yang diharapkan bisa memberikan kemanan pada fitur standar remote syslog di MikroTik dan menggantikan metode remote syslog menggunakan TLS-Proxy yang terlalu memakan biaya, waktu dan upaya dalam membuat dan mengoperasikannya.

1.5 Sistematika Penulisan

Bab I Pendahuluan, berisi: latar belakang, rumusan masalah dan hipotesis, batasan masalah, tujuan penelitian, dan sistematika penulisan.

Bab II Landasan Teori, berisi: hasil penelitian sejenis yang sudah pernah dilakukan sebelumnya, teori penunjang, dan referensi berupa buku, jurnal, dan laporan skripsi/tesis.

Bab III Metodologi Penelitian, berisi: penjelasan mengenai metode penelitian yang digunakan untuk memahami dan mengexplorasi obyek penelitian, hasil observasi / pengumpulan data, masalah yang terdapat pada obyek, dan gambaran umum proyek atau obyek penelitian, hingga Rencana Alur Penelitian.

Bab IV Pembahasan, berisi: rancangan proyek, implementasi *coding* dan desain, serta evaluasi rancangan. Selanjutnya alur pengerjaan proyek, metode testing, hingga hasil akhir penelitian dan pembahasan analisis hasil akhir penelitian, termasuk pembahasan hasil-hasil uji coba (testing). Data hasil akhir pengujian dapat berupa grafik, table, data monitoring, log system, dan lain-lain, dengan pembahasan.

Bab V Penutup, berisi kesimpulan dari hasil akhir penilaian proyek, dan saran.