Laporan Penelitian

Pengembangan Manajemen Keamanan Sistem dan Informasi dengan Penerapan Sistem Pendeteksi menggunakan OSSIM alienvault

disusun oleh:

Wilfridus Bambang Triadi Handaya, S.T., M.Cs.

Bernard Renaldy Suteja, S.Kom., M.Kom.

Fakultas Teknologi Informasi

Universitas Kristen Maranatha

Bandung

2011

LEMBAR IDENTITAS

1. Judul Penelitian
   Pengembangan Manajemen Keamanan Sistem dan Informasi dengan Penerapan Sistem Pendeteksi menggunakan OSSIM alienvault

2. Ketua/Penanggung Jawab Pelaksana Kegiatan Penelitian:
   Nama (lengkap dengan gelar) : Wilfridus Bambang Triadi H, S.T., M.Cs.
   NIK : 720248
   Jabatan Akademik / Golongan : Lektor / III D
   Fakultas / Jurusan : Teknologi Informasi / Teknik Informatika

3. Jumlah Tim Peneliti : 2 orang

4. Lokasi Pelaksana Penelitian : Fakultas Teknologi Informasi
   Universitas Kristen Maranatha

5. Lama Pelaksanaan : 3 bulan

6. Sumber Dana Penelitian : Universitas Kristen Maranatha

7. Biaya Penelitian : Rp. 7.550.000,-

Bandung, 23 Juni 2011

Ketua/ Penanggung Jawab Pelaksana

Wilfridus Bambang Triadi Handaya, S.T., M.Cs.

Mengetahui

Dekan Fakultas Teknologi Informasi	Ketua LPPM
Radiant Victor Imbar, S.Kom., M.T.	Ir. Yusak Gunadi Santoso, M.M.

LEMBAR PENGESAHAN

Judul Penelitian : Pengembangan Manajemen Keamanan Sistem dan Informasi dengan Penerapan Sistem Pendeteksi menggunakan OSSIM alienvault

Peneliti : 1. Wilfridus Bambang Triadi Handaya, S.T., M.Cs.
2. Bernard Renaldy Suteja, S.Kom., M.Kom.

Lokasi Pelaksana Penelitian: Fakultas Teknologi Informasi
Universitas Kristen Maranatha
Jl. Surya Sumantri no. 65
Bandung

Penelitian ini telah diselesaikan pada tanggal 23 Juni 2011 sebagai salah satu perwujudan Tridharma Perguruan Tinggi Universitas Kristen Maranatha

Bandung, 23 Juni 2011

Ketua Peneliti

Wilfridus Bambang Triadi Handaya, S.T., M.Cs.

Dekan Fakultas Teknologi Informasi

Radiant Victor Imbar, S.Kom., M.T.

Ketua LPPM

Ir. Yusak Gunadi Santoso, M.M.

INTISARI

Layanan suatu organisasi terhadap kustomer tidak lagi bersifat manual, tetapi sudah beranjak pada area digital, dimana ketersediaan data dan aplikasi berbasis *client/server* menjadi suatu keharusan sebagai optimalisasi dan fitur keunggulan. Untuk itu dibutuhkan penyedia layanan, pada sisi perangkat keras maupun perangkat lunak,

Korelasi *engine* yang disediakan dalam aplikasi OSSIM, dengan basis berasal dari aplikasi-aplikasi *open source* yang telah dikenal baik di bidang keamanan informasi, yang memberikan informasi secara rinci dan komprehensif dari berbagai level melalui antarmuka visualisasi, perangkat manajemen insiden keamanan dan pelaporan.

Informasi yang dihasilkan merupakan kumpulan data spesifik kepada pengguna berdasarkan jaringan atau sensor yang telah dipasang, dan akan berbeda sesuai dengan kebutuhan dari masing-masing pengguna, dan dapat dipergunakan sebagai *arsenal* dari kalangan professional auditor keamanan sistem.

Kata kunci : *OSSIM, sensor, jaringan*

1. Pengantar

1.1. Latar Belakang Masalah

Layanan suatu organisasi terhadap kustomer tidak lagi bersifat manual, tetapi sudah beranjak pada area digital, dimana ketersediaan data dan aplikasi berbasis *client/server* menjadi suatu keharusan sebagai optimalisasi dan fitur keunggulan. Untuk itu dibutuhkan penyedia layanan, pada sisi perangkat keras maupun perangkat lunak, dengan berbagai jenis kebutuhan serta fungsi yang dihadirkan, mulai dari web, email, ftp, hingga layanan multimedia, yang mencerminkan begitu eratnya era digitalisasi dengan kehidupan manusia pada umumnya.

Komunikasi jaringan melalui protokol dan port yang menjadi media untuk transfer data antara server dengan client menjadi jalur yang terbuka untuk semua pihak dapat mengakses aliran data di antara kedua sisi tersebut, baik pihak yang memiliki otorisasi atau legal maupun non legal. Investasi yang besar dan konfigurasi yang kompleks dalam menghadirkan layanan tersebut membuat data yang ada didalamnya harus dapat dilindungi dari pengaksesan oleh pihak-pihak yang tidak bertanggung jawab.

Administrator memiliki tanggung jawab yang besar dalam upaya pengamanan jaringan, sistem, serta data, membuat berbagai upaya dilakukan sebagai tindakan pengamanan. Salah satunya upaya pengamanan tersebut adalah menggunakan metode pendeteksi penyusupan atau serangan, atau yang lebih dikenal sebagai Intrusion Detection System (IDS) serta selanjutnya dapat ditingkatkan menjadi metode pencegahan sehingga tidak berakibat fatal terhadap keberlangsungan sistem, atau yang lebih dikenal dengan istilah Intrusion Prevention System (IPS).

Penelitian ini dalam implementasinya menggunakan aplikasi OSSIM yang merupakan singkatan dari Open Source Security Information Management. Dengan tujuan akhir yang ingin diperoleh adalah mendapatkan kompilasi lengkap dari perangkat lunak yang khusus untuk penanganan keamanan sistem, dan memberikan

informasi kepada administrator jaringan atau keamanan sistem mengenai aspek detail dari setiap *host, network, server*, hingga perangkat keras yang terpasang.

Korelasi *engine* yang disediakan dalam aplikasi OSSIM, dengan basis berasal dari aplikasi-aplikasi *open source* yang telah dikenal baik di bidang keamanan informasi, yang memberikan informasi secara rinci dan komprehensif dari berbagai level melalui antarmuka visualisasi, perangkat manajemen insiden keamanan dan pelaporan, berdasarkan klasifikasi aset-aset yang ada seperti *host, network, group*, dan *services*.

Informasi yang dihasilkan merupakan kumpulan data spesifik kepada pengguna berdasarkan jaringan atau sensor yang telah dipasang, dan akan berbeda sesuai dengan kebutuhan dari masing-masing pengguna, dan dapat dipergunakan sebagai *arsenal* dari kalangan professional auditor keamanan sistem.

1.2. Rumusan Masalah.

Layanan data yang diberikan dalam ruang lingkup Teknologi Informasi berhubungan antara *server* dan *client*. Penyedia layanan atau yang biasa disebut sebagai *server*, terhubung dalam suatu jaringan yang kompleks dan berada pada posisi utama atau garda depan dari sistem informasi dari organisasi. Berbagai serangan yang dapat muncul, baik itu dari pihak internal maupun eksternal, harus dapat diantisipasi sejak dini menggunakan berbagai penerapan kebijakan, baik pada sisi *software, hardware,* maupun *policy* yang ada.

Cara untuk meminimalisir ancaman terhadap keberlangsungan umur hidup sistem tersebut dapat dilakukan melalui pendekatan berbasis pendeteksian sebelum terjadinya serangan seperti yang dilakukan dalam penelitian ini, atau melalui pendekatan berbasis setelah terjadinya bencana. Berbagai masalah yang dapat didokumentasikan adalah sebagai berikut:

a. Bagaimana cara melakukan pendeteksian serangan melalui paket data yang melewati area lingkungan terbatas pada sisi server?
b. Bagaimana cara menterjemahkan berbagai *rule* untuk menghadapi pola aliran data secara spesifik sesuai dengan kebutuhan dan insiden yang terjadi?
c. Bagaimana mendokumentasikan secara digital log detil dan terstruktur insiden maupun pola aliran data dari jaringan?

1.3. Batasan Masalah

Ruang lingkup permasalahan yang ada sangat luas, membuat dalam penelitian ini penulis membatasi dalam pengimplementasian program ini, yaitu:

a. Membutuhkan penempatan sensor berupa sistem pendeteksi ini pada area DMZ (*demiliterized zone*) atau area terdepan pada infrastruktur server di organisasi.

1.4. Tujuan Penelitian

Tujuan dilakukannya penelitian ini adalah sebagai berikut:

a. Menciptakan sistem pendeteksi atau sensor terhadap area jaringan menggunakan aplikasi berbasis *open source*, yaitu alientvault OSSIM.
b. Memberikan laporan kepada administrator sistem mengenai upaya penyerangan terhadap sistem, melalui catatan atau log yang dihasilkan oleh aplikasi.
c. Menjadikan laporan dari aplikasi sebagai bukti digital yang mencatat segala upaya penyerangan atau penetrasi ke dalam suatu area server.
d. Menghadirkan sistem pencegahan penyusupan berdasarkan pola-pola serangan yang ditujukan kepada sistem, sebelum berakibat hilangnya layanan secara keseluruhan.

1.5. Manfaat Penelitian

Penelitian ini bertujuan untuk menghasilkan beberapa manfaat yaitu :

a. Mendokumentasikan serta memberikan laporan secara rinci kepada administrator sistem mengenai kondisi keamanan dari lalu lintas jaringan dan server, sehingga dapat dilakukan tindakan-tindakan pencegahan upaya penyerangan demi menjaga stabilitas dari layanan.

b. Hasil dari penelitian dapat diterapkan pada lingkup area server di universitas, dan digunakan sebagai media pemantauan penggunaan komputer publik, serta meningkatkan efisiensi dalam manajamen keamanan server dan jaringan.

2. OSSIM Alientvault

2.1. Pengertian

Open Source AlienVault Siem (OSSIM) adalah sistem keamanan yang komprehensif yang mencakup open source dari deteksi untuk menghasilkan metrik dan laporan ke tingkat eksekutif. AlienVault ditawarkan sebagai produk keamanan yang memungkinkan untuk mengintegrasikan ke dalam satu konsol, semua perangkat keamanan dan alat yang dimiliki di jaringan, dan pemasangan alat-alat open source seperti Snort, openvas, ntop dan OSSEC. Cara kerjanya adalah sistem melakukan penilaian risiko untuk setiap peristiwa dan hubungan yang terjadi. Selama proses korelasi, dari serangkaian pola, menghasilkan mekanisme baru untuk mendeteksi serangan atau masalah dengan jaringan.

Pengaksesan semua informasi yang dikumpulkan dan dihasilkan oleh sistem pengguna dapat memanfaatkan Web konsol, dan juga memungkinkan untuk mengkonfigurasi sistem dan melihat keadaan keseluruhan jaringan secara *realtime*.

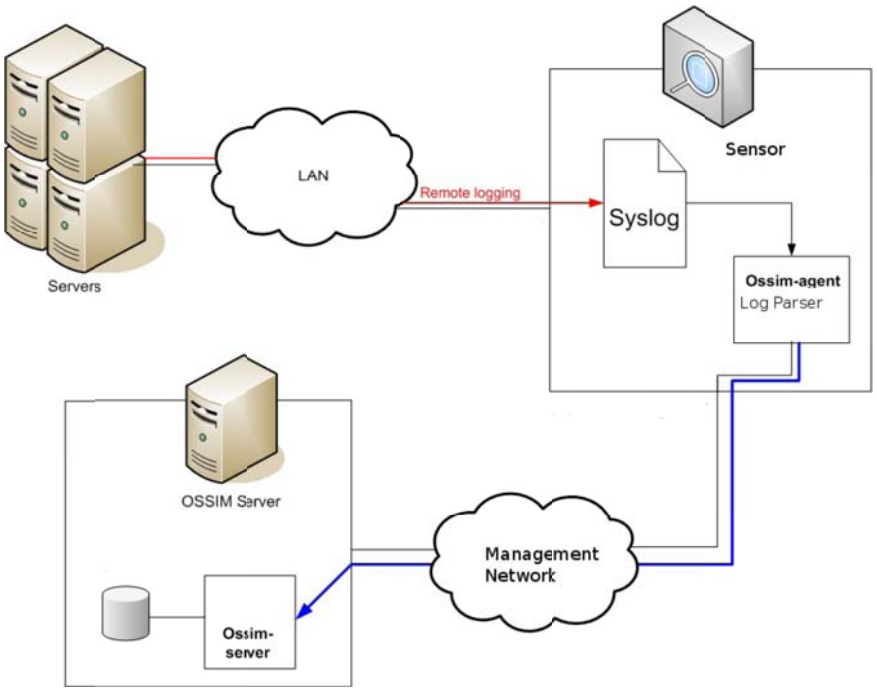
2.2. Mengapa harus melakukan instalasi?

AlienVault adalah produk yang mengintegrasikan lebih dari 30 alat Open Source. Sistem operasi dan berbagai aplikasi yang telah secara standar terdapat di dalamnya, telah dimodifikasi untuk meningkatkan fungsinya dalam sistem. Kompilasi berbagai aplikasi, kemudian penginstallan dan aktifitas konfigurasi system yang terintegrasi menjadikan salah satu keunggulan dari OSSIM alienvault, dan bentuk distribusinya dalam sebuah suatu installer yang meliputi komponen sistem operasi dan juga aplikasi pendukung. AlienVault installer didasarkan pada sistem operasi Debian GNU/Linux dan tersedia untuk arsitektur 32 dan 64 bit.

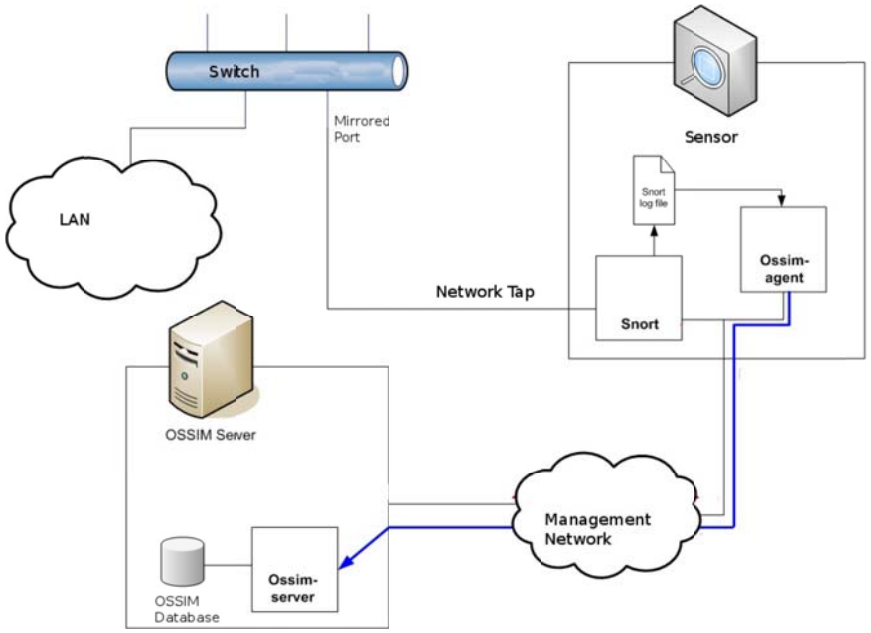
Prosesor dengan desain arsitektur 64 bit menjadi rekomendasi karena pengguna dapat memanfaatkan keuntungan dari arsitektur ini dalam hal kinerja. Dalam bagian tertentu dari instalasi dan tergantung pada lalu lintas dan jumlah peristiwa harus diperlakukan adalah kebutuhan untuk perangkat keras yang mampu menangani volume data yang besar. Arsitektur 64-bit juga memungkinkan penggunaan sejumlah besar memori fisik.

2.3. Skema Monitoring Jaringan

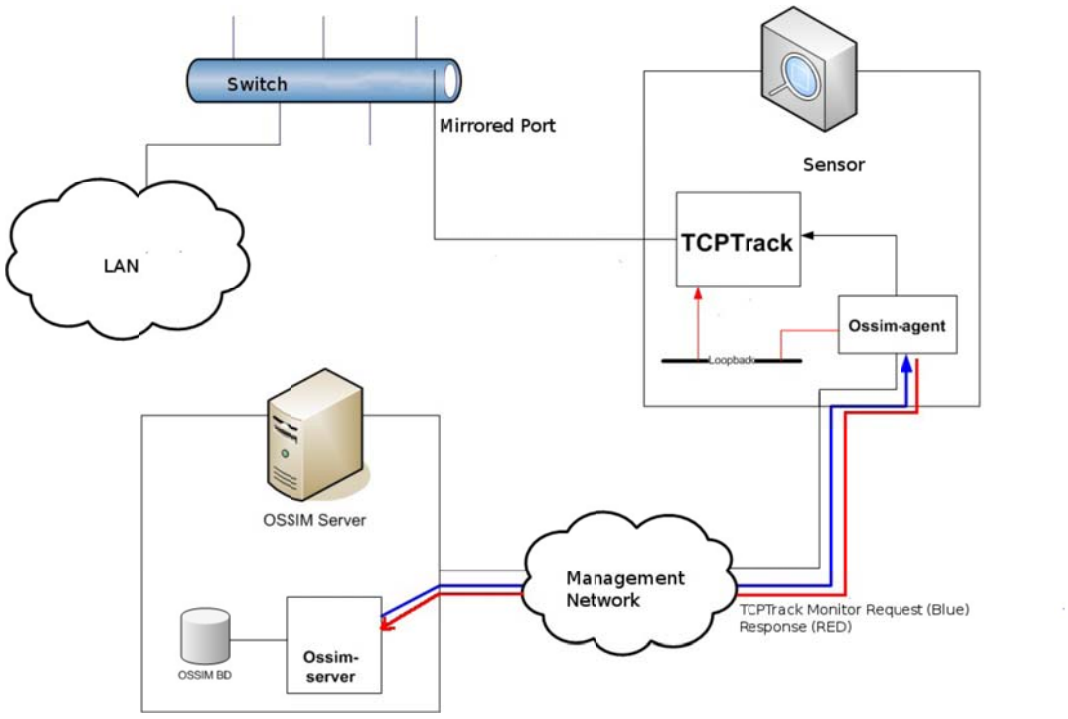
Cara kerja sistem OSSIM dalam mendapatkan data menggunakan sensor-sensor yang telah terintegrasi. Terdapat tiga (3) cara utama dalam tahap pengumpulan data dalam jaringan. Yang pertama adalah melakukan pemrosesan data seperti syslog seperti yang terlihat pada gambar di bawah ini:



Kemudian yang kedua melalui pemantauan network secara pasif pada suatu segment network menggunakan perangkat lunak atau software untuk melakukan monitoring lalu lintas jaringan seperti SNORT melalui suatu antarmuka promiscuous.



Untuk mekanisme ke tiga dari cara kerja sensor OSSIM adalah seperti terlihat pada gambar di bawah ini dengan menggunakan perangkat atau tool yang dapat melakukan query seperti halnya aplikasi tcpwatch.



2.4. Operasi Dasar

Keamanan peristiwa dihasilkan oleh berbagai aplikasi atau perangkat yang penulis miliki dalam jaringan. Kejadian-kejadian ini dikumpulkan dan dibakukan oleh sensor AlienVault, yang juga bertanggung jawab untuk mengirimkan data-data tersebut ke server pusat. Dalam tampilan AlienVault dapat memiliki sensor sebanyak yang Di butuhkan. Sebagai contoh, sebuah sensor dapat ditempatkan dalam DMZ, sensor di setiap kota atau menghabiskan sensor untuk memantau setiap jaringan perusahaan.

Sensor AlienVault termasuk satu set alat (Snort, ntop, Tcptrack, Arpwatch …) yang menganalisis semua lalu lintas jaringan dalam mencari masalah keamanan dan anomaly dari kejadian atau perilaku dari jaringan. Untuk memanfaatkan fungsi ini AlienVault dapat melihat semua lalu lintas pada jaringan, baik menggunakan sebuah hub, atau menyiapkan sebuah pencerminan rentang port atau port dalam jaringan elektronik.

Sensor selanjutnya mengirim aktifitas dan lalu lintas jaringan ke server tunggal AlienVault, yang kemudian melakukan penilaian risiko untuk setiap peristiwa, dan juga akan menempatkan proses korelasi. Setelah kedua proses telah terjadi, peristiwa disimpan dalam database AlienVault. Untuk mengakses informasi ini, seperti halnya konfigurasi sistem dan serangkaian metrik dan laporan penulis akan menggunakan Web Konsol AlienVault. Dari Web konsol juga akan memiliki akses ke informasi real-time ke sejumlah aplikasi yang akan memudahkan analisa negara keseluruhan jaringan penulis.

3. Instalasi Profil

Profil pada fungsi dari host baru dalam distribusi AlienVault dapat secara bebas melakukan perubahan konfigurasi profil yang digunakan. Ini dapat dikonfigurasi saat instalasi atau setelah instalasi. Secara default instalasi otomatis akan memungkinkan semua profil pada komputer yang sama.

3.1. Sensor

Profil sensor akan memungkinkan baik deteksi dan Kolektor AlienVault. Detektor berikut ini diaktifkan secara default:

- Snort

- Ntop
- Openvas
- P0f (Pasif OS Detection)
- Pads
- Arpwatch (Anomali perubahan mac)
- OSSEC (IDS untuk tingkat Host)
- Osiris
- Nagios
- OCS (Inventori)

Setelah sensor profil telah diaktifkan, dapat menonaktifkan detektor sehingga hanya menjaga fungsi koleksi. Untuk mendapatkan manfaat dari kemampuan deteksi alat-alat ini, penulis harus mengkonfigurasi jaringan sensor sehingga AlienVault:

- Memiliki akses ke jaringan yang dimonitor:
- Vulnerability Scanner, Access Control, WMI Agen, Syslog.
- Menerima semua lalu lintas jaringan. Perlu mengkonfigurasi port mirroring atau portspan atau menggunakan hub (HUB).
- Snort, ntop, Arpwatch, Fujos Generasi, Pads, P0f …

Profil Sensor perlu dilakukan konfigurasi agar sistem yang akan digunakan untuk monitoring siap untuk menerima *event* dari host remote menggunakan protokol Syslog. Setiap aplikasi atau perangkat akan memiliki konektor plug terkait yang menetapkan cara mengumpulkan peristiwa aplikasi atau perangkat, serta bagaimana *event* harus normal kembali sebelum mengirim data-data ke server pusat.

Distribusi AlienVault dapat memiliki sensor sebanyak yang dibutuhkan, pada dasarnya dalam hal jaringan yang sedang dipantau dan pada distribusi geografis dari organisasi yang akan dipantau menggunakan AlienVault. Umumnya, harus mengkonfigurasi jaringan sensor, tapi jika menginstal lebih dari satu antarmuka

jaringan dan lalu lintas rute atau mengkonfigurasi port-mirroring, pengguna dapat melakukan pemantauan lebih dari satu jaringan pada sensor yang sama.

3.2. Server

Fasilitas ini menggabungkan profil Siem dan komponen Logger. Sensor yang terhubung ke server untuk mengirim *event* standar AlienVault. Penyebaran lebih kompleks dapat memiliki lebih dari satu server dengan peran yang berbeda.

3.3. Persyaratan

3.3.1. Kebutuhan Hardware

Persyaratan perangkat keras untuk menginstal AlienVault sangat tergantung pada jumlah kejadian per detik dan bandwidth jaringan untuk menganalisis. Sebagai syarat minimal selalu dianjurkan untuk memiliki minimal memori fisik (RAM) sebesar 2 GB, sehingga mampu menganalisa jumlah *event* yang harus diproses server atau jumlah data yang disimpan dalam database. Untuk mengoptimalkan pemanfaatan sumber daya penting pengguna dapat secara selektif mengaktifkan hanya aplikasi dan komponen yang ada.

Perbedaan kinerja antara 32 bit dan 64 bit lebih dari cukup, jadi penulis selalu mencoba untuk memilih arsitektur ini saat memilih perangkat keras. Kebanyakan komponen AlienVault multithreaded, sehingga menggunakan prosesor multi-core juga mendapatkan kemajuan besar dalam kinerja.

Pemilihan kartu jaringan untuk menangkap lalu lintas, dapat mencoba untuk memilih yang didukung oleh driver e1000. Pengembangan driver ini memastikan kompatibilitas yang baik kartu jaringan dengan Debian GNU / Linux.

4. Instalasi

4.1. **Langkah demi langkah instalasi Otomatis**

Proses instalasi otomatis akan menginstal versi open source dari AlienVault profil semua-dalam-satu diaktifkan. Setelah instalasi selesai, pengguna secara manual refresh untuk mendapatkan manfaat dari versi Profesional AlienVault. Instalasi dilakukan hampir tanpa campur tangan pengguna. Instalasi otomatis dikonfigurasi dengan keymap AS dan semua teks dalam bahasa Inggris.

Network Setup

Pada bagian ini, perlu untuk mengkonfigurasi konfigurasi kartu jaringan, kemudian menggunakan alamat IP dengan akses internet selama proses instalasi. Alamat IP ini akan digunakan dalam antarmuka manajemen. Masukkan alamat IP dan klik ” Continue. “

netmask untuk digunakan dengan jaringan . Masukkan mask jaringan dan klik ” Continue. “

Alamat IP dari default gateway ke rute, jika jaringan memiliki gateway. Masukkan alamat IP dari default gateway dan klik ” Continue. “

Komputer yang diinstall system ini harus terhubung dalam jaringan server DNS (Domain Name Service). Jika memiliki sebuah server nama lokal di jaringan harus menjadi yang pertama dalam konfigurasi ini. dapat memasukkan banyak nameserver seperti yang inginkan. Masukkan alamat IP dari DNS (dipisahkan dengan spasi) dan klik ” Continue. “

Partisi disk

Proses pembuatan partisi pada harddisk dan ini akan menghapus data yang tersimpan pada hard drive .

Pilih ” Guided: use entire disk “dan klik” Continue. “

Jika komputer memiliki beberapa disk, pilih disk yang akan diinstal AlienVault dan klik Lanjutkan. Jika komputer memiliki satu disk cukup klik ” Continue “.


alienvault
creators of ossim
Partition disks
Note that all data on the disk you select will be erased, but not before you have confirmed that you really want to make the changes.
Select disk to partition:
SCSI1 (0,0,0) (sda) - 21.5 GB VMware, VMware Virtual S
Screenshot
Go Back
Continue

Mengatur penggunaan dan password

Setelah sistem dasar terinstal, installer akan memungkinkan untuk mengkonfigurasi account root. account pengguna lain dapat dibuat setelah instalasi selesai. Setiap password yang buat harus minimal 6 karakter dan harus berisi kedua karakter huruf besar dan huruf kecil dan karakter t baca. Berhati-hatilah ketika pengaturan password root , karena ini adalah rekening yang kuat. Hindari kamus kata-kata atau penggunaan setiap informasi pribadi yang dapat ditebak.

Masukkan password root dan klik ” Continue. “

Update instalasi

Instalasi dapat dihubungkan ke website AlienVault untuk mendownload versi terbaru dari setiap paket perangkat lunak yang disertakan dalam Siem AlienVault Profesional. Proses ini bisa memakan waktu hingga satu jam (tergantung pada koneksi internet). Bersabarlah dan tidak membatalkan proses ini.

Pilih ” Ya “dan klik” Continue. ” Setelah menginstal sistem akan diatur ulang ke baru AlienVault sistem.

Proses instalasi selesai, seperti yang terlihat pada gambar berikut:

5. Pengaturan

5.1. Konfigurasi Sistem

Untuk mempermudah konfigurasi jumlah besar alat yang termasuk dalam instalasi AlienVault konfigurasi terpusat dalam satu file. Setiap kali mengubah file ini harus menjalankan sebuah perintah yang akan bertanggung jawab untuk implementasi ini konfigurasi terpusat untuk masing-masing alat termasuk dalam AlienVault. File utama konfigurasi adalah sebagai berikut:

/etc/ossim/ossim_setup.conf

Pengguna dapat mengedit file tersebut dengan editor teks (vim, nano, pico …) atau menggunakan sebuah antarmuka untuk mengelola file konfigurasi. Untuk mulai antarmuka ini menggunakan perintah berikut:

ossim-setup

Untuk menerapkan perubahan pada file konfigurasi dan menghasilkan file-file konfigurasi untuk semua alat terpasang harus jalankan perintah berikut:

ossim-reconfig

5.2. Ubah Profil

Instalasi default memungkinkan semua profil pada mesin dipasang. Untuk mengubah lebar dari mesin harus mengeksekusi skrip OSSIM-setup dan memilih pilihan kedua (Ubah Profil Pengaturan). Tergantung pada profil yang penulis pilih untuk mengatur beberapa parameter atau orang lain:

All in One

a. Pilih antarmuka: Masukkan interface (dipisahkan dengan koma) yang mendapatkan semua lalu lintas jaringan.
b. Jaringan Profil: Masukkan jaringan (domestik) dalam format CIDR, dipisahkan dengan koma, bahwa sensor akan dapat melihat bagaimana panggilan interface-nya (misalnya, 192.168.0.0/24, 10.0.0.0 / 😎
c. Sensor AlienVault Nama: Nama yang diberikan kepada sensor yang diinstal pada mesin ini.
d. Pilih plugin: Pilih aksesoris yang harus diaktifkan untuk sensor ini. Plug-aktif hanya berdasarkan permintaan server AlienVault untuk korelasi dimonitor. plugin Detektor mengumpulkan peristiwa nyata-waktu, database file, soket ..

Sensor

a. Sensor AlienVault Nama: Nama yang diberikan kepada sensor yang diinstal pada mesin ini.
b. Pilih antarmuka: Masukkan interface (dipisahkan dengan koma) yang mendapatkan semua lalu lintas jaringan.
c. Jaringan Profil: Masukkan jaringan (domestik) dalam format CIDR, dipisahkan dengan koma, bahwa sensor akan dapat melihat bagaimana panggilan interface-nya (misalnya, 192.168.0.0/24, 10.0.0.0 / 😎
d. AlienVault Server Ip Address: Masukkan alamat IP dimana server mendengarkan AlienVault.
e. Pilih plugin: Pilih aksesoris yang harus diaktifkan untuk sensor ini. Plug-aktif hanya berdasarkan permintaan server AlienVault untuk korelasi dimonitor. plugin Detektor mengumpulkan peristiwa nyata-waktu, database file, soket ..

Server

a. Ip Address AlienVaul Mysql Server: Masukkan alamat IP komputer yang menjalankan database profil AlienVault. Pastikan memiliki database permanen yang tepat untuk terhubung dari mesin remote.
b. AlienVault Mysql server port: port mendengarkan mysql. (Default 3306)
c. AlienVault Mysql password: Password untuk root pada server MySQL.

Database

a. AlienVault Mysql password: Password untuk root pada server MySQL.
b. Jika hanya ingin mengkonfigurasi ulang profil saat ini, pilih profil yang digunakan dan juga akan diminta untuk memasukkan parameter konfigurasi.
c. Untuk menerapkan perubahan harus memilih (Terapkan dan menyimpan semua perubahan), atau menjalankan OSSIM-reconfig perintah.

Network Setup

Komputer menjalankan AlienVault memerlukan perawatan khusus dalam mendirikan jaringan.

a. Konfigurasi jaringan didefinisikan di file berikut:
   /etc/network/interfaces
b. Jika konfigurasi jaringan telah berubah, untuk menerapkan perubahan gunakan perintah berikut:
   /etc/init.d/networking restart

Setiap tim harus memiliki minimal satu alamat IP statis untuk komponen AlienVault yang dapat berkomunikasi satu sama lain dan administrator dari jarak jauh dapat mengakses mesin. Setiap antarmuka dengan alamat IP harus memiliki sebuah entri di file /etc/network/interface dengan cara sebagai berikut:

allow-hotplug eth0
iface eth0 inet static
address 192.168.1.133
netmask 255.255.0.0
network 192.168.0.0
broadcast 192.168.255.255
gateway 192.168.1.1
dns-nameservers 192.168.1.1

Antarmuka ini digunakan untuk mengumpulkan semua lalu lintas jaringan tidak harus merupakan alamat IP. Interface promiscuous tidak memerlukan konfigurasi khusus pada file konfigurasi jaringan.

5.3. Update AlienVault

Perintah berikut update sistem AlienVault:

apt-get update, apt-get dist-upgrade; ossim-reconfig

Sistem upgrade perangkat lunak yang digunakan dalam Setup AlienVault dirancang untuk memastikan bahwa versi yang benar yang digunakan. Memungkinkan pengembang untuk memblokir atau memaksa pembaruan AlienVault software tertentu pada sistem . Untuk alasan ini, tidak harus menyertakan sebuah repositori software baru di **/etc/apt/sources.list**

5.4. Versi Profesional

Selain mengembangkan versi opensource, AlienVault juga mengembangkan versi profesional yang memiliki nama AlienVault Profesional Siem. Rilis ini memperkenalkan sejumlah perbaikan pada fungsionalitas dan kinerja antara lain sebagai berikut:

a. Logger: Mass Storage. Log adalah memiliki tanda tangan digital sehingga dapat digunakan sebagai bukti ahli.

b. Skalabilitas
c. Yield: 30 kali kinerja versi open source.
d. Reliabilitas: redundansi dan ketersediaan tinggi.

5.5. OSSIM Server Configuration

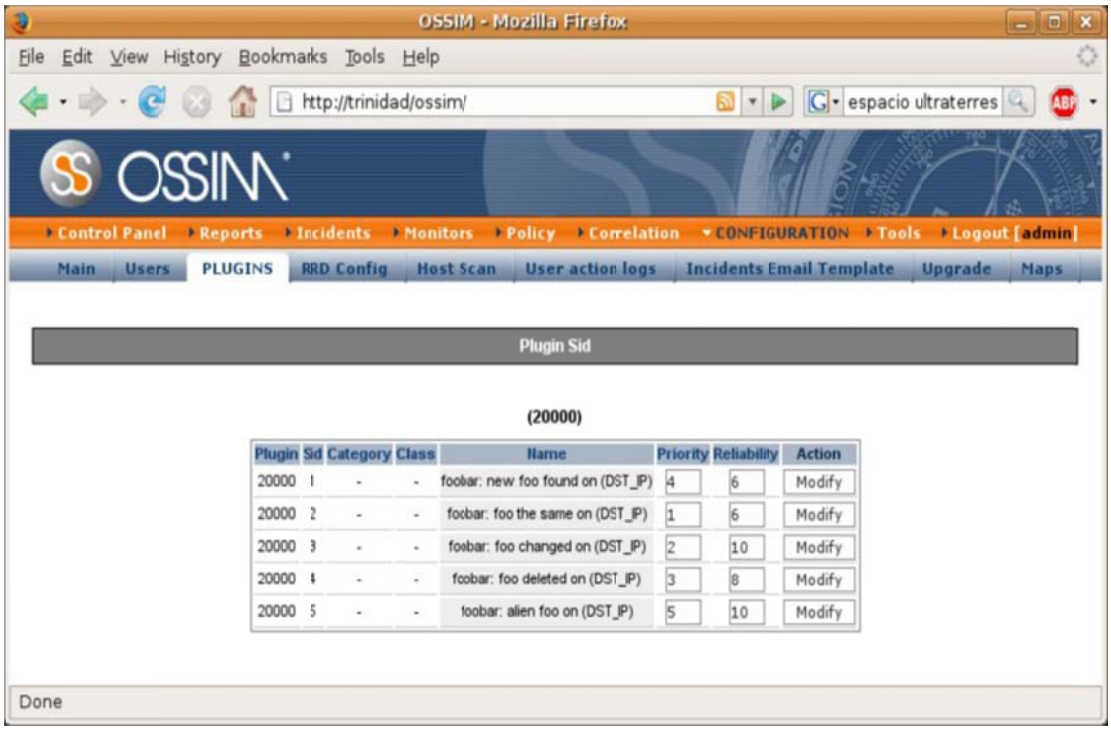
Tutorial berikut ini adalah contoh skrip SQL yang merupakan tahapan dalam contoh bagaimana membuat plugin secara khusus pada OSSIM Server, dimana diperlukan proses update database dengan informasi yang dibutuhkan oleh plugin tersebut.

```
cat > ./foobar.sql << __END__
-- foobar
-- plugin_id: 20000
--
-- \$Id:\$
--
DELETE FROM plugin WHERE id = "20000";
DELETE FROM plugin_sid where plugin_id = "20000";
INSERT INTO plugin (id, type, name, description) VALUES (20000, 1, 'foobar',
'Foobar demo detector');
INSERT INTO plugin_sid (plugin_id, sid, category_id, class_id, reliability,
priority, name) VALUES (20000, 1, NULL, NULL, 6, 4, 'foobar: new foo found on
(DST_IP)');
INSERT INTO plugin_sid (plugin_id, sid, category_id, class_id, reliability,
priority, name) VALUES (20000, 2, NULL, NULL, 6, 1, 'foobar: foo the same on
(DST_IP)');
INSERT INTO plugin_sid (plugin_id, sid, category_id, class_id, reliability,
priority, name) VALUES (20000, 3, NULL, NULL, 10, 2, 'foobar: foo changed on
(DST_IP)');
INSERT INTO plugin_sid (plugin_id, sid, category_id, class_id, reliability,
priority, name) VALUES (20000, 4, NULL, NULL, 8, 3, 'foobar: foo deleted on
(DST_IP)');
INSERT INTO plugin_sid (plugin_id, sid, category_id, class_id, reliability,
priority, name) VALUES (20000, 5, NULL, NULL, 10, 5, 'foobar: alien foo on
(DST_IP)');
__END__
```

Setelah selesai, plugin yang baru dibuat dapat dimasukkan ke dalam OSSIM Server menggunakan perintah berikut: **cat foobar.sql | mysql -u root -p ossim**

kemudian lakukan proses restart OSSIM Server dengan menjalankan perintah berikut: **/etc/init.d/ossim-server restart**

Plugin yang berhasil dimasukkan ke dalam sistem, dapat dicek me5lalui tampilan web browser, dengan mengklik menu **Configuration→Plugins**, seperti yang terlihat pada gambar di bawah ini.



Disarankan untuk merubah nilai dari konfigurasi yang ada, sehingga dapat mempengaruhi reliabilitas dan prioritas pada perlakuan plugin oleh sistem. Nilai yang dirubah pada bagian plugin_sid. Setelah melakukan perubahan, jangan lupa merestart OSSIM Server untuk mendapatkan hasilnya.

6. Saran dan Rekomendasi Umum

Dalam lingkungan produksi selalu disarankan untuk menggunakan arsitektur 64-bit, karena ada perbedaan besar dalam kinerja dibandingkan dengan 32 bit. Kemudian penulis mencoba untuk tidak pernah memasang sensor dalam lingkungan virtual karena cara di mana alat-alat ini virtualisasi mengelola antarmuka jaringan, yang menyebabkan sejumlah besar lalu lintas jaringan yang hilang tanpa dianalisis.

Tidak disarankan untuk memasang perangkat lunak yang memerlukan untuk mengubah atau menambah entri baru dalam file disimpan dalam repositori perangkat lunak (/etc/apt/sources.list)

AlienVault akan selalu mendukung versi stabil terbaru dari Debian GNU / Linux. Jika versi baru dari pengembang Debian bebas untuk memberikan panduan tentang bagaimana meng-upgrade ke versi baru.

Tidak ada batasan pada perangkat lunak yang dapat diinstal pada mesin tapi perlu diingat pemakaian memori tinggi dan CPU untuk beberapa aplikasi untuk menginstal perangkat lunak baru.

7. Lampiran

AlienVault Update Procedure

Introduction

Package upgrades are a great success of the APT system. APT, is a free user interface that works with core libraries to handle the installation and removal of software on the Debian GNU/Linux distribution and its variants (Such as Ubuntu). APT simplifies the process of managing software on Unix-like computer systems by automating the retrieval, configuration and installation of software packages.

AlienVault uses Debian as its base operating system. Both AlienVault Software and the operating system software are updated using APT.

Important Information

AlienVault has a complex system of package dependencies and preferences between the different packages as some of them will be available in different software repositories. For this reason the file /etc/apt/sources.list should never be modified. This could break the dependencies tree and make your system unstable.

Requirements

Internet access is required in the machine that is going to be updated.

It is also important to check before starting the installation that there is enough free disk space (At least 1GB free) that allows to download and install the new updated software.

Update Procedure

To update an AlienVault installation you need to execute the following commands:

```
# apt-get update
```

```
# apt-get dist-upgrade
```

```
# ossim-reconfig
```

The apt-get program uses this database to find out how to install packages requested by the user and to find out which additional packages are needed in order for a

selected package to work properly. To update this list, you would use the command **apt-get update** . This command looks for the package lists in the archives found in /etc/apt/sources.list (This file should never be modified unless requested by the AlienVault Team)
It's a good idea to run this command regularly to keep yourself and your system informed about possible package updates, particularly security updates.

Package upgrades can be achieved with a single command: **apt-get dist-upgrade**
ossim-reconfig generates all configuration files for the different AlienVault components based on the configuration stored in the two main configuration files of an AlienVault installation:

- **/etc/ossim/ossim-setup.conf:** AlienVault Profile configuration, plugins enabled/disabled, database connection parameters..
- **/etc/network/interfaces:** Network Configuration

Snort Rules

Snort rules are updated using the apt system. Users who prefer to update rules more often as well as those users that have paid a subscription or have written their own rules must follow this procedure in order to update the information in the AlienVault Database for the new Snort rules.

```
# perl /usr/share/ossim/scripts/create-sidmap.pl /etc/snort/rules/
```

This command must be executed in the box running the AlienVault Database and the directory passed as a parameter must contain the updated rules. After running this command go to the box running the AlienVault Server Profile (SIEM or Logger) and restart the AlienVault Server by running the following command:

```
# /etc/init.d/ossim-server restart
```

OpenVas / Nessus Rules

OpenVas rules are also updated using the apt system. Users who prefer to update rules more often as well as those users that have paid a subscription or have written their own rules must follow this procedure in order to update the information in the AlienVault Database for the new OpenVas scanning signatures.

Download the new rules:

```
#openvas-nvt-sync --wget
```

Restart OpenVas Scanner (This can take up to 20 minutes)

```
# /etc/init.d/openvas-scanner restart
```

Update the information in the Database (This command must be executed in the box running the AlienVault Database Profile):

```
# perl /usr/share/ossim/scripts/vulnmeter/updateplugins.pl migrate
```

Upgrade Validation

In case the upgrade fails it is always recommended running **ossim-reconfig** before troubleshooting. ossim-reconfig will reconfigure and restart theThings to do in order to check that the upgraded system is running fine:
- Connect to the AlienVault Web Interface - Check the Log files

1. AlienVault Agent: /var/log/ossim/agent.log
2. AlienVault Server (SIEM & Logger): /var/log/ossim/server.log
3. Syslog: /var/log/syslog

Removing unused package files

apt-get clean removes everything except lock files from /var/cache/apt/archives/ and /var/cache/apt/archives/partial/. Thus, if you need to reinstall a package APT should retrieve it again.
apt-get autoclean removes only package files that can no longer be downloaded.

Troubleshooting

Upgrade interrupted

If the upgrade process was interrupted for some reason try running the following commands again:

```
# apt-get update
```

```
# apt-get dist-upgrade
```

```
# ossim-reconfig
```

If the upgrade process was interrupted while packages were being configured you will get the following error

```
E: dpkg was interrupted, you must manually run 'dpkg --configure -a' to correct the problem.
```

To fix this execute the following command:

```
# dpkg --configure -a
```

After running this command execute ossim-reconfig:

```
# ossim-reconfig
```

In case none of the previous procedures copy the output of the previous commands and post it in the AlienVault Community Forums or contact the AlienVault Support Team in case you have paid support service.

Unmet dependencies

If the upgrade fails for some reason (E.g.: Internet connection failure during the upgrade process) you may get an error indicating that the system has unmet dependencies:

```
Reading package lists... Done
Building dependency tree
Reading state information... Done
You might want to run `apt-get -f install' to correct these.
The following packages have unmet dependencies:
  ossim-framework-daemon: Depends: ossim-framework (= 1:2.5.3-165) but 1:2.5.3-143 is installed
E: Unmet dependencies. Try using -f
```

In case you get this error run the following command:

```
# apt-get -f dist-upgrade
```

Take a look to the changes that will be performed during the upgrade (Make sure no important packages will be removed) and enter **Y** to continue the upgrade process.

```
opensourcesim:~# apt-get -f dist-upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
```

```
Correcting dependencies... Done
Calculating upgrade... Done
The following packages will be upgraded:
  alienvault-multitenancy alienvault-wizard ossim-agent ossim-compliance ossim-
contrib ossim-framework ossim-mysql ossim-server
8 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
2 not fully installed or removed.
Need to get 0B/104MB of archives.
After this operation, 32.8kB of additional disk space will be used.
Do you want to continue [Y/n]?
```

Execute ossim-reconfig:

```
# ossim-reconfig
```

OSSIM Agent Configuration

The following steps detail configuration of the agent for the plugin. This plugin is going to monitor syslog for the output, so a config file for the plugin must exist containing the plugin ID and how to match information in syslog. In this case, it matches only one sid, but as you can see from the above sql, there could be five patterns and five sub ids.
Contents of /etc/ossim/agent/plugins/foobar.cfg You can copy and paste into the shell. If you create the file manually, be sure to remove the backslashes before any '$' symbol.

```
cat > /etc/ossim/agent/plugins/foobar.cfg << __END__
;; foobar
;; plugin_id: 20000
;; type: detector
;; description: foobar demo plugin
;;
;; URL:
;;
;; \$Id:\$
[DEFAULT]
plugin_id=20000
[config]
type=detector
```

```
enable=yes
source=log
location=/var/log/user.log
create_file=false
process=
start=no
stop=no
startup=
shutdown=
event_type=event
regexp="(\S+\s+\d+\s+\d\d:\d\d:\d\d)\s+(?P<dst_ip>[^\s]*).*?FOOBAR.*?new foo
found"
plugin_sid=1
dst_ip={resolv(\$dst_ip)}
src_ip=0.0.0.0
date={normalize_date(\$1)}
[foobar - foo the same]
# Sep 7 12:40:55 eldedo FOOBAR[2054]: foo the same
event_type=event
regexp="(\S+\s+\d+\s+\d\d:\d\d:\d\d)\s+(?P<dst_ip>[^\s]*).*?FOOBAR.*?foo the
same"
plugin_sid=2
dst_ip={resolv(\$dst_ip)}
src_ip=0.0.0.0
date={normalize_date(\$1)}
[foobar - New changed]
# Sep 7 12:40:55 eldedo FOOBAR[2054]: foo changed
event_type=event
regexp="(\S+\s+\d+\s+\d\d:\d\d:\d\d)\s+(?P<dst_ip>[^\s]*).*?FOOBAR.*?foo
changed"
plugin_sid=3
dst_ip={resolv(\$dst_ip)}
src_ip=0.0.0.0
date={normalize_date(\$1)}
[foobar - New deleted]
# Sep 7 12:40:55 eldedo FOOBAR[2054]: foo deleted
event_type=event
regexp="(\S+\s+\d+\s+\d\d:\d\d:\d\d)\s+(?P<dst_ip>[^\s]*).*?FOOBAR.*?foo
deleted"
plugin_sid=4
dst_ip={resolv(\$dst_ip)}
src_ip=0.0.0.0
date={normalize_date(\$1)}
```

```
[foobar - alien foo]
# Sep 7 12:40:55 eldedo FOOBAR[2054]: alien foo
event_type=event
regexp="(\S+\s+\d+\s+\d\d:\d\d:\d\d)\s+(?P<dst_ip>[^\s]*).*?FOOBAR.*?alien foo"
plugin_sid=5
dst_ip={resolv(\$dst_ip)}
src_ip=0.0.0.0
date={normalize_date(\$1)}
__END__
```

We need to tell the agent that we have a new plugin. Edit the file /etc/ossim/agent/config.cfg and add the following line in the [plugin] section.
foobar=/etc/ossim/agent/plugins/foobar.cfg

Now to restart the agent so that it is aware of the new plugin information.
/etc/init.d/ossim-agent restart

Verification
This is a sample python script that will send a message to syslog. I parses the optios sent and sends a log message for each option that matches the case. The following code can be run as a script on any host that has Python installed.

```
#! /usr/bin/python
import syslog
import sys
syslog.openlog("FOOBAR", syslog.LOG_PID , syslog.LOG_USER )
for arg in sys.argv:
if arg == "1":
syslog.syslog(syslog.LOG_WARNING, "new foo found")
elif arg == "2":
syslog.syslog(syslog.LOG_WARNING, "foo the same")
elif arg == "3":
syslog.syslog(syslog.LOG_WARNING, "foo changed")
elif arg == "4":
syslog.syslog(syslog.LOG_WARNING, "foo deleted")
elif arg == "5":
syslog.syslog(syslog.LOG_WARNING, "alien foo")
syslog.closelog()
```

Run this program on the server for which you want to generate the event. The following will send the first type syslog message.

A sample OSSIM directive

OSSIM stores its rules on the server in a file named /etc/ossim/server/directives.xml. The rules are separated into directives. The following is an example ssh brute force directive. This rules from this directive obtains its information from the ssh auth.log plugin. In this case, the attacker could be switching different hosts to attack in attempt to escape detection on a single host, but this directive will detect those attempts between switched target hosts as well. The reliability begins at 3 after three failed attempts. Three more will raise it to 4. Five more will raise it 6, and then an additional 10 attempts will raise it to 8.

<directive id="20" name="Possible SSH brute force login attempt against DST_IP" priority="5">
<rule type="detector" name="SSH Authentication failure" reliability="3" occurrence="1" from="ANY" to="ANY" port_from="ANY" port_to="ANY" time_out="10" plugin_id="4003" plugin_sid="1,2,3,4,5,6">
<rules>
<rule type="detector" name="SSH Authentication failure (3 times)" reliability="+1" occurrence="3" from="1:SRC_IP" to="ANY" port_from="ANY" time_out="15" port_to="ANY" plugin_id="4003" plugin_sid="1,2,3,4,5,6" sticky="true">
<rules>
<rule type="detector" name="SSH Authentication failure (5 times)" reliability="+2" occurrence="5" from="1:SRC_IP" to="ANY" port_from="ANY" time_out="20" port_to="ANY" plugin_id="4003" plugin_sid="1,2,3,4,5,6" sticky="true">
<rules>
<rule type="detector" name="SSH Authentication failure (10 times)" reliability="+2" occurrence="10" from="1:SRC_IP" to="ANY" port_from="ANY" time_out="30" port_to="ANY" plugin_id="4003" plugin_sid="1,2,3,4,5,6" sticky="true">
</rule>
</rules>
</rule>
</rules>
</rule>
</rules>
</rule>
</directive>

The above directive only explored rules that are sensors. You You in his paper walks through an attack with a sample DCOM exploit (YouYou). Dominique Karg also goes through the meaning of the details for the XML syntax such as sticky .