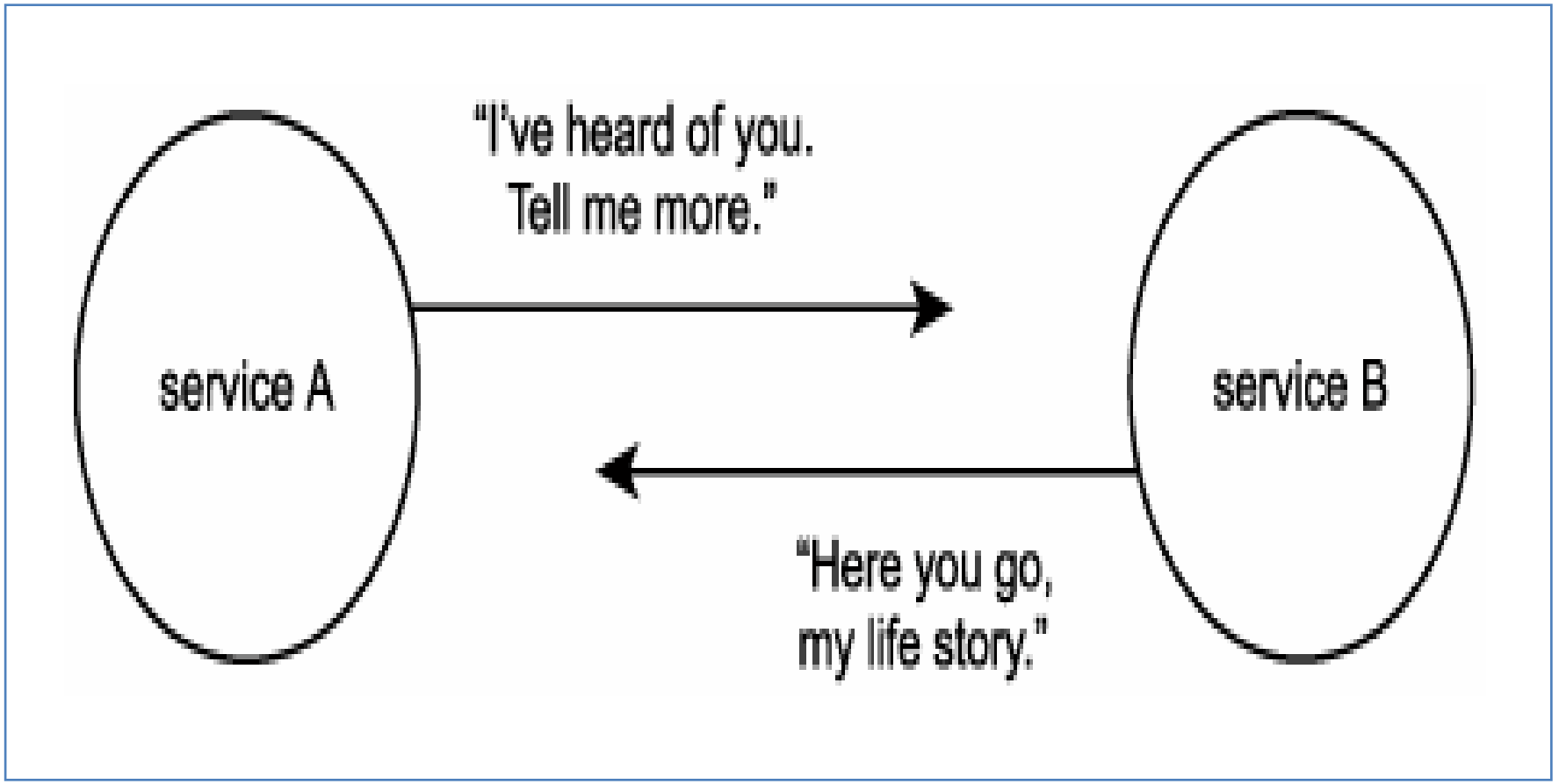
Metadata Exchange

- Requestor must have the WSDL of Provider for interaction
- The Metdata information includes
  - WSDL
  - XSD Schema
  - Policy
- The message sent by requestor must be valid according to
  - WSDL of provider
  - Policy of provider

WS-MetadataExchange

- Regardless of how much metadata a service makes available, the fact is that we still need to retrieve this information by either:
  - Manually locating it by searching for published documents
  - Manually requesting it by contacting the service provider entity (the service owner)
  - **Programmatically retrieving it via a public service registry**
  - Programmatically retrieving it by interacting with proprietary interfaces made available by the service provider entity

WS-MetadataExchange



The WS-MetadataExchange Specification

- This specification essentially allows for a service requestor to issue a standardized request message
  - that asks for some or all of the meta information relating to a specific endpoint address.

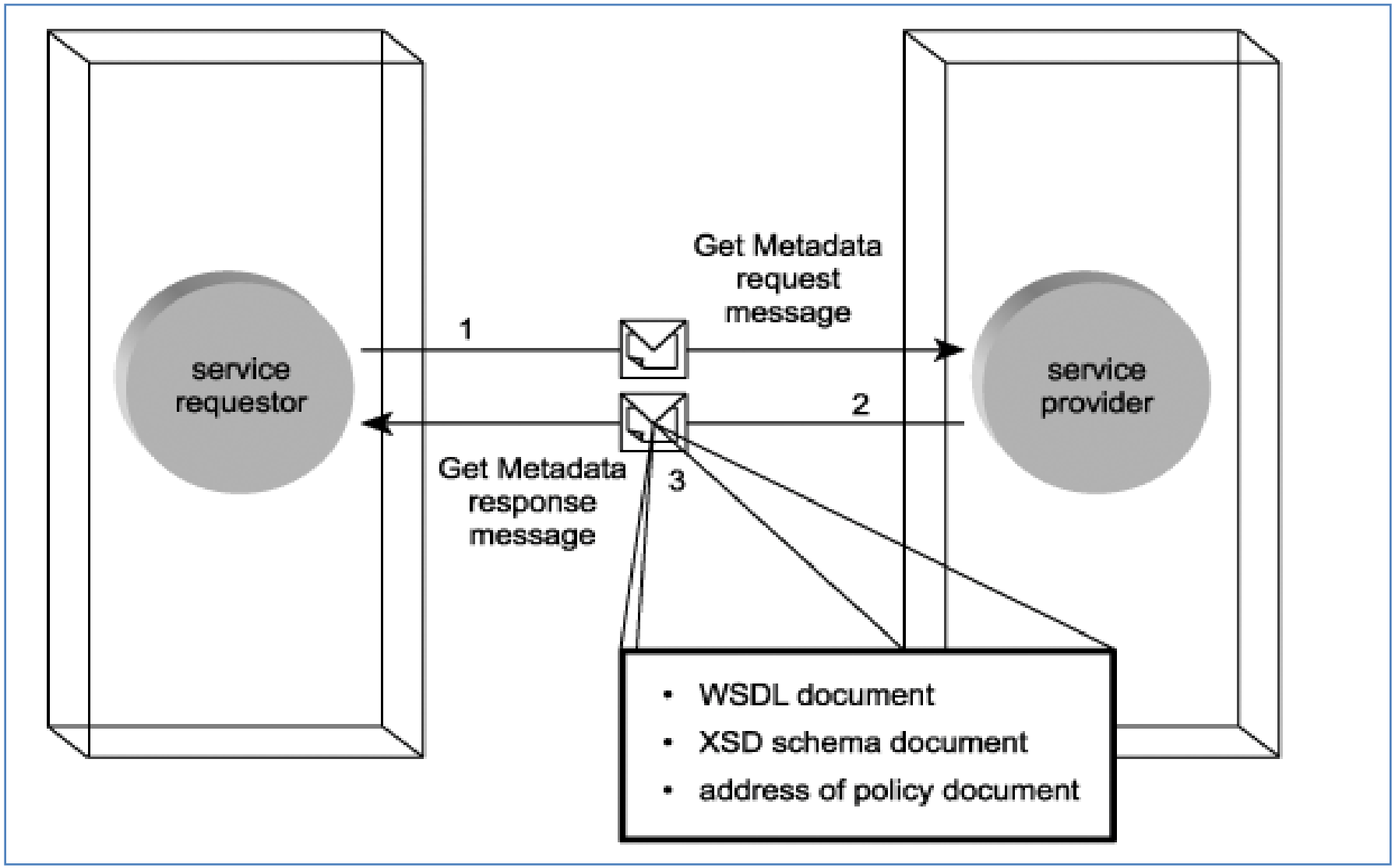
The WS-MetadataExchange Specification

- **Types of request messages**
  - Get WSDL
  - Get Schema
  - Get Policy
  - **Get Metadata**

Get Metadata Request and Response messages

- A service requestor can use metadata exchange to programmatically request available metadata documents associated with a Web service.
- To do so, it must issue a Get Metadata *request* message.
- This kicks off a standardized request and response MEP resulting in the delivery of a Get Metadata *response* message.

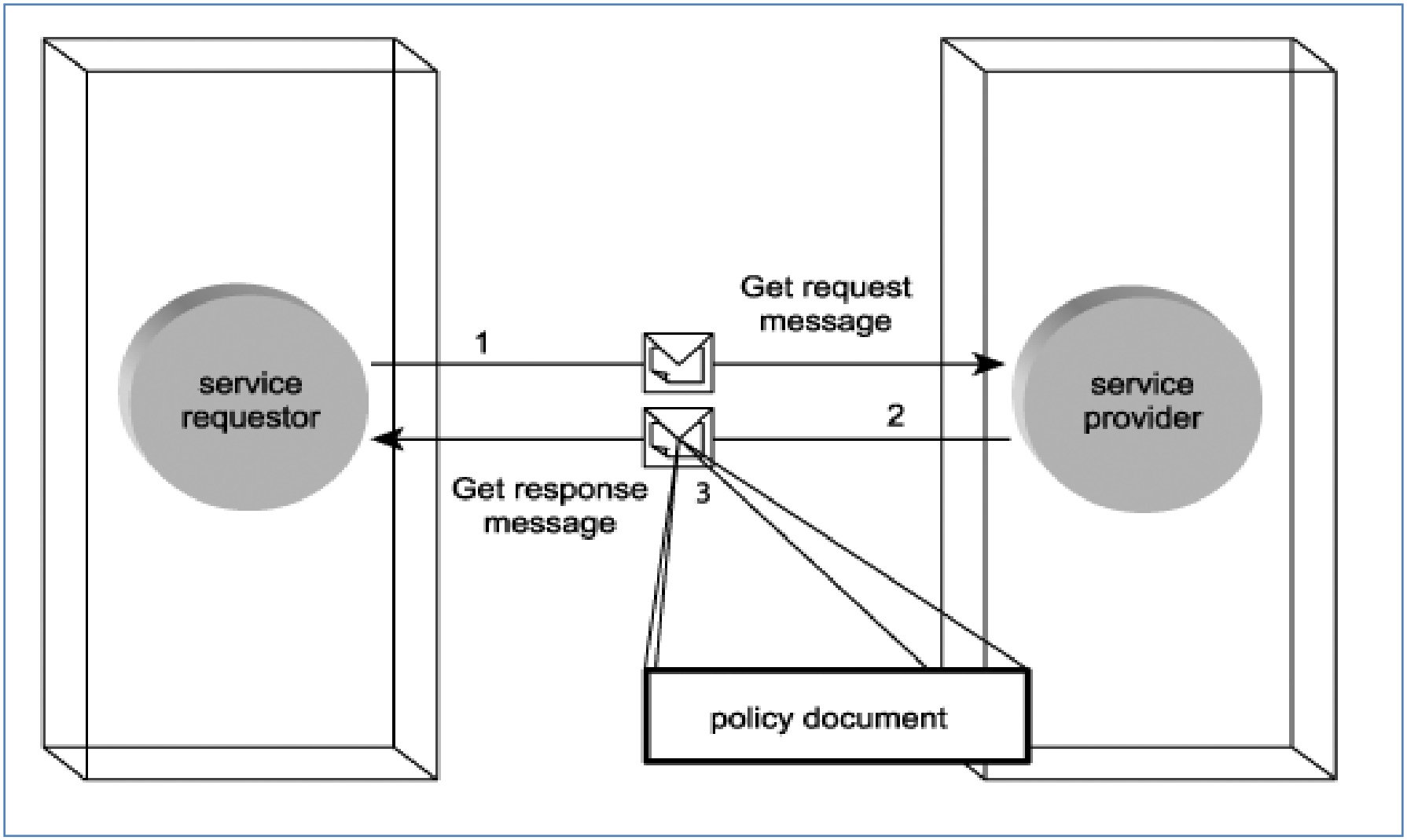
Get Metadata Request and Response messages



Get Request and Response messages

- Get Metadata Response message may contain references for some metadata information
- If the requestor needs the actual information for which reference was sent, it can use get request and get response messages.

Get Request and Response messages



Selective Retrieval of Metadata

- Meta documents describing services with comprehensive interfaces and features can be large in size
- Use of the selective get message therefore reduces the chances of unnecessary information being transported.
- Get Metadata sends only the essential information initially
- It is up to requestor to determine which actual information is required

Metadata Exchange and Service Description Discovery

- It also is important to note that metadata exchange does not really help service requestors discover service providers.
- Service registries, such as those implemented using the UDDI standard, can be used to discover service descriptions that meet certain search criteria.
- Hence service registries can be used in conjunction with metadata exchange messages.

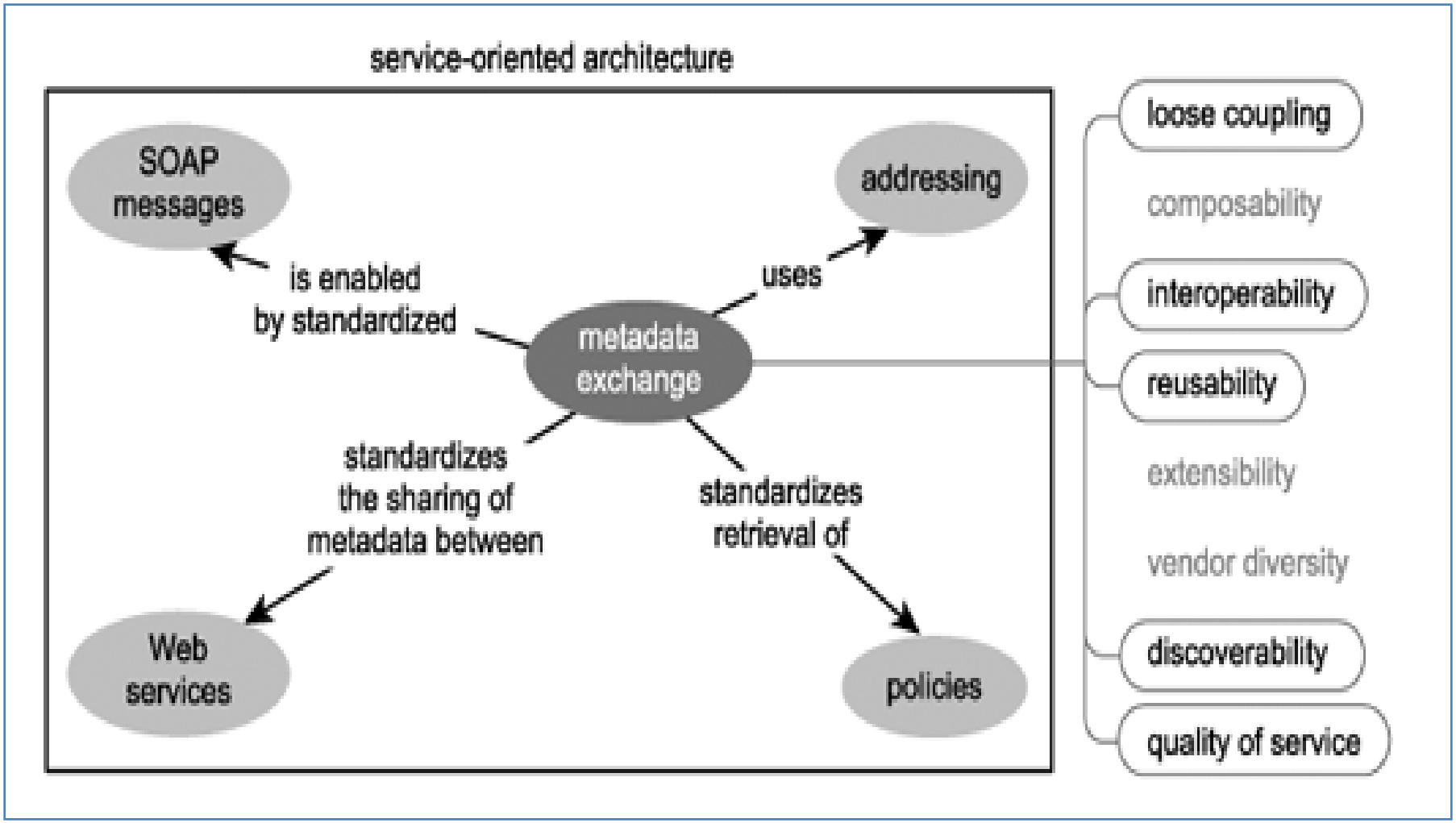
Metadata Exchange and Service Description Discovery

- Essentially, a service requestor could first query a public registry to retrieve the endpoint addresses of any Web service candidates that appear to provide the sought-after features.
- The same requestor could then employ metadata exchange to contact each candidate and request associated metadata documents.
- This would give the service requestor more information to better assess which service provider it should be working with.

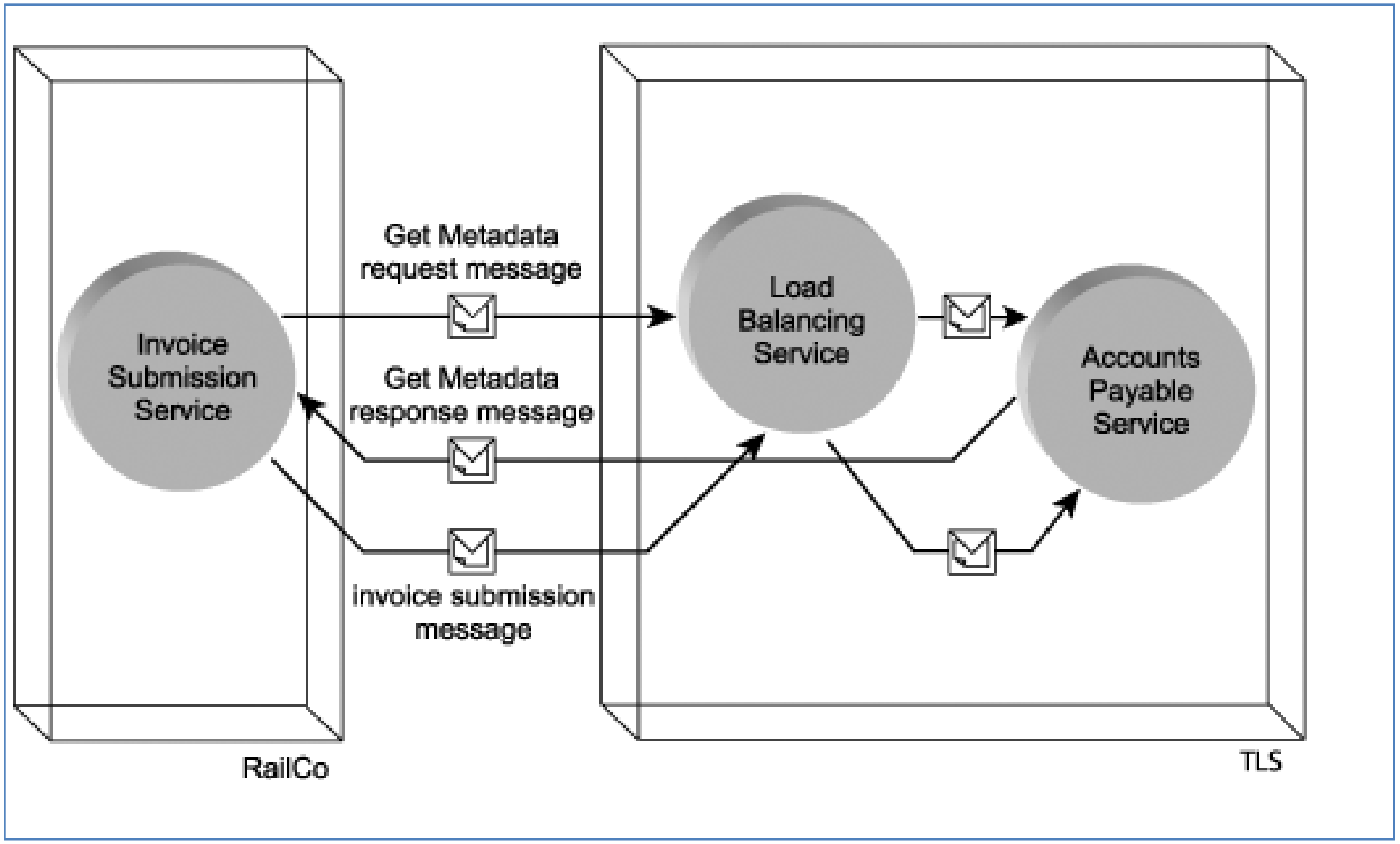
Metadata Exchange and Version Control

- As services evolve, the nature and scope of the functionality they deliver can undergo alterations.
- This can lead to new versions of a service's WSDL, XSD schema, or policy documents.
- Service-oriented application that supports metadata exchange can allow service requestors
  - to retrieve the latest service contract as often as they like.

Metadata Exchange and SOA



Case Study


Get Metadata
request message
Get Metadata
response message
invoice submission
message
Invoice
Submission
Service
Load
Balancing
Service
Accounts
Payable
Service
RailCo
TLS

So far, WS-Metadata exchange

- Metadata exchange allows service requestors to issue request messages to retrieve metadata for service providers.
- The WS-MetadataExchange specification standardizes two types of request messages: the Get Metadata request and the Get Request.
- Metadata exchange assists in improving the service description discovery process and in alleviating version control issues related to service meta information.
- Automated metadata retrieval leads to several standardized improvements within SOA and reinforces the loosely coupled nature of autonomous services.

WS-Security

- Security framework for Web services

- A family of security extensions parented by the WS-Security specification comprise such a framework

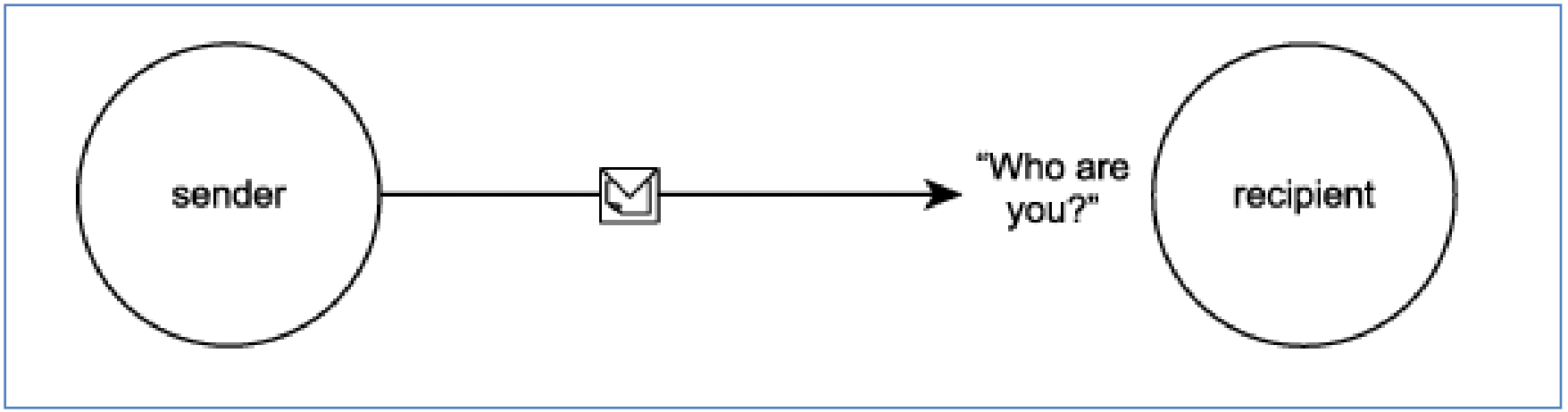
  - WS-Security
  - XML-Signature
  - XML-Encyption

Common Security Requirements

- Identity
- Authentication
- Authorization
- Confidentiality
- Integrity

Identification

- For a service requestor to access a secured service provider, it must first provide information that expresses its origin or owner.
- This is referred to as making a claim.



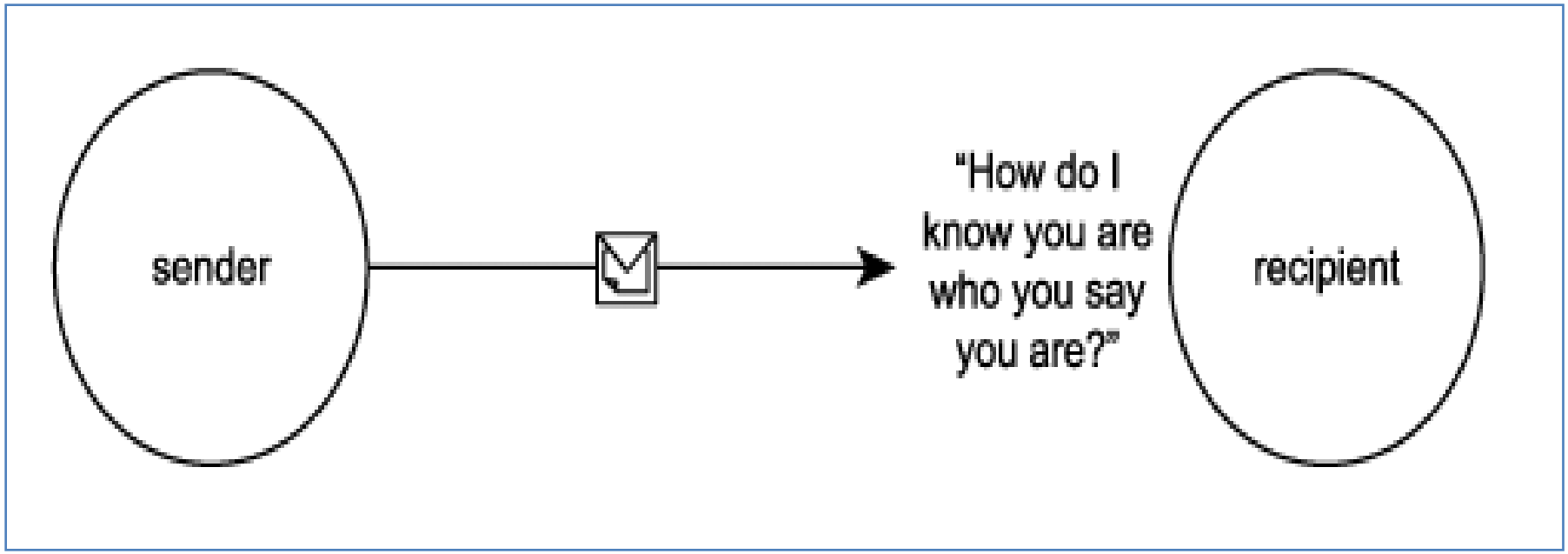
Identification

- Claims are represented by identification information stored in the SOAP header.

- WS-Security establishes a standardized header block that stores this information, at which point it is referred to as a token.

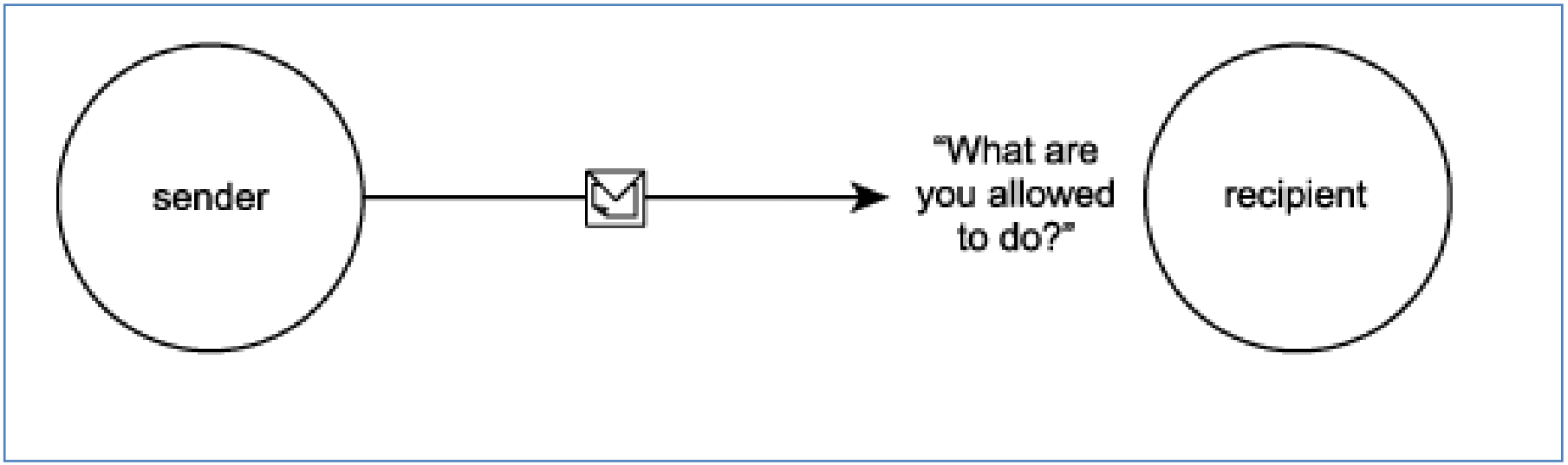
Authentication

- Authentication requires that a message being delivered to a recipient prove that the message is in fact from the sender that it claims to be.
  - Provide a proof of identity



Authorization

- Once authenticated, the recipient of a message may need to determine what the requestor is allowed to do.
- This is called authorization.



How can we propagate the authentication and authorization information for a service requestor across multiple services behind the initial service provider?

Single Sign-on

- The use of a single sign-on technology allows a service requestor to be authenticated once
  - and then have its security context information shared with other services that the requestor may then access without further authentication.
- There are three primary extensions that support the implementation of the single sign-on concept
  - SAML (Security Assertion Markup Language)
  - .NET Passport
  - XACML (XML Access Control Markup Language)

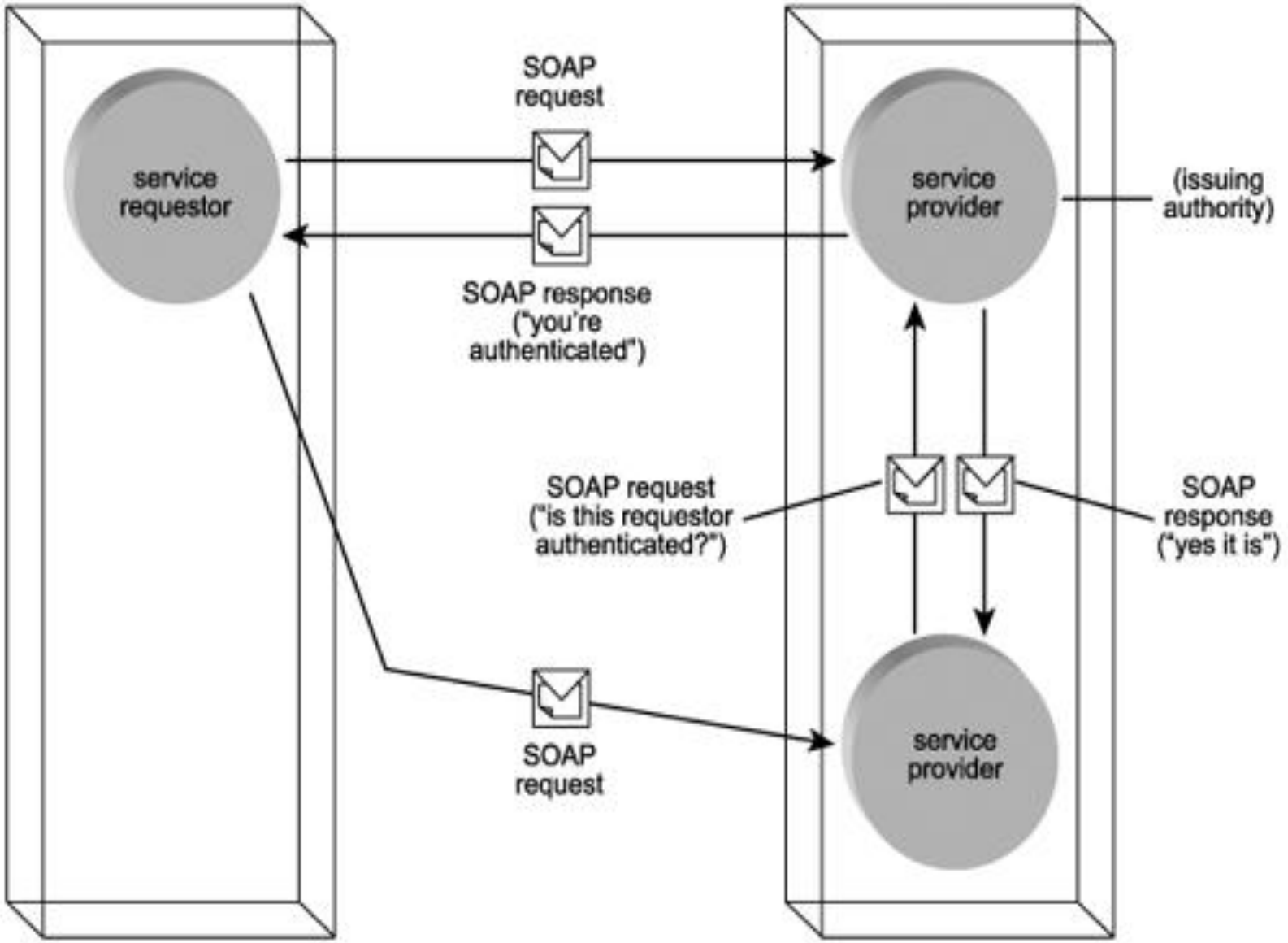
SAML

- SAML implements a single sign-on system in which the point of contact for a service requestor can also act as an issuing authority.

- This permits the underlying logic of that service not only to authenticate and authorize the service requestor,
  - but also to assure the other services that the service requestor has attained this level of clearance.

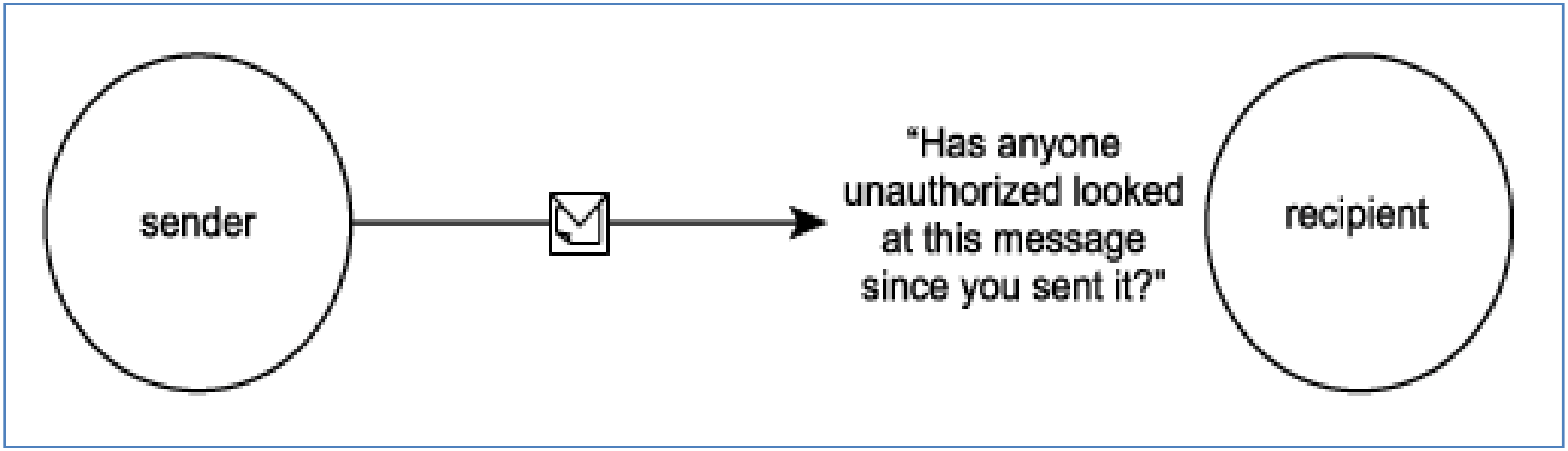
SAML

- Other services that the service requestor contacts, therefore, do not need to perform authentication and authorization steps.
- Instead, upon receiving a request, they simply contact the issuing authority to ask for the authentication and authorization clearance it originally obtained.
- The issuing authority provides this information in the form of assertions that communicate the security details.
  - authentication assertions
  - authorization assertions


service requestor
SOAP request
SOAP response ("you're authenticated")
service provider
(issuing authority)
SOAP request ("is this requestor authenticated?")
SOAP response ("yes it is")
SOAP request
service provider

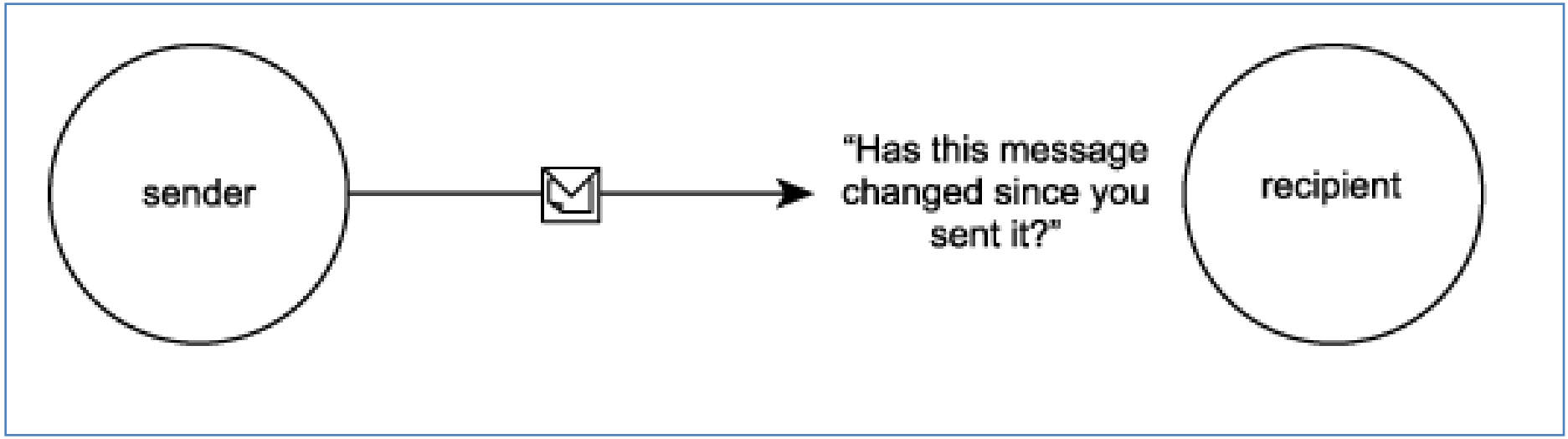
Confidentiality

- Confidentiality is concerned with protecting the privacy of the message contents.
- A message is considered to have remained confidential if no service or agent in its message path not authorized to do so viewed its contents.



Integrity

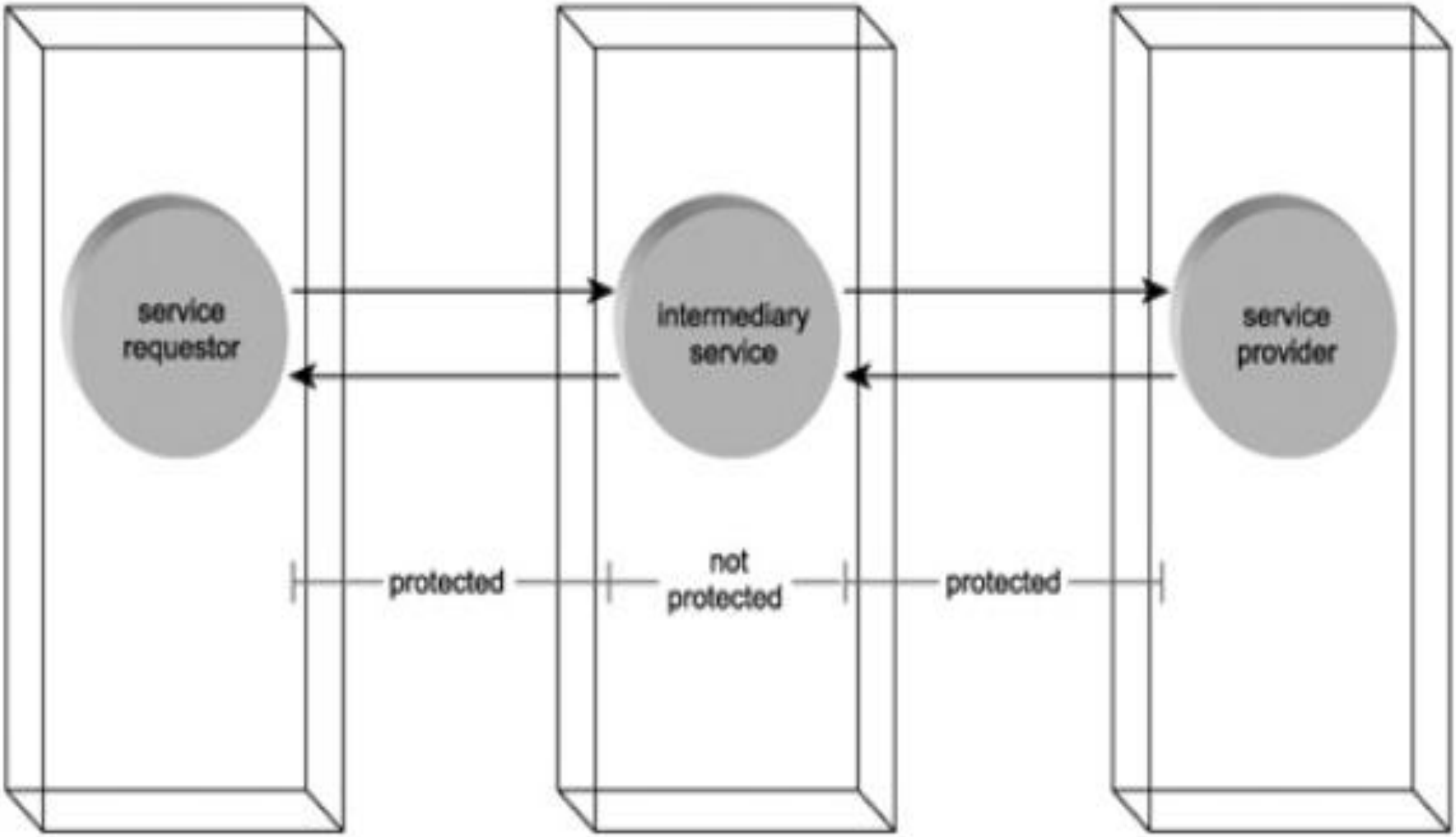
- Integrity ensures that a message has not been altered since its departure from the original sender.
- This guarantees that the state of the message contents remained intact from the time of transmission to the point of delivery.



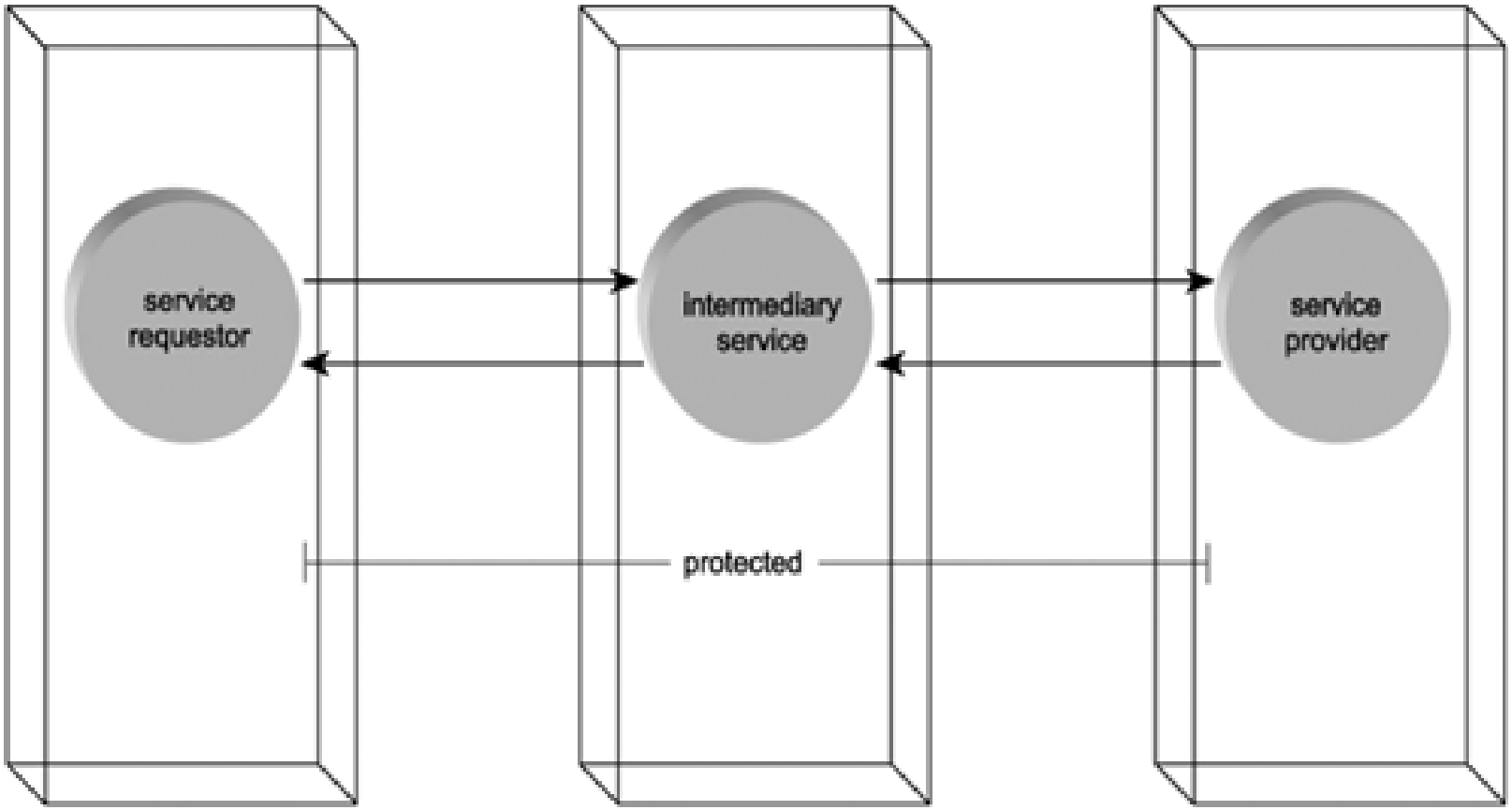
Transport-level security

- The type of technology used to protect a message determines the extent to which the message remains protected while making its way through its message path.
- Secure Sockets Layer (SSL), for example, is a very popular means of securing the HTTP channel upon which requests and responses are transmitted.
- However, within a Web services-based communications framework, it can only protect a message during the transmission between service endpoints. Hence, SSL only affords us transport-level security

Transport-level security



Message-level security



Encryption

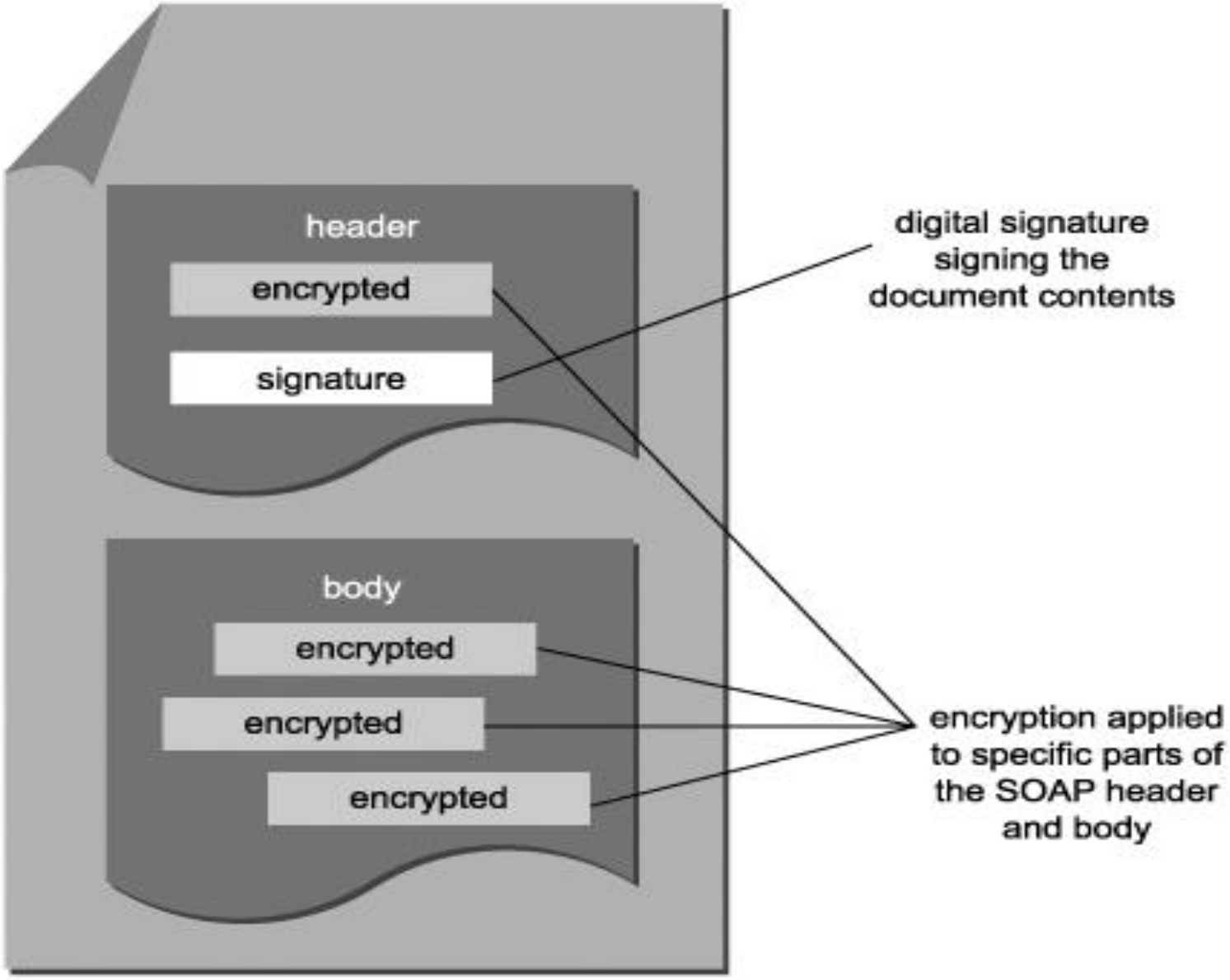
- XML-Encryption, an encryption technology designed for use with XML, is a cornerstone part of the WS-Security framework.

- It provides features with which encryption can be applied to an entire message or only to specific parts of the message (such as the password).

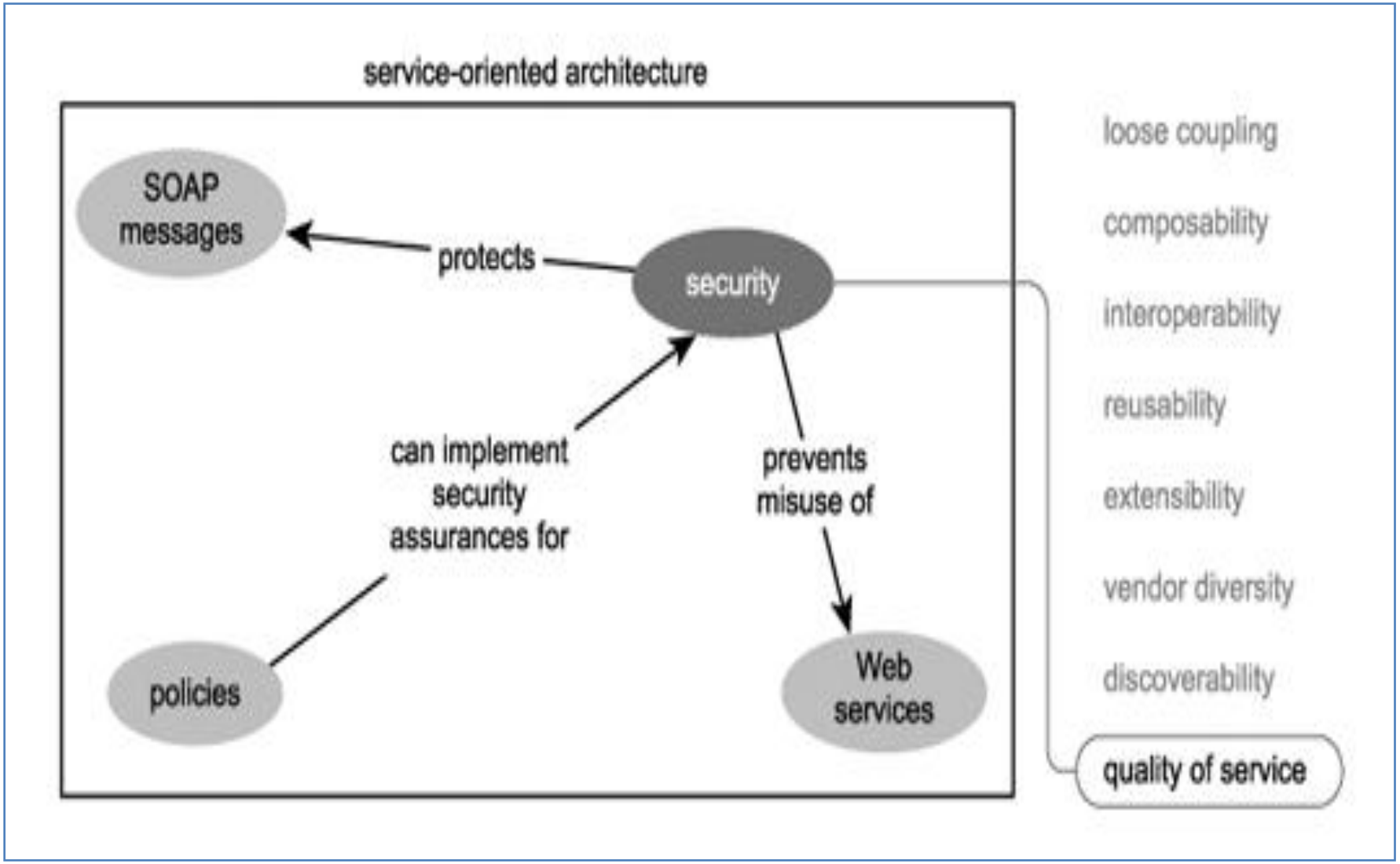
Digital signatures

- XML-Signature provides features that allow for an XML document to be accompanied by a special algorithm-driven piece of information that represents a digital signature.

- This signature is tied to the content of the document so that verification of the signature by the receiving service only will succeed if the content has remained unaltered since it first was sent.


header
encrypted
signature
body
encrypted
encrypted
encrypted
digital signature signing the document contents
encryption applied to specific parts of the SOAP header and body

Security and SOA



So far, WS-Security

- Security within SOA is a multi-faceted subject matter that encompasses the feature set of numerous specifications.
- The WS-Security framework governs a subset of these specifications, and establishes a cohesive and composable security architecture.
- The primary aspects of security addressed by these specifications are identification, authentication, authorization, integrity, and confidentiality, as well as non-repudiation.
- Two primary technologies for preserving the integrity and confidentiality of XML documents are XML-Encryption and XML-Signature