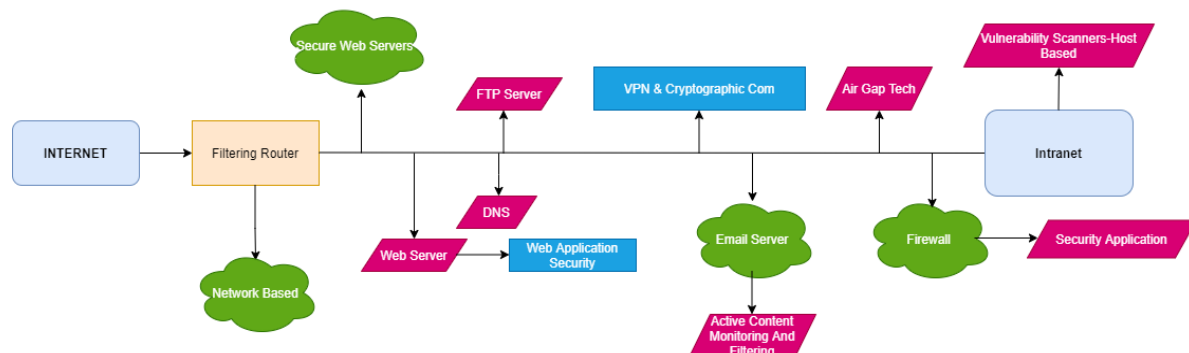


A.



Penjelasannya dimana desain dari akses dari pc public area yang membutuhkan koneksi internet untuk mengakses garçon seperti web garçon dispatch garçon dan ftp garçon, pertama melalui filtering yang ada pada router lalu berhadapan dengan firewall guna mencegah kerentanan jaringan. Pada pc yang berada original area yang tidak koneksi internet juga petlu berhadapan dengan firwal agar mencegah krentanan jaringan.

B.Membuat sebuah logic dengan menggunakan Snippet Code:

- Menvalidasi Input yang pertama apabila memungkinkan dimana verifikasi data yang diberikan oleh pengguna terhadap daftar putih tersebut.

```
if (is_numeric($id) == true) { ... }
```

- Menyiapkan kueri ddengan membuat kueri kita dengan menggunakan nama parameter yang diawali dengan titik dua sebagai placeholder.

```
$q = "SELECT username FROM users WHERE id = :id";
```

- Membuat pernyataan yang disiapkan pada SQL

```
$sth = $dbh->prepare($q);
```

**Sofian Raffi**  
**20190801277**

## **UTS KEAMANAN INFORMASI**

- Mengikat parameter dimana kita mengikat parameter ke kueri nya, misalnya dimana kita mengikat nilai pada ID.

```
$sth->bindParam(':id', $id);
```

- Menjalankan Kueri dimana setelah kita melewati parameter, kita dapat menjalankan kueri yang kita buat tadi.

```
$sth->execute();
```

- Langkah Keenam (Mengambil hasilnya) selanjutnya dimana setelah kita menjalankan kueri tersebut, kita dapat mengambil hasilnya untuk digunakan lebih lanjut.

```
$result = $sth->fetchColumn();
```

C. Pengujian terhadap Sense yaitu dengan Salah satu cara yaitu mencegah injeksi SQL adalah dengan menggunakan kueri parameter yang mudah dan mencegah dan/ atau memperbaiki kerentanan injeksi SQL, mulailah dengan membaca rekomendasi di In- Depth Defense Kueri berparameter diperlukan untuk membedakan antara kueri dan data yang dimasukkan oleh pengguna. Hal ini memungkinkan database untuk lebih mudah membedakan antara entri pengguna dan perintah database. Jika penyerang memasukkan perintah kueri, sistem akan memperlakukannya sebagai input pengguna dan sebagai perintah SQL yang tidak valid. Maka database tidak akan menjalankan perintah sql. Apa pun yang dimasukkan oleh pengguna diperlakukan oleh database sebagai data pengguna, bukan sebagai perintah SQL. Database akan diperlakukan sebagai data dan tidak akan pernah disalahartikan sebagai bagian dari perintah.

D.

```
<?php
```

```
$config = array(
```

**Sofian Raffi**  
**20190801277**

## **UTS KEAMANAN INFORMASI**

```
'config'      => 'F:/UTSKI/htdocs/UTSKI/openssl.cnf',

'default_md' => 'sofianr10',

'private_key_bits'      => '1717',

'private_key_type'      => 'OPENSSL_KEYTYPE_RSA',

'input_password'        => 'Artlas17',

'output_password'       => 'Artlas17',

);

// key Pair
$keypair = openssl_pkey_new($config);

// Private Key
openssl_pkey_export($keypair, $privKey, null, $config);

//Public Key
$publickey = openssl_pkey_get_details($keypair);
$pubKey = $publickey['key'];

?>

<html>

<head>

    <meta chaset="utf-8">
```

**Sofian Raffi**  
**20190801277**

## **UTS KEAMANAN INFORMASI**

<title>Snippet Code</title>

<head>

<body>

<textarea cols="120" rows="50"><?php echo \$privKey?></textarea>

<textarea cols="120" rows="50"><?php echo \$pubKey  
?></textarea>

</body>

</html>

D. Untuk mengupload proyek yang kita miliki, langkah pertama masuk ke Github

Setelah masuk, kemudian tekan tombol dengan ikon *plus* (+) di area pojok kanan atas sebelah foto profil dan pilih **New Repository**.

Kemudian kita akan diarahkan ke halaman pembuatan repositori. Ikuti langkah-langkah pengisian di bawah ini.

Jika pengisian sudah dilakukan maka silahkan tekan tombol **Create repository**  
Sebenarnya saat selesai membuat repositori kita akan diarahkan ke halaman petunjuk atau tata cara selanjutnya untuk mengupload file program ke GitHub.

Namun di sini kita akan pakai cara yang sedikit singkat saja.

Lanjut, sekarang kita beralih ke folder proyek yang akan diupload. Masuk ke dalam folder tersebut melalui terminal dan pastikan di dalam folder tersebut terdapat file program yang akan diupload nantinya.

```
$ cd hello-world/
```

Bash

Copy

Buat repo lokal dengan perintah

```
$ git init
```

Bash

Copy

**Sofian Raffi**  
**20190801277**

## **UTS KEAMANAN INFORMASI**

Perintah tersebut akan menghasilkan hidden folder bernama `.git` yang berguna untuk menyimpan semua data terkait *version control*. Untuk menampilkan hidden folder ini tekan kombinasi keyboard **Shift+H**

Kemudian tambahkan file baru ke repositori.

```
$ git add hello-world.py
```

Bash

Copy

Jika terdapat banyak file maka bisa gunakan

```
$ git add .
```

Bash

Copy

Simpan perubahan dan beri penjelasan atau komentar mengenai file tersebut.

```
$ git commit -m "Upload file"
```

Bash

Copy

Tambahkan URL remote dari repositori, untuk mendapatkan url remote ini kalian harus buka kembali halaman terakhir setelah repo selesai dibuat.

Pada bagian atas akan terlihat sebuah url yang mana url tersebutlah yang akan kita gunakan.

```
$ git remote add origin https://github.com/LendisFabri/hello-world.git
```

Bash

Copy

Kemudian upload folder projek ke github

```
$ git push -u origin main
```

Bash

Copy

Klik file `hello-world.py` untuk melihat kode di dalamnya.

Selamat sekarang kalian berhasil mengupload program kalian ke GitHub.