

Ulan Purnama Sari

09011181320003

KEAMANAN JARINGAN KOMPUTER



Ulan Purnama Sari

09011181320003

Program Studi Sistem Komputer

Fakultas Ilmu Komputer

Universitas Sriwijaya

2017

TASK 5

Evaluasi Keamanan Sistem “Actual Exploit”

Exploit adalah sebuah kode yang menyerang keamanan komputer secara spesifik. Exploit banyak digunakan untuk penentrasi baik secara legal ataupun ilegal untuk mencari kelemahan (Vulnerability) pada komputer tujuan. Bisa juga dikatakan sebuah perangkat lunak yang menyerang kerapuhan keamanan (*security vulnerability*) yang spesifik namun tidak selalu bertujuan untuk melancarkan aksi yang tidak diinginkan.

1. Pertama kita setting IP terlebih dahulu di Ubuntu Server dan DVL

```
bt ~ # ifconfig eth0 192.168.100.20 netmask 255.255.255.0_
bt ~ # ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:41:4E:02
          inet addr:192.168.100.20  Bcast:192.168.100.255  Mask:255.255.255.0
          UP BROADCAST NOTRAILERS RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:83 errors:0 dropped:0 overruns:0 frame:0
          TX packets:84 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:9828 (9.5 KiB)  TX bytes:9502 (9.2 KiB)
          Base address:0xd010 Memory:f0000000-f0020000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b)  TX bytes:0 (0.0 b)

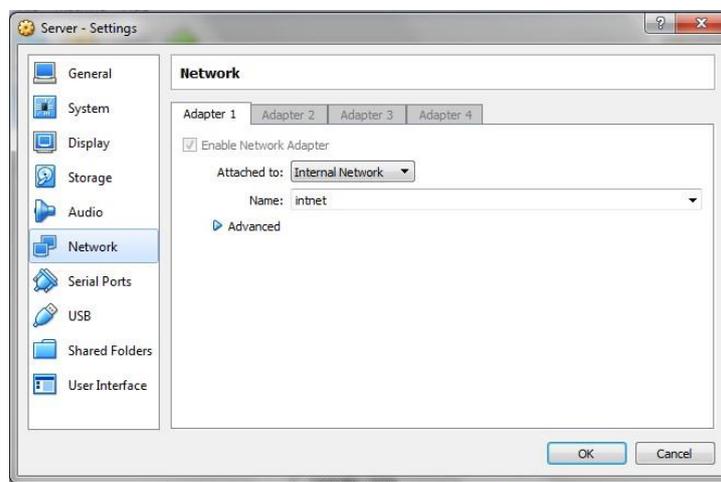
root@ubuntu:/home/ubuntu# ifconfig eth0 192.168.100.10 netmask 255.255.255.0
```

```
root@ubuntu:/home/ubuntu# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:10:8a:e2
          inet addr:192.168.100.10  Bcast:192.168.100.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe10:8ae2/64  Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:104 errors:0 dropped:0 overruns:0 frame:0
          TX packets:135 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:10002 (10.0 KB)  TX bytes:18516 (18.5 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128  Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:78 errors:0 dropped:0 overruns:0 frame:0
          TX packets:78 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:5688 (5.6 KB)  TX bytes:5688 (5.6 KB)

root@ubuntu:/home/ubuntu#
```

2. Setting network menjadi internal network seperti gambar dibawah ini.



3. PING IP Address ubuntu ke DVL

Kenapa harus melakukan perintah ping setelah setting ip address? Agar kita tau kedua jaringan tersebut saling tersambung atau belum. Pada gambar dibawah membuktikan bahwa jaringan ubuntu sama DVL telah tersambung.

```
bt ~ # ping 192.168.100.10
PING 192.168.100.10 (192.168.100.10) 56(84) bytes of data.
64 bytes from 192.168.100.10: icmp_seq=1 ttl=64 time=0.315 ms
64 bytes from 192.168.100.10: icmp_seq=2 ttl=64 time=1.13 ms
64 bytes from 192.168.100.10: icmp_seq=3 ttl=64 time=0.865 ms
64 bytes from 192.168.100.10: icmp_seq=4 ttl=64 time=0.569 ms
64 bytes from 192.168.100.10: icmp_seq=5 ttl=64 time=0.783 ms
```

4. Scanning nmap

Nmap -sV 192.168.100.10

Nmap -sv (service yg sedang berjalan)

```
root@ubuntu:/home/ubuntu# nmap -sV 192.168.100.20
Starting Nmap 6.40 ( http://nmap.org ) at 2017-03-15 19:38 PDT
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.100.20
Host is up (0.00033s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
631/tcp   open  ipp      CUPS 1.1
3306/tcp   open  mysql    MySQL (unauthorized)
MAC Address: 08:00:27:41:4E:02 (Cadmus Computer Systems)

Service detection performed. Please report any incorrect results at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.33 seconds
root@ubuntu:/home/ubuntu#
```

```
bt ~ # nmap -sV 192.168.100.10
Starting Nmap 4.20 ( http://insecure.org ) at 2017-03-16 02:43 GMT
Interesting ports on 192.168.100.10:
Not shown: 1694 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      (protocol 2.0)
53/tcp    open  domain
80/tcp    open  http     Apache httpd 2.4.7 ((Ubuntu))
1 service unrecognized despite returning data. If you know the service/version, please submit the following fingerprint at http://www.insecure.org/cgi-bin/servicefp-submit.cgi :
SF-Port22-TCP:U=4.20%I=7%D=3/16%Time=58C9FBCD%P=i686-pc-linux-gnu%r(NULL,2
SF:9,"SSH-2.0-OpenSSH_6.6p1\u00d0Ubuntu-2ubuntu2\r\n");
MAC Address: 08:00:27:10:BA:E2 (Cadmus Computer Systems)

Service detection performed. Please report any incorrect results at http://insecure.org/nmap/submit/ .
Nmap finished: 1 IP address (1 host up) scanned in 24.396 seconds
bt ~ # _
```

Seperti dapat kita lihat pada gambar diatas, dimana ketika kita memasukan perintah seperti gambar diatas maka aplikasi tersebut akan melakukan scanning port yang terbuka pada IP target (192.168.100.20). dimana pada langkah sebelumnya kita telah membuka/menjalankan seluruh aplikasi yang terdapat pada mesin DVL dan yang terjadi ketika kita melakukan scanning pada target yaitu mesin DVL maka aplikasi yang sedang berjalan itulah yang dianggap sebagai celah untuh menyerang karena port dalam keadaan terbuka.

Service ssh menggunakan bruteforce mencoba melakukan input password menggunakan beberapa tools, tools yang bisa digunakan adalah : Hydra dan Nmap.

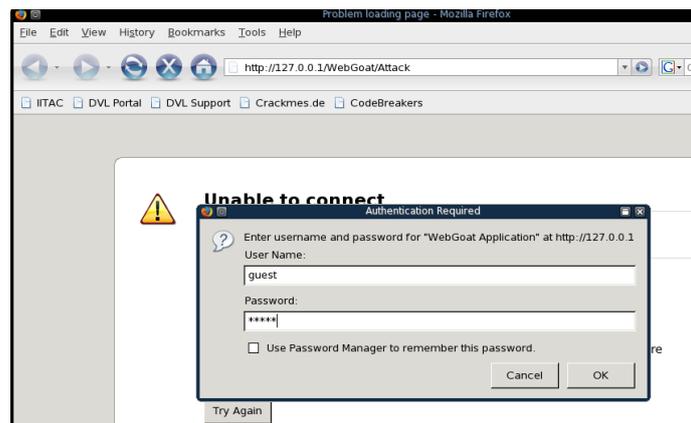
5. Web Goat pada Dvl

Program yang akan digunakan adalah Web Goat dengan port 80 yang merupakan bagian dari web exploitation.

Web Goat adalah Project Open Source yang dapat digunakan agar orang lain bisa belajar web hacking. Salah satunya adalah SQL Injection.



Gambar 5.1



Gambar 5.2

Gambar diatas membuktikan bahwa perintah dari gambar 5.1 berhasil masuk maka akan muncul tampilan seperti gambar 5.2 untuk kita bisa login.

Setelah itu kita masukkan username "guest" dan password "guest" untuk melanjutkan membuka Web Goat. Jika berhasil maka akan langsung masuk ke interfaces tampilan awal Web Goat seperti gambar 5.3



OWASP WebGoat V5.1

Thank you for using WebGoat!

This program is a demonstration of common web application flaws. The exercises are intended to provide hands on experience with application penetration testing techniques.

The WebGoat project is lead by Bruce Mayhew. Please send all comments to Bruce at webgoat@owasp.org.



OWASP
The Open Web Application Security Project



ASPECT SECURITY
Application Security Specialists

<p>WebGoat Design Team</p> <p>Bruce Mayhew David Anderson Rogan Dawes Laurence Casey (Graphics)</p> <p>Special Thanks for V5.1</p> <p>OWASP Spring of Code Erwin Geirnaert (http://www.zionsecurity.com/)</p> <p>To all who have sent comments</p>	<p>Lesson Contributors</p> <p>Aspect Security Sherif Koussa Romain Brechet</p> <p>Documentation Contributors</p> <p>Sherif Koussa (http://www.macadamian.com/) Erwin Geirnaert (http://www.zionsecurity.com/)</p>
---	---

[Start WebGoat](#)

Gambar 5.3



Logout ?

Http Basics

OWASP WebGoat V5.1 < Hints > Show Params Show Cookies Show Java Show Solution Lesson Plans

[Restart this Lesson](#)

Admin Functions
General
Code Quality
Concurrency
Unvalidated Parameters
Access Control Flaws
Authentication Flaws
Session Management Flaws
Cross-Site Scripting (XSS)
Buffer Overflows
Injection Flaws

- [Command Injection](#)
- [Blind SQL Injection](#)
- [Numeric SQL Injection](#)
- [Log Spoofing](#)
- [XPath Injection](#)
- [String SQL Injection](#)
- [LAB: SQL Injection](#)
 - [Stage 1: String SQL Injection](#)
 - [Stage 2: Parameterized Query #1](#)
 - [Stage 3: Numeric SQL Injection](#)
 - [Stage 4: Parameterized Query #2](#)
- [Database Backdoors](#)
- [Improper Error Handling](#)

Enter your name in the input field below and press "go" to submit. The server will accept the request, reverse the input, and display it back to the user, illustrating the basics of handling an HTTP request.

The user should become familiar with the features of WebGoat by manipulating the above buttons to view hints, show the HTTP request parameters, the HTTP request cookies, and the Java source code.

Enter your name: [Go!](#)

OWASP Foundation | Project WebGoat

Melakukan pencarian semua nama dengan last name Smith atau user name Smith Programmer String SQL Injection, Tidak melakukan filter input yang masuk.

OWASP WebGoat V5.1

String SQL Injection

Admin Functions
General
Code Quality
Concurrency
Unvalidated Parameters
Access Control Flaws
Authentication Flaws
Session Management Flaws
Cross-Site Scripting (XSS)
Buffer Overflows
Injection Flaws

Command Injection
Blind SQL Injection
Numeric SQL Injection
Log Spoofing
XPath Injection
String SQL Injection
LAB: SQL Injection

Stage 1: String SQL Injection
Stage 2: Parameterized Query #1
Stage 3: Numeric SQL Injection
Stage 4: Parameterized Query #2

Database Backdoors
Improper Error Handling
Insecure Storage
Denial of Service
Insecure Configuration
Web Services

Logout ?

Restart this Lesson

SQL injection attacks represent a serious threat to any database-driven site. The methods behind an attack are easy to learn and the damage caused can range from considerable to complete system compromise. Despite these risks, an incredible number of systems on the internet are susceptible to this form of attack.

Not only is it a threat easily instigated, it is also a threat that, with a little common-sense and forethought, can easily be prevented.

It is always good practice to sanitize all input data, especially data that will be used in OS command, scripts, and database queries, even if the threat of SQL injection has been prevented in some other manner.

General Goal(s):

The form below allows a user to view their credit card numbers. Try to inject an SQL string that results in all the credit card numbers being displayed. Try the user name of 'Smith'.

Enter your last name:

```
SELECT * FROM user_data WHERE last_name = 'Smith'
```

USERID	FIRST_NAME	LAST_NAME	CC_NUMBER	CC_TYPE	COOKIE	LOGIN_COUNT
102	John	Smith	2435600002222	MC		0
102	John	Smith	4352209902222	AMEX		0

OWASP Foundation | Project WebGoat

* Congratulations. You have successfully completed this lesson.

* Bet you can't do it again! This lesson has detected your successful attack and has now switched to a defensive mode. Try again to attack a parameterized query.

Enter your last name:

```
SELECT * FROM user_data WHERE last_name = 'test' or 1=1 --'
```

USERID	FIRST_NAME	LAST_NAME	CC_NUMBER	CC_TYPE	COOKIE	LOGIN_COUNT
101	Joe	Snow	987654321	VISA		0
101	Joe	Snow	2234200065411	MC		0
102	John	Smith	2435600002222	MC		0
102	John	Smith	4352209902222	AMEX		0
103	Jane	Plane	123456789	MC		0
103	Jane	Plane	333498703333	AMEX		0
10312	Jolly	Hershey	176896789	MC		0
10312	Jolly	Hershey	333300003333	AMEX		0
10323	Grumpy	White	673834489	MC		0
10323	Grumpy	White	33413003333	AMEX		0
15603	Peter	Sand	123609789	MC		0
15603	Peter	Sand	338893453333	AMEX		0
15613	Joesph	Something	33843453533	AMEX		0

Pada gambar diatas bahwa ketika kita mencoba melakukan percobaan dengan memasukan query seperti gambar diatas maka didapatlah data tersebut. Maksud dari query tersebut adalah :

Di awalnya dia akan menambahkan tanda petik dan akan membaca last name yg kita masukkan , maksud 1=1 adalah boolean true walaupun kita salah masih akan bernilai true. Itulah kesalahan dari program karna tidak memfilter terlebih dahulu.

Analisis:

Program Webgoat dapat digunakan untuk mendapatkan berbagai macam informasi dari target dengan layanan yang dapat dibuka atau terbuka. Penggunaan injection SQL yang telah dilakukan adalah pengaksesan data user melalui database dengan penggunaan kata kunci pencarian dari informasi yang diinginkan. Data dalam database yang didapatkan merupakan data penting seperti id user dan nomor-nomor penting.