网络安全课程

web应用安全

目标

- 学完本课程后，您将能够:
  - 了解常见web应用漏洞
  - 了解基本漏洞原理
  - 掌握常见漏洞攻击方式
  - 掌握漏洞修复方式

目录

Web应用安全介绍

- Web应用是以浏览器为客户端，利用HTTP/HTTPS协议与后台服务端应用进行交互的一种应用架构。Web应用安全指保护网站和在线服务免受因代码或应用容器存在的各种漏洞产生的不同安全威胁。
- 基于Web环境的互联网应用越来越广泛，企业信息化的过程中各种应用都架设在Web平台上，Web业务的迅速发展也引起黑客们的强烈关注，接踵而至的就是Web安全威胁的凸显，黑客利用网站操作系统的漏洞和Web服务程序的SQL注入漏洞等得到Web服务器的控制权限，轻则篡改网页内容，重则窃取重要内部数据，更为严重的则是在网页中植入恶意代码，使得网站访问者受到侵害

Web应用十大安全隐患（OWASP TOP 10）

- OWASP（Open Web Application Security Project）是一个开源的、非盈利的全球性安全组织，致力于应用软件的安
- **OWAS** 最可能、最常见、
- 下表显

2013年版《OWASP Top 10》		2017年版《OWASP Top 10》
A1 – 注入	→	A1:2017 – 注入
A2 – 失效的身份认证和会话管理	→	A2:2017 –失效的身份认证
A3 – 跨站脚本（XSS）	↘	A3:2017 –敏感信息泄漏
A4 – 不安全的直接对象引用 [与A7合并]	∪	A4:2017 – XML外部实体（XXE） [新]
A5 – 安全配置错误	↘	A5:2017 – 失效的访问控制 [合并]
A6 – 敏感信息泄漏	↗	A6:2017 – 安全配置错误
A7 – 功能级访问控制缺失 [与A4合并]	∪	A7:2017 – 跨站脚本（XSS）
A8 – 跨站请求伪造 （CSRF）	☒	A8:2017 – 不安全的反序列化 [新，来自于社区]
A9 – 使用含有已知漏洞的组件	→	A9:2017 –使用含有已知漏洞的组件
A10 – 未验证的重定向和转发	☒	A10:2017 – 不足的日志记录和监控 [新，来自于社区]

注入漏洞

- 将不受信任的数据作为命令或查询的一部分发送到解析器时，会产生诸如SQL注入、NoSQL注入、OS 注入和LDAP注入的注入缺陷。攻击者的恶意数据可以诱使解析器在没有适当授权的情况下执行非预期命令或访问数据。
- 注入能导致数据丢失、破坏或泄露给无授权方，缺乏可审计性或是拒绝服务。注入有时甚至能导致主机被完全接管。

失效的身份认证

- 通常，通过错误使用应用程序的身份认证和会话管理功能，攻击者能够破译密码、密钥或会话令牌，或者利用其它开发缺陷来暂时性或永久性冒充其他用户的身份。
- 攻击者只需要访问几个帐户，或者只需要一个管理员帐户就可以破坏我们的系统。根据应用程序领域的不同，可能会导致放任洗钱、社会安全欺诈以及用户身份盗窃、泄露法律高度保护的敏感信息。

敏感数据泄露

- 许多Web应用程序和API都无法正确保护敏感数据，例如：财务数据、医疗数据和PII数据。攻击者可以通过窃取或修改未加密的数据来实施信用卡诈骗、身份盗窃或其他犯罪行为。未加密的敏感数据容易受到破坏，因此，我们需要对敏感数据加密，这些数据包括：传输过程中的数据、存储的数据以及浏览器的交互数据。
- 这个领域的错误频繁影响那些本应该加密的数据。通常情况下，这些数据通常包括很多个人敏感信息（PII），例如：医疗记录、认证凭证、个人隐私、信用卡信息等。这些信息受到相关法律和条例保护，例如：欧盟《通用数据保护条例》（GDPR）和地方隐私保护法律。

XML外部实体（XXE）

- 许多较早的或配置错误的XML处理器评估了XML文件中的外部实体引用。攻击者可以利用外部实体窃取使用URI文件处理器的内部文件和共享文件、监听内部扫描端口、执行远程代码和实施拒绝服务攻击。
- XXE缺陷可用于提取数据、执行远程服务器请求、扫描内部系统、执行拒绝服务攻击和其他攻击。

失效的访问控制

- 未对通过身份验证的用户实施恰当的访问控制。攻击者可以利用这些缺陷访问未经授权的功能或数据，例如：访问其他用户的帐户、查看敏感文件、修改其他用户的数据、更改访问权限等。
- 攻击者可以冒充用户、管理员或拥有特权的用户，或者创建、访问、更新或删除任何记录。

安全配置错误

- 安全配置错误是最常见的安全问题，这通常是由于不安全的默认配置、不完整的临时配置、开源云存储、错误的 HTTP 标头配置以及包含敏感信息的详细错误信息所造成的。因此，我们不仅需要对所有的操作系统、框架、库和应用程序进行安全配置，而且必须及时修补和升级它们。
- 这些漏洞使攻击者能经常访问一些未授权的系统数据或功能。有时，这些漏洞导致系统的完全攻破。

跨站脚本（XSS）

- 当应用程序的新网页中包含不受信任的、未经恰当验证或转义的数据时，或者使用可以创建 HTML或JavaScript 的浏览器 API 更新现有的网页时，就会出现 XSS 缺陷。XSS 让攻击者能够在受害者的浏览器中执行脚本，并劫持用户会话、破坏网站或将用户重定向到恶意站点。
- XSS对于反射和DOM的影响是中等的，而对于存储的XSS，XSS的影响更为严重，譬如在受攻击者的浏览器上执行远程代码，例如：窃取凭证和会话或传递恶意软件等。

不安全的反序列化

- 不安全的反序列化会导致远程代码执行。即使反序列化缺陷不会导致远程代码执行，攻击者也可以利用它们来执行攻击，包括：重播攻击、注入攻击和特权升级攻击。
- 反序列化缺陷的影响不能被低估。它们可能导致远程代码执行攻击，这是可能发生的最严重的攻击之一。

使用含有已知漏洞的组件

- 组件（例如：库、框架和其他软件模块）拥有和应用程序相同的权限。如果应用程序中含有已知漏洞的组件被攻击者利用，可能会造成严重的数据丢失或服务器接管。同时，使用含有已知漏洞的组件的应用程序和API可能会破坏应用程序防御、造成各种攻击并产生严重影响。
- 虽然对于一些已知的漏洞其影响很小，但目前很多严重的安全事件都是利用组件中的已知漏洞。根据你所要保护的资产，此类风险等级可能会很高。

不足的日志记录和监控

- 不足的日志记录和监控，以及事件响应缺失或无效的集成，使攻击者能够进一步攻击系统、保持持续性或转向更多系统，以及篡改、提取或销毁数据。大多数缺陷研究显示，缺陷被检测出的时间超过200天，且通常通过外部检测方检测，而不是通过内部流程或监控检测。
- 多数成功的攻击往往从漏洞探测开始。允许这种探测会将攻击成功的可能性提高到近100%，攻击者依靠监控的不足和响应的不及时来达成他们的目标而不被知晓。

其他常见漏洞——文件上传漏洞

- 文件上传漏洞是指由于程序员在对用户文件上传部分的控制不足或者处理缺陷，而导致的用户可以越过其本身权限向服务器上上传可执行的动态脚本文件。这里上传的文件可以是木马，病毒，恶意脚本或者WebShell等。这种攻击方式是最为直接和有效的，“文件上传”本身没有问题，有问题的是文件上传后，服务器怎么处理、解释文件。如果服务器的处理逻辑做的不够安全，则会导致严重的后果。
- 文件上传漏洞本身是一个危害巨大的漏洞，WebShell更是将这种漏洞的利用无限扩大。大多数的上传漏洞被利用后攻击者都会留下WebShell以方便后续进入系统。攻击者在受影响系统放置或者插入WebShell后，可通过该WebShell更轻松，更隐蔽的在服务中为所欲为。

其他常见漏洞——web中间件相关漏洞

- web中间件用于提供系统软件和应用软件之间的连接，以便于软件各部件之间的沟通，其可以为一种或多种应用程序提供容器。中间件漏洞可以说是最容易被web管理员忽视的漏洞，因为这并不是应用程序代码上存在的漏洞，而是属于一种应用部署环境的配置不当或者使用不当造成的。
- 利用中间件（如jboss/weblogic/websphre/tomcat）相关的漏洞可以直接获取目标服务器的控制权限。

实验环境简介

http://172.23.3.19/site/login

使用环境：

1. 浏览器IE9、IE10、IE11、chrome浏览器、火狐
2. 分辨率1024*768;1280*800;1366*768;1440*900
3. 操作系统：Win7、Win8、winXP、Win2003、Win10。

实验一——SQL注入漏洞

- SQL注入是黑客对数据库进行攻击的常用手段之一，是一种通过操作输入来修改后台操作语句达到执行恶意sql语句来进行攻击的技术。SQL注入按变量类型可分为数字型注入和字符型注入，按注入方式可分为：
- 基于报错的注入，即页面会返回错误信息，或者把注入的语句的结果直接返回在页面中；
- 布尔型盲注，即可以根据返回页面判断条件真假的注入；
- 基于时间的注入，即不能根据页面返回内容判断任何信息，用条件语句查看时间延迟语句是否执行（即页面返回时间是否增加）来判断；
- 联合查询注入，可以使用union的情况下的注入；
- 堆查询注入，可以同时执行多条语句的执行时的注入。

实验二——暴力破解

- 暴力破解攻击是指攻击者通过系统地组合并尝试所有的可能性以破解用户的用户名、密码等敏感信息。攻击者往往借助自动化脚本工具来发动暴力破解攻击。
- 根据暴力破解的穷举方式，其攻击行为可以分为：
  - **字典攻击法**，使用字典中已存在的用户名、密码进行猜破
  - **穷举法**，攻击者首先列出密码组合的可能性（如数字、大写字母、小写字母、特殊字符等），然后按密码长度从1位、2位....构成不同的账号和密码对，然后逐个猜试
  - **组合式攻击法**，使用字典攻击和穷举法的组合攻击方式。

实验三——文件上传

- 文件上传漏洞是指由于程序员在对用户文件上传部分的控制不足或者处理缺陷，而导致的用户可以越过其本身权限向服务器上上传可执行的动态脚本文件。这里上传的文件可以是木马，病毒，恶意脚本或者WebShell等。这种攻击方式是最为直接和有效的，“文件上传”本身没有问题，有问题的是文件上传后，服务器怎么处理、解释文件。如果服务器的处理逻辑做的不够安全，则会导致严重的后果。
- 根据对服务器对上传文件的检测方式分类如下：
  - 客户端javascript校验
  - 服务端content-type校验
  - 服务端文件内容头校验
  - 服务端文件后缀名黑/白名单校验等

实验四——跨站脚本

- 当应用程序的新网页中包含不受信任的、未经恰当验证或转义的数据时，或者使用可以创建 HTML或JavaScript 的浏览器 API 更新现有的网页时，就会出现 XSS 缺陷。XSS 让攻击者能够在受害者的浏览器中执行脚本，并劫持用户会话、破坏网站或将用户重定向到恶意站点。
- xss类型一般分为三种：
  - 反射型XSS，把用户输入的数据“反射”给浏览器
  - 存储型XSS，把用户输入的数据“储存”在服务器端
  - DOM Based XSS，基于DOM的跨站，由于客户端脚本自身解析不正确导致的。

入侵防御

- 网络服务存在大量漏洞，例如心脏滴血（2014）、永恒之蓝（2017）等
- 安全策略的局限，只能根据流量的属性来决定允许或丢弃，不能识别是否存在攻击行为。
- 入侵防御（IPS，Intrusion Prevention System）特性

IPS签名

- 所谓签名，就是描述了网络入侵行为的一条规则，IPS通过比较报文特征和签名来检测和防御入侵行为。除了特征之外，签名中还包含处理动作，以及一些必要的信息，下面给出了一个典型的心脏滴血签名，签名信息显示了此漏洞相关的一些信息（具体的漏洞规则是不呈现的）：

- NGFW提供了IPS特征库（签名库），里面包含了针对各种已知攻击行为的签名信息。网络攻击层出不穷，为了识别出新的攻击行为，必须定期更新IPS特征库，才能够更好地防御攻击行为，保证网络安全。

实验1-IPS配置

- **实验目的**
  - 掌握入侵防御配置文件的配置方法。
- **配置思路**
  - 配置入侵防御安全配置文件
  - 在安全策略中引用安全配置文件
- **操作步骤**
  - IPS特性的主体配置是入侵防御配置文件和安全策略。入侵防御配置文件中定义了签名过滤器和例外签名；安全策略中定义了匹配条件（对哪些流量进行IPS检测）、动作（必须为允许），然后引用入侵防御配置文件。
- **验证：**
  - 使用DVWA作为带有漏洞的服务器，在攻击机上发起攻击，观察防御效果，查看IPS日志。

思考题

1. 如何开发更安全的web应用?


谢 谢
www.huawei.com