DEMO: Easy Deployment of a Secure Internet Architecture for the 21st Century

How hard can it be to build a secure Internet?

Ercan Ucan
ETH Zurich
first.lastname@inf.ethz.ch

Raphael M. Reischuk
ETH Zurich
lastname@inf.ethz.ch

Adrian Perrig
ETH Zurich
first.lastname@inf.ethz.ch

ABSTRACT

We propose a demonstration of *SCION*, a future Internet Architecture designed for the 21st century. We demonstrate SCION's various rich features (including DDoS defense, native multipath communication, high-speed anonymous routing) and its ease of deployment.

1. MOTIVATION

The Internet as we use it today has excelled all expectations. It permeates nearly all aspects of modern society, to the extent that even a brief service interruption can have catastrophic consequences on government, economic, and social operations.

Unfortunately, the Internet was not designed with the increasingly adversarial challenges that are present today: malicious Internet providers can wreak havoc on communications; oppressive regimes can censor unwanted content; mass surveillance poses a threat to user privacy; criminal organizations can extort money from any company or organization in the world by launching comparably inexpensive denial-of-service attacks.

Over the past few decades, patches have been proposed to improve security and resilience to Internet attacks. However, these solutions have been constrained by the Internet's design, business model, and legal contracts.

In this demonstration, we show how to deploy a new Internet architecture, SCION, that is explicitly designed to bypass known security problems, that offers high availability, and that can scale to accommodate the needs of millions of users and devices for the next decades. More precisely, SCION aims at achieving the following goals: (1) high availability even in the presence of (distributed) adversaries; (2) transparency and control over Internet paths and cryptographic keys; (3) efficiency, scalability, and extensibility of the involved procedures; (4) support for heterogeneous trust; and (5) a feasible deployment strategy.

CCS'16 October 24-28, 2016, Vienna, Austria

ACM ISBN 978-1-4503-4139-4/16/10.

DOI: http://dx.doi.org/10.1145/2976749.2989036

Creating a replacement Internet architecture is challenging. In essence, we are trying to fundamentally change the operation of the largest man-made infrastructure. So it is reasonable to ask why such an all-encompassing revolution should be possible at all.

SCION offers benefits even for the first deployers. It offers possibilities for a single ISP when deployed within the ISP's network, as the ISP can offer services comparable to high-availability leased lines to its customers at very low cost. To support early deployment further, we have created a small box that transparently converts traffic between our architecture and the legacy Internet.

SCION has been designed according to the existing model of incrementally growing and extending the Internet's features. We are well underway to achieving this goal, having completed its initial design and the first deployment steps.

2. THE SCION INTERNET ARCHITECTURE

Before talking about the specifics of our demonstration, we provide a brief overview of SCION. SCION, which is also an acronym for Scalability, Control, and Isolation On next-generation Networks, has experienced more than 5 years of research [9, 8, 1, 2, 3, 5] with over 70 person-years of research and development.

The main building block for properties such as high availability, transparency, scalability, and support for heterogeneous trust is SCION's concept of *Isolation Domains (ISDs)*. An ISD logically groups a set of *Autonomous System (ASes)*. An AS is a network domain under a uniform administration. An ISD is administered by one or multiple such ASes, forming the *ISD Core*. An ISD contains one or multiple regular ASes. The ISD is governed by a policy that is negotiated by the ISD Core. This policy defines the roots of trust that are used to validate bindings between named entities and their public keys (certificates) or their addresses (DNS).

An AS wishing to join an ISD purchases service from an AS already in that ISD, and thereby accepts that ISD's policy. An AS can select any AS that is part of an ISD it desires to belong to. We expect that large ISPs constitute the ISD's Core ASes, and their associated customers would participate in the ISD as well.

The organization of ISDs is hierarchical; sub-ISDs are also possible in SCION. ISDs may also overlap: an AS may be part of more than one ISD. Although an ISD does provide *isolation* from other networks, the main goal of an ISD is to provide *transparency* and to support *heterogeneous trust*

environments as well as *openness*, as we will show in this demo.

Routing in SCION uses two levels: intra-ISD and inter-ISD. Both levels leverage *Path-segment Construction Beacons* (PCBs) for the discovery and the establishment of routing paths. The Core AS of an ISD announces a beacon and disseminates it as a policy-constrained multi-path flood, either *within* an ISD (to discover intra-ISD paths) or *amongst* ISD Core ASes (to discover inter-ISD paths). The path construction beacons collect AS-level path information (cryptographically protected at each hop) as they traverse the network. These tokens are chained together by a source to create data transmission paths that traverse a sequence of ASes. Packets in SCION contain such AS-level path information, which avoids the need to maintain inter-domain routing tables at border routers. This concept is referred to as *Packet-Carried Forwarding State*.

Thanks to the inter-domain beaconing process, Core ASes learn paths to every other Core AS. Through the intra-domain beaconing process, ASes learn paths on how to reach their ISD Core.

With the approach of *source-selected paths*, source nodes combine at most three path segments (up-segment, core-segment, and down-segment) to form valid end-to-end paths. A source node in SCION does not need to search any topology to find valid paths.

Multipath Communication. Today's Internet does not natively support multipath at the network layer. Recent research has enabled multipath at the transport layer (cf., MPTCP [7]), it requires endpoints to enable this modified transport in their network stacks. In contrast, the SCION network socket supports multipath *by default*, i.e., traffic is always forwarded over the k best (preferably disjoint) paths, where k is a configurable parameter assuming that the underlying topology supports the choice. Splitting traffic onto k disjoint paths not only increases the availability, but also hampers espionage and surveillance. The bottom left portion of Figure 1 shows measurements of an actual HTTP(S) request served via SCION multipath communication, as we will show in the demonstration.

Moreover, SCION's multipath socket offers *explicit route control* by avoiding/blacklisting certain regions, which is a unique feature that does not exist in today's Internet. The top-right corner of our SCION proxy and browser extension in Figure 1 contains the control panel for this feature, as we will show in the demonstration.

Extensibility via SCION Extensions. We demonstrate how SCION allows for a wide range of extensions that are easily integrated in the core architecture, be it for DDoS protection [2], high-speed anonymous communication [3], or bump-in-the-wire connectivity for endhosts [4].

3. EASE OF DEPLOYMENT

In this section, we present the four different scenarios by which SCION can be deployed and used. In our demo, we are going to focus on the first two of the items presented below and show their characteristics and potential.

(1) Native Deployment

By native SCION deployment, we refer to a SCION network which does not rely on any BGP-based information for providing end-to-end connectivity. Thus, connectivity will be

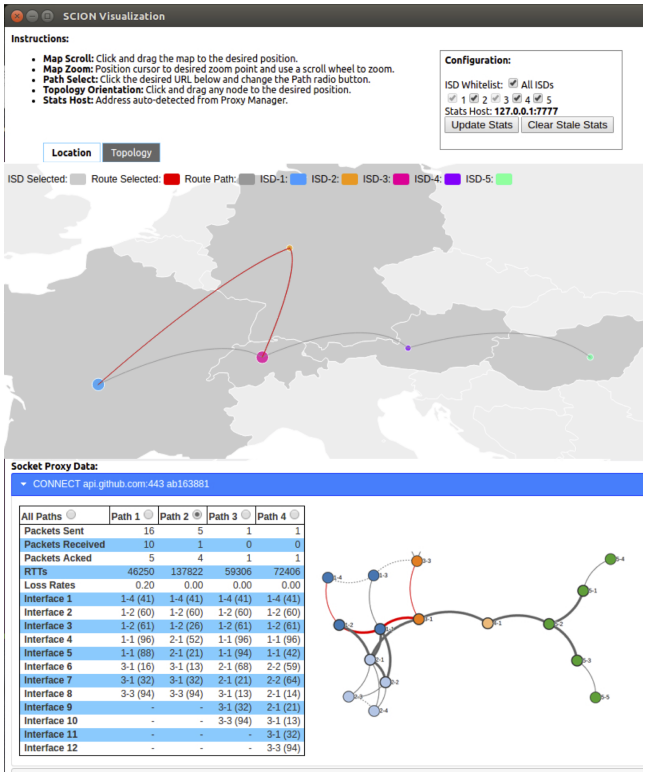

Figure 1: SCION's native multipath communication

assured regardless of what state the global BGP system may be in. A native SCION deployment requires dedicated physical machines installed in various ISPs. These machines run SCION infrastructure elements such as border routers, beacon servers, path servers, etc.

In a native deployment scenario, the security and availability properties that SCION offers would be at its highest. Therefore, a full SCION deployment at one or several directly connected ISPs will result in strong properties for communication amongst their respective customers. We refer to such contiguous deployments as *SCION islands*. In our early deployment, we have already one such SCION island in Switzerland, with the deployment of the ISPs *Swisscom* and *SWITCH*, in addition to our own infrastructure at ETH Zurich. In our demo, we will show an intuitive and easy-to-use management interface (cf., Figure 2) designed to simplify administration and to easily create and manage SCION ASes. In this setting, several corporations in Switzerland (including some prominent banks), interested in highly available and secure communication have also started test deployments using this infrastructure.

(2) SCION HTTP(S) Forward and Reverse Proxy

The SCION HTTP(S) Proxy, as depicted in Figure 3, is a program which can be used to browse the Web over a SCION infrastructure. The SCION proxy consists of two parts: Forward (Bridge) Proxy and Reverse Proxy. The two modules operate as follows: The forwarding proxy, which runs on an endhost, takes the incoming HTTP(S) requests from a standard web browser, such as Chrome or Firefox, and puts the requests onto a SCION multipath socket. The reverse proxy, running on a different endhost on the remote end, receives

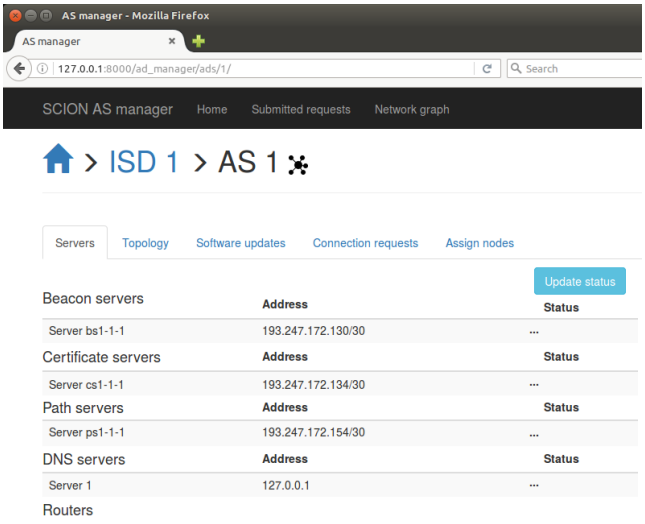

Figure 2: SCION's AS management web interface.

Figure 3: Deployment via SCION proxy.

this traffic on a SCION multipath socket and converts it back to HTTP(S) traffic. It then fetches the requested website from the targeted Web server, which may reside on the same machine as the reverse proxy, or on another location on the Internet.

In our demo, we will show our Chrome Web browser extension (cf., Figure 1), developed as a command and control center for the proxy, and also to visualize the traffic statistics and control various other things such as an ISD blacklisting feature.

(3) SCION VPN Gateway

SCIONizer, the SCION VPN Gateway, is a gateway appliance aimed for routing VPN traffic over the SCION network (cf., Figure 4). It receives VPN traffic (UDP) from off-the-shelf, commercial VPN software, converts it into SCION multipath-UDP (MPUDP) packets, and sends them over the SCION network. The peer (remote) SCION VPN Gateway residing on the remote endpoint receives these MPUDP packets over the SCION network, converts them back into UDP packets sent by the originator and delivers to the aimed VPN endpoint.

(4) DENA Box

Another approach we are working on for easy deployment of SCION involves a *bump-in-the-wire* interface device, which we refer to as DENA (Device for ENhancing Availability), to be placed between a customer's network and the Internet provider. The motivation for this work is that the end-users should be able to use SCION without carrying out complicated tasks, such as configuring their network devices or updating to a new network stack.

Figure 4: Deployment via SCION VPN Gateway.

The DENA device implements four main functionalities. For a given communication flow between a subscriber and a peer, it needs to: *1)* identify the presence of a peer (currently using steganographic techniques); *2)* if present, establish SCION path(s) to be used as fail-over; *3)* continuously measure the packet loss rate of the path(s); and *4)* fail-over to a SCION path if necessary. An alternative approach could be to deploy such interface devices at the ISPs themselves [6], completely removing end-user involvement.

4. REFERENCES

[1] David Barrera, Raphael M. Reischuk, Pawel Szalachowski, and Adrian Perrig. SCION five years later: Revisiting scalability, control, and isolation on next-generation networks. *arXiv e-prints*, 2015.

[2] Cristina Basescu, Raphael M. Reischuk, Pawel Szalachowski, Adrian Perrig, Yao Zhang, Hsu-Chun Hsiao, Ayumu Kubota, and Jumpei Urakawa. SIBRA: Scalable internet bandwidth reservation architecture. In *NDSS*, 2016.

[3] Chen Chen, Daniele E. Asoni, David Barrera, George Danezis, and Adrian Perrig. HORNET: High-speed onion routing at the network layer. In *CCS*, 2015.

[4] Tae-Ho Lee, Pawel Szalachowski, David Barrera, Adrian Perrig, Heejo Lee, and David Watrin. Bootstrapping Real-world Deployment of Future Internet Architectures. *arXiv/1508.02240*, 2015.

[5] Stephanos Matsumoto, Raphael M. Reischuk, Pawel Szalachowski, Tiffany Hyun-Jin Kim, and Adrian Perrig. Designing a global authentication infrastructure. *ArXiv*, (arXiv:1506.03392), 2015.

[6] Simon Peter, Umar Javed, Qiao Zhang, Doug Woos, Thomas Anderson, and Arvind Krishnamurthy. One Tunnel is (Often) Enough. In *ACM SIGCOMM*, 2014.

[7] Costin Raiciu, Christoph Paasch, Sebastien Barre, Alan Ford, Michio Honda, Fabien Duchene, Olivier Bonaventure, and Mark Handley. How Hard Can It Be? Designing and Implementing a Deployable Multipath TCP. In *NSDI*, 2012.

[8] The SCION Team. The official SCION webpage. http://www.scion-architecture.net, 2016.

[9] Xin Zhang, Hsu-Chun Hsiao, Geoffrey Hasker, Haowen Chan, Adrian Perrig, and David G. Andersen. SCION: Scalability, Control, and Isolation on Next-Generation Networks. In *IEEE S & P (Oakland)*, 2011.