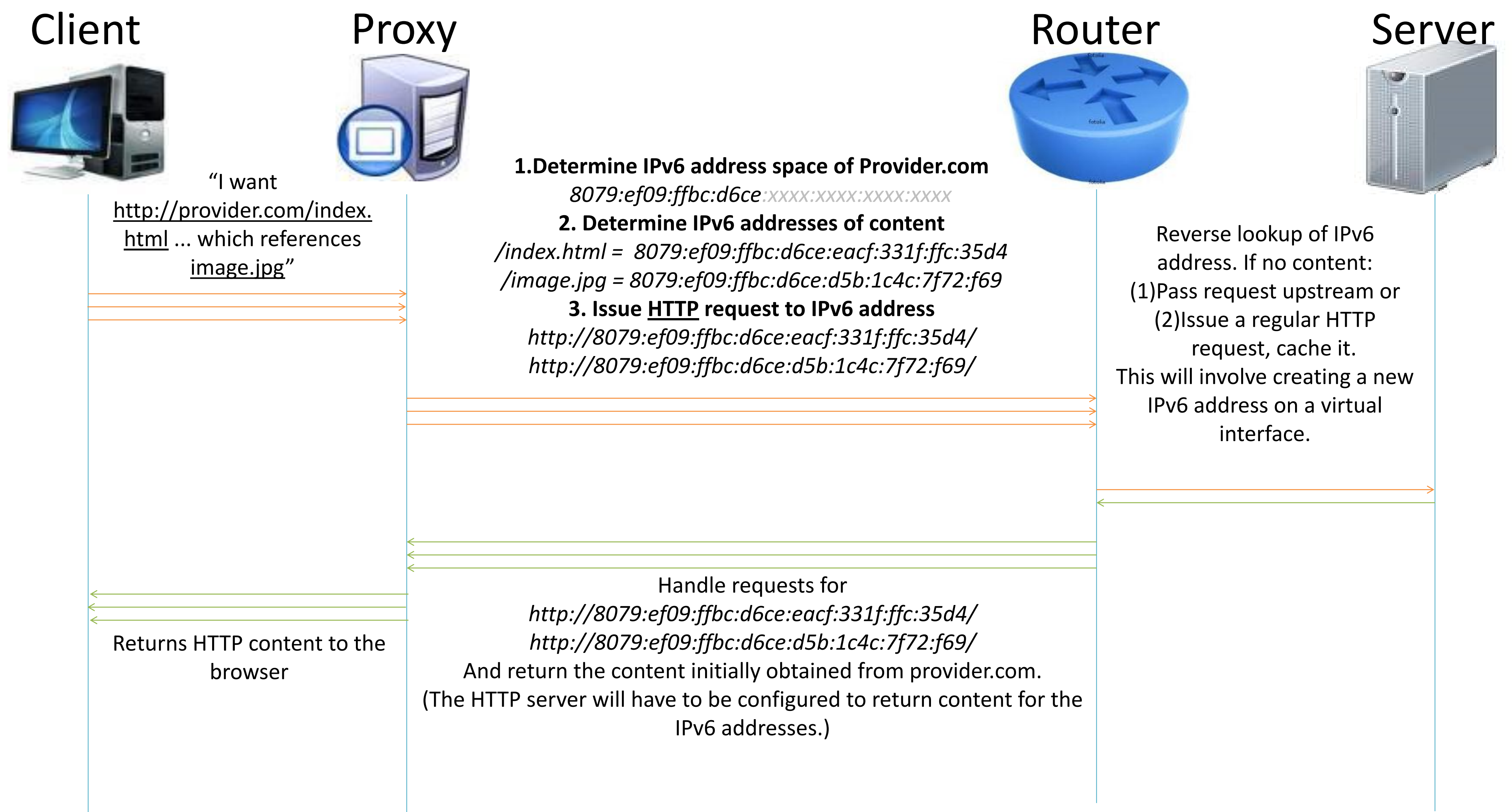


# IPv6 Addresses as Content Names in Information-Centric Networking

Suman Srinivasan, Henning Schulzrinne

Columbia University

{sumans, hgs}@cs.columbia.edu



The architecture diagram of our IPv6 content addressing system. In our system, the regular browser makes a HTTP request through a proxy, which translates HTTP requests to an IPv6 content addressing system. The request is sent out over the network, until a router on path that has the content responds to the request. The proxy then translates the retrieved content back into a HTTP response to the user's browser.

## IPv6 Features That Are Useful for Content Networking

- IPSec: security
- Multicasting: streaming video
- No packet fragmentation
- Better mobile support
- Jumbograms
- IPv6-over-IPv4 bridging mechanisms: use of our IPv6 content naming proposal in archaic or today's networks

## Sample Name to IPv6 Mappings

- **8079:1b37:2650:3af8:1d78:a723:dee0:2522**  
`http://TheEpochTimes.com/content/video.mp4`
- **8079:1b37:2650:3af8:eacf:331f:ffc:35d4**  
`http://TheEpochTimes.com/index.html`

## Currently implemented

- Content address registry as a web service (built in PHP) connected to a MySQL database.
- Requests to set/get content names and their corresponding IPv6 address mapping are done through simple put and get requests. We do plan to make this more scalable and hierarchical in the future.

## Current/Future Work

- Starting work on the full implementation of the IPv6 content naming architecture.
- Use netfilter, particularly libnetfilter\_queue and its Python language bindings, to handle and serve IPv6 content naming and addresses.