Ethernet–dot15 Bridging

Bridging between any two peers is only possible if they implement identical protocol stacks.

Higher layer protocols (e.g. ARP, IS-IS, IPv6) can include MAC Address data that is difficult or impossible for a bridge to modify.

Each stack can use a lower, peer-dependent, sublayer/protocol shim if each bridge interface implements the appropriate shim — encoding upper layer protocol parameters[1] in its protocol fields.[2]

A dot15 shim can encode a 48-bit MAC Address value in a dot15 64-bit address field ... making Ethernet–dot15 bridging practical.

This presentation provides detail and use cases.

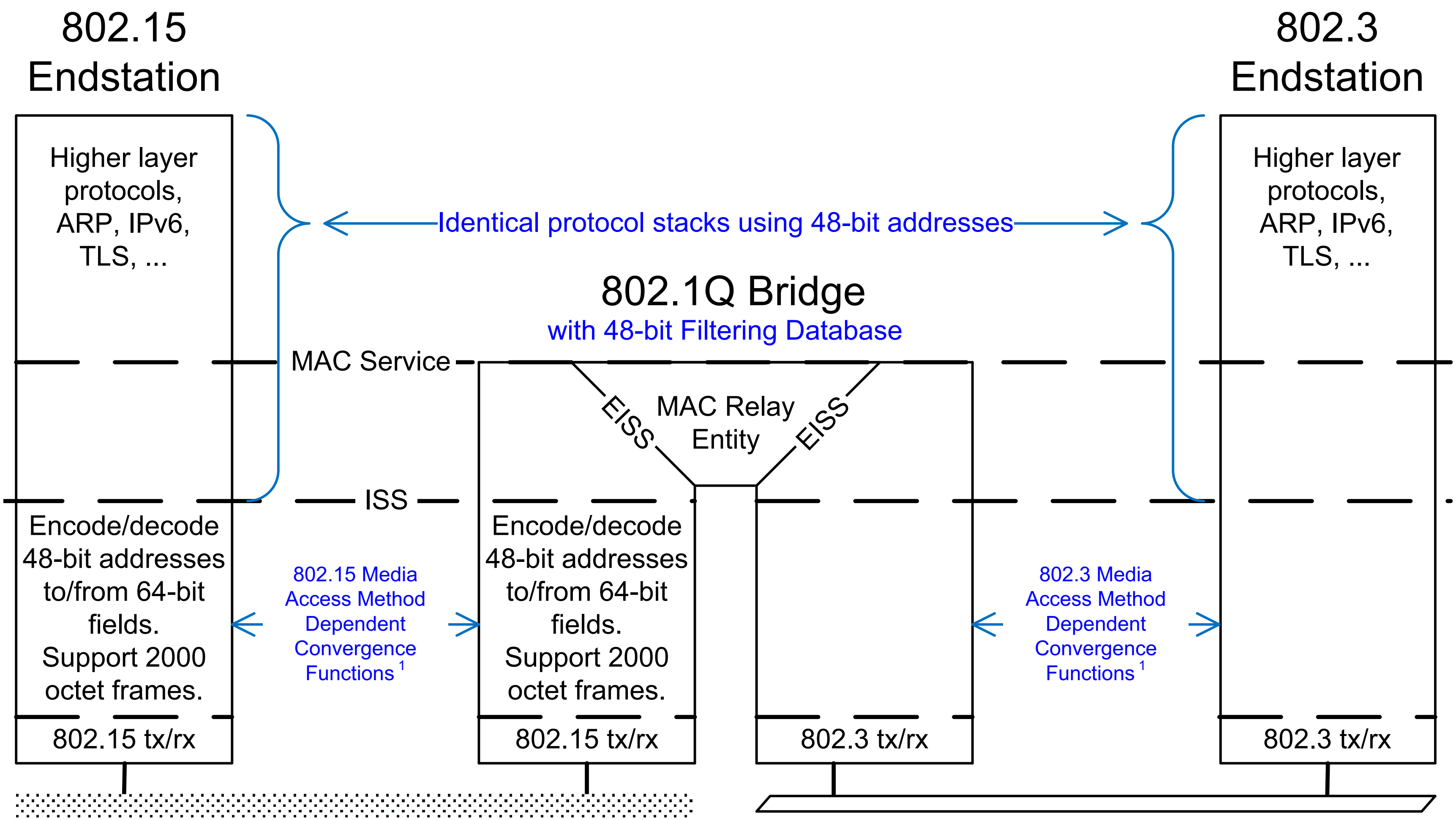
Mick Seaman
mickseaman@gmail.com

1. The parameters of the MAC Service provided by the shim.
2. Data and parameters passed by the shim to an underlying service, possibly but not necessarily the PHY.

Agenda

- Architecture
- 48-bit addresses for 802.15 stations
- 64-bit encoding of 48-bit addresses
- 802.15 64–48-bit address coexistence
- MAC Security (options)
- Use cases (example)

Architecture



[1] See 6.7 of 802.1Q

48-bit addresses for 802.15 stations

802.15 station assigns its own ‘locally administered’ 48-bit address by random selection from 46-bit or 45-bit pool.

- Preferred method, for bootstrapping at least, as does not require a management station directly attached to 802.15 media or limitation of any uniqueness guaranteeing protocol to local ‘air media’.
- Collision probability for perfect RNG (random number generator) is for a network of n stations selecting from an m bit pool is $\sim n^2/2m$.[1]
- Can be supplemented by ‘after the fact’ management protocol checking and recording.

Does not negate the utility of 64-bit unique address assignment, as that serves as a permanent ‘identity’ for management.

1. A fair approximation for n/m ~1/2, and very good for any feasible wireless LAN and RNG. See https://en.wikipedia.org/wiki/Birthday_problem.

64-bit encoding of 48-bit addresses [1]

Alternatives:

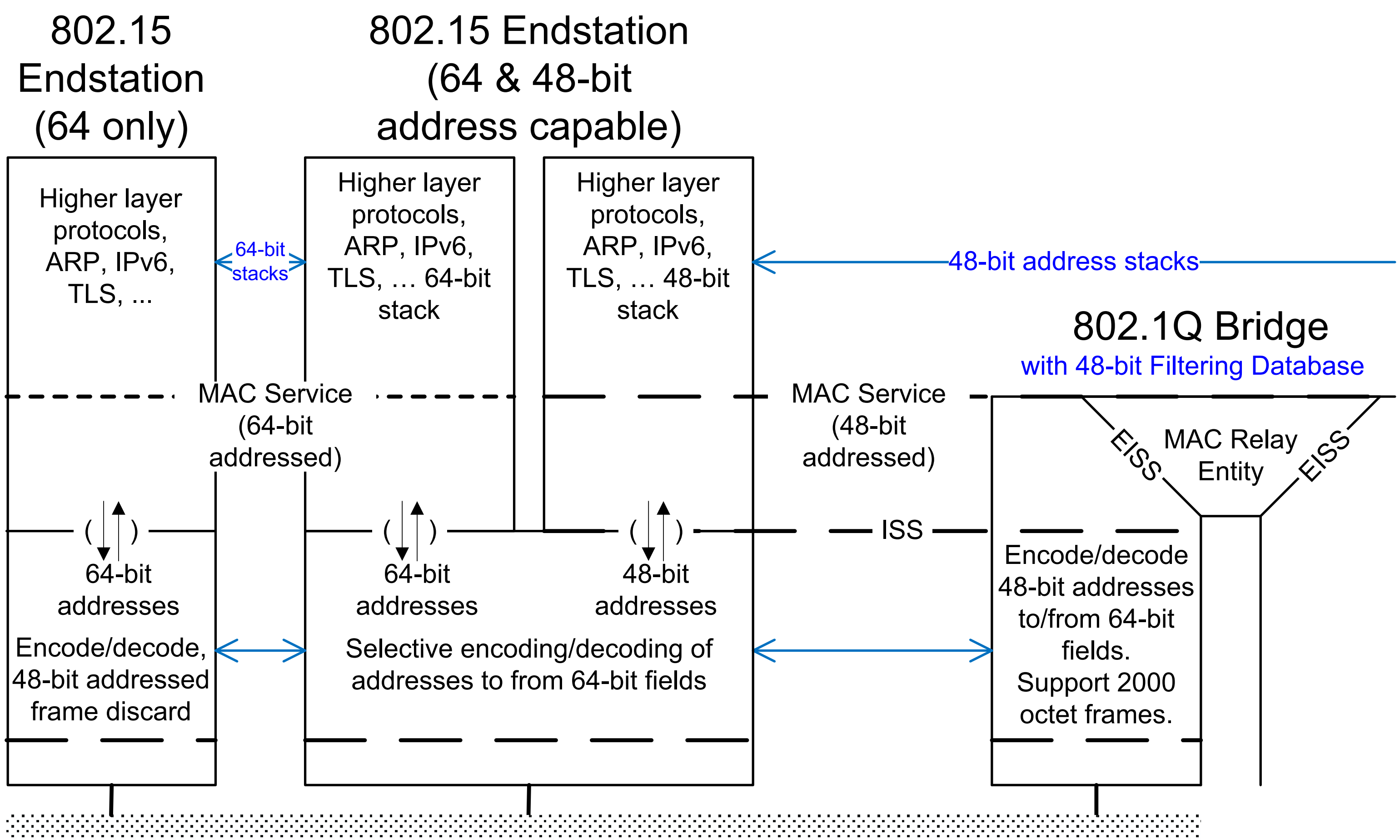
1. Use the deprecated EUI-48 to EUI-64 mapping algorithm with inserted 0xFF octets (see ‘Mapping an EUI-48 to EUI-64’ in the ‘Guidelines for use of EUI, OUI, and CID’ [2]) applying the algorithm to both unique and locally administered addresses and to both individual and group addresses.
   - Con: The algorithm is deprecated because it intrudes on the EUI-64 assigned space, and stations that have been assigned a compromised EUI-64 bit identifer will function incorrectly.
   - Pro: The algorithm is already ‘out in the wild’. Hopefully OUI assignees are aware of the potential conflict and do not assign addresses from the compromized space.
2. Encode the 48-bit addresses, whether unique or locally assigned, as 64-bit locally assigned addresses by prepending 16-bits, with the I/G (Multicast) bit set to the same value as that of the 48-bit address, choosing an appropriate SLAP quadrant for the following Y and Z bits (I suggest ‘1’ for each indicating SAI), with the remaining prepended bits also taking the value 1.
   - Con: A new option where one already exists. May conflict with existing 64-bit random assignment—although such an existing assigment is unlikely to be Ethernet bridging capable.
   - Pro: Avoids trampling on the EUI-64 (explicitly assigned) space. My preference.

---

1. Note: This is emphatically **not** a translation table, and the only 48-bit address values to be algorithmically encoded for 64-bit 802.15 transmission are those logically at the front of the frame (i.e. the MAC destination address and source address parameters of the ISS).
2. https://standards.ieee.org/wp-content/uploads/import/documents/tutorials/eui.pdf

802.15 64–48-bit address coexistence

A station can use both 48-bit and 64-bit MAC Address stacks:



MAC Security

Data Orgin Authenticity, Integrity, and (optionally) Confidentiality protection. Options for station to station communication:

1. 802.15 specific from Bridge Port (802.15 interface) to 802.15 station. MACsec from Bridge Port (802.3 interface) to 802.3 station.
2. MACsec from each Bridge Port to stations on its LAN.

   Ensures large frames are fragmented & reassembled correctly, and received frames were indeed 48-bit addressed.
3. MACsec from station to station.

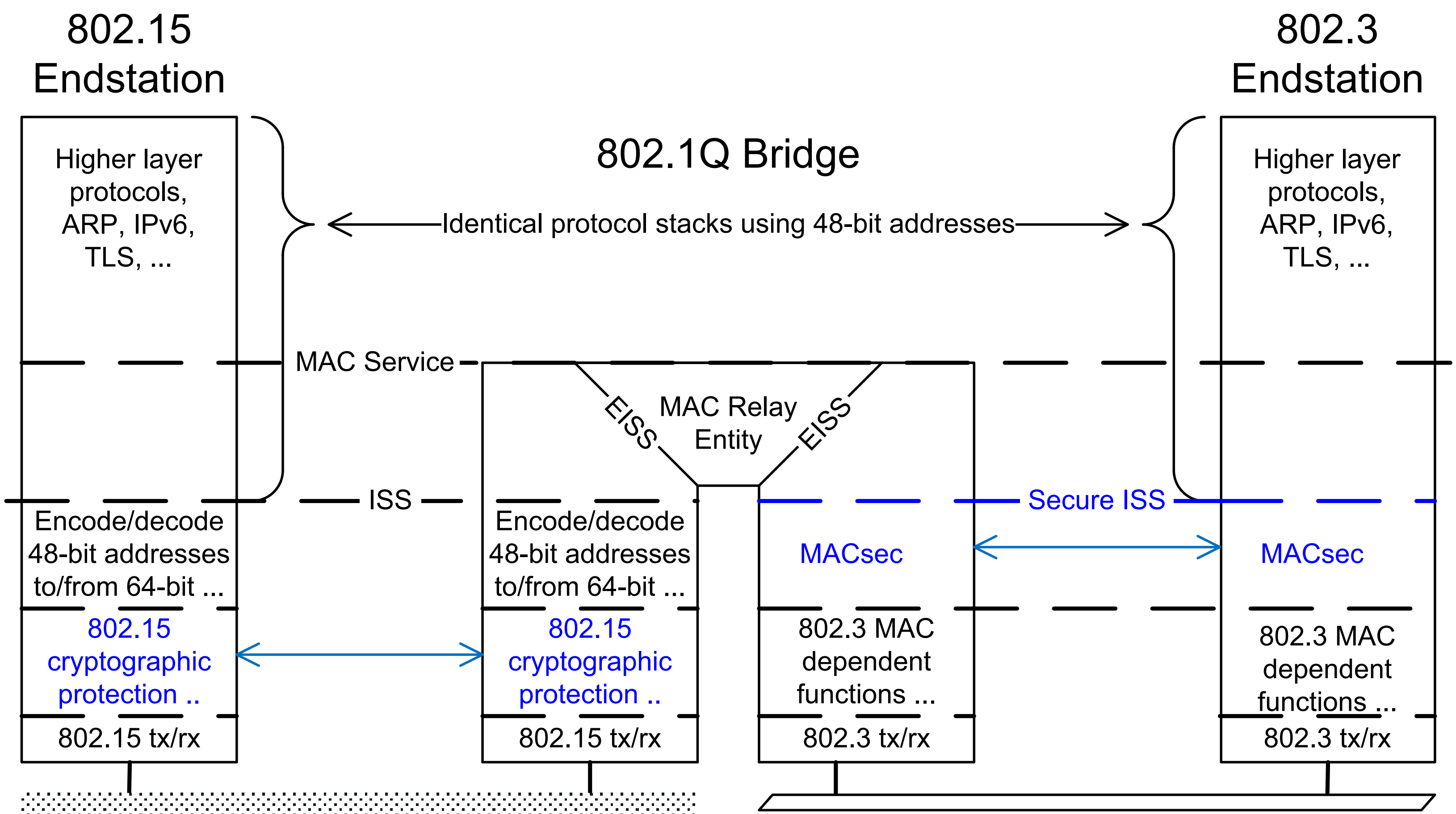
   Protects fragmentation/reassembly and addressing as per 2. Defends against corruption in intervening bridge(s), as it does not need to decrypt/re-encrypt relayed frames.

   Allows cut-through transmission by intervening bridges.[1]

Following slides provide interface stack details.

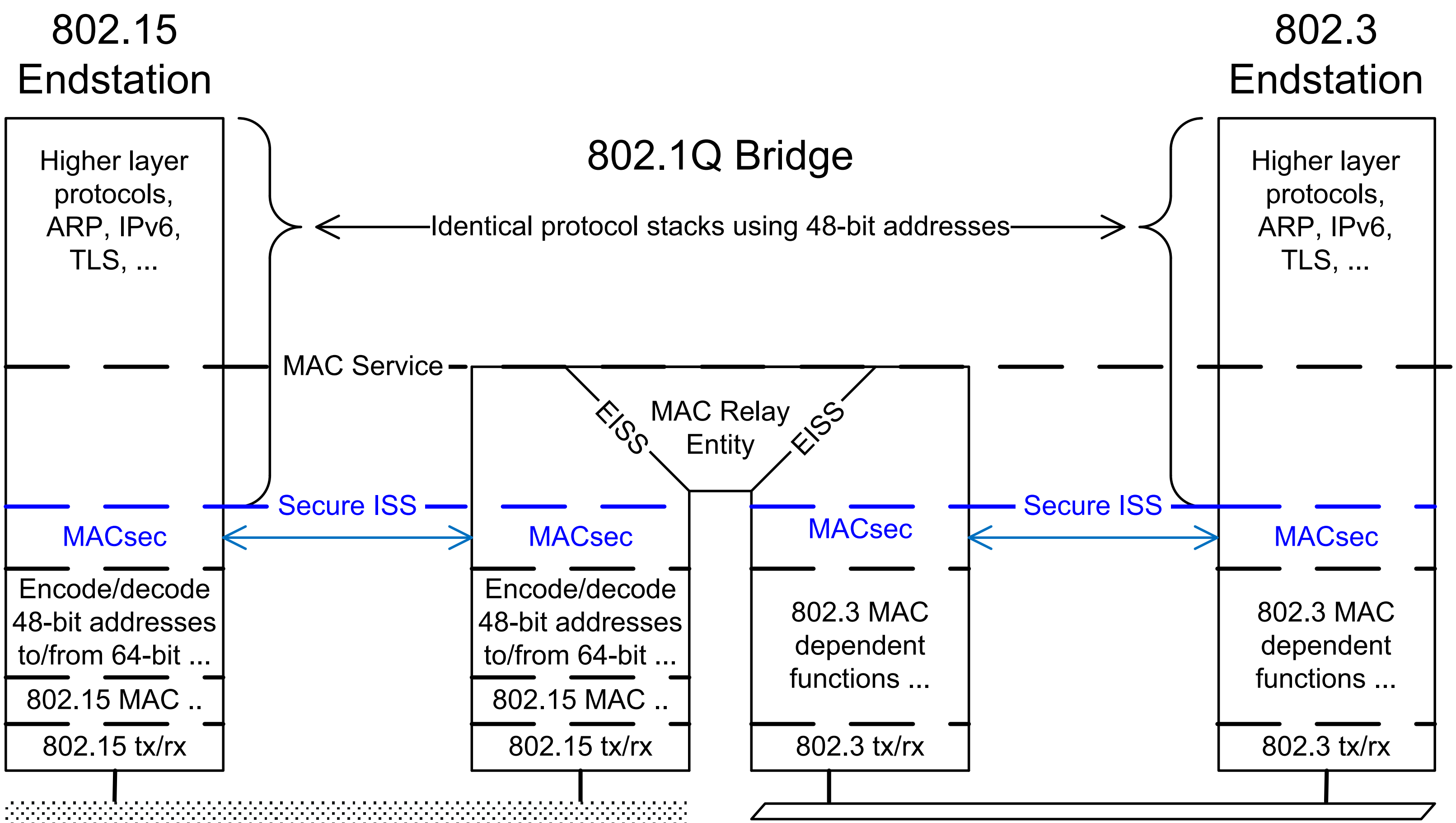
1. If the bridge management and the end stations participate in the same Group CA (secure Connectivity Association), the end stations do not need to use different cryptocaphic keys (SAKs) to protect frames destined for their end station peers and those used to communicate with the bridge.

MAC Security option (1)[1]

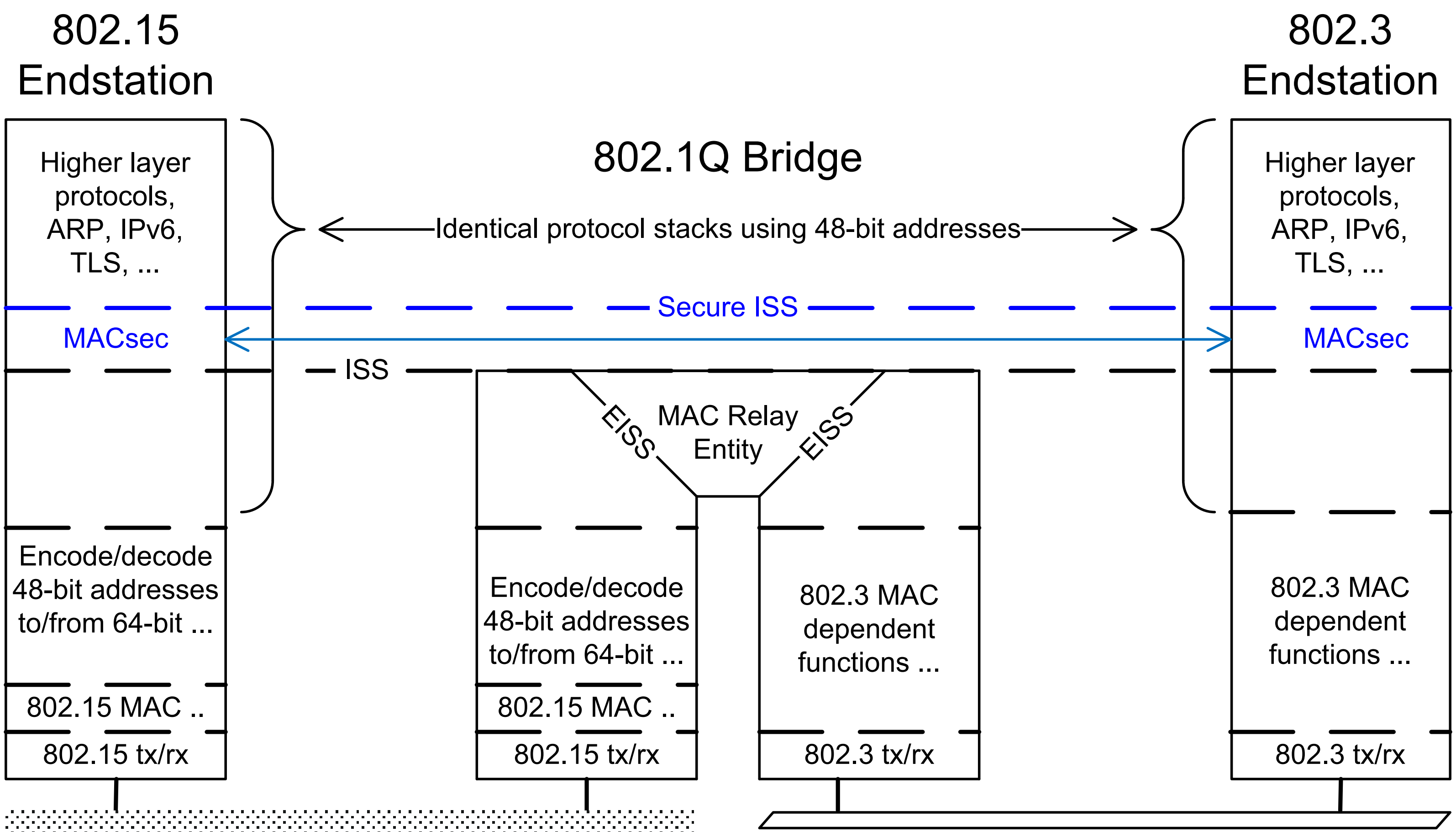


1. I don't think this is the best approach, but I include it for completeness. Some may find it natural, and it may well be a necessary first step for an endstation in a new environment.

MAC Security option (2)



MAC Security option (3)

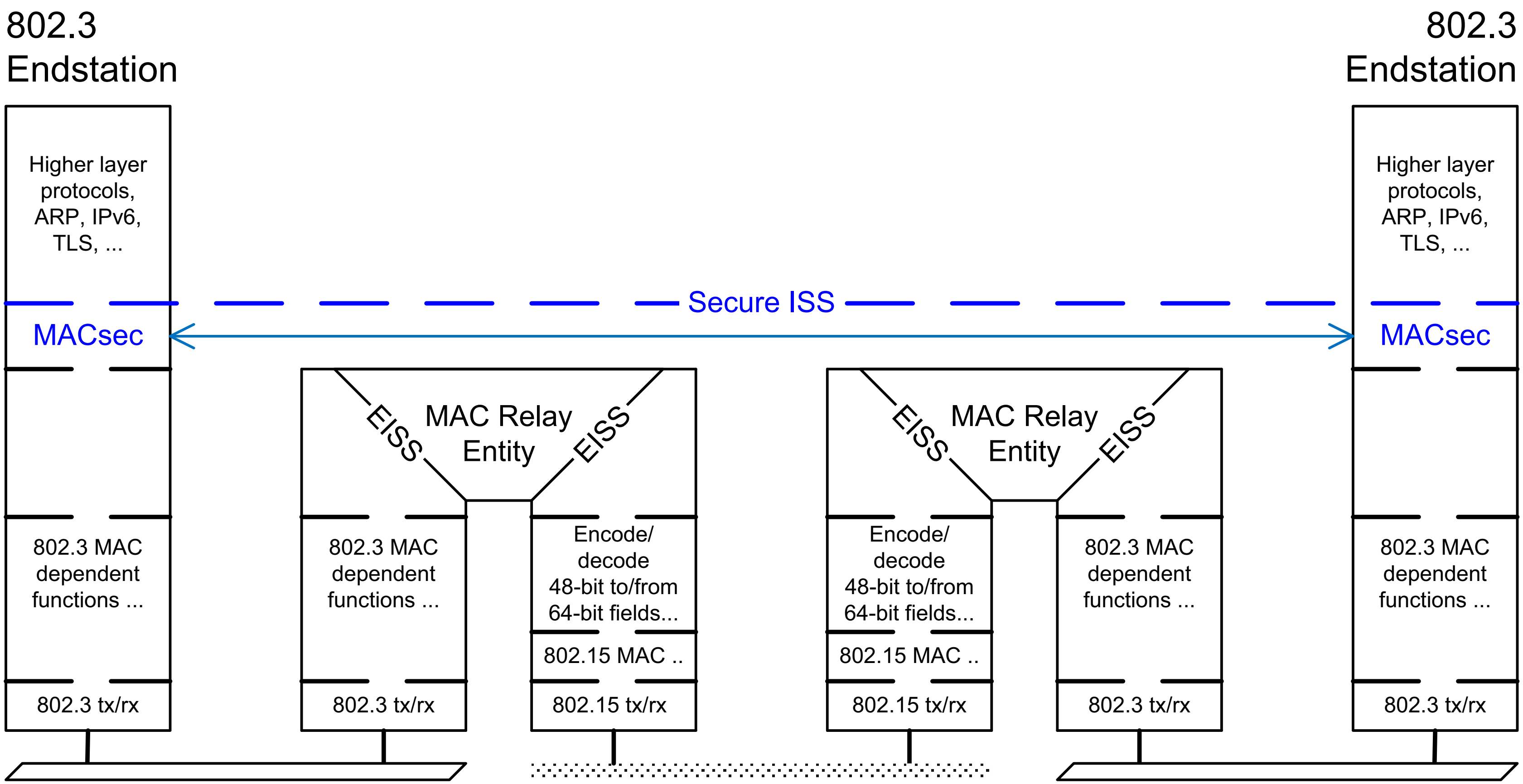


Use cases

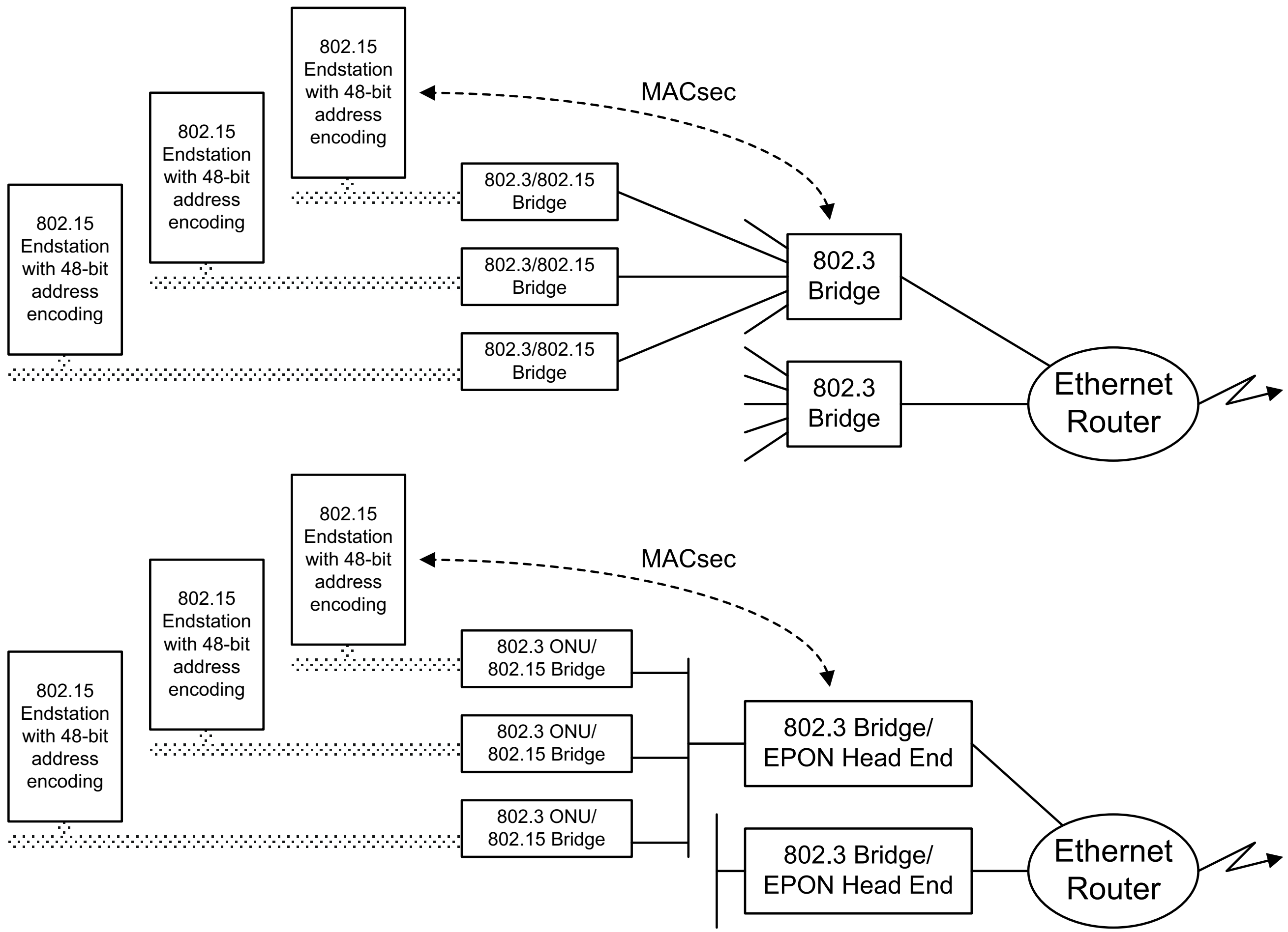
Showing just two of many, chosen to illustrate a wide range of possibilities:

1. Bridging an air gap — which might be as narrow as the thickness of a window pane — in an 802.3 link.
2. Connecting to stations in independent rooms/workspaces.

Use case — Bridging the air gap



Use case — Independent rooms/work spaces



Notes: Physical wiring alternatives shown for connecting 802.3/802.15 Bridges in the Ethernet network shown (dedicated wiring or EPON). Each 802.15 room/workspace LAN can support multiple endstations. MACsec can be used to protect/separate connectivity from individual end stations to the rightmost 802.3 Bridge shown using 'virtual ports' as per 7.5 of 802.1X-2020 [the description could be expanded to explicitly address the use case shown here if there is significant interest].