Безопасность PHP приложений

LoftSchool
от мыслителя к создателю

Нужна ли нам безопасность?

- Разработка программного обеспечения является достаточно сложной задачей - а вопросы безопасности делают ее ещё более сложной
- Недостатки могу проявится, а могут и не проявится при повседневном использовании
- Но обнаруженные не Вами недостатки могут дорого Вам стоит

Основные принципы

- Проверок много не бывает
- Нельзя верить никому!
- Любые действия пользователя могу быть направлены против вашего сайта! И будут направлены!

Угрозы безопасности сайта

- Внедрение кода (XSS)
- Внедрение SQL кода (SQL-инъекции)
- Внедрение исполняемого кода (PHP инъекция)
- Недостаточная обработка или нет обработки PHP
- Раскрытие данных сессия (сеансов)
- Слабая защита при аутентификации и авторизации
- Неконтролируемый доступ к файлам и БД

Внедрение HTML кода (XSS)

- Возможность модифицировать HTML код разметки

Причины

- Недостаточная проверка входных данных
- Использование прямого вывода параметров приложения
- Неправильная использование шаблонизатора

Возможные решения

- Любые данные полученные извне должны быть проверены
- Минимизируйте количество каналов получения пользовательских данных (все запросы должны идти через 1 файл)
- Не ищите ошибки во входных данных. Ищите то, что вам нужно. Например:

```
$age = abs((int)$_GET[‘age’]);
if($age < 18){
    exit(‘Повзрослей пока что!’);
}
```

Возможные решения

- Любые данные полученные извне должны быть проверены
- Минимизируйте количество каналов получения пользовательских данных (все запросы должны идти через 1 файл)
- Не ищите ошибки во входных данных. Ищите то, что вам нужно. Например:

  если необходимо вставить данные из визуального редактора, то необходимо использовать функцию htmlentities

Демонстрация

LoftSchool
от мыслителя к создателю

Работа с сеансами

При работе с сеансами (сессиями) необходимо включать настройку session.httponly

```
ini_set(‘session.httponly’, 1);
```

Отметка, согласно которой доступ к cookies может быть получен только через HTTP протокол. Это означает, что cookies не будут доступны через скриптовые языки, такие как JavaScript. Данная настройка позволяет эффективно защитить от XSS атак

Работа с сеансами

При смене пользовательских привилегий, авторизации пользователя, для предотвращении перехвата идентификатора сеанса необходимо вызывать функцию session_regenerate_id()

Демонстрация

LoftSchool
от мыслителя к создателю

Внедрение SQL кода

- Одна из самых распространенных угроз и опасных угроз
- Возможность модификации SQL операторов, используемых в БД

Причины:

- Непродуманность приложения и структуры БД
- Недостаточная фильтрация данных

Внедрение SQL кода

```
$query = $_GET['search'];
$res = mysqli->query('SELECT id, date, title, description …. FROM news
WHERE title LIKE ("%$query%")');
```

В $query - ')+AND+(id_author = '1)

```
SELECT id, date, title, description …. FROM news WHERE title LIKE ("%") AND
(id_author = 1)
```

Расщепление запроса

```
$id = $_POST['id'];
$result = $mysql->query('SELECT * FROM news WHERE id = '.$id);
```

Значение: 145; INSERT INTO admin(username, password) VALUES (`VasyaAdmin`, `openDoor`);

SELECT * FROM news WHERE id = 145; INSERT INTO admin(username, password) VALUES(`VasyaAdmin`, `openDoor`);

Использование UNION

```
$id = $_POST['id'];
$sql = “SELECT id, title, author, description FROM news WHERE id =”. $id;
```

Текст: -1 UNION SELECT 1, username, password, 1 FROM admin

Результат

```
SELECT id, title, author, descriptionm FROM news WHERE id = -1
UNION SELECT 1, username, password, 1 FROM admin
```

Экранирование хвоста запроса

$id = $_POST['id'];

$sql = “SELECT id, title, author, description FROM news WHERE id =”. $id .” AND author LIKE (‘a%’)”

Текст: -1 UNION SELECT login, password FROM admin --

Результат: SELECT id, title, author, description FROM news WHERE id =-1 UNION SELECT login, password FROM admin -- AND author LIKE (‘a%’)

Возможные решения

- Любые данные, полученные извне, подлежат обязательной проверке
- Максимально избегайте конкатенации в SQL запросах
- Используйте хранимые процедуры, представления или подготовленные запросы

Подготовленные запросы

```
$stmt = $pdo->prepare('SELECT * FROM products WHERE id_catalog = :catalog');
$stmt->bindParam(':catalog', $catalog, PDO::PARAM_INT);
$stmt->execute();
```

Использование представлений

```
CREATE VIEW vName AS
SELECT products.id_catalog as id, products.name, products.
mark,
products.count, products.price, products.description,
catalogs.name as title
    FROM products
    JOIN catalogs ON products.id_catalog = catalogs.id;
```

Хранимые процедуры

```
DELIMITER //
CREATE PROCEDURE spGetProductsByCatalog(id_catalog INT)
BEGIN
    SELECT products.id_product as id, products.name as title, products.mark,
    products.count, products.price, catalogs.name as catalog
    FROM products
    JOIN catalogs ON products.id_catalog = catalogs.id
    WHERE products.id_catalog = id_catalog;
END //
```

Демонстрация

LoftSchool
от мыслителя к создателю

Запретить пользователю вывод ошибок на экран

PHP.INI

```
display_errors =off
log_errors      =on
error_log   = /var/log/php.error.log
```

SCRIPT

```
ini_set('display_errors', 'off');
ini_set('log_errors', 'on');
ini_set('error_log', '/var/log/php.error.log');
```

LoftSchool
от мыслителя к создателю

Константы ошибок

Константа	Описание	Возможность перехвата
E_ERROR	Неустранимая ошибка	Нет
E_WARNING	Исправимая ошибка	Да
E_PARSE	Ошибка парсера	Нет
E_NOTICE	Потенциальная ошибка	Да
E_RECOVERABLE_ERROR	Опасная, но не фатальная ошибка (например, несоответствие типа)	Да
E_DEPRECATED	Предупреждение об использовании устаревшей функции или возможности	Да
E_ALL	Все ошибки	Нет

Демонстрация

LoftSchool
от мыслителя к создателю

Использование is - функций

- is_array
- is_bool
- is_callable
- is_double
- is_float
- is_int
- is_integer
- is_string
- is_long
- is_null
- is_numeric
- is_object
- is_real
- is_resource
- is_scalar

Проверка присутствия значения

filter_has_value(INPUT_POST, ‘field’);

- INPUT_POST
- INPUT_GET
- INPUT_SESSION
- INPUT_SERVER
  INPUT_ENV
- INPUT_COOKIE
- INPUT_REQUEST

Валидация данных

```
if(filter_input(INPUT_POST, 'age', FILTER_VALIDATE_INT)){
    echo "Данные были введеный КОРРЕКТНО"; }
```

- FILTER_VALIDATE_BOOLEAN
- FILTER_VALIDATE_EMAIL
- FILTER_VALIDATE_FLOAT
- FILTER_VALIDATE_INT
- FILTER_VALIDATE_IP
- FILTER_VALIDATE_MAC
- FILTER_VALIDATE_REGEXP
- FILTER_VALIDATE_URL

Проверка паролей

```php
$old_pass = 'dsasdadaadwqwcv32ca';
$check_pass = 'dsasdadaadwqwcv32ca';

$old_pass_hash = password_hash($old_pass, PASSWORD_DEFAULT);

if(password_verify($check_pass, $old_pass_hash)){
    echo "Все good";
} else {
    echo "Пароли не совпадают";
}
```

Демонстрация

LoftSchool
от мыслителя к создателю