Not Quickly, Just in Time: Improving the Timeliness and Reliability of Control Traffic in Utility Networks

Wagner Saback Dantas, Alysson Neves Bessani, Miguel Correia
Universidade de Lisboa, Faculdade de Ciências, LaSIGE – Portugal
wagners@lasige.di.fc.ul.pt,{bessani,mpc}@di.fc.ul.pt

Abstract

One important aspect of modern critical infrastructures (CI) like power grids is that the control of the physical processes (e.g., electricity transmission) is no longer done locally but in a distributed way with SCADA/PCS systems. This distributed control is done over utility networks, typically wide-area IP networks, that interconnect the CI facilities (e.g., transformation substations). This communication is critical because, if certain commands are not executed within certain time bounds, events with a severe socio-economic impact may happen (e.g., blackouts). However, utility networks often do not ensure the required timeliness and reliability, especially when accidental or malicious faults occur. This paper presents the Calm-Paranoid algorithm (CP), a novel overlay and multihoming routing strategy aimed to achieve adequate levels of timeliness and reliability in utility networks, even under harsh fault scenarios.

1. Introduction

Critical infrastructures (CI) like power grids have been greatly evolving for several years. One aspect of this evolution is that the control of the physical process (e.g., electricity generation and transmission) is no longer done locally at the different facilities of the CI company (e.g., transformation substations), but in a distributed way with systems and applications often called SCADA/PCS[1]. This distributed control is done over *utility networks* that interconnect the CI facilities, which are typically wide-area IP networks [9], [12].

Control applications exchange messages over the utility network following certain standard routines (e.g., for power grid operation, see [8]). Control traffic often takes commands that have to be executed within specific application-defined deadlines. This communication is critical because, if certain commands are not executed within certain time bounds, events with a severe socio-economic impact may happen (e.g., the destruction of generators or other components leading to long-duration blackouts). Applications however can tolerate a certain level of message losses (e.g., for periodic monitoring data).

Utility networks are composed by *well-provisioned channels*, i.e., high-bandwidth IP channels with a good level of redundancy, contracted to one or more *IP network service providers* (ISP). Some of these channels share network resources (e.g., routers and links) with other kinds of traffic, including external traffic from the Internet. CI facilities often use *multihoming* [1], i.e., are connected to at least two distinct ISPs.

Just like the Internet, utility networks may experience periods of unavailability due to the failure of routers, their ports, physical links, etc. These problems are far from uncommon [13] and unavailability may become even more frequent due to *denial-of-service attacks* (DoS), e.g., executed as cyber-war or cyber-terrorism acts [17]. These faults can be tolerated using retransmissions, but from the standpoint of control applications they may cause the time bounds for the execution of commands to be violated. These violations may lead to CI failures and the above-mentioned severe socio-economic impacts.

Overlay networks have been used as mechanisms to implement routing schemes that take into account specific application requirements. However, most of these schemes do not have the objective of providing timeliness guarantees (e.g., [3]). Others have the objective of improving the end-to-end communication latency, but not of attaining application-defined maximum delays (e.g., [2], [15]).

This paper addresses the problem of providing *timeliness and reliability assurances for control traffic in wide-area utility networks*, tackling the requirements of modern critical infrastructures. Our solution is based on a novel overlay/multihoming routing strategy, the Calm-Paranoid (CP) algorithm.

We assume that the utility network provides only a best-effort service, no latency and bandwidth guarantees. Theoretically it is possible to have these guarantees, e.g., using ATM or DiffServ/RSVP. However, ISPs typically do not provide a service with these guarantees, especially in wide-area networks. Thus, we propose a solution that does not require such guarantees from the network, only plain Internet-like IP network service.

The paper presents a preliminary comparison of CP with

1. Supervisory Control and Data Acquisition/Process Control Systems

other overlay/multihoming routing strategies [3], [4], [15] based on simulations. CP is also compared with a baseline scheme that floods the overlay network with the messages that have to be delivered in time. Simulations are based on a model of a realistic wide-area utility network with the WAN under the effect of accidental and malicious faults (DoS).

The simulations have the objective of answering two fundamental questions: *(a)* does CP provide better timeliness and reliability guarantees to control traffic than previous strategies? *(b)* Does it achieve that improvement at a reasonable cost in terms of messages sent, when compared with other strategies? The evaluation shows that CP indeed provides much better timeliness and reliability guarantees than other schemes in the literature, reaching the same results as flooding at a much lower cost.

2. Utility Network Properties

We consider a CI composed of a set of facilities scattered over a region (e.g., a country or a state) that communicate using an utility network with the following properties:

Property 1: Static territory-limited wide-area network. The utility network covers a geographically delimited zone and its organization tends to be static, i.e., the arrangement of local networks and the links between pairs of these local networks do not change often.

Property 2: Connectivity to the wide-area network via ISP multihoming. Facilities (i.e., local networks) are connected to the wide-area networks provided by two or more ISPs. This provides a basic level of redundancy and fault tolerance.

Property 3: Wide-area network provides timely communication in fault-free periods. In the normal case, without failures or network congestion, the utility network provides timely communication.

Property 4: Wide-area network failures affect the communication timeliness. Failures in network components like routers and links can affect the behavior of the utility network, impairing the timeliness required for control traffic.

3. Calm-Paranoid Algorithm

The design rationale of the Calm-Paranoid Algorithm (CP) is fundamentally driven by the CRUTIAL Reference Architecture [16], conceived to enhance dependability properties of CIs. This architecture models CIs as a WAN-of-LANs (Figure 1): a set of LANs (CI facilities) interconnected by a wide-area network (WAN).

Each LAN is connected to the WAN through a special gateway, the *CRUTIAL Information Switch* (CIS). The CIS provides two services: *(a)* the protection service, which we disregard in this paper (see, e.g., [5] for details); and *(b)* the communication service, which is the focus of the paper. We assume that the communication delays on the LANs are negligible so we focus on attaining the application-defined

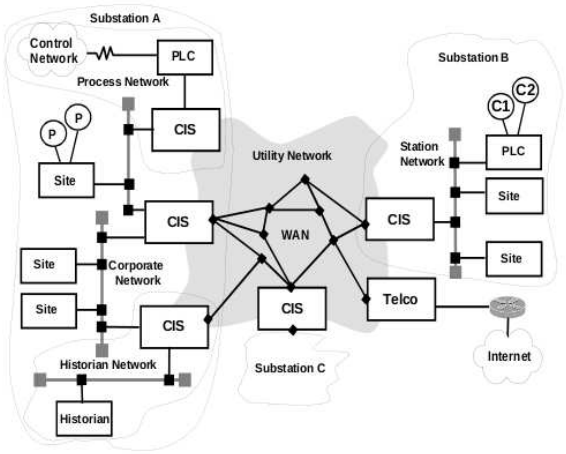

Figure 1. WAN-of-LANs Model [16].

delays in the WAN. CP is chiefly designed to enforce the required timeliness and reliability for control traffic. Henceforth, we designate each CIS a *CP node* to emphasize that here we are only interested in the communication service as primarily implemented by the CP algorithm.

Design rationale. CP nodes define an *overlay network* atop the IP network and run the CP algorithm to select overlay paths that are (expectedly) providing timely communication. CP nodes are connected to the WAN via *multihoming* [1], i.e., through connections provided by distinct ISPs. Therefore, messages can be transmitted via a variety of overlay paths. For multihoming to provide fault tolerance effectively, the networks of the ISPs have to share the minimum amount of resources, something that is not granted but can be assessed (e.g., [6]).

CP is a *probing-based one-hop source routing* scheme (like RON [3]). The overlay route of each message is defined at the sender (source routing), based on probing data and composed of at most one intermediary relaying CP node (one-hop). Routes are selected according to a metric of latency between pairs of CP nodes, the round-trip time (RTT). Therefore, each *logical channel* of the overlay interconnecting two CP nodes can be a direct IP channel between the nodes or a one-hop indirect channel (in which another CP node works as a relay).

CP is based on the idea of using *judicious spatial redundancy*. Messages are sent through one base channel, which we call the *calm channel*, plus one backup channel, called the *paranoid channel*. These channels are selected in a way that maximizes the number of possible retransmissions of the message without impairing its timely delivery. Therefore, for each message, CP starts by using a calm channel that does not offer the best RTT but permits messages to arrive in time, leaving the best RTT channels

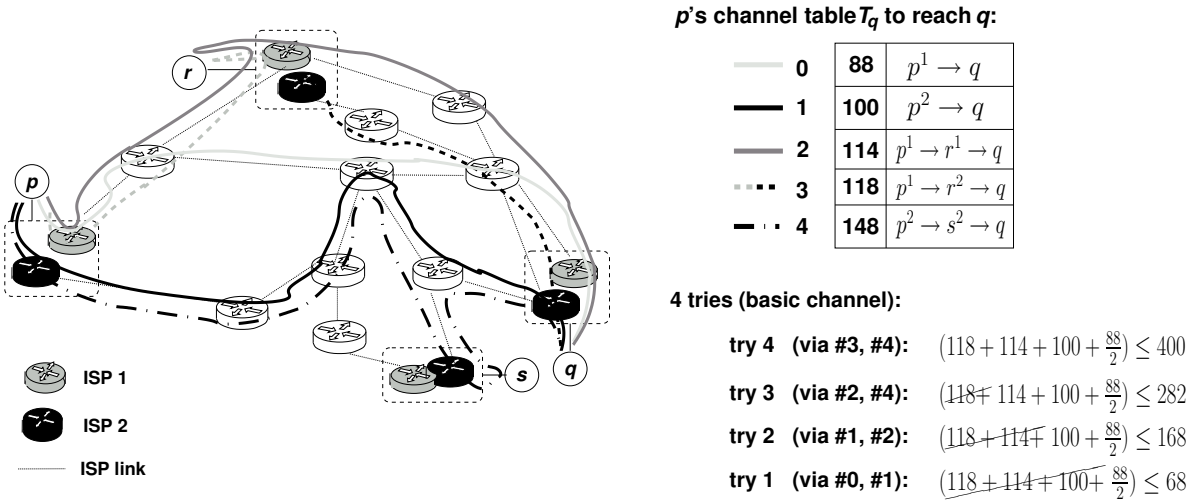

Figure 2. Example: p to q, initial deadline = 400ms.

to be used in retransmissions, when the time for delivering the message is shorter. The paranoid channel is selected in such a way that it has as little correlation as possible with the calm channel (in terms of links and routers used), while still able to deliver the message attending time constraints. As a result, the messages are not sent quickly (i.e., the algorithm does not try to improve latency), but in a timely way.

The algorithm. Each CP node has a set of tables describing (logical) channels that can be used to send messages to other CP nodes (one table per destination node). When a node p has to send a message m to q, it defines the route for m (i.e., the channel used for its transmission) based on the content of its table T_q. An entry i of table T_q ($T_q[i], 0 \leq i < |T_q|$) stores overlay/direct channel information such as $T_q[i].rtt$ (RTT estimate), $T_q[i].c$ (overlay itinerary) and $T_q[i].faulty$ (failure status). Table entries are ordered by the RTT field, i.e., given two entries i and j ($i \neq j$), if $T_q[i].rtt < T_q[j].rtt$, then $i < j$.

When node p is requested to send a message m to q with a given deadline $m.d$, it executes the following algorithm:

1) p sets t to the number of times it can try to send m ($0 < t \leq |T_q|$) using channels that *(a)* are not faulty and *(b)* allow the reception of m before its deadline $m.d$ (according to the RTT estimates in T_q);
2) p selects a calm channel $c = T_q[t-1].c$ such that it is possible to send m via faster channels (i.e., channel $T_q[j].c$ with $j < t-1$) if c fails and m still arrive within $m.d$;
3) p selects one paranoid channel c' as less correlated to c as possible, that also permits m to be delivered before $m.d$ expires;
4) p sends m through c and c' and sets a timeout of $T_q[t-1].rtt$ ms;
5) If an ACK is received before the timeout expires, the algorithm ends, otherwise p sets $m.d = m.d - T_q[t-1].rtt$ and goes to step 1.

Figure 2 illustrates how CP works. Consider a set of CP nodes $\{p, q, r, s\}$, each one in charge of traffic generated by control applications in distinct LANs. The LANs are interconnected by an utility network provided by two different ISPs (ISP 1 in grey and ISP 2 in black) and the channels table T_q of p contains 5 entries. We use the notation o^x to mean that the CP node $o \in \{p, r, s\}$ in the itinerary of a channel $T_q[i].c$ sends data via ISP $x \in \{1, 2\}$.

In the figure, CP node p is requested to send a message m with an initial deadline $m.d = 400$ ms to another CP node q. The figure shows a complete sequence of 4 possible tries (with the respective calculations and channels used), from the first transmission with $m.d = 400$ ms (in which channel #3 was selected as calm) to the last one with $m.d = 68$ ms (in which channel 0 – the best one – was selected as calm). Notice that, in each try, the selected paranoid channel had low correlation with the calm channel employed on the try. This is obtained by, whenever possible, using distinct ISP access links for both calm and paranoid channels.

4. Preliminary Evaluation

The evaluation was done using simulations on a realistic setting. We were forced to use simulations instead of real experiments by the practical impossibility of running experiments in a real utility network. However, simulations also allowed us to test many scenarios and faultloads, which would be very difficult to test in a real network.

The simulated CI is inspired in the topology of a real Italian ISP with 31 routers and 51 IP paths [14]. We assume a second ISP with same topology, but using different routers and links. All links have latency of 50 ms and bandwidth of 1

Strategy designation	Brief description
Best-Path (BP)	Overlay (RON [3]). Send via overlay channel with best RTT estimate. If failure, retransmit via the same channel at most 3 times with timeout of 3s.
Calm-Paranoid (CP)	Our strategy.
Flooding (F)	Overlay. Ultimate approach that sends the message through all 26 overlay channels available.
Multi-Path (MP)	Overlay (Mesh-routing [15], implemented as [4]). Send via 2 overlay channels: the direct path and a randomly chosen overlay channel (direct or not).
Primary-Backup (PB)	Non-overlay (used currently in most power grids [11]). Send always via one access link until it fails. If failure and there are redundant links, pick another one. Retransmit at most 3 times with timeout of 3s.

Table 1. The communication strategies evaluated.

Gbps. For each strategy (described in Table 1) and faultload (accidental or malicious) the time window simulated had 5 hours, with 90,134 application messages with deadlines of 1 to 4 seconds.

We compare CP with some previous overlay routing and multihoming strategies (Table 1) in order to answer two questions: *(a) given a set of messages with different deadlines, to what extent are they received in time with each strategy? (b) What are the transmission costs associated with CP in relation to other strategies?* Both questions refer to the feasibility of using CP to support the timeliness and reliability requirements of CI control communication over wide-area utility networks, namely when the network is under the effect of accidental failures and DoS attacks.

We analyzed the algorithms in several scenarios, with many faultloads, but here we present only two for space reasons: *(i)* accidental faults (only accidental faults with a certain pattern); *(ii)* attack campaign (accidental plus malicious faults). For each faultload, we investigate the costs and benefits using two metrics: *number of deadlines missed*, which gives the ability to provide timeliness (benefit); *communication overhead*, i.e., the amount of extra messages sent (cost). More details of our setup can be seen in [10].

The baseline cost-benefit for all strategies when the WAN is free of failures is summarized in Table 2. The results for all strategies when the WAN is under some faultload are presented in Figures 3, 4, 5 and 6. The y-axis of the fours graphs represent a normalized result of cost (resp. benefit) for a given strategy under some faultload with respect to the most costly (resp. worst benefit) solution.

Strategy	Cost (extra messages)	Benefit (deadlines missed)
F	2,172,314	0
CP	90,150	
BP	0	
MP	90,134	
PB	0	

Table 2. Baseline cost-benefit (fault-free case).

Accidental faults. Figures 3 and 4 show respectively the costs and benefits for the accidental faults scenario. We injected 74 faults on the simulation, following the model defined in [7], [13]. Only 30% of the faults lasted more than 30s and the fault pattern introduced was the same for all strategies.

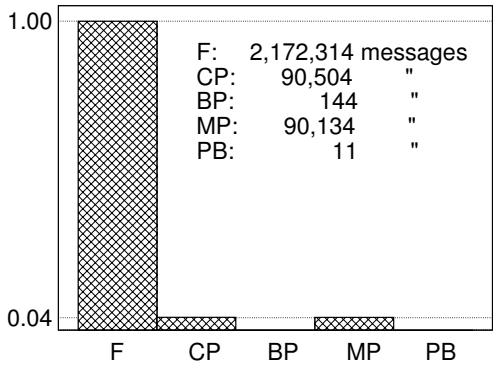

Figure 3. Cost of communication strategies with *accidental faults*.

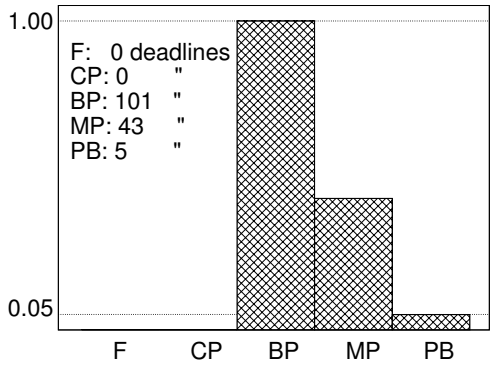

Figure 4. Benefits of communication strategies with *accidental faults*.

The CP strategy has a larger cost (number of extra messages sent) than all other strategies except flooding (Figure 3). However, CP also misses less deadlines than its counterparts at much lower cost than flooding: 90,504 extra messages versus more than 2 million. More precisely, CP managed to mask *all* failures that prevented deadlines

from being accomplished by the other strategies, just like flooding, but sending only 4% of its messages.

It is worth to notice that the primary-backup (PB) strategy, which is commonly used in many critical infrastructures, missed very few deadlines, which suggests that it is a good solution for dealing with accidental failures in backbones.

Malicious faults. Figures 5 and 6 respectively show the costs and benefits of using the communication strategies of Table 1 with a more severe faultload. Again, we injected 74 faults, following the same model as in the previous faultload. However, to represent the effect of DDoS attacks against the utility network, 80% of the faults had more than 30s. The fault pattern introduced was again the same for all strategies. As expected, more deadlines were missed than in the accidental faults scenario for every single strategy.

A few deadlines were missed with flooding (5 deadlines out of 90,134). The reason was that the faultload in some periods was so harsh that it was impossible for the message to arrive to its destination, even flooding the network (i.e., there was no path available). With CP, 37 deadlines out of 90,134 were missed. These were about 0.5% of all missed with MP that was the strategy with worst benefit, and 2.4% with BP, which was the best benefit strategy other than CP and F. Notice that in this scenario the primary-backup (PB) strategy missed 1,535 deadlines, which shows that it is not robust enough to maintain timely communication with severe faultloads.

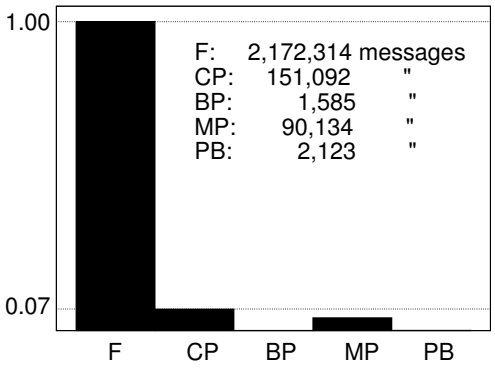

Figure 5. Cost of communication strategies with *malicious faults*.

As in the accidental case, CP introduced more overhead than all other strategies, except flooding. But, again, we could observe a good cost-benefit favoring CP in relation to its counterparts. When the faultload is made more harsh to simulate the effect of DoS attacks, CP provided much better results than the other strategies. CP experienced less deadlines missed and, at least, the same incremental cost in order of magnitude. Interestingly, even exploring spatial redundancy as CP does, MP could not exhibit the same results that CP did. The basic difference comes from the

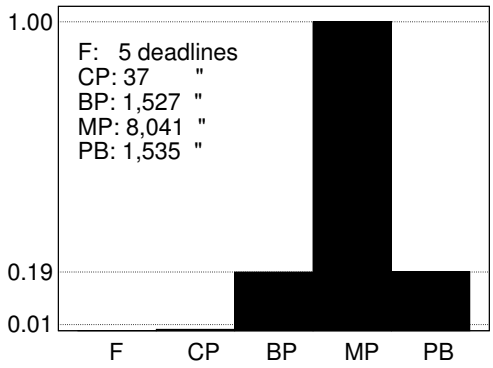

Figure 6. Benefits of communication strategies with *malicious faults*.

overlay channel selection strategy employed by each strategy: MP sends the message through the direct channel and a randomly selected backup overlay channel [4], [15], while our algorithm carefully chooses its channels to maximize the number of retransmissions that can be done.

5. Conclusion and Future Work

This paper presents the Calm-Paranoid algorithm (CP), a novel overlay selection path strategy to enhance timely and reliable communication over wide-area utility networks. The paper briefly presents the design rationale of the CP and how this algorithm works. It also shows an initial cost-benefit evaluation of CP.

The results give us an initial idea of the main benefits of CP in relation to other approaches (overlay-based or not). Overall, they show that CP can improve the timeliness and reliability of SCADA/PCS WAN communication, with a benefit that is not far from network flooding. The results also show that CP leads to much lower costs than flooding, which proves to be an extreme and cumbersome solution. CP provided a reasonable incremental overhead, equivalent to a solution applied in practice in utility networks, the PB strategy. Even though using the backup (paranoid) channel implied a higher absolute cost, its use was justifiable: it could keep timely communication up, helping to circumvent failures that caught other strategies as the WAN communication environment gets harsher.

This work can take several directions. An issue is to study mechanisms to improve the selection of paranoid channels for reducing the costs pointed here and increasing more application benefits as well. It includes both analyzing ways of using multiple paranoids with new helper strategies and tuning the algorithm to adaptively choose paranoids over those different strategies. For evaluation, we plan to improve the simulations by, for example, refining our characterization of rogue events and to do experiments in a real network, that has to be the Internet due to the impossibility of using an utility network.

Acknowledgments

This work was supported by the EC through project IST-4-027513-STP (CRUTIAL) and Alban scholarship E07D401192BR, and by the FCT through the Multiannual and the CMU-Portugal Programmes. We cordially thank Fabrizio Garrone and other partners of the CRUTIAL Project (`http://crutial.cesiricerca.it`) for the discussions and detailed information on the Italian power grid, which helped us to understand its functioning and construct our simulation testbed. We also thank Nuno Ferreira Neves for the valuable feedback about this work.

References

[1] A. Akella, B. Maggs, S. Seshan, A. Shaikh, and R. Sitaraman. A measurement-based analysis of multihoming. In *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '03)*, pages 353–364, August 2003.

[2] Y. Amir, C. Danilov, S. Goose, D. Hedqvist, and A. Terzis. An overlay architecture for high quality VoIP streams. *IEEE Transactions on Multimedia*, 8(6):1250–1262, December 2006.

[3] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris. Resilient overlay networks. In *Proceedings of the 18th ACM Symposium on Operating Systems Principles (SOSP'01)*, pages 131–145, October 2001.

[4] D. G. Andersen, A. C. Snoeren, and H. Balakrishnan. Best-path vs. multi-path overlay routing. In *Proceedings of the 3rd ACM SIGCOMM Conference on Internet Measurement (IMC '03)*, pages 91–100, October 2003.

[5] A. Bessani, P. Sousa, M. Correia, N. F. Neves, and P. Veríssimo. The CRUTIAL way of critical infrastructure protection. *IEEE Security & Privacy*, 6(6):44–51, 2008.

[6] W. Cui, I. Stoica., and R. H. Katz. Backup path allocation based on a correlated link failure probability model in overlay networks. In *Proceedings of the 10th IEEE International Conference on Network Protocols (ICNP '02)*, pages 236–245, November 2002.

[7] M. Dahlin, B. Baddepudi V. Chandra, L. Gao, and A. Nayate. End-to-end WAN service availability. *IEEE/ACM Transactions on Networking*, 11(2):300–313, April 2003.

[8] G. Deconinck, H. Beitollahi, G. Dondossola, F. Garrone, and T. Rigole. Testbed deployment of representative control algorithms. Project CRUTIAL EC IST-FP6-STREP 027513 Deliverable D9, January 2008.

[9] D. Dzung, M. Naedele, T. P. Von Hoff, and M. Crevatin. Security for industrial communication systems. In *Proceedings of the IEEE*, volume 93, pages 1152–1177, June 2005.

[10] G. Franceschinis, E. Alata, J. Antunes, Hakem Beitollah, A. N. Bessani, M. Correia, W. Dantas, G. Deconinck, M. Kaâniche, N. Neves, V. Nicomette, P. Sousa, and P. Veríssimo. Experimental validation of architectural solutions. Project CRUTIAL EC IST-FP6-STREP 027513 Deliverable D20, March 2009.

[11] F. Garrone. Private communication. October 2008.

[12] V. M. Igure, S. A. Laughter, and R. D. Williams. Security issues in SCADA networks. *Computers & Security*, 25:498–506, October 2006.

[13] A. Markopoulou, G. Iannaccone, S. Bhattacharyya, C. Chuah, Y. Ganjali, and C. Diot. Characterization of failures in an operational IP backbone network. *IEEE/ACM Transactions on Networking*, 16(4):749–762, August 2008.

[14] V. Rosato and et al. IRRIIS Project, Deliverable D2.1.2: Final report on analysis and modelling of LCCI topology, vulnerability and decentralised recovery strategies. http://www.irriis.org/File.aspx?lang=2&oiid=9135&pid=572, September 2007.

[15] A. C. Snoeren, K. Conley, and D. K. Gifford. Mesh-based content routing using XML. In *Proceedings of the 18th ACM Symposium on Operating Systems Principles (SOSP'01)*, pages 160–173, October 2001.

[16] P. Veríssimo, N. F. Neves, and M. Correia. The CRUTIAL reference critical information infrastructure architecture: a blueprint. *International Journal of System of Systems Engineering*, 1(1/2):78–95, 2008.

[17] C. Wilson. Terrorist capabilities for cyber-attack. In M. Dunn and V. Mauer, editors, *International CIIP Handbook 2006*, volume II, pages 69–88. Center for Security Studies, ETH Zurich, 2006.