Available online at www.sciencedirect.com
SCIENCE DIRECT®
Data & Knowledge Engineering 58 (2006) 381–409
DATA & KNOWLEDGE ENGINEERING
www.elsevier.com/locate/datak

Three architectures for trusted data dissemination in edge computing

Shen-Tat Goh [a,*], HweeHwa Pang [a], Robert H. Deng [b], Feng Bao [a]

[a] *Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore 119613, Singapore*
[b] *Singapore Management University, 469 Bukit Timah Road, Singapore 259756, Singapore*

Received 15 April 2005; accepted 7 May 2005
Available online 14 June 2005

Abstract

Edge computing pushes application logic and the underlying data to the edge of the network, with the aim of improving availability and scalability. As the edge servers are not necessarily secure, there must be provisions for users to validate the results—that values in the result tuples are not tampered with, that no qualifying data are left out, that no spurious tuples are introduced, and that a query result is not actually the output from a different query. This paper aims to address the challenges of ensuring data integrity in edge computing. We study three schemes that enable users to check the correctness of query results produced by the edge servers. Two of the schemes are our original contributions, while the third is an adaptation of existing work. Our study shows that each scheme offers different security features, and imposes different demands on the edge servers, user machines, and interconnecting network. In other words, all three schemes are useful for different application requirements and resource configurations.

Keywords: Data integrity; Edge computing; Data dissemination; Replication

* Corresponding author. Tel.: +65 6874 8457; fax: +65 6776 8109.
E-mail addresses: stgoh@i2r.a-star.edu.sg (S.-T. Goh), hhpang@i2r.a-star.edu.sg (H. Pang), robertdeng@smu.edu.sg (R.H. Deng), baofeng@i2r.a-star.edu.sg (F. Bao).

0169-023X/$ - see front matter
doi:10.1016/j.datak.2005.05.003

1. Introduction

Many Web services are served from central locations, and could suffer from a number of bottlenecks ranging from Web and database server loads, to network delays. Server overloads can usually be alleviated through load balancing on a server farm. In contrast, network latencies are usually beyond the control of the Web service operators, as traffic to and from remote users has to pass through long-haul networks operated by multiple network providers that are often congested. Although aggressive build-up in recent years by telecommunication companies has expanded the capacity of the long-haul networks, new technologies like Gigabit ethernet are making bandwidth much more affordable in the Metropolitan Area Networks (MAN). Given the relative price-performance of Wide Area Networks (WAN) versus MAN, the logical approach to reduce network latency is to push the Web services to the users, into the MANs.

Edge computing is being promoted as a strategy to achieve scalable and highly available Web services (e.g., [1]). Fig. 1 shows the generic architecture of an edge computing platform. It pushes business logic and data processing from central data centers, out to proxy servers at the "edge" of the network and within the MANs. There are several potential advantages: Running applications at the edge cuts down network latency and produces faster responses to user applications and partners' Web services. Adding edge servers near user clusters is also likely to be a cheaper way to achieve scalability than fortifying the servers in the central data center and provisioning more network bandwidth for every user. Finally, by lowering the dependency on a central data center, edge computing removes the single point of failure in the infrastructure, reducing its susceptibility to denial of service attacks and improving service availability.

In theory, edge computing is a natural extension of the Content Delivery Network (CDN) architecture [2,3]. In practice, pushing application logic to edge servers introduces a number of

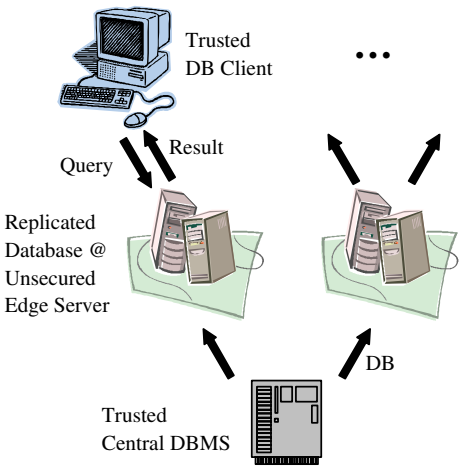

Fig. 1. Edge computing set-up.