Keamanan Jaringan

Onno W. Purbo

onno@indo.net.id
onno.purbo@xecureit.id
@onnowpurbo
XECUREIT.id

Daftar Isi

Kata Pengantar

Buku keamanan jaringan ini ditulis untuk pegangan teman2 pemula yang ingin mengimplementasikan keamanan pada jaringan IntraNet maupun Internet-nya.

Buku ini merupakan copy paste dari kumpulan tulisan-tulisan saya di wiki.

Semoga bisa bermanfaat.

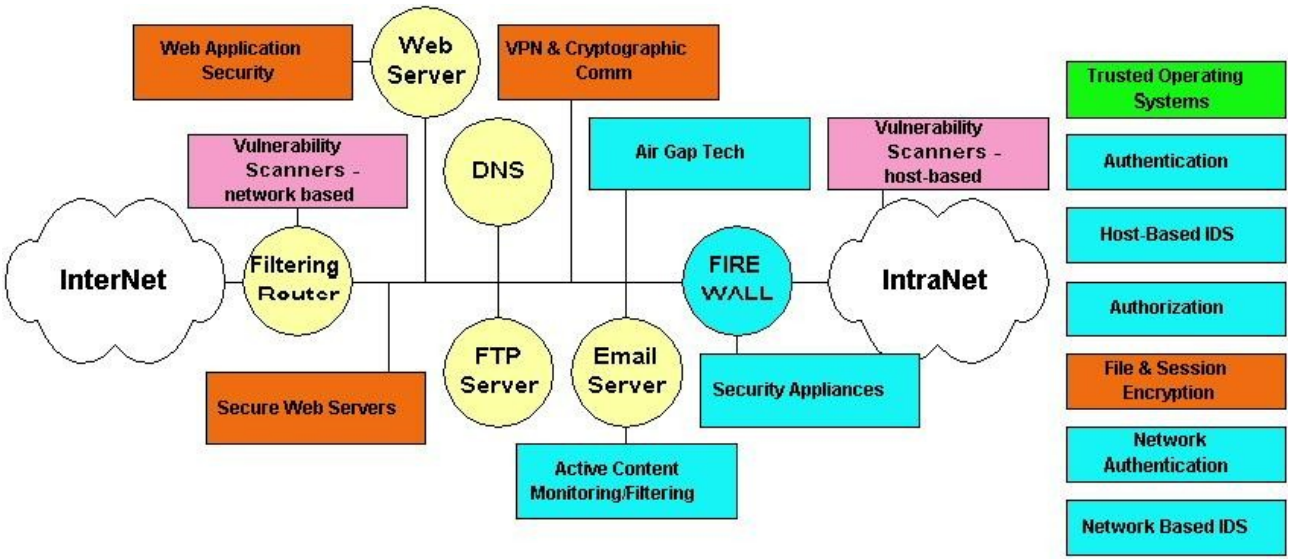
Jakarta, 5 Oktober 2017

Penulis

BAB 1 Gambaran Umum Network Security

Peta Teknologi Network Security

Network security menjadi sebuah pengetahuan yang wajib di miliki bagi mereka yang ingin secara serius berkiprah di Internet. Sialnya, teknologi telah berkembang sedemikian kompleks sehingga menuntut profesional network security untuk mempelajari banyak hal untuk betul-betul mengerti keseluruhan konsep & teknologi network security. Untuk memudahkan proses belajar, ada baiknya memperhatikan baik-baik gambar yang terlampir yang berisi peta teknologi network security. Referensi yang sangat baik tentang hal ini terdapat di http://www.sans.org.



Secara umum topologi jaringan komputer terdiri dari jaringan Internet publik yang menyebar ke seluruh dunia dan jaringan Intranet yang terdapat internal di perusahaan / institusi. Di antara InterNet dan IntraNet biasanya terdapat De-Militerized Zone (DMZ) yang di batasi oleh Filtering Router ke arah Internet, dan Firewall ke arah IntraNet. Pada De-Militerized Zone (DMZ) ini biasanya di pasang berbagai server, seperti, Mail Server, FTP Server, Web Server dan DNS Server.

Berdasarkan topologi jaringan di atas, kita dapat membagi teknologi network security tersebut menjadi empat (4) bagian besar, yaitu:

- Penetration testing
- Certificate Authority / PKI
- Vulnerability Testing
- Managed Security Services

Mari kita lihat teknologi yang menjadi bagian dari ke empat (4) bagian ini, secara umum,

Penetration Testing, terdiri dari:

- Active Content Monitoring / Filtering, biasanya di letakan di mail server di DMZ.
- Intrusion Detection - Host Based, biasanya di letakan di server di IntraNet maupun DMZ.

- Firewall, menjadi perantara IntraNet dengan DMZ dan InterNet.
- Intrusion Detection - Network Based, biasanya digunakan untuk memonitor IntraNet.
- Authorization, di jalankan di IntraNet.
- Air Gap Technology, di jalankan di De-Militerized Zone (DMZ).
- Network Authentication, di operasikan di IntraNet.
- Security Appliances, biasanya berbentuk hardware Firewall.
- Security Services: Penetration Testing, perusahaan di luar yang memberikan servis kepada kita.
- Authentication, dioperasikan di IntraNet.

Certificate Authority / PKI, merupakan pendukung teknologi yang lain & dapat dioperasikan di server di IntraNet, terdiri dari:

- Certificate Authority, di IntraNet maupun InterNet.
- File & Session Encryption, di operasikan di IntraNet
- VPN & Cryptographic Communications, di mulai di De-Militerized Zone dan digunakan untuk menembus ke Internet menuju IntraNet yang lain.
- Secure Web Servers, di operasikan di De-Militerized Zone (DMZ).
- Single Sign On, di server.
- Web Application Security, di Web server.

Vulnerability Testing, biasanya dilakukan oleh auditor atau security manager, antara lain adalah.

- Vulnerability Scanners - Host Based, di operasikan di server IntraNet
- Real-Time Security Awareness, Response & Threat Management, digunakan oleh security manager.
- Vulnerability Scanners - Network Based, di operasikan di filtering router yang terhubung langsung ke InterNet.

Managed Security Services, merupakan bagian manajemen (non-teknis) pendukung network security. Isu yang ada antara lain adalah:

- Enterprise Security Policy Implementation.
- Managed Security Services.
- Enterprise Security Administration.
- Security Services: Policy Development.
- Trusted Operating Systems, di install di semua komputer.
- Anti D.D.O.D Tools.

Selanjutnya, mari kita lihat berbagai konsep yang ada dengan penjelasan lebih detail.

Penetration Testing

- Active Content Monitoring / Filtering. Pada saat anda tersambung ke Internet, anda mengambil resiko dari virus komputer, java / Active-X script jahat dll. Tool ini akan memeriksa semua content yang masuk ke jaringan / komputer, secara kontinu mengupdate library-nya.

- Intrusion Detection - Host Based. Intrusion detection host based akan memonitor file log. Dia akan meresponds dengan alarm atau serangan balasan jika ada usaha user untuk mengakses data, file atau servis yang tidak di ijinkan.
- Firewall. Firewall adalah sebuah sistem atau group dari beberapa sistem yang melaksanakan kebijakan akses control antara dua jaringan.
- Intrusion Detection - Network Based. Network based intrusion detection akan memonitor jaringan dan akan meresponds dengan alarm pada saat dia mengidentifikasi adanya pola traffic yang tidak baik, seperti scanning, usaha denial of service maupun serangan lainnya.
- Authorization. Authentication, bertanya "siapa anda?". Authorization, bertanya "apakah anda berhak?". Dengan mekanisme authorization setiap pengguna yang akan mengakses resource harus memohon ke authorization server untuk memperoleh ijin.
- Air Gap Technology. Hardware/software jenis ini memungkinkan transfer data secara real-time antara Internet dengan back-end tanpa membuka lubang di firewall. Kadang solusi Air Gap mengharuskan secara fisik terjadi pemutusan sambungan ke jaringan luar. Air Gap memutuskan semua protokol jaringan, membatasi akses ke data di lapisan aplikasi saja, serta melakukan analisa content.
- Network Authentication. Tool ini menggunakan beberapa pendekatan untuk memperbaiki kemampuan sistem untuk membedakan antara yang berhak dan yang tidak berhak memperoleh akses.
- Security Appliances. Kombinasi hardware/software yang memberikan servis terbatas, seperti firewall, network load management dll. Karena sistem operasi-nya sangat terbatas, lebih mudah di manage & tidak menjadi sasaran serangan hacker seperti di general purpose UNIX atau Windows NT.
- Security Services: Penetration Testing. Organisasi konsultan yang mensimulasikan serangan hacker di dunia nyata maupun serangan social engineering. Mereka biasanya memberikan advis bagaimana memperbaiki pertahanan. Biasanya mereka menggunakan network-based vulnerability scanning tools.
- Authentication. Authentication adalah sebuah proses yang menentukan sesuatu atau seseorang adalah siapa atau apa. Cara paling sederhana dari proses authentikasi adalah logon password, sialnya sangat rentan untuk di curi. Cara lain untuk mengatasi ini adalah menggunakan token yang memungkinkan proses authentikasi lebih ketat lagi.

Certificate Authority / PKI

- Certificate Authority. Certificate Authority (CA) adalah organisasi yang memberikan dan memanage security credential dan public keys untuk enkripsi & dekripsi berita. Sertifikat yang di manage termasuk public keys yang memperkuat authentikasi, privacy & non-repudiation.
- File & Session Encryption. Enkripsi adalah sebuah proses yang mana data di ubah bentuknya sehingga sulit di buka dan di mengerti oleh orang yang tidak mempunyai authoritas untuk itu. Algoritma komputer yang canggih digunakan dalam proses enkrip & dekrip pada saat di butuhkan.
- VPN & Cryptographic Communications. Virtual Private Network (VPN) memungkinkan komunikasi aman melalui jaringan publik Internet. Hal ini sangat menghemat biaya untuk perusahaan dengan mobile worker atau cabang perusahaan, sehingga komunikasi dapat dilakukan tanpa perlu menggunakan jaringan telepon private yang mahal.
- Secure Web Servers.Tool yang memungkinkan kita memberikan servis web dalam sebuah lingkungan yang di rekayasa supaya lubang keamanan-nya minimal.

- Single Sign On. Paket software yang membantu pengguna agar dapat mengakses ke beberapa komputer tanpa perlu mengingat banyak password. Single Sign On pada dasarnya tidak mengubah proses di bawahnya, tapi menyembunyikan perbedaan yang ada melalui sebuah lapisan software tambahan.
- Web Application Security. Web application security akan memproteksi aplikasi web dan resource yang ada dari ancaman di Internet, seperti, mencuri aset perusahaan, pencurian kartu kredit, deface situs dll. Hal ini dilakukan degangn mendeteksi / menghalangi teknik hacking pada wilayah ini.

Vulnerability Testing

- Vulnerability Scanners - Host Based. Tool untuk menchek setting dari system untuk menentukan apakah sesuai / konsisten dengan kebijakan keamanan perusahaan. Tool ini biasa digunakan oleh auditor.
- Real-Time Security Awareness, Response & Threat Management. RTSA memungkinkan seorang security manager untuk melihat apa yang terjadi di perusahaan yang menggunakan banyak peralatan dari multiple vendor secara real-time melalui sebuah konsol. RTSA menolong mengurangi jumlah personel yang dibutuhkan untuk memonitor banyak peralatan.
- Vulnerability Scanners - Network Based. Software yang dapat mensimulasikan tabiat penyerang dan mempelajari sampai sekitar 600 kemungkinan kelemahan sistem yang sedang di serang.

Managed Security Services

- Enterprise Security Policy Implementation. EPSI memungkinkan manager security untuk mengautomasi setiap langkah keamanan dari console pusat, mulai dari creating, editing, approving, publishing, distribution, education, compliance, reporting dan maintenance. Tool ini akan memaksa sosialisasi, menchek pengertian pegawai, mencatat kejadian, dan mengukur compliance, yang pada akhirnya akan menolong manajemen resiko IT tanpa memberikan banyak beban ke staff yang terbatas.
- Managed Security Services. Vendor yang menawarkan managed security services berasumsi bahwa mereka akan memperoleh beberapa persen kerjaan sebagai outsource. Dengan cara tsb. administrator dapat mengerjakan kerjaan yang lain.
- Enterprise Security Administration. Tool ini mengadministrasi security tingkat enterprise, memastikan bahwa semua user di sebuah enterprise memperoleh hak dan kewajiban yang sama. Sistem ini terutama sangat bermanfaat untuk memberikan akses bagi user baru, dan, yang penting, menghilangkan semua akses bagi pegawai yang sudah keluar.
- Security Services: Policy Development. Konsultan yang membantu pengembangan kebijakan keamanan secara cepat. Mereka umumnya sudah mempunyai template agar kebijakan security dapat di implementasikan dengan cepat, sepertoi penggunaan e-mail yang baik, extranet hingga PKI.
- Trusted Operating Systems. Karena semua mekanisme ke amanan sangat tergantung pada sistem operasi, teknologi trusted O/S memberikan mekanisme satu-satunya pada O/S untuk bertahan terhadap serangan.
- Anti D.D.O.D Tools. Tool anti Ddos akan mengidentifikasi ketidak beresan penggunaan di jaringan. Jika terjadi ketidak beresan, tool akan berusaha mencek legitimasi akses dan merekomendasikan beberapa langkah preventif-nya.

Trend Keamanan Internet

Malware akan terus berkembang

Malware telah menjadi cara yang paling efektif bagi penyerang untuk mencapai target secara global. Perambatan malware telah menjadi metode serangan yang sangat telak selama beberapa tahun terakhir, dan akibatnya efektivitas sebagian besar produk antivirus dipertanyakan. Semakin banyak vendor keamanan menawarkan pertahanan perangkat lunak dari aplikasi jahat, namun tidak semua solusi vendor ini benar-benar efektif karena perangkat lunak perusak terus bermetamorfosis.

Salah satu tren yang muncul adalah munculnya malware memory-resident. Infeksi ini tidak akan bertahan dalam reboot dan sangat sulit dikenali secara forensik, namun seiring semakin banyak orang membiarkan komputer mereka terus berjalan, ini mungkin teknik serangan yang berhasil.

Sebagai pertahanan malware pada perusahaan dan komputer laptop pribadi menjadi lebih produktif, penyerang akan kembali menggeser teknik mereka. Tidak sulit untuk memprediksi bahwa lebih banyak penyerang akan beralih ke malware mobile. Karena kebanyakan perusahaan mengizinkan perangkat seluler untuk bergabung dengan jaringan WiFi internal perusahaan, perangkat seluler tersebut bisa sama mematikannya dan memungkinkan penyerang mendapatkan akses ke perut lunak perusahaan korporat.

Selain itu, karena layanan 4G dan 5G menyediakan bandwidth internet yang substansial, perangkat mobile ini dapat dimanfaatkan untuk serangan DDoS yang sangat manjur. Baru-baru ini, Lookout and Ponemon Institute memperkirakan bahwa pelanggaran data mobile bisa menghabiskan biaya milyard rupiah, jadi ini sesuatu yang harus diantisipasi perusahaan.

Komplexitas IoT menyebabkan isu keamanan dan serangan DDoS selanjutnya

Cisco Visual Networks Index (VNI) telah memperkirakan bahwa pada tahun 2020, akan ada lebih dari 26 miliar perangkat yang terhubung dengan jaringan IP. Seiring Internet Things (IoT) mencapai jaringan perusahaan korporat, rumah konsumen dan pemerintah daerah, risiko keamanan meningkat karena target gabungan lebih besar.

Dunia IoT memiliki masalah memiliki beragam protokol dan standar, perusahaan yang kurang memiliki keterampilan dengan sistem IoT, arsitektur yang terlalu rumit, produk dengan fitur keamanan lemah, tindakan keamanan yang lemah dan ketidakmatangan operasional. Semua itu menyebabkan lebih banyak masalah keamanan. Kami telah melihat serangan DDoS yang sangat besar yang bersumber dari perangkat IoT yang rentan.

Menerapkan AI dan mechine learning untuk keamanan

Banyak praktisi keamanan suka mengutip kutipan terkenal Thomas Jefferson "kewaspadaan terus menerus adalah harga kebebasan" saat mereka menekankan pentingnya visibilitas dan pemantauan. Hal ini akan baik dan bagus bila jumlah data untuk penyelidikan bisa dikelola. Namun, saat ini, sebagian besar perusahaan tidak dapat mengikuti dan lebih suka beralih ke Managed Security Service Provider (MSSP) untuk mendapatkan bantuan.

Di dunia modern, perangkat intelijen ancaman dan hibrida-TI, aktivitas pemantauan dan pengelolaan keamanan telah melampaui kemampuan manusia. Sekarang ada banyak vendor

keamanan dan penyedia layanan yang mengiklankan fitur seperti kecerdasan buatan, pembelajaran mesin, dan pembelajaran mendalam, algoritma lanjutan dan visualisasi data untuk membantu perusahaan mengenali dan merespons serangan. Baru-baru ini, IBM Watson membuka kemampuannya yang luas ke dunia cybersecurity.

Pada tahun 2017, perusahaan dapat berharap untuk dibanjiri dengan klaim produk yang sangat bagus ini dan kata kunci keamanan yang lebih baik. Kita dapat berharap bahwa pemasaran ini akan memberi jalan bagi produk nyata yang menunjukkan kemampuan canggih ini dan diterjemahkan ke dalam tindakan perlindungan keamanan yang nyata.

SDN muncul dari palung kekecewaan, dan SD-WAN akan terbang

Selama bertahun-tahun, grafik Gartner Hype Cycle telah membantu organisasi memvisualisasikan siklus jatuh tempo teknologi dan membedakan teknologi yang belum matang untuk adopsi perusahaan yang luas. Terlepas dari antisipasi kemampuan mengesankan dari software-defined networking (SDN), Gartner dengan benar telah menempatkan teknologi ini di dalam fase kekecewaan.

Dibandingkan dengan penyedia layanan besar, penyedia layanan cloud berskala besar dan multi-penyewa, dan lingkungan High Performance Computing (HPC), kebanyakan perusahaan tampak seperti teknologi yang lamban. Industri SDN berharap perusahaan akan mulai menerapkan SDN dan akhirnya akan mulai memanfaatkan API tenang, otomasi, programabilitas jaringan, dan fitur canggih seperti multi-tenancy, campus-slicing, dan segmentasi mikro.

Pada tahun 2016, kita menyaksikan banyak perusahaan yang mendidik diri mereka sendiri mengenai teknologi WAN (SD-WAN) yang didefinisikan perangkat lunak dan mengevaluasi produk dari segudang vendor di tempat itu. Pada tahun 2017, banyak perusahaan mungkin memiliki perpanjangan kontrak MPLS WAN dan upgrade router cabang yang akan menciptakan kegiatan untuk beralih ke SD-WAN. Karena itu, kami berharap penggunaan hybrid-WAN akan terus berkembang dalam beberapa tahun ke depan.

Keamanan Cloud menjadi semakin penting

Dalam beberapa tahun terakhir, telah terjadi beberapa pelanggaran keamanan cloud yang dipublikasikan, dan banyak organisasi masih menggunakan keamanan sebagai penghalang jalan, mencegah organisasi mereka memanfaatkan manfaat dari komputasi cloud. Sekedar informasi, keamanan merupakan inisiatif utama bagi AWS dan ekosistem dan pelanggan mitranya.

Kini ada saran keamanan dan praktik terbaik yang tersedia untuk diikuti oleh organisasi sehingga mereka memulai perjalanan mereka ke awan dengan aman. Bagi organisasi yang beroperasi di awan, ada beberapa praktik dan pedoman terbaik untuk mengaudit penerapan mereka. Ada juga sertifikasi keamanan awan seperti Cloud Security Alliance (CSA) Certificate of Cloud Security Knowledge (CCSK) dan (ISC) 2 Certified Cloud Security Practitioner (CCSP). AWS juga telah membuat ujian khusus Certified Advanced Security.

Karena lebih banyak diketahui tentang bagaimana menerapkannya dengan aman di lingkungan cloud dan perusahaan menyadari bahwa mereka dapat beroperasi dengan aman di cloud seperti di pusat data tradisional lokal, adopsi cloud akan meningkat. Namun, jika sebuah organisasi memiliki kebersihan keamanan yang buruk di lingkungan lokal mereka dan mereka menggunakan praktik

keamanan yang longgar di cloud, maka kita dapat mengharapkan lebih banyak pelanggaran keamanan cloud.

IPv6 traffic akan terus bertambah

Kemajuan internet global pada penyebaran IPv6 telah lamban namun stabil dalam beberapa tahun terakhir. Sekilas ke halaman Statistik Google IPv6 atau situs 6lab Cisco akan mengungkapkan bahwa IPv6 mendapatkan uap. IPv6 telah digunakan oleh sebagian besar penyedia layanan internet, penyedia internet broadband, dan operator seluler. Orang sekarang cenderung memiliki IPv6 yang berjalan di perangkat mobile di saku dan di rumah mereka.

Namun, perusahaan telah lamban untuk mengadopsi IPv6 di manapun pada jaringan internal mereka, kecuali beberapa perusahaan yang telah menerapkannya di edge internet mereka. Apa yang kita lihat adalah bahwa IPv6 mempercepat saat IPv4 mendekati habisnya. Berdasarkan hal tersebut, kita bisa mengharapkan adopsi IPv6 terus meningkat tajam. Namun dalam beberapa tahun, tingkat adopsi akan mulai melambat karena mencapai 50 persen dari total lalu lintas internet.

Dewasa ini, ada banyak ketidakpastian di dunia pada umumnya, dan industri TI secara tradisional sangat sulit diprediksi. Bayangkan jika lima tahun yang lalu anda bisa meramalkan dunia yang kita jalani hari ini. Mengantisipasi apa yang mungkin terjadi 18 sampai 24 bulan adalah mungkin, namun mencoba memprediksi tiga sampai lima tahun ke depan hampir mustahil.

Kami berada di tahap awal peningkatan IoT, cloud, AI, otomasi dan kinerja. Kita dapat berharap tentang kemajuan produk dan layanan TI dan kemampuan mereka untuk menciptakan bisnis baru dan menawarkan manfaat bagi masyarakat luas. Harapan terbaik untuk musim liburan yang menyenangkan dan optimisme untuk masa mendatang.

BAB 2 Tip Mengamankan Server

Mengamankan server Linux anda sangat penting untuk memproteksi data, hak cipta, dan waktu, dari tangan2 jahil para cracker. Sistem administrator bertanggung jawab untuk keamanan linux. Pada bagian ini akan di jelaskan beberapa tip untuk mengamankan instalasi linux standard.

Enkripsi Komunikasi Data

Semua data yang di kirimkan melalui jaringan akan terbuka untuk di monitoring. Enkrip data yang dikirim sebisa mungkin dengan password atau menggunakan kunci / key / sertifikat.

- Gunakan scp, ssh, rsync, atau sftp untuk melakukan file transfer. Kita juga dapat me-mount file system di remote server atau home directory kita menggunakan sshfs dan fuse tool.
- GnuPG memungkinkan kita untuk mengenkrip dan sign komunikasi data yang kita lakukan. GnuPG juga mempunyai sistem manajemen key yang baik dan akses ke berbagai directory public key.
- Fugu adalah tampilan grafis dari aplikasi commandline Secure File Transfer (SFTP). SFTP mirip dengan FTP, tapi tidak sama dengan FTP karena semua sesi komunikasi di enkrip. Artinya lebih sukar untuk di tembus oleh pihak ketiga. Aplikasi yang lain adalah FileZilla - sebuah client cross-platform yang mendukung FTP, FTP over SSL/TLS (FTPS), dan SSH File Transfer Protocol (SFTP).
- OpenVPN adalah cost-effective, dan ringan SSL VPN.
- Lighttpd SSL (Secure Server Layer) konfigurasi dan instalasi https
- Apache SSL (Secure Server Layer) konfigurasi dan instalasi https (mod_ssl)

Jangan gunakan FTP, Telnet, dan Rlogin / Rsh

Dalam kondisi jaringan yang normal, maka username, password, dari perintah FTP / telnet / rsh dan proses transfer file akan dengan mudah di tangkap oleh mereka yang berada di jaringan yang sama mengunakan sniffer. Solusi untuk hal ini dapat menggunakan OpenSSH , SFTP, atau FTPS (FTP over SSL), yang menambahkan enkripsi SSL atau TLS ke FTP. di turunan RedHat kita dapat menulis perintah berikut untuk membuang perintah NIS, rsh dan berbagai layanan yang kadaluarsa lainnya:

```
# yum erase inetd xinetd ypserv tftp-server telnet-server rsh-serve
```

Minimalkan Software Aplikasi untuk Minimalisasi Kelemahan

Apakah kita membutuhkan berbagai layanan terinstalasi? Hindari instalasi software yang tidak dibutuhkan untuk menghindari kelemahan di software. Gunakan RPM package manager seperti yum atau apt-get dan / atau dpkg untuk melihat semua software yang terinstalasi di sistem. Delete paket yang tidak di inginkan.

```
# yum list installed
# yum list packageName
# yum remove packageName
```

atau

```
# dpkg --list
# dpkg --info packageName
# apt-get remove packageName
```

Satu layanan jaringan per sistem atau per VM Instance

Jalankan layanan jaringan yang berbada di server atau VM instance yang terpisah. Hal ini membatasi jumlah layanan yang dapat di jebol. Sekedar contoh, jika seorang penyerang berhasil mengexploit sebuah software seperti Apache flow, dia akan memperoleh akses ke seluruh server termasuk layanan sepertu MySQL, e-mail server dan masih banyak lagi.

Menjaga Kernel Linux dan Software Tetap Up to Date

Menerapkan patch keamanan merupakan bagian penting dari menjaga server Linux. Linux menyediakan semua alat yang diperlukan untuk menjaga sistem anda diperbarui, dan juga memungkinkan untuk upgrade antar versi dengan mudah. Semua pembaruan keamanan harus ditinjau ulang dan diterapkan sesegera mungkin. Sekali lagi, gunakan manajer paket RPM seperti yum dan atau apt-get dan atau dpkg untuk menerapkan semua update keamanan.

```
# yum update
```

atau

```
# apt-get update && apt-get upgrade
```

Anda dapat mengkonfigurasi Red Hat / CentOS / Fedora Linux untuk mengirim notifikasi update paket yum update email. Pilihan lain adalah dengan menggunakan semua update keamanan melalui cron job. Dalam Debian / Ubuntu Linux Anda dapat menggunakan apticron untuk mengirim notifikasi keamanan.

Menggunakan Linux Security Extension

Linux hadir dengan berbagai patch keamanan yang dapat digunakan untuk menjaga terhadap program yang salah konfigurasi atau yang diambil alih pihak lain. Jika memungkinkan menggunakan SELinux dan ekstensi keamanan Linux lainnya untuk melakukan pembatasan pada jaringan dan program lainnya. Sebagai contoh, SELinux menyediakan berbagai kebijakan keamanan untuk kernel Linux.

SELinux

Saya sangat menyarankan menggunakan SELinux yang menyediakan sebuah Mandatory Access Control (MAC) yang fleksibel. Berdasarkan standar Linux Discretionary Access Control (DAC), sebuah aplikasi atau proses yang berjalan sebagai user (UID atau SUID) memiliki izin pengguna untuk objek seperti file, socket dan proses lainnya. Menjalankan sebuah kernel MAC melindungi sistem dari aplikasi berbahaya atau cacat yang dapat merusak atau menghancurkan sistem. Lihat dokumentasi Redhat resmi yang menjelaskan konfigurasi SELinux.

User Account dan Kebijakan Password Kuat

Gunakan perintah useradd/usermod untuk membuat dan memelihara account pengguna. Pastikan Anda memiliki kebijakan password yang baik dan kuat. Sebagai contoh, password yang baik mencakup minimal 8 karakter dan campuran huruf, angka, karakter khusus, huruf besar, huruf kecil,

dan lain-lain. Yang paling penting adalah memilih password yang bisa Anda ingat. Gunakan tool seperti "ripper John" untuk mengetahui password yang lemah pengguna pada server Anda. Konfigurasikan Pam_cracklib.so untuk menegakkan kebijakan password.

Di Ubuntu

```
apt-get install libpam-cracklib
```

Edit

```
vi /etc/pam.d/common-password
```

Tambahkan

```
password required pam_cracklib.so retry=2 minlen=10 difok=6
```

Umur Password

Perintah change mengubah jumlah hari antara perubahan password dan tanggal perubahan password terakhir. Informasi ini digunakan oleh sistem untuk menentukan kapan seorang pengguna harus mengubah passwordnya. File /etc/login.defs mendefinisikan konfigurasi situs-khusus untuk deretan shadow password termasuk konfigurasi umur password. Untuk menonaktifkan fitur umur password, masukkan:

```
chage -M 99999 userName
```

Untuk mendapatkan informasi password yang telah kadaluarsa, masukkan:

```
chage -l userName
```

Akhirnya, Anda juga dapat mengedit /etc/shadow seperti berikut:

```
{userName}:{password}:{lastpasswdchanged}:{Minimum_days}:{Maximum_days}:
{Warn}:{Inactive}:{Expire}:
```

Dimana,

- Minimum_days: Jumlah minimum dari hari yang diperlukan antara perubahan password, yaitu jumlah hari yang tersisa sebelum pengguna diijinkan mengubah passwordnya.
- Maximum_days: Jumlah maksimum hari password berlaku (setelah itu pengguna akan dipaksa untuk mengubah passwordnya).
- Warn : Jumlah hari sebelum password akan berakhir, pengguna akan diperingatkan bahwa passwordnya harus diubah.
- Expire : Tanggal mutlak saat login tidak mungkin lagi dapat dilakukan.

Sebaiknya untuk tidak mengedit /etc/shadow secara langsung:

```
# chage -M 60 -m 7 -W 7 userName
```

Paksa Ubah Password

Untuk memaksa mengubah password saat login pertama kali

```
chage -d 0 <username>
```

Pembatasan Penggunaan Password Lama

Anda dapat menghalangi / membatasi pengguna dalam menggunakan atau mendaur ulang password lama menggunakan Linux. Parameter modul pam_unix dapat dikonfigurasi untuk mengingat password terdahulu yang tidak dapat digunakan kembali.

Di Ubuntu edit,

```
vi /etc/pam.d/common-password
```

Tambahkan

```
password sufficient pam_unix.so use_authtok md5 shadow remember=13
```

atau

```
password sufficient pam_unix2.so use_authtok md5 shadow remember=13
```

Kunci Account User setelah beberapa kali gagal Login

Di Linux kita dapat menggunakan perintah faillog untuk memperlihatkan catatan faillog atau untuk menset batas kegagalan login. faillog akan mem-format tampilan / content dari catatan log dari database / log file /var/log/faillog. Dia juga dapat digunakan untuk menghitung dan membatasi kegagalan login. Untuk melihat percobaan login yang gagal, tulis:

```
faillog
```

atau lihat

```
/var/log/auth.log
```

Mengaktifkan faillog di Ubuntu, edit

```
vi /etc/pam.d/common-auth
```

Masukan di paling atas

```
auth required pam_tally.so no_magic_root
account required pam_tally.so deny=3 no_magic_root lock_time=300
```

Untuk mengunci sebuah account setelah gagal login, jalankan:

```
faillog -r -u userName
```

Kita dapat menggunakan perintah passwd untuk mengunci dan membuka kunci sebuah account:

```
# kunci account
passwd -l userName

# membuka kunci account
passwd -u userName
```

Verifikasi tidak ada Account dengan password kosong?

Ketik perintah berikut

```
# awk -F: '($2 == "") {print}' /etc/shadow
```

Kunci semua account dengan password kosong:

```
# passwd -l accountName
```

Pastikan tidak ada Account Non-Root yang mempunyai UID 0

Hanya account root yang mempunyai UID 0 dengan ijin penuh untuk mengakses system. Tuliskan perintah berikut untuk menampilkan semua account dengan UID yang di set 0:

```
# awk -F: '($3 == "0") {print}' /etc/passwd
```

Anda harusnya hanya melihat satu kalimat ini

```
root:x:0:0:root:/root:/bin/bash
```

Jika anda melihat kalimat yang lain, buang atau pastikan account tersebut memang di ijinkan untuk menggunakan UID 0.

Disable Login sebagai root

Jangan pernah login sebagai user root. Anda sebaiknya menggunakan sudo untuk menjalankan perintah level root jika diperlukan. sudo ini dapat meningkatkan keamanan sistem tanpa berbagi password root dengan pengguna lainnya dan admin. Perintah sudo juga menyediakan audit sederhana dan fitur pelacakan.

Edit

```
vi /etc/ssh/sshd_config
```

Pastikan

```
PermitRootLogin prohibit-password
StrictModes yes
```

atau yang lebih ketat,

```
PermitRootLogin no
```

Keamanan Fisik Server

Anda harus melindungi akses terhadap server Linux secara fisik. Lakukan konfigurasi pada BIOS seperti menonaktifkan boot dari perangkat eksternal seperti DVD/CD/USB. Anda juga dapat menambahkan password pada grub boot loader untuk memperketat akses terhadap server Linux. Anda juga disarankau untuk menyimpan Data penting yang terkait dengan produksi harus terkunci di IDCs (Internet Data Center) dan semua orang harus melewati semacam pemeriksaan keamanan sebelum mengakses server Anda.

Disable Layanan Yang Tidak Perlu

Nonaktifkan semua layanan yang tidak perlu dan daemon (layanan yang berjalan di latar belakang). Anda juga harus menghapus semua layanan yang tidak perlu dari sistem start-up. Ketikkan perintah berikut untuk melihat daftar semua layanan yang dihidupkan secara otomatis pada saat boot di runlevel #3:

```
chkconfig --list | grep '3:on'
```

Untuk mematikan layanan dan disable saat boot, masukkan:

```
service serviceName stop
chkconfig serviceName off
```

Di Ubuntu, bisa menggunakan

```
apt-get install sysv-rc-conf
sysv-rc-conf --list | grep '3:on'

service serviceName stop
sysv-rc-conf serviceName off
```

Alternatif perintah yang menarik

```
sysv-rc-conf apache2 on
sysv-rc-conf --list apache2
```

Alternatif perintah lain

```
update-rc.d <service> defaults
update-rc.d <service> start 20 3 4 5
update-rc.d -f <service> remove
```

Mencari Port Network yang Aktif

Gunakan perintah berikut untuk melihat port yang terbuka dan program yang berasosiasi dengan port tersebut:

```
netstat -tulpn
```

atau

```
nmap -sT -O localhost
nmap -sT -O server.example.com
```

Gunakan iptables untuk menutup port tersebut atau matikan layanan jaringan yang tidak di inginkan dan gunakan perintah chkconfig.

Detect Port Scan

Install

```
sudo apt-get install psad
```

Edit

```
vi /etc/syslog.conf

kern.info       |/var/lib/psad/psadfifo
```

Restart

```
/etc/init.d/sysklogd restart
/etc/init.d/klogd
```

Edit

```
vi /etc/psad/psad.conf

EMAIL_ADDRESSES             vivek@nixcraft.in;
HOSTNAME                    server.nixcraft.in;
HOME_NET                    NOT_USED;  ### only one interface on box
IGNORE_PORTS                udp/53, udp/5000;
ENABLE_AUTO_IDS             Y;
IPTABLES_BLOCK_METHOD       Y;
```

Restart

```
/etc/init.d/psad restart
```

Modif iptables

```
iptables -A INPUT -j LOG
iptables -A FORWARD -j LOG
```

Report

```
psad -S
```

Detail iptables ada di https://www.cyberciti.biz/faq/linux-detect-port-scan-attacks/

Hapus X Windows

X windows pada server tidak diperlukan. Tidak ada alasan untuk menjalankan X Windows pada server khusus mail dan server web Apache. Anda dapat menonaktifkan dan menghapus X Windows untuk meningkatkan keamanan server dan kinerja. Edit /etc/inittab dan ubah runlevel ke 3. Akhirnya, untuk menghapus sistem X Windows, masukkan:

```
# yum groupremove "X Window System"
```

Pengaturan Iptables dan TCPWrappers

Iptables adalah firewall (Netfilter) standar yang disediakan oleh kernel Linux. Gunakan firewall untuk memfilter lalu lintas dan hanya mengijinkan lalu lintas yang diperlukan. Anda juga dapat menggunakan sistem jaringan TCPWrappers ACL berbasis host untuk menyaring akses jaringan ke Internet.

Linux Kernel /etc/sysctl.conf Hardening

/etc/sysctl.conf adalah file yang digunakan untuk mengkonfigurasi parameter kernel pada saat runtime. Linux membaca dan menerapkan pengaturan dari /etc/sysctl.conf pada saat boot. Contoh konfigurasi pada /etc/sysctl.conf:

```
# Turn on execshield
kernel.exec-shield=1
kernel.randomize_va_space=1
# Enable IP spoofing protection
net.ipv4.conf.all.rp_filter=1
# Disable IP source routing
net.ipv4.conf.all.accept_source_route=0
# Ignoring broadcasts request
net.ipv4.icmp_echo_ignore_broadcasts=1
net.ipv4.icmp_ignore_bogus_error_messages=1
# Make sure spoofed packets get logged
net.ipv4.conf.all.log_martians = 1
```

Pisahkan Partisi Disk

Pisahkan file sistem operasi dari file user agar system lebih baik dan lebih aman. Pastikan file system berikut di mount di partisi yang berbeda:

```
/usr
/home
/var and /var/tmp
/tmp
```

Buat partis yang beda untuk root dari Apache dan FTP server. Edit file /etc/fstab dan pasitikan tambahkan opsi konfigurasi berikut:

- noexec - Tidak bisa menjalankan (execute) semua binary di partisi tersebut (binary tidak bisa di exec tapi script di ijinkan).
- nodev - Tidak mengijinkan device character atau device spesial lainnya di partisi tersebut (device file seperti zero, sda dll tidak dapat digunakan).
- nosuid - Tidak dapat menset akses SUID/SGID di partisi ini (menghalangi setuid bit).

Contoh isi /etc/fstab untuk membatasi akses user ke /dev/sda5 (ftp server root directory):

```
/dev/sda5  /ftpdata          ext3    defaults,nosuid,nodev,noexec 1 2
```

Kuota disk

Pastikan Kuota Disk diaktifkan untuk semua pengguna. Untuk menerapkan kuota disk, gunakan langkah-langkah berikut:

- Aktifkan kuota per sistem file dengan memodifikasi file /etc/fstab.
- Lakukan remount file system (s).
- Membuat database file kuot dan menghasilkan tabel penggunaan disk.
- Tetapkan kebijakan kuota.
- Lihat tutorial penerapan kuota disk untuk rincian lebih lanjut.

Matikan IPv6

Internet Protocol version 6 (IPv6) memberikan lapisan Internet baru dari TCP / IP protocol suite yang menggantikan Internet Protocol version 4 (IPv4) dan memberikan banyak manfaat. Saat ini ada alat yang baik keluar yang dapat memeriksa sistem melalui jaringan untuk masalah keamanan IPv6. Kebanyakan distro Linux mulai memungkinkan protokol IPv6 secara default. Cracker dapat mengirimkan lalu lintas data jahat melalui IPv6 yang tidak termonitor oleh admin. Kecuali konfigurasi jaringan menuntut hal itu.

Matikan Binari dengan SUID dan SGID Yang Tidak Diinginkan

Semua bit SUID/SGID yang di enable dapat di salahgunakan saat SUID/SGID executable ada masalah keamanan atau bug. Semua local atau remote user akan dapat menggunakan file tersebut. Ada baiknya kita mencari semua file tersebut, untuk mencarinya dapat menggunakan perintah berikut:

```
#See all set user id files:
find / -perm +4000
# See all group id files
find / -perm +2000
# Or combine both in a single command
find / \( -perm -4000 -o -perm -2000 \) -print
find / -path -prune -o -type f -perm +6000 -ls
```

Anda perlu melakukan analisa / investigasi pada setiap file yang di laporkan. See reported file man page for further details.

File yang World-Writable

Semua orang dapat memodifikasi file yang world-writable yang menyebabkan masalah keamanan. Gunakan perintah berikut untuk menemukan semua file yang di set world writable dan sticky bits:

```
find /dir -xdev -type d \( -perm -0002 -a ! -perm -1000 \) -print
```

Kita perlu melakukan analisa dari semua file yang di laporkan dan set ijin user dan group yang benar atau bahkan membuang / men-delete-nya sekalian.

File Noowner

File yang tidak ada pemiliknya dapat menyebabkan masalah keamanan. Cari file tersebut menggunakan perintah berikut

```
find /dir -xdev \( -nouser -o -nogroup \) -print
```

Kita perlu melakukan analisa pada setiap file yang di laporkan dan kita perlu memberikan user & group yang benar atau membuang / men-delete file tersebut.

Gunakan Layanan Authentikasi Terpusat

Tanpa system authentikasi yang terpusat, data user auth menjadi tidak konsisten, yang mungkin akan menyebabkan banyak data, credential, account yang out-of-date tidak ter-delete. Sebuah layanan authentikasi yang terpusat memungkinkan kita untuk memelihara kontrol terhadap data Linux / UNIX account dan authentikasi. Kita dapat menyimpan data auth yang tersinkonisasi pada beberapa server. Jangan menggunakan layanan NIS untuk authentikasi yang terpusat. Gunakan OpenLDAP untuk client dan server.

Kerberos

Kerberos akan melakukan authentikasi sebagai sebuah layanan authentikasi pihak ketiga yang dapat di percaya menggunakan cryptographic shared secret dengan asumsi packet akan berjalan di sebuah jaringan yang tidak aman yang dapat dibaca, di modifikasi dan di insert / dimasukan. Kerberos dibangun menggunakan symmetric-key cryptography dan membutuhkan sebuah pusat distribusi kunci (key). Kita dapat membuat remote login, remote copy, copy file inter-system secara aman dan berbagai pekerjaan dengan resiko tinggi secara aman dan banyak lagi yang dapat di kontrol oleh Kerberos. Oleh karenanya, jika pengguna melakukan authentikasi untuk layanan jaringan menggunakan Kerberos, maka pengguna yang tidak di undang yang berusaha untuk memperoleh password dengan cara memonitor traffic di jaringan pada dasarnya akan tersingkir dengan sendirinya. See how to setup and use Kerberos.

Logging dan Auditing

Kita perlu mengkonfigurasi logging dan auditing untuk mencatat semua usaha hacking dan cracking. Secara default syslog akan menyimpan data di directory /var/log/. Catatan ini sangat berguna untuk melihat jika ada software yang salah konfigurasi yang menjadikan sistem kita terbuka terhadap serangan.

Monitor Message Log yang mencurigakan menggunakan Logwatch / Logcheck

Membaca log menggunakan logwatch atau logcheck. Install menggunakan perintah

```
apt-get install logcheck
apt-get install logwatch
```

Dengan tool ini akan membuat pembacaan log menjadi lebih mudah. Kita dapat memperoleh laporan yang lebih detail dari hal yang mencurigakan di syslog melalui e-mail. Menjalankan menggunakan perintah

```
sudo -u logcheck logcheck
```

Sebuah contoh dari laporan syslog adalah sebagai berikut:

```
 ################### Logwatch 7.3 (03/24/06) ####################
       Processing Initiated: Fri Oct 30 04:02:03 2009
       Date Range Processed: yesterday
                             ( 2009-Oct-29 )
                             Period is day.
     Detail Level of Output: 0
           Type of Output: unformatted
         Logfiles for Host: www-52.nixcraft.net.in
  ##################################################################

 ---------- Named Begin ----------

 **Unmatched Entries**
    general: info: zone XXXXXX.com/IN: Transfer started.: 3 Time(s)
    general: info: zone XXXXXX.com/IN: refresh: retry limit for master ttttttttttttttttt#53 exceeded
(source ::#0): 3 Time(s)
    general: info: zone XXXXXX.com/IN: Transfer started.: 4 Time(s)
    general: info: zone XXXXXX.com/IN: refresh: retry limit for master ttttttttttttttttt#53 exceeded
(source ::#0): 4 Time(s)

 ---------- Named End ----------

  ---------- iptables firewall Begin ----------

 Logged 87 packets on interface eth0
   From 58.y.xxx.ww - 1 packet to tcp(8080)
   From 59.www.zzz.yyy - 1 packet to tcp(22)
   From 60.32.nnn.yyy - 2 packets to tcp(45633)
   From 222.xxx.ttt.zz - 5 packets to tcp(8000,8080,8800)

 ---------- iptables firewall End ----------

 ---------- SSHD Begin ----------
```

```
Users logging in through sshd:
  root:
    123.xxx.ttt.zzz: 6 times

---------- SSHD End ----------

---------- Disk Space Begin ----------

Filesystem            Size  Used Avail Use% Mounted on
/dev/sda3             450G  185G  241G  44% /
/dev/sda1              99M   35M   60M  37% /boot

---------- Disk Space End ----------

########## Logwatch End ##########
```

(Note output is truncated)

System Accounting menggunakan auditd

auditd adalah layanan untuk melakukan system auditing. Dia bertanggung jawab untuk menulis catatan audit ke disk. Pada saat startup, aturan di /etc/audit.rules akan di baca oleh daemon ini. Kita dapat membuka file /etc/audit.rules dan membuat perubahan seperti setup audit log file dan berbagai opsi lainnya. Dengan auditd kita dapat menjawab pertanyaan berikut:

- System startup and shutdown events (reboot / halt).
- Date and time of the event.
- User respoisble for the event (such as trying to access /path/to/topsecret.dat file).
- Type of event (edit, access, delete, write, update file & commands).
- Success or failure of the event.
- Records events that Modify date and time.
- Find out who made changes to modify the system's network settings.
- Record events that modify user/group information.
- See who made changes to a file etc.

See our quick tutorial which explains enabling and using the auditd service.

Secure OpenSSH Server

SSH protocol di rekomendasikan untuk melakukan remote login dan remote transfer file. Akan tetapi, ssh sangat terbuka akan serangan.

Instalasi dan Penggunaan Intrusion Detection System

Sebuah network intrusion detection system (NIDS) adalah sebuah intrusion detection system yang akan berusaha mendeteksi berbagai aktifitas yang tidak baik seperti serangan denial of service, port scans atau bahkan usaha untuk mengcrack ke dalam komputer dengan cara memonitor traffic jaringan.

Sebaiknya dilakukan mengecekan integritas software sebelum system online dan masuk ke lingkungan produksi / operasional. Jika di mungkinkan ada baiknya di install software AIDE sebelum system tersambung ke jaringan apapun. AIDE adalah sebuah host-based intrusion detection system (HIDS) dia akan memonitor dan analisa internal dari system.

Snort adalah perangkat lunak untuk deteksi intrusi yang mampu melakukan packet logging dan analisis lalu lintas real-time pada jaringan IP.

Protect File, Directory danEmail

Linux menawarkan perlindungan yang sangat baik terhadap akses data yang tidak sah. Perizinan file dan MAC mencegah akses yang tidak sah dari mengakses data. Namun, perizinan yang ditetapkan oleh Linux tidak relevan jika penyerang memiliki akses fisik ke komputer dan hanya dapat memindahkan hard drive komputer ke sistem lain untuk menyalin dan menganalisis data sensitif. Anda dapat dengan mudah melindungi file, dan partitons di Linux menggunakan alat berikut:

- gpg – untuk encrypt & decrypt file.
- Openssl – untuk proteksi Linux / UNIX password.
- Ecryptfs – enkripsi directory.
- Dll.

Mengamankan Email Server

Anda dapat menggunakan SSL certificate dan gpg key untuk mengamankan komunikasi e-mail pada kedua komputer server dan klien:

- Securing Dovecot IMAPS / POP3S Server dengan SSL
- Linux Postfix SMTP (Mail Server) SSL Certificate.
- Courier IMAP SSL Server Certificate.
- Sendmail SSL encryption untuk sending & receiving email.
- Enigmail: Encrypted mail untuk Mozilla thunderbird.

BAB 3 Firewall iptables

iptables adalah firewall, yang default di install di hampir semua distribusi Linux, seperti, Ubuntu, Kubuntu, Xubutu, Fedora Core, dll. Pada saat kita menginstalasi Ubuntu, iptables memang sudah terinstall, tapi default-nya mengijinkan semua traffic untuk lewat.

Memang banyak sekali dan bisa menjadi sangat sangat kompleks teknik konfigurasi iptables. Pada kesempatan ini kita hanya mencoba melakukan konfigurasi firewall / iptables yang sederhana saja.

Perintah Dasar

Anda dapat menulis,

```
$ sudo iptables -L
```

Akan keluar aturan “rules” yang sudah ada di iptables. Jika kita baru saja menginstalasi server, biasanya masih belum ada rules yang terpasang, kita akan melihat

```
Chain INPUT (policy ACCEPT)
target     prot opt source               destination

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

Option Dasar iptables

Berikut adalah beberapa option dasar yang sering digunakan dalam mengkonfigurasi iptables.

-A – Tambahkan rule / aturan ini ke rantai aturan yang ada. Rantai yang valid adalah INPUT, FORWARD and OUTPUT. Kita biasanya lebih banyak menggunakan rantai INPUT yang berdampak pada traffic yang masuk.

-L – memperlihatkan daftar aturan / rule yang ada iptables.

-m state – mengijinkan aturan di cocokan berdasarkan kondisi sambungan (connection state). Mengijinkan penggunaan option -–state.

--state – Mendefinisikan daftar dari kondisi / states bagi aturan untuk di cocokan. Beberapa state yang valid, adalah,

NEW – Sambungan baru, dan belum pernah terlihat sebelumnya.
RELATED – Sambungan baru, tapi berhubungan dengan sambungan lain yang telah di ijinkan.
ESTABLISHED – Sambungan yang sudah terjadi.
INVALID – Traffic yang karena berbagai alasan tidak bisa di identifikasi.

-m limit - Dibutuhkan oleh rule jika ingin melakukan pencocokan dalam waktu / jumlah tertentu. Mengijinkan penggunakan option --limit. Berguna untuk membatasi aturan logging.

--limit – Kecepatan maksimum pencocokan, diberikan dalam bentuk angka yang di ikuti oleh "/second", "/minute", "/hour", atau "/day" tergantung seberapa sering kita ingin melakukan pencocokan aturan. Jika option ini tidak digunakan maka default-nya adalah "3/hour".

-p – Protokol yang digunakan untuk sambungan.

--dport – Port tujuan yang digunakan oleh aturan iptables. Bisa berupa satu port, bisa juga satu range ditulis sebagai start:end, yang akan mencocokan semua port start sampai end.

-j - Jump ke target yang spesifik. iptables mempunyai empat (4) target default, yaitu,

> ACCEPT - Accept / menerika paket dan berhenti memproses aturan dalam rantai aturan ini.
> REJECT - Reject / tlak paket dan beritahu ke pengirim bahwa kita menolak paket tersebut, dan stop pemrosesan aturan dalam rantai aturan ini.
> DROP – Diam-diam tidak pedulikan paket, dan stop pemrosesan aturan di rantai aturan ini.
> LOG - Log / catat paket, dan teruskan memprosesan aturan di rantai aturan ini. Mengijinkan penggunaan option --log-prefix dan --log-level.

--log-prefix – Jika pencatatan di lakukan, letakan text / tulisan sebelum catatan. Gunakan kutip di text / tulisan.

--log-level – Pencatatan menggunakan syslog level. 7 adalah pilihan yang baik, kecuali kita perlu suatu yang lain.

-i – Lakukan pencocokan jika paket yang masuk dari interface tertentu.

-I – Insert / masukan aturan. Butuh dua (2) option, yaitu, rantai aturan yang mana, dan nomor aturan. Jadi -I INPUT 5 akan memasukan ke rantai INPUT dan menjadikannya aturan nomor 5 di daftar.

-v – Menampilkan lebih banyak informasi di layar. Sangat membantu jika ada beberapa aturan yang tampak mirip jika di tampilkan tanpa -v.

Pengijinan Sesi Sambungan Yang Terbentuk

Kita dapat mengijinkaan sesi sambungan yang terbentuk untuk menerima traffic, melalui perintah,

```
$ sudo iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

Mengijinkan Traffic Masuk ke Port Tertentu.

Di awal proses, sebaiknya iptables memblok semua traffic. Biasanya kita membutuhkan untuk bekerja melalui saluran SSH, oleh karenanya biasanya kita mengijinkan untuk traffic SSH dan memblok traffic lainnya.

Untuk mengijinkan traffic masuk ke default port SSH nomor 22, kita harus mengijinkan semua TCP traffic yang masuk ke port 22.

```
$ sudo iptables -A INPUT -p tcp --dport ssh -j ACCEPT
```

Dari daftar option di atas, kita dapat mengetahui bahwa aturan iptables tersebut mengatur agar

masukkan aturan ini ke rantai input (-A INPUT) artinya kita melihat traffic yang masuk. cek apakah protokol yang digunakan adalah TCP (-p tcp). Jika TCP, cek apakah packet menuju port SSH (--dport ssh). Jika menuju SSH, maka packet di terima (-j ACCEPT).

Mari kita cek aturan yang di bentuk oleh perintah di atas menggunakan perintah iptables -L,

```
$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:ssh
```

Selanjutnya, kita akan mengijinkan semua traffic web untuk masuk, gunakan perintah berikut

```
$ sudo iptables -A INPUT -p tcp --dport 80 -j ACCEPT
```

Cek aturan yang kita buat mengunakan perintah iptables -L, sebagia berikut,

```
$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:www
```

Kita harus secara spesifik mengijinkan TCP traffic ke port SSH dan Web, tapi kita belum mem-block apa-apa, dan semua traffic masuk bisa masuk.

Blocking Traffic

Jika aturan telah memutusan untuk menerima packet (ACCEPT), maka aturan selanjutnya tidak akan berefek pada packet tersebut. Karena aturan yang kita buat mengijinkan SSH dan Web traffic, selama aturan untuk memblok semua traffic kita letakan terakhir sesudah aturan mengijinkan SSH dan Web, maka kita akan tetap dapat menerima traffic SSH dan Web yang kita inginkan. Jadi kita harus menambahkan (-A) aturan untuk mem-block traffic di akhir.

```
$ sudo iptables -A INPUT -j DROP
$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:www
```

```
DROP       all  --  anywhere             anywhere
```

Karena kita tidak menentukan interface atau protokol yang digunakan, semua traffic ke semua pirt maupun semua interface akan di blok, kecuali web dan SSH.

Editing iptables

Masalah utama yang akan kita peroleh adalah, loopback port pada interface “lo” akan di blok. Oleh karena itu kita perlu mengijinkan agar menerima semua traffic untuk loopback (“lo”). Hal ini dapat dilakukan dengan cara meng-Insert (-I) aturan pada rantai INPUT bagi interface lo, agar masuk ke urutan paling atas.

```
$ sudo iptables -I INPUT 1 -i lo -j ACCEPT
$ sudo iptables -L
Chain INPUT (policy ACCEPT)
target     prot opt source               destination
ACCEPT     all  --  anywhere             anywhere
ACCEPT     all  --  anywhere             anywhere            state RELATED,ESTABLISHED
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:ssh
ACCEPT     tcp  --  anywhere             anywhere            tcp dpt:www
DROP       all  --  anywhere             anywhere
```

Kalau kita lihat di atas, aturan paling atas dan aturan paling bawah agak mirip, untuk melihat lebih detail dari aturan tersebut, kita dapat menggunakan perintah,

```
$ sudo iptables -L -v

Chain INPUT (policy ALLOW 0 packets, 0 bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 ACCEPT     all  --  lo     any     anywhere             anywhere
    0     0 ACCEPT     all  --  any    any     anywhere             anywhere            state
RELATED,ESTABLISHED
    0     0 ACCEPT     tcp  --  any    any     anywhere             anywhere            tcp dpt:ssh
    0     0 ACCEPT     tcp  --  any    any     anywhere             anywhere            tcp dpt:www
    0     0 DROP       all  --  any    any     anywhere             anywhere
```

Kita melihat lebih banyak informasi disini. Aturan untuk mengijinkan loopback sangat penting artinya, karena banyak program akan menggunakan interface loopback untuk berkomunikasi satu sama lain. Jika loopback tidak di ijinkan maka kemungkinan kita akan merusak program tersebut.

Logging / Pencatatan

Dalam semua contoh di atas, semua traffic tidak di log. Jika kita ingin untuk mencatat paket yang di dop, cara yang paling cepat adalah,

```
$ sudo iptables -I INPUT 5 -m limit --limit 5/min -j LOG --log-prefix "iptables denied: "
--log-level 7
```

Silahkan lihat di bagian atas untuk melihat aap yang terjadi dalam proses logging.

Saving iptables

Jika kita booting mesin yang kita gunakan, maka apa yang kita kerjakan sejauh ini akan hilang. Tentunya dapat saja kita mengetik ulang semua perintah yang kita masukkan satu per satu setiap kali reboot, agar lebih enak hidup kita, maka kita dapat menggunakan perintah iptables-save dan iptables-restore untuk menyimpan dan merestore iptables.

Bagi anda yang menggunakan Ubuntu terutama Ubuntu Fiesty, tampaknya Ubuntu Network Manager (masih beta) agak conflict dengan iptables. Oleh karenanya mungkin ada baiknya kita bypass Ubuntu Network Manager.

Dengan tidak menggunakan Ubuntu Network Manager, kita dapat men-save konfigurasi iptables agar di start setiap kali booting menggunakan perintah

```
$ sudo sh -c "iptables-save > /etc/iptables.rules"
```

Kita perlu memodifikasi /etc/network/interfaces agar aturan iptables yang kita gunakan dapat berjalan secara automatis. Memang kita perlu mengetahui ke interface mana aturan yang kita buat akan digunakan. Biasanya kita menggunakan eth0. Untuk interface wireless, kita dapat mencek penggunaaannya mengunakan perintah,

```
$ iwconfig
```

Kita perlu mengedit file /etc/network/interfaces misalnya menggunakan perintah

```
$ sudo nano /etc/network/interfaces
```

Jika kita sudah menemukan nama interface yang digunakan, maka di akhir interface kita dapat menambahkan perintah,

```
pre-up iptables-restore < /etc/iptables.rules
```

Selanjutnya di bawahnya kita tambahkan perintah sesudah interface down, menggunakan perintah,

```
post-down iptables-restore < /etc/iptables.rules
```

Contoh real konfigurasi interfaces adalah sebagai berikut,

```
auto eth0
iface eth0 inet dhcp
  pre-up iptables-restore < /etc/iptables.rules
  post-down iptables-restore < /etc/iptables.rules
```

Konfigurasi Startup di NetworkManager

Ubuntu Network Manager mempunyai kemampuan untuk menjalankan script pada saat dia mengaktifkan atau men-nonaktifkan interface. Untuk men-save aturan iptables pada saat shutdown, dan me-restore iptables saat startup, kita akan membuat script seperti itu. Untuk memulai, kita dapat mengedit file,

```
$ gksudo gedit /etc/NetworkManager/dispatcher.d/01firewall
```

Kita dapat memasukan script di bawah ini melalui editor, save dan exit.

```
#!/bin/bash

if [ -x /usr/bin/logger ]; then
	LOGGER="/usr/bin/logger -s -p daemon.info -t FirewallHandler"
else
	LOGGER=echo
fi

case "$2" in
	pre-up)
		if [ ! -r /etc/iptables.rules ]; then
			${LOGGER} "No iptables rules exist to restore."
			return
		fi
		if [ ! -x /sbin/iptables-restore ]; then
			${LOGGER} "No program exists to restore iptables rules."
			return
		fi
		${LOGGER} "Restoring iptables rules"
		/sbin/iptables-restore -c < /etc/iptables.rules
		;;
	post-down)
		if [ ! -x /sbin/iptables-save ]; then
			${LOGGER} "No program exists to save iptables rules."
			return
		fi
		${LOGGER} "Saving iptables rules."
		/sbin/iptables-save -c > /etc/iptables.rules
		;;
	*)
		;;
esac
```

Akhirnya, kita perlu memastikan bahwa Ubuntu Network Manager dapat menjalankan script tersebut. Melalui konsol, kita dapat menjalankan perintah berikut,

```
$ sudo chmod +x /etc/NetworkManager/dispatcher.d/01firewall
```

Sedikit Tip

Jika kita sering mengedit secara manual iptables. Perubahan iptables yang sering biasanya terjadi pada masa development, pada saat operasional sebetulnya tidak banyak perubahan aturan di iptables. Jika perubaha cukup banyak, maka sebaiknya kita menambahkan beberapa kalimat berikut ke file /etc/network/interfaces:

```
pre-up iptables-restore < /etc/iptables.rules
post-down iptables-save > /etc/iptables.rules
```

Kalimat "post-down iptables-save > /etc/iptables.rules" akan menyimpan aturan agar dapat digunakan lagi sesudah booting.

Penggunaan iptables-save/restore untuk Test Aturan

Jika kita berexperimen dengan iptables, ada baiknya menggunakan perintah iptables-save dan iptables-restore untuk mengedit dan test aturan yang kita buat. Untuk mengedit aturan iptables yang kita buat dapat menggunakan perintah berikut (misalnya menggunakan gedit),

```
$ sudo iptables-save > /etc/iptables.rules
$ gksudo gedit /etc/iptables.rules
```

Kita akan memperoleh sebuah file yang mirip dengan yang kita lakukan,

```
# Generated by iptables-save v1.3.1 on Sun Apr 23 06:19:53 2006
*filter
:INPUT ACCEPT [368:102354]
:FORWARD ACCEPT [0:0]
:OUTPUT ACCEPT [92952:20764374]
-A INPUT -i lo -j ACCEPT
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -i eth0 -p tcp -m tcp --dport 22 -j ACCEPT
-A INPUT -i eth0 -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -m limit --limit 5/min -j LOG --log-prefix "iptables denied: " --log-level 7
-A INPUT -j DROP
COMMIT
# Completed on Sun Apr 23 06:19:53 2006
```

Tampak dari file tersebut bahwa perintah tersebut adalah perintah iptables, tanpa ada “iptables”-nya. Kita dapat mengedit file ini, dan men-save jika telah selesai. Untuk melakukan test dapat di jalankan menggunakan perintah,

```
$ sudo iptables-restore < /etc/iptables.rules
```

Sesudah test, kita dapat mensave apa yang sedang di kutak-katik menggunakan perintah iptables-save ke file /etc/network/interfaces melalui perintah

```
$ sudo iptables-save > /etc/iptables.rules
```

Lebih Detail Tentang Logging

Untuk melihat lebih detail dari syslog kita perlu menambahkan rantai tambahan. Berikut adalah contoh dari /etc/iptables.rules memperlihatkan bagaimana setup iptables me-log dari syslog:

```
# Generated by iptables-save v1.3.1 on Sun Apr 23 05:32:09 2006
*filter
:INPUT ACCEPT [273:55355]
:FORWARD ACCEPT [0:0]
:LOGNDROP - [0:0]
```

```
:OUTPUT ACCEPT [92376:20668252]
-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
-A INPUT -i eth0 -p tcp -m tcp --dport 22 -j ACCEPT
-A INPUT -i eth0 -p tcp -m tcp --dport 80 -j ACCEPT
-A INPUT -i lo -j ACCEPT
-A INPUT -j LOGNDROP
-A LOGNDROP -p tcp -m limit --limit 5/min -j LOG --log-prefix "Denied TCP: " --log-level
7
-A LOGNDROP -p udp -m limit --limit 5/min -j LOG --log-prefix "Denied UDP: " --log-
level 7
-A LOGNDROP -p icmp -m limit --limit 5/min -j LOG --log-prefix "Denied ICMP: " --log-
level 7
-A LOGNDROP -j DROP
COMMIT
# Completed on Sun Apr 23 05:32:09 2006
```

Perhatikan ada rantai baru CHAIN di sebut LOGNDROP di awal file. Tampak standard DROP yang biasanya ada di bawah rantai INPUT sekarang digantikan oleh LOGNDROP dan menambahkan deskripsi protokol agar mudah membaca log tersebut. Akhirnya kita akan membuang / mendrop traffic di akhir rantai LOGNDROP. Beberapa catatan berikut akan memberikan keterangan apa yang terjadi,

--limit mengatur berapa banyak pencatatan dari dari sebuah aturan ke syslog
--log-prefix "Denied..." menambahkan prefix untuk memudahkan membaca syslog
--log-level 7 mengatur tingkat banyaknya informasi di syslog

Mematikan firewall

Jika kita membutuhkan untuk men-disable / mematikan firewall sementara, hal ini dapat dilakukan dengan mudah menggunakan perintah flush (-F), sebagai berikut,

```
$ sudo iptables -F
```

BAB 4 Secure Shell & Secure Copy

Secure Shell (SSH) adalah sebuah protokol jaringan untuk komunikasi data yang aman, layanan remote shell atau perintah eksekusi dan layanan jaringan lainnya yang aman antara dua komputer jaringan yang terhubung melalui saluran aman walaupun melalui jaringan yang tidak aman: server dan klien (masing-masing menjalankan server SSH dan program klien SSH). Spesifikasi protokol membedakan dua versi utama yang disebut sebagai SSH-1 dan SSH-2.

Aplikasi yang paling terkenal dari protokol adalah untuk akses ke account shell pada sistem operasi Unix. Ia dirancang sebagai pengganti Telnet dan lain protokol shell aman seperti rsh Berkeley dan rexec protokol, yang mengirim informasi, terutama password, di plaintext, membuat mereka rentan terhadap intersepsi dan pengungkapan menggunakan analisis paket. Enkripsi yang digunakan oleh SSH dimaksudkan untuk memberikan kerahasiaan dan integritas data melalui jaringan tidak aman, seperti Internet.

Instalasi openssh server

Instalasi

```
apt-get install openssh-server
```

Agar lebih aman, ada baiknya mengubah port ssh yang digunakan. Mengubah konfigurasi dengan cara mengedit file

```
vi /etc/ssh/ssh_config
```

Ubah

```
Port 22
```

Menjadi port yang lain, misalnya

```
Port 12345
```

Pastikan # di depan kata Port kita delete

Pertama kali login ke sebuah mesin

Saat kita pertama kali login ke sebuah server menggunakan ssh, misalnya

```
ssh onno@192.168.0.80
```

maka komputer kita berusaha mengenali identitas server tersebut, jika keluar

```
The authenticity of host '192.168.0.80 (192.168.0.80)' can't be established.
ECDSA key fingerprint is 10:d9:c9:21:24:8e:91:3e:3c:80:65:43:d2:96:59:1a.
Are you sure you want to continue connecting (yes/no)?
```

Berarti identitas server belum ada di database komputer kita, tepatnya di file .ssh/known_hosts kita perlu menjawab

    yes

Akan keluar warning / informasi

    Warning: Permanently added '192.168.0.80' (ECDSA) to the list of known hosts.

yang menyatakan bahwa identitas server tersebut dimasukan ke komputer kita, tepatnya di file ~/.ssh/known_hosts

Menjalankan perintah secara remote

contoh

```
ssh -l remoteuser remoteserver.com 'mkdir .ssh'
ssh -l remoteuser remoteserver.com 'touch ~/.ssh/authorized_keys'
cat ~/.ssh/id_dsa.pub | ssh -l remoteuser remoteserver.com 'cat >> ~/.ssh/authorized_keys'
```

SCP

Secure Copy atau SCP berarti mentransfer file komputer antara lokal dari host remote atau antara dua host remote. Ini berbasis pada protokol Secure Shell (SSH).

Istilah SCP mengacu pada dua hal, protokol SCP atau program SCP.

SCP protocol

Protokol SCP adalah protokol jaringan, berdasarkan protokol BSD RCP, yang mendukung transfer file antar host pada jaringan. SCP menggunakan Secure Shell (SSH) untuk transfer data dan menggunakan mekanisme otentikasi yang sama, sehingga memastikan keaslian dan kerahasiaan data dalam perjalanan. Klien dapat mengirim (upload) file ke server, termasuk atribut dasar (perizinan, cap waktu). Klien juga bisa meminta file atau direktori dari server (download). SCP berjalan di atas port TCP 22 secara default. Seperti RCP, tidak ada RFC yang mendefinisikan spesifik protokol.

Cara Kerja

Biasanya, klien memulai koneksi SSH ke host jarak jauh, dan meminta proses SCP dimulai di server jauh. Proses SCP remote dapat beroperasi dalam salah satu dari dua mode: mode sumber, yang membaca file (biasanya dari disk) dan mengirimnya kembali ke klien, atau mode wastafel, yang menerima file yang dikirim oleh klien dan menuliskannya (biasanya ke disk) pada remote host. Untuk kebanyakan klien SCP, mode sumber umumnya dipicu oleh flag -f (dari), sementara mode wastafel dipicu dengan -t (to). Flag ini digunakan secara internal dan tidak didokumentasikan di luar source code scp.

Remote to remote mode

Pada remote-to-remote secure copy, klien SCP membuka koneksi SSH ke host sumber dan meminta agar hal itu, pada gilirannya, membuka koneksi SCP ke tempat tujuan. (Mode remote-to-remote tidak beroperasi dengan membuka dua sambungan SCP dan menggunakan klien asal sebagai perantara). Penting untuk dicatat bahwa SCP tidak dapat digunakan untuk mengcopy dari sumber dari jarak jauh ke tujuan saat beroperasi dalam mode otentikasi kata sandi atau keyboard-interactive, karena ini akan mengungkapkan kredensial otentikasi server tujuan ke sumbernya. Namun, mungkin dengan metode berbasis kunci atau GSSAPI yang tidak memerlukan masukan dari pengguna.

SCP program

Program SCP adalah perangkat lunak yang mengimplementasikan protokol SCP sebagai daemon layanan atau klien. Ini adalah program untuk melakukan peng-copy-an yang aman. Program server SCP biasanya merupakan program yang sama dengan klien SCP.

Mungkin program SCP yang paling banyak digunakan adalah program scp command line, yang disediakan di sebagian besar implementasi SSH. Program scp adalah analog aman dari perintah rcp. Program scp harus menjadi bagian dari semua server SSH yang ingin menyediakan layanan SCP, karena fungsi scp sebagai server SCP juga.

Beberapa implementasi SSH menyediakan program scp2, yang menggunakan protokol SFTP dan bukan SCP, namun menyediakan antarmuka baris perintah yang sama seperti scp. scp kemudian biasanya merupakan symbolic link ke scp2.

Biasanya, sintaks program scp seperti sintaks dari cp:

Copying file ke host:

```
scp SourceFile user@host:directory/TargetFile
```

Copying file dari host:

```
scp user@host:directory/SourceFile TargetFile
scp -r user@host:directory/SourceFile TargetFolder
```

Perhatikan bahwa jika remote host menggunakan port selain default 22, anda bisa menentukannya dalam perintah. Misalnya, mengcopy file dari host:

```
scp -P 2222 user@host:directory/SourceFile TargetFile
```

Karena protokol SCP hanya melakukan transfer file, klien GUI SCP jarang ada, karena menerapkannya memerlukan fungsionalitas tambahan (setidaknya daftar direktori). Sebagai contoh, WinSCP default ke protokol SFTP. Bahkan saat beroperasi dalam mode SCP, klien seperti WinSCP biasanya bukan klien SCP murni, karena mereka harus menggunakan cara lain untuk menerapkan fungsi tambahan (seperti perintah ls). Hal ini pada gilirannya membawa masalah ketergantungan platform. Jadi, mungkin saja tidak mungkin bekerja dengan server SCP tertentu menggunakan klien GUI SCP, bahkan jika anda dapat bekerja dengan server yang sama menggunakan klien baris perintah biasa.

Tool yang lebih komprehensif untuk mengelola file melalui SSH adalah klien SFTP.

BAB 5 Virtual Private Network (VPN)

Virtual Private Network atau biasa disingkat dan dikenal umum sebagai VPN atau VPN tunnel per-definisi adalah sebuah mekanisme menyambungkan sebuah titik (atau biasa dengan node) pada sebuah jaringan komputer dengan titik yang lain melalui mediasi sebuah jaringan yang lain, dalam hal ini sebuah titik dapat berupa sebuah jaringan komputer lokal (atau biasa disebut LAN) atau sebuah komputer.

Sedangkan istilah tunnel sendiri (terlepas dari kata VPN) merupakan istilah generik yang menjelaskan bahwa sebuah hubungan antar titik pada sebuah jaringan komputer dilakukan melalui 'semacam terowongan' antar kedua titik. Macam tunnel bila dilihat dari Lapisan OSI dapat berupa tunnel layer 2 seperti tunnel PPP, tunnel PPPoE, VLAN dan sebagainya, tetapi tidak lazim disebut sebagai PPP VPN atau VLAN VPN.

VPN secara pengadaannya terbagi 2, yaitu :

- Voluntary tunnel, yaitu tunnel VPN yang dibuat secara sukarela oleh pengguna yang membutuhkan sambungan VPN antar titik pada jaringan komputernya.
- Compulsory tunnel, yaitu tunnel VPN yang secara khusus (baca : transparan) oleh ISP bagi pelanggan layanan VPN-nya.

VPN secara bentuk sambungannya terbagi 3, yaitu :

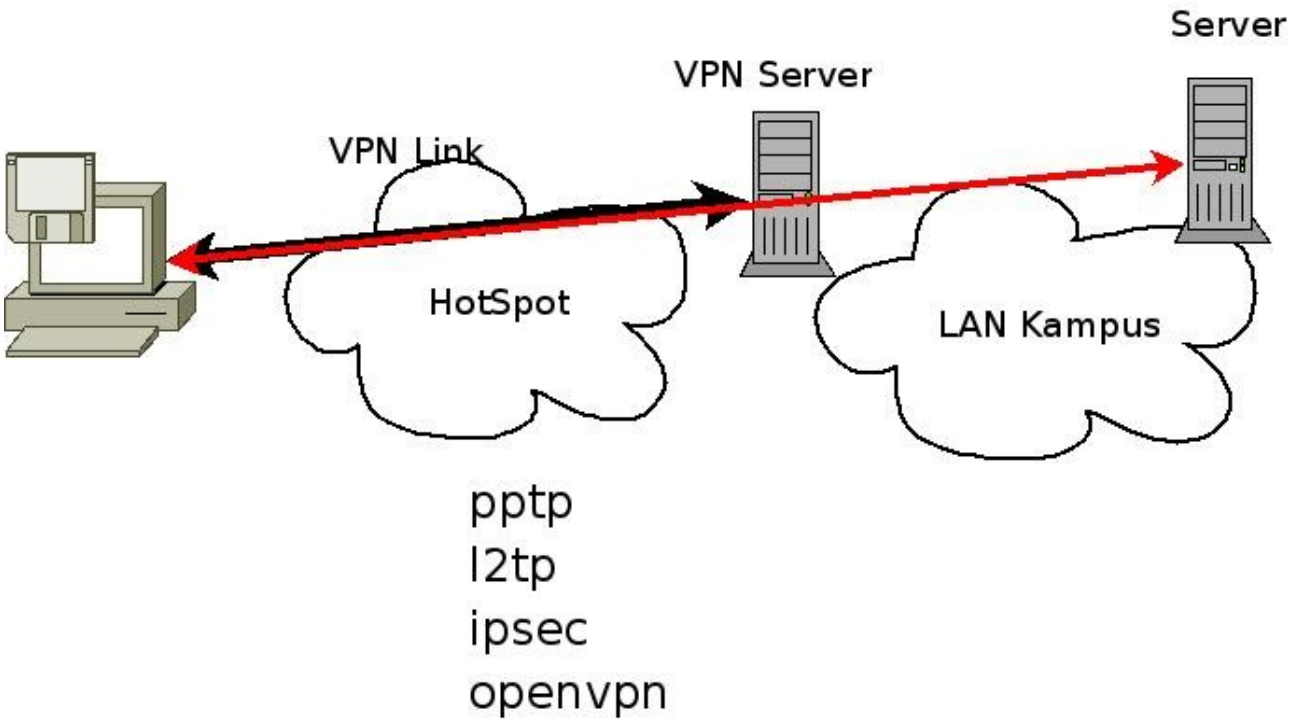
- Host-to-Host VPN, yaitu hubungan VPN secara langsung antar komputer.
- Site-to-Site VPN, yaitu hubungan VPN dilakukan antar router dari beberapa LAN.
- Host-to-Site VPN, yaitu hubungan VPN yang dilakukan oleh sebuah komputer kedalam sebuah jaringan LAN.

VPN secara pengamanannya terbagi 2, yaitu :

- Security VPN, yaitu metode sambungan VPN yang menerapkan beberapa hal terkait pengamanan komunikasi data - seperti enkripsi dan sebagainya. Contoh Security VPN : Point-to-Point Tunneling Protocol (atau PPTP), IP Security (atau IPSec), Layer 2 Tunneling Protocol (atau L2TP), Secure Socket Layer (atau SSL) dan sebagainya.
- IP VPN, yaitu metode sambungan VPN yang dilakukan oleh ISP melalui media IP secara keseluruhan didalam jaringan internalnya. Contoh IP VPN adalah mekanisme Multi Protocol Label Switching (atau MPLS) dan Virtual Private LAN Service (atau VPLS) dan seterusnya.

Media VPN sendiri dapat dilakukan melalui :

- Secara lokal LAN, yaitu berupa sambungan antara 2 titik atau lebih didalam sebuah jaringan lokalnya sendiri.
- Media jaringan pribadi WAN, yang biasanya VPN dilakukan langsung oleh pihak ISP
- Media internet, yang biasanya VPN dilakukan secara sukarela oleh pengguna.



Gambaran umum Virtual Private Network (VPN) terlihat pada gambar. Secara umum skenario yang ada adalah sebagai berikut,

- User menggunakan komputer / laptop mengakses melalui HotSpot / jaringan LAN / Internet.
- User login ke VPN Server.
- Laptop user akan terbentuk sambungan tambahan ke VPN Server. Sambungan ini merupakan "tunnel" yang semua paket yang lewat akan di enkripsi.
- Melalui "tunnel" yang di bentuk, laptop akan dapat mengakses Server yang ada di jaringan LAN yang ada di belakang VPN Server.

Instalasi PPTP

cek apakah kernel yang anda gunakan mendukung untuk melakukan MPPE.

```
# modprobe ppp-compress-18 && echo success
```

Instalasi PPTP

```
# apt-get install pptpd
```

Edit /etc/pptpd.conf

```
# vi /etc/pptpd.conf
```

Pastikan ada alokasi IP address

```
localip 192.168.0.1
remoteip 192.168.0.234-238,192.168.0.245
```

Restart PPTP

```
# /etc/init.d/pptpd restart
```

Menambahkan User PPTP ke PPP Password

```
# echo "username pptpd password *" >> /etc/ppp/chap-secrets
```

Instalasi OpenVPN

Install openvpn di Ubuntu

```
apt-get install openvpn
cp -Rf /usr/share/doc/openvpn/examples/easy-rsa/* /etc/openvpn/
```

Pada Ubuntu 8.10 akan di terlihat folder

```
/etc/openvpn/1.0
/etc/openvpn/2.0
```

Mungkin ada baiknya untuk pengguna Ubuntu 8.10, 9.04, 9.10 untuk memilih kita akan menggunakan konfigurasi 1.0 atau 2.0 dengan cara mengcopy

```
cp -Rf /etc/openvpn/2.0/* /etc/openvpn
```

Alternatif lain yang lebih susah, compile openvpn dari source code

```
cp openvpn-2.0.9.tar.gz /usr/local/src
cd /usr/local/src
tar zxvf openvpn-2.0.9.tar.gz
cd openvpn-2.0.9
./configure
make
make install
```

Anda tidak perlu mengcompile dari source code, jika sudah menginstalasi openvpn menggunakan apt-get install

Edit file vars di /etc/openvpn

```
# cd /etc/openvpn/
# vi vars

#this is to ensure secure data
export KEY_SIZE=1024
# These are the default values for fields
# which will be placed in the certificate.
# Don't leave any of these fields blank.
export KEY_COUNTRY=ID
export KEY_PROVINCE=DKI
export KEY_CITY=Jakarta
export KEY_ORG="Kerm.IT"
export KEY_EMAIL="onno@indo.net.id"
```

Membuat Certificate Authority (CA)

```
cd /etc/openvpn/
. ./vars
./clean-all
./build-ca

Country Name (2 letter code) [ID]:
State or Province Name (full name) [DKI]:
Locality Name (eg, city) [Jakarta]:
Organization Name (eg, company) [Kerm.IT]:
Organizational Unit Name (eg, section) []:Kerm.IT
Common Name (eg, your name or your server's hostname) []:yc0mlc.ampr.org
Email Address [onno@indo.net.id]:
```

Lihat keys apakah sudah di generate

```
ls -l /etc/openvpn/
ls -l /etc/openvpn/keys
```

Akan tampak file berikut

```
ca.crt
ca.key
index.txt
serial
```

Membuat Server Key

```
# ./build-key-server server

    Country Name (2 letter code) [ID]:
    State or Province Name (full name) [DKI]:
    Locality Name (eg, city) [Jakarta]:
    Organization Name (eg, company) [Kerm.IT]:
    Organizational Unit Name (eg, section) []:Kerm.IT
    Common Name (eg, your name or your server's hostname) []:yc0mlc.ampr.org
    Email Address [onno@indo.net.id]:

    Please enter the following 'extra' attributes
    to be sent with your certificate request
    A challenge password []:123456
    An optional company name []:Kerm.IT
    Using configuration from /etc/openvpn/openssl.cnf
    Check that the request matches the signature
    Signature ok
    The Subject's Distinguished Name is as follows
    countryName           :PRINTABLE:'ID'
    stateOrProvinceName   :PRINTABLE:'DKI'
    localityName          :PRINTABLE:'Jakarta'
    organizationName      :PRINTABLE:'Kerm.IT'
    organizationalUnitName:PRINTABLE:'Kerm.IT'
    commonName            :PRINTABLE:'yc0mlc.ampr.org'
    emailAddress          :IA5STRING:'onno@indo.net.id'
    Certificate is to be certified until Jan 13 03:34:36 2018 GMT (3650 days)
    Sign the certificate? [y/n]:y

    1 out of 1 certificate requests certified, commit? [y/n]y
    Write out database with 1 new entries
    Data Base Updated
```

Membuat Key User

Membuat key untuk user admin maupun user lainnya jika di perlukan

```
# ./build-key admin

    1 out of 1 certificate requests certified, commit? [y/n]y
    Write out database with 1 new entries
    Data Base Updated
```

Buat key untuk user lain jika di perlukan

```
./build-key-pass username
./build-key username
```

Membuat DH Parameter dari key

```
./build-dh

# openvpn --genkey --secret keys/ta.key

# openvpn --genkey --secret keys/ca.key
# openvpn --genkey --secret keys/ta.key
```

Test key

Test key

```
# openvpn --genkey --secret key
# openvpn --test-crypto --secret key
```

Test sambungan di 2 windows

Test yang sangat berguna melihat sambungan OpenVPN dari dua (2) Windows.

```
cd /etc/openvpn
cp -Rf /usr/share/doc/openvpn/examples/sample-config-files/ /etc/openvpn/
cp -Rf /usr/share/doc/openvpn/examples/sample-keys/ /etc/openvpn/
openvpn --config sample-config-files/loopback-client
openvpn --config sample-config-files/loopback-server
```

Jika di perlukan kita dapat menginstalasi OpenVPN Administrator. Contoh menginstalasi OpenVPN-Admin

```
# apt-get install mono openvpn-admin
```

Edit Server.conf

```
# vi /etc/openvpn/server.conf
```

isinya kurang lebih

```
# OpenVPN Server config file
# Which local IP address should OpenVPN listen on? (optional)
local 192.168.0.3

# Which TCP/UDP port should OpenVPN listen on?
port 1194

# TCP or UDP server?
proto udp

# "dev tun" will create a routed IP tunnel, which is what we want
dev tun

# SSL/TLS root certificate (ca), certificate
```

```
# (cert), and private key (key). Each client
# and the server must have their own cert and
# key file. The server and all clients will
# use the same ca file.
ca keys/ca.crt
cert keys/server.crt
key keys/server.key # This file should be kept secret
# Diffie hellman parameters.
dh keys/dh1024.pem

# Configure server mode and supply a VPN subnet
server 192.168.111.0 255.255.255.0

# Maintain a record of client <-> virtual IP address
# associations in this file.
ifconfig-pool-persist ipp.txt

# Push routes to the client to allow it
# to reach other private subnets behind
# the server. Remember that these
# private subnets will also need
# to know to route the OpenVPN client
# address pool (10.8.0.0/255.255.255.0)
# back to the OpenVPN server.
# push âroute 172.10.1.0 255.255.255.0"
# push âroute 192.168.0.0 255.255.255.0"
# If enabled, this directive will configure
# all clients to redirect their default
# network gateway through the VPN, causing
# all IP traffic such as web browsing and
# and DNS lookups to go through the VPN
; push "redirect-gateway"
# Certain Windows-specific network settings
# can be pushed to clients, such as DNS
# or WINS server addresses.
;push "dhcp-option DNS 172.10.1.2"
# Uncomment this directive to allow different
# clients to be able to âseeâ
client-to-client

# Ping every 10 seconds, assume that remote
# peer is down if no ping received during
# a 120 second time period.
keepalive 10 120

# For extra security beyond that provided
# by SSL/TLS, create an âHMAC firewallâ
# to help block DoS attacks and UDP port flooding.
; tls-auth keys/ta.key 0 # This file is secret
# Select a cryptographic cipher.
```

```
# This config item must be copied to
# the client config file as well.
;cipher BF-CBC # Blowfish (default)
;cipher AES-128-CBC # AES
;cipher DES-EDE3-CBC # Triple-DES

# Enable compression on the VPN link.
; comp-lzo

# The maximum number of concurrently connected
# clients we want to allow.
max-clients 250

# It's a good idea to reduce the OpenVPN
# daemonâs privileges after initialization.
user nobody
group nogroup

# The persist options will try to avoid
# accessing certain resources on restart
# that may no longer be accessible because
# of the privilege downgrade.
persist-key
persist-tun

# Output a short status file showing
status openvpn-status.log
log-append openvpn.log

# Set the appropriate level of log
# file verbosity.
#
# 0 is silent, except for fatal errors
# 4 is reasonable for general usage
# 5 and 6 can help to debug connection problems
# 9 is extremely verbose
verb 4

# Silence repeating messages. At most 20
# sequential messages of the same message
# category will be output to the log.
mute 20
```

Cara menjalankan VPN Server

Mengaktifkan VPN Server dengan server.conf (from www.openvpn.org)

```
# openvpn --config /etc/openvpn/server.conf
```

BAB 6 Web Application Firewall

Application firewall adalah firewall yang mengontrol input, output, dan / atau akses dari, untuk, atau dengan sebuah aplikasi atau layanan. Beroperasi dengan memantau dan pada dasarnya menghalangi panggilan input, output, atau layanan sistem yang tidak memenuhi kebijakan yang dikonfigurasikan pada firewall. Application firewall biasanya dibangun untuk mengontrol semua lalu lintas jaringan pada setiap lapisan OSI sampai ke lapisan aplikasi. Hal ini dapat mengontrol aplikasi atau layanan khusus, seperti firewall jaringan stateful, yang - tanpa software tambahan - tidak dapat mengontrol lalu lintas jaringan mengenai aplikasi tertentu. Ada dua kategori utama dari application firewall, application firewall berbasis jaringan dan application firewall berbasis host. Web Application Firewall adalah Firewall yang mengontrol input, output, dan / atau akses dari / ke sebuah web server.

ModSecurity

ModSecurity adalah firewall aplikasi web (WAF). Dengan lebih dari 70% dari serangan sekarang dilakukan pada tingkat aplikasi web, organisasi membutuhkan semua bantuan yang mereka bisa mendapatkan dalam membuat sistem mereka aman. WAF dikerahkan untuk meningkatkan lapisan keamanan eksternal untuk mendeteksi dan / atau mencegah serangan sebelum mereka mencapai aplikasi web. ModSecurity memberikan perlindungan dari berbagai serangan terhadap aplikasi web dan memungkinkan untuk pemantauan lalu lintas HTTP dan analisis real-time dengan sedikit atau tanpa perubahan infrastruktur yang ada.

ModSecurity merupakan open-source Web Application Firewall (WAF) untuk Apache Nginx dan web server IIS. firewall lapisan aplikasi ini dikembangkan oleh SpiderLabs Trustwave dan dirilis di bawah Apache License 2.0. ModSecurity melindungi website dari hacker dengan menggunakan seperangkat aturan ekspresi reguler untuk menyaring eksploitasi dikenal, memungkinkan pemantauan HTTP lalu lintas, penebangan, analisis real-time, dan deteksi serangan. Ada lebih dari 16.000 peraturan yang tersedia untuk mendeteksi serangan seperti SQL Injection, Cross-site Scripting (XSS), File Inclusion lokal, Remote File Inclusion dan aturan khusus aplikasi untuk banyak aplikasi web seperti Wordpress, Joomla, Drupal dll

Instalasi ModSecurity

Instalasi Ubuntu LTS server, or yang terbaru di mesin anda.
Instalasi Apache2 webserver, di setup dan di konfigurasi:

```
sudo apt-get install apache2 php5 php5-xmlrpc php5-mysql php5-gd php5-cli \
php5-curl mysql-client mysql-server
```

Instalasi dependensi yang dibutuhkan untuk modsecurity

```
sudo apt-get install libxml2 libxml2-dev libxml2-utils \
libaprutil1 libaprutil1-dev
```

Untuk mengguna 64bit, perlu menambahkan link berikut

```
ln -s /usr/lib/x86_64-linux-gnu/libxml2.so /usr/lib/libxml2.so
```

```
ln -s /usr/lib/x86_64-linux-gnu/libxml2.so.2 /usr/lib/libxml2.so.2
ln -s /usr/lib/x86_64-linux-gnu/libxml2.so.2.9.1 /usr/lib/libxml2.so.2.9.1
```

Instalasi ModSecurity

```
apt-get install libapache2-modsecurity modsecurity-crs
```

Periksa apakah modul mod_security dimuat.

```
apachectl -M | grep --color security
```

Akan keluar

```
security2_module (shared)
```

Anda akan melihat modul bernama security2_module (shared) yang menunjukkan bahwa modul dimuat.

Instalasi ModSecurity meliputi file konfigurasi yang disarankan yang harus diganti:

```
mv /etc/modsecurity/modsecurity.conf{-recommended,}
```

Reload Apache

```
service apache2 reload
```

Cek directory log Apache, akan ada file

```
ls -l /var/log/apache2/modsec_audit.log
```

```
-rw-r----- 1 root root 0 Mar 30 14:07 /var/log/apache2/modsec_audit.log
```

Konfigurasi ModSecurity

Apa adanya, ModSecurity tidak melakukan apa-apa karena kebutuhan aturan untuk bekerja. File konfigurasi default diset ke DetectionOnly yang mencatat permintaan sesuai dengan aturan yang cocok dan tidak memblokir apa-apa. Hal ini dapat diubah dengan mengedit file modsecurity.conf:

```
nano /etc/modsecurity/modsecurity.conf
```

Temukan kalimat ini

```
SecRuleEngine DetectionOnly
```

Ubah menjadi

```
SecRuleEngine On
```

Jika Anda mencoba ini pada server produksi, ubah direktif ini hanya setelah menguji semua aturan Anda.

Direktif lain yang perlu dimodifikasi adalah SecResponseBodyAccess. Ini mengkonfigurasi apakah tubuh respon dibuffered (yaitu dibaca oleh ModSecurity). Ini hanya diperlukan jika deteksi kebocoran data dan perlindungan diperlukan. Oleh karena itu, membiarkannya On akan menggunakan sumber daya droplet dan juga meningkatkan ukuran logfile.

Temukan

```
SecResponseBodyAccess On
```

Ubah menjadi

```
SecResponseBodyAccess Off
```

Sekarang kita akan membatasi data maksimum yang dapat diposting ke aplikasi web Anda. Dua parameter yang mengkonfigurasi ini:

```
SecRequestBodyLimit
SecRequestBodyNoFilesLimit
```

Parameter SecRequestBodyLimit menentukan ukuran data POST maksimal. Jika ada yang lebih besar dikirim oleh client maka server akan merespon dengan error 413 Request Entity Too Large. Jika aplikasi web Anda tidak memiliki file upload nilai ini dapat sangat dikurangi.

Nilai yang disebutkan dalam file konfigurasi

```
SecRequestBodyLimit 13107200
```

yaitu 12.5MB.

Sama dengan cara di atas adalah parameter SecRequestBodyNoFilesLimit . Perbedaan terutama pada besarnya data yang di POST dikurangi file uploads -- nilai ini sebaiknya "sekecil-kecilnya yang memungkinkan".

Nilai yang disebutkan dalam file konfigurasi

```
SecRequestBodyNoFilesLimit 131072
```

yaitu 128KB.

Diantara parameter-parameter diatas, parameter yang mungkin akan sangat mempengaruhi performance adalah SecRequestBodyInMemoryLimit. Parameter ini menentukan berapa banyak "request body" data (POST data) yang akan di simpan di RAM, selebihnya akan di simpan di harddisk (seperti swap). Jika kita menggunakan SSD, maka hal ini bukan sebuah masalah besar. Kita dapat menset lebih besar jika kita mempunyai RAM lebih,

```
SecRequestBodyInMemoryLimit 131072
```

Nilai ini 128KB seperti yang di set di file konfigurasi.

Testing SQL Injection

Sebelum melanjutkan dengan konfigurasi aturan, kita akan membuat script PHP yang rentan terhadap injeksi SQL dan mencobanya. Harap dicatat bahwa ini hanyalah sebuah script PHP untuk login tanpa penanganan sesi. Pastikan untuk mengganti password MySQL di script di bawah ini sehingga akan terhubung ke database:

/var/www/html/login.php

Jika password root MySQL adalah 123456, maka isinya

```
<html>
<body>
<?php
   if(isset($_POST['login']))
   {
      $username = $_POST['username'];
      $password = $_POST['password'];
      $con = mysqli_connect('localhost','root','123456','sample');
      $result = mysqli_query($con, "SELECT * FROM `users` WHERE
username='$username' AND password='$password'");
      if(mysqli_num_rows($result) == 0)
         echo 'Invalid username or password';
      else
         echo 'Logged in - A Secret for you....';
   }
   else
   {
?>
      <form action="" method="post">
         Username: <input type="text" name="username"/>

         Password: <input type="password" name="password"/>

         <input type="submit" name="login" value="Login"/>
      </form>
<?php
   }
?>
</body>
</html>
```

Script ini akan menampilkan form login. Dengan memasukan password yang tepat akan menampilkan pesan "A Secret for you."

Kita perlu password dalam database. Buat database MySQL dan table, kemudian masukkan username dan password.

```
mysql -u root -p123456

create database sample;
connect sample;
create table users(username VARCHAR(100),password VARCHAR(100));
insert into users values('jesin','pwd');
insert into users values('alice','secret');
quit;
```

Buka browser, masuk ke

```
http://ip-address-web-anda/login.php
```

misalnya

```
http://192.168.0.100/login.php
```

dan masukan pasangan kredensial

```
Username: jesin
Password: pwd
```

Kita akan melihat message yang mengindikasikan login sukses. Coba lagi, tapi masukan pasangan kredential yang salah -- kita akan melihat message

```
Invalid username or password
```

Langkah selanjutnya, kita dapat mencoba SQL injection untuk mem-bypass login page. Masukan perintah berikut di username:

```
' or true --
```

Perhatikan ada "spasi" sesudah -- , jika "spasi" tidak di tambahkan sesudah -- maka SQL injection ini tidak akan jalan. Biarkan password kosong. Tekan tombol login.

Simsalabim! script akan memperlihatkan semua message yang harusnya untuk user yang terauthentikasi!

Set Up Rules / Aturan

Untuk membuat hidup kita lebih mudah, akan banyak aturan yang di install bersama dengan mod_security. Hal ini di sebut CRS (Core Rule Set) dan lokasinya di

```
ls -l /usr/share/modsecurity-crs/

total 44
drwxr-xr-x 2 root root  4096 Mar 31 06:15 activated_rules
drwxr-xr-x 2 root root  4096 Mar 31 06:15 base_rules
drwxr-xr-x 2 root root  4096 Mar 31 06:15 experimental_rules
drwxr-xr-x 2 root root  4096 Mar 31 06:15 lua
```

```
-rw-r--r-- 1 root root 13774 Jul 13  2013 modsecurity_crs_10_setup.conf
drwxr-xr-x 2 root root  4096 Mar 31 06:15 optional_rules
drwxr-xr-x 2 root root  4096 Mar 31 06:15 slr_rules
drwxr-xr-x 8 root root  4096 Mar 31 06:15 util
```

Dokumen tersedia di

```
ls -l /usr/share/doc/modsecurity-crs/

total 16
-rw-r--r-- 1 root root  623 Jul 12  2013 changelog.Debian.gz
-rw-r--r-- 1 root root 1297 Jul  2  2012 copyright
-rw-r--r-- 1 root root 1138 Mar 16  2012 README.Debian
-rw-r--r-- 1 root root 1485 Jul  2  2013 README.md
```

Untuk me-load aturan / rules ini, kita perlu memberitahukan Apache untuk melihat directory tersebut. Edit file mod-security.conf

```
vi /etc/apache2/mods-enabled/security2.conf
```

Tambahkan aturan berikut di dalam <IfModule security2_module> </IfModule>:

```
Include "/usr/share/modsecurity-crs/*.conf"
Include "/usr/share/modsecurity-crs/activated_rules/*.conf"
```

Directory activated_rules sama dengan directory mods-enabled pada Apache. Rules / Aturan yang ada tersedia di directory:

```
/usr/share/modsecurity-crs/base_rules
/usr/share/modsecurity-crs/optional_rules
/usr/share/modsecurity-crs/experimental_rules
```

Kita perlu membuat symlinks di dalam directory activated_rules untuk mengaktifkan aturan tersebut.

Contoh, untuk mengaktifkan rules / aturan SQL injection.

```
cd /usr/share/modsecurity-crs/activated_rules/
ln -s /usr/share/modsecurity-crs/base_rules/modsecurity_crs_41_sql_injection_attacks.conf .
```

Apache harus di reload agar rules beroperasi / berefek.

```
service apache2 reload
```

Sekarang buka halaman login yang kita buat sebelumnya dan coba menggunakan SQL injection query pada kolom username. Jika kita sudah mengubah SecRuleEngine menjadi On, kita akan melihat 403 Forbidden error.

Jadi semua tergantung pada opsi DetectionOnly, injection akan berhasil tapi akan tercatat pada file modsec_audit.log .

Menulis Rules ModSecurity sendiri

Pada bagian ini, kita akan mencoba membuat aturan yang akan memblok request juga ada kata-kata "yang tidak di inginkan" di masukan ke dalam form HTML.

Pertama-tama, kita akan membuat script PHP yang akan mengambil input dari textbox dan akan menampilkannya kembali ke user.

/var/www/html/form.php

isinya

```
<html>
   <body>
      <?php
         if(isset($_POST['data']))
            echo $_POST['data'];
         else
         {
      ?>
            <form method="post" action="">
                  Enter something here:<textarea name="data"></textarea>
                  <input type="submit"/>
            </form>
      <?php
         }
      ?>
   </body>
</html>
```

Custom rules dapat di tambahkan ke semua file konfigurasi atau di letakan di directory modsecurity. Kita akan mencoba untuk menempatkan aturan / rules kita di sebuah file baru:

vi /etc/modsecurity/modsecurity_custom_rules.conf

Tambahkan perintah kalimat berikut:

```
SecRule REQUEST_FILENAME "form.php" "id:'400001',chain,deny,log,msg:'Spam detected'"
SecRule REQUEST_METHOD "POST" chain
SecRule REQUEST_BODY "@rx (?i:(pills|insurance|rolex))"
```

Save file kemudian reload Apache. Buka lagi

http://websiteanda.com/form.php

atau (contoh)

http://192.168.0.100/form.php

Masuk ke browser dan masukan text yang ada kata-kata: pills, insurance, rolex.

Anda akan akan melihat halman 403 dan catatan di log, atau hanya catatan di log berdasarkan konfigurasi SecRuleEngine. Sintaks untuk SecRule adalah

SecRule VARIABLES OPERATOR [ACTIONS]

Disini kita menggunakan chain action untuk mencocokan variable REQUEST_FILENAME dengan form.php, REQUEST_METHOD dengan POST dan REQUEST_BODY dengan regular expression (@rx) string (pills|insurance|rolex). Variable ?i: melakukan pencocokan yang tidak case sensitive. Jika ke tiga aturan tersebut berhasil cocok dengan baik, ACTION yang dilakukan adalah menolak dan mencatat di log dengan message "Spam detected." Chain action mensimulasi operasi logical AND untuk mencocokan ke tiga rules / aturan.
[edit] Excluding Host dan Directory

Kadang kala ada baiknya kita meng-exclude directory tertentu atau domain name jika dia menjalankan aplikasi seperti phpMyAdmin karena modsecurity akan memblok SQL queries. Juga sebaiknya meng-exclude admin backend dari aplikasi CMS seperti WordPress.

Untuk men-disable modsecurity untuk sebuah VirtualHost lakukan / tambahkan sebagai berikut

```
<IfModule security2_module>
   SecRuleEngine Off
</IfModule>
```

dalam bagian <VirtualHost>

Untuk directory tertentu:

```
<Directory "/var/www/wp-admin">
   <IfModule security2_module>
      SecRuleEngine Off
   </IfModule>
</Directory>
```

Jika kita tidak ingin mem-disable secara penuh modsecurity, gunakan SecRuleRemoveById untuk membuang aturan / rule / rule chain tertentu dengan menentukan ID-nya sebagai berikut

```
<LocationMatch "/wp-admin/update.php">
   <IfModule security2_module>
      SecRuleRemoveById 981173
   </IfModule>
</LocationMatch>
```

BAB 7 Intrusion Detection System (IDS)

Snort adalah free dan open source network intrusion prevention system (NIPS) dan network intrusion detection system (NIDS), yang di kembangkan oleh Martin Roesch tahun 1998. Snort saat ini di kembangkan oleh Sourcefire, dimana Roesch adalah founder dan CTO. Tahun 2009, Snort terpilih sebagai ‘greatest open source software of all time.’

Penggunaan SNORT

Snort’s open source network-based intrusion detection system (NIDS) yang mempunyai kemampuan untuk melakukan analisa traffic secara real time dan packet logging dari jaringan Internet Protocol (IP). Snort melakukan analisa protocol, pencarian content, pencocokan content. Snort juga dapat digunakan untuk mendeteksi serangan, termasuk, tapi tidak terbatas pada, operating system fingerprinting attempts, common gateway interface, buffer overflows, server message block probes, dan stealth port scans.

Snort dapat dikonfigurasi menggunakan tiga mode utama: sniffer, packet logger, dan network intrusion detection.

- Mode sniffer - snort akan membaca paket yang lewat dan menampilkan ke layar.
- Mode logger - snort akan mencatat paket yang lewat ke disk.
- Mode Intrusion Detection - snort akan memonitor semua traffic yang lewat, membandingkan dengan rule yang di definisikan oleh user.

Ada beberapa tool yang bisa di kawinkan dengan snort untuk administrasi, reporting, dan analisa log, seperti,

- Snorby
- BASE
- RazorBack

Install SNORT di Ubuntu 16.04

Cek Jaringan

```
ifconfig
```

catat nama interface yang nanti akan di monitor

```
ens18     Link encap:Ethernet  HWaddr 66:31:34:63:65:31
          inet addr:192.168.0.100  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::6431:34ff:fe63:6531/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:26658 errors:0 dropped:11 overruns:0 frame:0
          TX packets:9441 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:37165428 (37.1 MB)  TX bytes:751808 (751.8 KB)
```

maka interface yang dimonitor adalah

ens18

Siapkan Aplikasi Pendukung

sudo locale-gen id_ID.UTF-8

apt update
apt install oinkmaster snort snort-common snort-rules-default snort-doc

Akan di tanya

- interface yang akan di monitor, misalnya ens18
- range IP yang di monitor, misalnya 192.168.0.0/16

Cek Snort

snort -C

Jalankan Snort mode NIDS

snort -dev -l /var/log/snort/ -h 192.168.0.0/16 -c /etc/snort/snort.conf &

kalau ingin supaya bisa di baca di kemudian hari oleh wireshark harus di simpan dalam bentuk binary, dengan perintah

/usr/sbin/snort -m 027 -b -l /var/log/snort/ -u agung -c /etc/snort/snort.conf -S HOME_NET=[192.168.0.0/16] -i ens18 &

Supaya tidak rewel, sebaiknya permission /var/log/snort di jadikan

chmod 770 /var/log/snort

ini sebetulnya cara yang tidak baik.

SNORT: sniffer mode

Perintah yang penting

snort -v
snort -vd
snort -vde

Jika kita ingin melihat TCP/IP header di layar (sniffer mode), jalankan perintah

snort -v

Jika kita juga ingin melihat data aplikasi / payload yang di bawa oleh paket, gunakan perintah

```
snort -vd
```

Jika ingin lebih lengkap lagi dengan penjelasan, dan header data link layer, gunakan perintah,

```
snort -vde
```

atau

```
snort -d -v -e
```

Kita bisa melihat MAC address sumber, MAC address tujuan, IP address sumber & tujuan, semua parameter TCP dan juga data yang di bawa.

SNORT: packet logger mode

Snort adalah aplikasi untuk pendeteksi penyusup yang masuk ke jaringan komputer kita. Salah satu fitur yang akan sangat bermanfaat adalah merekam semua data yang terdengar oleh komputer yang mengoperasikan snort. Teknik ini biasanya akan sangat bermanfaat untuk melakukan proses forensic pada saat terjadi kejadian cybercrime. Pada kesempatan ini kita akan membahas lebih dalam tentang teknik melakukan perekaman paket data menggunakan snort.

Mode untuk merekam paket pada snort sering di sebut packet logger mode. Format rekaman paket merupakan format pcap library. Oleh karena itu, bisa dengan mudah di baca oleh aplikasi penyadapan seperti wireshark.

Folder untuk Merekam

Teknik merekam paket tidak berbeda jauh dengan teknik sniffer pada snort. Bedanya kita hanya perlu memberitahukan folder yang digunakan sebagai tempat untuk menyimpan data yang di rekam. Folder tersebut diberitahukan menggunakan switch -l, sebagai berikut,

```
snort -dev -l ./log
snort -dev -l /var/log/snort
```

Ketika Snort berjalan dalam mode ini, ia mengumpulkan setiap paket yang dilihatnya dan menempatkannya dalam hirarki direktori berdasarkan alamat IP dari salah satu host di datagram

Jika Kita hanya menggunakan switch -l saja, akan melihat snort kadang kala menggunakan alamat remote komputer sebagai directory di mana ia menempatkan paket dan kadang-kadang menggunakan alamat host lokal. Agar menyimpan log relatif ke jaringan lokal, Kita perlu memberitahukan snort mana yang home network / jaringan lokal:

```
snort -dev -l /var/log/snort -h 192.168.0.0/24
```

Perintah ini memberitahu Snort bahwa Kita ingin mencetak data link dan TCP/IP header serta data aplikasi ke dalam direktori /var/log/snort, dan Kita ingin log paket relatif terhadap jaringan kelas C 192.168.0.0. Semua paket yang masuk akan disimpan ke subdirektori dari direktori log, dengan nama direktori yang berbasis pada alamat remote host (non-192.168.0).

Catatan: Perhatikan bahwa jika kedua sumber dan tujuan host di jaringan lokal yang sama, mereka akan di catat ke direktori dengan nama berdasarkan nomor port yang lebih tinggi lebih dulu, jika sama, maka digunakan alamat sumber.

Logging Biner

Jika Kita berada di jaringan kecepatan tinggi atau Kita ingin log paket ke bentuk yang lebih kompak untuk analisis nanti, Kita harus mempertimbangkan logging dalam mode biner. Log paket dalam mode biner dalam format tcpdump ke file biner tunggal dalam direktori logging:

```
snort -l /var/log/snort -b
```

Perhatikan baris perintah perubahan di sini. Kita tidak perlu menentukan jaringan rumah lagi karena mode biner menyimpan semua ke dalam satu file, yang menghilangkan kebutuhan untuk memberitahukan bagaimana format struktur direktori output. Selain itu, kita tidak perlu berjalan dalam mode verbose atau menentukan switch -d atau -e karena dalam mode biner seluruh paket di catat, bukan hanya sebagian. Yang benar-benar Kita perlu lakukan untuk menempatkan Snort ke mode logger adalah untuk menentukan direktori logging pada baris perintah menggunakan switch -l sementara switch binary logging -b hanya memberitahu Snort untuk log paket dalam format output selain default teks ASCII biasa.

Membaca Log

Setelah paket telah dicatat ke file biner, Kita dapat membaca paket kembali dari file dengan sniffer yang mendukung format biner tcpdump (seperti tcpdump, Ethereal, atau wireshark). Snort juga dapat membaca paket kembali dengan menggunakan switch -r, yang menempatkan ke mode playback. Paket dari setiap tcpdump diformat file yang dapat diproses melalui Snort dalam mode yang dijalankan. Misalnya, jika Kita ingin menjalankan file log biner melalui Snort dalam mode sniffer untuk menampilkan paket ke layar, Kita dapat mencoba sesuatu seperti ini:

```
snort -dv -r packet.log
```

Kita dapat memanipulasi data di file dengan sejumlah cara melalui packet logging dan mode deteksi intrusi, serta dengan antarmuka BPF yang tersedia dari baris perintah. Misalnya, jika Kita hanya ingin melihat paket ICMP dari file log, hanya menentukan filter BPF pada baris perintah dan Snort hanya akan melihat paket ICMP dalam file:

```
snort -dvr packet.log icmp
```

Untuk informasi lebih lanjut bagaimana cara menggunakan interface BPF, ada baiknya membaca manual tcpdump dan snort.

Logging ASCII

Bagi kita yang hanya ingin membaca rekaman traffic dengan mudah menggunakan format text ASCII dapat menggunakan perintah berikut,

```
snort -A console -c /etc/snort/snort.conf -l /var/log/snort/ -K ascii
```

Jika dibuang -A console, maka hasil rekaman tidak akan di tampilkan di layar / console. Perintah yang digunakan adalah,

```
snort -c /etc/snort/snort.conf -l /var/log/snort/ -K ascii
```

Kita bisa melihat hasilnya pada folder

/var/log/snort

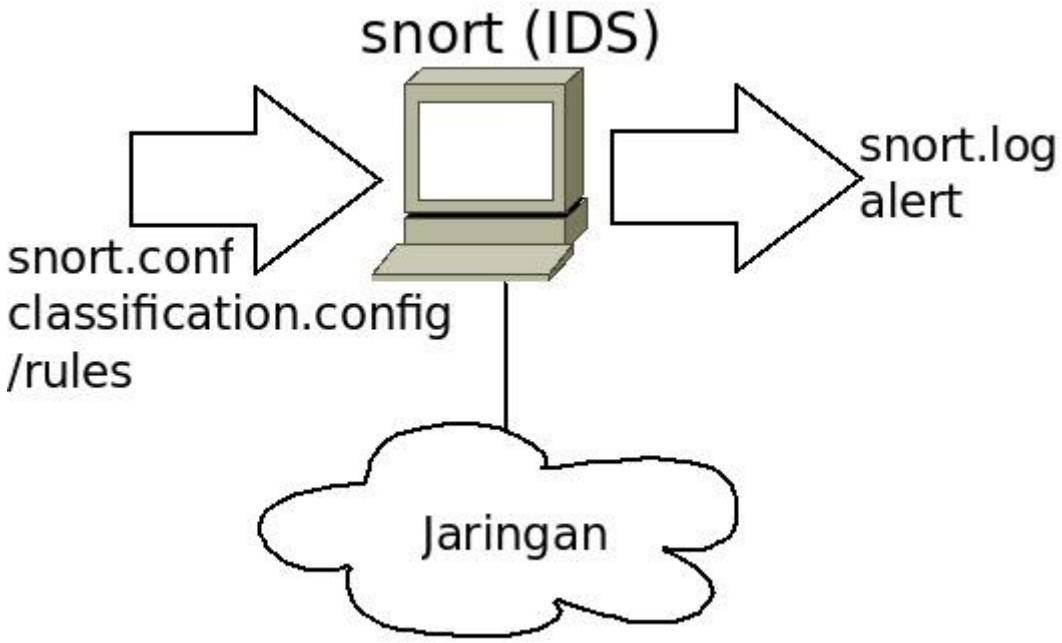
akan ada

- file snort.log - berbentuk binary, bisa di reply di wireshark.
- file alert - berbentuk ASCII, berisi laporan pelanggaran.
- folder IP address pelanggar, berisi file rekaman pelanggaran berbentuk ASCII.

SNORT: mode IDS

Pertahanan di dunia cyber akan sangat mengandalkan sensor pendeteksi penyusup yang biasa di kenal sebagai Intrusion Detection System (IDS) seperti di terangkan di artikel tentang IDS. Ada beberapa IDS open source yang bisa kita gunakan untuk mendeteksi penyusup, salah satu yang paling populer adalah snort. Teknik instalasi snort telah di jelaskan dengan detail di artikel instalasi snort .

Kecanggihan snort yang berpenampilan sangat sederhana karena memang hanya CLI, akan tampak cemerlang pada saat kita operasikan sebagai pendeteksi penyusup atau dalam bahasa keren-nya Intrusion Detection System (IDS). Pada kesempatan ini, akan di jelaskan snort untuk mendeteksi penyusup.



Pada gambar di atas, di perlihatkan konfigurasi snort yang di operasikan untuk medeteksi penyusup. Snort bisa di operasikan sebagai sebuah mesin secara independen / berdiri sendiri, atau di pasang di titik-titik strategis di jaringan untuk menangkap paket yang berseliweran.

Kira-kira cara berfikir snort seperti anti-virus di komputer menganalisa file, snort menganalisa paket yang lewat dan di bandingkan dengan referensi serangan yang ada. Semua konfigurasi snort ada di /etc/snort. Secara umum snort akan membaca minimal 2 file penting, yaitu, snort.conf yang

berisi konfigurasi snort, dan classification.config, yang berisi klassifikasi pelanggaran yang terjadi. Snort.conf yang akan menentukan bagaimana pola pendeteksian penyusup / serangan dilakukan. Daftar catatan pola serangan ada di folder /etc/snort/rules.

Berbasis pada snort.conf dan rules yang ada, maka snort akan mencatat alert yang ada di file alert dan juga akan merekam semua traffic yang dia dengar di snort.log yang nantinya bisa digunakan untuk kebutuhan forensik jaringan.

Konfigurasi snort.conf default biasanya mencukup untuk operasi sederhana. Mungkin yang justru akan banyak di ubah adalah file di bawah rules. Khususnya

```
/etc/snort/rules/local.rules
```

yang akan mencerminkan trap untuk jenis serangan tertentu.

Jika semua sudah di konfigurasi dengan benar maka untuk mengoperasikan snort sebagai IDS sebetulnya sangat mirip dengan snort untuk merekam / me-log traffic jaringan, hanya kita menambahkan switch -D untuk membuat snort sebagai daemon / server, misalnya menggunakan perintah

```
snort -c /etc/snort/snort.conf -l /var/log/snort/ -D
```

Untuk memastikan bahwa snort telah berjalan dengan baik, bisa ketik

```
ps ax
```

SNORT-RULES: Coba Menulis Rules untuk pemula

Sebuah pendeteksi penyusup / Intrusion Detection System (IDS) logika bekerjanya seperti anti-virus, engine pendeteksi hanya akan bekerja dengan benar kalau database-nya benar. Kalau serangan tersebut tidak ada dalam database maka serangan tersebut tidak akan terdeteksi. Sial-nya database tersebut adalah buatan manusia, yang harus telaten mendokumentasinya bentuk paket / isi paket serangan, kemudian menuangkannya ke dalam sebuah aturan / rules.

Snort Intrusion Detection System (IDS) mempunyai kemampuan yang baik untuk membaca paket yang lewat di jaringan. Snort IDS mirip dengan tcpdump / wireshark, tetapi memiliki output yang lebih bersih dan bahasa aturan yang lebih fleksibel. Sama seperti tcpdump / wireshark, snort akan mendengarkan antarmuka tertentu, atau membaca jejak paket dari sebuah file. Umumnya administrator keamanan diminta untuk melihat jejak paket untuk menganalisa serangan yang terjadi. Salah satu yang nampaknya akan amat sangat bermanfaat adalah kemampuan untuk menulis snort rules untuk mendeteksi serangan. Disini kita akan belajar bagaimana menggunakan snort untuk membaca jejak dan belajar bagaimana menulis aturan / rules baru.

Tujuan utama snort sebagai IDS adalah untuk bereaksi jika ada rules yang cocok dengan paket yang masuk. Reaksi yang di berikan Snort IDS bisa bermacam-macam tergantung kebutuhan / kemauan yang memprogram, bisa di catat / "log", bisa memberikan "alert" bagi administrator keamanan jaringan.

Alat Yang Dibutuhkan

server dengan snort yang di instalasi

Bacaan

- www.snort.org
- http://www.snort.org/docs/FAQ.txt - Snort FAQ
- http://www.snort.org/docs/snort_manual/node2.html - Snort Overview
- http://www.snort.org/docs/snort_manual/node16.html - How to Write Snort Rules and Keep Your Sanity

Beberapa perintah bermanfaat

Melihat perintah snort

```
snort –help
```

Contoh membaca log

```
snort –r /tmp/snort-ids-lab.log -P 5000 –c /tmp/rules –e –X -v
```

Versi snort yang baru punya masalah saat membaca checksum paket yang tidak benar. Kita perlu menambahkan kalimat

```
config checksum_mode : none
```

di bagian atas rules file jika kita memperoleh checksum problem.

Logging ASCII agar local.rules bisa di baca dengan mudah

```
snort -c /etc/snort/snort.conf -l /var/log/snort/ -K ascii -D
```

Rule Sederhana

Jika kita instalasi snort dengan baik maka, semua aturan snort biasanya di simpan di folder /etc/snort/rules. Aturan tersebut berupa file-file dengan nama yang sesuai dengan kategori serangan, misalnya, virus.rules adalah file berisi aturan snort yang akan mendeteksi paket yang kemungkinan membawa virus.

Bagi kita yang akan berexperimen dengan aturan snort, sebaiknya mengedit file /etc/snort/rules/local.rules. Beri keterangan dengan tanda '#' di depan-nya. Keterangan tersebut perlu dibuat untuk mengingatkan kita tentang percobaan / aturan apa yang kita buat.

Secara umum, semua rules mengikuti aturan:

```
action protocol address port direction address port (rule option)
```

Penjelasan format,

- Pilihan action adalah "log" atau "alert". "alert" akan menuliskan semua alert ke sebuah file "alert" yang sama. Sementara log akan menyimpan traffic untuk masing-masing IP address yang bermasalah pada sebuah folder untuk di analisa lebih lanjut.
- Bagian protocol harus di isi "tcp", "udp",or "icmp". "Any" tidak di ijinkan.

- Address dapat berupa notasi CIDR
- Port dapat menggunakan range dan operator "!. Contoh log paket ke sekumpulan mesin dengan port tidak antara 6000-6010

  log tcp any any -> 192.168.1.0/24 !6000:6010

- Operator arah "->" atau "<-" atau "<>" untuk traffic bi-directional antara dua address.

Disini kita akan belajar untuk membuat rules pendeteksi traffic telnet. Mengapa telnet perlu di deteksi? karena memang telnet sangat rentan untuk di sadap.

Semua aplikasi pada jaringan TCP/IP akan bekerja menggunakan nomor port tertentu. Telnet menggunakan nomor port 23. Aplikasi lain akan menggunakan nomor port yang lain, misalnya, web 80, https 443, smtp 25, pop3 110, imap 143. Daftar sebagian besar nomor port yang digunakan oleh aplikasi Internet bisa di lihat di file /etc/services.

Setelah mengetahui nomor port telnet 23, maka kita dapat membuat aturan snort sederhana untuk mendeteksi telnet, sebagai berikut,

alert tcp any any -> 192.168.0.100 23 (msg: "Ada yang telnet ke mesin!"; sid:1000001;)

arti dari rule di atas adalah,

- action: kasi tanda bahaya ("alert")
- semua paket ke telnet port (port 23)
- ke mesin 192.168.0.100
- tambahkan string yang bisa di baca admin "Ada yang telnet ke mesin!"
- sid - rule ID start dari 1000000

Rule option

Rule option dapat menentukan task yang harus dilakukan jika address dan protocol cocok. Contoh, snort rule untuk menangkap semua ICMP echo message,

alert tcp any any -> 192.168.10.2 any (itype: 8; msg: "ping detected";)

perhatikan mengapa itype = 8?

Catatan rule option:

- harus berada dalam kurung ( )
- harus di akhiri dengan ;
- Opsi yang menarik untuk di explorasi "content", "flags", dan ipoption".

Contoh Rule

alert tcp $HOME_NET 23 -> $EXTERNAL_NET any (msg:"TELNET login incorrect"; content:"Login incorrect"; flags: A+; reference:arachnids,127;)

alert udp $EXTERNAL_NET any -> $HOME_NET 53 (msg:"EXPLOIT BIND Tsig Overflow Attempt"; content:"|80 00 07 00 00 00 00 00 01 3F 00 01 02|/bin/sh";)

alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"SCAN FIN"; flags: F; reference:arachnids,27;)

alert tcp $EXTERNAL_NET any -> $HOME_NET 23 (msg:"MISC linux rootkit attempt lrkr0x";flags: A+; content:"lrkr0x";)5. alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-CGI view-source access ";flags: A+; content:"/view-source?../../../../../../../etc/passwd"; nocase;reference:cve,CVE-1999-0174;)

alert icmp any any -> any any (msg:"ICMP Source Quench"; itype: 4; icode: 0;)

alert tcp $EXTERNAL_NET any -> $HOME_NET 53 (msg:"DNS EXPLOIT named overflow";flags: A+;content:"thisissometempspaceforthesockinaddrinyeahyeahiknowthisislamebutanyway whocareshorizongotitworkingsoalliscool"; reference:cve,CVE-1999-0833;)

alert tcp $EXTERNAL_NET any -> $HOME_NET 139 (msg:"NETBIOS SMB ADMIN$access"; flow:to_server,established; content:"\\ADMIN$|00 41 3a 00|"; reference:arachnids,340; classtype:attempted-admin; sid:532; rev:4;)

alert ip $EXTERNAL_NET $SHELLCODE_PORTS -> $HOME_NET any (msg:"SHELLCODE sparc NOOP"; content:"|a61c c013 a61c c013 a61c c013 a61c c013|"; reference:arachnids,355; classtype:shellcode-detect; sid:646; rev:4;)

Restart Snort

Supaya local.rules bisa jalan dengan baik, logging ASCII agar local.rules bisa di baca dengan mudah

```
killall snort
killall snort
snort -c /etc/snort/snort.conf -l /var/log/snort/ -K ascii -D
```

BAB 8 Pertahanan Host

Tripwire

Tripwire adalah termasuk kategori Host Intrusion Detection System (IDS). Yang mendeteksi perubahan file di mesin yang mungkin dilakukan oleh penyerang.

Logika bekerja tripwire adalah dengan membuat baseline database dari file yang ada di system. Jika file tersebut berubah maka tripwire akan mencatat dan / atau memberitahukan administrator mesin.

Instalasi tripwire

Instalasi

```
apt-get install tripwire
```

masukan password

```
Enter site key passphrase
Enter local key passphrase
```

Pastikan konfigurasi tripwire aman dan hanya bisa di akses oleh root saja.

```
cd /etc/tripwire
chmod 0600 tw.cfg tw.pol
```

Edit Policy

edit policy

```
vi /etc/tripwire/twpol.txt
```

encrypt policy

```
cd /etc/tripwire
twadmin --create-polfile --cfgfile ./tw.cfg --site-keyfile ./site.key ./twpol.txt
```

Edit Configurasi

edit konfigurasi

```
vi /etc/tripwire/twcfg.txt
```

encrypt konfigurasi

```
cd /etc/tripwire
twadmin --create-cfgfile --cfgfile ./tw.cfg --site-keyfile ./site.key ./twcfg.txt
```

Inisialisasi Database

Inisialisasi baseline database

```
tripwire --init --cfgfile /etc/tripwire/tw.cfg \
--polfile /etc/tripwire/tw.pol --site-keyfile /etc/tripwire/site.key \
--local-keyfile /etc/tripwire/HOSTNAME-local.key
```

atau jika HOSTNAME anda adalah ubuntu maka

```
tripwire --init --cfgfile /etc/tripwire/tw.cfg \
--polfile /etc/tripwire/tw.pol --site-keyfile /etc/tripwire/site.key \
--local-keyfile /etc/tripwire/ubuntu-local.key
```

Ini akan membutuhkan waktu beberapa lama karena dia akan mencek seluruh harddisk.

Check System

Untuk mencek apakah terjadi perubahan file kita dapat melakukan

```
tripwire --check
```

Untuk server yang beroperasi 24/7 kita dapat menggunakan cron dan e-mail hasilnya ke administrator.

Update policy

Jika kita mengupdate policy, misalnya menambahkan / mengurangi folder yang akan di scan dll kita dapat melakukan edit policy

```
vi /etc/tripwire/twpol.txt
```

kemudian update policy

```
tripwire --update-policy --cfgfile ./tw.cfg --polfile ./tw.pol \
--site-keyfile ./site.key --local-keyfile ./HOSTNAME-local.key ./twpol.txt
```

atau jika HOSTNAME yang digunakan ubuntu maka

```
tripwire --update-policy --cfgfile ./tw.cfg --polfile ./tw.pol \
--site-keyfile ./site.key --local-keyfile ./ubuntu-local.key ./twpol.txt
```

Update secara regular

Kita perlu mengupdate secara periodik database tentang file system. Mohon di cek dulu sebelum melakukan update. Proses update dapat menggunakan perintah

```
tripwire --update -Z low
```

perintah di atas akan melakukan perbandingan antara database yang ada dengan file yang ada di system. Kemudian jalankan editor untuk memilih perubahan di database.

Jika kita menjalankan perintah ini dan memperoleh message error karena tidak ada file report, sebab utamanya kemungkinan karena check yang dilakukaan belakangan tidak dilakukan sesudah update. File report berada di folder

```
/var/lib/tripwire/report
```

dan menggunakan hostname sebagai nama, dilanjutkan dengan tanggal (yyyymmdd) dan waktu (tttttt). Jika kita baru saja menjalankan check dan menginginkan update untuk dilakukan menggunakan report file terakhir, maka kita dapat mengunakan opsi -r dan menggunakan report file terakhir

```
tripwire --update -Z low --twrfile host-yyyymmdd-tttttt.twr
```

Local Manual

```
/usr/share/doc/tripwire/README.Debian
```

Lynis

Lynis adalah aplikasi audit keamanan open-source berbasis host yang dapat mengevaluasi profil keamanan dan postur Linux dan sistem operasi mirip UNIX lainnya.

Dalam tutorial ini, Anda akan menginstal Lynis dan menggunakannya untuk melakukan audit keamanan pada server Ubuntu 16.04 Anda. Kemudian Anda akan mengeksplorasi hasil audit contoh, dan mengkonfigurasi Lynis untuk melewatkan tes yang tidak sesuai dengan kebutuhan anda.

Lynis tidak akan melakukan hardening sistem secara otomatis. Tetapi akan menawarkan saran yang menunjukkan bagaimana anda bisa melakukan pengerasan sistem sendiri. Dengan demikian, akan sangat membantu jika anda memiliki pengetahuan dasar tentang keamanan sistem Linux. Anda juga harus terbiasa dengan layanan yang berjalan pada mesin yang akan anda audit, seperti server web, database, dan layanan lain yang mungkin dipindai oleh Lynis secara default. Ini akan membantu anda mengidentifikasi hasil yang dapat anda abaikan dengan aman.

Catatan: Melakukan audit keamanan membutuhkan waktu dan kesabaran. Anda mungkin ingin meluangkan waktu untuk membaca keseluruhan artikel sekali sebelum menginstal Lynis dan menggunakannya untuk mengaudit server Anda.

Prasyarat

Untuk bisa menjalankan artikel ini, Anda memerlukan:

- Sebuah Ubuntu Server 16.04
- user dengan kemampuan sudo
- firewall

Step 1 — Instal Lynis di Server

Ada beberapa cara untuk menginstal Lynis. Anda dapat mengkompilasi dari sumber, mendownload dan menyalin biner ke lokasi yang sesuai pada sistem, atau Anda dapat menginstalnya menggunakan manajer paket. Menggunakan manajer paket adalah cara mudah untuk menginstal Lynis dan memperbaruinya, jadi itulah metode yang akan kita gunakan.

Namun, di Ubuntu 16.04, versi yang tersedia dari repositori bukanlah versi terbaru. Agar memiliki akses ke fitur terbaru, kami akan menginstal Lynis dari repository proyek.

Repositori perangkat lunak Lynis menggunakan protokol HTTPS, jadi kami harus memastikan bahwa dukungan HTTPS untuk pengelola paket telah terinstal. Gunakan perintah berikut untuk memeriksa:

```
dpkg -s apt-transport-https | grep -i status
```

Jika di instalasi, keluarnya adalah

```
Status: install ok installed
```

Jika belum di install, install menggunakan,

```
sudo apt-get install apt-transport-https
```

Sebelum menginstalasi Lynis, jika di perlukan tambahkan repository key

```
sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys
C80E383C3DE9F082E01391A0366C67DE91CA5D5F
```

Output

```
Executing: /tmp/tmp.AnVzwb6Mq8/gpg.1.sh --keyserver
keyserver.ubuntu.com
--recv-keys
C80E383C3DE9F082E01391A0366C67DE91CA5D5F
gpg: requesting key 91CA5D5F from hkp server keyserver.ubuntu.com
gpg: key 91CA5D5F: public key "CISOfy Software (signed software packages)
<software@cisofy.com>" imported
gpg: Total number processed: 1
gpg:               imported: 1  (RSA: 1)
```

Tambahkan repo Lynis,

```
sudo add-apt-repository "deb [arch=amd64]
https://packages.cisofy.com/community/lynis/deb/ xenial main"
```

Instal Lynis,

```
sudo apt-get update
sudo apt-get install lynis
```

Step 2 – Lakukan Audit

Perintah yang bisa dilakukan Lynis,

```
lynis show commands
```

Output

```
Commands:
 lynis audit
 lynis configure
 lynis show
 lynis update
 lynis upload-only
```

Lihat Lynis default profile,

```
lynis show settings
```

Output

```
# Colored screen output
colors=1

# Compressed uploads
compressed-uploads=0

# Use non-zero exit code if one or more warnings were found
error-on-warnings=0
...

# Upload server (ip or hostname)
upload-server=[not configured]

# Data upload after scanning
upload=no

# Verbose output
verbose=0

# Add --brief to hide descriptions, --configured-only to show configured items only, or
--nocolors to remove colors
```

Cek versi / update info

```
lynis update info
```

Output

```
 == Lynis ==
```

```
  Version          : 2.4.8
  Status           : Up-to-date
  Release date     : 2017-03-29
  Update location  : https://cisofy.com/lynis/
2007-2017, CISOfy - https://cisofy.com/lynis/
```

Alternatif lain,

```
lynis update check
```

Output

```
status=up-to-date
```

Untuk menjalankan audit sistem anda, gunakan perintah lynis audit system. Anda dapat menjalankan Lynis dengan privilege dan non-privilege (pentest) mode. Dalam mode yang terakhir, beberapa tes yang memerlukan hak akses root dilewati. Untuk itu sebaiknya menjalankan lynis menggunakan sudo

```
sudo lynis audit system
```

Ketika Lynis melakukan audit, ia menjalani sejumlah tes, terbagi dalam beberapa kategori. Setelah setiap audit, hasil tes, informasi debug, dan saran untuk pengerasan sistem ditulis ke standar output (layar). Informasi lebih rinci dicatat ke /var/log/lynis.log, sementara data laporan disimpan ke /var/log/lynis-report.dat. Data laporan berisi informasi umum tentang server dan aplikasi itu sendiri, jadi file yang harus anda perhatikan adalah file log. File log dibersihkan (ditimpa) pada setiap audit, jadi hasil dari audit sebelumnya tidak disimpan.

Setelah audit selesai, Anda akan meninjau hasilnya, peringatan, dan saran, dan kemudian menerapkan saran yang relevan.

Mari kita lihat hasil audit Lynis yang dilakukan pada mesin yang digunakan untuk menulis tutorial ini. Hasil yang anda lihat di audit anda mungkin berbeda, namun anda tetap bisa mengikuti.

Bagian penting pertama dari hasil audit Lynis adalah murni informasi. Ini memberitahu anda hasil dari setiap tes, dikelompokkan berdasarkan kategori. Informasi itu berupa kata kunci, seperti NONE, WEAK, DONE, FOUND, NOT_FOUND, OK, dan WARNING.

Output

```
[+] Boot and services
------------------------------------
 - Service Manager                          [ systemd ]
 - Checking UEFI boot                         [ DISABLED ]
 - Checking presence GRUB                       [ OK ]
 - Checking presence GRUB2                       [ FOUND ]
   - Checking for password protection             [ WARNING ]

..
```

```
[+] File systems
------------------------------------
 - Checking mount points
   - Checking /home mount point                [ SUGGESTION ]
   - Checking /tmp mount point                 [ SUGGESTION ]
   - Checking /var mount point                 [ OK ]
 - Query swap partitions (fstab)               [ NONE ]
 - Testing swap partitions                     [ OK ]
 - Testing /proc mount (hidepid)               [ SUGGESTION ]
 - Checking for old files in /tmp              [ OK ]
 - Checking /tmp sticky bit                    [ OK ]
 - ACL support root file system                [ ENABLED ]
 - Mount options of /                          [ OK ]
 - Checking Locate database                    [ FOUND ]
 - Disable kernel support of some filesystems
   - Discovered kernel modules: udf

...

[+] Hardening
------------------------------------
 - Installed compiler(s)                       [ FOUND ]
 - Installed malware scanner                   [ NOT FOUND ]
 - Installed malware scanner                   [ NOT FOUND ]

...

[+] Printers and Spools
------------------------------------
 - Checking cups daemon                        [ NOT FOUND ]
 - Checking lp daemon                          [ NOT RUNNING ]
```

Meskipun Lynis melakukan lebih dari 200 tes di luar kotak, tidak semua diperlukan untuk server Anda. Bagaimana Anda bisa tahu tes mana yang perlu dan mana yang tidak? Di situlah beberapa pengetahuan tentang apa yang seharusnya atau tidak boleh dijalankan di server ikut bermain. Misalnya, jika anda memeriksa bagian hasil audit Lynis, anda akan menemukan dua tes di bawah kategori Printers and Spools:

Output

```
[+] Printers and Spools
------------------------------------
 - Checking cups daemon                        [ NOT FOUND ]
 - Checking lp daemon                          [ NOT RUNNING ]
```

Apakah anda benar-benar menjalankan server printer di server Ubuntu 16.04? Kecuali anda menjalankan server cetak berbasis cloud, anda tidak perlu Lynis menjalankan tes itu setiap saat.

Sementara itu adalah contoh yang langsung terlihat dari tes yang bisa anda lewatkan, yang lain tidak begitu jelas. Ambil bagian hasil parsial ini, misalnya:

Output

```
[+] Insecure services
-----------------------------------
 - Checking inetd status                    [ NOT ACTIVE ]
```

Output ini mengatakan bahwa inetd tidak aktif, tapi itu diharapkan pada server Ubuntu 16.04, karena Ubuntu mengganti inetd dengan systemd. Mengetahui hal itu, anda dapat memberi tag pada tes itu sebagai salah satu yang tidak boleh dilakukan Lynis sebagai bagian dari audit di server anda.

Step 3 – Memperbaiki Lynis Audit Warning

Hasil audit Lynis tidak selalu membawa bagian warning, namun bila memang demikian, Anda akan tahu cara memperbaiki masalah yang diangkat setelah membaca bagian ini.

Peringatan dicantumkan setelah bagian hasil. Setiap peringatan dimulai dengan teks peringatan itu sendiri, dengan tes yang menghasilkan peringatan pada baris yang sama dalam tanda kurung. Baris berikutnya akan berisi solusi yang disarankan, jika ada. Baris terakhir adalah URL kontrol keamanan di mana Anda mungkin menemukan beberapa petunjuk tentang peringatan tersebut. Sayangnya, URL tidak selalu menawarkan penjelasan, jadi Anda mungkin perlu melakukan penelitian lebih lanjut.

Output

```
Warnings (3):
 ----------------------------
! Version of Lynis is very old and should be updated [LYNIS]
    https://cisofy.com/controls/LYNIS/

! Reboot of system is most likely needed [KRNL-5830]
    - Solution : reboot
      https://cisofy.com/controls/KRNL-5830/

! Found one or more vulnerable packages. [PKGS-7392]
      https://cisofy.com/controls/PKGS-7392/
```

Peringatan pertama mengatakan bahwa Lynis perlu diperbarui. Itu juga berarti audit ini menggunakan versi Lynis, sehingga hasilnya mungkin tidak lengkap. Ini bisa dihindari jika kami melakukan pemeriksaan versi dasar sebelum menjalankan hasilnya, seperti yang ditunjukkan pada Langkah 3. Perbaikan untuk yang satu ini mudah: update Lynis.

Peringatan kedua menunjukkan bahwa server perlu di-reboot. Itu mungkin karena pembaruan sistem yang melibatkan upgrade kernel dilakukan baru-baru ini. Solusinya disini adalah reboot sistem.

Bila ragu tentang peringatan apapun, atau hampir semua hasil tes, anda bisa mendapatkan lebih banyak informasi tentang tes tersebut dengan mengajukan pertanyaan kepada Lynis untuk test-id. Perintah untuk mencapainya yang mengambil formulir ini:

```
sudo lynis show details test-id
```

Jadi untuk peringatan kedua, yang memiliki test id KRNL-5830, kita bisa menjalankan perintah ini:

```
sudo lynis show details KRNL-5830
```

Output untuk tes tertentu berikut. Ini memberi anda gambaran tentang proses yang Lynis lakukan selama setiap tes yang dilakukannya. Dari keluaran ini, Lynis bahkan memberikan informasi spesifik tentang item yang menimbulkan peringatan:

Output

```
2017-03-21 01:50:03 Performing test ID KRNL-5830 (Checking if system is running on the latest installed kernel)
2017-03-21 01:50:04 Test: Checking presence /var/run/reboot-required.pkgs
2017-03-21 01:50:04 Result: file /var/run/reboot-required.pkgs exists
2017-03-21 01:50:04 Result: reboot is needed, related to 5 packages
2017-03-21 01:50:04 Package: 5
2017-03-21 01:50:04 Result: /boot exists, performing more tests from here
2017-03-21 01:50:04 Result: /boot/vmlinuz not on disk, trying to find /boot/vmlinuz*
2017-03-21 01:50:04 Result: using 4.4.0.64 as my kernel version (stripped)
2017-03-21 01:50:04 Result: found /boot/vmlinuz-4.4.0-64-generic
2017-03-21 01:50:04 Result: found /boot/vmlinuz-4.4.0-65-generic
2017-03-21 01:50:04 Result: found /boot/vmlinuz-4.4.0-66-generic
2017-03-21 01:50:04 Action: checking relevant kernels
2017-03-21 01:50:04 Output: 4.4.0.64 4.4.0.65 4.4.0.66
2017-03-21 01:50:04 Result: Found 4.4.0.64 (= our kernel)
2017-03-21 01:50:04 Result: found a kernel (4.4.0.65) later than running one (4.4.0.64)
2017-03-21 01:50:04 Result: Found 4.4.0.65
2017-03-21 01:50:04 Result: found a kernel (4.4.0.66) later than running one (4.4.0.64)
2017-03-21 01:50:04 Result: Found 4.4.0.66
2017-03-21 01:50:04 Warning: Reboot of system is most likely needed [test:KRNL-5830] [details:] [solution:text:reboot]
2017-03-21 01:50:04 Hardening: assigned partial number of hardening points (0 of 5). Currently having 7 points (out of 14)
2017-03-21 01:50:04 Checking permissions of /usr/share/lynis/include/tests_memory_processes
2017-03-21 01:50:04 File permissions are OK
2017-03-21 01:50:04 ===----------------------------------------------------------------===
```

Untuk peringatan ketiga, PKGS-7392, tentang vulnerable package, kami menjalankan perintah ini:

```
sudo lynis show details PKGS-7392
```

Output memberi kami lebih banyak informasi mengenai paket yang perlu diperbarui:

Output

```
2017-03-21 01:39:53 Performing test ID PKGS-7392 (Check for Debian/Ubuntu security updates)
2017-03-21 01:39:53 Action: updating repository with apt-get
2017-03-21 01:40:03 Result: apt-get finished
2017-03-21 01:40:03 Test: Checking if /usr/lib/update-notifier/apt-check exists
2017-03-21 01:40:03 Result: found /usr/lib/update-notifier/apt-check
2017-03-21 01:40:03 Test: checking if any of the updates contain security updates
2017-03-21 01:40:04 Result: found 7 security updates via apt-check
2017-03-21 01:40:04 Hardening: assigned partial number of hardening points (0 of 25). Currently having 96 points (out of 149)
2017-03-21 01:40:05 Result: found vulnerable package(s) via apt-get (-security channel)
2017-03-21 01:40:05 Found vulnerable package: libc-bin
2017-03-21 01:40:05 Found vulnerable package: libc-dev-bin
2017-03-21 01:40:05 Found vulnerable package: libc6
2017-03-21 01:40:05 Found vulnerable package: libc6-dev
2017-03-21 01:40:05 Found vulnerable package: libfreetype6
2017-03-21 01:40:05 Found vulnerable package: locales
2017-03-21 01:40:05 Found vulnerable package: multiarch-support
2017-03-21 01:40:05 Warning: Found one or more vulnerable packages. [test:PKGS-7392] [details:-] [solution:-]
2017-03-21 01:40:05 Suggestion: Update your system with apt-get update, apt-get upgrade, apt-get dist-upgrade and/or unattended- upgrades [test:PKGS-7392] [details:-] [solution:-]
2017-03-21 01:40:05 ===----------------------------------------------------------------===
```

Solusi untuk ini adalah mengupdate database paket dan mengupdate sistem.

Setelah memperbaiki item yang menyebabkan peringatan, anda harus menjalankan audit lagi. Audit selanjutnya harus bebas dari peringatan yang sama, walaupun peringatan baru bisa muncul. Dalam hal ini, ulangi proses yang ditunjukkan pada langkah ini dan perbaiki peringatannya.

Sekarang setelah anda tahu cara membaca dan memperbaiki peringatan yang dihasilkan oleh Lynis, mari kita lihat bagaimana menerapkan saran yang ditawarkan Lynis.

Step 4 — Implementasi Saran Audit Lynis

Setelah bagian peringatan, anda akan melihat serangkaian saran yang, jika diterapkan, dapat membuat server anda kurang rentan terhadap serangan dan malware. Pada langkah ini, anda akan belajar bagaimana menerapkan beberapa saran yang dihasilkan oleh Lynis setelah melakukan audit terhadap server Ubuntu 16.04. Proses untuk melakukan ini identik dengan langkah-langkah di bagian sebelumnya.

Saran spesifik dimulai dengan saran itu sendiri, diikuti oleh test-ID. Kemudian, tergantung pada pengujian, baris berikutnya akan memberi tahu anda persis perubahan apa yang harus dilakukan pada file konfigurasi layanan yang terpengaruh. Baris terakhir adalah URL kontrol keamanan di mana Anda dapat menemukan lebih banyak informasi tentang subjek.

Di sini, misalnya, adalah bagian saran parsial dari audit Lynis, yang menunjukkan saran yang berkaitan dengan layanan SSH:

Output

```
Suggestions (36):
 ----------------------------
 * Consider hardening SSH configuration [SSH-7408]
   - Details  : ClientAliveCountMax (3 --> 2)
     https://cisofy.com/controls/SSH-7408/

 * Consider hardening SSH configuration [SSH-7408]
   - Details  : PermitRootLogin (YES --> NO)
     https://cisofy.com/controls/SSH-7408/

 * Consider hardening SSH configuration [SSH-7408]
   - Details  : Port (22 --> )
     https://cisofy.com/controls/SSH-7408/

 * Consider hardening SSH configuration [SSH-7408]
   - Details  : TCPKeepAlive (YES --> NO)
     https://cisofy.com/controls/SSH-7408/

 * Consider hardening SSH configuration [SSH-7408]
   - Details  : UsePrivilegeSeparation (YES --> SANDBOX)
     https://cisofy.com/controls/SSH-7408/
...
```

Bergantung pada lingkungan anda, semua saran ini aman untuk diterapkan. Untuk membuat itu, bagaimanapun, anda harus tahu apa arti tiap petunjuk. Karena ini berhubungan dengan server SSH, semua perubahan harus dilakukan pada file konfigurasi daemon SSH, / etc / ssh / sshd_config. Jika Anda ragu tentang saran tentang SSH yang diberikan oleh Lynis, cari arahan dengan sshd_config. Informasi itu juga tersedia secara online. One of the suggestions calls for changing the default SSH port from 22. If you make that change, and you have the firewall configured, be sure to insert a rule for SSH access through that new port.

Seperti bagian peringatan, anda bisa mendapatkan informasi lebih rinci tentang sebuah saran dengan menanyakan Lynis untuk test id menggunakan sudo lynis menunjukkan rincian test-id. Saran lain mengharuskan anda untuk menginstal perangkat lunak tambahan di server anda. Sebagai contoh, misalnya:

Output

```
* Harden the system by installing at least one malware scanner, to perform periodic file
system scans [HRDN-7230]
   - Solution : Install a tool like rkhunter, chkrootkit, OSSEC
     https://cisofy.com/controls/HRDN-7230/
```

Sarannya adalah untuk menginstal rkhunter, chkrootkit, atau OSSEC untuk memenuhi tes pengerasan (HRDN-7230). OSSEC adalah sistem deteksi intrusi berbasis host yang dapat menghasilkan dan mengirim peringatan. Ini adalah aplikasi keamanan yang sangat bagus yang akan membantu beberapa tes yang dilakukan oleh Lynis. Anda dapat mempelajari lebih lanjut tentang alat ini dalam tutorial DigitalOcean ini. Namun, pemasangan OSSEC saja tidak menyebabkan tes

khusus ini berlalu. Instalasi chkrootkit akhirnya berhasil lolos. Ini adalah kasus lain di mana Anda kadang-kadang harus melakukan penelitian tambahan melebihi apa yang Lynis sarankan.

Mari kita lihat contoh lain. Berikut saran yang ditampilkan sebagai hasil uji integritas file.

Output

```
    * Install a file integrity tool to monitor changes to critical and sensitive files [FINT-4350]
      https://cisofy.com/controls/FINT-4350/
```

Saran yang diberikan dalam URL kontrol keamanan tidak menyebutkan program OSSEC yang disebutkan dalam saran sebelumnya, namun menginstalnya sudah cukup untuk lulus uji pada audit berikutnya. Itu karena OSSEC adalah alat pemantauan integritas file yang cukup bagus.

Anda dapat mengabaikan beberapa saran yang tidak berlaku untuk anda. Inilah contohnya:

Output

```
    * To decrease the impact of a full /home file system, place /home on a separated partition [FILE-6310]
      https://cisofy.com/controls/FILE-6310/

     * To decrease the impact of a full /tmp file system, place /tmp on a separated partition [FILE-6310]
      https://cisofy.com/controls/FILE-6310/
```

Secara historis, sistem file inti Linux seperti / home, / tmp, / var, dan / usr dipasang pada partisi terpisah untuk meminimalkan dampak pada keseluruhan server saat mereka kehabisan ruang disk. Ini bukan sesuatu yang sering anda lihat, terutama di server cloud. Sistem file ini sekarang hanya dipasang sebagai direktori pada partisi root yang sama. Tetapi jika Anda melakukan audit Lynis pada sistem semacam itu, Anda akan mendapatkan beberapa saran seperti yang ditunjukkan pada keluaran sebelumnya. Kecuali Anda berada dalam posisi untuk menerapkan saran tersebut, Anda mungkin ingin mengabaikannya dan mengonfigurasi Lynis sehingga tes yang menyebabkannya dihasilkan tidak dilakukan pada audit masa depan.

Melakukan audit keamanan menggunakan Lynis melibatkan lebih dari sekadar memperbaiki peringatan dan menerapkan saran; Ini juga melibatkan identifikasi tes yang berlebihan. Pada langkah selanjutnya, anda akan mempelajari cara menyesuaikan profil default untuk mengabaikan tes semacam itu.

Step 5 – Customisasi Audit Security Lynis

Pada bagian ini, anda akan belajar mengkustomisasi Lynis sehingga hanya menjalankan tes yang diperlukan untuk server anda. Profil, yang mengatur bagaimana pelaksanaan audit, didefinisikan dalam file dengan ekstensi .prf di direktori /etc/lynis. Profil defaultnya dinamai default.prf. Anda tidak mengedit profil default itu secara langsung. Sebagai gantinya, anda menambahkan perubahan apa pun yang Anda inginkan ke file custom.prf di direktori yang sama dengan definisi profil.

Buat file baru bernama /etc/lynis/custom.prf menggunakan editor teks anda:

```
sudo nano /etc/lynis/custom.prf
```

Mari kita gunakan file ini untuk memberi tahu Lynis untuk melewatkan beberapa tes. Berikut adalah tes yang ingin kami lewati:

FILE-6310: Used to check for separation of partitions.
HTTP-6622: Used to test for Nginx web server installation.
HTTP-6702: Used to check for Apache web server installation. This test and the Nginx test above are performed by default. So if you have Nginx installed and not Apache, you'll want to skip the Apache test.
PRNT-2307 and PRNT-2308: Used to check for a print server.
TOOL-5002: Use to check for automation tools like Puppet and Salt. If you have no need for such tools on your server, it's OK to skip this test.
SSH-7408:tcpkeepalive: Several Lynis tests can be grouped under a single test ID.

Jika ada tes di dalam id tes yang ingin anda lewati, inilah cara menentukannya. Untuk mengabaikan sebuah tes, Anda melewatkan petunjuk tes-skip tes ID yang ingin Anda abaikan, satu per baris. Tambahkan kode berikut ke file Anda:

/etc/lynis/custom.prf

```
# Lines starting with "#" are comments
# Skip a test (one per line)

# This will ignore separation of partitions test
skip-test=FILE-6310

# Is Nginx installed?
skip-test=HTTP-6622

# Is Apache installed?
skip-test=HTTP-6702

# Skip checking print-related services
skip-test=PRNT-2307
skip-test=PRNT-2308

# If a test id includes more than one test use this form to ignore a particular test
skip-test=SSH-7408:tcpkeepalive
```

Save & close file.

Lain kali anda melakukan audit, Lynis akan melewatkan tes yang sesuai dengan ID tes yang anda konfigurasikan di profil khusus. Pengujian akan diabaikan dari bagian hasil audit, dan juga bagian sarannya.

File /etc/lynis/custom.prf juga memungkinkan Anda mengubah pengaturan dalam profil. Untuk melakukannya, salin setelan dari /etc/lynis/default.prf ke /etc/lynis/custom.prf dan modifikasi di sana. Anda jarang perlu mengubah pengaturan ini, jadi fokuskan usaha anda untuk menemukan tes yang bisa anda lewatkan.

Selanjutnya, mari kita lihat apa yang Lynis sebut sebagai indeks hardening.

Step 6 – Menterjemahkan Hardening Index

Di bagian bawah setiap hasil audit Lynis, tepat di bawah bagian saran, anda akan menemukan bagian yang terlihat seperti berikut:

Output

```
Lynis security scan details:

 Hardening index : 64 [############        ]
 Tests performed : 206
 Plugins enabled : 0
```

Output ini memberi tahu anda berapa banyak tes yang dilakukan, bersama dengan indeks pengerasan, angka yang diberikan Lynis untuk memberi anda rasa seberapa aman server anda. Nomor ini unik untuk Lynis. Indeks hardening akan berubah sehubungan dengan peringatan yang anda perbaiki dan saran yang anda terapkan. Output ini, yang menunjukkan bahwa sistem yang memiliki indeks hardening 64 adalah dari audit Lynis pertama di server 16.04 Ubuntu baru.

Setelah memperbaiki peringatan dan menerapkan sebagian besar saran, audit baru memberikan hasil sebagai berikut. Anda dapat melihat bahwa indeks hardening sedikit lebih tinggi:

Output

```
Lynis security scan details:

 Hardening index : 86 [#################   ]
 Tests performed : 205
 Plugins enabled : 0
```

Indeks hardening bukanlah penilaian yang akurat tentang seberapa aman sebuah server, namun hanya merupakan ukuran seberapa baik server dikonfigurasi dengan aman (atau dikeraskan) berdasarkan tes yang dilakukan oleh Lynis. Dan seperti yang telah anda lihat, semakin tinggi indeksnya, semakin baik. Tujuan audit keamanan Lynis tidak hanya untuk mendapatkan indeks pengerasan yang tinggi, namun juga untuk memperbaiki peringatan dan saran yang dihasilkannya.

Kesimpulan

Dalam tutorial ini, anda menginstal Lynis, menggunakannya untuk melakukan audit keamanan pada server Ubuntu 16.04, menyelidiki bagaimana memperbaiki peringatan dan saran yang dihasilkannya, dan bagaimana menyesuaikan tes yang dilakukan Lynis.

Butuh sedikit waktu dan usaha ekstra, tapi ada baiknya investasi membuat mesin anda lebih aman, dan Lynis membuat proses itu lebih mudah.

Untuk informasi lebih lanjut tentang Lynis, lihatlah Persiapan dengan Lynis dalam dokumentasi resmi. Lynis adalah proyek open-source, jadi jika anda tertarik untuk berkontribusi, kunjungi halaman Lynis di GitHub.