GEMOM - Significant and Measurable Progress beyond the State of the Art

Habtamu Abie
Norwegian Computing Center, Oslo, Norway
Habtamu.Abie@nr.no

Ilesh Dattani, Milan Novkovic
Q-Sphere Ltd, London, UK
ilesh@q-sphere.com, milan@q-sphere.com

John Bigham
Queen Mary, University of London, London, UK
john.bigham@elec.qmul.ac.uk

Shaun Topham
Diginus, Peterborough, UK
shaun.topham@ukonline.co.uk

Reijo Savola
VTT Technical Research Centre of Finland, Oulu, Finland
Reijo.Savola@vtt.fi

Abstract

GEMOM (Genetic Message Oriented Secure Middleware) is an EU FP7 ICT project that focuses on the significant and measurable increase in the end-to-end intelligence, security and resilience of complex, distributed information systems. Complex, distributed software systems are virtually impossible to implement without heavy use of messaging infrastructure. While the existing state of the art achieves arbitrary resilience by a brute-force approach, self-healing is either rudimentary or non-existent. GEMOM, with its fluid, resilient, self-healing and adaptive messaging allows for flexible messaging solutions and offers monitoring, management and maintenance incrementally. Its adaptive security solution will learn and adapt to changing environments during run-time in the face of changing threats without sacrificing the efficiency, flexibility, reliability and security of the system. This paper describes the significant and measurable progress beyond state of the art envisioned by the GEMOM project.

1 Introduction: state of art

Message Oriented Middleware (MOM) increases the interoperability, portability, and flexibility of architectures by enabling applications to exchange messages with other programs without having to know what platform or processor the other application resides on within the network [11, 10, 15]. MOM is typically asynchronous and peer-to-peer, but most implementations support synchronous message passing as well. MOMs provide a service that allows the content provider and consumers to concentrate on the production and consumption of the transmitted information. MOMs are compatible with the Enterprise Service Bus (ESB) [5] architecture where the MOM message broker acts as the bus between applications. A significant advantage of the architecture is that it reduces the number of point-to-point connections required to allow applications to communicate.

Complex, multi-tier distributed software systems are virtually impossible to implement without heavy use of messaging infrastructure. The most common variant of these systems is the Publish/Subscribe scheme. Synchronous Request/Reply is easily overlaid on top of Publish/Subscribe making Publish/Subscribe the right proxy for overall messaging.

The existing state of the art achieves arbitrary resilience by a brute-force approach. Self-healing is either rudimentary or non-existent. GEMOM (Genetic Message Oriented Secure Middleware) [6] is an EU FP7 ICT project that will make advances in resilience, scalability and support for vulnerability assurance. Emerging commoditisation, convergence and the omni-presence of a broad variety of information and communication technologies opens up the way for more efficient and faster uptake of the Service Oriented Architecture (SOA) concepts. SOAs advocate information systems as compositions of seamlessly integrated services.

With respect to the current MOM [14, 8, 22, 23, 21] solutions, peer-to-peer and similar architectures need some kind of wrapper to bring all the different software systems together. MOMs provide the means to link the processes and systems and all the communication protocols, disparate software, platforms, etc., for all of these are then not so important as MOM can take care of that. In essence, MOM can make a distributed system environment or architecture...
behave like a virtual machine [26]. From this starting point, GEMOM will move further to work on the critical parts of resilience, robustness and security.

2 Resilience and self-healing

The best available systems claim to be scalable and resilient as they employ hot standby brokers with instant switch-over and no data loss. However, once switch-over is performed these systems have no means to compensate for the reliability loss by automatically finding another source of redundancy. Also, they are relatively prone to the incidence of feed failures as they often do not take redundant feeds into account. Hence there is a high maintenance requirement. Resilience is essentially by a brute-force approach. Self-healing is either rudimentary or non-existent, and where it requires skilled management. Functional and security integrity and resilience of complex information systems that are assembled as composition of a multitude of distributed services critically depends on the integrity, resilience, scalability and topological elasticity of messaging. Current systems are not able to guarantee holistic and systematic security, privacy and trust management either. GEMOM offers all these key qualities.

GEMOM is able to handle redundant message feeds, where needed, and redundant delivery paths. In the event of failure switch-over to redundant resource would be effectively instantaneous and with no information loss. In addition to entire message broker redundancy GEMOM offers redundancy of certain subsets or messaging segments. As part of its self-healing when redundant components are switched to and used, GEMOM finds and primes other nodes, feeds or paths as new redundant components.

The GEMOM resilience mechanisms are novel and sophisticated and put GEMOM ahead of current MOMs as they are now. Ongoing broadening of overall networking and further uptake of Web Services and GRID concepts has the potential to leverage GEMOM’s resilience features. As part of resilience and self-healing, GEMOM’s holistic and systematic security, privacy and trust management is beyond the state of the art.

Because MOMs are naturally placed at the interfaces between distributed systems, they often correspond to trust boundaries between the agents. For example the feeds can be of different trustworthiness, the processing channels and transformations in the MOM can introduce vulnerabilities and errors, and even normally reliable sources of data may have been compromised. Hence GEMOM supports the detection of vulnerabilities and bugs.

GEMOM looks at a number of possibilities in connection with the self-learning capabilities and optimisation approaches with respect to resilience. In that sense, the algorithmic approaches for this will involve the use of genetic and evolutionary techniques at some level as partial elements for the overall solution.

3 Management of security vulnerabilities and misconfiguration vulnerabilities

Detection of security vulnerabilities, input errors, misconfiguration error management and bug detection support will be integrated into the GEMOM system. Large software systems depend on many software components from different providers. When the components are installed, the implicit assumption is that they are benign. However, simple bugs in a component can expose the system to malicious code. Also when components are interfaced to each other, for example to form new services, there can be unexpected interactions, emergent properties and vulnerabilities introduced as the trust boundaries are crossed.

The need to handle malicious or unexpected input has led to an increased interest in robustness testing, where exceptional input is generated, either manually, randomly or semi-automatically in an attempt to detect vulnerabilities prior to their exploitation. In the search for vulnerabilities and the creation of a trusted network needs different kinds of knowledge to be used. Areas where GEMOM significantly extends the state of the art principally rely on:

- the development of intelligent techniques to support the search and discovery of vulnerabilities, violations of quality of service (QoS) and privacy policy, and other errors
- the development of the GEMOM threat and vulnerability management tool set for threat discovery, techniques to support generic, intelligent, adaptive approaches to robustness and security testing in a distributed environment, i.e., knowledge of the different kinds of vulnerabilities, knowledge about the software function (the transformations and the subscriber behaviour) and aspects of the semantics of the application domain, and knowledge about protocols used will be integrated into the development of the tools to find vulnerabilities and errors in the software that runs complex distributed systems using the GEMOM platform.

The fuzzing techniques that GEMOM will utilize will also advance the state of the art. Because of its ability to produce results, fuzzing [3] has a considerable potential in the area of security and privacy. Standard testing techniques (there are many variants of testing) focus on the correctness of software, while fuzzing concentrates on finding exploitable bugs and on breaking the code.

Formal software engineering methods and testing are best at, e.g., checking if the implementation is correct with
respect to a given set of outputs results, it is now becoming accepted as a legitimate branch of software testing and as a method for specification. Vulnerability is hard to define and can for example be exploited by unanticipated use of functionally correct software on a software infrastructure. Cross-site scripting is an example, where input from the client is echoed by the server to the client browser as part of an error handling process and interpreted by the browser. Fault injection/fuzzing is being increasingly used and gaining acceptance. Present techniques exploit function call graphs and crawling mechanisms, and have difficulty in handling cases where new threads are started [3].

In the design of a fuzzer, key concepts are vulnerability and code coverage, the exposed interfaces (attack surface), and trust boundaries. An example of the latter is where data is sent across a network from different sources, through the MOM to an application. MOMs are good points to analyse the deployed system for vulnerabilities and the linking of a security toolkit to the GEMOM platform is novel and beyond state of the art. It builds on the state of the art and concentrates on developing new techniques to extend model assisted fuzzing in the context of MOM and also exploiting semantic models, and new approaches to learning, e.g., inductive transfer where knowledge learned about some tasks is retained so as to efficiently learn a new task, will be adapted and applied to the creation of fuzzing procedures.

In GEMOM, there is an increasing need to be able to describe and reason about security requirements at the semantic level, in this case to automate the testing for security vulnerabilities, multiple and transient errors and configuration problems. In this context, it is not being suggested that fuzzing is a replacement for a proper formal and secure software development process such as those created by several companies, but as a very powerful tool for finding threats in developing and deployed systems, and providing a mechanism for continual security assurance against new threats [1].

In order to provide adaptive application-level inter-domain security, the GEMOM security toolset will also investigate the use of domain relationships to provide support for reasoning about the validity of data. These relationships may involve parameters that need to be established by training. GEMOM can add to the tools for decoupling of security from core services, and providing context-dependent adjustment.

4 Achieving interoperability and integration of information and communication systems

We see the Publish/Subscribe variant of MOM infrastructure (PS MOM) is the most efficient way to integrate medium to high complexity distributed systems. This messaging possesses a few key properties offering resource efficient means to system modelling: (1) PS MOM system could be modelled and re-factored with ease during run time as well as at design time, exchange of messages is connectionless and asynchronous, (2) system around PS MOM is inherently extensible, and (3) PS MOM is a powerful base for resource efficient implementation of scalability and resilience.

GEMOM further supports better interoperability and integration of information systems by allowing actual instances to be configured so various functions are subcontracted to one or more separated, external or federated entities. This separation allows for a different security layout of different individual, or clusters of, services. The most important advantages of this approach are as follows:

Focus of different functional clusters on different issues and/or core competences. For example, dynamism of messaging is often such that in terms of economy of offered solution it is common to have pure messaging and authentication and security related services separated. Pure messaging in highly scaleable environments could be very resource intensive. For example, Investment Bank trading floors sometimes witness an average of 100+ messages per second delivered to 1000+ users. At the same time the solution has to be highly resilient with a high degree of fault-tolerance. The solution often results in a farm of higher-end servers working, sometimes (and due to statistical properties of the underlying process, e.g., market liquidity), dangerously close to peaks of their performances.

Pure messaging here is expensive and susceptible to additional load. If all sensitive messages are encrypted with strong keys and as publishes and subscribers are on the other side of the scalability boundary of pure messaging it is often not needed to burden this message brokering business with additional elaborate security. For example, if it becomes possible to make an unauthenticated subscription on encrypted data, all that a subscriber gets is encrypted data with no means to decrypt it. Message brokering is not compromised while an incident is flagged on one or more security monitors awaiting resolution. If anyone is to take a message like this for potential decryption, if it is feasible at all, the chances are that by the time this task is the completed message will have lost its importance.

By separating security, privacy and trust services (e.g., authentication, authorization, key management, security monitoring, security metrics processing, etc.) as they are far less resource intensive, it is possible to implement them to far higher standards. The GEMOM security system is also primed for uptake of Federated Identity Management.
5 Optimizing security and protection of networked system

By being ultra-resilient, self-healing and self-optimising GEMOM leverages two key "commodities": (1) mainstream, commodities security features that come with standard machinery, operating systems, dedicated software and hardware, etc., and (2) abstract and so extensible notion of a fault. By being able to extend the meaning of what a fault is and measure it, management actors can be added to GEMOM enabled systems that could instantly cut off faulty or compromised actors.

The explosive growth of networking supported by the deployments of SOA, GRID and Utility Computing is changing the perception as to what a computer is and even what a simple application is. We are entering an era where A network becomes the computer and the network is becoming the entity that executes even simple software tasks, not individual, known machines. GEMOM, with its fluid, resilient and adaptive messaging is well placed to fully support this paradigm shift in the evolution of IT computing. The catalyst effect of having readily available services are likely to prompt more creativity as it would be easier to reuse functionality and so assemble new software systems. This easy access to a variety of services promises major qualitative and quantitative advancements in the evolution of information systems.

An adaptive security approach is necessary for GEMOM which has requirements for self-healing, adapting, evolving, fault-tolerance, security, self-active-vulnerability assessment, especially when the internal working model of the system and the environmental influences on the system are uncertain during run-time. Adaptive security in GEMOM refers to the GEMOM security solution that learns and adapts to changing environment at run-time in the face of changing threats [1, 25] without sacrificing the efficiency, flexibility, reliability and security of the GEMOM system. This involves gathering contextual information both within the system and the environment, analysing the collected information and responding to changes by adjusting internal working parameters such as selecting suitable encryption schemes, security protocols, security policies, security algorithms, different authentication and authorization mechanisms, etc. For quantitative assessment of the efficiency of the adaptation of GEMOM security the following information will be used: “throughput fluctuations in the message channels, processing power loading, time spent on the specific security functions, and quality of service of the system pertaining to metrics such as delays, denials, failures, etc.” [19].

A number of adaptive security systems have been developed recently supporting adaptation at different levels (from hardware-level to application-level) and for a number of reasons. That is, Context Aware and Adaptive Security for Wireless Networks [7], Adaptive Security in Complex Information Systems [19], Adaptable Security Manager for Real-Time Transactions [20], Dynamic Authentication for High-Performance Networked Applications [18], Intelligent Adaptive Firewall Architecture [27], threat-adaptive security policy [24], Self-contained object for secure information distribution systems [2], Adaptive Trust Negotiation and Access Control [16], Adaptive Security Policies Enforced by Software Dynamic Translation [9], and Security Architecture and Adaptive Security [25]. Several taxonomies have been introduced for classifying adaptive and reconfigurable systems [13, 12, 17]. A Survey of approaches to adaptive application security, and adaptive middleware can also be found in [4] and [17], respectively.

GEMOM is developing an adaptive security and QoS model that will consist of a continuous cycle of monitoring, assessment, and evolution to meet the challenges of the changing, multifaceted relationships within and between organizations in MOM-based business environments and today’s rising threat situation. It will include the integration of security monitoring, analysis and response functions and tool-set, security metrics, adaptive authentication/federated identity management trust system, adaptive authorization, adaptive QoS functionalities, adaptive security knowledge repository, tools and processes for pre-emptive vulnerability testing and vulnerability updating which are a corner-stone component of GEMOM security model. It will be deployed and managed as an integrated GEMOM infrastructure, not as a series of separately considered solutions. This integrated infrastructure approach and the tool-set which will be developed are important innovators for the project and are beyond state of the art.

Figure 1 depicts this proposed adaptive security model for GEMOM. The Adaptive Authentication/Identity Management allows the GEMOM system to choose among different authentication and identification methods based on runtime context specific parameters, such as system threat.

Figure 1. GEMOM adaptive security model.
or user-threat level. The authentication method may change at any point according to the values of the parameter of interest. The Adaptive Authorization allows access control policies to change based on context specific parameters. The parameters can be system-wide, user-specific, or both. The Adaptive Collectors sense and gather contextual information both within the system and the environment and distribute information about the security environment to the Adaptive Analyser and Adaptive Database. The Adaptive Analyzer processes collected data, along with other information (e.g., security policy, threat levels, or broker trust levels boundaries) and occasionally proposes actions to bring about a new stage. The Adaptive Database contains security knowledge repository, metrics, reputation information and suitable algorithms for using metrics and approaches to learning, e.g., inductive transfer where knowledge learned about some tasks is retained so as to efficiently learn a new task.

6 Benchmarking

Much of the GEMOM work will itself generate benchmarking data in terms of improvements of performance. For example, to be able to perform qualitative and quantitative analysis of various pathways in terms of volumes, resilience and vulnerabilities, and to maintain a variety of usage related information and generate views on that information so to instruct agents to probe new pathways and re-factor the system so it evolves as more efficient and resilient.

Further, valuable and usable information will be developed in terms of the user needs work, which is dedicated to the assessment and analysis of GEMOM requirements, determination of the success criteria for the commercial application and user acceptance of the services, technical design and specification of both the various components of the platform and case-study applications, and plan the user validation and test phase.

In addition to the valuable information resulting from these activities integral to carrying out the project, another set of benchmarks will be specifically developed at the beginning of the project Benchmark Development. Since qualities of service and security solutions strongly affect end-users, this task will study the implications of quality of service and security solutions on applications and end users. The technical solutions will be mapped into scenarios of usage. The scenarios will be evaluated with users, providing further indicators. Experiments will be conducted using the scenarios and benchmarks.

The overall result will be the development of a comprehensive set of benchmarks which will enable the progress of GEMOM to be monitored, starting from what is currently regarded as the state of the art and tracking progress as the boundaries are pushed forward. These will be both quantitative, arising from much of the empirical work, and qualitative, based amongst other things, upon the user needs interviews.

7 Forward-looking summary

This significant and measureable progress beyond state of the art will have impact on the GEMOM system architecture. The system architecture will be made up of a set of communicating nodes. Some of these nodes will be operational nodes and some managerial. The operational nodes can be classified as producer, consumer and broker nodes with their associated agents such as sensors, effectors, monitors, analysers, etc. An operational node contains the primary functionality of the node (e.g., brokerage) and sensors to assess the performance of the functionality (e.g., delay of attack probes) and effectors that can change the behaviour of the primary functionality. The sensors and effectors communicate with management nodes of different types. One management node could support network QoS, one could support security anomaly management (the anomaly detectors would be in the operational nodes), but there can be many and they can be layered. The management nodes make decisions about the run time operation of the system that require a wider perspective than the individual operational nodes.

The overall GEMOM system architecture has a structure similar to that of an adaptive system that utilises autonomic systems that mimic biological auto-immune systems at the microscopic level (operational level in this case) and that utilizes the behaviours of an ecosystem of disparate entities at the macroscopic level (managerial level in this case) [4, 25]. Hence we can speak of GEMOM having a genetic makeup.

The advances which GEMOM proposes to make in the area of messaging revolve around the notion of a fault. In addition to the intuitive understanding of what a fault might be, whereby an actor stops being operable, connection is lost, etc., GEMOM extends the notion of fault to include compromised security or inadequate bandwidth availability in the first iteration and compromised abstract notion of resource in its final iteration. The combination and integration of MOM and agent based systems resulting in resilient MOM is beyond state of the art.

Finally, the GEMOM security solution learns and adapts to changing environment at run-time in the face of changing threats without sacrificing the efficiency, flexibility, reliability and security of the GEMOM system. It is based in part on complex adaptive systems and integrates various tool-sets for monitoring, measuring, pre-emptive vulnerability testing and vulnerability updating, etc. This integrated infrastructure approach and the tool-set are important innovators for GEMOM and are beyond state of the art. They will
allow both the parametrical and structural adaption of the security solutions by utilizing the variations of the control parameters, and the quantitative and qualitative parameters of the system at run-time.

Acknowledgement. This paper has been compiled from the technical annex of the GEMOM contract [6] with the European Commission. The authors acknowledge the contributions to the technical annex made by the members of the GEMOM consortium.

References


