MODUL PRAKTIKUM

KEAMANAN KOMPUTER DAN JARINGAN

Dr. Yuni Arkhiansyah.,S.Kom,M.Kom

2026

DAFTAR ISI

1

FOOTPRINTING (PENGUMPULAN INFORMASI)

1.1. ALOKASI WAKTU DAN PERSIAPAN

Praktikum ini terdiri dari 8 percobaan, untuk menyelesaikan semua percobaan pada modul ini membutuhkan waktu 200 menit atau dua kali pertemuan. Pada pertemuan pertama dapat dimulai dari percobaan 1, 2, 3, dan 4, kemudian pertemuan berikutnya dilanjutkan percobaan 5, 6, 7, dan 8.

Untuk kelancaran praktikum, asisten praktikum sebaiknya sudah menyiapkan software-software yang dibutuhkan dalam praktikum ini dan didistribusikan kepada peserta praktikum sebelum praktikum dimualai. Asisten juga harus memantau ketersediaan akses internet pada semua komputer yang digunakan dalam praktikum karena semua percobaan membutuhkan akses internet.

1.2. DASAR TEORI

Footprinting merupakan suatu kegiatan dalam mengumpulkan informasi tentang target yang diinginkan. Misalnya, ketika perampok memutuskan untuk merampok sebuah bank, mereka tidak langsung masuk ke dalam bank dan mulai meminta uang secara paksa. Namun, mereka berusaha keras dalam mengumpulkan berbagai macam informasi yang berkaitan dengan bank tersebut, baik itu rute mobil berlapis baja milik bank, waktu pengiriman uang, kamera video pengawas, jumlah teller dan jalan keluar untuk melarikan diri, serta hal-hal lain yang akan membantu dalam kesuksesan perampokan pada bank tersebut.

Persyaratan yang sama berlaku untuk seorang *attacker* yang sukses, mereka harus mengumpulkan banyak informasi terlebih dahulu sebelum melaksanakan serangan pada target yang

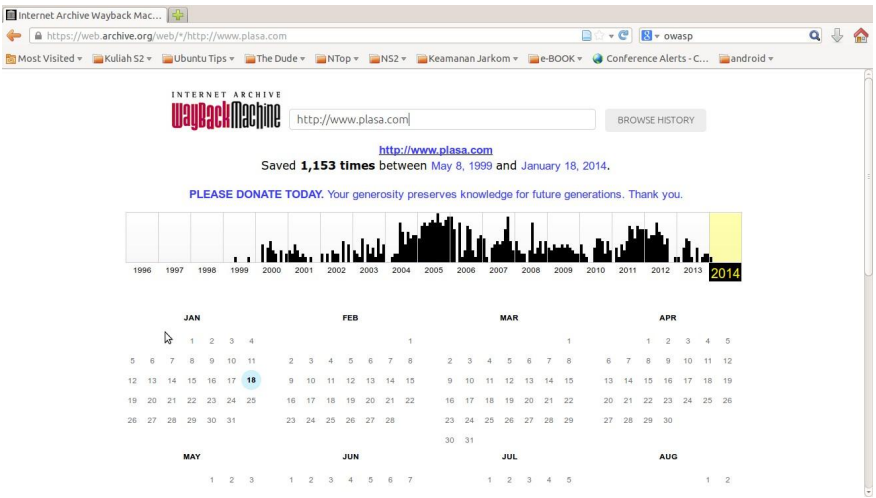
ditentukan. *Attackers* akan mengumpulkan informasi sebanyak mungkin tentang semua aspek keamanan target. *Attackers* pertama-tama harus mengetahui sesuatu mengenai target, hal-hal seperti nama domain, alamat IP, alamat fisik dan lokasi, nomor telepon, dan jenis database yang digunakan.

Alat yang paling sering digunakan dalam pencarian informasi tentang target adalah web browser dan koneksi Internet. Hal pertama yang biasa digunakan dalam mencari informasi awal adalah melalui Website target yang memang ditampilkan untuk publik. Apa yang diberikan pada website target merupakan informasi yang bias didapat secara cuma-cuma. Biasanya informasi ini diberikan kepada klien, pelanggan, atau masyarakat umum. Sebagian besar Website membahas tentang organisasi, dewan eksekutif, dan kepemilikannya. Selanjutnya *attackers* mungkin akan menggunakan informasi yang berkaitan dengan lokasi target untuk melakukan beberapa serangan lanjutan seperti *Dumster Diving*, *Wardriving*, dan *Wardialing*.

1.2.1. Dumster Diving (Electronic)

Dumster Diving adalah proses mencari data elektronik yang telah usang, tidak jelas, atau tua. Salah satu tempat untuk melihat dan menemukan informasi tersebut adalah di *Internet Archive* (www.archive.org). Proyek ini dimulai tahun 1996 dan masih aktif hingga saat ini.

Untuk memulai surfing pada *Wayback Machine*, ketik di alamat web dari situs atau halaman di mana ingin memulai kemudian tekan Enter. Gambar 1.1 menunjukkan sebuah contoh dari *Wayback Machine*. Angka ini menunjukkan screen capture dari situs web plasa pada tahun 1999.



Gambar 1.1. Contoh tampilan *Wayback Machine*

1.2.2. Menganalisa Kode Halaman Web

Informasi tambahan masih bisa didapatkan dari situs web dan hal ini membutuhkan kejelian untuk melalui setiap halaman web dan menganalisis *source code*-nya. Kita dapat memeriksa setiap halaman untuk mencari informasi-informasi seperti:

1. Alamat email
2. Link ke situs lain
3. Catatan atau komentar
4. Informasi yang mengidentifikasi aplikasi web atau program yang digunakan
5. Penghitungan struktur dan desain situs

Salah satu cara untuk memeriksa sebuah situs secara rinci adalah dengan menggunakan penangkap situs (*site rippers*). Meskipun kita secara manual bisa menjelajah situs tersebut, akan tetapi alat penangkap situs dapat mempercepat proses. *Site rippers* merupakan cara yang baik untuk membuat duplikat dari situs web yang dapat disimpan pada hard drive lokal. Dengan cara ini kita bisa memeriksa HTML kode dan mencari fragmen informasi lainnya.

Tool-tool yang dapat kita gunakan untuk menangkap situs antar lain:

1. **BlackWidow** - memungkinkan kita untuk melihat tampilan kode HTML, source code, link, alamat email, dan banyak lagi
2. **Teleport Pro** - Sebuah scanner situs berbasis Windows. Alat memetakan situs yang memungkinkan kita untuk meng-copy situs web dan review mereka secara lokal
3. **Wget** - Sebuah tool command-line berbasis Windows dan Unix yang dapat men-download isi sebuah situs web.
4. **Instant Source** - Bekerja dengan Internet Explorer dan akan menampilkan *source code* untuk bagian yang dipilih dari suatu halaman web. Alat ini juga akan menampilkan gambar, film Flash, dan file script pada halaman web.

1.2.3. Mencari Informasi Kepemilikan Domain

Internet berawal pada tahun 1969, pada saat itu hanya terdapat kumpulan jaringan-jaringan kecil dan sampai sekarang telah berkembang menjadi Internet yang kita kenal. Komunitas Internet (internet society) mengatur Internet. Kelompok nirlaba ing yang didirikan pada tahun 1992 yang bertujuan untuk mengendalikan kebijakan dan prosedur yang mendefinisikan bagaimana fungsi Internet. Salah satu kontrol otoritas ini adalah Internet Assigned Numbers Authority (IANA). IANA bertanggung jawab untuk melestarikan fungsi koordinasi pusat dari global Internet untuk kepentingan publik. IANA juga secara global mengelola nama domain dan alamat.

IANA (http://www.iana.org) adalah salah satu tempat yang dapat berfungsi sebagai titik awal untuk mengetahui lebih lanjut informasi tentang kepemilikan domain. Untuk mengetahui informasi lebih lanjut tentang kepemilikan domain, mulai dengan generic top-level domain link. Ini adalah tempat dimana kita dapat menemukan informasi lebih WHOIS. Database WHOIS adalah sebuah tool yang memungkinkan kita untuk menanyakan informasi sebuah organisasi ketika mereka mendaftarkan domain mereka. WHOIS biasanya dapat akan ditanyakan berdasarkan nama domain atau dengan IP. Semua informasi yang ditemukan di situs IANA dicari berdasarkan alamat domain. Saat meninjau WHOIS database, kita harus mencari informasi yang terkandung. Internet Corporation For Assigned Names (ICANN) mengharuskan semua pemegang

domain untuk mengirimkan informasi WHOIS. Informasi tersedia meliputi, pendaftar admin, penagihan, dan informasi kontak teknis. Orang non-security-minded mungkin akan menempatkan terlalu banyak informasi dalam catatan WHOIS, informasi berguna yang dapat digunakan oleh penyerang potensial.

IANA menawarkan titik awal yang baik untuk menyelidiki nama domain. Tapi ada tempat lain untuk melihat nama domain yaitu *Regional Internet Registries* (RIR). RIR bertugas mengawasi distribusi regional alamat IP dalam geografis wilayah di dunia. Kelima RIR adalah sebagai berikut :

- American Registry for Internet Numbers (ARIN)–Amerika Utara
- RIPE Network Coordination Centre (RIPE NCC)–Eropa, Timur Tengah, dan Asia Tengah
- Asia-Pacific Network Information Centre (APNIC)–Asia dan daerah Asia Pasifik
- Latin American and Caribbean Internet Address Registry (LACNIC)– Amerika Latin dan daerah Karribian
- African Network Information Centre (AfriNIC)–Afrika

1.3. TUJUAN

1. Mengetahui sejarah sebuah website dengan menggunakan wayback machine dan mencari informasi-informasi yang terkandung didalamnya.
2. Mencari informasi dengan mengcopy sebuah website dan menganalisanya secara local
3. Mengetahui jenis-jenis nama domain
4. Dapat mencari informasi di Database Whois melalui internet
5. Mengetahui penggunaan aplikasi nslookup dan tracert, serta aplikasi bantuan yang lain.

1.4. BAHAN DAN ALAT

1. Satu unit komputer dengan OS Windows XP yang dapat mengakses internet

2. Aplikasi browser (Mozilla FireFox atau Internet Exploler)
3. Software Teleport Pro (dapat di-*download* di http://www.tenmax.com/teleport/pro/download.htm)
4. Aplikasi nslookup dan tracert (biasanya sudah ada secara defaukt dalam sistem operasi)
5. Aplikasi Sam Spade (dapat di-*download* di http://www.softpedia.com/get/Network-Tools/Network-Tools-Suites/Sam-Spade.shtml)

1.5. LANGKAH PERCOBAAN

1.5.1. Percobaan 1: Melihat Sejarah Sebuah Website Dengan Menggunakan Wayback Mechine

1. Buka aplikasi web browser (misal: internet explorer atau mozilla firefox)
2. Masukkan alamat http://www.archive.org pada web browser
3. Masukkan alamat website target (misal: www.plasa.com)

Gambar 1.2. Memasukkan alamat target pada *Wayback Mechine*

4. Tekan "Take Me Back"
5. Akan muncul tampilan seperti gambar 1.1.
6. Kita bisa melihat perkembangan secara detail plasa.com mulai dari awal tahun website tersebut dibangun sampai kapan terakhir kali website diupdate. Kita dapat pula mencari informasi-informasi yang terkandung didalam record website tersebut dalam rangka untuk melakukan serangan.
7. Lihatlah isi website tersebut mulai dari Tahun 2000-2013 (minimal tiap tahun 2 website) dan perhatikan informasi yang terkandung didalamnya.

Tugas Percobaan 1 (dikerjakan pada saat praktikum) :

1. Ulangi seperti langkah 1-7 dan gantilah alamat website target dengan alamat website yang lain sebagai target (misal : www.google.com, www.yahoo.com, www.gmail.com dst) minimal 5 website yang berbeda. Tiap mahasiswa harus berbeda satu sama lain.
2. Print screen (seperti gambar 1.1) hasil pencarian tiap-tiap website tersebut dan jadikan sebagai bahan laporan

1.5.2. Percobaan 2: Menduplikasi sebuah website dengan website duplicator (Teleport Pro)

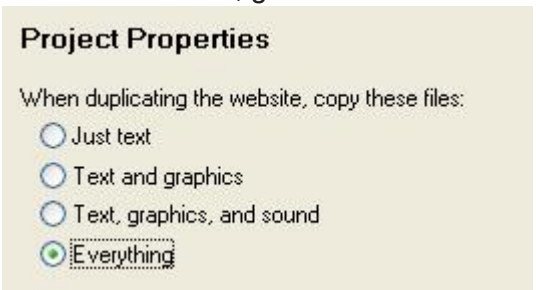
1. Download program "Teleport Pro" di internet. Anda dapat mencarinya melalui search engine seperti google, yahoo, altavista dst.
2. Install dan Jalankan program "Teleport Pro"
3. Pilih "Duplicate a website, including directory structure" kemudian "Next"

Gambar 1.3. Membuat duplikat *website*

4. Masukkan alamat website yang kita ingin *copy* (missal : www.plasa.com) kemudian "Next"

Gambar 1.4. Memasukkan alamat website

5. Klik "Everything" untuk meng-copy semua jenis file kemudian "Next". Anda bisa memilih jenis file yang ingin dicopy seperti "Just Text" hanya untuk text saja, "Text and Graphics" hanya untuk text dan gambar, "Text, Graphics, and sound" untuk text, gambar dan suara.



Gambar 1.5. Lima jenis file yang akan di-*copy*

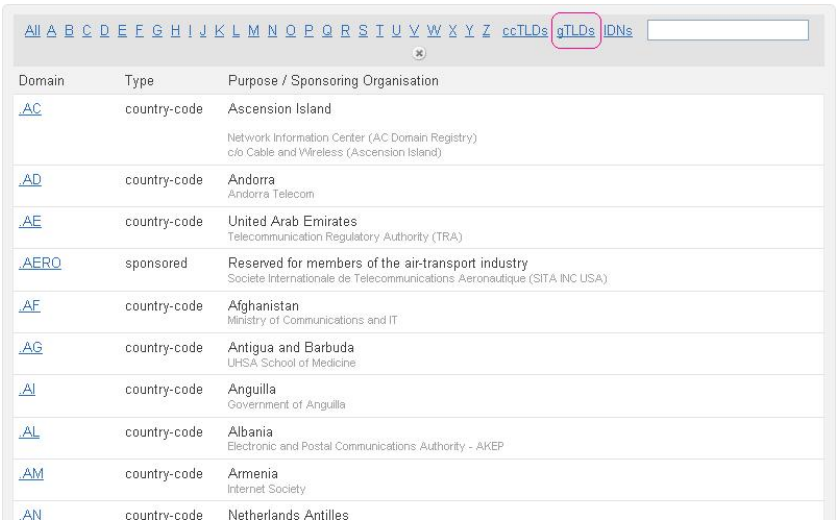
6. Klik "Finish" dan "Save". Kemudian pada menu bar pilih "Project" dan "Start" untuk memulai peng-copy-an website. Tunggu beberapa saat sampai website ter-*copy*.

Tugas Percobaan 2 (dikerjakan pada saat praktikum) :

1. Ulangi seperti langkah 1-6 dan gantilah alamat website target dengan alamat website yang lain sebagai target (misal : www.google.com, www.yahoo.com, www.gmail.com dst) minimal 3 website yang berbeda. Tiap mahasiswa harus berbeda satu sama lain.
2. Print screen (seperti gambar 1-1) hasil pencarian tiap-tiap website tersebut dan jadikan sebagai bahan laporan

1.5.3. Percobaan 3 : Mempelajari Jenis Nama Domain di IANA

1. Buka program web browser dan masukkan alamat http://www.iana.org/domains/root/db/



All A B C D E F G H I J K L M N O P Q R S T U V W X Y Z ccTLDs gTLDs IDNs

Domain	Type	Purpose / Sponsoring Organisation
.AC	country-code	Ascension Island Network Information Center (AC Domain Registry) c/o Cable and Wireless (Ascension Island)
.AD	country-code	Andorra Andorra Telecom
.AE	country-code	United Arab Emirates Telecommunication Regulatory Authority (TRA)
.AERO	sponsored	Reserved for members of the air-transport industry Societe Internationale de Telecommunications Aeronautique (SITA INC USA)
.AF	country-code	Afghanistan Ministry of Communications and IT
.AG	country-code	Antigua and Barbuda UHSA School of Medicine
.AI	country-code	Anguilla Government of Anguilla
.AL	country-code	Albania Electronic and Postal Communications Authority - AKEP
.AM	country-code	Armenia Internet Society
.AN	country-code	Netherlands Antilles

Gambar 1.6. Data nama-nama domain negara

2. Isilah table berikut ini berdasarkan data yang anda dapatkan diatas :

Domain	Type	Purpose/Sponsoring Organisation
.ID		
.UK		
.UY		
.SA		
.LK		
.DK		

1.5.4. Percobaan 4 : WHOIS melalui website

1. Carilah di google website yang menawarkan fasilitas WHOIS. Misalkan www.**whois**.net, www.**who.is**, www.geektools.com/**whois**.php, *dst.*
2. Pilihlah minimal 4 website yang menarwarkan fasilitas WHOIS berdasarkan hasil langkah 1

3. Carilah informasi tentang 3 nama domain target di WHOIS

Website WHOIS	Alamat Target	Hasil WHOIS
http://www	1.	
	2.	
	3.	
http://www	1.	
	2.	
	3.	
http://www	1.	
	2.	
	3.	
http://www	1.	
	2.	
	3.	

1.5.5. Percobaan 5 : DNS dengan nslookup

1. Buka “command Promt” melalui “Start” – “All Programs” – “Accessories” – “Command Prompt”
2. Ketiklah “C:\>nslookup www.hackthestack.com”

```
Server:  dnsr1.sbcglobal.net
Address:  123.91.121.1

Non-authoritative answer:
Name:    www.hackthestack.com
Address:  202.131.95.30
```

3. Isilah tabel berikut ini

Perintah nslookup	Hasil perintah
"C:\>nslookup www.gmail.com"	
"C:\>nslookup www.yahoo.com"	
"C:\>nslookup www.kaskus.us "	

1.5.6. Percobaan 6: DNS dengan menggunakan Tool Sam Spade

1. Program "Sam Spade 1.14" dapat di-*download* dari internet
2. Install program tersebut dan jalankan
3. Masukkan alamat website http://www.plasa.com dan pilih "Magic" kemudian enter

Gambar 1.6. Tampilan Sam Spade 1.14

```
Registrant:
  PT. Telekomunikasi Indonesia. Tbk idc@telkom.co.id +62.213860500
  PT. Telekomunikasi Indonesia. Tbk
  Jl. Kebon Sirih 37
  Jakarta,DKI Jakarta,ID 10340


Domain Name:plasa.com
Record last updated at 2010-05-10 05:26:48
Record created on 1998/6/26
Record expired on 2015/6/26


Domain servers in listed order:
  ns2.plasa.com    dns1.plasa.com
Administrator:
  Jl. Kebon Sirih 37
  Jakarta
DKI Jakarta,
ID
 10340

  name:(PT. Telekomunikasi Indonesia. Tbk)
mail:(idc@telkom.co.id) +62.213860500
  PT. Telekomunikasi Indonesia. Tbk
Technical Contactor:
  Jl. Mendawai 1 no 45
  Jakarta
DKI Jakarta,
ID
 12130

  name:(PT.Metranet)
mail:(it@mojopia.com) +62.2172795682
  PT.Metranet
Billing Contactor:
  Jl. Kebon Sirih 37
  Jakarta
DKI Jakarta,
ID
 10340

  name:(PT. Telekomunikasi Indonesia. Tbk)
mail:(idc@telkom.co.id) +62.213860500
  PT. Telekomunikasi Indonesia. Tbk

Registration Service Provider:
 name: Divisi Multimedia, PT Telekomunikasi Indonesia, tbk
 tel: +62.213860500
  fax: +62.213861226
  web:www.telkomhosting.com
```

Tugas Percobaan 6 (dikerjakan pada saat praktikum):

1. Masukkan 3 website target yang berbeda dengan menggunakan Tool Sam Spade 1.14 (caranya seperti langkah diatas)

2. Print screen hasil dari 3 website tersebut dan jadikan sebagai laporan

1.5.7. Percobaan 7: Mengidentifikasi Software Web Server menggunakan Sam Spade 1.14

1. Jalan program Sam Spade 1.14
2. Masukkan alamat http://plasa.com
3. Pada menu bar pilih "Tools" - "Browse web" kemudian "OK"

Gambar 1.8. Identifikasi web server

4. Perhatikan hasinya dan bisa kita pastikan bahwa plasa. com menggunakan Web Server Microsoft IIS/7.5

Tugas Percobaan 7 (dikerjakan pada saat praktikum) :

1. Identifikasi jenis software web server minimal 5 website yang berbeda
2. Tampilkan hasilnya sebagai bahan laporan

1.5.8. Percobaan 8: Mencari Lokasi Web Server dengan Tracert

1. Buka "command Promt" melalui "Start" - "All Programs" - "Accessories" - "Command Prompt"
2. Ketik "C:\>tracert www.wiley.com

```
Tracing route to www.wiley.com [64.143.198.41] over a maximum of 30 hops:

  1    <10 ms    <10 ms     10 ms  PROXY [172.20.1.1]
  2    <10 ms    <10 ms     66-162-219-65.gen.twtelecom.net [66.162.219.65]
  3     10 ms    <10 ms      209.163.157.165
  4    <10 ms     10 ms      core-dlfw.twtelecom.net [66.192.246.77]
  5     10 ms     10 ms       tran-dlfw.twtelecom.net [168.215.54.74]
  6     10 ms     10 ms      sl-gw40-fw-4-2.sprintlink.net [160.81.227.105]
  7     10 ms     10 ms       sl-bb22-fw-4-3.sprintlink.net [144.232.8.249]
  8     20 ms     10 ms       144.232.19.214
  9     10 ms     10 ms       dal-core-01.inet.qwest.net [205.171.25.45]
 10     20 ms     10 ms      iah-core-02.inet.qwest.net [205.171.8.126]
 11     10 ms     10 ms      iah-core-01.inet.qwest.net [205.171.31.1]
 12     40 ms     40 ms      tpa-core-02.inet.qwest.net [205.171.5.105]
 13     30 ms     30 ms      cntr-02.tpf.qwest.net [205.171.27.78]
 14     30 ms     30 ms      ms  msfc-02.tpf.qwest.net [63.146.176.26]
 15     30 ms     40 ms      ms  www.wiley.com [63.146.189.41]
Trace complete.
```

Tugas Percobaan 8 (dikerjakan pada saat praktikum) :

1. Carilah salah satu program traceroute yang berbasis GUI di internet
2. Download kemdian install
3. Trace website berikut dengan menggunakan program yang sudah diinstall :
   - www.google.com
   - www.plasa.com
   - www.ums.ac.id
   - www.gmail.com
4. Tampilkan hasilnya sebagai bahan laporan

2
SCANNING

2.1. ALOKASI WAKTU DAN PERSIAPAN

Praktikum ini terdiri dari 4 percobaan, untuk menyelesaikan semua percobaan pada modul ini membutuhkan waktu 200 menit atau dua kali pertemuan. Pada pertemuan pertama dapat dimulai dari percobaan 1 dan 2, kemudian pertemuan berikutnya dilanjutkan percobaan 3 dan 4.

Untuk kelancaran praktikum, asisten praktikum sebaiknya sudah menyiapkan software-software yang dibutuhkan dalam praktikum ini dan didistribusikan kepada peserta praktikum sebelum praktikum dimualai. Asisten juga harus memantau kondisi jaringan pada semua komputer dan menyediakan sebuah komputer sebagai target *scanning*.

2.2. DASAR TEORI

Jika footprinting diumpakan seperti lokasi atau rumah tempat informasi, maka *Scanning* seperti mengetuk dinding untuk menemukan semua pintu dan jendela. Selama footprinting, kita memperoleh daftar blok jaringan IP dan alamat IP melalui whois dan ARIN. Teknik ini memberikan administrator keamanan informasi berharga, termasuk nama karyawan dan nomor telepon, kisaran alamat IP, server DNS, dan server mail.

Modul ini akan membahas Tool-tool, teknik, dan metode yang digunakan untuk system komputer yang hidup. *Port scanning* adalah salah satu yang paling banyak digunakan untuk sistem identifikasi. Sebelum sistem dapat diserang, maka harus diidentifikasi terlebih dahulu. Sebagai contoh, penyerang mungkin memiliki eksploitasi yang dapat bekerja terhadap server Microsoft IIS. Bila hal tersebut diterapkan pada server Apache maka tidak akan berguna. Jadi,

attackers harus terlebih dahulu mengidentifikasi bahwa komputer target sebenarnya berjalan pada Server Microsoft IIS. Untuk membuat analisis kita tepat, kita harus mengasumsikan bahwa mengeksploitasi hanya dapat bekerja melawan IIS v5. Jika hal ini terjadi, mengetahui bahwa sistem menjalankan perangkat lunak Microsoft mungkin masih belum cukup. *Attackers* perlu tahu bahwa layanan ini khusus IIS v5. Di sinilah pentingnya *port scanning* dilakukan. *Port Scanning* tidak hanya mengidentifikasi port tetapi, tergantung pada tool tersebut digunakan, juga memberikan informasi tentang kemungkinan layanan berjalan pada port yang terbuka.

Anda juga harus memahami bagaimana tool *port scanning* bekerja. Bila kasus port scanning tidak bekerja, Anda juga harus tahu tool-tool dan teknik lain yang digunakan untuk menganalisis perangkat jaringan dan menentukan layanan apa saja yang terbuka.

2.2.1. ICMP (Ping)

ICMP adalah singkatan dari *Internet Control Message Protocol*. Salah satu langkah yang paling dasar dalam pemetaan jaringan adalah melakukan ping sweep otomatis pada range alamat IP dan blok jaringan untuk menentukan apakah ada sistem yang hidup. Ping secara tradisional digunakan untuk mengirim paket ICMP ECHO (Tipe 8) ke sistem target dalam upaya untuk memperoleh suatu ICMP ECHO_REPLY (Tipe 0) yang menunjukkan sistem target hidup. Meskipun ping dapat digunakan untuk menentukan jumlah sistem yang hidup dalam jaringan kecil-menengah, itu tidak efisien untuk jaringan yang lebih besar, jaringan perusahaan. Memindai IP Kelas A yang lebih besar dapat memakan waktu berjam-jam, jika tidak selesai dalam sehari. Anda harus belajar beberapa cara untuk menemukan sistem hidup.

Ping ditemukan pada hampir setiap sistem yang berjalan pada TCP / IP. Sementara Ping adalah alat konektivitas dasar hal ini berguna untuk mengidentifikasi mesin aktif. Ping bekerja dengan mengirim echo request ke sistem dan menunggu target untuk mengirim echo reply kembali. Contohnya adalah sebagai berikut:

```
C:\>ping 192.168.1.254

Pinging 192.168.1.254 with 32 bytes of data:

Reply from 192.168.1.254: bytes=32 time<10ms TTL=64
Reply from 192.168.1.254: bytes=32 time<10ms TTL=64
Reply from 192.168.1.254: bytes=32 time<10ms TTL=64
Reply from 192.168.1.254: bytes=32 time<10ms TTL=64

Ping statistics for 192.168.1.254:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

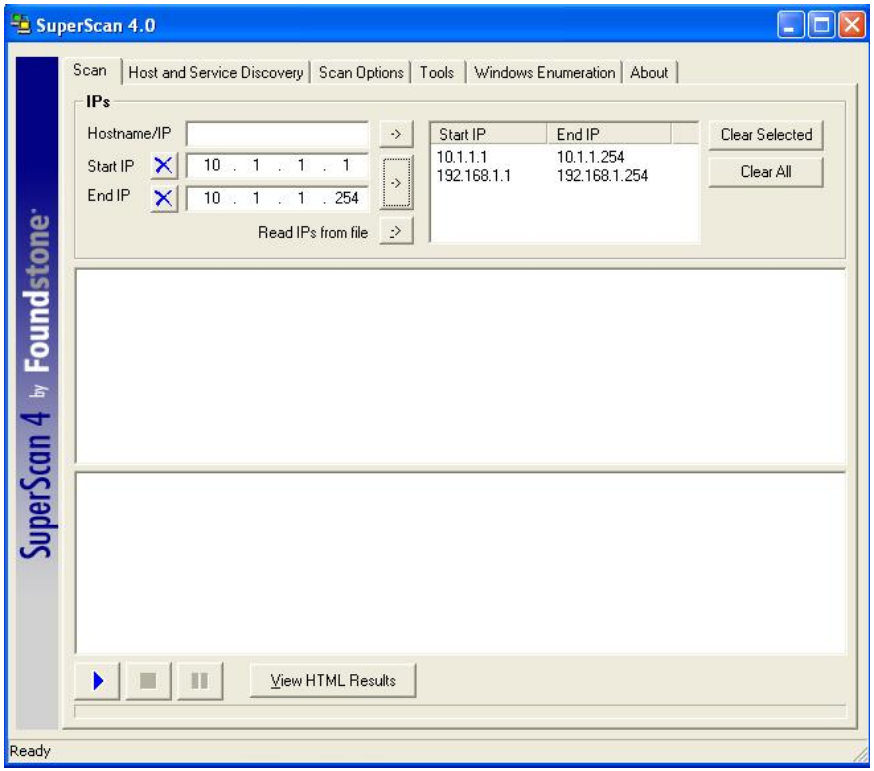
Jika komputer target tidak bisa diakses, batas waktu permintaan dikembalikan. Anda dapat melihat contoh ini di sini di mana saya diping host di 192.168.1.250 :

```
C:\>ping 192.168.1.250

Pinging 192.168.1.250 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.1.250:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Anda mungkin bertanya-tanya apa yang terjadi jika ICMP diblokir oleh situs target. Pertanyaan bagus. Hal ini tidak jarang datang di situs security-conscious yang telah memblokir ICMP di perbatasan router atau firewall. Meskipun ICMP mungkin diblokir, beberapa tool dan teknik yang lain dapat digunakan untuk menentukan apakah sistem benar-benar hidup. Namun, mereka tidak akurat atau seefisien ping sweep normal.

2.2.2. Port Scanning

Ketika lalu lintas ICMP diblokir, port scanning adalah teknik alternatif yang pertama untuk menentukan host hidup. Dengan memindai untuk port-port umum pada setiap alamat IP yang potensial, kita dapat menentukan host hidup jika kita bisa mengidentifikasi port terbuka atau sedang mendengar (*listening*) pada sistem target. Teknik ini bisa memakan waktu yang lama.



Gambar 2.1. Tampilan aplikasi SuperScan 4

Untuk Windows, alat yang kami sarankan adalah SuperScan. SuperScan akan menemukan host dan penemuan layanan dengan menggunakan ICMP dan TCP/ UDP. Dengan menggunakan port TCP UDP scan pilihan, Anda dapat menentukan apakah suatu host hidup atau tidak-tanpa menggunakan ICMP sama sekali. Seperti yang dapat Anda

lihat pada Gambar 2-1, cukup pilih kotak centang untuk setiap protokol yang ingin Anda gunakan dan jenis teknik yang anda inginkan.

Lapisan aplikasi berada di bagian atas protokol TCP / IP stack. Lapisan ini bertanggung jawab untuk mendukung aplikasi. Aplikasi biasanya dipetakan bukan oleh nama akan tetapi dengan port yang berhubungan. Port ditempatkan ke paket TCP dan UDP sehingga aplikasi yang benar dapat dikirimkan ke protokol yang diperlukan di bawah ini.

Meskipun layanan tertentu mungkin memiliki port yang telah ditetapkan, tidak ada spesifikasi yang menentukan bahwa sebuah layanan tidak bisa ditentukan pada port lain. Contoh umum adalah SMTP (Simple Mail Transfer Protocol). Port yang ditentukan adalah 25. Institusi Anda mungkin memblokir port 25 dalam upaya untuk menjaga Anda dari menjalankan mail server pada komputer lokal, tetapi bisa untuk mencegah Anda untuk menjalankan server mail Anda di port lokal yang lain. Alasan utama layanan telah ditetapkan port agar klien dapat dengan mudah menemukan layanan yang ada pada remote host. Sebagai contoh, server FTP ditentukan pada port 21 dan HTTP (Hypertext Transfer Protocol) server ditentukan pada port 80. Aplikasi klien seperti program FTP (File Transfer Protocol) atau browser menggunakan port secara acak, biasanya lebih besar atau diatas dari 1023. Terdapat 65.535 port TCP dan UDP. Port ini dibagi menjadi tiga kategori, yang meliputi port umum (0-1023), port terdaftar (1024-49151), dan port dinamis (49152-65535). Meskipun ada ratusan port dan aplikasi yang terkait dalam praktek, hanya beberapa ratus saja yang sedang digunakan secara umum.

PORT	SERVICE	PROTOCOL
20/21	FTP	TCP
22	SSH	TCP
23	Telnet	TCP
25	SMTP	TCP
53	DNS	TCP/UDP
67/68	DHCP	UDP
69	TFTP	UDP
80	HTTP	TCP
88	Kerberos	UDP
110	POP3	TCP
111	SUNRPC	TCP/UDP
135	RPC	TCP/UDP
139	NetBIOS Session	TCP/UDP
161/162	SNMP	UDP
389	LDAP	TCP
443	SSL	TCP
445	SMB over IP	TCP/UDP
1433	MS-SQL	TCP

Gambar 2.2. Port-Port Umum

2.2.3. TCP and UDP Port Scanning

Ingat bahwa TCP menawarkan komunikasi handal dan dianggap sebagai protokol koneksi. TCP membangun sambungan dengan menggunakan apa yang disebut sebagai jabat tangan tiga arah (*three-way handshake*).

Header TCP berisi sebuah field 1-byte untuk flag-flag. Flag-flag ini termasuk berikut:

- **ACK** - penerima akan mengirimkan ACK untuk mengakui data.
- **SYN** - Digunakan selama sesi setup tiga langkah untuk memberitahu pihak yang lainnya untuk memulai komunikasi dan digunakan untuk menyepakati urutan awal nomor.
- **FIN** - Digunakan selama shutdown normal untuk menginformasikan kepada host lain bahwa pengirim tidak

memiliki lebih banyak data untuk mengirim.

- **RST**-Digunakan untuk membatalkan sesi abnormal.
- **PSH**-Digunakan untuk memaksa data dikirim tanpa menunggu buffer terisi.
- **URG**-Digunakan untuk menunjukkan prioritas data.

Pada akhir komunikasi, TCP mengakhiri sesi dengan menggunakan apa yang disebut shutdown empat-langkah. TCP dirancang sedemikian rupa untuk menyediakan komunikasi yang kuat. Dari sudut pandang *scanning*, ini berarti TCP mempunyai kemampuan untuk mengembalikan berbagai jenis respon terhadap sebuah program scanning. Dengan memanipulasi fitur ini, *attackers* bisa memanipulasi paket untuk membuat server untuk merespon atau mencoba dan menghindari deteksi sistem deteksi intrusi (IDS). Banyak dari metode ini dibangun di untuk tool-tool port-scanning yang populer.

2.2.4. Tipe-Tipe Scan

- **TCP connect scan -** Jenis scan ini terhubung ke port host target dan menyelesaikan full three-way handshake (SYN, SYN/ ACK, dan ACK), sebagai keadaan TCP RFC (Request for Comments). Hal ini mudah terdeteksi oleh sistem target.
- **TCP SYN scan** - Teknik ini disebut scanning setengah terbuka (*half-open scanning*) karena koneksi penuh TCP tidak dilakukan. Sebaliknya, suatu paket SYN dikirimkan ke port target. Jika SYN/ACK diterima dari port sasaran, kita dapat menyimpulkan bahwa port tersebut dalam keadaan listening. Jika RST/ACK diterima, biasanya menunjukkan bahwa port tersebut tidak listening. Suatu RST/ ACK akan dikirim oleh sistem untuk melakukan port scan sehingga koneksi penuh tidak pernah dibentuk. Teknik ini memiliki keuntungan menjadi stealthier dari TCP penuh terhubung. Namun, salah satu kelemahan dari teknik ini adalah bahwa bentuk scanning ini dapat menghasilkan kondisi denial of service pada target dengan membuka sejumlah besar koneksi setengah terbuka. Tetapi jika Anda memindai sistem yang sama dengan jumlah tinggi koneksi ini, teknik

ini relatif aman.

- **TCP FIN scan** - Teknik ini mengirimkan suatu paket FIN ke port sasaran. Berdasarkan RFC 793 (http://www.ietf.org/rfc/rfc0793.txt), sistem target akan mengirim balik suatu RST untuk semua port yang tertutup. Teknik ini biasanya hanya bekerja pada UNIXbased TCP / IP stack.
- **TCP Xmas Tree scan** - Teknik ini mengirimkan suatu paket FIN, URG, dan PUSH ke port target. Berdasarkan RFC 793, sistem sasaran akan mengirim balik suatu RST untuk semua port yang tertutup.
- **TCP Null scan** - Teknik ini membuat off semua flag. Berdasarkan RFC 793, sistem sasaran akan mengirim balik suatu RST untuk semua port yang tertutup.
- **TCP ACK scan**- Teknik ini digunakan untuk memetakan aturan firewall. Hal ini dapat membantu menentukan apakah firewall adalah filter packet sederhana yang hanya mengizinkan koneksi yang ditentukan (koneksi dengan bit set ACK).
- **TCP Windows scan** - Teknik ini dapat mendeteksi port terbuka dan port yang difilter atau tidak pada beberapa sistem (misalnya, AIX dan FreeBSD) karena anomali dalam cara ukuran windows TCP yang dilaporkan.
- **TCP RPC scan** - Teknik ini khusus untuk sistem UNIX dan digunakan untuk mendeteksi dan mengidentifikasi port Remote Procedure Call (RPC) dan program mereka dan nomor versi terkait
- **UDP scan** - Teknik ini mengirimkan suatu paket UDP ke port sasaran. Jika port target merespon dengan sebuah pesan “ICMP port unreachable”, port ditutup. Sebaliknya, jika Anda tidak menerima pesan “ICMP port unreachable”, Anda dapat menyimpulkan bahwa port terbuka. Karena UDP dikenal sebagai connectionless protocol, akurasi teknik ini sangat tergantung pada banyak faktor yang terkait dengan pemanfaatan dan penyaringan dari jaringan target. Di samping itu, UDP scanning merupakan proses yang sangat lambat jika Anda mencoba untuk memindai perangkat yang menggunakan paket filtering berat. Jika anda berencana untuk melakukan scan UDP melalui Internet, bersiaplah untuk hasil yang tidak dapat

diandalkan.

2.3. TUJUAN

1. Mengetahui cara mendeteksi dan mencari informasi mengenai kondisi komputer yang aktif dalam sebuah jaringan
2. Mengetahui penggunaan software *scanner* seperti Anggry IP Scanner, Nmap dan Netcat sebagai alat bantu untuk mendapatkan informasi tentang sebuah komputer
3. Mengetahui cara mendapatkan informasi mengenai Port yang terbuka dan sistem operasi yang digunakan

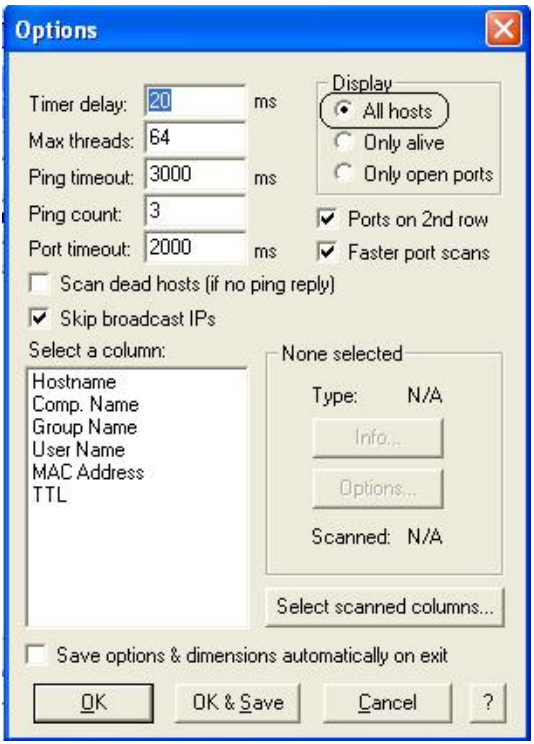
2.4. BAHAN DAN ALAT

1. Satu unit komputer dengan OS Windows XP
2. Software Nmap (dapat di-*download* di http://nmap.org/download.html)
3. Angry IP Scanner (dapat di-*download* di http://angryip.org/w/Download)
4. NetCat (dapat di-*download* di http://netcat.sourceforge.net/download.php)

2.5. LANGKAH PERCOBAAN

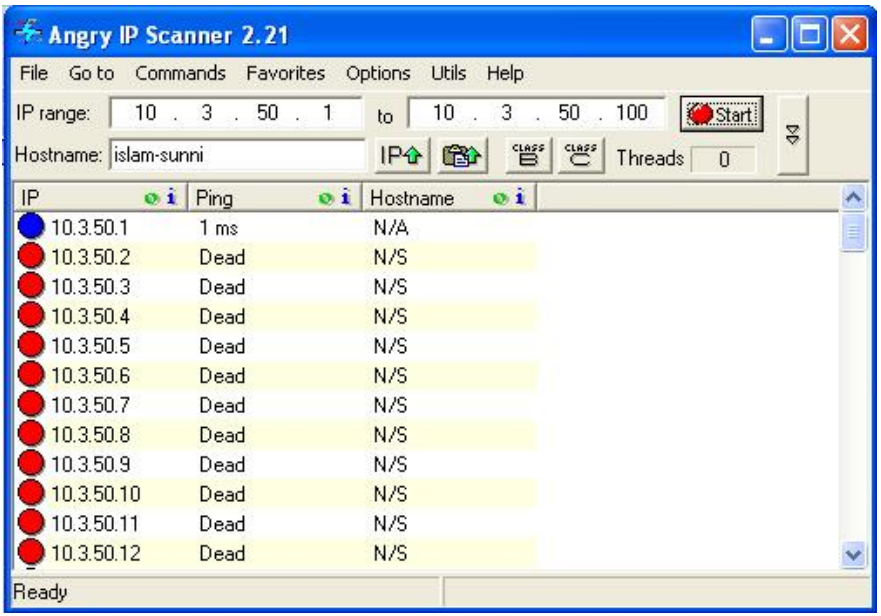
2.5.1. Percobaan 1 : Mencari Komputer yang hidup/ aktif dengan Program Angry IP Scanner

- Install program Angry IP Scanner dan jalankan, jika anda menggunakan versi terbaru mungkin tampilannya akan sedikit berbeda dengan modul ini.
- Pada menu bar pilih “Options” – “Options”
- Pada “Display” pilih “All hosts” untuk menampilkan semua host. Kemudian “OK”

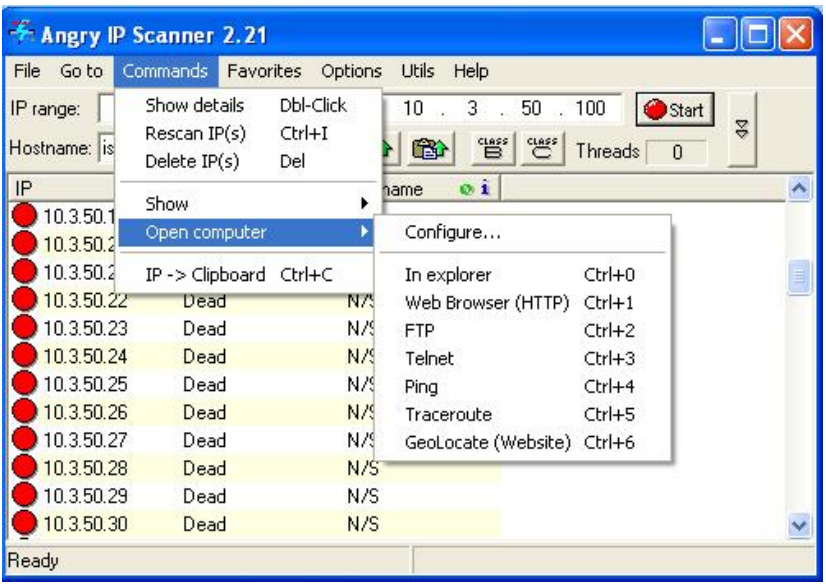


Gambar 2-3 Option pada Angry IP Scanner

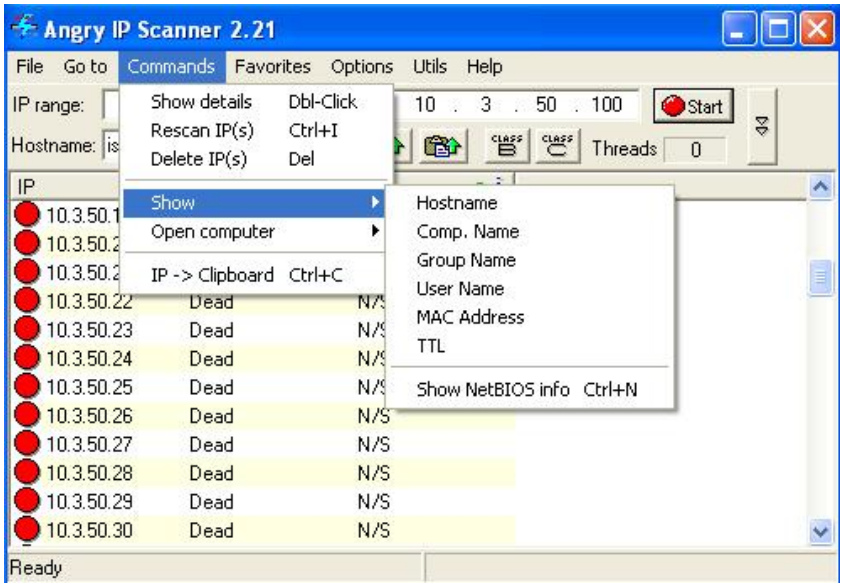
- Pada “IP Range” masukkan Alamat IP 10.3.50.1 to 10.3.50.100. Kemudian klik “Start”
- Perhatikan jumlah dan IP berapa saja yang aktif.
- Pelajari lagi fitur-fitur yang ditawarkan oleh Angry IP Scanner.



- Pada menu bar klik “Commands” – “Open Computers”. Anda bisa membuka IP tertentu berdasarkan layanan yang aktif misalnya adalah bila layanan web server aktif maka anda bisa membukanya dengan klik “Web Server (HTTP)”



- Anda juga bisa menampilkan keterangan-keterangan tambahan melalui “Command” - “Show”. Disana akan muncul “hostname”, “Comp. Name”, “Group Name”, “User Name” dst.

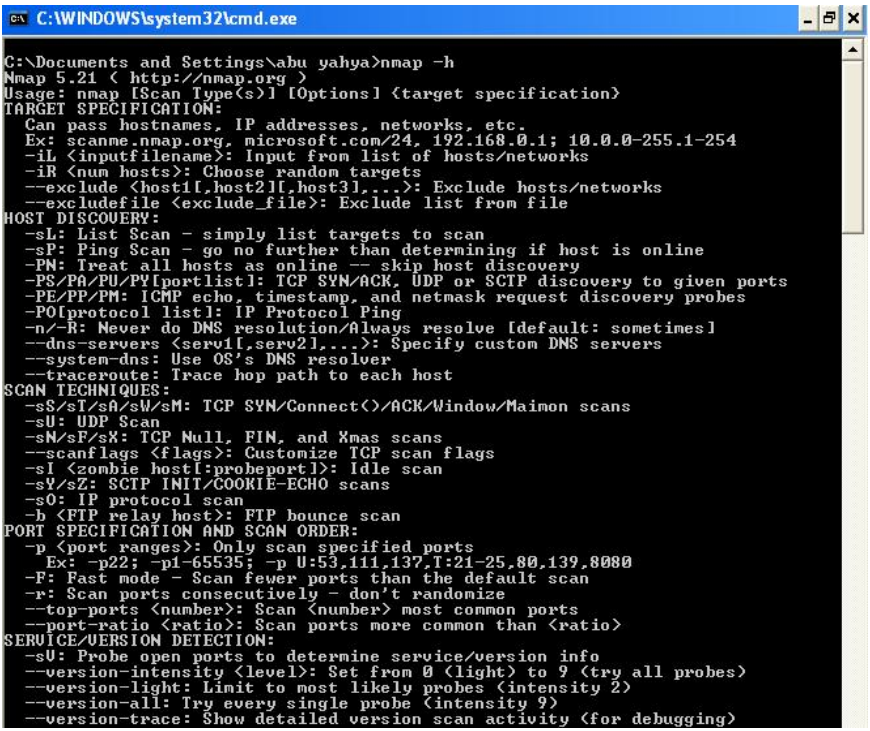


Tugas Percobaan 1 :

1. Gunakan Angry IP Scanner untuk mengscan IP Address 10.3.50.1 - 10.3.50.254
2. Tuliskan IP Address berapa saja yang aktif
3. Pada Menu “Options” – “Options”, settinglah agar hanya bisa men-scan computer yang hidup saja. Kemudian Scan ulang.
4. Print screen tampilan yang muncul dan jadikan sebagai laporan
5. Pada Menu “Options” – “Options”, settinglah agar bisa men-scan computer yang hidup dan portnya terbuka saja. Kemudian Scan ulang.
6. Print screen tampilan yang muncul dan jadikan sebagai laporan

2.5.2. Percobaan 2: Mencari Port yang terbuka dengan Nmap

- Install program nmap terlebih dahulu
- Buka “command prompt”
- Ketik “nmap” kemudian enter



- Lakukan scanning denga mengetik → nmap -sS 10.3.50. xxx (IP Address yang aktif)
- Print screen hasil scanning kalian
- Ketik → nmap -sT 10.3.50.xxx (IP Address yang aktif)
- Print out hasil scanning kalian. Perhatikan apakah hasilnya berbeda.
- scanning port lebih dari satu target computer gunakan perintah → Nmap -sS 10.3.50.xxx-xxx

Tugas Percobaan 2:

1. Lakukan scanning port dengan menggunakan nmap ke minimal 3 website memakai teknik -sS dan -sT
2. Print screen hasilnya dan jadikan bahan laporan
3. Sebutkan teknik-teknik scan yang bisa dilakukan oleh nmap dan jelaskan perbedaan masing-masing teknik tersebut.

2.5.3. Percobaan 3: Scanning Port dengan menggunakan Netcat

Langkah-langkah :

- Program Netcat bisa di download di internet
- Buka "command prompt" dan masuklah ke folder dimana anda menyimpan program netcat
- Untuk mengetahui fitur-fitur pada netcat ketik perintah :
- nc -help
- lakukan scanning port TCP mulai dari port 1-150 dengan mengetik perintah :
- nc -v -z -w2 10.3.50.xxx 1-150
- print screen hasilnya kemudian jadikan bahan laporan
- lakukan scanning port UDP mulai dari port 1-150 dengan mengetik perintah :
- nc -u -v -z -w2 10.3.50.xxx 1-150
- print screen hasilnya kemudian jadikan bahan laporan. Perhatikan perbedaannya.

Tugas Percobaan 3:

1. Lakukan scanning port mulai dari port 1-1000 dengan menggunakan netcat ke minimal 3 website.
2. Print screen hasilnya dan jadikan bahan laporan

2.5.4. Percobaan 4: Mendeteksi Sistem Operasi Target dengan Nmap

Langkah-langkah :

- Buka "command prompt"
- Ketik perintah : nmap -O 10.3.50.xxx
- Print screen hasilnya dan jadikan bahan laporan

- Ketik perintah : nmap -p80 -O 10.3.50.xxx
- Print screen hasilnya dan jadikan bahan laporan

Tugas Percobaan 4 :

1. Lakukan seperti langkah 1-5 diatas akan tetapi ganti IP Address nya dengan IP Publik website tertentu. (minimal 3 website)
2. Print Screen hasilnya dan jadikan sebagai bahan laporan

3

NETWORK MONITORING DAN LOG ANALYSIS

3.1. ALOKASI WAKTU DAN PERSIAPAN

Praktikum ini terdiri dari 3 percobaan, untuk menyelesaikan semua percobaan pada modul ini membutuhkan waktu 100 menit atau satu kali pertemuan.

Untuk kelancaran praktikum, asisten praktikum sebaiknya sudah menyiapkan software-software dan meng-*install*-nya pada komputer yang digunakan sebagai praktikum. Asisten juga harus memantau kondisi jaringan pada semua komputer dan memastikan komputer server dan client terhubung dengan baik.

3.2. DASAR TEORI

3.2.1. Footprinting

Footprinting adalah fase persiapan ketika penyerang mengumpulkan/ mencari sebanyak mungking informasi yang mungkin tentang target sebelum dilakukan penyerangan. Metode yang tepat sangat beragam. Di lab ini kita akan mempelajari teknik mengumpulkan informasi non-intrusive (tanpa mengganggu). Disini tidak ada sistem yang dilanggar atau diakses. Informasi datang dari sumber public terpecaya.

Pada umumnya penggunaan tool untuk footprinting adalah tool whois. Tool whois akan mengumpulkan semua informasi tentang perusahaan target dari beberapa database yang terdistribusi di dunia.

Bila alamat IP target telah diperoleh, penyerang bisa melacak rute antara sistem dan sistem target. Kebanyakan sistem operasi menyediakan sebuah utilitas 'traceroute' pada detail jalur perjalanan

IP paket antara dua sistem. Alat traceroute visual seperti Neotrace dan VisualRoute bahkan memberikan informasi cukup rinci pada setiap 'hop' pada rute, seperti informasi kemana pemilik dan jaringan tersebut terdaftar.

3.2.2. Logging

Logging merupakan prosedur di mana sebuah sistem operasi atau aplikasi merekam setiap kejadian dan menyimpan rekaman tersebut untuk dapat dianalisa di kemudian hari. Kejadian yang direkam ini bisa saja menyangkut sistem operasi, atau khusus program-program tertentu saja. Linux memiliki fasilitas logging yang sangat komprehensif.

Semua file log di Linux disimpan dalam direktori /var/log. Beberapa program/file log yang penting adalah :

a. lastlog

   Berisi rekaman kapan user login terakhir kali. Yang ditampilkan adalah nama login, port dan waktu login terakhir kali. Untuk memanggilnya cukup ketikkan lastlog.

b. xferlog (vsftpd.log)

   Mencatat semua informasi yang pernah login di ftp daemon. Data yang ditampilkan berupa waktu saat ini, durasi transfer file, host yang mengakses (baik nomor IP maupun nama host), jumlah file yang ditransfer, nama file, tipe transfer (Binary atau ASCII), perintah khusus yang diberikan (jika file dikompres atau tar), arah transfer (incoming, outgoing), modus akses (anonymous, guest, atau user resmi), nama user, layanan, metode otentikasi, dan user ID.

c. access_log

   Berisi rekaman untuk layanan http (HyperText Transfer Protocol) atau layanan web server. Access_log biasanya terdiri dari Nomor IP dari pengakses, jam dan tanggal akses, printah atau permintaan, dan kode status.

d. error_log

   Berisi rekaman pesan kesalahan atas service http atau web server. Error_log terdiri dari jam dan waktu, tipe kesalahan, alasan kesalahan, layanan, dan perintah yang dijalankan berikutnya (kadang-kadang).

e. messages

Rekaman kejadian sistem dan kernel, ditangani oleh dua daemon, syslogd merekam semua program yang dijalankan. Untuk mengkonfigurasikannya dapat mempergunakan syslog.conf. klogd, menerima dan merekam pesan kernel File messages dapat dilihat di / var/log/messages.

f. Host

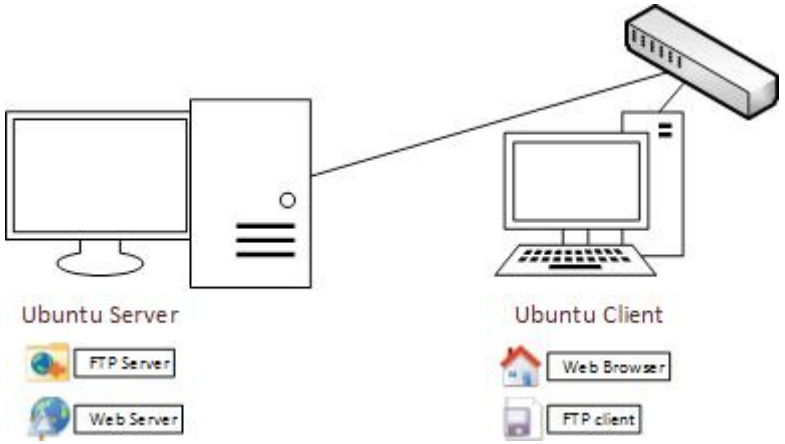
Merupakan sebuah utiliti yang sederhana yang dapat digunakan untuk melihat ip suatu host dengan memanfaatkan DNS.

3.3. TUJUAN

1. Mengenalkan konsep Manajemen Log di linux
2. Mengetahui berbagai macam file log yang ada di linux
3. Melakukan analisa terhadap file log yang ada di linux
4. Melakukan monitoring terhadap file log di linux

3.4. BAHAN DAN ALAT

1. Siapkan dua buah komputer dengan sistem operasi Ubuntu Linux yang terhubung dalam sebuah jaringan
2. Komputer pertama menggunakan sebagai server yang memiliki layanan WEB (menggunkan Apache2) dan FTP (menggunakan vsftpd)
3. Komputer kedua digunakan sebagai clien yang mengakses layanan pada server, komputer ini dilengkapi dengan Firefox sebagai web browser dan aplika Filezila sebagai FTP *client*



3.5. LANGKAH PERCOBAAN

3.5.1. Percobaan 1: Menggunakan perintah lastlog

1. Pada komputer server jalankan perintah lastlog

```
root@fki-ums:/# lastlog
```

contoh hasil:

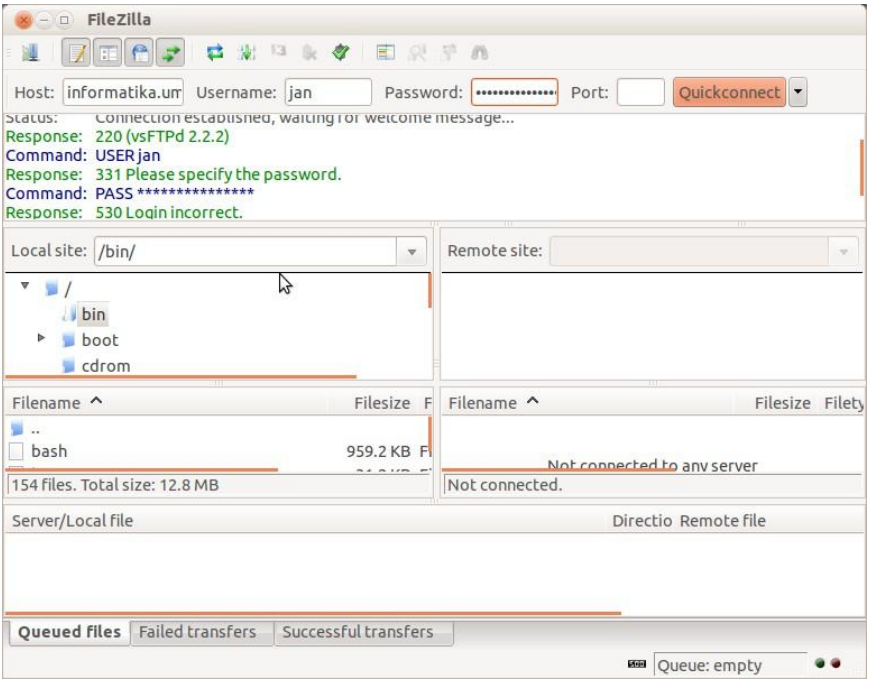
```
Username    Port   From   Latest
root                      **Never logged in**
daemon                    **Never logged in**
…
…
andi        pts/2         Mon Dec 20 11:03:14 +0700 2010
joko        pts/1         Mon Dec 20 11:01:45 +0700 2010
```

2. Amati hasil percobaan dan lakukan analisis terhadap hasil tersebut kemudian simpulkan mengapa hasilnya seperti itu.

3.5.2. Percobaan 2: Informasi yang pernah login di ftp daemon

1. Pastikan ftp server telah di-*install* (vsftpd), gunakan komputer klien untuk melakukan proses *download* sebuah

file pada komputer server menggunakan protokol ftp (anda dapat menggunakan program FileZilla pada komputer klien untuk melakukan upload dan download).



2. Setelah klient melakukan download, masuk ke server dan baca file vsftpd.log pada server dengan perintah:

```
$ sudo cat /var/log/vsftpd.log
```

contoh hasil:

```
Mon Dec 20 12:37:57 2010 [pid 2] CONNECT: Client
"192.168.56.1"
Mon Dec 20 12:37:58 2010 [pid 1] [joko] OK LOGIN:
Client "192.168.56.1"
Mon Dec 20 12:39:10 2010 [pid 2] CONNECT: Client
"192.168.56.1"
Mon Dec 20 12:39:10 2010 [pid 1] [joko] OK LOGIN:
Client "192.168.56.1"
Mon Dec 20 12:41:13 2010 [pid 2] CONNECT: Client
```

```
"192.168.56.1"
Mon Dec 20 12:41:13 2010 [pid 1] [joko] OK LOGIN:
Client "192.168.56.1"
Mon Dec 20 12:41:14 2010 [pid 3] [joko] OK
DOWNLOAD: Client "192.168.56.1", "/home/joko/
examples.desktop", 179 bytes, 5.39Kbyte/sec
```

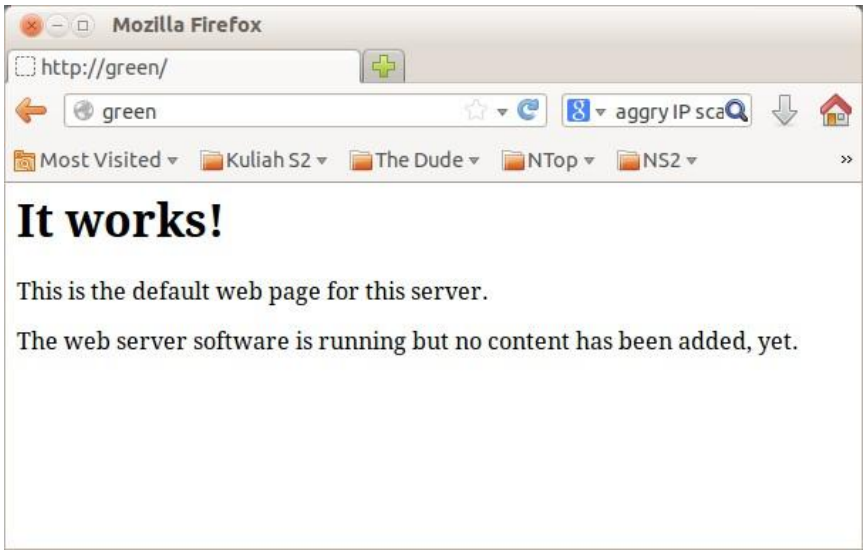
3. Gunakan komputer klien untuk melakukan proses upload sebuah file ke server. Setelah client melakukan upload, baca file vsftpd.log pada server.

```
root@fki-ums:/# cat /var/log/vsftpd.log
```

4. Amati hasil percobaan dan lakukan analisis terhadap hasil tersebut kemudian simpulkan mengapa hasilnya seperti itu.

3.5.3. Percobaan 3: Mengamati log pengaksesan sebuah halaman web

1. Pastikan paket http server sudah terinstal (apache web server), gunakan komputer clien untuk mengakses halaman web dari server tersebut menggunkan web browser.



2. Baca file error.log dan access.log pada komputer server.

```
# cat /var/log/apache2/error.log
# cat /var/log/apache2/access.log
```

3. Amati dan pelajari hasilnya, kemudian simpulkan!
4. Baca file log system pada komputer server.

```
# tail -f /var/log/syslog
# tail -f /var/log/messages
```

5. Amati hasil percobaan dan lakukan analisis terhadap hasil tersebut kemudian simpulkan mengapa hasilnya seperti itu.

4

FIREWALL

4.1. ALOKASI WAKTU DAN PERSIAPAN

Praktikum ini terdiri dari 6 percobaan, untuk menyelesaikan semua percobaan pada modul ini membutuhkan waktu 200 menit atau dua kali pertemuan. Pada pertemuan pertama dapat dimulai dari percobaan 1,2 dan 3, kemudian pertemuan berikutnya dilanjutkan percobaan 4, 5 dan 6.

Untuk kelancaran praktikum, asisten praktikum sebaiknya sudah menyiapkan software-software dan meng-*install*-nya pada komputer yang digunakan sebagai praktikum. Asisten juga harus memantau kondisi jaringan pada semua komputer dan memastikan komputer server dan client terhubung dengan baik.

4.2. DASAR TEORI

4.2.1. Firewall

Firewall adalah sebuah sistem pengaman, jadi firewall bisa berupa apapun baik hardware maupun software. Firewall dapat digunakan untuk memfilter paket-paket dari luar dan dalam jaringan di mana ia berada. Jika pada kondisi normal semua orang dari luar jaringan anda dapat bermain-main ke komputer anda, dengan firewall semua itu dapat diatasi dengan mudah.

Firewall merupakan perangkat jaringan yang berada di dalam kategori perangkat Layer 3 (Network layer) dan Layer 4 (Transport layer) dari protocol 7 OSI layer. Seperti diketahui, layer 3 adalah layer yang mengurus masalah pengalamatan IP, dan layer 4 adalah menangani permasalahan port-port komunikasi (TCP/UDP). Pada kebanyakan firewall, filtering belum bisa dilakukan pada level data link layer atau layer 2 pada 7 OSI layer. Jadi dengan demikian, sistem pengalamatan

MAC dan frame-frame data belum bisa difilter. Maka dari itu, kebanyakan firewall pada umumnya melakukan filtering dan pembatasan berdasarkan pada alamat IP dan nomor port komunikasi yang ingin dituju atau diterimanya.

Firewall yang sederhana biasanya tidak memiliki kemampuan melakukan filterin terhadap paket berdasarkan isi dari paket tersebut. Sebagai contoh, firewall tidak memiliki kemampuan melakukan filtering terhadap e-mail bervirus yang Anda download atau terhadap halaman web yang tidak pantas untuk dibuka. Yang bisa dilakukan firewall adalah melakukan blokir terhadap alamat IP dari mail server yang mengirimkan virus atau alamat halaman web yang dilarang untuk dibuka. Dengan kata lain, firewall merupakan sistem pertahanan yang paling depan untuk jaringan Anda.

Tetapi, apakah hanya sampai di situ saja fungsi dari perangkat firewall? Ternyata banyak firewall yang memiliki kelebihan lain selain daripada filtering IP address saja. Dengan kemampuannya membaca dan menganalisis paket-paket data yang masuk pada level IP, maka firewall pada umumnya memiliki kemampuan melakukan translasi IP address. Translasi di sini maksudnya adalah proses mengubah sebuah alamat IP dari sebuah alamat yang dikenal oleh jaringan diluar jaringan pribadi Anda, menjadi alamat yang hanya dapat dikenal dan dicapai dari jaringan lokal saja. Kemampuan ini kemudian menjadi sebuah fasilitas standar dari setiap firewall yang ada di dunia ini. Fasilitas ini sering kita kenal dengan istilah Network Address Translation (NAT).

Firewall bisa Anda dapatkan dengan berbagai cara. Jika tidak ingin repot-repot membuat dari nol, Anda harus mengeluarkan uang yang cukup banyak untuk membeli perangkat keras firewall yang sudah jadi dan tinggal Anda pasang saja di jaringan. Tetapi perlu diingat, tidak semua perangkat keras firewall dapat bekerja hebat dalam melakukan IP filtering. Jadi akan percuma saja uang yang anda keluarkan jika anda membeli firewall yang tidak andal.

Jika anda mau sedikit repot, namun hasilnya mungkin akan memuaskan anda, buat saja sendiri perangkat firewall anda. Yang anda perlukan hanyalah sebuah PC dengan processor dan memory yang lumayan besar dan sebuah aplikasi firewall

yang canggih dan lengkap yang dapat memenuhi semua kebutuhan Anda.

4.2.2. IPTables

Aplikasi firewall yang lengkap dan canggih pada umumnya juga mengharuskan Anda mengeluarkan kocek yang tidak sedikit. Seperti misalnya Checkpoint yang sudah sangat terkenal dalam aplikasi firewall, untuk memilikinya anda harus merogoh kocek yang lumayan banyak pula.

Namun jika anda pecinta produk-produk open source dan sudah sangat familiar dengan lingkungan open source seperti misalnya operating system Linux, ada satu aplikasi firewall yang sangat hebat. Aplikasi ini tidak hanya canggih dan banyak fasilitasnya, namun aplikasi ini juga tidak akan membuat kantong Anda dirogoh dalam-dalam. Bahkan Anda bisa mendapatkannya gratis karena aplikasi ini pada umumnya merupakan bawaan default setiap distro Linux. Aplikasi dan system firewall di sistem open source tersebut dikenal dengan nama IPTables.

Dengan menggunakan IPTables, Anda dapat membuat firewall yang cukup canggih dengan program open source yang bisa dengan mudah Anda dapatkan di Internet. Memang perlu diakui, firewall dengan menggunakan IPTables cukup sulit bagi pemula baik di bidang networking maupun pemula di bidang operating system Linux. Namun jika Anda pelajari lebih lanjut, sebenarnya firewall ini memiliki banyak sekali fitur dan kelebihan yang luar biasa.

Fitur yang dimiliki IPTables:

a. Connection Tracking Capability yaitu kemampuan unutk inspeksi paket serta bekerja dengan icmp dan udp sebagaimana koneksi TCP.
b. Menyederhanakan perilaku paket-paket dalam melakukan negosiasi built in chain (input,output, dan forward).
c. Rate-Limited connection dan logging capability. Kita dapat membatasi usaha-usaha koneksi sebagai tindakan preventif serangan Syn flooding denial of services(DOS).
d. Kemampuan untuk memfilter flag-flag dan opsi tcp dan address-address MAC.

Iptables mengizinkan user untuk mengontrol sepenuhnya jaringan melalui paket IP dengan system LINUX yang diimplementasikan pada kernel Linux. Sebuah kebijakan atau Policy dapat dibuat dengan iptables sebagai polisi lalulintas jaringan. Sebuah policy pada iptables dibuat berdasarkan sekumpulan peraturan yang diberikan pada kernel untuk mengatur setiap paket yang datang. Pada iptable ada istilah yang disebut dengan Ipchain yang merupakan daftar aturan bawaan dalam Iptables. Ketiga chain tersebut adalah INPUT, OUTPUT dan FORWARD.

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.

4.2.3. Perintah Dasar

Secara umum syntax iptables dapat ditulis dengan format:

```
$ sudo iptables ŧoption [Chain][Rule] -j [Target]
```

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 / tolak 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.

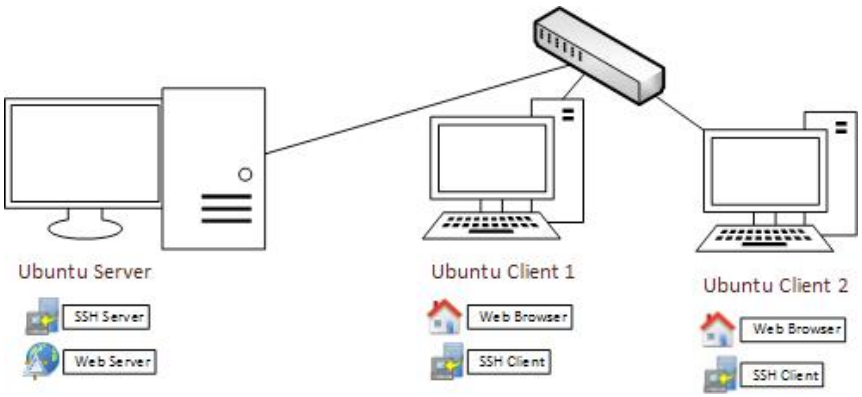
-v – Menampilkan lebih banyak informasi di layar. Sangat membantu jika ada beberapa aturan yang tampak mirip jika di tampilkan tanpa -v.

4.3. TUJUAN

1. Memahami penggunaan sistem firewall di Linux dengan menggunakan iptables
2. Mengetahui cara memblokir atau mengijinkan trafik jaringan sesuai dengan kriteria yang diinginkan

4.4. BAHAN DAN ALAT

1. Siapkan tiga buah komputer yang terhubung dalam sebuah jaringan
2. Satu komputer dengan sistem operasi ubuntu sebagai server yang memiliki layanan WEB dan SSH.
3. Dua komputer berikutnya sebagai klien yang akan mengakses layanan pada server.



4.5. LANGKAH PERCOBAAN

4.5.1. Percobaan 1: Menghapus “rules” pada iptables

1. Cek dahulu aturan “rules” yang sudah ada di iptables pada server, gunakan perintah ini:

```
$ sudo iptables -L
```

2. Lihat hasilnya, jika baru saja menginstalasi server biasanya masih belum ada rules yang terpasang.

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

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

```
Chain OUTPUT (policy ACCEPT)
```

3. Jika sudah terdapat tabel "rule" yang terpasang, hapus dulu dengan perintah:

```
iptables -F
```

Tabel "rule" dihapus karena kita akan mencoba mempelajari dari awal, jika pada sistem yang sesungguhnya (bukan latihan atau praktikum) hati-hati menggunakan perintah ini, disarankan untuk membackup tabel terlebih dahulu.

4.5.2. Percobaan 2: *Blocking* semua trafik

1. Pada server pastikan "rule" sudah terhapus dengan memberi perintah iptables -L seperti pada percobaan 1. Pada bagian ini kita akan mencoba untuk menghambat semua trafik sehingga tidak ada layanan yang dapat melaluinya.
2. Mengijinkaan sesi sambungan yang terbentuk untuk menerima traffic dari semua klien dengan perintah,

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

3. Lakukan pengujian berikut dari komputer klien:
   a. Dapatkah komputer terhubung dengan server? (gunakan perintah ping)
   b. Dapatkah komputer klien mengakses layanan SSH? (ssh dicoba melalui terminal linux atau software PuTTY)
   c. Dapatkah komputer klien mengakses layanan web? (web dicoba menggunakan Browser Firefox)

d. Blocking Semua Traffic dengan memberilkan perintah

```
$ sudo iptables -A INPUT -j DROP
```

4. Lakukan pengujian berikut dari komputer klien:
   a. Dapatkah komputer terhubung dengan server? (gunakan perintah ping)
   b. Dapatkah komputer klien mengakses layanan SSH? (ssh dicoba melalui terminal linux atau software PuTTY)
   c. Dapatkah komputer klien mengakses layanan web? (web dicoba menggunakan Browser Firefox)

4.5.3. Percobaan 2: Mengijinkan Traffic Masuk hanya ke Port SSH

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.

1. Pada server hapus dahulu tabel “rule” yang sudah terpasang dengan perintah

```
iptables -F
```

2. Mengijinkaan sesi sambungan yang terbentuk untuk menerima traffic dengan perintah,

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

3. 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).

4. 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
```

5. Untuk blocking traffic, kita harus secara spesifik mengijinkan TCP traffic ke port SSH, tapi kita belum mem-block apa-apa, dan semua traffic masuk bisa masuk. Jika aturan telah memutusan untuk menerima packet (ACCEPT), maka aturan selanjutnya tidak akan berefek pada packet tersebut. Karena aturan yang kita buat mengijinkan SSH traffic, selama aturan untuk memblok semua traffic kita letakan terakhir sesudah aturan mengijinkan SSH, maka kita akan tetap dapat menerima traffic SSH 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
DROP       all  --  anywhere       anywhere
```

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

6. Lakukan pengujian berikut dari komputer klien:
   a. Dapatkah komputer terhubung dengan server? (gunakan perintah ping)
   b. Dapatkah komputer klien mengakses layanan SSH? (ssh dicoba melalui terminal linux atau software PuTTY)
   c. Dapatkah komputer klien mengakses layanan web? (web dicoba menggunakan Browser Firefox)

4.5.4. Percobaan 4: Mengijinkan Traffic Masuk hanya ke Port Web dan SSH

1. Hapus dahulu tabel "rule" yang sudah terpasang dengan perintah

```
iptables -F
```

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

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

3. Mengijinkan semua traffic SSH untuk masuk, gunakan perintah berikut

```
$ sudo iptables -A INPUT -p tcp --dport 22 -j
ACCEPT
```

4. Cek aturan yang kita buat mengunakan perintah iptables -L, sebagai 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:www
ACCEPT     tcp  --  anywhere       anywhere
tcp dpt:ssh
```

5. Blocking Traffic

   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.

   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 port maupun semua interface akan di blok, kecuali web dan SSH.

6. Lakukan pengujian berikut dari komputer klien:
   a. Dapatkah komputer terhubung dengan server? (gunakan perintah ping)
   b. Dapatkah komputer klien mengakses layanan SSH? (ssh dicoba melalui terminal linux atau software PuTTY)

c. Dapatkah komputer klien mengakses layanan web? (web dicoba menggunakan Browser Firefox)

4.5.5. Percobaan 5: Blocking ip address

Cara memblock ip dan mac address yang berada di sebuah network merupakan hal mudah di Linux. Yang penting kita sudah mengetahui alamat ip dan mac adress yang dipakai oleh komputer target. Untuk mengetahui alamat ip dan MAC Address target, dapat dilakukan scanning pada jaringan terlebih dahulu. Anda bisa melakukan scan dengan bantuan ping dan arp. Cara yang lain untuk scanning ip adalah dengan menggunakan tools nmap, dengan melakukan nmap ke salah satu ip address pada sebuah jaringan, maka alamat ip lainnya (yang aktif) bisa ditemukan.

1. Hapus dahulu tabel "rule" yang sudah terpasang dengan perintah

```
iptables -F
```

2. Untuk melakukan bloc ip address, silahkan gunakan perintah iptables seperti berikut :

```
$ sudo iptables -I INPUT -s 10.0.2.212 -j DROP
```

Contoh di atas adalah perintah untuk mem-block ip 10.0.2.212 ke server (sesuaikan dengan ip dengan salah satu komputer klien saat anda melakukan praktikum).

3. Percobaan pada komputer klien 1 dan klien 2:
   a. Dapatkah komputer klien melakukan ping terhadap komputer server?
   b. Dapatkah komputer klien mengakses layanan SSH?
   c. Dapatkah komputer klien mengakses layanan web?
4. Menghapus blocking ip address
   Perintahnya,

```
$ sudo iptables -D INPUT -s 10.0.2.22 -j DROP
```

5. Lakukan pengujian berikut dari komputer klien 1 dan klien 2:

a. Dapatkah komputer terhubung dengan server? (gunakan perintah ping)
b. Dapatkah komputer klien mengakses layanan SSH? (ssh dicoba melalui terminal linux atau software PuTTY)
c. Dapatkah komputer klien mengakses layanan web? (web dicoba menggunakan Browser Firefox)

4.5.6. Percobaan 6: Blocking MAC Address

1. Untuk memblock MAC Address, sebenarnya hampir sama.

```
$ sudo iptables -A INPUT -m mac -mac-source
00:00:b4:aa:c1:34 -j DROP
```

(sesuaikan dengan MAC Address dengan komputer klien saat anda melakukan praktikum)

2. Percobaan pada komputer klien 1 dan klien 2:
   a. Dapatkah komputer terhubung dengan server? (gunakan perintah ping)
   b. Dapatkah komputer klien mengakses layanan SSH? (ssh dicoba melalui terminal linux atau software PuTTY)
   c. Dapatkah komputer klien mengakses layanan web? (web dicoba menggunakan Browser Firefox)
3. Untuk menghapus bloking MAC Address, tinggal menjalankan perintah yang sama dengan mengganti opsi -A (add) menjadi -D (delete)

```
$ sudo iptables -D INPUT -m mac -mac-source
00:00:b4:aa:c1:34 -j DROP
```

4. Percobaan pada komputer klien 1 dan klien 2:
   a. Dapatkah komputer terhubung dengan server? (gunakan perintah ping)
   b. Dapatkah komputer klien mengakses layanan SSH? (ssh dicoba melalui terminal linux atau software PuTTY)
   c. Dapatkah komputer klien mengakses layanan web? (web dicoba menggunakan Browser Firefox)

5
SNIFFING

5.1. ALOKASI WAKTU DAN PERSIAPAN

Praktikum ini terdiri dari 1 percobaan, untuk menyelesaikan semua percobaan pada modul ini membutuhkan waktu 100 menit atau satu kali pertemuan.

Untuk kelancaran praktikum, asisten praktikum sebaiknya sudah menyiapkan software-software dan meng-*install*-nya pada komputer yang digunakan sebagai praktikum. Asisten juga harus memantau kondisi jaringan pada semua komputer dan memastikan komputer server dan client terhubung dengan baik.

5.2. DASAR TEORI

5.2.1. Sniffer paket

Sniffer paket ialah sebuah aplikasi yang dapat melihat lalu lintas data pada jaringan komputer. Dikarenakan data mengalir secara bolak-balik pada jaringan, aplikasi ini menangkap tiap-tiap paket dan kadang-kadang menguraikan isi dari RFC (Request for Comments) atau spesifikasi yang lain. Berdasarkan pada struktur jaringan (seperti hub atau switch), salah satu pihak dapat menyadap keseluruhan atau salah satu dari pembagian lalu lintas dari salah satu mesin di jaringan. Perangkat pengendali jaringan dapat pula diatur oleh aplikasi penyadap untuk bekerja dalam mode campur-aduk (promiscuous mode) untuk “mendengarkan” semuanya (umumnya pada jaringan kabel).

Sniffer paket dapat dimanfaatkan untuk hal-hal berikut:

1. Mengatasi permasalahan pada jaringan komputer.
2. Mendeteksi adanya penyelundup dalam jaringan (Network Intusion).

3. Memonitor penggunaan jaringan dan menyaring isi isi tertentu.
4. Memata-matai pengguna jaringan lain dan mengumpulkan informasi pribadi yang dimilikanya (misalkan password).
5. Dapat digunakan untuk Reverse Engineer pada jaringan.

5.2.2. Wireshark – Network Protocol Analyzer

Wireshark adalah salah satu dari sekian banyak tool Network Analyzer yang banyak digunakan oleh Network administrator untuk menganalisa kinerja jaringannya. Wireshark banyak disukai karena antarmukanya yang menggunakan *Graphical User Interface* (GUI) atau tampilan grafis. Wireshark merupakan software untuk melakukan analisa lalu-lintas jaringan komputer, yang memiliki fungsi-fungsi yang amat berguna bagi profesional jaringan, administrator jaringan, peneliti, hingga pengembang piranti lunak jaringan.

Wireshark dapat membaca data secara langsung dari Ethernet, Token-Ring, FDDI, serial (PPP and SLIP), 802.11 wireless LAN , dan koneksi ATM.

Program ini juga sering digunakan oleh chatters untuk mengetahui ip victimnya maupun para chatter lainnya lewat typingan room.

Seperti namanya, Wireshark mampu menangkap paket-paket data/informasi yang melewati jaringan yang kita monitor. Semua jenis paket informasi dalam berbagai format protokol pun akan dengan mudah ditangkap dan dianalisa. Karenanya tak jarang tool ini juga dapat dipakai untuk sniffing (memperoleh informasi penting spt password email atau account lain) dengan menangkap paket-paket yang melewati jaringan dan menganalisanya.

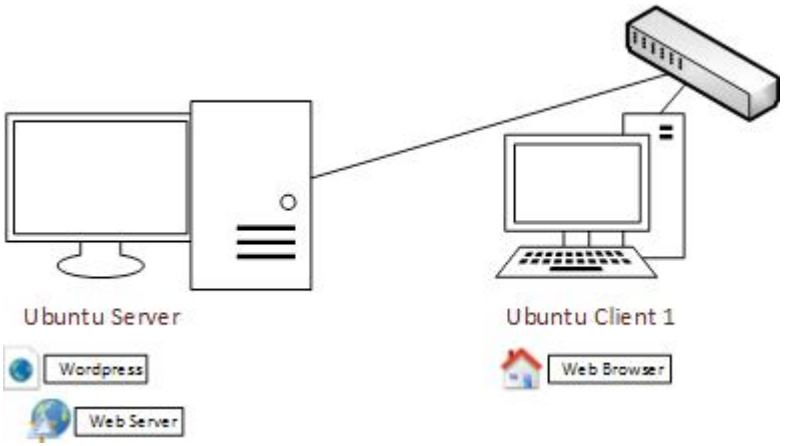
Wireshark berisi WinPcap, sebuah utilitas yang bekerja di latar belakang dengan kartu jaringan Anda. Penggunaan tools ini tidak disarankan untuk melakukan sesuatu yang negatif: hacking ke dalam sistem tanpa izin adalah kejahatan - jangan melakukannya! Modul ini dirancang untuk menunjukkan bagaimana aplikasi ini dapat menunjukkan bagaimana keamanan pada sebuah jaringan.

5.3. TUJUAN

1. Memahami konsep dasar Sniffing
2. Memahami penggunaan software sniffing

5.4. BAHAN DAN ALAT

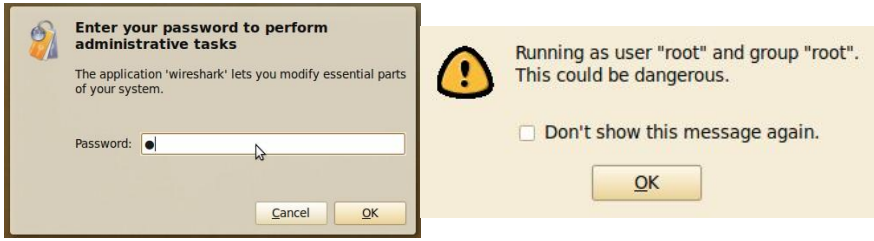
1. Siapkan dua unit komputer, satu komputer (ubuntu) sebagai server web yang berisi aplikasi wordpress, sedangkan komputer berikutnya sebagai klien yang mengakses aplikasi wordpress pada server tersebut
2. Pastikan komputer server sudah terdapat http server dan aplikasi wordpress yang dapat diakses dari komputer klien. Lakukan instalasi apabila belum ada.



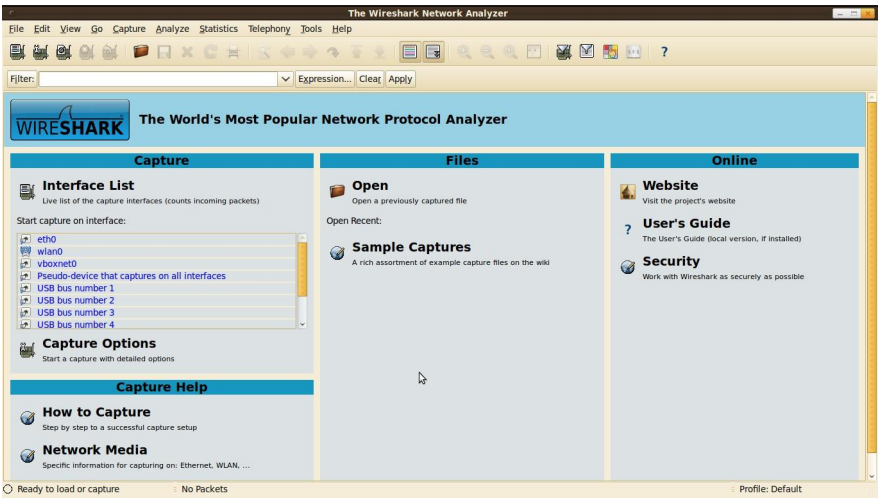
5.5. LANGKAH PERCOBAAN

1. Pada klien jalankan program Wireshark dari Application - Internet – Wireshark.

2. Masukkan pasword root jika diminta, karena membutuhkan akses root untuk menjalankan program ini.

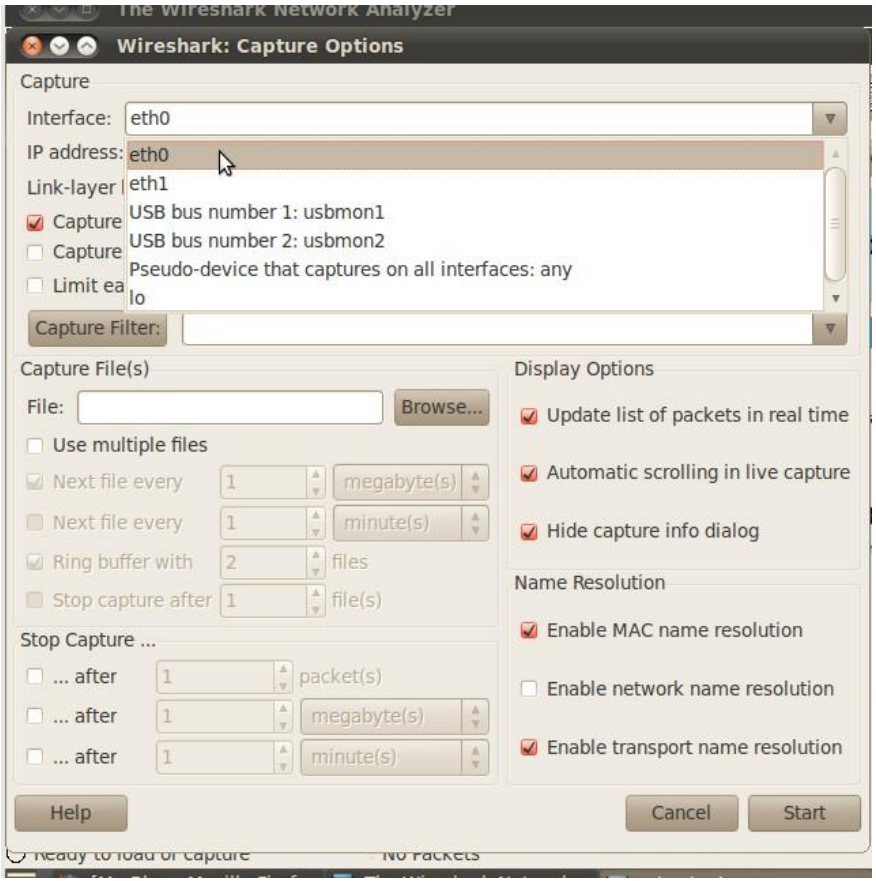


3. Tampilan Wireshark kira-kira seperti ini

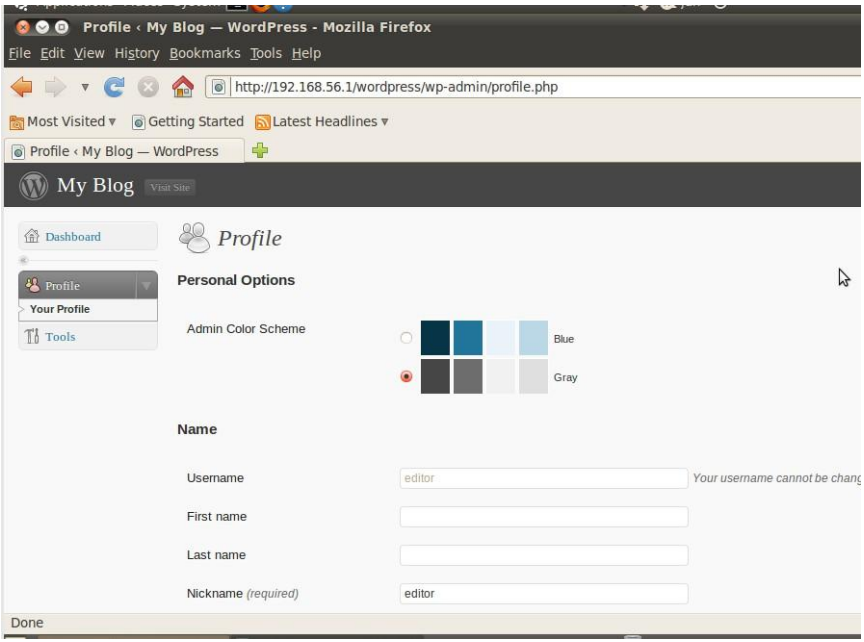


4. Selanjutnya masih dari komputer klien, jalankan browser kemudian akses aplikasi wordpress.
   (Tanyakan kepada asisten mengenai alamat server dan aplikasi wordpress yang digunakan)
5. Masukklah ke halaman login kemudian masukkan user dan password tetapi jangan login terlebuh dahulu.
   (Tanyakan user dan password yang dapat digunakan kepada asisten)

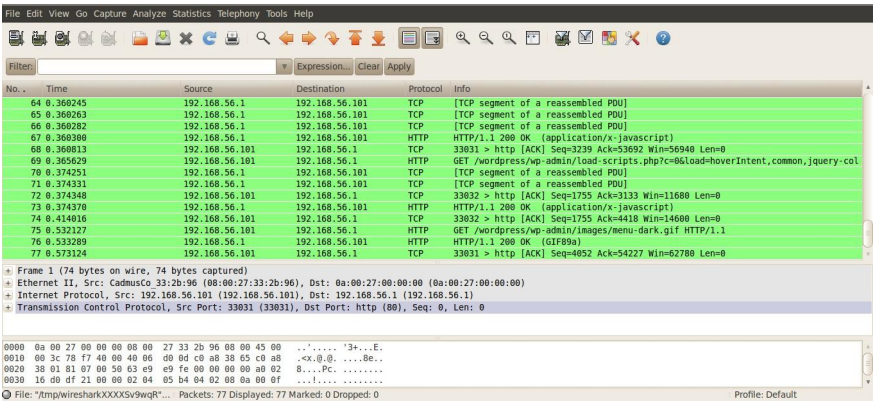
6. Kembali ke aplikasi wireshark, masuk ke menu dan pilih Capture – Option atau gunakan sortcut Ctrl + K.



7. Pada Option Interface, pilih Ethernet yang terpakai/yang tersambung dengan jaringan dalam kasus ini. Kemudian jalankan poses capture dengan menekan tombol Start.
8. Kembali lagi ke browser dan loginlah ke wordpress sehingga anda berhasil login.

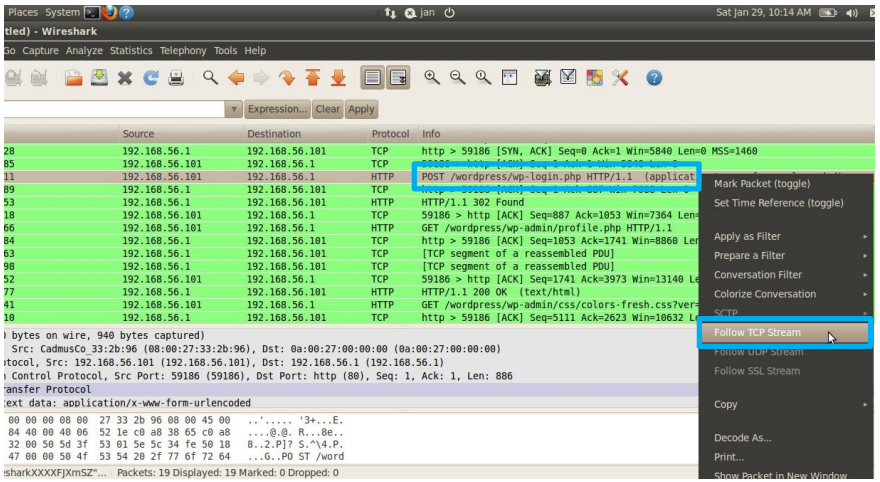


9. Kemudian masuk ke aplikasi wireshark dan hentikan proses capture dengan cara masuk ke Capture – Stop atau gunakan sortcut Ctrl+E.

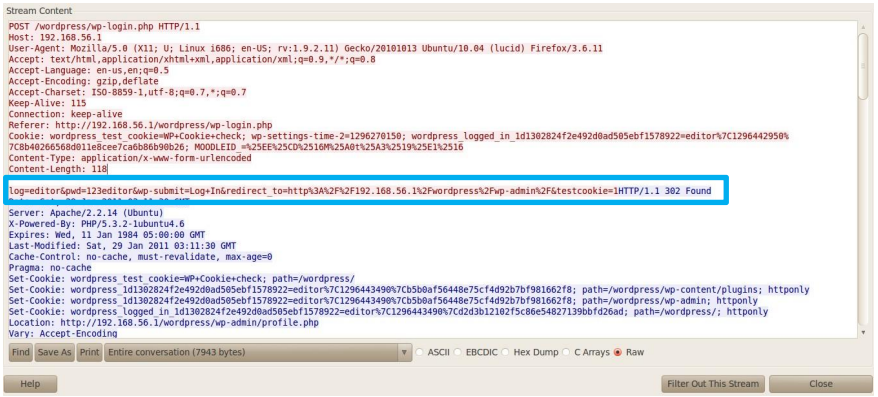


10. Biasanya login packet terdapat kata login atau sejenisnya, carilah info yang mengandung kata login. Dalam kasus ini kita menemukan packet dengan informasi POST / wordpress/wp-login.php HTTP/1.1 (application/x-www-

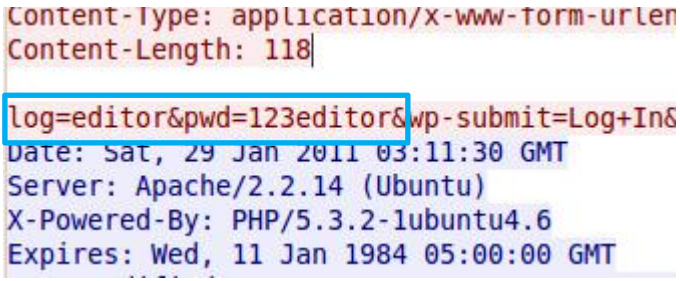
form-urlencoded) …. Klik kanan pada packet tersebut, pilih Follow TCP Stream.



11. Maka akan muncul informasi tentang packet data yang kita pilih. Disini anda dapat menemukan username dan password dari halaman wordpress. Biasanya ditandai dengan tulisan berwarna merah.



12. Perhatikan bahwa bahwa user “editor” memiliki password “123editor”



13. Simpan seluruh log sebagai lampiran laporan anda.

6

INTRUTION DETECTION SYSTEM (IDS)

6.1. ALOKASI WAKTU DAN PERSIAPAN

Praktikum ini terdiri dari 5 percobaan, untuk menyelesaikan semua percobaan pada modul ini membutuhkan waktu 200 menit atau dua kali pertemuan. Pada pertemuan pertama dapat dimulai dari percobaan 1 dan 2, kemudian pertemuan berikutnya dilanjutkan percobaan 3, 4 dan 5.

Untuk kelancaran praktikum, asisten praktikum sebaiknya sudah menyiapkan konfigurasi source-list apt agar proses instalasi dapat berjalan dengan lancar. Koneksi jaringan dengan repository sistem operasi Ubuntu juga perlu diperhatikan terutama pada percobaan 1 agar tidak mengganggu jalanya instalasi. Untuk menghindari kemungkinan kesalahan, asisten harus memastikan semua konfigurasi Tripwire belum pernah dilakukan perubahan, jika perlu *remove/uninstall* aplikasi Tripware jika sebelumnya pernah dipasang pada komputer tersebut.

6.2. DASAR TEORI

Untuk melakukan pengawasan secara otomatis terhadap penyusupan pada suatu system adalah dengan menggunakan *Intrution Detection System* (IDS). IDS bekerja dengan cara mendeteksi jenis serangan berdasarkan *signature* atau *pattern* pada aktifitas jaringan, kemudian melakukan blokade terhadap *traffic* atau aktifitas yang mencurigakan.

Tipe IDS secara garis besar dibagi 2, yaitu host base dan network base IDS. Termasuk dalam jenis network base adalah aplikasi Snort, sedangkan yang termasuk model host based adalah tripwire. Aplikasi Tripwire berfungsi untuk menjaga integritas file system dan direktori, yaitu dengan cara mencatat setiap perubahan

yang terjadi pada file dan direktori. Dalam modul ini kita hanya akan membahas host base IDS dengan Tripwire.

Tripware dapat dikonfigurasi untuk melakukan pelaporan melalui email bila menemukan perubahan file yang tidak semestinya, selain itu secara otomatis tripware juga dapat melakukan pemeriksaan file secara terjadwal melalui cron. Penggunaan tripwire biasanya digunakan untuk mempermudah pekerjaan yang dilakukan oleh System Administrator dalam mengamankan System.

Prinsip kerja tripwire adalah melakukan perbandingan file dan direktori dengan database yang sudah dibuat berdasarkan file dan direktori sebelum terjadi perubahan. Sehingga apa bila suatu file atau direktori mengalami perubahan, tripwire akan mengetahui perbedaan yang terjadi dengan cara membandingkan dengan database yang dimilikinya. Perbandingan tersebut meliputi perubahan tanggal, ukuran file, penghapusan dan lainnya. Setelah tripwire dijalankan, secara otomatis akan melakukan pembuatan database sistem. Kemudian secara periodik akan selalu melaporkan setiap perubahan pada file dan direktori.

6.3. TUJUAN

1. Mengenalkan pada mahasiswa tentang konsep integrator cek pada IDS
2. Mampu melakukan installasi, konfigurasi dan memakaai Tripwire sebagai program hostbase IDS dengan sistem integrator Checking

6.4. BAHAN DAN ALAT

1. Siapkan sebuah buah komputer dengan sistem operasi Ubuntu yang terhubung dalam jaringan internet
2. Komputer dapat mengakses repository Ubuntu dengan baik

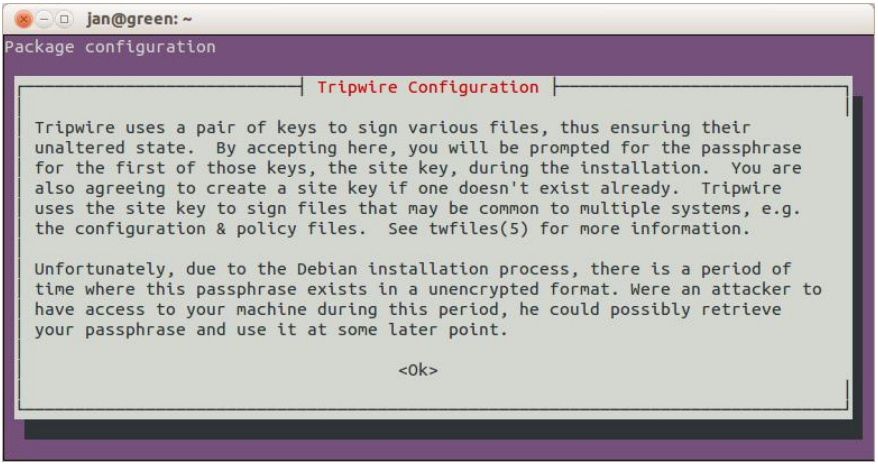
6.5. LANGKAH PERCOBAAN

6.5.1. Percobaan 1: Instalasi dan konfigurasi awal

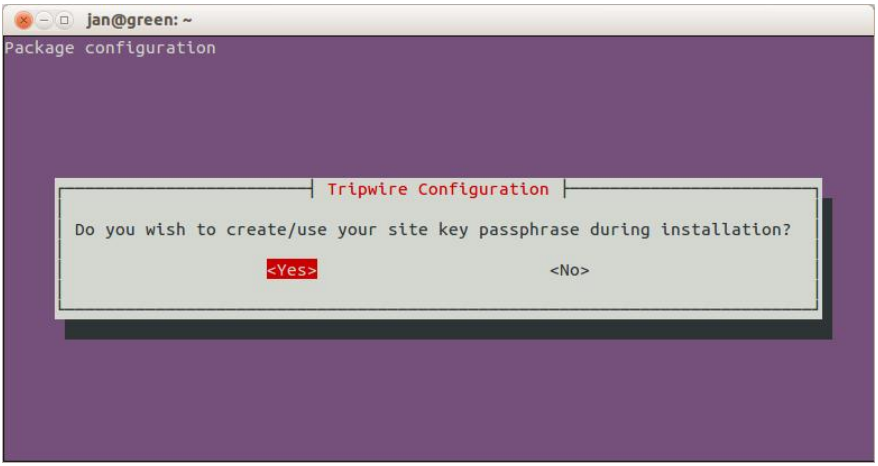
1. Langkah instalasi dimulai dengan membuka terminal dan menginstalnya tripwire dengan perintah

```
# apt-get install tripwire
```

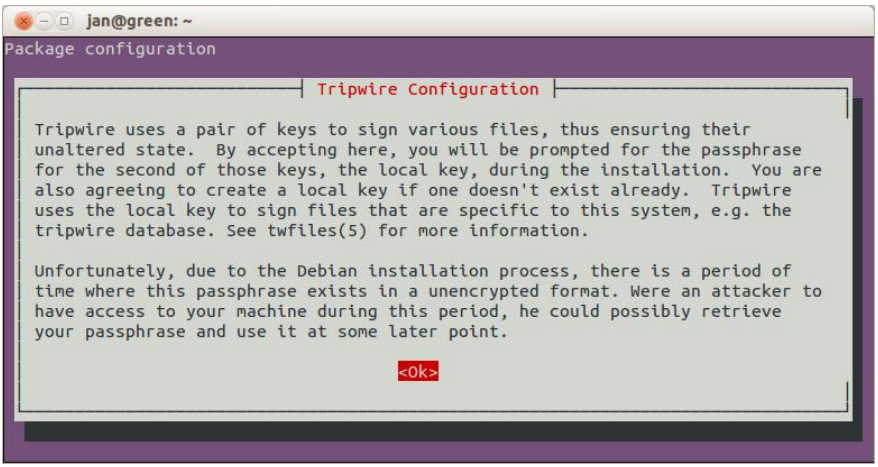
Pada awal instalasi ada petunjuk dan peringatan yang dibutuhkan dalam proses instalasi Tripwire, bacalah dengan seksama dan untuk melanjutkan, pilih OK.



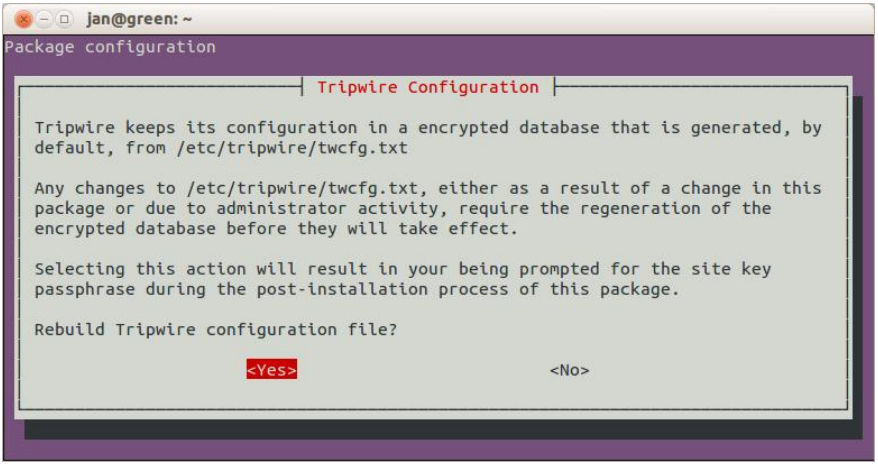
2. Muncul dialog untuk meminta membuat key passphrase, pilih yes.



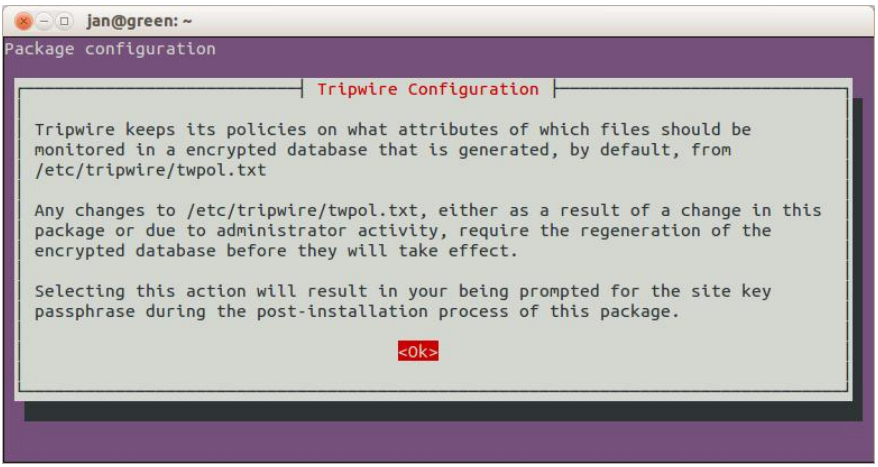
3. Ada peringatan kembali seperti awal instalasi, hal ini mengingatkan untuk membuat key yang aman.



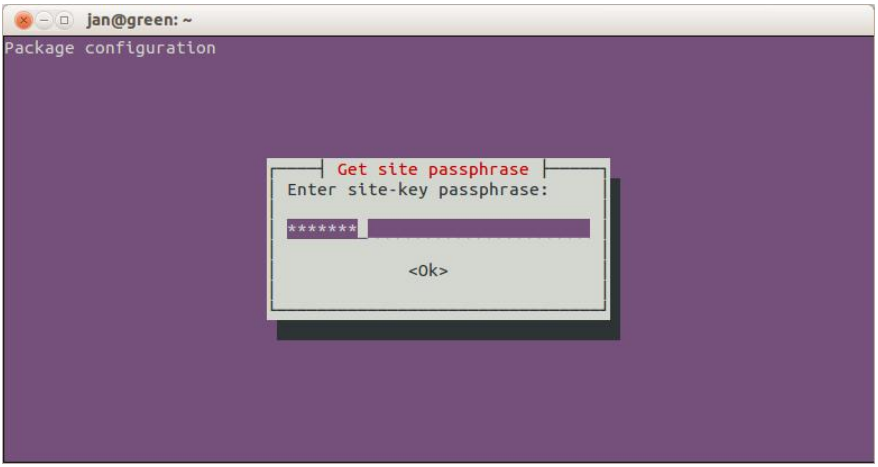
4. Kemudian ada konfirmasi apakah ingin membuat local paraphase atau tidak. Pilih yes karena paraphase juga dibuat pada local machine.
5. Kemudian ada permintaan konfirmasi apakah ingin untuk menyimpan hasil konfigurasi pada /etc/tripwire/twcfg. txt. Pilih yes untuk me-rebuild file configurasi. Konfirmasi untuk me-rebuild file konfigurasi dengan memilih Yes.



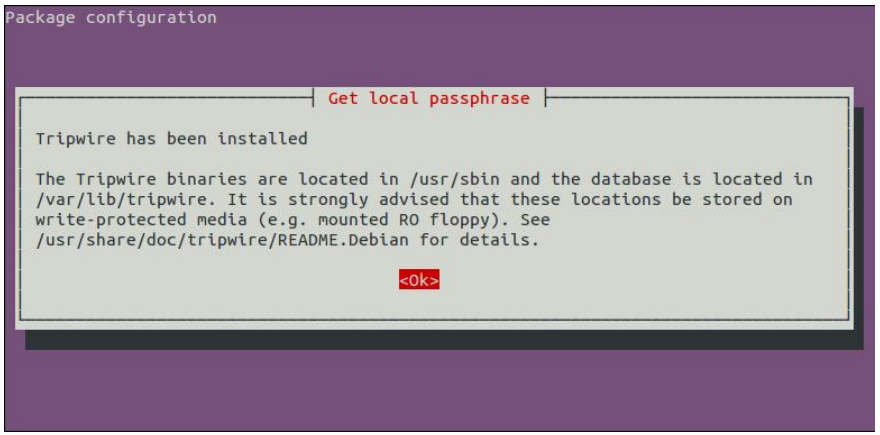
6. Setelah itu, ada permintaan konfirmasi apakah ingin menyimpan hasil policy pada /etc/tripwire/twpol.txt. Pilih yes untuk me-rebuild file policy.



7. Setelah itu, masukan local-key yang diinginkan kemudian memilih OK. Ulangi pada langkah selanjutnya memasukkan site-key setelah itu pilih OK. Pastikan passphrase yang dibuat selalu diingat, jika perlu dicatat untuk keperluan percobaan berikutnya.



8. Dan instalasi selesai setelah passphrase dimasukan dua kali. Kemudian muncul dialog yang menyatakan bahwa instasi Tripwire selesai.



9. Untuk meningkatkan keamanan, perlu dilakukan enkripsi file konfigurasi /etc/tripwire/twcfg.txt dengan menggunakan perintah

```
# twadmin --create-cfgfile --cfgfile ./tw.cfg \
--site-keyfile ./site.key ./twcfg.txt
```

6.5.2. Percobaan 2: Inisialisasi database pengecekan Tripwire

1. Dalam konfigurasi default tripware, file dan direktori yang diamati cukup banyak sehingga proses inisialisasi atau pembuatan database awal akan membutuhkan waktu yang lama. Agar proses percobaan tidak memakan waktu lama maka kita akan memonitor sebuah direktori saja. Buat direktori baru dengan perintah berikut ini.

```
# mkdir /home/praktikum
```

2. Sebagai direktori dan file yang akan kita ubah nanti, buat sebuah direktori dengan nama “dir-coba” dan sebuah file dengan nama “file-coba.txt”

```
# mkdir /home/praktikum/dir-coba
# gedit /home/praktikum/file-coba.txt
```

3. Copy file twpol sebelum melakukan perubahan

```
# cp /etc/tripwire/twpol.txt /etc/tripwire/twpol.
txt.backup
```

4. Lakukan perubahan pada konfigurasi twpool.txt agar hanya direktori /home/percobaan saja yang akan dimonitor

```
# gedit /etc/tripwire/twpol.txt
```

5. Hapus semua teks konfigurasi mulai dari baris

```
#
# Tripwire Binaries
#
```

6. Kemudian gantilah dengan konfigurasi berikut ini

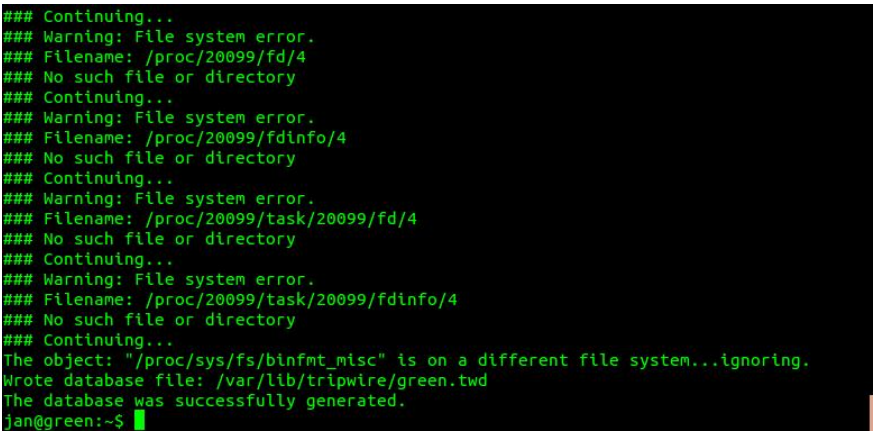
```
#
# Memonitor direktori percobaan
#
(
  rulename = "Direktori Percobaan",
  severity = $(SIG_HI)
)
{
    /home/praktikum          -> $(SEC_CRIT);
}
```

7. Selanjutnya lalukan inisialisasi database dengan perintah berikut

```
# 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
```

Untuk perintah HOSTNAME-local.key sesuaikan dengan hostname PC yang digunakan. Masukan key site passphrase yang telah dibuat sebelumnya.

Pada awal inisialisasi, akan terdapat beberapa warning dan error karena Tripwire belum mempunyai data yang sama pada databasenya, hal ini tidak masalah.



8. Untuk mengecek sistem terhadap adanya perubahan file-file dalam host, gunakan perintah.

```
# tripwire --check
```

Amati hasilnya dan copy dalam sebuah dokumen (akan digunakan dalam pengamatan dan analisa dalam laporan anda)

6.5.3. Percobaan 3: Melihat hasil monitoring Tripwire

1. Buka “file-coba.txt” dan tambahkan isi file dengan nama dan nim anda

```
# gedit /home/praktikum/file-coba.txt
```

2. Buat sebuah file dengan nama “file-coba2.txt” dalam dierektori “dir-coba”

```
# gedit /home/praktikum/dir-coba/file-coba2.txt
```

3. Cek perubahan pada sistem dengan perintah

```
# tripwire --check
```

Amati hasilnya dan copy dalam sebuah dokumen (akan digunakan dalam pengamatan dan analisa dalam laporan anda)

4. Bandingkan hasil pada langkah 3 ini dengan Percobaan 2 langkah 8 dan jelaskan perubahan yang terjadi

6.5.4. Percobaan 4: Update file policy Tripwire

1. Buat sebuah direktori lagi di dalam /home

```
# mkdir /home/praktikum2
```

2. Lakukan perubahan pada konfigurasi twpool.txt agar direktori /home/percobaan2 juga dimonitor oleh Tripwire

```
# gedit /etc/tripwire/twpool.txt
```

3. Tambahkan direktori praktikum2 dalam daftar direktori yang akan dimonitor

```
{
    /home/praktikum          -> $(SEC_CRIT);
    /home/praktikum2         -> $(SEC_CRIT);
}
```

4. Karena kita melakukan perubahan policy maka lakukan update seberti berikut ini

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

5. Cek perubahan pada sistem dengan perintah

```
# tripwire --check
```

6. Amati hasilnya dan copy dalam sebuah dokumen, selanjutnya bandingkan dengan hasil dari percobaan 3 kemudian buat penjelasan mengenai hasil tersebut.

6.5.5. Percobaan 4: Update database Tripwire

1. Misalkan anda mengubah/mengedit file tertentu, apabila database tidak diubah, perubahan file ini akan dideteksi oleh tripwire sebagai bentuk pelanggaran, walaupun perubahan tersebut legal. Buatlah sebuah file dalam direktori /home/praktikum dengan nama “file-coba2.txt”

```
# gedit /home/praktikum/file-coba2.txt
```

2. Cek perubahan pada sistem dengan perintah

```
# tripwire --check
```

   Amati hasilnya dan copy dalam sebuah dokumen (akan digunakan dalam pengamatan dan analisa dalam laporan anda)

3. Sebelum melakukan update database, perlu diperhatikan disini adalah file nama-file.twr, sesuaikan dengan file report tripwire tersebut. Jalankan perintah update berikut

```
#/usr/sbin/tripwire --update \
--twrfile /var/lib/tripwire/report/nama-file.twr
```

4. Cek lagi perubahan pada sistem dengan perintah

```
#tripwire --check
```

5. Amati hasilnya dan copy dalam sebuah dokumen, bandingkan dengan langkah nomer 2 kemudian buat penjelasan tentang hasil tersebut.

7

KEAMANAN LAYANAN WEB (SSL/TLS)

7.1. TUJUAN

1. Mengenalkan tentang konsep SSL
2. Membuat self-signed server certificate
3. Melakukan instalasi dan konfigurasi Apache2 + SSL/TLS
4. Melakukan testing instalasi

7.2. DASAR TEORI

7.2.1. Transport Layer Security

Transport Layer Security (TLS) adalah protocol untuk mengamankan komunikasi antar aplikasi lewat internet. TLS mengamankan konten pada layer aplikasi, seperti halaman web dan diimplementasikan pada layer transport, yaitu TCP. Untuk menjamin keamanan. data yang dikirim dienkripsi dan diotentikasi pada sisi server dan client. Secure Socket Layer (SSL) adalah protocol yang diciptakan sebelum TLS yang mengaplikasikan hal ini. SSL/TLS biasanya dioperasikan secara bersama-sama dengan HTTP, sehingga membentuk protocol baru yang disebut HTTPS, untuk mengamankan transaksi lewat web. Selain intu, protocol ini dapat digunakan untuk aplikasi-aplikasi lain seperti email, file transfer dan virtual private networks (VPN).

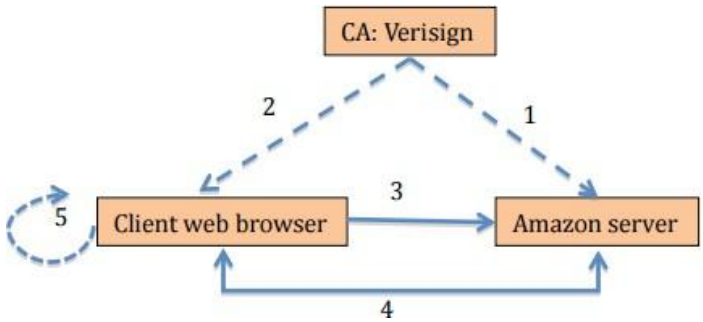
7.2.2. HTTPS

HTTPS menggabungkan protocol HTTP dan SSL/TLS untuk menjamin keamanan komunikasi antara Web server dan web browser. HTTPS beroperasi pada port 443 dan bukan pada port 80 seperti normalnya HTTP. HTTPS bekerja dengan menyediakan enkripsi untuk konten web dan otentikasi web

server. HTTPS tidak melakukan otentikasi client sehingga web sita tidak dapat melakukan otentikasi user selama koneksi. User harus melakukan sejumlah otentikasi tambahan seperti password,biometric atau metode otentikasi lain. Komunikasi SSL/TLS meliputi dua tahap yaitu handshaking dan data sending. Sebelum berkomunikasi, web site harus meminta certificate authority (CA) agar dapat menanda tangani (signing) digital certificate-nya yang berisi public key dari site. User yang menerima digital sertificate CA, segera memanggil sertifikat root, yang dimiliki ketika mereka menginstall web browser. Web browser seperti Internet Explorer atau Firefox sebelumnya telah dilengkapi dengan sejumlah sertifikat root dari bermacammacam perusahaan seperti VeriSign atau Entrust, yang memang menspesialiasisikan diri sebagai perusahaan yang bergerak di bidang sertifikasi

7.2.3. Cara Kerja HTTPS

Cara kerja HTTPS dapat dilihat dalam ilustrasi Gambar 7.1 dibawah ini :



Gambar 7.1. Proses Handshaking SSL pada HTTPS

- Langkah 1 - Verisign menandatangani sertifikat Amazon dengan publik key-nya
- Langkah 2 - Sertifikat CA dengan public key-nya akan terinstalpada browser client
- Langkah 3 - Koneksi lewat https
- Langkah 4 - Saling menukar sertifikat digital, termasuk publik key

- Langkah 5- Client melakukan verifikasi sertifikat Amazon menggunakan public key dari CA

Pada Gambar 7.1, begitu user mengkoneksikan diri dengan website lewat koneksi https, web server mengirim certifikatnya yang mengandung public key dari web site tersebut. User akan memverifikasi sertifikat ini dengan memakai preinstalled sertifikat root dari website CA.

Pada tahap kedua dari komunikasi SSL/TLS adalah tahapan enkripsi antara server dan client berdasarkan protocol kriptografi yang dinegosiasikan antara kedua belah pihak. Pada gambar berikut, begitu sertifikat digital server berhasil diverifikasi, maka browser dan server mulai saling bernegosiasi cipher yang hendak dipakai untuk pengkodean data dan verifikasi digital signature. Jika public key enkripsi sudah dipilih, kedua belah pihak mengenkripsikan data dengan public key masing-masing dan mendekripsi dengan private keynya. Untuk menghemat waktu, enkripsi public key hanya digunakan saat saling menukar session key (private key yang temporer) yang dipakai untuk data enkripsi.

7.2.4. Contoh Penerapan SSL dan Kebutuhan Sistem

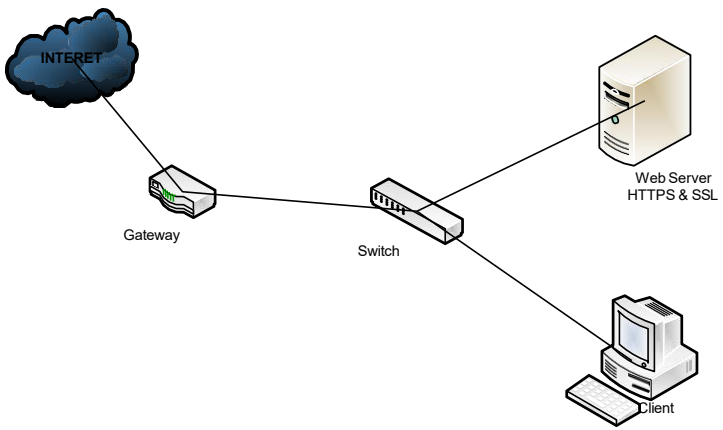
Gambar 7.2. Contoh Pemanfaatan HTTPS pada Web Citibank

Software yang dibutuhkan untuk menginstall webserver yang secure dan berbasis opensource adalah :

- Server Ubuntu/Debian
- Apache 2 : Apache2 ini adalah web server yang biasa gunakan pada sistem-sistem opensource. Anda dapat mengecek informasinya lewat http://httpd.apache.org/.
- OpenSSL 0.9.8k: OpenSSL adalah toolkit yang digunakan untuk mengimplementasikan protocol Secure Socket Layer (SSL v2/v3) dan Transport Layer Security (TLS v1). OpenSSL juga menyediakan general purpose library untuk kriptografi.
- Mod_SSL 2.2.11: Mod_SSL adalah add-on modul untuk Apache. Pada versi lama apache, user harus mengkompile paket ini, sedangkan pada versi baru, Mod_SSL sudah built-in pada server sebagai interface antara OpenSSL dan Apache2. Informasi mengenai Mod_SSL dapat dilihat pada http://www.mod_ssl.org.

7.3. LANGKAH PRAKTIKUM

Pada bagian ini akan dipraktekkan bagaimana menginstall web server yang secure dengan menggunakan SSL/TLS. Mengenai software yang dibutuhkan, bisa dibaca pada dasar teori.



Gambar 7.3. Topologi Praktek

Sebagai langkah awal installah server ubuntu/debian beserta software-software tersebut. seperti pada Gambar 7.3. Tahap paling penting dari praktikum ini adalah membuat public/private key, SSL sertifikat, certificate signing request (CSR).

Biasanya, server yang komersial, akan meminta otoritas pihak ke tiga seperti VeriSign untuk menandatangani sertifikatnya. Pada praktikum ini, kita tidak akan meminta bantuan pihak ketiga, namun menjadi CA sendiri (self signing CA). Kita juga akan membuat domain sendiri yang dihost secara local. Karena itu anda bisa memilih nama domain yang anda inginkan. Kerjakan langkah-langkah dibawah dan tuliskan pada laporan praktikum

1. Asumsikan Topologi pada Gambar 7.3. sudah dibuat, topologi bisa dibuat melalui topologi real maupun simulasi menggunakan virtual box
2. Login ke sistem Ubuntu Server anda, dengan mengetikkan user dan password kemudian masuk ke Root Mode
3. Update repository software
4. Lakukan instalasi LAMP Server (apache2, mysql-server, PHP5, PhpMyAdmin)
5. Instalasi packet software SSL dengan perintah **`apt-get install openssl ssl-cert`**

```
root@ubuntuserver:/# apt-get install openssl ssl-cert
Reading package lists... Done
Building dependency tree
Reading state information... Done
ssl-cert is already the newest version.
ssl-cert set to manually installed.
The following packages will be upgraded:
  openssl
1 upgraded, 0 newly installed, 0 to remove and 199 not upgraded.
Need to get 479 kB of archives.
After this operation, 1,024 B of additional disk space will be used.
Do you want to continue? [Y/n] _
```

Gambar 7.4. Instalasi OpenSSL

```
Do you want to continue? [Y/n]
Get:1 http://kambing.ui.ac.id/ubuntu/ trusty-updates/main openssl i386 1.0.1f-1u
buntu2.16 [479 kB]
Fetched 479 kB in 15s (31.8 kB/s)
(Reading database ... 58965 files and directories currently installed.)
Preparing to unpack .../openssl_1.0.1f-1ubuntu2.16_i386.deb ...
Unpacking openssl (1.0.1f-1ubuntu2.16) over (1.0.1f-1ubuntu2) ...
Processing triggers for man-db (2.6.7.1-1) ...
Setting up openssl (1.0.1f-1ubuntu2.16) ...
root@ubuntuserver:/# _
```

Gambar 7.5. Proses Instalasi Open SSL

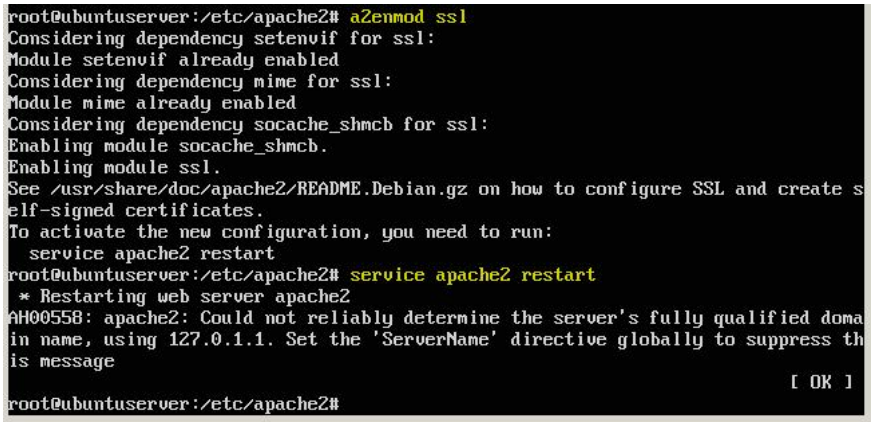
6. **Membuat Folder tempat sertifikat SSL ditempatkan**

Gambar 7.6. Cek Lokasi Folder SSL

Cek folder SSL di direktori `/etc/apache2`, jika belum ada, buat folder dengan nama ssl untuk menyimpan sertifikat SSL. Untuk keseragaman praktikum, tempatkan pada folder "/etc/apache2/ssl" . Jika belum ada, buat dulu foldernya : `# mkdir /etc/apache2/ssl`

Gambar 7.7. Buat Folder SSL

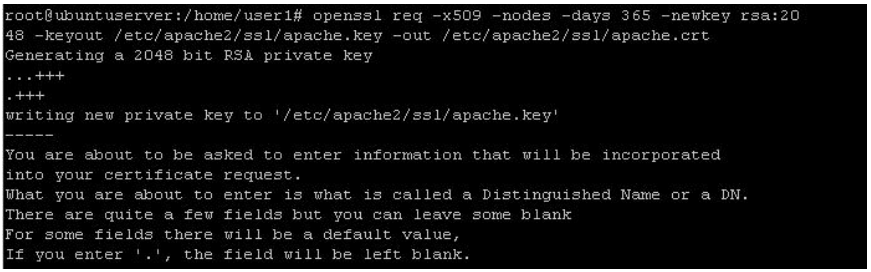
7. Aktifkan mod SSL dan Restart service apache2



Gambar 7.8. Mengaktifkan Mod SSL

8. Request sertifikat SSL, letakkan pada folder /etc/apache2/ssl yang sudah dibuat sebelumnya, untuk request sertifikat SSL tersebut ketikkan script dibawah ini :

```
openssl req -x509 -nodes -days 365 -newkey
rsa:2048 -keyout /etc/apache2/ssl/apache.key -out
/etc/apache2/ssl/apache.crt
```



Gambar 7.9.Proses Pembuatan SSL CERT

Apabila sertifikat sukses dibuat, akan muncul beberapa form yang harus diisi oleh user/root, isikan form tersebut sebagai berikut :

Gambar 7.10. Proses Pengisian Data SSL-CERT

Cek lokasi sertifikat tersebut, apakah sudah berada di direktori /etc/apache2/ssl

Gambar 7.11. Cek Hasil Request Sertifikat

Jika belum ada, lakukan proses request ulang terhadap sertiikat tersebut, atau lakukan restart pada service apache2. Jika sudah, lanjut ke langkah 9

9. Konfigurasi apache2 untuk menggunakan SSL secara deault

Untuk menghindari kesalahan konfigurasi, ada baiknya file default-ssl.conf dibackup terlebih dahulu seperti pada gambar 7.12 :

```
root@ubuntuserver:/etc/apache2# cd sites-available/
root@ubuntuserver:/etc/apache2/sites-available# ls
000-default.conf  default-ssl.conf
root@ubuntuserver:/etc/apache2/sites-available# cp default-ssl.conf default-ssl-backup.conf
root@ubuntuserver:/etc/apache2/sites-available# ls
000-default.conf  default-ssl-backup.conf  default-ssl.conf
root@ubuntuserver:/etc/apache2/sites-available#
```

Gambar 7.12. Backup File default-ssl.conf

Jika file default-ssl.conf sudah dibackup, selanjutnya lakukan konfigurasi pada file default-ssl.conf yang asli melalui editor nano, dengan perintah **`nano default-ssl.conf`**

```
<IfModule mod_ssl.c>
    <VirtualHost _default_:443>
        ServerAdmin webmaster@localhost
        DocumentRoot /var/www/html
        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined
        SSLEngine on
        SSLCertificateFile /etc/ssl/certs/ssl-cert-snakeoil.pem
        SSLCertificateKeyFile /etc/ssl/private/ssl-cert-snakeoil.key
        <FilesMatch "\.(cgi|shtml|phtml|php)$">
                        SSLOptions +StdEnvVars
        </FilesMatch>
        <Directory /usr/lib/cgi-bin>
                        SSLOptions +StdEnvVars
        </Directory>
        BrowserMatch "MSIE [2-6]" \
                        nokeepalive ssl-unclean-shutdown \
                        downgrade-1.0 force-response-1.0
        BrowserMatch "MSIE [17-9]" ssl-unclean-shutdown
    </VirtualHost>
</IfModule>
```

Gambar 7.13. File Konfigurasi default-ssl.conf

Terdapat sederetan baris panjang konfigurasi SSL, namun jika komentar tanda # dihilangkan, akan tampak daftar perintah seperti pada Gambar 7.13. Pada Gambar 13 terdapat baris yang di highlight dengan warna kuning, baris tersebut menunjukkan lokasi penyimpanan sertifikat default yang nantinya akan disesuaikan dengan sertifikat yang sudah dibuat pada langkah nomor 7 dan 8.

```
<IfModule mod_ssl.c>
    <VirtualHost _default_:443>
        ServerAdmin admin@example.com
        ServerName your_domain.com
        ServerAlias www.your_domain.com
        DocumentRoot /var/www/html
        ErrorLog ${APACHE_LOG_DIR}/error.log
        CustomLog ${APACHE_LOG_DIR}/access.log combined
        SSLEngine on
        SSLCertificateFile /etc/apache2/ssl/apache.crt
        SSLCertificateKeyFile /etc/apache2/ssl/apache.key
        <FilesMatch "\.(cgi|shtml|phtml|php)$">
                        SSLOptions +StdEnvVars
        </FilesMatch>
        <Directory /usr/lib/cgi-bin>
                        SSLOptions +StdEnvVars
        </Directory>
        BrowserMatch "MSIE [2-6]" \
                        nokeepalive ssl-unclean-shutdown \
                        downgrade-1.0 force-response-1.0
        BrowserMatch "MSIE [17-9]" ssl-unclean-shutdown
    </VirtualHost>
</IfModule>
```

Gambar 7.14. Konfigurasi Virtual Host SSL

Ubahlah konfigurasi pada Gambar 13 sesuaikan dengan konfigurasi pada Gambar 14.

10. Aktifkan virtualhost untuk SSL
    Aktifkan virtualhost untuk direktori SSL dengan perintah **`a2ensite default-ssl.conf`** seperti pada gambar 7.15

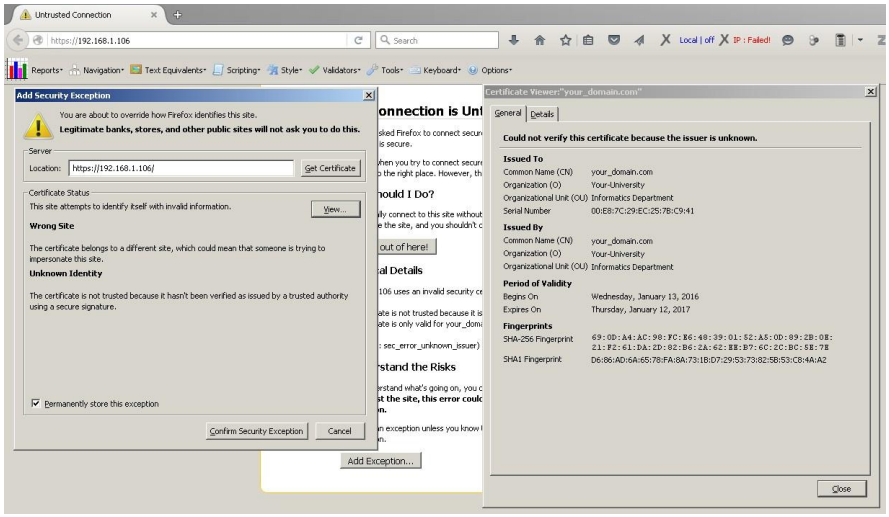
```
root@ubuntuserver:/etc/apache2/sites-available# a2ensite default-ssl.conf
Enabling site default-ssl.
To activate the new configuration, you need to run:
  service apache2 reload
root@ubuntuserver:/etc/apache2/sites-available# service apache2 reload
 * Reloading web server apache2
 *
root@ubuntuserver:/etc/apache2/sites-available#
```

Gambar 7.15. Aktifasi Virtualhost untuk SSL

Jika virtualhost sudah aktif, restart/reload service apache2 dengan perintah **`service apache2 reload`** seperti pada gambar 7.15

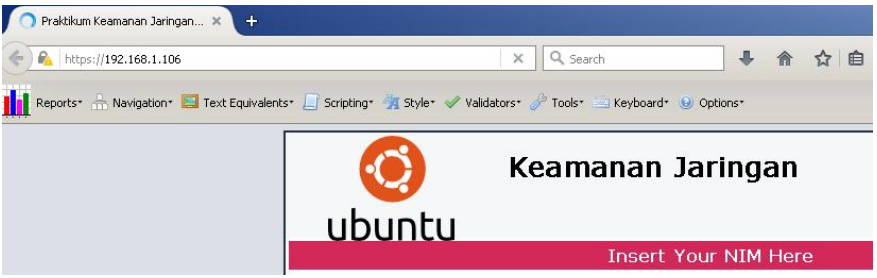
11. Tes Webserver Melalui Protocol HTTPS

    Untuk memastikan apakah sertifikat telah terinstall pada webserver, cek melalui browser firefox seperti pada Gambar 7.16



Gambar 7.16. Menampilkan Sertifikat SSL

Cara untuk menampilkan sertifikat tersebut adalah dengan mengetikkan *protocol* https diikuti dengan alamat IP *webserver* pada *address* bar Mozilla firefox **(https://ip-anda),** apabila muncul tab menu *add security exception*, user bisa langsung memilih *confirm security exception* maupun *get certificate* pada menu, namun bila user ingin melihat apakah sertifikat yang dibuat sesuai dengan langkah nomor 8, user bisa memilih menu *view certificate* dan akan menampilkan kotak dialog seperti menu sebelah kanan pada Gambar 7.15. Di Gambar 7.15 terlihat bahwa *publisher* dari sertifikat tersebut adalah *Informatics Department*, sesuai pada langkah nomor 8.



Gambar 7.17. Halaman Web HTTPS

Gambar 7.17 menampilkan website dengan protocol HTTPS bila pada proses sebelumnya memilih *get-certificate.*

7.4. TUGAS

1. Lakukan langkah - langkah praktikum dari nomor 1 - 17 dan catat kegiatan anda di file dokumen word
2. Jelaskan maksud dari perintah berikut :

```
openssl req -x509 -nodes -days 365 -newkey
rsa:2048 -keyout /etc/apache2/ssl/apache.key -out
/etc/apache2/ssl/apache.crt
```

3. Ulangi langkah-langkah praktikum dari nomor 1-17 dengan menambahkan penerapan DNS Server. Web server yang anda tampilkan harus bisa diakses dan menunjukkan alamat sebagai berikut https://nama-anda.com
4. Kumpulkan pada pertemuan berikutnya

8

PENGENALAN PORTSENTRY UNTUK MENCEGAH *NETWORK SCANNING*

ALOKASI WAKTU

Praktikum ini membutuhkan waktu 90 menit dan terdiri dari 3 percobaan

8.1. TUJUAN

1. Pengenalan konsep *intrusion detection* menggunakan portsentry
2. Mahasiswa memahami cara mengkonfigurasi portsentry
3. Mahasiswa mampu mempertahankan server dari percobaan *scanning*

8.2. PENDAHULUAN

Salah satu ancaman yang ditakuti sebagian besar pengguna *internet* ketika sistemnya terkoneksi dengan jaringan *internet* adalah serangan *hacker*. Penggunaan software seperti *firewall* dan *access control list* cukup membantu mengatasi serangan hacker. Namun pada kenyataannya, mengatasi serangan saja tidak cukup, administrator jaringan harus dapat mendeteksi serangan bahkan sebelum serangan tersebut dilakukan oleh hacker atau ketika serangan masih dalam tahap pengumpulan informasi. Proses seperti ini biasa disebut dengan istilah *Intrusion Detection* (IDS).

Dalam praktikum ke-8 ini akan diperkenalkan sebuah perangkat lunak IDS yang bernama portsentry. PortSentry adalah sebuah perangkat lunak yang di rancang untuk mendeteksi adanya *port scanning* & meresponds secara aktif jika ada *port scanning*. *Port scan* adalah proses scanning berbagai aplikasi servis yang dijalankan di *server Internet*. *Port scan* adalah langkah paling awal sebelum serangan di lakukan pada sebuah sistem.

8.3. DASAR TEORI

Portsentry

Cara kerja port sentry dengan melihat komputer yang melakukan scanning dan secara aktif akan memblokir mesin penyerang agar tidak dapat masuk atau tidak mendapatkan informasi dari server yang di-scan. PortSentry dapat langsung diinstal pada sistem operasi berbasis debian seperti ubuntu dan debian server dengan perintah 'apt-get'. Beberapa kemampuan yang dimiliki PortSentry:

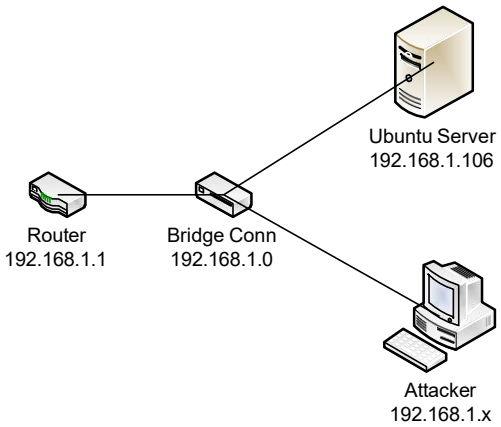
- Berjalan di atas soket TCP & UDP untuk mendeteksi scan port.
- Mendeteksi stealth scan, seperti SYN/half-open, FIN, NULL, X-MAS.
- PortSentry akan bereaksi secara real-time (langsung) dengan cara memblokir IP address si penyerang. Hal ini dilakukan dengan menggunakan ipchains/ipfwadm dan memasukan ke file /etc/host.deny secara otomatis oleh TCP Wrapper.
- PortSentry mempunyai mekanisme untuk mengingat mesin / host mana yang pernah terkoneksi dengannya. Dengan cara itu, hanya mesin / host yang terlalu sering melakukan sambungan (karena melakukan scanning) yang akan di blokir.
- PortSentry akan melaporkan semua pelanggaran melalui syslog dan mengindikasikan nama system, waktu serangan, IP mesin penyerang, TCP / UDP port tempat serangan dilakukan. Jika hal ini di integrasikan dengan Logcheck maka administrator system akan memperoleh laporan melalui e-mail.

Dengan adanya berbagai fitur tersebut, sistem yang di-*manage* akan cukup merepotkan attacker. Penggunaan PortSentry sangat mudah, konfigurasi defaultnya dapat langsung digunakan. Beberapa hal yang perlu diperhatikan pada saat konigurasi portsentry adalah file konfigurasi portsentry yang semuanya berlokasi di **`/etc/portsentry`**. Untuk mengedit file konfigurasi tersebut user membutuhkan privilige sebagai root. Beberapa hal yang perlu di konfigurasi adalah :

- File /etc/portsentry/portsentry.conf merupakan konfigurasi utama portsentry. Disini secara bertahap diset port mana saja yang perlu di monitor, responds apa yang harus di lakukan ke mesin yang melakukan portscan. Proses *setting* sangat mudah hanya dengan membuka / menutup tanda pagar (#).
- Pada file /etc/portsentry/always_ignore mendata semua IP address di LAN yang harus selalu di abaikan oleh portsentry. Artinya memasukan IP address ke sini, agar tidak terblokir secara tidak sengaja.
- Pada file /etc/portsentry/portsentry.ignore isikan IP address yang perlu di abaikan sama dengan isi file /etc/ portsentry/always_ignore.
- Pada file /etc/portsentry.modes user dapat menset mode deteksi yang dilakukan portsentry. Semakin baik mode deteksi yang dipilih (advanced stealth TCP/UP scanning), biasanya PortSentry akan semakin sensitive.

8.4. LANGKAH PRAKTIKUM

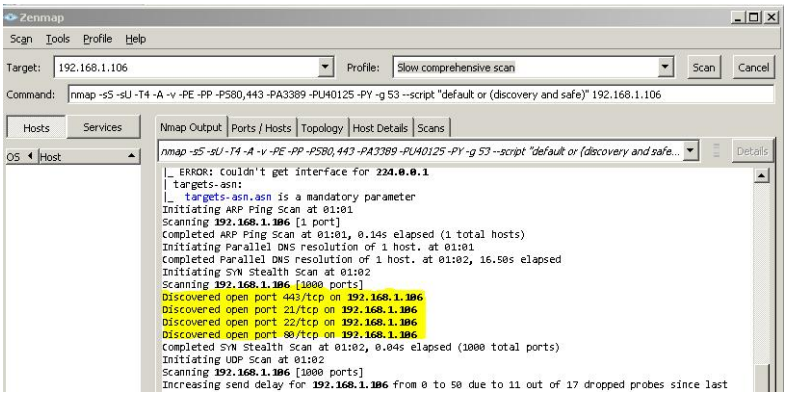
1. Persiapan semua kebutuhan praktek :
   - Nmap/Nmap-GUI/Zenmap diinstal pada windows PC
   - Windows PC sebagai *attacker* dengan IP 192.168.1.2-254
   - Ubuntu server sebagai *host* target dengan IP 192.168.1.106
   - Portsentry diinstal pada Ubuntu Server
   - Topologi jaringan



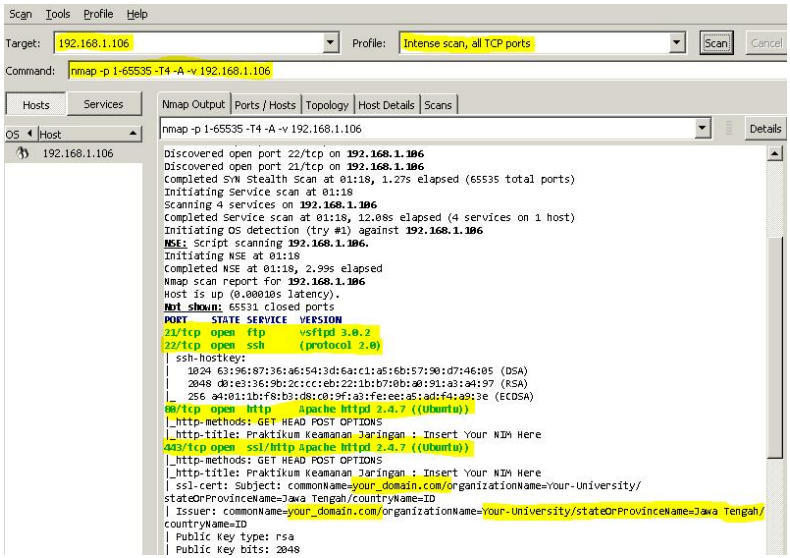
Gambar 8.1. Topologi Praktek Modul 8

2. Scanning

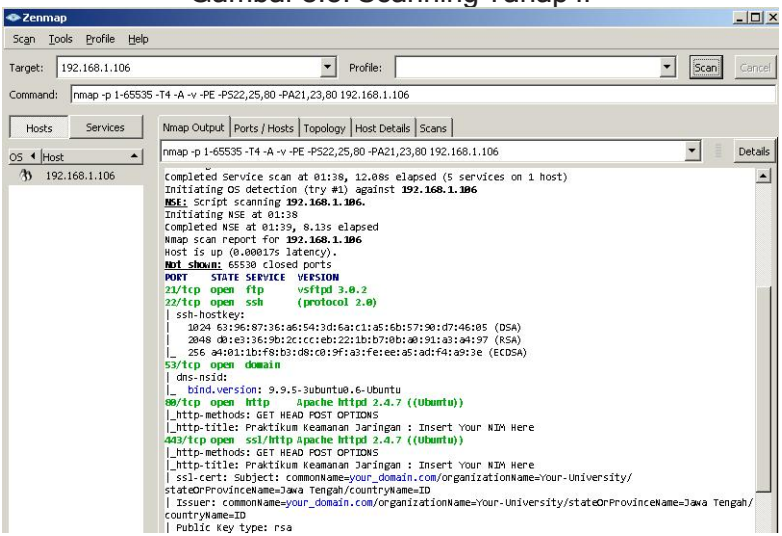
Attacker melakukan scanning menggunakan Nmap/Nmap-GUI/ Zenmap terhadap Server Ubuntu. Lakukan proses Scanning dengan berbagai macam mode scanning yang ada. Dari proses scanning tersebut bisa diketahui server ubuntu memiliki beberapa port yang terbuka. Lakukan scanning menggunakan PC client dengan berbagai mode scanning seperti pada gambar 8.2.sampai pada gambar 8.4



Gambar 8.2. Scanning Tahap 1



Gambar 8.3. Scanning Tahap II



Gambar 8.4. Scanning Tahap III

Dari scanning pada langkah di gambar 8.2 sampai dengan gambar 8.4 menunjukkan terdapat beberapa port pada ubuntu yang terbuka, antara lain port 21,22,50,80, dan 443. Pada langkah selanjutnya akan dikonfigurasikan software portsentry untuk memblokir proses scanning pada port-port yang terbuka.

3. Instalasi Portsentry

   Lakukan proses instalasi dengan mengetikkan perintah **`apt-get install portsentry`** seperti pada gambar 8.5

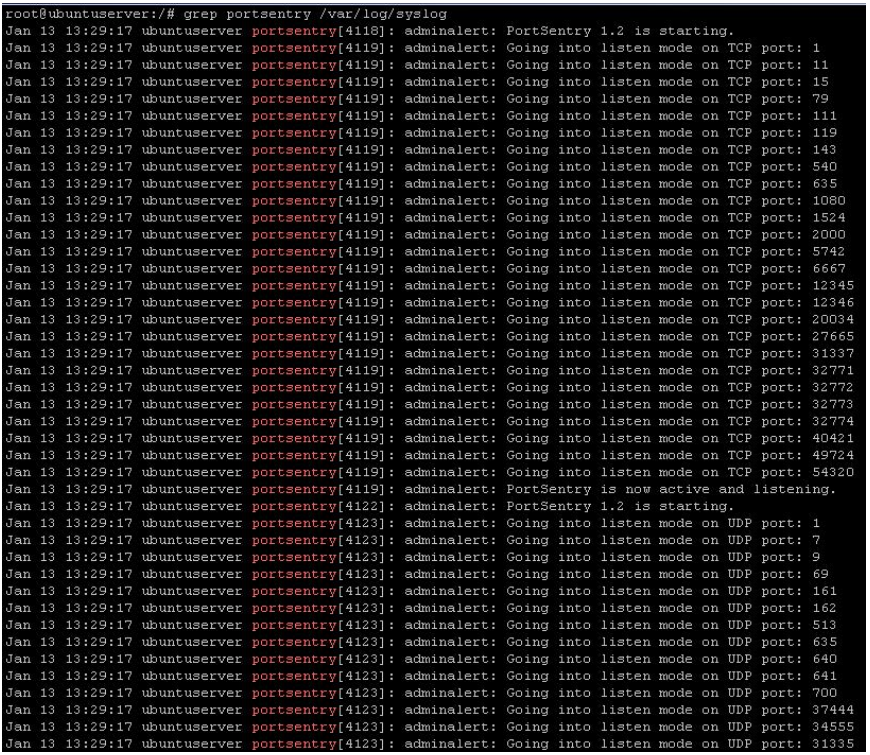
```
root@ubuntuserver:/# apt-cache search portsentry
portsentry - Portscan detection daemon
prelude-lml - Security Information Management System [ Log Agent ]
root@ubuntuserver:/# apt-get install portsentry
Reading package lists... Done
Building dependency tree
Reading state information... Done
Suggested packages:
  logcheck
The following NEW packages will be installed:
  portsentry
0 upgraded, 1 newly installed, 0 to remove and 189 not upgraded.
Need to get 69.3 kB of archives.
After this operation, 229 kB of additional disk space will be used.
Get:1 http://kambing.ui.ac.id/ubuntu/ trusty/universe portsentry i386 1.2-13 [69.3 kB]
Fetched 69.3 kB in 1s (50.4 kB/s)
Preconfiguring packages ...
Selecting previously unselected package portsentry.
(Reading database ... 59069 files and directories currently installed.)
Preparing to unpack .../portsentry_1.2-13_i386.deb ...
Unpacking portsentry (1.2-13) ...
Processing triggers for man-db (2.6.7.1-1) ...
Processing triggers for ureadahead (0.100.0-16) ...
Setting up portsentry (1.2-13) ...
Starting anti portscan daemon: portsentry in tcp & udp mode.
Processing triggers for ureadahead (0.100.0-16) ...
```

Gambar 8.5. Proses Instalasi Portsentry

```
root@ubuntuserver:/# ls -l /etc/portsentry/
total 20
-rw-r--r-- 1 root root 11681 May  1  2012 portsentry.conf
-rw-r--r-- 1 root root   491 Jan 13 13:29 portsentry.ignore
-rw-r--r-- 1 root root   699 May  1  2012 portsentry.ignore.static
```

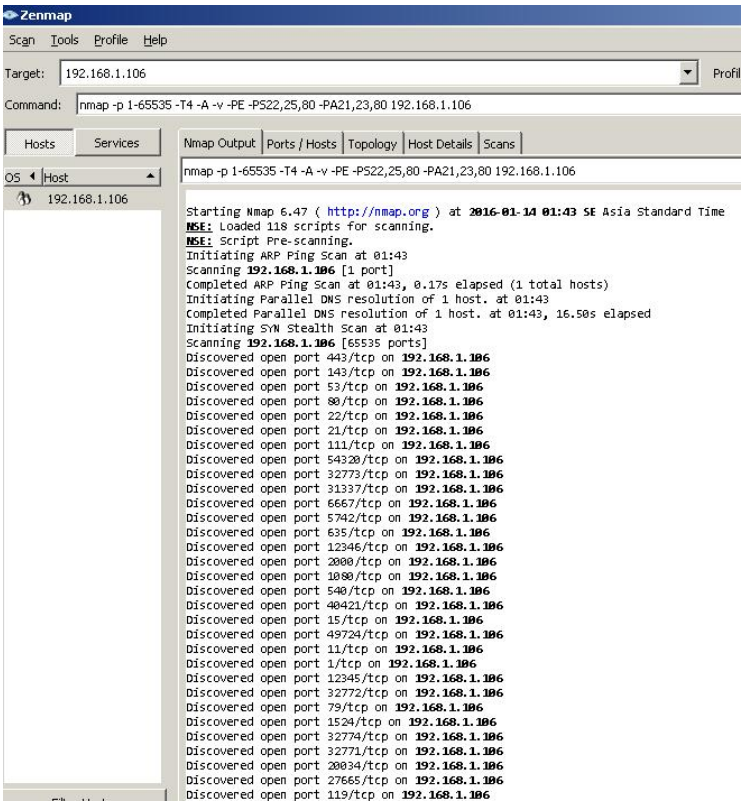
Gambar 8.6. Daftar File Konfigurasi Standar

Setelah proses instalasi selesai, seharusnya ada 3 file konfigurasi dasar seperti pada gambar 8.6, ketiga file tersebut memiliki permission root. Selanjutnya cek apakah portsentry bisa running di file log pencatatan sistem dengan mengetikkan perintah **`grep portsentry /var/log/syslog`** seperti pada gambar 8.7 dibawah ini :



Gambar 8.7. Portsentry Log Sistem

Pada tahap ini, portsentry sudah aktif secara default dan sudah bisa digunakan untuk mendeteksi port scanning. Selain itu portsentry akan menciptakan beberapa port tipuan seperti ditunjukkan pada gambar 8.9. port-port tersebut biasa digunakan untuk menipu attacker supaya attacker mengira terdapat port vulnerable yang bisa dieksploitasi. Selanjutnya lakukan scan ulang pada server. Pada gambar 8.9. menunjukkan terdapat banyak port yang muncul tiba-tiba, ditunjukkan dengan warna hijau.port-port tersebut tidak diketahui jenis servicenya



Gambar 8.8. Proses Scan Ulang pada Server

```
Not shown: 65504 closed ports
PORT      STATE SERVICE    VERSION
1/tcp     open  tcpwrapped
11/tcp    open  tcpwrapped
15/tcp    open  tcpwrapped
21/tcp    open  ftp        vsftpd 3.0.2
22/tcp    open  ssh        (protocol 2.0)
| ssh-hostkey:
|   1024 63:96:87:36:a6:54:3d:6a:c1:a5:6b:57:90:d7:46:05 (DSA)
|   2048 d0:e3:36:9b:2c:cc:eb:22:1b:b7:0b:a0:91:a3:a4:97 (RSA)
|_  256 a4:01:1b:f8:b3:d8:c0:9f:a3:fe:ee:a5:ad:f4:a9:3e (ECDSA)
53/tcp    open  domain
| dns-nsid:
|_  bind.version: 9.9.5-3ubuntu0.6-Ubuntu
79/tcp    open  tcpwrapped
|_finger: ERROR: Script execution failed (use -d to debug)
80/tcp    open  http       Apache httpd 2.4.7 ((Ubuntu))
|_http-methods: GET HEAD POST OPTIONS
|_http-title: Praktikum Keamanan Jaringan : Insert Your NIM Here
111/tcp   open  tcpwrapped
119/tcp   open  tcpwrapped
143/tcp   open  tcpwrapped
| imap-capabilities:
|_  ERROR: Failed to connect to server
443/tcp   open  ssl/http   Apache httpd 2.4.7 ((Ubuntu))
|_http-methods: GET HEAD POST OPTIONS
|_http-title: Praktikum Keamanan Jaringan : Insert Your NIM Here
| ssl-cert: Subject: commonName=your_domain.com/organizationName=Your-University/stateOrPro
| Issuer: commonName=your_domain.com/organizationName=Your-University/stateOrProvinceName=J
| Public Key type: rsa
| Public Key bits: 2048
| Not valid before: 2016-01-13T15:07:24+00:00
| Not valid after:  2017-01-12T15:07:24+00:00
| MD5:   8b93 f0c2 be4f 79d2 665f 2829 a156 c30d
|_SHA-1: d686 ad6a 6578 fa8a 731b d729 5373 825b 53c8 4aa2
|_ssl-date: 2096-05-28T17:33:57+00:00; +80y135d22h49m57s from local time.
540/tcp   open  tcpwrapped
635/tcp   open  tcpwrapped
1080/tcp  open  tcpwrapped
1524/tcp  open  tcpwrapped
2000/tcp  open  tcpwrapped
5742/tcp  open  tcpwrapped
```

Gambar 8.9. Port TCPwrapped

```
root@ubuntuserver:/# grep attackalert /var/log/syslog
root@ubuntuserver:/# grep attackalert /var/log/syslog
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to
 TCP port: 1
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Ignoring TCP response per configuration file sett
ing.
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to
 TCP port: 11
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to
 TCP port: 15
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to
 TCP port: 79
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to
 TCP port: 111
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to
 TCP port: 119
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to
 TCP port: 143
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to
 TCP port: 540
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to
 TCP port: 635
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to
 TCP port: 1080
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to
 TCP port: 1524
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to
 TCP port: 2000
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to
 TCP port: 5742
```

Gambar 8.10. Portsentry Mendeteksi Serangan

```
root@ubuntuserver:/# grep attackalert /var/log/syslog
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 1
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Ignoring TCP response per configuration file setting.
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 11
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 15
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 79
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 111
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 119
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 143
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 540
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 635
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 1080
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 1524
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 2000
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 5742
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 6667
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 12345
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 12346
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Host: 192.168.1.104 is already blocked. Ignoring
Jan 13 13:43:41 ubuntuserver portsentry[4190]: attackalert: Connect from host: 192.168.1.104/192.168.1.104 to TCP port: 20034
```

Gambar 8.11. Portsentry Mendeteksi Serangan II

Pada saat PC client melakukan scanning menggunakan NMAP, secara bersamaan lakukan pengecekan pada log server yang dipantau oleh portsentry dengan mengetikkan **`grep`**

attackalert /var/log/syslog seperti pada gambar 9.10 dan 9.11. Pada gambar 8.10 dan 8.11 tersebut menunjukkan keterangan bahwa, terdapat koneksi yang berasal dari host dengan alamat IP 192.168.1.104 melalui berbagai TCP port (1,11,15,79,111,119,143,540, dst) dari gambar 8.11 terdapat keterangan bahwa koneksi scanning dari IP 192.168.1.104 mendapat respon ignoring oleh server pada tahap SYN scan, dan pada tahap ACK scan koneksi dari IP 192.168.1.104 mendapat respon "already blocked". Pemblokiran pada tahap tersebut baru pemblokiran terhadap scan port, server belum melakukan pemblokiran terhadap koneksi TCP dan UDP yang establish dari IP 192.168.1.104. untuk memblokir koneksi dari TCP dan UDP tersebut dilakukan dengan cara masuk ke file file **/etc/portsentry/portsentry.conf** menggunakan editor nano, kemudian cari baris yang menunjukkan **BLOCK_TCP="0"** dan **BLOCK_UDP="0"**

```
#
# 0 = Do not block UDP/TCP scans.
# 1 = Block UDP/TCP scans.
# 2 = Run external command only (KILL_RUN_CMD)

BLOCK_UDP="1"
BLOCK_TCP="1"

####################
# Dropping Routes:#
```

Gambar 8.12. Block TCP dan UDP

Simpan konfigurasi dan restart service portsentry dengan perintah **service portsentry restart**, lakukan scanning kembali terhadap server menggunakan nmap dan lakukan remote connection dari PC client menggunakan putty ke alamat IP server, apakah user yang melakukan scanning dan berusaha login tadi dapat melakukan scanning dan login ke server ubuntu? tuliskan kesimpulan anda pada laporan praktikum

8.5. TUGAS MODUL 8

1. Berikan kesimpulan hasil praktikum yang anda lakukan.
2. Dengan bekerja hanya mendeteksi port scanning dimana sebaiknya portsentry ditempatkan pada topologi jaringan yang ada ?
3. Kumpulkan laporan praktikum pada pertemuan berikutnya atau melalui email ke asisten praktikum

9

ANTIVIRUS PADA GNU/LINUX SERVER

9.1. TUJUAN

Mahasiswa mampu mengkonfigurasi antivirus pada server linux

9.2. PENDAHULUAN

Seringkali orang menganggap Linux merupakan sistem operasi yang kebal malware. Anggapan tersebut kurang tepat. Sebenarnya tidak ada sebuah sistem operasi yang kebal dari malware. Malware untuk linux memang tidak sebanyak malware pada windows. Hal ini karena memang dibandingkan windows, pengguna linux itu tidak banyak. Sehingga target utama pembuat virus adalah OS windows.

Malware yang berjalan di windows berbeda dengan malware pada sistem linux. Seringkali malware pada windows tidak berjalan pada linux. Antivirus linux dapat juga digunakan untuk mendeteksi malware pada windows. Misalnya pada media penyimpanan (FTP server, dll) yang terhubung pada jaringan. Atau bisa digunakan juga untuk melakukan scan pada komputer windows yang terhubung dengan jaringan.

Jadi antivirus untuk linux ini sudut pandangnya adalah mengamankan file-file yang biasa digunakan sebagai file sharing terhadap komputer client yang mana komputer client tersebut menggunakan sistem operasi yang rentan terhadap virus/malware

Beberapa antivirus yang terkenal dan cocok digunakan pada linux server

- ClamAV Antivirus
- Avast Antivirus Home Edition (Linux)
- AVG Antivirus
- Bit Defender (Linux Edition)
- F-PROT Antivirus

9.3. LANGKAH PRAKTIKUM

1. Update repository paket software pada server ubuntu dengan perintah **`apt-get update.`**

   Langkah ini diperlukan, karena terkadang repository untuk beberapa antivirus tidak tersedia jika sistem tidak diupdate terlebih dahulu, atau paket software yang tersedia adalah versi lama. Tentunya pengguna tidak ingin antivirus yang diinstal bukan versi yang up to date.

```
[sudo] password for user1:
root@ubuntuserver:/home/user1# apt-get update
Ign http://kambing.ui.ac.id trusty InRelease
Hit http://kambing.ui.ac.id trusty-updates InRelease
Hit http://kambing.ui.ac.id trusty-security InRelease
Hit http://kambing.ui.ac.id trusty-backports InRelease
```

Gambar 9.1. Update Paket Software

2. Cek ketersediaan dan kompatibilitas software antivirus clamav untuk server ubuntu dengan perintah **`apt-cache search clamav`**

   Langkah ini diperlukan untuk mengetahui apakah antivirus clamav akan kompatibel dengan sistem ubuntu server yang digunakan untuk praktikum.

```
root@ubuntuserver:/home/user1# apt-cache search clamav
amavisd-new - Interface between MTA and virus scanner/content filters
clamav - anti-virus utility for Unix - command-line interface
clamav-base - anti-virus utility for Unix - base package
clamav-daemon - anti-virus utility for Unix - scanner daemon
clamav-dbg - debug symbols for ClamAV
clamav-docs - anti-virus utility for Unix - documentation
clamav-freshclam - anti-virus utility for Unix - virus database update utility
libclamav-dev - anti-virus utility for Unix - development files
libclamav6 - anti-virus utility for Unix - library
libclamunrar6 - anti-virus utility for Unix - unrar support
amavisd-new-postfix - part of Ubuntu mail stack provided by Ubuntu server team
clamassassin - email virus filter wrapper for ClamAV
clamav-milter - anti-virus utility for Unix - sendmail integration
clamav-testfiles - anti-virus utility for Unix - test files
clamav-unofficial-sigs - update script for 3rd-party clamav signatures
clamfs - user-space anti-virus protected file system
clamsmtp - virus-scanning SMTP proxy
clamtk - graphical front-end for ClamAV
claws-mail-clamd-plugin - ClamAV socket-based plugin for Claws Mail
courier-filter-perl - purely Perl-based mail filter framework for the Courier MTA
havp - HTTP Anti Virus Proxy
libc-icap-mod-clamav - transitional dummy package
libc-icap-mod-virus-scan - Antivirus Service for c-icap
libclamav-client-perl - Perl client for the ClamAV virus scanner daemon
nagios-plugins-contrib - Plugins for nagios compatible monitoring systems
proftpd-mod-clamav - ProFTPD module mod_clamav
python-clamav - Python bindings to ClamAV - transitional package
python-pyclamav - Python bindings to ClamAV
python-pyclamd - Python interface to the ClamAV daemon
clamtk-nautilus - Nautilus MenuProvider extension for clamtk
python3-pyclamd - Python 3 interface to the ClamAV daemon
```

Gambar 9.2. Cek Kompatibilitas Antivirus

3. Instal Clamav dengan perintah **`apt-get -y install clamav`**

```
root@ubuntuserver:/home/user1# apt-get -y install clamav
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  clamav-base clamav-freshclam libclamav6 libltdl7
Suggested packages:
  clamav-docs libclamunrar6
The following NEW packages will be installed:
  clamav clamav-base clamav-freshclam libclamav6 libltdl7
0 upgraded, 5 newly installed, 0 to remove and 201 not upgraded.
Need to get 3,568 kB of archives.
After this operation, 13.7 MB of additional disk space will be used.
```

Gambar 9.3. Proses Instalasi Clamav

4. Uji coba scan sistem server dengan perintah **clamscan** Gambar 9.4. dibawah ini menunjukkan proses scanning menggunakan engine clamscan

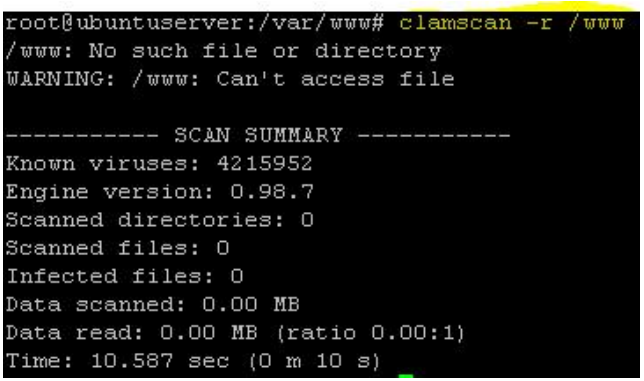
```
root@ubuntuserver:/home/user1# clamscan
/home/user1/.profile: OK
/home/user1/.bash_history: OK
/home/user1/.bash_logout: OK
/home/user1/.bashrc: OK

----------- SCAN SUMMARY -----------
Known viruses: 4215952
Engine version: 0.98.7
Scanned directories: 1
Scanned files: 4
Infected files: 0
Data scanned: 0.00 MB
Data read: 0.00 MB (ratio 0.00:1)
Time: 10.563 sec (0 m 10 s)
```

Gambar 9.4. Proses Scanning Menggunakan Clamscan Engine

Pada gambar menunjukkan bahwa clamav melakukan *scanning* terhadap virus di direktori /home/user1/ beserta file yang berada dalam direktori tersebut. Selain itu juga ditunjukkan bahwa clamav dapat mengenali 4215952 jenis virus/malware yang dapat menginfeksi sistem computer. Pada proses scanning tersebut tidak ditemukan virus dalam direktori /home/user1/.

5. Uji Coba Scan pada Direktori Lain

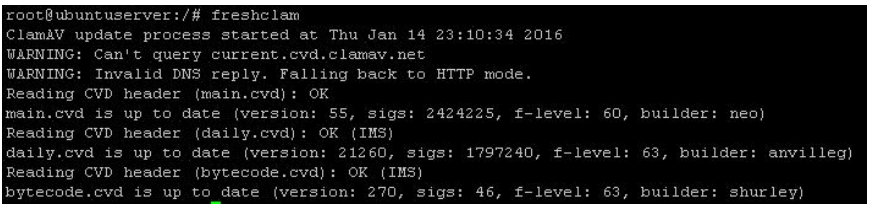
   Clamscan bisa dilakukan untuk melakukan scanning terhadap direktori lain pada sistem, sebagai contoh akan dilakukan scanning terhadap direktori webserver yang biasa diletakkan pada direktori /var/www. Cara melakukan scanning pada direktori /var/www/ adalah dengan mengetikkan perintah **`clamscan -r /www`**



Gambar 9.5. Scaning pada Direktori Web Server /var/www/

6. Update antivirus clamav dengan perintah `freshclam`

   Clamav dapat diupdate menggunakan perintah **`freshclam`**



Gambar 9.6. Update Clamav

7. Download test virus

   Untuk menguji clamav diperlukan virus sungguhan, namun pada praktikum ini akan digunakan sebuah file test virus yang dapat didownload menggunakan perintah

```
wget http://www.eicar.org/download/eicar.com
```

Gambar 9.7. download Test Virus

8. Scan Test-Virus

   Langkah selanjutnya adalah melakukan scanning dan menghapus test-virus yang sudah didownload tersebut dengan perintah **`clamscan –infected –remove –recursive ./`**

9.4. TUGAS MODUL 9

1. Dengan langkah scanning yang sama, lakukan scanning menggunakan clamav untuk setiap direktori dibawah direktori utama (root), catat hasilnya
2. Apakah konfigurasi user permission berpengaruh pada saat melakukan scanning pada direktori root? Untuk mengujinya lakukan scanning pada lokasi ./ root
3. Cobalah install aplikasi antivirus lain selain clamav (minimal 2) dan lakukan proses scanning
4. Catat langkah-langkah kegiatan praktikum dari langkah 1-8 beserta jawaban nomor 1, 2, dan 3 pada tugas modul di dalam file dokumen, kumpulkan pada pertemuan berikutnya.

10

KONFIGURASI KEAMANAN DASAR SERVER LINUX (UBUNTU)

10.1. TUJUAN

Mahasiswa dapat melakukan beberapa konfigurasi dasar untuk keamanan Server Ubuntu

10.2. DASAR TEORI

Berikut ini beberapa ancaman yang bisa terjadi pasca instalasi server ubuntu sampai server tersebut terhubung ke Internet

1. IP Spoofing

   IP spoofing juga dikenal sebagai *Source Address Spoofing*. Yaitu pemalsuan IP *attacker* sehingga sasaran menganggap alamat IP *attacker* adalah alamat IP dari host di dalam *network* bukan dari luar *network*. IP spoofing yang terjadi dalam jaringan lokal dapat diatasi dengan beberapa konfigurasi dasar pada server ubuntu

2. Denial of Service

   Jenis serangan terhadap sistem komputer atau server di dalam jaringan internet dengan cara menghabiskan resource yang dimiliki oleh komputer tersebut sampai komputer tersebut tidak dapat menjalankan fungsinya dengan benar sehingga secara tidak langsung mencegah pengguna lain untuk memperoleh akses layanan dari komputer yang diserang Ţersebut. Dalam sebuah serangan Denial of Service, penyerang akan mencoba untuk mencegah akses seorang pengguna terhadap sistem atau jaringan dengan menggunakan beberapa cara, salah satunya dengan membanjiri jaringan dengan banyak request terhadap sebuah layanan jaringan yang disedakan oleh sebuah host sehingga request yang datang dari pengguna lain tidak dapat dilayani oleh layanan tersebut. Teknik ini disebut sebagai

request flooding. Request flooding biasa teradi melalui paket ICMP, teknik ini dapat diatasi dengan memblokir ICMP request pada server ubuntu

3. Brute Force Password Login pada Server

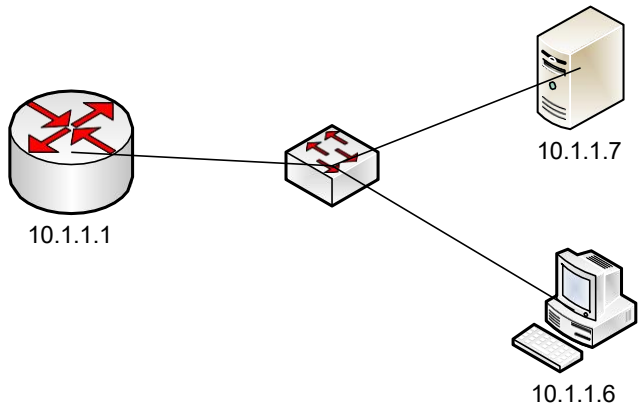
   Brute-force attack adalah sebuah teknik serangan terhadap sebuah sistem keamanan komputer yang menggunakan percobaan terhadap semua password yang mungkin digunakan. Serangan ini membutuhkan beberapa kali percobaan login, oleh karena itu serangan ini dapat diatasi dengan membatasi jumlah percobaan login yang dilakukan pada sistem.

4. Rootkit

   Rootkit adalah kumpulan software yang bertujuan untuk menyembunyikan proses, file dan data sistem yang sedang berjalan dari sebuah sistem operasi tempat dia bernaung. Rootkit awalnya berupa aplikasi yang tidak berbahaya, tetapi belakangan ini telah banyak digunakan oleh malware yang ditujukan untuk membantu penyusup menjaga aksi mereka yang ke dalam sistem agar tidak terdeteksi. rootkit hadir di beragam sistem operasi seperti, Linux, Solaris dan Microsoft Windows. Rootkit ini sering merubah bagian dari sistem operasi dan juga menginstall dirinya sendiri sebagai driver atau modul kernel.

10.3. LANGKAH PRAKTIKUM

Buat topologi jaringan seperti pada gambar berikut :



Gambar 10.1. Topologi Praktek Modul 10

Pada praktikum 10 ini akan dibuat topologi jaringan seperti pada gambar diatas. Pada komputer *server* 10.1.1.7 akan dilakukan beberapa konfigurasi keamanan yang meliputi pencegahan IP spoofing, pencegahan flooding pada ICMP yang berpotensi berkembang menjadi serangan DOS, menyamarkan port default untuk login SSH melalui remote client, pencegahan bruteforce terhadap password login, dan instalasi paket software pendeteksi rootkit dalam proses yang berjalan. Berikut ini langkah – langkah praktikum yang akan dilakukan :

1. Mencegah IP Spoofing

   IP Address Spoofing atau IP Spoofing merupakan salah satu cara menyerang dengan cara menyamarkan alamat IP Address. Berikut ini adalah cara menanggani hal tersebut dalam server ubuntu

   - Masuk ke file host.conf yang berada dalam direktori /etc/

Gambar 10.2. Konfigurasi Pencegahan IP Spoofing

   - Edit konfigurasi host.conf dengan mengganti parameter **no spoof off** menjadi **no spoof on.** Pada beberapa versi ubuntu server, tidak terdapat baris tersebut, parameter tersebut bisa ditambahkan sendiri

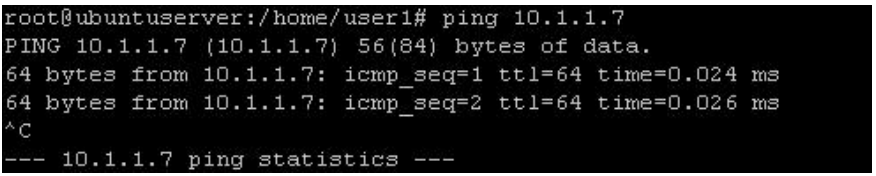
Gambar 10.3. Menambahkan Parameter Anti Spoofing

Simpan perubahan konfigurasi dengan CTRL+X, CTRL Y, <Enter>.

2. Mencegah Ping (ICMP request) pada Server

   ICMP request sebenarnya merupakan request paket data yang umum terjadi pada jaringan internet, namun ICMP request yang terlalu besar pada single system (bukan request broadcast) akan berpotensi menimbulkan denial of service yang dapat menghambat kinerja sistem baik dari segi performa sistem itu sendiri, maupun performa sistem itu dalam memberikan layanan kepada client. Pada tahap ke 2 akan dilakukan blokir terhadap ICMP request, namun sistem masih bisa memberikan layanan web server. Berikut ini langkah-langkah praktikum untuk mencegah ICMP request.

   - Kondisi awal, server ubuntu masih bisa menerima request ICMP



Gambar 10.4. Request ICMP pada Server

   - Edit konfigurasi sysctl dengan mengaktifkan fungsi icmp_echo_ignore_all dengan parameter 0 menjadi 1 untuk mencegah server menerima ICMP request yang berpotensi menjadi serangan DOS

Gambar 10.5. Konfigurasi icmp_ignore_all

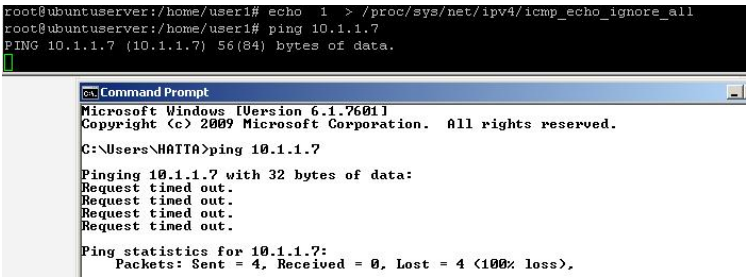
Untuk mengaktifkan icmp ignore dilakukan dengan cara mengetikkan perintah sebagai berikut :

```
echo  1  > /proc/sys/net/ipv4/icmp_echo_ignore_all
```

- Pada beberapa versi server ubuntu tidak tersedia fitur tersebut di file sysctl.conf, metode tersebut bisa digantikan dengan perintah pada IPtables sebagai berikut :

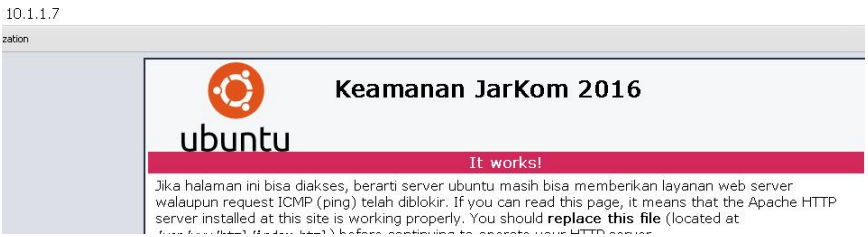
```
iptables  -I  INPUT  -i  ech0  -p   icmp  -s  0/0
-d  0/0   -j  DROP
```

- Tes ping dari remote client



Gambar 10.6. Percobaan Blokir PING

Komputer client sudah tidak bisa mengirim ICMP request ke server ubuntu, namun server tersebut masih dapat menyediakan layanan web server yang masih bisa diakses oleh client. Coba akses layanan web server melalui remote client dengan mengetikkan ip address server pada web browser seperti pada gambar dibawah ini :



Gambar 10.7. Tes Koneksi Melalui Web Server

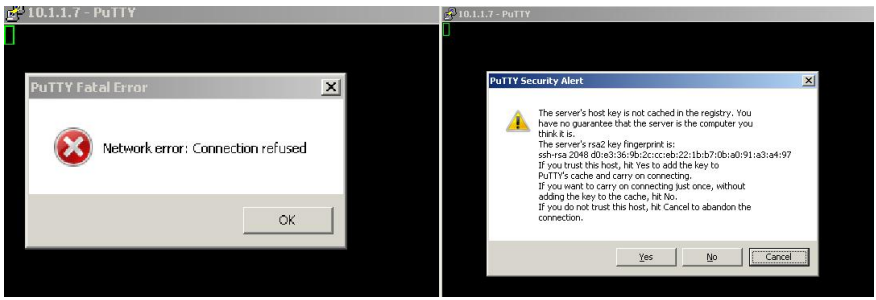
3. Optimasi Keamanan pada Port SSH di Server Ubuntu
   - Ubah nomor port untuk login ke server Ubuntu

     Buka file konfigurasi SSH di /etc/ssh/sshd_config dengan menggunakan editor NANO, masuk ke direktori tersebut dengan perintah **`nano /etc/ssh/sshd_config`**

Gambar 10.8. Konfigurasi SSHD Config

Setelah masuk kedalam file konfigurasi SSHD, cari port 22 lalu ubah angka 22 menjadi angka yang anda sukai dan mudah di ingat oleh anda. Misal ubah menjadi 5678, cari bari port 22 ganti menjadi 5678

Simpan konfigurasi SSHD yang baru dengan menekan CTRL+O, Keluar dari file konfigurasi SSHD dengan menekan CTRL + X, restart service sshd dengan mengetikkan perintah **restart ssh**. Coba login menggunakan putty pada port 22 dan pada port 5678, bandingkan hasilnya

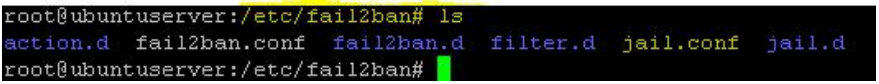


Gambar 10.9. Uji Coba Login Melalui Port SSH Default

4. Install fail2ban untuk Pencegahan Bruteforce Password
   - Install service fail2ban dengan perintah **apt-get install fail2ban**

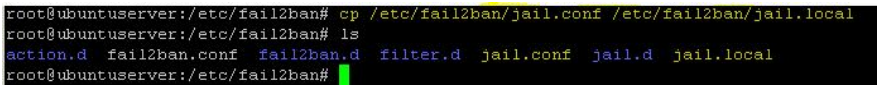
Gambar 10.10. Instalasi Fail2ban

   - Konfigurasi dasar fail2ban berada pada direktori /etc/ fail2ban yaitu pada file jail.conf



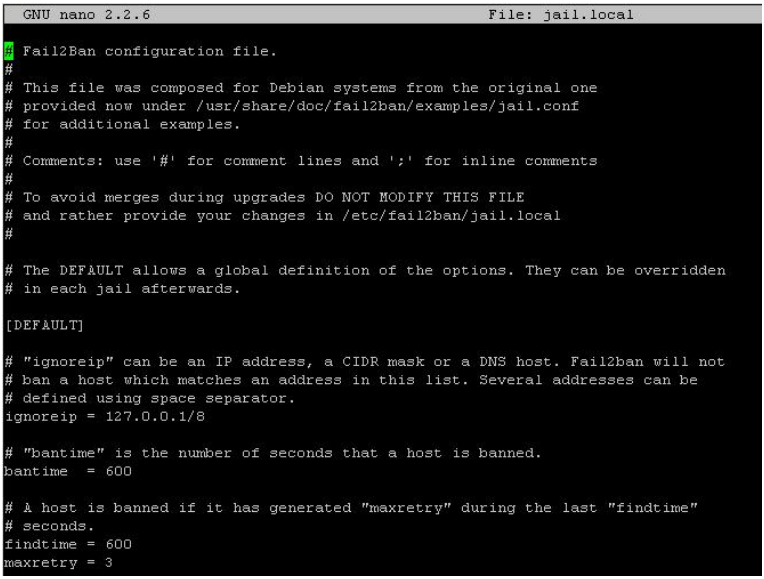
Gambar 10.11. Konfigurasi Dasar Fail2ban

   - Untuk menghindari kesalahan konfigurasi, dan proses overwrite yang tidak sengaja saat proses update, kita buat duplikat file jail.conf dan diberi nama jail.local. cara membuat copy file tersebut dengan perintah **sudo cp / etc/fail2ban/jail.conf /etc/fail2ban/jail.local**



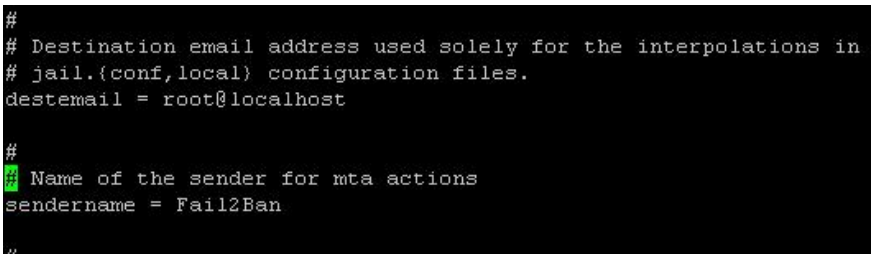
Gambar 10.12. Backup File Konfigurasi Fail2ban

   - Konfigurasi selanjutnya akan dilakukan pada file jail. local, untuk mengakses file tersebut dilakukan dengan mengetikkan **sudo nano /etc/fail2ban/jail.local**



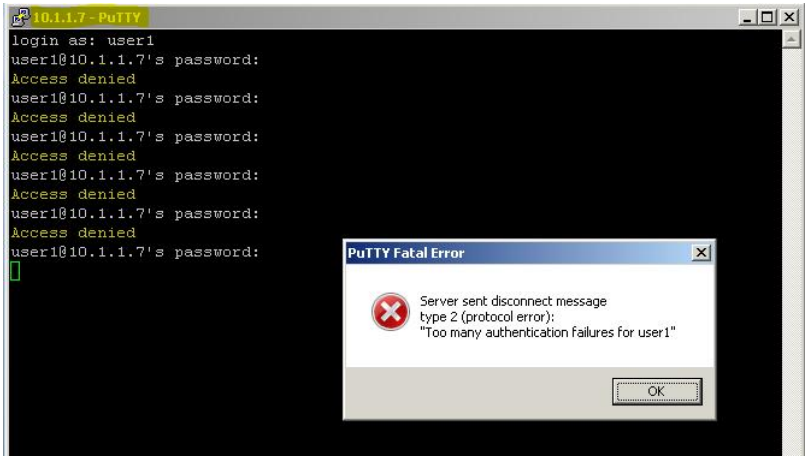
Gambar 10.13. Isi Konfigurasi File Jail.Local

- Ignoreip adalah bagian untuk memasukkan IP mana saja yang akan anda kecualikan (tidak bisa diblok), walaupun salah memasukkan password berkali-kali. Jika IP address yang ingin anda masukkan berjumlah lebih dari satu, silahkan pisahkan dengan spasi antara IP yang satu dengan yang lainnya.
- Bandtime adalah jumlah waktu sebuah IP akan diblokir dalam satuan detik. Biasanya secara default di bagian ini tertulis angka 600 yang berarti 10 menit. Silahkan anda sesuaikan dengan keinginan anda, jika angka 600 dirasa kurang tepat buat anda.
- Maxretry adalah jumlah percobaan yang dapat dilakukan oleh sebuah IP sebelum IP tersebut di blok. Jika disana tertulis 3, berarti setelah 3 kali salah memasukkan password, maka IP tersebut akan otomatis terblok. Silahkan anda edit bagian ini jika dirasa kurang pas. Semakin kecil angka yang dimasukkan, maka akan semakin kecil pula kesempatan brute force attack beraksi ke server anda.



Gambar 10.14. Isi Konfigurasi File Jail.Local

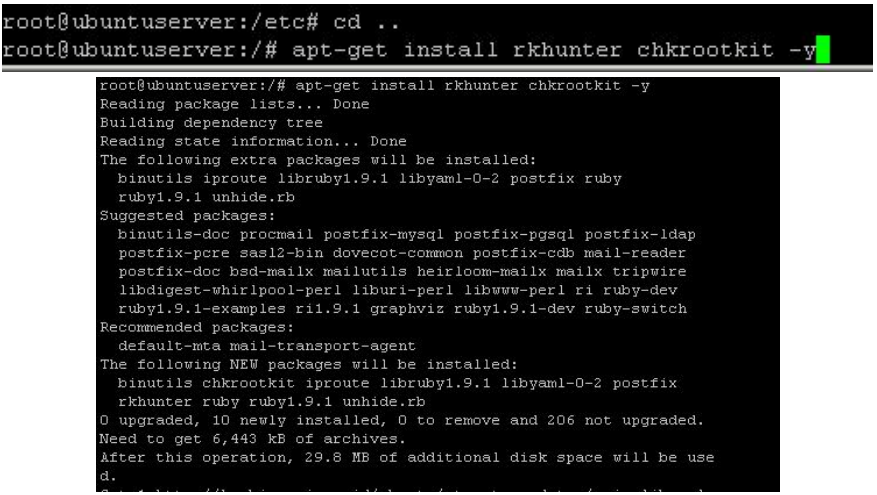
- destemail adalah bagian untuk memberikan laporan jika ada IP yang dicurigai telah berusaha melakukan serangan brute force.
- [ssh] berikut dropbear dan seterusnya kebawah bisa diatur pada bagian enable = true menjadi false atau sebaliknya. True adalah pengaturan untuk mengaktifkan (ON) dan false untuk menonaktifkan (off). Untuk portnya bisa anda sesuaikan apabila anda sudah mengganti port default dari masing-masing layanan (open ssh, dropbear dan lain-lain). Seperti contoh, jika port SSH telah diganti dari 22 ke 5678, maka pada bagian port tersebut juga harus diisi dengan port 5678.
- Fail2ban otomatis berjalan dengan konfigurasi default setelah diinstal, namun hal tersebut dapat dipastikan kembali dengan merestart service fail2ban dengan perintah **service fail2ban restart.**
- Untuk mencobanya, login pada server ubuntu menggunakan remote client melalui port ssh dengan sengaja memasukkan password salah berulang kali.



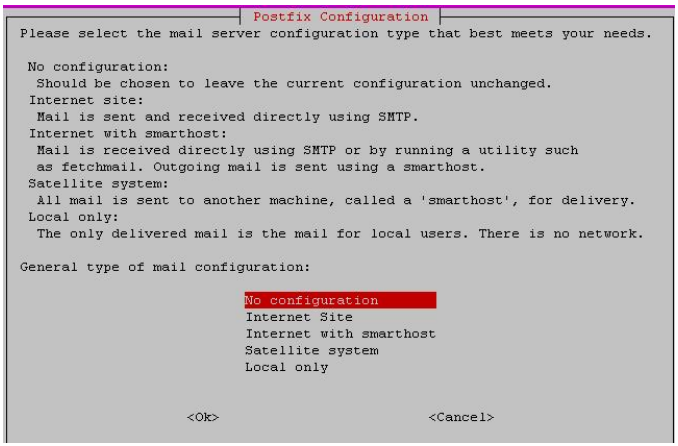
Gambar 10.14. Percobaan Login Secara Random

5. Install Rootkit Hunter

   Dibutuhkan paket software rootkit hunter dan chkrootkit untuk mendeteksi rootkit dalam server ubuntu. Lakukan Instalasi rootkit hunter dengan perintah **apt-get install rkhunter chkrootkit -y**

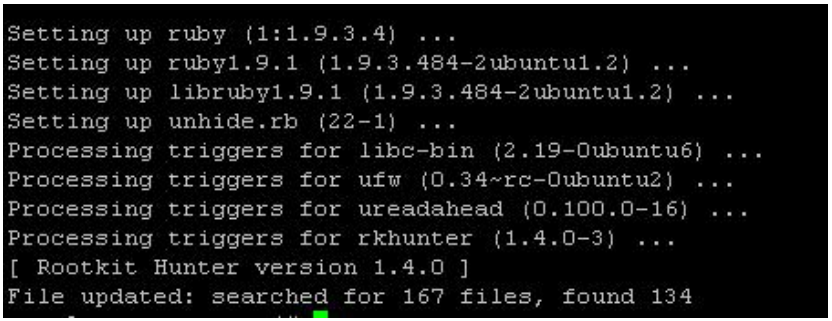


Gambar 10.15. Proses Instalasi Rootkit Hunter dan CHKROOTKIT



Gambar 10.16. Postfix Configuration

Pada gambar 10.16 terdapat postfix configuration, konfigurasi tersebut hanya diperlukan jika pada server ubuntu sudah terdapat email server dan DNS server. Dalam praktikum modul 10 ini isikan konfigurasi dengan memilih No Configuration



Gambar 10.17. Proses Instalasi Lanjutan

Ikuti proses instalasi sampai selesai seperti pada Gambar 10.17. bila proses instalasi sudah selesai, lakukan pengujian untuk mendeteksi rootkit dengan mengetikkan perintah **chkrootkit** seperti pada gambar 10.18

```
root@ubuntuserver:/# chkrootkit
ROOTDIR is `/'
Checking `amd'...                                           not found
Checking `basename'...                                      not infected
Checking `biff'...                                          not found
Checking `chfn'...                                          not infected
Checking `chsh'...                                          not infected
Checking `cron'...                                          not infected
Checking `crontab'...                                       not infected
Checking `date'...                                          not infected
Checking `du'...                                            not infected
Checking `dirname'...                                       not infected
```

Gambar 10.18 Proses Scanning CHKROOTKIT

```
root@ubuntuserver:/# rkhunter --check
[ Rootkit Hunter version 1.4.0 ]

Checking system commands...

  Performing 'strings' command checks
    Checking 'strings' command                              [ OK ]

  Performing 'shared libraries' checks
    Checking for preloading variables                       [ None found ]
    Checking for preloaded libraries                        [ None found ]
    Checking LD_LIBRARY_PATH variable                       [ Not found ]

  Performing file properties checks
    Checking for prerequisites                              [ OK ]
    /usr/sbin/adduser                                       [ OK ]
    /usr/sbin/chroot                                        [ OK ]
    /usr/sbin/cron                                          [ OK ]
    /usr/sbin/groupadd                                      [ OK ]
    /usr/sbin/groupdel                                      [ OK ]
    /usr/sbin/groupmod                                      [ OK ]
```

```
System checks summary
=====================

File properties checks...
    Files checked: 134
    Suspect files: 1

Rootkit checks...
    Rootkits checked : 307
    Possible rootkits: 0

Applications checks...
    All checks skipped

The system checks took: 54 seconds

All results have been written to the log file (/var/log/rkhunter.log)

One or more warnings have been found while checking the system.
Please check the log file (/var/log/rkhunter.log)
```

Gambar 10.19. Proses Scanning RKHUNTER

Setelah scanning chkrootkit selesai, coba juga scanning menggunakan RKHUNTER dengan mengetikkan perintah **rkhunter –check** seperti pada gambar 10.19.

Langkah- langkah praktikum diatas merupakan langkah dasar untuk mengamankan Server Ubuntu 14.04 Server yang biasa dilakukan administrator setelah instalasi awal ubuntu 14.04 selesai. Apabila server yang akan digunakan merupakan server produksi atau server yang menyimpan layanan penting seperti webserver, database, file sharing, ada baiknya metode dasar ini digabungkan dengan best practice keamanan jaringan yang lain seperti firewall IPtables, IDS, IPS, dan SSL

10.4. TUGAS MODUL 10

1. Tuliskan dalam sebuah laporan (word/pdf) langkah-langkah praktikum diatas
2. Tarik kesimpulan dalam setiap langkah praktikum
3. Kumpulkan laporan anda (nomor 1 & 2) pada pertemuan minggu depan

11

KEAMANAN PADA TELNET DAN SSH

11.1. Pendahuluan

11.1.1. Telnet

Secara teknis: Telnet adalah singkatan dari Telecomunications Network Protocol, merupakan remote login yang terjadi pada jaringan internet disebabkan karena adanya service dari protocol Telnet.

Fungsi utama Telnet untuk memungkinkan pengguna dapat mengakses komputer lain secara remote melalui jaringan internet.

- Nama package : telnetd
- File konfigurasi : inetd.conf
- Port number : 23
- Protokol layer transport

11.1.2. SSH

Secara teknis SSH adalah aplikasi pengganti remote login seperti telnet, rsh, dan rlogin, yang jauh lebih aman.

Fungsi Fungsi utama SSH adalah untuk mengakses mesin secara remote. Sama seperti telnet, SSH Client menyediakan User dengan Shell untuk remote ke mesin.

- Nama package : openssh
- File konfigurasi : sshd_config
- Port number : 22
- Protokol layer transport

11.1.3. Cara Kerja SSH

Pada saat client melakukan komunikasi dengan server

client mengirimkan request kemudian dibalas dengan server dan versi yang digunakan harus sama antara ssh client dan server. Setelah itu server mengirimkan public key yang digunakan untuk mengenkripsi session key yang dikirim oleh client. Session yang dikirim berupa username dan password. Setelah session diterima oleh server maka session tersebut didekripsi oleh server. Kemudian didekripsi lagi dengan public key dari client dan dikirm lagi ke client. Setelah client mendapatka verifikasi barulah client dapat mengakses ssh yang diinginkan.

11.2. Perbedaan cara Kerja SSH dan Telnet

Telnet merupakan aplikasi yang bisa membantu untuk mengakses sebuah komputer dengan standart port 23. Dengan telnet ini user bisa masuk untuk mendapatkan shell yang dituju dan dapat menjalankan perintah-perintah. Namun telnet memiliki kelemahan yakni data yang ditranmisikan melalui clear teks, sehingga mudah untuk ditangkap paket yang ditransmisikan oleh telnet tanpa didekrip. Berbeda dengan ssh, yang secara konsep sama, namun ssh ini memasuki komputer dengan kelebihannya yakni dapat mengkopi file terenkrip. File terenkrip ini dilakukan oleh client dan server. SSH memberikan alternative yang aman terhadap remote session dan file transfer prootocol. Protokol SSH mendukung otentikasi terhadap remote host, yang dengan demikian meminimalkan ancaman pemalsuan identitas client lewat IP address spoofing maupun manipulasi DNS. Aplikasi seperti Telnet tidak menggunakan enkripsi sedangkan SSH dilengkapi dengan enkripsi.