Journal of Network and Computer Applications 50 (2015) 80–91

Contents lists available at ScienceDirect

Journal of Network and Computer Applications

journal homepage: www.elsevier.com/locate/jnca

SIONA: A Service and Information Oriented Network Architecture

Mingwei Xu [a], Zhongxing Ming [a,*], Chunmei Xia [a], Jia Ji [a], Dan Li [a], Dan Wang [b]

[a] Department of Computer Science and Technology, Tsinghua University, China
[b] Department of Computing, The Hong Kong Polytechnic University, Hong Kong

ARTICLE INFO

Article history:
Received 31 January 2014
Received in revised form
14 May 2014
Accepted 30 June 2014
Available online 26 July 2014

Keywords:
Network architecture
Future internet
Information centric networking
Caching

ABSTRACT

The Internet is a great hit in human history. However, it has evolved greatly from its original incarnation. Content distribution is becoming a central role in today's Internet, which makes it difficult for the conventional host-to-host communication to meet the ever-increasing demands. In this paper, we present a Service and Information Oriented Network Architecture (SIONA) to fully optimize resources wherever they are. We describe SIONA's key design elements and illustrate how SIONA facilitates service-aware communication, name-based content delivery and mobility support, while keeping the network scalable and efficient. We implement *name-based sockets* (NBS) as a new transport API for applications to use names instead of addresses. We develop a lightweight *age-based cooperative caching scheme* to improve content distribution. We build an example system to show the feasibility and ease of use of SIONA. We conduct experiments in real networks to evaluate SIONA's transportation performance and perform a trace-based simulation to evaluate our caching scheme. Results show the advantage of our approach.

1. Introduction

Current Internet adopts a point-to-point communication model that emphasizes the end-to-end paths or channels. Packets carrying addresses simply flow in the channel. This model has achieved tremendous success. However, demands have changed tremendously since the Internet was created.

As an example, let us imagine an undergraduate in 1991 procrastinating instead of writing his junior paper. He wants to use the University's newfangled computer to test out a recently released application called WorldWideWeb. The student opens the application and types in http://www.princeton.com into his web browser. The machine then performs a lookup and finds the corresponding IP address.[1] The browser crafts its http request and sends it off to the server's IP address. The request is chunked into packets which travel through the hierarchical network to the server. It responds and similarly sends the requested page back to the student's IP. After not very long, the student realizes that there is very little content on the internet and promptly finishes his paper.

Today however, the world is a little bit different. It is much easier to imagine the procrastinating junior watching YouTube videos on his phone while walking. In this situation, he too starts by typing the web address into the computer (if you want to call it that) and it performs a lookup for YouTube. Although the DNS lookup returns an IP address, the similarities between the two scenarios diverge largely. That lookup does not return the address of a server YouTube.com but rather it may return one of many possible IP addresses that will each take him to YouTube. Additionally that IP address may not map to one server but rather there may be many YouTube servers all across the internet that are all claim to be that IP. As silly as this example may be, it exemplifies the ways in which the modern applications of the internet rely on techniques and protocols that violate the abstractions on which the network was built.

The economics have changed the possible starting points for the design of the Internet. Computing and storage resources are not scarce anymore. Location looses its meaning because the primary concern of users has become *WHAT* rather than *WHO* and *WHERE*. New requirements, such as mobility, have become prevalent for both end-users and service providers. The end-to-end principle has significant limitations when dealing with massive content delivery and supporting ubiquitous mobility. The network that provides access to services and information has become unnecessarily complex because of its inability to evolve.

In this paper, we present a Service and Information Oriented Network Architecture (SIONA) that provides a bold yet remarkably

* Corresponding author.
E-mail addresses: xmw@mails.tsinghua.edu.cn (M. Xu), mzx09@mails.tsinghua.edu.cn (Z. Ming), xcm@mails.tsinghua.edu.cn (C. Xia), jjia@mails.tsinghua.edu.cn (J. Ji), lidan@mails.tsinghua.edu.cn (D. Li), csdwang@comp.polyu.edu.hk (D. Wang).

[1] Although DNS technically had been published this almost certainly would have been from a static hosts file.

http://dx.doi.org/10.1016/j.jnca.2014.06.010
1084-8045/

simple augmentation of the existing overloaded stack. Our design objective is as follows:

- *Service and information oriented communication*: Massive services and information have changed the attitude of users. The design should enable users to communicate with services and getting information wherever they are.
- *Efficient data delivery*: Millions of users are requesting billions of content objects on a daily basis. Content levels are exploding exponentially. The design should support content transmissions effectively.
- *Scalability*: The number of world-wide data pieces is expected to be on the order of 10^{16} (Gantz et al., 2008). To cope with these contents, the system should be extremely scalable.
- *Inherent mobility support*: We have entered the age of mobile Internet. Current Internet protocols are not well-suited for mobile services. The architecture should provide ubiquitous mobility support.
- *Backwards-compatibility*: While a clean-slate solution may fundamentally solve the problems, it is painful to replace a sea of infrastructures and change the transport mechanisms of existing applications. It is desirable that a new architecture has backwards-compatible approaches to simplifying network infrastructure deployment.

We meet the above goals by the following means. First, we propose a name-based paradigm in which services and information are identified, published and subscribed by names. We define Information as static data, for example, a video file, and define Service as the functionalities that meet customers' demands, e.g., streaming services via YouTube. In SIONA, publishers register services and information to the network. Subscribers achieve contents by issuing request packets which contain the names of the service and information. The request packet is routed by its name to an arbitrary node that provides the desired data (like a service level anycast). We implement a new transport interface, called *name-based sockets (NBS)*, to support name based communications. We conduct experiments in real networks to demonstrate the effectiveness of NBS.

Second, we exploit a name-based two-dimensional routing, which leverages service names for inter-domain routing and information names for intra-domain routing. The underlying philosophy of two-dimensional routing is the emphasis on scalability and mobility, which will be explained in detail in Sections 2 and 3. We retain the Forwarding Information Base (FIB) of regular IP routers to forward packets based on the existing IP network, which achieves backwards-compatibility with today's IP network.

Finally, we use in-network caching to improve massive data distribution. In-network caching has the potential to improve network efficiency and content distribution performance. We develop a lightweight *age-based cooperative caching scheme (ABC)* in hopes of achieving the attractive features of in-network caching. We use web traces and real network topology to simulate ABC's performance and demonstrate that it noticeably outperforms traditional caching algorithms such as Least Recently Used (LRU).

Although previous works consider some of the problems we address, SIONA differs from them in many ways. Dona (Koponen et al., 2007) decouples a content from its location by using flat names, but clients in Dona can only benefit from caching locations along the path towards the root of the RH hierarchy. NDN (http://www.named-data.net/ndn-proj.pdf) proposes a new architecture in which communication is driven by named data. However, it is based on hierarchical names and is not service-aware. It also has difficulty in being compatible with current IP network. PURSUIT (http://www.fp7-pursuit.eu) builds a publish/subscribe-based Internet system, but limits the caching functionality to the scope of the rendezvous point. Serval (Nordström et al., 2012) develops a service-oriented network stack, but retains the host-to-host communication model after the routing of the first packet. ID/LOC split schemes (Farinacci et al., 2011) separate the identifier of a host from its location, but is based on the model of a flow of bytes from a source to a destination. In contrast, SIONA provides a coherent solution for service-and-information-oriented networking that a simple composition of previous solutions cannot achieve.

The rest of this paper is organized as follows. Section 2 presents SIONA's foundational ideas. Section 3 presents SIONA's architecture design. Section 4 presents the implementation elements, including packet formats, name-based sockets, router design and an age-based cooperative caching scheme. Section 5 builds a SIONA example system, conducts real scenario experiments to compare the performance of SIONA to regular TCP/IP and evaluates the proposed caching scheme using trace-based simulation. Section 6 discusses the scalability and consistency issues. Section 7 presents the related work. Section 8 concludes our work and points out future research directions.

2. Foundational ideas of SIONA

2.1. Name-based paradigm

SIONA is designed to be a service-and-information-oriented platform. Communication is driven by names rather than IP addresses. Applications can use names to directly express their intent to publish or access specific contents. SIONA explicitly distinguishes services from information, which means service and information are identified by different names. Service/information owners bind to names and announce names to the network. Subscribers achieve contents by subscribing named data instead of connecting to IP addresses. Such a service and information split scheme is designed to meet the goal of efficient content delivery, scalable routing and ubiquitous mobility support, as we will explore in detail in Sections 3 and 6.

2.2. Two-dimensional routing

SIONA exploits a name-based two-dimensional routing. The meaning of two-dimensional routing is twofold: (1) subscribers first use the service name to find the publisher, and then use the information name to achieve the content object, and (2) inter-domain routing is performed by service names and intra-domain routing is performed by information names. The design of explicitly distinguishing service and information is driven by the following motivations: (1) *Efficient and effective data delivery*: It is common that multiple copies of a content item are available in the network. Manipulating services independently avoids redundant consumption of service node resources (e.g., CPU cycles, memory, energy) (Shanbhag et al., 2011). (2) *Service migration*: The emergence of cloud computing and virtualization technology makes it trivial that the location of service dynamically changes. Extracting services from contents will benefit routing discovery when service migration happens. (3) *Scalable content routing*: Maintaining 10^{16} content names in routing tables cannot handle the necessary scales. Separating service from information enables the system to dramatically reduce the routing table by constructing a name hierarchy. We will explain the details of two dimensional routing in Section 3 and discuss the scalability issues in Section 6.1.