# SCION over XIA

Vikram Rajkumar, Yue-Hsun Lin, XIA & SCION Teams

2014-05-08

# Outline

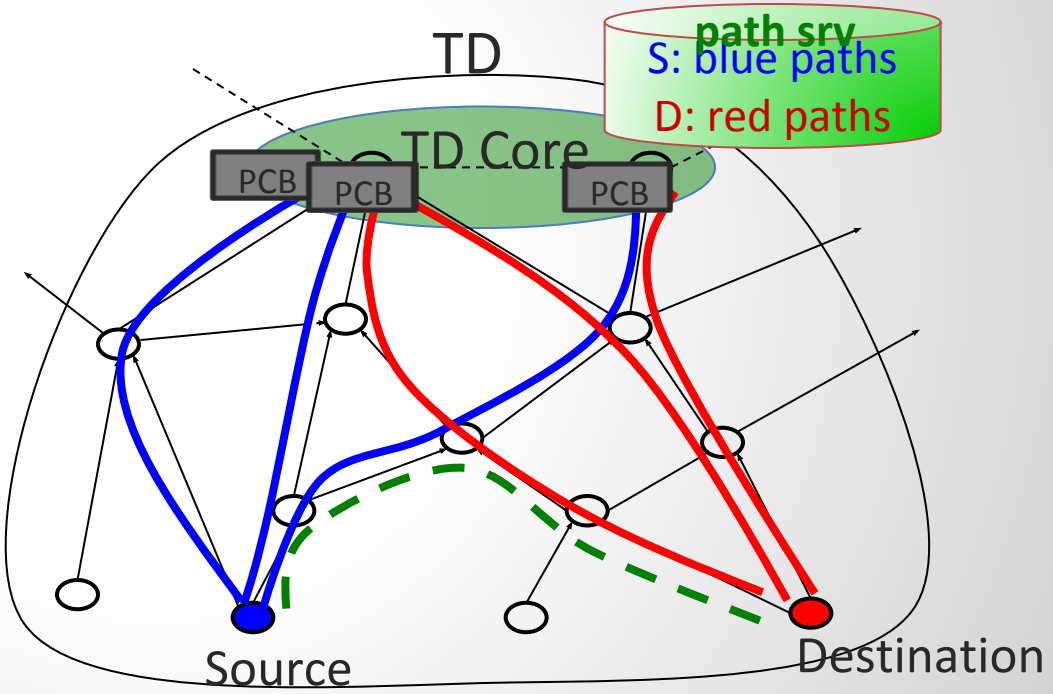
- SCION Overview
- Bootstrapping SCION over XIA
- Using Data Plane
- Implementation Details
- Current Status

# SCION Architectural Goals

- High availability, even for networks with malicious parties
  - Communication should be available if attacker-free path exists
- **Explicit trust for network operations**
- Minimal TCB: minimize trusted entities for any operation
  - Strong isolation from untrusted parties
- Operate with mutually distrusting entities
  - No single root of trust
- **Balanced route control for ISPs, receivers, senders**
- No circular dependencies during setup: enable rebootability
- Simplicity, efficiency, flexibility, and scalability

# SCION Architecture Overview

- Trust domain (TD)s
  - Isolation and scalability
  - Enforceable accountability
- Path construction
  - Path construction beacons (PCBs)
- Path resolution
  - Control
  - Explicit trust
- Route joining (shortcuts)
  - Efficiency, flexibility



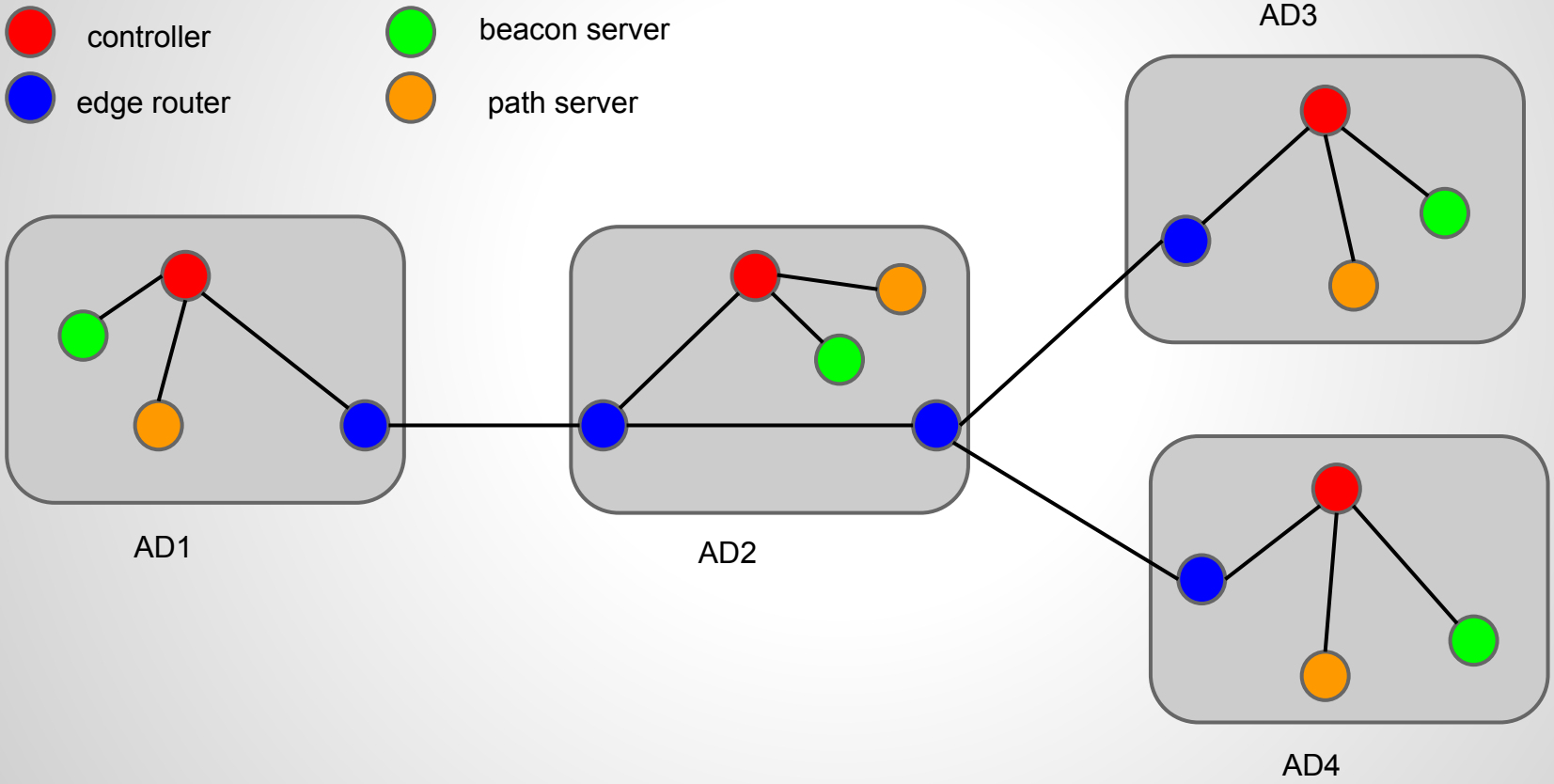
# SCION Components

- Certificate Server
  - Certificate, Policy, Topology, Key management
- **Beacon Server**
  - **Path Construction (PCB propagation, Path selection/registration (req), Path distribution)**
- **Path Server**
  - **Path registration/resolution**
- Border Router
  - Opaque Field verification, packet forwarding
- Switch
  - Abstract intra-domain routing
- Gateway
  - Backward compatibility

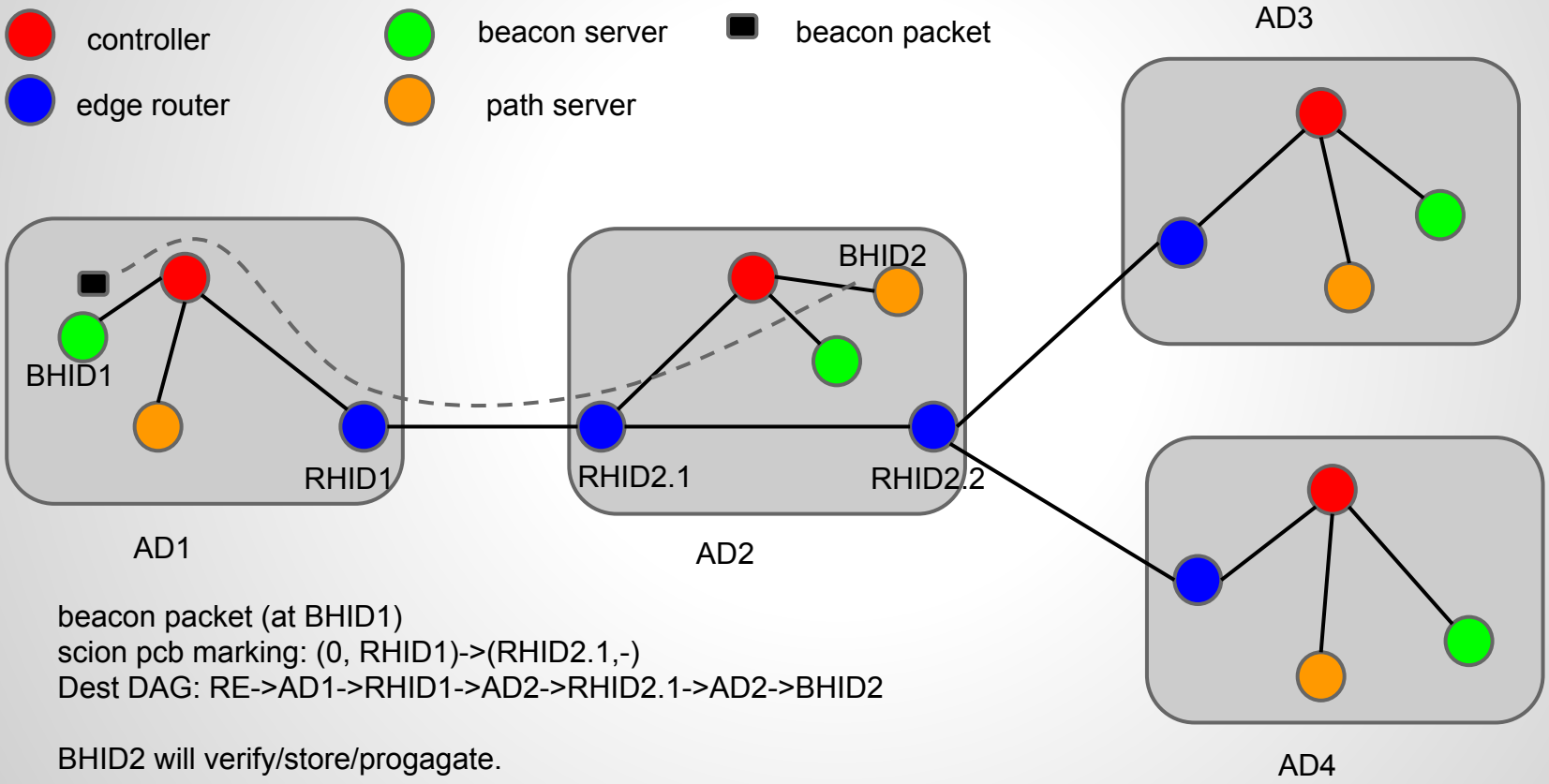
# Bootstrapping SCION over XIA

1. Each domain's controller populates intra-domain forwarding entries
   a. This includes entries for SCION Beacon and Path Server SIDs
2. Each domain's controller uses XBGP to setup forwarding entries for reaching other ADs
3. As defined by SCION, Beacon Servers propagate PCBs to other ADs
   a. Other ADs are reached using routes setup by XBGP
4. As defined by SCION, ADs use PCBs to construct and register paths with Path Servers

# Example topology



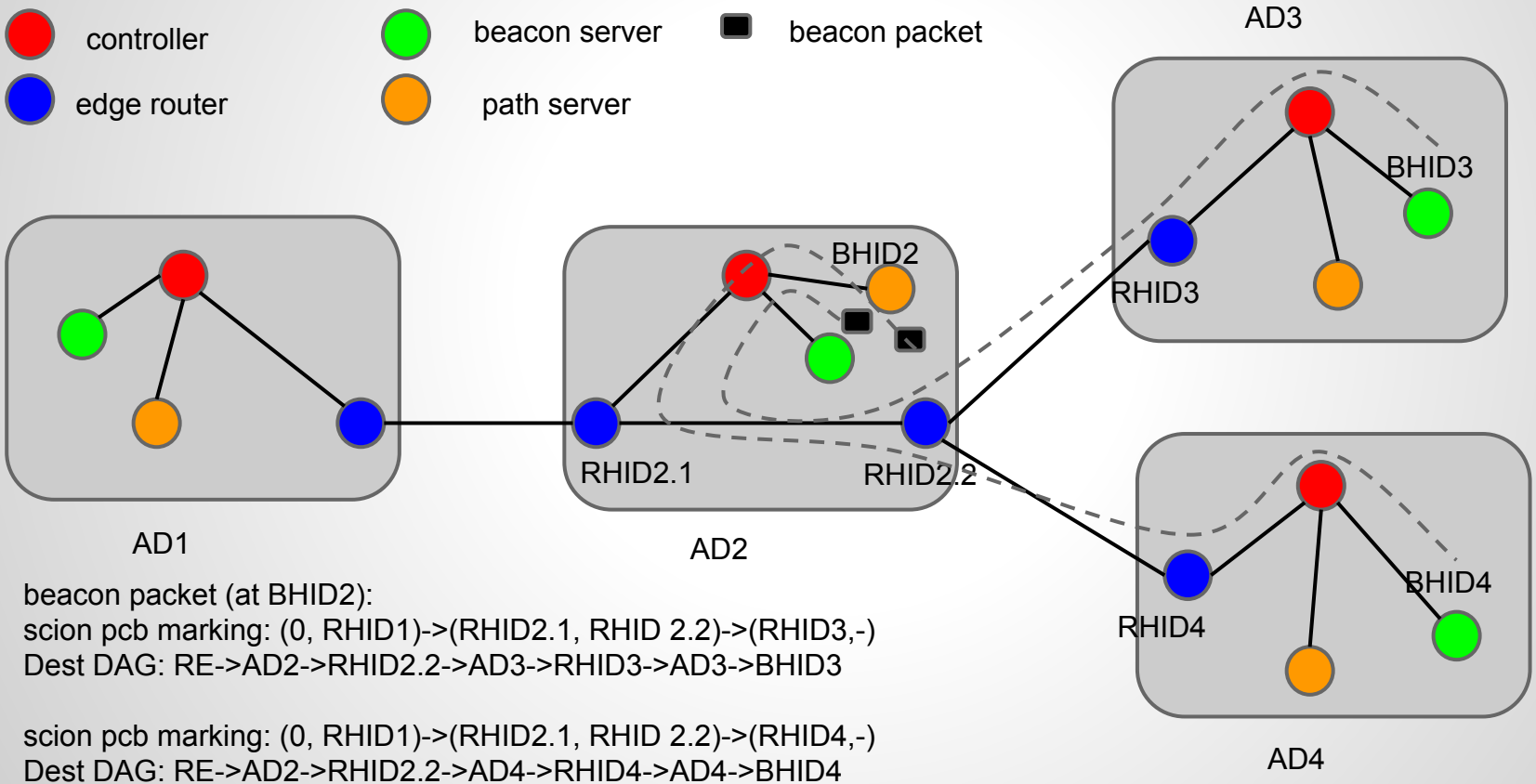
# Phase 1: beacon propagation




beacon packet (at BHID1)
scion pcb marking: (0, RHID1)->(RHID2.1,-)
Dest DAG: RE->AD1->RHID1->AD2->RHID2.1->AD2->BHID2

BHID2 will verify/store/progagate.

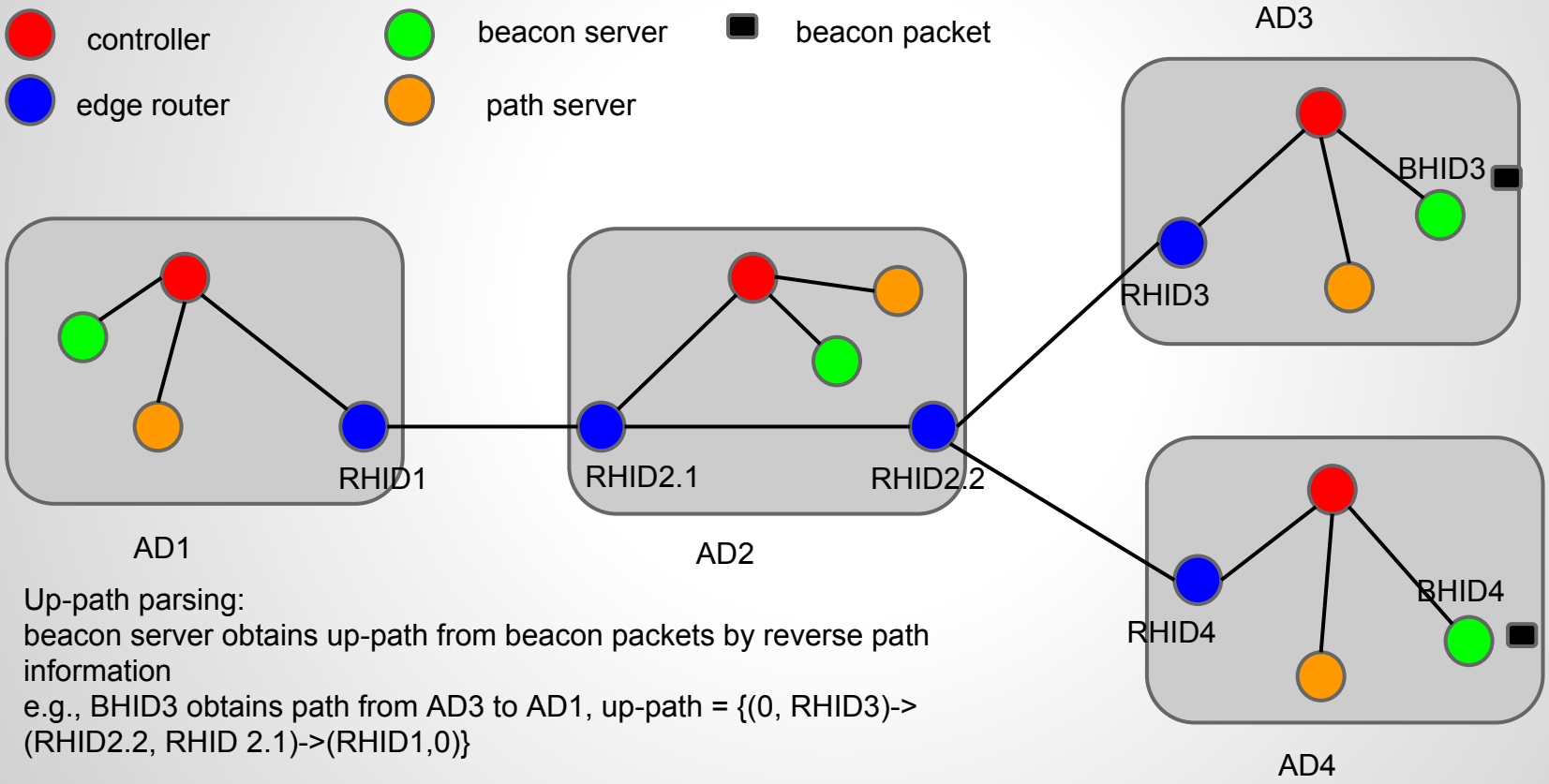
# Phase 1: beacon propagation




beacon packet (at BHID2):
scion pcb marking: (0, RHID1)->(RHID2.1, RHID 2.2)->(RHID3,-)
Dest DAG: RE->AD2->RHID2.2->AD3->RHID3->AD3->BHID3

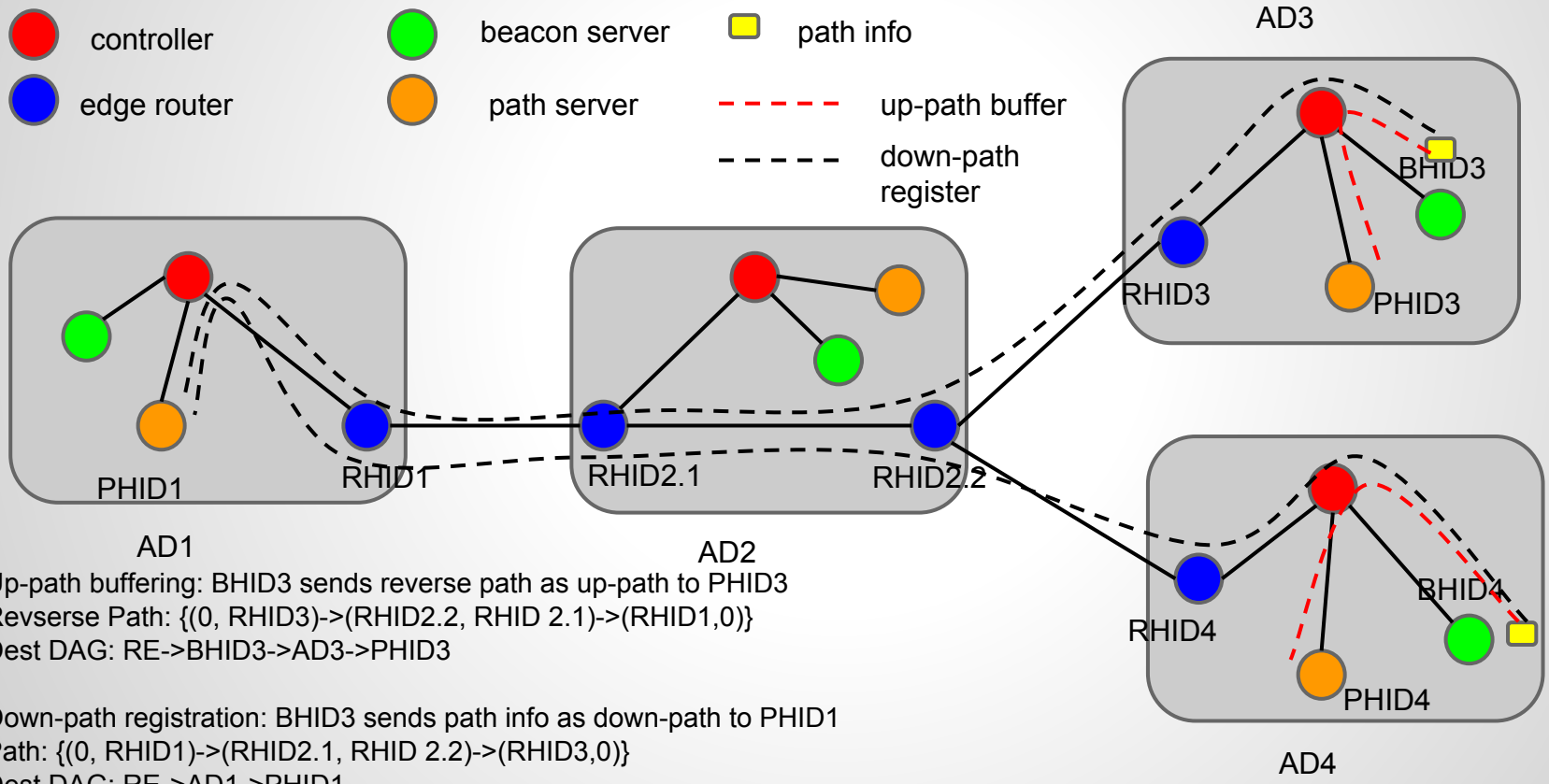
scion pcb marking: (0, RHID1)->(RHID2.1, RHID 2.2)->(RHID4,-)
Dest DAG: RE->AD2->RHID2.2->AD4->RHID4->AD4->BHID4

# Phase 2: path construction




Up-path parsing:
beacon server obtains up-path from beacon packets by reverse path information
e.g., BHID3 obtains path from AD3 to AD1, up-path = {(0, RHID3)-> (RHID2.2, RHID 2.1)->(RHID1,0)}

# Phase 2: path construction




Up-path buffering: BHID3 sends reverse path as up-path to PHID3
Revserse Path: {(0, RHID3)->(RHID2.2, RHID 2.1)->(RHID1,0)}
Dest DAG: RE->BHID3->AD3->PHID3

Down-path registration: BHID3 sends path info as down-path to PHID1
Path: {(0, RHID1)->(RHID2.1, RHID 2.2)->(RHID3,0)}
Dest DAG: RE->AD1->PHID1

# SCION over XIA Data Plane

- Allow the choice to use SCION for inter-domain routing, when explicit end-to-end trusted paths are desired
- Possibilities for who chooses the inter-domain routing protocol:
  - End-host can choose SCION:
    - On a per-packet basis but does not choose the SCION path
    - On a per-packet basis and chooses the SCION path
  - Transparent to end-host, domain controller can choose SCION:
    - On a per-packet basis
    - On a per-host basis
    - **On a domain-wide basis***

* We will implement this first for the sake of faster development.

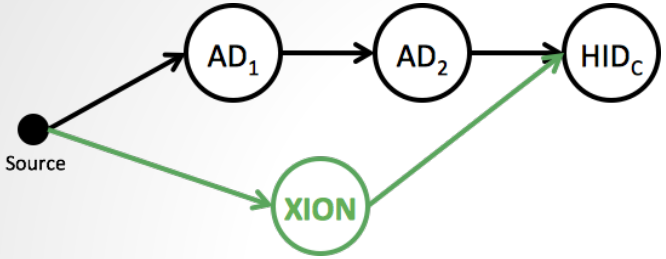
# Usage

1. We introduce a new principle type called XION for packets that should use SCION for inter-domain routing
    a. For the sake of discussion, assume the end-host will choose SCION by specifying a DAG of the form “XION: 123, HID: 456” rather than one of the form “AD: 123, HID: 456”
2. When a gateway router receives a packet with a XION destination domain, it will query the Path Servers and construct a SCION path to that domain
3. This path will then be added to the packet header as part of the XIA Extension Header
4. The gateway router and all subsequent routers in the domain forward packet to the first egress router
5. Egress router sends it to the next domain’s ingress router
6. Repeat this until packet eventually reaches the final destination AD’s ingress router
7. Routers within the final destination AD then forward packet to the final destination HID/SID

# XION Principle Type

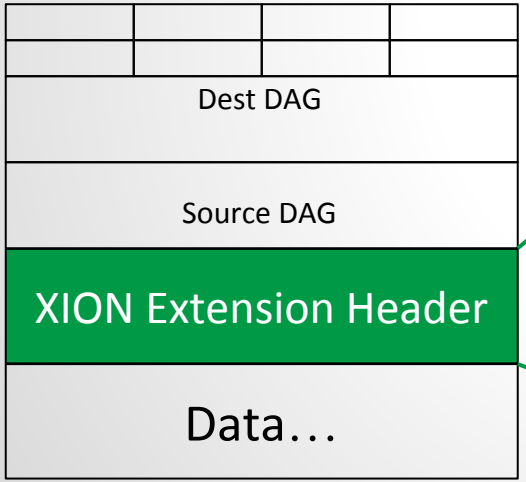
- How to handle XIDs which require additional routing information?
  - **Option 1:** Store the information in the XIA Extension Header.
  - **Option 2:** Encode the information in the DAG
    - e.g. with multiple/new XIDs or variable-length XIDs.
- We have chosen Option 1 for now as it will be simpler to implement

# XION Principal Type

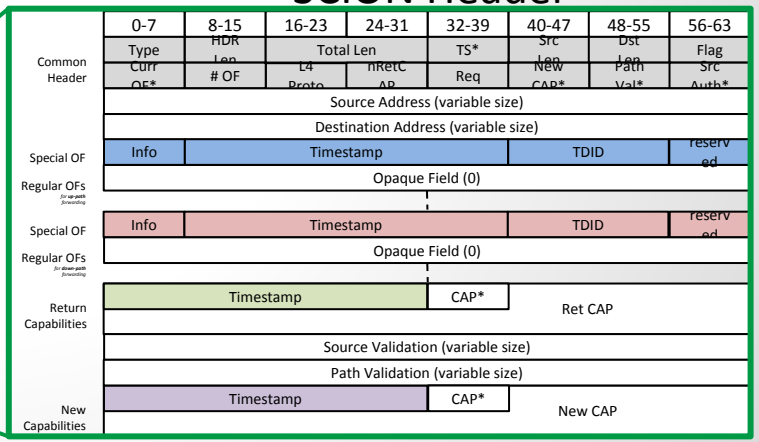



If Source needs ***Explicit Trust*** **on the Path (Network)** to HID

XIA Header




SCION Header



# Current Status

- Beacon service (done):
  - Generation at core AD: done
  - Propagation (including verification) at inter/stub ADs: done
- Path service (done):
  - Up-path parsing at inter/stub ADs
  - Up-path buffering at inter/stub ADs
  - Down-path registration at core AD
- Data plane:
  - Down-path request/response for destination ADs (path server) (done)
  - End-to-end path resolution (path server) (in progress)
  - Construction of XIA header using XION principal type (in progress)
  - End-to-end data communication between peer endhosts (in progress)

# Backup

# Trust Domain Decomposition

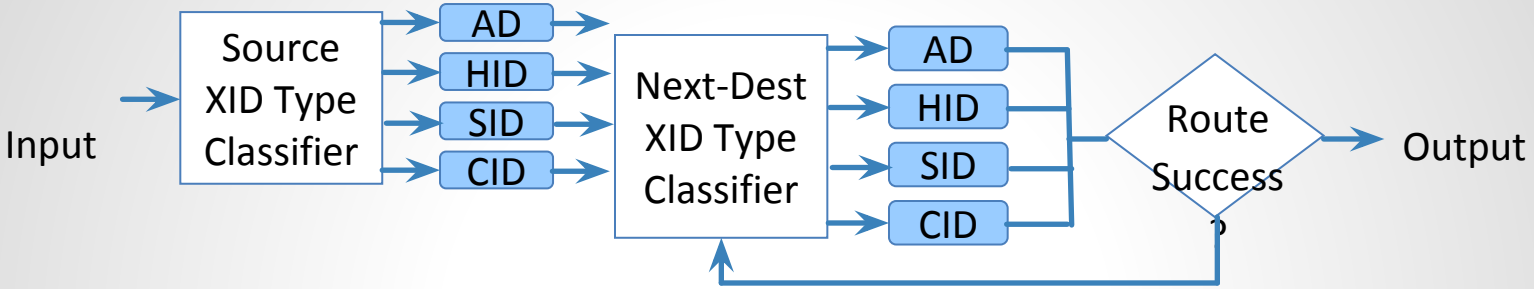
- Global set of TD (Trust Domains)
  - Map to geographic, political, legal boundaries
  - Usually corresponds to a jurisdiction
  - Provide enforceable accountability
- TD Core: set of top-tier ISPs that manage TD
  - Route to other TDs
  - Initiate path construction beacons
  - Manage Address and Path Translation Servers
  - Handle TD membership
  - Root of trust for TD: manage root key and certificates
- AD: Autonomous Domain
  - Transit AD or endpoint AD

# Path Construction

Goal: each endpoint learns multiple verifiable paths to its core

- Discovering paths via Path Construction Beacons (PCBs)
  - TD Core periodically initiates PCBs
  - ADs asynchronously propagate PCBs
- ADs perform the following operations
  - Collect PCBs
  - For each customer/peer AD, select which *k* PCBs to forward
  - Update cryptographic information in PCBs
- Endpoint AD receives at least *k* PCBs from each provider AD, selects *k* down-paths to advertise

# Forwarding Engine




- Principal-independent processing defines how to interpret the DAG
- Principal-dependent processing realizes forwarding semantics for each XID type

# Architecture