Available online at www.sciencedirect.com
SCIENCE DIRECT®

Electronic Notes in Theoretical Computer Science

Electronic Notes in Theoretical Computer Science 121 (2005) 87–99
www.elsevier.com/locate/entcs

# Hybrid Workflow and Bayesian Networks to Correlate Information in the Protection of Large Scale Critical Infrastructures

John Bigham[1] Xuan Jin[2] David Gamez[3] Chris Phillips[4]

*Department of Electronic Engineering*
*Queen Mary University of London*
*London*
*UK*

**Abstract**

Safeguard is a system that aims to improve the dependability and survivability of large complex critical infrastructures by using distributed autonomous agents to monitor and protect them. This paper describes the embedding of a workflow management system within one of the Safeguard agents to support real-time correlation of information from anomaly detectors, intrusion detection systems and other system monitors. The workflow management system interprets workflow models, which are represented by augmented Petri Nets modelling generic forms of attack or failure. The workflow management system also triggers appropriate responses automatically according to the reasoning results of Bayesian networks linked to transitions in the workflows. A case study example from the management of electricity distribution networks will be presented.

*Keywords:* Workflow, Bayesian Networks, Anomaly Detection

## 1 Introduction

Large Complex Critical Infrastructures (LCCIs) play a key role in modern life, providing services such as telecommunications, electricity, gas and water within and between countries. In recent years, far-reaching changes has been

[1] Email: john.bigham@elec.qmul.ac.uk
[2] Email: xuan.jin@elec.qmul.ac.uk
[3] Email: david.gamez@elec.qmul.ac.uk
[4] Email: chris.phillips@elec.qmul.ac.uk

doi:10.1016/j.entcs.2004.10.009

made on these infrastructures such as increased dependence on IP networks, increased import and export of electrical power and increased use of renewable energy supplies. These infrastructures are now highly interconnected and interdependent. This has made them more vulnerable to attacks, failures and accidents. Any malfunction in these infrastructures can cause serious international cascading failures. For example, in August 2003 the Blaster worm infected about 400,000 systems, six days later the Welchia (or Nachi) worm even infected Air Canada's check-in system and the US Navy and Marine Corps computers. On the very next day, the fast spreading Sobig.F worm reportedly produced over 100,000 copies of itself within the first 24 hours. Later this worm accounted for one in every 17 emails at its peak [1]. The high speed worm consumed a great deal of bandwidth, making the network congested and unstable. Another example is the cascading electrical power failure that happened in September 2003 when storm-tossed tree branch hit a Swiss transmission line and this caused another transmission line to overload and knock out French energy transmission to Italy. Detection and defence against attacks and failures in LCCIs is a field of study in which there are no silver bullets. This is due to the sheer size and diversity of attack and failure types. Correlation is an effective solution that combines distributed detection and response with integration of critical information from many sources.

Safeguard is a system that aims to enhance the dependability and survivability of LCCIs [2]. At present the availability and integrity of critical infrastructures are usually mainly monitored and maintained by human operators. Intrusion detection software has already been deployed in many LCCIs to help human operators monitor the system. However currently these software generate too many false positive and false negative alerts. Human operators are often overwhelmed when bursts of alerts arrive or misled by the wrong alert reports. More seriously, cascading alerts and failures can be aggravated when the operator cannot make decisions and act promptly. Safeguard uses agents to monitor and protect LCCIs by improving the capabilities of the automatic control functions and also helping human operators to make the right decisions and the right time. In Safeguard, the objective of its correlation agent is to make sense of diverse pieces of information and perform timely action. It correlates alerts in real-time from multiple heterogeneous detection systems. This paper describes the use of Petri-net modelled workflows to support correlation and monitoring of associated actions. Firstly a review of the main agents in the Safeguard system is given, followed by a description of the correlation agent in more details. A case study of the use of the correlation agent is then given.

## 2 Safeguard Agent System

### 2.1 Overview

The Safeguard agent system is implemented as a hierarchically layered agent system. It combines distributed detection and distributed response with integration of critical information from many sources. The Safeguard agent system has hybrid detector agents that monitor the operators, system components and system malfunction detectors within an infrastructure in order to assess the state of the system and if it contains erroneous data or under attack. Problems within the system, such as anomalous data or file integrity violations, will be identified. This information can be either passed to the operator or automatically acted upon, in order to prevent or limit inappropriate behaviour. The most important agents will now be covered in more detail.

### 2.2 Hybrid Detector Agents

The Hybrid detector agents (HDAs) are effectively sensors that are used to gather information about diverse aspects of the system. Typically, their role does not exceed passive monitoring, although some may perform certain actions on the managed system, but only if explicitly permitted by the action agent. HDAs combine known information with a dynamic model of the system's normal behaviour. A large number of different types of dedicated agents are placed in the system to monitor many aspects of system activity. Examples of HDAs are:

(i) a keystroke anomaly detector agent, which examines the keystroke patterns of the different operators. Significant anomalies in an operator's keyboard patterns could indicate that someone else is using their terminal or password;

(ii) an electricity data anomaly detector that examines data from the remote terminal units and checks if previously established relationships hold in the current data;

(iii) another electricity anomaly detector that looks at time differences in functions called when responding to well known commands from the control centre.

Because many of the HDAs in Safeguard are based on constructing different models of normality or defining invariants or approximate invariants in the system, they are capable of detecting anomalies that have not occurred before.

### *2.3 Wrapper Agents*

The Wrapper Agents are attached to the existing intrusion detection systems that gather information about the system and possible attacks on the system. Wrapper agents simply allow information from existing diagnostic and IDS components can be integrated within the Safeguard system. Information from wrapper agents is sent to the correlation agents. An example is file integrity checker wrapper agent, which monitors integrity violations on critical system files.

### *2.4 Workflow Correlation Agents and Action Agents*

The Workflow Correlation agents (WCAs) contain an embedded workflow management system. Predefined workflow models for the managed network are loaded to the workflow management system. Transitions of these workflow models are associated with predefined Bayesian network models. Workflow correlation agents are responsible for integrating information from the different HDAs or wrapper agents and reasoning about the state of the network and behaviour of operators. Some of these transitions are used to model actions or to communicate with a separate action agent. In this way the correlation and action agents work together to provide a quick response that rectifies problems as they arise. An example of available responses includes changing firewall policies when a worm is reported by the WCAs to stop the propagation of the worm in the network.

### *2.5 Man Machine Interface Agent*

The Man Machine Interface (MMI) agent is used to manage the agents and define the scope of their legitimate activity. However the MMI agent is not the interface the operator uses to control the system.

### *2.6 Functioning of the Agent System*

The architecture of the agent system and the system being monitored is given in Fig. 1 Different hybrid detector agents are positioned in the system based on the type of the activity they are monitoring. Information from these is passed on to the correlation agent, which makes an assessment of the trustworthiness of actors and data from the system components. Based on this, the privileges of the operators or topology of the system is modified over time by the action agent with the authorisation of the administrator operating through the man machine interface agent.

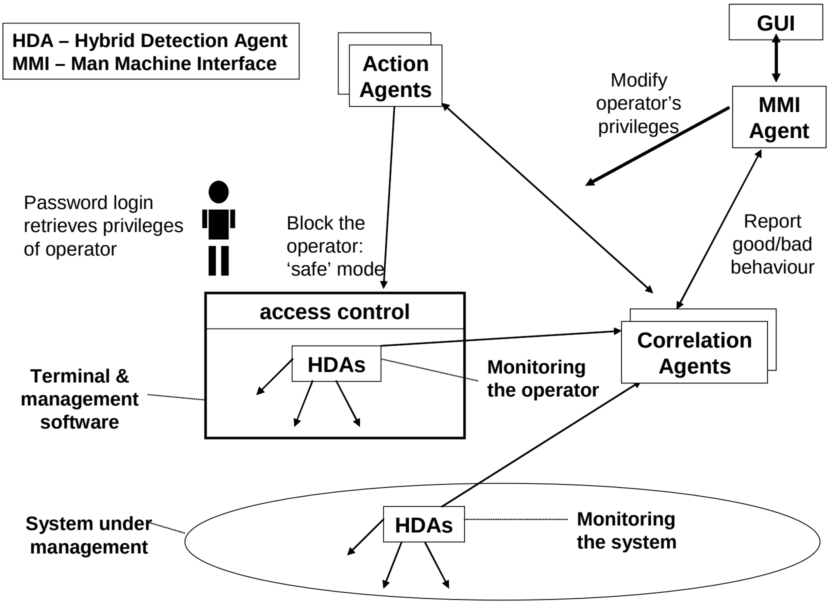
HDA – Hybrid Detection Agent
MMI – Man Machine Interface
Action Agents
GUI
Modify operator's privileges
MMI Agent
Password login retrieves privileges of operator
Block the operator: 'safe' mode
Report good/bad behaviour
access control
HDAs
Terminal & management software
Monitoring the operator
Correlation Agents
System under management
HDAs
Monitoring the system

Fig. 1. System monitoring through agent interaction

# 3 Hybrid Workflow and Bayesian Network Correlation

## 3.1 Workflow Overview

Workflows are defined by the Workflow Management Coalition as follows: "The automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules" [3]. They have been used in business for a number of years to model the flow of information within an organisation and the operations carried out on that information. These applications are oriented towards managing a complex sequence of activities.

## 3.2 Petri Net Modelled Workflow

W.M.P. van der Aalst [4] has described Petri Nets as a tool for modelling workflows and describes modelling with sequential activities, parallel activities, AND splits, OR splits (implicit and explicit), different joins and iteration.[5] This paper also describes standard ways that transitions in the workflow are triggered: Automatic, User, Message, and Time. In our research, we follow the terminology and notation of van der Aalst.

[5] A full explanation of explicit OR-split and implicit OR-split is available at page 20, [4]

### *3.3 Workflow Management System*

Workflow management systems are used to define, manage and execute workflows using software whose order of execution is driven by a computer representation of the workflow logic [5]. There are many commercial and non-commercial workflow management tools available such as Cosa [6] and Open-WFE [7]. Each application provides different functionalities and serves different users. In our research we chose the Bossa Workflow System [8] as our workflow management system because it has the following advantages:

(i) Bossa uses augmented Petri Nets to provide an intuitive way of modelling workflows and a way to verify workflow correctness. Extended Petri Nets even allow users to model time and include a hierarchy of workflow models.

(ii) Bossa is designed to be embedded and it is easy to define and dynamically load workflows in Bossa.

(iii) Bossa is written in Java, which can be platform independent. Also it is relatively easy to integrate Bossa with other Java code linked to workflow functions

(iv) Bossa is lightweight and fast. One reason is because tracking of position in a workflow is implemented without using separate threads for each workflow. This allows the agent to deal with a large number of different workflows relating to different event sequences in the monitored system.

### *3.4 Constructing Workflows*

Basic Petri Net modeled workflows for Bossa workflow management system have the following elements: Places, Transitions, and weighted Arcs. Workflows in our system can only be started at one point. This point should be marked by placing a single token at that point. Four different types of transition can be used to construct a workflow. The transition type is set using the first few letters of its name.

**Bayesian controlled transitions**

These transitions are fired when the probability of the node in the Bayesian network identified as corresponding to the transition exceeds the prescribed threshold.

**Message sending transitions**

When a workflow reaches a message sending transition a message is sent to another agent. For example, a message sending transition can send a message

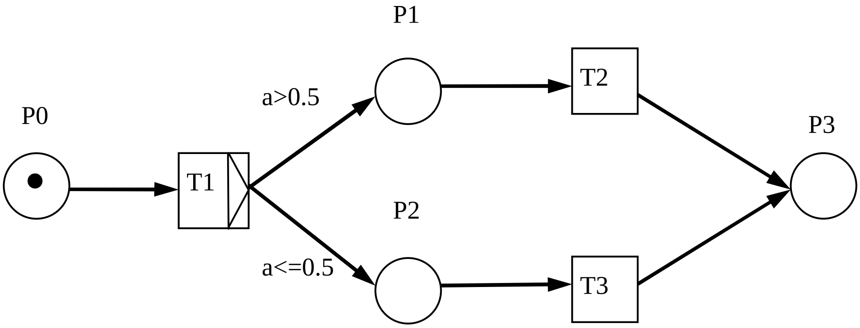

Fig. 2. Typical Petri Net modeled workflow

to the MMI agent with the subject 'WorkflowStarted'. The attributes of the workflow will also be sent to the receiving agent when the transition fires. It is also possible for the agent to send a message to itself. This feature can be used to start a workflow or set an observable in another workflow.

**Timer Transition**

A timer transition either sets a timer or checks to see if the timer has expired. Timer transition can be used to execute periodic tasks.

**Ordinary transitions**

An ordinary transition is used to route the workflow based on static or dynamic attributes. These transitions are always fireable when they are reached by the workflow.

A simple Petri Net modeled workflow using our interface to the Bossa workflow engine is shown as Fig. 2 Transition T1 has an explicit OR-split. Variable a is global to the particular workflow but invisible to other workflows. When transition T1 is fired, according the evaluation results of the post-conditions of T1, the token in place P0 will be transferred to P1 or P2.

### *3.5 Constructing Bayesian Networks*

Bayesian networks are constructed in an XML format using the graphical capabilities of the JavaBayes software [9]. The EBayes software [16] is embedded in the correlation agent and this is used when updating the beliefs of nodes in the Bayesian networks that are linked to transitions in the workflow. A Bayesian network provides a link between incoming messages from anomaly detectors and a transition in the workflow. Each Bayesian network contains one or more observable nodes which corresponding to incoming messages. Observables are linked by the Bayesian network to nodes associated

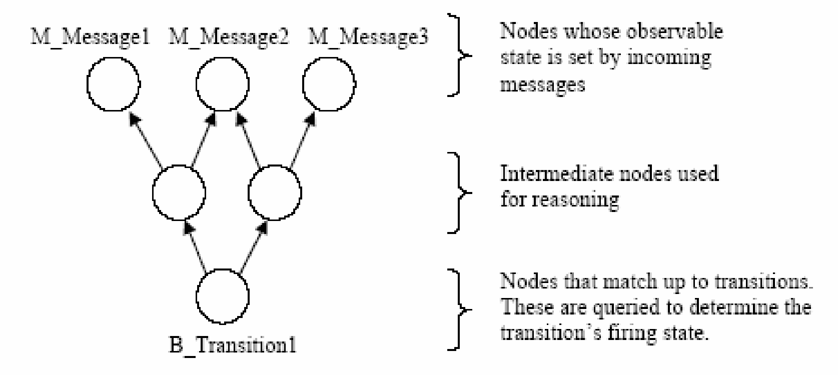
M_Message1
M_Message2
M_Message3
Nodes whose observable state is set by incoming messages
Intermediate nodes used for reasoning
Nodes that match up to transitions. These are queried to determine the transition's firing state.
B_Transition1

Fig. 3. Typical Bayesian network structure

with workflow transitions. The belief in the Bayesian network node linked with this associated Petri net transition node is queried to see if the transition should fire at runtime. Deterministic relationships between observables and transitions are handled as a degenerate case where the probabilities are 0 and 1. Observable nodes do not depend upon any other Bayesian network nodes as they are set directly by other agents of Safeguard system. Fig. 3 shows the structure. A tree structure is not required; the network can be an acyclic graph.

### 3.6 Hybrid Workflow and Bayesian Network Correlation

In our research we are applying workflows to model and monitor both normal and abnormal flow of activities within an organisation (such as a power plant or a telecommunication carrier) and the operators working in the control centre and repair facilities. It is understood that many of the workflows are not already in place. Messages from the different anomaly detectors are "pushed" to the correlation agent. Each of them has a value of true or false, which will be used to set observable nodes in the Bayesian networks in the appropriate workflows. (This could be generalized so that probabilities could be passed, but it is not so in the current implementation.) When repeated messages are sent the latest incoming message will update the corresponding observable node to it new value. These messages are stored by the correlation agent so that it can retrieve attributes of the message when a transition fires. By doing so we are able to correlate different reports and alerts from different intrusion detection agents at different times and stages of evolution of an attack.

A simple and abstract Bayesian network controlled workflow is shown as Fig. 4 (In section 4, a concrete case study of workflow correlation will be illustrated.) Transition T1 has an explicit OR-split and it is associated with

Bayesian network B1. Incoming messages from different anomaly detectors update the observable nodes in B1. Bayesian network will work out the probability of proposition B_Transition1. If the probability exceeds a certain threshold, transition T1 will fire. Then the token in place Place0 will be transferred to the corresponding place according to the value of the attribute a. Attribute a is an internal variable that is maintained by the workflow engine. Its value can be changed by messages from other agents. This routing continues until the end of workflow is reached.

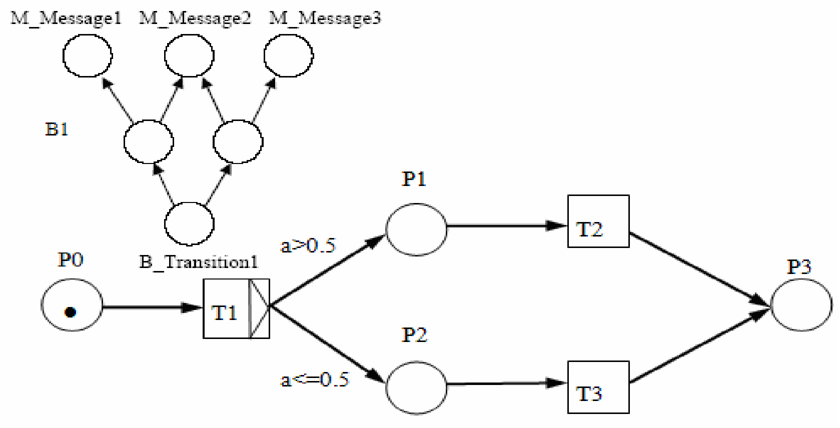

Fig. 4. Workflow controlled by Bayesian network

There can be hundreds of predefined workflow types existing in a single correlation agent and for each workflow type there could be many instances of this workflow running simultaneously. The Bossa workflow management system handles the concurrency and parallel execution.

## 4 Correlation Case Study

In this scenario a worm enters the electricity control centre's local area network and starts to scan and copy itself onto other machines. The Safeguard hybrid detector agents and wrapper agents detect some of the consequences, such as network congestion, scanning, and file modifications. This information is passed to the correlation agent.

Two families of worm were emulated for test purposes, namely Code Red and Slammer. Only the latter is reported here. Slammer uses UDP for its propagation and sends a single infection packet to a randomly chosen IP address. To control the propagation rate, the range from which this random IP address is chosen can be set using a command line argument. The impact of the worm emulation upon the electricity network and the Safeguard response were measured by using Network Probe to monitor the total network traffic. The worm emulation was initially run without Safeguard and then with Safeguard so that the effectiveness of the Safeguard response could be evaluated. In the Safeguard tests, a string was included inside the worm and a signature written for Prelude IDS to test the ability of Safeguard to respond to a known worm. Safeguard was also tested without Prelude to evaluate the ability to detect and respond to an unknown worm on the network. All the tests were run with a single vulnerable process running on one machine and a single malicious process on a second machine.

A set of workflows correspond to mechanisms to detect an unknown worm were constructed. The workflow shown in Fig. 5 is one of them. The node circled is an example of a node that is triggered by an associated network, which is not shown. This node is fired when there is some indication, though not certainty, of a worm attack. Other activities are used for waiting and sending messages to the MMI agent.

Linked workflows relate to monitoring patterns of behaviour of a suspected attacking host and another that is initialized for each suspected victim machine. These are initialized on the first indications and then they build up evidence of attacker or victim. All the workflows are within the correlation agent and run simultaneously. In Fig. 6 the output of a network probe is shown. Normal traffic on the electrical network was around 1Mbit/sec. The Slammer traffic led to an increase in traffic and the rapid detection of the worm, but in this case it took some time for the killing messages from the Safeguard system to get through so that the recent worm processes could be eliminated.

## 5 Conclusion and Future Work

A prototype system for enhancing the survivability and dependability of large scale infrastructures is being constructed and is being evaluated on two important kinds of critical infrastructure, namely electricity distribution and the management of a telecommunications network. Instances of all the main agents of the system have been constructed. Current work is on implementing the test beds for the electricity and telecommunications domains and on evaluating the techniques developed.

This paper has concentrated on an aspect of our approach to correlation. By correlation is meant the synthesis of information from diverse kinds of anomaly detector and IDS and making sense of this information. Information is synthesized for patterns of attack that can extend over a period of time. The time events occur can matter. For example, a scan in itself may not be

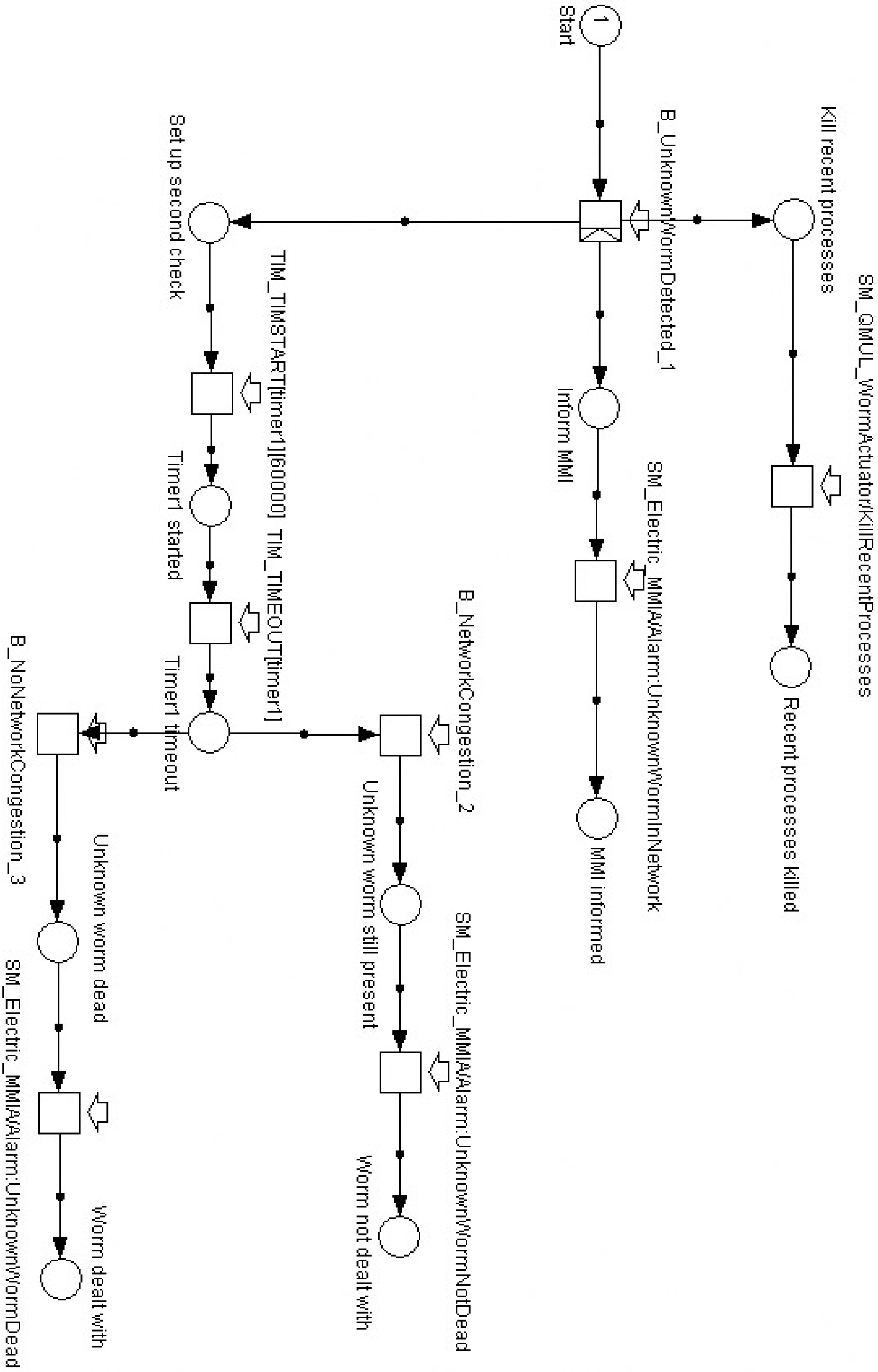
1
Start
B_UnknownWormDetected_1
Kill recent processes
SM_QMUL_WormActuator/KillRecentProcesses
Recent processes killed
Inform MMI
SM_Electric_MMI/A/Alarm:UnknownWormInNetwork
MMI informed
Set up second check
TIM_TIMSTART[timer1][60000] TIM_TIMEOUT[timer1]
Timer1 started
Timer1 timeout
B_NetworkCongestion_2
Unknown worm still present
SM_Electric_MMI/A/Alarm:UnknownWormNotDead
Worm not dealt with
B_NoNetworkCongestion_3
Unknown worm dead
SM_Electric_MMI/A/Alarm:UnknownWormDead
Worm dealt with

Fig. 5. Worm detection and elimination workflow

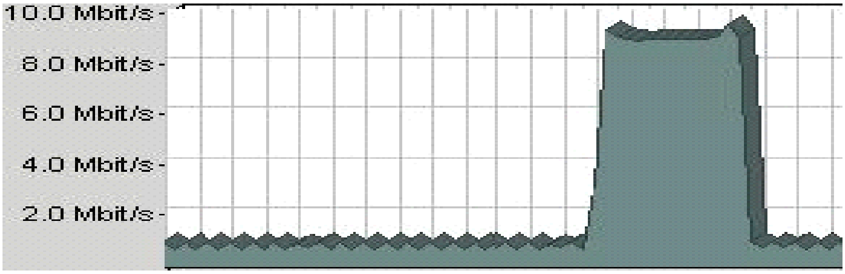

Fig. 6. A worm attack on the network and the effect on the traffic

significant, but in the context of a suspected kind of attack it can give supporting or contradictory evidence. An example of a successful use of workflows has been indicated

One important aspect of the approach not described here is the creation of a variety of anomaly detectors that detect deviations from normality. The system is trained under normal operation and then deviations from normality are detected. This is how the system can initially detect new kinds of attack. Work flows are used in an attempt to go a little further. For patterns of attack that have been analysed then a framework for confirmation and action is provided by the workflows. Action and synthesis of information can be intertwined.

## Acknowledgement

This work was partially funded by the CEC as part of the Safeguard project. The help from our partners Aplicaciones en Informática Avanzada (AIA) (Spain), ENEA (Italy), Linköping University (Sweden), Swisscom (Corporate Technology) (Switzerland) is much appreciated. Also Frank J. Wesslowski developed the graphical interface to the workflow software while he was an MSc student of the Electronic Engineering Department.

## References

[1] Thomas M. Chen, “Intrusion Detection for Viruses and Worms”, URL: http://engr.smu.edu/∼tchen/papers/iec2004.pdf

[2] Safeguard website, URL: http://www.ist-safeguard.org

[3] Layna Fischer (ed), “The Workflow Handbook 2003”, Published in association with the Workflow Management Coalition (WfMC)

[4] W.M.P. van der Aalst, *The Application of Petri Nets to Workflow Management*, Journal of Circuits, Systems, and Computers, 1998, 21-66.

[5] Workflow Management Consortium website, URL: http://www.wfmc.org

[6] Cosa website, URL: http://www.transflow.com/english

[7] OpenWFE website, URL: http://www.openwfe.org

[8] Bossa website, URL: http://www.bigbross.com/bossa.

[9] JavaBayes website, URL: http://www-2.cs.cmu.edu/∼javabayes/Home/index.html

[10] Ebayes website, URL: http://www-2.cs.cmu.edu/∼javabayes/EBayes/index.html