远程控制

正向连接和反向连接的区别

反向连接

控制主机监听一个端口，由受害主机反向去连接控制主机的过程，适用于受害主机出网（或出网且没有公网ip）的情况。例如，受害主机是一台**位于内网，并且没有公网ip**且能访问互联网的主机，控制主机**无法直接通过ip地址访问**到受害主机。所以此时需要在控制主机上监听一个端口，让受害机主动连接控制主机，从而实现对控制主机的控制。

反向连接时由**测试机主动去连接攻击机**，这样的好处在于不会受到测试机防火墙策略以及所在网络的NAT限制；

正向连接

受害主机监听一个端口，由控制主机主动去连接受害主机的过程，适用于受害主机具有公网ip的情况下。即攻击机设置一个端口（LPORT），Payload在测试机执行打开该端口，以便攻击机可以接入。

反向连接的实施过程

攻击机设置一个端口（LPORT）和IP（LHOST），并生成一个会使测试机执行后远程连接到攻击机的恶意程序，Payload在测试机执行后会主动连接攻击机IP的端口，这时如果在攻击机监听该端口会发现测试机已经连接。

攻击者先通过某个手段在目标机器上植入恶意代码，并且该代码可以被触法。攻击者设置一个端口和一个IP，当被攻击者执行了恶意代码后攻击者的机器就会获取被攻击者的代码。

漏洞扫描

1x01 工具们

工具：

- AWVS：漏洞扫描工具
- Beef：XSS漏洞利用工具
- SQLMAP：自动话sql注入工具：
  - 查询当前数据库：`sqlmap -u "IP" --cookie "xxx" --current-db`
  - 查询当前使用者：`--current user`
  - 爆破数据表：`-D xx --tables`
  - 爆破数据表表头：`-D xx -T xx --columns`
  - 爆破具体的列：`-D xx -T xx -C xx`
- Whatweb: 查询网页的基本信息 `whatweb IP`
- Wpscan:可以扫描WordPress中的多种安全漏洞
- Dirb：爆破用 `dirb -u http://IP`
- MeterSploit:功能非常强大的渗透工具

XSS攻击/SQL注入

XSS攻击：XSS攻击通常指的是通过利用开网页发时留下的漏洞，通过巧妙的方法**注入恶意指令代码到网页，使用户加载并执行攻击者恶意制造的网页程序。**这些恶意网页程序通常是JavaScript，但实际上也可以包括Java、 VBScript、ActiveX、 Flash 或者甚至是普通的HTML。攻击成功后，攻击者可能得到包括但不限于更高的权限（如执行一些操作）、私密网页内容、会话和cookie等各种内容。

SQL注入攻击的原理：恶意用户在提交查询请求的过程中**将SQL语句插入到请求内容中**，同时程序本身对用户输入内容过分信任而未对恶意用户插入的SQL语句进行过滤，导致SQL语句直接被服务端执行。

如何防护SQL注入

- 1. 使用参数化查询
- 2. 输入验证和过滤
- 3. 使用存储过程
- 4. 最小权限原则
- 5. 使用ORM框架
- 6. 使用准备语句
- 7. 使用安全的数据库连接
- 8. 避免动态拼接SQL语句
- 9. 使用防火墙和入侵检测系统
- 10. 定期更新和维护数据库软件

缓冲区溢出攻击

如何利用缓冲区溢出进行攻击?

攻击者可以故意将精心制作的输入馈入程序，程序将尝试将该输入存储在不够大的缓冲区中，因此输入会覆盖连接到缓冲区空间的部分内存。如果程序的内存布局定义明确，则攻击者可以故意覆盖已知包含可执行代码的区域。然后，攻击者可以用自己的可执行代码替换这些代码，这可以大大改变程序的工作方式。

例如，如果内存中的被覆盖部分包含一个指针（指向内存中另一个位置的对象），则攻击者的代码可以用另一个指向漏洞利用有效载荷的指针来替换该代码。这样就可以将整个程序的控制权转移给攻击者的代码。

缓冲区溢出攻击的类型有哪些?

缓冲区溢出攻击有很多类型，它们采用不同的策略并针对不同的代码段。以下是一些最著名的类型。

- 堆栈溢出攻击 - 这是最常见的缓冲区溢出攻击类型，涉及到调用堆栈*上的缓冲区溢出。
- 堆溢出攻击 - 这种类型的攻击针对开放的内存池中称为堆*的数据。
- 整数溢出攻击 - 在整数溢出中，算术运算得出对于要存储结果的整数类型而言太大的整数；这可能导致缓冲区溢出。
- Unicode 溢出 - Unicode 溢出通过将 Unicode 字符插入需要 ASCII 字符的输入中来创建缓冲区溢出。（ASCII 和 unicode 是使计算机表达文本的编码标准。例如，字母“a”由 ASCII 中的数字 97 表达。虽然 ASCII 码仅用于表达西方语言中的字符，但 unicode 可以为地球上几乎所有书面语言创建字符。因为 unicode 中有更多可用的字符，所以许多 unicode 字符大于最大的 ASCII 字符。）

如何防范缓冲区溢出攻击?

现代操作系统具有运行时保护，可帮助防护缓冲区溢出攻击。我们来探讨有助于防护漏洞利用风险的 2 种常见保护措施：

- 地址空间随机化 - 随机重新排列进程的关键数据区的地址空间位置。缓冲区溢出攻击通常依赖于了解重要的可执行代码的确切位置，地址空间的随机化可以使这种了解几乎不可能。

- 防止数据执行 - 标记内存的某些区域（可执行或不可执行），防止漏洞利用运行不可执行区域中的代码。