

[SCA] 日志设计文档

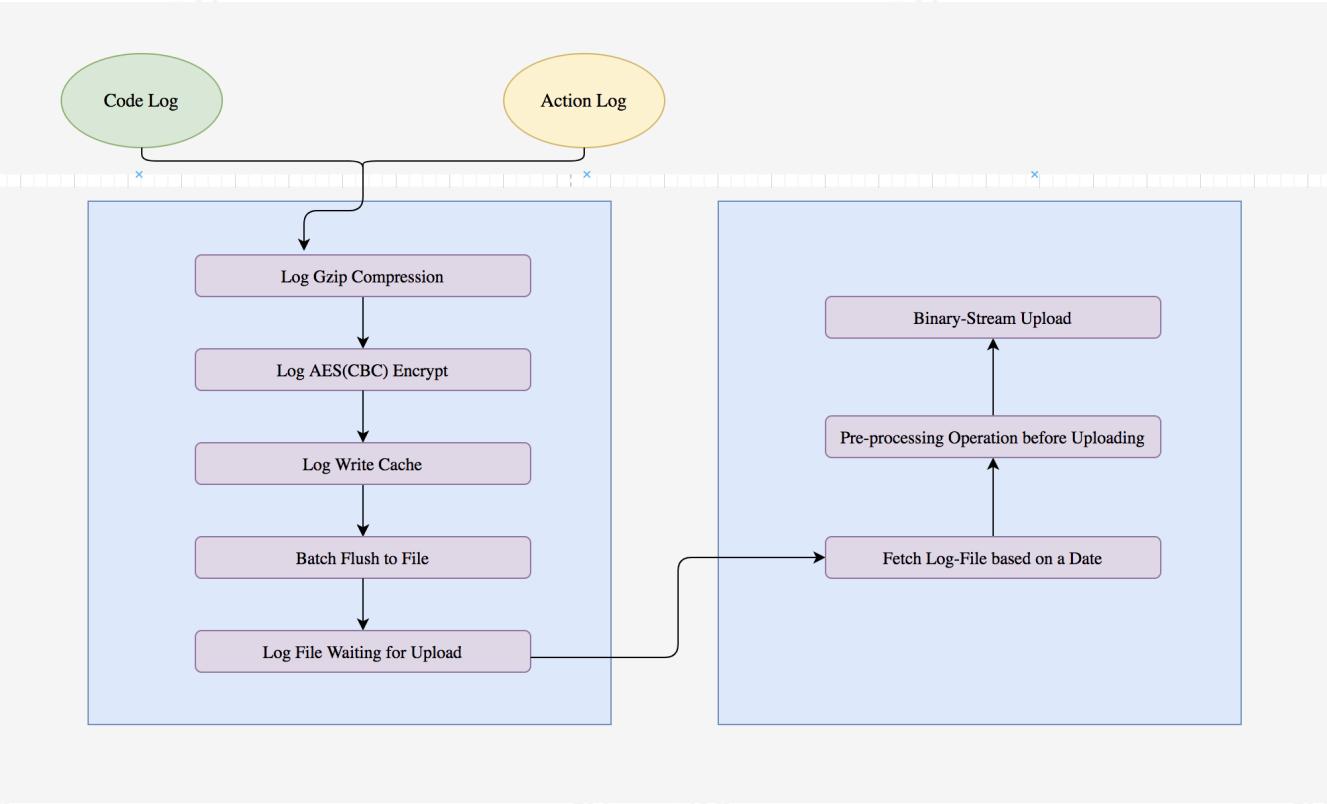
背景

为方便问题定位和数据分析，在开发过程中需要对一些用户行为、异常处理和关键代码打上日志。这样就需要有一种对性能要求不高、安全且不会丢失日志的框架设计，对业务提供易用的接口，同时能够方便地让APP上报日志，供开发人员分析问题。

功能需求

功能点	功能描述
日志存储	采用“先压缩再加密”的顺序，使用流式的加密和压缩，减少CPU使用。对于大日志分片处理，通过MMAP机制解决日志丢失问题，使用AES(CBC)进行日志加密确保日志安全性。
日志上报	APP采用POST请求通过二进制流方式进行日志上报。

整体设计



写日志接口

可记录不同类型的日志信息，日志类型可分为以下几类：

- 1. 代码级日志
- 2. 用户行为日志
- 3. 网络日志

日志上传

- 采用POST请求通过二进制流方式进行日志上报，后端需要字段作为请求参数，日志作为请求体，日志上报至Supply Chain数据平台。

Native Library

Logan

本地日志记录主要基于Logan，核心模块包括：

- gzip：对写入的日志进行压缩
- AES：对压缩后的日志进行AES(CBC)加密
- mmap：使用逻辑内存对磁盘文件进行映射，操作内存就相当于操作文件，回写文件交给OS保证，确保不会丢失日志
- I/O：根据日期(2020-10-10)对日志进行读写

实现方案

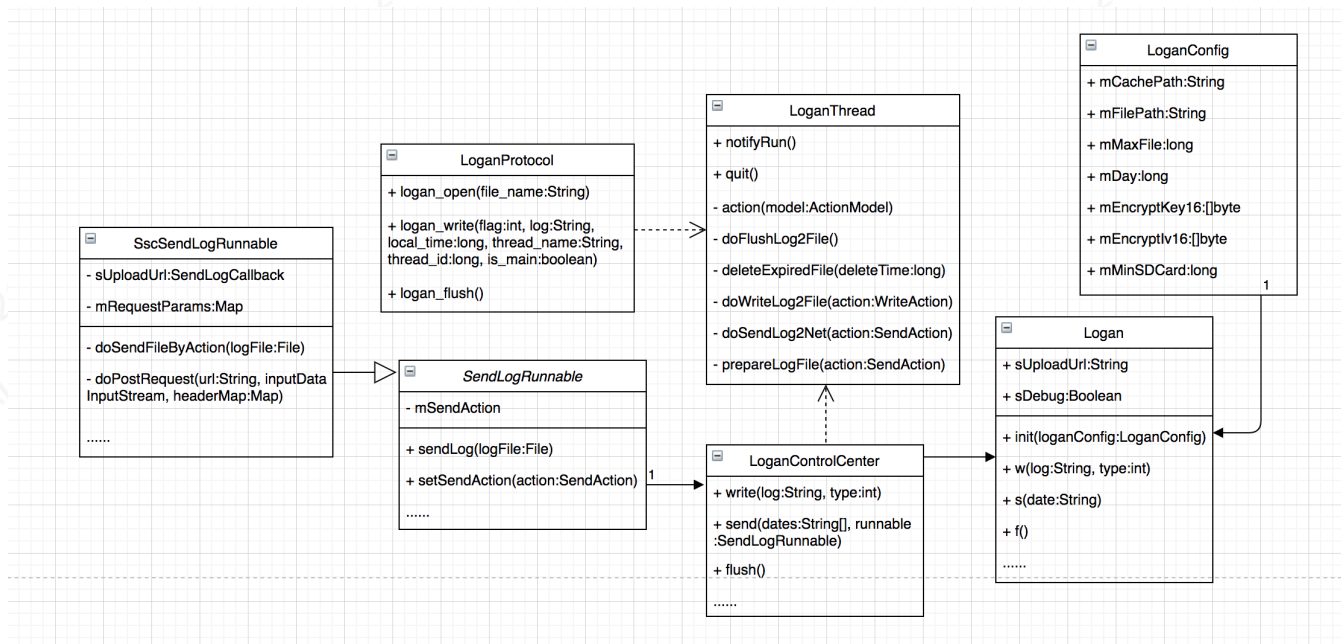
协议

key	type	description
log	string	日志内容
type	int	日志类型
ct	long	本地时间戳
tname	string	线程名称
tid	int	线程id
main	bool	是不是主线程

加密

使用AES(CBC)加密，指定16位的AES加密Key以及加密IV

- 准备对称密钥与初始向量：随机产生 AES 对称密钥 AES Key 及初始向量 IV；
- 对称加密：使用 AES Key 及 IV 对日志明文进行 AES(CBC) 对称加密，输出日志密文。



参考资料

[Upload API](#)