Academic work on path verification protocols

Design options,
associated overhead and performance properties

Antoine FRESSANCOURT

Path verification in academic work

Design dimensions of major proposals

- Design dimensions:
  - Mutual trust between the source and the destination?
  - Verification of the path:
    - By the destination node?
    - By every intermediate nodes?
  - Authentication of the source:
    - By the destination node?
    - By each intermediate node?
  - Probalistic or deterministic verification scheme?
  - Encryption of routing directive?

➔ Potential impact:

- Encryption load
- Header overhead
- Processing time in routers / network nodes

How to:

➔ Verify the path as accurately as possible?

➔ Minimize overhead?

➔ Reduce the (cryptographic) load as much as possible?

Path verification in academic work

Comparison table of major proposals

	Scheme characteristics				Header size (bytes) / intermediate nodes			Comparison / Min. header size	
	Mutual Source-dest. Trust	Validating node (Src / Dst / Interim)	Source authent. (Dst / Interim)	Deterministic / probabilist	n	4	2	4	2
ICING	No	Yes / Yes / Yes	Yes / Yes	Deterministic	79 + 42.n	247	163	10,29	6,79
Path trace	Yes / No	No / Yes / Partial	No / No	Deterministic	48	48	48	2,00	2,00
Src Auth. By Dst and Int nodes	Yes	No / No / No	Yes / Yes	Deterministic	48 + 16.n	112	80	4,67	3,33
OPT with Trust	Yes	Yes / Yes / Yes	Yes / Yes	Deterministic	68 + 16.n	132	100	5,50	4,17
Trustless OPT	No	Yes / Yes / Yes	Yes / Yes	Deterministic	84 + 16.n	148	116	6,17	4,83
OSV	Yes	Yes / Yes / Yes	Yes / Yes	Deterministic	108 + 2.n	116	112	4,83	4,67
PPV - IPv4	Yes	No / Yes / Partial	No / No	Probabilistic	64	64	64	2,67	2,67
PPV - IPv6	Yes	No / Yes / Partial	No / No	Probabilistic	88	88	88	3,67	3,67
SRv6 HMAC*	Yes	Integrity / Int. / Int.	No / No	Passive verification	40	40	40	1,67	1,67
RFL	Yes / No	Yes / Yes / Yes	Yes / Yes	Deterministic	38 + 4.n	54	46	2,25	1,92
MASK	Yes	Yes / Yes / Partial	Yes / Partial	Probabilistic	24	24	24	1,00	1,00
EPIC - L3	Yes	Yes / Yes / No	Yes / Yes	Deterministic	36 + 5.n	56	46	2,33	1,92
SR-TPP	Yes	No / Yes / Yes	Yes / Yes	Deterministic	48 + 16.n	112	80	4,67	3,33

** Linux implementation using SHA-256*

Path verification in academic work

Comparison table of major proposals

	Scheme characteristics				Header size (bytes) / intermediate nodes			Comparison / Min. header size	
	Mutual Source-dest. Trust	Validating node (Src / Dst / Interim)	Source authent. (Dst / Interim)	Deterministic / probabilist	n	4	2	4	2
ICING	No	Yes / Yes / Yes	Yes / Yes	Deterministic	79 + 42.n	247	163	10,29	6,79
Path trace	Yes / No	No / Yes / Partial	No / No	Deterministic	48	48	48	2,00	2,00
Src Auth. By Dst and Int nodes	Yes	No / No / No	Yes / Yes	Deterministic	48 + 16.n	112	80	4,67	3,33
OPT with Trust	Yes	Yes / Yes / Yes	Yes / Yes	Deterministic	68 + 16.n	132	100	5,50	4,17
Trustless OPT	No	Yes / Yes / Yes	Yes / Yes	Deterministic	84 + 16.n	148	116	6,17	4,83
OSV	Yes	Yes / Yes / Yes	Yes / Yes	Deterministic	108 + 2.n	116	112	4,83	4,67
PPV - IPv4	Yes	No / Yes / Partial	No / No	Probabilistic	64	64	64	2,67	2,67
PPV - IPv6	Yes	No / Yes / Partial	No / No	Probabilistic	88	88	88	3,67	3,67
SRv6 HMAC*	Yes	Integrity / Int. / Int.	No / No	-	40	40	40	1,67	1,67
		es / Yes / Yes	Yes / Yes	Deterministic	38 + 4.n	54	46	2,25	1,92
		s / Yes / Partial	Yes / Partial	Probabilistic	24	24	24	1,00	1,00
EPIC - L3		Yes / Yes / No	Yes / Yes	Deterministic	36 + 5.n	56	46	2,33	1,92
SR-TPP	Yes	No / Yes / Yes	Yes / Yes	Deterministic	48 + 16.n	112	80	4,67	3,33
SRv6-PoT	Yes	No / Yes / No	Yes / No	Deterministic	56	56	56	2,33	2,33

Recent internet drafts

** Linux implementation using SHA-256*

Path verification in academic work

Comparison table of major proposals

	Scheme characteristics							Comparison / Min. header size	
	Mutual Source-dest. Trust	Validating node (Src / Dst / Interim)	Source authent. (Dst / Interim)					4	2
ICING	No	Yes / Yes / Yes	Yes / Yes				163	10,29	6,79
Path trace	Yes / No	No / Yes / Partial	No / No				48	2,00	2,00
Src Auth. By Dst and Int nodes	Yes	No / No / No	Yes / Yes				80	4,67	3,33
OPT with Trust	Yes	Yes / Yes / Yes	Yes / Yes				100	5,50	4,17
Trustless OPT	No	Yes / Yes / Yes	Yes / Yes	Deterministic	84 + 16.n	148	116	6,17	4,83
OSV	Yes	Yes / Yes / Yes	Yes / Yes	Deterministic	108 + 2.n	116	112	4,83	4,67
PPV - IPv4	Yes	No / Yes / Partial	No / No	Probabilistic	64	64	64	2,67	2,67
PPV - IPv6	Yes	No / Yes / Partial	No / No	Probabilistic	88	88	88	3,67	3,67
SRv6 HMAC*	Yes	Integrity / Int. / Int.	No / No	-	40	40	40	1,67	1,67
RFL	Yes / No	Yes / Yes / Yes	Yes / Yes	Deterministic	38 + 4.n	54	46	2,25	1,92
MASK	Yes	Yes / Yes / Partial	Yes / Partial	Probabilistic	24	24	24	1,00	1,00
EPIC - L3	Yes	Yes / Yes / No	Yes / Yes	Deterministic	36 + 5.n	56	46	2,33	1,92
SR-TPP	Yes	No / Yes / Yes	Yes / Yes	Deterministic	48 + 16.n	112	80	4,67	3,33
SRv6-PoT	Yes	No / Yes / No	Yes / No	Deterministic	56	56	56	2,33	2,33

In these columns, we compare the path verification header's overhead compared with the path verification header of minimum size for a path with *n* nodes

** Linux implementation using SHA-256*

Path verification in academic work

Comparison table of major proposals

	Scheme characteristics				Header size				
	Mutual Source-dest. Trust	Validating node (Src / Dst / Interim)	Source authent. (Dst / Interim)	Deterministic / probabilist	n	4			
ICING	No	Yes / Yes / Yes	Yes / Yes	Deterministic	79 + 42.n	24			
Path trace	Yes / No	No / Yes / Partial	No / No	Deterministic	48	48	48	2,00	2,00
Src Auth. By Dst and Int nodes	Yes	No / No / No	Yes / Yes	Deterministic	48 + 16.n	112	80	4,67	3,33
OPT with Trust	Yes	Yes / Yes / Yes	Yes / Yes	Deterministic	68 + 16.n	132	100	5,50	4,17
Trustless OPT	No	Yes / Yes / Yes	Yes / Yes	Deterministic	84 + 16.n	148	116	6,17	4,83
OSV	Yes	Yes / Yes / Yes	Yes / Yes	Deterministic	108 + 2.n	116	112	4,83	4,67
PPV - IPv4	Yes	No / Yes / Partial	No / No	Probabilistic	64	64	64	2,67	2,67
PPV - IPv6	Yes	No / Yes / Partial	No / No	Probabilistic	88	88	88	3,67	3,67
SRv6 HMAC*	Yes	Integrity / Int. / Int.	No / No	-	40	40	40	1,67	1,67
RFL	Yes / No	Yes / Yes / Yes	Yes / Yes	Deterministic	38 + 4.n	54	46	2,25	1,92
MASK	Yes	Yes / Yes / Partial	Yes / Partial	Probabilistic	24	24	24	1,00	1,00
EPIC - L3	Yes	Yes / Yes / No	Yes / Yes	Deterministic	36 + 5.n	56	46	2,33	1,92
SR-TPP	Yes	No / Yes / Yes	Yes / Yes	Deterministic	48 + 16.n	112	80	4,67	3,33
SRv6-PoT	Yes	No / Yes / No	Yes / No	Deterministic	56	56	56	2,33	2,33

In general, the more security features, the less trust among nodes, the larger the overhead

** Linux implementation using SHA-256*

Path verification in academic work

Comparison table of major proposals

	Scheme characteristics				Header size				
	Mutual Source-dest. Trust	Validating node (Src / Dst / Interim)	Source authent. (Dst / Interim)	Deterministic / probabilist	n	intermedia			
ICING	No	Yes / Yes / Yes	Yes / Yes	Deterministic	79 + 42.n				
Path trace	Yes / No	No / Yes / Partial	No / No	Deterministic	48	48			
Src Auth. By Dst and Int nodes	Yes	No / No / No	Yes / Yes	Deterministic	48 + 16.n	112			
OPT with Trust	Yes	Yes / Yes / Yes	Yes / Yes	Deterministic	68 + 16.n	132			
Trustless OPT	No	Yes / Yes / Yes	Yes / Yes	Deterministic	84 + 16.n	148			
OSV	Yes	Yes / Yes / Yes	Yes / Yes	Deterministic	108 + 2.n	116			
PPV - IPv4	Yes	No / Yes / Partial	No / No	Probabilistic	64	64			
PPV - IPv6	Yes	No / Yes / Partial	No / No	Probabilistic	88	88	88	3,67	3,67
SRv6 HMAC*	Yes	Integrity / Int. / Int.	No / No	-	40	40	40	1,67	1,67
RFL	Yes / No	Yes / Yes / Yes	Yes / Yes	Deterministic	38 + 4.n	54	46	2,25	1,92
MASK	Yes	Yes / Yes / Partial	Yes / Partial	Probabilistic	24	24	24	1,00	1,00
EPIC - L3	Yes	Yes / Yes / No	Yes / Yes	Deterministic	36 + 5.n	56	46	2,33	1,92
SR-TPP	Yes	No / Yes / Yes	Yes / Yes	Deterministic	48 + 16.n	112	80	4,67	3,33
SRv6-PoT	Yes	No / Yes / No	Yes / No	Deterministic	56	56	56	2,33	2,33

Overhead depends on number of nodes on the path if intermediate nodes verify the path or authenticate the source in a deterministic way

⇔ Presence of « intermediate » fields in the header

** Linux implementation using SHA-256*

Path verification in academic work

Comparison table of major proposals

	Scheme characteristics				Header size				
	Mutual Source-dest. Trust	Validating node (Src / Dst / Interim)	Source authent. (Dst / Interim)	Deterministic / probabilist	n	4			
ICING	No	Yes / Yes / Yes	Yes / Yes	Deterministic	79 + 42.n	247			
Path trace	Yes / No	No / Yes / Partial	No / No	Deterministic	48	48			
Src Auth. By Dst and Int nodes	Yes	No / No / No	Yes / Yes	Deterministic	48 + 16.n				
OPT with Trust	Yes	Yes / Yes / Yes	Yes / Yes	Deterministic	68 + 16.n				
Trustless OPT	No	Yes / Yes / Yes	Yes / Yes	Deterministic	84 +	148			
OSV	Yes	Yes / Yes / Yes	Yes / Yes	Deterministic	108 + 2.n	116	112	4,83	4,67
PPV - IPv4	Yes	No / Yes / Partial	No / No	Probabilistic	64	64	64	2,67	2,67
PPV - IPv6	Yes	No / Yes / Partial	No / No	Probabilistic	88	88	88	3,67	3,67
SRv6 HMAC*	Yes	Integrity / Int. / Int.	No / No	-	40	40	40	1,67	1,67
RFL	Yes / No	Yes / Yes / Yes	Yes / Yes	Deterministic	38 + 4.n	54	46	2,25	1,92
MASK	Yes	Yes / Yes / Partial	Yes / Partial	Probabilistic	24	24	24	1,00	1,00
EPIC - L3	Yes	Yes / Yes / No	Yes / Yes	Deterministic	36 + 5.n	56	46	2,33	1,92
SR-TPP	Yes	No / Yes / Yes	Yes / Yes	Deterministic	48 + 16.n	112	80	4,67	3,33
SRv6-PoT	Yes	No / Yes / No	Yes / No	Deterministic	56	56	56	2,33	2,33

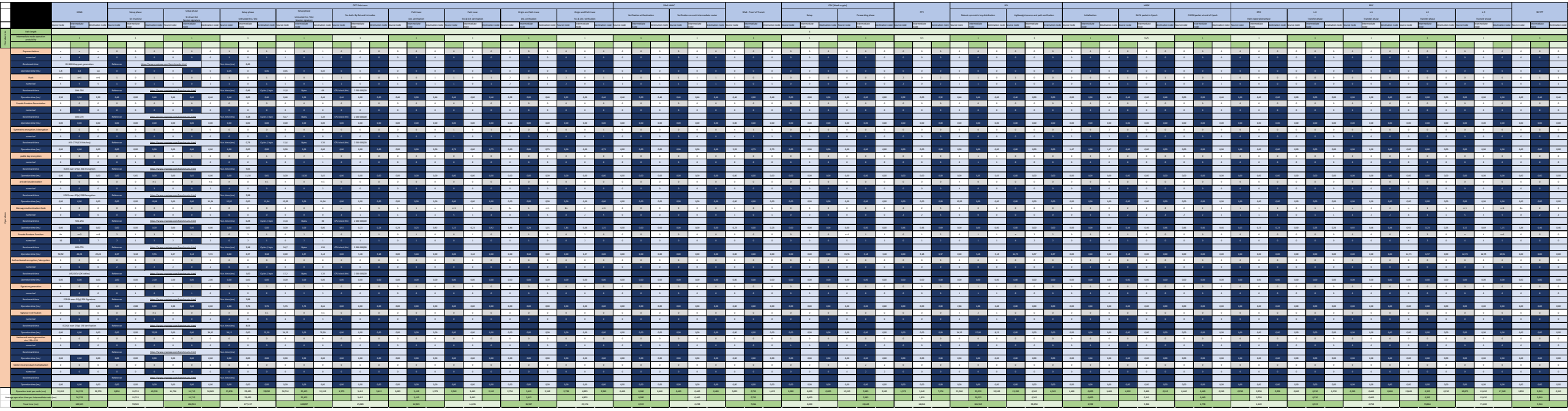
Probabilistic schemes have been introduced to allow a verification by intermediate nodes while limiting the linear overhead ⇔ verification done over threads of packets / network flows

** Linux implementation using SHA-256*

Cryptographic overhead comparison

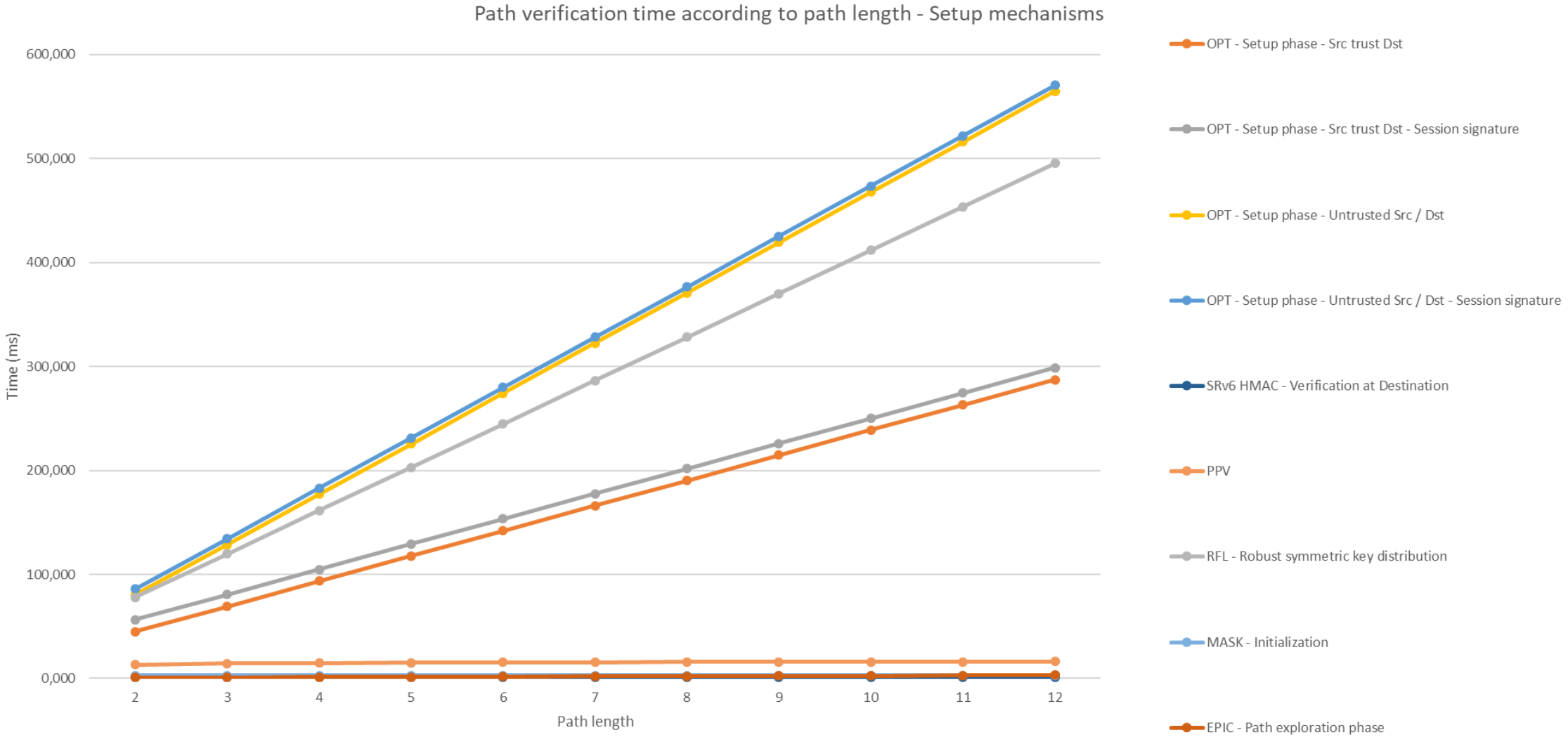
Based on a counting cryptographic function calls

Methodology

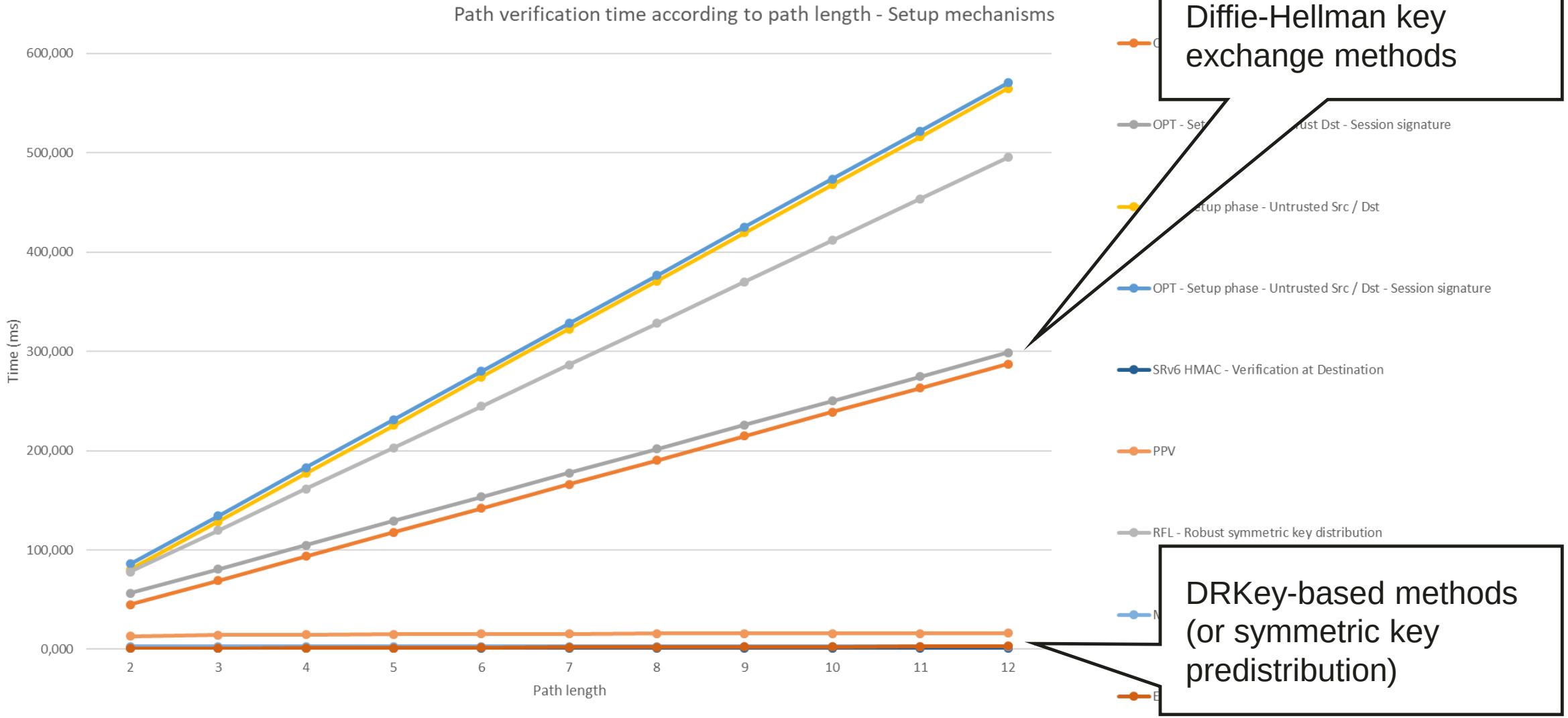


- Careful counting of cryptographic operations in path verification algorithms descriptions in published academic papers
- Cost of unitary cryptographic operation derived from Crypto++ library benchmark
- Considering end to end verification time and average per intermediate node verification time (taking probabilistic operations into account)

Setup mechanisms comparison



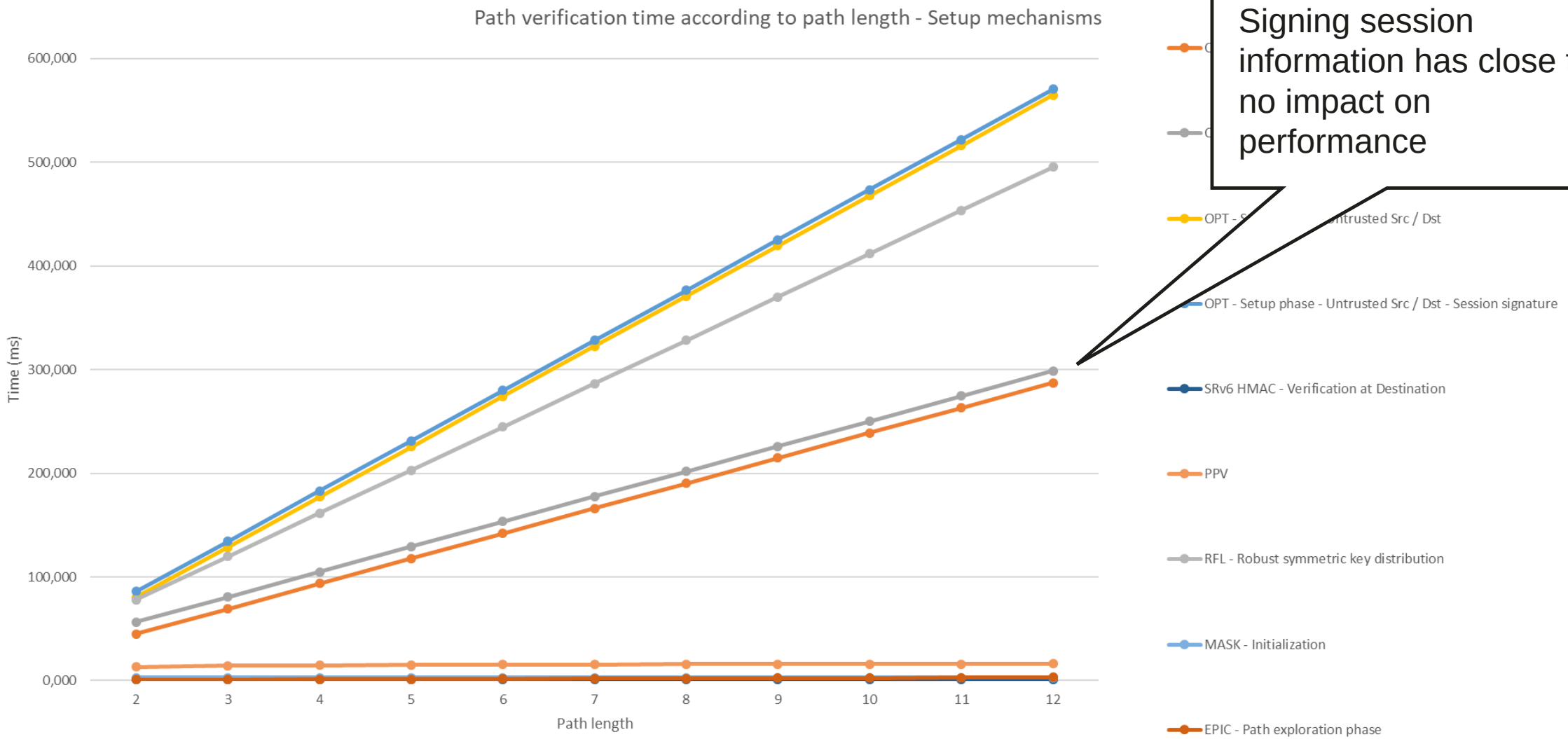
Setup mechanisms comparison



Diffie-Hellman key exchange methods

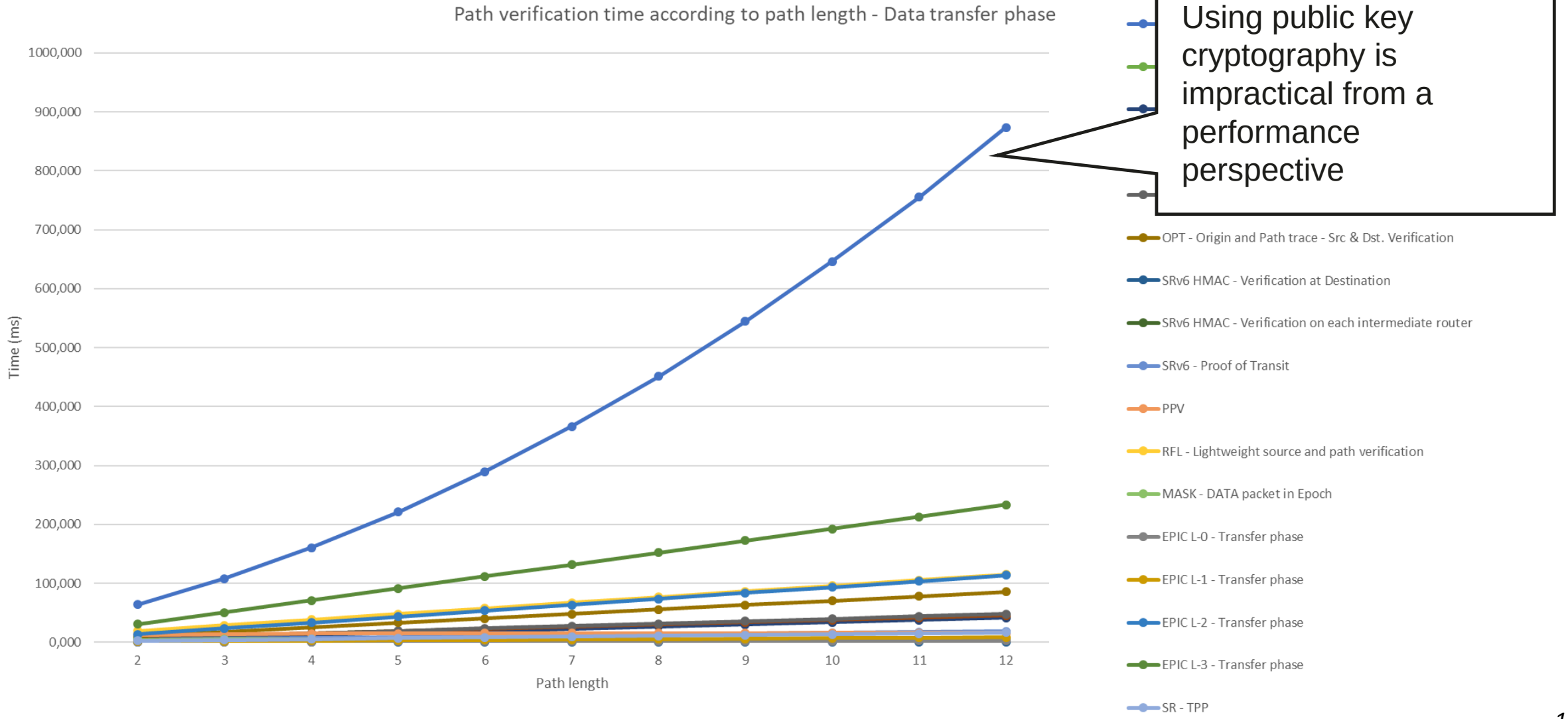
DRKey-based methods (or symmetric key predistribution)

Setup mechanisms comparison



Signing session information has close to no impact on performance

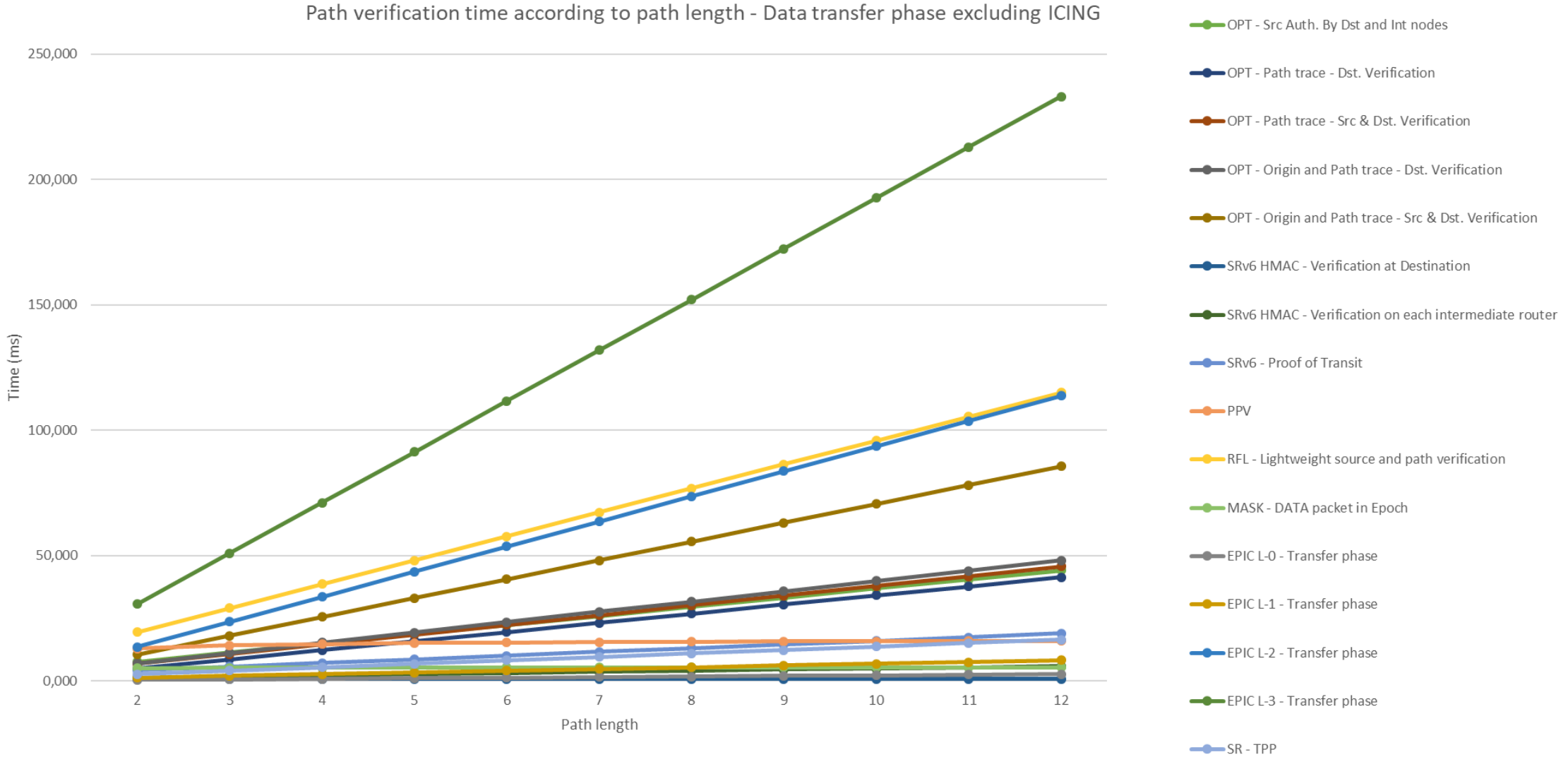
Data transfer phase comparison



Using public key cryptography is impractical from a performance perspective

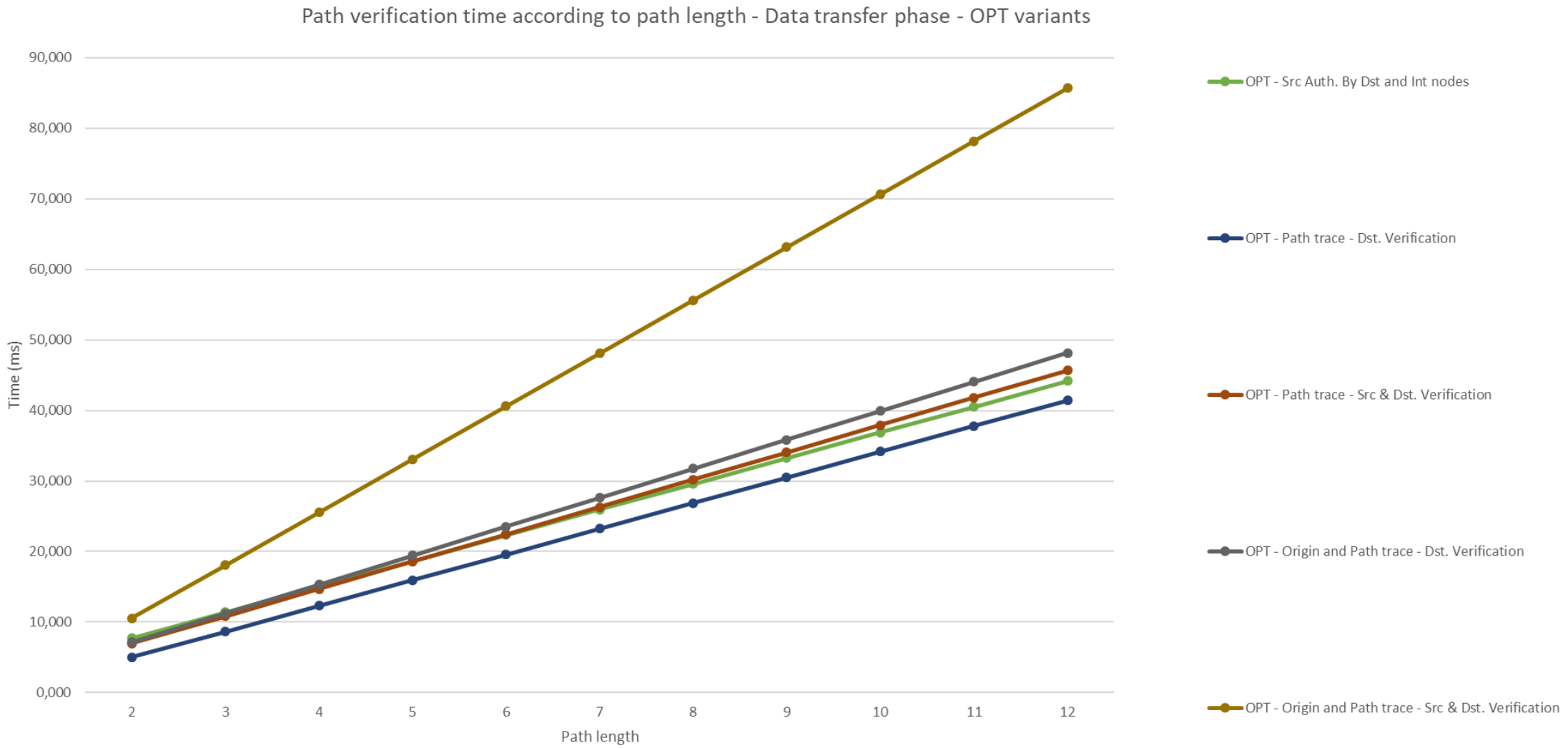
Data transfer phase comparison

Excluding ICING



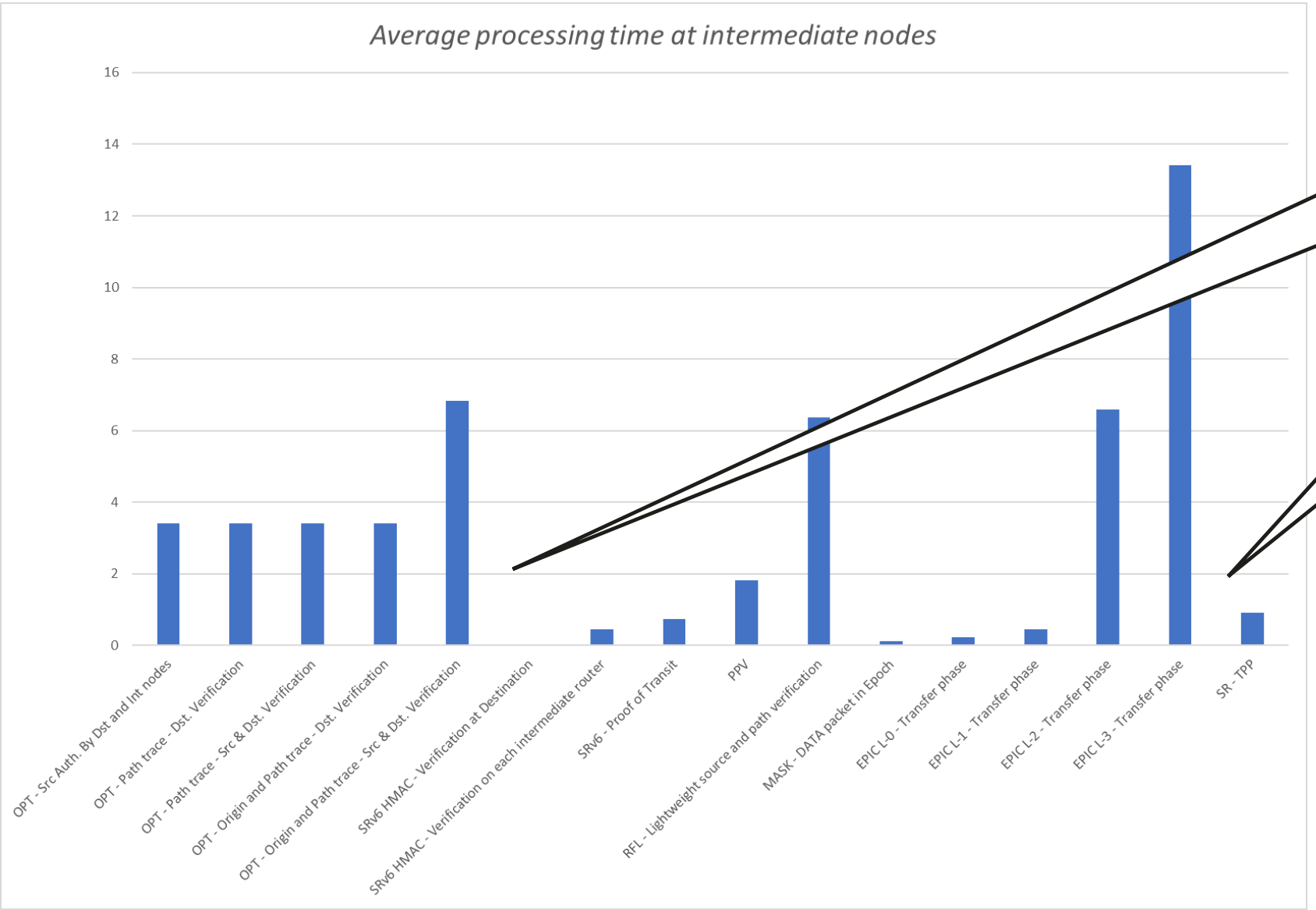
Data transfer phase comparison

Impact of protection level – OPT variants



Data transfer phase comparison

Average processing time at intermediate nodes



In OPT, performance impact of increased protection is limited wrt. EPIC

Thank you!

Contact: Antoine Fressancourt

antoine.fressancourt@huawei.com

antoine@aft.network

Path verification in academic work

References

ICING	Naous, Jad, et al. "*Verifying and enforcing network paths with ICING*." Proceedings of the Seventh Conference on Emerging Networking Experiments and Technologies. 2011.
Path Trace & OPT flavors	Kim, Tiffany Hyun-Jin, et al. "*Lightweight source authentication and path validation.*" Proceedings of the 2014 ACM Conference on SIGCOMM. 2014.
OSV	Cai, Hao, and Tilman Wolf. "*Source authentication and path validation in networks using orthogonal sequences.*" 2016 25th International Conference on Computer Communication and Networks (ICCCN). IEEE, 2016.
PPV	Wu, Bo, et al. "*Enabling efficient source and path verification via probabilistic packet marking.*" 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS). IEEE, 2018.
RFL	Wu, Bo, et al. "*RFL: Robust fault localization on unreliable communication channels.*" Computer Networks 158 (2019): 158-174.
MASK	Fu, Songtao, et al. "*MASK: practical source and path verification based on Multi-AS-Key.*" IEEE/ACM Transactions on Networking (2022).
EPIC	Legner, Markus, et al. "*{EPIC}: every packet is checked in the data plane of a {Path-Aware} Internet.*" 29th USENIX Security Symposium (USENIX Security 20). 2020.
SR-TPP	Zhou, Jiang, et al. "*SR-TPP: Extending IPv6 segment routing to enable trusted and private network paths.*" 2020 IEEE Symposium on Computers and Communications (ISCC). IEEE, 2020.
SRv6-HMAC	Filsfils, Clarence, et al. "*IPv6 segment routing header (SRH).*" IETF RFC 8754, available: https://datatracker.ietf.org/doc/rfc8754/ (2020).
SRv6-PoT	Iannone, Luigi et al. "*Segment Routing over IPv6 (SRv6) Proof of Transit.*" IETF Internet draft, available: https://datatracker.ietf.org/doc/draft-iannone-spring-srv6-pot/ (2024).

Verifying and enforcing network paths with ICING

ACM CoNEXT '11

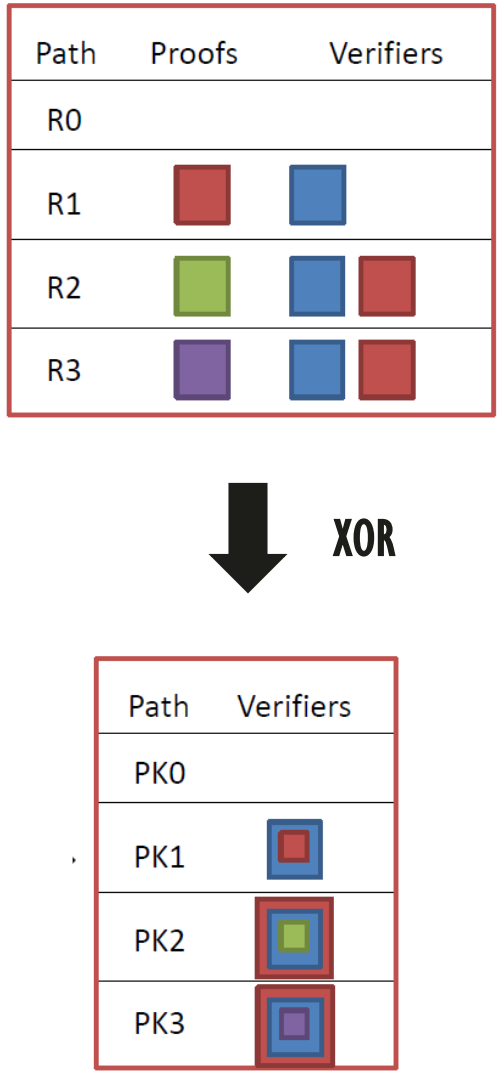
- Use of 2 key enablers:

1. Aggregate message authentication codes
   - To verify a path, a given node should verify that each node before it was traversed ➔ $O(n^2)$ verifier fields in the header in a classic setup
   - In Aggregate MACs, verification fields are XORed to reduce the number of fields to carry ➔ O(n) verifier fields
2. Self-certifying names
   - Nodes locally compute a private/public key pair $<k_i, N_i>$, and use Ni as their name
   - Symmetric keys are generated using a function f such that, for a pair of nodes i and j :

$$f(k_i, N_j) = f(N_j, k_i)$$

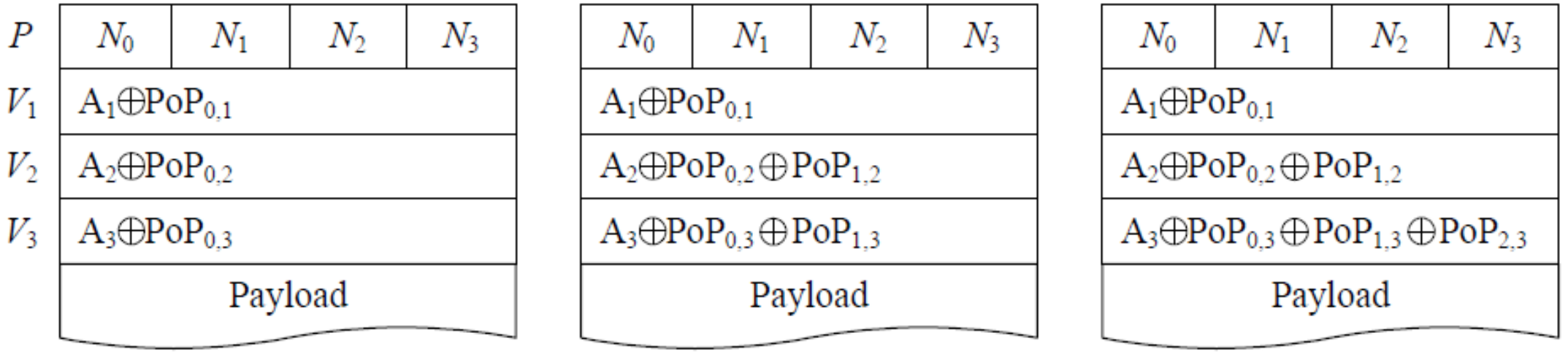
➔ *Non-interactive key exchange primitive*

- Use of both primitives for path verification



Verifying and enforcing network paths with ICING

Path verification in details



(a) updated by N_0 to be verified by N_1

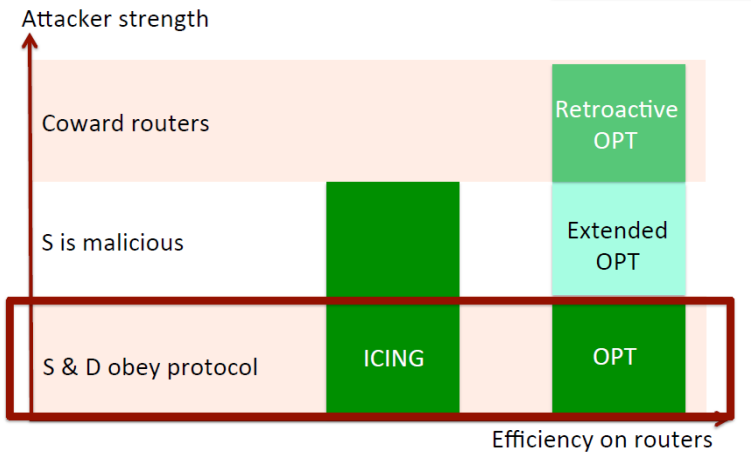
(b) updated by N_1 to be verified by N_2

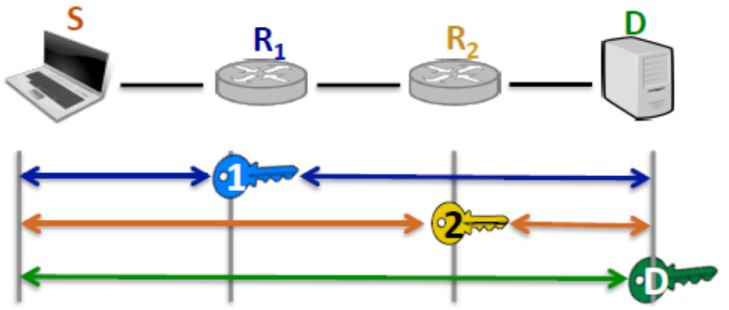
(c) updated by N_2 to be verified by N_3

Origin and Path Trace (OPT)

Lightweight source authentication and path validation – ACM SIGComm '14

- Design of 3 path verification variants to enhance ICING
  - Retroactive OPT ➔ Path verification after the packet has been received, used to analyze awkward behaviors
  - OPT ➔ Used to verify a path when the source and destination are trusted
  - Extended-OPT ➔ Used to verify the path in case the source is not trusted
- OPT principle:
  - S selects a path to D
  - Nodes establish shared secret key(s) with S & D
  - S prepares special fields for each node in the packet header ➔ Helps each router derive shared key & authenticate source
  - Each node updates a verification field in the packet header ➔ Helps downstream nodes validate path
- Advantage compared to ICING:
  - More efficient on routers: 0(1) verifier field instead of 0(n)
  - Coward routers attack prevention





Orthogonal Sequence Verification (OSV)

Source Authentication and Path Validation in Net. Using Orthogonal Sequences – ICCCN '16

Path verification mechanism with 3 characteristics:

1. Low Verification Time:
   - Use of an inner product computation based on Hadamard matrixes
     - M x m matrixes such that $HH^T = mI_m$

   ➔ Replacement of complicated cryptographic operations

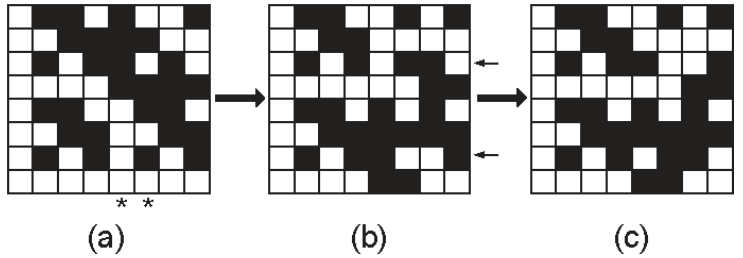
   ➔ Decreased verification time compared to ICING (~0.01µs Vs. 24.4µs)

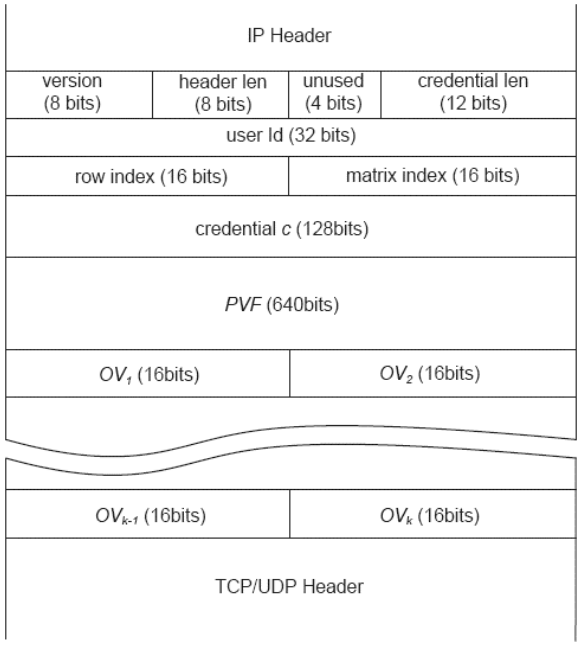
   ➔ Faster setup compared to OTP (0.15 ms Vs. 4 ms)

2. Low Packet Overhead:
   - Relatively small OSV header (~125 bytes)

   ➔ Smaller header if path length > 4

3. Non-dependence on Path Length:
   - Packet processing time in routers is almost independent of the number of hops that are traversed by the packet.
   - OSV's packet header overhead increased by 2 bytes for each additional hop.

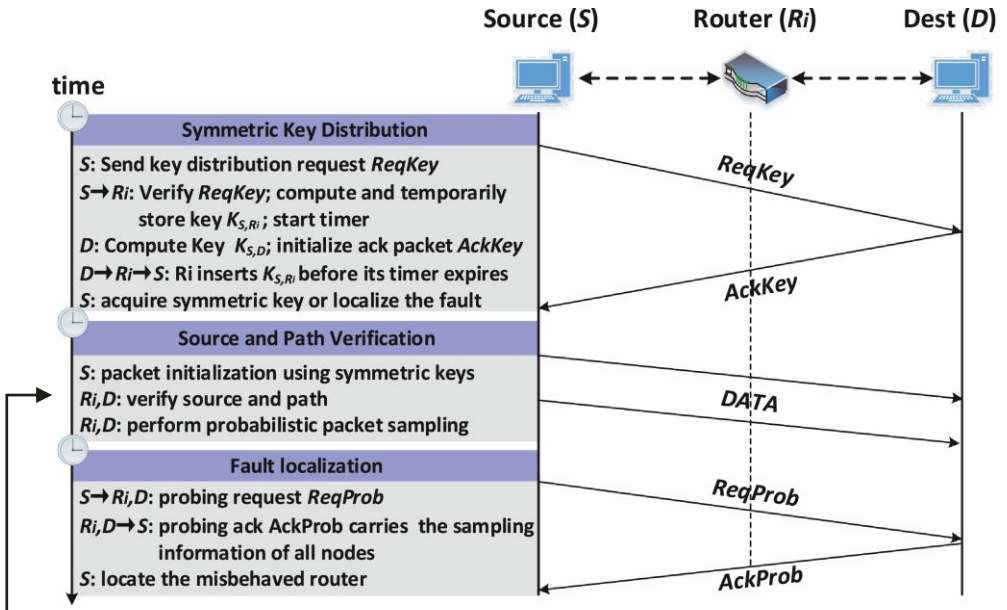




Robust Fault Localization (RFL)

RFL: Robust fault localization on unreliable communication channels

- Each packet's source is authenticated and the path is verified for each packet.
  - Setup phase using public key cryptography to allow S to retrieve symmetric keys used during an Epoch.
  - During data phase, symmetric key used by intermediate routers to verify a packet.
  - Verification uses ID of predecessor and marks computed for successor nodes (See Eq. 12)
  - Small overhead
- Additional reporting mechanism to locate router / nodes causing an issue



$$RFLheader = \{SessionID, epoch, PacketID, M_{Path}\}. \quad (9)$$

$$PacketID = H(SessionID||epoch||IP_{cst}), \quad (10)$$

$$M_{Path} = \langle M_1, M_2, \ldots, M_n, M_D \rangle, \quad (11)$$

$$M_D = PRF_{K_{S,D}}(M_{cst}^{in}||TTL_D \,||\, R_n). \quad (12)$$

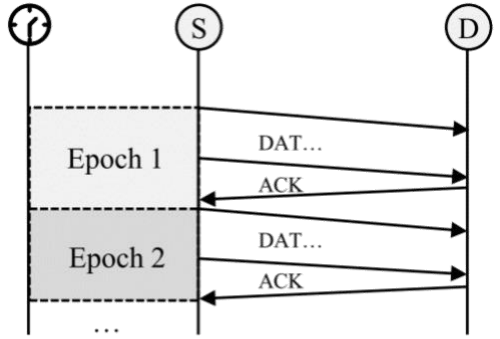
$$M_i = PRF_{K_{S,R_i}}\left(M_{cst}^{in}||TTL_i \,||\, R_{i-1} \,||\, M_{i+1} \,||\, \ldots \,||\, M_n \,||\, M_D\right).$$

MASK

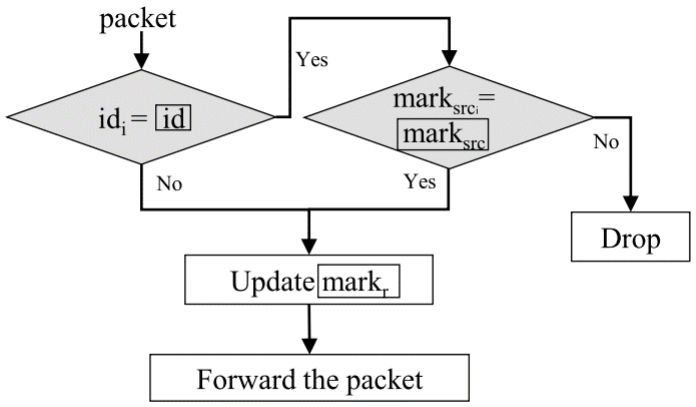
Practical Source and Path Verification Based on Multi-AS-Key

- Path verification based on a probabilistic packet marking mechanism
  - Similar as PPV by same authors

  ➔ *Fixed header size whatever the path length*
- Source is always authenticated by destination
- Key distribution and dataplane are separated to ease key management operations
  - Targeting reconstruction efficiency rather than full security
  - Key change frequently and are distributed by a Key Distribution Server



(a) Processing at end host



(b) Processing at router