1.7.5.4.数据库恢复

一旦数据库发生故障，就可以将数据库备份加载到应用系统，使数据库恢复到备份时的状态。差异备份策略进行数据库恢复步骤如下：

1．恢复全备份。从备份服务器上取得最近一次完全备份，将数据恢复到上次做完全备份的时间点。以上述备份方案为基础，假设在星期四上午 10：00，系统发生了灾难性的崩溃，需要用上周的完全备份。并将数据恢复到这个时刻。

2．恢复增量备份。恢复完全备份，从上次完全备份到故障发生这段时间的数据还是不完整的，因此需要通过增量备份产生的文件来进行数据恢复。从备份服务器上获取从上次完全备份完毕后到故障点之间的增量备份文件，利用 MySQL 的 Mysqlbinlog 恢复工具进行数据恢复。以 1 中的例子为例。从备份服务器取得星期一至星期四的增量备份文件。利用恢复工具将数据恢复到星期四凌晨的状态。

1.7.6.安全访问策略

身份认证安全解决方案。身份认证技术是对用户信息进行有效鉴别的技术，能够有效阻止非授权用户入侵。由于网络系统身份认证方式多以静态口令为主，因此存在一定问题，主要体现在以下几点：

易破解：用户选择常用词作为密码，容易被破译工具的破解；

密码泄露：人员流动性相对较大，授权密码可能出现被带出的情况，或多次使用同一密码；

不法分子会从电话线或者网络上进行密码截取，会轻易获得用户重要信息；

内部人员通过合法手段获得授权之后，没有按照相应规范对其进行使用。

为防止非法人员或者外部人员对电子政务系统展开越权访问或者非法访问，政府机构有必要对重要部门或者权限较高的用户身份展开严格审查。可通过实施动态密码认证的方式，准确对用户身份信息进行识别。

动态口令认证系统主要由认证服务器、后备服务器以及管理工作站所组成。其中认证服务器作为整体系统核心，安装在市级政务云计方，会通过和服务器相连的方式，对上网用户网络访问情况展开全面监控，以对其身份信息进行认证，

并提供与其身份级别相符的权限与资源。通常认证服务器主要由加密算法软件、实时运算以及认证管理等几部分所组成，该服务器有着较强的数据安全保护功能，会对所有数据进行加密与储存，并会在进行数据交换时，以加密的方式对数据进行传输。管理工作站则主要负责认证服务器管理界面供给工作，会在认证服务器与网络管理员间设置操作截面，以便管理人员展开用户管理与系统维护等一系列工作。

网络访问控制与隔离。在与公共信息网络或者其他网络进行连接时，需要实施物理隔离。政府部门在进行网络设计过程中，要按照安全需求与实际用途，对系统子网进行合理分布，有效降低网络结构安全风险。

1.7.6.1.安全物理隔离

因电子政务网络性质较为特殊，不建议与公共网络直接进行连接。因为与公共网络连接有着极大的安全风险，很容易会受到病毒或者黑客的攻击，所以在进行政务网络连接时，要做好内网与外网物理隔离，并要在内网内设置物理隔离卡，以降低公网用户对内网信息的攻击。

1.7.6.2.虚拟专用网

VPN 即虚拟专用网，是通过一个公用网络建立一个临时的、安全的连接，是一条穿过混乱的公用网络的安全、稳定的隧道。通过它可以帮助远程用户同政府的内部网建立可信的安全连接，并保证数据的安全传输。

1.7.6.3.配备防火墙

配置防火墙是较为常用的一种物理隔离技术，该项技术具有有效性较高以及经济、安全等方面的优势。通过对防火墙的运用，能够有效做好内外部网络访问控制与隔离工作，能够实现双向以及单向控制模式，可以对网络流量与时间形成有效管控。此外，借助防火墙网络地址转换功能，还可以对内网机构形成有效隐藏，能够妥善解决合法 IP 不足等方面的问题。

1.7.7.VPN 访问的安全加固

1.7.7.1.VPN 设备的安全管理

将 VPN 设备的管理界面和用户界面进行分离，采取 ACL 控制，VPN 设备的管理界面只能通过堡垒机进行访问；限制不必要端口如设备的 Redis 等被非授权 IP 访问；加强威胁情报，监控 VPN 设备的漏洞，以便及时对设备进行升级及加固。

1.7.7.2.VPN 用户的访问控制策略

加强 VPN 用户的管控，加强对特权人员的管理，设置策略对于设备、信息系统的运维人员进行严格管理，并从策略上限制此类人员登录 VPN 后只能访问堡垒机 IP。此外，堡垒机开启双因素认证登录，禁止在堡垒机中存储相关密码或登录凭据。

1.7.7.3.VPN 日志审计

构建完备的日志体系，VPN 系统记录日志最少应包括账号、登录 IP、获取 IP、访问 IP、访问协议、访问 URL 及并发等信息。并且，VPN 系统要与安全产品采用统一 NTP Server，保证各种设备日志时间的一致性。此外，VPN 系统要采用 Syslog、Rsync、Kafaka 等与第三方日志或态势平台进行对接，经过综合日志分析可以在第一时间内进行分析研判并有效溯源。

1.8. 同步使用安全保障措施

1.8.1.制定安全管理制度

依据国家、省、市各级网络安全相关政策、标准、规范，制定并落实与等级保护对象安全管理向配套的、包括等级保护对象的建设、开发、运行、维护、升级和改造等各个阶段和环节所应遵循的行为规范和操作规程。包括以下内容：

➢ 应用范围明确

管理制度建立首先要明确制度的应用范围，如机房管理、账户管理、远程访问管理、特殊权限管理、设备管理、变更管理、资源管理等方面。

➢ 行为规范规定

管理制度是通过制度化、规范化的流程和行为约束，来保证各项管理工作的规范性。

➢ 评估与完善

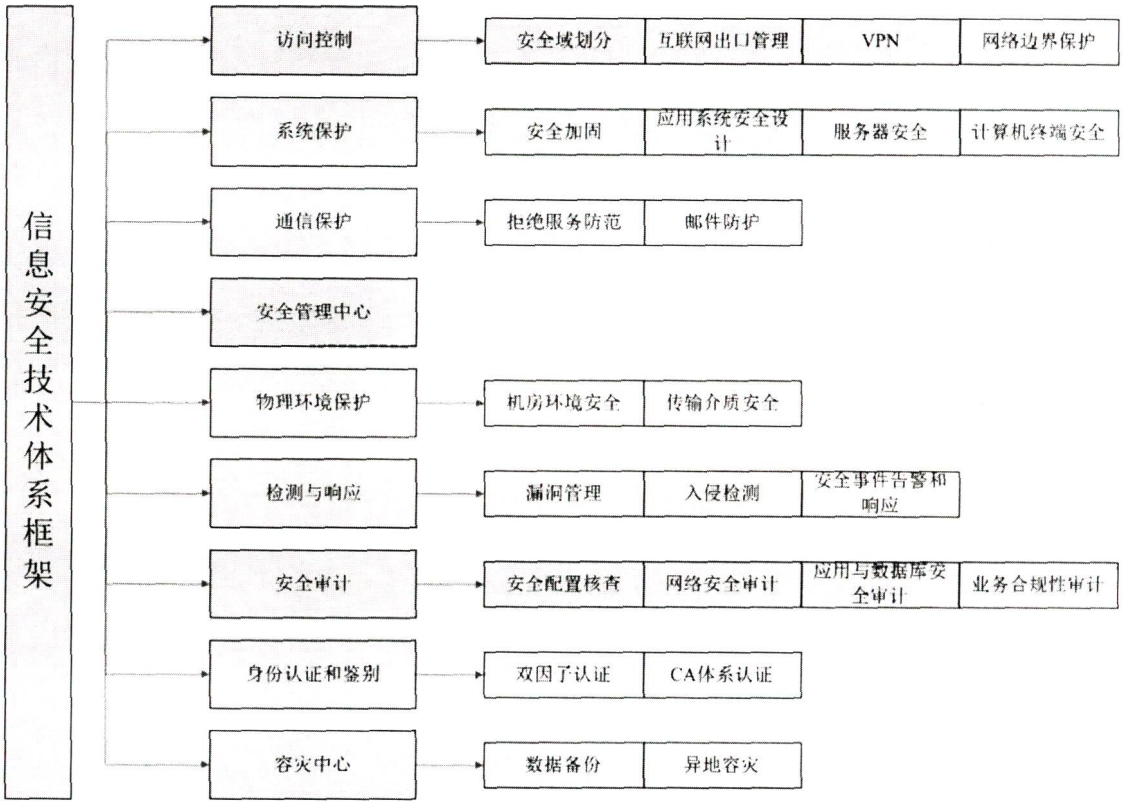
制度在发布、执行过程中，要定期进行评估，保留评估或评审记录。根据实际环境和情况的变化，对制度进行修改和完善，规范总体安全方针、安全管理制度、安全操作规程、安全运维记录和表单四层体系文件的一致性，必要时考虑管理制度的重新制定，并保留版本修订记录。

1.8.2.应用安全保障体系设计

应用安全技术体系的作用是通过使用安全产品和技术，支撑和实现安全策略，达到信息系统的保密、完整、可用等安全目标。

安全技术体系方面主要从技术角度提出了对信息系统的安全防护、检测、响应和恢复四种技术能力的要求。安全防护是根据系统存在的各种安全漏洞和安全威胁所采用的相应的技术防护措施，是安全保障体系的重心所在；入侵检测是随时监测系统的运行情况，及时发现和制止对系统进行的各种攻击；响应恢复是在安全防护机制失效的情况下，进行应急处理和响应，及时地恢复信息，减少被攻击的破坏程度。这些不同层次的安全技术根据安全策略和不同的需要，部署到合适的网络环境，为信息系统提供有效可靠的安全服务。

应用安全技术体系设计框架如下图所示：



1.8.3. 网站安全评估服务

针对以下评估流程中的各步骤进行准备。

1～3 步骤为评估准备阶段：

1、评估准备

明确责任与义务等，组建评估工作组，从人员方面做好准备。

2、评估计划

根据被评估单位信息系统的范围、规模，编制评估计划，细化评估工作步骤，明确各工作步骤中评估机构和被评估单位的工作职责和相互配合事项。

3、资料收集

在收集资料的同时，尽可能收集以下资料，为评估工作奠定基础。

技术资料

- 每个系统业务的设计文档；
- 系统各种业务应用网络拓扑（电子版）；
- 每个相关系统的业务流程，例如系统故障处理流程，业务服务提供方式等；

- 安全产品配备情况，当前已使用的安全产品及使用情况，配置情况；
- 特别的安全策略、应用系统特殊运行的环境、必须开放的端口、服务等；
- 其它有必要提醒和说明的技术资料。
- 管理资料
- 现有的信息系统安全体系框架；
- 信息系统安全管理制度；
- 遵循的行业法规或规范；
- 系统维护手册或制度等；
- 人员、密码、资料等管理方法；
- 安全管理策略文档。
- 日常维护资料
- 网络负载、利用率、网络质量等网络维护统计资料；
- 安全事件发生成功或未成功统计数据文档；
- 安全信息资料，包括签名备份、日志安全审计、安全产品日志备份等。

4～11 步骤为安全评估阶段，这些评估方法是评估主要方式——访谈、检查和测试的具体表现形式：

4、人工访谈

通过和管理层、决策层、执行层中的管理人员、安全员、系统维护员等各种角色的人员进行访谈，了解信息系统的大概状况。

5、业务数据流分析

结合资料阅读和人工访谈，了解信息系统中数据流转的节点和过程。

6、网络架构分析

网络架构分析是对信息系统的网络拓扑及网络层面细节架构的评估，主要从以下几个方面进行分析：网络安全规划、设备命名规范性、网络架构安全性、网络设备和链路冗余、设备选型及可扩展性、网络设备的 ACL、防火墙、隔离网闸、物理隔离、网络协议分析、网络流量分析、通信监控、通信加密、VPN 分析、网管系统、设备身份验证等。

7、漏洞扫描

漏洞扫描主要是通过扫描工具对信息系统的主机和网络进行安全扫描，来查

找网络设备、服务器主机、数据和用户账号/口令等安全对象目标存在的安全风险、漏洞和威胁。

8、人工配置核查

信息系统的网络设备安全策略的弱点和部分主机的安全配置错误等并不能被扫描工具全面发现，人工配置核查将对网络设备和主机以下几个方面进行核查：

- 是否最优的网段划分，保证了每个用户的最小权限原则；
- 路由器、交换机等网络设备的配置是否最优，是否配置了安全参数。

9、渗透测试

通过真实模拟黑客使用的工具、分析方法来进行实际的漏洞发现和利用的安全测试方法。这种测试方法可以非常有效的发现最严重的安全漏洞，尤其是与全面的代码审计相比，其使用的时间更短，也更有效率。

10、管理体系调研

就是从信息系统整体安全的角度对现有的管理体系进行全局性的调研和评估，它也包含了技术和管理方面的内容，具体包括：

- 所有安全控制、管理和使用措施是否正确和有效；
- 所有安全控制、管理和使用措施是否符合相关要求。

11、物理环境验证

主要通过现场查看的方式，对物理安全和环境进行评估。

12～14 步骤为安全分析阶段：

12、安全分析

根据评估结果的符合性判定情况，对现场评估的单个评估对象的单个评估项的评估结果是否符合要求进行判断，即单项判定，形成单项判定结论，判定结论分为三种情况：符合、不符合、不适用。

层面汇总分析主要是按照各个层面评估结果情况，分别统计物理安全、网络安全、主机系统安全、应用安全等各层面的不同安全控制的不同评估对象的单项评估结论。再根据单项判定结论，进行系统整体评估分析，分析单项判定结论为不符合的评估项是否影响系统的整体安全保护能力，分析系统的整体结构是否合理。

评估人员在评估结果分析的基础上，可以通过层面汇总分析和综合分析形成评估结论。

13、评估报告

通过对评估结果分析和形成的评估结论，编制评估报告。

评估报告编制完成后，评估单位根据评估协议书，提交的相关文档、评估原始记录和其他辅助信息。组织评估单位和被评估单位对评估报告进行评审。

14、加固改进建议

根据现状与要求之间的差距，分析系统存在的问题，并提出改进建议。

15、评估收尾

此阶段的主要工作为项目结束前的相关工作，包括报告递交、结过交流、经验传递、评估验收等。

1.8.4.渗透测试服务

1.8.4.1.服务概述

渗透性测试是对安全情况最客观、最直接的评估方式，主要是模拟黑客的攻击方法对系统和网络进行非破坏性质的攻击性测试，目的是侵入系统，获取系统控制权并将入侵的过程和细节产生报告给用户，由此证实用户系统所存在的安全威胁和风险，并能及时提醒安全管理员完善安全策略。

渗透性测试是工具扫描和人工评估的重要补充。工具扫描具有很好的效率和速度，但是存在一定的误报率，不能发现高层次、复杂的安全问题；渗透测试需要投入的人力资源较大、对测试者的专业技能要求很高（渗透测试报告的价值直接依赖于测试者的专业技能），但是非常准确，可以发现逻辑性更强、更深层次的弱点。

渗透测试服务通过利用目标应用系统的安全弱点模拟真正的黑客入侵攻击方法，以人工渗透为主，以漏洞扫描工具为辅，在保证整个渗透测试过程都在可以控制和调整的范围之内尽可能的获取目标信息系统的管理权限以及敏感信息。

渗透测试服务的主要流程如下：

1、信息收集

信息收集是指渗透实施前尽可能多地获取目标信息系统相关信息，例如网站注册信息、共享资源、系统版本信息、已知漏洞及弱口令等等。通过对以上信息的收集，发现可利用的安全漏洞，为进一步对目标信息系统进行渗透入侵提供基础。

2、弱点分析

对收集到的目标信息系统可能存在的可利用安全漏洞或弱点进行分析，并确定渗透方式和步骤实施渗透测试。

3、获取权限

对目标信息系统渗透成功，获取目标信息系统普通权限。

4、权限提升

当获取目标信息系统普通管理权限后，利用已知提权类漏洞或特殊渗透方式进行本地提权，获取目标系统远程控制权限。

1.8.4.2.测试内容

测试大类	测试项	测试目的
身份验证类	用户注册	检查用户注册功能可能涉及的安全问题
	用户登录	检查用户登录功能可能涉及的安全问题
	修改密码	检查用户修改密码功能可能涉及的安全问题
	密码重置	检查忘记密码、找回密码、密码重置功能可能涉及的安全问题
	验证码绕过	检测验证码机制是否合理，是否可以被绕过
	用户锁定功能	测试用户锁定功能相关的安全问题
会话管理类	Cookie 重放攻击	检测目标系统是否仅依靠 Cookie 来确认会话身份，从而易受到 Cookie 回放攻击
	会话令牌分析	Cookie 具有明显含义，或可被预测、可逆向，可被攻击者分析出 Cookie 结构
	会话令牌泄露	测试会话令牌是否存在泄露的可能
	会话固定攻击	测试目标系统是否存在固定会话的缺陷
	跨站请求伪造	检测目标系统是否存在 CSRF 漏洞
访问控制类	功能滥用	测试目标系统是否由于设计不当，导致合法功能非法利用
	垂直权限提升	测试可能出现垂直权限提升的情况
	水平权限提升	测试可能出现水平权限提升的情况
输入处理类	SQL 注入	检测目标系统是否存在 SQL 注入漏洞
	文件上传	检测目标系统的文件上传功能是否存在缺陷 ，导致可以上传非预期类型和内容的文件

	任意文件下载	检测目标系统加载/下载文件功能是否可以造成任意文件下载问题
	XML 注入	测试目标系统-是否存在 XML 注入漏洞
	目录穿越	测试目标系统是否存在目录穿越漏洞
	SSRF	检测目标系统是否存在服务端跨站请求伪造漏洞
	本地文件包含	测试目标站点是否存在 LFI 漏洞
	远程文件包含	测试目标站点是否存在 RFI 漏洞
	远程命令/代码执行	测试目标系统是否存在命令/代码注入漏洞
	反射型跨站脚本	检测目标系统是否存在反射型跨站脚本漏洞
	存储型跨站脚本	检测目标系统是否存在存储型跨站脚本漏洞
	DOM-based 跨站脚本	检测目标系统是否存在 DOM-based 跨站脚本漏洞
	服务端 URL 重定向	检查目标系统是否存在服务端 URL 重定向漏洞
信息泄露类	Error Code	测试目标系统的错误处理能力，是否会输出详尽的错误信息
	Stack Traces	测试目标系统是否开启了 Stack Traces 调试信息
	敏感信息	尽量收集目标系统的敏感信息
第三方应用类	中间件	测试目标系统是否存在 Jboss、WebLogic、Tomcat 等中间件
	CMS	测试目标系统是否存在 Dedecms、Phpcms 等 CMS

（一）网络层安全

针对该系统所在网络层进行网络拓扑的探测、路由测试、防火墙规则试探、规避测试、入侵检测规则试探、规避测试、无线网安全、不同网段 Vlan 之间的渗透、端口扫描等存在漏洞的发现和通过漏洞利用来验证此种威胁可能带来的损失或后果，并提供避免或防范此类威胁、风险或漏洞的具体改进或加固措施。

由于服务器系统和网络设备研发生产过程中所固有的安全隐患及系统管理员或网络管理员的疏忽，一般网络层安全漏洞包括以下安全威胁：

(1) 明文保存密码

由于管理员的疏忽，设备配置密码以明文的方式保存，这带来了一定的安全威胁。

(2) 未配置登录超时

对系统没有甚至登录超时的时间，这当登录系统没有及时退出的时候，可能导致被其他人利用。

(3) 未配置 AAA 认证

系统没有配置统一的 AAA 认证，这不便于权限的管理。

(4) 未配置管理 ACL

交换机没有配置管理 IP 的 ACL，可导致任意地址访问设备，应该增加 ACL 进行限制。

(5) 其他配置问题

服务器系统、数据库系统及网络设备在使用过程中由于管理人员或开发人员的疏忽，可能未对这些默认配置进行必要的安全配置和修改，这就很容易引起越权操作，从而导致信息泄漏或篡改。

（二）系统层安全

通过采用适当的测试手段，发现测试目标在系统识别、服务识别、身份认证、数据库接口模块、系统漏洞检测以及验证等方面存在的安全隐患，并给出该种隐患可能带来的损失或后果，并提供避免或防范此类威胁、风险或漏洞的具体改进或加固措施。

(1) 版本过低

系统版本过低，没有及时更新或升级，导致系统存在众多未修复的安全漏洞（如 Apache 版本过底，可能存在大量溢出漏洞）。

(2) 远程溢出漏洞

溢出漏洞的产生是由于程序中的某个或某些输入函数（使用者输入参数）对所接收数据的边界验证不严密而造成。根据程序执行中堆栈调用原理，程序对超出边界的部分如果没有经过验证自动去掉，那么超出边界的部分就会覆盖后面的存放程序指针的数据，当执行完上面的代码，程序会自动调用指针所指向地址的命令。根据这个原理，恶意使用者就可以构造出溢出程序。

(3) 本地提权漏洞

本地提权漏洞是指低权限、受限制的用户，可以提升到系统最高权限或比较大的权限，从而取得对网站服务器的控制权。

(4) 弱口令

弱口令通常有以下几种情况：用户名和密码是系统默认、空口令、口令长度过短、口令选择与本身特征相关等。系统、应用程序、数据库存在弱口令可以导致入侵者直接得到系统权限、修改盗取数据库中敏感数据、任意篡改页面等。

(5) 权限过大

权限过大是指某用户操作权限超出他本身安全操作权限范围之外，这存在一定的安全风险。

（6） 高危服务/端口开放

系统很多高危服务和端口会被默认开放，例如，80，443，843，8001 - 8010，其中 8001~8010 同时支持 TCP 和 UDP 协议，SSH、Telnet、X-windows、Rlogin、ms-rpc、SNMP、FTP、TFTP 等服务，这些服务和端口的开放可能会带来安全问题。

(7) 允许匿名 IPC$连接

允许匿名 IPC$连接，是一个远程登录功能，同时所有的逻辑盘(c$,d$,e$……)和系统目录 winnt 或 windows(admin$)资源可共享，存在一定的安全风险。

(8) 其他配置问题

系统可能存在未对某些高危默认配置进行必要的安全配置和修改，导致被恶意攻击者利用，从而导致信息泄漏或篡改等严重后果。

（三）应用层安全

通过采用适当测试手段，发现测试目标在信息系统认证及授权、代码审查、被信任系统的测试、文件接口模块报警响应等方面存在的安全漏洞,并现场演示再现利用该漏洞可能造成的客户资金损失，并提供避免或防范此类威胁、风险或漏洞的具体改进或加固措施。

应用程序及代码在开发过程中，由于开发者缺乏安全意识，疏忽大意极为容易导致应用系统存在可利用的安全漏洞。一般包括 SQL 注入漏洞、跨站脚本漏洞、上传漏洞、CSRF 跨站请求伪造漏洞等。

(1) SQL 注入

SQL 注入漏洞的产生原因是网站程序在编写时，没有对用户输入数据的合法性进行判断，导致应用程序存在安全隐患。SQL 注入漏洞攻击就是利用现有应用程序没有对用户输入数据的合法性进行判断，将恶意的 SQL 命令注入到后台数据库引擎执行的黑客攻击手段。

(2) 跨站脚本

跨站脚本攻击简称为 XSS 又叫 CSS (Cross Site Script Execution)，是指服务

器端的CGI程序没有对用户提交的变量中的HTML代码进行有效的过滤或转换，允许攻击者往 WEB 页面里插入对终端用户造成影响或损失的 HTML 代码。

(3) 表单绕过

表单绕过是指在登录表单时可以利用一些特殊字符绕过对合法用户的认证体系，这造成对用户输入的字符没有进行安全性检测，攻击者可利用该漏洞进行 SQL 注入攻击。

(4) 上传漏洞

上传漏洞是指网站开发者在开发时在上传页面中针对文件格式（如 asp、php 等）和文件路径过滤不严格，导致攻击者可以在网站上上传木马，非法获取 webshell 权限。

(5) 文件包含

目标网站允许用户调用网站程序函数进行文件包含，同时未对所包含文件的类型及内容进行严格过滤。

6) 已知木马

目标网站被攻击者植入恶意木马，已知木马包括攻击者在进行网站入侵时留下的后门程序和网页挂马两种：后门程序严重危害网站安全，攻击者可利用该后门直接获取整个网站的控制权限，可对网站进行任意操作，甚至以网站为跳板，获取整个内网服务器的控制权限；网页挂马严重危害网站用户安全，用户对已被挂马的网页进行浏览和访问，其 PC 机自动下载并执行木马程序，导致用户 PC 机被攻击，同时严重危害网站的信誉和形象。

(7) 敏感信息泄露

敏感信息泄漏漏洞指泄漏有关 WEB 应用系统的信息，例如，用户名、物理路径、目录列表和软件版本。尽管泄漏的这些信息可能不重要，然而当这些信息联系到其他漏洞或错误设置时，可能产生严重的后果。例如：某源代码泄漏了 SQL 服务器系统管理员账号和密码，且 SQL 服务器端口能被攻击者访问，则密码可被攻击者用来登录 SQL 服务器，从而访问数据或运行系统命令。

以下几个是比较典型的敏感信息泄露漏洞：

- 源码信息泄露；
- 备份信息泄露；

- 错误信息泄露；
- 测试账户泄露；
- 测试文件泄露；
- 绝对路径泄露；
- ……

(8) 恶意代码

恶意代码泛指没有作用却带来危险的代码，其普遍的特征是具有恶意的目的；本身是一个独立的程序，通过执行发生作用。由于应用系统存在可被利用的安全漏洞，可能已被恶意人员植入恶意代码以获取相应权限或用以传播病毒。

(9) 解析漏洞

解析漏洞是指没有对解析的内容进行严格的定义，被攻击者利用，可能会使系统对带有木马的文件进行了解析并执行，导致敏感信息被窃取、篡改，甚至是系统奔溃。

(10) 远程代码执行漏洞

远程执行任意代码漏洞是指由于配置失误(有时我们在用户认证只显示给用户认证过的页面和菜单选项，而实际上这些仅仅是表示层的访问控制而不能真正生效)，攻击者能够很容易的就伪造请求直接访问未被授权的页面。

(11) 任意文件读取

系统开发过程中没有重视安全问题或使用不安全的第三方组件等，导致任意文件可读取，可导致入侵者获得数据库权限，并利用数据库提权进一步获得系统权限。

(12) 目录遍历

目录遍历是指由于程序中没有过滤用户输入的../和./之类的目录跳转符,导致攻击者通过提交目录跳转来遍历服务器上的任意文件。

(13) 目录列出

目录列出是指攻击者通过对访问的 URL 分析，得到一个敏感的一级或二级目录名称，然后访问该目录，返回结果会将会显示指定目录及其子目录下的所有文件，从而可以寻找并获取敏感信息（如备份文件存放地址、数据库连接文件源码、系统敏感文件内容等），甚至可以通过在地址栏中修改 URL 来挖掘这个目

录结构。

(14) 跨站请求伪造

跨站请求伪造（Cross-site request forgery，缩写为 CSRF），也被称成为“one click attack”或者 session riding，通常缩写为 CSRF 或者 XSRF，是一种对网站的恶意利用。尽管听起来像跨站脚本（XSS），但它与 XSS 非常不同，并且攻击方式几乎相左。XSS 利用站点内的信任用户，而 CSRF 则通过伪装来自受信任用户的请求来利用受信任的网站。与 XSS 攻击相比，CSRF 攻击往往不大流行（因此对其进行防范的资源也相当稀少）和难以防范，所以被认为比 XSS 更具危险性。

(15) 弱口令

弱口令通常有以下几种情况：用户名和密码是系统默认、空口令、口令长度过短、口令选择与本身特征相关等。系统、应用程序、数据库存在弱口令可以导致入侵者直接得到系统权限、修改盗取数据库中敏感数据、任意篡改页面等。

(16) 不安全对象引用

不安全的对象引用是指程序在调用对象的时候未对该对象的有效性、安全性进行必要的校验，如：某些下载程序会以文件名作为下载程序的参数传递，而在传递后程序未对该参数的有效性和安全性进行检验，而直接按传递的文件名来下载文件，这就可能造成恶意用户通过构造敏感文件名而达成下载服务端敏感文件的目的。

(17) 安全配置错误

某些 HTTP 应用程序，或第三方插件，在使用过程中由于管理人员或开发人员的疏忽，可能未对这些程序或插件进行必要的安全配置和修改，这就很容易导致敏感信息的泄露。而对于某些第三方插件来说，如果存在安全隐患，更有可能对服务器获得部分控制权限。

(18) 链接地址重定向

重定向就是通过各种的方法将各种网络请求重新定个方向转到其它位置（如：网页重定向、域名的重定向、路由选择的变化也是对数据报文经由路径的一种重定向）。

而某些程序在重定向的跳转过程中，对重定向的地址未进行必要的有效性和

安全性检查，且该重定向地址又很容易被恶意用户控制和修改，这就可能导致在重定向发生时，用户会被定向至恶意用户事先构造好的页面或其他 URL，而导致用户信息受损。

(19) 跳转漏洞

跳转漏洞是指网站用户访问时对其输入的参数没有进行验证，浏览器直接返回跳转到指定的 URL，跳转漏洞可引发 XSS 漏洞，攻击者可利用这个漏洞进行恶意欺骗。

(20) 后台管理

后台管理是指由于网站设计者的疏忽、后台管理员的配置不当或失误，导致攻击者可通过某些非法手段直接访问后台数据库页面，从而获取重要敏感信息，上传木马，甚至可获取后台管理员的权限，可进行删除、添加等非法操作，篡改后台数据库数据。

(21) 会话管理

会话管理主要是针对需授权的登录过程的一种管理方式，以用户密码验证为常见方式，通过对敏感用户登录区域的验证，可有效校验系统授权的安全性，测试包含以下部分：

- 用户口令易猜解

通过对表单认证、HTTP 认证等方式的简单口令尝试，以验证存在用户身份校验的登录入口是否存在易猜解的用户名和密码。

- 是否存在验证码防护

验证码是有效防止暴力破解的一种安全机制，通过对各登录入口的检查，以确认是否存在该保护机制。

- 是否存在易暴露的管理登录地址

某些管理地址虽无外部链接可介入，但由于采用了容易猜解的地址（如：admin）而导致登录入口暴露，从而给外部恶意用户提供了可乘之机。

- 是否提供了不恰当的验证错误信息

某些验证程序返回错误信息过于友好，如：当用户名与密码均错误的时候，验证程序返回“用户名不存在”等类似的信息，通过对这一信息的判断，并结合 HTTP Fuzzing 工具便可轻易枚举系统中存在的用户名，从而为破解提供了机会。