

[智能电网]

智能电网云存储中支持合作的动态数据完整性保护方法研究

文/北京方胜有成科技股份有限公司 姜维权

针对智能电网中云存储数据完整性验证存在第三方审计不可靠,且无法支持动态更新的问题,提出一种支持合作的数据完整性保护方法。该方法是基于代数签名和分布式多个第三方审计相结合,支持多个第三方审计、动态更新、批验证,并且计算和通信开销较小。最后通过正确性和安全性证明此方案的有效性。

电网技术的发展,及智能电表的部署和传感技术等的大量应用,使电力系统产生了大量结构多样、价值巨大的数据,不仅提高了电网的管理和运行水平,而且还为政府部门、工业界与用户提供了更好的服务。高效的存储和保护海量电力数据的完整性存储是智能电网发展的关键。

利用分布式多TPA技术提出了一种数据完整性验证方法,处理不同用户的验证请求,并使用代数签名进行审计,支持公开验证;文献提出了一种使用代数签名的完整性验证方法,支持动态更新和无限次验证;文献提出了一种分层次索引结构的动态数据完整性验证方法,支持用户多粒度的动态操作,通信开销较低。

但是,分析对比这些方法,这些方案无法在满足数据动态更新、隐私保护和批验证的同时,还具有可靠的第三方。根据上述研究,本文提出一种分布式多TPA的云存储数据完整性验证方法,支持多个第三方审计、动态更新、批验证,并且计算和通信开销较小。

相关技术

1. 代数签名

代数签名是一种具有同态性和代数性的哈希函数,字符串 $S_0, S_1, \dots, S_{n-1}$ 构成的数据块的代

数签名定义为

$$AS_a(S_0, S_1, \dots, S_{n-1}) = \sum_{i=0}^{n-1} S_i \times a^i \quad (1)$$

代数签名的代数特性是指数据块代数签名的和与所对应的数据块和的代数签名结果一致,如公式为

$$AS_a(X) + AS_a(Y) = AS_a(X + Y) \quad (2)$$

2. 多分布式TPA技术

现有的数据完整性验证技术大都支持单个第三方验证,本文提出的验证方案,可以在验证云存储中的数据完整性的同时,也可以保障第三方的安全性,防止恶意的第三方泄露数据。

分布式多TPA系统是由多个TPA组成,通过负载均衡技术,共同承担负载压力,从而极大地提高了数据验证的效率。其云数据存储网络体系架构如图1所示。

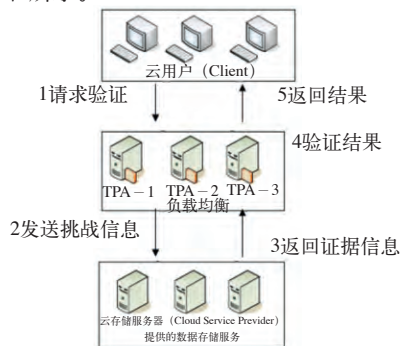


图1 云数据存储网络体系架构

智能电网的云存储数据完整保护方案

1. 智能电网的数据中心模型建立

模型中的两大主体分别是：VS，验证服务器，主要作用是对电力数据中心云存储中的数据进行随机抽样的数据完整性验证工作，保证电力数据仓库中数据的完整性；CSP，云服务平台，主要是为电力数据中心提供云计算和云存储服务，其特点是存储空间大，计算资源丰富。方案中，假设VS是信赖和独立的，其进行的完整性验证是正确且有保证的。则完整性验证模型的结构图如图2所示。

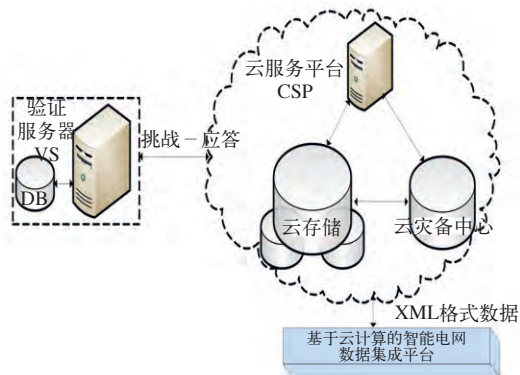


图2 完整性验证模型逻辑结构图

2. 云存储数据完整性验证方案

(1) 设计思想

本文中的协议主要有5个步骤，分别是初始化阶段，数据块签名阶段，挑战阶段，证据生成阶段和证据验证阶段。

1) 初始化阶段。由用户对参数进行初始化，得到主密钥 mk 和大随机数 r_1 和 r_2 ，其中 r_2 是公开的， (mk, r_1) 是私密的，只有云用户有权查看。

2) 数据块签名阶段。云用户对每个数据块进行代数签名，并将所有数据上传到云存储服务器，把与签名相关的数据发送给TPA集群。通过负载均衡和批量审核的方法，TPA集群中的某一个TPA共享并分发签名信息给其他TPA服务器。

3) 挑战阶段。云用户发送挑战信息给TPA集群，TPA集群对云存储服务器发送挑战信息。

4) 证据生成阶段。云存储收到挑战信息后，调用证据生成算法计算证据信息发送给TPA集群。

5) 证据验证阶段。TPA集群验证证据信息，并将结果返回给云用户。

(2) 算法实现

基于代数签名的数据完整性验证协议的具体步骤为：

1) 初始化阶段。主要工作是密钥生成。云用户生成主密钥 mk 和随机数 r_1 和 r_2 ，其中， r_2 是公开

的，云服务器和TPA集群都可以使用。

2) 数据块签名阶段。主要工作是数据块签名。将文件 M 分成 n 个大小相同的数据块 $\{m_1, m_2, m_3, \dots, m_n\}$ ，云用户使用代数签名算法为每一个数据块进行签名，云用户首先计算签名信息为 $\tau_i = \text{Sig}_\alpha(F_i)$ ，把所有的数据 $F = \{F_1, F_2, \dots, F_m\}$ 发送给云存储服务器，并且将相应的代数签名信息 $\tau = \{\tau_1, \tau_2, \dots, \tau_n\}$ 发送给TPA集群。通过负载均衡的方式，TPA集群中的某一个TPA将签名信息分发给每一个TPA集群进行保存，并为其建立一张索引表。

3) 挑战阶段。云用户发送一个挑战信息 k 给TPA集群进行数据完整性验证。其中， x 是云用户生成的一个随机数 $k = f_{mk}(r_1 + x)$ 。

4) 证据生成阶段。主要任务就是云存储服务器生成证据信息。云存储服务器接收到挑战信息 k 后，首先计算随机选择的数据块的位置为 $L_i = \sigma_k(r_2 + i)$ ， $0 \leq i < c$ 得到 $L = \{L_i\}_{0 \leq i < c}$ 。云存储服务器对随机选择的数据块进行求和运算： $\beta = \sum_{i \in L} F_i$ 。再调用代数签名算法对 β 进行代数签名得到 $\text{Sig}(\beta)$ 。并将 $\text{Sig}(\beta)$ 作为证据信息发送给TPA集群。

5) 证据验证阶段。主要工作是TPA集群对云存储服务器发送的证据信息进行验证。TPA集群收到云用户发来的挑战信息 k 后，首先，以同样的方法计算随机选择的数据块的位置： $L_i = \sigma_k(r_2 + i)$ ， $0 \leq i < c$ ，得到 $L = \{L_i\}_{0 \leq i < c}$ 。再通过索引表查到随机选择的数据块代数签名存储的TPA上，对其所有的签名信息进行求和运算，得到 $\sum_{i \in L} \tau_i$ 。然后TPA集群中的主TPA对式(1)进行判断

$$\text{Sig}_\alpha(\beta) = \sum_{i \in L} \tau_i \quad (3)$$

若等式(1)成立，则证明云用户中的数据正确的存储在云端，返回TRUE给云用户，否则，返回FALSE给云用户，证明云存储中的数据遭到了破坏。

(3) 批验证算法实现

很多情况下，多个云用户可能同时需要TPA集群进行完整性验证，验证者需要在较短的时间内完成多个验证任务。将多个验证任务串行验证并不有效。本文提出一种基于双线性对映射的性质对多个验证任务进行批验证，有效的提高了多任务情况下的验证效率。进一步利用递归二分法来定位验证失败的数据，有效的识别错误相对应的用户。

1) 批证据生成。云服务器同时收到 m 个挑战信息，则为每个挑战信息计算随机选择的数据块的位置 L_m ，并计算 $L = \prod_{m=1}^M L_m$ ，云存储服务器对

每个挑战信息计算 β_m ，并计算 $\beta = \prod_{m=1}^M \beta_m$ 。再调用代数签名算法对 β 进行代数签名得到 $Sig(\beta)$ 。并将 $Sig(\beta)$ 作为证据信息发送给TPA集群。

2) 批验证。TPA集群收到云用户发来的挑战信息 k 后，首先，以同样的方法计算随机选择的数据库的位置 L_m ，并得到 $L = \prod_{m=1}^M L_m$ 。再通过索引表查到随机选择的数据块代数签名所存储的TPA，得到每个挑战信息对应的签名 τ_m ，并计算 $\tau = \prod_{m=1}^M \tau_m$ 。然后TPA集群中的主TPA对等式(2)进行判断

$$Sig_{\alpha}(\beta) = \sum_{t \in M} \tau_t \quad (4)$$

若等式(2)成立，则证明云用户的数据正确的存储在云端，返回TRUE给云用户。否则，证明批量审计中某一个响应有错误，导致验证等式失败。

本文利用上述所说的递归二分法定位错误，因为TPA已经计算了每个文件对应的位置 L_m ，可以取 $\{L_m\}_{m \in [1, M]}$ 中签名一半值继续用等式(2)来验证，如果等式成立，则说明错误发生在后半半的数据中，对后半半的数据继续用递归二分法进行验证，直到找出所有错误数据。

案例分析

1. 安全性分析

理论1在本方案中，若云端中的数据块是完整的，则可以通过完整性验证。

证明：本文是基于代数签名的，代数签名既有代数特性又有可压缩性，比较适合进行数据完整性验证。本文方案的正确性证明为

$$\begin{aligned} Sig_{\alpha}(\beta) &= Sig_{\alpha}(\sum_{t \in L} F_t) \\ &= \sum_{t \in L} Sig_{\alpha}(F_t) \\ &= \sum_{t \in L} \tau_t \end{aligned} \quad (5)$$

2. 动态更新分析

本文方案支持数据动态更新，即云用户在任何时刻都能对云端的数据进行数据增加、更新和删除操作，同时，TPA集群可对云端中数据进行完整性验证。

为了支持数据的动态性，TPA集群需要更新存储数据块代数签名位置的索引表，云存储服务器同属需要更新数据库的索引表。在本方案中，云存储服务器端和TPA端都需要维护一个索引表，云存储服务器端和TPA集群端的索引表。

(1) 插入操作

本文提出的方案支持对云端中的数据插入操作，其与在本地进行数据插入操作类似，首先必须要确定数据插入的位置。假设云用户在第 i 个数据块后插入数据块 F^* ，则插入操作为：

1) 用户计算数据块 F^* 的代数签名 $\tau^* = Sig_{\alpha}(F^*)$ 。

2) 云用户将插入信息 $\{F^*, i\}$ 发送给云存储服务器端，继而把签名信息 $\{\tau^*, i\}$ 发送给TPA集群端。

3) 云存储服务器收到插入信息后，将数据块 F^* 插入到第 i 个数据块后，并更新索引表。TPA集群收到签名插入信息后，TPA集群中的主TPA根据负载均衡的原则，选择一TPA对其进行保存，并将其信息插入到索引表。

(2) 删除操作

支持对云端的数据删除操作，与在本地进行数据删除操作类似。但在云端进行数据删除操作是，云用户需要验证云端是否真正执行了数据删除操作。假设云用户要删除云端中第 i 个数据块，删除操作为：

1) 云用户发送一个删除信息 $\{i\}$ 给云端和TPA集群。

2) 云端收到删除信息后，删除第 i 个数据块，并更新索引表。TPA集群收到删除信息后，主TPA通过查找索引表，找到保存第 i 个数据块的签名信息，进行删除，再更新索引表信息。

数据删除操作完成后，云用户可以通过完整性验证算法进行验证，结果表明云存储的删除操作是否符合云用户的要求。

(3) 更新操作

支持对云端数据的更新操作，数据更新操作是最常用的一种云存储业务。对云端中的内容直接修改几乎是不可能的。所以，云端中的数据更新实际上就是使用新的数据块替换旧的数据块。假设用户希望修改第 i 个数 F_i 为 F'_i ，数据更新过程为：

1) 用户计算数据块 F'_i 的代数签名为 $\tau'_i = Sig_{\alpha}(F'_i)$ 。

2) 用户发送更新数据 $\{F'_i, i\}$ 给云端，发送更新签名数据 $\{\tau'_i, i\}$ 给TPA集群。

3) 云端收到更新信息后，把第 i 个索引的数据块更新为 F'_i 。TPA集群收到更新信息后，通过索引表找到第 i 个数据块签名信息，将其修改为 τ'_i ，并更新索引表。

结束语

智能电网中数据量巨大且应用价值高，高效的保护数据的完整性十分必要，本文提出了一种基于代数签名和多个TPA的数据完整性验证方法，并通过正确性和安全性分析证明了此方案的可行性。在本文的基础之上，仍需进一步研究如何减少云用户端的计算量问题，这将是下一步工作的主要内容。EA

(收稿日期：2015.06.18)

(编辑：白文亭)