基于RNN的恶意代码检测项目

需求规格说明书

第11组

2021年8月28日

版本：V1.3

变更记录

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **序号** | **版本** | **变更说明** | **修改人** | **审批人** | **日期** |
| 01 | V1.0 | 初稿 | 应淞斐 | 全体成员 | 2021.8.25 |
| 02 | V1.1 | 添加业务流程图 | 陈向龙 | 全体成员 | 2021.8.26 |
| 03 | V1.1 | 整理初步内容 | 周鹏 | 全体成员 | 2021.8.26 |
| 04 | V1.2 | 完善各模块详细内容 | 周鹏 | 全体成员 | 2021.8.28 |
| 05 | V1.3 | 添加描述性说明并排版 | 应淞斐 | 全体成员 | 2021.8.28 |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

目录

[1 引言 4](#_Toc23723)

[1.1 目的 4](#_Toc12109)

[1.2 适用范围 4](#_Toc4182)

[1.3 预期读者和阅读建议 4](#_Toc4124)

[1.4 术语和缩略语 5](#_Toc24343)

[1.5 需求描述约定 5](#_Toc8816)

[1.6 参考资料 6](#_Toc13817)

[2 项目概述 7](#_Toc22351)

[2.1 简介 7](#_Toc16748)

[2.1.1 项目背景 7](#_Toc13027)

[2.1.2项目目标 7](#_Toc16518)

[2.2 用户与角色 7](#_Toc23863)

[2.3 功能总体设计 8](#_Toc13126)

[·功能架构图 8](#_Toc30184)

[·功能列表 8](#_Toc17415)

[2.4 核心业务流程 9](#_Toc10827)

[·RNN识别处理模块 10](#_Toc3466)

[·Viper兼容模块 10](#_Toc18905)

[·RESTFUL API模块 11](#_Toc10809)

[2.5 核心用例图 11](#_Toc13023)

[3 功能性需求 12](#_Toc7824)

[3.1 RNN识别处理模块 12](#_Toc447)

[3.2 Viper兼容模块 12](#_Toc32727)

[3.3 RESTFUL API模块 12](#_Toc32595)

[4 非功能性需求 14](#_Toc392)

[4.1 质量需求 14](#_Toc28449)

[精度 14](#_Toc10011)

[可用性 14](#_Toc9354)

[可靠性和健壮性 15](#_Toc1302)

[可维护性 16](#_Toc15898)

[可扩展性 16](#_Toc4995)

[性能 16](#_Toc8349)

[易用性 16](#_Toc25920)

[安全性 16](#_Toc22446)

[4.2 约束 17](#_Toc29393)

[4.3 接口需求 17](#_Toc28625)

[用户接口 17](#_Toc23690)

[硬件接口 17](#_Toc15966)

[软件接口 17](#_Toc26453)

[通讯接口 18](#_Toc28103)

[4.4 技术需求 18](#_Toc20575)

[软硬件环境需求 18](#_Toc6394)

[4.5 文档需求 18](#_Toc14190)

[5 验收标准 20](#_Toc22259)

# 引言

## 目的

本需求说明旨在对基于RNN的恶意代码检测项目的功能架构及子系统的功能需求、非功能需求进行逐一分析；并对各系统接口、质量需求、文档需求和约束做出可行方案。

本需求规格说明书编写目的：

（1）在需求调研阶段，通过本文档，与系统用户进行系统需求的确认。

（2）在系统设计阶段，通过本文档，指导该系统的概要设计和数据库设计。

（3）在系统开发阶段，通过本文档，帮助相关人员全面了解用户需求与系统功能。

（4）系统测试和联调阶段，通过该文档，是编写测试用例的依据。

（5）在系统实施阶段，实施人员借助本文档完成系统的实施工作。

（6）在系统使用过程中，本文档作为用户使用的辅助说明文件。

（7）在系统验收阶段，本文档将作为主要验收依据。

## 适用范围

本文档适用于所有与本项目有关的软件开发阶段及其相关人员。

本项目开发目的主要为采用一种快速高效的方法准确检测恶意代码，产品适用范围包括检测或查杀恶意代码需求用户、网络安全人员、恶意代码检测研究人员等，主要定位于网络安全相关用户。

## 预期读者和阅读建议

|  |  |
| --- | --- |
| 预期读者 | 阅读建议 |
| 客户代表 | 重点阅读本文档需求部分 |
| 项目经理 | 重点阅读本文档各部分 |
| 系统分析人员 | 重点阅读本文档各部分 |
| 系统设计人员 | 重点阅读本文档各部分 |
| 技术开发人员 | 重点阅读本文档各部分 |
| 测试人员 | 重点阅读本文档各部分 |
| 其他人员 | 选择性阅读本文档 |

## 术语和缩略语

|  |  |  |
| --- | --- | --- |
| 缩略语/术语 | 全称 | 说明 |
| 恶意代码 | 恶意代码  (Malicious Code) | 恶意代码是指故意编制或设置的、对网络或系统会产生威胁或潜在威胁的计算机代码。 |
| RNN | 循环神经网络  (Recurrent Neural Network) | 一类以序列数据为输入，在序列的演进方向进行递归且所有节点（循环单元）按链式连接的递归神经网络。 |
| Opcode | 操作码  (Operation Code) | 描述机器语言指令中，指定要执行某种操作的机器码。 |
| Viper | Viper | 二进制分析和管理框架，用于存储二进制文件的主要接口和数据库，允许我们在导入时自动运行其他python模块，从而驱动自动化分析的核心。 |
| Restful API | representation state  transfer API | 即用URI表示资源，用HTTP方法(GET, POST, PUT, DELETE)表征对这些资源的操作。 |
| Python3 | Python3.0及以上版本 | 一种计算机编程语言。 |

## 需求描述约定

**1）需求标识方法**：

需求层次：分三个层次，第一层需求指主功能模块，第二层需求指功能模块的子功能，第三层次指子功能下的具体需求。

**2）需求跟踪的颗粒度**：跟踪到第二层功能需求。

**3）需求优先级定义**：

* 本文档统一规定对需求层次为二级以上的定义优先级，三层需求依据二层需求的优先级执行。
* 需求分析师应确定每个需求的优先级并写入软件需求分析说明书，需求的优先级的评价标准如下：

|  |  |  |
| --- | --- | --- |
| 级别定义 | 判断标准 | 采取的措施 |
| 高 | 满足以下任意一条时：   1. 需求实现的紧急程度为特急或紧急； 2. 国家或行业法律法规、标准要求的，客户明确要求的,满足正常业务必须的。 | 对于这些需求在项目实施过程中需重点投入资源，优先实现，只有在这些需求上达成一致意见，软件才会被接受；必须完美地实现。通常这类需求在当前版本必须实现。 |
| 中 | 满足以下任意一条时：   1. 客户隐含要求，对正常业务影响程度不大； 2. 需求实现的紧急程度为中； 3. 支持必要的系统操作，实现这些需求将增强产品的性能，是产品最终所要求的。 | 这些需求必须被实现，但如果项目实施中出现进度、资源等方面的冲突时，如果有必要，可以延迟到下一版本；需要付出努力，但不必做得太完美。 |
| 低 | 满足以下任意一条时：   1. 功能或质量上的附加功能； 2. 实现这些需求会使产品更完美，若不实现也不影响产品的功能与性能，属于锦上添花； 3. 求实现的紧急程度为低。 | 实现或不实现均可；  可以在项目组有较足够的时间时考虑这些需求的实现。 |

## 参考资料

1. GB/T 8567－88 软件需求规格说明书
2. GB/T 8567－2006计算机软件文档编制规范
3. Viper官方文档.https://viper-framework.readthedocs.io/en/latest/index.html
4. Restful API接口规范.https://www.jianshu.com/p/a6d9cf64e954
5. Pratikkumar Prajapati, Mark Stamp. Malware Analysis Using Artificial Intelligence and Deep Learning[J].Empirical Analysis of Image-Based Learning Techniques for Malware Classification.2021.3.24(411-435).

# 项目概述

## 简介

### 2.1.1 项目背景

随着信息技术的发展，计算机已经成为人们日常生活中重要的一部分，娱乐休闲、商务办公乃至国民生产，都与计算机技术息息相关。然而，伴随其应用的广泛性，计算机安全威胁也无处不在，对我们日常的计算机使用造成了极大的困扰和危害。虽然目前广泛使用的病毒查杀软件已经为我们提供了一定的安全保障，但是由于不可预知的新型恶意代码层出不穷且传播迅速，造成了传统的检测技术具有一定的滞后性，使得计算机安全问题存在着长期的隐患。

本项目基于机器学习方法对恶意代码检测技术进行研究，针对传统检测技术对未知类型恶意代码的检测能力的薄弱之处，主要采用RNN模型对恶意代码进行静态检测。本项目基于二进制文件进行特征提取用于恶意代码的检测和分类，并将该模块集成在开放恶意代码管理平台中，同时对外提供服务接口，使得系统能够面向更广泛的用户。本项目主要功能为对恶意代码的单文件及批量处理和判断，检测代码是否存在威胁性并给出其具体分类。

### 2.1.2项目目标

本项目核心功能为恶意代码的检测及分类。

用户上传代码文件或调用系统接口，即可使用平台上的检测功能，平台反馈检测的分类结果。功能需求方面，平台应提供对外API接口，支持批量处理和单文件处理两种处理方式。性能需求方面，要求系统检测分类结果的准确率在75%以上；单文件响应时间在1s内完成，多文件处理生成相应结果文件在用户本地，且要求能够容纳多用户同时响应。安全性需求方面，要求确保检测过程中恶意代码不会对主机造成侵入，同时具备良好的可维护性。

## 用户与角色

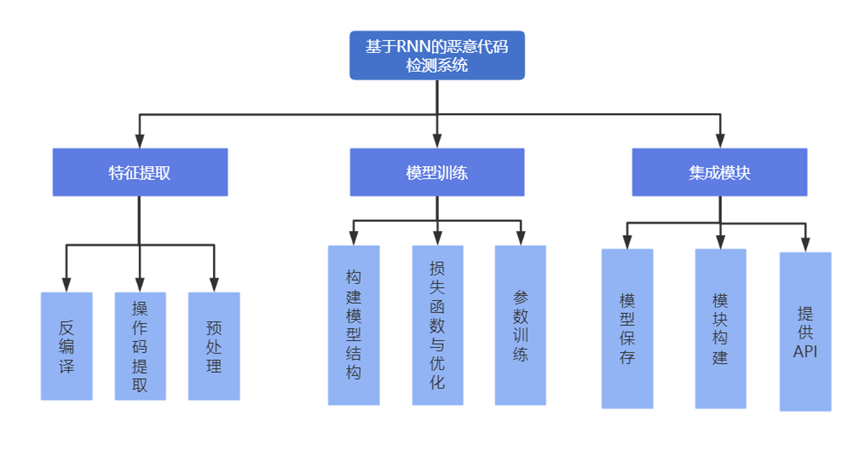
|  |  |
| --- | --- |
| **用户/角色** | **描述** |
| 普通用户 | 普通用户中对于网络安全有较强的防范意识，对于文件或代码有检测需求的用户，主要为项目的使用者。 |
| 网络安全人员 | 负责或研究网络安全的人员，可借助本平台进行恶意代码的检测和分析；或者从事相关专业领域，可作为开发者角色研究并使用本项目。 |

## 功能总体设计

### ·功能架构图

本项目的功能模块结构如图所示，主要分为特征提取、模型构建和集成模块三大部分。

其中特征提取可分为二进制文件反编译、操作码提取、数据预处理三个子模块，主要作为RNN模型训练的输入。而模型训练又可以分为模型结构构建、确定损失函数与优化、参数训练三个子模块，主要进行模型的选择及训练，提高分类检测的准确率。集成模块中，则分为模型保存、模块构建和API提供服务三个子模块，将训练模块集成到二进制文件开放管理平台中，并提供稳服务。



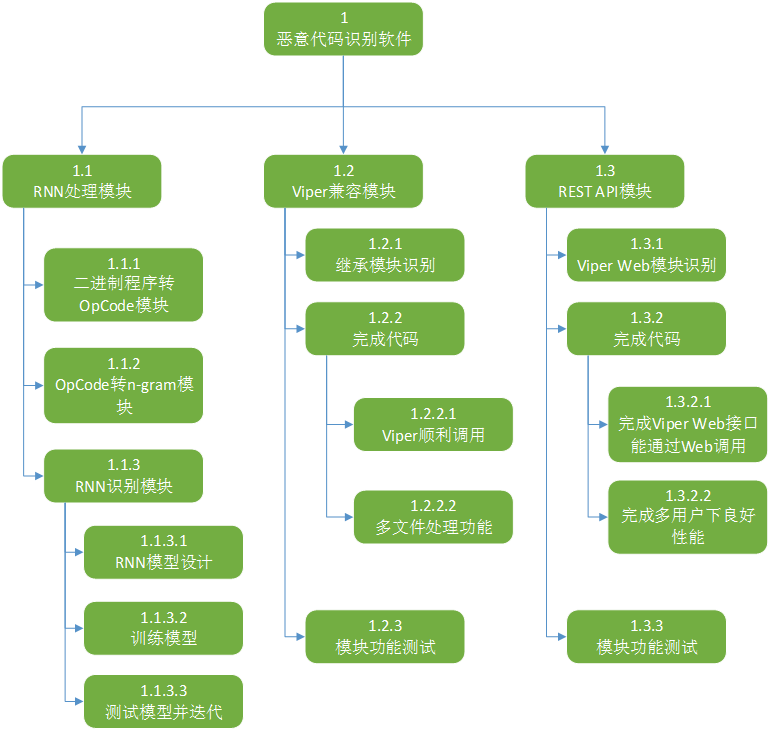
### ·功能列表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **系统** | **功能模块** | **子功能** | **功能编号** | **优先级** |
| 基于RNN的恶意代码检测项目 | RNN识别处理模块 | 二进制文件提取OpCode | 01 | 高 |
| 划分训练集测试集 | 02 | 高 |
| 训练模型 | 03 | 高 |
| 测试模型迭代 | 04 | 中 |
| Viper兼容模块 | Viper Web模块识别 | 05 | 高 |
| 模块功能测试 | 06 | 高 |
| 完成viper兼容 | 07 | 高 |
| Viper调用 | 08 | 高 |
| 多文件处理 | 09 | 低 |
| RESTFUL API调用 | 10 | 高 |
| RESTFUL API模块 | 调用功能测试 | 11 | 高 |
| 多文件处理 | 12 | 低 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

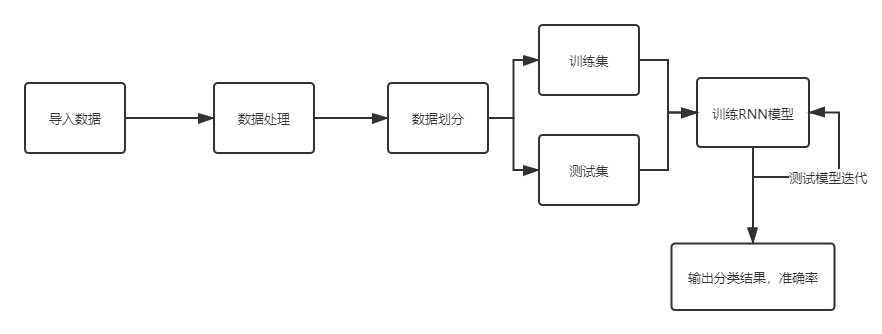
## 核心业务流程

核心业务流程如上图所示，主体分为三个循序渐进的子流程，分别是实现RNN处理检测、Viper兼容模块和实现Restful API接口调用。

首先进行RNN的处理检测，实现对恶意代码文件的特征提取和训练检测，达到较高的准确率要求和较快的响应速度；之后进行Viper平台的模块兼容，实现在该平台上的集成与顺利调用；最后将模块抽象为API接口，实现能够对外提供服务的功能，使得用户可以通过web调用该服务。



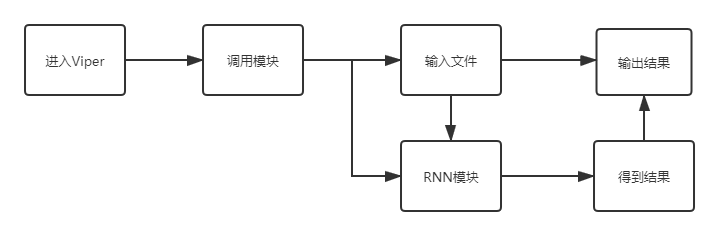
### ·RNN识别处理模块



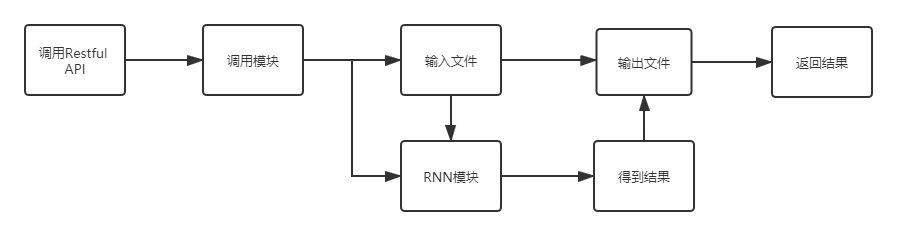
该业务流程主要工作如上图所示。该模块需要对恶意代码的二进制文件进行反编译，提取其中的操作码作为数据集进行导入；对提取的序列类型数据文件进行预处理操作，再划分为训练集和测试集，输入到RNN模型中进行训练。

其中在训练的过程中需要对参数反复调整，找到最优的模型参数，达到较高的准确率。对模型进行测试，使得该模块能够输出测试代码的分类结果及其类别概率。

### ·Viper兼容模块

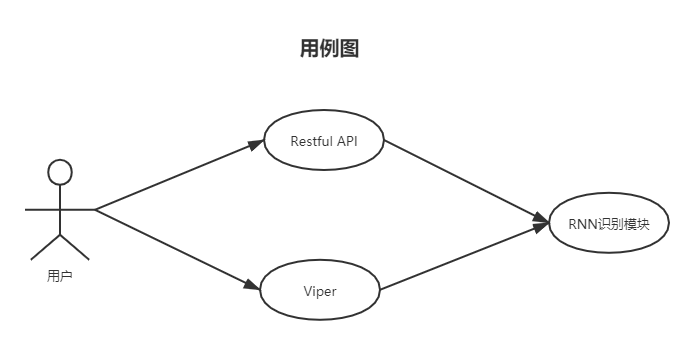
该业务流程主要工作如上图所示。该模块需要将RNN识别模块集成到Viper平台中，应根据Viper的扩展文档了解模块集成接口的编写，使得能够在该平台上直接调用该模块，模型检测并输出预测分类结果。

### ·RESTFUL API模块



该业务流程主要工作如上图所示。该模块需要将RNN检测模块抽象为接口对外提供服务，使得用户调用该接口时能够实现对恶意代码的检测及分类，且具有较高的准确率和响应速度。

## 核心用例图

**

# 功能性需求

## RNN识别处理模块

**该模块的核心需求：实现基于RNN和OpCode的恶意代码检测功能。**

|  |  |  |  |
| --- | --- | --- | --- |
| **需求编号** | **功能点** | **功能描述** | **优先级** |
| 01 | 导入数据 | 导入恶意代码样本 | 高 |
| 02 | 数据处理 | 将恶意代码样本转为Opcode | 高 |
| 03 | 数据划分 | 将Opcode按长度划分 | 高 |
| 04 | 导入训练集 | 将部分划分结果标记为训练集 | 高 |
| 05 | 导入测试集 | 将部分划分结果标记为测试集 | 高 |
| 06 | 训练RNN | 通过RNN模型对恶意代码的Opcode进行分类 | 高 |
| 07 | 测试模型迭代 | 迭代，优化模型 | 中 |

该子模块主要包括了7个功能需求，大部分为高优先级，为整个项目的核心功能需求。此模块的整体输入为带有标签的恶意代码家族样本数据集，输出为恶意代码家族的分类及其准确率。

## Viper兼容模块

**该模块的核心需求：将检测模块集成到开放恶意代码管理框架中。**

|  |  |  |  |
| --- | --- | --- | --- |
| **需求编号** | **功能点** | **功能描述** | **优先级** |
| 08 | Viper Web模块识别 | 实现的模块能被Viper识别 | 高 |
| 09 | 完成viper兼容 | 将检测模块集成到Viper平台 | 高 |
| 10 | Viper调用 | 使用Viper调用完成的模块 | 高 |
| 11 | 支持批量处理 | 支持多文件输入  以表格形式返回多个检测结果 | 低 |

该子模块主要包括了4个功能需求，大部分为高优先级，其中低优先级的批量处理可以按项目开发进度进行调整。此模块的整体输入为命令行调用检测模块，并输入恶意代码文件，输出为该恶意代码的分类及其准确率。

## RESTFUL API模块

**该模块的核心需求：提供Restful API访问接口。**

|  |  |  |  |
| --- | --- | --- | --- |
| **需求编号** | **功能点** | **功能描述** | **优先级** |
| 12 | RESTFUL API调用 | 遵循RESTFUL API的调用规范 | 高 |
| 13 | 支持批量处理 | 支持多文件输入  以表格形式返回多个检测结果件 | 低 |

该子模块主要包括了2个功能需求，暂时不作为整个项目的核心需求考虑。其中低优先级的批量处理可以按项目开发进度进行调整。此模块的整体输入为web调用该模块服务接口，输入恶意代码文件，输出为该恶意代码的分类及其准确率。

# 非功能性需求

## 质量需求

### ·性能

1. 准确率：对恶意代码检测的准确率应大于75%。
2. 响应速度：单文件响应时间在1s内完成；多文件处理生成相应结果文件保存在用户本地或以表格形式呈现，平均响应时间不超过5s。
3. 负载情况：能够容纳多用户同时响应。
4. 稳定性：与服务器连接稳定，可保持长时间运行。

### ·易用性

（1）易操作性：遵循标准的Restful API 调用方法，使得接口调用对于用户来说方便学习和操作；遵循viper调用方法的标准，使得用户在Viper中调用该模块便于学习和操作。

（2）易浏览性：输出结果符合用户阅读习惯，检测分类结果和准确率旁有文字提示说明。

### ·安全性

1. 恶意代码隔离性：应具备病毒防范能力，确保检测过程中恶意代码不会对主机造成侵入。

*其他非功能需求主要查看需求文档规格说明 + 测试执行记录内容。*

*指的是保护软件的要素，以防止各种非法的访问、使用、修改、破坏或者泄密。这个领域的具体需求产品的安全性、保密性和完整性三方面需求。*

*例如：要求对接入系统的用户进行身份验证。对不同角色的用户设置不同的权限，通过角色定义实现不同角色个性化菜单的定制，有效控制用户的功能权限。系统应提供日志记录和管理功能，记录所有用户访问系统的全部活动，并能够形成审计报告。要求在传输过程中对数据进行加密处理，保证数据传输的安全性和完整性。系统应具备病毒防范能力。防止主机崩溃方法和数据备份方法等。*

### ·可靠性和健壮性

*在这一部分应对所有的影响软件的可靠性需求进行足够详细的描述。应注意用数字说明所要求的可靠程度。 同时避免如“24x7”这样的陈述。*

*例如使用年度正常运行时间、月正常运行时间、维护时间、当机时间来说明系统的可靠程度；使用可允许的缺陷数量来界定系统质量，如最大缺陷数量、缺陷比例、安全操作——系统强壮性要求和操作的有效性要求，比如用户误操作的系统容错能力、操作的正常次序要求和有效性输入检查等等。*

*通常给出平均无故障时间或两次故障间的平均间隔时间等。*

### ·可维护性

*规定若干需求以确保软件是可维护的。例如：*

* 1. *软件模块所需要的特殊的耦合矩阵；*
  2. *使用行业标准、编码标准、开放式结构、可兼容语言、备份及复原和数据交换等。*
  3. *规定把软件从一种环境移植到另一种环境所要求的用户程序，用户接口兼容方面的约束。*

### ·可扩展性

1. 实现的模块可以作为viper框架的拓展，并且自身具有良好的可扩展性。

*说明该软件在需求或环境发生某些变化时，该软件对这些变化的适应能力的要求，如：*

1. *需求及流程变化；*
2. *操作方式变化；*
3. *机构人员变化；*
4. *空间地点变化(移动用户、分布式)；*
5. *操作系统环境变化。*

## 约束

1. **人力时间的约束**：开发小组一共6人，时间为3周。
2. **资金的约束**：项目计划资金为10万元
3. **项目开发环境**：Linux平台、Python 3.6及以上。
4. **项目开发工具**：二进制文件管理框架Viper、机器学习库PyTorch。
5. **恶意代码数据集**：实时恶意软件存储库theZoo、APTMalware样本库。

## 接口需求

*详细说明对系统的用户界面等的要求。还可包括和其它系统的接口、地址、协议等。*

### ·用户接口

*提供用户使用软件产品时的接口需求。例如，如果系统的用户通过客户端进行操作，就必须指定如下要求：*

* *对屏幕格式的要求；*
* *报表或菜单的页面、打印格式等用户对软件外观风格的一种要求。如：公司标志，界面色彩基调等。规格的定义方式可以采用草图或静态原型的方式表示，一般描述分为两个部分：整体描述和基于每个界面的细节描述。*
* *输入输出的相对时间；*
* *程序功能键的可用性。*

### ·硬件接口

要指出软件产品和系统硬部件之间每一个接口的逻辑特点和交互方式。还可能包括如下事宜：支撑什么样的设备，如何支撑这些设备，有何约定。

### ·软件接口

*在此要指定需使用的其他软件产品（例如，数据管理系统、操作系统或有关软件包），以及同其他应用系统之间的接口。对每一个所需的软件产品，要提供如下内容：*

1. *名字；*
2. *助记符；*
3. *规格说明号；*
4. *版本号；*
5. *来源。*

*对于每一个接口，这部分应说明与软件产品相关的接口软件的目的，并根据信息的内容和格式定义接口，但不必详细描述任何已有完整文件的接口，只要引用定义该接口的文件即可。*

### ·通讯接口

*指定各种通信接口。例如，局部网络的协议等等。*

## 技术需求

### ·开发环境需求

|  |  |  |
| --- | --- | --- |
| **类别** | **项目** | **需求描述** |
| 软件需求 | 操作系统 | Linux (Ubuntu 18.04 x64) |
| 编程环境 | Python 3.6及以上 |
| 基本配置 | Viper 2.0  PyTorch 0.2.0\_3 |
| 硬件需求 | CPU | 多核高速，以满足模型训练的高算力需求 |
| 运行内存 | 8G及以上 |
| 硬盘 | 虚拟机建议20G及以上，仅存储需求8G左右 |

## 文档需求

本项目需要完成文档结构如下所示。

| - 技术混合文档

| - 需求规格说明书

| - 概要设计说明书

| - 测试计划及用例

| - 风险管理报告

| - 管理文档

| - 项目工作计划书

| - 人员分工

| - 计划执行情况

| - 会议记录

| - 经济决策工具使用情况

| - 项目管理工具使用情况

| - 用户使用手册

| - 工作汇报文档

| - 项目总结

# 验收标准

详细说明对系统的验收要求。此要求将作为验收测试计划和测试的基线。如果所开发的产品能满足此要求，则项目可结束并由客户方按合同规定付款。