

# SCION Five Years Later: Revisiting Scalability, Control, and Isolation on Next-Generation Networks

David Barrera, Raphael M. Reischuk, Pawel Szalachowski, Adrian Perrig

Network Security Group  
ETH Zürich

## 1 Introduction

The Internet has been successful beyond even the most optimistic expectations. It permeates and intertwines with almost all aspects of our society and economy. The success of the Internet has created a dependency on communication as many of the processes underpinning the foundations of modern society would grind to a halt should communication become unavailable. However, much to our dismay, the current state of safety and availability of the Internet is far from being commensurate given its importance.

Although we cannot conclusively determine what the impact of a 1-hour, 1-day, or even 1-week outage of Internet connectivity on our society would be, anecdotal evidence indicates that even short outages have a profound negative impact on governmental, economic, and societal operations. To make matters worse, the Internet has not been primarily designed for high availability in the face of malicious actions by adversaries. Recent patches to improve Internet security and availability have been constrained by the current Internet architecture, business models, and legal aspects. Moreover, some of the fundamental design decisions of the current Internet inherently complicate secure operation.

To address these issues, we study the design of a next-generation Internet architecture that provides a fundamental building block: highly available point-to-point communication. In addition to availability, the architecture should offer security by design, it should provide incentives for deployment, and it should consider economic and political issues at the design stage.

As a solution to address these desired properties, we propose the inter-domain network architecture **SCION**, which is also an acronym for **Scalability, Control, and Isolation On Next-generation networks**. In this article, we present (a retrospective of) its goals and design decisions, its attacker model and limitations, and 5 years of research conducted since the initial publication [63].

## 2 Goals

In this section, we present high-level goals an inter-domain point-to-point communication architecture should satisfy; we illustrate why these goals are important and how they can be achieved. Finally, we briefly discuss non-goals, i.e., specific properties that we intentionally excluded from the design goals.

### 2.1 Availability in the Presence of Adversaries

Our overarching goal is the design of a point-to-point communication infrastructure that remains *highly available* even in the presence of distributed adversaries: as long as an attacker-free path between endpoints exists, that path should be discovered and used with guaranteed bandwidth between these endpoints.

Availability in the presence of adversaries is an exceedingly challenging property to achieve. An *on-path adversary* may drop, delay, or alter packets that it should forward, or inject packets into the network. The architecture hence needs to provide mechanisms to circumvent these malicious

elements. An *off-path adversary* could launch hijack attacks to attract traffic to flow through network elements under its control. Such traffic attraction can take various forms; for instance, an adversary could announce a desirable path to a destination (e.g., by using forged paths or attractive network metrics). Conversely, the adversary could render paths not traversing its network less desirable (e.g., by inducing congestion). An adversary controlling a large botnet could also perform Distributed Denial of Service (DDoS) attacks, congesting selected network links. Finally, an adversary could interfere with the discovery of legitimate paths (e.g, by flooding path discovery with bogus paths).

## 2.2 Transparency and Control

We aim to provide greater transparency and control for (1) forwarding paths of network packets, and (2) trust roots that are used for entity validation.

**Transparency and Control over Forwarding Paths.** When the network offers path transparency, endpoints know (and can verify) the forwarding path taken by network packets. Applications that transmit sensitive data can benefit from this property, as packets can be ensured to traverse certain Internet service providers (ISPs) and avoid others.

Taking transparency of network paths as a first property, we aim to additionally achieve path control, a stronger, more influential property, with which receivers can control incoming paths through which they are reachable. Given a path to a receiver, senders can control end-to-end paths. This seemingly benign requirement has various repercussions – beneficial but also fragile if implemented incorrectly. The beneficial aspects of path control for senders and receivers include:

- *Separation of network control plane and data plane.* To enable path control, the control plane (which determines networking paths) needs to be separated from the data plane (which forwards packets according to the determined paths). The separation ensures that forwarding cannot retroactively be influenced by control-plane operations, e.g., routing changes. Moreover, the separation contributes to enhanced availability as working forwarding paths cannot be disrupted by routing changes, but it also requires mechanisms to deal with link failures.
- *Enabling of multi-path communication.* Path control empowers multi-path communication by letting senders select multiple paths to carry packets towards their destinations. Multi-path communication is a powerful mechanism to enhance availability [4].
- *Defending against network attacks.* If the packet’s path is carried in its header (which is one way to achieve path control), then the destination can reverse the packet path to return its response to the sender, mitigating amplification attacks. Path control also enables circumvention of malicious network entities or congested network areas, providing a powerful mechanism against DoS and DDoS attacks.

The fragile aspects that need to be handled with particular care are the following.

- *Respecting ISPs’ forwarding policies.* If senders have complete path control, they may violate ISPs’ forwarding policies. We thus need to ensure that ISPs offer a set of policy-compliant paths amongst which senders can choose from.
- *Preventing malicious path creation.* A malicious sender could exploit path control for attacks, for example by forming malicious forwarding paths such as loops that consume increased network resources.
- *Scalability of path control.* Source routing does not scale to inter-domain networks, as a source would need to know the network topology to determine paths. To make path control scale, we ensure that sources select amongst a relatively small set of paths. We thus rely on source-selected paths instead of full-fledged source routing.
- *Permitting traffic engineering.* Fine-grained path control would inhibit ISPs from operating and performing traffic engineering. We thus seek to provide autonomous system (AS) level path control only at the level of ingress/egress interfaces, allowing ISPs to still control paths internally.

**Transparency and Control over Trust Roots.** Roots of trust are used for the verification of entities in today’s Internet. For example, verification of the server’s public key in a TLS certificate, or verification of a Domain Name Service (DNS) response in DNSSEC [6]. Transparency of trust roots provides the property that an end host or user can know the complete set of trust roots that it needs to rely upon for the validation of an entity certificate. Such enumeration of trust roots is complicated today because of intermediate Certification Authority (CAs) certificates that are not explicitly listed but implicitly trusted, e.g., in the TLS public key infrastructure (PKI). In fact, independent studies have counted over 300 roots of trust in the TLS PKI [17,1], but because of the lack of transparency there may be additional ones these studies have missed.

Providing control for trust root selection enables *trust agility* [40], allowing *users* to easily select or exclude the roots of trust they want to rely upon. The challenge then becomes the validation of each certificate, regardless of the choice of trust roots by users and network entities (e.g., web servers).

### 2.3 Efficiency, Scalability, and Extensibility

Despite the lack of availability and transparency, today’s Internet also suffers from a number of efficiency and scalability deficiencies: for instance, the Border Gateway Protocol (BGP), a global inter-domain routing protocol, encounters scaling issues in cases of network fluctuations, where routing protocol convergence can require minutes [52]. A 2006 earthquake in Taiwan that severed multiple undersea communication cables caused Internet outages throughout Asia for several days [9]. Moreover, routing tables have reached the limits of their scalability due to prefix de-aggregation (i.e., announcement of more specific prefixes) and multihoming [29]. Unfortunately, extending the memory size for routing tables is challenging as the underlying Ternary Content-Addressable Memory (TCAM) hardware is expensive and power-hungry, consuming on the order of a third of the total power consumption of a router. Extending the routing table memory would thus drastically increase cost and power consumption of routers.

Security and high availability come at a cost, usually resulting in lower efficiency and potentially diminished scalability. High performance and scalability, however, are required for viability in the current economic environment. We therefore explicitly seek *high efficiency* as a goal in the common case such that packet forwarding latency and throughput are at least as fast as current IP forwarding. Moreover, we seek *improved scalability* compared to the current Internet, in particular with respect to BGP and the size of routing tables.

One approach to achieve efficiency and scalability is to avoid router state wherever possible. We observe that modern block ciphers such as AES can be computed faster than performing memory lookups. For example, on current PC platforms, computing AES requires on the order of 50 cycles while fetching a byte from main memory requires around 200 cycles [2]. Moreover, modern block ciphers can be implemented in hardware with tens of thousands of gates, which is sufficiently small to replicate it profusely, which in turn enables high parallelism – the high complexity of a high-speed memory system prevents such replication at the same scale. We thus aim to place state into packet headers and protect that state cryptographically, enabling higher packet processing speeds and simpler router architectures compared to today’s IP routers. Besides higher efficiency, avoiding state on routers also prevents state-exhaustion attacks [54] and state inconsistencies.

Our goal of efficiency and scalability is in line with the *end-to-end principle*, which states that a function should be implemented at the network layer in which it can most effectively operate [53]. Since the end host has the most information about its internal state, functions such as bit error recovery, duplicate suppression, or delivery acknowledgments are best handled by the end host. Compared to the current Internet, SCION applies the end-to-end principle one layer lower in the protocol stack. Currently, most transport-layer functionality is handled by the end host according to this principle. However, in SCION, end hosts also assist with network-layer functionality such as path selection. End host path selection is communicated to the network by packet-carried forwarding information, which in turn removes the need for inter-domain routing tables at border routers. Consequently, end host path selection results in a simpler forwarding plane and thus more efficient routers.

To future-proof SCION, we design the core architecture and code base to be *extensible*, such that additional functionality can be easily built and deployed. SCION clients and routers should (without overhead or expensive protocol negotiations) discover the minimum common feature set supported by all intermediate nodes.

## 2.4 Support for Global but Heterogeneous Trust

Given the diverse nature of constituents in today's Internet with diverse legal jurisdictions and interests, an important challenge is how to scale authentication of entities (e.g., AS ownership for routing, name servers for DNS, or domains for TLS) to a global environment. The roots of trust of currently prevalent PKI models (monopoly and oligopoly) do not scale to a global environment because mutually distrusting entities cannot agree on a single trust root (monopoly model), and because the security of a plethora of roots of trust is only as strong as its weakest link (oligopoly model). We thus seek a trust architecture that supports *meaningful trust roots in a global environment* with mutually distrusting entities.

## 2.5 Deployability

Incentives for deployment are important to overcome the resistance for upgrading today's core Internet infrastructure. A multitude of features is necessary to offer the initial impulse: high availability even under control-plane and data-plane attacks (e.g., built-in DDoS defenses), path transparency and control, trust root transparency and control, high efficiency, robustness to configuration errors, fast recovery from failures, high forwarding efficiency, multi-path forwarding, etc.

If early adopters cannot obtain sufficient benefits from migrating to a new network architecture, even initial deployment is unlikely to be successful. Ideally, even the first deploying ISP can gain a competitive advantage, and start selling services to its initial customers.

Migration to the new architecture should require minimal added complexity to the existing infrastructure. Deployment should be possible by re-utilizing the internal switching infrastructure of an ISP, and only require installation or upgrade of a few border routers. Moreover, configuration of the new architecture should be similar to the existing architecture, such as in the configuration of BGP policies, minimizing the amount of additional personnel training.

Economic and business incentives are also of critical importance. ISPs should be able to define new business models and sell new services. Users should derive a business advantage from the new architecture, for example, obtain properties similar to a leased line at a fraction of the cost. Migration cost should be minimal, requiring only the deployment of low-cost routers. Finally, a new architecture should not disrupt current Internet business models, but maintain the current Internet topology and business relationships (e.g., support peering).

## 2.6 Non-Goals

We deliberately exclude certain properties and goals that could be deployed as additional functionality later on. For example, we do not consider multicast or efficient content dissemination as part of the basic communication infrastructure, as we recognize the significant complexity these features would add. Also, these features can be effectively added through an overlay leveraging a next-generation Internet architecture's basic communication infrastructure [19].

We additionally consider several other problems out of scope for a network architecture. A major category of current security problems are software vulnerabilities. While software vulnerabilities of end hosts are clearly out of scope, software vulnerabilities of network components such as routers can affect network operation. It is thus important to address these network vulnerabilities through robustness to malicious components and attempts to reduce them through a simple network architecture. Malicious Internet content (e.g., spam or phishing emails, malicious web pages) should not be directly addressed by the communication infrastructure. The architecture, however, should offer mechanisms that assist in defending against these threats.

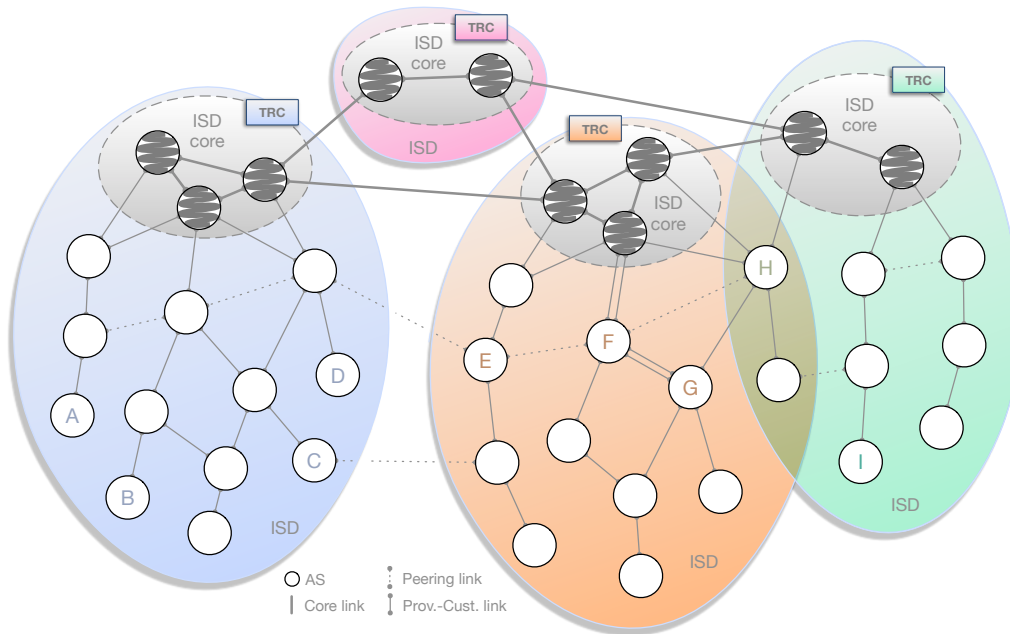


Figure 1: Four SCION ISDs with ISD Cores and ASes. The ISD Core ASes are connected via Core paths. Non-Core ASes are connected via customer-to-provider or peering links. AS H is contained in two ISDs.

### 3 SCION Architecture Overview

We now provide a high-level overview of the SCION architecture. A more detailed description is in papers available on our web site <http://www.scion-architecture.net>.

#### 3.1 SCION High-Level Overview

A fundamental building block to achieve the properties of high availability, transparency, scalability, and support for heterogeneous trust is **ISolation Domains (ISDs)**. An ISD constitutes a logical grouping of a set of **Autonomous Systems (ASes)**, as depicted in Figure 1. An ISD is administered by one or more ASes, which form the **ISD Core**. We refer to these ASes as **ISD Core ASes** or simply **Core ASes**. An ISD contains one or more regular ASes. The ISD is governed by a policy, called **Trust Root Configuration (TRC)**, which is negotiated by the ISD Core. The TRC defines the roots of trust that are used to validate bindings between named entities and their public keys (key certificates) or their addresses (DNS). As part of the TRC, every ISD has an associated human-understandable name space, which is globally-unique. The only global coordination that is required in SCION is hence the ISD name and number.

ASes join an ISD by purchasing service from an AS already in that ISD; joining an ISD thus constitutes an acceptance of the ISD’s TRC. Otherwise, they would select an ISP which is part of an ISD they desire to belong to. Typically, 3–10 current Tier-1 ISPs would constitute the ISD’s Core ASes, and their associated customers would participate in the ISD. We envision that ISDs will span areas with uniform legal environments that provide enforceable contracts. If two ISPs have a contract dispute they cannot resolve by themselves, such a legal environment can provide an external authority to resolve the dispute. All ASes within an ISD also agree on the TRC, i.e., the entities that operate the trust roots and set the ISD policies. ISDs will thus likely be formed along national boundaries or federations of nations, as entities within a legal jurisdiction can enforce contracts and agree on a TRC. ISDs are hierarchical, as SCION supports sub-ISDs. ISDs can also overlap in the sense that an AS may be part of several ISDs. Although an ISD provides isolation from other networks, the central purpose of an ISD is to provide transparency and to support

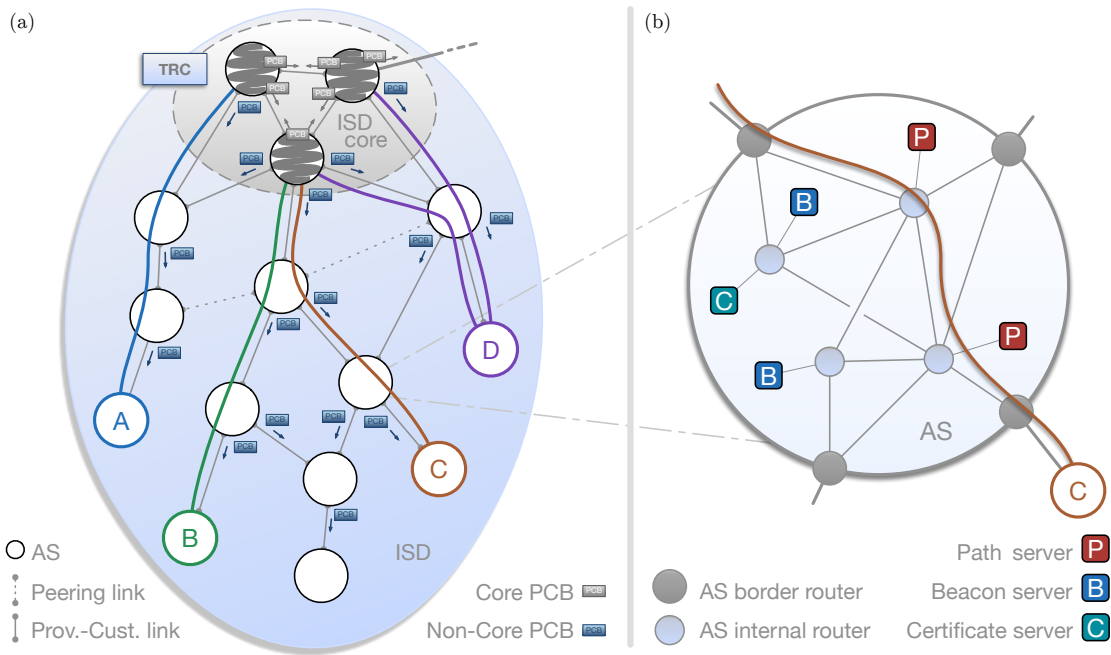


Figure 2: (a) SCION ISD with path construction beacons (PCBs) that are propagated from the ISD Core down to customer ASes, and path segments for ASes A, B, C, and D to the ISD Core. (b) Magnified view of an AS with its routers and servers. The path from AS C to the ISD Core traverses two internal routers.

heterogeneous trust environments. ISDs may seem to lead to “Balkanization” and prevent an open Internet, but they counter-intuitively provide openness and transparency, as we hope to elucidate in this article.

SCION uses two levels of routing, intra-ISD and inter-ISD. Both levels utilize **Path Construction Beacons (PCBs)** to discover and establish routing paths (see Figure 2a). An ISD Core AS announces a PCB and disseminates it as a policy-constrained multi-path flood either *within* an ISD (to discover intra-ISD paths) or *among* ISD Core ASes (to discover inter-ISD paths). PCBs accumulate cryptographically-protected AS-level path information as they traverse the network. These cryptographically protected contents (that we call opaque fields as described below) within received PCBs are chained together by sources to create a data transmission path segment that traverses a sequence of ASes. Packets thus contain AS-level path information avoiding the need for border routers to maintain inter-domain routing tables. We refer to this concept as **Packet-Carried Forwarding State (PCFS)**.

Through the inter-domain PCB transmission process, Core ASes learn paths to every other Core AS. Through the intra-domain PCB dissemination, ASes learn path segments on how to reach ISD Core ASes, which enable an AS to communicate with the ISD Core. Figure 2a shows some path segments from the ASes A, B, C, and D to the ISD Core.

We emphasize that PCFS in SCION is *different from* source routing, since a source node does not search a network topology graph to select its path. Instead, with the approach of *source-selected paths*, a source node combines at most three path segments (up-segment, core-segment, and down-segment). Since an arbitrary source up-segment combined with an arbitrary destination down-segment (along with an appropriate core-segment if necessary) results in a valid end-to-end path, a source node does not need to search any topology to find a path, thus, the approach is fundamentally different from source routing.

### 3.2 Control Plane: Beaconing for Route Discovery

We now discuss the control plane components and mechanisms in more detail. The control plane is responsible for discovering paths and making those paths available to end hosts. Figure 2b shows the main components that perform these operations in SCION: *beacon servers* discover path information; *path servers* disseminate such path information; and *certificate servers* manage and distribute cryptographic key material. Border routers provide the connectivity between ASes.

**Beacon servers** are responsible for the dissemination of PCBs (see Figure 2a). Beacon servers in a Core AS generate intra-ISD PCBs that are sent to all non-Core ASes of the ISD. Non-Core AS beacon servers receive such PCBs and re-send them to their customer ASes, which results in cycle-free PCB propagation and policy-compliant AS-level paths. Figure 3 shows PCBs that are propagated from the ISD Core down to customer ASes. At every AS, information about the interfaces of the AS is added to the PCB, whose overall size typically stays well below 1kB.

The beacon servers run a fault-tolerant protocol to ensure state consistency across all local servers. Periodically, a master beacon server generates a set of PCBs that it forwards to its customer ASes. In the case of inter-ISD communication, the beaconing process is similar to BGP’s route advertising process, although the process is periodic and PCBs are flooded multi-path over policy-compliant paths to discover multiple paths between any pair of ASes. SCION’s beacon servers can be configured to express all BGP policies, as well as additional properties (e.g., control of upstream ASes) that BGP cannot express.

**Path servers** store mappings from AS identifiers to sets of such announced path segments, and are organized as a hierarchical caching system similar to today’s DNS. ASes, through the master beacon servers, select the set of path segments through which they want to be reached, and upload them to a path server in the ISD Core.

**Certificate servers** store cached copies of TRCs retrieved from the ISD Core, store cached copies of other ASes’ certificates, and manage keys and certificates for securing intra-AS communication. Certificate servers are queried by beacon servers when validating the authenticity of PCBs (i.e., when a beacon server does not have a corresponding certificate).

An AS typically receives several PCBs representing several diverse path segments on how to reach various ISD Core ASes. Figure 2a shows two path segments for AS D. We call a path segment that leads *towards* an ISD Core an **up-segment**, and a path segment that leads *from* the ISD Core to an AS a **down-segment** – although path segments are typically bi-directional and thus support packet forwarding in both directions. More precisely, up-segments and down-segments are invertible: by flipping the order, an up-segment is converted to a down-segment and vice versa. Path servers learn up-segments by extracting them from PCBs they obtain from the local beacon servers. To reach its ISD Core, a host performs a path lookup at its local path server, fetching up-segments to the ISD Core. To reach a remote destination, a host additionally queries its path server for the down-segment of the destination AS. In case the local path server has no cached entry for the down-segment, it will query the destination AS’s Core path server.

How do the Core path servers know the down-segments of the destination? The beacon servers in an AS select the down-segments through which the AS desires to be reached, and register these path segments at the Core path servers. When links fail, segments expire, or better segments become available, the beacon servers keep updating the down-segments registered for their AS.

End-to-end communication is enabled by a combination of up to three path segments that form a SCION end-to-end path. More precisely, the source host in ISD  $\mathcal{I}$  sends a path resolution request to its local path server, who forwards the request to a core path server. In case the requested path’s destination is *within* the ISD  $\mathcal{I}$ , the core path server responds by immediately sending up to  $k$  down-segments to the local path server. In case the requested path’s destination is *outside* the ISD  $\mathcal{I}$ , then the core path server first requests the corresponding down-segments from the core path server in destination ISD  $\mathcal{J}$  before responding to the local path server. In both cases, the local path server returns up to  $k$  up- and down-segments to the requesting source, and if needed, a core-segment connecting the core of  $\mathcal{I}$  with the core of  $\mathcal{J}$ . Depending on the returned segments, SCION paths can be created as follows:

- Case 1 (**immediate path segment combination**, e.g., path B → D in Figure 2a): the last AS on the up-segment (ending at a Core AS) is the same AS as the first AS on the down-segment (starting at a Core AS). In this case, the simple combination of up- and down-segment creates a valid end-to-end path.
- Case 2 (**AS shortcut**, e.g., path B → C in Figure 2a): the up-segment and down-segment intersect at a non-Core AS. This is the case of a shortcut where up-segment and down-segment meet before entering the ISD Core. In this case, a shorter path is possible by removing the extraneous part of the path. The special case when the source’s up-segment contains the destination AS is treated in the same way, i.e., the intersection of both segments is omitted from the path.
- Case 3 (**peering shortcut**, e.g., path A → B in Figure 2a): a peering link exists between the two segments, so a shortcut via the peering link is possible. As in Case 2, the extraneous path segment is cut off. The peering link could be traversing to a different ISD. Respecting today’s business relationships between ISPs, at most one peering link can be traversed.
- Case 4 (**core-segment combination**, e.g., path A → D in Figure 2a, or A → I in Figure 1): the last AS on the up-segment is different from the first AS on the down-segment. This case requires an additional core-segment to connect the up- and down-segment. In case the communication remains within the same ISD (A → D), a local ISD core-segment is needed; otherwise (A → I), an inter-ISD core-segment is required.

Once an end-to-end path is chosen, this path is encoded in the SCION packet header, which makes inter-domain routing tables unnecessary for border routers: both the egress and the ingress interface of each AS on the path are encoded as PCFS in the packet header. The destination can respond to the source by inverting the end-to-end path from the packet header, or it can perform its own path lookup and construction.

SCION’s beaconing process has several important aspects. The periodicity is on the order of 10 seconds – in the current system a fresh set of beacons is sent over each inter-AS link to the neighboring ASes every 15 seconds. This beacon propagation process is thus asynchronous, i.e., PCBs are sent based on a local timer and are not propagated immediately upon arrival. The paths for propagation are selected based on a path quality metric with the goal of identifying consistent, diverse, efficient, and policy-compliant paths. *Consistency* refers to the requirement that there exists at least one property along which the path is uniform, such as an AS capability (e.g., anonymous forwarding) or link property (e.g., low latency). *Diversity* refers to the set of paths that are announced over time being as path-disjoint as possible to provide high quality multi-path options. *Efficiency* refers to the length, bandwidth, latency, utilization, and availability of a path, where more efficient paths are naturally preferred. *Policy compliance* refers to the requirement that the path adheres to the AS’s routing policy. Based on past PCBs that were sent, a beacon server scores the current set of candidate path segments and sends the  $k$  best segments as the next PCB. To provide some concreteness to this description, we currently use  $k = 5$ , and send PCBs every 15 seconds to each neighbor over each provider-to-customer link. SCION intra-ISD beaconing can scale to networks of arbitrary size, because each inter-AS link carries the same number of PCBs regardless of the number PCBs received by the AS.

Unlike in the current Internet, link failures are not automatically resolved by the network, but require more active involvement by end hosts. Since SCION forwarding paths are static, they break when one of the links fails. Link failures are handled by a three-pronged approach that typically masks link failures without any outage to the application and rapidly re-establishes fresh working paths. Specifically, (1) PCB dissemination occurs every few seconds, constantly establishing new working paths in case existing paths become unavailable. (2) ICMP-like control messages rapidly erase path segments with broken links from path servers and beacon servers – thereby triggering the dissemination of additional PCBs. Beacon servers then immediately send additional working paths after learning of a path failure. (3) Most importantly, SCION end hosts use multi-path communication by default, thus masking link failures to an application with another working path. As multi-path communication is very successful in achieving high availability (even in environments with very limited path choice [4]), SCION beacon servers actively attempt to create disjoint paths, SCION path servers make an effort to select and announce disjoint paths, and end hosts make an ef-



fort to compose path segments to achieve maximum resilience to path failure. Consequently, most path failures in SCION are imperceptible to the application, unlike the numerous short outages plaguing the current Internet [36,33].

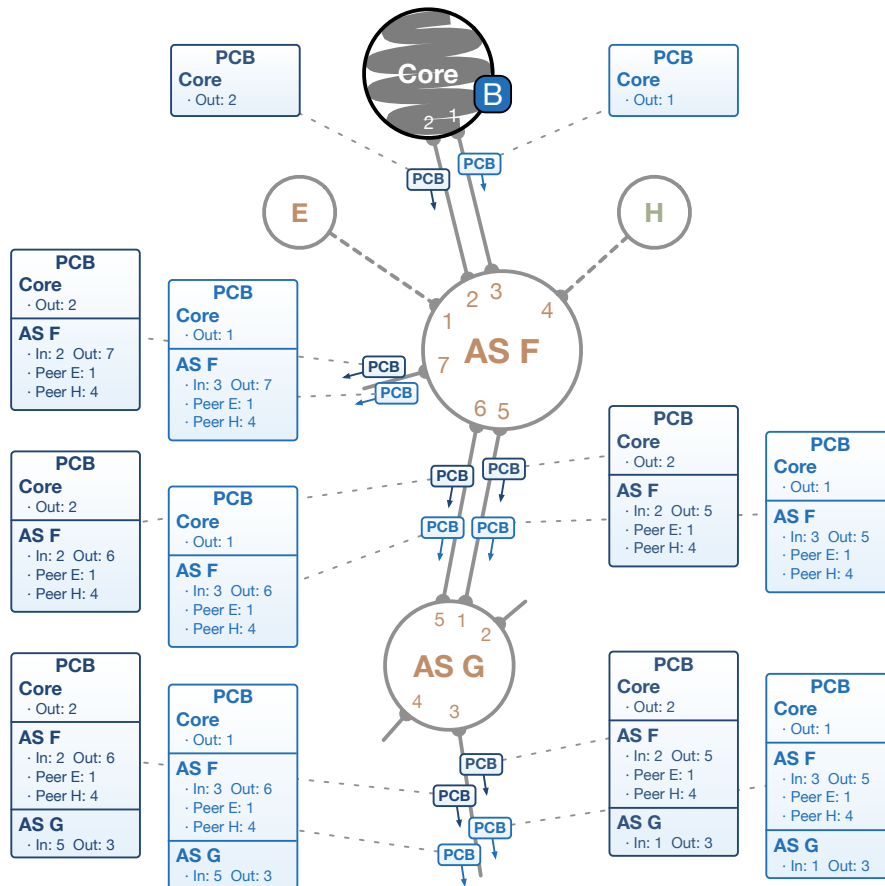


Figure 3: Intra-ISD PCB propagation from the ISD Core down to customer ASes. For the sake of illustration, the interfaces of each AS are numbered with consecutive integer values. In practice, each AS can choose any encoding for its interfaces. In particular, only the AS itself needs to understand its encoding.

Paths are represented at AS-level granularity, which by itself is insufficient for fine-grained path diversity; ASes often have several diverse connection points, and thus a disjoint path is possible despite the AS sequence being identical. For this reason, SCION encodes AS ingress and egress interfaces as part of the path, exposing a finer level of path diversity. Figure 3 demonstrates this feature: AS F receives two different beacons via two different links from the Core. Moreover AS F uses two different links to send two different beacons to AS G, each containing the respective egress interfaces. AS G extends the two beacons and forwards both of them over a single link to its customer.

An important optimization point is that SCION also supports peering links between ASes. Consistent with AS policies in the current Internet, PCBs do not traverse peering links. However, peering links are announced along with a regular path in a PCB. Figure 3 shows how AS F includes its two peering links in the PCB. If the same peering link is announced on two paths, then the peering link can be used for the end-to-end path. SCION also supports peering links that cross

ISD boundaries, which highlights the importance of SCION’s path transparency property; a source knows the exact set of ASes and ISDs traversed during the delivery of a packet.

Communication within an AS is handled by existing intra-domain communication protocols, such as IP, Multi-Protocol Label Switching (MPLS), or Software-Defined Networking (SDN) – border routers encapsulate the SCION packet inside an IP, MPLS, or SDN frame. Figure 2b shows an AS and a traversing intra-AS path from AS C to the ISD Core.

Inter-ISD beaconing operates similarly to intra-ISD beaconing, except that inter-ISD PCBs only traverse ISD Core ASes. The same path selection metrics apply, where an AS attempts to forward the set of most desirable paths to its neighbors. A difference, however, is that an AS forwards  $k$  beacons *per* source AS, with  $k = 3$ . The periodicity of beaconing is also reduced to once a minute or explicitly upon path changes. Similar to BGP, this process is inherently not scalable, however, as the number of ISDs and the corresponding number of Core ASes is small, this approach is viable.

*Security Aspects* For protection against malicious ASes and to provide a secure control plane, SCION is equipped with an arsenal of cryptographic mechanisms. We describe an overview in this article; the details are in companion papers [63,41].

The root of trust of an ISD is composed of root key certificates of trusted ISD Core ASes and Certification Authorities (CAs). The ISD’s TRC specifies which root key certificates are trusted and how many different signatures are required for each operation. For example, an AS certificate may require signatures from 2 different entities, an update to a new TRC may require signatures from 4 different entities.

The SCION control plane includes the *SCION Control Message Protocol* (SCMP). One challenge in the design of SCMP was how to enable efficient authentication of SCMP messages, as the naïve approach of adding a digital signature to SCMP messages could create a processing bottleneck at routers when many SCMP messages would be created in response to a link failure. We thus make use of an efficient symmetric key derivation mechanism called Dynamically Recreable Key (DRKey) [34]. In DRKey, each AS uses a local secret key known to SCION border routers to derive on-the-fly a per-AS secret key using an efficient Pseudo-Random Function (PRF). Hardware implementations of modern block ciphers enable faster computation than a memory lookup from DRAM, and therefore such dynamic key derivation can even result in a speedup over fetching the key from memory. For verification of SCMP messages, the destination AS can fetch the derived key through an additional request message from the originating AS, which is protected by a relatively slow asymmetric operation. However, local caching ensures that this key only needs to be fetched infrequently, about once per hour. As a consequence, SCION provides fully secured control messages with minimal overhead.

Similar to BGPSEC [39], each AS signs the PCB it forwards. This signature enables PCB validation by all entities. To ensure path correctness, the forwarding information within each Packet-Carried Forwarding State (PCFS) also needs to be cryptographically protected, however, signature verification would hamper efficient forwarding. Thus, each AS uses a secret symmetric key that is shared among beacon servers and border routers and is used to efficiently compute a Message Authentication Code (MAC) over the forwarding information. The per-AS information includes the ingress and egress interfaces, an expiration time, and the MAC computed over these fields, which is all encoded within an 8-byte field that we refer to as **Opaque Field (OF)**. We use the term opaque because the structure of the field is largely at the discretion of each AS and requires no coordination with any other AS – as long as the AS itself can extract how to forward the packet on to the next AS.

The specified ingress and egress interfaces uniquely identify the links to the previous and following ASes. If a router is connected via the same outgoing interface to 3 different neighboring ASes, 3 different egress interface identifiers would be assigned. The OF’s expiration time can be set on the granularity of seconds or hours, depending on the type of path. For the discussion of this overview, we only consider the common case where paths are long-lived and OFs have an expiration time on the order of 12 hours.

In terms of cryptographic mechanisms, we built in algorithm agility such that cryptographic methods can be easily updated and exchanged. The MAC validation of OFs is per-AS, so an AS can

independently (without interaction with any other entity) update its keys or cryptographic mechanisms. We support multiple signatures by an AS, thus, an AS can readily deploy a new signature algorithm and start adding those signatures to the beacons. The beacon selection metric will automatically start creating paths where each AS supports the new algorithm, enforcing consistency of the signature type. Validating inter-domain PCBs is accomplished by requiring connected ISDs to cross-sign their respective TRCs – consequently, any sequence of ISDs has a verifiable sequence of signatures, the details of which are described in our paper on the SAINT system [41].

### 3.3 Data Plane: Packet Carried Forwarding State (PCFS)

While the control plane is responsible for providing end-to-end paths, the data plane ensures packet forwarding using the provided paths. A SCION packet minimally contains a path; source and destination addresses are optional in case the packet’s context is unambiguous without addresses. Consequently, SCION border routers forward packets to the next AS based on the AS-level path in the packet header (which is augmented with ingress and egress interface identifiers for each AS), without inspecting the destination address and also without consulting an inter-domain routing table. Only the border router at the destination AS needs to inspect the destination address or packet purpose to forward it to the appropriate local host(s).

An interesting aspect of this forwarding is enabled by the split of locator (the path towards the destination AS) and identifier (the destination address) [18]: because in-network forwarding does not consider the local identifier, any source or destination address format is possible. Thus, a domain can select an arbitrary addressing format for its hosts, e.g., 4-byte IPv4, 6-byte medium access control, 16-byte IPv6, or 20-byte accountable IP (AIP [3]) address. A nice consequence is that an IPv4 host could directly communicate with an IPv6 destination.

Routers can efficiently forward SCION packets; in particular, the absence of inter-domain routing tables and the absence of complex maximum prefix matching performed by current routers enables construction of faster and more energy-efficient routers. During forwarding, a border router would first verify that the packet entered through the correct ingress interface. If the packet has not yet reached the destination AS, the egress interface defines the next hop. For illustration purposes, let us assume that an AS uses IPv4 switching to internally forward traffic, that the AS has fewer than 255 border interfaces (ingress- or egress-interfaces), and that each border interface is directly connected to the neighboring AS. The AS can select internal addresses such that all border interfaces follow a common addressing scheme, e.g., the IPv4 address is 10.1.1.X for  $0 < X < 255$ . The value X here uniquely identifies the preceding or following AS, and thus can serve as the ingress or egress interface identifier in the OF. Consequently, a border router can simply extract the egress interface identifier X from the OF, encapsulate the SCION packet into an IPv4 packet with the destination address of 10.1.1.X, and let the intra-domain routing and forwarding handle packet delivery to the egress interface.

### 3.4 Entity Validation Infrastructure

All entity authentication in SCION is based on traditional certificates, which bind identifiers to public keys and carry digital signatures that are verified by roots of trust, i.e., public keys that are axiomatically trusted.<sup>1</sup> The challenge is how to achieve trust agility to enable flexible selection of roots of trust, resilience to private key compromises, and efficient key revocation. We explore these issues in detail in our SAINT system [41], and provide a high-level overview in this article.

A central question is how to structure the trust roots. Today’s Internet follows two trust models: monopoly and oligopoly. In the *monopoly* model, a single root of trust is used for authentication. The DNSSEC PKI [6] or the Resource Public Key Infrastructure (RPKI) [51] used in BGPSEC are examples of the monopoly model as they both essentially rely on a single public key that serves as a root of trust to verify all subsequent entities. The monopoly model suffers from two main

---

<sup>1</sup> The reason we did not make use of self-certifying public keys [43,3] for long-term identities is because of their inherent inability for revocation and the complexities involved with key updates. For short-term identities, however, we do benefit from their features.

drawbacks: all parties must agree on a single root of trust, and the single root of trust represents a single point of failure because its misuse enables forging a certificate for an arbitrary entity. The *oligopoly* model does not fare much better – instead of a single root of trust, the oligopoly model relies on several roots of trust, all of which are equally and completely trusted. Instead of one single point of failure in the monopoly model, the oligopoly model thus exposes several points of failure. The prime example is the TLS PKI, featuring on the order of 1500 trusted entities including about 300 roots of trust [17,1]. Compromise of a single trusted entity enables forging a server certificate which allows man-in-the-middle attacks, as we have recently witnessed in several instances involving Comodo, DigiNotar, and Turktrust.

SCION’s ISDs provide solutions to these issues by allowing each ISD to define its own set of roots of trust, along with the policy governing their use. Such scoping of trust roots within an ISD greatly improves security, as compromise of a private key associated with a trust root cannot be used to forge a certificate outside the ISD.

An ISD’s trust roots and policy are encoded in the **TRC file**. The TRC file has a version number, a list of public keys that serve as roots of trust for various purposes, and policies governing how many signatures are required for different certificates, how many signatures are needed to update the TRC, etc. The TRC serves as a way to bootstrap all entity authentications. We now discuss two properties offered by the TRC: *trust agility* and *efficient revocation of trust roots*. **Trust agility** offers the selection of the sets of roots of trust to initiate validation of certificates. A user can thus select an ISD that she believes maintains an uncompromised set of trust roots. A challenge with trust agility is to maintain global verifiability of all entities, regardless of the user’s selection. SCION offers this property by requiring all ISDs with a link among them to cross-sign each other’s TRC files – as long as a network path exists, a validation path thus exists along that network path. **Efficient revocation of trust roots** is the second important property. In today’s Internet, trust roots are revoked manually, or through OS or browser updates, often requiring a week or longer until a large fraction of the Internet population has observed such revocations. In SCION, PCBs carry the version number of the current TRC, and the updated TRC is required to validate that PCB. An AS that realizes that it needs a newer TRC can request it from the AS from whom it has received the PCB. Following the distribution of PCBs, an entire ISD updates the TRC within tens of seconds.

SCION also introduces new mechanisms for the validation of network entities such as: ASes, path server and DNS responses, or web servers. We separate authentication into two different types based on their respective emphasis on either availability or security [41]. One type is **routing authentication** to authenticate PCBs, which has availability as the main requirement since control plane messages must be authenticated to provide communication paths. To achieve a higher level of security, additional servers would need to be contacted to provide resilience to private key compromises, but this in turn would hamper availability. In some cases, the requirement to communicate with additional servers introduces a circular dependency, because routing is required to communicate with these servers, but contacting these servers is needed to verify the routing message. Therefore, we ensure that all information required to authenticate routing messages flows in the same direction as the routing messages themselves, avoiding circular dependencies. In the case of SCION, AS certificates binding ASes to public keys flow from providers to customers, in the same direction as PCBs.

The second type is **service authentication**, which serves the purpose of authenticating services such as web servers or DNS replies. Since the control plane is operational when hosts communicate with servers, additional entities can contribute to ensure higher security to verify server authenticity, such as integrity log servers and validators in the Attack-Resilient PKI (ARPKI) system [8]. ARPKI is a highly secure PKI system based on log servers that keep a public log of all certificates to monitor CAs’ operations, and CAs and validators that monitor operation of log servers. By requiring multiple signatures on certificates, and by adding signatures on all operations we obtain the property that at least 3 malicious trusted entities within the same ISD are needed to perform a man-in-the-middle attack on a single domain. To further increase security, we designed PoliCert, a system to enable domains to specify their detailed security policy [60]. By storing the domain policies in an ARPKI log, policy consistency and integrity is ensured. In concert, ARPKI

and PoliCert achieve a high level of security for domains' certificates – all PKI attacks we have witnessed in the past decade would have been avoided in this framework. As a last line of defense, we propose efficient gossip protocols for verifying the consistency of log servers [14]. Clients randomly exchange short information about the logs to guarantee that any misbehavior will eventually be detected.

### 3.5 Incremental Deployment and Incentives

Support for incremental deployability plays a key role in the successful adoption of any network architecture. To this end, we have designed SCION to be deployable (by both ISPs and end users) without requiring substantial changes to the existing infrastructure.

**Incremental Deployment.** At a minimum, ISPs need only deploy a border router capable of encapsulating and decapsulating SCION traffic as it leaves or enters their network. SCION ASes must also deploy certificate, beacon, and path servers. These servers can run on commodity hardware and can optionally be replicated for increased availability, e.g., Figure 2b contains two path servers and two beacon servers. The current version of the SCION codebase uses IP as an underlying protocol, which allows existing intra-domain networking infrastructure and configuration.

In terms of creation and deployment of ISDs, we envision these to grow organically, with one ISD initially defined for each area with uniform legal environments. Tier-1 ISPs within those ISDs would become Core ASes. SCION facilitates the evolution of ISD and AS structure through efficient updates to the TRC file.

Deploying SCION to end user sites (e.g., homes or businesses) is designed to require little effort as well, initially requiring no changes to existing software, networking stacks, nor the replacement of end user network devices. For initial deployment, we achieve customer-friendly deployment through the design of a device (which we call DENA, or Device for ENhancing Availability) that can be installed at customer locations. DENA is a bump-in-the-wire middlebox that sits between the customer's WAN connection and their ISP. The device transparently monitors network flows and identifies remote flow endpoints where other DENAs are present. If a remote DENA is discovered for a given network flow, a SCION path is established as a fallback channel in case network communication through standard BGP paths is lost or becomes unreliable. DENA is able to switch between BGP and SCION paths transparently (providing pseudo-multi-path functionality to network applications), improving network availability for all devices behind the middlebox, and without requiring changes to networking stacks. Full details on the design, development and evaluation of DENA can be found in the paper [38].

**Deployment Incentives.** An important issue to consider is what incentives exist for various parties for deploying SCION. For end users, the benefits of using SCION are plentiful, ranging from higher throughput and lower latency communication (which translates to better quality phone calls, and higher resolution video streams), to fewer Internet outages. Users can also benefit from the SCION extensions, enabling, for example, low-latency anonymous communication.

By deploying SCION, ISPs can provide high-availability service to their customers, and simultaneously increase resilience to DDoS attacks for both themselves and their customers. These service offerings can enable new revenue streams for ISPs who have deployed SCION. There is a relatively low cost to transitioning existing BGP business models and policies to SCION, as these policies can be expressed and even extended. In SCION, such policies are less prone to attacks, but also to configuration errors since any error is constrained to a local domain.

Similarly to ISPs, businesses (possibly running their own ISDs) can benefit from highly available communication at lower cost. Deploying SCION provides better connectivity for customers, and higher resilience to DDoS. As additional benefits, path control reduces the possibility of industrial or government espionage, while transparency further provides deterrence for such practices. Finally, control over the businesses' PKI can prevent man-in-the-middle attacks.

Governments have shown interest in deploying SCION for a variety of reasons. The ability to avoid a global root of trust, and to select their own roots, allows governments to cooperate and

trust parties whom they deem fit. The open nature of the SCION codebase allows it to be deployed freely onto any supported (and possibly verifiable) hardware device. This would help governments in cases where particular hardware vendors cannot be trusted.

As of 2015, we have deployed a global SCION testbed which we are actively using to vet SCION’s functionality and security. As of July 2015, the testbed includes deployment nodes in 5 continents with 3 ISDs and 20 ASes. We are continually adding nodes at universities and corporate sites. Details and requirements for sponsoring a SCION node can be found on our website.

### 3.6 Extensions

SCION’s extensible architecture enables new systems that can take advantage of the novel properties and mechanisms provided. As compared to the current Internet, most of the benefits can be afforded through the use of PCFS, path transparency, and control. We briefly describe noteworthy protocols and systems that have been built as extensions to SCION.

**Path validation** – SCION, through its use of PCFS, paves way for the Origin and Path Trace (OPT) mechanism. OPT enables senders, receivers, and routers to cryptographically verify the path that the packets have traversed [34]. By leveraging the DRKey mechanism, routers can efficiently derive their key, verify the path, and update the path validation fields.

**Anonymity and privacy** – PCFS also provides advantages for privacy. With PCFS and path transparency, the source is able to select paths that appear more trustworthy (e.g., those that do not traverse certain ASes). In addition, the packet header can be further obfuscated such that ASes on the path cannot learn identifying details about the source or the destination, unless they are immediately connected to one of them. Proposals such as LAP [28] and HORNET [13] leverage SCION’s path selection infrastructure to offer high bandwidth and low latency anonymous communication.

**DDoS defense** – the hierarchical organization of ASes into a manageable number of ISDs enables neighbor-based contracts between pairs of ISDs, which in conjunction with path segments inside the ISDs allows for establishing efficient bandwidth guarantees between any two end hosts. Such bandwidth guarantees are leveraged by SIBRA [7] to prevent DDoS attacks at the architectural level: independent of the number of distributed bots, end hosts obtain protection against Internet-wide link-flooding attacks, one of the major threats in today’s Internet.

## 4 Case Studies

SCION improves many aspects of the current Internet. This section highlights some of the use cases that demonstrate unique properties offered by the new architecture.

**Constraining traffic flows through trusted ASes** – Path control and transparency are important properties for sensitive traffic, where a sender wants control and assurance over which ASes will be traversed, due to legal, secrecy, or safety considerations. For instance, banking or medical data, which is typically bound to strict data privacy regulations, can be constrained in SCION to traverse only selected authorized ASes. Furthermore, the OPT mechanism enables a sender and receiver to verify the exact path taken on a per-packet basis, with negligible overhead [34].

**Highly available communication for critical infrastructures** – Critical infrastructures such as financial networks and industrial control systems used for power distribution require a high degree of availability. Internet outages have been known to wreak havoc on day-to-day operations, for example preventing ATM withdrawals or payment terminal operations [10]. SCION’s control-plane isolation through ISDs, its stable data plane, and its multi-path operation all contribute to dramatically higher availability.

**High-speed web browsing** – Current congestion control hinders high-speed communication because the sender and receiver require time to determine their sending rate and to constantly perform congestion control. Consequently, the sending rate is usually below the maximum possible rate. In SCION, through the SIBRA [7] extension, the sender performs a resource reservation with its initial packet, and the receiver will likely obtain a reservation with a high sending rate that it can immediately start to use on the reverse path. On such a reservation, no congestion control is

needed; consequently, the web server can immediately start sending the web page at a high rate to the browser.

**Mobility support** – With the proliferation of mobile devices, supporting reliable communication can be challenging since these devices frequently connect and disconnect from (sometimes several) networks. SCION supports high availability through multi-path communication and provides a header extension to inform the other party of new down segments as it connects to a new network. Failing paths are discarded and new paths are dynamically discovered transparently to users and applications. One challenging case, however, is when both sender and receiver simultaneously move to a new network and all the previously established paths fail at the same time. In this infrequent case, a name resolution server needs to be contacted to fetch fresh down segments for the other party [55].

## 5 Attacks and Defenses

SCION dramatically improves network security as compared to the current Internet, which we illustrate based on three important classes of attacks and their defenses.

**Prefix hijacking** – Numerous Internet outages are due to the malicious or erroneous announcement of IP address space, which is also known as prefix hijacking. Perhaps the most famous case is the hijack of YouTube by Pakistan for internal censoring, resulting in a global outage of YouTube. In fact, hijacks that impact only a small portion of the Internet happen on a daily basis. SCION prevents such hijacking through several mechanisms. With ISDs, misconfigurations and attacks in one ISD do not affect others; digitally signed route announcements mean unauthorized injection of routes is not possible; and digitally signed path distribution allows verification of paths by the sender.

**Forged TLS certificates** – Compromised roots of trust have been used to create rogue TLS certificates. The government of Iran infamously used compromised roots of trust to create rogue TLS certificates for Google and Yahoo services to perform man-in-the-middle attacks on its citizens; Iran is suspected to have mounted the attack on the DigiNotar CA, who signed these certificates. The ISDs and the ARPKI system used in SCION prevent such attacks, as a CA’s authority is scoped to the ISDs where the CA is active in, and using ARPKI at least 3 trusted entities all need to be compromised to perform a successful man-in-the-middle attack. Moreover, the SCION root of trust update mechanism enables revocation of roots of trust within tens of seconds, enabling quick recovery from compromise.

**DDoS attacks** – Large-scale DDoS attacks have been widely used to prevent access to domains. For example, a large-scale attack against Estonia made several of their critical infrastructures inaccessible during one week in April 2007 [27]. SCION would have minimized the impact of these attacks. ISDs allow external traffic to be de-prioritized, thus enabling internal communication in case the attack originates outside the ISD. Critical infrastructures can keep some network paths to a destination secret, thus preventing an adversary from even sending traffic to that destination because the cryptographic OFs are necessary to use a path. The SIBRA extension offers powerful mechanisms for DDoS defense, as it guarantees minimal traffic rates between any pair of ASes, which cannot be lowered even by a large-scale botnet [7], even when using new types of DDoS attacks such as Crossfire and Coremelt [58,32].

## 6 Deployment Caveats

The allocation and structure of ISDs presents a challenge for the deployment of SCION. It remains unclear, for example, which ASes within an ISDs will or should become Core ASes. We envision that among a group of ASes who deploy a top-level ISD, the AS or ASes that can form peering agreements with core ASes in other ISDs should become core ASes in their own ISD. However, SCION itself does not require or impose strict rules regarding the allocation of ISDs; ISDs can overlap, which means an AS can belong to several ISDs (cf. AS H in Figure 1). Sub-ISDs are possible as well, offering the flexibility to start an ISD without needing to peer with core ASes

of other ISDs and enabling finer-grained control over routing isolation and authentication. In this context, the important properties SCION offers are path control and transparency: as long as communicating hosts can select and inspect the path of their packets, the question of ISD partition is of secondary nature.

Another challenge that could arise is that each AS will attempt to be its own ISD or will want to be part of the ISD Core. While too many top-level ISDs will pose a problem for SCION scalability, we observe that economically sound decisions will lead to larger ISDs due to economies of scale – because the startup costs and operation of an ISD Core AS are more expensive than a non-Core AS, the operation of a large ISD will amortize the cost over more non-Core ASes. Moreover, ASes preferentially associate with larger ISDs, which can offer better connectivity to other ISDs as well as to other ASes within the ISD. On the other hand, ISD growth is limited to only as large as entities can agree on the ISD’s TRC (i.e., roots of trust). Finally, ASes desiring to be part of the ISD Core are assessed in the same way current ASes assess peering: an AS is permitted into the Core if the current Core ASes deem it to be large enough to fulfill Core AS duties (e.g., participating in beacon and path server replication as illustrated in Figure 2b).

As expected in architectures with PCFS, packet headers are necessarily larger. Larger headers place a bound on goodput, since payload space is traded off for header space. The current SCION codebase implements the OF as an 8-byte field. Since every AS on an end-to-end path has to be represented through a corresponding OF, the overhead increases linearly with the number of ASes on the path. However, given that the average AS path in today’s Internet is four hops long (and decreasing) [16,37], the overhead introduced by SCION should not exceed 40-50 bytes on average. The performance penalty of transmitting more packets appears reasonable since per-packet forwarding performance is faster than routing-table-based architectures. While the default header size has not shown a performance disadvantage in our testing, many of the proposed SCION extensions add length to the header.

Certain extensions (e.g., SIBRA, HORNET) have been designed for a use case assuming pervasive deployment (i.e., deployment at a majority of SCION ASes). While the benefits of these extensions are clear, we must consider the efficiency implications of certain ASes not deploying the extensions. For example, data payload space may be lost due to additional signatures or key material for path validation on nodes that do not validate paths, leading to inefficiency in data transfer. We have designed our extensions to be compatible with non-deploying nodes, but future work should consider improvements such as opportunistic enabling of these extensions.

Due to path dissemination and registration dynamics, SCION beacon and path servers can incur a high overhead under specific circumstances. For example, if a given link’s state fluctuates frequently between available and unavailable (due to error, hardware fault, or an adversary), the beacon server would need to consistently update the set of paths that include that link, and serve new paths excluding that link. We expect that this case will be rare, but also easily detectable. Additionally, higher quality (uptime, availability) links will have higher probability of selection, minimizing the impact of rapid path fluctuations.

We have shown that the basic building blocks of SCION are relatively straightforward to understand and have many beneficial properties and applications. However, as more extensions and alternative PKIs are added to the architecture, the operational complexity of the architecture increases correspondingly. We believe that this additional complexity is worth the security, efficiency, and availability guarantees provided by the extensions. It is ultimately up to the networking and research community to decide which of these extensions will be deemed worthwhile for widescale deployment.

## 7 Related Work

Several efforts on redesigning the Internet have been made over the past two decades to satisfy the new requirements of emerging Internet-based applications. Such requirements include naming, routing, mobility, network efficiency, availability, and evolvability of the Internet. We discuss several projects in this space based on a loosely temporal order clustered by topics.



The idea of clustering the network into domains has been attempted since the early days of the Internet. The Nimrod routing architecture [11], to our knowledge, is the first published description of these concepts. Nimrod describes a hierarchy of clusters of hosts, routers, or networks. A secure version of Nimrod was later proposed [56]. FARA [15] proposes a general notion of an entity to include clusters of computers that can be reached as a communication endpoint.

The NewArch project [15] describes comprehensive requirements for a new Internet, such as separation of identity from location, late binding using association, identity authenticity, and evolvability. However, it mostly emphasizes a new direction for end-point entities while the packet delivery in the current IP network is left intact. NewArch uses the New Internet Routing Architecture (NIRA) [62] for inter-domain routing, which aims to introduce competition among ISPs in the core by providing route control to the end users who can choose domain-level paths.

NDN [30,44] decouples location from identity and uses identity for locating the corresponding content. NDN relies heavily on in-network caching of data and is useful for accessing popular static content. However, NDN's scalability would suffer in the face of an increasing amount of ephemeral content (e.g., voice or video calls). NDN would also require more complex and energy-consuming routers than IP. The Publish-Subscribe Internet Routing Paradigm (PSIRP) project supports information-centric networking based on a publish-subscribe approach [48]. They propose an elegant approach to reduce the state on routers by having packets carry Bloom filters to encode the next hops of a multicast packet [31]. The CCNx project provides a specific implementation of content-centric networking, developing detailed specifications and prototype systems [12].

Mobility-first [49] is an architecture that quickly maps billions of identities to their locations, yet does not propose a fundamental change in the underlying forwarding architecture in terms of security and availability. Nebula [5] addresses security problems in the current Internet. Nebula takes a so-called default-off approach to reach a specific service, where a sender can send packets only if an approved path to a service is available. The network architecture helps the service to verify whether the packet followed the approved path (i.e., supporting path verification). However, Nebula achieves this property at a high cost. All routers on the path need to perform computationally-expensive path verification for every single packet and need to keep per-flow state, limiting its usage to highly specialized services.

Serval [45] proposes name-based service discovery and routing, and introduces a new service-access layer that enables late binding of a service to its location. Late binding provides flexibility in migrating and distributing services, yet it attempts to optimize networking for special application-services (especially in data-center network) built on top of the current Internet.

XIA [26] proposes an evolvable network architecture that can easily adapt to the evolution of networks by supporting various principal types (where the principal includes but is not limited to service, content, host, domains, and path). XIA leverages SCION for secure and available data forwarding.

The Framework for Internet Innovation (FII) [35] proposes a new architecture to enable evolution, diversity, and continuous innovation, such that the Internet can be composed of a heterogeneous conglomerate of architectures. The ChoiceNet [61] architecture proposes an "economy plane" to enable network providers to offer new network-based services to customers, providing an network environment for improving innovation and competition.

Several architecture proposals suggest the approach of better path control for senders and receivers, for example Segment Routing [20], Pathlets [25], NIRA [62], i3 [57], or SNAPP [47].

Forward [23] and SysSec [59] are proposing to build secure and trusted Information and Communication Technology (ICT) systems by engaging academia and industry. Forward is an initiative by the European Commission to promote the collaboration and partnership between industry and academia in their common goal of protecting ICT infrastructures. The Forward project categorizes security threats to various ICT systems including individual devices, social networks, critical infrastructures (such as smart electric grids), and the Internet infrastructure, and it aims at coordinating multiple research efforts to build secure and trusted ICT systems and infrastructures. SysSec aims to consolidate the systems security research community in Europe, promoting cybersecurity education, engaging a think-tank in discovering the threats and vulnerabilities of the current and future Internet, creating an active research road map in the area, and developing a

joint working plan to conduct state-of-the-art collaborative research. Since Forward and SysSec currently focus on identifying and handling threats, we believe our proposed tasks to be a good addition to the projects by providing an architecture that would significantly reduce the attack surface. RINA [50] is a recursive inter-network architecture that provides unified APIs across all protocol layers. In RINA, all layers have the same functions with different scope and range, where a layer is a distributed application that performs and manages inter-process communication. We would make an effort to design our prototype to fit into this paradigm so that our architecture can support seamless integration with other higher-layer security protocols/mechanisms.

Many researchers are currently studying Software-Defined Networking (SDN), for example in the OpenFlow [42,46] project. These efforts mainly consider intra-domain communication, which SCION can leverage to communicate within a domain.

Several future Internet efforts provide testbeds for running and testing a new architecture, such as GENI [24], Fi-ware [22] and FIRE [21].

We have developed SCION with a focus on security and high availability for point-to-point communication, which is a unique perspective and can contribute to other future Internet efforts. For instance, even content-centric networking needs a routing mechanism to reach the data source. SCION can offer the routing protocol to support that functionality. Once a server is found in a service-based infrastructure or a nearby content cache is found in a content-centric architecture, point-to-point communication between the end host and the server will offer the highest communication efficiency, as pure forwarding is faster than server-based or content-based lookups. Similarly, SCION can provide the point-to-point communication fabric in a mobility-centric architecture. Consequently, SCION offers mechanisms that complement many previously proposed future Internet architectures.

## 8 Conclusions

We have presented SCION, a future Internet architecture that provides security, availability, transparency, control, and scalability. SCION offers numerous advantages over the current Internet and supports other future Internet proposals as an underlying building block for highly reliable point-to-point communication.

Despite its research maturity after 5 years of work, SCION is still in its infancy in terms of deployment. While requiring relatively small changes by ISPs and domains, broadening adoption is currently SCION's foremost goal. We expect that the potential benefits for various stakeholders will provide strong incentives to drive adoption, leading to islands of SCION deployment. In the long term, connections and mergers among islands will enable ever-increasing numbers of native SCION end-to-end connections.

Working on SCION has offered us the opportunity to consider Internet architectures from a clean-slate perspective. The absence of limiting constraints (imposed by the current Internet environment) has been particularly rewarding, as the deep exploration of a problem space enabled us to design a system with properties that were previously thought to be impossible. We anticipate that the insight into the possible applications of a secure, dynamic, and highly-available network will help engage the network community to leverage SCION for their applications, and contribute to the project.

## Acknowledgements

We thank the original SCION authors as well as current and past members of the ETH Zürich Network Security group and of the CMU Security group for their contributions to the project. We also thank Yih-Chun Hu and Brian Trammell for interesting discussions, feedback, and suggestions that improved this article.

## References

1. M. Abadi, A. Birrell, I. Mironov, T. Wobber, and Y. Xie. Global authentication in an untrustworthy world. In *Proceedings of Workshop on Hot Topics in Operating Systems (HotOS)*, May 2013.
2. K. Akdemir, M. Dixon, W. Feghali, P. Fay, V. Gopal, J. Guilford, E. Ozturk, G. Wolrich, and R. Zohar. Breakthrough aes performance with intel® aes new instructions. *White paper, June*, 2010.
3. D. G. Andersen, H. Balakrishnan, N. Feamster, T. Koponen, D. Moon, and S. Shenker. Accountable internet protocol (AIP). In *Proceedings of ACM SIGCOMM*, 2008.
4. D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris. Resilient overlay networks. In *Proceedings of ACM Symposium on Operating Systems Principles (SOSP)*, Oct. 2001.
5. T. Anderson, K. Birman, R. Broberg, M. Caesar, D. Comer, C. Cotton, M. J. Freedman, A. Haeberlen, Z. G. Ives, A. Krishnamurthy, W. Lehr, B. Loo, D. Mazieres, A. Nicolosi, J. M. Smith, I. Stoica, R. Renesse, M. Walfish, H. Weatherspoon, and C. S. Yoo. The nebula future internet architecture. In *The Future Internet*, Lecture Notes in Computer Science. 2013.
6. R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose. DNS Security Introduction and Requirements. RFC 4033 (Proposed Standard), Mar. 2005. Updated by RFCs 6014, 6840.
7. C. Basescu, R. M. Reischuk, P. Szalachowski, A. Perrig, Y. Zhang, H.-C. Hsiao, A. Kubota, and J. Urakawa. SIBRA: Scalable internet bandwidth reservation architecture. In *NDSS'16: Network and Distributed System Security Symposium (to appear)*, Feb. 2016.
8. D. Basin, C. Cremers, T. H.-J. Kim, A. Perrig, R. Sasse, and P. Szalachowski. ARPKI: Attack resilient public-key infrastructure. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, Nov. 2014.
9. BBC. Asia communications hit by quake. <http://news.bbc.co.uk/2/hi/asia-pacific/6211451.stm>, Dec. 2006.
10. A. T. BGPMON.net. Massive route leak causes internet slowdown. <http://www.bgpmon.net/massive-route-leak-cause-internet-slowdown/>, June 2015.
11. I. Castineyra, N. Chiappa, and M. Steenstrup. The Nimrod routing architecture. RFC 1992, Aug. 1996.
12. CCNx - Content Centric Networking. <http://www.ccnx.org>, July 2015.
13. C. Chen, D. Asoni, D. Barrera, G. Danezis, and A. Perrig. HORNET: High-speed onion routing at the network layer. *ArXiv*, (arXiv:1507.05724), July 2015.
14. L. Chuat, P. Szalachowski, A. Perrig, B. Laurie, and E. Messeri. Efficient gossip protocols for verifying the consistency of certificate logs. In *Communications and Network Security (CNS)*. IEEE, 2015. (to appear).
15. D. Clark, R. Braden, A. Falk, and V. Pingali. Fara: reorganizing the addressing architecture. In *Proceedings of the ACM SIGCOMM workshop on Future directions in network architecture*, FDNA '03, pages 313–321, New York, NY, USA, 2003. ACM.
16. A. Dhamdhere and C. Dovrolis. Twelve Years in the Evolution of the Internet Ecosystem. *IEEE/ACM Transactions on Networking*, 19(5):1420–1433, Sep 2011.
17. Electronic Frontier Foundation. SSL Observatory. <https://www.eff.org/observatory>, 2010.
18. D. Farinacci, V. Fuller, D. Meyer, and D. Lewis. The locator/id separation protocol (LISP). RFC 6830, Jan. 2013.
19. S. K. Fayazbakhsh, Y. Lin, A. Tootoonchian, A. Ghodsi, T. Koponen, B. Maggs, K. Ng, V. Sekar, and S. Shenker. Less pain, most of the gain: incrementally deployable ICN. In *Proceedings of ACM Sigcomm*, Aug. 2013.
20. C. Filsfils, S. Previdi, B. Decraene, S. Litkowski, and R. Shakir. Segment Routing Architecture (IETF draft). 2015. Retrieved on 2015.05.13.
21. FIREworks. <http://www.ict-fireworks.eu/>, 2008.
22. FIWARE. <http://www.fi-ware.eu/>, 2015.
23. ICT - FORWARD. <http://www.ict-forward.eu/>, 2010.
24. GENI - Exploring Networks of the Future. <http://www.geni.net/>, 2015.
25. P. B. Godfrey, I. Ganichev, S. Shenker, and I. Stoica. Pathlet routing. In *Proceedings of ACM SIGCOMM Conference on Data Communication*, 2009.
26. D. Han, A. Anand, F. Dogar, B. Li, H. Lim, M. Machado, A. Mukundan, W. Wu, A. Akella, D. G. Andersen, J. W. Byers, S. Seshan, and P. Steenkiste. XIA: Efficient support for evolvable internetworking. In *Proc. 9th USENIX NSDI*, San Jose, CA, Apr. 2012.
27. S. Herzog. Revisiting the estonian cyber attacks: Digital threats and multinational responses. *Journal of Strategic Security*, 4(2):49–60, 2011.
28. H.-C. Hsiao, T. H.-J. Kim, A. Perrig, A. Yamada, S. C. Nelson, M. Gruteser, and W. Meng. LAP: Lightweight anonymity and privacy. In *Proceedings of IEEE Symposium on Security and Privacy*, pages 506–520. IEEE, May 2012.

29. G. Huston. Bgp in 2014. <http://www.potaroo.net/ispcol/2015-01/bgp2014.html>, Jan. 2015.
30. V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard. Networking named content. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, CoNEXT '09, pages 1–12, New York, NY, USA, 2009. ACM.
31. P. Jokela, A. Zahemszky, C. Esteve Rothenberg, S. Arianfar, and P. Nikander. LIPSIN: Line speed publish/subscribe inter-networking. In *Proceedings of ACM SIGCOMM Conference on Data Communication*, pages 195–206, 2009.
32. M. S. Kang, S. B. Lee, and V. D. Gligor. The crossfire attack. In *Proceedings of IEEE Symposium on Security and Privacy (S&P)*, May 2013.
33. E. Katz-Bassett, C. Scott, D. Choffnes, I. Cunha, V. Valancius, N. Feamster, H. Madhyastha, T. Anderson, and A. Krishnamurthy. LIFEGUARD: Practical repair of persistent route failures. In *Proceedings of ACM SIGCOMM*, Aug. 2012.
34. T. H. Kim, C. Basescu, L. Jia, S. B. Lee, Y. Hu, and A. Perrig. Lightweight source authentication and path validation. In *Proceedings of ACM SIGCOMM*, Aug. 2014.
35. T. Koponen, S. Shenker, H. Balakrishnan, N. Feamster, I. Ganichev, A. Ghodsi, P. B. Godfrey, N. McKeown, G. Parulkar, B. Raghavan, J. Rexford, S. Arianfar, and D. Kuptsov. Architecting for innovation. *SIGCOMM Computer Communication Review*, July 2011.
36. N. Kushman, S. Kandula, and D. Katabi. Can You Hear Me Now?! It Must be BGP. In *Computer Communication Review*, Mar. 2007.
37. M. Khne and V. Asturiano. Update on AS Path Lengths Over Time. <https://labs.ripe.net/Members/mirjam/update-on-as-path-lengths-over-time>, 2012. Retrieved on 2015.07.12.
38. T.-H. Lee, P. Szalachowski, D. Barrera, and Perrig. Bootstrapping Real-world Deployment of Future Internet Architectures. *arXiv:1508.02240*, Aug. 2015.
39. M. Lepinski and S. Turner. An overview of BGPSEC. IETF Draft, <http://tools.ietf.org/html/draft-ietf-sidr-bgpsec-overview-02>, May 2012.
40. M. Marlinspike. SSL and the future of authenticity. <http://blog.thoughtcrime.org/ssl-and-the-future-of-authenticity>, Apr 2011.
41. S. Matsumoto, R. M. Reischuk, P. Szalachowski, T. H.-J. Kim, and A. Perrig. Designing a global authentication infrastructure. *ArXiv*, (arXiv:1506.03392), July 2015.
42. N. McKeown, T. Anderson, H. Balakrishnan, G. Parulkar, L. Peterson, J. Rexford, S. Shenker, and J. Turner. Openflow: enabling innovation in campus networks. *SIGCOMM Comput. Commun. Rev.*, 38(2):69–74, Mar. 2008.
43. R. Moskowitz, P. Nikander, E. P. Jokela, and T. Henderson. Host Identity Protocol (HIP). Technical report, RFC 5201 Internet Engineering Task Force, 2008.
44. NDN. Named Data Networkin (NDN) - A Future Internet Architecture. <http://www.named-data.net/>, June 2015.
45. E. Nordstrom, D. Shue, P. Gopalan, R. Kiefer, M. Arye, S. Ko, J. Rexford, and M. J. Freedman. Serval: An end-host stack for service-centric networking. In *Proceedings of Symposium on Networked Systems Design and Implementation (NSDI)*, 2012.
46. OpenFlow - Enabling Innovation in Your Network. <http://www.openflow.org/>, 2011.
47. B. Parno, A. Perrig, and D. Andersen. SNAPP: Stateless network-authenticated path pinning. In *Proceedings of the ACM Symposium on Information, Computer and Communications Security (ASIACCS)*, Mar. 2008.
48. Publish-Subscribe Internet Routing Paradigm (PSIRP) project. <http://www.psirp.org/>.
49. D. Raychaudhuri, K. Nagaraja, and A. Venkataramani. MobilityFirst: A robust and trustworthy mobility-centric architecture for the future internet. *ACM SIGMOBILE Mobile Computing and Communications Review*, July 2012.
50. RINA - The Pouzin Society. <http://rina.tssg.org/>, 2014.
51. Resource Public Key Infrastructure (RPKI). <https://www.arin.net/resources/rpki/>, 2015.
52. A. Sahoo, K. Kant, and P. Mohapatra. BGP convergence delay under large-scale failures: Characterization and solutions. *Computer Communications*, 32(7), May 2009.
53. J. H. Saltzer, D. P. Reed, and D. D. Clark. End-to-end arguments in system design. *ACM Transactions on Computer Systems*, 2(4), Nov. 1984.
54. M. Schuchard, E. Y. Vasserman, A. Mohaisen, D. F. Kune, N. Hopper, and Y. Kim. Losing control of the Internet: Using the data plane to attack the control plane. In *Proceedings of Network and Distributed System Security Symposium (NDSS)*, Feb. 2011.
55. A. Sharma, X. Tie, H. Uppal, A. Venkataramani, D. Westbrook, and A. Yadav. A global name service for a highly mobile internetwork. In *Proceedings of SIGCOMM*, Aug. 2014.
56. K. E. Sirois and S. T. Kent. Securing the nimrod routing architecture. In *Symposium on Network and Distributed Systems Security (NDSS)*, San Diego, California, Feb. 1997.

57. I. Stoica, D. Adkins, S. Zhuang, S. Shenker, and S. Surana. Internet indirection infrastructure. *IEEE/ACM Transactions on Networking*, Apr. 2004.
58. A. Studer and A. Perrig. The coremelt attack. In *Proceedings of the 14th European Symposium on Research in Computer Security (ESORICS)*, Sept. 2009.
59. SysSec. <http://www.syssec-project.eu/>, 2015.
60. P. Szalachowski, S. Matsumoto, and A. Perrig. PoliCert: Secure and flexible TLS certificate management. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, Nov. 2014.
61. T. Wolf, J. Griffioen, K. L. Calvert, R. Dutta, G. N. Rouskas, I. Baldine, and A. Nagurney. ChoiceNet: toward an economy plane for the internet. *ACM SIGCOMM Computer Communication Review*, 44(3):58–65, July 2014.
62. X. Yang, D. Clark, and A. W. Berger. NIRA: A new inter-domain routing architecture. *IEEE/ACM Transactions on Networking*, 2007.
63. X. Zhang, H.-C. Hsiao, G. Hasker, H. Chan, A. Perrig, and D. G. Andersen. SCION: Scalability, control, and isolation on next-generation networks. In *Proceedings of IEEE Symposium on Security and Privacy*, May 2011.