Bolt：分散式货币的匿名支付渠道[1]

Matthew Green，Ian Miers

（Information Security Institute，The Johns Hopkins University，Baltimore，21218）
E-mail：{mgreen,imiers}@cs.jhu.edu

摘要 比特币凭借着将交易公开透明地记录在区块链中，即一种解除了交易双方信任要求的全球公共账本而取得不错的成效。然而，随着比特币的举世瞩目的发展，将每一项交易都记录在区块链中也引发了人们对个人私密，消息延迟和系统可扩展性问题的思考。基于最近的建议“建立小额支付渠道”——双方只在发生数据冲突时使用分布式账本——我们在此介绍些关于建立匿名性支付渠道的技术。我们的提议将考虑到既达到安全，实时，私密的支付方式，且大大降低了支付网络的存储负担的方面。同时，我们还特地介绍三种支付渠道提议，其中包括允许人们通过信任度未知的中介进行支付的技术。更重要的是，我们的提议借助已充分研究的技术能够被高效地具体实施。
关键词 比特币，匿名支付渠道

1 介绍

比特币作为一种分散的电子货币已经越来越流行。在比特币中，每一笔交易都被记录在区块链，一种由一群分散平等的人维护的公共交易账本。虽然这项技术在低交易量的情况下已经被证明十分成功，但是依赖全球性分布式账本增加了人们对可扩展性的更深入的思考。由于在比特币中平均每十分钟有一兆字节块区域被加入区块链中，而比特币交易速率被限制在每秒不到十项新交易内。现在，在比特币讨论区中，人们针对几项提议正进行讨论。但没有一家公司可能会生产一种与集中化竞争的交易费率服务，如支付卡网络。一种解决当前问

[1] 收稿时间：2018.01.10

题很有发展潜力的方法是将大量比特币交易移除出区块链在保证系统的分散性和完整性的前提下。而去中心化的主流提议是使用支付通道，以闪电网站和双小额支付通道为例。相比于上传个人交易记录到区块链上，通道支付首先利用区块链建立在双方之间共享的支付交易。双方直接接触以支付款项——调整双方各自的所有份额——并且只在要求结束交易或解决双方争议时联系区块链。在没有直接交易渠道的情况下，双方还可通过中间人进行交易。这种交易渠道类型的主要优势是在没有新增可信任和中心组织的情况下它显著地减少交易量到达区块链。

然而虽然这种支付渠道提供了解决问题的方案，但他们在众所周知的个人私密保护不足上犯难了。尽管支付不经过区块链，但交易的任何一方都可以借助匿名的身份进行交易。通过建立一个渠道来支付，例如 Tor 宽带或网络，一名使用者秘密地将给定渠道的每一项支付和他在这条渠道上的所有其他支付联系起来。通过一个大众的中间体进行支付，这是存在很大问题的——比如货币兑换——由于中间人现在不得不被信任去保持你所有的付款记录。一些提议中，例如闪电网站，曾提议让交易通过多个中间节点来支付，然而这种方法大大增加了支付渠道的复杂性并且甚至在中间体的任一子节点都暴露支付交易信息。

尽管有些技术已经被提出来解决类似比特币货币子类的隐私问题，但这些解决方案没有解决支付渠道的设置问题。这是由于通道的成对结构。即使一个频道的资金来源是匿名的，在相同的渠道内重复

支付在本质上可连接。鉴于一项主要的渠道是网络支付——这通常被描述为一项更为私人的跟踪和在线广告之间的抉择。最后，我们强调对比特币隐私方面的担忧不只是理论上的。一些合资企业为执行分析和监视比特币交易而被建立，很可能是使用从交易处收集的辅助数据。

我们的贡献

bolt 由一套构建隐私保护不相联系的支付的技术渠道来分散货币。当强调了快速私密的去区块链交易，我们的构想加强了在维护分散支付时保护隐私的工作。与早期的提议不同，我们的提案只是模糊了来自中介的参与者身份，我们的提议建立了匿名的直接渠道，这些渠道可以安全、有针对性地解决纠纷，即使商家不知道付费方的身份。在更实际的研究中，我们提供了可以使用非常特殊但良好的研究的加密基元构建的结构实例——不需要昂贵的零知识证明技术，如 zkSNARKs。我们提出了三种构想：

单向通道付款

我们首先展示了如何构建单向支付通道，在这种渠道中，客户付钱给商家，而不透露她的身份，或者允许商家链接在同一频道上进行的交易。我们的建议使用 Camenisch 等人介绍的小型的电子现金模式，但是需要一些新的想法才能在通道设置中工作。最关键的是，我们提出了一种新的机制来实现简洁的开放和关闭，确保在区块链上消

耗的总带宽是不变的，不管在通道上交换的事务数或交换的值是多少。通过将这些渠道与匿名的潜在货币相结合，这种方法将产生完全匿名的非链式交易，可以用于支付诸如 web 浏览或匿名网络带宽等服务。

双向通道

接下来，我们将展示如何实现双向支付通道，使支付可以在客户和商家之间双向流动。这些通道允许各方交换任意的正和负的值，并且在那些经常提供退款的应用程序中很有用，或者在交易发起者不一定是付款人的情况下必须交换价值。我们的技术使用具有特殊协议的签名方案，其中有几个已知的实例。我们的方法的挑战是防止恶意的交易对方使用过时的信息来获得较早的平衡，同时保持该计划的匿名性。

间接渠道

最后，我们将展示如何使用双向支付通道来启用第三方支付，在这种情况下，未知信任度的中中间体当了一个“桥梁”，“允许两个未连接的双方交换价值”。关键的是，中间体既不了解当事人的身份，也不了解交易的金额。这种技术的可用性使得匿名支付通道在实践中可以使用，因为它减少了从 O(M2)到 O(M)的 M 方之间需要的开放通道(以及承诺的资金)的数量。我们现在为我们的构想提供了背景和思考。

1.1 支付渠道的背景

支付渠道是两个参与者之间建立的一种关系，这种关系是在一个分散的、被称为“总部”的货币网络中建立起来的。为了简单起见，我们将以商人和客户的身份来称呼双方，尽管我们注意到(在某些结构中)付款可能在双方的任何一个方向上移动。我们假设支付网络包括一种方法来验证已发布的交易，并根据公共规则解决争议。原则上，这些需求可以通过诸如比特币或以太币等协商一致网络的脚本系统来满足。[2]我们注意到，我们的建议关注的是支付渠道的隐私，而不是底层资金网络的隐私。匿名资金支付渠道,我们建议使用一个以区块链为基础的零现金的保护隐私的支付系统,虽然其他匿名系统可能就足够了。

当双方想要开通一个渠道时，双方就该渠道各自的平衡份额达成一致，我们用 Bmerch0 和 Bcust0 来表示。双方通过向网络支付款项来建立渠道。如果这些交易是正规的，网络则将提交的资金根据协议由第三者暂为保管。客户现在通过与商家交互的无约束行为进行支付。对于一些正或负的整额交易款 i,第 i 个付款可以被视为一个请求来更新 Bcusti:= Bcusti-1 -i 和 Bmerchi:= Bmerchi-1 + i,而交易的唯一限制条件是 Bmerchi >= 0 和 Bcusti>= 0。任意时刻，一方或双方都可以通过将一条通道关闭的消息发送到分布式账本来关闭通道。如果关闭消息表明双方对通道的当前状态存在分歧，则分布式账本会执行一种解决争议的算法来确定最终的渠道余额。在确保各

[2] 我们的技术要求网络验证一个(盲)签名和几个特殊的非交互式零知识证明。启用此功能将需要对有限的比特币脚本语言进行扩展。另一方面，以太网络拥有更丰富的脚本语言，并且有可能执行这些计算。

方都有机会提供其关闭信息后，各方可以使用一种连锁支付方式来恢复渠道余额的最终份额。

任何支付渠道都必须满足两个特殊的要求，我们称之为普遍的仲裁和简洁性：

1。普遍的仲裁。如果两方对共享信道的状态存在分歧，网络可以靠仲裁纠纷，而不需要任何私人信息。

2。简洁性。为了使支付具有可扩展性，发布到分布式账本上的信息必须是小型的——即它不应与渠道的余额、交易数量或交换的金额成线性增长。

后一种特性对于支付渠道来说是至关重要的，因为它排除了使每一个离线支付交易都被发布到分布式账本上，或者将完全的非链支付交互发送到分布式账本上的退化的解决方案。

匿名支付渠道

我们的目标是为支付渠道提供强大的隐私护盾。这才是我们现在讨论的重中之重。首先，支付渠道的性质意味着保护隐私不可能是绝对的。这两个参与者必须意识到一个通道已经建立或关闭，他们必须知晓通道的初始值。此外，我们要求在我们的设置中有一个客户必须负责发起支付，以此知道渠道的瞬时余额和支付历史。因此，匿名支付通道提供的匿名保证可以直观地描述如下:

在收到一些客户的付款后，商家就会了解到一个事实，即在一个开放的通道上发生了有效的支付(一些正值或负值)。而网络只知道某

种平衡的通道已经打开或关闭。

这些担保还延伸到通过一个与客户和商人有开放渠道的中间体进行支付的情况。在这种情况下，我们要求中间体只知道两个使用开放通道的用户之间发生有效支付的事实。

1.2 我们的构想概述

在这项工作中，我们研究了构建匿名支付通道的两种不同的范例。我们的第一个建设是建立在电子现金的基础上，或者说是由Chaum单向结构引入的电子现金模式，它允许从客户到商家的简单付款，同时保留传统支付渠道的匿名性和功能。我们的第二项建设扩展了这些想法，以允许不同价值的支付，在任何一个方向(例如:每一笔付款都可能有正或负的价值，以更复杂的中止条件为代价。最后，我们将展示如何扩展我们的第二个构建，以支持用户通过一个未知信任度的中间体匿名支付的路径支付。

我们现在展示了我们的构造背后的思考。

来自电子现金的单向支付渠道

电子现金计划是一种专门的协议，在这个协议中，被信任的一方被称为银行发行一次性令牌(称为硬币)，客户可以一次赎回。这些协议是实现单向支付通道的自然选择。让我们来考虑一下“一份理想的电子现金计划”。在这个提议中，商人扮演银行的角色，向顾客发出“匿名硬币”，然后把钱还给商人。为了关闭这个通道，客户将剩余

的硬币花在自己身上，并将证据提交给支付网络。商人可以通过提供一枚双掷硬币的证据来质疑顾客的陈述。

这个稻草人的协议存在几个弱点。最明显的是，它并不具有简洁性，因为关闭需要客户将她所有未使用的硬币寄出去。其次，有一个时间问题:商家在客户的资金被网络覆盖之前，不能向客户发放钱包，这一过程可能需要几分钟到几个小时。与此同时，客户必须确保可以在商家没有给他发钱包的情况下取回自己的资金，或者在钱包激活期间终止计划。最后,为了避免客户对框架的攻击(其中一个商人借硬币问题本身,然后指责客户重复花费),我们需要一个电子现金方案与一个叫做网络风险可免责的特定的属性:也就是说,它是可能的任何第三方(在我们的案例中网络)来区分真正的“双花”客户和由欺骗客户假“双花”的商人。

我们单向结构背后的思考

为了解决第一个问题，我们从一个小型电子现金计划开始。由 Camenisch 等人介绍，以下是一种电子现金形式，其中 B 币可以从存储在顾客身上的固定大小的钱包中产生(这里 B 是钱包大小的多项式)。虽然压缩电子现金降低了钱包的存储成本，但它并没有立即为我们的渠道提供一个简洁的关闭机制。在我们的构建中，关键的创新是一种新的机制，它减少了一个单一的固定大小的消息通道关闭，而这是在商家和客户之间增加非链交互的代价。

为了在我们的构想中创建一个支付渠道，客户首先提交一组用来

制定钱包的秘密。这些都嵌入在一个简洁的钱包承诺中，客户将客户的托管资金(以及一个临时的公共签名验证密钥 pkc)发送到支付网络。客户和商家现在参与了一个交互式渠道建立协议，其操作如下。客户首先产生 B 币消费交易，并附加到每一个非交互式零知识证明，每个硬币与钱包承诺挂钩。然后，她使用一个对称的加密方案分别加密每一个产生的交易，这样每个密文 Ci 就会嵌入一个单独的开销交易，以及密文 Ci+1 的解密密钥。客户使用她的秘密密钥分别签署了所产生的密文后，客户将签字的结果发送给商家保管。该方案的一个关键方面是，客户不需要证明任何密文都是格式良好的。

当客户希望关闭一个正在运行的通道并结余 N(0 < N B)时,她计算 j =(B-N)+ 1 并对网络发出签名消息(渠道 ID，j，kj)，kj 是第 j 密文的解密密钥。商家可以使用这个元组来解密每个密文 Cj，•••, CN，从而检测到频道上的进一步支出。如果客户欺骗商家,通过提供一个无效的解密密钥,如果密文解密一个无效的硬币,或者由此产生的交易表明,她已经双花任何硬币,商人可以发布这种作弊的无可争辩的证据到网络上,惩罚客户。

双向的支付渠道

相对于前一个构想，它的限制是它是单向的:所有的支付必须从客户流向商家。虽然这对于许多有用的应用程序来说是有一定的优势的，例如用于网络浏览的微支付，一些支付通道的应用程序需要支付从商家到客户。下面我们将进一步讨论，这种应用程序的一个显著的

例子是第三方支付，即双方通过中间体发送资金，中间体必须增加一个渠道的价值，同时减少另一个渠道的价值。

对于这些应用，我们提出了第二种结构，将现有(非匿名)支付通道的技术与盲签名和特殊的零知识证明结合起来。在现有的支付渠道系统中，客户和商家首先在初始渠道状态上达成一致，客户持有 Bcust0 第三方保管资金契约，商户在此余额契约上提供签名。当客户希望支付商人一个任意的正或负的资金,她进行交互式协议(1)证明先前的知识签名对当前余额Bcusti-1,(2)证明她拥有足够的余额，然后她(3)盲目地从商户中提取一个新的已签署的退款指令，其中包含更新后的余额 Bcusti = Bcusti-1-e。在任何时候，客户都可能会发布最近她的大部分在区块链中的交易，以赎回她的可用资金。

这种方法的主要挑战是防止不诚实的客户保留和使用她的退款指令的早期版本使通道关闭。为了防止这种情况发生，在每次支付期间，客户与商家进行交互，以向之前的状态提供一个撤销指令。只要客户的行为是诚实的，这个撤销指令就永远不会被链接到通道或任何以前的交易。但是，如果客户通过发布过时的退款指令行为不当，商家可以立即检测到这种情况，并将撤销指令提交到网络，作为客户的不法行为的证明(在这种情况下，网络将渠道的余额授予商家)。与电子现金方式不同的是，这一提案从一方可能会提前中止协议的可能性中提出;我们通过使用网络来实现公平来解决这个问题。

从直接支付到第三方支付

作为我们工作的结尾部分，我们展示了双向支付渠道如何被用来构建第三方支付，通过一个普通的、未知信任度的中介，甲方通过一个普通的、未知信任度的中介 I 来支付给乙方，而双方之前都建立了一个渠道。在实践中，这种能力消除了与所有同行保持渠道的需要。我们的建议的关键优势在于，I 不能通过交易链接查找个人用户，也不可以了解在给定的交易中支付的金额。类似地，即使 I 被破坏了，它也不能要求通过它的任何交易。这种技术使得匿名支付通道在实践中可用，只要存在一个高可用的(未知信任度的)中介来路由连接。我们在 4.3 中提供了实施的全部细节。

总结

我们的单向协议提供了与基础电子现金协议类似的隐私保证，并且在闭包中显示了明显的和必要的限制。客户和商家之间的支付是非交互式的，完全匿名的。另一方面，双向支付构建提供了一个稍微弱一点的保证:在协议执行期间，商家可以将客户置于一个无法进行未来交易的状态。这并不能阻止商家通过网络来关闭该频道，但它确实从两方面引起了人们对匿名的关注:

1 商家可以通过诱导人工终止来任意地或是是暂时地减少其他用户的匿名性。

2 商家可以将用户链接到重复的交易序列中，通过在序列中间终止用户。

对于许多传统的商业环境而言，这种中止的后果可能是微乎其微的:无论支付机制如何，商家都无法兑现承诺的商品，而客户几乎肯定会放弃。对于其他设置，比如小额支付，应该考虑这些可能性。在这样的设置中，如果用户的开放通道数量低于最小匿名设置，客户应该扫描网络，以便提前关闭并中止通道。

1.3 论文大纲

本文其余部分将按以下方式进行：在 x2 中，我们给出匿名支付通道的定义。在 x3 中，我们展示了我们计划的构建块。在 x4 中，我们描述了支付通道结构的协议，在 x5 中，我们展示了这些协议的具体实例。最后，在 x6 中我们讨论了相关的内容。

2 定义

注记

以 j 作为安全参数，定义 P(A(a); B(b)) ~ (c; d)表示一项在 A,B 之间进行的协议，a 是 A 的投入，c 是 A 的收入，b 是 B 的收入，d 是 B 的投入。同时，我们会定义 v 为可以忽略不计的功能，用 valmax 表示支付渠道的最高余额，并用一组整数{emin，emax}表示有效付款金额的范围。

2.1 匿名付款渠道

匿名支付渠道（APC）是在通过支付网络进行交互的双方之间建立的一种结构。在本节中，我们将描述匿名支付渠道方案的属性，它是用于建立这些通道的算法和协议的集合。然后我们解释如何使用这些方案在支付网络中构建渠道。我们现在提供一个正式的 APC 计划的定义。

定义 2.1（APC 计划）一个匿名支付渠道方案由一个可能的概率算法（KeyGen;InitC;InitM;退款;反驳;解决）和两个交互式协议（建立;支付）。这些在图 1 有定义。为完整性我们也定义一个可选的作用设置（1）由被信任的党运行为生成参量 pp，例如，一个共同的参考字符串。在某些实例化中，CRS 是不需要的。在这种情况下，我们设置 pp: = 1j[3]

匿名付款渠道

匿名支付渠道方案必须与支付网络结合使用，可以有条件地代管资金，并将这些托管交易资金绑定到某些数据（如比特币分布式账本）。我们现在描述了这些算法和协议是如何用于在支付网络上建立一个通道的。

为了实例化一个匿名支付通道，商人 M 首先生成一个长存的密码对（pkM; skM）~（pp），将其标识给所有客户。商人初始化它的状态 S。客户 C 生成一个暂时的密码对（pkC; skC），用于单个通道。

[3] 期待我们在 x5 中推荐的实例化，我们建议使用基于公共随机性的 CRS。

客户和商家同意各自的初始渠道余额 B0cust;B0merch 他们现在执行以下步骤:

1. 每个缔约方在商定的初始通道余额上执行 InitC 算法，以获得通道令牌 TC;tM.

2. 双方将这些令牌转交给付款网络，连同一笔交易以代管适当的资金。

密钥生成和通道初始化算法:
注册 (pp)。该算法生成一个密码对 (pk; sk), 供每个客户或商家使用。
InitP (pp;B0cust;B0merch;pk;sk)。对属于{c，m}的 p 这个算法是由每方在打开一个通道之前运行的。在输入初始信道平衡、公共参数和密时, InitC 算法输出该方的信道令牌 TP 和相应的秘密 cskP。
在客户 C 和商人 M 之间运行两方协议:
建立 C(pp;tm;cskC) ;M(pp;tc;cskM)。在输入公共参数和每个初始通道指令上, 建立协议激活了以前托管资金的双方之间的通道。如果互动成功, 商家收到建立和客户收到一个钱包 w. 任何一方均可获得不同的失效符号 L。
支付 C(pp; e;wold); M (pp;e;sold)。在输入参数 e, 和来自客户的钱包 wold，商家的现阶段 sold（通常是零）: 如果交易成功客户收到付款成功的消息 RC 和新的钱包 wnew 。商家收到付款成功消息 RM 和更新状态 snew。
渠道关闭和纠纷算法, 由客户和商家分别运行:
退款 (pp;tm;cskC;w). 在输入一个钱包 w, 输出一个客户渠道关闭信息 rcC。
反驳(pp;tc;s;rcC)。在输入商人的当前状态被卖和顾客渠道关闭消息, 输出一个商户通道关闭消息 rcM 和一个更新的商家状态 dnew。
争议解决算法, 由网络运行:
解决 (pp;tc;tm;rcc;rcM)。输入客户和商户的渠道令牌 TC;TM, 连同闭包信息 rcC, rcM (其中任一消息可能为空), 该算法输出的信道平衡 Bmerchnal;Bcustnal

3. 一旦资金已被证实由第三方托管，双方就执行建立议定书以启动付款渠道。如果双方不同意最初的渠道余额，该协议返回 L 并双方可以关闭频道。

4. 如果渠道建立成功，客户按需要多次启动付薪协议，直到一方或双方关闭渠道。

5. 如果客户希望关闭该通道，她将运行退款，并将 rcc 与识别

的通道一起传送到支付网络。[4]

6. 商户对客户的关闭指令进行反驳，以获取商家关闭指令 rcM。

在这一过程结束时，网络运行解决算法，以确定通道余额，并允许各方收集的托管资金的确定份额。

2.2 正确性和安全性

我们现在描述了一个匿名支付渠道方案的正确性和安全性。在这里，我们提供的思考，并在附录 A 提出正式定义。

正确性

非正式地，APC 方案是正确的，如果所有正确生成的参数 pp 且开放余额 BOcust;BOmerch 属于{0;::: valmax}，每一个正确的交易遵循上述范例，总是产生一个正确的结果。即，每个有效的支付协议的执行都能产生成功，而正确的反驳的最终结果则反映渠道的最终余额。

安全

匿名支付渠道方案的安全性由三种程序定义，我们称之为付款，匿名和余额。我们现在提供了对每个属性的非正式描述。

[4] 这里我们假设通道关闭是由客户发起的。在商家希望启动通道关闭的情况下，它可以向网络发送一条特殊消息，请求客户关闭通道。

付款匿名

直观地说，我们要求商家，即使是与恶意客户合作，也不会了解客户超出了协议之外信息的消费模式。在我们对匿名的定义中，延伸了对 Camenisch et al 的定义。商人与二者之一交互或者与 (1) 一系列的预定实施现实协议为顾客 C1;。。;CN，或 (2) 与模拟器 S，即执行客户的支付协议的一部分。在后一种实验中，我们假设一个模拟器，它可以访问实际协议中的参与者通常无法获得的侧信息，例如模拟活板门或随机预定的控制。我们要求模拟器有能力模拟任何客户，而不需要访问客户的钱包，而不知道被模拟的客户的身份。如果没有对方能确定她是否在世界 (1) 或 (2)。我们强调，这个定义意味着匿名，因为模拟器没有关于它正在模拟哪个方的信息。

余额

余额属性包括两个单独的规则，一个是商家，一个是客户。在这两种情况下，假设诚实执行的解决协议，这一属性确保没有共谋的敌对对方组可以从一个渠道提取更多的价值 (1) 部分的初始渠道资金，结合 (2) 的一套对 (或) 对方所做的合法付款。因为商家和客户都有不同接口，所以我们在两个稍微不同的规则中就有了这个属性。在每场交易中，相对的客户都可以访问那些商人的预定，并允许双方建立任意数量的初始余额渠道。然后，在相对选择的付款金额中，对方可能会发起 (导致对方发起) 资金协议的重复。最后，对方可以启动通道关闭，以获得渠道关闭信息 rcC，rcM。如果解决方案的输

出与所提供和支付的总价值不一致，则对方胜利。

3 技术初探

在本节中，我们将回顾一些基本的构造块，我们将在我们的构想中使用。

承诺计划

让Πcommit= (CSetup，Commit，Decommit) 是一个承诺计划，在输入参数后，CSetup 产生公共参数消息 M，和随机硬币 r，Commit 输出承诺 C;而 Decommit 输入参数和元组 (c，m，r) 。如果 C 是对消息的有效承诺，输出 1 否则为 0。在我们的实例化中，我们建议在循环群中使用基于离散对数假设的彼得森承诺方案。

对称加密方案

我们的结构需要一个高效对称加密方案以及一个一次性对称加密方案。我们定义一个对称加密方案Πsymenc = (SymKeyGen，SymEnc，SymDec)，其中 SymKeyGen 输出 "1bit" 答案。我们还利用了一次性加密方案 Πotenc = (OTKeyGen，OTEnc，OTDec)。在实践中，加密方案可以通过将明文编码为循环组 G 中的一个元素并由一个随机的群元相乘来实现。无论是哪种情况，我们的建设都要求这些方案提供 IND-CPA 安全。

伪随机函数

我们的单向构造需要一个伪随机函数 F，它支持高效的知识证明。对于我们的目的这个伪随机函数对多形式的输入空间都是安全的。除了标准的伪属性外，我们的协议还要求伪随机函数还应具有我们称为强前像抵制的属性。对于此属性，如果给定对一个随机种子 s 的 oracle 实现函数 Fs 的访问，则在该函数的域中没有任何人可以和一个输入点 x 和一对 (s0; x0)，这样 fs (x) = Fs0 (x0)，或者说概率很低。我们计划用 Dodis-Yampolskiy 脉冲实例化 F，公共参数是一组 g 的素数阶 q 与发生器 g。种子是一个随机值 s 属于 Zq 和函数是计算为 fs (x) = gˆ1 / (s + x) x 在一个多形式集中。在附录 D 中，我们展示 Dodis-Yampolskiy 脉冲满意的强前像抵制特性。

签名和有效的协议

我们的方案利用签名方案 Πsig = (SigKeygen; Sign;Verify) ，如 Camenisch 和 Lysyanskaya 所建议的。这些方案的特点是:(1) 这是一种协议，用户在承诺的价值上获得签名，而不让签名人了解有关消息的任何内容，以及 (2) 证明对签名知识的协议。在文献中提出了这些特征码的几个实例，包括基于强 RSA 假设 的构造和双线性群中的各种假设。对于安全性，我们假定所有签名都满足了在选择的消息攻击 (EU CMA) 下存在的伪造的属性。

非交互式零知识证明

我们使用几个标准结果来证明已提交值的结果，例如 (1) 已提交值的内容证明,(2) 证明已提交的值在范围内。当提到以上的证明，我们将使用 Camenisch 和 Stadler 的符号。例如，Pok{f (x; r): y = gˆx*hˆr，(1<= x<= n)}表示对整数 x 和 r 的知识的零知识证明，这样 y = gˆx*hˆr 且 1<= x<= n。所有不包含在 () 的值被认为是已证实的。我们的协议需要一个提供仿真萃取的证明系统，这意味着存在一个高效的证明提取器 (在规格 c 情况下，如使用模拟 CRS) 可以提取对方使用的证明来构造证明，即使对方也提供了模拟证明。在实践中，我们可以进行这些证明非使用各种电子高效证明技术。

4 协议

在本节中，我们介绍了我们的主要贡献，其中包括三协议实施匿名支付渠道。x4.1 中的第一个协议是一种基于现金技术的单向支付通道。 我们的第二个构想在 x4.2 中是允许双向支付，与一个更复杂的协议处理中止。最后，在 x4.3 中，我们提出了一种 三方付款的方法，其中双方通过中间人发送付款。

4.1 单向支付渠道

我们的第一个构想是以电子现金为模型建设 Camenisch et al 以实现高效和简洁的单向支付渠道。现在我们简要介绍一下这一结构。

小型电子现金

在小型的现金方案中，客户取消了能够生成 B 币的固定模式钱包。客户的钱包基于元组 (k; sk;b): k 是一种 (交互式生成) 的伪随机函数 F，sk 是客户的私钥，B 是钱包中的硬币数。一旦由商家签署，这个钱包可以用来生成多达如下的 B 硬币: 第 i 枚硬币包括一个元组 (s;t;π) 其中 s 是一个序列号计算为 s =Fk (i);T 是一个双花标签。如果相同的硬币花两次，就可以和双花标签结合起来，揭示客户的密匙 pk (或 sk);而 π 是以下语句的非交互式零知识证明:

1. 0 <=i<=B
2. 证明者知道 sk。
3. 证明者在钱包上有签名 (k; sk;B)。
4. 对于已签名的钱包，(s;T) 是正确的结构对。

这一结构确保了证明者可以瞬间检测到双倍开销，因为这两个交易将共享序列号 s。[5]证明者可以通过组合双花标记来恢复公钥。同时，个人硬币的花费不能与对方或用户挂钩。Camenisch et al. 展示了如何使用随机 oracle 模型中的强 RSA 或双线性假设，利用签名和证明技术来构建高效验证。随后的文章提出了标准模型中的高效证明。

[5] 在最初的契约电子现金建设中，关键 k 是使用客户和银行之间的交互协议生成的，这样，一方的诚实行为确保 k 是一致随机的。在我们的修改后的协议中，k 只会被客户选择。如果 PRF 是确定性的，并且证明系统是可靠的，那么这就不支持双开销。

实现简洁的关闭

让我们回想一下在单向支付渠道中使用小型现金的思考（见x1.2)。在这项建议中，商人扮演银行的角色，并向顾客发出一个 B 币的钱包，然后她可以（匿名）花回商家。为了关闭一个渠道，客户只需向自己花费任何未使用的硬币，从而证明给商家看，她不保留在渠道上的消费能力（因为任何随后的尝试消费这些硬币将被商家承认为双花)。不幸的是，小型的现金提供一个简洁的钱包，这不会立即产生一个简洁的协议，来关闭通道——由于客户不能简单地仅泄露钱包的秘密，而不会损害以前在频道上花费的硬币的匿名性。我们需要一个机制来简洁地揭示钱包里的硬币的一小部分，而不把它们全部暴露出来。同时，我们希望避免复杂的证明（例如，以 O (B) 为尺度的证明成本)。[6]

我们的做法是使用商家存储必要的信息来验证通道关闭。这需要对 Camenisch et al 的小型现金方案进行若干修改（需要对我们在x4.1.1 提供的方案进行新的分析)。首先，我们设计客户的 InitC 算法，使 PRK 种子 k 是由客户单独生成的，而不是客户和银行（商家）交互产生的。客户现在承诺钱包的秘密性并产生 wCom，并嵌入到客户的渠道指令 TC: = (wCom; pkc)，其中 pkc 是一个被证实的签名密码。当建立协议，以获得商户的签名 wCom，客户为商家提供了一系列签署的密文(C1;:;CB)，其中每个包含一个硬币花元组的形式

[6] 实际上，另一种建议是使用链式构造来构造硬币序号，在这里，每个 si 都被计算为前一个事务中使用的键的单向散列。这将允许客户通过在一个事务上发布一个秘密来撤销该通道。不幸的是，用标准的零知识技术证明 si 的正确性，需要 O(B)证明成本，而且，使用我们在这项工作中推荐的特殊的零知识证明技术似乎并不容易。

(s;t;π) 其中 π 是相同的正常小型现金证明，但仅仅证明 s;T 关于 wCom 是正确的 (尚未签署的商人)。这些密文是结构化的，以便为 第 j 项密文显示的密钥也将打开每个后续的密文。

这种方法的主要特点是，在通道打开时商家不需要知道这些密码文本是否真正包含有效的证据。为了在一个通道中显示剩余的 j 币，客户会显示密码文本的密钥，这使商家能够解锁所有剩余的硬币，并验证他们的承诺已经嵌入在客户的渠道指令。如果任何密文无法打开，或者如果附上的证明无效，商家可以很容易地证明客户渎职，并获得渠道的余额。这只需要对称加密和一种方法来连接到对称加密密钥｛两者都可以很容易地从标准构建块中构建。此外，我们的方案还需要一个一次性加密算法 OTEnc，其中钥的算法也是处于伪随机函数 F 的范围。

安全分析

定理 4.1

在假定 (1) F 为伪随机并提供强前像抵制的假设下，单向渠道方案满足了匿名和余额的性质，(2) 承诺方案是安全的，(3) 零知识系统是健全的，零知识，(4) 在选择的消息攻击下，签名方案是存在伪造的，签名提取是未知的，(5) 对称加密和一次性加密方案都是安全的。

4.2 双向支付渠道

上述结构的关键限制在于它是单向的，并且只支持从客户到商家的付款。此外，它只支持已证实价值的硬币。在本节中，我们描述了一个结构，它启用双向支付渠道，功能严谨的关闭，小型的钱包，并允许单一运行的支付协议转移任意值（受最大付款金额限制）。

在这个结构中，客户的钱包结构类似于以前的结构：它包括 BOcust，和一个随机的钱包公开签名密钥 wpk。当商家在内容上提供一个未知签名时，钱包就被激活了。签名的钱包是在前一协议中获得的，当交易被确认时，承诺被放置在锚交易和签名中。然而，客户和商家不使用一系列单独的硬币进行付款，而是简单地交换一个现有的签名钱包，价值 Bcust 为一个值得 Bcust 的新签名钱包（并嵌入一个新的钱包公钥 wpknew）。请注意，在这个结构中可以是正数或负数。客户使用零知识证明和签名与高效协议证明新的被请求的钱包的内容正确地被构造。在交易结束时，客户向商家透露 wpkold，以保证此钱包不能再用一次。旧的钱包被客户签名的已吊销消息与 wsk 对应的私钥失效。关闭渠道包括顾客张贴一个有效的钱包签字由商人对区块链。

这一结构的挑战是同时使现有的钱包失效，并签署新的。如果商家在旧钱包失效之前签上新的钱包，那么顾客可以在旧钱包里保留资金，同时继续使用新的。另一方面，如果商家在签署新的旧钱包之前可以将其作废，如果商家拒绝签署新的钱包，客户就没有办法关闭这个通道。

为了解决这个问题，我们把钱包分开用于交互式支付从过帐到执行通道关闭的值，并使用两个阶段协议获取每个这些值。而不是透露最近的钱包 w，C 关闭的渠道使用退款指令 rt 包含 Bcust，当前钱包的公钥，和签名的商家。

安全分析

正如我们在 x1.2 中指出的，双向协议的主要限制是恶意商家可能会中止协议。该协议的性质确保了客户不会因为这种中止而失去资金的风险，因为她可以简单地提供她的退款指令 rtw0 到 区块链，以恢复她的余额。因此，主要的限制是客户的匿名。一个恶意的商人可以使一个客户无法继续消费，并使她必须关闭她的渠道。这隐含地将付款链接到渠道一个问题，只有有限的关注，如果渠道是通过匿名货币支付。

更令人关注的是，恶意商家可能会使用中止来减少系统的匿名性集，导致多个通道进入非功能状态。在实际操作中，此攻击将在付款网络上生成一个可见的信号，从而使客户可以使用它来停止付款。然而，在我们的安全证明的范围内，我们通过简单的方法来解决这一问题，即在支付协议期间防止敌对商家中止。

定理 4.2

双向渠道方案在满意的约束条件下，在不终止支付协议的情况下，并假定 (1) 承诺方案是安全的，(2) 零知识系统仿真可提取性

和零知识，(3) 在选择的消息攻击下存在伪造的盲签名方案，(4) 在一次选择的消息攻击下，一次性签名方案存在伪造。

4.3 双向第三方付款

上面的双向构造的主要应用之一是启用第三方付款。在这些付款中，一方首先通过某一中间人给第二方 b 提供一些正价值的付款，并且 a 和 b 有开放渠道。在这种情况下，我们假设 A 和 B 作为客户和商家的渠道建立，而我扮演的角色是中间体。我们的目标是，在我不知道参与者的身份，或被转移的金额（她可以从她的渠道状态中学到的信息之外），她也不相信，以保障参与者的资金。这一建设与现有的匿名支付渠道方案形成了对比，给出了链 A 到我到 B，中间体总是了解数量和参与者。

在链接支付渠道的挑战是使用付款原子。一旦我已经支付给 B，则付款人只需要支付给中间体我。但是，如果 A 未能完成付款，这当然会使中间体处于风险中。同样，付款人也不一定会把她的资金输给一个未知信用度的中间人。没有纯粹的加密解决这个问题，因为它在本质上公平交换，这是个在多方协议中广泛研究的问题。

然而，我们有已知的技术，使用区块链 s 调解中止。

继续说 x4.2。付薪协议在两个阶段发生。第一部分是交换退款指令，这可用于回收托管资金。第二阶段为以后的付款生成一个匿名钱包。为了保证从一个链接的交易 A 到我到 B 是安全的，我们只需要确保两阶段的初始阶段都能完成或以原子的失败告终。

我们通过在退款指令中添加条件来实现这一点。这些条件是网络的简单语句，用于评估在解析协议期间检查标记的情况。特别地，为了防止如果付款人 a 没有交付相应的支付给我，而收件人 b 从我获得资金，我们介绍以下条件为 b 的退款指令:

1. B 必须在先前的钱包上出示吊销信息 (即使用 wsk 的签名)。

2. a 未将包含 wsk 的吊销指令张贴到分类帐中。

按条件 (1)，一旦 B 强制了我到 B 的付款渠道，则 A 到我不能被撤销，因为我有吊销指令。按条件 (2) 如果 A 到我已经被撤销了，B 不能强制我到 B 的付款因为 wpk 已经在分布式账本上了。

隐藏付款金额

我们的第三方支付构想也提供了一个额外有用的功能。因为在交易中我只是一个被动的参与者，并没有保持状态的任何一个渠道，没有必要让我了解到具体金额。如果 A 和 B 双方同意某一数额 (即双方在各自的渠道中都有资金)，则双方都不需要向我透露: 我只需要保证资金的余额是正确的。

要隐藏付款金额，我们必须从图 3 的付薪协议中修改用于构造 2 的校样语句。客户 A 现在承诺并在计算付款时使用此值作为机密输入，而不是向商家透露。同时，在支付为了调整 b 的钱包而进行的协议，b 现在证明他的钱包已经被调整过了。为此，我们将付薪协议中的证据更改为绑定到承诺但不进行透露:

接着 A 能向我证明双方付款取消或 (如果费用是非零的)，通过证明留下 B 与费用额外资金:

协议

我们现在将 A 和 B 的钱包更新的过程合并成一个单一的

协议，我们在图 4 中概述。具体步骤如下:

1. B 承诺并进行可变支付支付协议的第一步移动 (图 3)(上面所述的是隐藏式余额证明)，并向其新的钱包状态 wCom0b，证明了钱包的正确性，b 和承诺的随机性.

2. A 完成它自己的第一步移动，产生 wCom0a;并且另外计算一个证明到它的原始的钱包和新的钱包承诺的正确状态。它发送这些和 B 的新的钱包承诺和 A 到 i。

3. 我在核实证明之后，签发一个退款指令给它的新钱包 rtwa0 和 B一个条件退款指令 crtwrev0 作为其新的钱包。此标记嵌入了 B 必须生成的条件 a 一个旧钱包的吊销指令

4. A 在可变支付支付协议中完成第二步，以生成 revwa 旧钱包的吊销指令。它发送和 crt rev0 到 B。

5. B 完成第二步，生成 revwb 的旧钱包的吊销指令。验证它现在有一个有效的退款指令通过验证 revwa，它发送 revwa;revwb 至 i。

6. 我完成了与 A 和 B 单独的可变付款支付协议的剩余的变动，在他们的新的钱包给每个一个隐藏的署名。

安全和中止条件

我们省略了对该协议的完整安全分析。在这个构造中的一个挑战是，恶意的我可以有选择地中止协议在交易中的可能性。虽然系统不允许我窃取资金，但我可以迫使 A 和 B 向网络传输消息，以收回他们的资金。不幸的是，这似乎是基本的不可避免的。

我们注意到，匿名威胁在实践中受到限制，因为渠道本身可以用匿名货币提供资金，因此，将两个单独的通道连接起来并不能显示参与者识别。更重要的是，由于中间体只能在每个通道上使用此中止技术，因此商家没有可能在同一通道上链接单独的付款。最后，执行此中止技术的中间体将在网络上生成公共证据，从而允许参与者避免恶意中间体。

4.4 隐藏付款余额

上述每个结构都有一个隐私限制：当通道关闭时，将显示每个支付通道的余额。虽然个人可以通过使用匿名货币机制来为渠道提供资金来保护自己的身份和初始渠道余额，但有关渠道余额的信息会向网络泄露有用的信息。不过，我们注意到，在无争议的通道关闭的情况下，即使是这些信息也可以从公众中隐藏。在通道关闭时，客户会对最终通道余额作出承诺，并提供零知识证明，说明她拥有有效的通道关闭指令（例如在我们双向结构中，通道余额的签名）。在诸如 Zerocash 这样的系统中，向商家和客户支付兑换硬币的费用可以包括一份附加声明：在这笔交易中支付的金额与承诺一致，并且不超过

最初的资金创建通道的交易。我们将此类构想的具体细节留给未来的文章。

5 具体实例化

在实践中，我们预计，我们的匿名支付渠道建设将部署在一个支付网络，已经支持分散的匿名支付的顶部。这个网络的一个选择是 Zerocash 系统，虽然基于硬币混合的其他系统也可能是足够的。这些网络目前正处于商业发展之中。我们注意到，除了匿名资金的要求外，我们的协议不能在一成不变的的比特币中进行实例化：比特币的脚本语言太有限，无法评估高效盲签名、(大多数) 承诺计划，或者需要在增量通道方案。但是，新的合同网络 (如 Ethereum) 是一个可扩展平台，上面的协议可以被实例化。或者，可以将简单扩展添加到现有的支付网络中，以验证零知识证明和签名。

我们的支付渠道方案需要一个带有高效协议的签名方案，以及一个支持零知识证明的适当的 PRF。对于基于高效的单向现金方案的实例化，我们将读者介绍给 Camenisch et al. 和 Belenkiy et al. 的文章。这些文章演示如何实例化小型现金高效使用双线性组，高效数论 PRFs 和签名与高效协议。

双向支付方案的具体实例化需要一个承诺方案、一个具有高效协议的签名方案以获得盲签名，以及以下语句的零知识证明系统：

1. 两个由公共值表示的整数。
2. 证明知道对承诺中的价值的签名。

3. 已提交的整数在公共范围内。

这些组件中的每一个都可以用快速的原语和零知识证明来实例化，并且需要最小的计算来证明和开销。我们将读者介绍以了解更多关于证明技术的细节。我们建议使用彼得森承诺和基于双线性配对的签名方案，如 Camenisch 的方案。在该方案中，签名生成和证明要求每个操作的组操作少于 20 个，每个操作的平均成本为 1 毫秒。

我们注意到这些原语的速度足够快，以至于协议的数量至少比 Zerocash 中使用的 zkSNARK 证明要快两倍。这些证明需要比 Zerocash zkSNARKs 更宽的带宽，但它们只在当事方参与争端时才会发布到区块链（在单向协议中）。它们永远不会在双向支付协议中被发送到区块链。

6 相关内容

比特币的匿名和扩展

一些文章建议对比特币进行额外的隐私保护。Zerocoin，Zerocash 和相似的文章通过使用复杂零知识证明提供强的匿名性。一个单独的文章线试图通过混合交易（例如 CoinJoin ，CoinShu e，CoinSwap）来增加比特币的匿名性。像比特币一样，这些构造都要求所有的交易都存储在区块链。最后，最近的文章提出了将概率支付作为一种替代支付机制。

在支付渠道的隐私 Heilman et al.提出了一种区块链匿名交易的类型，以及用于非链支付的构造。这些方案只需要一个盲签名协议，

使得它们易于在比特币中部署。但是，非链协议不提供双方之间的匿名支付渠道。相反，在现有的匿名支付渠道网络中，这是各方保护其身份的一种方式。最后，他们的方案（像我们的起初建议）是基于现金的指令，不允许可变金额的高效转移。

闪电匿名限制

闪电网络不提供对通道参与者之间的付款匿名性，也就是说，商家可以看到启动付款的每个客户的通道标识。但是，该协议包括对路径支付的一些有限的匿名保护。这些操作的原理与洋葱路由网络类似，通过使用多个非密谋的中间体来模糊路径的原点和目标。不幸的是，这个建议从共谋问题: 给了链 A 到 I1 到 I2 到 I3 到 b，只有 I1 和 I3 必须串通才能恢复 A 和 B 的身份，因为路径上的所有交易共享相同的哈希 Timelock 协定 ID。此外，这种安全机制如果存在一个具有高效路径多样性的网络，这些保护是可行的。在闪电支付网络中，路径路由的实际可行性是一些争论的主题，因为大量的资金将被捆绑在维护开放渠道。更有可能的是，部署的通道将依赖于一个星型拓扑结构，客户和商家通过少数高可用性的一方进行交互，这就是我们在我们的建设中所解决的情况。

7 结论

在这项文章中，我们展示了如何在两个互不信任的当事人之间建立匿名支付渠道。我们的协议可以使用高效的加密基元来实例化，而

不受信任的第三方和（在许多实例中）不受信任的设置。任意价值的付款可以直接在当事人之间进行，或者通过中间连接，既不了解参与者的身份，也不知道所涉及的金额。再加上一个分散的匿名付款计划为渠道提供资金，他们在没有一个可信任的银行的情况下为私人即时匿名支付。

我们留下了两个主要的开放性问题。首先是为了研究扩展第三方支付协议以支持由三方组成的任意路径的必要细节。其次，我们没有考虑到在不暴露网络的渠道余额的情况下执行付款解决的问题（在发生争议时）。

参考文献

[ADMM14] Marcin Andrychowicz, Stefan Dziembowski, Daniel Malinowski, and Lukasz Mazurek. Secure multiparty computations on Bitcoin.
In Security and Privacy (SP), 2014 IEEE Symposium on, 2014.

[BCC+08] Mira Belenkiy, Jan Camenisch, Melissa Chase, Markulf Kohlweiss, Anna Lysyanskaya, and Hovav Shacham. Delegatable anonymous credentials.
Cryptology ePrint Archive, Report 2008/428, 2008.http://eprint.iacr.org/2008/428.

[BCKL08] Mira Belenkiy, Melissa Chase, Markulf Kohlweiss, and Anna Lysyanskaya. P-signatures and noninteractive anonymous credentials.
In TCC 2008, 2008.

[BCKL09] Mira Belenkiy, Melissa Chase, Markulf Kohlweiss, and Anna Lysyanskaya. Compact E-Cash and Simulatable VRFs Revisited.
In Pairing-Based Cryptography '09, 2009.

[BDJR97] M. Bellare, A. Desai, E. Jokipii, and P. Rogaway. A concrete security treatment of symmetric encryption.
In Foundations of Computer Science '97, pages 394{403, 1997.

[BGDM+10] Jean-Luc Beuchat, Jorge E Gonzalez-D_x0010_az, Shigeo Mitsunari, Eiji Okamoto, Francisco Rodr_x0010_guezHenr_x0010_quez, and Tadanori Teruya. High-speed software implementation of the optimal ate pairing over barreto{naehrig curves.
In Pairing-Based Cryptography-Pairing 2010. 2010.

[bit] Bitcoin Wiki: Maximum transaction rate.
https://en.bitcoin.it/wiki/Maximum_transaction_rate.

[BK14] Iddo Bentov and Ranjit Kumaresan. How to use Bitcoin to design fair protocols.

In Advances in Cryptology{CRYPTO 2014. 2014.
[Blo14] Block Chain Analysis. Block chain analysis.
http://www.block-chain-analysis.com/, 2014.
[blo16] Block size limit controversy.
https://en.bitcoin.it/wiki/Block_size_limit_controversy, February 2016.
[Bou00] Fabrice Boudot. Efficient proofs that a committed number lies in an interval.
In Advances in Cryptology EUROCRYPT 2000, 2000.
[Bra93] Stefan Brands. Untraceable online cash in wallet with observers.
In Advances in Cryptology CRYPTO 93, 1993.
[Bra97] Stefan Brands. Rapid demonstration of linear relations connected by boolean operators.
In EUROCRYPT 97, 1997.
[CC+08] Jan Camenisch, Rak Chaabouni, et al. Efficient protocols for set membership and range proofs.
In Advances in Cryptology-ASIACRYPT 2008. 2008.
[CDS94] Ronald Cramer, Ivan Damgard, and Berry Schoenmakers. Proofs of partial knowledge and simplied design of witness hiding protocols.
In CRYPTO '94, 1994.
[CFN90] David Chaum, Amos Fiat, and Moni Naor. Untraceable electronic cash.
In Proceedings on Advances in cryptology, 1990.
[Cha83] David Chaum. Blind signatures for untraceable payments.
In Advances in Cryptology, 1983.
[Cha15] Chainalysis. Chainalysis inc.
https://chainalysis.com/, 2015.
[CHL05] Jan Camenisch, Susan Hohenberger, and Anna Lysyanskaya. Compact e-cash.
In Advances in Cryptology EUROCRYPT 2005.
[CL02] Jan Camenisch and Anna Lysyanskaya. A signature scheme with efficient protocols.
In Security in communication networks. 2002.
[CL04] Jan Camenisch and Anna Lysyanskaya. Signature schemes and anonymous credentials from bilinear maps.
In Advances in Cryptology{CRYPTO 2004, 2004.
[CNS07] Jan Camenisch, Gregory Neven, and Abhi Shelat. Simulatable adaptive oblivious transfer.
In EUROCRYPT '07, 2007.
[CS97] Jan Camenisch and M. Stadler. Efficient group signature schemes for large groups.
In CRYPTO '97,1997.
[DFKP13] George Danezis, Cedric Fournet, Markulf Kohlweiss, and Bryan Parno. Pinocchio coin: Building Zerocoin from a succinct pairing-based proof system.
In Proceedings of the First ACM Workshop on Language Support for Privacy-enhancing Technologies, PETShop '13, 2013.
[DW15] Christian Decker and Roger Wattenhofer. A fast and scalable payment network with Bitcoin duplex micropayment channels.
In Stabilization, Safety, and Security of Distributed Systems, 2015.
[DY05] Yevgeniy Dodis and Aleksandr Yampolskiy. A verifiable random function with short proofs and keys.
In PKC '05, pages 416{431, 2005.

[Ell13] Elliptic. Elliptic enterprises limited.
https://www.elliptic.co/, 2013.
[eth] The Ethereum Project.
https://www.ethereum.org/.
[Gro06] Jens Groth. Simulation-Sound NIZK Proofs for a Practical Language and Constant Size Group Signatures, pages 444 459.
Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
[GS] Jens Groth and Amit Sahai. Efficient non-interactive proof systems for bilinear groups.
[HBG16] Ethan Heilman, Foteini Baldimtsi, and Sharon Goldberg. Blindly Signed Contracts: Anonymous On-Blockchain and Off-Blockchain Bitcoin Transactions. In BITCOIN '16, 2016.
[Max13] Gregory Maxwell. CoinJoin: Bitcoin privacy for the real world. Available at https://bitcointalk.org/index.php?topic=279249.0, August 2013.
[MGGR13] Ian Miers, Christina Garman, Matthew Green, and Aviel D. Rubin. Zerocoin: Anonymous distributed e-cash from Bitcoin.
In Proceedings of the 2013 IEEE Symposium on Security and Privacy, SP '13, 2013.
[MPJ+13] Sarah Meiklejohn, Marjori Pomarole, Grant Jordan, Kirill Levchenko, Damon McCoy, Georey M. Voelker, and Stefan Savage. A stful of bitcoins: Characterizing payments among men with no names.
In Proceedings of the 2013 Conference on Internet Measurement Conference, IMC '13, 2013.
[Pac15] Chris Pacia. Lightning Network skepticism.
https://chrispacia.wordpress.com/2015/12/23/
lightning-network-skepticism/#more-3249, December 2015.
[PD16] Joseph Poon and Thaddeus Dryja. The Bitcoin Lightning Network: Scalable Off-Chain Instant Payments.
https://lightning.network/lightning-network-paper.pdf, January 2016.
[Ped92] Torben Pryds Pedersen. Non-interactive and information-theoretic secure verifiable secret sharing.
In CRYPTO '92, 1992.
[PGHR13] Brian Parno, Craig Gentry, Jon Howell, and Mariana Raykova. Pinocchio: Nearly practical verifiable computation.
In Proceedings of the 34th IEEE Symposium on Security and Privacy, Oakland '13, pages238 252, 2013.
[Ps15] Rafael Pass and abhi shelat. Micropayments for decentralized currencies.
In ACM CCS '15, pages207 218, New York, NY, USA, 2015. ACM.
[Rat16] John Ratcli. The Lightning Network is so great that it has all kinds of problems.
http://codesuppository.blogspot.com/2016/02/the-lightning-network-is-so-great-that.html, February 2016.
[RS13] Dorit Ron and Adi Shamir. Quantitative analysis of the full bitcoin transaction graph.
In Financial Cryptography '13, 2013.
[SCG+14] Eli Ben Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. Zerocash: Decentralized anonymous payments from bitcoin.
In IEEE Security and Privacy, 2014.
[Sch91] Claus-Peter Schnorr. Efficient signature generation for smart cards.
Journal of Cryptology, 1991.

[Tow15] Anthony Towns. Better privacy with SNARKs.
https://lists.linuxfoundation.org/pipermail/lightning-dev/2015-November/000309.html,
November 2015.