Volume 4, Issue 3, March 2014 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Multicloud Collaboration: from Cloud Mashups to Cloud Proxies

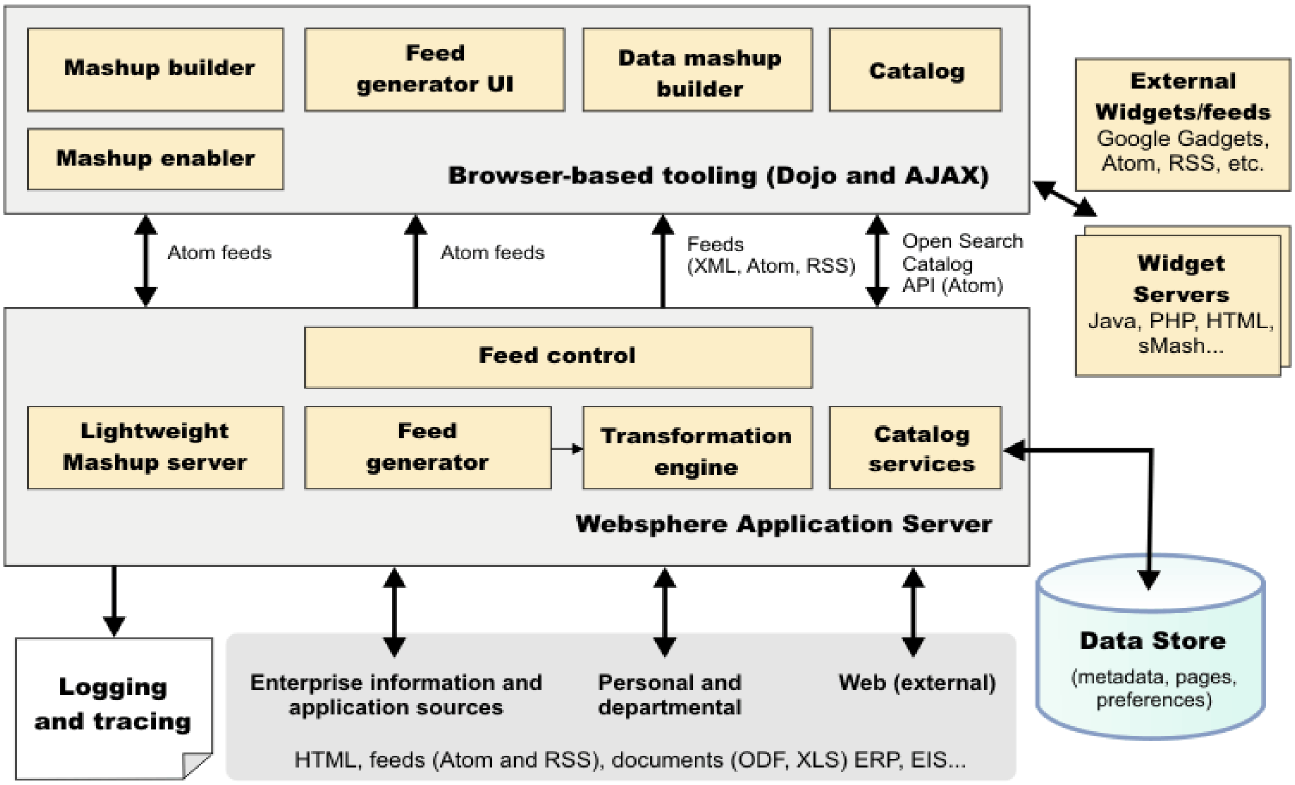
Dayakar Tammineni, Srinivasulu Pathakamuri
Department of CSE,
Visvodaya Engineering College, India

Abstract— Multicloud Collaborations are common in current day perspective using Cloud Mashups. However, to implement Cloud Mashups required preestablished agreements among providers. A proposed proxy-based multicloud framework using Cloud Proxies allows dynamic collaborations without preestablished collaboration agreements or standardized interfaces

Keywords—Cloud Mashups, Multicloud Collaborations, Cloud Proxies

I. INTRODUCTION

Cloud computing involve service providers, resource providers, and service client. Cloud Computing deliver applications as services such as Software as a Service (SaaS), Infrastructure as a Service (IaaS), and Platform as a Service (PaaS). As more and more organizations adopt cloud computing, cloud service providers (CSPs) are developing new technologies to enhance the cloud's capabilities. Cloud mashups combine services from multiple clouds into a single service or application. This service composition lets CSPs offer new functionalities to client at lower development costs. Consider an example cloud mashup IBM Mashup Center. IBM Mashup Center is an end-to-end enterprise mashup platform that enables the rapid, sharing, and discovery of reusable application building blocks such as widgets, feeds and mashups that can be easily assembled into new applications or leveraged within existing applications. Consider the Architecture of IBM Mashup Center.


Mashup builder
Feed generator UI
Data mashup builder
Catalog
Mashup enabler
Browser-based tooling (Dojo and AJAX)
External Widgets/feeds Google Gadgets, Atom, RSS, etc.
Atom feeds
Atom feeds
Feeds (XML, Atom, RSS)
Open Search Catalog API (Atom)
Widget Servers Java, PHP, HTML, sMash...
Feed control
Lightweight Mashup server
Feed generator
Transformation engine
Catalog services
Websphere Application Server
Logging and tracing
Enterprise information and application sources
Personal and departmental
Web (external)
HTML, feeds (Atom and RSS), documents (ODF, XLS) ERP, EIS...
Data Store (metadata, pages, preferences)

The IBM Mashup Center is designed with the goal of helping users at all skill levels create simple web applications from existing information sources by dragging and dropping widgets onto the page, and then wiring them together on-the-glass.

However, cloud mashups require preestablished agreements among providers as well as the use of custom built, proprietary tools that combine services through low-level, tightly controlled and constraining integration techniques. The research community is beginning to develop architectures, technologies, and standards to support collaboration among multiple cloud systems. However, these research proposals still remain constraining due to their provider-centric approach or limited scope.

While cloud standardization will promote collaboration, there are several hurdles to its adoption. From a market perspective, it is unlikely that multiple CSPs will agree on an easy and standardized way to access services, as this would give clients total freedom in changing providers, leading to increased open and direct competition with other providers. For cloud collaboration to be viable in the current environment researchers need to develop mechanisms that allow opportunistic collaboration among services without requiring standards and extensive changes to the cloud service delivery mode. This approach will allow incremental provisioning of collaborative services to client, which will continue to improve as more cloud services become interoperable in the future.

Some specific security issues associated with collaboration among heterogeneous clouds include

- Maintaining privacy of data and identity during collaboration
- Establishing trust among different cloud providers to encourage collaboration
- Addressing policy heterogeneity among multiple clouds

II. MULTICLOUD SYSTEMS COLLABORATION

Our proposed framework for generic cloud collaboration allows clients and cloud collaboration allows clients and cloud applications to simultaneously use services from and route data among multiple clouds. This framework supports universal and dynamic collaboration in a multicloud system. A proxy cloud is an edge node hosted software instance that a client or a CSP can delegate to carry out operations on its behalf. Depending on the context, the system can regard a network of proxies as a collection of virtual software instances connected via a virtual network or a set of physical nodes connected via an underlying network infrastructure.

The basic idea it to enable proxies that act on behalf of a subscribing client or a cloud to provide a diverse set of functionalities. As an example of proxy facilitated collaboration between clouds, consider a case in which a client or CSP wishes to simultaneously use a collection of services that multiple clouds offer. These proxy clouds can further delegate to other proxies if necessary and initiate the service request.

III. PROXY CLOUD ARCHITECTURES

Clouds consist of multiple network connected resource clusters such as servers, data warehouses that host geographically distributed virtual machines and storage components that ensure scalability, reliability, and high availability. A multicloud system that employs proxies for collaboration consists of three architectural components: multiple cloud computing systems, networks of proxies, and clients.

Proxy hosted by Cloud: As shown in Figure 1, each CSP can host proxies within its cloud infrastructure, manage all proxies within its administrative domain, and handle service requests from clients that wish to use these proxies for collaboration. The proxy instances might need to be CSP-specific.

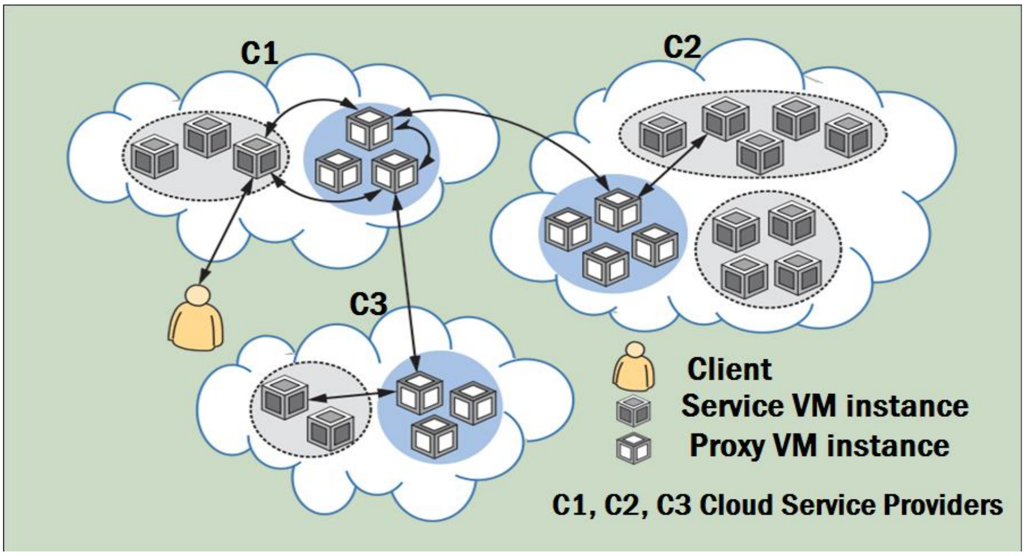


Figure 1. Client sends a request to cloud C1, which dynamically discovers the needs to use services from clouds C2 and C3. C1 employs proxies to manage these interactions.

Proxy as a Service: As shown in Figure 2, this scenario involves deploying proxies as an autonomous cloud that offers collaborative services to clients and CSPs. A group of CSPs that are willing to collaborate can manage this proxy-as-a-service cloud. A proxy service provider (PSP), can provide management. Clients directly subscribe to the proxy cloud service and employ them for intercloud collaboration.

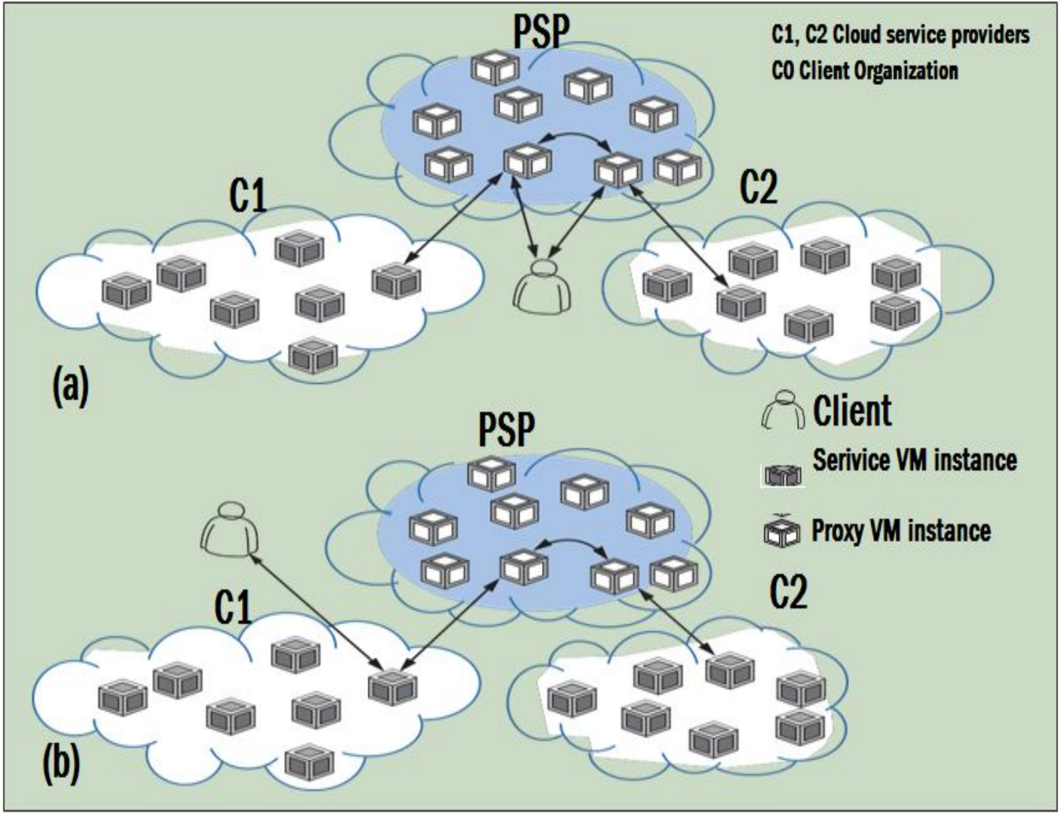


Figure 2. Proxy as a service. In this scenario, cloud service providers (CSPs) deploy proxies as an Autonomous cloud system and offer it as a service to clients. (a) A client employs two proxies to Interact with CSPs C1 and C2. (b) Alternatively, a client initiates a service request with C1, which Then discovers the need for a service from C2.

IV. SECURITY ISSUES IN MULTICLOUD COLLABORATION

Researchers and industry specialists have highlighted several security issues in cloud computing, including isolation management, data exposure and confidentiality, virtual OS security, trust and compliance, and mission assurance. Specific security issues emerge during dynamic sharing and collaboration across multiple clouds. In particular, issues pertaining to trust, policy, and privacy are a concern in multicloud computing environments.

Establishing trust and secure delegation

Security in clouds relies heavily on establishing trust relationships among the involved entities. The need for trust arises because a client relinquishes direct control of its assets security and privacy to a CSP. Doing so exposes a client's assets to new risks that are otherwise lessened or avoidable in an internal organization. These risks include insider security threats, weakening of data ownership rights, transitive trust issues with third-party providers in composite cloud services, and diminished oversight of system security.

From the client's point of view, employing on-premises proxies that are within the client's administrative domain can exacerbate trust issues. By using on-premises proxies a client maintains control over its assets while proxies process them during a collaborative service request. Proxy networks are a potential platform for developing proxy-based security architectures and solutions for multicloud systems. In addition, clients, clouds and proxies must implement mechanisms that ensure secure delegation, which entails the following:

- Fast agreements: Delegating to a proxy must establish, fast agreement between the delegator and proxy that lets the proxy act on the delegator's behalf. Techniques for delegation to a proxy must include mechanisms that restrict the proxy's behaviour, including data and resource access, to comply with delegator-specified constraints.
- Deviation from the expected behaviour: After delegation, a proxy must not deviate from the expected behaviour. It must act only on behalf of the delegator. After the proxy fulfils the service request, it can no longer act on the delegator's behalf. The proxy cannot modify the intended service request or misuse client assets, and it must not transitively delegate its capabilities to other proxies without the delegator's explicit consent.

Proxy heterogeneity and conflicts

When proxies enable dynamic collaboration between multiple CSPs, heterogeneous security policies can be the source of policy conflicts that result in security breaches. Proxies must monitor for and defend against such breaches. Even though existing policy evaluation mechanisms can verify individual domain policies, security violations can easily occur during integration. In multicloud collaboration using proxies, service requirements can drive dynamic, transient, and intensive interactions among different administrative domains. Thus, a proxy's policy integration tasks must address challenges such as semantic heterogeneity, secure interoperability, and policy evolution management.

Policy analysis generally includes property verification and conflict detection, as well as an analysis of the differences between policy versions. Policy integration aims to generate agreement on access rights for each party involved in a collaborative project. To avoid conflict detection, conflict resolution mechanisms must be used. For example, existing conflict resolution mechanisms such as Extensible Access Control Markup Language (XACML) policies can be used. However XACML policies are too restrictive because they only allow the selection of one resolution algorithm to resolve all identified conflicts.

Data Protection as a Service (DPaaS) enforces fine grained access control policies on data units through application confinement and information flow checking. DPaaS can achieve the solutions for proxy heterogeneity and conflict resolution management. DPaaS can accomplish user authentication either with a proprietary approach or using open standards such as OpenID and OAuth.

OpenID is an open standard that allows users to be authenticated by certain cooperating sites using a third party service, eliminating the for webmasters to provide their own ad hoc systems and allowing users to consolidate their digital identities. Users may create accounts with their preferred OpenID identity providers, and then use those accounts as the basis for signing on to any website which accepts opened authentication. The OpenID standard provides a framework for the communication that must take place between the identity provider and the OpenID acceptor. An extension to the standard facilitates the transfer of user attributes, such as name and gender from the OpenID identity provider to the relying party

OAuth is an open standard for authorization. OAuth provides a method for clients to access server resources on behalf of a resource owner such as a different client or an end-user. It also provides a process for end-users to authorize third-party access to their server resources without sharing their credentials, typically a username and password pair, using user-agent redirections. OAuth is a service that is distinct from OpenID.

REFERENCES

[1] Collaboration Chandrasekhar S and Singhal M *Collaboration in Multicloud Computing Environments: Framework and Security Issues,* IEEE Transactions on Cloud Computing Vol.46 No.2 Year 2013

[2] P. Mell and T. Grance, *The NIST Definition of Cloud Computing*, special publication 800-145, National Inst. Standards and Technology, 2011, p. iii+3

[3] R Buyya et al., *Market-Oriented Cloud Computing: Vision, Hype, and Reality of Delivering Computing as the 5thUtility* Proc. 9th IEEE/ACMInt'l Symp, *Cluster Computing and the Grid (CCGRID 09)*, IEEE CS 2009, pp. 599-616

[4] M.P. Papazogulu and W. Vanden Heuvel, *Blueprinting the Cloud*, IEEE Internet Computing, No 2011, pp. 74-79.

[5] S. Chandrasekhar et al., *Efficient Proxy Signatures Based on Trapdoor Hash Functions IET Information Security,* Dec. 2010, pp. 322-332

[6] N.R. Adam and J.C. Wortmann, *Security-Control Methods for Statistical Databases: A Comparative Study*, ACM Computing Surveys, Mar. 1089, pp. 515-556.

[7] L. Xiong S. Chitti and L.Liu, *Preserving Data Privacy in Outsourcing Data Aggregations Services* ACM Trans. Internet Technology, Aug. 2007, p. 17.

[8] E. Hammer-Lahav, ed., *The OAuth 1.0 Protocol*, IETF RFC 5849, Apr. 2010; http://tools.ietf.org/html/rfc5849