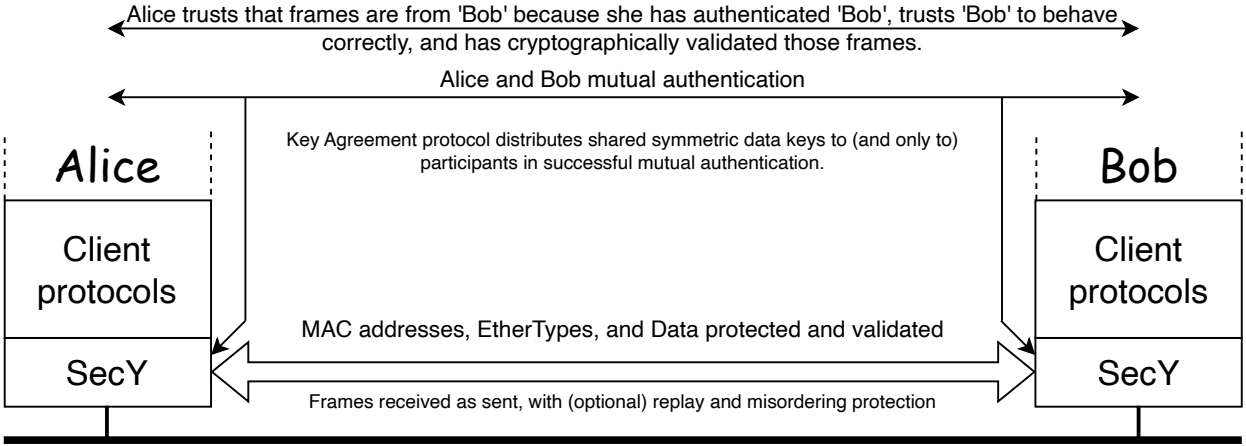
Trust, protection and verification with MACsec and MDsec

What needs to be protected, what can be protected, and what are the limits to an apparently general approach.

An absence of specific restrictions does not confer generality. Retrofitting security is not necessarily easy: specific uses cases need to be checked before anticipating success.



MACsec symmetric key (SAK) protection of frame addresses and data provides integrity and origin authenticity : Alice knows all valid frames were sent by Bob (and vice-versa) and not added or modified by an attacker.

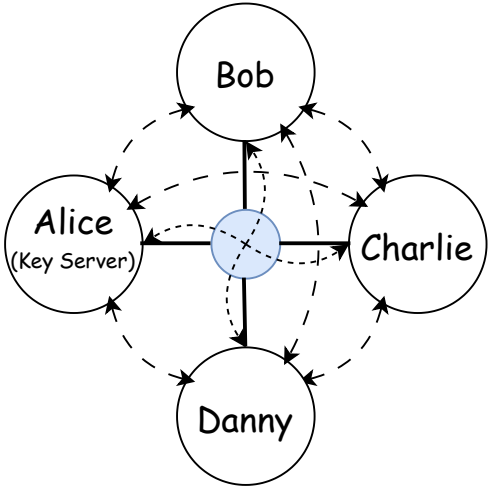
Trust (confidence) in received protocol data does not depend on protocol by protocol security measures (greater attack surface) which may even knowm to be missing or incomplete for new protocols.

Interface stacks as for simple point-to-point (prior slide), more SCs created

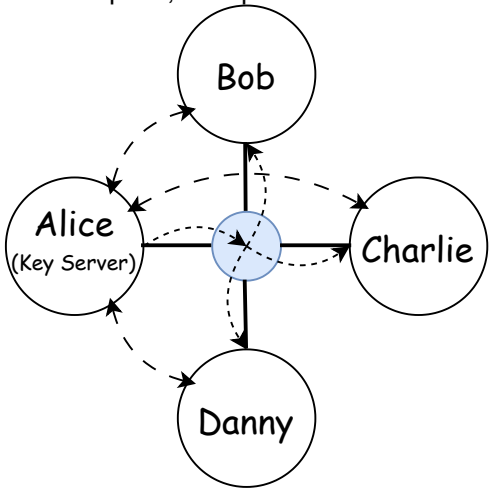
Alice (MKA Key Server) trusts cryptographically validated frames with Bob's source address are from 'Bob' because she has authenticated 'Bob', 'Charlie', and 'Danny', and trusts all of them to behave correctly.

Bob (not Key Server) trusts cryptographically validated frames with Alice's source address are from 'Alice' because he has authenticated 'Alice', and trusts her to only distribute to others that she trusts. Bob trusts validated 'Charlie', and 'Danny', and trusts all of them to behave correctly.
Bob (not Key Server)

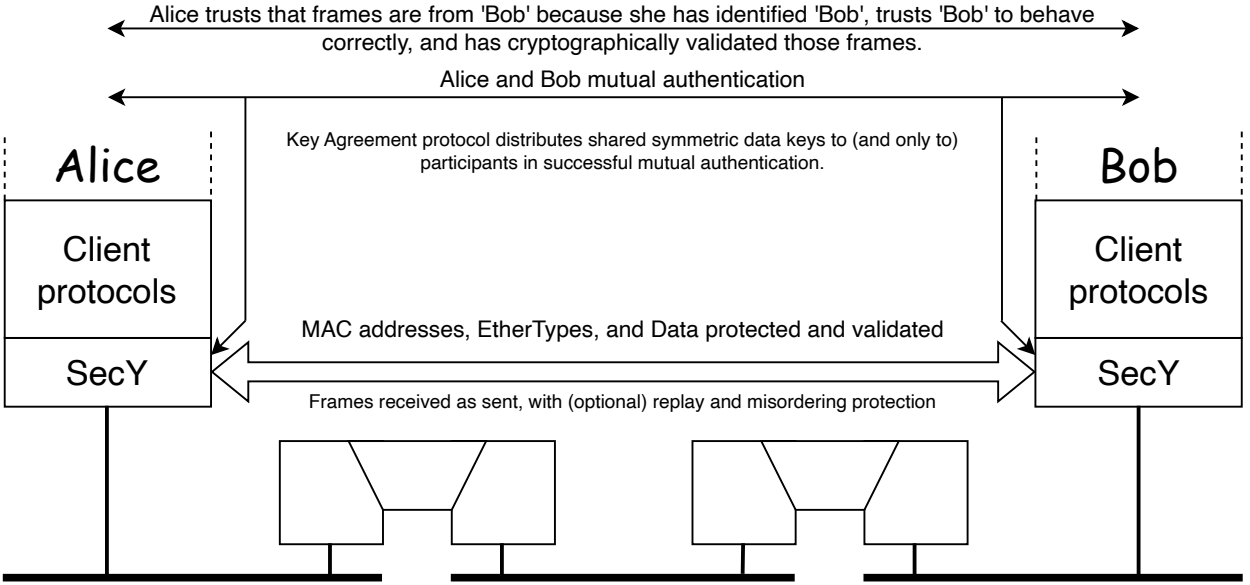
Multipoint (Group CA) any to any, multicast to all



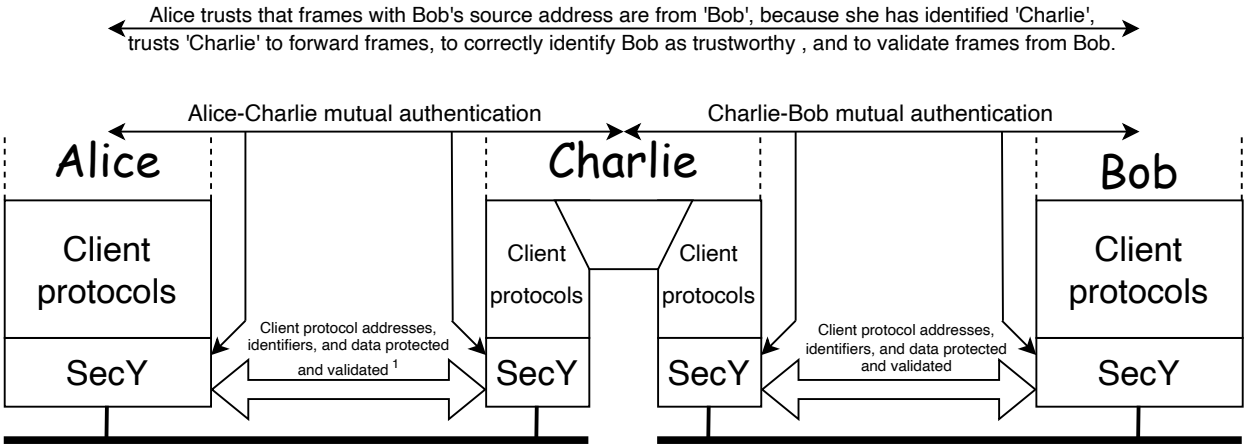
Point-to-multipoint, Group CA with subset of SCs[1]



1 With MKA Bo, C.., D.., just install SAs from Alice's transmit SC and report when those SAKs installed.



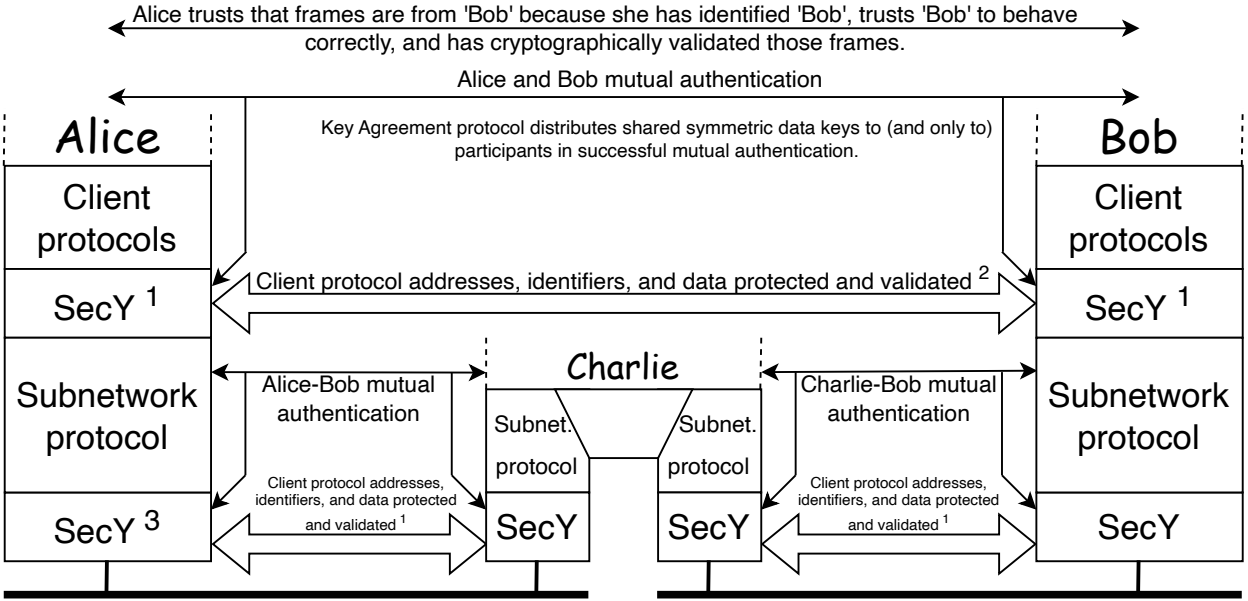
'TPMR-like' intermediate systems, transparent bridging and not participating in any Alice - Bob protocol. Can have Charlie, Danny as for multi-point and point-to-multipoint over real shared media.



[1] Hop by hop protection works (given the necessary transitive trust) even if frame forwarding (e.g., by Charlie) modifies the frame (e.g. by decrementing hop count).

Alice may be talking (through Charlie and possibly other intermediaries) to a number of 'Bob's, and may require Charlie to authenticate the binding of each 'Bob address' to that 'Bob' [2] and to check that only frames from that 'Bob' carry that 'Bob address'.

[2] By, e.g., where Charlie is a simple transparent bridge, requiring Bob to authenticate using an X.509 cert with RFC 10031 support.



[1] Hop by hop protection works (given the necessary transitive trust) even if frame forwarding (e.g., by Charlie) modifies the frame (e.g. by decrementing hop count).

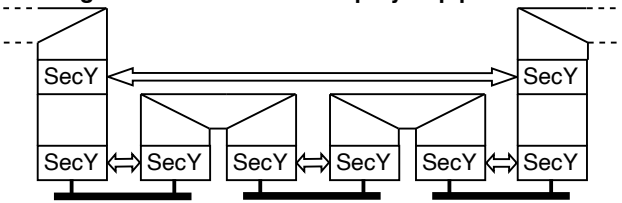
[2] End to end protection cannot protected any data modified by intervening relays (e.g., if Charlie is an IP router modifying hop count). Subnetwork protocol has to be a true shim (see, e.g., 11.7 and Clause 15 of IEEE Std 802.1AE) or protection has to be specified protocol by protocol (not 'AnySec').

[3] Alice can have many distant peers, each notionally supported by a separate 'virtual port' created on demand (see, e.g., 11.8 of IEEE Std 802.1AE).

Scenario: Organization's network paths transit one or more separately administered (or less trusted) peer subnetworks.

Segment end to segment end protection: path faults and attacks to be identified and contained to responsible segment

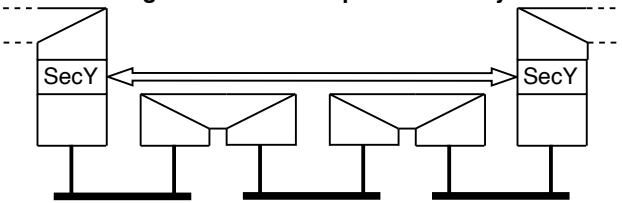
Segment end-to-end and hop-by-hop protection



Segment provider better defended, reduces external attacker access to customer organizations end equipment:

- key agreement has to run before e2e security, so potentially exposed to others.
- all frames to be validated by e2e SecY have been transmitted by an authenticated customer, and ingress binding to customer specific egress may be less error prone.

Segment end-to-end protection only



Less work for end equipment, especially if not double protection capable.

Internal operation of segment network more exposed:

- UNI could be separately protected, as for other e2e tx.
- key agreement frames could be filtered out (pattern matched) unless directed to UNI SecY (and then protected hop-by-hop to destination - a mix of these segment protection alternatives).