Improved Path Exploration in shim6-based Multihoming

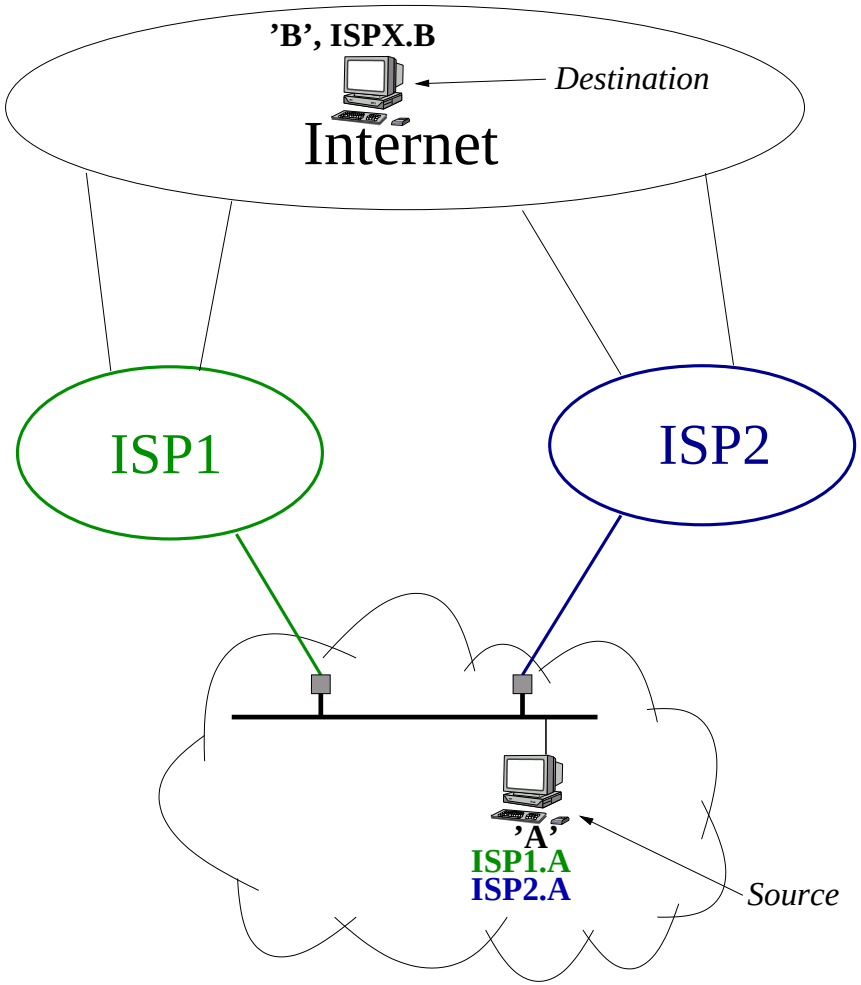
Sébastien Barré, Olivier Bonaventure

Université catholique de Louvain
http://inl.info.ucl.ac.be

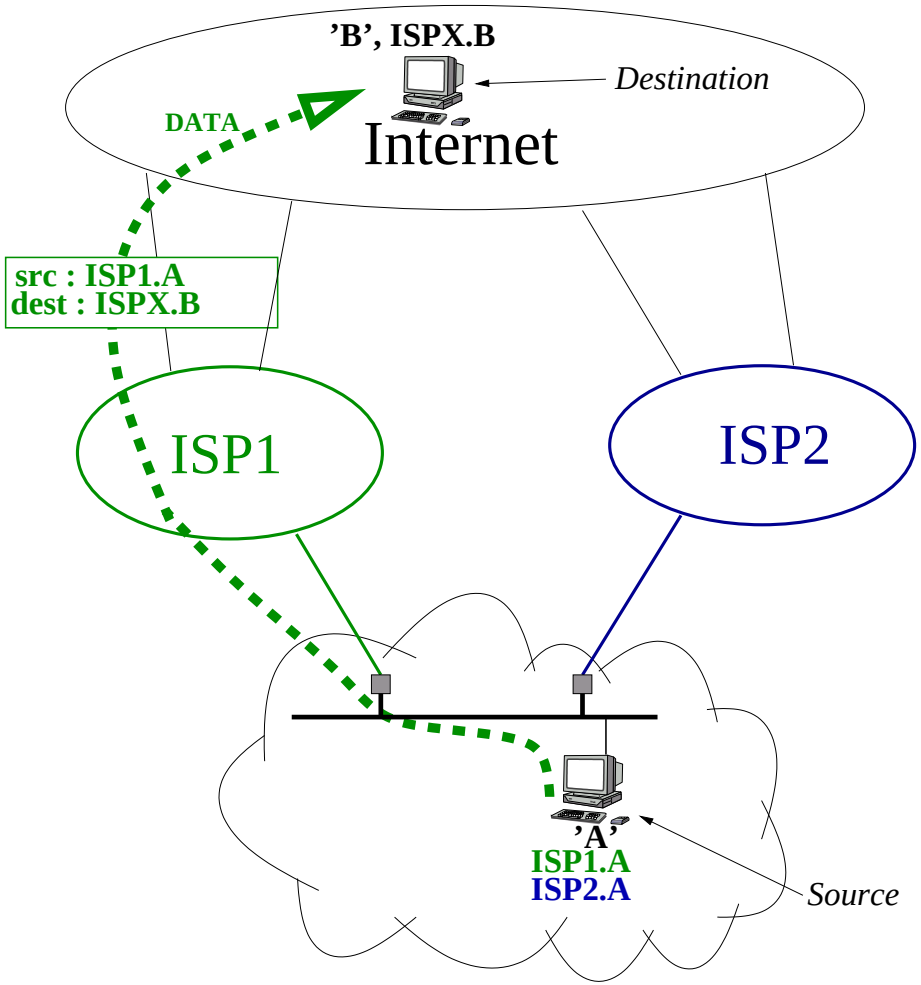
Aug. 31st, 2007

SIGCOMM 2007 Workshop "IPv6 and the Future of the Internet"

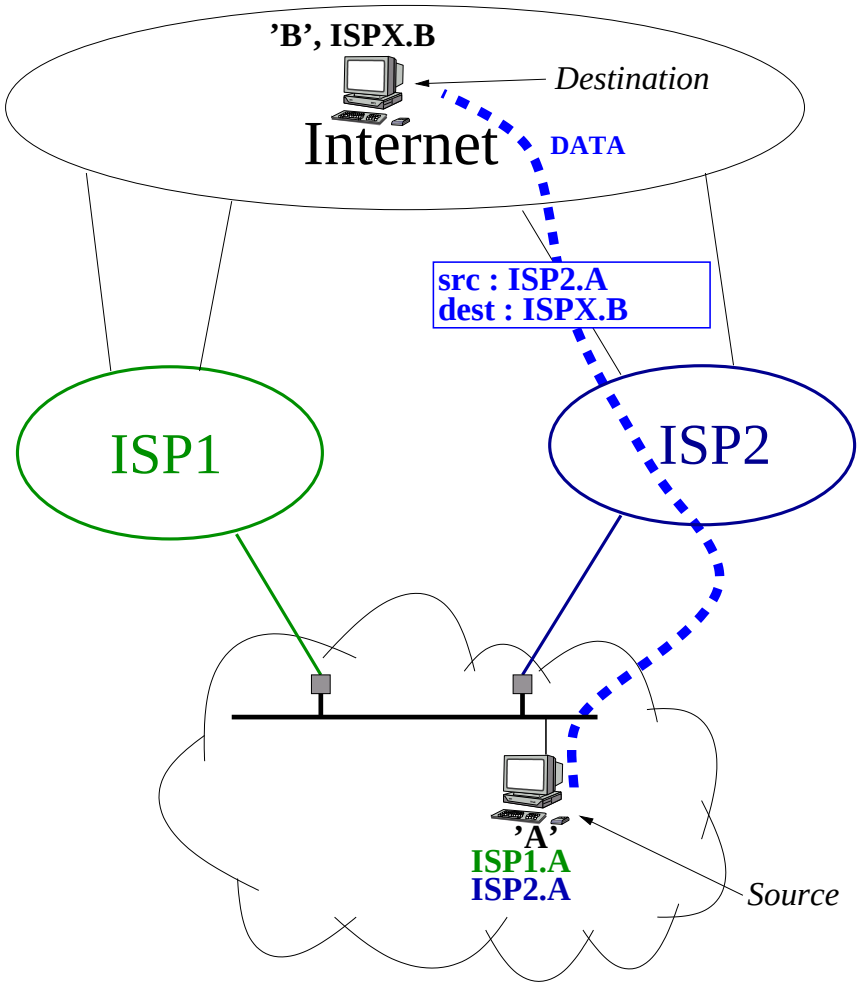
Host-centric multihoming (the context)



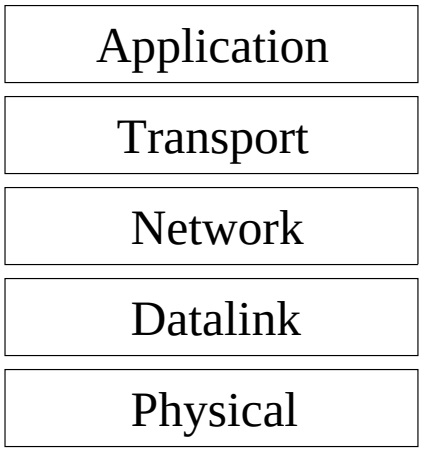
Host-centric multihoming (the context)



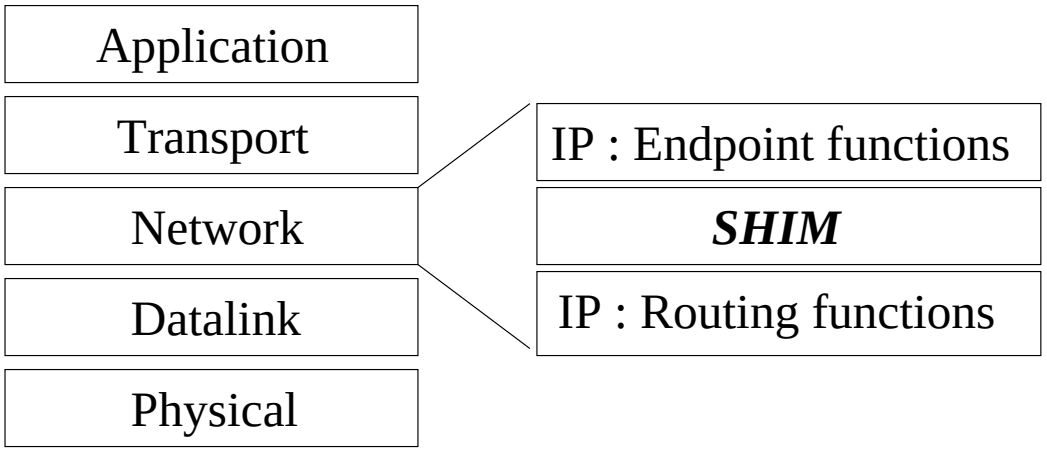
Host-centric multihoming (the context)



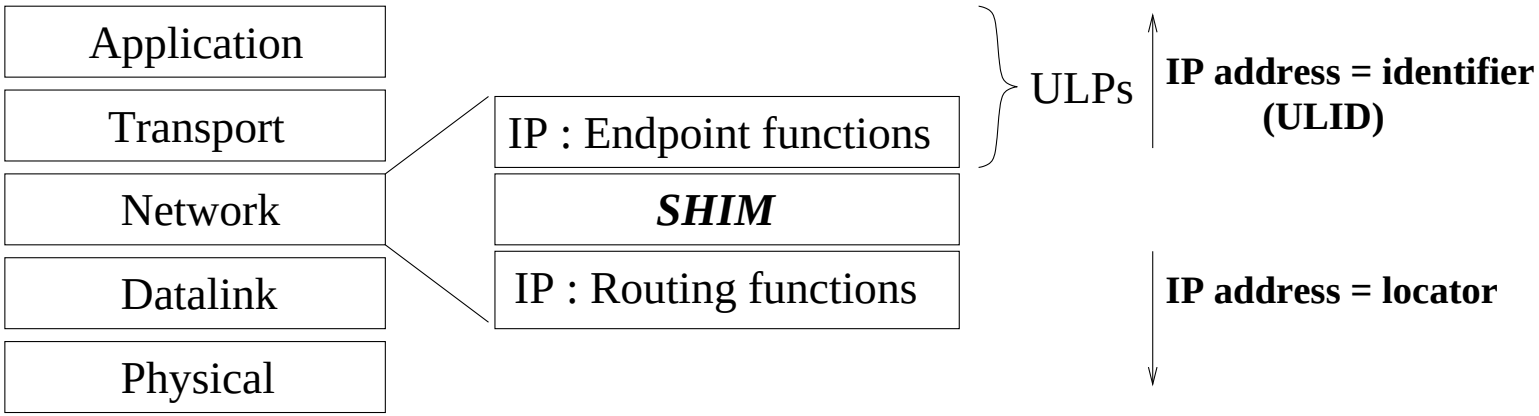
Locators vs Identifiers (ULIDs)



Locators vs Identifiers (ULIDs)



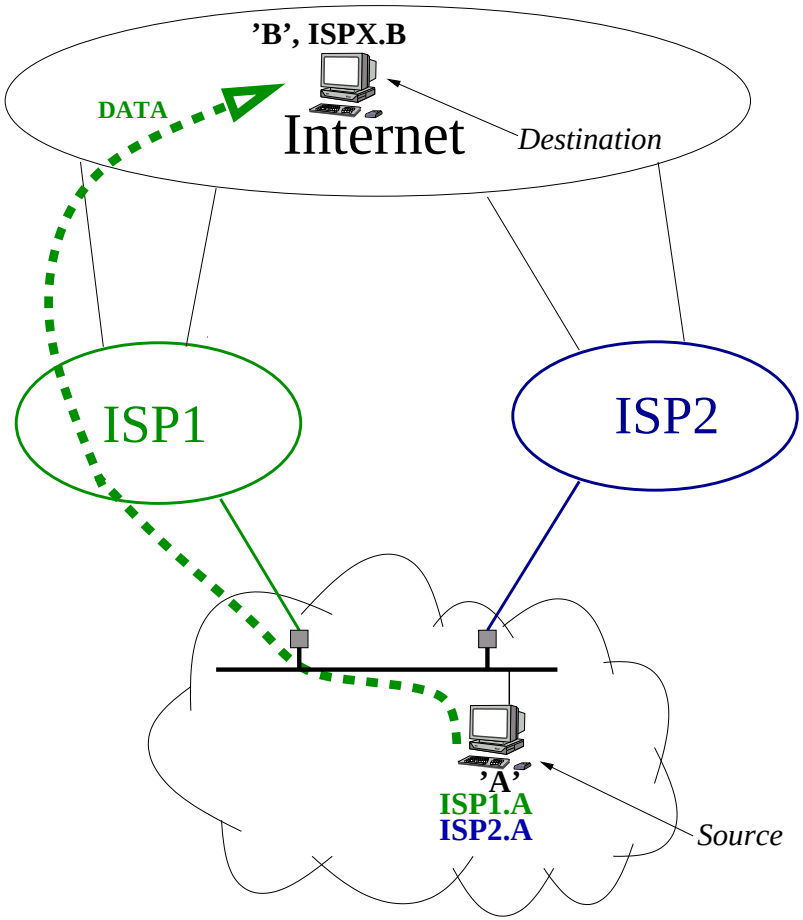
Locators vs Identifiers (ULIDs)



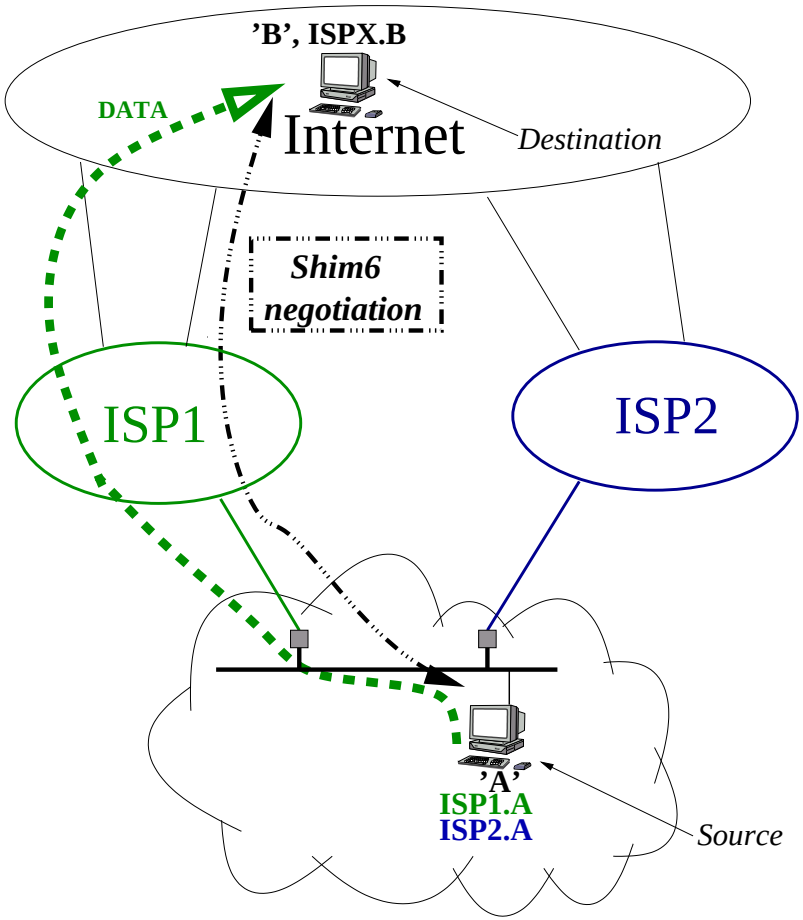
Locators vs Identifiers (ULIDs)

- ULID : Used as the identifier throughout a transport connection.
- locator : IPv6 address used for routing (locating the peer).
- Shim6 performs a mapping between ULIDs and locators, by use of context tags.

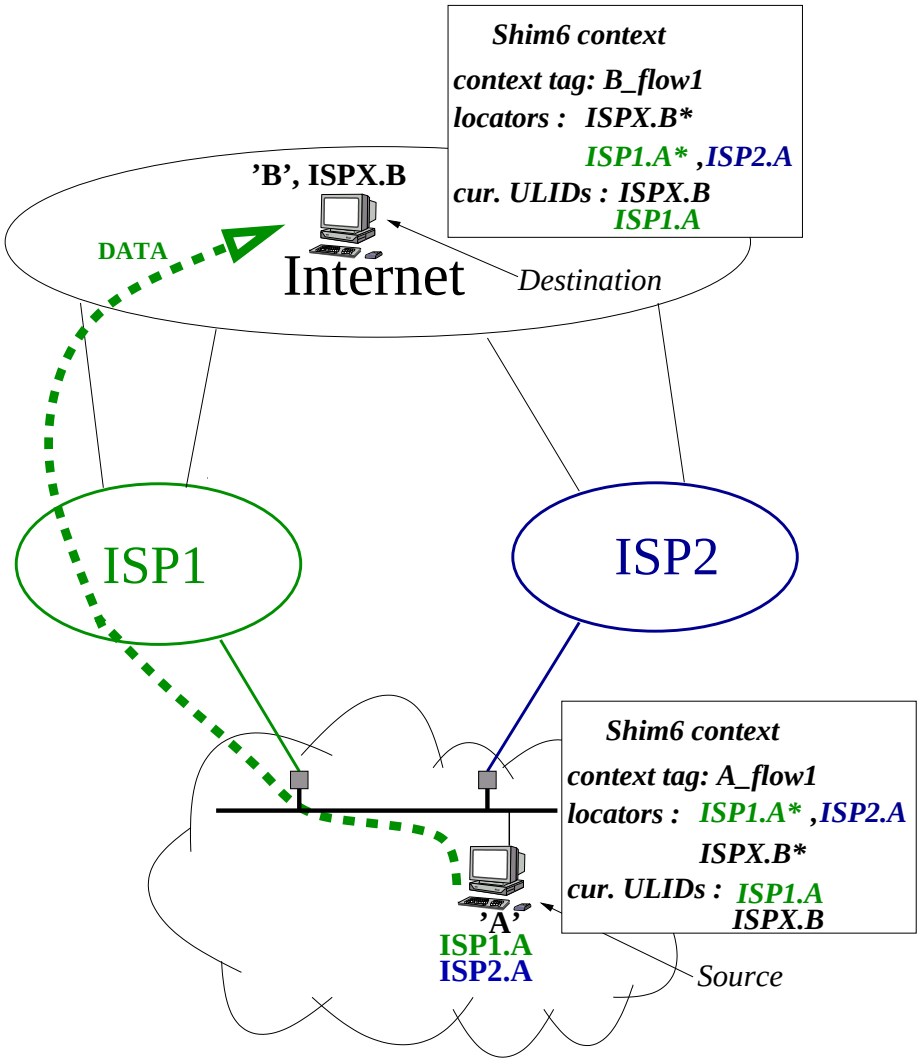
Shim6 operation



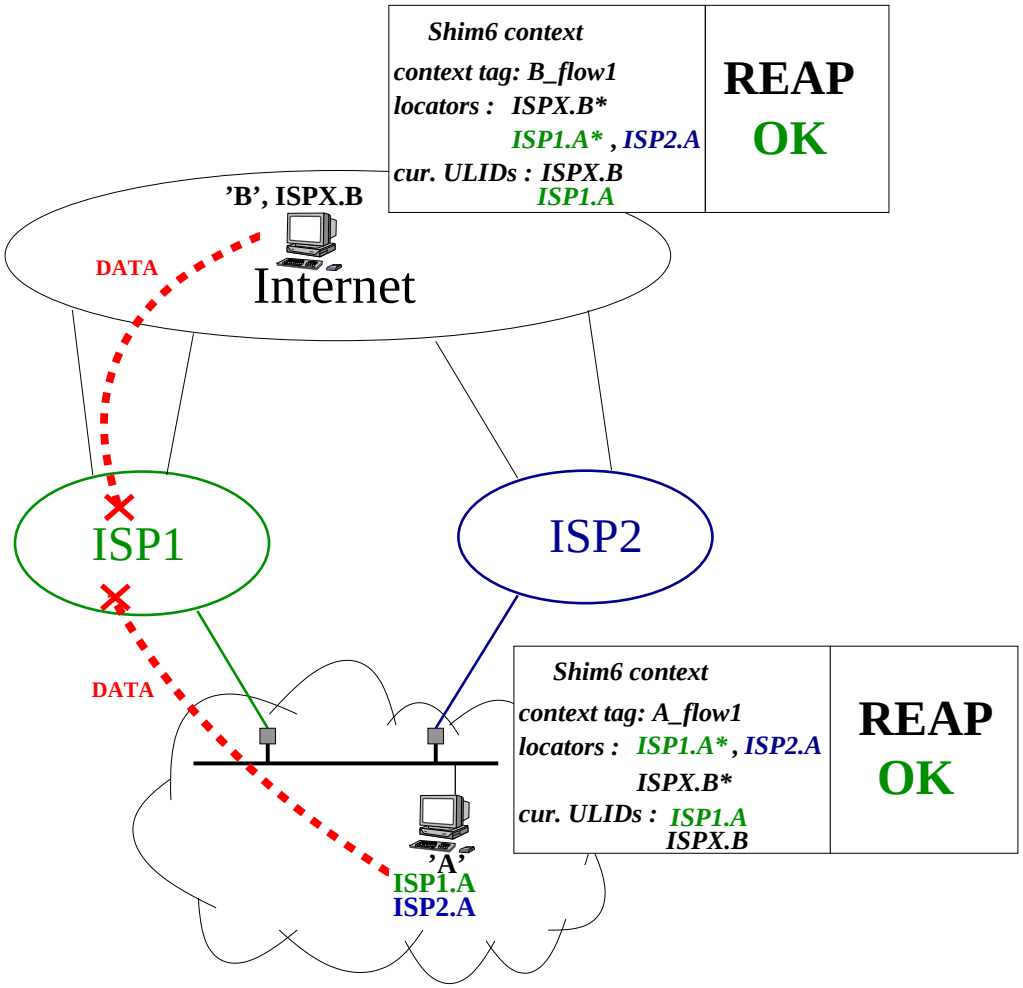
Shim6 operation



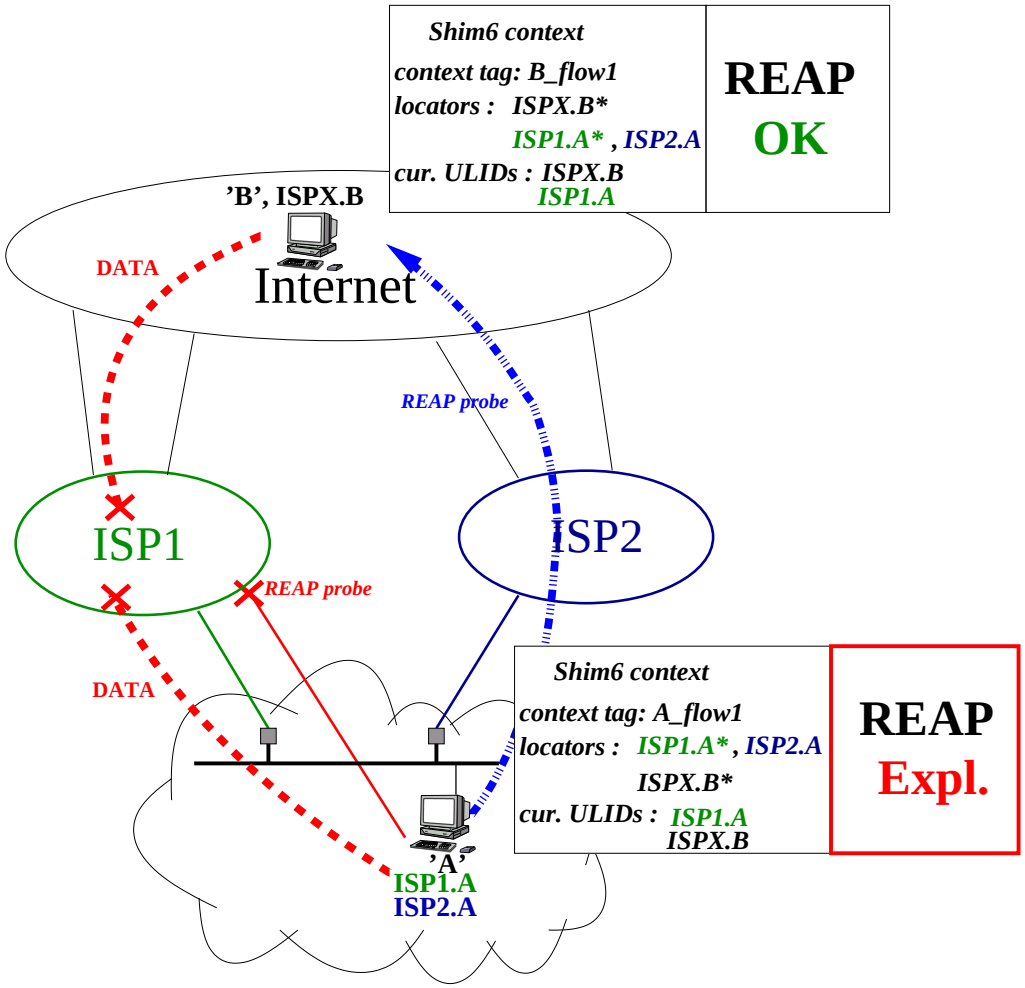
Shim6 operation



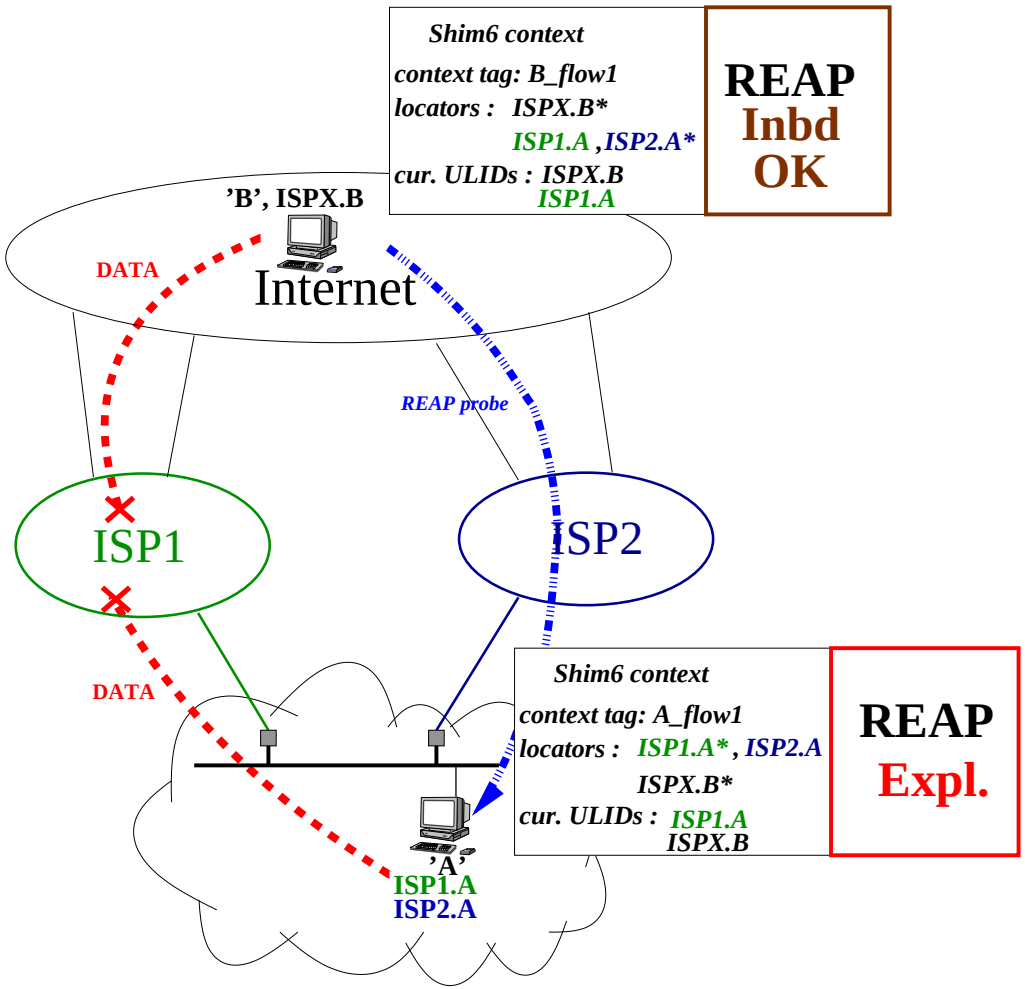
REAP operation



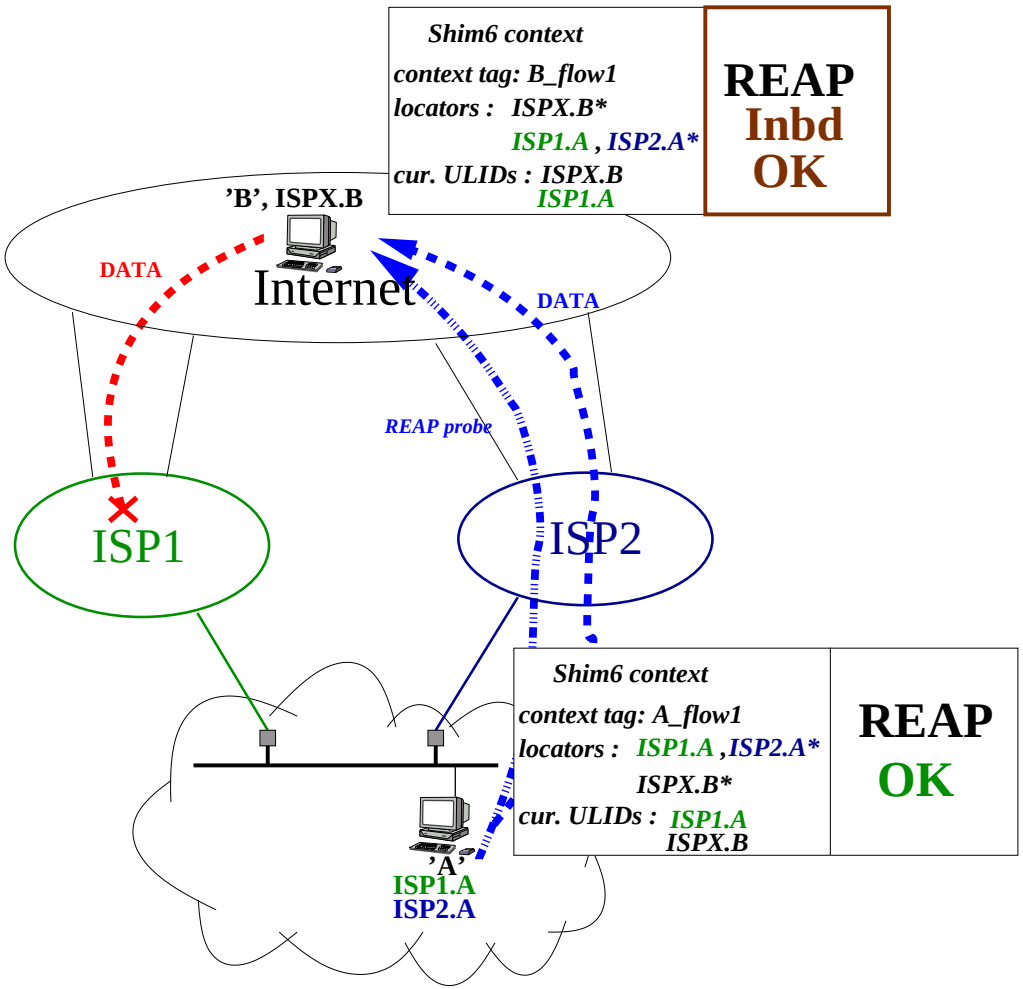
REAP operation



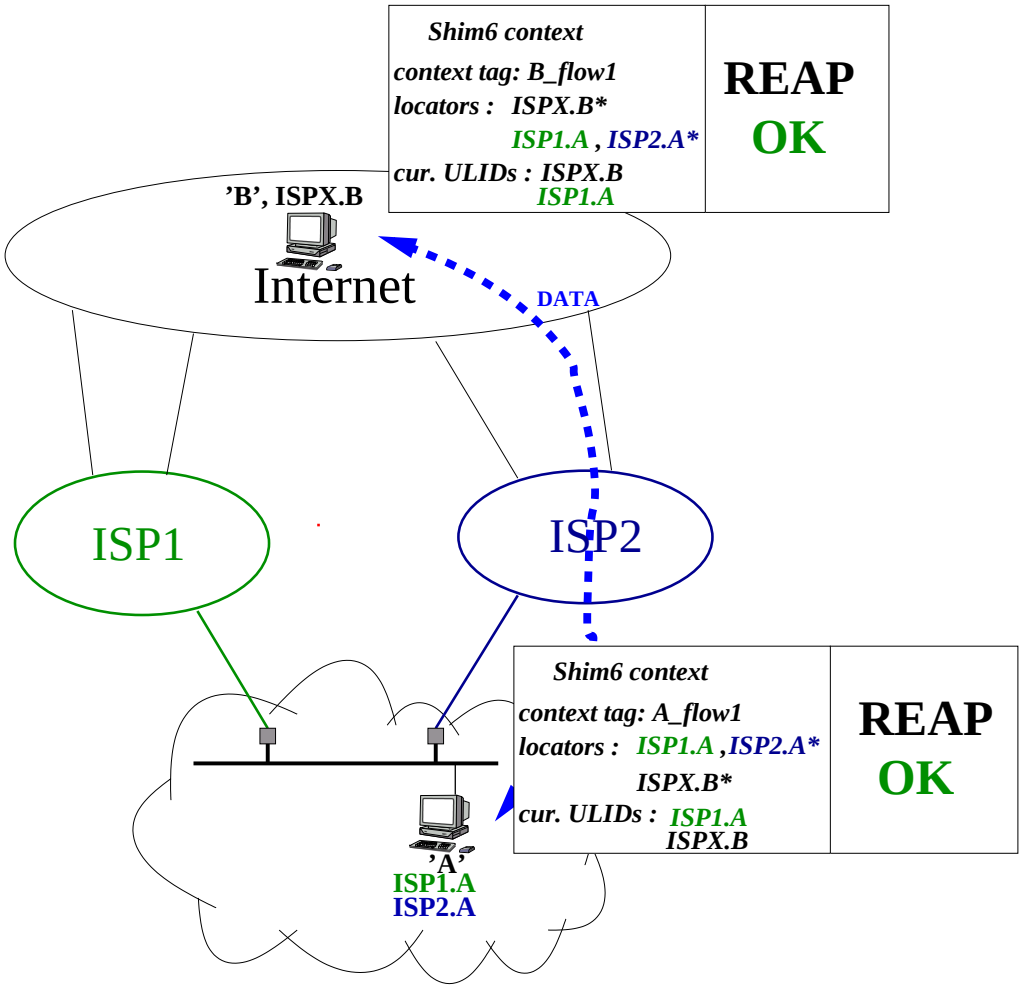
REAP operation



REAP operation



REAP operation



TCP connection survival

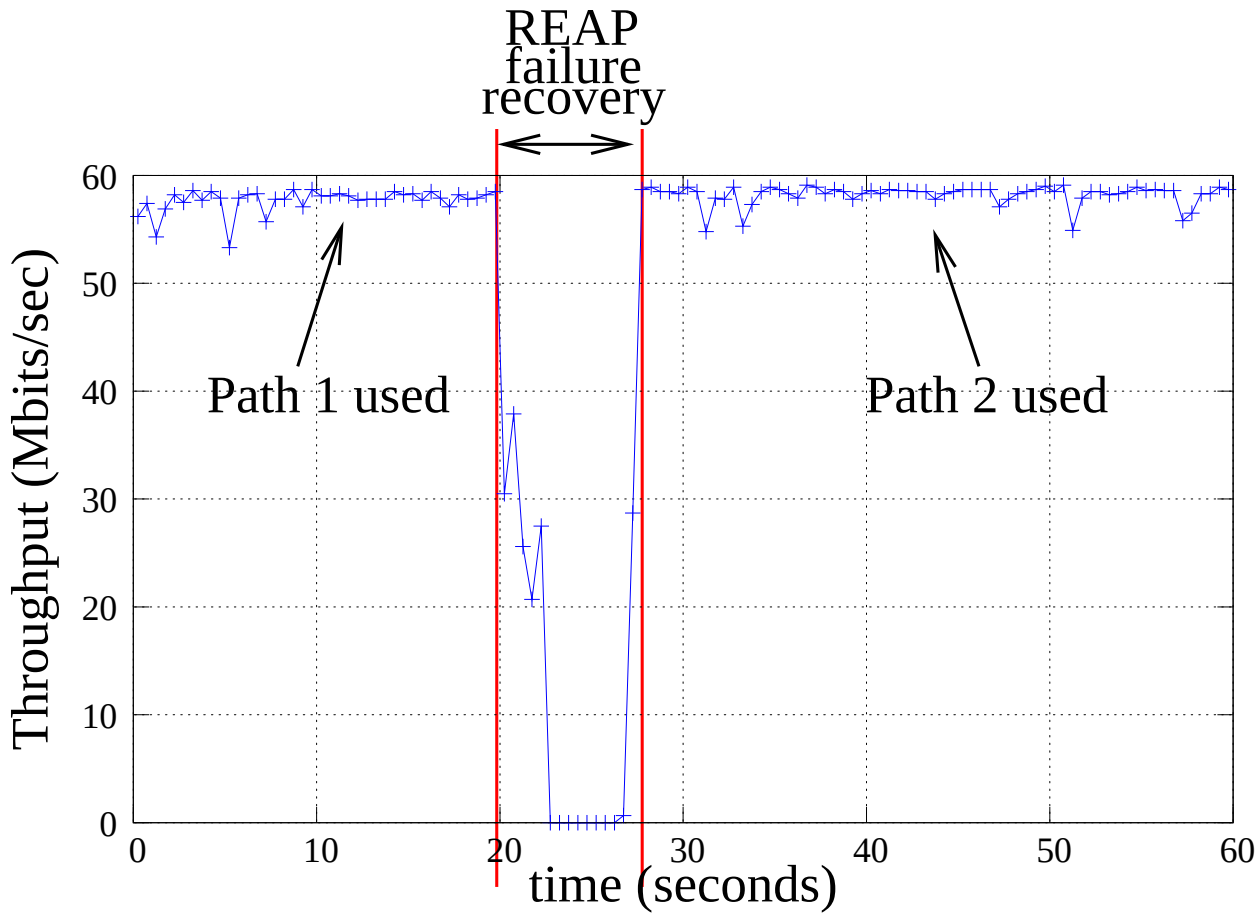


Figure: Evolution of throughput for an iperf TCP session

REAP : Failure recovery time

- The **detection time** is defined as the interval between the occurence of a failure and its detection by REAP timeout.
- The **exploration time** is the interval between leaving and coming back to the operational state.
  - Probes are sent to various paths by randomly selecting an address pair, thus a path.
  - 4 probes are sent with a 500ms interval, then exponential backoff is performed.

Shim6 implementation and lab

- LinShim6 v0.4.3, with Linux kernel 2.6.17.11
  - Available at http://inl.info.ucl.ac.be/LinShim6
- Click modular router, to meet the following needs :
  - Simulate broken paths : we are able to stop or restart any path of the testbed on a per-address pair basis.
  - Simulate path delays : We can configure a specific artificial delay for each address pair.
  - Perform source-address based routing : The testbed is configured with one queue for each address pair.

Lab configuration

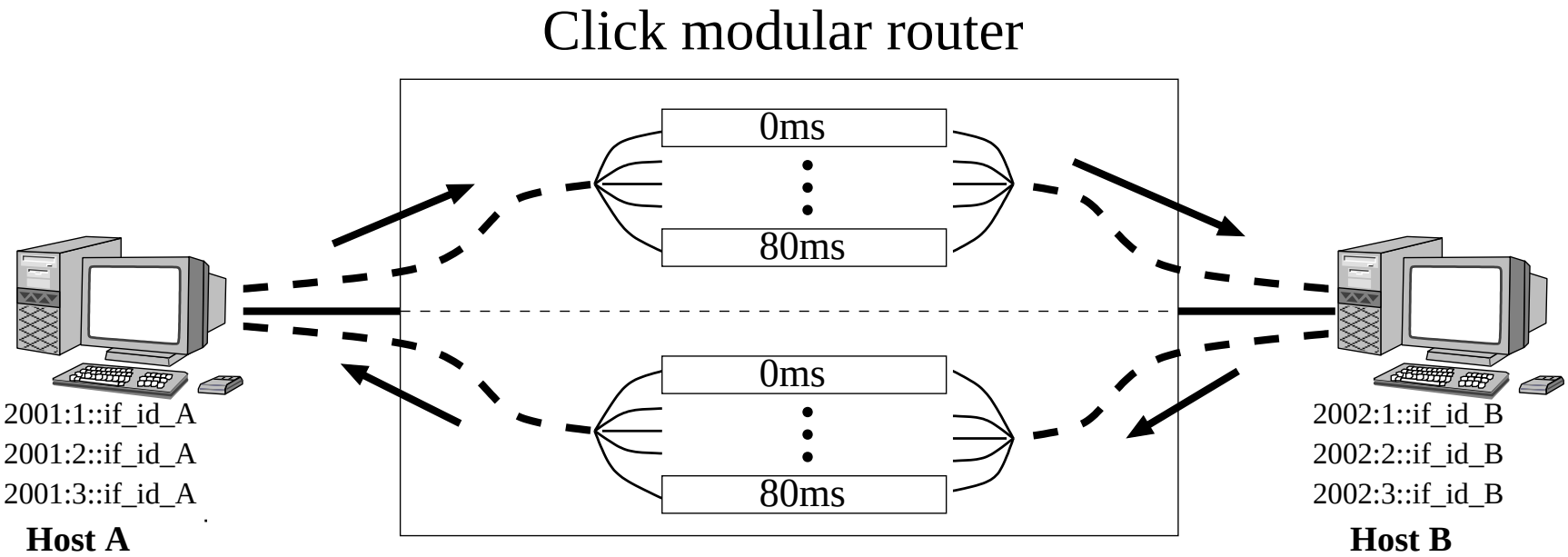
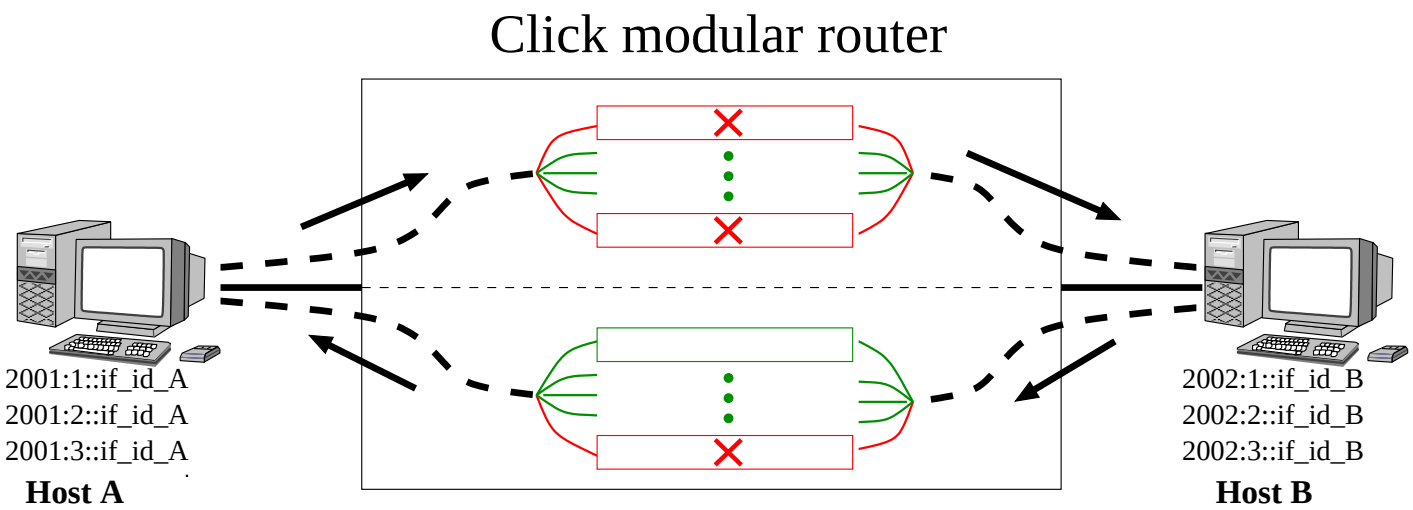


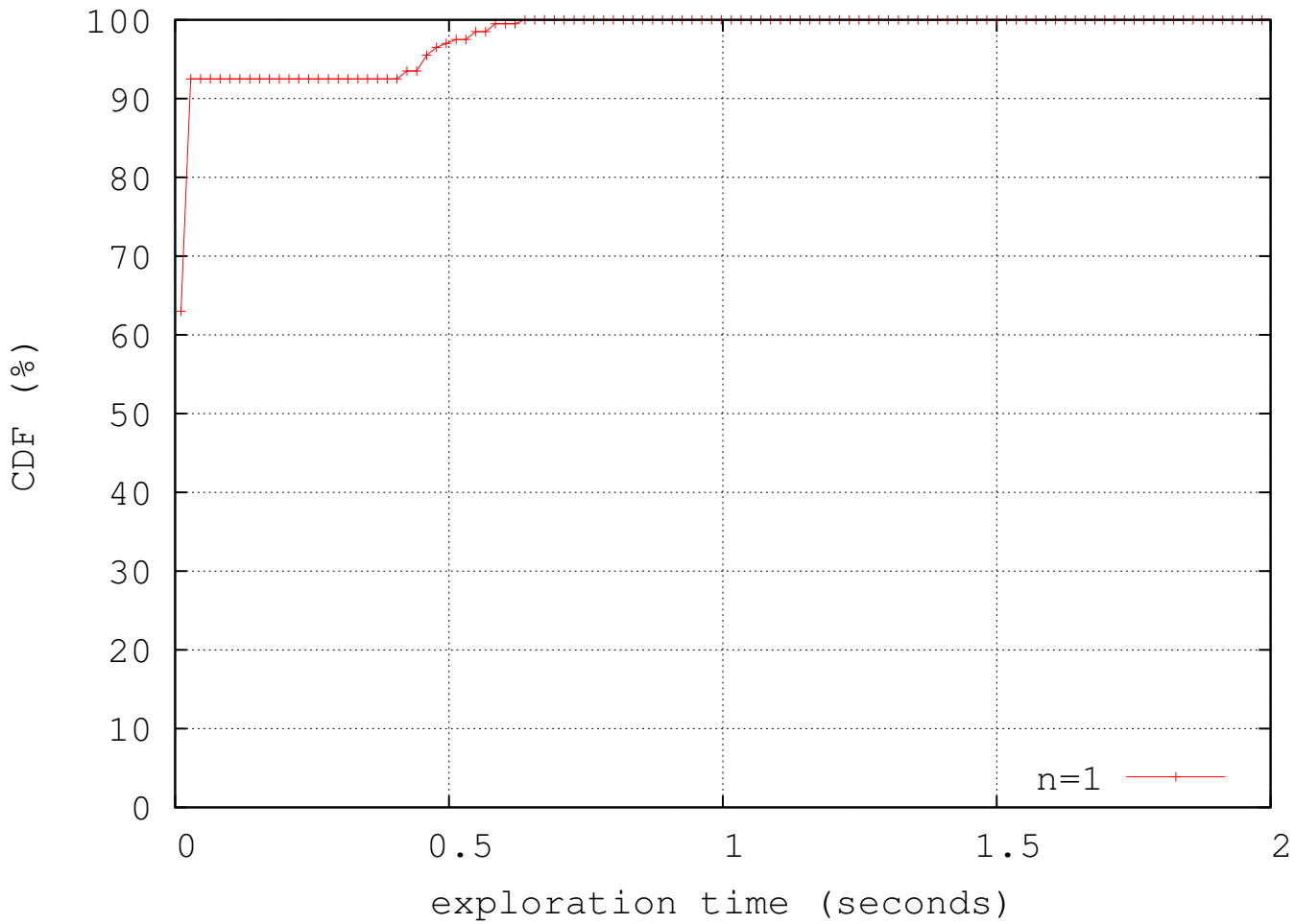
Figure: Testbed for REAP measurements

EX1 : Relation with the number of broken paths

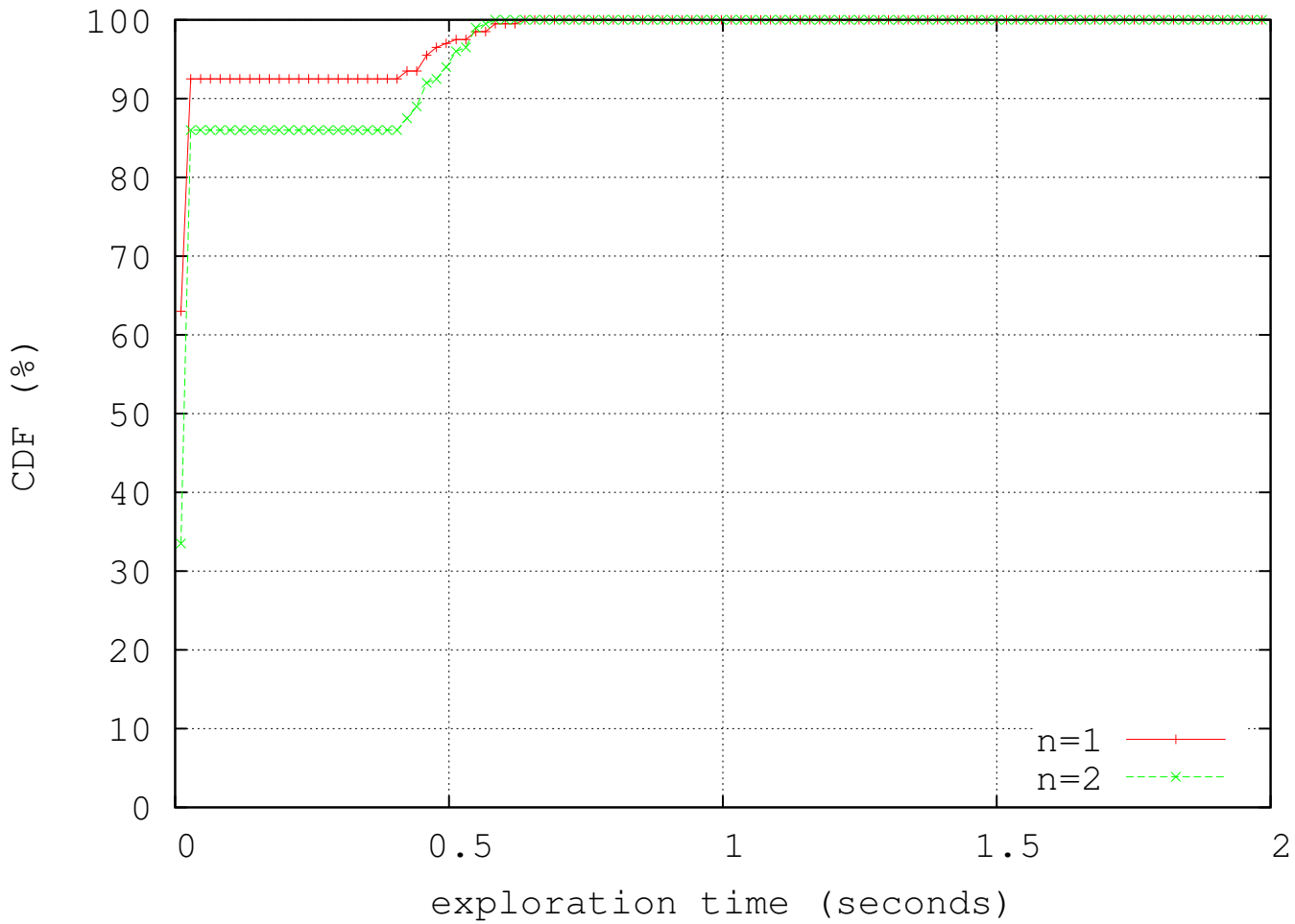
- n : number of broken queues
- The current queues for both directions are always broken (bidirectional failure)
- example for n = 3 :



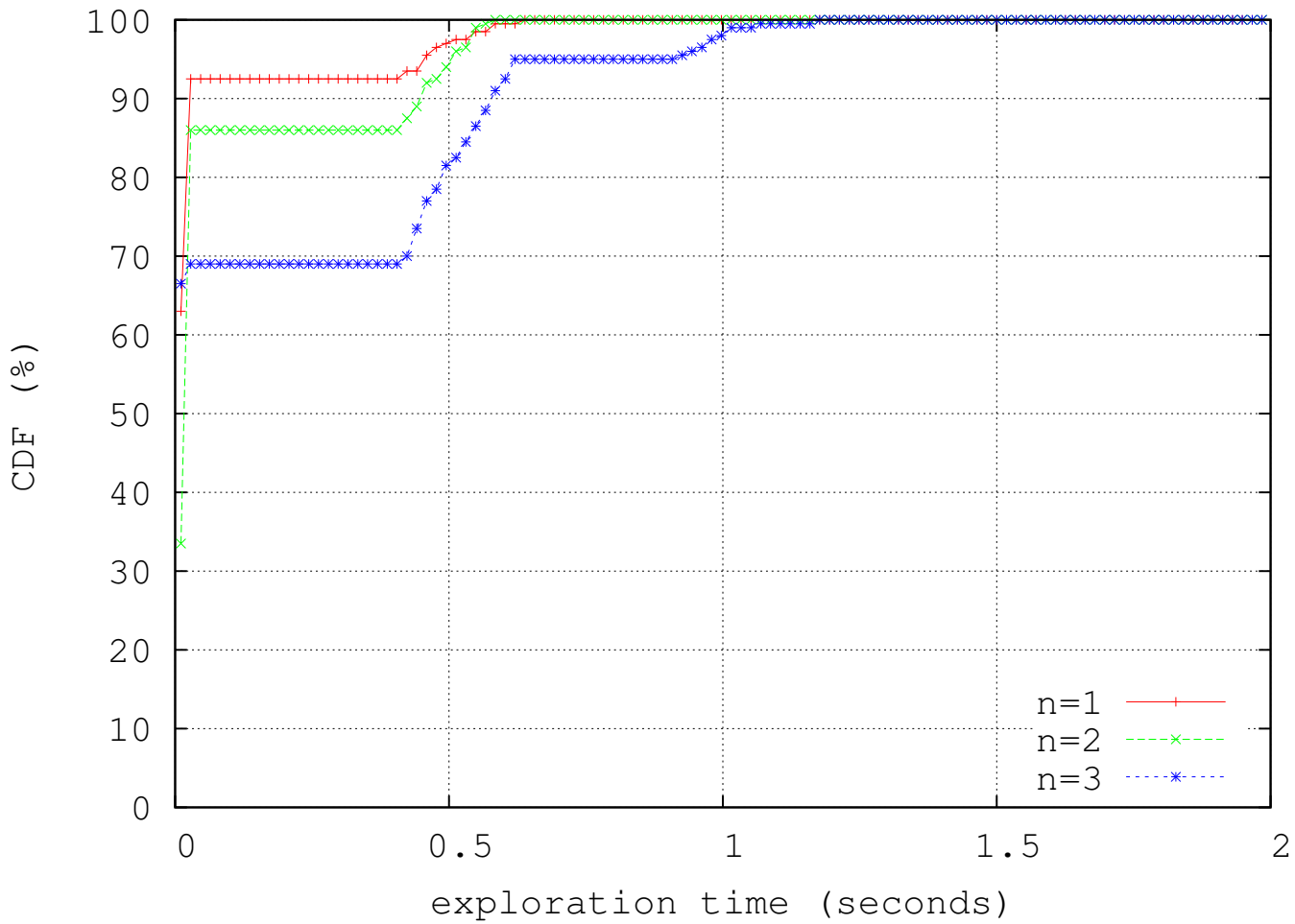
EX1 : Relation with the number of broken paths



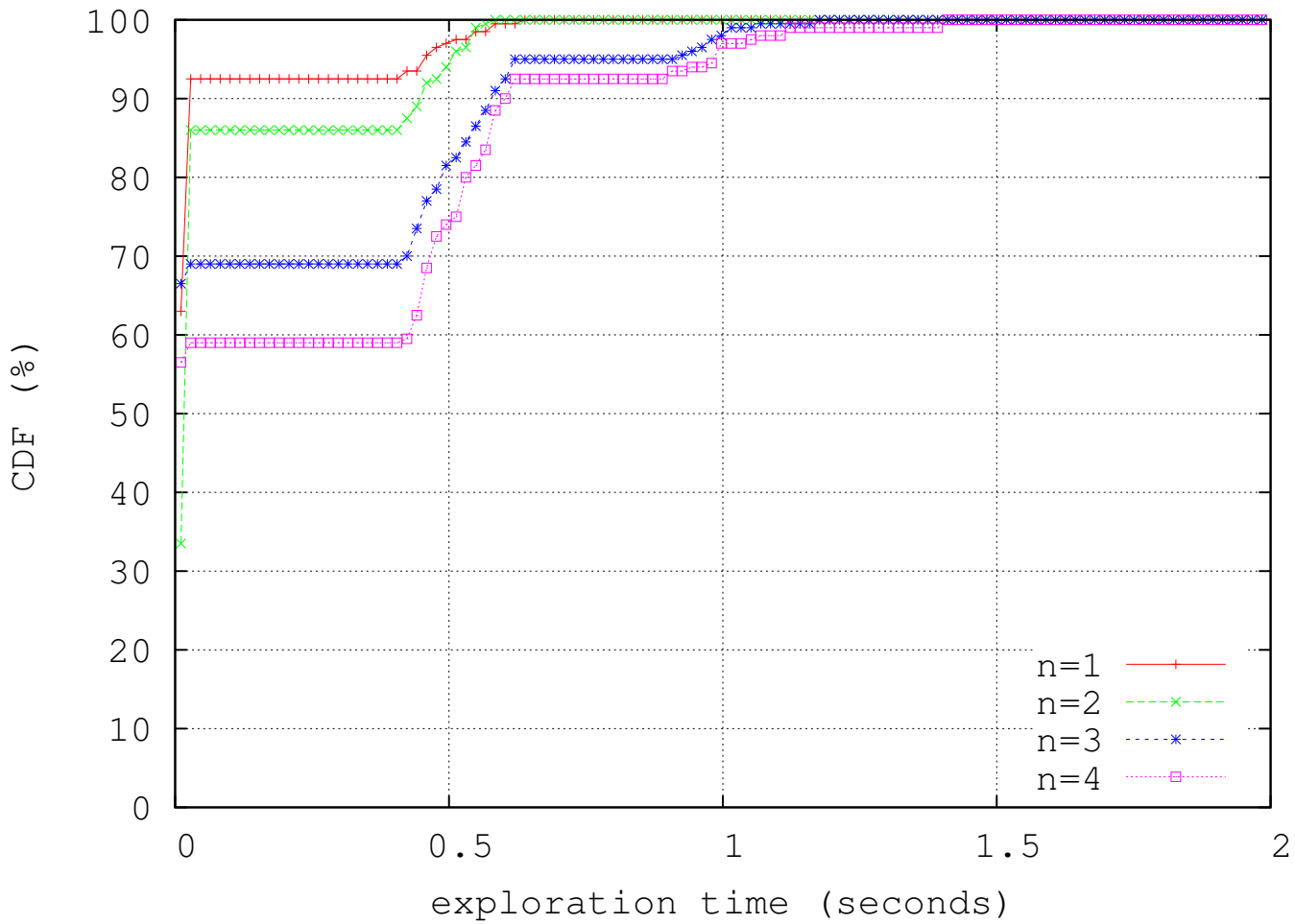
EX1 : Relation with the number of broken paths



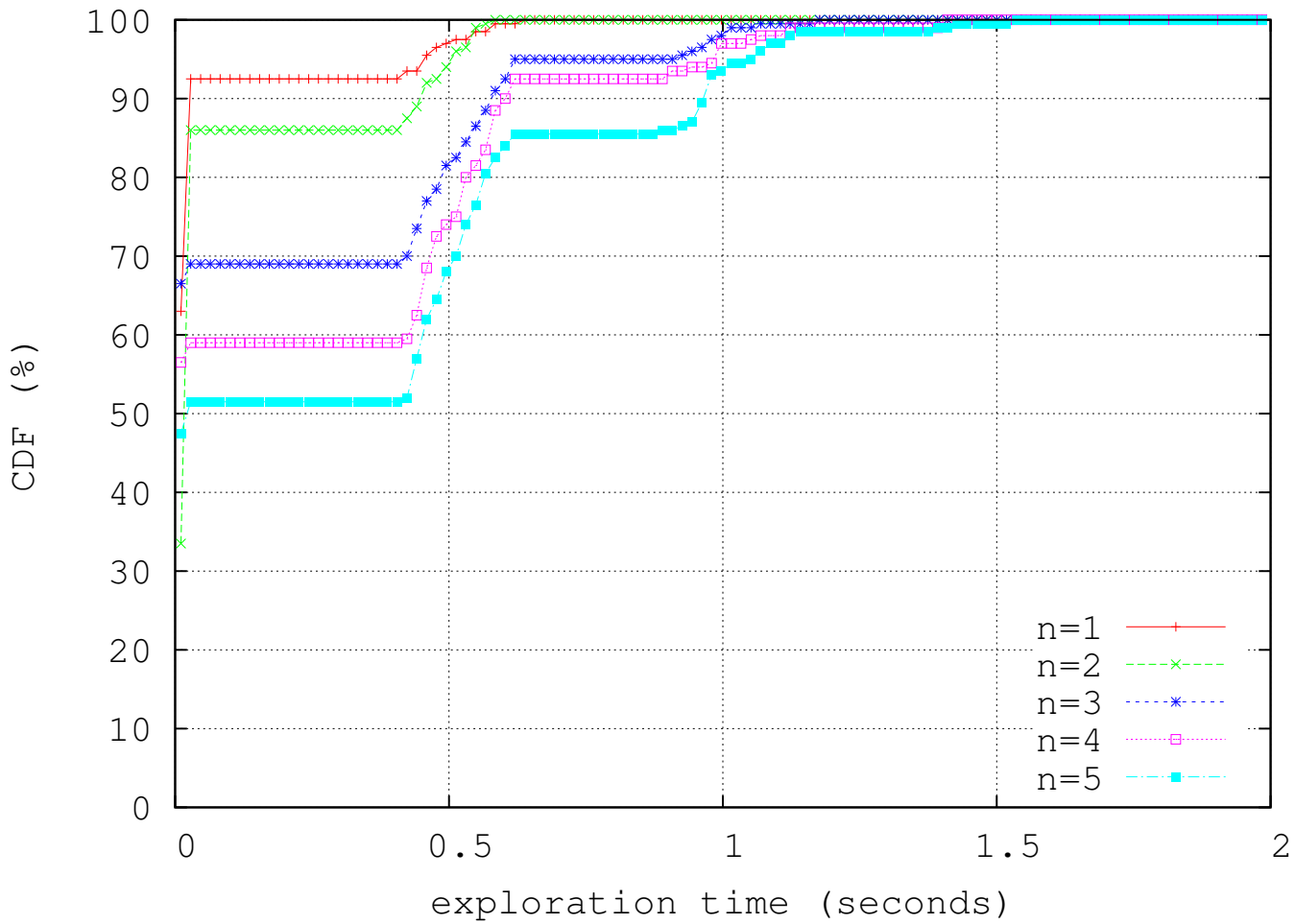
EX1 : Relation with the number of broken paths



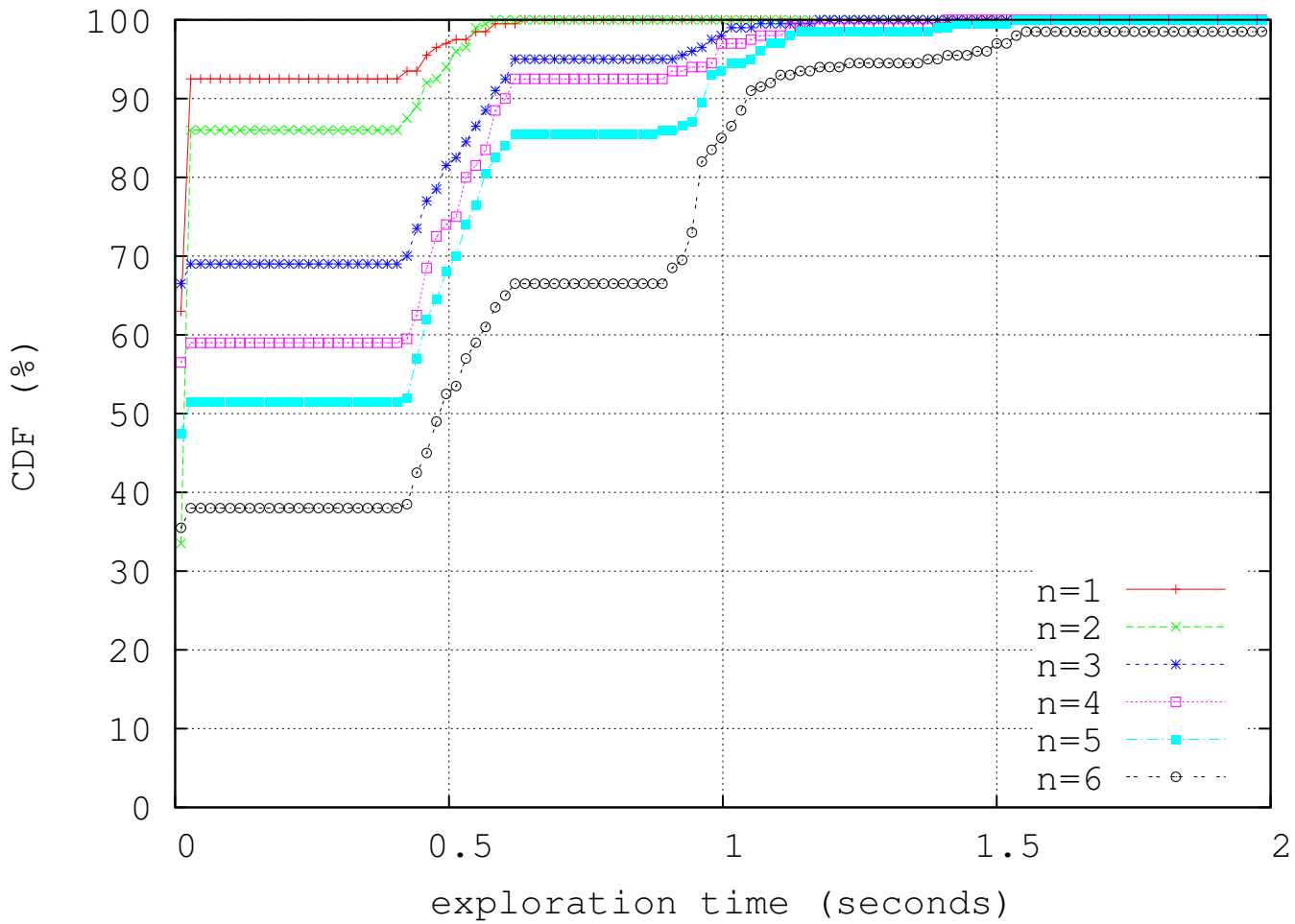
EX1 : Relation with the number of broken paths



EX1 : Relation with the number of broken paths

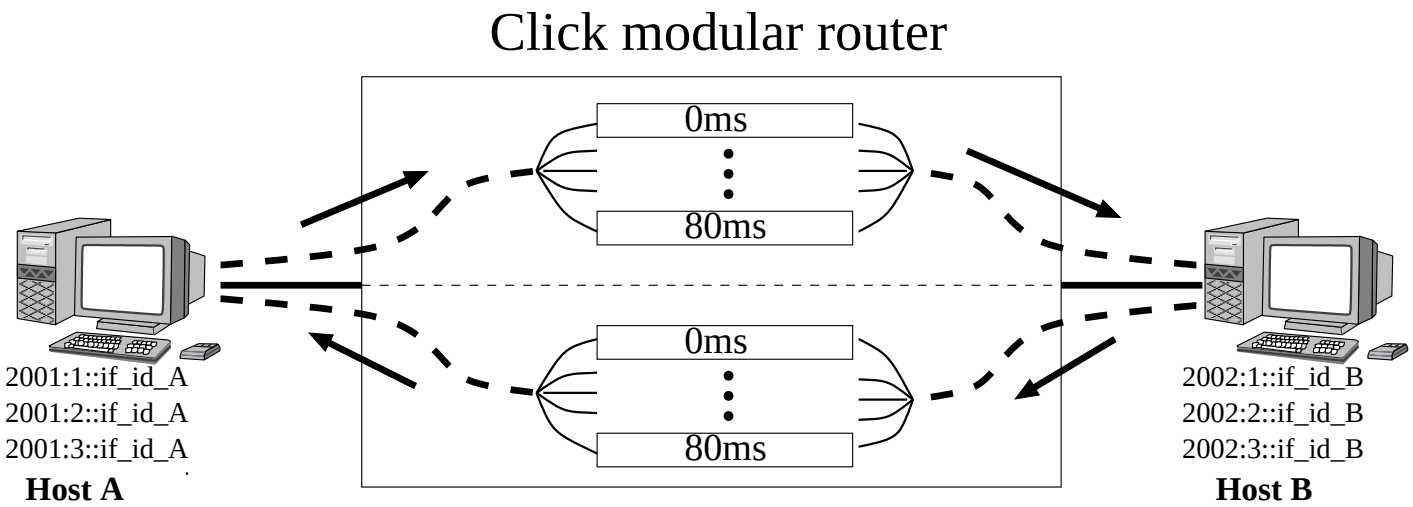


EX1 : Relation with the number of broken paths



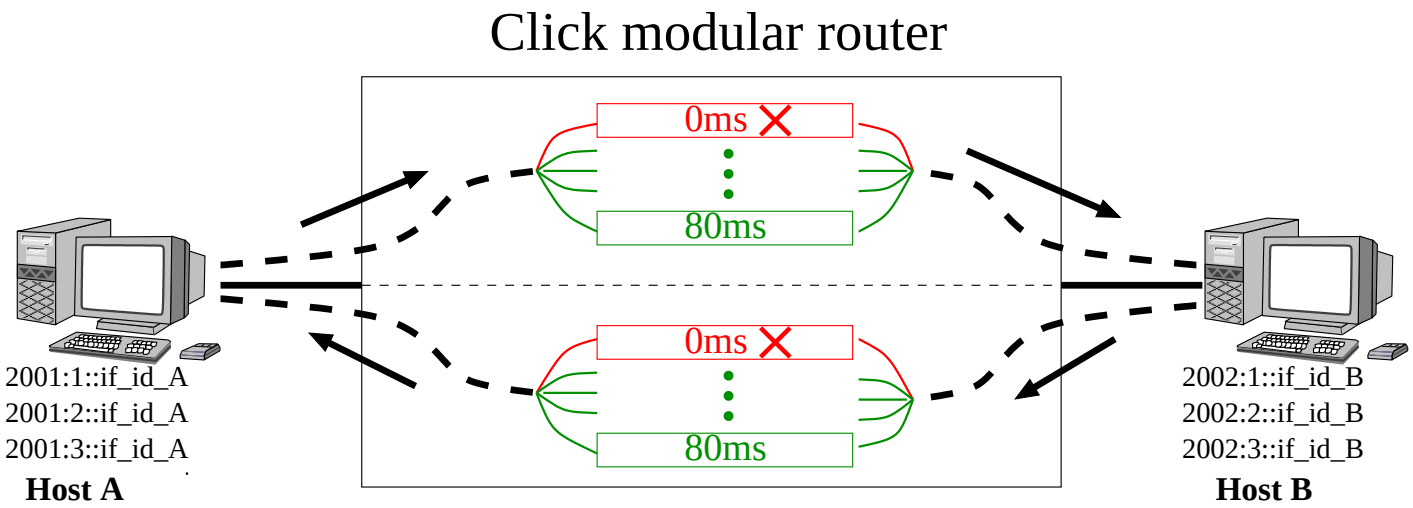
EX2 : Paths with different delays

- We have selected a path, is it the best one ?
- Let's compare in terms of delay.

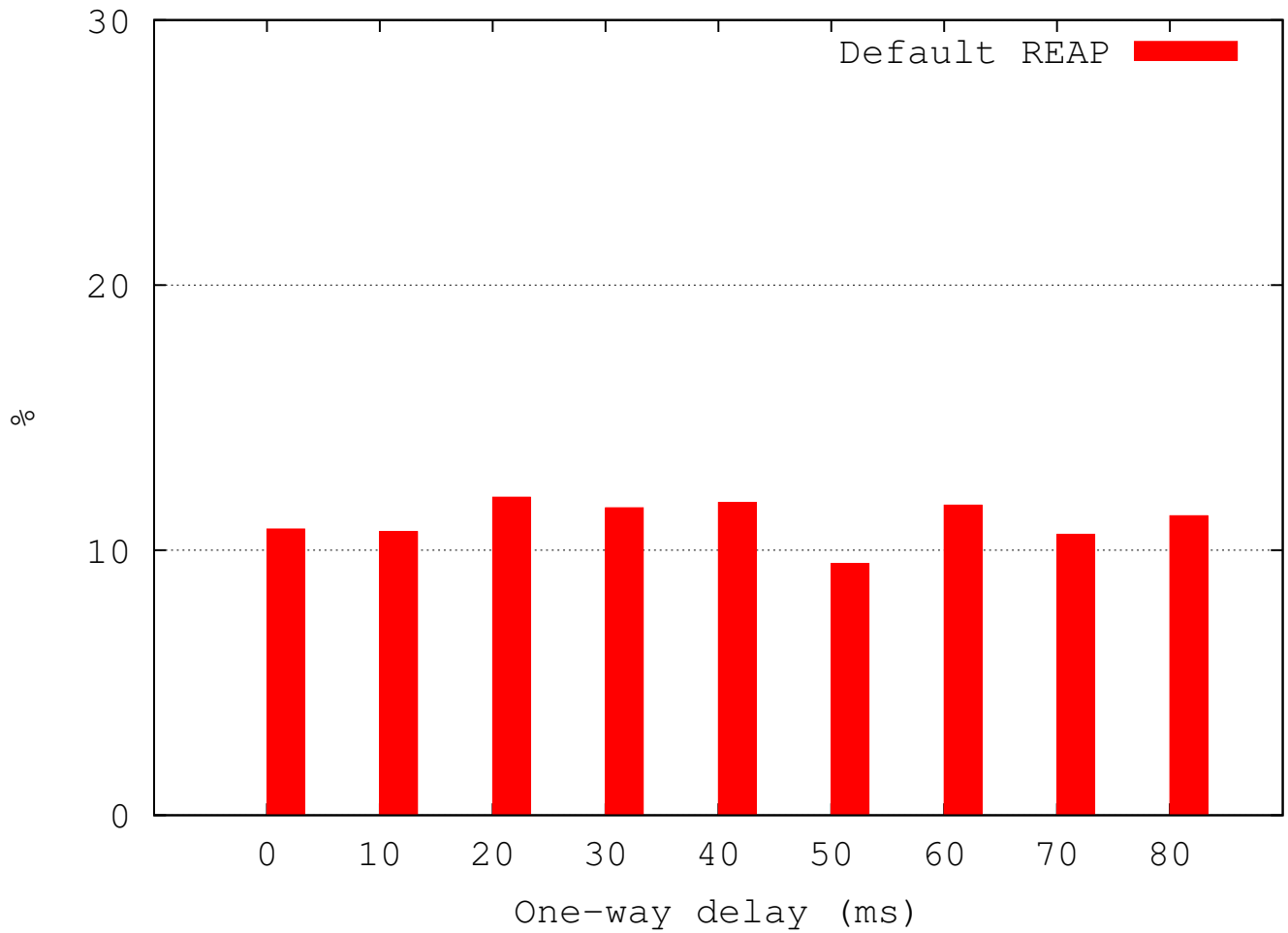


EX2 : Paths with different delays

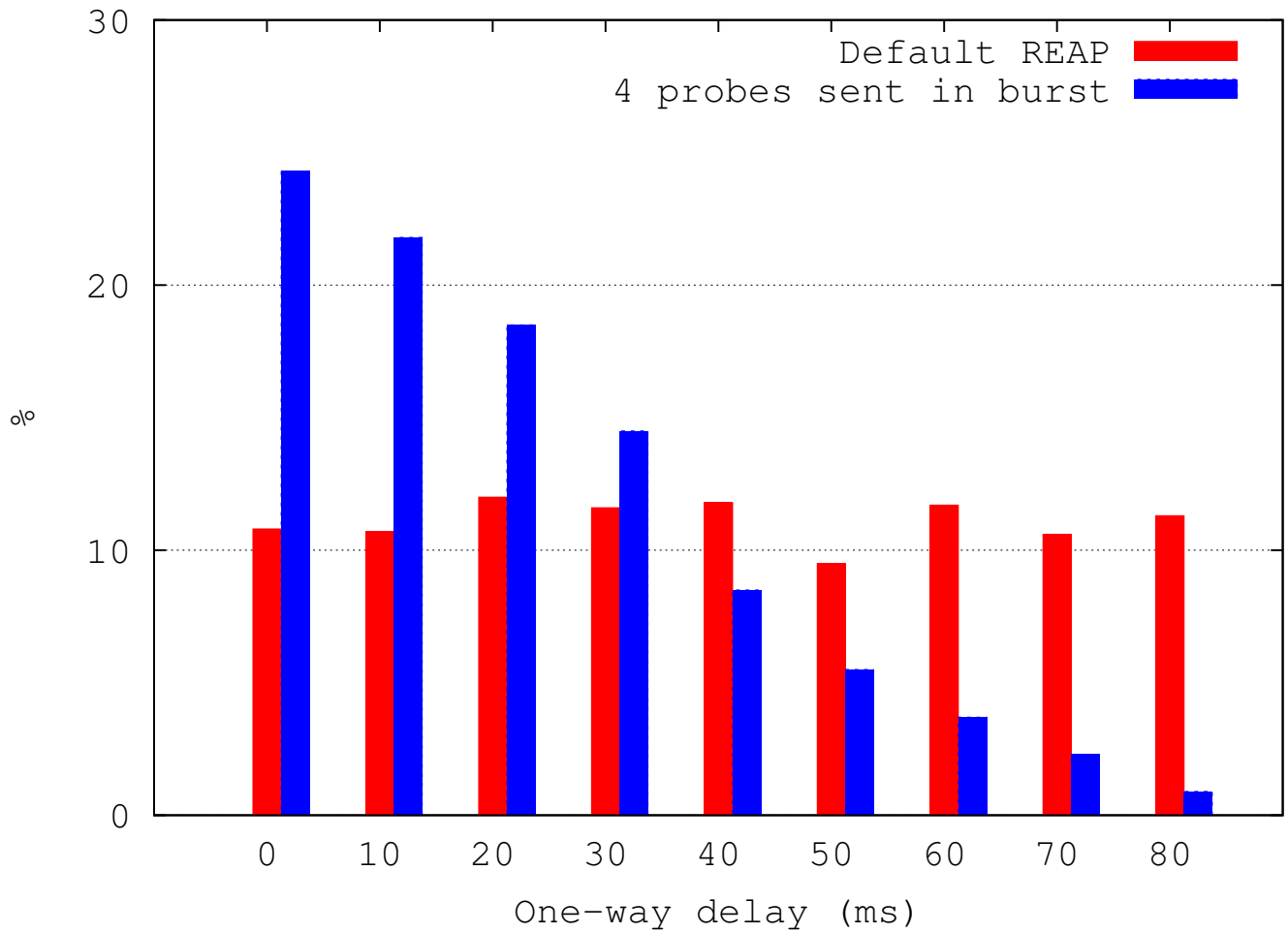
- We break one path for each direction.
- What are the alternative paths chosen by the protocol ?



EX2 : Paths with different delays



EX2 : Paths with different delays



Conclusions

- Our implementation for the Linux kernel shows that multihoming with Shim6 *is* possible.
- REAP provides an additional capability, that is, failure detection and recovery by timeouts and probe sending.
- In most cases the path exploration may be done reasonably fast.
- We argue that REAP could also be used for finding paths with better/best delays, although we should care not to overload the network with signalling load.

Future work

- Evaluation and optimization of the detection time
- Use heuristics to find better paths according to some criteria (delay, bandwidth,...)
- Find an ordering of address pairs for path exploration that would perform better than the current random selection.

Thanks !

Questions ?