|  |  |
| --- | --- |
| ICS | 35.240.60 |
| CCS | |  | | --- | |  |   L 67 |

团体标准

T/CASMES XXXX—2024

车联网安全软件开发技术规范

Technical specification for security software development in connected vehicles

XXXX - XX - XX发布

XXXX - XX - XX实施

中国中小企业协会  发布

目次

[前言 II](#_Toc202168205)

[1 范围 1](#_Toc202168206)

[2 规范性引用文件 1](#_Toc202168207)

[3 术语和定义 1](#_Toc202168208)

[4 缩略语 1](#_Toc202168209)

[5 开发环境 1](#_Toc202168210)

[6 开发过程 1](#_Toc202168211)

[6.1 概述 1](#_Toc202168212)

[6.2 软件开发启动 2](#_Toc202168213)

[6.3 需求分析与验证 2](#_Toc202168214)

[6.4 软件架构设计 2](#_Toc202168215)

[6.5 软件单元设计 3](#_Toc202168216)

[6.6 软件界面设计 5](#_Toc202168217)

[6.7 软件编程 5](#_Toc202168218)

[6.8 软件测试 5](#_Toc202168219)

[6.9 软件文档编写 5](#_Toc202168220)

[7 软件使用及维护 6](#_Toc202168221)

1. 前言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由浙江无界矩阵科技有限责任公司提出。

本文件由中国中小企业协会归口。

本文件起草单位：浙江无界矩阵科技有限责任公司、XXX、XXX。

本文件主要起草人：XXX、XXX。

车联网安全软件开发技术规范

* 1. 范围

本文件规定了车联网安全软件开发的开发环境、开发过程、软件使用及维护。

本文件适用于车联网安全软件开发实施。

* 1. 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 8566 系统与软件工程 软件生存周期过程

GB/T 8567 计算机软件文档编制规范

GB/T 11457 信息技术 软件工程术语

GB/T 15532 计算机软件测试规范

GB/T 20157 信息技术 软件维护

* 1. 术语和定义

GB/T 8566、GB/T 11457 界定的术语和定义适用于本文件。

* 1. 缩略语

下列缩略语适用于本文件。

CPU：中央处理器（Central Processing Unit）

IP：互联网协议（Internet Protoco）

* 1. 开发环境

车联网安全软件开发应在主流操作系统上开展，宜使用 Java、Python、Vue 等编程语言或 Linux 开源系统等编程环境。

* 1. 开发过程
     1. 概述

车联网安全软件的开发包括如下 7 个阶段：

1. 软件开发启动；
2. 需求分析与验证；
3. 软件架构设计；
4. 软件单元设计与实现；
5. 软件编程；
6. 软件测试；
7. 软件文档编写。
   * 1. 软件开发启动

应根据软件开发的程度和复杂性规划并启动车联网安全软件开发子阶段，并制定项目实施方案：

1. 确定开发目标和工作范围；
2. 定义软件生存周期计划；
3. 制定项目实施进度计划；
4. 开发队伍组建；
5. 确定软件开发过程中需要的工具。
   * 1. 需求分析与验证

应安排专人（一般是项目负责人）编写软件需求说明书，应包括该软件的功能性需求、非功能性需求和软件安全性需求。

软件需求说明书应当可验证，宜组织相关人员对其进行评审。

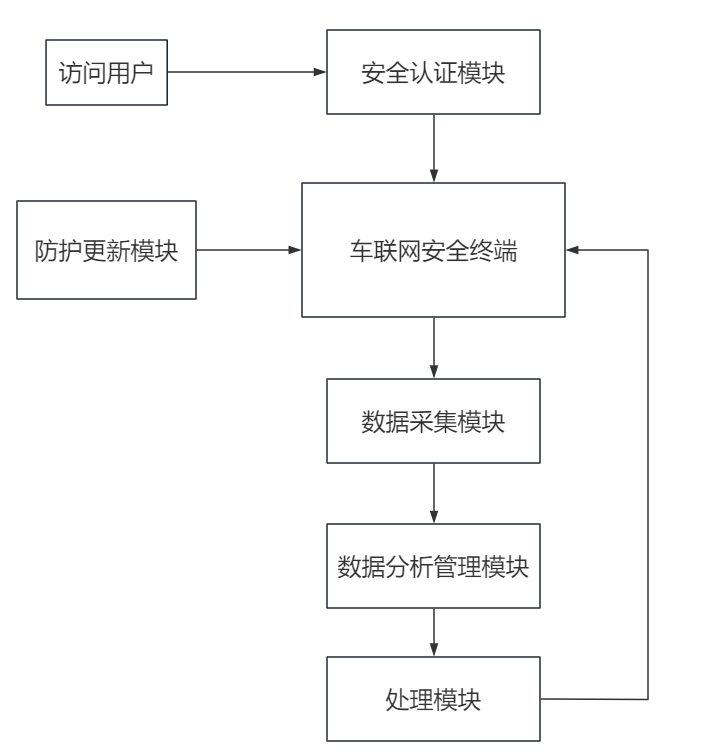
* + 1. 软件架构设计

车联网安全软件架构设计宜包括：

1. 总体设计，如需求规定、运行环境、基本设计概念等；
2. 接口设计；
3. 运行设计；
4. 系统数据结构设计。

车联网安全软件的结构如图 1 所示，宜由如下模块组成：

1. 安全认证模块：用于对试图进入车联网安全终端的访问用户进行安全认证，认证合格后才允许访问用户进入；
2. 数据采集模块：用于获取车联网终端所监测到的关联信息；
3. 数据分析管理模块：用于对获取的关联信息进行分析处理，并生成入侵风险值，从而根据入侵风险值情况来判断车联网安全终端是否存在被入侵风险，并生成对应的处理指令；
4. 处理模块：根据生成的处理指令进行相应的处理；
5. 防护更新模块：对车联网安全终端的防护模块进行优化更新。



1. 车联网安全软件结构示意图
   * 1. 软件单元设计
        1. 安全认证模块

该模块应能按如下步骤进行安全认证：

1. 获取访问用户的ID，若为权限ID，则进行进一步识别认证，若为非权限用户，则阻止其进一步登录；
2. 按如下方法获取登录的时间与常规登录时间的差值 ：
   1. 根据历史登录习惯,设定每日常规登录时间区间[，]，同时获取此次登录时间 ；
   2. 当 ，时，=0；

当 ，时，=||；

当 ，时，=||；

其中， 为0 点时刻， 为 24 点时刻。

1. 按如下方法获取登录IP地址与常规登录IP地址的距离差值：
   1. 根据历史登录习惯,设定常规登录IP地址距离范围区间 [0，]，同时获取此次登录IP地址 ；
   2. 当 0，时，=0，否则 =||；
2. 按公式（1）计算登录风险值 ：

()

式中：

—登录风险值；

—访问用户尝试登录的次数；

—自然常数；

、—比例系数；

—此次登录的时间与常规登录时间的差值；

—登录的时间与常规登录时间的标准差值；

—此次登录IP地址与常规登录IP地址的距离差值；

—登录IP地址与常规登录IP地址的标准距离差值；

—此次登录IP地址与上次登录IP地址的距离差值；

—此次登录时间与上次登录时间的差值。

1. 将计算出的登录风险值 与系统内预设的登录风险阈值 进行比对，当 ＞时，存在登录异常风险，应启用多因素认证，否则认为认证合格，许访问用户登录；
2. 多因素认证包括但不限于手机验证码认证和人脸识别认证，当通过其中的至少一种认证后，允许用户登录，否则阻止其进一步登录。
   * + 1. 数据采集模块

该模块应能采集包括但不限于如下信息：

1. 车联网安全终端监测到的：
   1. CPU占用率；
   2. 内存占用率；
   3. 网络流量变动情况；
2. 端口扫描检测时出现的日志异常信息。
   * + 1. 数据分析管理模块

该模块应能按公式（2）计算出入侵风险值 ：

()

式中：

—入侵风险值；

—该检测周期内端口扫描出现的日志异常次数；

—内存占用率变动值，按公式（3）进行计算；

—CPU占用率变动值，按公式（4）进行计算；

—网络流量变动值，按公式（5）进行计算；

—内存占用率随时间变化曲线；

—系统设置的标准内存占用率随时间变化曲线；

—CPU占用率随时间变化曲线；

—系统设置的标准CPU占用率随时间变化曲线；

—网络流量随时间变化曲线；

—系统设置的标准网络流量随时间变化曲线；

—检测周期开始时间；

—检测周期结束时间。

()

()

()

式中：

—检测周期内的最大内存占用率；

—检测周期内的最小内存占用率；

—检测周期内的最大CPU占用率；

—测周期内的最小CPU占用率；

—检测周期内的最高网络流量；

—检测周期内的最低网络流量。

* + - 1. 处理模块

该模块能根据生成的处理指令进行处理：

1. 将获得的入侵风险值 与系统内设置的第一入侵风险阈值 进行比对, 当＞时生成处理指令；
2. 处理指令包括一级处理指令以及二级处理指令；
3. 当生成处理指令时，将获得的入侵风险值 与系统内设置的第二入侵风险阈值 进行比对：
   1. 当 ＜时，生成一级处理指令，提醒终端管理人员对访问用户进行查看；
   2. 当 时，生成二级处理指令，强制弹出访问用户。
      * 1. 防护更新模块

该模块应能对软件的防护模块进行更新，包括：

1. 定时触发更新：根据设置的固定时间点进行更新；
2. 条件触发更新：
   1. 按公式（6）计算判断系数 ：

()

式中：

—判断系数；

—产生的漏洞种类；

—每种漏洞修复用时时间；

—每种漏洞修复所用的标准用时时间；

—防护模块内日志出现异常的次数；

—防护模块内产生的漏洞次数；

∈[1,m]。

* 1. 将判断系数 与系统预设的判断系数阈值 相比较，当H＞ H\_th时，自动对防护模块进行更新。
     1. 软件界面设计

应遵守如下原则：

1. 一致性：字体、颜色、图标风格等视觉效果一致，相同功能的操作逻辑一致；
2. 用户友好：围绕用户需求、习惯和能力设计界面，减少用户学习成本；
3. 科学性性：提供新手引导等，降低用户初次使用门槛；
4. 容错性：减少用户误操作概率，允许轻松撤销错误；
5. 易用性：具有错误、重要操作提示功能；
6. 简洁性：功能按优先级排序放置，减少冗余元素，复杂功能宜分步骤展示。
   * 1. 软件编程

应完成软件设计说明书的编写并通过评审后，根据软件需求说明书和软件设计说明书的内容进行编程。

* + 1. 软件测试

应按 GB/T 15532 的相关规定进行软件测试，包括但不限于：

1. 单元测试；
2. 集成测试；
3. 验收测试；
4. 回归测试。

测试前应进行测试需求分析，参考 GB/T 9386 制定测试计划。

测试后应整理和分析测试数据，评价测试效果和被测软件项，描述测试状态，完成软件测试报告,并通过测试评审。

* + 1. 软件文档编写

应根据 6.1 规定的开发过程相关活动，在相应阶段完成软件开发和用户文档编制，应包括：

1. 软件开发计划；
2. 软件需求规格说明；
3. 软件设计说明；
4. 软件用户手册；
5. 软件测试分析报告；
6. 软件产品规格说明。

软件文档的编制应符合 GB/T 8567 的相关规定，也可按照 GB/T 8567 规定的软件文档根据自身特点进行补充。

* 1. 软件使用及维护

软件应由相关部门进行发放、接收和使用，发放内容包括

应按 GB/T 20157 的相关规定进行软件的维护。

