پروژه ی نرم افزار دوره کارشناسی

گزارش شماره ۵ - اثبات انجام کار

آریا رادمهر - ۹۷۴۶۳۱۲۵

دکتر سجاد حق زاد کلیدبری

December 15, 2021

اثبات انجام کار (proof-of-work)

به یاد داشتیم که با استفاده از الگوریتم اجماع ضمنی توانستیم از پس حملات مختلفی که با آنها روبرو بودیم بربیاییم، اما کماکان یک حمله ی مهم یعنی sybil attack به راحتی ممکن است در این سیستم رخ بدهد. برای جلوگیری از این حمله ایده ی نبوغ آمیزی در میان آمد که اثبات انجام کار بود.

اثبات انجام کار بدین معناست که هر گره باید برای آنکه بتواند بلاکی را به بلاکچین اضافه کند، عملی را از قبیل به اشتراک گذاری منابع سخت افزاری و یا حل معادلات مربوط به هش انجام دهید تا ثابت کنید در ساخت بلاک جدید بی تاثیر نبوده اید. اگر این قبیل کار ها را در سیستم انجام دهید ، آن وقت میتوانید در سیستم ساخت و افزودن بلاک به صورت رندوم حضور داشته باشید. از دیگر مزیت های اثبات انجام کار این است که هر فرد به میزان کاری که انجام میدهد و به میزان منبع و انرژی که صرف میکند شانس خواهد داشت. پس همواره یک گره باید برای اینکه بتواند در سیستم شانسی برای ساخت بلاک و افزودن آن به شبکه داشته باشد و پاداش خود را دریافت کند، عمل به نسبت دشواری را انجام دهد.

فرآیند اثبات انجام کاری که باید مورد استفاده در سیستم قرار بگیرد میبایست از نظر کریپتوگرافی غیر قابل نفوذ باشد. پس استفاده از هش های کریپتوگرافیک بهترین گزینه برای این مسئله میباشند.

بیایید دوباره یک بلاک را با دقت بررسی کنیم.

با توجه به قوانین اجماع ضمنی که بالاتر گفته شد، یک بلاک (برای نمونه بلاک مرتبط با رمز ارز) شامل تراکنش های افراد و مبادلات سکه از فردی به فرد دیگر است که mempool نامیده شد. همچنین علاوه بر تراکنش های افراد در بلاک تراکنشی وجود دارد که در آن چند سکه (که قبلا از آن با عنوان پاداش ساخت بلاک یاد کردیم) به صاحب و سازنده ی بلاک منتقل خواهد شد.

بلاک زیر تا به اینجا نمایانگر خصوصیات گفته شده میباشد.

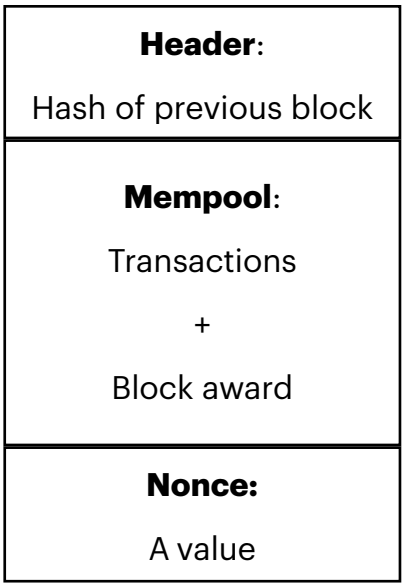
Header:
Hash of previous block
Mempool:
Transactions
+
Block award

سیستم اثبات انجام کار اینگونه است که از گره ها توقع دارد تا تمامی اطلاعات یک بلاک را که در بالا گفته شد را هش کرده به نحوی که مقدار آن هش از یک عدد معین (که آن را امگا - Ω مینامیم) همواره کمتر باشد. در این صورت آن بلاک قابل قبول در سیستم است و میتواند به عنوان بلاکی جدید به بلاکچین اضافه شود.

شرط پذیرش:
هش بدست آمده از محتویات بلاک < عدد ثابت Ω

اما میدانیم که اطلاعات داخل بلاک همواره ثابت میباشد و برای آن تنها یک هش محاسبه خواهد شد که منحصر به فرد میباشد و ممکن است که میزان هشی که از بلاک بدست می آید حتی به عدد تعیین شده نزدیک هم نباشد.
برای حل این مشکل میبایست بلاک ها علاوه بر ویژگی های header و همچنین mempool ، یک متغیر با عنوان nonce در خود داشته باشند.

متغیر nonce بدین گونه عمل میکند که به ازای هر مقدار از آن هش بلاک محاسبه میشود تا بررسی شود که آیا هش بلاک با مقدار nonce مشخص شده، از میزان عدد تعیین شده کمتر شده است یا خیر.



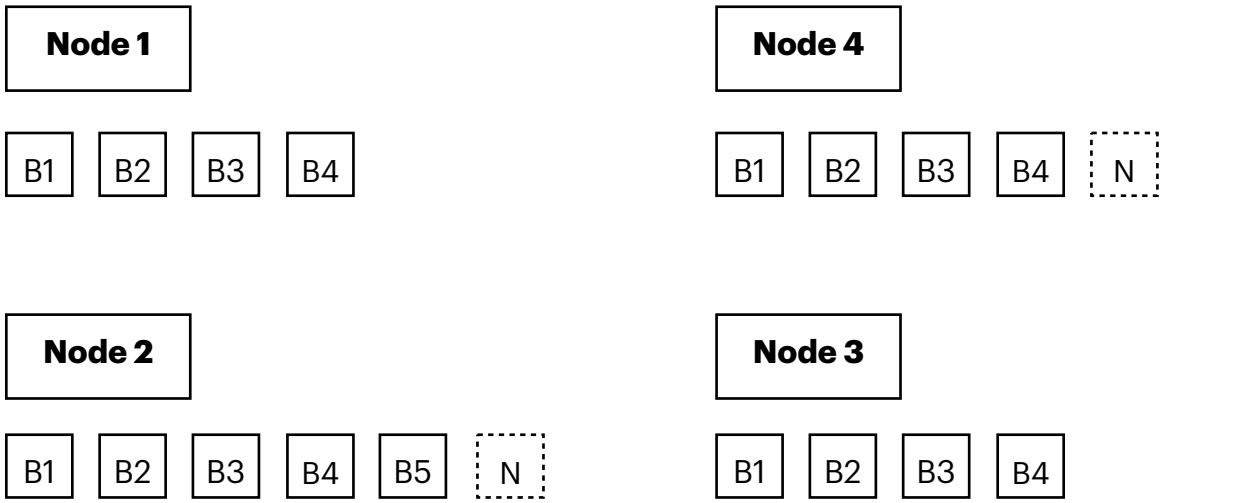
برای مثال در ابتدا متغیر nonce مقدار ۱ را در خود جای میدهد. سپس هش بلاک که اینبار علاوه بر header و mempool ، شامل nonce هم میباشد محاسبه خواهد شد، اگر شرط پذیرش بلاک برقرار نشد مقدار nonce برابر ۲ میشود تا بار دیگر هش محاسبه شود. این روند آنقدر ادامه پیدا میکند تا بالاخره مقدار nonce ی پیدا شود که بتواند هش تولیدی بلاک را در محدوده ی شرط پذیرش بلاک قرار دهد.

این نکته حائز اهمیت است که سیستم اثبات انجام کار puzzle friendly است، یعنی با این که پیدا کردن یک بلاک با شرایط گفته شده کار سخت و زمانبری میباشد، اما به محض پیدا شدن بلاکِ دارای شرط پذیرش، همه ی گره های دیگر میتوانند به یکباره هش آن بلاک را محاسبه کنند و بررسی کنند که آیا بلاک معتبر است و در شرط پذیرش قرار میگیرد یا خیر، تا در نهایت همگی آن بلاک را در بلاکچین خود قرار دهند. این باعث میشود تا کسی بدون سختی فرآیند اثبات انجام کار بلاک نامعتبری نسازد.

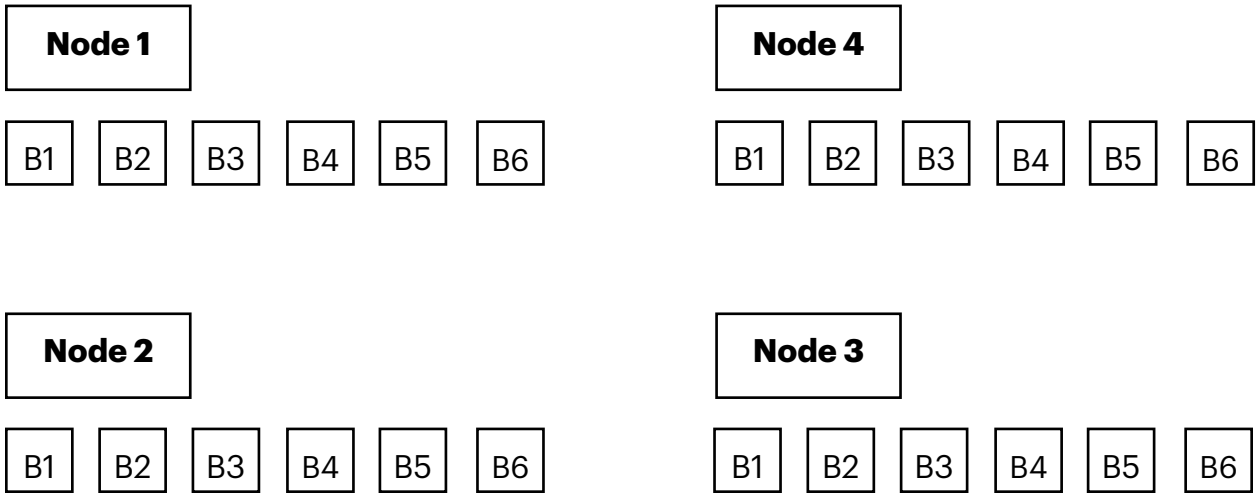
سوالی که در اینجا مطرح میشود این است که ممکن است به طور شانسی در یک زمان مشخص، یک یا چند گره ی دیگر نیز بلاکی را کشف کنند که دارای شرط پذیرش است. در این صورت کدام بلاک به عنوان بلاک معتبر بر روی بلاکچین تمامی گره ها قرار خواهد گرفت؟

در جواب این سوال یک قانون اساسی در پذیرش بلاک ها مطرح میشود بدین شکل که : همواره طولانی ترین زنجیر از بلاک ها، معتبر ترین بلاکچین میباشد.

بگذارید تا با یک مثال آنرا بررسی کنیم. در سناریوی پایین ۴ گره وجود دارند که مشغول فرآیند اثبات انجام کار هستند.



در شکل مشاهده میکنیم که از قضا گره های ۲ و ۴ به صورت همزمان هر کدام به ترتیب بلاک جدیدی را که شامل اصل پذیرش میباشد را بدست آورده اند. حال بر طبق اصل گفته شده در بالا بلاکی که گره ی ۲ آن را بدست آورده معتبر است ، زیرا بلاکچینی که گره ی ۲ توسط خود دارد طولانی تر است. این بدان معناست که گره ی ۲ فرآیند اثبات انجام کار بیشتری را انجام داده است و یک بلاک بیشتر از گره ی ۴ دارد، پس بلاکچین او معتبر خواهد شد و بقیه ی گره ها موظفند از آن پس همگی یک نمونه از بلاکچین گره ی ۲ را برای خود نگه دارند و بر روی کشف بلاک های بعدی آن تلاش کنند.



فرآیند چک کردن بلاکچین گره ها با یکدیگر در مدت زمان های معینی میتواند انجام بپذیرد.

از دیگر نکات مهم در فرآیند اثبات انجام کار این است که عدد معین امگا (Ω)، در اصل سختی شبکه را تعیین میکند، بدین معنا که با تغییر این عدد پیدا کردن بلاک جدید با شرط پذیرش ممکن است سریع تر یا کند تر به وقوع بپیوندد.

برای مثال فرض کنید امگا در دو حالت برابر اعداد زیر باشد.

Ω_1 = 000043753546 , Ω_2 = 000000003546

واضح است که پیدا کردن بلاک با شرط پذیرش Ω_2 به مراتب سخت تر از پیدا کردن بلاک با حالت امگا Ω_1 میباشد. زیرا بدست آوردن هش بلاکی که مقدارش از Ω_2 کمتر باشد به مراتب سخت تر از است و مقدار Ω_2 بسیار کمتر از Ω_1 است. پس چنانچه Ω_2 برای شبکه در نظر گرفته شود سختی شبکه بیشتر از زمانی است که Ω_1 برای شبکه تعیین شود.