



INFRASTRUCTURE AS A SERVICE BASED DISASTER SENSING

Mr.T.K.P Rajagopal M.E.,M.TECH.,(Ph.d)¹, E.Arun kumar², S.Maria³, R.Narmatha⁴
^{1,2,3,4} Department of CSE, Kathir College of Engineering

Abstract- Most of the banking server and data centres are placed in metropolitan cities, most of the metropolitan cities are in sea shore. For example in India: Chennai, Mumbai. In this paper we are finding out a solution to safe hand the data centres and banking servers. So that IaaS (Infrastructure as a Service) methodology is implemented. Benchmark, and the effectiveness is high for dynamic running in cloud. We propose a novel highly cost-effective and automatically storage capacity is determined whether a generated data set should be stored or not at runtime in the cloud.

Keywords—Mobile computing, cloud computing, mobile cloud, energy-efficient computing, database, cloud server

I. INTRODUCTION

Cloud Computing, which is an Connectivity-based progressing and usage of computer technical methodology. The ever cheaper and most effective processors, join with the software accessing as a service (SaaS) architectural computing, are in the pools of computing service data centres are transforming on a major scale. The improving network and bandwidth and network connections with reliable form will make it possible that users now indulge high qualified services from data and software that reside solely on remote data centres. Moving datas in the cloud computing provides high convenience for user they don't have to care about the complexities of direct management of hardwares. The pioneer in cloud computing domain vendors, Amazon Simple Storage Service (S3) and Amazon Elastic Computer Cloud are the best known examples. While these internet-based online services do provide huge amounts of storage space and personalized computing sources, then the computing place shift, however, is eliminating the responsibility of data maintenance use local machines at the same time. As a result, users are at the mercy of their data. Recent downtime of Amazon's S3 is such an example.

II. SYSTEM ANALYSIS

Proposed system

Security threats faced by CLOUD data storage can come from two different sources. On the one hand, a CSP can be self-interested, untrusted and possibly malicious.

Not only does it desire to move data that has not been or is rarely accessed to a lower tier of storage than agreed for monetary reasons, but it may also attempt to hide a data loss incident due to management errors, Byzantine failures and so on. On the other hand, there may also exist an economically motivated adversary, who has the capability to compromise a number of CLOUD data storage servers in different time intervals and subsequently is able to modify or delete users' data while remaining undetected by CSP for a certain period.

Advantages of proposed system

Because of the availability of cloud service provider the cloud disaster remote monitoring system can be executed successfully. High data transfer is possible due to the availability of higher bandwidth. So that while disaster data loss will not occur.

More data accuracy can be provided during roll back process. It also provide unlimited bandwidth for data transfer with.

III. PROBLEM DEFINITION

At present, the requirements of the system are done manually and hence there are number of limitations.

- The account holder can view that information only with the unique user id and password provided by the bank.
- Time consuming.
- As users no longer possess their data locally, it is of critical importance to assure users that their data are being correctly stored and maintained.

Thus in the system analysis phase of a project the existing system is thoroughly evaluated and all the drawbacks of the existing system are brought to light.

IV. MODULE DESCRIPTION

MODULE LIST

Cloud formation

In Cloud Formation Module it will take us to create country zone. While creating the country we need to specify the unique country code and code number for each and every state. According to this project we put some limits such as country can have only 10 branches. If we try to insert more than 10 branches it will provide a message about the limits. If the same country code was used create a new country it will provide a message as country code already exists error message. When proper data was entered for the branch, bank will be creates in that particular country successfully and a acknowledgement will be received.

Create customer

While creating the customer we need to specify various information's such as name, address, id proof, pan card number, passport number, branch code, If the customer has any existing account or not, if they have an existing account details of those account. And the referral account holder information has to be specified. After providing all those information when we click create customer, in the back round process we are creating the customer account number, Customer ID, Card number, and the card num.

Money integrity

In the Money integrity form we need to pass the account number, name of the account holder, branch code, account mode such as credit or debit of the person we are going to deposit. When those information's are passed successfully and we click the deposit amount button the amount will be deposited to that account number, a message will appear for the conformation of the deposited amount. Particular account holder bank details will be displayed in the form.

Money transfer

In the money transfer form we need to specify the both customers information such as account number, name of the account holder, branch code, account mode credit or debit, and the sender has to specify the amount of transfer. When user click transfer in the back round process the customer account balance will be checked with the transfer amount. If the customer balance is lower than the amount specified for transfer means an error message is specified for the user and the transaction will be incomplete. When the user has the sufficient balance the transaction will take place successfully the transfer amount will be reduced from the sender amount and that amount will be added to the receiver amount. After those process completed successfully a message will be displayed to the user about the transaction.

Personal banking through cloud

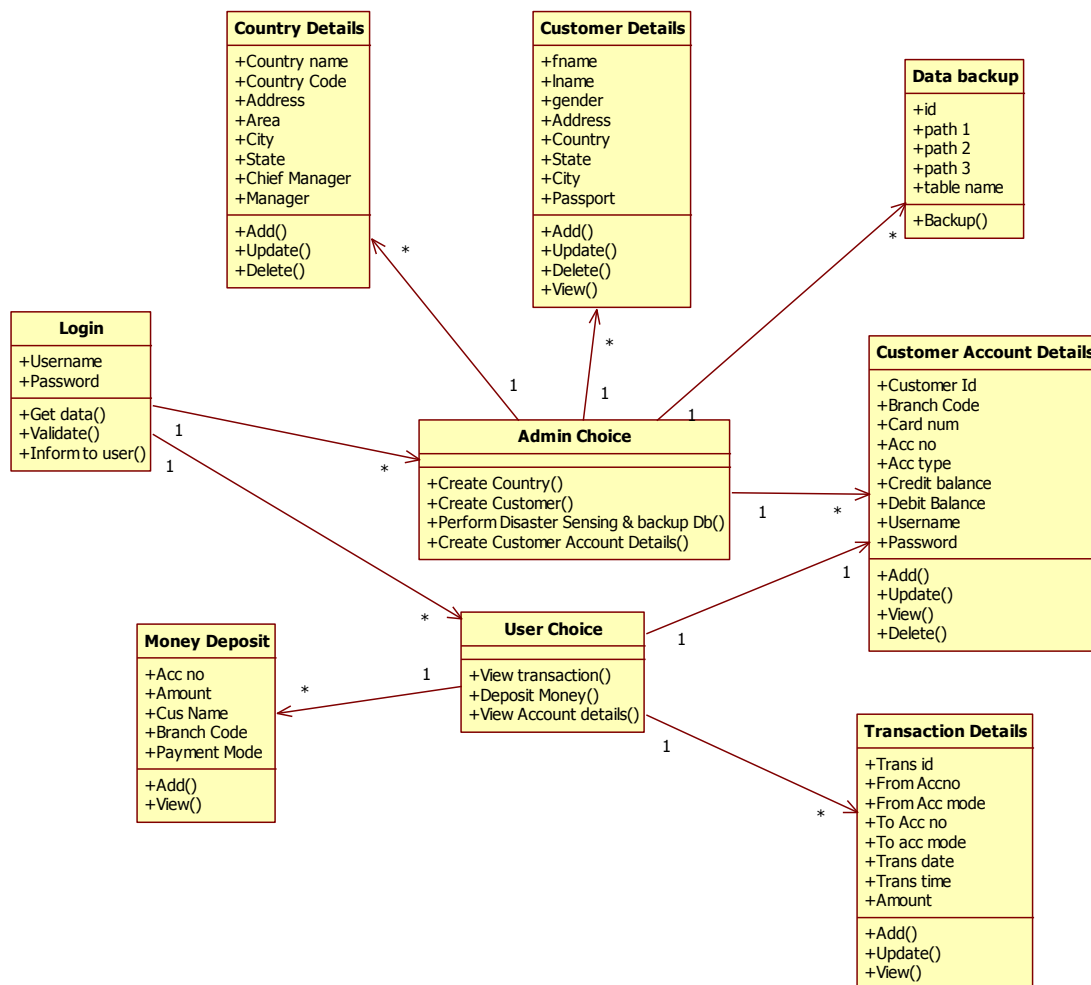
In the personal banking the saving account holders can check there account balance, list of transaction done by the customer, account personal information can be edited efficiently, Each

account holder will be provided by a unique user id and password, By using their user id and password they can view their account details as well as personal information's . Change password rights are provided to the customer to make their own security password.

Corporation banking through cloud

In the Corporate bank the current account holder has the rights to check there account information, list of transaction done in there account, Account balance, Account holders personal information's. The account holder can view that information only with the unique user id and password provided by the bank. The account holder has the rights to change their password according to there wise t maintain their security. If the account holder provides the wrong user ID or Password it will provide an error.

Class diagram:



V. SYSTEM DESIGN

Fundamental concepts of design

System design is a “How to” approach to the creation of a new system. System design goes through 2 phases. They are

1. Logical Design
2. Physical design

Logical design reviews the present physical system, prepares input and output specification, makes edit security and control specification.

Physical design maps out the details of the physical system, plans the system implementation devise a test and implementation plan.

Design process

Input design

Input design is the process of converting user-originated inputs to a computer-based format. Input design is one of the most expensive phases of the operation of computerized system and is often the major problem of a system. In the project, the input design is made in various web forms with various methods.

Login form:

In the administrator login form, the empty username and password is not allowed. The username if not exists in the database, the input is considered to be invalid and is not accepted. Likewise, during the login process of staff and customer, the username is a must and must be available in the user list in the database. Then only login is allowed.

Branch creation form:

During the branch creation option, duplicate entries are not allowed. The email-id, web site, etc are checked and validated and then only entered into the database.

Account creation form:

During account creation, the account number is automatically generated among all the branches. None of the field should be empty.

Transaction entry form:

In the transaction form, the transaction type such as deposit, withdrawal, etc should be mentioned. The amount within the range i.e. 100 –10000 only are allowed. Amount below the minimum amount mentioned during the account creation can be withdrawn.

Transfer entry form:

In the transfer entry form, the account number should not be same. The amount exceeding the minimum amount from the account should be transferred. Any account number of any branch can be used for the transfer.

Selection of two dates for report form:

The dates should be properly entered and from date should be lower than the date. If the transactions within the range are not present, then the message is appeared and the report is not generated.

Output design

Output design generally refers to the results and information that are generated by the system for many end-users; output is the main reason for developing the system and the basis on which they evaluate the usefulness of the application. In the project, the branch details, the account details, the transaction and transfer details, the daybook (all) , between two dates are the web forms in which the output is available.

Branch details:

The branch details are appeared in the order entered.

Account details:

The account details of the selected or logged branch can be viewed in the order created. The editing is possible in the output grid itself.

Transaction details:

The transaction details are displayed in the chronological order. The total deposits, withdrawal, interest and charges applied are also displayed.

Day book details:

The transactions along with transfer details are displayed as day book. It contains all the accounts of the particular logged branch and all or transactions between two dates are viewed.

Transaction details for customer:

The customer views all the transactions of his own or between two dates from his/her login itself. Another accounts output is restricted in customer login.

Code design

The main purpose of performing code design is to simplify the coding to achieve better coding. The loading is prepared in such a way that the internal procedures are more meaningful validation manager is displayed for each column.

The loading of the variable is done in such a way that the one other than the person who have developed the package can understand its purpose.

Database design

The database design is a must for any application developed especially more for the data store projects. Since the chatting method involves storing the message in the table and produced to the sender and receiver, proper handling of the table is a must.

In the project, branch and account table is designed to be unique in accepting the username and the length of the username and password should be greater than zero. The record is added in the current users table only when the signup is made.

The transaction table is common to all customers of all branches. The different users view the data in different format according to the privileges given.

The branch login name and password are separated from the branch details and the account number and password are separated from the account table so that fast retrieval of data is possible and thereby the normalization is made.

Development approach

To transform the “art” of system analysis and design in to an “engineering type” discipline. The feeling that there has to be a more clearly defined logical method for developing a system that meets user requirements has led to new techniques and methodologies that fundamentally attempt to the following:

- Improve productivity of analyst and programmers.
- Improve documentation and subsequent maintenance and enhancements
- Cut down drastically on cost overruns and delays.
- Impure communication among the user, analyst, designer and programmer.

Dataflow diagram

The DFD is also called as bubble chart. It is a simple graphical formalism that can be used to represent a system in terms of the input data to the system, various processing carried out on these data, and the output data is generated by the system.

VI. CONCLUSION

In this paper, we propose an effective and flexible distributed scheme with explicit dynamic data support to ensure the correctness of users' data in the cloud. We rely on erasure correcting code in the file distribution preparation to provide redundancies and guarantee the data dependability. This construction drastically reduces the communication and storage overhead as compared to the traditional replication-based file distribution techniques. By utilizing the homomorphism token with distributed verification of erasure-coded data, our scheme achieves the storage correctness insurance as well as data error localization: whenever data corruption has been detected during the storage correctness verification, our scheme can almost guarantee the simultaneous localization of data errors, i.e., the identification of the misbehaving server(s).

It is concluded that the application works well and satisfy the owner and customers. The application is tested very well and errors are properly debugged. The site is simultaneously accessed from more than one system. Simultaneous login from more than one place is tested.

REFERENCES

1. D. Shires, B. Henz, S. Park, and J. Clarke, "Cloudlet seeding: Spatial deployment for high performance tactical clouds," in Proc. World Congr. Comput.Sci., Comput. Eng., Applied Comput., 2012, pp. 1–7.
2. S. Solti, N. Garg, F. Zheng, J. Lai, Y. Shao, C. Zhang, E. Ziskind, A. Krishnamurthy, and R. Y. Wang, "Segank: A distributed mobile storage system," in Proc. 3rd USENIX Conf. File Storage Technol., 2004, pp. 239–252.
3. B.-G. Chun, S. Ihm, P. Maniatis, M. Naik, and A. Patti, "CloneCloud: Elastic execution between mobile device and cloud," in Proc. 6th Conf. Comput. Syst., 2011, pp. 301–314.
4. M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," IEEE Pervasive Comput., vol. 8, no. 4, pp. 14–23, Oct.-Dec. 2009.
5. S. M. George, W. Zhou, H. Chenji, M. Won, Y. Lee, A. Pazarloglou, R. Stoleru, and P. Barooah, "DistressNet: A wireless Ad Hoc and sensor network architecture for situation management in disaster response," IEEE Commun. Mag., vol. 48, no. 3, pp. 128–136, Mar. 2010.
6. Y. Wen, R. Wolski, and C. Krintz, "Online prediction of battery lifetime for embedded and mobile devices," in Proc. 3rd Int. Conf. Power-Aware Comput. Syst., 2005, pp. 57–72.
7. C. Shi, V. Lakafosis, M. H. Ammar, and E. W. Zegura, "Serendipity: Enabling remote computing among intermittently connected mobile devices," in Proc. 13th ACM Int. Symp. Mobile Ad Hoc Netw. Comput., 2012, pp. 145–154.
8. M. Alicherry and T. Lakshman, "Network aware resource allocation in distributed clouds," in Proc. IEEE Conf. Comput. Commun., 2012, pp. 963–971.
9. A. G. Dimakis, K. Ramchandran, Y. Wu, and C. Su, "A survey on network codes for distributed storage," Proc. IEEE, vol. 99, no. 3, pp. 476–489, Mar. 2011.
10. S. Kosta, A. Aucinas, P. Hui, R. Mortier, and X. Zhang, "ThinkAir: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading," in Proc. IEEE Conf. Comput. Commun., 2012, pp. 945–953.