EUROPEAN COMMISSION

DIGIT
Digital Europe Programme

Architecture, Infrastructure, Operations, and Maintenance of an SML Service

Version [1.10]

Status [Final]

Date: 02/02/2024

Table of Contents

1. INTRODUCTION

The Service Metadata Locator (SML) is one of the key components in a message exchange infrastructure that enables dynamic discovery of the participants' messaging capabilities. The purpose of the SML is to discover where participants publish and maintain their messaging capabilities.

DomiSML is the sample sample software provided by the European Commission to implement an eDelivery Service Metadata Locator for an eDelivery party to discover the URLs of other counterparties using eDelivery Access Points and their corresponding metadata. It is based on the eDelivery BDXL profile, an open technical specification for locating Access Points within a network, and on the PEPPOL SML Specification, a technical specification defining a BDXL administration API.

The purpose of the eDelivery building block is to enable the electronic exchange of digital data and documents in an interoperable, secure, reliable, and trusted way for businesses and public administrations. To promote and facilitate the electronic message exchange for various messaging networks of participants (also called business domains), eDelivery offers a number of baseline enabling services, among which the eDelivery SML service. Business domains whose message exchange networks either require a number of participants larger than (currently) 100,000 and/or more than "best effort" SLA are expected to establish and operate their own SML and stop relying on the eDelivery SML service.

1.1. Purpose of the Document

The purpose of this document is to provide needed information for planning, estimating costs, and setting up an SML service on par with the one currently offered by the eDelivery building block, using the DomiSML application. Some of the considerations in this document apply also in case another application is used to establish the SML service, but some of the technical details and cost items may vary to a smaller or larger degree depending on the selected application.

This document is intended for:

- The Directorate Generals and Services of the European Commission, the Member States (MS) as well as private sector organisations interested in hosting SML services using the DomiSML application. In particular:
    - **Administrators** will find it useful for understanding the DomiSML architecture and the existing eDelivery SML service infrastructure (as an example);
    - **Architects** will find it useful for determining how to best exploit the DomiSML application to create a fully-fledged SML solution;
    - **IT Analysts** will find it useful for understanding the communication between the different components of the DomiSML;
    - **Business Analysts** will find it useful for costs and resource planning.

1.2. Document content

The first chapter of the document contains a brief overview of the dynamic discovery process and an explanation of the role of the SML in the dynamic discovery process.

The following chapter provides a short description of implemented services, the application technology stack and its architecture.

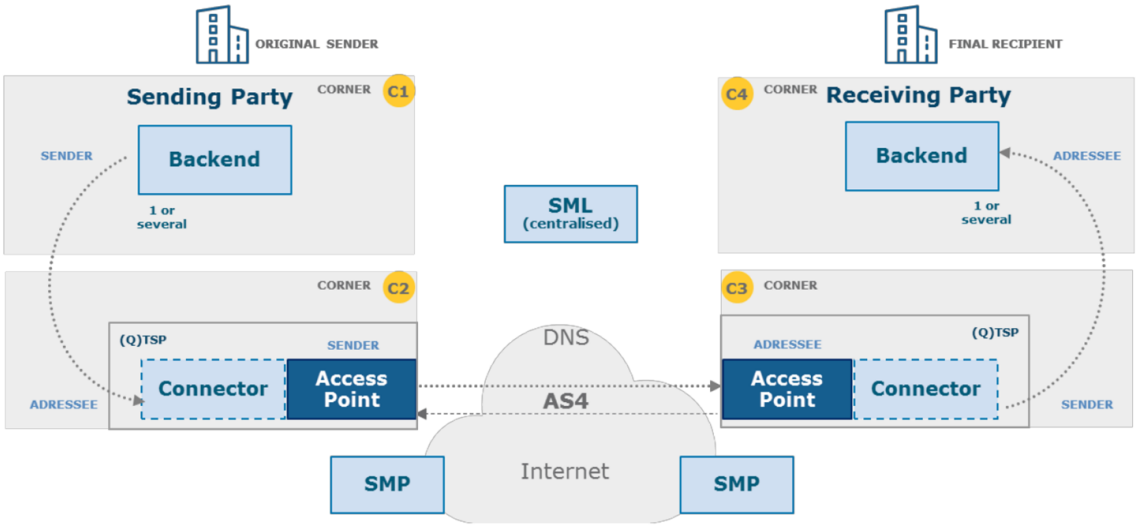
Chapter 4 contains a description of the eDelivery SML service infrastructure and chapter 5 includes security considerations when operating this type of service.

The last chapter describes the support, operational and maintenance tasks required to run a service like the eDelivery SML one.

2. eDELIVERY BUILDING BLOCK

In this document, an eDelivery network is any network using eDelivery standards/profiles and conformant components, irrespective of the components vendors and the network's business purpose. The eDelivery network is a messaging infrastructure working as a collection of distributed nodes conformant to the same technical rules and capable of interacting with each other. eDelivery prescribes technical specifications that can be used in any Policy Domain of the EU (e.g., justice, procurement, consumer protection, etc.) while enabling a secure, reliable and trusted exchange of documents and data (structured, non-structured and/or binary), whether cross-border or cross-sector.

The eDelivery building block uses the de-centralised four-corner model messaging topology, allowing direct communication between different parties without the need to set up proprietary bilateral channels. The parties use their own Backend systems to connect to the eDelivery Access Points for the message exchange, as illustrated in the figure below. The Access Points are interoperable and implement the same message exchange protocol following the same implementation guidelines.



Each participant in a given eDelivery network uses the above model to communicate with any other participant in the same network.

2.1. Dynamic Discovery of participants

To send messages using eDelivery-based messaging infrastructure, the access point of the sending party needs to know the communication information of the receiving party, e.g., service location (lookup address) and communication capabilities. Such information is referred to as 'service metadata' and can be obtained via either a *static* or a *dynamic discovery* process. The static discovery process uses a list of receiving parties and their service metadata stored in the sending access point. The metadata is configured in the sending access point and added to the message.

Dynamic discovery allows the sending access point to query an external service storing up-to-date service metadata about every potential receiving party in the network.

Dynamic discovery in eDelivery is implemented using three components, namely the Service Metadata Publisher (SMP), the Service Metadata Locator (SML) and the DNS server.

This section describes the Service Metadata Publisher (SMP), Service Metadata Locator (SML) and DNS Server components that facilitate and support the dynamic discovery process:

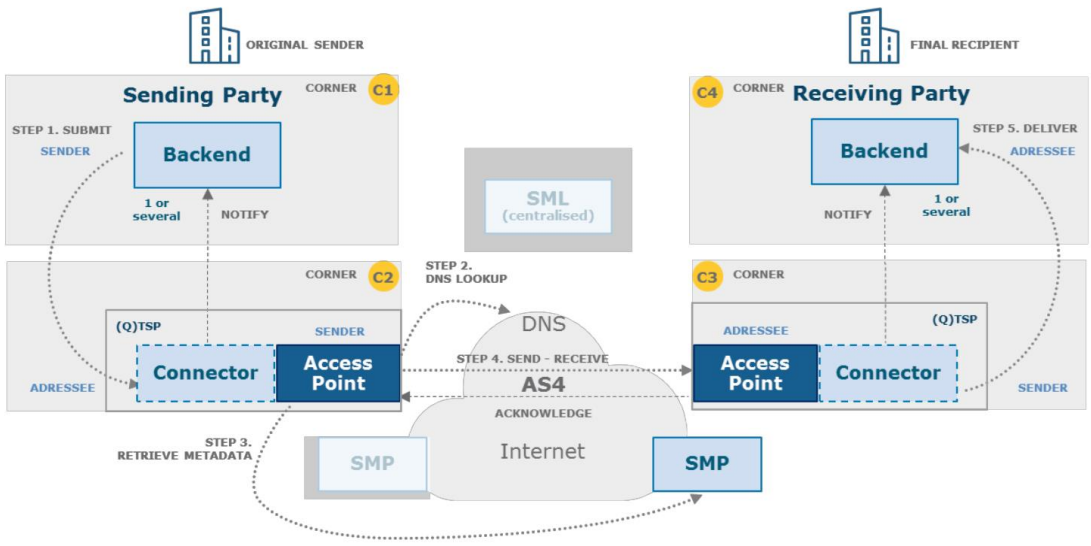
Dynamic discovery involves three types of component actors:

- **Service Metadata Publisher (SMP):** The SMP provides the service location and communication capabilities of receiving parties by storing, exchanging and performing capability lookups for APs. The SMPs operate in a distributed manner in a eDelivery network: there might be one or multiple SMPs in a given eDelivery network. An SMP stores the up-to-date service metadata of every receiving party of the network. Each receiving party publishes its capabilities in one and only one SMP. Hence, for the message exchange, the sending AP first needs to discover the address of the SMP associated to the receiving AP in order to retrieve the required information (i.e., the service metadata) about the receiving AP and receiving party. This information is necessary for the sending APs to send messages.

- **Service Metadata Locator (SML)**: The SML is a centralised component that stores the location of every SMP in the network and manages the resource records of the participants and SMPs in the DNS (Domain Name System) Server. The SML stores the unique identifier of all receiving parties and SMPs on the network in the DNS Server.

- **Domain Name System (DNS) Server**: The DNS Server stores DNS records identified by the unique identifier of each receiving party in the network. Each of these DNS records refers to the lookup address of the corresponding party's SMP. This service enables the sending AP to dynamically locate the SMP holding the service metadata of the receiving party.

The figure below illustrates the Dynamic discovery process:



STEP 1. SUBMIT: The sending party uses the backend system to create a message to be sent to a receiving party. At this stage, the sending party knows the unique identifier of the receiving party (e.g., a company VAT number) and the content of the data or document that it intends to send to the receiving party. After the message (in 'business' format) is created, the backend system submits it to its sending AP.

STEP 2. DNS LOOKUP: Upon submission of a message from the backend system, the sending AP converts the message into the AS4 format. In order to correctly create the AS4 message and route it to the receiving AP, the sending AP builds a canonical representation of the receiving party identifier by hashing it. The sending AP uses this canonical representation to perform a DNS lookup. As a result, the sending AP obtains the URL of the SMP publishing the metadata of the receiving party.

STEP 3. RETRIEVE METADATA: The sending AP then queries the resolved SMP to retrieve the metadata of the receiving party. The metadata includes all the necessary information for the sending AP to send messages to the receiving AP, including URL location and the electronic identity of the AP in the form of a X.509 Certificate.

STEP 4. SEND: The sending AP then builds and sends the AS4 message to the receiving party via the receiving AP according to the URL retrieved from the SMP. In this process, the retrieved AP's X.509 Certificate is used to encrypt payloads for transmission.

STEP 5. DELIVER: The message from the sending AP is translated back into 'business' format and delivered by the receiving AP to the receiving party.

3. DomiSML application

3.1. DomiSML services overview

The DomiSML interface is intended for B2B integration with the Service Metadata Publishers information systems. The purpose is to provide secure, reliable, auditable tools for SMPs to automatically manage participant records in the DNS. The core DomiSML operations are encapsulated in two web service interfaces defined by PEPPOL Transport Infrastructure SML specifications. The two web service interfaces are:

- **ManageServiceMetadataService-1.0.wsdl**: Definition of the services to manage SMPs data.
- **ManageBusinessIdentifierService-1.0.wsdl**: Definition of the service to manage Participant Identifiers.

For administration purposes, two additional web service interfaces have been implemented:

- **BDMSLService-1.0.wsdl**: monitoring and additional tools for managing the Service Metadata Publishers data.
- **BDMSLAdminService-1.0.wsdl**: Opposite to previous interfaces, this web service is not exposed to the internet and is intended to provide tools to the local Administrator for the configuration and administration of the DomiSML application.

Below follows a short description of implemented interface operations:

3.1.1. SMP Management (ManageServiceMetadataService)

Operation	Description
Create	Create a Service Metadata Publisher (SMP) metadata record, containing the metadata about the SMP.
Read	Retrieve the SMP record for the service metadata publisher.
Update	Update the SMP record for the service metadata publisher.
Delete	Delete the SMP record for the service metadata publisher.

3.1.2. Participant Management (ManageBusinessIdentifierService)

Operation	Description
Create CreateList	Create Participant(s) record in DomiSML and DNS.
Delete DeleteList	Delete Participant(s) record from DomiSML and DNS.
List	List (Pages) of Participants registered in SMP.
Migrate PrepareToMigrate	Migrate participant to new SMP.

3.1.3. SMP Management (DomiSML Service)

Operation	Description
ChangeCertificate	This operation allows the SMP owner to change the SMP's authentication certificate.
CreateParticipantIdentifier	This operation has similar function as the Create() operation in the ManageParticipantIdentifier interface with an additional option to define the Service name for NAPTR DNS record.
isAlive	Test all the components of DomiSML: Weblogic nodes, Database connection and DNS connection. The service is intended for internal monitoring system.

3.1.4. DomiSML Administration operations

Operation	Description
ChangeCertificate	This operation allows the Administrator to change the SMP's authentication (expired) certificate.

ClearCache	For performance optimization, DomiSML caches configuration values. This operation clears all cached values.
Manage Configuration (set, read, update)	These operations are used for DomiSML configuration management such as: proxy settings, keystore management, DNS integration data, SMP DNS authentication regular expression for Certificate DN validation, etc. (all the property changes are audited).
Manage SubDomains	These operations enable the Administrator to add, remove, update and list DomiSML domains.
Manage DomainCertificates	These operations enable the Administrator to manage domain authorization certificates.
Manage DNS custom records (A, NAPTR, CNAME)	These operations enable the Administrator to add or remove custom DNS record for types: A, NAPTR and CNAME.
CreateInconsistencyReport	This operation creates Inconsistency data report between Database and DNS server.

3.2. DomiSML technology stack and architecture

The DomiSML architecture follows a multi-tier architecture providing decoupled, flexible, and well-defined components. The multi-tier architecture allows developers to modify or add new functionalities (example: implementation of a new REST API), with minimal need to alter the whole application.

The application has the following layers from top to bottom:

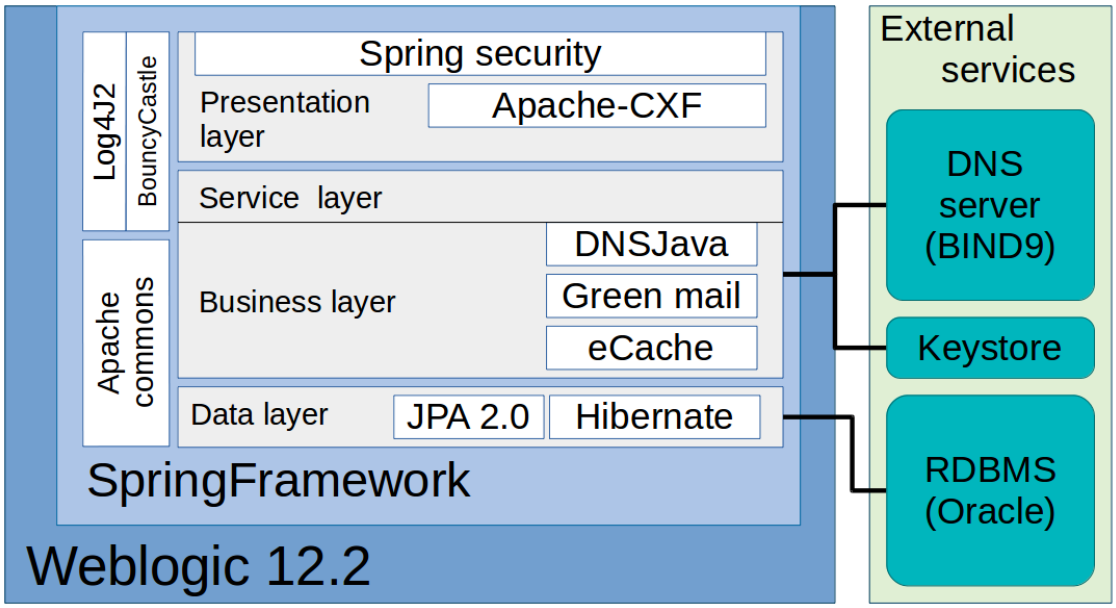
- **Presentation layer**: This is the topmost level of the application. It contains index page, simple DNS domain search tools and WSLD based web services. The main purposes of the layer are: web service SOAP binding (Apache-CXF), security validation (SpringSecurity), request data validation, forwarding request to services layer and forwarding response back to the caller.

- **Service layer:** It contains the complete SML services and it is an integration layer between the presentation layer and business components implemented in the business layer. Data critical services have implemented an authorization protection.

- **Business layer:** This layer contains the implementation of components for executing the SML business processes. Business components handle:

- service metadata and participant management,
- X509Certificate and keystore management using BuncyCastle library,
- DNS record management using DNSJava library for the DNS integration,
- database and DNS consistency verification,
- e-mail creation and sending using Greenmail library,
- application configuration management.

- **Data layer:** This layer handles data persistence to the database. The DomiSML business logic data are stored, read, or updated using standard Java Persistence API (JPA 2.0). The JPA implementation used in DomiSML is Hibernate. The data persistence also contains sufficient data redundancy to enable full recovery of all DNS records in case of sever DNS failure. The layer also audits all saved data changes using the Hibernate-Envers module. The Hibernate functionalities are also used for database scripts generation, which are included in the DomiSML deployment package.

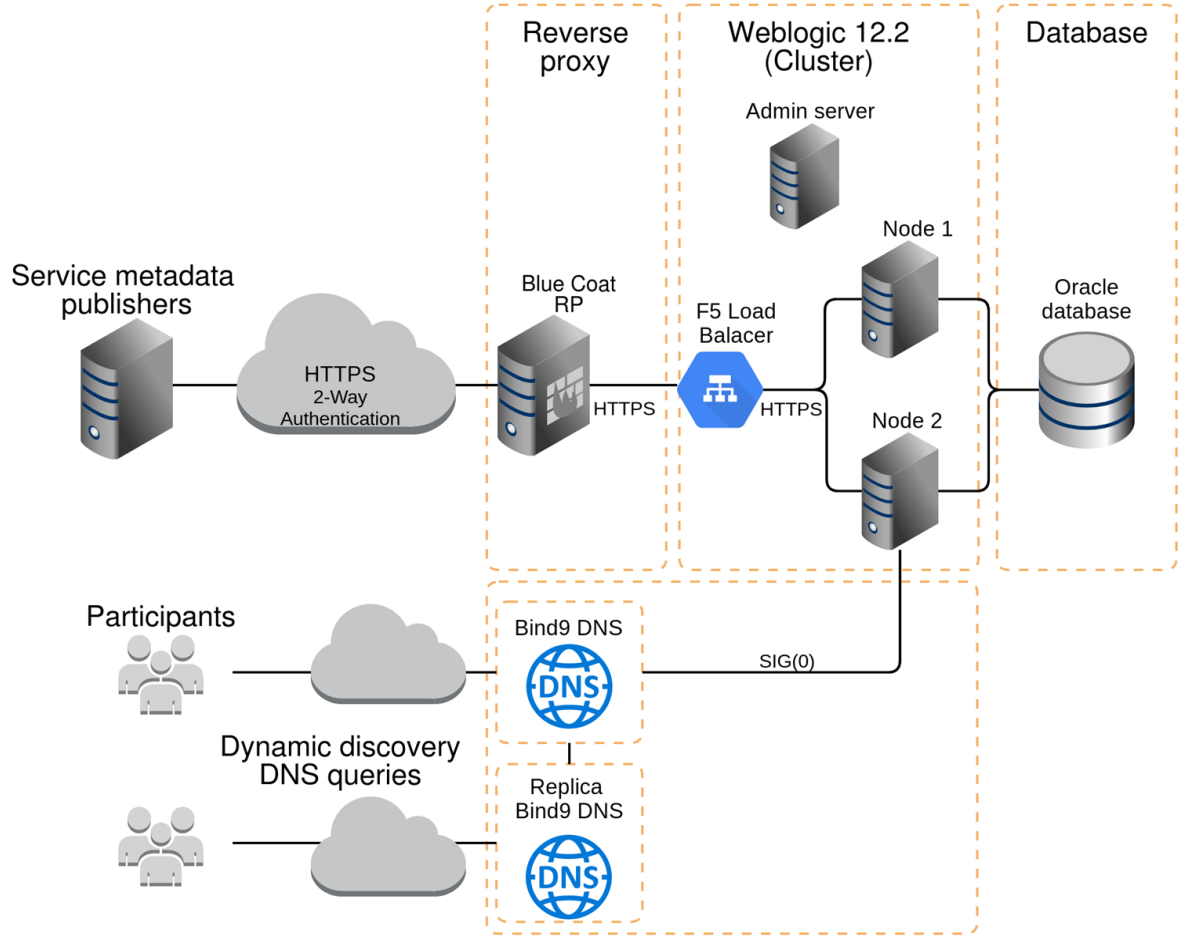


The DomiSML layers are glued with SpringFramework. The calls between the different layers respect the described hierarchy and strictly use the interfaces. The business and data layers are also connecting to external services: DNS server for publishing DNS records, Keystore for secure storing of SML private key, and Oracle database for saving DomiSML business data.

4. eDELIVERY SML INFRASTRUCTURE

The eDelivery SML service must handle a large number of requests and requires high uptime. The high-availability requirement is achieved by providing a component redundancy which allows a certain number of components (separate servers) to experience downtime without interrupting the services provided to the users. The possibility to dynamically adjust the count of redundant components also enables the adjustment of IT resources to various user request load and to economically use the resources.

To ensure high security, all component communication is performed over secure channels, like HTTPS and using DNS SIG(0).



Reverse proxy

A reverse proxy dispatches in-bound network traffic to a set of servers, presenting a single interface to the caller. A reverse proxy is also a security product; it relays network connections for the access to web content located on the EC internal network. The usage of reverse proxies is the best practice in the design of network architectures and is strongly encouraged by the EC security policy. Reverse proxy also terminates the incoming TLS connections and establishes new TLS sessions towards back-end DomiSML application. All caller data are transferred to the backend using various HTTP headers, as X-Forwarded-*, Client-Cert, etc.

Load balancer

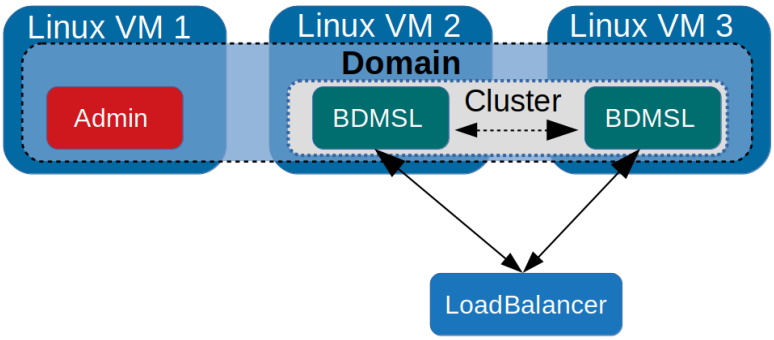
A load balancer distributes traffic across a number of servers and is used to increase capacity (concurrent DomiSML deployments) and reliability of the services.

Weblogic 12.2 Cluster environment

The DomiSML application is deployed on a WebLogic 12.2 cluster, where each of the cluster nodes uses dedicated virtualization of the Linux OS Server. Current deployment contains three virtual servers, one for the WebLogic admin server and two for WebLogic node servers. In case of increased load, additional WebLogic nodes can be added to the cluster. Adding new nodes on dedicated Linux virtualization enables also greater scalability and flexibility when using cloud infrastructure.

Virtualization servers have the following characteristics:

- OS: RHEL release 7.7 (Maipo)
- Processor: Intel(R) Xeon(R) CPU E5-2698 v4 @ 2.20GHz
- CPU: 2
- Physical memory: 8GB
- Weblogic assigned memory: Nodes 4GB and Admin 1GB
- Weblogic domain partition size: 55GB



- Weblogic Temp partition size: 15GB. The DomiSML uses temp folder while generating the inconsistency report. Because of the large DNS zone files, also the temp partition must have sufficient disk space.

DNS Server

The Domain Name System (DNS) is the key component in the eDelivery SML service infrastructure. To ensure high availability, two BIND9 servers in Master-Slave mode, with incremental synchronization (IXFR), are deployed on different locations. Each of the BIND servers are running on virtualized Linux OS with 4 CPUs and 16GB of physical memory. The servers have 50GB disk partition reserved for the DNS zone files. All resources can be expanded based on the number of registered participants.

During the operation, special attention must be given to provide enough physical memory and disk space. For 1Mio DNS records, the BIND server consumes approximatively 5GB disk space and 1,6GB of physical memory. The currently provided disk space and physical memory should be sufficient for handling up to 10Mio DNS records.

The BIND9 DNS server has also enabled the Domain Name System Security Extensions (DNSSEC). The DNSSEC extension enables eDelivery SML service users to validate DNS data and its origin, using the public-key cryptography. The enhanced security also comes with intensive memory consumption. The signed DNS zone file is approximately 10 times bigger than the standard zone (ex.: if a standard zone file is 100M size, then signed zone file has 1GB).

Database

The eDelivery SML service infrastructure uses the Oracle RAC Database. The recommended initial oracle table space size is 5GB. The actual table space consumption depends mostly on the number of registered participants and the number of audit records in the database. For example, exported data without audit tables for 500K participants and 200 SMP service providers takes approximately 250MB.

The audit data size tends to increase quickly and therefore should be regularly monitored and exported from the database to save space.

The number of current DomiSML database connections in production varies between 3 and 6 concurrent connections. The maximum concurrent connections count for both server nodes is set to 15.

5. SECURITY

Secure data exchange between access points starts with the provision of secure SML services. Therefore, the SML service provider must pay close attention to all security aspects when providing the SML services. The SML Services include DomiSML providing web-services for the SMP application integration as well as the DNS lookup services that enable the access point dynamic discovery function.

This chapter starts by describing the authentication and authorization model for the web-services based on the client X.509 Certificate. The next section of the chapter focuses on the handling the client certificate data if DomiSML is behind the Reverse proxy or if the HTTPS session terminates on the application server.

5.1. Authentication and Authorization

When the DomiSML web services are exposed to the internet, the web services must be secured with HTTPS sessions. The DomiSML is using the 2-way/mutual HTTPS (TLS) session, where the client authenticates using its X.509 certificate. The DomiSML has also the option to allow anonymous access for testing and demonstration purposes.

REMARK: *When* DomiSML *services are exposed to the internet, the administrators must make sure that the configuration property* ***unsecureLoginAllowed flag*** *is set to* ***false****.*

During a service call, the DomiSML verifies the validity and trust of the SMP's TLS certificate and authenticates the SMP. After successful authentication, the SMP is authorized to access a specific business domain to manage the SMP entry and its participant identifiers. The DomiSML application can perform 2 different types of authorizations: One is the particular Domain **certificate-issuer-**based method that would automatically authorize all the **certificate-issuer** trusted certificates. The other is the **certificate-based authorization method to** authorize an individual SMP X.509 certificate.

5.1.1. Certificate-issuer-based authorization

The authorization is suitable for business domains with a high number of SMP service providers. The SML domain owner must provide a dedicated issuer certificate for issuing all certificates for the SMP service providers in a particular domain. When the business owner issues a certificate to the SMP service provider, it automatically authorizes the certificate to access the DomiSML business domain.

DomiSML introduces the ability to define regular expressions on the SMP X509 Certificate's subject DN to filter the SMP authorization to specific certificates issued by the registered domain issuer. The functionality also enables business owners to use certificates issued for other purposes, as the case for also issuing the AP certificates. The regular expression is defined by the DomiSML configuration property, **authorization.smp.certSubjectRegex**.

Below is an example of a regular expression where the only SMP certificates allowed have subject CN starting with "SMP" or the subject DN containing organization unit (OU) with value: "PRODUCTION SMP:

Regular expression: ***^.*(CN=SMP_|OU=PRODUCTION SMP).*$***

REMARK: The system administrator must register and authorize the issuer certificate in the DomiSML and associate it to the domain.

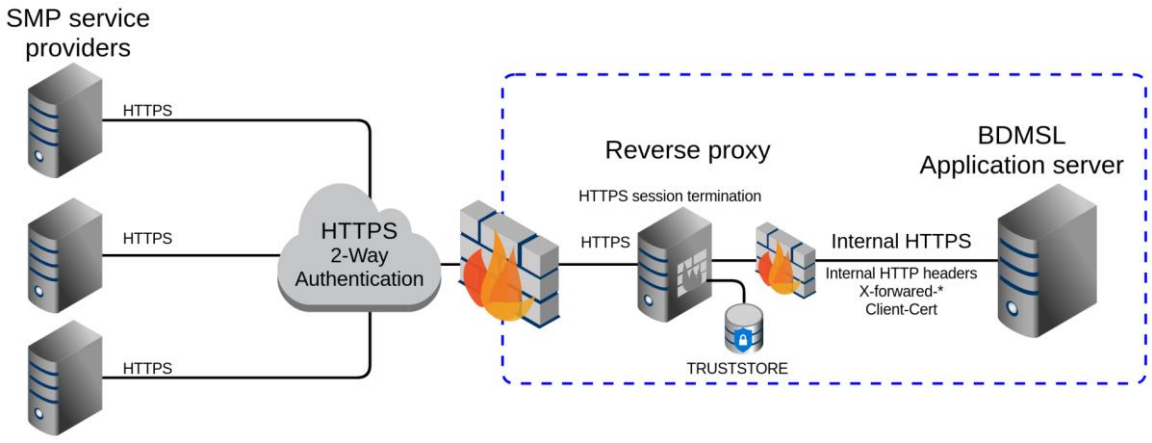
5.1.2. Certificate-based authorization

This authorization is suitable for business domains with a lower number of SMP service providers and where maintaining a dedicated certificate issuer for the domain SMP certificates is not an option. In this case, each SMP certificate must be added and authorized in the DomiSML business domain by the administrator.

5.2. DomiSML behind the reverse proxy with HTTPS termination

It is strongly advised to set up DomiSML behind a reverse proxy with a TLS termination (or SSL termination, or SSL offloading). In this environment configuration, the reverse proxy server handles incoming TLS connections.

In order to establish an HTTPS session with 2-way / mutual authentication (2-Way HTTPS), the truststore must be configured at the reverse proxy server. The reverser proxy server ensures that all unwanted TLS connections attempts are blocked at the entry point and are not reaching the backed application server.



During a successfully established 2-Way HTTPS session, the reverse proxy forwards request to the DomiSML application. The DomiSML identifies and authorizes the requestor using the client certificate metadata that is added to the HTTP header field: **Client-Cert**.

The **Client-Cert** header field contains the following URL encoded certificate values separated with the '&' character:

- sno: certificate serial number,
- subject: certificate subject DN,
- issuer: certificate issuer DN,
- validfrom: certificate valid from,
- validto: certificate valid to.

In order for the BDMLS to accept and use the Client-Cert header, the DomiSML property: **authentication.bluecoat.enabled** must be set to **true**.

REMARK: *Set this property to true ONLY when* DomiSML *is behind the reverse proxy and make sure Client-Cert header cannot be forwarded to* DomiSML *application by the external user.*

Below is an example of the Client-Cert HTTP header:

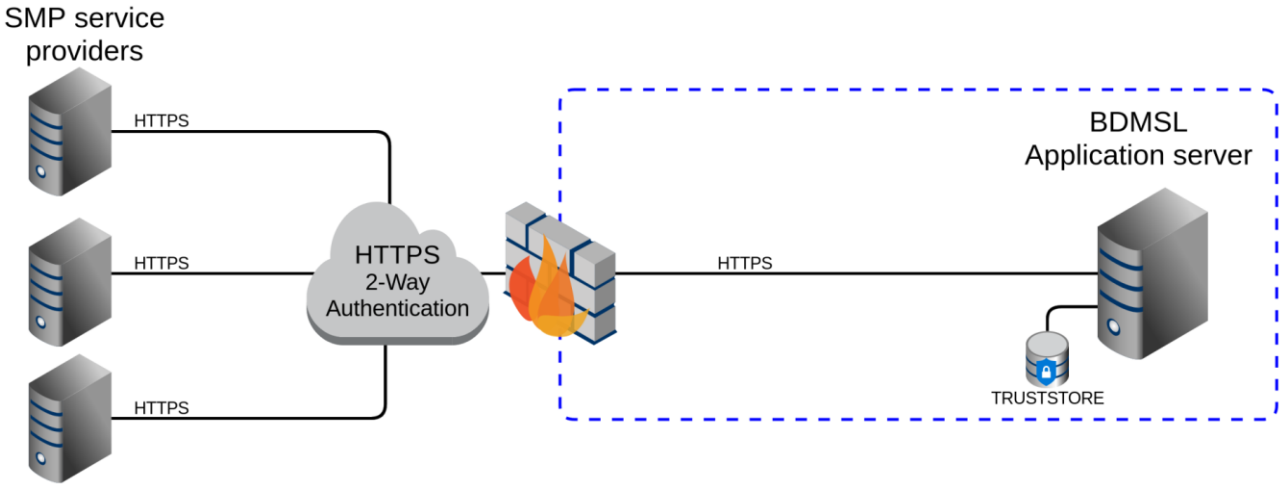
sno=234503&subject=CN=SMP_TEST-PRE-SET-EXAMPLE, OU=eDelivery, O=DIGITAL, C=BE&validfrom=Dec 6 17:41:42 2016 GMT&validto=Jul 9 23:59:00 2050 GMT&issuer=CN=rootCNTest,OU=B4,O=DIGIT,L=Brussels,ST=BE,C=BE

5.3. DomiSML application server HTTPS configuration

DomiSML can also be hosted in an application server that handles the 2-way TLS authentication.

The server accepts and validates incoming 2-Way TLS Client certificates according to the server configuration and the Truststore located at the application server (The HTTPS configurations is application server specific and is not described in this document).

When the application server establishes the HTTPS session, the client certificate(s) is/are passed to DomiSML in HTTP header field javax.servlet.request.X509Certificate. The DomiSML extracts the required data from the certificate and handles the client authentication and authorization the same way as for **Client-Cert** header case.



5.4. Certificate security policy

Once a certificate is registered in the domain, the following values are extracted from the certificate Subject DN and stored in the DomiSML certificate domain table: Common name (CN), organization (O), country (C) and the URL for the certificate revocation list (CRL).

Although the certificate must be validated at the HTTPS session, the DomiSML also validates the certificate using the "valid from" and "valid to" values and the registered CRL URL.

Because the domain authorization is based on the DN values including CN, O, and C, it is recommended to establish a security policy that allows the unique identification of the SMP service provider using these 3 fields.

6. Support, Operations and Maintenance

The eDelivery SML service support follows ITIL's 3-level Incident Management process. The first two lines of the support are organized by the eDelivery Support Office. The third line support is provided by the EC infrastructure service providers and eDelivery Technical Office. The former is responsible for the hosted services as the network, WebLogic domain, DNS servers, and Oracle database. The latter is responsible for the development and maintenance of the DomiSML software.

To ensure the quality of the eDelivery SML service, the strict separation of software development and IT operations is in place. Since the eDelivery Technical Office is responsible for the development and maintenance of the DomiSML application, the eDelivery Support Office (Second-Line Support) took over the IT operations.

First-Line Support

When users contact the Service Desk, the Service Desk collects as much information and diagnostics about the incident as possible, and even resolves the issue or provides explanation on the spot, if possible. This reduces resolution time for all minor incidents and common questions, which results in increased end-user satisfaction. The first line of support also notifies users of upcoming IT operation events and planned downtimes of the eDelivery SML services. Their responsibility is also to prepare monthly reports on service availability and user satisfaction.

The Second-Line Support

This level of support is handled by dedicated personnel with in-depth technical skills on the DomiSML application and eDelivery SML service infrastructure. The responsibilities are to register, investigate, escalate if needed, resolve, and close tickets. It also acts as the technical single point of contact to the Service Metadata Publishers and Domain Owners and provides support during establishing the connection to the eDelivery SML service.

The operational IT tasks performed by the Second-Level support also include the deployment of a new version of DomiSML, coordinating maintenance tasks such as renewing SML certificates, coordinating infrastructure changes, managing service availability monitoring, and administering the DomiSML.

Third-Line Support

As mentioned, this level of support is provided by the eDelivery Technical Office and the EC infrastructure service providers. The eDelivery Technical Office is responsible for resolving the ticket transferred by the Second-Line Support, answering demanding technical questions and enabling the necessary transfer of knowledge to the Support Office, updating DomiSML with dependent library security patches, adapting and testing DomiSML to updated EC infrastructure services, continuous performance improvements, investigation and resolving the complex IT Operational incidents. When a DomiSML version is ready for deployment, the responsibility of the Technical Office is to create a runbook with required detailed steps for executing the deployment. The deployment itself is then executed by the eDelivery Support Office.

EC infrastructure service providers are responsible for infrastructure maintenance as applying security patches to Linux OS, Oracle database, WebLogic application, DNS server, and all other components used by eDelivery SML services. Because of the advance use of the BIND9 servers, they tightly collaborate with the eDelivery Technical Office and the Support team on all DNS related issues/tasks.

6.1. Human resources

The following section describes the knowledge required and the human resource consumption for operating an SML service in a similar environment as the eDelivery SML service. The data provided are anecdotal and as such should be used only for orientation purposes. Resource consumption is estimated based on the experience of operating the eDelivery SML service with stable production.

The First-Line Support requires technical support background, strong attention to detail, high professional attitude standards, as well as English speaking and writing skills. Expected tasks are:

- Report generation: 1 person-day per month
- Preparing the IT operation event notifications for the user: 1/2 person-day per month
- User helpdesk interaction: 2 person-days per month.

The Second-Line Support and **IT operations** require good skills in analysing incidents and problems and should have intermediate or enhanced technical knowledge on WSDL-based web services, SQL queries, networking, java applications, and X509 certificates. Staff should also be familiar with products such as BIND9 Server, WebLogic 12.2 server, Oracle database, be able to use a tool like Oracle SQL Developer (or similar), SoapUI, and Portecle or KeyStore Explorer for X509 certificate handling. Expected tasks are:

- User helpdesk interaction: 5 person-days per month,
- Solving, reporting, and escalating if needed all technical issues: 5 person-days per month
- Updating expired certificates for ServiceMetadata publishers: 5 person-days per month
- Administering the DomiSML: 2 person-days per year
- Deployment of the new DomiSML versions: 2 person-days per year.

The Third-Line Support can be external suppliers, vendors, or in-house employees with extensive knowledge on Weblogic 12.2 and Java EE and Spring applications, Oracle Database, BIND9 DNS server. In case of new DomiSML feature requests, Java programming skills are also required, with good knowledge of the following technologies and libraries: Spring Framework, apache-cxf, Hibernate, BouncyCastle, maven, Log4J2 and DNSJava.

The main tasks executed by the Third-Line Support are:

- SML incident investigation forwarded by the Second-Line Support: 10 person-days per year
- DomiSML applications library updates: 5 person-days per year
- DomiSML feature and bugfix development: an estimate of the consumed time cannot be given because it is highly task dependent.

Consumption of person-days is not estimated for the following tasks because they are provided by EC infrastructure service providers. They should however also be considered by the future SML service providers:

- Linux OS security upgrades
- WebLogic server maintenance and Weblogic Security upgrades
- Oracle database maintenance
- Network maintenance (upgrading and security patching of ReverseProxy, load balancer, firewalls, network security scanning).

7. CONTACT INFORMATION

eDelivery Support Team

By email: EC-EDELIVERY-SUPPORT@ec.europa.eu

Support Service: 8am to 6pm (Normal EC working Days)