Integrating DDS into secure net-centric systems: A pragmatic approach

RTESS, Washington DC, July 2008

Ariel Salomon
Senior Software Engineer
Real-Time Innovations, Inc.
ariel.salomon@rti.com

Gerardo Pardo-Castellote, Ph.D.
Chief Technology Officer
Real-Time Innovations, Inc.
gerardo@rti.com
Agenda

- Motivation & Requirements
- Net-centric approach to DDS Security
- Proposed DDS Security Model
- Pragmatic Implementation of Secure Domains
  - Mandatory Access Control at Transport Layer
  - Operating System Security Integration
  - Performance Implications for Real-Time
- Integrated DDS Security with Role-Based Access Control (RBAC)
- Additional Concerns
- Conclusion
Motivation

- Net-centric systems rely on information sharing…
  - They are built on top a Shared Information Space or Common-Operational Picture
  - They assume information availability where and when it is needed
- As the system scope expands Security/IA becomes a concern
  - Information flow must be restricted to the intended/authorized recipients
  - Can no longer assume a single protection domain.
Application Requirements

- High-level goal:
  “control and restrict access to information such that it is only accessible to the intended/authorized recipients.”

- Secondary goals include:
  - guaranteed access to the information by the authorized users
    - prevent denial of service attacks
  - audit trails
  - ensure non-repudiation (guarantee identity of the source of the information)
  - deployable within the exiting Internet Environment

- This breaks down into
  - Functional requirements:
    - Authentication, Confidentiality, Integrity
    - Availability
  - Non-functional requirements:
    - Manageability, Accountability, Assurance
  - Deployment requirements:
    - Work in WAN environment, operate across NATs / IPv6 / lossy and bandwidth-constrained channels
    - Work in conjunction with other network security measures
Agenda

- Motivation & Requirements
- Net-centric approach to DDS Security
- Proposed DDS Security Model
- Pragmatic Implementation of Secure Domains
  - Mandatory Access Control at Transport Layer
  - Operating System Security Integration
  - Performance Implications for Real-Time
- Integrated DDS Security with RBAC
- Additional Concerns
- Conclusion
DDS Communications model

Provides a “**Global Data Space**” that is accessible to all interested applications.

- Data objects addressed by **Domain**, **Topic** and **Key**
- Subscriptions are **decoupled** from Publications
- Contracts established by means of **QoS**
- Automatic **discovery** and **configuration**
Security Terminology

- **Principals and Subjects**
  - The active entities
    - Principals are the entities that can be granted access control (e.g. the human users).
    - Subjects are active computer-system entities that bound to principals and operate on their behalf. Subjects perform the actions on the objects.

- **Objects**
  - The passive entities
    - can be accessed or manipulated
      - e.g. memory, files, sockets, semaphores, etc.

- **Access operations**
  - the actions that can be performed on the objects
    - e.g. read memory, write a message on a socket.

- **Security model**
  - The architecture, concepts, and algorithms used to define the access control restrictions on the objects and the access rights given to principals.
  - Examples are Mandatory Access Control (MAC), Role-based Access Control (RBAC), Discretionary Access Control (DAC), etc.

- **Security policy**
  - a set of goals, rules, and regulations regarding the access rights to information and objects.
  - defines what is allowed and what is not allowed within a particular security model.
  - Examples are the Bell LaPadula model, the Biba model, Chinese Wall, etc.
Discovery records about entities are also security objects.

security subject

Topic “green”  Topic “orange”
DDS communications model

- Publisher declares information it has and specifies the Topic
  - and the offered QoS contract
  - and an associated listener to be alerted of any significant status changes
- Subscriber declares information it wants and specifies the Topic
  - and the requested QoS contract
  - and an associated listener to be alerted of any significant status changes
- DDS automatically discovers publishers and subscribers
  - DDS ensures QoS matching and alerts of inconsistencies
**DDS access control model**

*Joining a domain is an access operation*

- Participant A declares credentials (ID) it has to join domain
- Participant B declares credentials it has to join domain
- DDS limits discovery
  - Verify credentials received over discovery, apply security policy
  - Do not communicate with peers if they do not have proper credentials
  - Output: Provide notification, audit trail for access failures
DDS access control model

Creating a reader or writer is an access operation

- Participant A declares credentials (ID) it has to join domain, publish data
  - and creates writer with topic, offered QoS contract, listener, ...
- Participant B declares credentials it has to join domain, subscribe to data
  - and creates reader with topic, requested QoS contract, listener, ...
- DDS limits discovery to allowed publishers and subscribers
  - Verify credentials received over discovery, apply security policy
  - Do not communicate with peers if they do not have proper credentials
  - Output: Provide notification, audit trail for access failures
Agenda

- Motivation & Requirements
- Net-centric approach to DDS Security
- Proposed DDS Security Model
  - Pragmatic Implementation of Secure Domains
    - Mandatory Access Control at Transport Layer
    - Operating System Security Integration
    - Performance Implications for Real-Time
- Integrated DDS Security with RBAC
- Additional Concerns
- Conclusion
Proposed DDS Security Model (Pardo, 2007)

- The approach is based on two ideas:
  - Secure Domains
  - Confidential Topics

- **Secure Domains**
  - Limit each Global Data Space (DDS *Domain*) to contain information at a single level of security
  - Only authorized participants are allowed to join a Domain
  - All domain communications are confidential
  - All information is accessible to all members of the domain

- **Confidential need-to-know Topics**
  - Within a Global Data Space allow participants to read and write *Topics* on a “need to know basis”
  - Separate “right to read” from “right to write”
  - Each Topic is authorized and protected separately
Confidential DDS Topics

- The RBAC model applied to DDS Topics

- Each **Topic** is assigned a list of “reader roles” and a list of “writer roles”.
  - ‘reader roles’ are the roles of the principals that can read the Topic
  - ‘writer roles’ are the roles of principals that can write the Topic

- Each **Participant** attached to the **Domain** is assigned a set of roles.
  - Write access to the **Topic** given only to **Participants** that have a role that appears in the list of “writer roles” for the **Topic**
  - Read access to the **Topic** given only to **Participants** that have a role that appears in the list of “reader roles” for the **Topic**
  - These access operation can be limited in terms of allowed QoS

- Result:
  - Limits access to the Topic only to the principals (the DDS **Participants**) that have the “need to know” for that Topic
  - A single domain can carry traffic of multiple security levels
Agenda

- Motivation & Requirements
- Net-centric approach to DDS Security
- Proposed DDS Security Model
- Pragmatic Implementation of Secure Domains
  - Mandatory Access Control at Transport Layer
  - Operating System Security Integration
  - Performance Implications for Real-Time
- Integrated DDS Security with RBAC
- Additional Concerns
- Conclusion
Mandatory Access Control at Transport Layer

- Use DTLS – datagram flavor of TLS/SSL
  - TLS/SSL provides a handshake to transfer certificates and establish security context
- Once verified DTLS establishes secure communications between each pair of participants
- Also provides integrity from cryptographic hash

Disadvantages:
- Multicast is not supported
- All-or-nothing security
RTI Secure Transport

- DDS
- DTLS
- RTPS Discovery Traffic
- RTPS User Traffic

Domain Participant 1

- DTLS traffic
- DTLS handshaking
- Encrypted RTPS
- RTPS Discovery Traffic
- RTPS User Traffic

Domain Participant 2

- DTLS handshaking
- RTPS discovery
- RTPS user traffic

Application 1

Application 2
Agenda

- Motivation & Requirements
- Net-centric approach to DDS Security
- Proposed DDS Security Model
- Pragmatic Implementation of Secure Domains
  - Mandatory Access Control at Transport Layer
  - Operating System Security Integration
  - Performance Implications for Real-Time
- Integrated DDS Security with RBAC
- Additional Concerns
- Conclusion
Operating System Security Integration

- Trusted operating systems (such as SELinux, Trusted Solaris) provide fine-grained access control for system resources:
  - Network ports
  - Inter-process communication (IPC) objects

- These policies can be mapped to the required resources to join a DDS domain
  - DDS Wire Protocol standard specifies ports used for discovery, default ports for user data
  - For a configured system, there is a required set of ports and/or IPC objects that must be used to join the domain

- This requires no changes to DDS, and provides another mechanism for domain separation
Agenda

- Motivation & Requirements
- Net-centric approach to DDS Security
- Proposed DDS Security Model
- Pragmatic Implementation of Secure Domains
  - Mandatory Access Control at Transport Layer
  - Operating System Security Integration
  - Performance Implications for Real-Time
- Integrated DDS Security with RBAC
- Additional Concerns
- Conclusion
Performance Implications of Security for Real-Time Systems

- Real-time systems must balance need for security with system constraints:
  - Mechanisms for security can not introduce lack of availability
  - Especially a concern for small embedded systems, in which case hardware acceleration may be a necessity
- Two major classes of cryptographic methods
  - Symmetric methods relatively efficient, especially with hardware acceleration
  - Asymmetric cryptographic methods are several orders of magnitude more expensive, but are needed for authentication and non-repudiation
- DTLS Transport uses both types of cryptographic algorithms
  - symmetric methods for encryption and message integrity (AES, SHA-1, etc.)
  - asymmetric for authentication and key establishment (RSA, DSA, DH)
Performance: Latency

Encryption of data transported on Gigabit Ethernet between moderate-performance Linux workstations (2GHz Opteron).
Performance: Throughput (1kB)

Encryption of data transported on Gigabit Ethernet between moderate-performance Linux workstations (2GHz Opteron).

- **UDPv4**: 500 Mb/s
- **DTLS Transport (AES128)**: 150 Mb/s
- **DTLS Transport (AES256)**: 120 Mb/s
- **DTLS Transport (IDEA-CBC)**: 90 Mb/s

1kB block size
Agenda

- Motivation & Requirements
- Net-centric approach to DDS Security
- Proposed DDS Security Model
- Pragmatic Implementation of Secure Domains
  - Mandatory Access Control at Transport Layer
  - Operating System Security Integration
  - Performance Implications for Real-Time
- Integrated DDS Security with RBAC
- Additional Concerns
- Conclusion
Integrated DDS Security

- Integrating an access control model into DDS can provide additional benefits
  - Finer-grained security than above mechanisms
  - Native DDS mechanisms for notification within application
  - DDS discovery can be used to bootstrap multicast security mechanisms

- Role-based access control is a good model for net-centric systems

- Plugin architecture for secure reference monitor can be used to avoid limiting decisions about access control model, extensibility
Configuring an Application for Secure DDS

Credential:
Name = Ariel_Rti
Public Key = ABCD1234...
Security Level = Secret
Roles = Engineer, Staff (signature)

- When creating participant, specify credentials and policy management plugin (in QoS)
- DDS checks credentials on entity creation
  - Return error value
- DDS checks credentials on discovery match
  - Event used to notify via listener or waitset
  - Notification may be sent to peer if allowed by permissions
Notifications

- Of local application of rules: fail to create
  - e.g. new return code: DDS_RETCODE_NOT_PERMITTED

- Of application of rules to discovered entities: events via listener or waitset
  - on_participant_matched()
  - onRejectedCredential(ID)
  - onAccessDenied(handle, topic)

- And existing events
  - on_subscription_matched()
  - on_publication_matched()
Role-based Access Control for Domains

- Access to join domains is mediated by the security plugin
  - Domains can be used to segregate security levels
- Roles can be used to limit:
  - Which users can join a domain
  - Who can create topics in the domain

**ROLES:**
Ariel → engineer, staff ; SECRET
Gerardo → admin, staff ; SECRET
Joe → user ; UNCLASSIFIED

<table>
<thead>
<tr>
<th>Domain ID</th>
<th>Security Level</th>
<th>Join Domain roles</th>
<th>Create Topic roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Unclassified</td>
<td>staff, user, engineer, admin</td>
<td>engineer, admin</td>
</tr>
<tr>
<td>4</td>
<td>Secret</td>
<td>staff, admin</td>
<td>admin</td>
</tr>
</tbody>
</table>
Role-based Access Control for Topics

- Access to topics within a domain can also be mediated by the security plugin
- Roles can be used to limit who can write/read individual topics

ROLES:
Ariel → engineer, staff ; SECRET
Gerardo → admin, staff ; SECRET
Joe → user ; UNCLASSIFIED

<table>
<thead>
<tr>
<th>Domain ID</th>
<th>Topic</th>
<th>Write roles</th>
<th>Read roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Square</td>
<td>engineer</td>
<td>staff, user</td>
</tr>
<tr>
<td>3</td>
<td>Circle</td>
<td>admin</td>
<td>staff, user</td>
</tr>
<tr>
<td>3</td>
<td>Triangle</td>
<td>staff</td>
<td>engineer</td>
</tr>
</tbody>
</table>
Role-based Access Control for QoS?

- Access to topics can be parameterized further
- Roles can be used to limit allowed levels of service to write/read individual topics

**ROLES:**
- Ariel → engineer, staff ; SECRET
- Gerardo → admin, staff ; SECRET
- Joe → user ; UNCLASSIFIED

<table>
<thead>
<tr>
<th>Domain ID</th>
<th>Topic</th>
<th>Write roles</th>
<th>Read roles</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Square</td>
<td>engineer</td>
<td>staff [RELIALBE], user [BE only]</td>
</tr>
<tr>
<td>3</td>
<td>Circle</td>
<td>admin</td>
<td>staff [KEEP ALL], user [KEEP LAST 1]</td>
</tr>
<tr>
<td>3</td>
<td>Triangle</td>
<td>staff</td>
<td>engineer</td>
</tr>
</tbody>
</table>


The security plugin can take additional information into account when making access decisions:

- QoS policies specified when creating entities
- Mandatory content filtering

```c
/* domain access control */
bool is_domain_join_permitted(credential, domain, participant_qos);

/* within-domain access control */
bool is_topic_create_permitted(credential, domain, topic, topic_qos);
bool is_read_permitted(credential, domain, topic, requested_qos, content_filter <inout>);
bool is_write_permitted(credential, domain, topic, offer_qos, content_filter <inout>);
```
DDS Protocol Updates for Security?

- In addition to integrated access control, protocol updates could provide:
  - Encapsulated key management protocol
  - Secure wrapper for DDS submessages

- Potential benefits
  - Variable level of security for different topics
    - Confidentiality and/or integrity and source authentication
  - Use of shared keys to support multicast security
Agenda

- Motivation & Requirements
- Net-centric approach to DDS Security
- Proposed DDS Security Model
- Pragmatic Implementation of Secure Domains
  - Mandatory Access Control at Transport Layer
  - Operating System Security Integration
  - Performance Implications for Real-Time
- Integrated DDS Security with RBAC
- Additional Concerns
- Conclusion
Additional Concern: Security Policy Management

- Complex systems require complex security policies
- Dynamic net-centric systems require dynamic security policies
- Authorization management for complex, dynamic net-centric systems needs to be agile for effective update when the system changes

Recommendation: Model-Based Security

Benefits
- Cost savings
- Enabled agility (growth, re-use)
- Improved security

Ulrich Lang, Ph.D.
Chief Executive Officer
ObjectSecurity LLC
ulrich.lang@objectsecurity.com
Solution: OpenPMF 2.0

Specify security requirements in abstract models

Generate enforceable security policies automatically

Protect & monitor applications automatically, e.g. for SOA
Solution: OpenPMF 2.0

Model-driven security management product
- Mainstream within 5 years (Gartner)

Benefits
- Align business and IT security
- Cost saving
- Enable IT agility (+ growth)
- Improved security

OpenPMF 2.0 for DDS
- Specify fine-grained, customizable authorization rules for DDS
- Governs information flows between DDS publishers and subscribers
- Related to XACML architecture, but with model-driven security support
- Next steps: Extend model-driven security management support for DDS

OpenPMF 2.0 Model-Driven Security Example

- Example: HIPAA healthcare regulatory requirement
- Security requirements mapped from EA layer of abstraction…
- …via fine-grained machine-enforceable authorization rules (OpenPMF PDL, XACML) …
- …to automatic run-time authorization enforcement of information flows (fine-grained, policy-driven, context-aware, content-aware etc.)

Example - Healthcare (HIPAA): “every doctor is only allowed to access the patient record of the patient they are currently treating, unless the patient is treated in a crisis context, or the patient consents etc. etc.”

Simplified rule excerpt: allow information flow if “caller X.509 cert. id doctor1” via “firewall IP…” calling “file patient1” on “database IP…” and from “hospital IP…” and “doctor1 is treating patient1” and/or “patient1 crisis”… (XACML or PDL):

Distributed runtime enforcement & monitoring (SOA, XACML, DDS, CORBA/CCM, …)
Additional Concern: Wide-Area Network (WAN) Support

- **Requirements**
  - Integrate DDS peers across more complex network configurations
  - Provide secure communication over WAN

- **Possible forms of solution**
  - WAN Transport architecture using NAT traversal protocols directly between peers
  - WAN Router architecture using statically configured proxy applications

- In both cases, access control mechanisms must be used to avoid introducing vulnerabilities
Additional Concern: OS Mechanisms for Multiple Levels of Security (MLS)

- Partitioned operating systems or hypervisors can provide multiple virtual machines each running at different security levels.
- If each partition can separately configure the allowed set of ports or IPC objects, DDS access can be segregated by security level.
- Partitioned CPU and memory resources can provide additional protection against denial-of-service due to a badly behaved component at a lower security level.
- These operating system mechanisms can be used to compartmentalize systems with more complex management of security policies.
Conclusion

- Security and Information Assurance is critical for net-centric systems as system scope expands
  - Systems getting more complex, more interconnected
  - More avenues of attack

- Today: Mechanisms exist for coarse levels of access control to critical data in DDS systems
  - Secure transport (DTLS)
  - Secure operating systems

- Future: Integrated DDS security will meet needs of systems as complexity increases
  - Fine-grained access control, scalable mechanisms
  - Capabilities to manage, control, and provide assurance