

# Transition and Tunnels



Engineering Workshops

# Transition

- There are really two types of cases that need to be addressed.
  - Network layer
    - How can we get v6/v4 packets across v4/v6 networks?
  - Host layer
    - How can a v6/v4 host access content on a v4/v6 host?

# Network layer transition

- Tunnels
- Dual Stack

# Tunnels

- Information from one protocol is encapsulated inside the frame of another protocol.
  - This enables the original data to be carried over a second non-native architecture.
- 3 steps in creating a tunnel
  - Encapsulation
  - Decapsulation
  - Management

# Tunnels

- There are at least 4 tunnel configurations:
  - Router to router
  - Host to router
  - Host to host
  - Router to host
- How the addresses are known determines the type of tunnel.
  - Configured tunnel
  - Automatic tunnel

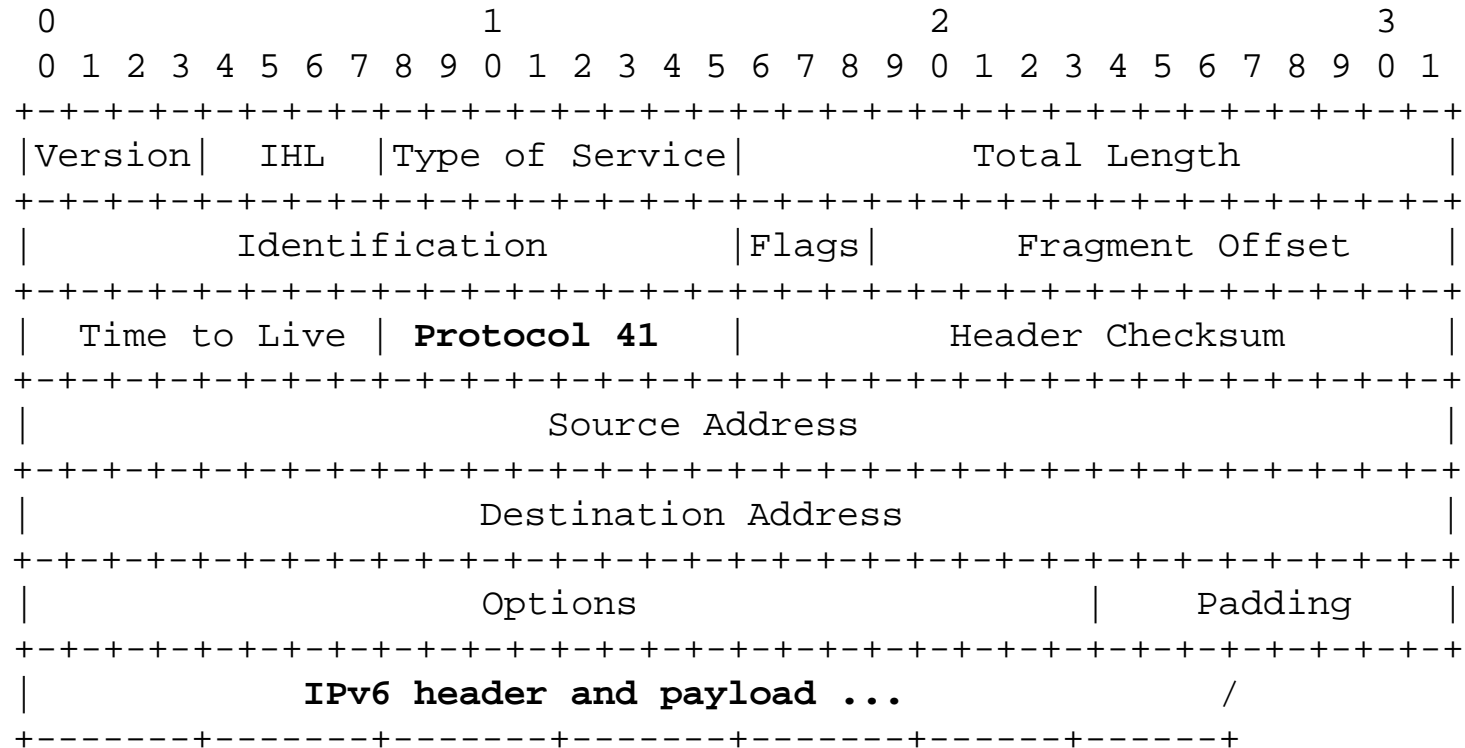
# Configured Tunnels

- Typically, configured tunnels connect IPv4/IPv6 dual-stack hosts or networks across IPv4-only networks to other dual-stack networks.
- Local network administrators arrange for a tunnel between IPv6 networks across IPv4-only networks.
- This was default dual-stack architecture on Abilene until 2002; there are still some configured tunnels supported by the Abilene NOC.

# Automatic IPv6-in-IPv4 tunnel

- A dual-stack host or network automatically creates a tunnel across an IPv4-only network
- Tunnel Types
  - 6to4: Most commonly deployed automatic tunnel format. Available with Windows XP
  - ISATAP: “Intranet” automatic tunnel format; not designed for public networks
  - Teredo: Promising, but still a work in progress. Designed to traverse NATs

# 6to4 Tunnel IPv4 Packet Format



Source: RFC 3056, Connection of IPv6 Domains via IPv4 Clouds

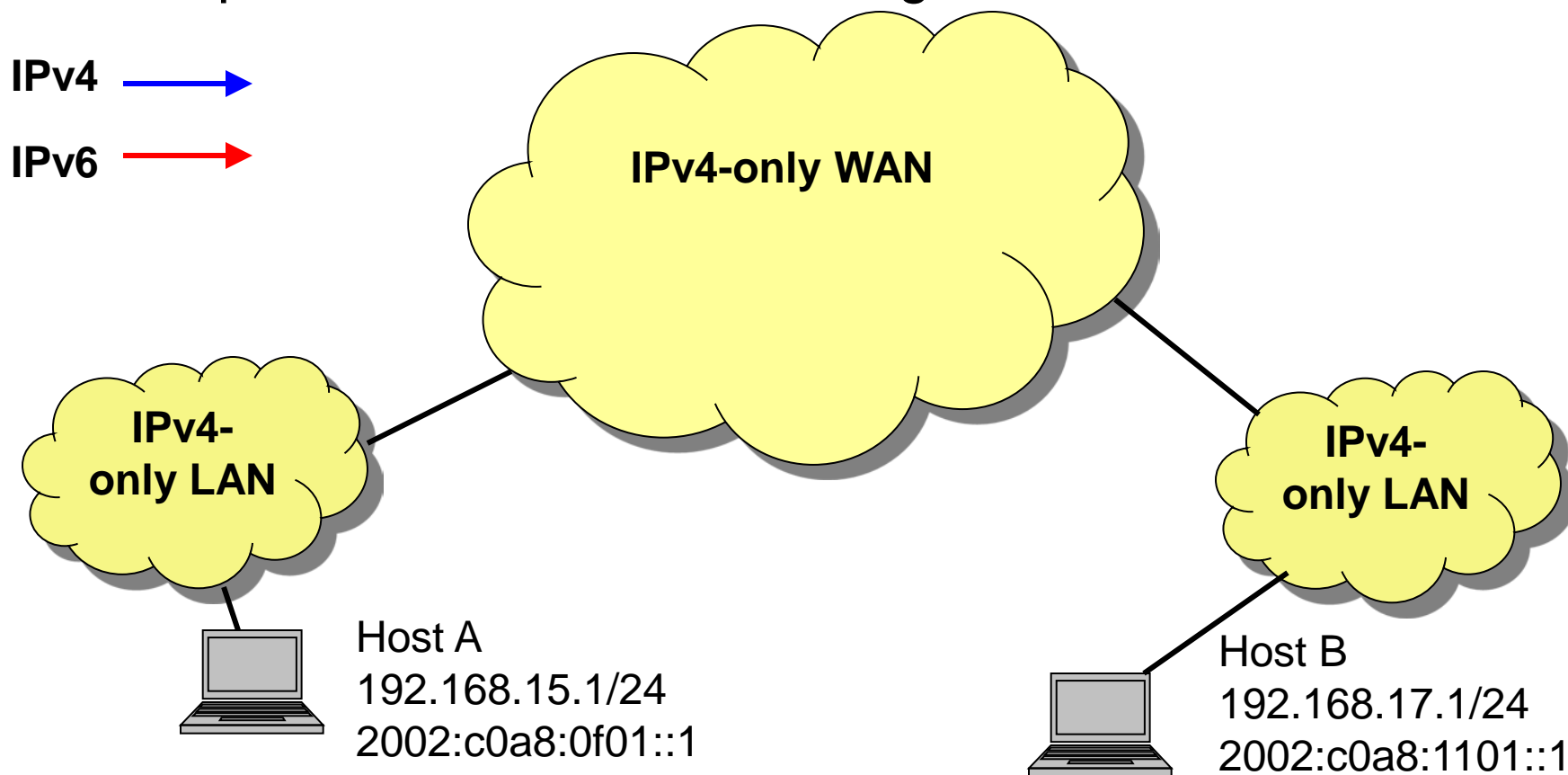
# IPv6 Address Format in 6to4

Bits 0-16	17-48	49-64	65-128
2002	IPv4 Address	SLA ID	Interface ID

For example, a Windows XP system with IPv4 address 207.75.164.119 could have a 6to4 IPv6 address of 2002:cf4b:a477::1

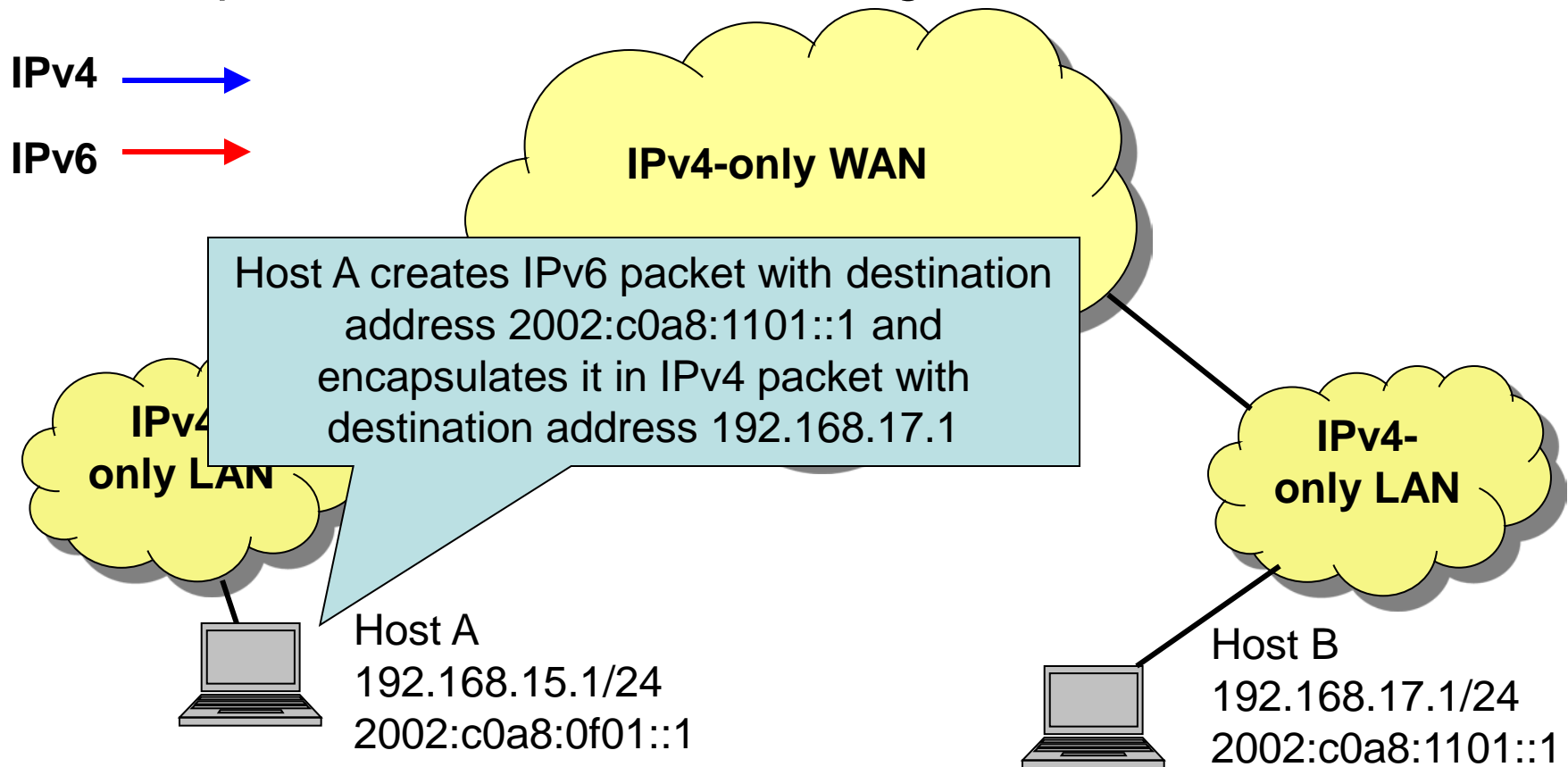
# 6to4 Implementation Scenarios (1 of 2)

Both host A and host B are on IPv4-only networks and both are capable of IPv6 6to4 tunneling



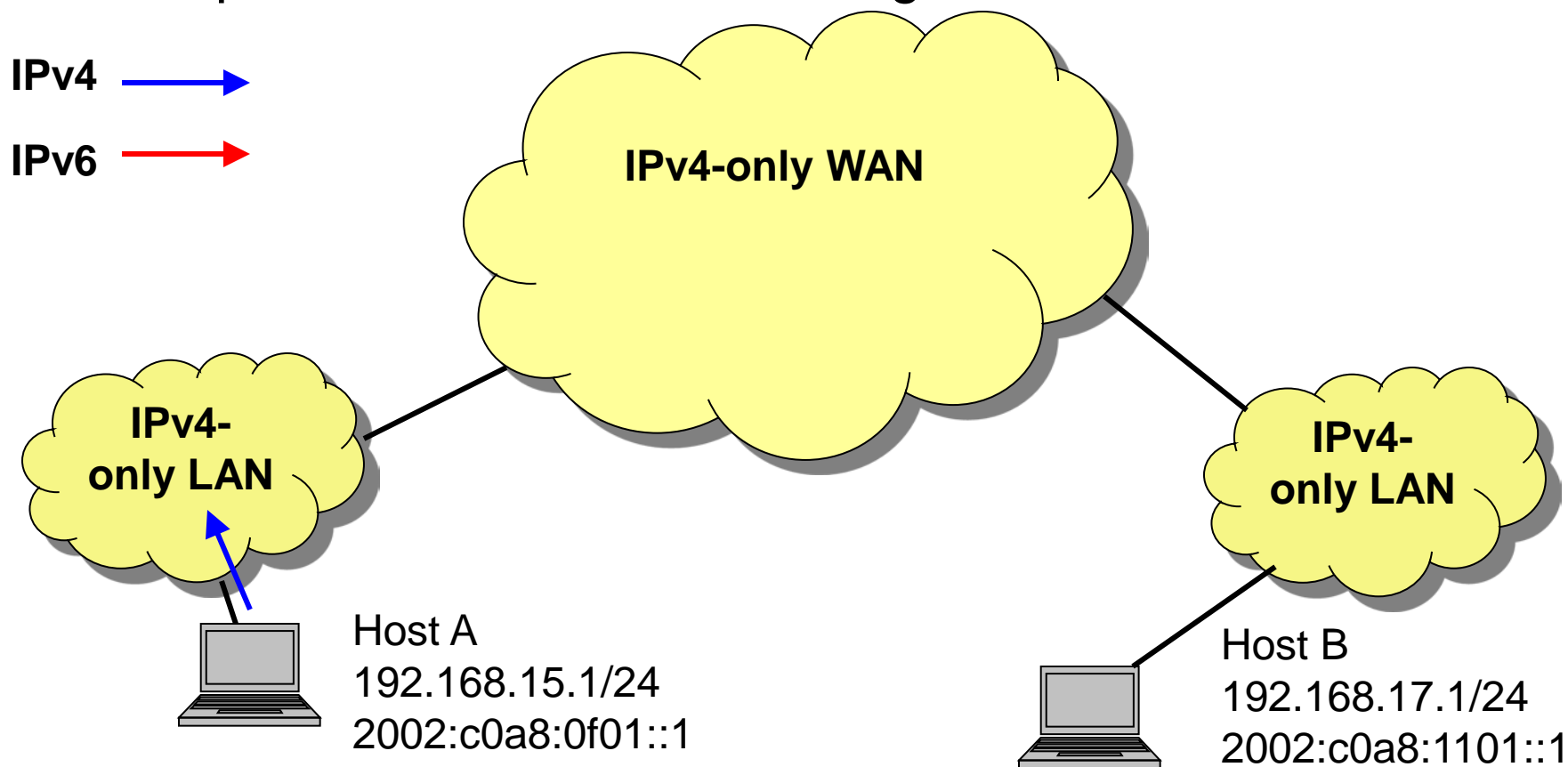
# 6to4 Implementation Scenarios (1 of 2)

Both host A and host B are on IPv4-only networks and both are capable of IPv6 6to4 tunneling



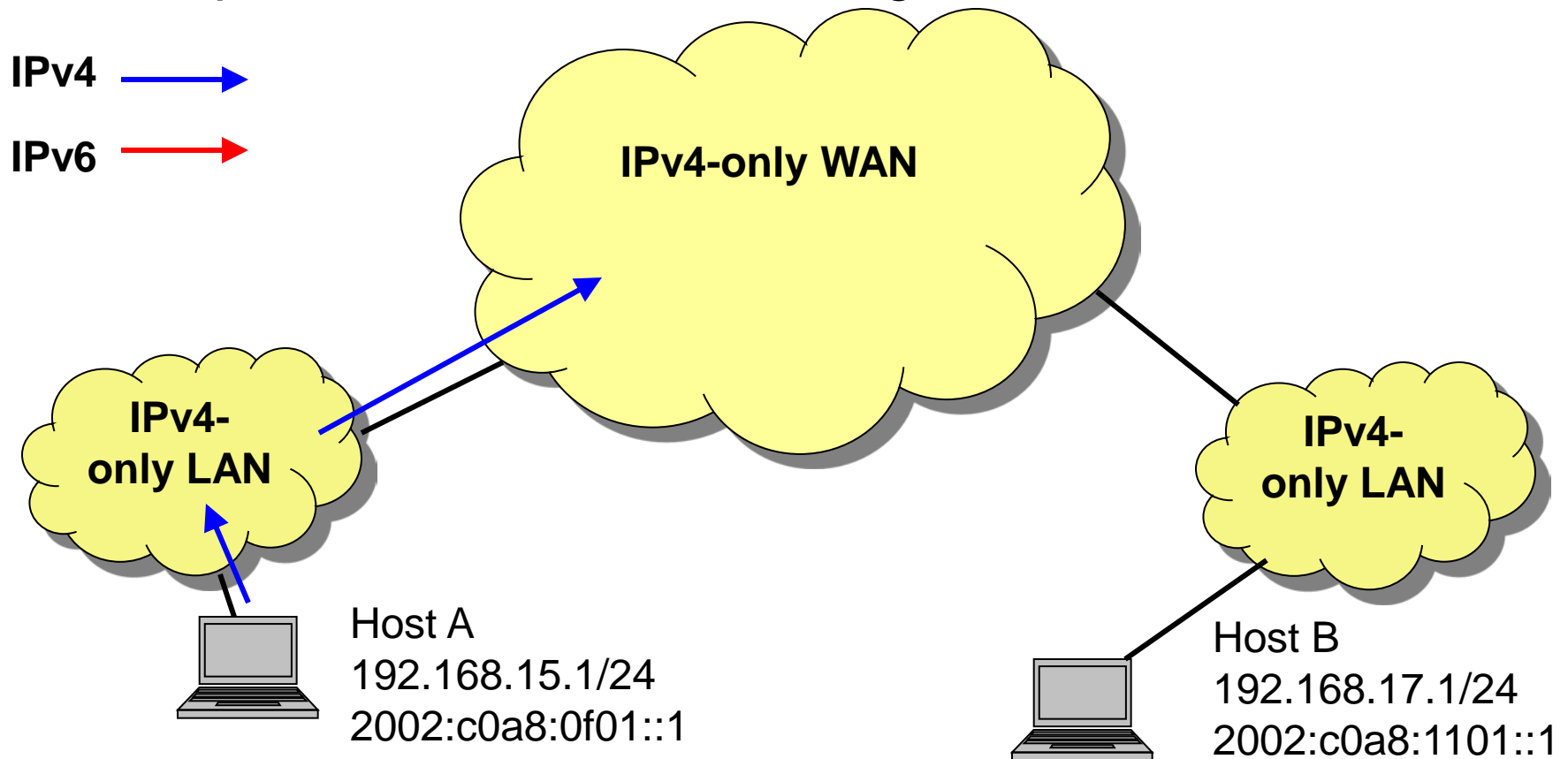
# 6to4 Implementation Scenarios (1 of 2)

Both host A and host B are on IPv4-only networks and both are capable of IPv6 6to4 tunneling



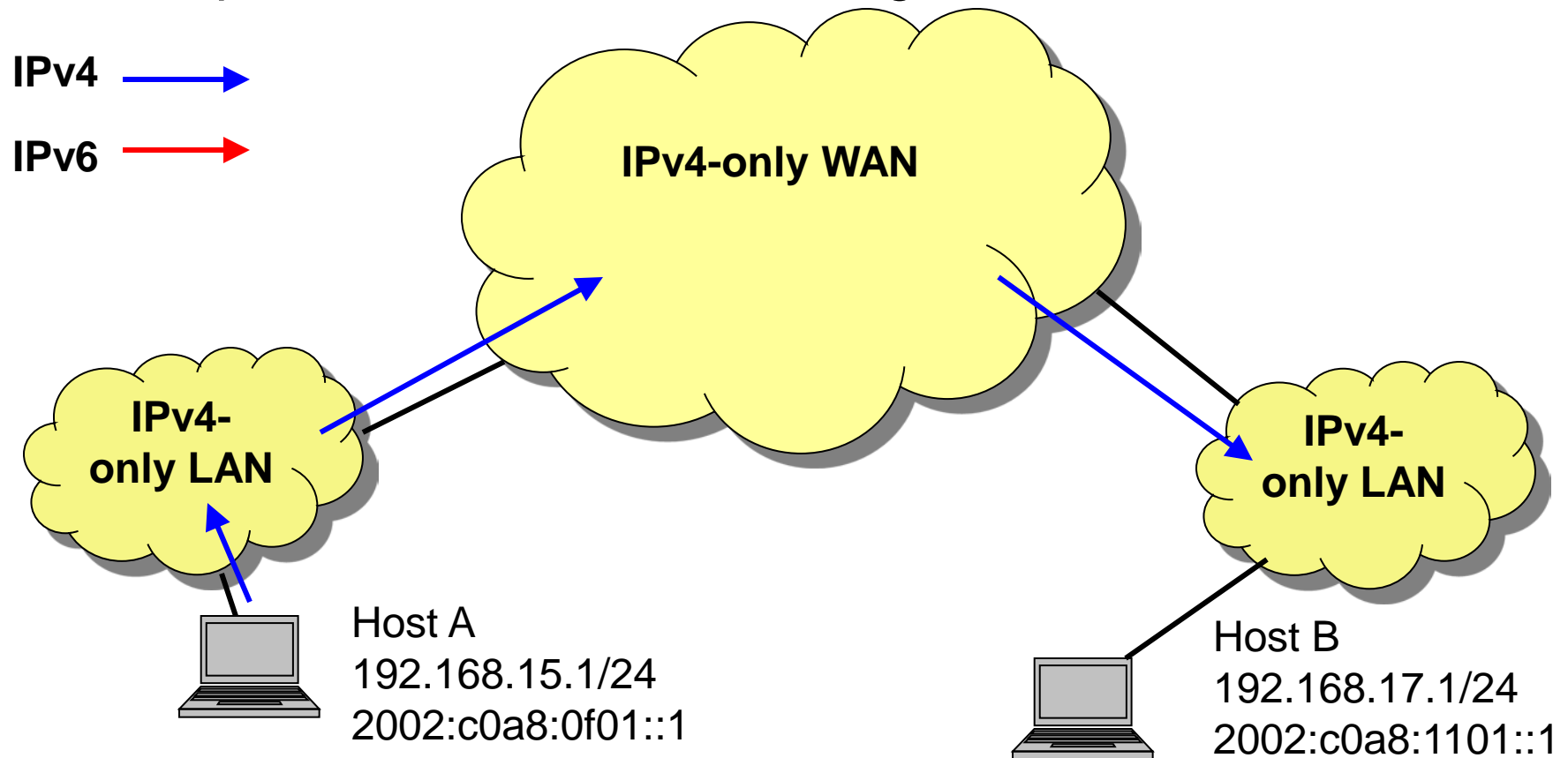
# 6to4 Implementation Scenarios (1 of 2)

Both host A and host B are on IPv4-only networks and both are capable of IPv6 6to4 tunneling



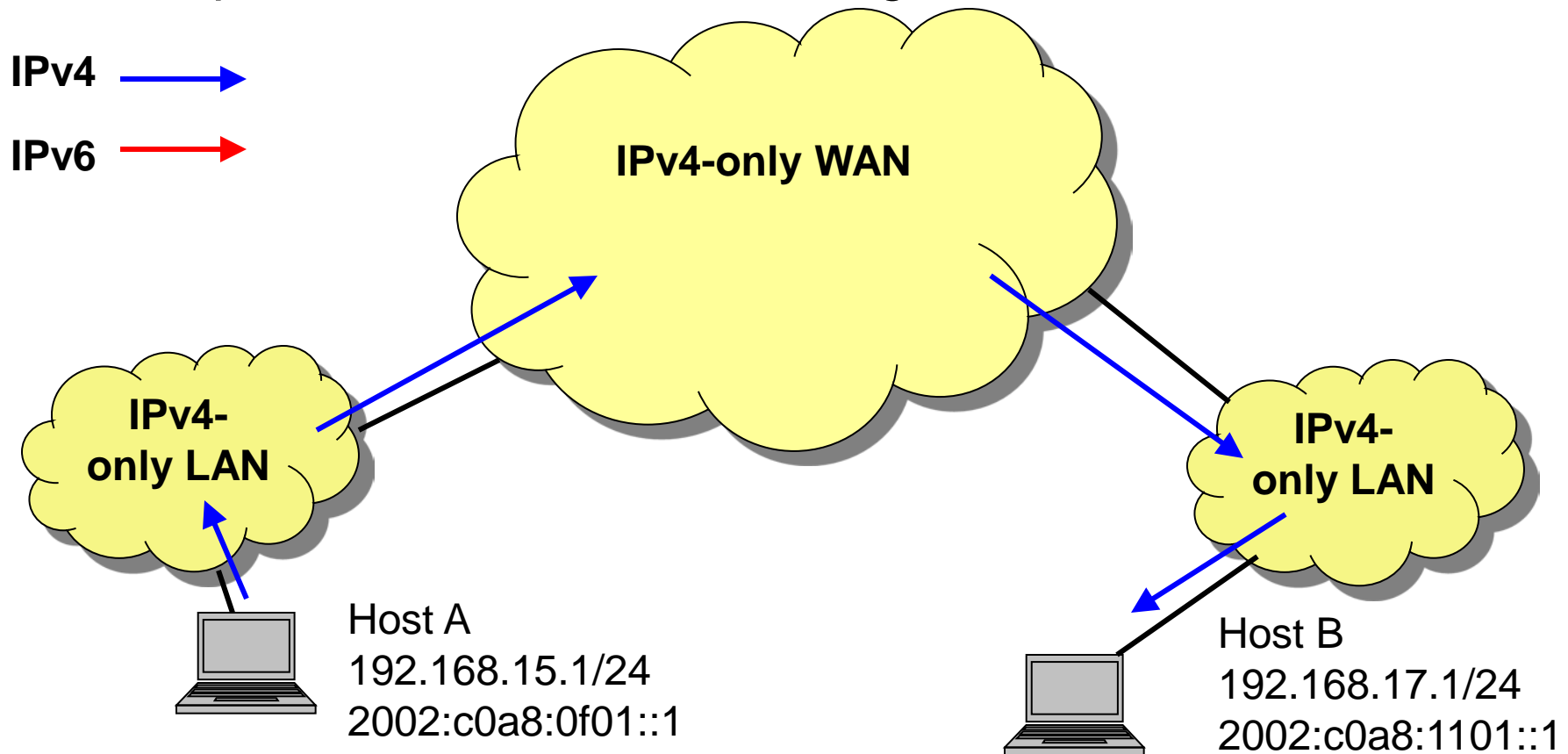
# 6to4 Implementation Scenarios (1 of 2)

Both host A and host B are on IPv4-only networks and both are capable of IPv6 6to4 tunneling



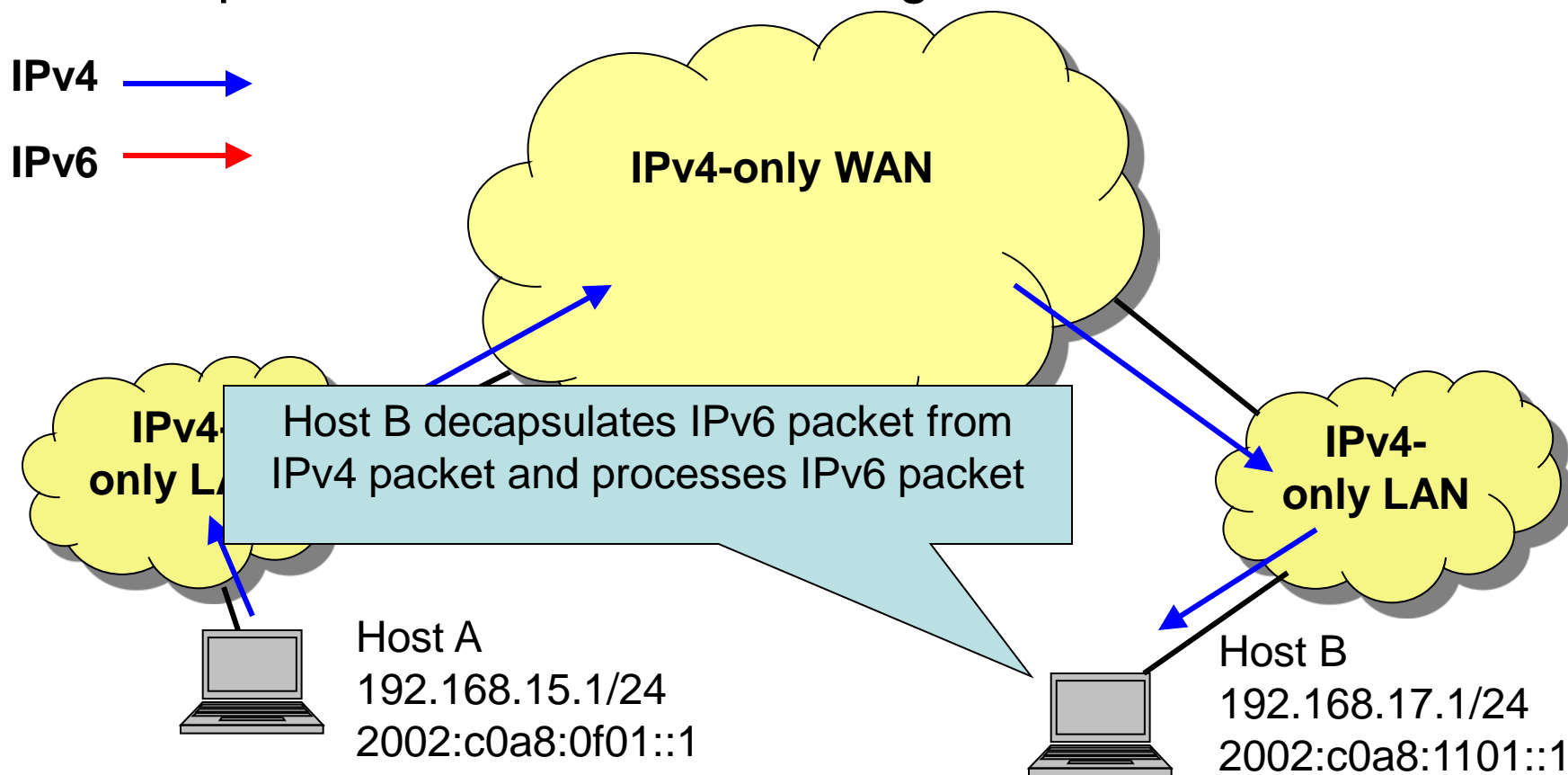
# 6to4 Implementation Scenarios (1 of 2)

Both host A and host B are on IPv4-only networks and both are capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (1 of 2)

Both host A and host B are on IPv4-only networks and both are capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (1 of 2)

## Observations

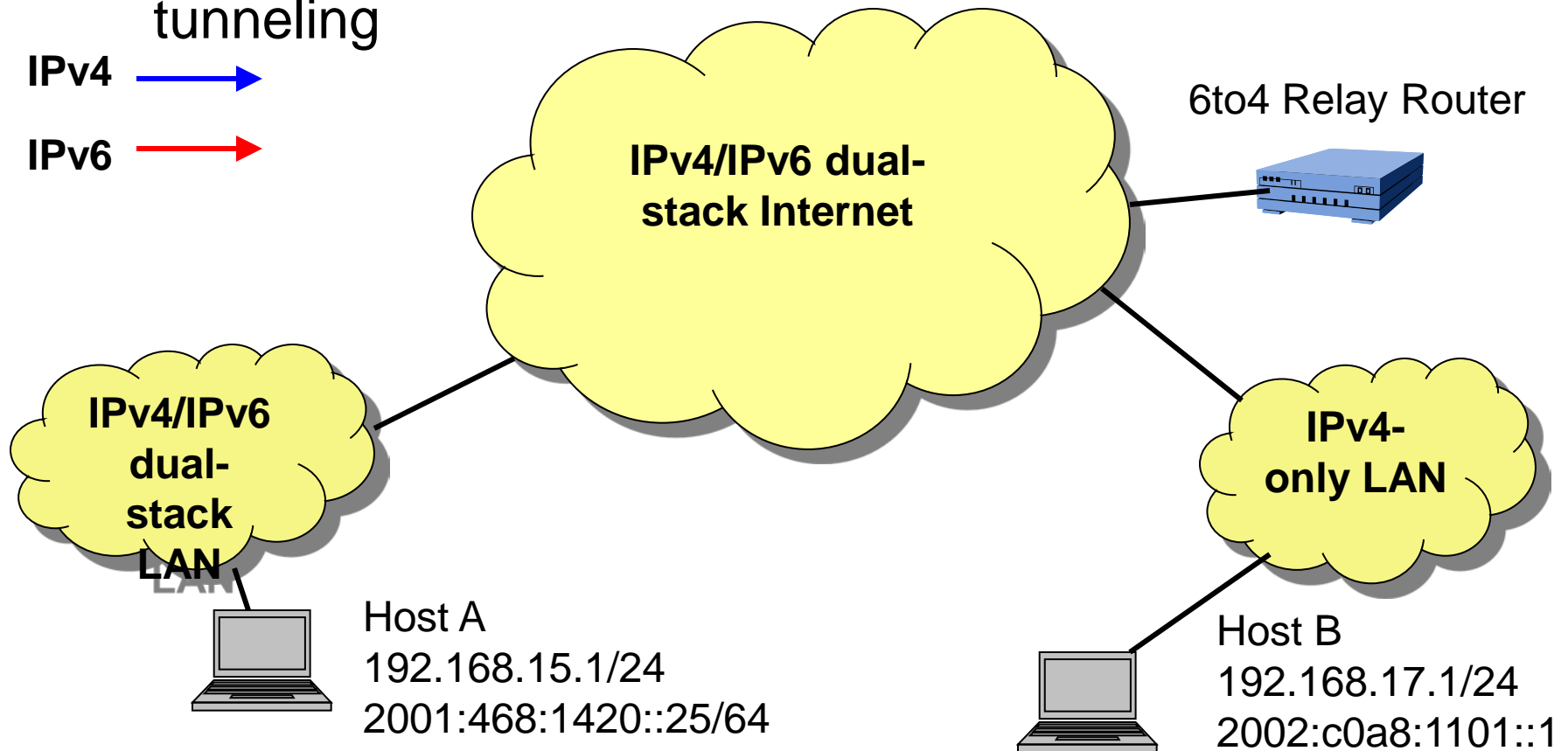
- Encapsulated IPv6 packets travel IPv4 routing path.
- No tunneling equipment or IPv6 infrastructure required between hosts

# 6to4 Implementation Scenarios (2 of 2)

Host A is on a native IPv6 network and host B is on an IPv4-only network, but is itself capable of IPv6 6to4 tunneling

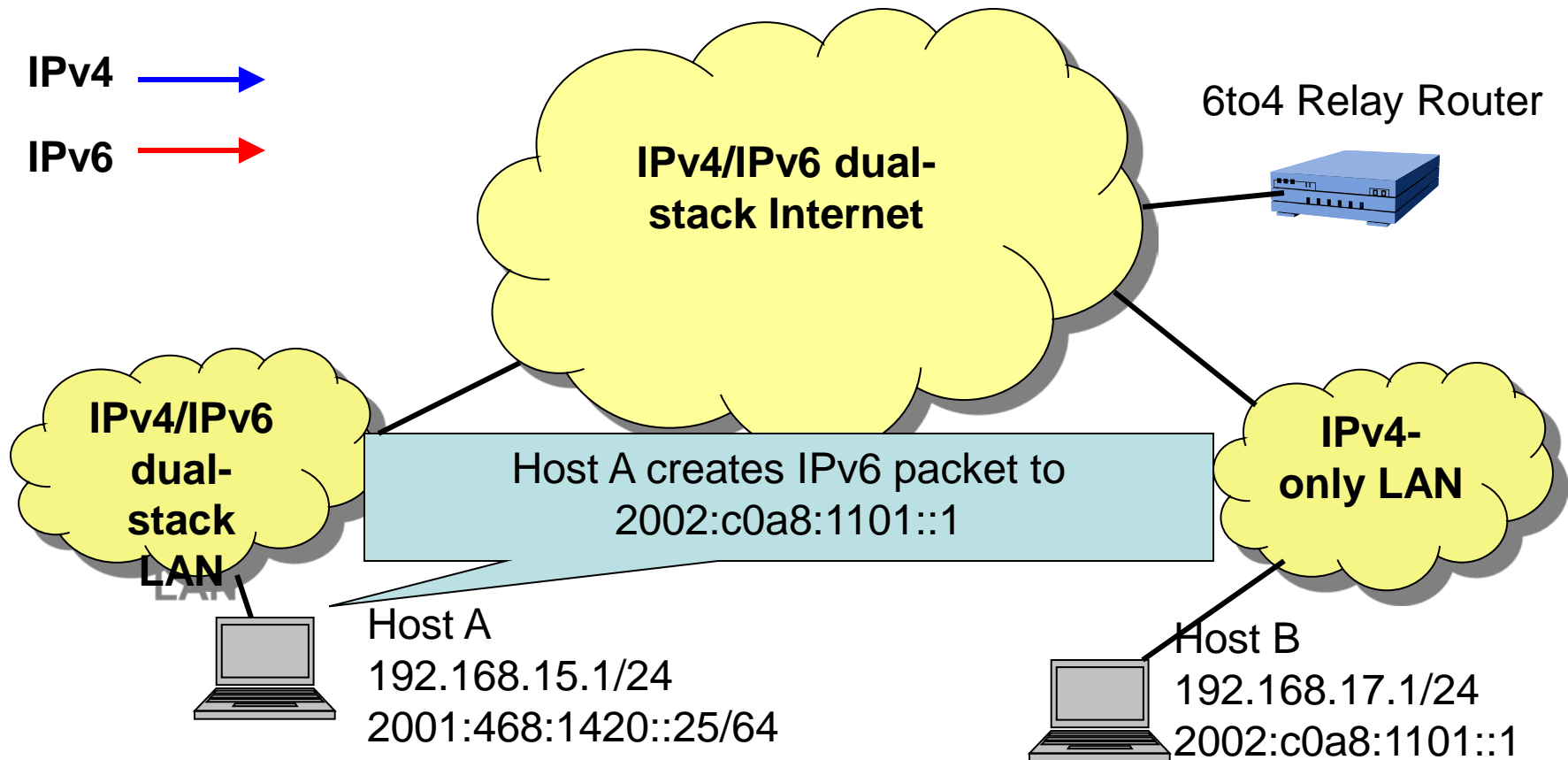
IPv4 

IPv6 



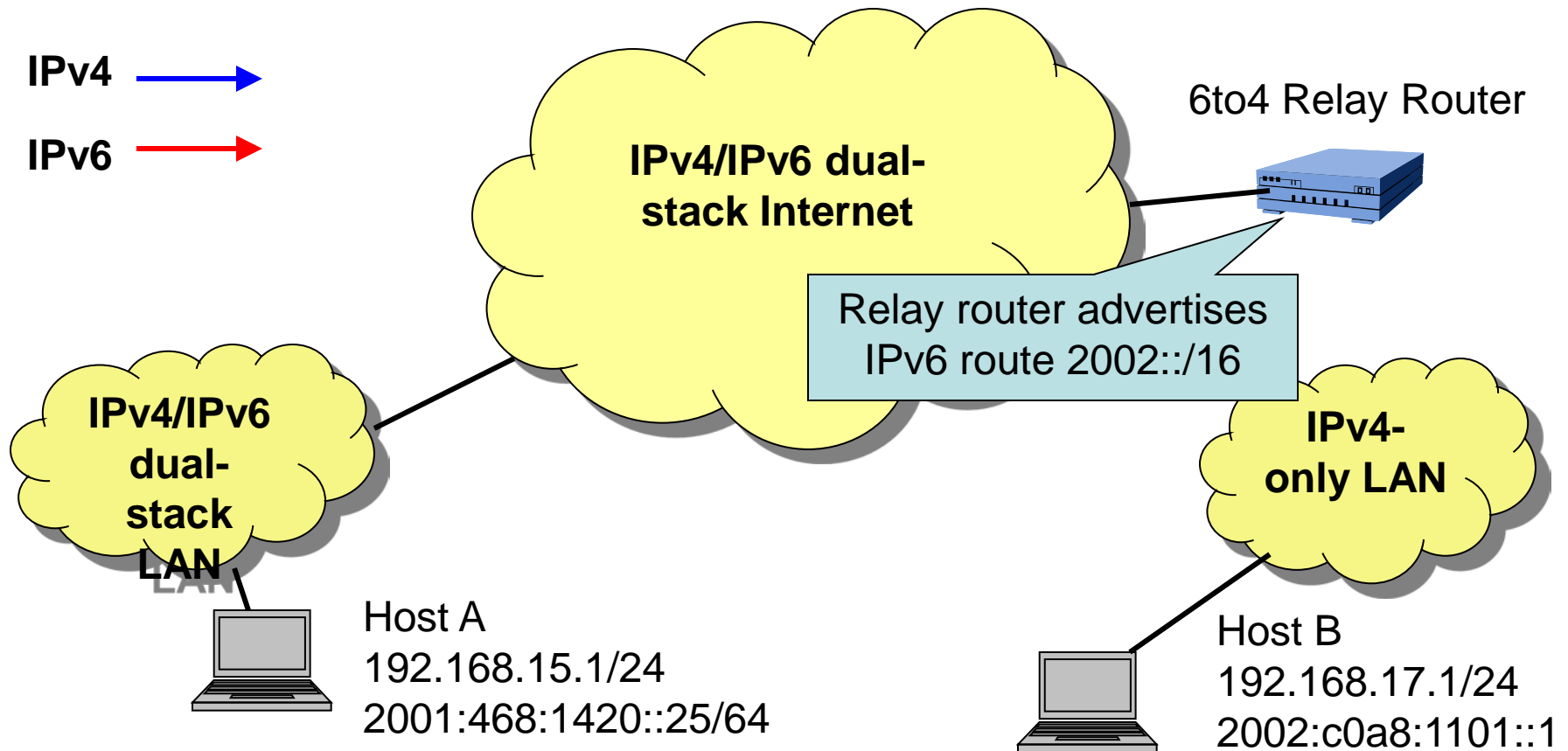
# 6to4 Implementation Scenarios (2 of 2)

Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



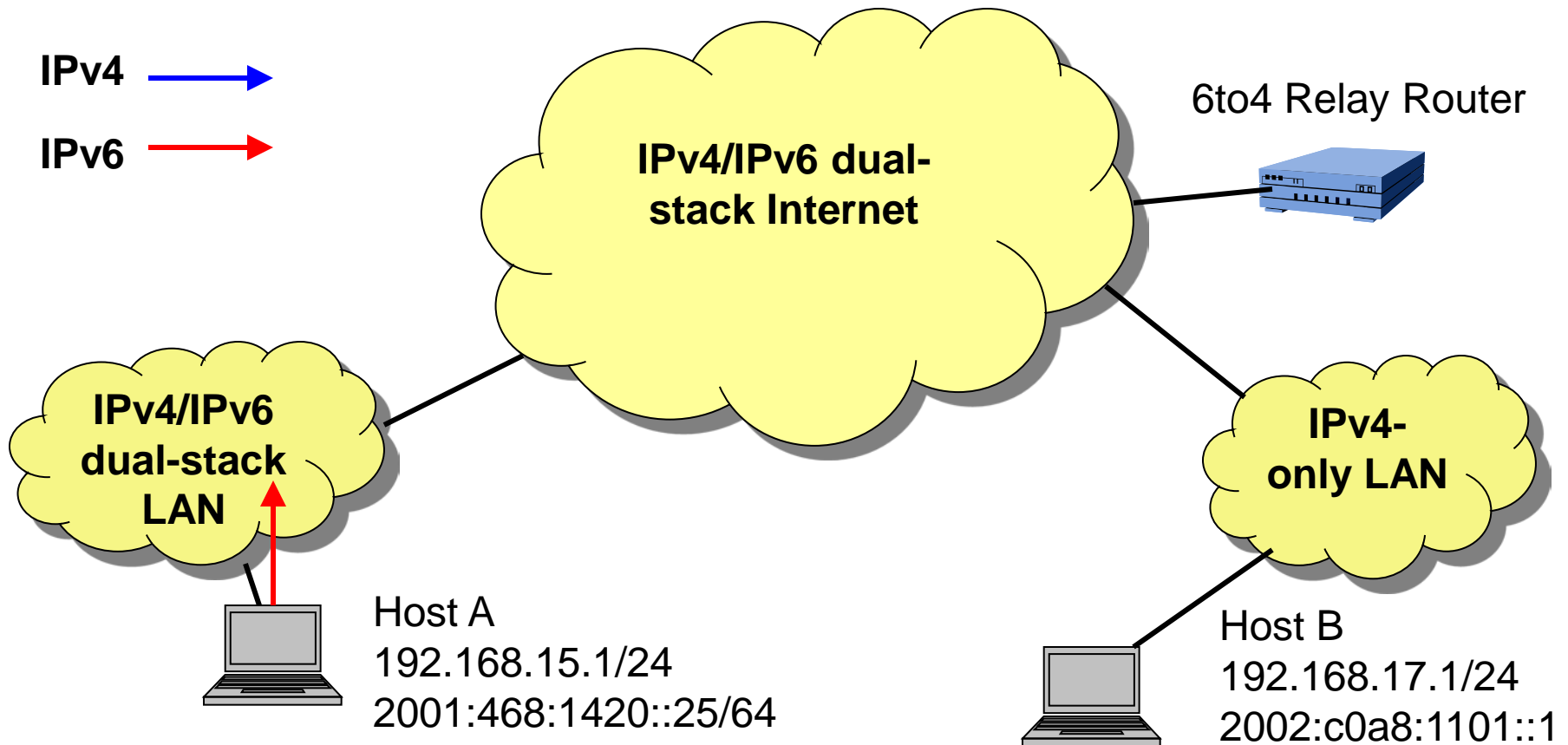
# 6to4 Implementation Scenarios (2 of 2)

Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



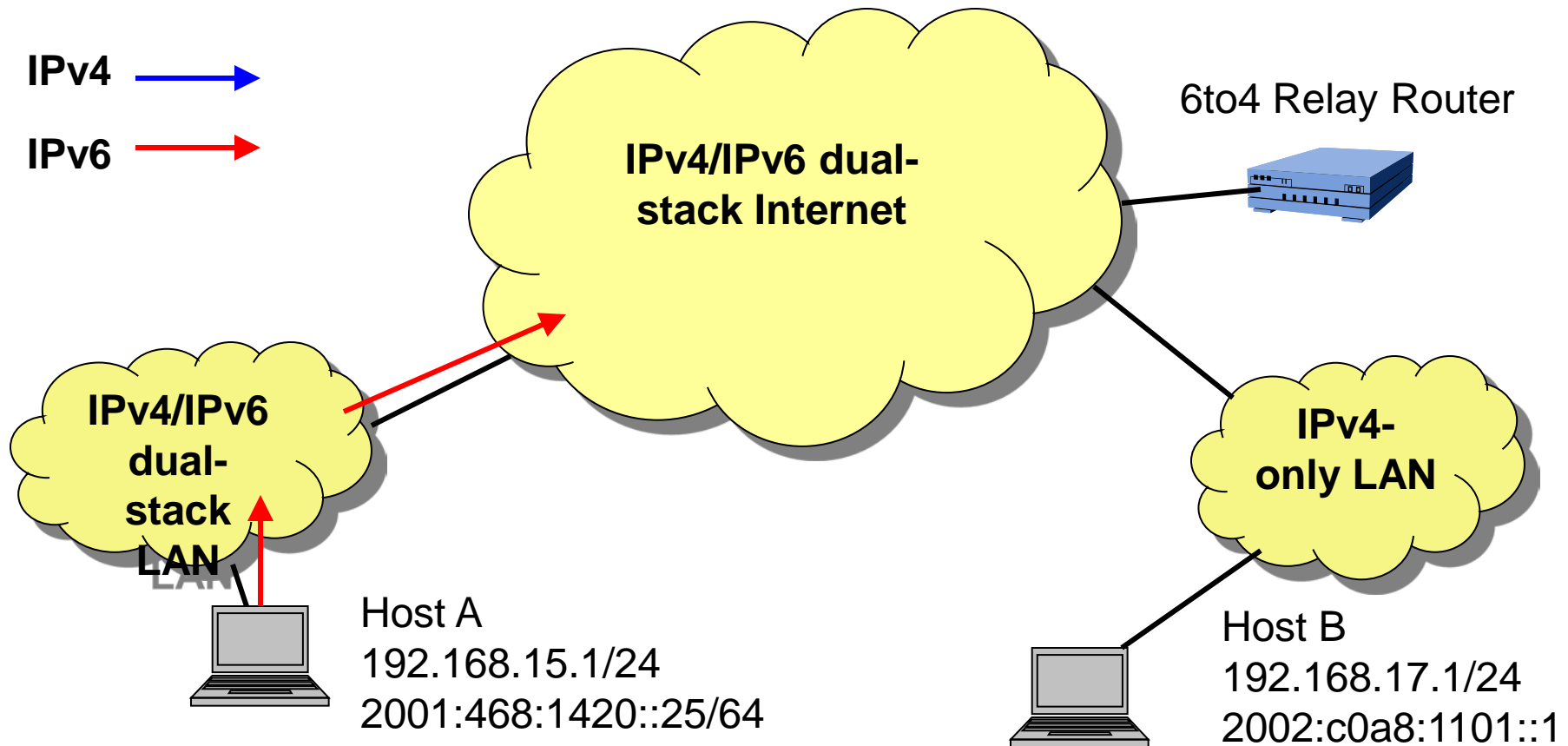
# 6to4 Implementation Scenarios (2 of 2)

Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



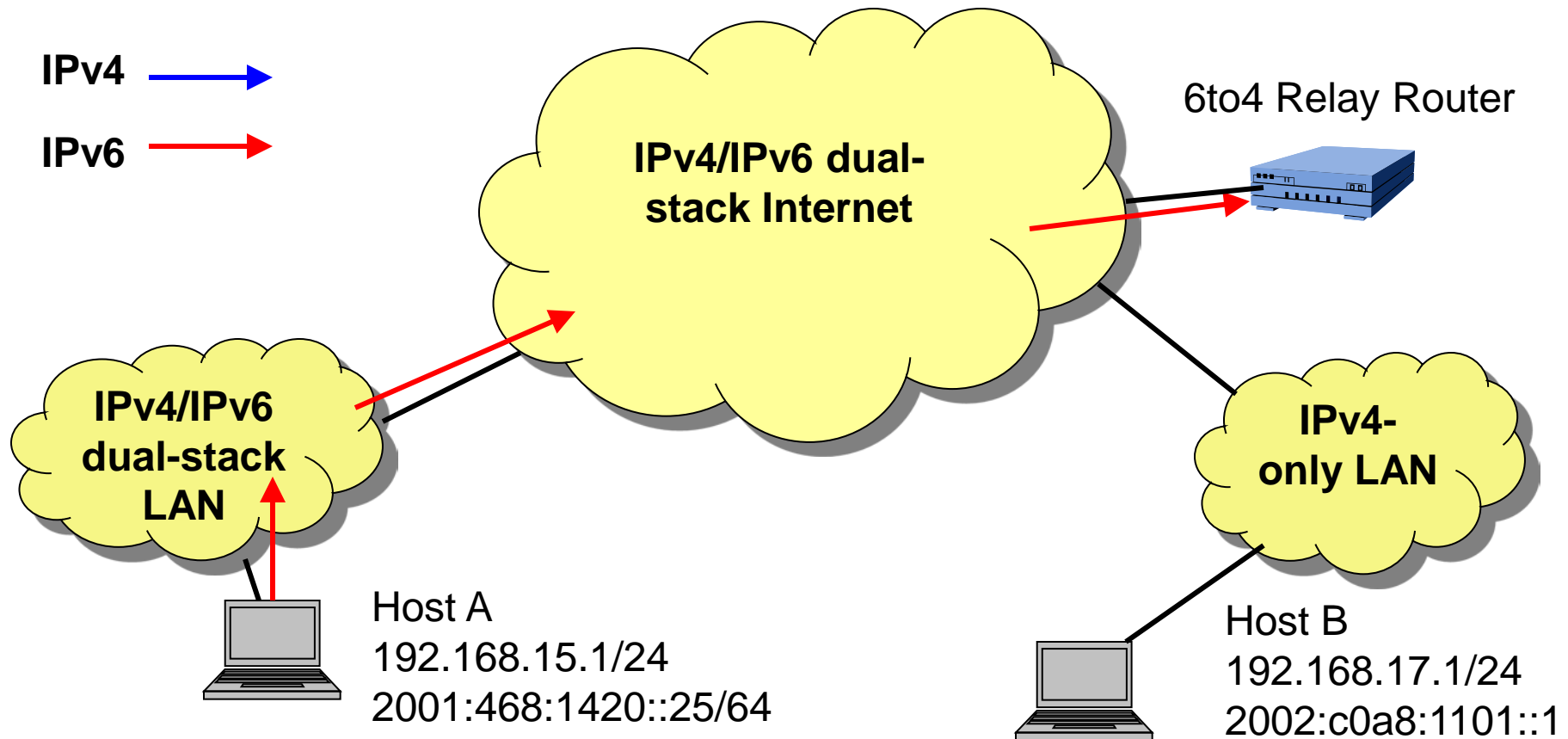
# 6to4 Implementation Scenarios (2 of 2)

Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



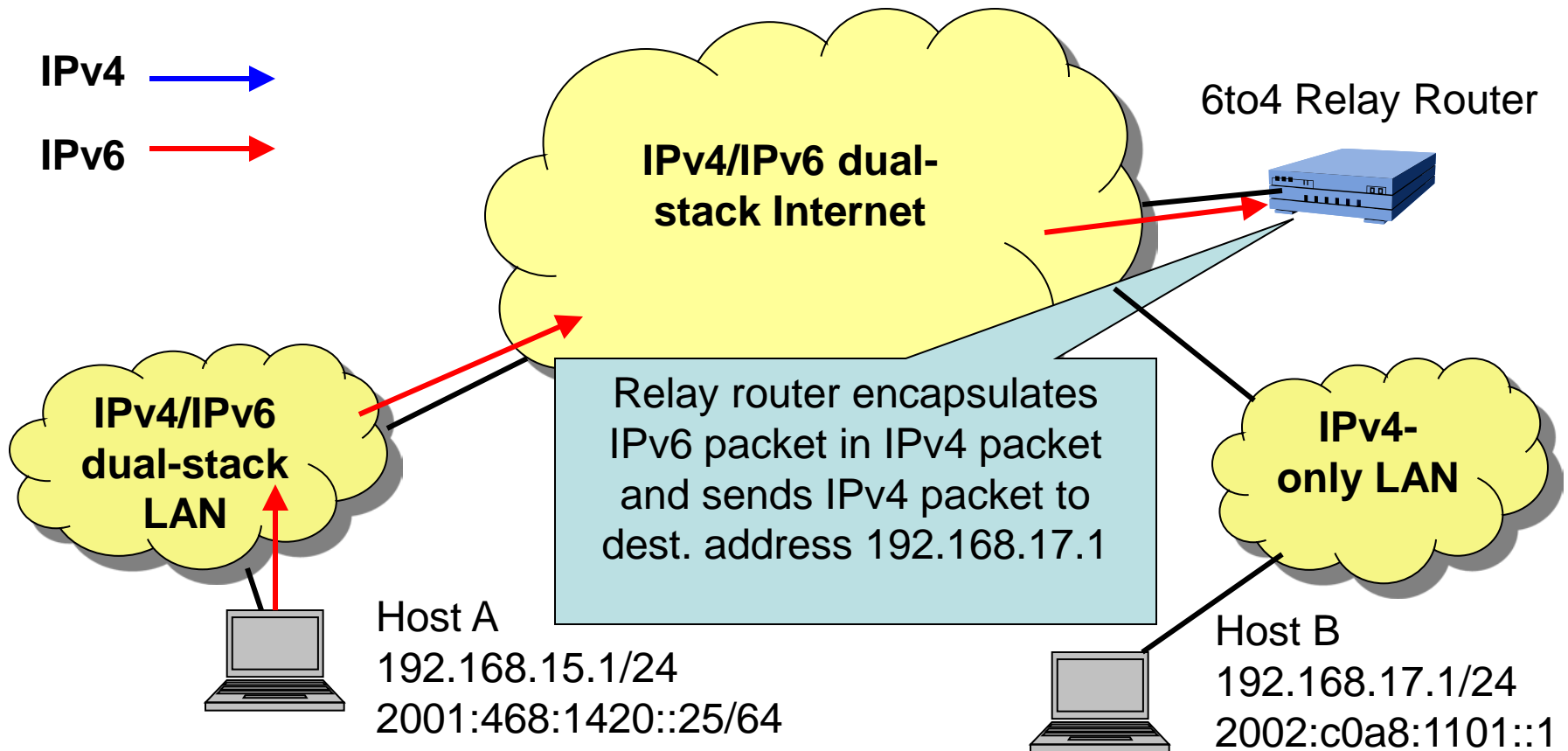
# 6to4 Implementation Scenarios (2 of 2)

Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



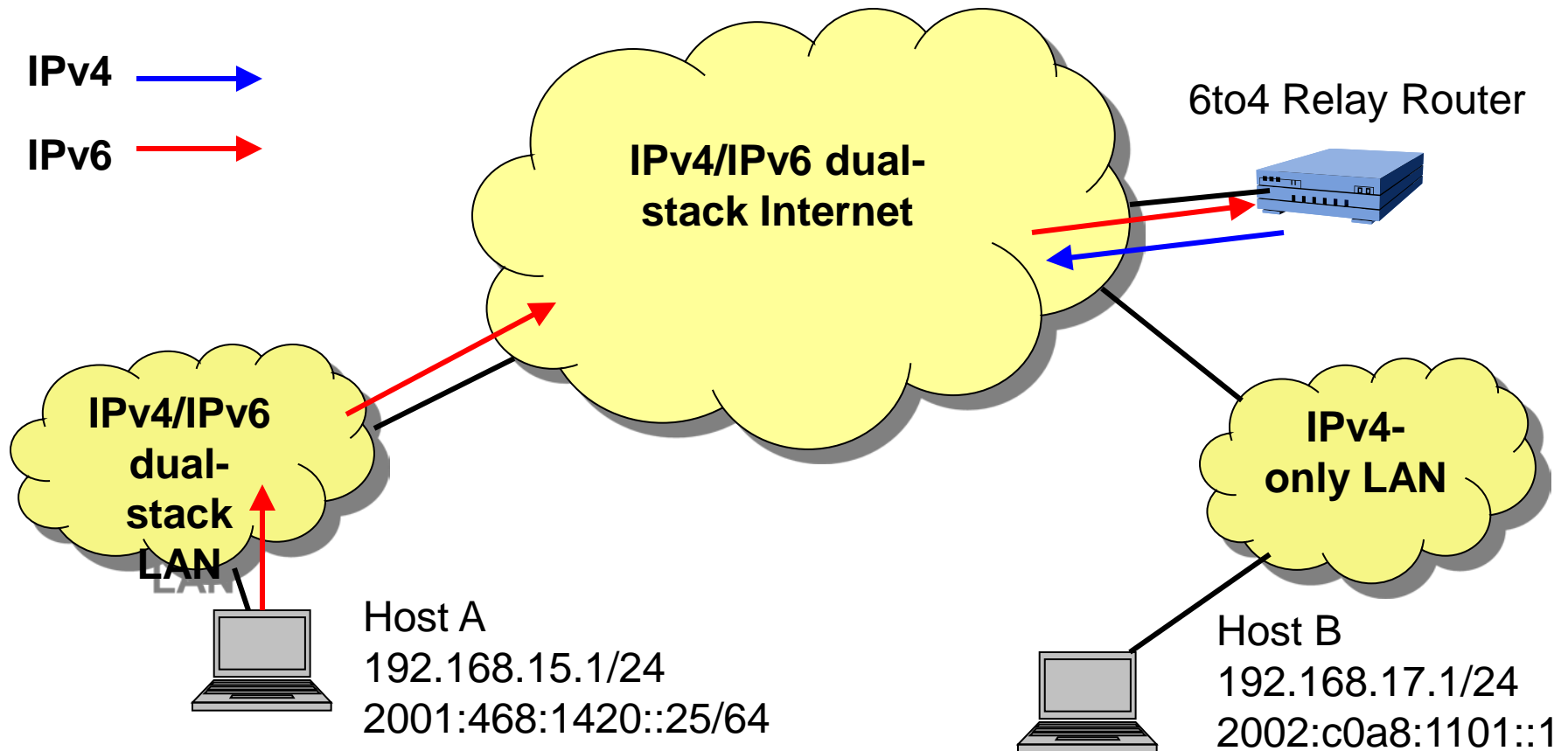
# 6to4 Implementation Scenarios (2 of 2)

Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



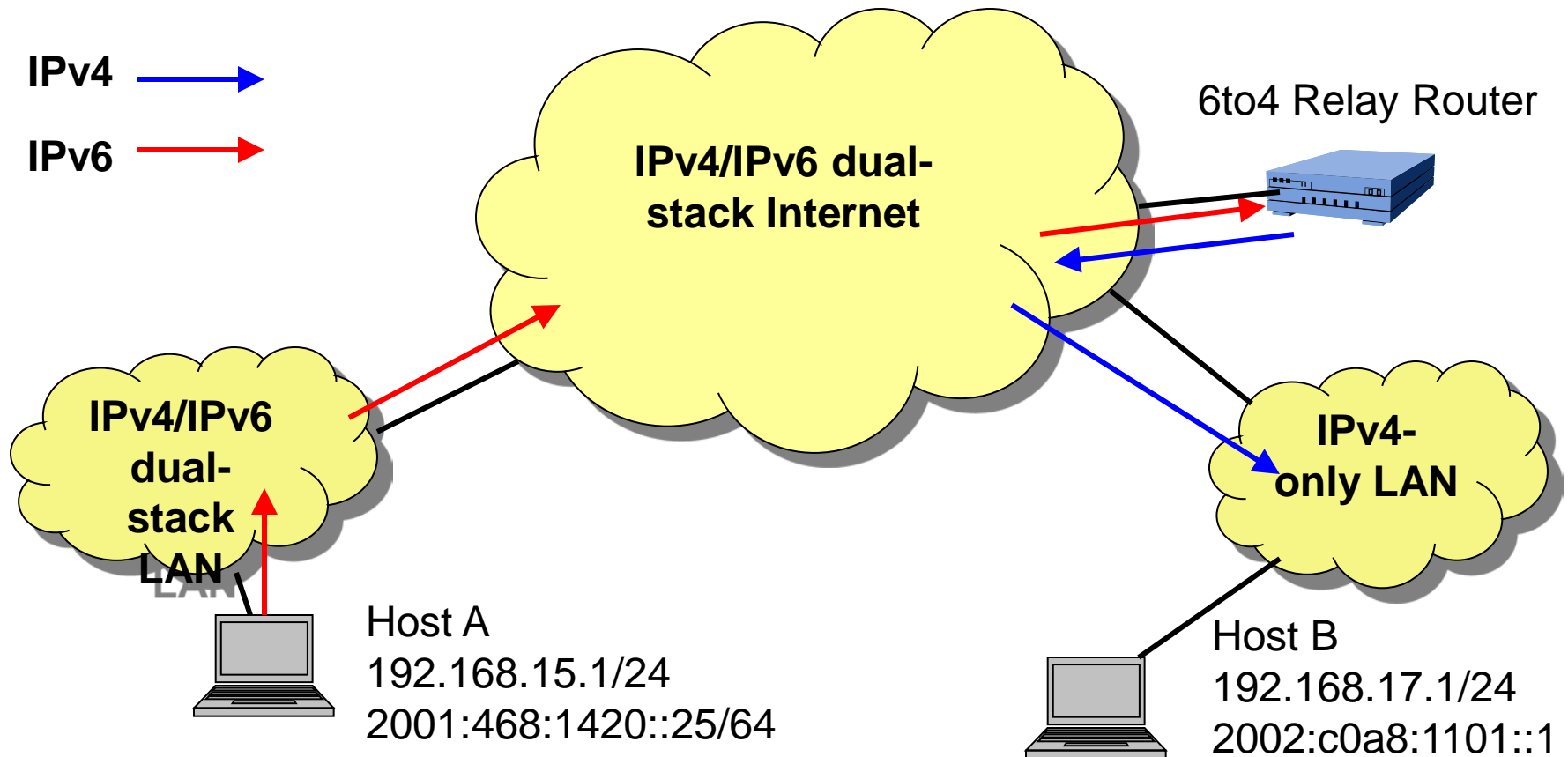
# 6to4 Implementation Scenarios (2 of 2)

Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



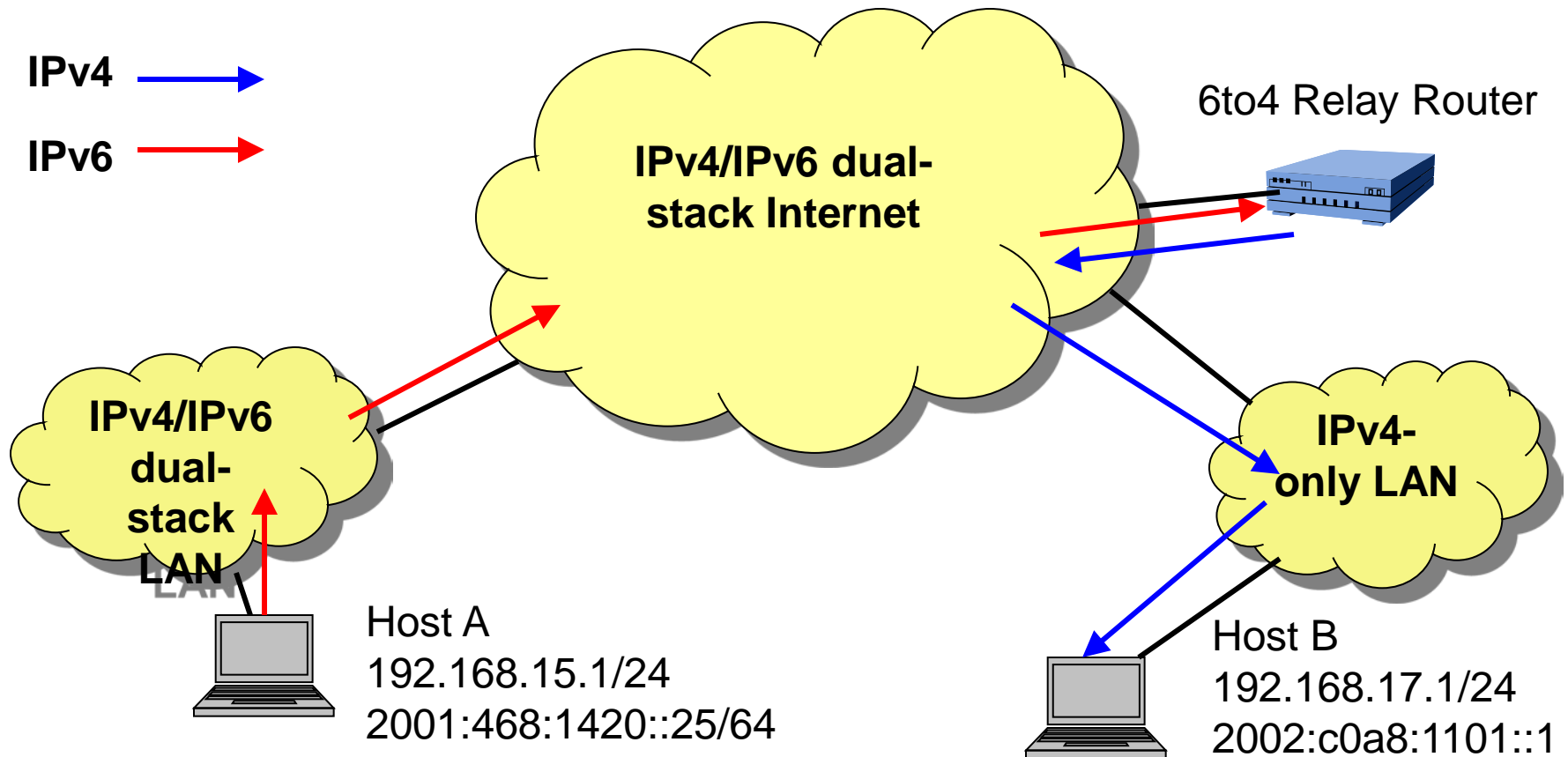
# 6to4 Implementation Scenarios (2 of 2)

Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



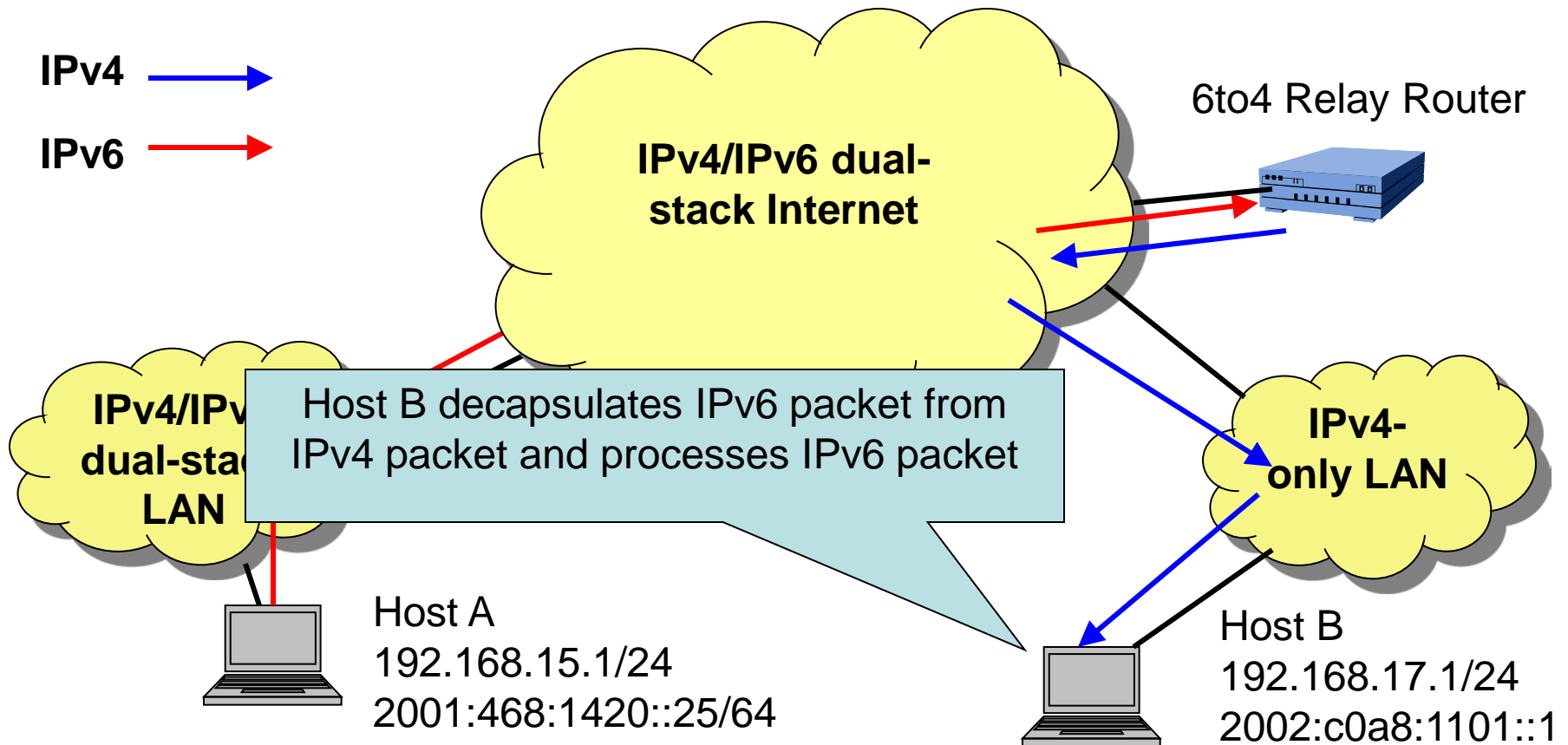
# 6to4 Implementation Scenarios (2 of 2)

Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (2 of 2)

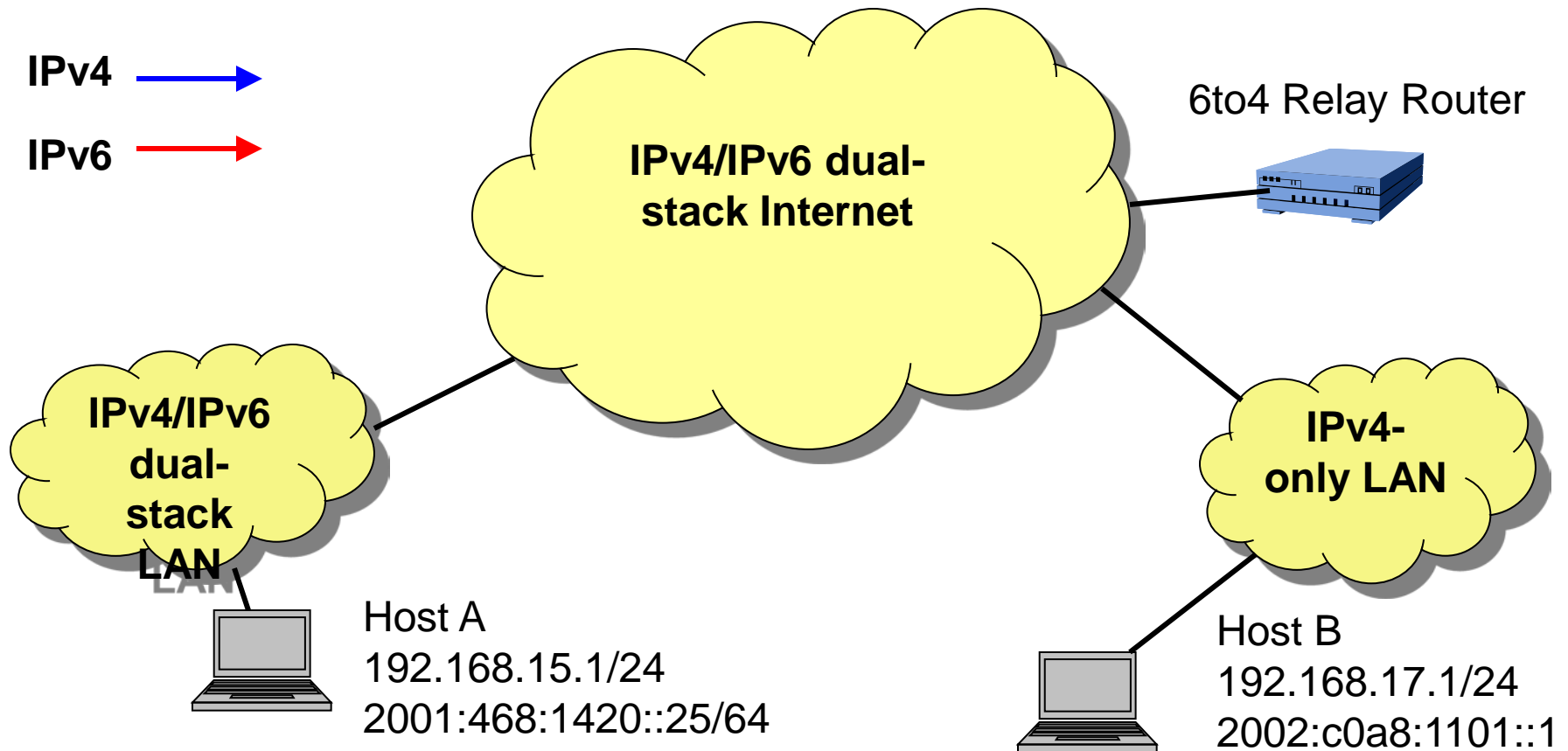
Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (2 of 2)

## Reverse Direction

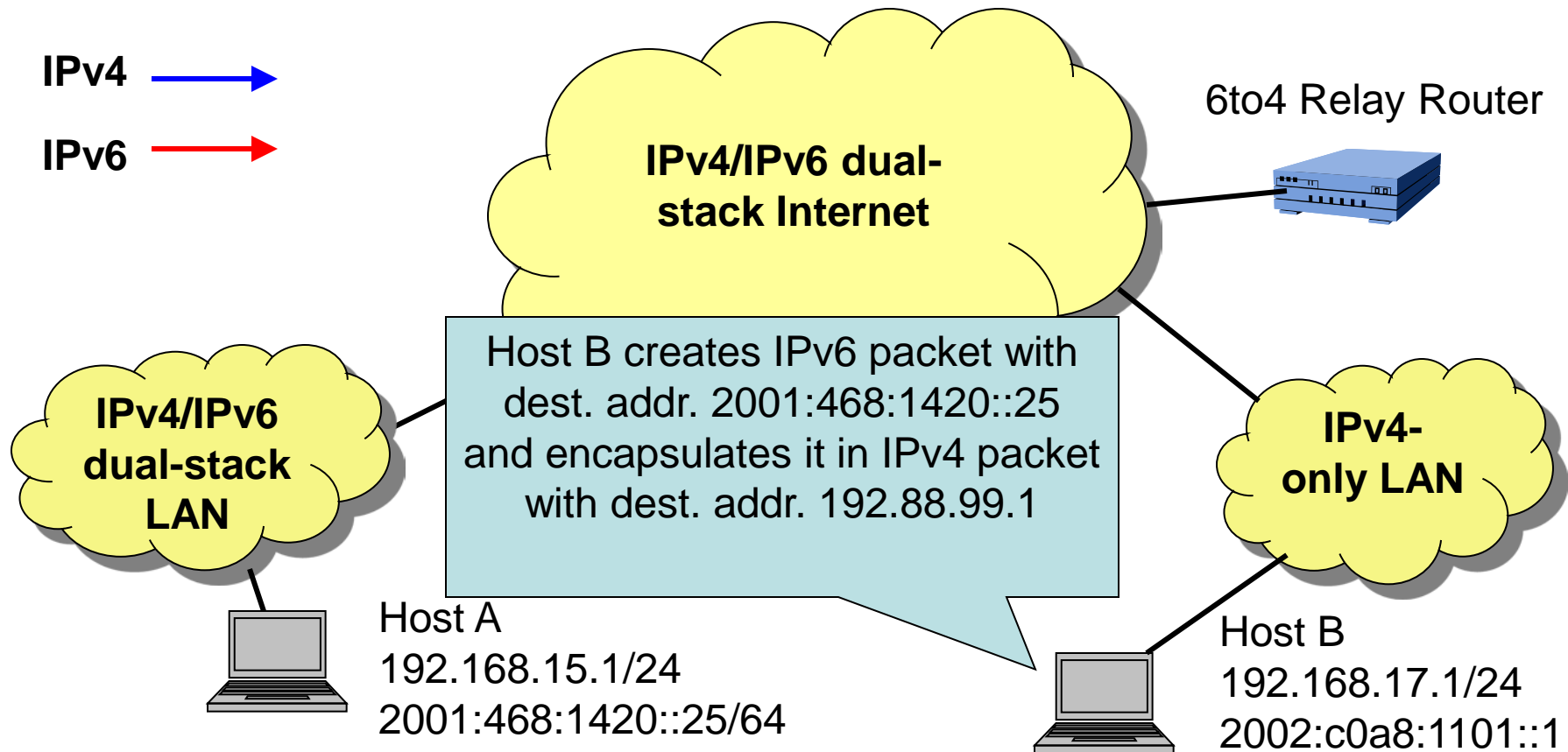
Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (2 of 2)

## Reverse Direction

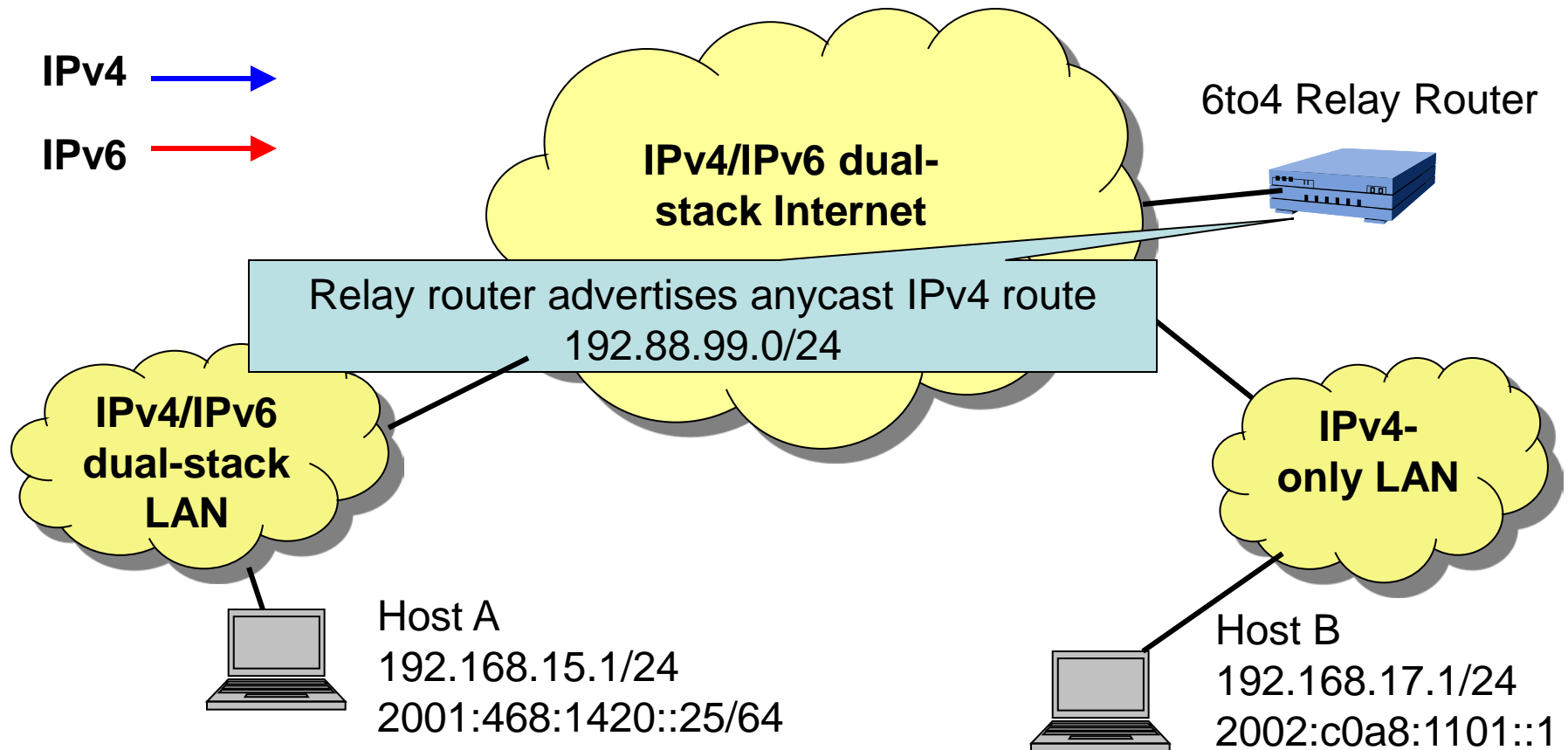
Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (2 of 2)

## Reverse Direction

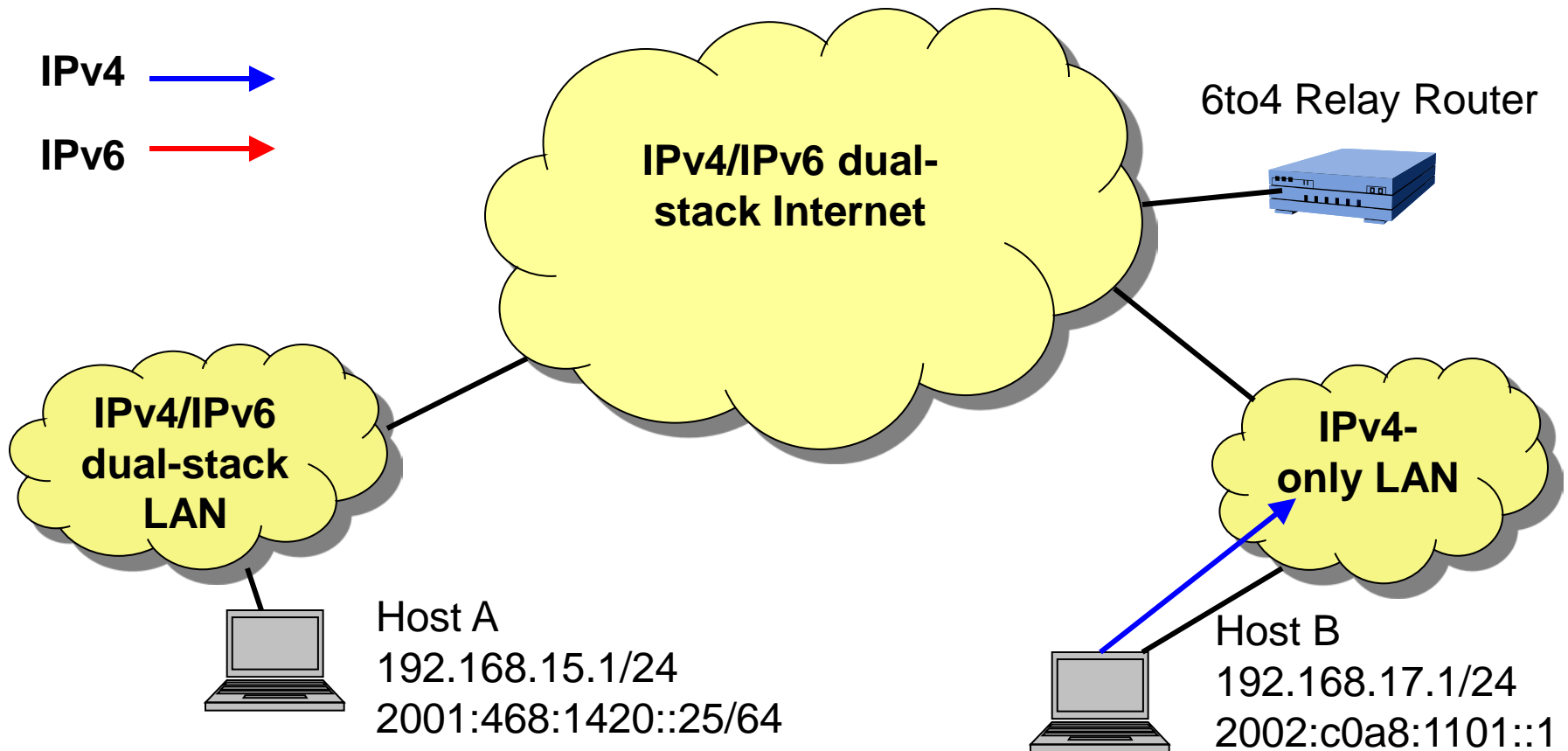
Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (2 of 2)

## Reverse Direction

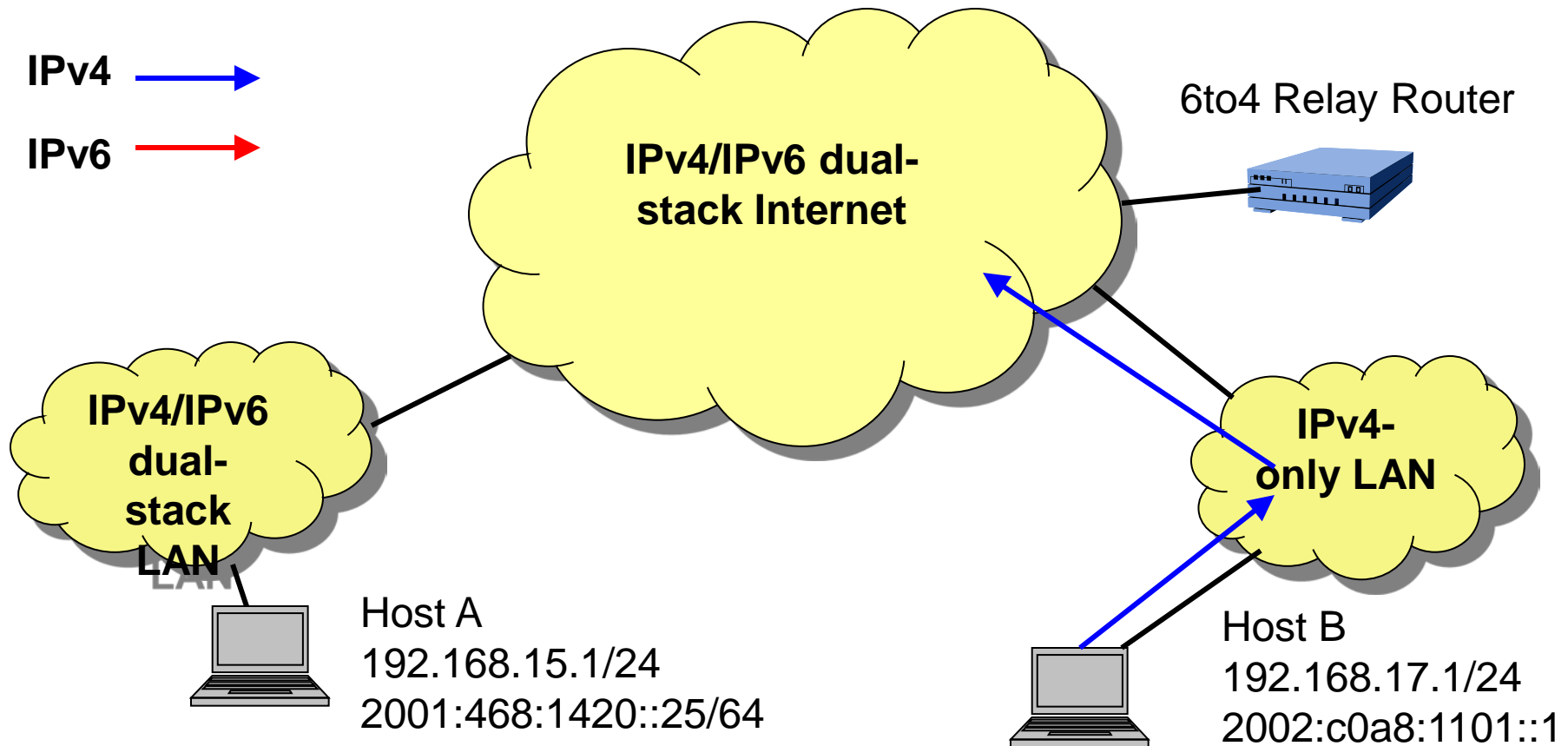
Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (2 of 2)

## Reverse Direction

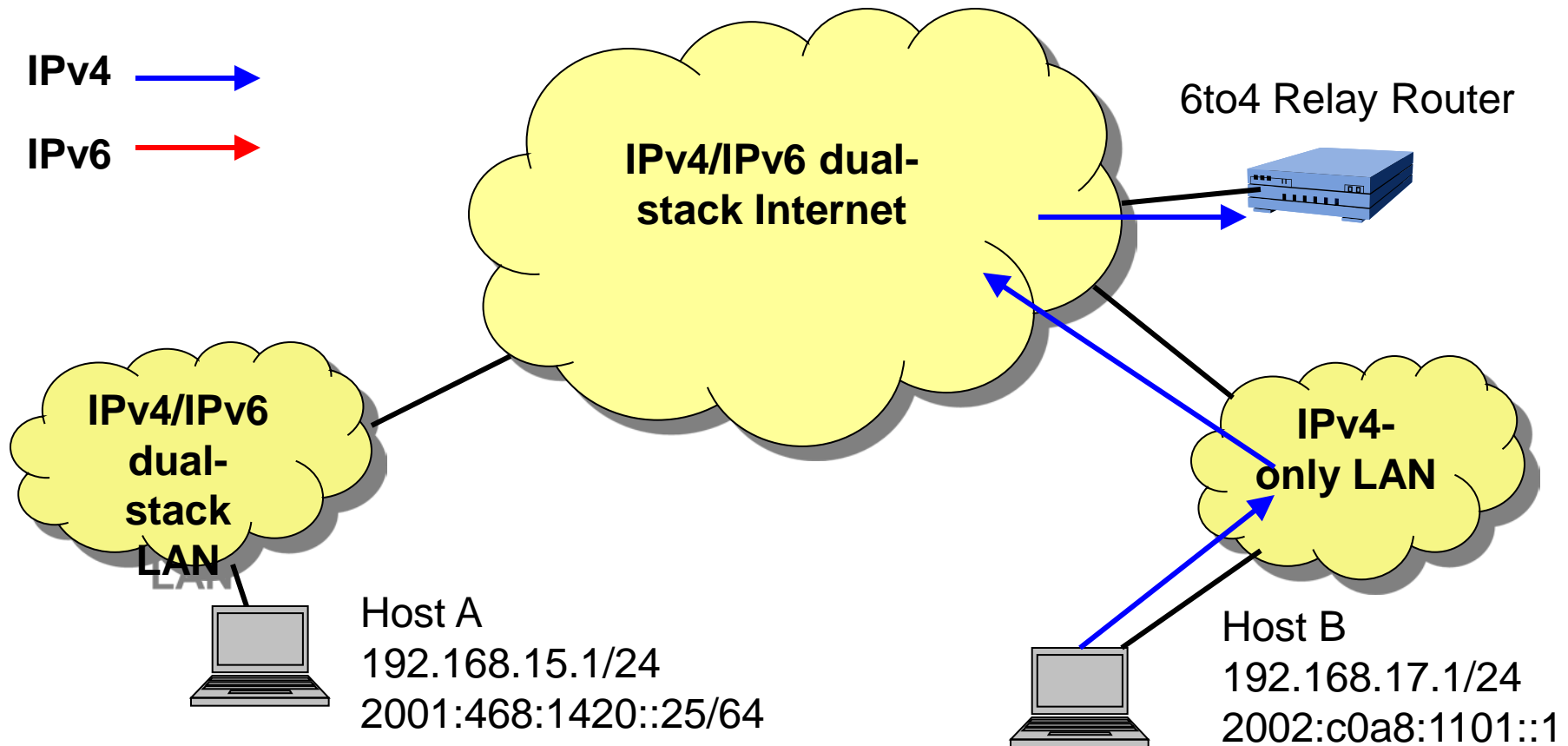
Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (2 of 2)

## Reverse Direction

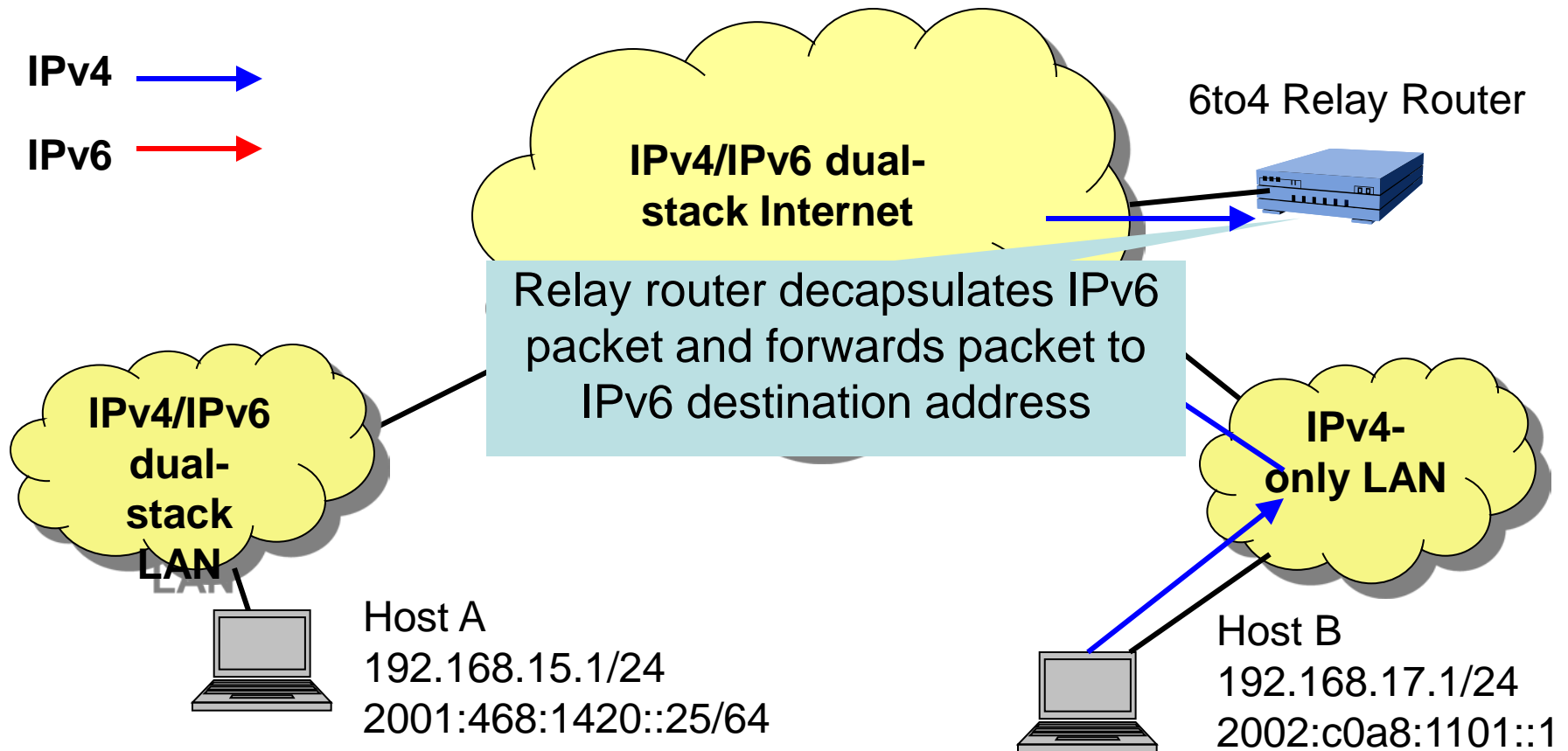
Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (2 of 2)

## Reverse Direction

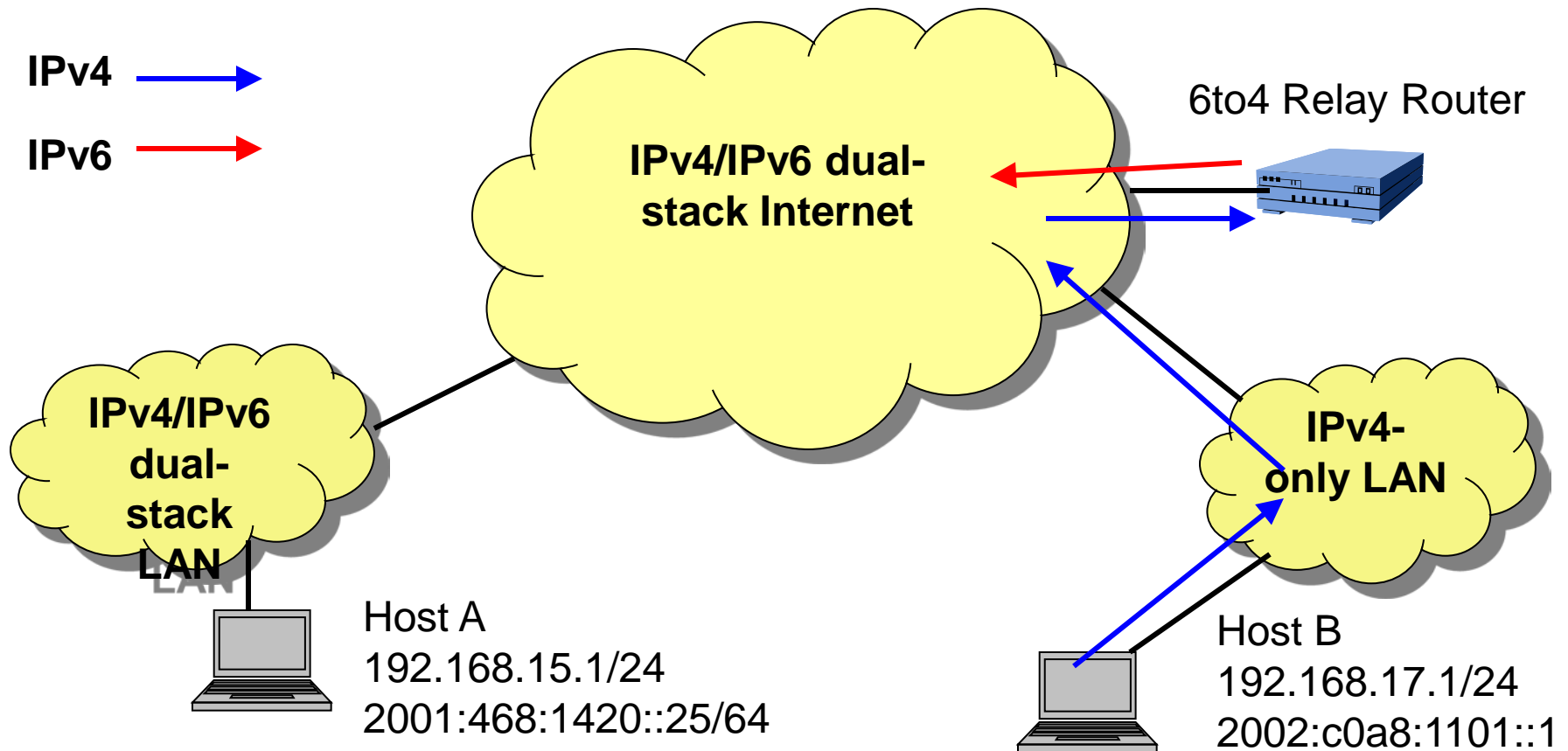
Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (2 of 2)

## Reverse Direction

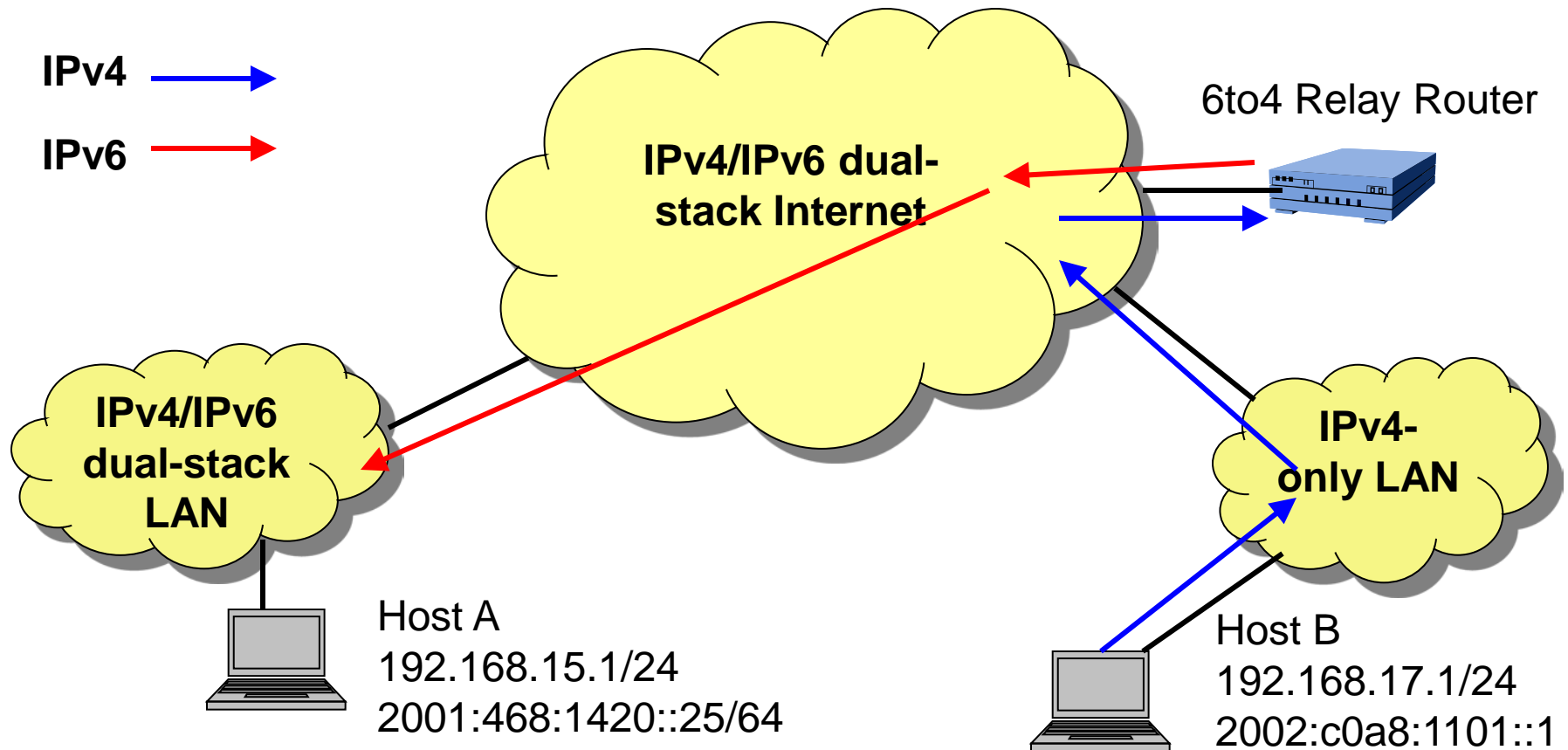
Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (2 of 2)

## Reverse Direction

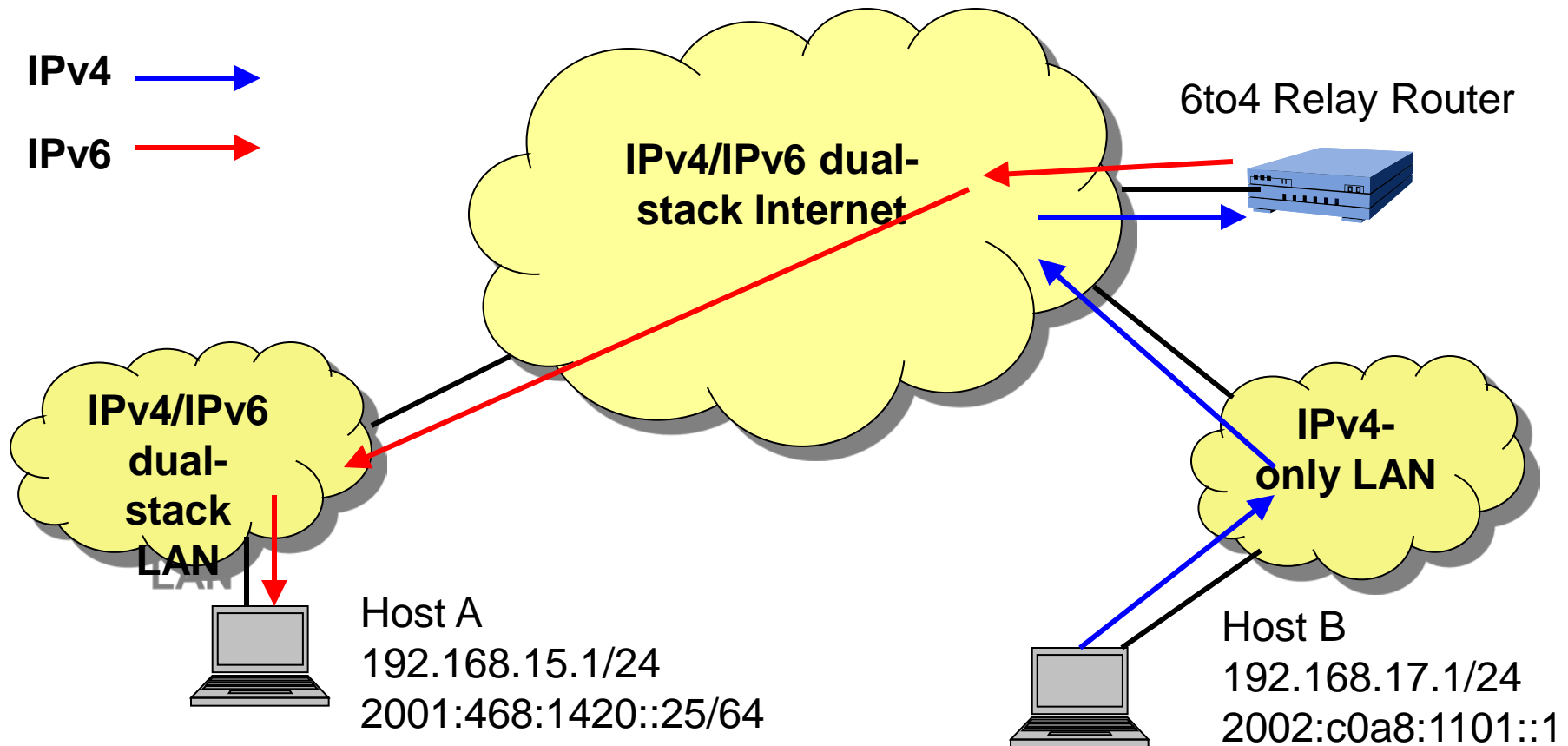
Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (2 of 2)

## Reverse Direction

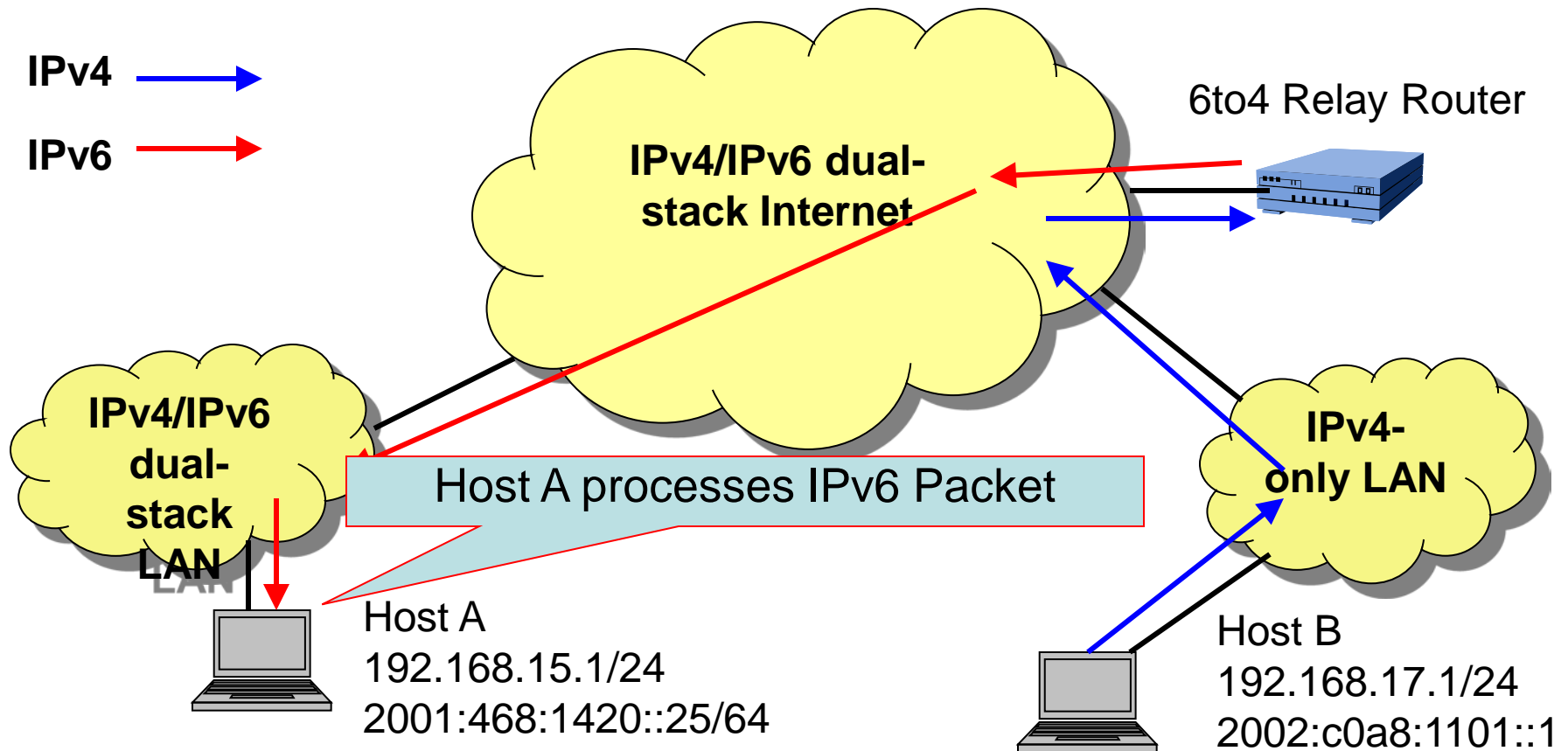
Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (2 of 2)

## Reverse Direction

Host A is on native IPv6 network, host B is on IPv4-only network, but is itself capable of IPv6 6to4 tunneling



# 6to4 Implementation Scenarios (2 of 2)

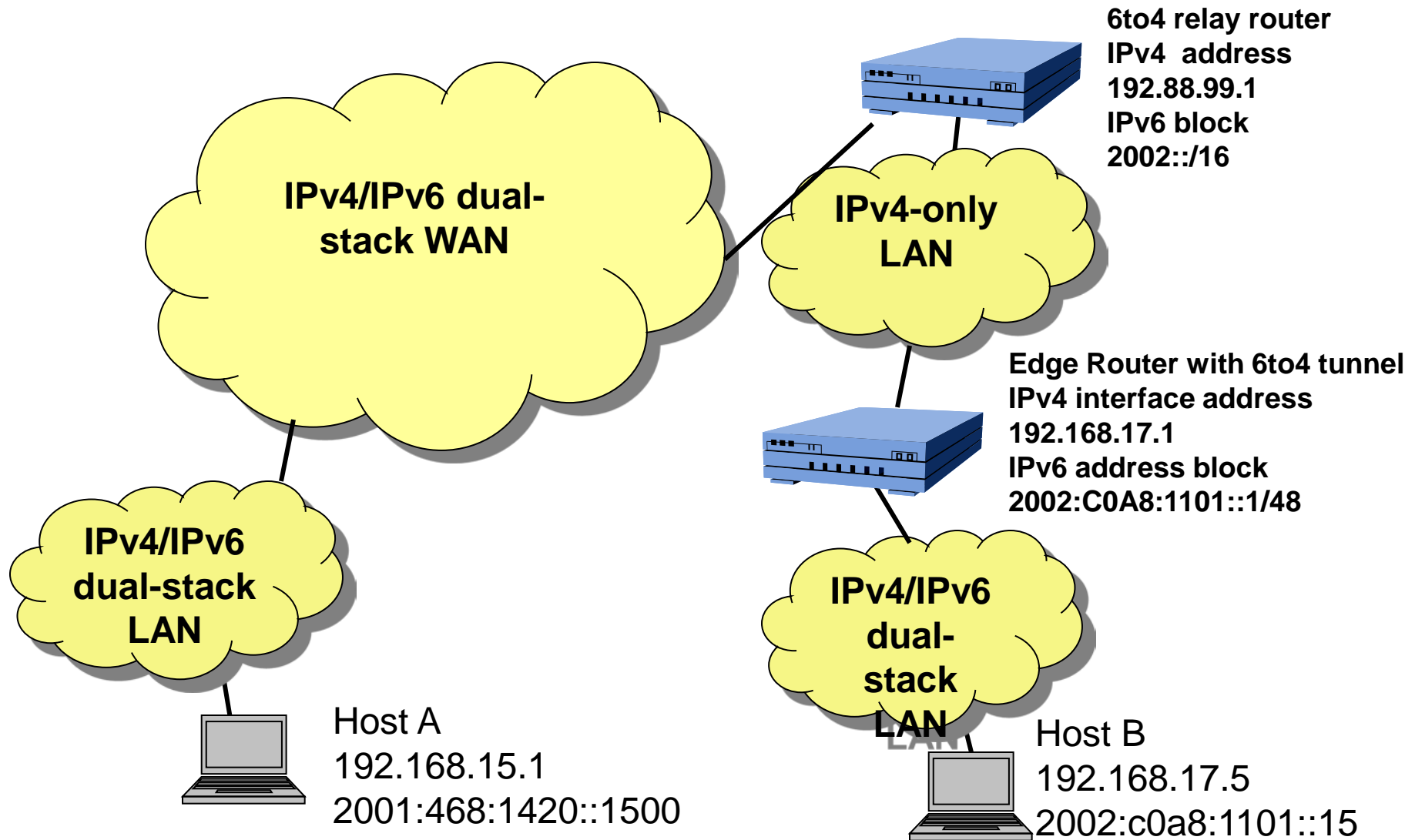
## Observations

- Asymmetric routes common
  - “Nearest” 6to4 relay router may not be the same for all end-points of a connection
- Placement of 6to4 relay routers can have significant impact on 6to4 tunnel performance

# Alternate 6to4 Scenario

- An edge router could be used instead of a host computer for any of the 6to4 tunnel endpoints.
- Edge router can provide a /48 IPv6 subnet for each IPv4 address.
- There are open-source Linux loads for the Linksys 54G/GS home routers that can run 6to4 relays
  - <http://www.linksysinfo.org/>

# Alternate 6to4 Scenario



# 6to4 Relay Platforms

- Cisco IOS releases that support IPv6
- Linux
- FreeBSD

# 6to4 Security Issues

See:

RFC 3964 – Security Considerations for IPv6

<http://www.ietf.org/rfc/rfc3964.txt>

# Dual Stack

- This is likely to be the predominant network-layer transition tool.
- It appears that when all the tools using tunnel mechanisms were being developed, no one thought viable dual-stack routers would show up as quickly as they in fact have.
  - Most backbones could be dual-stack very easily, and will be when there is a demand.

# Transition

- Tunnels will remain useful as a tool for connecting isolated hosts in home networks to v6 nets
  - Earthlink secure IPv6 in IPv4 tunnel using open-source Linux on Linksys 54G/GS  
<http://www.research.earthlink.net/ipv6/>

# Host level transition

- This is where transition could bog down.
- How do you make web and other servers transparently accessible to either v6 or v4 hosts?
- There are several approaches.
  - Dual stack
  - Bump-in-the-stack
  - NAT-like devices
  - Translators

# Translators

- Within Linux variants there is a tool called Faithd.
  - This is a transport layer translator.
- There are also header translators out there:
  - SIIT
  - Nat-PT (historical)
  - Socks
  - Various application specific translators

# Summary

- This is neither as hard as it was once thought nor as easy as we might like to make it.
- Dual Stack will be viable much sooner than was thought.