Adaptive and Evolving Security for Autonomic Messaging Middleware

Habtamu Abie, PhD
Senior Research Scientist
www.nr.no/~abie

Norwegian Computing Center
RISKnet Workshop, Oslo 02/04-2009
Presentation Outline

► Autonomic computing and message oriented middleware (MoMs)
  ▪ Motivation and Inspiration
  ▪ Best MoMs Limitations
  ▪ Stringent requirements

► Adaptive, autonomic and evolving security
  ▪ Special requirements for the adaptation
  ▪ Driving factors and needs

► GEMOM - Genetic Message Oriented Secure Middleware
  ▪ Advances
  ▪ Adaptive security
  ▪ Adaptive trust model
  ▪ Case Studies

► Potential Impacts and concluding remarks
Motivation: Complexity and Unmanageability

- **Division of Labor**
  - Developer develops application,
  - middleware manages execution

- **Application Management**
  - development of intelligent, open, and self-management systems

The Autonomic Nervous System Monitors and Regulates:
- Without requiring our conscious effort
- When we run, it increases our heart and breathing rate
Inspiration: Biological and ecosystem metaphors

► A complex adaptive system
  ▪ autonomic systems that mimic biological auto-immune systems
  ▪ behavior of an ecosystem

► Biological and ecological systems maintain integrity
  ▪ reacting to known changes
  ▪ adapting to unknown changes
  ▪ dying

► Adaptation
  ▪ macroscopic ecosystem level (e.g. system or species)
  ▪ microscopic biological level (e.g., molecular, cellular)
Inspiration: Dependable Rover

- Adaptive risk management
  - learns, adapts, prevents, identifies and responds to new or unknown threats in critical time
  - much like biological organisms adapt and respond to threats in their struggle for survival

- Thus, “the purpose of risk management is to change the future, not to explain the past” - Dan Borge

- Moral: Maximize the value of taking risk
Autonomic computing

► Autonomic computing
  ▪ proposes the development of intelligent, open, and self-management systems

► IBM defined four general properties a system should have to constitute self-management:
  ▪ self-configuring
  ▪ self-healing
  ▪ self-optimizing
  ▪ self-protecting

► Accompanied by four enabling properties or attributes
  ▪ self-awareness
  ▪ environment-awareness
  ▪ self-monitoring
  ▪ self-adjusting
Autonomic computing: Self-*

► Self-configuring capabilities
  ▪ enable the system to adapt to unpredictable conditions by automatically changing its configuration
  ▪ such as adding or removing new components or resources, or installing software changes without disrupting service

► Self-healing capabilities
  ▪ can prevent and recover from failure by automatically discovering, diagnosing, circumventing, and recovering from issues that might cause service disruptions

► Self-optimizing capabilities
  ▪ enable the system to continuously tune itself – proactively to improve on existing processes and reactively in response to environmental conditions

► Self-protecting capabilities
  ▪ can detect, identify, and defend against viruses, unauthorized access, and denial of service attacks
Message-oriented middleware (MoMs)

► MOMs
  ▪ increase the interoperability, portability, and flexibility of architectures
  ▪ typically asynchronous and peer-to-peer, but most implementation also support synchronous messaging
  ▪ reduce the number of point-to-point connections

► Best state of the art (SoA)
  ▪ existing commercial products
  ▪ OpenSource products
  ▪ current research activities
Best MoMs Limitations

- Scalable, resilient, and self-healing
  - hot standby brokers with instant switch-over and no data loss
  - no means to compensate for the reliability loss by automatically finding another source of redundancy
  - arbitrary resilience by a brute-force approach
  - self-healing is either rudimentary or non-existent

- Relatively prone to the incidence of feed failures
  - do not take redundant feeds into account

- Clustering topics into groups of one or more
  - but no group replication

- No holistic and systematic adaptive security, privacy and trust
  - No integration of threat and vulnerability management tool set and intelligent techniques to support vulnerability management assurance
Stringent requirements

- Modern MoM platforms have stringent requirements for
  - Resilient
    ◦ ability to keep going in given scenarios – by learning, evolving, etc., over time
  - Self-healing
    ◦ ability of the system to preserve its capabilities even in the event of failure of any individual or multiple components
  - Self-learning and optimization
    ◦ using genetic and evolutionary techniques
    ◦ creation of fuzzing procedures – inductive transfer where knowledge learned about some tasks is retained so as to efficiently learn a new task
  - Self-adapting
  - Evolving
  - Fault-tolerance
  - Self-active-vulnerability assessment
  - Adaptive, autonomic and evolving security
Adaptive, autonomic end evolving security

► Adaptive security refers to a security solution
  ▪ learns and adapts to changing environment at run-time in the face of changing threats
  ▪ anticipates threats before they are manifested

► Autonomic security refers to
  ▪ applying the idea of flexibility to the security space itself
  ▪ automating reconfiguration of the protection mechanisms
  ▪ resulting a self-protected system running (almost) without any user intervention

► Evolving security (long-term) refers to
  ▪ the modification of existing security functions and the generation of new functions
An adaptation is considered as a problem of optimal control of a specified object $F$:

- State $S$ of the object
- Its influence $Y$ on the environment depends on influences $X$ of the environment
- Set of adaptable factors $U$
- Goals $Z$ of the adaptive control are defined by specific constraints on the state of the object
AES involves

- Gathering contextual information both within the system and the environment
- Analyzing the collected information and making decisions with learning capabilities
- Responding to changes using the adaptation forms/methods
- Modifying existing functions/structures and generating new functions/structures
Adaptation forms

► Parameters adaptation
  ▪ the adaptation is achieved by specific variations of the control parameters vector

► Structure adaptation
  ▪ the adaptation is achieved by dynamic changes in the structure of the system

► Goal adaptation
  ▪ formal constraints on the state of the system

► In combination

► Emerging approach: cross-layer adaptation
  ▪ Vertical cooperation among multiple system layers
  ▪ Horizontal cooperation among multiple platforms
  ▪ Universal adaptation
    ◦ Combination of vertical and universal adaptation
Adaptation technologies

► Specifying adaptation behaviors
  ▪ statically specified using templates, classes or scripts
  ▪ dynamically specified in the form of code/scripts/rules

► Enforcing adaptation behaviors
  ▪ adaptation enforcement mechanisms in case of legacy applications
  ▪ adaptation behaviors can be integrated within new applications execution

► Conflict detection and resolution
  ▪ conflicts can occur between multiple adaptation behaviors and between adaptation behaviors and application execution
  ▪ system to detect and resolve conflicts that arise due to incompatibilities between configuration units (i.e. feature interactions) or due to the conflicting nature of the objectives they achieve
  ▪ runtime conflicts resolution has been achieved using microeconomic techniques (production and pricing analysis) or legal reasoning (logic, analysis, argumentation, hermeneutics)
Legal reasoning & microeconomic techniques

► Legal reasoning methods: Four methods used in legal practice, legal dogmatics and legal theory:
  ▪ logic
  ▪ analysis
  ▪ argumentation
  ▪ hermeneutics (interpretation theory)

► Microeconomic Techniques
  ▪ Production analysis - microeconomic techniques are used to analyze production efficiency, optimum factor allocation, costs, economies of scale and to estimate the firm's cost function
  ▪ Pricing analysis - microeconomic techniques are used to analyze various pricing decisions including transfer pricing, joint product pricing, price discrimination, price elasticity estimations, and choosing the optimum pricing method
Special requirements for the adaptation

- Algorithm must respond on the changes of the system on-the-fly
- Algorithm functioning should cause
  - minimal deviations of operations of the system from the normal mode
- Adaptation should address
  - reconfiguration of functional logic
  - architecture-wide reconfiguration
  - conflict-handling
Driving factors and needs

► Driving factors
  ▪ convergence of advanced electronic technologies (wireless, handheld, sensors, etc) and the Internet.
  ▪ promises anywhere, anytime access to data and computing
  ▪ changing nature and behavior of the environment
  ▪ systems to operate through failures and attacks

► Needs for dynamic adaptation
  ▪ heterogeneity of hardware, network, software.
  ▪ dynamics of the environmental conditions, especially at the wireless edge of the Internet
  ▪ limited resources (such as battery lifetime).

► Software adaptation technologies for:
  ▪ detecting and responding to environmental changes
  ▪ strengthening self-auditing capabilities of “always-on” systems
Adaptive security systems recently developed for different reasons

- Context Aware and Adaptive Security for Wireless Networks
- Adaptive Security in Complex Information Systems
- Adaptable Security Manager for Real-Time Transactions
- Dynamic Authentication for High-Performance Networked Applications
- Intelligent Adaptive Firewall Architecture
- Threat-adaptive security policy
- Adaptive Trust Negotiation and Access Control
- Adaptive Security Policies Enforced by Software Dynamic Translation

- A Survey of approaches to adaptive application security [Elkhodary]
Adaptive middleware technologies

► ACT (Adaptive CORBA Template) [Samimi]
  ▪ enables run-time improvements to CORBA applications in response to unanticipated changes in either their functional requirements or their execution environments

► RAPIDware [Samimi]
  ▪ focuses on adaptive middleware technologies

► MetaSockets
  ▪ provide low-level, adaptable communication
  ▪ example of host-infrastructure middleware

► TRAP (Transparent Reflective Aspect Programming)
  ▪ generator that enables adaptive behavior to be woven into existing programs
  ▪ example: Existing Java applications can be “upgraded” to use MetaSockets

► AUTONOMIA – An Autonomic Computing Environment [AUTONOMIA]

► Taxonomies of adaptive and reconfigurable systems [McKinley] [Sadjadi].
GEMOM – Genetic Message Oriented Secure Middleware

► EU FP7 ICT project grant agreement: 215327
► Research and develop a messaging platform that is resilient, evolutionary, self-organizing, self-healing, scalable and secure
► Focus
  ▪ reliability of message sourcing and delivery
  ▪ scalability with respect to message volumes
  ▪ replicating structural and dynamic properties of security metrics, policies, etc.
  ▪ pre-emptive vulnerability testing and updating
  ▪ holistic and systematic adaptive security, security measurement, monitoring, management and maintenance incrementally
GEMOM advances

► Resilient and self-healing
► Vulnerability management assurance
► Interoperability and integration of distributed systems
► Holistic and systematic adaptive security
  ▪ Monitoring, measurement, management and maintenance incrementally
► Scalability
  ▪ cooperating brokers, publishers and subscribers with sufficient replication of paths and messages
  ▪ Clustering topics into groups of one or more with group replication
GEMOM adaptive overlay nodes

Managerial layer

- RM
- ASM
- QM
- BM

Operational layer

- P
- B
- S

Overlay network of G-Nodes

Biological and ecosystem metaphors

- A complex adaptive system
  - autonomic systems that mimic biological auto-immune systems
  - behavior of an ecosystem

- Biological and ecological systems maintain integrity
  - reacting to known changes
  - adapting to unknown changes, or
  - dying

- Adaptation
  - macroscopic ecosystem level (e.g. system or species). Managerial level in this case
  - Microscopic biological level (e.g., molecular, cellular). Operational level in this case
• While each component implements local adaptation control loop, ASM implements global adaptation control loop
• Sensors are Anomaly Detectors, Security Monitors, Fault Detectors, QoS Monitors, Audit and Logging, etc.
GEMOM adaptive trust model

Compromised-based Trust Model
- Trustworthiness – Measuring the degree of confidence
  - Analysis and Categorization of Compromises
  - Interpretations of attacks and anomaly behavior
  - Monitor

Security-based Trust Model
- Publisher Level
- Broker Nodes Level
- Subscriber Level
- Communication Level

Adapt
- Analyze
- Adapt
Development of security metrics*

Requirements form the basis for measuring!

Decomposition of requirements

GEMOM Case Studies

► Collaborative business portal
  ▪ A portal intended as a generic platform to facilitate collaborative working between professionals in a local government setting with the range of professionals, e.g., the emergency services co-ordination business portal

► Dynamic linked exchange
  ▪ Exchange intended to match the procurement needs to available suppliers and specializing in local government / SME actors.

► Financial market data delivery
  ▪ Service to deliver trading signals for a range of financial markets to private and institutional investors

► Dynamic road management system
  ▪ A complex operational system for distributing road network traffic and mobility information to a wide range of potential consumers

► Banking Scenario: Money transfers
  ▪ A "Universal Banking HUB" in a central architectural position as a pervasive pivoting component of the bank's IT architecture
  ▪ Should be able to exchange, both internally and externally, several types of messages, each representing a specific kind of business fact
Potential Impacts [GEMOM], [Samimi]

- Robust group communication among users with disparate devices and networks
- Secure Self-healing systems to support mission-critical communication and computation under highly dynamic environmental conditions.
- Self-auditing systems that can report state inconsistencies, incorrect or improper use of components.
- Dynamic secure resource allocation in devices limited by battery-lifetime, bandwidth, or compute power
- Systematic secure evolution of legacy software to accommodate new technologies and adapt to new environments
- Enable systems to operate through failures and attacks
Concluding remarks

► Adaptation in Security Context
  ▪ Advantages
    ◦ contribution to the real-world information security with fuzzy definition and uncertain conditions
    ◦ access to the methods and tools from the Control Theory for the needs of the adaptation
    ◦ all the impacts above
  ▪ Disadvantages
    ◦ effectiveness is very dependant on the correct definition of security goals
    ◦ the additional resources required for the adaptation processes
    ◦ difficult to assure minimal deviations of operations of the system from the normal mode

► Increased
  ▪ resilience, scalability and support for vulnerability assurance

► Reaping the benefits of autonomic MoMs
  ▪ reliability via self-healing
  ▪ performance via self-adaptation
  ▪ security via self-protection (monitoring and vulnerability discovery
References

- GEMOM: http://www.gemom.eu/
Adaptive and Evolving Security for Autonomic Messaging Middleware

Habtamu Abie, PhD
Senior Research Scientist
Email: habtamu.abie@nr.no
Website: www.nr.no/~abie

Norwegian Computing Center
RISKnet Workshop, Oslo 02/04-2009

The End!