eZ systems

Share your information

Welcome!

eZ systems

Share your information

What is Wrong Here?

```
<?php
    $sql = "
        SELECT card_num, card_name, card_expiry
        FROM credit_cards
        WHERE uid = '{$_GET['uid']}'
    ";
?>
```

```
http://example.com/script.php?uid=42
```

```
SELECT card_num, card_name, card_expiry
FROM credit_cards
WHERE uid = '42'
```

!

```
http://example.com/script.php?uid=42'%20or%20''='
```

```
SELECT card_num, card_name, card_expiry
FROM credit_cards
WHERE uid = '42' or ''=''
```

What is Wrong Here?

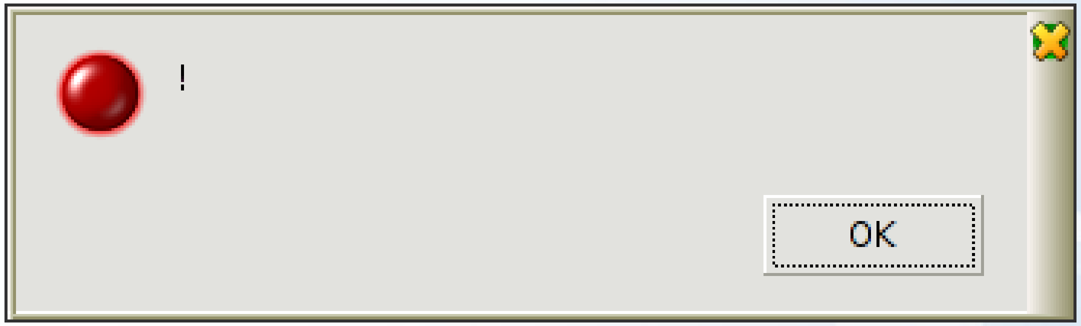
```
<html>
    <head>
        <title>Example</title>
    </head>
    <body>
        Name: <?php echo $_GET['name']; ?>
    </body>
</html>
```

```
http://example.com/script.php?name=derick
```

Name: derick

!

```
http://example.com/script.php?name=<script>alert('!');</script>
```



Casting

```
<?php
    $uid = (int) $_GET['uid'];
    $sql = "
        SELECT card_num, card_name, card_expiry
        FROM credit_cards
        WHERE uid = '{$uid}'
    ";
?>
```

```
http://example.com/script.php?uid=42
```

```
SELECT card_num, card_name, card_expiry
FROM credit_cards
WHERE uid = '42'
```

:-)

```
http://example.com/script.php?uid=42'%20or%20''='
```

```
SELECT card_num, card_name, card_expiry
FROM credit_cards
WHERE uid = '42'
```

eZ systems

Share your information

Filtering

```
<html>
    <head>
        <title>Example</title>
    </head>
    <body>
        Name: <?php echo htmlentities($_GET['name']); ?>
    </body>
</html>
```

```
http://example.com/script.php?name=derick
```

Name: derick

:-)

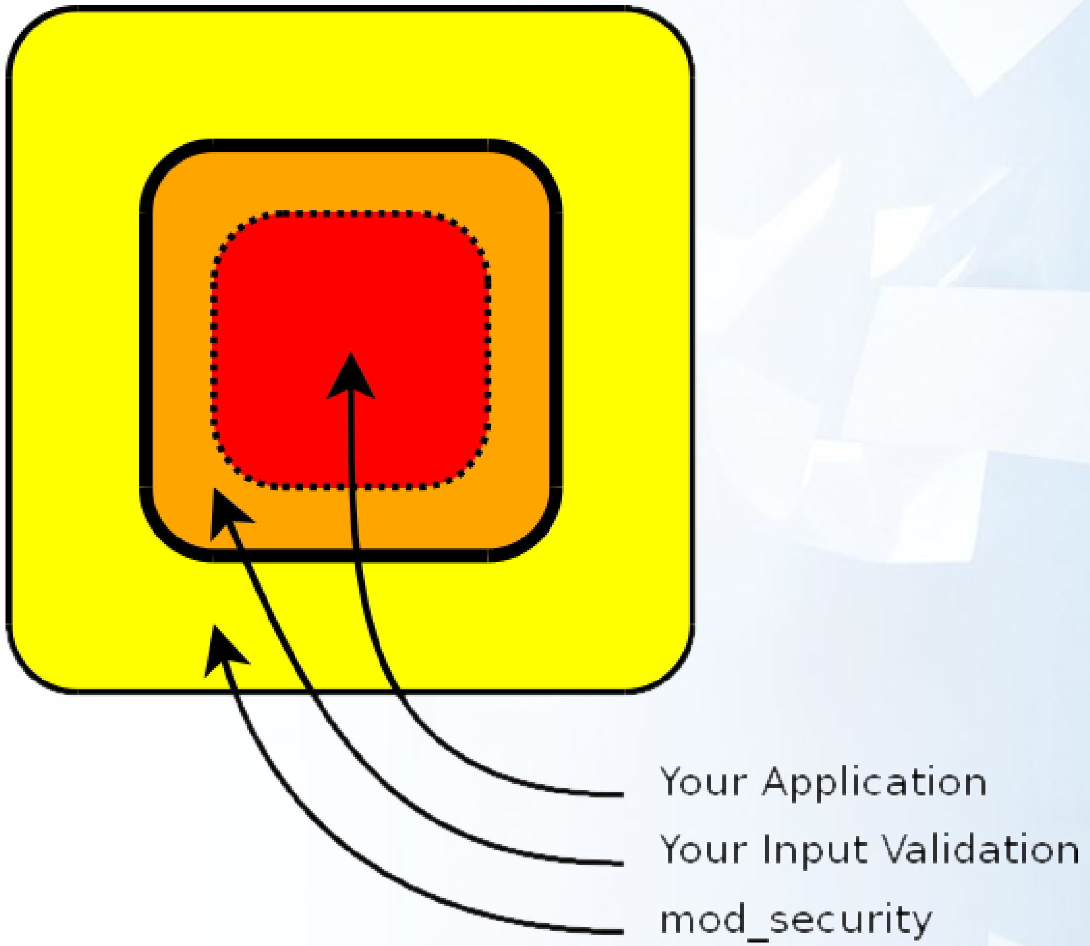
```
http://example.com/script.php?name=<script>alert('!');</script>
```

&lt;script>alert('!');&lt;/script>

eZ systems

Share your information

Multiple Levels of Defense



eZ systems

Share your information

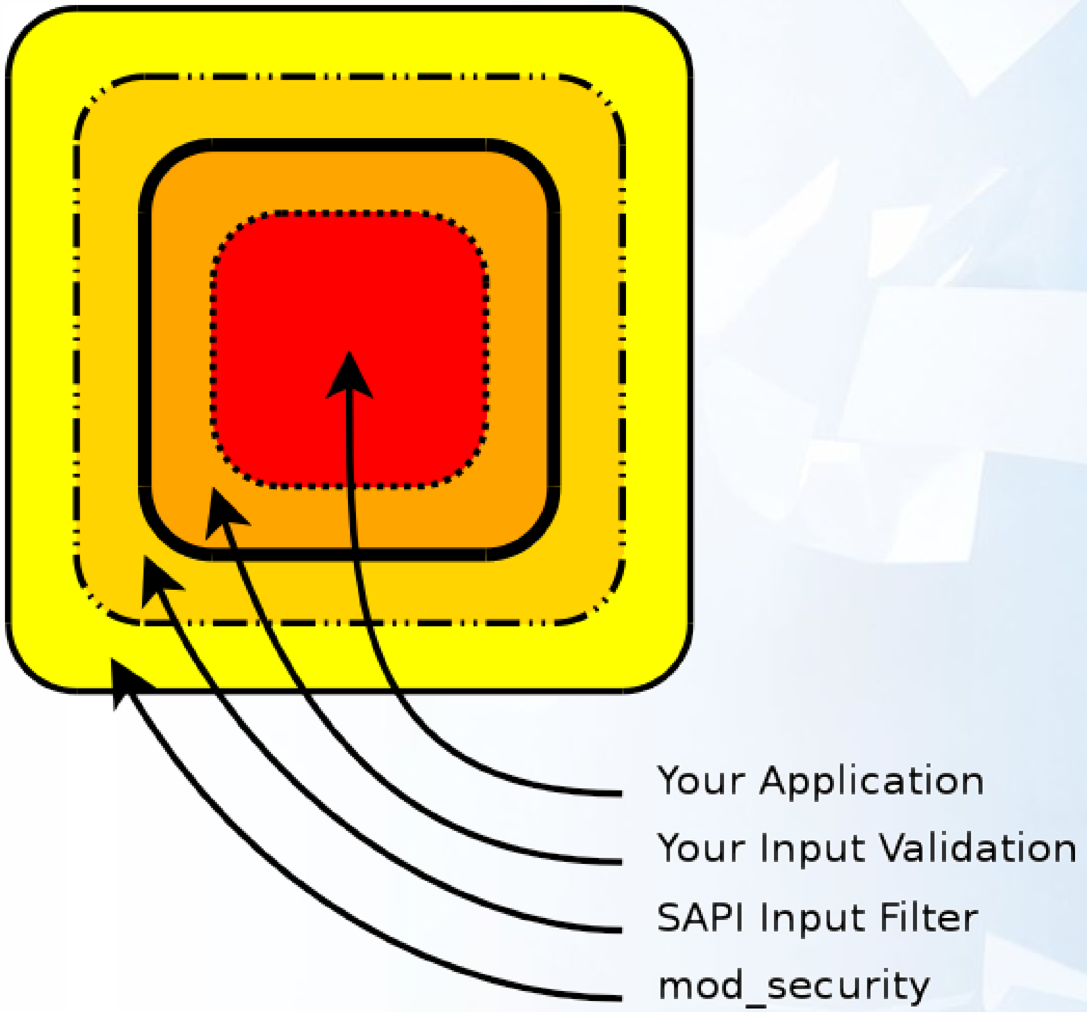
mod_security

- Apache module
- Can be used to avoid certain attack
- Is not PHP specific

eZ systems

Share your information

Multiple Levels of Defense++



eZ systems

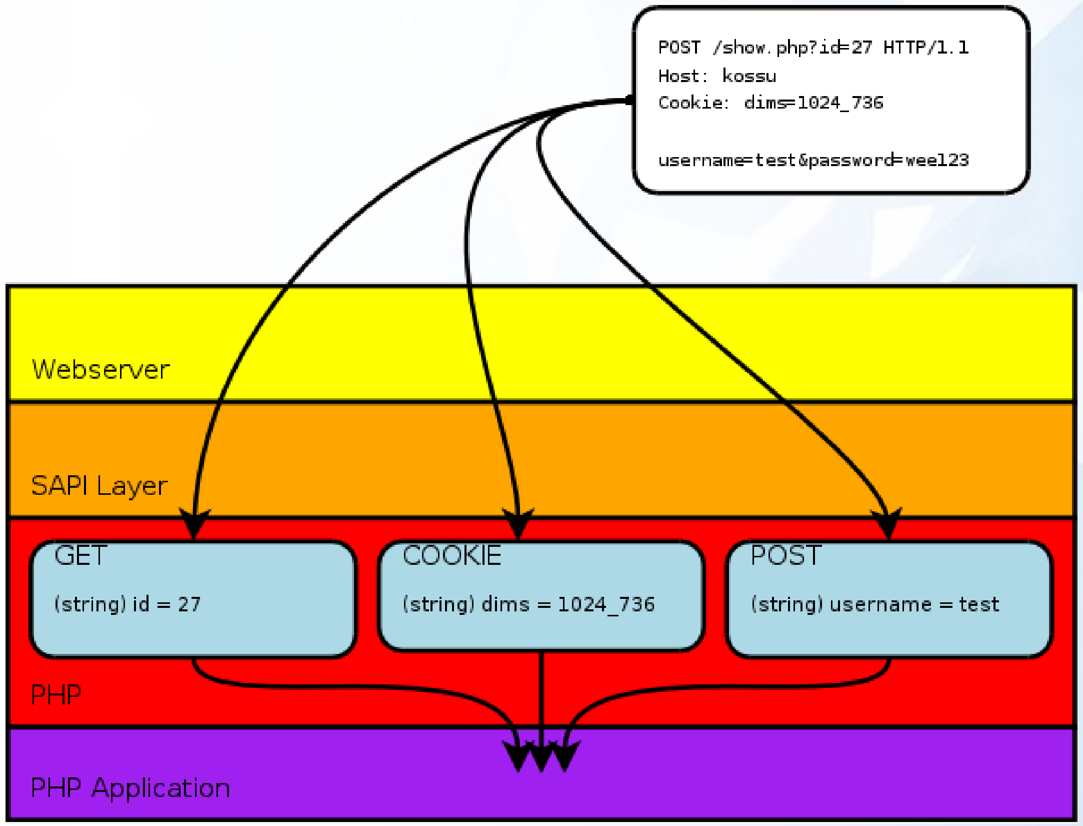
Share your information

SAPI Input filter

- Sits between PHP and the webserver
- Is used while fetching data from users sources
- Can be used to filter data
- Prohibit data from entering PHP
- Written as a C extension to PHP
- Server wide filter

eZ systems

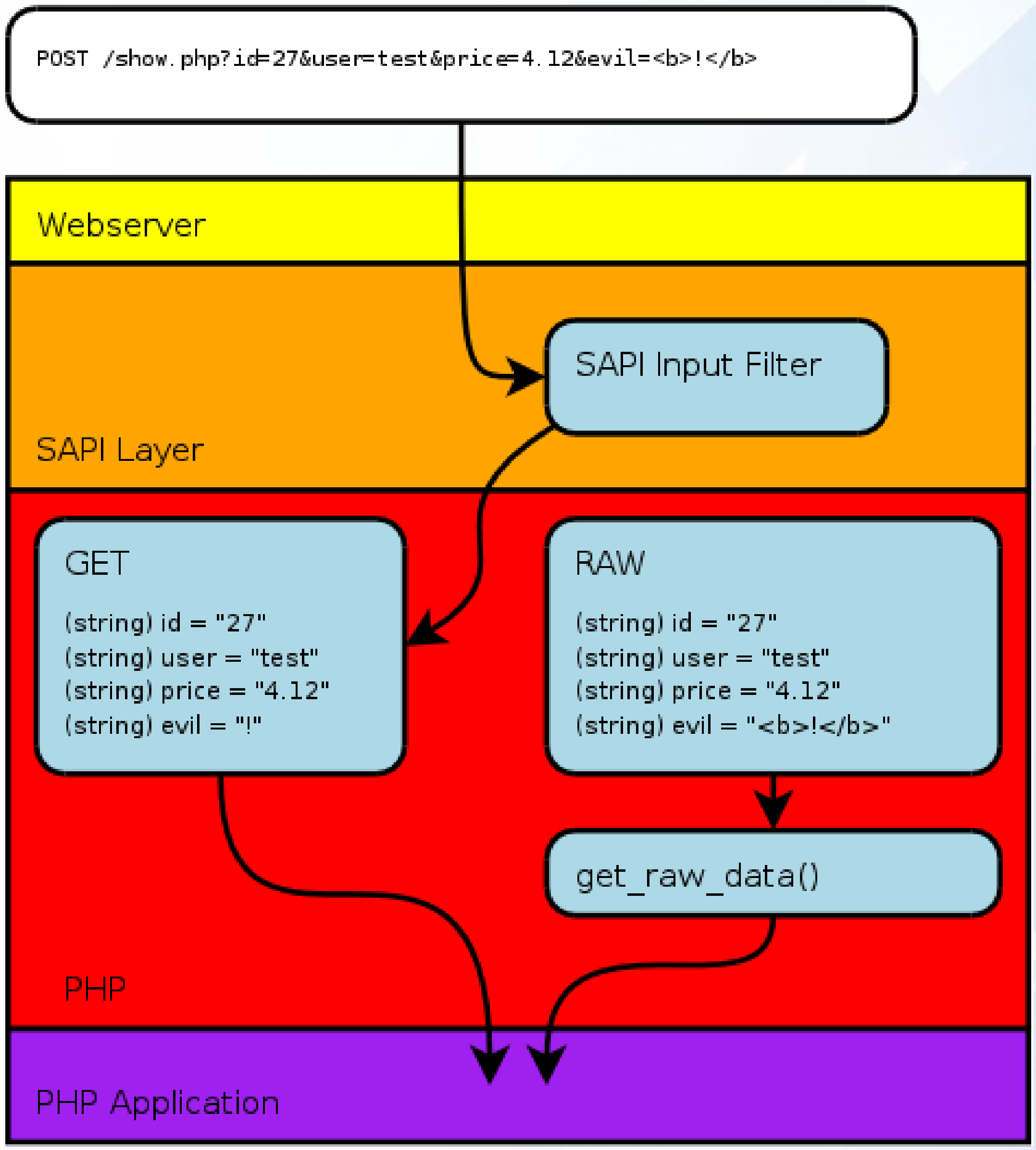
Current Situation



Share your information

eZ systems

First Idea of an Input Filter

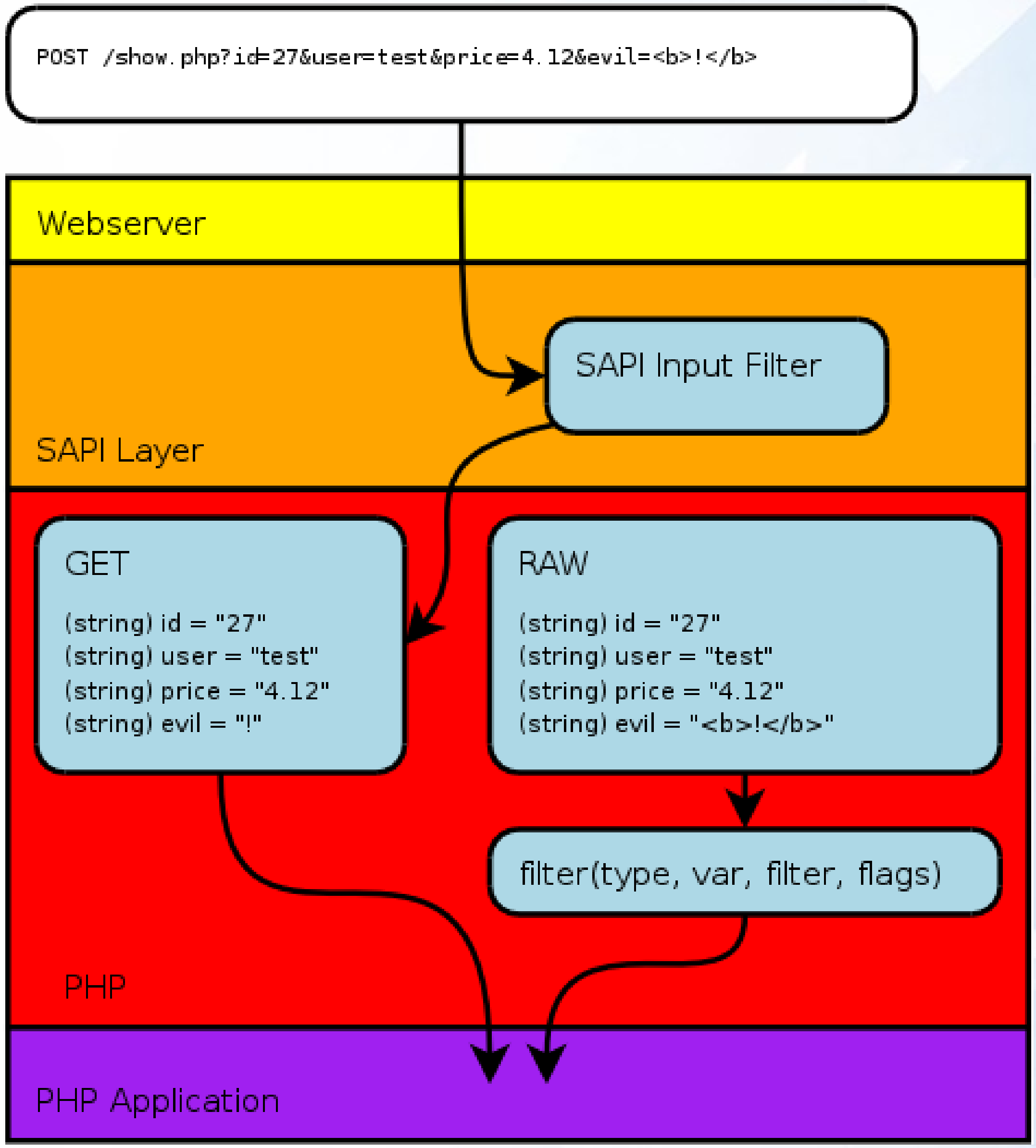


Share your information

eZ systems

Share your information

Second Idea of an Input Filter



PHP 5.2

- Comes with a filter extension
- Enabled by default
- Provides a default filter
- Provides two groups of accessing filters: sanitizing and logical
- Filters can have options to configure their behavior

eZ systems

Sanitizing filters try-out

```
<form action="" method="get">
<input type="text" name="data" maxlength="64" size="64"/>
<input type="submit"/>
</form>
<?php
if ( isset( $_GET['data'] ) ) {
    $filter = ini_get('filter.default');
    echo "The data filterered through '$filter' is:";
    var_dump( $_GET['data'] );
}
?>
```

Share your information

eZ systems

Share your information

Getting Input

```
<form action="" method="get">
Data: <input type="text" name="data" maxlength="64" size="64"/><br/>
<input type="submit"/>
</form>
<?php
$options = FILTER_FLAG_STRIP_HIGH;

if (isset($_GET['data'])) {
    $data = filter_input(
        INPUT_GET,              // source
        'data',                 // parameter name
        FILTER_SANITIZE_STRING, // filter
        $options                // options
    );
    var_dump( $data );
}
?>
```

eZ systems

Getting Input Arrays

```
<form action="" method="get">
Data 1: <input type="text" name="data[]" maxlength="64" size="64"/><br/>
Data 2: <input type="text" name="data[]" maxlength="64" size="64"/><br/>
<input type="submit"/>
</form>
<?php
$options = FILTER_FLAG_STRIP_HIGH | FILTER_REQUIRE_ARRAY;

if (isset($_GET['data'])) {
    $data = filter_input(
        INPUT_GET,              // source
        'data',                  // parameter name
        FILTER_SANITIZE_STRING, // filter
        $options                 // options
    );
    var_dump( $data );
}
?>
```

Share your information

eZ systems

More about dealing with arrays

- FILTER_REQUIRE_SCALAR (default): requires the input variable to be not an array
- FILTER_REQUIRE_ARRAY: requires the input variable to be an array
- FILTER_FORCE_ARRAY: converts the input variable to an array, even if a scalar was submitted

Share your information

Getting Input Arrays

```
<form action="" method="get">
Number: <input type="text" name="int" maxlength="64" size="64"/><br/>
String: <input type="text" name="string" maxlength="64" size="64"/><br/>
<input type="submit"/>
</form>
<?php
$definition = array(
    'int' => array(
        'filter' => FILTER_VALIDATE_INT,
        'options' => array( "min_range" => 1, "max_range" => 10 )
    ),
    'string' => FILTER_SANITIZE_SPECIAL_CHARS
);

if (isset($_GET['int'])) {
    $data = filter_input_array( INPUT_GET, $definition );
    var_dump( $data );
}
?>
```

Filtering Variables

```
<?php
$text = "This tekßt is göing\tto be changed\n";
$data = filter_var(
    $text,
    filter_id( 'special_chars' ),
    FILTER_FLAG_STRIP_HIGH
);
var_dump( $data );
?>
```

Sanitizing Filters

- Allows or disallows characters in a string
- Does not take care about data formats
- Does not transform types, but keeps strings

eZ systems

Share your information

Sanitizing filters

Is basically "strip_tags", but supports a couple of flags:

```
<?php
$flags = array(
    'FILTER_FLAG_NO_ENCODE_QUOTES', 'FILTER_FLAG_ENCODE_LOW',
    'FILTER_FLAG_ENCODE_HIGH', 'FILTER_FLAG_STRIP_LOW',
    'FILTER_FLAG_STRIP_HIGH'
);
$filter = FILTER_SANITIZE_STRING;
include 'presentations/slides/input-filter/render-form.php';
include 'presentations/slides/input-filter/check-data.php';
?>
```

Sanitizing filters

Is basically "url_encode", but supports a couple of flags:

```
<?php
$flags = array(
    'FILTER_FLAG_STRIP_LOW', 'FILTER_FLAG_STRIP_HIGH'
);
$filter = FILTER_SANITIZE_ENCODED;
include 'presentations/slides/input-filter/render-form.php';
include 'presentations/slides/input-filter/check-data.php';
?>
```

Sanitizing filters

Is basically "html_special_chars", but supports a couple of flags:

```
<?php
$flags = array(
    'FILTER_FLAG_STRIP_LOW', 'FILTER_FLAG_STRIP_HIGH',
    'FILTER_FLAG_ENCODE_HIGH',
);
$filter = FILTER_SANITIZE_SPECIAL_CHARS;
include 'presentations/slides/input-filter/render-form.php';
include 'presentations/slides/input-filter/check-data.php';
?>
```

eZ systems

Share your information

Sanitizing filters

By default doesn't do anything, but supports a couple of flags:

```
<?php
$flags = array(
    'FILTER_FLAG_ENCODE_AMP', 'FILTER_FLAG_ENCODE_LOW',
    'FILTER_FLAG_ENCODE_HIGH', 'FILTER_FLAG_STRIP_LOW',
    'FILTER_FLAG_STRIP_HIGH'
);
$filter = FILTER_UNSAFE_RAW;
include 'presentations/slides/input-filter/render-form.php';
include 'presentations/slides/input-filter/check-data.php';
?>
```

eZ systems

Share your information

Sanitizing filters

This only strips out illegal characters, no validation is done!
The allowed characters are: a-z A-Z 0-9 " ! # $ % & ' * + - / = ? ^ _ ` { | } ~ @ . [ ]

```
<?php
$flags = array();
$filter = FILTER_SANITIZE_EMAIL;
include 'presentations/slides/input-filter/render-form.php';
include 'presentations/slides/input-filter/check-data.php';
?>
```

Sanitizing filters

This only strips out illegal characters, no validation is done!
The allowed characters are: a-z A-Z 0-9 $ - _ . + ! * ' () , { } | \ ^ ~ [ ] ` < > # % " ; / ? : @ & =

```
<?php
$flags = array();
$filter = FILTER_SANITIZE_URL;
include 'presentations/slides/input-filter/render-form.php';
include 'presentations/slides/input-filter/check-data.php';
?>
```

eZ systems

Share your information

Sanitizing filters

Strips all chars which are not 0-9, + and -.

```
<?php
$flags = array();
$filter = FILTER_SANITIZE_NUMBER_INT;
include 'presentations/slides/input-filter/render-form.php';
include 'presentations/slides/input-filter/check-data.php';
?>
```

eZ systems

Share your information

Sanitizing filters

Strips all chars which are not 0-9, + and -, but supports some flags as well.

```
<?php
$flags = array(
    'FILTER_FLAG_ALLOW_FRACTION', 'FILTER_FLAG_ALLOW_THOUSAND',
    'FILTER_FLAG_ALLOW_SCIENTIFIC'
);
$filter = FILTER_SANITIZE_NUMBER_FLOAT;
include 'presentations/slides/input-filter/render-form.php';
include 'presentations/slides/input-filter/check-data.php';
?>
```

eZ systems

Sanitizing filters

Filter that applies magic quotes or actually "add_slashes"

```
<?php
$flags = array();
$filter = FILTER_SANITIZE_MAGIC_QUOTES;
include 'presentations/slides/input-filter/render-form.php';
include 'presentations/slides/input-filter/check-data.php';
?>
```

eZ systems

Validating Filters

- Analyses the data in a logical way
- Understands data formats
- Can transform type

Share your information

eZ systems

Share your information

Logical filters

```
<a name='form'/><?php
$flags = array( 'FILTER_FLAG_ALLOW_OCTAL', 'FILTER_FLAG_ALLOW_HEX' );
$filter = FILTER_VALIDATE_INT;
?>
<form action="" method="get">
data: <input type="text" name="data" maxlength="64" size="64"/>
<table>
<?php foreach( $flags as $flagName ) {
echo "<tr><td>$flagName</td><td><input type='checkbox'
name='$flagName'/></td></tr>";
} ?>
</table>
Min: <input type="text" name="min"/><br/>Max: <input type="text" name="max"/>
<input type="submit"/>
</form>
<?php
if (!empty($_GET['min']))
    $options['options']['min_range'] = (int) $_GET['min'];
if (!empty($_GET['max']))
    $options['options']['max_range'] = (int) $_GET['max'];
if (isset($_GET['data'])) {
    $o = 0;
    foreach( $flags as $flagName )
        if ( isset( $_GET[$flagName] ) )
            $o |= constant( $flagName );
    $options['flags'] = $o;
    $data = filter_input( INPUT_GET, 'data', $filter, $options );
    var_dump( $data );
}
?>
```

eZ systems

Share your information

Logical filters

Returns true for '1', 'on', 'yes' and 'true', false otherwise.

```
<?php
$flags = array();
$filter = FILTER_VALIDATE_BOOLEAN;
include 'presentations/slides/input-filter/render-form.php';
include 'presentations/slides/input-filter/check-data.php';
?>
```

Logical filters

```
<?php
$flags = array();
$filter = FILTER_VALIDATE_FLOAT;
include 'presentations/slides/input-filter/render-form.php';
include 'presentations/slides/input-filter/check-data.php';
?>
```

eZ systems

Share your information

Logical filters

```
<form action="" method="get">
Data: <input type="text" name="data" maxlength="64" size="64"/><br/>
Regexp: <input type="text" name="regexp" maxlength="64" size="64"/><br/>
<input type="submit"/>
</form>
<?php
$filter = FILTER_VALIDATE_REGEXP;
if (!empty($_GET['regexp']))
    $options = array( 'options' =>
        array( 'regexp' => $_GET['regexp'] ) );

if (isset($_GET['data'])) {
    $data = filter_input( INPUT_GET, 'data', $filter, $options );
    var_dump( $data );
}
?>
```

eZ systems

Share your information

Logical filters

```
<?php
$flags = array(
    'FILTER_FLAG_SCHEME_REQUIRED', 'FILTER_FLAG_HOST_REQUIRED',
    'FILTER_FLAG_PATH_REQUIRED', 'FILTER_FLAG_QUERY_REQUIRED'
);
$filter = FILTER_VALIDATE_URL;
include 'presentations/slides/input-filter/render-form.php';
include 'presentations/slides/input-filter/check-data.php';
?>
```

Logical filters

```
<?php
$flags = array();
$filter = FILTER_VALIDATE_EMAIL;
include 'presentations/slides/input-filter/render-form.php';
include 'presentations/slides/input-filter/check-data.php';
?>
```

eZ systems

Logical filters

```
<?php
$flags = array(
    'FILTER_FLAG_IPV4', 'FILTER_FLAG_IPV6', 'FILTER_FLAG_NO_RES_RANGE',
    'FILTER_FLAG_NO_PRIV_RANGE'
);
$filter = FILTER_VALIDATE_IP;
include 'presentations/slides/input-filter/render-form.php';
include 'presentations/slides/input-filter/check-data.php';
?>
```

Share your information

eZ systems

Share your information

Callback Filter

- Allows you to write your own callback filter
- Can not be used as default filter

Callback filter

```
<form action="" method="get">
Data: <input type="text" name="data" maxlength="64" size="64"/><br/>
<input type="submit"/>
</form>
<?php
$filter = FILTER_CALLBACK;
$callback = array( 'options' => array( 'Validate', 'My' ) );

if (isset($_GET['data'])) {
    $data = filter_input( INPUT_GET, 'data', $filter, $callback );
    var_dump( $data );
}

class Validate {
    function My( $text ) {
        if ( $text == 'PHP' ) {
            $text = 'PHP Rocks!';
            return $text;
        }
        return false;
    }
}
?>
```

eZ systems

Share your information

eZ components Goals

- Provide a solid platform for PHP application development
- Don't force a structure: no "framework"
- Clean and simple API
- Excellent documentation
- Keep backward compatibility for longer periods of time
- Stable and few regressions
- Clean IP, Open Source friendly

eZ systems

Share your information

UserInput

```
<?php
require 'ezc-setup.php';
if ( ezcInputForm::hasGetData() )
{
    $definition = array(
        'test' => new ezcInputFormDefinitionElement(
            ezcInputFormDefinitionElement::REQUIRED, 'int' ),
        'test2' => new ezcInputFormDefinitionElement(
            ezcInputFormDefinitionElement::REQUIRED, 'number_int' ),
    );

    try
    {
        $form = new ezcInputForm( INPUT_GET, $definition );
        echo $form->test, "\n";
        echo $form->test2, "\n";
    }
    catch ( ezcInputFormException $e )
    {
        die( $e->getMessage() );
    }
}
?>
```

eZ systems

Share your information

Resources

These slides: http://derickrethans.nl/talks.php
UserInput: http://components.ez.no/doc/UserInput
SQLite Input Filter: http://derickrethans.nl/sqlite_filter.php
Questions?: mailto:dr@ez.no