Towards Private Cloud Storage in Public Clouds

Yingfei Dong, Zhenhai Duan

Goal: In this project, we will build a *private cloud storage service in public clouds* to protect user privacy and to enable new applications on public clouds.

Motivations: Cloud storage has become a necessary component of our life. Almost everyone carries a personal mobile device (e.g., cell phone) which generally requires cloud servers to store and process its data. Furthermore, cloud storage applications such as Dropbox or Google Drive have become common utility for average users and many businesses to back up data, synchronize across multiple devices, or share files with other users, etc. Inherently, sensitive user data are stored in cloud storage.

Although basic protection (e.g., data confidentiality and integrity) for such storage services has been deployed, other security concerns (e.g., data and access privacy) about such services have not been adequately addressed. For example, it is easy to use encryption to achieve confidentiality against *outside attackers*, but it is much harder to deal with *inside attackers* such as compromised service components, rogue employees, or the provider itself who can directly access to the data. Therefore, privacy-focused cloud storage solutions is not only an attractive feature but also a urgent need.

Related Work: First, client-side encryption methods prevent insiders from seeing the contents of user files by maintaining encryption keys at the client hosts. However, many meta attributes of files (such as ownership, name, size, access time, sharing relationship, etc.) still allow an insider to infer many useful information about the data. Second, advanced encryption methods have been developed to enable useful cloud storage features. For example, Searchable Symmetric Encryption allows a user securely search encrypted files at the provider; Dynamic Searchable Symmetric Encryption based on Blind storage further enables users to hide the associated attributes such as file/folder names and sizes in a data set and hide which encrypted file is accessed.

While these methods achieve strong confidentiality and other advanced features, they do not address the issue of private data access and sharing: a user has to set up an account with a provider; and it has to authenticate to its provider to access its encrypted data or share with other users. As a result, the provider still knows who owns the data and how the data are shared, by mapping an encrypted file to a user account, identifying the IP address(es) of a user account, finding the size, access time, access source of a file, etc. Such private information can be used for side-channel attacks, e.g., inferring relationships among users.

To further prevent an insider from figuring out such private information, a proxy-based method is proposed, i.e., using an anonymous communication tool (such as Tor or I2P) to hide the true source who accesses the data in the cloud. However, such a method only works for a small number of users, and it is not scalable. Current anonymous communication tools cannot support a large number of anonymous connections and have very limited bandwidth. Tor has to limit the bandwidth of each connection to avoid a small number of connections using up all its bandwidth. Furthermore, to set up an account with a large storage space, a user usually has to pay the provider with a credit card or bank account. So, the provider can easily identify the real user associated with an account.

Recent research in decentralized virtual currency (VC) and anonymous credential methods present us a unique opportunity to address this issue: we can anonymously authenticate to a cloud provider and pay the provider via emerging electronic currencies such as Bitcoin (without relying on a trusted third party), such that the provider cannot know who owns an account and then break the data anonymity, e.g., a small company can then buy a large storage space from a public cloud provider for its employee to share, without exposing its identity or any private information to the cloud provider.

Proposed Research: We propose a *private cloud storage service* which allows a client to anonymously access and share its data stored at a public cloud storage provider, with the support of decentralized anonymous user authentication and payment. We assume clients and providers both use a form of electronic virtual currency to execute payments, such as Bitcoin. However, Bitcoin does not provide strong anonymity

because every transaction is known and then multiple transactions can be tied to a single user using various inference methods. We also try to address this issue in our project. To prevent a provider from tracking the client's data, we propose to develop a *decentralized anonymous authentication and payment (DAP)* system to allow a client to anonymously authenticate itself to the provider and execute anonymous payments to the provider. DAP also supports proxy-based anonymous communications in order to hide the physical location and address of a client. Our research will focus on the design of DAP and associated mechanisms.

In the simplest form, all clients and cloud providers can be part of the DAP system to provide the corresponding functionalities; however, this could incur a huge burden on clients. In order to relieve clients from performing the DAP functions, in our design, DAP consists of *super nodes* that support the basic functions of anonymous authentication, payment, and communications, and are independent of clients and providers. A client node or a provider node works with DAP to achieve anonymous authentication and payment transactions. As an incentive, a super node can collect transaction fees from client payments. We note that clients and cloud providers can also work as super nodes to collect fees; however, they are not required to be super nodes.

Super nodes work together as a decentralized public ledger that facilitates anonymous client authentication and payment. As in other VC systems, all nodes will have an address (pseudonym), and super nodes will keep a public ledger of all the transactions; in addition, they will also support mixing so that two addresses of the same client cannot be linked for improved privacy. Decentralized anonymous credentials will be realized in a similar fashion via a public ledger so that a client can be anonymously authenticated to a cloud provider. Furthermore, super nodes will also provide a Tor-like anonymous communications service for clients to hide their physical locations and IP addresses. Individual clients of a cloud provider interact with the DAP system to realize private storage in public cloud providers.

Experiments on NSFCloud. We will use NSFCloud to perform basic feasibility and performance tests in order to better understand how to improve our design and address corresponding issues. For feasibility, we need to figure out the details in developing such a large-scale private cloud storage system in public clouds. Building such a system using NSFCloud will help us explore and understand all the technical details and trade-offs of the system. For performance, we need to investigate the overhead of the scheme on super nodes (including supporting both Bitcoin-like VC and authentication functions and Tor-like anonymous communications services), and the overhead at a public cloud provider for supporting a large number of clients with private storage, in particular, for supporting anonymous authentication and payment (it is less likely for them to support anonymous communication services). In terms of experimental resource support, more specifically,

(1) We will build a public cloud provider (as experiments proceeds, multiple such providers will be built), which will support the private cloud storage service. It will interact with the DAP system to authenticate anonymous clients and obtain anonymous payment from clients. We will build such a cloud storage provider with existing open-source cloud software such as OpenStack, extended to support the decentralized anonymous authentication and payment.

(2) We will also need a large number of nodes to emulate clients of the public cloud storage provider to utilize the private storage service. They will interact with the DAP system to anonymously authenticate themselves to the cloud and to make anonymous payment to the cloud provider. In addition, they will also utilize the anonymous communications service of the DAP system to prevent the cloud storage provider from identifying the true source of a communicating party (client). For this part, we will first emulate a large number of clients at our local private cloud facilities at Florida State University and University of Hawaii. As project proceeds, we will also seek other opportunities to have a more diverse set of client devices in terms of connectivity, bandwidth, and processing power.

(3) We will need a reasonably large number of nodes to carry out the functionalities of the DAP system. We will similarly start with local resources at FSU and UH, and seek other opportunities as the project proceeds.