敏感信息的所有者使用一个安全的环境运行一个数据库，库中存放个人信息的加密版本，同时提供API来对个人信息进行处理，而另一个数据库中存放API的地址。当访问者请求所有者信息时，所有者将API的地址用访问者的密钥进行加密并写入区块链，访问者根据用自己的私钥解密API地址，调用API，获得关键信息。

API可以使用短期访问令牌，这些令牌在一段时间后过期，然后访问者需要重新获取访问令牌才能继续访问数据。

在数据中添加不可见的水印，以标记数据的来源。为每个用户或每个访问会话创建一个唯一的水印，然后将这个水印嵌入到他们访问的数据中。然后，如果数据被泄露或滥用，你可以通过检测水印来确定哪个用户或会话可能是来源。要用到隐写术。

我要访问某网站，网站请求我的敏感信息，于是，1. 我用私钥对一个消息签名，这个消息存放着区块链上个人信息hash值和当前时间。2. 我发送签名和公钥给网站。3.网站用公钥验证签名，从而确认区块链上信息是我本人的。4.网站在区块链上生成一个交易信息，公开它的签名和公钥，请求我的信息。5.我用它的公钥加密访问个人信息的API并在区块链上生成一个交易。6.网站访问用自己的私钥解密我的API地址，调用API获得信息（有水印的），

%并与hash值比对。

所有者将私人信息加密后传给访问者，但是加密方式比较特别，需要多方参与才能解密信息。首先，假设有3个其他方参与解密，那么所有者随机生成一个这3个其他方的解密顺序，并用访问者的公钥加密这个顺序放置在最终加密信息的头部，然后，依次用这三方的公钥加密一段信息，并为每一段信息生成hash，依次放置在最终加密信息的体中。访问者要获取明文信息则需要多方配合解密，而又可以用hash来验证参与方是否提供了真实信息，而参与方只有一部分信息，无法猜测最终的明文。访问者在搜集到所有的解密信息并验证过hash后得到解密的信息，然而这个明文信息中一部分数据被所有者隐去了，比如用\*来代替。但是，这时访问者通过比对自己数据库中的信息，已足以确认用户的身份，这个明文的最后还有完整明文信息的hash值，访问者猜测后可以用hash做验证。这个明文是以图片的方式给出的，图片中有用户的私人信息但是其中随机地隐去了一小部分，同时这个明文的图片中有访问者的水印，以防止访问者泄露图片后抵赖。

加入门限密码学

秘密共享：所有者生成一个随机的秘密（例如，用于加密信息的对称密钥），然后使用门限密码学（例如，Shamir的秘密共享方案）将这个秘密分割成多个部分。这样，只需要任意两个部分就可以恢复出原始的秘密。

加密：所有者将每个秘密部分分别用参与方的公钥加密，然后将所有的加密部分拼接在一起，形成最终的加密信息。

解密：访问者从任意两个参与方处获取解密部分，然后用他们的私钥解密，恢复出原始的秘密。然后，访问者可以使用这个秘密来解密信息。

在这个方案中，任何两个参与方的私钥都可以用来解密信息。这样，即使有一个参与方不可用或者不愿意合作，访问者仍然可以从其他参与方处获取解密部分，从而成功解密信息。

这种方案提供了很高的容错性和安全性，因为只需要任意两个参与方就可以解密信息，而且每个参与方只持有一部分秘密，无法独立解密信息。然而，这种方案也可能带来一些挑战，例如如何有效地生成和管理秘密部分，以及如何确保各参与方的协作等。在实现这个方案时， 需要考虑这些因素。