



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

## Langkah Praktikum

### Instalasi Persiapan Lingkungan Praktikum

#### 1. Langkah 1

a. Buka terminal pada sistem operasi Linux

b. Pada terminal, lakukan instalasi git:

```
sudo apt install git
```

Git adalah sistem kontrol versi yang digunakan untuk melacak perubahan dalam kode sumber selama pengembangan perangkat lunak. Untuk menginstal Git di sistem operasi Linux, gunakan perintah `sudo apt install git` pada terminal. Perintah ini akan mengunduh dan menginstal Git serta dependensinya dari repositori paket sistem. Setelah proses instalasi selesai, gunakan Git untuk mengelola proyek secara efisien.

#### 2. Langkah 2

c. Lakukan instalasi docker-compose dengan perintah:

```
sudo apt install docker-compose
```



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

```
(user@kali)-[~]
$ sudo apt install docker-compose
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libboost-dev libboost1.83-dev libbftw3-single3 libopenblas-dev libopenblas-pthread-dev libopenblas0 libpython3-all-dev libpython3.12
  libpython3.12-dev libxsimd-dev pipewire-alsa pipewire-audio python3-all-dev python3-beniget python3-gast python3-pythran python3.12-dev
  xtl-dev
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  cgroups-mount containerd docker.io libintl-perl libintl-xs-perl libmodule-find-perl libmodule-scandeps-perl libproc-processstable-perl
  libsort-naturally-perl needrestart python3-compose python3-docker python3-dockerpty python3-texttable runc tini
Suggested packages:
  containernetworking-plugins docker-doc aufs-tools btrfs-progs debotstrap rinse rootlesskit xfsprogs zfs-fuse | zfsutils-linux
Recommended packages:
  criu
The following NEW packages will be installed:
  cgroups-mount containerd docker-compose docker.io libintl-perl libintl-xs-perl libmodule-find-perl libmodule-scandeps-perl
  libproc-processstable-perl libsort-naturally-perl needrestart python3-compose python3-docker python3-dockerpty python3-texttable runc tini
0 upgraded, 17 newly installed, 0 to remove and 8 not upgraded.
Need to get 68.1 MB of archives.
After this operation, 273 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://http.kali.org/kali kali-rolling/main amd64 runc amd64 1.1.12-ds1-1 [2757 kB]
Get:2 http://http.kali.org/kali kali-rolling/main amd64 python3-texttable all 1.6.7-1 [11.9 kB]
Get:3 http://mirror.primelink.net.id/kali kali-rolling/main amd64 python3-dockerpty all 1.6.7-1 [255 kB]
Get:4 http://http.kali.org/kali kali-rolling/main amd64 tini amd64 0.19.0-1 [255 kB]
Get:5 http://xsrv.moratelindo.io/kali kali-rolling/main amd64 needrestart all 3.6-7 [59.7 kB]
Get:6 http://http.kali.org/kali kali-rolling/main amd64 python3-docker all 5.0.3-1 [90.2 kB]
Get:7 http://http.kali.org/kali kali-rolling/main amd64 python3-compose all 1.29.2-6 [113 kB]
Get:8 http://http.kali.org/kali kali-rolling/main amd64 python3-docker all 5.0.3-1 [90.2 kB]
Get:9 http://http.kali.org/kali kali-rolling/main amd64 python3-dockerpty all 1.6.7-1 [255 kB]
Get:10 http://http.kali.org/kali kali-rolling/main amd64 python3-texttable all 1.6.7-1 [11.9 kB]
Get:11 http://http.kali.org/kali kali-rolling/main amd64 libsort-naturally-perl all 1.03-4 [13.1 kB]
Get:12 http://http.kali.org/kali kali-rolling/main amd64 containerd amd64 1.6.24-ds1-1 [26.9 MB]
```

Docker-Compose adalah alat yang memungkinkan pengguna untuk mendefinisikan dan menjalankan aplikasi Docker yang terdiri dari beberapa wadah. Untuk menginstal Docker-Compose di sistem operasi Linux, gunakan perintah `'sudo apt install docker-compose'` pada terminal. Perintah ini akan mengunduh dan menginstal Docker-Compose serta dependensinya dari repositori paket sistem. Setelah proses instalasi selesai, maka dapat menggunakan Docker-Compose untuk mengelola aplikasi yang terdiri dari beberapa wadah Docker dengan mudah.

### 3. Langkah 3

d. Kemudian lakukan cloning github repository untuk lab percobaan bagian 1:  
`git clone https://github.com/adhiyaksactf/sqli-part1.git`

```
(user@kali)-[~]
$ git clone https://github.com/adhiyaksactf/sqli-part1.git
Cloning into 'sqli-part1' ...
remote: Enumerating objects: 122, done.
remote: Counting objects: 100% (122/122), done.
remote: Compressing objects: 100% (102/102), done.
remote: Total 122 (delta 29), reused 103 (delta 17), pack-reused 0
Receiving objects: 100% (122/122), 1.92 MiB | 1.28 MiB/s, done.
Resolving deltas: 100% (29/29), done.
```



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

Cloning adalah proses membuat salinan lokal dari repositori Git yang ada. Untuk melakukannya, gunakan perintah `git clone` di terminal, diikuti dengan URL repositori yang ingin diklon. Perintah `'git clone https://github.com/adhiyaksactf/sqli-part1.git'` akan membuat salinan lokal dari repositori `sqli-part1.git` di direktori saat ini di terminal. Setelah proses cloning selesai, kita akan memiliki salinan lengkap dari repositori tersebut di komputer lokal untuk digunakan dalam praktikum.

4. Langkah 4

e. Lakukan cloning github repository untuk lab percobaan bagian 2:

```
git clone https://github.com/adhiyaksactf/sqli-part2.git
```

```
(user@kali)~$ git clone https://github.com/adhiyaksactf/sqli-part2.git
Cloning into 'sqli-part2' ...
remote: Enumerating objects: 47, done.
remote: Counting objects: 100% (47/47), done.
remote: Compressing objects: 100% (38/38), done.
remote: Total 47 (delta 16), reused 38 (delta 8), pack-reused 0
Receiving objects: 100% (47/47), 1.33 MiB | 906.00 KiB/s, done.
Resolving deltas: 100% (16/16), done.
```

Dani Adrian.txt

File Edit View

Nama : Dani Adrian  
NIM : 225150201111009

Ln 1, Col 1 | 40 characters | 100% | Windows

Dengan menjalankan perintah di atas pada terminal, kita akan membuat salinan lokal dari repositori `sqli-part2.git` di direktori saat ini di terminal. Setelah proses cloning selesai, kita akan memiliki salinan lengkap dari repositori tersebut di komputer lokal untuk digunakan dalam praktikum.

5. Langkah 5

f. Masuk ke dalam direktori `sqli-part1` dengan perintah:

```
cd sqli-part1
```

g. Melakukan build docker untuk `sqli-part1` dengan perintah:

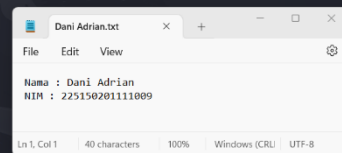
```
sudo docker-compose up -d
```



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

```
(user@kali)-[~]
$ cd sql-i-part1
(user@kali)-[~/sql-i-part1]
$ sudo docker-compose up -d
Pulling db (mariadb:latest)...
latest: Pulling from library/mariadb
bccd10f490ab: Pull complete
d9d8e1823c6f: Pull complete
4b658f15686b: Pull complete
183080ffcdde: Pull complete
fc35f7aae1e5: Pull complete
59efd043a883: Pull complete
676a7ad9f737: Pull complete
335ef3100b9e: Pull complete
Digest: sha256:a009cebdcd294d08590817a3ebdf3da822a1509187ba946ab7b384c8a333ac94
Status: Downloaded newer image for mariadb:latest
Building app
Sending build context to Docker daemon 4.611MB
Step 1/3 : FROM php:8.2-apache-bullseye
8.2-apache-bullseye: Pulling from library/php
c0ede72937fa: Pull complete
a110dc6bd4f3: Pull complete
2a676cd3cc4a: Pull complete
0074f28e265c: Pull complete
9a771575b0ad: Pull complete
1cbf0ec1c723: Pull complete
b1fc7c6855eb: Pull complete
9c2efc19334a: Pull complete
4e4e72dac5d9: Pull complete
9e3d53e714a1: Pull complete
```



Dengan menjalankan perintah tersebut, Docker akan membangun dan menjalankan kontainer Docker yang didefinisikan dalam file `docker-compose.yml` di mode detasemen (-d), yang berarti kontainer akan berjalan di latar belakang. Setelah selesai, kita dapat mengakses aplikasi yang berjalan di kontainer Docker tersebut.

## 6. Langkah 6

- h. Jalankan perintah berikut untuk mengecek apakah service sudah berjalan:  
`sudo docker ps`
- i. Apabila berhasil maka service akan berjalan pada port 81, sehingga dapat mengakses pada browser dengan alamat `localhost:81`



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

The image shows two screenshots. The top screenshot is a terminal window with the prompt `(user@kali)-[~/sqli-part1]`. It displays the output of `sudo docker ps`, showing a container named `sqli-part1` with image `mariadb:latest` and command `"docker-php-entrypoint..."`. The container is created 2 minutes ago and is up 2 minutes. The ports are `0.0.0.0:81->80/tcp` and `:::81->80/tcp`. The names are `sqli-part1-app` and `sqli-part1-db`. The bottom screenshot is a web browser window showing the `localhost:81` address. It displays a dark-themed login page titled `ADMIN PANEL` with a green key icon. There are input fields for `USERNAME` and `PASSWORD`, and a `LOGIN` button. A text editor window titled `Dani Adrian.txt` is open in the foreground, showing the text `Nama : Dani Adrian` and `NIM : 225150201111009`.

Dengan mengakses alamat tersebut di browser, kita dapat melihat layanan yang berjalan dengan baik pada port 81.

7. Langkah 7

- j. Kembali pada terminal Anda lakukan perintah untuk masuk ke direktori `sqli-part2`:

```
cd ../sqli-part2
```

- k. Melakukan build docker untuk `sqli-part2` dengan perintah:

```
sudo docker-compose up -d
```



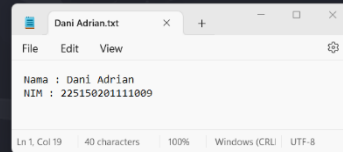
**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

```
(user@kali)~/sqli-part1
$ cd ../sqli-part2

(user@kali)~/sqli-part2
$ sudo docker-compose up -d
Creating network "sqli-part2_default" with the default driver
Building app
Sending build context to Docker daemon 3.119MB
Step 1/3 : FROM php:8.2-apache-bullseye
--> cdab551394a2
Step 2/3 : RUN docker-php-ext-install mysqli
--> Using cache
--> 95b366240e79
Step 3/3 : COPY ./app /var/www/html
--> e014dc4fc464
Successfully built e014dc4fc464
WARNING: Image for service app was built because it did not already exist. To rebuild this image you must use `docker-compose build` or `docker-c
ompose up --build`.
Creating sqli-part2-db ... done
Creating sqli-part2-app ... done

(user@kali)~/sqli-part2
$
```



Dengan menjalankan perintah tersebut, Docker akan membangun dan menjalankan kontainer Docker yang didefinisikan dalam file `docker-compose.yml` di mode detasemen (-d), yang berarti kontainer akan berjalan di latar belakang. Setelah selesai, kita dapat mengakses aplikasi yang berjalan di kontainer Docker tersebut.

## 8. Langkah 8

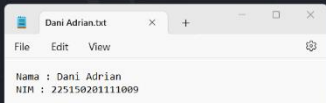
1. Jalankan perintah berikut untuk mengecek apakah service sudah berjalan:

`sudo docker ps`

- m. Apabila berhasil maka service akan berjalan pada port 82, sehingga dapat mengakses pada browser dengan alamat `localhost:82`

```
(user@kali)~/sqli-part2
$ sudo docker ps
CONTAINER ID   IMAGE          COMMAND                  CREATED        STATUS        PORTS                               NAMES
cd9ab00f99fd   sqli-part2    "docker-php-entrypoi..." 43 seconds ago Up 43 seconds 0.0.0.0:82->80/tcp, :::82->80/tcp   sqli-part2-app
40eb6c6060f0   mariadb:late  "docker-entrypoint.s..." 44 seconds ago Up 43 seconds 3306/tcp                               sqli-part2-db
b02a697648e1   sqli-part1    "docker-php-entrypoi..." 8 minutes ago  Up 7 minutes  0.0.0.0:81->80/tcp, :::81->80/tcp   sqli-part1-app
2c92d3536872   mariadb:late  "docker-entrypoint.s..." 8 minutes ago  Up 8 minutes  3306/tcp                               sqli-part1-db

(user@kali)~/sqli-part2
$
```



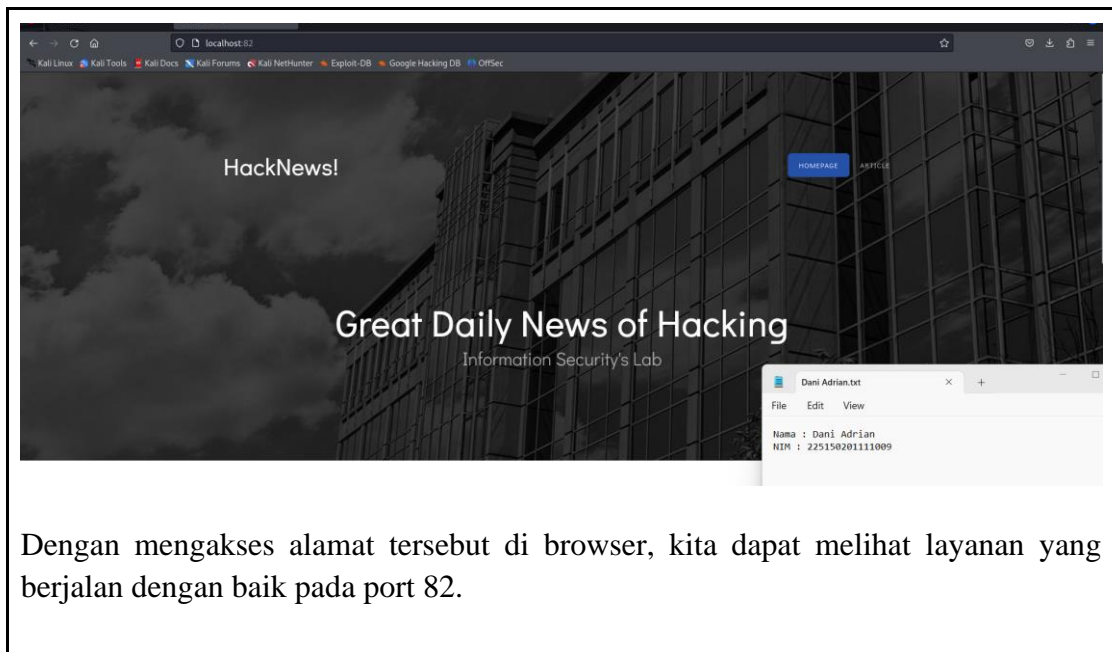


**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA**

---

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

---



Dengan mengakses alamat tersebut di browser, kita dapat melihat layanan yang berjalan dengan baik pada port 82.

### **SQL Injection Basic**

#### **1. Langkah 1**

- a. Masuk pada bagian lab pertama (localhost:81).
- b. Lakukan percobaan serangan SQL Injection untuk mem-bypass halaman login milik admin dengan menggunakan query berikut:

`\ OR 1=1 #`





**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas



Serangan SQL Injection: Pada kolom yang dimaksudkan untuk username atau email, masukkan query SQL Injection ' OR 1=1#. Dengan memasukkan ini, kita mencoba untuk memanipulasi kueri SQL yang dijalankan oleh halaman login.

2. Langkah 2

- c. Apabila Anda berhasil, Jelaskan bagaimana query yang Anda gunakan dapat mem-bypass laman tersebut, tanpa perlu mengetahui username dan password milik admin.

Pada langkah tersebut, query yang digunakan adalah ' OR 1=1#.

' : Ini adalah tanda kutip satu (single quote) yang menandakan awal dari string pada query SQL.

OR: Ini adalah operator logika yang digunakan untuk menggabungkan kondisi-kondisi dalam sebuah pernyataan SQL. Dalam kasus ini, digunakan untuk menyisipkan kondisi tambahan ke dalam pernyataan WHERE yang ada.





**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA**

---

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

---

**1=1** : Ini adalah kondisi logika yang selalu bernilai benar (true) dalam SQL. Dengan demikian, menggunakannya akan memastikan bahwa kondisi ini selalu terpenuhi, yang berarti baris data yang diperiksa akan selalu dipilih.

**#** : Ini adalah tanda pagar (hash) yang digunakan untuk menandakan komentar dalam SQL. Dengan meletakkan tanda pagar setelah query, kita dapat "menutup" query yang ada sehingga bagian lain dari query yang mungkin ada setelahnya akan diabaikan.

Dengan menggabungkan semua elemen ini, query ' OR 1=1# digunakan untuk menyisipkan kondisi tambahan (OR 1=1) ke dalam pernyataan SQL yang sedang dieksekusi, yang menyebabkan pernyataan tersebut selalu bernilai benar. Akibatnya, halaman login akan melewati verifikasi username dan password karena kueri yang dieksekusi secara efektif menjadi "temukan pengguna mana pun yang memiliki nama pengguna apa pun dan kata sandi apa pun" karena 1=1 selalu benar. Sehingga, halaman akan membiarkan akses tanpa memerlukan kredensial yang valid.

### **SQL Union Injection Basic**

Berikut ini langkah-langkah yang perlu dilakukan dalam menyelesaikan lab praktikum:

1. Langkah 1
  - a. Masuk pada bagian lab kedua (localhost:82).
  - b. Setelah berhasil masuk ke dalam web, pelajari bagaimana web tersebut bekerja, dan temukan letak celah dari website tersebut. (Celah atau titik injeksi berada di parameter url article)
  - c. Apabila Anda berhasil menemukan letak celah atau titik injeksi dari website tersebut, berikanlah tanda ' (single-quote) pada parameter tersebut. Jelaskan bagaimana dapat menyebabkan website tersebut menampilkan error.



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

Perhatikan parameter URL, khususnya **artikel**.

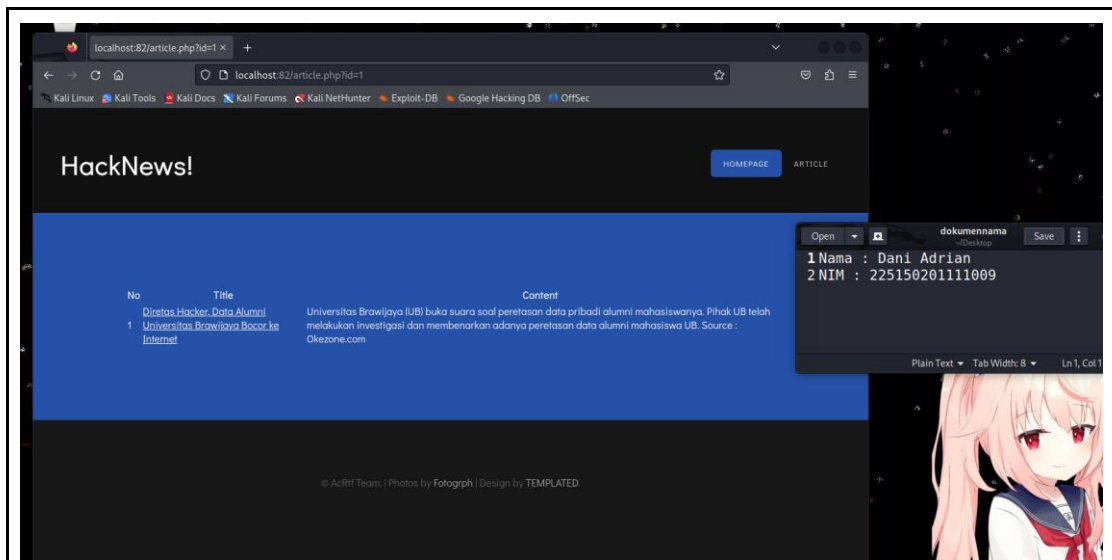
No	Title	Content
1	<a href="#">Diretas Hacker Data Alumni Universitas Brawijaya Bocor ke Internet</a>	Universitas Brawijaya (UB) buka suara soal peretasan data pribadi alumni mahasiswanya. Pihak UB telah melakukan investigasi dan membenarkan adanya peretasan data alumni mahasiswa UB. Source : Okezone.com
2	<a href="#">Indonesia kekurangan Bakat Cyber Security</a>	Indonesia kekurangan bakat cyber security dan itu menimbulkan masalah yang sangat nyata dalam industri strategis, pertahanan, kesatuan bangsa dan bisnis. Bayangkan bila terjadi perang cyber istilah beberapa tentara Cyber Indonesia yang dapat membela dan memperkuat pertahanan bangsa. Source : kominfo.go.id

Pilih salah satu Artikel



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas



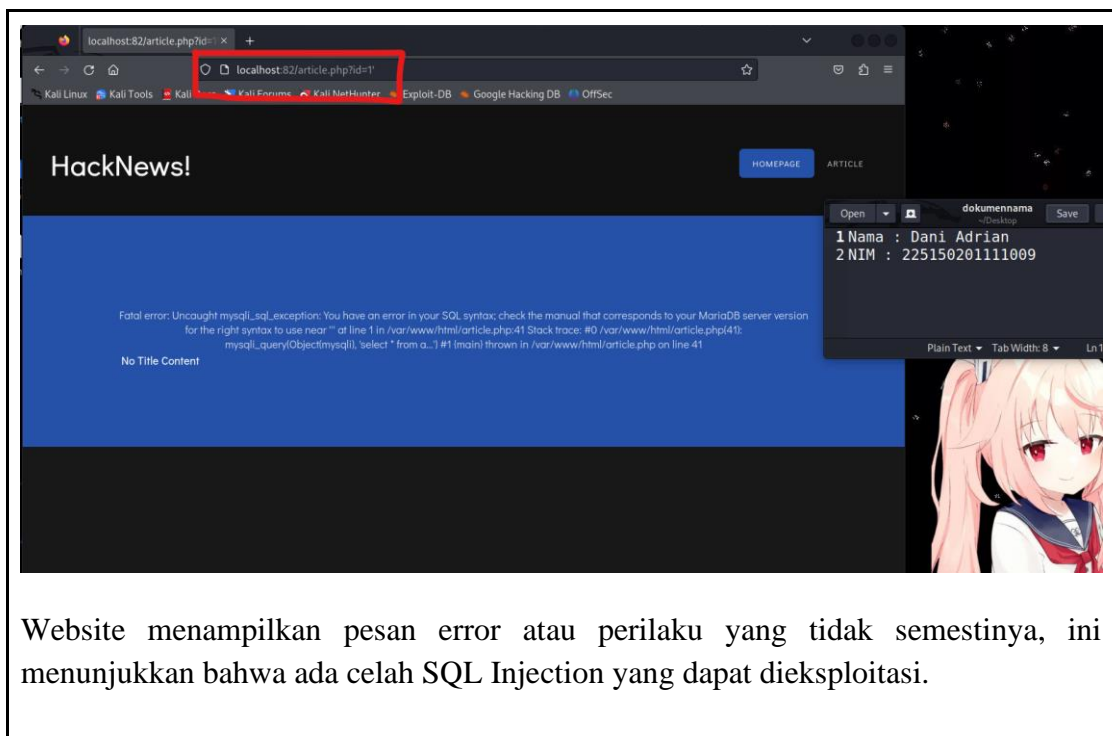
Tambahkan Tanda Single-Quote: Pada parameter `http://localhost:82/article.php?id=1`, tambahkan tanda single-quote (') sebagai input. Ini akan mengakibatkan sebuah kueri SQL yang salah saat dijalankan oleh server.

Sesudah ditambahkan single-quote :



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas



Website menampilkan pesan error atau perilaku yang tidak semestinya, ini menunjukkan bahwa ada celah SQL Injection yang dapat dieksploitasi.

2. Langkah 2

- d. Lakukan serangan berikutnya untuk mengetahui berapa jumlah kolom yang dimiliki database pada website tersebut dengan menggunakan query:

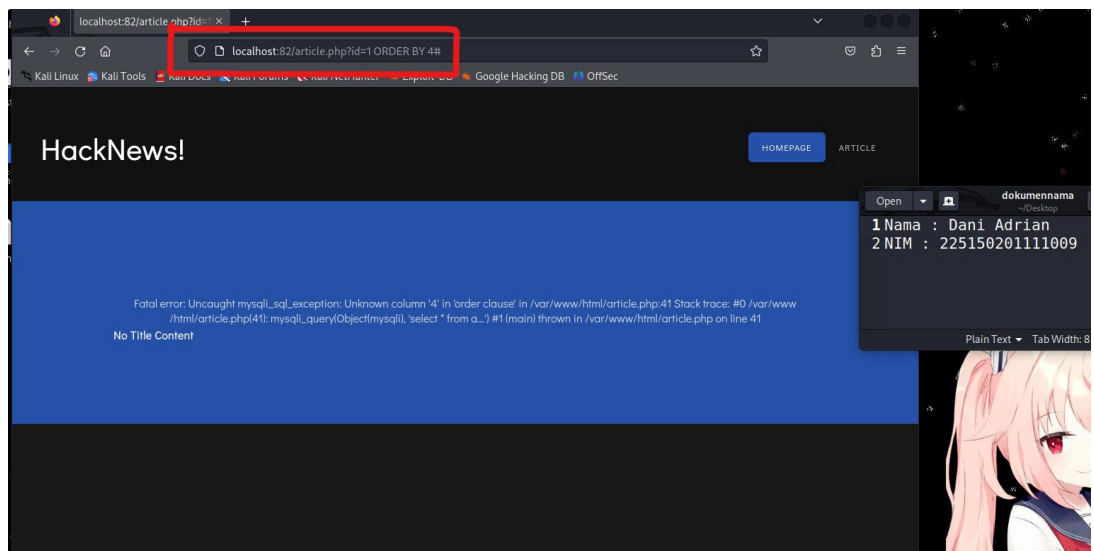
1 ORDER BY 4#

- e. Lalu coba kurangi nilai ORDER BY menjadi 3, lalu apa yang terjadi pada laman web tersebut?



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas



Jumlah kolom pada tabel database kurang dari 4, sistem database menghasilkan kesalahan karena mencoba mengurutkan hasil berdasarkan kolom yang tidak ada. Respons dari halaman web berupa pesan error.



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

localhost:82/article.php?id=1 ORDER BY 3#

Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

HackNews! HOMEPAGE ARTICLE

No	Title	Content
1	<a href="#">Diretas Hacker, Data Alumni Universitas Brawijaya Bocor ke Internet</a>	Universitas Brawijaya (UB) buka suara soal peretasan data pribadi alumni mahasiswanya. Pihak UB telah melakukan investigasi dan membenarkan adanya peretasan data alumni mahasiswa UB. Source : Okezone.com

© AcRif Team, | Photos by Fotogrph | Design by TEMPLATED

Open dokumennama Desktop  
1 Nama : Dani Adrian  
2 NIM : 225150201111009  
Plain Text Tab Width: 8

Melalui serangan SQL Injection yang kedua, dengan mengurangi nilai ORDER BY menjadi 3, kita mencoba mengkonfirmasi apakah jumlah kolom yang dimiliki oleh tabel database kurang dari 3. Hasilnya tidak akan terjadi kesalahan, dan halaman web kemungkinan akan berperilaku seperti biasa menunjukkan jumlah kolom yang dimiliki oleh tabel database lebih dari atau sama dengan 3.

### 3. Langkah 3

- f. Selanjutnya kita bisa mencari column yang dapat kita injeksi. dengan menggunakan query statement berikut:

```
1 UNION SELECT 'test',NULL,NULL
```



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

Query diatas mencoba untuk menyisipkan data string 'test' ke dalam hasil kueri bersama dengan dua nilai NULL. Kita mencoba untuk menemukan kolom yang dapat diinjeksi dengan menambahkan hasil UNION SELECT ke dalam kueri asli.

Query berhasil dieksekusi, kita melihat nilai 'test' ditampilkan di dalam salah satu bagian dari halaman web, menandakan bahwa kolom tersebut dapat diinjeksi.

4. Langkah 4

- g. Kita bisa menampilkan informasi terkait database, dengan mengganti 'test' menjadi @@hostname untuk menampilkan nama host dari DB, database() untuk nama database, @@version untuk menampilkan versi dari DB.

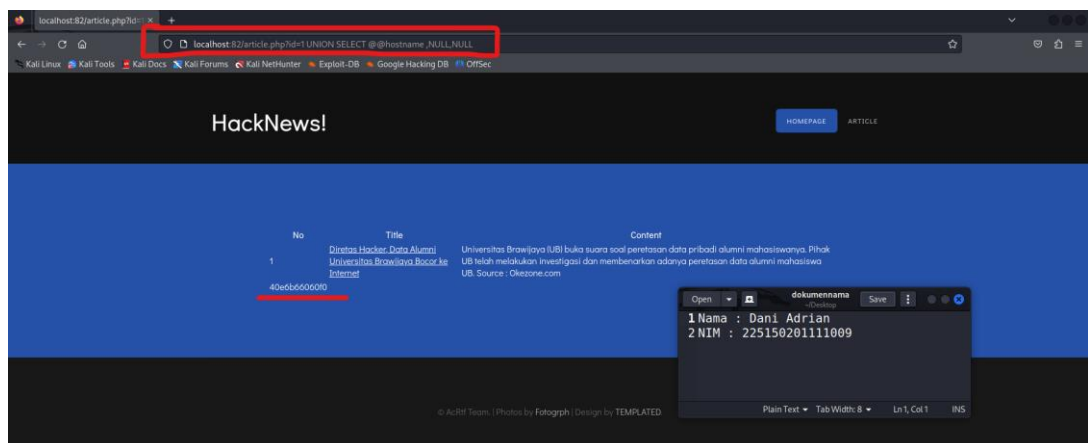




**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

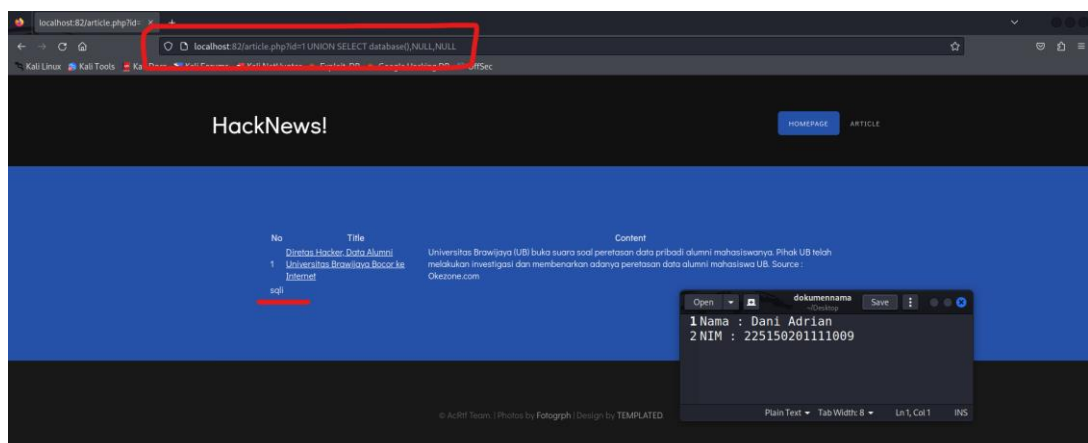
BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

Hostname :



Query diatas untuk menampilkan nama host dari database

Database :



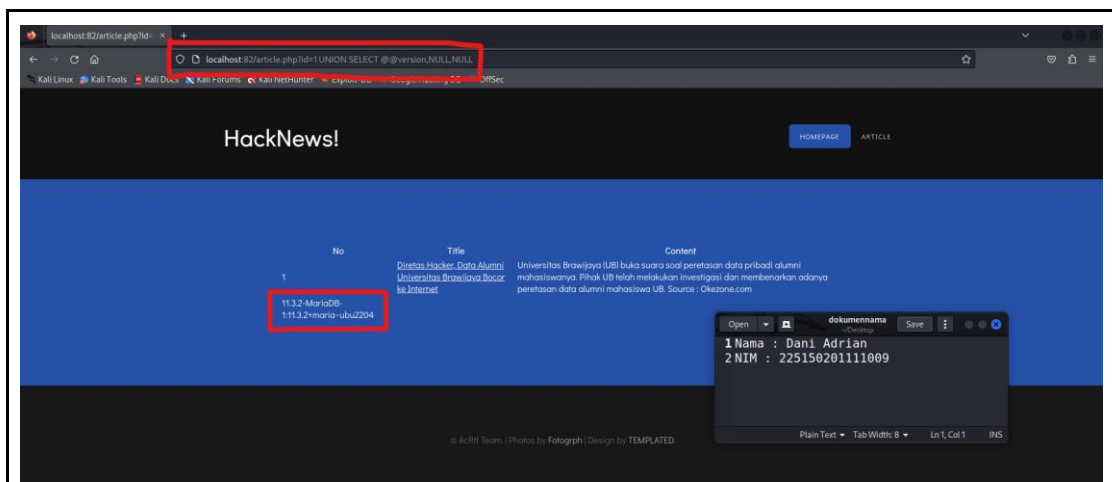
Query diatas untuk menampilkan nama database

Versi :



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas



Query diatas untuk menampilkan versi dari database

Hostname : 1 UNION SELECT @@hostname, NULL, NULL

Database : 1 UNION SELECT database(), NULL, NULL

Versi : 1 UNION SELECT @@version, NULL, NULL

5. Langkah 5

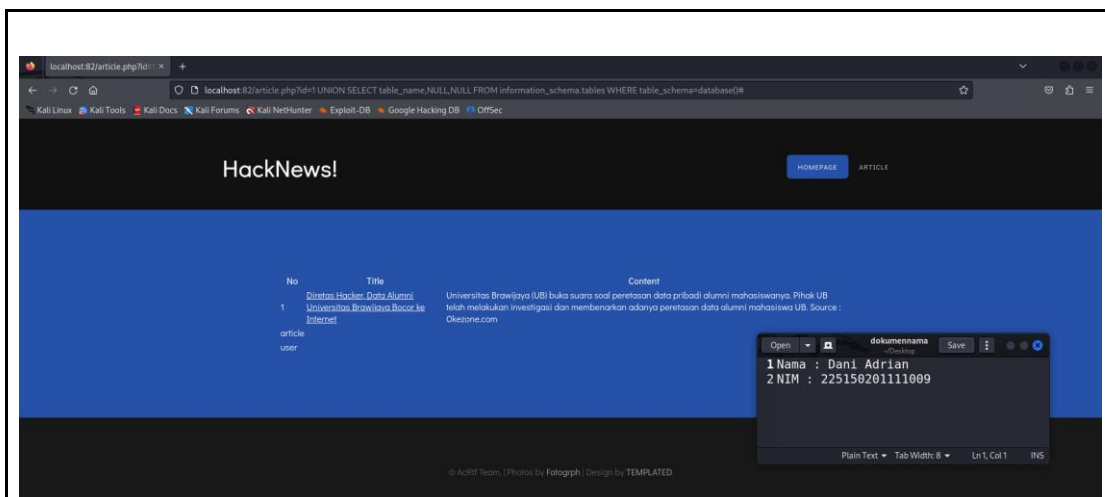
- h. Berikutnya tampilkan table yang dimiliki database dengan menggunakan query berikut:

```
1 UNION SELECT table_name, NULL, NULL FROM
information_schema.tables WHERE
table_schema=database() #
```



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas



Dalam query, kita menggunakan `UNION SELECT` untuk menyisipkan query ke dalam hasil yang ada. Kita memilih kolom `table_name` dari tabel `information_schema.tables` di mana `table_schema` sama dengan database yang sedang digunakan.

Kemudian kita memilih `user`.

## 6. Langkah 6

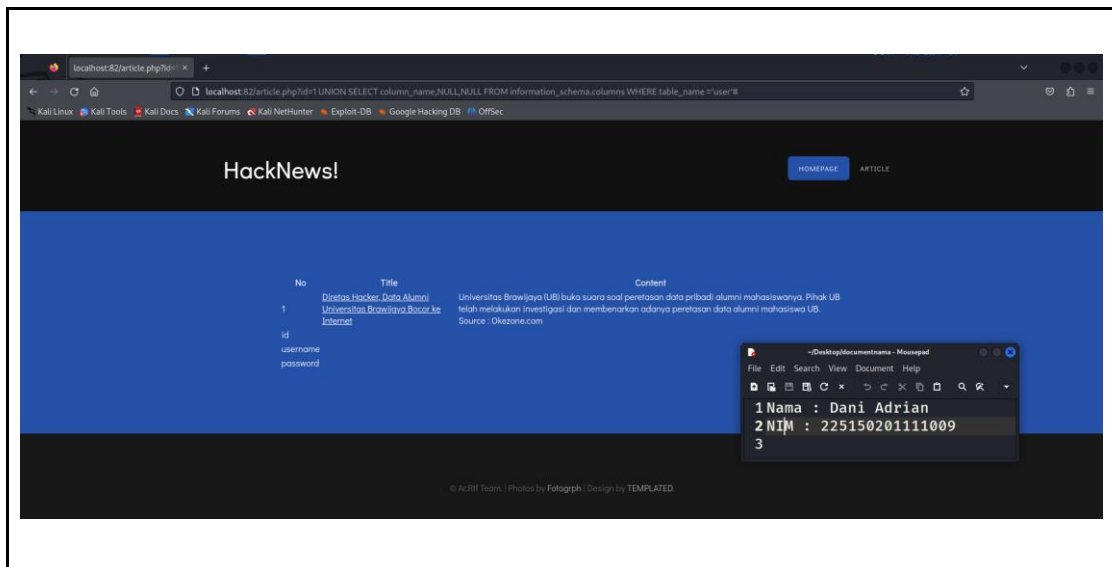
- i. Dari table yang telah Anda ekstraksi, tampilkan column dari table yang memiliki kemungkinan memuat username dan password dari admin. Dengan menggunakan query berikut, dan ganti nama\_table menjadi nama tabel yang Anda temukan.

```
1 UNION SELECT column_name,NULL,NULL FROM
    information_schema.columns
    WHERE table_name='nama_table' #
```



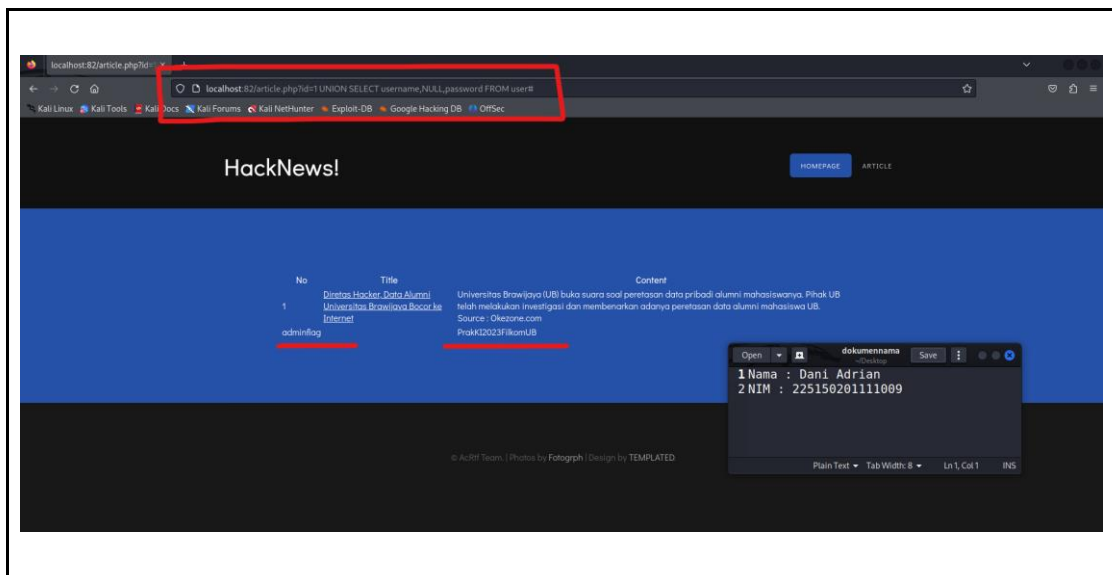
**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas



- j. Setelah Anda mengetahui nama kolom dari sebuah table, tampilkan value dari kolom tersebut. Dengan menggunakan query berikut, dan ganti kolom1 dan kolom2 sesuai dengan column yang telah Anda temukan:

```
1 UNION SELECT kolom1,NULL,kolom2 FROM nama_table#
```





**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

```
1 UNION SELECT username,NULL,password FROM user#
```

Query diatas bertujuan untuk mengganti kolom1 dan kolom2 dengan username dan password, dan ganti nama\_table dengan user.

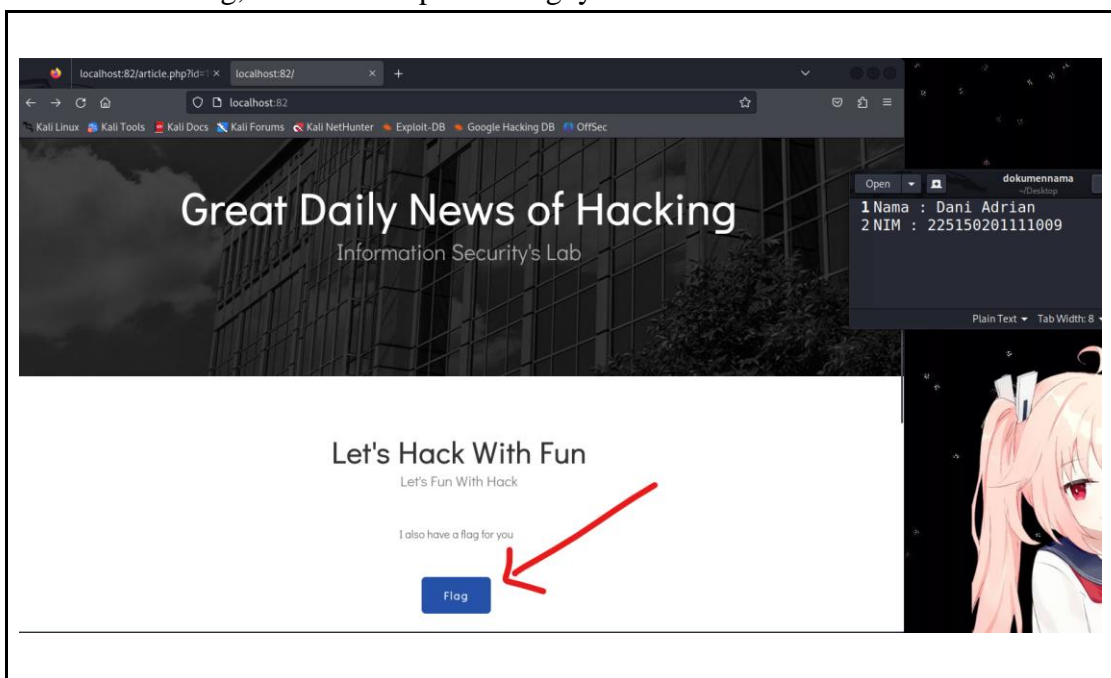
Kemudian kita akan mendapatkan username dan password nya, yaitu :

Username : adminflag

Password : PrakKI2023FilkomUB

## 7. Langkah 7

- k. Setelah Anda berhasil menemukan kredensial dari admin lakukan login pada laman flag, untuk mendapatkan flagnya.





**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

Akses halaman login untuk laman flag.

Gunakan kredensial admin (username & password) yang telah ditemukan untuk login ke laman flag Masukkan username dan password pada formulir login, dan tekan tombol login.

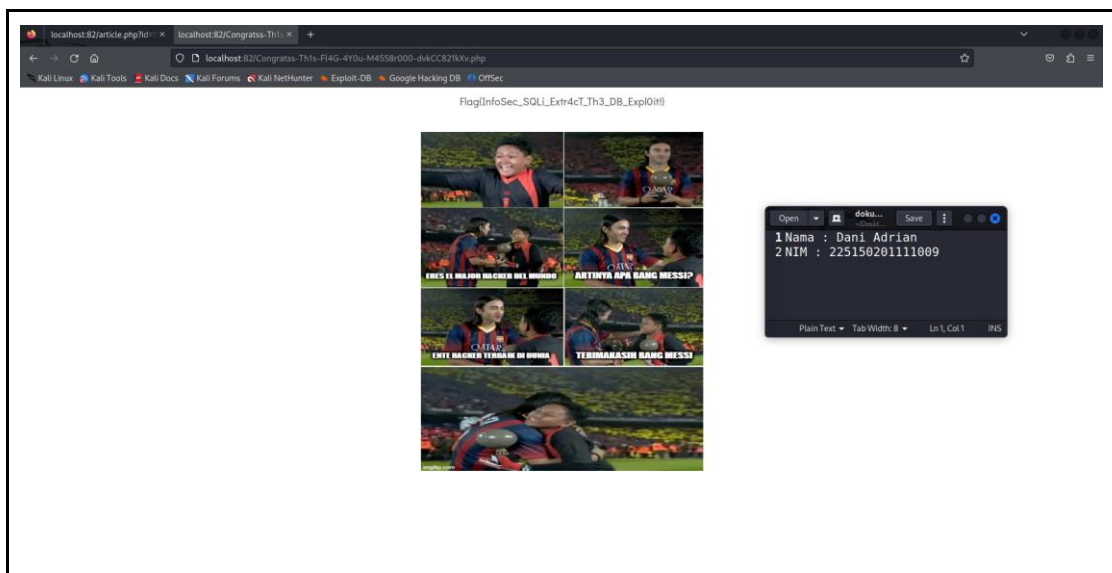
Dapatkan Flag

Flag :



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

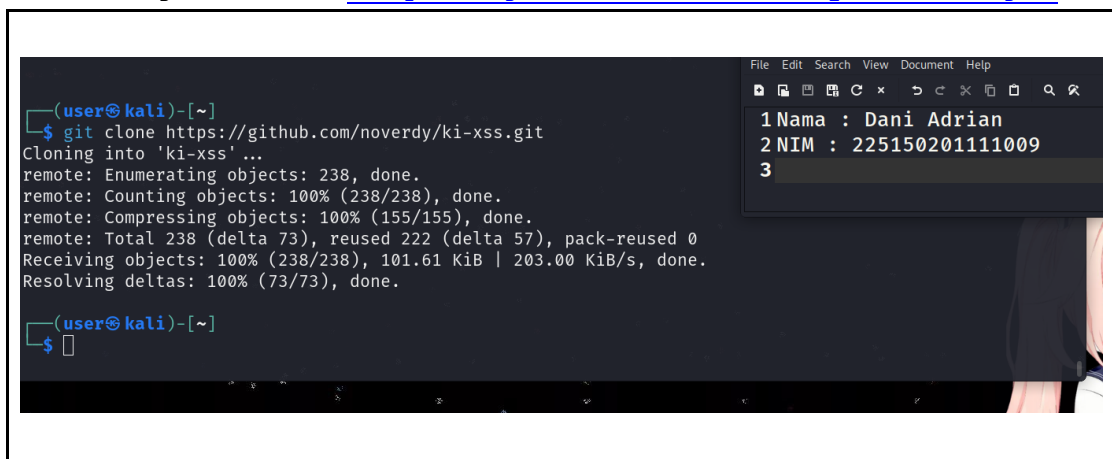
BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas



### Stored XSS (Cross-site Scripting)

#### 1. Langkah 1

- Buka terminal pada sistem operasi Linux.
- Kemudian lakukan cloning github repository untuk lab praktikum XSS:  
`git clone https://github.com/noverdy/ki-xss.git`







**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA**

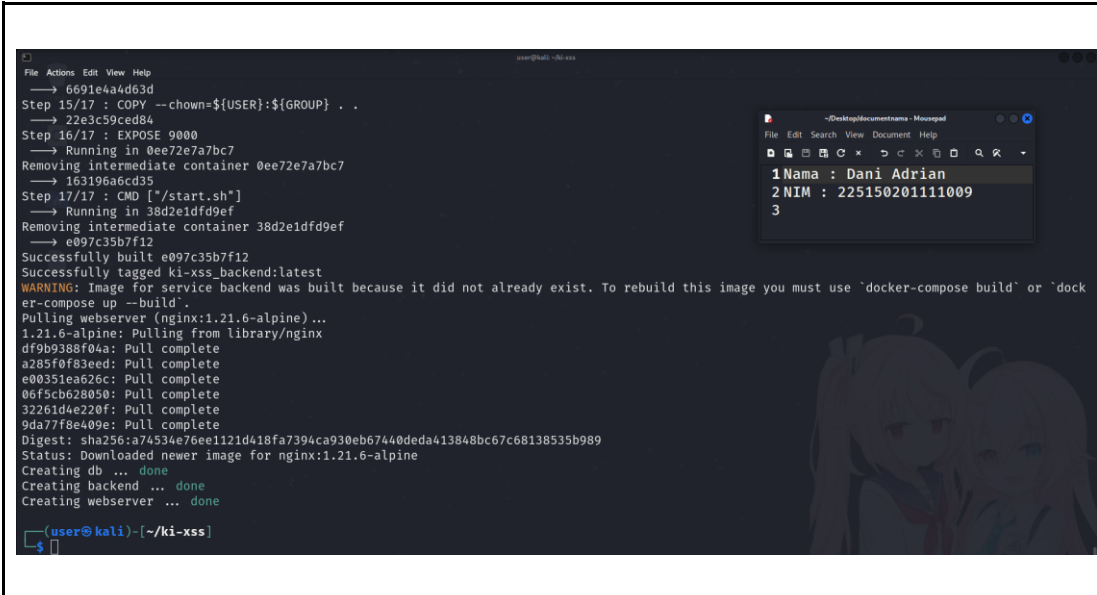
BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

c. Masuk ke dalam direktori ki-xss dengan perintah:

`cd ki-xss`

d. Melakukan build docker untuk sqli-part1 dengan perintah:

`sudo docker-compose up -d`



```
File Actions Edit View Help
→ 6691e4a4d63d
Step 15/17 : COPY --chown=${USER}:${GROUP} . .
→ 22e3c59ced84
Step 16/17 : EXPOSE 9000
→ Running in 0ee72e7a7bc7
Removing intermediate container 0ee72e7a7bc7
→ 163196a6cd35
Step 17/17 : CMD ["/start.sh"]
→ Running in 38d2e1dfd9ef
Removing intermediate container 38d2e1dfd9ef
→ e097c35b7f12
Successfully built e097c35b7f12
Successfully tagged ki-xss_backend:latest
WARNING: Image for service backend was built because it did not already exist. To rebuild this image you must use `docker-compose build` or `dock
er-compose up --build`.
Pulling webserver (nginx:1.21.6-alpine)...
1.21.6-alpine: Pulling from library/nginx
df9b9388f04a: Pull complete
a285f0f83eed: Pull complete
e00351ea026c: Pull complete
06f5cb628050: Pull complete
32261d4e220f: Pull complete
9da77f8e409e: Pull complete
Digest: sha256:a74534e76ee1121d418fa7394ca930eb67440deda413848bc67c68138535b989
Status: Downloaded newer image for nginx:1.21.6-alpine
Creating db ... done
Creating backend ... done
Creating webserver ... done
(user@kali)~/ki-xss
$
```

e. Jalankan perintah berikut untuk mengecek apakah service sudah berjalan:

`sudo docker ps`

f. Apabila berhasil maka service akan berjalan pada port 1337, sehingga dapat mengakses pada browser dengan alamat localhost:1337



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

The screenshot displays a Kali Linux terminal window with the command `sudo docker ps` executed, showing a list of running containers. A text box in the top right corner contains the user's name and NIM. Below the terminal, a web browser window shows the 'Notes App' interface, which includes a login button and a welcome message.

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS
AMES				
b02dd8b4b943	nginx:1.21.6-alpine	"/docker-entrypoint..."	3 minutes ago	Up 3 minutes
ebsserver				
d03a27798448	ki-xss_backend	"docker-php-entrypoi..."	3 minutes ago	Up 3 minutes
ackend				
19249d9b5230	mariadb:10	"docker-entrypoint.s..."	3 minutes ago	Up 3 minutes (healthy)
b				
c50b705865e5	sqli-part2	"docker-php-entrypoi..."	26 minutes ago	Up 26 minutes
qli-part2-app				
3a8ce06623fd	mariadb:latest	"docker-entrypoint.s..."	26 minutes ago	Up 26 minutes
qli-part2-db				
f213f405ec0e	sqli-part1	"docker-php-entrypoi..."	27 minutes ago	Up 27 minutes
qli-part1-app				
0f29b992c833	mariadb:latest	"docker-entrypoint.s..."	27 minutes ago	Up 27 minutes
qli-part1-db				

Notes App

Welcome to Notes App

Login to create your note or explore other people's notes!

2. Langkah 2

- Pertama lakukan pengujian HTML Injection pada bagian form Create Notes, berikan teks dengan ukuran h2 bertuliskan NIM Anda.



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

Dengan menyisipkan tag HTML seperti `<h2>NIM Anda</h2>` pada form Create Notes, kita dapat menguji apakah aplikasi membiarkan tag HTML tersebut dieksekusi dan ditampilkan sebagai HTML yang sebenarnya di halaman web.

- b. Berikutnya, Masukkan komentar dengan potongan kode berikut:
- ```
<script>alert("NIM ANDA")</script>
```



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

Mengapa hal tersebut dapat terjadi?

The screenshot displays a web browser window with the address bar showing 'localhost:1337/posts/stored-JacB'. The main content area is dark, and a small message box is visible in the center, indicating a successful update. Below the browser window, the 'Notes App' interface is shown, featuring a green notification bar that says 'Note updated successfully.' and a note titled 'stored' with the content '1 Nama : Dani Adrian', '2 NIM : 225150201111009', and '3'. The note is dated '2024-03-17 18:13:59' and is set to 'Public'. Below the note, there are 'Edit' and 'Delete' buttons. At the bottom right, a small window titled 'Mousepad' is open, showing the same text as the note: '1 Nama : Dani Adrian', '2 NIM : 225150201111009', and '3'.

Tujuan dari langkah ini adalah untuk melihat apakah aplikasi rentan terhadap serangan Cross-Site Scripting (XSS), di mana kode JavaScript yang tidak aman dieksekusi oleh browser pengguna.

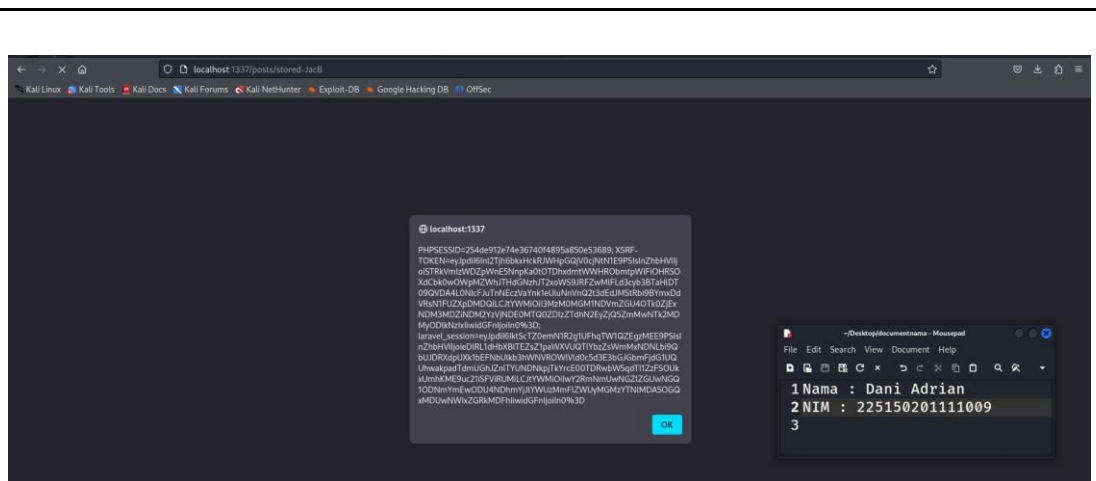


**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA**

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

- c. Berikutnya, kita akan menampilkan session cookie milik kita dengan kode berikut:

```
<script>alert (document.cookie)</script>
```



Browser secara otomatis mengirimkan session cookie ketika memuat halaman web yang diminta. Kode JavaScript yang dieksekusi di dalam halaman web memiliki akses ke session cookie tersebut karena cookie merupakan bagian dari informasi yang tersedia dalam lingkungan klien (browser). Dengan menggunakan document.cookie, kode JavaScript dapat mengakses dan menampilkan session cookie yang terkait dengan situs web yang sedang diakses.

## Kesimpulan

Praktikum ini menegaskan pentingnya memahami dan menguji kerentanan keamanan dalam aplikasi web. Dengan memahami cara kerja serangan seperti SQL Injection, kita dapat mengidentifikasi, mencegah, dan memperbaiki celah keamanan yang mungkin ada. Penggunaan alat-alat seperti Docker dan teknik-teknik khusus seperti SQL Injection membantu dalam menguji dan mengevaluasi keamanan aplikasi web. Melalui praktikum ini, kita diajarkan untuk memikirkan keamanan sejak awal dalam pengembangan perangkat lunak. Dengan



**LABORATORIUM PEMBELAJARAN ILMU KOMPUTER  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA**

---

BAB : KERENTANAN DAN ANCAMAN  
NAMA : DANI ADRIAN  
NIM : 225150201111009  
TANGGAL : 14/03/2024  
ASISTEN : Bernas Cakra Sakti Harisna  
Mohammad Seto Aji Pamungkas

---

mengidentifikasi dan memperbaiki celah keamanan sejak dini, kita dapat mencegah serangan potensial dan melindungi data sensitif dari akses yang tidak sah. Praktikum ini juga menjadi indikator untuk mengukur pemahaman peserta tentang konsep keamanan informasi secara umum. Kemampuan mereka dalam mengidentifikasi dan memperbaiki celah keamanan akan mencerminkan tingkat pemahaman mereka tentang konsep tersebut. Dengan demikian, praktikum ini memberikan wawasan yang berharga tentang celah keamanan yang mungkin ada dalam aplikasi web dan pentingnya mengimplementasikan langkah-langkah perlindungan yang tepat untuk melindungi data dan privasi pengguna.

### **Evaluasi**

Kesuksesan praktikum kali ini bergantung pada kesiapan dan keterampilan peserta dalam mengikuti instruksi dan menerapkan teknik-teknik yang diajarkan ketika praktikum dengan benar. Selain itu, penggunaan teknologi seperti Docker dan metode-metode khusus seperti SQL Injection juga perlu dievaluasi. Kemampuan peserta dalam menggunakan alat dan teknik ini dengan benar akan menghasilkan keberhasilan praktikum yang baik dan tepat. Evaluasi juga mencakup pemahaman peserta tentang konsep keamanan informasi secara umum. Kemampuan mereka dalam mengidentifikasi dan memperbaiki celah keamanan akan mencerminkan tingkat pemahaman mereka tentang konsep tersebut.

