



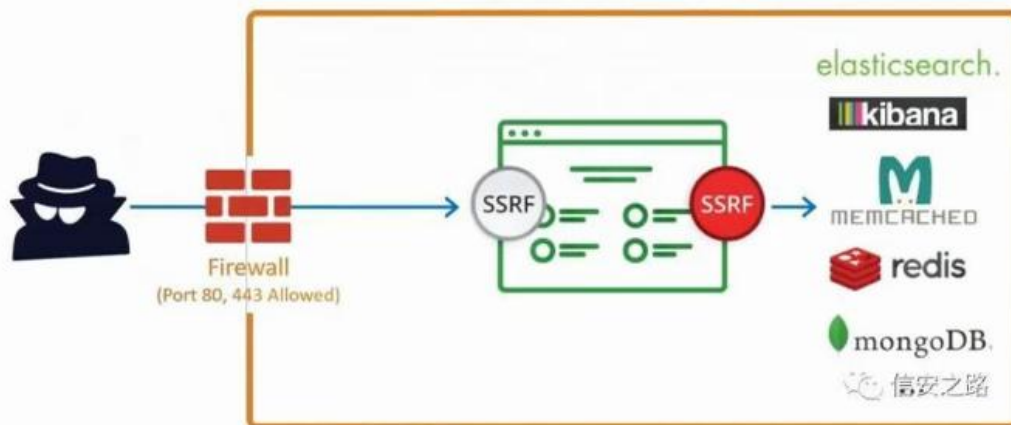
渗透测试工程师技术实战课

SSRF漏洞的原理以及分析



SSRF漏洞基础

- SSRF(Server-Side Request Forgery, 服务器端请求伪造)漏洞
- 原理：很多web应用都提供了从其他的服务器上获取数据的功能。
- 主要的攻击方式：
 - 对外网、内网、本地进行端口扫描(3306)
 - 攻击运行在内网或本地的有漏洞程序/web应用(比如溢出)
 - 指纹识别
 - File://读取本地文件
- 分享、转码服务、在线翻译
- 图片、文章收藏功能



● SSRF漏洞检测

- <http://www.xxx.com/image.php?image=http://www.xxc.com/a.jpg>
 - <http://www.xxx.com/image.php?image=http://127.0.0.1:22>
- <http://share.xxx.com/index.php?url=http://127.0.0.1>
- <http://title.xxx.com/title?title=http://title.xxx.com/as52ps63de>
- share
- wap
- url
- link
- src
- source
- ...



SSRF漏洞基础

- file_get_contents ()
 - fsockopen ()
 - curl_exec ()
-
- 环境部署
 - 利用
 - File
 - Dict
 - http
 - gopher

```
<?php
// create curl resource
$ch = curl_init();
// set url
curl_setopt($ch, CURLOPT_URL, $_POST["handler"]);
//return the transfer as a string
curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
// $output contains the output string
$output = curl_exec($ch);
// close curl resource to free up system resources
curl_close($ch);
echo $output;
?>
```



● SSRF漏洞攻击利用

- SSRF+redis 获取内网主机权限，利用SSRF来对redis的未授权访问执行命令。从而达到获取主机权限的目的



● SSRF漏洞绕过

● 更改IP地址写法

- 如果正则($^10(\.[2][0-4]\d|[2][5][0-5]|[01]?\d?\d){3}$$)
- 采用进制转换，127.0.0.1八进制：0177.0.0.1。十六进制：0x7f.0.0.1。十进制：2130706433
- 10.1/0.0.0.0
- 使用解析到内网的域名
- 利用解析URL所出现的问题
- 利用跳转
- 通过各种非HTTP协议
- 利用IPv6



● SSRF漏洞修复

- 限制返回信息的，例如请求文件，只返回文件是否请求成功，没有请求成功到文件统一返回错误信息。
- 对请求地址设置白名单，只允许请求白名单内的地址。
- 禁用除http和https外的协议，如：file://，gopher://，dict://等
- 限制请求的端口为固定服务端口，如：80，443



The image features a central figure of a person wearing a black hoodie, with their right hand pressed against their forehead in a gesture of stress or contemplation. The background is a vibrant blue digital landscape. It includes a glowing globe on the right side, a network of interconnected nodes and lines, and several padlock icons scattered throughout, symbolizing security or digital threats. The overall aesthetic is high-tech and futuristic.

谢谢观赏