本文档旨在从Hyperledger Fabric的角度，对中国人民银行发布的《金融分布式账本技术安全规范》文件进行梳理和解读，由Hyperledger中国工作组志愿者共同编写和审阅。

会议信息：

分工：（请在下方表格中填写个人信息，认领任务。若同一章节有多人认领，请自行沟通协调，进一步分工或协作共同完成）

|  |  |
| --- | --- |
| 章节 | 姓名+公司(可多人） |
| 6 基础硬件 | 吴旭(众享金联)，刘宇翔(mediconcen) |
| 7 基础软件 | 吴旭(众享金联)，王海龙（京东数科） |
| 8 密码算法 | 朱建伟(小米金融)，张保佳(小米金融) |
| 9 节点通讯 | 朱建伟(小米金融)，张保佳(小米金融) |
| 10 账本数据 | 朱建伟(小米金融)，张保佳(小米金融) |
| 11 共识协议 | 郭剑南(树根互联) |
| 12 智能合约 | 刘宇翔(Mediconcen)，江志坤(京东数科) |
| 13 身份管理 | 曹龙(京东数科)，刘宇翔(Mediconcen) |
| 14 隐私保护 | 刘宇翔(Mediconcen) |
| 15 监管支撑 | 吴旭(众享金联) |
| 16 运维要求 | 梁志超(众享金联)，吴旭(众享金联) |
| 17 治理机制 | 郭剑南(树根互联)，李天学(北理新源) |

* **原文**：《金融分布式账本技术安全规范》条例原文
* **评论**：分析Fabric对于对应条例的支持，可分为三种情形
  + 不满足：Fabric目前的架构和实现**很难满足**该条例的要求，应予以注意
  + 可满足：服务提供商需要通过**额外的增值服务、业务逻辑、软/硬件设施、或少量修改Fabric源码**，满足条例要求
  + 满足：Fabric**已经满足**该条例要求
  + 不适用：与Fabric无直接关系，或Fabric不满足条例的前提要求
  + 不清晰：条例中所作要求不够清晰，不足以做出判断
  + 定义：该条例仅陈述定义，未做具体要求
* **相关措施**：为了满足该条例要求，Fabric开发者，应用开发者，以及Fabric提供商应当采取的措施，以及需要注意的事项

|  |  |  |
| --- | --- | --- |
| **原文** | **评论** | **相关措施，备注** |
| 6 基础硬件 | | |
| 6.2.1 场地安全  部署的物理数据中心及附属设施符合以下要求：  ——对于云端部署模式，应保证用于金融行业数据中心运行环境位于高安全区域；  ——对于承担共识节点或记账节点功能的系统节点，宜保证金融分布式账本使用者业务运行、数据存储和处理的物理设备位于中国境内。 | 可满足  fabric对部署地点或数据中心无硬性要求，由使用者自行决定 |  |
| 6.2.2 硬件设备  应对设备运行状态、资源使用情况等进行监控，能在发生异常情况时发出告警。  应保证设备和存储介质在重用、报废或更换时，能对其承载的数据进行清除且不可恢复。  应保证不同节点使用的硬件设备具备一定的异构性。  对于云端部署模式，应在云端环境服务方的配合下，保证云端环境具备一定的异构性。 | 可满足  监控设备的运行状态与资源使用情况属于常规的设备监控，数据的清除依赖于存储介质，均不归属于fabric范畴。  Fabric并无特定硬件平台限制，官方亦发布有amd64，s390x架构适用的容器镜像  事实上，Fabric已经被所有主流云平台的区块链服务使用，证明了其满足硬件部署异构性 | 对于Fabric服务提供商，应满足条例中关于设备和存储介质的要求 |
| 6.2.3 节点部署安全  应保证关键节点冗余部署，保证系统可用性。  应避免将所有承担共识或记账的节点部署在同一机房内，应能在单一机房节点不可用时保证系统整体的可用性。  应保证将带有不宜共享数据的分布式账本节点放置于机构内部或受保护区域。  应保证部署节点的硬件设备存储容量可扩展，避免因数据容量达到上限而无法同步账本。 | 可满足  对于Fabric系统来说，关键节点应包括orderer以及在通道中组织所对应的锚节点。  kafka模式下一般都建议kafka集群位于同一数据中心，此种场景下会违背单一机房节点不可用时的系统整体可用性，因此不推荐使用，当前宜使用RAFT排序机制  peer节点与orderer节点均保存账本数据，可采用私有数据有限分发机制控制不宜共享数据的存储位置  fabric支持已有账本的重加载启动，因此可支持存储容量的扩展。 | 对于Fabric服务提供商，应满足条例中关于节点容灾，存储可拓展的要求，并在部署方案设计中结合业务保护限制敏感数据分布区域 |
| 6.2.4 硬件加密设备安全  对于使用硬件加密设备完成密码运算和密钥存储的分布式账本系统，所用硬件加密设备应满足如下要求：  ——使用的加密机设备应符合国家密码管理部门颁布的GM/T 0045—2016的要求；  ——使用的个人密码设备（如UKey、加密卡、带SE或TEE的移动终端等）应符合行业主管部门和国家密码管理部门的要求。 | 不满足  fabric支持的pkcs11不满足此种要求 |  |
| 6.3.1 网络架构安全  应保证共识节点或记账节点之间能直接进行网络通信或能间接进行消息传递。  在网络拓扑中，应防止单个节点故障而形成网络隔离。  应保证每个重要节点具有较大的局部聚集系数。 | 可满足  拓扑中，一个顶点的局部聚集系数等于所有与它相连的顶点之间所连的边的数量，除以这些顶点之间可以连出的最大边数。这意味着重要节点周围的节点应尽可能两两互通 | 对于依赖于节点发现服务与私有数据特性的分布式账本系统，需额外关注锚节点的的容灾设计 |
| 6.3.2 通信传输安全  应在参与分布式账本的节点之间建立安全传输通道，保证数据传输的完整性和不可篡改性。  应对数据和信息采取相应的防护措施，保证其能抵抗篡改、重放等主动或被动攻击。  应采用密码技术保证节点间通信过程中敏感信息字段或整个报文的保密性，应确保信息在存储、传输过程中不被非授权用户读取和篡改。  可采用有权限的网络访问控制，在参与分布式账本节点之间构建虚拟专用网络（VPN），降低网络攻击造成的危害。 | 可满足  Fabric的读写集机制能发现并报告节点账本遭受的篡改，若要从篡改中恢复账本，需要预先准备用于账本重放的备份。  fabric中节点之间可通过tls机制建立安全传输通道，经过密钥协商机制后，后续传输的报文都会被加密，从而保证了完整性与不可篡改性，同时非授权用户虽然有可能获取密文信息，但无法解密读取明文或篡改。  fabric中的交易都有一个唯一的标识txid，如果出现重放攻击，txid校验会发现该笔交易已经存在于账本中，重放攻击不会成功。但由于Fabric机制中commit阶段发生在排序阶段之后，在一个区块间隔内发生的重放攻击的交易会进入区块，消耗系统的资源，达到重放攻击的目的。  基础网络属于vpn或公网，对fabric没有影响。  总结如下，如果可以突破fabric的tls连接，发起重放攻击，不会引起账本中的错误结果，但会消耗计算和存储资源。 | Fabric本身不能抵御所有通信阶段的攻击，所以在服务提供商在设计架构时需要引入额外的安全假设和手段来保证 |
| 7 基础软件 | | |
| 7.2 账本结构  账本结构应具有防篡改性。账本结构宜使用块链式或近似块链式的存储结构，应使用哈希嵌套保证数据难以被篡改。  账本应具有数据校验功能。任何一条记录被非法篡改后都可通过历史账本数据回溯以快速检验出。 | 满足  Fabric的区块数据逻辑上采用块链式结构，同时使用哈希嵌套，具备防篡改性。每个区块都有一个哈希值。  可以检验,但是无法快速检验 |  |
| 7.3 共识模块  共识模块应能协调各系统参与方有序参与数据打包和共识过程，并保证各参与方的数据一致性。  系统无故障节点或欺诈节点时，应能在规定时间内达成一致的、正确的共识，输出正确结果。  在故障节点和欺诈节点的总数量不超过理论值的情况下，系统应能正常工作。 | 满足  fabric现阶段提供solo、kafka和raft三种共识机制，solo不建议用于生产环境，暂不讨论。kafka和raft均能保证所有数据的有序性，保证各参与方获得的数据一致性。  对于故障节点，kafka与raft均能保证在故障节点数目不超过理论值时，系统能正常工作。  对于欺诈节点，kafka与raft均属于CFT类共识协议，需结合peer节点的记账以及背书来抑制欺诈节点的交易。  fabric中欺诈节点不能单独地理解为peer节点，需结合背书策略来定义，即欺诈节点属于背书策略组织中的peer节点，在此定义下，如果欺诈节点发起的交易满足了背书条件（即超过了理论值，类似于达到51%攻击），则该欺诈行为将会成功；若不能满足背书条件，则该欺诈行为将会失败。非背书策略组织中的peer节点即使做出欺诈行为，fabric网络在校验交易时会自动忽略该行为，对交易的最终结果无影响。因此，在欺诈节点的总数量不超过理论值的情况下，fabric系统能正常工作。 |  |
| 7.4 分布式组网  系统参与方节点应在物理部署上进行分离，各节点基于网络通信协议和对等网络进行通信和数据互换。  各节点应独立存储具有一致性的账本数据，且保证任意单个节点故障都不影响整个系统的正常工作。  系统由分布在不同地点的节点互连而成，网络中可无中心节点。通信控制功能应分布在各节点上，且任一节点均至少与其他两个节点建立通信连接。 | 满足  需注意的是kafka共识模式中需规划kafka的数目以满足高可用特性，避免出现单点故障。  private data不属于账本数据，不会存储在账本上；但其hash值会写入账本，属于账本数据，同时保证了账本数据的一致性。 | Fabric服务提供商应留意实现条例  任一节点均至少与其他两个节点建立通信连接。 |
| 7.5 数据存储  账本数据应根据数据对象的类别独立存储，账户数据、交易数据、配置数据以及账本元数据等，应分别存储、分别管理、分别操作。  敏感信息应加密存储，并应有数据访问等权限的控制和管理。  节点CA证书及其私钥的存储应私密管理。  数据存储可选用结构化数据库、非结构化数据库或混合选用。数据库应选用安全高效并经过检验的主流稳定版本。 | 不清晰  Fabric中，交易数据、配置数据都是分别存储、分别管理、分别操作的。ledgerID是存在idStore。  疑问点：交易数据与配置数据都保存在区块中，但这两种类型的区块的地位是平等，可位于前后相互链接，这样是否可以说是分别存储和管理？  Fabric中节点CA证书及其私钥默认是明文存储的。  Fabric中交易数据采用文件系统存储。状态数据支持leveldb，couchdb。但Fabric较容易支持结构化数据库、非结构化数据库或混合。只需要实现相应的接口。  对于敏感信息应加密存储，fabric框架未自动实现此功能，信息的加密交由用户自行操作，可借鉴Fabric中基于证书属性的权限控制链码案例 |  |
| 7.6 智能合约  智能合约宜在可信的软件/硬件支持的环境中执行。智能合约代码存储和运行时，系统应具备相应的安全保护能力，不应允许未授权实体明文读取合约代码和状态。  智能合约应具备数据前向兼容的能力，版本迭代时，旧版本的合约应及时停用，并存档数据，新版本合约应能调用历史数据。  智能合约的运行机制宜有前向兼容的能力，当系统版本升级后，智能合约应能正常执行。  系统应通过有效的智能合约审核以确保合约代码所表达的逻辑无漏洞。智能合约的发布应引入相关方联合审核机制，审核流程应高效、严谨。 | 可满足  Fabric 2.0中引入了在系统外执行智能合约的机制，容易满足可信执行环境的要求。使用该方法也可避免明文存储源码。  Fabric中智能合约升级后，账本数据会沿用旧版本，高版本的合约依然能够通过shim接口读取账本数据和历史。  Fabric在大版本内保证向下兼容，Peer升级后依然能够保证旧版的智能合约持续运行。  Fabric 2.0的去中心化智能合约管理方式，有效得避免了单一方擅自部署升级合约以及背书策略，可以看做多方审核的机制，能够很好得支持相关方联合审核。 | Fabric智能合约开发项目应满足：  系统应通过有效的智能合约审核以确保合约代码所表达的逻辑无漏洞。 |
| 7.7 接口设计  应设计良好的接口，隐藏底层账本的细节，为应用层提供简洁的调用方法。  接口的设计原则应简洁明了，提供完整的功能，能完成交易和维护分布式账本数据，并且有完善的权限管理机制。  接口设计应考虑扩展性和兼容性。 | 满足  fabric官方提供了node.js和java两种sdk，同时支持golang、python等其它语言的sdk，为应用层提供了多种调用方法。  Fabric的关键节点peer和orderer分别使用grpc接口对外通信，并提供所有的通信消息体与方法。  Fabric通过通道ACL配置和链码来基于MSP进行权限管理  此外可使用tls证书和交易证书从接入连接到交易发起两阶段保证权限控制。 |  |
| 7.8 数据传输  传输数据过程中，应使用对称或非对称国密算法对数据进行加密，防止数据在传输过程中被窃取。 | 不满足  原生的fabric不支持国密算法。但可对Fabric的密码模块做拓展支持国密。有一定的改造难度。 |  |
| 7.9 时间同步  应保证节点之间的时间戳误差维持在共识协议允许的范围内。  可使用经过认证的中心化时间同步源进行节点间的时间同步。 | 可满足  fabric会校验接受到交易与节点自身的时间是否存在误差，若超过一定阈值，则会直接丢弃该交易。  若提供中心化时间同步源，所有节点与此同步源同步，则能保证全网络节点时间的一致性。此功能需借助于第三方同步源，不属于fabric服务。 | 应额外提供时间同步服务 |
| 7.10 操作系统  系统宜有针对不同操作系统的软件版本，宜支持三种及以上的操作系统或系统版本。 | 满足  Fabric发布版本支持多种操作系统，如centos、ubuntu、Redhat enterrise linux等。 | 无 |
| 8 密码算法 | | |
| 8.1 基本要求  分布式账本系统中的密码算法主要用于数据安全，即保护数据的保密性、完整性、真实性和不可否  认性，包括分组密码算法、流密码算法、非对称密码算法、密钥交换算法、密码杂凑算法和标识密码算  法等。分布式账本系统所使用的具体密码算法应符合GB/T 32905—2016、GB/T 32907—2016、GB/T 32918  —2016等相关国家标准以及GM/T 0006—2012、GM/T 0009—2012、GM/T 0010—2012、GM/T 0015—2012、  GM/T 0044—2016等相关行业标准。分布式账本系统应使用符合GB/T 37092—2018等相关国家标准以及  GM/T 0028—2014、GM/T 0039—2015等相关行业标准的密码模块进行密码算法运算和密钥存储。 | 不满足  Fabric中采用的密码算法是椭圆曲线(ecdsa)算法,并不是国密算法  通过硬件来进行密钥存储，Fabric只支持基于pkcs11接口的HSM | 现在的fabric修改加密算法难度还是太高，可以将密码算法这部分插件化。不必强制编译成一体。  软件实现在ssl层的某些地方是允许的,某大学申请的时候是允许的这个情况 |
| 8.2 保密性  保密性指信息不被泄露给非授权的用户和进程等实体的一种性质。  保密性通过密码加密功能实现，其算法包括对称密码算法和非对称密码算法。  通信双方在交换敏感信息时，应在建立连接之前，使用密码技术进行会话初始化，通过密钥交换算  法协商会话密钥。在通信过程中，应使用会话密钥对敏感信息或整个报文进行加密，并在加密时采取随  机数填充等技术，避免相同的明文数据在加密后生成相同的密文。  在存储敏感的业务数据、身份鉴别数据和密钥数据之前，应采用密码技术进行加密。 | 满足  在通信的时候，fabric可以设置使用TLS来保证数据通信过程的安全性。  ssl使用的证书的加密算法是椭圆曲线加密生成，椭圆曲线加密算法是抗cpa攻击的算法，所以不用担心选择明文攻击的问题，另一方面每次协商的密钥是随机生成的，也可以保证同样的明文得到不同的密文  通过配置pkcs11，钱包来处理存储敏感的业务数据、身份鉴别数据和密钥数据 | 部分数据可以是信用cpa的加密算法进行加密处理 |
| 8.3 完整性  完整性指数据没有受到未授权的更改，分布式账本中的完整性应用场景包括业务数据和密钥的完整  性保护。  应保障关键数据在传输和存储中的完整性，并在对数据处理前检验其完整性。  数据完整性可通过消息鉴别码（MAC）或数字签名保障。 | 满足  Fabric中的msp机制对整个链上数据的读，写，管理都进行了严格的限制。  数据整个区块的正确性也有区块hash来保证，并且交易上也有交易提交者的私钥签名。满足需求 | 无 |
| 8.4 真实性  真实性指一个实体是其所声称实体的特性。  应使用非对称加密、动态口令或数字签名等方式保障真实性。 | 可满足  真实性的验证需要依赖外部的人员审核等操作。fabric上面使用的都是数字身份，并没有办法保证人员身份的鉴别。但是在人工审核完身份之后，fabric上的数字身份可以进行身份识别。 | 无 |
| 分布式账本中真实性的应用场景包括：  ——进入重要物理区域人员的身份鉴别；  ——节点通讯双方的身份鉴别；  ——网络设备接入时的身份鉴别；  ——登录操作系统和数据库系统的用户身份鉴别；  ——应用系统的用户身份鉴别。 | 不适用  这一部分需要做的工作是需要在fabric之外额外再建立一套用户权限体系 | 人员控制,设备控制,这部分依赖外围的相关设施 |
| 8.5 不可否认性  可使用数字签名等密码技术生成可靠的电子签名来保障实体行为的不可否认性，系统中所需的具有  不可否认性的行为包括发送、接收、审批、创建、修改、删除、添加和配置等操作。  不可否认性的应用场景包括：  ——实体行为的确认；  ——背书方对实体行为的背书。 | 满足  私钥，证书和实体是一一对应的。没有办法否认。  背书方的背书也是需要签名来进行背书的，并且peer在进行commit的时候会验证背书节点的签名，所以已经保证了不可否认性。 | 无 |
| 8.6 随机性  密码算法执行过程中需要使用随机数时，应按照国家密码管理部门的要求生成随机序列，并符合  GB/T 32915—2016对随机性的要求。 | 不满足  fabric使用的随机数生成算法是crypto/rand,不满足国密标准 | 无  软件实现的随机数是不被承认的 |
| 8.7 密钥管理  密钥管理包括对密钥的生成、存储、分发、导入、导出、使用、备份、恢复、归档与销毁等环节进  行管理和策略制定的全过程。分布式账本系统应根据信息系统等级保护等级，满足GM/T 0054—2018中  对密钥管理的要求。  可通过秘密共享算法将密钥分解为多个子密钥分别存储或传输。 | 可满足  密钥管理本身并不是fabric的工作 | 服务提供商应提供完整的密钥管理服务 |
| 9 节点通讯 | | |
| 9.1 基本要求  分布式账本系统采取节点授权准入的原则，在节点通信过程中应保证数据的完整性、保密性。 | 满足  Fabric是联盟链，现有的模块已经实现msp的功能，可以保证节点是准入的。数据完整性已经通过区块hash来保证。  节点之间通信使用的是grpcs，满足保密性 |  |
| 9.2 节点身份验证  应使用符合第13章“身份管理”中要求的身份认证机制控制节点的接入。  采用密码技术对节点通信双方的身份进行验证。 | 可满足  Fabric在接入的时候需要持有对应的私钥和证书才可以接入。只有验证过并且拿到证书的相关组织才可以加入进来。 | 无 |
| 9.3 通信完整性  使用符合国家密码标准的消息鉴别码算法、数字签名等密码技术来提供通信中数据的完整性保护和校验。 | 不满足  需要进行国密支持。 |  |
| 节点间通信协议应具备应对通信延时、中断等情况的处理机制。 | 满足  节点间通信使用的是gossip协议，具备应对延时中断等情况。 |  |
| 当检测到数据的完整性遭到破坏时，接收节点可以采取措施从发送节点处重新获取数据。 | 满足  Peer节点连接到多个节点，数据不完整的时候，可以重置当前节点的数据，然后让peer节点与其他的节点重新同步来重新获取完整的数据。 |  |
| 9.4 通信保密性  在通信节点建立连接之前，应使用符合国家密码标准的密钥交换技术来产生双方共享的工作密钥，  并进行双向身份认证，确保通信节点是信息的真实授权方。 | 不满足  共享工作密钥的算法不是国密，这一部分需要改造。  需要进行国密改造。 |  |
| 通信节点应使用工作密钥对通信过程中的整个报文或会话进行加密处理。 | 满足  通信过程中使用工作密钥加密通过TLS实现。 |  |
| 应使用符合国家密码标准的技术来建立安全通信通道，避免因传输协议受到攻击而出现的保密性破  坏。 | 不满足 |  |
| 10 账本数据 | | |
| 10.1 完整性  应保证账本数据的生成、传输、存储、调用等操作不可被非授权方式更改或破坏。 | 满足  已经实现。fabric在用户进行修改的时候会对用户身份进行验证，来确保有权限的人才可以进行修改。 | 无 |
| 10.2 一致性  分布式账本中记账节点的账本数据应保持一致。对账本数据的写入和修改，须经各节点达成共识，  以确保各节点的数据一致性。当出现数据分叉时，应存在可用规则进行数据选择。 | 满足  Fabric底层的共识协议应使用raft，由raft来保证数据一致性。Fabric理论上不会出现分叉，除非触发了Bug |  |
| 10.3 保密性  应采用密码技术保证账本数据中的敏感数据在传输和存储过程中的保密性。  账本数据中敏感数据的保护密钥和账本数据本身应分开保存，并且保护密钥应支持存放在安全的密  码模块中。 | 可满足  敏感数据加密这个可以在智能合约里面实现，也可以由sdk进行加密处理。 | 服务提供商应实现：  账本数据中敏感数据的保护密钥和账本数据本身应分开保存，并且保护密钥应支持存放在安全的密码模块中。  保密是peer与peer之间的保密，pvdata，如果是让所有的数据保密，就是数据加密之后再上链  密钥存放在安全模块中，需要额外的供应商支持 |
| 10.4 有效性  各记账节点的账本数据应符合第 11 章“共识协议”中要求的共识协议保证账本数据的有效性，且满足以下  要求：  ——应能对节点存储的账本数据的有效性进行校验；  ——当某个节点的账本数据失效时，应使用符合第 11 章“共识协议”中要求共识协议保证账本数据的有效性。 | 不清晰   第十一章没有找到关于有效性的描述，  节点在进行账本同步的时候会验证区块的hash以及来保证数据的正确性，完整性。  当节点丢失数据的时候，可以与其他的节点通过gossip协议进行数据同步，来获取到有效的账本。 |  |
| 10.5 账本数据冗余  应保证账本数据在系统中具有冗余性，防止因单个节点失效而造成总账本数据的丢失。 | 满足  区块链自身具有多个peer来保证数据备份，已经完成该功能。 | 无 |
| 10.6 访问与使用  分布式账本应确保账本数据不被未授权的第三方获取，数据访问和操作应符合第 14 章“隐私保  护”中对认证授权、访问控制等方面的技术要求。 | 满足  只有加入了fabric网络的组织才可以访问fabric网络上的数据。未获得ca和未授权的组织无法访问orderer与peer，也就不存在非法访问。  这一部分也严重依赖密钥和证书的管理。 |  |
| 10.7 安全审计  记账节点对账本数据的操作应满足以下安全审计要求：  ——账本数据的访问应提供安全审计功能，审计记录包括访问的日期、时间、用户标识、数据内容  等审计相关信息； | 可满足 | 应注意在账本数据中记录用户标识  等审计相关的信息需要再讨论以下  参考auditd for Linux  外部传递进来的区块链上的时间可能是有危险的，可能一开始传递进来的就是不正确的时间，可能不满足,获取时间，一起写入账本  可以在账本上外挂一个auditd的来保证  系统级别的审计是否满足要求  日志保存下来用作审计，但是保存时间不会特别长 |
| ——数据变更应提供审计功能，审计记录不仅包括数据变更成功的记录，还应包括数据变更失败的  记录； | 满足  节点会对orderer传递过来的数据进行有效性校验，校验通过的才可以对底层db产生影响，校验不通过的交易会记录在区块上，并标记上区块校验不通过的原因,可以在区块的metadata的第三个字段看到每一个交易的校验信息，不同的编码代表不同的校验结果。 | fabric的日志如果包括在审计里面，那就是满足的 |
| ——节点有效性校验失败、一致性校验失败等情况下同步账本数据，应提供安全审计功能，审计记  录包括事件类型、原因、账本数据同步的节点、账本数据校验值等审计相关信息；  ——审计记录可由记账节点自行记录，不必写入账本。 | 可满足  支持一系列事件类型和原因，其他信息需要额外添加，挂载log出来即可 | 服务提供商应实现：  节点有效性校验失败、一致性校验失败等情况下同步账本数据，应提供安全审计功能，审计记  录包括事件类型、原因、账本数据同步的节点、账本数据校验值等审计相关信息； |
| 11 共识协议 | | |
| 11.1 基本要求  应根据业务特点选用适宜的共识协议，包括但不限于工作量证明、权益证明、授权股权证明、拜占庭容错等，应满足不同共识协议安全运行所必需的前提要求，且业务激励规则和技术运维安全上的机制设计应保障其自身安全。 | 满足  Fabric目前使用Execute-Order-Commit三阶段提交，基本满足拜占庭容错的要求，同时能够通过背书策略满足企业关于交易授权的要求 | 拜占庭这里如果是peer节点，不会产生影响，orderer是拜占庭节点影响不是特别大，会损害liveness |
| 11.2 合法性  应确保参与共识过程的节点经过验证，保证节点共识过程的加入和退出的合法性，以及节点ID与节  点实体的一一对应，以形成可信节点。 | 满足  Fabric网络中的所有节点均有自己的证书，用于唯一标识节点的身份。网络成员使用该证书确认节点实体。 | 退出是通过证书吊销来实现的  节点的公钥私钥来实现 |
| 11.3 正确性  共识协议依据的算法理论应公开或经过安全评估，如有修改应经过同行评议。  协议算法的测试应全面完整，宜应用形式化验证或进行代码审计以确保算法实现的正确性。  可信节点应为协议算法的运行提供安全可信的硬件软件(如服务器、操作系统等)基础，确保协议算法运行环境的安全性与可靠性。 | 可满足  Fabric的共识协议以论文形式发表在国际会议中，代码实现开源，得到多家机构的使用和验证。  Fabric的测试系统完成，覆盖率高，大的版本经过第三方安全机构审计。目前并未通过形式化验证。 | 可信执行环境需要服务供应商根据响应的规定，采用符合要求的硬件和软件  第一句话满足,第二句话,fabric的测试是比较完整的，fabric做过安全审计，没有做过形式化验证，第三句话不算作fabric自身要求，可以在fabric以外的系统来满足 |
| 11.4 终局性  算法应在可接受的有限时间内具有终局性。  所有参与共识的可信节点，经过一段可接受时间内的交互，应最终达成一致性结果。 | 不满足  Fabric不会因为作恶节点而停止交易的进行  如果说orderer被控制了，会失去终局性。  peer被控制了，还是有终局性 |  |
| 11.5 一致性  所有参与共识的可信节点得到的计算结果应是相同的，且符合共识协议。 | 满足  在容错范围内，所有节点会稳定得达到相同的状态 | 可信节点的问题，有待讨论，现在先把可信节点当作普通的节点看待  即使不考虑可信节点也是有一致性保证的，三阶段提交可以保证一致性 |
| 11.6 不可伪造性  系统中恶意节点占比不超过共识协议容错率时(如采用工作量证明时该比例约为1/2)，任何对系统当前状态进行恶意构造以欺骗其他可信节点所需要的时间，应不少于可接受范围。 | 满足  Fabric主要依赖于非对称密钥对区块进行签名，恶意构造数据需破解相关签名 | 恶意peer没有办法欺骗其他的peer，至于提到的恶意构造，更多的是pow的等算法 |
| 11.7 可用性  协议应具备抗DDoS攻击、处理恶意报文、识别恶意节点的能力，且应采取不转发、拒绝连接、黑名  单等措施缩小影响，使系统获得一定的主动防御能力，提高系统的可用性。  系统能始终在正常时间内对客户端的请求进行响应。 | 可满足  Fabric本身能够识别损坏或伪造的签名，不满足策略的交易等恶意行为，但无法自动拒绝连接，或维护黑名单。 | 需要在业务层面，根据响应的日志、事件和报警，作出反应，例如调整ACL，切断连接等等  Fabric现在的部分接口有流量限制，也可以通过外部系统实现流量控制  可以识别攻击者，也可以通过外部手段来识别合法的用户的恶意交易  Revoked list里面可能是失效名单也可能是黑名单  这里先写可满足，肯定需要外部系统的 |
| 11.8 健壮性  数据在遭受恶意攻击后被污染时，被攻击节点应通过与系统中其他可信节点交互等方式来检测出攻  击及数据污染。  系统中的节点如遇到网络故障等情况与系统断开连接，可能会出现与系统中其他节点状态不一致的  情况。在恢复连接后，通过与系统中其他可信节点交互等干预方法，保证节点数据恢复正常状态且受攻击前的数据不会丢失，并保持与正常节点间数据的一致性。 | 可满足  若某Fabric节点数据遭到篡改，通过比较该数据与其他节点的数据可以发现异常。同时后续数据的提交也会出现异常。  网络恢复后，Fabric节点会通过点对点网络与其他节点交互，补齐数据 | 目前Fabric自身没有主动的检查逻辑，可以在业务层实现定期的检查和报警  第一句话的健壮性是可以满足的，区块可以重新拉取，statedb也可以重建  第二句话,这里需要一定程度的人工干预，所以这里是可满足的  可能需要外部系统来干预 |
| 11.9 容错性  系统中恶意节点占比不超过共识协议的容错率时，系统应保证正常运作，且保持数据一致性 | 满足 |  |
| 11.10 可监管性  单次共识过程和系统运行的整个共识历史都应可审计、可监管，该历史应不可被纂改。 | 满足  每个区块以及区块中的交易，都由参与共识的节点进行签名，从这些签名可以复现共识过程，满足监管要求 | 日志系统存留下来即可,交易,区块本身就是可审计的 |
| 11.11 低延迟  共识协议应保持低响应延迟，满足金融系统对于数据同步的时间要求。 | 不清晰  低延迟如何定义？  从共识角度出发，Fabric具有终局性（Finality），所以无需等待数个区块再确认交易（类PoW算法），所以延迟相对较低 |  |
| 11.12 激励相容  应采用激励机制保障系统安全，计算系统可承载的价值上限，并对其上的应用进行检查，避免超过  安全阈值。 | 不清晰  ”激励机制“是否特指基于Token的方式，亦或是泛指基于政策的激励。但无论是哪种，Fabric都没有内置的能够限制价值承载的方法。另外，该条例所述的”承载的价值“不明确 | 更多的指向的是token的方式，承载的价值  这里的强制的应，是比较难接受的 |
| 11.13 可拓展性  协议应具备动态拓展能力，可允许在系统保持正常服务的前提下动态或静态增删节点。 | 满足  Fabric的节点可以动态增减 |  |
| 12 智能合约 | | |
| 12.1 基本要求  可支持非图灵完备智能合约和图灵完备智能合约，两者都应符合本章安全要求。 | 满足  Fabric链码为图灵完备智能合约 |  |
| 12.2 版本控制  应在源代码中通过金融分布式账本指定的方式定义版本号。 | 满足  Fabric中可定义智能合约版本号，并对版本号的格式限制宽松 | 链码开发者应实现  金融分布式账本格式限制 |
| 应在配置文件中定义版本号，该配置文件应与智能合约代码一同部署。 | 不清晰  配置文件，一同部署  1.x Fabric的CDS中包括有智能合约代码和版本号  2.x 的package中可以用label作为版本号 |  |
| 应在部署或升级操作时定义版本号。  智能合约升级后，应在金融分布式账本中保留前一版本。 交易信息中应明确调用的智能合约版本。 | 满足  Fabric链码部署和升级时都需要指定版本号  Fabric历史版本的链码包会以文件形式存储在peer上  Fabric调用链码可以指定调用的版本，但实际上会始终调用最新的版本。 | 链码开发者应实现  交易信息中应明确调用的智能合约版本。  wuxu版本是可以指定的，指定错了也是最新的  郭哥说显示调用老的合约也是可以的,2.x是肯定可以调用老的版本,1.x里面待确定  智能合约历史数据存储在peer里面，这一点没有疑问  交易信息中明确调用的智能合约版本？这一点待确认  交易中会保留chainid，也就相当于交易中保留了智能合约版本信息了  1.4x解析区块，判断是否有详细的合约版本信息，区块上记录了智能合约的版本 |
| 12.3 访问控制  应有相应的机制控制用户对智能合约的访问。 | 满足  Fabric链码部署目标为节点，而Fabric自带有对节点的控制机制来限制访问链码。  ps：Fabric可通过Policy的设置实现对智能合约的访问，更细粒度的权限控制需在智能合约中单独实现 |  |
| 应有相应机制在支持智能合约之间相互访问的条件下，限制错误智能合约的感染。 | 不清晰  错误智能合约的感染。  同channel下智能合约之间可相互读写，不同channel下智能合约之间可读 | ~~江志坤：框架上没有限制智能合约是错误的这个限制，更多的是看账本的维度了，业务逻辑的可以在代码逻辑层面修改~~ |
| 应有相应机制控制智能合约对外部环境的访问。 宜针对智能合约提供隔离的执行环境。 | 可满足  作为图灵完备合约，Fabric本身不限制对外部环境访问 | 运维开发者应实现  应有相应机制控制智能合约对外部环境的访问。  链码可以直接访问外部的数据，不过不建议访问外部数据， |
| 12.4 复杂度限制  宜从合约源代码总长度、资源消耗和执行时间等方面限制合约代码的复杂度。 | 满足  Fabric链码设有执行超时阈值，  Fabric 客户端调用有超时限制 |  |
| 12.5 原子性  智能合约的执行应有原子性，支持执行过程中发生错误时的回滚操作。 一旦出现异常，所有的执行应被回撤，以避免中间态导致数据不一致。 | 满足  Fabric一般交易采用两阶段提交机制来增强原子性  ps：交易过程为先模拟再提交，智能合约执行出错不会造成账本脏数据，无需回滚 | 链码，中间件开发者应实现相关条例  原本就是两阶段提交,保证原子性的 |
| 12.6 一致性  智能合约执行应具备一致性，合约在所有金融分布式账本网络节点上的执行结果应完全相同。 | 可满足  当所有节点均安装合约，不设私有数据，并在一个交易中指定所有节点进行背书，并确保读集一致，可满足所有节点执行结果相同 | 智能合约本身是可以不一致的，默认假设节点都是正常的，执行结果是可以满足的，达到一致性，执行逻辑可以有区别的  执行结果一致性，是可以满足的 |
| 多个节点同时实现合约时，应保证数据的完整性且数据同步不相互干扰。 | 满足  实现合约理解成链码执行 |  |
| 12.7 安全审计  智能合约的安全审计和评估对象应包括智能合约设计与业务逻辑安全、源代码安全审计、编译环境 审计及相关的应急响应机制等。 智能合约应经过相关专业技术人员的审计，并保留审计记录。 | 可满足  fabric只规定链码运行环境的接口，由此引入链码可插拔性，使得独立的链码审计是方便可行的  ps：智能合约业务逻辑和代码安全方面的审计，强烈依赖于参与的机构 | 由合资格审计机构，单位负责  有一些审计可以通过日志审核  chaincode源代码审计需要请第三方做 |
| 12.8 生命周期管理  从部署到废止的生命周期满足以下要求：  ——应有相应机制控制智能合约的部署行为，防止恶意部署智能合约； | 满足  Fabric链码部署分为安装和初始化两步。  只有组织管理员能够在组织属下的节点上安装链码  初始化权限可以通过链码包初始化策略来设置  ps：某一组织下的节点安装智能合约，只能由该组织的管理员用户执行 | 2.x需要arrpove的  1.x可以不设置实例化的policy，使用默认的，这个地方可能会产生问题  智能合约的废止是啥意思？1.4中可以直接关闭容器即可 |
| ——应提供智能合约的冻结功能，防止智能合约的漏洞持续影响系统； | 满足  冻结功能  可以通过Policy，ACL防止合约被调用 | ACL可以方式合约被调用，更正规一些 |
| ——应提供智能合约升级方案和机制以修复智能合约的漏洞；  ——应提供智能合约的废止功能； | 满足  Fabric链码升级后，可视为一种废止 | 废止是完全不使用了，ACL可以看作冻结，升级可以看作废止，郭哥：不让调用，关闭合约，约等于废止  冻结和废止的准确定义有待确认 |
| ——应支持权限可控的智能合约升级方法； | 满足  Fabric的链码升级与初始化是同质的。因此智能合约升级权限也可以通过链码包初始化策略来设置 | 2.x更好的支持了智能合约升级，权限可控 |
| ——应支持从金融分布式账本中获取与合约相关的原始数据来解析智能合约在金融分布式账本上 的业务数据； | 满足  可以支持从节点的数据库中获取原始数据 | 2.x之后链码的package可以直接解码出来看的  现在几乎是全透明的了 |
| ——应在合约更新升级、重新部署后，能安全地将原合约数据迁移至新合约。 | 满足  Fabric合约升级后，除非通过Init方法做数据修改，否则账本数据不发生任何变化 | 数据过来是肯定可以过来的，合约是无状态的，没有迁移的这个概念，节点的数据是可以迁移的，  这里有点像以太坊的概念 |
| 12.9 攻击防范  应有相应机制保证系统能对抗由智能合约引起的DDoS攻击，防止其长时间占用资源。 | 可满足  Fabric链码设有执行超时阈值  ps：特定智能合约只能通过特定节点访问特定数据源 | Ddos攻击有不同的理解  这里更多的是运维的人来做的 |
| 应有相应机制保障在系统遭受DDoS攻击、服务受到影响时，智能合约的运行可被干预。 | 可满足 |  |
| 应有相应机制防止隔离执行环境中的智能合约访问其执行环境之外的资源。 | 可满足  作为图灵完备合约，Fabric本身不限制对外部环境访问 | 运维开发者应实现 |
| 12.10 安全验证  应基于智能合约安全规则库和问题合约模式库实现智能合约的漏洞检测，可从合约源码和字节码两 方面进行安全扫描。 | 可满足  fabric只规定链码运行环境的接口，由此引入链码可插拔性，使得独立的安全验证是方便可行的  ps：Fabric没有对智能合约漏洞检查方面进行约束 | 安全审计部门可以对链码进行审计  张保佳：字节码指的是编译后的文件的话，可以在链码容器里面找到  依赖于第三方的或者外部的检测 |
| 应实现基于安全规则和配置信息自动生成安全智能合约模板的机制。  宜通过形式化方法验证智能合约代码的正确性。 | 可满足  ps：智能合约由模板自动生成可有参与机构根据业务场景进行约定和开发 | 由链码开发者负责，如采用合约支持语言下的代码生成器  Fabric官方没有这个生成模板，但是应该是比较好实现，但是添加了安全，就比较玄乎了  不过有一些默认的规则 |
| 13 身份管理 | | |
| 13.1 基本要求  应实现有效的用户身份管理，主要功能包括身份注册、身份核实、账户管理、凭证生命周期管理、身份鉴别、节点标识管理、身份更新和撤销等。同时，应保障身份信息的安全性，并对身份进行监管审计。 | 可满足  本章节解析基于  假设  - 账户对应Fabric的用户概念  - 身份对应Fabric CA的Identity概念  FabricCA大致实现的Identity功能包括  身份注册，身份鉴别，身份撤销，身份更新  Fabric已经实现的Fabric用户管理包括  登记，注销，重新登记  身份注册定义见 注2，具体章节13.3  身份核实定义见 注3，具体章节13.4  节点标识管理见章节13.8 | 江:身份核实是线下的操作,框架中这一部分是缺失的  曹:证书是可以确认身份的  核验的是线下的操作,fabric的ca是自己处理的，需要接入第三方的ca的证书  可以将商用的证书导入fabric ca中来,fabric1.4本身是支持rsa的  用户和账户的问题大家在讨论  在身份管理章节中出现的节点标识管理是什么  曹：fabric目前拥有身份管理的功能，基于PKI体系实现，实现了专门的组件（MSP），成员身份是基于标准的X509证书，结合所属的MSP进行身份的认证和校验；可以通过MSP来实现不同的权限控制，接入现有的PKI系统或者使用fabric-ca来生成证书；满足身份管理的基本要求，但缺少账户管理，缺乏审计的功能，这部分需要做补充实现 |
| 13.2 身份定义  身份是指涉及自然人及法人等实体的属性的集合。在金融分布式账本系统中，身份可以进行数字化标识（简称数字标识）。 | 满足 | 需要增加账户体系，添加身份凭证  刘:账户对应的是fabric中用户的概念，身份可以对应fabricCA中的identity，fabric可以根据attribute中的属性来进行权限控制  吴:身份可以对应到org这个层级,用户可以对应到admin,一般的证书签发都到法人,可以这么理解,应用账户没有找到合适的对应点  刘:不是系统账户的都是应用账户  曹:身份指的是实实在在的单位,需要自己添加映射关系,账户的观念还需要进一步处理  刘:描述部分需要修改,可满足是正确的 |
| 账户是身份的一个属性集合，分为系统用户账户和应用账户。系统用户账户包括普通成员账户、系统管理员账户和其他特定权限的系统用户账户，其中系统管理员账户具有最高权限（如部署智能合约）。在金融分布式账本系统中，一个身份可对应多个账户。每个账户应关联一个身份标识，即身份凭证。 | 假设  - 区块链核心系统中的账户 对应 系统用户账户  - 外围系统账户 对应 应用账户  - 普通成员账户 对应 Fabric User  - 系统管理员账户 对应 Fabric Admin User |  |
| 身份凭证是用户实体通过身份鉴别后，由鉴别者为用户出具的一种可信任的电子凭据，包括但不限于数字证书和公私钥对等，不同的鉴别及验证方式应遵循金融业的业务及监管要求。  注1：身份定义的实体范围为自然人和法人等，不包括设备实体。 | 假设身份凭证 Fabric 用户KeyPair以及关联的证书  按注5，这里的 “身份鉴别”应该改为“身份核实” |  |
| 注2：身份注册是指自然人及法人等实体向注册机构提供权威机构发行的法定身份证件等身份证明材料，申请获取账户和身份凭证。  注3：身份核实是指注册机构向身份信息权威机构核验注册者提供的身份证明材料是否与注册实体一致。  注4：账户授权是指身份注册机构对注册实体的账户进行权限分配的过程。  注5：凭证签发是指完成身份核实后，身份注册机构向注册实体的账户发行身份凭证。  注6：身份鉴别是指自然人及法人在使用分布式账本服务/活动过程中，对注册实体的凭证和属性进行鉴别的过程。 | 由定义得出，身份注册是由身份核实，（建立账户），凭证签发按顺序组合构成 |  |
| 13.3 身份注册  身份注册机构应建立健全身份生命周期的管理规章制度和信息系统，并确保身份生命周期管理过程中无信息泄露等安全问题。同时，身份注册机构应接受监管部门的紧急干预和审计追踪。 | 不适用  本段规范约束的对象的是身份注册机构。不属于fabric范畴 | 该功能需要第三方的服务提供支撑，比如现有的可信CA机构  江:fabric支持硬件密码吗？  张:支持pkcs11,pkcs11的支持仅仅只限于bccsp,tls是不支持的  刘:SKF,SDF  曹:感觉这部分的内容于fabric的关系不是很紧密,fabric只能确保不会被改变,基本上都保存在本地的MSP里面,上链啥的都会用到这些信息  张:法律法规需要额外的关注,  江:其他的例如个人信息需要额外的系统支持,这些系统建设相关的信息在其他的段落有规定的,等保三,等保四等有待讨论  刘:在一定程度上,国密算不算是一个问题  吴:身份注册更多的是偏向一点管理手段,流程与系统支撑上的东西 |
| 注册机构对注册实体信息的收集、使用、存储、传输、销毁等过程应符合国内外相关法律法规。  注册机构对注册实体信息的收集应符合最小化要求。  注册机构对注册实体信息的收集应取得信息主体的明示同意。  注册机构对注册实体信息的保存时间应符合最小化要求。 | 本段规范约束的对象的是身份注册机构。不属于fabric范畴  不清晰  国内外相关法律法规。  最小化要求。 |  |
| 注册实体应按照注册机构的要求提供实名登记信息和身份核实所需材料，并确保其真实有效。 | 本段规范约束的对象的是身份注册机构。不属于fabric范畴 |  |
| 注册过程中，注册机构应避免注册主体重复注册，并保证注册主体提交信息的传输安全和存储安全。 | 本段规范约束的对象的是身份注册机构。不属于fabric范畴  FabricCA有机制避免一个identity被重复注册，可供参考 |  |
| 13.4 身份核实  注册机构应核实注册实体身份，确保参与主体的身份真实可信。  注册机构应完善核实管理制度，增强核实系统能力。  核实过程应避免单人完成操作。  核实过程应避免主观臆测，应具备材料核验的技术手段。  核实过程应能接受监管审计。  在身份核实阶段不应弄虚作假、核实操作员不应渎职不作为、核实过程应严谨。身份核实错误导致的系统损失由身份注册机构承担。 | 可满足  本段规范约束的对象的是身份注册机构。不属于fabric范畴 |  |
| 存在隐私保护需求的金融分布式账本系统可使用匿名身份认证。但应遵循“前台自愿、后台实名”的原则，前台使用匿名标识，后台应能还原注册实体的实名身份。 | 谓词“可”  是否需要实名<-->标识 匹配系统 |  |
| 13.5 账户管理  13.5.1 账户创建  应具备账户管理功能。  每个账户应关联一个身份标识，并在交易数据中携带发送方的账户身份标识。 | 可满足  按照定义，身份标识等同于身份凭证  若需要满足 “交易数据中携带发送方的账户身份标识”  不应提供完整身份凭证信息，而是提供刚好足以识别发送方的部分凭证信息  Fabric交易上下文当中的Creator表示了交易发送方的部分信息（证书） | 需要添加账户体系，实现身份与账户的映射，适当的在智能合约中加以控制 |
| 账户的身份标识应在该分布式账本系统中具有全局唯一性，且不易被冒用。 | 可满足  按照定义，身份标识等同于身份凭证  Fabric 用户KeyPair 和 证书  那么通常会拥有全局唯一性 |  |
| 对于存在隐私保护需求的系统，应支持账户标识的匿名化处理。 | 条件性规范  不属于fabric范畴，实现时可考虑参考idemix  通过离线实名查询系统来同时保证匿名与实名制的需求 |  |
| 应设置普通用户账户、管理员账户和其他具有特定权限的系统账户。管理员账户应具有冻结和解冻其他账户的权限。 | 冻结，解冻定义见13.5.4 |  |
| 系统应提供上述三类账户相应的注册、授权、变更和注销功能。 | 可满足  通过FabricCA协同MSP机制满足 |  |
| 13.5.2 账户授权  授权应由身份注册机构完成或发起。  普通权限可由身份注册机构独立完成授权。 | 本段规范约束的对象的是身份注册机构。不属于fabric范畴 |  |
| 监管等特殊权限的授权，应由身份注册机构发起，并按照定义好的共识策略，由各个参与方共同决定，达成共识后完成授权。共识策略由身份注册机构制定，各参与方应知晓并认同该策略。 | 满足  Fabric Admin用于特殊权限账户，是需要共识后才能授权的 |  |
| 13.5.3 凭证签发  用户在获取其身份凭证时，凭证签发机构应通过权威机构对实名登记信息进行核验后，方可颁发身份凭证。 | 本段规范约束的对象的是身份注册机构。不属于fabric范畴 | 该功能需要第三方的服务提供支撑，比如现有的可信CA机构 |
| 系统为不同的用户分配不同的身份标识，标识应具有不易被冒用的特点。 | 同13.5.1 |  |
| 13.5.4 账户冻结和解冻  账户冻结应由注册机构发起，经共识后修改账户状态为冻结。冻结的账户不能进行交易。  账户解冻应由注册机构发起，达成共识解冻后的账户可继续进行交易。 | 可满足  本段规范约束的对象的是身份注册机构。不属于fabric范畴  可以通过channel.msp: revoke\_list来帮助实现  如果需要采用共识，调整ModPolicy成为多个管理员账户背书 | 该功能需要第三方的服务提供支撑，比如现有的可信CA机构  fabric端需要实时验证某个账户是否有效 |
| 13.5.5 账户锁定和恢复  为防止恶意的私钥重置，系统应设置一个重置私钥的锁定窗口期。  在锁定窗口期内，用户无法进行账户资产操作，但用户可通过原有的私钥来解除重置并恢复账户状态。  窗口期结束后，如果没有发生解除重置，用户才可执行公私钥对的重置操作。  重置私钥的锁定窗口期可通过系统治理参数进行设置。 | 不清晰  重置私钥 |  |
| 13.5.6 账户注销  账户应设定使用期限，过期账户应被注销。 | 可满足  假设Fabric User对应账户，revoke certificate 对应账户注销  可以采用定时任务根据证书有效期来注销 | 该部分功能，应在fabric中建立账户体系时，进行实现。需要一定的编码工作。 |
| 应提供注销申请功能。实体申请注销或因法律和监管等要求而强制注销账户时，应能及时注销。  应定期发布公开和可查询的账户注销列表。 | 本段规范约束的对象的是身份注册机构。不属于fabric范畴 |  |
| 若私钥丢失，应能在不修改账户身份标识的情况下重置公私钥对，保证账户的资产不丢失。 | 不清晰  按照定义，身份标识等同于身份凭证  私钥丢失，重置私钥  账户的资产不丢失。 |  |
| 对于已注销账户，在取消其登录和操作权限的同时，应长期或永久保留其登记信息，并永久保留其身份标识，避免重复分配。 | 可满足  假设revoke certificate 对应账户注销，则注销后Identity不变 |  |
| 13.6 凭证生命周期  13.6.1 基本要求  金融分布式账本的凭证管理应包括凭证的产生、存储、使用、撤销、终止整个过程的管理。 | 不清晰  这里的凭证和上文中的身份凭证是否等同  终止凭证 |  |
| 对不同金融业务所需凭证中包含的信息、数据格式和加解密规则，应编写专门的文档加以说明。  应明确用户身份凭证类型和内容，如数字证书、认证口令、生物特征等。 | 满足  假设身份凭证 Fabric 用户KeyPair以及关联的证书 |  |
| 应记录用户身份凭证建立、更改、授权、禁用、终止等操作日志，确保用户身份凭证相关操作可追溯。  应建立用户身份凭证信息防伪机制与管控措施，避免用户身份被冒用或相互转借等安全风险。 | 不清晰  终止凭证  防伪机制  相互转借等安全风险 存疑 |  |
| 13.6.2 凭证产生  应由用户向凭证提供方发起凭证申请。  凭证提供方应对用户进行身份核验（身份核验可通过离线或在线方式进行）。 | 本段规范约束的对象的是身份注册机构。不属于fabric范畴  同注5 |  |
| 应将账户身份标识和凭证之间的关系进行解耦，不允许二者存在数学运算关系。 | 不清晰  账户身份标识  与上文冲突  每个账户应关联一个身份标识，即身份凭证。 |  |
| 通过身份核验后，凭证提供方生成带有其数字签名的凭证发放给用户。 | 根据此条例理解，Fabric系统中的凭证应为用户证书 |  |
| 13.6.3 凭证发放  身份注册机构应保证数字身份凭证的私有部分的安全传输。  身份注册机构应确保数字身份凭证的私有部分不被第三方窃取。  身份注册机构可保留注册实体的数字身份凭证中可公开的部分，用于身份验证。 | 本段规范约束的对象的是身份注册机构。不属于fabric范畴  假设凭证私有部分是私钥 |  |
| 13.6.4 凭证存储  凭证应由用户和凭证提供方双方各自进行安全存储。  凭证存储应符合第14章“隐私保护”中相关个人隐私数据保护要求。  应明确用户身份凭证实现机制以及持久性存储的目的、方法和位置。  宜建立用户身份凭证存储方法评估程序。 | 可满足  应建立用户凭证存储机制  应建立凭证提供方安全存储机制 |  |
| 13.6.5 凭证流转  凭证流转应由用户发起，对凭证信息的访问应经过用户授权许可。 | 不清晰  凭证流转  场景不明确 |  |
| 13.6.6 凭证验证  凭证支持基于密码算法的验真功能。  凭证需求方应对用户提交的凭证进行真实性验证。  凭证信息的获取应遵循最小化原则。 | 满足  如有需要，可采用idemix |  |
| 13.6.7 凭证更新  凭证到期之前，应根据用户提出的更新请求，采取安全快捷的方式确保用户凭证重新生效。  应建立保障机制，确保用户数字身份属性变动时，应能及时更新属性值。 | 满足 |  |
| 变更属性核实确认后，应进行更新登记，并将结果反馈给用户完成更新流程。 | 可满足 |  |
| 13.6.8 凭证终止  凭证提供方应根据需要设置凭证的有效期限，凭证到期自动失效。 | 满足  Fabric 默认提前一周提醒证书过期  Fabric 任何证书到期都会自动失效 |  |
| 凭证生命周期管理的每个过程都应有对应的信息存证操作，存证通常是可通过密码学方式证明某件事或某项文件为真的证据。 | 可满足  按照安全审计 |  |
| 13.7 身份鉴别  应提供专门的组件或模块实现用户身份认证功能，该组件或模块应确保正确标识和鉴别相关个人、组、角色、设备、应用等主体身份信息和授权信息。 | 可满足  本段规范约束的对象的是身份认证组件。不属于fabric范畴 | 该功能需要第三方的服务提供支撑，比如现有的可信CA机构  fabric本身能够支持安全性及不可抵赖性等；但缺乏相关的安全分析等需求实现，相关的功能需要编码加以提取或者整理 |
| 应采取技术措施对通讯双方的身份进行认证。  应采取必要措施，防止认证信息被窃听或冒用。  应具有认证失败处理机制，可采取结束会话、限制非法认证次数和自动退出等操作。 | 满足 |  |
| 应使用安全并符合国家密码管理规定的算法和协议进行身份认证，并设置最小化反馈和安全退出措施。  涉及跨系统业务的分布式账本系统，应满足跨系统的身份认证要求。 | 可满足 |  |
| 使用公钥密码算法实现身份认证的分布式账本系统，在交易过程中的电子签名应具有不可抵赖性。  鉴别过程应采用数字签名等技术确保安全性，且不应传送鉴别凭证的私有部分。 | 满足  假设凭证私有部分是私钥 |  |
| 应采用密码技术和访问控制技术，确保用户口令等身份认证相关凭证信息的存储安全性。用户口令应经过散列函数处理后，进行加密保存，避免以明文或可逆加密、易于碰撞探测的方式保存用户口令。  应定期对用户账号的使用情况进行安全性分析，对登录时间、登录位置、访问时长、访问模块等进行综合分析，并评估账号安全风险。  对重要数据、业务或系统的操作，应采用双因素身份认证。 | 本段规范约束的对象的是用户系统。不属于fabric范畴 |  |
| 使用匿名身份认证的分布式账本系统，其匿名认证方式应具有匿名性、不可伪造性和不可链接性。 | 满足  可用idemix实现 |  |
| 13.8 节点标识管理  应明确节点授权机构及管理员。  通信节点加入系统之前，应由授权机构给予其在系统内唯一的节点标识，并提供与之对应的标识鉴别信息和标识凭证，授权机构应在凭证中指定节点角色。 | 可满足  实现节点标识时可以参考fabric Peer配置项peer.id  若以peer的X509证书作为凭证，则凭证中可以加入attr并与节点角色做对应关联  凭证中的节点角色只作为一个标签 |  |
| 标识凭证由授权机构确保其完整性和真实性，应符合第8章“密码算法”中对完整性和真实性的要求。  标识鉴别信息应具有不易被仿冒的特点，如数字、字母和特殊字符的组合，并设定更换期限，应在期限到来之前进行更换。  在传递及存储标识鉴别信息之前，应采用符合第8章“密码算法”中要求的保密性及完整性保护。  节点之间建立数据通信连接之前，应先通过标识鉴别信息实现双向身份认证，并建立一条安全的数据通信信道，该过程应符合第8章“密码算法”中对保密性和完整性的要求。  应具有节点标识认证失败时的处理机制，可采取结束通信、限制认证失败次数和超时自动结束等措施。 | 不满足  假设标识凭证为节点证书  除了第8章的国密要求之外，满足 |  |
| 13.9 身份更新和撤销  应提供实体对身份信息进行更新和撤销的功能。 | 满足 |  |
| 当实体对身份信息进行更新时，应针对身份信息进行重新核实。 | 本段规范约束的对象的是身份注册机构。不属于fabric范畴 |  |
| 13.10 身份信息安全性  13.10.1 基本要求  身份信息安全除满足GB/T 35273—2017中关于开展收集、保存、使用、共享、转让、公开披露等信息处理活动应遵循的原则和安全相关要求外，还应从保护身份标识和身份属性两方面保证安全性。  应根据风险要求，将身份数据元素存储在不同的系统中，并在相应系统中受到不同等级的保护，保护要求应依据具体业务进行特定说明。  应根据金融业务需求制定身份数据保密性要求，确保数据不暴露给未经授权方。  系统应提供基于属性的访问控制，在数据对象的整个生命周期中身份数据始终保持保密性、完整性，和可验证性。  除保护存储数据的流程之外，系统还应制定合规、认证和审计策略，并考虑监管要求和隐私保护要求。  应在用户注册时授予访问安全系统的权限。  可为常见用户级别预定义访问控制设置以对访问进行差异化管理，降低管理复杂性。  可具备授权功能作为访问控制的扩展。 | 可满足  假设身份标识是证件号码  身份属性为其他字段信息，如名字，户口所在地等  本段规范约束的对象的是身份信息系统。不属于fabric范畴 |  |
| 13.10.2 密钥安全性  应编写针对性的文档说明身份密钥的类型及生命周期管理，说明使用的密钥、算法、机制为身份标识提供保密性和完整性保护。  密钥管理系统应具有管理密钥创建、派生、分发、存储、安全性和其他管理安全审计功能。金融分布式账本系统中，身份密钥的管理应遵循金融行业密钥管理标准。  系统应在用户丢失密钥、密钥过期或受到其他危害时提供密钥轮换、销毁和替换的方法。由于需要限制风险暴露，密钥轮换、销毁和替换的管理应充分标准化。  用户身份密钥应采用符合金融市场所需的目标级安全要求的加密算法及密钥长度。 | 本段规范约束的对象的是身份密钥系统。不属于fabric范畴 |  |
| 13.10.3 安全加密  采用的密码算法和密码技术应符合国家密码管理部门颁布的GM/T 0045—2016。  应具有在客户端保护私钥及凭证的手段，如采用的个人密码设备应符合行业主管部门和国家密码管理部门的要求。 | 不满足 |  |
| 应具备时间戳功能，保证正确的交易顺序。 | 满足  能保证交易顺序，但不是通过时间戳功能 |  |
| 13.11 身份监管审计要求  13.11.1 监管  监管信息应至少包括金融监管信息，具体为现工作单位/就读学校、行业类型、居住国家/地区、民族、居民/非居民、出生日期、个人月收入、税务信息等监管数据项和反洗钱特色数据项。  身份注册机构应具备收集客户监管信息的方法和手段。  身份注册实体、凭证适用方应明确信息监管的目标、方式、范围、规则等，监管机构应征求并取得授权。  在特殊情况下，监管机构无需征得信息主体的授权同意，包括如下情况：  ——与国家安全、国防安全直接相关的；  ——与公共安全、公共卫生、重大公共利益直接相关的；  ——与犯罪侦查、起诉、审判和判决执行等直接相关的；  ——出于维护信息主体或其他个人的生命、财产等重大合法权益但又很难得到信息主体同意的；  ——所收集的信息是信息主体自行向社会公众公开的；  ——从合法公开披露的信息中收集主体信息的，如合法的新闻报道、政府信息公开等渠道；  ——根据信息主体要求签订和履行合同所必需的；  ——用于维护所提供的产品或服务的安全稳定所必需的，例如发现、处置产品或服务的故障；  ——信息控制者为新闻单位且其在开展合法的新闻报道所必需的；  ——信息控制者为学术研究机构，出于公共利益开展统计或学术研究的必要，且其对外提供学术研究或描述的结果时，对结果中所包含的信息进行去标识化处理的；  ——法律法规规定的其他情形。 | 本段规范约束的对象的是身份注册机构。不属于fabric范畴 |  |
| 监管权限应严格审批和使用，宜采用共识策略。 | 不清晰  与上文冲突  虽然这里采用关键字“宜“，但根据上文所述  监管等特殊权限的授权，应由身份注册机构发起，并按照定义好的共识策略，由各个参与方共同决定，达成共识后完成授权。共识策略由身份注册机构制定，各参与方应知晓并认同该策略。 |  |
| 13.11.2 审计  应对身份、账户、凭证的访问和更改提供安全审计功能，审计记录包括访问的日期、时间、用户标识、数据等审计相关信息。 | 可满足  依托传统安全审计 |  |
| 身份生命周期管理中需要通过共识机制完成的业务流程，应记录策略、共识节点、账本数据校验值等审计相关信息。 | 条件性规范  Fabric Identity生命周期管理不需要通过共识机制完成 |  |
| 审计记录不仅包括变更成功的记录，还应包括变更失败的记录。 | 依托传统安全审计 |  |
| 14 隐私保护 | | |
| 14.1 隐私保护原则  金融分布式账本上所有的隐私保护行为应符合GB/T 35273—2017中的“个人信息安全基本原则”， 且不违反金融业相关监管要求。 个人信息应以合法、公正、透明的方式处理。 个人信息收集目的应明确和合法，任何与目的不符合的方式不可采用。 信息收集应遵循最小化原则，个人信息收集应仅限于一切与信息收集目的相关且必要的数据。 个人信息应准确，如果需要应尽可能保持最新的信息。 在仅收集必要的个人信息的情况下，允许收集的数据以可识别的形式保存。 应确保个人信息以适度安全的方式处理，包括使用适当的技术或机制来对抗未经授权或非法的处 理、意外遗失、灭失或损毁等情况，其技术手段应符合金融业认定的技术方式和相关信息保密方式。 | 不适用 |  |
| 14.2 隐私保护内容  本标准中所涉及的隐私信息是指在金融分布式账本系统中，单独或者与其他信息相结合能识别特定 自然人身份或者反映特定自然人活动情况的各种信息，包括但不限于分布式账本系统中各方的账户信 息、鉴别信息、交易信息、个人身份信息、财产信息及其他反映特定自然人活动的各种信息。 | 定义 |  |
| 应将隐私信息按照敏感程度进行分级，并设置对应的隐私保护策略。低隐私保护要求类别信息经过 组合、关联和分析后可能产生高隐私保护要求类别信息，应根据实际情况采用对应的高隐私保护策略。 | 不适用  本段规范约束的对象的是隐私信息管理机构。不属于fabric范畴 |  |
| 14.3 隐私保护策略  14.3.1 概述  分布式账本提供的信息保密性和隐私保护的程度与执行效率方面存在制约关系，相关方应根据具体 场景选择不同的方法和技术，制定隐私保护策略，达到满足系统目标的平衡状态。 | 定义 |  |
| 14.3.2 信息公开可验证  公开交易内容信息以及交易方信息。 应对交易方身份信息进行标识和鉴别。 应确保交易方无法被冒用。 公开的信息应确保任何人能进行有效性和正确性的验证。 | 满足 |  |
| 14.3.3 信息加密可验证  应对交易内容信息以及交易方信息至少其一进行加密。  应确保参与方以及审计方拥有对加密信息解密验证的能力。  应确保除交易参与方以及审计方外，他人无法从加密信息获取任何其他信息。 | 可满足  Fabric应用完成的部分 | 开发者应注意  应对交易内容信息以及交易方信息至少其一进行加密。  应确保参与方以及审计方拥有对加密信息解密验证的能力。 |
| 应确保交易方无法被冒用。 | 不清晰  重复要求 |  |
| 应确保除交易参与方外他人无法伪造加密信息。 | 满足 |  |
| 应确保任何人可对加密的信息进行有效性和正确性的验证。 | 不清晰  与上文冲突  应确保除交易参与方以及审计方外，他人无法从加密信息获取任何其他信息。 |  |
| 14.3.4 信息由交易验证节点验证  应对交易内容信息以及交易方信息至少其一进行加密。 | 重复要求 |  |
| 应确保参与方、交易验证节点以及审计方拥有对加密信息解密验证的能力。  应确保除交易参与方、交易验证节点以及审计方外，他人无法从加密信息获取任何知识。 | 假设 交易验证节点等同于Fabric Peer  可满足  当它作为endorser的时候，可以设计Chaincode来实现解密验证的能力 |  |
| 应确保交易方无法被冒用。 应确保除交易参与方外他人无法伪造加密信息。 | 重复要求 |  |
| 应由交易验证节点负责对信息进行解密验证，以对其有效性和正确性进行验证。 | 可满足  同上  当它作为endorser的时候，可以设计Chaincode来实现解密验证的能力 |  |
| 交易验证节点承担 对已验证交易信息的担保责任。 | 满足  Fabric endorser成功的提案响应意味着承担责任 |  |
| 他人应通过交易验证节点的验证信息对交易的有效性和正确性进行验证。 | 满足 |  |
| 14.4 隐私保护技术要求  隐私信息生命周期是指在金融分布式账本上对隐私信息进行收集、传输、共识、存储、使用、销毁 等处理的整个过程。  应从认证授权、访问控制、保密、完整性、审计、监控、策略等方面，采取相应的 技术手段保证隐私信息全生命周期各环节不被未授权的第三方获取，并保护交易方的身份不被识别和冒 用。  隐私保护技术和方法包括认证授权、局部广播、摘要存储、变更标识、混淆技术以及零知识证明、 群签名、环签名、同态加密等算法组合，可根据业务场景组合解决方案，实现信息保密性和隐私保护的 目的。 | 定义 |  |
| 隐私保护技术要求如下：  ——信息采集时应有醒目提示信息，并明确告知客户哪些个人信息会被采集；  ——信息采集时应包含客户勾选同意或确认的操作步骤，应有明确授权；  ——信息采集时应默认对身份标识信息进行部分隐藏，同时提供全部显示手段；  ——信息采集时应对客户和采集的信息进行匹配认证，并对完整性进行校验；  ——信息采集时应明确告知收集信息的目的、处理方式、存储期限、智能合约逻辑内容；  ——信息传输时应对信息进行全量加密，加密的密钥和证书不能采用信息传输的同一传输通路进行 传递；  ——停止运营产品或服务时，应及时停止收集数据的活动，并及时告知客户和为客户提供信息注销 不可见的手段，并向其他节点或组织发布停止运营和处置数据的信息；  ——密钥发送客户后应明确告知其妥善保管密钥，并提供密钥更换手段；  ——信息存储时应对客户的隐私信息进行加密；  ——信息存储时应对客户身份标识信息进行摘要存储； ——信息在第三方存储时应告知客户并获得客户授权； ——信息展示时应对客户身份标识信息进行部分隐藏，可额外提供全显示手段，非密文展示应采取 去标识化措施；  ——信息展示时，对非本人展示应先获得信息所有者的授权，并对展示人进行认证；  ——信息使用时，应明确记录使用者、使用内容、使用时间、使用频率；  ——信息向外部扩散时，应告知客户并获得授权，并提供给客户延缓甚至中断扩散、减少扩散影响 的手段； ——应向客户提供信息备份和导出的手段，备份和导出的信息应加密处理，并向客户提供解密手段；  ——应向客户提供信息注销不可见的手段；  ——信息注销不可见时应获得客户认证和授权；  ——信息加工后产生的信息，也应满足上述各项要求。 | 定义 |  |
| 14.5 隐私保护监控与审计  应制定完备的隐私保护审计方案，审计内容包括隐私保护策略和隐私保护技术手段，审查形式包括 但不限于日常监控、定期审计、不定期审计。 | 可满足  依托传统安全审计 |  |
| 应审查所制定的隐私保护策略和隐私保护技术手段的合理性，包括但不限于对隐私保护原则的遵循 程度，对不同隐私保护等级的金融信息风险防范要求的匹配度，在当前技术环境下的适用性。  应审查隐私保护策略和隐私保护技术手段的执行过程，审查对象包括但不限于操作手册、操作记录 等支持性文档，确认执行过程遵循并实现既定的策略和技术手段。 应审查隐私保护策略和隐私保护技术手段的实际执行效果，确认隐私保护内容得到有效保护，达到 既定的风险防范要求。 应在必要时对隐私保护策略进行修订，对隐私保护技术手段进行变更，并按照新的策略和技术手段 实现隐私保护。 | 依托信息安全等级保护 |  |
| 具体地，可在分布式账本系统中设定超级账户，由其执行隐私策略和技术手段的变更。 | 满足  谓词 "可" |  |
| 针对不同隐私保护等级的金融信息制定不同的监控和审计规则和策略。 | 可满足  依托信息安全等级保护 |  |
| 15 监管支持 | | |
| 15.1 基本要求  金融分布式账本系统具有架构去中心、数据多副本、交易点对点、记录不可篡改的特点，与中心化系统有很大差异，不仅需要法律监管规则，也需要技术监管规则，以优化系统设计、保证系统安全、提高监管效率、降低合规成本。 | 满足 |  |
| 15.2 系统监管  应支持监管机构的接入，以满足信息审计和披露的要求。  应支持监管部门的监管活动，包括但不限于设置监管规则，提取交易记录，按需查询、分析特定业务数据等。  应支持监管机构访问最底层数据，实现穿透式监管。 | 可满足  fabric中的组织概念可对应到这里的监管机构，即可将监管机构作为一个peer组织来加入fabric网络，让这个组织加入fabric网络中的所有通道，则可获取到网络中的所有交易。进而可提取交易记录，按需查询以及分析特定的业务数据。当监管机构作为peer组织加入后，可部署自己的peer节点，获取账本数据，从而可也拥有了最底层的数据，实现穿透式、实时监管。 | 但fabric自身框架中没有监管者的身份，监管规则似乎无法直接映射到fabric中的概念，只能在业务层或自有的智能合约中实现。  在让监管组织加入通道之前，需要特别留意通道中关于更改组织的权限设置，防止遇到难以获得其他组织授权的问题  此外如需分析私有数据的源内容，需要进行链码配置的更新操作 |
| 15.3 信息管理  应支持还原匿名标识中的用户真实身份以及相关交易信息，配合交易审查，加强KYC管理。 | 可满足  fabric中所有用户都以证书的方式来标识身份，同时所有的交易都都有签名。因为证书的签发是具有一定的规范，无论是第三方CA还是自签CA签发的证书都会带有明显的身份信息标识，从而可以通过证书来还原交易的身份信息，配合交易审查。同时将根证书的添加到fabric网络中，需要其它方的签名校验，在这过程中也会对应到用户真实身份的识别。因此fabric是可以实现KYC管理的。  但这种KYC是分层管理的，各组织对自己签发的证书进行KYC管理，没有一个统一的组织能实现所有用户的KYC识别。此方面是否需要增强，还有待进一步讨论。 | 应构建用户真实身份与交易者身份证书关联的管理系统  需定义匿名标识、真实身份的具体定义。 |
| 15.4 事件处理  当系统或交易出现问题时，应能主动报警，采取适当纠正措施，并向监管机构、管理机构报送事件信息。 | 可满足  fabric中交易问题的正确与否需在最后peer落账时才能完全确认，能判定出出现问题的交易，也可通过接口实现主动报警，向监管机构、管理机构报送事件信息。但因为会将交易标记为非法交易，同时该笔交易已经经过fabric网络共识，同步到所有节点，无法采取纠正措施来更改交易。 | 服务提供商实现  区块链层负责拿到交易，之后问题发现、主动报警由应用层实现。 |
| 15.5 交易干预  应具备限制交易权限、冻结账户等功能，为监管机构提供交易干预的技术手段。 | 满足  fabric中所有的交易都需要证书，可通过吊销证书的方式来实现限制交易权限，从而为监管机构提供交易干预的技术手段。~~但fabric中现阶段还没有账户的概念，此功能不具备。~~ |  |
| 15.6 智能合约监管  应能按需将监管要求编码写入智能合约强制执行。  应能根据需要为监管机构提供交易行为统计数据，评价智能合约所提供服务的合规性。 | 满足  监管方可对各机构的智能合约进行合约代码审核，从而确认合约中是否有监管要求编码。在fabric2.0中，如果赋予监管方一定的权限，甚至可以不批准不符合监管要求的智能合约。 | 监管机构可以获取所有的交易数据，从而对交易行为进行分析统计。但该功能非fabric系统提供，而需借助于第三方系统来实现。 |
| 16 运维要求 | | |
| 16.1 基本要求  运维要求应符合GB/T 22239—2019中安全运维管理相关要求，同时还应包括设备管理、节点监控、节点版本升级、漏洞修复、备份与恢复、应急预案管理、权限管理、议案机制等功能。 |  |  |
| 16.2 设备管理  加密机应放于专门区域，指定专人管理，并定期进行维护管理。  应采用白名单机制控制对加密机的访问，阻止非授权设备访问加密机。  在报废加密机前，应将加密机内的密钥完全清除，确保加密机内的密钥等敏感数据无法被恢复重用。  严格控制加密机的变更操作，经过审批后才可进行变更操作，并须留下变更相关的审计日志。  加密机以外的设备应遵循GB/T 22239—2019中的相关要求。 | 加密机的管理更多偏向于人为的管理措施，非技术范畴。 |  |
| 对网络中的主节点性能，需要根据实际承载的业务场景协商接入标准，主节点应满足该协商标准。  注：网络中主节点性能指标包括CPU、内存、带宽等指标。 | ~~fabric节点性能随着部署服务器的性能不同而不同，具有一定的弹性。~~ |  |
| 16.3 节点监控  应对节点运行状态以及节点与其他节点的连接进行监控。 | 满足  fabric对于节点提供了health探测接口以及discovery服务，从而可对运行状态与连接进行监控。 |  |
| 应对节点进行高可用的架构，用来应对部分节点的异常。 | 满足  fabric本身就属于分布式服务，因此高可用架构时天然支持的。部分节点异常对整个网络是无影响的。 |  |
| 如发现部分节点运行异常，应在不影响服务运行的同时及时进行问题的排查，在规定的时限内将节点恢复至正常状态或启动备用节点，以保证业务的正常运行。  如果无法在规定的时限内恢复节点，则应将情况上报管理委员会。  如发现节点与其他节点的连接异常，应在不影响服务运行的同时，排查其他节点的故障，且应将情况上报管理委员会。 | 恢复手段属于管理措施，非技术范畴。 |  |
| 应收集系统中运行的状态数据，包括节点的远端同步节点数、账本同步平均耗时、节点的健康状态等，并写入日志供测试或者审查用，如异常应及时预警并处理。 | 可满足  fabric对于系统中运行数据可提供Prometheus或statsD等收集机制进行归拢，同时日志也可跟踪记录，以备审查。如出现异常，也可及时探测，为预警提供基础。节点的远端同步节点数与节点的健康状态两个状态数据fabric已支持，账本同步平均耗时待确认。 | 账本同步平均耗时暂时没有原生指标提供，需要在应用层进行增强 |
| 如发现恶意或者欺诈节点传播恶意损坏的账本数据，或者有节点篡改账本，应先定位节点位置，获取详细信息，上报管理委员会。 | 欺诈的orderer、peer等如果伪造了区块并传播，节点接收后无法通过验签，会直接丢弃；有待确认 |  |
| 16.4 节点版本升级  节点的版本升级前，应在测试环境进行验证，应保证升级过程中对业务的平滑过渡。除非发生重大安全事故，应避免采用所有节点同时停止服务进行升级的方式。 | 管理措施 |  |
| 节点的版本升级应支持向下兼容，升级后仍须支持旧版本的数据。 | 满足  fabric从1.0版本之后已支持向下兼容，同时支持旧版本的数据。 |  |
| 节点的版本升级应记录升级过程中相关操作日志，做到可审计、可追溯。  应制定版本升级失败的应急预案，在升级失败的情况下启动应急预案进行回滚，在规定时间内恢复节点的可用性。 | 管理措施 |  |
| 16.5 漏洞修复  应对节点的服务器进行定期漏洞扫描检查包括但不限于服务器本身的漏洞、分布式账本软件的漏洞、智能合约的漏洞，并对发现的安全漏洞和隐患提出修复方案进行审批，审批通过后进行修复，无法由运维修补的漏洞须评估可能的影响并上报。  进行漏洞修复，应不影响节点的账本数据、密钥等关键业务数据的历史数据。修复前需要在测试环境进行验证。如果漏洞修复影响到账本数据，则应通过共识协议进行数据的修正。  应记录漏洞修复的审批记录、操作历史等信息，做到可审计、可追溯。 | 可满足  如果漏洞修复影响到账本数据，fabric支持发起新的交易通过智能合约对数据进行修正，前提是交易的发起者需满足智能合约预先设立的条件。  其它要求均属于管理机制。  合约有漏洞可以升级合约。  如果漏洞修复影响到账本数据，则应通过合理的背书进行数据的修正。 |  |
| 16.6 备份与恢复  账本数据应根据业务需要进行实时备份。密钥等关键数据应定期备份，防止因设备损坏等原因造成数据或密钥丢失。  应保证备份内容安全可靠，并制定数据、密钥等内容的恢复策略，定期进行恢复演练。 | 可满足  分布式的节点天然具备了账本数据的实时互备份，同时fabric也支持从账本加载数据进行网络的重启动，从而支持了网络的恢复，可进行演练。  定期备份属于管理机制。 |  |
| 16.7 应急预案管理  应规定统一的应急预案框架，具体包括启动预案的条件、应急组织构成、应急资源保障、事后教育和培训等内容。  应制定重要事件的应急预案，包括应急处理流程、系统恢复流程等内容。  重要事件的应急预案包括但不限于节点因故障而停止服务、节点账本数据损坏、节点版本升级失败。 | 可满足  此项归属于管理机制。 |  |
| 16.8 权限管理  使用分布式账本的接口应做好权限管理，防止未授权的调用。 | 满足  fabric的接口权限有tls连接权限以及交易证书权限两方面，可防止未授权的调用。 |  |
| 应对账本数据做好权限管理，敏感信息（如资产信息）应经授权查看或者使用。 | 可满足 |  |
| 除非特殊情况下，不应删除本地的账本信息。 | 可满足 |  |
| 每一次权限操作应写入日志，尤其是账本敏感信息的查看和使用，用于复查和审计。 | 可满足 |  |
| 16.9 议案机制  议案从提出到生效应保证透明公开公正，生效过程中对业务的平滑过渡，除非发生重大安全事故，避免采用所有节点同时停止服务进行升级的方式。  注：因分布式账本通常应用于多组织之间的协同运作，核心组织应满足以上运维要求，其余组织可视情况将分布式账本托管于核心组织。 | 可满足  去中心管理机制 |  |
| 17 治理机制 | | |
| 17.1 基本要求  金融分布式账本系统的治理机制原则上遵循 GB/T22239—2019 中三级以上的安全管理制度、安全管理机构和人员安全管理相关要求。在此基础上， 宜结合金融分布式账本特点，采用以下治理结构对相关管控重点进行安全治理。 | 按照GB/T22239—2019 中三级以上的安全管理制度、安全管理机构和人员安全管理相关要求，配置机构及人员，并满足培训及考核要求。 |  |
| 17.2 治理结构  17.2.1 组织架构  管理委员会作为金融分布式账本系统安全治理的决策层，由主任委员、副主任委员和委员组成，可以由监管机构或由占领导地位的单一机构创建，也可以由多个机构或用户联合组建。  管理委员会下设安全管理机构作为安全治理的管理层，负责日常安全管理工作的统筹和异常情况的应急处置统筹。  安全管理机构下设日常管理团队和应急管理团队作为安全治理的执行层，负责日常管理和应急管理的具体执行。  治理的组织架构见图 2 | 管理机制 |  |
| 17.2.2 管理职责  管理委员会负责审核和批准金融分布式账本系统的风险偏好、安全管理策略以及其他关键事项。  安全管理机构负责统筹金融分布式账本安全管理，组织协调相关成员落实各项管理职责。  定期组织开展相关风险事件梳理、归纳与风险评估工作，并按事件的风险等级制定相应的紧急应对策略。  日常管理团队负责如节点管理等日常安全管理工作，除包含传统安全管理人员如系统管理员、网络管理员、安全管理员等，还应包含金融业务风险管理人员以及分布式账本安全管理员。  应急管理团队负责在紧急事件发生时，应确保能进行必要的应急处置，并联系相应的人员进行处理，包括系统管理员、网络管理员、安全管理员及系统研发员。  当金融分布式账本采用第三方云部署方式时，应增加对云部署方运行资质和安全能力的评估准入规则，并定期进行监管。 | 管理机制 |  |
| 17.3 管控重点  17.3.1 节点管理  17.3.1.1 节点加入  金融分布式账本系统的新节点加入实施准入控制并经过相应的流程审核，包括对节点真实身份核查、对其他节点运行和记账的影响分析以及节点协议签署。 | 可满足  管理机制 |  |
| 17.3.1.2 节点退出  金融分布式账本系统的节点退出应经过管理委员会审议，可采用“自愿退出”、“投票剔除”等退出方式，节点退出后剩余节点数量应不低于最低安全节点数量要求。 | 满足 |  |
| 17.3.1.3 节点最低数量  应根据共识协议要求，设置最低安全数量的共识节点，制定“共识节点数量低于预警阈值”的应急预案。 | 可满足  管理机制  每个组织的peer保证基本可用性，以及Orderer Quorum数量需要满足 |  |
| 17.3.1.4 节点登记与验证  应对接入的节点进行信息登记和验证，记录节点基本软硬件信息及与机构的对应关系。  17.3.1.5 节点管理操作  应对节点管理操作进行权限控制和身份验证，记录所有节点管理的操作历史，做到可审计、可追溯。 | 管理机制 |  |
| 17.3.1.6 节点实现  应保证节点的实现具有一定的异构性， 包括“部署异构”、“硬件异构”和“软件异构”。 宜采用三种或三种以上的节点实现版本。 应遵循第 6 章“基础硬件”中物理安全和第 7 章“基础软件”中操作系统的相关要求。 | 可满足  硬件部分参考第6章  Fabric只有一种节点实现版本 |  |
| 17.3.1.7 共识节点管理  根据不同的实现，金融分布式账本系统的共识节点采用不同的管理策略。针对不允许动态调整共识节点的分布式账本，在系统初始化时，应结合共识节点拥有者身份等其他基础设施组件，确保节点参数设置正确，保证系统安全运行；针对可动态调整共识节点的分布式账本，应加强添加或删除共识节点的权限管理，确保由可信节点来执行共识节点的添加或删除操作，必要时可对共识节点进行分层分级管理。 | 满足 |  |
| 17.3.2 干预机制  17.3.2.1 用户干预  可通过设置相应的管理账户，允许第三方管理员向金融分布式账本系统发布操作指令，获得相关账户信息和账户内容，并调用指定账户对应的智能合约，对用户进行特殊干预，限制单个用户操作分布式账本系统；干预操作类型可以是冻结、解冻、暂停、恢复、 结束、开启和强制转移等。 | 可满足  通过管理机制结合具体实现来满足 |  |
| 17.3.2.2 节点干预  可通过设置相应的管理账户，允许第三方管理员对节点进行特殊干预，包括节点临时终止或应急强制终止，限制单个节点接入分布式账本系统或者限制其参与共识与记账；干预操作类型可以是冻结、解冻、暂停、恢复、结束、开启和强制转移等。 | 可满足  通过管理机制结合具体实现来满足 |  |
| 17.3.2.3 系统干预  在遭遇特殊突发事件时，可对金融分布式账本实施系统干预，如大规模节点关断或系统关断；系统干预应经过管理委员会审议，并明确系统干预后的恢复策略、恢复方案和具体措施等。 | 可满足  通过管理机制结合具体实现来满足 |  |
| 17.3.2.4 干预功能管理  具备干预功能的管理账户应同样处于金融分布式账本系统中；应对干预功能进行权限控制和身份验证，保证只有特定权限的用户，在身份验证通过的前提下才能进行干预操作；应将干预权限分散给多个指定用户，避免因设定少数特定权限用户而导致的集中度风险，并防范分散式病毒攻击的风险。 | 可满足  管理机制 |  |
| 17.3.2.5 干预行为记录  所有的干预行为应存储在系统中并与所有节点进行共识；应记录干预行为的操作历史，确保操作历史不可更改并且可以被查询，做到可审计、可追溯。 | 可满足 |  |