Signposts: End-to-End Networking in a World of Middleboxes

Andrius Aucinas, Amir Chaudhry,
Jon Crowcroft, Sebastian Probst Eide,
Steve Hand, Anil Madhavapeddy,
Andrew W. Moore, Charalampos Rotsos,
Narseo Vallina-Rodriguez
University of Cambridge, UK
first.last@cl.cam.ac.uk

Richard Mortier
University of Nottingham, UK
richard.mortier@nottingham.ac.uk

ABSTRACT

This demo presents Signposts, a system to provide users with a secure, simple mechanism to establish and maintain communication channels between their personal cloud of named devices. Signpost names exist in the DNSSEC hierarchy, and resolve to secure end-points when accessed by existing DNS clients. Signpost clients intercept user connection intentions while adding privacy and multipath support. Signpost servers co-ordinate clients to dynamically discover routes and overcome the middleboxes that pervade modern edge networks. The demo will show a simple scenario where an individual's personal devices (phone, laptop) are interconnected via Signposts while sitting on different networks behind various middleboxes. As a result they will be able to fetch and push data between each other, demonstrated by, e.g., simple web browsing, even as the network configuration changes.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Design**]: Distributed networks; C.2.2 [**Network Protocols**]: Routing protocols

General Terms

Design, Experimentation, Measurement

Keywords

Naming, DNS, Middlebox, User-centered, Edge network

1. SIGNPOSTS AND EFFECTFUL NAMES

The modern Internet has broadly divided into two halves: a performant, global, core network, and an edge network through which end-users access services. Devices in the core network typically route to each other freely, and major content providers form significant networks in their own right (e.g., Facebook and Google), often referred to as "clouds".

However, activities that are simple in the cloud – such as establishing peer-to-peer links – are disproportionately complicated in the edge. A typical home network is obscured from the cloud by firewalls, Network Address Translators (NATs), and proxies, all of which complicate the creation of incoming connections. As a result, when a user wants their devices to communicate the only practical option is to route all content and communication via the cloud. There are substantial downsides to routing via the cloud as compared to ad-hoc channels over local networks – e.g., inability to operate without an Internet connection, increased potential for privacy violation, the relative lack of bandwidth and higher latency, as well as energy and financial costs [6, 2].

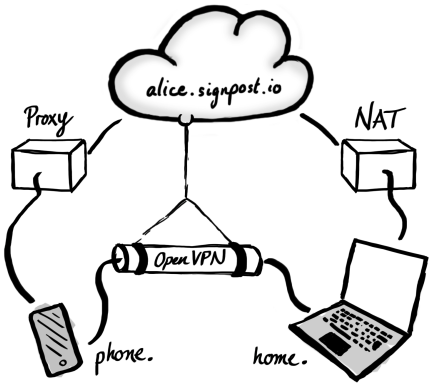

Figure 1: Schematic of local channel creation via a cloud-based Signpost server. Alice's machine `home` attempts to resolve `phone.alice.signpost.io`, which results in a route being created between the two devices (e.g. OpenVPN).

These factors all motivate the need for *Signposts*, our network service that enables user-centric control of connectivity in the edge network, both within a personal cloud of devices and across to the devices of others. Such a service must account for the need to operate when disconnected from the global Internet, whether due to intermittent connectivity or the interposition of middleboxes that break the all-pairs connectivity model of the traditional Internet.

Signposts automate the process of establishing and maintaining routes across the modern Internet by exploiting DNS as a ubiquitous signalling channel (§3). Every user registers a unique domain name (e.g., `alice.signpost.io`) and binds stable names to their devices as a one-off process (e.g., `phone` and `home`). DNS responses for these domains are served by authoritative Signpost servers running in the cloud and on

SIGCOMM'12, August 13–17, 2012, Helsinki, Finland.
ACM 978-1-4503-1419-0/12/08.

their home networks. This infrastructure takes care of resolving names into network addresses and setting up routes between devices. Since middleboxes make it impossible to predict a network's capabilities ahead of time, the act of DNS name resolution triggers actions that automatically setup appropriate NAT traversal, VPNs and proxies between clients and servers: the act of resolving a device's name has side-effects in the network, a scheme reminiscent of circuit-switched networks [5] and illustrated in Figure 1.

2. DEMO: ALICE'S PERSONAL CLOUD

Signposts are concerned solely with establishing communication channels between devices, not with the transport of data over those channels. Consider the simple demo scenario depicted in Figure 1. Alice has a Signpost server running in a globally visible location in the public cloud. This serves her public key and zone, `alice.signpost.io`, with DNSSEC providing a chain of signed attestations back to the DNS root that the record has not been tampered with en route. Alice has two network-connected devices to which she has bound the concrete names `phone` and `home` and for which the server will coordinate name resolution.

Automatically establishing a connection between `phone` and `home` via the Signpost then becomes relatively straightforward. One client, say `home`, initiates the process by attempting a DNS resolution of `phone.alice.signpost.io`. Normal DNS mechanisms cause this query to reach Alice's Signpost, which has been delegated the zone `alice.signpost.io`. The Signpost resolves the name `phone` by probing and establishing routes using a bidirectional signalling channel established between the Signpost and Alice's devices. The Signpost dynamically probes a variety of channels to determine whether multiple routes can be established, and create them if so. The result is that various channels (e.g., VPN, TOR, etc) are created between Alice's phone and computer, and at least one valid IP address endpoint returned in the DNS query if any viable route exists.

Without Signposts, Alice would have had to manually configure a port-forward at her NAT box, or run VPN software on her phone. Not normally visible to users, the demo incorporates a lightweight mechanism to display path existence and properties.

3. DISCUSSION

Why DNS? Most devices used to access the Internet are essentially anonymous from a network perspective, with only transient names if they have names at all. The Signpost network assigns secure, stable names to each device, and provides a way of resolving these names into concrete network addresses. Signposts extend the DNS protocol [7] for this purpose, for the following reasons:

- *Ubiquity.* DNS is among the most widely deployed services on the Internet. Effectively every Internet-connected client supports name resolution, and has access to the DNS when connected.
- *Reach.* As such a critical part of the Internet's infrastructure, and unlike TCP, HTTP and similar protocols, DNS tends not to be manipulated by middleboxes other than modified DNS servers themselves [3, 4].
- *Security.* The DNSSEC security extensions have recently been deployed on the live root servers [1]. DNSSEC provides origin authentication and integrity protection for DNS records, and (along with SSL) represents one of the two global public key infrastructures.

Identity. The Signpost system requires individuals to register their own domain. Thus, by running a Signpost server which has been delegated their zone, an individual has an authenticated public identity on the Internet via their public-private key-pair, using standard DNSSEC records. This pushes user security credentials into the naming fabric of the Internet, meaning that any other network service which can perform name lookups can efficiently verify a user key.

Multipath. In addition, Signposts solves the problem of multipath route discovery as name resolution results in dynamic probing. This integrates well into end-point upgrades such as multipath TCP.

4. SUMMARY

A feature of the development of the Internet has been the gradual erosion of the end-to-end principle. In a world of mobile gateways, NATs and other middlebox impositions, it is less clear what it means for the "ends" to connect. Signposts is a system which allows us to evolve past this current, rather unsatisfactory, state of affairs. It provides a control plane that enables *edge-to-edge* connectivity, between the many heterogeneous and often mobile edge networks that have now sprung up around the Internet.

More information and source code is available via `http://signpost.io`

5. ACKNOWLEDGEMENTS

This work was partially funded by the RCUK Horizon Digital Economy Research Hub grant, EP/G065802/1.

6. REFERENCES

[1] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose. DNS Security Introduction and Requirements. RFC 4033, IETF, March 2005.

[2] J. Baliga, R. Ayre, K. Hinton, and R. Tucker. Green cloud computing: Balancing energy in processing, storage, and transport. *Proceedings of the IEEE*, 99(1):149 –167, January 2011.

[3] B. Carpenter and S. Brim. Middleboxes: Taxonomy and Issues. RFC 3234, IETF, February 2002.

[4] M. Honda, Y. Nishida, C. Raiciu, A. Greenhalgh, M. Handley, and H. Tokuda. Is it still possible to extend TCP? In *Proceedings of the 11th ACM SIGCOMM Internet Measurement Conference*, IMC '11, pages 181–194, Berlin, Germany, November 2011. ACM.

[5] I. M. Leslie. Extending the local area network. Technical Report UCAM-CL-TR-43, University of Cambridge, Computer Laboratory, 15 JJ Thomson Avenue, Cambridge CB3 0FD, United Kingdom.

[6] A. Li, X. Yang, S. Kandula, and M. Zhang. Cloudcmp: comparing public cloud providers. In *Proceedings of the 10th ACM SIGCOMM Internet Measurement Conference*, IMC '10, pages 1–14, Melbourne, Australia, November 2010. ACM.

[7] P. Mockapetris. Domain names – concepts and facilities. RFC 1034, IETF, November 1987.