SQL injection problem exists for multiple functions below version 5.0 #553

Open feizi76 opened this issue on Aug 23, 2019 · 0 comments

feizi76 commented on Aug 23, 2019 • edited

I found a lot of such code in our extensive penetration test.

```
$sql=$Data->query("select * from users where id=$id"); $sql=$Data->where("id=$id")->select();
```

Such code is not pre-processed by sql during preprocessing. can be seen

```
protected function parseSql($sql,$parse) { if(true === $parse) { $options = $this->_parseOptions(); $sql = $this->db->parseSql($sql,$options); }elseif(is_array($parse)){ // SQL预处理
$parse = array_map(array($this->db,'escapeString'),$parse); $sql = vsprintf($sql,$parse); }else{ $sql = strtr($sql,array('__TABLE__'=>$this-
>getTableName(),'__PREFIX__'=>C('DB_PREFIX'))); } $this->db->setModel($this->name); return $sql; }
```

or

```
`public function where($where,$parse=null){
if(!is_null($parse) && is_string($where)) {
if(!is_array($parse)) {
$parse = func_get_args();
array_shift($parse);
}
$parse = array_map(array($this->db,'escapeString'),$parse);
$where = vsprintf($where,$parse);
}elseif(is_object($where)){
$where = get_object_vars($where);
}
if(is_string($where) && '' != $where){
$map = array();
$map['_string'] = $where;
$where = $map;
}
if(isset($this->options['where'])){
$this->options['where'] = array_merge($this->options['where'],$where);
}else{
$this->options['where'] = $where;
}


  return $this;


}`
```

And the official website also has a lot of such writings.http://www.thinkphp.cn/extend/246.html
Are all wrong demonstrations that will cause more SQL injection
Sql injection can also be performed in the MODEL.class.php limit function and the order function.

```
public function limit($offset,$length=null){ $this->options['limit'] = is_null($length)?$offset:$offset.','.$length; return $this; } $sql=$Data->where("id=$id")->order($id);
```

Assignees

No one assigned

Labels

None yet

Projects

None yet

Milestone

No milestone

Development

No branches or pull requests

1 participant