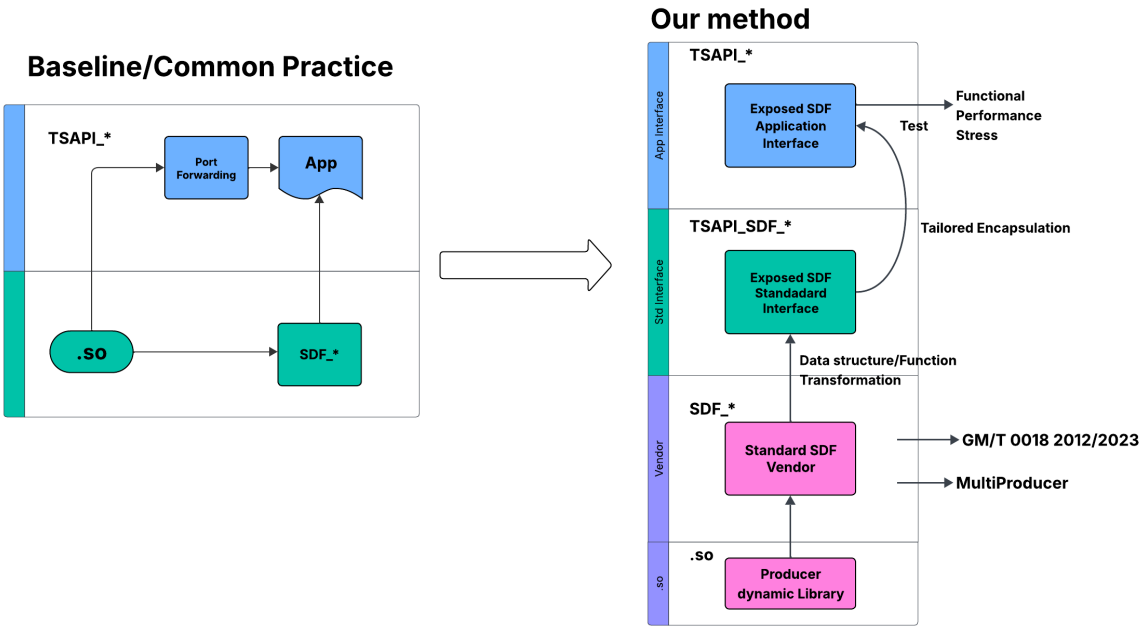
项目名称：

补全 Tongsuo 项目的 SDF 和 SKF 接口能力

方案描述

铜锁(Tongsuo)作为开源基础密码库，在保障数据全生命周期的安全性与隐私性方面发挥着关键作用。目前，其仅支持部分 SDF 接口且缺少 SKF 接口支持，限制了密码应用程序及用户对密码卡和智能密码钥匙设备的管理与控制效能。

鉴于此，本项目旨在依据 GM/T 0018-2023 和 GM/T 0016-2023 标准，完善 Tongsuo 的接口功能，实现对完整 SDF 接口的全面支持以及对 SKF 接口从无到有的开发，助力密码应用程序与用户更高效地管理密码设备。



项目进度

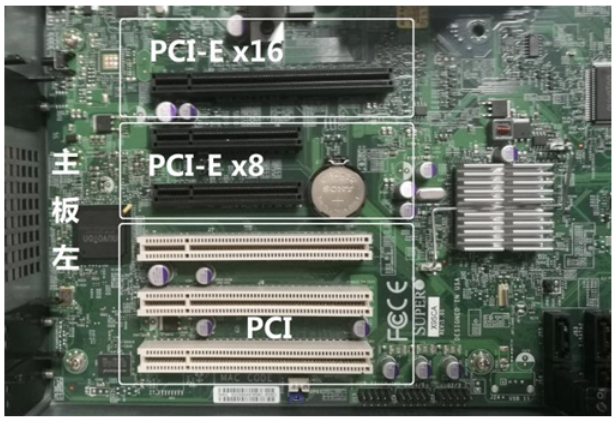
已完成工作：

- SDF/SKF接口适配层的构建，与厂商库(三卫信安)密码卡的完整适配，数据结构，函数指针的适配
- SDF/SKF用户编程接口，对齐2012标准，能够编译进项目，暴露符号表，供开发者使用标准接口
- SDF/SKF应用编程接口，为命令行测试定制的包装接口，为压力测试/功能测试提供的包装接口。
- SDF接口功能测试，编写测试代码对<openssl/sdf.h>进行调用测试，使用apps/openssl命令行工具进行功能可用性验证。
- 搭建了命令行工具框架，实现了标准化选项解析和功能分发，对齐行业CLI开发规范。
- 集成了GmSSL软实现动态库，完成了库加载、函数绑定、接口测试等关键环节。
- 全面梳理并实现了SDF（GM/T 0018-2023）、SDF(GM/T 0018 2012)、SKF(GM/T 0016-2012)、SKF(GM/T 0016-2023) SKF(GM/T 0017-2023) 标准接口，涵盖设备管理、密钥管理、加解密、签名验签、会话管理等功能。
- 梳理了硬件驱动、符号导出、构建系统等工程细节，实现了多平台兼容和高质量集成。

遇到的问题及解决方案：

硬件支持问题：

密码卡



PCIE插槽的密码卡无法接入笔记本电脑，需要Linux服务器或台式机环境支持。

在社区中联系到有需求的同行，想办法解决硬件支持的问题，搭建环境进行远程开发测试。学习了银河麒麟系统的使用方法，复杂情况下使用docker镜像用ssh反向连接实现内网服务器的远程连接，学习了内核模块的安装，驱动的使用，驱动的检查，学习了HSM使用/管理/开发的完整流程。通过广泛联系爱好者和同行，了解行业需求，解决实际问题。

```
[dhx@sm Tongsuo]$ lscpu
架构:                    aarch64
CPU 运行模式:            64-bit
字节序:                  Little Endian
CPU:                     64
在线 CPU 列表:           0-63
每个核的线程数:          1
每个座的核数:            4
座:                      16
NUMA 节点:               8
厂商 ID:                 Phytium
型号:                    2
型号名称:                FT-2000+/64
步进:                    0x1
BogoMIPS:                100.00
```

国密标准适配问题：

之前预想的国密会统一到GM/T 2023标准，但随着了解的深入，现有的厂商库基本都在支持2012标准，为了实现硬件测试。通过宏定义的方式向前兼容2023标准，解决标准适配的问题。广泛调查各种标准，确定最标准实际方式。并对标准进行梳理分析，编写开发文档。

多厂商的适配问题：

不同厂商对标准接口的实现大部分一样，但是具体到鉴权方式，同名接口的功能，同名接口的参数个数，存在细碎的不兼容问题。为了解决多厂商适配问题，保证项目可扩展性，设计构建了统一的适配层。

项目编译/项目构建问题：

Tongsuo使用perl脚本构建项目，为了完成项目的顺利构建，钻研了项目构建方式，顺利编译项目。

接口测试问题：

导出正确的符号表后，函数调用失败
开启项目调试选项，使用gdb反复调试，定位问题代码进行修复。

```
[dhx@sm apps]$ nm openssl | grep "SDF"
0000000009630d0 B SDF_ExchangeDigitEnvelopeBaseOnECC
0000000000459f5c t SDF_GenerateKey_loop
0000000000963098 B SDF_GenerateKeyPair_ECC
000000000064c288 T SDF_GetErrorString
0000000000632240 T TSAPI_SDF_CalculateMAC
0000000000630d28 T TSAPI_SDF_CloseDevice
0000000000630dec T TSAPI_SDF_CloseSession
000000000063244c T TSAPI_SDF_CreateFile
00000000006321b4 T TSAPI_SDF_Decrypt
00000000006325b0 T TSAPI_SDF_DeleteFile
0000000000631aa8 T TSAPI_SDF_DestroyKey
0000000000632128 T TSAPI_SDF_Encrypt
0000000000631bc4 T TSAPI_SDF_ExchangeDigitEnvelopeBaseOnECC
0000000000631b04 T TSAPI_SDF_ExchangeDigitEnvelopeBaseOnRSA
0000000000631420 T TSAPI_SDF_ExportEncPublicKey_ECC
00000000006310cc T TSAPI_SDF_ExportEncPublicKey_RSA
0000000000631398 T TSAPI_SDF_ExportSignPublicKey_ECC
000000000063103c T TSAPI_SDF_ExportSignPublicKey_RSA
0000000000632068 T TSAPI_SDF_ExternalEncrypt_ECC
0000000000631ccc T TSAPI_SDF_ExternalPublicKeyOperation_RSA
0000000000631e7c T TSAPI_SDF_ExternalVerify_ECC
00000000006318ac T TSAPI_SDF_GenerateAgreementDataAndKeyWithECC
0000000000631734 T TSAPI_SDF_GenerateAgreementDataWithECC
0000000000632614 T TSAPI_SDF_GenerateKey
00000000006314a8 T TSAPI_SDF_GenerateKeyPair_ECC
```

构建命令行应用

调用密码卡涉及打开设备，打开会话，获取权限，生成公钥，生成对称密钥……等过程，
由于这些繁琐的过程，采用传统OpenSSL命令行应用的设计方式需要传入大量参数，造成使用困难，代码可维护性差等等问题。
这里设计命令行应用仅进行功能可用性验证，后续完善命令行应用功能，考虑构建交互性更强的方式完成。

```
[dhx@sm Tongsuo]$ cd apps
[dhx@sm apps]$ openssl
Tongsuo> sdf -help
Usage: sdf [options]

General: Device Management Options
 options:
  -help                       Display this summary
  -version                    Display version information
  -device-info                Display device information
  -random val                 Generate random number of specified length

Export: Asymmetric Public Key Export Options:
 options:
  -export-encpubkey-ecc val     Export ECC encryption public key
  -export-signpubkey-ecc val    Export ECC signature public key
  -export-encpubkey-rsa val     Export RSA encryption public key
  -export-signpubkey-rsa val    Export RSA signature public key

Generation: Session Key Generation Options:
 options:
  -generatekeywith-kek val      Generate session key using KEK
```

```
  -generatekeywith-ipk-rsa val  Generate session key using RSA internal public key
  -generatekeywith-epk-rsa val  Generate session key using RSA external public key
  -generatekeywith-ipk-ecc val  Generate session key using ECC internal public key
  -generatekeywith-epk-ecc val  Generate session key using ECC external public key

Import: Session Key Import Options:
 options:
  -importkeywith-kek val        Import session key using KEK
  -importkeywith-isk-rsa val    Import session key using RSA internal private key
  -importkeywith-isk-ecc val    Import session key using ECC internal private key

Crypto: Operation Test Options:
 options:
  -extrsatest                   External RSA operation test
  -intrsatest                   Internal RSA operation test
  -inteccsigntest               Internal ECC signature test
  -exteccsigntest               External ECC signature test
  -exteccenctest                External ECC encryption test
  -symencdectest                Symmetric encryption/decryption test
  -calculatemac                 Calculate MAC test

Parameters:
 INDEX parm                     Key index number (1-100)
 LENGTH parm                    Random number length in bytes
```

SDF接口标准理解

SDF接口标准理解难度大，涉及硬件安全模块（HSM）、密钥管理、抗侧信道等复杂概念。通过查阅标准文档、厂商手册、与专家交流（如密标委、三未信安），逐步理清接口参数、算法标识、密钥生命周期等细节。

构建系统

Tongsuo/GmSSL等密码库的构建系统复杂，模块裁剪、符号导出、动态库加载等环节容易出错。通过分析Configure/build.info脚本、调试日志、gdb调试，掌握了分散式构建和符号绑定的工程方法。命令行工具开发涉及选项解析、功能分发、接口对齐等问题。采用表驱动+枚举令牌的行业最佳实践，提升了可维护性和扩展性。

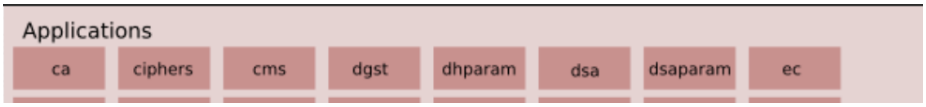
TSAPI接口与SDF标准

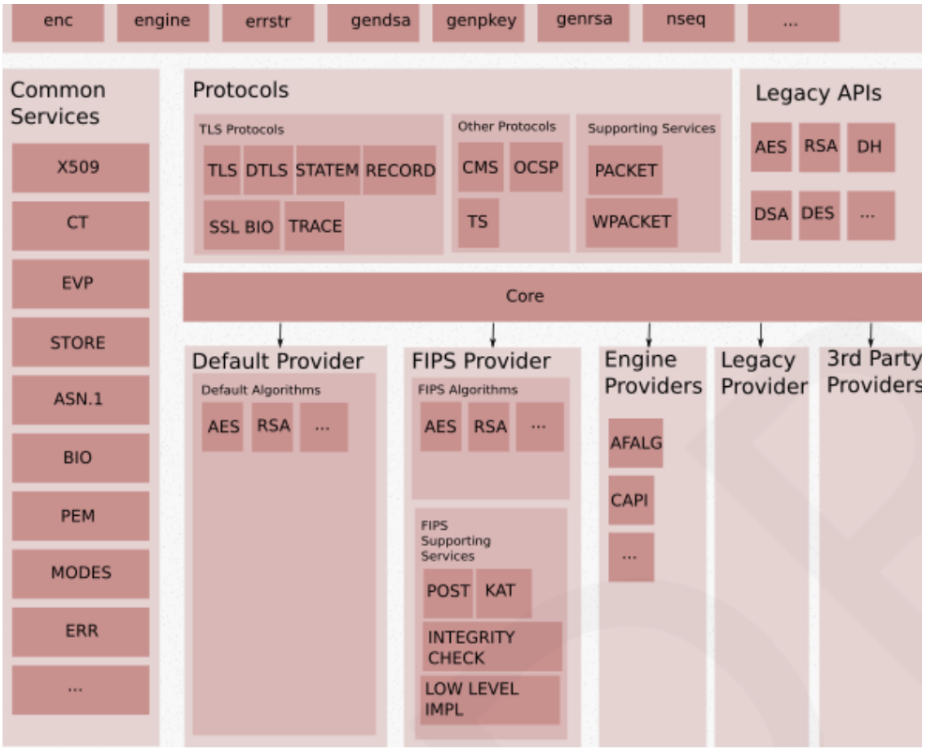
不完全一致，存在身份认证、密钥访问等实现差异。通过梳理调用链、裁剪碎片化代码，实现了标准化适配。
测试与集成环节，软实现与硬件库兼容性、接口覆盖率、符号导出等问题。通过接口清单整理、集成测试框架、日志与调试工具，确保了功能完整和可用性。

后续工作期望

- [ ] 对 SDF 和 SKF 接口进行全面的功能测试、性能优化以及安全性加固，确保在各种复杂场景下的稳定可靠运行。
- [ ] 对 SDF 和 SKF 接口EVP层，接入Provider
- [ ] 编写测试用例与文档
- [ ] 整理并完善测试用例，
- [ ] 形成完整的测试报告。

尽力取得更多社区支持，联合社区同行，构建SDF/SKF的provider支持




enc
engine
errstr
gendsa
genpkey
genrsa
nseq
...
Common Services
X509
CT
EVP
STORE
ASN.1
BIO
PEM
MODES
ERR
...
Protocols
TLS Protocols
TLS
DTLS
STATEM
RECORD
SSL BIO
TRACE
Other Protocols
CMS
OCSP
TS
Supporting Services
PACKET
WPACKET
Legacy APIs
AES
RSA
DH
DSA
DES
...
Core
Default Provider
Default Algorithms
AES
RSA
...
FIPS Provider
FIPS Algorithms
AES
RSA
...
FIPS Supporting Services
POST
KAT
INTEGRITY CHECK
LOW LEVEL IMPL
Engine Providers
AFALG
CAPI
...
Legacy Provider
3rd Party Providers