NSMP：一种去中心化、端到端加密通信协议

达哥(@BTCdage)

btcdage@proton.me

Btcdage.rf.gd

摘要：在数字时代，中心化通信平台带来了前所未有的便利，但也引发了对隐私审查、数据主权和单点故障的深切忧虑。本文档提出了一种创新的解决方案——**NextBTC 安全加密留言协议 (NSMP)**。NSMP 利用比特币区块链的去中心化、不可篡改和抗审查特性，构建了一个安全、私密的点对点通信层。本协议在设计上严格遵循**比特币至上主义 (Bitcoin Maximalism)** 的核心原则，旗帜鲜明地支持比特币主链维持其作为纯粹金融账本的定位。因此，NSMP 并非为比特币主链设计，而是部署于一个继承其安全哲学并专注于应用的平行链——**nextBTC**。本协议的实现，旨在用实践证明，复杂的链上应用与维护主链的简洁性可以并行不悖，而非相互妥协。该协议通过将端到端加密的消息分段嵌入到比特币交易的 OP_RETURN 字段中，实现了真正意义上的主权通信。与依赖中心化服务器的传统应用不同，NSMP 的所有数据都由比特币网络本身来保障其持久性和可用性。本白皮书将详细阐述 NSMP 的核心架构、密码学实现、链上交互协议及其相较于现有技术的独特优势。

一、绪论：通信的困境与比特币的哲学

1.1 现有通信模式的挑战

当今主流的即时通讯工具，无论是 Signal、Telegram 还是 WhatsApp，都依赖于中心化的服务器架构。这意味着：

审查风险：服务提供商有能力审查、阻止或删除信息。

隐私泄露：尽管内容可能被加密，但元数据（谁在何时与谁通信）仍然是可被收集和分析的。

服务中断：服务器宕机或公司停止运营将导致通信中断。

数据主权缺失：用户的身份和数据最终由服务商控制。

1.2 比特币：神圣账本与『OP_RETURN』的智慧

比特币网络不仅仅是一个点对点的电子现金系统，其底层区块链本质上是一个全球共识的、不可篡改的分布式数据库。然而，这种强大的能力也带来了一个持续的哲学辩论：我们应该如何使用这个**全球最宝贵的、有限的区块空间**？

我们坚信，比特币主链的首要职责是作为全球的最终价值结算层。它的纯粹性和简洁性是其长期安全和去中心化的根本保障。任何可能导致“状态膨胀”、提高全节点运行成本、从而削弱去中心化的行为，都应被视为对网络根基的侵蚀。

在此哲学指导下，比特币核心开发者在 2014 年引入的 OP_RETURN 及其 80 字节的限制，并非一个技术上的短板，而是**一个**

充满远见的设计哲学。它是一道“防火墙”，在允许必要的元数据锚定的同时，极大地提高了将主链用作通用数据存储的成本和难度，从而有效地保护了主链免受数据滥用的影响。我们的协议，正是建立在对这一设计的深刻认同之上。

二、 设计哲学：分层、职责与 nextBTC 的诞生

2.1 矛盾的统一：应用需求与主链保护

对去中心化、抗审查应用的需求是真实存在的。然而，直接在主链上满足这些需求，无异于要求世界黄金储备库去承担中央银行的所有日常业务。这是低效、昂贵且危险的。

2.2 nextBTC：一个价值的延续

为了解决这一矛盾，我们创建了 nextBTC。这是一个 fork 自 Bitcoin Core v29.x 的平行链。这个版本的选择至关重要，因为它代表了我们对坚守 80 字节限制这一核心价值观的承诺，以区别于后续可能放宽限制的版本。

nextBTC 的使命是：

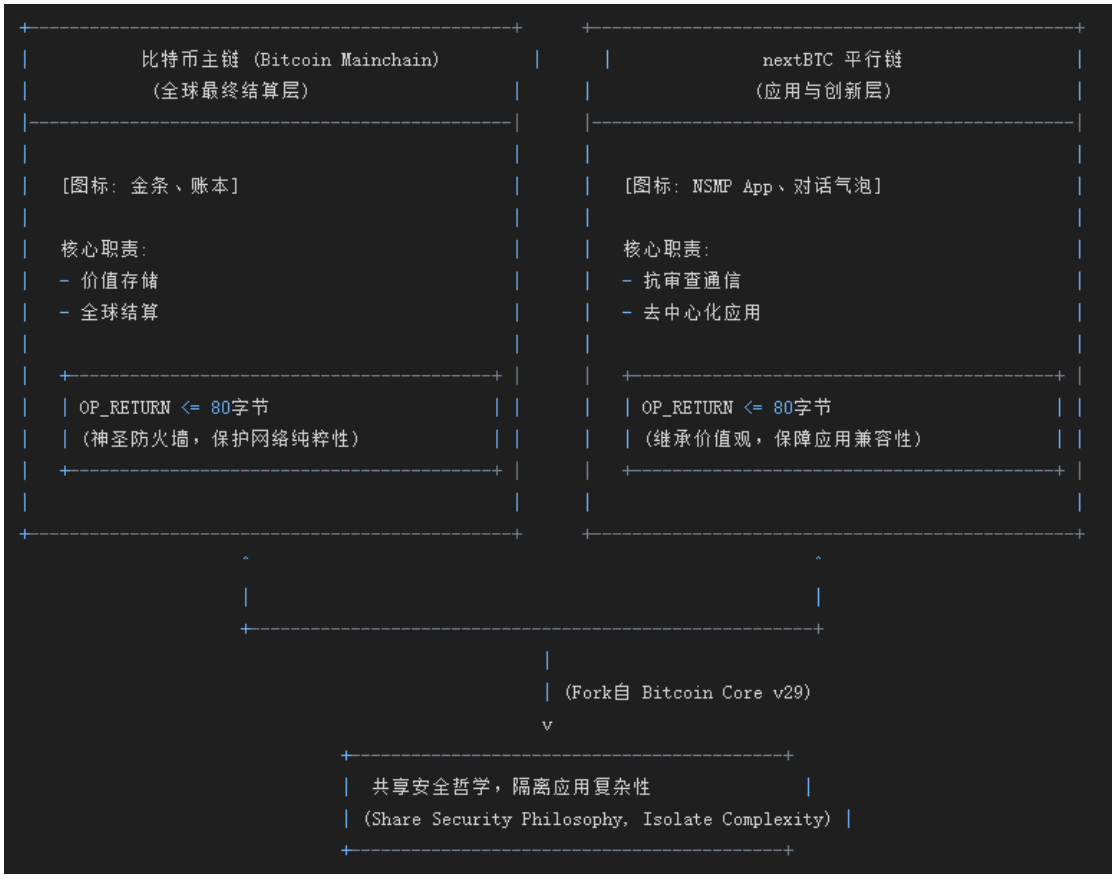
1. **继承**：完全继承比特币的安全模型和核心共识。
2. **隔离**：为 NSMP 这样的数据密集型应用提供一个专属的、高性能的运行环境，使其不会与主链的金融交易争夺宝贵的区块空间。
3. **探索**：作为一个专注于应用创新的平台，探索在不牺牲去中心化

前提下的更多可能性。

2.3 NSMP：一个实践性的论证

NSMP 的设计，本身就是对主链 80 字节限制合理性的一次致敬和论证。其复杂的分段和链上重组机制，直观地展示了在主链上进行此类操作的巨大摩擦力。

因此，NSMP 在 nextBTC 上的流畅运行，恰恰反证了它不应该在主链上运行。这完美地体现了我们的核心论点：保护主链的最佳方式，是为创新应用构建专用的平行层，而非改造主链本身。



三、NSMP 核心架构

NSMP 协议由三个核心层次构成：

身份层 (Identity Layer)：用户的身份并非注册于任何中心化服务器，而是由其自身掌握的、符合 secp256k1 标准的椭圆曲线密钥对（与比特币及 Nostr 协议兼容）来定义。私钥即身份，公钥即地址。

加密层 (Cryptography Layer)：采用先进的混合加密方案，结合非对称加密（ECDH）和对称加密（AES），确保消息的机密性、完整性和不可否认性。

数据持久层 (Data Persistence Layer)：利用比特币交易的 OP_RETURN 输出作为数据载体，将加密后的信息片段永久记录在区块链上。



四、核心算法与密码学协议

NSMP 的安全性基石在于其严谨的密码学设计。以下是消息从发送到接收的完整生命周期。

4.1 消息加密与封装（发送方）

发送一条消息涉及加密、签名和打包三个主要步骤。该流程采用了一种类 ECIES（椭圆曲线集成加密方案）的混合加密模式。

步骤 1：生成共享密钥 (ECDH)

1. 发送方为本次通信生成一个一次性的临时椭圆曲线密钥对

(Ephemeral Keypair)。

2.使用临时私钥与接收方的长期公钥执行椭圆曲线迪菲-赫尔曼(ECDH) 密钥交换算法，生成一个共享密钥 (shared_secret)。

3.为提高安全性，将此共享密钥通过 SHA256 哈希算法处理，生成一个 32 字节的衍生密钥 ecdh_derived_key。此密钥将作为“加密钥匙的保险箱”的钥匙。

步骤 2：对称加密 (AES-256-CBC)

1.生成一个 32 字节的、密码学安全的随机对称密钥 symmetric_key_K。此密钥将用于加密真正的消息内容。

2.使用 ecdh_derived_key，通过 AES-CBC 模式加密 symmetric_key_K，得到 k_enc。

关键实现细节：此过程处理了 PKCS#7 填充。一个 32 字节的密钥在加密时会被填充至 48 字节，因此 k_enc 的最终长度为 IV(16) + PaddedData(48) = 64 字节。这是协议稳健性的关键。

3.使用 zlib 算法压缩原始消息文本，以减小体积。

4.使用 symmetric_key_K,通过 AES-CBC 模式加密压缩后的消息正文。

步骤 3：数字签名 (ECDSA)

1.为确保消息的完整性和发送者身份的真实性，协议将 [临时公钥 + k_enc + 发送方长期公钥 + 加密后的消息正文] 进行拼接。

2.计算拼接后数据的 SHA256 哈希值。

3.使用发送方的长期私钥对该哈希值进行 ECDSA 签名。

步骤 4：数据包构造

最后，将所有组件按以下顺序拼接成一个完整的二进制数据包 full_message_packed：

[临时公钥 (33B)] + [k_enc (64B)] + [发送方公钥 (33B)] + [签名长度 (1B)] + [ECDSA 签名 (动态)] + [加密后的消息正文]

```
A[明文消息] --> B{zlib 压缩};
C[发送方长期私钥] --> D{生成 ECDSA 签名};
E[接收方长期公钥] --> F[ECDH 密钥交换];
G[生成临时密钥对] --> F;
F --> H[衍生出共享密钥 ecdh_derived_key];

I[生成随机对称密钥 K] --> J{AES 加密};
H --> J;
J --> K[生成加密后的密钥 k_enc];

B --> L{AES 加密};
I --> L;
L --> M[生成加密后的消息正文];

subgraph "数据打包"
    G_pub[临时公钥]
    K
    Sender_pub[发送方公钥]
    D_sig[签名]
    M
end

subgraph "签名数据源"
   G_pub_copy[临时公钥]
   K_copy[k_enc]
   Sender_pub_copy[发送方公钥]
   M_copy[加密后的消息正文]
end

G_pub_copy --> N{拼接};
K_copy --> N;
Sender_pub_copy --> N;
M_copy --> N;
N --> O{SHA256 哈希};
O --> D;

G_pub --> P[最终数据包 full_message_packed];
K --> P;
Sender_pub --> P;
D_sig --> P;
M --> P;
```

4.2 链上分段与广播协议

由于 OP_RETURN 的容量限制，上述加密数据包必须被分割。

1.分段头部设计：每个分段都包含一个 11 字节的统一头部，用于链上数据的发现与重组。

接收方公钥哈希前缀 (6 字节)： sha256(接收方公钥).digest()[:6]。用于快速过滤，接收方只需扫描这个哈希即可判断消息是否与自己相关，极大地保护了接收方隐私。

消息 ID (2 字节)：随机生成，用于将属于同一条消息的所有分段关联起来。

分段索引 (2 字节)：从 0 开始的当前分段序号。

总分段数 (1 字节)：消息被分割的总数量。

2.广播：协议为每个分段创建一笔独立的交易，并将 [头部 + 数据块] 作为 OP_RETURN 输出。发送完成后，发送方只需将第一笔交易的 ID (TXID) 安全地告知接收方即可。



4.3 消息检索与解密（接收方）

1.起始交易定位：接收方使用收到的 TXID，从区块链网络获取第一笔交易。

2.身份验证：提取 OP_RETURN 数据，计算自身公钥的哈希前缀，与数据头中的哈希进行比对。若不匹配，则终止流程。

3.链上数据扫描：若匹配，则从头部解析出 消息 ID 和 总分段数。随后，在近期区块中扫描具有相同 [接收方哈希前缀 + 消息 ID] 的其他交易，直到找齐所有分段。

4.重组与解密：

按分段索引顺序拼接所有数据块，还原出完整的 full_message_packed。

执行与加密相反的过程：

1.使用自己的长期私钥和包内的临时公钥执行 ECDH，计算出相同的 ecdh_derived_key。

2.用它解密 k_enc，得到 symmetric_key_K。

3.用 symmetric_key_K 解密消息正文，并用 zlib 解压。

4.提取发送方公钥和签名，对消息数据进行 ECDSA 验签。

5.完成：验签成功后，解密出的明文消息即可安全地呈现给用户。

五、关键特性与优势

极致的抗审查性：消息的载体是 nextBTC 交易，只要网络在运行，信息就无法被阻止或删除。

端到端的隐私保护：采用强大的混合加密，除了指定的接收方，包括矿工在内的任何网络参与者都无法窥探消息内容。

自托管身份主权：用户的身份由自己的私钥定义，无需许可，不受任何组织控制。

数据永久性：消息一旦被写入区块链并获得足够确认，便成为永久记录。

对区块链友好 (Blockchain-Friendly)：通过使用 OP_RETURN，协议避免了造成区块链 UTXO 集合的膨胀，这是一种对全节点资源更负责任的设计。

六、与相关技术的比较

6.1 vs. 中心化通信应用 (Signal, Telegram)

NSMP 的核心优势在于其去中心化。它消除了对可信第三方的依赖，从根本上解决了单点故障和平台审查的风险。

6.2 vs. 比特币铭文 (Ordinals/Inscriptions)

虽然两者都利用比特币技术存储数据，但目标和技术路径截然不同：

目的：铭文旨在创造公开的、可验证的数字资产；NSMP 旨在实现私密的、点对点的保密通信。

存储位置：铭文使用交易的见证数据 (Witness Data)，容量大但会增加 UTXO 集的负担；NSMP 使用 OP_RETURN，容量小但对链上状态无影响。

隐私性：铭文数据是完全公开的；NSMP 数据是完全加密的。

简而言之，铭文是“数字石碑上的公开雕刻”，而 NSMP 是“投入茫茫大海的加密漂流瓶”。

七、结论

NextBTC 安全加密留言协议 (NSMP) 不仅仅是一个应用，它是一种对比特币分层架构思想的忠实践行。它展示了如何在一个价值一致的平行链上，构建一个超越传统金融范畴的安全通信层，而无需对比

特-币主链的核心协议做出任何妥协。

通过严谨的密码学设计和对 nextBTC 的专属适配，NSMP 为那些寻求最高级别隐私、安全和数据主权的个人与组织，提供了一个切实可行的解决方案。更重要的是，它用代码和实践捍卫了一个核心信念：**让凯撒的归凯撒，让上帝的归上帝。**让比特币主链作为永恒的价值基石，让蓬勃的应用创新在专属的平行宇宙中绽放光芒。