从WAG角度看Web安全威胁

2011-1-8

ye_runguo@venustech.com.cn
叶润国 北京启明星辰信息安全技术有限公司

讨论提纲

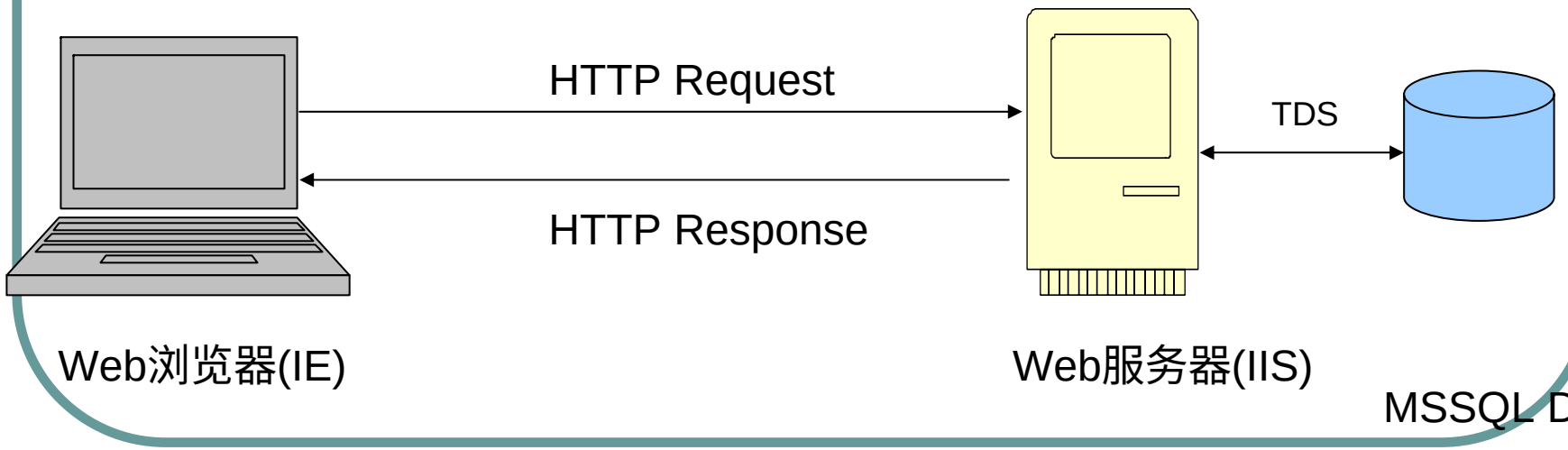
- 认识Web威胁
- WAF防御Web攻击
- 阅读资料推荐

Web系统基本框架(B/S)

Web应用便利性使得很多传统C/S应用都变换到了基于Web的B/S模式，并且这种趋势将越来越明显。

■政务网站、政务办公、企业ERP系统，网络银行

最好例子：启明星辰安全产品都B/S化了!



Web安全：安全问题重中之重

- 根据 Gartner 的调查，信息安全攻击有 75% 都是发生在 Web 应用层而非网络层面上。同时，数据也显示，2/3的 Web 站点相当脆弱，易受攻击。
- **2008年4月9日**赛门铁克公司今天发布了第十三期互联网安全威胁报告，该报告显示，网页已取代网络成为攻击活动的主要渠道，越来越多的在线用户会因为访问一些日常的网站而受到感染。
  - 报告数据基于上百万个互联网传感器、第一手的资料调查以及对于攻击活动的主动监控收集而成，为用户提供针对全球互联网安全现状的分析。

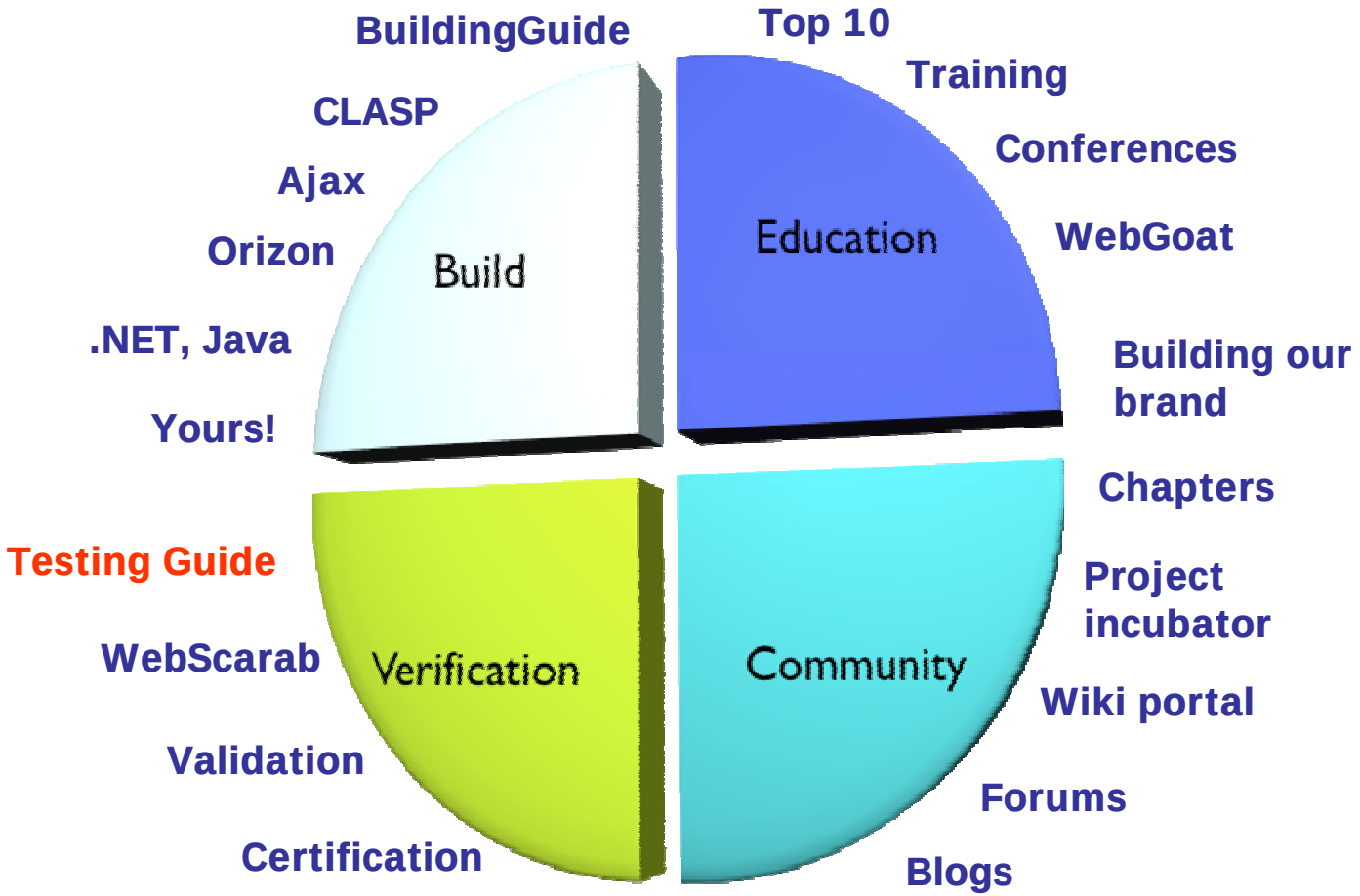
常见的Web攻击威胁

- Web服务器攻击
  - 包含在发送给Web服务器的HTTP请求中
  - CGI扫描、SQL注入攻击、CSRF攻击、XPath注入攻击。。。
- Web客户端攻击
  - 包含在由Web服务器返回给Web客户的Web页面中
  - 网络钓鱼、存储式XSS攻击、网页木马、Web ShellCode。

威胁相关概念

- **Threat**: A potential violation of security -ISO 7498-2
- **Impact**: consequences for an organization or environment when an attack is realized, or weakness is present.
- **Attack**: a Well-defined set of actions that, if successful, would result in either damage to an asset, or undesirable operation.
- **Vulnerability**: an occurrence of a weakness within software, in which the weakness can be used by a party to casue the software to modify or access unintended data, interrupt proper execution, or perform incorrect actions that were not specifically granted to the party who uses the weakness.
- **weakness**: a type of mistake in software that, in proper conditions, could contribute to the introduction of vulnerabilities within the software.

OWASP
The Open Web Application Security Project



OWASP Top 10

OWASP Top 10 – 2007 (Previous)	OWASP Top 10 – 2010 (New)
A2 – Injection Flaws	A1 – Injection
A1 – Cross Site Scripting (XSS)	A2 – Cross-Site Scripting (XSS)
A7 – Broken Authentication and Session Management	A3 – Broken Authentication and Session Management
A4 – Insecure Direct Object Reference	A4 – Insecure Direct Object References
A5 – Cross Site Request Forgery (CSRF)	A5 – Cross-Site Request Forgery (CSRF)
<was T10 2004 A10 – Insecure Configuration Management>	A6 – Security Misconfiguration (NEW)
A8 – Insecure Cryptographic Storage	A7 – Insecure Cryptographic Storage
A10 – Failure to Restrict URL Access	A8 – Failure to Restrict URL Access
A9 – Insecure Communications	A9 – Insufficient Transport Layer Protection
<not in T10 2007>	A10 – Unvalidated Redirects and Forwards (NEW)
A3 – Malicious File Execution	<dropped from T10 2010>
A6 – Information Leakage and Improper Error Handling	<dropped from T10 2010>

STRIDE威胁模型

- STRIDE：威胁分类模型(Microsoft)
  - Spoofing(假冒):
  - Tampering(窜改)：
  - Repudiation(否认)：
  - Information Disclosure(信息揭露)
  - Denial of Service(服务阻断)
  - Elevation of Privilege(特权提升)

DREAD

- 一种威胁风险分析计算模型(Microsoft)
  - Damage Potential (潜在的破坏性）
  - Reproducibility (再现性)
  - Exploitability (可利用性)
  - Affected Users (受影响用户)
  - Discoverability (可发现性)
- 风险定义：损失发生的可能性
  - R=Probability*Loss

WASC的Web威胁分类

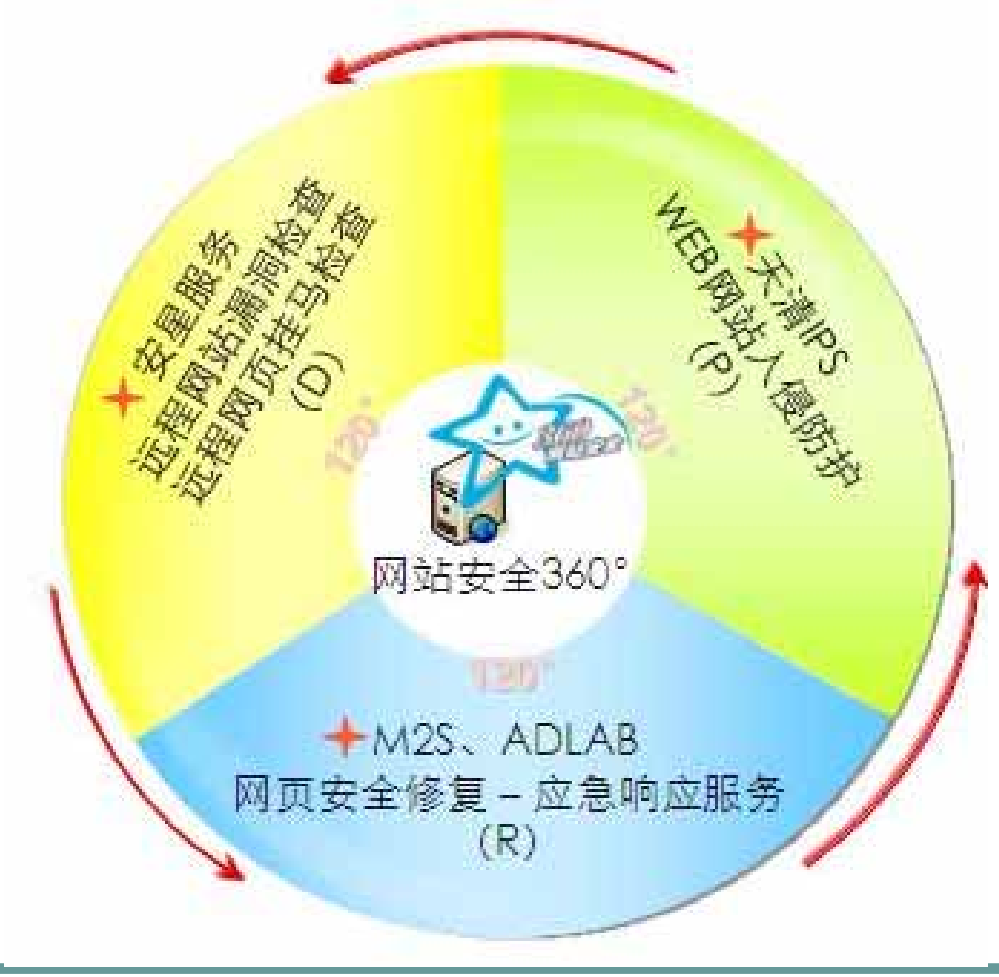
- WASC是一个由安全专家、行业顾问和诸多组织的代表组成的国际团体。他们负责为 WWW 制定被广为接受的应用安全标准。
- WASC 组织关键项目之一是“Web 安全威胁分类”，将 Web 应用所受到的威胁、攻击进行说明并归纳成具有共同特征的分类.
- *网址：http://www.webappsec.org/*

W

Attacks	Weaknesses
Abuse of Functionality	Application Misconfiguration
Brute Force	Directory Indexing
Buffer Overflow	Improper Filesystem Permissions
Content Spoofing	Improper Input Handling
Credential/Session Prediction	Improper Output Handling
Cross-Site Scripting	Information Leakage
Cross-Site Request Forgery	Insecure Indexing
Denial of Service	Insufficient Anti-automation
Fingerprinting	Insufficient Authentication
Format String	Insufficient Authorization
HTTP Response Smuggling	Insufficient Password Recovery
HTTP Response Splitting	Insufficient Process Validation
HTTP Request Smuggling	Insufficient Session Expiration
HTTP Request Splitting	Insufficient Transport Layer Protection
Integer Overflows	Server Misconfiguration
LDAP Injection	
Mail Command Injection	
Null Byte Injection	
OS Commanding	
Path Traversal	
Predictable Resource Location	
Remote File Inclusion (RFI)	
Routing Detour	
Session Fixation	
SOAP Array Abuse	

如何防御Web攻击威胁

- PDR模型



传统安全设备局限性

- Web攻击属于应用层攻击，不同于传统的网络层或操作系统层攻击
- 传统安全设备用来防御来自网络层的安全攻击，无法防御应用层网络攻击
  - 传统防火墙不对80端口网络流量进行检查
  - VPN和SSL网络层加密设备可以抵抗网络层数据泄密和伪造，但无法抵御Web安全攻击。
  - 传统基于攻击特征字匹配技术的IDS/IPS产品无法检测和抵御Web安全攻击,Web安全攻击变化多端，攻击特征无法枚举。

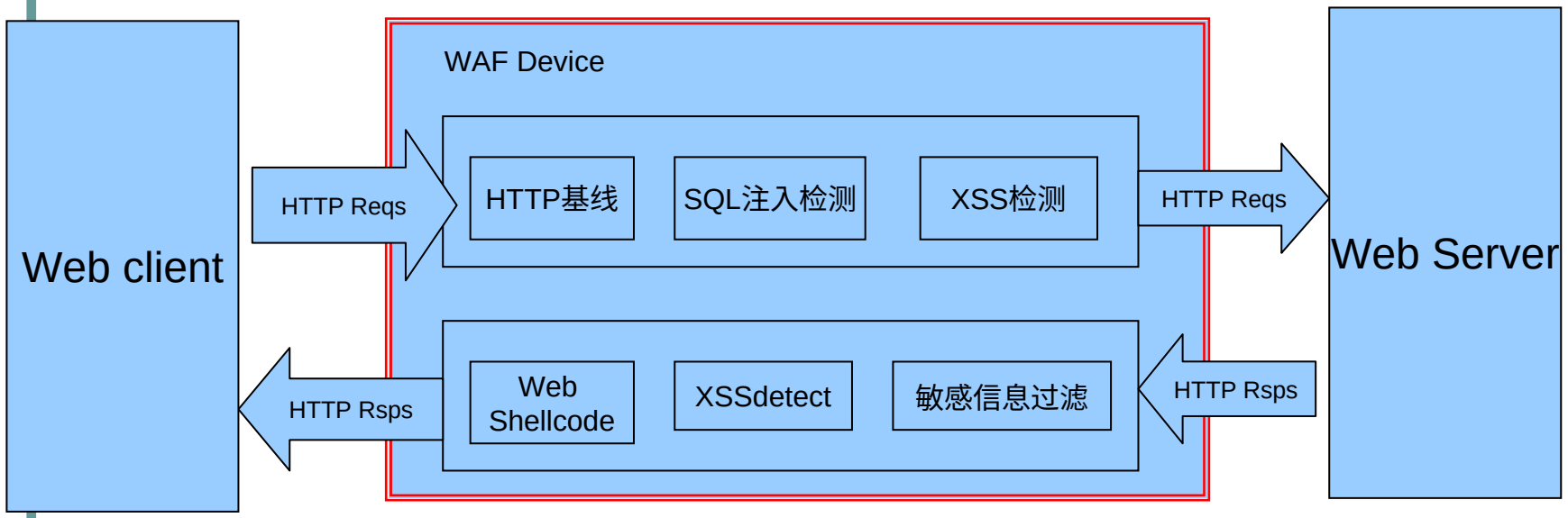
我们需要新的安全设备

- Web应用防火墙(WAF)
  - WAF代表了一类新兴的信息安全技术，用以解决诸如防火墙一类传统设备束手无策的Web应用安全问题。
  - 与传统防火墙不同，WAF工作在应用层，因此对Web应用防护具有先天的技术优势。
  - 基于对Web应用业务和逻辑的深刻理解
    - WAF对来自Web应用程序客户端的各类请求进行内容检测和验证，确保其安全性与合法性，对非法的请求将予以实时阻断，从而对各类网站站点进行有效防护。
    - WAF对由Web应用程序服务端返回的Web页面进行内容检测和过滤，从而确保Web客户端的安全。
- WAG?

从WIPS到WAF的演变

- WIPS：从传统IDS演变而来，增加了报文转发和恶意报文过滤功能
  - 复杂的流重组、协议解析等预处理工作
  - 只实现对HTTP请求的微观层面的安全检查，没有实现对HTTP响应的安全检查
- WAF：从反向Web代理演变而来，增加了对HTTP请求和HTTP响应的安全检查功能
  - 可以实现对HTTP流的双向的安全检查，从而可以实现对后台Web服务的全面防御

典型WAF功能模块图


Web client
HTTP Reqs
WAF Device
HTTP基线
SQL注入检测
XSS检测
HTTP Reqs
Web Server
HTTP Rsps
Web Shellcode
XSSdetect
敏感信息过滤
HTTP Rsps

当前两大入侵检测阵营

- **异常检测(白名单模型）**：检测与可接受行为之间的偏离程度。
  - 构建正常操作应该具有的用户轮廓，当用户活动与正常行为有重大偏离时即被认为是入侵。
  - 漏报率低，误报率高（可检测未知攻击，检测结果难以解释）。
- **误用检测（黑名单模型）**：检测与已知不可接受行为之间的匹配程度。
  - 收集非正常操作的行为特征，建立相关的特征库，当监测的用户或系统行为与库中的记录相匹配时，系统就认为这种行为是入侵。
  - 误报率低、漏报率高（仅检测已知攻击，特征库更新问题）
- **检测趋势**：异常检测和误用检测共存（互补性）

现有WAF检测能力限制

- 现有的WIPS/WAF产品中主要是针对HTTP请求进行入侵检测，当发现攻击时及时阻断
- 目前在WAF产品中所实现的入侵检测方法都为误用检测方法，包括SQL注入和XSS攻击检测
- 所有这些都是一种微观层面的基于误用检测模型的安全检测技术，只考虑了单个HTTP请求，没有从宏观和全局角度来考虑问题，因此不能构成对Web服务的全面安全防御。

WAF技术发展方向

- 误用检测和异常检测同时使用
  - 误用检测用于对付已知攻击
    - SQL注入、XSS
  - 异常检测用于对付未知攻击
    - 变形攻击、0-day漏洞攻击
- 主动感知Web应用系统上下文
  - 主动获知Web系统的网页结构
    - Web扫描器
  - 主动监控Web服务器的性能情况
    - 监控平均响应时间，监控服务器吞吐率

WAF还将面临哪些新的威胁

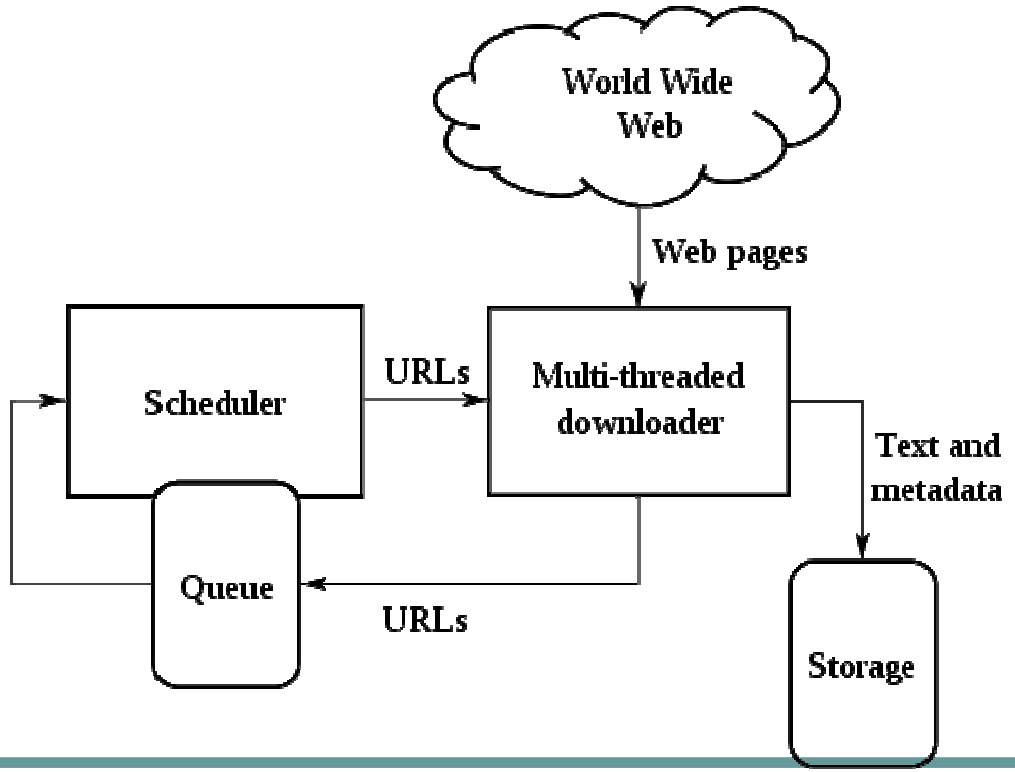
- 如何检测针对Web服务器的恶意扫描行为？
- 如何检测和防御Web-DDoS？
- 如何检测Web请求中的变形和未知攻击？
- 如何检测Web挂马行为？
- 如何检测和防御跨站请求伪造攻击？
- 如何防止Web系统信息泄露？

WAF识别恶意爬虫

- 我们发现，很多Web攻击发生时，它们所使用的Web攻击工具大都具有一种Web爬虫行为。
  - CC攻击(DDoS)：采用多个代理并行访问Web服务器上那些资源消耗较多的Web页面，导致Web服务DDoS
  - 僵尸DDoS：采用一组运行了Web爬虫的僵尸来不间断的爬Web服务器，使得Web服务器没有时间接待其它Web请求
  - Web漏洞扫描(包括SQL注入工具）：黑客采用常见的漏洞扫描工具对Web服务器进行漏洞扫描

Web爬虫示意图

- 网页获取=DOM解析+URL提取



如何识别Web爬虫行为(1)?

- 监测某Web客户端发出的一系列网页请求之间的时间间隔情况：
  - 如果为爬虫，则它发出的两个连续Web页面请求的时间间隔取较小值的概率较大；
  - 如果该Web客户端为正常用户，则它发出的两个连续Web页面请求的时间间隔取较大值的概率较大。
  - 假设检验/DS证据理论

WAF如何识别Web爬虫行为(2)?

- URL蜜罐方法
  - WAF修改网页，在网页中嵌入人眼看不见的超级链接，人眼看不见该超链，因此不会点击，但Web爬虫通过DOM树解析可以看到这个超级链接，因此会访问该URL。
  -
  - <img src="1_1.jpg" width=1px height=1px>

WAF如何识别Web爬虫(3)?

- 键盘或鼠标事件侦测法
  - 如果是人工浏览，则肯定会触发键盘事件或鼠标事件
  - 因此，如果监测到键盘或鼠标事件，则认为是人工浏览行为

- <script>document.onmousemove=my;
- Function my()
- {var img= new Image();
  img.src="/mousemovedevent.asp";}
- </script>

如何检测和防御Web-DDoS?

- 针对Web应用的DDoS攻击也称为HTTP-Flood攻击；
- 攻击者采用大量的HTTP访问请求高频率的访问被攻击Web网站，以至于消耗掉该Web网站的大部分资源，没有精力来为合法用户提供服务。
  - CC攻击：基于代理请求大文件或耗时多的HTTP请求
  - Web僵尸：发送大量的HTTP请求
  - 恶意的大量的Web爬虫同时爬你的Web网站

传统Web-DDoS检测防御方法弱点

- 传统Web-DDoS检测的缺点
  - 采用固定阈值来做检测，比如，每秒钟的HTTP请求数量超过700个，则认为发生了HTTP-Flood攻击，问题是：这个阈值是和环境相关的，如何设定这个固定阈值？
- 传统Web-DDoS防御缺点
  - 不能很好地针对不同的Web客户端类别做区分处理，采取一视同仁的处理方案

新的Web-DDoS检测和防御方法要求

- 不要依赖太多的经验参数来做Web-DDoS检测
- 能够正常的区分DDoS流量和正常访问流量，从而能够对DDoS流量进行有效的过滤，同时允许正常流量通过。
- 能够持续监控Web服务器性能变化情况，发现性能急剧下降时，应提前行动。

一个共同难题

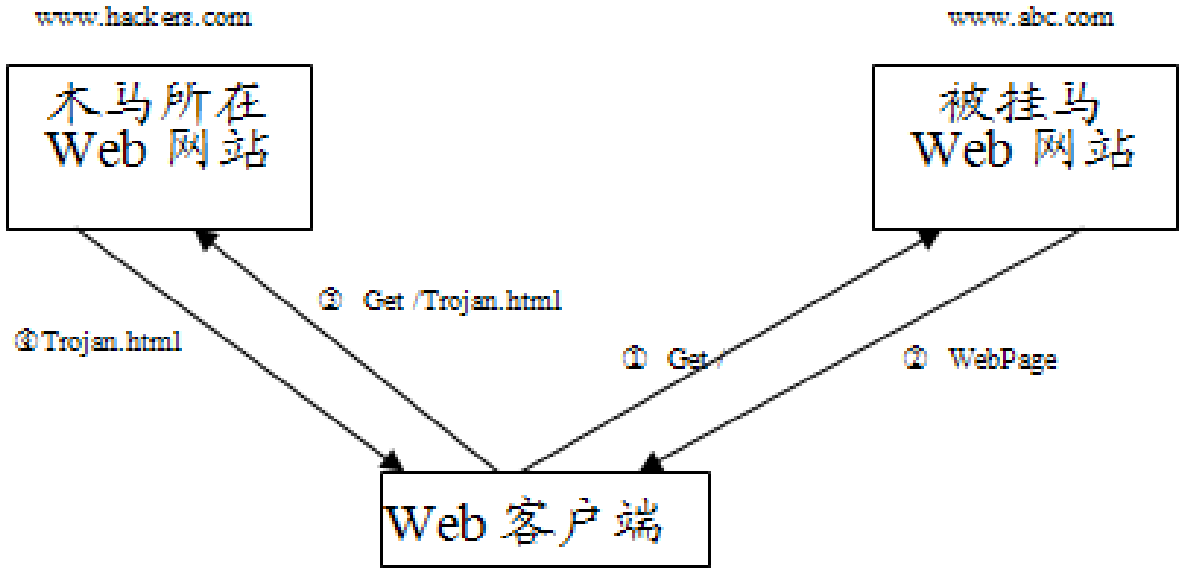
- 有大量的Web客户端通过NAT或者代理访问被WAF保护的Web业务系统：大多数Web客户为正常客户，只有一个人为恶意客户，如何将该恶意客户的Web流量与其它正常客户的Web流量进行区分并区别对待？
- 需要设计一个能够有效标识各Web客户端流量的方法。
  - Web爬虫识别和Web-DDoS需要共同面对的问题

如何检测Web请求中的变形和未知攻击

- 使用异常检测方法
  - 正向模型方法
- 胡博士的HTTP请求基线的入侵检测方法
  - 难点
    - 不是所有的异常都是攻击
    - 如何准确建立HTTP请求基线：手工、自学习
    - 如何实现基于HTTP请求基线的快速匹配

网页挂马原理图

<iframe src="http://www.hackers.com/trojan.html">



附图 1 受害者访问被挂马网站时的攻击过程

如何检测Web挂马行为?

- 传统方法
  - Web爬虫+沙箱方法：安星2
  - 不适合WAF
- WAF中方法
  - 误用检测方法：
    - 常见恶意URL匹配
  - 异常检测方法
    - 为关键网页建立网页基线，所述网页基线为该网页允许的内嵌URL集合，检测时通过基线比对法发现不速之客。
  - 共同难点：WAF如何提取出网页中所有的内嵌URL
    - 静态DOM、动态DOM解析

如何检测跨站请求伪造攻击?

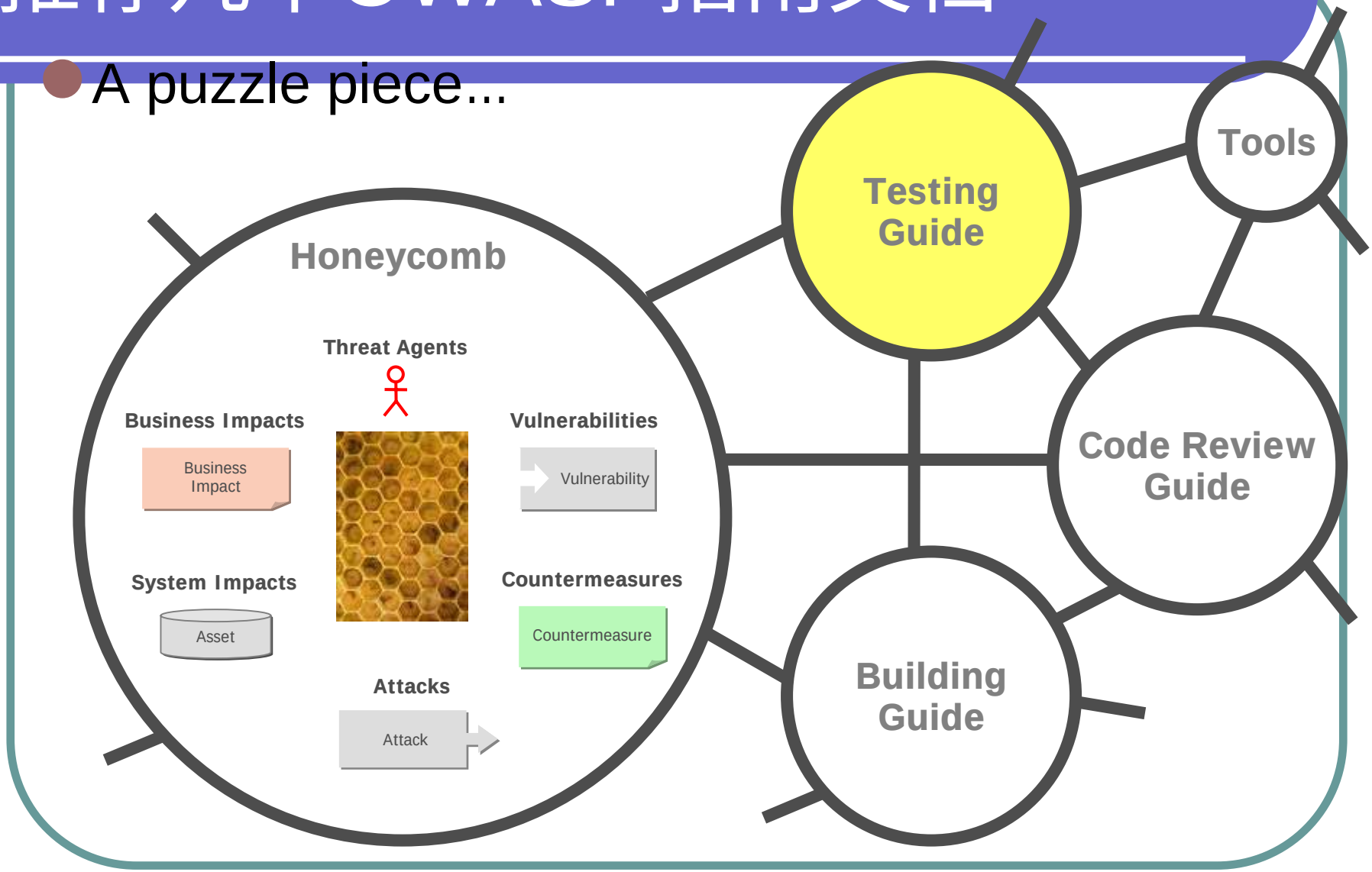
- CSRF漏洞根源
  - 用户认证后的凭证信息存储在Cookie中，Cookie在HTTP请求过程中被自动携带；
  - 整个HTTP请求中的参数都可以事先预知，因此攻击者可以伪造HTTP请求。
- 如何防御CSRF攻击
  - 加入随机因子，使得HTTP请求中的某个域不可成功伪造
    - 图形验证码、表单Token、Referer-Token等。
- <form method=post action=/submitform>
  - <input type=text name=acc><input type=text name=amount>
  - <input type=text name=recipie><input type=submit>
  - <input type=hidden name=token value='aex3j8d4r69grtf3y6tgd4a09e3j6'>

如何防止Web系统信息泄露？

- 对Web系统的返回Web页面内容进行监控和过滤
  - 500, 403错误页面，替换成正常页面
  - 防止泄露网页目录信息
    - URL重写
    - Web页面重构

推荐几个OWASP指南文档

- A puzzle piece...



Thank you for your attention