Applied Mechanics and Materials *Online: 2014-04-04*
ISSN: 1662-7482, Vols. 536-537, pp 489-493
doi:10.4028/www.scientific.net/AMM.536-537.489

A flexible authentication scheme with supporting multiple granularity of data integrity

Long Chen [1,a] , Xiaoyin Yi [2,b]

[1] Chongqing University of Posts and Telecommunications,china
[2] Chongqing University of Posts and Telecommunications,china
[a]email：419872647@qq.com, [b]email：yi13509408767@126.com

Keywords: Cloud Computing;Data Integrity;Granularity；Signature；Dynamic operation;

Abstract. To extend the flexibility of data integrity verification method,adapted to the different verification environment, proposed an improved solution that can support multi-granularity.It organizes files into three kinds of granularity such as data blocks,data sub-blocks and basic-blocks,basic-block realize data gathered to form data sub-block.Sign in the data sub-block,using signature of the sub-block to generate signature of block. Improvement program can achieve the verification of data blocks and sub-blocks. Validation of data block can reduce the data traffic in the validation process,two particle combination can improve the overall efficiency.In the proposed layered merkel hash tree is put forward,the dynamic operation can be supported by the sub-block or the block.Security, communication performance analysis show that the improvement program is effective and has a better practicability

Introduction

Data integrity has become one of the leading security threats in cloud storage,Ateniese et al[1] proposed PDP models for initially data integrity verification program,Erway et al[5] proposed a protocol to support fully dynamic data updating,Kaliski[7]is the first one took consider of POR program in their model.

In the verification C.Wang et al[3][4] use MHT as verification structure,use data block tags as the basic data of authentication,through the data block signature to bind data block and its label.However C.Wang corresponding BLS[6] signature scheme require a fixed block size,only applicable to smaller data blocks verification,causing excessive communication in authentication process,the efficiency is not high.In this paper,combine the idea of different granularity that is mentioned in literature [8],data security authentication scheme with basing on two types of granularity is proposed,a more flexible authentication Mode.in cloud environment the verified granularity is not the bigger the better,the larger granularity can reduce to traffic,but smaller particle size can improve the efficiency of verification.This article is for the two kinds of fine-grained[2]of data blocks and sub-blocks,in cloud environment the verify operations are more frequently, as the traffic is relatively less demanding case,verifier can choose to validate the sub-block units,while in the environment of traffic requirement more stringent,the verifier can through block to validate.It can improve the overall efficiency of verification and increase the flexibility of verification.At the same time put forward one type layered merkle hash tree,two kinds of particle size can be dynamically insert, delete and update,for the threat of the user,it increased that server verify the root node,ensure the real-time of error detection and security,confirmed that the user submits the data files,metadata,accuracy of signature process.It also can support the privacy and public verifiability.

Relevant Concepts

A.Homomorphic Linear Authentication

For each data block m_i,Any challenge values chal=$\{(i, v_i)\}_{i\in I}$,I=$\{s_1,...,s_c\}$are sent by verifier,the server can homomorphism build a validation tag value $\mu = \sum_{i\in I} v_i m_i$

B.Bilinear pairings

Let G_1, G_2 as two additive group,G_T is a multiplicative cyclic group,the order is a prime number P.u and v are the generator of $G_1$and G_2 respectively. The bilinear pairing $\hat{e}: G_1 \times G_2 \to G_T$ has the

following properties:1)Bilinear:for al $\forall u \in G_1, v \in G_2$ land $a,b \in Z_p$, $e(u^a, v^b) = e(u,v)^{ab}$ and $e(u^a, v^b) = e(u,v)^{ab}$ also $e(u_1u_2, v_1) = e(u_1, v_1)e(u_2, v_1)$ $e(u_1, v_1v_2) = e(u_1, v_1)e(u_1, v_2)$；2)Efficient computability:take any $u \in G_1, v \in G_2$ existence an effective algorithm $e(u,v)$;3)Non-degenerate:take any $u \in G_1, v \in G_2$ existence $e(u,v) \neq 1$.

The Proposed Schemes

Definition1:suppose the sub-block signature can be expressed as $\sigma_{i,j}$,block signature can be expressed as σ_i,its relation is： $\sigma_i = \Pi_{1 \le i \le n, 1 \le j \le r} \sigma_{i,j}$

Definition2: Let $H'_{(m_i)}$ represent the block label, $H_{(m_{i,j})}$ represent the sub-block label,its relation is：

$H'(m_i) = \prod_{1 \le i \le n, 1 \le j \le r} H(m_{i,j})$

A.Layered MHT

Layered MHT.It is a hierarchical structure division on the logical for MHT.The purpose is more vivid show that the integrity check process with two types of granularity , and support for dynamic operation of the sub-block and block. layered MHT structure can be shown in Figure 1, the tree is divided into two layers,I layer and J layer.Below the dotted line is J layer,a leaf node of J layer represents the hash value $h(H(m_{i,j}))$of a sub-block label $H(m_{i,j})$.Above the dotted line is I layer,a leaf node of I layer represents the hash value $h(H'(m_i))$of a block label $H'(m_i)$.Root hash value of J layer is the hash value of the leaf node of I layer.

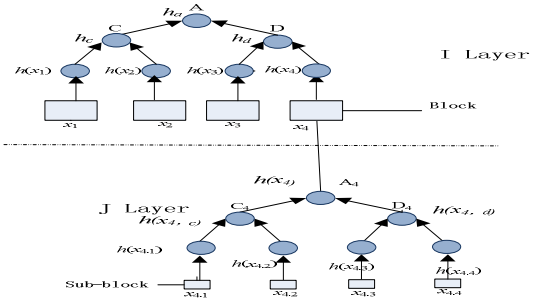

Figure 1Hierarchical MHT

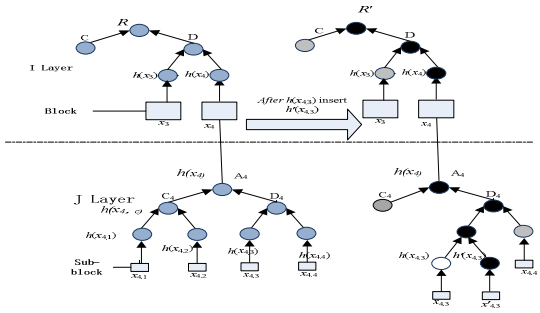

Figure 2 Insertion of sub-block

B.Data integrity verification protocol

Establishment phase:(Generate the key;Generated file tag;Generate authentication tag;)

Generate the key: User run the KeyGen algorithm to generate the public key and private key. First, user can randomly select a signature on(spk,ssk),random number $a \leftarrow Zp$ and randomly selected k elements $\{u_1, u_2, ..., u_k\} \subset z_p$ from G_1,order $\{w_k = u_k^a\}_{1 \le k}$;calculate $v=g^a$,then the private key sk=(a,ssk) and Public key pk =(spk,g,v, $\{w_k\}_{1 \le k}$, $\{u_k\}_{1 \le k}$).sk is retained by the user,pk is released to TPA and CSS.

Generate file tag: Given file $F=\{m_1,m_2,\cdots,m_n\}$, $m_i=\{m_{i,1},m_{i,2},\cdots,m_{i,r}\}$,$m_{i,j}=\{m_{i,j,1},m_{i,j,2},\cdots m_{i,j,k}\}$.the tag of file F is t with $t=name\|v\|g\|u_1\|\cdots\|u_k\|w_1\|\cdots\|w_k\|$ $SSig_{ssk}(name\|v\|g\|u_1\|u_2\|\cdots\|u_k\|w_1\|\cdots\|w_k)$.

Generate authentication tag: For sub-blocks $\{m_{ij}\}_{1 \le i \le n, 1 \le j \le r}$ computing signature $\sigma_{i,j} = (H(m_{i,j}).\prod_{k=1}^{k} u_k^{m_{i,j,k}})^a$, signature of block $\{m_i\}_{1 \le i \le n}$ as in: $\sigma_i = \prod_{1 \le j \le n, 1 \le j \le r} \sigma_{i,j} = ((\prod_{1 \le j \le n, 1 \le j \le r} H(m_{i,j})).\prod_{k=1}^{k} u_k^{\sum m_{i,j,k}})^a = (H'(m_i)).\prod_{k=1}^{k} u_k^{\sum m_{i,j,k}})^a$ $\Phi = \{\sigma_{i,j}\}_{1 \le j \le n, 1 \le j \le r}$ show sub-block signature collection. $\Phi' = \{\sigma_i\}_{1 \le j \le n}$ show block signature collection.Next,user will build file F Layered MHT,each leaf node of I layer is the hash value of data block tag $H'(m_i)$,each leaf node of J layer is the hash value of sub-block tag $H(m_{i,j})$,R represents the root node of tree,private key sign in the root value R:$sig_{sk}(H(R)) \leftarrow (H(R))^y$.Finally user send information $\{F,t,\Phi,\Phi',sig_{sk}(H(R))\}$ to serve.

Server Authentication:

After server receives the information $\{F,t,\Phi,\Phi',sig_{sk}(H(R))\}$,running VerifyUser algorithm , construct layered MHT ,calculate root node R,use $e(Sig_{sk}(H(R)), g) \stackrel{?}{=} e(H(R), v)$ to verify whether user sent to server the root value is consistent, If validation fails,refused to receive and store use data,return FLASE , otherwise , server validate each data block , In equation without privacy protection.

$e(\sigma',g) \stackrel{?}{=} e(\prod_{s_{1\le i\le s_c}} (H'(m_i))^{v_i} \cdot \prod_{1\le k\le I_3} u_k^{\mu'}, v)$ If any data block validate failed,CSP refused to store user's data,otherwise,CSP returned file that has accept stored information to the user side,in order to confirm user submit data authenticity,once user receive the information that have returned to the file storage,if file is miss or damaged without through the integrity verification process,user can demand compensation from the CSP.after user receives file storage information of CSP ,delete user's local data file and verification tag,only retain information that can be used to verify by user.

Validation phase(Generate challenge information;Generate evidence information;Verification)

Generate challenge information: If need to verify sub-block,verifier randomly select a composed subset $I=\{s_1,s_2,\ldots,s_c\}$ of c elements on collection [1,N].For each element i in I,randomly select an element j from the collection I_2,then choose a random element again, $v_{i,j} \leftarrow B \subseteq Z_p$,where j and i are one by one, $i \in I$, $j \in I_2$ .If validation of data block,verifier randomly select subset $I=\{s_1,s_2,\ldots,s_c\}$ on collection [1, N],hypothesis $s_1 \leqslant \cdots \leqslant s_c$,for each $i \in I$,select a random element $v_i \leftarrow B \subseteq Zp$.Then,use"chal" indicate the position of sub-block and block which need to check in inquiry stage . Finally , verifier s end inquiry information $chal\{(i,j,v_{i,j})\}_{i\in I,j\in I_2}$ or $chal\{(i, v_i)\}_{i\in I}$ to the prover.

Generate evidence information: When user permit TPA for public validation , after server receive the inquiry information $chal\{(i,v_i)\}_{i \in I}$ or $chal\{(i,j,v_{i,j})\}_{i\in I,j\in I_2}$,it will randomly selected $\{o_1,o_2,\ldots.o_s\} \subset z_p$ and calculating blinded mask $\{Q_k = (w_k)^{o_j} = (u_k^a)^{o_j}\}_{1\le k\le I_3}$; $Q_k \in G_1$,then generate authentication standard $\mu' = \sum_{i\in I} v_i m_{i,j}$ or $\mu'' = \sum_{i\in I,j\in I_2} v_{i,j} m_{i,j,k}$ that represent a linear combination of block and sub-block based on HLA and treatment for the blind, $\mu_j = \mu' / \mu'' + oh(Q_k) \in Z_p$. At the same time,blocks and sub-blocks were calculated the aggregate signature and corresponding blind treatment by server,as in $\overline{\sigma} = \sigma' / \sigma''.\prod_{1\le k\le I_3} Q_k = \prod_{s_1\le i\le s_c} (\sigma_i)^{v_i} / \prod_{s_1\le i\le s_c, 1\le j\le I_2} (\sigma_{i,j})^{v_{i,j}} \cdot \prod_{1\le k\le I_3} Q_k$ And sever provide auxiliary information AAI $\{\Omega_i / \Omega_{i,j}\}_{i\in I, j\in I2}$ to verifier,on layered MHT , I layer leaves $\{h(H(m_i))\}$ to brother nodes of on the path of the root R and J layer leaf node $\{h(H(m_{i,j}))\}$ to brother nodes of on the path of the root R $\{\overline{\sigma},\ \mu_j, Q_k, (H'(m_i),\Omega_i)_{i\in I} / (H(m_{i,j}),\Omega_{i,j})_{i\in I,j\in I_2}, sig_{sk}(H(R))\}$ As correctness of storing sent to the verifier.

Verification: After verifier receive server response to prove,running VerifyProof algorithm,if verify block,using auxiliary authentication information $\{H'(m_i),\Omega_i\}_{i\in I}$ Calculate root R and through the validation equation $e(Sig_{sk}(H(R)),g) \stackrel{?}{=} e(H(R),v)$ to verify the correctness of the root value R.If the validation fail,return FALSE,otherwise continue to verify equation with privacy protection,as in $e(\prod_{s_1\le i\le s_c} (\sigma_i)^{v_i} \cdot \prod_{1\le k\le I_3} Q_k, g) \stackrel{?}{=} e(\prod_{s_1\le i\le s_c} (H'(m_i))^{v_i} \cdot \prod_{1\le k\le I_3} u_k^{(\mu'+oh(Q_K))}, v)$ Establishment output TRUE;Otherwise output FALSE.If verify sub-block,calculate root use $\{H(m_{i,j}),\Omega_{i,j}\}_{i\in I,j\in I_2}$,using the same equation is given above to verify the correctness of R,finally Verification equation:

$$e(\prod_{s_1\le i\le s_c, 1\le j\le I_2} (\sigma_{i,j})^{v_{i,j}} \cdot \prod_{1\le k\le I_3} Q_k, g) \stackrel{?}{=} e(\prod_{s_1\le i\le s_c, 1\le j\le I_2} (H(m_{i,j}))^{v_{i,j}} \cdot \prod_{1\le k\le I_3} u_k^{(\mu''+oh(Q_K))}, v)$$

C.Dynamic operations

Dynamic operation of block and validation are similar to the literature[8],article mainly introduce the dynamic operation of sub-blocks and corresponding verification of block.

Insertion of sub-block: Assume that user wants to insert $m^*_{i,j}$ after sub-block $m_{i,j,}$ agreement as follows:

1)After calculate the new sub-block signature $\sigma^*_{i,j}$,send request message "update = $(I,i,j,m^*i,j,\sigma^*_{i,j})$ " that represent insert the sub-block to the server.

2)Server receives the request,After the verification execute insert operation:First,Storage $m^*_{i,j}$ and generate J layer leaf node $h(H(m^*_{i,j}))$.then,find the J layer leaf node $h(H(m_{i,j}))$ corresponding to I layer leaf node $h(H'(m_i))$,keep auxiliary authentication Information Ω_i,inserted into the new generated leaf node $h(H(m^*_{i,j}))$ in J layer,add a node as the parent node of node $h(H(m_{i,j}))$ and node $h(H(m^*_{i,j}))$,at last update all nodes related to hash value on the path,calculate new root R',add the signature $\sigma^*_{i,j}$ to Φ,and return the results $P_{update}=(\Omega_i, H'(m_i), Sig_{sk}(H(R),R'))$ to user.

3)After receiving the operation results,using $\{\Omega_i, H'(m_i)\}$ and authentication algorithm validate the effectiveness of mi and mi,j.If verification is invalid output FALSE,Otherwise, calculate the old root node R,through $e(Sig_{sk}(H(R),g) \stackrel{?}{=} e(H(R),v)$ to verify the authenticity of the root R,If equation does not equal output FALSE;otherwise user calculated $H'(m_i^*) = H'(m_i).H(m_{i,j}^*)$ and through $\{\Omega_i, H'(m_i^*)\}$ calculate the new root $R_{new,}$ comparing the values of R_{new} and R';If the two values are different output FALSE;Otherwise, output TRUE,then signature of the new root R' and $H'(m^*_i)$ corresponding block, the signature of the new root $Sig_{sk}(H(R'))$ and σ_i^* sent to server,server saves $Sig_{sk}(H(R'))$ with delete the old root signature files;While add σ_i^* to Φ' and delete the original signature σ_i .Finally,user conduct a verification in block unit.If the result is TRUE, user delete the local information $\{Sig_{sk}(H(R')), P_{update}, m_{i,j}^*, \sigma_i^*\}$. Figure 2 show that user want to insert $x'_{4,3}$ after $x_{4,3,}$ black circles indicate the node that is modified, gray circles indicate the auxiliary authentication information of block.

Performance Analysis

Verification of the sub-blocks and blocks are analysis from the communication cost,according to the literature [8] that gain a set of optimum particle size relationship,1G file as an example,select a different number of sub-blocks and blocks as authentication object,the same file with two different granularity validation that is generate the traffic are analysis,then,For different file sizes,verified with different particle size ratio for further performance analysis.

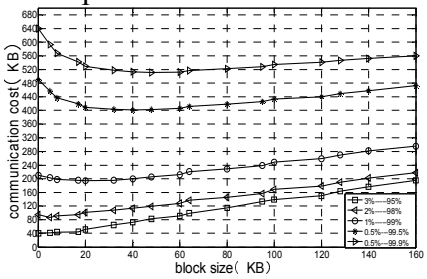

Figure 3 Communication cost comparison chart of different block size

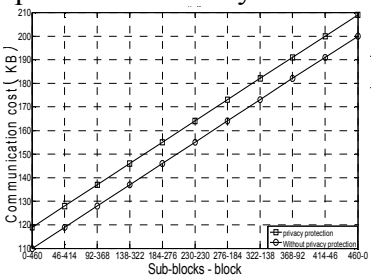

Figure 4 Communication cost comparison chart of 1G file

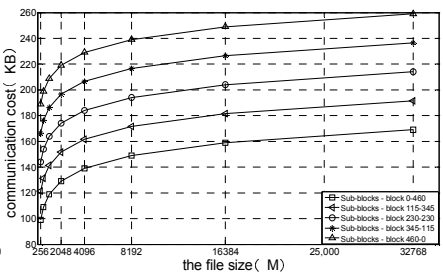

Figure 5 Communication cost comparison chart odifferent files

According to mathematical theory of probability can be conclude the number of test data $c=\log_{1-\rho}(1-t)$, ρ represent the probability of data error ,t is the probability of being detect.So when the error rate are 3%,2%,1%,0.5%,0.5%,and detection probability are 95% , 98% , 99% , 99.5% , 99.9% ,inquiry the number of data are 98, 195, 460, 1055and1380.Figure three shows under in the same block size,with the increase of the number of inquiries,communication cost will increase.

If the verifier data is not complete can be detected with a probability of 99%,then need to select the 460 data as a validation object.Figure four shows the file of 1G,block size is 512KB,sub-block size is 8KB,sub-block and block combined select 460 to verify , when sub-block is 0 , the number of blocks is 460,when sub-block is 46, the number of blocks is 414,and so on， select the number of two kinds of different granularity to verify are produce different communication cost .According to the chart,authentication of block that communication cost is significantly lower than the authentication of sub-block.

In figure five shows when the choice of sub-block and block are 0-460, 115-345, 230-230, 345-115, 460-0 respectively,in the case of file size is the same,when select the authentication of blocks more than sub-blocks,the traffic volume is relatively small.

Security Analysis

Storage correctness: If cloud server pass verifying equation,it must store the integrity of user data.

a)Due to the intractability of DLP in cryptography and Modulus problem with the commutativity, cloud server is impossible to forge an effective evidenceinformation

$$\{\bar{\sigma}, \mu_j, Q_K, (H'(m_i), \Omega_i)_{i\in I} / (H(m_{i,j}), \Omega_{i,j})_{i\in I, j\in I_2}, sig_{sk}(H(R))\}$$

By the following equation: $e(\bar{\sigma},g)\stackrel{?}{=}e(\prod_{s_1\le i\le s_c}(H(m_i))^{v_i}.\prod_{1\le K\le I_3}u_k^{(\mu'+oh(Q_k))}/\prod_{s_1\le i\le s_c,1\le j\le I_2}(H(m_{i,j}))^{v_{i,j}}.\prod_{1\le K\le I_3}u_k^{\mu'+oh(Q_K)},v)$ Also verification will not affected by privacy information $\{Q_k\}_{k\in I_3}$.

b)based on the challenge-response protocol.If response information of the server are true and effective , here $\mu_j=\mu'/\mu''+oh(Q_k)\in Z_p$ then μ'/μ'' also effective. Finally, the correctness of μ'/μ'' is prove the correctness of $\{m_i\}_{i\in I}/\{m_{i,j}\}_{i\in I,j\in I_2}$. This conclusion can be obtained from the anti-collision hash function and discrete logarithm certainty.

Privacy protection: In send to TPA verification certificate, as μ_j value is covered by information elements O_j that is randomly selected by μ'/μ'' information and server.For TPA, known Q_k ,However, due to the difficulty of the discrete logarithm problem,O_j is unknown, so privacy of μ'/μ'' can be guaranteed by μ_j, in the calculation Diffie-Hellman problem, TPA cannot get value of μ'/μ''.So, the problem of privacy protection is guaranteed.

Conclusion

This paper analyzes the integrity verification program to verify the operation of the different needs of data granularity.Put forward a flexible authentication scheme withsupporting multiple granularity of data integrity.Introduce the selectionof data validation,further improve the overall efficiency of verification.Increasing the verification of root in the server authentication,so the threat to users will timely and efficient detection. Among, dynamic operating is support to sub-blocks in order to better satisfy the needs of user. The program also combines publicly available audits and privacy protection features, finally, analyzing and comparing the program is effective and feasible.

Acknowledgements

ChongQin Natural Science Foundation（No.cstc2011jjA40031）

References

[1] G.Ateniese, R. Burns, R. Curtmola, J. Herring, L. Kissner,Z. Peterson, D. Song, Provable data possession at untrusted stores[C]. in Proc. of CCS'07, Alexandria, VA, 2007, pp. 598–609.

[2] Chen Long, Wang Guo-yin. A fine-grained data integrity verification method[J]. Chinese Journal of software, 2009, 20(4): 902-909.

[3] Cong Wang, Qian Wang, Kui Ren, Wenjing Lou. Privacy-Preserving Public Auditing for Data Storage Security in Cloud Computing [C]. INFOCOM, 2010 Proceedings IEEE. IEEE, 2010: 1-9.

[4] Qian Wang, Cong Wang, Jin Li, Kui Ren, Wenjing Lou. Enabling public verifiability and data dynamics for storage security in cloud computing [M]. Computer Security–ESORICS 2009. Springer Berlin Heidelberg, 2009: 355-370.

[5] C. Erway, A. Kupcu, C. Papamanthou, and R. Tamassia. Dynamic provable data possession[J]. in Proc. of CCS'09. Chicago, IL, USA: ACM, 2009.

[6] D. Boneh, B. Lynn, H. Shacham. Short Signatures from the Weil Pairing[C]. Proc. of Cryptology -Asiacrypt'2001. London, UK: Springer-Verlag, 2001: 514-532.

[7] Ari Juels and Burton S. Kaliski Jr., Pors: proofs of retrievability for large files[C]. Proceedings of the 14th ACM conference on Computer and communications security. ACM, 2007: 584-597.

[8] Chen Long, Li Junzhong.A Verifiable Method for Remote Data Integrity Supporting Different Granular Operation[J].Journal of Jilin University.2012.42(1).pp:295-299.

Advances in Mechatronics, Robotics and Automation II
10.4028/www.scientific.net/AMM.536-537

A Flexible Authentication Scheme with Supporting Multiple Granularity of Data Integrity
10.4028/www.scientific.net/AMM.536-537.489