# Dynamic and Distributed Security Management for NoC based MPSoCs

Siavoosh Payandeh Azad[1][0000-0001-9177-7779], Gert Jervan[1][0000-0003-2237-0187], and Johanna Sepulveda[2][0000-0003-3686-994X]

[1] Tallinn University of Technology, Tallinn, Estonia. siavoosh.azad@taltech.ee, gert.jervan@taltech.ee

[2] Institute for Security in Information Technology, Technical University of Munich, Munich, Germany. johanna.sepulveda@tum.de

**Abstract.** Multi-Processors System-on-Chip (MPSoCs) have emerged as the enabler technology for new computational paradigms such as Internet-of-Things (IoT) and Machine Learning. Network-on-Chip (NoC) communication paradigm has been adopted in several commercial MPSoCs as an effective solution for mitigating the communication bottleneck. The widespread deployment of such MPSoCs and their utilization in critical and sensitive applications, turns security a key requirement. However, the integration of security into MPSoCs is challenging. The growing complexity and high hyper-connectivity to external networks expose MPSoC to Malware infection and code injection attacks. Isolation of tasks to manage the ever-changing and strict mixed-criticality MPSoC operation is mandatory. Hardware-based firewalls are an effective protection technique to mitigate attacks to MPSoCs. However, the fast reconfiguration of these firewalls impose a huge performance degradation, prohibitive for critical applications. To this end, this paper proposes a lightweight broadcasting mechanism for firewall reconfiguration in NoC-based MPSoC. Our solution supports efficient and secure creation of dynamic security zones in the MPSoC through the communication management while avoiding deadlocks. Results show that our approach decreases the security reconfiguration process by a factor of 7.5 on average when compared to the state of the art approaches, while imposing negligible area overhead.

**Keywords:** MPSoCs · Network-on-Chip · Security.

## 1 Introduction

Multi-Processors Systems-on-Chip (MPSoCs) integrate a variety of different Intellectual Property (IP) hardware cores (e.g., processor, memory, interfaces) that communicates through a Network-on-Chip (NoC). The NoC integrates routers and links to exchange the IP cores data encapsulated as packets. MPSoCs enable the development of cutting edge technologies. The flexibility and computational power has turned MPSoCs into a perfect solution for many critical applications. They are widely adopted in several critical applications (e.g., automotive,

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_46

avionics) and deployed in different commercial products such as Tile-Mx100 from Tilera [1], MPPA from Kalray [2] or SCC from Intel [3]. However, the mixed-criticality nature of the MPSoCs has turned security into an important requirement. Integrating security at MPSoCs is challenging.

High VLSI integration levels, cache hierarchies and complex MPSoC architecture bring on a wide surface of attacks. For instance, MPSoCs applications are stored in external non-volatile memories which are prone to malicious modifications. Moreover, MPSoCs are able to support several applications, usually characterized by different levels of criticality. When executed simultaneously, these applications are forced to share several MPSoC resources, such as processors, memories and communication structure. Code injection attacks exploit these two aforementioned MPSoC characteristics to retrieve secret information (i.e., cryptographic keys, passwords), gain control over the critical applications or deny the completion of critical tasks. In order to mitigate code injection attacks, physical and temporal isolation of the applications is required.

Previous works have proposed isolation through software security mechanisms. However, it has been shown the effectiveness of the micro-architectural attacks even under restrictive software-controlled isolation scenarios [18, 20]. Thus, in order to guarantee the security of a system, hardware security must be also considered [25]. Hardware firewalls integrated in the network interface of the MPSoCs, between the IP core and the NoC routers, have shown to be an effective isolation mechanism. These firewalls store the security policy (set of security rules) of the system and verify packet-wise the communication rights. When the packet content matches the security policy, the transaction takes place. Otherwise, the transaction is discarded and an alarm is triggered to activate a recovery mechanism. In order to be compliant with the MPSoC protection requirements, firewalls should be dynamically reconfigured based on the criticality and security requirements of the tasks being executed on the system. Moreover, the reconfiguration latency should meet the performance requirements of the system. Previous works built security zones using a single manager and unicasting communication mechanism. However, these techniques do not scale with larger NoC sizes and impose large performance penalty due the reconfiguration overhead. In order to overcome these drawbacks, in this work we propose: i) a new method for partitioning the network for reconfiguration; ii) a light-weight broadcasting solution for secure firewall reconfiguration; and iii) a distributed solution for MPSoCs security managers.

The rest of this paper is organized as follows: Section 2 describes the related works. Section 3 provides details of the proposed multi-zone reconfiguration broadcasting mechanism. Section 4 evaluates the proposed mechanism and finally Section 5 concludes the paper.

## 2 Related Work

Security integration in NoC-based MPSoCs through hardware firewalls has been target of wide research [5, 7–11, 13, 16, 26]. According to the reconfigurabil-

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_46

ity and granularity characteristics of the firewalls, they can be classified as static/dynamic and single/multi-level firewalls. Static and single-level firewalls were proposed in [8, 10, 11]. These firewalls are appropriate for MPSoCs that support fixed and static applications characterized by a single level of criticallity and will fail to protect dynamic and mixed-critically MPSoCs (firewall policies should be updated and refreshed at run-time). In order to protect these MPSoCs enhanced firewall management is required. The work of [17, 19] proposed the integration of static and multi-level firewalls. Despite the good results for mixed-critical applications, the dynamic nature of some of these applications is not yet supported. Protection of MPSoCs with dynamic behaviour was proposed in the works of [24, 26]. These works presented reconfigurable firewalls able to built and reshape in run-time security zones that isolate a set of IP cores with the same criticality. The dynamic reconfigurable firewall-based MPSoC protection presents a wide design space exploration. The design of the dynamic firewalls structure has a huge impact on the system performance. These parameters include: I) location and the number of security managers (SM) that control the firewalls; II) firewalls size; III) granularity of the security control; IV) number of security levels; V) number of security zones; and vi) management of the reconfiguration traffic. The works of [12, 15, 22, 23, 27] explore the number of security zones and number of security levels. Despite the good results, the above-mentioned works have not considered the effect of firewalls size, granularity of the security control, distribution of security managers and management of the reconfiguration traffic. This exploration was partially addressed by the authors of [6]. However, the effect of impact of the location and number of Security Managers and the management of the reconfiguration traffic are still open questions. The goal of this work is to answer the above mentioned open issues. We propose and evaluate the impacts of a dynamic, lightweight, multi-layer, multi-manager, multi-zone security infrastructure in NoC-based MPSoC using packet broadcasting.

## 3 Multi-Zone Security Broadcasting

NoC-based MPSoCs are able to support several applications characterized by different levels of criticality. These application can be divided into smaller pieces of code, called tasks, which are split and mapped (dynamically) into the different MPSoC computation and storage resources. This fact, speeds up the execution of the application, but forces the peer IP interaction through the NoC. Collision among critical tasks may leak information regarding the critical and secret data. Moreover, the dynamic and mix-criticality nature of the different tasks require frequent updating capabilities of the security mechanisms. Previous attacks, such as Meltdown and Spectre [18, 20], have shown that efficient isolation among critical applications is mandatory. Hard and inefficient updating security checks may be exploited to gain control over the system resources and to leak the secret information. They also show that hardware firewall technology may be efficient in such a threats, once no operation is executed before the permissions are verified.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_46

Hardware-based security provides an efficient mean for access control enforcement. 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. Firewall information should be able to be updated, once the permissions of the applications being executed in the MPSoC may change. There are different alternatives to update the firewall information. One alternative is the uni-cast updating, however it incurs into a high system performance and power degradation.

In this work we propose a flexible and efficient security zone creation through the fast configuration of hardware-based firewall structure embedded into the Network Interface (NI) and security routers. The efficient update capability of the firewalls and routing mechanism is achieved through two techniques: I) utilization of local security managers at each zone; and II) hybrid communication, which uses a lightweight broadcasting approach for the security reconfiguration traffic and uni-casting communication for the remaining traffic. As a consequence efficient and elastic security zones are created into the MPSoC.

### 3.1 Multi-Zone Security Broadcasting Mechanism

NoC-based broadcasting enables the transmission of information to all the IP cores that belong to the network. Classical NoC-based communication is carried out through uni-cast data exchange between a sender-receiver pair. Information is wrapped as packets. Fig. 1 shows the packet format used in this work, which is compatible with the Open Core Protocol (OCP). It embodies information that specifies the criticality of the transaction. Note that any other format can be used. Our secure MPSoC architecture is composed of security routers, security managers and network interfaces equipped with firewalls. Firewalls match the packet content with the security rules stored in LUTs at the NIs. Fig. 2 shows the firewall fields. A matching packet is allowed to communicated, otherwise, it is discarded. Firewalls are classified as *initiator firewall*, which checks injected packets to the NoC, and *target firewall* which check ejected packets from the NoC. The values stored in this tables are configured by a security manager via reconfiguration packets. Upon receiving a reconfiguration packet, the NI searches for the valid entries in the firewall table which match the node-ID field. If the entry exists in the table, it is updated, otherwise a replacement table policy will take place. The replacement policy will use empty entries in the table if they exist, otherwise, will evict an existing entry. The replacement policy will have an effect on the performance of the system if the number of the entries in the firewall tables is too small. The network is partitioned into several security zones, each being organized by a local security manager. A single system-wide security manager, mapped on a secure node manages secure system booting and provides the policies for the local managers. The boundaries of such zones are defined and updated by the system manager depending on the application characteristics. The reconfiguration events happen once a new task is mapped on a node which has a different privilege than the previously defined value.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_46

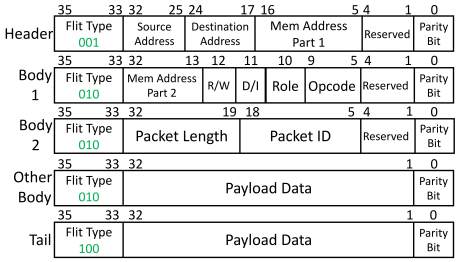



Fig. 1: Packet format

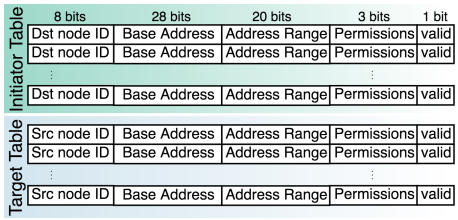



Fig. 2: Organization of firewall tables

Reconfiguration information is broadcasted from the local security manager to the nodes of the security zone.

The security router supports uni-casting and broadcasting mechanisms. Security routers are built upon a 5 input port wormhole switching and 3 pipeline stages as shown in Fig. 3. It is composed of four elements: i) input buffers, which store the input data arriving from an input port implemented as circular buffer FIFO; ii) routing unit, which selects the router output port in which incoming packets will be redirected, using Logic Based Distributed Routing (LBDR) mechanism; iii) switch allocator (SA), that grants the utilization of the crossbar switch to one of the input buffers; and iv) crossbar switch (Xbar), which links input to output ports of the router. Note that the security router can be built upon any other architecture. By managing the routing, security zones can be built. These zones apply only to the security management and has no effect on the network's packets (there will be no physical isolation for the normal packets). In order to have this property, we would need separate zone boundary and routing algorithm definition for security packets.

Previous works have implemented NoC broadcasting mechanisms [21]. They use a set of 8 bits $R_xy$ (routing bits) to define the underlying routing and two sets of 4 bits $C_x$ (connectivity bits) to express the connectivity of a router to its neighbors and regions. This information is used to compute the packet routing. In this work, an extra set of routing ($R_{xy\_bc}$) and connectivity bits ($C_{x\_bc}$) are used

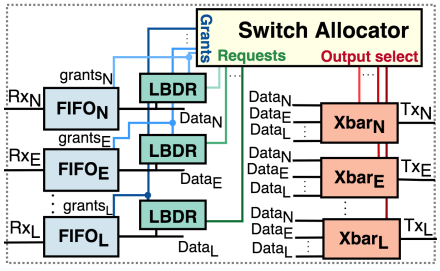



Fig. 3: Router architecture

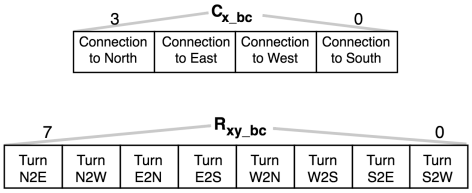



Fig. 3: Organization of $C_{x\_bc}$ and $R_{xy\_bc}$ bits

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_46

to implement the broadcasting and create security zones through the routing management.

Security zones are created using connectivity bits. They set the existence of a connection and the allowed communication direction. The $C_{x\_bc}$ bits are set to '1' if the communication with its adjacent neighbor router in a specific direction is allowed under the current security zone planning. Using the connectivity bits of each router for broadcasting messages, the NoC can be partitioned into different security zones. However, such portioning should be applied with care to avoid overlapping partitions. The connectivity bits for the broadcast messages can be dynamically reconfigured using the following approaches:

- To change the connectivity bits by the manager via reconfiguration packets.
- To modify the connectivity bits through OSR-lite[28].

However, the details of the reconfiguration mechanism for the dynamic update of the security zones are out of scope of this paper.

The routing unit uses the incoming input port signal (North, East, West, South, Local) information to select the output port. A set of signals describe the input direction of the routing logic. For example, signal $N' =' 1'$ means that the routing logic is located in the North input (hence shows the incoming direction of the broadcast packets).

The $R_{xy\_bc}$ bits represent the allowed turns for broadcasting packets (see Fig. 3). We define turn $X2Y$, as a turn from input $X$ to $Y$ of the router. Using this notation, the requests for broadcasting ($R_{N\_bc}, R_{E\_bc}, R_{W\_bc}, R_{S\_bc}, R_{L\_bc}$ for North, East, West, South and Local respectively) would be generated as in equation 1.

$$\begin{aligned}
R_{N_bc} &= [S' + L' + [E'.R_{xy}(2) + W'.R_{xy_bc}(4)]].C_{x_bc}(0)\\
R_{E_bc} &= [W' + L' + [N'.R_{xy_bc}(0) + S'.R_{xy_bc}(6)]].C_{x_bc}(1)\\
R_{W_bc} &= [E' + L' + [N'.R_{xy_bc}(1) + S'.R_{xy_bc}(7)]].C_{x_bc}(2)\\
R_{S_bc} &= [N' + L' + [E'.R_{xy_bc}(3) + W'.R_{xy_bc}(5)]].C_{x_bc}(3)\\
R_{L_bc} &= \neg L'
\end{aligned} \tag{1}$$

Signals $N', E', W', S'$ and $L'$ describe the direction of the incoming packet and are hardwired in each routing unit. For example, the request to north $R_{N\_bc}$ in Equation 1 is set to one if:

1. The router has connectivity towards north in the security zone ($C_{x\_bc}(0)$)
2. The routing logic is processing the broadcast packets:
   - from south or local packets; or
   - from West or East and West-to-North or East-to-North (these turns are allowed under the current routing algorithm).

These request signals are propagated to the *SA*. Upon receiving a broadcast message, *SA* awaits the release of the required output. The NI is enriched with a mechanism that suppresses the injection of broadcast messages from non-privileged nodes.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_46

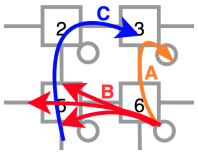
### 3.2 Discussion of deadlock-free

In order to avoid deadlocks, two cases should be considered: 1) a single broadcast message cannot contribute to the creation of deadlocks; and 2) multiple broadcast messages can not create deadlocks. To avoid deadlocks, the following assumptions must be met:

1. The broadcast routing algorithm must not add additional turns to the current networks routing.
2. Only security managers are allowed to broadcast.
3. Each network zone has one and only one manager.
4. The broadcast flits can only be forwarded if all destination FIFOs are empty (the router should block them until they become empty)

The first assumption guarantees that the allowed set of turns is kept. The work in [21] proves the deadlock freeness with the concurrent use of adaptive routing for broadcast and uni-cast messages. However, this approach requires the use of Virtual-Cut-Through (VCT) switching. This results in a poor buffer management. In order to avoid deadlocks and reduce the impact of VCT, in this work, we employ Dimension Ordered Routing (DOR) for the uni-cast and broadcasting traffics, supported with wormhole switching.

***Example:*** Figure on the right shows a deadlock scenario for wormhole switching without use of DOR. Packets A and B are broadcast messages and can only advance together. However, Packet C (a uni-cast packet) is waiting on A and blocking B. Since B is stuck, so is A, hence deadlock occurs.



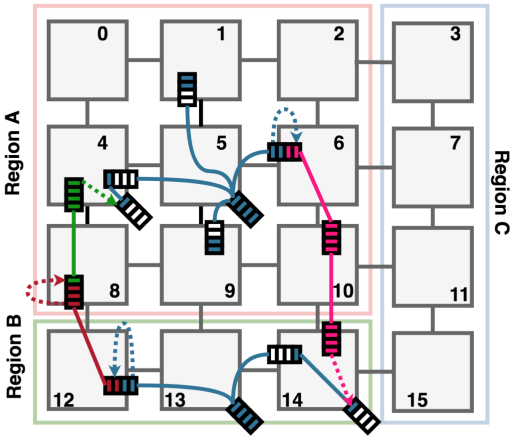




Fig. 4: Example of complex deadlock scenario formation using several regions

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_46

Table 1: Experimental setup details

| | | | |
|---|---|---|---|
| Network Size | 4X4 | NI FIFO depth | 32 flits |
| Routing algorithm | XY | Network traffic packet size | 8 flits |
| Network FIFO depth | 4 flits | Network Traffic PIR | 0.03 |

Second and third assumptions ensure that at each security zone only the security manager has the right to broadcast which avoids race conditions between multiple input ports of the same router.

The forth assumption ensures that the destination buffers for the broadcasting messages are balanced. The imbalance in the buffers would cause a situation where some broadcast flits will advance further while others can get stuck behind other packets which can cause a deadlock situation. This problem is caused by the fact that the normal uni-cast packets are not bounded to the security zones and can be transported among different zones. Fig 4 illustrates an example of such deadlocks due to imbalance of the buffers. The managers of Region A and B (node 5 and node 13) broadcast the reconfiguration packets (marked in Blue). The reconfiguration packets are broadcast packets and can be forwarded at the same time. The other packets (marked in Red and Green) are normal uni-cast packets. Note that the uni-cast packets are not bound by the security zones and can be forwarded according to the base routing algorithm. The dotted arrows show the dependency between the packets. Note the dependency of all reconfiguration packets in each region; The local port of the node 4 is occupied since blue flits in node 4 can not advance due to the flits being stuck in node 6. Similarly the reconfiguration packets in node 14 can not advance due to the flits getting stuck in node 12. Use of VCT, removes this problem entirely and ensures that there is no chance of starvation on the network.

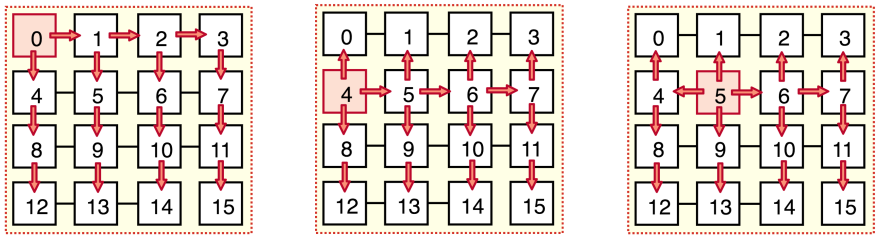
## 4 Experimental Results

In order to evaluate our proposed mechanism, we model the secure MPSoC in VHDL-RTL. We enhanced the Bonfire framework [4] in order to support three additional components: i) the dynamic firewall-based multi-level protection. They are integrated into the NI and are implemented as a LUT-based structure and a comparator; ii) the security router, as described in Fig. 3, a five input wormhole switching router, using FIFO input buffers, LBDR routing mechanism [14], a single switch allocator which controls the five output crossbar switches; and iii) the security manager. The details of the experimental setup are provided in Table 1.

### 4.1 Reconfiguration Latency Evaluation

In order to evaluate the reconfiguration latency of the proposed broadcasting scheme we employed two different scenarios: I) single security zone and a single manager, as in Fig. 5; II) three security zones and three security managers.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_46




(a) $\overline{D}$ : 3 (b) $\overline{D}$ : 2.5 (c) $\overline{D}$ : 1.875

Fig. 5: Different mapping scenarios of the security manager (marked in red) using broadcasting in a single zone

Results are compared to the work of [6]. The first scenario organizes the system in a single security broadcasting zone and a single manager (marked in red). The experiments explore the reconfiguration impact of three different mappings of the security manager node. We use the notation of average distance presented in [6] for distinguishing different manager locations as in Equation 2.

$$\overline{D} = \frac{\sum_{i=0}^{N-1} shortest\_path(ID_M, ID_i)}{N} \quad (2)$$

Where $N$ is the number of nodes in the network, $ID_i$ is the node identifier for the network tile $i$ and $M$ is the $id$ of the tile where the security manager task is mapped.

Fig. 6 depicts the network reconfiguration latency results for different system manager locations and different firewall table size, using broadcast mechanism in a network with a single zone under different traffic patterns (no-traffic, Uniform Random, Hot-Spot and Bit-reversal). In the "No Traffic" scenario, The network nodes do not send packets to the network and only reconfiguration packets are injected. This has been used as a baseline to compare other traffic pattern's behaviour. The results show that different manager placements can affect the dynamic reconfiguration latency. A distance of 3 increases the latency by 11% compared to distance of 1.875 and 2.76% compared to distance of 2.5 on average. In comparison with the work presented in [6], the current approach decreases the reconfiguration latency for different firewall sizes by a factor of 7.5 on average (776 cycles on average compared to 5807 cycles on average in [6]).

Figure 7 describes the effect of the size of the reconfiguration packets in the total network reconfiguration latency for different manager locations and under different load conditions. Similar to [6], the results confirm that larger reconfiguration packets will reduce the configuration latency.

Fig. 8 describes the organization of a $4 \times 4$ mesh-NoC using in the second scenario where three zones and three managers are implemented. Red arrows denote the broadcast path in each zone. Fig. 9 shows the configuration of the connectivity bits in each router under this second scenario (see Fig. 8). The links marked in red are disconnected in order to form the different security

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_46

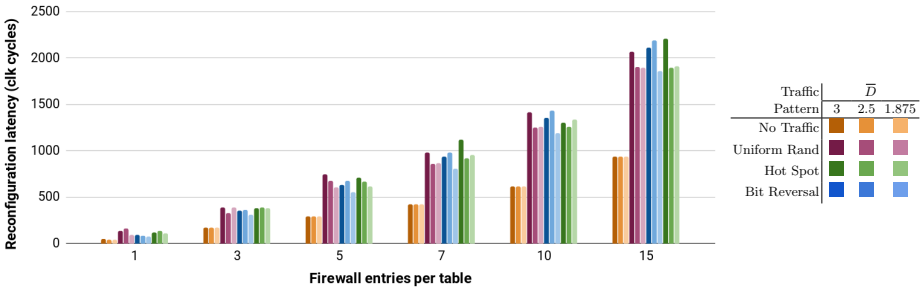



Fig. 6: Network reconfiguration latency for different security manager location and different firewall size under different load conditions (reconfiguration packet carries a single firewall entry).

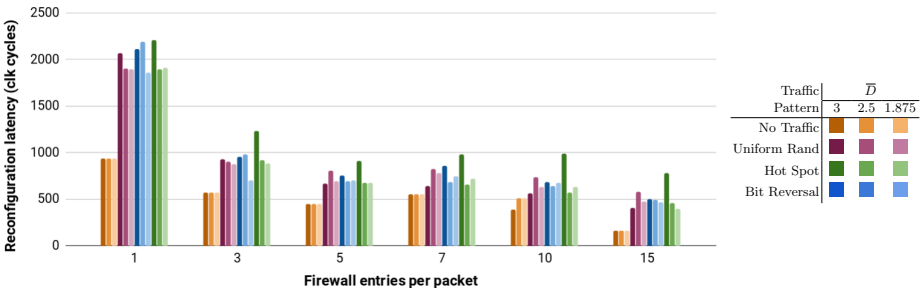



Fig. 7: Network reconfiguration latency for different security manager location and different reconfiguration packet size under different load conditions (firewall has 15 entries in all cases).

zones for broadcast mechanism. Partial reconfiguration (for selective dynamic reconfiguration) of a firewall or smaller firewall size (in case of area constrained embedded MPSoCs) has an impact on the reconfiguration latency. Fig. 10 depicts the network reconfiguration latency for different firewall table sizes, using broadcast mechanism in the second scenario (described in Fig. 8) under different load conditions. In this experiment each reconfiguration packet only updates one firewall location. Another factor that has an impact on the reconfiguration latency is the number of firewall entries that each reconfiguration packet can update. Fig. 11 depicts the network reconfiguration latency for different reconfiguration packet sizes, using broadcast mechanism in scenario II, described in Fig. 8 under different load conditions. The decreasing reconfiguration latency trend is clear with the growing number of firewall entries in per reconfiguration packet. The experimental results show that the proposed approach reconfigure the entire network firewalls on average in approximately 1602 clock cycles based on the reconfiguration policy and the traffic pattern.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_46

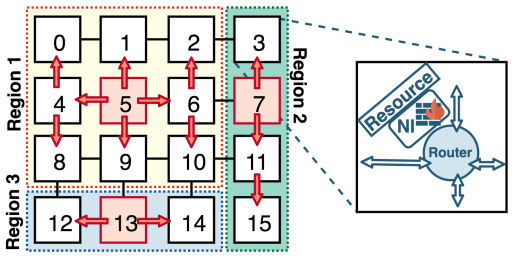



Fig. 8: Scenario II: Organization of different zones with distributed security managers (marked in red).

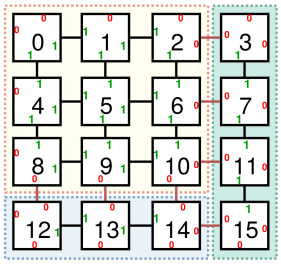



Fig. 9: Connectivity bits ($C_{x\_bc}$) configuration in Scenario II.

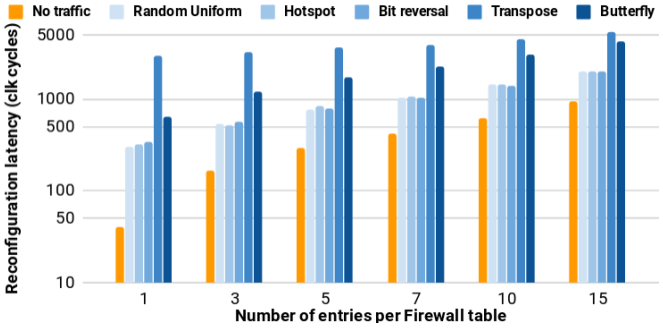



Fig. 10: Reconfiguration latency (scenario II) using different firewall table sizes under different load conditions.

### 4.2 Overhead Evaluation

In order to evaluate the impact of our approach, the baseline router (packet switching) and NI where compared with the security router and firewall-enhanced NI (with 16 entries per NI). These models have been synthesized using AMS $180nm$ CMOS technology standard cell library in Synopsys Design Compiler. Table 2 describes the imposed area overhead on the system using the proposed method. The area overhead on the simple router at hand is less than 11% and for the network interface is completely negligible. However, the baseline router size is very small. Considering a $4mm^2$ chip with 16-core NoC-based MPSoC, the overhead of our approach on the system area would be roughly 0.2%, which is negligible. Table 3 shows the critical path delay overhead which shows that the proposed method does not impact the performance of the original system.

## 5 Conclusions

Dynamic isolation of mixed-critical application is mandatory in current MPSoCs. Security zones can be used to isolate applications. Hardware-based firewalls have

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_46

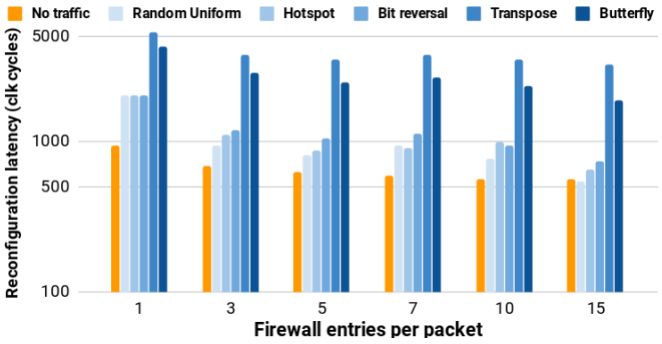



Fig. 11: Reconfiguration latency (scenario II) using different reconfiguration packet size under different load conditions.

Table 2: Area comparison of the proposed method

| | **Area ($\mu m^2$)** | | | **Overhead** |
|---|---|---|---|---|
| | **Combinatorial** | **sequential** | **Total** | **(%)** |
| Baseline Router | 38604.7 | 47388 | 85992.8 | – |
| Proposed Router | 45316.45 | 50097.5 | 95414.0 | 10.9 |
| Baseline NI | 311110.3 | 288852.8 | 599963.2 | – |
| Proposed NI | 311649.4 | 288931.9 | 600581.3 | 0.1 |

Table 3: Critical-path comparison of the proposed method

| | **Critical-Path Delay** ($ns$) | | Overhead |
|---|---|---|---|
| | Baseline | Proposed | (%) |
| Router | 4.85 | 4.81 | – |
| NI | 4.79 | 4.82 | 0.6 |

been proposed as a effective mean for application isolation and security zone creation. However, the efficient reconfiguration of firewall is mandatory. In this paper we have proposed a hybrid communication mechanism that support unicast and broadcast communication to achieve a good performance and efficient firewall update. Our security MPSoC infrastructure is based on a lightweight broadcasting and partitioning mechanism for firewall reconfiguration. Experimental results show that our approach provides a speed up by a factor of 7.5 compared to security updating using uni-casting approaches while imposing negligible area overhead on the system.

**Acknowledgements:** The authors would like to thank Prof. Dr. Ing. Thomas Hollstein for our discussion regarding deadlock-freeness challenges in broadcasting in NoC based systems.

## References

1. Accelerating the Data Plane With the TILE-Mx Manycore Processor.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_46

http://www.tilera.com/files/drim_EZchip_LinleyDataCenterConference_Feb2015_7671.pdf, accessed: 2017-03-13
2. KALRAY MPPA: A New Era of processing. https://de.slideshare.net/infokalray/kalray-sc13-external3, accessed: 2017-03-13
3. Using Intels Single-Chip Cloud Computer (SCC). https://communities.intel.com/docs/DOC-19269, accessed: 2017-03-13
4. Project Bonfire. https://github.com/Project-Bonfire (2016)
5. Achballah, A.B., et al.: FW_IP: A flexible and lightweight hardware firewall for NoC-based systems. In: 2018 International Conference on Advanced Systems and Electric Technologies (IC_ASET) (2018). https://doi.org/10.1109/ASET.2018.8379868
6. Azad, S.P., Niazmand, B., Jervan, G., Sepúlveda, J.: Enabling Secure MPSoC Dynamic Operation through Protected Communication. In: 25th IEEE International Conference on Electronics Circuits and Systems (ICECS). pp. 1–4 (2018)
7. Diguet, J.P., et al.: noc-centric security of reconfigurable soc. In: First International Symposium on Networks-on-Chip (NOCS'07)
8. Evain, S., et al.: From NoC security analysis to design solutions. In: IEEE Workshop on Signal Processing Systems Design and Implementation, 2005. pp. 166–171 (2005). https://doi.org/10.1109/SIPS.2005.1579858
9. Fernandes, R., et al.: A non-intrusive and reconfigurable access control to secure NoCs. In: 2015 IEEE International Conference on Electronics, Circuits, and Systems (ICECS). pp. 316–319 (2015). https://doi.org/10.1109/ICECS.2015.7440312
10. Fiorin, L., et al.: Security Aspects in Networks-on-Chips: Overview and Proposals for Secure Implementations. In: 10th Euromicro Conference on Digital System Design Architectures, Methods and Tools (DSD 2007). pp. 539–542 (2007). https://doi.org/10.1109/DSD.2007.4341520
11. Fiorin, L., et al.: Implementation of a reconfigurable data protection module for NoC-based MPSoCs. In: 2008 IEEE International Symposium on Parallel and Distributed Processing. pp. 1–8 (2008). https://doi.org/10.1109/IPDPS.2008.4536514
12. Fiorin, L., et al.: Secure Memory Accesses on Networks-on-Chip. IEEE Transactions on Computers **57**(9), 1216–1229 (2008). https://doi.org/10.1109/TC.2008.69
13. Fiorin, L., et al.: A Security Monitoring Service for NoCs. In: Proceedings of the 6th IEEE/ACM/IFIP International Conference on Hardware/Software Codesign and System Synthesis. pp. 197–202. CODES+ISSS '08, ACM, New York, NY, USA (2008). https://doi.org/10.1145/1450135.1450180, http://doi.acm.org/10.1145/1450135.1450180
14. Flich, J., Duato, J.: Logic-Based Distributed Routing for NoCs. IEEE Computer Architecture Letters **7**(1), 13–16 (Jan 2008). https://doi.org/10.1109/L-CA.2007.16
15. Grammatikakis, M.D., et al.: High-level security services based on a hardware NoC Firewall module. In: 2015 12th International Workshop on Intelligent Solutions in Embedded Systems (WISES). pp. 73–78 (2015)
16. Grammatikakis, M.D., et al.: Security in MPSoCs: A NoC Firewall and an Evaluation Framework. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems **34**(8), 1344–1357 (2015). https://doi.org/10.1109/TCAD.2015.2448684
17. Hu, Y., , et al.: Automatic ILP-based Firewall Insertion for Secure Application-Specific Networks-on-Chip. In: 2015 Ninth International Workshop on Interconnection Network Architectures: On-Chip, Multi-Chip. pp. 9–12 (2015). https://doi.org/10.1109/INA-OCMC.2015.9

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_46

18. Kocher, P., Horn, J., Fogh, A., , Genkin, D., Gruss, D., Haas, W., Hamburg, M., Lipp, M., Mangard, S., Prescher, T., Schwarz, M., Yarom, Y.: Spectre Attacks: Exploiting Speculative Execution. In: 40th IEEE Symposium on Security and Privacy (S&P'19) (2019)
19. Kornaros, G., Tomoutzoglou, O., Coppola, M.: Hardware-Assisted Security in Electronic Control Units: Secure Automotive Communications by Utilizing One-Time-Programmable Network on Chip and Firewalls. IEEE Micro **38**(5), 63–74 (Sep 2018). https://doi.org/10.1109/MM.2018.053631143
20. Lipp, M., Schwarz, M., Gruss, D., Prescher, T., Haas, W., Fogh, A., Horn, J., Mangard, S., Kocher, P., Genkin, D., Yarom, Y., Hamburg, M.: Meltdown: Reading Kernel Memory from User Space. In: 27th USENIX Security Symposium (USENIX Security 18) (2018)
21. Rodrigo, S., Flich, J., Duato, J., Hummel, M.: Efficient Unicast and Multicast Support for CMPs. In: Proceedings of the 41st Annual IEEE/ACM International Symposium on Microarchitecture. pp. 364–375. MICRO 41, IEEE Computer Society, Washington, DC, USA (2008). https://doi.org/10.1109/MICRO.2008.4771805, https://doi.org/10.1109/MICRO.2008.4771805
22. Sepúlveda, J., Fernandes, R., Marcon, C., Florez, D., Sigl, G.: A security-aware routing implementation for dynamic data protection in zone-based mpsoc. In: 2017 30th Symposium on Integrated Circuits and Systems Design (SBCCI). pp. 59–64 (Aug 2017)
23. Sepúlveda, J., Gogniat, G., Pedraza, C., Pires, R., Chau, W.J., Strum, M.: Hierarchical NoC-based security for MP-SoC dynamic protection. In: 2012 IEEE 3rd Latin American Symposium on Circuits and Systems (LASCAS). pp. 1–4 (2012). https://doi.org/10.1109/LASCAS.2012.6180312
24. Sepúlveda, J., et al.: Efficient and flexible NoC-based group communication for secure MPSoCs. In: 2015 International Conference on ReConFigurable Computing and FPGAs (ReConFig). pp. 1–6 (2015). https://doi.org/10.1109/ReConFig.2015.7393301
25. Sepúlveda, J., Diguet, J.P., Reinbrecht, C.: Security of Emerging Architectures. In: IEEE/ACM International Conference On Computer Aided Design (ICCAD) (2018), https://hal.archives-ouvertes.fr/hal-01911907
26. Sepúlveda, J., et al.: Dynamic NoC-based Architecture for MPSoC Security Implementation. In: Proceedings of the 24th Symposium on Integrated Circuits and Systems Design. pp. 197–202. SBCCI '11, ACM, New York, NY, USA (2011). https://doi.org/10.1145/2020876.2020921, http://doi.acm.org/10.1145/2020876.2020921
27. Sepúlveda, J., et al.: QoSS Hierarchical NoC-based Architecture for MPSoC Dynamic Protection. Int. J. Reconfig. Comput. **2012**, 3:3–3:3 (2012). https://doi.org/10.1155/2012/578363, http://dx.doi.org/10.1155/2012/578363
28. Strano, A., Bertozzi, D., Trivio, F., Snchez, J.L., Alfaro, F.J., Flich, J.: Osr-lite: Fast and deadlock-free noc reconfiguration framework. In: 2012 International Conference on Embedded Computer Systems (SAMOS). pp. 86–95 (July 2012). https://doi.org/10.1109/SAMOS.2012.6404161

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_46