A Security-aware Routing Implementation for Dynamic Data Protection in Zone-based MPSoC

Johanna Sepulveda
Institute for Security in Information Technology
Technical University of Munich
Munich, Germany
johanna.sepulveda@tum.de

Ramon Fernandes
Pontifical Catholic University of Rio Grande do Sul
Porto Alegre, Brazil
ramon.fernandes@acad.pucrs.br

César Marcon
Pontifical Catholic University of Rio Grande do Sul
Porto Alegre, Brazil
cesar.marcon@pucrs.br

Daniel Florez
University of Los Andes
Bogota, Colombia
dm.sepulveda2246@uniandes.edu.co

Georg Sigl*
Institute for Security in Information Technology
Technical University of Munich
Munich, Germany
Gsigl@tum.de

ABSTRACT

This work proposes a secure Network-on-Chip (NoC) approach, which enforces the encapsulation of sensitive traffic inside the asymmetrical security zones while using minimal and non-minimal paths. The NoC routing guarantees that the sensitive traffic communicates only through trusted nodes, which belong to a security zone. As the shape of the zones may change during operation, the sensitive traffic must be routed through low-risk paths. The experimental results show that this proposal can be an efficient and scalable alternative for enforcing the data protection inside a Multi-Processor System-on-Chip (MPSoC).

CCS CONCEPTS

• **Networks** → **Layering**; • **Security and privacy** → *Hardware-based security protocols*;

KEYWORDS

MPSoC, NoC, Security, Zones, Encapsulation

ACM Reference format:
Johanna Sepulveda, Ramon Fernandes, César Marcon, Daniel Florez, and Georg Sigl. 2017. A Security-aware Routing Implementation for Dynamic Data Protection in Zone-based MPSoC. In *Proceedings of SBCCI '17, Fortaleza - Ceará , Brazil, August 28-September 01, 2017,* 6 pages.
https://doi.org/10.1145/3109984.3109996

1 INTRODUCTION

Multi-Processor System-on-Chip (MPSoC) is characterized by its flexibility and high computational capability [1]. It integrates dozens of computation and storage Intellectual Property (IP) cores, which exchange information through a special communication architecture like a Network-on-Chip (NoC). This communication architecture exchanges packets from a source IP to a destination IP through communication paths composed by routers and links [2].

MPSoCs can support several applications, which may be stored on a chip or downloaded through external networks like the Internet. Applications are spread over the IPs of the MPSoC. Due to performance requirements and power constraints, application mapping may change during execution time. For critical applications, splitting an application into several IPs forces the sensitive data exchanging through shared NoC resources, exposing data to attacks as shown in [3–6].

MPSoCs are now target of several attacks [7, 8]. Malicious entities profit of the hyper-connectivity of Internet-of-Things (IoT) to download malware onto MPSoCs and infect IPs. Such kind of remote software-based attacks accounts for 80% of the security incidents in MPSoCs [9]. Remote timing attacks belong to this category. Such attacks exploit the data leakage caused by shared resources of the MPSoC: processing elements, memories, and the communication structure. Sensitive communication at NoCs must be protected. The work of [10] shows that by exploiting NoC communication collisions among sensitive and the attacker traffic, the secret key of a sensitive application may be retrieved.

Previous works [5, 11, 12] have shown that NoCs can be enhanced with security mechanisms to prevent and mitigate attacks. Firewalls, customized network protocols, and customized routers are used to build security zones. These zones encapsulate the sensitive traffic into trusted areas. They are constituted by a set of trusted IPs and routers, in which only sensitive and trusted traffic is exchanged, therefore avoiding collisions with malicious traffic. Customizing the router through routing modification is one of the most effective techniques to build security zones and protect traffic [6, 10]. In [6], the authors propose dynamic risk-based routing to encapsulate traffic in low-risk paths. The risk value is provided by the number of firewall activations. Despite the fast runtime configuration, the lowest-risk path cannot be guaranteed due to the minimal path constraint. In [10], a design-time approach based on

*Also with Fraunhofer Institute for Applied and Integrated Security.

SBCCI '17, August 28-September 01, 2017, Fortaleza - Ceará , Brazil

ACM ISBN 978-1-4503-5106-5…$15.00
https://doi.org/10.1145/3109984.3109996

region routing is used for guaranteeing the encapsulation of traffic inside asymmetric security zones. However, this approach is not suitable for reshaping the security zones at runtime.

To overcome such drawbacks, this work proposes Non-minimal Odd-Even Region-based routing NoC (NOE-RNoC), an architecture that combines the Region-Based Routing (RBR) [13] mechanism at design time and non-minimal adaptive routing at runtime to efficiently encapsulate sensitive traffic.

The contributions of this work are:

- Implementation of a non-minimal adaptive routing technique guided by the security metric - risk of the hop;
- Provide fast reconfiguration of the region-based routing mechanism; and
- Evaluation of performance, cost, and security.

This paper is divided into seven sections. Section 2 presents previous works on NoC-based security. Section 3 describes the target MPSoC and threat model. Section 4 presents the mechanisms for security zones protection. Section 5 describes the architecture of NOE-RNoC. Section 6 shows the experimental work and results. Finally, Section 7 presents our conclusions.

2 RELATED WORK

Security integration at NoC-based architectures has been demonstrated as an effective solution to protect heterogeneous MPSoCs. Such mechanisms avoid unauthorized data modification, extraction and system service deny. Security zones can be implemented through the NoC resources. The goal is to protect the MPSoC by encapsulating the sensitive traffic into trusted areas. The works of [10, 14] use routing to encapsulate the sensitive traffic. The authors of [10] present an RBR approach based on the security characteristics of the application. Despite the good results, it does not consider security zone reshaping during runtime. In the work of [14], the risk metric is used to guide the routing of sensitive traffic. The risk of the path is evaluated at the destination interface. When a risk threshold is exceeded, a new low-risk path is explored.

Four routing alternatives are used (deterministic, hop-based, weighted and bounded). All these routing algorithms are constrained by the minimal path, thus restricting the search of low-risk paths.

3 MPSOC DESCRIPTION AND THREAT MODEL

MPSoCs integrate a set of IP cores that process and store data for executing an application divided into tasks and split on the IPs of the MPSoC. Communication among IPs is performed through the NoC, a network of routers and links inside the chip. Figure 1 shows an example of an MPSoC of 16-IP cores linked through a 16-router NoC. The communication between the target IP (IP_t) linked at router R_3 and the source IP (IP_s) connected at router R_9 requires five commutations on the NoC (R_9, R_{10}, R_{11}, R_7, R_3) with XY routing.

We consider that the MPSoC executes a sensitive (S) and other types of applications simultaneously in the same chip. S is split into IP_9 and IP_3; thus, forcing the communication of sensitive data through the NoC (1). The NoC path used to communicate the sensitive data is called the sensitive path. The sensitive path of Figure 1 is constituted by the routers R_9, R_{10}, R_{11}, R_7, R_3. The NoC and

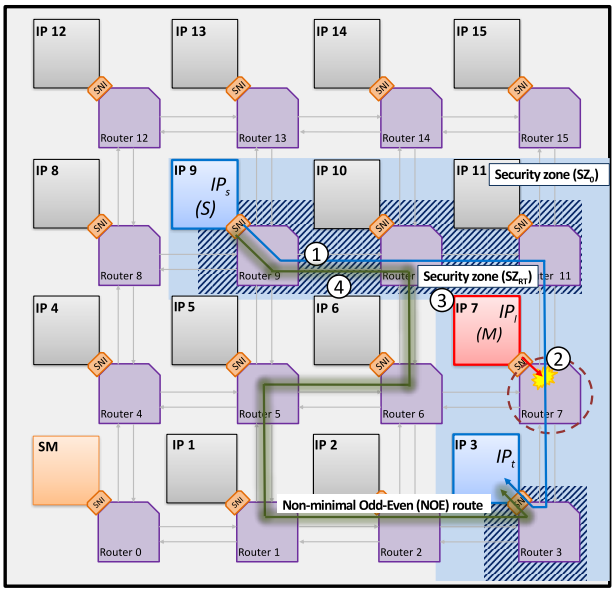

Figure 1: Example of an MPSoC with 16-IP cores containing sensitive traffic and secure zones.

interfaces are considered secure; i.e., the attacker cannot modify their behavior. The attacker can infect the IP cores by executing a malicious application (M) into the MPSoC. The malicious task may be installed on an IP; thus, turning it into an infected IP (IP_I).

Figure 1 shows the infected IP_7, which is linked directly to a router inside the sensitive path. The attacker can control the traffic injection and monitor the IPI throughput. As shown in [15, 16], an attacker may exploit communication collisions between the sensitive and malicious traffic to perform timing attacks. Collisions allow the attacker to recognize the traffic pattern of the sensitive traffic employing the degradation of the IP_I throughput. The collision in Figure 1 takes place at R_7 (2). As a result, the authors of [15] have shown that by observing traffic due to 76 AES encryption, IP_I can retrieve 12 of the 16 bytes of the secret key. Complementary, a brute force attack can be used to reveal the complete secret key.

The following preconditions are required to perform the attack:

- The attacker can infect an IP of the MPSoC;
- The attacker can control the traffic generation and monitoring of the infected IP; and
- Infected IP is in the sensitive path.

4 MECHANISMS FOR SECURITY ZONES PROTECTION

A Security Zone (SZ) is a physical space (continuous or disrupted) that wraps and isolates the IPs that execute sensitive applications. IPs that belong to the SZ are considered trusted among them. The task mapping of sensitive applications inside the MPSoC defines the shape of the SZ. However, if a trusted IP is attacked or the mapping of the application is modified at runtime, the SZ must be reshaped. The NoC routing can be employed to create SZs. The routing logic selects the router output for granted input. Therefore,

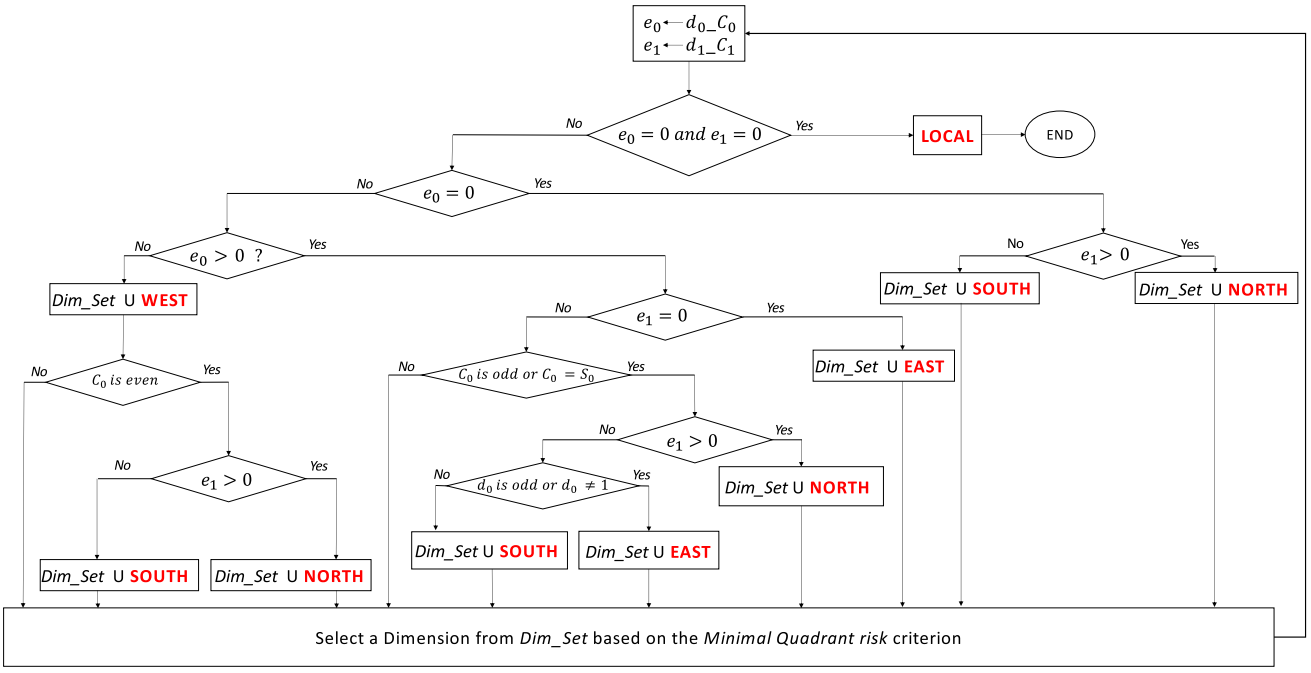

Figure 2: Non-minimal adaptive odd-even turn model (NOE).

it can be used to restrict the communication through hops inside the SZ. Reshaping the SZ implies runtime modification of the NoC routing, and the new route must be secure.

Firewalls embodied in the NoC interfaces are commonly used to protect traffic and enforce the security policy of the system. This information can be used to detect possible points of attack and to drive the runtime modification of the NoC routing. Figure 1 shows an initial continuous security zone SZ_0 (1) that includes IP_9, IP_{10}, IP_{11}, IP_7 and IP_3. However, IP_7 is infected at runtime and detected by the firewalls (2), which trigger a reshape of the SZ. The infected IP is removed from the SZ, and a new disrupted SZ is created (SZRT) as in (3). A new sensitive path is computed (4), which requires seven commutations on the NoC (R_9, R_{10}, R_6, R_5, R_1, R_2, R_3).

This work proposes an architecture able to establish and modify SZs, as well as to reroute sensitive traffic through non-minimal routes driven by the risk level of each hop. The proposed NoC supports dynamic SZs and protected communications inside the MPSoC by combining region-based routing (at design time) and Non-minimal Odd-Even routing (at runtime).

4.1 Region-based routing (design time)

Our region-based routing approach forces that the communication paths between any pair of IP cores that belong to the same SZ are performed inside the SZ. Determining the routes inside a region while guaranteeing deadlock-free routes is a complex task. The Segment-Based Routing (SBR) [17] and Region-Based Routing (RBR) [13] algorithms are used to determine routes for encapsulating traffic into a region. SBR is responsible for deadlock prevention and IP cores reachability, while RBR computes the routing tables.

SBR is composed of two steps: (i) segment computation, which splits the NoC into segments characterized by having a turn restriction to avoid deadlocks; and (ii) placement of routing restrictions. RBR uses the turn restrictions computed by SBR to find paths between all origins and destinations in the NoC. It includes three steps: i) routing computation, for each source-target IP cores pair; ii) region computation, that joins at each router multiple routing entries based on the input and output port values; and iii) region merge, which merges overlapping routing entries to reduce the size of the routing tables.

The designer can decide the IP core members of SZs and the mapping on the MPSoC. The SBR algorithm is used to compute the segments and turn restrictions required to keep traffic inside an SZ. The goal is to create the smallest possible segments that contain elements from the same SZ. The RBR algorithm searches the paths between each pair of IP_s and IP_t and creates the routing tables (RBR tables) for each router. Such network constitutes the Region-based routing NoC (RNoC). More details about the algorithm can be found in [10].

The high complexity of SBR and RBR turns prohibitive the utilization of such algorithms at runtime. Thus, when the security characteristics changes, a lighter approach must be used to find a low-risk path for sensitive traffic.

4.2 Non-minimal Odd-Even NOE routing (runtime time)

Non-minimal Odd-Even (NOE) is a low-cost adaptive routing algorithm able to follow different paths between a given (IP_s, IP_t) pair.

It is a deadlock-free adaptive approach that restricts the locations at which turns can be performed. It is based on two rules [18]:

- Rule1: Any packet is not allowed to take an *East-North* turn at any nodes located in an even column, and it is not allowed to take a *North-West* turn at any nodes located in an odd column; and
- Rule2: Any packet is not allowed to take an *East-South* turn at any nodes located in an even column, and it is not allowed to take a *South-West* turn at any nodes located in an odd column.

Figure 2 shows the behavior of NOE, which analyzes and compares the packet destination (d_0, d_1) and the current router position (c_0, c_1) to select the proper router output port. Packets can be routed adaptively in East, West, North or South directions. NOE can be used to find a low-risk path for sensitive packets at runtime.

The routing decision can be driven by the risk value of hops. At the NOE-RNoC, each hop quantifies the weighted risk value of the four quadrants as shown in [14]. The value of the neighbors' risk together and the turn restriction are taken into account to select the next hop. This technique avoids that packets are trapped into dangerous paths. Each time the risk value of a hop is updated, the new value is broadcasted into the NoC hops that belong to the column and row. The implementation of adaptive routing may present higher costs when compared to deterministic approaches. However, adaptation may become mandatory for reliability purposes and so the cost overhead is acceptable.

5 NOC ARCHITECTURE

The proposed Non-minimal Odd-Even Region-based routing NoC (NOE-RNoC) is a security-enhanced NoC that protects sensitive traffic inside the MPSoC. Sensitive traffic is encapsulated dynamically through low-risk paths inside a security zone. At design time, RNoC is used to determine routing tables. At runtime, new sensitive paths are created through NOE routing driven by the risk metric. The protected NOE-RNoC is based on a two-level NoC composed of three main components: Secure Network Interface (SNI), hop and Security Manager (SM). Their microarchitecture is shown in Figure 3.

SNIs implements the communication protocol (pack/unpack, route table, control) and security checking employing firewalls (security table). The packet structure is composed of 6 fields: i) source, to identify IP_s; ii) destination, to identify IP_t; iii) route, to store the secure route; iv) risk threshold, containing the maximum risk allowed per hop; v) operation, to identify the type of packet (control, data write, data read); and vi) payload, which is the data to be exchanged. The firewall-based traffic inspection enforces communication security. Each time a packet is injected or received, and security checking is performed. At the source IP, access control is performed by verifying the destination and operation fields of the packet. At the destination IP, authentication is performed by checking the source and operation fields. When security rules are violated, the firewall generates a notification, which will increase the risk value of the hop: i) at source router when the attack is identified at the source network interface; or ii) at the hops used by the malicious packet when the attack is identified at the destination network interface. Each time a new application is mapped into

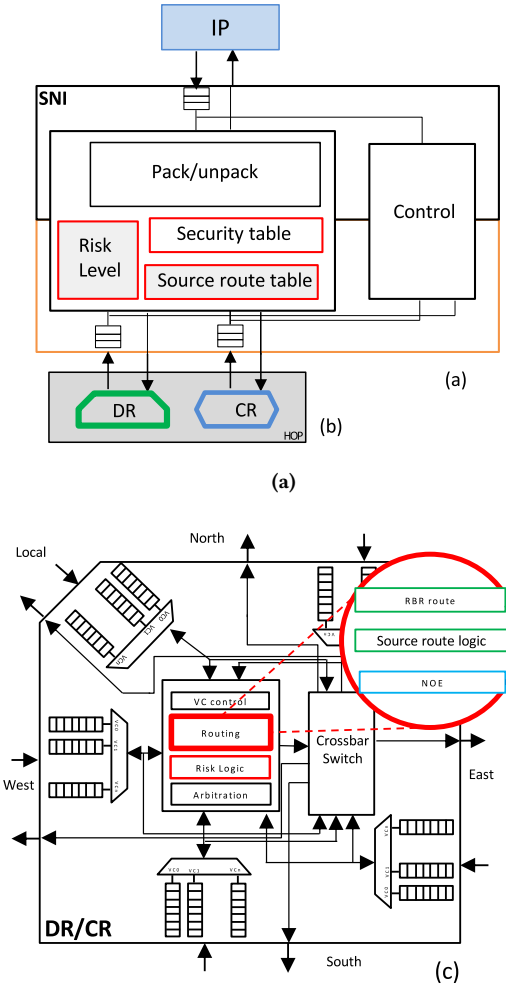

Figure 3: Microarchitecture of SNI (a) and Router (b).

the system, the risk level of the hops linked to the modified IPs is restarted.

Hops integrate the Data Routers (DRs) and Control Routers (CRs) to exchange data and control signals. It also includes the RISK logic block used to quantify and store the risk value of each hop as in [14]. Figure 3 shows the router structure. The difference between DR and CR is the link size and the routing implementation. DRs route packets using RBR tables. CRs use RBR tables and NOE (used only for seeker packets). Each hop stores two risk values: *Localrisk* (hop risk) and *Quadrantrisk* (risk of the line and column neighbors). *Localrisk* is used to determine if a hop is dangerous. Each time, a sensitive packet uses a DR, the *Localrisk* is compared to a risk threshold value. If exceeded, the packet is sent back to the IPs and SM is notified. *Quadrantrisk* is used for NOE routing. Quadrants

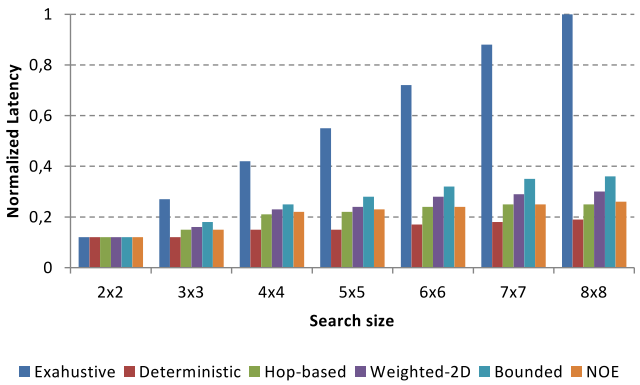

Figure 4: Results of the scalability of the approaches.

that force the transition between SZs are penalized to favor routing inside a single SZ.

SM is a light software layer executed in a trusted IP in charge of the configuration of firewalls and control of the recovery mechanism under a possible attack. It includes the untrusted hop removal from an SZ.

During execution time, the risk of a hop can be measured as in [14]. The risk is defined as the probability that a malicious process spies, denials the communication or corrupts the data in a NoC hop. The risk is measured by the number of firewall notifications due to the violation of security rules. When the risk of a hop inside a SZ overcomes the *RISKlevel* value, defined by the designer, the hop is removed from the SZ. Therefore, the IP cores of a SZ that use the removed hop must search for an alternative low-risk path. These IPs inject a seeker packet which is commuted through the CR. The routing decision at each router is based on the NOE algorithm that includes the risk value of each hop of the NoC. The seeker packet stores the route and then it is stored into the source route table of the SNI. SM performs hops removal and control of the seek process.

6 EXPERIMENTS AND RESULTS

NOE-RNoC is modeled in SystemC-TLM and VHDL-RTL by extending the NoC design framework presented in [14]. SHOC is a modular cycle accurate simulation environment, which supports a wide variety of components required for MPSoC simulation. This environment includes libraries of MPSoC attacks and tools for power and area estimation. NOE has been evaluated under three conditions: scalability, performance, and security. NOE-RNoC is compared with the approaches proposed in [14].

Figure 4 shows the impact of setting a new route by using the NOE and the previous approaches for different NoC sizes. The path length is equivalent to the diameter of the NoC. Results are expressed as a percentage of the exhaustive route search. Lower latency values represent efficient routing techniques. Results show that NOE is scalable and only hop-based and deterministic approaches overcome NOE. These approaches do not require the risk status broadcasting. Oblivious neighbor risk approaches may limit the search of low-risk paths. Among the approaches where routers

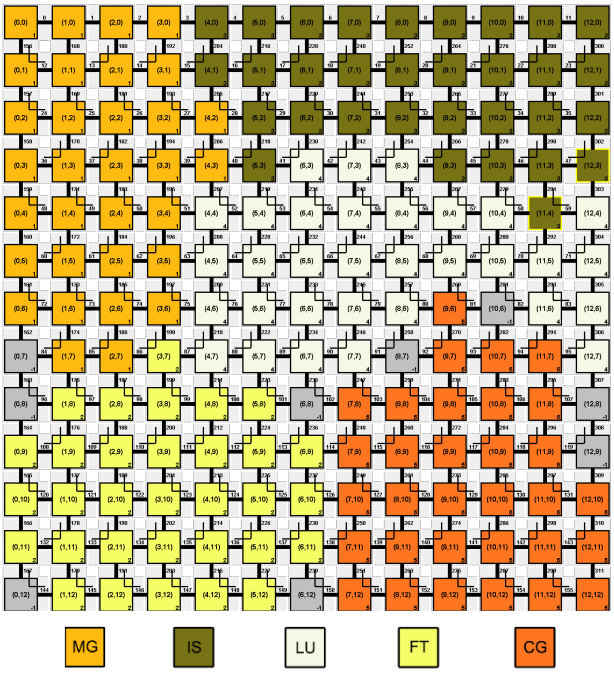

Figure 5: Mapping of NASA NAS benchmark in the target MPSoC.

are aware of the NoC risk (weighted and bounded), NOE presents the best performance. NOE enhances performance up to 8% and 15% when compared to the weighted and bounded approaches, respectively.

The performance evaluation was carried out on an MPSoC that supports five applications (MG, IS, LU, FT, CG) of the NASA Numerical Aerodynamic Simulation (NAS) Benchmark. Figure 5 shows the MPSoC mapping obtained by CAFES [19]. This tool optimizes the MPSoC mapping according to performance and power metrics. Each application is grouped into a single and continuous SZ. During operation time, two IP cores from each SZ start to behave maliciously. This experiment emulates the presence of hardware Trojans on the MPSoC. Thus, the routing inside the SZ must be modified.

Figure 6 shows the performance results of the NOE-RNoC under uniform traffic with some injection rates. The path reconfiguration was forced during 25% of the operation time. The results show that NOE achieves the best performance results, overcoming the hop-based approach. Despite NOE-RNoC requiring the broadcast of the risk values of the hops to quantify the *Quadrantrisk*, the performance of applications is not affected. NOE employs the CR for all the extra communication. Moreover, the path found by the hop-based approach falls into infected hops that were performing timing attack (heavy traffic injection to detect the degradation of throughput); thus, degrading the performance of the sensitive path. NOE was able to avoid such hops.

Table 1 summarizes the area, power and performance overhead of the security mechanisms as a percentage of the penalty of each

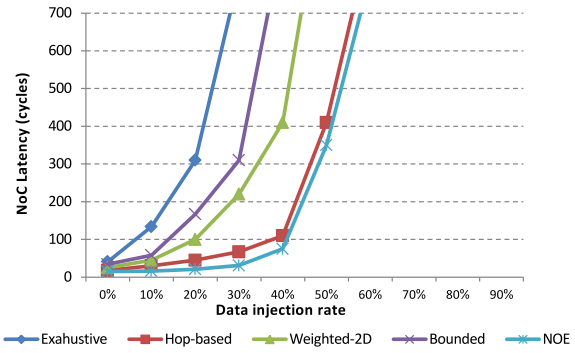

Figure 6: NoC latency results regarding data injection rates.

Table 1: Overhead when compared to a simple two-level NoC configuration.

Configuration	Latency	Area	Power
Deterministic	6.5%	9.6%	12.3%
Hop-based	8.3%	6.4%	5.1%
Weighted-2D	12.6%	14.3%	8.5%
Bounded	14.3%	8.6%	7.3%
NOE	7.8%	7.2%	5.8%

Table 2: Security evaluation results.

Attack scenario	Exhaustive	Hop	Weighted	Bounded	NOE
Overwrite memory	100%	100%	100%	100%	100%
Read memory	100%	100%	100%	100%	100%
Repeated packet	100%	86%	84%	100%	100%
Wrong destination	100%	87%	93%	100%	100%

configuration when compared to the same MPSoC without protection. Results show that NOE presents the best trade-off among the alternatives that are NoC risk-aware.

For the security evaluation, our approach was evaluated under four kinds of attacks. Table 2 shows the results of the security evaluation. Higher values of attack avoidance represent a better level of protection. Results show that NOE and Bounded approaches achieve the highest protection levels for all the attacks.

7 CONCLUSIONS

This work proposes NOE-RNoC, a security enhanced NoC architecture that combines region routing and non-minimal risk-based routing techniques to encapsulate sensitive traffic through low-risk paths. We present three main contributions. Firstly, we implement a non-minimal routing technique driven by a low-risk metric. Secondly, we show that the proposed architecture can protect sensitive traffic even when some IP cores inside the security zones are tampered. Consequently, NOE-RNoC can find routes at runtime. Thirdly, we show that NOE-RNoC is efficient and able to protect sensitive traffic. Future work aims to explore other non-adaptive weighted routing techniques for finding secure paths efficiently.

REFERENCES

[1] T. T. Ye, L. Benini, and G. D. Micheli, "Packetized on-chip interconnect communication analysis for MPSoC," in *Design, Automation and Test in Europe Conference and Exhibition (DATE)*, pp. 344–349, 2003.
[2] L. Benini and G. D. Micheli, "Networks on chips: a new SoC paradigm," *Computer*, vol. 35, pp. 70–78, Jan 2002.
[3] L. Fiorin, G. Palermo, S. Lukovic, and C. Silvano, "A data protection unit for NoC-based architectures," in *International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS)*, pp. 167–172, Sept 2007.
[4] M. D. Grammatikakis, K. Papadimitriou, P. Petrakis, A. Papagrigoriou, G. Kornaros, I. Christoforakis, and M. Coppola, "Security effectiveness and a hardware firewall for MPSoCs," in *IEEE International Conference on High Performance Computing and Communications, IEEE International Symposium on Cyberspace Safety and Security, IEEE International Conference on Embedded Software and Syst (HPCC,CSS,ICESS)*, pp. 1032–1039, Aug 2014.
[5] J. Sepúlveda, D. Flórez, and G. Gogniat, "Efficient and flexible NoC-based group communication for secure mpsocs," in *International Conference on ReConFigurable Computing and FPGAs (ReConFig)*, pp. 1–6, Dec 2015.
[6] J. Sepúlveda, D. Flórez, and G. Gogniat, "Reconfigurable security architecture for disrupted protection zones in NoC-based MPSoCs," in *International Symposium on Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC)*, pp. 1–8, June 2015.
[7] K. Patel, S. Parameswaran, and R. G. Ragel, "Architectural frameworks for security and reliability of MPSoCs," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, pp. 1641–1654, Sept 2011.
[8] D. M. Ancajas, K. Chakraborty, and S. Roy, "Fort-NoCs: Mitigating the threat of a compromised noc," in *ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 1–6, June 2014.
[9] L. Fiorin, C. Silvano, and M. Sami, "Security aspects in networks-on-chips: Overview and proposals for secure implementations," in *Euromicro Conference on Digital System Design Architectures, Methods and Tools (DSD)*, pp. 539–542, Aug 2007.
[10] R. Fernandes, C. Marcon, R. Cataldo, J. Silveira, G. Sigl, and J. Sepúlveda, "A security aware routing approach for NoC-based MPSoCs," in *Symposium on Integrated Circuits and Systems Design (SBCCI)*, pp. 1–6, Aug 2016.
[11] T. Wehbe and X. Wang, "Secure and dependable NoC-connected systems on an FPGA chip," *IEEE Transactions on Reliability*, vol. 65, pp. 1852–1863, Dec 2016.
[12] J. Porquet, A. Greiner, and C. Schwarz, "NoC-MPU: A secure architecture for flexible co-hosting on shared memory MPSoCs," in *Design, Automation Test in Europe (DATE)*, pp. 1–4, March 2011.
[13] J. Flich, A. Mejia, P. Lopez, and J. Duato, "Region-based routing: An efficient routing mechanism to tackle unreliable hardware in network on chips," in *International Symposium on Networks-on-Chip (NOCS'07)*, pp. 183–194, May 2007.
[14] J. Sepulveda, D. Flórez, R. Fernandes, C. Marcon, G. Gogniat, and G. Sigl, "Towards risk aware NoCs for data protection in MPSoCs," in *International Symposium on Reconfigurable Communication-centric Systems-on-Chip (ReCoSoC)*, pp. 1–8, June 2016.
[15] C. Reinbrecht, A. Susin, L. Bossuet, and J. Sepúlveda, "Gossip NoC – avoiding timing side-channel attacks through traffic management," in *IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, pp. 601–606, July 2016.
[16] M. J. Sepúlveda, J. P. Diguet, M. Strum, and G. Gogniat, "NoC-based protection for SoC time-driven attacks," *IEEE Embedded Systems Letters*, vol. 7, pp. 7–10, March 2015.
[17] A. Mejia, J. Flich, J. Duato, S. A. Reinemo, and T. Skeie, "Segment-based routing: an efficient fault-tolerant routing algorithm for meshes and tori," in *IEEE International Parallel Distributed Processing Symposium (IPDPS)*, pp. 1–10, April 2006.
[18] G.-M. Chiu, "The odd-even turn model for adaptive routing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 11, pp. 729–738, Jul 2000.
[19] C. Marcon, N. Calazans, E. Moreno, F. Moraes, F. Hessel, and A. Susin, "A framework for intrachip application modeling and communication architecture design," *Journal of Parallel and Distributed Computing*, vol. 71, pp. 714–728, May 2011.