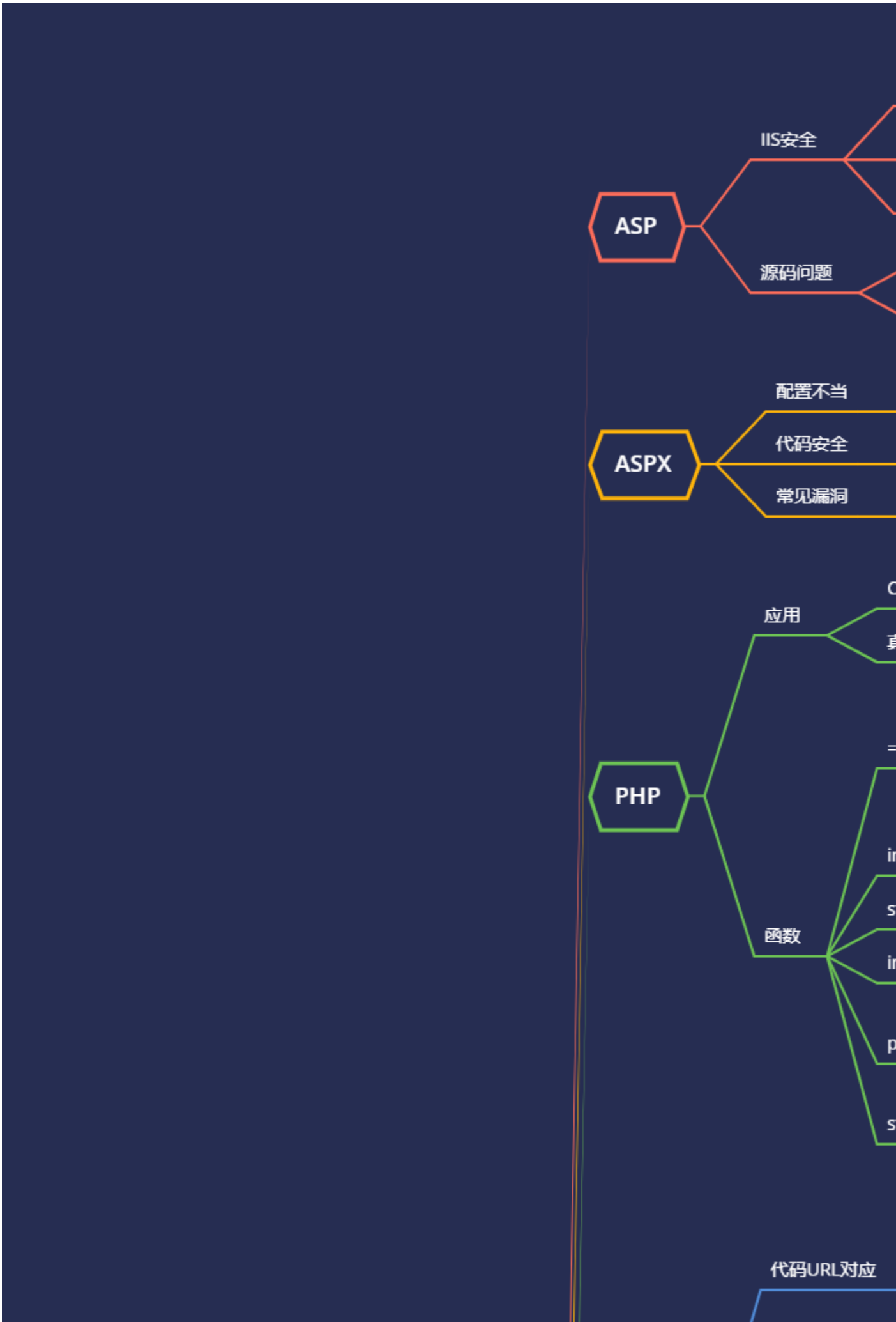
WEB 攻防-通用漏洞&PHP 反序列化&原生类&漏洞绕过&公私有属性


IIS安全
ASP
源码问题
配置不当
ASPX
代码安全
常见漏洞
应用
PHP
函数
代码URL对应

#知识点：

1、反序列化魔术方法全解

2、反序列化变量属性全解

3、反序列化魔术方法原生类

4、反序列化语言特性漏洞绕过

-其他魔术方法

-共有&私有&保护

-语言模式方法漏洞

-原生类获取利用配合

#反序列化利用大概分类三类

-魔术方法的调用逻辑-如触发条件

-语言原生类的调用逻辑-如 SoapClient

-语言自身的安全缺陷-如 CVE-2016-7124

#反序列化课程点：

-PHP&Java&Python

序列化：对象转换为数组或字符串等格式

反序列化：将数组或字符串等格式转换成对象

serialize()　　//将一个对象转换成一个字符串

unserialize()　//将字符串还原成一个对象

#PHP 反序列化漏洞

原理：未对用户输入的序列化字符串进行检测，导致攻击者可以控制反序列化过程，从而导致代码执行，SQL 注入，目录遍历等不可控后果。在反序列化的过程中自动触发了某些魔术方法。当进行反序列化的时候就有可能会触发对象中的一些魔术方法。

#魔术方法利用点分析：

触发：unserialize 函数的变量可控，文件中存在可利用的类，类中有魔术方法：

__construct()：//构造函数，当对象 new 的时候会自动调用

__destruct()：//析构函数当对象被销毁时会被自动调用

__wakeup()：//unserialize()时会被自动调用

__invoke()：//当尝试以调用函数的方法调用一个对象时，会被自动调用

__call()：//在对象上下文中调用不可访问的方法时触发

__callStatci()：//在静态上下文中调用不可访问的方法时触发

__get()：//用于从不可访问的属性读取数据

__set()：//用于将数据写入不可访问的属性

演示案例：

- 方法&属性-调用详解&变量数据详解
- CTF-语言漏洞-__wakeup()方法绕过
- CTF-方法原生类-获取&利用&配合其他

#方法&属性-调用详解&变量数据详解

对象变量属性：

public(公共的):在本类内部、外部类、子类都可以访问

protect(受保护的):只有本类或子类或父类中可以访问

private(私人的):只有本类内部可以使用

序列化数据显示：

private 属性序列化的时候格式是%00 类名%00 成员名

protect 属性序列化的时候格式是%00*%00 成员名

具体代码：

```
<?php
header("Content-type: text/html; charset=utf-8");
/*public private protected 说明
class test{
    public $name="xiaodi";
    private $age="29";
    protected $sex="man";
}
$a=new test();
$a=serialize($a);
print_r($a);
*/

/*__construct __destruct 魔术方法 创建调用__construct 2 种销毁调用
__destruct
class Test{
    public $name;
    public $age;
    public $string;
    // __construct：实例化对象时被调用.其作用是拿来初始化一些值。
    public function __construct($name, $age, $string){
        echo "__construct 初始化"."<br>";
        $this->name = $name;
        $this->age = $age;
        $this->string = $string;
    }
    // __destruct：当删除一个对象或对象操作终止时被调用。其最主要的作用是拿
来做垃圾回收机制。
    /*
     * 当对象销毁时会调用此方法
     * 一是用户主动销毁对象，二是当程序结束时由引擎自动销毁
```

涉及资源：

补充：涉及录像课件资源软件包资料等下载地址