

## لوبرنج:

### بروتوكول تداول العملة الرمزية اللامركزية

دانيال وانج

جي زهو

أليكس وانج

daniel@loopring.org

jay@loopring.org

alex@loopring.org

ماتيو فينيستون

matt.finestone@gmail.com

<https://loopring.org>

٢٠١٨ مارس

h  
t

#### نبذة مختصرة

لوبرنج هو بروتوكول مفتوح لبناء منصات التداول اللامركزية. لوبرنج يعمل كمجموعة عامة من العقود الذكية المسؤولة عن التداول والتسوية ، مع مجموعة غير مرتبطة بالسلسلة من الجهات الفاعلة التي تقوم بتجميع الأوامر وتوصيلها . البروتوكول مجاني وقابل للتوسيع ويعلم أساس بناء الكلمة التطبيقات اللامركزية (dApps) التي تتضمن وظائف منصات التداول . تعمل معاييره القابلة للتشغيل البيني على تسهيل التداول المجهول . ومن التحسينات المهمة على بروتوكولات منصه التداول اللامركزية الحالية القدرة على مزج الطلبات ومطابقتها مع الأوامر الأخرى الغير متشابهة ، وتجنب قيود أزواج التداول ثانوي العملة الرمزية وتحسين السيولة بشكل كبير . بوظف لوبرنج أيضاً حلاً فريداً وقوياً لمنع تشغيل الواجهة: وهي محاولة غير عادلة لإرسال المعاملات إلى كلية أسرع من مزود الحل الأصلي . لوبرنج قائم على البلوكتشين ، وقابل للانتشار على أي بلوكشين مع وظيفة العقد الذكي . في وقت هذه الكتابة ، انه قابل للتشغيل على اثيريوم [2] و Qtum [3] مع NEO [4] قيد الإنشاء.

العديد من المخاطر والقيود العملية في استخدام منصات التداول وفي كثير من الحالات نجحت في التخفيف من المخاطر الأمنية باستخدام تقنيات البلوكتشين لالغاء الوساطة . ومع ذلك ، عندما تصبح قابلية DEX كبنية أساسية حاسمة للاقتصاد الجديد ، هناك مجال كبير لتحسين الأداء . يهدف Loopring إلى توفير أدوات معيارية للبنية التحتية المذكورة من خلال dApp بروتوكولها المفتوح القائم على تطبيقات dApp .

## 2- مشهد التداول الحالي

### 1.2 أوجه القصور في منصات التداول المركزية

المخاطر الرئيسية الثلاثة لمنصات التداول المركزية هي ؛ 1) انعدام الأمان ، 2) انعدام الشفافية ، و 3) نقص السيولة.

ينشأ انعدام الأمان عن المستخدمين الذين يسلمون عادة السيطرة على مفاتيحهم الخاصة (الأموال) إلى كيان مركري واحد . هذا يعرض المستخدمين لاحتمال أن منصات التداول المركزية تقع فريسة للقرصنة الاشرار . إن مخاطر الأمان والقرصنة التي تواجه جميع منصات التداول المركزية معروفة جيداً [10] [11] ، ومع ذلك يتم قولها في كثير من الأحيان على أنها "حصص مخططة" لتداول العملة الرمزية . لا تزال منصات التداول المركزية تمثل مصادر مخترقة لهجمات القرصنة لأن خوادمهم تحفظ بملايين الدولارات من أموال المستخدمين . يمكن لمطوري منصات التداول أيضاً إجراء أخطاء بريئة وعرضية مع أموال المستخدمين . ببساطة ، لا يتحكم المستخدمون في العمل الرمزية الخاصة بهم عند إيداعها في منصه تداول مركري .

مع انتشار الأصول القائمة على البلوكتشين ، الحاجة إلى تداول هذه الأصول بين الأطراف المختلفة ازداد بشكل ملحوظ . مع طرح آلاف العمل الرمزية الجديدة - بما في ذلك ترميز الأصول التقليدية - هذه الحاجة أصبحت كبيرة . سواء كان تبادل العمل الرمزية لـ لوافع تداول المضاربة ، أو التحويل إلى شبكات الوصول عبر فاندة العمل الرمزية الأصلية الخاصة بها ، فإن القدرة على تداول واحد من الأصول المشفرة لأخرى هو الأساس للنظام الأكبر . في الواقع ، هناك طاقة محتملة للأصول [5] ، وتحقيق هذه الطاقة - فتح رأس المال - يتطلب ليس فقط تأكيد الملكية ، التي تسمح تقنيات البلوكتشين بثباته ، ولكن القدرة على نقل هذه الأصول وتحويلها بحرية .

على هذا النحو ، فإن تداول العمل الرمزية (القيمة) هي حالة استخدام مقدعة للتكنولوجيا البلوكتشين . حتى الآن ، ومع ذلك ، اتفق عشاق التشفير إلى حد كبير بتداول العمل الرمزية على منصات التداول المركزية التقليدية . هناك حاجة إلى بروتوكول لوبرنج لأنه ، كما البيتكوين [6] يؤكد بشكل طوعي على ، ما يتعلق بالفقد الإلكتروني للأقران (peer-to-peer) ، "تضييع الفوائد الرئيسية إذا كان لا يزال هناك حاجة إلى طرف ثالث موثوق به لمنع الإنفاق المزدوج" ، لذلك أيضاً الفوائد الرئيسية للأصول اللامركزية تفقد إذا كان يجب أن تمر عبر منصات تداول مركزية موثوقة ومبنية .

إن تداول العملات الرمزية اللامركزية في منصات التداول المركزية لا معنى له من الناحية الفلسفية ، حيث تفشل في دعم الفضائل التي تتبعها هذه المشاريع اللامركزية . هناك أيضاً

ويؤدي الافتقار إلى الشفافية إلى تعرض المستخدمين لخطر حدوث منصات تداول غير مشفوشة تتصرف بشكل غير عادل. ويمكن الفرق هنا في عوامل سوء تشغيل منصة التداول ، حيث لا يقوم المستخدمون فعليًا بالتداول في أصولهم الخاصة في منصات التداول المركزية ، ولكن بالأحرى يستخدمون IOU. عندما يتم إرسال العمل الرمزي إلى محفظة منصة التداول ، فإن منصة التداول تأخذ الوصاية ، وتقدم IOU. جميع منصات التداول بعد ذلك تتم بشكل فعال بين الـ IOU للمستخدمين. من أجل السحب ، يسترد المستخدمون قيمة نظام IOU الخاص بهم من خلال منصات التداول ، ويستمون العمل الرمزي إلى عنوان المحفظة الخارجية الخاصة بهم. خلال هذه العملية ، يوجد نقص في الشفافية ، ويمكن أن يتم إغلاق الحساب ، أو تجميد حسابك ، أو الإفلاس ، إلخ. ومن الممكن أيضًا أن يستخدموا أصول المستخدمين لأغراض أخرى أثناء الاحتياز ، مثل إقراضهم لأطراف ثالثة. يمكن أن يؤدي الافتقار إلى الشفافية إلى تكبد المستخدمين دون فقد كامل للأموال ، كما هو الحال في رسوم منصات التداول المرتفعة ، والتأخير في ذروة الطلب ، والمخاطر التقطيعية ، والأوامر التي يتم تنفيذها على المستوى الأمامي.

نقص السيولة. من وجهة نظر مشغلي منصات التداول ، السيولة المجزأة تمنع تسجيل الدخول في منصات التداول الجديدة بسبب سيناريوهين يشتركان الغزو. أولًا ، تفوت منصات التداول مع أكبر عدد من أزواج التداول ، لأن المستخدمين يجدون أنه من المرغوب فيه إجراء جميع صفقاتهم في منصة تداول واحدة. ثانياً ، تفوت منصات التداول مع أكبر دفتر طلبات ، بسبب فروق الأسعار المرغوبة عند كل زوج تداول. هذا لا يشجع المنافسة من الوافدين الجدد لأنه من الصعب عليهم بناء السيولة الأولية. ونتيجة لذلك ، فإن العديد من منصات التداول تسيطر على حصة كبيرة من السوق على الرغم من شكاوى المستخدمين وحتى حوادث الاختراق الرئيسية. تجدر الإشارة إلى أنه مع فوز منصات التداول المركزية بحصتها في السوق ، فإنها تصبح هدف اختراق دائمًا و في أي وقت. من وجهة نظر المستخدمين ، فإن السيولة المجزأة تقلل إلى حد كبير من تجربة المستخدم. في منصات التداول المركزي ، يمكن للمستخدمين التداول فقط داخل صناديق السيولة الخاصة بمنصات التداول ، ضد دفتر الطلبات الخاص بهم ، وبين أزواج العمل الرمزي المدعومة. للتداول في العملة الرمزية A من أجل العملة الرمزية B ، يجب على المستخدمين الذهاب إلى منصة تداول تدعم كلا العمل الرمزي أو التسجيل في منصات التداول المختلفة ، والكشف عن المعلومات الشخصية. غالباً ما يحتاج المستخدمون إلى تنفيذ الصفقات الأولية أو المتوسطة ، عادةً مقابل الـ ETH أو الـ BTC ، ودفع فروق أسعار العطاء في العملية. وأخيرًا ، قد لا تكون سجلات الطلبات عميقه بما فيه الكفاية لإتمام الصفقة دون أي انزلاق مادي. حتى إذا كانت منصة التداول تهدف إلى معالجة كميات كبيرة ، فليس هناك ما يضمن أن هذا الحجم والسيولة ليسا مزيفين [12]. والنتيجة هي مستودعات سيولة منفصلة ونظام حجزًا يشبه النظام المالي القديم ، مع حجم تداول مركزي هام على عدد قليل من منصات التداول . إن وعود السيولة العالمية بالحصانات لا تحمل أي ميزة داخل منصات التداول المركزية.

## 2.2 عدم ملاءمة منصات التداول اللامركزية

تختلف منصات التداول اللامركزية عن منصات التداول المركزية جزئياً لأن المستخدمين يحتفظون ببساطة على مفاتيحهم الخاصة (الأصول) عن طريق تنفيذ الصفقات مباشرة على البلوكشين الأساسي. من خلال الاستفادة من التكنولوجيا الآمنة للعمل المشفر نفسها ، فإنها تخفف بنجاح العديد من المخاطر المذكورة أعلاه المحيطة بالأمن. ومع ذلك ، فإن المشاكل لا تزال قائمة فيما يتعلق بالأداء والقيود الهيكلية.

غالباً ما تظل السيولة مشكلة حيث يجب على المستخدمين البحث عن الأطراف المقابلة عبر مجموعات ومعايير السيولة المتفاوتة. توجد تأثيرات السيولة المجزأة إذا لم تستخدم dApps أو DEXs بشكل عام معابر متعددة للتنقل المتداخل ، وإذا لم يتم مشاركة / نشر الطلبات عبر شبكة واسعة. يمكن أن تؤثر سيولة سجلات الأوامر المحدودة ، وعلى وجه التحديد ، على المرونة - مدى سرعة تجديد حد الأوامر المنفذة - بشكل ملحوظ على استراتيجيات التداول الأمثل [13]. إن غياب مثل هذه المعايير لم يؤد فقط إلى انخفاض السيولة ، بل إلى التعرض لمجموعة من العقود الذكية غير المحمية التي قد تكون غير آمنة.

علاوة على ذلك ، بما أن عمليات التداول تتم على سلسلة ، فإن DEXs تمثل حدود البلوكشين الأساسي ، وهي: قابلية التوسيع ، والتأخير في التنفيذ (التعدين) ، والتعديلات المكلفة على الطلبات. وبالتالي ، فإن سجلات أوامر البلوكشين لا توسيع بشكل جيد ، حيث أن تنفيذ الشفرة على البلوكشين يتطلب تكلفة (جاز) ، مما يجعل من الإلغاء المتعدد لأوامر أمرًا باهظاً.

وأخيراً ، لأن سجلات اوامر البلوكشين علنية ، المعاملة لتصنع امرا ما فانه يكون مرئياً من قبل المدققين حيث انها تنتظر حتى تتفق الى الكتلة التالية لها وتوضع في سجل الاوامر. هذا التأخير يعرض المستخدم لخطر أن يتوجه لللامام وأن يتحرك السعر أو التنفيذ ضده.

## 2.3 الحلول الهجينة

للأسباب المذكورة أعلاه ، فإن منصات التداول القائمة على البلوكشين البحث لديها قيود تجعلها غير قادر على المنافسة مع منصات التداول ات المركزية. هناك مقايسة بين الثقة الممثلة في السلسلة ، وسرعة منصات التداول المركزي ومرنة الطلب. تم البروتوكولات مثل Loopring و 0x [14] حلًا للتسوية على السلسلة بإدارة أوامر خارج السلسلة. تدور هذه الحلول حول العقود الذكية المفتوحة ، ولكنها تتخلى حدود قابلية التوسيع من خلال تنفيذ عدة وظائف خارج السلسلة وإعطاء نقاط مرنة في تحقيق الأدوار المهمة للشبكة. نهجنا لحل هجين من خلال هذه الورقة.

تكون حلقة الطلب صالحة عندما يمكن تنفيذ جميع المعاملات المركبة بسعر صرف يساوي أو أفضل من المعدل الأصلي المحدد ضمنياً من قبل المستخدم. للتحقق من صلاحية حلقة الطلب ، يجب أن تتفق العقود الذكية لبروتوكول لوبرنج حلقات الأوامر من منقين الحلقة حيث يكون سعر صرف الأصول لجميع الطلبات متساوياً أو أكبر من 1.

لتفرض أن أليس و بوب يرغبان في تبادل العملة الرقمية A و B. أليس لديه 15 عملة رقمية من A وتريد 4 عملة رقمية من B لها ، يملك بوب 10 عملة رقمية B ويريد 30 عملة رقمية A له.

من يشتري ومن يبيع؟ هذا يعتمد فقط على الأصل الذي نقوم بتحديده لتقديم عرض للأسعار. إذا كانت العملة الرقمية A هي المرجع ، فإن أليس ستشتري العملة الرقمية B بسعر  $\frac{4}{15}$  = A3.75 ، بينما سيبيع بوب العملة الرقمية B بسعر  $\frac{10}{30}$  = 10/30 A3.00. في حالة تحديد العملة الرقمية B كمرجع ، نفترض أن أليس ستبيع 15 عملة رقمية A بسعر  $\frac{15}{4}$  = 15/4

$0.26666667B$  و بوب سيشتري 10 عملة رقمية A بسعر  $\frac{30}{10} = 30/10 = 0.33333334B$ . وبالتالي ، من هو المشتري أو البائع اعتباطيا.

في الحالة الأولى ، تكون أليس على استعداد لدفع سعر أعلى (A 3.75) من السعر الذي يبيحه بوب لعملاته الرقمية مقابل (3.00A) ، بينما في الحالة الثانية ، يكون بوب على استعداد لدفع سعر أعلى (B0.33333334) من السعر الذي تقوم أليس ببيحه لعملاتها الرقمية مقابل (0.12666667B). من الواضح أن التداول ممكن عندما يرغب المشتري في دفع سعر متساوي أو أعلى من سعر البائع.

$$\frac{\frac{15}{4}}{\frac{30}{10}} = \frac{\frac{10}{4}}{\frac{15}{30}} = \frac{15}{4} \cdot \frac{10}{30} = 1.25 > 1 \quad (1)$$

وبالتالي ، للمقدرة على تنفيذ مجموعة من الأوامر n ، بشكل كامل أو جزئي ، نحتاج إلى معرفة ما إذا كان ناتج كل واحد من أسعار الصرف مثل نتائج أوامر الشراء برقم أكبر من أو يساوي 1. إذا كان الأمر كذلك ، يمكن أن تكون الأوامر n منفذة جزئياً أو كلياً [17].

إذا قمنا بإدخال نظير ثالث ، تشارلي ، بحيث ترغب أليس في إعطاء X1 للعملة الرقمية A وتحصل على y1 من العملة الرقمية B ، بوب يريد إعطاء X2 للعملة الرقمية B وتحصل على y2 للعملة الرقمية C ، وتريد تشارلي إعطاء X3 للعملة الرقمية C وتحصل على y3 للعملة الرقمية A . العملة الرقمية الضرورية موجودة ، والتداول ممكن إذا:

$$\frac{x_1 \cdot x_2 \cdot x_3}{y_1 \cdot y_2 \cdot y_3} \geq 1 \quad (2)$$

انظر القسم 7.1 لمزيد من التفاصيل حول أوامر لوبرنج

مع ذلك ، لا تزال هناك عيوب للنموذج الهجين أيضاً [15]. يقترح بروتوكول Loopring اختلافات ذات معنى في نهجنا لحل هجين من خلال هذه الورقة.

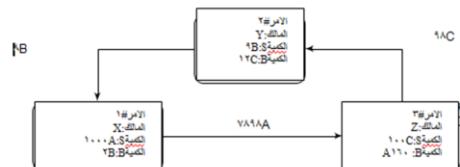
### 3 - بروتوكول loopring

ليس Loopring DEX نموذجي لبناء DEXs على العديد من تقنيات البلوكشين. تقوم بتقسيم الأجزاء المكونة لمنصة التداول التقليدية وتقديم مجموعة من العقود الذكية العامة والجهات الفاعلة اللامركزية في مكانها. وتشمل الأدوار في الشبكة المحافظ ، والمرحلات ، بلوكشين الكونسورتيوم في تقاسم السيولة ، ومستعرضين سجلات الطلبات ، وحلقة المنقين ، وخدمات ترميز الأصول. قبل تحديد كل منها ، يجب علينا أولاً أن نفهم أوامر Loopring.

#### 3.1 ترتيب الطوق

يتم التعبير عن أوامر gLoopring في ما نسميه نموذج أمر أحادي الاتجاه [16] (UDOM). UDOM يعبر عن الطلبات كطلبات تبادل العملة الرقمية ، المبلغ بيع / المبلغ شراء ، (المبلغ مقابل البيع / الشراء) بدلاً من المزادات ويسأل. نظراً لأن كل طلب هو سعر صرف بين عملتين رمزيتين ، فإن الميزة القوية للبروتوكول هي مزاج ومطابقة الأوامر المتعددة في دائرة التداول. باستخدام ما يصل إلى 16 طلباً بدلاً من زوج تداول واحد ، هناك زيادة كبيرة في السيولة وإمكانية تحسين الأسعار..

الشكل : 1 حلقة الطلب ل 3 أوامر



يوضح الشكل أعلاه حلقة الطلب من 3 أوامر. لكل طلب لبيع العملة الرقمية (العملة الرقمية بيع) هو طلب لشراء العملة الرقمية الأخرى (العملة الرقمية شراء). يقوم بإنشاء حلقة تسمح لكل أمر بتبادل العملة الرقمية المطلوبة دون الحاجة إلى طلب متعارض لزوجها. وبالطبع ، لا يزال من الممكن تنفيذ أوامر تداول الأزواج التقليدية ، في ما يتعلق بشكل أساسي بحلقة الطلب.

تعريف 3.1 (حلقة الطلب) دع  $C_n - 1, C_0, C_1, \dots, C_{n-1}, C_n$  لتكن عدد n من العملة الرقمية المختلفة،  $i \oplus 1 \rightarrow i \rightarrow 0 \rightarrow \dots \rightarrow 0 \rightarrow i \oplus 1 \rightarrow \dots \rightarrow i \rightarrow 0$  لتكن عدد n من الأوامر أو أي  $i \oplus 1, \dots, 0, \dots, i \rightarrow 0 \rightarrow i \oplus 1 \rightarrow \dots \rightarrow i \rightarrow 0 \rightarrow \dots \rightarrow 0 \rightarrow i \oplus 1 \rightarrow \dots \rightarrow i \rightarrow 0$  أن تكون أوامر n.

يمكن لهذه الأوامر تشكيل حلقة طلب للتداول:

$$O_{0 \rightarrow 1} \rightarrow \dots \rightarrow O_{i \rightarrow i \oplus 1} \rightarrow \dots \rightarrow O_{n-1 \rightarrow 0},$$

حيث n هو طول حلقة الطلب ، و  $i \oplus 1 \equiv i + 1 \bmod n$ .

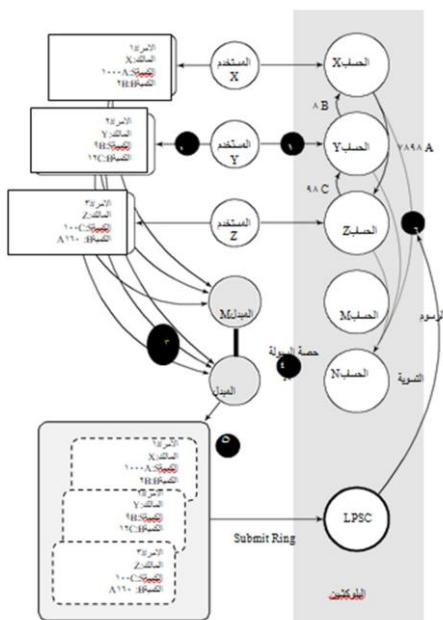
## 4- المشتركون في النظام

**1. ترخيص البروتوكول :** في الشكل 2، المستخدم Who يريد تداول العمل الرمزية ويأذن ل LPSC للتعامل مع عدد العمل الرمزية B التي يريد المستخدم بيعها. لا يؤدي هذا إلى قفل العمل الرمزية المستخدم ، الذي يظل حر في نقلها أثناء معالجة الطلب.

**2. إنشاء الطلب:** السعر الحالي وجز الطلب للعملة الرمزية B مقابل العملة الرمزية C يتم توفيرها من خلال المبدلات أو العوامل الأخرى المرتبطة بالشبكة ، مثل متصفحات حجز الطلب. يضع المستخدم Z (أمراً محدداً) يحدد المبلغ S والمبلغ B والمعلمات الأخرى من خلال أي واجهة محفظة مدمجة. يمكن إضافة مبلغ من LRx إلى الطلب كرسوم لمنقيين الحالة ؛ رسوم أعلى ل LRx تعني فرصة أفضل للمعالجة في وقت أسرع من قبل المنقيين. يتم توقيع تجزئة الطلب مع المفتاح الخاص للمستخدم Z.

**3. ثث الطلب:** ترسل المحفظة الطلب وتوقيعه إلى واحد أو أكثر من المبدلات.المبدلات تحدث حجز الطلب العام لها. لا يحتاج البروتوكول إلى حجوزات الطلب لكي يتم بناؤها بطريقة معينة ، مثل من يأتي أولاً يخدم أولاً. بدلاً من ذلك ، تمتلك المراحل القدرة على اتخاذ قرارات التصميم الخاصة بها في بناء حجوزات طلباتها.

**4. تقاسم السيولة :** ثبت المبدلات الطلب إلى مراحل (مبدلات) أخرى من خلال أي وسيلة اتصال عشوائية. مرة أخرى ، هناك مرونة كيف / اي من العقد تتفاعل . ولتسهيل مستوى معين من الاتصال بالشبكة ، هناك شبكة الترحيل (التبديل) مدمجة لتقاسم السيولة باستخدام البلوكشين المجمع . كما ذكرنا في القسم السابق ، تم تحسين شبكة الترحيل هذه للسرعة والشمولية .



الشكل 2: عملية تبادل loopring

يزود المشتركون في النظام بشكل مشترك جميع الوظائف التي تقدمها منصات التداول المركزي.

- المحافظ :** خدمة أو واجهة المحفظة العامة التي تمنح المستخدمين إمكانية الوصول إلى العمل الرمزية الخاصة بهم والطريق لإرسال أوامر إلى شبكة Loopring . سيتم تحفيز المحافظ لإنجاح الطلبات من خلال تقاسم الرسوم مع منقيين الحلقة (انظر القسم 8 ) . مع الإعتقد بأن مستقبل التداول سيحدث مع أمان محافظ المستخدم الفردي ، فإن ربط صناديق السيولة هذه من خلال بروتوكولنا هو أمر بالغ الأهمية.

- تحالف مشاركة سيولة البلوكشين/ شبكة الترحيل(التبديل):** شبكة التبادل لتقاسم الامر والسيولة عندما يقوم العقد بتشغيل برنامج تبادل لوبرنج ، فإنها تكون قادرة على الانضمام إلى الشبكة الموجودة ومشاركة السيولة مع المراحل الأخرى عبر البلوكشين المجمع. تحالف البلوكشين الذي تقوم ببنائه كأول تنفيذ لديه مشاركة طلب في الوقت الفعلي تقريباً (الكتل من ثانية إلى ثانية) ، وتزيل السجل القديم للسامح بتزويذ أسرع من خلال العقد الجديدة. بشكل خاص ، لا تحتاج المبدلات إلى الانضمام إلى هذا الاتحاد. يمكنهم العمل بمفردهم و عدم تقاسم السيولة مع الآخرين ، أو يمكنهم بدء وإدارة شبكة تقاسم السيولة الخاصة بهم.

- المبدلات / منقيين الحلقة (Relays/Ring-Miners) :** المبدلات أو المراحل هي العقد التي تتنافى الأوامر من المحافظ أو شبكة المراحل ، تحافظ على سجل حجوزات الطلب العامة والتداول ، وبث الأوامر اختيارياً إلى المراحل الأخرى (عن طريق أي وسط عشوائي خارج السلسلة) و / أو عقد شبكة الترحيل. تعدين الحلقة هي ميزة - وليس شرطاً - للمرحلات. وهو تقييل حسابياً ويتم خارج السلسلة تماماً. تسمى المراحل بخاصية تعدين الحلقة التي يتم تشغيلها على "منقيين الحلقة" ، الذين يتوجون حلقات الأوامر عن طريق تجميع الأوامر المتباينة معاً . تكون المراحل مجانية في (1) كيفية اختيارها للتواصل مع بعضها البعض ، (2) كيفية بناء سجلات أوامرها ، و (3) كيفية تعدين حلقات الأوامر (خوارزميات التعدين).

- العقود الذكية لبروتوكول لوبرنج (LPSC) :** مجموعة من العقود الذكية العامة و المجانية التي تتحقق من حلقات الأوامر المتلقاة من المنقيين ، تسوى وتحول العمل الرمزية نيابة عن المستخدمين بطريقة آمنة ، وتكفие منقيين الحلقة والمحافظ برسوم ، وتصدر الأخذات . يستعرضوا على العمل الرمزية التي تم إصدارها يستمعوا إلى هذه الأخذات لبقاء سجلات حجز الطلبات و التداول . انظر الملحق (A) للحصول على التفاصيل.

- خدمات ترميز الأصول (ATS):** الجسر بين الأصول التي لا يمكن تداولها مباشرة على لوبرنج . إنها خدمات مركزية تدار من قبل شركات أو منظمات جديرة بالثقة . يقوم المستخدمون بإيداع الأصول (مادية أو عملات ورقية أو عمل رمزية من سلاسل أخرى) ويحصلوا على العمل الرمزية التي تم إصدارها ، والتي يمكن استبدالها بالإيداع في المستقبل . لا يعد لوبرنج بروتوكول تداول عبر السلسلة (حتى يوجد حل مناسب) ، ولكن ATS يمكن من تداول العمل الرمزية [18] مع الأصول المادية وكذلك الأصول على تقنيات البلوكشين الأخرى.

## 6.2 تفاصيل السيولة

المرحلات حرة في تصميم كيفية مشاركة السيولة (الطلبات) مع بعضها البعض. إن اتحاد البلوكشين الخاص بنا ليس سوى حل واحد لإنجاز ذلك ، والنظام حر في التواصل والاتصال كما يحلو له. وبالإضافة إلى الانضمام إلى اتحاد البلوكشين ، يمكنها بناء وإدارة شؤونها الخاصة ، ووضع القواعد / الحواجز على النحو الذي يرونها مناسباً. يمكن أن تعمل المرحلات أياًً بمفردها ، كما هو واضح في تنفيذ المحفظة الحساسة لوقت. بالطبع ، هناك مزايا واضحة في التواصل مع المرحلات الأخرى في السعي إلى مؤشرات الشبكة ، ومع ذلك ، قد تستلزم نماذج العمل المختلفة تصاميم مشاركة غريبة وتقطيع الرسوم بأي عدد من الطرق.

## 7 - مواصفات البروتوكول

### 1.7 تحليل الطلب

الطلب عبارة عن حزمة من البيانات التي تصف القصد من تداول المستخدم. يتم تعريف أمر لوبرنچ باستخدام نموذج طلب أحدى الاتجاه ، أو UDOM ، كما يلي:

```
message Order {
    address protocol;
    address owner;
    address tokenS;
    address tokenB;
    uint256 amountS;
    uint256 amountB;
    uint256 lrcFee;
    unit256 validSince; // Seconds since epoch
    unit256 validUntil; // Seconds since epoch
    uint8 marginSplitPercentage; // [1-100]
    bool buyNoMoreThanAmountB;
    uint256 walletId;
    // Dual-Authoring address
    address authAddr;
    // v, r, s are parts of the signature
    uint8 v;
    bytes32 r;
    bytes32 s;
    // Dual-Authoring private-key,
    // not used for calculating order's hash,
    // thus it is NOT signed.
    string authKey;
}
```

لضمان أصل الطلب ، يتم توقيعه مقابل تجزئة معلوماته ، باستثناء authAddr ، مع المفتاح الخاص للمستخدم. يتم استخدام المعلم authAddr لتوقيع حلقات الطلب التي يتم ترتيب هذا الطلب كجزء منها ، والذي يمنع التشغيل الآمني. يرجى الرجوع إلى القسم 9.1 لمزيد من التفاصيل. يتم تمثيل التوقيع بواسطة 27 و sfields ، ويتم إرسالها بجانب معلومات الطلب عبر الشبكة. وهذا يضمن بقاء الطلب ثابتاً خلال فترة حياته الكاملة. على الرغم من أن الطلب لن يتغير أبداً ، لا يزال بإمكان البروتوكول حساب حالته الحالية استناداً إلى رصيد عنوانه بالإضافة إلى متغيرات أخرى.

1. **تعدين الحلقة (مطابقة الطلب)** : يحاول المنقبين تنفيذ الطلب كلّياً أو جزئياً بسعر الصرف المحدد أو بشكل أفضل من خلال مطابقته مع طلبات أخرى متعددة.

تعدين الحلقة هو السبب الرئيسي في أن البروتوكول قادر على توفير سيولة عالية على أي زوج . إذا كان المعدل الذي تم تنفيذه أفضل من ما يحدده المستخدم ٢ ، يتم مشاركة الهاشم بين جميع الطلبات في حلقة الطلب. كمكافأة ، يختار منقب الحلقة بين المطالبة بجزء من الهاشم (فصل الهاشم)، وإعادة LRX إلى المستخدم) ، أو ببساطة الاحتياط برسوم LRX.

2. **التحقق والتسوية** : يتم استلام حلقة الطلب بواسطة LPSC. يقوم بإجراء العديد من الفحوصات للتحقق من البيانات التي توفرها منقب الحلقة وتحدد ما إذا كان من الممكن تسوية حلقة الطلب كلّياً أو جزئياً (اعتماداً على معدل تنفيذ الطلبات داخل الحلقة والعمل الرمزية في محافظ المستخدمين). في حالة نجاح جميع عمليات التحقق ، يقوم العقد بتحويل العمل الرمزية إلى المستخدمين ويدفع رسوم منقب الحلقة والمحفظة في نفس الوقت . إذا كان رصيد المستخدم ٢ كما هو محدد بواسطة LPSC غير كافٍ ، فسيتم اعتباره منخفض: الطلب المنخفض سيتم تلقائياً تغييره إلى حجمه الأصلي إذا تم إيداع أموال كافية في عنوانه ، بخلاف الإلغاء ، الذي يعتبر عملية يدوية احادية الطريق ولا يمكن عكسها.

## 6 - المرونة التشغيلية

من المهم ملاحظة أن معيار لوبرنچ المفتوح يتيح للمشاركين مرونة كبيرة في كيفية عملهم. الفاعلون أحراز في تنفيذ نماذج أعمال جديدة وتوفير قيمة للمستخدمين ، وكسب رسوم LRX على الحجم أو غيرها من المقاييس في العملية (إذا اختاروا ذلك). النظام نظامي ويهدف إلى دعم المشاركة من العديد من التطبيقات.

### 6.1 سجل الطلبيات

يمكن للمرحلات تصميم سجلات طلباتها بأي عدد من الطرق لعرض ومطابقة طلبات المستخدمين. يتبع التنفيذ الأول لسجل الطلب الخاص بنا نموذج OTC ، حيث يتم وضع حد الأوامر على أساس السعر وحده. بعبارة أخرى ، لا تؤثر الطوابع الزمنية للأوامر على سجل الطلبات. ومع ذلك ، فإن المرحل حر في تصميم سجل الطلبات الخاص به بطريقة تحاكي محرك المطابقة النموذجي للتداول المركزي ، حيث يتم ترتيب الطلبات حسب السعر ، مع احترام الطوابع الزمنية أيضاً. إذا كان المرحل يميل إلى تقييم هذا النوع من سجل الطلبات ، فيمكنه امتلاك / دمج مع المحفظة ، وإرسال أوامر المحفظة هذه فقط إلى المرحل المنفرد ، الذي سيكون قادرًا على مطابقة الطلبات بناءً على الوقت. أي تكون من هذا القبيل ممكن.

في حين تتطلب بروتوكولات DEX الأخرى في بعض الأحيان المرحلات للحصول على موارد - الأرصدة الأولية للعملة الرمزية لوضع أوامر المستقبلين - تحتاج مرحلات لوبرنچ فقط إلى العثور على أوامر قابلة للتطابق لكي يتم إتمام الصفقة ، ويمكنها القيام بذلك بدون العمل الرمزية الأولية.

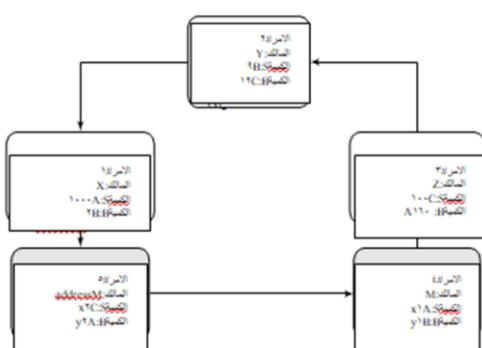
- بيع 10 ETH بسعر 300 دولار أمريكي / ETH . يمكن التعبير عن هذا الطلب على النحو التالي: Order(10, ETH, 3000, USD, False).
- بيع ETH بسعر 300 دولار أمريكي / ETH بـ 3000 دولار أمريكي. يمكن التعبير عن هذا الطلب على النحو التالي: Order(10, ETH, 3000, USD, True)
- اشتر 10 ETH بسعر 300 دولار أمريكي / ETH ، يمكن التعبير عن هذا الطلب على النحو التالي: Order(3000, USD, 10, ETH, True).
- أنفق 3000 دولار أمريكي لشراء أكبر عدد ممكن من ETH بـ 300 دولار أمريكي / ETH ، يمكن التعبير عن هذا الطلب على النحو التالي: Order(3000, USD, 10, ETH, False).

## 7.2 التحقق من الحلقة

لا تقوم عقود pring0Loo الذكية بإجراء عمليات حساب أو كمية سعر الصرف ، ولكن يجب أن تتحقق و تتحقق مما يزودها به متغيرين الحلقة لهذه القيم. يتم إجراء هذه الحسابات من قبل متغيرين الحلقة لسبعين رئيسين: (1) لغة البرمجة للعقود الذكية ، مثل الصالبة [19] على الأثيريوم ، لا يوجد لديها دعم لرياضيات العدد العشري ، لا سيما اعداد القرى ( $x, 1/n$ ) حساب الجذر رقم  $n$  للرقم العشري) ، و (2) من المستحسن أن يتم إجراء عملية حسابية خارج السلسلة للحد من حساب وتكلفة البلوكشين.

### 7.2.1 فحص فرع الحلقة

هذه الخطوة تمنع المراجعين من تحقيق جميع الهاشم بشكل غير عادل في حلقة الطلب عن طريق تنفيذ أوامر جديدة داخله. وبشكل أساسي ، بمجرد العثور على حلقة طلب صالحة بواسطة متغير الحلقة ، قد يكون من المغرى إضافة أوامر أخرى إلى حلقة الطلب لاستيعاب هامش المستخدمين (خصومات السعر) بشكل كامل. كما هو موضح في الشكل 3 أدناه ، فإن النتائج المحسوبة بدقة  $x_1$  و  $y_1$  و  $x_2$  و  $y_2$  ستجعل ناتج معدل جميع الطلبات هو 1 بالضبط ، وبالتالي لن يكون هناك خصم للأسعار.



الشكل 3: حلقة طلب مع الفرع

لا يشمل UDOM السعر (والذي يجب أن يكون رقمًا عشربيا بطبيعته) ، ولكن بدلاً من ذلك يستخدم المصطلح معدل أو ، والذي يتم التعبير عنه كمقدار / مبلغ. المعدل ليس رقم عشربي بل هو تعبير سيتم تقييمه فقط مع أعداد صحيحة أخرى غير موقعة عند الطلب ، للحفاظ على جميع النتائج الوسيطة كأعداد صحيحة غير موقعة وزيادة دقة الحساب

### 7.1.1 مبالغ الشراء

عندما يقوم منقب الحلقة بمطابقات حلقة الأوامر ، من الممكن أن يكون معدل أفضل قابلاً للتنفيذ ، ويسمح للمستخدمين بالحصول على مزيد من العمل الرمزية B بمبلغ B الذي حدده. ومع ذلك ، إذا كان buyNoMoreThanAmountB تم تعينه على True ، يضمن البروتوكول أن المستخدمين لا يتلقون أكثر من مبلغ B من العمل الرمزية B. وهكذا ، يحدد المعلم buyNoMoreThantokenB متى يعتبر buyNoMoreThantokenB الطلب منفذ كلبا. buyNoMoreThantokenB يطبق حد أقصى على المبلغ S أو المبلغ B ، ويسمح للمستخدمين بالتعبير عن نوايا تجارية أكثر دقة من أوامر الشراء / البيع التقليدية.

على سبيل المثال: مع  $amountB = 2$  و  $amountS = 10$  ، المعدل  $= 2/10 = 0.2$ . وهذا يكون المستخدم على استعداد لبيع 5 عمل رمزية لكل عمل رمزية B. يتطابق منقب الحلقة ويجد المستخدم معدل 4 ، مما يسمح للمستخدم بتلقي 2.5 عملة رمزية B بدلاً من 2. مع ذلك ، إذا كان المستخدم يريد 2 عملة رمزية buyNoMoreThanAmountB فقط وقام بتعيين إشارة True ، يقوم LPSC بإجراء المعاملة بسعر 4 وبيع المستخدم 4 عمل رمزية S لكل عملة رمزية B ، ويوفر بشكل فعال 2 عمل رمزية S. ضع في اعتبارك أن هذا لا يأخذ في الاعتبار رسوم التعدين (انظر القسم 8.1).

في الواقع ، إذا استخدمنا

```
Order(amountS,tokenS,
      amountB,tokenB,
      buyNoMoreThantokenB)
```

لتمثيل طلب في شكل مبسط ، فإنه بالنسبة لأسوق / ETH في منصات التداول التقليدي ، يمكن لمناذج البيع والشراء التقليدية التعبير عن الطلب الأول والثالث أدناه ، ولكن ليس الثاني:

كما يدعم LPSC إلغاء جميع الطلبات الخاصة بأي زوج تداول مع الحدث OrdersCancelled وبلغ جميع الطلبات لعنوان مع حدث AllOrdersCancelled.

#### 7.2.4 قياس الطلب

يتم قياس الطلب وفقاً لتاريخ المبالغ المنفذة والملغاة والرصيد الحالي لحسابات المرسلين. تجد العملية الطلب مع أصغر مقدار يتم تنفيذه وفقاً للخصائص المذكورة أعلاه وتستخدم كمرجع لقياس جميع المعاملات في حلقة الطلب.

يمكن أن يساعد العثور على الطلب ذو أدنى قيمة في معرفة حجم التنفيذ لكل طلب. على سبيل المثال ، إذا كان الطلب  $i-th$  هو أقل قيمة ، فإن عدد العمل الرمزية المباعة من كل طلب  $S$  و عدد العمل الرمزية المشتراء  $b$  من كل طلب يمكن حسابها على النحو التالي:

$$\begin{aligned} s^{i-1} &= s_i, \quad b^i = s^{i-1}/r^{i-1}, \\ s^{i\Theta 1} &= b^i, \quad b^{i\Theta 1} = s^{i\Theta 1}/r^{i\Theta 1}, \\ s^{i\Theta 2} &= b^{i\Theta 1}, \quad b^{i\Theta 2} = s^{i\Theta 2}/r^{i\Theta 2}, \\ &\dots \end{aligned}$$

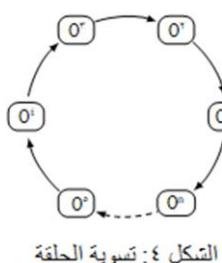
حيث  $s_i$  هو الرصيد المتبقى بعد تنفيذ الطلبات جزئياً.

أثناء التنفيذ ، يمكننا أن نفترض بأمان أي طلب في حلقة الطلب بأقل قيمة ، ثم يقوم بالنكرار عبر حلقة الطلب على الأكثر مرتبين لحساب حجم التنفيذ لكل أمر.

مثال: إذا كان الحد الأدنى المطلوب تنفيذه مقارنة بالطلب الأصلي هو 5٪ ، فإن جميع المعاملات في حلقة الطلب يتم تخفيضها إلى 5٪. بمجرد الانتهاء من المعاملات ، يجب تنفيذ كلية للطلب الذي كان يعتبر ذو أصغر كمية متبقية.

#### 7.3 تسوية الحلقة

إذا كانت حلقة الطلب تلبي جميع الفحوصات السابقة ، يمكن إغلاق حلقة الطلب ، ويمكن إجراء المعاملات. هذا يعني أن جميع الطلبات  $n$  تشكل حلقة طلب مغلقة ، متصلة كما هو موضح في الشكل: 4



الشكل 4: تسوية الحلقة

لإجراء المعاملات ، يستخدم LPSC العقد الذكي TokenTransferDelegate يجعل تحديث بروتوكول العقد الذكي أسهل حيث أن جميع الطلبات تحتاج فقط إلى تصريح هذا التقويض بدلاً من إصدارات مختلفة من البروتوكول.

هذا هو خطر الصفر ، إضافة القيمة صفر إلى الشبكة ، ويعتبر السلوك غير عادل من قبل منقب الحلقة. لمنع هذا ، تتطلب Loopring أن الحلقة الصالحة لا يمكن أن تحتوي على أي حلقات فرعية. للتحقق من ذلك ، يضمن LPSC أن لا تكون العمدة الرمزية في موضع الشراء

أو البيع مرتبين. في الرسم البياني أعلاه ، يمكننا أن نرى أن العمدة الرمزية A هي عمدة رمزية للبيع مرتبين وعمدة رمزية للشراء مرتبين ، والذي لن يتم السماح به.

#### 7.2.2 التحقق من تنفيذ السعر

يتم إجراء حسابات سعر الصرف في حلقة الطلب من قبل منقبين الحلقة لأسباب مذكورة أعلاه. يجب على LPSC التتحقق من صحتها. أولاً ، تتحقق من أن معدل الشراء الذي يمكن لمدقق الحلقة تنفيذه لكل طلب يساوي أو أقل من سعر الشراء الأصلي الذي يحدده المستخدم. يضمن ذلك للمستخدم الحصول على الأقل على سعر الصرف الذي طلبه أو أفضل من الصفة. وبمجرد تأكيد أسعار الصرف ، يضمن LPSC أن كل طلب في حلقة الطلب يشترك بنفس الخصم في السعر. على سبيل المثال ، إذا كان السعر المخفض هو  $\gamma$  ، فيكون السعر لكل طلب:

$$r_{0 \rightarrow 1} \cdot (1 - \gamma), r_{1 \rightarrow 2} \cdot (1 - \gamma), r_{2 \rightarrow 0} \cdot (1 - \gamma)$$

$$r_{0 \rightarrow 1} \cdot (1 - \gamma) \cdot r_{1 \rightarrow 2} \cdot (1 - \gamma) \cdot r_{2 \rightarrow 0} \cdot (1 - \gamma) = 1 \quad (3)$$

وبالتالي:

$$\gamma = 1 - \sqrt[3]{r_{0 \rightarrow 1} \cdot r_{1 \rightarrow 2} \cdot r_{2 \rightarrow 0}}. \quad (4)$$

إذا تجاوزت المعاملة عدد الطلبات ، يكون الخصم:

$$\gamma = 1 - \frac{1}{Q_{i-1} + 1} \cdot \frac{1}{r^i}. \quad (5)$$

حيث  $i-th$  هو معدل دوران الطلب للطلب  $i-th$ . من الواضح ، فقط عندما يكون معدل الخصم هو  $0 \leq \gamma \leq 1$  يمكن تنفيذ هذه الأوامر ؛ وسعر الصرف الفعلي للأمر  $i-th$  هو  $(0)$

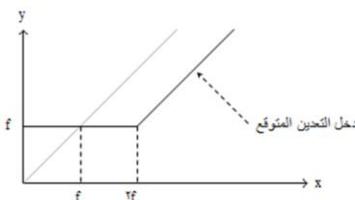
$$r^i = r^i \cdot (1 - \gamma), \quad r^i \leq r^i.$$

#### 7.2.3 تعقب وإلغاء التنفيذ

يمكن للمستخدم إلغاء الطلب جزئياً أو كلية عن طريق إرسال معاملة خاصة إلى LPSC ، تحتوي على تفاصيل حول الطلب والمبلغ المراد إلغاءه. يأخذ LPSC ذلك في الاعتبار ، يخزن المبلغ المراد إلغاءه ، ويعطى حدث OrderCancelled إلى الشبكة. يتبع LPSC المبالغ التي تم تنفيذها وإلغاؤها عن طريق تخزين قيمها باستخدام تجزئة الطلب كمعرف. هذه البيانات يمكن الوصول إليها بشكل عام و الاحداث OrderCancelled/ OrderFilled تُصدر عند تغييرها. تتبع هذه القيم أمر بالغ الأهمية ل LPSC خلال خطوة تسوية حلقة الطلب.

(منشى الطلب) رسوم ، و تساوي  $L_{RX}$  التي كان المستخدم سيدفعها إلى منقب الحلقة كرسوم. هذا يزيد القيمة الحدية حيث سيختار منقب الحلقة تقسيم الهامش إلى ضعف رسوم  $L_{RX}$  للطلب ، مما يزيد من الميل إلى خيار رسوم  $L_{RX}$ . يسمح هذا لمنقب الحلقة بالحصول على دخل ثابت على حلقات الطلب ذات الهامش المنخفض لمقاييسه تلقى دخل أقل في حلقات الطلب ذات الهامش الأعلى. يعتمد نموذج الرسوم لدينا على التوفيق بأنه مع نمو السوق وتضووجه ، سيكون هناك عدد أقل من حلقات الطلب ذات الهامش المرتفع ، مما يتطلب رسوم  $L_{RX}$  ثابتة كحافز.

نتهي بالرسم البياني التالي:



الشكل ٦: نموذج رسوم Loopring

حيث  $f$  هي رسوم  $L_{RX}$  ،  $x$  هو تقسيم الهامش ،  $y$  هو دخل التعدين.  $y = \max(f, x-f)$  كما هو مشار إليه بواسطة الخط الصلب ؛ إذا كانت رسوم  $L_{RX}$  للطلب 0 ، فإن المعادلة تكون  $y = \max(0, x-0)$  والتي تبسط إلى  $y = x$  كما هو محدد بالخط الرمادي.

النتائج هي:

1. إذا كان تقسيم الهامش 0 ، سوف يختار منقبين الحلقة رسوم  $L_{RX}$  السطحية ولا يزالون محفزيين.
2. إذا كانت رسوم  $L_{RX}$  هي 0 ، فإن نتائج الخط الرمادي والدخل يستند إلى نموذج الخط العام.
3. عندما يكون دخل تقسيم الهامش أكبر من  $x/2$  (رسوم  $L_{RX}$ ) ، يختار منقبين الحلقة تقسيم الهامش ويدفعوا إلى المستخدم.

تجدر الإشارة إلى أنه إذا كانت رسوم  $L_{RX}$  غير صفرية ، وبغض النظر عن الخيار الذي يختاره منقب الحلقة ، فسيكون هناك دائمًا نقل ل  $L_{RX}$  بين منقب الحلقة ومرسل الطلب. إما أن يحصل منقب الحلقة على رسوم  $L_{RX}$  ، أو يدفع رسوم  $L_{RX}$  مرة أخرى إلى المرسل لاتخاذ تقسيم الهامش.

لكل طلب في حلقة الطلب ، يتم دفع العمل الرمزية إلى الطلب التالي أو السابق بناءً على التنفيذ. ثم يتم دفع رسوم منقب الحلقة حسب نموذج الرسوم الذي يختاره منقب الحلقة. وأخيراً ، بمجرد إجراء جميع المعاملات ، يتم إصدار حدث RingMined.

### 7.3.1 الأحداث المصدرة

يصدر البروتوكول أحاديث تسمح للمرحلات و متصفحات الأوامر والممثلين الآخرين بتلقي تحديثات سجل الطلبات بأكبر قدر ممكن من الكفاءة. الأحداث المصدرة هي:

- OrderCancelled: تم إلغاء طلب معين.
- OrdersCancelled: تم إلغاء جميع طلبات زوج التداول من عنوان المالك.
- AllOrdersCancelled: تم إلغاء جميع طلبات جميع أزواج التداول من عنوان المالك.
- RingMined: تمت تسوية حلقة الطلب بنجاح. يحتوي هذا الحدث على بيانات متعلقة بكل نقل للعملة الرمزية داخل الحلقة.

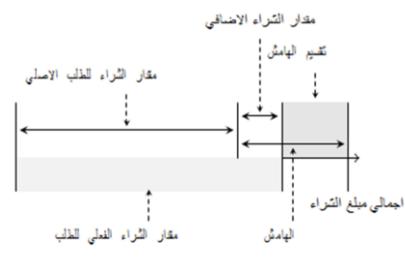
## 8 - العملة الرمزية $L_{RX}$

$L_{RX}$  هي رمز العملة الرمزية العامة لدينا.  $LRC$  هي العملة الرمزية ل Loopring على اثيريوم ،  $LQ$  على NEO ،  $LRN$  على Qtum ، الخ. سيتم إدخال أنواع  $L_{RX}$  الأخرى في المستقبل حيث يتم نشر  $L_{RX}$  على تقنيات البلوكشين العامة الأخرى.

### 8.1 نموذج الرسوم

عندما يقوم المستخدم بإنشاء طلب ، فإنه يحدد مبلغ  $L_{RX}$  دفعه إلى منقب الحلقة كرسوم ، بالإضافة إلى نسبة الهامش (marginSplitPercentage) الذي يتم إجراؤه بناء على الطلب الذي يمكن لمنقب الحلقة المطالبة به. وهذا ما يسمى بتقسيم الهامش. يتم ترك قرار أي واحد لل اختيار (الرسوم أو تقسيم الهامش) إلى منقب الحلقة .

تمثيل تقسيم الهامش:



الشكل ٥: هامش بنسبة ٦٠٪

إذا كان الهامش الموجود على حلقة الطلب صغيراً جداً ، فسيقوم منقب الحلقة بال اختيار رسوم  $L_{RX}$ . إذا كان الهامش ، على النقيض من ذلك ، الهامش يكون كبيراً بما فيه الكفاية لانتاج تقسيم الهامش الذي قيمة أكثر بكثير من رسوم  $L_{RX}$  ، سيختار منقب الحلقة تقسيم الهامش. هناك شرط آخر ، على أي حال: عندما يختار منقب الحلقة تقسيم الهامش ، يجب أن يدفع للمستخدم

عندما يسرق القائم بالتداول المسبق أمراً واحداً أو أكثر من معاملة توسيعة طلبات الحلقة الملعقة ؛ وبالنسبة إلى Loopring : عندما يسرق القائم بالتداول المسبق حلقة الطلب بالكامل من معاملة ملعقة.

عندما لا يتم تأكيد معاملة submitRing ولا تزال في تجمع المعاملات الملعقة ، يمكن لأي شخص أن يكتشف مثل هذه المعاملة بسهولة ويستبدل عنوان المنقب بعنوانه الخاص (filcherAddress) ، ثم يمكنه إعادة التوقيع على الحمولة باستخدام filcherAddress لاستبدال توقيع حلقة الطلب. يمكن أن يقوم المختلس بتعيين سعر أعلى للجاز وتقييم معاملة جديدة على أمل أن يقوم منقبين الكلة باختيار معاملته الجديدة في الكلة التالية بدلاً من المعاملة الأصلية .submitRing

الحلول السابقة لهذه المشكلة كانت لها جوانب سلبية مهمة: تتطلب المزيد من المعاملات وبالتالي تكافف منقبين الحلقة الكثير من الجاز ؛ وتأخذ على الأقل ضعف الكلتة لتسوية حلقة الطلب. حلنا الجديد ، التأليف المزدوج [21] ، ينطوي على آلية إنشاء مستويين من الترخيص للأوامر - واحد للتسوية ، والآخر للتتفقيب عن الحلقة.

#### عملية التأليف المزدوج:

1. لكل طلب ، سيولد برنامج المحفظة زوجاً عشوائياً من المفتاح العام / المفتاح الخاص ، ويضع زوج المفاتيح في مقتطف JSON التابع للطلب. (البديل هو استخدام العنوان المستمد من المفتاح العام بدلاً من المفتاح العام نفسه لتقليل حجم البایت. نستخدم authAddr لتمثيل المفتاح المالي مثل هذا العنوان ، و authKey لتمثيل المفتاح الخاص المطابق له).authAddr
2. حساب تجزئة الطلب مع جميع الحقول في الطلب باستثناء (r و s و v و authKey) ، وتتوقيع التجزئة باستخدام المفتاح الخاص بالمالك (وليس authKey).
3. سترسل المحفظة الطلب مع authKey إلى المرحلات من أجل تعدين الحلقة. سيتحقق منقبين الحلقة من أن authKey و authAddrare صحيح وأن توقيع الطلب صالح فيما يتعلق بعنوان المالك.
4. عند تحديد حلقة الطلب ، سيستخدم منقبين الحلقة authKey لكل طلب لتوقيع تجزئة الحلقة و minerAddress و جميع معلومات التعدين. إذا كانت حلقة الطلب تحتوي على n من الأوامر ، فستكون هناك توقيعات n بواسطة n من ال authKeys . نحن نسمي هذه التوقيع ب authSignatures . قد يحتاج منقب الحلقة أيضاً إلى توقيع تجزئة الحلقة مع جميع معلومات التعدين باستخدام المفتاح الخاص له .minerAddress

سوف يشارك منقبين الحلقة بنسب معينة من الرسوم مع المحفظ. عندما يضع المستخدم طلباً عبر محفظة ويتم تنفيذه ، يتم مكافأة المحفظة بجزء من الرسوم أو تقسيم الهاشم. على الرغم من أن هذه الوحدات ، ونماذج الأعمال الفريدة أو التطبيقات ممكنة ، إلا أن ميلنا هو أن تستلم المحفظ ما يقارب 20٪ - 25٪ من الرسوم المكتسبة. تمثل المحفظ هدفاً أساسياً لنكمال بروتوكول Loopring نظراً لأنها تحتوي على قاعدة المستخدمين ، ولكن مصدر الدخل قليل أو معدوم.

## 8.2 الحكم أو العمل اللامركزي

في الأصل ، بروتوكول Loopring هو بروتوكول اجتماعي بمعنى أنه يعتمد على التنسيق بين الأعضاء للعمل بفعالية نحو الهدف. وهذا لا يختلف عن بروتوكولات التشفير الاقتصادي بشكل عام ، بل إن فائدتها محمية إلى حد كبير من خلال نفس آليات مشاكل التنسيق [20] ، واتزان الحدث القاسي ، والعقلانية المحدودة. ولهذه الغاية ، لا تستخدم العمل الرمزية LRx فقط لدفع الرسوم ، ولكن أيضاً لمواومة المعاوز المالية للمشاركين في الشبكة المختلفة. ومثل هذا المواءمة ضروري لاعتماد واسع النطاق لأي بروتوكول ، ولكنه شديد الحدة بالنسبة لبروتوكولات منصات التداول ، بالنظر إلى أن النجاح يعتمد بشكل كبير على تحسين السبولة في نظام لامركزي قوي.

سيتم استخدام العمل الرمزية LRx لتفعيل تحديثات البروتوكول من خلال الإدارة اللامركزية. تخضع تحديثات العقود الذكية لحاملي العملة الرمزية لضمان الاستمرارية والسلامة ، والتخفيف من مخاطر سحب السبولة من خلال عدم التوافق. وبالنظر إلى أن العقد الذكي لا يمكن تغييرها بمجرد نشرها ، فهو خطير بان تستمر ال dApps أو المستخدمين النهائيين في التفاعل مع الإصدارات التي تم إيقافها وإبطال نظرهم من العقود المحدثة.

تعتبر قابلية الترقية أمراً حاسماً لنجاح البروتوكول حيث أنه يجب أن يتكيف مع متطلبات السوق وتقنيات البلوكشين الكامنة. ستسمح الإدارة اللامركزية من قبل أصحاب المصلحة في LRx بتحديثات بروتوكول العقد الذكي دون تعطيل ال dApps أو المستخدمين النهائيين ، أو الاعتماد بشكل كبير على تجريد العقود الذكية. في البداية ، سوف يتم ذلك من خلال عقد ذكي بسيط متعدد التوقيعات ، بهدف التقدم نحو نوع آلية DAO.

## 9- الحماية من الاحتيال و الهجوم

### 9.1 الوقاية من التداول المسبق

في منصات التداولات اللامركزية ، يتم التداول المسبق عندما يحاول شخص ما نسخ حل التداول لعفة أخرى ، و يجعله مقوضاً قبل المعاملة الأصلية الموجودة في تجمع المعاملات المعلق (mempool). ويمكن تحقيق ذلك عن طريق تحديد رسوم معاملات أعلى (سعر الجاز). المخطط الرئيسي للتداول المسبق في Loopring (واية بروتوكولات لمطابقة الطلب) هي عبارة عن سرقة الطلب:

## 10.2 عدم كفاية الرصيد

يمكن للمستخدمين الضاربين تسجيل وتوزيع الطلبات التي تكون قيمتها غير صفرية ، ولكن عنوانها لا يحتوي على رصيد في الواقع. يمكن للعقد أن ترافق وتلاحظ أن بعض الطلبات رصيدها الفعلي هو صفر ، وتحدد حالات الطلب وفقاً لذلك ، ثم تتجاهلها. يجب أن تقضى العقد الوقت لتحديث حالة الطلب ، ولكن يمكنها أيضاً اختيار تقليل الجهد من خلال ، على سبيل المثال ، وضع العناوين في القائمة السوداء وإفلات الطلبات ذات الصلة.

5. يستدعي منقب الحلقة دالة submitRing مع كافة المعلومات ، بالإضافة إلى authSignatures ليس جزءاً من المعاملة على السلسلة ، وبالتالي تظل غير معروفة لأطراف أخرى غير منقب الحلقة نفسه.

6. سيقوم بروتوكول Loopring الآن بالتحقق من صحة كل من authSignatureagainst مقابل authAddr لكل طلب ، ويرفض حلقة الطلب إذا كان أي authSignature مفقودة أو غير صالح.

والنتيجة هي الآن:

يبدأ بروتوكول Loopring كطبقة أساسية للتبادل اللامركزي. وبذلك ، فإن له تداعيات عميقة في كيفية تبادل الناس للأصول والقيمة. المال ، كسلعة وسيلة ، تسهل أو تحل محل تبادل المقايسة وتحل المصادفة المزدوجة لمشكلة الاحتياجات [22] ، حيث يجب أن يرغب اثنان من الأطراف المقابلة في بضاعة أو خدمة كل منهما. وبالمثل ، يهدف بروتوكول Loopring إلى الاستغناء عن اعتمادنا على مصادفة الرغبات في أزواج التداول ، وذلك باستخدام مطابقة الحلقة لتدوالات أكثر سهولة. وهذا مفيد لكيفية تبادل المجتمع والأسوق للعمل المشفرة اللامركبة ، والأصول القليدية ، وما وراء ذلك. في الواقع ، فإن البروتوكول التوافقى الذى يمكن أن يربط المتداولين على المال ، فإن العمل المشفرة اللامركبة تشكل تهديد لسيطرة دولة ما (المستهلكين / المنتجين) على نطاق واسع ، يشكل تهديداً نظرياً لمفهوم المال نفسه.

تشمل مزايا البروتوكول ما يلى:

- لا يتضمن إدارة الطلب خارج السلسلة والتسوية على السلسلة أي تضحيه في الأداء من أجل الأمان.
- زيادة السيولة بسبب تعدين الحلقة وتقاسم الطلبات.
- يحل التأليف المزدوج المشكلة الخبيثة في التداول المسبق التي يواجهها جميع الDEXs ومستخدميها اليوم.
- تتيح العقود الذكية العامة المجانية لأى dApp بناء أو التفاعل مع البروتوكول.
- يسمح التوحيد التقىسي بين المشغلين بتأثيرات الشبكة وتحسين تجربة المستخدم النهائي.
- تبقى الشبكة مرنة في تشغيل سجلات الطلب والتواصل.
- انخفاض العوائق التي تحول دون الدخول تعنى انخفاض تكاليف العقد المرتبطة بالشبكة والمستخدمين النهائيين.
- التداول مجھول مباشرة من محافظ المستخدم.

• يضمن توقيع الطلب (بواسطة المفتاح الخاص لعنوان المالك) بعدم إمكانية تعديل الطلب ، بما في ذلك authAddr

• توقيع منقب الحلقة (بواسطة المفتاح الخاص ب minerAddress ) ، إذا تم توفيره ، يضمن أنه لا يمكن لأى شخص استخدام هويته للتفقىب عن حلقة الطلب.

• يضمن authSignatures لا يمكن تعديل حلقة النظام بأكملها ، بما في ذلك minerAddress ، ولا يمكن سرقة أي أوامر.

يمنع التأليف المزدوج اختلاس الحلقة واحتلاس الطلب مع ضمان استمرار تسوية حلقات الطلب في معاملة واحدة. بالإضافة إلى ذلك ، تفتح ميزة "التأليف المزدوج" أبواباً أمام المرحلات لمشاركة الطلبات بطريقتين: المشاركة الغير قابلة للمطابقة والمشاركة القابلة للمطابقة. بشكل افتراضي ، تقوم Loopring بتشغيل نموذج OTC وتدعم فقط طلبات السعر المحدد ، معنى أنه يتم تجاهل الطوابع الزمنية للأوامر. ويعنى هذا أن تداول المسبق لا يؤثر على السعر الفعلى لذلك التداول ، ولكنه يؤثر على ما إذا تم تفيذه أم لا.

## 10- الهجمات الأخرى

### 10.1 هجوم Sybil او DOS

يمكن للمستخدمين الضاربين - بصفتهم أنفسهم أو بهويات مزورة - إرسال عدد كبير من الطلبات الصغيرة لمهاجمة عقد Loopring . ومع ذلك ، نظراً لأننا نسمح للعقد برفض الطلبات استناداً إلى معاييرها الخاصة - والتي قد تخفي أو تكشف عنها - فإن معظم هذه الطلبات سيتم رفضها لعدم تحقيق أرباح مرضية عند مطابقتها. من خلال تمكين المرحلات لإملاء كيفية إدارة الطلبات ، لا نرى هجوماً كبيراً جداً على النظام كتهديد.

## 12 - شكر وتقدير

نود أن نعرب عن امتناننا لمرشدينا ومستشارينا وللعديد من الناس في المجتمع الذين كانوا مرتاحين وسخين بمعرفتهم. على وجه الخصوص ، نود أن نشكر شو باي (من ChinaLedger) ، البروفيسور هابين كان، اليكس تشنج ، هونفهي دا ، بين كاو ، شياو تشوان وو ، تشنج وانغ ، وي بو ، نيان دونان ، جون شياو ، جيانغ تشيان ، جيانجزو شيانغ ، بيبنج جوه ، داهاي لي ، كفن لونج ، هواشيا شيا ، جون ما ، و انسيفالو باث لمراجعة وتقديم تعليقات حول هذا المشروع.

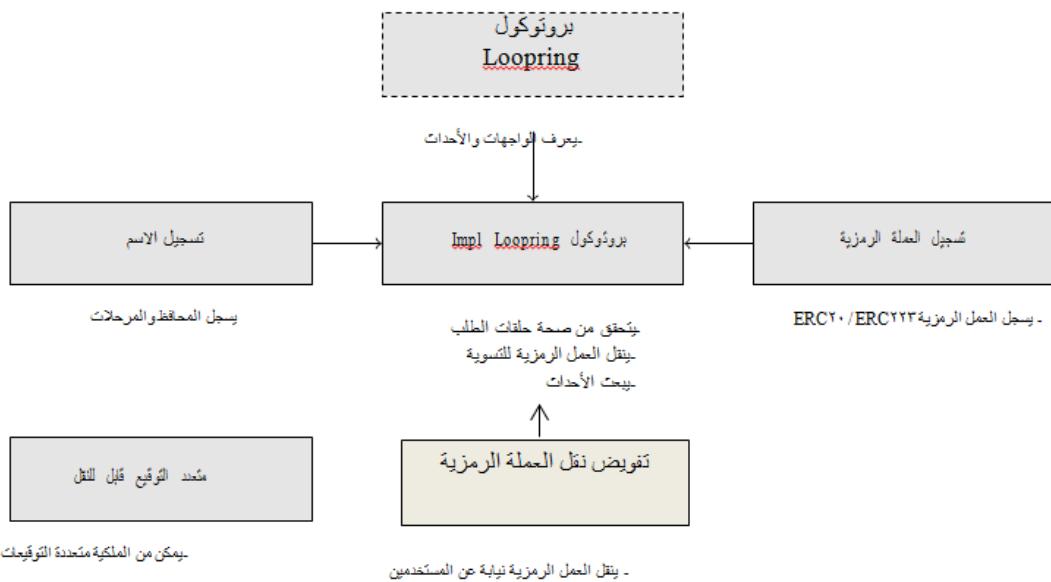
## المراجع

- [12] Rossella Agliardi and Ramazan Genay. Hedging through a limit order book with varying liquidity. 2014.
  - [13] Will Warren and Amir Bandeali. Ox: An open protocol for decentralized exchange on the ethereum blockchain, 2017.
  - [14] Iddo Bentov and Lorenz Breidenbach. The cost of decentralization. <http://hackingdistributed.com/2017/08/13/cost-of-decent/>, Accessed: 2018-03-05.
  - [15] Daniel Wang. Coinport's implementation of udom. <https://github.com/dong77/backcore/blob/master/coinex/coinex-backend/src/main-scala/com/coinport/coinex/markets/MarketManager.scala>, Accessed: 2018-03-05.
  - [16] Supersymmetry. Remarks on loopring. <https://docs.lopring.org/pdf/supersimmetry-loopring-remark.pdf>, Accessed: 2018-03-05.
  - [17] Fabian Vogelsteller. Erc: Token standard. URL <https://github.com/ethereum/EIPs/issues/20>, 2015.
  - [18] Chris Dannen. Introducing Ethereum and Solidity. Springer, 2017.
  - [19] Vitalik Buterin. Notes on blockchain governance, Accessed: 2018-03-05.
  - [20] Daniel Wang. Dual authoringlooprings solution to front-running URL <https://medium.com/loopring-protocol/dual-authoring-looprings-solution-to-front-running-d0fc9c348ef1>, 2018.
  - [21] Nick Szabo. Menger on money: right and wrong. <http://unenumerated.blogspot.ca/2006/06/menger-on-money-right-and-wrong.html>, Accessed: 2018-03-05.
- [1] Vitalik Buterin. Ethereum: a next generation smart contract and decentralized application platform (2013). URL {<http://ethereum.org/ethereum.html>}, 2017.
  - [2] Gavin Wood. Ethereum: A secure decentralised generalised transaction ledger. Ethereum Project Yellow Paper, 151, 2014.
  - [3] Patrick Dai, Neil Mahi, Jordan Earls, and Alex Norta. Smart-contract value-transfer protocols on a distributed mobile application platform. URL: <https://qntm.org/uploads/files/cf6d69348ca50dd985b60425ccf282f3.pdf>, 2017.
  - [4] Viktor Atterlohn. A distributed ledger for gamification of pro-bono time, 2018.
  - [5] Hernando de Soto. The Mystery Of Capital. Basic Books, 2000.
  - [6] Fabian Schuh and Daniel Larimer. Bitshares 2.0: Financial smart contract platform, 2015.
  - [7] Bancor protocol. URL <https://bancor.network/>, 2017.
  - [8] Yaron Velner Loi Luu. Kybernetwork: A trustless decentralized exchange and payment service. <https://kr.kyber.network/assets/KyberNetworkWhitepaper.pdf>, Accessed: 2018-03-05.
  - [9] Reuters. Coincheck. <https://www.reuters.com/article/us-japan-cryptocurrency-q-a/the-coincheck-hack-and-the-issue-with-crypto-assets-on-centralAccessed: 2018-03-05>.
  - [10] Robert McMillan. The inside story of mt. gox, bitcoins 460 dollar million disaster. 2014.
  - [11] Sylvain Ribes. Chasing fake volume: a crypto-plague, Accessed: 2018-03-10.

## الملحقات

### الملحق ( A ) على الاثريوم Loopring

#### A.1 العقود الذكية



الشكل ٧ : العقد الذكي

#### A.2 التطبيق

تم تطبيق العقد الذكي التالية على الشبكة الرئيسية ل ايثيريوم:

- LRC: 0xEF68e7C694F40c8202821eDF525dE3782458639f
- TokenRegistry: 0xa21c1f2AE7f721aE77b1204A4f0811c642638da9
- TokenTransferDelegate:  
0xc787aE8D6560FB77B82F42CED8eD39f94961e304
- NameRegistry: 0x0f3Dce8560a6010DE119396af005552B7983b7e7
- LoopringProtocolImpl:  
0xc80BbAb86cED62CF795619A357581FaF0cB46511
- TransferableMultsig: 0x7421ad9C880eDF007a122f119AD12dEd5f7C123B

To contact the translator  
[gmail.com@Mail](mailto:gmail.com@Mail): [mamoun.jamaladdin](mailto:mamoun.jamaladdin)  
[almize@Telegram](tg://resolve?domain=almize):