A Layered Naming Architecture

Michael Walfish

MIT Computer Science and Artificial Intelligence Lab

Joint work with:
H. Balakrishnan, M. Krohn, K. Lakshminarayanan,
S. Ratnasamy, S. Shenker, I. Stoica, J. Stribling

CSAIL
MIT COMPUTER SCIENCE AND ARTIFICIAL INTELLIGENCE LABORATORY

IRTF HIP RG

6 August 2004

Outline

I. Overview of “Layered Naming Architecture”

II. Application-level example

III. Network-level examples

“A Layered Naming Architecture”

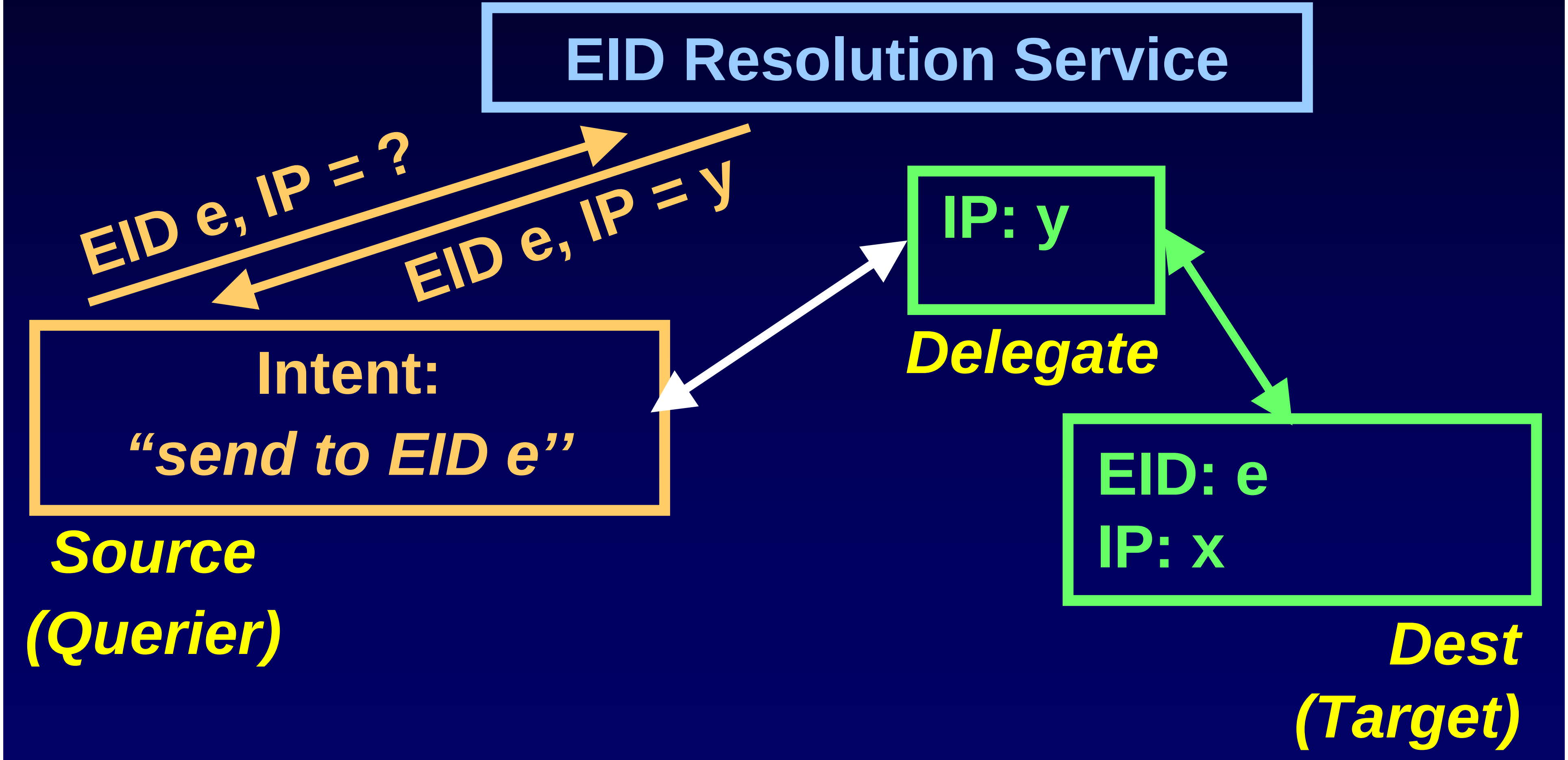
- View: naming could solve some arch. probs.

- Principle 1: “don’t bind names too early”
  - Need two new types of names
  - SIDs (Service IDs)
  - EIDs (End-point IDs)

- Principle 2: “names should be flat”

“A Layered Naming Architecture”, Cont.

- Principle 3: “let names resolve to delegates”



EID Resolution Service

EID e, IP = ?

EID e, IP = y

Intent:

“send to EID e”

Source (Querier)

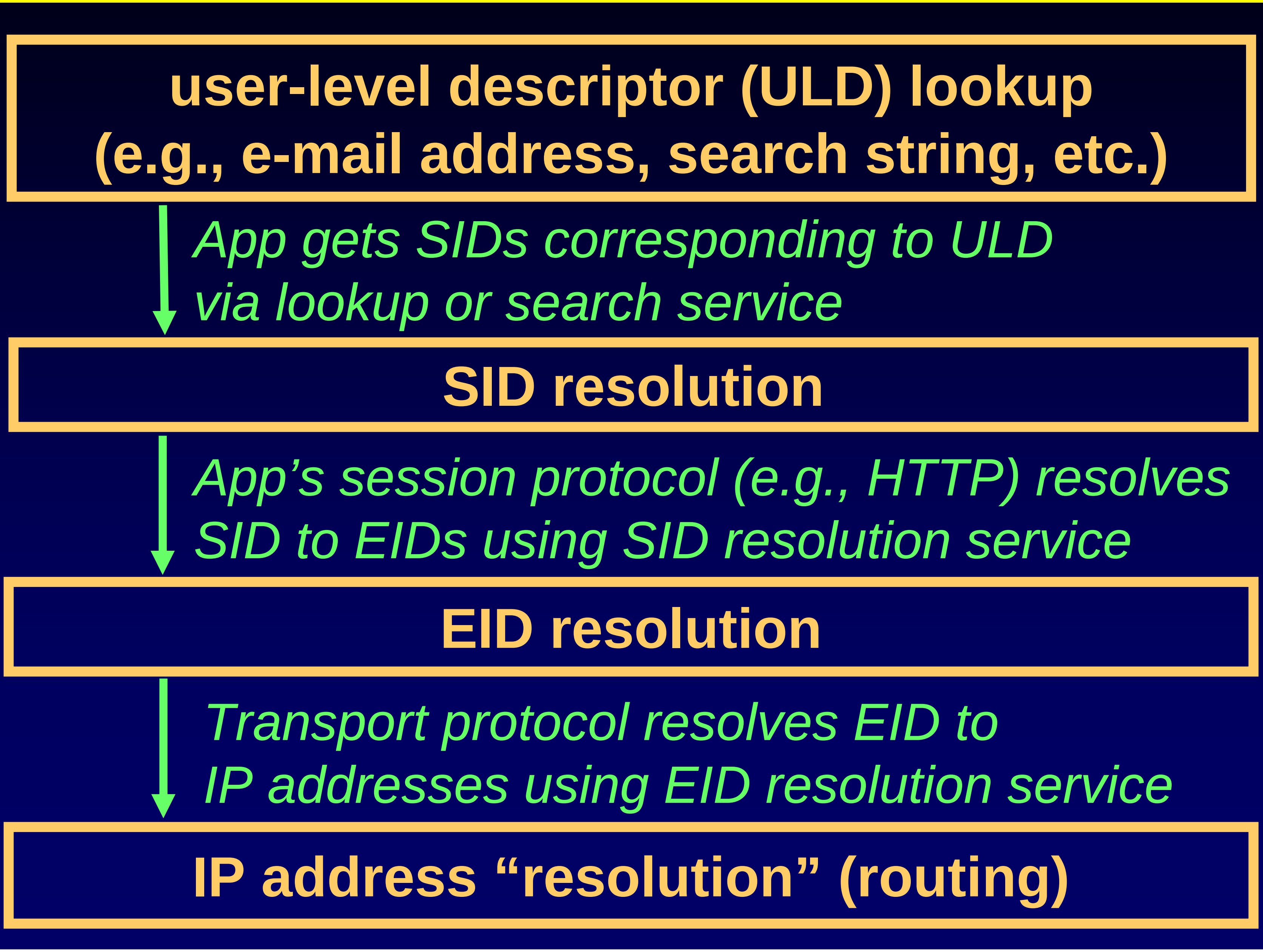
IP: y

Delegate

EID: e
IP: x

Dest (Target)

The Layers



user-level descriptor (ULD) lookup
(e.g., e-mail address, search string, etc.)

App gets SIDs corresponding to ULD
via lookup or search service

SID resolution

App's session protocol (e.g., HTTP) resolves
SID to EIDs using SID resolution service

EID resolution

Transport protocol resolves EID to
IP addresses using EID resolution service

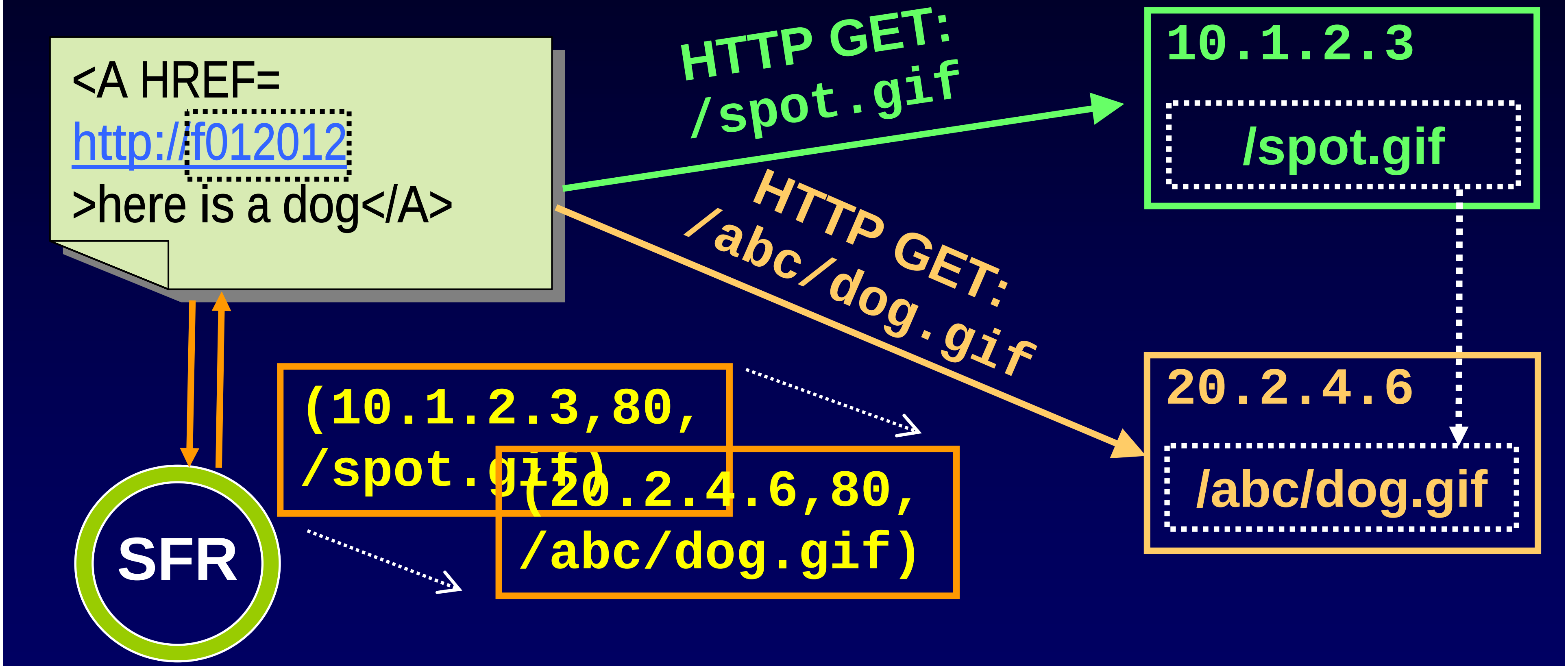
IP address "resolution" (routing)

Benefits

- Mobility and multi-homing (from HIP)
- Data and services become first-class
  - Because they can be persistently named
- Architectural coherence for middleboxes

SIDs in Action

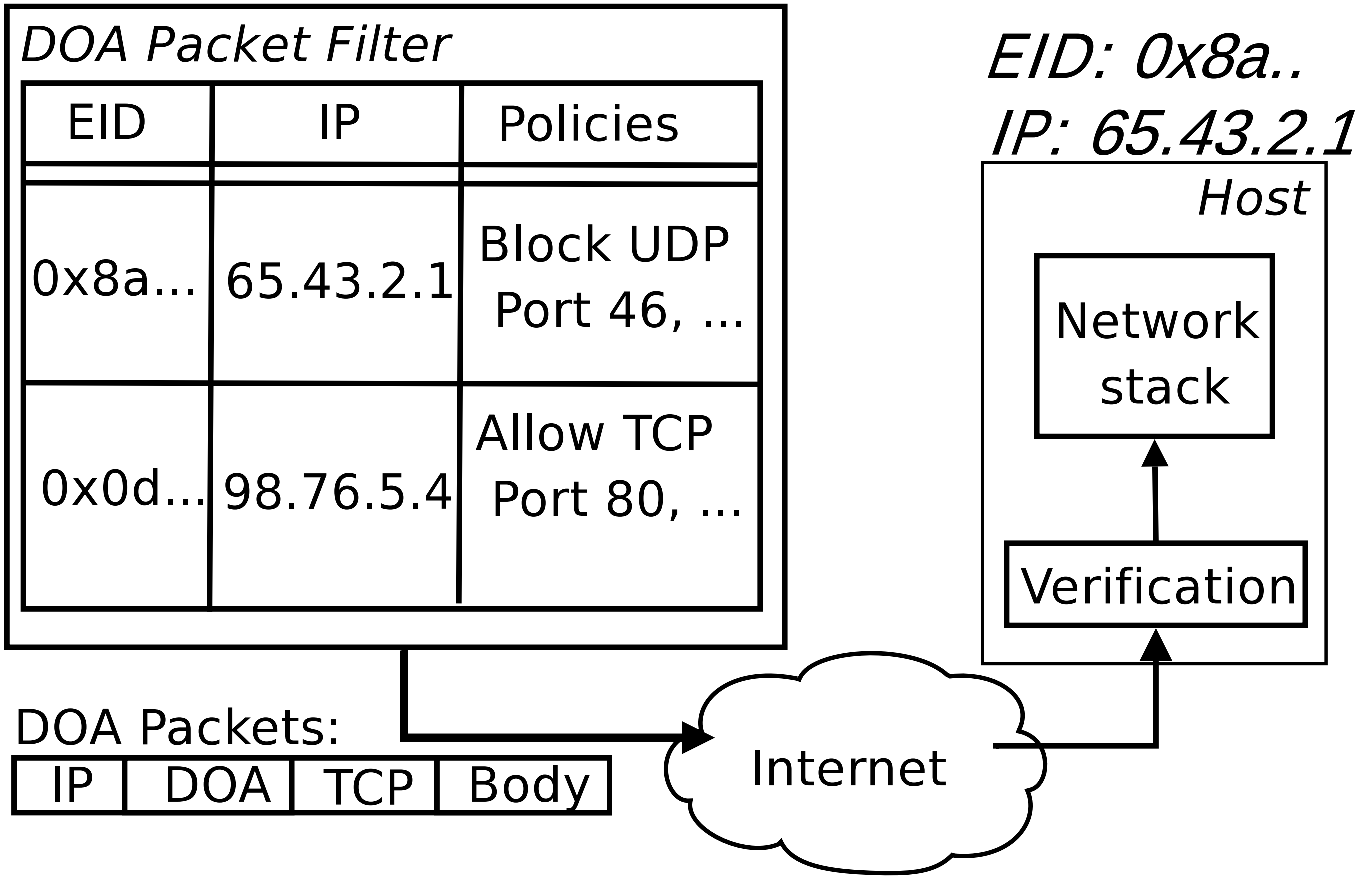
One example:



(Could use EIDs instead of 10.1.2.3, 20.2.4.6)

EIDs in Action (1): Remote Packet Filter

- Imagine third-party firewall services
  - Need robust notion of host identity
  - Need ability to delegate

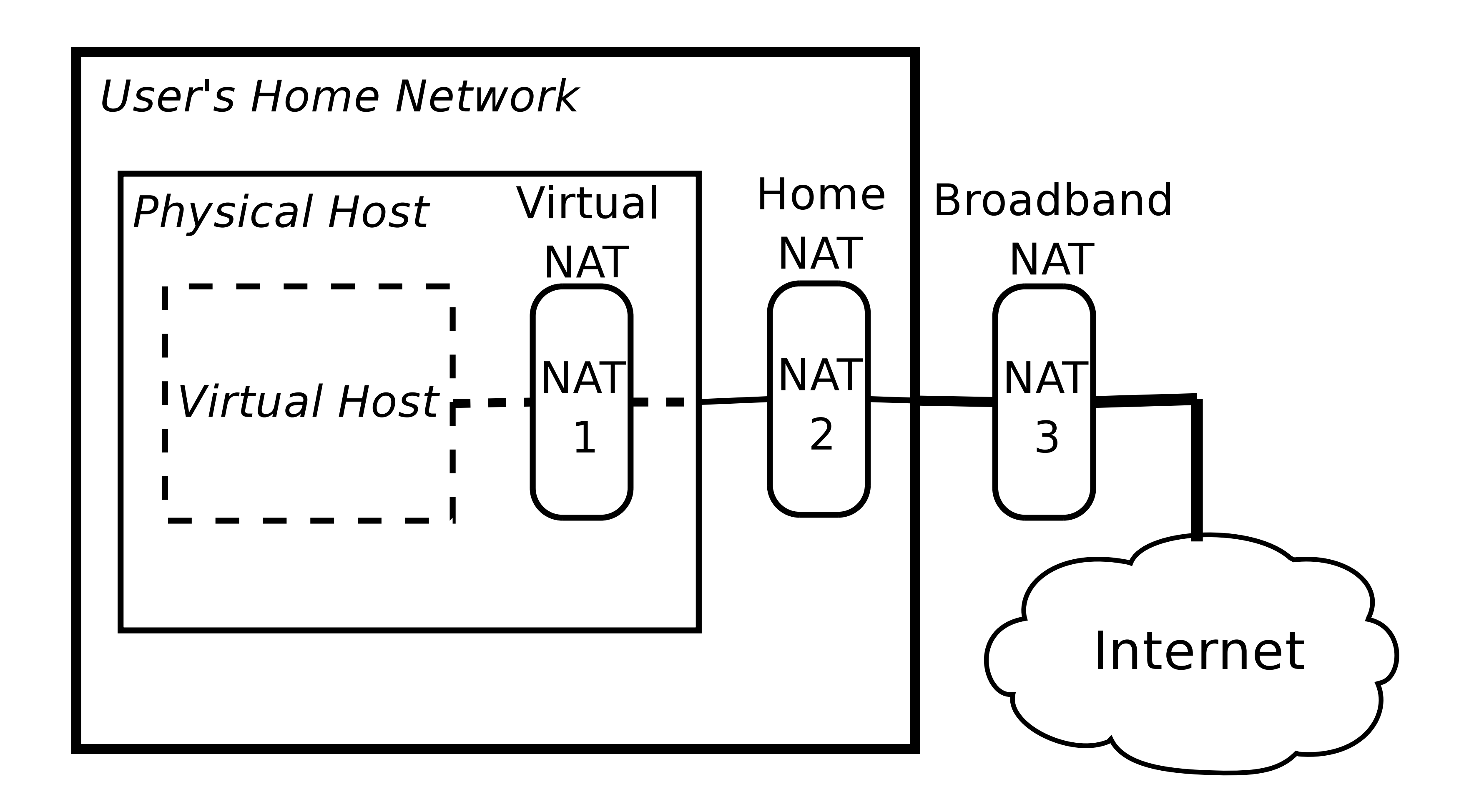


EID	IP	Policies
0x8a...	65.43.2.1	Block UDP Port 46, ...
0x0d...	98.76.5.4	Allow TCP Port 80, ...

DOA Packets:

IP	DOA	TCP	Body

EIDs in Action (2): Cascaded NATs



- EIDs (not overloaded ports!) help demux

High-level Points

- Not focusing on specifics of implementation for now . . .
- Insights about network-level IDs apply to application-level IDs (and vice-versa!)
- Flat names, delegation powerful primitives
- These primitives have several benefits
  - mobility / multi-homing
  - services and data get first-class names
  - coherent story for middleboxes