**国家标准《信息安全技术 代码安全审计规范》**

**（征求意见稿）编制说明**

1. 工作简况
   1. 任务来源

《代码安全审计规范》 是国家标准化管理委员会2015年下达的信息安全国家标准制定项目。由信息安全共性技术国家工程研究中心主要负责进行规范的起草，中国科学院信息工程研究所、国家保密科技测评中心、北京信息安全测评中心、中国信息安全测评中心、中国电子技术标准化研究院、公安部第三研究所等单位参与起草。

* 1. 主要工作过程

1、2015年7月，组织参与本规范编写的相关单位召开项目启动会，成立规范编制小组，确立各自分工，进行初步设计，并听取各协作单位的相关意见。

2、2015年8-9月，根据任务书的要求，规范编制小组开展考察调研和资料搜集工作，研究国内外代码安全相关的材料，研究资料包括国内外安全编码标准，代码审计相关标准，行业标准，各大高校研究成果，漏洞库以及主流产品的规则库等各方面。对《C安全编码标准》，《Java安全编码标准》，ISO/IEC的标准《Information technology -- Programming languages, their environments and system software interfaces -- C secure coding rules》《航天型号软件C语言安全子集》，CWE（Common Weakness Enumeration通用软件缺陷列表），《OWASP安全编码规范快速参考指南》，，《NASA-GB-A301 Software Quality Assurance Audits Guidebook》，《NASA-GB-8719.13 NASA Software Safety Guidebook》，北京邮电大学，西安电子科技大学等科研成果, 以及Findbug, Fortify等产品规则库进行了重点分析，进行标准漏洞收集整理，形成编制思路。

3、2015年9月底，组织编制组专家进行编制思路讨论，形成意见汇总处理表。本次会议上主要对标准的定位，应该覆盖的内容，编写角度等进行了讨论。

4、2015年10月，整理出源代码安全审计规范的框架体系

5、2015年11月- 2016年1月，按照制定的框架结构，进行分类方法研究，研究了7PK，CWE Development view，CWE Research view等分类方法，最后确定审计规则分类的原则和方法主要参照CWE Development view。

6、2016年2-9月，形成《代码安全审计规范》编制组内草案初稿V1.0。V1.0版本分C、Java两个部分，其中融合了CWE、CERT等各方关于C语言和Java源代码的规则，剔除了其中通用性较差或不易操作的规则。

7、2016年10月，组织编制组内专家进行评审。

8、 2016年 11月-2017年5月，根据专家意见进行修改，增强审计的描述，增加概述等章节，形成草案初稿V2.0。

9．2017年6月， 邀请安全标准专家进行评审。

10．2017年7月， 根据专家意见，由于只有一个国标号，因此将标准由C、Java两部分抽象提炼成一个标准，补充完善了审计指标和审计方法的内容；将代码示例移到附录中作为补充性资料，形成了草案初稿V3.0。

11．2017年8月， 根据专家意见，增加了源代码安全审计目的、审计时机、审计人员、审计方法以及审计过程等章节；将原章节的源代码安全审计要求改为源代码安全弱点审计列表，并对弱点的二级分类取消，均衡各审计条款粒度，形成草案V4.0.

12.2017年10月，在厦门安标委第二次会议周上，经过征求意见、会议讨论、最终经专家组决定将该标准由草案推进为征求意见稿。

1. 编制原则和主要内容

2.1 编制原则

本标准的研究与编制工作遵循以下原则：

（1）通用性原则

对国内外知名源代码安全编码标准进行总结、归纳，同时参考吸纳国内外相关领域的先进成果并融入标准，审计规则范围覆盖了编码阶段的常见缺陷。

（2）可操作性和实用性原则

对于比较抽象的审计规则，给出了规范代码示例和不规范代码示例等补充性资料。

（3）简化原则

对大量规则进行筛选提炼，经过剔除、替换，保持整体结构合理且维持原意和功能不变。

（4）完备性原则

融合了国内外代码审计相关标准规范，以及主流的代码审计工具审计规则，保证了审计指标的完备性。

2.2 主要内容

《代码安全审计规范》主要依据项目要求，制定了当前较为常用的编程语言源代码层面的安全审计规范。

|  |
| --- |
| 本规范的审计对象是源代码，描述了审计目的、审计时机、审计人员、审计方法以及审计过程的规范，并描述了源代码安全弱点审计列表供审计时参考。  文本主体由6个章节正文组成。  第1章、第2章和第3章为标准的固定格式要求，说明《信息安全技术 代码安全审计规范》标准的使用范围、引用的其他标准、使用到的术语定义。  第4章 《源代码安全审计概述》 描述了审计目的、审计时机、审计人员、审计方法。  第5章是对源代码安全审计过程的具体描述。  第6章描述了源代码审计通用弱点检查列表，供审计过程中参考使用。根据常见的软件缺陷分类进行了各小节的划分并列出了相关的具体审计指标。  附录A描述了源代码审计报告的参考内容。  附录B中对于部分比较抽象的规则给出了代码层面不规范用法示例和规范用法示例。  最后给出了参考文献。  考虑到计算语言有很多种，我们以典型的结构化语言（C）和面向对象语言（Java）为规范示例目标，讨论源代码安全缺陷的常见问题。部分规则给出了详细的代码示例。 |

1. 主要试验（或验证）的分析、综述报告，技术经济论证，预期的经济效果

本标准中审计条款所附的所有的C语言和Java语言代码示例都在相应环境下进行了验证。

本标准将为代码安全审计的服务开展提供技术规范，将使国内软件代码安全审计拥有可靠的标准和依据。同时也将为相关工具的开发提供规则需求支持，对于引导国产自主可控代码安全检测工具的开发以及网络安全审查工作的开展具有重要意义。

1. 采用国际标准和国外先进标准的程度，以及与国际、国外同类标准水平的对比情况，或与测试的国外样品、样机的有关数据对比情况

无

1. 与有关的现行法律、法规和强制性国家标准的关系

当前，国内一些标准中明确提到需要实施代码审计：

**（1）《信息安全等级保护基本要求》标准**

一级要求“应在软件安装之前检测软件包中可能存在的恶意代码”（5.2.4.5）；

二级要求“应要求开发单位提供软件源代码，并审查软件中可能存在的后门”（6.2.4.5）；

三级要求 “应委托公正的第三方测试单位对系统进行安全性测试，并出具安全性测试报告”（7.2.4.7）；

四级要求“应要求开发单位提供软件源代码，并审查软件中可能存在的后门和隐蔽信道”（8.2.4.5）。

**（2）GB/T 22080-2008 《信息技术 安全技术 信息安全管理体系要求》**

提出了“防范恶意代码和移动代码”，“应用中正确处理”，“技术脆弱性管理”的要求；

应用系统必须以相应的控制措施提供相应的功能。

**（3）IT审计工程**

为验证安全功能的实现，必然需要相应的测试结论提供相应的支持：

“防范恶意代码和移动代码”，“应用中正确处理”，“技术脆弱性管理”等要求均可以利用代码审查进行控制目标的验证。

**（4）支付卡行业数据安全标准PCI**

明确提出了由独立于开发团队的内部组织或第三方专业机构进行代码安全审查。

综合而言，上述标准均对代码审计提出了要求，但目前并没有具体规范供参考执行，本项目研究的代码审计标准将会对现有的标准形成较好的补充。

1. 重大分歧意见的处理经过和依据

详见标准意见汇总处理表。

1. 国家标准作为强制性国家标准或推荐性国家标准的建议

建议本标准作为推荐性国家标准发布实施。

1. 贯彻国家标准的要求和措施建议（包括组织措施、技术措施、过渡办法等内容）

无

1. 其他事项说明

（1）关于审计结果指标量化判定问题说明：

本标准审计目标在于发现源代码中存在的弱点而不是漏洞，由于弱点只有在特定应用环境（某些情况下多个弱点才构成一个漏洞）下才构成漏洞，而且具体危害针对漏洞应用环境不同而不同，因此量化审计指标不具有实际意义。鉴于这种情况， 项目组不给出实际审计结果量化指标。

（2）关于适用对象的问题说明：

根据项目合同要求，本标准审计的源代码对象以C语言和Java语言为主，对其他语言审计可以将此标准做为参考来执行。审计对象仅限于源代码，审计目的是源代码层面安全弱点问题，不包含软件开发生命周期中的其他阶段如需求分析、设计、测试、部署、运维管理等安全问题。

2017年10月，经专家讨论，本标准适用对象从仅针对于C语言和Java语言，调整为不针对特定语言的源代码安全审计规范，但参考实例仍以C语言和Java语言为主。

（3）关于正确性说明：

对标准所列的审计条款，尽可能给出了示例代码，并在Java或C相应环境下进行了验证，但不排除个别示例在特定的环境下存在错误的可能性。

标准编制组

2018年6月