# 4 密钥和地址

You may have heard that bitcoin is based on cryptography, which is a branch of mathematics used extensively in computer security. Cryptography means "secret writing" in Greek, but the science of cryptography encompasses more than just secret writing, which is referred to as encryption. Cryptography can also be used to prove knowledge of a secret without revealing that secret (digital signature), or prove the authenticity of data (digital fingerprint). These types of cryptographic proofs are the mathematical tools critical to bitcoin and used extensively in bitcoin applications.

你可能听说过，比特币是基于密码学，密码学是在计算机安全中广泛使用的一个数学分支。

密码学在希腊语中的意思是“秘密书写”，但密码学不仅是秘密书写，还指加密。

密码学也可以用于：证明了解一个秘密，而不用揭示这个秘密（数字签名）；或证明数据的不会泄露秘密（数字签名），或证明数据的真实性（数字指纹）。

这些类型的密码学证明对比特币来说是关键的数学工具，并且在比特币应用中被广泛使用。

Ironically, encryption is not an important part of bitcoin, as its communications and transaction data are not encrypted and do not need to be encrypted to protect the funds. In this chapter we will introduce some of the cryptography used in bitcoin to control ownership of funds, in the form of keys, addresses, and wallets.

然而，加密不是比特币的重要部分，比特币的通信和交易数据都没有被加密，也不需要加密来保护资金。

在本章中，我们介绍一些在比特币中用于控制资金所有权的密码学，形式是密钥、地址和钱包。

## 4.1介绍

Ownership of bitcoin is established through digital keys, bitcoin addresses, and digital signatures.

比特币的所有权是通过下列建立的：数字密钥、比特币地址、数字签名。

The digital keys are not actually stored in the network, but are instead created and stored by users in a file, or simple database, called a wallet. The digital keys in a user’s wallet are completely independent of the bitcoin protocol and can be generated and managed by the user’s wallet software without reference to the blockchain or access to the internet. Keys enable many of the interesting properties of bitcoin, including decentralized trust and control, ownership attestation, and the cryptographic-proof security model.

数字密钥并不存储在网络中，而是由用户生成和存储，存储在文件或简单数据库（钱包）中。

用户钱包中的数字密钥完全独立于比特币协议，可由用户钱包软件生成和管理，而无需参考区块链或访问互联网。

密钥实现了比特币的许多有趣特性，包括去中心化信任和控制、所有权证明、密码证明安全模型。

Most bitcoin transactions require a valid digital signature to be included in the blockchain, which can only be generated with a secret key; therefore, anyone with a copy of that key has control of the bitcoin. The digital signature used to spend funds is also referred to as a witness, a term used in cryptography. The witness data in a bitcoin transaction testifies to the true ownership of the funds being spent.

大多数比特币交易都需要一个有效的数字签名才会被存储在区块链中，数字签名只能用私钥生成；

因此，任何拥有私钥的人都可以控制那个比特币。

用于花费资金的数字签名也被称为见证（witness），这是密码学中使用的一个术语。

比特币交易中的见证数据证明了对花费的资金拥有所有权。

Keys come in pairs consisting of a private (secret) key and a public key. Think of the public key as similar to a bank account number and the private key as similar to the secret PIN, or signature on a check, that provides control over the account. These digital keys are very rarely seen by the users of bitcoin. For the most part, they are stored inside the wallet file and managed by the bitcoin wallet software.

密钥是成对出现的，一个私钥，一个公钥。

公钥就像银行的账户，私钥就像支票上的签名，它提供对账户的控制。。

比特币的用户很少直接看到这些数字密钥。

多数情况下，比特币钱包软件把它们存储在钱包文件中，并管理它们。

In the payment portion of a bitcoin transaction, the recipient’s public key is represented by its digital fingerprint, called a bitcoin address, which is used in the same way as the beneficiary name on a check (i.e., "Pay to the order of"). In most cases, a bitcoin address is generated from and corresponds to a public key. However, not all bitcoin addresses represent public keys; they can also represent other beneficiaries such as scripts, as we will see later in this chapter. This way, bitcoin addresses abstract the recipient of funds, making transaction destinations flexible, similar to paper checks: a single payment instrument that can be used to pay into people’s accounts, pay into company accounts, pay for bills, or pay to cash. The bitcoin address is the only representation of the keys that users will routinely see, because this is the part they need to share with the world.

在比特币交易的支付环节，收款人的公钥是用它的数字指纹来表示的，称为“比特币地址”，就像支票上的收款人姓名。

多数情况下，比特币地址由一个公钥生成，并与其对应。

然而，并非所有比特币地址都代表公钥，它们也可以代表其他支付对象，例如脚本，将在本章后面看到。

这样一来，比特币地址就是资金接收者的抽象，使得交易的目的地更加灵活，就像支票一样：可以支付给个人账户、公司账户、支付单支、支付现金。

比特币地址是用户经常看到的密钥的唯一表示，因为他们需要把这部分告诉其他人。

First, we will introduce cryptography and explain the mathematics used in bitcoin. Next, we will look at how keys are generated, stored, and managed. We will review the various encoding formats used to represent private and public keys, addresses, and script addresses. Finally, we will look at advanced use of keys and addresses: vanity, multisignature, and script addresses and paper wallets.

首先，我们将介绍密码学，并解释在比特币中使用的数学知识。

然后，了解密钥如何被产生、存储和管理。

我们将了解各种编码格式，用于表示私钥和公钥、地址、脚本地址。

最后，介绍密钥和地址的高级用途：比特币靓号、多签名、脚本地址、纸钱包。

### 4.1.1 公钥密码学和加密货币

Public key cryptography was invented in the 1970s and is a mathematical foundation for computer and information security.

公钥密码学发明于1970年代，它是计算机和信息安全的一个数学基础。

Since the invention of public key cryptography, several suitable mathematical functions, such as prime number exponentiation and elliptic curve multiplication, have been discovered. These mathematical functions are practically irreversible, meaning that they are easy to calculate in one direction and infeasible to calculate in the opposite direction. Based on these mathematical functions, cryptography enables the creation of digital secrets and unforgeable digital signatures. Bitcoin uses elliptic curve multiplication as the basis for its cryptography.

自从发明了公钥密码学之后，已经发现了一些合适的数学函数，例如：素数求幂和椭圆曲线乘法。

这些数学函数都是不可逆的，就是说，很容易在一个方向上计算，但无法在反方向上计算。

基于这些数学函数，密码学能够创建数字秘密和不可伪造的数字签名。

比特币使用椭圆曲线乘法作为其密码学的基础。

In bitcoin, we use public key cryptography to create a key pair that controls access to bitcoin. The key pair consists of a private key and—​derived from it—​a unique public key. The public key is used to receive funds, and the private key is used to sign transactions to spend the funds.

在比特币中，我们用公钥密码学创建一个密钥对，用于控制比特币。

密钥对包括一个私钥和一个公钥（用私钥生成）。

公钥用于接收资金，私钥用于对交易进行签名，以花费资金。

There is a mathematical relationship between the public and the private key that allows the private key to be used to generate signatures on messages. This signature can be validated against the public key without revealing the private key.

公钥和私钥之间有一个数学关系，使得可以使用私钥对消息生成签名。

可以用公钥验证这个签名，而不需要泄露私钥。

When spending bitcoin, the current bitcoin owner presents her public key and a signature (different each time, but created from the same private key) in a transaction to spend those bitcoin. Through the presentation of the public key and signature, everyone in the bitcoin network can verify and accept the transaction as valid, confirming that the person transferring the bitcoin owned them at the time of the transfer.

当花费比特币时，比特币的当前所有者需要在交易中提供他的公钥和签名（签名每次都不同，但都是用同一私钥生成的）。

通过提交公钥和签名，比特币网络中的所有人都可以验证和承认这个交易是有效的，确认转钱的这个人对比特币有所有权。

Tip：In most wallet implementations, the private and public keys are stored together as a key pair for convenience. However, the public key can be calculated from the private key, so storing only the private key is also possible.

提示：在大多数钱包中，为了方便，是把私钥和公钥作为一个密钥对来存储的。

但是，可以用私钥计算得到公钥，所以也可以只存储私钥。

### 4.1.2 私钥和公钥

A bitcoin wallet contains a collection of key pairs, each consisting of a private key and a public key. The private key (k) is a number, usually picked at random. From the private key, we use elliptic curve multiplication, a one-way cryptographic function, to generate a public key (K). From the public key (K), we use a one-way cryptographic hash function to generate a bitcoin address (A).

比特币钱包中包含一些密钥对，每个密钥对有一个私钥和一个公钥。

私钥（k）是一个数字，通常是随机选择的数字。

有了私钥，就可以使用椭圆曲线乘法来生成公钥（K），椭圆曲线乘法是一个单向加密函数。

有了公钥（K），再使用一个单向加密哈希函数生成比特币地址（A）。

In this section, we will start with generating the private key, look at the elliptic curve math that is used to turn that into a public key, and finally, generate a bitcoin address from the public key. The relationship between private key, public key, and bitcoin address is shown in [Private key, public key, and bitcoin address](https://github.com/bitcoinbook/bitcoinbook/blob/develop/ch04.asciidoc#k_to_K_to_A).

本节从生成私钥开始，讲述椭圆曲线运算如何用私钥生成公钥，最后在用公钥生成比特币地址。

下图显示了私钥、公钥、比特币地址之间的关系。

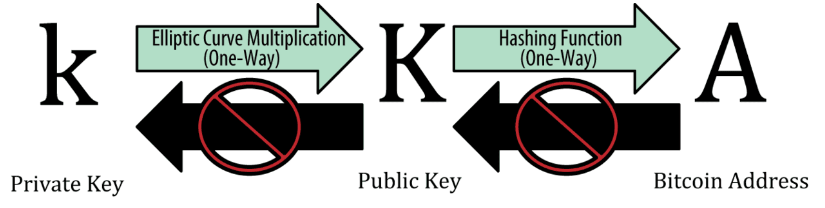


Figure 1. Private key, public key, and bitcoin address

**Why Use Asymmetric Cryptography (Public/Private Keys)?**

**为什么使用非对称密码学（公钥/私钥）？**

如果把比特币交易比作一张支票，比特币地址就是收款人，也就是我们在收款人一栏写的内容。

支票的收款人可能是银行账户持有者的名字，也可能是公司、机构，甚至现金。

因为支票不需要指定一个账户，而是用一个抽象的名字作为收款方，这使它成为一种灵活的支付工具。

Bitcoin transactions use a similar abstraction, the bitcoin address, to make them very flexible. A bitcoin address can represent the owner of a private/public key pair, or it can represent something else, such as a payment script, as we will see in [[p2sh]](https://github.com/bitcoinbook/bitcoinbook/blob/develop/ch04.asciidoc#p2sh). For now, let’s examine the simple case, a bitcoin address that represents, and is derived from, a public key.

比特币地址使用了类似的抽象（比特币地址），使之非常灵活。

比特币地址可以表示一对公钥/私钥的所有者，也可以表示其它东西，例如支付脚本。

现在，让我们来看一个简单的例子，从一个公钥得到一个比特币地址。

The bitcoin address is derived from the public key through the use of one-way cryptographic hashing. A "hashing algorithm" or simply "hash algorithm" is a one-way function that produces a fingerprint or "hash" of an arbitrary-sized input. Cryptographic hash functions are used extensively in bitcoin: in bitcoin addresses, in script addresses, and in the mining Proof-of-Work algorithm.

使用单向加密哈希，可以从公钥得到比特币地址。

一个哈希算法是一个单向函数，它对一个任意大小的输入产生一个指纹或哈希。

加密哈希函数在比特币中被广泛使用：比特币地址、脚本地址、挖矿的工作量证明算法。

The algorithms used to make a bitcoin address from a public key are the Secure Hash Algorithm (SHA) and the RACE Integrity Primitives Evaluation Message Digest (RIPEMD), specifically SHA256 and RIPEMD160.

用于从公钥生成比特币地址的算法是SHA和RIPEMD，具体地说是SHA256和RIPEMD160。

Starting with the public key K, we compute the SHA256 hash and then compute the RIPEMD160 hash of the result, producing a 160-bit (20-byte) number:

以公钥 K作为输入，计算其SHA256哈希，然后对结果再计算RIPEMD160 哈希，得到一个长度为160位（20字节）的数字。

A = RIPEMD160(SHA256(K))

where K is the public key and A is the resulting bitcoin address.

K是公钥，A是生成的比特币地址。

Tip：A bitcoin address is not the same as a public key. Bitcoin addresses are derived from a public key using a one-way function.

**提示：**比特币地址不等于公钥。

比特币地址是用单向哈希函数对公钥计算后生成。

Bitcoin addresses are almost always encoded as "Base58Check" (see [Base58 and Base58Check Encoding](https://github.com/bitcoinbook/bitcoinbook/blob/develop/ch04.asciidoc#base58)),

这种编码格式提供了下列的平衡：简洁表示、可读性、错误检测和预防。

Base58是Base64的一个子集，也使用大小写字母和数字，但舍弃了一些容易读错和在特定字体中容易混淆的字符。具体来说，Base58不含Base64中的0（数字0）、O（大写字母o）、l（小写字母 L）、I（大写字母i），以及+和/这两个字符。简而言之，Base58就是由不包括（0，O，l，I）的大小写字母和数字组成。

Example 2. Bitcoin’s Base58 alphabet

例2：比特币的Base58字母表

123456789ABCDEFGHJKLMNPQRSTUVWXYZabcdefghijkmnopqrstuvwxyz

To add extra security against typos or transcription errors, Base58Check is a Base58 encoding format, frequently used in bitcoin, which has a built-in error-checking code. The checksum is an additional four bytes added to the end of the data that is being encoded. The checksum is derived from the hash of the encoded data and can therefore be used to detect and prevent transcription and typing errors. When presented with Base58Check code, the decoding software will calculate the checksum of the data and compare it to the checksum included in the code. If the two do not match, an error has been introduced and the Base58Check data is invalid. This prevents a mistyped bitcoin address from being accepted by the wallet software as a valid destination, an error that would otherwise result in loss of funds.

为了增加额外的安全性（防止输入和转录错误），Base58Check是一种Base58编码格式，常用于比特币，它有内置的错误检查码。

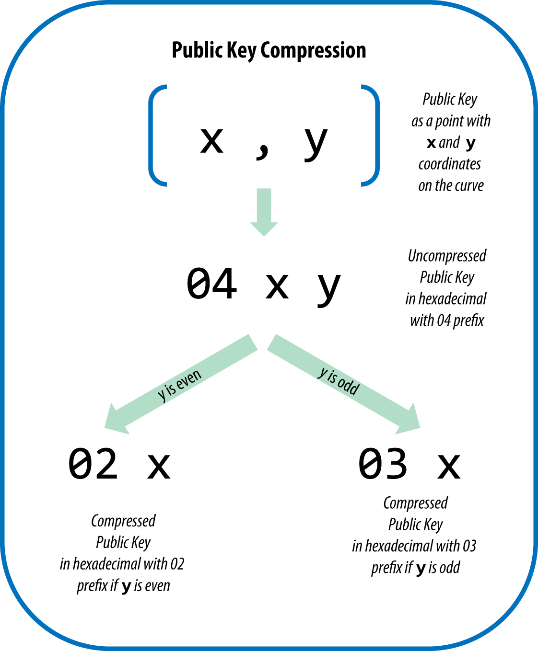


Figure 7. Public key compression

Here’s the same public key generated previously, shown as a compressed public key stored in 264 bits (66 hex digits) with the prefix 03 indicating the y coordinate is odd:

下面是前面产生的相同的公钥，显示为264比特（66个十六进制数字）的压缩公钥，前缀是03，表示y坐标是奇数。

K = 03F028892BAD7ED57D2FB57BF33081D5CFCF6F9ED3D3D7F159C2E2FFF579DC341A

This compressed public key corresponds to the same private key, meaning it is generated from the same private key. However, it looks different from the uncompressed public key. More importantly, if we convert this compressed public key to a bitcoin address using the double-hash function (RIPEMD160(SHA256(K))) it will produce a different bitcoin address. This can be confusing, because it means that a single private key can produce a public key expressed in two different formats (compressed and uncompressed) that produce two different bitcoin addresses. However, the private key is identical for both bitcoin addresses.

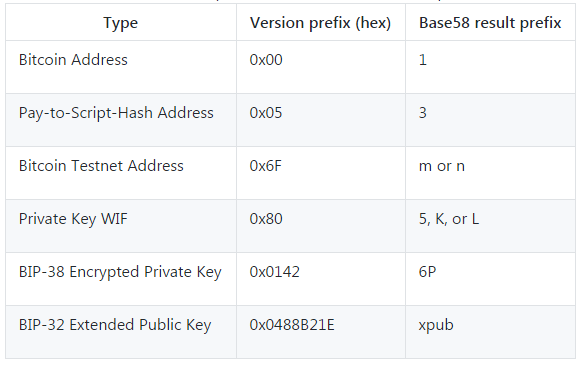
这个压缩公钥对应同一私钥，即，它是由同一私钥所生成。

但是，它与未压缩公钥看起来不同。

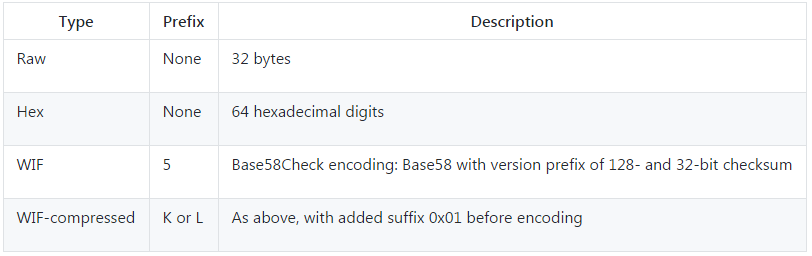
更重要的是，如果用RIPEMD160(SHA256(K))将压缩公钥转化成比特币地址，会产生不同的比特币地址。

这让人迷惑，因为这意味着，一个私钥可以生成两种不同格式的公钥（压缩格式和未压缩格式），而生成两个不同的比特币地址。但是，这两个比特币地址对应同一个私钥。

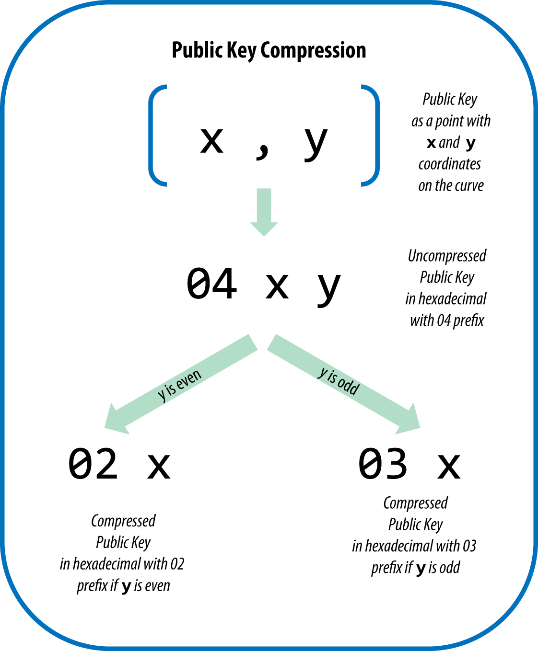
Compressed public keys are gradually becoming the default across bitcoin clients, which is having a significant impact on reducing the size of transactions and therefore the blockchain. However, not all clients support compressed public keys yet. Newer clients that support compressed public keys have to account for transactions from older clients that do not support compressed public keys. This is especially important when a wallet application is importing private keys from another bitcoin wallet application,



表：私钥表示



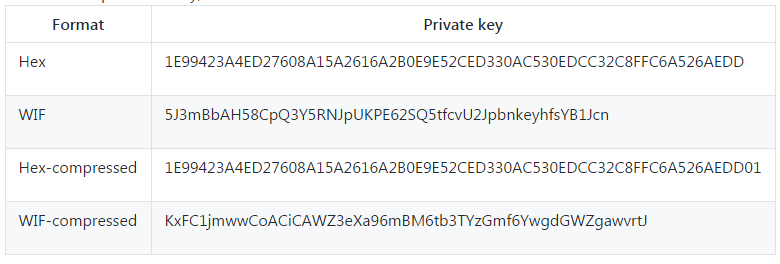
**图：公钥表示**



公钥通常表示为：（04, x, y）

前缀04表示未压缩公钥，前缀02或03表示压缩公钥。

表：相同的私钥，不同的格式



表：BIP-38加密私钥（使用AES）

