Identity

身份

What is an Identity?

什么是身份

The different actors in a blockchain network include peers, orderers, client applications, administrators and more. Each of these actors has an identity that is encapsulated in an X.509 digital certificate. These identities really matter because they **determine the exact permissions over resources that actors have in a blockchain network.** Hyperledger Fabric uses certain properties in an actor's identity to determine permissions, and it gives them a special name -- a **principal**. Principals are just like userIDs or groupIDs, but a little more flexible because they can include a wide range of an actor's identity properties. When we talk about principals, we're thinking about the actors in the system -- specifically the actor's identity properties which determine their permissions. These properties are typically the actor's organization, organizational unit, role or even the actor's specific identity.

区块链网络中包括不同的角色：peers，orderers，client app和administrators等。每一个角色都拥有一个封装在X.509数字证书中的身份标识。这些身份非常的重要，因为 **身份决定了不同角色在区块链网络中拥有资源的权限。** Hyperledger Fabric使用角色身份中的某些属性来决定权限，并给了这些属性一个特殊的名字 -- **principal**。Principals 就像用户ID（userIDs）或 群组ID（groupIDs）一样，但是相对更灵活一些，因为它们（principals）包含了很多有关用户身份的属性。在我们讨论pricipals时，我们是在考虑系统中的各个角色 -- 特别是角色的身份属性决定了他们的权限。这些属性通常是标识了角色所在的组织，组织单位，角色，甚至是角色的特殊身份。

Most importantly, **an identity** must be **verifiable** (a real identity, in other words), and for this reason it must come from an authority **trusted** by the system. A membership service provider (MSP) is the means to achieve this in Hyperledger Fabric. More specifically, an MSP is a component that represents the membership rules of an organization, and as such, it that defines the rules that govern a valid identity of a member of this organization. The default MSP implementation in Fabric uses X.509 certificates as identities, adopting a traditional Public Key Infrastructure (PKI) hierarchical model.

更重要的是，**身份**必须是 **可验证的**（换句话来说，一个真实的身份），因此身份必须来自系统 **信任** 的权威。membership service provider (MSP)就是为了在Hyperledger Fabric中实现这个目的。进一步来说，MSP是代表组织中成员规则的组件，因此，MSP定义了一种规则用于管理成员在组织中的有效身份。Fabric默认的MSP实现是利用X.509证书作为身份，采用了传统分层架构的公钥基础设施（PKI）。

A Simple Scenario to Explain The Use of an Identity

一个简单的场景来解释用户身份

Imagine that you visit a supermarket to buy some groceries. At the checkout you see a sign that says that only Visa, Mastercard and AMEX cards are accepted. If you try to pay with a different card -- let's call it an "ImagineCard" -- it doesn't matter whether the card is authentic and you have sufficient funds in your account. It will be not be accepted.

想象你在超市中买一些杂货。在收银处你看到一个招牌上面写着仅可使用Visa，Mastercard 和AMEX。如果你想用不同的卡来付款 -- 让我称之为“ImagineCard” -- 尽管这张卡是有效的并且也拥有足够的余额，但是你仍然不能用它来进行支付。

Having a valid credit card is not enough -- it must also be accepted by the store! PKIs and MSPs work together in the same way -- PKI provides a list of identities, and an MSP says which of these are members of a given organization that participates in the network.

一张有效的信用卡是不够的 -- 它还要被卖家接受。PKIs和MSPs采用相同的方式工作 -- PKI提供身份列表，MSP确定哪些身份是属于参与到网络中特定组织中的成员

PKI certificate authorities and MSPs provide a similar combination of functionalities. A PKI is like a card provider -- it dispenses many different types of verifiable identities. An MSP, on the other hand, is like the list of card providers accepted by the store -- determining which identities are the trusted members (actors) of the store payment network. **MSPs turn verifiable identities into the members of a blockchain network**.

PKI证书颁发机构和MSP提供了类似的功能组合。PKI就像信用卡提供商 -- 商店里被接受的信用卡提供商列表一样 -- 确定了哪些身份是属于在商场付费网络中的可信成员。**MSPs将可验证的身份转变为区块链网络中的成员**。

Let's drill into these concepts in a little more detail.

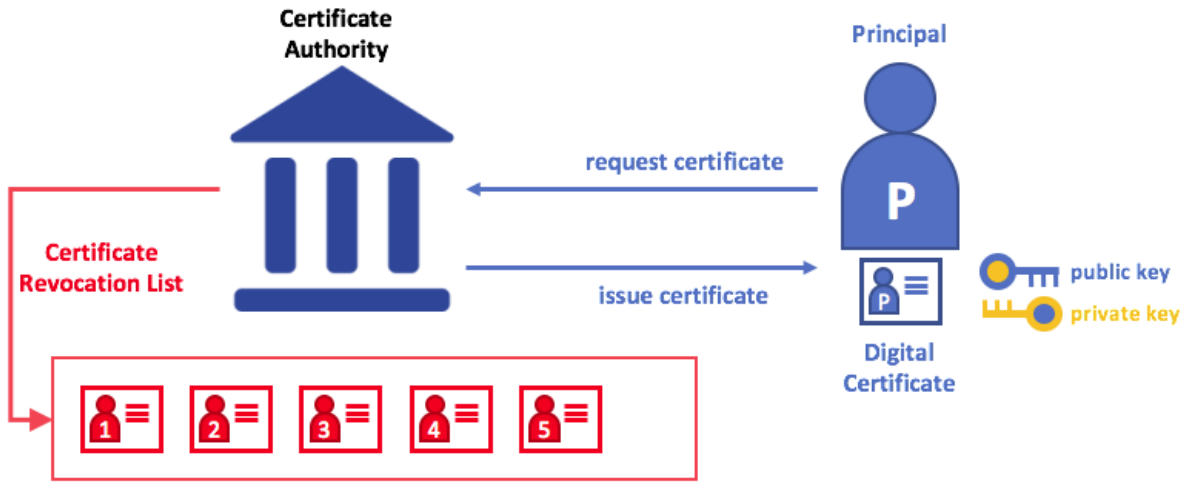
让我们更深入的了解这些概念

What are PKIs?

什么是PKIs?

A public key infrastructure (PKI) is a collection of internet technologies that provides secure communications in a network. It's PKI that puts the **S** in **HTTPS** -- and if you're reading this documentation on a web browser, you're probably using a PKI to make sure it comes from a verified source.

公钥基础设施（PKI）是网络技术的集合，它提供了网络中的安全通信。正是PKI在 **HTTPS**中添加了 **S** -- 如果你正在通过浏览器来读本文档，你或许正在使用PKI来确定本文档来自可信的来源。



*The elements of Public Key Infrastructure (PKI). A PKI is comprised of Certificate Authorities who issue digital certificates to parties (e.g., users of a service, service provider), who then use them to authenticate themselves in the messages they exchange with their environment. A CA's Certificate Revocation List (CRL) constitutes a reference for the certificates that are no longer valid. Revocation of a certificate can happen fora number of reasons. For example, a certificate may be revoked because the cryptographic private material associated to the certificate has been exposed.

*公钥基础设施（PKI）中的要素。PKI包括：证书颁发机构，它颁发证书给用户（例如，服务提供者，服务的用户等），然后用户利用证书在自身所处场景下的信息交互中证明自己的身份。CA的证书撤销列表（CRL）是由失效证书构成。证书的撤销可能是由多种原因造成的。比如，与证书相关的私密信息（如，私钥）泄露造成证书撤销。

Although a blockchain network is more than a communications network, it relies on the PKI standard to ensure secure communication between various network participants, and to ensure that messages posted on the blockchain are properly authenticated. It's therefore really important to understand the basics of PKI and then why MSPs are so important.

尽管区块链不仅仅是一个通信网络，但是它仍需要依赖PKI标准来确保在网络参与者间的通信安全，并确保在区块链中发布的消息是被认证过的。因此，理解PKI的基础知识和采用MSPs的原因是非常重要的。

There are four key elements to PKI:

PKI的四个关键要素：

- **Digital Certificates**
- **Public and Private Keys**
- **Certificate Authorities**
- **Certificate Revocation Lists**
- **数字证书**
- **公私密钥对**
- **证书颁发机构**

- **证书撤销列表**

Let's quickly describe these PKI basics, and if you want to know more details, Wikipedia is a good place to start.

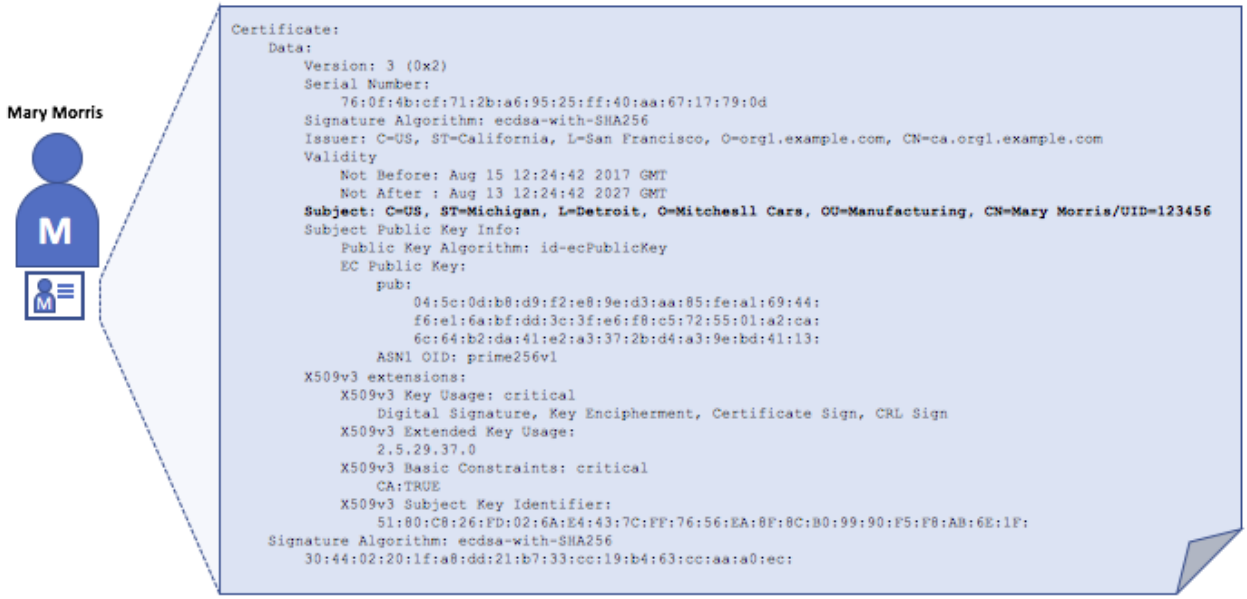
让我们快速的了解一下PKI的基础知识，如果你想要了解更多细节，Wikipedia可以为你提供帮助。

Digital Certificates

数字证书

A digital certificate is a document which holds a set of attributes relating to a party. The most common type of certificate is the one compliant with the X.509 standard, which allows the encoding of a party's identifying details in its structure. For example, John Doe of Accounting division in FOO Corporation in Detroit, Michigan might have a digital certificate with a `SUBJECT` attribute of `C=US, ST=Michigan, L=Detroit, O=FOO Corporation, OU=Accounting, CN=John Doe /UID=123456`. John's certificate is similar to his government identity card -- it provides information about John which he can use to prove key facts about him. There are many other attributes in an X.509 certificate, but let's concentrate on just these for now.

数字证书，是包含了一方多种属性的文档。最常见的证书类型是符合X.509 标准的, 它是由编码后的用户信息组成的特殊结构。例如， 来自密歇根州底特律市FOO公司财务部门的John Doe，他的证书 主体 的属性为 `C=US, ST=Michigan, L=Detroit, O=FOO Corporation, OU=Accounting, CN=John Doe /UID=123456`. John的证书和他的身份证相同 -- 都提供了关于John的关键信息。在X.509证书中还包含了一些其他信息，我们现在来看看都有些什么。



A digital certificate describing a party called John Doe. John is the `SUBJECT` of the certificate, and the highlighted `SUBJECT` text shows key facts about John. The certificate also holds many more pieces of information, as you can see. Most importantly, John's public key is distributed within his certificate, whereas his private signing key is not. This signing key must be kept private.

John Doe所持有的证书。John是证书的主体，主体描述一系列关于John的关键信息。正如你所见，证书中同样包含了许多其他信息。更重要的是，John的公钥在证书中展示，但私钥并不展示。签名私钥必须保密。

What is important is that all of John's attributes can be recorded using a mathematical technique called cryptography (literally, "***secret writing***") so that tampering will invalidate the certificate. Cryptography allows John

to present his certificate to others to prove his identity so long as the other party trusts the certificate issuer, known as a **Certificate Authority** (CA). As long as the CA keeps certain cryptographic information securely (meaning, its own **private signing key**), anyone reading the certificate can be sure that the information about John has not been tampered with -- it will always have those particular attributes for John Doe. Think of Mary's X.509 certificate as a digital identity card that is impossible to change.

John的所有属性都可以通过基于数学的密码学技术（字面意思，"*密写*"）进行记录，因此，任何篡改都会使证书失效。通过密码学，John可以向任何信任 **证书颁发机构** （CA）的人展示他的证书并证明自己的身份。只要CA保管好某些加密信息（意思是，**私钥**），任何人看到这份证书都可以确定其中描述John的信息是未被篡改过的 -- 同时这些信息将一直伴随着John Doe。想想Mary的X.509证书可以作为无法修改的数字身份证。

Authentication & Public keys and Private Keys

身份认证 & 公钥与私钥

Authentication and message integrity are important concepts of secure communication. Authentication requires that parties who exchange messages can be assured of the identity that created a specific message. Integrity requires that the message was not modified during its transmission. For example, you might want to be sure you're communicating with the real John Doe than an impersonator. Or if John has sent you a message, you might want to be sure that it hasn't been tampered with by anyone else during transmission.

身份认证与消息完整性是安全通信中的重要概念。身份认证需要在信息交互中的各方确定消息创建者的身份。完整性需要消息在传输过程中未被修改。举个例子，你想要确定跟你通信的人是真的John Doe而不是冒充者，或者当你接受到John发送的信息，你希望确定消息在传输过程中没有被任何人篡改过。

Traditional authentication mechanisms rely on **digital signature mechanisms**, that as the name suggests, allow a party to digitally **sign** its messages. Digital signatures also provide guarantees on the integrity of the signed message.

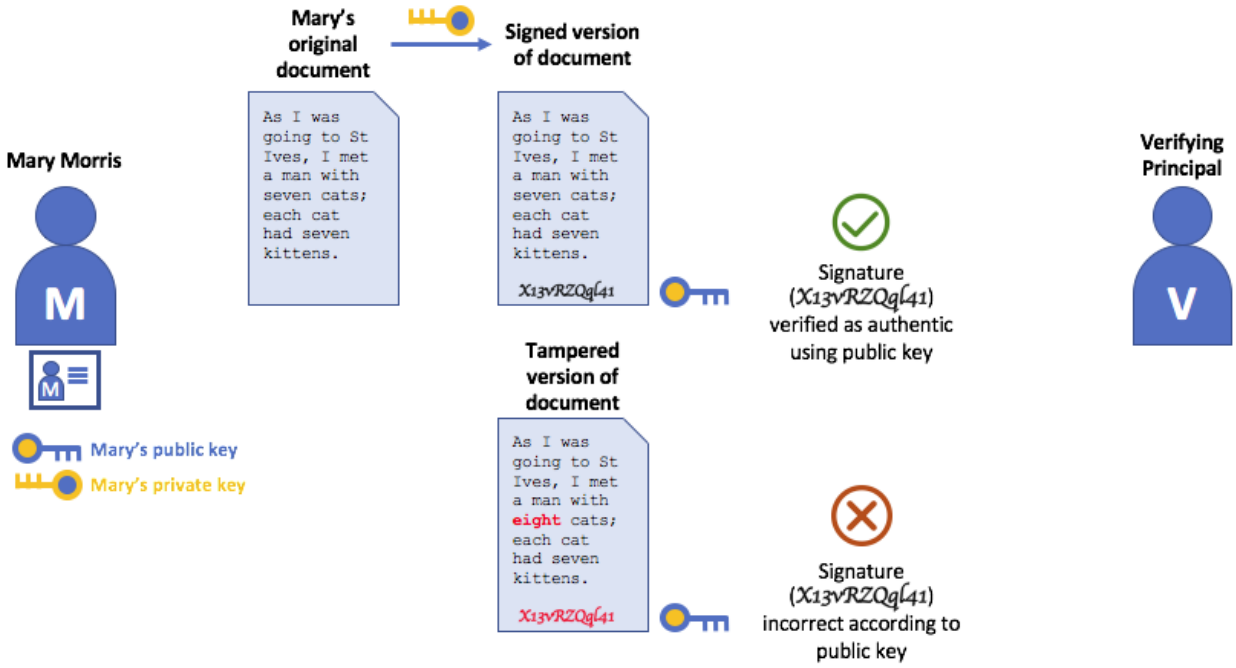
传统的身份认证机制依赖于 **数字签名**，就如名字一样，允许用户利用数字化的方式对消息进行 **签名**。数字签名同时还可以保证被签名消息的完整性。

Technically speaking, digital signature mechanisms require require for each party to hold two cryptographically connected keys: a public key that is made widely available, and acts as authentication anchor, and a private key that is used to produce **digital signatures** on messages. Recipients of digitally signed messages can verify the origin and integrity of a received message by checking that the attached signature is valid under the public key of the expected sender.

从技术上讲，数字签名机制需要各方持有两个存在密码学关系的密钥：公钥，可以大范围使用，充当身份验证锚；私钥，给消息添加 **数字签名**。消息接收者可以通过发送者的公钥来验证消息的来源和消息的完整性。

The unique relationship between a private key and the respective public key is the cryptographic magic that makes secure communications possible. The unique mathematical relationship between the keys is such that the private key can be used to produce a signature on a message that only the corresponding public key can match, and only on the same message.

公私密钥间存在一种特殊的关系，使安全通信成为可能。密钥间存在独特的关系，私钥用于产生消息签名，仅有对应的公钥可以匹配验证，且仅在同一条消息上。



In the example above, to authenticate his message Joe uses his private key to produce a signature on the message, which he then attaches to the message. The signature can be verified by anyone who sees the signed message, using John's public key.

在上面的例子中，Joe为了证明消息是自己所发，使用自己的私钥对消息进行签名，并把数字签名附在消息上。任何看到这条消息的人，可以使用Joe的公钥来验证这条消息。

Certificate Authorities

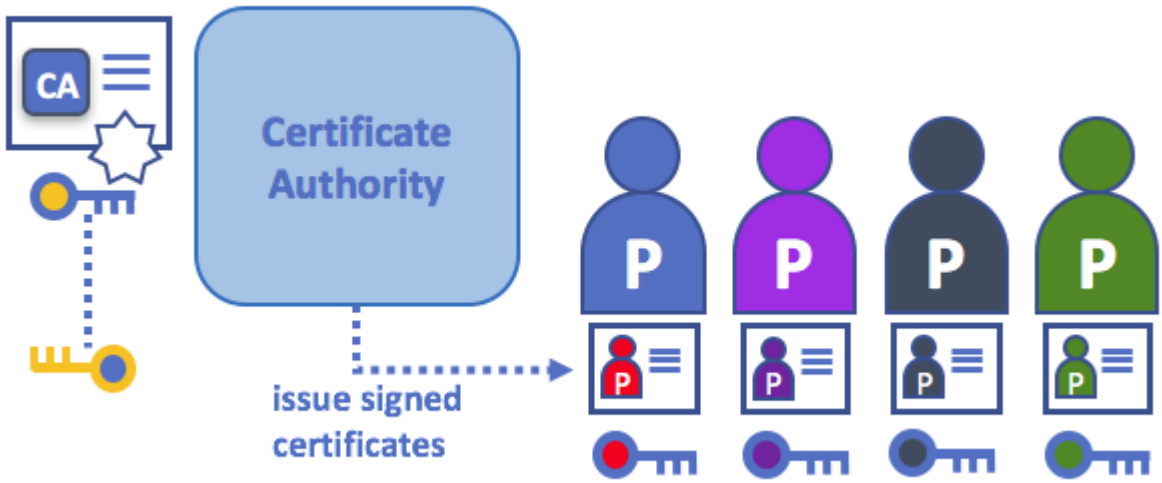
证书颁发机构

As you've seen, an actor or a node is able to participate in the blockchain network, via the means of a **digital identity** issued for it by an authority trusted by the system. In the most common case, digital identities (or simply **identities**) have the form of cryptographically validated digital certificates that comply with X.509 standard, and are issued by a Certificate Authority (CA).

正如你所见，一个角色或一个节点在拥有被系统认可的权威颁发 **电子身份**后，才能加入到区块链网络中。在大多数情况下，电子身份（或称 **身份**）是符合X.509标准的有效数字证书，由证书颁发机构（CA）办法。

CAs are a common part of internet security protocols, and you've probably heard of some of the more popular ones: Symantec (originally Verisign), GeoTrust, DigiCert, GoDaddy, and Comodo, among others.

CA是网络安全协议中的常见部分，你可能听过一些著名的CA：Symantec（最初是Verisign），GeoTrust，DigiCert，GoDaddy，还有Comodo等等。



*A Certificate Authority dispenses certificates to different actors. These certificates are digitally signed by the CA (i.e, using the CA's private key), and bind together the actual actor with the actor's public key, and optionally with a comprehensive list of properties. Clearly, if one trust the CA (and knows its public key), it can (by validating the CA's signature on the actor's certificate) trust that the specific actor is bound to the public key included in the certificate, and owns the included attributes.

*CA为不同的角色颁发证书。这些证书会被CA签名（即，使用CA的私钥进行签名），与角色的公钥进行绑定，并且可选的添加一些与角色相关的信息。显而易见的，如果一个用户相信CA（并知道CA的弓腰），它可以确定（通过验证用户证书上的签名）某个用户与证书上的公钥有关连，并拥有证书上描述的特性。

Crucially certificates can be widely disseminated, as they do not include neither the actors' nor the actual CA's private keys. As such they can be used as anchor of trusts for authenticating messages coming from different actors.

关键的是证书可以广泛传播，因为证书中既不包含用户的私钥也不包含CA的私钥。所以这些证书可以用于验证消息中的签名来确定消息的来源。

In reality, CAs themselves also have a certificate, which they make widely available. This allows the consumers of identities issued by a given CA to verify them by checking that the certificate could only have been generated by the holder of the corresponding private key (the CA).

事实上，CA本身也有用证书，并在大范围内有效。当用户使用被指定CA颁发的证书时，可以通过验证证书上的签名是否与CA的私钥相关联的方式来验证证书的有效性。

In the Blockchain setting, every actor who wishes to interact with the network needs an identity. In this setting, you might say that **one or more CAs** can be used to **define the members of an organization's from a digital perspective**. It's the CA that provides the basis for an organization's actors to have a verifiable digital identity.

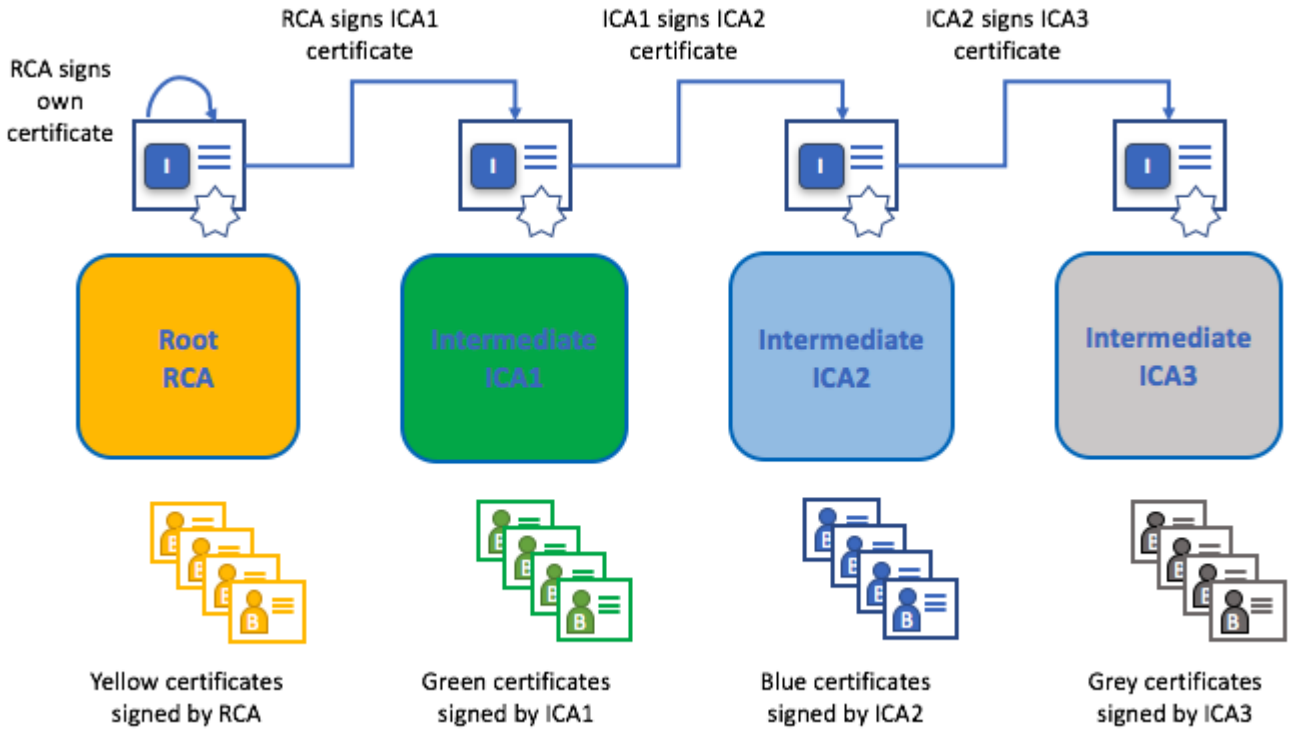
在区块链设置中，每个用户在与区块链网络交互时都需要一个身份。在这种条件下，你可以会说 **一个或多个CA** 可以用于 **在数字角度来定义组织中的成员**。CA为组织中的参与者提供了可验证的数字身份基础。

Root CAs, Intermediate CAs and Chains of Trust

根CA，中间CA和链式信任

CAs come in two flavors: **Root CAs** and **Intermediate CAs**. Because Root CAs (Symantec, Geotrust, etc) have to **securely distribute** hundreds of millions of certificates to internet users, it makes sense to spread this process out across what are called *Intermediate CAs*. These Intermediate CAs have their certificates issued by the root CA or another intermediate authority, allowing the establishment of a "chain of trust" for any certificate that is issued by any CA in the chain. This ability to track back to the Root CA not only allows the function of CAs to scale while still providing security -- allowing organizations that consume certificates to use Intermediate CAs with confidence -- it limits the exposure of the Root CA, which, if compromised, would endanger the entire chain of trust. If an Intermediate CA is compromised, on the other hand, there is a much smaller exposure.

CA分为两种：**根CA**和 **中间CA**。因为根CA（Symantec，Geotrust等）需要分发不计其数的证书给网络中的用户，通过 **中间CA**将证书分发至用户是非常有价值的。这些中间CA的证书是由根CA或其他权威机构办法，用于为任何处于证书链中的CA所颁发的证书构建“链式信任”。这种能力可以追随到根CA，不仅可以提高CA的扩展性，同时还可以保证安全性——（扩展性）允许组织使用中间CA所提供的证书——它限制了根CA的暴露，如果CA失效，整条信任链都将受到威胁。



A chain of trust is established between a Root CA and a set of Intermediate CAs as long as the issuing CA for the certificate of each of these Intermediate CAs is either the Root CA itself or has a chain of trust to the Root CA.

只要每一个中间CA的证书是由根CA或中间CA所颁发，那么就可以在根CA与一系列中间CA间建立证书链。

Intermediate CAs provide a huge amount of flexibility when it comes to the issuance of certificates across multiple organizations, and that's very helpful in a permissioned blockchain system. For example, you'll see that different organizations may use different Root CAs, or the same Root CA with different Intermediate CAs -- it really does depend on the needs of the network.

中间CA提供了极大的灵活性，当需要为多个组织颁发证书时，它同样对需要权限的区块链系统提供很大的帮助。比如，你可以看到不同的组织可能使用不能的根CA，或者使用同一个根CA和不同的中间CA——这取决于所在网络的具体需求。

Fabric CA

Fabric CA

It's because CAs are so important that Fabric provides a built-in CA component to allow you to create CAs in the blockchain networks you form. This component -- known as **fabric-ca** is a private root CA provider capable of managing digital identities of Fabric participants that have the form of X.509 certificates. Because Fabric-CA is a custom CA targetting the Root CA needs of Fabric, it is inherently not capable of providing SSL certificates for general/automatic use in browsers. However, because **some** CA must be used to manage identity (even in a test environment), fabric-ca can be used to provide and manage certificates. It is also possible -- and fully appropriate -- to use a public/commerical root or intermediate CA to provide identification.

因为CA是非常重要的，所以Fabric提供了内置的CA组件来帮助用户创建区块链网络中的CA。这个组件被称为——**fabric-ca**，它是一个私有的根CA，被用来提供Fabric网络中的参与者所需要的数字身份（以X.509证书的形式）。因为Fabric-CA是针对Fabric根CA的需求而设立的自定义CA，所以它本身并不能提供SSL证书在浏览器中常规/自动的作用。然而，因为 **有些**CA必须用来管理身份（甚至是在测试环境），fabric-ca可以用来提供、管理证书。使用公共/商业的根CA或中间CA来提供身份认证也是可以的--并完全支持的。

If you're interested, you can read a lot more about fabric-ca in the CA documentation section.

如果你有兴趣，可以了解更多关于fabric-caCA部分文档.

Certificate Revocation Lists

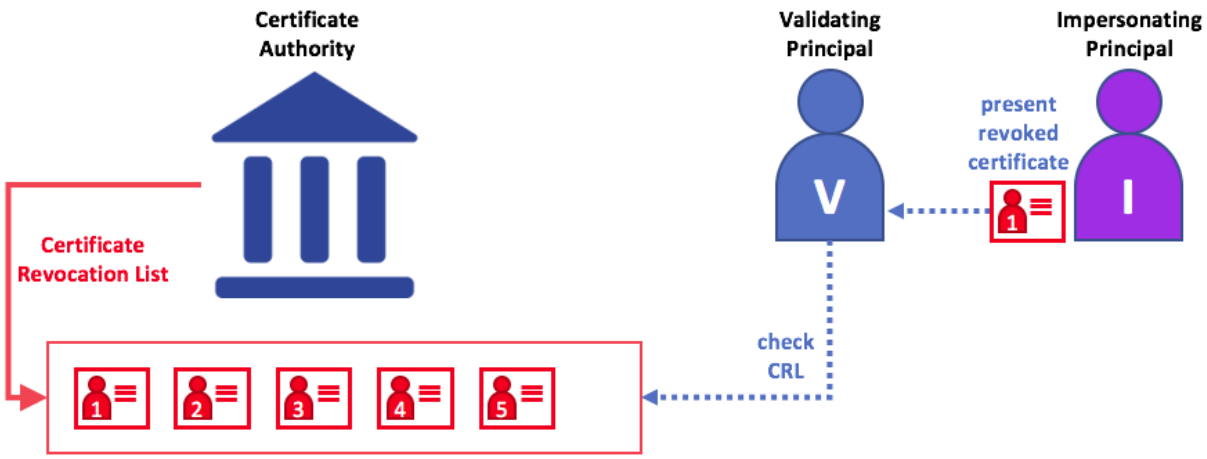
证书撤销列表

A Certificate Revocation List (CRL) is easy to understand -- it's just a list of references to certificates that a CA knows to be revoked for one reason or another. If you recall the store scenario, a CRL would be like a list of stolen credit cards.

证书撤销列表非常容易理解 -- 它只是一个CA知道撤销原因的证书列表。如果你回想超市的场景，证书撤销列表是一个被盗取的信用卡列表。

When a third party wants to verify another party's identity, it first checks the issuing CA's CRL to make sure that the certificate has not been revoked. A verifier doesn't have to check the CRL, but if they don't they run the risk of accepting a compromised identity.

如果一个第三方想要去验证另一个方的身份，它首先会去检查颁发证书的CA的CRL以保证证书未被撤销。检查CRL并不是必须的，但是如果不这么做，就可能会承担身份无效的风险。



Using a CRL to check that a certificate is still valid. If an impersonator tries to pass a compromised digital certificate to a validating party, it can be first checked against the issuing CA's CRL to make sure it's not listed as no longer valid.

使用CRL去检查一个证书是否有效。如果一个冒充者试图使用一个过期的数字证书骗过一个验证方，可以先去检查颁发证书的CA的CRL以确保证书是否有效。

Note that a certificate being revoked is very different from a certificate expiring. Revoked certificates have not expired -- they are, by every other measure, a fully valid certificate. This is similar to the difference between an expired driver's license and a revoked driver's license. For more in depth information into CRLs, click here.

需要注意的是，证书的撤销是不同于证书的过期。撤销的证书并没有过期 --它们是完全有效的证书。就好比过期的驾照和撤销的驾照之间的关系。想要了解更多关于CRL的信息，请点击这里.

Now that you've seen how a PKI can provide verifiable identities through a chain of trust, the next step is to see how these identities can be used to represent the trusted members of a blockchain network. That's where a Membership Service Provider (MSP) comes into play -- **it identifies the parties who are the members of a given organization in the blockchain network**.

现在你已经看到了PKI是怎么通过信任链来提供可验证的身份了，下一步将会介绍这些身份是怎么代表区块链网络中的可信成员的。这就是MSP发挥作用的地方 -- **它来确定谁是区块链网络中一个给定组织中的成员**。

To learn more about membership, check out the conceptual documentation on MSPs.

想要了解更多关于成员的信息，请参招概念文档MSPs.