网站源代码安全测试规范

目的

制定本测试规范作为应用程序、网站源代码安全测试和评估的依据。

通过本类型的测评发现应用程序、源代码中包括OWASP十大Web漏洞在内的安全漏洞，识别、定位存在的安全漏洞，并分析漏洞风险，提出整改建议，提高系统的安全性。

测试依据

GB/T-17544信息技术 软件包 质量要求和测试

OWASP十大Web漏洞（Open Web Application Security Project，开放式Web应用程序安全项目，缩写为OWASP）

测试范围

源代码安全测试的范围可以是以C、C++、JAVA等开发语言编写的应用程序或网站的全部源代码，也可以是某个独立的业务模块或关键的业务流程模块的源代码。

测试方法

采用源代码分析工具对系统源代码的安全性进行测试，识别、定位代码存在的安全漏洞，并分析漏洞风险。

源代码选择

选择全部源代码进行测试时，需首先经代码编译，生成应用程序或网站，由委托方确认应用程序功能或网站内容无误。  代码量较大时，一般选择部分源代码进行测试。源代码的选择由委托测试方和测试方共同协商确定，对选定代码的测试结果仅对被测代码有效，不能作为评价全部源代码的依据。

测试工具

支持C/C++、JAVA、.NET等开发语言，内置安全代码规范，能够对代码自动地进行数据流、语义、结构、控制流、配置五个方面的分析的测试工具，包括：  Fortify公司的Source Code Analysis  Security Innovation公司的CxSuite

工具安装与配置

按照所选用的测试工具手册及技术文档的要求选择工具安装所依赖的硬件和软件，安装测试工具，启动测试工具自检，确认测试工具安装成功，运行正常。

选择安全代码规范，若委托方有明确的测试需求，按照测试需求配置工具；若委托方无明确的测试需求，选择测试工具的默认配置，至少包括跨站脚本、SQL注入、系统信息泄露等。

代码的导入与扫描

将系统源代码按功能模块分类，以文件夹形式存放。采用源代码分析工具的静态扫描分析模式选取源代码目录，设置扫描深度、扫描类型等参数，进行自动扫描。

当代码中存在相互引用关系过于复杂、重复代码过多、程序质量低等问题时，工具扫描可能会出现中断，此时需与开发源代码的系统分析人员一起分析代码，调整扫描参数，重新启动扫描或继续扫描。

扫描结束后，工具自动生成扫描报告。

测试内容

跨站脚本；

SQL注入；

系统信息泄露；

程序中存在密码硬编码；

拒绝服务攻击；

系统资源无法释放；

信任边界模糊；

工具支持的其他测试内容。

测试分析

工具的扫描结果生成的文档作为测试记录的参考。

人工分析扫描报告，提取被扫描源代码的基本信息，对发现的代码安全性问题按严重程度进行分类，并统计漏洞数目。

着重分析扫描报告中的高风险和中风险问题。