SunSoft
A Sun Microsystems Company

GSS-API

Connectathon ‘98

Jack Kabat

jkabat@eng.sun.com

Outline

- Overview
- Architecture description
- GSS-API concepts
- Example
- Role of mechanisms
- Summary

GSS-API

What is it ?

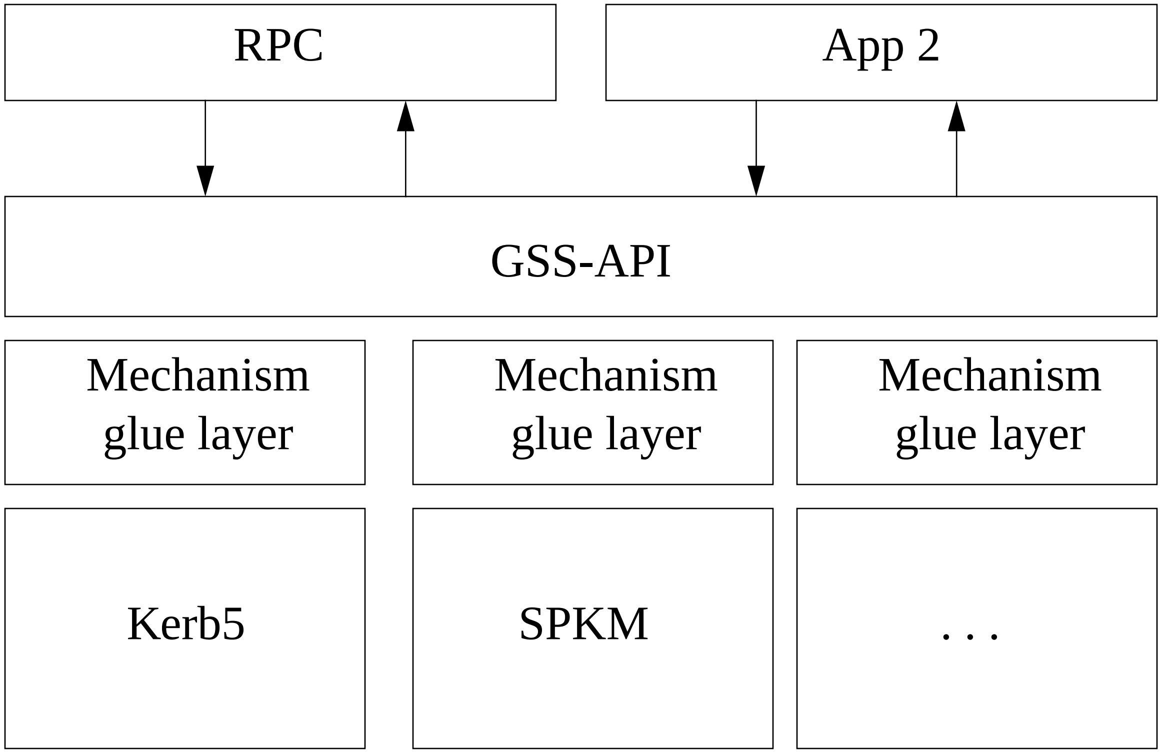
- Generic Security Services Application Programming Interface
- IETF RFC 2078 and language bindings

Goals

- supports a range of security services such as authentication, integrity, and privacy
- allows for plug-ability of different security mechanisms without changing application layer
- transport independent

Architectural Overview

- enables to change security mechanism without affecting the application layer



GSS-API Concepts

Credentials

- entity’s security identity

Contexts

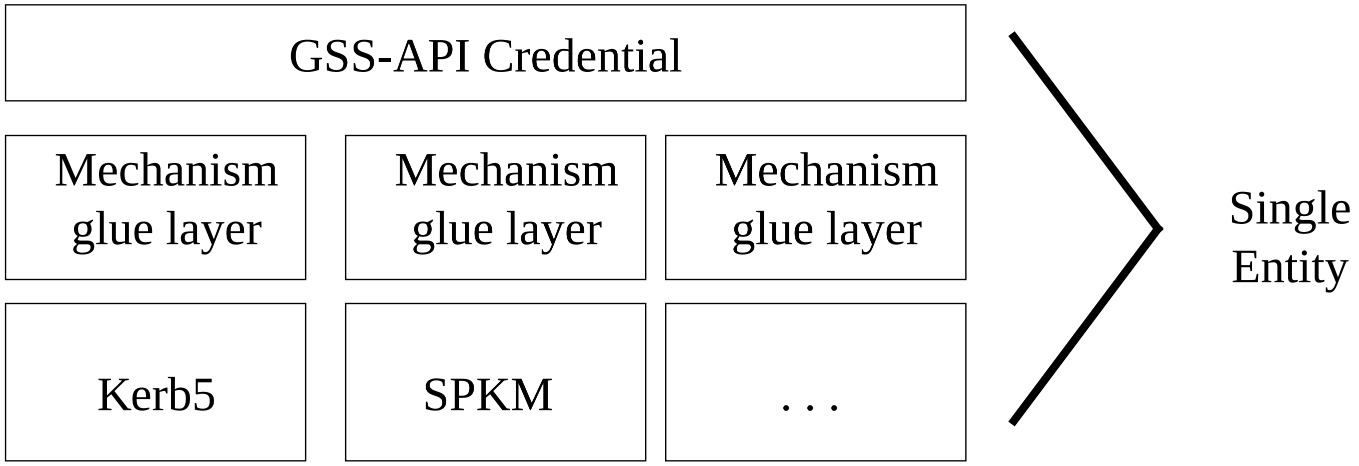
- established between peers
- handshake protocol

Per-message operations

- authentication, integrity and privacy services available over the established context
- mechanism dependent

Credentials

- represents a single entity
- initiator, acceptor, or both
- may contain multiple credential elements
- identifies data needed by each mechanism in order to establish contexts on behalf of a particular principal



Contexts

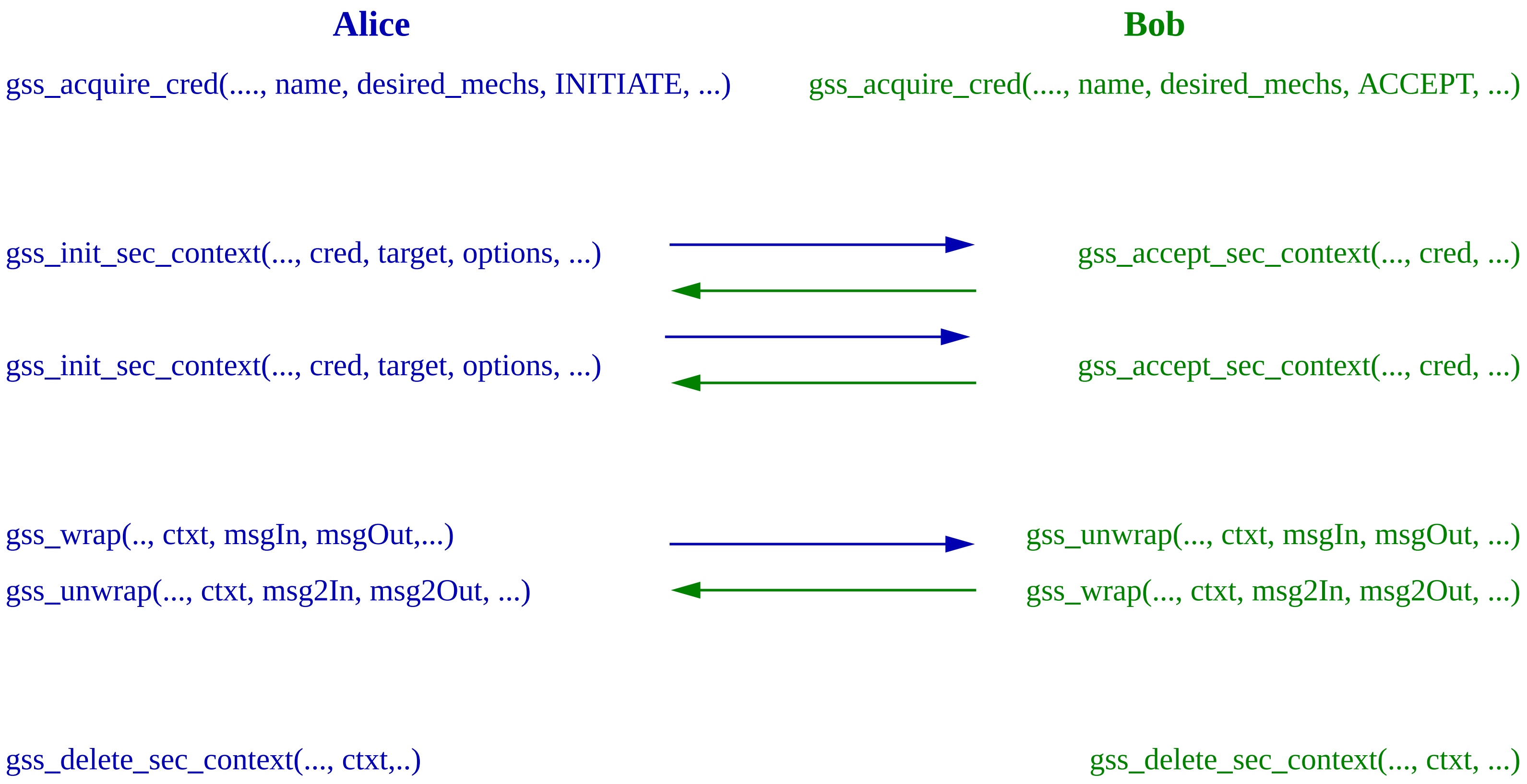
- established between peers using locally obtained credentials
- allows for negotiation of security services (mutual authentication, replay detection, sequencing, algorithm negotiation)
- flexible in the number of tokens exchanged between peers
- transport independent
- support for multiple simultaneous contexts between peers using same credential

Per Message Services

- available on established contexts
- per message authentication, integrity, privacy, sequence and replay detection
- transport independence
- QOP field selects the level of protection

Putting It All Together

Example GSS-API peers



Alice

Bob

gss_acquire_cred(...., name, desired_mechs, INITIATE, ...)

gss_acquire_cred(...., name, desired_mechs, ACCEPT, ...)

gss_init_sec_context(..., cred, target, options, ...)

gss_accept_sec_context(..., cred, ...)

gss_init_sec_context(..., cred, target, options, ...)

gss_accept_sec_context(..., cred, ...)

gss_wrap(.., ctxt, msgIn, msgOut,...)

gss_unwrap(..., ctxt, msgIn, msgOut, ...)

gss_unwrap(..., ctxt, msg2In, msg2Out, ...)

gss_wrap(..., ctxt, msg2In, msg2Out, ...)

gss_delete_sec_context(..., ctxt,..)

gss_delete_sec_context(..., ctxt, ...)

Role of Mechanisms

- defines token formats, protocols, and procedures to implement the services available through the GSS-API
- provides cryptographic routines to achieve desired security levels
- implements very different security technologies
  e.g. symmetric key, public key, hardware devices

Summary

- single API for wide range of security services
- enables to dynamically plug in security mechanisms
- drives security services implementation to mechanism layer
- transport independent