4a Coin: Web tabanlı kriptopara ağı

Yasin Aktimur

Özet~

Bitcoin, ethereum ve litecoin gibi birinci nesil kritoparalar, para basma işini GPU ve CPU gibi makinelerin güçlerine göre dağıttıklarından dolayı gerçek zamanlı olan ve aslında anında işleyebilecekleri ödemeleri hafıza havuzu (mempool)'da biriktirip

belirledikleri zorluk değerine uygun hash özetini bulana kadar nonce adında bir rakamı rassal veya sürekli artacak şekilde değiştirip tekrar tekrar deneyerek madencilik yapıyorlar. Bu çözüm, bir dijital parayı değer saklama aracı olarak kullanmak isterseniz dahice.

Fakat bu dijital paraları kahve ücretini ödemek veya internetten bir ürün satın almak üzere kullanılmaya kalktığınızda hem bu madencilere komisyon ödemek, hemde uzun süre beklemek zorunda kalmak bir dezavantaj olarak göze çarpıyor.

Paypal veya Western Union gibi büyük ödeme çözümlerinin gücü bildiğiniz üzere serverlarından ve yazılımlarından geçiyor.

Bizde web servisi olarak çalışan, merkezi olmayan, blockchain ile korunan bir uçtan uca bağlı (p2p) kripto para üzerine çalışmaya başladık. Böyle bir durumda sistemin ayakata kalabilmesi ve güvenliğin sağlanması için server maliyeti ödemek yerine sistemde node olan server'lar madencilik yapmak yerine sadece server oldukları için 44 saat boyunca online kalmak şartıyla ödeme alabilecekleri bir sistem kurduk. İnsanlar boşuna uğraşmak yerine gerçekten işe yarar bir hizmet verdikleri için bunun karşılığını alıyorlar.

Server olarak kaldıkları sürece ödüllendirildiklerinden dolayı insanların serverlarına bu sistemi kurmaları için bir neden doğmuş oluyor.

İnsanlar uçtan uca bir ödeme gönderdiğinde serverlar bu ödemenin zaman, gönderdici, alıcı, miktar, bir önceki işlemin özeti ve gerçektende göndericinin gönderdiğini belirten bir dijital imzanında içinde bulunduğu verileri zorluk gözetmeksizin sadece sonu 4A ile bitecek şekilde özetini bularak (Tüm sistemlerin ortak bir noktada buluşması ve ödemeleri doğrulayabilmesi için blockchain mimarisinin bir gerekliliği olduğu için bu özeti buluyorlar.) veritabanlarına eklerler. Herhangi bir şekilde blocksize, mining gibi kavramlar söz konusu olmadığı için kayıt işlemi anında gerçekleşir. Aynı saniye içinde 100 adet işlem gerçekleştiği taktirde oluşabilecek karmaşayı engellemek için instagram ve mozilla gibi şirketlerinde kullandığı celery project adında bir görev kuyruğu yönetim kütüphanesinden yararlanırız.

Güvenlik~

Eğer siteye girdiğinizde fark ettiyseniz visa'dan daha güvenli dedim.

Bu bir çokları için iddialı bir cümle olarak gözükebilir fakat bir internet sitesinden bir ürün aldığınızda aslında ne yaptığınızı iyi biliyorsanız aldığınız risklerin farkındasınızdır.

Risk & Sorun~

Ödeme formunda bulunan kısma kredi kartı, cvv gibi bilgileri yazdığınız zaman ağda bulunan kötü niyetli kişiler bu formlardan gönderdiğiniz verileri okuyabilirler. Sırf bu yüzden ödeme çözümleri sizden sitenize SSL sertifikası eklemenizi isterler. bu SSL sertifikaları aslında kriptoloji kullanarak verilerinizi saklar ve alıcıya ulaşana kadar korurlar. Peki satıcı kötü niyeliyse ne olacak?

Çözüm~

4acoin Elliptic Curve Digital Signature Algorithm (ECDSA) dijital şifreleme algoritmasını kullanarak ödemelerinizi şifreleyerek ağa dijital şifrenizi vermek yerine ödemeyi yapmak istediğinizi teyit ettiğiniz bir imza gönderir. Bu aslında satoshi nakamoto'nun fikri ve bitcoin'de

p2pkh olarak geçiyor. Böylece aslında 4acoini kredi kartına benzetirsek kimse sizin kredi kartı bilgilerinizi hiç bir zaman görmemiş oluyor. Ayrıca SSL belgesinede artık ihtiyaç yok çünkü ağınızı dinleyebilen kötü niyetli bir kullanıcı'nın ödeme imzanızı görmesi hiç bir şeyi değiştirmez ki bu zaten her yerde açık olarak paylaşılır. Böylece ilk olarak güvenlik ve SSL gibi sertifika sorunlarından kurtulmuş olduk bile.

Komisyon Çözümü~

Yerel bir ödeme çözümünden faydalandığınız zaman %5 ile % 20 arasında satışlarınız üzerinden çalıştığınız firmalara komisyon ödemek zorunda kalırsınız. 4A Coin'de kullandığınız coinleri zaten node'lar ürettiği için sizden komisyon istemezler bu sayede ne transactionlar için

nede herhangi başka bir nedenle komisyon ödemezsiniz.

Cüzdanlar~

wallet_id ~

4A01eaedb37fc09fdb94c6d632adf9f63d

private_key ~

cbc949239a333559f5dd8b0b5cf3d32923c2cab37c2bde9c8042a3dafe59a6b9

Cüzdanınız aslında bir ecdsa anahtar çiftidir. En başta bu iş için RSA'i kullandık fakat RSA'de keylerin çok fazla uzun olması ve gerekesiz yer kaplaması yüzünden ecdsa'ya geçiş yapmak zorunda kaldık.

Kullanıcılar'ın Public keyleri, private key'leri ve public keylerinin işlenmesi ile oluşmuş wallet adresleri bulunur. Public key sistemde kullanıcılar tarafından gözükmez. Onun yerine public keyin işlenmesi ile oluşan bir özetin kısa versiyonunu cüzdan olarak görürler.

```
def generate_wallet_from_pkey(public_key):
    binmnmn = public_key.encode('utf-8')
    first_step = 34 - len(settings.CURRENCY)
    wallet_id = hashlib.sha256(binmnmn).hexdigest()
    wallet_id = wallet_id[-first_step:]
    wallet_id = "".join((settings.CURRENCY, wallet_id))
    return wallet_id
```

Bu fonksiyon basitçe verilen public keyin sha256 özetinden bir cüzdan yaratır. Public keyler dijital imzaların onayı için gereklidir bu yüzden işlemlerde public keyi saklamak gerekir.

Ödemeler~

Bir kullanıcı ödeme yaptığında, ödeme zamanı (epoch formatında & GMT), gönderici adresi, alıcı adresi, bir önceki işlemin özeti ve gönderilen miktar bir sözlüğe aktarılır. Bu sözlük farklı bilgisayarlarda farklı şekilde dizilebilir, bunu engellemek için bu sözlüğün içeriğinin a'dan z'ye düzenli bir şekilde ve herkeste aynı sonucu verecek şekilde dizilmesi gerekir.

```
data = collections.OrderedDict(sorted(data.items()))
```

Yukarıdaki kod ile global bir şekilde doğru çalışabilen stabil bir sözlük yaratabiliyoruz.

Son olarak bu sözlüğün özetini alarak veritabanına kayıt edip bunu kayıt ettiğimizi diğer serverlara broadcast yani yayın yapıyoruz.

Peer to Peer~

Sistemin p2p olmasını sağlamak için tcp portundan ve web socket teknolojisinden yararlanırız. Scriptin çalışması için python3 kullanmak gerekir. Çünkü realtime işlemleri sağlayabilmek için twister matrix kütüphanesini ile Autobahn python kütüphanelerini kullanırız. Global olarak kullandığımız port 9000. porttur.

Realtime işlemlerde 3 farklı broadcasting tipi bulunur birincisi ***"hey merhaba ben yeni bir node'um lütfen beni ağınıza ekleyin"*** diğeri ise ***"hey merhaba ben yeni bir işlemim lütfen beni doğrulayın"*** şeklindedir. Bu server tarafında ayrıştırılarak gerekli işlemler yapılır. 3. broadcasting tipi proof of cloud kısmında açıklanacaktır.

Madencilik~

Toplamda 300.000.000 Premined olmak üzere toplamda 450.000.000 adet 4a Coin üretilecektir. 150 milyon adeti Proof of Cloud

olarak adlandırılan yöntem ile kazılacaktır.

Proof of Cloud~

Proof of cloud yada POC madencilik yerine server olarak süreye dayalı bir kazanç yöntemi anlamına gelir. Her bir node 44 saat online kalarak 44 saatin sonunda "hey merhaba ben 44 saattir online'ım son işlemlerimi inceleyerek online olup olmadığımı yani veritabanımı kontrol edin." diye bir mesaj gönderir. eğer son 44 saattir işlemleri onayladığını kanıtlarsa ödülü almaya hak kazanacaktır.

Celery & Redis~

Sistemde belirli işlemlerin belirli saatlerde tekrar etmesi için görevleri otomatik olarak kontrol etmeye yarayan bir kütüphane olan Celery'i kullanırız. Celery çalışmak için redis'e ihtiyaç duyar. REDİS saf sürüm olarak Linux üzerine yazılmış açık kaynaklı NoSQL(NoSQL : Şemasal olarak "ilişkisel olmayan" verileri depolayan veritabanı sistemlerine verilen addır.NoSQL tam anlamıyla "SQL Kullanılmıyor" değil "Sadece SQL Kullanılmıyor" anlamına gelen "not-only-SQL" anlamında kullanılıyor.) yazılımıdır.

Sistemin ayakta kalması~

Sistemin yüksek yükleri kaldırabilmesi ve dayanıklılığını sağlamak için Python WSGI HTTP Server olan Gunicorn 'Green Unicorn' ve Nginx kullanırız; Nginx; yüksek eş zamanlı çalışma kabiliyeti, yüksek performans ve düşük hafıza kullanımına odaklanılarak tasarlanmış bir Web sunucusudur. Aynı zamanda ters vekil sunucusu, yük dengeleyici ve HTTP ön belleği olarak da kullanılabilir.

Tüm bu sistemleri sürekli ayakta tutabilmek için aslında supervisord kullanıyorduk fakat supervisord python3 ile çalışmadığı için mozilla vakfının geliştirdiği circusd kullanmaya başladık ve supervisiord'dan daha fazala memnun kaldık. Kurulumu ve kullanımı çok daha kolay yalnız çalışmak için tornado framework'e ihtiyaç duyuyor ve tornado'nun 4.5.3 sürümüyle doğru çalışıyor. Zaten requirements.txt 'de otomatik olarak bu versiyon kuruluyor fakat bu detayı bilmeniz önemli.