# International Journal on Advances in Systems and Measurements


2008 vol. 1 nr. 1

The *International Journal On Advances in Systems and Measurements* is Published by IARIA.
ISSN: 1942-261x
journals site: http://www.iariajournals.org
contact: petre@iaria.org



Sponsored by IARIA
www.iaria.org

**International Journal On Advances in Systems and Measurements**
**Volume 1, Number 1, 2008**

---

- Leon Reznik, Rochester Institute of Technology, USA
- Chi Zhang, Juniper Networks, USA

**Sensor Networks**

- Steven Corroy, Philips Research Europe – Eindhoven, The Netherlands
- Mario Freire, University of Beira Interior, Portugal / IEEE Computer Society - Portugal Chapter
- Jianlin Guo, Mitsubishi Electric Research Laboratories America, USA
- Zhen Liu, Nokia Research – Palo Alto, USA
- Winston KG Seah, Institute for Infocomm Research (Member of A*STAR), Singapore
- Radosveta Sokkulu, Ege University - Izmir, Turkey
- Athanasios Vasilakos, University of Western Macedonia, Greece

**Electronics**

- Kenneth Blair Kent, University of New Brunswick, Canada
- Josu Etxaniz Maranon, Euskal Herriko Unibertsitatea/Universidad del Pais Vasco, Spain
- Mark Brian Josephs, London South Bank University, UK
- Michael Hubner, Universitaet Karlsruhe (TH), Germany
- Nor K. Noordin, Universiti Putra Malaysia, Malaysia
- Arnaldo Oliveira, Universidade de Aveiro, Portugal
- Candid Reig, University of Valencia, Spain
- Sofiene Tahar, Concordia University , Canada
- Felix Toran, European Space Agency/Centre Spatial de Toulouse, France
- Yousaf Zafar, Gwangju Institute of Science and Technology (GIST), Republic of Korea
- David Zammit-Mangion, University of Malta-Msida, Malta

**Testing and Validation**

- Cecilia Metra, DEIS-ARCES-University of Bologna, Italy
- Krzysztof Rogoz, Motorola, Poland
- Rajarajan Senguttuvan, Texas Instruments, USA
- Sergio Soares, Federal University of Pernambuco, Brazil
- Alin Stefanescu, SAP Research, Germany
- Massimo Tivoli, Universita degli Studi dell'Aquila, Italy

**Simulations**

- Tejas R. Gandhi, Virtua Health-Marlton, USA
- Ken Hawick, Massey University - Albany, New Zealand
- Robert de Souza, The Logistics Institute - Asia Pacific, Singapore
- Michael J. North, Argonne National Laboratory, USA

**International Journal On Advances in Systems and Measurements**
**Volume 1, Number 1, 2008**

## Foreword

Finally, we did it! It was a long exercise to have this inaugural number of the journal featuring extended versions of selected papers from the IARIA conferences.

With this 2008, Vol. 1 No.1, we open a long series of hopefully interesting and useful articles on advanced topics covering both industrial tendencies and academic trends. The publication is by-invitation-only and implies a second round of reviews, following the first round of reviews during the paper selection for the conferences.

Starting with 2009, quarterly issues are scheduled, so the outstanding papers presented in IARIA conferences can be enhanced and presented to a large scientific community. Their content is freely distributed from the www.iariajournals.org and will be indefinitely hosted and accessible to everybody from anywhere, with no password, membership, or other restrictive access.

We are grateful to the members of the Editorial Board that will take full responsibility starting with the 2009, Vol 2, No1. We thank all volunteers that contributed to review and validate the contributions for the very first issue, while the Board was getting born. Starting with 2009 issues, the Editor-in Chief will take this editorial role and handle through the Editorial Board the process of publishing the best selected papers.

Some issues may cover specific areas across many IARIA conferences or dedicated to a particular conference. The target is to offer a chance that an extended version of outstanding papers to be published in the journal. Additional efforts are assumed from the authors, as invitation doesn't necessarily imply immediate acceptance.

This particular issue covers papers invited from those presented in 2007 and early 2008 conferences. The papers cover mechanisms, techniques and applications using agile technology, modular design, process-aware and workflow diagrams for life cycles. Particular experiments are reported on semantic data processing and security-critical applications.

We hope in a successful launching and expect your contributions via our events.

First Issue Coordinators,
Jaime Lloret, Universidad Politécnica de Valencia, Spain
Pascal Lorenz, Université de Haute Alsace, France
Petre Dini, Cisco Systems, Inc., USA / Concordia University, Canada

**International Journal On Advances in Systems and Measurements**
**Volume 1, Number 1, 2008**

**CONTENTS**

# Activity Monitoring for large honeynets and network telescopes

Jérôme Francois, Radu State and Olivier Festor
Madynes research group
INRIA-LORIA
615, rue du jardin botanique
54600 Villers-les-Nancy
Nancy, France

***Abstract*—This paper proposes a new distributed monitoring approach based on the notion of centrality of a graph and its evolution in time. We consider an activity profiling method for a distributed monitoring platform and illustrate its usage in two different target deployments. The first one concerns the monitoring of a distributed honeynet, while the second deployment target is the monitoring of a large network telescope. The central concept underlying our work are the intersection graphs and a centrality based locality statistics. These graphs have not been used widely in the field of network security. The advantage of this method is that analyzing aggregated activity data is possible by considering the curve of the maximum locality statistics and that important change point moments are well identified.**

***Index Terms*—honeypot, backscatter, telescope, monitoring, intersection graphs, centrality, locality statistics**

## I. Introduction

The motivations of this paper are twofolds. The first motivation of our work is related to the conceptual approaches and algorithms required to perform distributed monitoring. If we consider a distributed monitoring platform for a given target deployment (please see figure 1), several questions must be addressed.

- Do all management agents observe the same type of events ? If no, how can we correlate a distributed view and aggregate the commonly observed evidence?
- Can we discover a temporal behavior of the whole platform ? Do some agents tend to observe the same type of behavior during a particular time of the day, while others remain to hold a localized and very isolated observation behavior ?

A second motivation of our work came from a very realistic requirements. We are part of a large honeynet distributed over the Internet. Each individual honeypot monitors backscatter packets and incoming attacks. When working on the resulted datasets, we were challenged by the lack of methods capable to compare such a distributed platforms and to detect temporal/spatial trends in the observed traffic patterns. In our work we had to process similar attack traffic from a different security monitoring platform (a network telescope) and compare it to the results obtained from the honeynet. This paper extends our previous works [1] and [2].

Our paper is structured as follows: in section 2, a generic method for analyzing a distributed monitoring platform is described. This method uses graph intersections in order to model the distributed platform and to follow their temporal evolution. Section 3 describes two realistic distributed environments (a honeynet and a network telescope) and section 4 shows how this method can be used for them. An analysis concerning IP related headers is done for the two data sources and additional results concerning differences and analogous behavior between these two are presented. Section 5 presents related works and finally section 6 concludes the paper.

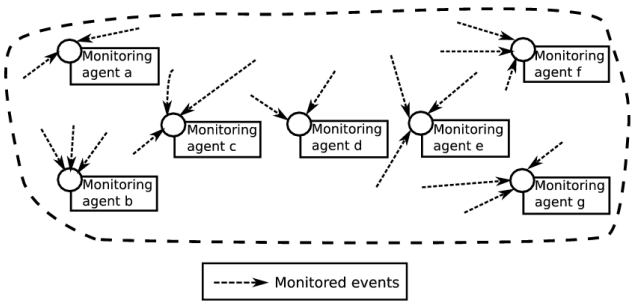



Fig. 1. Distributed monitoring model

## II. Intersection graphs

The method based on intersection graphs has been introduced in [3] for profiling communications patterns between the users of a high profiled enterprise. Actually, the data used were the exchanged emails and the goal was to detect if someone was aware of the Enron scandal before it was revealed. Thanks to this method, the authors observe that there were significative changes of the graph topology and highlight the responsable nodes which are in reality people. Therefore, using this technique seems to be a good way to detect behavior changes of the attacks in the Internet and IP addresses which are concerned by these changes.

### A. Graphs and activity profiling

A graph is composed of several nodes and arcs. Two nodes are linked if there is a relation between them. A relation can be: similarity, difference, or communication exchanges. The relation will be formally defined for each deployment target in the following sections. We consider that arcs are not directed and that the graph is an undirected graph. The adjacency matrix of a graph is a boolean square matrix where each line and each column represents a node. It is defined as :

$$A_{ij} = 1 \ if \ an \ arc \ between \ i \ and \ j \ exists, \ 0 \ else$$

$$where \ i \ and \ j \ are \ 2 \ vertices \ of \ the \ graph$$

Since we consider a undirected graph, the adjacency matrix is symmetric :

$$A_{ij} = A_{ji}(symmetrical \ matrix)$$

As we want to connect nodes which share or don't share some characteristics, it is totally useless for a node connected to be connected to itself and we will consider this statement as an assumption in all this article.

If we consider the figure 2, the corresponding adjacency matrix is :

$$A = \begin{array}{c|ccccccc} & a & b & c & d & e & f & g \\ a & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ b & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ c & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ d & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ e & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ f & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ g & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{array}$$

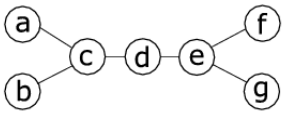



Fig. 2. An undirected graph

### B. Central node

Generally, a central node is interesting because it has multiple direct or indirect relations. Using the most central node we can evaluate the centrality of the graph by counting the number of relations (arcs). A simple method to detect this node could be to get the node which has the maximum number of neighbors.

For example, in figure 2 the most connected nodes are c and e with 3 neighbors. However, if we consider the node d, this one seems to be also well connected, although it has only 2 neighbors. In fact, if a node has only few relations but these relations lead to nodes that are well connected, then the original node is interesting and central. Therefore, we can consider not only the direct neighbors but a subgraph of all nodes which are located in an area defined by the distance from the evaluated node. The centrality is the number of arcs of the subgraph. This is the main idea used in [3].

In figure 2, considering an exploring distance k = 2, nodes c and e have a centrality of 4. For the node d, the associated value is 6. Based on this method, the central node is d.

Another way to get the central nodes is to use the eigenvalues and eigenvectors, as proposed in [4]. Assuming an adjacency matrix $A$, $x$ an eigen vector and $\lambda$ the corresponding eigen value, we have :

$$A \times x = \lambda \times x$$

The more central node is the highest value in the eigenvector of the highest eigenvalue. Considering the figure 2 and the previously introduced adjacency matrix, this vector is (-0.5, 0, -0.316, 0.500, 0.000, -0.447, -0.447). The maximal value is the fourth which corresponds to the node $d$ once again.

Thus, different methods can be used and we propose to use the first one in this paper because it is done easily by walking in the graph and because we can compute the centrality incrementally for different distances i.e. by increasing the depth of the walking contrary to the second methods where the eigenvectors and eigenvalues are to be recomputed for each submatrix.

### C. Locality statistics

A graph can vary over the time and thus we need to somehow capture and describe variations in the centrality. The main idea is to consider at each time instant the central node and the associated centrality and to analyze the temporal behavior of these two entities. The intuition behind is that when major graph changes occur in the topologies of a graph, the relations between nodes change and this will be reflected by a change in the centrality too. So, detecting changes in the graph can be highlighted by looking for the maximal centrality as proposed in [3]. This method has the advantage that one value is an indicator of the graph topology contrary to have one value per node. If more details are needed, the central node which is responsible of the maximal centrality can be detected and the appearance or disappearance of a node implies that its relationships increased or respectively decreased.

The following formula describes formally the maximal locality statistic, described in the previous paragraph :

$$\psi_k(v) = \ number \ of \ arcs \ of \ the \ subgraph$$

$$of \ neighbors \ of \ v \ at \ a \ maximal \ distance \ k$$

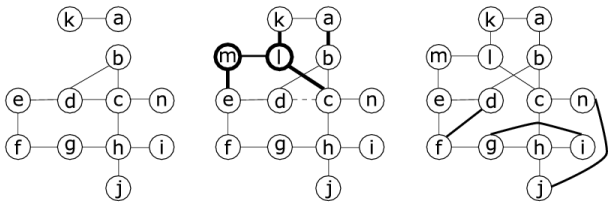
$$M_k = \max_{v \in nodes} \psi_k(v) \quad (1)$$

Actually, the number of neighbors at a maximal distance $k$ is computed for each node. Then $M_k$ is the maximum value that were be calculated.

Consider the example of the evolution of a graph which is described below and presented briefly in the figure 3:

- t = 1 : 10 nodes, 11 arcs

- t = 2 : node and arcs added but with isolated node
- t = 3 : increase of number of arcs
- t = 4 et 5 : 5 arcs added
- t = 6 : 5 nodes removed, about linear graph
- t = 7 : increase of nodes and arcs
- t = 8 : remove only one node which was isolated
- t = 9 : increase of nodes and arcs
- t = 10 : 5 nodes removed, non linear but scattered graph




(a) t = 1 (b) t = 2 (c) t = 4 et t = 5

Fig. 3. Graph time series (bold line : adding, dashed line : removing)

Figure 4 presents the result of this formula with different values of k = 1..4. For k = 0, the value is always 0 which is normal because in this case no neighbors are concerned and only the current node composes the subgraph. Varying $k$ allows to select information and especially to limit the subgraph of extended neighbors in order to avoid to have a constant maximal locality statistic which corresponds to a subgraph covering all the graph.

The values for k = 3 and k = 4 are identical and that means that for k less than 3 it's possible to find a node having the associated subgraph of neighbors covering the total graph. This observation shows that the choice of k is important. k must not be too small because important information might not be revealed. If k is to large, all the graph is covered. In our case, the value of k = 2 seems to be a good choice.

In the figure 4, the plot for k = 2 increases up to 5 because the graph has more and more nodes and arcs. We can also observe that due to the linearity of the graph, the locality statistics decreases (t = 6). The maxima locality statistics allowed to observe this evolution. Large values of this statistics are to be associated with major changes in the inter-node relationships.

It is also important to observe the responsible nodes associated to the peaks of the maximal locality statistic (maximum centrality). In the previous example, node c is always central.

The major goal is not only to show the evolution of the topology of the graph but in fact to discover new nodes that might become important. For instance, for time instants 3 and 4, node c is the only central node. This centrality is equal to 12 and respectively 15. The same analysis for the node g shows that its values goes from 6 to 12. In all cases, its centrality is lower that the one of c, but the evolution of g is more interesting. This type of behavior can be put into evidence by a standardized locality statistics at time $t$:

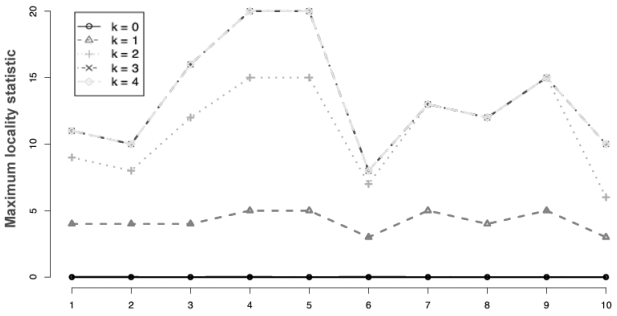



Fig. 4. Locality statistics according to time

$$\tilde{\psi}_{k,t}(v) = \frac{(\psi_{k,t}(v) - \hat{\mu}_{k,t,\tau}(v))}{\max(\hat{\sigma}_{k,t,\tau}(v), 1)}$$

$$\hat{\mu}_{k,t,\tau}(v) = \frac{1}{\tau} * \sum_{t'=t-\tau}^{t-1} \psi_{k,t'}(v)$$

$$\hat{\sigma}_{k,t,\tau}(v) = \frac{1}{\tau - 1} \sum_{t'=t-\tau}^{t-1} (\psi_{k,t'}(v) - \hat{\mu}_{k,t,\tau}(v))^2$$

$$\tilde{M}_{k,t} = \max_{v \in nodes} \tilde{\psi}_{k,t}(v) \quad (2)$$

In fact, in the formula 2, the centrality is standardized with respect to previous values of a sliding window. The size of the window is $\tau$. Therefore we compute for each node the size of the subgraph which contains the neighbors at a maximal distance $k$. Then we calculate the common average value during the sliding window: $\hat{\mu}_{k,t,\tau}(v)$. Then, the variance is computed: $\hat{\sigma}_{k,t,\tau}(v)$. Therefore, each node have an associated standardized value for the centrality which is $\tilde{\psi}_{k,t}(v)$. The standardized locality statistics is the maximum value between all $\tilde{\psi}_{k,t}(v)$. Nodes which tend to remain constant will have a low value. In figure 5, the interesting plot for k = 2 shows that for example between time instants 4 and 5 when the graph does not change, the associated value decreases quickly. This is due to the low value of $\tau = 5$.

When central nodes are extracted, node g becomes the only central node at time 4, showing that node c was only central at the beginning. Thus, the importance of c is lowered over time and a new node g can become an important node.

Besides, there is a peak at the beginning of the curves due to the initialization of the sliding window. During this stage, when a new node appears it becomes often the more central node or at least one of the highest central node. It is not a real problem as that the apparition of new node is an important fact. Finally, by comparing the peaks of the figure 2 and 5, there are not at the same positions because the figure 5 illustrates the dynamicity of the graph. Therefore, even if the maximum locality statistic increases during 3 time units which means that the peak is the last value, the standardized locality statistics can be a previous one if the increasing is more important at the beginning than at the end.

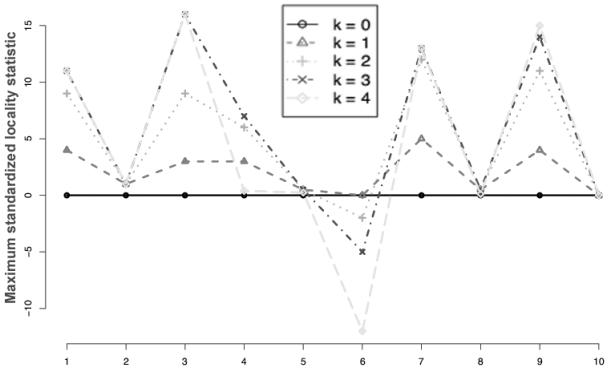



Fig. 5. Locality statistics according to time ($\tau = 5$)

### D. From graphs to network monitoring

If we consider a distributed monitoring platform, we can use a graph model to represent the relationships among the monitoring agents. Each agent is represented by a node in the graph. The major idea is to consider an arc between two nodes, if and only if the associated agents have observed a different activity. To illustrate this idea, if we consider different honeypots of a honeynet and each honeypot monitors commonly used parameters like source IP addresses, source ports, destination ports, an arc between two nodes exists if both agents have a little overlap in the observed parameters, they should be linked and it will be highlighted by the locality statictics.

## III. Data description

### A. Network telescope

The principle of network telescope is simple. A monitoring device saves all incoming traffic to a specific range of IP addresses. In fact, these addresses are unused and cover a range which is generally a subnetwork of consecutive addresses. The main characteristic of a telescope is its size which is generally huge. It is possible to create more interactive network telescopes which emulate diversified services like shown in [5], but in our case the telescope is totally passive and just records the incoming packets. Because the monitored addresses are normal and are secret, an attacker is unable to know these ones and attacks can be targeted against these.

We used in our work data from the telescope developed in the CAIDA project [6]. The monitored addresses form an A class network and the number of addresses is $2^{24}$. This huge telescope gathers data from a fraction of $\frac{1}{256}$ of the Internet. Only backscatter packets are captured by this telescope. Backscatter packets are generated indirectly by a denial of service attacks and for a comprehensive overview, the reader is referred to [7]. Basically, a backscatter packet contains an the ack field set as it is a response. The basic scenario is as follows: an attacker does a SYN flooding of a victim in order to force the victim to reply to each packet. The attacker can spoof the source IP addresses in order to hide her identity and avoid additional bandwidth consumption on her side. The victim of the denial of service attack replies to the spoofed addresses and these replies are called backscatter packets. The figure 6 shows a simple scenario where an attacker spoofs three IP addresses but only one is assigned to a real and legitimate network interface. The others are a part of the addresses of a telescope which collects these backscatter packets. Therefore the response can be captured by the telescope. The assumption of that the telescope monitors only backscatter packets is limited because some of this packets can be generated by an ACK port scanning. Moreover, the telescope stores also the ICMP response which can be due to a ICMP echo request for instance.

During our analysis, only the period from 26 to 36 August 2004 is studied on a hour by hour basis. About 460 millions of packets have been gathered during this period corresponding to 24.1 GB of data. For more information about the data, please refer to the table I.

| | | Network Telescope |
|---|---|---|
| #Observed IP source addresses | | 116 777 216 |
| Number of incoming packets | 2004/08/26 | 52 784 835 |
| | 2004/08/27 | 88 411 307 |
| | 2004/08/28 | 142 096 855 |
| | 2004/08/29 | 77 094 947 |
| | 2004/08/30 | 51 850 438 |
| | 2004/08/31 | 45 742 568 |
| Number of unique source IP addresses | 2004/08/26 | 171 257 |
| | 2004/08/27 | 244 643 |
| | 2004/08/28 | 241 883 |
| | 2004/08/29 | 242 491 |
| | 2004/08/30 | 231060 |
| | 2004/08/31 | 246 982 |
| Size of data | 2004/08/26 | 3,8 MB |
| | 2004/08/27 | 6,3 GB |
| | 2004/08/28 | 1,5 GB |
| | 2004/08/29 | 5,5 GB |
| | 2004/08/30 | 3,7 GB |
| | 2004/08/31 | 3,3 GB |

TABLE I
Global information about the telescope data

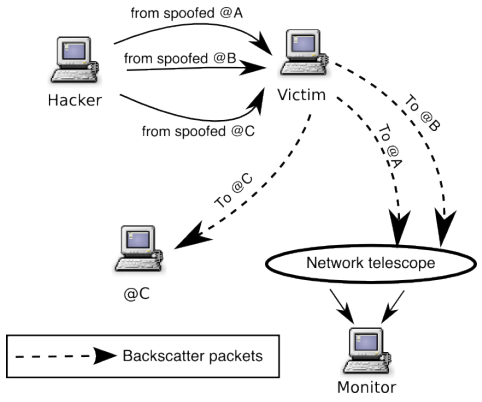



Fig. 6. Backscatter principle

### B. Honeynet

A honeypot is described in [8] as an environment where vulnerabilities are deliberately introduced. Malicious in-

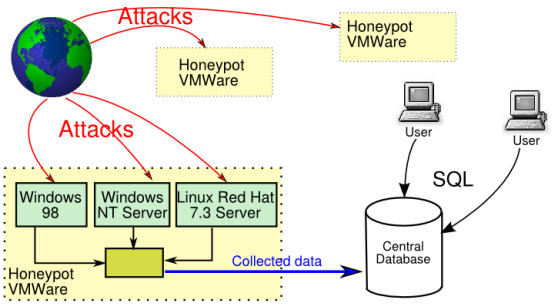



Fig. 7. Leurre.com honeynet architecture

truders are lured into attacking such a system and providing useful information to security officers and researchers. Such information typically includes details about the source of the attack, temporal patterns in this activity and the tools used during and after an attack. More recently, honeypots and honeynets have been used to observe the behavior and spreading of automated malware like worms and autorooters. The basic idea behind a honeypot is that a vulnerable system is simulated to the outside and more or less simulated services are exposed in order to achieve an interaction with the attacker (or automated malware). The degree of interactions can vary from simple and low interaction honeypots (like the ones described in [9]) and up to complete worm capturing architectures (the mwcollect project is a very good example of such an architecture), or even human driven high interaction honeypots. The first description of such a honeypot, although not named as such, can be found in the [10], where a human network administrator manually emulates a rogue vulnerable system in order to study an intruder.

However, only one honeypot is not sufficient for a sound analysis at a Internet scale level. Several honeypots can be grouped into a network which is called an honeynet. In this case, all honeypots share their informations with others and they are dispersed over all the Internet.

For our work, the honeynet of the Leurre.com project was used. This network consists of 129 individual systems run by 43 honeypots. Each individual honeypot uses 3 distinct IP addresses and emulates 3 different operating systems (one operating system per address : Windows NT server, Windows 98, and Linux Red Hat 7.3). The number of monitored IP addresses is $3 \times 43$ which is very lower than for the telescope. However, the IP addresses are well distributed in IP domains contrary to the telescope whose the data can be biased by attacks targeted specific IP domains. Data is collected locally and centralized in a database. There are low interactions honeypot and the collected data are stored in a central Database accessed by SQL request as you can see on the honeynet description in the figure 7.

The period of our study covers the data from May to December 2004 and includes more than 11 millions IP packets. The period is sliced into weeks. The table II gives the exact details about the analyzed data.

| | | Honeypot |
|---|---|---|
| #monitored addresses | | 129 |
| Number of incoming packets | 05 | 475 519 |
| | 06 | 1 211 820 |
| | 07 | 1 495 525 |
| | 08 | 1 821 534 |
| | 09 | 1 371 280 |
| | 10 | 2 317 525 |
| | 11 | 2 292 083 |
| | 12 | 1 451 770 |
| Number of unique source IP addresses | 05 | 18 392 |
| | 06 | 39 419 |
| | 07 | 34 011 |
| | 08 | 49 076 |
| | 09 | 60 666 |
| | 10 | 77 032 |
| | 11 | 84 485 |
| | 12 | 82 500 |
| Size of data | 05 | 69 MB |
| | 06 | 176 MB |
| | 07 | 217 MB |
| | 08 | 264 MB |
| | 09 | 199 MB |
| | 10 | 337 MB |
| | 11 | 333 MB |
| | 12 | 211 MB |

TABLE II
GLOBAL INFORMATION ABOUT THE HONEYNET DATA. THE MONTHS ARE REPRESENTED IN NUMBER (05, 06, 07...)

## IV. INTERSECTION GRAPHS APPLICATION

In this section, the intersection graphs method is applied to the previously described monitoring platform : honeynet and network telescope. Several aspects will be studied: source IP addresses, source ports, attack tools used, misconfigurations and targeted services.

### A. Source IP addresses

*1) Honeynet:* The goal of our first analysis is to analyze the distributed views of the honeypots with respect to the source IP addresses and identify the ones that stand out of the crowd, ie that capture suspect source addresses that are not captured by other honeypots.

Nodes represent the different honeypot platforms. For each nodes, the sets with captured source addresses are compared. Two nodes are linked only if the intersection between the corresponding sets represents less than a threshold $\alpha$ of the union of addresses. If nodes were really distinct, there would be more and more arcs and the locality statistic would increase. The normalized locality statistic permits to detect when the topology changes significantly and to detect the honeypots which are responsible for the new maximal locality. These central honeypots could be considered as interesting because they detects particular source IP addresses.

Determining the threshold is not easy. In fact, it depends on the objective. For example, some characteristics (like source IP addresses) are more variable and so normally the thresholds will be very low because we should not see the same value many times. Other characteristics have often the same value as the targeted port of an attack (like web servers). Therefore, the conclusions have to consider these thresholds in order to say if the different nodes see really

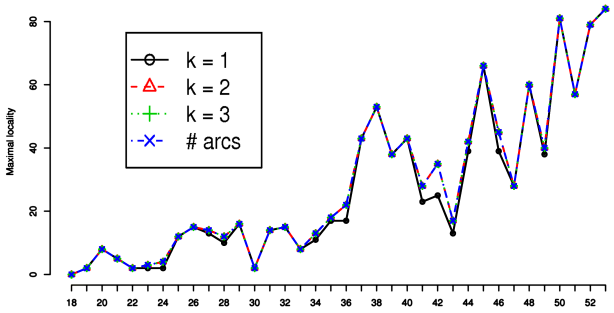



Fig. 8. Maximal locality, (shared addresses *alpha* $\leq 0.25\%$), x-axis are the week numbers

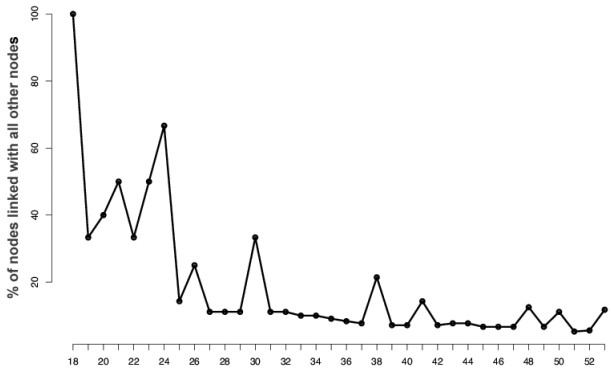



Fig. 9. Number of central nodes with the maximum locality for the honeynet

different things or not. Moreover, tuning them to obtain result similarities between the Honeynet and the telescope is a good way to evaluate how these monitoring platform kinds are different by comparing the thresholds.

After some tests, for small thresholds $\alpha$, the plots tend to overlap and a good setting of this value is 0.25%, where only few points are not overlapped. The figure 8 shows the maximum locality and the total number of arcs in the graph and all the curves are very similar and close to the number of arcs. It means that for $k = 1$, a node is linked to each other one except for few cases which means that at least one honeypot is very different in terms of observed IP addresses. Therefore, the figure 9 shows the number of nodes with the maximal centrality and so the ones which are linked with each others. There are some peaks but the curve decreases and tends to the value of 10%. Obviously, the corresponding honeypot platforms can be known and this information is useful for improving the analysis of honeypot data by limiting the amount of its.

The figure 10 represents the standardized locality with $\tau = 5$ weeks. Using the method of the intersection graphs, we can observe that when the value of the maximum standardized locality statistics is low, the topology of the graph is constant, while high values indicate major topology changes. The plots are generally overlapping and there are 8 peaks. The concerning central nodes have been extracted and some nodes (6) appear several time. Therefore, the 6 honeypots corresponding to these nodes are very different with respect to the remaining ones.

*2) Network telescope:* The goal of this study is similar to the previous honeynet analysis. We wanted to detect if a part of a telescope detects source IP addresses which are not detected by other parts. The range of IP addresses monitored is sliced into several /16 subnetworks. Because of the size of the telescope is a /8, we consider $2^8 = 256$ subnetworks. This division is logically equivalent to a distributed monitoring model described in figure 1. When this model is instantiated, we obtain the architecture illustrated in figure 11. In fact, each subnetwork of the telescope is considered as an entity for which there is one monitoring agent.

The nodes are the subnetworks and two nodes are linked if the intersection of their source IP addresses is less than

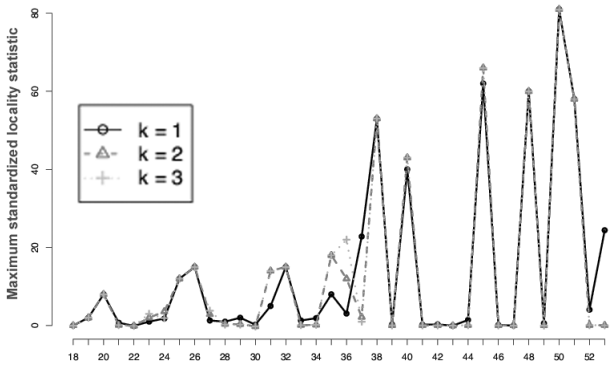



Fig. 10. Honeynet source IP addresses analysis - Standardized locality with $\tau = 5$ (shared addresses $\leq 0.25\%$)

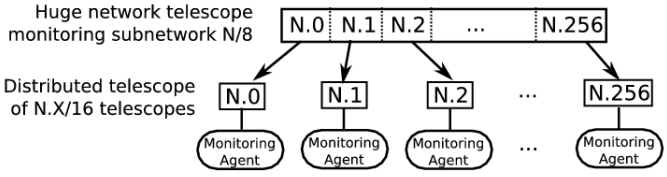



Fig. 11. A distributed telescope

a threshold $\alpha$ of their union. If the subnetworks were really different in term of observed source IP addresses, a lot of links would appear and the locality statistic would increase.

We have tested threshold values of 5% and the maximum locality statistic is always 0 except for the first hour which is probably due to a lack of data at the beginning of the capture (because the August 26 is the first day of August for which we have data). A threshold value of 5% is low but we also intended to compare honeynet (threshold was 0.25%) and telescopes and we concluded that there is a high redundancy of information in the telescope case.

*B. Source ports*

A second goal was to detect plateforms which observe port source addresses that other honeypots have not observed. Only packets with both flags SYN and ACK were considered. This kind of packets are in fact backscatter packets. In this particular case, the perceived source ports are in fact ports which have been attacked with IP spoofed packets. Thus, this study is relevant to attacked ports.

*1) Honeynet:* A node in the graph is a honeypot platform and similar to the previous case, an arc links 2 nodes if the set intersection of their source ports is lower than a threshold $\beta$ of the union of the source ports. Therefore, if honeynets were different, the locality statistic of these nodes would increase and the plots of the maximal locality statistic would show it. The plots corresponding to the unnormalized maximal locality statistic are represented in figure 12 (for a threshold of 10%) and respectively in figure 13 for a threshold of 25%. A threshold of 25% implies that the number of arcs is higher and the different plots are not overlapping. However, the aim of our work was to detect platforms that are different and a 25% threshold means that we consider 2 honeypots different even if they share one quarter of their source ports. If we consider both thresholds 10% and 25% we observe that the peaks in both plots are located at the same time instants and such the threshold of 10% is sufficient for detecting topology changes. The plots of the maximal centralized locality statistic with a sliding window size of 5 look like the figure 12 and 13.

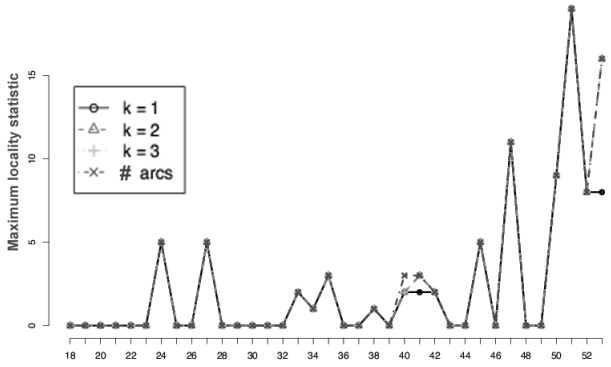



Fig. 12. Honeynet source ports analysis - locality statistic (shared ports $\leq$ 10%)

If we consider now the plots for a threshold of 10%, at many time instants the number of arcs is 0. In these cases the honeypots share more than 10% of the detected attacked ports. The ports are coded with 2 bytes in the TCP header and so $2^{16}$ ports are theoretically possible. However only few ports out of this large pool are really used and correspond to known deployed services.

Although several peaks are visible, the maximum locality is not very high and it's probably due to the low quantity of data at the honeynet. For instance,if the ports detected would be completely different between the 43 honeypots, the number of arcs would be : $\sum_{i=43-1}^{1} i = 946$.

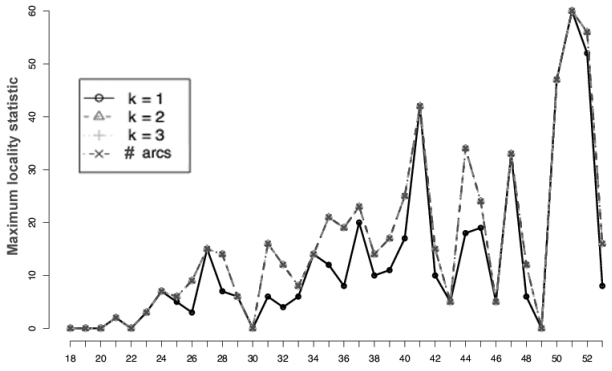



Fig. 13. Honeynet source ports analysis - locality statistic (shared ports $\leq$ 25%)

*2) Network telescope:* The packets that have been captured by the telescope are only backscatter packets and so the source ports of these packets are in fact attacked ports. It's interesting to study them in the same manner that we have done it for the honeynets. The difference here is that the nodes are the subnetworks of size /16 of the range of monitored IP addresses. Our goal was to detect if sometimes, only particular ports were attacked.

Using a threshold of 5% we obtained the plots shown in figure 14. The number of arcs and the locality statistic is close to 0. The source ports shown by the different subnetworks are the same. The conclusion is the same as for the honeynet case : attackers attack frequently the same ports and the telescope can detect this phenomena.

A peak appears clearly on the figure 14 and in fact there are 3 subnetworks detecting unusual source ports. This is opposed to the honeynet case for which a peak is not always significant due to a low amount of data. Because a telescope monitors a fraction of $\frac{1}{256}$ of the Internet, a high peak like its shows a real specific phenomena at this time and this peak is a proof of attacks on original ports.

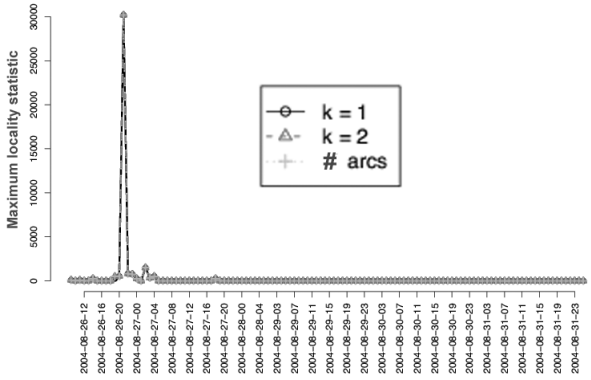



Fig. 14. Telescope source ports analysis - locality statistic (shared ports $\leq$ 5%)

### C. Attack tools

A TCP session is established thanks to the 3-way handshake. First the initiator sends a packet with flag SYN and a random sequence number (also called Initial Sequence Number -ISN). The correspondent acknowledges the packets with an acknowledgment number equal to the previous sequence number + 1. Finally the initiator acknowledges this reply. Some attack tools use always the same sequence number or do not use a good (high entropy) random number generator. Consequently, the acknowledgment numbers are either always the same, or depend on the use of a specific exploit code. We looked if the same attack tool was used to attack different computers and for this work we considered also the the backscatter packets (replies of attacks). In this experiment, only the honeynet is considered.

In this case, the construction of the graphs consists in considering nodes as honeypots and two nodes will be linked if they share more than a threshold of the union of their observed acknowledgment numbers. Using a threshold of 90% the plots are given in figure 15. In general the acknowledgment numbers are different between platforms because the number of arcs is low. This is due to the diversification of the attack tools.

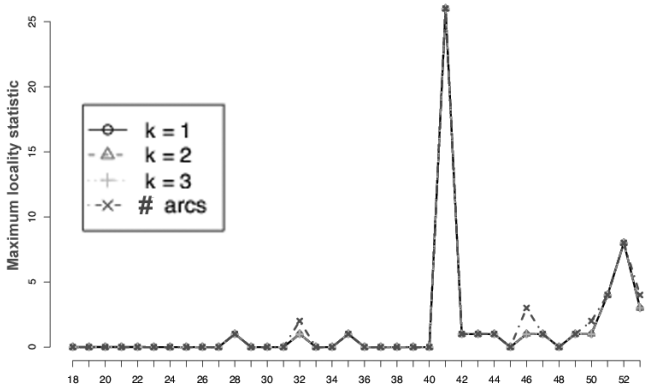
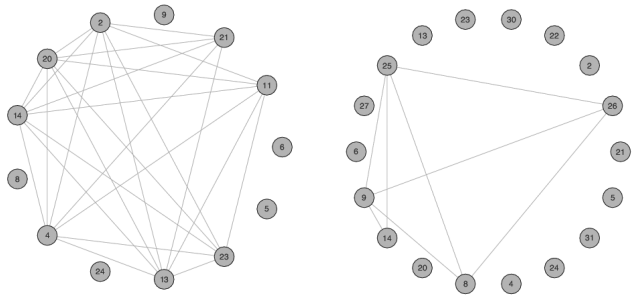



Fig. 15. Honeynet acknowledgment numbers analysis - locality statistic (shared acknowledgment numbers $\geq 90\%$)

Two peaks are clearly visible and in these case the plots are overlapping. This shows the presence of one or central honeypot linked with all others. Using the standardized locality statistic with a sliding window size of 5, the obtained plots are similar because the standardization is made thanks to previous values, which are mostly equal to 0. The figure 16 presents the graphs of weeks 41 and 52 corresponding to the peaks. In the figure 16(a), many nodes are linked with many others. A lot of honeypots have detected about the same acknowledgment numbers (threshold $\geq 90\%$) and the use of the same attack tools is undeniable. However for the second peak in week 52, (shown in the figure 16(b)) the picture is totally different and only some honeypots are concerned. In this case, this is probably due to a same attack tool with a bad random numbers generator which implies that the same generated number is used several times and detected by different honeypots.



(a) week 41 (b) week 52

Fig. 16. Intersection graphs for acknowledgment numbers shown by the honeynet

### D. Detecting misconfigurations

During our previous analysis, many source IP addresses were invalid like many local addresses. It can be due to some attackers but smart ones prefer to use valid addresses in order to be undetected. Therefore, most of them can be considered as misconfiguration problems on user computers or at the Internet service provider

*1) Sources:* There are many types of addresses that are dedicated to specific use and that shouldn't be use on Internet. The table III gives a summary of such addresses as well as their target deployment usage. However, we were amazed by the large quantity of observed IP addresses that should in theory never appear on the Internet. Several factors jointly produce them: misconfigured enterprise routers/firewalls, missing ISP level ingress/egress filtering and maybe defective devices.

| Range | Description |
|---|---|
| 10.0.0.0 → 10.255.255.255 | Class A private addresses |
| 172.16.0.0 → 172.31.255.255 | Class B private addresses |
| 192.168.0.0 → 192.168.255.255 | Class C private addresses |
| 224.0.0.0 → 239.255.255.255 | Class D multicast addresses |
| 240.0.0.0 → 255.255.255.255 | Class E addresses reserved for experimental use |
| 127.0.0.0 → 127.255.255.255 | Loopback addresses |
| 0.0.0.0 → 0.255.255.255 | addresses of network 0 (class A) |
| 169.254.0.0 → 169.254.255.255 | addresses of DHCP client which can't obtain an address from the server |
| 192.0.2.0 → 192.0.2.255 | Loopback addresses |

TABLE III
Abnormal source addresses on Internet

The left barchart of figure 17 shows the proportion (per 100 000) of the different type of abnormal addresses

considering unique IPs in comparison with the total number of unique IPs for the observed days in the case of the telescope. This graph allows to observe both the main types of abnormal addresses and their corresponding global proportion.

There is a category which is about constant (colored in black). It is the proportion of network 0 addresses (class A). Normally 0.0.0.0 can be used only as source broadcast address on local segments but not on the global Internet. However the global proportion increases significantly from June to August with peaks in June, at the end of August and the beginning of September. Very strangely is also the apparition of multicast addresses as source addresses. Multicast addresses can be only used as a destination address and will never appear as source addresses. Moreover this increase in abnormal addresses is also due to private IP addresses used in outgoing reply packets. These packets are received by the telescope (and for these packets the source appears to be a private IP address).

An attacker is able to forge such packets thanks different software like [11] but as previously introduced, discovering the attacker is easier in this case. Moreover, these packets are backscatter packets which means that main of them are responses from victims which don't forge the packets, such that we can safely assume that the majority is not malicious. The most probably source of these packets are misconfigured routers/firewalls/NATs. This increase can be also caused by an ISP deploying some new policy based routing rules, which were misconfigured. The concerned computers are connected to Internet but don't receive the responses of their own requests. Another justification of the apparition of private addresses (the class C for instance, which are generally used by home users) are a definite evidence of misconfigured network devices. However, the main issue is that the ISP does not block these addresses. The observed results can be generalized beyond the simple observed traffic as follows:

$2^{24}$ : *IP addresses monitored by the telescope*
$2^{32}$ : *all possible IP addresses*
*Assuming that about* 75% *of addresses are used on Internet*
$y$ : *number of IP addresses concerned by an analysis*
$x$ : *estimation of the number of IP addresses corresponding to the same analysis for the whole Internet*

$$x = \frac{2^{32} * 0.75 * y}{2^{24}}$$

This type of generalization can be applied to all the observed data in this paper

We performed a similar analysis with the data from the honeynet (at the right on the same figure 17) but in this case, a bar represents a month period. The results show a different pattern than the backscatter analysis. First the graph shows two peaks but not at the same time. The first in May and the second in July. The usage of private class IP addresses is also significant and the explanation might be the same i.e. the misconfiguration of local network and providers that don't do ingress filtering. However the main type of abnormal IPs is the range of addresses automatically assigned by a computer when the DHCP server don't respond to its request for obtaining an address. The cause is probably due to local networks with a non valid configuration of the DHCP service.
For comparing the two traces, we had to compare data from backscatter traffic observed from the telescope with data (directly incoming and backscatter) from the honeynets. We could not rely entirely on only the backscatter traffic from the honeynets due to the lack of massive datasets.

*2) Open Windows specific ports:* The Windows operating systems uses a series of defaults ports for its proprietary network protocols: ports 137, 138 and 139. The Netbios service is designed for sharing resources on a local network and this port is not only useless on the Internet but represents one major entry point for malware and malicious intruders. Moreover the port 445 is also a dangerous port because it is used for file sharing and many worms (Sasser and mutants exploit). To prevent these attacks, these ports should be filtered by a firewall.

Considering the telescope, the figure 18 shows the number of unique IPs with an open port per 100 000 unique IPs. Receiving a backscatter response of a given port means that the port was open during the connexion of the attacker performing the denial of service attack. The ports 137,138 and 445 seem to be protected even if there is a little peak for the port 445 in November. However it's clear that the port 139 is less filtered as we can see on the several peaks of the graphs. It seems that in 2004, professional networks and home computers were generally protected by firewalls contrary to some years before, but this is seen through traces of Denial of service attacks. Since, most victims are typically either enterprises or blackhats waging Internet wars, these low numbers are justified.

The honeypot data contains only one IP address having the port 139 open, such that the use of honeypot is not a good way to detect this kind of misconfiguration. Only a telescope with a large range of IP can efficiently detect it. However you can notice that the only visible port is also the one which is the most frequently observed as opened by the telescope.

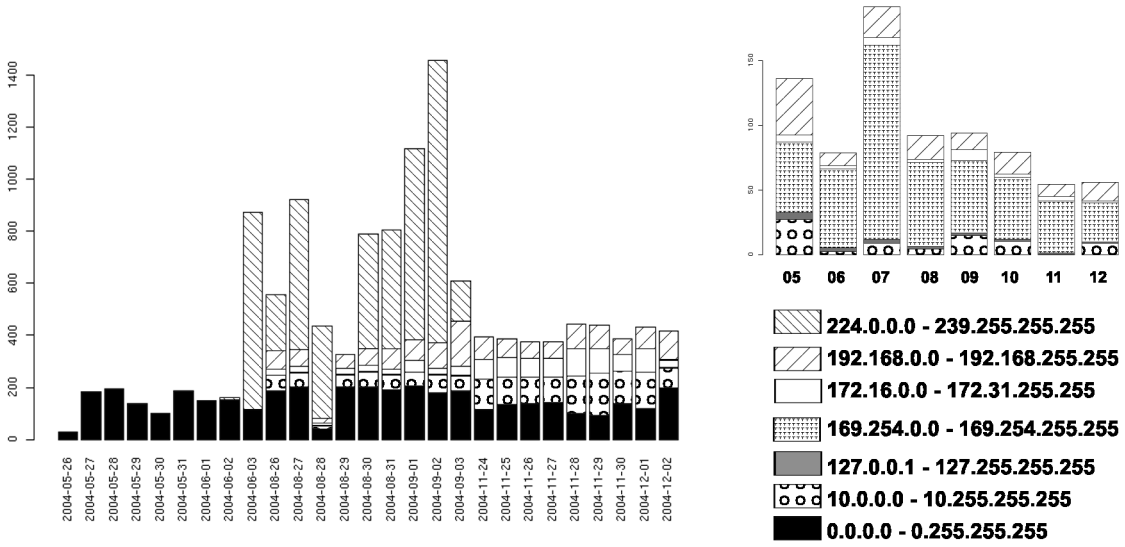



Fig. 17. Number of unique IP addresses of the different categories of abnormal IPs per 100 000 unique IP addresses. (Left : backscatter data, right : honeypot data by month)

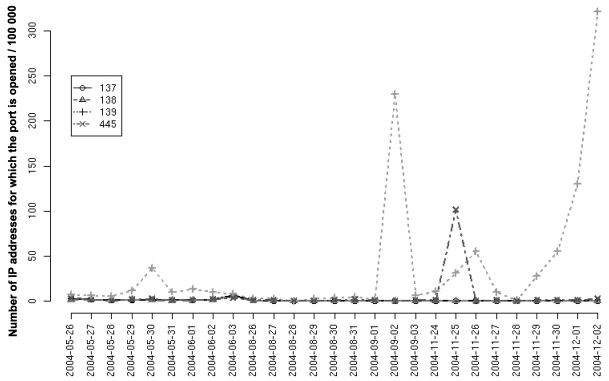



Fig. 18. Number of unique IPs with an open port per 100 000 unique IP addresses and according to each specific windows port. The chart represents the backscatter data of the telescope.

*3) Analysis of ICMP 'Destination unreachable' message:* When a host connects to another host which is not available, an ICMP message is sent to the source with the type 3 equal to 'Destination unreachable message'. An additional code [12] is also used to provide additional information. We analyzed the following 8 codes in our work:

- 0 : net unreachable
- 1: host unreachable
- 2 : protocol unreachable
- 3 : port unreachable
- 4 : fragmentation needed and don't fragment was set
- 9 : communication with destination network is administratively prohibited
- 10 : communication with destination host is administratively prohibited
- 13 : communication administratively prohibited

Polite firewalls will typically answer with codes 9, 10 or 13 to show that a device or service is filtered. Although such information can be very helpful when troubleshooting a network like detecting firewall misconfigurations, it can leak information about existing devices/open ports to an attacker and could determine him to try more advanced reconnaissance techniques. Less polite firewalls, configured by more security conscious network managers might directly reply with TCP packet whose the RST bit is set.

The figure 19 shows the evolution of the ICMP type 3 message codes. The left graph is about the telescope and highlights clearly a main change between October and November. First of all, the code 13 decreases much which can be due to a significant change in the behavior of network administrator which prefers to limit the revealed information. Moreover, the code 3 becomes the most popular code. This code means that the port is unreachable and so that the host exists. Therefore this change shows that the attacks are much well targeted from November and most of them are port scanning. The bars about honeypots is the right one on the figure 19. Once again, there is a change but it is smoother than for the telescope with the same observation as before, i.e. a decrease of code 13 and an increase of code 3. Finally, the main difference is that the honeynet detects the change earlier than the telescope.

### *E. Most attacked services*

A natural question is related to which services are the most attacked services. We did this analysis on backscatter data for the different monitoring methods. Therefore, the packets reflect denial of service attacks. There are four main services which are attacked:

- The most attacked port and consistently ranked number 1 over all this period is port 80: it seems that

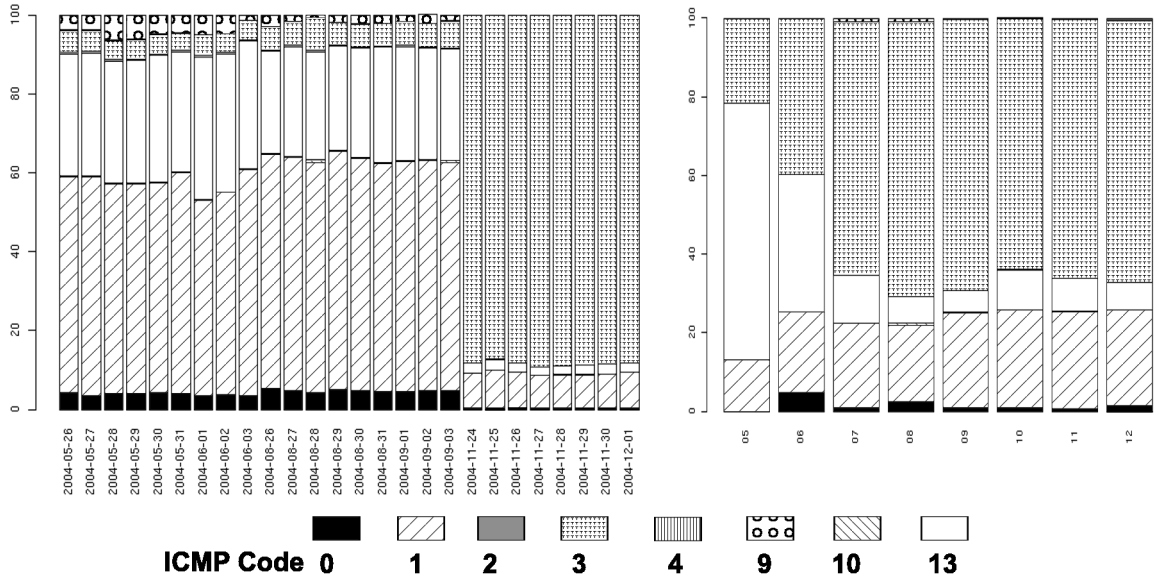



Fig. 19. Proportion of the different ICMP codes for the icmp type 3 (Destination unreachable). Backscatter telescope data are represented at the left and honeypot data is at right

web servers are the major target of denial of service attacks,

- port 6667 shows up frequently in the attacks. This port is typically used for IRC talks (or IRC anonymizing proxies like psyBNC). We suppose that these attacks are targeted at specific servers and can be associated to Internet war games waged to take the control of a a IRC channel,
- Name Servers (port 53) are also attacked (although to a lesser extend than IRC),
- Attacks against BGP routers (port 179) are also highly interesting and can be observed, since these attacks aim at either de-connecting a network domain, or can serve as preliminaries for a routing prefix hijack.

The table IV compare the most attacked services between the telescope (3 days) and the honeypot for May. Then we can see that the overlap of the ports is small : only the port 80. However if we consider table V in September, the overlap is totally different because 7 ports appear in the Honeypot and in the Telescope data. To conclude, even if sometimes, the two methods allow to get the same results, it appears that the results can be different and therefore the methods can be considered as complementary.

Moreover, in these tables (IV and V) an interesting fact is to have the port 7000 which is known as a backdoor. In the table VI the ports which are in the most attacked services with known vulnerabilities are listed. The vulnerabilities are common backdoors or ports used for the spreading of a worm. So, the attackers try also to do targeted denial of service attacks to open ports which are not reserved for a normal service.

Thus, we can conclude that ports are opened even if no service are traditionally associated and for which a vulnerability is known.

| Port | Vulnerability |
|---|---|
| 1011 | Augudor |
| 1025 | Spybot |
| 1433 | Spybot |
| 6000 | Lovgate |
| 7000 | SubSeven |
| 7001 | Freak88 |
| 7300 | NetMonitor |
| 8000 | Gaobot |

TABLE VI
Some services which are in the most attacked services and which present known vulnerabilities

## V. Related works

The honeypots and honeynets are presented in [8] where general definitions and platform description are are given. That reference containts also results about the localization of the attacks or the observation of worm spreading in the context of the Leurre.com project. [13] is also an introduction to the different kinds of honeypot and highlights the different advantages of them and less frequently addressed question like legality or privacy problems.

In [14], the same authors propose a more elaborated method to study the data of the honeynet. In fact, the authors cluster the different captured network packets using the Levenshtein distance in order to group packets which are due to the same attack.

In [9], the goal of the paper is to determine the degree of the interaction of a honeypot needed to collect useful data, while in the same time avoiding to collect too much useless data. Even if it seems that a low level interaction honeypot is sufficient, the use of a high level of interaction degree is needed to correctly configure the low level interaction.

| Honeypot May | | Telescope 2004-05-26 | | 2004-05-27 | | 2004-05-28 | |
|---|---|---|---|---|---|---|---|
| 80 | 35 (63.64) | 80 | 734 (7.61) | 80 | 973 (10.03) | 80 | 980 (16.27) |
| 6667 | 5 (9.09) | 21 | 15 (0.16) | 21 | 17 (0.18) | 139 | 14 (0.23) |
| 3389 | 3 (5.45) | 6667 | 15 (0.16) | 4662 | 15 (0.15) | 21 | 13 (0.22) |
| 7000 | 3 (5.45) | 139 | 13 (0.13) | 139 | 13 (0.13) | 22 | 11 (0.18) |
| 1107 | 1 (1.82) | 1002 | 12 (0.12) | 25 | 11 (0.11) | 113 | 10 (0.17) |
| 1205 | 1 (1.82) | 22 | 10 (0.10) | 8080 | 11 (0.11) | 25 | 10 (0.17) |
| 1214 | 1 (1.82) | 8080 | 10 (0.10) | 110 | 10 (0.10) | 8080 | 9 (0.15) |
| 1235 | 1 (1.82) | 110 | 9 (0.09) | 113 | 10 (0.10) | 443 | 8 (0.13) |
| 1254 | 1 (1.82) | 113 | 8 (0.08) | 135 | 10 (0.10) | 110 | 6 (0.10) |
| 1271 | 1 (1.82) | 111 | 6 (0.06) | 22 | 8 (0.08) | 178 | 6 (0.10) |

TABLE IV
THE MOST ATTACKED SERVICES DURING MAY WHICH HAVE SENT SYN/ACK. THE FIRST NUMBER IS THE PORT AND THE SECOND THE NUMBER OF UNIQUE IP ADDRESSES WHICH ARE CONCERNED. THE NUMBER BETWEEN PARENTHESIS IS THE PERCENTAGE ACCORDING TO ALL UNIQUE COUPLE IP ADDRESS - OPEN PORT

| Honeypot September | | Telescope 2004-09-01 | | 2004-09-02 | | 2004-09-03 | |
|---|---|---|---|---|---|---|---|
| 80 | 116 (50.88) | 80 | 956 (14.89) | 80 | 1100 (19.66) | 80 | 508 (17.69) |
| 7000 | 49 (21.49) | 7000 | 37 (0.58) | 139 | 413 (7.38) | 7000 | 24 (0.84) |
| 7100 | 11 (4.82) | 7200 | 13 (0.20) | 7000 | 30 (0.54) | 7100 | 21 (0.73) |
| 22 | 9 (3.95) | 7100 | 12 (0.19) | 7100 | 22 (0.39) | 7200 | 18 (0.63) |
| 7200 | 7 (3.07) | 21 | 10 (0.16) | 7200 | 18 (0.32) | 3389 | 12 (0.42) |
| 7090 | 6 (2.63) | 25 | 9 (0.14) | 21 | 14 (0.25) | 21 | 11 (0.38) |
| 3389 | 4 (1.75) | 22 | 8 (0.12) | 3389 | 11 (0.20) | 8080 | 8 (0.28) |
| 21 | 3 (1.32) | 443 | 8 (0.12) | 22 | 10 (0.18) | 139 | 5 (0.17) |
| 113 | 2 (0.88) | 8080 | 8 (0.12) | 8080 | 8 (0.14) | 6000 | 5 (0.17) |
| 6667 | 2 (0.88) | 3389 | 7 (0.11) | 25 | 7 (0.13) | 1524 | 2 (0.07) |

TABLE V
THE MOST ATTACKED SERVICES DURING SEPTEMBER WHICH HAVE SENT SYN/ACK. THE FIRST NUMBER IS THE PORT AND THE SECOND THE NUMBER OF UNIQUE IP ADDRESSES WHICH ARE CONCERNED. THE NUMBER BETWEEN PARENTHESIS IS THE PERCENTAGE ACCORDING TO ALL UNIQUE COUPLE IP ADDRESS - OPEN PORT

Network telescopes have been the focus of several research works. In [15], the authors assume a simplified model and propose a simple formula to compute the probability of observing a denial of service attack with a telescope. An updated result in [16] shows with another telescope that the previous model is to simple and that spoofed addresses are not uniformly randomly generated. An interesting work is presented in [15] and leads to the evaluation of the the aggressivity of denial of service attacks. Finally, the authors in [5] propose to use high interactive telescopes with emulated services in order to learn more application specific attacks. Network telescopes are also name darknets and the authors in [17] introduces the greynets which are small telescopes with some unused addresses scattered within a set of used IP addresses. They evaluate their efficiency depending on the number of probes, ie. the number of unused addresses.

The reference book in system administration [7] includes several examples on the use of graphs and the centrality of a node by using eigen vectors. The first work applying these techniques to security monitoring is [3], where the email exchanges in the enron database is analyzed in order to prove that that some employees had inside level information on the fraudulos management. The same method was applied to network security in [18] for end user level activity profiling. The goal was to detect if the websites visited by employes can be associated to a normal type of behavior and how malware spreading can be detected if abnormal activity is observed.

## VI. Conclusion

In the work presented in this paper we were challenged by several research questions. Firstly, we needed a generic method to analyze both telescope and honeynet data. The main goal was to compare these two ways of gathering malicious network traffic. While a telescope monitors a large range of consecutive IP addresses, the honeynet monitors a limited set of IP addresses dispersed over the Internet. The amount of data is much higher for the telescope if compared to the honeyet. A second contribution of our work was to assess the utility of each method to collect network information. For instance, we have observed that a honeynet is sufficient for learning the distribution of source addresses, contrary to telescope for which a high redundancy might become an obstacle in the analysis.

On the other hand, both methods did provide similar results about the services/ports that are attacked, but the telescope is superior when detecting less frequently attacked services. This is quite obvious, due to the much higher data volume. Concerning the used attack tools, the honeynet permitted to show that these are more and more diversified and sophisticated. Regarding the misconfigurations, the network telescope and the honeynet are about equivalent for most of the studied cases.

The central concept underlying our work are the intersection graphs. These graphs have not been used widely

in the field of network security. The advantage of this method is that analyzing aggregated data is possible by considering the curve of the maximum locality statistic and the maximum standardized locality statistics. This is possible because these plots are closely related to the trend of the variation in the topology of a graph. This method allows also to identify the nodes, which are important in the graph. Importance can be assimilated with monitoring agents that observe unusual network activities. The main difficulty encountered during our work is related to processing such large datasets: data counts to more than 200 GB and this task pushed our computational resources to their limits. Future work will address more advanced data mining and statistical analysis techniques.

Several papers individually analyzed either telescope data or honeynet data, but none had tried yet to compare these two data source simultaneously. Our work is to the best of our knowledge the first attempt to compare the two methods over the same time period.

**Acknowledgment** We thank Fabien Pouget and Marc Dacier from the Leurre.com project for their collaboration in the honeypot project. Moreover, we would like to thank Emile Aben and Colleen Shannon at CAIDA for granting us access to the telescope backscatter data.

## References

[1] J. Francois, R. State, and O. Festor, "Large scale activity monitoring for distributed honeynets," in *ICIMP '07: Proceedings of the Second International Conference on Internet Monitoring and Protection*, 2007, p. 6.

[2] R. State, J. Francois, and O. Festor, "Tracking global wide configuration errors," in *IEEE / IST Workshop on Monitoring, Attack Detection and Mitigation*, Tubingen/Germany, 09 2006, H.: Information Systems.

[3] C. E. Priebe, J. M. Conroy, D. J. Marchette, and Y. Park, "Scan statistics on enron graphs," *Comput. Math. Organ. Theory*, vol. 11, no. 3, pp. 229–247, 2005.

[4] M. Burgess, *Analytical network and system administration*. John Wiley & Sons Ltd., 2004.

[5] V. Yegneswaran, P. Barford, and D. Plonka, "The design and use of internet sinks for network abuse monitoring," 2004.

[6] C. Shannon, D. Moore, and E. Aben, "The caida backscatter-2004-2005 dataset - may 2004 - november 2005, http://www.caida.org/data/passive/backscatter_2004_2005 _dataset.xml."

[7] J. Mirkovic, S. Dietrich, D. Dittrich, and P. Reiher, *Internet Denial of Service : Attack and Defense Mechanisms*, ser. Radia Perlman Computer Networking and Security. Prentice Hall PTR, december 2004.

[8] F. Pouget, M. Dacier, and H. Debar, "Attack processes found on the Internet," in *NATO Research and technology symposium IST-041/RSY-013 "Adaptive Defence in Unclassified Networks", 19 April 2004, Toulouse, France*, Apr 2004.

[9] F. Pouget and T. Holz, "A pointillist approach for comparing honeypots," in *DIMVA 2005, Conference on Detection of Intrusions and Malware & Vulnerability Assessment, July 7-8, 2005, Vienna, Austria - Also published in LNCS Volume 3548*, Jul 2005.

[10] A. R. W. Cheswick, S Bellowin, *Firewalls and Internet Security: Repelling the Wily Hacker*. Addison Wesley, 1994.

[11] G. Roualland and J.-M. Saffroy, "http://ippersonality.sourceforge.net."

[12] IANA, "http://www.iana.org/assignments/icmp-parameters," 2005.

[13] I. Mokube and M. Adams, "Honeypots: concepts, approaches, and challenges," in *ACM-SE 45: Proceedings of the 45th annual southeast regional conference*. New York, NY, USA: ACM, 2007, pp. 321–326.

[14] F. Pouget and M. Dacier, "Honeypot-based forensics," in *AusCERT2004, AusCERT Asia Pacific Information technology Security Conference 2004, 23rd - 27th May 2004, Brisbane, Australia*, May 2004.

[15] D. Moore, C. Shannon, D. J. Brown, G. M. Voelker, and S. Savage, "Inferring internet denial-of-service activity," *ACM Trans. Comput. Syst.*, vol. 24, no. 2, pp. 115–139, 2006.

[16] K. E. Giles, D. J. Marchette, and C. E. Priebe, "On the spectral analysis of backscatter data," in *Hawaii International Conference on Statistics and Related Fields*, 2004.

[17] W. Harrop and G. Armitage, "Defining and evaluating greynets (sparse darknets)," in *LCN '05: Proceedings of the The IEEE Conference on Local Computer Networks 30th Anniversary*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 344–350.

[18] D. J. Marchette, "Statistical opportunities in network security," in *35th Symposium on the interface*, 2003.

# Applicability of Host Identities to Implement Non-Repudiable Service Usage

Seppo Heikkinen
*Tampere University of Technology*
*firstname.lastname@tut.fi*

## Abstract

*In a typical roaming scenario the accounting information received from the roaming partner is expected to be trustworthy. Things like fear of losing one's reputation have been working as disincentives for fraudulent behaviour between the large operators. However, when smaller players enter the market and steps are taken towards more dynamic relationships as in the visions of ubiquitous computing environments, the need for reliable records becomes paramount. Thus, secure accounting mechanisms are needed for ensuring correct compensation amongst the interoperating partners. On top of that, the partners need to be authorised with sufficient granularity to be able to engage in the transaction in the first place. The mere authentication should not be enough.*

*In this article we present a solution concept for ensuring non-repudiation of the service usage, so that cryptographically secure accounting records can be generated, and the parties involved in the transaction make their commitments only to the resources actually consumed. The solution is based on the employment of Host Identity Protocol (HIP) and hash chains, so that we can provide a convenient binding between the identity and authorisation information. Also, in order to avoid service hijacking, mechanisms for binding this information to the actual traffic are discussed.*

**Keywords**: hash chains, host identity, non-repudiation, service

## 1. Introduction

The communication environment is changing. As the ubiquitous computing paradigms gain more momentum and the technological development allows more dynamic usage patterns and relationships, more and more small players enter the market to get their piece of the service provisioning cake. Naturally, these players want to ensure that they receive authentic users that are able to pay for the service usage. On the other hand, the players vouching for the liability of the users want to make sure that the generated expenses are within certain limits, i.e., they want to control how much risk they are willing to take on behalf of their customers. This requires measures to ensure the correct authorisation for the users of the systems.

Thus, we have service providers, who want to receive compensation for the provision of their service resources. They are complemented by the third parties, such as home operators, who help in authenticating users and ensuring that the generated costs will be covered. Finally, we have the users, who want to make sure that they receive the service that is promised and that it is correctly charged. After all, the appearance of unauthorised charges on phone bills, i.e., cramming, is not unheard of amongst the consumers [1]. Sometimes, the user may not even have clear notion about the identity of the responsible service provider, as is often the case with visited access networks, even though the access network might be in the possession of authentication material generated by the home network.

As the interaction and the established relationships are more dynamic in nature and lasting perhaps only one transaction, typical assumption that the loss of reputation is incentive to ensure the correctness of accounting records is no longer valid. Hence, we need mechanisms that create secure accounting records so that the service transaction is undeniable and authentic for the both parties of the transaction. We propose such a simple non-repudiable mechanism that takes advantage of Host Identity Protocol (HIP) and hash chains. Our focus is on the interaction of the user and the service, not so much in the negotiation between the service and the third party nor the bootstrapping of trust between the user and the third party.

HIP already provides end point authentication and simple key exchange, but it does not currently address the problem of authorisation to the sufficient detail. In order to implement the suggested Non-Repudiable Service Usage (NoRSU), one point of this article is to

discuss how to include authorisation tokens into HIP and what are the consequences. Additionally, the hash chains are employed to introduce an incremental payment solution, i.e., a chain of tokens is created by repeatedly hashing a secret seed value. Thus, the service provider is able to generate undeniable charging records and the user can be sure that the charging is based on actual use. As HIP assigns cryptographic identities to the communication end points, the tokens can be tightly bound to the actual communication. HIP also introduces a handshake procedure for negotiating and establishing a security association between the end points. For the benefit of performance this procedure can be overloaded with the compensation related information. Thus, no additional roundtrips are introduced.

This article is organised as follows. The next section discusses the related work. The third section describes the details of the proposed system and the section after that gives examples of two use cases. The fifth section discusses the limitations the implementations have to take into account and suggests ways to efficiently encode the used information in order to overcome these limitations. The sixth section analyses the solution in terms of threats that can be faced. Finally, the seventh section concludes the article.

## 2. Related work

HIP is an experimental proposal for future network architectures that introduces a new identity layer between the network and transport layers [2]. This allows decoupling the dual role of the IP addresses. That is, currently they function as end point identities and locators. In the HIP model the end points are identified by their cryptographic identifiers, called Host Identity Tags (HIT), which are derived from their public keys. This accommodates for end host authentication and simple key exchange. Thus, the parties are able to setup a security association between themselves, which can be used to protect the control information exchange. Additionally, the protection of subsequent data transport is possible with IPsec ESP [3]. Other transports can be defined, too.

HIP uses four messages in the so called base exchange to establish the identity of the parties and to create the needed keying material with the help of Diffie-Hellman key exchange (see Figure 1). For the purposes of the paper the initiator and the responder can be considered as the client and the server, respectively. Besides securing the message exchange, the protocol mitigates denial of service (DoS) attacks by introducing a puzzle scheme.

An initial proposal for including authorisation to HIP has been introduced, but that work is still very much in the draft stage and basically provides a placeholder for the certificates [4]. Like the proposal, [5] and [6] also discuss the possibility of including Simple Public Key Infrastructure (SPKI) certificates in the protocol, but do not analyse the use case thoroughly, even though [5] provides a prototype implementation adapted to grid environments. There is also a general sketch of an attachment architecture, which includes both HIP and compensation related issues in [7]. A solution employing hash chains and KeyNote credentials to implement One Time Password (OTP) coins was depicted in [8], even though without clear binding to the actual communication. Similar ideas were used to sketch a high level solution presented in [9], but it used SPKI certificates instead of KeyNote and already took advantage of HIP to ensure the binding to the actual traffic. The text presented here extends that work with additional details.

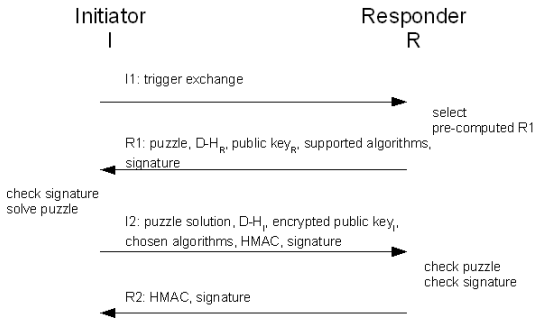



**Figure 1. HIP base exchange**

Hash chains have been used for password solutions and such one-time password authentication was suggested already in 1981 by Lamport [10]. The idea of hash chains is based on the irreversible nature of the hash functions. In other words, you are not able to calculate the source value once you have the result of the function. Hash chain is created by applying a secure hash function in successive fashion to a secret seed value and then using the values of the hash calculations in reverse order. So, it is very easy to check by applying one hash operation to the previously received value that the current value is part of the chain, but very hard to calculate additional values without the knowledge of the initial seed value of the chain. The idea behind hash chains is illustrated in Figure 2.

There exists also several other works, which have considered employing hash chains to introduce non-repudiable billing and micropayments in various

scenarios, so the concept is not new. [11] uses hash chains to implement a payment solution for ad hoc networks, but it requires the use of smart card technology to control the release of hash chain values. [12] also presents a protocol for undeniable billing with entity authentication and privacy support for mobile networks roaming access using hash chains, although it requires online interaction with the home network.

x = secret seed value

H = Hash function

n = length of chain

$H^n(x) = H(H^{n-1}(x))$

$H^n(x)$ = hash chain anchor

release chain values:

$H^n(x), H^{n-1}(x), H^{n-2}(x), ...$

**Figure 2. Idea behind hash chains**

## 3. HIP based non-repudiation

This section describes how the hash chains are integrated with the HIP base exchange.

### 3.1 General overview

Our proposal, which is based on the aforementioned HIP, works in the way depicted in Figure 3 (HIP specific parameters left out). The basic idea is to add extra information to the HIP messages in order to negotiate the usage of non-repudiative accounting within the communication. So, in a sense, we are negotiating a non-repudiation association in addition to the identity association.

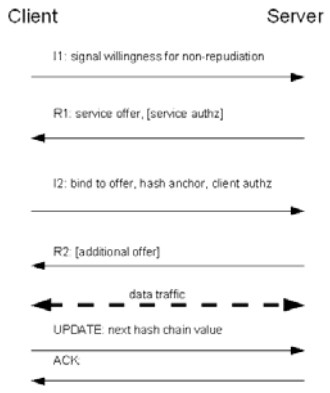



**Figure 3. Base exchange with non-repudiation enhancements**

A new HIP parameter is needed to signal the intent to access certain service with the capability of using NoRSU. Note that the server could also send this kind of indication to the client at some later point using the HIP UPDATE packets along with the corresponding offer, if the client tried to access a service, for which the server required extra accounting (provided they had an existing HIP association). Figure 4 gives an example of the said HIP parameter for indicating the use of non-repudiation for certain service and it also shows the general Type-Length-Value (TLV) format of HIP parameters (C bit denotes possible critical parameter).

| Type (15 bits) | | C | Length (16 bits) |
|---|---|---|---|
| Subtype of NoRSU | Encoding of name | | Service name length (16 bits) |
| Service name with the indicated encoding + padding if needed (variable length) | | | |

**Figure 4. New HIP parameter for signalling the use of non-repudiation for a specific service**

### 3.2. Modified base exchange messages

The tasks for individual HIP messages are as follows. The I1 message functions as a trigger message as in basic HIP base exchange [2], but with the addition of possibility to signal the capability of the client to engage in a NoRSU exchange as discussed above. The server's response, R1, contains an offer in the form of an SPKI certificate for the usage parameters, including the number of tokens needed for certain amount of time or byte count, e.g., you need one token per minute or you need one token per 100 kilobytes. That is really up to the charging scheme of the provider, but generally the value of one token should be kept small in order to avoid big losses in case of abuse. There could also be an additional advice of charge functionality telling the value of one hash chain token in monetary terms. However, this does not take into account the possibility that the tariffs are different between different parties, i.e., some client have a "better deal", because they are subscribers of some favoured organisation, for instance.

The offer is protected by a signature, which also binds the provider to the offer. The signature could also be made by a trusted third party (TTP) in order to guarantee that the server is a legitimate one, but this could also be done by using an additional authorisation certificate (see subsection 3.3). The latter case is more flexible and gives more control of the tariffs to the

service provider, naturally within the limits of the third party authorisation.

Offer should also contain validity date, so that the provider has better control of the expiration of the used offers. This allows, for example, using different tariffs at different times of the day. Naturally, if the session continues after the validity period the parties should renew their contract provided the new offer is satisfactory. It is the responsibility of the client to make sure that no additional hash chain values are sent with the assumption that the old offer is still valid. The server can in this case just stop serving the client, if there is no response to the new offer.

Note that instead of offering certain time or rate based traffic the offer could just be for the use of certain service, which could be described by a profile or a service name. This naturally requires that there is common consensus about the semantics of such profiles, but that can be agreed when establishing trust relationship, e.g., a roaming agreement, with the third party. There could also be several offers, e.g., choice between time and byte count, but this has restrictions as space is limited (see section 5).

If the offer meets the requirements of the client, it sends a response in the I2 message, which contains the signed acceptance of the offer. The acceptance is indicated by calculating a hash over the offer and signing it. Additionally, the response must contain the hash anchor value, which the server can use to validate the subsequent values, i.e., it acts as the starting point for the hash chain, which the client has created. It can also be used to identify the whole hash chain among several parallel chains.

The fourth message of the exchange, i.e., R2, can just acknowledge the validity of the offering process and, for instance, show as a summary what kind of agreement is in effect. Some advanced scenarios are possible, though. One could relate to special offers, i.e., if the customers of certain operator were allowed to get even cheaper service, R2 could contain a special offer with a reduced tariff. This could mean, for example, a longer interval between subsequent hash chain values. The client would need to send an additional control message to sign the offer with a new hash chain anchor value. Otherwise, it could still be charged the higher price. This could be found out, though, when (or if) the client disputes the costs and presents the alternative offer. So, in low value transactions it could be possible to just use R2 to signal that the hash chain value interval is shifted (for the benefit of the client). Another approach is that the value of token is lower between service provider and the third party, thus the generated bill is lower.

### 3.3. Authorisation issues

While the service provider might have some external knowledge about the client's liability for service usage based on its identity, generally the client also needs to attach an authorisation statement from TTP that states that the client is trustworthy to receive the specified service for the specified amount. The service might be specified based on service types or it could be specified on the provider level. In other words, the specified service and the service offer identities should then match. This is a slightly less flexible option, but provides more security, because overspending can be controlled more easily. In case the service granularity is just based on the service type, the client could use several different service providers for the maximum amount defined in the certificate during the validity period. Of course, at the time of the clearing the third party would notice this and could initiate appropriate procedures against the client. This is really no different from the way post-paid phone calls are charged. Thus, TTP has the liability, but it can still control what sort of certificates it issues and hence manage its own customer risk. Issuing only short lived authorisations is also one way of mitigating risk.

```
(
 (cert
  (subject (hash sha1 <hash value>))
  (issuer (hash sha1 <hash value>))
  (target-service-url (hash sha1 <hash value>)
  (amount-max (time (s 3600)))
  (propagate)
  (validity
   (not-before 2008-07-29_12:00:00)
   (not-after 2008-07-30_12:00:00))
)
 (signature (dsa-sha1 <sig>))
)
```

**Figure 5. Example of TTP certificate**

In Figure 5 we give an example of a third party authorisation in the form of an SPKI certificate, which allows certain subject to access the indicated service. The subject is identified by the hash of the public key, even though one could also use HIT to identify the party. However, as HIT includes IPv6 kind of interpretation (see [13]), there is slightly larger chance of collision than in the case of hashing a public key. The target service is also indicated with the hash value calculated from the service URL (or just with suitable URN) and the maximum service time is also indicated in order to limit the "credit" of the client. Note that

TTP certificates could authorise various other things as well and act as a policy distribution mechanism. This is really up to the agreement made by the service provider and the third party.

The example certificate also includes the propagate option, which allows the client to assign similar rights to some other entity, but ultimately it is still responsible for the incurred costs. Of course, there is no obligation for the third party to allow the delegation in the first place, but the privacy of the client is better served, if there is a possibility of delegating the authorisation to an ephemeral identifier, which is visible to the external observers of the base exchange. This kind of setting also enables the user to pay for service usage of others. Naturally, the client is also responsible for issuing a separate certificate signed with the original identity that authorises the ephemeral identifier to use the TTP certificate (see Figure 6). There should be an expiry time as well. This kind of scenario then requires that the certificates are encrypted, so that the correspondence of identifiers is not evident to the outsiders. Obviously, this does not provide anonymity towards the service provider.

```
(
 (cert
  (subject (hash sha1 <hash value>))
  (issuer (hash sha1 <hash value>))
  (validity
   (not-before 2008-07-30_08:00:00)
   (not-after 2008-07-30_09:00:00))
 )
(signature (rsa-sha1 <sig>))
)
```

**Figure 6. Example of delegation certificate**

The client should pay attention to the validity times and authorised amounts in the certificates, so that it has valid authorisation available, if the service requests new negotiation after the previous hash chain values have been used up to the specified maximum. At this point there is no need to do the whole base exchange again and the parties can take advantage of the HIP control packets to update the association.

### 3.4. Hash token handling

HIP UPDATE packets are used to transmit the next hash chain value, when it is due. This requires additions for HIP specifications. That is, a new HIP parameter needs to be defined, such as depicted in Figure 7, which identifies the used hash chain and the next value. The UPDATE must also be acknowledges with the corresponding ACK packet in order to make sure that the packet has not been lost. For added security, the parameter should be encrypted, so that someone else cannot capture the hash value and use it to pay for its own service. Naturally, the server should detect this kind of case and prevent the use as it knows which chain is related to which client, but to some less scrupulous servers just the acquisition of the token can be enough.

| Type (15 bits) | C | Length (16 bits) |
|---|---|---|
| Hash chain id length (16 bit) | | Hash value length (16 bit) |
| Hash chain id (variable length) | | |
| Hash value (variable length with possible padding) | | |

**Figure 7. HIP parameter to convey hash chains**

Alternative approach would be to integrate the transmission of hash chain values into the transport protocols, e.g., IPv6 headers could be used in the use case described below. However, this would require making similar modification to every transport case for which the non-repudiation mechanism was applied. Clearly, it is easier to use more general approach with the available HIP update mechanism.

When the service wants to cash in the tokens, it contacts the third party in question and presents the given offer, the response and the relevant authorisation certificates. Also, the amount of tokens and the last received token value are submitted, so that the third party can verify the correct amount of used hash chain tokens. The third party compensates the service provider and at later point presents a bill to the client.

## 4. Use cases

Here we discuss potential use cases for the suggested NoRSU method. The cases presented deal with network access and streaming services.

### 4.1 Network attachment

The network attachment scenario is very much the basic use case for NoRSU. Thus, the idea is to "pay" one's net usage with the exchanged tokens and prove that one is authorised to receive service. This could mean, for instance, allocation of certain amount of transmitted bytes per time unit.

In the network setup we assume that we have an access point (or possibly several of them) and an access point controller, which also functions as a gateway to the external networks. The user makes the

initial attachment to the access points, but the actual base exchange is run with the access point controller. The setup resembles the architecture given in [14], which allows even the link level frames to be transmitted to the controller. Note, however, that the access points still can exhibit enough intelligence to check the validity of the puzzle solution, so that the invalid packets do not even reach the controller, hence further mitigating the denial of service concerns.

Even though we are working in the access domain and discuss mainly the interaction between the user and the access controller (corresponding the client and the server of the previous section), one could also device ways to include the home domain into the online transaction. For instance, a setup envisaged in [7] could be one alternative.

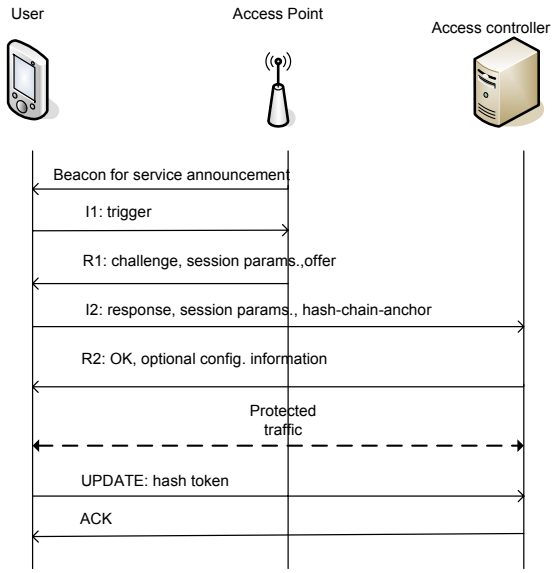



**Figure 8. Attaching to network using non-repudiable service usage**

The message exchange between the parties is depicted in Figure 8. We envisage the access points to be intelligent ones, so that they are able to respond to the initial I1 message with the precalculated R1 message, which also contains the offer for the network access use in the form of an SPKI certificate (see example in Figure 9). Naturally, there has been previous communication between the access point and the controller regarding the contents of R1 messages.

The user's response comes in the I2 message and it is forwarded to the controller in case the puzzle solution is correct. An example of the attached certificate is given in Figure 10. Once the controller has accepted the user as valid communication partner, R2 message is sent as an acknowledgement of the transaction as in typical base exchange. The accounting part is implemented with the help of HIP UPDATE packets as depicted earlier.

```
(
 (cert
  (issuer (hash sha1 <hash value>))
  (offer (time 1 (s 60)))
  (validity (not-after 2008-10-30_12:00:00))
  )
 (signature (rsa-sha1 <sig>))
)
```

**Figure 9. Example of offer certificate**

```
(
 (cert
  (hash-of-offer sha1 <hash value>)
  (hash-chain-anchor sha1 <anchor value>)
  (issuer (hash sha <hash value>)
  )
 (signature (rsa-sha1 <sig>))
)
```

**Figure 10. Example of response certificate**

While one might make the assumption that the network setup ensures that no traffic hijacking or redirecting can take place, it is not for certain in all environments. So, there is need to bind the actual traffic to the used identities and hash chains. The binding to the negotiated association could be done either on link or network layer, for which the base exchange provides keying material.

As discussed in [14] the link layer security can be extended all the way to the controller, which makes it transparent to the user from the network layer point of view. The similar kind of link layer setting is envisaged in the network attachment procedure described in [15] and it is based on the similar HIP alike protocol.

On the network layer the binding to the actual traffic can be done with the help of IPsec. In other words, the participants also establish IPsec association during the base exchange. As the same keying material is used to for the association setup, the binding to the tokens can be ensured. However, this basically requires that the user tunnels all the traffic to the controller that imposes extra overhead. This is similar as is envisaged to be done with Protocol for carrying Authentication for Network Access (PANA) based IPsec access control solution [16], even though key management solution is different. Using modified transport mode ESP might be one solution, but it violates the original end-to-end idea of it and requires changes to the packet processing at the controller side, i.e., it has to first process (and

remove) ESP part before forwarding the packet towards its final destination. Additionally, if the end user wishes to setup HIP associations with other hosts, one need to make sure that there is no Security Parameter Index (SPI) collisions with the existing association with the controller. Similar concerns touch Bound End-to-End Tunnel (BEET) mode [17]. Thus, tunnel mode and link layer approaches provide more feasible approach. Also, they have the possibility of protecting the privacy of the user in terms of with what other nodes it communicates. The actual binding is negotiated during the base exchange.

### 4.2 Accessing streaming service

Here we consider the case where a client wishes to access a streaming service provided by the server. This could be a multimedia service, such as downloading a song or a video, which is using Real-time Protocol (RTP) for executing the transport of streams [18]. Note that generally the stream is described beforehand, for example, with Session Description Protocol (SDP), but here we only concentrate on the transport part.

RTP itself provides little security, so in order to make the strong binding to the actual negotiation, we use the secure profile of RTP, i.e., Secure RTP (SRTP), which provides integrity and confidentiality services along with replay protection [19]. While using IPsec with real time traffic might be an option as well, the added latency and jitter can degrade the quality performance of such solution significantly [20]. However, many current tools might still favour IPsec due to more tried and interoperable key management.

RTP is a framework that is intended to be extensible enough to allow easy creation of profiles to meet the requirements of applications requiring transport of different kinds of real-time data, e.g., Voice over IP (VoIP). It consists of two different protocols: RTP for transporting the actual data and RTP control protocol (RTCP), which is used to report the characteristics of the connection, such as the quality of service, and convey information about the participants [18]. Hence, in very simplified terms one can consider RTP to be flowing from the server to the client and RTCP from the client to the server. Note, however, that RTP is intended to be applicable to multicast scenarios as well, although in our discussion we are concentrating on unicast transmission as HIP associations are mutual. HITs could be applied in multicast solutions, though.

There exists some work that has discussed the integration of SRTP with HIP [21], and that is the basis of this use case. Basically, the idea is to bind the RTP stream to the negotiation that has taken place within the base exchange. As SRTP leaves the question of key management open, we can use the HIP mechanisms to create the common master secret that can be used to establish the required session keys for the real-time session. [21] defines the additional parameters that have to be included in the HIP base exchange in order to achieve this, although the definitions are not yet complete. The modified protocol flow is depicted in Figure 11. Alternative is to run the offer-response interaction in the base exchange and then do the SRTP negotiation after that using the UPDATE packets. In any case, the UPDATE packets are used for re-keying.

The use of SRTP parameters provides the participants an agreement about the used encryption and authentication algorithms and their corresponding key lengths. Also, key derivation function is agreed, so that session keys can be derived from the master key and master salt. The salt is also exchanged, but the key is extracted from the keying material that is created during the base exchange (an index to the keying material can be provided). Other RTP specific parameters can be exchanged as well, such as those indicating the synchronisation source for identifying the participant and rollover and initial sequence numbers for packet indexing.

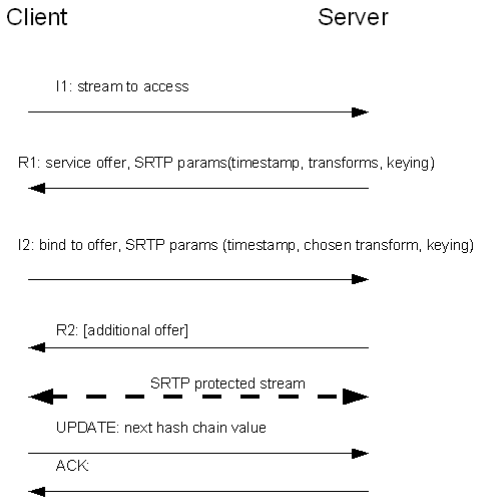



**Figure 11. Using NoRSU with SRTP**

However, we still need to add the non-repudiation property to this solution according to our suggested mechanisms. Thus, the I1 message already signals the client's intention to access a streaming service, so the R1 message can contain the corresponding response, which gives both the service offer and the proposed SRTP parameters. I2 contains the selected SRTP parameters along with the client's response to the offer. R2 does not contain any additional SRTP data. After

the exchange the both parties have an understanding about the frequency of the release of tokens.

During the service use the client needs to transmit the tokens to the server and this can be done using the same HIP UPDATE packets as described in the previous section. Another option that would integrate accounting more tightly with the stream itself would be to use the reporting functionality of RTCP, or more precisely, Secure RTCP (SRTCP), which provides the same services for RTCP as SRTP provides to RTP [19]. However, RTP philosophy does not take into account the acknowledgement of packets as loss of a single packet is not consider that important. Thus, detecting loss of packets transmitting the hash tokens would end up increasing the complexity. Also, as mentioned previously, we wish to prefer the general approach with the employment of HIP UPDATE.

So, in this use scenario we have described how a streaming service could take advantage of non-repudiation. The stream is strongly bound to the used identities, because the keying material used to protect the multimedia stream is derived from the negotiation done during the base exchange. That, in turn, translates to the used identities.

## 5. Implementation restrictions

In terms of packet size, overloading of HIP messages faces some challenges. This section talks about the relevant restrictions.

### 5.1 Available space in frames

While using the previously described certificates for authorisation is somewhat straightforward, one has to remember that the usage of HIP sets some restrictions. Mainly, due to the defined packet format, the length of HIP parameters is restricted to 2008 bytes, which has to accommodate the mandatory base exchange parameters, as well [2]. Also, the HIP headers (being logically IPv6 extension headers) are specified as unfragmentable, i.e., the IPv6 implementations are not allowed to fragment the header in order to meet the maximum transmission unit (MTU) of packets. This is mainly intended for avoiding DoS attacks caused by invalidly fragmented packets.

One additional limitation, i.e., around 1500 bytes, for MTU could be the use of Ethernet links on some section of the paths, but the situation can be actually worse than that. IPv6 specification states that the minimum IPv6 implementations are allowed to assume 1280-byte MTUs [22], and according to the survey done in [23] many of the current IPv6 paths seem to use it (well over 40% of the surveyed paths). Thus, length restriction due to small MTU cannot be ignored. Naturally, in environments that support higher MTUs or link layer fragmentation, like in many wireless technologies, the requirements are more relaxed, but one still needs to consider the whole path. Note that HIP is intended to be usable with IPv4 as well, where minimum MTU requirements are different, but focus of our discussion is on IPv6.

If one considers the typical HIP base exchange, it is quite obvious that the most of the information content is in R1 and I2 messages. Hence, they are the most restrictive ones for our purposes. Unfortunately, they are also the messages that will be carrying our extra payloads, i.e., offers and responses. As the amount of bytes changes from application to application, it is not easy to tell the exact amount needed for each messages, but looking at the HIP specifications and the available base exchange parameters, one can make an estimation of the used bytes. Table 1 presents mandatory HIP parameters for R1 and I2 messages along with a couple of most likely optional ones (R1_counter for indicating the current generation of the puzzle and Echo for echoing the transmitted value back), which are used to enhance the security properties of the protocol.

**Table 1. Estimated sizes of R1 and I2 messages**

| Parameter | R1 bytes | I2 bytes |
|---|---|---|
| [R1_counter] | 16 | 16 |
| Puzzle | 16 | |
| Solution | | 24 |
| Diffie-Hellman[1] | 200 | 200 |
| HIP_transforms | 16 | 8 |
| Host-id[2] | 250 | |
| Encrypted host-id[3] | | 280 |
| [Echo_request/response] | 20 | 20 |
| HMAC | | 24 |
| HIP_signature | 140 | 140 |
| | | |
| Total | 658 | 712 |

[1]Assumed just one 1536-bit D-H group (up to two groups could be proposed)
[2]1024-bit RSA key with 100 bytes domain part (which is optional)
[3]Encrypted using 128-bit AES-CBC (encryption is not mandatory)

As mentioned, table figures are just an estimation, which takes into account the "typical" parameters. The situation would be quite different, if one were to require, for instance, larger Diffie-Hellman groups and

longer keys in the name of better security (which is quite understandable, for instance, in the case of long term keys). Additionally, one might need extra parameters to signal additional associations, such as the case when negotiating the use of IPsec ESP for subsequent traffic.

As discussed in [9] the key types have significance as well. In the table above we assumed the use of RSA keys, but HIP also allows employing DSA keys. RSA and DSA keys provide roughly the same level of security for similar key lengths, but the RSA key generally has a shorter representation, because with DSA you basically also have to transmit domain parameters (this can be avoided in some special scenarios, though) [24][25]. However, the DSA signature takes less space than the corresponding RSA signature, which is dependant on the key size.

When considering these two things together, one can come to a conclusion that if one has to transmit both the key and signature, the RSA is more optimal solution length-wise. However, if it is required that only the signature is transmitted, DSA is a better alternative. Thus, this leads to a conclusion that within our context RSA is better suited for host identities, whereas the trusted third parties could use DSA keys to generate the signatures for the certificates. It is, after all, assumed that the parties have pre-established trust relationships with the trusted third party and know their relevant public keys.

The most optimal solution for this case would be elliptic curve cryptography (ECC), because it offers shorter key sizes and its performance is comparable to DSA [26]. Currently, however, HIP does not specify the possibility to use ECC keys for host identities. This should be a viable research direction for the future, especially considering the increase in key lengths over time.

## 5.2 Encoding

As discussed in the previous subsection, our working environment is somewhat restricted when it comes to the length of the messages. Thus, there is also need to consider the encoding of the embedded certificates. The previous examples were given using S-expressions, which, while human readable and good for examples, are unsuited for transmitting on the wire. For instance, the signatures and other binary data could be presented with base64 encoding, which clearly is wasteful when it comes to the used space.

SPKI drafts define the possibility to use canonical S-expressions, which aim at more efficient packing of the information [27]. It is also a form, which is expected to be used, when doing operations, such as hashing, on expressions. It basically presents the expressions as binary byte strings, i.e., octets, and precedes every token with the length value. This allows presenting binary data in a concise way, but still the textual tokens use the space inefficiently. In Figure 12 we present an example of canonical form of the delegation certificate given in Figure 6 (binary data omitted and line breaks added for readability). Using this encoding the length is reduced by around 60 bytes, but still the size of the certificate is around 350 bytes.

```
((4:cert(7:subject(4:hash4:sha120:<omitted>))
(6:issuer(4:hash4:sha120:<omitted>))
(8:validity(10:not-before19:2008-07-30_08:00:00)
(9:not-after19:2008-07-30_09:00:00)))
(9:signature(8:rsa-sha1128:<omitted>)))
```

**Figure 12. Example of canonical encoding of S-expression (formatted for readability)**

However, there is some work that considers binary encoding of SPKI certificates and that would suit our purposes as well. A SPKI authorisation certificate presented in [28] took a little over 250 bytes. It contained hashes of the issuer and subject public keys, validity dates, a simple attribute and a DSA signature, which actually could be made even more concise. In [28] it took 190 bytes, because it also contains the public key of the certifier (without domain parameters). Thus, if one makes the representation of the signature more concise, it is possible to save over 100 bytes. After all, HIP base exchange already conveys the public keys.

**Table 2. Examples of records of the efficient encoding scheme**

| Expression type | Implied size in bytes |
|---|---|
| Subject 1024 bit rsa public key hashed using sha1 | 20 |
| Subject expressed with HIT | 16 |
| Valid end date | 4 |
| Valid start and end date | 8 |
| Signature 1024 bit rsa using sha1 | 128 |
| Signature 1024 bit dsa using sha1 | 40 |
| Signature 2048 bit dsa using sha1 | 56 |

So, if we take into use similar kind of encoding that only has a 2-byte type field and a variable length value field per one record. The length is expected to be implicit based on the type. The type encodes much of the expression itself, e.g., we might have different

types for issuers expressed with HITs or just with SHA-1 hash values, i.e., multiple textual tokens are reduced. Also, some expressions, such as validity times can be reduced to more concise form by encoding multiple values into a record and using seconds to express dates. Thus, we are driving towards utmost efficiency at the expense of flexibility. Table 2 shows an example of some encoded expression types and the implied size of the following value field.

Table 3 shows how many bytes different certificates could take using this efficient encoding. When comparing, for instance, the efficient encoding of delegation certificate to the canonical encoding, the reduction is almost 50%. Note that encoded values could contain additional structure inside them, e.g., the encoding of the actual offer expression takes into account values such as the amount of hash tokens needed, the used unit, and the amount of units. So, one should be able to express things like 1 hash token per 60 seconds or per thousand kilobytes. A more complex offer would include the possibility to point to an external offer, which could be an XML document giving more details, but in the access scenario the client ought to be able to access the said document, i.e., have connectivity before connectivity service has been agreed on. We are, however, aiming for simplicity.

## 5.3 Summary of restrictions

When we consider the previous discussion regarding MTU and consumed bytes in HIP parameters, it is obvious to question, whether the suggested certificates can be included within the HIP messages. Taking into account the figures in Table 1 and amount of bytes needed for IPv6 and HIP fixed headers (both take 40 bytes), it can be concluded that with a safe margin one can use roughly 400 bytes for certificate information (R1 can contain bit more). One should also not forget the HIP parameter for signalling the non-repudiation and the target service. In a case of simple service naming (like a hash), the parameter would take 32 bytes, but with other encodings it naturally could be larger. There is room for optimisation, though. If we were to leave out the domain part of the host identity, which basically can contain Fully Qualified Domain Name (FQDN) or Network Access Identifier (NAI), we can save around 100 bytes compared to the given figures. Considering that we are planning on giving explicit authorisation for the used identities in the form of certificates, the dropping of domain part is not so crucial.

Now, if we also look at the data given in Table 3, we can come up with estimates for the amount of data added due to the certificates. If we first consider R1 message, one can expect that it contains an offer for the service, but also an authorisation issued to the server by a TTP. The table actually just gives figures for client authorisation, but the amount of needed bytes is similar. Thus, those two certificates fit within the constraints given. For I2 message one needs the response and also the authorisation ensuring the liability of the client and this should not be a problem, either. However, if we want to support the advanced scenario, where the right to use the service is delegated to another entity (or, in case of privacy protection, to another identifier of the same entity), we are hanging on the very edge of our constraints. Thus, implementations would need more care in such circumstances. The negotiation of key management procedures for additional protocols, such as those depicted in the use case of SRTP, might have to be postponed to the UPDATE messages after the base exchange has completed.

If such additional roundtrips are undesirable, there is still room for further optimisation in the used certificates. As the HIP packets already contain signatures of the client and the server, the end point generated certificates for offers and responses can do without signatures. This saves well over 100 bytes (in case of RSA signatures) and enables one to fit all the authorisation statements within the limits we have set. The downsides of this approach are increased storage requirements and added complexity, because the parties need to store the whole HIP packets instead of a set of certificates. The clearing party has to also be able to understand HIP structures.

It is worth noting that the previous discussion assumes 1024-bit keys, whereas longer keys make it even harder to fit the information within the HIP

**Table 3. Byte count for different certificates**

| Offer | bytes | Response | bytes | TTP-client | bytes | Client deleg. | bytes |
|---|---|---|---|---|---|---|---|
| Issuer | 22 | Hash-of-offer | 22 | Subject | 22 | Subject | 22 |
| Offer | 6 | Chain-anchor | 22 | Issuer | 22 | Issuer | 22 |
| Validity-end | 6 | Issuer | 22 | Target serv. | 22 | Validity-range | 10 |
| Rsa-sign-1024 | 130 | Rsa-sig-1024 | 130 | Amount-max | 6 | Rsa-sig-1024 | 130 |
| | | | | Propagete | 2 | | |
| | | | | Validity-range | 10 | | |
| | | | | Dsa-sig-2048 | 56 | | |
| Total | 164 | | 196 | | 140 | | 184 |

header. It should be noticed, though, that the constant increase in computing power and the developments in the mathematical algorithms is bound to raise the bar for the required key lengths. The 1024-bit keys are considered to be adequate for the next couple of years, but scenarios needing longer term solutions, such as those related to the trusted third parties, should already use 2048-bit keys [29]. This further motivates the need to look into the possibility of using ECC keys.

One additional length consideration relates to the use of hashes. Even though SHA-1 is a very common hash function, it is showing some weaknesses [30]. Therefore, one should also consider the use of advanced forms, such as SHA-256, instead in places that require hashing of relatively free form messages. The increase in length is just 12 bytes, though, but can build up when used in multiple places Note, though, that when hashing public keys and the attacker wants to find another key that hashes to the same value, it is very unlikely to find such a value, because one is not able to modify the source value at will and still retain the required structure for a public key. This also applies to the case of HITs, even though they are shorter than SHA-1 hash values. The case where this matters most is the hashing of the offer of the server to indicate to which offer the client is binding itself.

## 6. Analysis

The following subsections analyse the potential threats and the corresponding countermeasures from the viewpoint of our proposal.

### 6.1 Threats within the context of the solution

In this section we discuss the potential threats that can emerge in an environment that plans on adopting the suggested token based solution. One should note that not all of them have a technical countermeasure, but those should then resort to other measures offered by the society in case of agreement dispute, such as litigation. While this subsection concentrates just on listing the threats, the way the countermeasures take place is discussed in the next one.

As has been described earlier the interaction is mainly between the user and the service, but from the threat analysis perspective one has to also remember the existence of a third party, such as the home operator, who acts as a trust and liability broker between the entities. There might also be an external entity, who could try to interfere with the service provisioning.

In the case of compensation of the service usage, the setup can pose several threats to different parties. The most obvious threats are that the service is not paid for or that the service is not received after paying. Especially when the user pays after the service usage (post-paid), there is a chance that the user repudiates it, i.e., claims that he has never used the service and the cost claims are unfounded.

Different collusion scenarios can be envisaged. In other words, two of the parties conspire against the remaining one. The home operator could assure the trustworthiness of the user without any intention of compensating the service afterwards at the time of the clearing. On the other hand, the user and the service could collude against the home operator in order to make the home operator compensate the service without the user having no intention of paying his bill later on. While being perhaps the most unlikely case of these, the service and the home operator could try to make the user pay more than the user originally thought (misleading advertising is another matter).

Double spending can occur when otherwise valid tokens are replicated to pay for several different transactions. In a similar sense, the service might try to charge the accessed service more than once. Very close is also the case of overspending, when the user consumes more resources than he can afford, i.e., overly large amount of tokens is created and used.

Hijacking of information by an unauthorised party could also take place. The payment tokens or the actual payment could be stolen by some other service or another user could try to use the tokens to pay for his service. Also, instead of tokens, another user could try to hijack the paid service from the legitimate user.

Integrity of the compensation agreement could be facing threats, as well. Either the user or the service provider might try to modify the agreed terms, so that the later claims would be more favourable to them. This also includes forging of additional tokens so that the service could claim more resource usage than really took place.

User privacy is always an existing threat in any communication system and it will become even more important as the transition towards ubiquitous communication takes place. This is especially evident in our proposed solution, which makes heavy use of different kind of identities. User privacy is at stake if the identity information is disclosed to unauthorised parties, who are then able to track the users, for instance. Also, users may also wish to prevent others from learning what sort of services they are using and what sort of usage patterns they follow. Thus, the users should be in control of the disclosure of information about themselves and their actions.

As mentioned above, within the limits of our proposed solution, some of these threats can only be addressed through litigation. For instance, if some party refuses to pay even when faced with technical evidence, the other parties have to initiate legal procedures in order to get the promised compensation. This is not, however, different from the case, where a user refuses to pay his post-paid subscription or credit card bill. This is a business risk, which should be embedded in the business models of the players.

Generally, when faced with collusion of other parties, the legal action with the technical evidence is the only solution. Naturally, fear of losing one's reputation can be enough disincentive for the home operator to not to cheat the user as the user trust is the very foundation of its business model. In the following section we analyse the properties of our proposal, which can provide solutions to the technical threats presented above.

### 6.2 Technical measures against the threats

The basic components in the proposed solution are the use of hash chains and the binding of the identities to them. The hash chains provide the means to pay for the service usage in a piecemeal fashion, i.e., as long as the service is received, additional hash chain values can be submitted. Analogously, as long as the server keeps receiving new hash values that are part of the chain, the service is provided. Thus, in case of malicious party, no further compensation or resource provisioning is provided, i.e., the granularity of the commitment is better controlled. Generally, the value of a single hash token should be kept small as that is the amount that can be lost in the case of misbehaviour.

The hash chain values have the added benefit of being easily verifiable, because the receiver has to only compute one hash function in order to make sure that the received value is part of the chain. Naturally, the used hash function has to be secure enough, so that the receiver is not able to calculate future values. This ensures that only the entity that has knows the secret seed of the hash chain knows the transmitted values beforehand. Hence, the service provider cannot easily create additional tokens, so that it could make cost claims for unused resources. Also, as the value of single token is kept small, the required effort of brute force attack clearly outweighs the benefit.

Non-repudiation property of the solution comes from the binding of the identities to the presented offers and responses. When the user presents the anchor value of the hash chain, he has signed the statement with his identity and also included in the statement the reference to the received offer. This, along with the assumption that the hash function is irreversible, dictates that only that user has been able to create the said hash values. Thus, if the user denies using the service, the service provider only needs to present the anchor value signed by the user and the last received chain value in order to prove that the user has used service with the offered terms. The service provider can naturally deny that it has provided any service, but from the point of view of our solution concept it does not matter as the user already has consumed the desired resources. The service, however, cannot deny that is has given a service offer on certain terms.

The trust to the client's ability to pay comes from the associated TTP certificate, which authorises the client to use a certain maximum amount of commodity. This can be seen as the credit the client has in the eyes of TTP and as an acknowledgement that TTP knows the client. This way the server has certainty that someone will provide compensation for the provided resources, because ultimately TTP has accepted the liability in the case of misuse when issuing the certificate to the client. It is then up to the agreement made between the third party and the client to settle the costs. This does not differ from the typical post-paid business model commonly used in the telecom or credit card industry.

It is also possible to grant an authorisation certificate for the service as well, to be presented as proof of its trustworthiness. Although, as stated above, in case the server is not providing the service it promised to deliver, the client just can stop sending any hash chain values. If the service in question is other than the typical access scenario, like buying a song, then the motivation to include such authorisation might be different. This is basically a risk management decision for the client. Of course, if the song, for example, is streamed, then the client has better control of what it is receiving and can pay it piecemeal, like in the second use case scenario described earlier.

The employment of HIP provides a natural way of taking advantage of the accompanying cryptographical identifiers for presenting the identities of the parties. As it also provides a key management solution, it can be used to create the necessary association so that the actual data traffic can be bound to the same identities, even though it requires an existing data traffic protection mechanism, such as IPsec. Thus, even though IP addresses could be spoofed, the mutually agreed keying material ensures that the traffic is useful only to the valid partners.

Fine tuning of the solution is done through the extra attributes given in the certificates and the procedures the parties conduct during the transaction. When the client clearly states the service provider identity in the response message, no other service provider can claim the costs, even though it somehow could manage to get hold of the hash tokens. It is possible for the home operator to give more granular authorisations to the client and only allow certain service providers or service types. One should remember, though, that if the same authorisation allows the use of several service providers for the specific service type, the maximum allowed resource consumption could not be controlled without online access to the home operator, which tends to complicate things and decrease performance. However, this can be found out during the clearing procedure and extra claims made towards the user. This is basically a risk management decision for the home operator, when deciding what sort of granularity to use in the issued authorisations.

In any case, the server has to remember to check the provided anchor values, so that it is not possible for the client to use the same anchor value within the validity period of the same offer. Otherwise the client might be able to use the same hash chain values again, but the server would only be able to bill them once. This could also be prevented by having an individual session identifier in every R1, but cannot be done without breaking the basic HIP properties as the signature in R1 is pre-computed for the sake of mitigating denial of service possibility.

The privacy of the client is preserved through the decoupling of authentication and authorisation. TTP issued certificate can state that the ephemeral identifier assigned to the client is trustworthy for certain actions. Naturally, TTP is able to connect this to a real identity. Also, this introduces additional overhead in terms of additional interaction with TTP, especially if the

**Table 4. Threats and possible countermeasures**

| Technical threats | Preventive measures | Threats handled through litigation |
|---|---|---|
| User denies having used the service | Binding of identity to the hash chain | • User and home operator collude against service<br>• Server and home operator collude against user<br>• User and service collude against home operator<br>• User is charged too much at the time of clearing<br>• User refuses to pay at the time of clearing<br>• Service does not get money from the home operator at the time of clearing<br>• Privacy of the user (e.g., home operator releases information about the user without user consent) |
| Service provider does not provide agreed service | Stop transmitting additional hash tokens | |
| User gets no or other service that he paid for | Stop transmitting additional hash tokens | |
| Service hijacked by another user | Bind the payloads to the negotiated keying material | |
| Intercepted tokens used by another user to pay for his service | Service needs to ensure the strong binding between the identity and the hash chain, protection of tokens with the negotiated association | |
| User double spends the created compensation tokens | Authorise specific service, check anchor value uniqueness | |
| Service charges user multiple times | Non-repudiable accounting records are accepted only once | |
| Other service "cashes" the tokens | User authorises specific provider | |
| Service creates additional valid user tokens | Secure hash function prevents creating additional usage records and user signature protects the hash anchor value | |
| User modifies the offer to more favourable one | Offer is protected with signature | |
| Service modifies the offer to more favourable one | Offer is protected with signature | |
| User overspends his credit | User is authorised only to spend certain maximum amount | |
| Privacy of the user | Use of ephemeral identities and delegation | |

identifier is changed often. This allows providing anonymity towards the service providers, though. The other option is that the TTP provides an authorisation for the long term identifier and the client constructs an ephemeral identifier for which it delegates the authorisation. Even though this is a more flexible option, it does not provide complete anonymity towards the service providers, because they need both the delegation and the original authorisation. However, it does, like the other alternative, provide privacy protection against external observers, because the real identity does not have to be visible, not even in the form of its HIT.

In Table 4 we have summarised the different kinds of threats that might emerge in this kind of service usage concept. Additionally, the table present how the suggested solution can answer to the technical threats.

## 7. Conclusion

In this paper we have proposed a simple accounting scheme to be used in conjunction with HIP. With his kind of solution the service provider is able to get undeniable evidence that it is entitled to compensation for the provision of its resources to a certain client. On the other hands, the client can control the charging procedure, so that it is only billed for the costs that are based on the actual usage.

We have showed that the combination of HIP and hash chains can provide a secure solution for non-repudiable service usage that also takes into account the binding to the actual data traffic. This is further enhanced with the employment of authorisation certificates to increase the level of trust the client and server have for each others. Thus, it also provides an authorisation mechanism for the participants.

However, the used environment poses some problems, namely in the form of length restrictions, that need to be taken into consideration. We have considered the most common IPv6 path MTU, 1280 bytes, and concluded that even though it is possible to introduce the solution to this environment, the advanced scenarios providing better privacy support and the delegation of service authorisation may face difficulties with certain implementations. There is a possibility for length optimisation, although at the expense of increased complexity. Also, the choice of used key types has considerable impact on that and RSA based host identities are better suited for the most length restricted cases. This still calls for efficient encoding mechanisms, which have the downside of limiting the flexibility.

It is worth remembering, however, that the wide adoption of HIP is still years away, so the restrictions set by the current MTUs can be quite different in the future networks. It shows, though, that this is additional incentive for pushing for higher MTUs. Also, the research done with technologies that provide shorter key lengths, such as ECC, provides measures to answer to these restrictions, even though the requirement for having larger key sizes goes hand in hand with the increase in computing power. In any case, the host identity enabled environment provides many interesting directions for the development of secure charging schemes.

## Acknowledgment

The author wishes to thank Tuure Vartiainen and prof. Jarmo Harju for comments and suggestions.

## References

[1] Federal Communication Commission, "Unauthorized, Misleading, or Deceptive Charges Placed on Your Telephone Bill - Cramming", FCC Consumer Facts, online article, available in http://www.fcc.gov/cgb/consumerfacts/cramming.html (accessed 01/2009), Jul 2008.

[2] Moskowitz R., Nikander P., Jokela P. (Ed.), Henderson T., "Host Identity Protocol", IETF RFC 5201, Apr 2008.

[3] Moskowitz R., Nikander P., Jokela P., "Using ESP transport format with HIP", IETF RFC 5202, Apr 2008.

[4] Heer T., Varjonen T., "HIP Certificates", IETF Internet-Draft draft-varjonen-hip-cert-01 (work in progress), Jul 2008.

[5] Laganier J., Vicat-Blanc Primet P., "HIPernet: A Decentralized Security Infrastructure for Large Scale Grid Environments", The 6th IEEE/ACM International Workshop on Grid Computing, Nov 2005.

[6] Tschofenig H., Nagarajan A., Ylitalo J., Shanmugam M., "Traversal of HIP aware NATs and Firewalls", IETF Internet-Draft draft-tschofenig-hiprg-hip-natfw-traversal-02, expired, Jul 2005.

[7] Heikkinen S., Priestley M., Arkko J., Eronen P.,Tschofenig H., "Securing Network Attachment and Compensation", Proceedings of the Wireless World Research Forum Meeting (WWRF#15), Nov 2005.

[8] Blaze M. et al., "TAPI: Transactions for Access Public Infrastructure", Proceedings of Personal Wireless Communications (PWC2003), Sep 2003.

[9] Heikkinen S., "Non-repudiable service usage with host identities", Proceedings of the Second International Conference on Internet Monitoring and Protection (ICIMP07), Jul 2007.

[10] Lamport L., "Password authentication with insecure communication", Communications of the ACM, vol. 24, no. 11, 1981.

[11] Tewari H., O'Mahon D., "Multiparty micropayments for Ad Hoc Networks", Proceedings of the IEEE Wireless Communications and Networking Conference, Mar 2003.

[12] Zhou J., Lam. K., "Undeniable Billing in Mobile Communication", Proceedings of 4th ACM/IEEE International Conference on Mobile Computing and Networking, Oct 1998.

[13] Nikander P., Laganier J., Dupont F., "An IPv6 Prefix for Overlay Routable Cryptographic Hash Identifiers (ORCHID)", IETF RFC 4843, Apr 2007.

[14] Calhoun P. et al., "Light Weight Access Point Protocol", IETF Internet-Draft draft-ohara-capwap-lwapp-04 (work in progress), Mar 2007.

[15] Rinta-aho T. et al., "Ambient Network Attachment", Proceedings of 16th IST Mobile and Wireless Communications Summit, Jul 2007.

[16] Parthasarathy M., "PANA Enabling IPsec based Access Control", IETF Internet-Draft draft-ietf-pana-ipsec-07 (work in progress), Jul 2005.

[17] Nikander P., Melen J., "A Bound End-to-End Tunnel (BEET) mode for ESP", IETF Internet-Draft draft-nikander-esp-beet-mode-09 (work in progress), Aug 2008.

[18] Schulzrinne H., Casner S., Frederick R., Jacobson V., "RTP: A Transport Protocol for Real-Time Applications", IETF RFC 3550, Jul 2003.

[19] Baugher M., McGrew D., Naslund M., Carrara E., Norrman K., "The Secure Real-time Transport Protocol (SRTP)", IETF RFC 3711, Mar 2004.

[20] Bou Diab W., Tohme S., Bassil C., "Critical vpn security analysis and new approach for securing voip communications over vpn networks", Proceedings of the 3rd ACM workshop on Wireless multimedia networking and performance modeling, Oct 2007.

[21] Tschofenig H., Shanmugam M., Muenz F., "Using SRTP transport format with HIP", IETF Internet-Draft draft-tschofenig-hiprg-hip-srtp-02 (expired), Oct 2006.

[22] Deering S., Hinder R., "Internet Protocol, Version 6 (IPv6) Specification", IETF RFC 2460, Dec 1998.

[23] Wang Y., Ye S., Li X., "Understanding Current IPv6 Performance: A Measurement Study", Proceedings of the 10th IEEE Symposium on Computers and Communications (ISCC'05), Jun 2005.

[24] Eastlake D., "RSA/SHA-1 SIGs and RSA KEYs in the Domain Name System (DNS)", IETF RFC 3110, May 2001.

[25] Eastlake D., "DSA KEYs and SIGs in the Domain Name System (DNS)", IETF RFC 2536, Mar 1999.

[26] Cronin E., Jamin S., Malkin T., McDaniel P., "On the Performance, Feasibility, and Use of Forward-Secure Signatures", Proceedings of the 10th ACM conference on Computer and communications security, Oct 2003.

[27] Ellison C. (Ed.), "Simple Public Key Certificate", IETF Internet-Draft draft-ietf-spki-cert-structure-06.txt, expired, Jul 1999.

[28] Arbaugh W., Keromytis A., Farber D., Smith J., "Automated Recovery in a Secure Bootstrap Process", Internet Society 1998 Symposium on Network and Distributed System Security, Mar 1998.

[29] National Institute of Standards and Technology, "Recommendation for Key Managent - Part 1: General (Revise)", NIST Special Publication 800-57, May 2006.

[30] Wang X., Yin Y., Yu H., "Finding Collisions in the Full SHA-1", Proceedings of Crypto'05, Aug 2005.

# Consistency Checking of Web Service Contracts

M. Emilia Cambronero
Department of Computer Science
University of Castilla-La Mancha
SPAIN
Email: emicp@info-ab.uclm.es

Joseph C. Okika
Department of Computer Science
Aalborg University, Aalborg
DENMARK
Email: ojc@cs.aau.dk

Anders P. Ravn
Department of Computer Science
Aalborg University, Aalborg
DENMARK
Email: apr@cs.aau.dk

***Abstract*—Behavioural properties are analyzed for web service contracts formulated in Business Process Execution Language (BPEL) and Choreography Description Language (CDL). The key result reported is an automated technique to check consistency between protocol aspects of the contracts. The contracts are abstracted to (timed) automata and from there a simulation is set up, which is checked using automated tools for analyzing networks of finite state processes. Here we use the Concurrency Work Bench. The proposed techniques are illustrated with a case study that include otherwise difficult to analyze fault handlers.**

**Keywords:**
Web Services contract, consistency, WS Choreography, WS Orchestration.

## I. Introduction

Service Oriented Architecture (SOA) [1] reorganizes series of previously operational software applications and support infrastructure into an interconnected set of services, each accessible through standard interfaces and messaging protocols. It promotes services that are distributed, heterogeneous, autonomous and open in nature. SOA is particularly applicable when multiple applications running on varied technologies and platforms need to communicate with each other. With SOA, enterprises can mix and match services to perform business transactions with less programming effort. SOA is implemented with web service technology. Thus there is consensus today, that a web service is a programmable component that provides a service and is accessible over the Internet. They are based on standards like Simple Object Access Protocol (SOAP) [2], [3], [4], can be standalone, or linked together to provide enhanced functionality.

Businesses depend on web services, therefore their properties are of great importance, and informal checking and consensus approaches to when a service is good enough may not suffice. A business will only reluctantly use enterprise applications offered as open web services, because of the high risks involved in using untrusted services from unknown providers. Formal contracts defining the desired properties are therefore studied intensively today, because they are a way to manage the risks that come with the interaction among these inter-organizational services.

Traditionally, contracts in an object oriented setting consider only the functional aspect (pre-condition, post-condition, invariant) of an interface specification. A pre-condition is a constraint that must be satisfied before calling a method or

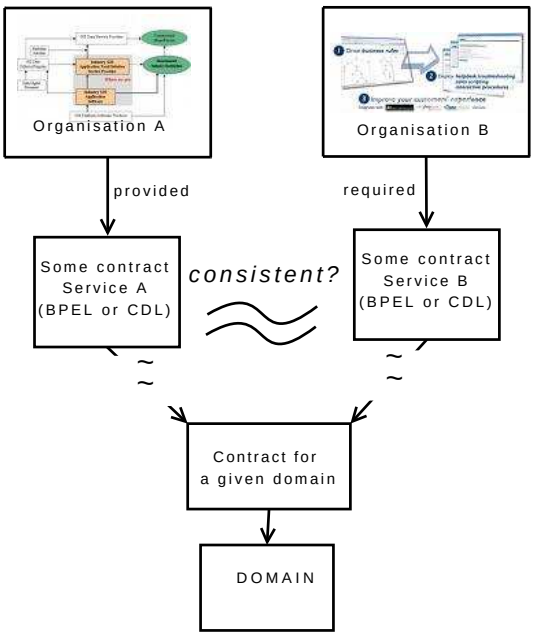



Fig. 1. Analysis of Web Service Contracts

operation; it checks for valid arguments. A post-condition is a corresponding property that is true when the call completes; it is the input-output relation. Finally, an invariant is a constraint on the state of an object; it must hold before and after any operation, and clearly after initialization of the object. These concepts, as popularized by Meyer's "Design by Contract" [5], are, however, just part of the properties exhibited by web services. Since web services are intrinsically distributed, they are by nature concurrent programs, and thus their overall functionality depends not only on correct implementation of the local functionality by sequential algorithms, but even more on the interplay between local functionality and global behavior (protocols and timing).

In this paper we focus on protocol or behavioural aspects of service contracts. There are several proposals for contract specification standards for web services, see e.g. [6] for an overview. Prominent among these standards are the Business Process Execution Language (WS-BPEL)[7] and Choreography Description Language (WS-CDL) [8]. BPEL offers a programming model for specifying the orchestration of web services whereas CDL specifies the choreography of

interacting services. However, when web service contract are specified using either BPEL or CDL, there is no assurance that they are consistent unless verified. Though there are efforts toward this form of analysis, there remain challenges in the area of automated approach to checking consistency in addition to other properties.

In previous work [9] we have demonstrated a viable solution to the problem of checking for functional and behavioural properties of individual services. This is done through translation of the specifications to timed automata followed by model checking for relevant properties. In [10] we considered the problem of consistency across specifications and identified a need to set up a correspondence between the individual automata. The novel contribution in this paper is to make such a consistency check practical by translating the automata to CCS, the input language for the Concurrency Work Bench. As demonstrated by a case study, this technique is applicable and gives a handle for automating yet another consistency check for web services.

*Directly Related Work:* Web Service contracts is attracting a lot of attention and several researchers propose various approaches and frameworks toward specification and analysis. For instance [11], [12], [13], [14] looks at it from a formal semantics viewpoint, whereas [15], [16] propose languages for specifying contracts. All these points to the fact that there is an important need for contracts to be specified and analyzed.

An earlier treatment of contracts in an object-oriented paradigm is Design by Contract [5]. Similar treatment concerning components is found in [17]. Here, the functional specification is achieved through assertions; which consists of preconditions, post-conditions and invariants. The framework in [18] takes a pragmatic approach at code level where the assertions are part of the language. We agree that these functional specifications are important in order to specify a formal agreement between a service provider and its clients. It expresses what a client should do before making a service request and what the provider will give as result of it.

Among the related work of Web Service contracts is [19]. It proposes to visualize contracts by graph transformation rules. Apart from expressing contracts in terms of pre- and post-conditions of operations together with invariants, they introduced the notions of provided and required contracts. With this, they use the provided contracts to create the test cases and test oracles whereas the required interfaces are used to drive the simulation. We like their treatment of functional specifications, but it needs to be supplemented with other aspects, and one may gain something by investigating model checking as a supplement to testing.

Quantitative aspect are researched in [20], [21], [22]. The Web Service Level Agreement (WSLA) framework [20] is targeted at defining and monitoring SLAs for Web Services. WSLA enables service customers and providers to unambiguously define the agreed performance characteristics and the way to evaluate and measure them. We want to mention here that WSLA complements Web Service Definition Language (WSDL) [23], [24], which is an XML grammar that describes the capabilities of Web services through its interface descriptions. WSLA is used to define a contract between service provider and service requester, but its treatment of functional behavior is limited.

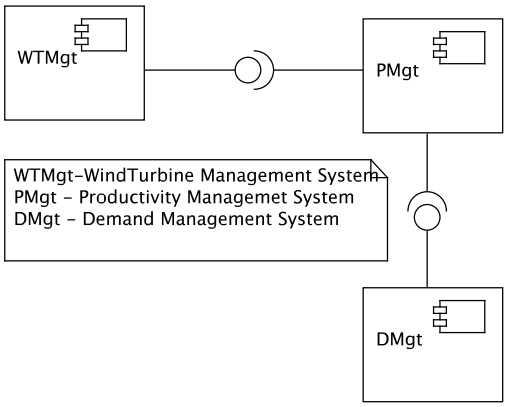



Fig. 2. Wind Turbine Management System Components

The above mentioned contributions focus on a single web service language, and either the functional or the behavioral side of a contract. We extend their perspective by considering the overall consistency of a service specified in languages covering more than one aspect. Furthermore we demonstrate how existing tools are adapted for such checks.

*Overview:* In Section II, we give a detailed presentation of Web Service contracts where the aspects of contracts are described. We introduce in this section, a case study of a Windmill Management System. Section III details the analysis of Web Service contracts. General consistency, satisfiability, and application specific issues are presented. A comparison with other approaches follows and finally, we conclude in Section V.

## II. Web Service Contracts

To manage the risks that come with the interaction among several services, the service provider and a consumer must have a contract that specifies the details of the service. As mentioned before, it is important to note, however, that there are different aspects of contract in play when dealing with web services. First, there is the functional aspect which describes the functional properties, and second, there is the protocols aspect which specifies the behaviour as a sequence of messages, events, signals, etc. There is also the extra functional QoS (Quality of Service) requirements aspect. This is further illustrated following the example presented in the following subsection.

### A. Example

We consider a Windmill Management System. The system monitors and controls wind turbines, and it has several components which are web services located in different places. We focus on three of these components, because it gives us

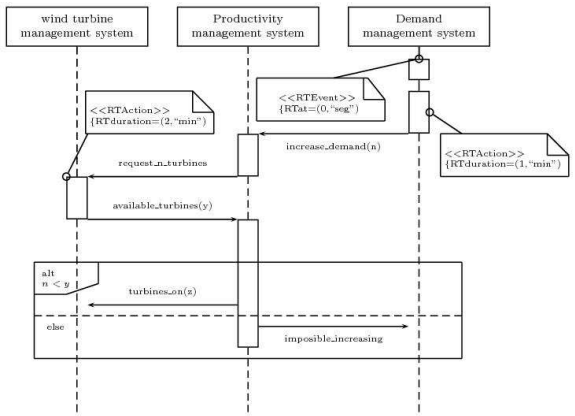



Fig. 3. Wind Turbine Management System Sequence Diagram

the scenario needed to specify a web service contract. The components are briefly described below and shown as an UML component diagram in Figure 2. The interaction between these services are illustrated using a RT-UML sequence diagram, shown in Figure 3. The informal requirements for the components are:

- Wind Turbine Management: sends a report to Productivity management every hour.
- Productivity Management: receives and analyzes the report from Wind Turbine Management.
- Demand Management: generates a report of power needs for Productivity Management.

We look at this example from two perspectives; WS-CDL and WS-BPEL. WS-CDL provides a definition of the information formats being exchanged by all participants. In other words, it specifies the protocols. WS-BPEL provides the message exchanges and functions as viewed by one participant. It describes the functionality of a single business process offered as a service by an enterprise.

### B. Contract Aspects in WS-CDL

CDL offers a model for specifying a common understanding of message exchanges. This language describes the choreography of web services systems, that is, the relationships between the composite services in a peer-to-peer environment. It uses the WS definition language (WSDL) to define and locate common type definitions.

WS-CDL is a very verbose notation, therefor the key concepts of contracts in WS-CDL are summarized below, while a full description of the demand management system is found in appendix A.

*Interface:* In WS-CDL, each interface is associated with a particular role, where a `roleType` enumerates potential observable behaviors a participant can exhibit when interacting with other participants. The syntax is the following:

```
<roleType name="DemandRoleType">
      <description type="description"/>
      <behaviour name="DemandBehaviour"
           interface="WSDLDemandType" />
</roleType>
```

The `behaviour` element defines an optional interface attribute, which identifies a WSDL interface type.

*Functional Specification: pre-conditions, post-conditions and invariants:* In WS-CDL these elements are defined by means of *workunits*; which define the constraints that must be fulfilled for making progress and describe some activities within a choreography. The constraints are give by *XPath 2.0* expressions.

XPath 2.0 supports date and time variables, so we can use these variables in WS-CDL as well. Furthermore, XPath provides a number of functions to manage these datatype values.

```
<workunit name="demand increase detected"
      guard="cdl:equal(cdl:getVariable
         ('tns:DemandClock'),'','),'0:00')"
      block="true">
  <assign roleType="DemandRoleType">
    <copy name="calculateincrease"
          causeException="true">
      <source variable="true"/>
      <target variable=
       "cdl:getVariable('detectedincreaseDone','','')"/>
    </copy>
  </assign>
</workunit>
```

A *workunit*'s `guard` element establishes the condition, which has to be fulfilled to perform the workunit activities. This element allows us to define pre-conditions. Postconditions and invariants can be introduced by appending a workunit with the condition as a guard at the end of the normal workunit flow. In order to define a condition we use XPath and XML Schema expressions.

*Protocol:* A *sequence* of activities is modeled in WS-CDL using the ordering structure `sequence`, which contains a set of activities that can perform sequentially.

A *non-deterministic choice* is implemented in WS-CDL using the ordering structure `choice`. The WS-CDL standard says that when two or more activities are specified here, only one of these is selected and the other ones are disabled. It is assumed that the selection criteria for those activities are non-observable.

The following WS-CDL code corresponds to the fragment in which the productivity system sends a message to the turbine system for the turbines to be turned on or else it sends a message to the demand system to indicate that it is not possible to satisfy the new demand. As you can see, it is modeled in WS-CDL by a choice activity in which we have two activities, and only one of them can be finally executed.

```
<choice>
  <workunit name="alt_else1_if"
            guard="Available == true" block="true">

      <interaction name="TurbinesOn_interaction"
            operation="TurbinesOn"
            channelVariable=
                 "Productivity2WindTurbineChannel">
        <participate relationshipType=
                    "ProductivityWindTurbine"
             fromRole="ProductivityRoleType"
             toRole="WindTurbineRoleType"/>
        <exchange name="TurbinesOnExchange"
                  action="request"/>
      </interaction>
```

```
  </workunit>

  <workunit name="alt_else1_else"
          guard="Available != true" block="true">

      <interaction name="Imposible_interaction"
                operation="Imposible"
                channelVariable=
                      "Demand2ProductivityChannel">
          <participate relationshipType=
                           "ProductivityDemand"
                fromRole="ProductivityRoleType"
                toRole="DemandRoleType"/>
          <exchange name="ImposibleExchange"
                    action="request"/>
      </interaction>
  </workunit>
</choice>
```

An *external choice* is implemented in WS-CDL using the ordering structure *workunit*, since it allows us to establish conditions to execute the corresponding activity. For that purpose, we may use the guards of workunits, by including in a guard an expression related with the value of a variable.

In WS-CDL, we use the workunit `repeat` to implement repetition. A workunit that completes successfully must be considered again for matching (based on its guard condition), if its repetition condition evaluates to `true`.

```
<workunit name="alt_else1_if"
         guard="Available == true"
         repeat="false" block="true" >

   <interaction name="TurbinesOn_interaction"
         operation="TurbinesOn"
         channelVariable=
             "Productivity2WindTurbineChannel">
       <participate relationshipType=
                        "ProductivityWindTurbine"
          fromRole="ProductivityRoleType"
          toRole="WindTurbineRoleType"/>
       <exchange name="TurbinesOnExchange"
                    action="request"/>
   </interaction>
</workunit>
```

*Timing:* Lower bounds, upper bounds, explicit clocks, reset and stop operations are handled by XPath and XML Schema.

XPath 2.0 supports date and time variables, so we can also use these variables in WS-CDL. Actually, XPath provides a number of functions to manage these datatype values. These variables can be used in particular to delay the execution for a certain time, or to establish the instant at which some actions must be executed. For that purpose, we may use the guards of workunits, by including in a guard an expression related with the value of a time variable.

Specifically, we use the XPath and XML Schema notation to specify the time aspects as follows:

*a) Explicit clocks:* are introduced by `xs:time`.

*b) Bounds:* are specified inside a workunit guard. In fact, as we capture delays or instants of execution, the specific expressions allowed are those constructed using the operators `op:time-equal op:time-less-than` and `op:time-greater-than` of XPath 2.0. We can also use the `hasDeadlinePassed` operation, which is defined in the WS-CDL specification to manage timing.

*c) Reset.:* In WS-CDL we reset a clock using an `assign` activity, which creates or changes the variable defined by the target element using the expression defined by the source element (in the same role).

*d) Stop.:* In order to model that a clock is stopped, we can capture the value of the time, of this specific instant, in a clock variable and then, when we want to initiate the time again, we can use the clock variable to continue from this point. We use two `assign` activities to capture and change the time value.

*e) Synchronization.:* The `interaction` WS-CDL element defines how the parties in a web services are synchronized. An interaction activity involves two roletypes, and an exchange of information between them. Actually, in WS-CDL several exchanges of information are allowed in a single interaction, and they can be either `request` or `respond` types, and these actions can be synchronous or asynchronous, depending on the `align` attribute.

```
<interation name="The demand management system
           sends increase in power demand to
           the productivity system"
   operation= = "sendIncreasing"
   channelVariable="Demand2ProductivityC">
   <description type="description">
     Sending the necessary increase of demand
   </description>
   <participate
      relationshipType= "DemandProductivity"
      fromRole="DemandRoleType"
      toRole="ProductivityRoleType" />
   <exchange name= "CalculatedIncerasing"
     informationType="Increase_demandType"
     action="request">
   </exchange>
   <timeout
     time-to-complete= "cdl:minor(cdl:getVariable
       ('tns:Clock1','','1:00')">?
</interaction>
```

In the `time-to-complete` attribute the timeframe in which an interaction must complete is specified. Then, when this time expires (after the interaction was initiated) and the interaction has not completed, a timeout occurs and the interaction finishes abnormally, causing an exception block to be executed in the choreography. The optional attributes `fromRoleTypeRecordRef` and `toRoleTypeRecordRef` are XML-Schema lists of references to record elements that will take effect at both roleTypes of the interaction.

*Faults:* Choreographies may have one exception block, which consists of some (possibly guarded) *workunits*, but only one of them can be finally executed (the first one whose guard evaluates to true). When the exception block is executed, the choreography terminates abnormally, even if the default exception workunit has terminated correctly. Exceptions are the following:

*f) Interaction failures:* For instance, sending of a message failed.

*g) Timeout errors:* For instance, an interaction did not complete within the alloted time.

*h) Application failures:* These are for instance illegal expressions.

*CDL in summary:* Overall CDL is a coordination language which focuses on the communication between agents providing the services. It is therefore very appropriate to give it

a semantics by translation into a network of communicating processes.

### C. Contract Aspects in WS-BPEL

BPEL is a programming language to specify the behavior of a participant in a choreography. It allows existing Web services to be orchestrated into composite services. Choreography is concerned with describing the message interchanges between participants.

WS-BPEL is verbose also, so we do not include full descriptions; but as for WS-CDL, we present the WS-BPEL contract aspects below:

*Interface:* In WS-BPEL, the services with which a business process interacts are modeled as `partnerLinks`. Each `partnerLink` is characterized by a `partnerLinkType`, which defines the roles played by each of the services in the conversation and specifies the `portType` provided by each service to receive messages within the context of the conversation. These `portTypes` are defined in the WSDL document, and each role specifies exactly one WSDL `portType`.

In order to utilize operations via a `partnerLink`, the binding and communication data, including *endpoint references (EPR)*, for the `partnerLink` must be available. The fundamental use of endpoint references is to serve as the mechanism for dynamic communication of port-specific data for services. An example fragment of a `partnerLink` is:

```
<partnerLinks>
<partnerLink name="productivity">
partnerLinkType="as:productivityDemandMSLT"
   myRole="DemandMS"
   partnerRole="productivity" />
</partnerLinks>
```

The endpoint references syntax is:

```
<service-ref reference-scheme="http://example.org">
  <foo:barEPR xmlns:foo="http://example.org">
   ... </foo:barEPR>
</service-ref>
```

*Functional Specification: preconditions, postconditions and invariants:* WS-BPEL uses several types of expressions to implement the functional part of a web service contract:

- Boolean expressions. These expressions can appear inside a transition, a join, a while, and an if condition.
- Deadline expressions. The WS-BPEL elements that use these expressions are until-expressions of onAlarm and wait.
- Duration expressions. These appear in the `for` expression of `onAlarm` and `wait`, and the `repeatEvery` expression of `onAlarm`.
- Unsigned Integer expressions, that include counter values `startCounterValue`, `finalCounterValue`; as well as branches in a `forEach`.
- General expressions inside assign activities.

*Protocol: sequence, choice, and iteration:*

- A sequence of activities is modeled by the `sequence` structured activity. It contains one or more activities that are performed sequentially, in the lexical order in which they appear.

  An example is the Productivity process which is given as a sequence as follows:

```
<sequence>
 <if
 bpel:getVariableProperty('x','time:level')==0>
   <then>
  <!-Process productivity (invoke) - ->
   <assign>
<copy>
<from partnerLink="productivityMS"
endpointReference="myRole" />
<to>&increaseData.productivityMSRef </to>
</copy>
 </assign>
 <invoke name="increaseDemand"
          partnerLink="productivity"
          portType="as:productivityPT"
          operation="process"
          inputVariable="increaseData">
   <correlations>
   <correlation set="increaseIdentification"
   </correlations>
 </invoke>
 </if>
</sequence>
```

- Choice. Both non-deterministic and external choice are expressed in WS-BPEL by means of `pick` activities, which waits for the occurrence of an event and then executes the activity associated with it. When several events occur simultaneously, an implementation dependent choice is made. Thus, in analysis, the choice must be modeled as non-deterministic.
- Conditional. WS-BPEL contains a conventional conditional statement as well.
- Iteration. WS-BPEL uses the `while` and `repeatUntil` activities, to model iteration.

```
<while>
   <condition>
      $numberWindTurbine < 10
   </condition>
   <scope>
     ...
   </scope>
</while>

<repeatUntil standard-attributes>
    standard-elements
    activity
    <condition expressionLanguage="anyURI"?>
       ... bool-expr ...
    </condition>
</repeatUntil>
```

*Timing:* Lower bounds, upper bounds, explicit clocks, reset and stop operations are specified as in WS-BPEL using XPath and XML Schema.

*i) Explicit clocks, lower and upper bounds:* They are defined using XML Scheme notations, as explained before.

*j) Reset:* In WS-BPEL we can reset the clock using an `assign` activity, which copies data from one variable to another.

```
<assign validate="yes|no"? standard-attributes>
   standard-elements
     (<copy keepSrcElementName="yes|no"?>
     from-spec
        to-spec
      </copy> |
      <extensibleAssign>
        ...assign-element-of-other-namespace...
      </extensibleAssign>) +
</assign>
```

*k) Stop:* In order to model that a clock is stopped in WS-BPEL we do as in WS-CDL.

*l) Concurrency and Synchronizations:* They are implemented in WS-BPEL using a `flow` activity, which provides concurrency and synchronization. A `flow` completes when all of the activities enclosed by it have completed.

```
<flow standard-attributes>
    standard-elements
    <links>?
        <link name="NCName">+
</links>
        activity+
</flow>
```

*Faults:* Business processes are usually of long duration. They can manipulate data in back-end databases and line-of-business applications. Error handling in this environment is both difficult and business critical. The overall business transaction can fail or be canceled after many transactions have been committed. In this cases, the partial work done must be undone or repaired as best as possible. Error handling in WS-BPEL processes therefore leverages the concept of compensation, that is, application-specific activities that attempt to reverse the effects of a previous activity that was carried out as part of a larger unit of work that is being abandoned. It thus provides the means for a forward error recovery.

Specifically, WS-BPEL provides constructs to declare fault handling and compensation.

*m) Compensation handler:* WS-BPEL allows scopes to delineate that part of the behavior that is meant to be reversible in an application-defined way by specifying a compensation handler. A `compensationHandler` is simply a wrapper for an activity that performs compensation.

```
<compensationHandler>
   activity
</compensationHandler>
```

It is invoked with `compensateScope`, when an explicit scope is compensated, or `compensate` when successfully completed inner scopes are compensated in reverse order. A compensation handler for a scope is available for invocation only when the scope completes successfully.

```
<compensateScope target="NCName"
                 standard-attributes>
   standard-elements
</compensateScope>

<compensate standard-attributes>
   standard-elements
</compensate>
```

Compensations may only be invoked in `catch`, `catchAll`, `compensationHandler` and `terminationHandler` activities, where termination handlers provide the ability for scopes to control the semantics of forced termination by disabling the scope's event handlers and terminating its primary activity and all running event handler instances.

*n) Fault handling:* In a business process it can be thought of as a mode switch from the normal processing in a scope. Fault handling in WS-BPEL is designed to implement backward error-recovery in that it aims to undo or repair the partial and unsuccessful work of a scope in which a fault has occurred. The completion of the activity of a fault handler, even when it does not rethrow the handled fault, is not considered successful completion of the attached scope. Compensation is not enabled for a scope that has had an associated fault handler invoked.

Explicit fault handlers attached to a scope provide a way to define a set of custom fault-handling activities, defined by `catch` and `catchAll` constructs. Each `catch` construct is defined to intercept a specific kind of fault, defined by a fault `QName`. If the fault name is missing, then the catch will intercept all faults with the same type of fault data. A `catchAll` clause can be added to catch any fault not caught by a more specific fault handler.

```
<faultHandlers>
   <catch faultName="QName"?
      faultVariable="BPELVariableName"?
      ( faultMessageType="QName" | faultElement="QName" )?>*
      activity
   </catch>

   <catchAll>?
      activity
   </catchAll>
</faultHandlers>
```

There are various sources of faults in WS-BPEL. A fault response to an `invoke` activity is one source of faults, where the fault name and data are based on the definition of the fault in the WSDL operation. A `throw` activity is another source, with explicitly given name and/or data. WS-BPEL defines several standard faults with their names, and there may be other platform-specific faults such as communication failures.

*BPEL summary:* BPEL is essentially a programming language. However it has some features that are specially tailored to make it easier to build robust systems that can recover from a variety of faults. It includes features for expressing internal concurrent activities; they should however be used with care, because it is not always easy to comprenhed the interaction with compensations and fault handlers.

## III. Analyzing Web Service Contract

Having described all the elements of specifications, we now present the translation to automata. In order to perform this translation, we note that WS-CDL and WS-BPEL are XML based languages for describing Web Services. The timed automata formalism we use is UppAal [25]; and it is represented by another XML document, thus, the translation has been developed with XSLT [26], XML Style sheets Language for Transformation, which is a language for transforming XML documents into other XML documents.

Figure 4 shows how the translation works: we have created some XSL style sheets, where we use XSLT instructions to extract the information from the WS-CDL document, and then the UppAal document is automatically generated. This document can be opened with the UppAal tool, and thus, we can use the model-checker of UppAal to verify some properties of interest. The tool can also run simulations of the model. We have also created some XSL style sheets to perform the same translation for WS-BPEL documents.

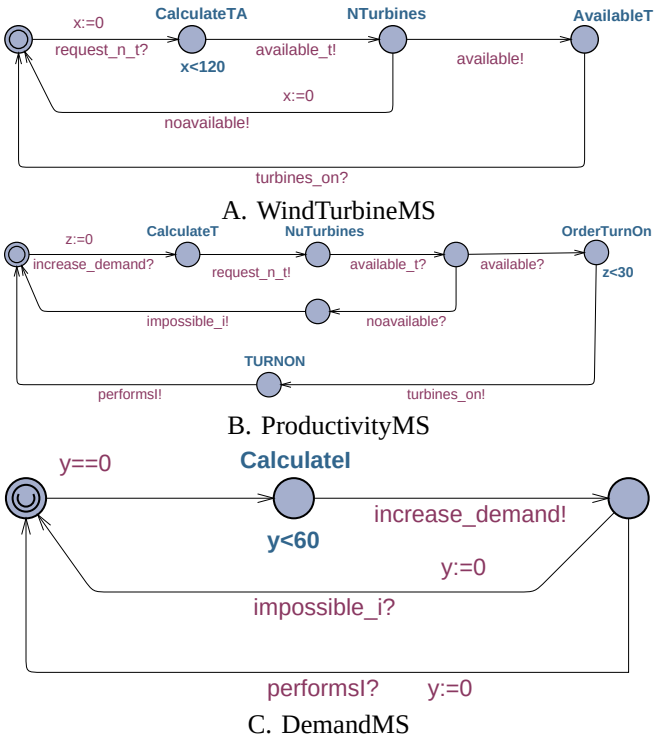



Fig. 4. Wind Mill Management System modeled in UppAal

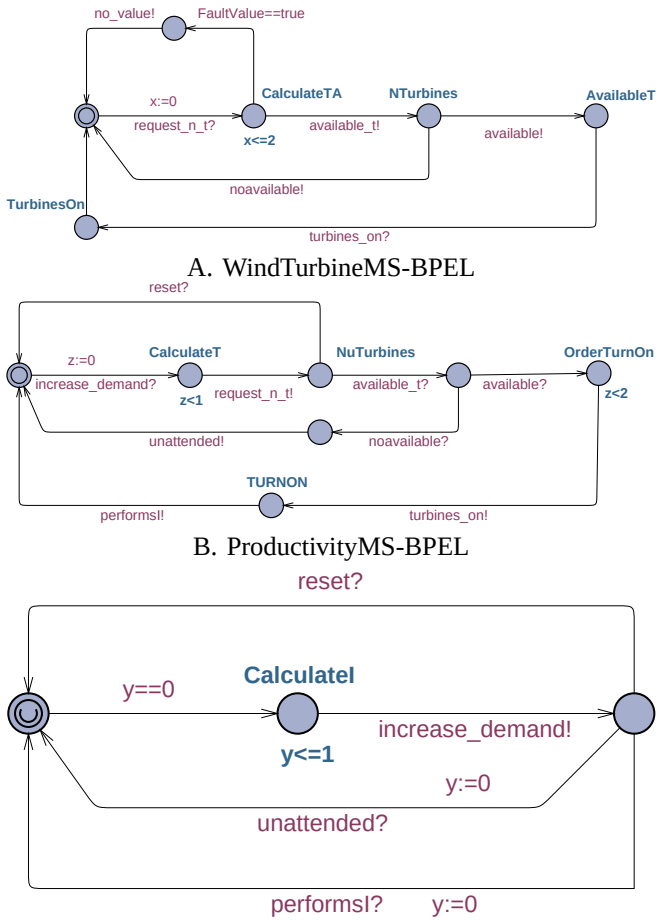



Fig. 5. Wind Mill Management System modeled in UppAal - from BPEL

For the two aspects we can check the following.

*General Properties:* We check the absence of deadlock for the CDL and for the BPEL; thus we check that the system is able to progress from start to termination; in UppAal this is easily formulated:

$A[]not\ deadlock$

This property holds for both systems.

The system should also be useful. If there are enough available turbines to fulfill the increase of demand, then the Productivity Management system shall send the command to turn on some of them to the Wind Turbine management system. This is formulated as the invariant that says that for all computations ($A$) and for all states ([]), the two automata locations coincide:

$A[]\ WindTurbineMS.AvailableT \rightarrow ProductivityMS.OrderTurnOn$

This example prpoerty holds as well.

*Meeting the demand:* Here we check for a BPEL property that the methods can be executed satisfying the contracts or generating the exceptions. For instance, when the demand system sends a message to the productivity system, because it detects an increase in the power demand (the message $increase\_demand$). Also, the Wind Turbine Management system always sends the number of available turbines on Productivity Management system's demand. This is represented in UppAal as follows:

$A[]\ ProductivityMS.NuTurbines \rightarrow WindTurbineMS.CalculateTA$

which holds as well.

*Model checking summary:* The form of checking that has been shown above is really exhaustive testing. Analysis of what properties to check depends on a systematic inspection of both requirements and the design by some review process, for instance Software Reviews, Code Inspections, and other proactive management processes whose purpose is to eliminate or to find and remove errors in product design as early as possible.

## IV. Consistency Checking - Simulation

To check whether the two individually derived models are consistent, we use the concept of (bi-)simulation. A (bi-)simulation is an equivalence relation between state transition systems, associating systems which behave in the same way in the sense that one system simulates the other and vice-versa. The automata generated from the two contract aspects specification systems (WS-CDL, WS-BPEL) turn out to be bi-similar in the following aspects:

- they both accept the same operation sequence; since the WS-CDL specified the protocols, while WS-BPEL contains the operation names but with more information.
- they also accept the same message sequence. Thus, the state that receives the message (e.g. $increase\_demand$

in the example in Figure 4) is followed by a state that sends the message ($request\_n\_t$) inn both automata. The automaton from WS-BPEL may contain some internal states.

We use another model checking tool CWB-NC to check the consistency. We first map the contract captured by both BPEL and CDL to CCS [27], one of the the design languages for CWB, which has communication similar to UppAal; actually, UppAal was developed by people who had prior experience with CCS and the Concurrency Workbench. With the analogous roots, we have not found it useful to spend much time on whether this simple mapping preserves the semantics; it is fairly obvious that it does. More languages such as timed actions version of CCS, CSP, basic lotos, etc are supported as well in the CWB tool; it performs model checking, preorder checking and equivalence checking. As mentioned above, we focus on equivalence checking which allows to identify the behaviourally/observationally equivalent states in a system.

One may ask, why CWB is not used throughout the analysis, since it includes model checking. The answer lies in the lack of state variables; CWB can model the communication structure only, whereas UppAal supports state variables with bounded domains as well as clocks.

*Translation from Uppaal to CWB CCS (CDL):* We translate the contract specification models in UppAal to a process algebra CCS to allow us to check consistency. The Wind Mill management system consists of 3 processes as shown below:

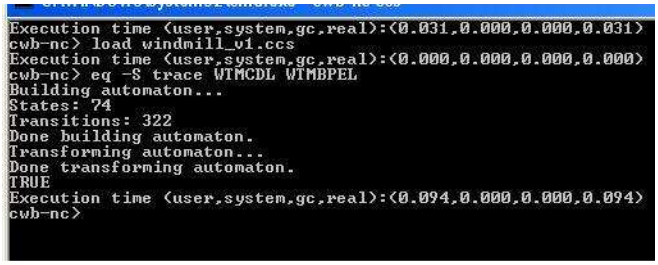
```
proc WTMCDL = (WMC | DMC | PMC)\
{request_n_t, available_t,
 noavailable, available,
 increase_demand, unattended,
 performsI}
```

Processes WMC, DMC, and PMC correspond to Windturbine management system, demand management system and productivity management system respectively as modeled in Figure 5. The three processes communicate through synchronization events. For instance, $request\_n\_t$ in Windturbine management and productivity management.

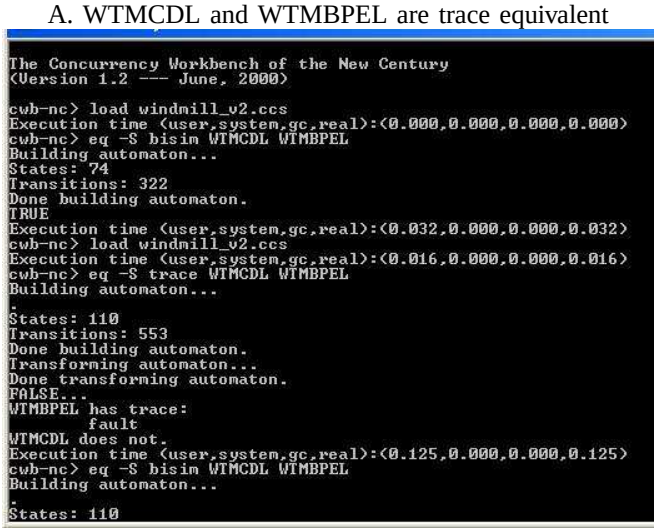
*Translation from Uppaal to CWB CCS (BPEL):* Similar to the translation of CDL, we translate the contract specification models in UppAal to a process algebra CCS. However, we have more processes from the BPEL contract specifications. These additional processes are fault handlers, compensation handlers and event handlers; but we focus on a fault handler. One can easily add other processes without violating consistency, since they are abstracted away when checking against CDL. In this case, the Wind Mill management system consists of 4 processes as shown below:

```
proc WTMBPEL = (WMC | DMC | PMC | FH)\
{fault, reset
 request_n_t, available_t,
 noavailable, available,
 increase_demand, unattended, performsI}
```

Processes WMC, DMC, and PMC correspond to windturbine management system, demand management system and productivity management system respectively as modeled in Figure 5. The three processes communicate through synchronization events. For instance, *request_n_t* in windturbine management and productivity management.




A. WTMCDL and WTMBPEL are trace equivalent




B. WTMCDL and WTMBPEL are trace bisimilar but not with fault handling

Fig. 6. Consistency Checking using CWB-NC

*The simulation results:* Figure 6 shows the result of bisimilarity check between CDL and BPEL. The first check, `eq -S bisim WTMCDL WTMBPEL` checks that they are bisimilar. The system has 74 states and 322 transitions. The CWB-NC reports that the processes are bisimilar as well as trace equivalent as shown in Figure 6 A. Recall that the fault handling events are hidden. Hence the bisimilarity. However, when the fault handler is made part of the system, the CWB-NC reports as expected that they are not trace equivalent. The lower part of Figure 6 B shows this result of checking that the two processes are trace equivalent. It shows that the result is $FALSE$ with an additional information that WTMBPEL has trace: $fault$ while WTMCDL does not. Therefore we note that CDL can only be consistent with an abstract version of BPEL where fault handlers are hidden.

## V. Comparison with other approaches

Several model checking approaches has been employed to provide some form of analysis. An illustrative example which is well-explained is [28]. It deals with specification in only BPEL where both the abstract model and executable model are specified. The approach is based on Petri nets where a communication graph is generated representing the

process's external visible behaviour. It verifies the simulation between concrete and abstract behaviour by comparing the corresponding communication graphs.

Abouzaid and Mullins [29] propose a BPEL-based semantics for a new specification language based on the $\pi$-calculus, which will serve as a reverse mapping to the $\pi$-calculus based semantics introduced by Lucchi and Mazzara [30]. The mapping in this work is implemented in a tool integrating the toolkit HAL and generating BPEL code from a specification given in the BP-calculus. Unlike in our approach, this work covers the verification of BPEL specifications through the mappings while the consistency of the new language and the generated BPEL code is yet to be considered. As a future work, the authors plan to investigate a two way mapping. We expect that our approach will be useful in this setting by taking care of the consistency part of their approach.

In [31] the authors have presented an approach different from model checking: a state propagation approach. It uses preconditions and postconditions, and computes weakest execution states. The authors argue that descriptions of preconditions and postconditions are easier and more intuitive compared to linear temporal logic formulae for example. However, similar to the above mentioned approaches, only one language is considered. In this case, consistency checking of Web service function invocations using OWL-S metadata descriptions.

Compared to our approach, the final goal is similar: that is checking of consistency. However, there are some differences in the approach. First, our approach considers more than one language. This is because CDL has a more detailed capture of abstract processes compared to the BPEL abstract processes. Further, BPEL is a programming language to specify the behavior of a participant in a choreography whereas choreography is concerned with describing the message interchanges between participants. In addition, a choreography definition can be used at design time by a participant to verify that its internal processes will enable it to participate appropriately in the choreography. With this, certain properties of individual services can be verified as well as verifying the consistency between the protocols in both BPEL and CDL. This can also be extended with some domain specific languages.

## VI. Conclusion

We have presented an approach for the analysis of web service contracts which uses model checking as its prime tool. The analysis is kept manageable by separating contract aspects and analyzing them individually. The price we pay for this aspect oriented analysis is a check for consistency between the individually derived models. However, this check by setting up a bi-simulation between automata can perhaps be automated, because the configurations of the two automata are systematically related through naming conventions and similarities in the WS-CDL and WS-BPEL constructs. The ideas are illustrated with an example specification of a Wind Turbine Management System which consists of three major components (with their services).

In the current contribution, we demonstrate the approach using timed automata as used in the UppAal tool [25], but in other contexts [32] we have experimented with using JML [33] for the functional aspects. We have not touched on verification of timing aspects, although this work was initiated in [9]. Thus the use of UppAal is to some extent a practical decision. We feel that it is well justified for the kinds of analyses that we discuss, because they are concerned with checking the properties of the service as such. For checking implementation conformance, it may not be ideal, and a translation to JML may be much more useful, in particular since Java may be an underlying implementation language, and JML is a formal specification language tailored to Java. Its basic use is thus the formal specification of the behavior of Java program modules. This direction is, however, not the main line of our investigation. The immediate work facing us is to streamline the tool fragments developed for these experiments, and in particular to make true the claim that the bi-simulation can be integrated in a more automated analysis process. It is well known that model checking has its limits, and investigations are also being done of theorem proving approaches [34] which may be more suitable for full implementation conformance checking.

## Acknowledgment

The second author is funded by the Nordunet3 Project "Contract-Oriented Software Development for Internet Services".

## References

[1] T. Erl, *Service-Oriented Architecture: Concepts, Technology, and Design*. Prentice Hall PTR, 2005.

[2] Y. Lafon and N. Mitra, "SOAP Version 1.2 Part 0: Primer (Second Edition)," W3C, W3C Recommendation, Apr. 2007, http://www.w3.org/TR/2007/REC-soap12-part0-20070427/.

[3] S. Seely, *SOAP: Cross Platform Web Service Development Using XML*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2001, foreword By-Kent Sharkey.

[4] A. Karmarkar, M. Gudgin, M. Hadley, Y. Lafon, J.-J. Moreau, H. F. Nielsen, and N. Mendelsohn, "SOAP version 1.2 part 1: Messaging framework (second edition)," W3C, W3C Recommendation, Apr. 2007, http://www.w3.org/TR/2007/REC-soap12-part1-20070427/.

[5] B. Meyer, *Object-oriented software construction (2nd ed.)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1997.

[6] J. C. Okika and A. P. Ravn, "Classification of SOA Contract Specification Languages." in *Proceedings of The IEEE International Conference on Web Services (ICWS)*, Sep. 2008, to appear.

[7] T. Andrews, F. Curbera, H. Dholakia, Y. Goland, J. Klein, F. Leymann, K. Liu, D. Roller, D. Smith, S. Thatte, I. Trickovic, and S. Weerawarana, *BPEL4WS, Business Process Execution Language for Web Services Version 1.1*, IBM, 2003. [Online]. Available: http://download.boulder.ibm.com/ibmdl/pub/software/dw/specs/ws-bpel/ws-bpel.pdf

[8] N. Kavantzas, D. Burdett, G. Ritzinger, T. Fletcher, and Y. Lafon, "Web services choreography description language version 1.0," W3C, W3C Working Draft, Dec. 2004, http://www.w3.org/TR/2004/WD-ws-cdl-10-20041217/.

[9] G. Diaz, J. J. Pardo, M. E. Cambronero, V. Valero, and F. Cuartero, "Verification of Web Services with Timed Automata," in *Proceedings of First International Workshop on Automated Specification and Verification of Web Sites*, vol. 157. Springer Verlags Electronics Notes in Theoretical Computer Science series, 2005, pp. 19–34.

[10] E. Cambronero, J. C. Okika, and A. P. Ravn, “Analyzing Web Service Contracts - An Aspect Oriented Approach.” in *Proceedings of the International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies (UBICOMM’2007)*. IEEE Computer Society Press, November 2007, pp. 149 – 154.

[11] G. Castagna, N. Gesbert, and L. Padovani, “A theory of contracts for web services,” in *PLAN-X ’07, 5th ACM-SIGPLAN Workshop on Programming Language Technologies for XML*, jan 2007.

[12] S. Carpineti, G. Castagna, C. Laneve, and L. Padovani, “A formal account of contracts for Web Services,” in *WS-FM, 3rd Int.l Workshop on Web Services and Formal Methods*, ser. LNCS, no. 4184. Springer, 2006, pp. 148–162.

[13] H. Davulcu, M. Kifer, and I. V. Ramakrishnan, “CTR-S: A Logic for Specifying Contracts in Semantic Web Services,” in *Proceedings of WWW2004*, May 2004, pp. 144–153.

[14] G. Pu, X. Zhao, S. Wang, and Z. Qiu, “Towards the semantics and verification of bpel4ws,” *Electr. Notes Theor. Comput. Sci.*, vol. 151, no. 2, pp. 33–52, 2006.

[15] D. Reeves, B. Grosof, M. Wellman, and H. Chan, “Toward a declarative language for negotiating executable contracts,” in *In Proc. AAAI-99*, 1999.

[16] C. Prisacariu and G. Schneider, “A formal language for electronic contracts,” in *9th IFIP International Conference on Formal Methods for Open Object-Based Distributed Systems (FMOODS’07)*, ser. Lecture Notes in Computer Science, M. Bonsangue and E. B. Johnsen, Eds., vol. 4468. Springer, June 2007, pp. 174–189.

[17] A. Beugnard, J.-M. Jezequel, N. Plouzeau, and D. Watkins, “Making Components Contract Aware,” *Computer*, vol. 32, no. 7, pp. 38–45, 1999.

[18] B. Meyer, *Eiffel: the language*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1992.

[19] R. Heckel and M. Lohmann, “Towards contract-based testing of web services,” 2004.

[20] A. Keller and H. Ludwig, “The WSLA Framework: Specifying and Monitoring Service Level Agreements for Web Services,” *Journal of Network and Systems Management*, vol. 11, no. 1, pp. 57–81, March 2003.

[21] “Web Services Agreement Specification (WS-Agreement),” https://forge.gridforum.org/projects/graap-wg/document/WS-AgreementSpecification/en/7, 2004.

[22] “Web Services Architecture,” W3C Working Group Note, www.w3.org/TR/ws-arch/, Feb 2004.

[23] E. Christensen, F. Curbera, G. Meredith, and S. Weerawarana, *Web Services Description Language (WSDL) 1.1*, 1st ed., W3C, March 2001, URL: http://www.w3c.org/TR/wsdl.

[24] D. Booth and C. K. Liu, “Web services description language (WSDL) version 2.0 part 0: Primer,” W3C, Candidate Recommendation, March 2006.

[25] J. Bengtsson, K. G. Larsen, F. Larsson, P. Pettersson, and W. Yi, “UPPAAL in 1995,” in *Tools and Algorithms for Construction and Analysis of Systems*, 1996, pp. 431–434. [Online]. Available: citeseer.ist.psu.edu/article/bengtsson96uppaal.html

[26] J. Clark, “XSL Transformations (XSLT) Version 1.0,” W3C, Tech. Rep. REC-xml-19980210, 1998, http://www.w3.org/TR/xslt. [Online]. Available: citeseer.nj.nec.com/bray98extensible.html

[27] R. Milner, *Communication and Concurrency*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1989.

[28] A. Martens, “Consistency between executable and abstract processes,” in *EEE ’05: Proceedings of the 2005 IEEE International Conference on e- Technology, e-Commerce and e-Service (EEE’05) on e-Technology, e-Commerce and e-Service*. Washington, DC, USA: IEEE Computer Society, 2005, pp. 60–67.

[29] F. Abouzaid and J. Mullins, “A calculus for generation, verification and refinement of bpel specifications,” *Electron. Notes Theor. Comput. Sci.*, vol. 200, no. 3, pp. 43–65, 2008.

[30] R. Lucchi and M. Mazzara, “A pi-calculus based semantics for ws-bpel,” *J. Log. Algebr. Program.*, vol. 70, no. 1, pp. 96–118, 2007.

[31] T. Kaizu, T. Noro, and T. Tokuda, “A state propagation method for consistency checking of web service function invocations in web applications,” in *ICWE ’06: Workshop proceedings of the sixth international conference on Web engineering*. New York, NY, USA: ACM, 2006, p. 18.

[32] GI-Dagstuhl, “Modelling contest: Common component modelling example (cocome).” [Online]. Available: http://agrausch.informatik.uni-kl.de/CoCoME

[33] J. Leavens, “JML’s rich, inherited specification for behavioural subtypes,” in *Proc. 8th International Conference on Formal Engineering Methods (ICFEM06)*, ser. LNCS, vol. 4260. Springer, 2006.

[34] P. Giambiagi, O. Owe, A. P. Ravn, and G. Schneider, “Language-based support for service oriented architectures: Future directions,” in *ICSOFT (1)*, J. Filipe, B. Shishkov, and M. Helfert, Eds. Setúbal, Portugal: INSTICC Press, September 2006, pp. 339–344.

## APPENDIX A: WS-CDL DESCRIPTION OF THE DEMAND MANAGEMENT SYSTEM

```
<?xml version="1.0" encoding="UTF-8"?>
<package author="SCTR Group" name="" version="1.0">

<token name="WindTurbineRef" informationType="StringType"/>
<token name="ProductivityRef" informationType="StringType"/>
<token name="DemandRef" informationType="StringType"/>

<roleType name="WindTurbineRoleType">
<description type="description"/>
<behaviour name="WindTurbineBehaviour"/>
</roleType>

<roleType name="ProductivityRoleType">
<description type="description"/>
<behaviour name="ProductivityBehaviour"/>
</roleType>

<roleType name="DemandRoleType">
<description type="description"/>
<behaviour name="DemandBehaviour"/>
</roleType>

<relationship name="DemandProductivity">
<role type="DemandRoleType"/>
<role type="ProductivityRoleType"/>
</relationship>

<relationship name="ProductivityWindTurbine">
<role type="ProductivityRoleType"/>
<role type="WindTurbineRoleType"/>
</relationship>

<channelType name="Demand2ProductivityChannelType">
<role type="ProductivityRoleType"/>
<reference>
<token name="ProductivityRef"/>
</reference>
</channelType>

<channelType name="Productivity2WindTurbineChannelType">
<role type="WindTurbineRoleType"/>
<reference>
<token name="WindTurbineRef"/>
</reference>
</channelType>

<choreography>
<relationship type="DemandProductivity"/>
<relationship type="ProductivityWindTurbine"/>

<variableDefinitions>
<variable name="Demand2ProductivityChannel"
  channelType="Demand2ProductivityChannelType"/>
<variable name="Productivity2WindTurbineChannel"
  channelType="Productivity2WindTurbineChannelType"/>

<variable name="Available" informationType="xsd:boolean"
  roleTypes="Productivity"/>
<variable name="WindTurbineClock"
  informationType="tns:Clock" roleTypes="WindTurbine"/>
<variable name="DemandClock" informationType="tns:Clock"
  roleTypes="Demand"/>
<variable name="ProductivityClock"
  informationType="tns:Clock" roleTypes="Productivity"/>
<variable name="detectedincreaseDone"
   informationType="tns:boolean" roleTypes="Demand"/>
</variableDefinitions>

<assign roleType="Productivity">
```

```
<copy name="Available_assign">
<source expression="true"/>
<target variable="Available"/>
</copy>
</assign>

<assign roleType="Demand">
<copy name="detectedincrease">
<source expression="false"/>
<target variable="detectedincreaseDone"/>
</copy>
</assign>

<sequence>
  <workunit name="demand increase detected"
    guard="cdl:equal(
       cdl:getVariable('tns:DemandClock'),
          '','',),'0:00')" block="true">
  <assign roleType="DemandRoleType">
  <copy name="calculateincrease"
        causeException="true">
  <source variable="true"/>
  <target variable=
     "cdl:getVariable('detectedincreaseDone',
        '','')"/>
  </copy>
  </assign>
</workunit>

  <interaction name="Demand management system"
    operation="sendIncreasing"
    channelVariable="Demand2ProductivityChannel">
  <participate relationshipType="DemandProductivity"
    fromRole="DemandRoleType"
    toRole="ProductivityRoleType"/>
  <exchange name="CalculatedIncreasing"
    action="request"/>
  <timeout time-to-complete= "cdl:minor(
      cdl:getVariable('tns:DemandClock',
          '','',),'0:01')"/>
  </interaction>

  <interaction name="RequestTurbines_interaction"
    operation="RequestTurbines"
    channelVariable="Productivity2WindTurbineChannel">
  <participate
    relationshipType="ProductivityWindTurbine"
    fromRole="ProductivityRoleType"
    toRole="WindTurbineRoleType"/>
  <exchange name="RequestTurbinesExchange"
    action="request"/>
  <timeout time-to-complete= "cdl:minor(
      cdl:getVariable('tns:ProductivityClock','',''),
      '0:02')"/>
  </interaction>

  <interaction name="AvailableTurbines_interaction"
    operation="AvailableTurbines"
    channelVariable="Productivity2WindTurbineChannel">
  <participate
    relationshipType="WindTurbineProductivity"
    fromRole="WindTurbineRoleType"
    toRole="ProductivityRoleType"/>
  <exchange name="AvailableTurbinesExchange"
    action="request"/>
  </interaction>

  <choice>
    <workunit name="alt_else1_if"
      guard="Available == true" block="true">
      <interaction name="TurbinesOn_interaction"
        operation="TurbinesOn"
        channelVariable="Productivity2WindTurbineChannel">
      <participate
        relationshipType="ProductivityWindTurbine"
        fromRole="ProductivityRoleType"
        toRole="WindTurbineRoleType"/>
      <exchange name="TurbinesOnExchange"
        action="request"/>
      </interaction>
    </workunit>
    <workunit name="alt_else1_else"
      guard="Available != true" block="true">
      <interaction name="Imposible_interaction"
        operation="Impossible"
        channelVariable="Demand2ProductivityChannel">
      <participate relationshipType="ProductivityDemand"
        fromRole="ProductivityRoleType"
        toRole="DemandRoleType"/>
      <exchange name="ImposibleExchange"
        action="request"/>
      </interaction>
    </workunit>
  </choice>
</sequence>
</choreography>
</package>
```

## Appendix B: CCS Description of the Wind Mill Management system in CDL and BPEL

```
*********************************************
*  This models the Wind Mill Management System
*
*  CDL system is consistent with abstract BPEL
*
*********************************************

****  CDL Specification Description  ***********

proc WTMCDL = (WMC | DMC | PMC)\
{request_n_t, available_t,
noavailable, available,
increase_demand, unattended, performsI}
*********************************************

proc WMC =
request_n_t.'available_t.('noavailable.WMC
+ 'available.WMC)

proc PMC =
increase_demand.'request_n_t.available_t.
(available.'performsI.PMC
+ noavailable.'unattended.PMC)

proc DMC =
increase_demand.(unattended.DMC + performsI.DMC)

************************ BPEL  ******************

proc WTMBPEL = (WMC | DMC | PMC | FH)\{fault, reset
request_n_t, available_t,
noavailable, available,
increase_demand, unattended, performsI}
*************************************************

proc FH = fault.'reset.FH

proc WMB =
request_n_t.('novalue.WMB + 'available_t.
('noavailable.WMB + 'available.turbines_on.WMB))

proc PMB =
increase_demand.'request_n_t.
('reset.PMB + (available_t.
(available.'turbines_on.'performsI.PMB
+ noavailable.'unattended.PMB)))

proc DMB =
increase_demand.('reset.DMB +
(unattended.DMB + performsI.DMB))
```

# Exception Based Enterprise Rights Management : Towards a Paradigm Shift in Information Security and Policy Management

Jean-Henry Morin

University of Geneva – CUI

*Jean-Henry.Morin@unige.ch*

## Abstract

*Enterprise DRM is still dominated by vendor driven proprietary approaches fundamentally lacking interoperability features and essentially relying on strong cryptography lacking the flexibility to accommodate unanticipated work situations requiring exceptional actions. Consequently users increasingly circumvent corporate security policies just to get their work done and such incidents simply go unnoticed. From a management and security point of view this represents a risk in an increasingly compliance driven and networked economy. This paper explores the opportunity to apply an exception-based model for Enterprise DRM building on the proposition that monitoring security policies could be as effective as strong enforcement and provide more accurate information to manage and tune corporate digital policies.*

***Keywords***: DRM, Exception Management, Monitoring

## 1. Introduction

This study draws on two research streams in the field of DRM. First in the media DRM sector trying to address the hard problem of managing rights for digital artifacts in ways allowing to accommodate for fair use (i.e., supporting the Copyright Balance Principle [1]). Second in the Enterprise DRM sector where these technologies gained much visibility following corporate scandals to help address governance, risk and compliance issues (GRC) [3].

The key question underlying this study stems from exactly the same initial questions raised in the media sector. Namely, is Enterprise DRM (and by extension information-centric security) following the wrong path with the wrong assumptions? This is what led to designing a model for managing exceptions in DRM environments [3] hypothesizing that the users weren't criminals *a priori*. Both areas appear to share similar properties but for different reasons. The main contribution of this paper is to raise the issue in similar terms in the corporate sector and to propose applying our model in Enterprise DRM environments as a feature enabling better usability and efficiency, increased traceability and monitoring of legitimate uses instead of untraceable security policy circumvention and ultimately a way for security professionals to tune policies based on real usage patterns.

This paper is structured as follows. After further describing the problem, section 2 presents the Exception Management model. The application of the proposed model is discussed in section 3. A possible architecture is described in section 4. Section 5 outlines related work. Concluding remarks and future work are presented in section 6.

### 1.1. Issues and objectives

While the market of information-centric security has now matured to a point where Enterprise DRM is a known technology, this industry is still struggling with interoperability issues. Solutions are still proprietary, offering limited mechanisms for generic interoperability among them. Assuming organizations increasingly need to engage in ad-hoc, short-lived and dynamic collaborations requires these systems to be able to accommodate such exchanges across organizations not necessarily having the same Enterprise DRM system.

While this is a critical issue and an enabling factor for the broad endorsement and deployment of Enterprise DRM based systems, there still remains a hard problem to be addressed. How do DRM enabled systems manage or are able to deal with so called exceptions? In order to further emphasize this critical issue, let us illustrate this issue in the media sector before transposing it to the corporate environment.

Let's start with the *Copyright Balance* principles that should underline public policy regarding DRM as proposed by E. Felten in a column of CACM [1]: "*Since lawful use, including fair use, of copyrighted works is in the public interest, a user wishing to make lawful use of copyrighted material should not be prevented from doing so by any DRM system.*". This sound principle is exactly at the forefront of our work making the case for such "Exception Provisioning" in DRM enabled systems.

Drawing on this principle and applying it to the corporate environment for information security leads to defining the *Enterprise Security Balance Principle* :

"*When legitimate use of, or access to, managed or secured corporate resources is in the interest of the company, an employee or business partner wishing to do so should not be prevented from doing so by any Enterprise DRM or security system.*"

Now, contrary to the initial principle that applies essentially to the media and entertainment sector with respect to lawful and fair use rights any individual may claim, the above-derived principle is idealistic and irresponsible given the much different nature of corporate resources. As a result we need to augment it with an additional property. Namely requiring that an auditable trace be systematically logged. Consequently, the revised *Enterprise Security Balance Principle* becomes:

"*When legitimate use of, or access to, managed or secured corporate resources is in the interest of the company, an employee or business partner wishing to do so should not be prevented from doing so by any Enterprise DRM or security system provided an auditable trace be systematically logged.*"

To further support our proposition and our assumption, let's review a few facts and figures from the industry. There is very little evidence about circumvention of corporate security policies for obvious reasons that in most cases such incidents go unnoticed unless problems occur thus revealing the incidents. However, recently these questions appear to be increasingly studied in the light of risk and compliance issues. For example, a recent survey from EMC's RSA security division [4] shows interesting results. According to the survey, 53 % admit working around corporate security policies just to get their work done. Another interesting figure comes from a Cisco white paper based on a survey conducted among 2000 IT professionals in 10 countries [5] reveals among the top reasons for violating corporate IT policies are that (a) it doesn't match the reality and what is needed to do their job, (b) they need to access applications not included in the company's IT policy to get their job done.

Such figures are clear indications of a problem and mismatch between corporate security policies and the actual day-to-day operations where regular employees are led to circumventing these security policies just to be able to accomplish their work. What does this mean for the employees, the company and security professionals?

For the employees, we can clearly imagine the amount of extra burden put on them in situations where they ultimately need to be "creative" to do their job. Consequently, this lack of usability may lead to additional stress with respect to their responsibility when "breaking the rules". Moreover, this leads to additional inefficiencies and most importantly untraceable policy transgressions. All this has a direct cost for the company in addition to the increased level of risk for the company (e.g., data leakage, compliance, undocumented actions, etc.). Ultimately, the security professionals have no way to monitor such incidents in order to evolve and tune corporate security policies according to the actual needs of the company and its employees.

This in turn raises another question about the underlying assumption of corporate security. Until now, most of the enterprise security is following a "closed" model whereby anything that isn't explicitly authorized is forbidden. Enterprise DRM follows the same pattern basically persistently protecting content using strong cryptography thus forcing employees to potentially circumvent security policies and procedures in order to accommodate day-to-day operations that oftentimes haven't been anticipated and factored in the policies. Such examples are numerous and include sharing passwords and accounts, using removable media, etc.

This approach suffers from the same limitations found in the media DRM sector criminalizing the user / employee by default. In other words, not trusting him. We argue that one should put back the trust where it belongs. Shouldn't employees be trusted unless otherwise witnessed? By all means, if a company has employed someone, it has placed trust in this person. When an employees' judgment commands to do something, he usually is accountable for it. Now, using backdoors definitely worsens the problem while one might simply argue that if an employee claims he needs to do something, he knows best.

This is exactly the motivation behind the idea of introducing exception management in Enterprise DRM. Anyone claiming he has the right to do something should *a priori* be trusted provided he is willing to leave a trace for monitoring and accountability. This represents a major paradigm shift in how we approach security. Most people are trustworthy and consequently security shouldn't be a constraint (enforced) but rather a help (monitoring).

### 1.2. Using Credentials for Exceptions

Our approach is based on using some form of credentials whereby a DRM module would provide an entry point to evaluate locally held credentials that could have precedence over the attached rules and be traceable (i.e., auditable). The process could be rather straightforward as it would be comparable to the existing verification of locally held licenses in the users' license-store. For example, let's imagine that a new employee is provided with such a credential showing he is affiliation and status together with other administrative tokens. Such credentials would be stored on the users computer (e.g. in a credential store) and made available to the DRM module (enforcement point) when evaluating rights at runtime.

This rather elegant approach allows to potentially handling many situations where explicit policy specification would simply be too cumbersome or simply impossible to anticipate and formalize. In the case of fair use, it is commonly agreed that non-commercial use of copyrighted material in academic environments is free. Being a faculty or a student would allow having an academic credential delivered by the university.

In a general way, such an approach allows to capture generic rights management in the form of groups or communities. Being a member of a group provides a generic right with respect to content when accessed by its members. Further refinement could consider a hierarchy of credentials for example within a company where management would be provided credentials with broader rights than those of staff members.

## 2. The Exception Management Model

The proposed model presented in detail in [3] involves two additional entities to traditional DRM based environments: a Credential Manager and an Exception Manager.

The Credential Manager is an entity that emits, revokes and manages credentials. It can be any structure, such as an enterprise, an academic entity, or a national entity. It does not have to be known by the Content Owner neither at credential generation time, nor at content creation time; but it has to be able to prove its legitimate existence as well as the motivation leading to generating credentials.

The Exception Manager is an extension of the traditional License Manager found in all DRM based environments. It verifies if a credential may qualify to give access to a piece of rights enabled content. The Exception Manager checks if the credential is valid, if it has not been revoked and if it may be applicable to the content. Thus it verifies if the Credential Manager has legal existence and evaluates the reasons that led to generating the specific credential. If the credential passes these verifications, a Short-Lived License may be granted providing access to the content for a limited time. Moreover, the operation is logged as a trace for further proof of legitimate activity. Short-Lived Licenses are thus meant to give an exceptional access to content, and their validity is thus limited in time. They can give more or less rights depending on the type of the detected exception and some optional metadata information attached to the content indicating specific constraints on the Short-Lived License.

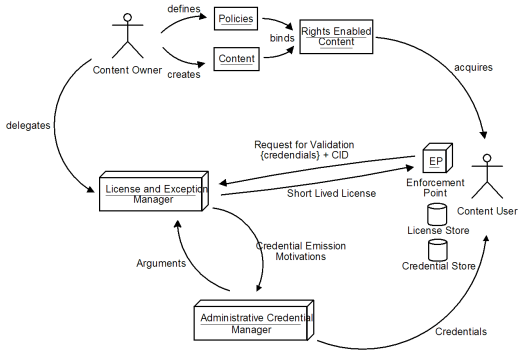



Figure 1. Exception-based Model.

As a general overview of the model, Figure 1 highlights the main difference with a traditional DRM model. First, the content users obtain credentials from Credential Managers. These credentials are then stored in a credential store alongside the local license store to be used by the enforcement point. Compared to a classical DRM model where the enforcement point only has the choice to grant or denying access or eventually try to acquire a license, in the credential based model, credentials held by users can be sent to the exception manager and used to check if the user

qualifies for an exception. If so a corresponding Short-Lived License is issued and returned for use.

As a result, content protection, credential creation, exception verification and corresponding authorization are decoupled. This approach provides greater flexibility than the classical DRM model allowing Credential Managers unknown to content owners to inform Enforcement Points that an exceptional situation may be taken into consideration in situations where the user has no explicit rights to access the content in the form of a traditional license.

While providing flexibility to content users this approach still gives final control to the Exception Manager by allowing it to verify several points mentioned above leading to evaluating the legitimacy of the requested exception. Content Owners only have to care about the way they wish to protect their assets, ad hoc decisions being taken by the Exception Manager in case of exceptional situations. Finally, based on the logs of the credential manager the content owners can request audits of these logs either in case of fraud suspicion or simply as a regular validation procedure of the credential manager.

Lets now describe in further details the credential based model for managing exceptions in DRM systems. We first present the specifics of the content protection process when using exceptions before describing the exception management itself.

## 2.1. Content protection

Content protection in the context of an exception based model differs from its traditional representation. This section explores the main differences introducing or refining the concepts of core policies, certification delegation, exception handling delegation and rights distribution.

**Core Policies**. At the very beginning of the content protection process the definition of policies is driven by the need to protect a content asset. But this process follows a path leading from this simple content protection to the need of having flexibility in any situation. Following this path results in producing complex policies required to deal with all particular situations that may arise.

In the proposed exception based model, only core Policies should be associated to content. Core Policies are the set of policies needed to efficiently protect the content in most situations. These policies have to reflect enterprise strategy, the most important requirements concerning the content and all usual situations that may occur. Thus policies embedded into the rights enabled content should not include other considerations, such as policies dealing with extremely rare situation consequently considered as exceptions.

In this context all policies added to provide further flexibility not in the scope of usual policies are considered as potentials exceptions and should thus be handled using the credentials based exception handling model.

**Credential Properties**. Credentials have the following set of properties:

*Known Source*: Credentials must contain information about the Administrative Credential Manager who generated them, in order to be able to verify its legal existence as well as the motivations that led to credential generation.

*User Bound*: Each credential is bound to a single user or role, affiliated to the Administrative Credential Manager, able to prove that he is the legitimate owner of the credential.

*Limited validity*: Credentials are limited in time; their validity period is included in the credential.

*Revocable*: The Administrative Credential Manager can revoke a credential it has generated at any time.

Note that information about the nature of the credential, the reasons explaining why it has been created are not embedded into the credential. This approach allows to modify the scope of credentials generated by an Administrative Credential Manager for a single user, by widening the set of motivations, narrowing it or refining it, without having to revoke the credential and having to generate new ones. This provides additional flexibility, while retaining control over the number of credentials.

**Credential Generation**. In the model, generation of credentials that may lead to exceptions is delegated to Administrative Credential Managers. This indicates that credential owners can legitimately ask for the rights to access a piece of content in a given context.

Resulting credentials do not provide any direct access grant to a piece or type of content, but only indicates that even if their owner does not have the rights - in the form of a license - to access a piece of content and if the credential is recognized, he may be

entitled to the right to access the content due to an exceptional situation.

**Exception Handling Delegation**. As stated before, the goal of the credential based model is manifold. First, it provides a way to reduce the complexity and size of rights and policy managed contents. Second it provides more flexibility in handling special or unanticipated situations as content needn't be modified to deal with such situations. Finally, it simplifies the role of content owners allowing them to produce contents and protect them with the most important and representative policies, not having to deal with all possible situations.

As a result, businesses are provided with a flexible way to delegate handling of particular situations potentially allowing exceptions. In this model, exceptions are detected, verified and handled by an Exception Manager not involving directly the content producer, nor having to modify the content in order to adapt to new exceptional situations. Activity logging is done for further audit by interested parties.

## 2.2. Exception management

In this section we explore in further details the process of rights verification, exception detection and short lived license acquisition.

**Rights Verification**. A central role in the proposed exception based model is the rights verification process. As stated before, the way the enforcement point manages rights verification in our model differs from the usual way. Figure 2 depicts the underlying sequence of actions that have to be completed.

When a user wants to access content (1), the held licenses are taken from the users' license store (2) and the enforcement point tries to use them for the requested action (3). This part of the process is exactly the same as done traditionally. If existing licenses match content policies, access is granted (4a). If none of the licenses are applicable to the content, available credentials are taken from the local credential store (4b) and content identification is extracted (5). These information are signed and sent (6) with the information about the way the content is being accessed, to the Exception Manager for further verification (7). This next step tries to detect possible exceptions instead of simply denying access to the content. The enforcement point then waits for an answer which can eventually be a short lived license, if an exception is considered, and uses it (8) to then grant access to the content (9) and store the license (10) or a deny if not (11).

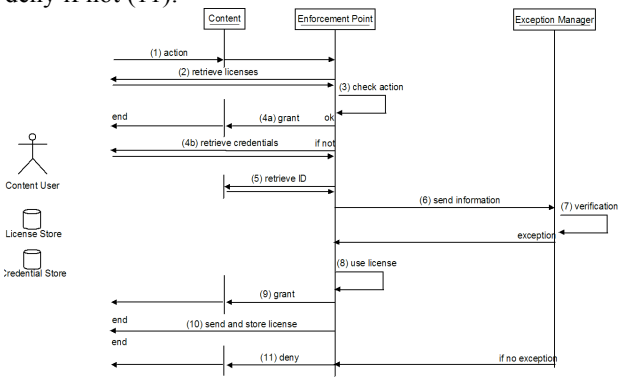



Figure 2. Rights Verification Sequence Diagram.

**Exception Detection**. When the exception manager receives the credentials, as well as content identification and the usage context, it tries to detect if a suitable combination is applicable for an exception. For each credential multiple steps are involved. These are illustrated in Figure 3.

First, the exception manager has to verify if the credential has been generated by an existing and valid Administrative Credential Manager (1). To achieve this task, the credentials have to be examined in order to retrieve information about their creator, and then verify their legal existence. The next step is to verify if the credential really belongs to the user trying to access the content (2). If it is the case, the exception manager checks if the credential is still valid (3) and asks the credential manager if it has not revoked it (4). Administrative Credential Manager verifies it (5), and then sends an answer (6). Credentials not complying with any of these rules are ignored (7). Last step is then to check if the credential can be applied to the content in the context in which the content is to be used. To do so the Exception Manager asks the Administrative Credential Manager for the motivations that have led to a credential generation (8) and the Manager sends back its signed answer (9). This answer may include textual information that can be analyzed, parsed; it may also contain any other kind of information such as a certificate emitted by a content owner indicating that a contract has been signed by both parties, or even another credential emitted by another recognized Administrative Credential Manager. If this last verification succeeds - i.e., if any of the retrieved information is accepted (10) - an exception is applicable and the short lived license acquisition process can start (11). When all credentials have been verified, a short lived license or a deny is sent back to

the enforcement point depending on the result of the process (12).

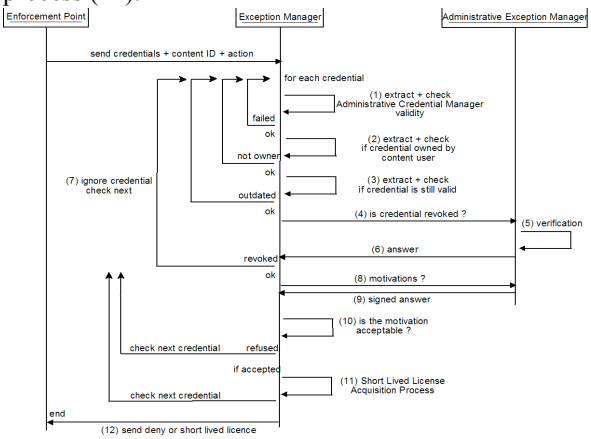



Figure 3. Exception Detection Sequence Diagram

**Short Lived License Generation**. The short lived license generation process is started when an exception has been detected and is applicable. This is a recursive process creating a license based on all exceptions that have been detected as applicable for a single access to a rights enabled content.

At this stage, the Exception Manager knows that it has to deal with an exception situation and knows what credentials have raised what kind of exception. The short lived license is built incrementally analyzing all exceptions. In order to emit such a short lived license some precautions have to be taken in order to manage issues of precedence and potential conflicting exceptions.

Figure 4 presents the different steps of this process. First, each exception has to be logged for traceability purpose (1). The log has to keep all required information to justify the exception. This includes the identification of the content, the credentials that led to an exception, the motivations signed by the Administrative Exception Manager and the context of use, i.e., the foreseen type of content access. Once all required information have been logged, the rights the specific exception may grant to the user are compared to the rights granted by previous exceptions, and the license is refined (2). Differences may occur based on the provided reasons. For instance, a first credential may raise an exception with motivation "academic use", and a second credential may indicate that there is a "research agreement with the content owner". First credential would allow limited use, but second one would allow access to additional features, or a more detailed output. Once all exceptions have been handled, the short lived license can be generated (3).

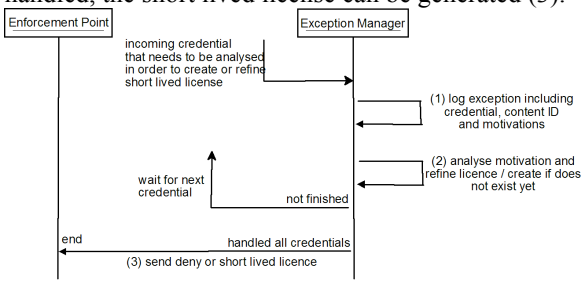



Figure 4. Short Lived License Acquisition Sequence Diagram.

The log of all exceptions is needed in order to be able to detect Administrative Credential Managers, or users abusing the system - and eventually blacklist them -, and keep a global trace of content usage.

The validity of the license will be usually short (from a single access to a few days validity) or with limited use (read only) as each credential can be revoked at any time. But the effective validity is a matter of specific policies bound to the content owner which may eventually also be set as a core policy attached to the content. The final decision is thus left to the Exception Manager responsible for this task.

## 3. Applying the Model to Enterprise DRM

Lets now put the model into perspective of Enterprise DRM. Figure 5 shows the resulting diagram.

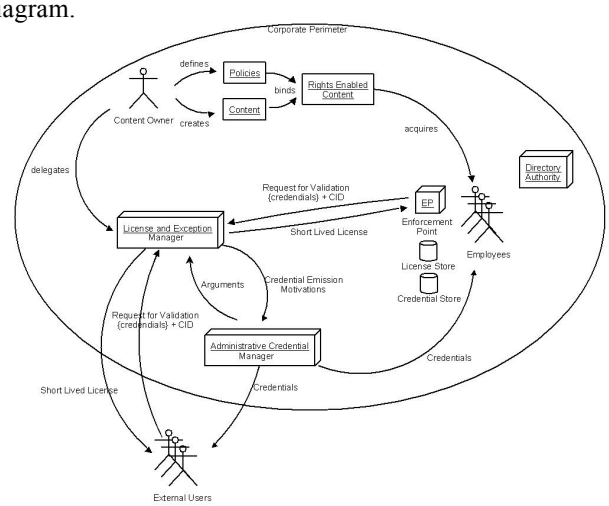



Figure 5. Enterprise Exception Based DRM

Applying it to the corporate sector appears to offer several simplifications to the model as well as some potential advantages for collaboration with external partners.

The first simplification comes from the fact that basically all the components of the architecture lie within the corporate perimeter. Content producers, owners and users are part of the same company. The only external entities being external partners with whom collaborations exist. Content producers and owners being internal production and application of policies to produce rights enabled content is much simplified. Moreover combination with enterprise wide applications and content management systems and repositories is also internal.

The DRM license server is also enterprise bound and serves the employees for all regular DRM related interactions.

Employees being part of the organization also simplifies administration in terms of having access to a corporate directory authority (e.g., LDAP, AD, etc.).

The Credential Manager is also bound to the corporate infrastructure and can easily interact with the directory authority to emit credentials for employees. It may be asked by employees to produce a Credential for an external user. In this case the credential is provided to the external user for sporadic uses on a case-by-case basis.

The Exception Manager is internal to the company and serves short-lived licenses to employees and external partners alike based on the provided credentials and exception requests. It may interact with the Credential Manager to request additional information when needed.

Every actor keeps a trace in logs of each transaction. This may be made available in real-time to security policy auditors through appropriate tools to monitor how effective policies are or in case alerts are set, to take prompt action in the event a malicious user attempts to do something highly sensitive. This is a powerful approach to managing corporate digital policies thus allowing tuning policies according to real usage situations. Moreover, management dashboards can be built to capture in real-time potential compliance risks.

## 4. Architecture Overview: Attribute Certificates

The basic idea behind the proposed approach is to make use of a credential based scheme. This raises however the issue of who and how these credentials are managed. To this end, we propose the use of PKI infrastructures which are already well established techniques. Moreover, certification authorities are accustomed to handling similarly sensitive aspects of security. The model would also perfectly fit the operation of such services with registration authorities, issuing services, revocation lists, etc.

Instead of using X.509 public key certificates (PKCs), we propose to use X.509 Attribute Certificates (ACs), RFC3281 [6], having a similar structure to PKCs without the public key. ACs can hold attributes specifying relevant information such as roles, affiliations, temporary situations or whatever is needed to evaluate exceptions.

Such credentials would be delivered to the user, together with other administrative tokens, passwords, etc., by the institution / organization to which the user is affiliated. A credential would hold several information such as a known lifetime (expiry date), a unique ID (affiliation, employee number, etc.) within the domain of the institution delivering the credential, and any other relevant information that should be used when evaluating whether or not an exception or waiver is applicable.

From thereon, the DRM system, upon deciding whether or not to render the content, could be required by the user to first check for locally held credentials. Then based on these credentials, further actions could be undertaken in order to acquire the corresponding license and thus grant the user access based on his situation. The important point to note here is that basically the content remains persistently protected. It is processed just as if it were in a situation without exception request. The rendering is done within the usual trusted renderer and basic rules, identified as mandatory for example can still be enforced. A proof of concept prototype was implemented and discussed in [26]

## 5. Related Work

To the best of our knowledge, we have not been able to find any related initiative in Enterprise DRM as it is considered to defeat the purpose. In this section we focus on highlighting projects, DRM standards and

architectures that not only consider DRM from the content owner's perspective, but also from the consumer's viewpoint. A more detailed overview of DRM evolution and key contributions, which have led to consider such issues, can be found in [7].

DRM raises issues involving different interests thus leading to often incompatible requirements of actors in the value chain. While most existing DRM solutions are content provider centric and are meant to protect their rights, there has been little attention given to the consumer side of rights management. In order to raise awareness, help reconcile these interests and to support the emergence of a common European position with respect to consumer and user issues of DRM solutions, the EU INDICARE project [8] was launched. It aimed at investigating issues like consumer acceptability of DRM systems, their interface and functionality, as well as policy issues linked to privacy and access to information. One of the main outputs of the project was its Consumer's Guide to Digital Rights Management, published in ten European languages. This guide provides concise, neutral and understandable information about what DRM is and why it matters to consumers.

The disruption to rights balance is currently illustrated by the fact that currently most DRM solutions bind content to hardware devices physically; while such an approach provides straight-forward security for content owners, it cruelly limits content usage by preventing often legitimate behaviors such as space shifting (i.e., ability to transfer content among devices) and fair use rights traditionally enjoyed for decades now. To tackle this issue, Sun Microsystems introduced Project DReaM (DRM everywhere available) [9], a project to create an open-source standard for interoperable DRM that relies on user authentication alone rather than devices. Project DReaM includes the DRM-OPERA architecture and makes it available in the form of an open-source community Java development project.

DRM-OPERA is an open DRM architecture [10] aiming at enabling the interoperability between different DRM systems. It has been specified and prototyped within project OPERA of the Eurescom organization. Among other activities, the OPERA project has produced an overview of state-of-the art DRM systems and standardization activities as of 2002 [11]. The DRM-OPERA architecture offers two interesting features that differentiate it from other solutions. First, it makes usage licenses independent of the underlying DRM system by offering its own license management. Then, usage licenses are bound to users instead of, as it is common with existing solutions, to devices.

While DRM future was discussed in silos across the industry be it consortiums like Coral [12] or standard initiative like DMP [13], there was no place where the whole community of all of the digital content stakeholders could come to discuss, define, and develop the future of digital content and DRM. To tackle this issue, Sun Microsystems decided in August 2005 to provide a virtual meeting place for all those contributing to this effort by creating the Open Media Commons [14], an open source community project, and a tool by sharing the internal project DReaM with the community under the OSI-approved Common Development and Distribution License (CDDL). One of the aims of the Open Media Commons community is to create an open environment where creators, content owners, consumers, network operators, technology providers and consumer electronics device manufacturers can work together to address the technical problems associated with DRM [15].

The Marlin Joint Development Association [16], is a consumer electronics industry technology development alliance formed by Intertrust Technologies, Matsushita Electric Industrial (Panasonic), Royal Philips Electronics, Samsung Electronics, and Sony Corporation that aims at creating a set of specifications for an open standard interoperable DRM platform for consumer electronics. In order to provide interoperability of content whatever distribution mode, DRM technology and standard are used, Marlin JDA specifications aim at providing a single technology toolkit to build DRM functions into their devices to support commonly used content distribution modes and thus avoid conflicts due to proprietary DRM technologies and standards. Marlin's authentication is user-based: it defines that user should be able to use content on any device they own and thus that content be tied to user identities and not device identities. While hiding issues such as content and device ownership that will need to be tackled, such a design is a step towards the copyright balance as defined previously. Marlin JDA is closely related to the Coral Consortium and as such, Marlin-based devices are able to interoperate with Coral-enabled DRM systems even if those systems do not use Marlin DRM components. It relies on Intertrust's NEMO [17] and Octopus technologies [18].

The Digital Media Project [19] is an independent standards initiative lead by Dr. Chiariglione, the founder of MPEG, aiming at tackling specific issues of DRM environment mainly related to the balance

between content owner and consumer rights. The DMP defines its mission as being to "promote continuing successful development, deployment and use of Digital Media that respect the rights of creators and rights holders to exploit their works, the wish of end users to fully enjoy the benefits of Digital Media and the interests of various value-chain players to provide products and services" [19]. The project standardizes appropriate protocols aiming at supporting the functions value-chain users need to execute and provides an Interoperable DRM Platform (IDP) specification [20] derived from MPEG-21 [21] standards and including an extended subset of MPEG-REL [22]. The IDP is based on requirements that have been derived from three sources, and which the platform has to be able to represent. The first one, Traditional Rights Usages (TRUs) covers usages exercised by media users and enjoyed in the pre digital era. The second one, Digital Enabled Usages (DEU), are usages either not possible or not considered in the analog domain. Finally the Digital Media Business Models (DMBM) is a set of TRUs and DEUs assembled to achieve a goal.

Other research works aim at proposing solutions to protect the copyright in a balanced way for copyright holders and users. The problem of managing exceptions is considered a hard problem and has been mainly explored in the context of fair use and rights expression languages. For instance, in [23], authors explore how rights management systems can be designed and implemented in a way that preserves the traditional copyright balance, especially with copyright's concern for the public domain and for the legitimate fair use. The authors are against leaving the determination of fair use in the rights holder's hands. Indeed they emphasize the fact that collective public interest may run contrary to the rights holder's individual interest and thus there may be a strong incentive for the rights holder to deny access. The authors doubt that system designers will be able to anticipate the range of access privileges that may be appropriate to be made of a particular work.

The analysis led in [24] suggests certain accommodations that DRM architectures, and especially their rights expression language components, should make to adequately express certain core principles of copyright law. Authors make two recommendations. The first recommendation proposes changes to the XrML REL vocabulary [25] to be able to highlight limitations on copyright exclusivity in cases such as fair use or first sale and rights transfer situations. The second one, goes toward the need for the creation of an Open Rights Messaging Layer. Indeed, their paper highlights current lack of rights messaging or transaction protocol that would provide standardized means for retrieving and disseminating rights information and policies, and issuing rights grants or permissions.

The details describing how these approaches relate to the model underlying the implementation presented in this paper are further discussed in [3]. In summary, it is legitimate to state that exception management in DRM systems remains an open question.

## 6. Conclusion and Future Work

This paper proposes a paradigm shift in information-centric security by expanding to the corporate sector work done on exception management in DRM environments. We argue that monitoring as an alternative to strong cryptography-based information security could provide increased efficiency while still preserving the much needed monitoring and tracking required in increasingly regulated environments where governance, Risk and Compliance issues are critical.

As a corollary, given such an approach, it would provide security policy professionals with a much needed feedback on security incidents and circumventions that are most often unnoticed today.

To this extent we argued for the need of an Enterprise Security Balance Principle whereby employees should be more trusted and given the flexibility to officially force security policies without having to unlawfully circumvent them based on their judgment. Since all actions are logged, security policy auditing and evolution becomes an added feature of the approach.

Further research and data is needed to validate our assumptions on security policy circumvention and the efficiency / usability issue. A prototype implementation of the approach in the context of a real Enterprise DRM system is a necessary step towards advancing our work.

Finally, recent evidence based on a study conducted in South Korea [27] suggests that among the major drivers of organizational adoption of Enterprise DRM, Compliance might not be the primary factor. While identified as being among them, it appears that Knowledge Management (KM) and Inter-organizational Structures (IOS) rank higher. In which case, following the adage “what can do the most can do the least”, if sound rights managed KM and IOS

embodies monitoring and audit trails compliance could be a "built-in" feature. Further study is needed to validate these propositions.

## References

[1] E. Felten, "DRM and Public Policy", in Communications of the ACM, V. 48, No. 7, July 2005, p. 112.

[2] J.-H. Morin and M. Pawlak, "Towards a Global Framework for Corporate and Enterprise Digital Policy Management", in Journal of Information System Security (JISSec), G. S. Dhillon (Ed.), 2006, Vol 2, No. 2, ISSN 1551-0123, pp. 15-24.

[3] J.-H. Morin, M. Pawlak, "A Model for Credential Based Exception Management in Digital Rights Management Systems", in proceedings of First International Conference on Global Defense and Business Continuity, ICGD&BC 2007, Second International Conference on Internet Monitoring and Protection, IEEE, July 1-6, 2007, Silicon Valley, USA.

[4] RSA Security, "The 2008 Insider Threat Survey", Oct. 2008.

[5] Cisco Systems Inc, "Data Leakage Worldwide: The effectiveness of Security Policies", White Paper, Aug. 2008.

[6] S.Farrell, R.Houslez, RFC3281, Internet Society, Apr 2002

[7] J.-H. Morin and M. Pawlak, "From Digital Rights Management to Enterprise Rights and Policy Management: Challenges and Opportunities", chapter 9 in Advances in Enterprise Information Technology Security, F. Herrmann and D. Khadraoui (Eds), Information Science Reference, IGI Global, July 2007, pp 169-188.

[8] INDICARE. The Informed Dialogue about Consumer Acceptability of DRM Solutions in Europe, from http://www.indicare.org/

[9] Fernando, G., Jacobs, T. and Swaminathan V., "Project DReaM An Architectural Overview", White Paper, Sun Microsystems, Sept. 2005.

[10] EURESCOM P1207 OPERA, An Open DRM Architecture. from http://www.eurescom.de/public/projectresults/P1200-series/P1207-D2.asp

[11] EURESCOM P1207 OPERA, Overview of state-of-the art DRM systems and standardization activities, from http://www.eurescom.de/public/projectresults/P1200-series/P1207-TI.asp

[12] CORAL Consortium Corporation, from http://www.coral-interop.org/

[13] DMP, Digital Media Project, from http://www.dmpf.org/

[14] OMC, Open Media Commons, from http://www.openmediacommons.org/

[15] Open Media Commons FAQ's, from http://www.openmediacommons.org/faqs.html

[16] Marlin JDA, CE and DRM Technology Leaders to Create a DRM Toolkit for Consumer Devices, from http://www.intertrust.com/main/news/2003_2005/050119_marlin.html

[17] Bradley, W.B. and Maher, D.P,. The NEMO P2P Service Orchestration Framework. In IEEE (Ed.), 37th Hawaii International Conference on System Science. IEEE.

[18] Intertrust, Octopus Principles of Operation. Internal Memo.

[19] DMP, Digital Media Project, from http://www.dmpf.org/

[20] DMP, Digital Media Project. Approved Document No 3. Technical Specification: Interoperable DRM Platform, from http://www.dmpf.org/open/dmp0653.zip

[21] MPEG-21 Multimedia Framework, from http://www.chiariglione.org/mpeg/standards/mpeg-21/mpeg-21.htm

[22] MPEG-REL, Multimedia framework (MPEG-21), Part 5: Rights Expression Language, from http://www.iso.ch/iso/en/CombinedQueryResult.CombinedQueryResult?queryString=21000-5

[23] D. Burk, J. Cohen, Fair Use Infrastructure for Copyright Management Systems, 11 Harv. J. Law & Tech., 2002.

[24] Mulligan, D., Burstein, A., and Erickson, J. "Supporting Limits on Copyright Exclusivity in a Rights Expression Language Standard. A requirements submission to the OASIS Rights Language Technical Committee.", Samuelson Law, Technology & Public Policy Clinic, and The Electronic Privacy Information Center, August 2002.

[25] XrML, eXtended rights Markup Language, from

[26] J.-H. Morin and M. Pawlak, "Exception-Aware Digital Rights Management Architecture Experimentation" in proceedings of 2008 International Conference on Information Security and Assurance (ISA 2008), IEEE, April 24-26, 2008, Busan, Korea, pp. 518-526.

[27] J.-H. Morin and A. Zeelim-Hovav, "Strategic Value and Drivers behind Organizational Adoption of Enterprise DRM : Setting the Stage", in proceedings of 7th Annual Security Conference, Las Vegas, NV, USA, June 2-3, 2008.

# Monitoring overlay path bandwidth using an inline measurement technique

Cao Le Thanh Man, Go Hasegawa and Masayuki Murata
Graduate School of Information Science and Technology, Osaka University
1-3 Yamadagaoka, Suita, Osaka 560-0871, Japan
{mlt-cao, hasegawa, murata}@ist.osaka-u.ac.jp

## Abstract

*We introduce ImSystem, a distributed system that infers real-time information concerning the available bandwidth of all paths in the overlay networks. The key concept in ImSystem is that, when the overlay hosts transmit overlay traffic, the overlay hosts deploy the traffic to perform inline network measurements. Inline network measurement is a method for measuring available bandwidth using only the packets transmitted in a data flow, instead of injecting probe traffic onto the network. ImSystem performs supplemental active measurement only when overlay traffic is insufficient for inline measurement, and therefore injects very little probe traffic onto the network. We also enhance the system to ImSystemPlus and ImSystemLight, deploying IP network topology information. In ImSystemPlus, the conflicts of the supplemental active measurements are greatly reduced, and, in ImSystemLight, the number of exchange messages sent by the overlay nodes is reduced. The simulation results show that the proposed systems can monitor the bandwidth in real-time while using only a small amount of probe traffic if the overlay traffic is sufficient.*

Keywords: End-to-end measurement, available bandwidth, inline measurement, overlay network, active measurement

## 1 Introduction

Overlay networks have been proposed as a way to improve Internet routing, due to quickly detecting and recovering from path outages and periods of degraded performance. Overlay networks are deployed on end-hosts running the overlay protocol software without the cooperation of the core of the network. The end-hosts (overlay nodes) are in charge of routing the overlay traffic. That is, they control the sequence of the overlay nodes that the traffic traverses before reaching its destination. Thus, the network end-hosts should collect network resource information in order to form an overall view of the entire network so as to optimize the path selection. Some metrics of IP network resources are propagation delay, packet loss ratio, capacity, and available bandwidth. When the overlay network obtains sufficient information, the path selection is good, and, in time, the performance of the overlay network can be greatly improved.

We focus on the task of monitoring an important metric of IP network resources: the end-to-end available bandwidth. For routing in the overlay network, the fluctuation of bandwidth should be reported in small time scales. Therefore, the measurement tasks should be performed periodically in short intervals. However, measuring the available bandwidth of $N^2$ paths of a network, where $N$ is the number of network nodes, requires a great deal of probe traffic. A number of studies [1-3] have focused on reducing the overhead. The methods proposed in these studies utilize the fact that the network paths are overlapping, with the assumption that the topology of the IP network is known. These methods carry out direct measurements on some overlay paths and indirectly estimate the bandwidth on the remained paths, deploying the measurement results of other network paths. However, the advantage of topology information appears to be limited because the amount of required probe traffic is still large, for example, on the order of $Nlog(N)$ [1, 2] or $N$ [3].

In a previous study [4] we have introduced a new version of TCP, called Inline measurement TCP (ImTCP). ImTCP can transmit data like previous TCP versions. However, ImTCP can also measure the available bandwidth of the path followed by TCP packets. When a sender transmits data packets, ImTCP first stores a group of up to several packets in a queue and then

subsequently forwards them at a transmission rate determined by the measurement algorithm. Each group of packets corresponds to a probe stream. Then, considering ACK packets as echoed packets, the ImTCP sender estimates available bandwidth of the network path between the TCP sender and TCP receiver. We name the technique inline measurement. The simulation results in [4] shows that ImTCP can yield measurement results with relative errors smaller than 20% every few RTTs without degrading transmission throughput. Moreover, studies in [5, 6] validates the measurement results for ImTCP in the real Internet environments.

The present paper is an extended version of our previous work in [7]. In this paper, we propose ImSystem, which infers the available bandwidth of all of the overlay network paths in real time. ImSystem utilizes the overlay traffic flows for measurement of the available bandwidth, using an inline measurement technique. When the transmission of overlay traffic occurs frequently, ImSystem works in a completely silent fashion, that is, ImSystem sends no probe traffic to the network. The system injects a small amount of probe traffic onto the network only when the overlay traffic is insufficient for obtaining up-to-date information by inline measurement.

We also propose enhanced versions of ImSystem, ImSystemPlus and ImSystemLight. Both two systems work under the assumption that the topology of the IP network is known. ImSystemPlus predicts the conflicts of the active measurements on the overlapping paths and delays some measurements in order to reduce the number of conflicts. On the other hand, in ImSystemLight, the overlay nodes estimate the bandwidth of a path using information concerning its overlapping paths. They then reduce the number of messages used to exchange measurement results of the paths.

The simulation results show that the proposed systems can provide up-to-date bandwidth information of overlay network paths while performing few additional active measurements. The proposed systems send almost no probe traffic when the amount of overlay traffic is sufficiently large.

The remainder of the present paper is organized as follows. In Section 2, we discuss some related research. Sections 3, 4 and 5 describe the respective designs of the three proposed systems: ImSystem, ImSystemPlus and ImSystemLight, respectively. In Session 6, we present a number of simulation studies to validate the proposed systems. Finally, Section 7 presents conclusions and a discussion of future research.

## 2 Related study

Resilient Overlay Networks (RON) proposed in [8] monitors the IP network by active measurements in fixed intervals. Every overlay host sends probe traffic for measurement of propagation delay and packet loss to all other hosts in the network. From the measurement results, the hosts estimate the throughput of data transmission on the overlay paths. The overhead for the information collection is $O(N^2)$, where $N$ is the number of overlay nodes. Therefore, [8] also points out that RON can work only with 50 or fewer nodes. In an effort to reduce the load of probe traffic on the network, [1] introduces a system that infers the available bandwidth of $N^2$ paths, in which the measurement overhead is reduced to the order of $Nlog(N)$. In this case, the accuracy becomes 90% of that when the measurements are performed on the full mesh. The method requires topology information, which is inferred by network tools such as `traceroute`. In addition, [2] deploys algebraic functions to reduce the measurement overhead to $O(Nlog(N))$. However, this requires a master node for managing all of the data processing. BRoute [3] leverages the fact that most Internet bottlenecks are on path edges as well as the fact that edges are shared by several different paths. BRoute performs bandwidth measurements on a number of paths using a hop-by-hop active measurement tool called Pathneck and infers the bandwidth of the remaining paths using the AS-level topology. In BRoute, the measurement overhead is further reduced to the order of $N$. This method also requires a master node with which to collect and process data from all hosts. The systems proposed in the present paper do not require a master node with which to manage the entire system while monitoring available bandwidth with a much smaller number of active measurements, compared to the existing methods proposed in [1-3, 8].

## 3 ImSystem

ImSystem is formed by software programs (called ImSystem programs) that are installed in overlay nodes. ImSystem is located between the overlay network and the IP network. ImSystem programs monitor the available bandwidth information of overlay paths and present this information to the overlay networks. ImSystem is independent of the overlay network; it can work with any overlay routing algorithms. Each ImSystem program collects the available bandwidth of the paths that start from the node where it locates and exchanges the measurement results with each other. The bandwidth information is yielded mainly by inline mea-

surement, the measurement technique that does not inject probe traffic onto the network. Active measurements are also used in case inline measurement results are not available.

### 3.1 Inline network measurement

Inline network measurement is the concept of performing active measurement using packets transmitted in a data flow. Inline network measurement can achieve high accuracy while not sending probe traffic into the network. We have developed ImTCP (Inline measurement TCP) [4], a Reno-based TCP that performs inline network measurement. Figure 1 illustrates the working of ImTCP. When ImTCP sender transmits data packets, it periodically stores a group of up to several packets in a queue and subsequently forwards them at determined intervals. Each group of packets corresponds to a probe stream. Then, the ImTCP sender checks the arrival intervals of the corresponding ACK packets. Under the supposition that all packets in a group traverse the same network path, if the arrival intervals are larger than the sending intervals, the available bandwidth of the network path between ImTCP sender and receiver is smaller than the transmission rate of the probe stream. Otherwise, the available bandwidth is larger. Using this fact, the ImTCP sender determines the available bandwidth by changing the transmission rate of probe streams. The measurement algorithm is similar to that of Pathload [9] and PathChirp [10]. However, ImTCP does not search for available bandwidth from 0 bps to the upper limit of the physical bandwidth with every measurement as these algorithms do. Instead, ImTCP limits the bandwidth measurement range using statistical information from previous measurement results so that one measurement is performed in as short as some RTTs. By limiting the measurement range, ImTCP avoids sending probe packets at an extremely high rate and keep the number of probe packets small so that it does-not affect other data flows. Our experiment results in [4] show that ImTCP can yield measurement results within 20.8% of the actual available bandwidth in intervals as short as some RTTs without degrading transmission throughput.

### 3.2 Filtering inline measurement results

We assume that ImTCP is deployed in all overlay hosts so that inline network measurement can be performed in every TCP connection used in the transmission of overlay traffic, and ImTCP senders pass all inline measurement results to the ImSystem program.

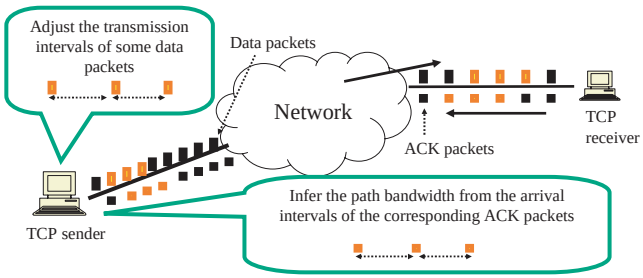



**Figure 1. Key concept of inline network measurement**

Each ImSystem program sends messages to exchange the measurement results with the ImSystem programs in other overlay hosts. The message includes the name of the beginning and end nodes of the path, the result of the measurement performed on that path and the validity term of the result. The validity term will be mentioned in the next Subsection. By exchanging the messages, every ImSystem program can obtain quickly the information of all paths in the overlay networks.

Inline measurement yields measurement results in small intervals such as a number of RTTs. Therefore, if the ImSystem programs exchange every result, the number of messages will be extremely large. In order to decrease the number of exchange messages, ImSystem programs send the messages to report the measurement results only when they detect a change in the results. However, the measurement results always fluctuate due to both the measurement errors and actual changes in the available bandwidth. The problem is how to determine which changes in the measurement results were caused by real available bandwidth changes. Here, we introduce Equation (1), as proposed in [11], for abrupt change detection.

$$g_k = (1-\alpha)g_{k-1} + \alpha(y_k - \mu)^2, g_0 = 0. \quad (1)$$

In Equation (1), $y_k$ is the current inline measurement result, $\mu$ is the mean of the $K$ latest results, where $K$ is the number of inline measurement results yielded since the last message was sent. The maximum value of $K$ is set to 15 in the following simulation experiments. In addition, $g_k$ is an indicator of an abrupt change at the current sample, and $\alpha$ is the forgetting parameter, taking a value between 0 and 1. We set $\alpha$ to 0.5 and use a simple threshold rule as follows. If $g_k$ is larger than the threshold ($h$), then we conclude that an actual change has occurred, otherwise the assumption is that no change occurred. Here, $h$ is set to 120. This value is sufficient to rule out all significant changes in

approximately 100-Mbps network paths.

### 3.3 Supplemental active measurement

In the case ImTCP is not available, ImSystem performs active measurements on the paths in every $T$ (s), where $T$ is the maximum length of the time that an active measurement may take. Even when ImTCP is deployed in the system, there are cases in which there is no overlay traffic on a certain path for a long time. During this period, ImTCP cannot perform inline measurements and the information concerning the available bandwidth of the path cannot be updated. In such cases, ImSystem waits a short time for new overlay traffic to arrive. The waiting time depends on how long the current measurement results can maintain their accuracy when the network environment changes with time. We refer to the time as the validity term of the current result. If there is no new overlay traffic during the validity term, ImSystem performs supplemental active measurements on that path in order to update its available bandwidth information.

We now consider the length of the validity term of an inline measurement result. The validity term corresponds to how long the measurement result can maintain its accuracy in the future environment. We consider the measurement results delivered in the past as a time series and predict the trend of the changes in the correct value of available bandwidth in the future. By doing this, we can calculate the period in which the current result remains valid.

Here, we apply a model introduced in [12]:

$$X_t = m_t + s_t + Y_t,$$

where $X_t$ is the time series of measurement results. In addition, $m_t$ is the part that shows the trend of the time series and is set to be a linear because the measurement intervals are short. In the case of inline measurements, the intervals are a number of RTTs and the term $s_t$ shows the periodical changes. In a short period, $s_t$ can be considered as a linear change. In addition, $Y_t$ is an independent and identically distributed random variable. $Y_t$ shows the random noise of the measurements. We assume that $Y_t$ has a normal distribution $N(0, \sigma^2)$.

We rewrite $X_t$ as follows:

$$X_t = a_0 \cdot t + b_0 + Y_t,$$

where $a_0$ and $b_0$ are fixed values that can be calculated using the integrated moving average method. Variance $\sigma$ is also calculated from the disparity in the trend and the measurement results.

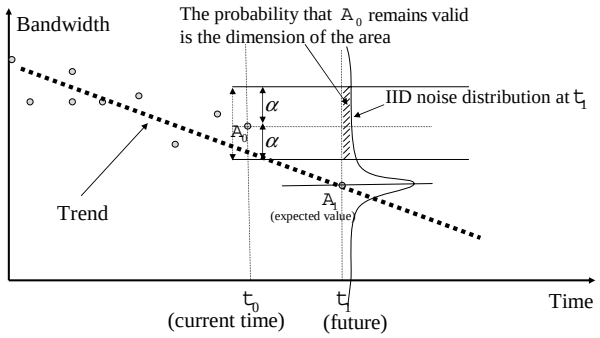



**Figure 2. The accuracy of the previous result in the future environment**

In Figure 2, we assume that at the time $t_0$, ImSystem sends messages to report the measurement result of $A_0$. Based on the measurement results just before $t_0$, we determine the trend of the changes in the available bandwidth of the path, as shown by the line in the figure.

We next consider the timing $t_1$ in the future. We examine the probability that the real available bandwidth remains at approximately $A_0$. This is the probability that the real available bandwidth appears in $[A_0 - \alpha, A_0 + \alpha]$, where $\alpha$ is $0.2A_0$, since study in [4] shows that the relative errors of ImTCP measurement results are within 20%. At this timing, the expected value of the measurement result, $A_1$, is:

$$A_1 = a_0 \cdot t_1 + b_0.$$

We assume that the measurement results at the time $t_1$ has the distribution $N(A_1, \sigma^2)$. Thus, the probability that the measurement result falls in $[A_0 - \alpha, A_0 + \alpha]$ is

$$q_{t_1} = \int_{A_0-\alpha}^{A_0+\alpha} \frac{1}{\sqrt{2\pi}\sigma} exp - \frac{(x - (a_0 \cdot t_1 + b_0))^2}{2\sigma^2} dx.$$

We assume that the measurement result $A_0$ becomes invalid at the time $t_1$ if the probability $q_{t_1}$ falls below 1%. The validity term is then calculated as $t_0 - t_1$ where $t_1$ is the smallest solution of the following inequality:

$$\int_{A_0-\alpha}^{A_0+\alpha} \frac{1}{\sqrt{2\pi}\sigma} exp - \frac{(x - (a_0 \cdot t_1 + b_0))^2}{2\sigma^2} dx \leq 0.01.$$

Thus, the validity term is long if the available bandwidth does not change significantly. That is, $a_0$ is approximately zero. Then, ImSystem can save active measurements. On the other hand, if the available bandwidth changes dramatically, ImSystem will perform active measurements just after inline measurement to quickly update the bandwidth information.

# 4 ImSystemPlus

In ImSystem and other previously proposed systems [1-3], there are the cases in which two or more overlapping paths are probed by active measurements at the same time. The common characteristic of the active measurement algorithms for available bandwidth is that, they require the probe traffic to fill up the unused bandwidth of the target path for some time. Therefore, in the case when the overlapping part of the paths includes a tight link (a link in which the unused bandwidth is smallest in the path), the probe packets of two different measurements may conflict to each others, causing degradation in measurement performance. In addition, the simultaneous transmission of probe traffic on the overlapping parts may cause localized congestion in the networks.

To avoid conflicts of measurements, ImSystemPlus program does not start active measurements right after the validity term of the current measurement result of the path expires. Instead, ImSystemPlus program considers whether or not the active measurements on other paths conflict with its measurement. In case there is high probability of conflict, the program delays its measurement for a certain time.

## 4.1 Conflict avoidance for measurements

In ImSystemPlus, the messages that the hosts exchange with each other have an additional field, showing the time when the validity term expires. If the term expires without any new overlay traffic transmission appearing on the correspondent path, ImSystemPlus program will perform active measurements to update the result. Therefore, ImSystemPlus program can know when the programs on other hosts schedule the performance of their active measurements. This time is referred to as the measurement time.

Figure 3 shows an example when the new measurement time of Path $b$ is too close (within $T$ (s)) to that of Paths $a$ and $c$. We assume that Paths $a$ and $b$, and $b$ and $c$ are overlapping, which means that the paths share one or more links. The active measurements on these paths will then come into conflict with each other.

To avoid probable conflicts on these paths, we propose a strategy that moves the measurement time of Path $b$ to the right side, far away from that of Paths $a$ and $c$. We consider the probability of moving the measurement time of Path $b$ $k \cdot T$ seconds to the right side, where $k$ = 0, 1, 2, ... To determine the probabilities, we consider the followings:

- Active measurements on Paths $a$ and $c$ may not be performed at the scheduled time due to the arrival of data transmission on these paths. If this probability is high, the probability of moving the measurement time of Path $b$ should be low.
- If the overlapping parts of the Paths $a$ and $b$, $c$ and $d$ are not so large, then the conflict of the measurement may not cause serious problems. In this case the probability of moving the measurement time of Path $b$ should be low. Next, we explain how to calculate these probabilities.

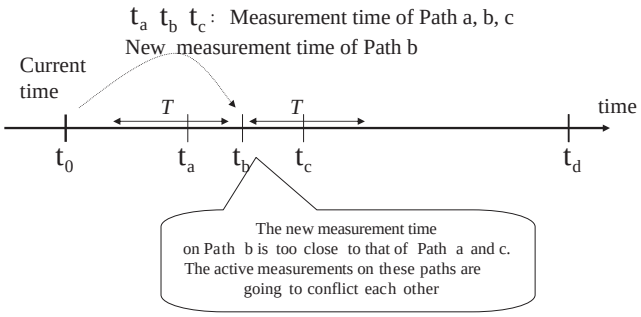



**Figure 3. Conflict in active measurements on overlapping paths**

## 4.2 Overlapping index

The degree to which two path overlaps each other is related to the effect that the simultaneous measurement on the two paths may have on the network. If the overlapping part is large, the conflict of the measurements will have a worse effect on the network and its performance. ImSystemPlus deploys the concept of path overlapping introduced in [13].

$$Joint(a,b) = \frac{Latency(G)}{min(Latency(a), Latency(b))}.$$

Here, $G$ is the overlapping part of Paths $a$ and $b$. $Latency()$ shows the transmission delay of the entire network path or part of the network path. $Joint()$ is an index taking a value between 0 and 1, which indicates the degree to which the two paths overlap each other.

## 4.3 Probability that a scheduled active measurement will be performed

We model the arrivals of data transmission on each overlay path as a Poisson process. The intervals between two arrivals on Path $x$ ($x$ is $a$, $b$, $c$ ... ) has the distribution of $E_x(\lambda_x)$, where $\lambda_x$ is calculated based on the transmission history of Path $x$.

Assume that the last measurement result of Path $x$ expires at $t_x$. An active measurement is scheduled to be performed at this time. However, during the period from the current time ($t_0$) to $t_x$, a data transmission may arrive. In this case, the active measurement scheduled at $t_x$ will not be performed. Due to the loss of the memory property of an exponential distribution, the probability that there is no data transmission during the period from $t_0$ to $t_x$ is: $P_x = e^{-\lambda_x \cdot (t_x - t_0)}$. This is also the probability that active measurement is performed at $t_x$.

### 4.4 Probability for moving measurement time

When the new measurement time $t_y$ of the measurement result on Path $y$ is decided, we examine other measurement times that are approximately $t_y$ in order to determine if there is any probable conflict measurements. We calculate the sum ($Q$) of the probability of the probable conflict measurements at approximately time $t_y$:

$$Q(t_y) = \begin{cases} S(t_y) & S(t_y) < 1 \\ 1 & S(t_y) \geq 1 \end{cases}$$

where

$$S(t_y) = \sum_{x; t_y - T < t_x < t_y + T} P_x \cdot joint(x, y). \qquad (2)$$

The probability that we do not move the measurement time $t_y$ to the right side is:

$$H^0 = 1 - Q(t_y)$$

Similarly, the probability that we set the measurement time of Path $y$ to $t_y + k \cdot T$ is:

$$H^{k \cdot T} = \prod_{h=0..k-1} Q(t_y + h \cdot T) \cdot (1 - Q(t_y + k \cdot T))$$

Here, $k = 1, 2...$ Note that when $k$ is sufficiently large, the part $P_x$ of $S(t_y + k \cdot T)$ calculated in Equation (2) approaches 0 (because when $t_x$ is sufficiently large, the probability that there is no data transmission in the period $[t_0, t_x]$ approaches 0). Then, $Q(t_y + kT) = 0$ and $H^{hT}$ with $h > k$ will be 0. This means that the measurement time cannot be delayed for a long time.

## 5 ImSystemLight

In ImSystem, in order to keep the bandwidth information up-to-date in each overlay node, each overlay node sends messages to all of the other overlay nodes to report any new measurement results. In this section, we show how to decrease the number of messages while maintaining the highest possible degree of accuracy with respect to the bandwidth information.

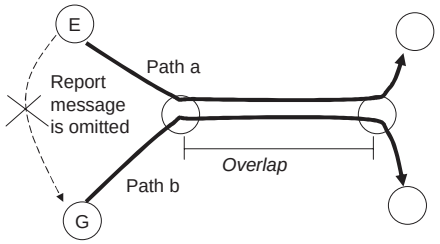



**Figure 4. Reducing report messages in ImSystemLight**

Like ImSystemPlus, ImSystemLight also deploys information about the topology of the underlying IP network to reduce the traffic caused by the communication between overlay nodes. Figure 4 is used to explain how ImSystemLight reduces the report messages. A node (node E) omits report messages relating to the inline measurement results on path $a$ that should be sent to node G if there is more than one path starting from G that overlaps path $a$. On the other hand, node G, which always updates its database when a report concerning the bandwidth of path $a$ arrives, will instead estimate the bandwidth of path $a$ automatically, in case the report is omitted by node E and the previous information becomes invalid. Among the paths starting from node G, let path $b$ be the path that has the longest overlap with path $a$. Under the assumption that path $a$ and path $b$ share the same tight link (the link with smallest available bandwidth), node G uses the available bandwidth of path $b$ as the estimated bandwidth of path $a$.

The probability that the tight link of path a appears on the overlapping part of path a and path b, is:

$$g(a, b) = \frac{Latency(Overlap(a, b))}{Latency(a)}.$$

Here, $Overlap(a, b)$ is the overlapping part of path $a$ and path $b$. In ImSystemLight, the probability that node E reports the measurement result of path a to node G ($p(E, a, G)$) is set as follows:

$$p(E, a, G) = 1 - g(a, b).$$

We use this setting because, if $g$ is low, the probability of incorrect estimation is high. In this case, node E should report the measurement results to avoid the

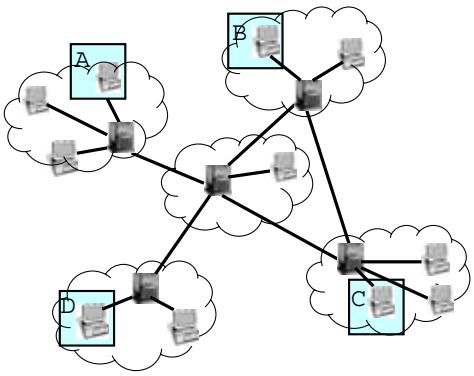



**Figure 5. Network topology for examine the work of ImSystem**

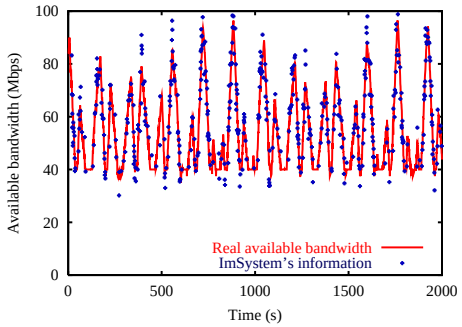



**Figure 6. Information about path D-B**

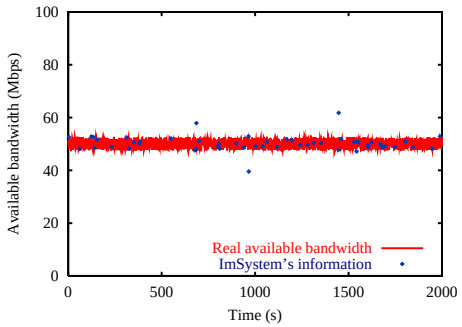



**Figure 7. Information about path B-C**

degradation in information accuracy. On the other hand, if $g$ is high, the estimation is reliable. In this case, node E can omit the message without causing a significant degradation in bandwidth information accuracy. Note that the messages will not be omitted if they report measurement results for active measurements. These results always have high accuracy, so that they are important for maintaining the accuracy of the information collected by the systems.

## 6 Simulation experiments

In this chapter, we evaluate the performance of the proposed systems in some different network topologies.

### 6.1 Collecting bandwidth information in ImSystem

We first examine the work of ImSystem in a simple topology shown in Figure 5. There is a four-node overlay network built upon an IP network. The capacity of the links in IP network is 100 Mbps. In addition to overlay flows, non-overlay traffic also exists on the IP link, referred to as cross-traffic. The rate of cross traffic at one link is uniformly distributed in $[M - 0.05M, M + 0.05M]$, where $M$ is the average rate, independent of the rate changes at other links. $M$ changes as follows. After every second, $M$ is increased by $b$ Mbps. When $M$ reaches 60 Mbps, it is decreased by $b$ Mbps every second, until reaching 0 Mbps. $M$ is then increased by $b$ Mbps every second, and so on. $b$ is randomly determined in the range $[1, 50]$ Mbps. For the links on the path between B and C, the average rate of cross traffic $M$ is kept constant at 50 Mbps. Overlay flows at the overlay paths are generated according to a Poisson process with an average arrival rate of $F$. All overlay paths have the same value of $F$. Overlay flow duration has exponential distribution with an average of 20 s. Overlay flow rate is uniformly distributed in the range [100 Kbps, 1 Mbps].

The active measurement is assumed to be Pathload. The time required by one active measurement is 10 s. Active measurement results are uniformly distributed in $[A - 0.1A, A + 0.1A]$, where $A$ is the real available bandwidth value. The active measurement rate is 250 Kbps.

The time required for one inline measurement is set to 1 s. In fact, ImTCP can yield results in smaller intervals. We assume that ImSystem takes an average of the measurement results every second. Inline measurement results are uniformly distributed in $[A+0.2A, A-0.2A]$, where $A$ is the real available bandwidth value. The relative error is calculated from the ImTCP simulation results in [4].

Figure 6 shows the changes of the real available bandwidth in the overlay path from host D to host B. In this case $F$ is set to 0.2. The figure also shows how the ImSystem program on the third host (host A) observes the bandwidth on this path. In this case, since the bandwidth changes dramatically over time, ImSystem updates the information frequently. Simi-

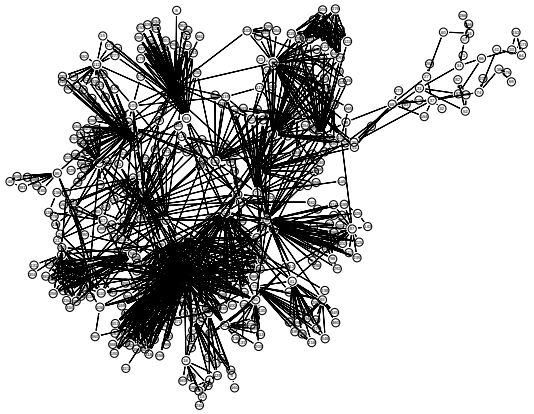


**Figure 8. Sprint network topology**

larly, Figure 7 shows how the ImSystem program on host A observes the available bandwidth of the path from B to C. The real value of the available bandwidth is also shown. Since the non-overlay traffic on the path is set at a constant 50 Mbps, the available bandwidth of the path does not fluctuate significantly. Therefore, in this case, we can see that ImSystem updates the bandwidth information in larger time intervals in order to decrease the number of messages sent into the network.

### 6.2 Simulation setting

We next evaluate ImSystem, ImSystemPlus, and ImSystemLight in larger network topologies. We deploy the following three topologies for IP networks in the simulation experiments.

- The topology of the Sprint network. We use the topology of the Sprint backbone network which is inferred by Rocketfuel [14]. The topology includes 467 nodes and 1280 links. Figure 8 illustrates the topology.
- Random topology. The network begins with an initial topology of three nodes. We then add new nodes to the initial topology and create links from the new nodes to the existing nodes. The probability that a new node has a link to node $i$ is

$$p_i = 0.01 + (1 - 0.01)^n \frac{1}{n}$$

  where $n$ is the number of existing nodes. This probability ensures that the new node is a connected node, that is, the new node has a link to at least one of the other node. The final topology has the same node number as the Sprint network and has 1282 links.
- BA model [15] topology. The network begins with an initial topology of three nodes. We then add new nodes to the initial topology until the number of nodes becomes the same as that of the Sprint network (467 nodes). The probability $p_i$ that the new node has a link to node $i$ is

$$p_i = \frac{\sum_j k_i}{k_j}$$

  where $k_i$ is the degree of node $i$. As the number of nodes reaches 467, the topology has 1388 links.

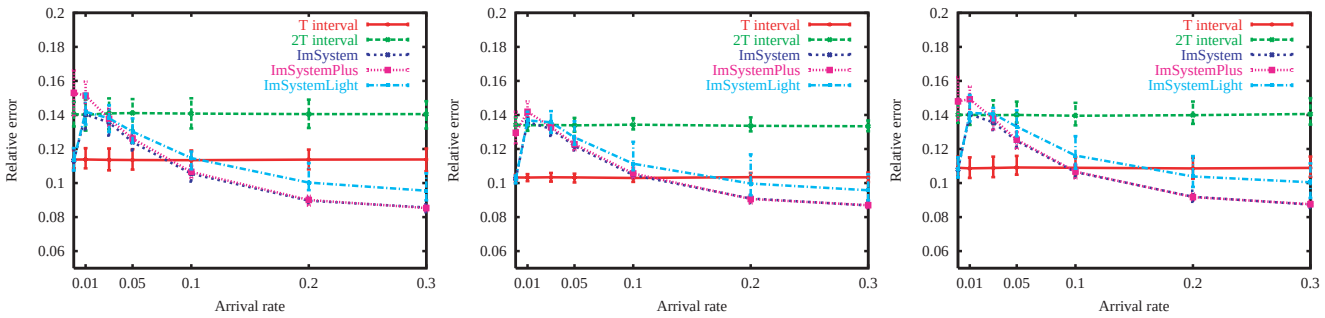
In the following simulations, the assumptions on cross traffic, overlay traffic and measurement tools are the same as those for the simulation mentioned in the previous subsection. The overlay network has 10 nodes, which are randomly distributed in the IP network. We perform 10 simulations with different distributions of overlay nodes. The time for each simulation is 2000 s.

For comparison, we also perform the simulations in which the active measurement results are periodically deployed in all overlay paths (full mesh) at fixed $T$ and $2T$ intervals, where $T$ is the maximum time for an active measurement to be performed. ($T$ is set to 15 (s)). In order to reduce the number of conflicts in the measurements, the nodes begin measurements at random times.

### 6.3 Accuracy of bandwidth information and the amount of probe traffic

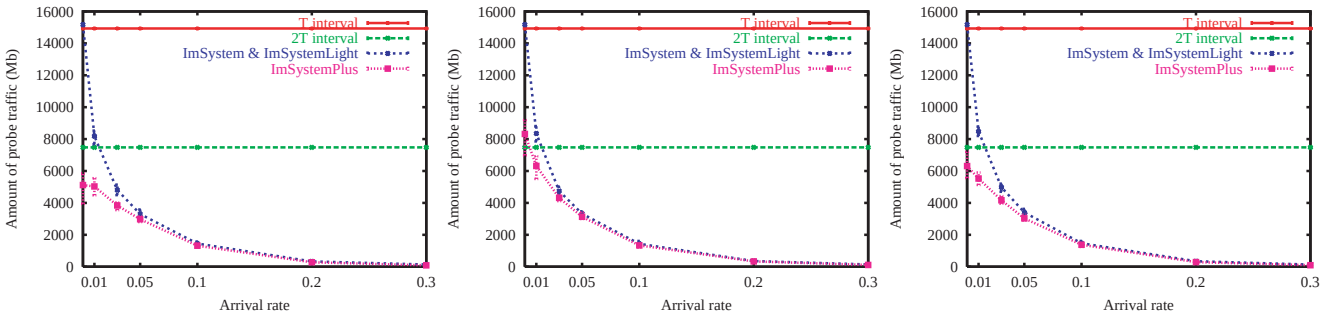
Figure 9 shows the average as well as the maximum, minimum value of the amount of probe traffic for active measurements performed by ImSystem, ImSystemPlus and ImSystemLight through 10 simulations in three different topologies. Also shown are the relative errors when the active measurement are performed in all overlay paths in every $T$ intervals, in the curve "$T$ interval", and $2T$ intervals, in the curve "$2T$ interval". The horizontal axes of these figures show values of $F$, the average arrival rates of the overlay traffic at the overlay nodes.

From Figure 9 we can see that, in all the topologies, the relative errors of the systems have the same trend. That is, in case there is no overlay traffic, ImSystem, ImSystemPlus as well as ImSystemLight has the same error as when active measurements are performed in every $T$ (s). The three proposed systems show their advantages when the arrival rate of overlay flow becomes higher than 0.1; they introduce error smaller than when the paths are actively measured in $T$ intervals. ImSystemPlus avoids the conflict of measurements so it sends to the network less probe traffic,




(a) Sprint network topology (b) Random topology (c) BA model topology

**Figure 9. Relative errors of collected bandwidth information**




(a) Sprint network topology (b) Random topology (c) BA model topology

**Figure 10. Probe traffic**

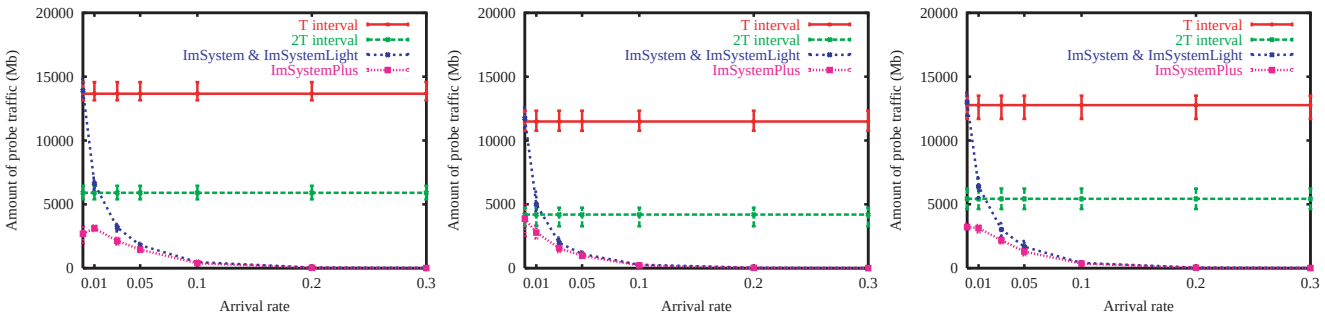
and therefore the error of the bandwidth information is a little larger than that of ImSystem. ImSystemLight also has a little larger error than the others because, as we will show later, it reduces the messages exchanged between overlay nodes.

Comparing Figure 9(b) with Figure 9(a) and Figure 9(c), we can see that in the Sprint network and BA model topologies, the error of bandwidth information collected by active measurement in $T$ or $2T$ intervals is higher than that in a random topology. That is because in the Sprint network and BA model topologies, many overlay network paths are sharing the same IP links, this leads to the fluctuation of available bandwidth due to the overlay traffic.

Figures 10(a), 10(b) and 10(c) show the amount of active probe traffic sent during a simulation. We can see that proposed systems always use much smaller active measurement than when active measurements are performed in $T$ or $2T$ intervals. When the arrival rate of overlay flow comes to 0.3, the proposed systems completely do not use the active measurements. ImSystemPlus uses less probe traffic than ImSystem and ImSystemLight in 2000 s of the simulation because it tends to delay the conflicting measurements.

### 6.4 Number of conflicting active measurements

We next examine the probe traffic that conflicts with other traffic in the present simulations and calculate the amount of probe traffic that shares one or more links with other probe traffic. The results are shown in Figure 11. In all three topologies, ImSystemPlus can eliminate most of the conflicting probe traffic that exists in ImSystem. The proportion is highest when there is no overlay traffic. This is due to the function of detecting and avoiding conflicts in the measurement of ImSystemPlus. The characteristics of the topologies have a slight effect on ImSystemPlus. In a random network, the overlapping index of the overlay paths is




(a) Sprint network topology (b) Random topology (c) BA model topology

**Figure 11. Conflicting probe traffic**

**Table 1. Number of the exchange messages**

| Topology | F | ImSystem | ImSys.Light | Ratio |
|---|---|---|---|---|
| Sprint network | 0 | 5435 | 5435 | 1.00 |
| | 0.01 | 4577 | 4206 | 0.92 |
| | 0.03 | 6184 | 5233 | 0.85 |
| | 0.05 | 7862 | 6462 | 0.82 |
| | 0.1 | 10982 | 8787 | 0.80 |
| | 0.2 | 13805 | 10963 | 0.79 |
| | 0.3 | 14740 | 11718 | 0.79 |
| Random | 0 | 5435 | 5435 | 1.00 |
| | 0.01 | 4557 | 4321 | 0.95 |
| | 0.03 | 6214 | 5535 | 0.89 |
| | 0.05 | 7928 | 6960 | 0.88 |
| | 0.1 | 10853 | 9382 | 0.86 |
| | 0.2 | 13504 | 11625 | 0.86 |
| | 0.3 | 14417 | 12409 | 0.86 |
| BA model | 0 | 5435 | 5435 | 1.00 |
| | 0.01 | 4733 | 4326 | 0.91 |
| | 0.03 | 6599 | 5501 | 0.83 |
| | 0.05 | 8400 | 6820 | 0.81 |
| | 0.1 | 11646 | 9182 | 0.79 |
| | 0.2 | 14463 | 11325 | 0.78 |
| | 0.3 | 15435 | 12106 | 0.78 |

small so that ImSystemPlus cannot reduce as many probe conflicts as in the Sprint network and BA model topologies.

### 6.5 Number of exchange messages

Table 1 shows the average number of messages that a node in ImSystem and ImSystemLight sends during the simulation. The last column shows the ratio between the ImSystem and ImSystemLight. ImSystem deploys inline measurement, the measurement accuracy of which is not as high as that of stand-alone measurement tools. Therefore, the nodes exchange many messages in order to increases the accuracy of the collected information. ImSystemLight, as expected, uses fewer messages in comparison with ImSystem while maintaining the highest possible accuracy. The topology also has little affect on the work of ImSystemLight. In the Sprint network and BA model topologies, the overlay paths overlap significantly so that ImSystemLight can reduce the number of exchange messages to a greater degree than in a random network.

## 7 Conclusion

In the present paper, we proposed ImSystem, which monitors the available bandwidth of all end-to-end paths in an overlay network in real time. The proposed system is based primarily on inline network measurement, that is, ImSystem deploys active overlay data flow for measurement. Therefore, the system injects little probe traffic onto the network while inferring the available bandwidth in a real-time fashion. We also proposed ImSystemPlus and ImSystemLight. In these systems, conflicts in measurement traffic and the data exchanged between overlay nodes are reduced.

In future works, we will examine how scalable are the proposed systems. We will also implement and evaluate their performance in real network environments.

## References

[1] C. Tang and P. McKinley, "On the cost-quality tradeoff in topology-aware overlay path probing," in *Proceedings of the 11th ICNP*, Nov. 2003.

[2] Y. Chen, D. Bindel, H. Song, and R. Katz, "An algebraic approach to practical and scalable overlay network monitoring," in *Proceedings of ACM SIGCOMM 2004*, Aug. 2004.

[3] N. Hu and P. Steenkiste, "Exploiting internet route sharing for large scale available bandwidth estimation," in *Proceedings of IMC'05*, Oct. 2005.

[4] C. L. T. Man, G. Hasegawa, and M. Murata, "ImTCP: TCP with an inline measurement mechanism for available bandwidth," *Computer Communications*, vol. 29, no. 10, pp. 1614–2479, 2006.

[5] T. Tsugawa, G. Hasegawa, and M. Murata, "Background TCP data transfer with inline network measurement," *IEICE Transactions on Communications*, vol. E89-B, pp. 2152–2160, Aug. 2006.

[6] T. Tsugawa, C. L. T. Man, G. Hasegawa, and M. Murata, "Inline bandwidth measurements: Implementation difficulties and their solutions," in *Proceedings of E2EMON 2007*, May 2007.

[7] C. L. T. Man, G. Hasegawa, and M. Murata, "Inferring available bandwidth of overlay network paths based on inline network measurement," in *Proceedings of ICIMP 2007*, July 2007.

[8] D. Andersen, H. Balakrishnan, M. Kaashoek, and R. Morris, "Resilient overlay networks," in *Proceedings of SOSP 2001*, Oct. 2001.

[9] M. Jain and C. Dovrolis, "End-to-end available bandwidth: Measurement methodology, dynamics, and relation with TCP throughput," in *Proceedings of ACM SIGCOMM 2002*, Aug. 2002.

[10] V. Ribeiro, R. Riedi, R. Baraniuk, J. Navratil, and L. Cottrell, "PathChirp: Efficient available bandwidth estimation for network paths," in *Proceedings of PAM 2003*, Apr. 2003.

[11] M. Basseville and I. V. Nikiforov, *Detection of Abrupt Changes: Theory and Application*. Prentice-Hall, Inc., 1993.

[12] P. J. Borockwell and R. A. Davis, *Introduction to time series and forecasting*. Springer-Verlag NewYork, Inc., 1996.

[13] M. Zhang and J. Lai, "A transport layer approach for improving end-to-end performance and robustness using redundant paths," in *Proceedings of the USENIX 2004 Annual Technical Conference*, June 2004.

[14] N. Spring, R. Mahajan, and D. Wetherall, "Measuring ISP topologies with Rocketfuel," in *Proceedings of SIGCOMM 2002*, Aug. 2002.

[15] A. Barabasi and R. Albert, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, pp. 47–97, 2002.


# Preliminary 2009 Conference Schedule

**http://www.iaria.org/conferences.html**

**NetWare 2009**: June 14-19, 2009 - Athens, Greece

- SENSORCOMM 2009, The Third International Conference on Sensor Technologies and Applications
- SECURWARE 2009, The Third International Conference on Emerging Security Information, Systems and Technologies
- MESH 2009, The Second International Conference on Advances in Mesh Networks
- AFIN 2009, The First International Conference on Advances in Future Internet
- DEPEND 2009, The Second International Conference on Dependability

**NexComm 2009:** July 19-24, 2009 - Colmar, France

- CTRQ 2009, The Second International Conference on Communication Theory, Reliability, and Quality of Service
- ICDT 2009, The Fourth International Conference on Digital Telecommunications
- SPACOMM 2009, The First International Conference on Advances in Satellite and Space Communications
- MMEDIA 2009, The First International Conferences on Advances in Multimedia

**InfoWare 2009:** August 25-31, 2009 – Cannes, French Riviera, France

- ICCGI 2009, The Fourth International Multi-Conference on Computing in the Global Information Technology
- ICWMC 2009, The Fifth International Conference on Wireless and Mobile Communications
- INTERNET 2009, The First International Conference on Evolving Internet

**SoftNet 2009**: September 20-25, 2009 - Porto, Portugal

- ICSEA 2009, The Fourth International Conference on Software Engineering Advances
  - SEDES 2009: Simpósio para Estudantes de Doutoramento em Engenharia de Software
- ICSNC 2009, The Fourth International Conference on Systems and Networks Communications
- CENTRIC 2009, The Second International Conference on Advances in Human-oriented and Personalized Mechanisms, Technologies, and Services
- VALID 2009, The First International Conference on Advances in System Testing and Validation Lifecycle
- SIMUL 2009, The First International Conference on Advances in System Simulation

**NexTech 2009**: October 11-16, 2009 - Sliema, Malta

- UBICOMM 2009, The Third International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies
- ADVCOMP 2009, The Third International Conference on Advanced Engineering Computing and Applications in Sciences
- CENICS 2009, The Second International Conference on Advances in Circuits, Electronics and Micro-electronics
- AP2PS 2009, The First International Conference on Advances in P2P Systems
- EMERGING 2009, The First International Conference on Emerging Network Intelligence
- SEMAPRO 2009, The Third International Conference on Advances in Semantic Processing