



基于多方隐私求并的卫星互联网跨域一致性接入控制

凌国玮¹, 唐鹏¹, 胡经纬², 邱卫东^{1*}

1. 上海交通大学计算机学院 (网络空间安全学院、密码学院), 上海 200240, 中国

2. School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore 637371, Singapore

* 通信作者. E-mail: qiuwd@sjtu.edu.cn

收稿日期: 2025-10-23; 修回日期: 2026-01-07; 接受日期: 2026-02-24; 网络出版日期: 2026-03-04

国家重点研发计划项目 (批准号: 2023YFB3106500)、国家自然科学基金 (批准号: 62441227) 和国家建设高水平公派研究生项目 (批准号: 202506230132) 资助

摘要 近地轨道卫星互联网正迈向跨管理域的协同供给范式. 其各运营域对于终端的接入放行通常依赖于各域黑/白名单的一致性同步. 但是, 如果各域的准入名单出现不一致的情况, 则有可能被敌手利用从而引发实际的攻击. 尽管以可靠广播互发名单虽可同步, 但由于白名单表征可接入标识、黑名单表征被撤销或受限标识, 敌手易据此复原终端在不同域的漫游轨迹, 带来了极其严重的隐私泄露风险. 为此, 本文提出多运营域黑/白名单安全求并协议: 各域仅提交本域标识集合, 协议仅公开全网有效并集而不泄露交集与标识出现次数. 方案采用对称与公钥原语并行设计, 基于 SM2 加法同态加密方案并结合椭圆曲线加法同态快速解密, 实现高效的黑/白名单安全求并, 同时可抵抗合谋攻击. 本文所提协议在半诚实模型下被证明是安全的. 实验结果表明本文方案比现有最先进方案更加高效. 比如, 当存在 8 个运营域时, 本文方案在广域网环境下比现有方案快大约 1.16 倍到 3.32 倍. 本文的完整代码实现可以在 GitHub 仓库中得到: <https://github.com/ShallMate/MPSU>.

关键词 跨域卫星互联网, 接入控制, 一致性同步, 隐私保护, 多方隐私集合求并

1 引言

近年来, 随着大规模近地轨道 (low Earth orbit, LEO) 卫星建设的快速推进, 卫星互联网正走向跨运营商、跨管理域的协同提供模式. 在工程实践中, 各运营域分别维护两类名单进行接入放行的快速判定^[1~3]: 允许接入白名单与吊销黑名单. 它们被直接映射为各运营域的允许或拒绝策略, 实现方式是通过远程用户拨号认证 (remote authentication dial in user service, RADIUS) 协议^[4].

但是, LEO 星座的链路容量与连通性随时间和空间强烈波动^[5]. 最近, Deng 等^[6]对“时变瓶颈链路”的测量与攻击分析表明, 控制与策略信息在跨地面站、跨域传播时会受到拥塞与断续覆盖的放大效应影响. 这导致了各运营域负责终端准入的黑/白名单的跨域同步延迟, 使同一标识在相邻管理域出现“此处拒绝、彼处放行”的分裂判定. 以黑名单为例, 当某运营域先行将一个证书或设备拉黑, 而相邻域的撤销检查与名单同步滞后, 终端

引用格式: 凌国玮, 唐鹏, 胡经纬, 等. 基于多方隐私求并的卫星互联网跨域一致性接入控制. 中国科学: 信息科学, 2026, 56: 539–559, doi: 10.1360/SSI-2025-0443

Ling G W, Tang P, Hu J W, et al. Consistent cross-domain access control for satellite Internet via multi-party private set union. Sci Sin Inform, 2026, 56: 539–559, doi: 10.1360/SSI-2025-0443

在域与域之间漫游时就可能出现前者拒绝、后者放行的分裂判定. 已有工作^[1,7]表明这类情况会带来“择域攻击”: 敌手(或已被撤销终端)在多运营域/多地面站协同的环境里, 有意识地接入那些同步速度相对滞后的运营域或地面站, 从而在某域已吊销或已拉黑的情况下, 改走另一域继续获得放行. 为了避免这类情况, 各个运营域需要以最快速度定期同步黑/白名单. 由于考虑到轨道时延、链路中断以及带宽受限等各方面的因素, 需要保证各运营域进行黑/白名单一致性同步所造成的开销尽可能的小^[8~11]. 最简单的同步方法是通过可靠广播互发各域的黑/白名单进行同步, 但是这会带来极其严重的隐私泄露问题.

首先, 在多运营域协同的卫星互联网中, 白名单本质上等同于运营侧的客户与资产清单(即哪些终端/证书被登记为“可接入”), 而黑名单等同于事件与封堵清单(即哪些标识被撤销/拉黑). 因此, 各运营域能根据名单条目与交叉匹配推断出哪些终端或证书在多域同时出现, 进而重建漫游轨迹与业务关系. 具体地说, 可以根据证书序列号、设备标识、地面站接入时间窗等特征, 推断同一终端在不同运营域、不同地面站与覆盖波束的多次出现, 从而重建跨域活动轨迹与通信关系, 进一步形成可链接与“谁在和谁互通”的侧写风险^[1,12]. 同时, LEO网络的高动态与可测性让这种侧写更易实现. 最近, Lai等^[9]在LEO测量与系统方面的工作就展示了敌手使用名单条目与链路侧迹象对齐的“拼图素材”, 更精细地推断出了各运营域的客户分布、业务增长与区域化覆盖策略. 因此, 除了考虑各运营域同步黑/白名单所造成的开销以外, 还应保证各运营域能够做到最小化该过程的隐私泄露.

为避免黑/白名单互发带来的隐私泄露风险, 同时做到对各运营域黑/白名单的高效一致性同步. 本文采用多方隐私集合求并(multi-party private set union, MPSU)协议作为跨运营域的黑/白名单一致性机制. 事实上, 黑/白名单的安全跨域同步本就是MPSU协议的经典应用场景^[13,14]. 具体地说, 各域仅在本域内部生成黑名单或白名单的标识集合, 然后作为参与方执行MPSU协议, 最终只产出全网一致的有效黑/白名单. 由于MPSU协议的性质, 各个运营域不会得到各域黑/白名单的交集结果以及出现在某一域的个数, 阻止了敌手进行侧写分析. 但是, 考虑到卫星互联网环境下轨道时延长、带宽受限等因素, 现有的MPSU协议^[15~18]仍然不能满足各个运营域快速同步黑/白名单的需求. 通常来说, 完全基于对称密码原语的MPSU协议^[15,17]在计算开销方面具有更大的优势. 但是, 如果运营域较多, 完全基于对称密码原语的MPSU协议无法很好扩展^[17,18]. 因此, 本文基于对称密码原语与公钥密码原语并行的方式构建MPSU协议. 此外, 现今最先进的基于公钥密码原语的MPSU协议^[18]使用Exp-ElGamal^[19]作为加法同态加密方案, 同时引入多方并行解密以及零密文预先生成的方式提升性能. 然而, Exp-ElGamal是一种椭圆曲线加法方案, 解密需要求解小指数离散对数, 而文献^[18]完全忽略了这一点. 不仅如此, 文献^[18]在明文上进行补零以区分随机元素和真实元素的操作只会进一步压缩明文空间, 使得解密性能极其低下.

综上所述, 面向卫星互联网跨运营域黑/白名单同步的接入控制场景, 存在三项核心挑战.

- 输出语义上, 需要保证所有运营域获得一致的并集结果以实现统一策略, 而非仅由单一接收方获得输出; 这要求协议在多方输出条件下仍能保持强合谋下的隐私边界.
- 隐私目标上, 除并集本身外, 还需抑制由中间结果推断“元素归属域”与“出现频次”等关联信息泄露.
- 工程约束上, 卫星互联网网络波动显著, 端到端完结时延对同步开销高度敏感. 为实现可部署的跨域协同, 协议必须以完成时间为首要优化目标, 并重点降低同态加密相关计算(尤其是解密阶段)的端到端开销.

基于以上的核心挑战以及现有方案的各种限制, 本文提出了一个新的高效多运营域黑/白名单安全求并协议, 以满足卫星互联网环境下各运营域黑/白名单的高效隐私保护同步, 进而实现全网一致的接入控制. 该协议在公钥密码与对称密码两方面相比于之前的方案进行了如下的优化. 在公钥密码技术方面: 基于国密SM2加法同态加密方案^[20], 将其扩展为多密钥版本, 并结合当下最高效的椭圆曲线加法同态解密FastECDLP^[21], 提升现今最高效的MPSU协议^[17,18]的性能; 而在对称密码技术方面: 提出了新的成员不经意传输(membership oblivious transfer, MOT)协议与混淆并解密(Shuffle&Decrypt)协议, 进一步提升方案的性能.

本文的主要贡献可被总结如下.

- 本文基于MPSU提出了一个高效的多运营域黑/白名单安全求并协议. 该协议可保证各运营域在最小化隐私泄露的前提下迅速地各域黑/白名单的整合, 避免各域名单不一致所带来的潜在攻击. 本文所提所有协议

在半诚实模型下被证明是安全的,同时可抵御任意的合谋攻击。

- 本文分别提出了新的 MOT 协议与 Shuffle&Decrypt 协议作为方案的子模块,目的是适配所提方案同时进一步提升方案的整体性能。
- 本文结合多密钥国密 SM2 加法同态加密方案^[20]与其快速解密算法^[21],降低了现有协议^[16~18]底层使用 Exp-ElGamal^[19]加法同态加密方案所带来的大量计算开销。特别是在解密部分的性能会得到较大的提升,因此相比于现有协议具有更强的可扩展性。
- 实验结果表明,当运营域数量为 8,本文方案在局域网环境下,相较于现如今最高效的方案^[16~18],运行时间分别提升约 3.32 倍、1.60 倍和 1.16 倍;在广域网环境下,对应提升约 3.52 倍、1.38 倍和 1.12 倍。同时,本文还证明了方案在有较多运营域 (50 个) 参与求并时,仍可取得更优的性能。

2 相关工作

本节主要对本文所涉及的两方面研究内容的现有工作进行简要回顾。其一是卫星互联网安全相关研究,其二是 MPSU 协议的研究进展。

2.1 卫星互联网安全相关研究

目前关于卫星互联网安全的研究主要围绕物理层认证与抗欺骗、隐私保护与位置匿名、网络/协议层攻防与入侵检测 (含时变瓶颈与链路泛洪),以及多方协同与可持续治理等方向展开。Topal 等^[22]提出了一种新的物理层认证方法,应用于 LEO 卫星之间的星间通信链路。在所提出的认证方法中,多个接收卫星通过将接收到的多普勒频移测量值与合法发射端的参考运动信息进行比较,从而验证发射端的身份,并在选定的判决规则下融合各自的决策。Chu 等^[23]提出一种结合非正交多址接入和多波束预编码技术的 LEO 卫星物联网网络,可以支持海量设备接入并抑制波束间和卫星间干扰。他们通过大量的仿真结果证明了该算法的有效性和鲁棒性。Jedermann 等^[24]提出了一种被动攻击方法揭示了 LEO 卫星通信下的用户位置隐私隐患。此攻击方法只需在地面观察“漂移”的通信卫星的下行信号,就可以大规模利用这些宽波束信号来推测用户所处的区域。此外,他们还讨论了该攻击的可迁移性,并提出了潜在的应对措施。Chen 等^[25]认为 LEO 卫星网络的可持续扩展受制于恶劣、拥挤且不均衡的物理环境,主要体现在两方面:横向扩展限制,即轨道拥挤下的卫星安全间距约束;以及纵向扩展限制,即容量供给的均一性与全球需求分布不均衡之间的错配。他们认为,以往的研究忽视了这些物理扩展瓶颈,因此提出了一个“信息-物理协同设计框架”加以解决这个问题。Oh 等^[2]指出在 LEO 卫星星座数量很少 (如由资金不足的国家所建设的独立卫星) 时,这些独立星座往往是不切实际且浪费资源的,同时还会因为低轨卫星的轨道动力学而不可持续。因此,他们提出了一种多方去中心化的星座模式,要求不同参与方各自向共享星座贡献少量卫星。他们表示这种多方星座在部分参与方退出时仍具有鲁棒性,并且能够减少经济成本、容量浪费以及轨道占用。Wigchert 等^[26]研究了 LEO 卫星系统中的欺骗攻击检测问题。他们通过无人机在不同高度注入伪造的 Iridium 消息,验证了空中欺骗攻击的可行性,并提出基于物理层信号图像化与自编码器的异常检测方法,并表示该方法能够有效检测不同高度下的欺骗攻击。Deng 等^[6]系统性地研究了 LEO 卫星网络中由时变瓶颈链路引发的安全风险。他们发现瓶颈链路可能被利用实施链路泛洪攻击 (link flooding attack, LFA),并提出了风险分析工具 SKYFALL,用于模拟不同 LFA 行为并评估对网络服务的影响。他们的研究表明,若攻击者操控部分用户终端持续拥塞瓶颈链路,合法流量吞吐量可被削减约 3.4 倍。Classen 等^[27]对苹果手机中集成的专有卫星通信协议进行了逆向工程,分析了其安全性与隐私特性。他们基于仿真平台揭示了协议与基础设施设计中的问题,例如精简协议消息缺乏完整性保护、需依赖互联网初始化等。此外,该研究还展示了多种绕过限制的方法,包括滥用位置共享发送任意短信以及在区域受限国家通过卫星发送 iMessage,从而表明该协议存在被利用以规避审查和运营商控制的风险。Smailes 等^[28]提出了 SatIQ 系统,通过捕捉发射机硬件在物理层信号中的细微差异实现 LEO 卫星下行链路的认证。该系统基于孪生神经网络与自编码器提取高效的信号特征编码,并在 1029 万条 Iridium 卫星消息上进行了验证。他们的实验结果表明, SatIQ 在不同发射机和长时间运行下依然保持稳定性能,并在对

抗条件下证明其能够有效抵御欺骗与重放攻击. He 等^[29]提出了 STINIDF 框架, 利用联邦学习与条件扩散模型结合差分隐私机制, 解决了卫星-地面一体化网络入侵检测中的非独立同分布数据挑战. 他们在自主构建的数据集(包括信号干扰、用户数据报协议洪泛与干扰攻击等场景)上进行了测试. 结果表明 STINIDF 在各方面指标取得了较好的效果, 同时在隐私保护与入侵检测性能之间实现了有效平衡. McAmis 等^[30]研究了大学团队开发的小型卫星安全性问题. 他们通过访谈与代码审计发现, 这类项目普遍存在可被地面攻击者利用的漏洞, 安全实践水平差异极大. 他们的研究指出, 不安全的小型卫星带来显著风险, 并提出了改进业余组织及更广泛应用场景中小卫星安全性的实用建议.

2.2 多方隐私集合求并研究进展

本节对 MPSU 协议的研究进展进行一个简要的回顾. Kissner 等^[31]基于加法同态加密计算多项式的思想提出了第一个 MPSU 协议. 该协议的问题是大量的加法同态运算以及高次多项式计算, 在实际应用中完全不可接受. Frikken^[32]通过降低多项式的次数改进了文献[31]. 但是, 由于必须在密文多项式上执行多点求值, 加法同态运算的数量在集合规模上依旧是平方复杂度. Blanton 等^[33]提出了一个基于不经意排序和通用安全多方计算的 MPSU 协议. 但是, 该协议对通用 MPC 操作的高度依赖导致其效率较差. Vos 等^[34]提出了基于位向量表示的 MPSU 协议, 参与方通过对位向量执行安全“或”运算来共同计算并集, 并使用 Exp-ElGamal^[19]进行加法同态加密. 近几年提出的高效 MPSU 协议^[15~18]均基于秘密共享的隐私成员测试(secret-shared private set membership test, SSPMT)与不经意传输(oblivious transfer, OT). 第一个这样的 MPSU 协议是由 Liu 等^[15]提出的, 该协议主要基于对称密码原语, 比以往工作快了两个数量级以上. 该协议的问题在于事实上在标准半诚实模型下并不安全, 因为他们假设了得到并集的一方不会与其他的任一参与方进行合谋. 随后, Gao 等^[16]提出了一个基于公钥密码原语的 MPSU 协议, 该协议相比于文献[15]的 MPSU 协议来说达到了更优的复杂度. 但是, Dong 等^[17]很快指出了 Gao 等^[16]的协议在面对任意串通时事实上也并不安全. 同时, 他们给出了一个实际攻击, 证明它与文献[15]的 MPSU 协议一样易受共谋攻击. 同时, Dong 等^[17]也相应地分别提出了基于对称密码原语与公钥密码原语的两个 MPSU 协议, 同时容忍任意的合谋攻击. 最近, Gao 等^[18]观察到现有的 MPSU 协议的几个问题, 比如难以应用到大量参与方的场景、运行时参与方之间需要互相等待同步严重浪费资源、交互次数过多轮数过多等. 针对这些问题, 他们提出了一种基于 Exp-ElGamal^[19]的高效 MPSU 协议, 并在实验中证明了该协议比文献[16, 17]的 MPSU 协议都要快. 本文也观察到了文献[18]的 MPSU 协议的一些问题. 比如, 该协议使用 Exp-ElGamal 作为加法同态加密方案, 事实上需要在解密过程中求解小指数的椭圆曲线离线对数问题. 但是该协议完全忽略了这一点, 这会导致一旦集合元素长度稍大, 解密的性能将会不可接受^[21]. 同时, 他们还使用“补零”的方式以区分有效输出与随机值, 更进一步缩短了可用的有效元素位长. 此外, 该协议事实上需要大量的离线计算以生成在线阶段的零密文以换取在线阶段的优势. 这些问题都导致了文献[18]的 MPSU 协议难以直接用于卫星互联网环境下各运营域黑/白名单的高效求并. 因此, 本文基于 SM2 加法同态加密方案^[20], 结合椭圆曲线加法同态快速解密算法^[21], 构造了新的 MPSU 协议, 以适应本文的应用需求.

3 预备知识

3.1 符号定义

除非有特殊说明, 本文所有符号均遵循本节定义. λ 表示统计安全参数, 而 κ 表示计算安全参数. $\mathcal{P}_i$ 表示第 i 位协议的参与方/运营域, 设一共有 m 个运营域. X_i 表示第 i 位运营域进行接入控制的黑/白名单集合. 为了方便描述, 假设 X_i 的基数相等, 均为 n . $[n]$ 代表从 1 到 n 的整数集合 $\{1, \dots, n\}$. $[a, b]$ 代表 $\{a, a+1, \dots, b\}$. p 表示一个大素数, ℓ 是同态加密方案的明文最大位长.

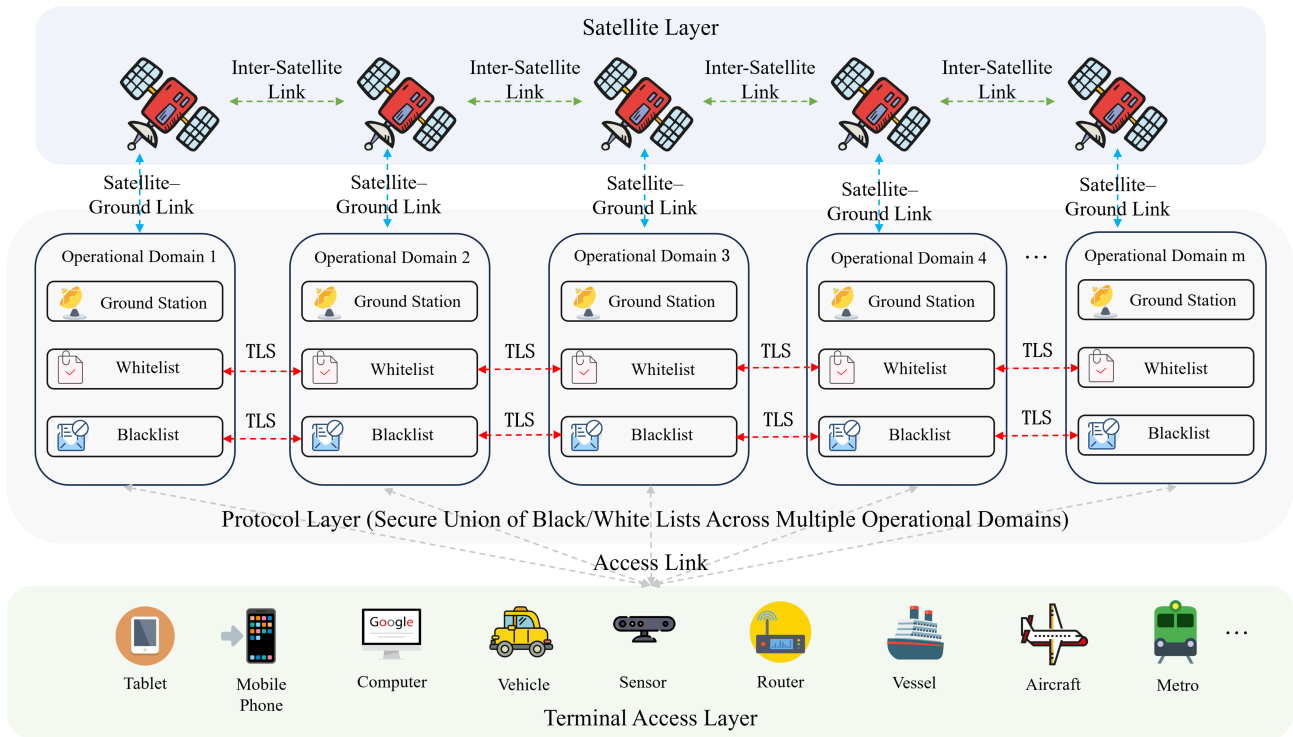


图1 卫星互联网多运营域接入控制系统架构。

Figure 1 System architecture of multi-operator domain access control in satellite Internet.

3.2 系统模型

系统架构. 本文的系统架构如图1所示, m 个运营域定期地执行黑/白名单的同步任务. 执行完毕后通过联合生成的黑/白名单进行终端的接入控制.

安全模型. 本文采用安全多方计算 (multi-party computation, MPC) 常用的基于模拟的范式^[35] 分析协议的安全性. 本文所提协议被证明在半诚实模型是安全的. 具体地说, 协议的参与方会诚实地执行协议, 但是会根据他得到的信息尝试推断他人的数据. 在一个有 m 个参与方的协议中, 本文最多允许敌手能够攻破其中的 $m-1$ 个参与方. 即本文所提所有协议均可抵抗 $m-1$ 合谋攻击.

定义1 (多方 MPC 协议的半诚实安全性) 令 $\mathcal{F} : (\{0,1\}^*)^m \rightarrow (\{0,1\}^*)^m$ 是 m 方协议 Π 的理想功能. X_i 分别代表第 i 位参与方 $\mathcal{P}_i$ 的输入. 令 $X = \{X_i\}_{i \in [m]}$. $\mathcal{F}_i(X)$ 代表 $\mathcal{P}_i$ 的理想输出. 对于任意基数小于等于 $m-1$ 的集合 $\mathcal{C} = \{i_1, \dots, i_t\} \subseteq [m]$, 令 $X_{\mathcal{C}}$ 表示 $\{X_{i_1}, \dots, X_{i_t}\}$, $\mathcal{F}_{\mathcal{C}}$ 表示 $\{\mathcal{F}_{i_1}(X), \dots, \mathcal{F}_{i_t}(X)\}$. 参与方 $\mathcal{P}_i$ 在执行协议 Π 时所看到的视图表示为 $\text{VIEW}_i(X)$. 对于任意的 $\mathcal{C}$, $\text{VIEW}_{\mathcal{C}}(X)$ 表示 $\mathcal{P}_{i_t}$ 们的联合视图 $\{\text{VIEW}_{i_1}(X), \dots, \text{VIEW}_{i_t}(X)\}$. 如果称一个协议 Π 在半诚实模型下安全地实现了确定性的理想功能 $\mathcal{F}$ 且抗至多 $m-1$ 合谋, 那么能构造一个概率多项式时间模拟器 $\mathcal{S}$, 对于任意的 $\mathcal{C} \subseteq [m]$, 均可使得

$$\mathcal{S}(\mathcal{C}, X_{\mathcal{C}}, \mathcal{F}_{\mathcal{C}}) \equiv \text{VIEW}_{\mathcal{C}}(X), \quad (1)$$

其中 $\equiv$ 代表计算不可区分.

适用边界与讨论. 本文采用半诚实模型刻画跨运营域协同中的“诚实且好奇”的现实约束: 参与方按协议执行但会尝试从可见信息推断他域名单. 本文不覆盖主动偏离协议的恶意行为, 例如输入投毒 (提交畸形/策略性构造的名单以影响输出)、选择性中止 (在观察到中间结果后退出)、畸形密文注入与拒绝服务等. 在工程部署中, 上述风险可通过输入授权与格式校验、签名审计日志、超时重试与速率限制等机制进行缓解; 在密码学层面, 将本文方案扩展至恶意安全需要为关键子过程引入一致性与可验证性组件, 我们在第6节中给出扩展路线.

$$\mathcal{F}^{\text{OT}}$$

The sender $\mathcal{P}_1$ inputs two messages (m_0, m_1) , and the receiver $\mathcal{P}_2$ inputs a choice bit $b \in \{0, 1\}$. The receiver obtains m_b , while the sender learns no information about b .

图 2 2 选 1 的 OT 协议的理想功能 $\mathcal{F}^{\text{OT}}$.
Figure 2 Ideal functionality $\mathcal{F}_{\text{OT}}$ for 1-out-of-2 OT.

$$\mathcal{F}^{\text{MPSU}}$$

Each party $\mathcal{P}_{i \in [m]}$ holds a private set X_i , and $\mathcal{P}_1$ acts as the designated receiver. All parties $\mathcal{P}_{i \in [m]}$ obtain $U = \bigcup_{i=1}^m X_i$.

图 3 MPSU 协议的理想功能 $\mathcal{F}^{\text{MPSU}}$.
Figure 3 Ideal functionality $\mathcal{F}^{\text{MPSU}}$ for the MPSU protocol.

3.3 不经意传输

不经意传输 (oblivious transfer, OT)^[36] 是 MPC 领域的一个基本原语, 其理想功能如图 2 所示. 在标准设置 (2 选 1 的 OT 协议) 中, 发送方输入两个等长消息 (m_0, m_1) , 同时接收方输入一个选择位 $b \in \{0, 1\}$. 在协议执行结束时, 接收方得到 m_b 同时不会得到任何关于 m_{1-b} 的信息, 发送方对于 b 一无所知. Ishai 等^[37] 提出了 OT 扩展 (OT extension, OTE) 技术, 这能使得双方通过 κ 个 OT 基本实例通过高效的对称密码操作换取大量的 OT 实例.

3.4 多方隐私集合求并

多方隐私集合求并 (multi-party private set union, MPSU) 是一种最近逐渐受到关注的 MPC 原语. 本文将设计一种新的 MPSU 协议, 并将其应用到卫星互联网跨域一致性接入控制中, 使得各个运营域能够进行高效的隐私保护黑/白名单同步. MPSU 协议的理想功能如图 3 所示. 它能够使得多个参与方 $\mathcal{P}_i$ 输入自身的集合 X_i , 最后参与方得到并集 $U = \bigcup_{i=1}^m X_i$. 本文不再沿用现有的 MPSU 协议^[15~18] 只有一个参与方收到输出的设置, 而是让所有的参与方都得到输出. 这是因为跨域一致性接入控制要求各域对同一标识给出一致的放行或拒绝决策, 因而各域必须共享一致的全网有效名单. 此外, 本文也注意到, 在某些跨主体协同场景中, “并集本身” 可能被视为敏感信息外溢^[38]. 本文的设计目标聚焦于阻止交集与出现次数等关联信息泄露, 但并不消除输出并集带来的业务层可见性. 当业务侧不允许全量共享时, 本文协议仍可作为同步基础层使用, 并通过部署形态控制输出面, 例如仅向策略控制点输出、按需查询式输出、结合访问控制与审计对输出集合进行受控披露. 上述方向不改变本文核心机制, 但需要额外的系统层策略约束与权限治理.

安全目标与泄露. 本文考虑半诚实敌手与至多 t (可取 $m - 1$) 合谋. 协议的显式输出为并集 U . 除 U 以及合谋方自身输入 X_c 所必然蕴含的信息外, 协议不应泄露额外关联信息. 具体而言, (1) 归属隐私: 对任意 $u \in U$, 合谋方除判断 u 是否属于自身输入外, 不应进一步确定 u 由哪些非合谋参与方贡献 (除非该信息已可由 U 与 X_c 直接推出); (2) 频次隐私: 对任意 $u \in U$, 合谋方不应获知 u 在非合谋参与方中的出现次数 (除非该信息已可由 U 与 X_c 直接推出). 当 $m = 2$ 时, 上述两类隐私在 “仅存在一个非合谋方” 的情形下退化为不可避免泄露的边界情况.

3.5 基于国密 SM2 的加法同态加密

基于国密 SM2 的加法同态加密是 Tang 等^[20] 提出的基于椭圆曲线的加法同态加密方案. 该方案保留了原本标准方案中的推荐曲线参数和密钥生成算法, 仅对原 SM2 方案的加解密算法作出简单调整, 使其具备加法同态性质. 它拥有比著名的 Exp-ElGamal^[19] 以及 Paillier^[39] 更优秀的性能. 此外, 它还可以与现今最高效的求解椭圆曲线离散对数算法 FastECDLP^[21] 结合, 以实现高效的解密效率与较大的明文空间. 目前, 基于国密

SM2 的加法同态加密方案结合 FastECDLP 已经被集成到由蚂蚁集团所开发的底层密码学库 YACL 中¹⁾. 同时, FastECDLP 已经被应用于实际的区块链项目 XELIS 中²⁾, 其性能比传统的 BSGS 算法^[40] 高出两到三个数量级. 因此, 考虑到基于国密 SM2 的加法同态加密的高效性以及面向网络空间安全自主可控的战略需求, 本文选用基于国密 SM2 的加法同态加密作为底层加法同态加密组件. 为了适配多方计算黑/白名单并集, 本文还需要将国密 SM2 加法同态加密扩展为多密钥同态加密方案. 设有 m 个参与方 $\mathcal{P}_{i \in [m]}$, 该方案由以下算法组成.

- **KeyGen**(λ, m): 输入安全参数 λ 和参与方个数 m , 各个参与方随机选择各自的部分私钥 $sk_i \leftarrow \mathbb{Z}_p$, 并联合计算公钥: $pk = \sum_{i=1}^m sk_i * G$, 其中 G 是椭圆曲线的基点, 公钥 pk 被公开, 各方保护 sk_i 不被泄露.
- **Enc**(pk, x): 任一参与方 $\mathcal{P}_i$ 可输入明文 $x \in \{0, 1\}^\ell$, 随机选择 $r \leftarrow \mathbb{Z}_p$, 计算密文

$$c = (c_1, c_2) = (r * G, x * G + r * pk). \quad (2)$$

- **ParDec**(sk_i, c): 任一参与方 $\mathcal{P}_i$ 可输入部分私钥 sk_i 对密文 $c = (c_1, c_2)$ 进行部分解密. 即 $c' = (c_1, c'_2) = (c_1, c_2 - c_1 * sk_i)$.
- **FulDec**($sk_{i \in [m]}, c$): 所有参与方共同完成对密文 $c = (c_1, c_2)$ 的解密, 即

$$v = c_2 - \sum_{i=1}^m c_1 * sk_i. \quad (3)$$

进而调用 $x = \text{FastECDLP}(v)$ 得到明文. 注意, 由于椭圆曲线加法同态加密算法在解密时天然需要解密离散对数, 因此明文 x 必须在预先设置好的明文空间 $\{0, 1\}^\ell$ 中, 否则 FastECDLP 返回 $\perp$, 代表解密失败.

- **ReRand**(c): 对密文 $c = (c_1, c_2)$ 进行重新随机化, 随机选择 $r \leftarrow \mathbb{Z}_p$, 计算重随机化密文 $c' = (c_1 + r * G, c_2 + r * pk)$.
- **HomoAdd**(c, c'): 输入两个密文 $c = (c_1, c_2), c' = (c'_1, c'_2)$, 进行同态加法运算则是把密文部分直接相加即可, 即加法同态后的密文则是 $(c_1 + c'_1, c_2 + c'_2)$.
- **ScaMul**(c, q): 明文乘密文操作. 输入明文 x 的密文 $c = (c_1, c_2)$ 以及标量 $q \in \mathbb{Z}_p$, 生成 $x \cdot q$ 的密文 $c' = (q * c_1, q * c_2)$.

基于国密 SM2 的加法同态加密方案已经被证明具有 IND-CPA 安全性^[20], 扩展为多密钥方案后本质上方案的加解密算法流程并没有被修改, 也不会对安全性造成影响. 因此, 本文仍称上述多密钥同态加密方案为“基于国密 SM2 的加法同态加密方案”, 并省略其关于正确性以及安全性的讨论.

3.6 秘密共享的隐私成员测试

秘密共享的隐私成员测试 (secret-shared private membership test, SSPMT) 协议的理想功能如图 4 所示, 它是现有的高效 MPSU 协议^[15~18] 的重要组成部分. 它可以并行地输出 $\mathcal{P}_2$ 的元素 y_i 是否属于 $\mathcal{P}_1$ 的集合 X_i 的秘密共享. 本文仍然沿用这一原语进行 MPSU 协议的构造. 这一原语的构造方法一般是通过不经意可编程伪随机函数 (oblivious programmable pseudorandom function, OPPrf) 协议结合两方判等电路得到, 具体可查阅文献^[15~18]. 本文直接使用目前最高效的 OPPrf 协议^[41] 以及 SecretFlow SPU 库^[42] 中 SEMI2K 协议簇实现该协议.

3.7 布谷鸟哈希

布谷鸟哈希^[43] 是一种哈希表数据结构, 它使用 k 个哈希函数 $h_j: 0, 1^* \rightarrow [B]$ 来确定 n 个元素的位置在 B 个桶中的位置. 为插入新元素 e , 布谷鸟哈希会将其放入某个候选位置 $h_j(e)$, 其中 $j \in [k]$. 若目标位置已被另一元素 e' 占据, 则现有元素 e' 会被逐出, 并重新安置到其另一个候选位置 $h_i(e')$, 其中 $i \in [k]$ 且 $i \neq j$. 这一过程递归进行, 直至所有元素成功放置, 或达到预设的重定位阈值. 若在耗尽允许的逐出次数后仍插入失败, 则将最

1) <https://github.com/secretflow/yacl/tree/main/examples/hesm2>.

2) <https://docs.xelis.io/features/wallet/ecdlp>.

$\mathcal{F}^{\text{SSPMT}}$

Consider two parties $\mathcal{P}_1$ and $\mathcal{P}_2$.

- $\mathcal{P}_1$ inputs B sets $\{X_i\}_{i \in [B]}$, and $\mathcal{P}_2$ inputs B elements $\{y_i\}_{i \in [B]}$.
- For each $i \in [B]$, the protocol generates $b_{1,i} \in \{0, 1\}$ returned to $\mathcal{P}_1$ and $b_{2,i} \in \{0, 1\}$ returned to $\mathcal{P}_2$.
- If $y_i \in X_i$, then $b_{1,i} \oplus b_{2,i} = 1$; otherwise, $b_{1,i} \oplus b_{2,i} = 0$.

图 4 SSPMT 协议的理想功能 $\mathcal{F}^{\text{SSPMT}}$.

Figure 4 Ideal functionality $\mathcal{F}^{\text{SSPMT}}$ for the SSPMT protocol.

 $\mathcal{F}^{\text{MOT}}$

Consider two parties: the receiver $\mathcal{P}_1$ and the sender $\mathcal{P}_2$.

- $\mathcal{P}_1$ inputs B sets $\{X_i\}_{i \in [B]}$, and $\mathcal{P}_2$ inputs B elements $\{y_i\}_{i \in [B]}$ together with B message pairs $\{(m_{i,0}, m_{i,1})\}_{i \in [B]}$.
- For each $i \in [B]$, the protocol outputs a message $m_i \in \{m_{i,0}, m_{i,1}\}$ to $\mathcal{P}_1$.
- If $y_i \notin X_i$, then $m_i = m_{i,0}$; otherwise, $m_i = m_{i,1}$.

图 5 MOT 协议的理想功能 $\mathcal{F}^{\text{MOT}}$.

Figure 5 Ideal functionality $\mathcal{F}^{\text{MOT}}$ for the MOT protocol.

后的元素存入单独的备用区. 根据 Pinkas 等^[44]的估计, 当 $B = 1.27n$ 且 $k = 3$ 时, 布谷鸟哈希无需额外的备用区, 插入失败的概率小于 $2^{-\lambda}$.

4 本文方案

本节对所提的多运营域黑/白名单安全求并协议进行一个详细的描述, 包括基础模块的定义与构造、宏观维度的构造思想, 以及协议执行的具体细节. 此外, 对于本节给出的所有协议, 都给出了在半诚实模型下的安全性证明.

4.1 基础组件

在介绍正式的协议的具体构造前, 需要给出一些需要用到的基础组件, 如成员不经意传输 (membership oblivious transfer, MOT) 协议、混淆并解密 (Shuffle&Decrypt) 协议.

4.1.1 成员不经意传输 (MOT)

MOT 协议的理想功能如图 5 所示, 最开始由 Gao 等^[16]提出. 在该协议中, 有两个参与方 $\mathcal{P}_1$ 与 $\mathcal{P}_2$, 其中 $\mathcal{P}_1$ 持有 B 个集合 $\{X_i\}_{i \in [B]}$, 而 $\mathcal{P}_2$ 持有 B 个元素以及相应的消息对 $\{(m_{i,0}, m_{i,1})\}_{i \in [B]}$. 对于每一个 $i \in [B]$, 如果 $y_i \notin X_i$, 则 $\mathcal{P}_1$ 得到 $m_i = m_{i,0}$, 否则得到 $m_i = m_{i,1}$. 需要注意的是, 为了适配本文后续的协议构造, 本文所呈现的 MOT 协议与文献 [16] 中给出的理想功能并不相同. 在文献 [16] 中, 如果 $y_i \in X_i$, 那么 $\mathcal{P}_1$ 得到 $m_i = m_{i,0}$, 否则得到 $m_i = m_{i,1}$, 这与本文在图 5 中所描述的理想功能是相反的.

想要实现 MOT 协议, 最常用的方式是通过调用安全的 SSPMT 协议, 然后利用 OT 扩展 (OT extension, OTE) 来实现大量的 $\mathcal{F}^{\text{OT}}$. 本文也是基本采取的此类构造方式. 本文所提的 MOT 协议的具体执行流程如协议 1 所示. 首先, $\mathcal{P}_1$ 与 $\mathcal{P}_2$ 分别输入 $\{X_i\}_{i \in [B]}$ 与 $\{y_i\}_{i \in [B]}$ 调用 SSPMT 协议的理想功能 $\mathcal{F}^{\text{SSPMT}}$ (见图 4). 由 $\mathcal{F}^{\text{SSPMT}}$ 可知, 对于每一个 $i \in [B]$, 如果 $y_i \in X_i$, 则 $b_{1,i} \oplus b_{2,i} = 1$ 成立, 否则 $b_{1,i} \oplus b_{2,i} = 0$ 成立. 然后, 双方继续调用 B 次 $\mathcal{F}^{\text{OT}}$ 来完成协议. 具体过程是: 对于 $i \in [B]$, $\mathcal{P}_1$ 输入 $b_{1,i}$, $\mathcal{P}_2$ 的输入根据 $b_{2,i}$ 确定, 如果 $b_{2,i} = 0$, 那么输入为 $(m_{i,0}, m_{i,1})$, 否则为 $(m_{i,1}, m_{i,0})$. $\mathcal{P}_1$ 取得 m_i , 最后 $\mathcal{P}_1$ 得到的 $\mathcal{F}^{\text{OT}}$ 的输出就是 MOT 协议的输出.

MOT 协议正确性分析. 根据 $\mathcal{F}^{\text{SSPMT}}$ 可知, 对于 $i \in [B]$, 如果 $y_i \in X_i$, 则 $b_{1,i} \oplus b_{2,i} = 1$ 成立, 否则

协议 1 成员不经意传输 (MOT).

输入: 接收方 $\mathcal{P}_1$ 输入 B 个集合 $\{X_i\}_{i \in [B]}$, 发送方 $\mathcal{P}_2$ 输入 B 个元素 $\{y_i\}_{i \in [B]}$ 与 B 个消息对 $\{(m_{i,0}, m_{i,1})\}_{i \in [B]}$;

参数: OT 协议的理想功能 $\mathcal{F}^{\text{OT}}$, SSPMT 协议的理想功能 $\mathcal{F}^{\text{SSPMT}}$;

1: $\mathcal{P}_1$ 输入 $\{X_i\}_{i \in [B]}$, $\mathcal{P}_2$ 输入 $\{y_i\}_{i \in [B]}$ 调用 $\mathcal{F}^{\text{SSPMT}}$; 对于每一个 $i \in [B]$, $\mathcal{P}_1$ 取得 $b_{1,i} \in \{0, 1\}$, $\mathcal{P}_2$ 取得 $b_{2,i} \in \{0, 1\}$, 使得如果 $y_i \in X_i$, 则 $b_{1,i} \oplus b_{2,i} = 1$ 成立, 否则 $b_{1,i} \oplus b_{2,i} = 0$ 成立;

2: **for** $i \in [B]$ **do**

3: $\mathcal{P}_1$ 作为接收方输入 $b_{1,i}$, $\mathcal{P}_2$ 作为发送方调用 $\mathcal{F}^{\text{OT}}$, 其中 $\mathcal{P}_2$ 的输入根据 $b_{2,i}$ 确定, 如果 $b_{2,i} = 0$, 那么输入为 $(m_{i,0}, m_{i,1})$, 否则为 $(m_{i,1}, m_{i,0})$. $\mathcal{P}_1$ 取得 m_i ;

4: **end for**

输出: 接收方 $\mathcal{P}_1$ 输出 $\{m_i\}_{i \in [B]}$;

 $\mathcal{F}^{\text{Shuffle\&Decrypt}}$

Each party $\mathcal{P}_{i \in [m]}$ holds a random permutation $\pi_i : [N] \rightarrow [N]$ and a partial secret key $\text{sk}_i \in \mathbb{Z}_p$. Party $\mathcal{P}_1$, acting as the receiver, holds a set of additively homomorphic ciphertexts under the homomorphic SM2 cryptosystem, denoted by $C = \{c_j\}_{j \in [N]}$, where $c_j = \text{HomoSM2.Enc}(\text{pk}, x_j)$ and the public key is defined as $\text{pk} = \sum_{i=1}^m \text{sk}_i * G$. The ideal functionality outputs $\{x_{\pi(j)}\}_{j \in [N]}$ to $\mathcal{P}_1$, where $\pi = \pi_m \circ \pi_{m-1} \circ \dots \circ \pi_1$.

图 6 Shuffle&Decrypt 协议的理想功能 $\mathcal{F}^{\text{Shuffle\&Decrypt}}$

Figure 6 Ideal functionality Shuffle&Decrypt for the Shuffle&Decrypt protocol

$b_{1,i} \oplus b_{2,i} = 0$. 因此, 当 $b_{2,i} = 0$ 时, 可知 $b_{1,i} \oplus b_{2,i} = b_{1,i}$ 成立. 因此, 当 $b_{1,i} \oplus b_{2,i} = 0$, 即 $y_i \notin X_i$ 时, $\mathcal{P}_1$ 会得到 $m_{i,0}$, 因为此时 $\mathcal{P}_1$ 的输入是 $(m_{i,0}, m_{i,1})$. 同理, 当 $b_{1,i} \oplus b_{2,i} = 1$, 即 $y_i \in X_i$ 时, $\mathcal{P}_1$ 会得到 $m_{i,1}$. 继续讨论当 $b_{2,i} = 1$ 的情况, 注意此时 $b_{1,i} \oplus b_{2,i} = b_{1,i} \oplus 1$, 且 $\mathcal{P}_1$ 的输入是 $(m_{i,1}, m_{i,0})$. 因此, 如果 $b_{1,i} = 0$, 那么 $b_{1,i} \oplus b_{2,i} = 1$, 即代表 $y_i \in X_i$, 此时 $\mathcal{P}_1$ 会得到 $m_{i,1}$, 否则会得到 $m_{i,0}$. 综上, 不管是哪种情况, 都能符合如果 $y_i \notin X_i$, 则 $\mathcal{P}_1$ 收到 $m_{i,0}$ 成立, 否则收到 $m_{i,1}$. 协议 1 的正确性分析完毕.

定理1 (MOT 安全性) 协议 1 在半诚实模型下安全地实现了图 5 中的理想功能 $\mathcal{F}^{\text{MOT}}$.

证明 由于此协议的交互过程是通过调用理想的 $\mathcal{F}^{\text{SSPMT}}$ 与 $\mathcal{F}^{\text{OT}}$ 实现. 因此, 在理想世界中, 不管是模拟腐败的 $\mathcal{P}_1$ 的视图还是模拟腐败的 $\mathcal{P}_2$ 的视图, 仅需调用理想的 $\mathcal{F}^{\text{SSPMT}}$ 与 $\mathcal{F}^{\text{OT}}$ 即可.

4.1.2 混淆并解密 (Shuffle&Decrypt)

Shuffle&Decrypt 协议是指有一个参与方 $\mathcal{P}_1$ 持有一个加法同态加密的密文集合 $C = \{c_j\}_{j \in [N]}$, 但私钥由多个参与方以 $\text{sk} = \sum_{i=1}^m \text{sk}_i$ 的形式共同管理. 此时需要让 m 个参与方共同参与解密过程, 才能完成解密. 同时, Shuffle&Decrypt 协议还要求得到解密明文的 $\mathcal{P}_1$ 失去明文与密文的对应关系. 换句话说, 需要在“混淆”顺序后并解密. 为了让协议实现抗 $m-1$ 合谋的安全, 每一个参与方都需要生成一个随机置换 π_i . 协议的理想功能如图 6 所示. 各个 $\mathcal{P}_{i \in [m]}$ 各自持有随机置换 $\pi_i : [N] \rightarrow [N]$, 而参与方 $\mathcal{P}_1$ 持有基于国密 SM2 的加法同态密文集合 $C = \{c_j\}_{j \in [N]}$. 在协议执行结束时, $\mathcal{P}_1$ 得到 $\{x_{\pi(j)}\}_{j \in [N]}$, 其中 $\pi = \pi_m \circ \pi_{m-1} \circ \dots \circ \pi_1$.

Shuffle&Decrypt 协议的具体执行流程如协议 2 所示. 为了保证解密算法的高效性, 此处需要使用目前最高效的椭圆曲线加法同态解密的组件 FastECDLP [21], 用于求解解密过程中的小指数离散对数问题. 首先, 每一方 $\mathcal{P}_{i \in [m]}$ 不断地使用自己的置换 π_i 作用于密文集合 C , 然后不断地发给下一个参与方 $\mathcal{P}_{(i+1) \bmod m}$. 最后, 被完成多方混淆的密文集合会重新回到 $\mathcal{P}_1$ 手中. 然后, 参与方 $\mathcal{P}_1$ 把混淆后的密文 $\{c_j = c_{j,1}, c_{j,2}\}_{j \in [N]}$ 的第一部分 $\{c_{j,1}\}_{j \in [N]}$ 给其余参与方 $\mathcal{P}_{i \in [2, N]}$ 进行部分解密. 注意这一步是为了让其他所有的参与方可以并行地完成此联合解密过程, 以免在此等待. 最后, $\mathcal{P}_1$ 计算 $v_j = c_{j,2} - \sum_{i=1}^m c_{i,j}^{\text{par}}$ 并使用 FastECDLP 恢复明文 $x_j = \text{FastECDLP}(v_j)$. 需要注意的是, 基于椭圆曲线的加法同态加密方案都要受到明文空间的限制, 因此如果明文超过了可接受的明文空间 $\{0, 1\}^\ell$, 那么将无法解密. 在正式地介绍多运营域的黑/白名单安全求并协议时, 本文会解释这一缺点不仅不会对协议造成任何影响, 同时反过来还会加快其性能, 成为协议的优势.

协议 2 多方混淆并解密 (Shuffle&Decrypt).

输入: 每一个参与方 $\mathcal{P}_{i \in [m]}$ 各自持有随机置换 $\pi_i : [N] \rightarrow [N]$, 部分私钥 $\text{sk}_i \in \mathbb{Z}_p$, 参与方 $\mathcal{P}_1$ 持有基于国密 SM2 的加法同态密文集合 $C = \{c_j\}_{j \in [N]}$, 其中 $c_j = \text{HomoSM2.Enc}(\text{pk}, x_j)$ 且 $\text{pk} = \sum_{i=1}^m \text{sk}_i * G$;

参数: 基于国密 SM2 的加法同态加密方案 $\text{HomoSM2}(\text{Enc}, \text{ParDec}, \text{FulDec}, \text{ReRand}, \text{HomoAdd})$, 适用于椭圆曲线加法同态加密的快速求解离散对数算法 $\text{FastECDLP}^{[21]}$;

```

1: for  $1 \leq i \leq m$  do
2:    $\mathcal{P}_i$  重随机并置换  $C$  通过  $\pi_i: C = \pi_i(\{c_j = \text{HomoSM2.ReRand}(\text{pk}, c_j)\}_{j \in [N]})$ , 并把计算得到的  $C$  交给下一位参与方  $\mathcal{P}_{(i+1) \bmod m}$  (注意当循环结束后密文集合又会回到  $\mathcal{P}_1$ );
3: end for
4: 参与方  $\mathcal{P}_1$  发送  $\{c_j = c_{j,1}, c_{j,2}\}_{j \in [N]}$  的第一部分  $\{c_{j,1}\}_{j \in [N]}$  给其余参与方  $\mathcal{P}_{i \in [2,m]}$ ;
5: 所有  $\mathcal{P}_{i \in [m]}$  计算  $\{c_{i,j}^{\text{par}} = \text{sk}_i * c_{j,1}\}_{j \in [N]}$ .  $\mathcal{P}_{i \in [2,N]}$  将计算得到的  $\{c_{i,j}^{\text{par}}\}_{j \in [N]}$  发送回  $\mathcal{P}_1$ ;
6: for  $j \in [N]$  do
7:    $\mathcal{P}_1$  计算  $v_j = c_{j,2} - \sum_{i=1}^m c_{i,j}^{\text{par}}$ ;
8:    $\mathcal{P}_1$  调用  $\text{FastECDLP}$  得到  $x_j = \text{FastECDLP}(v_j)$ ;
9: end for
输出:  $\mathcal{P}_1$  输出  $\{x_j\}_{j \in [N]}$ . 注意如果  $x_j = \perp$ , 那么说明密文  $c_j$  超出了可接受的明文空间  $\{0, 1\}^\ell$ , 无法正常解密, 丢弃;
```

Shuffle&Decrypt 协议正确性分析. 协议 2 的正确性是显然的, 各方仅是依次对密文进行了混淆然后进行并发的联合解密操作. 最后 $\mathcal{P}_1$ 得到的 $\{x_j\}_{j \in [N]}$ 事实上已经是通过 $\pi = \pi_m \circ \pi_{m-1} \circ \cdots \pi_1$ 混淆后的解密结果.

定理2 (Shuffle&Decrypt 安全性) 协议 2 在半诚实模型下安全地实现了图 6 中的理想功能 $\mathcal{F}^{\text{Shuffle&Decrypt}}$, 且抗 $m-1$ 方合谋.

证明 要证明该协议是在半诚实模型下抗 $m-1$ 方合谋安全, 则需要证明任意 $m-1$ 方视图的联合分布在理想世界与现实世界中不可区分. 考虑到参与方 $\mathcal{P}_1$ 的特殊性, 还需要考虑 $\mathcal{P}_1$ 不是敌手以及 $\mathcal{P}_1$ 是敌手两种情况.

- $\mathcal{P}_1$ 不是敌手. 在该情况下, 需要证明的是其余参与方 $\mathcal{P}_{i \in [2,m]}$ 的视图的联合分布在理想世界与现实世界中不可区分. 注意 $\mathcal{P}_{i \in [2,m]}$ 的视图都是国密 SM2 加法同态方案的密文或部分密文. 此外, 在此场景中, 由于 $\mathcal{P}_1$ 总是诚实的, 其余参与方 $\mathcal{P}_{i \in [2,m]}$ 无法完成解密. 在他们看来, 这些密文与均匀随机的随机值不可区分, 因此他们的视图是可模拟的.

- $\mathcal{P}_1$ 是敌手. 在该情况下, 需要额外证明 $\mathcal{P}_1$ 的视图在理想世界与现实世界中不可区分, 其他与上述情况类似. 给定输出 $\{x_j\}_{j \in [N]}$, 模拟器加密得到 $\{c_j = \text{HomoSM2.Enc}(\text{pk}, x_j)\}$ 并随机生成一个置换 $\pi : [N] \rightarrow [N]$. 然后模拟器可以计算 $C = \pi(\{c_j\}_{j \in [N]})$ 模拟得到从 $\mathcal{P}_m$ 那里得到的 C . 由于总是有一方无法与合谋方合谋, 此时模拟器生成的 C 与实际协议的 $\mathcal{P}_1$ 从 $\mathcal{P}_m$ 那里得到的 C 从计算上不可区分. 继续模拟协议, 注意, $\mathcal{P}_1$ 收到剩下 $\mathcal{P}_{i \in [2,m]}$ 的 $\{c_{i,j}^{\text{par}}\}_{j \in [N]}$ 后, 只能与剩下至多 $m-2$ 方合谋. 因此, $\{c_{i,j}^{\text{par}}\}_{j \in [N]}$ 对于这些合谋的参与方来说是均匀随机的.

以上两种情况都证明, 即便是有任意 $m-1$ 个参与方被攻破同时进行合谋, 合谋方在理想世界与现实世界的视图仍是不可区分的, 因此学习不到更多的信息, 安全性得证.

4.2 方案描述

现如今的高效 MPSU 协议^[15~18]在设计上具有相似的结构. 计算 m 个集合 $X_i, i \in [m]$ 的并集的核心思想如下:

$$X_1 \cup (X_2 \setminus X_1) \cup \cdots \cup (X_m \setminus (X_1 \cup \cdots \cup X_{m-1})), \quad (4)$$

其中, 从 $\mathcal{P}_1$ 开始, 先从 $\mathcal{P}_2$ 收集 $X_2 \setminus X_1$, 得到 $X_1 \cup X_2$. 接着从 $\mathcal{P}_3$ 收集 $X_3 \setminus (X_1 \cup X_2)$, 得到 $X_1 \cup X_2 \cup X_3$, 如此继续, 直到从 $\mathcal{P}_m$ 收集 $X_m \setminus (X_1 \cup \cdots \cup X_{m-1})$, 最终得到 $X_1 \cup \cdots \cup X_n$. 从 $\mathcal{P}_i, i \in [m]$ 的视角来看, 对于其集合中的每个元素 $x_{i,k} \in X_i$, 它必须检查该元素是否已经出现在 $X_1 \cup \cdots \cup X_{i-1}$ 中. 如果 $x_{i,k} \in X_1 \cup \cdots \cup X_{i-1}$, 则 $\mathcal{P}_i$ 必须以某种方式去掉这个元素, 以确保它不会出现在最终结果中. 该成员关系检测在本文中由 MOT 协议 (协议 1) 结合国密 SM2 加法同态加密方案^[20] 共同完成, 在 $\mathcal{P}_i$ 与 $\mathcal{P}_j (i < j)$ 之间两两执行. 具体的实现方式是

在 P_i 与 P_j 调用 MOT 协议时, P_j 的 OT 协议输入选择

$$\{\text{HomoSM2.Enc}(\text{pk}, 0), \text{HomoSM2.Enc}(\text{pk}, r_k)\}_{k \in [B]}, \quad (5)$$

其中 r_k 是从 $\mathbb{Z}_p$ 选择的随机值, $B = 1.27n$. 根据 MOT 协议, 可知如果 P_j 的元素 $x_{j,k} \in X_j$ 没有出现在 X_i 中, 则 P_i 会得到零的密文, 否则会得到随机数 r_k 的密文. 然后 P_i 调用 HomoSM2.ReRand 来重随机化该密文, 进而回发给 X_j . 由于 P_j 会和 $j-1$ 个 P_i 执行 MOT, 可知只有当 $x_{j,k} \notin X_1 \cup \dots \cup X_{j-1}$ 时, 才能得到 $j-1$ 个全零的密文对于 $x_{j,k} \in X_j$. 然后通过调用 HomoSM2.HomoAdd 实现

$$c_{j,k} = \text{HomoSM2.Enc}(\text{pk}, x_{j,k}) + \sum_{i=1}^{j-1} c_{i,j,k}, \quad (6)$$

其中 $c_{i,j,k}$ 代表 P_i 与 P_j 执行 MOT 协议后 P_i 向 P_j 返回的对应 $x_{j,k}$ 的密文. 也就是说, 如果 $c_{j,k}$ 仍然是 $x_{j,k}$ 的合法密文, 则说明 $x_{j,k} \notin X_1 \cup \dots \cup X_{j-1}$, 否则 $c_{j,k}$ 会被称一个随机的密文.

为了区分随机值的密文和 $x_{j,k}$ 的密文, Gao 等^[18] 采用的方式是在对 $x_{j,k}$ 加密前对它进行“补零”. 通过解密后判断低位是否有 λ 个 0 来判断是否是合法元素, 误判率为 $2^{-\lambda}$. 但是, 这种方式会严重地降低解密的性能. 首先, 因为文献 [18] 采用的是 Exp-ElGamal 方案^[19] 进行加法同态操作, 这同样也是一种基于椭圆曲线的加法同态加密方案, 也需要控制明文长度在 $\{0, 1\}^\ell$ 中. 因此, 明文的实际空间仅为 $\{0, 1\}^{\ell-\lambda}$. 根据 Tang 等^[21] 的评估, ℓ 为 54 比特时使用 BSGS 算法^[40] 需要 20 min 才能完成一次解密. 而此时在一般的统计安全参数 $\lambda = 40$ 的设置下, $\ell - \lambda = 54 - 40 = 14$. 这意味着仅只有 14 比特的空间, 也就是说所有参与方的输入集合的全集仅只有 $2^{14} = 16384$ 种可能, 这明显不切实际. 本文采用的是基于国密 SM2 的加法同态加密^[20], 这是一种比 Exp-ElGamal 方案更高效的基于椭圆曲线的加法同态加密方案, 同时配备现有的最高效的基于椭圆曲线加法同态解密 FastECDLP^[21] 来加速解密过程. 此外, 本文不再采用补零的措施来区分是否为正常明文, 而是通过 FastECDLP 是否解密成功来进行判断. 这是因为如果 $c_{j,k}$ 仍然是 $x_{j,k}$ 的合法密文, 那么 FastECDLP 一定能够完成解密. 当然如果不再是合法密文, 那么 FastECDLP 是以压倒性的优势返回解密失败 $\perp$. 这是因为 $r_j \leftarrow \mathbb{Z}_p$, 则该密文不再是合法密文, 其对应的明文是均匀分布于 $\mathbb{Z}_p$ 的. 由于 p 是 256 位的大素数, 这个非法明文刚好落于 $\{0, 1\}^\lambda$ 的概率 (即错误率) 是 $2^{-(256-\ell)}$. 由于 ℓ 一般取 32 到 64, 错误率甚至比“补零”的方法低得多, 同时还大幅提升了解密性能. 明文空间的劣势被本文转化为了相对于现有文献 [18] 的优势.

多运营域黑/白名单安全求并协议的完整执行流程如协议 3 所示. 协议的第 1~4 行都属于预准备阶段, 此时主要进行的是各个运营域将自身的黑或白名单插入到哈希表中, 并执行一些加密运算, 这过程无需任何交互. 而从第 5~11 行, P_i 与 P_j ($i < j$) 之间两两执行了 MOT 协议并由 P_1 完成了密文的整合, 得到密文集合 C , 其中 $|C| = (m-1)B$. 由前面的分析可知, 此时的 C 要么是随机值的密文, 要么就是除了 X_1 之外的合法并集元素的密文. 在第 12 行调用 Shuffle&Decrypt 协议 (即协议 2) 后 P_1 就能成功地得到并集, 并且失去密文集合 C 与 $U = \bigcup X_i$ 的对应关系, 由于每一方都添加了随机置换 π . 这个过程也是抗 $m-1$ 合谋的. 在协议的最后, P_1 负责广播 U 给其他运营域 $P_{i \in [2, m]}$.

多运营域黑/白名单安全求并正确性分析. 可以考虑两种情况, 取决于某个特定元素 $x \in X_j$ (来自运营域 P_j) 是否存在于某个集合 X_i 中 (来自运营域 P_i), 其中 $1 \leq i < j \leq m$.

• **情况 1:** $x \in X_1 \cup \dots \cup X_{j-1}$. 在这种情况下, P_i 会从 P_j 的 MOT 协议中接收到至少一个随机值的加密形式 $\text{HomoSM2.Enc}(\text{pk}, r_k)$. 因此, 在第 9 行, 密文相加得到的结果几乎不可能再是 $\text{HomoSM2.Enc}(\text{pk}, x)$, 而会得到一个随机的 $\mathbb{Z}_p$ 中元素的加密形式. 这在后续的 Shuffle&Decrypt 中只有 $2^{-(256-\ell)}$ 完成解密, 这几乎不可能. 因此, x 不会包含在最终结果中.

• **情况 2:** $x \notin X_1 \cup \dots \cup X_{j-1}$. 在这种情况下, P_i 会从 P_j 的 MOT 协议中全都接收到零密文 $\text{HomoSM2.Enc}(\text{pk}, 0)$. 因此, 在第 9 行, 密文相加的得到的结果就是 $\text{HomoSM2.Enc}(\text{pk}, x)$. 而在这种情况下 FastECDLP 一定能回复出明文 x , 且 x 将被包含在最终结果中.

协议 3 多运营域黑/白名单安全求并.

输入: 运营域 $\mathcal{P}_{i \in [m]}$ 各自持有黑名单或白名单集合 $X_i = \{x_{i,j}\}_{j \in [n]}$;

参数: 基于国密 SM2 的加法同态加密方案 $\text{HomoSM2}(\text{Enc}, \text{ParDec}, \text{FulDec}, \text{ReRand}, \text{HomoAdd})$, 哈希桶参数 B , 以及布谷鸟哈希的哈希函数个数 t ;

- 1: 每一个 $\mathcal{P}_{i \in [m]}$ 随机选择 $\text{sk}_i \leftarrow \mathbb{Z}_p$, 并共同计算 $\text{pk} = \sum_{i=1}^m \text{sk}_i * G$;
- 2: 每一个 $\mathcal{P}_{i \in [m]}$ 都将自身的黑/白名单集合 $X_i = \{x_{i,j}\}_{j \in [n]}$ 插入到一个有 B 个桶的哈希表 $T_{S,i}$ 中;
- 3: 除了 $\mathcal{P}_1$ 以外的每一个 $\mathcal{P}_{i \in [2,m]}$ 都将自身的黑/白名单集合 $X_i = \{x_{i,j}\}_{j \in [n]}$ 插入到一个有 B 个桶的布谷鸟哈希表 $T_{C,i}$ 中;
- 4: 除了 $\mathcal{P}_1$ 以外的每一个 $\mathcal{P}_{i \in [2,m]}$ 加密 $T_{C,i}$, 即设置 $c_{i,k} = \text{HomoSM2.Enc}(\text{pk}, T_{C,i}[k])$, 其中 $k \in [B]$, 布谷鸟哈希的空白位置用 0 密文填充;
- 5: **for** $1 \leq i < j \leq m$ **do**
- 6: $\mathcal{P}_i$ 和 $\mathcal{P}_j$ 调用 mOT 协议 (即协议 1), 其中 $\mathcal{P}_i$ 使用输入 $T_{S,i}$ 作为接收方; $\mathcal{P}_j$ 使用输入长度为 B 的明文 $T_{C,j}$ 与 B 个消息对 $\{\text{HomoSM2.Enc}(\text{pk}, 0), \text{HomoSM2.Enc}(\text{pk}, r_k)\}_{k \in [B]}$ 作为发送方, 其中 $r_k \leftarrow \mathbb{Z}_p$, $\mathcal{P}_i$ 取得输出 $\{c_{i,j,k}\}_{k \in [B]}$;
- 7: $\mathcal{P}_i$ 重随机化密文 $c_{i,j,k} = \text{HomoSM2.ReRand}(\text{pk}, c_{i,j,k})$ 并将其发送给 $\mathcal{P}_j$, 其中 $k \in [B]$;
- 8: **end for**
- 9: 每一个 $\mathcal{P}_{j \in [2,m]}$ 整合重随机化密文到加密的布谷鸟哈希表中: $c_{j,k} = c_{j,k} + \sum_{i=1}^{j-1} c_{i,j,k}$ 通过调用 HomoSM2.HomoAdd ;
- 10: 每一个 $\mathcal{P}_{j \in [2,m]}$ 发送加密的 $c_j = \{c_{j,k}\}_{k \in [B]}$ 给 $\mathcal{P}_1$;
- 11: $\mathcal{P}_1$ 计算密文并集 $C = \cup_{j=2}^m c_j$ 并初始化 $U = X_1$;
- 12: 令 $N = |C|$, 所有运营域 $\mathcal{P}_{i \in [m]}$ 各自持有随机置换 $\pi_i : [N] \rightarrow [N]$ 共同调用 $\text{Shuffle\&Decrypt}$ 协议 (即协议 2), 以 C 作为解密密文集合同时 $\mathcal{P}_1$ 作为接收方, 如果解密失败, 即对于某个密文 $c \in C$, 调用 HomoSM2.FulDec 而无法得到明文, 则丢弃该密文; 如果解密成功, 则 $\mathcal{P}_1$ 将解密得到的明文加入 U ;
- 13: $\mathcal{P}_1$ 广播 U 给其他运营域 $\mathcal{P}_{i \in [2,m]}$;

输出: 所有的运营域 $\mathcal{P}_i$ 按黑名单或白名单并集 U 进行接入控制;

此外, 对于任意 $x \in X_1$, 它都会被包含在最终结果中. 因此, 协议 3 的正确性是成立的.

现在开始证明协议的安全性. 该协议可被证明在半诚实模型下是安全的且抗 $m-1$ 方合谋. 需要注意的是, 将本文面向“全体参与方输出并集”的协议进一步提升到恶意安全, 需要在关键步骤中额外处理输入一致性、消息正确性与选择性中止等问题, 并通常需要引入可验证机制, 从而带来额外计算与通信开销. 本文聚焦于在半诚实模型下给出可证明安全且工程可部署的高效方案; 关于恶意安全扩展所需的组件与可能的实现路径, 将在第 6 节中给出.

定理3 (MPSU 安全性) 协议 3 在半诚实模型下安全地实现了图 3 中的理想功能 $\mathcal{F}^{\text{MPSU}}$, 且抗 $m-1$ 方合谋.

证明 存在 $\mathcal{P}_1$ 是否被攻破两种情况, 即 $\mathcal{P}_1$ 不是敌手或 $\mathcal{P}_1$ 是敌手. 需要证明的是, 在这两种情况下, $m-1$ 个腐败方的视图的联合分布在理想世界与现实世界中不可区分.

- $\mathcal{P}_1$ 不是敌手. 在协议执行的第 5~8 行, 被腐败方的视图在理想世界中的联合分布可以调用理想的 $\mathcal{F}^{\text{MOT}}$ 得到. 在协议执行的第 12 行, 被腐败方的视图在理想世界中的联合分布可以调用理想的 $\mathcal{F}^{\text{Shuffle\&Decrypt}}$ 得到. 因此, 被腐败方的联合视图是可模拟的.

- $\mathcal{P}_1$ 是敌手. $\mathcal{P}_1$ 通过给定的输出 U , 加密得到 $C = \{\text{HomoSM2.Enc}(\text{pk}, u)\}_{u \in U}$, 然后随机生成置换 $\pi : [N] \rightarrow [N]$, 生成从其他参与方得到的密文集合 C 的视图. 调用理想的 $\mathcal{F}^{\text{Shuffle\&Decrypt}}$ 即可得到在第 12 行的视图. 进而调用理想的 $\mathcal{F}^{\text{MOT}}$ 得到第 5~8 行的视图. 因此, 在 $\mathcal{P}_1$ 是敌手的情况下, 被腐败方的联合视图仍然是可模拟的.

以上两种情况都证明, 即便是有任意 $m-1$ 个参与方被攻破同时进行合谋, 合谋方在理想世界与现实世界的视图仍是不可区分的, 因此学习不到更多的信息, 安全性得证.

4.3 明文长度的选择

由于考虑到明文长度 ℓ 直接决定解密性能, 那么 ℓ 的选取就显得非常重要了. 要让 ℓ 尽可能地小 (解密才能更快), 同时还要保证所有参与方的元素能映射到 $\{0, 1\}^\ell$ 以可忽略的概率 ($< 2^{-\lambda}$) 出现碰撞. 设一个哈希函数 $f : \{0, 1\}^* \rightarrow \{0, 1\}^\ell$ 将参与方的集合元素映射到 $\{0, 1\}^\ell$ 上. 如果有 m 个参与方, 每一个参与方的集合元素大小

表 1 多运营域黑/白名单安全求并性能分析总结. $|c|$ 和 ℓ 分别代表明文与密文的位长. 如果调用对象是同态操作, 则仅统计操作次数. 如果调用对象是协议, 统计的是调用次数乘以复杂度.

Table 1 Performance analysis summary of multi-domain black/white list secure union. $|c|$ and ℓ denote the bit lengths of plaintext and ciphertext, respectively. If the invoked entity is a homomorphic encryption operation, only the number of operations is counted. If the invoked entity is a protocol, the statistic represents the number of invocations multiplied by its complexity.

Module	Computation overhead (invocations)	Communication overhead (volume/invocations)
Enc	$(m-1)(mB+n)$	—
ReRand	$\frac{m(m-1)B}{2}$	—
HomoAdd	$\frac{m(m-1)B}{2}$	—
MOT	$\frac{m(m-1)}{2}\mathcal{O}(n)$	$\frac{m(m-1)}{2}\mathcal{O}(n)$
Shuffle&Decrypt	$\mathcal{O}(mn)$	$\mathcal{O}(mn)$
Plaintext	—	$(m-1) U \cdot \ell$
Ciphertext	—	$\frac{m(m-1)B}{2} \cdot c $

是 n , 那么套用生日攻击, 可以得到此时的碰撞概率为

$$1 - e^{-\frac{(mn)^2}{2^{\ell+1}}}. \quad (7)$$

可进一步推得

$$\ell > 2 \log(mn) - 1 - \log \ln \left(\frac{1}{1-q} \right), \quad (8)$$

其中 q 指代碰撞概率. 注意由于当 $q=0$ 时存在 $q \approx \ln(1+q)$, 因此, 进一步能推得

$$\ln \left(\frac{1}{1-q} \right) = \ln \left(1 + \frac{q}{1-q} \right) \approx \frac{q}{1-q}. \quad (9)$$

因此, ℓ 的极小值事实上应该为 $\ell = 2 \log(mn) + \lambda - 1$. 取一个极端情况, 以 $n = 2^{16}$ 为例, $m = 50$, 此时 $\ell = 2 \log(50 \cdot 2^{16}) + 40 - 1 \approx 83$ 发生碰撞的概率才是可忽略的. 因此, 为了保证协议的性能且避免元素编码碰撞带来的正确性风险, 本文在系统部署中引入一个预处理阶段: 由协同控制面为当前同步周期维护一个全局的无碰撞编码表, 并将其分发至各参与方. 各参与方据此将本域元素确定性地映射到 $\{0, 1\}^\ell$, 同时保证在该周期内全网范围内不发生编码碰撞.

5 实验与分析

本节对本文所提方案进行理论分析与仿真实验以证明本文方案的高效性.

5.1 理论分析

通过记录协议 3 中调用基于 SM2 的加法同态加密的 (Enc, ParDec, FulDec, ReRand, HomoAdd) 等算法的次数以及调用其他协议的次数以统计协议的开销. 设布谷鸟哈希表桶的个数 $B = 1.27n$. 首先, 在协议的第 4 行, 可以看到除了 $\mathcal{P}_1$ 之外的参与方都需要执行 B 次加密操作, 因此一共需要调用 $(m-1)n$ 次 Enc. 然后, 从第 5~8 行能看出, $\mathcal{P}_i$ 和 $\mathcal{P}_j$ 一共需要调用 $\frac{m(m-1)}{2}$ 次 MOT 协议, 并执行了一共 $m(m-1)B$ 次 Enc 以及 $\frac{m(m-1)B}{2}$ 次 ReRand. 同时, 此处还会产生 $\frac{m(m-1)B}{2}$ 个密文的通信量. 在协议的第 9 行, 一共需要 $\frac{m(m-1)B}{2}$ 次 HomoAdd. 此外, 在协议的第 10 行产生 $(m-1)B$ 个密文的通信量. 最后在第 12 行调用一次 Shuffle&Decrypt 协议, 并在第 13 行产生 $(m-1)|U|$ 个明文的通信量, 其中 $|U|$ 指的是并集的基数. 本文将该分析的结果总结于表 1 中.

5.2 实验仿真

5.2.1 实验设置

本节实验的实验环境、涉及的对基线, 以及数据集被总结如下.

• 实验环境. 所有实验均在一台操作系统为 Ubuntu 20.04 的个人笔记本电脑上进行, 配备 16 GB 内存和第 11 代英特尔酷睿 i7-1165G7 @ 2.80 GHz 处理器. 本文所提方案的完整实现超过 9000 行 C++ 代码 (不包括第三方库). 在广域网 (WAN) 环境下, 通过 Linux 系统的 “tc” 命令实现带宽限制. 对于多个运营域的通信, 本文使用 BRPC 库³⁾实现, 这是一个工业级的 RPC 框架. 为了实验的方便, 设置所有域的黑白名单大小是相等的, 均为 $n \in \{2^8, 2^{12}, 2^{16}\}$. 协议底层的 SSPMT 协议通过一个高效的 OPRF 协议^[41]与 SecretFlow SPU 库⁴⁾中的 SEMI2K 协议簇实现. SM2 加法同态加密^[20]结合 FastECDLP^[21]的实现源自蚂蚁集团的 YACL 库⁵⁾. 由于需要扩展到多密钥版本, 本文对其进行了一定量的修改⁶⁾. SM2 加法同态加密的明文长度可设置为 $\ell = 40$. 根据 FastECDLP^[21], 这需要一个大约为 2.76 GB 的预计算表. 所有源代码已经被上传到 GitHub 仓库⁷⁾以便于读者理解本文的协议.

• 对比基线. 本文选择现有的三个高效 MPSU 协议^[16~18]作为本文的对比方案, 他们均代表了现有的 MPSU 协议的最先进水平. 当对比 WAN 设置下的协议性能时, 遵循文献 [18] 的设置, 采用 400 Mbps 的带宽与 80 ms 的延迟.

• 数据集. 为了保证本文方案可以直接应用于多运营域黑/白名单安全求并. 本文并非像文献 [16~18] 那样使用随机生成的元素进行实验评估. 而是使用公开的黑/白名单数据集来模拟各运营域对各自黑/白名单进行安全求并的完整过程. 由于目前没有卫星互联网运营域黑/白名单的公开数据集. 本文直接使用了两个一般场景下的黑/白名单数据集进行仿真实验. 黑名单直接对 IPsum⁸⁾数据集采样得到. IPsum 基于 30 多个不同的公共恶意 IP 地址列表, 现在已经收集了超过 16 万个恶意 IP 地址. 白名单直接来自 “全球可路由前缀日清单”⁹⁾. 它用于刻画互联网上当日对外公告的 IP 网段, 单日快照规模约为 100 万条 IPv4 前缀与 20 万条 IPv6 前缀. 模拟生成各方的黑/白名单的方法是相同的. 具体地说, 首先固定各方的输入大小 n , 然后对于每一个参与方 P_i , 都随机采样 n 个元素作为该参与方的黑/白名单.

数据集代表性与外推边界. 受商业敏感与合规约束影响, 真实卫星互联网运营域的黑/白名单通常无法公开获取. 本文因此选取公开可验证的真实世界名单数据作为替代, 并通过随机划分与参数控制来构造多运营域输入. 需要强调的是, 本文协议的计算与通信开销主要由集合规模 n 、参与方数量 m 以及并集规模 $|U|$ 等统计特征决定, 而与名单条目来自何种行业语义基本无关. 因此, 本文所选用的数据集可作为评估规模化同步任务的合理代理. 与此同时, 本文强调该替代数据无法完全覆盖卫星网络中可能存在的特殊业务分布与策略约束, 真实运营环境下的进一步验证属于后续工作.

5.2.2 实验结果

仿真实验的主要实现结果被整理于表 2 中. 该表同时给出了所提方案与现有方案^[16~18]在局域网与广域网两类网络条件下的端到端性能对比 (运行时间与通信开销). 总体来看, 所提方案在两类网络下均表现出显著的时延优势; 其中, 在局域网环境下优势最为稳定与全面, 而在广域网环境大部分设置均具有优势, 仅在部分设置下没有取得最优的结果.

具体到局域网环境, 所提方案在表中给出的所有配置下均取得最短运行时间. 以具有代表性的 $n = 2^{16}$, $m = 8$ 为例, 所提方案用时 497.18 s, 明显优于文献 [18]/文献 [17]/文献 [16] 的 517.27/771.40/1764.44 s, 对应加速比分别为 $1.04 \times / 1.55 \times / 3.55 \times$. 这一结果表明, 在低时延、抖动小的链路条件下, 所提方案能够在全配置上稳定取得最短执行时间. 在广域网环境下, 整体趋势与局域网一致, 但存在轻微的小规模例外: 当输入为 $n = 2^8$ 且运营域数量 $m \in \{3, 4\}$ 时, 文献 [17, 18] 在少数点位上取得了更短的用时. 其主要原因在于: 在该区间内, 端到端时延往往由网络交互相关的固定代价主导, 而非由随 n 线性增长的主体计算项主导. 具体而言, 不同方案之间的差

3) <https://github.com/apache/brpc>.

4) <https://github.com/secretflow/spu>.

5) <https://github.com/secretflow/yacl/tree/main/examples/hesm2>.

6) <https://github.com/ShallMate/MPSU/tree/main/hesm2>.

7) <https://github.com/ShallMate/MPSU>.

8) <https://github.com/stamparm/ipsum>.

9) <https://publicdata.caida.org/datasets/routing/routeviews-prefix2as/>.

表 2 (网络版彩图) 多运营域黑/白名单安全求并与现有方案^[16~18]的性能对比. 如果现有工作为最优结果, 则以粗体标注; 若本文提出的方案为最优结果, 则以红色标注. 次优的结果以蓝色标注. †: 在广域网下且 $n = 2^8$ 时, 各方案差距收敛, 受 RTT 与固定通信开销影响更明显.

Table 2 (Color online) Performance comparison between our secure union of blacklists/whitelists over multi-operator domains and existing schemes [16~18]. If the existing work is optimal, it is highlighted in **bold**; if the proposed work is optimal, it is highlighted in **red**. Second-best results are highlighted in **blue**. †: In the WAN setting with $n = 2^8$, the performance differences among schemes converge, largely due to the dominant effect of RTT and fixed communication overhead.

	Input size (n)	Protocol	$m = 3$	$m = 4$	$m = 6$	$m = 8$
LAN runtime (s)	2^8	[16]	1.12	1.79	3.85	6.53
		[17]	0.69	1.19	2.60	4.09
		[18]	0.64	0.83	1.40	2.23
		Ours	0.35	0.67	1.31	2.14
	2^{12}	[16]	15.71	27.12	60.54	104.35
		[17]	6.35	11.90	27.52	45.06
		[18]	5.93	9.18	17.54	29.33
		Ours	4.79	9.10	16.53	27.04
	2^{16}	[16]	297.48	482.51	1018.22	1764.44
		[17]	106.46	189.14	435.54	771.40
		[18]	95.71	161.59	303.37	517.27
		Ours	75.95	144.05	283.46	497.18
WAN runtime (s)	2^8	[16]	9.80	15.25	27.66	39.02
		[17]	4.21	5.18	10.12	15.37
		[18]	4.35	5.33	8.41	11.02
		Ours	5.97 [†]	6.31 [†]	7.01	7.93
	2^{12}	[16]	32.04	49.62	95.81	149.72
		[17]	12.79	21.16	39.78	63.89
		[18]	11.80	17.76	28.08	43.67
		Ours	10.54	14.98	22.81	33.89
	2^{16}	[16]	353.33	576.23	1196.77	2055.14
		[17]	128.30	221.82	456.06	777.22
		[18]	111.73	170.96	330.14	597.54
		Ours	83.71	153.77	298.95	520.92
Communication overhead (MB)	2^8	[16]	1.46	2.20	3.68	5.16
		[17]	2.34	3.51	5.85	8.19
		[18]	1.10	1.74	3.23	4.97
		Ours	1.44	2.88	7.21	13.47
	2^{12}	[16]	20.25	30.48	50.96	71.47
		[17]	8.15	12.82	23.95	38.07
		[18]	11.82	19.28	37.87	59.41
		Ours	12.51	25.03	62.58	116.81
	2^{16}	[16]	321.40	483.69	808.79	1134.21
		[17]	65.41	98.12	163.50	228.90
		[18]	184.41	301.37	584.79	934.20
		Ours	108.68	217.45	543.17	1012.96

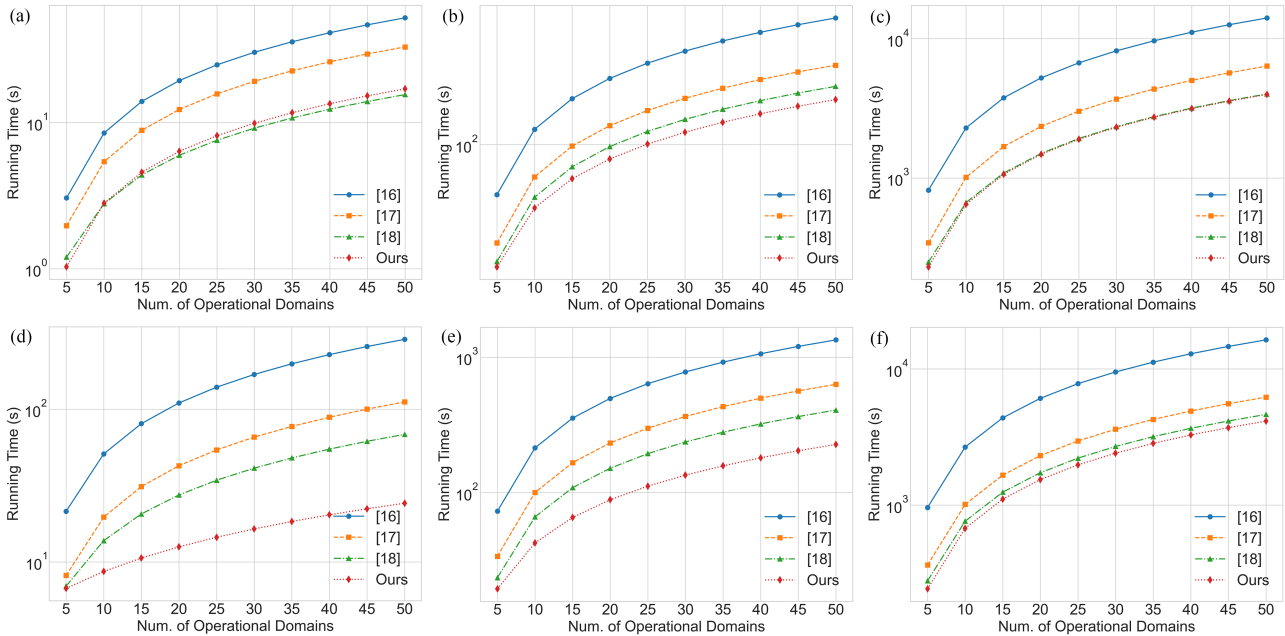


图 7 (网络版彩图) 当运营域数量从 5~50 时, 局/广域网下各个协议的运行时间对比. (a) 局域网 ($n = 2^8$); (b) 局域网 ($n = 2^{12}$); (c) 局域网 ($n = 2^{16}$); (d) 广域网 ($n = 2^8$); (e) 广域网 ($n = 2^{12}$); (f) 广域网 ($n = 2^{16}$).

Figure 7 (Color online) Runtime comparison of the protocols in LAN and WAN as the number of operator domains increases from 5 to 50. (a) LAN ($n = 2^8$); (b) LAN ($n = 2^{12}$); (c) LAN ($n = 2^{16}$); (d) WAN ($n = 2^8$); (e) WAN ($n = 2^{12}$); (f) WAN ($n = 2^{16}$).

异更多由关键交互轮次、消息封装/序列化以及连接维护等固定开销决定, 从而导致优势在小规模下收敛, 甚至在个别点位出现非最优. 然而, 随着 n 增大或 m 增多, 重新取得并放大整体优势. 这是因为当运营域数量增多, 本文方案在计算开销方面的优势被进一步扩大. 仍以 $n = 2^{16}, m = 8$ 为例, 广域网环境下所提方案仅需 520.92 s 完成执行, 优于文献 [18] 的 597.54 s (1.15 $\times$), 文献 [17] 的 777.22 s (1.49 $\times$), 以及文献 [16] 的 2055.14 s (3.95 $\times$), 显示出在更贴近真实部署的中大规模工况中, 所提方案的时延优势更为稳健与显著. 通信开销方面, 当输入规模较小时, 通信开销最优的方案是文献 [17]. 而当输入大小变大后, 通信开销最优的方案是文献 [18]. 而所提方案在部分设置下达到了次优的通信开销. 特别是当运营域较多 ($m = 8$) 时, 大部分设置下本文方案均达到了次优的通信开销. 尽管在 $n = 2^{16}$ 时没有达到次优, 但是也与文献 [16, 18] 的通信开销差距不大. 总体而言, 所提方案在局域网/广域网两类网络环境下均呈现出更优的完结时延与接近次优的通信成本, 体现出良好的可扩展性与部署价值.

本文还给出了当运营域数量更多时, 所提方案与现有方案^[16~18]的性能对比. 图 7 给出了当运营域数量从 5~50 时, 不同方案在不同输入大小 n 以及不同网络环境下的运行时间. 可以看到, 相比现有方案, 本文方案在广域网环境下取得了全面的领先. 在局域网环境下, 也仅在当 $n = 2^8$ 时, 本文方案的运行时间会稍慢于文献 [18]. 因此, 这说明了在真实时延与多运营域场景中所提方案具备更好的可扩展性与更短的运行时间.

此外, 本文给出 MOT 协议 (协议 1) 与 Shuffle&Decrypt 协议 (协议 2) 独立运行的性能. 这两个协议在不同输入大小以及不同网络环境下的运行结果被总结在表 3 中. 需要注意的是 MOT 协议是一个两方协议, 在本文的所提方案中有两个运营域两两各自调用它. 因此, 在报告 MOT 协议的性能时仅报告 $m = 2$ 时的运行结果. 而 Shuffle&Decrypt 协议则是一个多方协议, 需要所有的运营域共同完成. 因此在报告它的性能时, 参与方的数目与评估安全求并时相同, 同样是 $m \in \{3, 4, 6, 8\}$. 结果, 随着输入规模的增大, 两类协议的运行时间和通信量均呈近似线性增长, 表现出良好的可扩展性. 在局域网环境下, 协议整体运行效率较高, 运行时间均保持在秒级至数十秒量级. 而在广域网环境下, 尽管运行时间有所上升, 但总体仍维持在可接受范围内.

最后, 考虑卫星互联网链路存在时延较高、带宽受限以及丢包波动等特征, 本节进一步在网络仿真环境中对上述链路参数进行分组变化实验, 以量化端到端完结时延对链路条件的敏感性. 在该实验中, 首先固定参与方数

表3 MOT协议与 Shuffle&Decrypt协议的运行时间与通信开销.

Table 3 Runtime and communication overhead of the MOT protocol and the Shuffle&Decrypt protocol.

	Input size (n)	MOT		Shuffle&decrypt		
		$m = 2$	$m = 3$	$m = 4$	$m = 6$	$m = 8$
LAN runtime (s)	2^8	0.08	0.06	0.07	0.08	0.09
	2^{12}	1.26	1.06	1.27	1.33	1.51
	2^{16}	23.18	20.8	30.17	52.45	73.18
WAN runtime (s)	2^8	0.41	0.54	0.71	1.04	1.37
	2^{12}	1.64	1.57	1.95	2.35	2.87
	2^{16}	24.33	21.69	31.40	54.36	75.71
Communication overhead (MB)	2^8	0.39	0.08	0.11	0.17	0.24
	2^{12}	2.86	1.28	1.80	2.84	3.87
	2^{16}	41.41	20.62	29.45	47.44	62.41

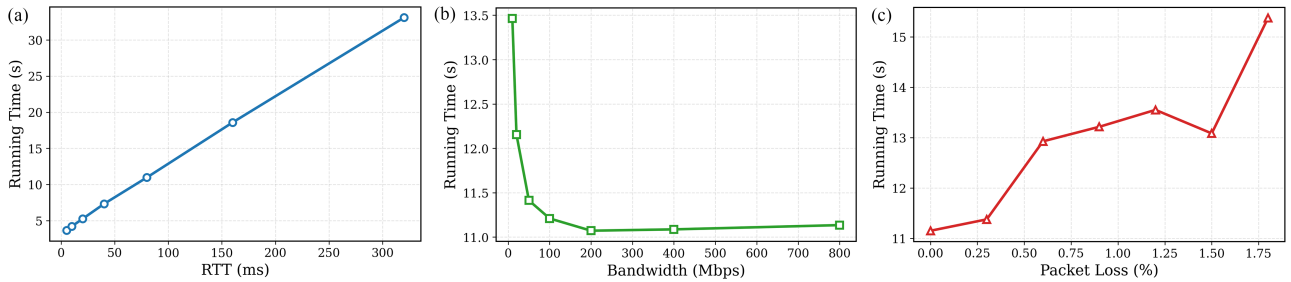


图8 (网络版彩图) 固定输入规模 $m = 4, n = 2^{12}$ 时, 协议在不同链路条件下的运行时间敏感性分析. (a) 往返时延 (RTT); (b) 可用带宽; (c) 丢包率.

Figure 8 (Color online) Sensitivity of the protocol runtime to network conditions with fixed input size $m = 4$ and $n = 2^{12}$. (a) Round-trip time (RTT); (b) available bandwidth; (c) packet loss rate.

为 $m = 4$ 、输入规模为 $n = 2^{12}$, 并分别改变三类链路参数: RRT、链路可用带宽与丢包率.

- RRT: 仅改变 RRT, 取值为 $\{5, 10, 20, 40, 80, 160, 320\}$ ms; 同时将链路可用带宽固定为 400 Mbps、丢包率固定为 0%.
- 带宽: 仅改变链路可用带宽, 取值为 $\{10, 20, 50, 100, 200, 400, 800\}$ Mbps; 同时将链路往返时延固定为 80 ms、丢包率固定为 0%.
- 丢包率: 仅改变丢包率, 取值为 $\{0\%, 0.1\%, 0.5\%, 1\%, 2\%\}$; 同时将链路往返时延固定为 80 ms、链路可用带宽固定为 400 Mbps.

从图 8(a) 中可以看出, 当 $n = 2^8$ 时, 协议的运行时间会受到 RTT 的严重影响. 因此, 这进一步说明了在表 2 中本文协议在 2^8 且广域网设置下不是最优是合理的. 图 8(b) 展示了协议在不同可用带宽条件下的运行时间变化情况. 当带宽从较低水平逐步提升时, 运行时间显著下降; 而当带宽超过一定阈值后, 运行时间趋于稳定, 进一步增加带宽对性能改善的影响十分有限. 这一现象说明, 在中高带宽环境下, 协议已不再受吞吐能力限制, 其性能瓶颈主要来自固定的通信轮次与计算开销. 而图 8(c) 反映了不同丢包率条件下协议运行时间的变化趋势. 实验结果显示, 在低至中等丢包率范围内, 协议运行时间仅出现温和增长, 表现出一定的鲁棒性; 而在较高丢包率条件下, 运行时间出现明显上升, 表明频繁的重传与链路不稳定性会对协议整体时延产生显著影响.

总体而言, 上述实验结果表明, 该协议在卫星互联网中常见的高带宽但高往返时延链路条件下仍能够保持稳定且可预测的性能表现; 同时, 在存在一定程度链路不稳定与随机丢包的情况下, 协议展现出良好的鲁棒性. 然而, 在极端高时延或持续高丢包的卫星链路环境中, 其运行时间仍不可避免地受到网络条件的显著制约.

6 讨论与未来工作

本文协议在半诚实模型下给出安全性证明. 若考虑恶意敌手, 参与方可能主动偏离协议以破坏正确性或扩大泄露面, 典型方式包括提交畸形输入/密文、发送不一致的中间消息, 以及选择性中止. 为将本文方案提升到恶意安全, 需在不显著破坏弱网场景效率的前提下, 引入以下三类机制.

- 输入与消息正确性: 要求各方对关键消息提供一致性与有效性保障, 例如对同态密文的群元素合法性、对重随机化步骤的正确性, 以及对解密份额的正确性进行可验证检查, 从而防止畸形消息导致输出被污染. 具体的实现方法可以参见文献 [45].
- 恶意安全子协议替换: 将本文所提出的半诚实模型下 MOT 协议所使用的扩展替换为恶意安全 OT 扩展 [46, 47]; 对 Shuffle&Decrypt 协议引入可验证混洗或可验证重随机化 [48, 49], 并对每方的部分解密份额增加可验证正确性证明, 以确保即便存在主动偏离也能维持输出正确性.
- 选择性中止与工程缓解: 在强合谋假设下, 完全公平通常需要额外机制支持. 工程部署中可采用超时与重试、审计追责与惩罚, 以及仲裁节点/策略控制点协助等方式降低中止风险 [50, 51].

综合而言, 本文工作为半诚实模型下的高效可部署方案提供了基础, 而恶意安全扩展需要在可验证性组件与系统开销之间做进一步权衡.

此外, 从实验结果中可以看出, 本文所提出方案在通信量方面相对于现有方案 [16~18] 并非最优. 因此, 本文在此处给出进一步优化通信开销的优化方向, 作为本文的后续工作.

- 密文体积优化: 对椭圆曲线点采用压缩编码与更紧凑序列化 [52], 结合批量打包降低每个消息的头部开销, 从而降低密文传输项.
- 降低 RTT 敏感性: 在广域网场景优先合并可并行消息, 减少同步点与关键交互轮次, 并采用连接复用与流水线化隐藏 RTT.
- 并集分发优化: 在接入控制的业务形态下采用“增量同步”, 仅广播新增/撤销项, 将 $(m-1)|U|$ 的分发成本降为 $(m-1)|\Delta U|$, 其中 ΔU 代表增加的输出结果项; 或提供“受控输出/按需拉取”的部署选项以进一步降低带宽瓶颈.

7 结论

本文面向 LEO 卫星互联网的跨运营域接入控制, 提出多运营域黑/白名单安全求并协议, 实现了隐私保护的各域黑/白名单一致性接入控制. 该协议在半诚实模型下满足对任意 $m-1$ 方合谋的安全性. 实验结果表明, 不管是在局域网还是广域网, 所提方案在不同参数设置下总体运行时间优于现有方案. 此外, 本文还证明了当运营域数量较多时, 所提方案仍相比于现有方案具有优势.

参考文献

- 1 Koisser D, Mitev R, Yadav N, et al. Orbital trust and privacy: Sok on PKI and location privacy challenges in space networks. In: Proceedings of the 33rd USENIX Security Symposium, Philadelphia, 2024
- 2 Oh S, Vasishth D. A call for decentralized satellite networks. In: Proceedings of the 23rd ACM Workshop on Hot Topics in Networks, Irvine, 2024. 25–33
- 3 Tian B, Zhang X, Zhai E, et al. Safely and automatically updating in-network ACL configurations with intent language. In: Proceedings of the ACM Special Interest Group on Data Communication, Beijing, 2019. 214–226
- 4 Goldberg S, Haller M, Heninger N, et al. RADIUS/UDP considered harmful. In: Proceedings of the 33rd USENIX Security Symposium, Philadelphia, 2024
- 5 Chen S Z, Sun S H, Kang S L, et al. Key technologies for 6G integrated satellite-terrestrial mobile communication. Sci Sin Inform, 2024, 54: 1177–1214 [陈山枝, 孙韶辉, 康绍莉, 等. 6G 星地融合移动通信关键技术. 中国科学: 信息科学, 2024, 54: 1177–1214]
- 6 Deng Y, Wu Q, Lai Z, et al. Time-varying bottleneck links in LEO satellite networks: identification, exploits, and countermeasures. In: Proceedings of the 32nd Annual Network and Distributed System Security Symposium, San Diego, 2025

- 7 Larisch J, Choffnes D R, Levin D, et al. Crlite: a scalable system for pushing all TLS revocations to all browsers. In: Proceedings of 2017 IEEE Symposium on Security and Privacy, San Jose, 2017. 539–556
- 8 Li Y, Li H, Liu W, et al. A networking perspective on Starlink's self-driving LEO mega-constellation. In: Proceedings of the 29th Annual International Conference on Mobile Computing and Networking, Madrid, 2023
- 9 Lai Z, Li Z, Wu Q, et al. Mind the misleading effects of LEO mobility on end-to-end congestion control. In: Proceedings of the 23rd ACM Workshop on Hot Topics in Networks, Irvine, 2024. 34–42
- 10 Zhao D, Zhang X, Lee M. Satpipe: deterministic TCP adaptation for highly dynamic LEO satellite networks. In: Proceedings of IEEE Conference on Computer Communications, London, 2025. 1–10
- 11 Liang H, Chen F C, Ji X S, et al. Development status and applied research on mimic technologies for space-ground integration information network. *Sci Sin Inform*, 2019, 49: 799–818 [梁浩, 陈福才, 季新生, 等. 天地一体化信息网络发展与拟态技术应用构想. *中国科学: 信息科学*, 2019, 49: 799–818]
- 12 Henry R, Goldberg I. Formalizing anonymous blacklisting systems. In: Proceedings of the 32nd IEEE Symposium on Security and Privacy, Berkeley, 2011. 81–95
- 13 Hogan K, Luther N, Schear N, et al. Secure multiparty computation for cooperative cyber risk assessment. In: Proceedings of IEEE Cybersecurity Development, Boston, 2016. 75–76
- 14 Burkhart M, Strasser M, Many D, et al. SEPIA: privacy-preserving aggregation of multi-domain network events and statistics. In: Proceedings of the 19th USENIX Security Symposium, Washington, 2010. 223–240
- 15 Liu X, Gao Y. Scalable multi-party private set union from multi-query secret-shared private membership test. In: Proceedings of the 29th International Conference on the Theory and Application of Cryptology and Information Security, Guangzhou, 2023. 237–271
- 16 Gao J, Nguyen S, Trieu N. Toward a practical multi-party private set union. *Proc Priv Enhancing Technol*, 2024, 2024: 622–635
- 17 Dong M, Chen Y, Zhang C, et al. Breaking free: efficient multi-party private set union without non-collusion assumptions. In: Proceedings of the 34th USENIX Security Symposium, Seattle, 2025
- 18 Gao J, Nguyen S, Blanton M, et al. PULSE: parallel private set union for large-scale entities. In: Proceedings of the 2025 on ACM SIGSAC Conference on Computer and Communications Security, Taipei, 2025
- 19 Cramer R, Gennaro R, Schoenmakers B. A secure and optimally efficient multi-authority election scheme. In: Proceedings of International Conference on the Theory and Application of Cryptographic Techniques, Konstanz, 1997. 103–118
- 20 Tang F, Ling G W, Shan J Y. Additive homomorphic encryption schemes based on sm2 and sm9. *Journal of Cryptologic Research*, 2022, 9: 535–549 [唐飞, 凌国玮, 单进勇. 基于国密 SM2 和 SM9 的加法同态加密方案. *密码学报*, 2020, 9: 535–549]
- 21 Tang F, Ling G, Cai C, et al. Solving small exponential ECDLP in EC-based additively homomorphic encryption and applications. *IEEE Trans Inform Forensic Secur*, 2023, 18: 3517–3530
- 22 Topal O A, Karabulut-Kurt G. Physical layer authentication for LEO satellite constellations. In: Proceedings of IEEE Wireless Communications and Networking Conference, Austin, 2022. 1952–1957
- 23 Chu J H, Chen X M. Robust precoding design for inter-satellite cooperation-based low-earth orbit satellite Internet of Things. *Sci Sin Inform*, 2023, 53: 387–401 [褚建杭, 陈晓明. 基于星间协作的低轨卫星物联网鲁棒预编码设计. *中国科学: 信息科学*, 2023, 53: 387–401]
- 24 Jedermann E, Strohmeier M, Lenders V, et al. RECORD: a reception-only region determination attack on LEO satellite users. In: Proceedings of the 33rd USENIX Security Symposium, Philadelphia, 2024
- 25 Chen Y, Liu L, Li Y, et al. Unraveling physical space limits for LEO network scalability. In: Proceedings of the 23rd ACM Workshop on Hot Topics in Networks, Irvine, 2024. 43–51
- 26 Wigchert J, Sciancalepore S, Oligieri G. Detection of aerial spoofing attacks to LEO satellite systems via deep learning. *Comput Netws*, 2025, 269: 111408
- 27 Classen J, Heinrich A, Portner F, et al. Starshields for iOS: navigating the security cosmos in satellite communication. In: Proceedings of the 32nd Annual Network and Distributed System Security Symposium, San Diego, 2025
- 28 Smailes J, Köhler S, Birnbach S, et al. Satiq: extensible and stable satellite authentication using hardware fingerprinting. *ACM Trans Priv Secur*, 2025, 29: 1–35
- 29 He J, Li X, Zhang X, et al. A synthetic data-assisted satellite terrestrial integrated network intrusion detection framework. *IEEE Trans Inform Forensic Secur*, 2025, 20: 1739–1754
- 30 McAmis R, Haas G, Sim M, et al. Short: unencrypted flying objects: security lessons from university small satellite developers and their code. In: Proceedings of the 3rd USENIX Symposium on Vehicle Security and Privacy, Seattle, 2025. 269–278
- 31 Kissner L, Song D X. Privacy-preserving set operations. In: Proceedings of the 25th Annual International Cryptology Conference, Santa Barbara, 2005. 241–257
- 32 Frikken K B. Privacy-preserving set union. In: Proceedings of the 5th International Conference on Applied Cryptography and Network Security, Zhuhai, 2007. 237–252

- 33 Blanton M, Aguiar E. Private and oblivious set and multiset operations. In: Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security, Seoul, 2012. 40–41
- 34 Vos J, Conti M, Erkin Z. Fast multi-party private set operations in the star topology from secure ands and ors. Cryptology ePrint Archive, 2022. <https://eprint.iacr.org/2022/721>
- 35 Goldwasser S, Micali S, Rackoff C. The knowledge complexity of interactive proof-systems (extended abstract). In: Proceedings of the 17th Annual ACM Symposium on Theory of Computing, Providence, 1985. 291–304
- 36 Rabin M O. How to exchange secrets with oblivious transfer. Cryptology ePrint Archive, 2005. <https://eprint.iacr.org/2005/187>
- 37 Ishai Y, Kilian J, Nissim K, et al. Extending oblivious transfers efficiently. In: Proceedings of the 23rd Annual International Cryptology Conference, Santa Barbara, 2003. 145–161
- 38 Vos J, Conti M, Erkin Z. Sok: collusion-resistant multi-party private set intersections in the semi-honest model. In: Proceedings of IEEE Symposium on Security and Privacy, San Francisco, 2024. 465–483
- 39 Pascal P. Public-key cryptosystems based on composite degree residuosity classes. In: Proceedings of International Conference on the Theory and Application of Cryptographic Techniques, Prague, 1999. 223–238
- 40 Shanks D. Class number, a theory of factorization, and genera. In: Proceedings of a Symposium of a Mathematical Society, 1971. 415–440
- 41 Raghuraman S, Rindal P. Blazing fast PSI from improved OKVS and subfield VOLE. In: Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, Los Angeles, 2022. 2505–2517
- 42 Ma J, Zheng Y, Feng J, et al. Secretflow-spu: a performant and user-friendly framework for privacy-preserving machine learning. In: Proceedings of the 2023 USENIX Annual Technical Conference, Boston, 2023. 17–33
- 43 Pagh R, Rodler F F. Cuckoo hashing. In: Proceedings of the 9th Annual European Symposium, Aarhus, 2001. 121–133
- 44 Pinkas B, Schneider T, Tkachenko O, et al. Efficient circuit-based PSI with linear communication. In: Proceedings of the 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, 2019. 122–153
- 45 Ling G, Tang P, Tang F, et al. Privacy-preserving authorized set matching via dishonest majority multiparty computation. IEEE Trans Dependable Secure Comput, 2025, 22: 7614–7631
- 46 Boyle E, Couteau G, Gilboa N, et al. Efficient two-round OT extension and silent non-interactive secure computation. In: Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security, London, 2019. 291–308
- 47 Raghuraman S, Rindal P, Tanguy T. Expand-convolute codes for pseudorandom correlation generators from LPN. In: Proceedings of the 43rd Annual International Cryptology Conference, Santa Barbara, 2023. 602–632
- 48 Neff C A. A verifiable secret shuffle and its application to e-voting. In: Proceedings of the 8th ACM Conference on Computer and Communications Security, Philadelphia, 2001. 116–125
- 49 Haines T, Goré R, Tiwari M. Machine-checking multi-round proofs of shuffle: Terelius-wikstrom and bayer-groth. In: Proceedings of the 32nd USENIX Security Symposium, Anaheim, 2023. 6471–6488
- 50 Shmatikov V, Mitchell J C. Analysis of a fair exchange protocol. In: Proceedings of the Network and Distributed System Security Symposium, San Diego, 2000
- 51 Rivinius M, Reisert P, Rausch D, et al. Publicly accountable robust multi-party computation. In: Proceedings of the 43rd IEEE Symposium on Security and Privacy, San Francisco, 2022. 2430–2449
- 52 Eagle P N J, Galbraith S D, Ong J. Point compression for koblitz elliptic curves. Adv Math Commun, 2011, 5: 1–10

Consistent cross-domain access control for satellite Internet via multi-party private set union

Guowei LING¹, Peng TANG¹, Jingwei HU² & Weidong QIU^{1*}

1. *School of Computer Science, Shanghai Jiao Tong University, Shanghai 200240, China*

2. *School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore 637371, Singapore*

* Corresponding author. E-mail: qiuwd@sjtu.edu.cn

Abstract The low Earth orbit (LEO) satellite Internet is evolving toward a cross-domain collaborative service paradigm. Each administrative domain typically relies on the consistency of local blacklists and whitelists to regulate terminal access. However, inconsistency among these access lists may be exploited by adversaries to launch practical attacks. Although reliable broadcast can synchronize these lists, both the whitelists (representing authorized identities) and blacklists (representing revoked or restricted identities) inherently reveal sensitive information. An adversary can easily infer the roaming trajectories of terminals across domains, leading to severe privacy leakage risks. To address this problem, this paper proposes a secure multi-domain black/white-list union protocol, in which each domain submits only its own identifier set, and the protocol reveals only the global union of valid identifiers without leaking the intersection or the occurrence frequency of any identifier. The proposed protocol adopts a hybrid design of symmetric and public operations, leveraging the SM2 additively homomorphic encryption scheme combined with elliptic-curve-based fast homomorphic decryption to achieve efficient and privacy-preserving black/white-list union, while remaining resilient against collusion attacks. We prove that our protocol is secure under the semi-honest model. Experimental results demonstrate that our protocol significantly outperforms state-of-the-art schemes. For instance, in a WAN setting with 8 domains, our protocol achieves approximately 1.16× to 3.32× speedup compared with existing schemes. The complete implementation of our protocol is available on GitHub at <https://github.com/ShallMate/MPSU>.

Keywords cross-domain satellite Internet, access control, consistency synchronization, privacy-preserving, multi-party private set union