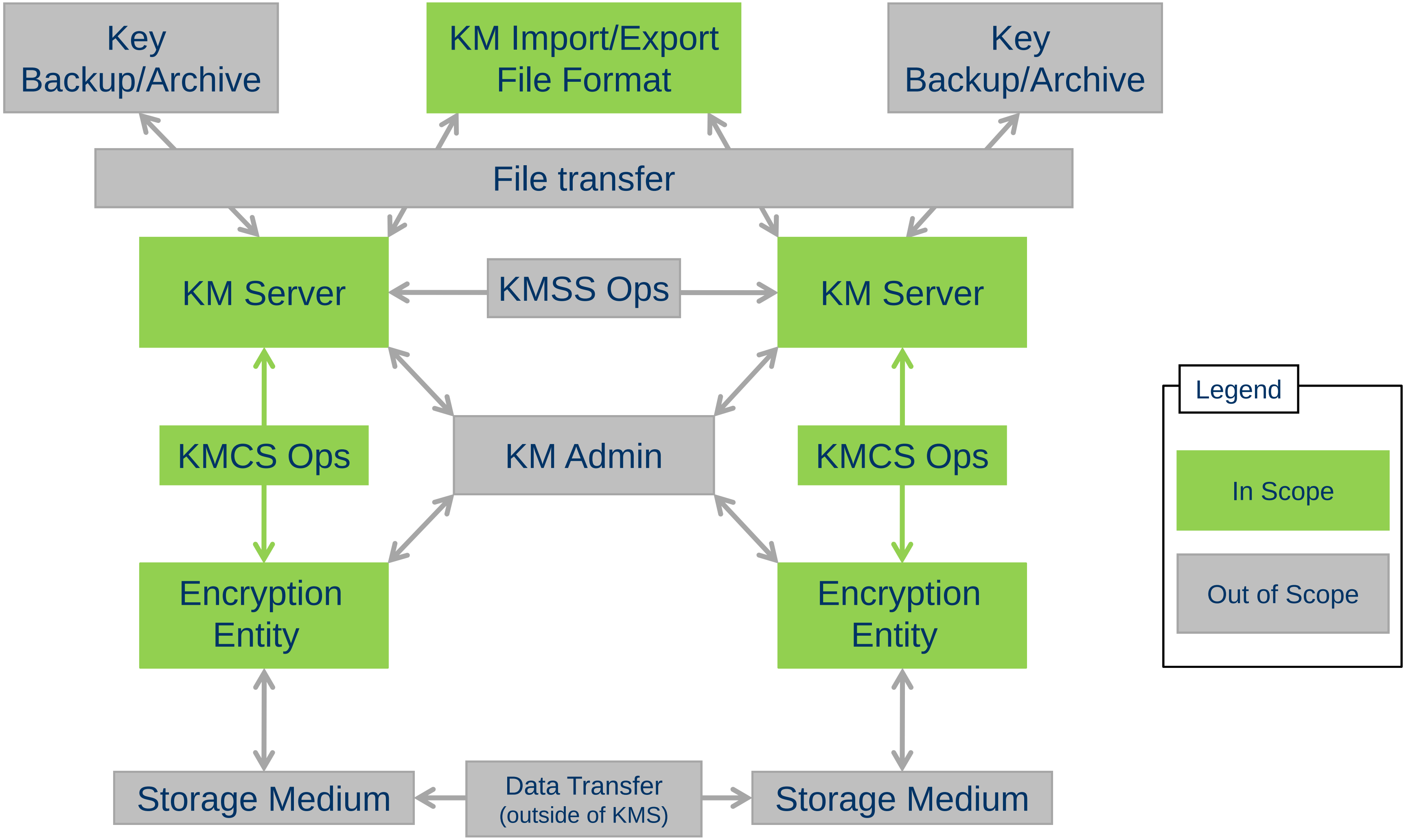
P1619.3 and KMIP

Understanding the Differences and Similarities

Understanding P1619.3

- P1619.3 is a complete architecture for managing keys used to encrypt stored data
  - This includes data stored in databases, on disk, on tape, in a file, etc…
- P1619.3 is composed of:
  - Name Spaces
    - Key, device and object globally unique identifiers
  - Objects
    - Keys and all associated attributes
    - Devices and all associated attributes
    - Groups of devices and or keys
  - Policies
    - Rules for handling of keys by key management servers and encryption devices
  - Operations
    - Generation, Retrieval, Storage of keys, policies & objects
  - Messaging
    - Format and syntax required to perform operations
  - Transport
    - TLS secure transport used to pass messaging from a KM Client to a KM Server

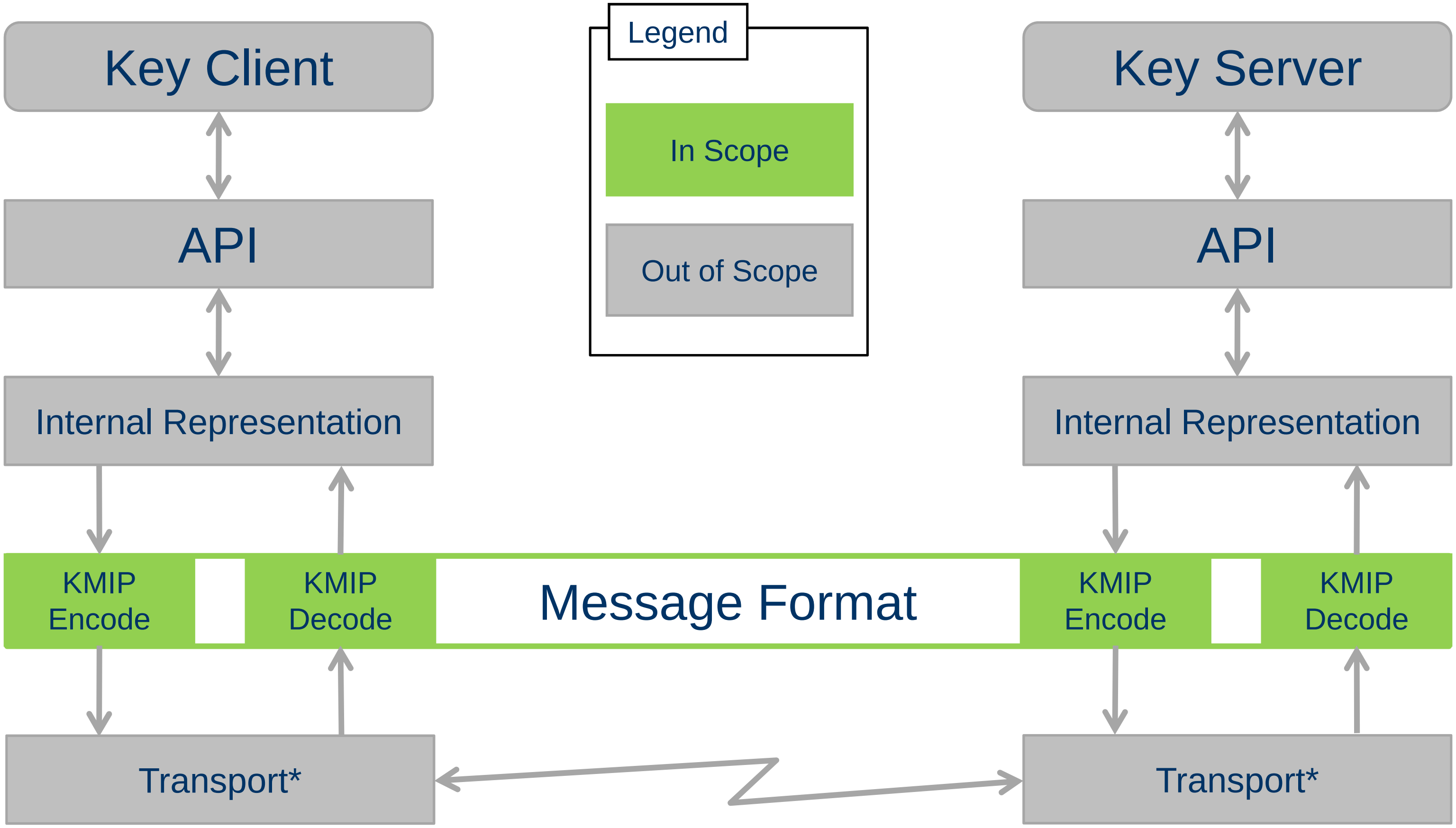
P1619.3 Architecture



Understanding KMIP

- KMIP is an application agnostic messaging format that allows for the management of keys
  - Allows a Key Client to communicate with a Key Server using a common set of messages
- KMIP consists of:
  - Tag, Type, Length, Variable (TTLV) Messaging including
    - Objects
    - Attributes
    - Client to Server Operations
    - Server to Client Operations
    - Message Contents
    - Message Format
    - Message Encoding
    - Error Handling

KMIP Transport Level Encoding



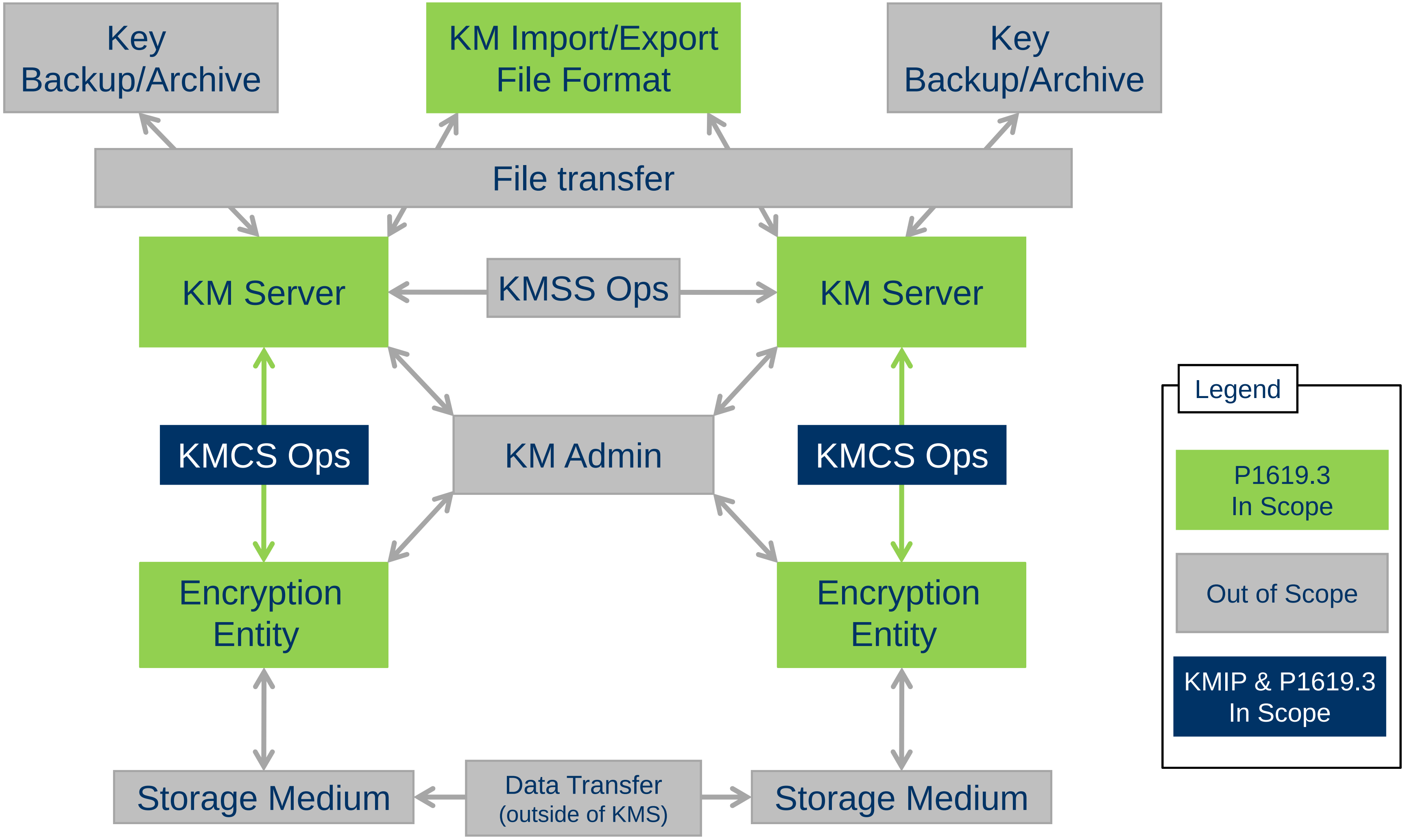
* Transport requires a secure communication protocol (e.g. HTTPS, TLS, etc…)

P1619.3 and KMIP - Mapping the Standards

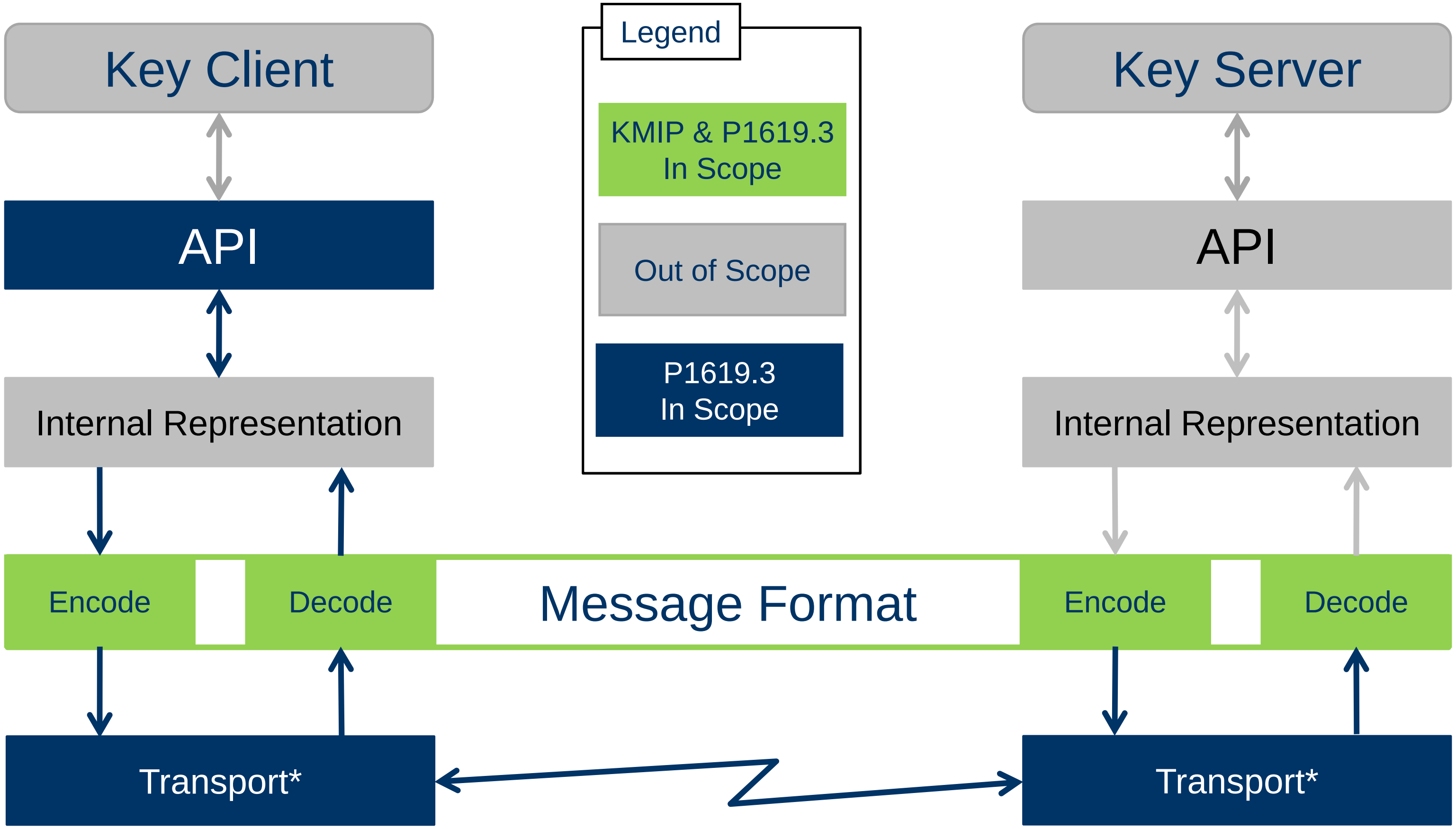
So What is the Difference? <

- KMIP currently defines the base requirements to provide key management interoperability
  - By not adding a set of architectural requirements KMIP can be used in multiple environments
  - Does not require traditional networks for connectivity
- P1619.3 is defining a complete architecture that will ensure interoperability between storage KM Clients and KM Servers
  - By specifying all requirements such as transports, messaging, name spaces and other components of the architecture interoperability is more likely between the client and server
- It is quite possible that P1619.3 could make use of KMIP when it is completed by OASIS

How KMIP Relates to P1619.3 Architecture



How P1619.3 Relates to KMIP Transport Level Encoding



* Transport requires a secure communication protocol (e.g. HTTPS, TLS, etc…)

Mapping Effort

Status

- Approximately 90% complete
  - Still some attribute mappings that need to occur for P1619.3 areas that are unclear or have proposals pending
- Currently KMIP provides for all objects, attributes, policies and operations with one exception
  - Some of these will require use of extensions as defined in current KMIP draft
- Exception
  - P1619.3 defines an additional Server to Client operation (Get Status)
    - Allows the server to request current operational state of the end point, KM Client or Cryptographic Unit (definition not complete)

IEEE P1619.3 Concerns

P1619.3 needs additional work to conform with KMIP requirements

- Proposals have been put forward to re-define P1619.3 around KMIP
- Level of effort still to be determined
- Areas that are still “To Be Defined” (TBD) require proposals

Recommendations

- Share mapping with IEEE P1619.3
  - Let them modify/comment mappings document as is
  - Request all modifications and comments be returned via Liaison

KMIP Required vs. Optional Items

- Clarification of required vs. optional objects, attributes, etc…
  - Define minimum requirements for usage of KMIP
  - Define usage requirements versus compliance requirements
    - What shall be used, what should be used, what is not required for client and server
    - Not all external standards that would make use of KMIP would require all functions that we currently require servers to implement
- Does not mean we redefine compliance requirements
  - Compliance is ours to define for interoperability requirements

> Comments & Questions