

# Pemrograman Web Berbasis Framework



## Pertemuan 7 : Keamanan Aplikasi

Hasanuddin, S.T., M.Cs.

Prodi Teknik Informatika UAD

[hasan@uad.ac.id](mailto:hasan@uad.ac.id)



# Pokok Bahasan

- Pendahuluan
- Penanganan Error Reporting
- Antisipasi serangan XSS
- Validasi Form
- Pengamanan System Core Framework
- Autentikasi dengan Session
- Pengamanan pada konfigurasi httpd.conf

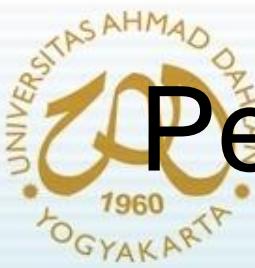
TIK :

Setelah mengikuti kuliah ini mahasiswa dapat mengetahui dan memahami faktor keamanan aplikasi pada Web Framework



# Pendahuluan

- Secara umum suatu Framework telah dilengkapi fasilitas yg memudahkan pembuatan aplikasi termasuk dalam hal keamanan aplikasi.
- Faktor/celah serangan terhadap aplikasi :
  - Error reporting
  - Halaman login
  - Form input
  - Alamat URL
  - Konfigurasi server



# Penanganan Error Reporting

- Buka file index.php pada root folder
- Ganti baris program berikut :

```
error_reporting(E_ALL);
```

- Menjadi :

```
error_reporting(0);
```

- Buka file error\_db.php pada : system/application/errors/
- Ganti baris program berikut :

```
<h1><?php echo $heading; ?></h1>
<?php echo $message; ?>
```

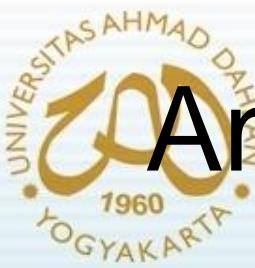
- Menjadi :

```
<h1><?php //echo $heading; ?></h1>
<?php echo //$/message; ?>
```



# Antisipasi Serangan XSS

- XSS Filtering digunakan untuk mencegah kode-kode jahat yg jika masuk ke sistem akan menyebabkan error sehingga menimbulkan celah masuk ke sistem secara ilegal.
- Buka file config.php pada : system/application/config/
- Ganti baris program berikut :  
  
`$config['global_xss_filtering'] = FALSE;`
- Menjadi :  
  
`$config['global_xss_filtering'] = TRUE;`



# Antisipasi Serangan XSS (2)

- Jangan biasakan gunakan fungsi `$_POST($variabel)`, ganti dengan `$this->input->post($variabel, TRUE)`
- Contoh :

`$data['lihat'] = $_POST['dataku'];`

- Ubah menjadi :

`$data['lihat'] = $this->input->post('dataku', TRUE)`



# Validasi Form

- Validasi form digunakan untuk meminimalkan kesalahan input (termasuk antisipasi serangan melalui form atau **SQL Injection**)
- Untuk menjalankan validasi form harus mengaktifkan library form\_validation, melalui :
  - Autoload.php
    - \$autoload['libraries'] = array('form\_validation');
  - Secara manual
    - \$this->load->library('form\_validation');



# Validasi Form (2)

- Pengaturan manajemen proses antara controller, model dan view
- Contoh :

```
<?php
class Form extends Controller {
    function index() {
        if ($this->form_validation->run() == FALSE)
        {
            $this->load->view('myform');
        }
        else {
            $this->load->view('formsuccess');
        }
    }
?>
```



# Validasi Form (3)

- Penggunaan validation\_rule
- Format :

**\$this->form\_validation->set\_rules();**

- Contoh :

```
$this->form_validation->set_rules('username', 'Username', 'required');  
$this->form_validation->set_rules('password', 'Password', 'required');  
$this->form_validation->set_rules('passconf', 'Password Confirmation',  
'required');  
$this->form_validation->set_rules('email', 'Email', 'required');
```



# Validasi Form (4)

- Penggunaan Cascading Rule
- Merupakan penggunaan multi rule
- Contoh :

```
$this->form_validation->set_rules('username', 'Username',  
'required|min_length[5]|max_length[12]');  
$this->form_validation->set_rules('password', 'Password',  
'required|matches[passconf]');  
$this->form_validation->set_rules('passconf', 'Password Confirmation',  
'required');  
$this->form_validation->set_rules('email', 'Email', 'required|valid_email');
```



# Validasi Form (5)

- Penggunaan Fungsi **Callback**
- Merupakan penggunaan fungsi khusus untuk validasi
- Contoh :

```
$this->form_validation->set_rules('username', 'Username',  
'callback_username_check');
```

- Akan menjalankan fungsi :

```
function username_check($str) {  
    if ($str == 'test') {  
        $this->form_validation->set_message('username_check', 'The %s  
field can not be the word "test"');  
        return FALSE;  
    } else {  
        return TRUE;  
    }  
}
```



# Pengamanan System Core Framework

- Pindahkan posisi folder system (tanpa mengikutsertakan folder Application) di atas dari root folder
- Awal :
  - Root folder
    - system
    - application
- Menjadi :
  - Root folder
    - system
    - application



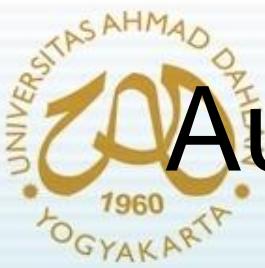
# Pengamanan System Core Framework (2)

- Agar folder sistem dapat dikenali, buka file index.php di root folder
- Ubah baris program berikut :

`$system_folder = "system";`

- Menjadi :

`$system_folder = “.../system”;`



# Autentikasi dengan Session

- Penggunaan umum Session dalam hal layanan login pengguna
- Untuk menggunakan layanan session harus diaktifkan library dengan cara :
  - Autoload.php
    - `$autoload['libraries'] = array('session');`
  - Secara manual
    - `$this->load->library('session');`



# Autentikasi dengan Session (2)

- Setelah login dan account benar, data disimpan pada session dengan perintah :  
`set_userdata();`

- Contoh :

```
$newdata = array(  
    'username' => $data['user'],  
    'status' => 'ok');  
  
$this->session->set_userdata($newdata);
```



# Autentikasi dengan Session (3)

- Setiap akan masuk ke prosedur yg butuh otorisasi, maka isi session dicek apa sesuai atau tidak
- Contoh :

```
$status = $this->session->userdata('status');
If (!isset($status) || $status != 'ok') {
    $this->loginulang();
} else {
    $this->inbox();
}
```



# Autentikasi dengan Session (4)

- Jika user keluar dari accountnya, maka data session harus dihapus.
- Contoh :

```
$newdata = array(  
    'username' => ' ', 'status' => '');
```

```
$this->session->unset_userdata($newdata);
```



# Pengamanan pada konfigurasi httpd.conf

- Buka file httpd.conf (misal di : C:/xampp/apache/conf/)
- Ubah baris program :  
`#LoadModule rewrite_module modules/mod_rewrite.so`
- Menjadi :  
`LoadModule rewrite_module modules/mod_rewrite.so`
- Buka file config.php di folder system/application/config/
- Ubah baris program :  
`$config['index_page'] = "index.php";`
- Menjadi :  
`$config['index_page'] = "";`



# Pengamanan pada konfigurasi httpd.conf (2)

- Buat file .htaccess pada root direktori CI (sejajar dengan file index.php)
- Masukkan kode program berikut :

RewriteEngine On

RewriteCond % {REQUEST\_FILENAME} !-f

RewriteCond % {REQUEST\_FILENAME} !-d

RewriteRule ^(.\*)\$ index.php/\$1 [L]



## Referensi :

Wardana, ***Menjadi Master PHP dengan Framework CodeIgniter***, Elexmedia Komputindo, Jakarta, 2010.