FTP64:
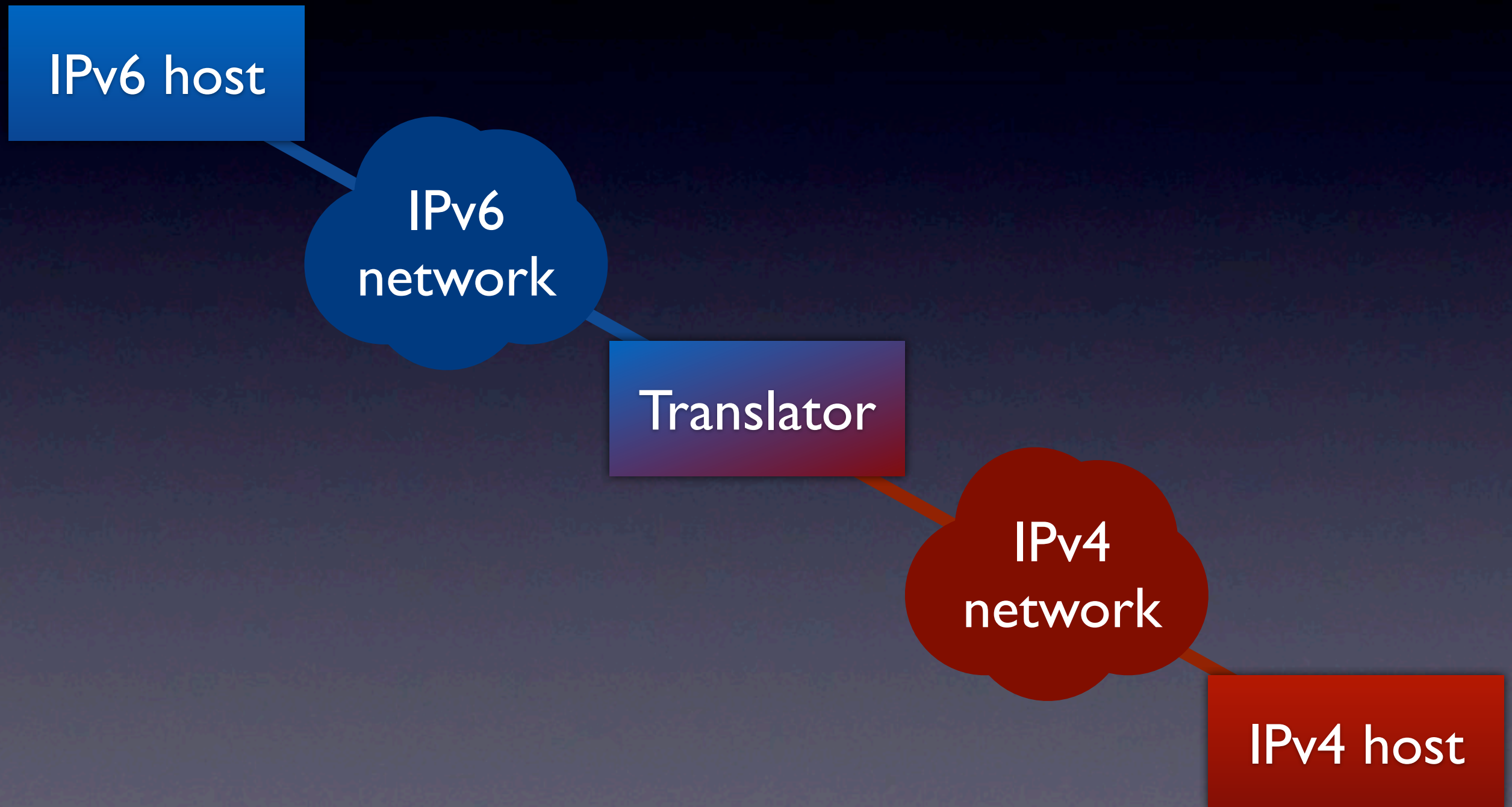## making FTP work through IPv6→IPv4 translators

draft-van-beijnum-behave-ftp64-05

IETF-75, Stockholm, july 2009
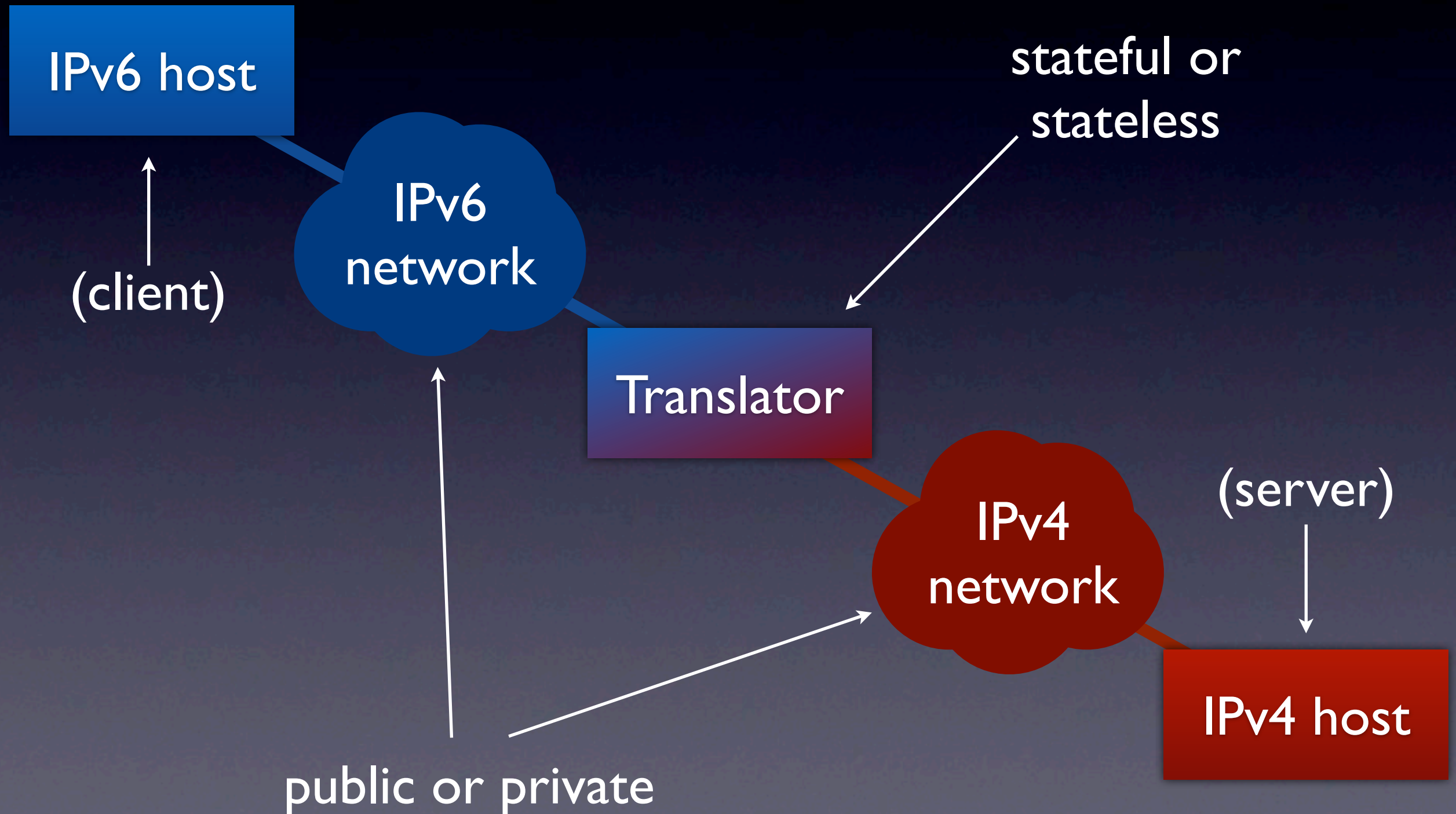
apps area & behave working group meetings

Iljitsch van Beijnum

IPv6→IPv4 translation

IPv6 host

IPv6 network

(client)

stateful or stateless

Translator

(server)

IPv4 network

IPv4 host

public or private

3

FTP and IPv6

- RFC 959 (1985):
  - passive mode (PASV): v4 address+port
  - active mode (PORT): v4 address+port
- RFC 2428 (1998):
  - passive mode (EPSV): only a port
  - active mode (EPRT): address family, address, port

4

Non-problem?

- If all clients and servers use EPSV FTP works without trouble through translators!

Installed base, servers

- 2598 reachable IPv4 servers with anonymous access tested by Dan Wing:

  - EPSV:

    - 65% working EPSV

    - 28% "50x command not implemented"

    - 6% time out setting up data connection

    - 1% "425 can't open data connection"

Installed base, servers

- 96% have working PASV

- 2% non-working PASV but working EPSV

- 2% nonworking PASV / EPSV

- All servers use control channel address in 227 response to PASV

- Many servers react badly to "EPSV X"

  - with X = 1: IPv4, X = 2: IPv4

Installed base, clients

- IPv6 clients, not extensively surveyed:

  - Firefox and Safari use EPSV

  - MacOS/FreeBSD command line client uses EPSV

  - Windows command line client uses EPRT

  - clients don't use "EPSV X"

Possible solutions

1. Update servers to support EPSV

2. Update clients to fall back on PASV despite IPv4 address in 227 response

3. Implement application layer gateway in the translator

- The draft *mandates* 1 and 2 and *describes* 3.

Server requirements

- MUST:

  - support EPSV

    - ability to switch off EPSV/EPRT

    - report whether EPSV available in FEAT

  - only use control channel address in PASV

Client requirements

- MUST support passive mode (EPSV)

- Recommended that after EPSV failure (5xx or timeout) retry with PASV

  - assume that address in 227 response is control channel remote address

- MUST NOT use arguments with EPSV

ALG

- EPSV→PASV is easy
- EPRT→PORT is harder with a stateful translator → could be left unimplemented
- Bother with stateful port 20 handling?
  - not too hard, but still in use?
- Don't try to translate three-way FTP
- Go into transparent mode after AUTH