On-Demand Blind Packet Forwarding

30 September 2021

Irfan Simsek

Networking Technology Group

Institute for Computer Science &
Business Information Systems
University of Duisburg-Essen

Agenda

- Motivation
- Blind Packet Forwarding (BPF)
- Selective Masked Routing
- Fully BPF On Demand
- Implementation and evaluation
- Conclusion

Motivation

- Network Address Confidentiality (NAC)
  - classifies all third parties and nodes as adversaries and limits access to packet addresses in cleartext exclusively to communicating endpoints
  - → Sender/recipient and relationship unlinkability
    - Packets can not be linked to source/destination and to communicating endpoints
- Blind Packet Forwarding (BPF)
  - realizes NAC and its unlinkability properties
  - redesigns packet forwarding and its associated network functions to blind ones transferring and processing packet addresses in end-to-end encrypted form
  - builds on Locator/Identifier (Loc/ID) Split
  - separately masks identifiers and locators
- Semi-BPF masks only identifiers
  - NAC and its unlinkability properties apply only to identifiers and communicating endpoints
- Fully BPF masks both locators and identifiers
  - NAC and its unlinkability properties apply to both parts of addresses and communicating endpoints as well as domains and local networks
  - **Issue:** Full blindness requires to set up and maintain masked routing tables in entire domains → Costly process
  - **Idea:** Only nodes on the route between two communicating endpoints need to maintain according masked routing table entries

BPF – Public key Encryption with Keyword Search (PEKS)

$$KeyGen \rightarrow \left(A_{pub}, A_{priv}\right)$$

$$PEKS(A_{pub}, W) \rightarrow E(W)$$

$$Trapdoor(A_{priv}, V) \rightarrow T(V)$$

$$Test\big(E(W), T(V)\big) \rightarrow 1 \Leftrightarrow W = V, \rightarrow 0 \; otherwise$$

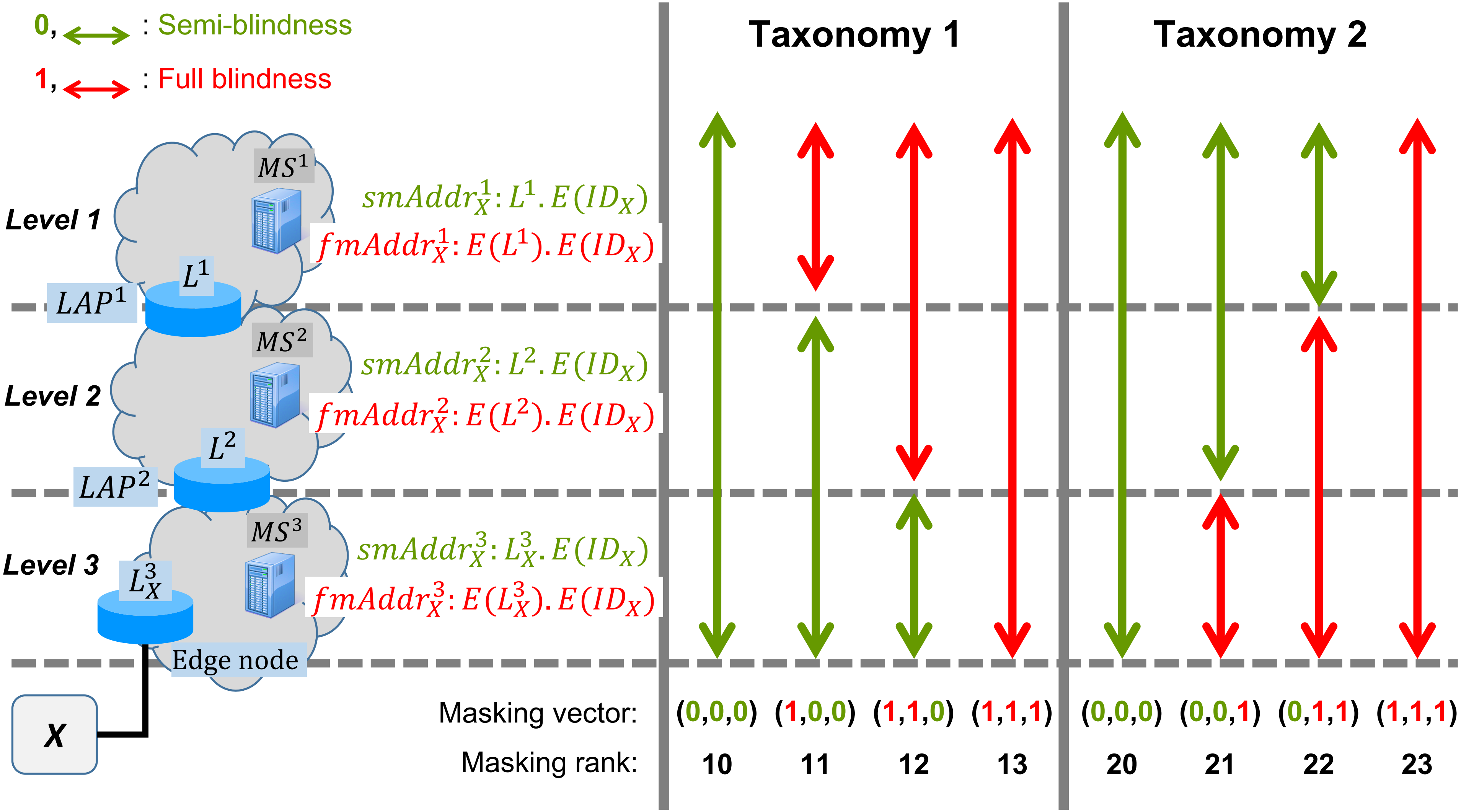
- PEKS enables to correctly determine for two ciphertexts whether their cleartext values are the same, without decrypting the ciphertexts.

- PEKS encryption function is not deterministic

BPF – Architecture & Blindness Taxonomies

0, ⟷ : Semi-blindness

1, ⟷ : Full blindness



$smAddr_X^1: L^1.E(ID_X)$

$fmAddr_X^1: E(L^1).E(ID_X)$

$smAddr_X^2: L^2.E(ID_X)$

$fmAddr_X^2: E(L^2).E(ID_X)$

$smAddr_X^3: L_X^3.E(ID_X)$

$fmAddr_X^3: E(L_X^3).E(ID_X)$

	Taxonomy 1				Taxonomy 2			
Masking vector:	(0,0,0)	(1,0,0)	(1,1,0)	(1,1,1)	(0,0,0)	(0,0,1)	(0,1,1)	(1,1,1)
Masking rank:	10	11	12	13	20	21	22	23

BPF – Masked Routing & Packet Forwarding

$Test(E(N_i),T(N_j))$
for each entry i and j
in table and update msg

mLoc., Trapd.	Port	Distance
$E(N_1),T(N_1)$	0	0
$E(N_2),T(N_2)$	1	1
$E(N_3),T(N_3)$	1	2

$Test(E(N_3),T(N_i))$
for each entry i

$Test(E(N_3),T(N_3)) \rightarrow 1$

↔

$Test(E(N_i),T(N_j))$
for each entry i and j
in table and update msg

mLoc., Trapd.	Port	Distance
$E(N_2),T(N_2)$	0	0
$E(N_1),T(N_1)$	1	1
$E(N_3),T(N_3)$	2	1

$Test(E(N_3),T(N_i))$
for each entry i

$Test(E(N_3),T(N_3)) \rightarrow 1$

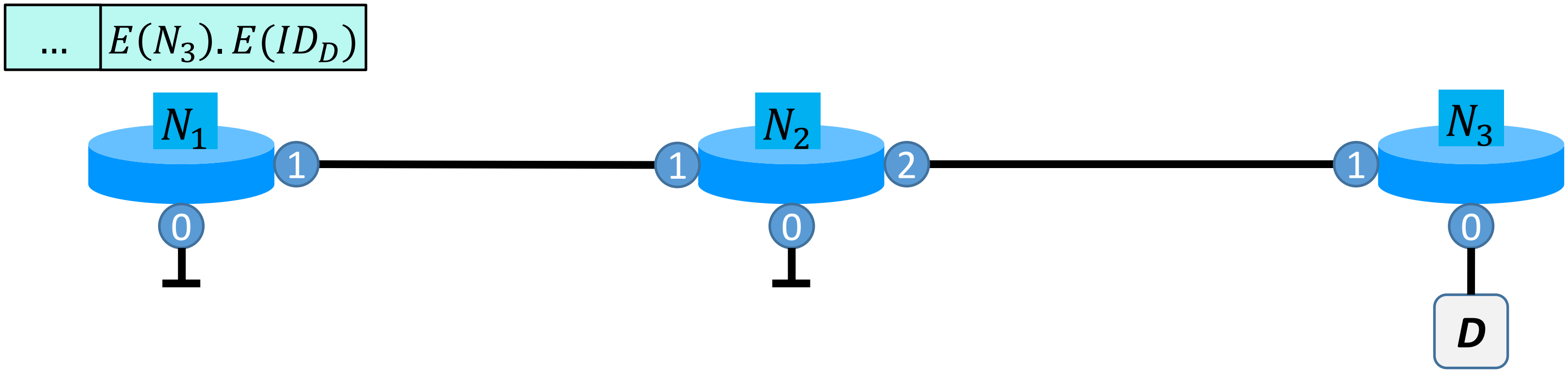
↔

$Test(E(N_i),T(N_j))$
for each entry i and j
in table and update msg

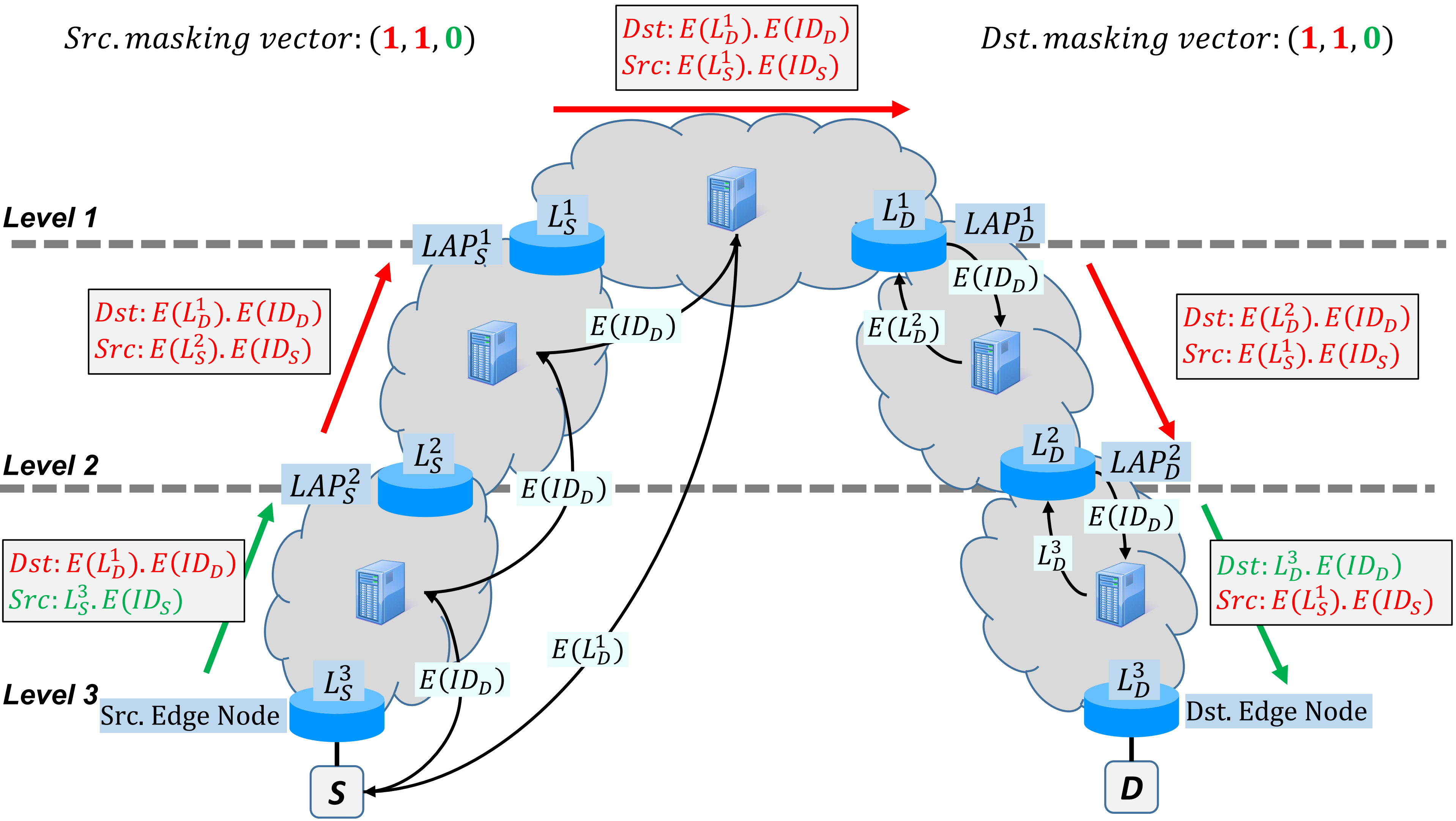
mLoc., Trapd.	Port	Distance
$E(N_3),T(N_3)$	0	0
$E(N_2),T(N_2)$	1	1
$E(N_2),T(N_2)$	1	2

$Test(E(N_3),T(N_i))$
for each entry i

$Test(E(N_3),T(N_3)) \rightarrow 1$



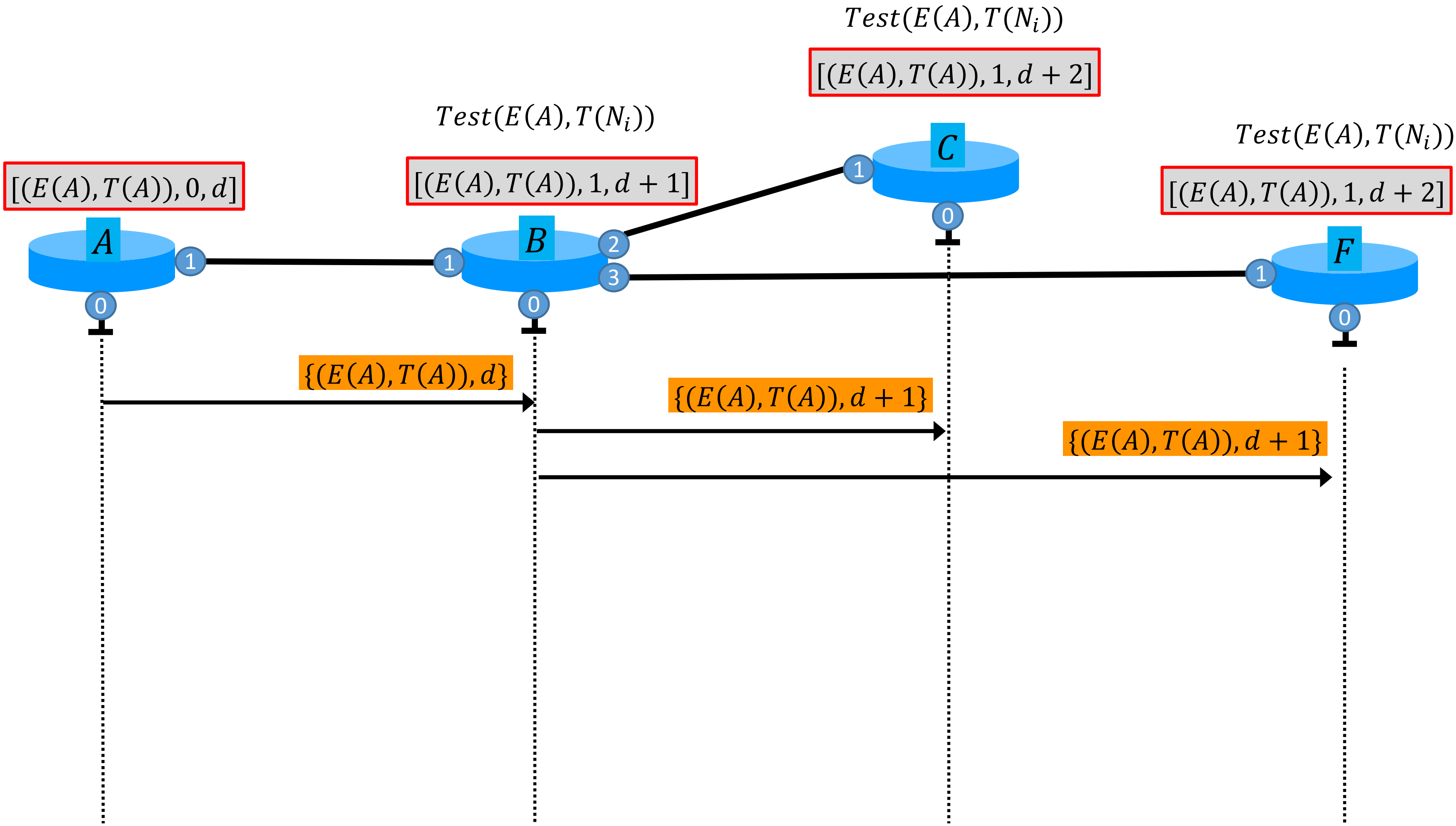
BPF – Packet Delivery



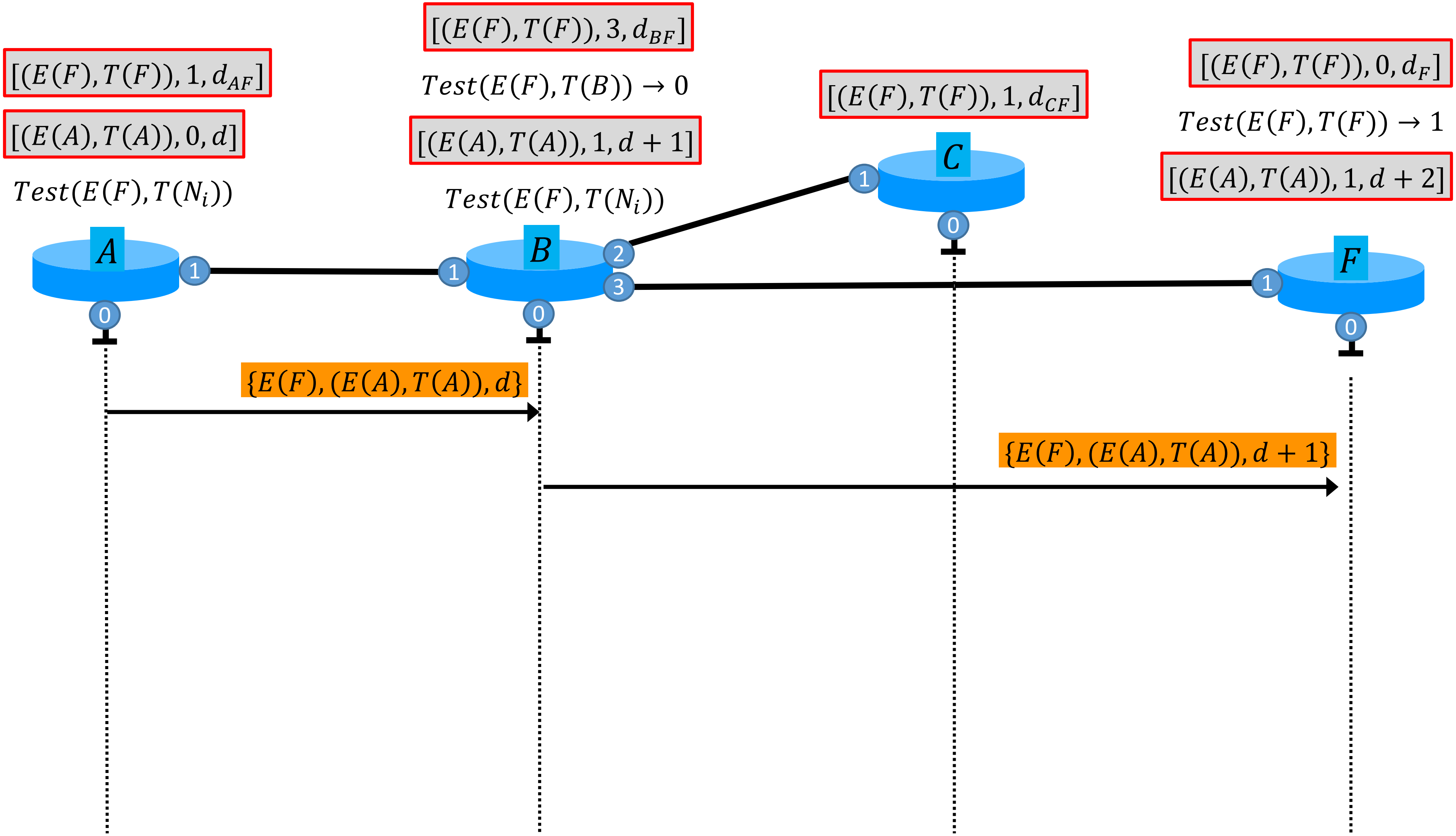
$Src.masking\ vector: (1, 1, 0)$

$Dst.masking\ vector: (1, 1, 0)$

Selective Masked Routing – Case 1


$Test(E(A), T(N_i))$
$[(E(A), T(A)), 1, d + 2]$
$Test(E(A), T(N_i))$
$[(E(A), T(A)), 1, d + 1]$
$Test(E(A), T(N_i))$
$[(E(A), T(A)), 0, d]$
$[(E(A), T(A)), 1, d + 2]$
A
B
C
F
1
1
2
3
1
1
0
0
0
0
$\{(E(A), T(A)), d\}$
$\{(E(A), T(A)), d + 1\}$
$\{(E(A), T(A)), d + 1\}$

Selective Masked Routing – Case 2



$[(E(F),T(F)),1,d_{AF}]$

$[(E(A),T(A)),0,d]$

$Test(E(F),T(N_i))$

$[(E(F),T(F)),3,d_{BF}]$

$Test(E(F),T(B)) \rightarrow 0$

$[(E(A),T(A)),1,d+1]$

$Test(E(F),T(N_i))$

$[(E(F),T(F)),1,d_{CF}]$

$[(E(F),T(F)),0,d_F]$

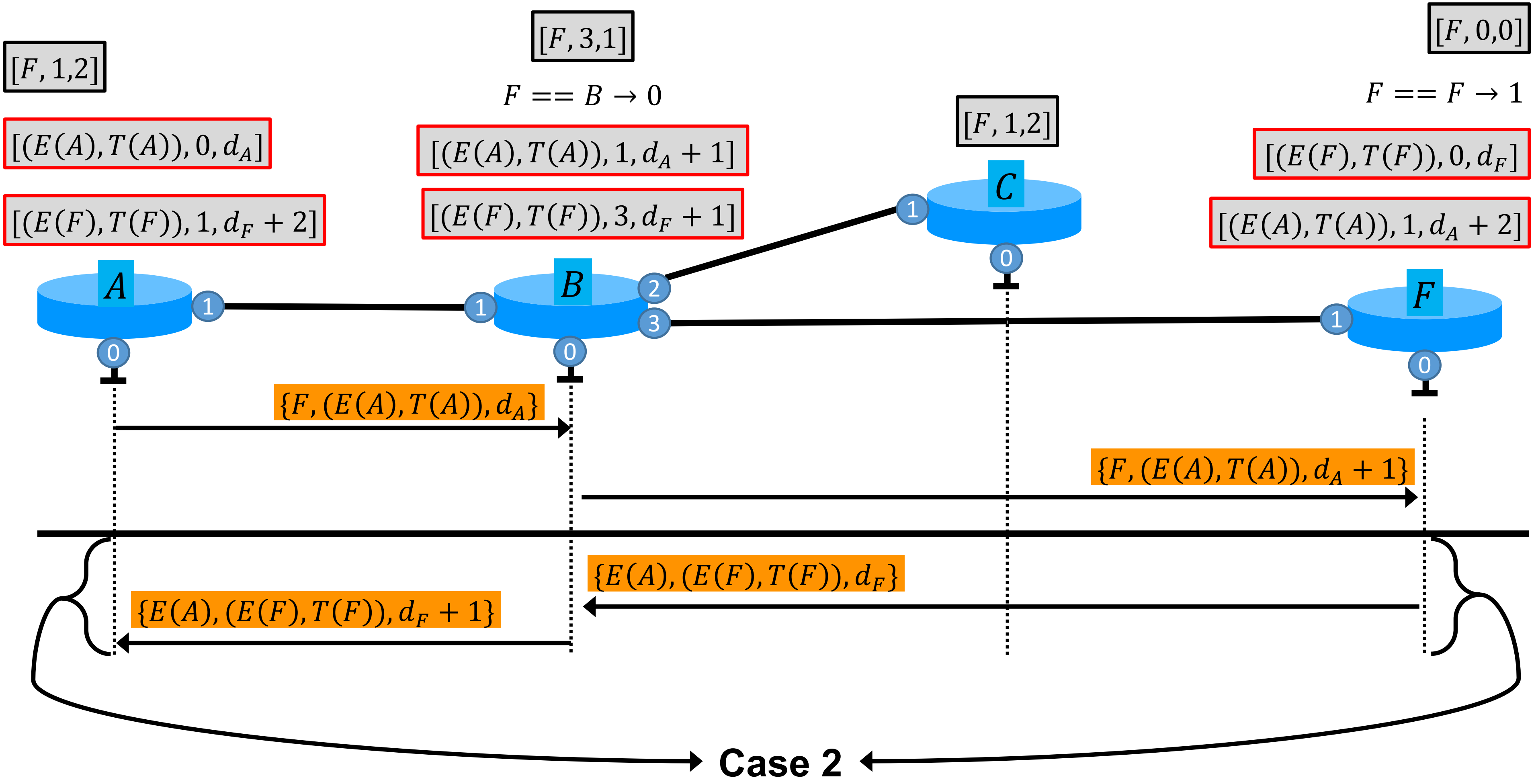
$Test(E(F),T(F)) \rightarrow 1$

$[(E(A),T(A)),1,d+2]$

$\{E(F),(E(A),T(A)),d\}$

$\{E(F),(E(A),T(A)),d+1\}$

Selective Masked Routing – Case 3



$[F, 1{,}2]$

$[(E(A), T(A)), 0, d_A]$

$[(E(F), T(F)), 1, d_F + 2]$

$[F, 3{,}1]$

$F == B \rightarrow 0$

$[(E(A), T(A)), 1, d_A + 1]$

$[(E(F), T(F)), 3, d_F + 1]$

$[F, 1{,}2]$

$[F, 0{,}0]$

$F == F \rightarrow 1$

$[(E(F), T(F)), 0, d_F]$

$[(E(A), T(A)), 1, d_A + 2]$

$\{F, (E(A), T(A)), d_A\}$

$\{F, (E(A), T(A)), d_A + 1\}$

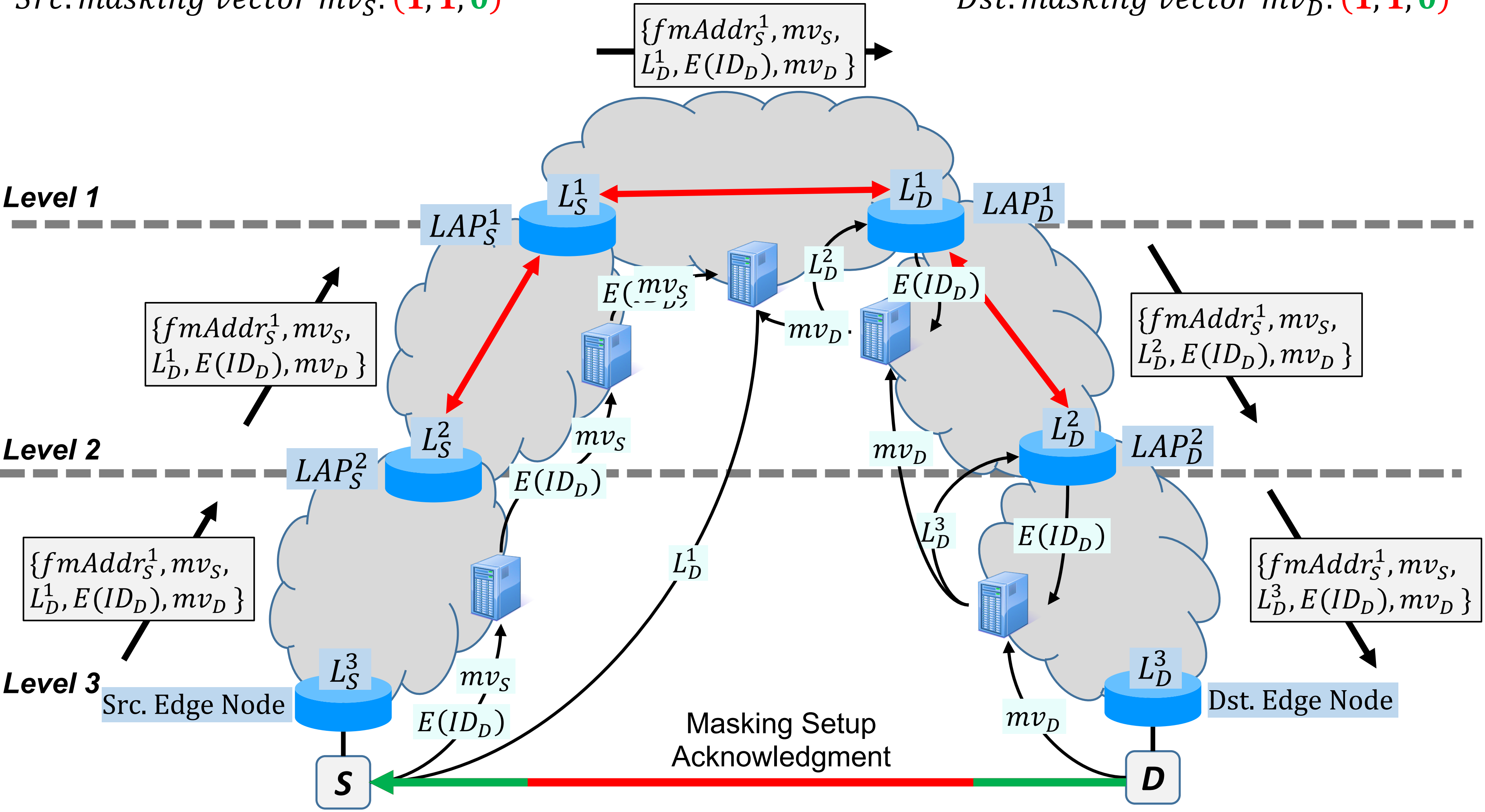
$\{E(A), (E(F), T(F)), d_F\}$

$\{E(A), (E(F), T(F)), d_F + 1\}$

Case 2

Fully BPF On-Demand

$Src.masking\ vector\ mv_S$: $(1, 1, 0)$ $Dst.masking\ vector\ mv_D$: $(1, 1, 0)$



$\{fmAddr_S^1, mv_S, L_D^1, E(ID_D), mv_D\}$

$\{fmAddr_S^1, mv_S, L_D^1, E(ID_D), mv_D\}$

$\{fmAddr_S^1, mv_S, L_D^1, E(ID_D), mv_D\}$

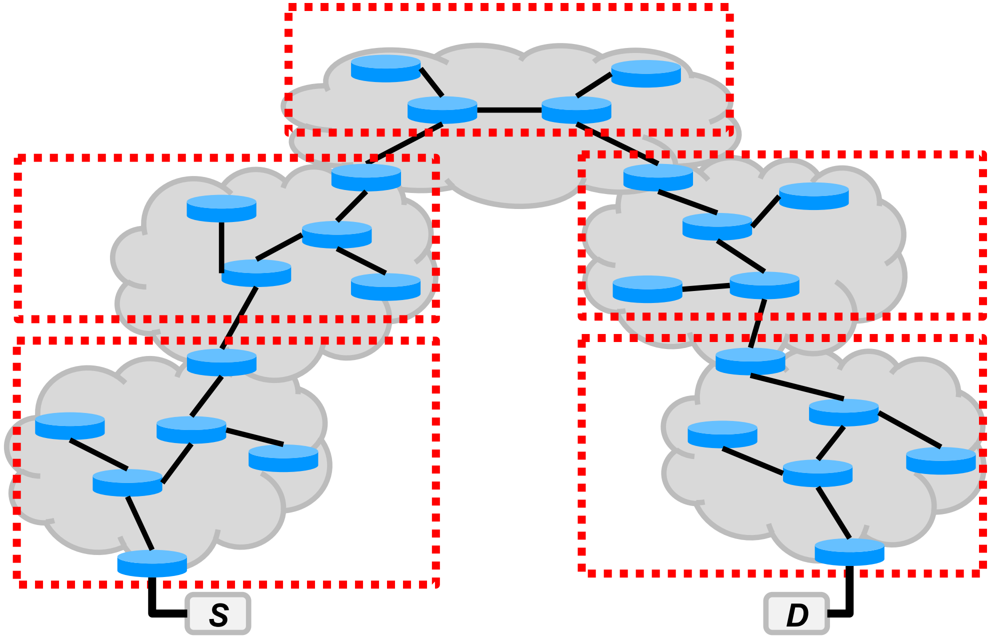
$\{fmAddr_S^1, mv_S, L_D^2, E(ID_D), mv_D\}$

$\{fmAddr_S^1, mv_S, L_D^3, E(ID_D), mv_D\}$

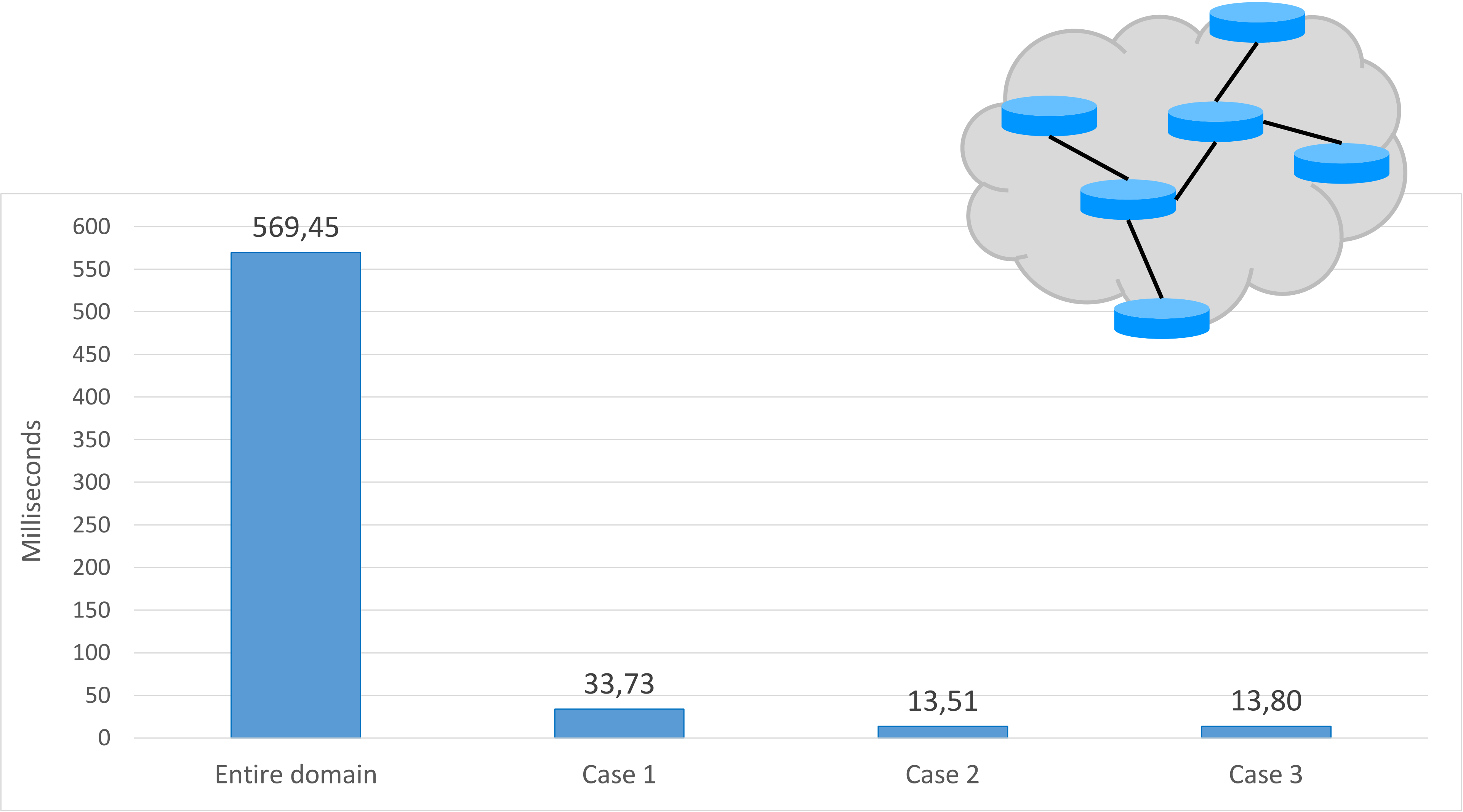
Masking Setup Acknowledgment

Implementation & Testbed

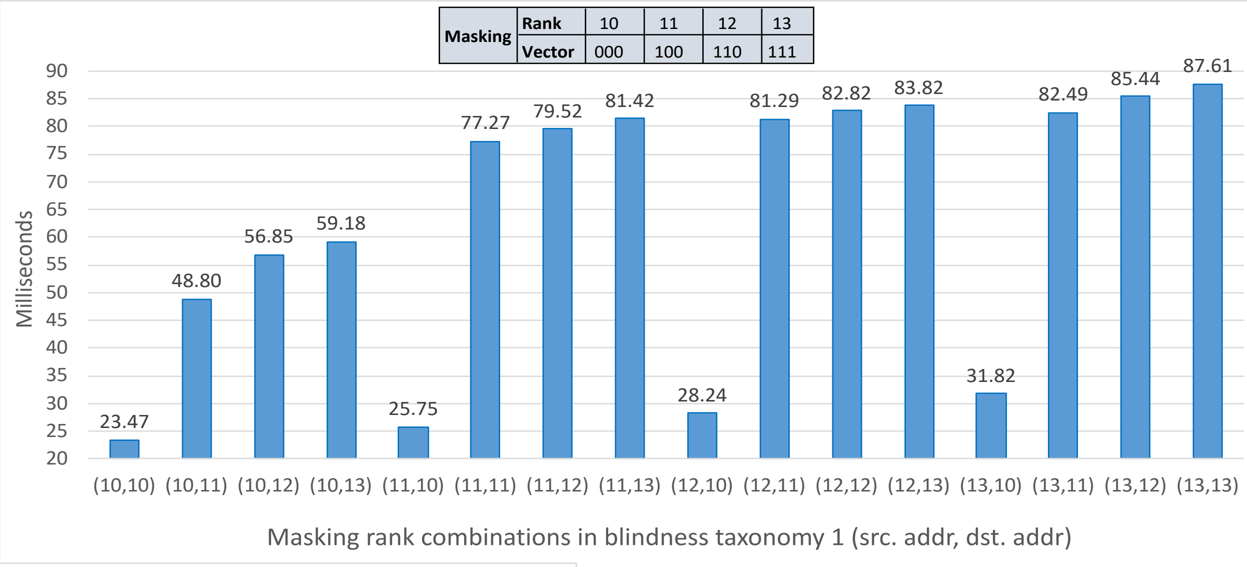
- OLV-OpenFlow
  - replaces the Type Length Value (TLV)-based mechanism in OpenFlow with an Offset Length Value (OLV)-based proceeding
- OLV-NOX: OLV-OpenFlow compatible controller
- Blind Network Stack (BNS)
- Emulated by Mininet (Intel Core i5-7200U 2.50 GHz CPU)



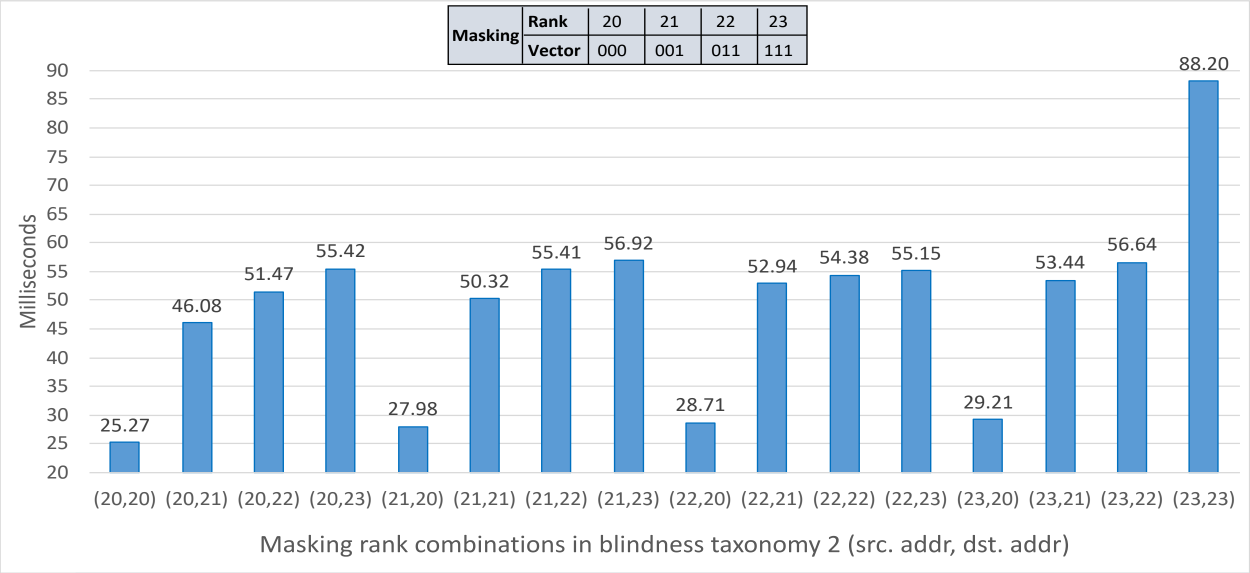
Evaluation – Masked Routing Table Entry Setup Times



Evaluation – Masking Setup Times in Blindness Taxonomies 1 & 2



Masking	Rank	10	11	12	13
	Vector	000	100	110	111



Masking	Rank	20	21	22	23
	Vector	000	001	011	111

Conclusion

- Full blindness in a domain needs to set up and maintain masked entries in the entire domain
  - Costly process
- On-Demand BPF provides selective masked routing and full blindness on demand
- Performance

Thank you for your attention!

Questions?