GSI Health

The “Direct Project” Reference Implementation Architecture






NwHIN Direct Approach

- Develop specifications for a secure, scalable, standards-based way to establish universal health addressing and transport for participants
- Send encrypted health information directly to known, trusted recipients over the Internet (Push Model)
- Participants include providers, laboratories, hospitals, pharmacies and patients
- Standards and service descriptions designed to address the key Stage 1 requirements for Meaningful Use
- http://wiki.directproject.org/home

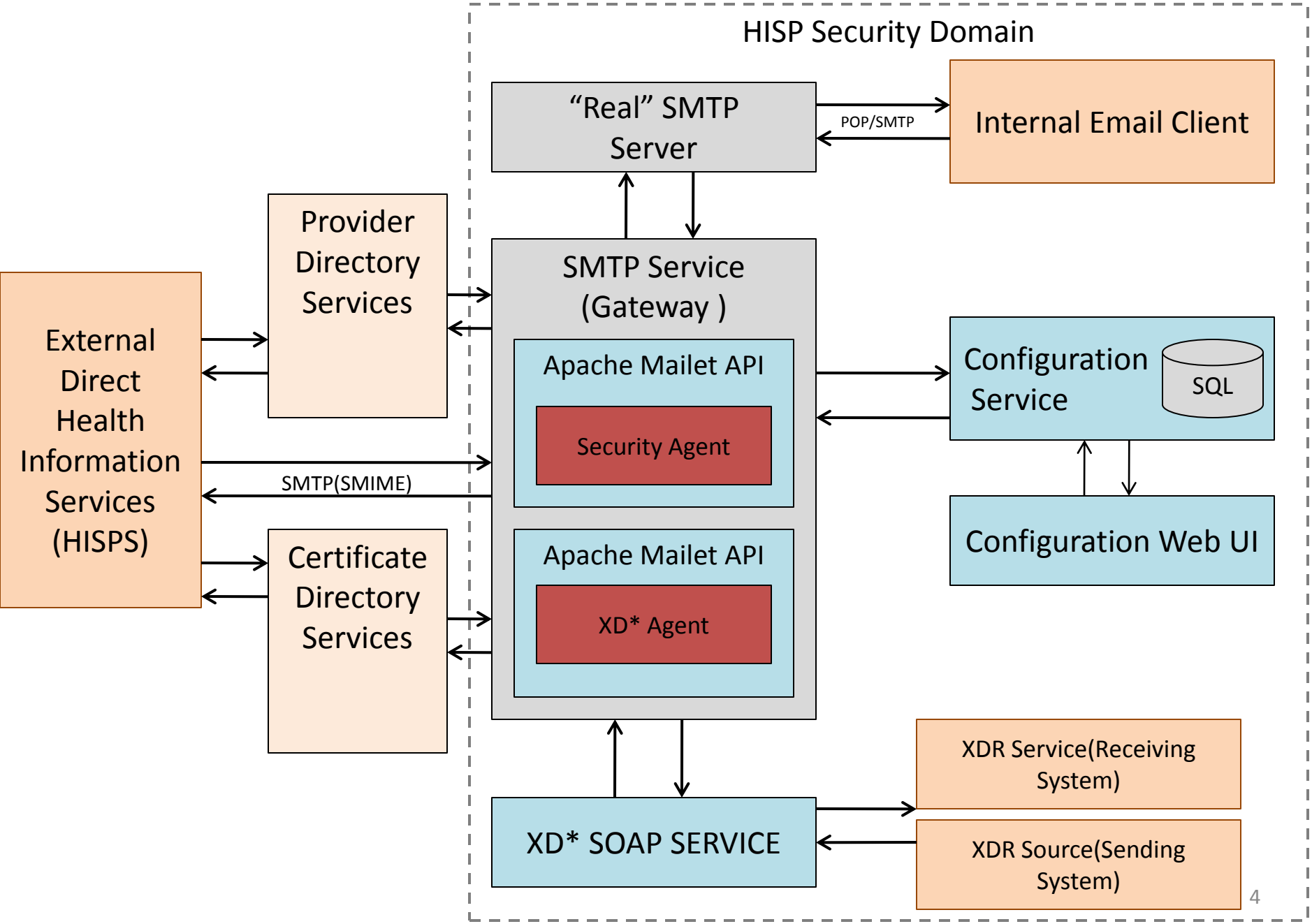
GSI Health

Direct Reference Implementation

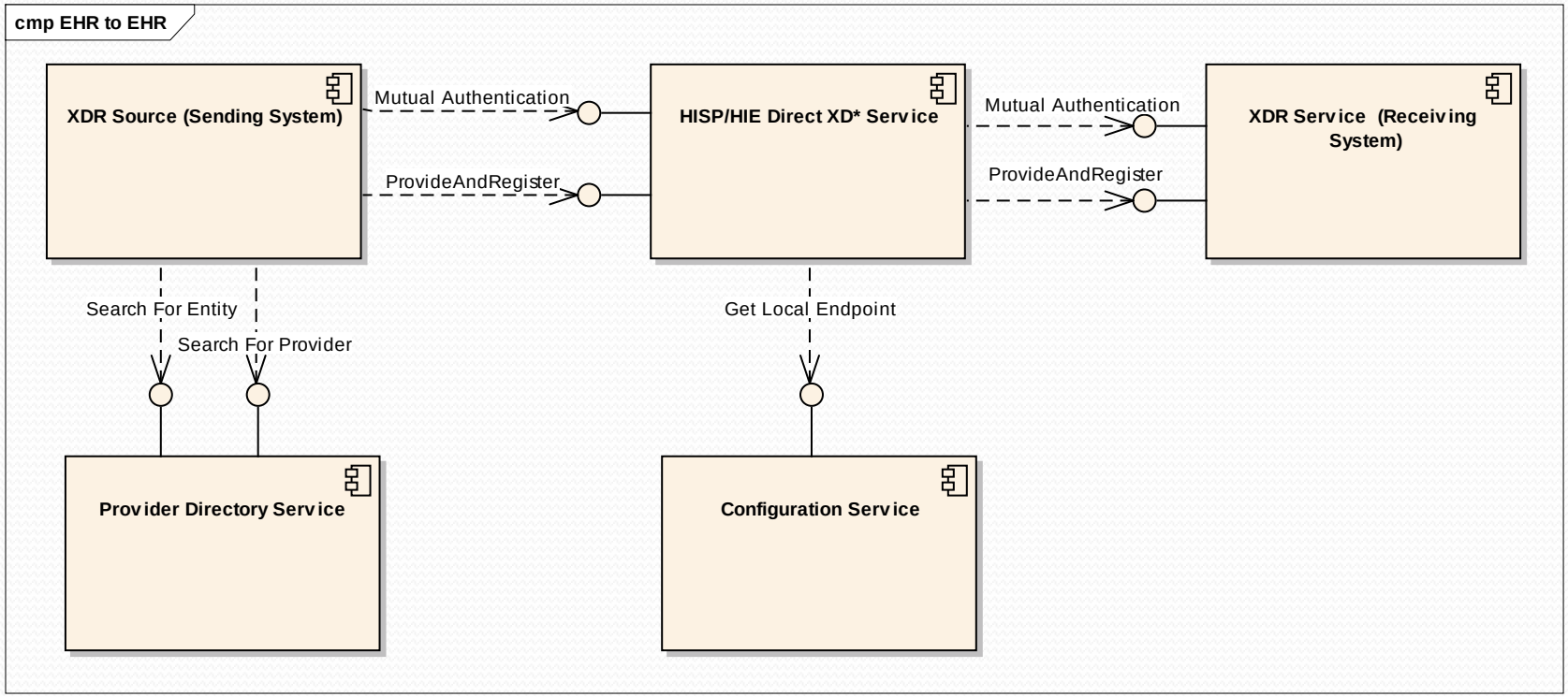
- Open-source reference implementation and associated libraries implementing the Direct Project specification
- Implementations in Java and in C Sharp(.Net)
- Actually implemented and used in several pilot projects. New York Hudson Valley and Rhode Island have hooked their pilots together (HISP to HISP)
- Multiple EHRs vendors in the pilots
- http://wiki.directproject.org/Reference+Implementation+Workgroup

GSI Health

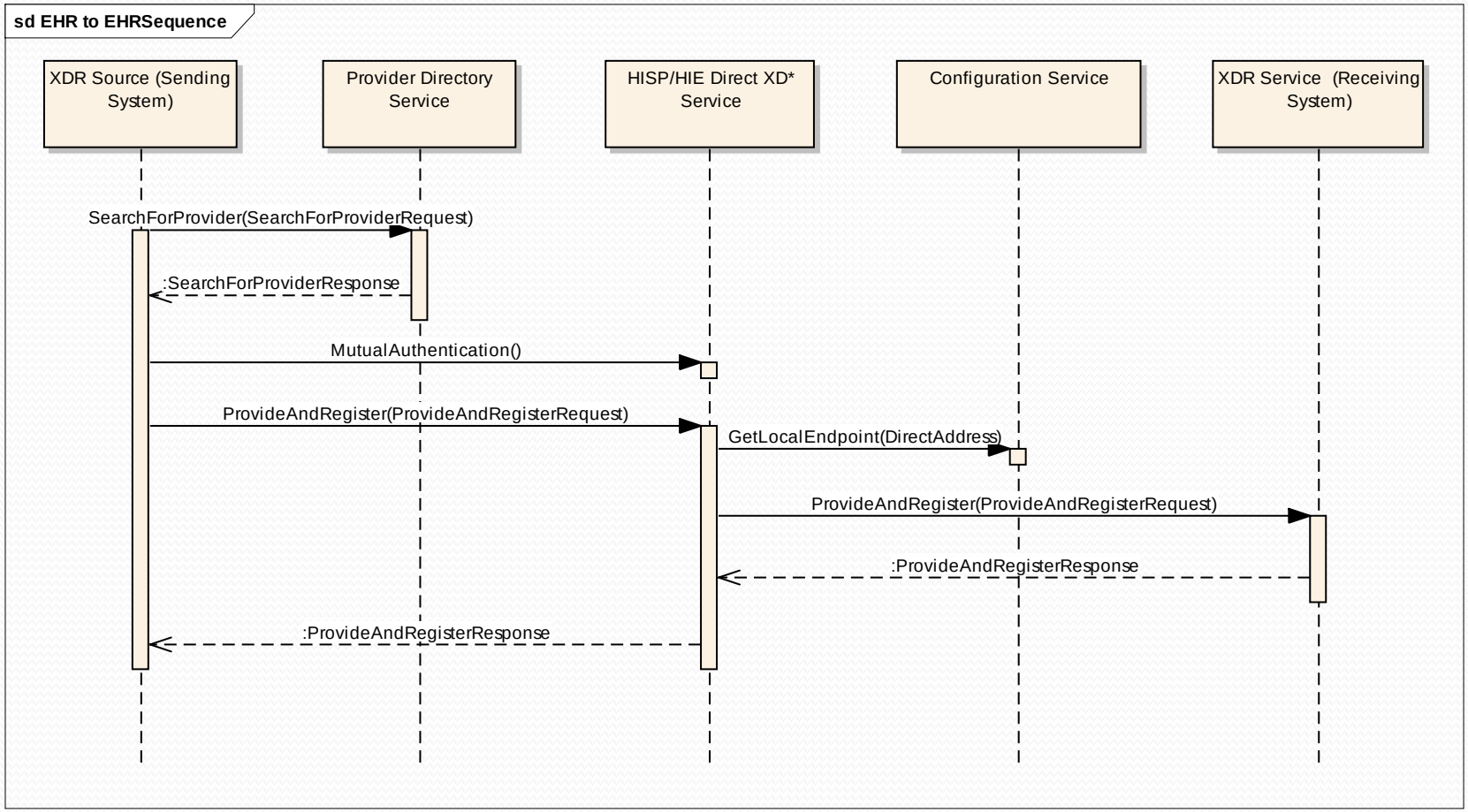
JAVA REFERENCE IMPLEMENTATION PLUS DIRECTORY SERVICES



EHR to EHR


cmp EHR to EHR
XDR Source (Sending System)
Mutual Authentication
ProvideAndRegister
HISP/HIE Direct XD* Service
Mutual Authentication
ProvideAndRegister
XDR Service (Receiving System)
Search For Entity
Search For Provider
Get Local Endpoint
Provider Directory Service
Configuration Service

EHR to EHR Sequence

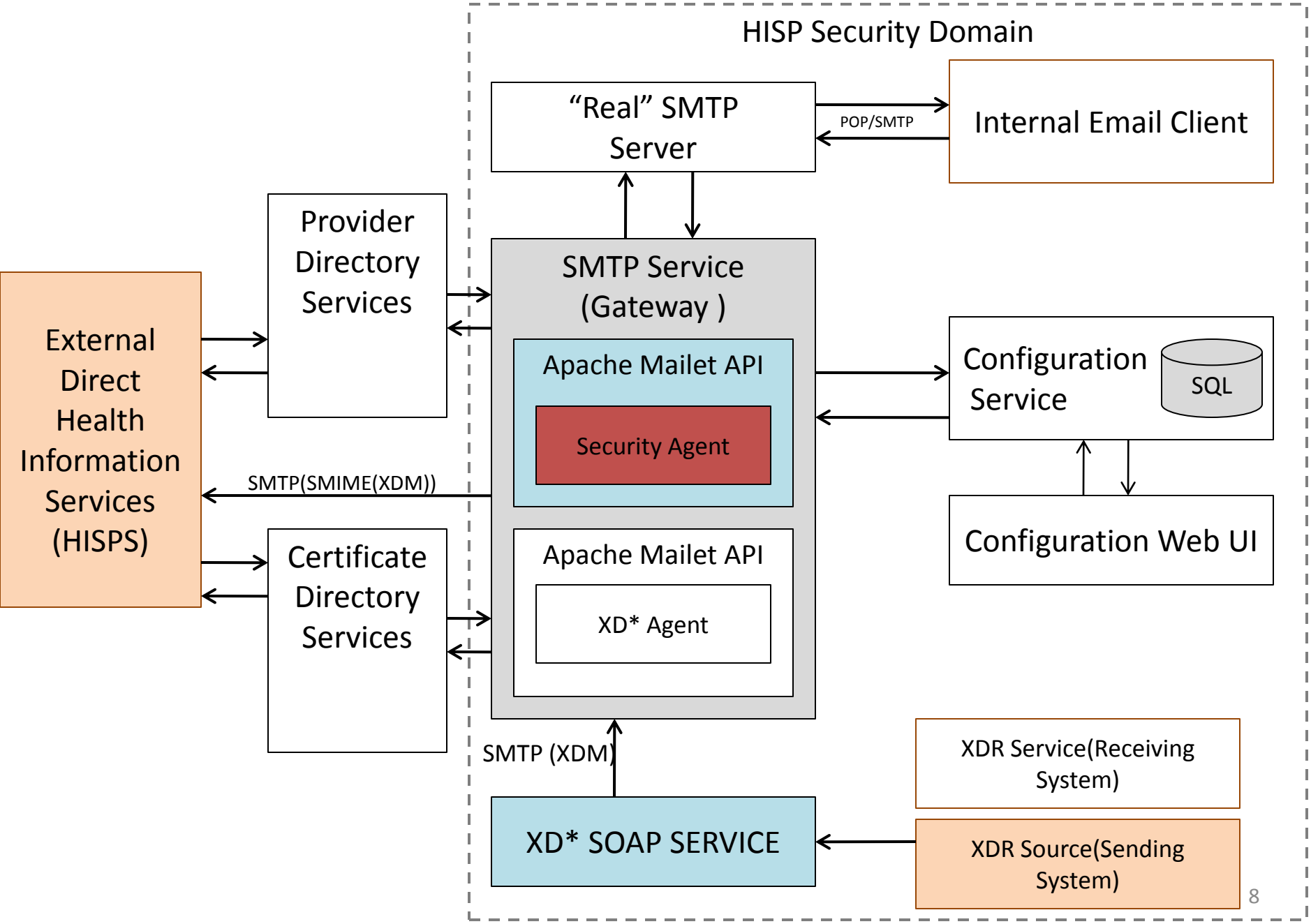


Why the SMTP Backbone ?

- Allows for the inclusion of providers without EHRs in the Direct model
- Allows for a security model that does not rely on a strong federation
  - Strongly federated security with dictated CA structure, like the “Federal Bridge”, seem to be difficult to implement
  - Without strong federation, unanticipated push between two random TLS based SOAP systems is not simple (possible?)
  - Using the Direct “Certificate Directory” model allows for unanticipated SMIME with “dynamic certificate exchange”

GSI Health

THE MAILET, ENABLING SECURE SMTP BASED SERVICES

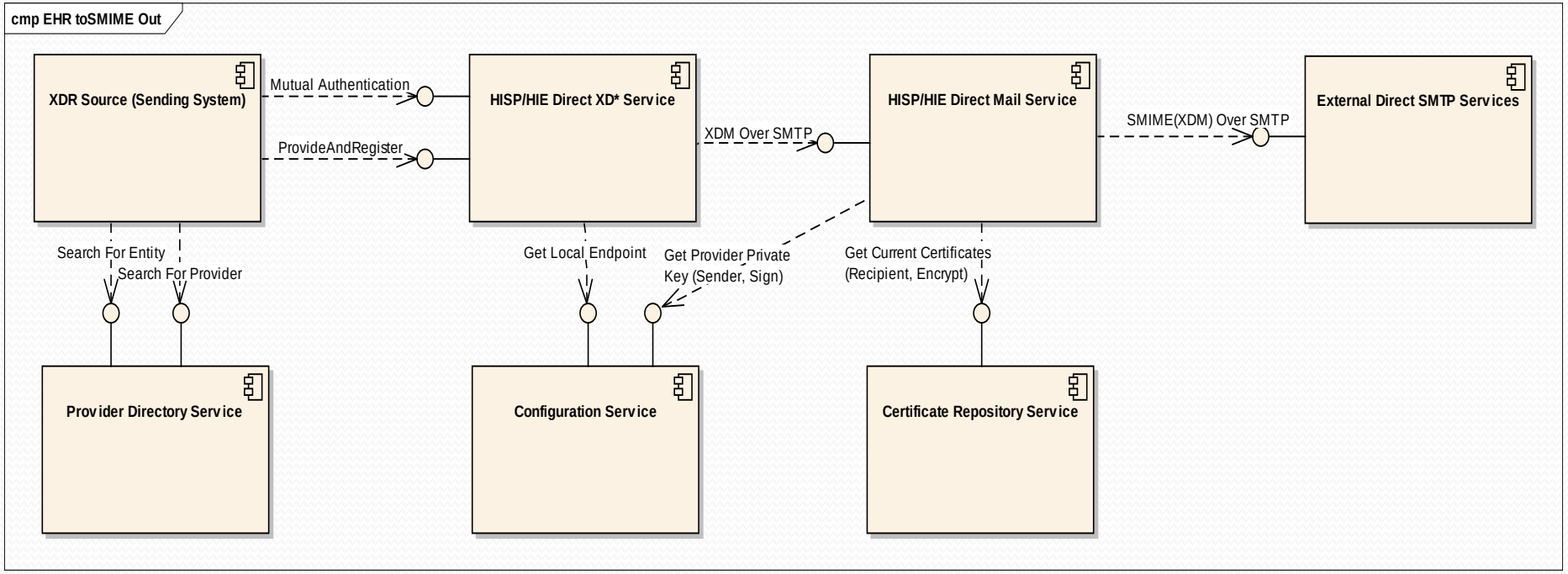


What Apache Mailets Get You

- "In-flow" programmatic access to the (S)MIME message without cumbersome polling or queuing
- Allows for dynamic certificate exchange, decryption and signature validation
- Allows for dynamic conversion to more SOA friendly protocols
- Extremely simple "injection" mechanism
  - Configuration based

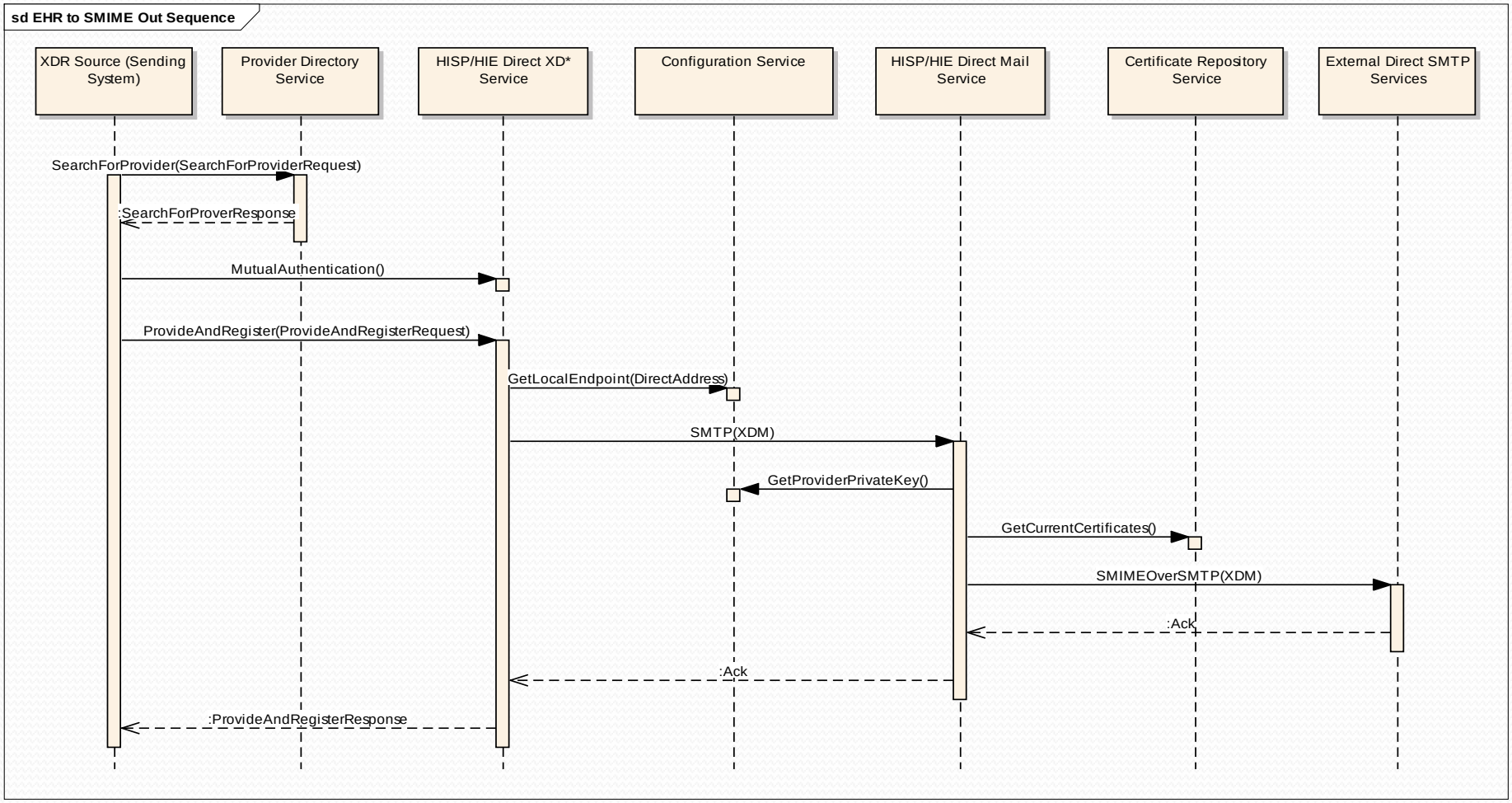
GSI Health

SOAP to SMTP

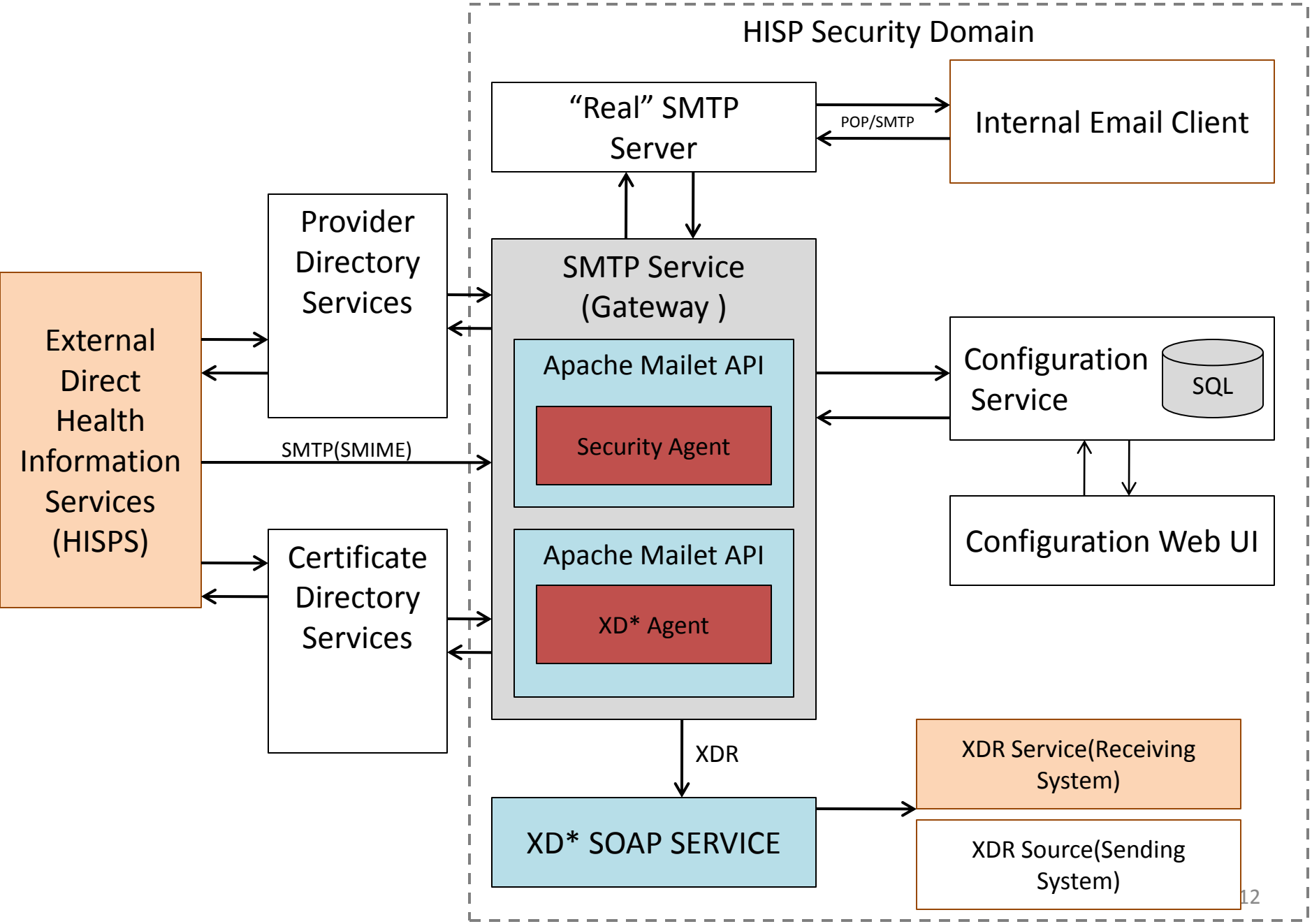


GSI Health

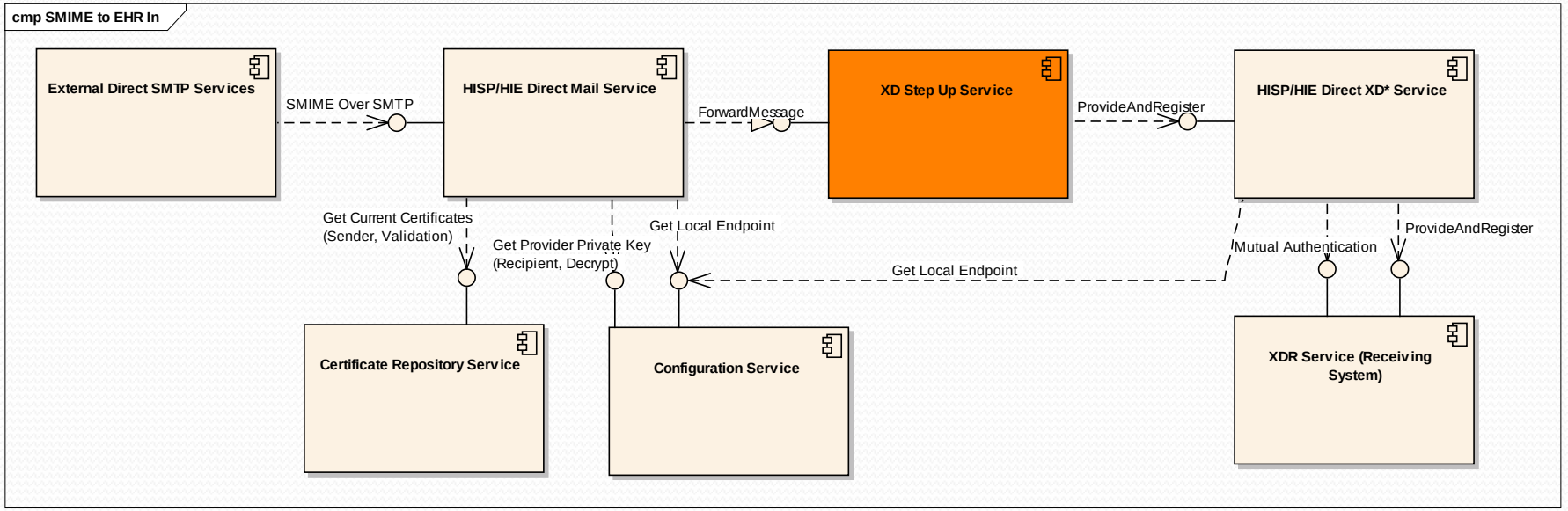
SOAP to SMTP Sequence



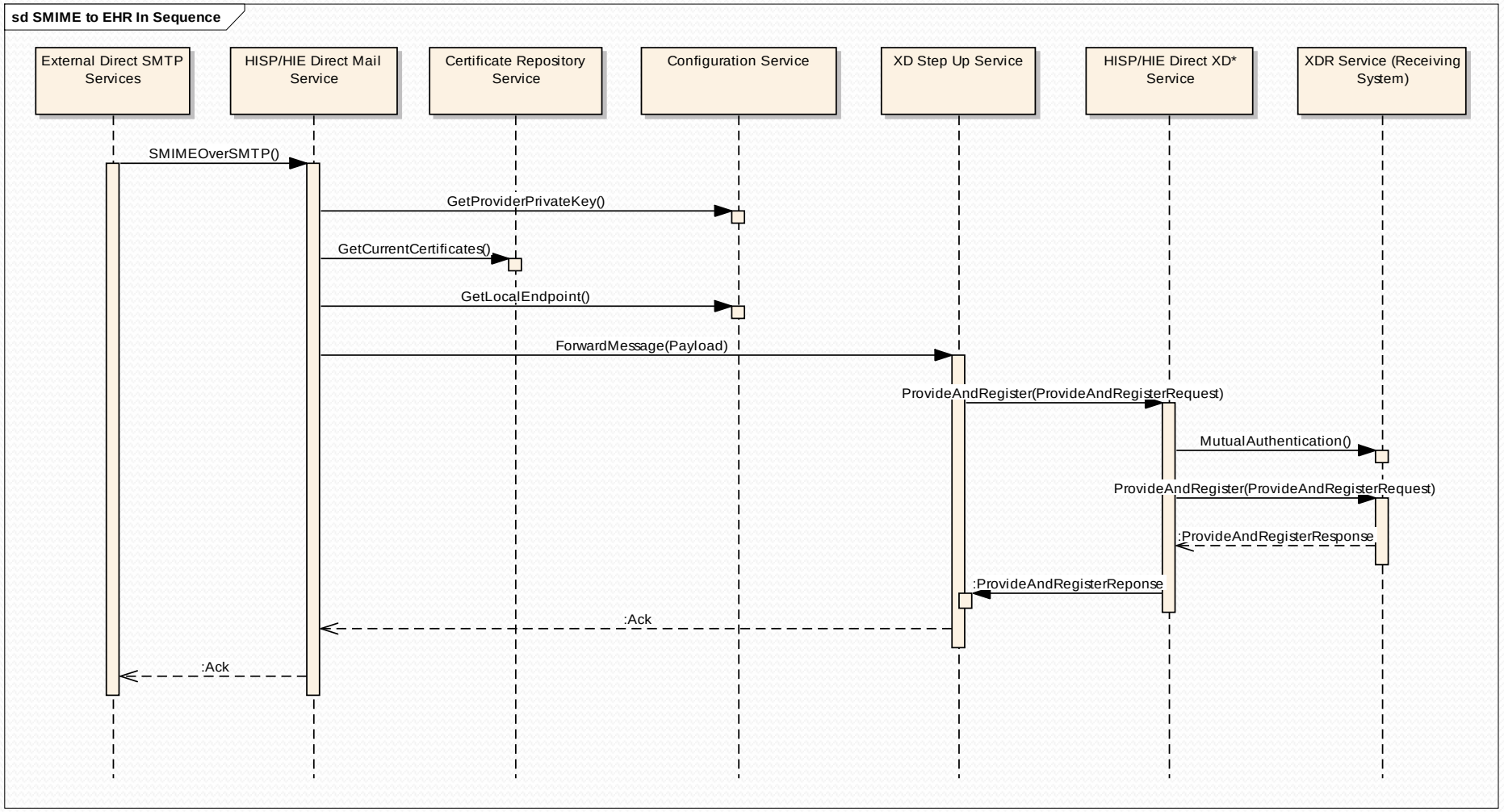
THE MAILET, ENABLING SECURE SMTP BASED SERVICES



SMTP to SOAP


cmp SMIME to EHR In
External Direct SMTP Services
SMIME Over SMTP
HISP/HIE Direct Mail Service
ForwardMessage
XD Step Up Service
ProvideAndRegister
HISP/HIE Direct XD* Service
Get Current Certificates
(Sender, Validation)
Get Provider Private Key
(Recipient, Decrypt)
Get Local Endpoint
Get Local Endpoint
Mutual Authentication
ProvideAndRegister
Certificate Repository Service
Configuration Service
XDR Service (Receiving System)

SMTP to SOAP Sequence



Conclusions and Questions ?

- The Direct specification and reference implementation has been an incredible example of cooperative open source development
- Multiple “connectathons” and extensive jUnit testing help make the implementation rock solid
- Architecture seems as clean as possible with multiple protocols
- Still firming up the Provider Directory detailed requirements
- Certificate Directory now uses DNS, may or may not change

GSI Health