PTS 文档

2021-12-22

目录

产品描述

介绍

渗透测试是模拟黑客可能使用的攻击技术和漏洞发现技术，采用可控制、非破坏性质的方法和手段对目标系统的安全作深入的探测，发现系统最脆弱的环节，能够直观的让管理人员知道自己系统所面临的问题。

产品优势

安全可控

测试用例可灵活选择，漏洞细节完全私有化，并订制完备的风险规避预案，保障测试过程中业务连续性不受影响

标准的测试规范

遵从业界著名的OSSTMM与OWASP测试框架，选取最佳实践进行操作，有效避免因过程不规范导致的业务异常等风险

成熟丰富的知识库

根据百度多年安全攻防经验，积累总结出一套成熟丰富的渗透测试用例知识库，保障渗透测试内容的全面性以及测试漏洞的深入挖掘

专业安全团队

安全服务团队成员主要由 CISP、CISSP、PMP等经过资质认证的高级工程师组成

应用场景

应用系统

对注入、跨站脚本、越权、CSRF、信息泄露、恶意文件执行、业务风险等威胁进行安全测试

iOS 应用

涵盖iOS客户端、服务端、本地包安全策略、敏感信息、数据通信等方面，对iOS APP进行全面、精细的安全测试

Android应用

涵盖Android客户端、服务端、本地包安全策略、敏感信息、数据通信等方面，对Android APP进行全面、精细的安全测试

全网渗透

对企业全网络的安全进行综合评估与渗透测试，发现网络中存在的风险威胁与漏洞，让管理者了解掌握全网所面临的问题

产品定价

产品定价

您在百度智能云渗透测试产品页面在线提交申请服务表单，提交申请后会有专人与您联系，并为您提供产品报价等服务。

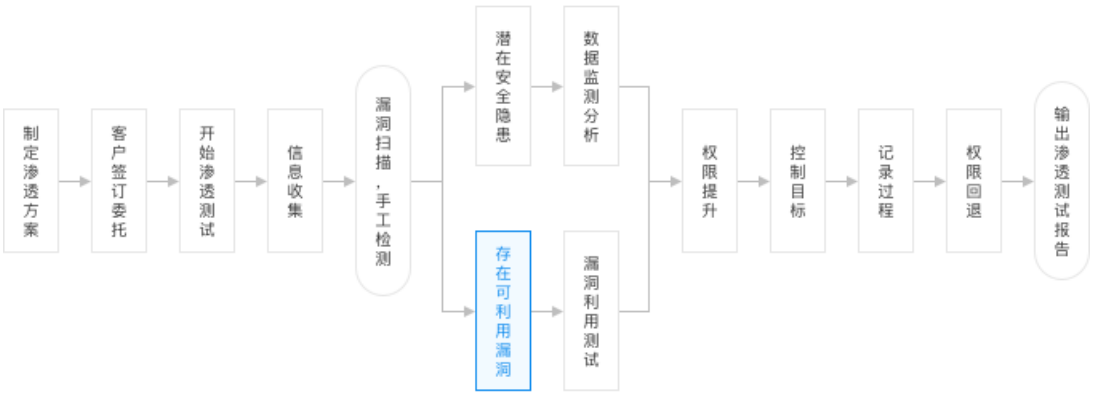
操作指南

开通渗透测试服务

您可以登录百度智能云渗透测试服务官网，然后点击立即申请，完成渗透测试服务在线申请表的填写，并在需求简述中，简单描写您的主要需求，我们会安排相关的人员跟进为您提供服务。

渗透测试流程

渗透测试流程严格依照下图执行，采用可控制的、非破坏性质的渗透测试，并在执行过程中把握好每一个步骤的信息输入/输出，控制好风险，确保对客户的网络不造成破坏性的损害，保证渗透测试前后信息系统的可用性、可靠性保持一致，流程图如下所示：



渗透测试方法

渗透测试模拟黑客的入侵思路与技术手段，黑客的攻击入侵需要利用目标网络的安全弱点，渗透测试也是同样的道理。以人工渗透为主，辅助以攻击工具的使用，这样保证了整个渗透测试过程都在可以控制和调整的范围之内。

针对各应用系统的渗透测试方法包括以下方法但不局限于以下方法：

测试类型	测试描述
信息收集	信息收集是渗透攻击的前提，通过信息收集可以有针对性地制定模拟攻击测试计划，提高模拟攻击的成功率，同时可以有效的降低攻击测试对系统正常运行造成的不利影响。 信息收集的方法包括端口扫描、操作系统指纹判别、应用判别、账号扫描、配置判别等。
端口扫描	通过对目标地址的TCP/UDP端口扫描，确定其所开放的服务的数量和类型，这是所有渗透测试的基础。通过端口扫描，可以基本确定一个系统的基本信息，结合安全工程师的经验可以确定其可能存在以及被利用的安全弱点，为进行深层次的渗透提供依据。
口令猜测	本阶段将对暴露在公网的所有登录口进行口令猜解的测试，找出各个系统可能存在的弱口令或易被猜解的口令。猜解成功后将继续对系统进行渗透测试，挖掘嵌套在登录口背后的漏洞、寻找新的突破口以及可能泄漏的敏感信息，并评估相应的危害性。猜解的对象包括：WEB登录口、FTP端口、数据库端口、远程管理端口等。
远程溢出	这是当前出现的频率最高、威胁最严重，同时又是最容易实现的一种渗透方法，一个具有一般网络知识的入侵者就可以在很短的时间内利用现成的工具实现远程溢出攻击。 对于在防火墙内的系统存在同样的风险，只要对跨接防火墙内外的一台主机攻击成功，那么通过这台主机对防火墙内的主机进行攻击就易如反掌。
本地溢出	本地溢出是指在拥有了一个普通用户的账号之后，通过一段特殊的指令代码获得管理员权限的方法。使用本地溢出的前提是首先要获得一个普通用户的密码。也就是说由于导致本地溢出的一个关键条件是设置不当的密码策略。 多年的实践证明，在经过前期的口令猜测阶段获取的普通账号登录系统之后，对系统实施本地溢出攻击，就能获取不进行主动安全防御的系统的控制管理权限。
脚本测试	脚本测试专门针对Web服务器进行。根据最新的技术统计，脚本安全弱点为当前Web系统尤其存在动态内容的Web系统存在的主要比较严重的安全弱点之一。利用脚本相关弱点轻则可以获取系统其他目录的访问权限，重则将有可能取得系统的控制权限。因此对于含有动态页面的Web系统，脚本测试将是必不可少的一个环节。
权限获取	通过初步信息收集分析，存在两种可能性，一种是目标系统存在重大的安全弱点，测试可以直接控制目标系统；另一种是目标系统没有远程重大的安全弱点，但是可以获得普通用户权限，这时可以通过该普通用户权限进一步收集目标系统信息。接下来尽最大努力取得超级用户权限、收集目标主机资料信息，寻求本地权限提升的机会。这样不停的进行信息收集分析、权限提升的结果形成了整个的渗透测试过程。

渗透测试内容

通过可控的安全测试技术对约定范围内的系统进行渗透测试，同时结合业界著名的OSSTMM与OWASP测试框架组合成最佳实践进行操作。

本次渗透测试将参考OSSTMM测试框架中的以下技术方法：

- 信息收集和状态评估
- 网络节点枚举和探测
- 系统服务和端口扫描验证
- 应用层测试
- 漏洞挖掘与验证

本次渗透测试将参考OWASP 2016最新发布的十大Web应用漏洞排名，并使用测试框架中相应的技术方法：

- SQL注入
- 跨站脚本（XSS）
- 恶意文件执行
- 不安全的直接对象引用

- 跨站请求伪造（CSRF）
- 信息泄漏和错误处理不当
- 认证和会话管理失效
- 不安全的加密存储
- 不安全的通讯
- URL访问失效

渗透测试用例库

为保证渗透测试内容的全面性以及测试漏洞的深入挖掘，百度安全总结了多年的渗透测试经验与丰富的渗透测试用例，以下是百度安全用于渗透测试的用例库，测试条目涵盖了全面/最新的漏洞与测试方法，将根据不同应用系统的特性进行有针对性的匹配测试。

测试类型	测试条目	测试类型
信息收集	目录爬行遍历	这个阶段将通过浏览、目录爬行的方式捕获/收集应用的资源。
	搜索引擎侦测	搜索引擎，比如Baidu、Google，能够用来发现公开发布的网页应用结构或者错误页面等相关问题。
	应用程序发现	本项测试发现以web服务器的网页应用作为目标。本项测试对于发现细节/寻找突破尤为有效，比如发现用于管理的应用脚本，或旧版本的文件/控件，在测试、开发或维护过程中产生的已不用的脚本。
	分析错误代码-信息泄漏	在渗透性测试过程中，网页应用可能泄露原本不想被用户看见的信息。错误码等信息能让测试者了解应用程序使用的有关技术和产品。很多情况下，由于异常处理和程序代码的不合理，甚至不需要任何特殊技术或工具，都很容易触发产生错误代码的条件从而产生错误代码导致被攻击者利用。
配置管理测试	SSL/TLS测试	SSL和TLS是两个以加密的方式为传输的信息提供安全隧道的协议，具有保护、加密和身份认证的功能。 这些安全组件在应用中非常关键，因此确保高强度的加密算法和正确的执行非常重要。 本项测试的模块为：SSL版本、算法、密钥长度、数字证书、有效期。
	基础配置信息测试	Web应用基础架构由于其内在的复杂性和关联性，一个微小的漏洞就可能对同一服务器上的另一个应用程序产生严重的威胁，甚至破坏整个架构的安全。为了解决这些问题，对配置的管理和已知安全问题进行深入审查尤为重要。
	应用程序配置信息测试	通常在应用程序开发和配置中会产生一些没有考虑到的信息，而这些信息暂时被发布后的Web应用程序所隐藏。 这些信息可能从源代码、日志文件或Web服务器的默认错误代码中泄露。
	文件扩展名处理测试	通过Web服务器或Web应用程序上的文件扩展名能够识别出目标应用程序使用的技术，例如扩展名JSP与ASP。文件扩展名也可能暴露与该应用程序相连接的其它系统。
	旧文件、备份文件、未引用文件测试	Web 服务器上存在多余的、可读、可下载的文件，并且用于备份的文件，是信息泄漏的一大源头。因为它们可能包含应用程序或数据库的部分源代码，安装路径以及密码等敏感信息。本项测试验证这些文件是否存在于发布的Web应用系统上。
	应用程序管理接口测试	许多应用程序的管理接口通常使用一个公用路径，路径获取后可能面临猜测或暴力破解管理密码的风险。此项测试目的是找到管理接口，并检测是否可以利用它来获取管理员权限。
	HTTP请求方法与XST测	Web服务器可以配置为多种请求方式，如Get、Post、Put、Delete等，此项测试将鉴定Web服务器是否

	法与XST测试	允许具有潜在危险性的HTTP请求方法，同时鉴定是否存在跨网站追踪攻击（XST）。
认证测试	证书加密通道传输安全性测试	本项测试试图分析用户输入Web表单中的数据，如为了登录网站而输入的登录凭据是否使用了安全的传输协议，以免受到攻击。
	用户枚举测试	本项测试为了验证是否可能通过与应用程序的认证机制交互（提示信息），收集有效的用户。这项测试好于暴力破解，一旦获取有效的用户名后，就可针对性的进行密码攻击。
	字典猜解测试	本项测试鉴定应用系统是否存在默认的用户账户或可猜测的用户名/密码组合（遍历测试）。
	口令暴力猜解测试	当遍历攻击失败，测试者可尝试使用暴力破解的方式进行验证。暴力破解测试肯能可能碰到锁定用户或IP等限制。
	验证绕过测试	本项测试尝试以非常规的方式企图绕过身份认证机制，使得应用程序资源失去正常的保护，从而能够在没有认证的情况下访问这些受保护的资源。
	密码重置/找回漏洞测试	本项测试鉴定应用程序的“忘记密码”功能是否起到足够的保护，检查应用程序是否允许用户在浏览器中存储密码。
	用户注销缓存漏洞测试	检查注销和缓存功能能否得到正确实现。
	多因素认证漏洞测试	多因素身份验证将测试以下认证方式的安全性： 一次性密码（OTP）所生成的验证码; USB加密设备; 基于X.509证书的智能卡; 通过SMS发送的随机一次性密码; 只有合法用户知道的个人信息;
会话管理测试	会话管理测试	本项测试分析会话管理模式和机制，鉴定发送给客户端浏览器的会话验证码的安全性，鉴定是否能够打破这一机制从而绕过用户会话。如：对Cookie实行反向工程，通过篡改Cookies来劫持会话。
	Cookie属性测试	Cookies通常是恶意用户攻击合法用户的关键途径。本项测试将分析应用程序在分派Cookie时如何采取必要的防护措施，以及这些已正确配置的Cookie属性。
	会话固定测试	本项测试鉴定当应用程序在成功验证用户后不再更新Cookie 时，能否找到会话固定漏洞并迫使用户使用攻击者已知的Cookie 。
	会话变量泄漏测试	由于会话验证码联系了用户身份和用户会话，它所代表的是保密信息。本项测试鉴定会话验证码是否暴露在漏洞中，并试着追溯会话攻击。
	CSRF跨站请求伪造测试	跨站伪造请求指在Web应用中，迫使已通过验证的未知用户执行非法请求的方法。本项测试鉴定应用程序是否存在这种漏洞。
授权测试	路径遍历测试	本项测试鉴定是否能够找到一种方法来执行路径遍历攻击并获成功得服务器返回的信息。
	授权绕过测试	本项测试核实如何对某个角色或特权实施授权模式以便获得保留的功能和资源。
	权限提升测试	本项测试确认用户是否可能采用特权提升攻击的方式修改自己在应用程序内部的特权或角色。
	跨站脚本反射测试	反射式跨站脚本攻击 (XSS) 是非持久性跨站脚本攻击的另一个名称。该攻击不会使用存在漏洞的Web应用程序加载，而使用受害者载入的违规的URI 。本项测试将确认应用程序对来自用户提交的恶意代码是否进行了存储或反射处理，对各类非法字符进行了严格过滤。

数据验证测试	存储跨站脚本测试	储存式跨站脚本（XSS）是一种最危险的跨站脚本。允许用户存储数据的Web应用程序都有可能遭受这种类型的攻击。
	SQL 注入测试	SQL注入测试检测是否有可能将数据注入到应用程序中，以便在后端数据库中执行用户定制的SQL查询。如果应用程序在没有合理验证数据的情况下使用用户输入创建SQL查询，那么说明该应用程序存在SQL注入漏洞。成功利用这一类别的漏洞会导致未授权用户访问或操作数据库中的数据。
	Code 注入测试	代码注入测试检测是否有可能在应用程序中注入稍后由Web服务器执行的代码。
	SQL OS Commanding	本项测试将设法通过HTTP请求在应用程序中注入OS命令。
	缓冲区溢出测试（字符串格式）	本项测试将检查不同类型的缓冲区溢出漏洞。
数据验证测试	Web服务信息收集	进行Web 服务测试的第一步是确定WS入口点和链接图标。
	XML架构测试	XML需要有合法的格式才能正确的运作。当服务器端进行XML语句分析时，不合规格的XML将会出错。一个解析器需要在整个XML信息中按照序列的方式彻底运行，这样才能评估XML格式是否合格。XML解析器通常占用较多的CPU资源。某些攻击通过发送大量或者不合规的XML信息来利用这个漏洞。
	SQL XML内容级别测试	内容级别的攻击对象是Web服务及其使用的应用程序的服务器，包括Web服务器、数据库、应用程序服务器、操作系统等等。内容级别攻击向量包括：1）SQL注入/XPath注入 2）缓存溢出 3）命令注入。
	HTTP GET参数/REST测试	许多XML应用程序是通过HTTP GET查询传输参数来使用的。在HTTP GET字符串例如，超长的参数（2048字符）、SQL语句/注入（或OS注入参数）中传输恶意内容时，Web服务将会受到攻击。
	重放测试	重放攻击的威胁在于攻击者可以伪装成一个合法用户的身份，然后不被察觉的情况下进行一些恶意操作。本项测试将检测Web服务是否存在重放漏洞。

渗透测试验证

针对渗透测试过程中发现的漏洞进行层次性的安全加固，特别对于通过渗透测试发现的漏洞进行适应性的补救和加固措施，在保证不影响正常业务的情况下，配合客户实施加固方案。

对于加固后的系统进行重复性的渗透测试验证，以保障漏洞不可利用性，同时验证安全加固措施的有效性。确保客户对外提供安全稳定的服务。

常见问题

使用类问题

渗透测试有哪些收益？

通过渗透测试，可以在发生黑客入侵之前，提前对系统的安全状况进行全面详细的检测，能够直观的让管理人员及时掌握了解系统存在的威胁与脆弱性，对发现的问题进行加固，减少因黑客入侵给企业带来的损失。

会不会对业务造成影响？

百度采取多种风险控制措施，保障渗透测试过程不会对企业正常业务造成影响。

渗透测试与漏洞扫描有什么区别？

漏洞扫描是利用工具对系统可能存在的漏洞与威胁进行自动化扫描体检，检测速度比渗透测试更快，但检测的颗粒度比较粗，

没有渗透测试精细，并且存在误报的情况；而渗透测试是人工对系统可能存在的漏洞与威胁进行手动检测，能够做到漏洞零误报，定位更精准，准确性更高。

如何避免风险？

我们在实施渗透测试中采取以下措施来减小风险：

- 双方确认

进行渗透测试前，必须获得企业的同意与授权。对于任何渗透测试的对象的变更和测试条件的变更也都必须获得双方的同意并达成一致意见，方可执行。

- 工具选择

为防止造成真正的攻击，在渗透性测试项目中，会严格选择测试工具，杜绝因工具选择不当造成的将病毒和木马植入的情况发生。

- 时间选择

为减轻渗透性测试对用户网络和系统的影响，渗透测试安排在不影响系统正常业务运作的时间段进行，具体时间主要限制双方协调和商定的时间范围内。

- 范围控制

百度承诺不会对授权范围之外的网络设备、主机和系统进行漏洞检测、攻击测试，严格按照渗透测试范围内限定的对象进行测试。

- 策略选择

为防止渗透性测试造成用户网络和系统的服务中断，百度在渗透性测试中不使用含有拒绝服务的测试策略，不使用未经系统许可的方式进行渗透测试。

- 项目沟通

在实施过程中，除了确定测试人员以外，还要确定客户方配合人员，建立双方直接沟通的渠道，并保持及时、充分、合理的沟通。

- 系统备份和恢复措施

为避免实际渗透测试过程中可能会发生不可预知的风险，在渗透测试前相关管理人员应对系统或关键数据进行备份、确保相关的日志审计功能正常开启，一旦在出现问题时，可以及时的恢复运转。

在渗透测试过程中，如果出现被评估对象没有响应或中断的情况，应当立即停止测试工作，与客户方配合人员一起分析情况，在确定原因后，及时恢复系统，并采取必要的预防措施（比如调整测试策略）之后，确保对系统无影响，并经客户同意之后才可继续进行。