Future Internet Architecture and its Security Implications

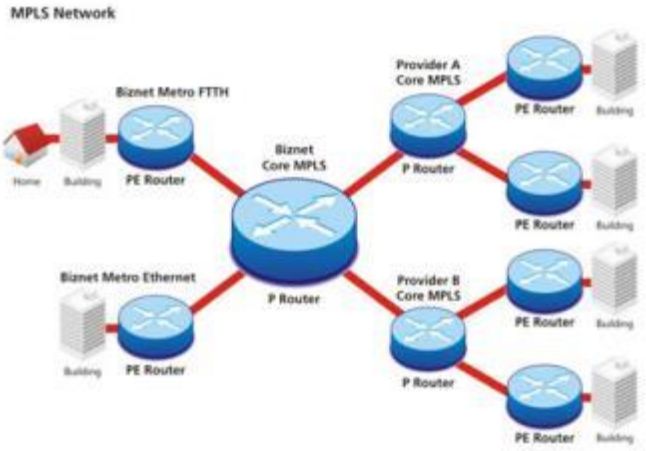
Hilfi Alkaff, Rohan Sharma, Abhishek Sharma

CS598 MCC: Network Security

Evolution of Computer Architecture

Circuit Switched

Packet Switched



Next Architecture

?

- Each architecture developed to solve extant problem
- Telephony: basic voice communication
- Packet-switching: few computers, many users
- Current problem: Many computers, fewer users

Problem with Today's Networks

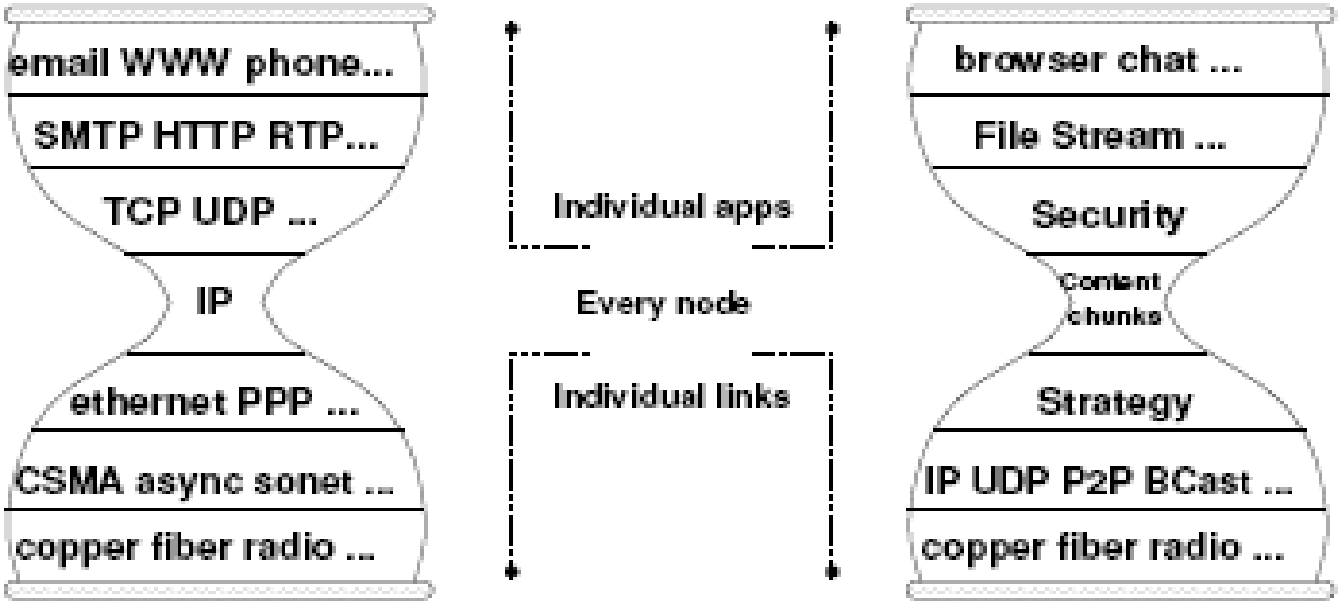
- URLs and IP addresses are overloaded with locator and identifier functionality
  - Moving information = changing it's name => 404 file not found
- No consistent way to keep track of *identical copies*
  - No consistent *representation of information* (copy-independent)
- Multiple Interface
  - Most mobile phones could access the network in multiple ways (3G & WiFi)
- Information dissemination is inefficient
  - Can't benefit from existing copies (e.g. local copy on client)
  - No "anycast": e.g., get "*nearest*" copy
  - Problems like *Flash-Crowd effect, Denial of Service, …*

Problem with Today's Networks

- Can't trust a copy received from an untrusted node
  - Security is host-centric
  - Mainly based on *securing channels* (encryption) and *trusting servers* (authentication)

Content-Centric Networking (CoNext ’09)

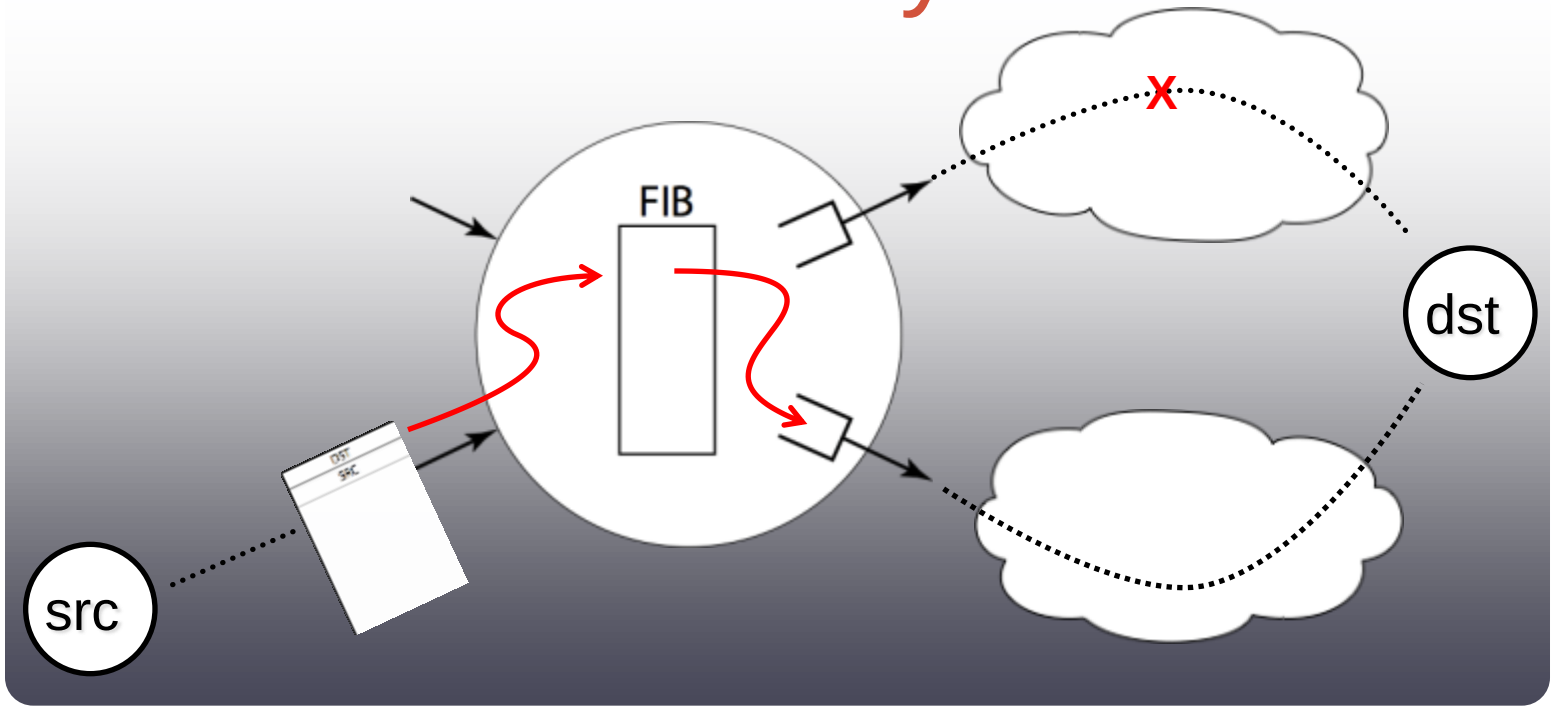
- Content-centric networking (CCN) is an alternative approach to the data approach on the network rather than the location approach, based on concept of *what* rather than *where*.



Content-Centric Networking

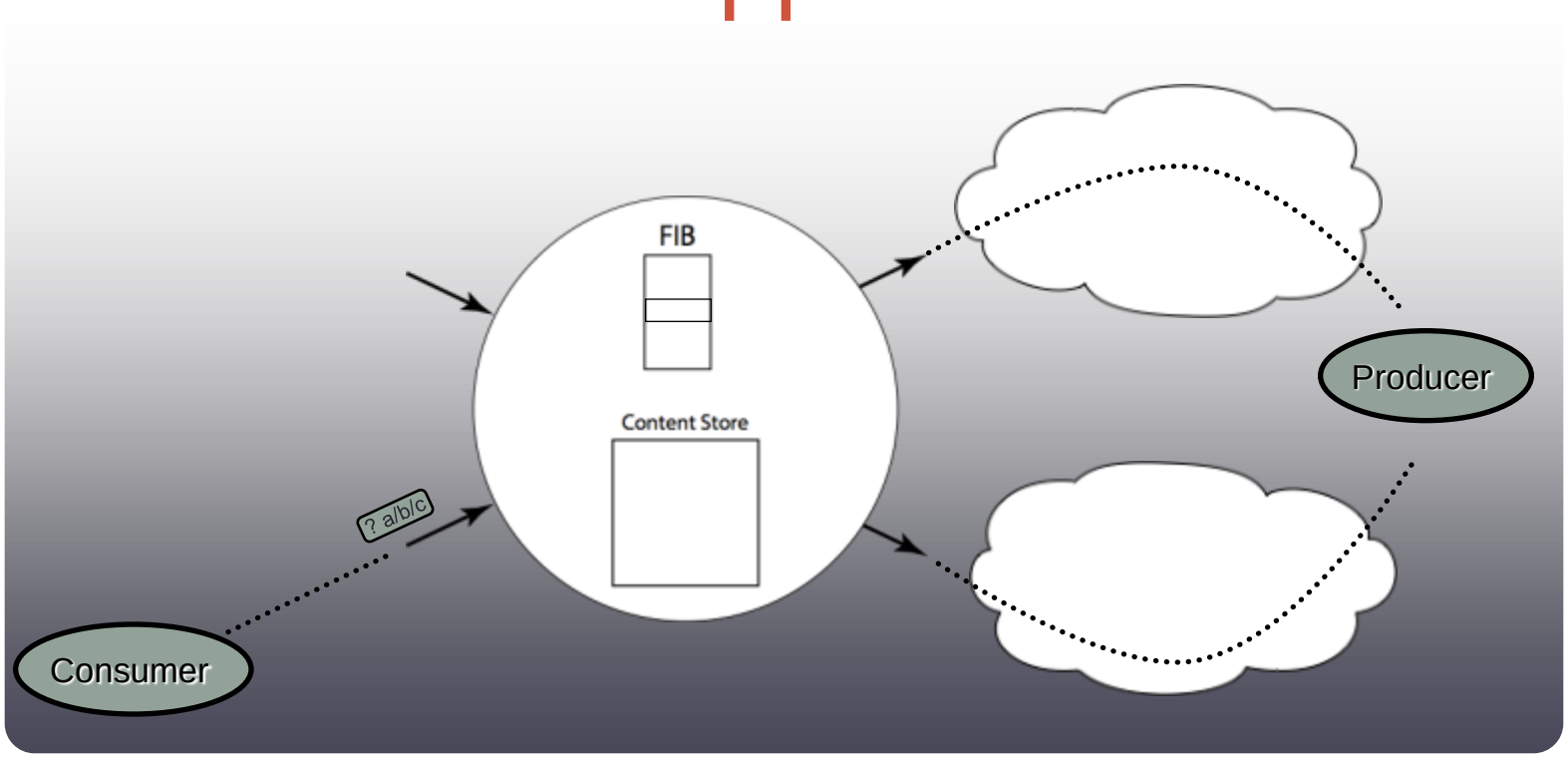
- Two packet types: *Interest* and *Data*
- Hierarchical content naming scheme
  - Allows dynamic content generation: *active names*
- CCN node has 3 components: FIB, Content Store and PIT
  - FIB: Forwarding table, allows multiple output faces
  - Content Store: Buffer, also caches *Data* packets
  - PIT: Pending *Interest* Table

Today

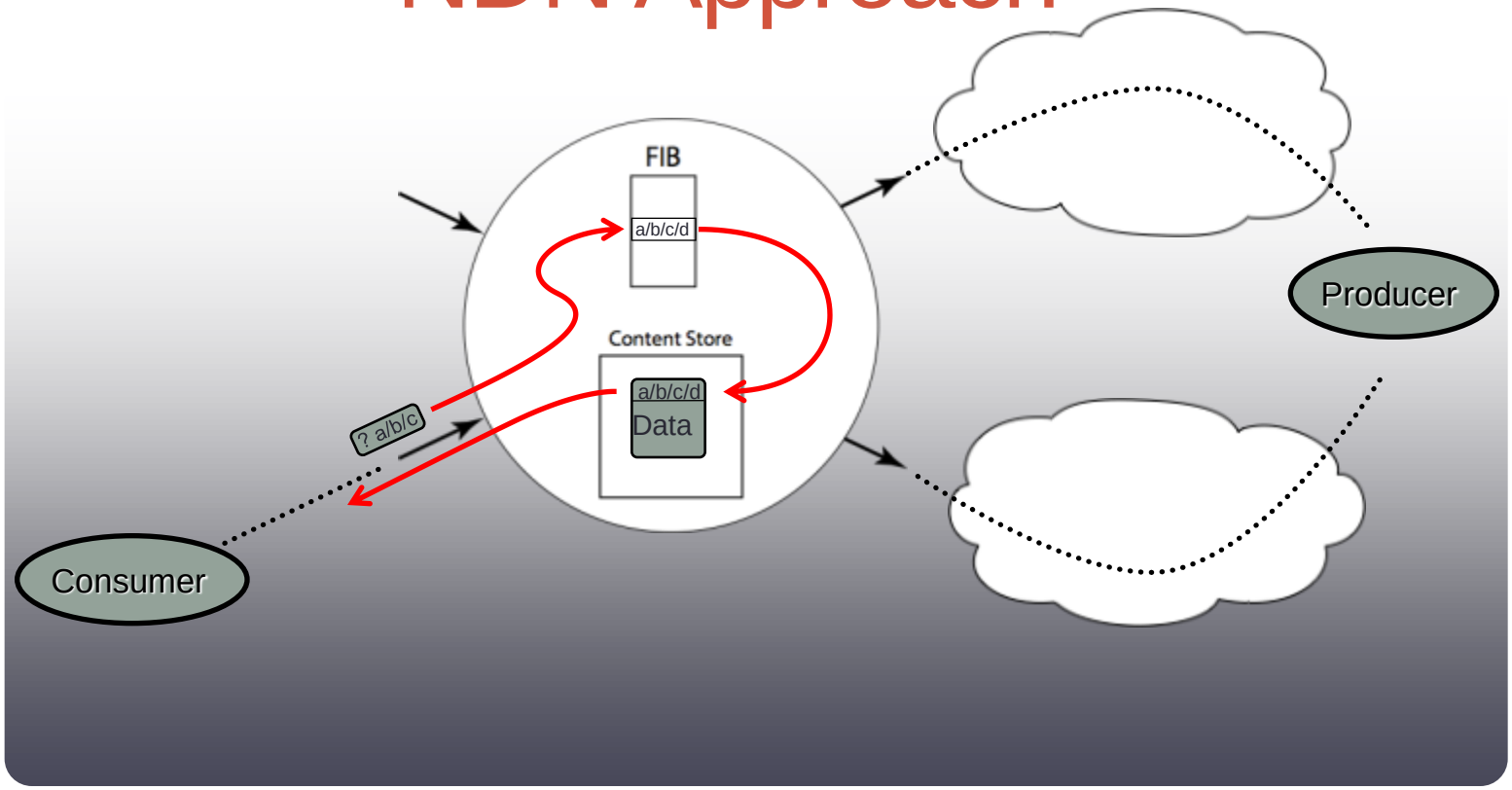


Path determined by global routing, not local choice

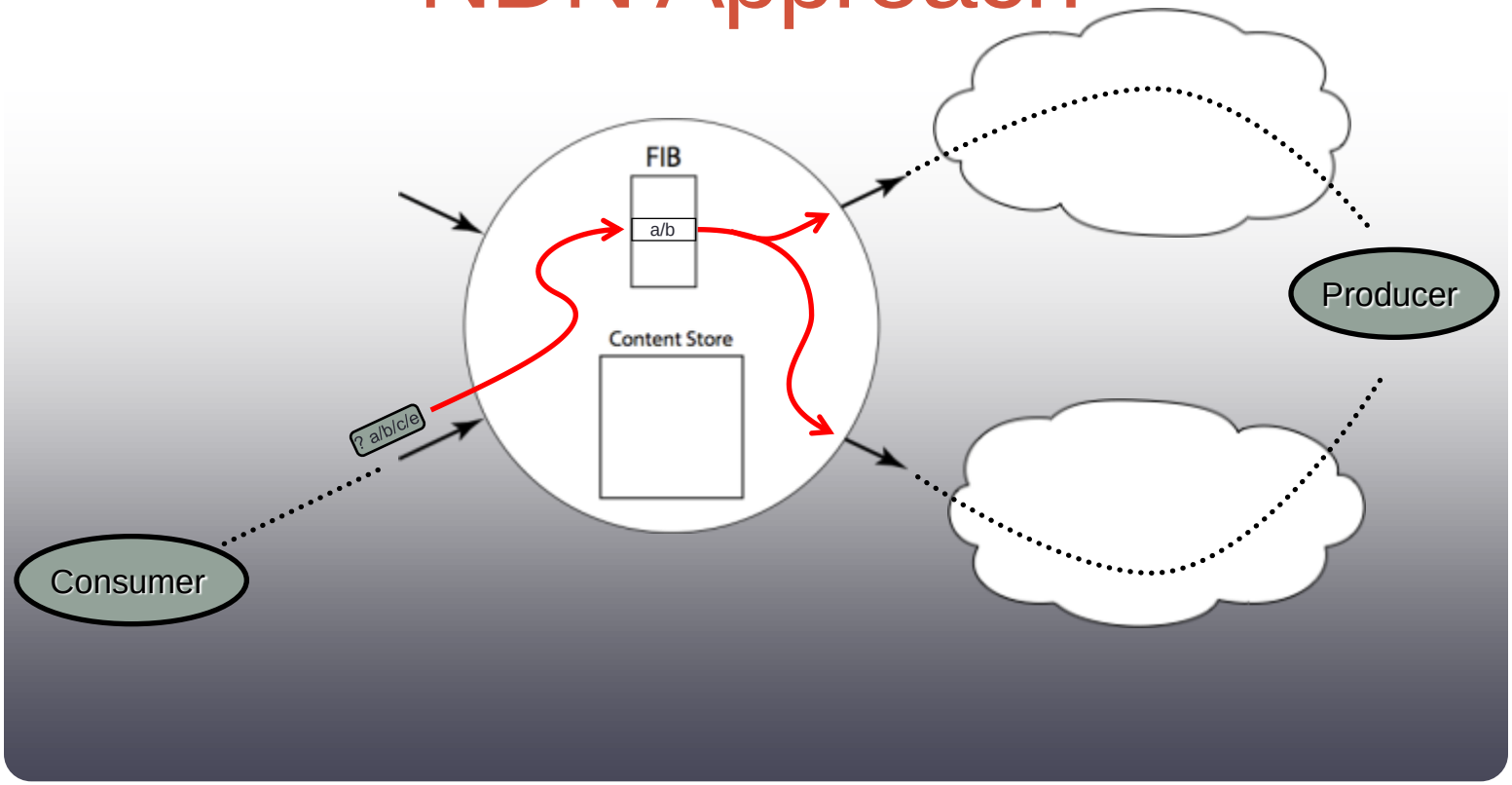
NDN Approach



NDN Approach



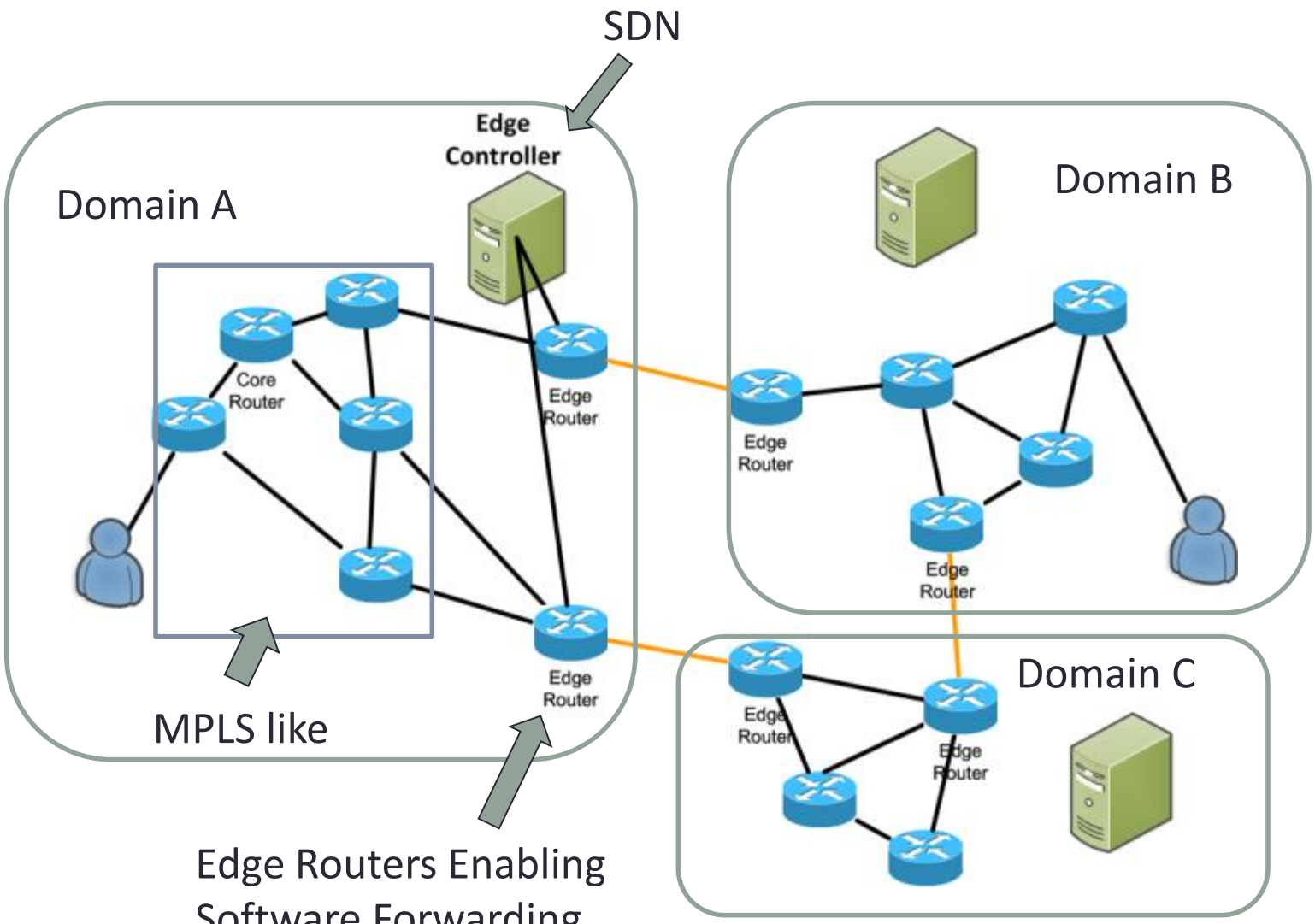
NDN Approach



- Packets say ‘what’ not ‘where’ (no src or dst)
- Forwarding decision is local
- Upstream performance is measurable

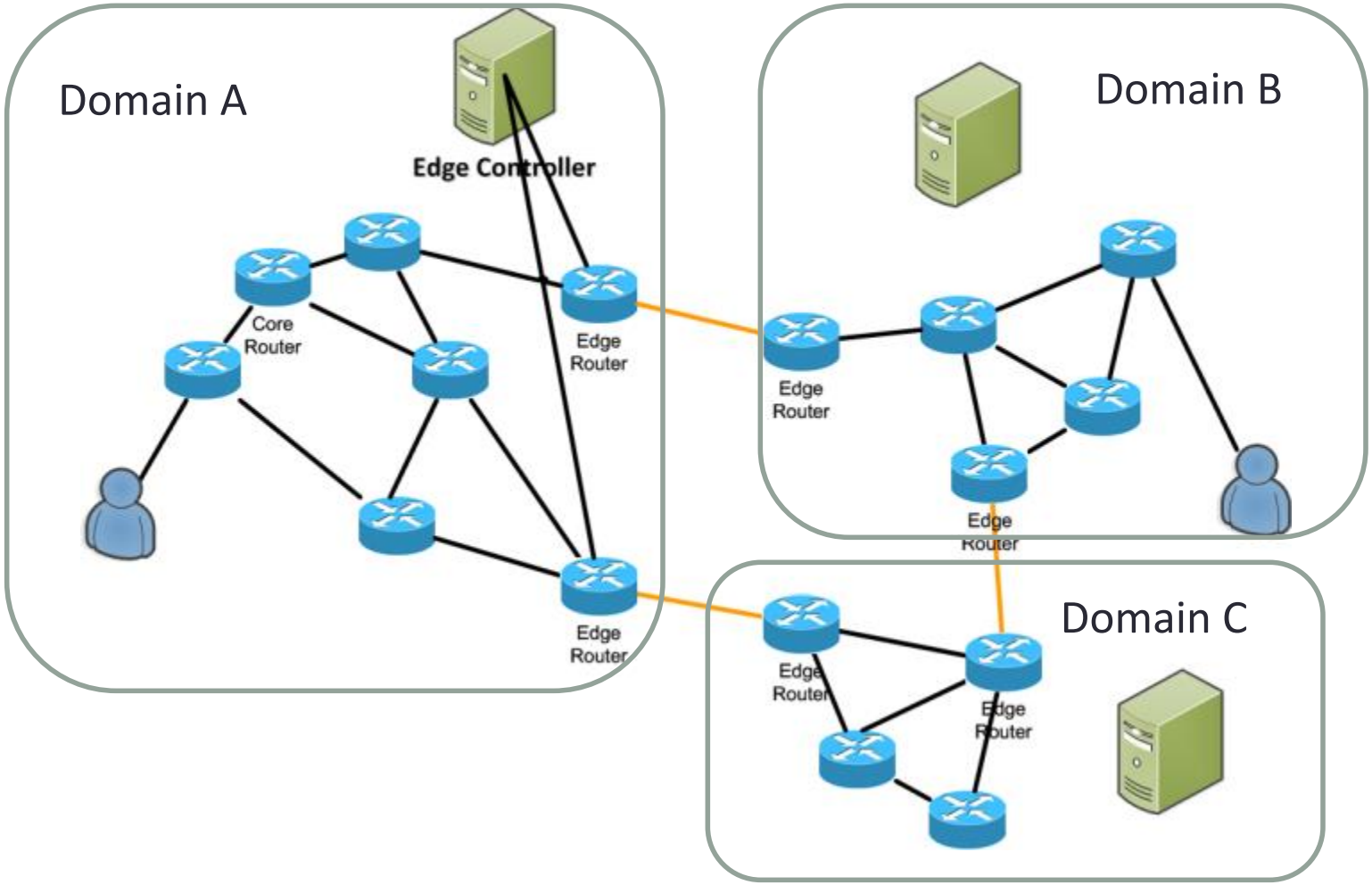
Architecture?

Software Defined Internet-Architecture (SDIA)
Hotnets '12



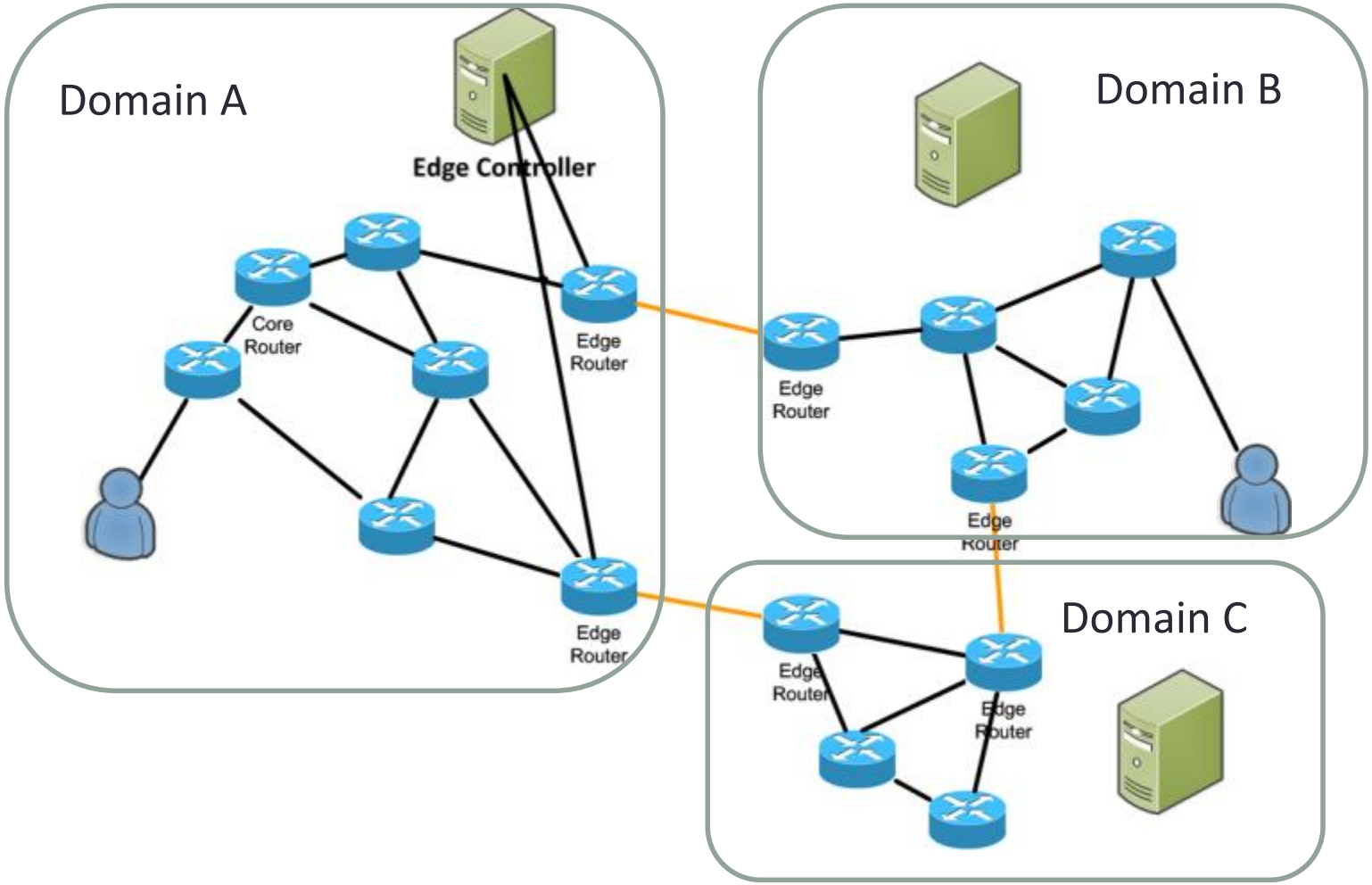
SDIA

- Separate inter-domain and intra-domain addressing



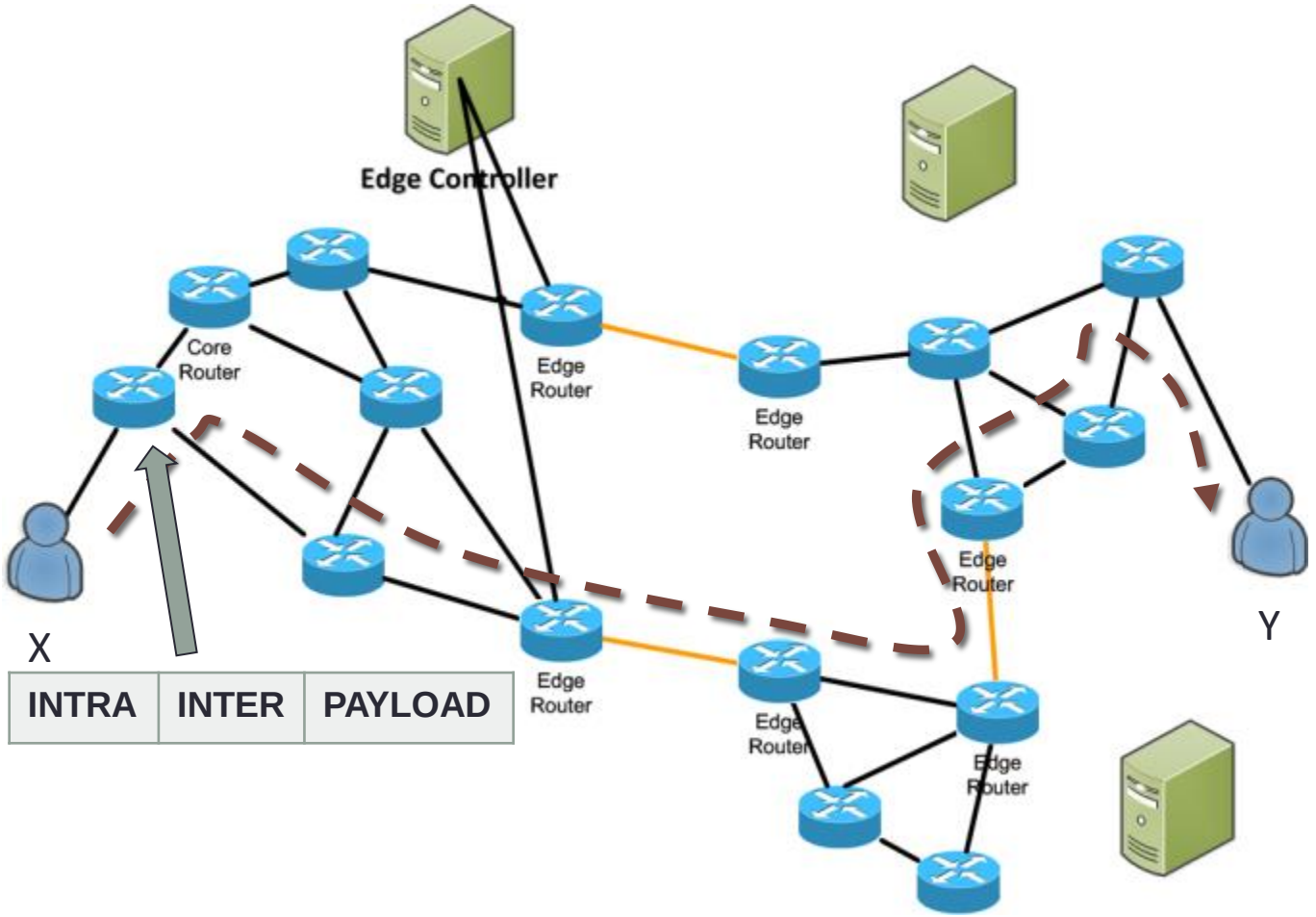
SDIA

- Push intelligence to the edges



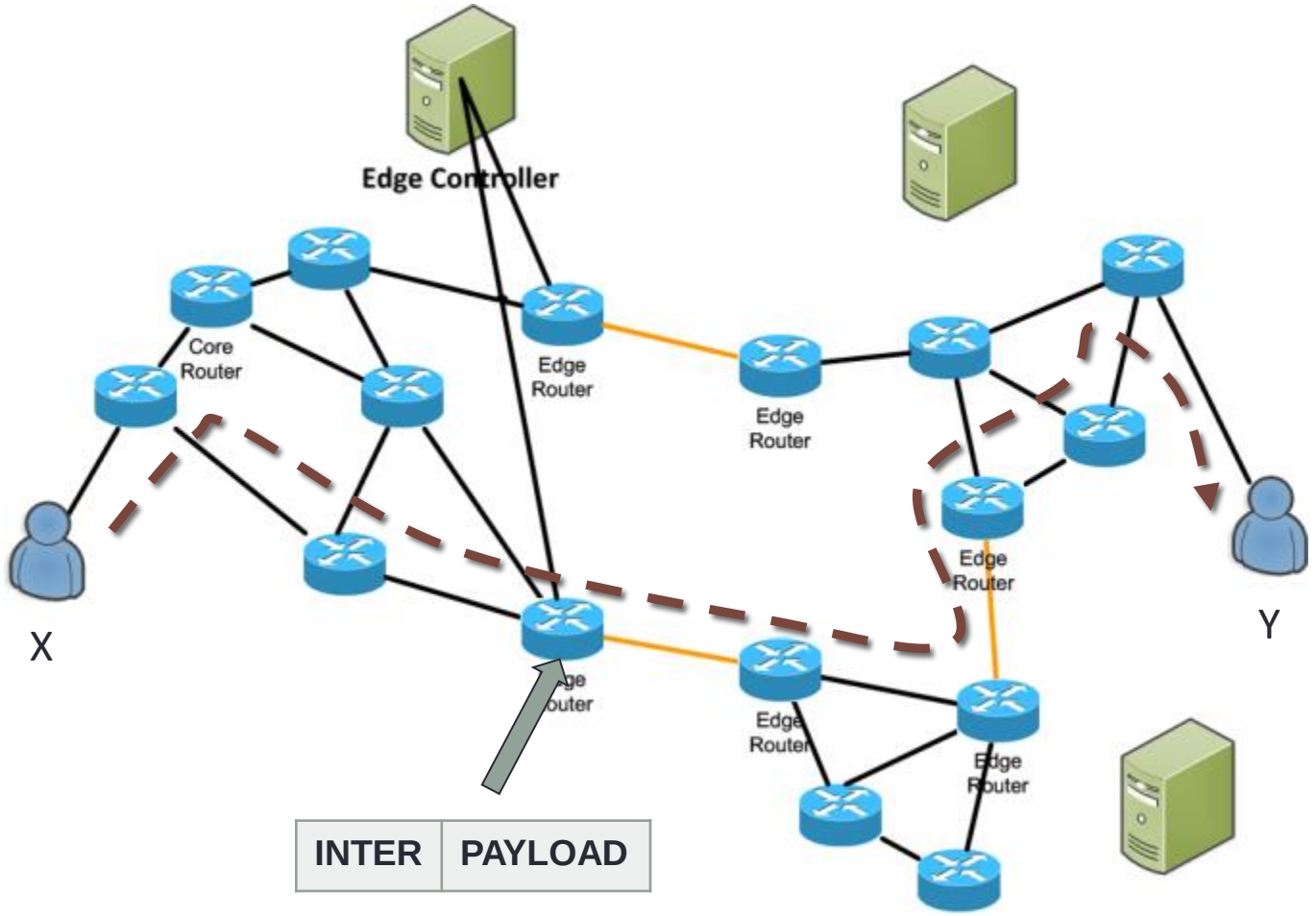
SDIA

- X to Y communication



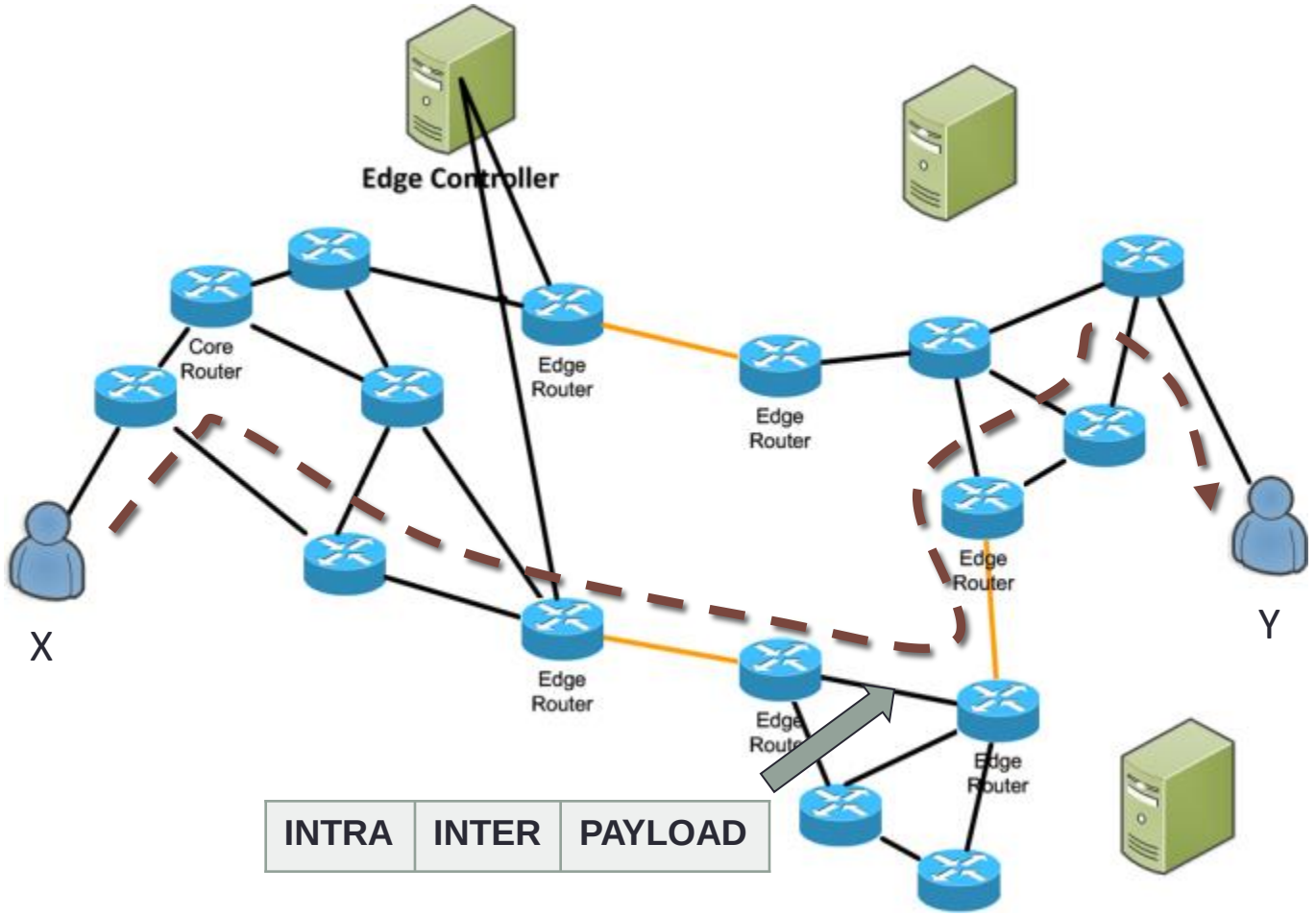
SDIA

- X to Y communication



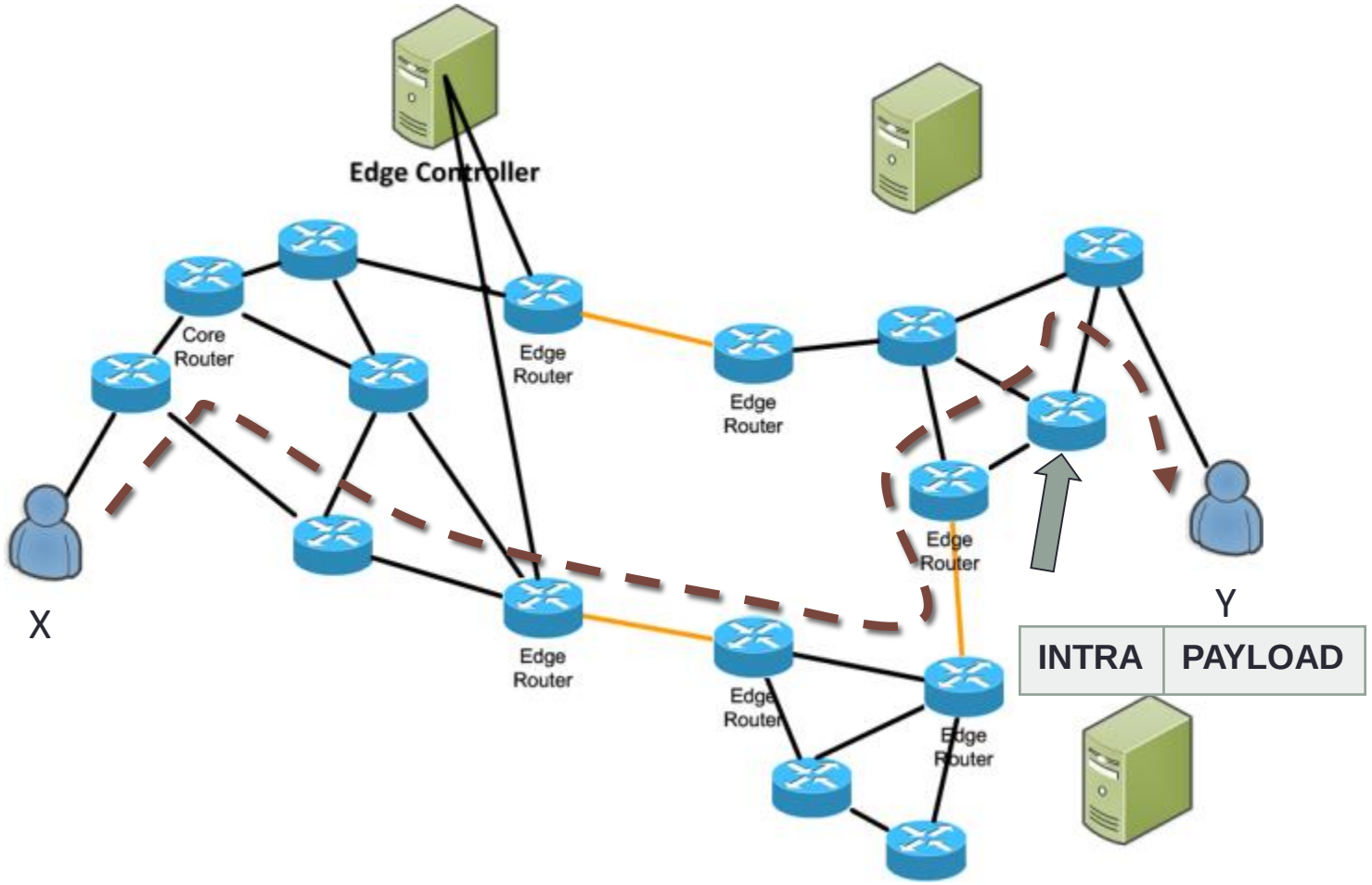
SDIA

- X to Y communication



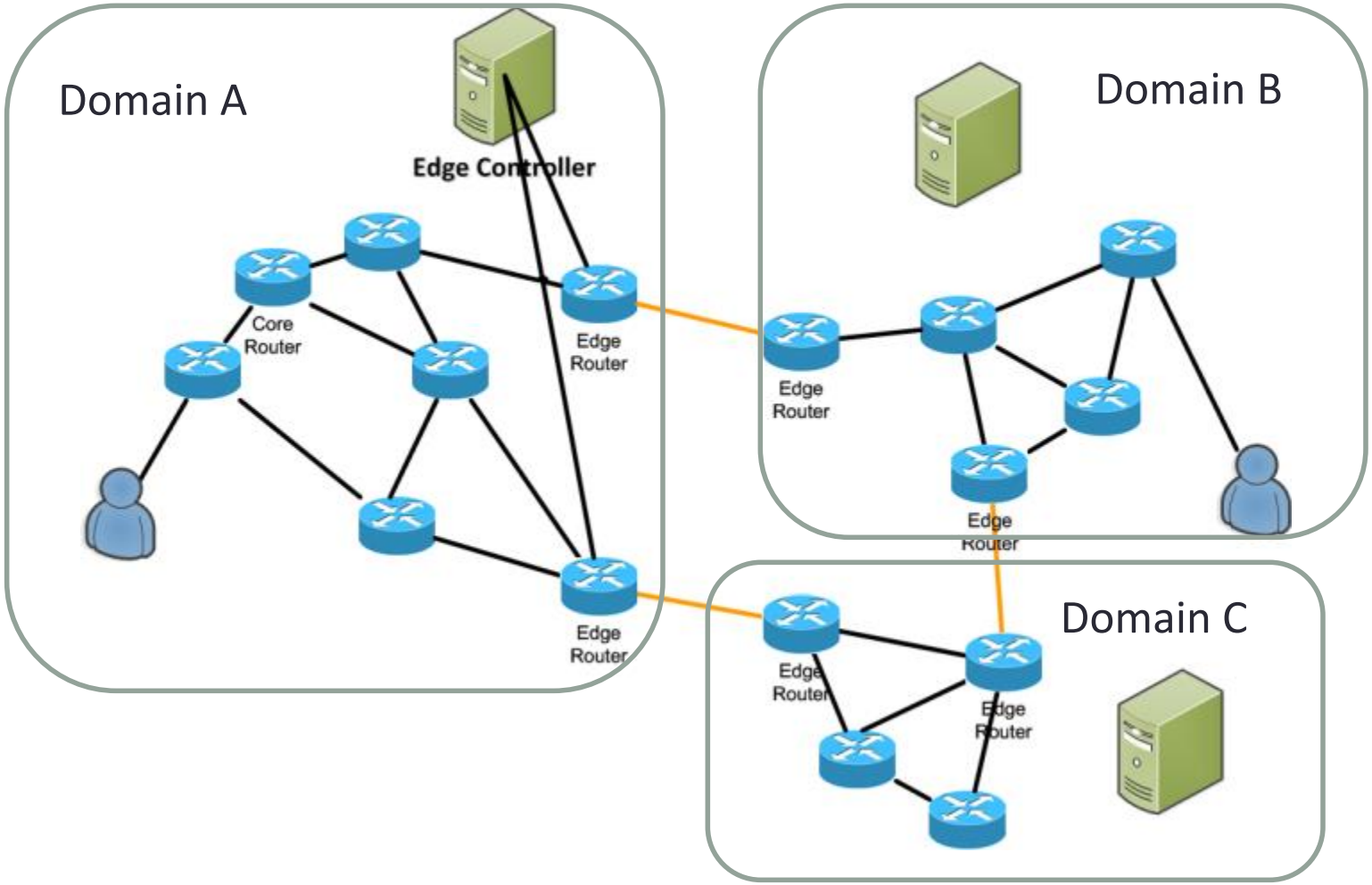
SDIA

- X to Y communication



SDIA

- Pros: Implement new architecture from the edges



Securing Content

Content Packet = ⟨ *name*, *data*, *signature* ⟩

Any consumer can ascertain:

- Integrity: is data intact and complete?
- Origin: who asserts this data is an answer?
- Correctness: is this an answer to my question?

Content Based Security

- In CCN, the content itself (rather than its path) is protected
- One can retrieve the content from the closest source and validate it
- All content is digitally signed
- Signed info includes hash of the public key used for signing
- We still need some kind of a Public Key Infrastructure (PKI)

DoS Resistance

Many current DoS + DDoS attacks/threats become irrelevant because of NDN architecture

- Content caching:
  - Multiple interests for same content are collapsed
  - One copy of content per “interested” interface is returned
  - Content not forwarded w/out prior state set up by interests
  - Stateful routing helps to fight/push back attacks

Some (new) attack opportunities may be possible, but it is much more resistant to DoS attacks than what we have today.

Privacy Benefit

- Interests lack “source address”
  - Data can be routed back without knowing consumer identity/position
- One interest may correspond to multiple consumers
- Caches reduce effectiveness of observers close to producers

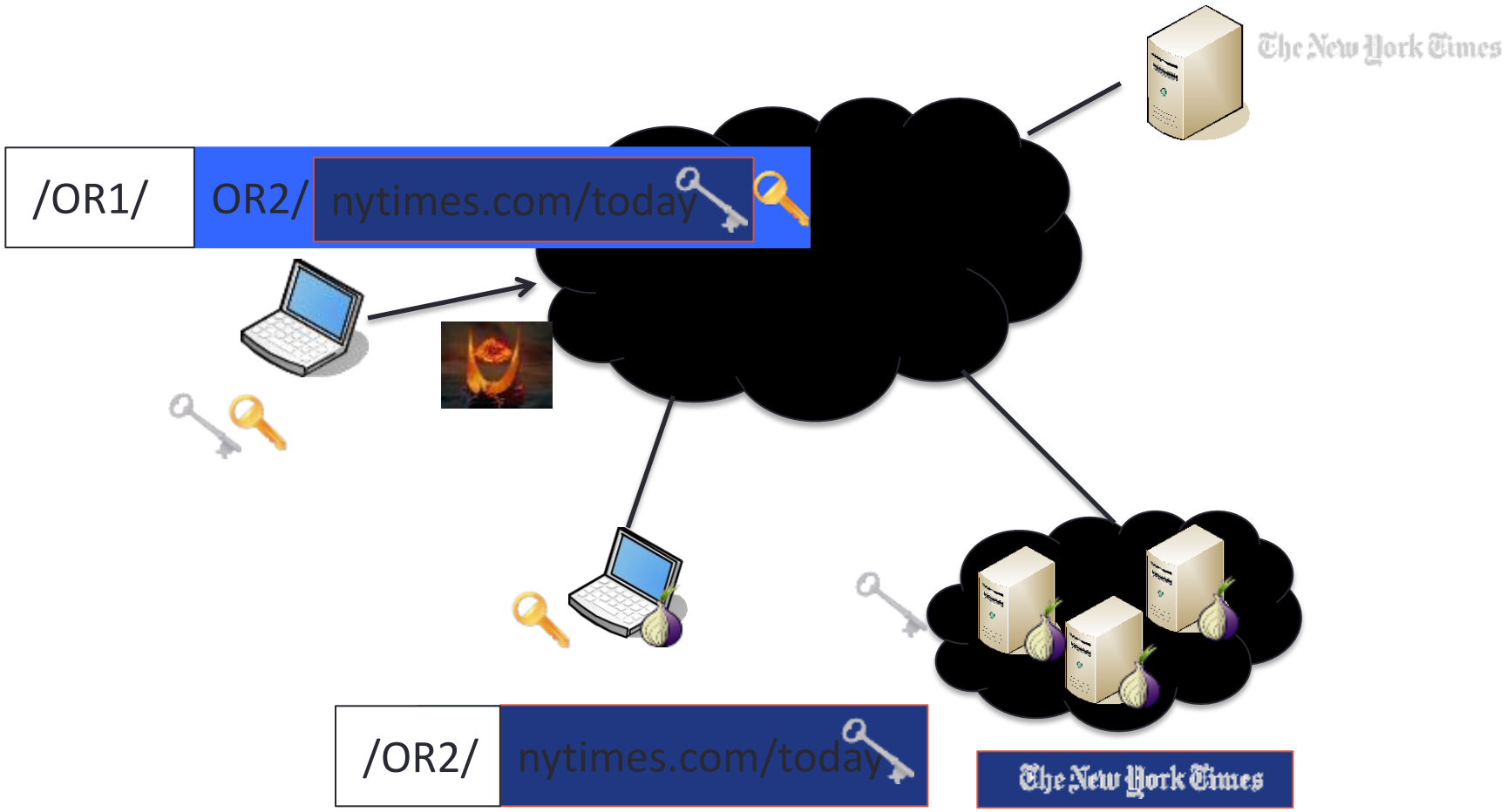
Privacy Challenges in NDN

- *Name Privacy:* semantically related names
  - Interested in “/healthonline/STDs/..”
- *Content Privacy:* unencrypted public content.
  - Retrieved content is an “.mp3” file
- *Signature Privacy:* leaked signer(publisher) identity
  - Retrieved content is signed by “match.com”
- *Cache privacy:* detectable cache hits/misses
  - Interests from this user usually misses caches.
- Government censorship?
  - More intuitive names tied with the location

Named Data Onion Routing (NDSS ‘12)

- Consists of client and anonymizing router (AR) software
- Layers of encrypted Interests reside inside the name component of interests
  - E.g.,: */anonymizer/Enc(Timestamp* || *key* || *Interest)*
- Content is encrypted with the client-provided key on its way back
  - Encapsulation is published under the requested name and signed by ARs.

Example

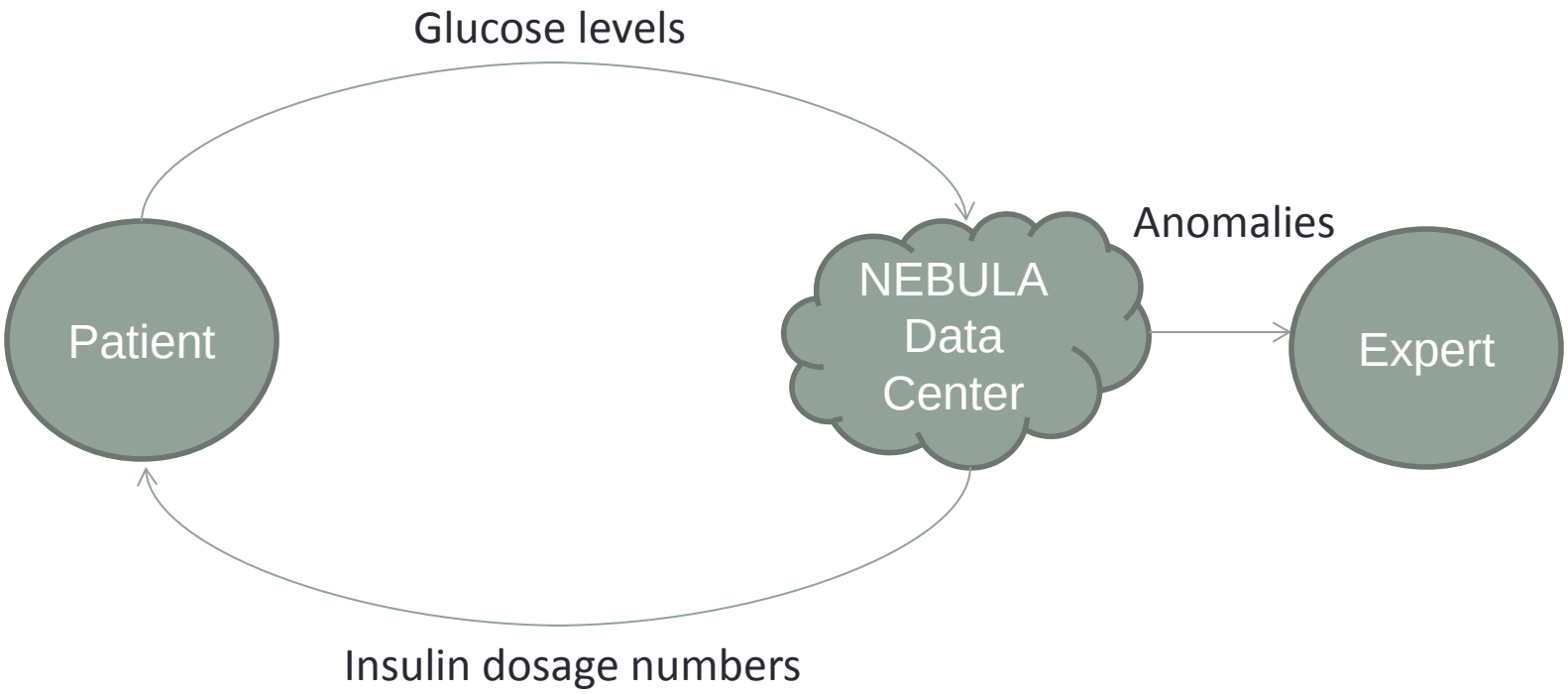

The New York Times
/OR1/
OR2/
nytimes.com/today
/OR2/
nytimes.com/today
The New York Times

Discussions

- DDoS: What if senders and receivers collude?
- Deploying it incrementally?
  - Edge router able to translate between CCN and traditional internet?
  - Controller might be responsible in the case of SDIA
- Could you put up a Content router that doesn't play by the rules?
- Could you insert yourself in the middle of a CCN network?
- If you could “own” a CCN element, would you be able to launch attacks on availability? Or integrity and confidentiality?

NEBULA

- Motivation
  - Realize full potential of the cloud
  - Example: cloud-based healthcare application

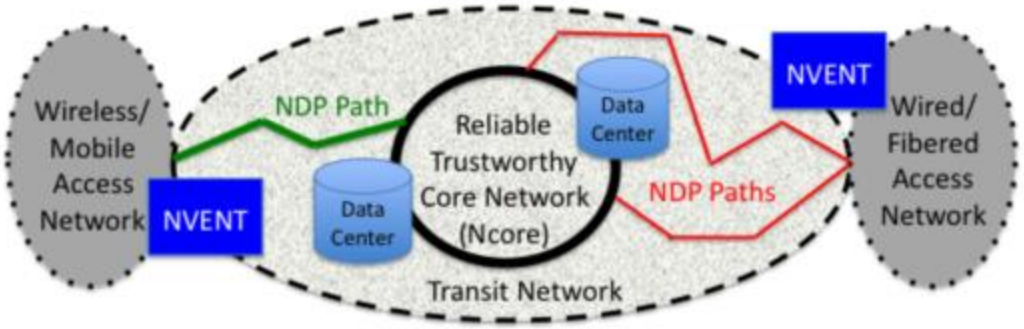


NEBULA

- Issues with application
  - High availability
  - Access independent of location
  - Security of paths
- Technical challenges
  - Specifying stakeholder policies
  - Enforcing stakeholder policies
  - Flexibility and extensibility of technologies

NEBULA

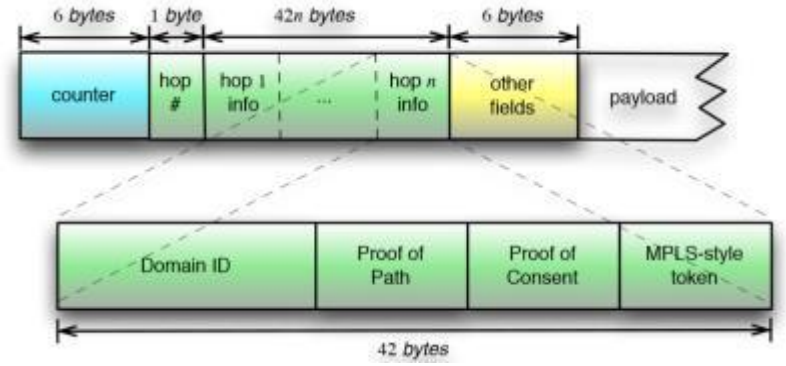
- Design
  - NDP
    - Data plane
    - Policy enforcement
  - NVENT
    - Control plane
    - Service access
    - Policy specification
    - Path discovery
  - NCore
    - High performance and availability core routers
    - Interconnect data centers redundantly



Source: http://www.nets-fia.net/Meetings/Nov10/Kickoff-public-Nebula.pdf

NEBULA

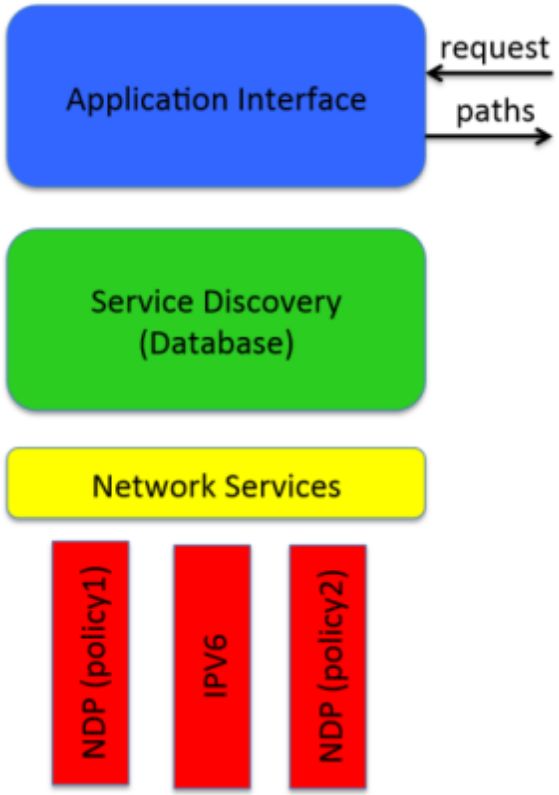
- NDP
  - Was the route *authorized*?
    - Check proof of consent
  - Was the route *followed*?
    - Check proof of path
  - What actions to take?
    - Check MPLS-style token



Source: http://www.nets-fia.net/Meetings/Nov10/Kickoff-public-Nebula.pdf

NEBULA

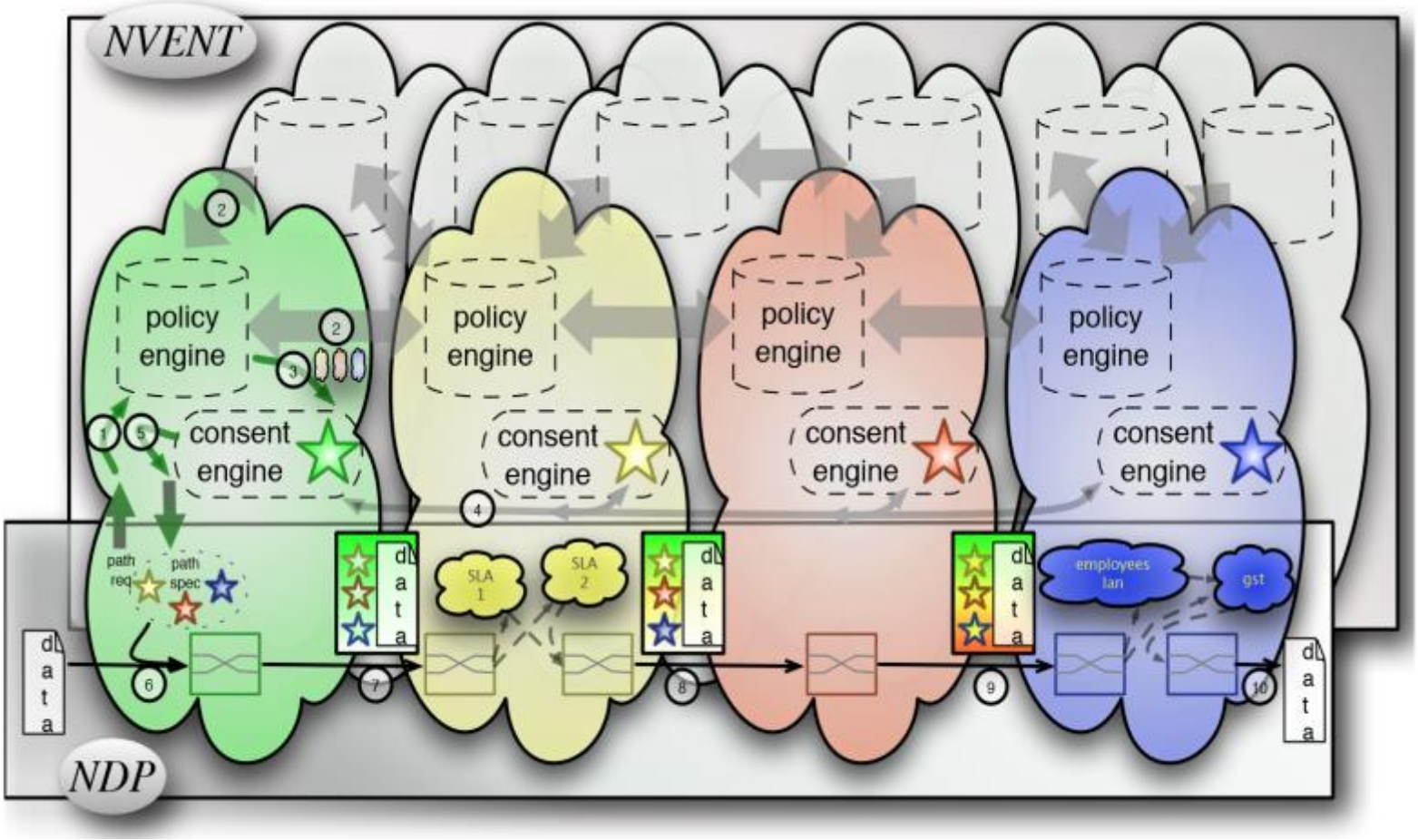
- NVENT
  - “Service-centric” interface
  - Path discovery
  - Multipath routing
  - Dynamic path construction



Source: http://www.nets-fia.net/Meetings/Nov10/Kickoff-public-Nebula.pdf

NEBULA

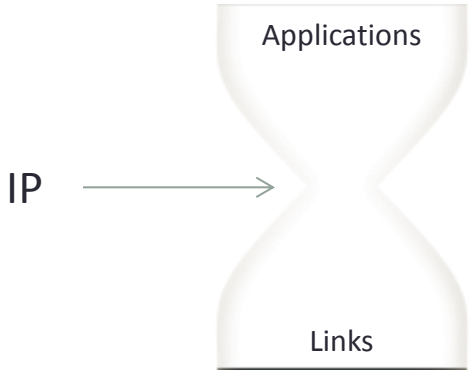
- Interface



Source: http://www.nets-fia.net/Meetings/Nov10/Kickoff-public-Nebula.pdf

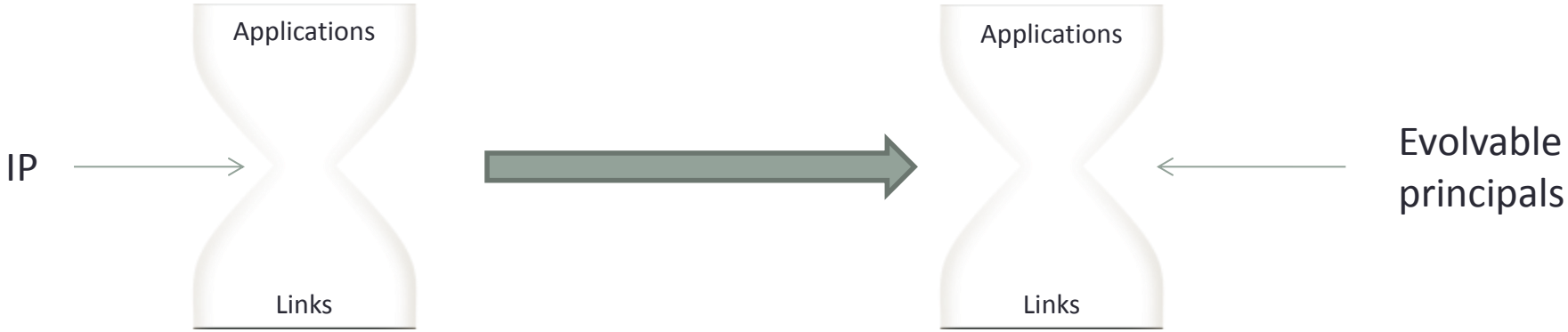
eXpressive Internet Architecture (XIA)

- Motivation
  - IP as a narrow waist
  - What should the internet be centered on?
    - Content? [NDN]
    - Mobility? [MobilityFirst]
    - Cloud? [NEBULA]
  - How will this evolve?
  - Trust in the network



eXpressive Internet Architecture (XIA)

- Key ideas
  - *Principals*
    - Hosts, services, contents, etc.
  - *Intrinsic security*
    - “Self-certifying identifiers”
  - Fallback
    - Incremental deployment

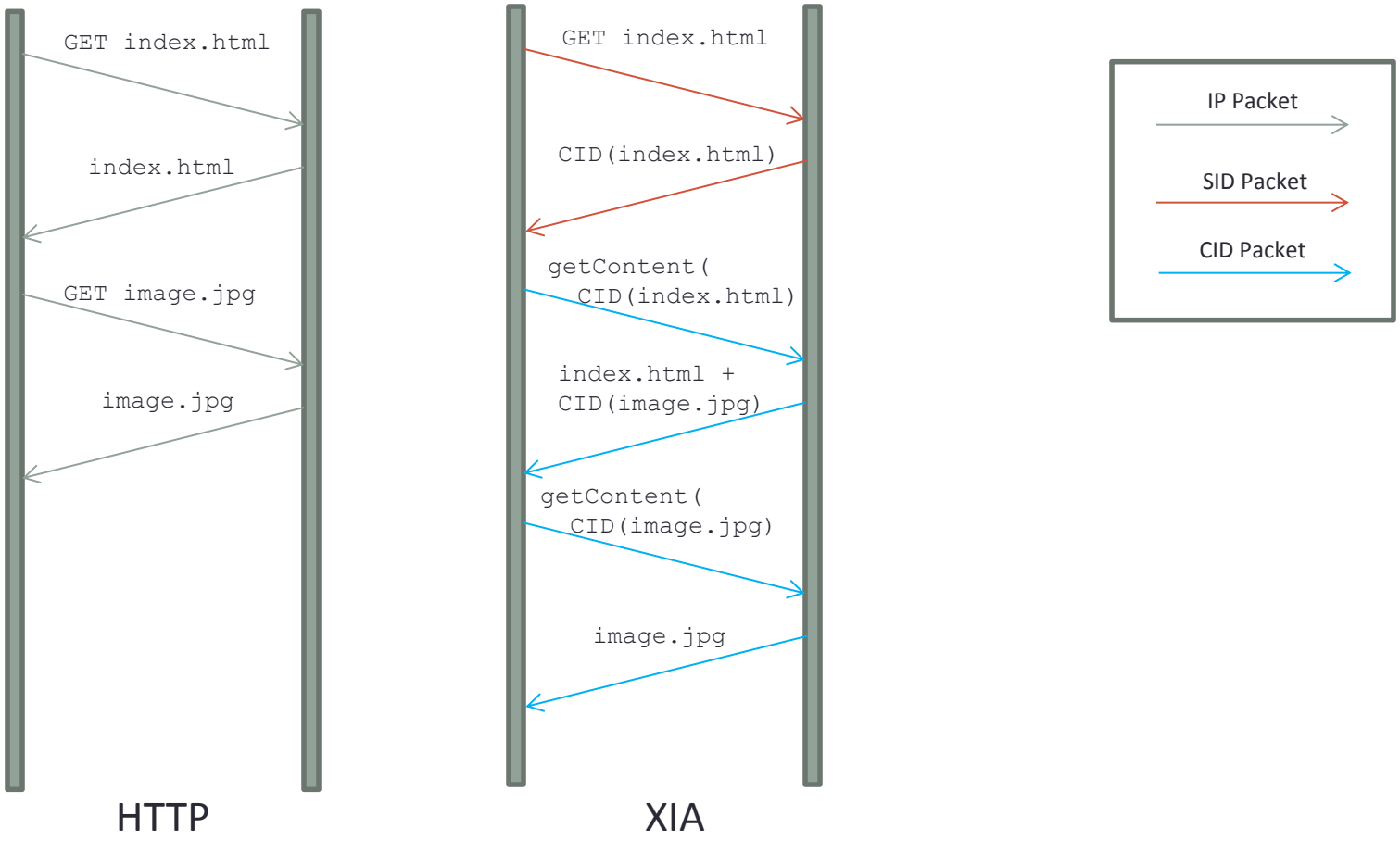


eXpressive Internet Architecture (XIA)

- Principal-specific support
  - Communication semantics
    - Networks and hosts: `bind(), recv(), send()`
    - Content: `getContent(), putContent()`
  - XID
  - Allocation and mapping of XIDs to security properties
    - Networks and hosts: HID := `hash(publicKey)`
    - Content: CID := `SHA1SUM(content)`
  - Per-hop processing and routing
    - Networks and hosts: hierarchical routing based on ADs
    - Content: “shortcut routing”

eXpressive Internet Architecture (XIA)

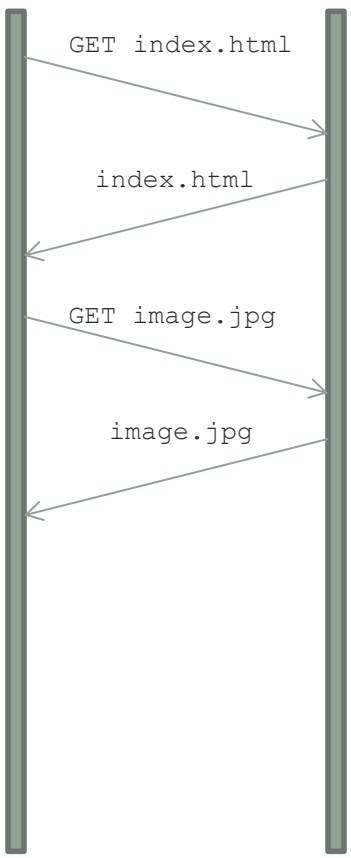
- Example with multiple principals



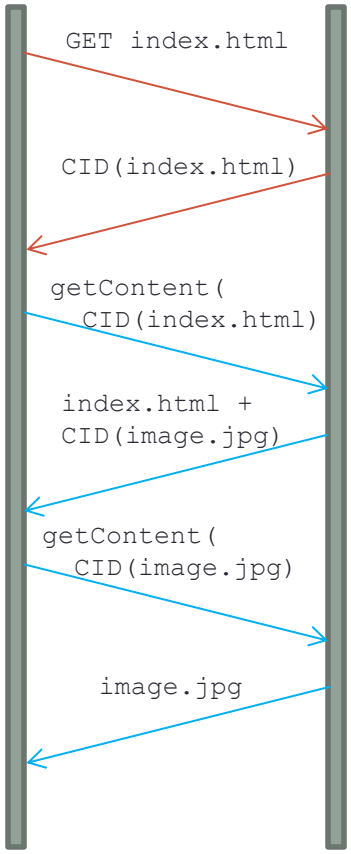
Picture adapted from Anand et. *al*, "XIA: An Architecture for an Evolvable and Trustworthy Internet" Tech Report

eXpressive Internet Architecture (XIA)

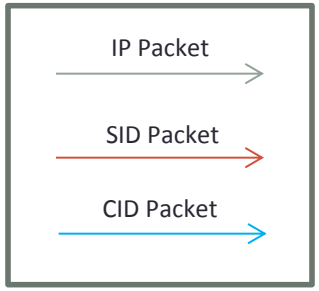
- Example with multiple principals



HTTP



XIA



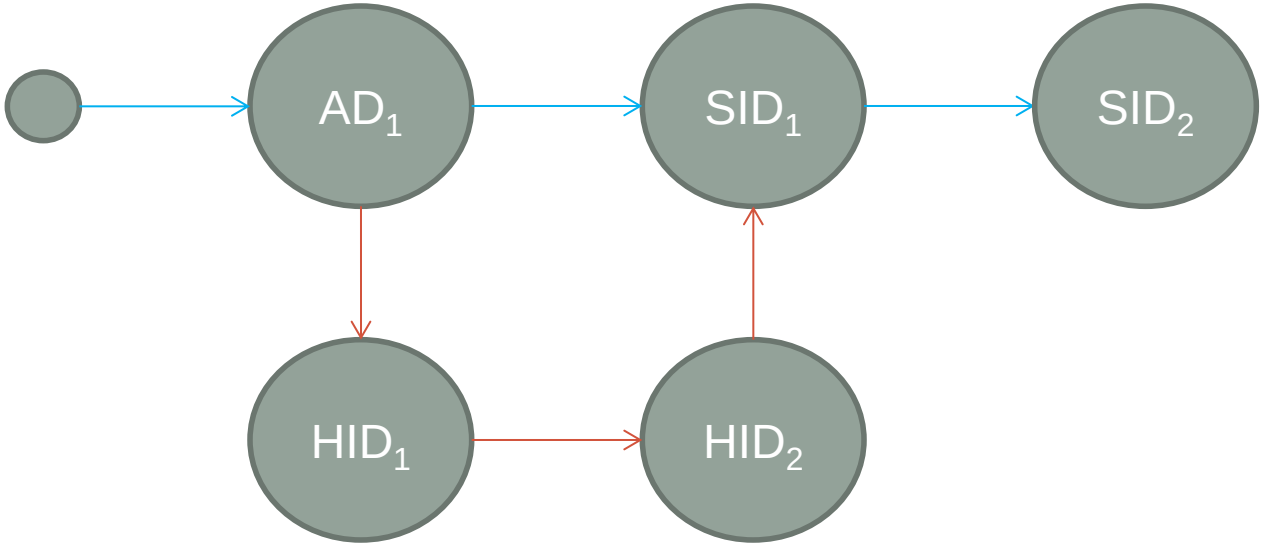
Intrinsic security!

- `hash(index.html) = CID(index.html)`
- `hash(image.jpg) = CID(image.jpg)`

Picture adapted from Anand et. *al*, "XIA: An Architecture for an Evolvable and Trustworthy Internet" Tech Report

eXpressive Internet Architecture (XIA)

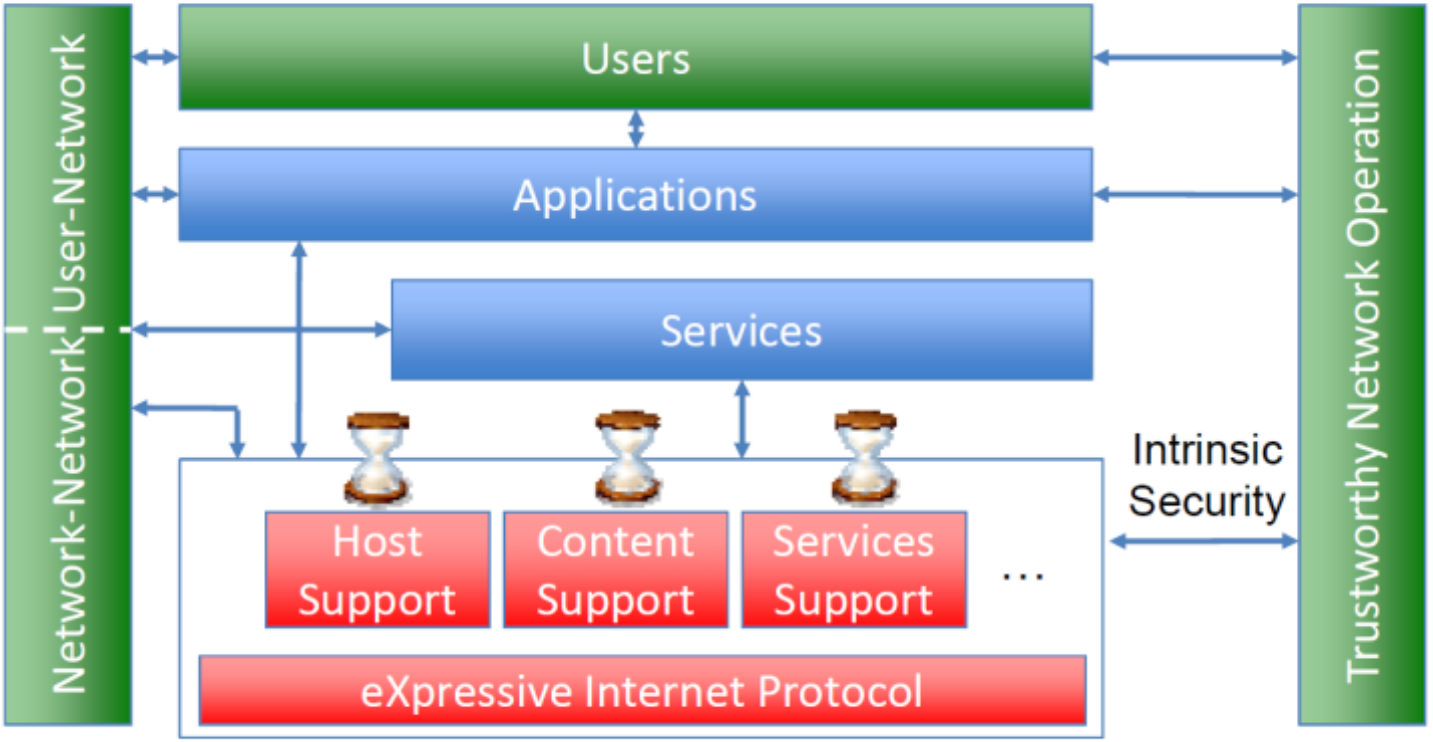
- Issues
  - Handling fallback
  - Routing
- Solution: DAG-based addressing!



  - Very flexible, e.g. shortcut routing, binding, source routing, etc.

eXpressive Internet Architecture (XIA)

- Architecture



Source: http://www.cs.cmu.edu/afs/cs/project/xia/www/Documents/XIA-Stanford.pdf

A Layered Naming Architecture for the internet

- Improving *naming* aspect of the architecture
- Motivation
  - DNS names and IP addresses rigid, tied to pre-existing structure
  - Lack of mechanism for directly naming data and services. Treat services and data as first class Internet objects
  - Integrate middleboxes into the Internet architecture
- Three Levels of name resolution
  - User level descriptors to service identifiers
  - Service identifiers to end point identifiers
  - End point identifiers to IP addresses

Basic design principles

- Names should bind protocols only to the relevant aspects of the underlying structure
  - Applications forced to resolve service and data names down to IP address
  - Service Identifiers(SID) - host independent services and data name
  - Endpoint Identifiers(EID) - Uniquely identifies a host
  - Two additional layers of name resolution
    - SIDs to EIDs
    - EIDs to IP address

Basic design principles(cont'd)

- Names should not impose arbitrary restrictions on the elements to which they refer
  - Movement and replication of services/data causes existing references to become invalid
- Flat namespace
  - Flat namespace scheme for SIDs and EIDs
  - Flat namespaces lack inherent structure
  - Enable flexible migration
  - Flat namespaces can be implemented via DHT

Basic design principles(cont'd)

- A network entity should be able to direct resolutions of its name not only to its own location, but also to locations or names of chosen delegates
  - Incorporate intermediaries(middleboxes)
  - Enables architecturally sound intermediaries
  - Some protection against DoS attacks

Network Capabilities: The Good, the Bad and the Ugly

- Problem: DoS attack
  - Any host can be flooded with unwanted traffic
  - Any host can be flooded with unwanted traffic
  - Unauthorized traffic
  - Congestion blocks legitimate TCP connections
  - No built-in authentication

Possible Approach

- Connection-oriented
  - Receiver blocks senders by default
  - Allows senders only after authenticating and establishing network layer connection
  - Capabilities
- Datagram
  - Receivers allows access by default
  - Explicitly denies malicious senders by propagating filters to the network

Capabilities: Connection-oriented approach

- Overview
  - Categorize traffic into good and bad.Prioritize good
  - Good traffic - belongs to established network-layer connections
  - Bad traffic - Failed to obtain authorization
  - Issues: Connection-setup requests vulnerable to DoS i.e."Denial Of Connection"

Capabilities(cont'd)

- Overview
  - Capability Request
    - Receiver and intermediate routers stamp(tokenize) packets from senders
  - Data Transmission
    - Sender includes the capability in all packets sent to the receiver
    - Intermediate routers verify their part of the capability
  - Capabilities have expiration timestamps
  - Keeps verification points stateless. No need for traditional packet filters, inter-ISP relations

Denial Of Capability

- DoS attack on capability requests
  - DoS can start before legitimate client gets capability
  - Old legitimate clients can still connect and send packets.
  - Requests are unprotected
- Datagram solution
  - Capability requests are nothing but datagrams
  - Removes need for capability/connection oriented approach

Datagram Solutions - First Attempt

- Fair-queuing of capability requests
  - Request datagrams are few compared to general datagram traffic
  - No sender can forward more than its share of requests

- Issues
  - Multiple attackers
  - Send requests at same rate at legitimate clients

Datagram Approach

- Basic components of the right solution have already been identified

- Basic components
  - Unforgeable path information inside each packet
  - Propagation of filtering rules
  - Scalable and secure distribution of filtering state

Unforgeable path

- Must verify packet origin address
- Internet allows fake source IP addresses

- Solution: Packet Marking
  - Subset of upgraded routers stamp packets
  - Receiver uses stamps to verify path and source

Propagation of filtering rules into the network

- Stop attack before it reaches bottleneck
  - Receiver classifies source as malicious. Creates filtering rules.
  - Must be propagated to intermediate routers
  - n malicious senders means n filtering rules

- Issue
  - No ISP can support every filtering rule of each client

Scalable distribution of filtering state

- Push filtering state close to attack sources
- Secure propagation of filtering state across ISPs
- AITF - pushes filtering state as close to attack source

AITF summary

- Victim sends filtering request to its network gateway
- Network gateway ***temporarily*** blocks the undesired traffic
- Network gateway propagates request to attacker's gateway
- Attacker's gateway propagates request to attacker
- Disagreement leads to an iterative procedure till attack stops

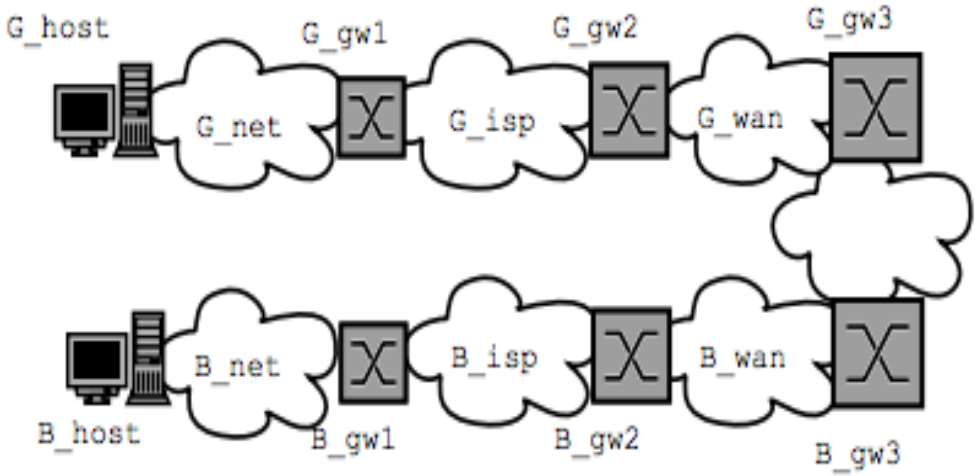


Fig. 1

EXAMPLE ATTACK PATH FROM ATTACKER *B_host* TO VICTIM *G_host*

AITF summary (cont'd)

- False filtering request
  - 3 way handshake to authenticate request
  - Assumptions
    - No node in the path has been compromised
    - Attacker cannot intercept path

AITF summary (cont'd)

- False filtering request
  - 3 way handshake to authenticate request
  - Assumptions
    - No node in the path has been compromised
    - Attacker cannot intercept path

FIA Discussion

- Clean state vs. evolutionary
  - Clean state: No restriction/assumption on architectural design, no burden of incremental design
  - Evolutionary: Backward compatible, incremental development
- Integration of key research areas into one architecture
  - Most projects emphasize/deal with a particular problem or a set of particular problems
  - Need to integrate different requirements and resulting architectures
- Interface among stakeholders
  - Multiple stakeholders – users, ISPs, data providers
  - Provide flexible and explicit interfaces to allow interaction, enforce policies and laws

Papers Referenced

- Networking Named Content, CoNEXT ‘09
- Software-Defined Internet Architecture, HotNets ‘12
- Privacy Risks in Named Data Networking: What is the Cost of Performance? SIGCOMM CCR ’12
- ANDaNA: Anonymous Named Data Networking Application, NDSS ’12
- NEBULA - A Future Internet That Supports Trustworthy Cloud Computing (Whitepaper)
- XIA: An Architecture for an Evolvable and Trustworthy Internet (Tech Report)
- Active Internet Traffic Filtering: Real Time Response to Denial-of-Service attacks. USENIX 2005
- A Layered Naming Architecture for the Internet. SIGCOMM 2004

Thank you