

کاربردهای بلاک چین

اکنون که بابت کوین آشنا شدیم، اجازه دهید بلاک چین را به عنوان یک بستر برنامه‌ی کاربردی (application platform) مورد بررسی قرار دهیم. امروزه وقتی از «بلاک چین» حرف می‌زنیم، منظورمان بستری برای تولید برنامه‌های کاربردی است که در اصول طراحی بابت کوین اشتراک دارند. البته اغلب از این واژه به اشتباه برای ارجاع به چیزهای زیادی استفاده می‌شود که هیچ ارتباطی با ویژگی‌های اصلی بلاک چین بیت کوین ندارند.

در این فصل ابتدا ویژگی‌های بلاک چین بیت کوین را به عنوان یک بستر برنامه‌ی کاربردی بررسی خواهیم کرد، و سپس به تشریح عملکردهای پایه در تولید برنامه‌های کاربردی که عناصر ساختمانی هر برنامه‌ی بلاک چین هستند، می‌پردازیم. پس از آن چند برنامه‌ی کاربردی مهم که از این عملکردهای پایه استفاده می‌کنند، مانند سکه‌ی رنگی، کانال پرداخت، و کانال پرداخت هدایت‌شده (شبکه‌ی آذرخش)، معرفی خواهیم کرد.

مقدمه

بیت کوین به عنوان یک سیستم ارز و پرداخت غیر متمرکز طراحی شد. با این حال، بخش اعظم کارکرد آن از سازه‌های سطح-پایینی مشتق شده است که می‌توانند کاربردهای بسیار گسترده‌تری داشته باشند. در واقع، بیت کوین بر اساس اجزایی مانند حساب، کاربر، تراز و پرداخت ساخته نشده است؛ به جای آن، همان طور که در فصل ۶ دیدیم، بیت کوین از یک زبان اسکرپت نویسی تراکنش-محور با کارکردهای رمزنگاری سطح-پایین استفاده می‌کند. درست همان گونه که مفاهیم سطح-بالا مانند حساب، تراز و پرداخت را می‌توان از این عملکردهای پایه استخراج کرد، آنها کاربردهای پیچیده‌ی متعدد دیگری نیز می‌توانند داشته باشند. بنابراین، بلاک چین بیت کوین می‌تواند تبدیل به یک بستر برنامه‌ی کاربردی شود که سرویس اعتماد در اختیار برنامه‌های کاربردی (مانند قراردادهای هوشمند) می‌گذارد، و از هدف اولیه‌ی خود (پرداخت و ارز دیجیتال) بسیار فراتر می‌رود.

عناصر ساختمانی (عملکردهای پایه)

اگر سیستم بیت کوین به درستی و برای مدتی طولانی کار کند، ضمانت‌های مشخصی ارائه می‌کند که می‌توانند به عنوان عناصر ساختمانی برای ساخت برنامه‌های کاربردی به کار برده شوند. این عناصر ساختمانی عبارتند از:

عدم امکان خرج-دوباره

اساسی ترین ضمانتی که الگوریتم اجماع غیر متمرکز بیت کوین ارائه می کند، این است که یک UTXO را نمی توان دو بار خرج کرد.

برگشت ناپذیری

همین که یک تراکنش در بلاک چین ثبت شده و با استخراج بلاک های بعدی مقدار کار کافی به آن اضافه شود، داده ی این تراکنش برگشت ناپذیر خواهد بود. برگشت ناپذیری ضمانت خود را از انرژی مصرف شده کسب می کند، چون بازنویسی بلاک چین مستلزم صرف انرژی [بسیار زیاد] برای تولید اثبات-کار جدید است. هر چه تعداد بلاک های قرار گرفته روی بلاک حاوی یک تراکنش بیشتر شود، انرژی لازم برای بازنویسی آن بیشتر شده، و در نتیجه برگشت ناپذیری آن افزایش خواهد یافت.

بی طرفی

شبکه ی غیر متمرکز بیت کوین تراکنش های معتبر را صرف نظر از مبدأ یا محتویات آنها منتشر می کند. این بدان معنا است که هر کسی می تواند در هر زمانی یک تراکنش معتبر با کارمزد کافی تولید کند و مطمئن باشد که تراکنش او در شبکه منتشر شده و در بلاک چین ثبت خواهد شد.

امنیت برچسب زمانی

قواعد اجماع بیت کوین هر بلاکی را که برچسب زمانی آن در آینده یا گذشته ای دور باشد، رد می کنند. به عبارت دیگر، برچسب زمانی بلاک های تأیید شده کاملاً قابل اعتماد است. برچسب زمانی یک بلاک تضمین می کند که ورودی تمام تراکنش های موجود در آن قبل از این تاریخ-زمان خرج نشده اند.

بررسی مجوز

امضاهای دیجیتال که در یک شبکه ی غیر متمرکز اعتبارسنجی شده اند، مجاز بودن محتویات یک تراکنش را تضمین می کنند. یک اسکریپت حاوی امضای دیجیتال را نمی توان بدون کلید خصوصی دارنده ی آن (که در این اسکریپت اشاره شده) اجرا کرد.

امکان حسابرسی

تمامی تراکنش ها عمومی هستند و می توان آنها را حسابرسی کرد. تمام تراکنش ها و بلاک ها می توانند در یک زنجیره ی ناگسسته تا بلاک زاینده ردگیری شوند.

تراز پذیری

در هر تراکنش (به استثنای تراکنش پایگاه سکه)، مقدار ورودی ها باید با مجموع خروجی ها و کارمزد تراکنش برابر باشد. به بیان دیگر، نمی توان در یک تراکنش مقداری بیت کوین خلق یا نابود کرد. مجموع خروجی های یک تراکنش نمی تواند از ورودی ها بیشتر باشد.

انقضای ناپذیری

یک تراکنش معتبر هرگز منقضی نمی شود. اگر یک تراکنش امروز معتبر باشد، در آینده ی نزدیک یعنی مادامی که خروجی های آن خرج نشده و قواعد اجماع تغییر نکرده باشند، معتبر خواهد بود.

انجام
اگر یک تراکنش بیت کوین با SIGHASH_ALL امضا شده باشد، یا بخش‌هایی از آن با هر نوعی از پرچم SIGHASH امضا شده باشند، بدون نامعتبر کردن این امضا (و در نتیجه نامعتبر کردن کل تراکنش) نمی‌توان آن را تغییر داد.

یکپارچگی تراکنش
تراکنش‌های بیت کوین یکپارچه هستند: یک تراکنش یا معتبر (قابل استخراج) است یا نیست. به عبارت دیگر، نمی‌توان یک تراکنش را به اجزای آن تقسیم کرده و هر بخش را جداگانه اعتبارسنجی یا استخراج کرد. در هر لحظه از زمان، یک تراکنش یا استخراج می‌شود یا نمی‌شود.

مقادیر گسته (تقسیم‌ناپذیر)
خروجی‌های یک تراکنش گسته و مجزا هستند و مقدار آنها را نمی‌توان تقسیم کرد. یک خروجی را یا باید به طور کامل خرج کرد، یا اصلاً خرج نکرد.

حد نصاب
وقتی محدودیت‌های چندامضایی بر یک اسکریپت تحمیل شوند، تأیید آن وابسته به تأمین حد نصاب «M-از-N» خواهد بود.

قفل زمانی / پیر شدن
هر بخش از یک اسکریپت که دارای قفل زمانی نسبی یا مطلق باشد، فقط بعد از گذشت زمان مشخص شده (در سررسید) می‌تواند اجرا شود.

تکثیر
غیرمتمرکز بودن بلاک چین تضمین می‌کند که بعد از استخراج یک تراکنش و کسب تأییدیه‌های کافی، آن تراکنش در کل شبکه تکثیر شده و ماندگار خواهد بود.

حفاظت در مقابل جعل
یک تراکنش فقط خروجی‌های موجود و اعتبارسنجی شده را می‌تواند خرج کند. خلق خروجی موهوم یا جعل آنها غیرممکن است.

ثبات
در غیاب تقسیم شبکه بین معدنچیان نامتوافق، احتمال نامعتبر شدن یا تغییر سازماندهی بلاک‌هایی که در بلاک چین ثبت شده‌اند، متناسب با افزایش عمق آنها به صورت نمایی کاهش خواهد یافت. همین که یک بلاک به عمق کافی برسد، میزان پردازش و انرژی لازم برای تغییر آن به گونه‌ای بالا می‌رود که هر گونه تغییر را عملاً غیرممکن می‌کند.

ثبت حالت خارجی
تراکنش‌ها می‌توانند حالت گذار خود (در یک ماشین حالت خارجی) را از طریق عملگر OP_RETURN به صورت داده به بیرون بفرستند.

مقدار نشر محدود و قابل پیش‌بینی

مقدار نشر بیت‌کوین آهنگی قابل پیش‌بینی دارد، و مقدار کل آن هرگز به سقف ۲۱ میلیون بیت‌کوین نخواهد رسید.

البته این فهرست کامل نیست و با اضافه شدن ویژگی جدید به بیت‌کوین، عناصر ساختمانی بیشتری به آن اضافه خواهند شد.

ساخت برنامه‌ی کاربردی از عناصر ساختمانی

عناصر ساختمانی بیت‌کوین یک بستر اعتماد می‌سازند که می‌توان از آن برای ساخت برنامه‌های کاربردی استفاده کرد. در زیر به چند نمونه از کاربردهای این عناصر ساختمانی (در برنامه‌هایی که موجود هستند) اشاره می‌کنیم:

اثبات-وجود (دفتر اسناد رسمی دیجیتال)

برگشت‌ناپذیری + برچسب زمانی + انقضاناپذیری. با ثبت یک اثرانگشت دیجیتال در بلاک چین می‌توان وجود یک سند در زمان ثبت آن را اثبات کرد (برچسب زمانی). این اثرانگشت بعد از ثبت قابل تغییر نیست (برگشت‌ناپذیری)، و برای همیشه در بلاک چین ذخیره خواهد شد (انقضاناپذیری).

سرمایه‌سازی (فانوس دریایی)

ثبتات + یکپارچگی + انسجام. اگر ورودی و خروجی یک تراکنش جمع‌آوری سرمایه را امضا کنید (انسجام)، دیگران همچنان می‌توانند در این تراکنش شرکت کنند (یکپارچگی)، تا زمانی که هدف [مقدار خروجی] مشخص شده برآورده شود (ثبتات).

کانال پرداخت

حد نصاب + قفل زمانی + عدم امکان خرج-دوباره + انقضاناپذیری + انسدادناپذیری + بررسی مجوز. از یک اسکریپت چندامضایی «۲-از-۲» (حد نصاب) با قفل زمانی مشخص (قفل زمانی) می‌توان به عنوان تراکنش «تسویه حساب» یک کانال پرداخت استفاده کرد، تراکنشی که منقضی نمی‌شود (انقضاناپذیری) و هر یک از گیرندگان (بررسی مجوز) می‌توانند آن را در هر زمان دلخواه خرج کنند (انسدادناپذیری). این گیرندگان سپس می‌توانند با ایجاد تراکنش‌های واسپاری برای خرج کردن این تراکنش «تسویه حساب» (عدم امکان خرج-دوباره) با قفل زمانی کوتاه‌تر (قفل زمانی) آن را خرج کنند.

سکه‌ی رنگی

اولین کاربرد بلاک‌چین که در این فصل بررسی خواهیم کرد، سکه‌ی رنگی (colored coin) است.

سکه‌ی رنگی به مجموعه‌ای از فناوری‌های مشابه گفته می‌شود که از تراکنش‌های بیت‌کوین برای تولید، مالکیت و تبادل دارایی‌های برون‌زاد غیر از بیت‌کوین استفاده می‌کنند. برون‌زاد (extrinsic) به آن معنا است که این دارایی‌ها برخلاف خود بیت‌کوین که یک دارایی درون‌زاد بلاک‌چین است، به طور مستقیم در بلاک‌چین بیت‌کوین ذخیره نمی‌شوند. از سکه‌ی رنگی برای ثبت و ردگیری دارایی‌های دیجیتال و همچنین دارایی‌های فیزیکی (ملموس) اشخاص ثالث، و خرید و فروش آنها از طریق گواهی مالکیت استفاده می‌شود. سکه‌ی رنگی دارایی دیجیتال می‌تواند شامل گواهی سهام، امتیاز رسمی، اموال مجازی (اقلام بازی)، یا تقریباً هر نوع امتیاز مالکیت معنوی (علامت تجاری، حق‌التألیف، و غیره) باشد. سکه‌ی رنگی دارایی ملموس نیز شامل گواهی مالکیت کالا (طلا، نقره، نفت)، سند زمین، اتومبیل، کشتی، هواپیما و غیره است.

اصطلاح «رنگی» به معنای مشخص بودن ارزش آن (مثلاً، ۱ ساتوشی، ۱۰۰ ساتوشی، و غیره) است، درست مثل سکه یا اسکناسی که ارزش مشخصی دارد. اولین پیاده‌سازی سکه‌ی رنگی EPOBC بود که به یک خروجی ۱-ساتوشی ارزش برون‌زاد نسبت می‌داد. پیاده‌سازی‌های جدیدتر سکه‌ی رنگی از عملگر اسکریپت OP_RETURN برای ذخیره کردن فراداده (در همراهی با پایگاه‌های داده‌ی خارجی که این فراداده را به دارایی‌های مشخص مرتبط می‌سازند) در یک تراکنش استفاده می‌کنند. مشهورترین پیاده‌سازی‌های سکه‌ی رنگی عبارتند از: OpenAssets [<http://www.openassets.org/>] و Colu [<http://coloredcoins.org/>]; نکته‌ی منفی آن که این دو پیاده‌سازی با یکدیگر سازگار نیستند، و از سکه‌های یک سیستم نمی‌توان در دیگری استفاده کرد.

استفاده از سکه‌ی رنگی

سکه‌های رنگی معمولاً از طریق یک کیف پول مخصوص ایجاد، منتقل و مدیریت می‌شوند، کیف پولی که می‌تواند فراداده‌ی پروتکل سکه‌ی رنگی متصل به تراکنش‌های بیت کوین را تفسیر کند. از آنجا که ممکن است فراداده‌ی سکه‌های رنگی در یک کیف پول بیت کوین معمولی از بین بروند، باید در انتقال آنها به برنامه‌های کیف پول مرسوم دقت کرد. به طریق مشابه، سکه‌های رنگی را نیز نباید به آدرس‌هایی که توسط کیف پول‌های معمولی مدیریت می‌شوند، فرستاد؛ یک سکه‌ی رنگی را فقط به کیف پولی بفرستید که این نوع بیت کوین را شناخته و قابلیت مدیریت آنها را داشته باشد. هر دو سیستم Open Assets و Colu از آدرس‌های سکه‌ی-رنگی مخصوص برای اجتناب از این خطر و اطمینان از آن که این سکه‌ها به کیف پول‌های غیرسازگار یا سکه‌ی رنگی ارسال نخواهند شد، استفاده می‌کنند.

همچنین، اکثر برنامه‌های کاوشگر بلاک چین همه-منظوره قادر به شناسایی و دیدن سکه‌های رنگی نیستند؛ به جای آن باید از یک کاوشگر سکه‌ی رنگی که قادر به خواندن و تفسیر فراداده‌ی تراکنش‌های سکه‌ی رنگی باشد، استفاده کنید. برای دریافت برنامه‌ی کیف پول و کاوشگر بلاک چین سازگار با سیستم Open Assets به <https://www.coinprism.info/>، و برای دریافت برنامه‌ی کیف پول و کاوشگر بلاک چین سازگار با سیستم Colu به <http://coloredcoins.org/explorer/> مراجعه کنید. کیف پول معمولی کوپبی برای مدیریت سکه‌های رنگی یک افزونه عرضه کرده است که می‌توانید آن را از <http://coloredcoins.org/colored-coins-copay-addon/> دریافت کنید.

نشر سکه‌ی رنگی

هر یک از پیاده‌سازی‌های سکه‌ی رنگی از روشی متفاوت برای تولید سکه‌ی رنگی استفاده می‌کنند، ولی همه‌ی آنها عملکرد یکسانی دارند، در فرآیند تولید سکه‌ی رنگی که به آن نشر گفته می‌شود، یک تراکنش آغازگر به نام تراکنش نشر (issuance transaction) دارایی مورد نظر را در بلاک چین بیت کوین ثبت کرده و یک شناسه‌ی دارایی (asset ID) ایجاد می‌کند که برای ارجاع به آن دارایی به کار می‌رود. همین که یک دارایی نشر یافت، می‌توان آن را از طریق تراکنش انتقال (transfer transaction) بین آدرس‌های سکه‌ی رنگی رد و بدل کرد.

دارایی‌هایی که به صورت سکه‌ی رنگی منتشر می‌شوند، می‌توانند خواص متعددی داشته باشند. یک سکه‌ی رنگی می‌تواند تقسیم‌پذیر یا تقسیم‌ناپذیر باشد، یعنی مقدار دارایی در یک تراکنش انتقال می‌تواند یک عدد صحیح (مثل ۵) یا یک عدد اعشاری (مثل ۴٫۳۲۱) باشد. همچنین، یک دارایی می‌تواند نشر ثابت (fixed issuance) داشته باشد، یعنی فقط یک بار و به مقداری مشخص منتشر شده، یا بازنشر (reissue) هم داشته باشد، یعنی بعد از نشر اولیه (توسط تراکنش آغازگر) مقادیر جدیدی از آن دارایی منتشر شود.

سرانجام، برخی از سکه‌های رنگی امکان دادن تقسیم سود هم دارند، یعنی اجازه می‌دهند تا پرداخت‌های بیت‌کوین به مالکان یک دارایی سکه‌ی رنگی به نسبت سهم مالکیت بین آنها تقسیم شود.

تراکنش سکه‌ی رنگی

فرا داده‌ای که به یک تراکنش سکه‌ی رنگی معنا می‌دهد، معمولاً به کمک کد اجرایی OP_RETURN در یکی از خروجی‌ها که به آن خروجی نشان‌گذار (marker output) گفته می‌شود، ذخیره خواهد شد. پروتکل‌های مختلف سکه‌ی رنگی از کدگذاری‌های متفاوتی برای محتویات داده‌ی OP_RETURN استفاده می‌کنند.

ترتیب خروجی‌ها و موقعیت خروجی نشان‌گذار می‌تواند معنای خاصی در پروتکل سکه‌ی رنگی داشته باشد. برای مثال، در Open Assets هر خروجی قبل از خروجی نشان‌گذار یک تراکنش نشر (دارایی)، و هر خروجی بعد از خروجی نشان‌گذار نشانه‌ی یک تراکنش انتقال است. خروجی نشان‌گذار، با ارجاع به ترتیب (موقعیت مکانی) خروجی‌های تراکنش، مقادیر و رنگ‌های خاصی به این خروجی‌ها نسبت می‌دهد.

از طرف دیگر، در سیستم Colu خروجی نشان‌گذار یک کد اجرایی را کدگذاری می‌کند که چگونگی تفسیر فراداده را مشخص خواهد کرد. کدهای 0x01 تا 0x0F نشانه‌ی یک تراکنش نشر هستند. به دنبال هر کد نشر یک شناسه‌ی دارایی (یا شناسه‌ی دیگر) می‌آید که می‌توان از آن برای بازیابی اطلاعات این دارایی از یک منبع خارجی (مثل، بیت‌تورنت) استفاده کرد. کدهای 0x10 تا 0x1F نشانه‌ی یک تراکنش انتقال هستند. فراداده‌ی تراکنش انتقال فقط حاوی اسکرپت‌هایی است که مقدار انتقال دارایی از ورودی (ها) به خروجی (ها) را [با ارجاع به اندیس آنها] مشخص می‌کند، بنابراین ترتیب این ورودی (ها) و خروجی (ها) برای تفسیر اسکرپت مهم است.

اگر فراداده آنقدر بزرگ باشد که در OP_RETURN جا نشود، پروتکل سکه‌ی رنگی می‌تواند از ترفندهای دیگر برای ذخیره‌سازی این فراداده استفاده کند، مثلاً قرار دادن فراداده در اسکرپت وصول (redeem script)، به همراه کدهای اجرایی OP_DROP برای اطمینان از آن که این اسکرپت آنها را نادیده خواهد گرفت. ساز و کار دیگری که به کار گرفته شده، استفاده از یک اسکرپت چندامضایی «۱-از-N» است که در آن فقط اولین کلید عمومی یک کلید عمومی واقعی است که می‌تواند این خروجی را خرج کند، و فراداده‌ی کدگذاری‌شده در کلیدهای بعدی نوشته می‌شود.

برای تفسیر درست فراداده در یک تراکنش سکه‌ی رنگی لازم است از کیف پول یا کاوشگر بلاک مناسب استفاده کنید؛ در غیر این صورت، این تراکنش درست مثل یک تراکنش «معمولی» بیت‌کوین با یک خروجی OP_RETURN به نظر خواهد رسید. به عنوان مثال، در اینجا با استفاده از پروتکل سکه‌ی رنگی یک دارایی به نام MasterBTC ایجاد و نشر کرده‌ایم. این دارایی MasterBTC رسید یک نسخه‌ی هدیه از کتاب حاضر است. این قبیل رسیدها را می‌توان انتقال داد، معامله کرد، و یا با استفاده از یک کیف پول سازگار با سکه‌ی رنگی وصول کرد. در این مثال خاص از کیف پول و کاوشگر بلاک <http://coinprism.info/> که با پروتکل سکه‌ی رنگی Open Assets سازگار است، استفاده کرده‌ایم. شکل ۱-۱۲ تراکنش نشر این دارایی (<https://www.coinprism.info/tx/10d7c4e022f35288779be6713471151ede967caaa39eecd35296aa36d9c109ec>) را در کاوشگر بلاک Coinprism نشان می‌دهد. همان طور که کاوشگر Coinprism نشان می‌دهد، این تراکنش ۲۰ نسخه‌ی مجانی از کتاب حاضر به آدرس سکه‌ی رنگی `akTnsDt5uzpioRST76VFRQM8q8sBFnQiwcx` منتشر کرده است.

از آدرسی که در این مثال می‌بینید، به هیچ وجه استفاده نکنید! هر مبلغ یا سکه‌ی رنگی که به این آدرس فرستاده شود، برای همیشه از بین خواهد رفت.



Transaction

Hash	10d7c4e022f35288779be6713471151ede967c...
Date	Sunday, August 17, 2014 5:42:41 PM
Fee paid	0.0001 BTC
Assets transacted	1

Transaction confirmed

Confirmations	137057 confirmations
Time	Sunday, August 17, 2014 5:...
Block	0000000000000000000150ab5...
Height	316117

Bitcoin

	← akTnsDtSuzpioRST76VFRQM8q8sBF...	-0.0001	Fees	0.0001
Free copy of "Mastering Bitcoin" AcuRVsoa81hoLHmVTNxRD8KpTqLXe...				
	+ Issued assets	-20	akTnsDtSuzpioRST76VFRQM8q8sBFnQ...	20

شکل ۱۲-۱ نمایش تراکنش نشر در کاوشگر coinprism.info.

شناسه‌ی تراکنش در تراکنش‌های نشر همانند یک شناسه‌ی «معمولی» بیت‌کوین است. اگر همین تراکنش را در یک کاوشگر بلاک که با سکه‌ی رنگی سازگار نیست (مانند blockchain.info)، باز کنید، آن را مانند یک تراکنش معمولی خواهید دید؛ شکل ۱۲-۲ را ببینید.

Transaction View information about a bitcoin transaction

10d7c4e022f35288779be6713471151ede967c...

1HgyyGxKLq7ZCfH3d8EVEFuG15oLn7Us (0.01 BTC - Output)



1HgyyGxKLq7ZCfH3d8EVEFuG15oLn7Us - (Unspent)
 Unable to decode output address - (Unspent)
 1HgyyGxKLq7ZCfH3d8EVEFuG15oLn7Us - (Spent)

0.000006 BTC
 0 BTC
 0.000004 BTC

0.0001 BTC

شکل ۱۲-۲ نمایش تراکنش نشر در یک کاوشگر بلاک که با سکه‌ی رنگی سازگار نیست.

همان طور که می‌بینید، کاوشگر blockchain.info قادر به تشخیص این تراکنش به عنوان یک تراکنش سکه‌ی رنگی نیست؛ در واقع، این کاوشگر خروجی دوم تراکنش را به عنوان «Unable to decode output address» شناسایی کرده است. اگر در این صفحه گزینه‌ی «Show scripts & coinbase» را انتخاب کنید، جزئیات بیشتری از این تراکنش خواهید دید؛ شکل ۱۲-۳ را ببینید.

از آنجا که کاوشگر blockchain.info قادر به تشخیص خروجی دوم نیست، آن را با علامت «Strange» علامت‌گذاری کرده است، ولی همچنان می‌توانید فراداده‌ای موجود در این خروجی نشان‌گذار را ببینید:

OP_RETURN 4f41010001141b753d68747470733a2f2f6370722e736d2f466f796b777248365559
 (decoded) "0A^^^u=https://cpr.sm/FoykwrH6UY"

Output Scripts

OP_DUP OP_HASH160 3d8e347d8b1a6e342a114603d3d462 OP_EQUALVERIFY OP_CHECKSIG	OK
OP_RETURN 4f41010001141b753d68747470733a2f2f6370722e736d2f466f796b777248365559 (decoded) "0A^^^u=https://cpr.sm/FoykwrH6UY"	Script
OP_DUP OP_HASH160 3d8e347d8b1a6e342a114603d3d462 OP_EQUALVERIFY OP_CHECKSIG	OK

شکل ۱۲-۳ اسکریپت‌های تراکنش نشر.

اگر با فرمان زیر تلاش کنیم اطلاعات این تراکنش را با استفاده از bitcoin-cli به دست آوریم:

```
$ bitcoin-cli decoderawtransaction 'bitcoin-cli getrawtransaction
10d7c4e022f35288779be6713471151ede967caaa39eecd35296aa36d9c109ec'
```

خروجی دوم به صورت زیر خواهد بود (برای تمرکز روی خروجی دوم، بقیه‌ی قسمت‌های خروجی را حذف کرده‌ایم):

```
{
  "value": 0.00000000,
  "n": 1,
  "scriptPubKey": "OP_RETURN
4f41010001141b753d68747470733a2f2f6370722e736d2f466f796b777248365559"
}
```

پیشوند 4f41 در مقدار کُد OP_RETURN معادل حروف «OA» مخفف «Open Assets» است که کمک می‌کند پروتکل کدگذاری این قرارداد را تشخیص دهیم. مقادیر بعدی هم معادل هگزادسیمال یک رشته‌ی کاراکتری با کدگذاری آسکی هستند که اگر آن را کدگشایی کنیم، به رشته‌ی `u=https://cpr.sm/FoykwrH6UY` خواهیم رسید. دنبال کردن این URL هم ما را به تعریف این دارایی (با کدگذاری JSON) می‌رساند:

```
{
  "asset_ids": [
    "AcuRVsoa81hoLHmVTNXrRD8KpTqUXeqwGH"
  ],
  "contract_url": null,
  "name_short": "MasterBTC",
  "name": "Free copy of \"Mastering Bitcoin\"",
  "issuer": "Andreas M. Antonopoulos",
  "description": "This token is redeemable for a free copy of the book \"Mastering Bitcoin\"",
  "description_mime": "text/x-markdown; charset=UTF-8",
  "type": "Other",
  "divisibility": 0,
  "link_to_website": false,
  "icon_url": null,
  "image_url": null,
  "version": "1.0"
}
```

قرینگی

قرینگی (Counterparty) یک لایه‌ی پروتکل است که روی بیت‌کوین ساخته می‌شود. مانند سکه‌های رنگی، با پروتکل قرینگی می‌توان دارایی و ژتون مجازی تولید و داد و ستد کرد. علاوه بر آن، قرینگی امکان مبادله‌ی غیر متمرکز دارایی‌ها را نیز فراهم می‌کند. قرینگی همچنین می‌تواند قراردادهای هوشمند را بر مبنای ماشین مجازی اتریوم (Ethereum Virtual Machine)، یا EVM، پیاده‌سازی کند.

درست مثل پروتکل‌های سکه‌ی رنگی، پروتکل قرینگی قرارداد را با استفاده از کُد عملیاتی OP_RETURN با آدرس‌های چند امضایی «۱-از-N» [که قرارداد را به جای کلیدهای عمومی قرار می‌دهد] در تراکنش‌های بیت‌کوین

جامسازی می‌کند. با استفاده از این ساز و کارها، قرینگی یک لایه‌ی پروتکل در داخل تراکنش‌های بیت‌کوین می‌سازد. البته (مانند سکه‌های رنگی) فقط آن دسته از برنامه‌های کیف پول و کاوشگرهای بلاک‌چین که با پروتکل قرینگی سازگار باشند، یا برنامه‌های کاربردی که از کتابخانه‌های قرینگی استفاده کرده باشند، می‌توانند اطلاعات این لایه‌ی پروتکل اضافی را تفسیر (کدگشایی) کنند.

پروتکل قرینگی به نوبه‌ی خود می‌تواند به عنوان یک بستر برای تولید برنامه‌های کاربردی و سرویس‌های مختلف مورد استفاده قرار گیرد. برای مثال، Tokenly بستری است که روی پروتکل قرینگی ساخته شده و اجازه می‌دهد شرکت‌ها، هنرمندان و تولیدکنندگان محتوا برای اعلام مالکیت خود ژتون منتشر کنند و این محتواها، محصولات و سرویس‌ها را خرید و فروش کرده یا اجاره دهند. از دیگر برنامه‌هایی که از پروتکل قرینگی استفاده می‌کنند، می‌توان به بازی‌های کامپیوتری (مثل افسون آفریش) و پروژه‌های پردازش گسترده (مانند فولدینگ کوین) اشاره کرد. برای اطلاعات بیشتر درباره‌ی پروتکل قرینگی به <https://counterparty.io/>، و برای دریافت فایل‌های منبع-باز این پروژه به <https://github.com/CounterpartyXCP> مراجعه کنید.

کانال پرداخت و کانال حالت

کانال پرداخت (payment channel) یک ساز و کار بدون اعتماد برای مبادله‌ی تراکنش‌های بیت‌کوین بین دو طرف، خارج از چارچوب بلاک‌چین بیت‌کوین، است. این تراکنش‌ها که فقط پس از ثبت در بلاک‌چین بیت‌کوین معتبر خواهند شد، در واقع معاملات برون-زنجیره (off-chain) هستند که نقش سفته را در تسویه‌های آتی ایفا می‌کنند. از آنجا که این تراکنش‌ها تسویه نمی‌شوند، می‌توان آنها را بدون تأخیر تسویه‌ی معمول مبادله کرد؛ به عبارت دیگر، تراکنش‌های کانال پرداخت را می‌توان با حجم بسیار بالا، تأخیر ناچیز (زیر میلی ثانیه)، و ریزدانه‌ی زیاد (در حد ساتوشی) انجام داد.

در حقیقت، واژه‌ی کانال معنای استعاری دارد. کانال حالت یک سازه‌ی مجازی است که مبادله‌ی حالت بین دو طرف (خارج از بلاک‌چین) را نشان می‌دهد. در واقع، هیچ «کانال» واقعی بین طرفین وجود ندارد و ساز و کار انتقال زیرین هم یک کانال نیست. اصطلاح کانال را فقط برای نمایش این رابطه و حالت مشترک بین دو طرف (خارج از بلاک‌چین) به کار می‌بریم. برای درک بهتر این مفهوم، یک استریم TCP را در نظر بگیرید. از دید برنامه‌های کاربردی سطح-بالا این استریم TCP یک «سوکت» است که دو برنامه‌های کاربردی را از طریق اینترنت به یکدیگر متصل می‌کند. ولی اگر به ترافیک شبکه دقت کنید، یک استریم TCP چیزی نیست جز یک کانال مجازی روی بسته‌های IP. نقاط انتهایی این استریم TCP بسته‌های IP را بر اساس شماره‌ی توالی آنها مرتب کرده و به یکدیگر می‌چسبانند تا توهمی از یک استریم بایت [پیوسته] ایجاد کنند. در لایه‌های زیرین، چیزی جز بسته‌های منفصل وجود ندارد. به طریق مشابه، یک کانال پرداخت چیزی نیست جز یکسری تراکنش، که وقتی به درستی مرتب شده و به یکدیگر متصل شوند، نوعی تعهد قابل وصول ایجاد می‌کنند که می‌توان (حتی وقتی هیچ اعتمادی نسبت به سمت مقابل کانال وجود ندارد) به آن اعتماد کرد.

در این قسمت صورت‌های مختلف کانال پرداخت را بررسی خواهیم کرد. ابتدا، ساز و کارهای مورد استفاده برای ایجاد یک کانال پرداخت یک-طرفه برای سرویس‌های ریزپرداخت (مثل استریم ویدئوی پولی) را تشریح می‌کنیم. سپس، این ساز و کار یک-طرفه را توسعه داده و کانال پرداخت دو-طرفه را معرفی خواهیم کرد. سرانجام، نشان می‌دهیم که چگونه می‌توان این کانال‌های دو-طرفه را در یک شبکه‌ی مبتنی بر مسیریابی به صورت نقطه-به-نقطه به یکدیگر متصل کرد، کاری که اولین بار تحت نام شبکه‌ی آذرخش انجام شد.

کانال پرداخت بخشی از یک مفهوم وسیع‌تر موسوم به کانال حالت (state channel) است که تغییر حالت برون-زنجیره را نمایش می‌دهد، تغییر حالتی که در نهایت در یک بلاک‌چین تسویه (ثبت) خواهد شد. کانال پرداخت در واقع یک کانال حالت است که در آن حالت تغییر یابنده چیزی نیست جز تراز یک پول (ارز) مجازی.

کانال حالت: مفاهیم پایه و واژه‌شناسی

یک کانال حالت وقتی بین دو طرف برقرار می‌شود که یک تراکنش خاص که حالتی مشترک بین آنها را در بلاک چین قفل می‌کند، ایجاد شود. به این تراکنش خاص تراکنش تأمین سرمایه (funding transaction) یا تراکنش قلاب (anchor transaction) گفته می‌شود. برای ایجاد و برقراری کانال، این تراکنش باید در شبکه‌ی بیت کوین منتشر شده و استخراج شود. در مورد کانال پرداخت، این حالت قفل شده تراز اولیه‌ی کانال (بر حسب ارز مورد نظر) است.

پس از آن، دو طرف کانال تعدادی تراکنش امضا شده، موسوم به تراکنش تعهد (commitment transaction)، مبادله می‌کنند که این حالت اولیه را تغییر می‌دهند. این تراکنش‌ها همگی تراکنش‌های معتبر هستند، از این لحاظ که هر یک از طرفین می‌توانند آنها را تسویه کنند، ولی عمده‌اً تا زمان بسته شدن کانال آنها را معوق نگه می‌دارند. سرعت تغییر حالت فقط محدود به توانایی طرفین کانال در ایجاد و امضای تراکنش و ارسال آن به طرف مقابل است. در عمل این به آن معنا است که طرفین یک کانال می‌توانند در هر ثانیه هزاران تراکنش مبادله کنند.

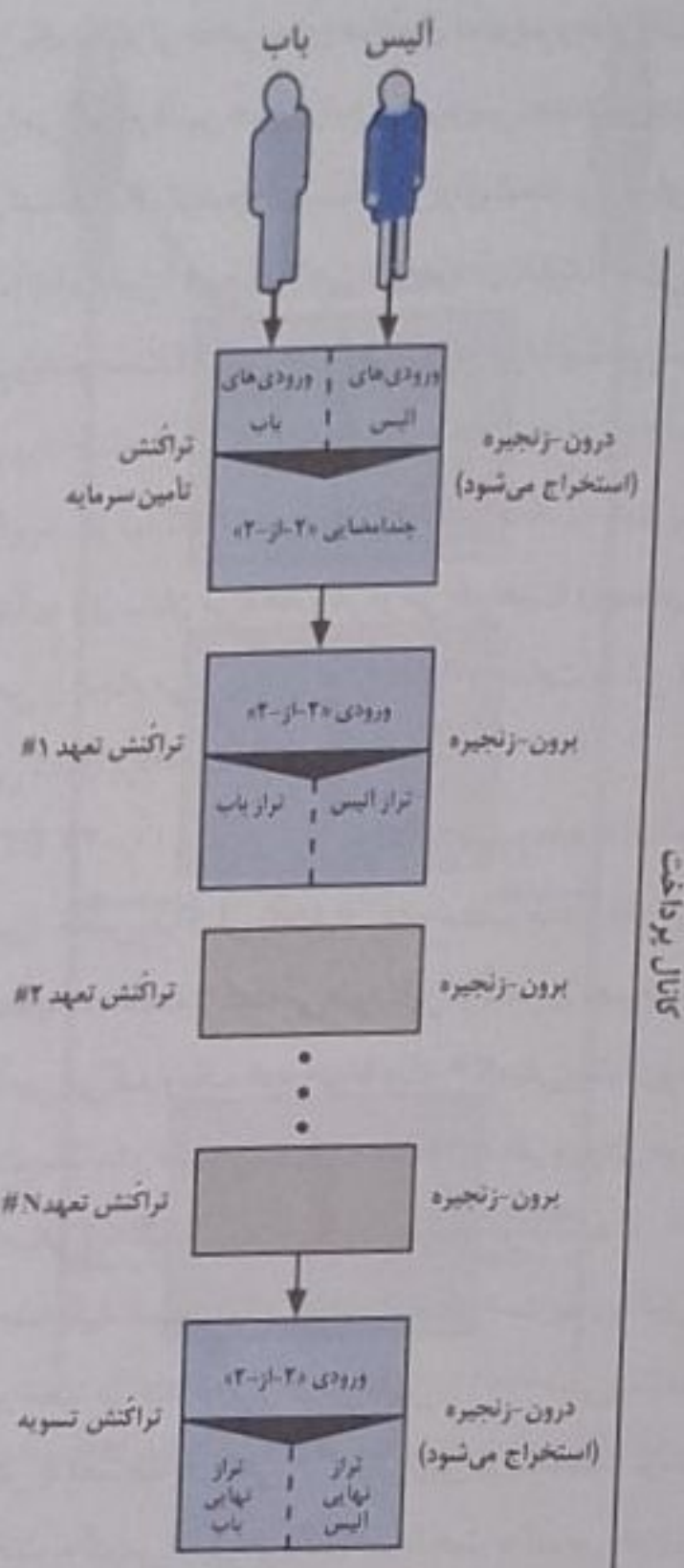
در مبادله‌ی تراکنش‌های تعهد، دو طرف کانال حالت‌های قبلی را نامعتبر می‌کنند، به طوری که همیشه فقط یک تراکنش تعهد قابل وصل می‌تواند وجود داشته باشد. این ویژگی مانع از آن می‌شود تا اگر هر یک از طرفین بخواهند تقلب کنند، نتوانند به طور یکطرفه و با یک حالت منقضی شده که بیشتر از حالت فعلی از آن منتفع می‌شوند، اقدام به بستن کانال کنند. ادامه‌ی این فصل به بررسی ساز و کارهایی خواهیم پرداخت که می‌توانند برای نامعتبر کردن حالت قبلی به کار گرفته شوند. سرانجام، کانال می‌تواند با همکاری و موافقت طرفین، با ارسال تراکنش تسویه (settlement transaction) نهایی به بلاک چین، یا به صورت یکطرفه، با ارسال آخرین تراکنش تسویه به بلاک چین از سوی هر یک از دو طرف کانال، بسته شود. گزینه‌ی بستن یکطرفه برای مواردی لازم است که اتصال یکی از طرفین به طور غیرمنتظره قطع شود. تراکنش تسویه نهایی نشان‌دهنده‌ی آخرین حالت (وضعیت) کانال است و در بلاک چین ثبت (تسویه) می‌شود.

در تمام طول عمر یک کانال فقط دو تراکنش باید برای استخراج شدن به بلاک چین ارسال شوند: تراکنش تأمین سرمایه و تراکنش تسویه. در بین این دو تراکنش، طرفین کانال می‌توانند به هر تعداد تراکنش تعهد مبادله کنند، ولی هیچ کس (جز خود آنها) این تراکنش‌ها را نخواهد دید و هیچ کدام از آنها به بلاک چین فرستاده نخواهند شد. در شکل ۱۲-۴ یک کانال پرداخت بین آلیس و باب می‌بینید که تراکنش‌های تأمین سرمایه، تعهد و تسویه در آن نشان داده شده‌اند.

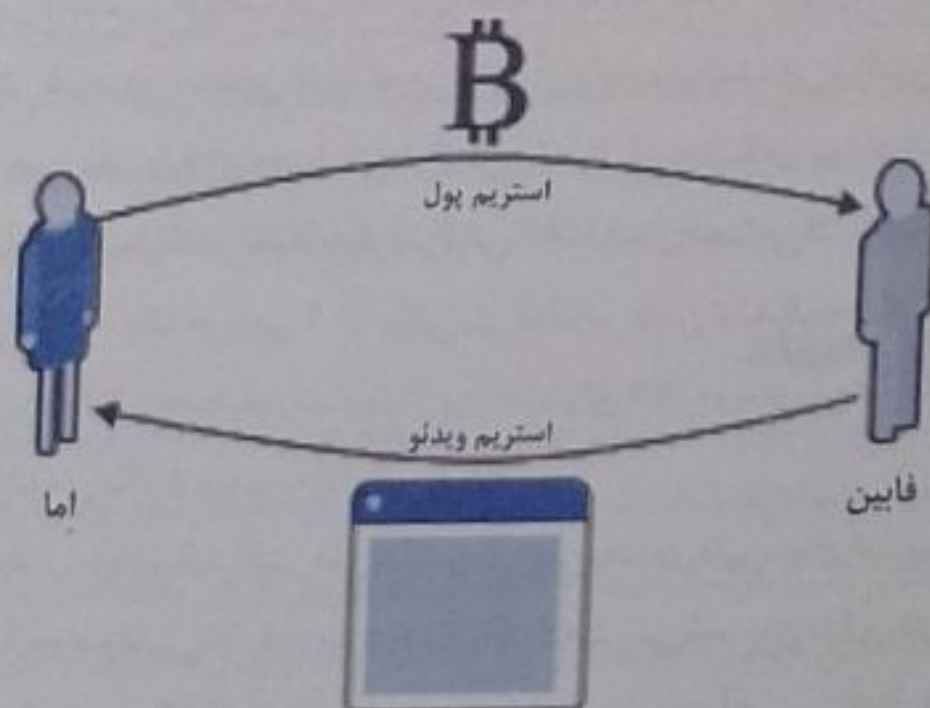
نمونه‌ای از کانال پرداخت ساده

تشریح کانال‌های حالت را با یک مثال بسیار ساده شروع می‌کنیم. کانالی که در این قسمت نشان می‌دهیم، یک کانال یک-طرفه (unidirectional) است، که در آن دارایی (پول یا هر ارزش دیگر) فقط در یک جهت جریان می‌یابد. همچنین، برای سادگی بحث، با این فرض ساده‌انگارانه شروع می‌کنیم که هیچ از طرفین کانال قصد تقلب ندارند. همین که با ایده‌ی اساسی کانال آشنا شدید، به چگونگی ایجاد کانال‌های بدون اعتماد خواهیم پرداخت که در آنها طرفین کانال نمی‌توانند تقلب کنند، حتی اگر قصد آن را داشته باشند و یا به این کار اقدام کنند.

در مثال این قسمت فرض می‌کنیم کانال پرداخت ما دو طرف به نام‌های اِما و فابین دارد. فابین یک سرویس استریم ویدئو دارد که با استفاده از یک کانال ریزپرداخت (micropayment channel)، صورت‌حساب آن را بر حسب ثانیه صادر می‌کند. قیمت این سرویس ۰/۰۱ میلی‌بیت [کوین] (۰/۰۰۰۰۱ BTC) بر ثانیه ویدئو، معادل ۳۶ میلی‌بیت (۰/۰۳۶ BTC) بر ساعت ویدئو است. اِما هم یک کاربر است که این سرویس استریم ویدئو را از فابین خریده است. رابطه‌ی اِما و فابین در شکل ۱۲-۵ نشان داده شده است.



شکل ۴-۱۲ یک کانال پرداخت بین آلیس و باب، به همراه تراکنش های تأمین سرمایه، تعهد و تسویه.



شکل ۵-۱۲ اما از فابین یک استریم ویدئو می خرد و بهای آن را بر حسب ثانیه از طریق یک کانال پرداخت می پردازد.

در این مثال، اما و فابین از یک نرم افزار خاص برای مبادله‌ی استریم ویدئو و کانال پرداخت استفاده می کنند. اما این نرم افزار را در مرورگر وب خود اجرا می کند، و فابین هم آن را روی سرور ویدئو خود نصب و اجرا کرده است. این نرم افزار در داخل خود کارکردهای اساسی یک کیف پول بیت کوین برای ایجاد و امضای تراکنش های بیت کوین را دارد. توجه کنید که طرفین این کانال پرداخت (اما و فابین) هیچ اطلاعی از وجود آن ندارند و حتی نمی دانند «کانال پرداخت» چیست. چیزی که آنها می بینند یک استریم ویدئو است که برای هر ثانیه‌ی آن پول پرداخت می شود.

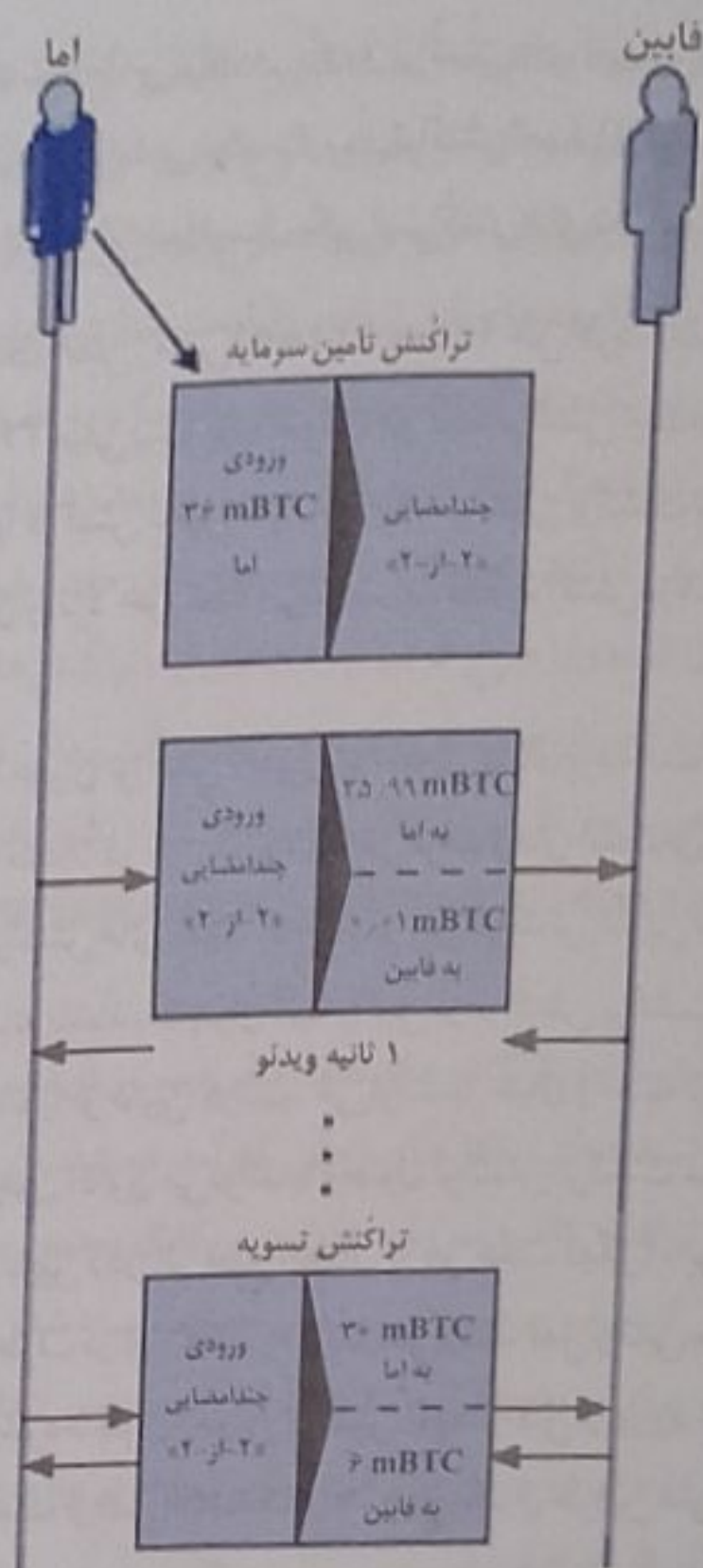
برای برپایی این کانال پرداخت، اما و فابین یک آدرس چندامضایی «۲-از-۲» می سازند و هر یک از آنها کنترل کلید خود را در دست می گیرند. از دیدگاه اما، نرم افزار [کانال پرداخت- استریم ویدئو] یک کد QR با یک آدرس P2SH (که با «۳» شروع می شود) به وی نشان می دهد و از او می خواهد تا ودیعه‌ای معادل یک ساعت تماشای ویدئو بپردازد. اما پرداخت به این آدرس را انجام می دهد. تراکنش اما، پرداخت به این آدرس چندامضایی، همان تراکنش تأمین سرمایه یا تراکنش قلاب این کانال پرداخت است.

پرداخت ۳۶ میلی بیت (BTC ۰.۰۳۶) به عنوان تراکنش تأمین سرمایه به اما اجازه می دهد تا سقف یک ساعت استریم ویدئو تماشا کند. در این مورد خاص، تراکنش تأمین سرمایه سقف مقدار قابل انتقال در این کانال را تعیین می کند، مقداری که به آن ظرفیت کانال (channel capacity) گفته می شود. این تراکنش با مصرف یک یا چند ورودی در کیف پول اما، مقدار پول لازم برای ایجاد کانال را تأمین می کند و یک خروجی با مبلغ ۳۶ میلی بیت در وجه آدرس چندامضایی «۲-از-۲» به وجود می آورد که به طور مشترک توسط اما و فابین کنترل می شود. (البته اگر ورودی مورد استفاده در کیف پول اما بیش از BTC ۰.۰۳۶ باشد، یک خروجی اضافی [به عنوان تنه‌ی پول] نیز تولید خواهد شد.)

همین که تراکنش تأمین سرمایه تأیید شد، اما می تواند تماشای استریم ویدئو خود را شروع کند. برای شروع، نرم افزار اما یک تراکنش تعهد تولید و امضا می کند که تراز آدرس فابین را ۰.۰۱ میلی بیت بستانکار کرده و ۳۵/۹۹ میلی بیت به آدرس اما برمی گرداند. این تراکنش با مصرف خروجی ۳۶ میلی بیتی تولید شده توسط تراکنش تأمین سرمایه، دو خروجی تولید می کند: یک پرداخت به آدرس فابین، و یک بازپرداخت به آدرس خود اما. این تراکنش هنوز کاملاً قابل خرج کردن نیست، چون برای این منظور باید دو امضا داشته باشد، ولی در حال حاضر فقط امضای اما را دارد. وقتی سرویس دهنده‌ی فابین این تراکنش را دریافت می کند، آن را امضا کرده و به همراه ۱ ثانیه ویدئو به اما برمی گرداند. اکنون هر دو طرف کانال یک تراکنش تعهد دو-امضایی دارند که می توانند آن را وصول کنند؛ این تراکنش تراز کانال را به روز-رسانی می کند. اما طرفین کانال هنوز قصدی برای وصول این تراکنش و ارسال آن به شبکه‌ی بیت کوین ندارند.

در دور بعد، نرم افزار اما یک تراکنش تعهد دیگر (تراکنش #۲) تولید و امضا می کند که با استفاده از همان خروجی «۲-از-۲» تراکنش تأمین سرمایه، یک خروجی ۰.۰۲ میلی بیتی به آدرس فابین تولید کرده و یک خروجی ۳۵/۹۸ میلی بیتی به آدرس اما برمی گرداند. این تراکنش جدید عبارت است از پرداخت برای ۲ ثانیه ویدئو. نرم افزار فابین دومین تراکنش تعهد را نیز امضا کرده و به همراه ۱ ثانیه‌ی دیگر از ویدئو به اما برمی گرداند.

به همین ترتیب، نرم افزار اما با ارسال تراکنش های تعهد به سرویس دهنده‌ی فابین، بهای استریم ویدئو را پرداخت می کند. با گذشت هر ثانیه، تراز کانال به نفع فابین بالا رفته، و تراز حساب اما کاهش می یابد. برای مثال، وقتی ۶۰۰ ثانیه (۱۰ دقیقه) از شروع این استریم ویدئو می گذرد، ۶۰۰ تراکنش تعهد ایجاد و امضا شده است، و آخرین تراکنش تعهد (تراکنش #۶۰۰) دو خروجی دارد که تراز کانال را به دو بخش، ۶ میلی بیتی به آدرس فابین و ۳۰ میلی بیت برگشت به آدرس اما، تقسیم خواهد کرد.



شکل ۱۲-۶ کانال پرداخت بین اما و فابین، به همراه تراکنش‌های تعهد که تراز کانال را به‌روز می‌کنند.

سرانجام، اما دکمه‌ی «توقف» استریم ویدئو را کلیک می‌کند. اکنون اما یا فابین هر کدام می‌توانند تراکنش حالت نهایی را برای تسویه به شبکه‌ی بیت‌کوین ارسال کنند. این تراکنش آخر همان تراکنش تسویه است و بهای تمام استریم ویدئوی مصرف‌شده را به فابین پرداخت کرده و بقیه‌ی مبلغ تراکنش تأمین سرمایه را به اما برمی‌گرداند. شکل ۱۲-۶ کانال پرداخت بین اما و فابین را به همراه تراکنش‌های تعهد که تراز کانال را به‌روز می‌کنند، نشان می‌دهد. در پایان فقط دو تراکنش در بلاک‌چین [بیت‌کوین] ثبت می‌شوند: تراکنش تأمین سرمایه که کانال پرداخت را ایجاد می‌کند، و تراکنش تسویه که تراز نهایی را به درستی بین طرفین کانال تخصیص می‌دهد.

ایجاد کانال بدون اعتماد

کانالی که در قسمت قبل تشریح کردیم، به خوبی کار می‌کند، ولی فقط به شرط آن که دو طرف همکاری کنند، و هیچ تقلب یا نقضی پیش نیاید. اجازه دهید ببینیم این کانال تحت چه شرایطی می‌تواند دچار نقص شود و راه برطرف کردن آن نواقص چیست:

- همین که اما تراکنش تأمین سرمایه را ایجاد و امضا کند، برای پس گرفتن باقیمانده‌ی پول خود به امضای فابین نیاز خواهد داشت. اگر فابین به یکباره ناپدید شود، پول اما که در یک تراکنش «۲-از-۲» گیر کرده، عملاً برای همیشه از دست رفته است. کانالی پرداختی که در قسمت قبل دیدیم، می‌تواند (مثلاً) در صورت قطع ناگهانی ارتباط، قبل از آن که حداقل یک تراکنش تسویه به امضای دو طرف رسیده باشد، منجر به از بین رفتن سرمایه‌ی یکی از طرفین شود.

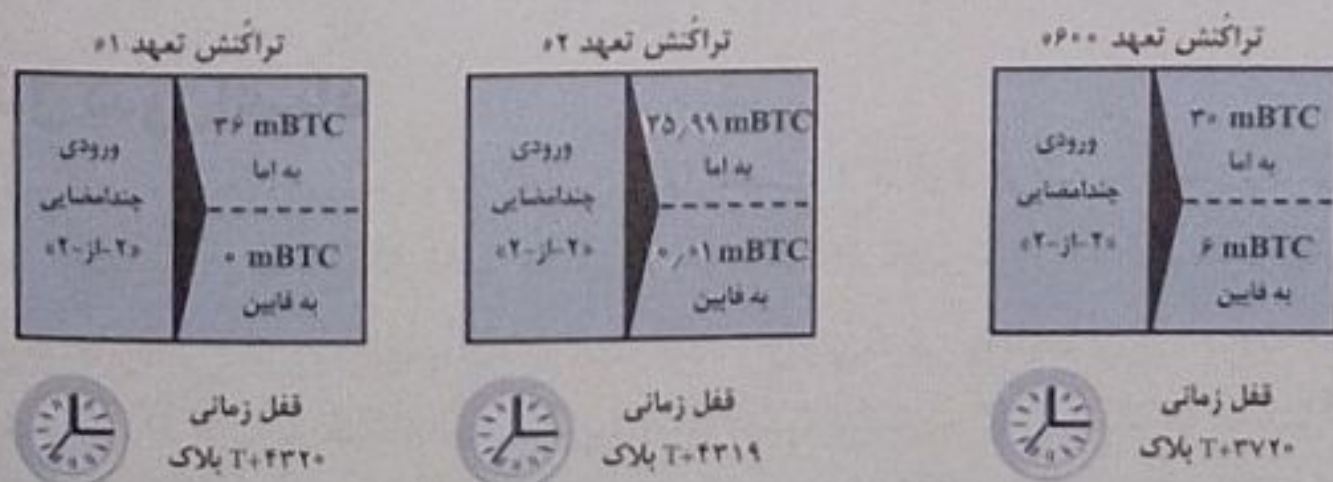
• وقتی کانال همچنان برقرار است، اما می‌تواند هر یک از تراکنش‌های تعهد امضا شده توسط فایین را انتخاب کرده و به بلاک چین بفرستد. برای مثال، وقتی اما می‌تواند با ارسال تراکنش تعهد #۱ هزینه‌ی فقط ۱ ثانیه ویدئو را بپردازد، چرا پول ۶۰۰ ثانیه را بدهد؟ در واقع، اما می‌تواند با ارسال یکی از تراکنش‌های تعهد قبلی به بلاک چین، دست به تقلب بزند.

هر دو مشکل بالا را می‌توان به کمک قفل زمانی (nLocktime) حل کرد. اجازه دهید ببینیم چگونه. اما نمی‌تواند بدون ضمانت برگشت سرمایه، ۳۶ میلی بیت پول خود را در یک تراکنش چند امضایی «۲-از-۲» به خطر بیندازد. برای حل این مشکل، اما همزمان با تراکنش تأمین سرمایه، یک تراکنش برگشت سرمایه نیز ایجاد می‌کند. او تراکنش تأمین سرمایه را امضا می‌کند، ولی آن را نزد خود نگه می‌دارد. اما فقط تراکنش برگشت سرمایه را امضا کرده و به فایین می‌فرستد و امضای او را می‌گیرد.

این تراکنش برگشت سرمایه به عنوان تراکنش تأمین سرمایه‌ی کانال پرداخت عمل کرده و قفل زمانی آن حداکثر عمر این کانال را مشخص می‌کند. مثلاً در این مورد، اما می‌تواند زمان انقضای قفل nLocktime را ۳۰ روز یا ۴۳۲۰ بلاک آینده تعیین کند. تمام تراکنش‌های تعهد که بعد از این تراکنش تأمین سرمایه تولید می‌شوند، باید عمری کوتاه‌تر از این قفل زمانی زمانی داشته باشند، تا بتوان آنها را قبل از تراکنش برگشت سرمایه وصول کرد. بعد از آن که اما تراکنش برگشت سرمایه امضا شده را از فایین گرفت، می‌تواند با خیال راحت تراکنش تأمین سرمایه را (که از قبل آماده و امضا کرده) به او ارسال کند، چون اکنون می‌تواند با وصول تراکنش برگشت سرمایه بعد از انقضای قفل زمانی، حتی در صورت ناپدید شدن ناگهانی فایین (در اثر قطع اتصال یا هر علت دیگر)، پول خود را پس بگیرد.

تمامی تراکنش‌های تعهد که دو طرف در این کانال رد و بدل می‌کنند، قفل زمانی خواهند داشت، ولی این قفل زمانی با مبادله‌ی هر تراکنش تعهد کوتاه‌تر می‌شود تا بتوان آخرین تراکنش تعهد را قبل از ورود تعهد بعدی که باعث نامعتبر شدن آن خواهد شد، وصول کرد. به خاطر استفاده از قفل nLocktime، هیچ یک از طرفین نمی‌توانند تراکنش‌های تعهد را قبل از انقضای قفل زمانی آنها به شبکه‌ی بیت کوین بفرستند. اگر همه چیز خوب پیش برود، آنها بدون دردسر و با همکاری یکدیگر کانال را با یک تراکنش تسویه می‌بندند، به طوری که دیگر نیازی به مبادله‌ی یک تراکنش تعهد اضافی بین آنها نیست. اما اگر اشکالی پیش آید، هر یک از طرفین می‌توانند با ارسال آخرین تراکنش تعهد [به شبکه‌ی بیت کوین] و نامعتبر کردن تمام تراکنش‌های تعهد قبلی، این حساب را تسویه کنند.

به عنوان مثال، اگر تراکنش تعهد #۱ برای ۴۳۲۰ بلاک بعد قفل شده باشد، آنگاه تراکنش تعهد #۲ باید برای ۴۳۱۹ بلاک بعد قفل شود؛ به همین ترتیب، تراکنش تعهد #۶۰۰ می‌تواند ۶۰۰ بلاک قبل از معتبر شدن تراکنش تعهد #۱ خرج شود. همان طور که در شکل ۱۲-۷ مشاهده می‌کنید، قفل زمانی هر تراکنش تعهد از تعهد قبلی کوتاه‌تر است تا بتوان قبل از معتبر شدن تراکنش تعهد قبلی، و همچنین قبل از سر رسید شدن تراکنش برگشت سرمایه، آن را خرج کرد.



شکل ۱۲-۷ هر تراکنش تعهد قفل زمانی کوتاه‌تری دارد، که اجازه می‌دهد قبل از معتبر شدن تعهدات قبلی خرج شود.

قابلیت ارسال زودرس یک تراکنش تعهد به شبکه‌ی بیت کوین تضمین می‌کند که گیرنده بتواند این خروجی را خرج کند، و همچنین مانع از وصول تراکنش‌های تعهد دیگر (از طریق خرج کردن خروجی آنها) خواهد شد. این ضمانت توسط بلاک چسب بیت کوین که جلوی خرج-دوباره‌ی یک خروجی و خرج کردن آن قبل از انقضای قفل زمانی را می‌گیرد، ارائه می‌شود و به هر تراکنش تعهد اجازه می‌دهد تا تعهدهای قبل از خود را نامعتبر سازد. [اولین نمونه از کانال پرداخت یک-طرفه در سال ۲۰۱۵ توسط یک گروه آرژانتینی در یک برنامه‌ی استریم ویدئو پیاده‌سازی و معرفی شد.]

کانال‌های حالت از قفل زمانی برای اعمال قراردادهای هوشمند در بُعد زمان استفاده می‌کنند. در این مثال دیدیم که چگونه این بُعد زمان تضمین می‌کند آخرین تراکنش تعهد قبل از تعهدهای قبلی معتبر شود. بدین ترتیب، آخرین تراکنش تعهد را می‌توان به شبکه‌ی بیت کوین ارسال کرده و با خرج کردن ورودی‌های آن، تراکنش‌های تعهد پیشین را نامعتبر کرد. اعمال قراردادهای هوشمند با قفل زمانی مطلق، که برای پیاده‌سازی آن به چیزی جز قفل زمانی مطلق سطح-تراکنش (nLocktime) نیاز نیست، همچنین مانع از تقلب هر یک از طرفین کانال می‌شود. در ادامه نشان خواهیم داد که چگونه می‌توان با استفاده از قفل زمانی سطح-اسکرپت (CHECKLOCKTIMEVERIFY و CHECKSEQUENCEVERIFY) کانال‌های حالت پیچیده‌تر، انعطاف‌پذیرتر و مفیدتر ایجاد کرد.

قفل زمانی تنها روش برای نامعتبرسازی تراکنش‌های تعهد قبلی نیست. در قسمت بعد خواهیم دید که چگونه می‌توان از یک کلید فسخ برای همین منظور استفاده کرد. قفل‌های زمانی کارآمد هستند، ولی دو عیب برجسته دارند. با تعیین یک مقدار حداکثر برای nLocktime در تراکنش تأمین سرمایه، طول عمر کانال محدود می‌شود. از آن بدتر این که، هر چه عمر کانال طولانی‌تر انتخاب شود، یکی از طرفین مجبور است، در صورت بسته شدن غیر مترقبه‌ی کانال، زمان بیشتری برای برگشت سرمایه‌ی خود انتظار بکشد. برای نمونه، اگر (مانند مثال قبل) طول عمر کانال ۳۰ روز تعیین شود، در صورت قطع ارتباط، هر یک از دو طرف باید ۳۰ روز برای سررسید پول خود منتظر بماند. در واقع، هر چه زمان nLocktime بیشتر باشد، انتظار برای برگشت سرمایه هم طولانی‌تر خواهد بود.

مشکل دیگر این که تعداد تراکنش‌های تعهدی که طرفین می‌توانند مبادله کنند، محدود است، چون با هر تراکنش تعهد باید یک واحد از قفل زمانی کاسته شود. در مثال قبل تعداد تراکنش‌های تعهد به ۴۳۲۰ محدود است، چون به nLocktime مقدار ۴۳۲۰ بلاک (معادل ۳۰ روز) داده شده است. تعیین فاصله‌ی زمانی ۱ بلاک بین تراکنش‌های تعهد خطر دیگری نیز در بر دارد. با این الزام فشار زیادی به طرفین کانال وارد می‌آید، چون آنها باید دائماً گوش به زنگ باشند، آنالین بمانند و ویدئو تماشا کنند، و آماده باشند تا در هر لحظه تراکنش تعهد مناسب را تولید و ارسال کنند.

اکنون که با چگونگی استفاده از قفل زمانی برای نامعتبرسازی تعهدهای پیشین آشنا شدیم، می‌توانیم تفاوت بین بستن دو-طرفه‌ی کانال، و بستن آن به صورت یک-طرفه از طریق ارسال یک تراکنش تعهد به شبکه‌ی بیت کوین را مشاهده کنیم. تمام تراکنش‌های تعهد قفل زمانی دارند، بنابراین انتشار یک تراکنش تعهد [در شبکه‌ی بیت کوین] همیشه با انتظار برای انقضای قفل زمانی آن همراه خواهد بود. ولی اگر دو طرف کانال بر سر تراز نهایی توافق داشته باشند و هر دو تراکنش تعهدی که این تراز را به واقعیت تبدیل می‌کند، در اختیار داشته باشند، می‌توانند یک تراکنش تسویه بدون قفل زمانی بسازند. در بستن دو-طرفه‌ی کانال، هر دو طرف آخرین تراکنش تعهد را گرفته و بر مبنای آن یک تراکنش تسویه کاملاً مشابه می‌سازند، تراکنشی که دیگر قفل زمانی ندارند. آنها هر دو این تراکنش را با خیال راحت امضا می‌کنند، چون می‌دانند تقلبی در کار نیست و چیز بیشتری عاید آنها نمی‌شود. با امضا و انتشار دو-طرفه‌ی این تراکنش تسویه، دو طرف می‌توانند کانال را بسته و بلافاصله تراز خود را وصول کنند. در بدترین حالت، یکی از طرفین کانال از همکاری خودداری کرده و طرف دیگر را مجبور می‌کند تا کانال را به صورت یک-طرفه و بر مبنای آخرین تراکنش تعهد ببندد. ولی در این وضعیت آنها باید برای رسیدن به پول خود تا منقضی شدن قفل زمانی این تراکنش انتظار بکشند.

تعهد قابل فسخ نامتقارن

روش بهتر برای تغییر دادن حالت (نامعتبرسازی) تراکنش تعهد قبلی این است که آن را به طور صریح فسخ کنیم. با این حال، چنین کاری ساده نیست. یکی از ویژگی‌های کلیدی بیت کوین این است که وقتی یک تراکنش معتبر شد، دیگر برای همیشه معتبر می‌ماند و باطل نمی‌شود. تنها راه برای نامعتبر کردن یک تراکنش این است که قبل از استخراج آن، ورودی‌های این تراکنش را در یک تراکنش دیگر دوباره خرج کنیم. علت استفاده از قفل زمانی در کانال پرداخت ساده‌ی قسمت قبل نیز همین بود، چون با این کار می‌توان اطمینان داشت که آخرین تراکنش تعهد قبل از معتبر شدن تعهدهای پیشین قابل خرج کردن خواهد بود. با این حال، الزام به ایجاد توالی زمانی مناسب بین تراکنش‌های تعهد مشکلات و محدودیت‌هایی به وجود می‌آورد که استفاده از کانال‌های پرداخت را سخت می‌کند.

هر چند یک تراکنش را نمی‌توان فسخ (باطل) کرد، می‌توان آن را طوری ساخت که مطلوب (خواستنی) نباشد. روش کار این است که به هر دو طرف کانال یک کلید فسخ (revocation key) بدهیم که بتوانند در صورت اقدام برای تقلب، طرف مقابل را تنبیه کنند. ساز و کار فسخ تراکنش‌های تعهد قبلی برای اولین بار در شبکه‌ی آذرخش پیشنهاد شد. برای تشریح ساز و کار کلیدهای فسخ، یک کانال پرداخت پیچیده‌تر برای مبادله بین دو فرد به نام‌های هیتش و ایرن می‌سازیم. هیتش و ایرن صاحب دو صرافی بیت کوین در هند و ایالات متحده هستند که به نیابت از مشتریان خود در این دو کشور، مبادله‌ی بیت کوین انجام می‌دهند. در حال حاضر، هیتش و ایرن برای مبادلات خود به بلاک چین بیت کوین متکی هستند، ولی این به معنای پرداخت کارمزد قابل توجه به ازای هر تراکنش و همچنین انتظار طولانی (برای استخراج و تأیید چندین بلاک) است.

هیتش و ایرن برای ایجاد یک کانال پرداخت به توافق رسیده، و برای راه‌اندازی آن هر کدام ۵ بیت کوین به عنوان تأمین سرمایه پرداخت می‌کنند؛ به عبارت دیگر، تراز اولیه‌ی کانال ۵ بیت کوین برای هیتش و ۵ بیت کوین برای ایرن است. درست مانند مثال قسمت قبل، این تراکنش تأمین سرمایه حالت کانال را به صورت چندامضایی «۲-از-۲» قفل می‌کند. این تراکنش می‌تواند یک یا چند ورودی (با مجموع ۵ بیت کوین) برای هیتش، و یک یا چند ورودی (با مجموع ۵ بیت کوین) برای ایرن داشته باشد. البته برای پوشش دادن کارمزد تراکنش‌ها، مبلغ واقعی این تراکنش باید کمی بیشتر از ظرفیت کانال (۱۰ بیت کوین) باشد. این تراکنش همچنین یک خروجی دارد که مبلغ ۱۰ بیت کوین را به یک آدرس چندامضایی «۲-از-۲» که توسط هیتش و ایرن کنترل می‌شود، قفل می‌کند. در صورتی که مجموع ورودی‌های هیتش و ایرن از ۵ بیت کوین توافقی آنها به اضافه‌ی کارمزد تراکنش‌ها بیشتر باشد، ممکن است در تراکنش تأمین سرمایه یک یا دو تراکنش تنمه (به آدرس هیتش و ایرن) نیز وجود داشته باشد. این یک تراکنش واحد است با ورودی‌هایی که دو طرف کانال آنها را تأمین و امضا کرده‌اند. تراکنش تأمین سرمایه باید با همکاری هیتش و ایرن ساخته شده و قبل از ارسال به شبکه‌ی بیت کوین توسط هر دو امضا شود.

در گام بعد، هیتش و ایرن به جای یک تراکنش تعهد که توسط هر دو طرف امضا شده باشد، دو تراکنش تعهد مختلف می‌سازند که نامتقارن (asymmetric) هستند. تراکنش تعهد هیتش دو خروجی دارد: خروجی اول پرداخت بی‌تاخیر ۵ بیت کوین به ایرن، و خروجی دوم پرداختی به مبلغ ۵ بیت کوین به خودش که فقط بعد از یک قفل زمانی به طول ۱۰۰۰ بلاک می‌تواند وصول شود؛ این تراکنش توسط ایرن امضا می‌شود. خروجی‌های این تراکنش شبیه زیر هستند:

Input: 2-of-2 funding output, signed by Irene

Output 0 <5 bitcoin>:

<Irene's Public Key> CHECKSIG

Output 1:

<1000 blocks>

CHECKSEQUENCEVERIFY

DROP

<Hitesh's Public Key> CHECKSIG

تراکنش تعهد ایرن نیز دو خروجی دارد: خروجی اول پرداخت بی تأخیر ۵ بیت کوین به هیتش، و خروجی دوم پرداختی به مبلغ ۵ بیت کوین به خودش که فقط بعد از یک قفل زمانی به طول ۱۰۰۰ بلاک قابل وصول خواهد بود. این تراکنش توسط هیتش امضای شود. خروجی های این تراکنش چنین هستند:

Input: 2-of-2 funding output, signed by Hitesh

Output 0 <5 bitcoin>:

<Hitesh's Public Key> CHECKSIG

Output 1:

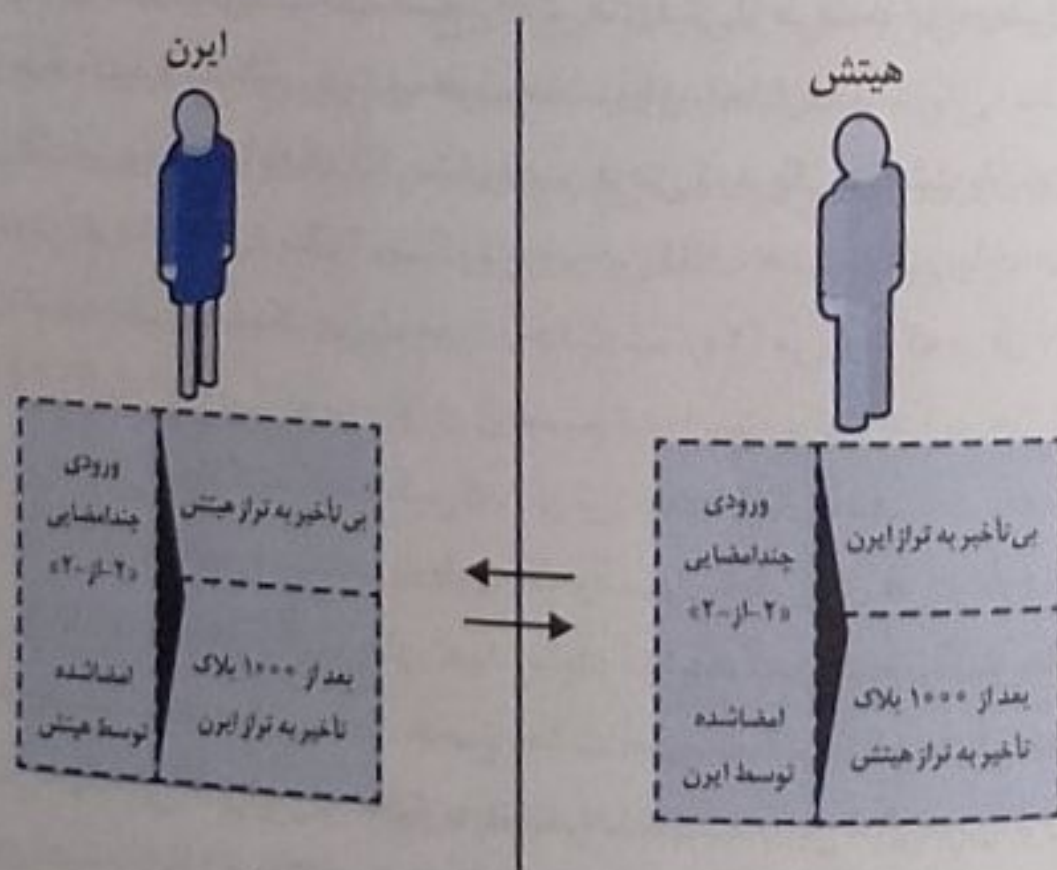
<1000 blocks>

CHECKSEQUENCEVERIFY

DROP

<Irene's Public Key> CHECKSIG

بدین ترتیب، هر دو طرف یک تراکنش تعهد در اختیار دارند که خروجی تأمین سرمایه‌ی «۲-از-۲» را خرج می‌کند، و این ورودی توسط طرف مقابل امضا شده است. دارنده‌ی این تراکنش می‌تواند در هر زمان که اراده کند، این تراکنش را امضا کرده و (با کامل کردن چند امضایی «۲-از-۲») آن را در شبکه‌ی بیت کوین منتشر کند. با این حال، این اقدام با فاصله‌ای کوتاه از سوی طرف مقابل تلافی خواهد شد. با اعمال یک تأخیر کوتاه (در اینجا، حدوداً یک هفته) بر زمان وصول یکی از خروجی‌ها، هر یک از طرفین در موقعیت منفی برای اقدام یک‌طرفه (و نقد کردن این تراکنش) قرار خواهد گرفت، ولی این تأخیر به‌نهایی برای وادار کردن آنها به رفتار منصفانه کافی نیست. این تراکنش‌های تعهد نامتقارن در شکل ۸-۱۲ نشان داده شده‌اند. اکنون آماده‌ایم تا آخرین عنصر این طرح را معرفی کنیم: یک کلید فسخ که به طرف مغبون اجازه می‌دهد با وصول کل تراز کانال طرف متقلب را تنبیه کند. همان‌طور که دیدیم، هر یک از این دو تراکنش تعهد یک خروجی «تأخیردار» دارند. اسکرپیت وصول این خروجی باید به دارنده اجازه دهد آن را بعد از ۱۰۰۰ بلاک وصول کرده، یا طرف مقابل با داشتن یک کلید فسخ آن را وصول کند. بنابراین، وقتی هیتش یک تراکنش تعهد تولید می‌کند و آن را برای امضا به ایرن می‌فرستد، اسکرپیت وصول آن را طوری تنظیم می‌کند که علاوه بر قابل وصول بودن بعد از ۱۰۰۰ بلاک، با ارائه‌ی یک کلید فسخ هم بلافاصله قابل وصول



شکل ۸-۱۲ دو تراکنش تعهد نامتقارن با پرداخت تأخیردار به طرف دارنده‌ی آن تراکنش.

باشد. هتیش کلید فسخ این تراکنش را هم تولید کرده و نزد خود محفوظ نگه می دارد؛ او فقط زمانی این کلید را فاش می کند که بخواهد حالت کانال را تغییر داده و این تعهد را فسخ کند. اسکرپت خروجی دوم تراکنش تعهد هتیش چنین خواهد بود:

```
Output 0 <5 bitcoin>:
  <Irene's Public Key> CHECKSIG
```

```
Output 1 <5 bitcoin>:
```

```
IF
  # Revocation penalty output
  <Revocation Public Key>
ELSE
  <1000 blocks>
  CHECKSEQUENCEVERIFY
  DROP
  <Hitesh's Public Key>
ENDIF
CHECKSIG
```

ایرن می تواند با خیال راحت این تراکنش را امضا کند، چون انتشار آن در شبکه بیت کوین باعث می شود پولی که طلب دارد، بلافاصله به وی مسترد شود. هتیش این تراکنش را نزد خود نگه می دارد، ولی می داند انتشار یکجانبه‌ی این خروجی باعث بسته شدن کانال می شود و برای گرفتن پول خود باید ۱۰۰۰ بلاک (حدود یک هفته) صبر کند. وقتی کانال به حالت بعدی می رود، هتیش باید قبل از موافقت ایرن با امضای تراکنش تعهد بعدی، آن را فسخ کند. برای این منظور، او باید کلید فسخ خود را به ایرن بفرستد. همین که ایرن کلید فسخ این تراکنش تعهد را به دست آورد، می تواند با خیال راحت تراکنش تعهد بعدی را امضا کند. او می داند که اگر هتیش بخواهد (با منتشر کردن تعهد قبلی) دست به تقلب بزند، می تواند با این کلید فسخ خروجی تأخیردار هتیش را وصول کند. به عبارت دیگر، اگر هتیش تقلب کند، ایرن هر دو خروجی را تصاحب خواهد کرد.

پروتکل فسخ دو-طرفه است، یعنی هر بار که حالت کانال عوض می شود و دو طرف تراکنش های تعهد جدید رد و بدل می کنند، کلیدهای فسخ تراکنش های تعهد قبلی را مبادله کرده و تراکنش های طرف مقابل را نیز امضا می کنند. با قبول حالت جدید از سوی دو طرف کانال، و دریافت کلید فسخ تراکنش های قبلی از طرف مقابل، وصول این تراکنش ها غیر ممکن خواهد شد چون هر دو طرف کلید فسخ لازم برای تنبیه طرف متقلب را در اختیار دارند.

اجازه دهید طرز کار این پروتکل را با یک مثال نشان دهیم. فرض کنید یکی از مشتریان ایرن می خواهد ۲ بیت کوین برای یکی از مشتریان هتیش بفرستد. برای ارسال ۲ بیت کوین روی این کانال، هتیش و ایرن باید حالت کانال را عوض کنند تا با تراز جدید همخوانی داشته باشد. آنها به یک حالت جدید (حالت شماره ۲) می روند که در آن ۱۰ بیت کوین تراز کل کانال به صورت ۷ بیت کوین برای هتیش و ۳ بیت کوین برای ایرن تقسیم شده است. برای رفتن به حالت جدید، هتیش و ایرن هر دو باید تراکنش های تعهد جدیدی ایجاد کنند که منعکس کننده ی تراز جدید کانال باشد.

مانند قبل، این تراکنش ها نامتقارن هستند، به طوری که وصول آنها را برای هر دو طرف نامطلوب می کند. نکته ی کلیدی این است که قبل از مبادله و امضای تراکنش های تعهد جدید، آنها باید کلید فسخ تراکنش های تعهد قبلی را به یکدیگر بفرستند تا این تراکنش ها نامعتبر شوند. در این حالت خاص، حالت جدید کانال با منافع هتیش همسو است، بنابراین او هیچ دلیلی برای انتشار تراکنش تعهد قبلی که از ایرن در اختیار دارد، نخواهد داشت. با این حال، برای ایرن حالت ۲ نسبت به حالت ۱ با ضرر همراه است، زیرا در حالت ۱ تراز بالاتری دارد. وقتی ایرن کلید فسخ تراکنش تعهد قبلی خود (حالت ۱) را به هتیش می دهد، در حقیقت توانایی نفع بردن از حالت قبلی کانال را از خود سلب می کند، چون هتیش با در دست داشتن کلید فسخ

تراکنش تعهد قبلی ایرن می‌تواند بلافاصله هر دو خروجی آن را وصول کند. به عبارت دیگر، اگر ایرن اقدام به انتشار حالت قبلی کانال کند، هتیش می‌تواند با وصول هر دو خروجی حق خود را پس بگیرد.

اما نکته‌ی مهم این است که فسخ به طور خودکار اتفاق نمی‌افتد. هر چند هتیش از توانایی تنبیه ایرن برای اقدام به تقلب برخوردار است، ولی برای این کار باید به دقت حواسش به بلاک چین باشد تا هر گونه نشانه‌ی تقلب از سوی ایرن را کشف کند. اگر هتیش متوجه انتشار تراکنش تعهد قبلی خود توسط ایرن [در شبکه‌ی بیت کوین] شود، به مدت ۱۰۰۰ بلاک وقت دارد تا دست به اقدام متقابل بزند و با استفاده از کلید فسخ خود تقلب ایرن را خنثی کرده و با تصاحب تراز کل کانال (۱۰ بیت کوین) او را تنبیه کند. تعهد قابل فسخ نامتقارن در ترکیب با قفل زمانی نسبی (CSV) کارایی بسیار بهتری در پیاده‌سازی کانال‌های پرداخت دارد، و یکی از ابداعات بسیار مهم در فناوری ارزهای رمزبنیان محسوب می‌شود. به کمک این ساختار، یک کانال پرداخت می‌تواند به طور نامحدود باز بماند و میلیون‌ها تراکنش تعهد بینابینی در آن انجام شود. در پیاده‌سازی اولیه‌ی شبکه‌ی آذرخش، حالت تعهد با یک اندیس ۴۸-بیتی شناسایی می‌شود، که به یک کانال واحد اجازه می‌دهد تا بیش از ۲۸۱ هزار تریلیون ($2/8 \times 10^{18}$) تراکنش (حالت) بینابینی منحصر به فرد داشته باشد.

قرارداد قفل زمانی درهم (HTLC)

کانال‌های پرداخت را می‌توان با استفاده از نوع خاصی از قرارداد هوشمند باز هم توسعه داد، قراردادی که به طرفین اجازه می‌دهد تا به کمک یک «کلید سرّی قابل وصول دارای مهلت انقضا» پرداخت انجام دهند. این ویژگی که به قرارداد قفل زمانی درهم (Hash Time Lock Contract) یا به اختصار HTLC معروف است، در هر دو نوع کانال پرداخت دو-طرفه و هدایت‌شده کاربرد دارد.

اجازه دهید ابتدا مفهوم «درهم» در HTLC را توضیح دهیم. برای ایجاد یک HTLC، ابتدا طرف گیرنده‌ی این پرداخت یک کلید سرّی، R، تولید کرده و سپس درهم آن، H، را محاسبه می‌کند:

$$H = \text{Hash}(R)$$

از این H می‌توان در یک اسکرپت قفل‌کننده‌ی خروجی استفاده کرد، و هر کس این کلید را در اختیار داشته باشد، می‌تواند آن خروجی را وصول کند. کلید سرّی R، که به آن پیش‌تصویر گفته می‌شود، فقط برای محاسبه‌ی H کاربرد دارد و بعد از آن دیگر نقشی ایفا نمی‌کند.

بخش دوم HTLC یک «قفل زمانی» است. اگر هیچ کس نتوانست با ارائه‌ی درهم سرّی H در زمان مقرر این پرداخت را وصول کند، پرداخت‌کننده می‌تواند بعد از انقضای مهلت «پول» خود را پس بگیرد. برای این منظور از یک قفل زمانی مطلق (CHECKLOCKTIMEVERIFY) استفاده می‌شود. اسکرپت پیاده‌سازی یک HTLC می‌تواند به شکل زیر باشد:

```
IF
  # Payment if you have the secret R
  HASH160 <H> EQUALVERIFY
ELSE
  # Refund after timeout.
  <locktime> CHECKLOCKTIMEVERIFY DROP
  <Payee Pubic Key> CHECKSIG
ENDIF
```

اگر شما (یا هر کس دیگر) کلید سرّی R را در اختیار داشته باشید، می‌توانید درهم آن (H) را محاسبه کنید، و به کمک آن بخش IF اسکرپت بالا را فعال کرده و این خروجی را وصول کنید. ولی اگر این کلید سرّی ارائه نشود، بعد از سپری شدن مهلت مشخص شده، پرداخت‌کننده می‌تواند با فعال کردن بخش ELSE این اسکرپت اقدام به پس گرفتن «پول» خود کند.

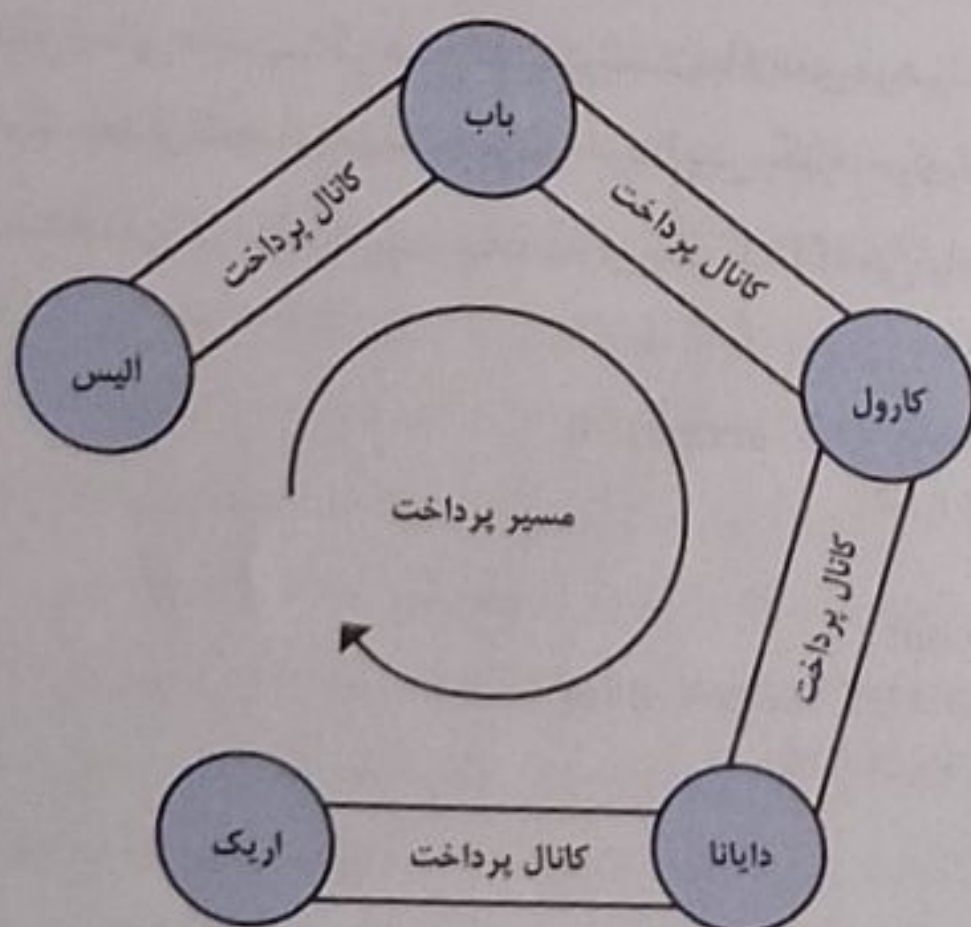
این ساده‌ترین روش پیاده‌سازی HTLC است. این نوع از HTLC توسط هر کس که کلید سری R را در اختیار داشته باشد، قابل وصول است. با تغییر دادن اسکرپت وصول می‌توان اشکال مختلفی از HTLC ایجاد کرد. برای مثال، با اضافه کردن عملگر CHECKSIG و کلید عمومی یک فرد خاص به بخش IF می‌توانید وصول آن را به همان گیرنده (که البته حتماً باید کلید سری R را نیز در اختیار داشته باشد) محدود کنید.

کانال پرداخت هدایت‌شده (شبکه‌ی آذرخش)

شبکه‌ی آذرخش (Lightning Network) یک شبکه‌ی هدایت‌شده‌ی پیشنهادی از کانال‌های پرداخت دو-طرفه با اتصال نقطه-به-نقطه است. در چنین شبکه‌ای افراد می‌توانند پرداخت‌های خود را از یک کانال به کانال دیگر هدایت کنند، بدون آن که نگران اعتمادپذیری گره‌های میانی باشند. شبکه‌ی آذرخش در سال ۲۰۱۵ به عنوان شبکه‌ای بنا شده بر پایه‌ی مفهوم کانال پرداخت معرفی شد و تاکنون پیشنهادهای زیادی برای بهبود آن صورت گرفته است. حداقل پنج طراحی مختلف برای پیاده‌سازی شبکه‌ی آذرخش پیشنهاد شده است، و گروه‌های زیادی آن را به صورت نمونه‌ی اولیه پیاده‌سازی کرده‌اند. البته همه‌ی این نمونه‌ها به خاطر استفاده از segwit (شاهد تفکیک‌شده) فقط روی testnet (شبکه‌ی آزمایشی بیت‌کوین) می‌توانند اجرا شوند و هنوز روی بلاک چین اصلی بیت‌کوین (mainnet) فعال نشده‌اند. شبکه‌ی آذرخش فقط یکی از روش‌های پیاده‌سازی کانال پرداخت هدایت‌شده است، و طرح‌های متعدد دیگری (مانند Teechan و Tumblebit) نیز هستند که تحقق همین هدف را دنبال می‌کنند.

یک نمونه‌ی ساده از شبکه‌ی آذرخش

اجازه دهید ببینیم شبکه‌ی آذرخش چگونه کار می‌کند. در این مثال پنج شرکت‌کننده داریم: آلیس، باب، کارول، دایانا و اریک. این پنج نفر چهار کانال پرداخت دو-طرفه بین خود ایجاد کرده‌اند: آلیس یک کانال پرداخت با باب دارد؛ باب با یک کانال پرداخت به کارول متصل است؛ کارول به دایانا؛ و دایانا به اریک. برای سادگی فرض می‌کنیم همه‌ی این افراد کانال‌های پرداخت خود را با ۲ بیت‌کوین شروع کرده‌اند، یعنی ظرفیت کل هر کانال ۴ بیت‌کوین است. همان طور که در شکل ۹-۱۲ می‌بینید، این چهار کانال پرداخت دو-طرفه در اتصال با یکدیگر یک شبکه‌ی آذرخش می‌سازند، که آلیس را به اریک متصل می‌کند.



شکل ۹-۱۲ از اتصال چند کانال پرداخت دو-طرفه یک شبکه‌ی آذرخش شکل می‌گیرد که می‌تواند پرداخت‌های آلیس را به اریک برساند (هدایت کند).

آلیس می‌خواهد ۱ بیت کوین به اریک پردازد، ولی هیچ کانال پرداخت مستقیمی به او ندارد. ایجاد یک کانال پرداخت جدید بین آلیس و اریک مستلزم یک تراکنش تأمین سرمایه است که باید در بلاک چین بیت کوین ثبت شود. آلیس میل ندارد با گذاشتن سرمایه‌ی بیشتر یک کانال پرداخت جدید باز کند. اما آیا راهی وجود ندارد که آلیس بتواند پرداخت خود به اریک را به طور غیرمستقیم انجام دهد؟ شکل ۱۲-۱۰ فرآیند گام-به-گام هدایت (مسیریابی) پرداخت آلیس به اریک از طریق تعدادی تراکنش تعهد HTLC روی کانال‌های پرداخت بین این دو را نشان می‌دهد.

در اینجا فرض ما بر این است که همه‌ی این پنج نفر شبکه‌ی آذرخش را در سیستم خود پیاده‌سازی کرده‌اند و می‌توانند مسیرهای مناسب بین کانال‌های پرداخت را شناسایی کنند. گره شبکه‌ی آذرخش (LN) آلیس پرداخت‌های انجام‌شده به باب را ثبت می‌کند، و در ضمن می‌تواند از طریق اینترنت به گره LN اریک نیز متصل شود [توجه کنید که این فقط یک اتصال اینترنتی ساده است، نه یک کانال پرداخت]. گره LN اریک یک عدد تصادفی تولید کرده و آن را به عنوان کلید سری R به کار می‌گیرد. اریک کلید سری R را نزد خود نگه می‌دارد و آن را نزد هیچ کس افشا نمی‌کند؛ به جای آن، درهم H این کلید را محاسبه کرده و آن را به گره آلیس می‌فرستد (گام ۱ در شکل ۱۲-۱۰). پس از دریافت این درهم، آلیس یک مسیر بین گره LN خودش و یک گره LN اریک می‌سازد. در قسمت بعد درباره‌ی الگوریتم مسیریابی به طور مفصل توضیح خواهیم داد، ولی در حال حاضر فقط فرض می‌کنیم گره LN آلیس قادر به یافتن یک مسیر کارآمد بین خودش و اریک هست.

گره آلیس سپس یک تراکنش تعهد HTLC به مبلغ ۱۰۰۰۳ بیت کوین، قابل پرداخت به درهم H، با مهلت بازپس‌گیری ۱۰ بلاک (بلاک فعلی + ۱۰) می‌سازد (گام ۲ در شکل ۱۲-۱۰)؛ ۱۰۰۰۳ بیت کوین اضافی برای جبران مخارج و پاداش به گره‌های بینابینی برای مشارکت در این مسیر پرداخت است. آلیس با کسر کردن ۱۰۰۰۳ بیت کوین از تراز کانال پرداخت خود با باب در یک تراکنش HTLC، این تراکنش را به وی پیشنهاد می‌کند. این تراکنش HTLC چنین می‌گوید: «آلیس ۱۰۰۰۳ بیت کوین از تراز کانال پرداخت خود را به باب تعهد می‌کند، اگر باب کلیدی سری R را در اختیار داشته باشد؛ در غیر این صورت آن را بعد از گذشت ۱۰ بلاک پس می‌گیرد.» اکنون تراز کانال پرداخت بین آلیس و باب به وسیله‌ی تراکنش‌های تعهد با سه خروجی بیان می‌شود: ۲ بیت کوین به تراز باب، ۹۹۷ بیت کوین به تراز آلیس، و ۱۰۰۰۳ بیت کوین تعهد آلیس در تراکنش HTLC. توجه کنید که تراز کل آلیس به مقدار تعهدشده در این تراکنش HTLC کاهش یافته است.

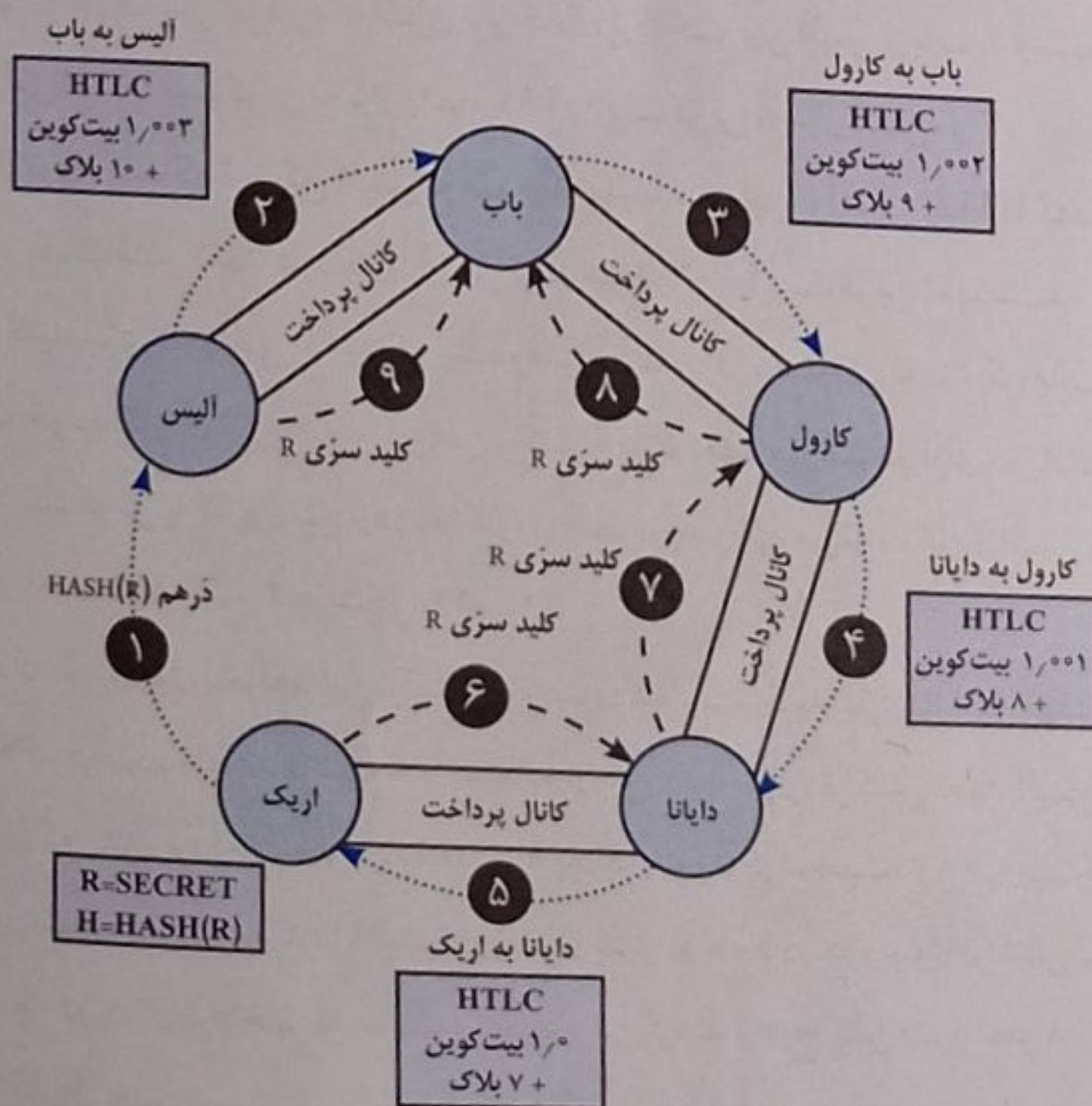
اکنون باب یک تراکنش تعهد دارد که با آن می‌تواند مبلغ ۱۰۰۰۳ بیت کوین تعهدشده توسط آلیس در این تراکنش را با ارائه‌ی کلید سری R در طی ۱۰ بلاک آینده وصول کند. با دریافت این تعهد، گره باب یک تراکنش HTLC روی کانال پرداخت خود با کارول می‌سازد. در این تراکنش، باب تعهد می‌کند در ازای دریافت درهم H در مهلت ۹ بلاک، مبلغ ۱۰۰۰۲ بیت کوین به کارول پردازد؛ اما کارول فقط در صورتی می‌تواند این تراکنش را وصول کند که کلید سری R را در اختیار داشته باشد (گام ۳ در شکل ۱۲-۱۰).

باب می‌داند که اگر کارول بخواهد این تراکنش را وصول کند، باید کلید سری R را در مهلت ۹ بلاک آینده به وی بدهد، و او هم ۹ بلاک فرصت خواهد داشت تا با استفاده از این کلید سری تراکنش تعهد آلیس را به نفع خود وصول کند. باب به خاطر آن که کانال پرداخت خود را به مدت ۹ بلاک متعهد کرده است، ۱۰۰۰۱ بیت کوین به عنوان کارمزد برمی‌دارد. اگر کارول نتواند تراکنش HTLC باب را وصول کند، او هم قادر به وصول تراکنش HTLC آلیس نخواهد بود؛ در این صورت هر دو کانال پرداخت به حالت سابق برمی‌گردند و هیچ کس ضرر نخواهد کرد. اکنون تراز کانال پرداخت بین باب و کارول چنین است: ۲ بیت کوین به تراز کارول، ۹۹۸ بیت کوین به تراز باب، و ۱۰۰۰۲ بیت کوین تعهد باب در تراکنش HTLC.

اکنون کارول یک تراکنش تعهد در اختیار دارد که با آن می تواند مبلغ $1/002$ بیت کوین تعهد شده توسط باب را با ارائه کلید سری R در طی ۹ بلاک آینده وصول کند. پس از دریافت این تعهد، گره کارول یک تراکنش HTLC روی کانال پرداخت خود با دایانا می سازد. کارول در این تراکنش تعهد می کند در ازای دریافت درهم H در مهلت ۸ بلاک، مبلغ $1/001$ بیت کوین به دایانا بپردازد؛ اما دایانا در صورتی قادر به وصول این تراکنش است که کلید سری R را داشته باشد (گام ۴ در شکل ۱۰-۱۲). از دید کارول، اگر دایانا بخواهد این تراکنش را وصول کند، باید کلید سری R را در طی ۸ بلاک آینده به وی بدهد، و او هم ۸ بلاک فرصت دارد تا با این کلید سری تراکنش تعهد باب را به نفع خود وصول کند. در این میان، کارول به دلیل متعهد کردن کانال پرداخت خود به مدت ۸ بلاک، $1/001$ بیت کوین به عنوان کارمزد برمی دارد. اکنون تراز کانال پرداخت بین کارول و دایانا چنین است: ۲ بیت کوین به تراز دایانا، $1/999$ بیت کوین به تراز کارول، و $1/001$ بیت کوین تعهد کارول در تراکنش HTLC.

سرانجام، گره دایانا یک تراکنش HTLC روی کانال پرداخت خود با اریک می سازد و در این تراکنش تعهد می کند که در ازای دریافت درهم H در طی ۷ بلاک آینده، مبلغ ۱ بیت کوین به اریک بپردازد (گام ۵ در شکل ۱۰-۱۲). تراز کانال پرداخت بین دایانا و اریک اکنون چنین است: ۲ بیت کوین به تراز اریک، ۱ بیت کوین به تراز دایانا، و ۱ بیت کوین تعهد دایانا در تراکنش HTLC.

ولی اینجا آخر خط است، چون اریک کلید سری R را در اختیار دارد، و می تواند مبلغ تعهد شده در تراکنش HTLC دایانا را وصول کند. پس، او کلید سری R را به دایانا می فرستد و با وصول ۱ بیت کوین، آن را به تراز کانال پرداخت خود اضافه می کند (گام ۶ در شکل ۱۰-۱۲). اکنون تراز کانال پرداخت بین دایانا و اریک به این صورت در می آید: ۱ بیت کوین به تراز دایانا، و ۳ بیت کوین به تراز اریک.



شکل ۱۰-۱۲ هدایت (مسیریابی) گام-به-گام یک پرداخت در شبکه‌ی آذرخش.

بعد از این که دایانا کلید سرّی R را از اریک دریافت کرد، می‌تواند تراکنش HTLC کارول را وصول کند. پس، دایانا کلید R را به کارول می‌فرستد و با وصول این تراکنش، ۱۰۰۱ بیت‌کوین به تراز کانال پرداخت خود اضافه می‌کند (گام ۷ در شکل ۱۲-۱۰). با این کار، تراز کانال پرداخت بین کارول و دایانا چنین خواهد بود: ۰٫۹۹۹ بیت‌کوین به تراز کارول، و ۳٫۰۰۱ بیت‌کوین به تراز دایانا. همان طور که می‌بینید، دایانا کارمزد ۰٫۰۰۱ بیت‌کوینی مشارکت در این مسیر پرداخت را دریافت کرده است.

با ادامه یافتن همین مسیر به سمت عقب، کلید سرّی R به هر یک از شرکت‌کنندگان در این مسیر اجازه می‌دهد تا تراکنش‌های HTLC معوقه‌ی خود را وصول کنند. کارول با وصول ۱۰۰۲ بیت‌کوین از باب، تراز کانال پرداخت خود را به «۰٫۹۹۸ بیت‌کوین به تراز باب، و ۳٫۰۰۲ بیت‌کوین به تراز کارول» می‌رساند (گام ۸ در شکل ۱۲-۱۰). و سرانجام، باب با وصول تراکنش HTLC آلیس، تراز نهایی این کانال را به «۰٫۹۹۷ بیت‌کوین به تراز آلیس، و ۳٫۰۰۳ بیت‌کوین به تراز باب» تغییر می‌دهد (گام ۹ در شکل ۱۲-۱۰).

در نهایت، آلیس موفق شده است بدون باز کردن یک کانال پرداخت مستقیم با اریک، ۱ بیت‌کوین به او پرداخت کند. در این میان حتی نیازی نیست گره‌های بینایی مسیر پرداخت به یکدیگر اعتماد داشته باشند. از طرف دیگر، پاداش کوچکی که در ازای یک تعهد کوتاه‌مدت (حداکثر چند ده بلاک) و بی‌خطر در این مسیر پرداخت به دست می‌آورند، انگیزه‌ی خوبی برای مشارکت آنها محسوب می‌شود.

انتقال و مسیریابی در شبکه‌ی آذرخش

تمامی ارتباطات بین گره‌های LN (شبکه‌ی آذرخش) به صورت نقطه-به-نقطه رمزگذاری می‌شوند. علاوه بر آن، گره‌های LN دارای یک کلید عمومی بلندمدت نیز هستند که در فرآیند احراز هویت بین یکدیگر از آن به عنوان شناسه استفاده می‌کنند.

وقتی یک گره LN می‌خواهد پرداختی به گره دیگر انجام دهد، ابتدا باید با متصل کردن کانال‌های پرداخت بینایی که ظرفیت کانال کافی دارند، یک مسیر [پرداخت] در این شبکه بسازد. گره‌های LN به طور منظم اطلاعات مسیریابی، شامل کانال‌های باز خود، ظرفیت هر کانال، و کارمزد مشارکت در مسیریابی پرداخت‌ها، را در شبکه منتشر می‌کنند. این اطلاعات مسیریابی را می‌توان به روش‌های گوناگون با دیگران به اشتراک گذاشت، و با پیشرفت فناوری شبکه‌ی آذرخش احتمال ظهور پروتکل‌های مسیریابی مختلف بیشتر نیز می‌شود. برخی پیاده‌سازی‌های شبکه‌ی آذرخش از پروتکل IRC به عنوان ساز و کار انتشار اطلاعات مسیریابی به گره‌های دیگر استفاده می‌کنند؛ برخی دیگر بر اساس مدل مسیریابی P2P ساخته شده‌اند که (مانند روش انتشار تراکنش‌ها در شبکه‌ی بیت‌کوین) اطلاعات مسیریابی را به گره‌هایی که بلاواسطه به آنها متصل هستند، سرازیر می‌کنند. پیشنهادهایی برای پیاده‌سازی مدل‌های مسیریابی ترکیبی نیز ارائه شده‌اند (از جمله Flare: مشعل)، که در آنها گره‌های شبکه به دو دسته‌ی گره‌های محلی (موسوم به گره همسایه) و گره‌های دوردست (موسوم به گره راهنما) تقسیم می‌شوند.

در مثال قسمت قبل، گره LN آلیس از یکی از همین پروتکل‌های شناسایی مسیر برای یافتن یک (یا چند) مسیر که گره او را به گره اریک متصل کند، استفاده کرده است. همین که آلیس موفق به شناسایی یک مسیر شد، با انتشار یکسری دستورات تو در تو و رمزگذاری شده، کانال‌های پرداخت مجاور را به یکدیگر متصل کرده و مسیر پرداخت را آماده‌سازی می‌کند.

نکته‌ی مهم این است که فقط گره آلیس این مسیر را به طور کامل می‌شناسد، و تمام گره‌هایی که در این مسیر پرداخت مشارکت دارند، فقط گره‌های مجاور خود را می‌بینند. برای مثال، از نقطه نظر کارول، این مسیر فقط یک پرداخت از باب به دایانا است؛ در واقع، کارول نمی‌داند که خود باب هم به عنوان واسطه‌ی انتقال پرداخت آلیس عمل می‌کند، یا آن که دایانا این پرداخت را به اریک هدایت خواهد کرد. این یک ویژگی کلیدی شبکه‌ی آذرخش است، چون محرمانگی پرداخت‌ها را تضمین کرده و رفتارهایی مانند جاسوسی، تحریم یا جانبداری را بسیار دشوار خواهد کرد. ولی آلیس چگونه می‌تواند بدون فاش کردن اطلاعات حساس برای گره‌های بینابینی، این مسیر پرداخت را برقرار کند؟

شبکه‌ی آذرخش از یک پروتکل مسیریابی-پیازی (onion-routed) بر اساس طرحی موسوم به Sphinx (ابولیهول) استفاده می‌کند. این پروتکل مسیریابی تضمین می‌کند که فرستنده‌ی [حواله‌ی] پرداخت بتواند یک مسیر پرداخت را در شبکه‌ی آذرخش ایجاد کند، به طوری که:

- گره‌های بینابینی بتوانند فقط آن بخش از اطلاعات مسیریابی را که مربوط به خود آنها است، رمزگشایی و اعتبارسنجی کرده و از آن برای یافتن (شناسایی) گره‌های مجاور و اتصال به آنها استفاده کنند.
- گره‌های بینابینی، غیر از گره قبل و بعد خود، هیچ چیز درباره‌ی سایر بخش‌های مسیر ندانند.
- گره‌های بینابینی نتوانند بدانند یک مسیر چقدر طول [چند گره] دارد، و خود آنها در کدام بخش از این مسیر قرار دارند.
- هر بخش از مسیر به گونه‌ای رمزگذاری می‌شود که نفوذگرانی که از سطح-شبکه به آن حمله می‌کنند، نتوانند بسته‌های مربوط به بخش‌های مختلف این مسیر را به یکدیگر مرتبط کنند.
- برخلاف تور (پروتکل مسیریابی-پیازی ناشناس‌کننده در اینترنت)، چیزی به نام «گره خروج» وجود ندارد که بتوان آن را تحت نظر گرفت، چون این پرداخت‌ها به بلاک چین بیت کوین فرستاده نمی‌شوند، و فقط به نظر می‌آیند که چند گره مشغول به روز کردن تراز حساب‌های فیمابین خود هستند.

با استفاده از این پروتکل مسیریابی-پیازی، آلیس هر بخش از مسیر شناسایی شده را، با شروع از انتهای مسیر و برگشت به ابتدای آن، در یک لایه‌ی رمزگذاری می‌پیچد. به عبارت دیگر، آلیس ابتدا پیامی را که باید به دست اریک برسد، با استفاده از کلید عمومی اریک رمزگذاری می‌کند. سپس، این پیام رمزگذاری شده را در داخل پیامی که با کلید عمومی دایانا رمزگذاری کرده، قرار می‌دهد و مشخص می‌کند که گیرنده‌ی بعدی آن اریک است. پس از آن، پیام [رمزگذاری شده‌ی] دایانا را در داخل پیامی که با کلید عمومی کارول رمزگذاری شده، می‌گذارد و می‌گوید که این پیام باید به دست دایانا برسد. در نهایت، پیام کارول را در پیامی که با کلید عمومی باب رمزگذاری کرده، قرار می‌دهد و مشخص می‌کند که گیرنده‌ی بعدی آن کارول است. همان طور که می‌بینید، پیامی که آلیس ساخته، چهار لایه دارد که مانند لایه‌های پیاز روی یکدیگر قرار گرفته‌اند. آلیس این پیام را به باب می‌فرستد، که فقط می‌تواند بیرونی‌ترین لایه را رمزگشایی و باز کند. باب بعد از باز کردن این لایه، پیامی به آدرس کارول می‌بیند که می‌تواند آن را به کارول بفرستد [هدایت کند]، ولی خودش قادر به رمزگشایی آن نیست. با دنبال کردن این مسیر، می‌توان دید که پیام آلیس مرحله به مرحله رمزگشایی و هدایت می‌شود تا سرانجام به اریک برسد. گره‌های بینابینی (باب، کارول، دایانا) فقط گره‌های قبل و بعد از خود را می‌شناسند و نمی‌دانند مقصد نهایی این پیام کیست.

اطلاعاتی که در هر لایه قرار دارد، عبارتند از: تراکنش HTLC پیشنهادی به گره بعدی، مبلغ در حال ارسال، کارمزد مشارکت، و قفل زمانی CLTV انقضای این تراکنش HTLC (بر حسب بلاک). با انتشار اطلاعات مسیریابی، هر گره یک تعهد HTLC ساخته و آن را به گره بعدی می‌فرستد [هدایت می‌کند].

در اینجا شاید از خود پرسید آیا گره‌های بینابینی نمی‌توانند با دانستن طول مسیر، موقعیت نسبی خود در آن مسیر پرداخت را حدس بزنند؟ مگر نه این که آنها پیامی دریافت می‌کنند و آن را به گره بعدی می‌فرستند؛ پس، آیا کوتاه‌تر شدن این پیام با هر گرهی که جلو می‌رود، سرنخی برای فهمیدن طول مسیر و موقعیت یک گره بینابینی در آن مسیر نیست؟ برای جلوگیری از اتفاق، طول مسیرهای پرداخت همیشه به طور ثابت ۲۰ پرش (گره) در نظر گرفته شده، و جاهای خالی با اطلاعات تصادفی پر می‌شود. هر گره بینابینی فقط مقصد پرش بعدی (گره‌ی که باید این پیام را به آن بفرستد) و یک پیام رمزگذاری شده با طول ثابت می‌بیند. فقط گره آخر (گیرنده) است که پرش بعدی (گره مقصد) ندارد. برای تمام گره‌های بینابینی، طول مسیری که در ادامه باید پیموده شود، همیشه ۲۰ پرش است.

مزایای شبکه‌ی آذرخش

شبکه‌ی آذرخش یک فناوری مسیریابی لایه‌ی-دو است. این فناوری را می‌توان در هر نوع بلاک‌چین که از چند قابلیت پایه مانند تراکنش‌های چندامضایی، قفل زمانی، و قراردادهای هوشمند ساده پشتیبانی کند، پیاده‌سازی کرد. اگر شبکه‌ی آذرخش به عنوان یک لایه روی شبکه‌ی بیت‌کوین قرار گیرد، می‌تواند ظرفیت، محرمانگی، ریزدانگی و سرعت آن را به طور چشمگیری افزایش دهد، بدون آن که اصول عملکرد بدون اعتماد و بی‌واسطه‌ی این شبکه را قربانی کند:

محرمانگی

پرداخت‌های شبکه‌ی آذرخش بسیار محرمانه‌تر از پرداخت‌های روی بلاک‌چین بیت‌کوین هستند، چون به صورت علنی انجام نمی‌شوند. هر چند شرکت‌کنندگان در یک مسیر پرداخت [گره‌های بینابینی] می‌توانند عبور پرداخت‌ها را ببینند، ولی از هویت فرستنده و گیرنده اطلاع ندارند.

نظارت‌ناپذیری

شبکه‌ی آذرخش جاسوسی و تحریم بیت‌کوین را بسیار دشوار کرده، و آن را به یک ارز تقریباً نظارت‌ناپذیر تبدیل می‌کند.

سرعت

پرداخت‌های یک شبکه‌ی آذرخش (به جای چند دقیقه‌ی معمول در شبکه‌ی بیت‌کوین) در چند میلی‌ثانیه تسویه می‌شوند، چون برای تسویه‌ی تراکنش‌های HTLC نیازی به تجمع تراکنش‌ها در یک بلاک و استخراج آن نیست.

ریزدانگی

در یک شبکه‌ی آذرخش، پرداخت‌ها هیچ محدودیتی از نظر مبلغ ندارند، و حتی می‌توانند به خردی ۱ ساتوشی باشند.

ظرفیت

یک شبکه‌ی آذرخش ظرفیت سیستم بیت کوین را صدها یا حتی هزاران برابر افزایش می‌دهد. تعداد پرداخت‌ها (در ثانیه) در یک شبکه‌ی آذرخش عملاً هیچ محدودیتی ندارد، و تنها چیزی که آن را محدود می‌کند، ظرفیت و سرعت گره‌های شرکت‌کننده در این شبکه است.

عملکرد بدون اعتماد

مسیرهای پرداخت در یک شبکه‌ی آذرخش از گره‌هایی تشکیل می‌شوند که هیچ اعتمادی به یکدیگر ندارند. بنابراین، شبکه‌ی آذرخش نه تنها اصل عملکرد بدون اعتماد شبکه‌ی بیت کوین را نقض نمی‌کند، بلکه حتی پارامترهای اجرایی آن را به طور چشمگیری گسترش می‌دهد.

البته همان طور که قبلاً گفتیم، پروتکل شبکه‌ی آذرخش تنها روش برای پیاده‌سازی کانال‌های پرداخت هدایت‌شده نیست، و سیستم‌های دیگری مانند Teechan و Tumblebit نیز برای این منظور پیشنهاد شده‌اند. با این حال، شبکه‌ی آذرخش از مدتی قبل روی testnet منتشر و فعال شده است. گروه‌های نرم‌افزاری مختلفی هم در حال رقابت برای پیاده‌سازی نهایی شبکه‌ی آذرخش هستند و روی یک استاندارد همکاری مشترک (موسوم به BOLT) کار می‌کنند. بسیار محتمل است که شبکه‌ی آذرخش اولین شبکه‌ی کانال پرداخت هدایت‌شده باشد که در آینده‌ای نزدیک عملیاتی شود.

نتیجه‌گیری

در این فصل تعدادی از کاربردهای نوظهور که می‌توانند با استفاده از بلاک چین بیت کوین به عنوان بستر اعتماد ساخته شوند، معرفی کردیم. این کاربردها دورنمای بیت کوین را به چیزی فراتر از پرداخت و یا یک ابزار مالی ساده گسترش می‌دهند، و دیگر کاربردهایی که اعتماد در آنها نقش حیاتی دارد، را نیز در بر می‌گیرند. بلاک چین بیت کوین، با غیرمتمرکز کردن بنیان اعتماد، به بستری تبدیل شده است که در حوزه‌های مختلف صنعت کاربردهای انقلابی گسترده‌ای خواهد داشت.