

——微信与移动Web开发之

第4讲 被动回复消息

课程概要

- 首先讲解XML的基础知识。
- 在有了XML的基础之后，讲解XML格式的微信消息，以文本消息格式为例讲解如何回复消息。
- 在已经实现被动回复消息的基础上，实现消息加密解密的功能

本次课程目录

- 一 . XML解析
- 二 . 接收并回复消息
- 三 . 消息加密处理

第一节

- 一 . XML解析
- 二 . 接收并回复消息
- 三 . 消息加密解密

XML基础

- 可扩展标记语言 (Extensible Markup Language) 。
- XML很像超文本标记语言的标记语言。
- XML的标签没有被预定义，需要自行定义标签。

与HTML的差异：

- XML不是HTML的替代。
- XML被设计用来传输和存储数据，其焦点是数据的内容。
- HTML被设计用来显示数据，其焦点是数据的外观。

XML示例：

```
<?xml version="1.0" encoding="UTF-8"?>
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me</body>
</note>
```

XML语法规则

- XML必须有根元素
- 所有元素都必须有一个关闭标签
- 标签对大小写敏感
- 属性必须加引号
- 标签必须被正确嵌套
- 空格会被保留（HTML会把多个连续的空格字符裁减为一个）
- XML声明存在的话需要放在第一行，但它不是必须的
`<?xml version="1.0" encoding="utf-8"?>`

- 错误的格式：
//没有根元素
`<name>Albert</name>`

//大小写不统一
`<name>Albert</Name>`

//没有关闭标签
`<name>Albert`

//属性没有加引号
`<name sex=male>Albert</name>`
- 正确的格式：
`<person>
 <name sex="male">ALbert</name>
 <age>27</age>
</person>`

XML什么也不做

- XML仅仅是规定了如何去编写，统一了格式方便程序去解析。
- 至于解析后的数据，则需要程序员自己开发软件去支持。
- XML提供的是统一的格式化规则，统一的解析规则。基于此，互联网的数据传输兼容性大大提升，不需要针对每种格式去编写对应的解析程序。

PHP解析XML

- PHP内置了处理XML的扩展，比较简单的扩展SimpleXML可以把XML格式的字符串解析成SimpleXMLElement对象。使用SimpleXMLElement对象提供的方法可以获取需要的数据。
- 使用simplexml_load_string(\$xml)函数对\$xml进行解析。
- XML标签的名称就会被解析为对象的属性。
- 使用get_object_vars函数可以把PHP对象转换成数组。
- DOMDocument类可以加载XML数据为DOM，之后使用方法操标签数据。

使用其他扩展库解析XML

- 使用composer下载houdunwang/xml
composer require houdunwang/xml
- 使用方式：

```
$result = houdunwang\xml\Xml::toArray($xml);
```

此方法会把XML格式的文本数据转换成PHP的数组。

第二讲

一 . XML解析

二 . 接收并回复消息

三 . 消息加密解密

微信服务器转发文本消息

```
<xml>  
<ToUserName> <![CDATA[toUser]]> </ToUserName>  
<FromUserName> <![CDATA[fromUser]]> </FromUserName>  
<CreateTime> 1348831860</CreateTime>  
<MsgType> <![CDATA[text]]> </MsgType>  
<Content> <![CDATA[Hello]]> </Content>  
<MsgId> 1234567890123456</MsgId>  
</xml>
```

微信服务器转发文本消息的XML描述。客户端发送消息后，微信服务器会加入MsgId字段，此字段表示消息在微信服务器的唯一ID。

参数的解释

ToUserName	开发者微信号
FromUserName	发送方帐号（一个OpenID）
CreateTime	消息创建时间（整型）
MsgType	消息类型，文本消息为text
Content	文本消息内容
MsgId	消息id，64位整型

开发者服务器如何回复

```
<xml>  
<ToUserName><![CDATA[toUser]]></ToUserName>  
<FromUserName><![CDATA[fromUser]]></FromUserName>  
<CreateTime>12345678</CreateTime>  
<MsgType><![CDATA[text]]></MsgType>  
<Content><![CDATA[你好]]></Content>  
</xml>
```

- FromUserName与ToUserName互换，返回格式化后的结果即可实现原样返回消息。

不支持的消息类型

用户发送在微信下载的表情或者是自定义的表情会提示“不支持的消息类型，暂无法显示”。微信默认自带的表情可以显示，因为默认自在的表情实际上是文本表示的消息。



第三讲

- 一 . XML解析
- 二 . 接收并回复消息
- 三 . 消息加密解密

微信公众号设置开启加密

- 微信公众平台打开基本配置，重新设置服务器设置选项的消息加密模式为安全模式。
- 由于配置这个选项会导致微信服务器重新认证，所以需要先在开发者服务器上注释消息处理函数，开启URL验证。配置通过后，再改回消息处理模式。

下载加解密示例代码

- 打开微信公众平台技术文档。
- 左侧消息管理菜单中点击消息加解密说明。
- 内容介绍的最下面有下载链接。
- 解压缩后仅仅使用PHP的示例代码即可。
- 注意需要更改代码的几个地方才可以使用：
 1. 由于从PHP5.3.3版本以后，不能再使用与类名相同的方法名作为构造函数，所以对于高版本的PHP，需要把wxBizMsgCrypt.php与pkcs7Encoder.php中和类名相同的方法改成 __construct()方法。
 2. 示例代码中的demo.php文件中格式化回复消息缺少了参数，不需要参看demo.php文件。

加解密基本过程

在已经实现明文模式传输的基础之上，加入加密解密的功能。

- 启用加密以后，微信服务器会把消息加密后传输，同时会在请求里加上一些加解密需要的参数。
- 开发者服务器接收到消息以后，验证消息的可靠性，之后先进行解密，处理后，需要回复的消息要先进行加密，然后返回结果。
- 在收到消息后，多了解密的过程，回复消息时，多了加密的过程。
- 这时候，微信开发者配置中设置的加解密密钥就要起作用了，同时要用到的还有公众号信息的开发者ID (AppId)。

实现消息加密功能

- 参数说明：

\$token：用户设置的值；\$encodingAesKey：用户自己设置的字符串；

\$appid：微信公众号的ID；

- 解密消息：

```
$pc = new WXBizMsgCrypt($token, $encodingAesKey, $appid);
```

```
//$msg_sign：消息签名；$msg：解密后的数据赋值给$msg
```

```
$errCode = $pc->decryptMsg($msg_sign, $timestamp, $nonce, $postdata, $msg);
```

- 加密消息：

```
$pc = new WXBizMsgCrypt($token, $encodingAesKey, $appid);
```

```
//加密后的数据会赋值给$encryptMsg
```

```
// $data是要加密的数据
```

```
$errCode = $pc->encryptMsg($data, $timestamp,$nonce,$encryptMsg);
```

感谢聆听！

THANK YOU FOR YOUR ATTENTION