باسمه تعالی

گزارش پروژه زنجیره بلوکی

رمزنگاری پیشرفته

دکتر سلماسی‌زاده

اعضای گروه:

پوریا دادخواه

کیخسرو خسروانی

ابوالفضل یوسفی

پروژه انتخابی:

صرافی غیرمتمرکز (DEX)

UniSwap

فهرست مطالب

معرفی تجارت غیرمتمرکز و کابردها

با ظهور بلاکچین و فناوری غیر متمرکز و ورود قراردادهای هوشمند به آن‌ها، قابلیت ایجاد برنامه‌های متنوع زیادی با اهداف مختلف بر یک بستر غیرمتمرکز به‌وجود آمد که باعث شد سازمان‌ها، برنامه‌ها و سرویس‌های زیادی که دارای مشکلات جدی متمرکز بودن داشتند در این چارچوب پیاده شوند. یکی از این دسته‌ها، تجارت غیرمتمرکز یا Decentralized Finance (Defi) است که سرویس های تجاری مانند صرافی‌های غیرمتمرکز، بازارهای مالی و بورس، توکن‌های خرید و فروش اجناس (NFT) هستند که نسبت به مدل‌های متمرکز خود که به یک نهاد خارجی وابسته‌اند فواید زیادی دارند که مهم‌ترین‌ها را می‌توان اشاره کرد:

- هزینه‌های کمتر: پلتفرم‌های DeFi معمولاً به دلیل حذف واسطه‌ها، کارمزد کمتری نسبت به موسسات مالی سنتی دارند.
- شفافیت بیشتر: تراکنش های روی بلاکچین عمومی و شفاف هستند که به کاربران اجازه می‌دهد یکپارچگی سیستم را تأیید کنند.
- افزایش دسترسی: DeFi برای هر کسی که به اینترنت متصل است باز است و کاربران می‌توانند از هر کجای دنیا به خدمات مالی دسترسی داشته باشند.
- کنترل بیشتر بر وجوه: کاربران بدون نیاز به واسطه، کنترل کاملی بر وجوه خود دارند.

با این حال، DeFi دارای معایبی نیز است، از جمله:

- عدم قطعیت نظارتی: فقدان وضوح نظارتی می‌تواند منجر به عدم اطمینان و ریسک برای سرمایه‌گذاران شود.
- خطرات امنیتی: پلتفرم‌های DeFi در برابر هک‌ها و سوء استفاده‌ها آسیب‌پذیر هستند و تغییرناپذیری تراکنش‌های بلاکچین می‌تواند بازیابی وجوه از دست رفته را دشوار کند.
- موانع فنی: پیمایش پلتفرم‌های DeFi برای کاربران غیر فنی ممکن است دشوار باشد، که می‌تواند دسترسی آنها را محدود کند.

برخی از پروژه‌های مهم در فضای DeFi و اهداف آنها به شرح زیر می‌باشد:

- Uniswap : یک صرافی غیرمتمرکز است که بر روی بلاکچین اتریوم کار می‌کند. هدف آن ارائه یک روش ساده و کارآمد برای تجارت توکن های ERC-20 بدون نیاز به واسطه است. Uniswap از یک سیستم بازارساز خودکار (AMM) برای تسهیل معاملات و تشویق ارائه‌دهندگان نقدینگی برای عرضه نقدینگی به پلتفرم استفاده می‌کند.
- Aave : یک پلتفرم وام‌دهی غیرمتمرکز است که به کاربران امکان می‌دهد بدون نیاز به واسطه، ارزهای دیجیتال را قرض کنند و وام دهند. هدف آن فراهم کردن دسترسی کاربران به خدمات وام‌دهی شفاف، ایمن و غیرمتمرکز است.
- Compound: Compound یک پلتفرم غیرمتمرکز وام‌دهی است که به کاربران اجازه می‌دهد از دارایی‌های ارزهای دیجیتال خود سود کسب کنند. هدف آن ارلئه راه کارآمدتر و شفاف‌تر به کاربران برای کسب درآمد غیرفعال از دارایی‌هایشان است.
- MakerDAO : یک پلتفرم غیرمتمرکز است که به کاربران اجازه می‌دهد تا استیبل‌کوین‌ها را ایجاد و مدیریت کنند، که ارزهای دیجیتالی هستند که با ارزش دارایی‌های دنیای واقعی مرتبط هستند. هدف آن ارائه راهی پایدار و مطمئن برای نگهداری و انتقال ارزش به کاربران است.

صرافی های غیرمتمرکز و اهداف آن‌ها

پس از معرفی کلی تجارت غیرمتمرکز اکنون به یکی از شاخه‌های آن که صرافی غیرمتمرکز است می‌پردازیم.

DEX یا صرافی غیرمتمرکز نوعی از صرافی ارزهای دیجیتال است که به صورت غیرمتمرکز و بدون نیاز به واسطه یا مقامات مرکزی برای تسهیل معاملات فعالیت می کند. برخلاف صرافی‌های متمرکز (CEX) که برای مدیریت دفتر سفارش و مطابقت با خریداران و فروشندگان به یک اپراتور مرکزی متکی هستند، DEXها از مدل‌های سازنده بازار خودکار (AMM) یا مکانیسم‌های غیرمتمرکز دیگر برای فعال کردن معاملات همتا به همتا (peer to peer) استفاده می‌کنند.

علاوه بر این، DEX ها با هدف ارتقای شفافیت و باز بودن در اکوسیستم ارزهای دیجیتال هستند. با کار بر روی یک شبکه بلاکچین غیرمتمرکز، DEX ها یک دفتر کل عمومی ارائه می‌کنند که به کاربران امکان می‌دهد صحت و یکپارچگی معاملات و تراکنش‌ها را تأیید کنند.

DEXها برای برطرف کردن چالش‌های پیاده‌سازی غیرمتمرکز و غیرمعتمدتر یک صرافی و ارائه خدمات مشابه صرافی‌های سنتی از یکسری ابزارهای کلید استفاده می‌کنند که در ادامه به توضیح مختصر برخی مهم‌ترین ابزارها می‌پردازیم:

سازندگان بازار خودکار یا AMMها (Automated market makers): نوعی مدل صرافی غیرمتمرکز هستند که از «ربات‌های پول» الگوریتمی استفاده می‌کنند تا خرید و فروش دارایی‌های رمزنگاری شده را برای معامله‌گران فردی آسان کنند و قیمت یک ارز دیجیتال را بر اساس عرضه و تقاضای آن تعیین می کنند. به جای معامله مستقیم با افراد دیگر مانند یک دفتر سفارش سنتی، کاربران مستقیماً از طریق AMM معامله می‌کنند.

بازارسازان نهادهایی هستند که وظیفه دارند نقدینگی یک دارایی قابل معامله را در بورسی که ممکن است غیر نقدشونده باشد، فراهم کنند. بازارسازان این کار را با خرید و فروش دارایی‌ها از حساب‌های خود با هدف کسب سود، اغلب از اسپرد انجام می‌دهند - شکاف بین بالاترین پیشنهاد خرید و کمترین پیشنهاد فروش. فعالیت تجاری آنها باعث ایجاد نقدینگی می‌شود و تاثیر قیمت معاملات بزرگتر را کاهش می‌دهد.

قراردادهای هوشمند: DEX ها معمولاً در یک شبکه بلاکچین مانند اتریوم کار می‌کنند و از قراردادهای هوشمند برای خودکارسازی فرآیند معاملات استفاده می‌کنند. قراردادهای هوشمند برنامه‌هایی هستند که بر روی بلاکچین اجرا می‌شوند و با استفاده از قابلیت اجرای برنامه‌های سطح بالا مانند solidity که turing complete هستند به صورت توزیع شده در گره‌های مختلف شبکه اجرا شده و به طور خودکار معاملات را بر اساس قوانین و شرایط از پیش تعریف شده اجرا کنند.

اوراکل‌های غیرمتمرکز: DEX ها اغلب از اوراکل‌های غیرمتمرکز برای ارائه اطلاعات دقیق قیمت ارزهای دیجیتال استفاده می‌کنند. اوراکل‌ها خدماتی هستند که قراردادهای هوشمند مبتنی بر بلاکچین را با منابع داده خارج از زنجیره مانند فید قیمت ارزهای دیجیتال یا سایر شبکه‌های بلاکچین متصل می‌کنند.

کیف پول: DEXها برای تسهیل معاملات به کیف پول‌های ارز دیجیتال کاربران متکی هستند. کاربران معمولاً کیف پول خود را به پلتفرم DEX متصل می‌کنند تا معاملات را فعال کنند و این پلتفرم برای انجام معاملات با کیف پول تعامل می‌کند.

پروتکل‌های تعاملی: برخی از DEXها از پروتکل‌های قابلیت همکاری برای فعال کردن تجارت زنجیره‌ای اســـتفاده می‌کنند. این پروتکل‌ها به کاربران اجازه می‌دهد تا ارزهای دیجیتال را در شبکه‌های مختلف بلاکچین مانند اتریوم و بیت کوین معامله کنند.

در آخر باید به چالش‌ها و نقاط ضعف کلی این نوع صرافی‌ها در مقابل صرافی‌های عادی اشاره کنیم که علاوه بر نقاط ضعف کلی سامانه‌های DeFi که در بخش قبل به آن اشاره شد به صورت اختصاصی می‌توان به این موارد نیز اشاره کرد:

نقدینگی: DEX ها اغلب نقدینگی کمتری نسبت به CEX دارند، که می تواند خرید یا فروش ارزهای دیجیتال با قیمت مورد نظر را دشوارتر کند. این به این دلیل است که DEXها به جای یک اپراتور متمرکز، به کاربران برای ارائه نقدینگی به پلتفرم متکی هستند.

اجرای آهسته‌تر: DEXها ممکن است زمان تراکنش‌های کندتری نسبت به CEX داشته باشند، به خصوص در زمان‌های تقاضای بالا. این به این دلیل اســت که DEXها برای پردازش تراکنش‌ها به شــبکه بلاکچین متکی هســتند که تایید آن ممکن اســت بیشــتر از تراکنش‌های روی یک پایگاه داده متمرکز طول بکشد.

جفت‌های معاملاتی محدود: DEXها ممکن اســت انتخاب محدودتری از جفت‌های معاملاتی در مقایســه با CEX داشــته باشــند، که می‌تواند معامله با ارزهای دیجیتال کمتر محبوب یا خاص را دشوارتر کند.

پیچیدگی: DEXها ممکن است منحنی یادگیری تندتری داشته باشند و استفاده از آنها پیچیده‌تر از CEXها باشد، به خصوص برای کاربران مبتدی ارزهای دیجیتال.

کارمزدهای بالاتر: DEXها ممکن اســت در مقایســه با CEXها کارمزد بالاتری داشــته باشــند، به خصــوص برای کاربرانی که نیاز به تجارت حجم زیادی از ارزهای دیجیتال دارند. این به این دلیل است که DEXها برای پردازش تراکنش‌ها به هزینه‌های گاز پرداختی به شبکه بلاکچین متکی هستند که در زمان تقاضای بالا می‌تواند گران باشد.

پس از آشنایی با ساختار کلی صرافی‌های غیرمتمرکز به بررسی ساختار uniswap می‌پردازیم.

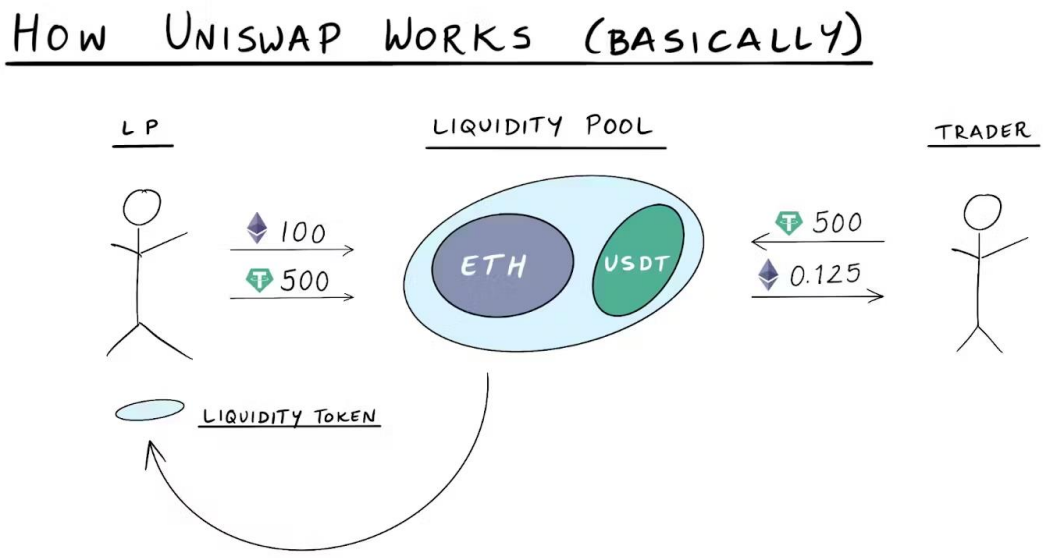
معرفی پروژه uniswap و طرز کار کلی

پروتکل Uniswap مجموعه ای از قراردادهای هوشـــمند اســـت که با هم یک AMM ایجاد می کنند. Uniswap همچنین یک رابط وب ایجاد کرد که به پروتکل Uniswap متصل می‌شود و دارای یک سیستم حاکمیتی مبتنی برتوکنUNI است که دارندگان توکن UNI دارای قدرت رای برای Uniswap توزیع شده به نسبت موجودی UNI کاربر هستند.

برخلاف صـرافی‌های متمرکز که از کتاب‌های سـفارش برای تجارت بین خریداران و فروشـندگان اسـتفاده می‌کنند، Uniswap از اسـتخرهای نقدینگی اسـتفاده می‌کند. ارائه‌دهندگان نقدینگی (LPs) توکن‌ها را به یک اسـتخر نقدینگی سـپرده‌گذاری می‌کنند و با کارمزدی که هنگام انجام معاملات ســـایر کاربران ایجاد می‌شـــود، پاداش دریافت می‌کنند. معامله گران هنگام انجام معاملات، کارمزدی را به ارائه دهندگان اسـتخر نقدینگی می پردازند. Uniswap هزینه ای برای لیست توکن ها دریافت نمی کند و نیازی به ثبت نام کاربر ندارد.

کد Uniswap منبع باز اســـت و می توان آن را برای ایجاد صـــرافی های جدید فورک کرد. یک پروژه قابل توجه مبتنی بر Uniswap اما با توکن و سیستم حاکمیتی متفاوت، Sushiswap است که در آگوست 2020 راه اندازی شد.

اکوسیستم Uniswap دارای سه نوع کاربر است: ارائه دهندگان نقدینگی، معامله گران و توسعه دهندگان.



ارائه دهندگان نقدینگی (LPs) تشـویق می شـوند که توکن های ERC-20 را به اسـتخرهای نقدینگی مشـترک با پتانسـیل کسـب کارمزد و سـایر مشـوق های نقدینگی کمک کنند. آنها در قراردادهای هوشـمند Uniswap توکن‌هایی را برای دور زدن نقدینگی یا قفل کردن ارائه می‌کنند و با توکن‌های نقدینگی که نشـــان‌دهنده مشـــارکت آنها در اســـتخر نقدینگی اســـت، پاداش می‌گیرند. توسـعه‌دهندگان می‌توانند از صـرافی‌های غیرمتمرکز (DEX)، مانند Uniswap، برای راه‌اندازی توکن‌های جدید به صـورت خودکار در حالی که غیرمتمرکز هسـتند، اسـتفاده کنند. صـدها برنامه، ابزار و کیف پول غیرمتمرکز مالی (DeFi) از Uniswap، از جمله پروژه‌های محبوب مانند Aave، inch1، و Compound استفاده می‌کنند. دارایی، مالیه، سرمایه گذاری.

رقبایی مانند Sushiswap و Coinbase پلتفرم‌های وام دهی، لانچرهای سکه و پلتفرم‌های NFT را ارائه می‌کنند که ممکن است Uniswap در آینده آنها را ادغام کند. در ژوئن 2022، یونی‌سوآپ Genie، یک گردآورنده بازار NFT را خریداری کرد.

Uniswap تاکنون 3 ورژن از پروژه را منتشـر کرده که به طور مختصـر تحولات آن را در هرمرحله و بهبودهای صـورت گرفته را بررسی می کنیم:

نسخه 1

Uniswap v1 اولین بار در نوامبر 2018 در شبکه اصلی اتریوم راه اندازی شد. Uniswap v1 فقط از جفت های تجاری ETH-ERC20 پشـتیبانی می کرد، بنابراین کاربران فقط می توانسـتند ETH را با یک توکن ERC-20 تعویض کنند. به عنوان مثال، اگر شخصی می خواست USDC را با DAI مبادله کند، باید USDC را با ETH تعویض می کرد و سپس به استخر ETH-DAI می رفت تا ETH را با DAI تعویض کند. برای انجام معاملات بین دو دارایی در Uniswap v1 معمولاً باید دو سوآپ انجام شود که منجر به کارمزد و لغزش بیشتر می شود.

برای کاربران در مقایسـه با یک تعویض. Uniswap v2 اجازه داد تا معاملات با اسـتفاده از یک سـوآپ هدایت شـوند و هزینه‌های بیش از حد لغزش و گاز پرداختی توسط کاربران در Uniswap نسخه 1 را حذف کرد.

Uniswap v1 همچنین مفهوم توکن های ارائه دهنده نقدینگی را معرفی کرد. هر ارائه‌دهنده نقدینگی مقداری توکن LP متناسب با درصـد نقدینگی که به اسـتخر نقدینگی اضـافه می‌کند، دریافت می‌کند. همه توکن های LP نشـان دهنده دارایی هسـتند که با تسـهیل معاملات در پروتکل با ارائه نقدینگی، کارمزدی را در Uniswap ایجاد می کند. همه توکن‌های LP نشـان‌دهنده کمک LP به استخر هستند و می‌توان آن‌ها را فروخت، معامله کرد، یا سوزاند تا توکن‌های سپرده‌شده را بازخرید کرد. علاوه بر این، هر معامله در Uniswap هزینه معاملاتی 0.3 درصـدی را به همراه داشـت و به‌طور خودکار به عنوان پاداشـی برای LPهایی که نقدینگی ارائه می‌کردند، ارسال می‌شد.

نسخه 2

Uniswap v2 در می 2020 راه اندازی شـد. با معرفی اسـتخرهای نقدینگی ERC20-ERC20 در Uniswap v1 بهبود یافت. پس از راه اندازی Uniswap نسـخه 2، شـخصـی که می خواهد USDC را با DAI مبادله کند، می تواند در صـورت وجود اسـتخر USDC-DAI، USDC را مستقیماً با DAI تعویض کند، به جای اجرای دو مبادله از USDC-ETH و سپس از ETH-DAI با استفاده از Uniswap v1.

علاوه بر این، Uniswap v2 یک اوراکل قیمت یا ابزاری را که برای مشاهده اطلاعات قیمت در مورد یک دارایی اسـتفاده می‌شـود، معرفی کرد. میانگین قیمت یک دارایی را در یک دوره بلوکی قیمت میانگین وزن زمانی یا(TWAP) با تقسـیم قیمت انباشـته بر طول مدت زمان محاسـبه می‌کند. این اوراکل های قیمت یک جزء حیاتی برای بسـیاری از برنامه های مالی غیرمتمرکز، از جمله موارد مربوط به مشتقات، وام، معاملات حاشیه و بازارهای پیش بینی هستند.

نسخه 3

Uniswap v3 در ماه مه 2021 راه اندازی شد و نسبت به Uniswap v2 بهبود یافته است. Uniswap v3 معرفی می کند:

نقدینگی متمرکز، به هر LP کنترل دقیقی بر محدوده قیمتی که سرمایه آنها تخصیص داده می شود، و

چندین ردیف کارمزد، به LPها اجازه می دهد تا به طور مناسب برای پذیرش درجات مختلف ریسک، جبران شوند.

Uniswap v3 همچنین روی بلاک چین های لایه 2 از جمله Polygon، Arbitrum و Optimismin علاوه بر اتریوم راه اندازی شد. Uniswap V3 اوراکل قیمت خود را بهبود بخشید و ادغام آن را در برنامه‌ها و خدمات آسان‌تر و ارزان‌تر کرد.

Uniswap نسخه 3 به LP ها سه سطح کارمزد جداگانه ارائه می دهد - 0.05٪، 0.30٪، و 1.00٪. LP هایی که از Uniswapv3 استفاده می کنند انعطاف پذیری بیشتری در حاشیه ریسک خود دارند. سطوح کارمزد کمتری برای موقعیت‌های با ریسک کمتر، مانند استیبل کوین‌ها، و سطوح کارمزد بالاتر برای موقعیت‌های پرخطرتر، جفت توکن‌های غیرهمبسته، انتظار می‌رود.

Uniswap v3 همچنین دارای هزینه های پروتکل مشابه Uniswap v2 است. کارمزدها به‌طور پیش‌فرض خاموش می‌شوند، اما می‌توانند توسط حاکمیت بر اساس هر استخر روشن شوند و بین 10 تا 25 درصد هزینه‌های LP تعیین می‌شوند.

علاوه بر این، مجوز Uniswap v3 به‌روزرسانی شد به طوری که ممکن است کد برای دو سال در پاسخ به فورک Sushiswap برای استفاده تجاری فورک نشود. تیم Uniswap بیانیه عمومی زیر را در مورد تصمیم به روز رسانی مجوز تجاری خود اعلام کرد.

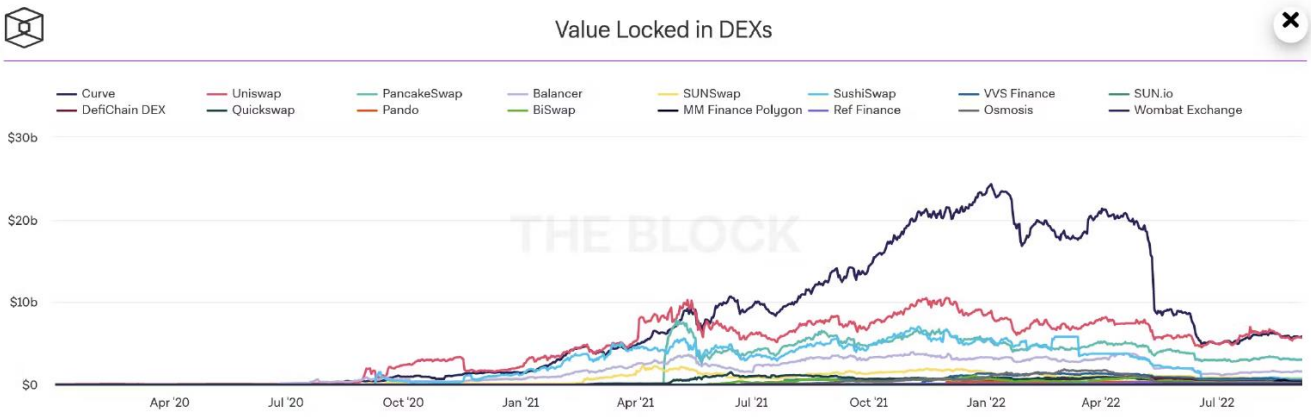
در آخر پیش از بررسی دقیق و جزئیات پروتکل و قراردادهوشمند آن، به بررسی بازار uniswap و حجم معاملات آن می‌پردازیم:

مشتری

مشتریان Uniswap شامل توسعه دهندگان، معامله گران و ارائه دهندگان نقدینگی هستند. توسعه‌دهندگان برنامه‌ها و ادغام‌های DeFi را بر اساس پروتکل Uniswap توسعه می‌دهند، معامله‌گران با استفاده از Uniswap ارزهای رمزنگاری شده مختلف را مبادله می‌کنند، و ارائه‌دهندگان نقدینگی در معاملات Uniswap کارمزد ایجاد می‌کنند. فراتر از توسعه دهندگان، معامله گران و ارائه دهندگان نقدینگی، Uniswap همچنین جامعه کاربران، توسعه دهندگان، طراحان و مربیان را از طریق رسانه های اجتماعی تقویت می کند. آنها از دیسکورد، توییتر، ردیت و انجمن حکومتی Uniswap برای موفقیت پروتکل Uniswap استفاده می کنند.

اندازه بازار

دو عامل اصلی تعیین کننده اندازه بازار برای Uniswap وجود دارد. اولین مورد TVL inDeFi و تولنایی DeFi برای دور کردن سهم بازار از محصولات مالی سنتی است. این ثانیه تعداد تراکنش هایی است که در DeFi مربوط به Uniswap انجام می شود. نمودار زیر مقدار قفل شده در DEX های محبوب را از جولای 2020 تا آگوست 2022 نشان می دهد.



Source: The Block

DeFi TVL

از سپتامبر 2022، بازار اتریوم دیفای حدود 16 میلیارد دلار TVL دارد.

آرتم تولکاچف، بنیانگذار و مدیر عامل BondAppetit و سـرمایه گذار DeFiHelperthe، معتقد اسـت که بازار DeFi می تواند بیش از 100 برابر اندازه خود نسبت به ارزش بازار 240 میلیارد دلاری خود در نوامبر 2021 تا سال 2026 رشد کند.

DeFi به طور بالقوه می تواند سـهم بازار را از امور مالی سـنتی بگیرد زیرا مردم به ایمنی قراردادهای هوشـمند اعتماد می کنند. در سال 2021، صنعت بانکداری مصرفی در سطح جهان 2.3 تریلیون دلار و بازار سرمایه 121 تریلیون دلار تخمین زده شد. تا ژوئن 2022، کل ارزش بازار شرکت های دولتی در سراسر جهان 105 تریلیون دلار برآورد شده است. اگر پروتکل‌های TVLof DeFi با برداشتن سهم بازار از محصولات مالی سنتی به رشد خود ادامه دهند، در آن صورت Uniswap احتمالاً شاهد افزایش قابل توجهی در بازار آدرس‌پذیر آن خواهد بود.

تراکنش‌های دیفای

عامل دیگری که بر اندازه بازار Uniswap تأثیر می گذارد، علاوه بر افزایش TVL DeFiand DeFi که سـهم بازار را از محصـولات مالی سـنتی دور می کند، تعداد فزاینده تراکنش هایی اسـت که در Uniswap انجام می شـود. هرچه تراکنش‌های بیشـتری در DeFi و از طریق Uniswap انجام شـود، اندازه بازار بالقوه Uniswap بزرگ‌تر می‌شـود، زیرا تراکنش‌های بیشـتر منجر به کسـب ارزش Uniswap از طریق کارمزدهایی می‌شود.
دو عاملی که بر تعداد فزاینده تراکنش‌ها و انباشـت ارزش در Uniswap تأثیر می‌گذارند، فعالیت ربات و سـایر محصـولات DeFi هستند که بر روی Uniswap ساخته شده‌اند.

فعالیت ربات

هرچه افراد بیشـتری در DeFi شـرکت کنند، توسـعه‌دهندگان را به سـمت ایجاد ربات‌های بیشـتری سـوق می‌دهد که قادر به انجام خدمات از طرف کاربران هسـتند مانند معاملات آربیتراژ، ترکیب خودکار موقعیت نقدینگی، و بسـیاری از اقدامات دیگر. در ژوئن

2022، 50.2 درصد از کل معاملات در Uniswap توسط ربات‌های حداکثر قابل استخراج (MEV) و 21.3 درصد دیگر از معاملات توسط «افراد ناشناس» که مشکوک به ربات هستند انجام شد.

پروتکل هایی با استفاده از Uniswap

DeFi شامل بسیاری از قراردادهای هوشمند جداگانه است که بر روی عملکردهای قراردادهای هوشمند دیگر برای ایجاد محصولات جدید و محصولات پیچیده تر DeFi ساخته شده است. یک تشبیه رایج برای توصیف DeFi این است که پروتکل‌ها/قراردادهای هوشمند DeFi مانند بلوک‌های لگو هستند که توسعه‌دهندگان می‌توانند آن‌ها را به هر طریقی که می‌خواهند برای ایجاد محصولات مالی پیچیده‌تر جمع‌آوری کنند. به عنوان مثال، پروتکل های محبوب مانند Curve Finance یک (DEX)و Compound Finance یک پلتفرم وام دهی قراردادهای هوشمندی دارند که توسط پروتکل های دیگر بر روی آنها ساخته شده است. Convex بر روی Curve Finance برای افزایش پاداش برای سهامداران CRV و ارائه دهندگان نقدینگی ساخته شده است، و Yearn Finance بر روی Compound Finance برای تجمیع بازده و ترکیب خودکار ساخته شده است.

سه نمونه از پروژه‌هایی که در بالای Uniswap برای بهینه‌سازی بازده برای تامین‌کنندگان نقدینگی و صرفه‌جویی در زمان برای کاربران ساخته می‌شوند عبارتند از Popsicle Finance، Gelato Network و .Mellow Protocolپروژه‌ها همچنین بر روی Uniswap ساخته شده‌اند تا با ارائه DEXaggregators با قابلیت مسیریابی معاملات، معاملات را برای کاربران خود بهینه کنند. از طرف کاربران برای دریافت بهترین قیمت ها. چهار پروژه قابل توجه با استفاده از Uniswap برای بهینه سازی معاملات برای کاربران خود عبارتند از 0x، Cowswap، Paraswap و 1 .inchدر ژوئن 2022، فعالیت Uniswap از جمع‌آوری‌کننده‌های DEX حدود 12 درصد از کل فعالیت تجاری در Uniswap را به خود اختصاص داد.

توضیح مدل سیستمی پروژه

Uniswap یک pool توزیع شـــده برای مبادله رمز ارز هســـت (ERC20 token) به کاربرها اجازه می‌دهد که دو رمز ارز را با یکدیگر در اینجا تبادل کنند.

به این صـــورت که مثلا یک فرد به نام liquidity provider از هر رمز ارز مثلا A و B دارد و یک liquidity pool می‌ســـازد. حال مثلا اگر کاربر Alice بخواهد رمزارز خود که از A هســـت را به B تبدیل کند در این pool این کار را انجام می‌دهد و 0.3% کارمزد برای انجام آن می‌پردازد. به این ترتیب معاملات انجام می‌شود و liquidity provider سود می‌برد.

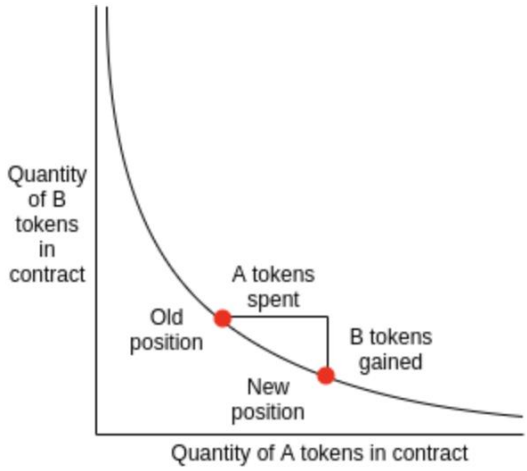
این پروتکل روی بلاک Ethereum قرار دارد و در سال 2018 به وسیله Hyden Adams ساخته شده است.

Uniwap از AMM و Auto-model market making اسـتفاده می‌کند که مدل قدیمی central order book را حذف می‌کند. Central order book نیاز به یک نهاد سوم شخص دارد که هر فرد به او بگوید چه مقدار از رمز ارز خود را به چه قیمت می‌خرد/ می‌فروشد و آن نهاد افراد را به هم وصل می‌کند.

در این مدل قیمت گذاری از یک رابطه ساده تبعیت می‌کند

Constant product automated market maker

$$x * y = k$$



For larger horizontal movements (coins spent) there are diminishing returns to the vertical movement (coins received).

که x مقدار coin نوع اول و y مقدار coin نوع دوم است. مثلا فرض کنید این pool برای Doge coin-Shiba باشد و LP مقدار x از Shiba و y تا از Doge coin در pool قرار می‌دهد.

حال چند سوال پیش می آید :

1) چگونه قرار است pool با xy=k کار می کند
2) اگر کل کوین های یکی از انواع تمام شود یا کم شود چه؟

برای پاسخ به سوال اول بیایید ببینیم محاسبات مقدار ارز مبادله شده و ارزش هر ارز بعد از تغییر مقدار هر ارز موجود در pool چگونه محاسبه می شود؟

یا به عبارت بهتر رابطه xy = k چگونه کار می کند.

این طرح یک constant value AMM است که مبنای کار uniswap می باشد.

فرض کنید در یک pool در ابتدا 50000 تا ارز A و 50000 تا ارز B داریم و ارزش هردو هم برابر است.

حال یک نفر درخواست مبادله 7000 تا از ارز B با ارز A را داشته باشد.

انگاه با کمک رابطه xy=k چون k ثابت است مقدار A نهایی را محاسبه می کنیم و با تفاضل از A کنونی مقدار A لازم برای برقراری این مبادله را حساب می کنیم.

$$57000 * x_A = 2500000000 = 50000 * 50000$$
$$x_A = 43859.65$$
$$50000 - 43859.65 = 6140.35 = \Delta x_A$$

پس ما 7000 تا از ارز B را با 6140.35 تا از ارز A مبادله می کنیم.

حال بررسی می کنیم که قیمت های نهایی در pool کنونی چند می شود:

قیمت اولیه هر دو در ابتدا با هم برابر و برابر 1 دلار بوده است.

قیمت ارز A حاصل :

$$\frac{initial\ amount}{final\ amount} * initial\ price = final\ price$$

$$\frac{50000}{43859.65} * 1 = 1.1399$$

پس قیمت جدید ارز A 1.1399 هست

قیمت ارز B :

$$\frac{50000}{57000} * 1 = 0.877$$

پس قیمت جدید ارز B 0.877 هست.

روند بالا نحوه محاسبه مقادیر لازم برای مبادله و قیمت های نهایی پس از مبادله در uniswap را بیان می کند.

جواب به سوال دوم راحت است پایداری این مدل بر مبنای equilibrium point هست
یعنی مثلا فرض کنید یک مقدار زیادی Doge coin را با shiba عوض کند در نتیجه مقدار آن کم و قیمت آن زیاد می شود و مقدار shiba زیاد و قیمت آن کم می شود. حال اگر قیمت shiba در این pool از قیمت shiba در بازار معاملات واقعی کمتر باشد افرادی برای کسب سود shiba را از این pool ارزان تر خریداری می کنند و در بازار اصلی می فروشند و این خرید و فروش را ادامه می دهند تا قیمت در این pool با قیمت در بازار اصلی یکی شود که نقطه حاصل همان نقطه equilibrium point است.

این کار سه فایده برای pool دارد :

1) قیمت ارز ها در pool با قیمت ارز ها در بازار اصلی برابر می شود
2) قیمت ارز ها در این pool به خاطر خاصیت equilibrium point دنبال کنند قیمت ها در بازار اصلی است و این oracle یک ارایه دهنده قیمت خوب برای دیگر مدل ها و smart contract ها هست.
3) پراکندگی دو رمز ارز متعادل می شود.

در اینجا ممکن است بگوییم liquidity provider ضرر کرده است زیرا مقداری از ارزهای خود را ارزان تر از قیمت بازار اصلی فروخته است. به این ضرر اسمی Impermanent loss گویند .

حال آیا واقعا ضرر کرده است؟
بیایید چک کنیم:
در ادامه یک مثال از نحوه کار کرد ریاضیات مدل uniswap را می بینیم.

فرض کنید یک liquidity provider به یک pool 50/50 که هنوز مقداری در آن تزریق نشده است به مقدار ارزش برابر از دو رمز ارز را تزریق کند:

قیمت DAI را 1 دلار و ETH را 500 دلار در نظر می گیریم
پس اگر او 10000 تا DAI و 20 تا ETH تزریق کند به هر دو رمز ارز مقدار ارزش برابر 10000 را داده است.
در این صورت

$$x = 20 \ , \quad y = 10000 \ , \ xy = 200000 = k$$

حال فرض کنید قیمت ETH در دنیای واقعی به 550 بالا رود.
افرادی برای سود کردن ETH با قیمت 500 را در اینجا با DAI مبادله می کنند تا قیمت ETH به 550 برسد:
بعد از رسیدن قیمت ETH به 550 مقدار های نهایی هر کوین به صورت زیر است:

10988.9 تا DAI

19.07 تا ETH

پس مقدار 0.93 تا ETH را با 488.09 تا DAI مبادله کرده است.

جمع ارزش ها در ابتدا 20000 بوده و در الان با مقادیر جدید و قیمت های جدید جمع ارزش به 20976.59 رسیده است.حال دیده می شود در اینجا liquidity provider مقدار 976.59 دلار سود کرده است ولی اگر دارایی خود را pool نمی کرد مقدار $10000 + 20 * 550 = 21000$ می داشت که با مقایسه با دارایی الان که 20976.59 بوده 23.41 دلار ضرر کرده است به این ضرر impermanent loss گویند.

که تازه بخشی از آن ضرر با fee دریافت کرده از مبادله پوشانده می شود

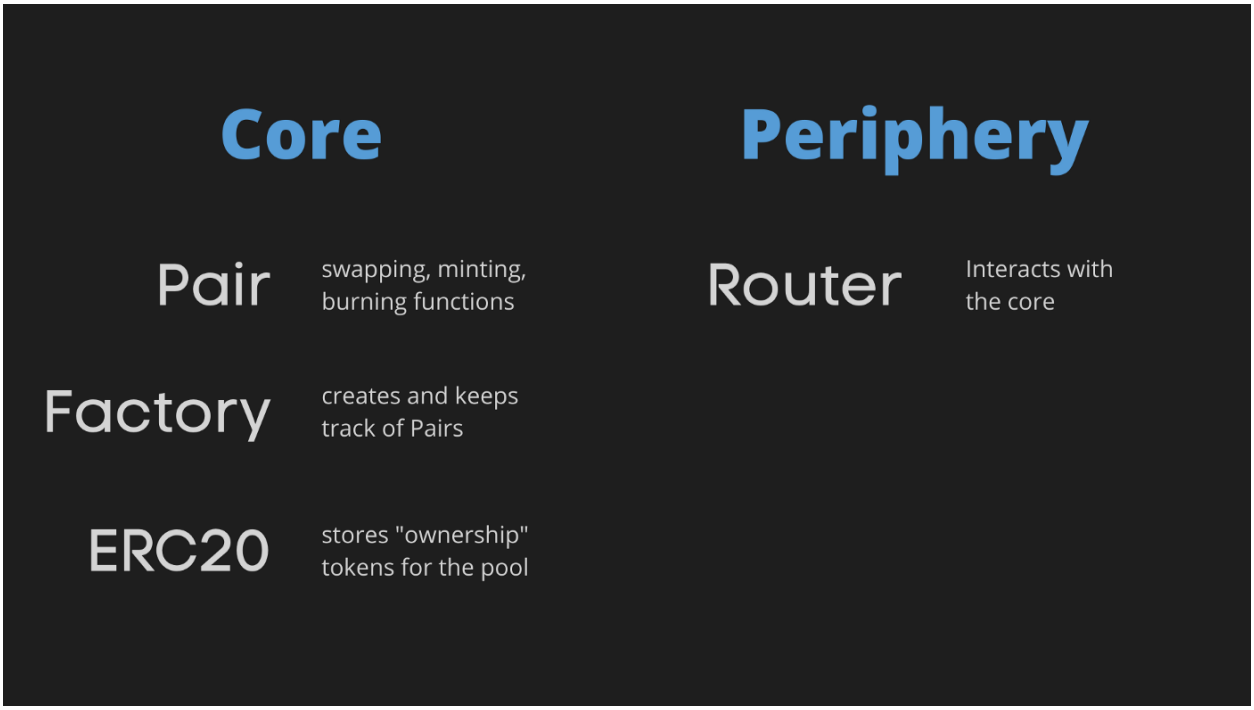
پس در واقع impermenant loss ضرری است که liquidity provider اگر دارایی خود را در pool نمی گذاشت و قیمت ETH ناگهانی بالا نمی رفت این ضرر را نمی کرد

بدون در نظر گرفتن این نوسانات بازار سود او از تبادل مالی حاصل می شود که به ازای هر تراکنش 0.3٪ از آن را به عنوان کار مزد دریافت می کند و در حالتی که $collected\ fee > impermenent\ loss$ باشد سود می برد.

ساختار پیاده سازی پروژه و قراردادهای هوشمند آن

Uniswap در مجموع دارای 4 تا smart contract هست

که در دو دسته core و periphery قرار دارند. مطابق شکل زیر :



Core مسئول ذخیره سرمایه ، توکن ها ، تولید pair ، عملیات مبادله رمز ارز ، انتقال جایزه ها و ... هست و شامل سه smart contract زیر است:

1) Pair
2) Factory
3) ERC20

که در ادامه به معرفی آنها می پردازیم:

Pair :

این قرارداد هوشمند وظیفه مبادلات رمزارزها و minting (واریز سرمایه به pool توسط LP) و burning (برداشت سرمایه از pool توسط LP) را بر عهده دارد.

کد تابع swap (مبادله رمز ارز) به صورت زیر است:

```
157
158     // this low-level function should be called from a contract which performs important safety checks
159     function swap(uint amount0Out, uint amount1Out, address to, bytes calldata data) external lock {
160         require(amount0Out > 0 || amount1Out > 0, 'UniswapV2: INSUFFICIENT_OUTPUT_AMOUNT');
161         (uint112 _reserve0, uint112 _reserve1,) = getReserves(); // gas savings
162         require(amount0Out < _reserve0 && amount1Out < _reserve1, 'UniswapV2: INSUFFICIENT_LIQUIDITY');
163                                                                                  A bunch of assertions
164         uint balance0;
165         uint balance1;
166         { // scope for _token{0,1}, avoids stack too deep errors
167         address _token0 = token0;
168         address _token1 = token1;
169         require(to != _token0 && to != _token1, 'UniswapV2: INVALID_TO');
170         if (amount0Out > 0) _safeTransfer(_token0, to, amount0Out); // optimistically transfer tokens
171         if (amount1Out > 0) _safeTransfer(_token1, to, amount1Out); // optimistically transfer tokens
172         if (data.length > 0) IUniswapV2Callee(to).uniswapV2Call(msg.sender, amount0Out, amount1Out, data);
173         balance0 = IERC20(_token0).balanceOf(address(this));
174         balance1 = IERC20(_token1).balanceOf(address(this));
175         }
176         uint amount0In = balance0 > _reserve0 - amount0Out ? balance0 - (_reserve0 - amount0Out) : 0;
177         uint amount1In = balance1 > _reserve1 - amount1Out ? balance1 - (_reserve1 - amount1Out) : 0;
178         require(amount0In > 0 || amount1In > 0, 'UniswapV2: INSUFFICIENT_INPUT_AMOUNT');
179         { // scope for reserve{0,1}Adjusted, avoids stack too deep errors
180         uint balance0Adjusted = balance0.mul(1000).sub(amount0In.mul(3));
181         uint balance1Adjusted = balance1.mul(1000).sub(amount1In.mul(3));
182         require(balance0Adjusted.mul(balance1Adjusted) >= uint(_reserve0).mul(_reserve1).mul(1000**2), 'UniswapV2: K');
183         }
184
185         _update(balance0, balance1, _reserve0, _reserve1);
186         emit Swap(msg.sender, amount0In, amount1In, amount0Out, amount1Out, to);
187     }
```

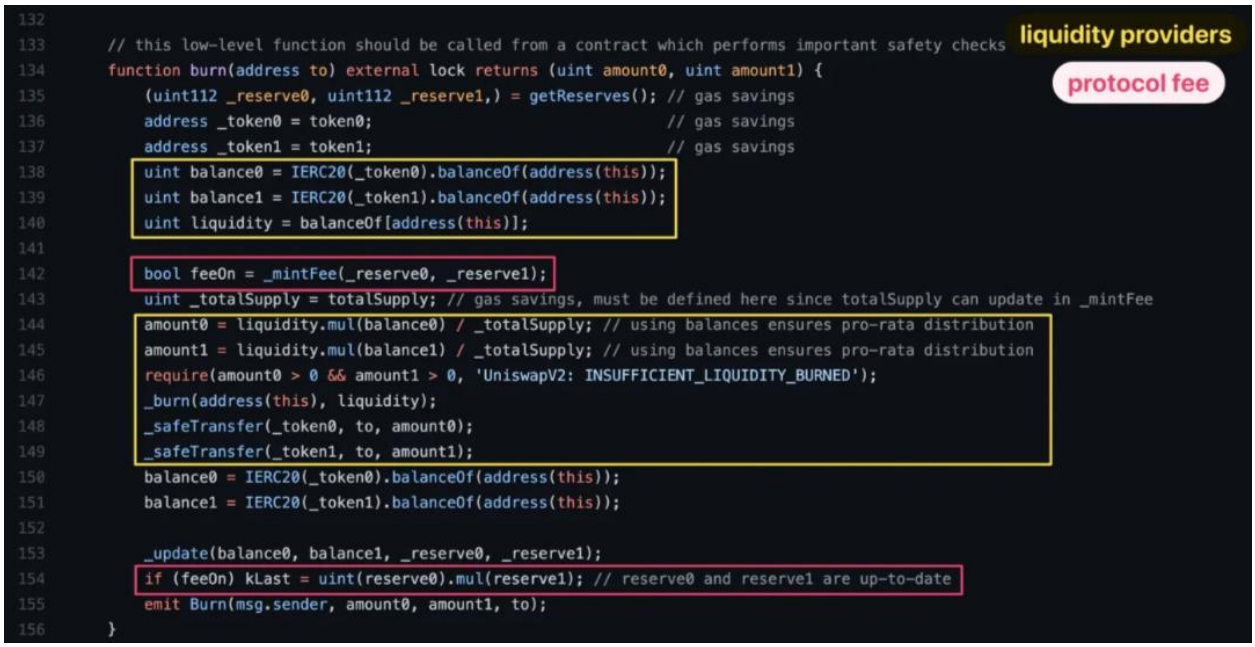
کد زیر تابع mint را نشان می دهد

```
109     // this low-level function should be called from a contract which performs important safety checks
110     function mint(address to) external lock returns (uint liquidity) {
111         (uint112 _reserve0, uint112 _reserve1,) = getReserves(); // gas savings
112         uint balance0 = IERC20(token0).balanceOf(address(this));
113         uint balance1 = IERC20(token1).balanceOf(address(this));
114         uint amount0 = balance0.sub(_reserve0);
115         uint amount1 = balance1.sub(_reserve1);
116
117         bool feeOn = _mintFee(_reserve0, _reserve1);
118         uint _totalSupply = totalSupply; // gas savings, must be defined here since totalSupply can update in _mintFee
119         if (_totalSupply == 0) {
120             liquidity = Math.sqrt(amount0.mul(amount1)).sub(MINIMUM_LIQUIDITY);
121            _mint(address(0), MINIMUM_LIQUIDITY); // permanently lock the first MINIMUM_LIQUIDITY tokens
122         } else {
123             liquidity = Math.min(amount0.mul(_totalSupply) / _reserve0, amount1.mul(_totalSupply) / _reserve1);
124         }
125         require(liquidity > 0, 'UniswapV2: INSUFFICIENT_LIQUIDITY_MINTED');
126         _mint(to, liquidity);
127
128         _update(balance0, balance1, _reserve0, _reserve1);
129         if (feeOn) kLast = uint(reserve0).mul(reserve1); // reserve0 and reserve1 are up-to-date
130         emit Mint(msg.sender, amount0, amount1);
131     }
```

liquidity providers

protocol fee

کد زیر تابع burn را نشان می دهد




```
132
133     // this low-level function should be called from a contract which performs important safety checks
134     function burn(address to) external lock returns (uint amount0, uint amount1) {
135         (uint112 _reserve0, uint112 _reserve1,) = getReserves(); // gas savings
136         address _token0 = token0;                                // gas savings
137         address _token1 = token1;                                // gas savings
138         uint balance0 = IERC20(_token0).balanceOf(address(this));
139         uint balance1 = IERC20(_token1).balanceOf(address(this));
140         uint liquidity = balanceOf[address(this)];
141
142         bool feeOn = _mintFee(_reserve0, _reserve1);
143         uint _totalSupply = totalSupply; // gas savings, must be defined here since totalSupply can update in _mintFee
144         amount0 = liquidity.mul(balance0) / _totalSupply; // using balances ensures pro-rata distribution
145         amount1 = liquidity.mul(balance1) / _totalSupply; // using balances ensures pro-rata distribution
146         require(amount0 > 0 && amount1 > 0, 'UniswapV2: INSUFFICIENT_LIQUIDITY_BURNED');
147         _burn(address(this), liquidity);
148         _safeTransfer(_token0, to, amount0);
149         _safeTransfer(_token1, to, amount1);
150         balance0 = IERC20(_token0).balanceOf(address(this));
151         balance1 = IERC20(_token1).balanceOf(address(this));
152
153         _update(balance0, balance1, _reserve0, _reserve1);
154         if (feeOn) kLast = uint(reserve0).mul(reserve1); // reserve0 and reserve1 are up-to-date
155         emit Burn(msg.sender, amount0, amount1, to);
156     }
```

Factory : برای تولید pair (زوج ارز) و نظارت بر آن استفاده می شود.

کد قسمت تولید pair آن به صورت زیر است

```
function createPair(address tokenA, address tokenB) external returns (address pair) {
    require(tokenA != tokenB, 'UniswapV2: IDENTICAL_ADDRESSES');
    (address token0, address token1) = tokenA < tokenB ? (tokenA, tokenB) : (tokenB, tokenA);
    require(token0 != address(0), 'UniswapV2: ZERO_ADDRESS');
    require(getPair[token0][token1] == address(0), 'UniswapV2: PAIR_EXISTS');
    bytes memory bytecode = type(UniswapV2Pair).creationCode;
    bytes32 salt = keccak256(abi.encodePacked(token0, token1));
    assembly {
        pair := create2(0, add(bytecode, 32), mload(bytecode), salt)
    }
    IUniswapV2Pair(pair).initialize(token0, token1);
    getPair[token0][token1] = pair;
    getPair[token1][token0] = pair; // populate mapping in the reverse direction
    allPairs.push(pair);
    emit PairCreated(token0, token1, pair, allPairs.length);
}
```

ERC20 :

زمانی که LP به pool بودجه می‌دهد به جای آن pool ownership token می‌گیرد که به واسطه آن‌ها پاداش کارمزد حاصل از مبادله را دریافت می کند.

و وقتی پول خود را باز بخواهد با تحویل دادن این token سرمایه خود را به همراه کارمزدهایی که از تراکنش ها گرفته به دست می آورد

کار ERC رسیدگی به این توکن ها ، تولید و سوزاندن آنها است. در حالی که ویژگی switchable protocol فعال باشد که الان نیست، 1/6 کارمزد به uniswap team و 5/6 به LP ها ولی الان کل آن به LP ها می‌رود.

Periphery :

کار این بخش فراهم کردن یک api برای کار با uniswap هست.

تنها از یک smart contract تشکیل شده است که آن Router است.

Router : در واقع API برای استفاده از توابع core است و نقش آن یک ارتباط دهنده بین کاربر و smart contract های core است.

Uniswap دارای یک interface در مدل خود است که آن در قرارداد هوشمند Router قرار دارد.

علاوه بر این مدل های مختلف دیگری خارج از uniswap هم interface مناسبی برای کار با uniswap ارایه می دهند از قبیل : zapper – 1inch - matcha

کاربرد دیگر uniswap، price oracle

به خاطر Arbitrage که در قسمت equilibrium loss بررسی کردم، قیمت‌ها در uniswap خیلی نزدیک به قیمت‌های بازار واقعی است. پس uniswap price oracle یک تخمین خیلی خوب از قیمت‌های واقعی بازار است.

امنیت پروژه و نتایج بررسی های audit آن

این قرارداد هوشمند تاکنون بسیار مورد audit قرار گرفته است. قبل از پرداختن به بررسی‌های صورت گرفته بر روی Uniswap به مبحث audit قراردادهای هوشمند می‌پردازیم.

Audit قراردادهای هوشمند در سراسر اکوسیستم DeFi برای ارائه یک بررسی عمیق از کد یک پروتکل، کمک به شناسایی اشکالات، کد ناکارآمد و راه‌حل‌هایی برای این مسائل استفاده می‌شود. این امری حیاتی است که قراردادهای هوشمند غیرقابل دستکاری باشند و audit را به بخشی کلیدی از فرآیند امنیتی هر پروژه بلاک چین تبدیل کند. audit کد برای هر برنامه‌ای مهم است، اما آنها به ویژه برای برنامه‌های غیرمتمرکز (dApps) مهم هستند، زیرا بلاک‌چین‌هایی که در بالای آن ساخته شده‌اند تغییر ناپذیر هستند. اگر آسیب‌پذیری کد منجر به از دست رفتن سرمایه کاربر شود، این وجوه قابل بازیابی نیست. تا به امروز بیش از 5 میلیارد دلار به دلیل هک ها در DeFi از دست رفته است.

Audit قرارداد هوشمند، شامل تجزیه و تحلیل دقیق کد قرارداد هوشمند یک پروتکل برای شناسایی آسیب‌پذیری‌های امنیتی، شیوه‌های کدگذاری ضعیف و کد ناکارآمد، قبل از شناسایی راه‌حل‌هایی است که این مشکلات را حل می‌کند. ممیزی ها به اطمینان از امنیت، قابلیت اطمینان و عملکرد برنامه های غیرمتمرکز در سراسر Web3 کمک می کند. در طول audit قرارداد هوشمند، تیمی از کارشناسان امنیتی، کد، منطق، معماری و اقدامات امنیتی برنامه را بررسی می‌کنند تا هر گونه مشکل احتمالی را با استفاده از فرآیندهای خودکار و دستی شناسایی کنند. آنها به طور خاص به دنبال هر منطقه ای از کد هستند که می‌تواند در برابر حملات مخرب آسیب پذیر باشد و همچنین هر منطقه ای را برای بهبود جستجو می‌کند.

کد قرارداد هوشمند در نهایت روی یک بلاک چین مانند Ethereum مستقر خواهد شد. پس از اجرای قراردادها، هر کسی – از کاربران نهایی گرفته تا عوامل مخرب – می‌تواند به آنها دسترسی داشته باشد، به همین دلیل است که همه آسیب‌پذیری‌ها باید قبل از راه‌اندازی یا به‌روزرسانی یک برنامه غیرمتمرکز برطرف شوند. پس از اتمام audit، گزارش خلاصه ای را منتشر می کنند که جزئیاتی را در مورد یافته‌های آنها، نحوه حل و فصل آنها و هر موضوع دیگری همراه با نقشه راه برای حل مسائل باقی‌مانده ارائه می‌دهد. پس از یک audit جامع قرارداد هوشمند، پروژه‌ها می توانند قراردادهای خود را با اطمینان از اینکه یکپارچگی برنامه امن است و وجوه کاربر محافظت می شود، مستقر کنند.

Audit قراردادهای هوشمند از انواع تکنیک ها و ابزارها برای کاهش نقاط ضعف و قوی تر کردن پروتکل ها استفاده می کند.

مرحله 1. جمع آوری اسناد

پروژه‌ای که در حال audit قرار می گیرد باید مستندات فنی شامل پایگاه کد، whitepaper، معماری و هر ماده مرتبط دیگر را در اختیار محققین قرار دهد. مستندات باید راهنمای سطح بالایی از هدف کد دستیابی، دامنه آن و اجرای دقیق را ارائه دهد.

مرحله 2. تست خودکار

آزمایش خودکار هر وضعیت احتمالی یک قرارداد هوشمند را بررسی می‌کند و هشدارهایی را در مورد مسائلی که می‌تواند عملکرد یا امنیت قرارداد را تضعیف کند، ارائه می‌دهد. محققین همچنین ممکن است آزمایش‌های یکپارچه‌سازی، تست‌های واحد بر روی عملکردهای فردی و تست نفوذ را انجام دهند که آسیب‌پذیری‌های امنیتی را بررسی می‌کند.

مرحله 3. بررسی دستی

تیمی از کارشناسان امنیتی هر خط کد را به دقت بررسی می‌کنند و خطاها و آسیب‌پذیری‌ها را شناسایی می‌کنند. در حالی که تست‌های خودکار برای شناسایی اشکالات در کد به خوبی کار می‌کنند، مهندسان انسانی توانایی بیشتری در تشخیص مشکلات مربوط به منطق قرارداد یا معماری، شیوه‌های کدنویسی ضعیف که از نظر فنی صحیح هستند و تست‌های خودکار را پشت سر می‌گذارند، فرصت‌های بهینه‌سازی gas و نقاط ضعف برای حملات متداول را دارند.

مرحله 4. طبقه بندی خطاهای قرارداد

هر خطا بر اساس شدت بهره‌برداری که می‌تواند فعال کند طبقه بندی می‌شود:

بحرانی (Critical) – بر عملکرد ایمن یک پروتکل تأثیر می‌گذارد.

عمده (Major) – خطاهای متمرکز و منطقی که می‌تواند منجر به از دست دادن بودجه کاربر یا کنترل پروتکل شود.

متوسط (Medium) – بر عملکرد یا قابلیت اطمینان پلتفرم تأثیر می‌گذارد.

جزئی (Minor) – کد ناکارآمدی که امنیت برنامه را به خطر نمی‌اندازد.

اطلاعاتی (information) – مربوط به سبک یا بهترین شیوه‌های صنعت.

مرحله 5. گزارش اولیه

محققین یک گزارش اولیه تهیه می‌کنند که نقص‌های کد و سایر مسائل را خلاصه می‌کند، همراه با بازخورد در مورد اینکه چگونه تیم پروژه می‌تواند آنها را برطرف کند. برخی از ارائه دهندگان خدمات قرارداد هوشمند تیمی از متخصصان دارند که به رفع هر اشکالی که پیدا شده است کمک می‌کند. با حل همه مسائل، پروژه‌ها می‌توانند اطمینان حاصل کنند که قراردادهای هوشمند آن‌ها برای استقرار آماده است.

مرحله 6. انتشار گزارش نهایی

محققین تمام یافته‌ها را در گزارش نهایی تفصیلی درج می‌کند و همه مسائل به عنوان حل‌شده یا حل‌نشده علامت‌گذاری می‌شوند. این گزارش به تیم پروژه داده می‌شود و اغلب به صورت عمومی منتشر می‌شود تا کاربران و سایر ذینفعان یک پروتکل از شفافیت کامل برخوردار باشند.

آسیب پذیری های رایج قرارداد هوشمند

در اینجا آسیب‌پذیری‌های رایجی که بخشی از چک‌لیست‌های بررسی قرارداد هوشمند فعلی هستند، آورده شده است.

مسائل مربوط به ورود مجدد (reentrancy attack)

هنگامی که یک تابع قرارداد هوشمند، یک قرارداد خارجی غیرقابل اعتماد را فراخوانی می‌کند، یک حمله ورود مجدد می‌تواند رخ دهد، که این قرارداد خارجی را قادر می‌سازد تا وجوه کاربر را تخلیه کند یا سایر اقدامات مخرب را با فراخوانی بازگشتی قرارداد اصلی انجام دهد.

سرریز و زیر جریان عدد صحیح (Integer Overflow and Underflow)

زمانی که یک قرارداد هوشمند یک عملیات محاسباتی را انجام می‌دهد که عددی بیش از ظرفیت ذخیره‌سازی کنونی است که منجر به محاسبات نادرست می‌شود، می‌تواند رخ دهد.

Frontrunning Opportunities

کد ساختار ضعیف می تواند اطلاعاتی را در مورد خریدهای آتی توسط dApp نشان دهد، که سایر کاربران می‌توانند به منظور قفل کردن سود تضمین‌شده با هزینه پروتکل، از آن استفاده کنند.

حمله تکرار (Replay attack)

حملات تکرار زمانی اتفاق می‌افتند که داده‌ها به‌طور مخرب به تأخیر افتاده یا تکرار می‌شوند تا گیرنده را خراب کنند، به‌ویژه در طول یک رویداد هارد فورک که در آن پیام‌های روی سیستم به‌روز شده برای استخراج وجوه از سیستم قدیمی استفاده می‌شود.

آسیب پذیری اعداد تصادفی (Random Number Vulnerability)

اگر یک dApp یک عدد تصادفی را با یک عدد شناخته شده عمومی، مانند هش بلوک، ایجاد کند، در معرض سوء استفاده قرار می‌گیرد، به همین دلیل است که بسیاری از پروتکل‌ها در Chainlink VRF برای تصادفی بودن استفاده می‌کنند.

خطرات متمرکز سازی

مرکزی‌سازی نقاط شکست واحدی را معرفی می‌کند که در صورت به خطر افتادن یک کلید خصوصی یا مشابه، می‌تواند امنیت یک پروتکل را تضعیف کند. قفل‌های زمانی و اعطای امتیازات به DAO تکنیک‌های رایجی هستند که با خطرات متمرکزسازی سروکار دارند.

نسخه کامپایلر آنلاک شده

تعدادی نسخه کامپایلر برای Solidity وجود دارد. dApps باید نسخه کامپایلر مورد استفاده خود را قفل کند تا کاربران نتوانند آن را با نسخه دیگری کامپایل کنند، که می تواند منجر به بایت کدهای مختلف و عوارض ناخواسته شود.

حال که با مفهوم audit یک قرارداد هوشمند آشنا شدیم، به audit قرارداد هوشمند خود (uniswap) می‌پردازیم.

هر نسخه از این قرارداد هوشمند (تاکنون 3 نسخه عرضه گردیده است) بسیار مورد بررسی واقع شده و با توجه به قدمت بیشتر دو نسخه ابتدایی، تعداد مشکلاتی که بر روی این دو شماره کشف شده‌اند زیاد است. علاوه بر کشف مشکلات به صورت عادی، این گروه نرم‌افزاری، یک طرح bug bounty در یک مقطع زمانی اجرا کرد که نتیجه آن رفع مشکلات بسیاری از این پروژه بوده است.

همانطور که گفته شد بر روی هر کدام از نسخه‌های این محصول بررسی‌های زیادی شده است. در این‌جا مشکلات بررسی شده بر روی نسخه سوم بیشتر مورد هدف هستند، هر چند برای آشنایی بیشتر با آسیب‌پذیری‌های نسخه‌های دیگر چند مثالی آورده شده است:

- **نسخه 1**

طی گزارشی که در ژانویه 2019 ارائه شده است، 7 تهدید از این نسخه استخراج شده که 1 عدد از این تهدیدات نیمه بحرانی قلمداد شده است و 6 تهدید دیگر جزئی محسوب شده‌اند. آسیب‌پذیری نیمه بحرانی بر روی تابع transferFrom ارائه شده و به این نحو است که مهاجم

می‌تواند در شرایط خاصی مقداری یا کل پول یک pool را بدزدد. تهدیدات دیگری نیز در این گزارش استخراج شده که شدت زیادی ندارند البته امکان وجود آسیب‌پذیری‌های جدی در گزارش‌های دیگر وجود دارد.

- **نسخه 2**

گزارشی در سال 2020 از بررسی نسخه دوم منتشر گردید که در این گزارش 3 باگ و 7 پیشنهاد برای تقویت ارائه شده است. لیست باگ‌ها به این شرح است:

1. Incompatible with token with fees on transfer
2. Fix to liquidity deflation introduces race condition
3. Integer overflow in sqrt

نتیجه این گزارش برطرف شدن تهدیدات و تقویت پروژه مدنظر است که در انتها استفاده از این محصول تأیید شده است.

- **نسخه 3**

نسخه سوم که نسخه مورد هدف ما است توسط آقایان Mikhail Vladimirov و Dmitry Khovratovich مورد بررسی قرار گرفته است و در تاریخ 2021 منتشر شده است. در این گزارش اشاره شده که مشکل مهمی پیدا نشده است و صرفا چند مشکل متوسط کشف گردیده است. بررسی‌هایی که انجام شده‌اند عبارتند از:

1. **ارزیابی عمومی کد:** کد از نظر وضوح، سازگاری، سبک و اینکه آیا از بهترین شیوه های کد قابل اجرا برای زبان برنامه نویسی خاص استفاده شده پیروی می‌کند، بررسی می‌شود.
2. **تجزیه و تحلیل استفاده از قرارداد:** استفاده از موجودیت‌های مختلف تعریف‌شده در کد مورد تجزیه و تحلیل قرار می‌گیرد. بررسی می‌شود که موجودیت‌ها در مکان‌های مناسب تعریف شده باشند و دامنه دید و سطوح دسترسی آنها مرتبط باشد. در این مرحله، معماری کلی سیستم و چگونگی ارتباط بخش‌های مختلف کد با یکدیگر درک شده است.
3. **تجزیه و تحلیل کنترل دسترسی:** برای آن دسته از قسمت‌هایی از کد که می‌توان به آن‌ها دسترسی خارجی داشت، اقدامات کنترل دسترسی تجزیه و تحلیل می‌شود. در این مرحله، نقش‌ها و مجوزهای کاربر و همچنین دارایی‌هایی که سیستم باید از آنها محافظت کند درک می‌شوند.
4. **تحلیل منطق کد:** منطق کد توابع خاص برای صحت و کارایی آنالیز می شود. بررسی می‌شود که کد واقعاً کاری را که قرار است انجام دهد انجام می‌دهد، الگوریتم‌ها بهینه و صحیح هستند و از انواع داده‌های مناسب استفاده می‌شود. همچنین بررسی می‌شود که کتابخانه‌های خارجی مورد استفاده در کد به‌روز و مرتبط با وظایفی هستند که در کد حل می‌کنند. در این مرحله همچنین ساختارهای داده مورد استفاده و اهداف مورد استفاده آنها مورد مطالعه قرار می‌گیرند.

تهدیدات متوسط در این گزارش عبارتند از:

1. **چک ناکافی:** این چک تضمین نمی‌کند که مقدار نقدینگی تخصیص‌یافته برای یک tick از حداکثر نقدینگی تجاوز نمی‌کند. به عنوان مثال، ممکن است یکی از تیک #1 تا تیک #5، maxLiquidity را به موقعیت از تیک #1 تا تیک #5 تزریق کند، و

شـخص دیگری می‌تواند نقدینگی بیشـتری را به موقعیت‌های تیک #2 تا تیک #4 تزریق کند. بنابراین برای تیک شـماره 3، کل نقدینگی بیش از حداکثر نقدینگی خواهد بود. این چک در فایل tick.sol به صورت زیر است:

```
require (liquidityGrossAfter<=maxLiquidity, 'LO');
```

2. **سرریز بافر**: سرریز بافر در یکی از خطوط کد رخ می‌دهد که البته این مشکل در نسخه منتشر شده رفع شده است. این مشکل در فایل FullMath.sol قرار دارد و این کد به صورت زیر است:

```
return mulDiv (a, b, denominator) + (mulmod (a, b, denominator) > 0 ? 1 : 0);
```

برای مقادیر a, b, denominator زیر سرریز بافر رخ می‌دهد:

a = 535006138814359, b = 432862656469423142931042426214547535783388063929571229938474969, denominator = 2

مشـکلات جزئی دیگر که مطرح شـده‌اند، مشـکلات جزئی هسـتند و به صـورت «نامگذاری بد»، «نقصـان کد»، «عدم مسـتند بودن»، «غیربهینه بودن» و ... دسـته‌بندی می‌شـوند. تعداد این مشـکلات جزئی حدود 159 عدد اسـت. عملا زمانی به یک مشـکل، جزئی گفته می‌شـود که امکانِ بودن یک آسیب‌پذیری وجود داشته باشد یا این که چیزی اضافه باشد و ایجاد دردسر کند و یا در موارد بسیار غیرمحتمل رخ دهد و نتیجه آن نیز بسیار اندک باشد.

در کل بررسـی‌های انجام‌شـده را می‌توان در حوزه امنیت قرارداد هوشـمند، دسـترس‌پذیری آن و کاربردی بودن تقسـیم‌بندی کرد و نتیجه هر audit بهتر شـدن پروژه نسـبت به audit قبلی اسـت، چرا که بعد از هر audit باید بررسـی‌هایی صـورت بگیرد تا مبادا خطایی اضـافه شـده باشد یا این که خطا به صورت ناقص از بین رفته باشد.

پروژه‌های رقیب uniswap

موفقیتuniswap باعث ایجاد رقبا و کپی‌های متعددی شده است، اما این رقابت در نهایت توسعه‌دهندگان را به افزودن ویژگی‌های نوآورانه برای جذب کاربران به سمت خود سوق می‌دهد. در اینجا چند رقیب Uniswap را بررسی خواهیم کرد که می توانید برای نیازهای معاملاتی غیرمتمرکز ارزهای دیجیتال خود از آنها استفاده کنید. بسیاری از آنها از مفهوم AMM مشابه با uniswap استفاده می‌کنند، اما برخی از آن‌ها رویکردهای منحصر به فرد خود را ارائه می‌دهند. چندین پروژه رقیب برای پروژه uniswap در زیر آمده‌اند که عبارتند از:

1inch

با وجود محبوبیتuniswap، نقدینگی کم همچنان یک مانع بزرگ است. نقدینگی کم به معنای نوسان و لغزش قیمت بالا (تفاوت بین قیمت معامله مورد انتظار و قیمت معامله انجام شده) است. نقدینگی نیز در چندین صرافی پراکنده شده است که این مشکل را تشدید کرده است. 1inch یک جمع کننده DEX است که این مشکل را برطرف می‌کند. با کنار هم قرار دادن چندین DEX در یک پلتفرم معاملاتی، 1inch نقدینگی را باز می‌کند، جهت تجارت را هدایت می‌کند و بهترین قیمت را پیدا می کند.

معاملات 1inch حتی می توانند به چند قسمت یا در چندین صرافی تقسیم شوند. 1inch در حال حاضر دارای حدود 12 صرافی از جمله Uniswap، Balancer و Oasis است. تجمیع‌کننده‌های DEX ابزارهای عالی برای کاهش لغزش هستند، اما کاربران اغلب به دلیل پیچیدگی معامله، کارمزدهای معاملاتی بالایی (کارمزد gas) را تجربه می‌کنند. 1inch در درجه اول با صرفه جویی در زمان به معامله‌گران کمک می‌کند زیرا آنها مجبور نیستند به صورت دستی هر صرافی را برای بهترین نرخ‌ها جستجو کنند .

یک جنبه بسیار منحصر به فرد 1inch این است که از سفارشات محدود نیز پشتیبانی می‌کند، که در فضای DeFi نادر است. شما به سادگی می‌توانید توکنی را که می‌خواهید خرج کنید و توکنی را که می‌خواهید دریافت کنید تنظیم کنید، قیمت دلخواه و مدت زمان سفارش را تعیین کنید. اگر قیمت توکن به قیمتی که شما مشخص کرده‌اید تغییر کند، 1inch به طور خودکار معامله را برای شما با بهترین نرخ های موجود در بازار انجام می‌دهد.

PancakeSwap

Pancakeswap یک صرافی غیرمتمرکز است که از زنجیره هوشمند Binance استفاده می کند. این پلتفرم از یک گروه ناشناس از توسعه دهندگان می‌آید. این پلتفرم توجه قابل توجهی را به خود جلب کرده است و به کاربران بسیاری در سراسر جهان خدمات ارائه می دهد.

Pancakeswap از زنجیره هوشمند Binance نسبت به بلاکچین ترجیح داده شده اتریوم استفاده می‌کند. این ویژگی به پلتفرم اجازه می‌دهد تا خدمات ارزان‌تری را به کاربران خود ارائه دهد. علاوه بر این، دارای حاکمیت اجتماعی است که به کاربران امکان farm نقدینگی را می‌دهد. به طور مشابه، Pancakeswap توکن‌ها را ارائه می‌دهد و ویژگی های خاص دیگری را در خود جای می‌دهد. Pancakeswap به شما امکان می‌دهد فوراً بین توکن‌های مختلف BEP-20 که روی بلاکچین هوشمند Binance کار می‌کنند، مبادله کنید. این پلتفرم همچنین دارای توکن مخصوص به خود به نام CAKE است که تعدادی عملکرد را در اکوسیستم PancakeSwap انجام می‌دهد.

SushiSwap

SushiSwap یک بازارساز خودکار است. این پلتفرم در طول سال‌ها از طریق صرافی غیرمتمرکز خود، کاربران زیادی را به خود جذب کرده است. به طور مشابه، قراردادهای هوشمندی را برای ایجاد بازار برای یک جفت توکن ارائه می‌دهد. SushiSwap عملیات خود را در سپتامبر 2020 به عنوان یک فورک Uniswap آغاز کرد. با این حال، این پلتفرم با هدف تنوع بخشیدن به بازار AMM و معرفی ویژگی های اضافی به پایگاه Uniswap است.

SushiSwap با وجود اینکه یک نام جدید در صنعت است، طی چند سال توجه قابل توجهی را به خود جلب کرده است. این پلتفرم از طریق شبکه FTX و برند Alameda Research فعالیت می کند. در اصل، Sushiswapبه عنوان یک AMM وجود دارد. این اجازه می دهد تا جابه‌جایی نقدینگی معاملات خودکار بین دو دارایی ارز دیجیتال انجام شود .

dYdX

dYdX یک پلتفرم معاملاتی غیرمتمرکز مبتنی بر اتریوم است که در تجارت مشتقات و مارجین تخصص دارد. همانطور که احتمالاً می دانید، تجارت در اوراق مشتقه مانند قراردادهای آتی و اختیار معامله در حال حاضر تحت سلطه صرافی های متمرکز مانند Binance Futures و Bybit است. dYdX در تلاش است تا با آوردن مشتقات در زنجیره، از طریق قراردادهای هوشمند اتریوم، این را تغییر دهد.

منحصر به فرد ترین ویژگی dYdX قراردادهای دائمی آن است که می‌توانید از آنها مانند قراردادهای دائمی که در صرافی‌های متمرکز محبوب هستند استفاده کنید. آنها به شما این امکان را می‌دهند که در موقعیت‌های اهرمی وارد شوید و در مورد حرکات قیمت ارزهای دیجیتال مختلف حدس و گمان بزنید و به شما فرصتی می‌دهند که به صورت لانگ یا کوتاه بپردازید.

از طریق همکاری باStarkware، dYdX قراردادهای دائمی خود را در راه حل مقیاس پذیری StarkEx لایه 2 اتریوم مستقر کرده است. StarkEx امکان پردازش تراکنش‌ها را بسیار ارزان‌تر از لایه 1 اتریوم می‌کند و dYdX را به گزینه‌ای مناسب برای معامله‌گران فعال تبدیل می‌کند. در حال حاضر، پلتفرم dYdX در حال راه‌اندازی توکن مدیریتی DYDX خود است که احتمالاً عملکرد پلتفرم را حتی بیشتر گسترش می‌دهد.

Binance DEX

بایننس یک صرافی ارزهای دیجیتال است که بیشترین حجم معاملات روزانه ارزهای دیجیتال را پردازش می‌کند. این شرکت فعالیت خود را در سال 2017 آغاز کرد. در ابتدا، این شرکت در چین مستقر بود. اما بعداً به دلیل افزایش مقررات دولت چین، مقر خود را منتقل کرد. در سال 2019، بایننس نیز وارد شبکه صرافی غیرمتمرکز شد.

Binance DEX پاسخ این شرکت به سایر صرافی های غیرمتمرکز است. این یک موتور تطبیق سفارش غیرمتمرکز است که بر روی فناوری زنجیره بایننس کار می کند. به طور مشابه، به کاربران این امکان را می دهد که یک کیف پول ایجاد کنند و توکن ها را از طریق شبکه تست Binance DEX مبادله کنند. این پلتفرم کاربران را قادر می‌سازد تا کلیدهای خصوصی خود را نگه دارند و کیف پول خود را مدیریت کنند. همچنین ویژگی‌های امنیتی مختلفی را ارائه می‌دهد و استفاده از آن آسان است.

Serum

Serum یک پروتکل و اکوسیستم تبادل غیرمتمرکز است. هدف این پلتفرم ارائه سرعت قابل توجه و هزینه‌های کم تراکنش برای امور مالی غیرمتمرکز است. در درجه اول، در Solana عمل می کند و کاملاً بدون مجوز است. Solana یک پلتفرم بلاکچین است که به دنبال مقیاس‌پذیری کاربر از طریق زمان تسویه تراکنش‌های سریع تر است. Serum به کاربران امکان خرید و فروش ارزهای دیجیتال را می‌دهد.

Serum از یک order book غیرمتمرکز استفاده می‌کند که توسط قراردادهای هوشمند اجرا می‌شود. هدف این ویژگی منعکس کردن مبادلات سنتی با تطبیق خریداران و فروشندگان است. علاوه بر این، به شرکت‌کنندگان در قیمت‌گذاری و اندازه سفارش انعطاف پذیری می‌دهد. Serum به کاربران امکان کنترل کامل بر معاملات خود را می‌دهد که اکثر کاربران ترجیح می‌دهند. این پلتفرم همچنین نوآوری هایی را هدف قرار می‌دهد که می تواند آن را از سایر پلتفرم ها مانند Uniswap و Sushi متمایز کند.

جزئیات رمزگذاری های استفاده شده در پروژه

پروتکل Uniswap از انواع اولیه رمزنگاری برای ایمن سازی عملیات خود استفاده می کند، از جمله توابع هش، امضای دیجیتال و رمزگذاری متقارن و نامتقارن:

1. توابع هش: برای تولید خروجی های با اندازه ثابت از ورودی های با اندازه متغیر استفاده می شود؛در Uniswap از تابع هش Keccak-256 برای اطمینان از یکپارچگی داده ها و تراکنش ها در بلاک چین برای تولید آدرس‌های قطعی، برای توکن‌ها و استخرها و همچنین برای تولید درختان Merkle برای ذخیره‌سازی کارآمد و اثبات گنجاندن یا عدم وجود داده‌ها استفاده می‌شود.

2. امضای دیجیتال : برای احراز هویت مبدا پیام و اطمینان از یکپارچگی آن استفاده می شود. Uniswap از طرح امضای ECDSA (الگوریتم امضای دیجیتال منحنی بیضوی) برای اعتبارسنجی تراکنش ها و اطمینان از اینکه تنها طرف های مورد نظر می توانند آنها را اجرا کنند، استفاده می کند.

3. تسهیم راز: از Shamir's Secret Sharing برای توزیع کلیدهای خصوصی قراردادهای هوشمند Uniswap بین توسعه دهندگان آن استفاده می شود تا از هر نقطه شکستی جلوگیری شود.

4. رمزگذاری متقارن: Uniswap از الگوریتم رمزگذاری AES برای اطمینان از محرمانه بودن و یکپارچگی داده های ارسال شده از طریق شبکه استفاده می کند. در API پروتکل Uniswap پیاده سازی شده است که از رمزگذاری HTTPS برای محافظت از داده ها در حین انتقال استفاده می کند.

هنگامی که کاربر از طریق مرورگر وب یا برنامه تلفن همراه خود با Uniswap تعامل می کند، تمام داده های منتقل شده بین دستگاه کاربر و سرورهای Uniswap با استفاده از HTTPS با رمزگذاری AES رمزگذاری می شود. این شامل اعتبار ورود کاربر، داده های تراکنش و سایر اطلاعات حساس است.

علاوه بر این، Uniswap از رمزگذاری برای ایمن سازی کلیدهای خصوصی کیف پول کاربران خود استفاده می کند. هنگامی که کاربران کیف پول خود را در Uniswap ایجاد می کنند یا به آن دسترسی پیدا می کنند، کلیدهای خصوصی آنها با استفاده از الگوریتم رمزگذاری AES-256-CBC قبل از ذخیره در دستگاه یا مرورگر کاربر رمزگذاری می شود.

5. رمزگذاری نامتقارن: Uniswap از پروتکل تبادل کلید (Elliptic Curve Diffie-Hellman) ECDH برای ایجاد یک کلید مخفی مشترک بین طرفین استفاده می کند بدون اینکه آن را در معرض استراق سمع قرار دهد.

در پروتکل Uniswap، ECDH به عنوان بخشی از فرآیند امضای تراکنش ها استفاده می شود. هنگامی که یک کاربر یک تراکنش را برای اجرای یک معامله در Uniswap امضا می کند، کلید خصوصی او با یک عدد تصادفی ترکیب می شود تا یک امضای منحصر به فرد ایجاد شود. سپس این امضا برای اجرای معامله به قرارداد هوشمند Uniswap ارسال می شود.

برای اطمینان از امنیت این فرآیند، از ECDH برای ایجاد یک کلید مخفی مشترک بین کاربر و قرارداد هوشمند استفاده می شود. این کلید مشترک برای رمزگذاری شماره تصادفی مورد استفاده در تولید امضا استفاده می‌شود و اطمینان حاصل می‌کند که نمی‌تواند توسط شخص ثالث رهگیری یا دستکاری شود.

به طور کلی، Uniswap از طیفی از رمزنگاری های اولیه برای اطمینان از امنیت،یکپارچگی، حریم خصوصی و کارایی پروتکل خود و تراکنش های انجام شده بر روی پلت فرم خود استفاده می کند.

کارهای آینده و ایده های بهبود پروژه

در اینجا چند ایده بالقوه برای بهبود پروژه Uniswap وجود دارد که چند مورد را صـــرفا اشـــاره می‌کنیم و یک مورد را با جزئیات بیشتری شرح می‌دهیم:

کاهش هزینـه ها: Uniswap می تواند راه هایی را برای کاهش هزینـه هایی که کاربران برای تجارت روی پلتفرم می پردازند، بررسی کند. این می تواند شامل بهینه سازی مصرف گاز یا کاوش در ساختارهای جدید هزینه ای باشد که برای کاربران مقرون به صرفه تر است.

بهبود رابط کاربری: رابط کاربری Uniswap را می توان بهبود بخشـید تا کاربر پسـندتر و در دسـترس طیف وسـیع تری از کاربران باشد. این می تواند شامل ساده کردن فرآیند تجارت، ارائه منابع آموزشی بیشتر یا بهبود دسترسی به تلفن همراه باشد.

کاوش راه‌حل‌های لایه ۲: Uniswap می‌تواند راه‌حل‌های لایه ۲، مانند زنجیره‌های جانبی یا کانال‌های حالت، را برای بهبود مقیاس‌پذیری و ســـرعت پلتفرم بررســـی کند. این می تولند به کاهش زمان تراکنش و هزینه های گاز کمک کند و در عین حال امنیت و عدم اعتماد پلت فرم را حفظ کند.

ضرر LP ها در مواقع معادل سازی قیمت:

بر اساس پروتکل uniswap و اعمال قیمت گزاری بر اساس عرضه و تقاضا، عده‌ای میتوانند با عرضه مقدار زیادی از یک ارز باعث ایجاد یک حباب از ارز دیگر شده و قیمت غیر واقعی نسبت به دنیای بیرون ایجاد کرده و سود کنند که در اینجا ضرر اصلی به LP های میرسد. ایده حل این مشکل را به این صورت بیان می‌کنیم:

اضافه شدن یک تابع به قرارداد هوشمند router که از یکی از بازارهای اصلی قیمت دو ارزی که در حال استفاده است را با قیمت مدل خود چک کند و اگر اختلاف این دو خیلی زیاد است کار زیر را انجام دهد: علاوه بر استخر اصلی، خود provider یک pool دیگر مجازی ایجاد کند که هرموقع قیمت از یک ترشهولدی بیشتر شد، خودش از خودش آن ارزی که قیمتش بیشتر است را خرید کند که ضرر نکند.

جمع بندی:

در این گزارش پس از آشـــنایی با مفهوم تجارت غیرمتمرکز و امکانات و فوایدی که فراهم می‌آورد به دســـته خاصـــی از آن یعنی صـرافی‌های غیرمتمرکز پرداختیم و سـاختار کلی و چالش‌های آن‌ها را بیان کردیم. سـپس یکی از محبوب‌ترین صـرافی‌های فوق که uniswap بوده را معرفی کرده و ضـمن آشـنایی با پروژه و هدف آن و ابزارها و مشـتریانی که دارد حجم بازار فعلی و مسـیر پیش رو را نشـان دادیم. در ادامه نیز به توضـیح دقیق پروتکل و مدل سـیسـتمی آن و پیاده‌سـازی عملی این پروتکل با قراردادهای هوشـمند پرداخته و امنیت آن را نیز با نتایج audit های انجام شـده روی آن بررسـی کردیم. در آخر به مقایسـه این پروژه با پروژه‌های رقیب پرداخته و ایده‌هایی برای بهبود این پروتکل ارائه کردیم.