WeDPR Documentation

发布 ***v1.5.0***

WeDPR Community

2021 年 **09** 月 **29** 日

Contents

CHAPTER 1

WeDPR-Lab是什么

WeDPR是一系列**即时可用场景式**隐私保护高效解决方案套件和服务（参见WeDPR白皮书），由微众银行区块链团队自主研发。方案致力于解决业务数字化中隐私不"隐"、共享协作不可控等隐私保护风险痛点，消除隐私主体的隐私顾虑和业务创新的合规壁垒，助力基于隐私数据的核心价值互联和新兴商业探索，营造公平、对等、共赢的多方数据协作环境，达成数据价值跨主体融合和数据治理的可控平衡。

WeDPR具备以下特色和优势：

- **场景式解决方案**：已基于具有共性的场景需求，提炼出公开可验证密文账本、多方密文决策、多方密文排名、多方密文计算、多方安全随机数生成、选择性密文披露等高效技术方案框架模板，可应用于支付、供应链金融、跨境金融、投票、选举、榜单、竞拍、招标、摇号、抽检、审计、隐私数据聚合分析、数字化身份、数字化资质凭证、智慧城市、智慧医疗等广泛业务场景。
- **即时可用**：高性能、高易用、跨平台跨语言实现、不依赖中心化可信服务、不依赖可信硬件、支持国密算法标准、隐私效果公开可验证，5分钟一键构建示例应用。
- **透明可控**：隐私控制回归属主，杜绝数据未授权使用，在『数据可用而不可见』的基础上，进一步实现数据使用全程可监管、可追溯、可验证。

WeDPR全面拥抱开放，将陆续开源一系列核心算法组件，进一步提升系统安全性的透明度，提供更透明、更可信的隐私保护效果。WeDPR-Lab就是这一系列开源的**核心算法组件**的集合。

1.1 为什么使用WeDPR-Lab

WeDPR-Lab目前主要包含两个开源仓库隐私保护场景式解决方案WeDPR-Lab-Core和密码组件WeDPR-Lab-Crypto。

WeDPR-Lab-Core：https://github.com/WeBankBlockchain/WeDPR-Lab-Core
https://gitee.com/WeBankBlockchain/WeDPR-Lab-Core

WeDPR-Lab-Crypto：https://github.com/WeBankBlockchain/WeDPR-Lab-Crypto
https://gitee.com/WeBankBlockchain/WeDPR-Lab-Crypto

- **丰富的多语言接口**

对于隐私保护场景式解决方案WeDPR-Lab-Core和密码组件WeDPR-Lab-Crypto，我们均提供了Java和C的FFI适配接口，具有涵盖桌面端、服务端、移动端的全平台SDK，实现了跨语言、跨平台的适配。

- **可插拔的接口级设计**

WeDPR倡导"依赖解耦、配置灵活"的设计理念，设计上极致模块化，每个算法都可独立复用。用户可按照实际所需功能定制化WeDPR使用接口，最小化减少依赖，实现了开发者对WeDPR系列开发包的自由裁剪和轻量部署。

- **高性能，高安全**

WeDPR系列开发包具有高效的内存利用率与运行速度，性能与C和C++实现匹敌，同时具有内存安全与线程安全，为开发者提供了高性能、高安全的隐私保护算法组件及易用、易扩展的使用体验。

WeDPR-Lab-Crypto v1.2.0版本开源主要内容如下：

- 核心密码算法组件：**n选k不经意传输算法**：
  - 其中，n和k均为任意正整数，k<n。

  对于以下场景：
  - 数据方的数据目录中共有n条消息记录
  - 查询方选择k个消息的索引向数据方查询消息

  不经意传输算法能实现的具体隐私效果是：
  - 数据方无法得知查询方的查询索引，即：查询方查询索引隐私；
  - 除了所查索引的消息外，查询方无法得知数据方数据目录中的其他消息，即：数据方数据隐私。
- **二进制接口**，包括所有核心密码算法的高性能二进制接口；

WeDPR-Lab-Crypto v1.1.0版本开源主要内容如下：

- **核心密码算法组件**，包括：
  - 分组加密算法：包括AES-256、国密SM4；
  - 哈希算法：包括SHA3、BLAKE2、RIPEMD-160；
  - 椭圆曲线计算：包括椭圆曲线BN128的点加、点乘及双线性对操作；
  - 数字签名算法：包括Ed25519；
  - 零知识证明的聚合验证：包括加和证明的聚合验证、乘积证明的聚合验证。
- **二进制接口**，包括所有核心密码算法的高性能二进制接口；
- **FFI接口**，支持交叉编译跨语言、跨平台所调用的FFI适配接口。

WeDPR-Lab-Crypto v1.0.0版本开源主要内容如下：

- **核心密码算法组件**，包括：
  - 基础编解码；
  - 公钥加解密算法，包括基于Secp256k1曲线的ECIES加解密；
  - 哈希算法，包括Keccak256哈希算法与国密SM3；
  - 签名及验证，包括ECDSA签名与国密SM2；
  - 离散对数系统的零知识证明算法，包括加和证明及验证、乘积证明及验证；
  - 零知识范围证明及验证；
  - 基于椭圆曲线的可验证随机函数VRF(Verifiable Random Functions)。
- **FFI接口**，支持交叉编译跨语言、跨平台所调用的FFI适配接口。

WeDPR-Lab-Core v1.5.0版本开源主要内容如下：

- **多方密文决策ACV核心算法**，支持全密文决策、全流程可验证的多方隐私决策。
- ACV场景式解决方案的一个**交互式样例**，实现以下主要功能：
  - 密文空白选票的颁发

  - 决策密文选票的生成
  - 密文决策过程的零知识证明生成与验证
  - 密文决策结果的汇总
  - 汇总过程的零知识证明生成和验证
  - 决策结果的验证
- **Rust SDK**，封装底层算法，提供易用、易扩展、跨语言的编程接口；
- 其他**基础工具**代码。

WeDPR-Lab-Core v1.4.0版本开源主要内容如下：

- 将WeDPR-Lab Crypto v1.1.0新增的零知识证明的聚合验证算法运用于VCL公开可验证密文账本。
- 聚合验证算法在VCL中应用后的具体效果是：

对于多组密文记录及其加和关系证明（或乘积关系证明），无需针对每个加和证明（或乘积证明）分别进行验证，而只需执行一次聚合验证，就可对所有加和证明（或乘积证明）进行验证。

聚合验证算法通过减少开销较大的椭圆曲线点运算的个数，将VCL中对应零知识证明批量验证的效率提升了近60%。

WeDPR-Lab-Core v1.3.0版本开源主要内容如下：

- **密钥生成及管理的核心算法**，包括：
  - 密钥助记词的生成
  - 基于助记词的密钥生成
  - 基于分层结构的密钥派生
- **Rust SDK**，封装底层算法，提供易用、易扩展、跨语言的编程接口；
- **FFI接口**，支持交叉编译跨语言、跨平台所调用的FFI适配接口；
- 其他**基础工具**代码。

WeDPR-Lab-Core v1.2.0版本开源主要内容如下：

- **选择性认证披露SCD核心算法**，支持多种断言证明的高效稳定实现；
- SCD场景式解决方案的一个**交互式样例**，实现以下主要功能：
  - 选择性披露证书颁发
  - 零知识断言披露证明
  - 零知识断言披露验证
  - 选择性属性明文披露和验证
- **Rust SDK**，封装底层算法，提供易用、易扩展、跨语言的编程接口；
- **FFI接口**，支持交叉编译跨语言、跨平台所调用的FFI适配接口；
- 其他**基础工具**代码。

WeDPR-Lab-Core v1.1.0版本开源主要内容如下：

提供更为丰富的密码学算法组件，具体包括：

- **密码算法组件**
  - 签名验证
  - 哈希算法
  - 公钥加解密ECIES
- **FFI接口**，新增支持交叉编译、跨语言、跨平台所调用的FFI适配接口。

WeDPR-Lab-Core v1.0.0版本开源主要内容如下：

- **公开可验证密文账本VCL**的一个**交互式样例**，实现以下主要功能：
  - 密文金额发行
  - 密文金额四则运算关系验证
  - 密文金额范围验证
- **三类零知识证明算法**的高效稳定实现，包括
  - 密文加和关系证明
  - 密文乘积关系证明
  - 密文范围证明
- **Rust SDK**，封装底层算法，提供易用、易扩展、跨语言的编程接口；
- **FFI接口**，支持交叉编译跨语言、跨平台所调用的FFI适配接口；
- 其他**基础工具**代码。

我们期望能够通过代码开源的方式：

- 有效降低使用隐私保护算法组件的技术门槛；
- 减少业务系统集成隐私保护特性的开发成本；
- 助力全行业伙伴安全、合规地开展数据业务。

欢迎社区伙伴参与WeDPR-Lab的共建，一起为可信开放数字新生态的构建打造坚实、可靠的技术底座。

CHAPTER 2

安装

WeDPR-Lab核心库使用跨平台的系统语言Rust编写，体验前需要安装相关的开发环境。

2.1 安装Rust环境

1. 安装nightly版本的Rust开发环境，可参考示例安装指引。
2. 若有疑问，可进一步参考Rust官方文档。

2.2 拉取WeDPR-Lab源代码

使用git命令行工具，执行如下命令。

```
git clone https://github.com/WeBankBlockchain/WeDPR-Lab-Core.git
```

或

```
git clone https://gitee.com/WeBankBlockchain/WeDPR-Lab-Core.git
```

CHAPTER 3

公开可验证密文账本场景方案

3.1 VCL解决方案优势

公开可验证密文账本（Verifiable Confidential Ledger, VCL）具有以下核心优势：

- 全密文记账，满足企业级安全和性能要求
- 账目密文数值之间的算术、逻辑全关系证明
  - A + B = C, A * B = D
  - A > B, A != B

其典型的应用场景如下：

- 涉及数字化资产流通的场景
  - 支付、清算
- 涉及多方之间共享账本信息的场景
  - 供应链金融
  - 跨境金融服务

其重要功能和指标如下：

- 功能特性（以支付类应用为例）
  - 支持任意数据类型的资产或权益类型（数值型和非数值形）
  - 交易者身份和交易金额隐匿
  - 基于零知识证明算法
  - 不依赖可信硬件执行环境
  - 不依赖可信第三方服务
  - 国密算法和监管支持
- 性能指标
  - 百字节记录大小
  - 微秒级交易处理延时

  - 万级每秒交易并发量

3.2 VCL目标场景业务痛点示例 – 支付中的隐私风险

实习生美丽终于毕业啦！来到新城市工作的美丽决定不时去花店买几束花装饰自己的出租屋。

美丽每次都会在楼下花店买玫瑰和桔梗。买了几次之后，每次美丽准备第二天再去采购的时候，就会收到了花店发来的短信："美丽小姐您好，明早新到一批您喜爱的玫瑰与桔梗，欢迎前来选购"。美丽一边惊喜一边又诧异她怎么知道我的买花习惯，于是去花店的时候和店员聊起这件事。店员自豪地说她们会定期整理花店账本，并对每个顾客的支付记录进行分析，就可以提前"贴心"提醒顾客啦。美丽听完恍然大悟，才知道原来自己日常的买花喜好及行为仿佛已经完全被花店"监控"了。不仅是花店提醒短信，自从买花以来美丽就经常收到花瓶推荐广告、驱虫广告等，美丽心想应该也是花店账目信息出售或泄露导致的吧，心里有种说不上来的滋味，决定利用自己穿越时空的超能力改变这一现状。

美丽之前在花店的支付，泄露了所有支付明细，包括买花类别、数量、消费金额、消费时间等。花店及入侵者通过账本记录，配合大数据分析等手段，分析出了美丽的消费习惯：一般什么时候来买花、喜欢什么花、消费水平等，及由此推测出的其他日常行为，导致了严重的美丽隐私泄露问题。

美丽心想，花店其实只需确定她每次支付完成后，账本记账是正确的就可以了，于是这次美丽选择采用*WeDPR-Lab*提供的方案进行支付。这次，美丽实现了对自己交易内容的完全隐匿，花店记账时再也无法看到美丽具体的支付细节内容，但又可以验证美丽本次的支付是有效且正确的，也能通过将本次消费与历史账目加和来更新得到最新账目，也就是说在保证美丽隐私的同时又能保证正常的记账规则。而且，由于花店账本也没有记录美丽的消费具体内容，所以即便有黑客等入侵，得到的也只是一堆密文，无法获得任何有用的信息。美丽再也不会担心自己被花店监控而收到花店的"关怀"提醒短信，也不用担心自己的消费信息泄露而被其他服务商"热心"推荐了。

在传统支付方案中，隐私保护需要完全依赖于信任一个第三方平台，譬如上述场景中的花店支付系统，但这个第三方平台的安全性又如何保证呢？公开可验证密文账本VCL方案，完全不依赖支付中间服务商的信任，仅依靠密码学困难性理论，通过密码学承诺、场景化高效零知识证明，在遵循账本记账规则的同时，既能保证账本维护者甚至外部监管机构对账本正确性的验证，又能有效保护用户支付明细的隐私安全。

3.3 VCL核心算法组件特性

WeDPR-Lab开源社区版中提供了VCL解决方案的核心算法组件和使用示例，具体包括：

- 公开可验证密文账本的Demo示例
- 明文账本数据向密文凭证转换的通用方法
  - 明文账本数据转换为密文凭证
  - 密文状态下证明密文凭证的所有权
- 密文凭证之间的代数约束关系的零知识证明
  - 密文状态下证明v1 + v2 =? v3
  - 密文状态下证明v1 * v2 =? v3
- 密文凭证的范围约束关系的零知识证明
  - 密文状态下证明v1 > 0
- 其他基础工具代码

3.4 VCL示例快速体验

完成安装步骤之后，进入示例目录，运行公开可验证密文账本Demo。

```
cd WeDPR-Lab-Core/solution/verifiable_confidential_ledger
cargo run
```

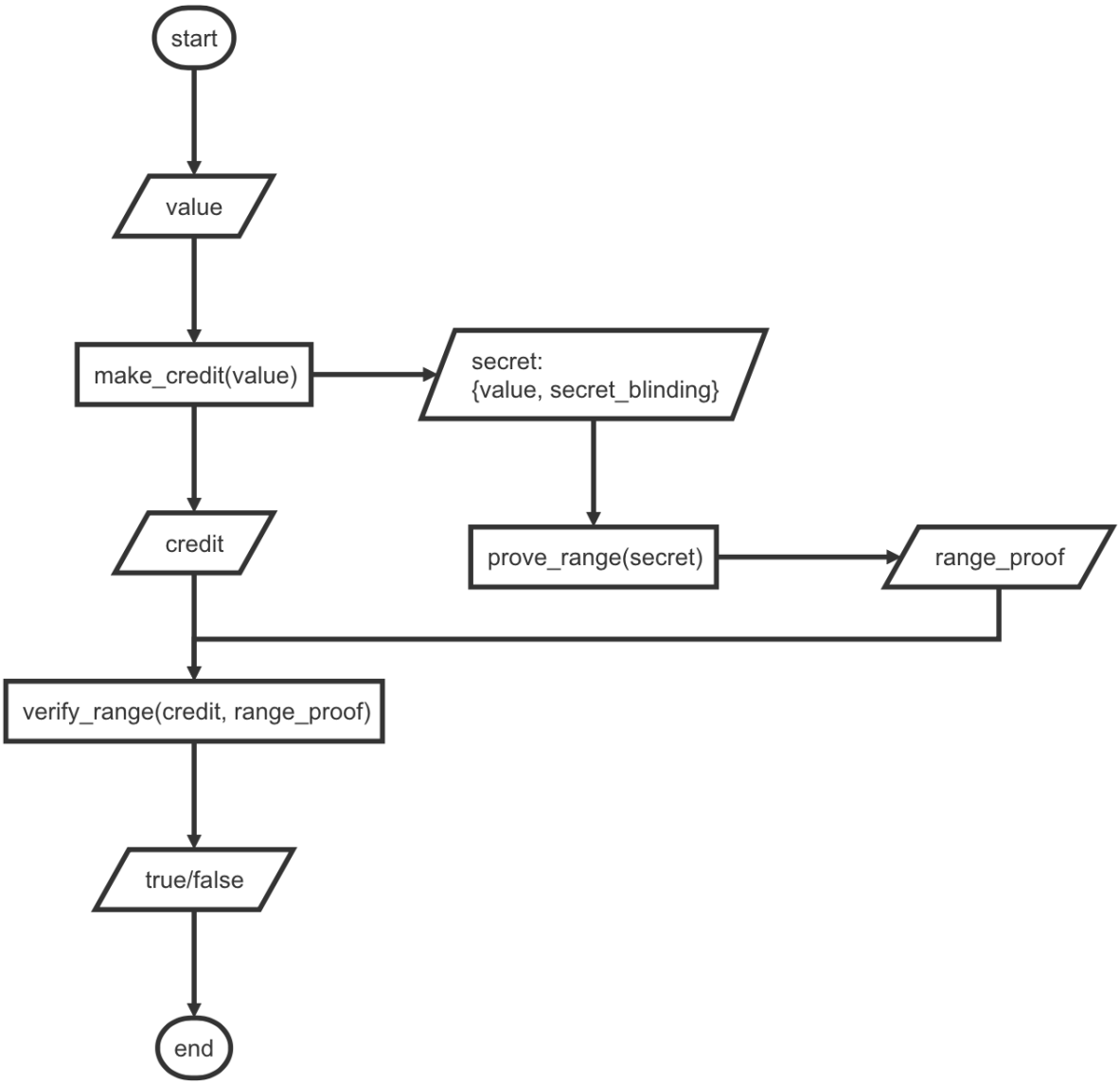
cargo run之后按照demo指引，设置中文或英文作为演示语言，按步骤输入即可进行体验。

具体流程描述如下：

1. 用户对于自己输入的数值金额v1, v2, v3，生成密文凭证c1, c2, c3。
2. 在密文状态下，用户生成三类金额的数值关系的零知识证明，具体包括：
   - 加法关系证明：v1 + v2 = v3；
   - 乘法关系证明：v1 * v2 = v3；
   - 非负数证明：v1 > 0
3. 验证以上三类零知识证明的正确性。
4. 与明文账本对比，体验密文账本数据效验的隐私保护效果。

3.5 VCL示例流程图

范围证明：


start
value
make_credit(value)
secret:
{value, secret_blinding}
credit
prove_range(secret)
range_proof
verify_range(credit, range_proof)
true/false
end

CHAPTER 4

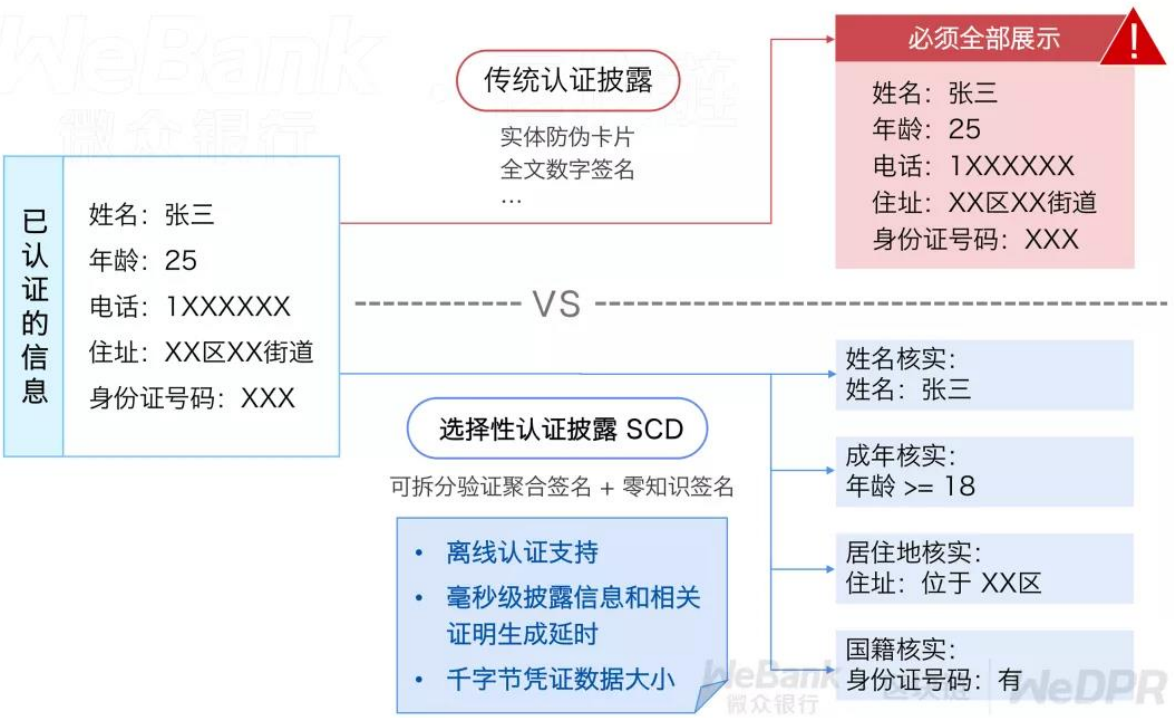
选择性认证披露场景方案

4.1 SCD隐私保护解决方案

在信息时代，信息披露无处不在，无论是个人还是企业，在日常活动中，都需要向不同实体披露经过认证的信息。例如，个人每次入住旅店时，需要出示本人身份证；企业每次参与竞标，需要提供经过相关部门认证的资质证明。

在传统方式中，为了获得此类认证后的凭证，用户往往需要向认证部门提供一系列认证材料，并完成相关认证手续，才能获得由相关部门签发的凭证，交互代价很高，往往只能支持捆绑认证，即多项数据只能一同认证且一同披露明文。

回到入住旅店的场景，虽然旅店要求客人出示身份证的主要目的是验证其为合法且满足年龄要求的公民，但这一看似平常的行为，也泄露了客人额外的敏感信息，如家庭住址、身份证号码等。在类似的企业资质证明中，尽管很多时候，仅需要证明企业特定财务情况满足一定的下限，而在传统的信息披露方式中，却不得不要求企业披露完整的敏感财务信息，由此未能实现隐私数据最小化信息披露，带来了不必要的隐私数据泄露风险。



针对这类场景痛点，WeDPR**选择性认证披露解决方案**，提供了最小化认证数据披露效果，**仅需认证一次**，在随后的信息披露过程中，用户可以自主选择需要披露的证书属性值明文，或者使用更为神奇的零知识断言披露方式，仅需提供“自身证书属性值满足认证条件”的断言证明（例如，证明我的年龄大于20岁，但不告诉对方具体几岁），而不提供任何隐私数据明文信息。

在技术层面，选择性认证披露的核心功能包括：用户可以在不依赖可信第三方服务的前提下，支持已认证证书中**任意类型的属性集合**中**任意子集的隐私数据披露**，并支持**基于属性值的零知识断言披露**，从而实现用户在证明自身具备某些属性值同时，完美保护了隐私属性值的明文和其他不必要披露的隐私属性值。

此外，选择性认证披露还进一步支持多个属性值同时进行零知识断言披露，仅当所有属性值都满足条件时，才能验证通过，当任一条件不满足时，不会泄露其他条件的满足情况，真正意义上实现了认证隐私数据的**最小化信息披露**。

SCD场景式解决方案实现了“属性隐私同时可认证可监管”的业务模型，支持二次开发，能快速满足有“隐私认证身份凭证”需求的业务场景，如私密数字凭证、智慧医疗金融等。SCD场景式解决方案目前已经集成到微众银行自主研发并开源的**WeIdentity 分布式数字身份解决方案**中（WeIdentity github仓库地址；WeIdentity gitee仓库地址）。

4.2 SCD核心算法组件特性

WeDPR-Lab开源社区版中提供了SCD解决方案的核心算法组件和使用示例，具体包括:

- 选择性认证披露的Demo示例
- 选择性披露证书的全流程功能
  - 选择性披露证书的颁发
  - 选择性披露证书的混淆
  - 选择性披露认证的零知识断言证明
  - 选择性披露认证的零知识断言验证
  - 选择性属性明文披露和验证
- 各类数值逻辑关系的零知识断言的证明和验证

  - 相等
  - 大于
  - 小于
  - 大于等于
  - 小于等于
- 其他基础工具代码

4.3 SCD示例快速体验

为了便于用户更易理解选择性认证披露的效果，我们在SCD Demo中设定了如下具体示例应用场景：

用户申请“优秀青年”奖项，该奖项的申请条件为：年龄在[18,40]区间内，且贡献级大于6。根据具体贡献值，奖项又划分为：

- 贡献级 = 10，为 一等奖；
- 贡献级 = 9 ，为 二等奖；
- 贡献级 = 7或8 ，为 三等奖。

奖项授予方通过审核用户提供的个人认证信息，向用户反馈其是否具有该奖项的申请资格及对应的奖项评级。

完成安装步骤之后，进入示例目录，运行：

```
cd WeDPR-Lab-Core/solution/selective_certificate_disclosure
cargo run
```

cargo run之后按照Demo指引，设置中文或英文作为演示语言，按步骤输入即可进行体验。

体验流程描述如下：

1. 用户基于属性模板填写个人凭证信息：年龄及贡献级。
2. 用户请求权威机构进行个人凭证信息认证，获得认证后的凭证。
3. 为防止权威机构对认证凭证的使用进行跟踪，用户对认证后的凭证进行混淆，获得混淆凭证。
4. 用户选择个人信息凭证披露方式（分别描述为以下5、6步骤）。
5. 用户仅提供断言证明，证明满足全部申报条件， 但不透露任何字段值。
   - 用户生成并向奖项授予方提交“自身凭证信息满足奖项申请条件”的断言证明。

   （这种披露方式，奖项授予方无法直接获得用户个人凭证信息的明文，但通过对其断言证明的验证，即可判断该用户是否具有奖项申请资格。）
6. 用户提供贡献级明文信息及其正确性证明， 但不透露年龄。
   - 用户为其贡献级信息生成正确性证明。
   - 用户生成“自身年龄满足奖项的年龄申请条件”的断言证明。
   - 用户向奖项授予方提交贡献级明文、贡献级正确性证明、年龄断言证明。

   （这种披露方式，奖项授予方直接获得用户个人凭证信息中的贡献级明文，从而进行奖项等级评定；但无法直接获得用户的年龄信息，只能通过对其年龄断言证明的验证，判断该用户是否满足奖项的年龄申请条件。）
7. 用户获得奖项授予方验证并返回的奖项资格验证结果。

CHAPTER 5

多方密文决策场景方案

5.1 多方密文决策

投票决策作为用户表达观点和意见的重要方式，在日常生活中应用广泛，如小区公共事务决策、党群事务决策、股东大会表决等。目前也已产生大量电子投票应用，来简化线下纸质化投票、唱票、计票的繁琐流程。

但在传统电子投票中，投票者投出的选票均为明文，可能包含投票者身份信息及其选择的候选人信息，所以整个投票过程的隐私性和正确性都强依赖于计票者的信誉。计票者知道每张选票的选择，有机会篡改选票，甚至公布错误计票结果。

所以传统电子投票应用仍无法有效解决对投票者的隐私保护、对计票者的行为监督以及对计票结果的公开验证。

5.2 ACV隐私保护解决方案

WeDPR多方密文决策（Anonymous Ciphertext Voting，ACV）解决方案，可有效实现：

- 投票者的选票为密文格式，除投票者本人之外，任何人无法获得投票者的选择。
- 将中心式的计票权力分散至多个计票者，只有一定数量的计票者共同参与，才能计票成功。
- 通过零知识证明，实现公众对投票者密文选票、计票者计票过程及计票结果的公开可验证。

其重要功能和指标如下：

- 核心功能特性：（以投票应用为例）
  - 支持多种常用投票规则
  - 投票者身份和投票选择隐匿
  - 投票者可独立验证自己投出选票被正确计入结果
  - 公众可独立验证投票结果正确性
  - 不依赖可信第三方服务或可信硬件执行环境
- 性能指标
  - 百字节投票记录大小

  - 毫秒级投票处理延时
  - 千级每秒投票处理并发量

5.3 ACV核心算法组件特性

WeDPR-Lab开源社区版中提供了ACV解决方案的核心算法组件和使用示例，具体包括：

- 多方密文决策的Demo示例
- 明文决策数值向密文决策凭证转换的通用方法
- 多方联合统计密文决策凭证的通用方法
- 密文决策凭证的格式约束的零知识证明
- 密文决策凭证之间的代数约束关系的零知识证明
  - 密文状态下证明v1 + v2 <= v3
- 密文决策凭证的范围约束关系的零知识证明
  - 密文状态下证明v1 > 0
- 密文决策凭证统计过程中计算约束的零知识证明
  - 密文状态下证明统计所需密钥x = 统计者真实密钥y
- 其他基础工具代码

5.4 ACV示例快速体验

完成安装步骤之后，进入示例目录，运行公开可验证密文账本Demo。

```
cd WeDPR-Lab-Core/solution/anonymous_ciphertext_voting
cargo run
```

cargo run之后按照demo指引，设置中文或英文作为演示语言，按步骤输入即可进行体验。

为了便于用户更易理解多方密文决策ACV的效果，我们设定了如下示例场景。

- 4个投票者为3个候选人进行投票, 每个投票者都可向其中任意一个或多个候选人投出包含一定数值的密文选票，”,
- 3个计票者需合作才能统计出每个候选人的最终得票。

整个demo流程中，用户将体验隐匿密文投票的全过程，具体包括

1. 投票者如何使用密文选票进行匿名投票;
2. 计票者如何联合解密得到计票结果;
3. 任意验证者如何借助零知识证明来验证整个过程中投票者与计票者行为的正确性。

具体流程描述如下：

1. 每个投票者进行身份认证，认证后获得并公布密文空白选票。 其中，密文空白选票代表的数额表示：该投票者可以投出的密文选票总额上限。
2. 投票者选择为每个候选人分别投出的数额。
3. 投票者将确定好的候选人及数额转化为密文选票，并公布密文选票。
4. 投票者生成并公布以下3个零知识证明，分别用于证明：
   - 投给每个候选人的密文选票数额非负;
   - 每个密文选票格式正确（否则会导致后续计票失败）;

- 投票者投出的密文选票数额之和小于等于其初始选票数额。

5. 任意验证者通过步骤1、3、4公布的密文选票与零知识证明，验证每个投票者的密文选票是否正确、合法。
6. 针对每一个候选人，3个计票者联合统计其得票，公布各自的局部统计结果。
7. 每个计票者生成并公布一个零知识证明，用于证明：
   - 计票者公布的计票信息是使用正确的计票者密钥计算而得，而不是随意构造而得。
8. 任意验证者通过计票者在步骤6、7公布的密文统计结果和零知识证明，验证计票者统计过程的正确性。
9. 任意公众都可通过汇集流程6中各计票者的局部统计结果，获得每个候选人的最终得票。

（其中，步骤5、8也可在步骤9之后进行集中验证。）

CHAPTER 6

SDK

WeDPR Lab提供多种SDK调用方式，开发者可在多种平台、多种终端使用WeDPR Lab的密码组件，构建不同的隐私保护应用场景。

6.1 Java SDK

6.1.1 环境依赖

WeDPR Lab Java SDK 依赖如下：

6.1.2 快速体验

下载仓库

```
git clone https://github.com/WeBankBlockchain/WeDPR-Lab-Java-SDK.git && cd ./WeDPR-
↪Lab-Java-SDK
```

或

```
git clone https://gitee.com/WeBankBlockchain/WeDPR-Lab-Java-SDK.git && cd ./WeDPR-
↪Lab-Java-SDK
```

根据操作系统访问**release**页面获取对应动态库，以**linux**为例，支持**mac**、**linux**和**windows**版本

```
curl -LO https://gitee.com/WeBankBlockchain/WeDPR-Lab-Core/releases/v1.2.0-Java-
↪SDK/libffi_java_crypto.so
curl -LO https://gitee.com/WeBankBlockchain/WeDPR-Lab-Core/releases/v1.2.0-Java-
↪SDK/libffi_java_vcl.so
curl -LO https://gitee.com/WeBankBlockchain/WeDPR-Lab-Core/releases/v1.2.0-Java-
↪SDK/libffi_java_scd_1_1.so
```

拷贝动态库至加载路径

```
cp ./*.so ./demo/src/main/resources/WeDPR_dynamic_lib
```

编译项目

```
bash ./gradlew clean build
```

进入项目目录

```
cd demo/dist
```

运行demo

```
java -cp "apps/*:conf/:libs/*" com.webank.wedpr.demo.DemoMain
```

FAQ

当提示The WeDPR dynamic library was not found时，请检查并下载对应操作系统的动态库防止于WeDPR_dynamic_lib文件夹下

当系统openssl版本为1.0.x时，scd动态库为1_0版本，当系统openssl版本为1.1.x时，scd动态库为1_1版本

6.2 Android SDK

6.2.1 环境依赖

WeDPR Lab Android SDK 依赖如下：

6.2.2 快速体验

编译项目前，需要准备好 Android Studio、并安装NDK 。

- 下载仓库

```
git clone https://github.com/WeBankBlockchain/WeDPR-Lab-Android-SDK.git && cd ./
↪WeDPR-Lab-Android-SDK
```

或

```
git clone https://gitee.com/WeBankBlockchain/WeDPR-Lab-Android-SDK.git && cd ./
↪WeDPR-Lab-Android-SDK
```

- 获取动态库库：根据需求访问依赖库地址页面下载对应版本的动态库
- 拷贝动态库至加载路径：将解压好的动态库放至demo/app/src/main/jniLibs对应路径下
- 查看demo：查看MainActivity.java

6.3 iOS SDK

6.3.1 环境依赖

WeDPR Lab iOS SDK 依赖如下：

6.3.2 快速体验

编译项目前，需要准备好 xcode、安装cocoapods，并注册ios开发者账号 。

- 下载仓库

```
git clone https://github.com/WeBankBlockchain/WeDPR-Lab-iOS-SDK.git && cd ./WeDPR-
↪Lab-iOS-SDK
```

或

```
git clone https://gitee.com/WeBankBlockchain/WeDPR-Lab-iOS-SDK.git && cd ./WeDPR-
↪Lab-iOS-SDK
```

- 获取静态库：访问依赖库地址页面下载对应版本的动态库
- 添加静态库至xcode：将libffi_c_crypto.a与libffi_c_vcl.a添加至xcode
- 查看demo：查看ViewController.m了解demo调用页面

CHAPTER 7

接口文档

VCL接口说明文档：点击这里

SCD接口说明文档：点击这里

密码模块接口说明文档：点击这里

CHAPTER 8

常见问题解答

- 问：**WeDPR安全吗?**
- 答：WeDPR依托区块链等分布式可信智能账本技术，融合零知识证明、安全多方计算、同态加密等密码学前沿技术，为数据安全和数据隐私提供技术保障。WeDPR-Lab系列核心算法组件代码完全公开透明，安全性可由开源社区和合作伙伴公开检验。

- 问：**WeDPR符不符合国密?**
- 答：方案设计完全符合国密算法标准，可以自由替换适配国密算法接口，支持国密相关的证书体系、软硬件安全产品的无缝扩展对接。

- 问：**WeDPR和区块链有什么关系?**
- 答：WeDPR的核心理念是提供场景优化的高效隐私保护解决方案，其中重要的目标场景之一就是弱信任模型下的分布式多方协作场景，这与区块链技术的定位非常契合。WeDPR相关方案也结合了区块链的特性进行设计，解决了对应区块链应用中的隐私痛点，两者联合实现了 1+1 > 2 的效果。

- 问：**有技术问题需要咨询?**
- 答：如在实操方面遇到阻碍或想和开发者们随时随地交流，欢迎关注【微众银行区块链】公众号，回复【小助手】进群。

CHAPTER 9

隐私保护科普专题

- 第1论：隐私和效用不可兼得？隐私保护开辟商业新境地
- 第2论：隐私合规风险知几何？数据合规商用需过九重关
- 第3论：密码学技术何以为信？深究背后的计算困难性理论
- 第4论：密码学技术如何选型？初探理论能力边界的安全模型
- 第5论：密码学技术如何选型？再探工程能力边界的安全模型
- 第6论：密码学技术如何选型？终探量子计算通信的安全模型
- 第7论：密码密钥傻傻分不清？认识密码学中的最高机密
- 第8论：密钥繁多难记难管理？认识高效密钥管理体系
- 第9论：密码学原语如何应用？解析单向哈希的妙用
- 第10论：密码学原语如何应用？解析密码学特有的数据编解码
- 第11论：密码学原语如何应用？解析密码学承诺的妙用
- 第12论：密码学原语如何应用？解析密文同态性的妙用
- 第13论：密码学原语如何应用？走近门限密码算法
- 第14论：硬件化方案坚不可摧？揭秘可信硬件TEE的是非功过
- 第15论：数字化契约如何守护？密码学数字签名共性解析
- 第16论：数字化契约如何守护？解析群/环签名的妙用
- 第17论：数字化契约如何守护？解析多方门限签名的妙用
- 第18论：数字化契约如何守护？解析盲签名的妙用
- 第19论：数字化契约如何守护？解析聚合签名的妙用