Сеть Axelar:

Соединение приложений с блокчейн экосистемами

Проект 1.0
январь 2021 г.

Анотация

Появляются многочисленные блокчейн-экосистемы, которые предоставляют уникальные и отличительные функции, привлекательные для пользователей и разработчиков приложений. Однако связь между экосистемами очень редкая и фрагментарная. Чтобы приложения могли беспрепятственно взаимодействовать между экосистемами блокчейна, мы предлагаем Axelar. Стек Axelar предоставляет децентрализованную сеть, протоколы, инструменты и API, которые обеспечивают простую межсетевую связь. Набор протоколов Axelar состоит из протоколов трансграничной маршрутизации и передачи. Децентрализованная открытая сеть валидаторов питает сеть; любой может присоединиться, использовать его и участвовать. Византийский консенсус, криптография и механизмы поощрения предназначены для достижения высоких требований к безопасности и живучести, уникальных для межсетевых запросов.

1 Введение

Системы блокчейн быстро набирают популярность и привлекают новые варианты использования для токенизации активов, децентрализованных финансов и других распределенных приложений. Несколько основных платформ, таких как Ethereum, Monero, EOS, Cardano, Terra, Cosmos, Avalanche, Algorand, Near, Celo и Polkadot, предлагают различные функции и среды разработки, которые делают их привлекательными для различных приложений, вариантов использования и конечных пользователей.[5, 11, 4, 21, 20, 23, 24 19, 6, 14, 25]. Однако полезные функции каждой новой платформы в настоящее время предлагаются менее чем 1% пользователей экосистемы, а именно владельцам собственного токена на этой платформе. Можем ли мы позволить разработчикам платформ легко подключать свои блокчейны к другим экосистемам? Можем ли мы позволить разработчикам приложений создавать наилучшую платформу для своих нужд, сохраняя при этом связь между несколькими блокчейн-экосистемами? Можем ли мы позволить пользователям взаимодействовать с любым приложением в любой цепочке блоков прямо из их кошельков?

Чтобы соединить экосистемы блокчейна и позволить приложениям беспрепятственно взаимодействовать между ними, мы предлагаем сеть Axelar. Валидаторы коллективно запускают протокол византийского консенсуса и запускают протоколы, облегчающие межсетевые запросы. Любой может присоединиться к сети, участвовать и использовать ее. Базовая сеть оптимизирована

для высоких требований к безопасности и живучести, уникальных для запросов между цепочками. Сеть Axelar также включает набор протоколов и API. Основные протоколы:

- Протокол межсетевого шлюза (CGP). Этот протокол аналогичен протоколу пограничного шлюза в Интернете. Этот протокол используется для соединения нескольких автономных блокчейн-экосистем и отвечает за маршрутизацию между ними. Блокчейнам не нужно «говорить на каком-то специальном языке», разработчикам их платформ не нужно вносить какие-либо пользовательские изменения в свои цепочки, и их цепочки можно легко подключить к глобальной сети.

- Протокол межсетевой передачи (CTP). Этот протокол аналогичен протоколам прикладного уровня File Transfer, Hypertext Transfer

  Protocols в Интернете. Это стек протоколов уровня приложения, который находится поверх протоколов маршрутизации (таких как CGP и другие технологии маршрутизации). Разработчики приложений могут подключать свои децентрализованные приложения к любой цепочке для выполнения запросов между цепочками. Пользователи могут использовать протокол CTP для взаимодействия с приложениями в любой цепочке, используя простые вызовы API, аналогичные HTTP-запросам GET/POST. Разработчики могут блокировать, разблокировать и передавать активы между любыми двумя адресами на любых платформах блокчейна, запускать триггеры межсетевых приложений (например, децентрализованные приложения в цепочке A, могут обновлять

его состояние, если какое-либо другое приложение в цепочке B удовлетворяет некоторым критериям поиска (процентная ставка > *X*)

ивыполнять общие межсетевые запросы между приложениями в разных цепочках (смарт-контракт в цепочке A может вызывать для обновления состояния смарт-контракта в цепочке B). Этот протокол обеспечивает возможность компоновки программ в блокчейн-экосистемах.

Сеть Axelar предлагает следующие преимущества:

- *Для разработчиков блокчейн-платформ*: возможность легко подключать свои блокчейны ко всем другим блокчейн-экосистемам. Для подключения к сети необходимо настроить только пороговую учетную запись в цепочке.

- *Для разработчиков децентрализованных приложений*: Разработчики приложений могут размещать свои децентрализованные приложения где угодно, блокировать,

разблокировать, передавать активы и взаимодействовать с приложениями в любой другой цепочке через CTP API.

- *Для пользователей*: пользователи могут взаимодействовать со всеми приложениями в экосистеме прямо из своих кошельков.

Платформа для создателей. Наконец, сеть Axelar — это платформа для разработчиков и глобального сообщества. Его модель управления открыта для всех. Разработчики могут предлагать новые точки интеграции, маршрутизацию и протоколы уровня приложений, а пользователи могут решать, принимать ли их, путем голосования по предложениям, и, в случае одобрения, валидаторы примут изменения.

1.1 Существующие решения по функциональной совместимости

Предыдущие попытки решить проблему взаимодействия между блокчейнами относятся к одной из четырех категорий: централизованные биржи, интероперабельные экосистемы, упакованные активы и токен-мосты. Ниже мы кратко суммируем эти подходы.

Централизованные системы. На сегодняшний день централизованные системы являются единственными по-настоящему масштабируемыми решениями для функциональной совместимости.

потребности экосистемы. Они могут относительно легко перечислить любой актив или подключить любую платформу. Тем не мение,

централизованные системы, как известно, имеют различные проблемы с безопасностью и недостаточно хороши для поддержки формирующейся децентрализованной финансовой системы, которая требует надежной безопасности, прозрачности и открытого управления. Сами по себе они не могут управлять децентрализованными приложениями по мере их роста.

Центры взаимодействия. Такие проекты, как Cosmos, Polkadot, Ava labs, направлены на обеспечение взаимодействия между *боковые цепи* родные для их экосистем, использующие настраиваемые протоколы межсетевой связи [23, 25, 24]. Например, можно раскрутить сайдчейн (Cosmos Zone), который может взаимодействовать с Cosmos Hub. Сайдчейн должен быть основан на консенсусе Tendermint и использовать протокол, изначально понятный Cosmos Hub. Подключение к другим блокчейнам и экосистемам, говорящим на разных языках, остается за внешними технологиями.

Парные мосты. Обернутые активы (например, обернутые биткойны) пытаются заполнить недостающий пробел в межсетевой совместимости в экосистеме. Одним из примеров является tBTC [9], который представляет собой специальный протокол, в котором для защиты переводов используется умная комбинация смарт-контрактов и обеспечения. Эти решения требуют значительных инженерных усилий для создания — для каждой пары цепочек разработчики должны создать новый смарт-контракт в цепочке назначения, который анализирует доказательства состояния из исходной цепочки (аналогично тому, как каждая боковая цепочка может, в принципе, анализировать состояние других цепочек). . С использованием этого подхода было развернуто лишь несколько мостов. Эти подходы не масштабируются, когда один из базовых блокчейнов хочет обновить свои правила консенсуса или формат транзакций. Это связано с тем, что все смарт-контракты, которые зависят от состояния этих цепочек, должны быть обновлены. Также необходимо настроить валидаторы и потребовать от них блокировать различные активы, чтобы обеспечить избыточное обеспечение любой передачи активов.

Мы также видели несколько других одноцелевых мостов от разработчиков платформ, которые переписывают логику перехода состояния в смарт-контрактах, чтобы соединяться с другими экосистемами.1, 7]. Они страдают от многочисленных проблем с масштабируемостью, не позволяют экосистеме масштабироваться равномерно и вводят дополнительные зависимости для приложений. Например, при изменении одной платформы необходимо будет обновить все смарт-контракты на всех мостах. Это в конечном итоге поставит экосистему в тупик, где никто не сможет обновиться. Наконец, если один одноцелевой мост соединяет платформы A и B, а второй одноцелевой мост соединяет B и C, это не означает, что приложения на A смогут взаимодействовать с приложениями на C. Возможно, потребуется создать еще один одноцелевой мост. цель моста или перепрограммировать логику приложения.

Другие попытки решить вопрос совместимости включают федеративные оракулы (например, Ren [8]), и взаимодействующие блокчейны для конкретных приложений [10].

Подводя итог, можно сказать, что существующие решения для функциональной совместимости требуют серьезной инженерной работы как от разработчиков платформ, так и от разработчиков приложений, которые должны понимать различные протоколы связи для связи между каждой парой экосистем. Таким образом, интероперабельность практически отсутствует в сегодняшнем пространстве блокчейна. В конце концов, разработчики платформ хотят сосредоточиться на создании платформ и оптимизировать их для своих вариантов использования, а также иметь возможность легко подключаться к другим блокчейнам. И разработчики приложений хотят создавать децентрализованные приложения на лучших платформах для своих нужд, при этом используя пользователей, ликвидность и взаимодействуя с другими децентрализованными приложениями в других цепочках.

2 В поисках масштабируемой межсетевой связи

По сути, кроссчейн-коммуникация требует, чтобы разнородные сети могли общаться на одном языке. Чтобы решить эту проблему, мы объясним набор протоколов Axelar, опишем его высокоуровневые свойства и объясним, как эти свойства относятся к ядру масштабируемой межсетевой связи.

1. *“Интеграция «подключи и работай».* От разработчиков платформы блокчейна не требуется выполнять тяжелую инженерную или интеграционную работу, чтобы говорить на каком-то «пользовательском языке» для поддержки кроссчейн. Кроссчейн-протокол должен иметь возможность беспрепятственно подключать любой существующий или новый блокчейн. Новые

активы должны добавляться с минимальными усилиями.

2. *Кроссчейн маршрутизация.* Такие функции, как обнаружение сетевых адресов, путей маршрутизации и сетей, лежат в основе Интернета и поддерживаются BGP и другими протоколами маршрутизации. Точно так же, чтобы облегчить связь между экосистемами блокчейна, нам необходимо поддерживать обнаружение адресов в них, приложений и маршрутизацию.

3. *Поддержка возможности обновления.* Если одна из экосистем блокчейнов изменится, это не должно повлиять на совместимость других блокчейнов. Система должна распознавать обновления, и для их поддержки должны требоваться минимальные усилия (т. е. не следует переписывать «логику перехода состояний» и приложения не должны ломаться).

4. *Единый язык для приложений.* Приложениям нужен простой протокол для блокировки, разблокировки, передачи и связи с другими приложениями, независимо от того, в какой цепочке они находятся. Этот протокол должен быть независимым от цепочки и поддерживать простые вызовы, аналогичные протоколам HTTP/HTTPS, которые позволяют пользователям и браузерам взаимодействовать с любым веб-сервером. По мере того, как все больше сетей и активов присоединяются к протоколам маршрутизации более низкого уровня, приложения должны иметь возможность использовать их для связи без перезаписи своих программных стеков.

Далее мы суммируем требования безопасности, которым должны соответствовать эти протоколы.

1. *Децентрализованное доверие.* Сеть и протоколы должны быть децентрализованными, открытыми и позволять каждому справедливо участвовать.

2. *Высокая безопасность.* Система должна удовлетворять высоким гарантиям безопасности. Системе необходимо сохранять безопасность активов и состояния по мере их обработки кроссчейн-сетью.

3. *Высокая живучесть.* Система должна удовлетворять высоким гарантиям живучести, чтобы поддерживать приложения, использующие ее кроссчейн-функции.

Удовлетворить подмножество этих свойств легко. Например, можно создать федеративную мультиподписную учетную запись со своими друзьями и заблокировать/разблокировать активы в соответствующих цепочках. Такие системы по своей природе уязвимы для сговора и атак цензуры, и у валидаторов нет надлежащих стимулов для их защиты. Создание децентрализованной сети и набора протоколов, в которых каждый может участвовать при правильном стимулировании, может обеспечить беспрепятственную межсетевую коммуникацию, но решение этой сложной проблемы требует тщательного сочетания протоколов консенсуса, криптографии и проектирования механизмов.

3 Сеть Axelar

Сеть Axelar предоставляет единое решение для кроссчейн-коммуникаций, отвечающее потребностям как разработчиков платформ — от них не требуется никаких работ по интеграции, так и разработчиков приложений — один простой протокол и API для доступа к глобальной ликвидности и связи со всей экосистемой.

Сеть Axelar состоит из децентрализованной сети, которая объединяет блокчейн-экосистемы, говорящие на разных языках, и набора протоколов с API-интерфейсами поверх них, что позволяет приложениям легко выполнять межсетевые запросы. Сеть соединяет существующие автономные блокчейны, такие как Bitcoin, Stellar, Terra, Algorand, и центры взаимодействия, такие как решения, такие как Cosmos, Avalanche, Ethereum и Polkadot. Наша миссия состоит в том, чтобы позволить разработчикам приложений создавать такие приложения проще, используя универсальный протокол и API, не развертывая свои проприетарные межсетевые протоколы или переписывая приложения по мере разработки новых мостов.

Сэтой целью мы разработали набор протоколов, который включает в себя протокол межсетевого шлюза (см. раздел 6) и протокол межсетевой передачи (см. раздел 7).

Основным компонентом сети являются базовые децентрализованные протоколы. Валидаторы коллективно поддерживают сеть Axelar и запускают узлы, которые защищают блокчейн Axelar. Они избираются пользователями в процессе делегирования. Валидаторы получают право голоса пропорционально делегированной им доле. Валидаторы достигают консенсуса в отношении состояния нескольких блокчейнов, к которым подключена платформа. Блокчейн отвечает за поддержку и работу межсетевых протоколов маршрутизации и передачи. Правила управления позволяют участникам сети принимать протокольные решения, например, какие блокчейны соединять и какие активы поддерживать.

Блокчейн Axelar следует модели Delegated Proof-of-Stake (DPoS), аналогичной Cosmos Hub. Пользователи выбирают валидаторов, которые должны связать свою долю, чтобы участвовать в консенсусе и поддерживать высокое качество обслуживания. Модель DPoS позволяет поддерживать большой набор децентрализованных валидаторов и надежные стимулы, чтобы гарантировать, что валидаторы несут ответственность за поддержание мостов и долей схем криптографических порогов. В рамках консенсуса валидаторы запускают легкое клиентское программное обеспечение других блокчейнов, что позволяет им проверять состояние других блокчейнов. Валидаторы сообщают об этих состояниях в блокчейн Axelar, и как только их становится достаточно, состояние биткойнов, эфириума и других цепочек записывается в Axelar.

Впоследствии базовый уровень Axelar знает о состоянии внешних блокчейнов в любой момент времени, создавая «входящие мосты» из других блокчейнов. Валидаторы коллективно поддерживают *учетные записи с пороговой подписью* в других цепочках блоков (например, 80% валидаторов должны одобрять и совместно подписывать любую транзакцию из нее), что позволяет им блокировать и разблокировать активы и состояние в цепочках, а также публиковать состояние в других цепочках блоков, «исходящих мостах». В целом, сеть Axelar можно рассматривать как *децентрализованный перекрестный оракул для чтения/записи*.

В оставшейся части документа описываются предварительные сведения и строительные блоки, лежащие в основе сети (Раздел 4), некоторые технические детали сети (раздел 5), протокол межсетевого шлюза (раздел 6) и протокол межсетевой передачи (раздел 7).

4 подготовительные мероприятия

4.1 Обозначения и предположения

давайте обозначим *Vr* как набор валидаторов Axelar на раунде R. У каждого валидатора есть *вес*, число в (0, 1] обозначая право голоса этого конкретного валидатора. Веса всех валидаторов в сумме дают 1. Валидатор *правильный* если она запускает узел, соответствующий правилам протокола Axelar. Для финализации блоков или для подписи межсетевых запросов Axelar требуются правильные валидаторы общего веса > F . Мы называем параметр $F \in [0.5, 1]$ *порог протокола*.

Axelar может быть основан на *мгновенная окончательность Delegated-Proof-of-Stake* блокчейн. Валидаторы работают *Византийский отказоустойчивый консенсус (BFT)* в каждом раунде *i* завершить блокировать i-й. Как только блок завершен, выполняется новый консенсус BFT для завершения *i* + 1 блокировать и так далее. Валидаторы избираются путем делегирования доли. Пользователь с некоторой долей может выбрать запуск узла валидатора или делегировать свое право голоса (долю) существующему валидатору, который затем голосует от его имени. Набор валидаторов можно обновлять, валидаторы присоединяются к набору или покидают его, а пользователи делегируют или отменяют делегирование своего права голоса.

Различные блокчейны работают с разными сетевыми предположениями. *Синхронная связь* означает, что существует фиксированная верхняя граница Δ времени доставки сообщения, где Δ известно и может быть встроено в протокол. *Асинхронная связь* означает, что доставка сообщений может занять сколь угодно много времени, и известно, что протоколы BFT не могут быть построены для асинхронных сетей даже при наличии только одного злонамеренного валидатора. Реалистичным компромиссом между синхронией и асинхронией является допущение *частично синхронная связь*. Сеть может быть полностью асинхронной до некоторого неизвестного времени глобальной стабилизации (GST), но после GST связь становится синхронной с известной верхней границей Δ [17].

Типичные блокчейны работают в предположении > *F* правильных валидаторов. Для синхронных сетей обычно устанавливается *F = 1/2*, но для более слабого предположения о частично синхронной сети *F = 2/3*. Биткойн, его форки и текущая Proof-of-Work версия Ethereum работают только при условии синхронности. Другие, такие как Algorand и Cosmos, требуют только частичной синхронизации. При соединении цепочек через Axelar соединение работает, предполагая самые сильные сетевые предположения из этих цепочек, что является синхронностью, например, в случае соединения Биткойна и Космоса. Сам блокчейн Axelar работает в частично синхронном режиме и поэтому требует *F = 2/3*, но можно улучшить пороговое требование, предполагая, что другие существующие цепочки блоков безопасны и используя их безопасность.

4.2 Криптографические предварительные сведения

Цифровые подписи. А *схема цифровой подписи* представляет собой набор алгоритмов (*Keygen, Подписать, Подтвердить*). *Кейген* выводит пару ключей (PK, *SK*). Только владелец SK может подписывать сообщения, но любой может проверить подписи с помощью открытого ключа PK. Сегодня большинство блокчейн-систем используют одну из стандартных схем подписи, такую как ECDSA, Ed25519 или несколько их вариантов.2, 3].

Пороговые подписи. А *схема пороговой подписи* дает возможность группе *н* стороны разделить секретный ключ для схемы подписи таким образом, чтобы любое подмножество *t* + 1 или несколько сторон могут сотрудничать для создания подписи, но не подмножество t или меньшее количество сторон может создать подпись или даже узнать какую-либо информацию о секретном ключе. Подписи, создаваемые пороговыми протоколами для ECDSA и EdDSA, выглядят идентично подписям, создаваемым автономными алгоритмами.

Схема пороговой подписи заменяет *Кейген* и *Подписать* алгоритмы обычной схемы подписи с распределенным *н* партийные протоколы *T.Кейген*, *T .Sign*. Для этих протоколов обычно требуется как общедоступный широковещательный канал, так и частные парные каналы между сторонами, и они обычно включают несколько раундов связи. После успешного прохождения*T.Кейген* каждый

пользователь имеет долю si секретного ключа SK и соответствующего открытого ключа PK. То*T .Sign* протокол позволяет этим сторонам произвести подпись для данного сообщение, которое действительно под открытым ключом PK. Эта подпись может быть проверена любым пользователем *Проверяющий* алгоритм оригинальной схемы подписи.

4.3 Свойства пороговых сигнатур

Есть несколько свойств пороговой схемы, которые особенно желательны для децентрализованных сетей:

Защита от нечестного большинства. Некоторые пороговые схемы имеют ограничение, заключающееся в том, что они безопасны.

только тогда, когда большинство n партии честные. Таким образом, пороговый параметр*m* должен быть меньше, чем *n*/2 [15]. Это ограничение обычно сопровождается тем фактом, что 2t + 1 честные стороны нужны для подписания, хотя только *t* + 1 поврежденные стороны могут вступить в сговор для восстановления секретного ключа. Схемы, которые не страдают от этого ограничения, называются*обезопасить себя от нечестного большинства*.

Как обсуждается далее в разделе 5.2, кроссчейн-платформы должны максимально повысить безопасность своих сетей и быть в состоянии терпеть как можно больше коррумпированных сторон. Таким образом, необходимы схемы, которые могут терпеть нечестное большинство.

Предварительная подпись, неинтерактивная онлайн-подпись. Стремление уменьшить нагрузку на общение

Когда стороны подписывают сообщение, в нескольких последних протоколах определена значительная часть работы по подписи, которая может быть выполнена «в автономном режиме», до того, как станет известно сообщение, которое нужно подписать.18, 13]. Результат этой автономной фазы называется*предварительная подпись*. Изготовление предварительных подписей рассматривается как отдельный протокол*T .Presign*в отличие от*T.Кейген* и *T .Sign*. Выходы протокола предварительной подписи должны храниться сторонами в секрете до тех пор, пока они не используют их на этапе подписания. Позже, когда сообщение для подписи станет известно, останется сделать лишь небольшой объем дополнительной «онлайновой» работы в*T .Sign* для завершения подписи.

онлайн*T .Sign* Этап не требует никакого общения между сторонами. Каждая сторона просто выполняет локальные вычисления для сообщения и предварительной подписи, а затем объявляет свою долю.*ся* подписи. (После публикации эти подписи s1, . . . , st+1 делятся легко комбинируются кем угодно, чтобы раскрыть фактическую подпись s) Это свойство называется *неинтерактивная онлайн-подпись*.

Надежность. Пороговые схемы гарантируют только то, что часть злоумышленников не может подписывать сообщения или узнать секретный ключ. Однако эта гарантия не исключает возможности того, что злоумышленники могут заблокировать всех остальных от создания ключей или подписей. В некоторых схемах злонамеренное поведение даже одной стороны может привести к*T.Кейген* или *T .Sign* прервать без полезного вывода. Единственный выход — перезапустить протокол, возможно, с разными сторонами.

Вместо этого для децентрализованных сетей мы хотим *T.Кейген* и *T .Sign* добиться успеха, если хотя бы *t* + Одна из сторон честна, даже если некоторые злоумышленники отправляют сообщения в искаженном формате или удаляют сообщения в протоколах. Это свойство называется*прочность*.

Атрибуция вины. Способность выявлять плохих актеров в *T.Кейген* или *T .Sign* называется *приписывание вины*.

Без атрибуции вины трудно надежно исключить или наказать недобросовестных участников, и в

этом случае расходы, налагаемые на недобросовестных участников, должны нести все. Это свойство также важно для децентрализованных сетей, где злонамеренное поведение должно быть идентифицируемо и экономически дестимулировано с помощью правил сокращения.

Безопасность в параллельных настройках. Схема подписи должна быть безопасной в параллельных условиях, где несколько экземпляров алгоритмов генерации ключей и подписи могут быть задействованы параллельно. (Драйверс

и др. [16] например, показал атаку на мультиподписные схемы Schnorr в этих настройках). Существуют версии как схем ECDSA, так и схем Шнорра, которые удовлетворяют этим свойствам [13, 22].

ECDSA и EdDSA на сегодняшний день являются наиболее широко используемыми схемами подписи в пространстве блокчейна.

Таким образом, пороговые версии обеих схем были в центре внимания недавнего возрождения исследований и разработок.

Читатели, интересующиеся современными технологиями, могут обратиться к [22, 13, 18] и недавний обзорный документ [12]

5 Сеть Axelar

5.1 Проектирование открытой кроссчейн сети

Мосты, поддерживаемые сетью Axelar, поддерживаются пороговыми учетными записями, так что (почти) все валидаторы должны коллективно авторизовать любой межсетевой запрос. Проектирование сети, в которой каждый может участвовать в обеспечении безопасности этих мостов, требует выполнения следующих технических требований:

- *Открытое членство.* Любой пользователь должен иметь возможность стать валидатором (по правилам сети).
- *Обновления членства.* Когда валидатор честно покидает систему, его ключ должен быть соответствующим образом отозван.
- *Поощрения и слэшинг.* Злонамеренные валидаторы должны быть идентифицируемы, а их действия должны быть идентифицированы и рассмотрены протоколом.
- *Консенсус.* Пороговые схемы сами по себе определяются как автономные протоколы. Для распространения сообщений между узлами нам нужны как широковещательные, так и двухточечные частные каналы. Более того, валидаторы должны согласовывать последнее состояние каждого вызова пороговых схем, поскольку они часто имеют несколько раундов взаимодействия.
- *Ключевой менеджмент.* Как обычные валидаторы в любой системе PoS должны тщательно охранять свои ключи, так и валидаторы Axelar должны охранять свои пороговые доли. Ключи нужно вращать, разделять между онлайн и оффлайн частями и т.д.

Axelar начинает с модели Delegated Proof-of-Stake, в которой сообщество выбирает набор валидаторов для достижения консенсуса. Обратите внимание, что стандартные пороговые схемы относятся к каждому игроку одинаково и не имеют понятия «вес» в консенсусе. Следовательно, сеть должна адаптировать их, чтобы учитывать вес валидаторов. Простой подход заключается в назначении нескольких пороговых долей более крупным валидаторам. Ниже описаны три основные функции, которые коллективно выполняют валидаторы.

- *Генерация порогового ключа.* Существующие алгоритмы генерации порогового ключа для

стандартных схем подписи блокчейна (ECDSA, Ed25519) представляют собой интерактивные протоколы между несколькими участниками (см. Раздел 4). Специальная транзакция в сети Axelar дает указание валидаторам начать выполнение этого протокола с отслеживанием состояния. Каждый валидатор запускает процесс порогового демона, который отвечает за безопасное хранение секретного состояния. Для каждой фазы протокола:

1. Валидатор хранит состояние протокола в своей локальной памяти.

2. Он вызывает секретный демон для генерации сообщений в соответствии с описанием протокола для других валидаторов.

3. Он распространяет сообщения либо через широковещательную рассылку, либо через частные каналы другим валидаторам.

4. Каждый валидатор выполняет функции перехода состояния, чтобы обновить свое состояние, перейти к следующему этапу протокола и повторить описанные выше шаги.

В конце протокола в цепочке Axelar генерируется пороговый открытый ключ, и его можно отобразить обратно пользователю (например, для депозитов) или приложению, сгенерировавшему первоначальный запрос.

- *Подписание порога.* Запросы на подпись в сети Axelar обрабатываются аналогично запросам на генерацию ключей. Они вызываются, например, когда пользователь хочет вывести актив из одного из протоколов. Это цепи. интерактивные протоколы, и переход состояния между раундами запускается как
функция сообщений, распространяемых через представление блокчейна Axelar и локальную память каждого валидатора.

- *Обработка изменений членства в Валидаторе.* Набор валидаторов необходимо периодически менять, чтобы новые заинтересованные стороны могли присоединиться к набору. После обновления набора валидаторов нам нужно обновить пороговый ключ, который будет использоваться в новом наборе. Таким образом, если бы мы позволили любому присоединяться в любое время, нам пришлось бы очень часто обновлять пороговый ключ. Чтобы предотвратить это, мы меняем валидаторов каждыеT блоки. В интервалахT патроны, набор B_p и пороговый ключ фиксированы. В каждом раунде, который является целым кратным параметраT , мы обновляем набор валидаторов следующим образом:

1. В любом раунде R, состояние Axelar отслеживает текущий набор валидаторов V^R. $V^{R+1} = V^R$ пока не R + 1 кратно T .

2. Во время раундов ((i − 1)T, iT) пользователи публикуют связывающие/разъединяющие сообщения.

3. В конце раунда iT , эти сообщения применяются к V^{iT-1} *чтобы получить* V^{iT} .

- *Генерация порогового ключа и подпись при наличии меняющихся валидаторов.* Блокчейн Axelar может выдать запрос на новый ключ или пороговую подпись в раунде p. Процесс подписания занимает больше одного раунда, и мы не хотим замедлять достижение консенсуса, поэтому просим произвести подпись до раунда R + 10 пусков. В частности, валидаторы начинают раунд R + 10 только после просмотра сертификата на раунд R + 9 и подпись для каждого запроса кейгена/подписи, выданного на раунде R. Итоги всех раундов R запросы должны быть включены в блок R + 11. Другими словами, раунд R блокирует предложение, не содержащее результатов раунда R - 11 считается недействительным, и валидаторы по нему не голосуют. Чтобы убедиться, что все пороговые сообщения подписаны до обновления набора валидаторов, Axelar не выдает никаких пороговых запросов в течение раунда, равного -1,-2, . . . ,-9 по модулю T .

5.2 Безопасность сети

Безопасность систем блокчейна зависит от различных криптографических и игровых протоколов, а также от децентрализации сети. Например, в блокчейнах с доказательством доли без надлежащих стимулов валидаторы могут вступить в сговор и переписать историю, похищая в процессе средства других пользователей. В сетях с доказательством работы без достаточной децентрализации довольно легко создавать длинные форки и двойные траты, как доказали многочисленные атаки на Bitcoin Gold и Ethereum Classic.

Большая часть исследований безопасности блокчейна была сосредоточена на суверенных цепях. Но как только цепочки взаимодействуют, необходимо учитывать новые векторы атак. Например, предположим, что Ethereum взаимодействует с небольшой цепочкой блоков X через прямой мост, контролируемый двумя смарт-контрактами, одним на Ethereum и одним на X. Помимо инженерных проблем, которые мы суммировали в разделе 1.1, нужно решить, что происходит, когда предположения о доверии X нарушаются. В этом случае, если ETH переместился в X, валидаторы X могут вступить в сговор, чтобы подделать историю X, в которой они держат все ETH, опубликовать поддельные доказательства консенсуса в Ethereum и украсть ETH. Ситуация еще хуже, когда X соединен с несколькими другими цепями через прямые мосты, где, если X разветвляется, эффекты распространяются через все мосты. Настройка руководящих принципов управления восстановлением для каждого парного моста — непосильная задача для любого отдельного проекта.

Сеть Axelar решает проблемы безопасности, используя следующие механизмы:

- *Максимальная безопасность.* Axelar устанавливает порог безопасности на уровне 90%, а это означает, что почти все валидаторы должны будут вступить в сговор, чтобы снять любые средства, которые заблокированы его сетью, или подделать доказательства состояния.[1]. На практике было замечено, что валидаторы PoS имеют очень большое время безотказной работы (около 100%), если они должным образом мотивированы. Следовательно, сеть Axelar будет производить блоки даже несмотря на этот высокий порог. Однако в редких случаях, когда что-то пойдет не так и сеть остановится, ей нужны надежные резервные механизмы для перезагрузки системы, описанной ниже.

- *Максимальная децентрализация.* Поскольку в сети используются пороговые схемы подписи, количество валидаторов может быть максимально большим. Сеть не ограничена количеством валидаторов, которые мы можем поддерживать, лимитами транзакций или сборами, которые могут возникнуть, например, при использовании мультиподписей в разных цепочках, где сложность (и сборы) увеличиваются линейно с количеством валидаторов.[2]

- *Надежные резервные механизмы.* Первый вопрос, который необходимо решить в сети с высокими порогами безопасности, как указано выше, — это то, что происходит, когда сама сеть останавливается. Допустим, сама сеть Axelar зависает. Можем ли мы иметь запасной механизм, который позволил бы пользователям восстановить свои средства? Чтобы устранить любую потенциальную остановку самой сети Axelar, каждая учетная запись порогового моста в блокчейне X, которую коллективно контролируют валидаторы Axelar, имеет «ключ аварийной разблокировки». Этим ключом можно поделиться

*1 Окончательный параметр, который будет выбран для развертывания сети, может быть скорректирован.

*2 Для некоторых блокчейнов мультиподписи предлагают разумную альтернативу, когда количество газа невелико, а поддерживаемые форматы сообщений подходят. Но они не подходят для двух самых крупных платформ, таких как Биткойн и Эфириум.

между тысячами сторон и может даже быть настраиваемым ключом для блокчейна X, который используется в сообществе этой цепочки. Следовательно, если сеть Axelar остановится, этот ключ будет действовать как запасной вариант и позволит восстановить активы (подробнее см. Ниже).

- *Максимальная децентрализация резервных механизмов.* Этот резервный механизм включает вторичный *набор для восстановления* пользователей, в котором каждый может участвовать без каких-либо затрат. Этим пользователям не нужно быть в сети, запускать узлы или координировать свои действия друг с другом. Их «дежурят» только в том случае, если сеть Axelar зависает и не может восстановиться. Безопасность сети повышается за счет очень высокого порога для основного набора валидаторов и максимально децентрализованного вторичного набора для восстановления.

- *Совместное управление.* Общий протокол управляет сетью Axelar. В совокупности пользователи могут голосовать за то, какая цепочка должна поддерживаться через ее сеть. Сеть также будет выделять пул средств, которые можно использовать для возмещения пользователям расходов в случае непредвиденных чрезвычайных ситуаций, которые также контролируются с помощью протоколов управления.

Ниже обсуждаются различные механизмы безопасности.

Механизмы возврата. Когда Axelar останавливается из-за высокого порога, «ключ аварийной разблокировки» берет на себя управление сетью. Существует несколько способов создания экземпляра этого ключа разблокировки, и некоторые цепочки/ приложения могут использовать другой вариант для «набора восстановления» или полностью отказаться:[3]

- *Вариант а.* Поделитесь ключом между фондами блокчейн-проектов и уважаемыми людьми в сообществе.

- *Вариант б.* Разделите выбор между сторонами, избранными через делегированный механизм PoS.

- *Вариант в.* Для учетных записей, управляющих активами и информацией для цепочки/приложения X, предоставьте общий

  ключ заинтересованным сторонам/валидаторам X. Предполагая, что у X есть механизмы управления, те же механизмы управления могут применяться для определения курса действий, если Axelar остановится.

Теперь, учитывая идентификаторы пользователей для восстановления и их открытые ключи, простой протокол создает общий доступ к ключу восстановления, о котором никто не знает. Кроме того, пользователям набора для восстановления не нужно находиться в сети до тех пор, пока не будут вызваны для восстановления через механизмы управления. Следуя стандартным распределенным протоколам генерации ключей, каждый валидатор Axelar использует случайное значение. Секретный ключ восстановления генерируется путем суммирования этих значений. Вместо того, чтобы выполнять суммирование в открытом виде, все общие ресурсы шифруются открытыми ключами пользователей восстановления, а затем гомоморфно складываются (это предполагает аддитивное гомоморфное шифрование и дополнительный уровень нулевого разглашения, оба из которых легко доступны). Результатом этого протокола является открытый ключ восстановления. *RPK* и потенциально тысячи шифров (под открытыми ключами пользователей восстановления) долей соответствующего секретного ключа *Enci(si)*, которые распространяются среди их владельцев (например, размещаются в цепочке). Бридж-контракты Axelar включают возможность возмещения средств с использованием *RPK* при определенных условиях. Наконец, также можно обновить этот ключ восстановления и даже изменить набор пользователей, владеющих его акциями, без каких-либо действий со стороны участвующих акционеров.

Если цепочка X, связанная с Axelar, разорвется, есть пара вариантов:

- Установите ограничения на стоимость активов в долларах США, которые могут быть перемещены в/из X в любой день. Таким образом, вредоносная цепочка X может украсть лишь небольшую часть всех активов, которые связаны с ней, прежде чем валидаторы Axelar обнаружат это и сработают механизмы управления из следующих пуль.

- Модуль управления Axelar можно использовать для голосования по поводу того, что происходит в таких ситуациях. Например, если возникает несерьезная ошибка и сообщество перезапускает X, управление Axelar может принять решение о перезапуске соединения с того места, где оно было прервано.

- Если ETH переместился в X, пользовательский ключ восстановления Ethereum может определить, что произойдет с активами

  ETH.

*3 Окончательное развертывание в сети Axelar будет завершено ближе к запуску сети.

6 Протокол межсетевого шлюза (CGP)

В этом разделе мы объясняем протокол межсетевого шлюза и механизмы маршрутизации на двух основных примерах, общих для многих приложений:

Синхронизация состояний (раздел 6.2). Размещать информацию о состоянии исходного блокчейна S в состояние целевого блокчейна *D*.
(Например, опубликуйте заголовок блока биткойнов в блокчейне Ethereum.)

Передача активов (Раздел 6.3). Перенести цифровой актив из S к D и обратно.
(Например, перенесите биткойны из блокчейна Биткойн в блокчейн Эфириума, а затем обратно в блокчейн Биткойн.)

Для простоты будем считать, что цепочка D имеет хотя бы минимальную поддержку смарт-контрактов, но S может быть любой блокчейн.

6.1 Аккаунты в других сетях

Чтобы соединить разные цепочки, в каждой цепочке создаются пороговые учетные записи, которые контролируют поток ценности и информации через них. Для цепи *Chain*, обозначим счет через $Chain_{Axelar}$.

Биткойн-счет. Для биткойнов и других цепочек смарт-контрактов валидаторы Axelar создают пороговый ключ ECDSA в соответствии с разделом 5.1. Этот ключ управляет учетной записью ECDSA в биткойнах и является адресом назначения, куда пользователи отправляют депозиты. Персонализированные пороговые ключи могут быть созданы по запросу пользователя. Ключ может периодически обновляться, а последний ключ и персонализированные ключи можно найти, запросив узел Axelar.

Аккаунт порогового моста в цепочках со смарт-контрактами. Обозначим цепь через SC. валидаторы создают пороговый ключ ECDSA или ED25519 согласно разделу 5.1, в зависимости от того, какой тип ключа поддерживает цепочка. Обозначим этот ключ через PK_{Axelar}, когда нет никакой двусмысленности относительно того, о какой цепочке идет речь. Этот ключ управляет учетной записью смарт-контракта на SC, обозначаемой SC_{Axelar}, и любое приложение на SC может запрашивать SK_{Axelar} чтобы узнать PK-адрес этого ключа. Таким образом, любое приложение SC может распознавать сообщения, подписанные SK_{Axelar}. Протокол также должен учитывать ротацию значений PK_{Axelar}. Это происходит следующим образом:

1. Инициализировать SK_{Axelar} на SK. Он хранит PK_{Axelar} как часть своего состояния, которое инициализируется как значение генезиса на Axelar. SK_{Axelar} также включает правила обновления PK.

2. Обновить PK_{Axelar}, транзакция формата (*обновление,* PK_{new}) должен быть представлен с подписью действующего SK_{Axelar}. Затем устанавливается договор $PK_{Axelar} = PK_{new}$.

3. Каждый раз, когда валидаторы обновляют пороговый ключ для SC с PK^i к PK^{i+1}, Axelar запрашивает, чтобы валидаторы использовали SK^i подписи (*обновление,* PK^{i+1}). Впоследствии эта подпись размещается на SC_{Axelar}. Которые обновляют PK_{Axelar}.

6.2 Синхронизация состояний

Давайте qs обозначим как произвольный вопрос о состоянии цепочки S. Примеры таких вопросов включают:

- *"В каком раунде блока, если любой, появилась транзакция tx?»*
- *"Каково значение определенного поля данных?»*
- *"Что такое корневой хэш Меркла входа состояния S в блок-раунде 314159?*

Давайте обозначим as как правильный ответ на qs и предположим, что конечный пользователь или приложение требуют, чтобы as быть размещенным в цепочке D. Сеть Axelar отвечает этому требованию следующим образом:

1. Пользователь отправляет запрос qs на одном из бридж-аккаунтов (которые впоследствии подхватываются валидаторами) или напрямую в блокчейн Axelar.

2. В рамках консенсуса Axelar каждый валидатор должен запускать программное обеспечение узла для цепочек. S, D. Валидаторы Axelar запрашивают API своей цепочки S программное обеспечение узла для ответа as и сообщить ответ в цепочку Axelar.

3. Один раз > *F* взвешенные валидаторы сообщают об одном и том же ответе в раунде R, Axelar просит валидаторов подписать as.

4. Используя пороговую криптографию, валидаторы подписывают as. Подпись включена в блок R + 11.

5. Любой может взять значение со знаком as из блока R + 11 и опубликовать в *D*.

6. Заявка обслужена. Любое приложение на D теперь может принимать значение со знаком as, запрос D_{Axelar} для последних PK_{Axelar}, и убедитесь, что подпись as соответствует PK_{Axelar}. Валидаторы также публикуют as на бридж аккаунт в цепочке D, которые приложения могут получить.

6.3 Межсетевая передача активов

Сеть позволяет передавать цифровые активы между цепочками, расширяя рабочий процесс синхронизации состояний раздела. 6.2.

Достаточный запас pegged-S токенов печатаются и контролируются D_{Axelar} при его инициализации. Предположим, пользователь требует обмена x количество токенов в исходной цепочке S за x количество pegged-S токенов в цепочке назначения D, подлежащий депонированию в D-адрес wd по выбору пользователя. Мы представляем полностью общий рабочий процесс, который поддерживает произвольные исходные цепочки S - даже такие сети, как Биткойн, которые не поддерживают смарт-контракты:

1. Пользователь (или приложение, действующее от имени пользователя) отправляет запрос на перенос (x, w_d) на учетную запись порогового моста, которая впоследствии направляется в сеть Axelar.

2. Валидаторы Axelar используют пороговую криптографию для коллективного создания нового адреса депозита. d_s за S. Они публикуют d_s к блокчейну Axelar.

3. Пользователь (или приложение, действующее от имени пользователя) узнает d_s путем мониторинга блокчейна Axelar. Пользователь отправляет x количество S-токенов на адрес d_s через обычный S-транзакцию tx_s используя ее любимое программное обеспечение для цепочки *S*.
   (Из-за порогового свойства d_s, токены нельзя потратить из d_s если пороговое количество валидаторов не согласовано это сделать.)

4. tx_s размещен на Axelar. Валидаторы запрашивают API своей цепочки S программное обеспечение узла для существования tx_s и, если ответ «верный», сообщает ответ в цепочку Axelar.

5. Как только > F взвешенные валидаторы сообщают «true» для tx_s в раунде R, Axelar просит валидаторов подписать транзакцию a_d который посылает x количество pegged-S токенов от D_{Axelar} к w_d.

6. Используя пороговую криптографию, валидаторы подписывают a_d. Подпись включена в блок R + 11.

7. Любой может взять значение со знаком a_d из блока R + 11 и опубликовать *D*.

8. Запрос был обслужен, как только a_d размещен на D передача обрабатывается.

Теперь предположим, что пользователь требует выкупить *x'* количество обернутых-S токенов из цепочки *D* вернуться к цепочке *S*, подлежащий депонированию в S-адрес w_s по выбору пользователя. Рабочий процесс выглядит следующим образом:

1. Пользователь инициирует запрос на передачу (*x'*, w_s) путем внесения *x'* количество wrapped-S токенов в Cd через обычный D-транзакцию с использованием ее любимого программного обеспечения для сети *D*

2. (*x'*, w_s) опубликовано на Axelar. Валидаторы запрашивают API своего узла ноды цепи D о существовании (*x'*, w_s) и, если ответ «верный», сообщается ответ в цепочку Axelar.

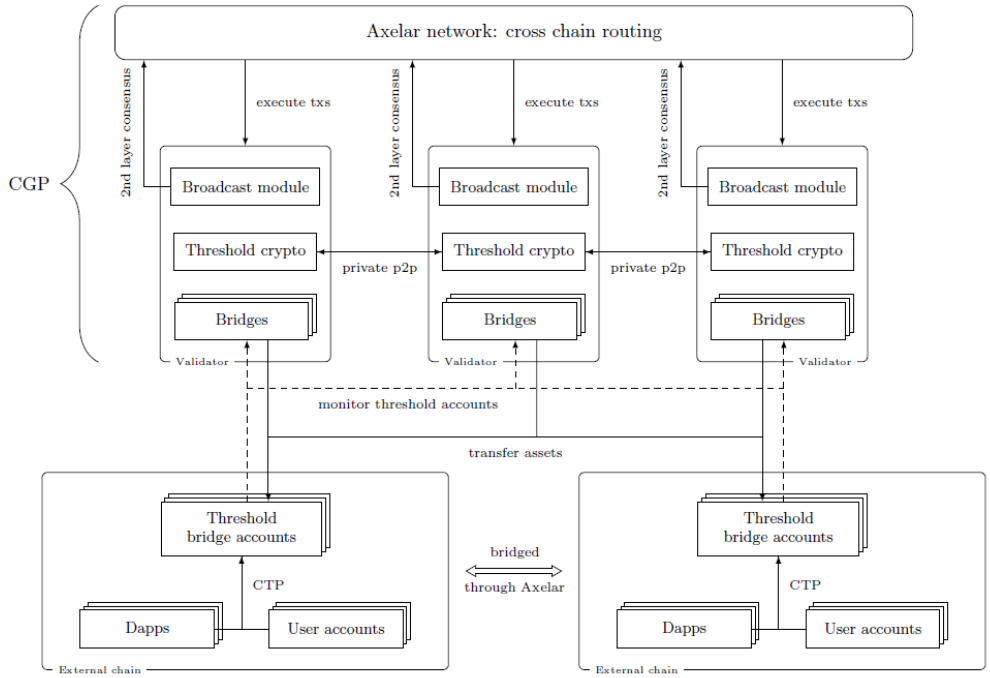

Рисунок 1: Схема компонентов

3. Как только > F взвешенные валидаторы сообщают «true» для (x', w_s) в раунде R, Axelar просит валидаторов подписать транзакцию a_s который посылает x' количество S токенов от S_{Axelar} к w_s.

4. Используя пороговую криптографию, валидаторы подписывают a_s. Подпись включена в блок R + 11.

5. Любой может взять значение со знаком a_s из блока R + 11 и опубликовать *S*.

6. Запрос был обслужен, как только a_s размещен на S передача обрабатывается.

Дополнительные запросы, поддерживаемые уровнем маршрутизации CGP, включают блокировку, разблокировку или передачу активов по цепочке.

Достижение атомарного потока транзакций между цепочками. В зависимости от типа межсетевого запроса Axelar пытается обеспечить выполнение соответствующих транзакций в нескольких цепочках или ни в одной. Для этого каждый запрос может находиться в одном из следующих состояний в блокчейне Axelar:*(инициализировано, ожидается, завершено, истекло время ожидания)*. Если *тайм-аут* на стадии pending срабатывает, запрос возвращает код ошибки. Некоторые события тайм-аута также начинают *возвращать деньги* событие: например, если актив из одной цепочки необходимо перевести в актив в другой цепочке, если принимающая цепочка не обработала транзакцию, актив возвращается исходному пользователю.

7 Протокол межсетевой передачи (CTP)

CTP — это протокол уровня приложения, который позволяет приложениям легко использовать кроссчейн-функции. Мы объясняем интеграцию, сосредоточив внимание на функциях передачи активов (например, используемых в DeFi). Эти приложения обычно состоят из трех основных компонентов: графического интерфейса пользователя, смарт-контрактов в одной цепочке и промежуточного узла, который публикует транзакции между интерфейсом и смарт-контрактами. Внешние интерфейсы взаимодействуют с кошельками пользователей, чтобы принимать депозиты, обрабатывать снятие средств и т. д. Приложения могут использовать кроссчейн-функции.
путем вызова CTP-запросов, аналогичных методам HTTP/HTTPS GET/POST. Эти запросы впоследствии принимаются уровнем CGP для выполнения, а результаты возвращаются пользователям.

- *CTP-запросы*. Разработчики приложений могут размещать свои приложения в любой цепочке и интегрировать свои смарт-контракты с учетными записями порогового моста для выполнения запросов CTP.

- *Пороговые промежуточные счета*. Предположим, разработчик приложения строит свои контракты в цепочке A. Затем он будет ссылаться на пороговые мостовые контракты, чтобы получить поддержку между цепочками. Этот контракт позволяет приложениям:

  – Зарегистрируйте блокчейн, с которым он хотел бы общаться.

  – Зарегистрируйте активы в этой цепочке блоков, которые он хотел бы использовать.

  – Выполнять операции над активами, такие как прием депозитов, обработка снятия средств и другие функции (аналогично, скажем, вызовам контрактов ERC-20).

Предположим, известное приложение DeFi, MapleSwap, которое изначально находится в цепочке A, регистрируется с пороговой учетной записью моста. Валидаторы Axelar коллективно управляют самим контрактом в соответствующей цепочке. Предположим, пользователь хочет внести депозит в торговую пару между активами X и Y, которые находятся в двух цепочках соответственно. Затем, когда пользователь отправляет такой запрос, он направляется через учетную запись порогового моста в сеть Axelar для обработки. Форма там, выполняются следующие шаги:

1. Сеть Axelar понимает, что это приложение зарегистрировано для межсетевой поддержки активов. Он генерирует ключ депозита, используя пороговую криптографию и консенсус для пользователя в соответствующих цепочках A и B.

2. Связанные открытые ключи возвращаются в приложение и отображаются для пользователя, который может использовать свои любимые

   кошельки для отправки депозитов. Соответствующий секретный ключ является общим для всех валидаторов Axelar.

3. Когда депозиты подтверждены, Axelar обновляет свой межсетевой каталог, чтобы записать, что пользователь в соответствующих цепочках депонировал эти активы.

4. Валидаторы Axelar выполняют многосторонние протоколы для создания пороговой подписи, которая позволяет обновлять учетную запись порогового моста в цепочке A, где находится приложение.

5. Затем запрос CTP возвращается в смарт-контракты приложения DeFi, которые могут обновлять свое состояние, обновлять формулы доходности, обменные курсы или выполнять другие условия, связанные с состоянием приложения.

На протяжении всего этого процесса сеть Axelar на высоком уровне действует как децентрализованный оракул для чтения/записи между цепочками, CGP — это уровень маршрутизации между цепочками, а CTP — протокол приложения.

Дополнительные межсетевые запросы. CTP поддерживает более общую кросс-цепочку между приложениями в блокчейнах, например:

- Выполнение служб имен открытых ключей (PKNS). Это универсальный каталог для сопоставления открытых ключей с телефонными номерами/дескрипторами Twitter (несколько проектов, таких как Celo, предоставляют эти функции на своих платформах).

- Триггеры кроссчейн-приложений. Приложение в цепочке A может обновить свое состояние, если какое-то другое приложение в цепочке B удовлетворяет критериям поиска (процентная ставка< *Икс*).

- Компонуемость смарт-контрактов. Смарт-контракт в цепочке A может обновлять свое состояние в зависимости от состояния контрактов в цепочке B или инициировать действие для обновления смарт-контракта в цепочке B.

На высоком уровне эти запросы могут быть обработаны, поскольку в совокупности протоколы CTP, CGP и сеть Axelar могут передавать и записывать произвольную поддающуюся проверке информацию о состоянии через блокчейны.

8 Заключение

Втечение последующих лет важные приложения и активы будут созданы на основе нескольких блокчейн-экосистем. Сеть Axelar можно использовать для включения этих блокчейнов в единый уровень межсетевой связи. Этот уровень обеспечивают протоколы маршрутизации и уровня приложений, отвечающие требованиям как разработчиков платформ, так и разработчиков приложений. Разработчики приложений могут использовать лучшие платформы для своих нужд и использовать простой протокол и API для доступа к глобальной кросс-чейн ликвидности, пользователей и связи с другими цепочками.

использованная литература

[1] Алтея Пегги. https://github.com/cosmos/пегги. [Процитировано на странице 2.]

[2] Детерминированное использование алгоритма цифровой подписи (dsa) и алгоритма цифровой подписи на основе эллиптических кривых (ecdsa). https://tools.ietf.org/html/rfc6979. [Процитировано на странице 5.]

[3] Алгоритм цифровой подписи на основе кривой Эдвардса (eddsa). https://tools.ietf.org/html/rfc8032. [Цитируется

страница 5.]

[4] Технический документ Eos.io v2. https://github.com/EOSIO/Documentation/blob/master/TechnicalWhitePaper.md. [Цитируется на стр. 1.]

[5] Ethereum: безопасная децентрализованная обобщенная книга транзакций. https:// ethereum.github.io/yellowpaper/paper.pdf. [Процитировано на странице 1.]

[6] Ближайший официальный документ. https://near.org/papers/the-official-near-white-paper/. [Процитировано на странице 1.]

[7] Радужный мост. https://github.com/near/радужный мост. [Процитировано на странице 2.]

[8] Рен: виртуальная машина, сохраняющая конфиденциальность, на которой работают финансовые приложения с нулевым разглашением

https:

информации. //whitepaper.io/document/419/ren-litepaper. [Процитировано на странице 3.]

[9] tbtc: децентрализованный погашаемый токен erc-20, поддерживаемый btc. https://docs.keep.network/tbtc/index.

пдф. [Процитировано на странице 2.]

[10] Thorchain: децентрализованная сеть ликвидности. https://thorchain.org/. [Процитировано на странице 3.]

[11] Курт М. Алонсо. Ноль к монеро.https://www.getmonero.org/library/Zero-to-Monero-1-0-0.

пдф. [Процитировано на странице 1.]

[12] Жан-Филипп Омассон, Адриан Амелинк и Омер Шломовиц. Обзор пороговой подписи ecdsa. Cryptology ePrint Archive, отчет 2020/1390, 2020.https://eprint.iacr.org/2020/1390. [Цитируется страница 6.]

[13] Ран Канетти, Николаос Макрияннис и Уди Пелед. Uc неинтерактивный, проактивный, пороговый

Cryptology ePrint Archive, отчет 2020/492, 2020 г. ecdsa. https://eprint.iacr.org/2020/492. [Цитируется страница 6.]

[14] Технические документы cLabs. https://celo.org/papers. [Процитировано на странице 1.]

[15] Иван Дамгард, Томас Пелле Якобсен, Йеспер Буус Нильсен, Якоб Иллеборг Пагтер и Михаэль Бэксванг Остергорд. Быстрый порог ECDSA с честным большинством. В*SCN*, том 12238 из *Конспект лекций по информатике*, страницы 382–400. Спрингер, 2020.[Процитировано на странице 6.]

[16] Ману Дрейверс, Касра Эдалатнеджад, Брайан Форд, Эйке Кильц, Джулиан Лосс, Грегори Невен и Игорь Степанов. О безопасности двухраундовой мультиподписи. В*Симпозиум IEEE по безопасности и конфиденциальности*, страницы

1084–1101. ИИЭР, 2019.[Процитировано на странице 6.]

[17] Синтия Дворк, Нэнси Линч и Ларри Стокмайер. Консенсус при наличии частичной синхронности. https:// groups.csail.mit.edu/tds/papers/Lynch/jacm88.pdf. [Процитировано на странице 5.]

[18] Росарио Дженнаро и Стивен Голдфедер. Один раунд порогового ecdsa с идентифицируемым

прерыванием. Cryptology ePrint Archive, отчет 2020/540, 2020 г.https://eprint.iacr.org/2020/540. [Процитировано на странице 6.]

[19] Йосси Гилад, Ротем Хемо, Сильвио Микали, Георгиос Влахос и Николай Зельдович. Algorand: Масштабирование византийских соглашений для криптовалют. Материалы 26-го симпозиума по принципам операционных систем, 2017 г.https://dl.acm.org/doi/pdf/10.1145/3132747.3132757. [Процитировано на странице 1.]

[20] Эван Керейакс, До Квон, Марко Ди Маджио и Николас Платиас. Деньги Terra: стабильность и принятие.

https://terra.money/Terra_White_paper.pdf. [Процитировано на странице 1.]

[21] Ангелос Киайяс, Александр Рассел, Бернардо Давид и Роман Олейников. Ouroboros: доказуемо безопасный блокчейн-протокол с доказательством доли.https://eprint.iacr.org/2016/889.pdf. [Процитировано на странице 1.]

[22] Челси Комло и Ян Голдберг. Frost: гибкие сигнатуры порога Шнорра, оптимизированные для раундов. Cryptology ePrint Archive, отчет 2020/852, 2020 г.https://eprint.iacr.org/2020/852. [Процитировано на странице 6.]

[23] Джэ Квон и Итан Бухман. Космос: сеть распределенных реестров. https://cosmos.network/resources/whitepaper. [Цитируется на страницах 1 и 2.]

[24] Лавинная команда. Лавинная платформа. https://www.avalabs.org/whitepapers. [Цитируется на страницах 1 и 2.]

[25] Гэвин Вуд. Polkadot: Видение гетерогенной многоцепочечной структуры. https://polkadot.network/PolkaDotPaper.pdf. [Цитируется на страницах 1 и 2.]