ISSUE
09
JUNE
2011

FOUR-MONTHLY
NEWSLETTER OF

IEEE TCSIM

this issue

TCSIM Mailing list

In order to facilitate communication and information sharing with TC members, a new TCSIM mailing list has been created using the IEEE Listserv system.

To send a message to the new TCSIM list, just send mail to TCSIM@LISTSERV.IEEE.ORG

If you are not currently subscribed to the list please send a message to Prof. Kaushik Chowdhury at krc@ece.neu.edu who is currently managing all the subscriptions to the new TCSIM list. You can also search for the TCSIM list at http://listserv.ieee.org/

We hope you can make use of this new list to share simulation related information with the TCSIM community.

Chair's Message by Dr. Dave Cavalcanti

Dear TCSIM colleagues,

welcome to the 2011 Q2 issue of the TCSIM newsletter.

This issue includes two very interesting articles selected by the editorial team. The first article by Romanel discusses a framework for simulating the evolution of biological networks. The second article from our colleagues El-Said and Woodring looks at security and privacy challenges in RFID systems. The authors propose an authentication framework for the tag cloning problem. I'm also glad to announce that the TCSIM leadership team will organize a special issue of the IEEE Computer Magazine on "Modeling and Simulation of Smart and Green Computing Systems". You can see the CFP at the end of this issue. We're very excited with this opportunity and looking forward to receiving your contributions to make this special issue a success.

Enjoy the reading!

Editorial Team

Dr. Mostafa El-Said, Editor-in-Chief
Dr. Kaushik Chowdhury, Editor
Dr. Chittabrata Ghosh, Editor
Dr. Tommaso Mazza, Editor

Evolving Programs to Simulate Molecular Evolution

Alessandro Romanel

romanel@ens.fr

Équipe sémantique et interprétation abstraite
Laboratoire d'informatique de l'École normale supérieure
45, rue d'Ulm, 75230 Paris Cedex 05, France.

Abstract

We present a formal approach to study the evolution of biological networks.

We use the BlenX language and its stochastic simulator to model and simulate networks in connection with evolutionary algorithms. Variability is obtained by mutating the structure of BlenX programs and networks are selected at any generation by using a fitness function.

1 INTRODUCTION

Over the last years an increasing interest in using evolutionary approaches to study biological networks has continuously grown [1-4]. Understanding how networks emerged during evolution can help us to understand their basic properties, such as the role of complexity and the importance of topology and feedback loops. In [5] we developed a framework to allow the study of networks evolution, which is based on BlenX [6] and the Beta Workbench (for short BetaWB) [7]. We proposed a framework for simulating the evolution of biological networks where evolution proceeds through selection acting on the variance generated by random mutation events, and individuals replicate in proportion to their performance, referred to as fitness. This follows the idea that to simulate evolution by natural selection, we must be able to express populations of individuals, variability and fitness. The work presented in [5] and here summarized was done when the author was at the Microsoft Research – University of Trento COSBI. A similar approach has been later presented in [11].

2. THE FRAMEWORK

The Language: BlenX [6,8] is a language based on the process calculi and rule-based paradigms and is explicitly designed to model the interactions of biological entities such as proteins and other biochemical species. It is a stochastic language in the sense that the probability and speed of the interactions are specified in the programs that are written in this language. In this respect, the BetaWB [7] is a tool that provides a network-free stochastic simulator for BlenX, based on an efficient variant of the Gillespie algorithm [9]. We refer the reader to [6,8] for a detailed presentation of the BlenX language. We consider a protein as a biological entity composed by a set of sensing domains, a set of effecting domains and an internal structure. Sensing domains are the places where the protein receives signals, effecting domains are the places that a protein uses for propagating signals, and the internal structure codifies for the mechanism that transforms an input signal into a protein conformational change, which can result in the activation or deactivation of other domains. Each biological entity is modelled in BlenX as a box, which is a computational entity composed by a set of interfaces and an internal program (see Fig. 1a). Interfaces have associated sorts and are the places where a box can communicate through message-passing with other boxes. Sensing and effecting domains are represented by interfaces and their structures as sorts. The mechanism of message-passing is used to implement the interaction of proteins: sensing domains receive activation (e.g. phosphorylation) and deactivation messages sent to the protein that are then computed by the internal program to change the structure of other protein domains; effecting domains are used instead to send messages to other proteins and so to activate or inhibit them. The exchange of messages can happen between boxes whose interfaces sorts have a certain degree of affinity (described by a relation denoted as α) which codes the strength of their interaction capability. These affinities are calculated by definable expressions, which can be declared as simple real numbers if the interactions that they are accounting for follow elementary mass action law, or they can be arbitrary functions if the interactions are not elementary. In our evolutionary framework a system specified by a BlenX program represents and individual (see Fig. 1b) while a population consists of a set of different BlenX programs.

The genetic algorithm: Evolution proceeds through selection acting on the variance generated by random mutation events. Individuals replicate in proportion to their performance, referred to as fitness. This process is modelled through the Func. 1.

This algorithm slightly differs from the generic evolutionary algorithms used in computer science, being closer to real biological observations made for the asexual reproduction of organisms. Each individual in the population is codified using a BlenX program, and the boxes in each program are the abstraction of all the entities present in that individual. The interaction among these entities result in the behaviour of the network we want to study. There are four main procedures in the algorithm:

- *GenerateInitialPopulation*: the initial population can be generated randomly, from a predefined network configuration to be used as a starting point, or it can be a network with no interactions. All the individuals in the initial population can be equal at the beginning, as they will be differentiated later by the mutation phase;
- *Simulate*: each individual in the population is simulated separately using the BetaWB stochastic simulator;
- *ComputeFitness*: the output of the simulation is used to compute the fitness value of the current individual. Note that the fitness value is problem-dependent;
- *Replicate&Mutate*: this is the most important part of the algorithm. Like in a real environment, individuals with the highest fitness values are more likely to survive, repli-

cate and produce a progeny that resembles them, being not, however, completely equal to them.

The *Replicate&Mutate* function creates a new population with the same number of individuals of the current generation, using as a base the current individuals. At each step it chooses one individual, with probability proportional to its fitness. This is achieved by constructing a cumulative probability array a from the fitness array, generating a random number in the range 0...a[PopulationSize], and then finding the index into which the random number falls. The selected individual will replicate and pass to the next generation. During the replication, to each protein in the genome of the individual is given the chance to mutate, according to a probability. A mutation is selected among all the possible types by the *GetRandomMutation* function, and this mutation is applied. Finally the individual, which can be either equal to its predecessor or mutated, is added to the new population.

Mutations: Mutations affect the network dynamics. For example, mutations in a DNA sequence can change the protein amino-acid sequence, leading to changes in its tertiary structure with implications on the affinity of this protein with other proteins or substrates. Similarly, events at DNA level as gene duplication or domain shuffling can alter network structure and dynamics. A computer program which is used to mimic evolution must implement random mutations in individuals during the replication as well. Given an individual (see Fig. 2a), we considered the following types of mutations:

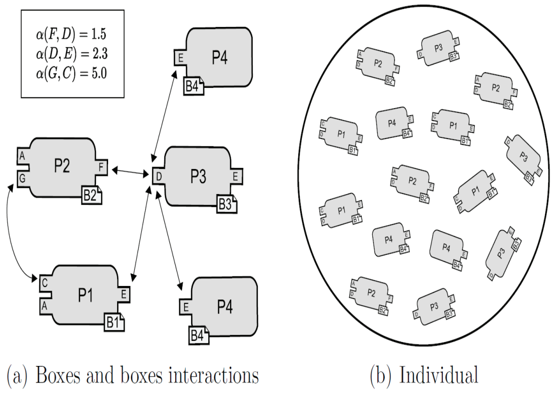
• *Duplication and deletion of proteins*: Gene duplication at DNA level is implemented with a duplication of the box associated with the protein the gene codifies for. The new box will have a similar internal program and the same interfaces, while interface sorts will be new but will have the same interaction capabilities (Fig. 2b).
• *Mutation of domains*: Point mutations in DNA can change the protein amino-acid sequence, and consequently lead to the mutation of a domain and to changes in the interaction capabilities of the protein to which it belongs. In our formalism, this is achieved by changing the α relation on the two interfaces that take part in the interaction (Fig. 2c).
• *Duplication and deletion of domains*: Domain duplication or deletion is more complex as it involves not only interfaces or rates, but requires also modification of the internal program in response to stimuli. Duplicating or removing domain can be easily done acting on the interfaces and on the sorts; however, for these domains to act as sensing or effecting domains in cooperation or in antagonism with the existing ones, the internal program must also be changed. We devised several possible modifications of the behaviour when a domain is added. These mutations are obtained by manipulating the structure of the internal programs (Fig. 2d). We assume that the internal programs have a standard structure, so that transformations can be standardize as well. Examples of standard internal programs are detailed presented in [5].

Measure of fitness: When analyzing evolution of specific biological systems, one need to consider the fitness benefit of that system to the organism (i.e. to its reproductive success). While it is usually complicated to define and measure such fitness contribution, network dynamics can provide a good proxy in case of biological networks. As the concentrations of the entities involved in such networks will define the proper functioning of the network, how these concentrations fit a specific time course would determine how well the network operates. We include some common operations that can be performed on concentration traces, and a way of finding entities based on their characteristics, such as the number and binder identifiers, or their state. This is important in a language like BlenX where the whole system, and all the entities that can appear in a simulation, are not specified in the program but can be generated dynamically during the simulation.

Constraints: We understand that with our framework it is possible to generate countless combinations, interactions and mutations. Many interactions or mutation can be possible and make sense from the point of view of a program syntax and semantics, but have little or no sense from the biological perspective. We addressed this issue by providing a configurable way of specifying constraints on mutations, their probability and which class, or type, of protein or domain they can affect.

3. CASE STUDY

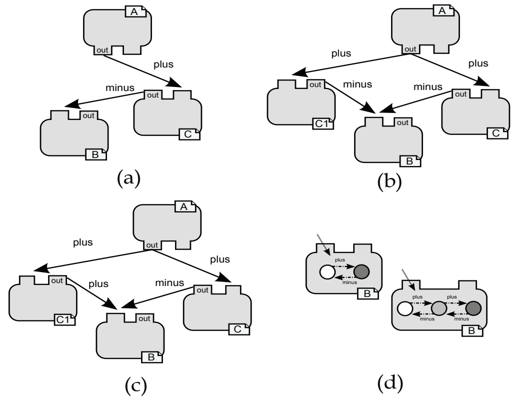
The mitogen-activated protein kinase cascade (MAPK cascade) is a series of three protein kinases which is responsible for cell response to growth factors [10]. We used a simplified MAPK cascade as a starting point for testing our evolutionary framework. In particular, we analyzed the evolution of a population according to a fitness function which captures the essential behavior of the MAPK model. We generated an initial population of 500 individuals describing an ancestral organism that possess all the base proteins but lacks a signalling system similar to the MAPK cascade as observed today. The dynamics of each individual is then simulated; we ran each individual for 7000 simulation steps and we remove the signal at a given simulation step. Using the output of the simulation, we then measured for each individual the corresponding fitness. The fitness function we implemented is based on the integration of the simulation traces of the kinases and measures how rapidly the output of an active kinase increases and how much the output of the same kinase persists after removing the signal before returning back to the initial condition. We iterated the evolution algorithm for 2000 generations, for different values of fitness function parameters. We did not obtain individuals with a perfect MAPK cascade network, but some individuals had very good fitness values and showed the two directions in which evolution went to build an ultra-sensitive switch, namely forming longer cascades with multiple kinases or having multiple phosphorylation sites. More details about our case study can be found in [5].



(a) Boxes and boxes interactions

(b) Individual

Fig. 1: a) The small squares on the border of boxes are the interfaces; C, D E,... are the sorts; P1, P2,... are internal programs and B1, B2,... are the names of the boxes (e.g. protein species). The arrows are the graphical representation of the relation α. Numbers represent stochastic rates. b) Examples of an individual.



(a)

(b)

(c)

(d)

Fig. 2: Different kinds of mutations: in (a) the initial configuration; in (b) duplication of protein C followed by mutation of domain out of C1 in (c). Finally, (d) displays how the internal structure could change to accommodate the duplication of a domain. Plus and minus are the activation and inhibition messages, respectively.

Funct. 1:

```
EvolutionAlgorithm ():
Population = GenerateInitialPopulation();
for i == 0 to generations do
  for each Individual in Population do
     output = Simulate(Individual);
     fitnesses[Individual] = ComputeFitness(output);
  NewPopulation=Replicate&Mutate(fitnesses,Population);
  Population = NewPopulation;
```

Biography

Alessandro Romanel

Alessandro obtained a MSc (2006) and a PhD (2010) in Computer Science from the University of Trento. He carried out his doctoral research in computational systems biology at CoSBi, advised by Professor Corrado Priami. In 2004 he worked as software analyst and developer at CREATE-NET International Research Center. In 2008 he was a visiting researcher at the CIS department at the University of Pennsylvania, supervised by Professor Benjamin C. Pierce. From January till December 2010 he was a post-doctoral researcher at CoSBi. From January 2011 he is a post-doctoral researcher at INRIA, working with Jérôme Feret in the AbstractCell project at the École normale supérieure.

References

[1] O. Soyer, S. Bonhoeer. Evolution of complexity in signaling pathways. PNAS (103) (2006) 16337-16342

[2] T. Pfeier, S. Schuster. Game-theoretical approaches to studying the evolution of biochemical systems. Trends Biochem Sci. 1(30) (Jan 2005) 20-5

[3] T. Pfeier, O. Soyer, S. Bonhoeer. The evolution of connectivity in metabolic networks. PLoS Biol. 7(3) (2005)

[4] P. Francois, V. Hakim. Design of genetic networks with specified functions by evolution in silico. Proc. Natl. Acad. Sci. 2(101) (2004) 580-5.

[5] L. Dematte, C. Priami, A. Romanel, O. Soyer. Evolving BlenX programs to simulate the evolution of biological networks. Theoretical Computer Science 408(1):83-96, 2008.

[6] L. Dematte, R. Larcher, A. Palmisano, C. Priami, A. Romanel. Programming Biology in BlenX. In Systems Biology for Signaling Networks, Springer, 2010.

[7] L. Dematte, C. Priami, A. Romanel. The Beta Workbench: a computational tool to study the dynamics of biological systems. Briefings in Bioinformatics 9(5):437-449, 2008.

[8] A. Romanel. Dynamic Biological Modelling: a language-based approach. PhD thesis, University of Trento, 2010.

[9] D. T. Gillespie. Exact stochastic simulation of coupled chemical reactions. J. Phys. Chem., 81(25):2340–2361, 1977.

[10] C. Y. Huang and J. E. Ferrell. Ultrasensitivity in the mitogen-activated protein kinase cascade. Proc Natl Acad Sci USA, 93(19):10078–10083, September 1996.

[11] M. Kwiatkowski A formal computational framework for the study of molecular evolution. (2010) Ph.D. thesis in Informatics.

Abstract

RFID technology plays a key role in various areas of interest without guaranteeing security and privacy issues. Limitations of tag design make privacy and security enforcement a special challenge. In this paper, we introduce the essential components of an RFID system. Subsequently the RFID tag-to-reader authentication is presented as a solution framework for the tag cloning problem. The proposed solution involves receiving EPC RFID Generation-2 standard response messages, which vary according to power level when sent a tag KILL command.

Moreover, an empirical analysis is conducted to determine the necessary power level to kill the RFID tag for different brands and types. The intention of the conducted experiments is to determine the necessary power levels at which various RFID tags were able to be killed when preprogrammed with a specific KILL password. The authors found that results of power levels at which tags killed themselves varied widely.

An Empirical Study for Protecting Passive RFID Systems against Cloning

Mostafa M. El-Said and *Ira Woodring*

elsaidm@gvsu.edu - irawoodring@gmail.com

School of Computing and Information Systems
Grand Valley State University
Allendale, MI 49418

1. INTRODUCTION

The RFID system consists of three main components:
i. RFID tag (Transponder)
ii. RFID reader (Interrogator)
iii. RFID Backend System

Passive RFID tags consist of four main sub-units such as shown in Fig. 1. These units manage the tag's communication phases such as follows [1 and 6].

Energizing - Phase I: The tag is beaconed by an incoming RF signal from the reader.
Communication - Phase II: The tag's antenna detects the electromagnetic wave and induces energy into a capacitor. The capacitor feeds power into the tag's microchip. The tag performs the requested operation and returns a response back to the reader.
Unload - Phase III: The tag remains silent (capacitor unloaded) waiting for further commands from the reader. To complete these communication phases, the reader adjusts the signal power level according to the desired operation such as tag programming, tag query, tag access, tag lock and tag KILL [1, 3].

Passive RFID tags are very popular because of their simple design and cheaper cost (tag costs $0.05) [1]. The tag's simple design contradicts with the tag's required security. The major problem with an RFID tag is that it acts as an "always on" device in an open system. It talks with any RFID reader unit without any restrictions, releasing sensitive information about a certain object. Therefore, there is a pressing need to secure the RFID system.
The passive Tag-Reader security level has been overlooked by RFID industry because passive tags have been designed without security in mind. This risk resulted from the quick transition made from non-secure barcode based systems to non-secure RFID based systems. Consequently, a lot of backdoors have been opened for attackers to skim and clone tags with the same tag ID [3-4]. In this paper we will focus on solving the tag cloning problem.

Research efforts in securing RFID systems are gathering pace as RFID providers look for having an applicable solution across different tag types. However, the majority of the work done in this area suffers from the following weakness because:
- These solutions require a significant change to the tag chip design.
- Cryptographic function resources required by proposals [1, 5, and 7] are not available in the current passive tags.
- None of these proposals has been validated using simulation or real experimentation.

Looking beyond the issues addressed by today's RFID solutions, the proposed research aims to:
(i) Introduce a tag → reader authentication mechanism by controlling the power level of the KILL command. This results in protecting RFID systems against hacking using cloned tags.
(ii) Devise a mechanism to determine the correct power level of the KILL command to avoid killing the tag accidentally during the tag → reader authentication.

In this paper, we intended to scale up the work presented in [2] and use the KILL command for dual purposes. The original intent of the KILL command was to protect consumers' privacy and render tags inoperable as soon as they leave the point of sale.

2. TAG AUTHENTICATION SOLUTION APPROACH

In this paper, we conduct an empirical analysis to measure the effectiveness of the Kill-password technique to authenticate RFID tags. To render an RFID tag inoperable, a reader must send a kill command that includes the correct tag-specific kill password. Once this command is sent, the tag will respond with one of three messages:

- The tag was killed successfully,
- The tag did not gain enough power to complete the kill function, or
- The kill password was incorrect.

Our experimental analysis is based on controlling the power level of the reader, an authentication program can guarantee that there is not enough power to kill the tag, and will then be able to use the kill function to enable a simple password dialog. At this point the reader, issues an adjusted power-level Kill-command to verify the authenticity of the tag in the field and decide whether to deny or allow this tag on the system. So, the reader should know what power level to send to authenticate the tag. In order to do this, we built a testbed of software and hardware. The software component is a middleware that allows interfacing with the reader and relay messages to the target tag. The hardware component is made from various types of RFID readers and tags such as given in Fig. 2.

The testbed consists of the following components:
- Alien ALR-9814 RFID Portal
- Tags used for testing included:
  o Alien 9454 M-Tag Inlay
  o UPM RAFLATAC 3000707 Inlay
  o Alien 9440 Squiggle
  o UPM RAFLATAC 3000794 Frog Inlay
  o Avery AD-622 Inlay
  o Avery AD-220 Inlay
- The reader was connected to a LAN, with IP addressing occurring via the LAN router's internal DHCP mechanism.
- Power kill levels were obtained for all of the tags at a read range of 1.5 meters
- Testing software was written in Java, using the HTTP interface capability of the reader. The software included features for querying, writing, and killing tags, as well as changing power levels of the connected antennae.
- Application development unit (Compaq Presario 2100 laptop)

The tag programming phase is illustrated in Fig. 3.

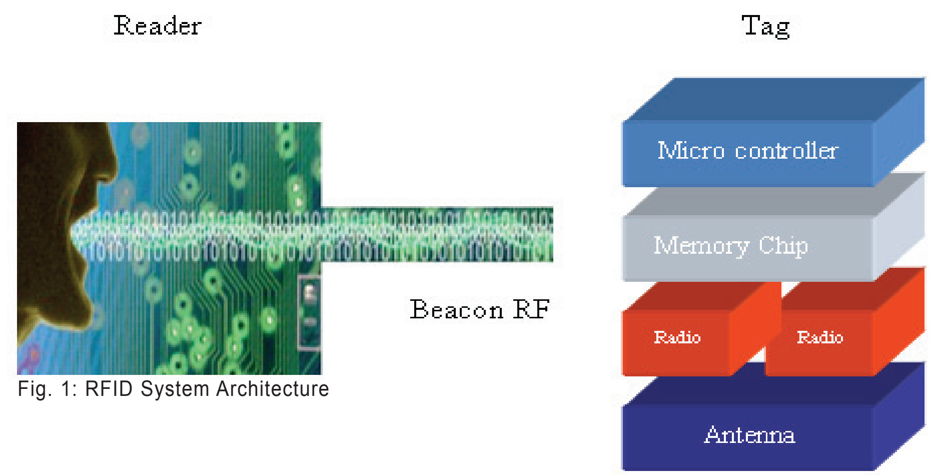


Fig. 1: RFID System Architecture

The procedure for writing tags merely programmed a tag EPC. Sending this command did not always result in a successful tag programming however; a revision early in the development of the software was to add a loop to send the command multiple times. This loop lead to much more consistent results with programming tags, yet took considerably longer to complete, upward of eleven seconds each time since this feature also included the one second delay between commands sent to the reader.

The read test described in Fig. 4 was conducted by initially programming the tag to be tested with a unique EPC (00 11 22 33 00 11 22 33). This was done to ensure that the tag could be recognized even if there happened to be another tag somewhere in the reader's field. The reader's power level was then set to 0, and the query command sent to the reader. An array of tags would then be returned from the reader and the results displayed in the message area. If multiple tags were found in the tag area, they were then found and removed. This process continued until the query verified that only one tag was in the field at a time. The power level was then increased by a tenth of a decibel, the tag query sent, and so on.

The power level range was increased to the reader's maximum value (15 db) so as to ensure that the tag had really been killed; in earlier tests it was found that though tags seemed to be dead at one power level, that they would appear at an increased power level, or even at the same power level given another query such as given in Fig. 5. It was therefore deemed prudent to test over the entire power range of the reader to make sure that a tag that seemed killed was really dead.

Biographies

Mostafa M. El-Said

Mostafa El-Said has received the M.S. and Ph.D. in Computer Science and Engineering from University of Louisville in 2000 and 2003 respectively. Also, Dr. El-Said received the B.S. and M.S. degrees in Electrical Engineering from Zagazig University, Egypt in 1992 and 1997 respectively. Dr. El-Said's research interests include designing Smart Autonomic VoIP and VANET Systems. He is currently the Vice Chair of the IEEE Simulation Letters. Also, he is currently a member of the international programme committee and a reviewer for IEEE SmartGridComm, IEEE CCNC, SIMUTools, CGAMES and IJIGS. Dr. El-Said joined the faculty in The Pennsylvania State University, Information Sciences and Technology (IST) School in 2003. Since 2004, he has been with the Grand Valley State University (GVSU), where he is currently an Associate Professor in the School of Computing and Information Systems. He is the founder and the director of the Wireless Systems Lab and the director of the Data Communication Center in GVSU.

Ira Woodring

Ira Woodring has received the B.Sc. from the Grand Valley State University in 2010. Ira's research interests include designing Smart RFID systems. Recently, he published an article in conducting an Empirical Study for Protecting Passive RFID Systems against Cloning. The study has been published in The Sixth International Conference on Information Technology: New Generations. Currently He is the System Administrator for the school of Computing and Information Systems at Grand Valley State University.

References

[1] Ari Juels, Ronald L. Rivest, Michael Szydlo (2003). “The blocker tag: selective blocking of RFID tags for consumer privacy”. CCS ‘03: Proceedings of the 10th ACM conference on Computer and communications security

[2] Ari Juels, Strengthening EPC Tags Against Cloning WiSe ‘05: Proceedings of the 4th ACM workshop on Wireless security, September 2005

[3] Bolan, C. (2006), “The Lazarus Effect: Resurrecting Killed RFID Tags”, Proceedings of the 4th Australian Information Security Management Conference, 4th December, 2006, Edith Cowan University, Perth, Western Australia

[4] D. R. Thompson, N. Chaudhry, and C. W. Thompson (2006), “RFID security threat model” in Proc. Acxiom Laboratory for Applied Research (ALAR) Conf. on Applied Research in Information Technology, Conway, Arkansas, Mar. 3, 2006.

[5] Dang Nguyen Duc, Jaemin Park, Hyunrok Lee, Kwangjo Kim (2006). “Enhancing Security of EPCglobal Gen-2 RFID Tag against Traceability and Cloning”. The 2006 Symposium on Cryptography and Information Security Hiroshima, Japan, Jan. 17-20, 2006

[6] Daniel M. Dobkin, Titus Wandinger (2008). “The RF in RFID: A Radio-oriented Introduction to Radio Frequency Identification” Available Online at: www.wj.com/documents/Articles_PDF/RF_in_RFID_v0.1.pdf

[7] David Molnar, Andrea Soppera, David Wagner (2005). “Privacy for RFID through trusted computing”. Proceedings of the 2005 ACM workshop on Privacy in the electronic society Publisher: ACM Soppera, David Wagner (2005). “Privacy for RFID through trusted computing”. Proceedings of the 2005 ACM workshop on Privacy in the electronic society Publisher: ACM

3. CONCLUSIONS AND FUTURE WORK

The main objective of the proposed research is to introduce a low cost solution for the RFID tag cloning problem by enforcing tag authentication. We have presented a simple protocol for passive RFID tags, especially EPCGlobal Class-1 Gen-2 RFID tags. Our protocol achieves desirable security features of a RFID system including: implicit tag-to-reader authentication.

This approach was deemed as a possible way to implicitly verify an RFID tag; if a correct password was sent to the tag, it should respond with an error code message, specifically “Not enough power to complete command”. The authors found that results of power levels at which tags killed themselves varied widely. This is one of the first attempts to create a universal tag killing power level database for each tag profile.

Future work will seek to develop a light weight reader → tag solution.

Tag	Tag Killed Power Value in dB
Alien 9454 M-Tag Inlay	4.7
UPM RAFLATAC 3000707 Inlay	4.1
Alien 9440 Squiggle	3.5
UPM RAFLATAC 3000794 Frog Inlay	8.8
Avery AD-622 Inlay	8.6
Avery AD-220 Inlay	3.9

Tab. 1: The table describes a direct relationship between the Tag's Consistent Read Value for 5 times and the Tag Killed Value.

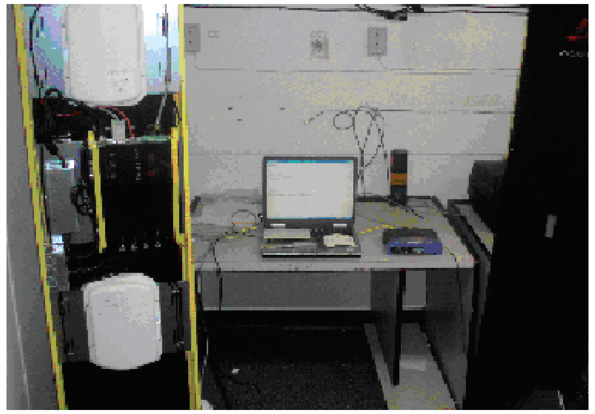
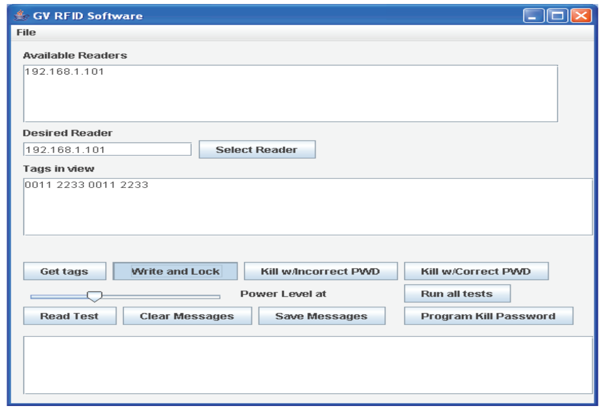


Fig. 2: RFID Testbed



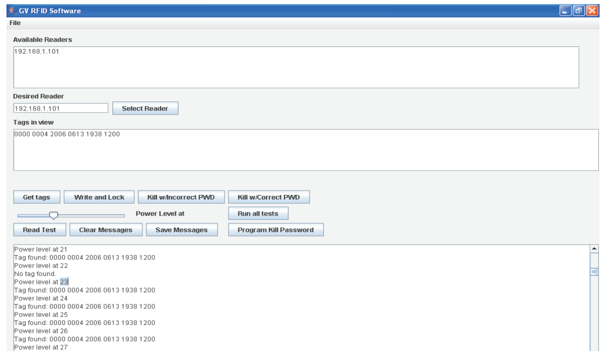
GV RFID Software

Available Readers: 192.168.1.101

Desired Reader: 192.168.1.101

Tags in view: 0011 2233 0011 2233

Fig. 3: Tag Programming with an EPC Number



GV RFID Software

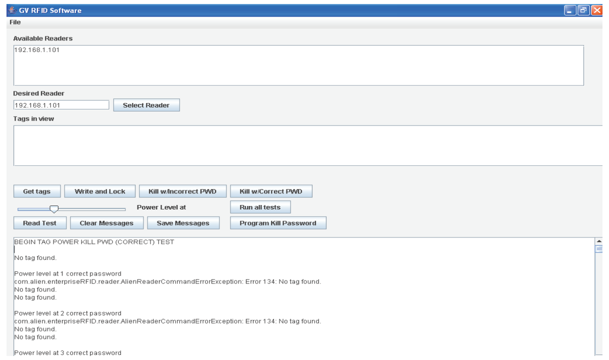
Available Readers: 192.168.1.101

Desired Reader: 192.168.1.101

Tags in view: 0000 0004 2006 0613 1938 1200

```
Power level at 21
Tag found: 0000 0004 2006 0613 1938 1200
Power level at 22
No tag found.
Power level at 23
Tag found: 0000 0004 2006 0613 1938 1200
Power level at 24
Tag found: 0000 0004 2006 0613 1938 1200
Power level at 25
Tag found: 0000 0004 2006 0613 1938 1200
Power level at 26
Tag found: 0000 0004 2006 0613 1938 1200
Power level at 27
```

Figure 4. Tag Read Test



GV RFID Software

Available Readers: 192.168.1.101

Desired Reader: 192.168.1.101

Tags in view:

```
BEGIN TAG POWER KILL PWD (CORRECT) TEST

No tag found.

Power level at 1 correct password
com.alien.enterpriseRFID.reader.AlienReaderCommandErrorException: Error 134: No tag found.
No tag found.
No tag found.

Power level at 2 correct password
com.alien.enterpriseRFID.reader.AlienReaderCommandErrorException: Error 134: No tag found.
No tag found.
No tag found.

Power level at 3 correct password
```

Fig. 5: Tad Power killing Test

Recent and Upcoming Sponsored Events

PADS 2011 (25th International Workshop on Principles of Advanced and Distributed Simulation)
June 14-17, Braga Portugal
https://sites.google.com/site/pads2011/

DS-RT 2011 The 15th International Symposium on Distributed Simulation and Real Time Applications
September 4-7, 2011
MediaCITYUK, Salford, near Manchester, UK
http://c4i.gmu.edu/events/conferences/2011/DS-RT/

CMSB 2011 The 9th International Conference on Computational Methods in Systems Biology
September 21-23, 2011
Institut Henri Poincaré, Paris, France
http://contraintes.inria.fr/CMSB11/

Issue 09 June 2011

Call for Papers: Special Issue on Modeling and Simulation of Smart and Green Computing Systems

Final submissions due: 1 March 2012
Acceptance notification: 1 May, 2012
Final papers due: 31 May 2012
Publication date: September 2012

Sustainable and efficient utilization of available energy resources is perhaps the fundamental challenge of the current century. Academic and industrial communities have invested significant efforts in developing new solutions to address the energy efficiency challenges in several areas from IT and telecommunications, to green buildings and cities, and smart grid. Large scale and complex computing control and communication systems play a key role in almost all of these areas. Modeling and simulation methodologies for such complex and large-scale systems are necessary for comprehensive performance evaluation that precedes costly prototyping activities.

This special issue aims to disseminate the latest advances in modeling and simulation of smart and green computing systems, which are critical from the viewpoints of sustainable economic growth and environmental conservation. This special issue focuses on methodologies, simulation tools and techniques for evaluating computing, control and communication systems for achieving energy efficiency leading to long-term sustainability. Appropriate topics of interest include but are not limited to:

- *Modeling and simulations of energy efficient computing systems*
- *Modeling and simulations of green communications systems*
- *Modeling and simulations of smart grid applications*
- *Simulation of intelligent transportation systems*
- *Building and energy management simulations*
- *Modeling and simulations of nature inspired computing and communication systems*
- *Innovative modeling and simulation methodologies, and tools*
- *Prototypes and testbeds of energy efficient computing and communication systems.*

Articles should be understandable to a broad audience of science and engineering professionals. The writing should be practical and original, avoiding too much focus on theory, mathematics, jargon, and abstract concepts. Accepted papers will be professionally edited for content and style. All manuscripts are subject to peer-review on both technical merit and relevance to Computer's readership. Paper submissions are handled electronically. For author guidelines and information on how to submit a manuscript, please visit: http://www.computer.org/portal/web/peerreviewmagazines/computer.

Guest Editorial Team:
Prof. Kaushik Chowdhury (Lead)
Northeastern University, USA
Dr. Dave Cavalcanti,
Philips Research North America, USA
Prof. Mostafa El-Said
Grand Valley State University, USA
Dr. Tommaso Mazza
Center for Integrative Biology - UNITN, Italy
Dr. Chittabrata Ghosh
Nokia Research Center, Berkeley, USA