Object Management Group

140 Kendrick Street
Building A Suite 300
Needham, MA 02494
USA

Telephone: +1-781-444-0404
Facsimile: +1-781-444-0320

Request For Proposal

DDS Security

OMG Document: mars/2010-12-37

Letters of Intent due: April 30, 2011
Submissions due: November 7, 2011

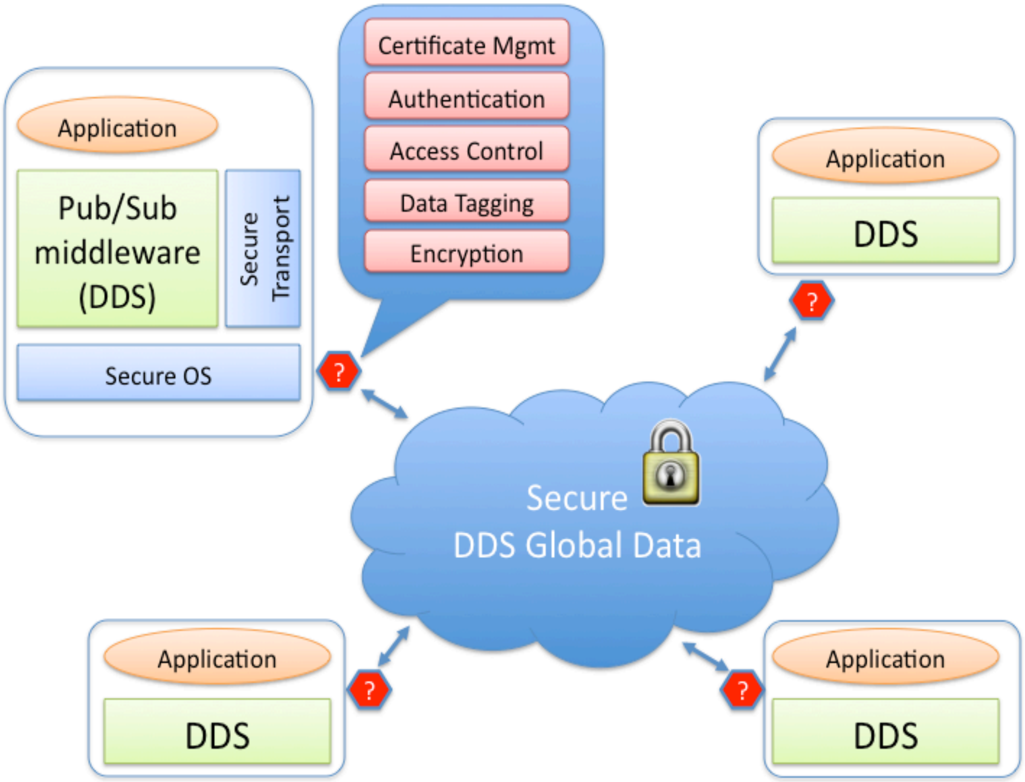
Objective of this RFP

The *Data Distribution Service for Real-Time Systems* (DDS) is the Object Management Group (OMG) standard for data-centric publish subscribe. The objective of this RFP is to define the set of Extensions to DDS required to provide Information Assurance for systems built using DDS.

The OMG DDS standard has experienced a record-breaking adoption within the Aerospace and Defense domains, and is swiftly expanding into new domains such as Transportation, Financial Services, Telecommunications, and SCADA. Many of these systems control critical functions or transmit sensitive information, which must be protected.

The OMG DDS standard defines local interfaces that can be used by an application to access a virtual “Global Data Space” where applications can publish and subscribe data. To provide Information Assurance application developers must either secure the network and computers where DDS runs, or rely on vendor-specific extensions, which are neither portable nor interoperable. The objective of this RFP is to remediate that by defining:

- A standard set of security interception points within the operation of DDS middleware implementations and corresponding security-interceptor SPI contracts at each interception point used to enable configuration and portability of Secure DDS applications.
- A set of pre-defined implementations of those security-interceptor SPIs contracts (a.k.a security plugins SPIs), implementing a standard security model for DDS so that the security policies can be defined and deployed in a portable manner across DDS vendors
- Any extensions to the DDS Interoperability Wire Protocol needed to enable interoperability of Secure DDS applications.



For further details see Chapter 6 of this document.

1.0 Introduction

1.1 Goals of OMG

The Object Management Group (OMG) is the world's largest software consortium with an international membership of vendors, developers, and end users. Established in 1989, its mission is to help computer users solve enterprise integration problems by supplying open, vendor-neutral portability, interoperability and reusability specifications based on Model Driven Architecture (MDA). MDA defines an approach to IT system specification that separates the specification of system functionality from the specification of the implementation of that functionality on a specific technology platform, and provides a set of guidelines for structuring specifications expressed as models. OMG has established numerous widely used standards such as OMG IDL[IDL], CORBA[CORBA], Realtime CORBA [CORBA], GIOP/IIOP[CORBA], UML[UML], MOF[MOF], XMI[XMI] and CWM[CWM] to name a few significant ones.

1.2 Organization of this document

The remainder of this document is organized as follows:

Chapter 2 - *Architectural Context* - background information on OMG's Model Driven Architecture.

Chapter 3 - *Adoption Process* - background information on the OMG specification adoption process.

Chapter 4 - *Instructions for Submitters* - explanation of how to make a submission to this RFP.

Chapter 5 - *General Requirements on Proposals* - requirements and evaluation criteria that apply to all proposals submitted to OMG.

Chapter 6 - *Specific Requirements on Proposals* - problem statement, scope of proposals sought, requirements and optional features, issues to be discussed, evaluation criteria, and timetable that apply specifically to this RFP.

Appendix A – *References and Glossary Specific to this RFP*

Appendix B – *General References and Glossary*

1.3 Conventions

The key words "**must**", "**must not**", "**required**", "**shall**", "**shall not**", "**should**", "**should not**", "**recommended**", "**may**", and "**optional**" in this document are to be interpreted as described in RFC 2119 [RFC2119].

1.4 Contact Information

Questions related to the OMG's technology adoption process may be directed to *omg-process@omg.org*. General questions about this RFP may be sent to *responses@omg.org*.

OMG documents (and information about the OMG in general) can be obtained from the OMG's web site (*http://www.omg.org/*). OMG documents may also be obtained by contacting OMG at *documents@omg.org*. Templates for RFPs (like this document) and other standard OMG documents can be found at the OMG Template Downloads Page at *http://www.omg.org/technology/template_download.htm*

2.0 Architectural Context

MDA provides a set of guidelines for structuring specifications expressed as models and the mappings between those models. The MDA initiative and the standards that support it allow the same model specifying business system or application functionality and behavior to be realized on multiple platforms. MDA enables different applications to be integrated by explicitly relating their models; this facilitates integration and interoperability and supports system evolution (deployment choices) as platform technologies change. The three primary goals of MDA are portability, interoperability and reusability.

Portability of any subsystem is relative to the subsystems on which it depends. The collection of subsystems that a given subsystem depends upon is often loosely called the *platform,* which supports that subsystem. Portability – and reusability - of such a subsystem is enabled if all the subsystems that it depends upon use standardized interfaces (APIs) and usage patterns.

MDA provides a pattern comprising a portable subsystem that is able to use any one of multiple specific implementations of a platform. This pattern is repeatedly usable in the specification of systems. The five important concepts related to this pattern are:

1. *Model* – A model is a representation of a part of the function, structure and/or behavior of an application or system. A representation is said to be formal when it is based on a language that has a well-defined form ("syntax"), meaning ("semantics"), and possibly rules of analysis, inference, or proof for its constructs. The syntax may be graphical or textual. The semantics might be defined, more or less formally, in terms of things observed in the world being described (e.g. message sends and replies, object states and state changes, etc.), or by translating higher-level language constructs into other constructs that have a well-defined meaning. The optional rules of inference define what unstated properties you can deduce from the explicit statements in the model. In MDA, a representation that is not formal in this sense is not a model. Thus, a diagram with boxes and lines and arrows that is not supported by a definition of the meaning of a box, and the meaning of a line and of an arrow is not a model—it is just an informal diagram.

2. *Platform* – A set of subsystems/technologies that provide a coherent set of functionality through interfaces and specified usage patterns that any subsystem that depends on the platform can use without concern for the details of how the functionality provided by the platform is implemented.

3. *Platform Independent Model (PIM)* – A model of a subsystem that contains no information specific to the platform, or the technology that is used to realize it.

4. *Platform Specific Model (PSM)* – A model of a subsystem that includes information about the specific technology that is used in the realization of that subsystem on a specific platform, and hence possibly contains elements that are specific to the platform.

5. *Mapping* – Specification of a mechanism for transforming the elements of a model conforming to a particular metamodel into elements of another model that conforms to another (possibly the same) metamodel. A mapping may be expressed as associations, constraints, rules, templates with parameters that must be assigned during the mapping, or other forms yet to be determined.

For example, in case of CORBA the platform is specified by a set of interfaces and usage patterns that constitute the CORBA Core Specification [CORBA]. The CORBA platform is independent of operating systems and programming languages. The OMG Trading Object Service specification [TOS] (consisting of interface specifications in OMG Interface Definition Language (OMG IDL)) can be considered to be a PIM from the viewpoint of CORBA, because it is independent of operating systems and programming languages. When the IDL to C++ Language Mapping specification is applied to the Trading Service PIM, the C++-specific result can be considered to be a PSM for the Trading Service, where the platform is the C++ language and the C++ ORB implementation. Thus the IDL to C++ Language Mapping specification [IDLC++] determines the mapping from the Trading Service PIM to the Trading Service PSM.

Note that the Trading Service model expressed in IDL is a PSM relative to the CORBA platform too. This highlights the fact that platform-independence and platform-specificity are relative concepts.

The UML Profile for EDOC specification [EDOC] is another example of the application of various aspects of MDA. It defines a set of modeling constructs that are independent of middleware platforms such as EJB [EJB], CCM [CCM], MQSeries [MQS], etc. A PIM based on the EDOC profile uses the middleware-independent constructs defined by the profile and thus is middleware-independent. In addition, the specification defines formal metamodels for some specific middleware platforms such as EJB, supplementing the already-existing OMG metamodel of CCM (CORBA Component Model). The specification also defines mappings from the EDOC profile to the middleware metamodels. For example, it defines a mapping from the EDOC profile to EJB. The mapping specifications facilitate the transformation of any EDOC-based PIM into a corresponding PSM for any of the specific platforms for which a mapping is specified.

Continuing with this example, one of the PSMs corresponding to the EDOC PIM could be for the CORBA platform. This PSM then potentially constitutes a PIM, corresponding to which there would be implementation language specific PSMs derived via the CORBA language mappings, thus illustrating recursive use of the Platform-PIM-PSM-Mapping pattern.

Note that the EDOC profile can also be considered to be a platform in its own right. Thus, a model expressed via the profile is a PSM relative to the EDOC platform.

An analogous set of concepts apply to Interoperability Protocols wherein there is a PIM of the payload data and a PIM of the interactions that cause the data to find its way from one place to another. These then are realized in specific ways for specific platforms in the corresponding PSMs.

Analogously, in case of databases there could be a PIM of the data (say using the Relational Data Model), and corresponding PSMs specifying how the data is actually represented on a storage medium based on some particular data storage paradigm etc., and a mapping from the PIM to each PSM.

OMG adopts standard specifications of models that exploit the MDA pattern to facilitate portability, interoperability and reusability, either through ab initio development of standards or by reference to existing standards. Some examples of OMG adopted specifications are:

1. *Languages* – e.g. IDL for interface specification, UML for model specification, OCL for constraint specification, etc.

2. Mappings – e.g. Mapping of OMG IDL to specific implementation languages (CORBA PIM to Implementation Language PSMs), UML Profile for EDOC (PIM) to CCM (CORBA PSM) and EJB (Java PSM), CORBA (PSM) to COM (PSM) etc.

3. *Services* – e.g. Naming Service [NS], Transaction Service [OTS], Security Service [SEC], Trading Object Service [TOS] etc.

4. *Platforms* – e.g. CORBA [CORBA].

5. *Protocols* – e.g. GIOP/IIOP [CORBA] (both structure and exchange protocol), XML Metadata Interchange [XMI] (structure specification usable as payload on multiple exchange protocols).

6. *Domain Specific Standards* – e.g. Data Acquisition from Industrial Systems (Manufacturing) [DAIS], General Ledger Specification (Finance) [GLS], Air Traffic Control (Transportation) [ATC], Gene Expression (Life Science Research) [GE], Personal Identification Service (Healthcare) [PIDS], etc.

For an introduction to MDA, see [MDAa]. For a discourse on the details of MDA please refer to [MDAc]. To see an example of the application of MDA see [MDAb]. For general information on MDA, see [MDAd].

Object Management Architecture (OMA) is a distributed object computing platform architecture within MDA that is related to ISO’s Reference Model of Open Distributed Processing RM-ODP[RM-ODP]. CORBA and any extensions to it are based on OMA. For information on OMA see [OMA].

3.0 Adoption Process

3.1 Introduction

OMG adopts specifications by explicit vote on a technology-by-technology basis. The specifications selected each satisfy the architectural vision of MDA. OMG bases its decisions on both business and technical considerations. Once a specification adoption is finalized by OMG, it is made available for use by both OMG members and non-members alike.

Request for Proposals (RFP) are issued by a *Technology Committee* (TC), typically upon the recommendation of a *Task Force* (TF) and duly endorsed by the *Architecture Board* (AB).

Submissions to RFPs are evaluated by the TF that initiated the RFP. Selected specifications are *recommended* to the parent TC after being *reviewed* for technical merit and consistency with MDA and other adopted specifications and *endorsed* by the AB. The parent TC of the initiating TF then votes to *recommend adoption* to the OMG Board of Directors (BoD). The BoD acts on the recommendation to complete the adoption process.

For more detailed information on the adoption process see the *Policies and Procedures of the OMG Technical Process* [P&P] and the *OMG Hitchhiker’s Guide* [Guide]. In case of any inconsistency between this document and the [P&P] in all cases the [P&P] shall prevail.

3.2 Steps in the Adoption Process

A TF, its parent TC, the AB and the Board of Directors participate in a collaborative process, which typically takes the following form:

- *Development and Issuance of RFP*

  RFPs are drafted by one or more OMG members who are interested in the adoption of a standard in some specific area. The draft RFP is presented to an appropriate TF, based on its subject area, for approval and recommendation to issue. The TF and the AB provide guidance to the drafters of the RFP. When the TF and the AB are satisfied that the RFP is appropriate and ready for issuance, the TF recommends issuance to its parent TC, and the AB endorses the recommendation. The TC then acts on the recommendation and issues the RFP.

- *Letter of Intent (LOI)*

  A Letter of Intent (LOI) must be submitted to the OMG signed by an officer of the member organization which intends to respond to the RFP, confirming the organization’s willingness to comply with OMG’s terms and conditions, and commercial availability requirements. (See section 4.3 for more information.). In order to respond to an RFP the organization must be a member of the TC that issued the RFP.

- *Voter Registration*

  Interested OMG members, other than Trial, Press and Analyst members, may participate in specification selection votes in the TF for an RFP. They may need to register to do so, if so stated in the RFP. Registration ends on a specified date, 6 or more weeks after the announcement of the registration period. The registration closure date is typically around the time of initial submissions. Member organizations that have submitted an LOI are automatically registered to vote.

- *Initial Submissions*

  Initial Submissions are due by a specified deadline. Submitters normally present their proposals at the first meeting of the TF after the deadline. Initial Submissions are expected to be complete enough to provide insight on the technical directions and content of the proposals.

- *Revision Phase*

  During this time submitters have the opportunity to revise their Submissions, if they so choose.

- *Revised Submissions*

Revised Submissions are due by a specified deadline. Submitters again normally present their proposals at the next meeting of the TF after the deadline. (Note that there may be more than one Revised Submission deadline. The decision to set new Revised Submission deadlines is made by the registered voters for that RFP.)

- *Selection Votes*

  When the registered voters for the RFP believe that they sufficiently understand the relative merits of the Revised Submissions, a selection vote is taken. The result of this selection vote is a recommendation for adoption to the TC. The AB reviews the proposal for MDA compliance and technical merit. An endorsement from the AB moves the voting process into the issuing Technology Committee. An eight-week voting period ensues in which the TC votes to recommend adoption to the OMG Board of Directors (BoD). The final vote, the vote to adopt, is taken by the BoD and is based on technical merit as well as business qualifications. The resulting draft standard is called the *Alpha Specification*.

- Business Committee Questionnaire

  The submitting members whose proposal is recommended for adoption need to submit their response to the BoD Business Committee Questionnaire [BCQ] detailing how they plan to make use of and/or make the resulting standard available in products. If no organization commits to make use of the standard, then the BoD will typically not act on the recommendation to adopt the standard - so it is very important to fulfill this requirement.

- Finalization

  A Finalization Task Force (FTF) is chartered by the TC that issued the RFP, to prepare an Alpha submission for publishing as a Formal (i.e. publicly available) specification, by fixing any problems that are reported by early users of the specification. Upon completion of its activity the FTF recommends adoption of the resulting Beta (draft) specification. The parent TC acts on the recommendation and recommends adoption to the BoD. OMG Technical Editors produce the Formal Specification document based on this Beta Specification.

- Revision

  A Revision Task Force (RTF) is normally chartered by a TC, after the FTF completes its work, to manage issues filed against the Formal Specification by implementers and users. The output of the RTF is a Beta specification reflecting minor technical changes, which the TC and Board will usually approve for adoption as the next version of the Formal Specification.

3.3 Goals of the evaluation

The primary goals of the TF evaluation are to:

- Provide a fair and open process
- Facilitate critical review of the submissions by members of OMG
- Provide feedback to submitters enabling them to address concerns in their revised submissions
- Build consensus on acceptable solutions
- Enable voting members to make an informed selection decision

Submitters are expected to actively contribute to the evaluation process.

4.0 Instructions for Submitters

4.1 OMG Membership

To submit to an RFP issued by the Platform Technology Committee the submitter or submitters must be either Platform or Contributing members on the date of the submission deadline, while for Domain Technology RFPs the submitter or submitters must be either Contributing or Domain members. Submitters sometimes choose to name other organizations that support a submission in some way; however, this has no formal status within the OMG process, and for OMG's purposes confers neither duties nor privileges on the organizations thus named.

4.2 Submission Effort

An RFP submission may require significant effort in terms of document preparation, presentations to the issuing TF, and participation in the TF evaluation process. Several staff months of effort might be necessary. OMG is unable to reimburse submitters for any costs in conjunction with their submissions to this RFP.

4.3 Letter of Intent

A Letter of Intent (LOI) must be submitted to the OMG Business Committee signed by an officer of the submitting organization signifying its intent to respond to the RFP and confirming the organization's willingness to comply with OMG's terms and conditions, and commercial availability requirements. These terms, conditions, and requirements are defined in the *Business Committee RFP Attachment* and are reproduced verbatim in section 4.4 below.

The LOI should designate a single contact point within the submitting organization for receipt of all subsequent information regarding this RFP and the submission. The name of this contact will be made available to all OMG members. The LOI is typically due 60 days before the deadline for initial submissions. LOIs must be sent by fax or paper mail to the "RFP Submissions Desk" at the main OMG address shown on the first page of this RFP.

Here is a suggested template for the Letter of Intent:

This letter confirms the intent of <organization required> (the organization) to submit a response to the OMG <RFP name required> RFP. We will grant OMG and its members the right to copy our response for review purposes as specified in section 4.7 of the RFP. Should our response be adopted by OMG we will comply with the OMG Business Committee terms set out in section 4.4 of the RFP and in document omg/06-03-02.

<contact name and details required> will be responsible for liaison with OMG regarding this RFP response.

The signatory below is an officer of the organization and has the approval and authority to make this commitment on behalf of the organization.

<signature required>

4.4 Business Committee RFP Attachment

This section contains the text of the Business Committee RFP attachment concerning commercial availability requirements placed on submissions. This attachment is available separately as an OMG document omg/06-03-02.

4.5 Responding to RFP items

4.5.1 Complete proposals

A submission must propose full specifications for all of the relevant requirements detailed in Chapter 6 of this RFP. Submissions that do not present complete proposals may be at a disadvantage.

Submitters are highly encouraged to propose solutions to any optional requirements enumerated in Chapter 6.

4.5.2 Additional specifications

Submissions may include additional specifications for items not covered by the RFP that they believe to be necessary and integral to their proposal. Information on these additional items should be clearly distinguished.

Submitters must give a detailed rationale as to why these specifications should also be considered for adoption. However submitters should note that a TF is unlikely to consider additional items that are already on the roadmap of an OMG TF, since this would pre-empt the normal adoption process.

4.5.3 Alternative approaches

Submitters may provide alternative RFP item definitions, categorizations, and groupings so long as the rationale for doing so is clearly stated. Equally, submitters may provide alternative models for how items are provided if there are compelling technological reasons for a different approach.

4.6 Confidential and Proprietary Information

The OMG specification adoption process is an open process. Responses to this RFP become public documents of the OMG and are available to members and non-members alike for perusal. No confidential or proprietary information of any kind will be accepted in a submission to this RFP.

4.7 Copyright Waiver

Every submission document must contain: (i) a waiver of copyright for unlimited duplication by the OMG, and (ii) a limited waiver of copyright that allows each OMG member to make up to fifty (50) copies of the document for review purposes only. See Section 4.9.2 for recommended language.

4.8 Proof of Concept

Submissions must include a “proof of concept” statement, explaining how the submitted specifications have been demonstrated to be technically viable. The technical viability has to do with the state of development and maturity of the technology on which a submission is based. This is not the same as commercial availability. Proof of concept statements can contain any information deemed relevant by the submitter; for example:

> “This specification has completed the design phase and is in the process of being prototyped.”
>
> “An implementation of this specification has been in beta-test for 4 months.”

> "A named product (with a specified customer base) is a realization of this specification."

It is incumbent upon submitters to demonstrate the technical viability of their proposal to the satisfaction of the TF managing the evaluation process. OMG will favor proposals based on technology for which sufficient relevant experience has been gained.

4.9 Format of RFP Submissions

This section presents the structure of a submission in response to an RFP. *All submissions* must contain the elements itemized in section 4.9.2 below before they can be accepted as a valid response for evaluation or a vote can be taken to recommend for adoption.

4.9.1 General

- Submissions that are concise and easy to read will inevitably receive more consideration.
- Submitted documentation should be confined to that directly relevant to the items requested in the RFP. If this is not practical, submitters must make clear what portion of the documentation pertains directly to the RFP and what portion does not.
- The key words "**must**", "**must not**", "**required**", "**shall**", "**shall not**", "**should**", "**should not**", "**recommended**", "**may**", and "**optional**" shall be used in the submissions with the meanings as described in RFC 2119 [RFC2119].

4.9.2 Required Outline

A three-part structure for submissions is required. Part I is non-normative, providing information relevant to the evaluation of the proposed specification. Part II is normative, representing the proposed specification. Specific sections like Appendices may be explicitly identified as non-normative in Part II. Part III is normative specifying changes that must be made to previously adopted specifications in order to be able to implement the specification proposed in Part II.

PART I

- •A cover page carrying the following information (a template for this is available [Inventory]):
  - The full name of the submission

  - The primary contact for the submission
  - The acronym proposed for the specification (e.g. UML, CORBA)
  - The name and document number of the RFP to which this is a response
  - The document number of the main submission document
  - An inventory of all accompanying documents, with OMG document number, short description, a URL where appropriate, and whether they are normative.

- List of OMG members making the submission (see 4.1) listing exactly which members are making the submission, so that submitters can be matched with LOI responders and their current eligibility can be verified.

- Copyright waiver (see 4.7), in a form acceptable to the OMG.

  One acceptable form is:

  "Each of the entities listed above: (i) grants to the Object Management Group, Inc. (OMG) a nonexclusive, royalty-free, paid up, worldwide license to copy and distribute this document and to modify this document and distribute copies of the modified version, and (ii) grants to each member of the OMG a nonexclusive, royalty-free, paid up, worldwide license to make up to fifty (50) copies of this document for internal review purposes only and not for distribution, and (iii) has agreed that no person shall be deemed to have infringed the copyright in the included material of any such copyright holder by reason of having used any OMG specification that may be based hereon or having conformed any computer software to such specification."

  If you wish to use some other form you must get it approved by the OMG legal counsel before using it in a submission.

- For each member making the submission, an individual contact point who is authorized by the member to officially state the member's position relative to the submission, including matters related to copyright ownership, etc. (see 4.3)

- Overview or guide to the material in the submission

- Overall design rationale (if appropriate)

- Statement of proof of concept (see 4.8)

- Resolution of RFP requirements and requests

Explain how the proposal satisfies the specific requirements and (if applicable) requests stated in Chapter 6. References to supporting material in Part II should be given.

In addition, if the proposal does not satisfy any of the general requirements stated in Chapter 5, provide a detailed rationale.

- Responses to RFP issues to be discussed

  Discuss each of the "Issues To Be Discussed" identified in Chapter 6.

PART II

The contents of this part should be structured based on the template found in [FORMS] and should contain the following elements as per the instructions in the template document cited above:

- Scope of the proposed specification
- Proposed conformance criteria

  Submissions should propose appropriate conformance criteria for implementations.

- Proposed normative references

  Submissions should provide a list of the normative references that are used by the proposed specification

- Proposed list of terms and definitions

  Submissions should provide a list of terms that are used in the proposed specification with their definitions.

- Proposed list of symbols

  Submissions should provide a list of special symbols that are used in the proposed specification together with their significance

- Proposed specification

PART III

- Changes or extensions required to existing OMG specifications

  Submissions must include a full specification of any changes or extensions required to existing OMG specifications. This should be in a form that enables “mechanical” section-by-section revision of the existing specification.

4.10 How to Submit

Submitters should send an electronic version of their submission to the *RFP Submissions Desk* (*omg-documents@omg.org*) at OMG Headquarters by 5:00 PM U.S. Eastern Standard Time (22:00 GMT) on the day of the Initial and Revised Submission deadlines. Acceptable formats are Adobe FrameMaker source, ODF (ISO/IEC 26300), OASIS Darwin Information Typing Architecture (DITA) or OASIS DocBook 4.x (or later).

Submitters should make sure they receive electronic or voice confirmation of the successful receipt of their submission. Submitters should be prepared to send a single hardcopy version of their submission, if requested by OMG staff, to the attention of the “RFP Submissions Desk” at the main OMG address shown on the first page of this RFP.

5.0 General Requirements on Proposals

5.1 Requirements

5.1.1 Submitters are encouraged to express models using OMG modeling languages such as UML, MOF, CWM and SPEM (subject to any further constraints on the types of the models and modeling technologies specified in Chapter 6 of this RFP). Submissions containing models expressed via OMG modeling languages shall be accompanied by an OMG XMI [XMI] representation of the models (including a machine-readable copy). A best effort should be made to provide an OMG XMI representation even in those cases where models are expressed via non-OMG modeling languages.

5.1.2 Chapter 6 of this RFP specifies whether PIM(s), PSM(s), or both are being solicited. If proposals specify a PIM and corresponding PSM(s), then the rules specifying the mapping(s) between the PIM and PSM(s) shall either be identified by reference to a standard mapping or specified in the proposal. In order to allow possible inconsistencies in a proposal to be resolved later, proposals shall identify whether the mapping technique or the resulting PSM(s) are to be considered normative.

5.1.3 Proposals shall be *precise* and *functionally complete*. All relevant assumptions and context required for implementing the specification shall be provided.

5.1.4 Proposals shall specify *conformance criteria* that clearly state what features all implementations must support and which features (if any) may *optionally* be supported.

5.1.5 Proposals shall *reuse* existing OMG and other standard specifications in preference to defining new models to specify similar functionality.

5.1.6 Proposals shall justify and fully specify any *changes or extensions* required to existing OMG specifications. In general, OMG favors proposals that are *upwards compatible* with existing standards and that minimize changes and extensions to existing specifications.

5.1.7 Proposals shall factor out functionality that could be used in different contexts and specify their models, interfaces, etc. separately. Such *minimalism* fosters re-use and avoids functional duplication.

5.1.8 Proposals shall use or depend on other specifications only where it is actually necessary. While re-use of existing specifications to avoid duplication will be encouraged, proposals should avoid gratuitous use.

5.1.9 Proposals shall be *compatible* with and *usable* with existing specifications from OMG and other standards bodies, as appropriate. Separate specifications offering distinct functionality should be usable together where it makes sense to do so.

5.1.10 Proposals shall preserve maximum *implementation flexibility*. Implementation descriptions should not be included and proposals shall not constrain implementations any more than is necessary to promote interoperability.

5.1.11 Proposals shall allow *independent implementations* that are *substitutable* and *interoperable*. An implementation should be replaceable by an alternative implementation without requiring changes to any client.

5.1.12 Proposals shall be compatible with the architecture for system distribution defined in ISO's Reference Model of Open Distributed Processing [RM-ODP]. Where such compatibility is not achieved, or is not appropriate, the response to the RFP must include reasons why compatibility is not appropriate and an outline of any plans to achieve such compatibility in the future.

5.1.13 In order to demonstrate that the specification proposed in response to this RFP can be made secure in environments requiring security, answers to the following questions shall be provided:

- What, if any, are the security sensitive elements that are introduced by the proposal?
- Which accesses to security-sensitive elements must be subject to security policy control?
- Does the proposed service or facility need to be security aware?
- What default policies (e.g., for authentication, audit, authorization, message protection etc.) should be applied to the security sensitive elements introduced by the proposal? Of what security considerations must the implementers of your proposal be aware?

The OMG has adopted several specifications, which cover different aspects of security and provide useful resources in formulating responses. [CSIV2] [SEC] [RAD].

5.1.14 Proposals shall specify the degree of internationalization support that they provide. The degrees of support are as follows:

a) Uncategorized: Internationalization has not been considered.

b) Specific to <region name>: The proposal supports the customs of the specified region only, and is not guaranteed to support the customs of any other region. Any fault or error caused by requesting the services outside of a context in which the customs of the specified region are being consistently followed is the responsibility of the requester.

c) Specific to <multiple region names>: The proposal supports the customs of the specified regions only, and is not guaranteed to support the customs of any other regions. Any fault or error caused by requesting the services outside of a context in which the customs of at least one of the specified regions are being consistently followed is the responsibility of the requester.

d) Explicitly not specific to <region(s) name>: The proposal does not support the customs of the specified region(s). Any fault or error caused by requesting the services in a context in which the customs of the specified region(s) are being followed is the responsibility of the requester.

5.2 Evaluation criteria

Although the OMG adopts model-based specifications and not implementations of those specifications, the technical viability of implementations will be taken into account during the evaluation process. The following criteria will be used:

5.2.1 Performance

Potential implementation trade-offs for performance will be considered.

5.2.2 Portability

The ease of implementation on a variety of systems and software platforms will be considered.

5.2.3 Securability

The answer to questions in section 5.1.13 shall be taken into consideration to ascertain that an implementation of the proposal is securable in an environment requiring security.

5.2.4 Conformance: Inspectability and Testability

The adequacy of proposed specifications for the purposes of conformance inspection and testing will be considered. Specifications should provide sufficient constraints on interfaces and implementation characteristics to ensure that conformance can be unambiguously assessed through both manual inspection and automated testing.

5.2.5 Standardized Metadata

Where proposals incorporate metadata specifications, usage of OMG standard XMI metadata [XMI] representations must be provided as this allows specifications to be easily interchanged between XMI compliant tools and applications. Since use of XML (including XMI and XML/Value [XML/Value]) is evolving rapidly, the use of industry specific XML vocabularies (which may not be XMI compliant) is acceptable where justified.

6.0 Specific Requirements on Proposals

6.1 Problem Statement

Current DDS Systems meet Information Assurance requirements by isolating DDS applications into a security enclave running at "system high". Inside the "protected domain" applications are authorized to publish and subscribe to any data in the DDS Global Data Space. Once inside the protected domain applications are not authenticated; their data and meta-data is sent unencrypted, often using multicast. The standard OMG DDS infrastructure provides no guarantees (other than those provided by the physical protection of the system) on confidentiality, pedigree, or integrity of the information.

What is needed is a set of Information Assurance extensions to the DDS standard that provides the necessary support for Authentication, Authorization and Access Control, Confidentiality, Integrity, and Non-repudiation for all the real-time data sent over DDS.

One possible approach would be to enforce Mandatory Access Control (MAC) on all applications that join a DDS Global Data-Space, requiring them to be authenticated and have the necessary credentials. Beyond access to the DDS Global Data Space, the desired approach should provide finer-grain (e.g. role-based or more generally, policy-based) access control to specific DDS Topics and even to specific fields within DDS Topics. It should ensure confidentiality of the information, integrity, pedigree, and non-repudiation. Finally, it should be able to handle the scalable publish-subscribe deployment scenarios, specifically the one-to-many (multicast) distribution of encrypted information and maintain the DDS real-time QoS in the distribution of information to multiple subscribers.

The required Security Architecture needs to be configurable (e.g. via newly added QoS polices) to provide simple access to standard, interoperable, pre-defined security policies out-of-the box. At the same time it should remain open and extensible (e.g. via “plug-in” SPIs) so that application developers can integrate with pre-existing Identity Management Mechanisms, Authorization Policy repositories, or cryptographic libraries, which might be program or project specific.

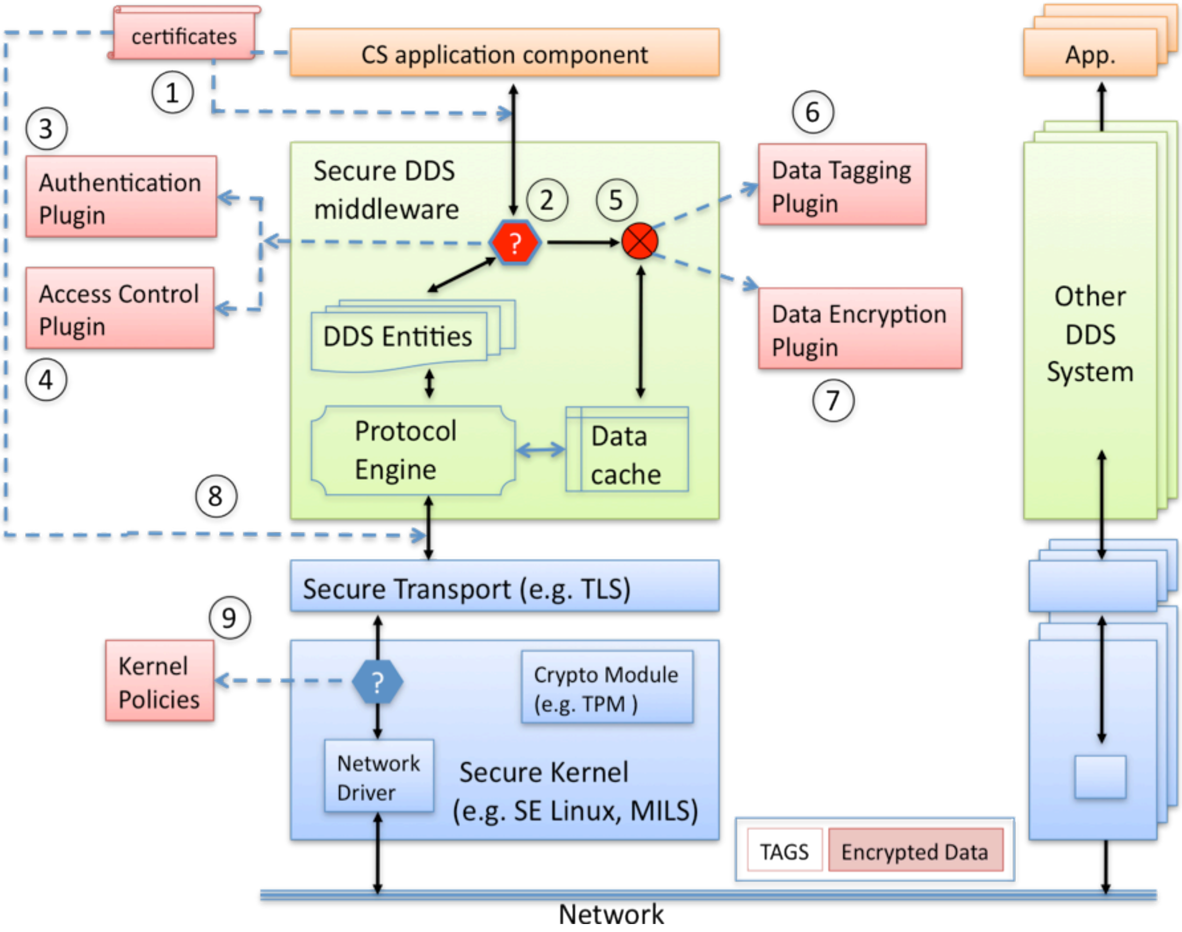
6.2 Scope of Proposals Sought

This RFP solicits submissions for:

(1) A PIM and language PSMs for a set of security interception points within the operation of DDS middleware implementations and corresponding security-interceptor SPI contracts at each interception point (security plugin SPIs) that provide Authentication, Authorization, Data Tagging, Signing, and Encryption

(2) A set of pre-defined behaviors for security plugin SPI implementations (security plugins) that must be supported by compliant implementations of this specification, which can be used to define and enforce security policies out-of-the-box.

(3) Any extensions to the DDS-RTPS Wire Protocol necessary to support secure interoperability between DDS Systems that use one of the pre-defined behaviors for the security plugins.



6.2.1 PIM and language PSMs for a set of interception points to be used by DDS implementations and corresponding security plugin SPIs that provide access to Authentication, Authorization, Data Tagging, Signing, and Encryption

There are two primary reasons for requiring a pluggable architecture:

First, high-grade security needs to be grounded in hardware and operating system facilities. For example many computers include a Trusted Platform Module (TPM) that can be used to establish the identity of both the hardware and key elements of the software stack. This module can provide services such as (a) Public key management and authentication services, where the private keys are always kept in the hardware so that they are never visible outside the chip; (b) boot-time hardware verification of the Operating System and Services software integrity (e.g., by verifying the cryptographic signature of the software elements) so that tampering can be prevented; and (c) sealed storage, that is, small amounts of secure storage for sensitive information such as private keys.

Because these functions are performed in hardware, they can be more efficient than equivalent software approaches. More importantly, the TPM hardware typically includes anti-tamper measures that make it invulnerable to software attacks and very resistant to hardware-based attacks.

A pluggable architecture may be used to integrate TPM as well other facilities such as biometric readers or secure ID cards used form multi-factor authentication.

Second, DDS systems are often part of a larger infrastructure that mixes multiple software technology stacks as well as legacy application components. Many of these systems have their own security infrastructure (e.g. key management, authentication and identity services, single-sign-on, etc.). In some cases the security infrastructure might even be custom built for a particular application. To enable such systems to securely use DDS it is imperative that DDS provides the means to “plugin” and adapt these external components and use them to enforce the security policies.

Third, a pluggable architecture allows third parties, such as security providers, to provide implementations of the SPIs.

This “pluggable” approach is analogous to the Java™ Authentication and Authorization Service (JAAS), which performs authentication and authorization in a pluggable fashion.

6.2.2 Pre-defined behaviors for a set security plugin SPI implementations (security plugins) that must be supported by compliant implementations of this specification and can be used to define and enforce security policies out-of-the-box.

There are two mains reasons for requiring pre-defined security plugins.

First, many users do not have the expertise or time required to develop their own plug-ins. Having out-of-the-box pre-defined plugins that provide the critical capabilities is therefore essential to enable this uses to build secure systems using DDS.

Second, many systems that use DDS integrate components independently developed by separate vendors. The only form of centralized design authority is often a separate organization that does no development, limiting itself to defining the Global Data space in terms of Topics, Schemas, and Qos. Moreover, once a system is deployed, new components are deployed and integrated, often without access to common source code used for the original components. DDS currently enables this kind of deployment cleanly: all that is needed is access to the Type Description and Topic Names, which can even be discovered at run-time using the standard DDS SPIs.

To allow such “up front” Global Data space design and support incremental evolution of systems it is required to provide standardized security model and mechanisms, such that they can be included in the design phase and are guaranteed to be present.

6.2.3 Any extensions to the DDS-RTPS Wire Protocol necessary to support secure interoperability

The DDS-RTPS Interoperability Wire Protocol specification will need to be extended to support many aspects of Secure DDS such as: secure discovery and propagation of security credentials, data-tagging, data encryption, data-signing, and integration of secure transports.

The standard DDS-RTPS Wire Protocol leverages multicast for scalability and ease of configuration. Extending the use of multicast so that it can be used to also send secure data is important to preserve the performance characteristics of DDS systems.

6.2.4 Goals of this specification

This specification must define a PIM for a DDS Security Model and a plugin architecture aligned with that security model, enabling DDS to consistently use Authentication, Authorization, Data Tagging, Signing, and Encryption. The PIM for this plugin will specify how it interacts with the regular DDS operations, including the operations it might intercept and any new return values that might be introduced.

The PIM must then be mapped to all the PSMs that DDS currently supports so that it is possible to use it regardless of the programming language the developer uses.

In addition the specification must define a set of pre-defined plugins that allow users to easily configure their system and define policies that utilize the DDS Security Model.

Finally there must be a set of extensions to the DDS-RTPS Interoperability Wire protocol allowing DDS systems implemented using middleware from different vendors to interoperate securely.

6.3 Relationship to other OMG Specifications and activities

6.3.1 Relationship to OMG specifications

Main relevant specifications:

- The DDS specification (formal/2005-12-04)
- DDS-RTPS Interoperability Wire Protocol specification (formal/2009-01-05)
- The Extensible and Dynamic Topic Types for DDS Beta Specification (ptc/2010-05-12)

6.3.2 Relationship to other OMG Documents and work in progress

Main relevant specifications:

- The Web-Enabled DDS Specification. RFP (mars/2009-09-19).
- The Data Tagging and Labelling for Security and Privacy RFI (c4i/2007-09-04)
- The Joint Thales & Prismtech Response to the Data Tagging and Labelling for Security and Privacy RFI (c4i/2008-05-01)

6.4 Related non-OMG Activities, Documents and Standards

- Java™ Authentication and Authorization Service (JAAS). http://download.oracle.com/javase/6/docs/technotes/guides/security/jaas/JAASRefGuide.html

- Java™ Cryptography Architecture JCA. http://download.oracle.com/javase/6/docs/technotes/guides/security/crypto/CryptoSpec.html

- Trusted Platform Module (TPM) Specifications". Trusted Computing Group. http://www.trustedcomputinggroup.org/resources/tpm_main_specification

- SAML. http://www.oasis-open.org/committees/download.php/27819/sstc-saml-tech-overview-2.0-cd-02.pdf

- Common Architecture Label IPv6 Security Option (CALIPSO)

- National Institute of Standards and Technology, "FIPS PUB 188 Standard Security Labels for Information Transfer", September 1994

- The Joint C3 Information Exchange Data Model. http://www.mip-site.org/010_Public_Home_News.htm

- Datagram Transport Layer Security (DTLS). http://tools.ietf.org/html/rfc4347

6.5 Mandatory Requirements

6.5.1 Proposals shall define a Platform Independent Security Model for DDS. The model shall be independent of the programming language used to build DDS applications.

6.5.1.1 The Security Model for DDS shall support mechanisms that establish the ability of a DDS Domain Participant to run in a platform.

6.5.1.2 The Security Model for DDS shall support mechanisms to configure and access the credentials of the underlying DDS Domain Participants.

[1] The mechanism shall allow the application to configure the security credentials that will be used by DDS Domain Participants it creates.

[2] The mechanism shall allow the application to access the security credentials of the DDS Domain Participants it has created.

6.5.1.3 The Security Model for DDS shall allow specification of authorization policies

[1] The authorization policies shall control the DDS Domain Participants that are allowed to join a DDS Domain.

[2] The authorization policies shall control access to DDS discovery information (DDS Built-in Topics).

[3] The authorization policies shall control the DDS Domain Participants allowed to publish a Topic on the domain.

[4] The authorization policies shall control the DDS Domain Participants allowed to subscribe to a Topic on the domain.

[5] The authorization policies shall control the DDS Domain Participants allowed to publish to a DDS Partition on the domain.

[6] The authorization policies shall control the DDS Domain Participants allowed to subscribe to a DDS Partition on the domain.

6.5.1.4 The Security Model for DDS shall include the concept of data tagging.

[1] The Model shall support applying different Tags to data written by each DDS DataWriter.

6.5.1.5 The Security Model for DDS shall support mechanisms for ensuring the integrity for the data delivered.

[1] The Model shall provide the means to ensure traceability, pedigree, and tamper-detection of the data published.

[2] The Model shall support digital signatures for the data.

[3] The Model shall support encryption of the data.

[4] The Model shall support the use of separate cryptographic keys for the data written by each DDS DataWriter.

6.5.2 Proposals shall define a collection of Platform Independent interception points and SPI contracts that can be implemented by DDS Security Plugins. The definition of the plugin SPIs shall be independent of the programming language used to build DDS applications.

6.5.2.1 The Plugin SPIs shall allow applications to exchange credentials with a DDS Domain Participant.

[1] The Plugin SPIs shall allow full support for exchanging credentials for authentication.

[2] The Plugin SPIs shall support delegation of authority for authentication.

6.5.2.2 The Plugin SPIs shall allow an external plugin to perform all the authorization functions of the Security Model for DDS.

[1] The Plugin SPIs shall allow full support of the authorization policies defined by the DDS Security model.

[2] The Plugin SPIs shall support delegation of authority.

[3] The Plugin SPIs shall support delegation of authority for authorization separately for each DDS Topic within a DDS Domain.

6.5.2.3 The Plugin SPIs shall allow an external plugin to perform all the tagging and tag-accessing functions of the Security Model for DDS.

6.5.2.4 The Plugin SPIs shall allow an external plugin to perform all the encryption and decryption functions of the Security Model for DDS.

6.5.2.5 The Plugin SPIs shall allow an external plugin to perform all the digital signing and verification functions of the Security Model for DDS.

6.5.3 Proposals shall define a collection of built-in Platform Independent plugins that implement the Platform Independent interfaces.

6.5.3.1 The built-in plugins shall allow means for accessing and configuring the DDS Security policies

6.5.4 Proposals shall define Platform Specific Mappings for the built-in plugins to all the language PSMs supported by DDS.

6.5.4.1 The mapping to the Language specific PSM shall be consistent with the mappings of the DDS PIM to that language PSM.

6.5.5 Proposals shall define how the Real-Time Publish Subscribe Protocol DDS Interoperability Wire Protocol (RTPS) is used to allow DDS applications to interoperate securely, specifically:

6.5.5.1 Proposals shall use RTPS in a way that does not break interoperability with existing RTPS compliant applications.

6.5.5.2 Proposals shall use the extensibility mechanisms already present in RTPS to add any security-related features to the protocol.

[1] Proposals shall not introduce new RTPS submessages but may introduce new built-in Topics that reuse the existing RTPS submessages.

[2] Proposals shall define how identity credentials produced by the authentication plugin are encapsulated and propagated by the standard RTPS discovery mechanisms.

[3] Proposals shall define how authorization tokens produced by the authorization plugin are encapsulated and propagated by the standard RTPS discovery mechanisms.

[4] Proposals shall define how security-related metadata, such as, tags, digital signatures, etc. appear in the RTPS submessages.

[5] Proposals shall define how the encrypted data produced by the encryption plugin appears in the RTPS submessages.

[6] Proposals shall define how the RTPS protocol in its use of multicast can be secured.

6.6 Optional Requirements

6.6.1 Proposals may define authorization policies that control the content that a DDS Domain Participant is allowed to publish on a Topic.

6.6.2 Proposals may define authorization policies that control the content that a DDS Domain Participant is allowed to subscribe on a Topic. The policies may support granularity at the level of specific DDS data-objects (identified by their Key fields) within a DDS Topic.

6.6.3 Proposals may define authorization policies that control the Qos Policies values that a DDS Domain Participants can use when publishing a Topic.

6.6.4 Proposals may define authorization policies that control the Qos Policies values that a DDS Domain Participant can use when subscribing to a Topic.

6.6.5 Proposals may define data-tagging plugins that apply different tags for each data-sample published by a DDS DataWriter.

6.6.6 Proposals may define built-in plugins that interoperate with standard authentication and authorization protocols and services, such as, LDAP and SAML.

6.6.7 Proposals may define a PSM mapping of the DDS-RTPS Interoperability Wire Protocol to a secure transport, such as, DTLS.

6.6.8 Proposals may define a PSM of the DDS-RTPS Interoperability Wire Protocol allowing interoperability over Unidirectional Transports.

6.7 Issues to be discussed

6.7.1 Proposals shall discuss the DDS Security mechanisms that can be integrated and deployed in a SELinux platform.

6.7.2 Proposals shall discuss the DDS Security mechanisms that can be integrated and deployed in a MILS platform.

6.7.3 Proposals shall discuss the operation of a compliant implementation in a MLS environment.

6.7.4 Proposals shall discuss how the DDS Security mechanisms can be shared with the mechanisms provided by other technology platforms, such as, Web Services.

6.7.5 Proposals shall justify the situations where this specification chooses to introduce a new standard rather than reuse an existing standard.

6.8 Evaluation Criteria

- The level of effort required to integrate the new extensions into the current DDS Model and implementations.
- How well the proposal preserves the architectural integrity of the DDS data-centric publish-subscribe model.
- How well the proposal preserves the performance and scalability of DDS.
- Clarity and completeness of the submission.
- Degree of use of existing security standards and infrastructure, such as LDAP, SAML, and DTLS.

6.9 Other information unique to this RFP

None

6.10 RFP Timetable

The timetable for this RFP is given below. Note that the TF or its parent TC may, in certain circumstances, extend deadlines while the RFP is running, or may elect to have more than one Revised Submission step. The latest timetable can always be found at the OMG *Work In Progress* page at http://www.omg.org/schedules under the item identified by the name of this RFP. Note that "<month>" and "<approximate month>" is the name of the month spelled out; e.g., January.

Event or Activity	Actual Date
Preparation of RFP by TF	
RFP placed on OMG document server	*November 8, 2010*
Approval of RFP by Architecture Board *Review by TC*	*December 9, 2010*
TC votes to issue RFP	*December 10, 2010*
LOI to submit to RFP due	*April 30, 2011*
Initial Submissions due and placed on OMG document server ("Four week rule")	*May 23, 2011*
Voter registration closes	*June 13, 2011*

Initial Submission presentations	*June 20-24, 2011*
Preliminary evaluation by TF	*June 20-24, 2011*
Revised Submissions due and placed on OMG document server ("Four week rule")	*November 7, 2011*
Revised Submission presentations	*December 5-9, 2011*
Final evaluation and selection by TF *Recommendation to AB and TC*	*December 5-9, 2011*
Approval by Architecture Board *Review by TC*	*December 5-9, 2011*
TC votes to recommend specification	*December 9, 2011*
BoD votes to adopt specification	*March 2012*

Appendix A References and Glossary Specific to this RFP

A.1 References Specific to this RFP

[C4I_DATA_TAGGING] Data Tagging and Labeling for Security and Privacy RFI. OMG document c4i/2007-09-04

[DTLS] Datagram Transport Layer Security. IETF RFC 4347. http://tools.ietf.org/html/rfc4347

[FIPS140] NIST, “FIPS PUB 140-1 Security Requirements for Cryptographic Modules”, January 1994

[JAAS] Java Authentication and Authorization Architecture for the Java™ SE Development Kit 6: http://download.oracle.com/javase/6/docs/technotes/guides/security/jaas/JAASRefGuide.html. Now part of Java SE Security: http://www.oracle.com/technetwork/java/javase/tech/index-jsp-136007.html

[MILS] Multiple Independent Levels of Security: Alves-Foss, W. S. Harrison, P. Oman and C. Taylor (2007). "The MILS Architecture for High Assurance Embedded Systems". International Journal of Embedded Systems. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.76.6810&rep=rep1&type=pdf

[MLS] Multi-Level Security: CSC-STD-004-85: Computer Security Requirements - Guidance For Applying The Department Of Defense Trusted Computer System Evaluation Criteria In Specific Environments (25 June 1985) http://csrc.nist.gov/publications/secpubs/rainbow/std004.txt

[NSS] Network Security Services (NSS). http://www.mozilla.org/projects/security/pki/nss/

[OPENSSL] OpenSSL: The Open Source toolkit for SSL/TLS. http://www.openssl.org/

[PKI] The Public Key Infrastructure (PKI) refers to a collection of technology that supports the creation, management, storing, distribution, and revocation of digital certificates. There are many standards such as X.509, PGP, etc. under this umbrella.

[SAML] Security Assertion Markup Language (SAML). OASIS SAML V2.0 specification http://saml.xml.org/saml-specifications#samlv20

[SELINUX] SELinux: US NSA Security-Enhanced Linux. http://www.nsa.gov/research/selinux/index.shtml.

[TLS] The Transport Layer Security (TLS) Protocol Version 1.2. IETF RFC 5246. http://tools.ietf.org/html/rfc5246

[TPM] Trusted Platform Module (TPM) Specifications". Trusted Computing Group. http://www.trustedcomputinggroup.org/resources/tpm_main_specification

A.2 Glossary Specific to this RFP

DDS – Data Distribution Service and OMG specification for Data-Centric Publish Subscribe middleware.

DTLS – Datagram Transport Security. A Datagram version of TLS (see below).

JAAS – Java™ Authentication and Authorization Service. The Java™ security framework for user-centric security to augment the Java code-based security. Starting with version 1.4 JAAS is integrated into the JRE.

MILS - Multiple Independent Levels of Security. MILS is a high-assurance security architecture based on the concepts of separation and controlled information flow; implemented by separation mechanisms that support both untrusted and trustworthy components.

MLS - Multi-Level Security. MLS is the application of a computer system to process information with different sensitivities (i.e., at different security levels), permit simultaneous access by users with different security clearances and needs-to-know, and prevent users from obtaining access to information for which they lack authorization.

RTPS – Real-Time Publish-Subscribe Protocol. The Wire protocol used by DDS.

SAML - Security Assertion Markup Language. an XML-based standard for exchanging authentication and authorization data between security domains

SPI – Service Provider Interfaces is a software mechanism to support replaceable components. It is an interface intended to be implemented by maintainers of a service infrastructure.

SELinux – Security Enhanced Linux. A Linux feature that provides a mechanism for supporting access control security policies, including mandatory access controls.

TLS- Transport Layer Security are standard cryptographic protocols that provide communications security over the Internet.

TPM – Trusted Platform Module. A specification detailing a secure crypto-processor that can store cryptographic keys that protect information.

Appendix B General Reference and Glossary