International Journal of Science and Engineering Applications
Volume 3 Issue 4, 2014, ISSN-2319-7560 (Online)

A Secure Multi-Service Network Connectivity in Cloud Technology

Jayanthi Vagini K
James College of Engineering and Technology
Kanyakumari, T.N., India

Hemalatha M
Hindustan College of Arts and Science
Coimbatore, T.N., India

Amarnath C.T.K
James College of Engineering and Technology
Kanyakumari, T.N., India

Abstract: Cloud computing transforms the way of information technology (IT) by consuming, managing, promising improved cost efficiencies, accelerated innovation, faster time-to-market, and the ability to scale applications on demand. Here the Secure service is considering a key requirement for cloud computing consolidation as a robust and feasible multipurpose solution. In this paper, we propose secure multi-service network connectivity between on-premises and cloud through various services in cloud environment. To provide dynamic scalability for various applications, there has been an increasing trend in business organizations to outsourcing their data to remote cloud at Cloud Service Provider (CSP). In addition, in order to provide cloud-centric building blocks and infrastructure in the areas of secure application connectivity used by Service Bus and Access Control services, designed specifically to the cloud. All external connections come through a load balancer, which is a key to Cloud computing. Here the two main areas of compute and storage functionality as a cloud service.

Keywords: Storage, Service Bus, Access Control, Load Balancer, Cloud Computing;

1. INTRODUCTION

Cloud computing is one of the most emerging technology trends today. Cloud Computing is often described as "resources accessed via a browser over the Internet." However, this definition has become increasingly insufficient to characterize the breadth of applications and use cases for the cloud, and the networks that must support them. The US National Institute of Standards and Technology (NIST, http://csrc.nist.gov) defines it as follows: Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model promotes availability and is composed of five essential characteristics, three delivery models, and four deployment models. The five key characteristics of cloud computing include on-demand self-service, ubiquitous network access, location-independent resource pooling, rapid elasticity, and measured service [1, 2]. Security, reliability, confidentiality, liability, privacy etc are the main concerns on the topic of cloud computing technique. There are two types of securities threats that arise in cloud and these can be defined as: Internal Threats: These are caused internally in the cloud where the CSP can leak the information of the user or may modify it for its own purpose. The users require that their data remain secure over the CSP and they need to have a strong assurance from the cloud servers that CSP store their data correctly without tampering or partially deleting because the internal operation details of service providers may not be known to the cloud users. External Threats: These are caused by some external agents and outside party who can use the stored data of the user for some wrong purpose or leak or modify and delete the data to fulfill his own requirements. Integrity: It facilitates in the recognition of any alteration that has been occurred in the data stored in cloud. It refers to the protection of data from unauthorized deletion, modification or fabrication or we can say in general it refers to the security of the data from malicious parties. Confidentiality: This ensures that the data has been accessed by the authenticated or authorized parties [3]. **Packaged Software -** With packaged software a customer would be responsible for managing the entire stack – ranging from the network connectivity to the applications, as shown in Fig. 1. The industry has defined three categories of Cloud Services which are also called as cloud delivery models are software as a service (SaaS), platform as a service (PaaS), and infrastructure as a service (IaaS). The relationship between IaaS, PaaS, and SaaS in cloud services maps to the components in an IT infrastructure, as shown in Fig. 2. **IaaS** - With Infrastructure as a Service, the lower levels of the stack are managed by a vendor. Some of these components can be provided by traditional hosters – in fact most of them have moved to having a virtualized offering. Very few actually provide an OS. The customer is still responsible for managing the OS through the Applications. For the developer, an obvious benefit with IaaS is that it frees the developer from many concerns when provisioning physical or virtual machines. This was one of the earliest and primary use cases for Amazon Web Services Elastic Cloud Compute (EC2). Developers were able to readily provision virtual machines (AMIs) on EC2, develop and test solutions and, often, run the results 'in production'. The only requirement was a credit card to pay for the services. IaaS – a set of infrastructure level capabilities such as an operating system, network connectivity, etc. that is delivered as pay for use services and can be used to host applications. In IaaS, the cloud provider supplies a set of virtualized infrastructural components such as virtual machines (VMs) and storage on which customers can build and run applications. The application will eventually reside on the VM and the virtual operating system. Issues such as trusting the VM image, hardening hosts, and securing inter-host communications are critical areas in IaaS. **PaaS** - With Platform as a Service, everything from the network connectivity through the runtime is provided and managed by the platform vendor. In fact because we don't provide access to the underlying virtualization or operating system today, we're often referred to as not providing IaaS. PaaS offerings further reduce the developer burden by additionally supporting the platform runtime and related application services. With PaaS, the developer can, almost immediately, begin creating the business logic for an application. Potentially, the increases in productivity are considerable and, because the hardware and operational aspects of the cloud platform are also managed by the cloud platform provider,

applications can quickly be taken from an idea to reality very quickly. PaaS – higher level sets of functionality that are delivered as consumable services for developers who are building applications. PaaS is about abstracting developers from the underlying infrastructure to enable applications to quickly be composed. PaaS enables programming environments to access and utilize additional application building blocks. Such programming environments have a visible impact on the application architecture, such as constraints on which services the application can request from an OS. For example, a PaaS environment might limit access to well-defined parts of the file system, thus requiring a fine-grained authorization service. **SaaS** - With Software as a Service, a vendor provides the application and abstracts you from all of the underlying components. SaaS
– Applications that are delivered using a service delivery model where organizations can simply consume and use the application. Typically an organization would pay for the use of the application or the application could be monetized through ad revenue. Finally, in SaaS, the cloud providers enable and provide application software as on-demand services. Because clients acquire and use software components from different providers, crucial issues include securely composing them and ensuring that information handled by these composed services is well protected [4,5].

Fig. 1 Packaged software

Fig.2 Cloud deliver models as IaaS, PaaS and SaaS

The rest of the paper is organized as follows. Section 2 describes the research background and related work. Section 3 addresses our cloud service architecture and surveys conventional connectivity techniques, and the construction of firewall and load balancing, respectively. In Section 4, we present the security roles and tasks. Finally, we conclude this paper in Section 5.

2. BACKGROUND AND RELATED TOOLS

Cloud Platform is a comprehensive PaaS offering including: Cloud OS, Extending Relational Database to the Cloud, and Cloud Service. The Windows Azure platform provides an Internet-based cloud computing environment for running applications and storing data in Microsoft data centers on the world. You can browse to it at http://windowsazure.com portal for the Windows Azure platform.

A) The Windows Azure platform consists of the Windows Azure cloud-based operating system, which provides the core compute and storage capabilities required by cloud-based applications as well as some constituent services – specifically the Service Bus and Access Control – that provide other key connectivity and security-related features. Cloud OS computes applications in the cloud and provides storage application management through Virtual Network, as shown in Fig. 3. The storage services provide storage for binary and text data, messages, and structured data. The storage services include: The Blob service, for storing binary and text data. The Queue service, for storing messages that may be accessed by a client. The Table service, for structured storage for non-relational data. The drives, for mounting an NTFS volume accessible to code running in your Cloud service, Programmatic access to the Blob, Queue, and Table services is available via the Cloud Managed Library and the Cloud storage services REST API. Blobs, tables, and queues hosted in the cloud, close to your computation. Authenticated access and triple replication to help keep your data safe. Easy access to data with simple REST interfaces, available remotely and from the data canter, Access is via a storage account – you can have multiple storage accounts per live id.

B) The Windows Azure platform also comes with a cloud-based relational database called SQL Azure™, allowing you to move your on-premises relational databases and logic to the cloud. Relational database model delivered as a service which appears to be a Database server to the client, and deployment of multiple databases across multiple datacenters. Cloud-based relational database provides logical server in the portal that is a Gateway server that understands TDS protocol, execute a create DB Command to create a new database, Looks like Database server to TDS Client, Can add and remove DBs easily from application to scale up and down based on business needs, Actual data stored on multiple backend data node. Logical optimizations supported by Indexes, Query plans etc. Physical optimizations not supported because of File Groups, Partitions etc… and finally transparently manage physical storage.

Reporting provided as a service, reports authored using existing tools (BIDS) and uploaded to the cloud, reports can have rich Data Visualizations (Maps, Charts, Tables) and be exported to variety of rendering formats (Excel, Word, PDF), reports can be rendered as part of an app using the Report Viewer control, Directly view the reports in the browse, Web Service interface to render and manage reports.

Data Sync used to sync data between on-premise and the Geo-with

Fig. 3 Cloud-based OS Service

Fig. 4 Cloud-based relational database

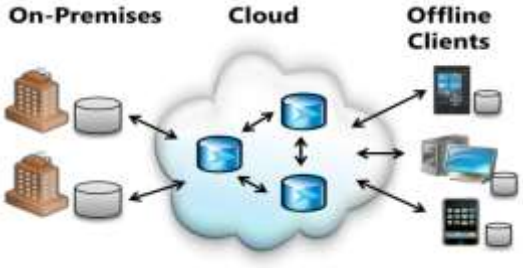

Fig.5 On-premises with cloud and Offline

C) Cloud Service provides services that can be used by any apps – hosted in Cloud OS, on-premises, or hosted in another environment, as shown in Fig. 6.

Fig. 6 Cloud service

These services are really key components you would need for building distributed, connected applications. we talk about connecting to your existing on-premises applications and enabling the composition of hybrid (Cloud + on-premises) applications.

There are currently two Services: the Service Bus & the Access Control Service. The Service Bus and Access Control services that were once collectively known as the .NET Services now run directly within Windows Azure.

Windows Azure now provides secure connectivity natively via Service Bus and Access Control, in much the same way that it also provides compute and storage as a cloud service.

Windows Azure is an operating system as a service – you can think of it as Windows in the cloud. It provides a cloud computing fabric, hosted within Microsoft data centers, for creating, deploying, managing, and distributing applications and services on the Internet, as shown in Fig. 7.

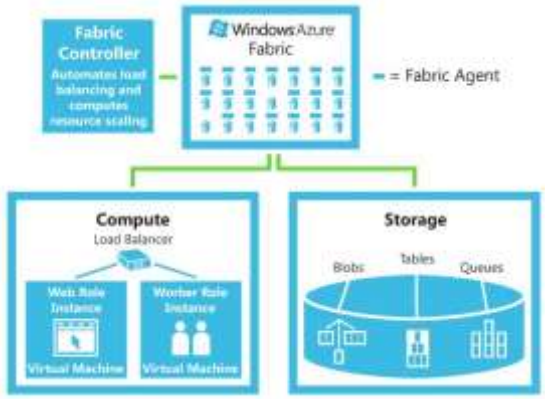

Fig. 7 Windows Azure Fabric

The Windows Azure fabric provides two main areas of functionality: compute (e.g., executing an application) and storage (e.g., storing data on disk), the foundational building blocks for all cloud applications.

In addition to these core services, Windows Azure also comes with Service Bus and Access Control capabilities, which make it easier to extend your .NET applications into the cloud.

Compute: The compute service offered by Windows Azure makes it possible to "execute" your applications in the cloud. The compute service provides you with a way to run your applications on a Windows Server running in a virtual machine hosted in Microsoft data center. When you deploy and application to Windows Azure, you're deploying it to execute within this type of highly-scalable environment. It's important to note that the Windows Azure storage services are designed to be very simple and highly scalable.

Storage: They provide fundamental services for BLOB storage, queue storage, and simple table storage. You interact with these services through a simple REST API based on HTTP requests. You manipulate data in the storage services through traditional POST, PUT, and DELETE requests, and your retrieve information from the storage services using simple GET requests. This approach makes it possible for anyone to integrate with the storage services, regardless of their platform.

Today the Service Bus and Access Control provide core functionality related to secure application connectivity and federated access control as described here:

Service Bus provides network infrastructure for connecting applications over the Internet, using a variety of different messaging patterns, in a way that's capable of traversing firewalls and NAT devices without forfeiting the security afforded by these devices, shown in Fig. 8.

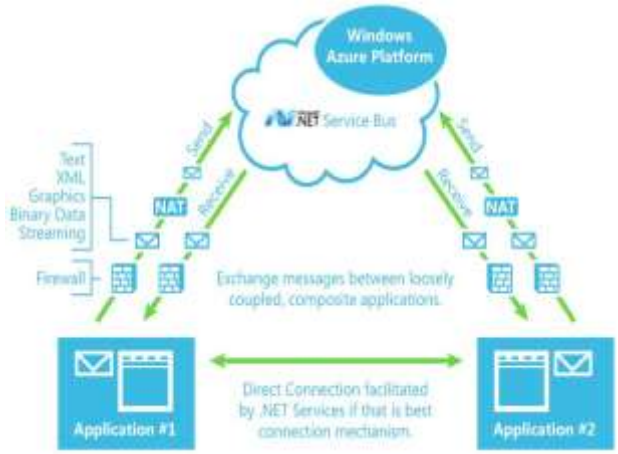

Fig. 8 Service Bus Access Control

Access Control provides claims-based access control in the cloud. It includes a claims transformation engine that federates with identity providers like ADFS v2 (Active Directory Federation Services, as shown in Fig. 9.

Fig. 9 Access Control

Cache is a distributed, in-memory application cache for cloud applications. There are two primary use cases for the Cache. First a session state provider for cloud applications. Secondly

a data cache layer for Cloud Applications that use Relational Databases or Cloud Storage. It's important to understand that Cache is provided as a service. Instead of having to install or manage software on machines or instances, you simply provision, configure, and use the service. This service abstraction also provides more flexibility – so you can dynamically increase or decrease the cache size as needed. Finally, with Cache is uses the same programming model for both the cloud and on-premises. Each of these services is available using open protocols and standards, including REST, SOAP, Atom/Atom Pub, which means developers on any platform can easily integrate with them, referred from [6, 7, 8, and 9].

3. ARCHITECTURE AND TECHNIQUES

This section explores a strong security assurance to the users based on secure multi-service network connectivity in the cloud computing environments. The work mainly used for Identity and Access control services in cloud servers from untrusted and outsourced storages.

We introduce cloud service architecture for key points here is that all external connections come through a load balancer, this includes storage. Inter-role communication (notice there is no load balancer) and TCP ports directly to Worker Roles (or Web Roles). We will still use the storage to communicate async and reliably via queues for a lot of options. However, inter-role communication fills in when you need direct synchronous communication. The load balancers are a key to Cloud computing, as shown in Fig. 10.

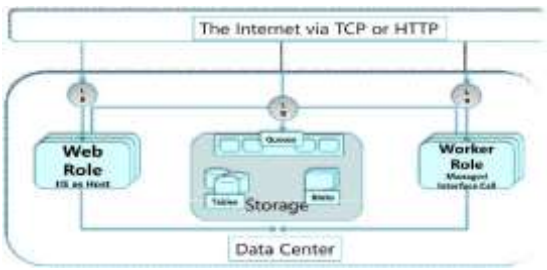

Fig. 10 Cloud Service Architecture.

For example, User uploads large image file, Image inserted into blog storage, Message placed on queue incl BLOB URI and metadata, Worker role is polling queue. Reads message from queue, Worker role processes message, reads from BLOB storage, generates thumbnail, Thumbnail and metadata stored in Table storage, Message deleted from queue. Finally, Cloud Service supports standard IP protocols. Enables hybrid apps access to on-premises servers. Allows remote administration of cloud apps. Simple setup and management. Integrated with cloud Service Model. And also Web, Worker and VM Roles supported. In order to improve secure multi-service network connectivity between on-premises and cloud, we make use of following techniques to construct new applications and challenges.

Service Bus: One of the most common needs in large-scale distributed applications is application connectivity. In fact, application integration is usually one of the most costly and troublesome areas of IT. Today it's common for many organizations to use an enterprise service bus (ESB) solution to address these challenges, as shown in Fig. 11.

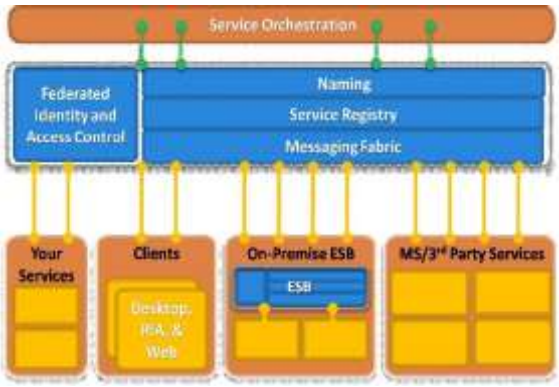

Fig. 11 Internet Service Bus

Tackling bidirectional communication at Internet scope is not trivial due to some of today's networking realities. The Service Bus is designed to provide a general purpose application bus, available on the internet at internet scale. You can really thin of the Service Bus as being similar to an Enterprise Service Bus that many enterprise organizations have today. However, we believe that when providing a Service Bus as a programmable service on the internet, there are a wider range of scenarios for many more types of organizations. Fundamentally, the .NET Service Bus is about connecting applications across network and application boundaries and making key message exchange patterns such as publish and subscribe messaging very simple.

The Access Control service is designed to provide rules-driven, claims-based access control for applications. Essentially, this allows you to define authorization rules for your applications using the claims-based approach that we are adopting within many Microsoft products and technologies and that is becoming adopted in the industry. Here's how it works: the on-premises service connects to the relay service through an outbound port and creates a bidirectional socket for communication tied to a particular rendezvous address. Network policy managed through cloud portal. Granular control of connectivity between cloud roles and external machines. Automatic setup of IPSec through Tunnel firewalls/NAT's through hosted SSL-based relay. Network policies enforced & traffic secured via end-to-end certificate-based IPSec. DNS name resolution based on endpoint machine names.

Relayed Connectivity: Despite these connectivity challenges, some of today's most popular Internet applications are inherently bidirectional. Consider things like instant messaging, online multiplayer games, and peer-to-peer file sharing applications that use protocols such as Bit Torrent, which accounts for a large percentage of all Internet traffic today. These applications have written the low-level networking logic to traverse firewalls and NAT devices, and to create direct peer-to-peer connections when possible. They typically accomplish this through a central relay service that provides the connectivity logic, as shown in Fig. 12.

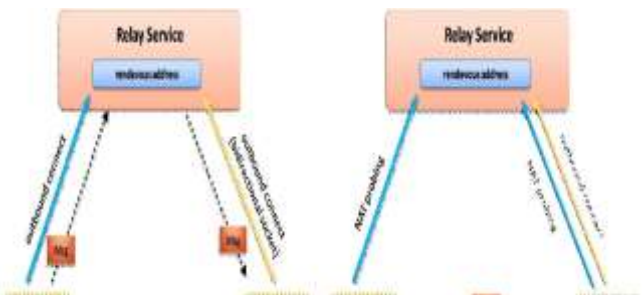

Fig. 12 Establishing a Relay Connection

Fig.13Establishing a Direct Connection

Direct Connectivity: In addition to relayed communications, the Service Bus also provides a capability for establishing direct connectivity between clients and services in order to improve performance and throughput. Clients and services still communicate with the relay through a common rendezvous address but then the relay tries to help them connect directly to one another in order to avoid future relayed transmissions, as shown in above Fig. 13.

The High Scale Application Archetype: High scale applications often follow this sort of an pattern as Inbound connectivity comes through a load balancer(LB), Requests are round robin routed, Load balancer is typically aware of the state of the web servers (i.e. are they up). There are one or more tiers or groups of stateless web or app servers; by stateless we mean that they do not hold state between client requests. Stateless means that simple load balancing works Stateless means that the failure of a web server does not cause major issues for application- it is simply removed from the load balancer. Applications will often perform processing in the background. Improves response time for users, Allows load peaks to be buffered in queues, as shown in Fig. 14.

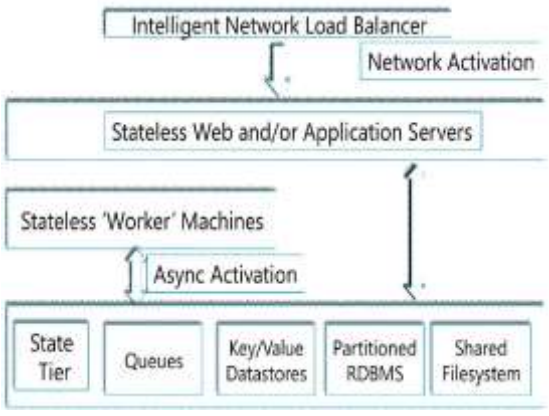

Fig. 14 Inbound Connectivity with Load Balancer

To realize these functions, our cloud service is comprised of four roles: Web Role, Worker Role, Admin Role, and also VM Role. The above techniques involving some procedures based on roles. Moving Applications to the Cloud with VM Role and Web/Worker Role.

4. SECURITY ROLES AND TASKS

Roles are defined in a Service Model, may define one or more Roles per Service. A role definition specifies VM size, Communication Endpoints, Local storage resources, etc. At runtime each Role will execute on one or more instances (up to 20 per subscription). A role instance is a set of code, configuration, and local data, deployed in a dedicated VM.
The Service model defines the shape of a service- the roles it will have, endpoints it will listen on and types of VMs that will be run. At runtime each role will run at a given scale. Specifically each role will be deployed onto and executed on one or more VMs. A VM runs a single role. The various security roles involved in running a cloud account.

Cloud OS currently supports the following two types of roles:A web role is a role that is customized for web application programming as supported by IIS 7 and ASP.NET. A worker ole is an executable (you can create your own web server, host a database, etc.). Inbound on any TCP Port, HTTP/HTTPS.A worker role is a role that is useful for generalized development, and may perform background processing for a web role. A web role is hosted on IIS, HTTP/HTTPS, ASP.NET, Fast CGI + PHP. A service must include at least one role of either type, but may consist of any number of web roles or worker roles.

A worker role is started by a call to a well know managed code interface RoleEntryPoint. A worker role must extend this class and override the Start() method.

Role Lifecycle: All roles may extend RoleEntryPoint, Roles report status via RoleEnvironment. The fabric calls RoleEntryPoint methods as it starts and stops a role. CloudWorkerHost process is started. Worker Role assembly is loaded and surfed for a class that derives from RoleEntryPoint. This class is instantiated. RoleEntryPoint.OnStart() method is called. Called by Fabric on startup, allows you to perform initialization tasks. Reports Busy status to load balancer until you return true. RoleEntryPoint.Run() method is called. Main logic is here – can do anything, typically infinite loop. Should never exit. If the RoleEntryPoint.Run() method exits, the RoleEntryPoint.OnStop() method is called when role is to be shutdown, graceful exit. 30 Seconds to tidy up. CloudWorkerHost process is stopped. The role will recycle and startup again. As a role changes state it will raise the Status Check event. A status of Busy will mean the load balancer will not route requests to the instance, as shown in above Fig. 15.

Fig. 15 Role Lifecycle

Virtual Machine Role: The role is the VM. Use Windows services, scheduled tasks, etc. You configure and maintain the OS. Provided to help you move applications to Cloud OS, it enables you to have full control over the OS image, create

your VHD locally, upload the VHD to storage, deploy a service package that uses the custom OS image, Specify <OsImage href="20101020BaseVM.vhd" /> in the .cscfg., as shown in Fig. 16.

Fig. 16 VM Role with Web/Worker Role

Admin Access & Startup Tasks:
Task Types: Simple [Default] – System waits for the task to exit before any other tasks are launched. Background – System does not wait for the task to exit. Foreground – Similar to background, except role is not restarted until all foreground tasks exit.

5. CONCLUSION

In this paper, we presented a construction of secure multi-service network connectivity through a various services for cloud applications to avoid security risks. We also focused on how cloud delivery models helps to provide dynamic scalability to outsourcing their data to remote cloud and surveys connectivity between cloud roles and external machines, and the construction of firewall and load balancing in the areas of secure application connectivity and federated access control via Service Bus and Access Control services. The contribution of the paper is to understand how the on-premises service connects to the relay service through an outbound port and creates a bidirectional socket for communication tied to a particular assignation address, the two main areas of compute and storage functionality as a cloud service and cloud computing platform, respectively.

6. REFERENCES

1. Cloud Security Alliance, “Security Guidance for Critical Areas of Focus in Cloud Computing V3.0,” released in 2009, permanent archives at http://www.cloudsecurityalliance.org/guidance/csaguide.v. 0.pdf D. Catteddu and G. Hogben, “Cloud Computing: Benefits, Risks and Recommendations for Information Security,” ENISA, 2009; http://www.enisa.europa.eu/act/rm/files/deliverables/cloud-computing-risk- assessment/at_download/fullReport.
2. "Architecture and Applications of a Versatile Small- Cell, Multi-Service Cloud Radio Access Network Using Radio-over-Fiber Technologies "Liang Zhang is currently with the State Key Lab of Advanced Optical Communication Systems and Networks, Shanghai Jiao Tong University, Shanghai, China 200240.
3. Kartik Sharma, Renuka Sharma, Gitesh Dalal, “A Secure Protocol for Data storage Security in cloud computing”- International Journal of Scientific & Engineering Research, Volume 4, Issue 6, June-2013 2348 ISSN 2229-5518.
4. Hassan Takabi and James B.D. Joshi, University of Pittsburgh , Gail-Joon Ahn, Arizona State University, “Security and Privacy Challenges in Cloud Computing Environments”- THE IEEE COMPUTER AND RELIABILITY SOCIETIES, Nov/Dec 2010, pp.24-31,1540-7993/10.
5. A Developer’s Guide to Service Bus in Windows Azure AppFabric..http://go.microsoft.com/fwlink/?LinkID=1504
6. A Developer’s Guide to Access Control in Windows AzureAppFabrichttp://go.microsoft.com/fwlink/?LinkID= 150835, http://www.windowsazure.com
7. Hung-Chang Hsiao, Tainan, Hsueh-Yi Chung, National Cheng Kung University, Tainan, Haiying Shen, Clemson University, Clemson, Yu-Chang Chao, Industrial Technology Research Institute South, Tainan, “ Load Rebalancing for Distributed File Systems in Clouds”- IEEE Transactions on Parallel and Distributed Systems (TPDS) , May 2013, Volume 24, Issue 5, pp. 951-962, ISSN : 1045-9219
8. David Chappell is Principal of Chappell & Associates (www.davidchappell.com) in San Francisco, California, Windows%20SSO%20v1%200-- Chappell.pdf.
9. "Network Implications of Cloud Computing" 2011 Technical Symposium at ITU Telecom World.